Engineering an Adaptive and Socially-aware Feedback Acquisition

Malik Almaliki

Bournemouth University

A thesis submitted in partial fulfilment of the requirements of Bournemouth University for the degree of Doctor of Philosophy

October 2015

Copyright

Abstract

Self-adaptive software systems are characterised by their ability to monitor changes in their dynamic environment and react to these changes when needed. Adaptation is driven by these changes in the internal state of the system and its external environment. Social Adaptation is a kind of adaptation which gives users' feedback a primary role in shaping adaptation decisions. Social Adaptation infers and employs users' collective judgement on the alternative behaviours of a system as the main driver in tailoring adaptation decision. Users' collective judgement is determined through individual users' feedback collected during the lifetime of the software.

Social Adaptation still lacks systematic and efficient engineering mechanisms of the acquisition process of users' feedback. The goal of this thesis is to devise an engineering method for a systematic and adaptive acquisition of users' feedback. Given the various contextual information which could influence how feedback should be collected from users, this thesis looks at the acquisition process itself as an adaptive process. The goal of such adaptation is to optimize the quality of obtained feedback without affecting users' experience. In order to achieve the goal of this thesis, several empirical studies with software engineering experts and end-users have been conducted. This helped gaining insights into how the role of users' feedback is perceived by software experts and how users behave and react to feedback acquisition. The outcomes of the empirical studies are then exploited to achieve the aim of thesis.

The findings informed by these studies suggest that users' behaviours to feedback acquisition highly varies and an adaptive feedback acquisition is highly needed to cater for differences in behaviours, improve users' satisfaction, feedback quality and software success. To tackle this problem, the concept of Persona is employed to aid software engineers understand the various users' behaviours and improve their ability to design feedback acquisition techniques more efficiently. The personas are developed based on a mixture of the qualitative and quantitative studies conducted throughout this thesis. In addition, this thesis proposes PAFA, a Persona-based method for a systematic design of an Adaptive Feedback Acquisition and reports on its evaluation. Finally, this thesis is also meant to contribute to the knowledge of software engineering community on developing systematic ways for feedback engineering which are hoped to lead to a better quality feedback and maintained users experience.

Contents

Copyright I
Abstract II
Acknowledgment VIII
Publications resulted from this thesis IX
Declaration of co-authors contribution to the published work of this thesis IX
1. Chapter 1: Introduction 1
1.1 Research Aim 4
1.2 Research Questions 5
1.3 Research Objectives 5
1.4 Research Methodology 7
1.5 Thesis Structure 7
1.6 Summary 9
2. Chapter 2: Literature Review 11
2.1 Social Adaptation and Related Tobics 12
2.1.1 User Modelling 12
2.1.2 User-Centred Design (UCD) & Participatory Design (PD) 13
2.1.3 End-User Computing 13
2.1.4 Recommendation Systems & Collaborative filtering 14
2.1.5 Software Product line (SPL) And Dynamic Software Product Line (DSPL) 14
2.1.6 Awareness Requirements for adaptive systems 15
2.2 Users' involvement and feedback in socially adaptive software 16
2.2.1 The role of user's involvement in the software lifecycle 17
2.2.2 Incentives and rewards for Empowering user involvement 18
2.2.3 Software-Obtained End-user feedback 19
2.2.4 Feedback Gathering tools & classifications 20
2.3 Personas and Feedback Acquisition 26
2.3.1 Scenarios and Personas 27
2.3.2 Contextual Design, Ethnography and Personas 27
2.3.3 Participatory Design and Personas 28
2.3.4 Efficiency of Fiction in Personas 28
2.3.5 Theory of Mind and Personas 29
2.4 Personality types and Socially-aware Feedback Acquisition 29
2.4.1 Extraversion and Introversion 30
2.4.2 Sensing/Intuition 30
2.4.3 Thinking/Feeling 31
2.4.4 Judging/Perceiving 32
2.5 Requirement Engineering techniques and Feedback Acquisition 32

2.5.1 Goal Modelling and feedback acquisition 33
2.5.2 Use Cases and Feedback Acquisition 34
2.6 Summary 34
3. Chapter3: Social Adaptation and feedback acquisition foundations &challenges 35
3.1 Introduction 35
3.2 Research Method (Expert Opionon) 36
3.2.1 Expert Selection 36
3.2.2 Design, Test and Distribution of the Survey 37
3.3 First Phase Results 38
3.3.1 Social Adaptation Benefits and Value 38
3.3.2 Implementation Choices (Autonomy and Feedback Acquisition) 43
3.4 Second Phase result 47
3.5 Threats to Validity 49
3.6 Summary 49
4. Chapter 4: Users' Behaviours to Feedback Acquisition (A mixed method study) 50
4.1 Introduction 50
4.2 Research Method (Mixed Method Paradigm) 51
4.2.1 Major types of mixed methods design 52
4.3 Qualitative Phase Design 54
4.3.1 Interview Design 54
4.3.2 Sampling 56
4.3.3 Analysis 56
4.4 Qualitative Findings 56
4.4.1 Explicit Feedback Advantages 58
4.4.2 Motivations for Feedback Provision 58
4.4.3 Feedback Acquisition Methods 60
4.4.4 Pause of Feedback Requests 60
4.4.5 Timing for Feedback requests 61
4.4.6 Feedback Visibility 61
4.5 Quantitiative Phase Design 62
4.5.1 Sampling 62
4.5.2 Analysis 63
4.6 Quantitative Findings 63
4.6.1 Feedback Acquisition Methods and Feedback Types 64
4.6.2 Motivations for Accepting/Ignoring Feedback Requests 66
4.6.3 Users' Clusters 70
4.7 Discussion 73

4.8 Related Studies 74
4.9 Summary 75
5. Chapter 5: Culture and users' behaviour to feedback Acquisition 77
5.1 Introduction 77
5.2 Research Methodology 79
5.2.1 Focus Group Design 80
5.2.2 Sampling 80
5.3 Analysis 82
5.4 Findings 82
5.4.1 Visibility and Similarity of Others Feedback 83
5.4.2 Volume of Already Given Feedback 86
5.4.3 Social recognition 87
5.4.4 Feedback acquisition as a social activity 89
5.5 Summary and related work 90
6. Chapter 6: Persona-based Adaptive Feedback Acquisition 92
6.1 Introduction 92
6.2 Persona and feedback acquisition 93
6.3 Benefits of Personas 94
6.4 Personas Creation 95
6.4.1 Conducting a qualitative research 96
6.4.2 Form hypotheses, foundations and ideas for further investigation 96
6.4.3 Investigate the formed hypotheses, foundations and ideas quantitatively .. 96
6.4.4 Segment users based on statistical cluster analysis 96
6.4.5 Create a persona for each segment (in collaboration with domain experts) 96
6.4.6 Domain Experts Involvement 97
6.5 Expert interview design 98
6.6 Persona representation and building components 98
6.7 Developed Personas 100
6.8 Personas benefits for feedback acquisition Design 105
6.8.1 Engagement 105
6.8.2 Discussion 105
6.8.3 More efficiency in identifying requirements 106
6.8.4 Release your thinking from your own mental model 106
6.8.5 Validation 106
6.9 Discussion 107
6.10 Summary 108
7. Chapter 7: PAFA- A Design Method for Adaptive Feedback Acquisition 109
7.1 First phase: Personas to Scenarios 111
7.2 Second phase: Scenarios to Goal modelling 112

7.3 Third Phase: Goal model to use-cases 114

7.3.1 Strategic Dependency model (SD) 115

7.3.2 Strategic Rationale model (SR) 115

7.4 Fourth phase: Adaptation Engineering 116

7.4.1 Variability Modelling Using Software Product Lines Techniques 117

7.4.2 Users direct input 121

7.4.3 Personality questions 123

7.4.4 Social Adaptation 124

7.5 Final phase: Evolvement 124

7.6 Summary 124

8. Chapter 8: PAFA Evaluation 126

8.1 Introduction 126

8.2 Study Design 128

8.2.1 Aim and case study scenarios 129

8.3 Data collection and procedure 134

8.3.1 Questionnaire 135

8.3.2 GQM 136

8.3.3 Goal, Questions and Metrics 137

8.3.4 Observation 139

8.4 Participants selection 141

8.5 Analysis 143

8.6 Findings 143

8.6.1 PAFA Maintainability (*Goal 1: how PAFA FACETS CAN be reasonably updated with any changes or enhancements*) 143

8.6.2 PAFA Completeness *(Goal 2: the extent to which the components of PAFA cover the holistic design of feedback acquisition)* 144

8.6.3 PAFA Reliability and Integrability (Goal 3 and 8) 144

8.6.4 PAFA Portability (*Goal 5: the extent to which PAFA can be adopted in other application domains)* 149

8.6.5 PAFA Maturity *(Goal 7: the extent in which enough literature is readily available about PAFA's components)* 150

8.6.6 PAFA Efficiency (Goal 4: the amount of effort, resources and expertise required by PAFA if adopted.) 150

8.6.7 PAFA Usability *(Goal 6: measuring the ease to learn, interpret and use PAFA)* 154

8.7 Client Confirmatory Phase 158

8.7.1 Study design and participants selection 158

8.7.2 Findings 158

8.8 Lessons learned 161

8.9 Threats to validity 163

8.10 Summary.....163
9. Chapter 9: Discussion, Future work and Conclusion..... 164
9.1 PAFA applications.....164
9.1.1 PAFA and Feedback Acquisition.....164
9.1.2 Social Adaptation and PAFA.....165
9.2 Thesis Contributions.....165
9.2.1 Identification of the engineering challenges for socially adaptive systems165
9.2.2 Knowledge base of users' behaviour to feedback acquisition 166
9.2.3 Investigation of the role of culture on users' behaviour to feedback 166
9.2.4 Building constructs of personas of users' behaviour to feedback.....166
9.2.5 The development of a systematic and adaptive feedback acquisition 166
9.2.6 Bridging the gap between software engineering and user experience.....167
9.3 Future work.....167
9.4 Summary.....169
10. References..... 172
11. APPENDICES 185
11.1 Appendix 1 185
11.2 Appendix 2 197
11.3 Appendix 3 199
11.4 Appendix 4 208
11.5 Appendix 5 211
11.6 Appendix 6 215
11.7 Appendix 7 226

Acknowledgment

First of all, I would like to express my sincere gratitude to my first supervisor Dr. Raian Ali. Raian’s experience, hard work and enthusiasm have made this PhD an enjoyable and feasible journey. Raian has also been extremely encouraging and energizing me throughout my PhD journey. He always managed to shed the light on any challenge that has been encountered throughout this journey whether it was related to my PhD or even my personal life since he was also there as a friend. This made me see and deal with things in a clearer and better way. His positive and optimistic attitude has encouraged and inspired me throughout these last three years. I would also like to thank my second supervisor Dr. Cornelius Ncube for his helpful feedback during certain stages of this PhD.

I am also very grateful to the members of the Computing and Informatics Department at Bournemouth University for making this journey a pleasant one. Thank you all for your friendship, collaboration, support and interest in my work. Throughout my PhD, I have been fortunate to have the support from many friends and experienced researchers who have inspired me, provided me with support, wisdom, encouragement and guidance. In particular, I would like to thank Prof. Keith Phalp for his support and it has been a privilege to have the benefit of his counsel. In addition, I’m grateful to the funding bodies of my PhD which are the Saudi Ministry of Education and Taibah University, KSA. Their fund, help, supervision and guidance have absolutely made this PhD feasible. I would also like to acknowledge the Graduate School at Bournemouth University for the financial, educational, and research support.

I’m also grateful to the participants who took part in this thesis for their effort, time and valuable input that helped me achieving the aim of this PhD. Additionally, without the support and encouragement of my family members and my parents in particular, I couldn’t have accomplished this journey. Thank you very much for helping me find the strength to keep moving forward and reaching for my goals. I could not have done this without your endless love and support. Finally, I really hope that this work can be beneficial to people who share the same interest as this thesis and the Software Engineering community in general.

Publications resulted from this thesis

- Almaliki, M., Ncube, C. and Ali, R., 2015. Adaptive Software-based Feedback Acquisition: A Persona-based Design. *In: The IEEE Ninth International Conference on Research Challenges in Information Science (RCIS'15)* 13-15 May 2015 Athens, Greece (see Chapters 6, 7).
- Almaliki, M., Jiang, N., Ali, R. and Dalpiaz, F., 2014. Gamified Culture-aware Feedback Acquisition. *In: The 2nd International Workshop on Crowdsourcing and Gamification in the Cloud (CGCloud 2014), Co-located with UCC 2014.* 8 December 2014 London, UK (see Chapter 5).
- Almaliki, M., Ncube, C. and Ali, R., 2014. The Design of Adaptive Acquisition of Users Feedback: an Empirical Study. *In: The IEEE Eighth International Conference on Research Challenges in Information Science (RCIS 2014).* 28-30 May 2014 Marrakesh, Morocco (see Chapter 4).
- Almaliki, M., Faniyi, F., Bahsoon, R., Phalp, K. and Ali, R., 2014. Requirements-driven Social Adaptation: Expert Survey. *In: The 20th International Working Conference on Requirements Engineering: Foundation for Software Quality (REFSQ 2014)* 7-10 April 2014 Essen, Germnay (see Chapter 3).
- Hosseini, M., Moore, J., Almaliki, M., Shahri, A., Phalp, K. and Ali, R., 2014. Wisdom of the Crowd within enterprises: Practices and challenges. *Computer Networks. Special Issue on Crowdsourcing, Volume 90, 29 October 2015, Pages 121–132*[1].
- Almaliki, M. and Ali, R., 2016. Persuasive and Culture-aware Feedback Acquisition. *In: The 11th International Conference on Persuasive Technology. PERSUASIVE 2016 (PT-16).* 5-7 April 2016 Salzburg, Austria (see Chapter 5).

Declaration of co-authors contribution to the published work of this thesis

The author of this thesis was the first author of all the resulted publications of this thesis. The contribution of the first author was as follows:

- Forming and articulating the idea and aim of each paper.
- Deciding upon the appropriate methodology to be adopted in each paper (e.g. Mixed Method Design).
- Designing and implementing the empirical studies presented in each paper (e.g. developing interview scripts, recruiting the participants, collecting the data, etc).
- Analysing and interpreting the collected data and draw the conclusions (e.g statistical analysis, qualitative and quantitative analysis, etc).

[1] This particular publication was encouraged and informed by the initial results of the studies of this thesis on the topic the paper introduces. The author of this thesis participated in the design of the study protocol and the refinement and reporting of the results discussed in the paper.

- Reporting the findings and fully writing each paper.

The co-authors contributed to the published papers in terms of verifying and validating the studies' findings by comparing them against the actual responses from the participants. They also provided guidance and feedback on the structure and the overall articulation of the papers' message. In addition, they gave insights on the methodology and also checked the writing quality and suggest modifications on some parts of the text. Furthermore, the co-authors enriched the papers with the appropriate terminologies in certain places especially those related to the venue where the papers were published.

1. CHAPTER 1: INTRODUCTION

The uncertainty of the environment in which software systems could be used has increased the complexity of software engineering mission. In paradigms, such as cloud computing and context-aware systems, engineers need to develop software which will be eventually working in new unknown environment and this could compromise the goal for which the system is designed. This has put pressure on the software engineering community to start looking for inspiration in other related fields (e.g., robotics, artificial intelligence and control theory) for new ways for designing and managing systems and services (Brun et al., 2012). The engineering of complex software systems which incorporate an autonomous ability to reconfigure themselves in response to changes in their environment, called self-adaptation, has become a need (Jeary et al. 2011) and a main goal for the software engineering community.

A self-adaptive software system can be described as a closed-loop system that depends on a feedback control loop to adjust itself in response to changes at runtime (Cheng et al. 2009; Salehie and Tahvildari 2009). In such systems the feedback coming from the monitor component of the system could indicate changes which make the current software operation wrong or not the most suitable operation and, thus, make it necessary to take an adaptation action. The changes can be internal within the boundary of software system itself e.g., failure, or external context changes e.g., increasing requests from users.

Self-adaptive software is intended to monitor itself and its context at runtime. It detects changes, analyses them, decide how to react and act to execute such decisions. The software system responses to these changes are to satisfy or maintain the satisfaction of certain requirements. An example of self- adaptive software could be a web service that, in order to keep running for a long time, needs to monitor the current number of users and decide upon the processing power and the storage space to provide. This can be done by collecting information about its current users, analysing this information in order to detect failures or problems, deciding on how to solve these problems and acting to apply those made decisions.

There are four common properties of self-adaptive systems called self-*:

- The first property is called self-protection which provides the ability to monitor security breaches and act to prevent or recover from their effects.
- The second property is called self-optimization which makes the system able to manage the resources availability to satisfy the requirements.
- The third property is called self-healing that provides the ability to monitor faults that have occurred or could occur and heal them or prevent failures.
- Finally, self-configuration which means the ability of the system to reconfigure itself in order to adapt to changes by installing, updating, integrating, or/ and

composing/decomposing software entities (e.g. installing updates) (Salehie and Tahvildari 2009).

The main motivation of self-adaptive systems is to improve the satisfaction of users' requirements and this naturally requires autonomous adaptation to changes of its internal state and its environment. Self-adaptability is highly dependent on feedback that is provided by the systems itself, namely the monitor component, about its current state and its environment. This feedback is the driver for triggering and taking autonomous adaptation actions. However, recent research (Ali et al. 2011b; Ali et al. 2012) observed that there are some adaptation drivers which cannot be monitored by only automated means. An example of these drivers is the users' judgments on the different behaviours of the system in terms of their quality and validity. Users' judgments enable the system to know how its adaptation is being evaluated and accepted by them so that it improves over time. Giving users' a voice in tailoring adaptation actions and enabling them to give feedback and express their opinions of software behaviour quality is called "Social Adaptation". This kind of adaptation will be the focus of this thesis.

In (Ali et al. 2012), Social Adaptation is defined as "the system autonomous ability to analyse users' feedback and choose upon an alternative behaviour which is collectively shown to be the best for meeting requirements in a context" . In Social adaptation, users' feedback empowers the system ability to evaluate each of its alternative behaviours at runtime. The software behaviour that is shown by users to be correct and more efficient in a certain context of use will be adopted when that context occurs. Social Adaptation claims to have the benefit of improving the transparency of the self-adaptive system and raising users' trust in it, since users are treated as first-class entities in both the engineering and also the operation of such systems. In fact, over time of using the software, users may be able to shape the decision-making process in a way that can only be done by today's experts. Figure 1 outlines the process of Social Adaptation.

Some researchers have pursued similar visions under themes such as requirement aware self-adaptive systems (Adomavicius and Tuzhilin 2005), requirement monitoring at run-time (Fickas and Feather 1995), and Social Adaptation in pervasive software systems (Esfahani and Malek 2010). All these efforts adhere to a notion of representing users' requirements or trust relationship among users as run-time objects that can be used by the system to reason about the adaptation process. Other researchers use the term of socially-adaptive software differently to refer to software agents which are socially adaptive in the sense of their ability to comply to social norms, e.g. (van Riemsdijk 2013). Social Adaptation, as described in (Ali et al. 2012), is unique in the sense that instead of catering to the requirement of a user or subset of users, it harnesses the "wisdom of the crowd" to adapt the system in a way that is deemed best by end-users' collective judgement rather than the decisions of an elite group of users or those of developers. To put it another way, Social Adaptation pursues the goal of a democratic-like, consensus-based social approach to adapting software systems to meet users' requirements.

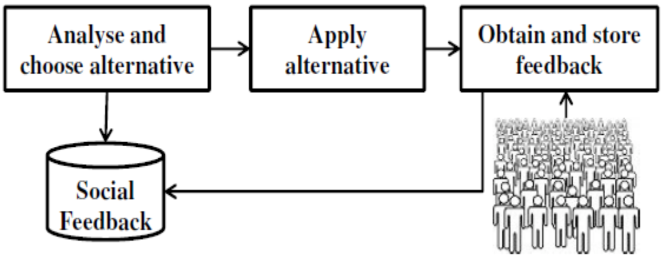


FIGURE 1SOCIAL ADAPTATION LOOPS (ALI ET AL. 2012).

As explained earlier, the feedback loop in self-adaptive systems plays a significant role in enabling self-adaptive systems. It is responsible for collecting information about the software internal states and its context, analysing this information and making decision about the software, e.g. installing a new update. The Adaptation loop is consisted of four different parts (*Collect*, *Analysis*, *Decide* and *Act*) and each one of them has its own role that contributes to the system adaption process (Salehie and Tahvildari 2009).

- The *Collect* part is responsible for collecting, correlating data about the system and its environment and transferring them to the analysis part.
- The *Analysis* is responsible for analysing the data provided by the collect activity and the history of the system, in order to detect when a change (response) is required.
- The *Decide* function decides on what should be changed and how to change it to achieve the best alternative
- The *Act* is responsible for applying the actions determined by the deciding process. Figure 2, shows the four activities in the adaptation loop.

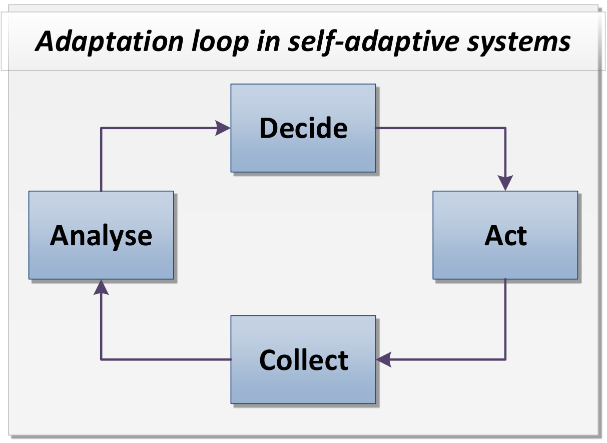


FIGURE 2 ADAPTATION LOOP IN SELF-ADAPTIVE SYSTEMS

Although feedback control loops represent the heart of any self-adaptive software system and must become first-class entities to a first-class entity in engineering of self-adaptive systems (Nzekwa et al. 2010), it is observed that control loops are often hidden or abstracted when presenting the architecture of self-adaptive systems (Brun et al. 2009; Cheng et al. 2009). Understanding and reasoning about the control loops in self-adaptive systems is an important step to be taken in order to improve the engineering approaches of self-adaptive systems from an ad-hoc, trial-and-error approach to a more systematic approach (Cheng et al. 2009).

The Collect activity in the adaptation loop plays a vital role since all decisions that are made on the system depend on this activity. However, the engineering of the activity is still facing a range of issues. Quality and amount of collected feedback are examples of these issues (What is the required sample rate? How reliable is the sensor data? Is there a common event format across sensors?) (Brun et al. 2009).

In Social Adaptation, users work as monitors. This introduces a range of challenges for the engineering of this human-based monitor. Examples of such challenges include: the specification of the type of users who are capable and should be authorized to provide feedback, the specification of the suitable time to ask for feedback and, also, to stop asking for feedback. In addition, social and cultural differences between users introduce another range of challenges for the engineering of feedback acquisition since users' behaviour and reaction to feedback acquisition could be highly influenced by their cultural backgrounds (i.e. how could social interactions among users with different cultural backgrounds affect their willingness to give feedback?). Cultural differences among users play a noticeable role in the software success (Reinecke and Bernstein 2007). Software systems that are successful in one culture may not be successful in another (Honold 2000; Nocera 2014; Sturm et al. 2015) therefore they need to be adaptive to fit cultural specifics (Nielsen 1996). Feedback acquisition also needs to be adaptive to fit cultural differences among users in order to empower software's success.

Reviewing the literature, this thesis could not identify systematic approaches for engineering feedback acquisition. The impact of users' feedback and how users behave when providing feedback is still ambiguous as discussed in (Pagano and Maalej 2013). The lack of engineering processes for feedback acquisition would lead to poorly designed feedback collection mechanisms and this could harm the quality of collected feedback, users' experience and the software success and evolution decisions (Pagano and Bruegge 2013). Owing to its importance, novel engineering approaches are needed to better conduct this activity, the acquisition of feedback from users. The quality and trustworthiness of feedback will improve the quality of Social Adaptation and thus the software.

1.1 RESEARCH AIM

In the light of the above mentioned challenges and the lack of engineering approaches for users' feedback acquisition in software applications, the aim of this thesis is to devise an engineering method for a systematic and adaptive acquisition of users' feedback. Given the various contextual information which could influence how feedback should be collected from users, this thesis looks at the acquisition process itself as an adaptive process. The goal of such adaptation is to optimize the quality of obtained feedback, response rate, users' satisfaction and ultimately the software success. This is studied in the context of socially-adaptive systems and scooped to users' feedback collected on the quality and validity of software behaviour.

1.2 RESEARCH QUESTIONS

Based on the aim of this thesis the following questions were derived:

- Q1: From the perspective of both software developers and end-users, what are the engineering foundations and challenges for empowering the role of users' feedback in software systems?
- Q2: How do users differ in their behaviours and what is the effect of cultural background with respect to feedback acquisition?
- Q3: How to develop a socially-aware and adaptive feedback acquisition and how can it impact the quality of collected feedback, response rate and users' satisfaction?

1.3 RESEARCH OBJECTIVES

In order to achieve the aim of this thesis, the research will be conducted to reach the following objectives:

- **Objective 1: Identifying the Engineering Foundations and Challenges of Users' Feedback Acquisition**

As a first step in achieving the goal of this thesis, one needs to understand and identify the motivations and the main challenges in the area of feedback acquisition for socially-adaptive systems. An Expert Opinion was conducted to gather and analyse the knowledge of experts in a wide range of related areas with Requirement Engineering and Adaptive Systems as the main areas. This includes areas such as HCI, Human Factors in Computing, Psychology, Privacy and Security Engineering, Socio-Technical Systems Engineering and Social Computing. This study aimed at identifying software engineering foundations and challenges in the area of users' feedback acquisition. The reason for conducting an Expert Opinion was the noticeable lack of research on engineering feedback acquisition and the role of such feedback in shaping and validating software runtime adaptation decisions. The survey provides software engineers with insights and challenges to consider when engineering the acquisition of users' feedback and demonstrate the acceptability of this thesis from industrial and research perspective (see Chapter 3).

- **Objective 2: Understanding of Users' Behaviours with regard to Feedback Acquisition**

Since users play a vital role in shaping the adaptation process in socially-adaptive systems, there is a need to study and understand their behaviours in relation to feedback acquisition in socially-adaptive software. An empirical study following a Mixed Method Approach (Sequential Exploratory Design) is conducted to better understand users' perspectives and behaviours to feedback acquisition. The findings provide software engineers with a knowledge base of users' different behaviours and perceptions in relation to feedback acquisition to be considered when engineering users' feedback acquisition (see Chapter 4).

- **Objective 3: Studying the Cultural Impact and Social Factors in Motivating Users to Give Feedback**

Based on the outcome of Objective 2, the variant cultural backgrounds of users initially indicate a noticeable impact on how they are socially motivated to give feedback. For example, users who have a Middle Eastern background can be more motivated to give feedback by different social factors (e.g. visibility of others feedback) than users with a western background. In order to further investigate this cultural impact, a follow up qualitative study to the previous mixed method study is conducted on users coming from two different cultural backgrounds (Middle Eastern users and European users). The objective was to discover whether there is actually a socio-cultural effect on the users' acceptance and behaviours with regard to feedback acquisition. This study is conducted through four focus groups to investigate the core social factors that influence their willingness to give feedback and how these factors culturally affect them. The outcome of this objective can help in developing socially-aware feedback acquisition that can adapt to different types of users in terms of their cultural backgrounds in order to maximize feedback quality, users' satisfactions and motivations to give feedback (see Chapter 5).

- **Objective 4: Representing Users' Different Behaviour Using the Concept of Persona**

Based on the findings gathered from Objective 2 and 3, a set of building constructs for the development of personas to represent users' behaviour to feedback acquisition are proposed. In addition, a set of personas are also introduced to represent users' different behaviours. The personas will aid software engineers to better understand these variations in behaviour and improve their ability to design feedback acquisition techniques more efficiently. On the other side, the personas building constructs/components will help software engineers developing and building new persona of users' behaviour to feedback when needed or preferred (e.g. in special contexts) (see Chapter 6).

- **Objective 5: Designing and Evaluating a Persona-based Adaptive Feedback Acquisition**

The findings of **Objectives 1, 2, 3 and 4** are exploited to devise a software engineering method that allow for a systematic conduct of an adaptive and socially-aware feedback acquisition of users' feedback. The method allows engineers to produce an adaptive acquisition that is able to cater for different types of users. E.g. A certain type of users might be only willing to provide feedback in certain social settings therefore the adaptive feedback acquisition should be able to adapt itself to that type of user when asking for feedback.

To this end, this thesis provides software engineers with a systematic method for an adaptive/socially-aware acquisition of users' feedback to improve the design phase of such an activity. The resulting method PAFA (**P**ersona-based **A**daptive **F**eedback **A**cquisition) serves as a guide for software engineers in the development phase of the feedback acquisition process.

PAFA is validated by applying it in practice following a case study approach. Three real case study scenarios with regard to feedback acquisition were given to software engineers. They were then asked to use PAFA to systematically develop an adaptive feedback acquisition for these scenarios. A follow up phase with clients was also conducted to review the method from an industrial point of view. The findings of this evaluation helped us to discover the weak points in the method and refine it (see Chapters 7, 8).

1.4RESEARCH METHODOLOGY

In order to achieve the aim of this thesis, an empirical software engineering approach is followed. Empirical software engineering is a recent discipline that attempts to positively affect the practice of software engineering by comparing theory to reality and to move towards well-founded decisions to drive the software development process. The methods adopted in empirical software engineering are highly inspired by social sciences and they lead to the creation of theories or frameworks that explain what the researcher observes and measures (Easterbrook et al. 2008). Based on the formulated theories, one can introduce evidence-based changes that are grounded in scientific research to the development process of software to empower its success. Therefore, a mixture of several empirical studies (involving both quantitative and qualitative methods) with software experts, industrial experts and users were adopted to gain a better understanding of users' behaviour expert's point of view in several contexts with regard to feedback acquisition. In addition, a set of software engineering methods were used in the solution space to integrate the findings of the empirical studies as discussed in Chapter 7. In the evaluation stage, a case study approach was employed involving multiple sessions and scenarios (see Chapter 8). Further explanation is presented about the research methodologies adopted to fulfil the objectives of this thesis and the rationale behind their adoption in the next chapters.

1.5THESIS STRUCTURE

This thesis is structured as follows. Chapter 2 presents a literature review of topics related to the context of this thesis. In Chapter 3 an Expert Survey study is presented aiming at identifying the motivations and the main challenges in the area of feedback acquisition for socially-adaptive systems. Chapter 4 reports on a mixed method study conducted to better understand users' behaviours with regard to feedback acquisition. Chapter 5 follows up with the findings of Chapter 4 and investigates the role of culture on users' behaviour to feedback. In Chapter 6 the concept of persona is introduced and adopted to integrate and represent the users' behaviours discussed in Chapter 4 and 5. Chapter 7 introduces a method of the design of feedback acquisition called a Persona-based Adaptive Feedback Acquisition method (PAFA). Chapter 8 evaluates and report on PAFA whereas Chapter 9 gives a discussion on PAFA, summary of contribution and future work. Figure 3 shows a roadmap of the thesis.

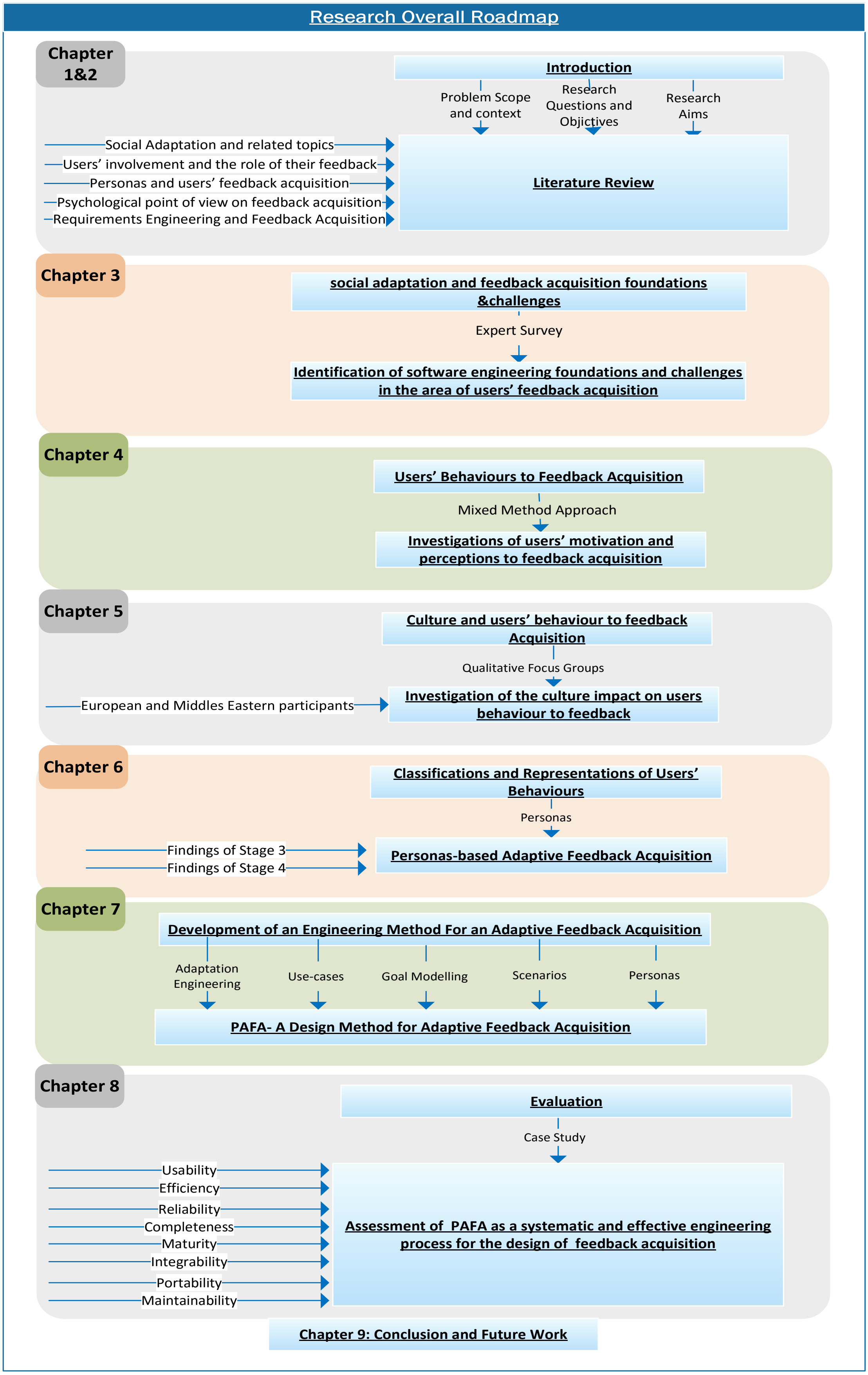

Research Overall Roadmap
Chapter 1&2
Introduction
Problem Scope and context
Research Questions and Objictives
Research Aims
Social Adaptation and related topics
Users' involvement and the role of their feedback
Personas and users' feedback acquisition
Psychological point of view on feedback acquisition
Requirements Engineering and Feedback Acquisition
Literature Review
Chapter 3
social adaptation and feedback acquisition foundations &challenges
Expert Survey
Identification of software engineering foundations and challenges in the area of users' feedback acquisition
Chapter 4
Users' Behaviours to Feedback Acquisition
Mixed Method Approach
Investigations of users' motivation and perceptions to feedback acquisition
Chapter 5
Culture and users' behaviour to feedback Acquisition
Qualitative Focus Groups
European and Middles Eastern participants
Investigation of the culture impact on users behaviour to feedback
Chapter 6
Classifications and Representations of Users' Behaviours
Personas
Findings of Stage 3
Findings of Stage 4
Personas-based Adaptive Feedback Acquisition
Chapter 7
Development of an Engineering Method For an Adaptive Feedback Acquisition
Adaptation Engineering
Use-cases
Goal Modelling
Scenarios
Personas
PAFA- A Design Method for Adaptive Feedback Acquisition
Chapter 8
Evaluation
Case Study
Usability
Efficiency
Reliability
Completeness
Maturity
Integrability
Portability
Maintainability
Assessment of PAFA as a systematic and effective engineering process for the design of feedback acquisition
Chapter 9: Conclusion and Future Work

FIGURE 3 ROADMAP OF THE THESIS.

1.6 SUMMARY

This chapter gave an introduction to the context and domain of this thesis, discussed the rational for this thesis. It also introduced the aim, research questions, objectives and scoop of this thesis. In addition, this chapter gave a brief overview of the methodology adopted throughout the thesis, the structure and roadmap of the thesis. In the next chapter, a review of the up to date research topics and domains related to Social Adaptation and the acquisition of users' feedback in software applications is presented.

2. CHAPTER 2: LITERATURE REVIEW

This chapter presents a review of the up to date research topics and domains related to Social Adaptation and the acquisition of users' feedback in software applications. This review covers two parts as follows:

- **Problem space:** this part gives a review of topics related to the problem domain of this thesis. In general, **Section 2.1** givers a review of topics that share similar visions to Social Adaptation which is the context of this thesis. In addition, **Section 2.2** gives a review of topics related to the role of users' involvement in software development through their feedback which is the main element discussed throughout this thesis.
- **Solution space:** this part gives a review of literature that constructs potential solutions to the problem domain of this thesis. Since feedback acquisition is a multidisciplinary topic (i.e. it is not a Software Engineering issue but it also studies users' behaviour and physiological views to feedback acquisition), this part of the literature is built with this aspect in mind. In general, **Section 2.3** gives a review of the concept of Personas and how it could potentially impact the solution space of this thesis. In addition, **Section 2.4** looks at the problem of this thesis and contributes to the solution space from a physiological point of view. Finally, **Section 2.5** gives a review of the potential Software Engineering approaches that can impact the outcome of this thesis. Figure 4 visualises the structure of this thesis's literature review.

This literature review is hoped to formulate areas of research related to socially-adaptive systems with clear boundaries for this thesis and for the other researchers who share with this thesis the vision of socially-adaptive systems and the need to engineer the acquisition of users' feedback. Figure 4 gives a brief overview of some of the topics covered in this thesis's literature review.

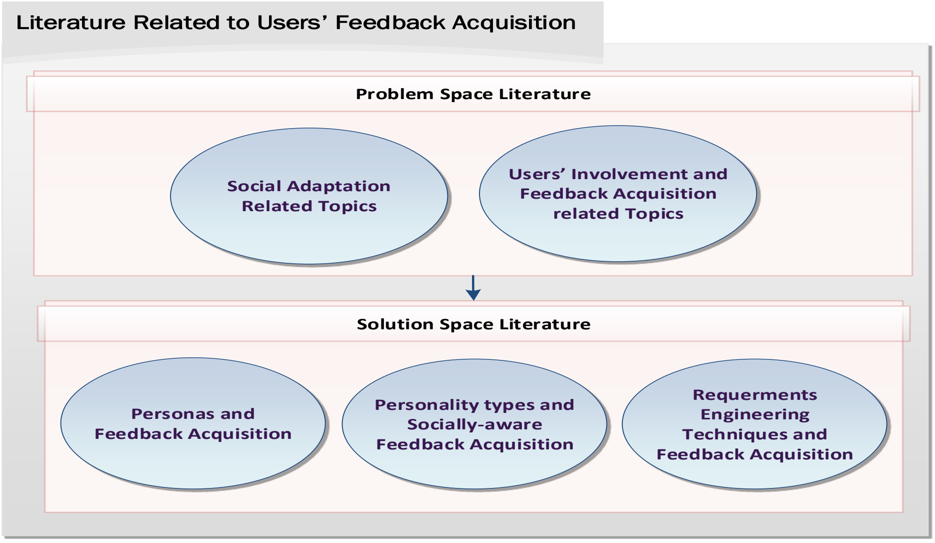

Literature Related to Users' Feedback Acquisition
Problem Space Literature
Social Adaptation Related Topics
Users' Involvement and Feedback Acquisition related Topics
Solution Space Literature
Personas and Feedback Acquisition
Personality types and Socially-aware Feedback Acquisition
Requerments Engineering Techniques and Feedback Acquisition

FIGURE 4 VISUAL VIEW OF THE MAIN TOPICS COVERED IN THIS THESIS'S LITERATURE REVIEW

2.1SOCIAL ADAPTATION AND RELATED TOBICS

Social Adaptation (as discussed in Chapter 3) has shown to be a useful concept for developers and clients of adaptive software and offers them valuable benefits. The ultimate goal of Social Adaptation is to satisfy users' requirements efficiently by enabling users to steer and tailor the adaptation process. Users are enabled to provide feedback on the validity and the quality of the software alternatives and the system will analyse users feedback and then adapt accordingly (Ali et al. 2012).

Social Adaptation may disrupt the current development paradigm of self-adaptive systems, in terms of time to market or deploy, by reducing the upfront effort in design phase to the barest minimum. By taking the socialized view of adaptation, the system will only provide a platform for users to express their preferences, whilst design decisions are collectively made by users at run-time. This indeed makes Social Adaptation different from other approaches (e.g. agile software development) in which the variability points of the software and execution environment will be learnt at run-time based on feedback provided by users. This is a realistic assessment since users of today's software system vary widely in their preferences (perhaps, influenced by culture, norms, age group, location etc). Designers of adaptive systems will focus on engineering open, configurable, and extensible platforms, instead of debating functionality choices. In the following section a review of the literature is provided on common topics that are related to the concept of Social Adaptation in empowering requirement engineering approaches in software systems.

2.1.1 USER MODELLING

User modelling is the process of developing and modifying a user model. The objective of user modelling is to empower software systems with the ability to adapt to the user's specific needs and preferences (Fischer 2001). The system should "say the 'right' thing at the 'right' time in the 'right' way". A user model represents a collection of personal information related to specific users. It represents the basis for any adaptive actions to the system's behaviour.

The data to be included in the model depends on the purpose of the system. It can include personal information about users such as names, ages, interests, knowledge, preferences, users' behaviour, etc. There are several design patterns for user models. Dynamic user models are one of the common design patterns for user models that allow for updated representations of users. The user models are influenced by the dynamic changes in users' interests and learning progress or interactions with the system. Therefore, the models can always be updated and take the current needs and preferences of the users into account (Hothi and Hall 1998; Johnson and Taatgen 2005).

User models are similar to Social Adaptation in their ability to adapt to the needs and preferences of different types of users in order to satisfy their requirements. However, Social

Adaptation can also adapt to the users' changing environment and contexts and not only to their preferences and needs. It also enriches user models by an additional driver of runtime which is the user's feedback. In this way, users' feedback can empower user models by providing additional information about the users and the changing context of use. This will improve users' satisfaction and the software system efficiency in meeting their requirements.

2.1.2 USER-CENTRED DESIGN (UCD) & PARTICIPATORY DESIGN (PD)

User-centred design (UCD) is a design process that pays a very high attention to the needs, wants, and limitations of end users of a product and tests this product on actual users to validate the design (Abras et al. 2004). It tends to avoid forcing the users to change their behaviour to use the product; it rather ensures that the final product is what users should be comfortable with. UCD design process can help software designers to meet the goal of a software product designed for their users. Users' requirements are considered from the beginning and included in the product cycle.

Participatory Design (PD) is also a design process that builds upon the same principle of UCD but with more emphases on users' involvement in the design process. In UCD, the roles of the researcher/developer are distinct and the user is not really a part of the team, but is spoken for and represented by the designer/developer. However, in PD, the roles of the researcher/developer blur and the user becomes a vital component of the design process (Sanders 2002).

UCD, PD and Social Adaptation are related in the way they involve users in deciding how software should behave. In UCD and PD, user's behaviour and needs and their perception of the usability of software guide the design process. This means the user is only involved at the design phase and the process is fully controlled by the analysts. However, in Social Adaptation users are involved at run-time by giving them the ability to provide feedback which contains information about their judgment of the quality of software behaviours. This process of collecting feedback is controlled by the software itself and the feedback collected is also used by the software, with minimum intervention of designers, to reconfigure itself in a way that meets users quality expectations.

2.1.3 END-USER COMPUTING

End User- Computing refers to systems that provide users with the ability to modify the software artefacts or even making them able to program and produce working applications (Nardi 1993). An example could be an Intelligent Room System that allows users to modify it to meet their own needs, assume that the system can turn on the lights when the user gets into the room, the user can modify (or add) this function and make the system opens the blinds instead.

End-User Computing relates to Social Adaptation in sharing the same concept in relation to uncertainty about the environment where the system works. End-User Computing and Social

Adaptation are similar in the motivation to provide the end-user with the power to change the system in a way that fulfils their specific needs in changing contexts. However, Social Adaptation, in contrast to End-User Computing, relies on users' feedback as an explicit way to trigger changes and moreover it aggregates feedback and infer the collective judgement of users instead of relying on one user feedback so that adaptation is accelerated and empowered to cope with large scale systems

2.1.4 RECOMMENDATION SYSTEMS & COLLABORATIVE FILTERING

Recommender systems are software systems that use specific tools and techniques to provide suggestions for items that are of use to a user (Ricci et al. 2011). The goal of recommender systems is to present to different types of users the items/products that are more likely to fall into their scope of interest. Recommender systems are widely used and applied in a variety of applications. Movies, music, news, books and research articles can be popular examples of recommender systems applications nowadays.

Collaborative Filtering is one of the techniques used by recommender systems. The main idea of Collaborative Filtering is to provide recommendations or predictions to the user depending on the opinions of other like-minded users (Terveen and Hill 2001). Collaborative Filtering techniques create a model from a users' past behaviour (e.g. items purchased and/or numerical ratings given to those items) and similar decisions made by other users; then use the developed model to predict items/products that are more likely to fall into their scope of interest (Melville and Sindhwani 2010). The motivation of collaborative filtering is the idea that people usually get the best recommendations from other people with similar tastes to themselves.

Collaborative Filtering can potentially benefit and help in enabling socially-adaptive systems. Since socially-adaptive systems have the ability to reconfigure themselves in response to users' feedback, Collaborative Filtering approaches can empower the decision making process of socially-adaptive systems in predicting users' preferences and priorities (based on the preferences and priorities of other like-minded users) thus achieving more satisfying requirements. Collaborative Filtering can also benefit socially-adaptive systems in economising the amount of collected feedback from users. For example, when a user A provides his feedback about a certain aspect of a software feature then there could be no need to capture user B's feedback if they are like-minded users thus, reducing the amount of collected feedback.

2.1.5 SOFTWARE PRODUCT LINE (SPL) AND DYNAMIC SOFTWARE PRODUCT LINE (DSPL)

The Software Product Lines (SPL) paradigm refers to the methods, tools and techniques used for developing a collection of similar software systems, differing in certain aspects but sharing commonality in others, from a shared set of software entities (Clements and Northrop 2002), E.g. the components of Boeing 757 and 767 are 60% in common and represent a clear example of how different products could share a certain set of assets. To express the variability in SPL one can use Variability Modelling which is a common way to describe and manage the variants of a system to enable technology to deliver similar software systems in a fast, consistent and comprehensive way with a high quality. Variability can be expressed using different approaches and one of them is the Feature Model which is a popular technique for describing commonality and variability in SPL. A Feature Model is a tree-structural notation to model feature level commonality and variability in a graphical way to make them more understandable (Lee et al. 2002). The Feature Model was first introduced in the Feature-Oriented Domain Analysis (FODA) method by (Kang et al. 1990). Since that time it has become widely adopted by the software product line community.

Runtime adaptation is becoming a key feature in many modern software applications and can be noticed in large scale systems or even small mobile applications. Dynamic Software Product-Lines (DSPLs) refers to the ability to manage the variability at run-time in order to adapt their configuration to a changing context and environment (Hallsteinsen et al. 2008). DSPL and Social Adaptation are both able to adapt to their changing environment and context in order to achieve the user's goal. However, Social Adaptation enriches DSPL by an additional driver of runtime reconfiguration which is the user feedback. In this way, feedback can maximize the efficiency of the adaptation and configuration decisions since users are given a voice in the runtime configuration of products in DSPL. This includes users' feedback on the behaviour of the feedback acquisition itself. This will ultimately maximize their satisfaction about the derived product. To enable the role of users' feedback, one can adopt Feature Modelling to elicit and model the relevant features for such role.

2.1.6 AWARENESS REQUIREMENTS FOR ADAPTIVE SYSTEMS

Requirements engineering approaches are being challenged by the development of self-adaptive systems that can operate in dynamic and volatile contexts. However, the high level of uncertainty in dynamic environment and contexts makes it hard to formulate, validate and manage requirements for these systems. Different contexts might require different requirements and unpredicted contexts might even lead to entirely new requirements (Sawyer et al. 2010). To help empower adaptivity and encounter uncertainty in self-adaptive systems, runtime reasoning about requirements should be enabled using the concepts of meta requirements called "Awareness Requirements" which aims at providing a runtime monitoring capability to the system.

An Awareness Requirement is a requirement about the system's requirements. More generally, Awareness Requirements indicate and talk about the success and failure of the system's requirements. When an actor or user starts to pursue a requirement, eventually the requirement will succeed or fail (Silva Souza et al. 2011). The benefit of Awareness Requirements is to make sure the system is meeting its desired goals more efficiently. When

Awareness Requirements indicate that a system's requirement has not been achieved, the system must take an alternative to achieve its goal.

Awareness Requirements and Social Adaptation are motivated by the same aim which is the desire to empower adaptive systems in meeting their desired goals more efficiently. However, Social Adaptation claims that software requirement cannot be always fully monitored especially in extremely volatile contexts by Awareness Requirements and users should be involved at certain point to talk about the success and failure of a requirement. Social Adaptation advocates the role of users' involvement as monitors via their feedback to assess the quality and validity of software adaptations in meeting the desired goals (Ali et al. 2012).

The feedback will empower adaptation with user's perception of actions and decisions taken. On the other side, the type of adaptation needed decides which feedback to collect and from whom. For example, if the software adaptation is critical and credible feedback is needed then it might be asked from certain type of users only (e.g. users who provide trustworthy feedback). This motivated this thesis to study users' different behaviours and motivations to feedback acquisition as discussed in Chapter 4, Sections 4.4, 4.6 and Chapter 5, Section 5.4. In the next section, this thesis gives a review of topics related to the role of users' involvement in software development through their feedback which is the main element discussed throughout this thesis.

2.2 USERS' INVOLVEMENT AND FEEDBACK IN SOCIALLY ADAPTIVE SOFTWARE

A Self-Adaptive system has the ability to adjust its behaviour in order to respond to internal and external change. A key element of a Self-adaptive system is monitoring its internal state and environment so that appropriate adaptations can take place when needed. In highly complex systems, monitoring the environment is either very difficult or very expensive. Therefore, some adaptations are not always possible since there is no monitor in place to trigger them (Whittle et al. 2010).

This highlights the need for involving users in the adaptation loop (Paramythis 2004) and the role of their input or feedback as a way to monitor the environment and provide environmental information to the system (Whittle et al. 2010). The main idea is that, the limitations of fully monitoring the environment suggests that a human commentary can potentially be used to develop and deliver a clearer view of the operating context of a Self-Adaptive System. Although there is not much literature available on the role of users' involvement and feedback in socially adaptive software, this section presents the recent available related literature.

2.2.1 THE ROLE OF USER'S INVOLVEMENT IN THE SOFTWARE LIFECYCLE

User involvement is an established research field in software engineering (Kujala 2003) and is studied in connection with human-computer-interaction (Bekker and Long 2000). Studies show that user involvement is a key factor in developing useful and usable systems even though its value may have been overlooked in software engineering practice (Kujala et al. 2005). Researchers highlight the fact that user involvement plays a significant role in improving the quality of requirements (Kujala et al. 2005) and software systems. Recently, user involvement has been changing significantly. For instance, users changed from trained programmers or technical staff to normal users (e.g. typical users' who don't have a strong technical or programming experience) (Grudin 1991a) which resulted in a significant change in developers' perceptions about users (Kanstrup and Christiansen 2006). In addition, software users and the context of use are unknown before the delivery of the software due to the different application platforms and the mobile devices. This plays a key factor in increasing the distance between developers and users (Grudin 1991b).

Current software engineering approaches lack theories for the involvement of users and their communities in the software lifecycle (Maalej and Pagano 2011; Pagano and Bruegge 2013). In the age of simple knowledge access and social media, the lack of users' involvement can harm the software's success. Restricting users' involvement and not giving them a channel to deliver their voices can result in having frustrated users who can even meet in social forums to argue against the software and harm its reputation. This lack of user involvement in the software lifecycle has several disadvantages (Maalej and Pagano 2011):

- There are little indicators, apart from download and sales statistical information, about the real usage of the software and the users' changing requirements in a dynamic environment. This can negatively impact the software success and users' experience since users' voices about their experiences with the software are not heard.
- A huge loss of valuable knowledge that can greatly improve the software when knowledgeable users are not able to share their experiences that are driven by their needs to empower the software.
- Possible destruction of the software can happen by frustrated users who feel that their voices are not heard. These users might stop using the software especially if an easy web search provides another open source alternative. They can also harm the reputation of the software by participating in campaigns against it (e.g. on Facebook) instead of improving it.

End-user involvement is particularly relevant in software engineering activities such as requirements engineering (RE). The collected end-user inputs define what to develop and therefore have a significant influence on the success of software projects (Kujala 2003). The vision of Social Adaptation's (Ali et al. 2012) is indeed to empower users' involvement, on a group basis, in the software evolution process by enabling users to steer the adaptation process via their given feedback. In Social Adaptation users are always treated as a first-class entity and given a channel to deliver their voices and experiences about the software. This can result in a more satisfied users and more successful software. This vision of Social Adaptation represents a good example of how users' involvement in the software lifecycle can be enabled and how new software engineering approaches should take place to accommodate such users' involvement.

2.2.2 INCENTIVES AND REWARDS FOR EMPOWERING USER INVOLVEMENT

User involvement in software system development is considered to be an important factor influencing software success or failure (Wong and Tate 1994). There has been an increase in research investigating the link between user involvement and successful software development (Mumford and Weir 1979; Doll and Torkzadeh 1988; Tait and Vessey 1988). There has also been an improvement and increase in the development of methods and techniques to help improving user' involvement in the software lifecycle (Mumford and Weir 1979; Doll and Torkzadeh 1988; Sharma et al. 1991).

In the so-called user-generated content, which refers to media content that are created by users that are made publicly available on the internet, the use of social incentives is a key technique to empower users' willingness to contribute to user-generated content. Social incentives are not based on anything tangible rather than only allowing the user to feel good as an active member of the community. These incentives can be in the form of receiving recognition for their work, connecting with others, and self-expression (Michahelles 2009; Smadja 2009). Examples of user-generated content can include posting reviews and feedback on a product or a software service, users uploading their own videos on YouTube, contributing to Wikipedia, etc.

Another approach used to empower users' involvement is the adoption of Incentive-centred design (ICD). ICD is defined as the process of designing a system that respects motivated behaviours/individuals by providing appropriate incentives to induce desirable behaviours from users (i.e. greater and better participation) which will result in a more successful system and satisfied users (Wash and MacKie-Mason 2006). In ICD, system developers observe tendencies in users' behaviour to provide the appropriate incentives in return (Jian et al. 2012). An example of an application of ICD would be the achievement system on some of the sport kits such as Nike. In such a system when users achieve personal records, there will be pre-recorded audio feedback from some of the famous sports athletes acknowledging the achievement and also praising the user.

In addition, gamification techniques are also highly valuable in empowering user's involvement. It refers to the use of game mechanics and techniques in non-game contexts to further engage users in solving problems (Deterding et al. 2011; Zichermann and Cunningham 2011; Huotari and Hamari 2012). Gamification has been studied and applied in several domains for the purpose of improving user engagement (Hamari 2013). In Software Engineering, gamification techniques have been timidly adopted to engage and motivate all the players involved in the development of software systems (Dubois and Tamburrelli 2013). A key gamification technique is the use of rewards for players who complete certain tasks. The rewards can be in the form of points, achievement levels (Hamari and Eranti 2011), the filling of a progress bar (O'Brien 2010), and/or giving the user virtual currency (Hamari and Eranti 2011). Social competition is another factor that can be used in gamification to empower users' engagement by for example making the rewards for accomplishing tasks visible to other users to encourage them to compete and more engage with the system (Reeves and Read 2013).

Although all of the previous techniques and methods can be highly useful to empower user's involvement with software systems, in the context of feedback acquisition for socially-adaptive systems it is still ambiguous how these techniques could be exploited and used in order to further improve users' willingness to give feedback. In addition, since cultural differences among users play a noticeable role in software success (Reinecke and Bernstein 2007), these method and techniques need to be culturally aware to allow for more efficiency in appropriately selecting the right incentives for users. A reward technique might be successful for users with Middle Eastern backgrounds but invalid for western users. This highlights the need for more research and investigations on how to exploit available incentive and reward methods in the context of feedback acquisition taking into account the effect of users' behaviour and cultural differences.

2.2.3 SOFTWARE-OBTAINED END-USER FEEDBACK

In software systems users can be involved in two ways, either by being actively working on a specific software engineering activity (e.g. suggest modifications and enhancements, perform tests), or by influencing an engineering decision about the software (e.g. users might provide feedback, rate specific decision, or influence the opinion of others) (Maalej and Pagano 2011). In Requirement Engineering, users' involvement via their given feedback is an important factor for software developers (Pagano and Bruegge 2013) since software requirements are identified only after the software is being deployed, once users have a chance to use the software and provide feedback representing their experience with the software (Ko et al. 2011).

Users' feedback has shown to be a highly useful source of information for software developers. It contains valuable information that can help to improve software quality, identify missing features and advertise and market the software (e.g. having high positive ratings can increase the sales numbers) which can improve users' experience and satisfaction in return (Pagano and Bruegge 2013). The main motivation to appreciate user feedback seems to be its origin; the user. Software developers/companies are interested in users' feedback for two reasons. The first reason is for marketing purposes. Users buy the software therefore developers/companies are always willing to satisfy users' needs. Users can sometimes harm the reputation of the software and the company if they become frustrated with the software.

Consequently, software developers/companies keep seeking users' feedback to assess the acceptance of their software. The second reason is that developers need a real-time feedback from users' about the environment, features being used, errors that occur and in which context. This feedback is helpful and useful to improve software quality and users' satisfaction (Pagano and Bruegge 2013).

However, users' feedback design methods (if they exist at all) in software systems are not systematic and rather ad hoc (Maalej et al. 2009). Little is known about how the way developers collect and work with users' feedback and how users perceive and behave in regard to feedback acquisition in software applications. Apart from error reports, there is no commonly agreed practice on how to provide or gather user feedback during software evolution (Pagano and Bruegge 2013). There is a high need for tools and approaches to support the process of collecting, structuring, analysing, and tracking user feedback. This is indeed the aim of this thesis which goal is to systemise the process of end-user feedback acquisition in software applications. This can improve the quality of collected feedback and economize the amount of requested feedback.

2.2.4 FEEDBACK GATHERING TOOLS & CLASSIFICATIONS

End-user feedback plays an important role in improving software engineering processes, success and users' satisfaction. Current software engineering practices for gathering users' feedback are not systematic and rather ad hoc. A previously published research proposed a social software engineering process called "Snail" that thoroughly and continuously involves users by establishing interaction channels and integrating user communities (Maalej and Pagano 2011). It aims at making users as integral part of software applications and software engineering processes via their continuous given feedback.

In Snail the community activities and the users' interactions with the software application and environment are continuously observed (snail tentacles). Users can give feedback about the software application during its usage, and can be proactively asked to do so when a problem occurs. Individual user feedback and community activities are continuously analysed to filter, aggregate, and prioritize the information, and identify inconsistency or conflicts. The software engineers work on the feedback and employ community channels to connect to users for discussions and clarifications. Engineering decisions about the software can then be made considering the opinion of the community, and then communicated back to the community. Finally, updates and changes are published, restarting the cycle again (Maalej and Pagano 2011).

Although Snail is a powerful approach in improving the socialness of software, it can be further complemented with a systematic practice to gather users feedback (e.g. when to proactively ask users' for feedback?). This will maximize the quality of obtaining users' feedback and positively contribute to the goal of the Snail approach. This is indeed the main aim of this thesis, providing software engineers with a systematic and adaptive way to gather users' feedback.

2.2.4.1 FEEDBACK GATHERING TOOLS

With regard to feedback acquisition, there are several available tools for gathering users' feedback in software systems in different forms (e.g. text, images and videos) and many of them also include context capturing functionality. Common examples are the feedback tools in applications such as Eclipse, Adobe Acrobat or Microsoft products. They allow users to send their feedback as well as contextual information captured by the tool to software engineers in a structured form (Maalej et al. 2009). Karahasanoviæ et al. (2005) presented a tool to gather continuous users' feedback for research purposes. In this tool, users are regularly asked for explicit feedback to be sent to the researchers. EUREKA (Maalej and Happel 2009), is a plug-in for the Eclipse environment that provide users with input fields which can be triggered from time to time (see Figure 5). The input data with automatically generated contextual information is then sent to a server back-end. These two approaches are intended as a replacement to traditional diary or observational studies but can also fit the idea of a "continuous feedback acquisition and requirement engineering.

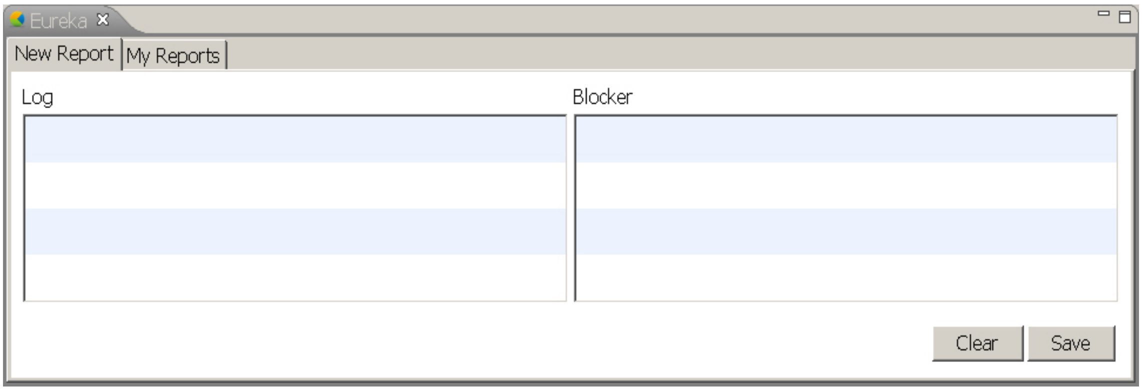


FIGURE 5 EUREKA PLUGIN FOR ECLIPSE

Other approaches are not application specific and can be integrated within an application framework. Few studies suggested to continuously and remotely collect users' feedback for the purpose of enabling software engineers to adjust their applications to changing needs and requirements. They give a major focus on the importance of observing the contextual information related to users' feedback as a major enabler for understanding their feedback (Maalej et al. 2009). Seyff et al. (2010) and Ali et al. (2012) also claim that future software applications should support users in providing feedback about their needs or the validity and quality of software behaviour at runtime.

Seyff et al. (2010) introduced a smartphone app called iRequire. iRequire allows users to provide feedback about their needs in situ, using photos, videos, sound or simply text comments. The authors reported on a first study conducted to show the feasibility of a tool-supported requirements elicitation. They envision the concept of self-adaptive systems that adapt to users' needs without the help of engineers. Similarly, VoiceYourView[2] integrates advanced speech recognition techniques, Neuro-Lingustic Programming (NLP), and data presentation techniques to allow individuals to deliver their feedback by using speech or text messaging on anything they like. The main idea in VoiceYourView is to allow users to say whatever they want about a product or a software service and then their feedback is processed automatically to identify the main themes and sentiment among all users. Finally, product stakeholders/software engineers can view a summary of the major themes and how users feel about those themes (Whittle et al. 2010)**.**

In addition, there are several social media enabled tools like UserVoice[3], Get-Satisfaction[4], and IdeaStorm[5] that gather users' feedback and allow them to collaboratively share new ideas and vote on existing services or suggestions. Bajic and Lyons (2011) found that these collaborations focus users' efforts, which leads to more homogeneous feature requests and easier decision on how the application should evolve. Going through all of the previously mentioned tools, it might be observed that there could be a lack or unavailability of a common systematic engineering practice for conducting the process of feedback acquisition in software applications. This unavailability of a common systematic practice can have negative implications on the feedback quality and the software success.

Moreover, some of these tools lack the ability to integrate directly into the software applications. They work as an additional tool for submitting users' feedback enriched with contextual information. This makes the process of giving feedback a separate task from the actual user's task (Maalej and Pagano 2011). This can highly affect the contextual information when switching from a task to another to give feedback which might lead to a misunderstanding of the given feedback. Finally, the previously mentioned tools lack an integrated back communication channel from the software engineers or the software systems to the users. The existence of such a channel can always keep users aware of the actions and modifications their given feedback triggered on the system which can highly improve their satisfaction and trust in the software. The aim of this thesis is to provide software engineers with a systematic and adaptive method to gather users' feedback (see Chapter 7). This method allows for customizing the feedback acquisition to the nature of software (e.g. health-critical software) and hence improve the feedback acquisition capability of integration with different software applications.

2.2.4.2 FEEDBACK CLASSIFICATION

In order to understand, identify and analyse user needs and preferences and to derive or adjust requirements accordingly, software engineers extensively communicate with the users and other stakeholders and seek their feedback with regard to the developed software. In practice, these communication activities are different and depend on the project development setting, the

[2] http://www.voiceyourview.com.
[3] https://www.uservoice.com
4 https://getsatisfaction.com
5 www.ideastorm.com

project phase and the motivation of users to provide input. Some authors (Maalej et al. 2009) use the communication methods used to gather users' feedback during the software lifecycle as references to classify users' feedback. They distinguish between *pull communications* where feedback is pulled from the user and *push communications* where feedback is pushed by the user (Maalej et al. 2009). Other authors (Maalej and Pagano 2011) classified users' feedback in terms of its type. They distinguish between *qualitative feedback* or rich data (e.g. words, pictures, and objects) and *quantitative feedback* that lead to numbers and statistics such as user polls and opinion surveys.

However, Maalej et al. (2009) and Maalej and Pagano (2011) also classify feedback in terms of *explicit* and *implicit* feedback. Users' feedback can be explicit when the user has the will to provide the input or implicit when the user unintentionally provides input information (e.g. when they are observed in their environment to elicit useful information). Pagano and Bruegge (2013) also classified users' feedback in terms of its purpose for example, *error reports, feature requests,* and *feedback on existing features* which includes *improvement and enhancement requests*. (Maalej et al. 2009) has provided a more concrete classification of users' feedback (see Figure 6). They provide four main classes of feedback:

- ***Class1: Pull Communication & Explicit Feedback***

In this class software engineers typically gather (in an off-line mode) explicit user input in a pull communication process. In such a process, software engineers may conduct face-to-face interviews or workshops with users and stakeholders. They may also conduct surveys or write clarification requests. In all these activities users are explicitly willing to give feedback on the validity, quality, preference or needs in regard to a software solution.

The nature of this communication process can sometimes affect the quality of collected feedback and lead to a misleading interpretation of user input. First, users sometimes do not know exactly what they want and cannot always deliver their preferences and needs in a clear and accurate way (Ulrich and Eppinger 2007). It is also difficult for them to remember accurately their experience with the software and the issues that they encountered. Users might also forget to mention contextual information that might be of a particular importance for understanding their needs and engineering product requirements. For this reason, requirements engineers cannot always understand the problems in an efficient and effective way when following a pull communication process for gathering users' input.

- ***Class2: Push Communication & Explicit Feedback***

In many cases, a traditional face-to-face communication between users and engineering teams is not possible especially for a continuous feedback due to physical distribution, cost or policy constraints. In class 2, a huge number of new requirements, problems/bugs and needs are delivered explicitly and in a push way from users to software engineers. Users are actively communicating (online while using the software or offline after trying the software) enhancements request, reporting issues, needs and requirements in emails, forums, pop ups, etc.

For example, in socially adaptive system online applications with a large user base such as Twitter and YouTube, users request new features, suggest ideas and enhancements, or vote for and rank a new deployed requirement in a "democratic" process.

This class can certainly provide a more holistic view of users' experience with the software if major contextual information such as the users' interaction sequence and feedback attitude is collected. This can highly improve the understandably of users' feedback and guarantee a better response by the software engineers according to the given feedback. (Bettenburg et al. 2008).

In addition, there is a lack of communication between users and software engineers. Software engineers typically seek clarification from users to better understand the exact situation and why the feedback (e.g. enhancement request) was given. Asking for users' clarification is a time-consuming process and users might lose their interest to respond. Moreover, usually several iterations are needed to clarify the actual issue. A better communication channel or system should be in place in order to improve the communication between users and the software engineers. Moreover, explicit feedback in a push communication needs considerable effort and motivation from users. In order to be useful, a user must be motivated enough to add a useful input and meaningful comments. Users' main concern is to accomplish their tasks and not to give feedback. They might either provide a low quality input or just avoid giving feedback. This highlights the need for having an incentive scheme to encourage users to submit their feedback and engage with a system.

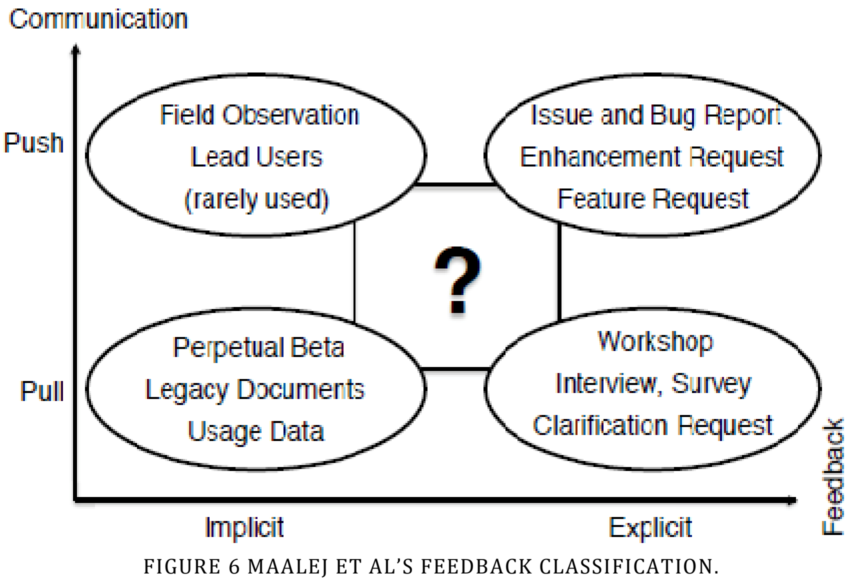


FIGURE 6 MAALEJ ET AL'S FEEDBACK CLASSIFICATION.

- ***Class3 Pull Communication & Implicit Feedback***

In this class, software engineers can get information about users and their needs by analysing gathered users' feedback in terms of system usage statistical and behavioural data. Some software systems such as the widely used web browser Mozilla Firefox come with a modular plug-in architecture, which triggers automatic updates several times a week. On the other hand, these applications collect users' usage data implicitly and infer recommendations and modifications on their updates according to the collected data.

Although this implicitly collected data can be very useful, it is rarely sufficient without further explicit input and clarification from the users. Also implicit feedback lacks sufficient amount of contextual data since the tools used to collect this data are very specific in terms of collected context information due to some privacy reasons. In addition, implicit feedback is machine-oriented and may be difficult to be reviewed, commented or understood by users themselves. It also does not even provide any back channels for clarifying ambiguities in data. Therefore, submitted data can be useful but it can also include a lot of "noise" and irrelevant input.

- ***Class4: Push Communication & Implicit Feedback***

In the final class, users implicitly deliver feedback to software engineers. This typically happens when users are being observed during their usage of the system where they might implicitly reveal useful input about the software product to the engineering team. For example, the so-called lead users who often push useful implicit feedback to the software engineers. The lead users (Stefan and Ashok 1998) are specific type of users who have a high creativity potential and technical competence. The lead users typically adjust the product by themselves to meet their needs more efficiently. Therefore, they implicitly push feedback to software engineers on how to improve the software product. Implicitly pushed feedback is a very useful source of information and can lead to a better understanding of users' needs, preferences and identification of new requirements.

- Maalej et al. (2009) provided a relatively comprehensive classification of user input/feedback in terms of the communication method used (push/pull) and the kind of feedback collected (implicit/explicit). However, this thesis argues that the communication method in class2 can also be a *pull* and not only *push.* In some situation, the software system might proactively aske users for their explicit feedback (online while using the software or offline after using the software). For example, when a user is struggling to use one of the software features then the system might ask them for their feedback by showing an online pop up. In addition, the authors included *issue and bug report, enhancement request* and *feature request* as the main feedback types in the second class (see Figure 6). However, there are still some missing types of feedback that are useful and important to software engineers and users and can be included in this class. An example of a missing feedback type can be *praise feedback* which can serve as indicator for software engineers to provide them with a measure of users' satisfaction about a particular aspect of the software. This thesis extends and modify (Maalej et al. 2009)'s classification as shown in Figure 7. The rest of the literature review give a review of topics that constructs potential solutions to the problem domain of this thesis. The next section gives a review of the concept of Personas and how it could potentially impact the solution space of this thesis.

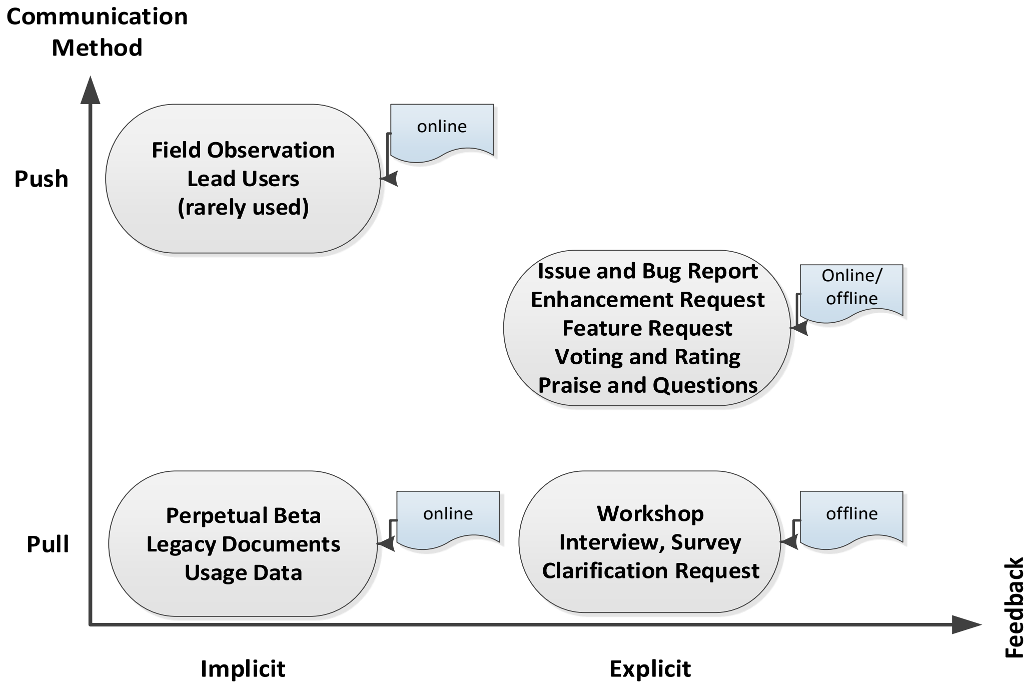


FIGURE 7 FEEDBACK CLASSIFICATION (ADAPTED FROM (MAALEJ ET AL. 2009))

2.3PERSONAS AND FEEDBACK ACQUISITION

Personas are defined as fictional characters that are developed depending on actual users' data to represent different types of users in the design process (Cooper 1999b). These fictional characters are given names and photos to bring life to them. In qualitative researches about users, one of the important questions is how the gathered data is going to be used and represented in a meaningful way. Persona is a powerful tool to represent qualitative users' data. It is used to encapsulate the aspects of discovered user categories and to draw discussions about these categories which can help in the design process of software systems in general (Weber 2010b).

In this thesis, personas are adopted (see Chapter 6, Section 6.7) as they allow software engineers to better understand the diversity of users' behaviours and their needs with respect to feedback acquisition in socially adaptive software. This understanding will positively impact the design of the adaptive acquisition of users' feedback. In addition, Personas are good starting point to initiate detailed discussion about the different types of users (personas) which could highlight new design opportunities for the feedback acquisition activity. Generally, Personas do not eliminate other approaches (i.e. scenarios) but complement them or are used when another approach is impractical (Grudin and Pruitt 2002; Pruitt and Grudin 2003). The rest of section 2.3 will review approaches related to personas to further emphasise the power of personas.

2.3.1 SCENARIOS AND PERSONAS

Personas are the foundation on which to build scenarios which are considered as a natural element of Persona-based design (Grudin and Pruitt 2002; Pruitt and Grudin 2003). (Carroll 2000), an interaction design theorist, defines a scenario as a story that has a setting, agents or/and actors who have goals and objectives, and a sequence of actions and events. Given that scenario-based design has actors or agents, these actors or /and agents are typically not defined in a way that promote generative and interactive engagement among designers. Consider the following Carroll's example:

"An accountant wishes to open a folder on a system desktop in order to access a memo on budgets. However, the folder is covered up by a budget spreadsheet that the accountant wishes to refer to while reading the memo. The spreadsheet is so large that it nearly fills the display. The accountant pauses for several seconds, resizes the spreadsheet, moves it partially out of the display, opens the folder, opens the memo, resizes and repositions the memo and continues working"(Carroll 2000).

Looking at the above example it can be seen that the character involved in the scenario (the accountant) does not enjoy a high level of likeness. This lack of lifelessness of characters in such scenarios has been criticised by scenario-based design researchers who suggested using caricatures, perhaps some shocking or extreme ones to bring life to these characters (Nielsen 2002). In relation to this, Bødker (2000) noted: *"it gives a better effect to create scenarios that are caricatures.. it is much easier . . . to relate to. . . . Not that they "believe" in the caricatures, indeed they do not, but it is much easier to use one's common sense judgment when confronted with a number of extremes than when judging based on some kind of "middle ground."".*

On the other hand and as previously stated, Personas are fictional people who have life, names, likenesses, occupations, age, gender, ethnicity, status, etc. Indeed this is what makes Personas different from scenarios which can be constructed around personas, but the personas come first (Grudin and Pruitt 2002).

2.3.2 CONTEXTUAL DESIGN, ETHNOGRAPHY AND PERSONAS

Contextual Design is a structured and well-defined user-centred design process that was developed by Beyer and Holtzblatt (1997). Contextual Design core philosophy is to use ethnography methods to collect data about users/customers of a product/software and their interactions with it in their normal work environment/the field, interpret and consolidate the collected data in a meaningful way, and then apply these findings into a final product/software (Beyer and Holtzblatt 1997).

Contextual Design is a useful and powerful approach for collecting and analysing behavioural data. The ethnographic data collected, when available in a reasonable depth, provides the most help in developing realistic Personas. This makes Contextual Design a strong candidate for informing Personas which represent people's behaviours based on actual data (Beyer and Holtzblatt 1997). In addition, contextual design stresses the power of communication among team members to share knowledge obtained in the field. Personas are indeed a tool to achieve this and thus a natural partner to contextual design (Grudin and Pruitt 2002; Pruitt and Grudin 2003)

2.3.3 PARTICIPATORY DESIGN AND PERSONAS

Participatory Design is a design process that focuses on actively involving the eventual users of a system or application in the design process to help ensure the result meets their needs and is usable. In participatory design the roles of the designer blur and the user becomes a vital component of the design process (Sanders 2002). Participatory Design shares the same goal as personas in that they both stress the power of engaging developers with people's behaviours and also improve developers' ability to anticipate behaviours of familiar people. When designing for a relatively small, accessible group of people, Participatory Design makes the most sense. But in relatively large groups of people/users, the product/software development process becomes more challenging for participatory design (Pruitt and Grudin 2003).

Traditional participatory design gives a great deal of strong focus on socio-political and quality of life issues. However, these issues are becoming more significant and challenging nowadays as the reach and sophistication of computing extends. Despite the fact that the industry and many companies nowadays are trying hard to engage these issues at a high level of abstraction, most usability and interaction design approaches avoid addressing these issues due to their challenging nature.

On the other hand, Personas take socio-political and quality of life issues to the surface. Each Persona has a life, gender, age, race, socio-economic background, work, etc. This provides an effective tool for recognizing, understanding and perhaps changing assumptions about users. If one populated a Persona set of adolescent users with middle-aged white males, it would be clear that this is a mistake (Pruitt and Grudin 2003).

2.3.4 EFFICIENCY OF FICTION IN PERSONAS

People normally easily engage with fictional characters in novels, movies, and television shows. They shout and argue/discuss passionately about what these fictional characters have done on the screen or after the novel ends. In fact, fictional characters in on-going television dramas sometimes come to resemble normal people to some extent (Pruitt and Grudin 2003).This highlights the power of fiction in Personas and their ability to engage the design team more in fruitful discussions. However, too much fiction could be harmful and the developers of Personas should bear in mind that they are design tools in the first place (Goodwin 2001).

In movies and television shows, many actors are prepared by observing and talking to people who will resemble the fictional character the actors will play. It is the same case with Personas where the fictional characters are based on real data collected by empirically investigating users/people. A fiction based on research can be powerful and useful to communicate (Pruitt and Grudin 2003). For example, watching an actor resembling a character that surrender slowly to a dementia disease on a television show, one can recognize, understand the disease and perhaps design a new technology to support dementia patients, if the character played is based on actual observation and data.

2.3.5 THEORY OF MIND AND PERSONAS

For many years, psychology scientists have been exploring the so-called theory of mind, which relates to people's ability to predict other peoples' behaviours by understanding their mental states. People in every day of their lives use partial knowledge to make inferences, predictions and expectations about people around them. Whenever they say or do certain things, they predict the reactions of others and they always keep learning from these experiences (Premack and Woodruff 1978).

Personas are indeed a way to invoke this powerful human capability and bring it to the design phase of a software/product. Personas, as previously stated, are generative and engaging; once the design team are fully engaged with personas, the can effortlessly project them into new behavioural situations (Pruitt and Grudin 2003). For example, if team members were told that "John has purchased a new phone", they can easily extrapolate and predict how this could impact his behaviour. This can then lead to the team creating different scenarios based on their predictions. Generally, people do this kind of extrapolation all the time and are very skilled at it. The next section of this literature review looks at the problem of this thesis and contributes to the solution space from a physiological point of view.

2.4 PERSONALITY TYPES AND SOCIALLY-AWARE FEEDBACK ACQUISITION

People's personalities are different and play a vital role in affecting their relationships, decisions, and life in general. (Jung 2014), the founder of analytical psychology, believed that people are either energized by the outside world (Extraversion) or their own internal world (Introversion). This theory led Myers, the developer of the Myers-Briggs Type Indicator (MBTI) (Briggs 1976), to investigate the rationale for the personality differences among people to make Jung's theory understandable and useful in impacting different aspects of people's lives.

(Myers et al. 1985) believed that many problems that involve human interactions and personal choices could be tackled more efficiently with Jung's theory of psychological types in mind. This eventually led to the development of Myers-Briggs Type Indicator (initially as a questionnaire), which provide structure for understanding similarities, differences and psychological preferences among human beings (Briggs 1976). In the context of this thesis, the acquisition of users' feedback can be conducted in a social settings thus identifying user's personality types could play a vital role in enabling a socially-aware feedback acquisition. This can be utilised by having a feedback acquisition process that can employ personality questions to identify users' different behaviour to feedback and adapt accordingly as discussed in Chapter 7, Section 7.4.3.

2.4.1 EXTRAVERSION AND INTROVERSION

The first and widely popular pair of psychological types is Extraversion and Introversion. Extraversion refers to the state of being energized by the world outside the self. Extravert people enjoy being socially active and tend to be more enthusiastic, talkative, and animated. They enjoy spending time with other people and find it less rewarding to spend time alone. On the other hand, introversion refers to the state of being mostly concerned with one's inner world. Introvert people prefer self-reflection to social interactions.

They also prefer to observe before taking part in an activity. Introverts tend to be more quiet, 'peaceful', and reserved (Myers et al. 1985). The concept of "Extraversion and Introversion" is included in established models of personality in various forms. Examples can include; Big Five model (Barrick and Mount 1991), Hans Eysenck's three-factor model (Eysenck 1950), Raymond Cattell's 16 personality factors (Cattell 1966) and the Minnesota Multiphasic Personality Inventory (Schiele et al. 1943).

- The following statements generally apply to Extraversion:
  - People describe me as "outgoing" or as a "people person."
  - I feel comfortable in groups and prefer working in them.
  - I have a wide range of friends and I know lots of people.
- The following statements generally apply to Introversion:
  - People describe me as "reflective" or "reserved."
  - I feel comfortable being alone and prefer to do things on my own.
  - I prefer to know just few people well.

2.4.2 SENSING/INTUITION

The second pair of psychological types is Sensing and Intuition which reflects what people focus their attentions on. Sensing type people pay attention to physical reality, what they see, hear, touch, taste, and smell. They are concerned with reality and what is occurring in the present. They like to see the practical use of things and learn better when they can observe how to use what they are learning. Experience for sensing people speaks louder than words (Myers et al. 1980; Myers et al. 1985).

On the other hand, intuitive types prefer a learning atmosphere in which an emphasis is placed on meaning and associations. They prefer learning by thinking a problem through than by practical experience. They also value working with abstract theories even if they do not know how they will use them in practice. They describe or remember events more as an impression of what they were like than as facts or detailed information of what happened (Myers et al. 1980; Myers et al. 1985).

- The following statements generally apply Sensing types:
  - They remember events as facts and details of what actually happened.
  - They address problems by going through actual facts until they comprehend the problem.
  - They believe in experience more than words.
- The following statements generally apply to Intuition people:
  - They remember events by what they read "between the lines" about their meaning.
  - They address problems by trying different ideas and possibilities.
  - They believe in impressions and metaphors more than actual experiences.

2.4.3 THINKING/FEELING

The third pair of psychological types is Thinking and Feeling which reflects people's decision preferences. Thinking type people like to find objective truth and logical principles to be applied when making a decision regardless of the specific situation involved. They like to analyse the whole situation and its pros and cons and then be consistent and logical in making a decision (Myers et al. 1980; Myers et al. 1985).

On the other side, feeling type people tend to put an emphasis on issues that can be personalized while they consider other people's motivations in a specific situation. They usually pay attention to the values and what is best for the people involved. They tend to do whatever will cause or maintain harmony (Myers et al. 1980; Myers et al. 1985).

- The following statements generally apply to Thinking people:
  - They search for logical explanations or solutions to mostly everything.
  - They make decisions with their heads and want to be always fair.
  - They appreciate telling the truth (even if it is harmful to others) more than being tactful.
- The following statements generally apply to Feeling people:
  - They appreciate harmony and get nervous when it is missing.
  - They usually search for what is important to others and show concern for others.
  - They tend to make decisions with their hearts and want to be sympathetic.
  -

2.4.4 JUDGING/PERCEIVING

The fourth pair of psychological types is Judging and Perceiving which reflects how people regard complexity. Judging types enjoy organized and structured information. They prefer a planned/organized way of life and feel more comfortable when decisions are made. They also like to have as much control over their lives as possible. On the other hand, Perceiving people tend to enjoy a more flexible and spontaneous way of life and usually adapt to the world rather than trying to organize it (Myers et al. 1980; Myers et al. 1985).

- The following statements generally apply to Judging people:
    - They usually prefer to have things decided.
    - They prefer to make a list of things to do.
    - They always plan work in advance to avoid rushing when a deadline is approaching.
- The following statements generally apply to Perceiving people:
    - They usually prefer to stay open to respond to whatever happens.
    - They tend to be loose and casual. They prefer to keep plans to a minimum.
    - They are usually stimulated by an approaching deadline.

As a consequence to Jung's theory and Myers-Briggs Type Indicator, personality questions or tests emerged and have been widely used and available on the internet. These tests are intended to help people inferring their personality type (e.g. Extraversion or Introversion) after answering some personality-related questions.

This sensibly opens the gate for a highly potential candidate to do the adaptation of feedback requests by employing the idea of personality questions or tests (Myers et al. 1985). Users' answers to these questions can map them to the persona or personas elements they belong to (see Chapter 6, Section 6.7 for the developed personas of users' behaviour to feedback acquisition). For example, users who turn to be extraversion based on their answers to some personality questions can be mapped to personas that represents socially active users. Another example can be that users who turn to be a thinking type based on their answers can provide a trustworthy feedback and mapped to personas that impose a good feedback quality (see Chapter 7, Section 7.4.3 for more details on how to use such a concept). The next section gives a review of the potential Software Engineering approaches that can impact the outcome of this thesis.

2.5 REQUIREMENT ENGINEERING TECHNIQUES AND FEEDBACK ACQUISITION

Requirement Engineering employs a variety of techniques to understand, elicit and model users' requirements. Since feedback acquisition is by itself a software and users have their requirements and preferences on how it should be developed. This section sheds the light on two of the widely popular techniques that could potentially contribute to the requirement engineering phase of feedback acquisition in software applications.

2.5.1 GOAL MODELLING AND FEEDBACK ACQUISITION

Goal modelling is a widely used technique in the early phases of software requirement engineering. Goal modelling has been introduced into RE for a variety of reasons, within different RE activities and to achieve different objectives (Mylopoulos et al. 1999). It is concerned with how the intended system meets its goals, why the system is needed and how the stakeholders' interests may be addressed (Yu 1997). In general, it improves the efficiency of the requirement engineering process and offers modelling concepts to represent the rationale of social and technical actors in a socio-technical systems (e.g. users' rational in a socially-adaptive system). This is done through notions like goals, softgoals, decomposition, actors and their interaction (Yu and Mylopoulos 1998). Such features are essential for enabling the design of systems that are adaptive to changes (Fickas and Feather 1995). In particular, goal modelling benefits RE in the following (Yu and Mylopoulos 1998):

- Investigation and expression of the relationships between a system and its environment (i.e. what the system should do in specific context and why). This could identify the reasons for the needs of the system which gives software engineers a better understanding of the system.
- Better clarification of requirements because the process of specifying goals normally leads to asking "why", "how" and "how else". This provides less risk of either missing requirements, or of over-specifying (asking for things that are not needed).
- Better way of dealing with conflicts in which it can identify and help to make decisions about cost, performance, flexibility, security and other goals. It can also reveal conflicted interests between stakeholders. The nature of goal modelling allow it to identify conflicts because meeting one goal can interfere with meeting other goals.
- Better measurement of requirement completeness in which requirements can be considered complete if they fulfil all the goals in the goal model.

In the context of feedback acquisition, goal modelling can help in providing software engineers with a better understanding of users' goals with regard to feedback acquisition and the different alternatives to reach those goals. It also helps to identify conflicts between users' goals and the business/system goals which ultimately help in making decisions that prioritize the goals to achieve. For example, for some users, the goal for giving feedback is to collect a tangible incentive. However, this goal could conflict with the business goals in that does not provide enough budget to satisfy this users' goal. This maximizes software engineers' ability to better design an adaptive feedback acquisition that can cater for conflicts among goals and take the right alternative to satisfy stakeholders' goals (see Chapter 7, Section 7.2 for more elaboration on the use of Goal Modelling to impact the outcome solution of this thesis).

2.5.2 USE CASES AND FEEDBACK ACQUISITION

Scenario-based techniques have been used by the software engineering community to understand, model and validate users' requirements (Jacobson 1992; do Prado Leite et al. 1997; Ralyté et al. 1999). Among these techniques, Use Cases have received a special attention in RE in particular. Use Cases in UML (Fowler 2004) are used to describe the interactions between the system and its actors (e.g. users and feedback acquisition). An actor is an external element that can interact with the system (users who give feedback). A Use Case is a description of a sequence of actions that a system performs to result value for an actor. A single use case can generate several scenarios which explain the possible alternatives an actor could take when interacting with system (e.g. the identification of the different scenarios/alternatives of actions for a certain user when giving feedback in different contexts).

In the context of feedback acquisition, use cases could highly impact the design of an adaptive feedback acquisition. They offer software engineers the ability to capture the interactions of users with the feedback acquisition and the possible alternatives of their interactions. For example, for some users, the use case of Supplying Feedback would have a flow in which all the control is given to the user as a primary actor (e.g. deciding when to be asked for feedback). However, for some other users, the flow could be different in which some of the control is given to the feedback acquisition (e.g. collecting implicit feedback about their usage of the software). Capturing users' possible interactions using use cases gives software engineers a better elicitation of user' requirements with regard to feedback acquisition and impact how the feedback acquisition should cater for such interactions (see Chapter 7, Section 7.3 for more elaboration on the use of use cases to impact the outcome solution of this thesis).

2.6 SUMMARY

This chapter presented a review of the state of the art in relation to Social Adaptation, the acquisition of users' feedback in software applications and potential approaches to impact the solution space of this thesis. The next chapter tackles the first objective of this thesis and reports on a two-phase Expert Survey to identify core benefits, domain areas and challenges for Social Adaptation and users' feedback acquisition.

3. CHAPTER3: SOCIAL ADAPTATION AND FEEDBACK ACQUISITION FOUNDATIONS &CHALLENGES

To fulfil the first objective of this thesis, this chapter reports on a two-phase Expert Survey to identify core benefits, domain areas and challenges for Social Adaptation and users' feedback acquisition. It presents the research method used, the study design, the results and the conclusion. The findings provide practitioners and researchers in adaptive systems engineering with insights into this emerging role of users, or the crowd to stimulate future research to solve the open problems in this area.

3.1INTRODUCTION

In the self-adaptive software community there has been a great deal of emphasis on architectures to support design and development of adaptation, models for anticipating and reacting to changes in the managed system and methods for verifying properties of these systems (Oreizy et al. 1999; Cheng et al. 2009). Ultimately, self adaptivity is a meta-computing capability which enables a system to reason about itself and its dynamic environment so that it can formulate the right decisions to reach stakeholders' requirements (Fickas and Feather 1995). While success on these foundational fronts has contributed significantly to the field, the role of users in the adaptation process has only recently become a main focus. This can be partly attributed to lessons learnt from successfully deployed self-adaptive systems such as Rainbow (Garlan et al. 2004), where it was found that the adaptation process was not transparent to users. An example of such transparency limitations can be illustrated by the insufficient explanation offered by self-adaptive system about why a course of actions was chosen instead of alternative actions to meet the users' requirements.

Early research in self-adaptive systems limited users' ability to steer adaptation with the good intention of maximizing system autonomy and minimizing human efforts. However, this would lead to adaptation decisions that were valid but only temporary since users were not given a voice in the iterative validation of these decisions after software was deployed (Ali et al. 2012). Consequently, one of the identified research challenges in the engineering of self-adaptive software systems road map is:

[To devise a way of] "analysing feedback types from human-computer interaction and devising novel mechanisms for exposing the control loops to the users, keeping the users of self-adapting systems in the loop to ensure their trust" (Cheng et al. 2009).

Although the role of users in the adaptation process has recently been recognized (Esfahani and Malek 2010; Ali et al. 2011a; Ali et al. 2012; Pagano and Bruegge 2013), there is still a lack of consensus and holistic approaches on how to engage the users/ crowd in that process. This study addresses this problem and details a two-phase Expert Survey to gather and analyse the knowledge of experts in adaptive systems research. In this thesis the acquisition of users' feedback is given a special focus due to its vital role in enabling this kind of adaptation and the lack of engineering approaches for it. The survey provides practitioners and researchers in self-adaptive systems with insights and challenges to consider when involving users, individually or as a crowd, in the adaptation process.

The objectives of conducting an Expert Survey were to poll the opinion of experts on (i) the principles and primitives for enabling Social Adaptation (ii) the role of users' feedback in steering software adaptation, and (iii) the engineering of software-based feedback acquisition. The findings provide practitioners and researchers in adaptive systems engineering with insights on this emerging role of users, or the crowd, and stimulate future research to solve the open problems in this area.

3.2RESEARCH METHOD

Expert Opinion/Elicitation is a technique used to serve different types of purposes, and can be used as an assistance to identify problems, clarifying issues related to a specific topic or field and for evaluation purposes with any state of design by consulting experts. In Expert Opinion there are no specific requirements for the participation of experts, apart from having experience and knowledge of the domain under the discussion. However it is important to make sure that these experts who are consulted are objective regarding the design of the evaluated topic otherwise it will be hard to make sure the obtained data is impartial (Knol et al. 2010).

3.2.1 EXPERT SELECTION

Expert selection can have a high effect on the survey outcomes and the acceptability of the result in the wider community (Gordon 1994). Since this thesis is tackling a multidisciplinary research area and in order to have a diversity of viewpoints, it targeted experts from Requirement Engineering and Adaptive Systems research community with additional focus on at least one different related domain: HCI, Human Factors in Computing, Psychology, Privacy and Security Engineering, Socio-Technical Systems Engineering and Social Computing. The inclusion criteria allowed for experienced participants who are knowledgeable in their respective fields, evidenced by proven publication track record. Although the majority of the experts work in academia, they either worked in industry previously or were engaged in collaborative projects involving industrial partners.

To make sure these participants had sufficient experience and knowledge about the discussed issue, some assessment questions regarding their knowledge and experience were asked at the beginning of the questionnaire. According to expert elicitation practitioners, the number of experts to be included should be at least six, otherwise the quality of the conclusions and their generalizability can be doubtful (Cooke and Probst 2006). In the first phase of the survey 35 experts were invited; 29 forms were returned. Considering the average actual time taken to complete the survey (35 minutes), the size of the form and the amount of effort required to complete it, we consider this as a good response rate. In the second phase the 29 experts who participated in the first phase were invited and only 21 forms were returned. Then additional 5 experts were invited so that a total of 26 forms were completed.

3.2.2 DESIGN, TEST AND DISTRIBUTION OF THE SURVEY

Online questionnaires were used as a data collection method for the study because the experts were widely distributed geographically (five countries). The questionnaire contained both types of questions: open-ended questions and close-ended questions. The open-ended questions were used to minimize the risk of missing significant information and to give participants a space to include information they felt was relevant. Closed questions were employed to get a better response rate and to put less effort on participants when answering the questionnaire (Leung 2001). Appendix 1 shows the Expert Survey submitted to the participants.

Survey questions were deduced and extracted from two talks, given by two of the author of this thesis, followed by a brainstorming session on Requirements-driven Social Adaptation. The sessions took place on March 2013 as a part of a project meeting, which included academics in the computing departments of three universities. The participants set included 12 researchers who have a variety of relevant expertise including Requirements Engineering, Self-adaptive Systems, Dynamic Software Product Lines, Cloud Computing, Machine Learning and Human Factors in Computing.

The questions focused on the value and benefits of Social Adaptation for both developers and clients, its application areas, whether it has to be autonomous or semiautonomous and its technical development challenges. A good part of the discussion focused on the acquisition of users' feedback, how to engineer it, and whether it should be adaptive as well. The survey script contained 25 questions discussing and investigating these points. Questionnaires need to be tested on typical respondents before the actual data collection stage begins to ensure their readiness and clarity (Franklin et al. 2003).

The questionnaire was tested first on three respondents who met the inclusion criteria. After the test and revision, experts were sent an email containing a brief description of the purpose of the survey and asking them to participate in the Expert Survey. A period of two weeks was given for them to come back with their input. Surprisingly, the response rate was high (29 out of 35) which is an indicator that the field is relevant to Requirements Engineering which is a primary research area of the surveyed experts. The quality of the selected experts, the use of the second phase as a confirmatory phase, the carful design and test of the questions can all indicate a good quality of the data collected in this study.

3.3FIRST PHASE RESULTS

The returned survey forms were analysed and responses were cleaned up and irrelevant/ inconsistent answers were excluded. A descriptive analysis on the quantitative part of the survey was conducted to describe the data and to get the feel of it. A qualitative analysis was applied to the open-ended questions of the survey which included coding the response and creating categories to identify patterns and trends in the responses[6].

3.3.1 SOCIAL ADAPTATION BENEFITS AND VALUE

Social Adaptation claims to offer valuable benefits for both developers and users. This claim raises important questions that need to be addressed by experts. The following 4 questions attempt to dig a little deeper, that is to understand, not only to what extent Social Adaptation is beneficial but also to understand better the nature and context of those benefits among different groups. The questions also vary in their focus. In brief, Q1 to Q3 consider benefits for developers and clients or users with Q4 attempting to consider areas that are either particularly fruitful or, in contrast do not offer particular benefits. This is then followed by a sub-section that deduce (based on the findings of Q1 to Q4) potential challenges to supporting these benefits to developers and clients (see Section 3.3.1.1).

Q1: How would you rate the benefits of Social Adaptation: (a) For software developers? (b) For software clients.

Beneficiary / Rating	High	Medium	Low
Developer	13	14	2
Client	20	8	1

A rating of *Low* implies Social Adaptation is not beneficial; *Medium* implies that there are benefits but not necessarily significant; significant benefits are rated as *High*. There is a consensus among experts that Social Adaptation, if realised, is a useful concept to developers (93% chose medium/high) and clients (96% chose medium/high). The higher perceived benefit to clients is perhaps not surprising, as users will have more active role in steering the adaptation process.

Q2: What are the benefits of Social Adaptation for software developers?

Social Adaptation, as indicated by experts' responses, offers valuable benefits for both developers and users of adaptive software. **[Finding 2.1]** Acquired knowledge through users' feedback can be used to build and refine models used by the system or to improve the accuracy of reactive or predictive adaptive algorithms for various aspects of the self-adaptive system (as indicated by 50% of experts' responses). New knowledge may also reveal latent requirements

[6] The nature of open-ended questions in this survey allowed the experts to freely comments on their answers to some of the questions. This resulted in a richer data introduced by the experts which allowed the author to qualitatively extract more findings.

that were not known before. Developers of self-adaptive systems can therefore use Social Adaptation to: (1) improve problem resolution tactics by identifying bugs and scenarios that cause software crashes and poor performance, (2) better prioritise requirements and maximise the productivity of limited development resources, (3) identify the distribution of software use across age groups, geo-location, time of day etc., and (4) build knowledge-bases of contextual profiles, which are hard to elicit at design time where the users have not used the system in real settings yet.

In contrast to Q1, where the numbers are revealing; this open question gave a great deal of insightful comments from the expert survey. In terms of benefits, respondents noted that Social Adaptation: *"Provides insights from the user perspective to software developers on aspects of the system that need to change."* (EX24). *"Learning about and adapting to new (or un-elicited) requirements and making software more aware of new contexts seamlessly."* (EX25). *"Up-to-date knowledge - accessible unobservable knowledge - able to react to new events (in a faster way) - more knowledge shared knowledge"* (EX1). *"Considering adaptation early avoids making hard and expensive changes afterwards when the system is running."* (EX12).

[Finding 2.2] Future socially-driven adaptive systems may disrupt the current development paradigm of self-adaptive systems, in terms of time to market or deploy, by reducing the upfront effort in design phase to the barest minimum (as indicated by 30% of experts' responses). By taking the socialized view of adaptation, the system will only provide a platform for users to express their preferences, whilst design decisions are collectively made by users at run-time. This indeed makes Social Adaptation different from other approaches (e.g. agile software development) in which the variability points of the software and execution environment will be learnt at run-time based on feedback provided by users which makes. This is a realistic assessment since users of today's software system vary widely in their preferences (perhaps, influenced by culture, norms, age group, location etc). Designers of adaptive systems will focus on engineering open, configurable, and extensible platforms, instead of debating functionality choices.

Q3: What are the benefits of Social Adaptation for users?

The ultimate goal of Social Adaptation is to satisfy users' requirements efficiently by enabling users to steer and tailor the adaptation process. The experts agreed that Social Adaptation was of most benefit to users. This correlates with responses to the first question. **[Finding 3.1]** The benefits cited included improved trust (users feel their voice is considered) (as indicated by 60% of experts' responses), **[Finding 3.2]** user satisfaction (software behaves according to users' judgement) (as indicated by 80% of experts' responses), **[Finding 3.3]** transparency (adaptation decisions is visible to users) (as indicated by 30% of experts' responses), and **[Finding 3.4]** confidence in self-adaptive system (as indicated by 20% of experts' responses) *"1- Acknowledgement of clients' opinion 2 Visible involvement of clients in* *the adaptation"* (EX22), *" Having a software with an adaptive behaviour based on similar users / past experience and participation / involvement in a community is very gratifying to many users"* (EX5).

The data also revealed that the involvement of users and their ability to collectively configure the software on the fly at run-time could result in users perceiving the software as a partner rather than a tool *"Clients will have more user-friendly software as a result of the analysis performed on information received from them. Their confidence and trust in the adaptive systems may grow. They will be able to provide focused and real-time feedback to the developers which can in return empower them."* (EX3). Indeed this is consistent with the overarching objective of self-adaptation, where the software is expected to adapt to users' needs and not the other way around. Since Social Adaptation is a group-based adaptation, the software will be able to quickly adapt to group norms and beliefs, without the conventional maintenance-evolution phases.

Experts also indicated that users' involvement should be accommodated under some constraints and users should not be always involved in shaping and validating software adaptation. Restricting users' involvement is due to a negative effect on users' experience (e.g. annoyance), which might arise when involving them too much *"if every system would ask often to confirm something, the end-users would be overwhelmed and they would not react at all"* (EX1).

Aside from the apparent benefits of Social Adaptation to users (as listed above), one unusual finding is the perceived impact on the software life-cycle of self-adaptive software systems. The idea of fully automated user-driven evolution or reduced human involvement in the evolution process already raises many challenges. An example of these challenges could be the way users' feedback is being collected and analysed by the system. The findings and challenges in this area will be discussed in question 5.

Q4: Can you nominate certain areas and application domains where Social Adaptation: (a) has distinguished benefits (b) should not be used?

A common theme raised by the respondents was the need for a user-centred or human-oriented software and user bases in which preferences of the entire user population collectively steer the adaptation. This thesis classifies the identified application areas as follows:

- **[Finding 4.1] Mobility intensive systems** (as indicated by 35% of experts' responses) where software is used in different contexts, e.g., driving navigation systems *"Telecom industry will be highly interested in such applications, Content intensive applications"* (EX10).
- **[Finding 4.2] Large-scale systems** such as SaaS clouds (as indicated by 22% of experts' responses), where the software is in global demand and developers are unable to elicit preferences of groups of users distributed around the world *"Any software system that has a very large community of users, e.g., smartphone applications."* (EX7).

- **[Finding 4.3] Real time management systems** where crowd-sourcing will empower the system monitor and enhance the decision making (as indicated by 10% of experts' responses). For example, evacuation scenarios or congestion management at train stations or airports *"- unobservable areas - areas with lots of traffic and different end-users and needs (airport, central train stations, shopping malls) - mobile devices (apps with unknown end users)"* (EX1).
- **[Finding 4.4] Highly interactive systems:** These are software that is frequently used for a variety of purposes where it is hard to know a priori how users will judge quality in the diverse contexts of use and human-computer interaction. Mobile, pervasive, and social networking applications fall into this category of highly interactive systems (as indicated by 5% of experts' responses). *"Systems with repetitive tasks. The driving navigation system is a good example. Perhaps an operating system, adapts to common usage. Also embedded electronics like refrigerators, heating, etc. could adapt to behaviour without criticality."* (EX14).
- **[Finding 4.5] Prototyping tools** in moderately dynamic systems (as indicated by 30% of experts' responses), where feedback from Social Adaptation can be used to infer user needs before a final implementation is carried out. Here, there is an assumption that the rate of adaptation is relatively controllable by human development effort after the final system is deployed *"in prototyping for requirements engineering activities possibly, to find out which model to focus on in the actual implementation"* (EX16)

The experts considered Social Adaptation inapplicable in the following domains:

- **[Finding 4.6] Critical systems** where wrong Social Adaptation could result in disaster or huge financial loses (as indicated by 90% of experts' responses).
- **[Finding 4.7] Security sensitive applications** (as indicated by 50% of experts' responses).
- **[Finding 4.8] Non- or less-interactive systems** under the control of centralised authorities such as payroll system or an embedded system (as indicated by 40% of experts' responses). A comment from an expert with regard to areas outside the scope of Social Adaptation says *"Safety Critical Systems where real-time data input may result in disasters e.g. a nuclear power plant. High Secure Systems - that is sealed/closed systems e.g. military missile systems."* (EX3).

It is interesting to uncover the subtle difference between "personalised" adaptation and "social" adaptation as evident in candidate application areas listed above. The former refers to a type of user driven adaptation where the objective is to meet the requirement of user(s) with mutual non-conflicting preferences. Crucially, some users may choose not to conform to popular opinion, therefore, they should be given the freedom to deviate from the choice deemed best by the group (e.g. for some privacy reasons). On the other hand, Social Adaptation is a different concept as the preferences of the entire user base (including conflicting ones) is collectively used by the system to adapt in a way deemed best for the group. Figure 8 illustrates the difference between these concepts. Applying Social Adaptation in the previous domains is a promising opportunity to empower adaptation quality. The reason is that the potential to get a wide range and large volume of users' feedback is high and that the users' feedback is meaningful as the interaction between users and the software is intensive.

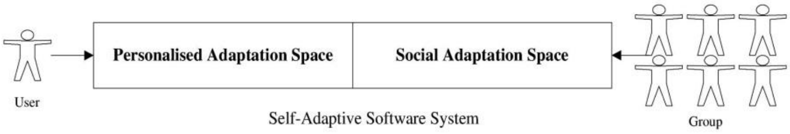


FIGURE 8 PERSONALIZED VS. SOCIAL ADAPTATION IN SELF-ADAPTIVE SOFTWARE.

3.3.1.1 CHALLENGES TO SUPPORTING SOCIAL ADAPTATION BENEFITS TO DEVELOPERS AND USERS

Utilising on-line Learning: The role of on-line learning is a key in realising open platforms for socially-driven adaptation since there is a question of: how do we build a system when little is known about the users of the system. Existing work in the self-adaptation literature mainly uses learning-based approaches to model decision-making about computational configurations at run-time (e.g. (Tesauro et al. 2006; Elkhodary et al. 2010)). Unlike these efforts, the role of learning here is that of learning user trends, behaviours, and adaptively resolving conflicts among user preferences at run-time. Since user behaviour and preferences are not static, i.e. users themselves evolve, a fully open social-adaptation platform should empower users to decide the protocols and resolution tactics for their collaboration. Consequently, **[Challenge 1]** there is the problem of identifying what learning models to use for enabling user interaction and conflict resolution at run-time and **[Challenge 2]** how the process of mining users' feedbacks should be conducted to inform recommendations which are consistent with the system's requirements.

Gauging User Involvement: The challenges include **[Challenge 3]** how to measure users' involvement with the system as a main descriptor of their feedback, **[Challenge 4]** identifying the degree to which users are allowed to configure the software on the fly at run-time and **[Challenge 5]** and specifying restrictions for their involvement (e.g. users can provide feedback after a certain period of use).

Monitoring Adaptation Spaces: From Figure 8, it can be observed that while a user is able to independently tune parameters in their personalised adaptation space, the system's Social Adaptation space, which also affects the user, is a product of a collective configuration. **[Challenge 6]** The research challenge here is to develop models and languages to allow users to specify their preference on the way their requirements are reached (e.g. when to rely on the crowd and when to take their personal choices). Another challenge **[Challenge 7]** is to develop mechanisms to allow software to deduce such users' preference without getting them explicitly involved.

3.3.2 IMPLEMENTATION CHOICES (AUTONOMY AND FEEDBACK ACQUISITION)

The following questions dig much deeper and attempt to look at issues to consider in implementing Social Adaptation, with Q5 examining autonomy and Qs 6-8 focussing on aspects and developments challenges of users' feedback acquisition process since it plays a significant role in enabling Social Adaptation. This is then followed by a sub-section that deduce (based on the findings of Q5 to Q8) potential challenges to supporting these benefits to developers and clients (see Section 3.3.1.1).

Q5: Knowing that relying on Social Adaptation is a user's choice, are there cases where software should still ask users to confirm its adaptation decision?

The degree of autonomy in socially-adaptive systems has been always debated. Should the system take an autonomous full control on the adaptation process? Or should the user interfere sometimes? This thesis extracts, analyses and discusses experts' opinion in regard to autonomy in Social Adaptation. 96% of the respondents agree that user confirmation is essential before an adaptation action autonomously suggested by software is allowed to impact the system. The remaining 4% claim that adaptation actions should not require user confirmation since this is why self-adaptive systems are autonomous *"User should be not too much bothered. Moreover I think that user identification should be automatic in the mechanism adopted to get user feedback."* (EX20).

Even though the consensus tilts towards user involvement in confirming adaptation actions, many experts believe that the answer to this question is not a binary (yes or no). In many cases, the choice of whether user should be involved depends on the type of services provided by the adaptive system (e.g. its criticality), the implication of the action on user's privacy, security, and financial spending. Also, "nice to have" autonomous Social Adaptation actions should not outweigh core functionalities of the system, hence users need to choose what is important to them in each context.

Q6: Can the quality of collected feedback be affected by the way it is collected?

All respondents agreed that the way feedback is collected has an effect on the quality of the feedback. The exact implication of the collection mechanism can be manifested in terms of:

- **[Finding 6.1]** Time (as indicated by 60% of experts' responses): Asking users for feedback when they are busy may result to poor responses or it may be discarded. Finding the right time to ask for feedback is important *"Asking in a busy moment of the end-user will result in only yes/no answers. Asking if the end-user is bored will result maybe in creative feedback, but not necessarily high quality."* (EX1).
- **[Finding 6.2]** User interface (as indicated by 70% of experts' responses): Short, concise, clear questions should be preferred to long, complicated ones. The user interface should

allow users to express their biases using wider band of options, such as Likert scale (Likert 1932), rather than conventional yes or no options.

- **[Finding 6.3]** Language (as indicated by 75% of experts' responses): The phrasing of the question, for example based on the language proficiency of the user, will determine how users interpret and respond to questions. Even for experienced users, misinterpretation is sometimes inevitable due to the ambiguity inherent in natural languages *"simple interfaces are essential, avoid long texts or complicated questions. Things such as "like" and "dislike" may be especially effective"* (EX5).
- **[Finding 6.4]** Quality of users (as indicated by 60% of experts' responses): This involves asking the right user population for feedback and ensuring the size of users is representative of the group characteristics *"Is it the right user group? How representative is the feedback? User profiles? Can you capture user's perception or viewpoint of the question asked? How is the feedback question phrased? Any domain specific language?"* (EX3).
- **[Finding 6.5]** User's mood (as indicated by 30% of experts' responses): Some interesting responses suggest strongly that the mood of users should be factored into the feedback acquisition process. While the mood of users may impact the quality of feedback, it is still hard to monitor the emotional state of users (e.g. happy, bored, excited, angry etc.) during feedback acquisition. Perhaps advances in recognizing emotions through facial expression (Adolphs 2002) could be helpful in this area.

Q7: Is the development of feedback acquisition mechanisms technically challenging?

Experts stress that developing software-based feedback acquisition poses a variety of technical challenges due to changing context of use and users' evolvement. Almost two-thirds (65%) of the responses indicated that the users selection and interaction style is a key challenge which includes incentivising users to give feedback, when to ask for feedback, who to ask for feedback, how to interact with users without annoying them and the usability degree when giving *feedback "Uncertainty, building a user-friendly interface, convincing the user of the importance of it."*(EX7). *"The design should incentivize impatient and ignorant users to give feedback. Implementation would not be the problem, designing is the main challenge."(*EX18).

Responses also indicated that, engineering of users' feedback acquisition is a multidisciplinary process and it has a potential usefulness and strong relationships with various domains such as, Requirement Engineering, Ubiquitous systems, HCI, Context aware systems Social Science, Psychology, Recommendation Systems and Machine learning.

From the various experts' answers to the engineering challenges of a software-based feedback acquisition, it can be deduced that the engineering of an adaptive software-based feedback acquisition stands out as a technically challenging process. The first step of gathering feedback already raises many questions: What type of feedback should be asked of users? How should feedback be cleansed, represented, processed, filtered, and selectively adopted for use by the systems? If users are relied upon to steer the adaptation process then the system must be equipped with capabilities to cope with the ambiguity flaws of natural languages.

Additionally and in relation to the findings of the previous question (Q6), the extracted engineering challenges of the feedback acquisition can also be considered as factors that can affect negatively/positively the quality of collected feedback. Addressing these challenges can possibly improve the quality of collected feedback. For example, developing an incentive scheme for users might improve the quality of their given feedback.

Q8: If feedback acquisition is adaptive, what could be the adaptation drivers?

Context is important for the choice of feedback acquisition methods, and experts agreed that an adaptive feedback acquisition mechanism is a necessary enabler to decide ways of acquiring feedback. Some possible drivers for such adaptive mechanism suggested were:

- **[Finding 8.1] User experience** (as indicated by 85% of experts' responses): E.g. usage frequency. This could inform how often users should be asked for feedback. For example, a less frequent user may find providing feedback meaningless, since they hardly use the software in the first place *"Usage information of user's laptop (or other smart devices). E.g. if a user is browsing web sites or watching a video, probably he/she is free."* (EX17).
- **[Finding 8.2] Application constraints** (as indicated by 20% of experts' responses): Such as the application model, domain model, and level of interactivity of the software are likely to influence ways of acquiring feedback *"this should include several components, including a user model, application model, domain model, and a general feedback or adaptation model."* (EX8).
- **[Finding 8.3] Direct enquiry** (as indicated by 20% of experts' responses): Involves asking the users if they wish to provide feedback, if yes, how often they wish to do so and what methods they would like to use for providing such feedbacks *"Ask the user what they prefer. When is the best time to give feedback, what form would they like?"* (EX14).

The identified trend here is that drivers of adaptive feedback acquisition should not be studied in isolation. Such drivers may trade-off against each other. A user that provides feedback frequently, for example, will only find answering the direct enquiry questions useful, as a way of improving his/her feedback provision.

3.3.2.1 RESEARCH CHALLENGES TO IMPLEMENTING SOCIAL ADAPTATION

Degree of Autonomy: It is interesting to observe that although experts advocate that Social Adaptation is useful for meeting users' requirements, autonomously, based on the crowd feedback, they still believe individual users should be in the loop during the decision-making process of their software. **[Challenge 8]** It raises the question of how much control users are willing to surrender to software systems. For example, in modern autopilot assistant systems, pilots take a supervisory role while software controls the flight of airplanes. The challenge here seems to be psychological in nature, since users are happy to trust the system when they are involved in the decision-making. Does this mean users trust their own socially-generated decisions less than expert knowledge encoded in systems such as Auto-pilot? Suppose, users were able to collaboratively fly an aircraft, would it land safely? Perhaps, this trustworthiness issue is why experts believe that Social Adaptation should not be used in critical systems but in less critical systems (See Q4).

Impact of Collection Approach and Importance of Mood: Investigation into the impact of user mood on the quality of feedback in specific application domains may require evidences from psychology *"This is mainly a psychological issue, finding the right time and modality and give incentives to the user for providing a good feedback"* (EX5). Advances in neuroinformatics could be helpful in this area. Some experts suggest that feedback should only be requested for features that are frequently used by the user. **[Challenge 9]** This will require mechanisms for monitoring user's feature usage statistics/trends and using these results to inform which feedbacks are requested from the user.

[Challenge 10] Some domain-specific feedback acquisition languages and mechanisms might be needed. Some feedback mechanisms may work better in some application areas than others. This challenge is akin to problem in requirement elicitation based on application areas and user experience. Perhaps some lessons can be learnt from the requirements community to address this challenge.

[Challenge 11] Additionally, software developer could turn to mature fields like HCI to learn how interfaces are built to gather feedback from users in a variety of contexts or even to use innovative features such as voice-based feedback acquisition rather than purely text which might make the process easier and more enjoyable for users *"[users] will provide more feedbacks to a system that can support voice recognition than others without this feature."* (EX17).

Impact of User Selection and Interaction Styles: In a software-based feedback acquisition, a further important challenge is catering for the users' selection and interaction style. More specifically, there are challenges in the following aspects **[Challenges 12-18]**: 1) modelling users styles (including incentives, 2) deciding when to ask for feedback, 3) deciding what type of feedback to ask for, 4) deciding with whom to interact, 5) deciding how to interact and avoid annoying or confusing users, 6) deciding how to design for maximized usability in feedback acquisition and 7) deciding how to ensure trust and reliability of acquired feedback.

Feedback Acquisition Drivers: [Challenge 19] the challenge here is indeed the need to identify the relevant drivers of the adaptive acquisition of users' feedback and **[Challenge 20]** engineer these drivers in a way that is non-intrusive to users. In addition, Social Adaptation is applicable and useful in various domains and the availability of a systematic approach for engineering an adaptive users' feedback acquisition is highly valuable. It could bring promising benefits for users and developers in the different domain where adaptation is recommended and, perhaps, different disciplines like marketing and e-commerce**. [Challenge 21]** Therefore, the development of an application-independent framework for an adaptive users' feedback acquisition is also a key challenge of users' feedback acquisition.

3.4 SECOND PHASE RESULT

From the Expert Survey responses, a set of core findings and challenges were deduced and extracted in the area of Social Adaptation and engineering of users' feed-back acquisition. In order to confirm the set of extracted findings and the degree of relevance and difficulty of the extracted challenges to the Requirements Engineering research community, the thesis conducted a second phase survey. 29 experts who responded in the first phase were invited and 21 forms were returned. Then 5 new experts were invited who attended at least one of the seminars given by the author of this thesis on Social Adaptation. They all responded which made a total of 26 completed form in this phase.

The survey was designed and delivered following the approach used in designing the previous Expert Survey (see Section 3.2). Before experts answer the survey, they were given a brief reminder about the purpose of the first-phase Expert Survey and then a brief description about the second-phase survey and the purpose of it. In addition, a brief description before each set of challenges was given to highlight why it was extracted/identified as a challenge to give a clearer vision to experts before answering the questions. The questions were developed to discuss and gather experts' opinion in regard to the following three points:

- Confirming the findings of the first phase. The study focused on the debatable findings which did not receive a high percentage of consensuses in the first phase. The findings were marked using the tag **[Finding x.y]** in Section 3.3. Three options for each finding were given: *Agree, Partially Agree, and Disagree.*
- Measuring the degree of challenge in each of the extracted challenges (the challenges were marked using the tag **[Challenge x.y]** in Section 3.3). The following three options were given to the participants: *[Ch: A]*: It is challenging and it requires significantly new approaches, *[Ch: B]*: It is challenging but it can still be solved by extending and customizing existing approaches. [*Ch: C]*: It is not really challenging and solutions already exist in the literature.
- Measuring the relevance degree of each challenge to the area of Requirement Engineering (RE). Three options were given to the experts: *[RE: A]*: It is very relevant to RE research. [*RE: B]*: It is not strictly relevant to RE research, but having a solution for it is still beneficial to RE. *[RE: C]*: The challenge and solution are not relevant to RE research and practice.

The following tables present a summary of the second survey findings:

TABLE 1 THE CONFIRMATION OF EXPERTS ON THE FINDINGS OF THE FIRST PHASE.

Finding	Agree	Partially	Disagree		Finding	Agree	Partially	Disagree
[2.1]	50%	46%	4%		**[4.6]**	**81%**	15%	4%
[2.1]	**69%**	27%	4%		**[4.7]**	34%	54%	12%
[3.1]	65%	34%	4%		**[4.8]**	50%	38%	12%
[3.2]	**73%**	27%	0%		**[6.1]**	**92%**	8%	0%
[3.3]	38%	58%	4%		**[6.2]**	**85%**	15%	0%
[3.4]	58%	31%	11%		**[6.3]**	**88%**	12%	0%
[4.1]	**69%**	23%	8%		**[6.4]**	**77%**	19%	4%
[4.2]	50%	38%	12%		**[6.5]**	**69%**	31%	0%
[4.3]	46%	46%	8%		**[8.1]**	**81%**	19%	0%
[4.4]	**85%**	15%	0%		**[8.2]**	65%	30%	4%
[4.5]	65%	27%	8%		**[8.3]**	50%	46%	4%

In Table 2, a high degree of challenge is given to engineering challenges related to enabling users to steer the adaptation process and the degree in which they are willing to steer it (e.g. challenge 6 and 8). This high degree of challenge is perhaps due to the lack of models and languages for enabling users to express their adaptation preferences and the lack of studies on the degree of autonomy in socially-adaptive systems. Another noticeable high degree of challenge was given to challenges related to engineering feed-back acquisition for different application areas and empowering adaptivity in it (e.g. challenge 10, 19 and 20). The reason behind this high degree of challenge could be the obvious lack of systematic approaches for engineering feedback acquisition.

In addition, challenges related to users' involvement, feedback collection and interaction styles and feedback mining to inform adaptations show a high degree of relevance to RE (e.g. challenge, 2, 5, 6, 10, 13, 14 and 16). This high degree is perhaps because experts believe that users' involvement in the adaptation process, ability to provide feedback in their preferable way and the system's ability to react to their feed-back accordingly is a user' requirement that should be systematically engineered and efficiently met.

TABLE 2 THE CHALLENGE DEGREE AND THE RELEVANCE TO RE OF EACH OF THE CHALLENGES OF THE FIRST PHASE.

Challenges	[Ch:A]	[Ch:B]	[Ch:C]		[RE:A]	[RE:B]	[RE:C]
[1]	46%	46%	7%		50%	42%	8%
[2]	15%	77%	7%		**65%**	27%	8%
[3]	38%	54%	7%		42%	46%	12%
[4]	27%	57%	15%		50%	42%	8%
[5]	15%	65%	19%		**61%**	35%	4%
[6]	**61%**	34%	7%		**84%**	11%	4%
[7]	50%	50%	0%		50%	42%	8%
[8]	**58%**	34%	7%		**65%**	27%	7%
[9]	15%	77%	7%		57%	27%	15%
[10]	**61%**	35%	4%		**69%**	31%	0%
[11]	15%	54%	30%		27%	57%	15%
[12]	42%	50%	8%		46%	34%	19%
[13]	35%	50%	15%		**65%**	27%	7%
[14]	19%	58%	23%		**62%**	34%	4%
[15]	24%	50%	8%		50%	42%	8%

[16]	31%	61%	8%	**65%**	23%	12%
[17]	35%	46%	19%	54%	38%	8%
[18]	35%	57%	7%	58%	35%	7%
[19]	**58%**	42%	0%	46%	50%	4%
[20]	**54%**	46%	0%	54%	42%	4%
[21]	46%	46%	8%	50%	46%	4%

3.5 THREATS TO VALIDITY

The expert survey has three main threats to validity:

- The first threat is one of the common issues when designing a questionnaire and relates to ensure whether the questions were understood by all experts as intended. This threat is addressed as a pilot test was conducted on typical respondents then some questions were revised and modified to ensure clarity. This was done for both phases of the survey.
- The second relates to the low percentage of the experts who have industrial experience in adaptive systems. The reason is that adaptive systems are not yet widely applied in industry and much of the work is still in academia. This could mean that the results are flavoured with more judgements coming from academia than industry.
- The third relates to the fact that Social Adaptation is a forward-looking way of developing adaptive systems. This would mean that the answers of the experts are fairly speculative. However, given that most of the elements of this domain as well as the survey questions are directly related to the main areas of expertise of the experts (e.g. requirements engineering, adaptive systems, HCI, and social computing) we would consider that the answers are good enough to draw credible insights.

3.6 SUMMARY

This chapter has synthesized findings from a two-phase Expert Survey of 29 experts in the first phase and 26 experts in the second phase on the topic of Social Adaptation and the challenges posed by the mechanisms for collecting user feedback, to steer the adaptation process. The consensus among experts is that Social Adaptation is a highly beneficial concept to both developers and clients of self-adaptive systems. However, enabling Social Adaptation is a technically challenging process due to the lack of models and mechanisms for enabling such a concept. Engineering approaches are highly needed for Social Adaptation to empower users' involvement in shaping adaptation decisions and to systematically develop the feedback collection process and interaction styles as well as feedback mining. The chapter has highlighted research challenges in the areas of providing an enabling platform for Social Adaptation and the design of adaptive feedback acquisition mechanisms that fits user context. The next chapter investigates peoples' preferences and behaviour to feedback acquisition from a user point of view through a mixed method approach.

4. CHAPTER 4: USERS’ BEHAVIOURS TO FEEDBACK ACQUISITION (A MIXED METHOD STUDY)

As discussed in the previous chapters, users’ feedback is a main source of knowledge on how users perceive the role of software in meeting their requirements. Despite this role of users’ feedback, there is a lack of systematic engineering approaches on how to design its acquisition mechanisms. To fulfil the second objective of this thesis, in this chapter, an empirical study is conducted following a mixed-method sequential exploratory approach to explore the main drivers of adaptation and understand users’ behaviour when being asked to provide feedback. The findings are meant to enrich the knowledge base for developers and researchers in users-centric, or crowd-centric, adaptation. It also highlights areas of study for future research in the area. This chapter reports on the research method, study design, the result and then draws a conclusion.

4.1INTRODUCTION

Traditionally, the research in engineering adaptive systems has deliberately limited the role of users in steering the adaptation process with the good intention of maximizing the system’s autonomy and minimizing the need for human's intervention. There has been a great deal of emphasis on architectures to support design and development of adaptation, models for anticipating and reacting to changes in the managed system and methods for verifying properties of these systems (Oreizy et al. 1999; Cheng et al. 2009). Ultimately, self adaptivity is a meta-computing capability which enables a system to reason about itself and its dynamic environment so that it can formulate the right decisions to reach users' requirements (Fickas and Feather 1995). Thus, users and their requirements are main drivers for adaptation.

Overlooking the role of users in forming adaptation decisions and the reliance on software developers and design-time validation, steered by developers, would lead to adaptation decisions that eventually and sometimes very quickly, become invalid. To keep the software up-to-date with regards to users’ needs, users should be given a voice in shaping adaptation as a lifelong process (Ali et al. 2012). Giving users an active role makes adaptation more transparent and increases their confidence in the system (Garlan et al. 2004). In this direction, one of the research challenges identified in the engineering of self-adaptive software systems road map is: [To devise a way of] analysing feedback types from human-computer interaction and devising novel mechanisms for exposing the control loops to the users, keeping the users of self-adapting systems in the loop to ensure their trust" (Oreizy et al. 1999).

Thus users can collectively enrich the adaptation decision making ability. Social Adaptation has the benefit of keeping the software and developers’ knowledge about users updated. Since users are treated as first-class entities in both the engineering and also the operation of such systems, Social Adaptation is also expected to improve transparency and raise user trust in self-adaptive systems. In fact, over a long time using the software, users may be able to shape the decision-making process in a way that can only be done by today's experts.

A core element of involving users in the adaptation process is that their feedback is obtained while using the software in different contexts. The lack of engineering processes for feedback acquisition leads to poorly designed feedback collection mechanisms and this in turn harms the quality of collected feedback, users' experience and the quality of adaptation and evolution decisions (Pagano and Bruegge 2013). Despite of this role of users' feedback, there is a lack of research on how to engineer feedback acquisition in a way that guarantees quality of the obtained information and, the same time, maintains user's experience.

In this chapter, an empirical study is conducted to understand users' different perspectives and behavioural aspects to feedback acquisition for socially-adaptive software. A mixed method (sequential-exploratory approach), consisting of interviews and a questionnaire, was followed. The findings shows that the acquisition of feedback is best designed as an adaptive process itself. This section also elaborates on a set of important factors which should be catered for when deigning that adaptation. The findings contribute to the knowledge base for developers and researchers on tackling the diverse challenges of a systematic development of feedback acquisition for the user centric, or crowd-centric, adaptation.

4.2RESEARCH METHOD (MIXED METHOD PARADIGM)

In recent years, approaches of mixed method have gained more attention by the research community. "Mixed methods research is the type of research in which a researcher or a team of researchers combines elements of qualitative and quantitative approaches (e.g., use of qualitative and quantitative viewpoints, data collection, analysis, inference techniques) for the purpose of breadth and depth of understanding and corroboration" (Johnson et al. 2007). (Creswell et al. 2003) also define mixed methods study as "the collection or analysis of both quantitative and/or qualitative data in a single study in which the data are collected concurrently or sequentially, are given a priority, and involve the integration of the data at one or more stages in the process of research". The main objective of mixed methods approaches is to gain a more complete and deeper understanding of human behaviour and experience by combining two different research methods within a research study (Morse 1991).

Since this thesis is studying users' behaviour, mixed methods approaches are appropriate candidates to be adopted. A qualitative approach would allow to explore and gain insights of users' behaviour in relation to providing feedback in software applications. A quantitative approach will help to elaborate on the findings of the qualitative approach and to allow for more generalizability. Without excluding other ways of understanding users behaviours, mixed method was a natural choice especially that the author had little idea on how users perceive feedback requests. The literature also lacks systematic studies on this problem. Thus, the author cannot even come up with a concrete set of questions and be sure that they are relevant. For this reason, qualitative research was used first to figure out relevant factors from the point of view of users and then used that to design a questionnaire, i.e. quantitative approach, which helped to reach a larger set of participants.

4.2.1 MAJOR TYPES OF MIXED METHODS DESIGN

Methodologists have paid a great attention in order to define the different types of mixed methods designs. Creswell and Clark (2011) define four types of mixed methods designs; the Triangulation Design, the Embedded Design, the Sequential Explanatory Design, and the Sequential Exploratory Design. Due to the nature of this research and the available resources, a review is given of two relevant designs that can be suitable candidates to be adopted in this thesis; Sequential Explanatory Design and Sequential Exploratory Design. This work presents the major characteristics of each design as well as the strengths and weaknesses of each.

4.2.1.1 THE SEQUENTIAL EXPLANATORY DESIGN

The Sequential Explanatory Design consists of a two-phase mixed methods design. According to (Creswell et al. 2003), the first phase of the design consists of the collection and analysis of quantitative data while the second phase consists of the collection and analysis of qualitative data. The steps in this design are illustrated in Figure 9. In The Explanatory Design, the priority is normally given to the quantitative phase, and then the two phases are integrated in the interpretation phase of the result (Creswell et al. 2003).

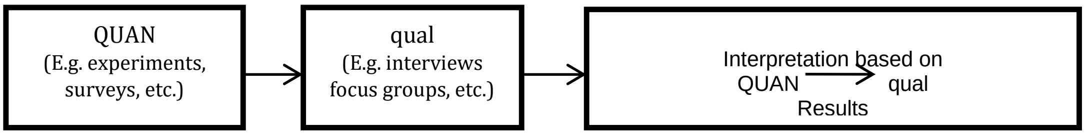

QUAN
(E.g. experiments, surveys, etc.)
qual
(E.g. interviews focus groups, etc.)
Interpretation based on
QUAN
qual
Results

FIGURE 9 EXPLANATORY DESIGN, ADOPTED FROM (CRESWELL AND CLARK 2011).

The overall objective of this design is to use the qualitative results to assist in explaining and elaborating on the initial result of the quantitative study (Creswell et al. 2003). For example, this method can be useful when unexpected or surprising results in the quantitative phase need to be more investigated and explained (Morse 1991). The qualitative data is the way to understand and examine these unexpected or surprising results. The core strength of this design is its simplicity in regard to the effort needed to conduct it (e.g. the second part might be simple when there is few surprising or unexpected result) (Creswell and Clark 2011). The core weakness of this design is the possibility of missing important information due to its quantitative nature at the first stage which might restrict and limit participants to certain answers.

4.2.1.2 THE SEQUENTIAL EXPLORATORY DESIGN

The Sequential Exploratory Design consists of a two-phase mixed methods design which has some features that are similar to the explanatory design. In this design the first phase is the collection and analysis of qualitative data then the second phase is the collection and analysis of quantitative data. In the Exploratory Design the results of the qualitative phase can be used to develop the instruments of the second quantitative phase (Creswell and Clark 2011).

(Creswell et al. 2003) indicated that "the purpose of this design is to use quantitative data and results to assist in the interpretation of qualitative findings". The exploratory design shows its usefulness when examining elements of an emergent theory and when generalizing qualitative findings to populations (Johnson and Onwuegbuzie 2004). Figure 10, shows the flow and the emphasized phase of the Sequential Exploratory Design for the purpose of instruments development.

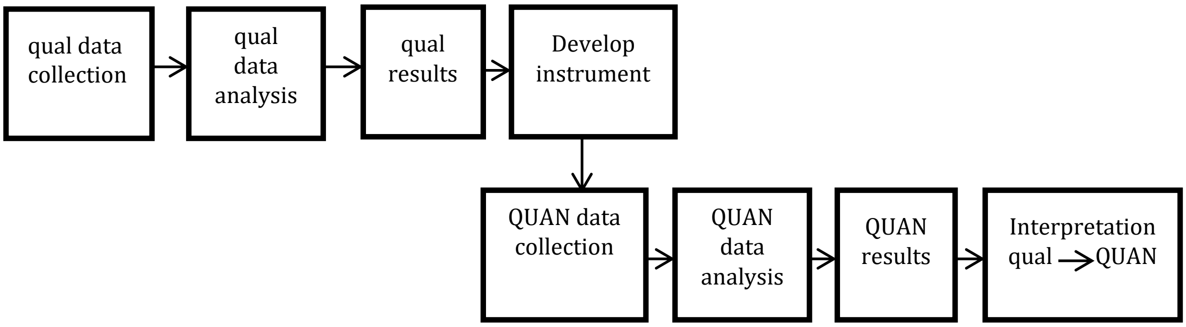


FIGURE 10 EXPLORATORY DESIGN: INSTRUMENT DEVELOPMENT MODEL, ADOPTED FROM (CRESWELL AND CLARK 2011).

The purpose of starting the design qualitatively is to explore and gain insights about a phenomenon and then build the second quantitative phase to help interpret the findings of the first phase. Researchers first qualitatively study and investigate the research topic with a few participants then the qualitative result (themes and codes) guide the development of a quantitative survey instrument. In this design, the qualitative and quantitative designs can be integrated through the development of the survey instrument (Creswell et al. 2003).

The Exploratory Design has some advantages due to its two-phase build and the fact that only one type of data is collected and analysed at a time. Notable advantages of this design include:

- The two separate phases of this design make it straightforward to conduct.
- The inclusion of the quantitative phase maximizes the qualitative approaches acceptability in quantitative-biased communities (Creswell and Clark 2011).

However, there are a number of challenges associated with such a design (Creswell and Clark 2011). Notable challenges include:

- This two-phase approach requires a large amount of time to implement. Researchers should be aware of this factor and allow enough time into their study's plan.
- Researchers will have to decide whether the same individuals will serve as participants in both the qualitative and quantitative phases.

- A decision must be made by researchers about the data to be used from the qualitative phase to develop the quantitative phase and how to use these data to generate quantitative measures (see Qualitative Phase findings).

The Sequential Exploratory Design is an appropriate candidate to adopt in this thesis. The exploratory emphasis of this design is a suitable fit to the problem investigated by this thesis. This is due to the lack of studies in the domain of this thesis which made it hard to make any assumptions of what is relevant to users and their perception of feedback requests. Starting the study qualitatively allowed to explore and get enough insights on how users behave and interact when receiving feedback requests from software applications. Adding a quantitative phase which takes more emphasis than the qualitative phase to the study allows to assess the interpretation of the qualitative findings and see what results to generalize to the targeted population. The design was clearly integrated in two stages, in designing the quantitative survey by building on the result of the qualitative phase and in the discussion of the overall results. Table 3, outlines the overall procedures of the adopted mixed method.

4.3QUALITATIVE PHASE DESIGN

Qualitative methods have been shown to be helpful in studying and gaining deep and better understanding of human behaviour (Stevens 2010). Since understanding the behaviour of users in relation to feedback acquisition in software applications is a high concern for this thesis, the qualitative phase is a good fit to adopt as a first phase in the methodology. This allows to get enough insights on users' behaviour to feedback acquisition in order to develop the second phase of the methodology.

Interview is one of the common methods of data collection in qualitative research (Gill et al. 2008). Interviewing can be distinguished from other qualitative data collection techniques in that it is much more exploratory in nature and much more flexible in location, scheduling and range of participants. It is also an effective method for gaining a deep insight and understanding of other's behaviour. For the previous reasons and the nature of the study which aims to understand people' behaviour and perceptions with regards to feedback acquisition in software applications, interviews were adopted as a data collection method in the first exploratory phase. This phase will guide the development of the second phase, the quantitative phase.

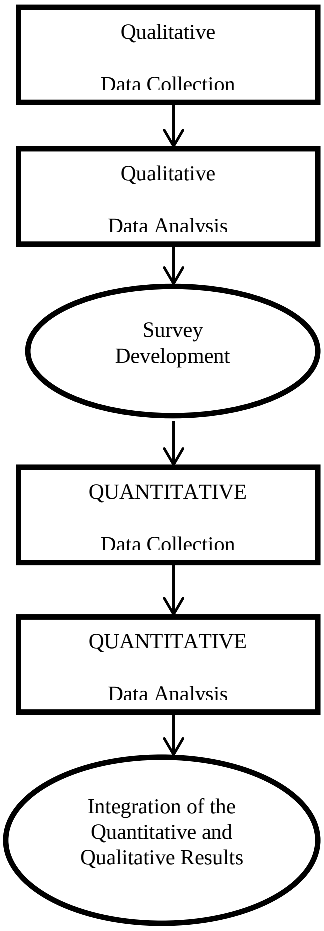
4.3.1 INTERVIEW DESIGN

Semi-structured interviews were conducted with 7 participants to explore their perception of software-mediated feedback acquisition. Participants were carefully selected in order to guarantee a high level of diversity and to avoid bias towards only certain group of people (e.g. different personal characteristic). These face-to-face interviews were conducted at Bournemouth University and each interview lasted for about 40 minutes.

An interview protocol was developed in consultation with the literature and the results of a group discussion which followed a research seminar given by the author on the topic. The protocol was revised after 2 initial pilot interviews. In the protocol, the first set of questions (Personal Information) was developed to ensure diversity in participants' personal characteristics which allowed for different types of people. The second set of questions (general software and computer familiarity) was developed to ensure that participants are familiar with software applications and computers in general and to ensure that the inclusion criteria are fully met by all participants. The last set of questions (experiences and behaviours) was developed to study participants' perception and attitude with regard to software-mediated feedback acquisition. Appendix 2 shows the interview script.

By combining the interviewee's answers, the author was able to draw foundations and develop a solid base that helped in the design of the second quantitative phase of the study. At the beginning of each interview session, each participant signed a consent form. The project was approved by Bournemouth University Research Ethics Committee (UREC). Each participant received £7 lunch voucher as an appreciation for taking part in the study.

TABLE 3 VISUAL MODEL OF THE MIXED METHODS SEQUENTIAL EXPLORATORY DESIGN PROCEDURES.

Phase	Procedure
 Qualitative Data Collection Qualitative Data Analysis Survey Development QUANTITATIVE Data Collection QUANTITATIVE Data Analysis Integration of the Quantitative and Qualitative Results	• Semi-structured interviews • Coding. • Using an inter-coder agreement check. • Generating themes. • Connecting, comparing and interrelating themes. • Developing of the survey questions based on the qualitative findings. • Random sampling for potential participants. • On-line self-selection questionnaire • Descriptive Analysis. • Cluster Analysis. • Interpretation and explanation of the quantitative and qualitative results.

4.3.2 SAMPLING

Purposeful sampling is a common technique in qualitative research (Creswell 2013). In this study, a purposeful sampling was used to select candidate participants to be interviewed. The study's inclusion criteria allowed for participants who are either students or university staff members coming from different backgrounds, within an age range of 18 to 50 and average computer users (various use of software applications e.g. internet browsing). The sampling criteria were developed to allow for more accuracy and variety in selecting participants. In addition, this study targeted the academia sector due to easy access to academic participants in different academic institutions all over the world. This enhances the study by gathering different perceptions and opinions regarding feedback acquisition in software applications.

In qualitative research there is no emphasis on the quantity of participants and the number of participants depends on the saturation point. Saturation is the process in which a continuous sampling of relevant cases should continue until no new theoretical insights are being gathered from the data (Baker and Edwards 2012). In this research 7 participants were interviewed which allowed to reach a reasonable saturation and get enough insights that allowed to develop the second quantitative phase. Table 4, shows the characteristics of the sample.

TABLE 4 THE CHARACTERISTICS OF THE INTERVIEWEES

Participants	Age	Gender	Education Level	Home Country
P1	19	Male	Undergraduate	UK
P2	29	Male	Postgraduate	Nigeria
P3	24	Female	Postgraduate	Nigeria
P4	19	Female	Undergraduate	KSA
P5	23	Male	Undergraduate	UK
P6	28	Male	Undergraduate	USA
P7	26	Female	Postgraduate	KSA

4.3.3 ANALYSIS

Interviews were audio taped and transcribed verbatim Analysis was performed in several steps which included: (1) initial exploration of the gathered data by reading the transcripts; (2) coding data by labelling and segmenting the text; (3) using an informal inter-coder agreement check to verify codes (the author and another researcher worked on verifying codes and a third researcher was approached for solving conflicts); (4) using codes to generate themes by gathering similar codes together; (5) connecting, comparing and interrelating themes. Credibility of the findings was maximized by the inter-coder agreement and academic advisor's auditing (Miles and Huberman 1994; Creswell 2013).

4.4QUALITATIVE FINDINGS

As previously stated the interview design covered different angles of user behaviour with regard to feedback acquisition in software applications. When analysing the data by grouping like-minded (similar in meaning or context) quotations, 31 codes and sub-codes began to emerge which were further grouped into 6 themes. Table 5 shows the themes, codes and sub-codes. Each theme consists of two or more codes/sub-codes which are briefly discussed and illustrated further with interview excerpts:

TABLE 5 A BREAKDOWN OF THE THEMES, CODES AND SUB-CODES OF THE ANALYSIS.

Theme1: Explicit Feedback Advantages [1.1] Evident channel for delivering users' voice and raising developers' awareness [1.2] Better for ethical reasons *This theme was deduced from interview questions such as (How do you think you should have been asked for feedback?, How would you prefer to be asked for feedback?, etc)*
Theme2: Motivation for Feedback Provision [2.1] Visibility of feedback’s effect on the system [2.2] Usability and simplicity: o Language used [2.3] Visibility of opposite feedback [2.4] Reasonable number of feedback requests [2.5] The exciting nature of feedback subject [2.6] Positive experience [2.7] Negative experience [2.8] Less interruption and distraction [2.9] Device used [2.10] Raising public awareness [2.11] Being forced by the software: o Low quality feedback *This theme was deduced from interview questions such as (Tell me what do you think of the idea of asking users for feedback when using software applications (e.g. Skype quality feedback popup after a call)?, Do you remember a time you were asked for feedback, tell me about it?, Did you provide the feedback? Why (what was your motivation)? Why not?, Have you ever been annoyed when you were being asked for feedback? Why?, How would you prefer to be asked for feedback (e.g. when, popup)?,etc)*
Theme3: Feedback Acquisition Methods [3.1] Email: o More personalized o More preferable for qualitative feedback o More time space and less interruption o Reasonable number of feedback requests [3.2] Passive feedback requests [3.3] Quantitative feedback forms [3.4] Combination of qualitative and quantitative (not only quantitative) *This theme was deduced from interview questions such as (Have you ever been annoyed when you were being asked for feedback? Why?, How do you think you should have been asked (e.g. by email at your free time)?, How would you prefer to be asked for feedback (e.g. when, popup)?, etc)*
Theme4: Pause of Feedback Requests [4.1] Same feedback is given [4.2] Lack of interest [4.3] Passive feedback is preferable *This theme was deduced from interview questions such as (Have you ever received multiple requests at different times to provide feedback? Tell me about it?, Do you remember a time you were asked for feedback, tell me about it?, Do you remember if you were asked for feedback before this time and what did you do?, etc)*
Theme5: Timing for Feedback Requests [5.1] Enough time before requesting feedback

[5.2] On recent service or product [5.3] Reminder is needed [5.4] Avoid work time or hours *This theme was deduced from interview questions such as (Do you remember where you were at that time (example, at work)?, Do you remember the time you were asked for it (e.g. afternoon)?, How were you asked for it and in which context (example, during playing a game by popping up a feedback form)?, etc)*
Theme6: Feedback Visibility [6.1] Ability to see what others said [6.2] The trend of current feedback *This theme was deduced from interview questions such as (How do you think you should have been asked for feedback?, How would you prefer to be asked for feedback?, etc)*

4.4.1 EXPLICIT FEEDBACK ADVANTAGES

Participants were asked about the value and advantages of being explicitly asked for feedback. The majority of participants emphasized two different aspects as the core advantages of explicit feedback:

- [1.1] Evident channel for delivering users' voice and rising developers' awareness *"Explicit feedback would be a lot better. That's why most people that release software go to conferences and tell software programmers, game reviewers, and different people to talk about the software they produce".*
- [1.2] Better for ethical reasons *"I think from an ethical standpoint, explicit feedback is much friendlier to a user because they know exactly what they're submitting".*

4.4.2 MOTIVATIONS FOR FEEDBACK PROVISION

The past experiences of the interviewees with feedback requests enabled them to identify the factors that have a noticeable effect on their motivations to give feedback on a software service or a product. The following dimensions of motivations were extracted from participants' responses:

- [2.1] Visibility of feedback's effect on the system.

Some participants indicated that being able to see the impact of their feedback on the software plays a core role in motivating them to give feedback *"The problem that I always have when I'm asked for feedback is; does my feedback really count?"*

- [2.2] Usability and simplicity

Another factor that motivates users to give feedback is the simplicity, and usability of the method and the language used to get their feedback *"why do I have to give a four paragraphs of feedback on a product when I can just rate or rank it", "I think it is all about the word you use versus a word that might intrigue people".*

- [2.3] Visibility of opposite feedback

An interesting factor that can increase users' motivations to provide feedback is the user' ability to see different types of feedback that conflict with their own perception or opinion about a product or a service provided by a software application "*If negative feedback is given for example by a seller on eBay and I had positive experience with them then Yes I would give feedback to I could warn people that there's a chance that you might like this seller".*

- [2.4] Reasonable number of feedback requests

Receiving a large number of feedback requests might lower users' motivation to respond to feedback requests due to the annoyance that it typically causes *"I think receiving emails all the time asking for feedback is quite bothersome because you're getting loads of emails in your inbox".*

- [2.5] The exciting nature of feedback subject.

Being obsessed or passionate about a product or software service plays a role in motivating some users to give feedback or write a review about it *"I give feedback on things that make me happy. But when it is just like gloves and shoes and regular things, I do not even do that".*

- [2.6] Positive experience

Being significantly happy and satisfied with a product or a software service greatly increases users motivation to review or give feedback about it *"I bought a cooling pad for my processor and I wrote a couple of things, but not so much so I just wrote that like how it came really quickly and it was really good, it fits perfectly, it is cheap.".*

- [2.7] Negative experience

Interestingly all respondents agreed that being unhappy, unsatisfied or in need for improvements with a product or software service is a key factor that drives them to give feedback about such as raising complaints *"On eBay when I didn't get my item in the end I had to leave a negative feedback".*

- [2.8] Less interruption and distraction

"I wouldn't suggest anyone to pop up into my window when I'm busy doing some work. It is just an offense".

- [2.9] Device used

"If you ask me for a feedback that requires me to write many sentences while using my smartphone then surely I won't reply. But if I was using my pc then I might respond".

- [2.10] Raising public awareness

"The reason I gave feedback was because I wanted to make sure that everyone learns about this specific service and its negative and positive sides".

- [2.11] Being forced by the software

An observation was made by many respondents on being forced by the software to give feedback (e.g. popup dialogs) as a motivation to respond to these requests *"it is just a kind of gentle way to force me to give feedback".* However, being forced to give feedback can cause a low quality feedback *"I would give a low rating from the frustration or high rating just to get it away from me".*

4.4.3 FEEDBACK ACQUISITION METHODS

Participants were asked to recall some of their past experiences with regard to feedback acquisition and their reaction to it in order to extract their preferred method. Various methods were extracted from their responses:

- [3.1] Email

Emails give more time for users to respond to feedback requests as well as less interruption. They are also better for qualitative feedback *"I think emails are good if you want someone to actually sit down and write a couple of sentences about how they feel about your service".* Emails are also more personalized *"If you say something that is meant to be only for me such as asking me by my name then I would responds, why not?"*

- [3.2] Quantitative feedback request

"I prefer multiple choice or ratings because it is just easier, simpler, and faster".

- [3.3] Passive feedback forms

This thesis defines passive feedback as a feedback that is given by the user on voluntary bases and without being asked to do so by the software. An example can be a rating panel to the left side of a website page. One participant indicated that having feedback requests sent to them by all means (i.e. emails, popups) is annoying and a feedback channel to deliver their voice when needed should be passively available (e.g. feedback form in a website) *"I find it problematic to send me any kind of feedback requests. If I'm not happy with something I will go to their website and complain right to them".*

- [3.4] Combination of qualitative and quantitative

"In the real world, the best way is to sort of have a combination of both qualitative and quantitative "Rate and then If you have any other comments, please leave them below".

4.4.4 PAUSE OF FEEDBACK REQUESTS

From the participants' responses this theme was created which encapsulates the dimensions in which sending or asking for feedback should be paused or stopped:

- [4.1] Same feedback is given

When the same feedback or response is given by the user (i.e. same rating each time) "*I've done it like once or twice but it usually comes up a lot. It just keeps popping up. If I keep sending them the same answer then there would be no point of asking again".*

- [4.2] Lack of interest

When there is an ignorance or lack of users' interest in the software service or product *"I find it annoying is when I'm using my phone and I do not really care to rate an application and it keeps saying; will you please rate this application?"*

- [4.3] Passive feedback is preferable

When users do not prefer to be proactively asked for feedback and would rather passively give feedback.

4.4.5 TIMING FOR FEEDBACK REQUESTS

All participants indicated that timing for sending feedback requests is a critical factor that can affect their response to the requests. Participants indicated some timing factors that should be taken into account when initiating a feedback request by software:

- [5.1] Enough time before requesting feedback

Enough time should be given to the users to familiarise them with the service or product before asking for feedback in order to maximise the quality and truthiness of the feedback *"I think a user needs some time to really get a good evaluation of what they're using before they are asked for feedback"*.

- [5.2] On recent service or product

Feedback should be requested on a product or a service that the user has used recently so that the user is still interested in it and find some excitement in expressing their opinion.

- [5.3] Reminder is needed

Some users need to be reminded to respond to feedback requests *"I do not delete feedback emails so that I'll remember to come back to it but sometimes I kind of forget to come back"*.

- [5.4] Avoid work time or hours

Asking for feedback when users are busy might affect their willingness to respond to these requests *"I wouldn't expect myself to probably send a feedback during working hours"*.

4.4.6 FEEDBACK VISIBILITY

- [6.1] Ability to see what others said

Being able to see others feedback and reviews is a positive mechanism that affects users' perception about certain software service or a product.

- [6.2] The trend of current feedback

Visibility of others feedback/reviews could also motivate users to give feedback or write a review about the provided service.

Some of the themes explained above are interrelated with each other. For example, being motivated to give feedback and being targeted by the wrong feedback acquisition method makes users lose interest in responding. Another example is that, being motivated and targeted by the right acquisition method but at the wrong time makes users ignore responding to the feedback request and vice versa.

The interviewees described their behaviour and past experiences with regard to feedback acquisition in software applications. Analysing the qualitative data resulted in six dominant themes. The six themes/dimensions were used as a foundation for developing a quantitative measure. The next section describes the process used for developing, conducting, analysing and reporting the second phase of the study.

4.5QUANTITIATIVE PHASE DESIGN

The aim of the quantitative phase is to combine the unique strengths of both qualitative and quantitative paradigms to further investigate and assess the finding from the first phase and allow for more generalizability. For this phase, a questionnaire with 31 questions with various types (i.e. single choice and multiple choice questions) was developed. Findings from the qualitative study served as the basis for developing the survey's script with regard to users' perception and reaction to feedback requests.

The questionnaire was first piloted on six participants who met the sampling criteria. The feedback from those participants was used to revise and refine the questionnaire before distributing it to the larger sample of participants. The revised and refined questionnaire was then sent by email to, selected students and staff members. The invitation email contained a brief description of the purpose of the questionnaire and asking them to take part in the study. The e-mail also contained a web link to the questionnaire and instructions for accessing it. The questionnaire itself started with an introduction to the topic of interest so as to familiarize the participants with the subject matter. Participants were also informed about what is expected from them and how the results of the questionnaires will be used. The data collection took place between September 15 and November 16, 2013. Five days after distributing the questionnaire, e-mail reminders were sent to participants who did not respond to the invitation. The questionnaire and data gathering went through the ethics approval process and was cleared by the Bournemouth University Research Ethics Committee (UREC). Appendix 3 shows the questionnaire submitted to participants.

4.5.1 SAMPLING

A simple random sampling approach was used to recruit participants. The advantage of the simple random sampling method is that it minimizes bias in selecting participants and the result are generalizable to other populations groups (Tashakkori and Teddlie 1998). An access to students and staff members email contacts at Bournemouth University was gained by the author and a computer software program was used to generate and extract a random set of emails and mailing lists that was then used as a selected sample.

In order to counter-balance the geographic and demographic homogeneity of the on-campus participants, a convenience sampling technique was used to recruit more participants from different countries such as Egypt, KSA, Ireland, China and the Netherlands. Table 6 shows the characteristics of the participants. A total of 150 participants started the survey and 100 completed forms were returned. When considering the average time to complete the survey (25 minutes), the size of the form and the amount of effort required completing it, the author considers this number of participants to be a good rate of return. The survey was closed once reached 100 participants. The author considered this as a reasonable number of responses especially that the initial analysis of participants responses at that stage showed that some clear trends and cluster were already established.

4.5.2 ANALYSIS

The returned questionnaires were analysed and cleaned up and irrelevant and inconsistent responses were excluded. A statistical analysis of the survey was conducted to describe the data (Williams 2003). Then a cluster analysis was conducted to group similar users into initial clusters according to their behaviour to feedback acquisition in software applications. The statistical analysis was carried out using Qualtrics (www.qualtrics.com) which is widely used online survey software. The cluster analysis was conducted using Weka tool (Hall et al. 2009). Weka is a data mining tool that is widely used, free, open source Java application and is readily available on the internet. It provides algorithms and computational paradigms that allow computers to discover structure in databases and perform predictions.

4.6 QUANTITATIVE FINDINGS

The participants' demographics were analysed using cross tabulation and frequency counts and then summarized. Participants where compared over their age, gender, level of education and country. From this analysis a high level of diversity among participant is clearly presented maximizing the generalizability of the findings. For example, participants come from different countries (19 different countries such as KSA, Brazil, UK, Iran, Germany, and USA), with different ages, genders and levels of education as shown in Table 6.

TABLE 6 PARTICIPANTS' DEMOGRAPHIC INFORMATION

		Age Range					**Gender**		
		18-25	**26-34**	**35-54**	**55-64**	***Total***	**Male**	**Female**	***Total***
level of education	**High school**	3	0	0	0	3	3	0	3
	Associate degree	0	0	0	0	0	0	0	0
	Bachelor's degree	9	3	6	0	18	13	5	18
	Master's degree	6	36	10	3	55	30	25	55
	Professional degree	0	0	1	1	2	1	1	2
	Doctorate degree	3	11	5	0	19	10	9	19
	Others	0	3	0	0	3	2	1	3
	Total	**21**	**53**	**22**	**4**	**100**	**59**	**41**	**100**

In addition and as shown in Table 7 and Table 8, the majority of the participants represent a typical set of software users who use a typical and diverse set of popular software applications rather than domain specific software (e.g. desktop applications) for everyday life activities (e.g. Movie Players). Therefore, their feedback reflects their experience with popularly used software applications. This supports the generalizability of the finding.

TABLE 7 THE TYPE OF SOFTWARE FREQUENTLY USED BY THE PARTICIPANTS

Software Applications	Usage Rate
Desktop Applications such as MS Office, Movie Players, etc.	**94%**
E-commerce such as Ebay and Amazon, Online Shopping, etc.	**73%**
Mobile Apps: applications installed on your mobile	**84%**
Search Engines such as Google, Skyscanner for flights and Venere and Booking for hotel reservation, etc.	**92%**
Social Networking such as Facebook, Twitter, Wikis as an editor, etc.	**81%**
Web Applications such as online documents editors like Google Doc, online calendars, storage services like DropBox, etc.	**80%**

TABLE 8 THE TYPE OF ACTIVITIES SOFTWARE APPLICATIONS FREQUENTLY USED FOR.

Activities Performed	Usage Rate
For professional reasons: their work requires that	**63%**
For academic reasons: their study/research requires that	**84%**
Daily life activities (booking, online shopping, looking for bus schedule, taxi number, etc)	**86%**
Entertainment (gaming, social networks for entertainment purposes, etc.)	**64%**
For social interaction (social networks, blogs, forums, etc.)	**75%**

4.6.1 FEEDBACK ACQUISITION METHODS AND FEEDBACK TYPES

Before digging deeper into users' preferences with regard to the methods that have been used to collect their feedback, this study investigated whether they actually like to be asked for feedback explicitly. 70% of the participants provided negative responses. This reflects the high need for novel mechanisms to increase users' engagement as evaluators of software applications. This also provides strong evidence that the current explicit feedback acquisition processes are poorly engineered and conducted. This suggests that new novel mechanisms and engineering approaches are needed to change users' negative views toward feedback acquisition. Some comments from participants further explain the logic and motivation behind their negative views (see Table 9):

TABLE 9 SAMPLES OF USERS' COMMENTS ON FEEDBACK REQUESTS

Users Comments
"I find it hindering and unprofessional."
"They often ask several times about the same thing."
"The benefits are always not clear to me as a user."

Participants were further asked to choose their preferred types of feedback and methods used for gathering such feedback. A number of answers were extracted from their responses which highly confirm and enhance the previous findings in the first phase of this study (see qualitative findings). Participants' answers vary over the following items:

4.6.1.1 FEEDBACK TYPES

Explicit feedback: such as sending emails to users asking their feedback. Explicit feedback includes:

- **Qualitative feedback** which is preferred by 9% of participants. An example of this type would be writing sentences or lines of texts to communicate users' thoughts in a free-style.
- **Quantitative feedback** which is preferred by 48% of the participants. An example of this type would be rating and giving stars to a set of quality attributes such as comfort, and speed.
- **A combination of qualitative and quantitative** which is preferred by 55% of the participants. This means the user has the choice to use their preferred one such as rating and giving stars with the ability to add text if they need.

Implicit feedback: such as monitoring and analysing users' usage of the software application. Participants showed less interest in implicit feedback. Only less than 20% choose implicit feedback as their preferred method. Privacy issues and ethical related factors could be the reason behind this low interest. For example, a user commented that *"The implicit modality is also interesting, but I tend to be concerned with my privacy and disallow this option"*.

4.6.1.2 ACQUISITION METHODS

- **Passive feedback acquisition** method is preferred by51% of participants. In this method users submit their feedback on a voluntarily base and without being proactively asked by the software (i.e. through a contact us form).
- **Offline feedback acquisition**[7] method is preferred by 33% of the participants. In this method users submit their feedback offline (i.e. after using the software). An example of this method would be sending the user an email or an SMS message asking for their feedback.
- **Online feedback acquisition** method is preferred by 54% of the participants. In this method users provide their feedback online while using the software. An example of this method could be showing the user a feedback popup dialogue while using the software.
- **Using Hints or tips** as a method to collect users' feedback is preferred by 31% of the participants. An example would be showing a user a hint message telling the users that they can go to a feedback centre such as a forum specifically designed for this purpose and leave their feedback.

[7] Offline feedback acquisition (as opposed to online feedback requests) refers to the feedback requests that do not interrupt users while using the software. For example, sending the user an email or an SMS message asking for their feedback after they have used the software.

Table 10 shows some comments from the participants further explain their logic and motivation of their choices:

TABLE 10 USERS' COMMENTS ON FEEDBACK ACQUISITION METHODS.

Users Comments
"Definitely online, real-time sounds ok, but this really depends on what I am doing and how much time I have to be altruistic."
"I prefer 'pull' over 'push' - emails etc I can pull when it is convenient; popups and other 'push' mechanisms intrude & interrupt flow."
"I hate popups. I prefer seeing noticeable small box somewhere I can see on the website asking me to leave a feedback."

4.6.2 MOTIVATIONS FOR ACCEPTING/IGNORING FEEDBACK REQUESTS

Further analysis of the participants' responses extracted the following dimensions of motivations which highly enhance and confirm the previous findings (see qualitative findings):

4.6.2.1 USERS' EXPERIENCE

The first set of motivations that were identified by users is related to users' experience with feedback requests that can negatively or positively affect their willingness to respond to a feedback requests. Users' responses varied over the following factors:

- **Simplicity of feedback requests** was indicated by 64% of the participants as a key factor that can highly influence their willingness to give feedback. An example of this factor could be the time a feedback request requires a user to think about and answer.
- **Timing for feedback requests** was also indicated by 75% of the participants as a vital factor that can greatly affect their willingness to give feedback. For instance, when a user is engaging with some other activity they might not respond.
- **Awareness of the usage and impact of the feedback on the system** was mentioned by 54% of the participants as a valuable factor in motivating them to give feedback. Users can be motivated to give feedback if they are able to know how their feedback would be used and whether it has been taken into account to improve their experience or led to any changes.
- **Privacy** is also an important factor that can affect the willingness of around 31% of the participants. An example of a privacy factor that can decrease a user' willingness to give feedback could be the ability of others to see or infer the user's given feedback.
- **Familiarity** with the software was also indicated by 42% of the participants as an important factor that can affect their willingness to give feedback. For instance, users are more motivated when they have enough experience with the software application before responding to feedback requests.

Although users find it against their privacy to use their implicit feedback (see qualitative findings: explicit feedback advantages) , privacy issues do not seem to play a significant role in motivating users to respond to explicit feedback invitations as indicated by the low percentage given by users to privacy reasons (31%). Timing for feedback request has been also highly emphasized by both users and experts as discussed in (see Chapters 3, Section 3.3.2) to be a highly important factor that can negatively or positively affect users' willingness as well as the quality of their feedback. Table 11 present participants' responses when asked the following question:

"How do you feel about the feedback requests which come at the wrong time (a popup dialogue when you are navigating a website and moving to another page, a hint in a YouTube video to encourage you to rate it)?"

TABLE 11 USERS' ANSWERS IN REGARD TO WRONGLY TIMED FEEDBACK REQUESTS.

Answers	Percentage
Are OK with me	**7%**
I think this is one of the ways which puts a gentle pressure on me so that I give feedback	**11%**
Decrease my willingness to give feedback	**58%**
I may give less truthful feedback just to get rid of the dialogue	**26%**
I believe it is an inconsiderate way to force me to give feedback	**43%**

4.6.2.2 INTERFACE DESIGN

Users emphasized the impact that interface design has on their behaviour and willingness to give feedback. The interface design can easily increase or decrease users' willingness to respond to feedback requests. It can also positively or negatively affect the quality of the feedback as discussed in (Chapters 3, 4). Users' responses varied on the following design factors:

- **Language used** in the feedback request (i.e. friendliness, succinctness and clarity) was indicated by 52% of the respondents as a design factor that can influence their willingness to give feedback. This reflects the need and importance of carefully wording feedback requests and selecting the right language that fits the context of use and the type of users - e.g. formal language might be more suitable for professional users.
- **Graphical design** of the feedback request (i.e. font size, colours and the kind of photos used in the acquisition interface) was also indicated by 31% of the respondents as a design factor that plays a modest role in motivating them to give feedback. The low percentage reflects users' need for simple and straightforward feedback requests that are not full of graphically complicated presentations. However, this does not exclude the need for feedback requests that are graphically displayed in an attractive and reasonable way (e.g. readable font size).
- **Simplicity and complexity** degree of the method used to provide the feedback (i.e. clicking, a voice message, text with/without auto-completion) was mentioned by 74% of the participants as a factor that plays a vital role in motivating them to give feedback. The high percentage highlights the fact that users always avoid spending too much time and effort responding to feedback requests that require too much time and effort to submit their input. This reflects the importance of keeping feedback request processes and interfaces simple and straightforward as much as possible to leverage users' response rate and the quality of feedback

- **Fitness of the design and content** of feedback request to the context of use is a significant factor that can greatly influence users'' willingness to give feedback as indicated by 81% of the respondents. Example of such a factor could be - showing less details and simpler content when the user is using a smartphone. The relatively high percentage reflects users' frustration about feedback requests that do not take the context of use into consideration - e.g. presenting a complex and detailed feedback interface when the user is using a smartphone. This also indicates the importance of the context of use in relation to the interface design of user feedback requests.
- **Information provided** is considered as a modest factor that can also affect users' willingness to give feedback as mentioned by 24% of the participants. Example of such a factor could be showing users a summary or statistics of other people's given feedback. Although the information provided by the feedback request interface is not highly important for some users, it is still considered by some users as an encouraging factor to respond to feedback requests. A dynamic feedback interface that shows a runtime statistics and summaries of feedback already given on a certain aspect of the software is still a valuable interface design factor that can empower user response rate and satisfaction.

4.6.2.3 SOCIAL FACTORS

Almost half of the participants indicated that several social factors, when considered in feedback acquisition, can noticeably affect their willingness to give feedback and their level of engagement with the software application. The following factors were emphasized:

- **Visibility and similarity of others feedback**: 47% of the participants indicated that being able to see others feedback and compare it against their own opinion about a service or a product can greatly affect their willingness to give feedback. For example, the majority of participants indicated that being able to see other people's feedback first and then compare it against their own opinion and having the option to accept or reject to give feedback can encourage them to give feedback. A user commented: *"Giving feedback is a community experience and it helps to feel among others"*. However, the rest of indicated that visibility and similarity of others feedback will have no effect on their willingness to give feedback by all means.
- **Volume of already given feedback**: 52% of the participants indicated that the volume of already provided feedback on service or a product can affect their willingness to give feedback. For example, participants' willingness to give feedback increases when there are only few people who provided feedback on a service or a product. The remainder indicated that the volume of already given feedback has no effect on their willingness to give feedback by all means.

- **Social recognition**[8]: 57% of the participants indicated that being socially recognized as a feedback provider is an important factor that can increase their willingness to give feedback and engage more with the software application. There could be some constraints on this though. For example, some participants emphasized that, it is nice to be visible only when others can see their feedback which led to some changes on the system. The rest of the participants indicated that social recognition has no effect on their willingness to give feedback by all means. This could be due to privacy reasons as some users commented*" I do not like others in my social network to see my feedback. I want to remain anonymous"*, *"I am less likely to leave feedback if I am easily identifiable".*
- **Feedback acquisition as a social activity**: 63% of the participants indicated that feedback acquisition as a social or game activity is not an important factor that affects their willingness to give feedback. Example of such an activity could be the users' ability to visualize how their direct and indirect social contacts are rating a certain service and how their feedback influenced the trend in their community. This negative response could be due to the desire for simplicity of feedback acquisition process. For example, a user commented: *"I would generally say No. If I want to give a feedback, it would be feedback alone. I usually do not want any continuation from there."*

However, the rest of the participants showed a positive interest in such an activity and even suggested some ideas on how to conduct the feedback acquisition as asocial or game activity. A user commented:*" maybe a chat feedback dialog box would be nice (sending the feedback live and looking at different users sending feedback at the same time). I do not like it to be more complicated or time consuming."*

The previously mentioned factors provide a clear vision about the conflicts and variety among users' behaviours and preferences with regard to feedback acquisition and the related social factors. This highlights the need to have a systematic way to develop such an activity that fits all different behaviours and preferences of users.

4.6.2.4 VOLUME AND FREQUENCY OF FEEDBACK REQUESTS

Participants indicated that if the frequency or volume of feedback request from one software application (e.g. smartphone app) is very high it might result in a negative reaction from them towards that software application (i.e. stop using the software application) and can also reduce their willingness to respond to feedback requests (see Table 12). This is indeed a an important issue that should encourage software developers to systemize the volume and frequency of

[8] This thesis defines Social Recognition in the contexts of feedback acquisition as the public acknowledgment or visibility of a feedback provider. For example, a feedback provider can be socially recognized, seen or visible (e.g. by giving them a badge, highlighting their name as an influential feedback provider, etc) when their given feedback led to some changes on the system.

feedback requests sent from a software application in a way that does notcause a negative reaction by users but empower their engagement and response rate to feedback requests.

TABLE 12 EFFECT OF HIGH FEEDBACK REQUESTS VOLUME ON USERS

Effect of High Feedback Requests Volume on Users	Percentage of Users
It is fine with me, I like to give feedback often	**3%**
It is fine with me as long as I am not forced to give answers	**13%**
I tend to respond to some of them	**14%**
I tend to give less focused or less truthful feedback	**10%**
It leads me to give a negative feedback as the requests make me feel annoyed	**7%**
I tend to ignore all of them and I tend to consider it as a spam	**53%**
I tend to stop using the software sending me these requests	**21%**

4.6.3 USERS' CLUSTERS

Since there is a high level of variety among users' behaviours and preferences with regards to feedback acquisition in software applications, cluster analysis was used to further discover natural groupings in the data and to group similar participants together. The K-means clustering method was adopted (Alsabti et al. 1998). K-means clustering is one of the widely used techniques to analyse a given set of data in order to produce meaningful clusters that can explain the natural grouping in data (Zwitch 2013). The initial clustering of participants served as an initial guide for the feedback acquisition process in which each group of similar users can be approached for feedback in a way that fits their preferences.

4.6.3.1 MAIN CLUSTERS

After conducting an intensive cluster analysis on the collected data, four main clusters that represented different groups of users were extracted. These initial clusters expressed the grouping criteria among users thus their behaviour with regard to feedback acquisition.

Some variables such as age, nationality and level of education showed no significant influence on users' behaviour with regard to feedback acquisition. As shown in Table 13, the most influential clustering variables that drive users' behaviour with regard to feedback acquisition are:

- Users' acceptance of feedback requests (likeness of being asked for feedback).
- Methods used to gather users' feedback (e.g. offline by sending an email).
- Users' preferable type of feedback (e.g. explicitly asked for feedback).
- Users' acceptance of being reminded to respond to a feedback request.
- Social Variables:
  - Visibility and similarity of others feedback and its effect on users' willingness to give feedback.
  - Volume of already given feedback.

- Social recognition of feedback providers and its effect on users' willingness to give feedback.
- Feedback acquisition as a social activity and its effect on users' willingness to give feedback.

The previous drivers/variables for users' behaviour in regard to feedback acquisition are highly correlated and can collectively influence a user's assignment to a particular cluster. These variables have shown a correlation coefficient of 0.8483 which considered to be a relatively high correlation. A logistic regression analysis was also conducted to predict the accuracy of the extracted clusters and users' assignment to a particular cluster (e.g. the accuracy degree in which a USERx belongs to Clusterx). The above users' behaviour clustering variables (see Table10) were used as predictor and an overall prediction accuracy of 89.5% was achieved which is considered to be a good rate of accuracy.

4.6.3.2 CLUSTERS DESCRIPTION

Cluster 1 (*feedback antagonists*) and Cluster 2 (*passive and stingy people*): these clusters represent a group of users who have negative views/perceptions towards all feedback acquisition methods. These user groups prefer not to be asked for feedback or to be reminded about it. Even social factors have no noticeable effect on their willingness to give feedback. The only thing that differentiates between both groups is the feedback acquisition method that they prefer if they were to be asked for feedback. In cluster 1, users prefer ***online*** methods such as feedback popup dialogue while using the software. However, in cluster 2 users' first preferred method is the ***passive*** one whereas their second preference is the online method. However, the two cluster groups have very similar negative views/perceptions about feedback acquisition. This encourages software developers to seek and tailor ways to fit these groups in order to change negative perceptions into a positive one and engage them more with the software.

TABLE 13 INITIAL CLUSTERS OF USERS' BEHAVIOUR TO FEEDBACK ACQUISITION.

	N	Likeness to be asked	Method	Explicit/ Implicit	Reminder	Visibility-Willingness increases	Social Activity-interest	Social recognition-willingness increases-impact	Feedback Volume	Feedback Similarity
Cluster 1	38	No	Online		No	No	No	No	No	No
Cluster 2	27	No	Passive+ Online		No	No	No	No	No	No
Cluster 3	21	Yes	Offline	Very Explicit	Yes	Yes_ If able to see others feedback first	No	Yes	Few-increase	50%
Cluster 4	14	No	Hint+ Online	Implicit is also OK	No	Yes_ If able to see others feedback first	Yes	Yes	Large-increase	Similar-increase

Cluster 3 (*privacy fanatic and generous people*): this cluster represents the most positive users group among the 4 clusters. Users in this group do not mind to be asked ***offline*** for feedback and even sent a reasonable number of reminders to respond to feedback requests. However, they are very concerned about their privacy and therefore they put a great emphasis on the importance of asking them for feedback ***explicitly*** rather than in an implicit way (e.g. implicitly collecting information about their software usage). In addition, users' willingness in this group (to give feedback) is positively affected by the following social factors:

- **Volume of already given feedback**: users in this group indicated that a high number of feedbacks given on a service or a product empower their willingness to give feedback (positive correlation).
- **Visibility of other users' feedback**: They indicated that being able to see other people's feedback first and then having the option to accept/reject to give feedback can encourages them to give feedback.
- **Social recognition**.

Cluster4 (*privacy tolerant and socially ostentatious people*): this cluster represents the second positive users group among the 4 clusters. Users in this group do not like to be asked for feedback or reminded about it but their willingness to give feedback is highly affected by some social factors/variables such as similarity and visibility of others feedback. They also do not mind to be ***implicitly*** reached for feedback (e.g. implicitly collecting information about their software usage). However, their first preferable method (if they were to be asked for feedback) is using ***hints and tips*** to gather their feedback (e.g. by telling them that they can go to a feedback centre for this purpose and leave their feedback) whereas their second preference is the ***online*** method. Additionally, users' willingness in this group (to give feedback) is positively affected by the following factors:

- **Volume of already given feedback**: users in this group indicated that a low number of feedbacks given on a service or a product empower their willingness to give feedback (negative correlation).
- **Visibility and similarity of other users' feedback**: They indicated that being able to see other people's feedback first (that are similar to their feedback/opinion) and then having the option to accept/reject to give feedback can encourages them to give feedback.
- **Social recognition.**
- **Feedback acquisition as a social activity.**

The previous different types of users' clusters reflects the need to have an adaptive feedback collection mechanism which can highly empower and improve different aspects such as users' satisfaction, feedback quality, users' engagement with the software, software adaptation quality, etc.

4.7 DISCUSSION

This study of users' behaviour with regards to feedback acquisition provides a clearer view and a deeper understanding of users' behaviour and perceptions. It also answers the questions that were highly ambiguous and unknown about users' reactions to feedback requests such as what motivates users to give feedback and why some users hold negative views about feedback.

Combining the findings of the first phase study and the second phase study, the results showed that users' perceptions and behaviours with regard to feedback acquisition significantly vary and are affected by a number of factors. The variety among users' behaviours and the diversity of contextual information and design elements which affect that perception highlights the great need for an adaptive feedback acquisition process which accommodates such variety in autonomous or semi-autonomous way. It also raise awareness that feedback acquisition systems would need to be richer and, amongst other things, allow users to configure the way to receive feedback requests and express what information they would like to know before they give feedback and whether they want to see the effect of their feedback on the quality of service or the decision for the next release of the system. .

The results show that there are a number of main factors (which encapsulate a number of sub-factors) that noticeably influence users' behaviour with regard to feedback acquisition in software applications. These main factors are; Feedback Acquisition Methods, Feedback Types, Users' Experience, Interface Design, Social Factors, Volume and Frequency of Feedback Requests. These behavioural factors should be highly considered by software developers at the early stages of the feedback acquisition development process.

Having an adaptive feedback acquisition that can cater for such diversity is needed to make users looks more positively to feedback requests. This will have a positive side effect on the feedback quality and truthfulness, users' involvement as decision makers, users' satisfaction and trust in the system. The good feedback, in quantity and quality, will increase developers knowledge about their users and software and software adaptation and success and help them decide how to evolve it or adjust it to enhance its role in meeting users expectations (see Chapter 3, Section 3.3.1).

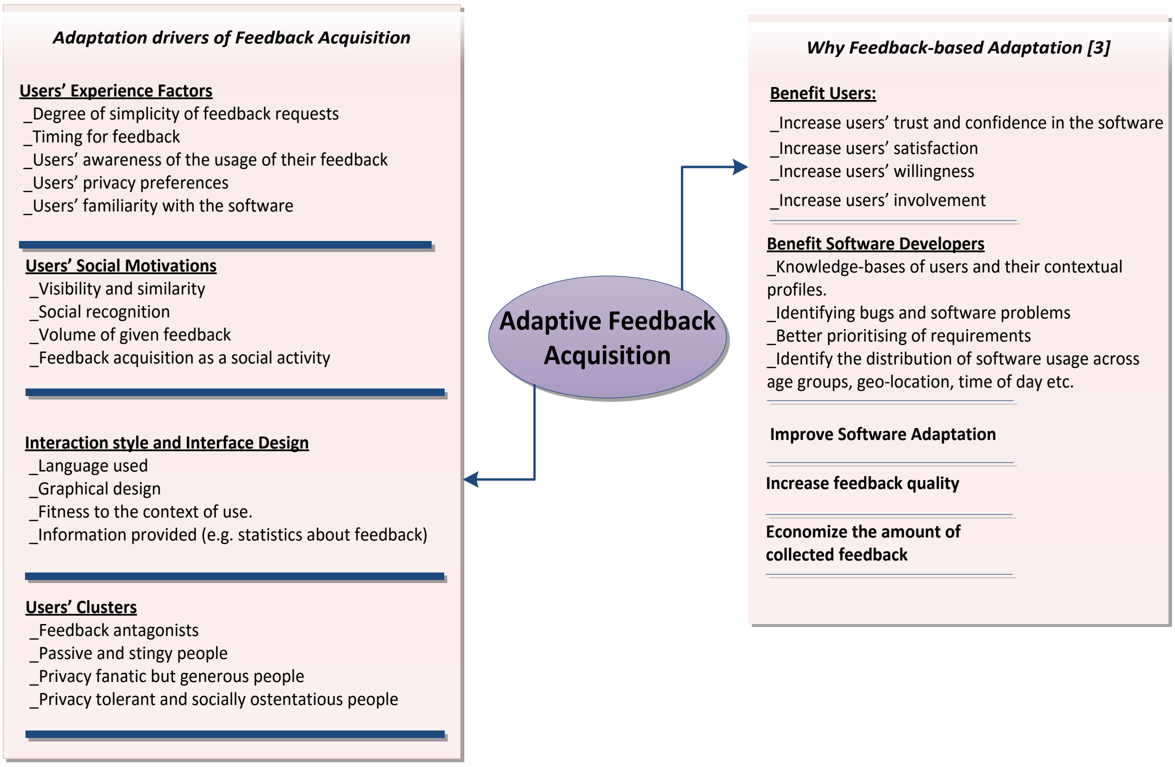
Figure 11 presents an initial application-independent conceptual framework for the design of an adaptive feedback acquisition. It summarizes the expert survey findings in Chapter 3 that are related to the motivation of an adaptive acquisition of users' feedback (right side of the figure). In Chapter 3, experts in software engineering agreed that availability of an adaptive feedback acquisition is a necessary enabler to decide ways of acquiring feedback and to empower the success of socially-adaptive software in particular and software systems in general. It also summarizes the findings of this chapter and depicts the adaptation drivers from users' perspectives.

Though the principles in conducting mixed methods approach were carefully followed, this study would still have three main threats to validity:

- While the methodology was effective in identifying and describing users' behaviour and perception with regards to feedback acquisition, it is possible that it did not identify all the important aspects and factors that can affect and influence their behaviour in this regard.
- One of the most common issues when designing a questionnaire is to know whether the questions were understood by all participants as intended and in a similar way to one another. This threat was somehow addressed as a pilot test was conducted on typical respondents who met the inclusion criteria of this study then some questions were revised and modified to ensure that all participants share almost a common understanding of the questions.
- The sample size for the quantitative phase (100 participants) would be considered medium; a bigger group of participants might produces results that could be more generalized to other groups. Future research would further investigate the findings in this chapter and perhaps study feedback acquisition for more specific groups of users and feedback.

4.8 RELATED STUDIES

Hennig-Thurau et al (Hennig-Thurau et al. 2004) introduced several motives for users' engagement in an electronic word of mouth communication. Although the findings of users' motivations to provide feedback are highly similar to (Hennig-Thurau et al. 2004), the starting point was different in that the main focus was on users' motivations for giving feedback in the context of Socially-adaptive systems where users are targeted with feedback requests from software applications to assess the quality of the software behaviour. In other words, the study



Adaptation drivers of Feedback Acquisition
Users' Experience Factors
_Degree of simplicity of feedback requests
_Timing for feedback
_Users' awareness of the usage of their feedback
_Users' privacy preferences
_Users' familiarity with the software
Users' Social Motivations
_Visibility and similarity
_Social recognition
_Volume of given feedback
_Feedback acquisition as a social activity
Interaction style and Interface Design
_Language used
_Graphical design
_Fitness to the context of use.
_Information provided (e.g. statistics about feedback)
Users' Clusters
_Feedback antagonists
_Passive and stingy people
_Privacy fanatic but generous people
_Privacy tolerant and socially ostentatious people
Adaptive Feedback Acquisition
Why Feedback-based Adaptation [3]
Benefit Users:
_Increase users' trust and confidence in the software
_Increase users' satisfaction
_Increase users' willingness
_Increase users' involvement
Benefit Software Developers
_Knowledge-bases of users and their contextual profiles.
_Identifying bugs and software problems
_Better prioritising of requirements
_Identify the distribution of software usage across age groups, geo-location, time of day etc.
Improve Software Adaptation
Increase feedback quality
Economize the amount of collected feedback

FIGURE 11 CONCEPTUAL FRAMEWORK FOR AN ADAPTIVE ACQUISITION OF USERS' FEEDBACK.

focuses on users' reaction, perceptions and motivations to give explicit feedback in response to feedback requests to evaluate the software application's quality and validity in meeting user requirements.

Additionally, Pagano and Bruegge (Pagano and Bruegge 2013) conducted an empirical case study on five professional software development companies to explore the current practice of users' involvement via their feedback. Their study mainly focused on the stages after feedback has been collected (e.g. structure, analyse, and track users' feedback) and not much attention was paid to the earlier stage where feedback collection activity takes place. Additionally, the study was built on the fact that users' behaviour with regards to feedback is an important factor to be studied to allow software developers to understand and know their audiences. Hence tailoring the right acquisition method to each different type of users in a systematic manner will have positive implications on the quality of the software, users' feedback and satisfaction.

Furthermore, Pagano and Maalej (2013) conducted an exploratory study that analysed over one million reviews from the Apple AppStore. One of their study's objectives was to investigate the impact of users' already given feedback on the user community. Their findings suggested that visibility of already given feedback has a noticeable negative/positive impact on the app ratings as well as the community (e.g. users' experience). This thesis argues that the visibility of others feedback has a larger range of effect and can also negatively/positively affect users' willingness to participate and respond to feedbacks requests (see users' behaviour in Cluster2 and Cluster4).

In relation to feedback acquisition and as explained in *section 2.2.3* of this report, there are several available tools for gathering users' feedback in software systems in different forms (e.g. text, images and videos) and many of them also include context capturing functionality. Common examples are; UserVoice[9], Get-Satisfaction[10], IdeaStorm[11], VoiceYourView[12], EUREKA (Maalej and Happel 2009)and iRequire (Seyff et al. 2010). However, all of these tools are limited in terms of adaptivity to various users' behaviour. This can highly harm the quality of collected feedback and users' experience thus software's success. In addition, a systematic practice to gather users feedback is still missing in these tools (e.g. when to proactively ask users' for feedback?). This indeed highlights the need for a systematic and adaptive way to gather users' feedback.

4.9 SUMMARY

This chapter reported on an empirical mixed method study to explore and investigate users' behaviour with regard to feedback acquisition in software applications. Users' were studied first qualitatively and then quantitatively to enhance the results and allow for more generalization.

[9] https://www.uservoice.com
[10] https://getsatisfaction.com
[11] www.ideastorm.com
[12] http://www.voiceyourview.com.

The study found that users' behaviour with regard to feedback acquisition highly varies and is influenced by a number of behavioural factors. The results suggest that systematic approaches and mechanisms to conduct an adaptive feedback acquisition are highly needed. These approaches and mechanisms should fit and adapt to each different user type and should highly consider the factors that influence users' behaviour during the feedback acquisition process. Availability of such systematic approaches for an adaptive feedback acquisition can greatly improve the quality of users' feedback, users' satisfaction and the quality of socially-adaptive software. In the next chapter a follow up with the results of this chapter is qualitatively done to investigate the role of culture on peoples' preferences and behaviour to feedback acquisition. In general, the next chapters of this thesis further investigate and study the effect of the social factors (e.g. visibility of others feedback, social recognition, etc) on the design of an adaptive feedback acquisition.

5. CHAPTER 5: CULTURE AND USERS' BEHAVIOUR TO FEEDBACK ACQUISITION

In order to meet the third objective of this thesis (which is discovering whether there is a socio-cultural effect on the users' acceptance and behaviours with regard to feedback acquisition), this chapter present a follow up qualitative investigation to the previous chapter. It adopts focus group method to investigate the role of culture on users' behaviour to feedback acquisition. It gives an introduction to the role of culture on users' behaviour to software systems in general and then it reports on the study design, analysis, findings, discussion and conclusion.

5.1 INTRODUCTION

Culture as a term is difficult to define due to the fact it has multiple and sometimes conflicting definitions as described by cultural anthropologists. Therefore, the term cannot be pinned down to a specific or precise definition (Kroeber and Kluckhohn 1952). However, after a deep analysis of the available definitions of culture and their classification into different categories, Kroeber and Kluckhohn (1952) concluded that "culture consists of patterns, explicit and implicit, of and for behaviour acquired and transmitted by symbols, constituting the distinctive achievements of human groups, including their embodiment in artefacts; the essential core of culture consists of traditional (i.e., historically derived and selected) ideas and especially their attached values; culture systems may, on the one hand, be considered as products of action, on the other, as conditional elements of future action".

Culture highly varies across nations and societies. The essence of these variations is far into the past and could be considered stable in the long term (Reinecke and Bernstein 2011). Therefore, it can be assumed that they will not change in the future (Hofstede 1991). From a pragmatic point of view there are some de-facto considerations of different characteristics which describe certain cultures. These characteristics are mainly related to people's geographical and demographic attributes which are, in turn, linked to certain norms and value systems (Rogers 2008). For example, Hofstede believes that the best way to study and represent cultures is people's national groups, Hofstede indicated that the national groups "are usually the only kind of units available for comparison and [thus are] better than nothing" (Hofstede 2002).

Two groups that are considered significantly different from that perspective (different national groups) are the European/western and the Middle Eastern people. While Western/European people enjoy more personal freedoms and live with more spontaneity, Asian/Middle Eastern people tend to have low individualism, make more references to others, emphasize group goals and follow the overall expectations of the group (Triandis 1993; Nelson and Fivush 2004; Basu-Zharku 2011; Hofstede 2011).

TABLE 14 MOTIVATION TO GIVE FEEDBACK VS. USERS' COUNTRIES

	MIDDLE EASTERN					WESTERN			
	KSA	Iran	Egypt	Σ		UK	NL	Spain	Σ
1- Does the visibility of other people's feedback affect your willingness to give feedback?	70%	40%	50%	63%		33%	60%	50%	41%
2- Does the similarity of your opinion to other people's opinions, shown via their feedback, affect your willingness to give feedback?	70%	60%	75%	69%		33%	20%	33%	38%
3- Does the number / volume of feedback already provided by other people on a subject affect your willingness to give feedback on it?	75%	60%	50%	73%		42%	60%	33%	50%
4- Does a social recognition / visibility of you as a feedback provider affect your willingness to give feedback?	90%	60%	50%	84%		45%	40%	50%	38%
5- Would conducting the acquisition of feedback as a social activity be interesting to you?	80%	60%	25%	69%		3%	20%	33%	10%

In the context of software system, the software ability to adapt to the different cultures in the users' space is not only important for achieving usability in wider context but also a key requirement for professional and ethical reasons (Yunker 2002). At present, despite the availability of internationalisation, most software designs typically follow western cultural cues. This has resulted in a design gap when users from different cultures (i.e. eastern cultures) use the software within their cultural frame. An example can be easily found in the different ways people from all over the world use social networks. This is probably because software industry is largely led by western management and developers (Reinecke and Bernstein 2007).Therefore, systems that are marketed worldwide need to be localized to fit the different cultures (Nielsen 1996) as designs which are successful in one culture may fail dramatically in others (Honold 2000; Abdelnour-Nocera et al. 2013).

In the light of the result of the previously conducted empirical studies (see Chapter 4, Section 4.6), there is an indication that the variant cultural backgrounds of users has a noticeable impact on users' behaviour and how they are socially motivated to give feedback. Table 14 presents the extracted preliminary results with two cultural groups: Middle Eastern users and European users. The percentages provided in the table represent the positive answers the given questions. '*Σ*' is the acronym for the overall percentage. While a number of researchers have already investigated cultural differences in relation to related to software such as how graphics, language, object formatting, colours, and layout of web sites and other user preferences is perceived in different cultures (Marcus 2003; Callahan 2005; Aykin et al. 2006; Frandsen-Thorlacius et al. 2009), to the author's best knowledge no studies have yet investigated how users with different backgrounds behave in response to feedback acquisition and how their culture frame affect their motivations to feedback requests.

This chapter qualitatively focuses (using focus group) on the effect of culture on feedback acquisition following up the quantitative results presented in Chapter 4. It further investigates the effect and impact of cultural backgrounds on users' social motives to give feedback. The focus is on two different cultural backgrounds: Middle Eastern users and European users. The results of this study are meant to further help achieve the goal of this thesis in which we will be able to devise systematic method for conducting a socially aware feedback acquisition that can adapt to different types of users in terms of their cultural backgrounds (Middle Eastern and European users) to maximize feedback quality, users' satisfactions and motivations to give feedback. This chapter reports on the research method, study design, the result and then draws a conclusion.

5.2RESEARCH METHODOLOGY

In empirical research, researchers might need to qualitatively follow up or build upon quantitative results for the purpose of explaining or further investigating the quantitative results (Creswell et al. 2003). This design is suitable to researchers who need qualitative data to explain significant, non-significant or surprising quantitative results (Morse 1991). This design is also helpful in case researchers want to compose groups based on the quantitative results and follow up with these groups through qualitative research (Tashakkori and Teddlie 1998). In this thesis, these groups are the Middle Eastern and European users identified by the quantitative results.

The previous quantitative results of this research (see Chapter 4, Section 4.6) indicated that the variant cultural backgrounds of users have a noticeable impact on users' behaviour and how they are socially motivated to give feedback. That was only an indicator which would need confirmation and clarification. To achieve that, this study follows up the quantitative results with a qualitative phase through focus groups which is a powerful tool to get insights and stimulate discussions in a small group of participants.

A focus group is a qualitative research method in which a group of people are gathered to be asked about their opinions, beliefs, or attitudes regarding an issue, phenomena, service, etc. The questions should be asked in an interactive setting which allows participants to talk freely about their thoughts to other group's members. A focus group is shown to be useful to study people behaviour and attitude and it is often supported by the brainstorming technique when conducting the interactive session activity to allow for more information and thoughts to be discovered. Given the previous reasons, the nature of the study where social/group interaction is needed and the type of the asked questions which mainly relate to the culture impact (Middle Eastern and European cultures) on social perception of feedback acquisition of both the feedback provider and those who watch it, this study adopted Focus Group as a data collection method in this investigatory phase.

5.2.1 FOCUS GROUP DESIGN

Four Semi-structured focus groups were conducted in two countries (Saudi Arabia and the UK) with 27 participants to further explore how Middle Eastern and European users behave in response to feedback acquisition and how their culture frame affect their motivations to feedback requests. Participants were carefully selected in order to guarantee a high level of diversity and to avoid bias (e.g. various age groups, backgrounds and gender). The first two focus groups took place on June-2014 (at Bournemouth University, UK) and were conducted with European participants to investigate how their culture impacts their perception to feedback acquisition. Whereas the other two focus groups were held on July-2014 and conducted with Saudi participants to investigate the Middle Eastern culture's impact to feedback acquisition. Each focus group session lasted for about an hour which makes an amount of four hours in total.

The focus groups protocol was developed in the light of the previous quantitative results of this research in which an indication to cultural differences between European and Middle Eastern users on what motivate them to provide feedback was discovered. In particular, the four social factors (*Feedback acquisition as a social activity, Social recognition, Volume of already given feedback and Visibility and similarity of others feedback*) that influence how Middle Eastern and European users are socially motivated to give feedback served as a foundation to develop the protocol of the focus group (see Appendix 4 for the focus group protocol) .

The protocol was iteratively reviewed and revised by 3 researchers to ensure clarity and understandability. Participants were briefed to the session and the discussed topic through a 10 minutes presentation in which some example of feedback acquisition in software application were also given to more familiarise the participants with the discussed topic. The project was approved by Bournemouth University Research Ethics Committee (UREC). Each participant received £15 Amazon vouchers as an appreciation for taking part in the study.

5.2.2 SAMPLING

Purposeful sampling is a common technique in qualitative research (Creswell 2013). In this study, purposeful sampling was used to recruit 27 participants to take part in four focus groups. The inclusion criteria of this study allowed for participants who are European or were born and raised in Europe to take part in the first two focus groups dedicated to study European people (7 participants for the first focus group and 6 for the second). On the other hand, participants who are Saudis or were born and raised in Saudi Arabia were recruited to take part in the other two focus groups dedicated to study Middle Eastern people in which 7 participants took part in each focus group.

In addition, the inclusion criteria allowed for participants within an age range of 18 to 71 and average computer users who use typical and diverse set of popular software applications rather than domain specific software for everyday life activities. This sampling criterion were developed to allow for more variety in selecting participants as well as reflecting users' experience with popularly used software applications which will maximize the generalizability of the results. See Table 15 for participants' characteristics.

The author assumed that Saudi users could be a good fit to represent the Middle Eastern culture. It is due to the fact that Saudi Arabia is one of the largest countries in the Middle East and it could fairly represent users' cultures in the region epically when it comes to the use of software. Recent statistics indicate that Saudi Arabia has over 6 million active Facebook users which is the highest Facebook user rate in the region. With more than 3 million active Twitter users, Saudi Arabia takes the lead not only in the Middle East, but better yet, it leads the world in its Twitter users' growth rate. In addition, more than 90,000,000 videos are watched daily on YouTube in Saudi Arabian which is more than any daily YouTube video views number in the world (Socialclinic 2013). According to the previous statistics and to the resources available to the author and time, the author assumes that Saudi Arabia is a reasonable fit to be adopted in this study to represent Middle Eastern culture to feedback acquisition.

TABLE 15 THE CHARACTERISTICS OF THE FOCUS GROUP PARTICIPANTS

	Participants' Characteristics			
	Participant	**Age**	**Gender**	**Home Country**
European Participants	**P1**	58	Female	Italian
	P2	45	Female	English
	P3	22	Male	Polish
	P4	71	Male	French
	P5	34	Female	Polish
	P6	43	Female	French
	P7	49	Female	Swiss
	P8	39	Male	Sweden
	P9	56	Male	Irish
	P10	35	Female	Romania
	P11	41	Male	UK
	P12	27	Female	Polish
	P13	19	Male	Sweden
	Total	**13 Participants**		
Middle Eastern Participants	**P1**	41	Female	KSA
	P2	45	Female	KSA
	P3	35	Female	KSA
	P4	18	Male	KSA

	P5	20	Male	KSA
	P6	27	Female	KSA
	P7	55	Male	KSA
	P8	30	Male	KSA
	P9	22	Male	KSA
	P10	18	Female	KSA
	P11	61	Male	KSA
	P12	28	Female	KSA
	P13	25	Male	KSA
	P14	19	Male	KSA
	Total	**14 Participants**		

5.3ANALYSIS

Focus groups were analysed following the same method adopted in the previous qualitative study of this research (see chapter 4, Section 4.3.3) in which they were audio taped and transcribed verbatim analysis was performed in several steps which included: (1) initial exploration of the gathered data by reading the transcripts; (2) coding data by labelling and segmenting the text; (3) using an inter-coder agreement check to verify codes (two researchers worked on verifying codes and a third researcher was approached for solving conflicts); (4) using codes to generate themes by gathering similar codes together; (5) connecting, comparing and interrelating themes. Credibility of the findings was maximized by using an inter-coder agreement check and academic advisor's auditing (Miles and Huberman 1994; Creswell 2013).

5.4FINDINGS

As previously stated the focus group design covered how the four social factors (*Feedback acquisition as a social activity, Social recognition, Volume of already given feedback and Visibility and similarity of others feedback*) influence Middle Eastern and European users with regard to feedback acquisition. In this section, these social factors are used to structure, represent and discuss the main themes of the findings. Table 16 shows the themes and codes that highlight the encountered behavioural differences between the two studied groups (Middle Eastern and European users) to feedback acquisition.

TABLE 16 A BREAKDOWN OF THE THEMES AND CODES OF THE ANALYSIS.

Theme1: Visibility and similarity of others feedback
Anonymity of feedback providers. Cross conversation Feedback objectivity and relevancy Language used among given feedback. Gender, social position or a personal relationship with a feedback provider

Theme2: Volume of already given feedback Feedback objectivity and relevancy.
Theme3: Social recognition Feedback objectivity and relevancy. Suitable and unsuitable uses of social recognition More beneficial with close friends and small community. Social recognition can result in ignoring unrecognized users' feedback Social recognition might result in addiction especially for young users.
Theme4: Feedback acquisition as a social activity Feedback objectivity and relevancy.

5.4.1 VISIBILITY AND SIMILARITY OF OTHERS FEEDBACK

Generally speaking, the responses from Middle Eastern participants and European participants are noticeably different when they were asked whether the visibility of others feedback (the ability to others feedback before giving feedback) and the similarity of their feedback to others feedback would have effect on their willingness to give feedback. Although feedback visibility plays a role in motivating both Middle Eastern and European participants to give feedback, Middle Eastern participants seemed to be more concerned and socially motivated by this factor than European participants. This trend became more obvious when they were asked whether knowing the similarity of their feedback to others would affect their willingness to give feedback. One the participants mentioned in this regard *"Of course. I'm normally interested in replying to reviews that I do not agree with"*. Figure 12 gives a general view of how the different cultures' impact on this social factor could affect the quality of given feedback[13].

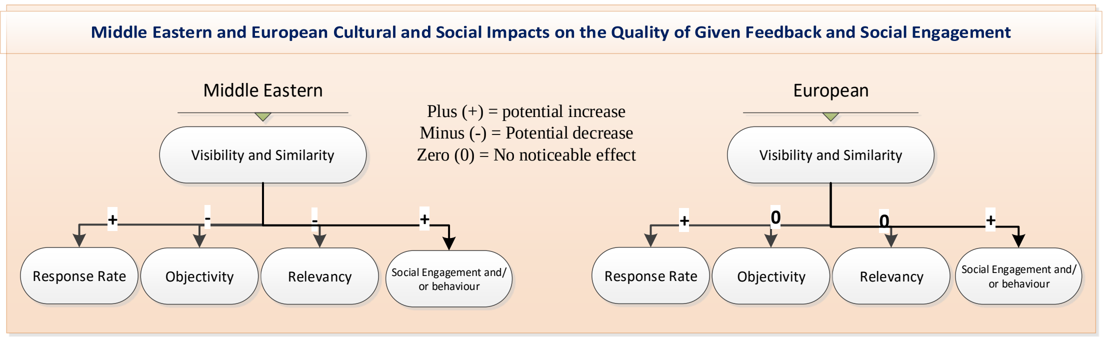

Middle Eastern and European Cultural and Social Impacts on the Quality of Given Feedback and Social Engagement
Middle Eastern
Visibility and Similarity
Plus (+) = potential increase
Minus (-) = Potential decrease
Zero (0) = No noticeable effect
European
Visibility and Similarity
+
-
-
+
Response Rate
Objectivity
Relevancy
Social Engagement and/ or behaviour
+
0
0
+
Response Rate
Objectivity
Relevancy
Social Engagement and/ or behaviour

FIGURE 12 THE EFFECT OF FEEDBACK VISIBILITY AND SIMILARITY IN MIDDLE EASTERN AND EUROPEAN CULTURE.

In addition, the following dimensions of this theme were extracted from the participants' responses:

5.4.1.1 ANONYMITY OF FEEDBACK PROVIDERS

Users ability to give feedback anonymously (i.e. using nicknames instead of the real names) is shown to play a role in motivating both parties (Middle Eastern and European users) to give feedback especially when the given feedback is publicly open and seen by other users. This is

[13] The figures shown in the finding section of this chapter are a reflection of the analysed qualitative input by the participants in a graphical way. For example, when participants say that anonymity of their names can cause the so-called online disinhibition effect, this could mean less objectivity (represented on the figure as MINUS) of the given feedback. This effect (Plus, Minus or Zero) can be easily inferred from the reported findings in each section.

perhaps due to several factors (i.e. dissociative anonymity, invisibility, solipsistic introjection, dissociative imagination, and minimization of authority) as discussed in (Suler 2004) which allow users to enjoy more freedom in expressing their opinion about a product or a software service. One of the participants said "*I would be more interested in truly engaging when no one knows me. It feels like you are free to say whatever you like and you will still be unknown*". However, anonymity does not seem to be a motivating factor when a feedback provider is part of an online closed community such as a Facebook group. In general, users prefer to know the identity of the feedback provider in their closed social network/community since it makes them more comfortable participating and discussing a software service or a product.

On the downside, anonymity seems to affect the objectivity and relevancy of given feedbacks by both parties but it is more prominent with the Middle Eastern than European users. Their overall responses indicated that anonymity could results in the so-called *online disinhibition effect* (Suler 2004) which gives users space to escape from their social constrains. However, this feeling of freedom can result in an overexpression and less objectivity and relevancy of their opinion about a service or a product. One of the participants commented *"The problem is that some people think they can say anything or be unfair or even harm others because nobody knows them. That's not the point of being anonymous"*. Figure 13 gives a clearer view of how the different cultures' impact on this social factor could affect the quality of given feedback.

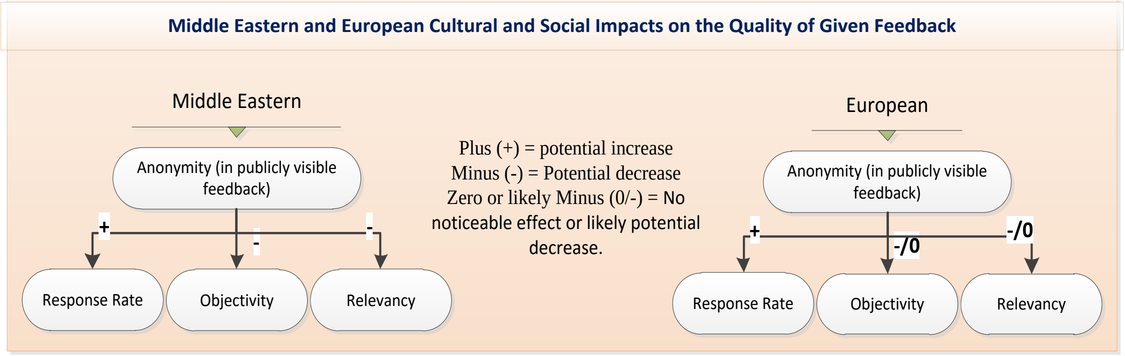

Middle Eastern and European Cultural and Social Impacts on the Quality of Given Feedback
Middle Eastern
Anonymity (in publicly visible feedback)
+
-
-
Response Rate
Objectivity
Relevancy
Plus (+) = potential increase
Minus (-) = Potential decrease
Zero or likely Minus (0/-) = No noticeable effect or likely potential decrease.
European
Anonymity (in publicly visible feedback)
+
-/0
-/0
Response Rate
Objectivity
Relevancy

FIGURE 13 THE EFFECT OF ANONYMITY ON FEEDBACK IN MIDDLE EASTERN AND EUROPEAN CULTURE.

5.4.1.2 *CROSS CONVERSATION, FEEDBACK OBJECTIVITY AND RELEVANCY AND LANGUAGE USED:*

In the context of cross conversation (irrelevant feedback) and how it affects the provided feedback, users' responses indicated that feedback given by European users might enjoy a slightly higher degree of relevancy and objectivity than the feedback given by Middle Eastern users. This is perhaps due to the fact that Middle Eastern users are more socially engaged which could put some constrains on the relevancy and objectivity of their given feedback However, in all cases cross conversation (irrelevant feedback), subjectivity and the harshness degree in the language used among already given feedback can results in a low response rate by both parties

(Middle Eastern and European users). It can also result in a harm to the software product or the provided service.

An example of this was given by one of the participants "if *the cross conversations or the harshness of the used language leverage among users feedback such as two large groups of users fighting around irrelevant specific religious or political party, this can highly result in users from either group degrading and disliking the service or the product just to cause a harm to the service provider who they think he might be from the opposite party".* Figure 14 gives a clearer view of how the different cultures' impact on this social factor could affect the quality of given feedback.

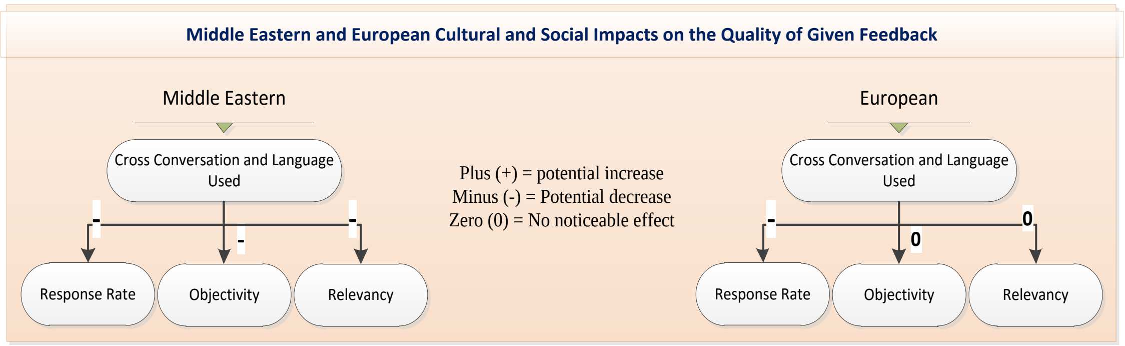

Middle Eastern and European Cultural and Social Impacts on the Quality of Given Feedback
Middle Eastern
Cross Conversation and Language Used
-
-
-
Response Rate
Objectivity
Relevancy
Plus (+) = potential increase
Minus (-) = Potential decrease
Zero (0) = No noticeable effect
European
Cross Conversation and Language Used
-
0
0
Response Rate
Objectivity
Relevancy

FIGURE 14 THE EFFECT OF CROSS CONVERSATION AND LANGUAGE USED ON FEEDBACK IN MIDDLE EASTERN AND EUROPEAN CULTURE.

5.4.1.3 *GENDER, SOCIAL POSITION OR A PERSONAL RELATIONSHIP WITH A FEEDBACK PROVIDER*

Being able to know/see the gender, social position or your personal relationship with a feedback provider gives a clear view about how users from the different two cultures (Middle Eastern and European cultures) are socially motivated to give feedback. In the European culture, users do not seem to be influenced by these factors to give feedback at all. This is perhaps due to the fact that they feel more socially- freewheeling than Middle Eastern users as described by some of the participants. A participant commented *"I believe living in this community (European community) makes you less socially-dependent and do not easily accept to accomplish things with the help of your parents or friends for example. You always want to do things by yourself. This is how we grew up and I believe this is the typical European life".*

On the other hand, users coming from Middle Eastern culture are significantly different to the European users in their perception of these factors. They feel highly motivated by these factors to give feedback especially with people/users they know (i.e. having a personal relationship with the person asking for feedback or the users giving feedback). Interestingly, the gender of a feedback provider is considered to be source of curiosity that motivates Middle Eastern users to give feedback which is not the case with the European users. Males/females would find it interesting to see how females/males think of a particular software service or product.

One of the participants said *“**[TRANSLATED FROM ARABIC]** in certain context, I would really love to see how males’ feedback would be on a certain aspect of a software such as the interface colours. You know we love girly colours and this will always make my feedback clashes with males which is fun”*. On the down side, motivating Middle Eastern users by these factors can result in a questionable quality of their given feedback in terms of its objectivity and relevancy to the discussed software service or product. Some of the Middle Eastern participants mentioned that they always tend to be supportive and on the side of their friends (i.e. software service provider who is asking for their feedback) regardless of the discussed product and their real opinion about it. Although this can be harmful to the quality of the feedback, it gives a nice example of a caring relationship among users in which friends will always be protective and caring of the reputation and social status of each other. Figure 15 gives a clearer view of how the different cultures’ impact on this social factor could affect the quality of given feedback.

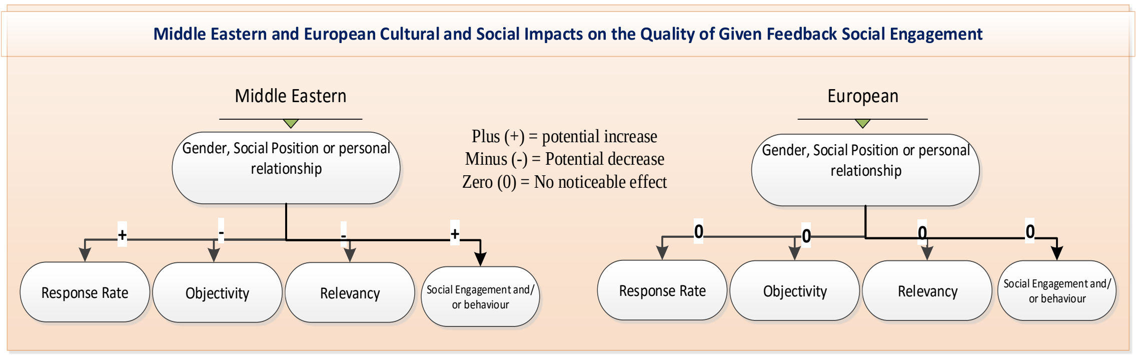

Middle Eastern and European Cultural and Social Impacts on the Quality of Given Feedback Social Engagement
Middle Eastern
Gender, Social Position or personal relationship
+
-
-
+
Response Rate
Objectivity
Relevancy
Social Engagement and/ or behaviour
Plus (+) = potential increase
Minus (-) = Potential decrease
Zero (0) = No noticeable effect
European
Gender, Social Position or personal relationship
0
0
0
0
Response Rate
Objectivity
Relevancy
Social Engagement and/ or behaviour

FIGURE 15 THE EFFECT OF GENDER, POSITION OR RELATIONSHIP ON FEEDBACK IN MIDDLE EASTERN AND EUROPEAN CULTURE.

5.4.2 VOLUME OF ALREADY GIVEN FEEDBACK

When participants were asked whether the number of feedback already provided on a software service or a product would affect their willingness to give feedback, Middle Eastern users showed a consensus on that they would like to provide feedback if there were only few feedback given rather than a large number of already given feedback. One of the participants commented *“**[TRANSLATED FROM ARABIC]** it makes me feel sorry when only few reviews are given especially when the app or service is good. I would certainly find a time to write my own review and help”*. In comparison, European users had a similar attitude but they perceive this as a less important factor when compared to Middle Eastern users.

5.4.2.1 FEEDBACK OBJECTIVITY AND RELEVANCY

The degree of feedback relevancy and objectivity given by both parties (Middle Eastern and European users) does not seem to be influenced by the volume of already given feedback. However, the low number of already given feedback can sometimes impose potential risk to the software service or product. Several users from both parties indicated that when a low number of feedback/reviews is already given it makes them lose interest in the provided service since it does not seem to be popular among users otherwise it would be highly reviewed by a large number of users. One of the participants commented *"The first thing I do before downloading an app is to look at the number of reviews. A low number means to me less popular and useful"*. Figure 16 gives a clearer view of how the different cultures' impact on this social factor could affect the quality of given feedback.

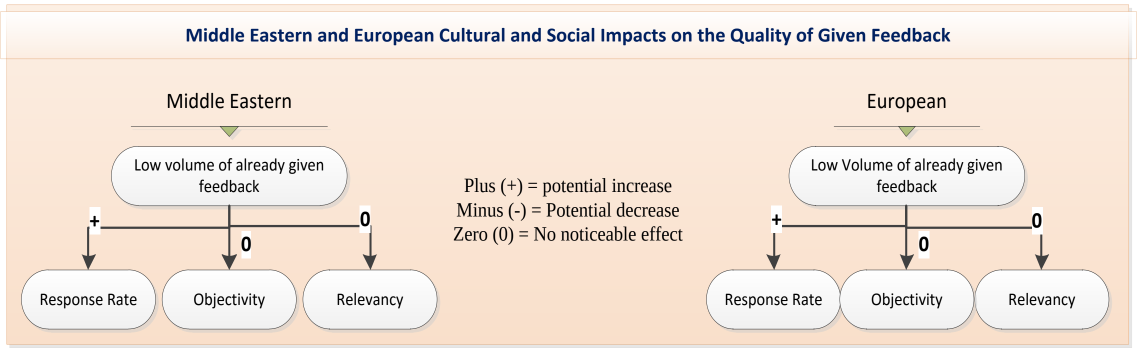

Middle Eastern and European Cultural and Social Impacts on the Quality of Given Feedback
Middle Eastern
Low volume of already given feedback
+
0
0
Response Rate
Objectivity
Relevancy
Plus (+) = potential increase
Minus (-) = Potential decrease
Zero (0) = No noticeable effect
European
Low Volume of already given feedback
+
0
0
Response Rate
Objectivity
Relevancy

FIGURE 16 THE EFFECT OF FEEDBACK VOLUME ON FEEDBACK IN MIDDLE EASTERN AND EUROPEAN CULTURE.

5.4.3 SOCIAL RECOGNITION

Participants were asked whether being recognized by the community as feedback providers would affect their willingness to give feedback. The responses from the two groups were noticeably different. Users from Middle Eastern backgrounds indicated that being socially recognized as a feedback provider is an influential factor that could positively maximize their willingness to give feedback. There could be still some constraints on this, e.g., some participants commented that *"it is nice to be visible only when others can see their feedback which led to some changes on the system"*.

On the other hand, European users seemed to be far less motivated by the same aspect as few participants would have thought in a similar way. In fact, some of the European participants mentioned that some of the socially recognized feedback providers might have been selected and sponsored by the software/service provider for marketing reasons (e.g. being a celebrity). One of the participants commented *"You know nowadays in advertisements they use popular names and faces to attract people. It can be the same case for feedback too"*. Although Middle Eastern users are also aware of this threat but they would still be more interested in the social aspect (social recognition) regardless of the potential threats it might cause. Generally, these potential threats could result in users having negative attitudes towards the provided software service or the product. Figure 17 gives a general view of how the different cultures' impact on this social factor could affect the quality of given feedback.

5.4.3.1 FEEDBACK OBJECTIVITY AND RELEVANCY

Social recognition as a motivating factor to feedback acquisition does not seem to play a role in affecting the objectivity and relevancy of feedback given by European users. In contrast to, the degree of relevancy and objectivity of feedback given by Middle Eastern users to the provided service might be slightly harmed. This is due to the fact that Middle Eastern users are more socially involved with their community and this can push them sometimes to act differently when they are being socially recognized as feedback providers. One of the participants commented *"You know lots of people could be watching me. I will always try my best to ideal in their eyes"*.

This indeed could have negative impact on their given feedback such as imposing favouritism in their opinions regarding the provided software service which made them socially recognized at the first place. However, this social aspect could have a positive impact on the social behaviour within the online society of Middle Eastern users since socially recognized users would feel more socially constrained and their behaviour is always under the spot light of the community. This could result in an overall improved social behaviour and an online user community.

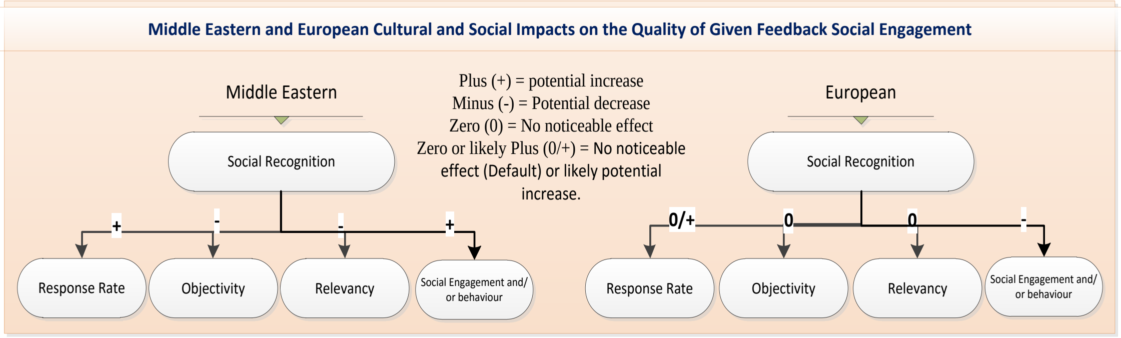

Middle Eastern and European Cultural and Social Impacts on the Quality of Given Feedback Social Engagement
Middle Eastern
Plus (+) = potential increase
Minus (-) = Potential decrease
Zero (0) = No noticeable effect
Zero or likely Plus (0/+) = No noticeable effect (Default) or likely potential increase.
European
Social Recognition
+
-
-
+
Response Rate
Objectivity
Relevancy
Social Engagement and/ or behaviour
Social Recognition
0/+
0
0
-
Response Rate
Objectivity
Relevancy
Social Engagement and/ or behaviour

FIGURE 17 THE EFFECT OF SOCIAL RECOGNITION ON FEEDBACK IN MIDDLE EASTERN AND EUROPEAN CULTURE.

5.4.3.2 *SUITABLE AND UNSUITABLE USES OF SOCIAL RECOGNITION*

As indicated by users' responses from both parties (Middle Eastern and European users) social recognition could be more beneficial and motivating if it is used in a small and closed community of users where all of the users somehow know each other (i.e. Facebook groups). This makes users more interested in being socially recognized to people who they know and more motivating for users to follow up a socially recognized person who they know and trust. One of the participants commented *"It would be nice if we are all friends and know each other. I will trust and follow who I know and I will be happy to be recognized in front of people who care about me, right?"*

On the other hand, users indicated that social recognition can lead to users following socially recognized users only and ignoring the feedback of unrecognized users even if their feedback is far better in quality. This of course could harm the provided software service and its reputation since vital information and knowledge about the provided service could be overlooked. In addition, some users mentioned that social recognition could lead to users' addiction to the used software especially young users who are eager to be socially recognized and more socially active. It is a trade-off between using this factor to benefit users or harm them and the provided service.

5.4.4 FEEDBACK ACQUISITION AS A SOCIAL ACTIVITY

Similar to the above dimension, Middle Eastern users showed a much higher interest in conducting feedback acquisition as a social activity (i.e. social games) and emphasized that it would increase their willingness to give feedback. This was true especially for young users. Example of such an activity could be the users' ability to visualize how their direct and indirect social contacts are rating a certain service and how their feedback influenced the trend in their community. Compared to this, the majority of interviewed European showed a negative trend towards this factor. In fact, they believe feedback requests should be straightforward and simple by default and conducting it as a social activity could make the process of feedback acquisition more discouraging, complex and distracting from the main purpose which is evaluating a provided software service. One of the participants said in this regard *"I think it should not look more than what it is supposed to do. It is to get your feedback and not a game to play with"*

5.4.4.1 FEEDBACK OBJECTIVITY AND RELEVANCY

Conducting the feedback acquisition as a social activity could have a harmful effect on the objectivity and relevancy of the given feedback by Middle Eastern users. It is perhaps due to users' engagement with the activity more than the provided service as well as the burden of social constraints. One of the participants commented *"I really do not know what to say but I imagine I would be nicer about my opinion to people I know more than others".* Although this might result in given feedback that does not reflect users' true opinion about the provided software service, it could also result in a more socially active users which could be a positive sign to the software and the users' community. Users' satisfaction with the social activity should not affect their view about the software service. This highlights the need to carefully design and apply this factor in the context of feedback acquisition in software applications. Figure 18 gives a clearer view of how the different cultures' impact on this social factor could affect the quality of given feedback.

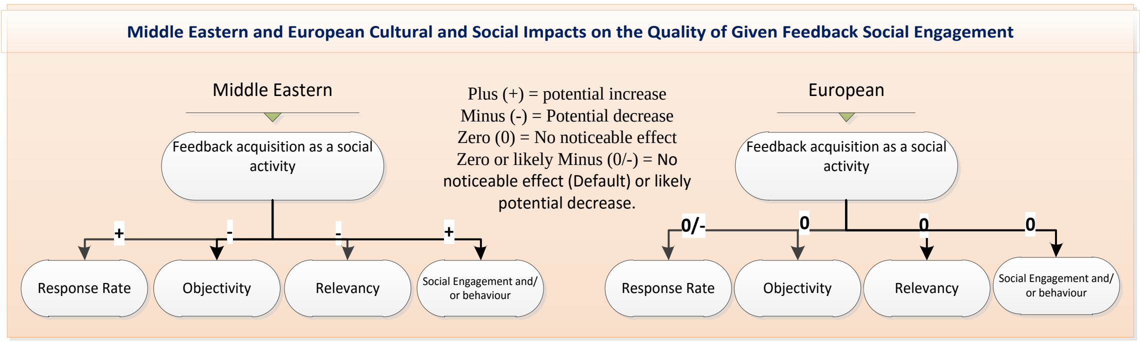

Middle Eastern and European Cultural and Social Impacts on the Quality of Given Feedback Social Engagement
Middle Eastern
Feedback acquisition as a social activity
Plus (+) = potential increase
Minus (-) = Potential decrease
Zero (0) = No noticeable effect
Zero or likely Minus (0/-) = No noticeable effect (Default) or likely potential decrease.
European
Feedback acquisition as a social activity
+
-
-
+
0/-
0
0
0
Response Rate
Objectivity
Relevancy
Social Engagement and/ or behaviour
Response Rate
Objectivity
Relevancy
Social Engagement and/ or behaviour

FIGURE 18 THE EFFECT OF FEEDBACK ACQUISITION AS A SOCIAL ACTIVITY ON FEEDBACK IN MIDDLE EASTERN AND EUROPEAN CULTURE.

5.5 SUMMARY AND RELATED WORK

Looking back at the extracted findings, one can observe that the so called cultural dimensions collectivism/individualism are highly reflected in European/Western and Middle Eastern/Asian users' behaviours to feedback acquisition and how the culture plays a notable role in socially motivation them to give feedback. Individualism is mostly seen in western/European cultures, whereas collectivism is mostly seen in Asian/Middle Eastern and, African cultures (Triandis 1993; Nelson and Fivush 2004).

In western cultures (individualistic societies) people value self-reliance, independence, autonomy and personal achievement (Skillman 1999). They enjoy more personal freedoms and live with more spontaneity than people in Eastern cultures. In the context of feedback acquisition, individualism is indeed reflected in the results where the European users found to be more socially-independent and showed less interest in being socially motivated to give feedback than Middle Eastern users.

In eastern cultures (Collectivistic societies) people always value cooperation, solidarity, and harmony (Skillman 1999), and tend to make more references to others, emphasize group goals, and follow the overall expectations of the group (Basu-Zharku 2011). In the context of feedback acquisition, collectivism is highly reflected in the results where the Middle Eastern users found to be more socially collaborative and showed a noticeable interest in being socially motivated to give feedback than European users. This highlights the need for a culture-aware feedback acquisition that can understand these variations in culture and adapt accordingly to enhance users' experience as well as software quality.

In relation to this work, a number of HCI studies such as (Barber and Badre 1998), (Pfeil et al. 2006), (Sheppard and Scholtz 1999), (Singh et al. 2005), (Schmid-Isler 2000), (Marcus and Gould 2000), (Chau et al. 2002), (Robbins and Stylianou 2002), (Frandsen-Thorlacius et al. 2009), (Tsikriktsis 2002) and (Cakir et al. 2005) have been conducted to investigate the cultural diversity in the World Wide Web and to explore the impact that cultural differences have on website design, usability, e-commerce, and computer-mediated communication.

Generally, these studies concluded that culture is a key factor that impacts users' perceptions in regards to websites design, usability, e-commerce, and computer-mediated communication. They advocate the high need to cater for cultural differences in regards to the World Wide Web which will result in a positive impact on users' experience and satisfaction.

In the context of feedback acquisition in software applications, this thesis advocated that cultural variations have a noticeable impact on how Middle Eastern and European users' behaviour in response to feedback acquisition and how they are socially motivated to give feedback. The results suggest that having a method for the design of an adaptive and cultural-aware feedback acquisition could highly affect the quality of the collected feedback as well as users' satisfactions and response rate to feedback acquisition. The next chapter employs the concept of persona to represent users' different behaviours investigated in this chapter and the previous one and help guiding the design phase of feedback acquisition.

6. CHAPTER 6: PERSONA-BASED ADAPTIVE FEEDBACK ACQUISITION

As discussed in the previous chapters, users' behaviours and preferences to feedback acquisition highly differ. Integrating and accommodating these differences between users in terms of how feedback should be requested is a complex task and requires a careful engineering process. In this chapter, this problem is tackled by employing the concept of Persona to aid software engineers understand the various users' behaviours and improve their ability to design feedback acquisition techniques more efficiently. It proposes a set of building constructs for personas with regard to feedback acquisition and also introduces a set of personas based on a mixture of qualitative and quantitative studies (see Chapters 4, 5).

6.1INTRODUCTION

Users' feedback (as previously discussed throughout this thesis) is a main source of knowledge to guide software autonomous and semi-autonomous adaptation, maintenance, and evolution decisions. The reliance on users' feedback and their collective judgement to shape such decisions is called Social Sensing (Ali et al. 2011a) and Social Adaptation (Ali et al. 2012). In practice, software developers/companies are interested in users' feedback for two main reasons. The first relates to classic business and marketing purposes. This views users as clients and developers/companies should be always responsive to their feedback and emerging needs. Also, users can sometimes harm the online reputation of the software and, thus, the company if they leave constantly negative feedback. Consequently, software developers/companies keep seeking users' feedback to assess the acceptance of their software. The second reason relates to the need for real-time feedback from users' about the environments, features being used, errors occurring and in which context. This feedback is meant for a more detailed view on the use of software and help directly in its adaptation and evolution (Ali et al. 2011a; Pagano and Bruegge 2013).

In software systems users can be involved in different ways. They can be actively working on a specific software engineering activity. For example, in development styles such as users-centred design, users can suggest modifications and enhancements and perform tests at the development stage.

Alternatively, users can influence the engineering decision about software without being directly part of the engineering process. For example, users might provide feedback, rate specific decision, or influence the opinion of others whether to use certain software (Maalej and Pagano 2011). From a Requirement Engineering perspective, users' involvement via their feedback while they are using the software in practice is more credible to assess how the software is playing its role in meeting their requirements in practice (Ali et al. 2011a; Ali et al. 2012; Pagano and Br 2013). In fact, many of the users' requirements are only identified after the software is being deployed and once users get the chance to use it in a real context (Ko et al.

2011). This becomes even more evident when considering the requirements which emerge because of the existence of competitive technology and peer pressure.

Since giving feedback is generally a voluntary activity, the design of feedback acquisition should focus on the volunteers; the users. However, the previous studies in (see Chapters 3, 4, 5) showed that users' perception and behaviour with regard to feedback acquisition significantly vary and are affected by a number of factors such as interface design, volume and frequency of feedback requests, the language used, etc. This highlights the need for an adaptive feedback acquisition that can cater for such diversity to make users more motivated to respond to feedback requests. This will also have a positive side-effect on the feedback quality, users' engagement with the software, users' satisfaction and trust in the system. Poorly-designed feedback acquisition can harm the collected feedback quality, users' experience and software's success.

To follow up with the results of the previous studies of this thesis (presented in Chapters 4, 5), this chapter presents a possible solution to the challenge of integrating users' different behaviours and perceptions in the design of an adaptive feedback acquisition by employing the concept of Persona (Cooper 1999a). Personas are meant to increase software engineers' understanding of various users' behaviours and improve their ability to design feedback acquisition more efficiently. A set of behavioural personas were developed based on a mixed methods approach (sequential-exploratory approach) and in the next chapter a persona-based method is proposed for the design of adaptive feedback requests.

6.2 PERSONA AND FEEDBACK ACQUISITION

Given the high diversity in users' behaviour and perception to feedback acquisition, which was demonstrated in the previous chapters of this thesis, the challenge is how to represent this diversity in an actionable and meaningful way that can inform the design of an adaptive feedback acquisition. To tackle this challenge, the Persona concept is adopted. Persona, as a concept, has its root in marketing then Cooper (1999a) proposed the use of Persona as an interactive design tool to model user experience in software development (Moore 2002; Pruitt and Grudin 2003).

Cooper (1999) advocates the need to redirect the focus of the development process towards end users and their requirements and proposes personas as fictional characters that represent different types of users and their behaviours based on data gathered from ethnographic and empirical analysis of actual users. In (Idoughi et al. 2012), personas are defined as "a descriptive model of the user, encompassing information such as user characteristics, goals and needs"

Overall, personas as a user experience design tool gained popularity in both academic and practitioners' communities in the field of software development. Personas as fictional characters are given names, age, gender, photos, occupations, etc. This could reflect important characteristics of the persona or, sometimes, just to bring life to them and make them more engaging at the design phase. Personas have shown to be a powerful tool to represent the aspects of discovered user categories and draw discussions about these categories which can help in the design process of software systems in general (Weber 2010a) and the adaptive feedback acquisition in particular.

6.3 BENEFITS OF PERSONAS

Personas are not just a design tool but it is also meant to enhance engagement, communication and reality at the design phase of software systems (Nielsen 2002; Pruitt and Grudin 2003; Canossa and Drachen 2009; Miaskiewicz and Kozar 2011). As discussed in (Nielsen 2002; Pruitt and Grudin 2003; Canossa and Drachen 2009; Miaskiewicz and Kozar 2011), the main benefits of personas as user experience design tool are as follows:

- Personas make the design process easier in which engineers relate to human face and name instead of abstract user/customer data.
- Personas supply a shared, fast and effective form of communication among software engineers and designers.
- Personas describe user needs and wants which limit stakeholders' ability to shape users to their convenience.
- Personas minimize self-referential designs in which designers unconsciously predict their own mental models. This helps individuals realizing how the users/customers are different from themselves.

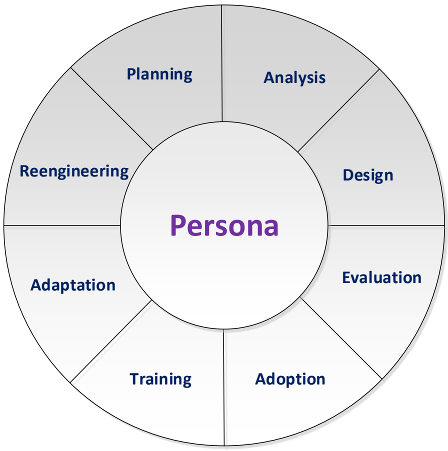


FIG.19. THE ROLE OF PERSONAS IN THE DESIGN PHASE. ADAPTED FROM (SEFFAH ET AL. 2009)

- Personas also help engineers keep the focus on the limited subset of users (persona) at a time which can result in more robust design decisions.
- Personas are useful for software/product validation purposes in which proposed designs, features and solutions can be reviewed and evaluated against the needs described by an individual persona.

- The information personas contain can be an inspiration source for the design team throughout the design phase (see Figure 19) (Norman and Draper 1986).

In the light of the previously mentioned benefits, persona is adopted as a design tool to direct the design process of an adaptive feedback acquisition. It allows software engineers to better understand the diversity of users' behaviours and their needs with respect to feedback acquisition in socially adaptive software towards an effective, fast and shared way of communication. This understanding will positively impact the design of the adaptive acquisition of users' feedback. In addition, Personas are good starting point to initiate detailed discussion about the different types of users (personas) which could highlight new design opportunities for the feedback acquisition activity as explained in Section 6.

6.4PERSONAS CREATION

Creation of Personas is still a challenging task and there is not one right way or method to create personas (Mulder and Yaar 2006). Ultimately, what the researcher needs to do is to aggregate the qualitative or quantitative data they got about the users into an actionable and meaningful story that can impact the design of a certain product (Spool 2007). Personas are affected by several factors that play a role on how they should be created. Generally, it depends on the following factors (Mulder and Yaar 2006):

- The targeted audience for the personas and their needs in order to agree to use personas, i.e. the type of information the persona should deliver to software engineers.
- How the personas will be used and for what types of decisions, i.e. is the persona only for initiating discussion or driving the design of a certain product/software?
- The time, money and resources available for the researcher to invest in the creation of Personas.
- The type of research undertaken, i.e. qualitative research, qualitative and quantitative (mixed method) research or quantitative research.

The approach to the creation of personas largely depends on the type of research conducted. Since the previous published users' studies of this thesis with regard to feedback acquisition followed a mixed method approach, this thesis followed the guidelines for personas creation proposed in (Mulder and Yaar 2006) and designed the personas according to steps shown in Figure 20. In the previous work (see Chapters 4, 5), an empirical study was conducted to understand users' different perspectives and behavioural aspects to feedback acquisition in software applications. A mixed method (sequential-exploratory approach), consisting of qualitative (interviews) and quantitative (questionnaires) approaches, was followed. Since (Mulder and Yaar 2006) guidelines are generic, this thesis adapts and specifies the approach to the context of feedback acquisition through the following steps:

6.4.1 CONDUCTING A QUALITATIVE RESEARCH

The qualitative phase is useful to reveal insights and initial understanding into user behaviours and attitudes. The qualitative phase allowed the author to explore and gain insights of users' behaviour in relation to providing feedback in software applications.

6.4.2 FORM HYPOTHESES, FOUNDATIONS AND IDEAS FOR FURTHER INVESTIGATION

The qualitative phase is useful to help the researcher producing initial and relevant hypotheses, foundations or ideas about users' behaviours and attitude in relation to a certain software/product. These ideas can be then further investigated quantitatively on a larger group of users. The qualitative phase was helpful to figure out the relevant factors of users' behaviours to feedback acquisition. This has been done in Chapters 4 and 5.

6.4.3 INVESTIGATE THE FORMED HYPOTHESES, FOUNDATIONS AND IDEAS QUANTITATIVELY

The hypotheses, foundations and ideas resulted from the previous steps (qualitative phase) are used to help designing a follow up quantitative approach. In this step the quantitative approach is used to assess the interpretation of the qualitative findings and maximize results generalizability. This also impacts personas validity and credibility since they are based on actual data of a larger group of users. This has been done in the quantitative phase of the study conducted in Chapter 4.

6.4.4 SEGMENT USERS BASED ON STATISTICAL CLUSTER ANALYSIS

In this step, statistical algorithms take an active role in guiding the personas creation in which similar users with regard to their behaviours and attitude are grouped together into clusters. To simplify, the researcher feed a set of variables into statistical analysis software, and it looks for naturally occurring clusters based on some set of commonalities. It tries many different ways of segmenting users through an iterative process. In Chapter 4, Section 4.6.3, cluster analysis was used to discover natural groupings in the data and to group similar users together with regard to their behaviours and attitudes to feedback acquisition.

6.4.5 CREATE A PERSONA FOR EACH SEGMENT (IN COLLABORATION WITH DOMAIN EXPERTS)

The final step towards the creation of persona is taking the clusters resulted and making them real. This can be done by adding names, photos, and stories to each cluster to transform them into real people. In this stage, four initial personas were developed based on the four clusters presented in Chapter 3, Section 4.6.3 (see Appendix 5. For the initial personas)

6.4.6 DOMAIN EXPERTS INVOLVEMENT

Generally speaking, domain experts' involvement can have a high effect on the study's outcome and the acceptability of its results in the wider community (Gordon 1994). Domain experts' involvement in the persona creation process highly impacts the validity and quality of the created personas. In this phase 11 experts (see Table 17) from industry and academia were involved to evaluate, validate and further improve (e.g. by adding more information to the personas based on their experience) the resulted personas and assess the effectiveness and efficiency of using them to inform the design of feedback acquisition and how they can be used. The selection of experts was as follows:

- **Industrial experts**: six experts from four highly successful companies in the domain of feedback acquisition were interviewed. Two are small-medium enterprises (SMEs) and the other two are large scale international companies. The work of the two SMEs is primarily on customers' feedback acquisition, analysis and reports generation and they have a noticeable record of success stories with some of the world's largest brands. The industrial point of view is vital to assess representativeness and validity of the users' behaviour each persona represents and encapsulates due to their profound experience with users' behaviour and groups to feedback acquisition. It is also important to assess the effectiveness and efficiency of using personas to inform the design of feedback acquisition and how they can be adopted.

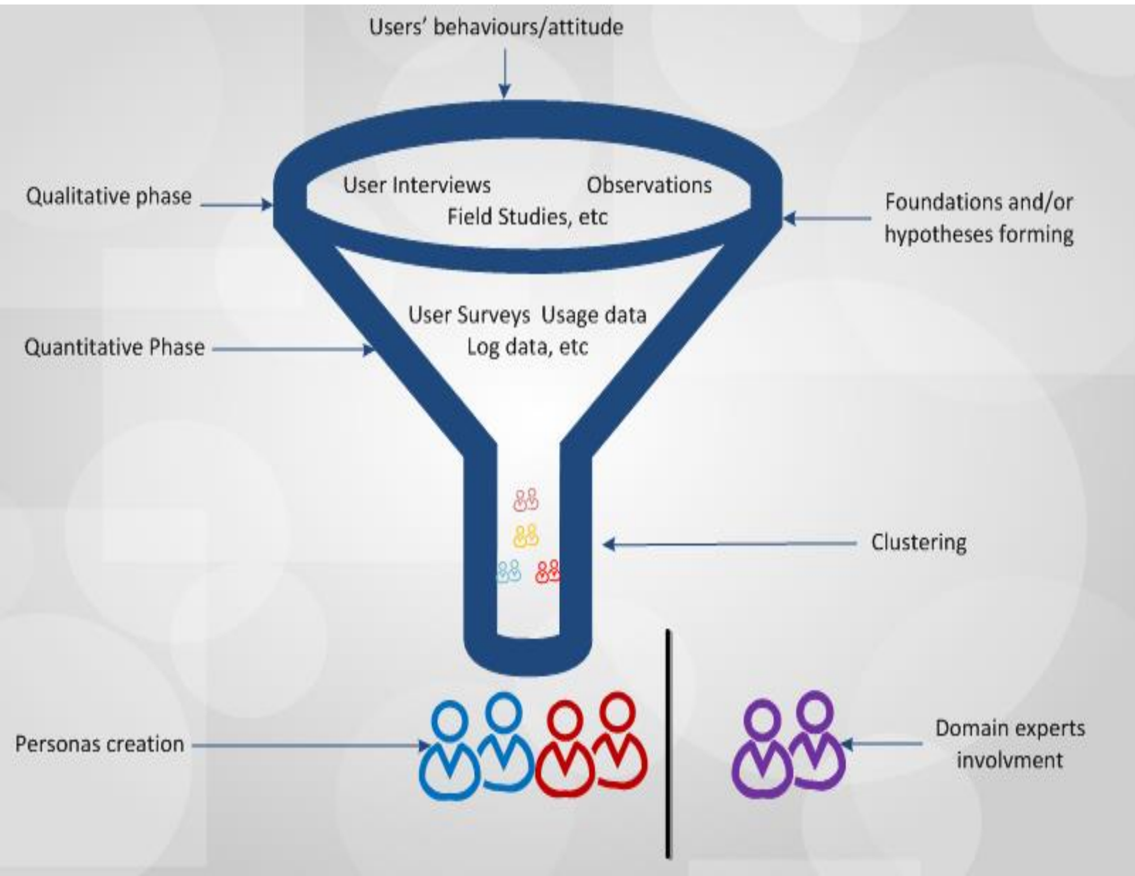


FIGURE 20 A VISUAL VIEW OF OUR PERSONA CREATION APPROACH.

TABLE 17 EXPERTS CHARACTERISTICS.

Experts	Sector	Years of experience	Expertise/Role
Expert 1	Industry (Small-medium Enterprise)	16	Managing director and co-founder
Expert 2	Industry (Small-medium Enterprise)	3	Client service director
Expert 3	Industry (Large Enterprise)	15	Principle Engineer and user UE expert
Expert 4	Industry (Large Enterprise)	15	Researcher and user-centred design expert
Expert 5	Industry (Small-medium Enterprise)	9	Product support manager
Expert 6	Industry (Small-medium Enterprise)	4	Sales manager
Expert 7	Academia	6	User-centred design expert
Expert 8	Academia	7	HCI expert
Expert 9	Academia	11	Persona expert
Expert 10	Academia	9	HCI expert
Expert 11	Academia	4	User-centred design expert

- **Academia experts**: five experts from academia who are highly experienced and knowledgeable in user-centric design and persona (evidenced by quality publications and track record) were interviewed. The academia point of view was valuable to assess the design and representation of the personas (i.e. style and format). It was also important to assess the semantic and understandability of the personas.

6.5 EXPERT INTERVIEW DESIGN

A Semi-structured interview protocol was developed to discuss and assess the validity, representativeness, adoption and design of the initially four created personas. The personas were introduced to the experts prior to the interviews to allow them to familiarize themselves with the personas and provide a better reflection on them (see Appendix 5 for the interview script). Seven interviews were conducted face to face whereas the rest were conducted online using Skype due to accessibility difficulties to some experts. Each interview lasted for about 1 hour and at the beginning of each interview session, each expert signed a consent form.

Experts were invited by an email containing a brief description of the purpose of the interview and asking them to participate in it. Experts were also informed about how their input to the study will be used. The data collection took place between January 7 and February 4, 2015. The response rate was high (11 out of 15) which is an indicator that the field is relevant and timely especially to users' feedback, personas and Requirements Engineering which are primary research areas of the experts.

6.6 PERSONA REPRESENTATION AND BUILDING COMPONENTS

Generally a persona represents users'/people's behaviour patterns, goals, skills, attitudes towards certain product/software plus a few fictional personal/demographic details to make it a realistic character. However, in computing fields, such as HCI, there is a lack of detailed studies consensus on what information should be contained in a persona, how this information should be represented and used to impact the design process of software (Idoughi et al. 2012). Goodwin (2001) suggested that when creating a persona a researcher should focus first on the critical information for the design such as: the workflow and behaviour patterns, goals and attitudes of the persona, then adding the personal/demographic information (can be fictional and based on designers' own assumptions (Chang et al. 2008; Nielsen and Storgaard Hansen 2014), such as what the persona does after work (i.e. he goes home to watch movies with his dog).

Courage and Baxter (2005) suggested a more concrete/detailed approach for representing personas than Cooper's one (Cooper 1999a). They introduced a set of a persona's components combined with a textual formatted guide to the construction of personas. These components are: Photograph, Identity, Status, Goals, Knowledge and Experience, Tasks, Relationships, Psychological profile and Needs, Attitude and Motivation, Expectations, Disabilities (Courage and Baxter 2005).

This thesis refines these components and proposes common building constructs for the creation of personas for a socially-aware and adaptive feedback acquisition that fit the context of this thesis and the information availability of the previously conducted studies. These building constructs are meant to act as a guide to help software engineers creating personas (from scratch) or modifying personas with regard to feedback acquisition (see Table 18). For example, developing a socially-aware and adaptive feedback acquisition for users' with certain disabilities/needs (e.g. dementia patients) could require the creation of new personas that could represent differences among their behaviour. However, these building construct are not meant

TABLE 18 BUILDING CONSTRUCTS/ COMPONENTS FOR THE CREATION OF PERSONAS FOR A SOCIALLY-AWARE AND ADAPTIVE FEEDBACK ACQUISITION.

components	Description
Identity	includes a short statement/status describing the overall persona's attitude to feedback acquisition (i.e. anti-user of the application)
Profile (fictional)	Includes the first name and a picture of the persona. It also includes a description of basic demographic information such as age group, gender, profession, etc. *Note: in this work, fictional information is only meant to bring life to the persona and make it memorable and should not impact the design of the feedback acquisition.*
Goals	Indicates persona's goals of responding to feedback requests in software applications.
Behaviour	Describes persona's behaviour and attitude to feedback acquisition. This can (not necessarily) include: the method preferred for feedback acquisition, the motivations to give feedback, the discouragements elements, the social factors that affect the personas behaviour and the persona's privacy concerns.
Culture Suitability	Indicates the persona's suitability to a certain culture. *Note: in this thesis, culture suitability does notrestrict a persona to a certain culture. It just gives a slight and initial indication of its potential suitability to that culture. This thesis mainly studied the difference between western and middle eastern only.*

to be restrictive and software engineers could enrich them with more components when needed or preferred (e.g. the persona’s preferred language, graphical interface, etc).

6.7DEVELOPED PERSONAS

By enriching the results of the previously conducted studies on users’ behaviour to feedback acquisition with the experts’ answers, the author was able to assess, validate and refine the initially created four personas which are Linda, Jack, Mark, Amy. This process resulted in a total of 7 personas (the three added personas (Sara, Hana and Richard) were developed based on the analysis of the experts’ responses) that encapsulate diverse behaviours of users to feedback acquisition. Some personas are also enriched with information about the cultural impact on users’ behaviours to feedback acquisition as discussed in Chapter 5, Section 5.4.

Each persona encapsulates the knowledge of potential users in relation to feedback acquisition in software applications which was gathered from the conducted user studies (see Chapter 4, 5). The created personas are meant to help software engineers to understand and perhaps predict the behaviour of the users in order to guide the design of an adaptive feedback acquisition towards better functionalities, feedback quality and users’ satisfaction. The created personas are summarized in Table 19. In addition, in Figures 21, 22, 23, 24, 25, 26 and 27 a complete view of the developed personas is introduced.

Experts also provided insights on each of the seven developed personas. An industrial expert commented on Linda’s behaviour and representativeness of the user group she

TABLE 19 SUMMARY OF THE DEVELOPED PERSONAS FOR THE DESIGN OF AN ADAPTIVE FEEDBACK ACQUISITION

Persona’s name	Profile	Statement
Linda	Privacy tolerant and socially ostentatious	“Giving feedback is a social and community experience and it helps to feel among others”
Jack	Privacy fanatic and generous	“I think emails are good if you want someone to actually sit down and write a couple of sentences about how they feel about your service popups and other 'push' mechanisms intrude & interrupt flow.”
Mark	Passive and stingy	“I find it problematic, hindering and unprofessional to send me any kind of feedback requests. If I'm not happy with something I will go to their website and complain to them”
Sara	Incentive seeker	“What’s for me in it? In fact, I wonder why people would give feedback for free.”
Hana	Perfectionist/complainer	“I’m perfectionist and I always seek perfection, If I tiny thing is wrong then of course I will speak”
Richard	Loyal and passionate	“If I’m passionate about something, can’t stand negative reviews about it. I would always defend it. As simple as that”
Amy	Impact seeker	“The benefits of my feedback are always not clear to me as a user”

encapsulates as *"I can absolutely see that's a common behaviour type and a mental approach to it, yes, that's a clearly identifiable set of individuals, heavy user of social software, considerate view about giving feedback and how it helps individuals and her place in the social network"*. Another industrial expert reflected on her behaviour as a growing trend especially with social media websites such as YouTube, Facebook and Twitter. Another industrial expert commented on Jack's behaviour as one of current observed behaviour of users to feedback requests especially when it comes to privacy concerns. The expert believed that software companies nowadays are not doing a good job of explaining how feedback is used and collected which can trigger a privacy concerns to some people. This can eventually harm the software and people's trust in it.

In relation to Mark, an expert form academia commented on Mark's behaviour and representativeness of the user group he encapsulates as *"Yeah, I'm Mark. Absolutely. The method Mark prefers [which is passive?] is really useful for business professionals instead of proactively asking them for feedback"*. In addition, one expert from academia believes that Sara is the person who does not really want to give feedback but will do so for the sake of the incentive. The only problem is that the quality of her feedback is always questionable since the motive behind it is only the desired incentive. In relation to Hana, an industrial expert indicated that Hana's group of users creates troubles to feedback collectors as they would give negative feedback which is not necessarily a good reflection of the quality but only their innate desire to criticize and optimize.

Another academia expert commented on Richard's behaviour as *"a lot of people are like Richard, no matter what you do, they're still going to love your brand, and they're still going to engage with the brand. Once they are so in love with that brand, they would do anything for it -- even if that brand really annoyed them"*. In relation to Amy's persona an industrial expert believes that software companies are not paying enough attention to close the loop with their users and keep them inform about how their feedback impacted the software. Ideally they should say to their users *"you asked us this, we've done that, job done and there isn't nearly enough of that done, it is still into this blackness"*. However, people give feedback, nothing seems to happen and this is why users or costumers lose interest in the service.

Looking at the personas, one can observe that some of the personas share similar characteristics. However, this should not mean they should be catered for similarly by the adaptive feedback acquisition. For example, Hana, Sara and Richard share the same characteristics with regard to the feedback method they respond to. However, each one of them has different motivations to give feedback which impact the quality of their given feedback. This thesis emphasizes that the adaptive feedback acquisition should cater for these nuances regardless of the shared characteristics among personas.

Linda

Profile: Privacy tolerant and socially ostentatious
Age:20
Gender: Female
Job: Undergraduate student
Socially affected to give feedback: Yes
Culture Suitability: Middle Eastern-like

Statement: *"Giving feedback is a social and community experience and it helps to feel among others"*.
Goals: Impact the software with her feedback + raising others awareness about the used software + being socially recognised.

Behaviour to feedback:
Linda is an undergraduate university student and spends a great deal of time on her computer studying as well as heavily social networking (i.e. Facebooking). **[Discouragement]** In general, she is not a big fan of the idea of dull and typical feedback requests and reminders coming from software applications. **[Motivation]** However, she gets interested in replying to feedback requests when the feedback requests socially motivate her to do so (i.e. by making her socially recognized for her helpful feedback). This is perhaps due to her likeness of social networking and the time she spends socialising with others/friends on the internes which made her motivated towards socially enriched feedback requests. Generally, Linda is positively affected by one or more of the following social factors to give feedback:

- **Volume of already given feedback**: She gets enthusiastic to give feedback when there is low number of feedbacks already given on a software. She believes it's helpful to increase the number of given feedback which will then result in other users having a better and richer idea about the software.
- **Visibility and similarity of other users' feedback**: Linda also gets more interested to give feedback if she is able to see other users' feedback on the software first and then having the option to accept/reject to give feedback.
- **Social recognition:** Since Linda appreciates social networking and gives it a great deal of her time, she likes to be socially recognized for her given feedback which she believes could help others and make her socially popular.
- **Feedback acquisition as a social activity:** This social factor also makes Linda motivated to give feedback as well as engaging with software. For example, she gets enthusiastic to feedback requests when she is able to visualize how her social friends are rating a certain software and how their feedback influenced the trend in her community.

[Method] In addition, Linda prefers to be approached for feedback by using hints and tips to gather her feedback (e.g. by telling her that she can go to a feedback centre for this purpose and leave her feedback) or by using an online method as a second option (i.e. popups while she is using the software). **[Privacy]** Interestingly, Linda does not mind to be implicitly reached for feedback (e.g. implicitly collecting information about her software usage)

FIGURE 22 LINDA: PRIVACY TOLERANT AND SOCIALLY OSTENTATIOUS.

Jack

Profile: Privacy fanatic and generous
Age:35
Gender: Male
Job: Researcher
Socially affected to give feedback: Yes
Culture Suitability: Middle Eastern-like

Statement: *"I think emails are good if you want someone to actually sit down and write a couple of sentences about how they feel about your service popups and other 'push' mechanisms intrude & interrupt flow."*

Goals: Impact the software with his feedback + raising others awareness about the used software + being socially recognised.

Behaviour to feedback:

Jack as a researcher spends most of his time on the computer working on his research as well as networking with other researchers. **[Motivation]** Jack believes in the power feedback in general and its positive impact. He is a very positive person towards feedback requests and reminders coming from software application.
[Method] However, he prefers to be asked for feedback in an offline way (i.e. through emails or text messages).
[Discouragement] He believes online feedback request (i.e. popups) could somehow be intruding and interrupting especially when he is working on his research and deeply thinking.
[Privacy] In addition, Jack is always concerned about his privacy and therefore he does not accept to implicitly collect feedback from him (i.e. tracking his usage of the software). **[Motivation]** In addition, Jack is a socially motivated feedback provider and his willingness to give feedback is positively influenced by one or more of the following social factors:

- **Social recognition:** He likes to be socially recognized for his valuable and trustworthy feedback which he believes could help others and raise the social awareness about the software in use.
- **Volume of already given feedback:** He gets enthusiastic to give feedback when there is high number of feedbacks already given on a software. This means to jack the software is popular and deserves his feedback.
- **Visibility of other users' feedback:** Jack also gets more interested to give feedback if he is able to see other users' feedback on the software first and then having the option to accept/reject to give feedback.

FIGURE 21 JACK: PRIVACY FANATIC AND GENEROUS.

Mark

Profile: Passive and stingy
Age:50
Gender: Male
Job: Business man
Socially affected to give feedback: No
Culture Suitability: Neutral

Statement: "*I find it problematic, hindering and unprofessional to send me any kind of feedback requests. If I'm not happy with something I will go to their website and complain right to them*".
Goals: Get my voice heard when I need.

Behaviour to feedback:

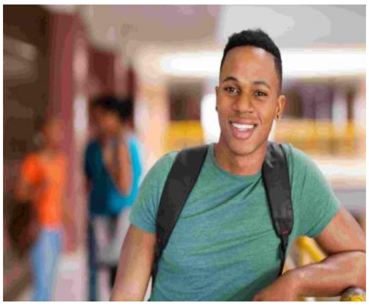
Mark is a business man and he spends a lot of time on his computer working on his business. He holds a very negative view about feedback request coming from software applications. He does not have the time to be responding to feedback request due to his heavy workload.

[Discouragement] Mark thinks feedback request coming from software applications can waste his time and he doesn't tolerate to be asked for feedback at all (whether it's online of offline feedback request). In fact, he thinks that feedback requests that interrupt him while he is working are an impolite way to get information out of him. Since Mark doesn't tolerate to be asked for feedback at the first place, he is not affected by any social factors to give feedback at all (i.e. social recognition does not make him happy to give feedback).

[Method and Motivation] However, Mark believes that there should be a channel for him to deliver his opinion whenever he likes by making him able to submit his feedback on a voluntarily base and without being proactively asked by the software (i.e. through a contact us form).

FIGURE 24 MARK: PASSIVE AND STINGY.

Richard



Profile: Loyal and passionate
Age:18
Gender: Male
Job: High school student
Culture Suitability: Neutral

Statement: *"If I'm passionate about something, can't stand negative reviews about it. I would always defend it. As simple as that"*

Goal: To feel better when defending and praising what he is passionate about.

Behaviour to feedback:

Richard is a high school student and he is highly passionate about his new smartphone. His passion makes him blind to any drawbacks of his smartphone. He is not a big fan of the idea of being asked for/reminded to give feedbacks by software applications.

[Method] However, when it comes to something he loves he happily respond with a positive input regardless of the way he is being asked for the feedback (i.e. offline or real-time)

[Motivation] The main motivation that drives Richard willingness to give feedback is his passion and loyalty about certain product/software. **[concerns]** However, the quality of his feedback can be questionable since he tends to exaggerate in praising and defending what he loves.

FIGURE 23 RICHARD: LOYAL AND PASSIONATE.

Sara

Profile: Incentive Seeker
Age:28
Gender: Female
Job: Supermarket cashier
Culture Suitability: Neutral

Statement: *"what's for me in it?, In fact, I wonder why would people give feedback for free?"*
Goal: To win tangible incentives.

Behaviour to feedback:

Sara is a supermarket cashier and she highly believes in tangibly rewarding customers for their loyalty (i.e. customers win a free product after certain visits to the supermarket). She thinks the same applies to feedback request coming from software applications.

[Motivation] She argues that her the effort and time she spends giving feedback should be tangibly rewarded.

[Method] As long as there is an incentive, she is happy to respond to feedback requests regardless of the way she is being asked for the feedback (i.e. offline or real-time method).

[Concerns] However her response would be mostly positive and not well thought. This is due to her desire to get the incentive no matter how the feedback she gives looks like. This can have a negative effect on overall reputation of the software/product due to the low quality feedback that doesn't objectively represent her experience.

FIGURE 26 SARA: INCENTIVE SEEKER.

Hana

Profile: Perfectionist/complainer
Age:24
Gender: Female
Job: Hotel receptionist
Culture Suitability: Neutral

Statement: *"I'm perfectionist and I always seek perfection, If I tiny thing is wrong then of course I will speak"*

Goal: To express her disappointment and sometimes ability to criticise+ seeking perfection.

Behaviour to feedback:

Hana is a hotel receptionist and her job requires her to seek perfection due to the size of criticism she receives from the hotel guests.

[Method] She wouldn't mind to be asked for feedback by software applications and she would always reply but mostly with a negative response regardless of the way she is being asked for feedback (i.e. offline or real-time method). She is a very picky person and never get satisfied no matter how good is the provided software/service.

[Motivation] The main motivation that drives Hana willingness to give feedback is her desire to achieve perfection and her ability to criticise any thing.

[Concerns] However, the quality of her feedback can be questionable since she tends to exaggerate in criticism which could eventually result in an exaggerated harm to the software/product.

FIGURE 25 HANA: PERFECTIONIST/COMPLAINER.

Amy

Profile: Impact seeker
Age:29
Gender: Female
Job: School teacher
Socially affected to give feedback: No
Culture Suitability: Slightly Western-like

Statement: *"The benefits of my feedback are always not clear to me as a user."*

Goal: To consider her feedback and see the impact of it on the software.

Behaviour to feedback:

Amy is a school teacher and spends a great deal of time on the internet reading and researching educational related topics. She is not a big fan of the idea of being asked for/reminded to give feedbacks by software applications.

[Discouragement] She does not believe her given feedback is going to be considered or lead to any changes/improvements on the software. She does not even get influenced or motivated by any social factors to give feedback (i.e. visibility of others feedback on the software doesn't really make her want to give feedback).

[Method] However, sometimes she can be tolerant to online feedback request (i.e. showing her a feedback popup dialogue while she is using the software).

[Discouragement] This is due to the fact that she doesn't accept the idea of having her email inbox filled with feedback requests or feedback reminders.

[Motivation] She tolerates the online ones since she has the control to respond or dismiss it at only one click sometimes. In conclusion, Amy can act more positively to feedback request if her feedback is considered and she can see its impact on the software.

FIGURE 27 AMY: IMPACT SEEKER

6.8 PERSONAS BENEFITS FOR FEEDBACK ACQUISITION DESIGN

Experts identified some of the benefits that the previous personas can offer to software engineers when adopted to inform the design of an adaptive feedback acquisition. These benefits are as follows:

6.8.1 ENGAGEMENT

Several experts believed in the power of the previous personas to engage software engineers with the design of the feedback acquisition. This is perhaps due to the nature of the developed personas in which fictional information make them more interesting and attractive. An expert commented *"I assume they are very engaging and fun to work with. The picture and other profile information makes you feel you are working with a real person. This is really different from working with only dull descriptions about users".*

6.8.2 DISCUSSION

Generally speaking, personas are highly powerful to stimulate discussions among the design team (Nielsen 2002; Canossa and Drachen 2009; Miaskiewicz and Kozar 2011). Experts also believe the introduced personas can lead to fruitful discussions among the design team of an adaptive feedback acquisition. This discussion can ultimately lead to a better understanding and identification of persona behaviours to feedback acquisition. An expert commented *"I would use these personas to understand the users' behaviours to feedback acquisition. Actually I find it a good way to stimulate discussion and help designers better understand their users"*.

6.8.3 MORE EFFICIENCY IN IDENTIFYING REQUIREMENTS

Identifying users' requirements and preferences on feedback acquisition is a highly challenging task. The experts' point of view suggests that the introduced personas offer a suitable solution to address this challenge. This is illustrated in one of their comments *"If I put my software engineer hat I would say, using these personas could save me a lot of time and effort to identify users requirements and preferences to feedback acquisition. Different people have different characteristics, and it depends on what your software is targeting. If it is targeting Linda, then your feedback acquisition mechanism would look different from Mark. Depending on these different behaviours represented by the personas, you could derive your software requirements"*

6.8.4 RELEASE YOUR THINKING FROM YOUR OWN MENTAL MODEL

Personas can noticeably reduce designers' unconscious bias when designing software. It helps them avoid being limited to their own mental models about how users would be like (Nielsen 2002; Canossa and Drachen 2009; Miaskiewicz and Kozar 2011). Some experts believe the introduced personas can help software designers realize how people are different from themselves when it comes to feedback acquisition.

Generally, this can positively impact the success of software since its designers were able to limit the effect of their own mental models on software design. An expert commented *"I can see how those personas can aid individuals realizing how the users/customers are different from themselves in the context of feedback acquisition. They would open the process up enormously because the danger is we all pursue things on our own, preconceptions, and of course, mine is different than someone else's is"*

6.8.5 VALIDATION

Software validation is vital phase that determines software success or failure. Personas are shown to be useful for software/product validation purposes in which proposed designs, features and solutions can be reviewed and evaluated against the needs described by an individual persona (Nielsen 2002; Canossa and Drachen 2009; Miaskiewicz and Kozar 2011). Experts agreed that the introduced personas can be highly useful to validate the developed feedback acquisition against the behaviour and preferences of the persona it was developed for. An expert said *"validation is always a bit hit and miss. I think these personas would be definitely a useful* *tool for validating your developed feedback acquisition. If the developed acquisition is meant to fit Jack but it does not seem to fit his motivations then certainly there is something wrong"*

6.9DISCUSSION

The literature contains several approaches on the use of personas to inform the design of software applications such as (Blomquist and Arvola 2002; Guðjónsdóttir and Lindquist 2008). However, the majority of these approaches do not noticeably employ the power of personas to directly inform the actual design of the software and limit their usage to the abstract level as communication tool. This led to a gap and lack of tractability between personas and the actual design thus the underestimation of personas power.

On the down side, Personas have been criticized mostly because they could be too fictional and have no clear relationship to real users' data and therefore any data gathered cannot be considered scientific (Pruitt and Grudin 2003). However, in this thesis persona creation is highly based on data of actual users gathered from empirical studies previously conducted (see Chapters 4, 5) except for some accessorial data which were only used to bring life to the personas). Also, this utilization of personas is not to restrict users to them but rather to initiate a discussion between the stakeholders involved in the feedback acquisition process. This means that the refinement of these personas, creation of others, and eliminating some of them would be still possible within the context and throughout the life time of a certain project or software.

This also means that the refinement and evolution of these personas would most likely be different amongst projects depending on the specifics of each project and also the nature of users' involved, products and services which are the subject of feedback, etc. In addition, this thesis emphasizes that personas should be clearly communicated to software engineers to ensure the clearly understand how to use them. For example, they should be aware that some fictional data in the persona (i.e. the picture or name) are only to bring life to it and make it memorable and should not impact the design of the software. Similar precautionary procedure should be followed when introducing personas to users who may simply reject being similar to a certain persona because of the picture of the age.

Additionally and in contrast to the benefits of personas mentioned previously, some experts believed that the use of personas could limit the thinking and imagination of software engineers to only the set of proposed personas and could result in them not considering other users who were not represented by personas. One of the experts commented *"the only thing that concerns me about personas is that, you perhaps start to isolate your thinking and segregate things a bit too much. It is probably not worth relying on thinking, 'this is the 7 type of people we've got and that's it, that's the end of it'. So it probably could isolate your thinking a little bit and maybe lead you down the wrong track"*.

This thesis also does not claim the developed personas cover all users' types to feedback acquisition and further research in this area could result in more or less personas. However, it assumes that the developed personas cover the most common and observed types of you users' behaviours to feedback acquisition based on the studies previously conducted as well as the expert survey undertaken. These personas could create different scenarios and thus are not only meant to stereotype users but rather to generate the space of variability and commonalities on how feedback should be requested and obtained.

Although this thesis carefully followed the principles in developing the personas and conducting the expert interviews, this work would still have three main threats to validity:

- While the methodology was effective in identifying users' behaviour and creating personas to reflect their behaviour with regards to feedback acquisition, it is possible that the personas did not capture all the aspect and factors that can affect the users' behaviours it represents.
- The developed personas were validated from an expert point of view only. The users have not contributed to the validation and creation process. Allowing users to contribute to the validation of the personas can result in a more robust set of personas.
- A common threat to the validity when designing an interview is whether the questions were understood by all experts as intended. This threat was somehow addressed as the interview script went through iterative revisions and modifications by two research members to ensure clarity.

6.10 SUMMARY

To conclude, this chapter provides a clearer view and a deeper understanding of users' different behaviours to feedback acquisition represented in seven personas of users' behaviour to feedback acquisition. It also provides software engineers with the building constructs for developing personas for a social-aware and adaptive feedback acquisition. Additionally, this chapter gives a clear view on how the introduced personas can benefit software engineers when designing an adaptive feedback acquisition.

The next chapter introduces a **P**ersona-based Method for **A**daptive **F**eedback **A**cquisition (PAFA). PAFA adopts the proposed personas to inform the design of an adaptive feedback acquisition. The method relies on some novel techniques which have not been used in the context of feedback acquisition before such as collaborative filtering and personality questions. The investigation of the method (see Chapter 7) will investigate these elements.

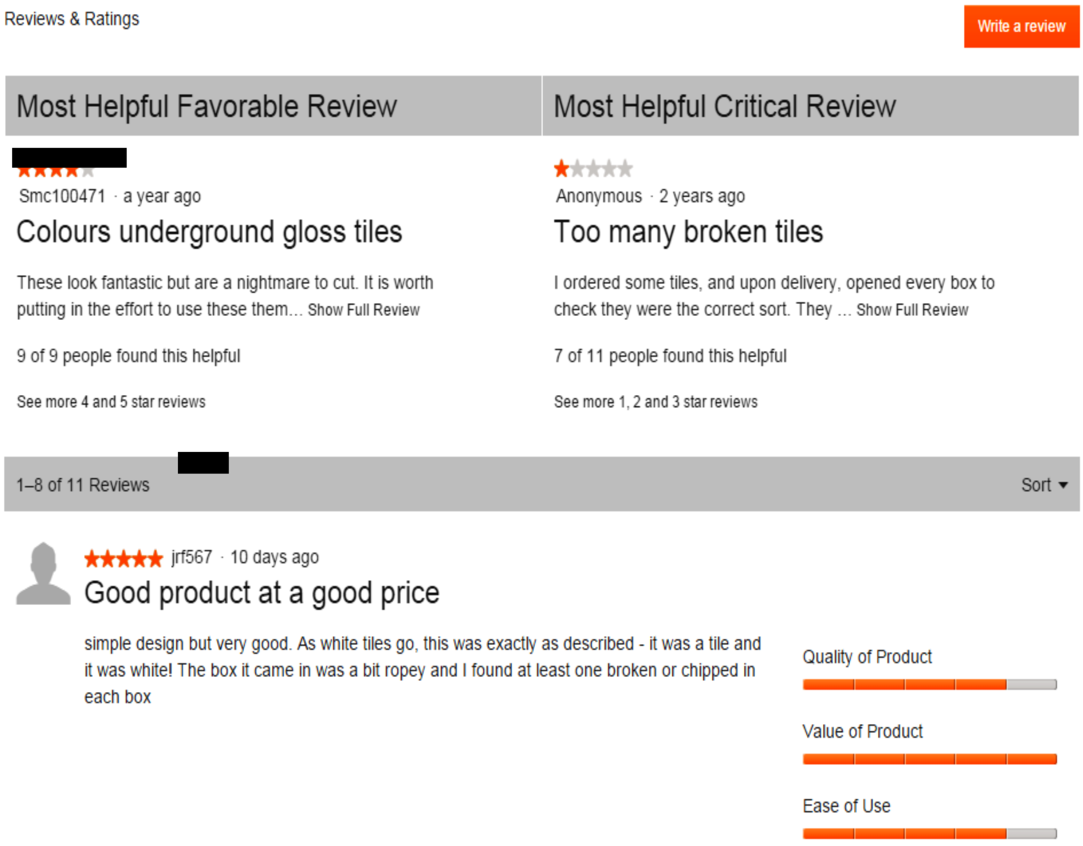
7. Chapter 7: PAFA- A Design Method for Adaptive Feedback Acquisition

In relation to the previously introduced personas (see Chapter 6, Section 6.7), one open question is that, as a software engineer, how would I use these personas to inform the design of an adaptive feedback acquisition? To answer this question the author worked with experts to firstly identify candidate approaches for this purpose (see Chapter 6, Section 6.4.6). In the light of these identified approaches and the literature review conducted on Persona-based design as well as the author's experience obtained through the empirical case studies done in this thesis, this thesis proposes a five-phase **P**ersona-based method for the design of an **A**daptive **F**eedback **A**cquisition (PAFA).

- ***Running example***

In order to provide a clear and efficient illustration of the PAFA phases, this thesis adopts the case of one of the large businesses which has a successful story with regard to costumers' feedback acquisition. The development company of the feedback acquisition is called Feedback Ferret who collaborated with the author through the expert interviews phase. The name of the business is anonymized and referred to it as CompanyX. Feedback Ferret works on developing methods for customers' feedback acquisition, analysis and reports generation and their clients include world's largest brands such as BMW, Asda, Argos, etc. CompanyX needs to find out how their customer experience match up to their business objective to deliver outstanding services, product range and value for money.

To tackle this need, CompanyX employs Multi-channel customer feedback acquisition through Text messages, E-mails and Web feedback forms which allow Customers to give their closed and open-ended feedback. Customers optionally provide their phone numbers and emails at the checkout when making any purchase. This information is then used by CompanyX to send an automated SMS and email feedback requests to the customers sometime after making the purchase. Customers can also (on a voluntary basis) go to the company website to fill in a feedback form when needed and then click submit to deliver their voice. The feedback requests sent to costumers (SMS or email) contain a link to a feedback form where customers can star rate the provided service and comment in a free text style if needed. Customers can also like, dislike or comment on others' feedback. Figure 29 shows a snapshot of CompanyX feedback page:



Reviews & Ratings

Write a review

Most Helpful Favorable Review	Most Helpful Critical Review
[redacted] Smc100471 · a year ago **Colours underground gloss tiles** These look fantastic but are a nightmare to cut. It is worth putting in the effort to use these them... Show Full Review 9 of 9 people found this helpful See more 4 and 5 star reviews	★☆☆☆☆ Anonymous · 2 years ago **Too many broken tiles** I ordered some tiles, and upon delivery, opened every box to check they were the correct sort. They ... Show Full Review 7 of 11 people found this helpful See more 1, 2 and 3 star reviews

1–8 of 11 Reviews [redacted] Sort ▾

★★★★★ jrf567 · 10 days ago

Good product at a good price

simple design but very good. As white tiles go, this was exactly as described - it was a tile and it was white! The box it came in was a bit ropey and I found at least one broken or chipped in each box

Quality of Product

Value of Product

Ease of Use

FIGURE 29 SNAPSHOT OF COMPANYX FEEDBACK PAGE.

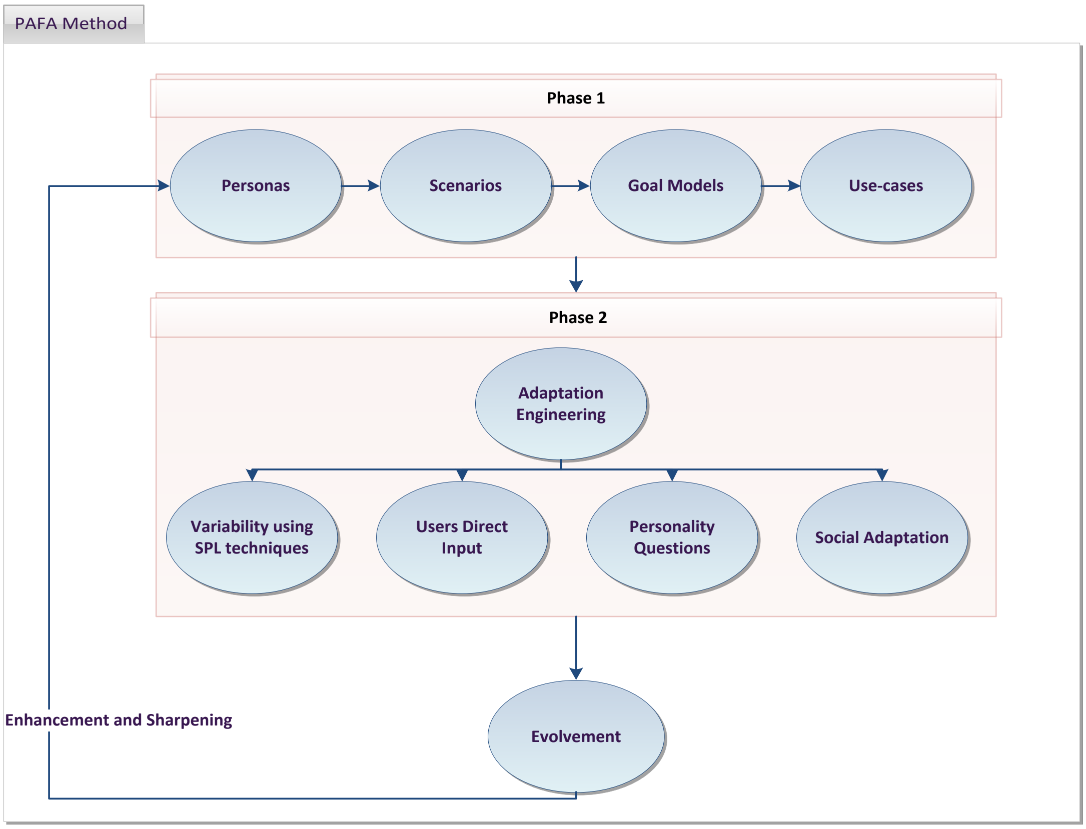


FIGURE 28 A VISUAL VIEW OF THE PAFA METHOD FOR THE DESIGN OF AN ADAPTIVE FEEDBACK ACQUISITION.

However, the feedback requests provided by CompanyX were designed for all customers universally. This led to a lack of adaptivity to the company's different types of customers' behaviours and preferences when it comes to feedback provision. This lack of adaptivity could harm customers' satisfaction and experience which can then harm the company's reputation in the long-term.

In the following, PAFA adopts the case of CompanyX with feedback acquisition to showcase how PAFA method can be applied in practice and used to improve the efficiency and adaptivity of current feedback acquisition processes. Led by Personas, PAFA goes through five phases (personas to scenarios, scenarios to goal models, goal models to use-cases, adaptation engineering and evolvement). These phases are motivated by the need to adaptively consider diversity in users' behaviours to feedback acquisition and hence the proposition of PAFA's phases as presented in Figure 28.

It is important to highlight that the seven personas introduced in PAFA are meant to serve as a generic foundation to drive the design process of feedback acquisition for popularly used applications/software. However, the long and short term evolution of these personas would most likely be different amongst projects depending on the specifics of each project and also the nature of users' involved, products and services which are the subject of feedback, etc. In addition, PAFA does not restrict software engineers to adopt all the seven personas in the design of the feedback acquisition. Stakeholders should decide on the personas they need to cater for in their feedback acquisitions. For example, CompanyX might decide not to cater for Sara (incentive seeker) to avoid low quality feedback.

In principle, the preferences of a service provider would decide their selection of the personas to support. It may also lead to creating variation of the baseline personas to fit their definition of a relevant user or client. This may require preferences on the costs, the speed of getting feedback, the need for high quality feedback, targeting certain age bands and culture backgrounds, etc. This thesis emphasized that such decisions are taken alongside PAFA and not only at the initial stage of selecting personas to cater for. This will become clearer when presenting variability models, goal and feature models, where the choice of the technique is subject to such preferences.

7.1FIRST PHASE: PERSONAS TO SCENARIOS

In this phase, software engineers should start the design process with deriving multiple scenarios from each persona following the rules for scenario authoring as discussed in (Davis 1993; Achour 1999; Nielsen 2003). Generally, a scenario is characterised by initial and final states. The initial state describes a precondition for the scenario to be triggered whereas the final state defines a state reached at the end of the scenario (e.g. see Scenario 1, 2, or 3). A scenario is also composed of a flow of several actions which describes the transition from the initial state to the final state (e.g. see Scenario 1, 2, or 3) (Rolland et al. 1998). A scenario should also be authored with respect to the persona's behavioural aspects when applicable such as the persona's goals, motivations, methods, concerns and privacy preferences.

This adds a more detailed description about the personas which gives software engineers a better understanding about the various possible behaviours of users as well as discovery of their requirements/preferences or goals in different contexts (Maiden and Robertson 2005). For example, Linda[14] can have several scenarios when approached for feedback by CompanyX such as:

- **Scenario 1:** Linda buys one of CompanyX products online and then she is asked for feedback after the purchase ended. The feedback request is to evaluate her online shopping experience by using an offline method (email). The feedback request contains a link to a feedback form and does not contain any social aspect in it (i.e. volume of already given feedback). Linda's willingness decrease to give feedback and she ignores the request.
- **Scenario 2:** Linda buys a product X and then a pop up feedback request is triggered to ask her about her satisfaction with the payment method. The pop up shows the number of customers that liked or disliked the payment method. It also shows how her Facebook friends rated the payment system in CompanyX and how their feedback is affecting the overall reputation. Linda's willingness increases and she responds to the feedback request.
- **Scenario 3:** Linda is navigating the CompanyX website looking for product X when a pop up feedback request is triggered to ask her about ease of navigating the website. The pop up feedback request shows her how her previous feedbacks on some of the website aspects were influential and liked/recognized by many costumers. The pop up also shows that her response to could lead to awarding her a badge as an appreciation for her feedback. Linda's willingness increases and she responds to the feedback requests. Linda is then awarded a badge on her profile status.

By using scenarios software engineers can extract more information and knowledge about Linda's requirements and preferences when asked for feedback by CompanyX. This gives more efficiency to the design of the feedback acquisition which will ultimately impact customers/users' satisfaction and the product/company's success. The efficiency stems from the fact that personas guide the decision on what we should be caring of when creating the scenario.

7.2 SECOND PHASE: SCENARIOS TO GOAL MODELLING

Goal modelling is a widely used technique during the early phases of software requirement engineering. It improves the efficiency of the requirement engineering process and offers modelling concepts to represent the rationale of social and technical actors in a socio-technical

[14] The persona Linda is used as a working example throughout the phases of PAFA in this chapter. However, Linda is supported by other working examples (e.g. Jack, Sara, etc) in certain phases of PAFA (e.g. Figure 31) to offer more clarification/explanation of the phases.

systems through notions like goals, softgoals, decomposition, actors and their interaction (Yu and Mylopoulos 1998). Goals are intentions and goal models capture the rationale of actors which nicely fit to the description of personas (Jones and Maiden 2005) and their elaborations as scenarios (Faily and Fléchais 2014).

Scenarios authored in the previous phase can aid software engineering to achieve a better extraction and identification of each personas goals, softgoals and the relationship between the identified goals and softgoals as a basic step to enable the expression of preferences and the qualification between the alternative ways to fulfil goals. Generally, a scenario consists of one or several actions and the combination of these actions describes a unique path leading from the initial to the final states of agents (e.g. Linda). Therefore, scenarios are sufficient enough to express most of the behaviours (Linda's multiple scenarios in the previous section) that are necessary for the purpose of building goal models (Rolland et al. 1998).

This step of PAFA leads to the creation of a goal model that gives a clearer visual and structured view of each persona's goals and the alternatives to reach them. For example, one of Jack's motivations (goals) to give feedback is to positively impact the products of CompanyX with his feedback. However, privacy is an important factor (softgoal) for Jack that, if not respected through at least in one of the alternatives to achieve his goal, can lead to rejecting feedback requests (see Figure 31). Using scenarios to develop goal models is indeed a common practice in requirements engineering as discussed in (Rolland et al. 1998; Jones and Maiden 2005) in which guidelines for such transition were introduced. Figure 30 presents an example of

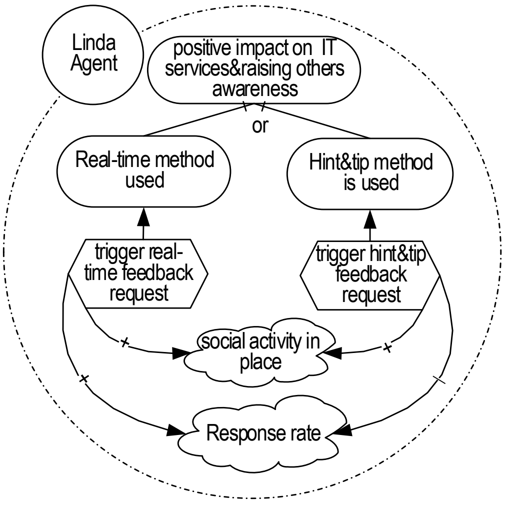

Linda Agent
positive impact on IT services&raising others awareness
or
Real-time method used
Hint&tip method is used
trigger real-time feedback request
trigger hint&tip feedback request
social activity in place
Response rate

FIGURE 31 EXAMPLE OF POTENTIAL SCENARIO-BASED GOAL MODEL YIELDED FROM SCENARIO 3

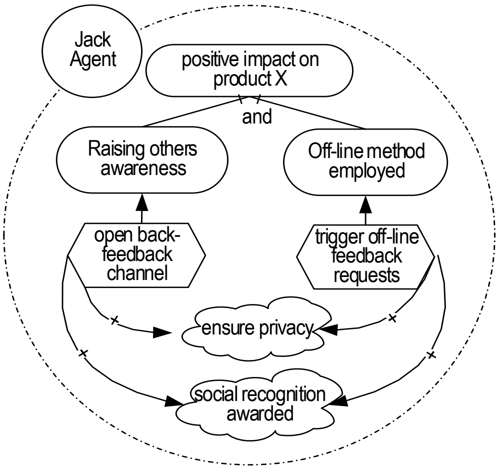
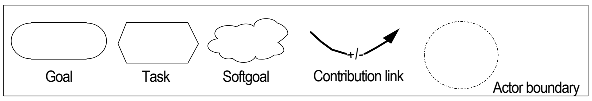

Jack Agent
positive impact on product X
and
Raising others awareness
Off-line method employed
open back-feedback channel
trigger off-line feedback requests
ensure privacy
social recognition awarded

FIGURE 30 EXAMPLE OF ONE OF POTENTIAL GOAL MODELS FOR JACK.

Legend



an initial Tropos (Castro et al. 2002; Bresciani et al. 2004) goal model that could result from Scenario 3 introduced previously.

As previously stated, goal modelling is also a powerful way to represent the possible alternatives to satisfy certain goals (Mylopoulos et al. 1999). Such feature is important for the analysis and the design of adaptive systems (Fickas and Feather 1995). Figure 32 shows an example of 2 alternatives (T1 and T2) to the goal of Sara (the incentive seeker). This gives the software engineers a better knowledge about the possible alternatives and the impact of adopting each alternative which ultimately inform the design of the feedback acquisition. In addition, contextual information can also be added to the developed goal model (Ali et al. 2010) to show how a certain context could affect how goals are met and the alternatives to be adopted to meet them. For example, the context ***c1*** in Figure 32 indicates that the tangible incentive is only given on feedback requests that takes more than 5 minutes to complete.

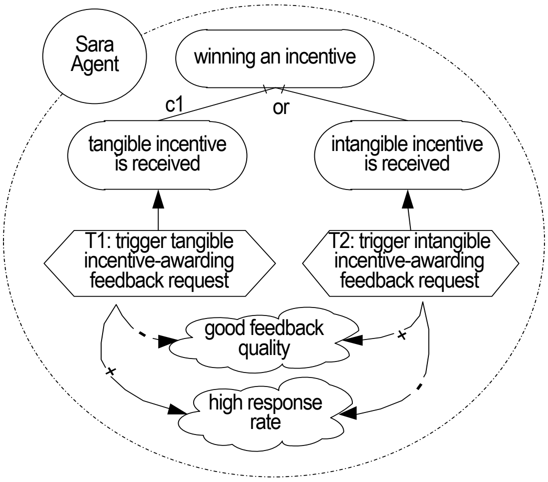

Sara Agent
winning an incentive
c1
or
tangible incentive is received
intangible incentive is received
T1: trigger tangible incentive-awarding feedback request
T2: trigger intangible incentive-awarding feedback request
good feedback quality
high response rate

FIGURE 32, POTENTIAL ALTERNATIVE TO REACH SARA'S GOAL.

7.3THIRD PHASE: GOAL MODEL TO USE-CASES

Although goal modelling provides software engineers with a better understanding of each persona’s goals and softgoals and the different alternatives to reach those goals, it is limited, and probably not meant, to capture the interactions of an actor (persona) with the software. This can lead to missing important information about persona’s requirements. To tackle this issue and in the light of the goal models developed in the previous phase, software engineers should derive use-cases to capture the interactions of each persona with the software.

This can be achieved following approaches discussed in (Santander and Castro 2002; Jones and Maiden 2005) which advocate that combining goal models with use-cases is indeed a powerful way towards a better requirement engineering process. This minimizes the risk of overlooking some of users’ requirements when the design phase starts. For example, for Mark, the use case of Supplying Feedback would have a flow in which all the control is given to him as a primary actor. He would also have the option to configure the feedback collection process through the use case of Control Feedback Collection (see Figure 33 and Table 20). However, for

Linda, the software could be a primary actor for such a use case which could autonomously execute certain actions, i.e. issuing the request and showing her social recognition level. Looking at the goal model in Figure 30, one can derive a use case model for Jack as presented in

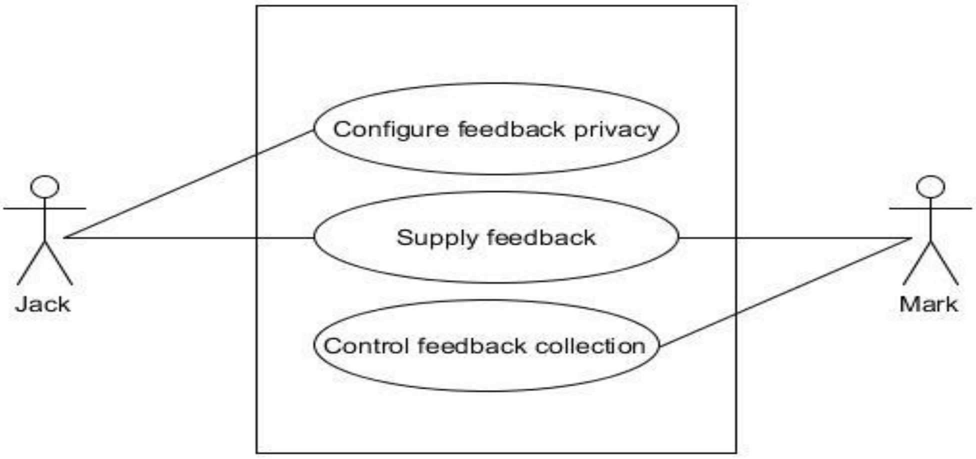


FIGURE 33 FA GOAL MODEL-BASED USE CASE OF JACK AND MARK.

Figure 33 and Table 20.

For gudilines on the generationg use cases from goal models, one can refer to Santander and Castro (Santander and Castro 2002) prposed approach for such atransition (see Figure 34[15]). Their approach deals with goals models developed using i* framework (Yu 2011) which consists of two main modelling components:

7.3.1 Strategic Dependency model (SD)

The SD model represents a network of dependency relationships between various actors in an organisational context. The model describes who an actor is and who depends on the work of an actor in the context of the model.

7.3.2 Strategic Rationale model (SR)

The SR model represents the modelling of the reasons associated with actors and their dependencies and gives a rational on how actors achieve their goals and soft goals. The i* framework can be used in requirements engineering to understand the problem domain. SD models and SR models can then be utilized to generate use cases (Santander and Castro 2002). This can result in capturing the interactions of an actor (persona agent) with the software and a better identification of requirements specification

TABLE 20 THE SPECIFICATION OF THE USE CASE SUPPLY FEEDBACK.

Use Case Name	Supply Feedback
Trigging Event	Buying product x
Brief Description	When a Company'x customer buys a product (online or offline) a feedback request is sent to the customer asking for his overall satisfaction with the service provided.

[15] For more details/guidelines about each step shown in Figure 34, please consult (Santander and Castro 2002).

Actors	Jack and Mark			
Preconditions	Actors visit CompanyX's website and buy a product.			
Post conditions	Actors receive feedback requests and take actions.			
Flow of events	**Actors**		**Feedback system response**	
	Jack	**Mark**	**to Jack**	**to Mark**
	1- Visits CompanyX's website. 2- Navigate the website looking for a product x. 3- Buys the product and leaves the website. 4- Receives an email from CompanyX asking for feedback. 5- Clicks on the link to the feedback form. 6- Gives feedback and click submit.	1- Visits CompanyX's website. 2- Navigate the website looking for a product x. 3- Buys the product and leaves the website.	1.1 Asks for Jacks' email address. 1.2 Sends an offline feedback requests by email. 1.3 Includes a link to the feedback form in the email. 1.4 Explains how the collected feedback is going to be used. 1.5 Assures identity and personal information is anonymised. 6.1 sends occasional updates about the impact of the given feedback on product x.	3.1 print information on how to leave feedback when needed on the payment bill.
Exception conditions	5.1 if Jack does not click on the link, a reminder email is sent by the feedback system. 6.1 if Jack does not continue the feedback form and clicks submit, a reminder email is sent.			

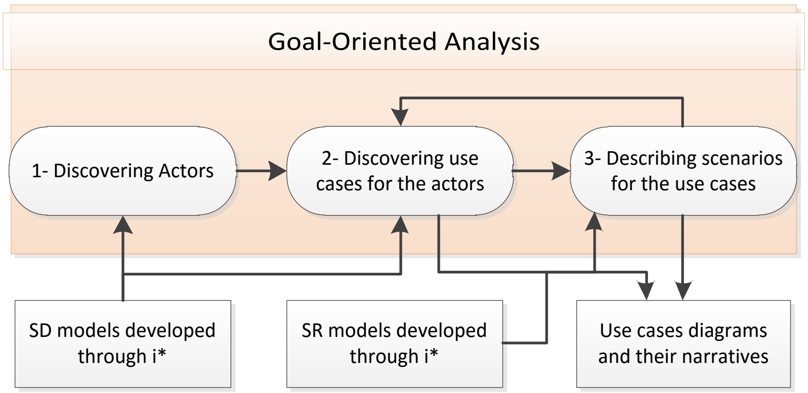

Goal-Oriented Analysis
1- Discovering Actors
2- Discovering use cases for the actors
3- Describing scenarios for the use cases
SD models developed through i*
SR models developed through i*
Use cases diagrams and their narratives

FIGURE 34 STEPS FOR THE TRANSITION PROCESS BETWEEN I* AND USE CASES (SANTANDER AND CASTRO 2002).

In this phase software engineers are also recommended to review the resulted requirements specification, i.e. the goal models and the use cases and their narratives, against the needs of the personas. This can help minimizing the risk of missing requirements at later stages which can negatively impact the success of feedback acquisition hence the user experience and the software.

7.4FOURTH PHASE: ADAPTATION ENGINEERING

In the previous steps, personas were used to originate scenarios and these led to a set of goal models which in turn led to define use cases (the diagram and the narrative) on how the interaction between the users and the software for feedback acquisition should take place. In this stage a commonality and variability analysis amongst these interactions should take place. This would result in some form of variability model of feedback acquisition characteristics visible to the user, e.g. a Feature Model (Batory 2005). These models capture the variability and commonality of the features of the different personas and allow configuration and adaptation to take place in one or more of the following styles:

7.4.1 Variability Modelling Using Software Product Lines Techniques

The Software Product Lines (SPL) paradigm refers to the methods, tools and techniques used for developing a collection of similar software systems differing in certain aspects but sharing commonality in others, from a shared set of software entities (Clements and Northrop 2002), E.g. the components of Boeing 757 and 767 have 60% in common and represent a clear example on how different products could share certain set of assets. To express variability in SPL one can use Variability Modelling which is a common way to describe and manage the variants of a system to enable technology to deliver similar software systems in a fast, consistent and comprehensive way with a high quality. Variability Modelling comes in different approaches. Popular ones include:

- **Feature Model**

Feature Model is a technique for describing commonality and variability in SPL. A Feature Model is a tree-structural notation to model feature level commonality and variability in a graphical way to make them more understandable (Lee et al. 2002). Since its first introduction in 1990, feature modelling (Kang et al. 1990) has been the most popular technique to model commonality and variability of software product lines (mostly due to the simple and intuitive notation). In feature modelling, commonalities and variabilities are modelled from the perspective of product features (i.e. stakeholder visible characteristics of products) in a product line that are of stakeholders' concern (Kang and Lee 2013). For example, the inclusion of social factors in a feedback acquisition process may be of interest to Linda, i.e., a service feature, but how this inclusion happens may not be of interest to Linda as long as it is done securely. However, it will be an important concern for the designer of the feedback acquisition when there are alternative ways (e.g. privacy levels) to implement such a service. Feature modelling helps in modelling such variability which will impact the design of the provided service (e.g. right inclusion of the right social factors in feedback acquisition).

- **Orthogonal Variability Model**

Orthogonal Variability Models (Pohl et al. 2005) (OVM) is another notation for modelling software product line variability. The main deference between OVM and feature models is that OVM only document the variabilities present in a product line. However, feature models model both the common aspects of a product line and its variabilities. Therefore, OVM elements are either variation points or variants. Variation points refers to elements that may vary, while variants refers to the deferent possible realizations of a variation point. Another deference with feature models is that OVM elements are not hierarchical, since each variation point is orthogonal to the rest (Simmonds and Bastarrica 2011).

Feedback acquisition relates to SPL in which feedback requests can be differing in certain aspects but sharing commonality in others. This means a variability modelling on feedback acquisition is needed to ensure further adaptivity of feedback requests to specific contexts or individual's needs. Software engineers (enlightened by the previous phases and in collaboration with other stakeholders, e.g. domain experts and business administrators) can use feature modelling (due to its popularity, simplicity and efficiency in modelling variability as previously stated) to tackle such a need. For example, the choice to include certain functionalities or a feature of a feedback request could be decided according to the nature of the domain in general. Privacy-sensitive software, e.g. health related, will maximize privacy issue and thus those features related to social recognition (meant to meet the requirements of people like Jack) should be just optional or subject to confirmation from the users. Some other features could be then decided by the clients, i.e. the software or the product company, based on factors like their need for volume and/or quality. For example, this would result on decisions on the incentives features meant to meet the requirements of people similar to Sara to get just tangible monetary return. Such variability modelling will enable a better customization of feedback request till it arrives to users.

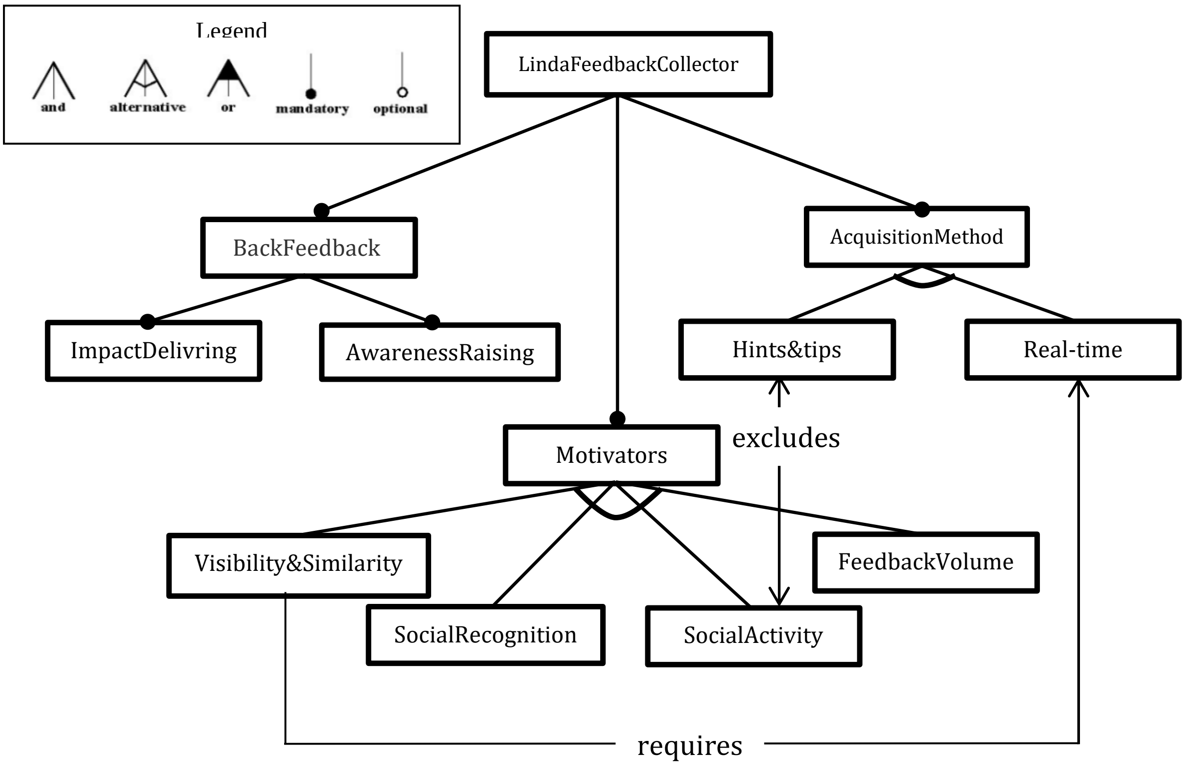


FIGURE 35 LINDA'S FEATURE MODEL FOR FEEDBACK ACQUISITION.

To give a clearer example, Figure 35 shows a feature model of Linda's persona which was developed for CompanyX in the light of the previous phases. However, due to some business related goals and preferences, not all of the features are possible to be implemented as well as some of the features can be extended as appears in Figure 36 to meet CompanyX's goals and preferences. To develop such a feature model, one can refer to Yijum's guidelines for driving a feature model from a goal model (Yu et al. 2008). In addition, business related goals, preferences and context, technical context (Dalpiaz et al. 2012), previously developed scenarios, use cases and their narratives provide a rich source of information that should also be used by software engineers to further consolidate the created feature model.

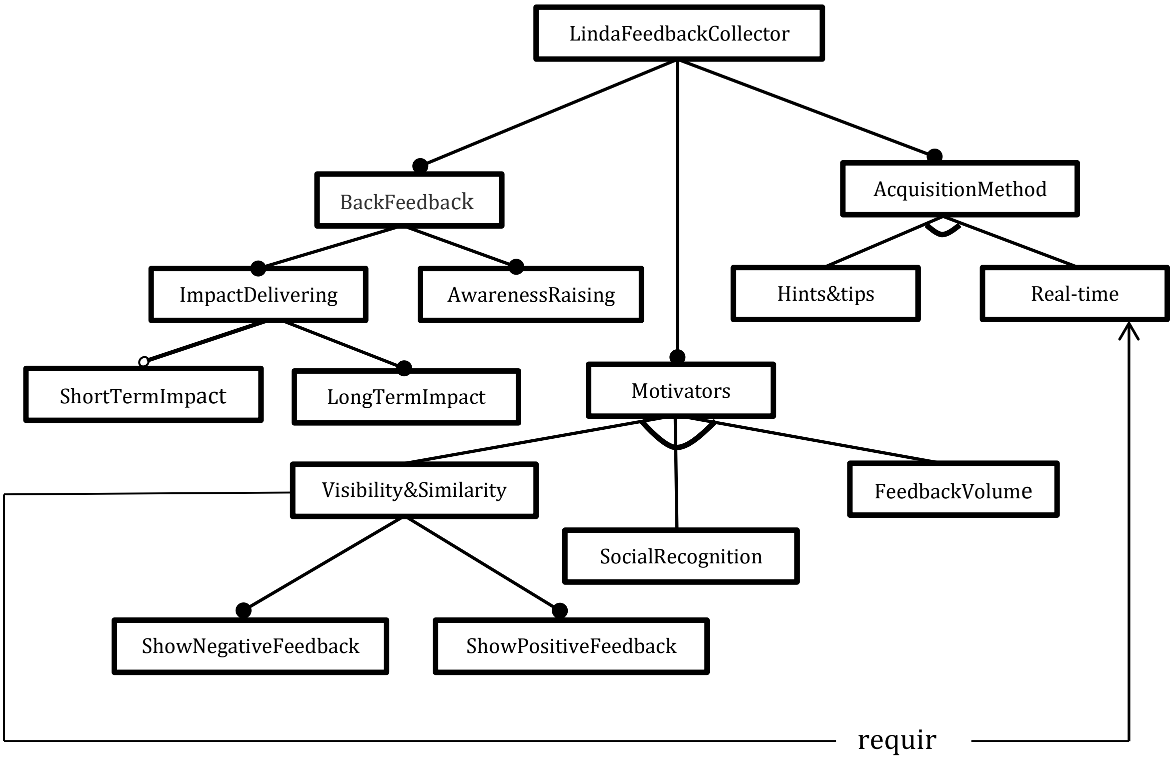


FIGURE 36 LINDA'S CONFIGURED FEATURE MODEL FOR FEEDBACK ACQUISITION.

Resulted feature models can also be presented to users to allow them to customize the features to their own preferences. However, users' customization should be allowed within a space that does not conflict with the business goals and preferences. This will protect users' experience from potential harm if their customizations are not met due to a conflict with a business goal or preference. Table 21 shows the involvement of stakeholders in PAFA phases.

TABLE 21 STAKEHOLDERS INVOLVEMENT IN PAFA METHOD.

Stakeholder	Involvement Phase
Software engineers	Complete involvement in all phases.
End-users	Active involvement in the adaptation engineering through two phases; their *direct input* and c*onfiguration of feature models* to fit individual preferences.
Business client	Active involvement in two stages: • The development stage through *scenarios* for verification and confirmation reasons with software engineers. • The adaptation engineering stage through their c*onfiguration of feature models* to fit their business goals and contexts.

7.4.1.1 GUIDELINES FOR GENERATING FEATURE MODELS FROM GOAL MODELS

Before thinking about generating a feature model from a goal model one should understand the meaning and difference between goals and features. Goals represent stakeholder's intentions thus they are space of intentions which may or may not be fulfilled. On the other hand, Features represent properties of concepts or artefacts (Czarnecki and Eisenecker 2000). Features represent system's functions or properties and should be annotated with conditions describing when to select them (or, optional, alternative conditions).

Since goal models capture the rationale and the quality criteria for achieving a goal, Yijum (Yu et al. 2008) believed that they are potential candidates for the generation of feature models. Feature models are concerned with representing the variability in a system to be developed. Therefore, to generate the feature models, one needs to identify the subset of a goal model that is intended for the system to be developed. The generation can be done as follows:

- The AND decompositions of in a goal model generally correspond to Mandatory features as seen in Figure 37.

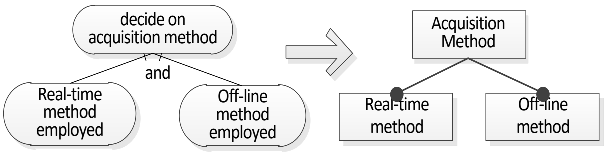


FIGURE 37 AND DECOMPOSITIONS; GOALS TO FEATURES.

- When the selection of subgoals is restricted to some quality criteria (softgoals), they are considered runtime variability and therefore result in mandatory features (see Figure 38).

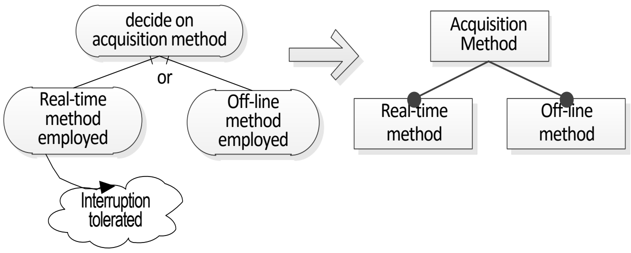


FIGURE 38 OR AND SOFTGOAL TRANSITION TO FEATURES.

- Alternative and Optional features do not have matches in the AND/OR decompositions in the goal models. Therefore, to generate these types of features one need to analyse if some of the OR decompositions are, in fact, XOR which suggest that exactly one subgoal must be achieved (see Figure 39).

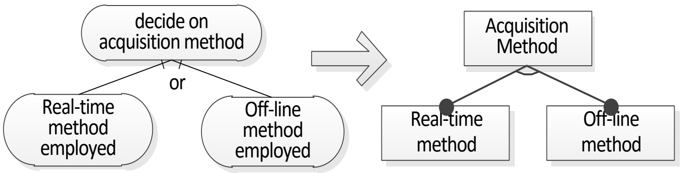


FIGURE 39 XOR GOALS TO FEATURES.

- The normal OR decomposition logically transform into a feature that is refined into a set of OR features (Figure 40).

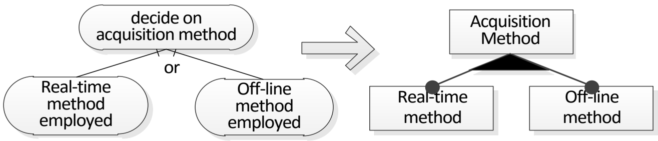


FIGURE 40 OR GOALS TO FEATURES.

- Finally, when a goal is OR-decomposed into at least one non-system subgoal (goals that are delegated to the environment), only the system subgoals will be transformed into optional features (Figure 41).

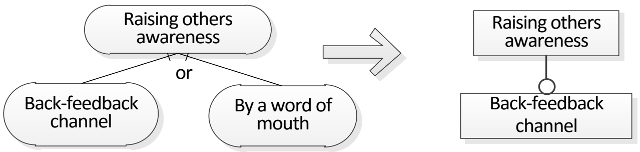


FIGURE 41 OR NON-SYSTEM SUBGOALS TO FEATURES.

In addition, in software product lines many features, are not always applicable or advisable: they are context dependent. A feature may be advisable only in a business or technical context. A business context refers to social, business, or organizational characteristics of the deployment environment (e.g. feedback visibility in health-critical systems). On the other hand, a technical context specifies technical prerequisites for a feature (e.g. platform support) which, If not respected, can hinder the operationalization of a leaf feature. Technical contexts of certain features are typically known and documented in advance by the development team (Dalpiaz et al. 2012) .

To express such context constraints on feature models, Dalpiaz (Dalpiaz et al. 2012) introduced contextual feature models, which explicitly shows the effect of business and/or technical contexts on features and configurations. For example SocialRecognition feature can be implemented when only the business context ***bc*** holds as shown in Figure 42.

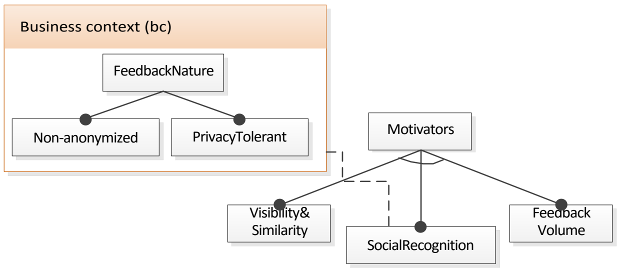


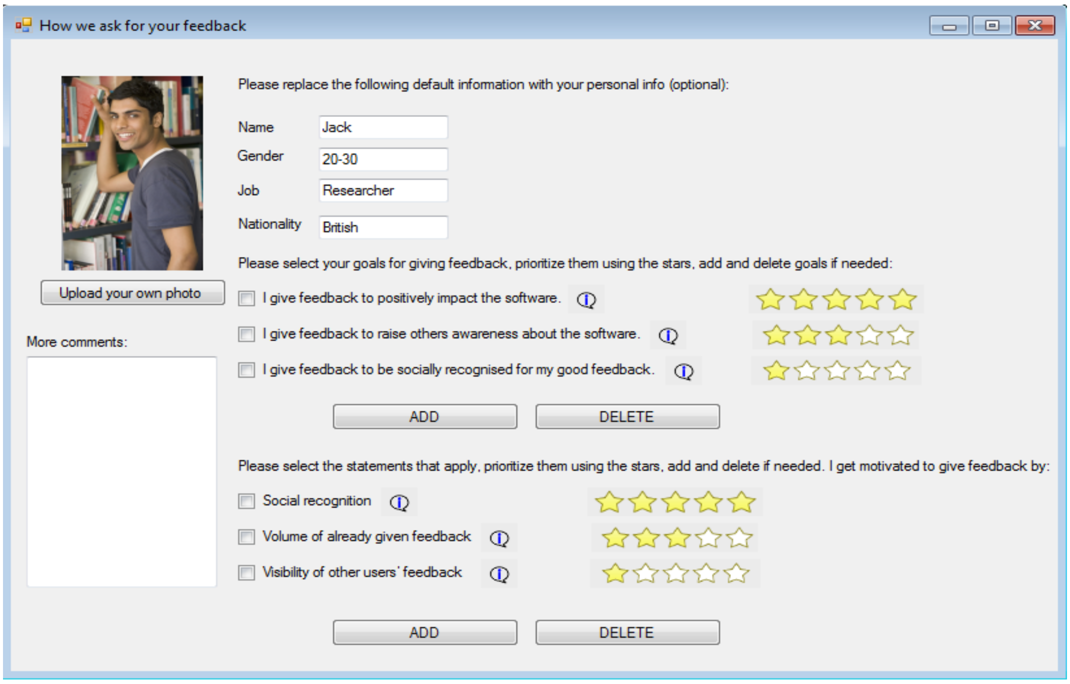
FIGURE 42 EXAMPLE OF A CONTEXTUALIZED FEATURE MODEL.

7.4.2 Users direct input

This can be done by allowing the users to customize the variable design of feedback requests. They can do that through the personas themselves where they can select the persona that reflects them the most and customize it the way they wish to provide feedback. For example, common personas are represented to user Y when installing software X for the first time. User Y then has the option to select the persona that he feels somehow reflects his behaviour to feedback requests mainly by looking at the brief statement describing them. User Y is also able to customize the selected persona through an interactive interface (i.e. prioritise their goals by pulling the highly important one to the top of the list, drag and drop some aspect from another persona to their selected persona, adding their own pictures and names/nicknames to the persona, etc).

This could make users' experience more enjoyable as well as impact the success of the feedback acquisition since it is designed based on trustworthy information coming directly from the users. More lessons on how to design such an interactive approach can be borrowed from HCI approaches to interface design to help software developers employing such an approach. That is, the design of the interaction with users should be itself engineered and tested its usability. Figure 44 shows an initial and potential example of an interactive interface representing Jack persona.

Although this approach could be efficient to cater for the commonality and variability among users, software engineers need to be cautious about how the personas are presented to users. Users might reject a persona that highly represents them just because they are younger or of a different gender, etc. This is a challenge to handle the use of personas in general as a user-friendly customization tool and opens the gate for more research in this area.



How we ask for your feedback

Please replace the following default information with your personal info (optional):

Name	Jack
Gender	20-30
Job	Researcher
Nationality	British

Please select your goals for giving feedback, prioritize them using the stars, add and delete goals if needed:

- □ I give feedback to positively impact the software. ★★★★★
- □ I give feedback to raise others awareness about the software. ★★★☆☆
- □ I give feedback to be socially recognised for my good feedback. ★☆☆☆☆

Please select the statements that apply, prioritize them using the stars, add and delete if needed. I get motivated to give feedback by:

- □ Social recognition ★★★★★
- □ Volume of already given feedback ★★★☆☆
- □ Visibility of other users' feedback ★☆☆☆☆

FIGURE 43 EXAMPLE OF AN INTERACTIVE INTERFACE TO ALLOW USERS TO CUSTOMIZE JACK PERSONA TO THEIR OWN PREFERENCES

7.4.3 Personality questions

As previously discussed (see Chapter 3) people's personalities are different and play a vital role in affecting their relationships, decisions, and life in general. Carl Jung, the founder of analytical psychology (Jung 2014), believed that people are either energized by the outside world (Extraversion) or their own internal world (Introversion). This theory led Isabel Myers, the developer of the Myers-Briggs Type Indicator (MBTI) (Briggs 1976), to investigate the rationale for the personality differences among people to make Carl Jung's theory understandable and useful in impacting different aspects of people's lives.

Myers believed that many problems that involve human interactions and personal choices could be tackled more efficiently with Carl Jung's theory of psychological types in mind (Myers et al. 1985). This eventually led to the development of Myers-Briggs Type Indicator (initially as a questionnaire), which provide structure for understanding similarities, differences and psychological preferences among human beings (Briggs 1976)

As a consequence to Carl's theory and Myers-Briggs Type Indicator, personality questions or tests emerged and have been widely used and available on the internet. These tests are intended to help people inferring their personality type (e.g. Extraversion or Introversion) after answering some personality-related questions.

In the context of PAFA, These personality types related questions can be introduced to users when installing the software for the first time and then their answers can be analysed semi-autonomously to inform if their personality type (Extraversion/Introversion, Sensing/Intuition, Thinking/Feeling or Judging/Perceiving). This then can give an indication about the persona they should be mapped to. An example of potential question to find out if a user an Extraversion or Introversion can be:

- I prefer to work:
  - Alone.
  - In a team.
- I consider myself to be:
  - Social.
  - Private.
- When I have free time:
  - I almost always prefer to do something with others.
  - I usually prefer to do something with others.
  - I sometimes like to be with others but also enjoy spending time by myself.
  - I usually prefer to spend time alone.
  - I almost always prefer to spend time alone.

However, to highly benefit from personality questions, software engineers are highly encouraged to have a good knowledge about models of personality, the appropriate questions to ask and perhaps work in collaboration with experts from psychology or sociology to further maximize the potential benefits. This indeed opens the gate for a further research in this area.

7.4.4 Social Adaptation

Social Adaptation is defined as a system's autonomous ability to analyse users' feedback and choose an alternative configuration which is collectively shown to be the best for satisfying requirements in a certain context (Ali et al. 2012) . The concept of Social Adaptation can be itself applied to customize and adapt feedback requests. Users' given feedback about the software behaviour would then include their feedback on the feedback acquisition request itself. This provides a valuable source of information to discover how a certain acquisition method fits a group of users. This valuable information can be then utilized to adapt the acquisition method. Social adaptation could be advanced by utilizing techniques like collaborative filtering. .

Collaborative Filtering is one of the techniques used by recommender systems to provide recommendations or predictions to the user depending on the opinions of other like-minded users (Terveen and Hill 2001). The motivation of collaborative filtering is the idea that people usually get the best recommendations from other people with similar tastes to themselves.

Collaborative Filtering is also one of the techniques for enabling Social Adaptation and can potentially benefit and help in discovering commonality and variability among users behaviour to feedback acquisition. For example, If X and Y are two like-minded customers of CompanyX represented by Jack (the persona), then a prediction about Y's preferable social factor can be made based on X's preferable social factor. That is, users could share their preferences over the feedback acquisition methods then the software could derive recommendations for them based on what similar/like-minded users chose.

7.5FINAL PHASE: EVOLVEMENT

This phase is indeed an important phase to close the loop, evolve and sharpen the initially created personas enlightened by the acquired information in the previous phases. This thesis recommends software engineers to use this valuable information to enhance the personas, making them more representative and detailed and perhaps add or eliminate some of them. Perhaps some lessons from software evolution such as the corrective, adaptive and perfective maintenance approaches (Lientz and Swanson 1980) can be adopted for refining and evolving the initial personas.

7.6 SUMMARY

The literature contains some approaches on the use of personas to inform the design of software applications such as (Blomquist and Arvola 2002; Guðjónsdóttir and Lindquist 2008; Faily and Fléchais 2014). However, the majority of these approaches do not noticeably employ the power of personas to directly inform the actual implantations of the software and limit their usage to the abstract level as communication tool. This led to a gap and lack of tractability between personas and low level implementations thus the underestimation of personas power.

The novelty and power of PAFA comes from its ability to combine powerful software engineering and user experience methods to systematically inform the design of an adaptive feedback acquisition. This reduces the gap between personas and implementations, increases traceability and shows the real power of personas as a design tool. This can ultimately lead to maximize users' satisfaction and software's success as discussed in Chapters 3 and 4.

In addition, PAFA does not restrict software engineers to adopt all the seven personas in the design of the feedback acquisition. Stakeholders should decide on the personas they need to cater for in their feedback acquisitions. For example, a certain company might decide not to cater for Sara (incentive seeker) in its feedback acquisition to avoid low quality feedback.

In principle, the preferences of a service provider would decide their selection of the personas to support. It may also lead to creating variation of the introduced baseline personas to fit their definition of a relevant user or client. This may require preferences on the costs, the speed of getting feedback, the need for high quality feedback, targeting certain age bands and culture backgrounds, etc. This thesis emphasizes that such decisions are taken alongside PAFA and not only at the initial stage of selecting personas to cater for. This will become clearer when presented in the variability models (i.e. goal and feature model), where the choice of the technique is subject to such preferences.

Additionally and as previously investigated in this thesis, there are also some other key factors that can highly impact the success of the feedback acquisition that should be taken into by software engineers at the design phase. Those factors can include timing for feedback requests, usability and simplicity of feedback requests, device used, interface design, language used, fitness of the design and content, etc (see Chapters 3, 4, Section 4.6). The next chapter report on PAFA evaluation using a case study approach consisting of three real scenarios.

8. CHAPTER 8: PAFA EVALUATION

In a follow up to the previous Chapter where PAFA was introduced, this chapter reports on the method used to evaluate PAFA, the design of the evaluation study, the findings, the lessons learned, the threats to validity and finally the conclusion.

8.1INTRODUCTION

Generally, software engineers form two separate communities; the first one includes those who develop and maintain software systems, and the second one includes those who devise methods and tools that they would like the former group to adopt to build their software systems (Kitchenham et al. 1997). In recent years, one can observe a noticeable increase in the number of software engineering methods and tools offered by the second community. The offered methods and tools usually claim a better ability to improve some characteristic of software, its development, or its maintenance. However, the first community (software developers) seeks methods and tools that will make them more productive and improve the quality of their outcome software (Kitchenham et al. 1995). This inevitably led to questions such as *how to ensure that the offered methods could lead to positive improvement? How to evaluate and validate those methods? How to proceed? Do we do a survey? An experiment? A case study?*. To answer these questions and decide on how to evaluate PAFA, this thesis consults the literature to briefly investigate experimentations in software engineering and discuss the conditions under which each type of investigation is appropriate and then discusses the approach to adopt to evaluate and validate PAFA method.

Although, software engineering method evaluation is a classic topic in software engineering, reviewing the literature it is observable that software engineering tools, methods and techniques dominate the literature but interestingly little research is given on the evaluation and validation of these methods and tools as discussed in (Arthur et al. 1986; Runeson and Höst 2009). In (Basili et al. 1986) software-engineering experimentation was classified as follows:

- **Single-project studies**, which study objects in a single team and a single project.
- **Multi project studies**, which study objects in a single team and a set of projects.
- **Replicated-project studies**, which study objects across a set of teams and a single project.
- **Blocked subject-project studies**, which study objects across a set of teams and a set of projects.

This classification appears in many published software-engineering experiments/studies as a well-established and widely adopted reference which help explaining how these experiments/studies were carried out and understanding how the investigation was done. However, Kitchenham and Pickard (Kitchenham et al. 1995) believed this classification should be advanced and extended to consider and include the formality of the experimental design. They extended the classification as follows:

- If the investigation looks at a single project, they call it a case study, because it is not possible to have a formal experiment without replication.

- If the investigation looks at many projects or a single type of project that is replicated several times, it can be either a case study or a formal experiment. A formal experiment requires appropriate number of replication and subjects and objects chosen at random within the constraints of an experimental design.
- When the investigation looks at many teams and many projects, it could be a formal experiment or a survey, depending on whether the selection of teams and projects was planned or ad- hoc.

Therefore, they suggest that any software-engineering experimentation can be considered as a case study, a formal experiment, or a survey (Kitchenham et al. 1995; Kitchenham 1996; Kitchenham et al. 1997).

The differences between these methods are also reflected in their scale. Formal experiments are carefully controlled and usually small in scale whereas case studies often look at what is happening on a typical project which gives it a larger scale. The small scale of formal experiments could result in some issues when the investigator/researcher tries to increase the scale from the laboratory to a real project. Therefore, case studies are highly important for industrial evaluation of software engineering methods and tools since they can minimize scale-up issues (Kitchenham et al. 1995; Runeson and Höst 2009). On the other hand, surveys try to investigate what is happening broadly over large groups of projects which gives it the largest scale. The selection of any of the previous methods is highly important because the design, analysis techniques and conclusions they yield differ (Kitchenham et al. 1995).

The choice of the methods is dependent on several factors such as the size and nature of the project. If the investigator is trying to choose one of several competing methods, they might adopt a formal experiment or a case study. However, if the investigator is trying to establish a pilot method to assess the impact it could make (which is the case for PAFA), they will probably select to do a case study (Kitchenham et al. 1995; Runeson and Höst 2009). Whereas, a survey is helpful to document the benefits after a change has already been implemented across a large number of projects the change (Kitchenham et al. 1995).

A case study is usually preferable to a formal experiment or surveys when the impact of the investigated method or tool can be assessed only at a high level. This happens when many detailed changes throughout the investigation process take place which can be challenging to assess and trace (Kitchenham et al. 1995; Kitchenham 1996). This holds true for PAFA since it includes several detailed components for engineering feedback acquisition which makes it hard to assess or trace the impact of the method at the component level. This suggests that the impact of adopting such a method can be only sensed at a high level (e.g. the increase in the response rate to feedback requests after adopting PAFA). In addition, the ultimate impact of PAFA cannot be instantly assessed since PAFA requires a period of time to evolve (e.g. the evolvement of personas through the life time of the software) and adapt to reach its desired impact. Additionally, the time limit for this study and available resources do not give the luxury to adopt any other approach.

Although case studies cannot achieve the scientific rigor of formal experiments, they can provide sufficient information to help investigators judge if a specific method will benefit their organization or project. Therefore, it is often found suitable in software engineering research (Höst and Runeson 2007; Wohlin et al. 2012). This led to the question; Why Case Studies are suitable in Software Engineering?

Software engineering involves the process of development, operation, and maintenance of software and its related artefacts. Software engineering research is mostly concerned with investigating how the development, operation and maintenance are conducted by software engineers and other stakeholders in certain conditions. Software development is carried out by individuals, groups and organizations, thus social and political questions are of importance for this development. That is, software engineering is a multidisciplinary domain that involves domains where case studies normally are adopted and conducted. This led to suggest that many research questions in software engineering are suitable for case study research (Runeson and Höst 2009; Wohlin et al. 2012). In addition, experimentation in software engineering has shown that there are many factors impacting on the outcome of a software engineering activity (e.g. the contextual factors) (Shull et al. 2002). This appears clearly when for example a researcher tries to replicate studies. Case studies offer a flexible approach which does not need a strict boundary between the studied object and its environment/context (Wohlin et al. 2012). Enlightened by the previous benefits of case studies, the available resources, time and nature of research, this thesis adopts case study as a method to evaluate and validate PAFA method.

8.2 STUDY DESIGN

The design of case studies is almost the same for any kind of empirical study (e.g. the designs proposed by (Kitchenham et al. 2002; Wohlin et al. 2012)). However, as case study has a more flexible design strategy, it could have a significant amount of iteration over its steps if needed (Andersson and Runeson 2007). For example, if insufficient data was collected for the analysis, more data collection may be planned etc. The main restriction to its flexibility is that the case study should have a specific aim identified from the beginning (Runeson and Höst 2009). Generally, a typical case study design consists of the following phases (Höst and Runeson 2007; Yin 2013) which are adopted in this study:

- Case study design: aim/ objectives are defined and the case study is planned.
- Data collection is planned: procedures and protocols for data collection are defined.
- Collecting evidence: executing the planned data collection on the studied case.
- Analysis of collected data.
- Reporting.

8.2.1 AIM AND CASE STUDY SCENARIOS

The aim of conducting this evaluation case study is *to assess to what extent PAFA method provides a systematic and effective engineering process for the design of an adaptive feedback acquisition.* In particular, this evaluation is meant to find out how PAFA is effective and systematic in the way it helps software engineers identify and cater for users' different requirements and behaviours with regard to feedback acquisition (as previously investigated in this thesis). According to the previous studies (see chapter 4, 5 and 6) there is a high diversity of users' behaviour to feedback acquisition and a lack of requirement engineering approaches to cater for these variations. Many of vital users' requirements when it comes to feedback acquisitions (e.g. social factors) are overlooked by current approaches, if there are any!

To conduct this evaluation, formative assessment principles are followed in which the yielded information from the assessment/evaluation will be used to reflect on PAFA method for improvement. Formative evaluation involves the collection of data and information during the development process of a certain instrument/tool that can be used to improve its effectiveness. Formative means that the instrument/tool is in its formative or early stages, and evaluation refers to the process of gathering data to determine the strengths and weaknesses of the instrument/tool. Thus, formative evaluation is "a judgment of the strengths and weaknesses of instruction in its developing stages, for the purposes of revising the instruction to improve its effectiveness and appeal" (Tessmer 1993).

This indeed fits the nature of the evaluation's aim and suggests formative assessment as an appropriate approach to direct the evaluation process of PAFA method. In the following, a case study that consists of three real scenarios is provided. The provided scenarios are meant to help the participants to work on PAFA and showcase how it can be applied in practice. The resulted outcome is hoped to positively impact the provided scenarios (e.g. introduce adaptivity to the given scenarios).

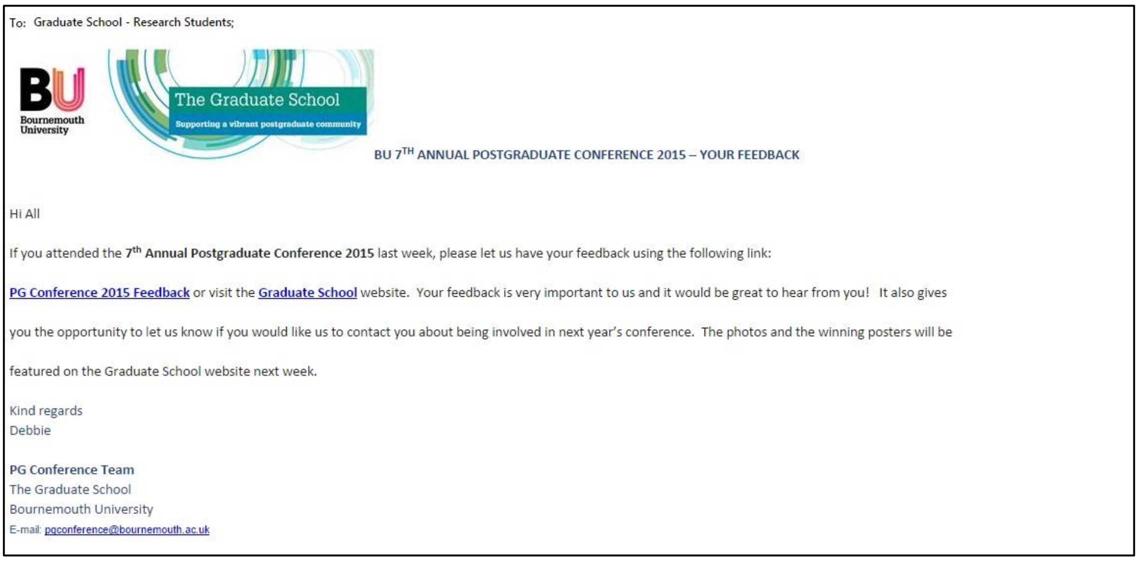
8.2.1.1 CASE STUDY SCENARIO 1: FEEDBACK SYSTEM AT THE GRADUATE SCHOOL, BOURNEMOUTH UNIVERSITY (BU)

The scenario provided below is a result of an interview conducted with a member from Graduate School who has a direct involvement with the feedback collection process at the school. The main aim of the interview was to understand how the feedback system at the school works.

The mission of the Graduate school at Bournemouth University (BU) is to provide support to all BU postgraduate students, whether academically or by enhancing the student experience (e.g. social events). Therefore, they are normally encouraged by the Higher Education Academy (HEA) in the UK to always hear from students about their experience, needs and concerns to inform future improvements. This is done by conducting online surveys using designed by an online tool called Bristol Online Survey. The survey design and questions asked are normally informed by the HEA to ensure appropriate and understandable questions are being introduced taking into account the targeted population. For example, wording and language used for international students are carefully designed to ensure they are culturally appropriate and understandable by students who do not speak English as their first language.

From time to time, postgraduate students receive emails from the Graduate School inviting them to take part in online surveys. Typically, the surveys are designed to hear students' suggestions and complains or to measure their satisfaction about the services provided by the school. The survey invitation emails contain the school logo, a short introduction about the survey purpose and a link to the survey form to fill in (see Figure 45). The surveys normally run between March and May in which students would potentially have some free time to take part in the surveys. After allowing sometime since the first invitation to the survey, students are then sent personalised reminders (e.g. including the students names in the invitation) to consider taking part in the survey if they have not already done. Those reminders are believed by the interviewed member to increase the response rate to the survey requests. The school also provide incentives (e.g. Amazon vouchers, prize draws, etc) for time consuming surveys in order to maintain a good response rate.

However, the survey/feedback requests provided by the graduate school are restrictive in their design. They do not enjoy a high level of adaptivity to the students' different types of behaviours and preferences with regards to feedback provision. For example, feedback is being asked by emails only whereas some students might prefer to be asked for feedback using a different method such as pop ups or not to be asked for feedback at all. The interviewed member commented in this regard *"It is still quiet restrictive and inflexible the way we did it by email"*. Another example could be the lack of social elements in the feedback requests which can improve students/users response rate and experience such as the visibility of others feedback. A further example, no back-feedback channel is introduced to students who seek to see the impact of their feedback on the services provided by the school which can harm their experience and trust their voice is being heard. This lack of adaptivity could highly harm customers' satisfaction and experience which can then harm the Graduate School reputation among students in the long-term.



To: Graduate School - Research Students;

BU Bournemouth University

The Graduate School

Supporting a vibrant postgraduate community

BU 7TH ANNUAL POSTGRADUATE CONFERENCE 2015 – YOUR FEEDBACK

Hi All

If you attended the 7th **Annual Postgraduate Conference 2015** last week, please let us have your feedback using the following link:

PG Conference 2015 Feedback or visit the Graduate School website. Your feedback is very important to us and it would be great to hear from you! It also gives you the opportunity to let us know if you would like us to contact you about being involved in next year's conference. The photos and the winning posters will be featured on the Graduate School website next week.

Kind regards
Debbie

PG Conference Team
The Graduate School
Bournemouth University
E-mail: pgconference@bournemouth.ac.uk

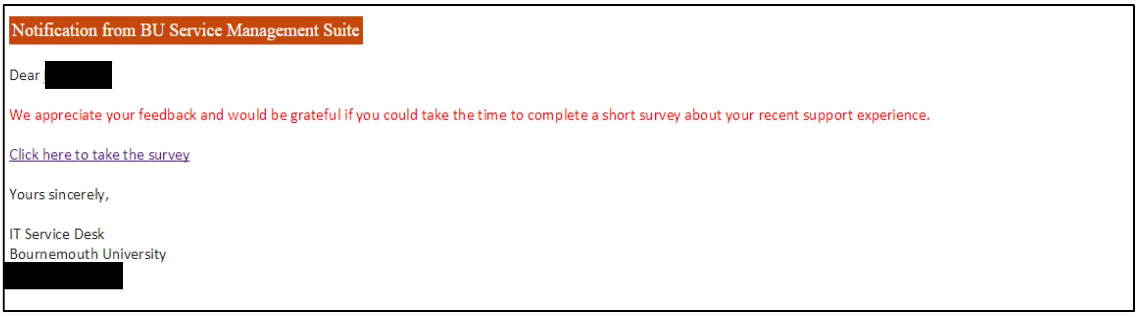
FIGURE 44 A SNAPSHOT EXAMPLE OF ONE THE GRADUATE SCHOOL FEEDBACK REQUEST BY EMAIL.

8.2.1.2 CASE STUDY SCENARIO 2: IT SERVICE DESK FEEDBACK SYSTEM AT BOURNEMOUTH UNIVERSITY (BU).

The scenario provided below is a result of an interview conducted with a member from IT service desk who is in a direct involvement with the feedback collection process at the service desk. The main aim of the interview was to understand how the feedback system at the IT service desk works.

The IT Services at BU is responsible for providing academic and administrative computing, networking and applications, media services and support at BU. It encompasses all activities which use information technology in order to deliver a service, provides or processes information and includes support for users. They are highly concerned about the quality of the service they provide and the satisfaction of their users. Therefore, they always make sure the voice of users is always heard to inform future improvements. When an IT incident or request that was created or reported by the users (student or staff member) is closed, a feedback email/request is automatically sent out to the user. The survey/feedback invitation email contains the IT service desk logo, invitation to the user to take part in a short survey on the quality of the service provided regarding their recent IT incident/request and a link to the survey form to fill (see Figures 46 and 47).

However, the feedback requests/emails sent by the IT service desk are designed for all costumers universally. They are static and restrictive in their design and do not enjoy a high level of adaptivity to the users' different types of behaviours and preferences with regards to feedback provision. For example, feedback is being asked by emails only whereas some students or staff might prefer to be asked for feedback using a different method such as pop ups or not to be asked for feedback at all. Another example could be the missing social elements in the feedback requests which can improve students/users response rate and experience such as the visibility of others feedback. An additional example, no back-feedback channel is introduced to users who seek to see the impact of their feedback on the services provided by the IT service desk which can harm their experience and trust their voice is being heard. This resulted in the IT service desk suffers from a low response rate as believed by the interviewed member. In addition, this lack of adaptivity could harm users' satisfaction and experience which could then negatively impact the IT service reputation among students in the long-term.



Notification from BU Service Management Suite

Dear [redacted]

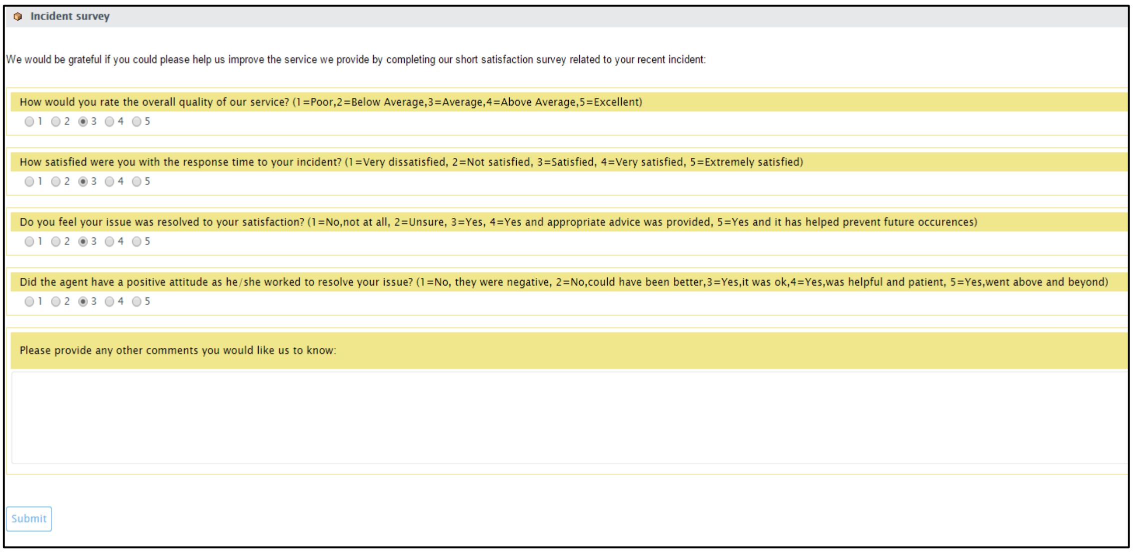
We appreciate your feedback and would be grateful if you could take the time to complete a short survey about your recent support experience.

Click here to take the survey

Yours sincerely,

IT Service Desk
Bournemouth University
[redacted]

FIGURE 45 A SNAPSHOT OF THE FEEDBACK REQUEST SENT BY THE IT HELP DESK.



Incident survey

We would be grateful if you could please help us improve the service we provide by completing our short satisfaction survey related to your recent incident:

How would you rate the overall quality of our service? (1=Poor,2=Below Average,3=Average,4=Above Average,5=Excellent)
○ 1 ○ 2 ● 3 ○ 4 ○ 5

How satisfied were you with the response time to your incident? (1=Very dissatisfied, 2=Not satisfied, 3=Satisfied, 4=Very satisfied, 5=Extremely satisfied)
○ 1 ○ 2 ● 3 ○ 4 ○ 5

Do you feel your issue was resolved to your satisfaction? (1=No,not at all, 2=Unsure, 3=Yes, 4=Yes and appropriate advice was provided, 5=Yes and it has helped prevent future occurences)
○ 1 ○ 2 ● 3 ○ 4 ○ 5

Did the agent have a positive attitude as he/she worked to resolve your issue? (1=No, they were negative, 2=No,could have been better,3=Yes,it was ok,4=Yes,was helpful and patient, 5=Yes,went above and beyond)
○ 1 ○ 2 ● 3 ○ 4 ○ 5

Please provide any other comments you would like us to know:

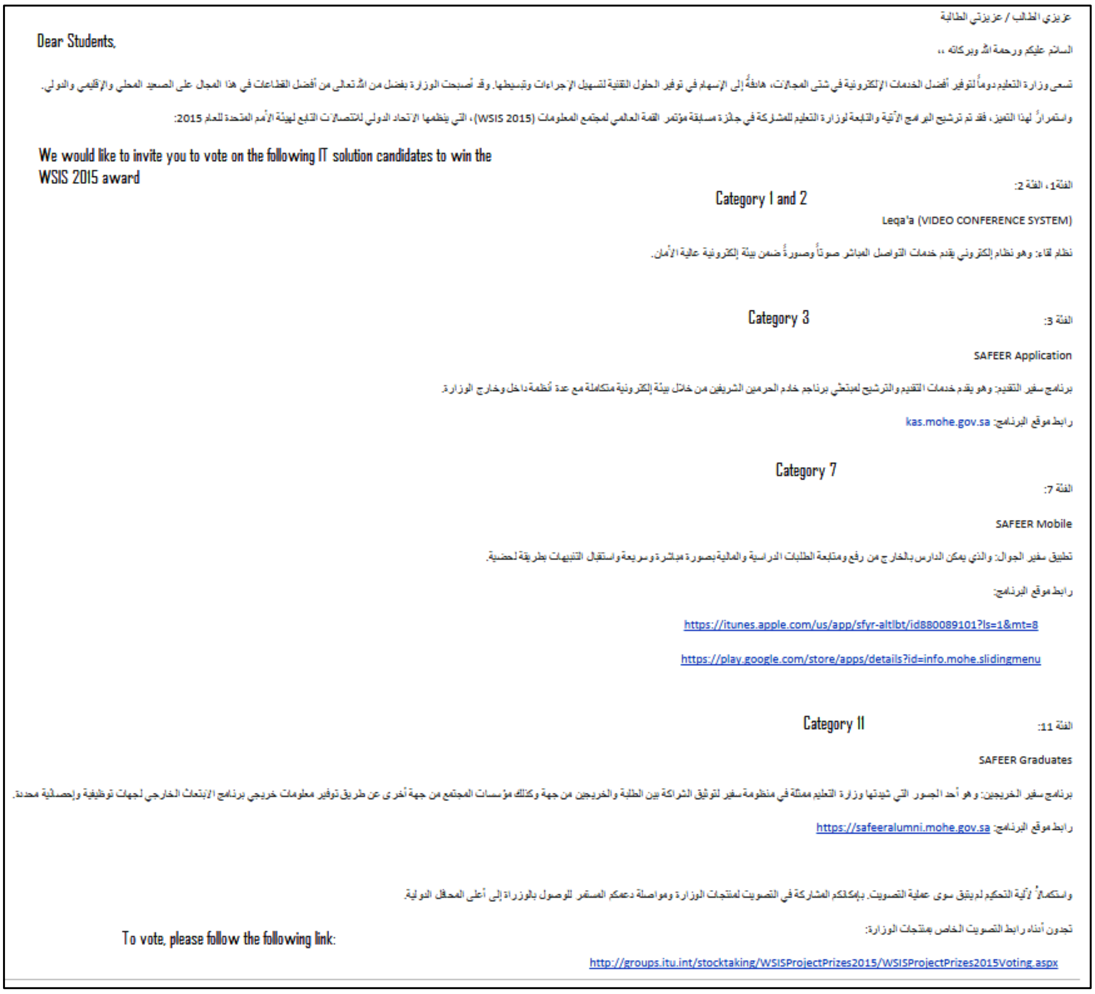
FIGURE 46 AN EXAMPLE OF THE IT FEEDBACK FORM.

8.2.1.3 *CASE STUDY SCENARIO 3: FEEDBACK SYSTEM OF THE STUDENT PORTAL FOR SAUDI STUDENTS IN THE UK (SAFEER)*

The scenario provided below is a result of an interview conducted with a member from the IT department at the Saudi Cultural Bureau in London who is in a direct involvement with the feedback collection process at Safeer. The main aim of the interview was to understand how the feedback system at Safeer works.

The mission of Safeer is to provide IT services to Saudi students studying in the UK to help them managing their scholarships online (e.g. requesting letters, requesting financial sponsorships, etc). The portal also offers several services to students, such as Q&A, Recommended Universities, etc. From time to time, students receive real-time feedback pop ups once they entered Safeer. Typically, the pop ups ask the students to rate the provided IT service and comment, complain or suggest improvement if needed. In addition, from time to time the students receive feedback emails from Safeer asking them to vote on the services provided by the portal (see Figure 48).

However, the survey/feedback requests provided by Safeer are restrictive in their design. They do not enjoy a high level of adaptivity to the students' different types of behaviours and preferences with regards to feedback provision. For example, feedback is being asked offline (emails) and real-time (pop ups) without catering for some students experience and preferences (e.g. student who prefer emails might find pop ups intrusive). In addition, some students might prefer not to be asked for feedback at all. Another example could be the missing social elements in the feedback requests which can improve students/users response rate and experience such as the visibility of others feedback. A further example is the lack of a back-feedback channel that could be important to some students who seek to see the impact of their feedback on the services provided. This can harm their experience and make them doubtful if their voice is being heard and acted upon by the service provider. This lack of adaptivity could highly harm students' satisfaction and experience which can then harm Safeer's reputation among students in the long-term.



عزيزي الطالب / عزيزتي الطالبة

Dear Students,

السلام عليكم ورحمة الله وبركاته ،،

تسعى وزارة التعليم دوماً لتوفير أفضل الخدمات الإلكترونية في شتى المجالات، هادفةً إلى الإسهام في توفير الحلول التقنية لتسهيل الإجراءات وتبسيطها. وقد أصبحت الوزارة بفضل من الله تعالى من أفضل القطاعات في هذا المجال على الصعيد المحلي والإقليمي والدولي.

واستمراراً لهذا التميز، فقد تم ترشيح البرامج الآتية والتابعة لوزارة التعليم للمشاركة في جائزة مسابقة مؤتمر القمة العالمي لمجتمع المعلومات (WSIS 2015)، التي ينظمها الاتحاد الدولي للاتصالات التابع لهيئة الأمم المتحدة للعام 2015:

We would like to invite you to vote on the following IT solution candidates to win the WSIS 2015 award

الفئة1، الفئة 2:

Category 1 and 2

Leqa'a (VIDEO CONFERENCE SYSTEM)

نظام لقاء: وهو نظام إلكتروني يقدم خدمات التواصل المباشر صوتاً وصورةً ضمن بيئة إلكترونية عالية الأمان.

Category 3

الفئة 3:

SAFEER Application

برنامج سفير التقديم: وهو يقدم خدمات التقديم والترشيح لمبتعثي برنامج خادم الحرمين الشريفين من خلال بيئة إلكترونية متكاملة مع عدة أنظمة داخل وخارج الوزارة.

رابط موقع البرنامج: kas.mohe.gov.sa

Category 7

الفئة 7:

SAFEER Mobile

تطبيق سفير الجوال: والذي يمكن الدارس بالخارج من رفع ومتابعة الطلبات الدراسية والمالية بصورة مباشرة وسريعة واستقبال التنبيهات بطريقة لحضية.

رابط موقع البرنامج:

https://itunes.apple.com/us/app/sfyr-altlbt/id880089101?ls=1&mt=8

https://play.google.com/store/apps/details?id=info.mohe.slidingmenu

Category 11

الفئة 11:

SAFEER Graduates

برنامج سفير الخريجين: وهو أحد الجسور التي شيدتها وزارة التعليم ممثلة في منظومة سفير لتوثيق الشراكة بين الطلبة والخريجين من جهة وكذلك مؤسسات المجتمع من جهة أخرى عن طريق توفير معلومات خريجي برنامج الابتعاث الخارجي لجهات توظيفية وإحصائية محددة.

رابط موقع البرنامج: https://safeeralumni.mohe.gov.sa

واستكمالاً لآلية التحكيم لم يتبق سوى عملية التصويت. بإمكانكم المشاركة في التصويت لمنتجات الوزارة ومواصلة دعمكم المستمر للوصول بالوزارة إلى أعلى المحافل الدولية.

تجدون أدناه رابط التصويت الخاص بمنتجات الوزارة:

To vote, please follow the following link:

http://groups.itu.int/stocktaking/WSISProjectPrizes2015/WSISProjectPrizes2015Voting.aspx

FIGURE 47 AN EXAMPLE OF ONE OF SAFEER'S FEEDBACK EMAILS.

8.3DATA COLLECTION AND PROCEDURE

Typically, elements of other empirical research methods such as, surveys, interviews and/or observations are used for data collection in case studies. The data collected in an empirical study can be quantitative, qualitative or hybrid (Runeson and Höst 2009). However, a combination of qualitative and quantitative data usually gives better understanding of the studied phenomenon and consolidates its outcome (Seaman 1999; Runeson and Höst 2009), e.g. what is sometimes called “mixed methods” (Robson 2002). This study adopts a mixed method approach in which surveys (quantitative) and observation (qualitative) are employed to increase the credibility of the study as well as provide better understanding of PAFA evaluation. The evaluation process was conducted as follows:

- The participants were grouped in one room and given a briefing 15 minutes presentation about the purpose of the study, PAFA, its aim and how to use it.
- Then a copy of the following documents were handed out to the participants (beside sending them in advance to the participants for familiarization purposes)
    - Personas for Feedback Acquisition. This contains the 7 personas that were developed for feedback acquisitions which will be used during the study (see Chapter 6).
    - PAFA Method Manual. This gives a description about PAFA method and how to use it (see Chapter 7).
    - PAFA Evaluation Protocol. This introduces the study procedure to the participants, the tasks and the questions to be asked (see Appendix 6).
    - Participants cconsent form and information sheet (see Appendix 7).
- The study was conducted in three different locations (see participants selection, Section 8.4) with three different groups of software engineers and lasted for an average of 3 to 4 hours.
- An individual from each group acted as a client representing one of the previous case studies scenarios (e.g. acting as a representative of the Graduate School at Bournemouth University). The representative also had the option to assign one persona to each member of the software engineers. Then each individual used PAFA manual to engineer a feedback acquisition that fits their assigned persona. In addition, some steps or phases of PAFA manual were optional to do (e.g. using personality questions, Social Adaptation or users direct input to engineer adaptations). This can reduce the effort and time required by the participants to conduct the study and increase their willingness to take part in it. In addition, having the participants working individually on PAFA could minimize problems that could emerge in group dynamics (e.g. group influence on an individual, interaction and communication patterns). Furthermore, having each participant working on PAFA individually (e.g. not having each step done by an individual and passed over to another individual to complete the next step) can minimize conflicts that could emerge between the team members (e.g. an individual might find the

scenarios developed by his colleague are not helpful for him to develop a goal model and lack lots of information).

- Software engineers were encouraged to keep an active engagement with the client throughout the study for verification and consultation purposes (see Appendix 6).
- During the evaluation process, software engineers were asked to take a survey on each component of PAFA as well as PAFA as a whole (see next section).
- Additionally, the participants were continuously observed by the study facilitator during the evaluation process to further assess how PAFA is being used, ask/answer questions and offer clarifications when needed.
- A follow up 10 to15 minutes discussion session was then held to discuss the participants' experience with PAFA. This gave the participants the chance to freely express their feelings/experience about PAFA which added clearer, collective and richer judgments on PAFA and positively impact the assessment process.
- A confirmatory study with real clients from both the IT Desk and the Graduate school at Bournemouth University was conducted to reflect on, enhance and confirm the findings of PAFA evaluation.

8.3.1 QUESTIONNAIRE

Self-completion questionnaires were used as a data collection method which was printed out and given to participants complemented with a booklet to allow for an extra space for participants' answers if needed. The questionnaire contained both types of questions: open-ended questions and close-ended questions (e.g. ratings). The open-ended questions were used to minimize the risk of missing significant information and to give participants a space to include information they felt was relevant to PAFA method evaluation. Closed questions were employed to stimulate/trigger participants thinking and put less effort on them when answering the questionnaire (Leung 2001).The questions were developed around the goals of this study that can be seen in Table 22.

The questionnaire included questions that were asked about PAFA at the component level (i.e. Persona to Scenarios, Scenarios to Goal Models, Goal Model to Use-cases and Adaptation Engineering (see Chapter 7)) and another set of questions that were asked about PAFA as a whole (see Appendix 6). The questionnaire also introduced the aim of the study to the participants. The participants were also informed about what is expected from them and how the results of the questionnaires will be used through an information sheet that was enclosed with the questionnaire. The data collection took place between July and September, 2015. The questionnaire and data gathering went through the ethics approval process by Bournemouth University.

8.3.2 GQM

The Goal Question Metric approach (GQM approach) is a well-established approach that was developed to address the needs for a goal-oriented approach that can support the measurement of processes and products in the domain of software engineering. The GQM approach or paradigm supports a top-down approach for identifying the goals behind measuring software processes and products, and then using those goals to decide what to measure (choosing metrics) (Basili 1992; Differding et al. 1996; Van Solingen and Berghout 1999). The GQM paradigm also supports a bottom-up approach to interpreting or reasoning about data based on the previously defined goals and questions.

In general, the GQM approach offers a valuable help for planning and performing the analysis and evaluation of processes and products from all activities and phases of a software engineering project. In a case-study, the definition of what data to collect should also be based on a goal-oriented measurement technique, such as the Goal Question Metric method (GQM) (Van Solingen and Berghout 1999; Runeson and Höst 2009). Over the years, the GQM approach has been applied by software engineers and practitioners in different contexts with good success (Basili 1992; Differding et al. 1996; Stoddard II 1999; Van Solingen and Berghout 1999). GQM approach defines three levels of measurements as follows:

- **Conceptual level (Goal):** at this level the goal of the measurement is defined for a variety of reasons, with respect to various models of quality, from various points of view and relative to a particular environment.
- **Operational level (Question):** at this level a set of questions is used to define the assessment of a certain goal is going to be done.
- **Quantitative level (Metric):** at this level a set of metrics is identified and associated with every question in order to answer it in a measurable way.

As previously stated, the GQM is a well-established approach that relies on the idea that measurement should be goal-oriented; e.g. all data collection should be based on a rationale that is explicitly documented. This approach offers several advantages such as (Basili 1992; Differding et al. 1996; Stoddard II 1999; Van Solingen and Berghout 1999):

- It helps in the identification of useful and relevant metrics,
- it keeps the focus on why the metrics are being collected,
- and the goals provide a rational and context for the analysis and interpretation of the collected data.

Encouraged by these benefits, this study employs the concept of GQM as an underpinning for its evaluation as appears in Figure 49 in which the aim of the case study constructs the goal whereas the actual procedure is the goal related questions and metrics. This will help identifying appropriate metrics to evaluate, analyse and interpret PAFA more rationally which will ultimately allow for a better evaluation process. In addition, this study enriches the quantitative nature of the GQM with a qualitative facet to maximize the power and coverage of the evaluation process of PAFA (see next section).

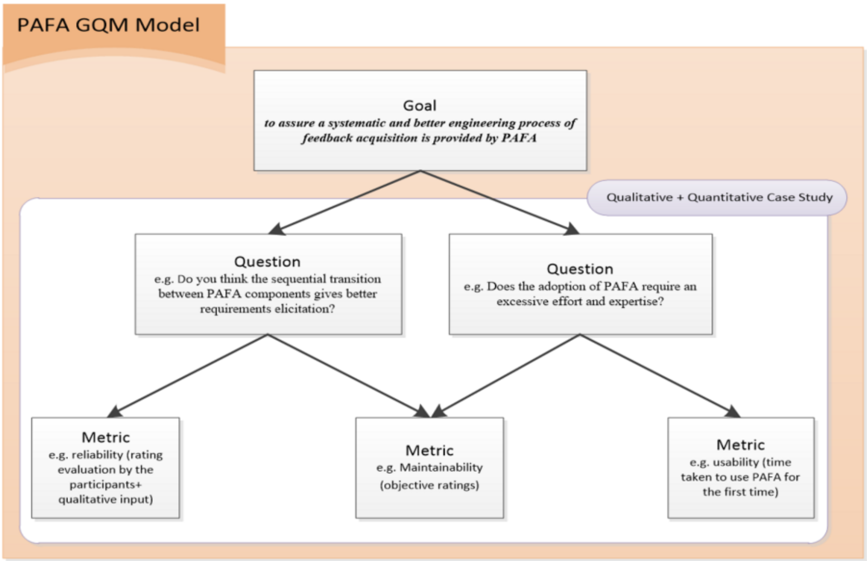

PAFA GQM Model
Goal
to assure a systematic and better engineering process of feedback acquisition is provided by PAFA
Qualitative + Quantitative Case Study
Question
e.g. Do you think the sequential transition between PAFA components gives better requirements elicitation?
Question
e.g. Does the adoption of PAFA require an excessive effort and expertise?
Metric
e.g. reliability (rating evaluation by the participants+ qualitative input)
Metric
e.g. Maintainability (objective ratings)
Metric
e.g. usability (time taken to use PAFA for the first time)

FIGURE 48 GQM MODEL FOR PAFA EVALUATION.

8.3.3 GOAL, QUESTIONS AND METRICS

A broad investigation of the literature of software engineering, software quality and usability (Wirth 1971; Liskov 1972; Jackson 1975; Parnas 1976; Scott 1978; Clements 1981; Gaffney Jr 1981; Arthur et al. 1986; Kitchenham 1996; Jung et al. 2004) led to the identification of several quality measures or metrics that are commonly recognized in software engineering methodologies. These measures are; *Understandability, Completeness, Reliability, Efficiency, Integrity, Maintainability, Integrability, Portability, Testability, Usability, Maturity, Flexibility, Reusability, Interoperability, Coupling and Cohesiveness.* This thesis adopts the applicable measures/metrics to the context and available resources of this study in relation to PAFA evaluation and validation. It also gives an example of goals, potential questions and metrics for each question as shown in Table 22 (see Appendix 6 for the full list of questions).

The author would also like to notify the reader that some of the metrics can be informed qualitatively based on the qualitative input by the participants (e.g. question for ***Goal 3*** in Table 22). In addition, all of the metrics appear in Table 22 indicate perceived measures (i.e. the participants perception or point of view about PAFA and its components). However, only one metric related to efficiency indicates an explicit/objective measure which is the time taken to use PAFA by the participants for the first time. This measure is independent of the participants own perception or point of view about PAFA.

TABLE 22 MEASURES ADOPTED TO EVALUATE PAFA WITH EXAMPLE QUESTIONS.

Goal 1	
To measure how PAFA artefacts can be easily updated with any changes or enhancements.	
Example Questions	**Metric**
E.g. is a change to one component of PAFA likely to require a change to the other components? (E.g. if a developer goes back during the development process to one of the components and update it with more information about the users (e.g. more scenarios are introduces), would this require him/her to go again through the following components to update them too?	**Maintainability (**qualitative input + rating evaluation)
Goal 2	
To investigate the extent in which the components of PAFA cover the holistic design of feedback acquisition	
Example Questions	**Metric**
E.g. Do you think more components need to be added to PAFA to further improve it?	**Completeness** (number of needed/missing components + rating by participants + qualitative input)
Goal 3	
To measure the extent to which PAFA can meet its claims with regard to catering for users' different requirements and preferences to feedback acquisition	
Example Questions	**Metric**
E.g. Do you think the sequential transition between PAFA components provides an effective way for identifying various requirements of users' to feedback acquisition? Please elaborate.	**Reliability** (rating evaluation by the participants + qualitative input)
Goal 4	
To investigate the amount of effort, resources and expertise required by PAFA if adopted.	
Example Questions	**Metric**
E.g. Does the adoption of PAFA require an excessive effort and expertise?	**Efficiency**(time taken to use PAFA for the first time + resources/expertise needed (qualitative input))
Goal 5	
To investigate the extent in which PAFA can be adopted in other application domains.	
Example Questions	**Metric**
E.g. Does PAFA show applicability to be adopted in different application domains such health critical systems? Please elaborate.	**Portability (**rating by participants + qualitative input + qualitative input)
Goal 6	
To measure the ease to learn, interpret and use PAFA.	
Example Questions	**Metric**
E.g. On a 1 to 5 scale, is enough information is given to show how to use each component of PAFA?	**Usability (**rating by participants + qualitative input)
Goal 7	
To investigate the extent to which enough literature is readily available about PAFA's components.	
Example Questions	**Metric**
E.g. Rate the widespread use of each component of PAFA by software engineers?	**Maturity (**rating by participants + qualitative input)
Goal 8	

To measure to what extent PAFA components integrate and work correctly together.	
Example Questions	**Metric**
E.g. Does each phase in PAFA provide a solid foundation towards the following phase?	**Integrability** **(**rating by participants + qualitative input)
Note; a question can be applicable to multiple metrics as indicated in Figure 49 and not restricted to only one metric.	

8.3.4 OBSERVATION

Observation is a key qualitative research method in academic disciplines such as Anthropology, Sociology, Education, Development Studies, Psychology and Management Studies. It offers a powerful way for capturing and understating people/participant's behaviour and attitude for a certain aim by observing and recording their activities from a distance that does notaffect the way they behave (olivia 2012).

In this study, observation method stands out as a powerful candidate to be adopted in the evaluation process of PAFA method. It allows to observe and reflect on how the participants are using PAFA, where they find it challenging, where they find it easy and enjoyable, etc. It also offers more freedom to both the observer and the participants to probe, ask and answer questions during the evaluation process and offer clarifications if needed. This indeed consolidates the evaluation process of PAFA in which quantitative metrics/measures will be enriched with qualitative interpretations or when some aspects of PAFA cannot be quantitatively measured. This will ultimately result in a more solid and helpful conclusions that can positively impact the final product of PAFA.

However, the question is how does one go about conducting observation? In response to this question (Whyte 1979) noted that, while there is no one way that is best for conducting research using participant observation, the most effective work is done by researchers who view informants as collaborators. Generally, conducting observations involves a number of activities and considerations for the researcher such as; selecting the participants, establishing the rapport (e.g. getting to know the participants), the processes for conducting observations, keeping field notes, and writing up one's findings (Kawulich 2005). In (Oswald and Schoepfle 1987) three types of processes for conducting observations are discussed:

- The first one is called descriptive observation, in which the observer observes anything and everything, assuming that he/she knows nothing; the disadvantage of this type is that it can lead to the collection of data that may not be relevant to the study and could be time costly.
- The second one of is called selective observation, in which the researcher focuses on different types of activities to help delineate the differences in those activities (Angrosino and Mays de Pérez 2000). The disadvantage of this type is that it can lead to missing important/relevant information due to the researcher's criteria for selecting the activities to be observed.

- The third type is called focused observation, in which more emphasized observation supported by interviews/questions takes place. In this type the participants' insights, behaviour or attitude guide the researcher's decisions about what to observe, ask, clarify, etc. In this study, a focused observation approach is followed due to the reasonable flexibility/freedom it offers (e.g. ability to ask questions during the observation) when observing the participants within the focus of PAFA method evaluation. During the observation process, notes and voice recordings of and time spent on tasks by participants taking part in PAFA evaluations will be taken, qualitatively analysed and reported.

8.3.4.1 WHAT TO OBSERVE

Merriam (1998) noted that the most vital element in determining what a researcher should observe is the researcher's purpose for doing the study in the first place. *"Where to begin looking depends on the research question, but where to focus or stop action cannot be determined ahead of time"* (Merriam 1998). In the context of this study, this thesis follows the suggestions of DeWalt and DeWalt (2010) on what the observer should normally observe. They suggest that observers should sort out the regular from the irregular activities (e.g. observing how participants develop scenarios from personas and point out any irregularity); look for variation to view the event from a variety of viewpoints (e.g. observing all participants doing the same tasks to have a holistic point of view about their performance); look for the negative cases or exceptions (e.g. observing where participants find it challenging to do a task in PAFA), etc.

8.3.4.2 THE TYPES OF THE OBSERVER

The degree to which the observer involves himself/herself in participation in the culture under study makes a difference in the quality and amount of data collected. (Schwartz and Schwartz 1955; Gold 1958; Spradley and Baker 1980; Bernard and Gravlee 2014) provided a description of observer stances as appears in Table 23:

TABLE 23 OBSERVER STYLES.

Type of the observer	Level of Involvement	Disadvantages/Advantages
Non-Participatory/Passive Participation	Observer has no contact with population and is only in the bystander role	Limit the observer to build rapport, ask questions or immerse in the study field.
Moderate Participation	The observer keeps a balance between the involvement and detachment roles.	This allows a good level of involvement and important detachment to maintain objectivity.
Active Participation	The observer acts as a member of the studied group for the sake of complete comprehension	This allows the researcher to become more involved in the population. However, there is a potential risk of bias and subjectivity to influence the final conclusion.

Complete Participation	The observer is completely attached to and integrated with the population of the study beforehand (e.g. being already a member of particular population of a certain study).	This maximizes the risk of losing all levels of objectivity. This can highly harm what the study concludes.

In this study, the observer takes a moderate participation approach to allow for an effective engagement with participants (e.g. by asking/answering questions, discuss raised issues, etc). Additionally, this approach minimize bias by restricting the observer to certain limit of involvement as discussed in the previous table. This could also offer participants more freedom in working with PAFA without having the observer intrusively involved with them.

8.4PARTICIPANTS SELECTION

Since PAFA method is a product that is meant to be used by software engineering practitioners or expert, this study adopts an expert opinion sampling technique (Cooke and Probst 2006) in which it targeted three experienced groups of software engineering researchers (19 participants) from three different universities (Bournemouth University, Utrecht University and University of Brighton). This selection criterion is adopted due to the experience and knowledge the participants enjoy in relation to PAFA method and its components (e.g. goal modelling, feature modelling, etc). In order to further assess the participants experience and knowledge with particular components of PAFA, some assessment questions were introduced to them before the study started (see Appendix 6).

Although the participants work in academia, some of them either worked in industry previously or were engaged in collaborative projects involving industrial partners. This helped in viewing the potential of adopting PAFA method in industry (see Table 24). The participants were invited using emails after several video conferencing meetings were conducted with representatives of each research group to organise and finalize the study sittings, recourses, dates, etc. The participants were also given a £40 Amazon voucher as an appreciation for their participation. Figure 50 and 51 shows a snapshot taken of the participants at Utrecht University while working on PAFA which gives an indication of the observation process. Knowledge level

TABLE 24 THE PARTICIPANTS CHARACTERISTICS (KNOWLEDGE LEVEL. 1= LOW, 2= MEDIUM, 3= HIGH)

Participants	years of experience	Knowledge level in goal modelling	Knowledge level in feature modelling	Knowledge level in UML
Participants from Bournemouth University				
1	4	2	2	3
2	10	3	3	3
3	6	2	2	2
4	2	1	1	2
5	1	1	1	2
6	3	2	2	3
7	3	3	3	3

8	7	3	3	3
Participants from Utrecht University				
9	10	3	3	3
10	4	1	1	2
11	5	1	1	1
12	2	1	1	1
13	8	1	1	2
14	2	1	1	1
Participants from University of Brighton				
15	5	3	1	1
16	7	1	1	2
17	10	3	3	3
18	3	3	3	3
19	11	3	3	3
Average	5.4	1.8	1.7	2.3

FIGURE 49 SOFTWARE ENGINEERS WORKING ON PAFA (UTRECHT UNIVERSITY)

FIGURE 50 THE SOFTWARE ENGINEER (THE RIGHT SIDE) DISCUSSING PAFA ARTEFACTS WITH THE CLIENT (UTRECHT UNIVERSITY).

8.5 ANALYSIS

The returned evaluation forms were analysed and responses were cleaned up and irrelevant/inconsistent answers were excluded. A descriptive analysis on the quantitative part of the survey was conducted to describe the data and to get the feel of it. A qualitative analysis was also applied to the observations and the open-ended questions of the survey and which included coding the response and creating categories to identify patterns and trends in the responses. The quantitative and qualitative findindgs are then reported in conjuction wht each othe.

8.6 FINDINGS

The findings are structured in line with the previously proposed goals and metrics for PAFA evaluation (see Table 22). In addition, the observation (qualitative) part of the evaluation is reported in conjunction with the quantitative part and distinguished in the text using a ***Bold*** and ***Italic*** format (e.g. see Section 8.6.3.4).

8.6.1 PAFA MAINTAINABILITY (*GOAL 1: HOW PAFA FACETS CAN BE REASONABLY UPDATED WITH ANY CHANGES OR ENHANCEMENTS*)

In response to questions regarding PAFA maintainability (see Appendix 6), the majority of the participants (81%) indicated that PAFA has a limited level of maintainability in which a change to one component would likely require a change to the other following components/phases. This is indeed due to the sequential nature of PAFA in which each phase constructs the foundation for the following phase (e.g. personas to scenarios). However, this hierarchical nature of PAFA considered by most of the participants as a vital and powerful way in identifying and sharpening users' requirements by going through several phases of requirement engineering. For example, one participant commented with regards to the sequential transition between PAFA phases *"when I move from one phase to another, it feels like I'm really exploring my persona's requirements more and more"*.

Another valid point of view was suggested by the rest of the participants who believed that a change to one component of PAFA might not lead to a change to the following components. For example, a change to one of the users' intentional goals (e.g. feeling satisfied) might not result in a change to the features to be implemented. One of the participants commented in this regard *"I would say it really depends on the change made. Adding "feeling happy" as goal for a certain persona might not require changing the feature model or use-case resulted from this goal model)"*. This highlights how PAFA is still flexible and maintainable to work with (in specific contexts) despite its sequential nature. This also advocates the need to for software engineers when using PAFA to carefully analyse the nature of changes and whether a change to one of the facets requires a change to other facets of PAFA.

8.6.2 PAFA COMPLETENESS *(GOAL 2: THE EXTENT TO WHICH THE COMPONENTS OF PAFA COVER THE HOLISTIC DESIGN OF FEEDBACK ACQUISITION)*

More than half of the participants (65%) did not think (disagreed and strongly disagreed) PAFA needs to be complemented by more phases or components (e.g. more UML models) in order to improve it (see Figure 52). They believe PAFA is already encapsulating a reasonable amount of modelling to enable a systematic identification of users' requirements (with regard to feedback acquisition) through its sequential phases. Adding more components to PAFA could easily harden the process and make it more complex as commented by one of the participants *"I really think it has got enough modelling. More modelling will definitely make it complex and time-consuming for me at least".*

The participants who replied with yes though that PAFA could also (but not necessarily) be complemented with other modelling techniques (e.g. sequence diagram, activity diagram, etc) as one of them commented *(Why use-cases? I think a sequence diagram or activity diagram could also add value in capturing how the personas interact with the system. However, I believe having a very detailed use-case could eliminate the need for such models".* In addition, although this group of participants think PAFA could be improved by adding more components/phases, they still agree with the first group in which this could add more complexity to the whole method. In general, the participants' responses suggest that PAFA proposes a reasonable level of modelling to cater for personas' behaviour with regard to feedback acquisition. Adding more modelling or components could negatively impact the usability of PAFA and ultimately the potential of adopting it.

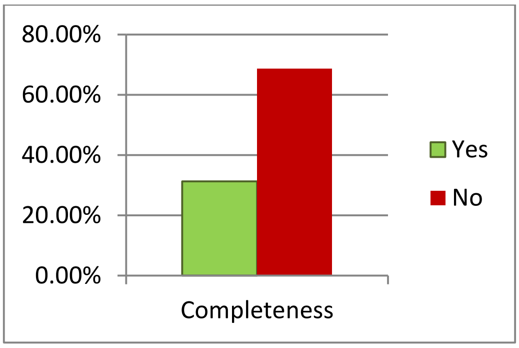


FIGURE 51 PAFA SCORE ON COMPLETENESS MEASURE.

8.6.3 PAFA RELIABILITY AND INTEGRABILITY (GOAL 3 AND 8)

Reliability and Inegrability in the context of PAFA represents two sides of the same coin. Reliability (Goal 3) is concerned with the extent PAFA can meet its claims with regard to catering for users' different requirements and preferences to feedback acquisition. On the other hand, Inegrability (Goal 8) is concerned with the extent to which PAFA components integrate and work correctly together to provide an efficient way of catering for users' different requirements and preferences. In general, PAFA scored a reasonably high level of reliability and integrability (see Figure 53) as around 85% of the participants thought that the sequential transition between PAFA phases provides an effective way for identifying various requirements of users' to feedback acquisition (e.g. moving from step 1 to step 4, in each phase a sharper and clearer view of users requirements was observed). One of the participants commented in this regard "*when I move from one phase to another, it feels like I'm really exploring my persona's requirements more and more*".

On the other side, only 15% of the participants thought that the sequential transition between the phases does not really help in offering better identification of users requirements. They argue that some of the requirements could be missed when moving from a phase to another. For example, if the transition from scenarios to goal models is not conduced in a systematic way (e.g. following approaches suggested in Chapter 7) can result in having goal models that do not necessarily represent what the scenarios are advocating. One of them commented *"you really need to follow a precise approach when moving between phases. I'm not sure if I have picked up all of the goals from the scenarios"*. This highlights the need to have a facilitator who enjoys a high level of experience in PAFA and the approaches it advocates. The facilitator could work as a quality checker to verify the developers follow the right approach when moving between PAFA phases.

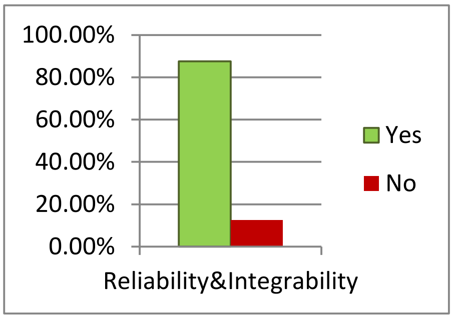


FIGURE 52 RELIABILITY AND INTEGRABILITY (THE PARTICIPANTS RESPONSES ON EFFECTIVENESS OF THESEQUENTIAL TRANSITION BETWEEN PAFA PHASES).

Additionally, each phase in PAFA, as indicated by 67% of the participants, provides a solid foundation towards the following phase (e.g. personas description was highly helpful to develop multiple scenarios). One participant commented *"sure, the persona description did me a great favour when I tried to do the scenarios"*. The rest of the participants thought that some of PAFA's phases do not integrate well with each other as said by one of them *"I do not really know why to have a use-case after the goal model. I'm not sure if it adds value to your method"*. This could be caused by the lack of understanding the purpose of some PAFA phases or the high simplicity of personas assigned to some participants. This simplicity might make them feel they fully understand its requirements by going through the first two phase's only whiteout the need to go through several modelling phases.

In addition and more specifically, Reliability and Inegrability of each phase of PAFA varies and was perceived by the participants as follows:

8.6.3.1 PHASE ONE (PERSONAS TO SCENARIOS)

A reasonably high level of Reliability and Inegrability was given to PAFA's first phase as positively rated by around 65% of the participants (agreed or strongly agreed) (see Figure 54). They believed that the scenarios (developed based on personas during the evaluation study) gave a better understanding of the persona's requirements (e.g. better understanding of the persona's behaviour in different context). They also thought that the persona's narrative was relatively helpful to generate the scenarios but still generic. This indeed advocates the need for the transition from personas to scenarios in order to have a sharper view of each persona's requirements as mentioned by one of the participants *"I personally find the scenarios fun to do and let me dig deep in my persona' possible reactions"*.

On the other hand, around 15% of the participants did not see the need for scenarios (disagreed) and believed that moving straight-ahead to goal models or even use-cases was possible and could save time and effort. One of them commented *"I could see that the information I need to do the goals is already mentioned in my persona discretion. Why to put more effort on doing scenarios!"*. This is perhaps due to the level of details of the persona they were assigned by the client and the case study provided. For example, the narrative of a persona like Sara in the context of the Graduate School feedback case study might not need more exploration and investigation by moving to scenarios due to her simple and clear behaviour at the first place. This would allow software engineers to skip the scenarios phase in PAFA (although it is not recommended due to the possibility to overlook some of the requirements) once they believe the description of a certain persona in a certain context is enough for the transition to the goal modelling phase directly.

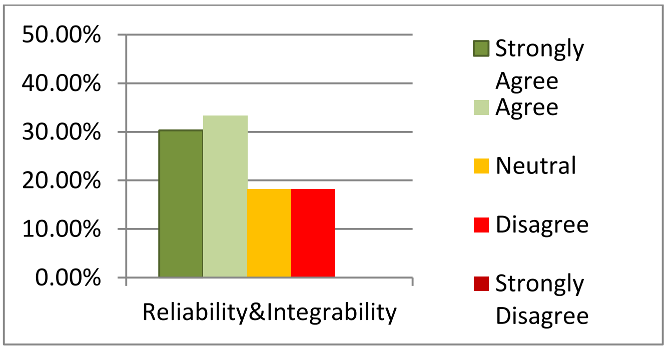


FIGURE 53 RELIABILITY AND INTEGRABILITY OF PHASE 1 (PERSONAS TO SCENARIOS) OF PAFA.

8.6.3.2 PHASE TWO (SCENARIOS TO GOAL MODELS)

In this phase, PAFA Reliability and Integrability scored lower than the first phase (see Figure 55). Around 55% of the participants (agreed or strongly agreed) indicated that goal models (developed based on scenarios during the evaluation study) provided a better understanding of the persona's goals/requirements (e.g. better view of a persona's goals and soft-goals). It also gave a better understanding and visualisation of the business goals (e.g. the Graduate School goals when collecting feedback) and how they relate to the persona's requirements (i.e. the goal model can show the persona's intentions to give feedback and whether their intentions are conflicted or in line with the business goals to collect feedback). This indeed highlights the need for this phase to take place after scenarios in order to have a more sharper view about persona's and business goals, soft-goals and how to reach them.

However, 20% (disagreed) of the participants thought that the goal modelling phase does not really add value to PAFA and it would save time and effort if it is taken out from PAFA. One of the participants commented *"I really wonder why I need to do the goal model thingy. To me it just does not add more information than the scenarios".* This is perhaps caused by the limited experience of some of the participants with goal modelling (as indicated in questions related to their experience with goal modelling, see Appendix 6) and its usefulness in software engineering activities.

However, some of the participants believed that it was a highly beneficial factor for both the clients and the software engineers. It gave software engineers the power to deliver a better view and understanding of the personas' and business goals to the client. This ultimately helped the client and software engineers negotiating and making decisions on how to reach goals as mentioned by one of the participants *"the goal model seems to be very interesting to the client, it seemed to be friendly and understandable by them. To be fair it also helped me clarify and discuss with client".*

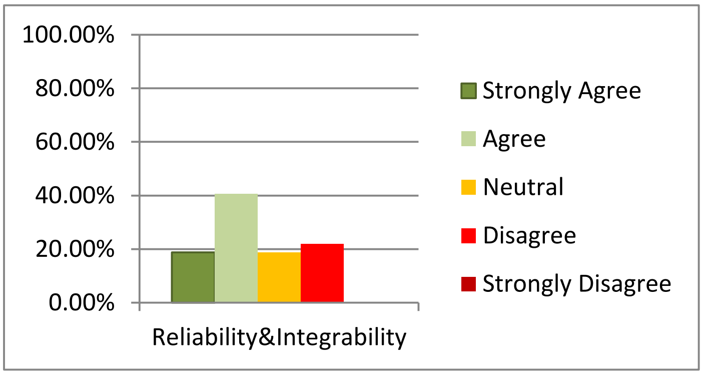


FIGURE 54 RELIABILITY AND INTEGRABILITY OF PHASE 2 (SCENARIOS TO GOAL MODELS) OF PAFA.

8.6.3.3 PHASE THREE (GOAL MODELS TO USE-CASES)

In this phase, PAFA showed a relatively good Reliability and Integrability as indicated by around 70% of the participants (agreed or strongly agreed) (see Figure 56). They believed that the use-cases (developed based on goal models during the evaluation study) gave a better and clearer view of the persona's requirements (e.g. a clearer view about the personas' potential interaction with the software when being asked for feedback). One of the participants commented in this regard "*now the use case seems to explain the whole thing in a very detailed manner".* They also mentioned that the transition from the goal models to use-cases was indeed essential in order to capture the persona's potential interaction with the system. This highlights the importance for this phase to take place after goal models in order to have a sharper view about persona's interaction with the feedback system.

On the other side, around 10% of the participants (disagreed or strongly disagreed) thought that the goal models developed in the previous phase was not helpful enough for the development of the use-cases. They believed that the link was missing between the two phases and PAFA did not provide a clear way to move from goal models to use cases as commented by one of the participants *"I really struggled to find the link between the goal model and the use case. How do I move form goals to use-cases?"* This is perhaps due to the fact that PAFA manual (see Chapter 7) does not provide all the details needed for novice software engineers to move from goal models to use cases. It only gives a comprehensive overview of the well-known approaches for such a transition which suits the need for medium and expert software engineers as indicated in their responses. Furthermore and throughout the PAFA manual, references to well-known approaches for all the transitions it proposes were provided for the purpose of giving software engineers (especially novice ones) the chance to further read and understand how to move between PAFA phases.

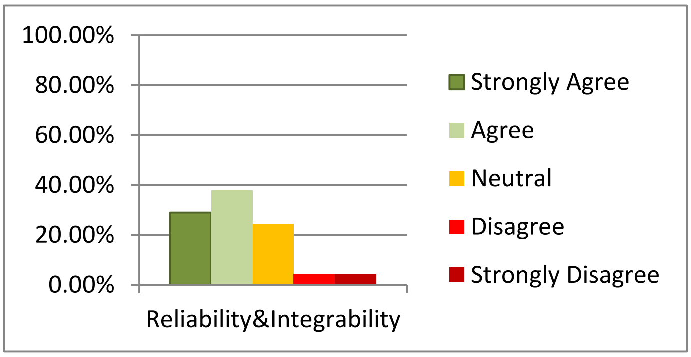


FIGURE 55 FIGURE 9 RELIABILITY AND INTEGRABILITY OF PHASE3 (GOAL MODELS TO USE-CASES) OF PAFA.

8.6.3.4 PHASE FOUR (ADAPTATION ENGINEERING16)

PAFA Reliability and Integrability also enjoys a good level in this phase as indicated by around 56% of the participants (agreed or strongly agreed) (see Figure 57). They believed that the feature models (developed based on goal models during the evaluation study) gave a better understanding of the personas' requirements (e.g. it shows personas' preferred features with regard to feedback acquisition in a certain context). "*It basically summarizes the whole thing in an understandable representation"*.

They also mentioned that the feature modelling stage was needed in order to sum up the previous steps and give a clearer view of the persona's preferred requirements/features that can even be discussed with and understood by the clients as said by one of the participants *"and I*

[16] The evaluation was only conducted on Feature Modelling excluding the rest of the adaptation engineering alternatives (e.g. personality questions) due to the available resources (e.g. available expertise, study time and effort, etc). Additionally, Feature Modelling could be a suitable choice at such level (i.e. first-touch engineering of feedback acquisition).

think it is a user-friendly model. I do not think a normal person would find it difficult to understand, do you?". This was clearly observed during the evaluation study where the clients and software engineers found it beneficial to have such a model. It helped them stimulating more discussion and making decisions about the features to implement for each persona bearing in mind the overall business goals. This highlights the importance for adaptation engineering to take place after the previous phases.

On the down side, around 10% (disagreed) of the participants believe that there is a lack of a systematic way to deduce a feature model based on the previous phases as said by one of the participants *"my question is how to do it. It seems a bit fuzzy to me".* Indeed, this could be a limitation to PAFA for novice software engineers ***(as observed during the evaluation)*** since it only explains how to move from goal models to feature models. It does not provide a systematic way on how to benefit from the scenarios and use-cases to enrich the feature models.

However, the question is whether PAFA needs to propose such a systematic way? Participants who enjoy a high level of experience in software engineering in general and feature modelling in particular (as observed and discussed during the evaluation) highly believed that it is a straightforward process to extract features and contextual information from scenarios and use-cases to enrich the resulted feature model. They also thought that having several approaches (in addition to the goal to feature model proposed approach in PAFA) to build the feature model based on the previous phases could add a lot of complexity and effort on software engineers to adopt PAFA in the future. One participant commented in this regard *"why to make things harder! I believe if you really built the previous phases in a proper way, then you fully understand your personas needs which makes it straightforward to extract the features form them. It is a common sense I believe".*

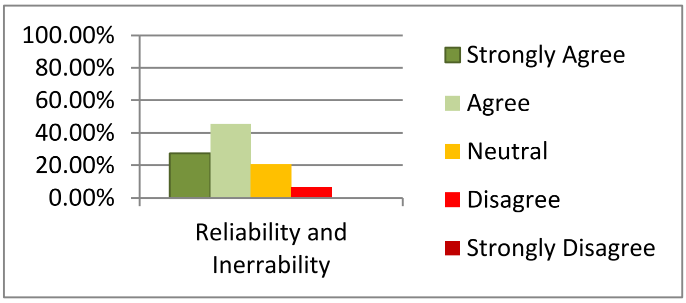


FIGURE 56 RELIABILITY AND INTEGRABILITY OF PHASE4 (FEATURE MODELLING) OF PAFA.

8.6.4 PAFA PORTABILITY (*GOAL 5: THE EXTENT TO WHICH PAFA CAN BE ADOPTED IN OTHER APPLICATION DOMAINS)*

Participants were asked whether PAFA shows applicability and scalability to be adopted in different application domains to design feedback acquisition such as health-critical systems or privacy-critical systems. Interestingly, all the participants reach a consensus that PAFA is flexible and portable enough to be adopted in different application domains. This is due to the powerful and widely used requirement engineering techniques that are adopted in PAFA as said by one of the participants *"the modelling techniques you are using in PAFA are popular in software engineering and are used in different application domains. This can make your method highly portable I suppose"*.

However, some of the participants were concerned about the use of the proposed personas in different domains without taking into account the characteristics of such domains. For example, privacy tolerance or the involvement of some social factors in the feedback acquisition for Linda should be decreased in health critical systems where personal information could cause a harm to some personnel. This sheds the light on PAFA's power and the emphasis it puts on the importance of involving the client (e.g. the company's ethics expert) during the development process of the feedback acquisition to overcome such issues.

8.6.5 PAFA MATURITY *(GOAL 7: THE EXTENT IN WHICH ENOUGH LITERATURE IS READILY AVAILABLE ABOUT PAFA'S COMPONENTS)*

In response to questions regarding PAFA maturity and whether each one of its component/phase has already a rich literature to refer to for more information when needed, around 80% of the surveyed participants (see Figure 58) positively responded (agreed or strongly agreed). One of them commented *"of course, plenty of literature is easily accessible for every phase of PAFA I suppose"*. This is indeed evidenced by the rich and trustworthy references PAFA manual provides to the available literature with regard to the approaches it adopts (see Chapter 7). The rest of the participants preferred to stay on the fence and this is perhaps due to their lack of awareness about the literature available with regards to some of PAFA's phases as said by one of them *"I know there is lots of literature about use-cases and scenarios but I'm not really aware of the others such as goal models and feature models". Participants' response to the statement "*each component/phase of PAFA has already a rich literature to refer to for more information.

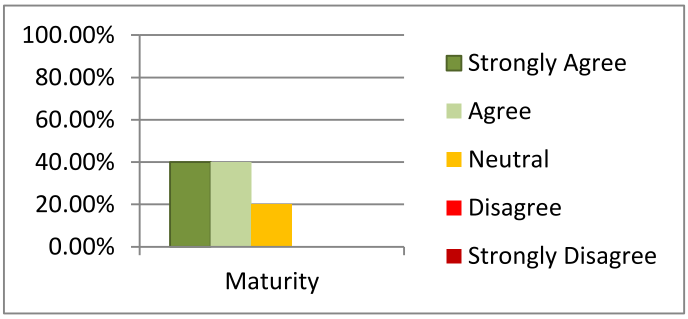


FIGURE 57 PAFA SCORE ON MATURITY (PARTICIPANTS' RESPONSE TO THE STATEMENT "EACH COMPONENT/PHASE OF PAFA HAS ALREADY A RICH LITERATURE TO REFER TO FOR MORE INFORMATION).

8.6.6 PAFA EFFICIENCY (GOAL 4: THE AMOUNT OF EFFORT, RESOURCES AND EXPERTISE REQUIRED BY PAFA IF ADOPTED.)

In general, the efficiency of PAFA (as a whole) seemed to be a debatable matter among the participants. The participants with medium to low experience in requirements engineering ***(as***

observed during the study) either were unsure about PAFA efficiency or believed that PAFA lacks efficiency and is a time-consuming process (see Figure 59). One of the participants commented in this regard *"for someone like me who is not really familiar with some of the modelling languages used in PAFA, It will take me some time to understand and work on and this is simply inefficient to me!"*

On the other side, highly experienced participants in requirements engineering ***(as observed)*** were positive about PAFA efficiency and thought it requires reasonable expertise, time and effort as said by one of them *"I personally think that the used modelling techniques (e.g. feature model) in PAFA are somehow widely known for experts in requirements engineering but to be fair, let's say some of the experts have never heard about Feature Modelling for example, then honestly it is not something hard to learn in a short time!"*. Indeed, the participants' comments suggest that their experience plays a vital role in affecting their perceptions about PAFA efficiency as mentioned by one of the participants "*if I do it for a second time it will surely take me less time to get it done because now I feel more familiar with the modelling in it (meaning PAFA)"*.

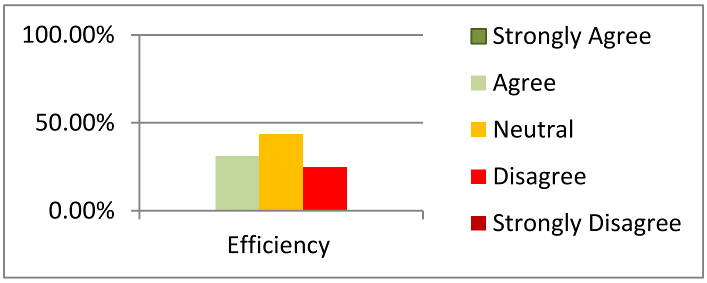


FIGURE 58 PAFA SCORE ON EFFICIENCY MEASURE (PARTICIPANTS' RESPONSE TO THE STATEMENT "IS PAFA A TIME-CONSUMING PROCESS?").

In addition, observing the participants during the study, the average time taken by all the participants to finish the study was around two hours and a half. However, the time taken on each phase of PAFA varies as seen in Figure 60. The first phase (Persona to Scenarios) took the highest time by the participants. This is rationally caused by their need to get familiar with PAFA at first (which cost them some time) before they start the development process. This is evidenced by the decrease in the time consumed on the following phases where the participants became more familiar with the personas and scenarios. In general, the results suggest PAFA as a time efficient method in which it was conducted by the participants in around two hours and a half bearing in mind the size of PAFA, available expertise, study conditions (e.g. participants did not have the time to familiarise themselves with PAFA before the study took place).

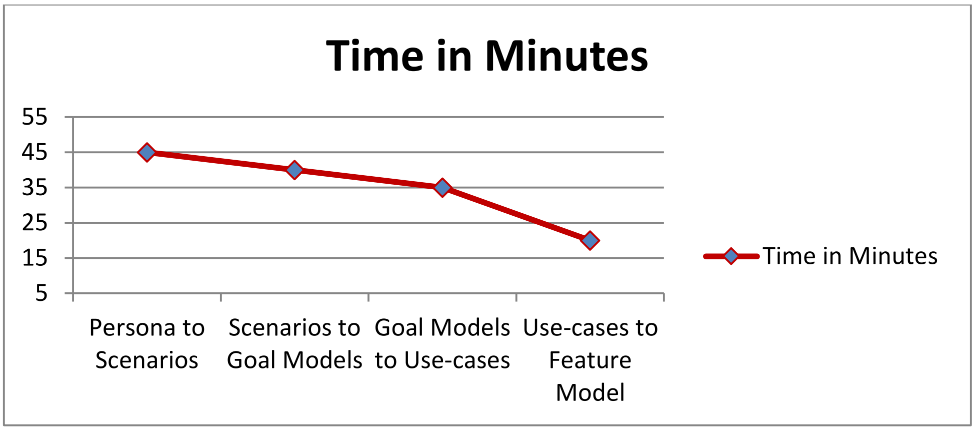


FIGURE 59 AVERAGE TIME TAKEN IN EACH PHASE.

In addition and more specifically, efficiency of each phase of PAFA varies and was perceived by the participants as follows:

8.6.6.1 PHASE ONE (PERSONAS TO SCENARIOS)

PAFA's efficiency enjoys the highest level in this phase as 68% the participants (see Figure 61) responded negatively (disagreed or strongly disagreed) when asked whether it was a hard and time-consuming process to come up with the scenarios. This is due to the simplicity of the step in which no modelling is involved comparing to the other steps/phase of PAFA as said by one of the participants *"Nothing really complex here. It is fun and easy to do. Just let your imagination free!"*. On the other hand, only 6% of the participants (agreed or strongly agreed) thought this phase required them a lot of time and effort as commented by one of them *"I really needed to think lot about the possible scenarios and write them in the proper form"*. This could be really caused by the lack of knowledge they might have with regard to scenarios developments in general.

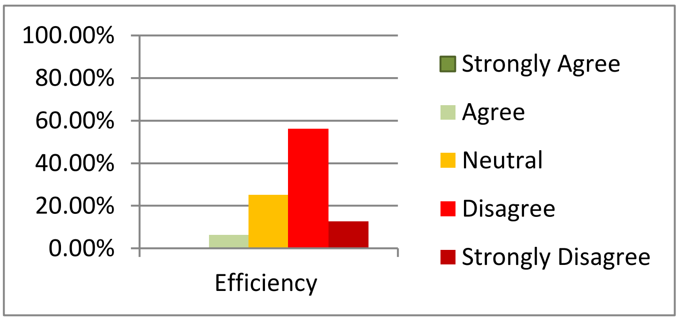


FIGURE 60 EFFICIENCY OF THE FISRT PHASE OF PAFA (PARTICIPANTS' RESPONSE TO THE STATEMENT "IS PAFA'S FIRST PHASE A TIME-CONSUMING PROCESS?").

8.6.6.2 PHASE TWO (SCENARIOS TO GOAL MODELS)

Around 37% of the participants (neutral) (see Figure 62) were unsure and preferred to stay on the fence when asked about the efficiency of this phase as one of them commented "I'm actually confused and cannot give a sharp answer. My knowledge about goal modelling is not that good so it might be my problem not the method efficiency! I will just go neutral about it". On the negative side, around 25% of the participants believed (agreed or strongly agreed) that it was a hard and time-consuming process to develop the goal model from the scenarios (e.g. goals and soft-goals were hard to identify). In response to this, around 37% of the participants positively viewed the efficiency of this phase of PAFA (disagreed or strongly disagreed). Once participant commented *"of course it would be inefficient if you have poorly developed scenarios at the first place. I personally believe if you develop high quality ones then I believe it straightforward to move to goal models"*

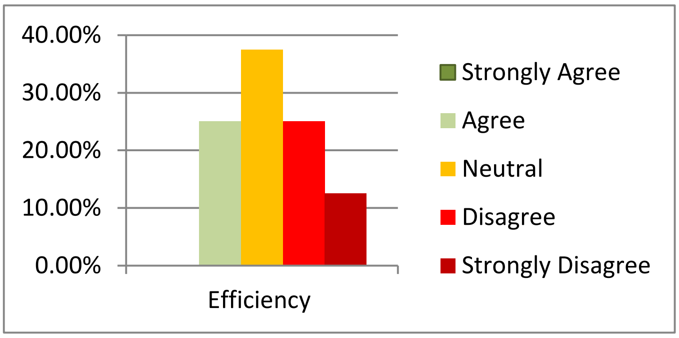


FIGURE 61 EFFICIENCY OF PHASE 2 OF PAFA (PARTICIPANTS' RESPONSE TO THE STATEMENT "IS PAFA'S SECOND PHASE A TIME-CONSUMING PROCESS?").

8.6.6.3 *PHASE THREE (GOAL MODELS TO USE-CASES)*

Around 26% of the participants (see Figure 63) were unsure about the efficiency of phase three and preferred to stay on the fence due to reasons related to their level of knowledge about the development of either goal models or use-cases. On the negative side, around 26% of the participants believed (agreed or strongly agreed) that this phase of PAFA was hard and time-consuming process to come up with the use-cases (e.g. the goal model did not provide enough information to build a complete use-case). One of the participants commented "*The link between the goal model and use-case was missing. I did not know how to do the transition"*. In response to this, around 47% of the participants positively viewed the efficiency of this phase of PAFA. Once participant commented *"As long as you are aware or knowledgeable enough about the ways to move from goals to use-cases then things will be much easier to do. I think the manual clearly state some of the approaches to do the job!"*

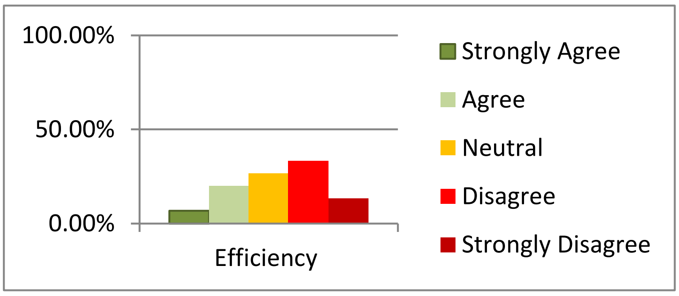


FIGURE 62 EFFICIENCY OF PHASE 3OF PAFA (PARTICIPANTS' RESPONSE TO THE STATEMENT "IS PAFA'S THIRD PHASE A TIME-CONSUMING PROCESS?").

8.6.6.4 PHASE FOUR (USE-CASES TO FEATURE MODELS[17]*)*

Around 33% of the participants (see Figure 64) went neutral with regard to the efficiency of this phase. As mentioned in the previous phase this could be due to reasons related to their level of knowledge about the modelling techniques in this phase and its preceding. On the negative side, around 31% of the participants believed (agreed or strongly agreed) that It was a time-consuming process to develop the feature model as said by one of the participants *"It took me time and effort to filter the information yielded from the previous steps and come up with the features, no clear way on how to do things was provided"*. On the positive side, around 33% of the participants positively viewed the efficiency of this phase. *"Once I finished the use-case for Hana, It feels like I cumulated enough information about her from previous phases and I would easily know what features she likes"*.

In addition, PAFA manual provided explanations and several widely known references to approaches on the transition from phase three to phase four (see Chapter 7). However, time and conditions of the study (e.g. the participants did not have time to familiarize themselves with PAFA manual in advance of the study) could affect how some of the participants negatively viewed the efficiency of PAFA in this phase in particular and PAFA as a whole in general.

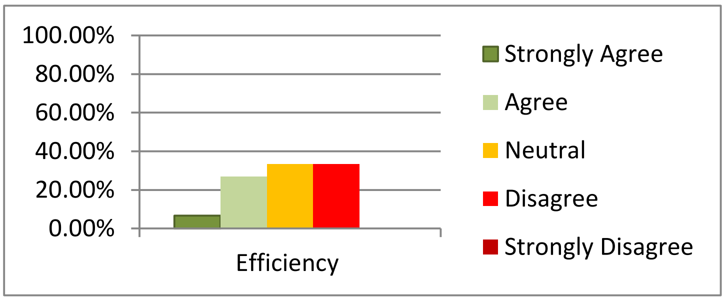


FIGURE 63 EFFICIENCY OF PHASE 4 OF PAFA (PARTICIPANTS' RESPONSE TO THE STATEMENT "IS PAFA'S FOURTH PHASE A TIME-CONSUMING PROCESS?").

8.6.7 PAFA USABILITY *(GOAL 6: MEASURING THE EASE TO LEARN, INTERPRET AND USE PAFA)*

In general, PAFA usability was measured from several angles and the overall result suggest that PAFA enjoys a high level of usability as discussed in the following dimensions:

8.6.7.1 DIMENSION1: INFORMATION PROVIDED ON PAFA'S AIM AND IMPORTANCE

PAFA evaluation study (see Appendix 6) tried to measure whether PAFA manual (see Chapter 7) provides enough information and explanation on PAFA's aim importance at the first place. In

[17] The evaluation was only conducted on Feature Modelling excluding the rest of the adaptation engineering alternatives (e.g. personality questions) due to the available resources (e.g. available expertise, study time and effort, etc). Additionally, Feature Modelling could be a suitable choice at such level (i.e. first-touch engineering of feedback acquisition).

response to this measure, the majority of the participants' responses (94%) came positive (agreed or strongly agreed) (see Figure 65) indicating a relatively high level of usability in regard to this dimension. One participant commented in this regard "*It is stated clear in the manual, at least to me, why you need such method".* However, 6% of the participants went neutral about this dimension and the reason could be inferred from one of the participant's comments "well, I really do not have enough knowledge to judge the importance of PAFA on software success. Probably someone with more expertise in this domain could tell!"

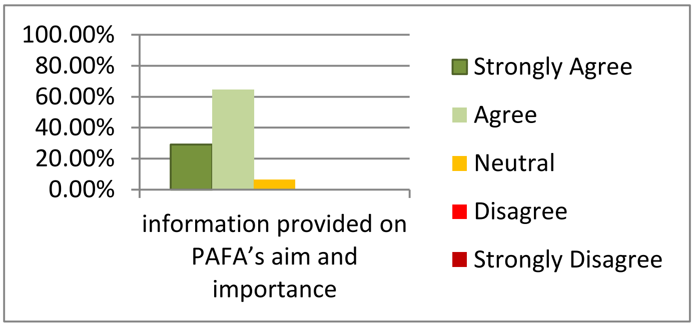


FIGURE 64 PAFA USABILITY (PARTICIPANTS' RESPONSES ON THE STATEMENT "IS ENOUGH INFORMATION PROVIDED ON PAFA'S AIM AND IMPORTANCE?").

8.6.7.2 DIMENSION2: UNDERSTANDING, INTERPRETING AND USING PAFA

Understanding PAFA and how to use it plays a critical role on its success and adoption by others. This led to the need to find out how the participants found it to understand, interpret and use PAFA based on the provided manual (see Chapter 7). In response to this dimension, around 68% of the participants (see Figure 66) were positive (agreed or strongly agreed) which gives PAFA credit for clarity and simplicity in information delivery as commented by one of the participants *"I see that the manual comes clear about explaining PAFA and how one can use it. The examples did a great job"*. On the down side, only 9% of the participants negatively viewed this dimension and asked for a better clarification of PAFA's techniques. This could be due to the level of details provided by PAFA manual about approaches to on how to move from one step to another as mentioned by one of the participants *"my question is how to do it. It seems a bit fuzzy to me. I need more explanation*".

However, it is a trade-off between having a reasonable amount of information and too much information (unneeded information) as said by one of the participants *"too much reading and literature"*. The rest of the participants couldn't judge whether PAFA was delivered clearly. One of the comments shed the light on the reason for their confusion *"If I was an expert in every phase of PAFA then I could probably judge the amount of information provided. For example, I think the feature model needs more explanation. However, an expert in feature modelling could have another view".*

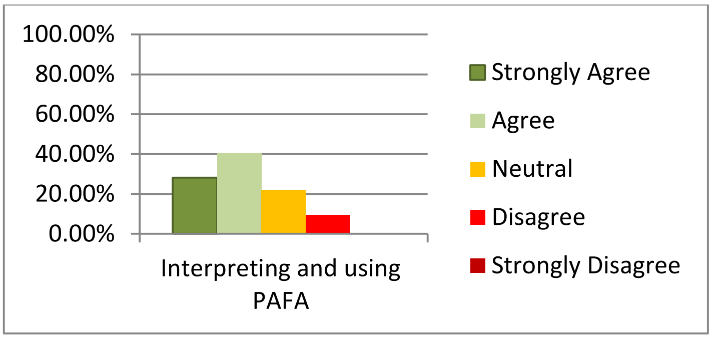


FIGURE 65 PAFA USABILITY (UNDERSTANDING, INTERPRETING AND USING PAFA).

8.6.7.3 *DIMENSION3: LEARNABILITY*

Learnability is one of the important factors' for the success of software engineering methods in general since it determines the potential of adopting and using such methods especially by novice software engineers. This dimension was investigated in PAFA evaluation to see how PAFA method can be easily learnt by novice software. The participants' responses were based on their experience in the domain (e.g. teaching new software methods to students). The majority of the participants (81%) thought that PAFA can be easily grasped and learnt by beginners at software engineering (see Figure 67). This is due to the low level of complexity in its modelling techniques as well as the reasonable amount of literature (supported with examples) it offers to practitioners to explain how to go about PAFA. One of the participants commented in this regard *"If you have the manual and the examples plus the needed references to literature, I think it is more than enough for anyone to easily learn your method"*.

On the downside, around 19% of the participants believed novice software engineers could find it challenging to learn PAFA unless more literature and explanation is provided in its manual. This is really controversial and is a trade-off between having enough information and too much information (unneeded information) as said by one of the participants *"too much reading and literature"* (as discussed in the previous dimension). However, PAFA still provides references to literature in regard with its modelling techniques to help software engineers further exploring and understanding PAFA when needed.

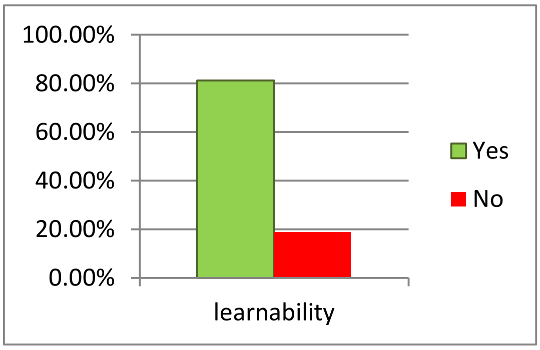


FIGURE 66 PAFA USABILITY (LEARNABILITY).

More specifically, usability of each phase of PAFA scored slightly different in terms of the information provided to define the each 'phases concept (e.g. scenarios), why they are used for PAFA and how to use them in practices. In general, all phases enjoy relatively high positive responses by the participants ranging from 80% to 90% (se Figures 68, 69, 70, 71). This is a highly positive indicator about PAFA's usability and shows the balanced usability between the phases. This could lead to less effort for software engineers when moving between PAFA phases as complexity level would stay similar in all phases. One of the participants commented in this regard *"I feel like I spent almost the same time and effort on each step. This means they were all somehow in the same level of difficulty to me"*.

The rest of the participants have some concerns about certain phases (e.g. goal modelling, feature modelling) and the low level of information provided on the practice of such phases. As previously discussed, this is really *a matter of how the participants perceive the provided information. For example, medium to highly experienced participants were always positive about the amount of information provided by PAFA manual whereas beginners were always asking for more information. However, having more information could also decrease PAFA's usability due to what was mentioned by one of the participants as "too much reading".*

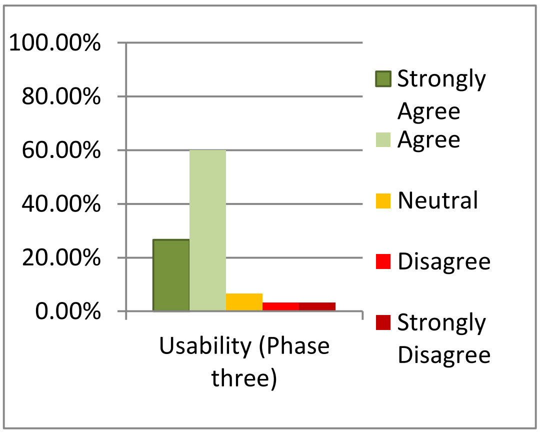


FIGURE 67 USABILITY OF PHASE 3 OF PAFA.

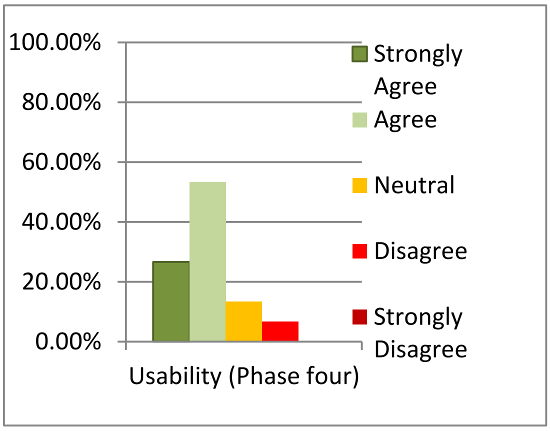


FIGURE 70 USABILITY OF PHASE 4 OF PAFA.

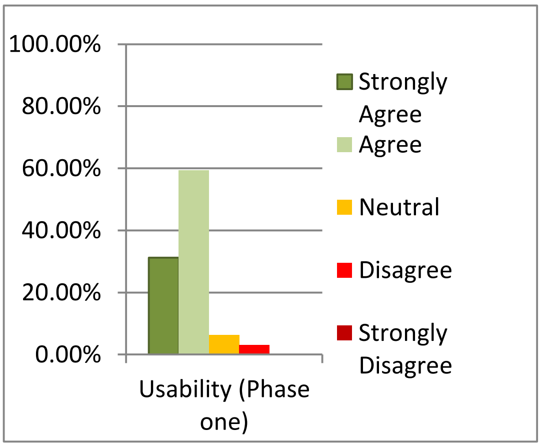


FIGURE 68 USABILITY OF PHASE 2 OF PAFA.

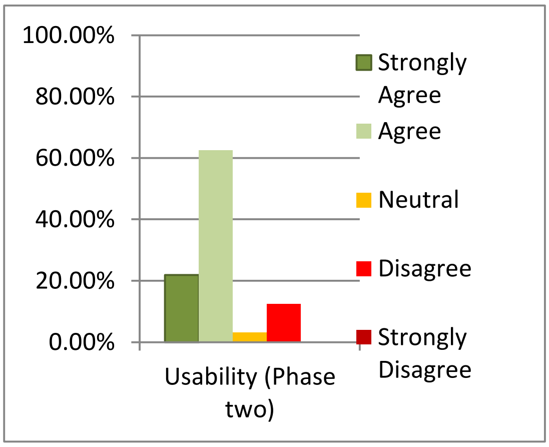


FIGURE 69 USABILITY OF PHASE 1 OF PAFA

8.7CLIENT CONFIRMATORY PHASE

The purpose of this phase is to assess the impact of using and adopting PAFA for the design of feedback acquisition in real case studies. The assessment comes from the client (Graduate School and IT service desk representatives at BU) point of view to enhance, complement and counter balance the findings of the previous phase of PAFA evaluation which manly comes from the software engineers' point of view.

8.7.1 STUDY DESIGN AND PARTICIPANTS SELECTION

In order to reach the aim of this study, four semi-structured interviews were conducted with participants from the Graduate School and IT department at Bournemouth University (two participants from each) in relation to the aim of this phase. These face-to-face interviews were conducted at Bournemouth University and each interview lasted for about 45 minutes. A purposeful sampling method was used to select candidate participants in which the author only interviewed participants who directly deal with users' feedback collection at the Graduate School or the IT department/desk. This selection criterion is hoped to give more credibility to the findings of this phase since all participants/clients enjoy a high level of experience and knowledge with regard to users' feedback collection.

The interview protocol was developed in consultation with the literature (Creswell 2013) and the questions were designed to investigate and discuss the aim of this phase (see Appendix 7). The interviews were then audio taped and transcribed verbatim as a preparation for analysis. Analysis was performed following the qualitative approach adopted throughout this thesis (see chapters 5 and 6)

8.7.2 FINDINGS

The findings of this phase are structured under two main themes as follows:

8.7.2.1 PAFA BENEFITS TO CLIENTS

This theme encapsulates the clients' positive point of view about PAFA, the techniques it proposes and the benefit it could offer. This is expressed in the following codes:

- **Personas advantages:**

In the first place, personas which represent the foundation for PAFA were highly regarded as useful by the interviewees due to the representations they offer about users' behaviour to feedback. They believed the presented personas could help them changing the typical view and assumptions they already have in mind about users' preferences and behaviour to feedback. One of the participants commented *"You know we as IT people always think we understand the users and we know what they want but in reality things are a bit different. I really did not think of some of the personas you showed like the perfectionist one (Hana)"*.

In addition, the interviews indicated that the presented personas can also promote several benefits such as a better engagement and discussion between the clients and the software engineers. This can positively improve the requirements identification process and ultimately the design of the feedback acquisition. This is perhaps due to the nature of the developed personas (e.g. the picture, narrative, demographics, etc) which makes them more interesting and attractive. An interviewee commented *"they are really attracting and fun. The whole profile makes you closer to the users and gives you the feeling that you are really developing for real people".* This indeed shows the importance of the developed personas and how they could inspire, engage and stimulate thinking of both software developers and clients about users' behaviour with regard to feedback acquisition.

- **Understanding business goals and helping making decisions:**

The participants were asked whether PAFA could increase their understanding about their business goals (with regard to feedback acquisition) and improve the decision making process. For example (the goal model can show the users' intentions to give feedback and whether their intentions are conflicted or in line with your business goals to collect feedback) (e.g. Sara, the incentive seekers, could provide low quality feedback. This could go against your goal to collect high quality feedback).

In general, the participants' responses were positive as indicated in one of the given comments "*I could see some great business value in it, whether it is for improving customer satisfaction or for making decisions about the way we go about gathering costumers' feedback. For example, decisions could be made based on the goal model you showed me and the pluses and minuses which help knowing which route to take bearing in mind for example our available budget. Even the last model you showed (feature model) could also offer some help in knowing which feature to go for or remove!"* In general, the participants thought that PAFA, if well understood and used by the client, can offer a good value to the business by improving their the way they gather feedback, costumers' satisfaction and ultimately the business' success.

- **Better understanding of users behaviours:**

In general, the clients believed that the way PAFA is designed could offer them a better understanding of their costumers/users behaviour when approaching them for feedback. This is presented in the hierarchical link between PAFA phases (E.g. moving from step 1 to step 4) which is believed by the client to provide a useful and detailed breakdown of the possible behaviours of each persona (i.e. the scenarios phase presents a better view of potential behaviours of a persona in different contexts).

One of the clients commented in this regard *"Although I'm not experienced in software development but it feels like that each phase of PAFA shows more details about the persona's behaviour from a different angle and this makes us more aware of our costumers' behaviours".* This highlights the power of PAFA in providing businesses with a better view of their costumers' behaviour with regard to feedback acquisition and enables them to cater for these behaviours bearing in mind their own business goals.

- **Highlighting the importance to cater for users' behaviour to feedback:**

"I did not know it could be that crucial to take care of their behaviour when we ask for feedback!" This was one of the comments made by the participants when discussing PAFA and how it enables an adaptive way to users' behaviour when collecting feedback. They believed that PAFA would encourage them to think again and more seriously about the way they gather customers/users' feedback and the impact it could have on customers' satisfaction, the given feedback and the provided service.

This was illustrated in one of the participants' comments *"Now I can see how important it is to take into account those behaviours. It is really nice to see how the behaviour can have an impact on the quality of feedback and the customers' satisfaction too".* In addition, PAFA gain more credits as some of the clients easily grasped and understood PAFA. One of them commented "I think you did a great job in making it more public-oriented method. I did not need to be an expert in computing to understand your method". This shows more potential for PAFA to be adopted in businesses.

8.7.2.2 CHALLENGS OF PAFA ADOPTION

On the down side, the interviewees identified two main challenges that could be faced when adopting PAFA. These challenges are as follows:

- **Business Acceptance of PAFA**

Some of the critical points about PAFA that were raised by the clients is that, PAFA adoption could be met with rejection by businesses due the revolutionary approach it advocates in comparison to the traditional ones being used for feedback collection. One of the participants commented *"I see that it could be hard to convince people to use your method. People do not normally feel comfortable to move to something more systematic and methodical like PAFA. They just find the old ways easier".* This suggests the need to carful market and communicate PAFA to industry. Fortunately, PAFA manual (see chapter 7), as indicated by the participants (see previous section), highly emphasized the importance of PAFA and the positive impact it could have on users' satisfaction, feedback quality and software success. This could highly increase the potential of having PAFA adopted in industry.

- **Cost and Expertise:**

Cost and expertise were mentioned by the participants as the most important factors that can affect the potential to adopt PAFA in industry. One of the participants commented in this regard *"we are always restricted to a certain budget. Using PAFA means we will need more money for incentives and also we will need to hire new people who can work on PAFA or train our people to work on it".* This is indeed a valid point of view since PAFA, like probably any other software engineering method, requires certain expertise to work with. Fortunately, the expertise required by PAFA are not highly rare and can be easily met since PAFA employs popularly used software engineering techniques (e.g. use-cases) as indicated by the participants (see previous section). This increase PAFA acceptance rate in industry since the required expertise are easily learnt as indicated by the participants (see previous section) and widely available.

In general, the client point of view was vital to assess whether PAFA could positively impact their feedback acquisition process. It also complements and enhances the software engineers' point of view about PAFA as discussed in the first phase of this study. Apart from the challenges raised by the clients about PAFA, their inputs generally suggest that PAFA could highly improve the way they gather users' feedback, their understanding business goal, customers' behaviours and satisfaction and ultimately the success of their provided services.

8.8LESSONS LEARNED

As the method used generally succeeded to evaluate PAFA has, there are some resulted lessons during the evaluation process that worth considering for future use of such an evaluation method. The following discuss these lessons:

- PAFA manual, as mentioned by some of the participants, contained a large amount of information that they did not need to read at least during the study (e.g. the amount of theoretical information about the techniques used by PAFA). This could be a valid point in this particular case which suggests the need to minimize the amount of literature/information available to fit the context of the evaluation conducted and the experience level of the participants (highly experienced participants would prefer less information). However, bearing in mind that the ultimate target of PAFA is software engineers in general regardless of their experience level (e.g. novice software engineers), the manual tries to encapsulate a reasonable amount of literature and references that could provide maximum help to the targeted audience in order to fully understand and use the method.
- In person explanation of PAFA was ***observed*** to offer a great help to the participants which explains the need to have an expert in PAFA (besides having the manual) who can facilitate and lead the development process. This role is mainly concerned with offering further clarifications and explanations to developers and conduct quality checks on the produced artefacts. For examples, the facilitator could verify that developers produce the appropriate models and follow the right approaches to move from one step to another.
- The short discussion/focus group conducted with the participants after the evaluation was highly valuable and helped the participants to freely and better reflect on PAFA. This suggests the need to allow some time to freely discuss with the developers during or after the evaluation process of such methods which would positively impact the final product.

- It was also observed that working individually showed a good value as each individual could easily understand his own work better when moving between the phases (e.g. an individual might find the scenarios developed by other software engineers useless for him/her to develop a goal model and lack lots of information). However, they also needed to collaborate at certain stage (e.g. to help stimulate ideas). This suggests the need to allow for collaboration between the participants but also control how and when they collaborate to minimize any negative effects that could result from a group work (e.g. dominant individuals could result in others adopting the same way they work).
- The client involvement was highly valuable and important to the success of PAFA. It was ***observed*** that the client involvement during the scenarios, goal models and feature models development process stimulated a great deal of discussion which positively impacted the development process. For example, the client could contribute as a quality checker in which they verify the developed artefact against their business goal and then suggest modifications to developers. In addition, the client involvement was appreciated from a business point of view (see section F) in which they could have better and deeper understanding of their business goals with regard to the context of the method under evaluation.

In the light of the findings of this study and the lessons learned, this evaluation suggests the involvement of several stakeholders for PAFA which could maximize the success of the designed feedback acquisition (see Table 25).

TABLE 25 STAKEHOLDERS INVOLVEMENT IN PAFA METHOD.

Stakeholder	Involvement Phase	Roles
Software engineers	Complete involvement in all phases.	Development of PAFA artefacts
End-users	Active involvement in the adaptation engineering phase.	Expressing their preferences ether through their *direct input* or c*onfiguration of feature models* (see chapter 7) to fit their individual needs.
Business clients (e.g. head manager, budget manager, ethical expert, etc)	Active involvement in three phases: The development stage through *Scenarios* and *Goal Models* The adaptation engineering phase through *Feature Models.*	Quality checker of produced artefacts with respect to the business goals and needs. Configuration *of feature models* to fit their business goals and contexts (e.g. business ethical expert could decide not to go for social recognition features due to the context of their business where treats privacy is a high concern).
PAFA Expert/Facilitator	Complete involvement in all phases.	Leading and managing the development process by offering clarifications and explanations to developers and conduct quality checks on the produced artefacts.
HCI Expert	Active involvement through the *Evolution Phase*.	Reflect on how the continues process of personas evolvement should take place (e.g. how to capture emerging or changing behaviours of users and express them in the personas, add personas or even eliminate some). This role is delegated to HCI expert due to the expertise they enjoy in relation to this phase of PAFA.

8.9THREATS TO VALIDITY

- The examples given in PAFA manual could have influenced the quality of the participants' response. For example, they could have follow or adopt to a large extent the ideas and information represented by the example. To minimise this effect, the study facilitator continually encouraged the participants to think out of box and generate their own idea.
- As any study that contains incentives subjects, the vouchers given to the participants could have affected in a positive way the truthiness of their judgments about PAFA. However, to minimize such an effect, the incentives were handed out to the participants after doing the study and not in advance of it.
- The time limit given to the participants was tight and could affect the quality of their performance as raised by some of them. However, highly experienced participants with regard to PAFA felt comfortable with the time limit and believed that it would not cause any harmful effect to the quality their outcome. This could counter-balance the effect on the quality of the produced work.
- The clients involved in the first phase of this study were actually experienced software engineers. This could have affected how they involved in the process. For example, they forget sometimes their client role and start discussing the modelling parts of PAFA (e.g. how the goal model was created from the scenarios). To minimize this effect, the study facilitator kept monitoring the discussion between the clients and the engineers.

8.10 SUMMARY

In this chapter, a two-phase style method for evaluating PAFA (a Persona-based Adaptive Feedback Acquisition) was introduced using real case studies and then reported on the findings. In general, PAFA scored reasonably high with regard to the proposed measures (e.g. usability, efficiency, reliability, etc). This indeed shows the potential of PAFA and how it could positively impact the practice of feedback acquisition in software applications. The findings suggest that PAFA offers a notable benefits and potential to improve the process of feedback acquisition in software applications and enable a systematic, user-centric and adaptive way for collecting users' feedback. This can ultimately lead to a better users' satisfaction, feedback quality and software success.

9. CHAPTER 9: DISCUSSION, FUTURE WORK AND CONCLUSION

To conclude this thesis, this chapter presents discussions on the design of feedback acquisition, thesis contributions to knowledge, future work and conclude the thesis.

9.1PAFA APPLICATIONS

The Persona-based Adaptive Feedback Acquisition (PAFA) proposed by this thesis enjoys flexibility and applicability to benefit different application domains. This section discusses some of the potential usage of PAFA in different domains.

9.1.1 PAFA AND FEEDBACK ACQUISITION

One can ask a question, is PAFA only for feedback acquisition, can it be for any other domains? What is special about it? To answer these questions one need to understand the reason PAFA was built this way. Going back to the result of previous studies of this thesis (see chapters 4, 5 and 6), one can observe that users' perceptions and behaviours with regard to feedback acquisition significantly vary and are affected by a number of factors. This variety among users' behaviours highlights the great need for an adaptive feedback acquisition method which could cater for such variety. Having such a method will have a positive side-effect on the feedback quality and users' satisfaction and trust in the system. The starting point of PAFA was to understand and explain this variety in users' behaviour with regard to feedback in particular by using personas and then provide a development process that treats users' behaviour as a priority.

The special about PAFA is that it explains and looks at users' behaviour from different angles through its phases (e.g. goal models explain users' intentions whereas use-cases explain their interactions with the software). This constructs a vital key for the success of PAFA since it provides a clear explanation and efficient catering for users' behaviours which are key element s in having a successful of feedback acquisition thus users' experience and software application. However, this does not really limit PAFA to feedback acquisition only. In fact, it gives it more credit and makes it more applicable to be adopted in other domains where understanding and catering for users' behaviours construct a vital element for the success of the software. For example, PAFA could offer help for designing smart home software for dementia patients as long as personas that represent the patients' different behaviours with regard to the context of the software (e.g. patients' interactions with home control assistant applications) are developed at the first place.

9.1.2 SOCIAL ADAPTATION AND PAFA

Since users' feedback plays a vital role in enabling Social Adaptation (Ali et al. 2012) or crowed-based adaptation where the context of this thesis takes place, PAFA could potentially be an appropriate fit for such systems. In Social Adaptation, users work as monitors. This introduces a range of challenges for the engineering of this human-based monitor. Examples of such challenges include: the specification of the type of users who are capable and should be authorized to provide feedback and the methods used to gather this feedback. In addition, social and cultural differences between users introduce another range of challenges for the engineering of feedback acquisition since users' behaviour and reaction to feedback acquisition could be highly influenced by their cultural backgrounds (i.e. how could social interactions among users with different cultural backgrounds affect their willingness to give feedback?). These challenges led to this research devising a systematic and adaptive way to address these challenges by proposing PAFA.

PAFA offers potential solutions to those challenges. It provides a clear description of users behaviours (represented in 7 personas), methods preferred and social and cultural differences between users and their effect on feedback. PAFA also provides an efficient adaptive and systematic method to cater for such behaviours. This really improves users' satisfaction and feedback quality hence the quality of the feedback-based adaptation which provides a step forward towards enabling better socially-adaptive systems.

9.2 THESIS CONTRIBUTIONS

This thesis contributes to knowledge through the following as discussed in the following sections.

9.2.1 IDENTIFICATION OF THE ENGINEERING CHALLENGES FOR SOCIALLY ADAPTIVE SYSTEMS

The first contribution of this research is the identification of the engineering foundations and challenges in the area of feedback acquisition for socially-adaptive systems (see Chapter 3). This was done through the conduct of an Expert Survey to gather and analyse the knowledge of experts in adaptive systems for the purpose of identifying software engineering foundations and challenges with regard to the context of this thesis. This provides software engineers with insights and challenges to consider when engineering the acquisition of users' feedback and demonstrate the acceptability of this thesis from industrial and research perspective.

9.2.2 KNOWLEDGE BASE OF USERS' BEHAVIOUR TO FEEDBACK ACQUISITION

The second contribution of this thesis in the knowledge base provided about users' different behaviours and perceptions with regard to feedback acquisition (see Chapter 4). This was done through an empirical study following a Mixed Method Approach (Sequential Exploratory Design) to better understand users' perspectives and behaviours to feedback acquisition. This contribution is meant to enrich the knowledge base for developers and researchers in user-centric, or crowd-centric, adaptation. It also highlights areas of study for a future research in the area.

9.2.3 INVESTIGATION OF THE ROLE OF CULTURE ON USERS' BEHAVIOUR TO FEEDBACK

The third contribution of this thesis is the investigation of the cultural role and social factors in motivating Users to Give Feedback (see Chapter 5). This was done through the conduct of 4 focus groups with users coming from two different cultural backgrounds (Middle Eastern users and European users) to investigate the core social factors that influence their willingness to give feedback and how these factors culturally affect them. This is meant to provide software engineers with more knowledge about the culture role in affecting users' behaviours to feedback acquisition and how to cater for such role when designing the acquisition process.

9.2.4 BUILDING CONSTRUCTS OF PERSONAS OF USERS' BEHAVIOUR TO FEEDBACK

The fourth contribution of this thesis is providing software engineers with the building constructs for the creation of personas for a socially-aware and adaptive feedback acquisition. These building constructs are meant to act as a guide to help software engineers creating (from scratch) or modifying personas with regard to feedback acquisition (see Table 19). For example, developing a socially-aware and adaptive feedback acquisition for users' with certain disabilities/needs (e.g. dementia patients) could require the creation of new personas that could represent differences among their behaviour. In addition, this thesis provides seven personas that represent the different types of users' behaviours with regard to feedback acquisition in software applications.

9.2.5 THE DEVELOPMENT OF A SYSTEMATIC AND ADAPTIVE FEEDBACK ACQUISITION

The fifth and main contribution of this research is the development of a Persona-based Adaptive Feedback Acquisition method (PAFA) (see Chapters 7 and 8). PAFA provides software engineers with an adaptive and systematic way for the acquisition of users' feedback which goes in line with the overall aim of this thesis. This ultimately improves users' satisfaction, feedback quality and software success.

9.2.6 BRIDGING THE GAP BETWEEN SOFTWARE ENGINEERING AND USER EXPERIENCE

The seventh contribution of this research is how it bridges the gap between software engineering and user experience design. This is evident in the novelty and power of PAFA that comes from its ability to combine powerful software engineering and user experience methods to systematically inform the design of an adaptive feedback acquisition. This reduces the gap between personas and their impact on the design, increases traceability and shows the real power of personas as a design tool.

9.3 FUTURE WORK

Figure 72 shows a collective and comprehensive view of the design elements for an adaptive feedback acquisition deduced from on the previous studies conducted throughout this research (see Chapters 3, 4, 5). It suggests that software engineers should not look at the design of feedback acquisition from the users' side only (e.g. motivations to give feedback) but also from the software side (e.g. users' usage of the software). Monitoring how users use the software (e.g. through implicit or explicit collection of their usage data) could give indications about their behaviour to feedback acquisition. For example, users who have never responded to feedback requests in the past (based on their previous behaviour with feedback requests) could be mapped to a certain persona that represent this specific behaviour. This data and statistics could also help deciding when to ask for feedback. For example, when the data shows that the user has already used a certain feature of the software for a certain amount of time then a feedback request should be triggered.

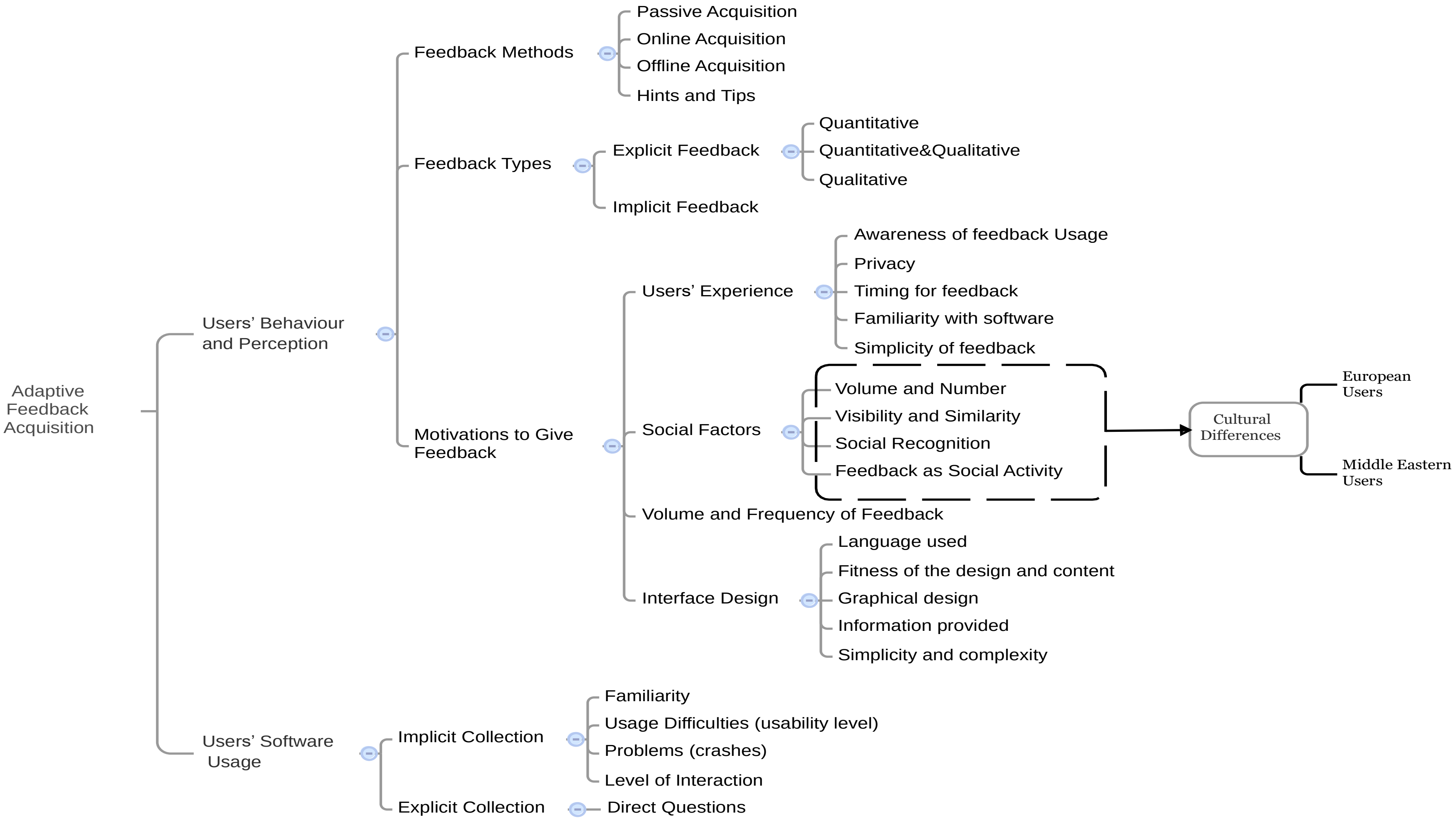


FIGURE 71 A VISUAL VIEW OF THE DESIGN ELEMENTS OF AN ADAPTIVE FEEDBACK ACQUISITION.

In future work, the power of PAFA could be further maximized by encapsulating more of these design elements presented in Figure 65. For example, the seven developed personas (see Chapter 6) could be enriched with more details about each persona's preferences on the graphical design of the feedback request. Another example could be the use of users' usage data and statistic in conjunction with the personality question (see Chapter 7) to more accurately map users' to their potential persona. This opens the gate for more research and studies to further investigate how PAFA could encapsulate such design elements. This will eventually further improve the design process of feedback acquisition (guided by PAFA) hence users' experience, feedback quality and the software success.

9.4 SUMMARY

This research was motivated by the need for an adaptive and systematic engineering approach for the design of feedback acquisition in socially-adaptive systems due to the lack of such an approach. Having an adaptive and systematic acquisition of user' feedback can positively improve users' satisfaction, feedback quality and the success of socially-adaptive systems. Poorly-designed feedback acquisition can harm the collected feedback quality, users' experience and software's success (Ko et al. 2011; Pagano and Bruegge 2013). To reach the aim of this thesis, several empirical studies and software engineering techniques were conducted/used to collectively help reaching this aim; the design of an adaptive and systematic feedback acquisition. This also led to the answering of the research questions raised by this thesis as follows:

- **Research question 1: from the perspective of both software developers and end-users, what are the engineering foundations and challenges for empowering the role of users' feedback in software systems?**

  In order to answer this question, an Expert Opinion was conducted (see Chapter 3) to highlight the engineering challenges and foundations in the area of feedback acquisition for socially-adaptive systems. The findings of this study were synthesized from a two-phase Expert Survey of 29 experts in the first phase and 26 experts in the second. Generally, the consensus among experts is that Social Adaptation is a highly beneficial concept to both developers and clients of self-adaptive systems. However, enabling Social Adaptation is a technically challenging process due to the lack of models and mechanisms for enabling such a concept. Engineering approaches are highly needed for Social Adaptation to empower users' involvement in shaping adaptation decisions and to systematically develop the feedback collection process and interaction styles as well as feedback mining. The study has highlighted research challenges in the areas of providing an enabling platform for Social Adaptation and the design of adaptive feedback acquisition mechanisms that fits users' contexts of use and different behaviours.

- **Research question 2: How do users differ in their behaviours and what is the effect of cultural background with respect to feedback acquisition?**

To take a step further towards the aim of this thesis, a user's point of view was needed to complement the expert's one and answer the second question of this thesis. A sequential mixed method study to explore and investigate users' behaviour with regard to feedback acquisition in software applications was conducted (see Chapter 4). The results suggested that users' behaviour with regard to feedback acquisition highly varies and is influenced by a number of behavioural factors. In line with the Expert Opinion, the results also suggest that systematic approaches and mechanisms to conduct an adaptive feedback acquisition are highly needed. These approaches and mechanisms should fit and adapt to each different user type and should highly consider the factors that influence users' behaviour during the feedback acquisition process. Availability of such systematic approaches for an adaptive feedback acquisition can greatly improve the quality of users' feedback, users' satisfaction and the quality of socially-adaptive software

In addition, a follow up (to the sequential mixed method study) qualitative study was also conducted with Middle Eastern and European users to investigate the role of culture in affecting their behaviours to feedback acquisition (see Chapter 5). The result of this study suggests that culture plays a noticeable role in how users perceive feedback requests and also suggest the need for an adaptive feedback acquisition to cater for such cultural differences. This can improve the quality of the collected feedback as well as users' satisfactions and response rate to feedback acquisition.

- **Research question 3: How to develop a socially-aware and adaptive feedback acquisition and how can it impact the quality of collected feedback, response rate and users' satisfaction?**

Based on the answers of the previous questions, the thesis employed the concept of persona to integrate users' different behaviours to feedback acquisition in the design process of feedback requests. This resulted in the introduction of the main building construct to develop personas for feedback acquisition as well as the creation of seven personas that represent the common behaviours of users to feedback (see Chapter 6).

This was followed by the introduction of a Persona-based Adaptive Feedback Acquisition method (PAFA) which uses the developed personas as a starting point to inform the design of feedback acquisition (see Chapter 7). Led by Personas, PAFA goes through five phases (personas to scenarios, scenarios to goal models, goal models to use-cases, adaptation engineering and evolvement). These phases are motivated by the need to adaptively consider diversity in users' behaviours to feedback acquisition as presented in Figure 28.

In addition, a two-phase style method for evaluating PAFA (a Persona-based Adaptive Feedback Acquisition) was introduced using real case studies (see Chapter 8). The evaluation study indicates that PAFA offers a notable benefits and potential to improve the process of feedback acquisition in software applications and enable a systematic, user-centric and adaptive way for collecting users' feedback. This can ultimately lead to a better users' satisfaction, feedback quality and software success. In general, PAFA scored reasonably high with regard to the proposed evaluation measures (e.g. usability, efficiency, reliability, etc). This indeed shows the potential of PAFA and how it could positively impact the practice of feedback acquisition in software applications.

10. REFERENCES

Abdelnour-Nocera, J., Clemmensen, T. and Kurosu, M., 2013. Reframing HCI Through Local and Indigenous Perspectives. *International Journal of Human-Computer Interaction*, 29 (4), 201-204.

Abras, C., Maloney-Krichmar, D. and Preece, J., 2004. User-centered design. *Bainbridge, W. Encyclopedia of Human-Computer Interaction. Thousand Oaks: Sage Publications*.

Achour, C. B., 1999. Guiding Scenario Authoring1. *Information Modelling and Knowledge Bases X*, 51, 152.

Adolphs, R., 2002. Recognizing emotion from facial expressions: Psychological and neurological mechanisms. *Behavioral and cognitive neuroscience reviews*, 1 (1), 21-62.

Adomavicius, G. and Tuzhilin, A., 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17 (6), 734-749.

Ali, R., Dalpiaz, F. and Giorgini, P., 2010. A goal-based framework for contextual requirements modeling and analysis. *Requirements Engineering*, 15 (4), 439-458.

Ali, R., Solis, C., Omoronyia, I., Salehie, M. and Nuseibeh, B., 2012. Social adaptation: When software gives users a voice. *7th International Conference on Evaluation of Novel Approaches to Software Engineering*, Wrocław, Poland. Available from: http://eprints.gla.ac.uk/70756/ [Accessed

Ali, R., Solis, C., Salehie, M., Omoronyia, I., Nuseibeh, B. and Maalej, W., 2011a. Social sensing: When users become monitors. *19th ACM SIGSOFT symposium and the 13th European conference on Foundations of software engineering*, Szeged, Hungary. Available from: http://eprints.gla.ac.uk/70757/ [Accessed

Ali, R., Solis, C., Salehie, M., Omoronyia, I., Nuseibeh, B. and Maalej, W., 2011b. Social sensing: When users become monitors. *Esec/fse 2011*, Szeged, Hungary. Available from: http://eprints.gla.ac.uk/70757/ [Accessed

Alsabti, K., Ranka, S. and Singh, V., 1998. An efficient k-means clustering algorithm. *Proceedings of the 11th International Parallel Processing Symposium*.

Andersson, C. and Runeson, P., 2007. A spiral process model for case studies on software quality monitoring—method and metrics. *Software Process: Improvement and Practice*, 12 (2), 125-140.

Angrosino, M. V. and Mays de Pérez, K. A., 2000. Rethinking observation: From method to context. *Handbook of qualitative research*, 2, 673-702.

Arthur, J. D., Nance, R. E. and Henry, S. M., 1986. A procedural approach to evaluating software development methodologies: the foundation.

Aykin, N., Quaet-Faslem, P. H. and Milewski, A. E., 2006. Cultural ergonomics. *Handbook of Human Factors and Ergonomics, Third Edition*, 177-190.

Bajic, D. and Lyons, K., 2011. Leveraging social media to gather user feedback for software development, *Proceedings of the 2nd international workshop on Web 2.0 for software engineering* (pp. 1-6): ACM.

Baker, S. E. and Edwards, R., 2012. *How many qualitative interviews is enough? Expert voices and early career reflections on sampling and cases in qualitative research* [Working Paper].

Barber, W. and Badre, A., 1998. Culturability: The merging of culture and usability, *Proceedings of the 4th Conference on Human Factors and the Web* (pp. 1-14).

Barrick, M. R. and Mount, M. K., 1991. The big five personality dimensions and job performance: a meta‐analysis. *Personnel psychology*, 44 (1), 1-26.

Basili, V. R., 1992. Software modeling and measurement: the Goal/Question/Metric paradigm.

Basili, V. R., Selby, R. W. and Hutchens, D. H., 1986. Experimentation in software engineering. *Software Engineering, IEEE Transactions on*, (7), 733-743.

Basu-Zharku, I., 2011. Effects of collectivistic and individualistic cultures on imagination inflation in Eastern and Western cultures. *Student Pulse*, 3 (2), 1-15.

Batory, D., 2005. *Feature models, grammars, and propositional formulas*. Springer.

Bekker, M. and Long, J., 2000. User Involvement in the Design of Human—Computer Interactions: Some Similarities and Differences between Design Approaches. *People and Computers XIV—Usability or Else!* : Springer, 135-147.

Bernard, H. R. and Gravlee, C. C., 2014. *Handbook of methods in cultural anthropology*. Rowman & Littlefield.

Bettenburg, N., Just, S., Schröter, A., Weiss, C., Premraj, R. and Zimmermann, T., 2008. What makes a good bug report?, *Proceedings of the 16th ACM SIGSOFT International Symposium on Foundations of software engineering* (pp. 308-318): ACM.

Beyer, H. and Holtzblatt, K., 1997. *Contextual design: defining customer-centered systems*. Elsevier.

Blomquist, Å. and Arvola, M., 2002. Personas in action: ethnography in an interaction design team, *Proceedings of the second Nordic conference on Human-computer interaction* (pp. 197-200): ACM.

Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, F. and Mylopoulos, J., 2004. Tropos: An agent-oriented software development methodology. *Autonomous Agents and Multi-Agent Systems*, 8 (3), 203-236.

Briggs, K. C., 1976. *Myers-Briggs type indicator*. Consulting Psychologists Press Palo Alto, CA.

Brun, Y., Di Marzo Serugendo, G., Gacek, C., Giese, H., Kienle, H., Litoiu, M., Müller, H., Pezzè, M. and Shaw, M., 2009. Engineering self-adaptive systems through feedback loops. *Software Engineering for Self-Adaptive Systems*, 48-70.

Bødker, S., 2000. Scenarios in user-centred design—setting the stage for reflection and action. *Interacting with computers*, 13 (1), 61-75.

Cakir, H., Bichelmeyer, B. A. and Cagiltay, K., 2005. Effects of cultural differences on e-mail communication in multicultural environments. *Electronic Journal of Communication/La Revue Electronique de Communication*, 15 (1&), 2.

Callahan, E., 2005. Interface design and culture. *Annual review of information science and technology*, 39 (1), 255-310.

Canossa, A. and Drachen, A., 2009. Play-personas: behaviours and belief systems in user-centred game design. *Human-Computer Interaction–INTERACT 2009.* Springer, 510-523.

Carroll, J. M., 2000. *Making use: scenario-based design of human-computer interactions*. MIT press.

Castro, J., Kolp, M. and Mylopoulos, J., 2002. Towards requirements-driven information systems engineering: the Tropos project. *Information systems*, 27 (6), 365-389.

Cattell, R. B., 1966. The scree test for the number of factors. *Multivariate behavioral research*, 1 (2), 245-276.

Chang, Y.-n., Lim, Y.-k. and Stolterman, E., 2008. Personas: from theory to practices, *Proceedings of the 5th Nordic conference on Human-computer interaction: building bridges* (pp. 439-442): ACM.

Chau, P. Y., Cole, M., Massey, A. P., Montoya-Weiss, M. and O'Keefe, R. M., 2002. Cultural differences in the online behavior of consumers. *Communications of the ACM*, 45 (10), 138-143.

Cheng, B. H., de Lemos, R., Giese, H., Inverardi, P., Magee, J., Andersson, J., Becker, B., Bencomo, N., Brun, Y. and Cukic, B., 2009. *Software engineering for self-adaptive systems: A research roadmap*. Springer.

Clements, P. and Northrop, L., 2002. *Software product lines*. Addison-Wesley.

Clements, P. C., 1981. *Function Specifications for the A-7E Function Driver Module*. DTIC Document.

Cooke, R. M. and Probst, K. N., 2006. *Highlights of the expert judgment policy symposium and technical workshop*. Resources for the Future Washington, DC.

Cooper, A., 1999a. *The inmates are running the asylum:[Why high-tech products drive us crazy and how to restore the sanity].* Vol. 261. Sams Indianapolis.

Cooper, A., 1999b. The Inmates are Running the Asylum–Why High-Tech Products Drive Us Crazy and How 2 to Restore the Sanity. *Sams. isbn: 0-67231-649-8*.

Courage, C. and Baxter, K., 2005. *Understanding your users: A practical guide to user requirements methods, tools, and techniques.* Gulf Professional Publishing.

Creswell, J. W., 2013. *Qualitative Inquiry And Research Design: Choosing Among Five Approaches*. 3rd edition.: SAGE Publications, Inc

Creswell, J. W. and Clark, V. L. P., 2011. *Designing and conducting mixed methods research*. Thousand

Oaks, CA: Sage Publications, Inc.

Creswell, J. W., Plano Clark, V. L., Gutmann, M. L. and Hanson, W. E., 2003. *Advanced mixed methods research designs*. Thousand Oaks, CA: Sage.

Czarnecki, K. and Eisenecker, U. W., 2000. Generative programming. *Edited by G. Goos, J. Hartmanis, and J. van Leeuwen*, 15.

Dalpiaz, F., Ali, R. and Giorgini, P., 2012. Aligning software configuration with business and IT context, *Advanced Information Systems Engineering* (pp. 206-221): Springer.

Davis, A. M., 1993. *Software requirements: objects, functions, and states*. Prentice-Hall, Inc.

Deterding, S., Dixon, D., Khaled, R. and Nacke, L., 2011. From game design elements to gamefulness: defining gamification, *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments* (pp. 9-15): ACM.

DeWalt, K. M. and DeWalt, B. R., 2010. *Participant observation: A guide for fieldworkers*. Rowman Altamira.

Differding, C., Hoisl, B. and Lott, C. M., 1996. *Technology package for the goal question metric paradigm*. Fachbereich Informatik, Univ.

do Prado Leite, J. C. S., Rossi, G., Balaguer, F., Maiorana, V., Kaplan, G., Hadad, G. and Oliveros, A., 1997. Enhancing a requirements baseline with scenarios. *Requirements Engineering*, 2 (4), 184-198.

Doll, W. J. and Torkzadeh, G., 1988. The measurement of end-user computing satisfaction. *MIS quarterly*, 12 (2).

Dubois, D. J. and Tamburrelli, G., 2013. Understanding gamification mechanisms for software development, *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering* (pp. 659-662): ACM.

Easterbrook, S., Singer, J., Storey, M.-A. and Damian, D., 2008. Selecting empirical methods for software engineering research. *Guide to advanced empirical software engineering.* Springer, 285-311.

Elkhodary, A., Esfahani, N. and Malek, S., 2010. FUSION: a framework for engineering self-tuning self-adaptive software systems, *Proceedings of the eighteenth ACM SIGSOFT international symposium on Foundations of software engineering* (pp. 7-16): ACM.

Esfahani, N. and Malek, S., 2010. Social computing networks: a new paradigm for engineering self-adaptive pervasive software systems, *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 2* (pp. 159-162): ACM.

Eysenck, H. J., 1950. *Dimensions of personality*. Vol. 5. Transaction Publishers.

Faily, S. and Fléchais, I., 2014. Eliciting and visualising trust expectations using persona trust characteristics and goal models, *Proceedings of the 6th International Workshop on Social Software Engineering* (pp. 17-24): ACM.

Fickas, S. and Feather, M. S., 1995. Requirements monitoring in dynamic environments, *Requirements Engineering, 1995., Proceedings of the Second IEEE International Symposium on* (pp. 140-147): IEEE.

Fischer, G., 2001. User modeling in human–computer interaction. *User modeling and user-adapted interaction*, 11 (1-2), 65-86.

Fowler, M., 2004. *UML distilled: a brief guide to the standard object modeling language*. Addison-Wesley Professional.

Frandsen-Thorlacius, O., Hornbæk, K., Hertzum, M. and Clemmensen, T., 2009. Non-universal usability?: a survey of how usability is understood by Chinese and Danish users, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 41-50): ACM.

Franklin, S., Walker, C. and Canada, S., 2003. *Survey methods and practices* [online]. Ottawa : Statistics Canada, c2010.

Gaffney Jr, J. E., 1981. Metrics in software quality assurance, *Proceedings of the ACM'81 conference* (pp. 126-130): ACM.

Garlan, D., Cheng, S.-W., Huang, A.-C., Schmerl, B. and Steenkiste, P., 2004. Rainbow: Architecture-based self-adaptation with reusable infrastructure. *Computer*, 37 (10), 46-54.

Gill, P., Stewart, K., Treasure, E. and Chadwick, B., 2008. Methods of data collection in qualitative research: interviews and focus groups. *British dental journal*, 204 (6), 291-295.

Gold, R. L., 1958. Roles in sociological field observations. *Social forces*, 217-223.

Goodwin, K., 2001. Perfecting your personas. *Cooper Interaction Design Newsletter*.

Gordon, T. J., 1994. The delphi method. *Futures research methodology*, 1-33.

Grudin, J., 1991a. Interactive systems: Bridging the gaps between developers and users. *Computer*, 24 (4), 59-69.

Grudin, J., 1991b. Systematic sources of suboptimal interface design in large product development organizations. *Human-Computer Interaction*, 6 (2), 147-196.

Grudin, J. and Pruitt, J., 2002. Personas, participatory design and product development: An infrastructure for engagement, *PDC* (pp. 144-152).

Guðjónsdóttir, R. and Lindquist, S., 2008. Personas and scenarios: Design tool or a communication device? *Proceedings of COOP', Carry Le Rouet, France*.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. and Witten, I. H., 2009. The WEKA data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11 (1), 10-18.

Hallsteinsen, S., Hinchey, M., Park, S. and Schmid, K., 2008. Dynamic software product lines. *Computer*, 41 (4), 93-95.

Hamari, J., 2013. Transforming homo economicus into homo ludens: A field experiment on gamification in a utilitarian peer-to-peer trading service. *Electronic commerce research and applications*, 12 (4), 236-245.

Hamari, J. and Eranti, V., 2011. Framework for designing and evaluating game achievements. *Proc. DiGRA 2011: Think Design Play*, 115, 122-134.

Hennig-Thurau, T., Gwinner, K. P., Walsh, G. and Gremler, D. D., 2004. Electronic word-of-mouth via consumer-opinion platforms: what motivates consumers to articulate themselves on the internet? *Journal of interactive marketing*, 18 (1), 38-52.

Hofstede, G., 1991. Cultures and organizations: Software of the mind. *New York: McGrawHill*, 1997, 23-47.

Hofstede, G., 2002. Dimensions do not exist: A reply to Brendan McSweeney. *Human relations*, 55 (11), 1355-1361.

Hofstede, G., 2011. Dimensionalizing cultures: The Hofstede model in context. *Online readings in psychology and culture*, 2 (1), 8.

Honold, P., 2000. Culture and context: An empirical study for the development of a framework for the elicitation of cultural influence in product usage. *International Journal of Human-Computer Interaction*, 12 (3-4), 327-345.

Hothi, J. and Hall, W., 1998. An evaluation of adapted hypermedia techniques using static user modelling, *Proceedings of the second workshop on adaptive hypertext and hypermedia* (pp. 45-50).

Huotari, K. and Hamari, J., 2012. Defining gamification: a service marketing perspective, *Proceeding of the 16th International Academic MindTrek Conference* (pp. 17-22): ACM.

Höst, M. and Runeson, P., 2007. Checklists for Software Engineering Case Study Research, *ESEM* (pp. 479-481).

Idoughi, D., Seffah, A. and Kolski, C., 2012. Adding user experience into the interactive service design loop: a persona-based approach. *Behaviour & Information Technology*, 31 (3), 287-303.

Jackson, M. A., 1975. *Principles of program design*. Vol. 123790506. IMA.

Jacobson, I., 1992. Object oriented software engineering: a use case driven approach.

Jeary, S., Musial, K. and Phalp, K. T., 2011. Exploring the requirements process for a complex, adaptive system in a high risk software development environment.

Jian, L., MacKie-Mason, J. K., Chiao, B., Levchenko, A., Zellner, A., Kmenta, J., Dreze, J. and Oberhofer, W., 2012. Incentive-centered design for user-contributed content. *The Oxford Handbook of the Digital Economy, Oxford University Press Oxford*, 399-433.

Johnson, A. and Taatgen, N., 2005. User modeling. *The handbook of human factors in web design*, 424-438.

Johnson, R. B. and Onwuegbuzie, A. J., 2004. Mixed methods research: A research paradigm whose time has come. *Educational researcher*, 33 (7), 14-26.

Johnson, R. B., Onwuegbuzie, A. J. and Turner, L. A., 2007. Toward a Definition of Mixed Methods Research. *Journal of Mixed Methods Research*, 1 (2), 112-133.

Jones, S. and Maiden, N. A., 2005. RESCUE: An integrated method for specifying requirements for complex socio-technical systems. *Requirements Engineering for Socio-Technical Systems*, 245-265.

Jung, C. G., 2014. *Psychological types*. Routledge.

Jung, H.-W., Kim, S.-G. and Chung, C.-S., 2004. Measuring software product quality: A survey of ISO/IEC 9126. *IEEE software*, 21 (5), 88-92.

Kang, K. C., Cohen, S. G., Hess, J. A., Novak, W. E. and Peterson, A. S., 1990. *Feature-oriented domain analysis (FODA) feasibility study*. DTIC Document.

Kang, K. C. and Lee, H., 2013. Variability modeling. *Systems and Software Variability Management.* Springer, 25-42.

Kanstrup, A. M. and Christiansen, E., 2006. Selecting and evoking innovators: combining democracy and creativity, *Proceedings of the 4th Nordic conference on Human-computer interaction: changing roles* (pp. 321-330): ACM.

Karahasanoviæ, A., Anda, B., Arisholm, E., Hove, S. E., Jørgensen, M., Sjøberg, D. I. and Welland, R., 2005. Collecting feedback during software engineering experiments. *Empirical Software Engineering*, 10 (2), 113-147.

Kawulich, B. B., 2005. Participant Observation as a Data Collection Method.

Kitchenham, B., Linkman, S. and Law, D., 1997. DESMET: a methodology for evaluating software engineering methods and tools. *Computing & Control Engineering Journal*, 8 (3), 120-126.

Kitchenham, B., Pickard, L. and Pfleeger, S. L., 1995. Case studies for method and tool evaluation. *IEEE software*, 12 (4), 52-62.

Kitchenham, B. A., 1996. Evaluating software engineering methods and tool—part 3: selecting an appropriate evaluation method—practical issues. *ACM SIGSOFT Software Engineering Notes*, 21 (4), 9-12.

Kitchenham, B. A., Pfleeger, S. L., Pickard, L. M., Jones, P. W., Hoaglin, D. C., El Emam, K. and Rosenberg, J., 2002. Preliminary guidelines for empirical research in software engineering. *Software Engineering, IEEE Transactions on*, 28 (8), 721-734.

Knol, A. B., Slottje, P., Van Der Sluijs, J. P. and Lebret, E., 2010. The use of expert elicitation in environmental health impact assessment: a seven step procedure. *Environmental Health*, 9 (1), 19.

Ko, A. J., Lee, M. J., Ferrari, V., Ip, S. and Tran, C., 2011. A case study of post-deployment user feedback triage, *Proceedings of the 4th International Workshop on Cooperative and Human Aspects of Software Engineering* (pp. 1-8): ACM.

Kroeber, A. L. and Kluckhohn, C., 1952. Culture: A critical review of concepts and definitions. *Papers. Peabody Museum of Archaeology & Ethnology, Harvard University*.

Kujala, S., 2003. User involvement: a review of the benefits and challenges. *Behaviour & information technology*, 22 (1), 1-16.

Kujala, S., Kauppinen, M., Lehtola, L. and Kojo, T., 2005. The role of user involvement in requirements quality and project success, *Requirements Engineering, 2005. Proceedings. 13th IEEE International Conference on* (pp. 75-84): IEEE.

Lee, K., Kang, K. and Lee, J., 2002. Concepts and guidelines of feature modeling for product line software engineering. *Software Reuse: Methods, Techniques, and Tools*, 62-77.

Leung, W.-C., 2001. How to design a questionnaire. *Student BMJ*, 9, 187-189.

Lientz, B. P. and Swanson, E. B., 1980. Software maintenance management: a study of the maintenance of computer applications software in 487 data processing organizations: Addison-Wesley, Reading, MA.

Likert, R., 1932. A technique for the measurement of attitudes. *Archives of psychology*.

Liskov, B. H., 1972. A design methodology for reliable software systems, *Proceedings of the December 5-7, 1972, fall joint computer conference, part I* (pp. 191-199): ACM.

Maalej, W. and Happel, H.-J., 2009. From work to word: How do software developers describe their work?, *Mining Software Repositories, 2009. MSR'09. 6th IEEE International Working Conference on* (pp. 121-130): IEEE.

Maalej, W., Happel, H.-J. and Rashid, A., 2009. When users become collaborators: towards continuous and context-aware user input, *Proceedings of the 24th ACM SIGPLAN conference companion on Object oriented programming systems languages and applications* (pp. 981-990): ACM.

Maalej, W. and Pagano, D., 2011. On the socialness of software, *Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference on* (pp. 864-871): IEEE.

Maiden, N. and Robertson, S., 2005. Developing use cases and scenarios in the requirements process, *Proceedings of the 27th international conference on Software engineering* (pp. 561-570): ACM.

Marcus, A., 2003. Global/intercultural user interface design. *Human-Computer Interaction: Design Issues, Solutions, and Applications*, 355-381.

Marcus, A. and Gould, E. W., 2000. Crosscurrents: cultural dimensions and global Web user-interface design. *interactions*, 7 (4), 32-46.

Melville, P. and Sindhwani, V., 2010. Recommender systems. *Encyclopedia of machine learning.* Springer, 829-838.

Merriam, S. B., 1998. *Qualitative Research and Case Study Applications in Education. Revised and Expanded from" Case Study Research in Education."*. ERIC.

Miaskiewicz, T. and Kozar, K. A., 2011. Personas and user-centered design: how can personas benefit product design processes? *Design Studies*, 32 (5), 417-430.

Michahelles, F., 2009. Business Aspects of the Internet of Things.

Miles, M. B. and Huberman, A. M., 1994. *Qualitative data analysis: An expanded sourcebook*. 2nd edition.: Sage.

Moore, G. A., 2002. Crossing the chasm.

Morse, J. M., 1991. Approaches to qualitative-quantitative methodological triangulation. *Nursing research*, 40 (2), 120-123.

Mulder, S. and Yaar, Z., 2006. *The user is always right: A practical guide to creating and using personas for the web*. New Riders.

Mumford, E. and Weir, M., 1979. *Computer systems in work design--the ETHICS method: effective technical and human implementation of computer systems: a work design exercise book for individuals and groups*. Wiley.

Myers, Isabel Briggs and B, P., 1980. *Gifts differing: Understanding personality type*. Davies-Black Pub.

Myers, I. B., McCaulley, M. H. and Most, R., 1985. *Manual, a guide to the development and use of the Myers-Briggs type indicator*. Consulting Psychologists Press.

Mylopoulos, J., Chung, L. and Yu, E., 1999. From object-oriented to goal-oriented requirements analysis. *Communications of the ACM*, 42 (1), 31-37.

Nardi, B. A., 1993. *A small matter of programming: perspectives on end user computing*.

Nelson, K. and Fivush, R., 2004. The emergence of autobiographical memory: a social cultural developmental theory. *Psychological review*, 111 (2), 486.

Nielsen, J., 1996. International user interfaces.

Nielsen, L., 2002. From user to character: an investigation into user-descriptions in scenarios, *Proceedings of the 4th conference on Designing interactive systems: processes, practices, methods, and techniques* (pp. 99-104): ACM.

Nielsen, L., 2003. A model for personas and scenarios creation. *Roskilde, Denmark 27th November, 2003*, 71.

Nielsen, L. and Storgaard Hansen, K., 2014. Personas is applicable: a study on the use of personas in Denmark, *Proceedings of the 32nd annual ACM conference on Human factors in computing systems* (pp. 1665-1674): ACM.

Nocera, J. A., 2014. The Politics of Co-configuration in Participatory Design: A Technological Frame Perspective.

Norman, D. A. and Draper, S. W., 1986. User centered system design. *New Perspectives on Human-Computer Interaction, L. Erlbaum Associates Inc., Hillsdale, NJ*.

Nzekwa, R., Rouvoy, R. and Seinturier, L., 2010. Modelling feedback control loops for self-adaptive systems, *Third International DisCoTec Workshop on Context-Aware Adaptation Mechanisms for Pervasive and Ubiquitous Services* (pp. 1-6).

O'Brien, C., 2010. Get ready for the decade of gamification. *San Jose Mercury News. http://www. mercurynews. com/ci_16401223*.

olivia, 2012. *Difference Between Observation and Interviewing as Methods of Data Collection | Difference Between* [online]. Available from: http://www.differencebetween.com/difference-between-observation-and-vs-interviewing-as-methods-of-data-collection/ [Accessed 11 Jan 2013].

Oreizy, P., Heimbigner, D., Johnson, G., Gorlick, M. M., Taylor, R. N., Wolf, A. L., Medvidovic, N., Rosenblum, D. S. and Quilici, A., 1999. An architecture-based approach to self-adaptive software. *IEEE Intelligent systems*, 14 (3), 54-62.

Oswald, W. and Schoepfle, G. M., 1987. Systematic fieldwork: Vol. 1. Foundations of ethnography and interviewing: Newbury Park, CA: Sage Publications.

Pagano, D. and Br, B., 2013. User involvement in software evolution practice: a case study. *Proceedings of the 2013 International Conference on Software Engineering*, San Francisco, CA, USA. 2486920: IEEE Press. 953-962.

Pagano, D. and Bruegge, B., 2013. User involvement in software evolution practice: a case study, *Proceedings of the 2013 International Conference on Software Engineering* (pp. 953-962): IEEE Press.

Pagano, D. and Maalej, W., 2013. User Feedback in the AppStore: An Empirical Study, *Proceedings of the 21st International Conference on Requirements Engineering* (Vol. 16, pp. 1.99): IEEE.

Paramythis, A., 2004. Towards Self-Regulating Adaptive Systems, *LWA* (pp. 57-63): Citeseer.

Parnas, D. L., 1976. On the design and development of program families. *Software Engineering, IEEE Transactions on*, (1), 1-9.

Pfeil, U., Zaphiris, P. and Ang, C. S., 2006. Cultural differences in collaborative authoring of Wikipedia. *Journal of Computer-Mediated Communication*, 12 (1), 88-113.

Pohl, K., Böckle, G. and van der Linden, F. J., 2005. *Software product line engineering: foundations, principles and techniques*. Springer Science & Business Media.

Premack, D. and Woodruff, G., 1978. Does the chimpanzee have a theory of mind? *Behavioral and brain sciences*, 1 (04), 515-526.

Pruitt, J. and Grudin, J., 2003. Personas: practice and theory, *Proceedings of the 2003 conference on Designing for user experiences* (pp. 1-15): ACM.

Ralyté, J., Rolland, C. and Plihon, V., 1999. Method enhancement with scenario based techniques, *Advanced information systems engineering* (pp. 103-118): Springer.

Reeves, B. and Read, J. L., 2013. *Total engagement: How games and virtual worlds are changing the way people work and businesses compete*. Harvard Business Press.

Reinecke, K. and Bernstein, A., 2007. Culturally adaptive software: moving beyond internationalization. *Usability and Internationalization. Global and Local User Interfaces.* Springer, 201-210.

Reinecke, K. and Bernstein, A., 2011. Improving performance, perceived usability, and aesthetics with culturally adaptive user interfaces. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 18 (2), 8.

Ricci, F., Rokach, L. and Shapira, B., 2011. *Introduction to recommender systems handbook*. Springer.

Robbins, S. S. and Stylianou, A. C., 2002. A study of cultural differences in global corporate web sites. *Journal of Computer Information Systems*, 42 (2), 3-9.

Robson, C., 2002. *Real world research*. Vol. 2. Blackwell publishers Oxford.

Rogers, P., 2008. Fifty years of intercultural study: A continuum of perspectives for research and teaching.

Rolland, C., Souveyet, C. and Achour, C. B., 1998. Guiding goal modeling using scenarios. *Software Engineering, IEEE Transactions on*, 24 (12), 1055-1071.

Runeson, P. and Höst, M., 2009. Guidelines for conducting and reporting case study research in software engineering. *Empirical software engineering*, 14 (2), 131-164.

Salehie, M. and Tahvildari, L., 2009. Self-adaptive software: Landscape and research challenges. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 4 (2), 14.

Sanders, E. B.-N., 2002. From user-centered to participatory design approaches. *Design and the social sciences: Making connections*, 1-8.

Santander, V. F. and Castro, J. F., 2002. Deriving use cases from organizational modeling, *Requirements Engineering, 2002. Proceedings. IEEE Joint International Conference on* (pp. 32-39): IEEE.

Sawyer, P., Bencomo, N., Whittle, J., Letier, E. and Finkelstein, A., 2010. Requirements-aware systems: A research agenda for re for self-adaptive systems, *Requirements Engineering Conference (RE), 2010 18th IEEE International* (pp. 95-103): IEEE.

Schiele, B., Baker, A. and Hathaway, S., 1943. The Minnesota multiphasic personality inventory. *Journal-Lancet*.

Schmid-Isler, S., 2000. The language of digital genres-a semiotic investigation of style and iconology on the world wide web, *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on* (pp. 9 pp. vol. 2): IEEE.

Schwartz, M. S. and Schwartz, C. G., 1955. Problems in participant observation. *American journal of sociology*, 343-353.

Scott, L. R., 1978. An engineering methodology for presenting software functional architecture, *Proceedings of the 3rd international conference on Software engineering* (pp. 222-229): IEEE Press.

Seaman, C. B., 1999. Qualitative methods in empirical studies of software engineering. *Software Engineering, IEEE Transactions on*, 25 (4), 557-572.

Seffah, A., Kolski, C. and Idoughi, D., 2009. Persona comme outil de design de services interactifs: principes et exemple en e-maintenance, *Proceedings of the 21st International Conference on Association Francophone d'Interaction Homme-Machine* (pp. 333-336): ACM.

Seyff, N., Graf, F. and Maiden, N., 2010. Using mobile re tools to give end-users their own voice, *Requirements Engineering Conference (RE), 2010 18th IEEE International* (pp. 37-46): IEEE.

Sharma, R. S., Conrath, D. W. and Dilts, D. M., 1991. A socio-technical model for deploying expert systems. I. The general theory. *Engineering Management, IEEE Transactions on*, 38 (1), 14-23.

Sheppard, C. and Scholtz, J., 1999. The effects of cultural markers on web site use, *Proceedings of the 5th Conference on Human Factors and the Web* (Vol. 3).

Shull, F., Basili, V., Carver, J., Maldonado, J. C., Travassos, G. H., Mendonça, M. and Fabbri, S., 2002. Replicating software engineering experiments: addressing the tacit knowledge problem, *Empirical Software Engineering, 2002. Proceedings. 2002 International Symposium n* (pp. 7-16): IEEE.

Silva Souza, V. E., Lapouchnian, A., Robinson, W. N. and Mylopoulos, J., 2011. Awareness requirements for adaptive systems, *Proceedings of the 6th international symposium on Software engineering for adaptive and self-managing systems* (pp. 60-69): ACM.

Simmonds, J. and Bastarrica, M. C., 2011. Modeling variability in software process lines. *Departamento de Ciencias de la Computación. Universidad de Chile*.

Singh, N., Zhao, H. and Hu, X., 2005. Analyzing the cultural content of web sites: A cross-national comparision of China, India, Japan, and US. *International Marketing Review*, 22 (2), 129-146.

Skillman, G. D., 1999. Intergenerational conflict within the family context: A comparative analysis of collectivism and individualism within Vietnamese, Filipino, and Caucasian families.

Smadja, F., 2009. Mixing financial, social and fun incentives for social voting. *World Wide Web Internet And Web Information Systems*.

Socialclinic, 2013. *The State of Social Media in Saudi Arabia 2012* [online]. The Social Clinic. Available from: http://www.thesocialclinic.com/the-state-of-social-media-in-saudi-arabia-2012-2/ [Accessed 24-09].

Spool, J. M., 2007. *Making Personas Work for Your Web Site: An Interview with Steve Mulder* [online]. Available from: http://www.uie.com/articles/mulder_interview/ [Accessed 28 November].

Spradley, J. P. and Baker, K., 1980. *Participant observation*. Vol. 195. Holt, Rinehart and Winston New York.

Stefan, T. and Ashok, N., 1998. Note on lead user research Harvard Business Online.

Stevens, M., 2010. *Chapter 7: Selected Qualitative Methods: Why Use Qualitative Methods* [online]. Painconsortium.nih.gov. Available from: http://painconsortium.nih.gov/symptomresearch/chapter_7/sec2/cmss2pg1.htm [Accessed 12 Jan 2013].

Stoddard II, R. W., 1999. Method for estimating software operation and performance using a goal-question-metric paradigm: Google Patents.

Sturm, C., Oh, A., Linxen, S., Abdelnour Nocera, J., Dray, S. and Reinecke, K., 2015. How WEIRD is HCI?: Extending HCI Principles to other Countries and Cultures, *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems* (pp. 2425-2428): ACM.

Suler, J., 2004. The online disinhibition effect. *Cyberpsychology & behavior*, 7 (3), 321-326.

Tait, P. and Vessey, I., 1988. The effect of user involvement on system success: a contingency approach. *MIS quarterly*, 91-108.

Tashakkori, A. and Teddlie, C., 1998. *Mixed methodology: Combining qualitative and quantitative approaches*. Vol. 46. SAGE Publications, Incorporated.

Terveen, L. and Hill, W., 2001. Beyond recommender systems: Helping people help each other. *HCI in the New Millennium*, 1, 487-509.

Tesauro, G., Jong, N. K., Das, R. and Bennani, M. N., 2006. A hybrid reinforcement learning approach to autonomic resource allocation, *Autonomic Computing, 2006. ICAC'06. IEEE International Conference on* (pp. 65-73): IEEE.

Tessmer, M., 1993. *Planning and conducting formative evaluations: Improving the quality of education and training*. Psychology Press.

Triandis, H. C., 1993. Collectivism and individualism as cultural syndromes. *Cross-Cultural Research*, 27 (3-4), 155-180.

Tsikriktsis, N., 2002. Does culture influence web site quality expectations? An empirical study. *Journal of service research*, 5 (2), 101-112.

Ulrich, K. T. and Eppinger, S. D., 2007. *Product design and development*. McGraw-Hill New York.

van Riemsdijk, M. B., 2013. ***Socially adaptive software*** [online]. Available from: http://www.awareness-mag.eu/view.php?article=004666-2013-02-07&category=Artificial+Intelligence [Accessed 10 June 2013].

Van Solingen, R. and Berghout, E., 1999. *The Goal/Question/Metric Method: a practical guide for quality improvement of software development*. Vol. 40. McGraw-Hill London.

Wash, R. and MacKie-Mason, J. K., 2006. Incentive-centered design for information security, *1st Conference on USENIX Workshop on Hot Topics in Security* (Vol. 1).

Weber, J. L., 2010a. *Privacy and Security Attitudes, Beliefs and Behaviours: Informing Future Tool Design* [http://hdl.handle.net/10012/5391]. (Master of Mathematics). University of Waterloo.

Weber, J. L., 2010b. *Privacy and Security Attitudes, Beliefs and Behaviours: Informing Future Tool Design.*

Whittle, J., Simm, W. and Ferrario, M.-A., 2010. On the role of the user in monitoring the environment in self-adaptive systems: a position paper, *Proceedings of the 2010 ICSE Workshop on Software Engineering for Adaptive and Self-Managing Systems* (pp. 69-74): ACM.

Whyte, W. F., 1979. On making the most of participant observation. *The American Sociologist*, 56-66.

Williams, A., 2003. How to… Write and analyse a questionnaire. *Journal of Orthodontics*, 30 (3), 245-252.

Wirth, N., 1971. Program development by stepwise refinement. *Communications of the ACM*, 14 (4), 221-227.

Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B. and Wesslén, A., 2012. *Experimentation in software engineering*. Springer Science & Business Media.

Wong, E. and Tate, G., 1994. A study of user participation in information systems development. *Journal of Information Technology*, 9, 51-51.

Yin, R. K., 2013. *Case study research: Design and methods*. Sage publications.

Yu, E., 2011. Modelling strategic relationships for process reengineering. *Social Modeling for Requirements Engineering*, 11, 2011.

Yu, E. and Mylopoulos, J., 1998. Why goal-oriented requirements engineering, *Proceedings of the 4th International Workshop on Requirements Engineering: Foundations of Software Quality* (Vol. 15).

Yu, E. S., 1997. Towards modelling and reasoning support for early-phase requirements engineering, *Requirements Engineering, 1997., Proceedings of the Third IEEE International Symposium on* (pp. 226-235): IEEE.

Yu, Y., Lapouchnian, A., Liaskos, S., Mylopoulos, J. and Leite, J. C., 2008. From goals to high-variability software design. *Foundations of Intelligent Systems.* Springer, 1-16.

Yunker, J., 2002. *Beyond borders: Web globalization strategies*. New Riders.

Zichermann, G. and Cunningham, C., 2011. *Gamification by design: Implementing game mechanics in web and mobile apps.* " O'Reilly Media, Inc.".

Zwitch, R., 2013. *Clustering Search Keywords Using K-Means Clustering* [online]. R-bloggers: Available from: http://www.r-bloggers.com/clustering-search-keywords-using-k-means-clustering/ [Accessed 9 Dec 2013].

11. APPENDICES

11.1 APPENDIX 1

First-Phase Expert Survey on Socially-Adaptive Software

Expert Questionnaire Introduction:

Dear Participant:

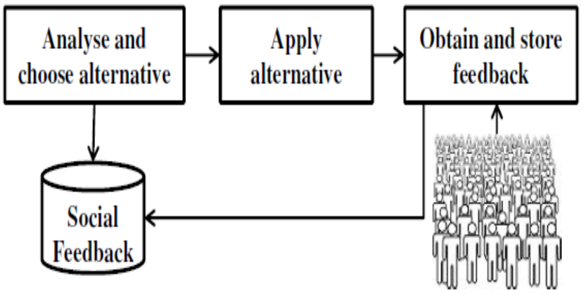
Thank you for helping me out. .

My name is Malik Al Maliki and I am a PhD student at Bournemouth University. I'm interested in gathering experts' opinions in order to identify the potential core issues in the area of feedback acquisition for socially-driven software adaptation. We mean by socially driven software adaptation, the software system that is able to collect and analyse users' explicit feedback on quality at run-time in order to choose a behaviour that is shown to be the best quality in a certain context. If feedback of users shows that an alternative behaviour of the system is being collectively judged by user to be the most appropriate alternative then the system will adopt that alternative in order to maximize users' satisfaction.

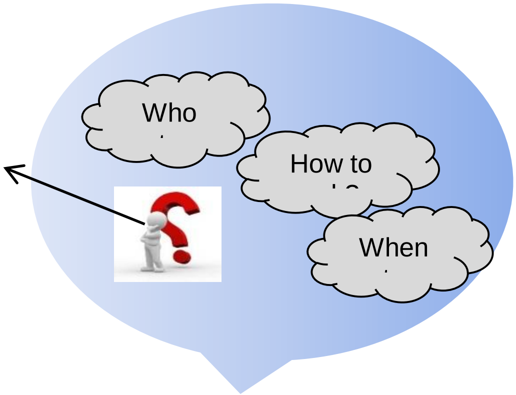
Example: MoDriver is mobile software designed to guide drivers. MobDrive has two ways of delivering information to users: video-based and interactive. Please remember that a realistic number of behaviours grows exponentially if you consider the refinement of these two macro behaviours. As designers we do not know which behaviour to adopt. We know that certain contextual factors like Skills in using Mobile devices, the screen size, whether a driver is driving or stopping, age, etc. could influence the a driver judgement of the quality (readability of fonts, ability to hear the sound, etc.) but we do not know exactly how and how this evolves as well. Our solution is to obtain the feedback from drivers when using each of the behaviour, monitor the context and record all that. When MobDrive has to decide the default way to deliver information, it will analyse the data recorded and infer it.

Data Protection

Bournemouth University is a registered Data Controller. Any information that you supply will be held anonymously and securely in accordance with the Data Protection Act 1998 and will only be used for the purposes of this survey. Your personal details will not be made available outside the University. Any queries regarding Data Protection should be addressed to the Information Officer at information@bournemouth.ac.uk. Cookies, personal data stored by your Web browser, are not used in this survey.



Social Adaptation Loop (Ali et al. 2012)



Our research focus

I'm also interested in having your opinion about the value and socio-technical impact of the idea of having an adaptive mechanism to collect users' feedback in these socially driven software systems. And what kind of benefits could we gain from having such a mechanism. This adaptive acquisition of feedback can adapt itself to different type of users depending on their behaviour and context when providing feedback (e.g. asking for feedback at the user's preferable time.).

Because of your expertise in domains that potentially have a strong relationship with my research, I am inviting you to complete the following questionnaire. Your participation will be very helpful and will assist me in my educational endeavour.

The following questionnaire will require approximately …..to complete.

Thank you for taking the time to complete this questionnaire and the data collected will provide useful information and will be kept confidential and used for research purposes only.

If you have any questions or if you are interested in learning about the results of this study, please email me and I will keep you up to date on the project progress.

Malik Al Maliki
malmaliki@bournemouth.ac.uk

General Information

1. Please enter your first and last name

2. What is the highest degree you have received?

○ Bachelor Degree
○ MSc Degree
○ PhD Degree
○ Other (please specify) ____________________

3. Your research interests (please insert text)

4. Number of years working on dynamic/adaptive aspects of software systems (please insert numbers)

5. Please indicate your level of expertise in relation to adaptive/dynamic aspects of software systems

	low (e.g., no direct experience, anecdotal knowledge only)	medium (e.g., some direct experience, but wide reading)	high (e.g., primary focus of my professional work)
Adaptive/Dynamic aspects of Software Systems	○	○	○

Familiarization with the Topic

6. How would you rate the benefits of Social Adaptation for developers of adaptive software ?

	low (It has no notable value)	medium (reasonable value)	high (great value)
Social Adaptation	○	○	○

7. How would you rate the benefits of Social Adaptation for clients of adaptive software. By client we mean end-users, system administrators, business managers, etc.?

	low (It has no notable value)	medium (reasonable value)	high (great value)
Social Adaptation	○	○	○

8. Can you enumerate some benefits of Social Adaptation for developers of adaptive software? (please insert text)

9. Can you enumerate some benefits of Social Adaptation for clients of adaptive software? By client we mean end-users, system administrators, business managers, etc.? (please insert text)

10. Can you nominate certain areas and application domains where Social Adaptation has a distinguished usefulness? (please insert text)

11. From your point of view, what are the areas, application domains and situations where we should NOT use Social Adaptation? (please insert text)

12. Knowing that relying on software to socially-adapt in an autonomous way is a user’s choice, do you think that there are cases where software should still ask users to confirm its adaptation decision?

○ Yes, please give some examples ____________________
○ No, can you explain why? ____________________

13. How would you rate the value of allowing users to validate/confirm Social Adaptation decisions

	low (it has no notable value)	medium (reasonable value)	high (great value)
Validation through users feedback	○	○	○

Our Main Topic

14. Do you think that developing software-based feedback acquisition is technically challenging?

○ Yes, please mention some challenges? ____________________
○ No, please explain why? ____________________

15. Do you think the quality of collected feedback can be affected by the way it is collected (i.e. the time to ask, the user interface used, the language of the question, etc.)?

○ Yes, can you explain how? ____________________
○ No, can you explain why? ____________________

16. How would you rate the value of having an adaptive way to collect feedback from users (e.g. software asks for a brief feedback using a quick and simple dialogue when user is busy and vice versa)?

	low (it has no notable value)	medium (reasonable value)	high (great value)
Adaptive acquisition of feedback	○	○	○

17. If we have an adaptive acquisition of users feedback, what should be the drivers of adaptation, i.e. the factors to take into account when deciding the What, When, Where, How, Whom, How Much in getting feedback? (Please insert text)

18. From your point of view, what would be the main reasons for users to give feedback? (please insert text)

19. From your point of view, what would be the main reasons for users to ignore feedback requests? (Please insert text)

20. Do you think incentivizing users to give feedback can be done as a socially-visible activity (e.g. users who provide feedback would be happier when they get social recognition and this would increase their response rate)?

○ Yes
○ No, can you explain why? ____________________

21. According to your answer to the previous question, can you think of any potential social incentives to increase users' response to feedback requests? (Please insert text)

22. From your point of view, please enumerate the main engineering challenges in developing an adaptive acquisition of users' feedback. (Please insert text)

23. Can you mention some domains that have a strong relationship to the design of an adaptive feedback acquisition? (please insert text)

24. How do you think that your current and previous research relates to ours and is there anything you would like us to know? (Please insert text)

Second-Phase Expert Survey on Socially-Adaptive Software

Introduction

Thank you for helping me out. .

Before you start taking they survey, we would like to remind you with the purpose of our previous survey. The purpose of our previous survey was to gather experts' opinions in order to identify the potential core issues in the area of feedback acquisition for Social Adaptation. By Social Adaptation, we mean the software ability to collect users' explicit feedback on quality at run-time and use it to adapt by switching to a behaviour shown by the feedback to better fit a certain context.

The survey consists of 14 questions, some are repeated with different response options. In average, the survey would take you 10-15 minutes. Thank you for taking the time to complete this questionnaire.

Data Protection

Bournemouth University is a registered Data Controller. Any information that you supply will be held anonymously and securely in accordance with the Data Protection Act 1998 and will only be used for the purposes of this survey. Your personal details will not be made available outside the University. Any queries regarding Data Protection should be addressed to the Information Officer at information@bournemouth.ac.uk. Cookies, personal data stored by your Web browser, are not used in this survey.

Malik Al Maliki
malmaliki@bournemouth.ac.uk

Social Adaptation

In Social Adaptation, software will learn from feedback provided by users. Learning here is about the collective trend and behaviour of users, their interaction, and conflict resolution. In this regard, 2 main challenges were identified by experts.

1- Please rank the challenge degree of each of these 2 challenges:

- A: It is challenging and it requires significantly new approaches
- B: It is challenging but it can still be solved by extending and customizing existing approaches
- C: It is not really challenging and/or solutions already exist in the literature

	A	B	C
Challenge 1: Obtaining, analyzing and mining user feedbacks to inform recommendations that are consistent with the system's requirements	❍	❍	❍
Challenge 2: Identifying appropriate learning models for Social Adaptation which are capable to process requirements and users feedback and evolve the system over time.	❍	❍	❍

2- Please rate the relevance degree of these 2 challenges to the research in Requirement Engineering (RE):

- A: It is very relevant to RE research
- B: It is not strictly relevant to RE research, but having a solution for it is still beneficial to RE
- C: The challenge and solution are not relevant to RE research and practice

	A	B	C
Challenge 1: Obtaining, analyzing and mining user feedbacks to inform recommendations that are consistent with the system's requirements	❍	❍	❍
Challenge 2: Identifying appropriate learning models for Social Adaptation which are capable to process requirements and users feedback and evolve the system over time.	❍	❍	❍

Please select the appropriate answer for each of the following statements:

3- Experts stated that the key benefit of Social Adaptation for software developers is to:

	Agree	Partially agree	Disagree
improve the knowledge of the adaptive system and its users. E.g. it helps identify the distribution of software use and quality judgments and conflicts across age groups, geo-location, time of day, etc.	❍	❍	❍
improve the development of adaptive systems by reducing the time to deploy, the upfront effort in the design phase and increasing speed to react to changes required. i.e. validation is done by users as a lifelong activity	❍	❍	❍

4- Experts stated that the key benefit of Social Adaptation for software users is to:

	Agree	Partially agree	Disagree
improve users' trust in the adaptive system; users feel their voice is considered	❍	❍	❍
improve users' satisfaction, software behaves according to users' judgments	❍	❍	❍
improve transparency, adaptation process and decisions are visible to users	❍	❍	❍
improve users' confidence in adaptive systems, users are confident that software will do what they wish to a good quality	❍	❍	❍

5- Experts stated that Social Adaptation is highly applicable and useful in the following domains:

	Agree	Partially agree	Disagree
Mobility intensive systems such as driving-navigation systems where software is used by users in different contexts	❍	❍	❍
Large-scale systems where Social Adaptation avoid us the cost and complexity of traditional validation and lifelong users acceptance test	❍	❍	❍
Real time management systems such as traffic management where crowd-sourcing will empower the system monitor and, hence, the decision making	❍	❍	❍
Highly interactive system where it is hard to know a priori how users will judge quality in the diverse contexts of use and human-computer interaction	❍	❍	❍
Prototyping tools, where users' feedback can be used to infer user needs before a final implementation is carried out	❍	❍	❍

6- Experts stated that Social Adaptation should NOT be applied in the following domains:

	Agree	Partially agree	Disagree
Critical systems: Life-critical software used in hospitals is an example	❍	❍	❍
Security and Privacy sensitive applications	❍	❍	❍
Non- or less-interactive systems under the control of centralized authorities such as payroll system or an embedded system	❍	❍	❍

Experts' responses indicated that users' involvement in tailoring and validating software adaptation through feedback provision should be accommodated under some constraints and users should not be always involved. In this regard, 4 concrete challenges have been identified.

7- Please rate the degree of challenge of each of these 4 challenges:

- A: It is challenging and it requires significantly new approaches
- B: It is challenging but it can still be solved by extending and customizing existing approaches
- C: It is not really challenging and solutions already exist in the literature

	A	B	C
Challenge 1: Learning how to measure users' involvement with the system as a main measure of their potential role in tailoring adaptation	❍	❍	❍
Challenge 2: Identifying the degree to which users are allowed to configure the software on the fly at run-time	❍	❍	❍
Challenge 3: Identifying the possible restrictions for users' involvement, e.g. users of certain expertise can provide feedback	❍	❍	❍
Challenge 4: Measuring to which degree users are willing to get uninvolved and trust the software system' autonomous control, i.e. when to stop or minimize users involvement	❍	❍	❍

8- Please rate the relevance degree of these 4 challenges to the research in Requirement Engineering (RE):

- A: It is very relevant to RE research
- B: It is not strictly relevant to RE research, but having a solution for it is still beneficial to RE
- C: The challenge and solution are not relevant to RE research and practice

	A	B	C
Challenge 1: Learning how to measure users' involvement with the system as a main measure of their potential role in tailoring adaptation	❍	❍	❍
Challenge 2: Identifying the degree to which users are allowed to configure the software on the fly at run-time	❍	❍	❍
Challenge 3: Identifying the possible restrictions for users' involvement, e.g. users can provide feedback after certain period of use	❍	❍	❍
Challenge 4: Measuring to which degree users are willing to get uninvolved and trust the software system' autonomous control, i.e. when to stop or minimize users involvement	❍	❍	❍

Social Adaptation allows a user to rely on the collective judgement of the users' community. However, the user should still be enabled to make his/her personal decisions. In this regard, 2 research challenges have been identified by experts.

9- Please rate the degree of challenge of each of these 2 challenges:

- A: It is challenging and it requires significantly new approaches
- B: It is challenging but it can still be solved by extending and customizing existing approaches
- C: It is not really challenging and solutions already exist in the literature

	A	B	C
Challenge 1: Developing models and languages to allow users to specify their preferences on the way to reach their requirements. i.e. to specify when to rely on the crowd and when to take their personal choices	❍	❍	❍
Challenge 2: Developing mechanisms to allow software to deduce such users' preferences without getting them explicitly involved	❍	❍	❍

10- Please rate the relevance degree of these 2 challenges to the area of Requirement Engineering (RE):

- A: It is very relevant to RE research
- B: It is not strictly relevant to RE research, but having a solution for it is still beneficial to RE
- C: The challenge and solution are not relevant to RE research and practice

	A	B	C
Challenge 1: Developing models and languages to allow users to specify their preferences on the way to reach their requirements. i.e. when to rely on the crowd and when to take their personal choices	○	○	○
Challenge 2: Developing mechanisms to allow software to deduce such users' preferences without getting them explicitly involved	○	○	○

Engineering Feedback Acquisition

11- The following factors have an impact on the quality of collected feedback:

	Agree	Partially agree	Disagree
Time: asking users for feedback when they are busy may lead to poor response rate, untruthful feedback, or it may be discarded	○	○	○
User interface: Short, concise, clear questions should be preferred to long complicated ones	○	○	○
Language: The phrasing of the question, for example based on the language proficiency of the user, will determine how users interpret and respond to questions	○	○	○
Quality of users: asking the right user population for feedback and ensuring the size of users is representative of the group characteristics	○	○	○
Mood of users and emotional state (e.g. happy, bored, excited, angry etc.) during feedback acquisition	○	○	○

12- Experts agree that an adaptive feedback acquisition process is needed to increase the response rate and quality of collected feedback. The following can be possible drivers for such an adaptive process:

	Agree	Partially agree	Disagree
User experience: e.g. usage frequency. This could inform how often users should be probed for feedback	○	○	○
Application constraints: such as the application model, domain model, and level of interactivity of the software.	○	○	○
Direct enquiry to know how to customize the process. This involves asking the users if they wish to provide feedback, how often they wish to do so and what methods they'll like to use for providing feedback	○	○	○

Experts indicated that, engineering of an adaptive software-based feedback acquisition stands out as a technically challenging process. In this regard, the following 13 challenges have been identified by experts:

13- Please rank the challenge degree of each of these 13 challenges:

- A: It is challenging and it requires significantly new approaches
- B: It is challenging but it can still be solved by extending and customizing existing approaches
- C: It is not really challenging and solutions already exist in the literature

	A	B	C
Challenge 1: Learning how interfaces are built (with regard to usability factors) to gather feedback in various contexts.	○	○	○
Challenge 2: Monitoring users' context of use to decide the right interaction style for different types of users and contexts	○	○	○
Challenge 3: Developing appropriate interaction styles for different types of users and contexts of use (e.g. avoid asking for long textual responses when a user is using a Smartphone)	○	○	○
Challenge 4: Deciding the right time for different types of users to be asked for feedback	○	○	○
Challenge 5: Identifying the types of feedback that should be asked from users (e.g. qualitative, quantitative or both)	○	○	○
Challenge 6: Identifying mechanisms for monitoring user's feature usage statistics/trends to inform which feedback are requested from the user (e.g. on which feature and how often she/he should be probed for feedback).	○	○	○
Challenge 7: Identifying the relevant adaptation drivers in such an adaptive mechanism	○	○	○
Challenge 8: Engineering the relevant adaptation drivers in a non-intrusive way to users	○	○	○
Challenge 9: Identifying who to ask for feedback (the right user population to be asked)	○	○	○
Challenge 10: Measuring the quality and trustworthiness of users' feedback	○	○	○
Challenge 11: Engineering of an application-independent framework for users' feedback acquisition	○	○	○
Challenge 12: Identifying domain-specific feedback acquisition languages and mechanisms	○	○	○
Challenge 13: Developing an incentive scheme to maximize users' engagement and response rate.	○	○	○

14- Please rate the relevance degree of these 13 challenges to the area of Requirement Engineering (RE):

- A: It is very relevant to RE research
- B: It is not strictly relevant to RE research, but having a solution for it is still beneficial to RE
- C: The challenge and solution are not relevant to RE research and practice

	A	B	C
Challenge 1: Learning how interfaces are built (with regard to usability factors) to gather feedback in various contexts.	○	○	○
Challenge 2: Monitoring users' context of use to decide the right interaction style for different types of users and contexts	○	○	○
Challenge 3: Developing appropriate interaction styles for different types of users and contexts of use (e.g. avoid asking for long textual responses when a user is using a Smartphone)	○	○	○
Challenge 4: Deciding the right time for different types of users to be asked for feedback	○	○	○

Challenge 5: Identifying the types of feedback that should be asked from users (e.g. qualitative, quantitative or both)	○	○	○
Challenge 6: Identifying mechanism for monitoring user’s feature usage statistics/trends to inform which feedback is requested from the user (e.g. on which feature and how often she/he should be probed for feedback).	○	○	○
Challenge 7: Identifying the relevant adaptation drivers in such an adaptive mechanism	○	○	○
Challenge 8: Engineering the relevant adaptation drivers in a non-intrusive way to users	○	○	○
Challenge 9: Identifying who to ask for feedback (the right user population to be asked)	○	○	○
Challenge 10: Measuring the quality and trustworthiness of users’ feedback	○	○	○
Challenge 11: Engineering of an application-independent framework for users’ feedback acquisition	○	○	○
Challenge 12: Identifying domain-specific feedback acquisition languages and mechanisms	○	○	○
Challenge 13: Developing an incentive scheme to maximize users’ engagement and response rate.	○	○	○

Would you like to add any comments?

11.2 APPENDIX 2

Users’ Interviews on Users’ Feedback Acquisition

Introduction Script:

Thank you for helping me out. I’m interested in learning about the ways people act and behave when being asked for feedback from software applications. With you permission I would like to record the session so that I can review it later for research purpose only.

Do you have any questions regarding the information letter or the consent form?

Do you agree to volunteer?

With your permission, I’m going to begin recording now. Remember that this is to learn about you, there are no wrong answers and every bit of information you can give me is helpful.

Interview Protocol:

Time of Interview:
Date:
Place:
Interviewee:
Interviewer:

Personal information:

Tell me a little about you?

What do you do?

How old are you?

General Software and Computer Familiarity:

What kind of computer devices do you use?

Which one do you use the most?

When do you always use your computer during your day?

What kind of software application do you always use?

Experiences and behaviour:

Tell me what do you think of the idea of asking users for feedback when using software applications (e.g. Skype quality feedback popup after a call)?

Do you remember a time you were asked for feedback, tell me about it (e.g. Skype, I show some example from Amazon, etc.)?

Do you remember where you were at that time (example, at work)?

Do you remember the time you were asked for it (e.g. afternoon)?

How were you asked for it and in which context (example, during playing a game by popping up a feedback form)?

Were you using your personal computer or another device (e.g. smartphone)?

What kind of software were you using at that time?

Did you provide the feedback? Why (what was your motivation)? Why not?

Do you remember if you were asked for feedback before this time and what did you do (I give an example)?

Have you ever been annoyed when you were being asked for feedback? Why?

How do you think you should have been asked (e.g. by email at your free time)?

Have you ever received multiple requests at different times to provide feedback (I give an example)? Tell me about it?

How was your reaction to it?

How would you prefer to be asked for feedback (e.g. when, popup)?

Do you have any other comments?

If I had any follow up questions would you mind if I emailed you?

Students Survey on Users' Feedback Acquisition

Invitation Email:

Dear Participant,

My name is Malik Al Maliki and I am a PhD student at DEC school. I'm interested in learning about the ways people act and behave when being asked for feedback from software applications. I would like to invite you to participate in an online questionnaire and provide us with your valuable input. By completing the survey, you will be entered automatically in to a free prize draw to win 1 of 3 £10 Amazon vouchers. ***Moreover, Participants who complete the survey and add thoughtful comments in the comment textbox provided for some questions will be entered in to another free prize draw to win 1 of 2 £25 Amazon vouchers.***

Your input will be used only for the purposes of the research.

Below is a link to our online expert survey. The survey is user-friendly and you should be able to complete it within 5-10 minutes.

We appreciate your participation and your feedback will be greatly appreciated. To begin, simply click on the link below, or cut and paste the entire URL into your browser to access the survey:

Survey URL:

https://qtrial.qualtrics.com/SE/?SID=SV_8eKSG2gTw5oS4KN

If you have any questions please do not hesitate to contact me. Thank you for your participation.

Best,

Malik AL Maliki,

Bournemouth University, UK

Survey Introduction:

Dear Participant,

The purpose of this survey is to find out how people act and behave when being asked for feedback (e.g. rating an item, like or dislike, text feedback, etc) by software applications.

By completing the survey, you will be entered automatically in to a free prize draw to win 1 of 3 £10 Amazon vouchers. ***Moreover, Participants who complete the survey and add thoughtful comments in the comment textbox provided for some questions will be entered in to another free prize draw to win 1 of 2 £25 Amazon vouchers.***

Data Protection:

Bournemouth University is a registered Data Controller. Any information that you supply will be held anonymously and securely in accordance with the Data Protection Act 1998 and will only be used for the purposes of this survey. Your personal details will not be made available outside the University. Any queries regarding Data Protection should be addressed to the Information Officer at information@bournemouth.ac.uk. Cookies, personal data stored by your Web browser, are not used in this survey.

Best,

Malik Almaliki

MALMALIKI@BOURNEMOUTH.AC.UK

Survey Questions:

I have read and understood the above introduction and I would like to participate in this survey

○ Yes
○ No

1- Please select your age

○ 18-25
○ 26-34
○ 35-54
○ 55-64
○ 65 or over
○ I prefer not to answer

2- Please select your gender

○ Male
○ Female
○ I prefer not to answer

3- Please select your nationality (optional)

………………………………………………………………………

4- Please select your current level of education?

○ No schooling completed
○ Some high school, no diploma
○ Associate degree
○ Bachelor’s degree
○ Master’s degree
○ Professional degree
○ Doctorate degree
○ Others ____________________

5- Please enter your main profession. Please give some details about the area of work (e.g. student in tourism..) and feel free to mention more than one if applicable

………………………………………………………………………

6- Please indicate if you would like to be (optional)

□ Sent the results of this survey
□ Contacted if we have future studies (mostly 2-3 time per year)
□ Contacted if we need to clarify or discuss further some of your responses

7- Please provide your name (Optional)

………………………………………………………………………

8- Please provide your Email address (preferably your work email address)

………………………………………………………………………

9- Please choose the kind of software systems you use frequently and significantly

□ Desktop Applications: programs installed on your machine (e.g., MS Office, Image Editors, Movie Players, etc.)
□ Web Applications: programs you access via internet (e.g., Online documents editors like Google Doc, online calendars, storage services like DropBox, etc.)
□ Social Networking: to have contacts and interact with them (Facebook, Twitter, Forums, Blogs, Wikis as an editor, etc.)
□ Mobile Apps: applications installed on your mobile
□ Search Engines: could be generic such as Google, or domain-specific such as Skyscanner for flights and Venere and Booking for hotel reservation, etc.
□ E-commerce (like Ebay and Amazon, Online Shopping)
□ Others (please mention them) ____________________

10- Please choose the kinds of software which noticeably and frequently asks you to give feedback (e.g. to rate an item or a service, to like or dislike, to recommend to friend, to report abuse, etc)

❑ Desktop Applications, programs installed on your machine (e.g, MS Office, Image Editors, Movie Players, etc.)
❑ Web Applications, programs you access via internet (e.g., Online documents editors like Google Doc, online calendars, storage services like DropBox, etc.)
❑ Social Networking: to have contacts and interact with them (Twitter, Facebook, Twitter, Forums, Blogs, Wikis as an editor, etc.)
❑ Mobile Apps: applications installed on your mobile
❑ Search Engines (Generic such as Google, and domain specific such as Skyscanner for flights and Venere and Booking for hotel reservation, etc.)
❑ E-commerce (like Ebay and Amazon)
❑ Others (please mention them) ____________________

11- Please choose the kinds of software which noticeably and frequently allows you to give feedback (e.g. to rate an item or a service, to fill a "contact us" form, to like or dislike, to recommend to friend, to report abuse, etc)

❑ Desktop Applications, programs installed on your machine (e.g, MS Office, Image Editors, Movie Players, etc.)
❑ Web Applications, programs you access via internet (e.g., Online documents editors like Google Doc, online calendars, storage services like DropBox, etc.)
❑ Social Networking: to have contacts and interact with them (Twitter, Facebook, Twitter, Forums, Blogs, Wikis as an editor, etc.)
❑ Mobile Apps: applications installed on your mobile
❑ Search Engines (Generic such as Google, and domain specific such as Skyscanner for flights and Venere and Booking for hotel reservation, etc.)
❑ E-commerce (like Ebay and Amazon)
❑ Others (please mention them) ____________________

12- Which of these devices do you use frequently and significantly?

❑ Desktop computer
❑ Tablet
❑ Mobile/Smart Phones

13- What computing device do you typically use when you tend to be willing to provide feedback?

❑ Desktop computer
❑ Tablet
❑ Mobile/Smart Phones
❑ No preferences

14- Why do you typically use software systems (websites, search engines, MS Office, DropBox, Social Media, Email applications, ...etc)

□ For professional reasons: my work requires that
□ For academic reasons: my study/research requires that
□ Daily life activities (booking, online shopping, looking for bus schedule, taxi number, etc)
□ Entertainment (gaming, social networks for entertainment purposes, etc.)
□ For social interaction (social networks, blogs, forums, etc.)
□ Others ____________________

15- Do you like it when software systems ask you to give your feedback about their services?

○ Yes, very much
○ Yes, to a certain extent
○ Not much, but it is still fine to be asked
○ Not really, I do not see many benefits providing it
○ Others ____________________

16- What is the way you prefer for gathering feedback from you?

□ I prefer to be explicitly asked for feedback, e.g., by sending me an email or showing me a popup dialogue which I can respond to or just ignore.
□ I prefer that feedback is taken from me implicitly, e.g., by monitoring and analyzing my use of the systems (e.g. navigation path, click history, the time I spend on each step, what mistakes I typically do, etc)
□ I prefer to have the choice to go to the feedback center on voluntary basis and leave a feedback there, e.g., "contact us" form, or a forum/blog designed specifically for that.
□ I do not like to give feedback regardless the way it is obtained
□ Others or I would like to add a comment ____________________

17- What would be the main reasons that would stop you from providing feedback or responding to a feedback request?

□ I think it is mainly for commercial reasons and not about improving user's experience
□ When it requires significant time to think about and answer
□ When It comes at the wrong time, e.g., when I am engaging with some other activity
□ When I am unaware of how my feedback is being used and for which purpose
□ When I am unaware if my feedback has been taken into account or led to any change
□ Privacy reasons, e.g., when other people can know or infer that I provided that feedback
□ When I am unfamiliar with the subject of the question or do not have enough experience to give a meaningful answer
□ Others or I would like to add a comment ____________________

18- Which of the following factors have a significant effect on your willingness to provide feedback and respond to feedback requests?

- ❑ The language used in the feedback request (friendliness, succinctness, clarity, etc.)
- ❑ The graphical design of the feedback request (font size, colors used, the kind of photos used, etc.)
- ❑ Simplicity/Complexity of the method allowed to provide the feedback (clicking, a voice message, text with/without auto-completion, etc.)
- ❑ Fitness of the design and content of feedback request to the computing device I am using (e.g., when it give less details and simpler content on smart phones)
- ❑ Fitness of the design and content of the feedback request to my current focus when asked for feedback (asking for the quality of a certain feature of a Calendar software should be done when I have just used it..)
- ❑ Fitness of the design and content of the feedback request to my work status when asked for feedback (simplified version of it if I am busy..)
- ❑ When it gives me some information, summary or statistics about the feedback which has been already given by other people
- ❑ Others or I would like to add a comment ____________________

19- Which of the following communication methods do you prefer to provide feedback

- ❑ Off-line: by sending me a message asking for feedback such as an email, Facebook message, mobile text message etc.
- ❑ Real-time: for example by showing me a popup dialogue when I am using the software or showing some vote or form at a corner of a webpage I am navigating
- ❑ By giving me small hints and tips, while using the software/webpage, that I can go to a feedback center such as "contact us" form in a website or a forum specifically designed for this purpose, etc.
- ❑ Others or I would like to add a comment ____________________

20- How do you feel about the feedback requests which come in the wrong time (a popup dialogue when you are navigating a website and moving to another page, a hint in a YouTube video to encourage you to rate it)

- ❑ Are OK with me
- ❑ I think this is one of the ways which puts a gentle pressure on me so that I give feedback
- ❑ Decrease my willingness to give feedback
- ❑ I may give less truthful feedback just to get rid of the dialogue
- ❑ I believe it is an inconsiderate way to force me to give feedback
- ❑ Others or I would like to add a comment ____________________

21- What kind of feedback do you usually prefer to give?

- ❑ Qualitative feedback (e.g. writing sentences or lines of texts to communicate my thoughts in free-style).
- ❑ Quantitative feedback with a fixed list of items (e.g. rating and giving stars to an already provided set of quality attributes such as comfort, speed, etc.).
- ❑ Quantitative feedback with a flexible list of items to which I and other people can add (e.g. rating and giving stars to a list of attributes and allowing me and other people to add other attributes we think they are also relevant).
- ❑ A combination of qualitative and quantitative where I have the choice to use my preferable one (e.g. rating and giving stars with the ability to add text if I need).
- ❑ Others or I would like to add a comment ____________________

22- If you ignore or forget to respond to a feedback request, would sending you a reminder motivate you to give that feedback?

- ❑ Yes, definitely I appreciate reminders as I may forget to provide response on time
- ❑ Yes, reminders could be a gentle pressure on me to give feedback as I may not forget but feel lazy to do it.
- ❑ Generally speaking, reminders are not that useful. if the feedback request is really important to me, I would have responded to it from the first time.
- ❑ No, reminders are often annoying
- ❑ No, and when it repeats, it makes me ignore even other different feedback requests from the same person/software
- ❑ Others or I would like to add a comment ____________________

23- I would be more motivated to give feedback

- ❑ When I have a positive experience with the software service
- ❑ When I have a negative experience with the software service
- ❑ It does not really matter, I can give feedback even if my experience was neither very good nor very bad
- ❑ Others or I would like to add a comment ____________________

24- How does the visibility of other people's feedback affect your willingness to give feedback?

- ❑ If I must give my feedback first in order to be able to see other people's feedback, I will be LESS motivated to give feedback
- ❑ If I must give my feedback first in order to be able to see other people's feedback, I will be MORE motivated to give feedback
- ❑ If I am able to see other people's feedback first and then I have the option to accept/reject to give feedback, I will be MORE motivated to give feedback
- ❑ Being able to see other people feedback does not have any noticeable impact on my willingness to give my feedback
- ❑ Others or I would like to add a comment ____________________

25- How does the similarity of your opinion to other people's opinions, shown via their feedback, affect your willingness to give feedback?

- ❑ My willingness increases when I see that the majority of people think differently from me
- ❑ My willingness increases when I see that the majority of people think similarly to me
- ❑ My willingness decreases when I see that the majority of people think differently from me
- ❑ My willingness decreases when I see that the majority of people think similarly to me
- ❑ Others or I would like to add a comment ____________________
- ❑ This has no noticable impact on my willingness to give my feedback

26- When looking at other people's feedback, how do you decide the trustworthiness of their given feedback?

- ❑ By looking at their previous feedback
- ❑ By looking at the used language in their feedback/reviews (how serious it is, supported by facts, etc)
- ❑ By looking at their reputation (e.g., how many people following them, their business reputation like in Ebay and Amazon, etc)
- ❑ By looking at how their feedback/reviews are rated by other people
- ❑ Others or I would like to add a comment ____________________

27- How does the number / volume of feedback already provided by other people on a subject affect your willingness to give feedback on it?

- ❑ My willingness increases when there is only few people who provided feedback
- ❑ My willingness decreases when there is only few people who provided feedback
- ❑ My willingness increases when there is already a large number of people who provided feedback
- ❑ My willingness decreases when there is already a large number of people who provided feedback
- ❑ The number of people who provided feedback has no effect on my willingness to give feedback
- ❑ Others or I would like to add a comment ____________________

28- How does a social recognition / visibility of you as a feedback provider affect your willingness to give feedback?

- ❑ It is an incentive to me when I am recognized and my feedback is visible to everyone
- ❑ It is an incentive to me if I am seen/recognized but only when I can decide who can see that, e.g. when I am given the choice to select some of my Facebook or Twitter contacts/groups to see my feedback
- ❑ It is nice to be visible only when others can see my feedback which led to some changes on the system, i.e. when it had an impact
- ❑ It is nice to be visible when my feedback is aligned with the main trend of other people who provided feedback
- ❑ It is nice to be visible when it has something different from the main trend of other people who provided feedback
- ❑ Others or I would like to add a comment ____________________
- ❑ It does not affect my willingness to give feedback (I do not care about it)

29- What would be your feeling when you receive a high number of different feedback requests from one software service (desktop application, website, mobile app. etc.):

- □ It is fine with me, I like to give feedback often
- □ It is fine with me as long as I am not forced to give answers
- □ I tend to respond to some of them
- □ I tend to give less focused or less truthful feedback
- □ It leads me to give a negative feedback as the requests make me feel annoyed
- □ I tend to ignore all of them and I tend to consider it as a spam
- □ I tend to stop using the software sending me these requests
- □ To you, what is a reasonable number of feedback requests from a software per day? ____________________

30- Would conducting the acquisition of feedback as a social activity be interesting to you. For example, you can always visualize how your direct and indirect social contacts are rating a certain service and how your feedback influenced the trend in your community. Software may suggest you to team up with certain users who have similar feedback, etc.

- ○ Yes ____________________
- ○ No ____________________

31- Would you like to suggest any possible way to conduct feedback acquisition and visualization as a social activity?

- ○ Yes (Please elaborate on this as much as you can) ____________________
- ○ No (Please elaborate on this as much as you can) ____________________

11.4 APPENDIX 4

Focus Group on Users' Behavioural to Software-based Feedback Acquisition: The Session Guide

Session Opening:

At first, participants will be given a sheet of information that explains how the session will go, what they are expect to do, how we will use the information obtained during the session and how they can contact me for further information. In addition, a consent form will be also given to each participant as well as a demographic information sheet to collect their demographics which will help in assuring diversity and cultural fit to our inclusion criteria.

Introduction:

To familiarize the participants with our topic, I will brief the participants to our discussed topic (Feedback Acquisition) and I will also present some slides that contain some examples of feedback requests (i.e. Skype call quality feedback popup), information/diagram about what we found with regard to social factors in affecting users behaviors and how culture could play a part in it too. Some of the examples will be shown repetitively to the participants throughout the session to refresh their background knowledge of the topic.

Opining the discussion:

Before opening the discussion I will get their permission to record the session (the permission is also taken in the consent form) and then I will ask them a set of introductory questions to open the conversation such as "Can you describe your experience with feedback/reviews in software applications?" and "Can you tell me about the last time you were asked for feedback?", "what kind of forums and feedback channels have you participated in?"..

Other introductory sub-questions might emerge here!

Visibility of other feedback means that you can see what others said about the service or the product and how many of them favorite that, etc. From your past experience and observation of the attitude of other people, do you think the *visibility and* of feedback provided by other users would have an impact on the following dimension, if yes then how:

The response rate (i.e. users might be more/less willing to respond to feedback requests when they can see others feedback)

Users' engagement level (i.e. users' engagement with the software might increase or decrease when they can see the gender, the age, the level of expertise, the location of some other feedback providers)

Feedback consistency (i.e. users' might give iteratively inconsistent feedback depending on the already provided feedback and who provided it, e.g. the social position of some

feedback providers could create a bias in the subsequent feedbacks of other users)
Feedback objectivity (i.e. users' might not give objective feedback depending on the language used by some feedback providers)
Feedback relevancy (i.e. users' might give irrelevant feedback when being affected by others feedback or their language)

More sub-questions and discussion could emerge regarding the following factors of visibility:

Social Position
Similarity.
Anonymity of feedback providers and Translucency
Language used (formal vs casual, absurd vs polite, short vs long, sharp vs. fuzzy, etc.)
Gender of feedback provider (male vs female, known vs. unknown, genuine vs fake vs suspicious identity, etc)
Personal relationship with a feedback provider (personal contact, group membership, similar profile, other relevant attributes)

From your past experience and observation, do you think the *social recognition (i.e. badges for being a loyal feedback provider, visibility as influential feedback providers who their feedback impacted the product/software, sharing or reposting your feedback, etc.)* for feedback providers impact the following dimension, if yes then how:

The response rate (i.e. some users might be more willing to respond to feedback requests when they are being socially recognized).
Users' engagement level (i.e. some users might be more engaged with the software when they are being socially recognized).
Feedback consistency (i.e. users' might give inconsistent feedback depending on how they are socially seen)
Feedback objectivity (i.e. some users might be more objective with their feedback since they are socially recognized and what they say is always judged by others).
Feedback relevancy (i.e. feedback relevancy level might increase when you are socially recognized).
More sub-questions and discussion will emerge.

From your past experience and observation, how do you think conducting the *feedback acquisition as a social activity* (user might be seen or not) impact the following dimension, if yes then how:

The response rate (i.e. some users might be more willing to respond to feedback requests when statistics of how his online social community (e.g. friends in his circle/online friends) is providing feedback are shown).
Users' engagement level (i.e. some users might be more engaged with the software when they are teamed with another socially active feedback provider (like keeping checking your friends comments on FB).
Feedback consistency (i.e. users' might give inconsistent feedback depending on the type of the social activity the software put them in such as being part of 2 different groups at the same time)
Feedback objectivity (i.e. some users might be less objective with their feedback when being as a part of certain social community that could influence their objectivity (being a member of staff in a certain company would affect your feedback of a product produced by that company or a partner company, etc)).
Feedback relevancy (i.e. feedback relevancy level might decrease when you are socially

teamed with another feedback provider who could influence your behaviour).
More sub-questions and discussion will emerge

From your past experience and observation, how do you think conducting *the volume/amount of given feedback* impact the following dimension, if yes then how:

The response rate (i.e. some users might be more willing to respond to feedback requests when few feedback are already provided).
Users' engagement level (i.e. some users might be more engaged with the software when the volume of given feedback by others is high).
Feedback consistency.
Feedback objectivity.
Feedback relevancy (i.e. feedback relevancy level might decrease when the volume/number of given feedbacks is high).
More sub-questions and discussion will emerge

Another question could be: is the type of service, e.g. a dating website or a website for donation, would have an influence on the way you give feedback including visibility and social activity, etc.

What do you like/dislike in the current feedback acquisition methods

What do you like to see in future feedback requests

After the discussion is finished the participant will be given some time to comment or add any feedback regarding what has been discussed then they will be thanked again for their participation and the lunch voucher will be given out.

Initial Personas for Feedback Acquisition

The following 4 personas represent different types of users' behaviours in relation to feedback acquisition in software application. Table 19 shows the components of each persona.

Mark

Profile: Feedback antagonists
Age:50
Gender: Male
Job: Business man
Socially affected to give feedback: No

Statement: *"I find it problematic, hindering and unprofessional to send me any kind of feedback requests. If I'm not happy with something I will go to their website and complain right to them".*

Goals: Complete the task in hand without interruption from feedback request.

Behaviour to feedback:

Mark is a business man and he spends a lot of time on his computer working on his business. He holds a very negative view about feedback request coming from software applications. He does not have the time to be responding to feedback request due to his heavy workload. Mark thinks feedback request coming from software applications can waste his time and he doesn't tolerate to be asked for feedback at all (whether it's online of offline feedback request). In fact, he thinks that feedback requests that interrupt him while he is working are an impolite way to get information out of him. Since Mark doesn't tolerate to be asked for feedback at the first place, he is not affected by any social factors to give feedback at all (i.e. social recognition does not make him happy to give feedback). however, Mark believes that there should be a channel for him to deliver his opinion whenever he likes by making him able to submit his feedback on a voluntarily base and without being proactively asked by the software (i.e. through a contact us form).

Culture Suitability: Neutral

Amy

Profile: passive and stingy
Age:29
Gender: Female
Job: School teacher
Socially affected to give feedback: No

Statement: "*The benefits of my feedback are always not clear to me as a user."*

Goal: Complete the task in hand without interruption from feedback request.

Behaviour to feedback:

Amy is a school teacher and spends a great deal of time on the internet reading and researching educational related topics. She has always been passive towards all feedback acquisition methods. In general, she does not like the idea of being asked for/reminded to give feedbacks by software applications because she does not believe her given feedback is going to be considered or lead to any changes/ improvements on the software. She does not even get influenced or motivated by any social factors to give feedback (i.e. visibility of others feedback on the software doesn't really make her want to give feedback). However, sometimes she can be tolerant to online feedback request (i.e. showing her a feedback popup dialogue while she is using the software). This is due to the fact that she doesn't accept the idea of having her email inbox filled with feedback requests or feedback reminders. She tolerates the online ones since she has the control to respond or dismiss it at only one click sometimes.

Culture Suitability: Neutral

Linda

Profile: Privacy tolerant and socially ostentatious
Age:20
Gender: Female
Job: Undergraduate student
Socially affected to give feedback: Yes

Statement: *"Giving feedback is a social and community experience and it helps to feel among others".*

Goals: Impact the software with her feedback + raising others awareness about the used software + being socially recognised.

Behaviour to feedback:

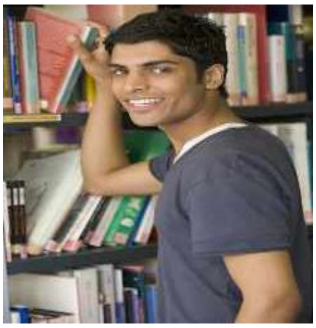
Linda is an undergraduate university student and spends a great deal of time on her computer studying as well as heavily social networking (i.e. Facebooking). In general, she is not a big fan of the idea of feedback requests and reminders coming from software applications. However, she gets interested in replying to feedback requests when the feedback requests socially motivate her to do so (i.e. by making her socially recognized for her helpful feedback). This is perhaps due to her likeness of social networking and the time she spends socialising with others/friends on the internes which made her motivated towards socially enriched feedback requests. Generally, Linda is positively affected by the following social factors to give feedback:

- **Volume of already given feedback**: She gets enthusiastic to give feedback when there is low number of feedbacks already given on a software. She believes it's helpful to increase the number of given feedback which will then result in other users having a better and richer idea about the software.
- **Visibility and similarity of other users' feedback**: Linda also gets more interested to give feedback if she is able to see other users' feedback on the software first and then having the option to accept/reject to give feedback.
- **Social recognition:** Since Linda appreciates social networking and gives it a great deal of her time, she likes to be socially recognized for her given feedback which she believes could help others and make her socially popular.
- **Feedback acquisition as a social activity:** This social factor also makes Linda motivated to give feedback as well as engaging with software. For example, she gets enthusiastic to feedback requests when she is able to visualize how her social friends are rating a certain software and how their feedback influenced the trend in her community.

In addition, Linda prefers to be approached for feedback by using hints and tips to gather her feedback (e.g. by telling her that she can go to a feedback centre for this purpose and leave her feedback) or by using an online method as a second option (i.e. popups while she is using the software). Interestingly, Linda does not mind to be implicitly reached for feedback (e.g. implicitly collecting information about her software usage)

Culture Suitability: Middle Easterns-like

Jack



Profile: Privacy fanatic and generous
Age:35
Gender: Male
Job: Researcher
Socially affected to give feedback: Yes

Statement: *"I think emails are good if you want someone to actually sit down and write a couple of sentences about how they feel about your service popups and other 'push' mechanisms intrude & interrupt flow."*

Goals: Impact the software with his feedback + raising others awareness about the used software + being socially recognised.

Behaviour to feedback:

Jack as a researcher spends most of his time on the computer working on his research as well as networking with other researchers. Jack believes in the power feedback in general and its positive impact. He is a very positive person towards feedback requests and reminders coming from software application. However, he prefers to be asked for feedback in an offline way (i.e. through males or text messages). He believes online feedback request (i.e. popups) could somehow be intruding and interrupting especially when he is working on his research and deeply thinking. In addition, Jack is always concerned about his privacy and therefore he does not accept to implicitly collect feedback from him (i.e. tracking his usage of the software).

In addition, Jack is a socially motivated feedback provider and his willingness to give feedback is positively influenced by three social factors:

- **Social recognition:** He likes to be socially recognized for his valuable and trustworthy feedback which he believes could help others and raise the social awareness about the software in use.
- **Volume of already given feedback:** He gets enthusiastic to give feedback when there is high number of feedbacks already given on a software. This means to jack the software is popular and deserves his feedback.
- **Visibility of other users' feedback:** Jack also gets more interested to give feedback if he is able to see other users' feedback on the software first and then having the option to accept/reject to give feedback.

Culture Suitability: Westerns-like

Interview Guide on Persona-based Feedback Acquisition

Session Opening:
At first, Interviewee was given a sheet of information that explains how the session will go, what they are expect to do, how we will use the information obtained during the session and how they can contact me for further information. In addition, a consent form was also given to each expert to assure the interviews are ethically undertaken.

Objectives of the study:

- Evaluate and refine Personas that we previously developed to initially represent users' different behaviors to feedback acquisition.
- Assess the effectiveness and efficiency of using personas to inform the design of feedback acquisition and how personas can be used?

Introduction
Thank you for helping me out. I'm interested in investigating how users behave/react to feedback acquisition from industrial point of view. In addition, this study will also explore the use of Personas as a way to help developers inform the design of the feedback acquisition methods. With you permission I would like to record the session so that I can review it later for analysis and research purposes only.
Do you have any questions regarding the information letter or the consent form?
Do you agree to volunteer?
With your permission, I'm going to begin recording now. Remember that there is no wrong answer and every bit of information you can give me is helpful.

Interview Protocol

Time of Interview:
Date:
Company name:
Company size:
Interviewee Name:
Interviewee role in the company:
Years of experience:

Persona evaluation and refinement related questions:

- Do you employ the approach of Personas in your company, even implicitly or in non-formal settings, to inform the design of feedback acquisition? If yes, then how?

The four initial personas, sent already for review beforehand, will be shown to the interviewee and then the following questions will be asked repetitively on each persona:

According to your experience, does this persona represent a frequent, typical or considerable observed behavior of users?
How do you comment on the design and format of the persona (i.e. style, language used, structure, readability, understandability, etc)?
How would you modify/criticize it to make it more reflective of the behavior it represents (i.e. what does it lack, what does it misrepresent)? Please note that you may choose to split it or create an opposite version of it or add some contextual information in which such observed

behavior is likely to happen.

- Can you tell us if you have seen other users' behaviors/patterns to feedback acquisition that have not been captured by the presented personas?

After each question some sub-questions could naturally emerge.

Questions related to Personas usage to inform the design of feedback acquisition

- From your point of view, do you think the previously presented personas are beneficial to inform the design of feedback acquisition and to what extent? (the interviewee will be encouraged to elaborate and give example)

- What do you think of the following statements (the interviewees will be encouraged to elaborate on their answers) :

Personas make the design process of feedback acquisition easier in which engineers relate to human face and name instead of abstract user/customer data.
Personas supply a shared, fast and effective form of communication among software engineers and designers when designing feedback acquisition.
Personas describe user needs and wants with regard to feedback acquisition which limit stakeholders and developers ability to shape users to their convenience or own mental models.
Personas help engineers of feedback acquisition to keep the focus on the limited subset of users (persona) at a time which can result in more robust design decisions.
Personas are useful for feedback acquisition in the validation phase in which proposed designs and solutions can be reviewed and evaluated against the needs described by an individual persona.
Personas can help developers to drive various scenarios about users' behavior to feedback acquisition.
The created scenarios can result in a better elicitation and prioritization of users' requirements and expectations/preferences of feedback requests (i.e. primary personas can help prioritizing requirements)

- If you were to adopt personas to inform the design of feedback acquisition, what would be the development process (the stakeholders to involve, the sessions and their settings, the steps to go through and the conditions to observe) to do that?

- Generally speaking, what would you criticize about the adoption of personas to direct the design of feedback acquisition?
After each question some sub-questions could naturally emerge.

3.1 PAFA Evaluation Protocol

1- Session Opening:

At first, Interviewees will be given a sheet of information that explains how the session will go, what they are expected to do, how we will use the information obtained during the session and how they can contact me for further information. In addition, a consent form will also be given to each participant to assure the study is ethically undertaken.

2- Objectives of the study:

The aim of this study is to evaluate and refine our newly proposed PAFA; a software engineering method for the design of feedback acquisition. The purpose of this evaluation is to find out to what extent PAFA is effective and systematic in helping software engineers identify and cater for users' different requirements and behaviours with regard to feedback acquisition.

In the following, a case study that consists of three real scenarios is provided. The provided scenarios are meant to help the participants to work on PAFA and showcase how it can be applied in practice. The resulted outcome is hoped to positively impact the provided scenarios (e.g. introduce adaptivity to the given scenarios).

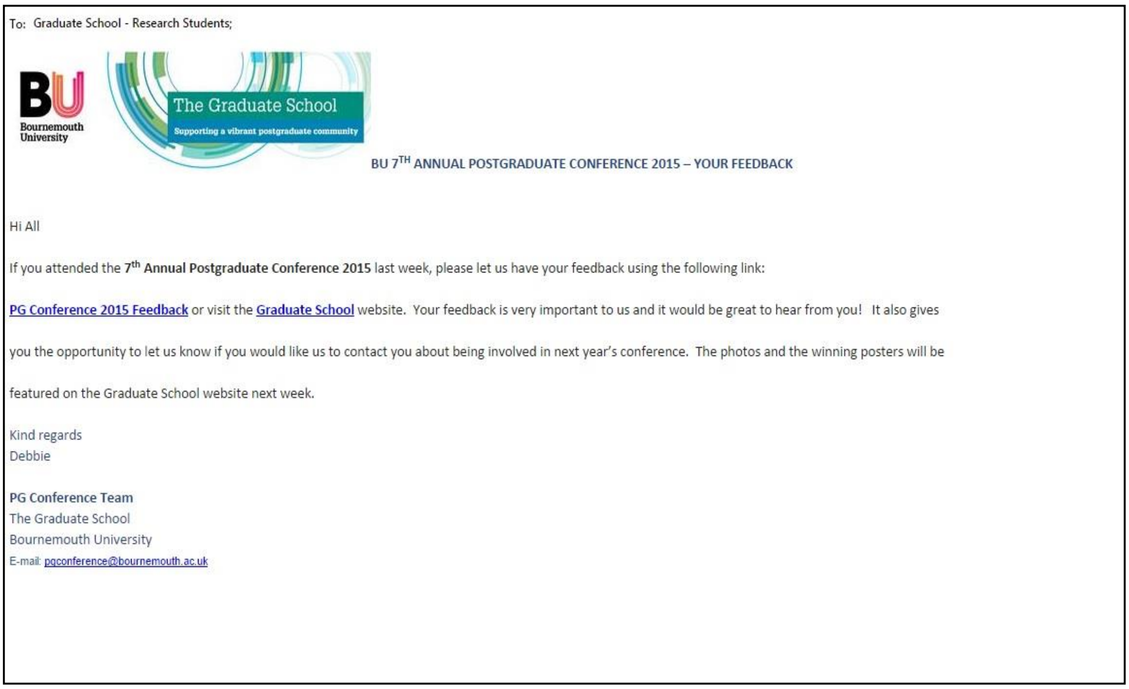
i. Case study: Scenario 1- *Feedback System at the Graduate School, Bournemouth University(BU)*

The scenario provided below is a result of an interview conducted with a member from Graduate School who is in a direct involvement with the feedback collection process at the school. The main aim of the interview was to understand how the feedback system at the school works.

The mission of the Graduate school at Bournemouth University (BU) is to provide support to all BU postgraduate students, whether academically or by enhancing the student experience (e.g. social events). Therefore, they are normally encouraged by the Heir Education Academy (HEA) in the UK to always hear from students about their experience, needs and concerns to inform future improvements. This is done by conducting online surveys using designed by an online tool called Bristol Online Survey. The survey design and questions asked are normally informed by the HEA to ensure appropriate and understandable questions are being introduced taking into account the targeted population. For example, wording and language used for international students are carefully designed to ensure they are culturally appropriate and understandable by students who do not speak English as their first language.

From time to time, postgraduate students receive emails from the Graduate School inviting them to take part in online surveys. Typically, the surveys are designed to hear students' suggestions and complains or to measure their satisfaction about the services provided by the school. The survey invitation emails contain the school logo, a short introduction about the survey purpose and a link to the survey form to fill in (see Figure 1). The surveys normally run between March and May in which students would have more free time to take part in the survey. After allowing sometime since the first invitation, students are then sent personalised reminders to consider taking part in the survey if they have not already done. Those reminders are believed by the interviewed member to increase the response rate to the survey requests. The school also provide incentives (Amazon vouchers) for time consuming surveys in order to maintain a good response rate.

However, the survey/feedback requests provided by the graduate school are restrictive in their design. They do not enjoy a high level of adaptivity to the students' different types of behaviours and preferences with regards to feedback provision. For example, feedback is being asked by emails only whereas some students might prefer to be asked for feedback using a different method such as pop ups or not to be asked for feedback at all. The interviewed member commented in this regard "It is still quiet restrictive and inflexible the way we did it by email". Another example could be the missing social elements in the feedback requests which can improve students/users response rate and experience such as the visibility of others' feedback. A further example, no back-feedback channel is introduced to students who seek to see the impact of their feedback on the services provided by the school. This can harm their experience and trust in having their voice heard by the school. This lack of adaptivity could highly harm customers' satisfaction and experience which can then harm the Graduate School reputation among students in the long-term (see Chapters 3, 4).



To: Graduate School - Research Students;

BU 7TH ANNUAL POSTGRADUATE CONFERENCE 2015 – YOUR FEEDBACK

Hi All

If you attended the **7th Annual Postgraduate Conference 2015** last week, please let us have your feedback using the following link:

PG Conference 2015 Feedback or visit the Graduate School website. Your feedback is very important to us and it would be great to hear from you! It also gives you the opportunity to let us know if you would like us to contact you about being involved in next year's conference. The photos and the winning posters will be featured on the Graduate School website next week.

Kind regards
Debbie

PG Conference Team
The Graduate School
Bournemouth University
E-mail: pgconference@bournemouth.ac.uk

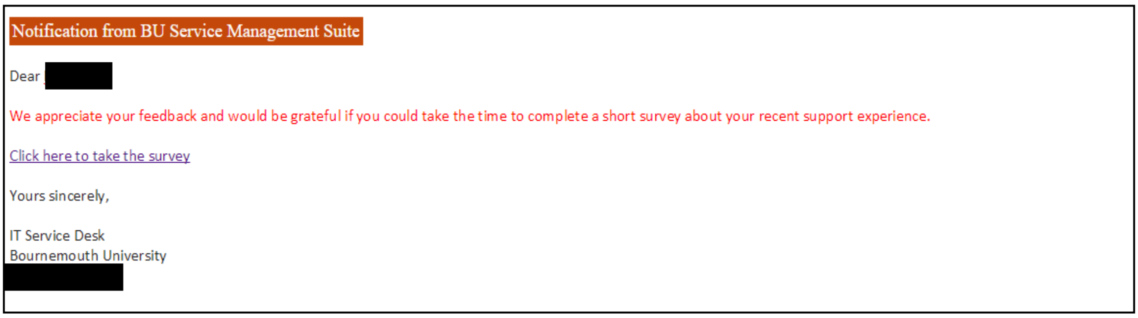
FIGURE 72 A SNAPSHOT EXAMPLE OF ONE THE GRADUATE SCHOOL FEEDBACK REQUEST BY EMAIL.

ii. Case study: Scenario 2- *IT service desk Feedback System at Bournemouth University (BU)*.

The scenario provided below is a result of an interview conducted with a member from IT service desk who is in a direct involvement with the feedback collection process at the service desk. The main aim of the interview was to understand how the feedback system at the IT service desk works.

The IT Services at BU is responsible for providing academic and administrative computing, networking and applications, media services and support at BU. It encompasses all activities which use information technology in order to deliver a service, provides or processes information and includes support for users. They are highly concerned about the quality of the service they provide and the satisfaction of their users. Therefore, they always make sure the voice of users is always heard to inform future improvements. When an IT incident or request that was created or reported by the users (student or staff member) is closed, a feedback email/request is automatically sent out to the user. The survey/feedback invitation email contains the IT service desk logo, invitation to the user to take part in a short survey on the quality of the service provided regarding their recent IT incident/request and a link to the survey form to fill (see Figures 2 and 3).

However, the feedback requests/emails sent by the IT service desk are designed for all costumers universally. They are static and restrictive in their design and do not enjoy a high level of adaptivity to the users' different types of behaviours and preferences with regards to feedback provision. For example, feedback is being asked by emails only whereas some students might prefer to be asked for feedback using a different method such as pop ups or not to be asked for feedback at all. Another example could be the missing social elements in the feedback requests which can improve students/users response rate and experience such as the visibility of others feedback (see Chapters 3, 4). An additional example, no back-feedback channel is introduced to users who seek to see the impact of their feedback on the services provided by the IT service desk which can harm their experience and trust their voice is being heard. This resulted in the IT service desk suffers from a low response rate as believed by the interviewed member. In addition, this lack of adaptivity could highly harm customers' satisfaction and experience which can then harm the IT service reputation among students in the long-term (see Chapters 3, 4).



Notification from BU Service Management Suite

Dear [redacted]

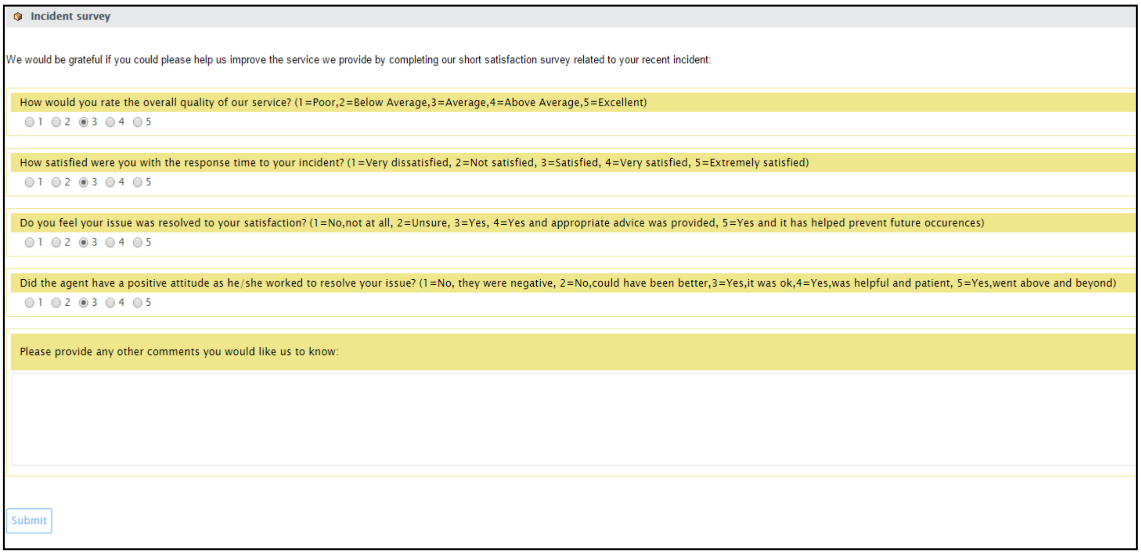
We appreciate your feedback and would be grateful if you could take the time to complete a short survey about your recent support experience.

Click here to take the survey

Yours sincerely,

IT Service Desk
Bournemouth University
[redacted]

FIGURE 73 A SNAPSHOT OF THE FEEDBACK REQUEST SENT BY THE IT HELP DESK.



Incident survey

We would be grateful if you could please help us improve the service we provide by completing our short satisfaction survey related to your recent incident:

How would you rate the overall quality of our service? (1=Poor,2=Below Average,3=Average,4=Above Average,5=Excellent)

○ 1 ○ 2 ● 3 ○ 4 ○ 5

How satisfied were you with the response time to your incident? (1=Very dissatisfied, 2=Not satisfied, 3=Satisfied, 4=Very satisfied, 5=Extremely satisfied)

○ 1 ○ 2 ● 3 ○ 4 ○ 5

Do you feel your issue was resolved to your satisfaction? (1=No,not at all, 2=Unsure, 3=Yes, 4=Yes and appropriate advice was provided, 5=Yes and it has helped prevent future occurences)

○ 1 ○ 2 ● 3 ○ 4 ○ 5

Did the agent have a positive attitude as he/she worked to resolve your issue? (1=No, they were negative, 2=No,could have been better,3=Yes,it was ok,4=Yes,was helpful and patient, 5=Yes,went above and beyond)

○ 1 ○ 2 ● 3 ○ 4 ○ 5

Please provide any other comments you would like us to know:

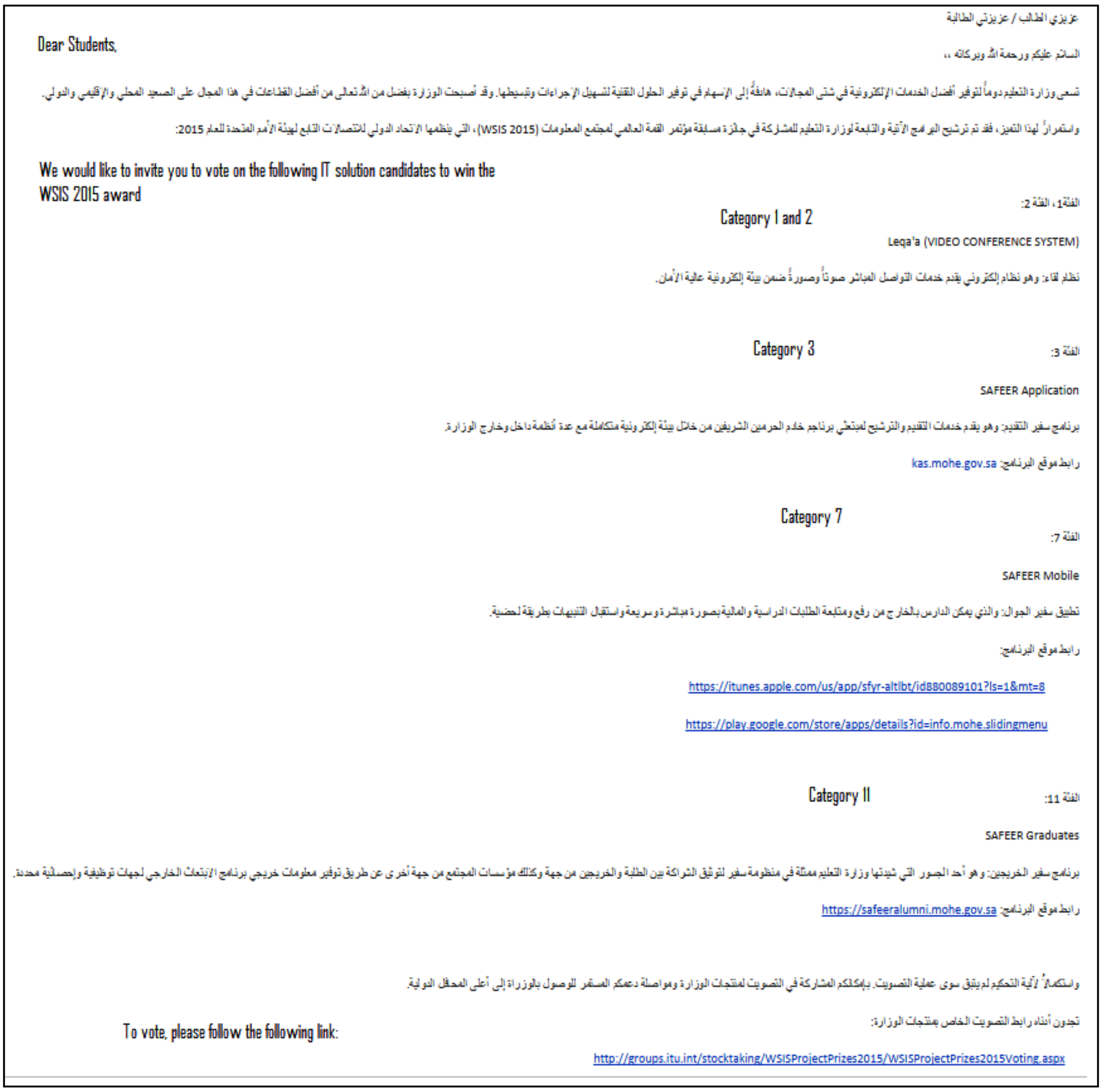
FIGURE 74 AN EXAMPLE OF THE IT FEEDBACK FORM.

Case study: Scenario 3- *Feedback System of the Student Portal for Saudi Students in the UK (Safeer)*

The scenario provided below is a result of an interview conducted with a member from the IT service desk at the Saudi Cultural Bureau in London who is in a direct involvement with the feedback collection process at Safeer. The main aim of the interview was to understand how the feedback system at Safeer works.

The mission of Safeer is to provide IT services to Saudi students studying in the UK to help them managing their scholarships online (e.g. requesting letters, requesting financial sponsorships, etc). The portal also offers several services to students, such as Q&A, Recommended Universities, etc. From time to time, students receive real-time feedback pop ups once they entered Safeer. Typically, the pop ups ask the students to rate the provided IT service and comment, complain or suggest improvement if needed. In addition, from time to time the students receive feedback emails from Safeer asking them to vote on the IT services provided (see Figure 4).

However, the survey/feedback requests provided by Safeer are restrictive in their design. They do not enjoy a high level of adaptivity to the students' different types of behaviours and preferences with regards to feedback provision. For example, feedback is being asked offline (emails) and real-time (pop ups) without catering for some students experience and preferences (e.g. student who prefer emails might find pop ups intrusively harmful to their experience). In addition, some students might prefer not to be asked for feedback at all. Another example could be the missing social elements in the feedback requests which can improve students/users response rate and experience such as the visibility of others feedback .A further example, no back-feedback channel is introduced to students who seek to see the impact of their feedback on the services provided by Safeer which can harm their experience and trust their voice is being heard. This lack of adaptivity could highly harm students' satisfaction and experience which can then harm Safeer's reputation among students in the long-term (see Chapters 3, 4).



عزيزي الطالب / عزيزتي الطالبة

Dear Students,

السلام عليكم ورحمة الله وبركاته ،،

تسعى وزارة التعليم دوماً لتوفير أفضل الخدمات الإلكترونية في شتى المجالات، هادفةً إلى الإسهام في توفير الحلول التقنية لتسهيل الإجراءات وتبسيطها. وقد أصبحت الوزارة بفضل من الله تعالى من أفضل القطاعات في هذا المجال على الصعيد المحلي والإقليمي والدولي.

واستمراراً لهذا التميز، فقد تم ترشيح البرامج الآتية والتابعة لوزارة التعليم للمشاركة في جائزة مسابقة مؤتمر القمة العالمي لمجتمع المعلومات (WSIS 2015)، التي ينظمها الاتحاد الدولي للاتصالات التابع لهيئة الأمم المتحدة للعام 2015:

We would like to invite you to vote on the following IT solution candidates to win the
WSIS 2015 award

الفئة1، الفئة 2:

Category 1 and 2

Leqa'a (VIDEO CONFERENCE SYSTEM)

نظام لقاء: وهو نظام إلكتروني يقدم خدمات التواصل المباشر صوتاً وصورةً ضمن بيئة إلكترونية عالية الأمان.

Category 3

الفئة 3:

SAFEER Application

برنامج سفير التقديم: وهو يقدم خدمات التقديم والترشيح لمبتعثي برنامج خادم الحرمين الشريفين من خلال بيئة إلكترونية متكاملة مع عدة أنظمة داخل وخارج الوزارة.

رابط موقع البرنامج: kas.mohe.gov.sa

Category 7

الفئة 7:

SAFEER Mobile

تطبيق سفير الجوال: والذي يمكن الدارس بالخارج من رفع ومتابعة الطلبات الدراسية والمالية بصورة مباشرة وسريعة واستقبال التنبيهات بطريقة لحظية.

رابط موقع البرنامج:

https://itunes.apple.com/us/app/sfyr-altlbt/id880089101?ls=1&mt=8

https://play.google.com/store/apps/details?id=info.mohe.slidingmenu

Category 11

الفئة 11:

SAFEER Graduates

برنامج سفير الخريجين: وهو أحد الجسور التي شيدتها وزارة التعليم ممثلة في منظومة سفير لتوثيق الشراكة بين الطلبة والخريجين من جهة وكذلك مؤسسات المجتمع من جهة أخرى عن طريق توفير معلومات خريجي برنامج الابتعاث الخارجي لجهات توظيفية وإحصائية محددة.

رابط موقع البرنامج: https://safeeralumni.mohe.gov.sa

واستكمالاً لآلية التحكيم لم يتبق سوى عملية التصويت. بإمكانكم المشاركة في التصويت لمنتجات الوزارة ومواصلة دعمكم المستمر للوصول بالوزارة إلى أعلى المحفل الدولية.

تجدون أدناه رابط التصويت الخاص بمنتجات الوزارة:

To vote, please follow the following link:

http://groups.itu.int/stocktaking/WSISProjectPrizes2015/WSISProjectPrizes2015Voting.aspx

FIGURE 75 SNAPSHOT OF ONE OF SAFEER'S FEEDBACK EMAILS.

3- Procedure:

- You should have received a copy of Feedback Acquisition Personas, PAFA Method Manual and Consent Form in advance. This is hoped to help you familiarize yourself with the study context and what is expected from you before the study takes place (please notify me if you haven't received any of the above or you did not have time to read them).
- An individual from your team (can be nominated by the group before starting the study) will act as a client representing one of the previous case study scenarios and will have the option to select the preferred scenario. The representative will also assign one persona to you. You will then refer to PAFA Method Manual to individually develop for your assigned/selected persona following the steps/guidelines provided in the next section.
- You should keep an active engagement with the client throughout the study for verification and consultation purposes (to do so please follow the suggestion in the next section).
- During the evaluation process, you will be asked to answer a survey questions on each component of PAFA as well as PAFA as a whole while they are using the method.
- Additionally, you will be continuously observed during the evaluation process to access more information about how PAFA is being used, ask/answer questions and offer clarifications when needed.
- A follow up 30 minutes group discussion will be then held to discuss participants' experience with PAFA. This will meant to let you freely express your feelings/experience about PAFA

which will add clearer, collective and richer judgments on PAFA which will positively impact the refinement process.

Thank you for helping me out. With your permission I will be observing, taking notes, asking questions, voice recording part of or the full session so that I can review it later for analysis and research purposes only.

Do you have any questions regarding the information letter or the consent form?

Do you agree to volunteer?

○ Yes
○ No

4- Questionnaire:

General Questions

1- Please list your affiliations:
……………………………………………………………………………………………………

2- What is the highest degree you have received?
Bachelor Degree
MSc Degree
PhD Degree
Other (please specify) ____________________

3- Number of years working in the domain of software engineering (please insert numbers)
………………………………………………………………………………

4- Please indicate your level of expertise in relation to the following:

	low (e.g., no direct experience, anecdotal knowledge only)	medium (e.g., some direct experience, but wide reading)	high (e.g., primary focus of my professional work or have advanced knowledge)
Requirement Engineering in General	○	○	○
Goal Modelling	○	○	○
Feature Modelling	○	○	○
UML modelling	○	○	○

PAFA Components Questions

Components	Steps
Persona to Scenarios	**Step1:** Guided by PAFA manual (section 1), let's try to generate a minimum of 3 scenarios from your allocated persona (please use the provided booklet to write the scenarios).

Questions on *Step1*: On a scale of 1 to 5 please answer the following questions:

Q1: Enough information was provided by PAFA report to define the concept of scenarios and why to use them in the context of PAFA:

●	●	●	●	●
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Q2: Enough information was provided by PAFA report on how to develop scenarios from personas (e.g. important elements which each scenario should include were mentioned in the report):

● Strongly Agree ● Agree ● Neutral ● Disagree ● Strongly Disagree

Q3: The developed scenarios gave a better understanding of the persona's requirements (e.g. better understanding of the persona's behaviour in different context):

● Strongly Agree ● Agree ● Neutral ● Disagree ● Strongly Disagree

Q4: The persona's narrative was helpful enough to help me generate the scenarios (e.g. the behaviour description of the persona was rich of helpful information):

● Strongly Agree ● Agree ● Neutral ● Disagree ● Strongly Disagree

Q5: It was hard and time-consuming process to come up with the scenarios:

● Strongly Agree ● Agree ● Neutral ● Disagree ● Strongly Disagree

Q6: The persona was generic and a transition from personas to scenarios was needed in order to have a sharper view about persona's requirements:

● Strongly Agree ● Agree ● Neutral ● Disagree ● Strongly Disagree

Please elaborate on your answers and add any comments you might have regarding this step:
……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
…………………………………………………………………

Intermediate step: Please consult the client on the developed scenarios and refine them, add more scenarios or delete some if needed.

Scenarios to Goal Models	**Step2:** Guided by PAFA manual (section 2), let's try to develop a Goal Model/Models for one or more of the scenarios we developed in ***Step1***. (Please use the provided booklet to draw the model).

Questions on *Step2*: On a scale of 1 to 5 please answer the following questions:

Q1: Enough information was provided by PAFA report to define the concept of Goal Model and why to use them in the context of PAFA:

● Strongly Agree ● Agree ● Neutral ● Disagree ● Strongly Disagree

Q2: Enough information was provided by PAFA report on how to derive a Goal Model from scenarios (e.g. references to known approaches for deriving a goal model from scenarios were mentioned):

● Strongly Agree ● Agree ● Neutral ● Disagree ● Strongly Disagree

Q3: The resulted goal model provided a better understanding of the persona's goals/requirements (e.g. better view of a persona's goals and soft-goals):

● Strongly Agree ● Agree ● Neutral ● Disagree ● Strongly Disagree

Q4: The scenarios developed in ***Step1*** were helpful and informative enough to help me derive the goal model (e.g. no struggle in identifying persona's goals from the scenarios):

●	●	●	●	●
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Q5: It was hard and time-consuming process to develop the goal model from the scenarios (e.g. goals and soft-goals were hard to identify):

●	●	●	●	●
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Q6: The scenarios developed in ***step1*** were still generic and a transition from scenarios to goal models was needed in order to have a sharper view about persona's goals, soft-goals and how to reach them:

●	●	●	●	●
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Please elaborate on your answers and add any comments you might have regarding this step:

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

…………………………………………………

Goal Model to Use-case	**Step3:** Guided by PAFA manual (section 3), let's try to develop use-case/cases based on previously developed goal model in ***step2*** (please use the provided booklet to draw the use-case).

Questions on *Step3*: On a scale of 1 to 5 please answer the following questions:

Q1: Enough information was provided by PAFA report to indicate the purpose of adopting use-cases in the context of PAFA:

●	●	●	●	●
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Q2: Enough information was provided by PAFA report on how to create a use-case enlightened by goal models (e.g. an overview and references of known approaches for deriving a use-case from goal models were mentioned):

●	●	●	●	●
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Q3: The developed use-cases gave a better understanding of the persona's requirements (e.g. it gave me a clearer view about the personas potential interaction with the software when being asked for feedback):

●	●	●	●	●
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Q4: The goal model developed in **step 2** was helpful enough to help me develop the use-case (e.g. the way to reach goals inform the potential users' interaction with the system):

●	●	●	●	●
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Q5: It was hard and time-consuming process to come up with the use-cases (e.g. the goal model did not provide enough information to build a complete use-case):

●	●	●	●	●
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Q6: The transition from the goal models in ***step 2*** to use-cases was needed in order to capture the persona's potential interaction with the system:

○ Strongly Agree ○ Agree ○ Neutral ○ Disagree ○ Strongly Disagree

Please elaborate on your answers and add any comments you might have regarding this step:
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
…………………………………………………

Adaptation Engineering	**Step4:** Guided by PAFA manual (section 4.c), let's try to develop a feature model enlightened by the information yielded from the previous steps ***(step1, step2 and step3)*** (please use the provided booklet to draw the feature model).

Questions on *Step4*: On a scale of 1 to 5 please answer the following questions:

Q1: Enough information was provided by PAFA report to define the concept of adaptation engineering and why to use them in the context of PAFA:

○ Strongly Agree ○ Agree ○ Neutral ○ Disagree ○ Strongly Disagree

Q2: Enough information was provided by PAFA report on how to develop a feature model to cater for variability based on the information collected in the previous steps/phases:

○ Strongly Agree ○ Agree ○ Neutral ○ Disagree ○ Strongly Disagree

Q3: The developed feature model gave a better understanding of the persona's requirements (e.g. it shows personas' preferred features with regard to feedback acquisition in a certain context):

○ Strongly Agree ○ Agree ○ Neutral ○ Disagree ○ Strongly Disagree

Q4: The previous steps/phases were helpful enough to help me develop the feature model (e.g. scenarios, goal models and use-cases provided enough information to identify the persona's preferred features in different contexts):

○ Strongly Agree ○ Agree ○ Neutral ○ Disagree ○ Strongly Disagree

Q5: It was a time-consuming process to develop the feature model (e.g. too much information yielded from the previous steps which takes a long time to go through and come up with the features) :

○ Strongly Agree ○ Agree ○ Neutral ○ Disagree ○ Strongly Disagree

Q6: The feature model was needed in order to sum up the previous steps in more clearer view of the persona's preferred requirements/features that can even be discussed with and understood by the clients:

○ Strongly Agree ○ Agree ○ Neutral ○ Disagree ○ Strongly Disagree

Please elaborate on your answers and add any comments you might have regarding this step:
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………

……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
…………………………………………………………………

Intermediate step: Please consult the client on the developed feature models and add, eliminate some feature if needed by the client.

Evolution	**Step5:** Let's try to refine and modify your assigned persona using the information yielded from the previous steps ***(step1, step2, step3 and step4)*** (please annotate the original persona with the appropriate information or comments).

PAFA Overall Questions

Please answer the following questions about PAFA as a whole:

Q1: Enough information is given to explain the purpose of PAFA method and how it was developed?

●	●	●	●	●
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Please elaborate on your answer (optional):

……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
………………………

Q2: Enough information is given to explain the importance of PAFA method (e.g. its ability to cater for different behaviours to feedback)?

●	●	●	●	●
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Please elaborate on your answer (optional):

……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
………………………

Q3: Generally speaking, PAFA method was not hard to understand (e.g. it was explained in a clear way):

●	●	●	●	●
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Please elaborate on your answer (optional):

……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
………………………

Q4: Overall, PAFA method was not hard to use :

●	●	●	●	●
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Please elaborate on your answer (optional):

……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
………………………

Q5: Overall, using PAFA method can be a time-consuming process:

●	●	●	●	●
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Please elaborate on your answer (optional):

……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………

………………………

Q6: Based on your experience (e.g. teaching new software methods to students), do you think PAFA can be easily learnt by novice software engineers:

Yes

No

Please elaborate on your answer (optional):
……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
………………………

Q7: Is a change to one component of PAFA likely to require a change to the other components/phases? (E.g. if you go back during the development process to one of the components and update it with more information about the persona (e.g. more scenarios are introduces), would this require you to go again through the following components to update them too if needed?

Yes

No

Please elaborate on your answer (optional):
……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
………………………

Q8: Do you think more components/phases need to be added to PAFA to further improve it (e.g. more UML models to be added to further express requirements or more adaptation engineering methods should be added to cater for variability)?

Yes

No

Please elaborate on your answer (optional):
……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
………………………

Q9: Do you think PAFA is complex and some of its components/phases need to be taken out (e.g. the use-case phase does not add much to PAFA)?

Yes

No

Please elaborate on your answer (optional):
……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
………………………

Which phase was the hardest to implement and why?
……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
………………………

Q10: Do you think the sequential transition between PAFA components provides an effective way for identifying various requirements of users' to feedback acquisition (e.g. moving from step 1 to step 4, in each phase a sharper and clearer view of users requirements was observed)? Please elaborate

Yes

No

Please elaborate on your answer (optional):
……………………………………………………………………………………………………
……………………………………………………………………………………………………

………
………………………

Q11: Does PAFA show applicability and scalability to be adopted in different application domains to design feedback acquisition such as health-critical systems or privacy-critical systems?

Yes

No

Please elaborate on your answer (optional):
………
………
………
………………………

Q12: Each one of PAFA component/phases has already a rich literature to refer to for more information when needed:

●	●	●	●	●
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Q13: Does each phase in PAFA provide a solid foundation towards the following phase (e.g. personas description is helpful enough to develop multiple scenarios)?

Yes

No

Please elaborate on your answer (optional):
………
………
……………………………………………………………………………………………………

Please elaborate on your answers and add any comments you might have regarding PAFA:
………
………
………
………
………
………
………
…………………………………………………………

11.7 APPENDIX 7

Examples on PAFA- A Design Method for Adaptive Feedback Acquisition

Example 1:

Persona

Linda

Profile: Privacy tolerant and socially ostentatious
Age:20
Gender: Female
Job: Undergraduate student
Socially affected to give feedback: Yes
Culture Suitability: Middle Eastern-like

Statement: *"Giving feedback is a social and community experience and it helps to feel among others".*
Goals: Impact the software with her feedback + raising others awareness about the used software + being socially recognised.

Behaviour to feedback:
Linda is an undergraduate university student and spends a great deal of time on her computer studying as well as heavily social networking (i.e. Facebooking). **[Discouragement]** In general, she is not a big fan of the idea of dull and typical feedback requests and reminders coming from software applications. **[Motivation]** However, she gets interested in replying to feedback requests when the feedback requests socially motivate her to do so (i.e. by making her socially recognized for her helpful feedback). This is perhaps due to her likeness of social networking and the time she spends socialising with others/friends on the internes which made her motivated towards socially enriched feedback requests. Generally, Linda is positively affected by one or more of the following social factors to give feedback:

- **Volume of already given feedback**: She gets enthusiastic to give feedback when there is low number of feedbacks already given on a software. She believes it's helpful to increase the number of given feedback which will then result in other users having a better and richer idea about the software.
- **Visibility and similarity of other users' feedback**: Linda also gets more interested to give feedback if she is able to see other users' feedback on the software first and then having the option to accept/reject to give feedback.
- **Social recognition:** Since Linda appreciates social networking and gives it a great deal of her time, she likes to be socially recognized for her given feedback which she believes could help others and make her socially popular.
- **Feedback acquisition as a social activity:** This social factor also makes Linda motivated to give feedback as well as engaging with software. For example, she gets enthusiastic to feedback requests when she is able to visualize how her social friends are rating a certain software and how their feedback influenced the trend in her community.

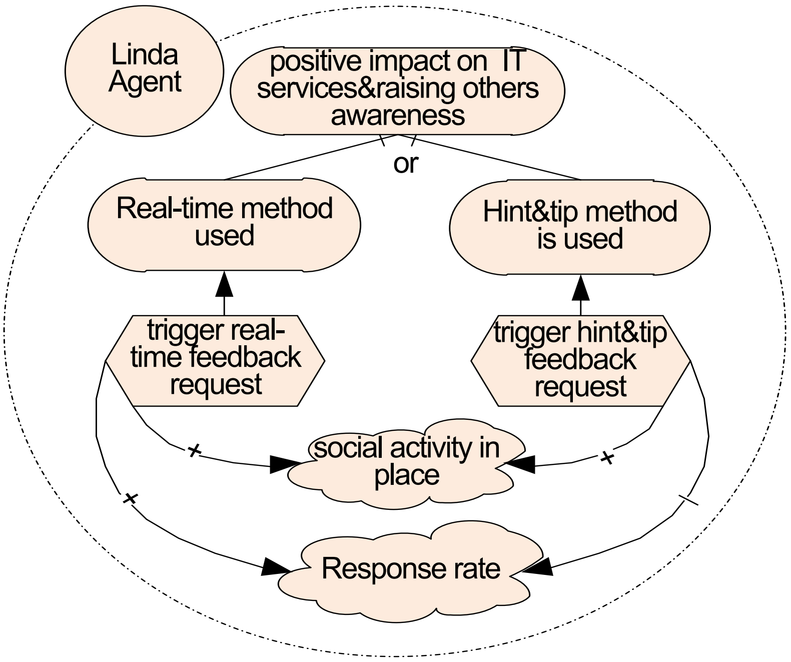
[Method] In addition, Linda prefers to be approached for feedback by using hints and tips to gather her feedback (e.g. by telling her that she can go to a feedback centre for this purpose and leave her feedback) or by using an online method as a second option (i.e. popups while she is using the software). **[Privacy]** Interestingly, Linda does not mind to be implicitly reached for feedback (e.g. implicitly collecting information about her software usage)

Scenarios

- **Scenario 1:** Linda calls the IT desk to request a password change of her university account. The IT Desk addresses her request and then emails (off-line method) her a feedback request. The feedback request is to evaluate her experience and satisfaction with the recent provided IT service. The feedback request contains a link to a feedback form and does not contain any social aspect in it (i.e. volume of already given feedback). Linda's willingness decreases to give feedback and she ignores the request.

- **Scenario 2:** Linda calls the IT desk to request a password change of her university account. Right after her request was addressed a pop up feedback request is triggered to ask her about her satisfaction with the service provided. The pop up shows the number of students liked and disliked the IT provided IT services. It also shows how her classmates rated the service and

Goal Model

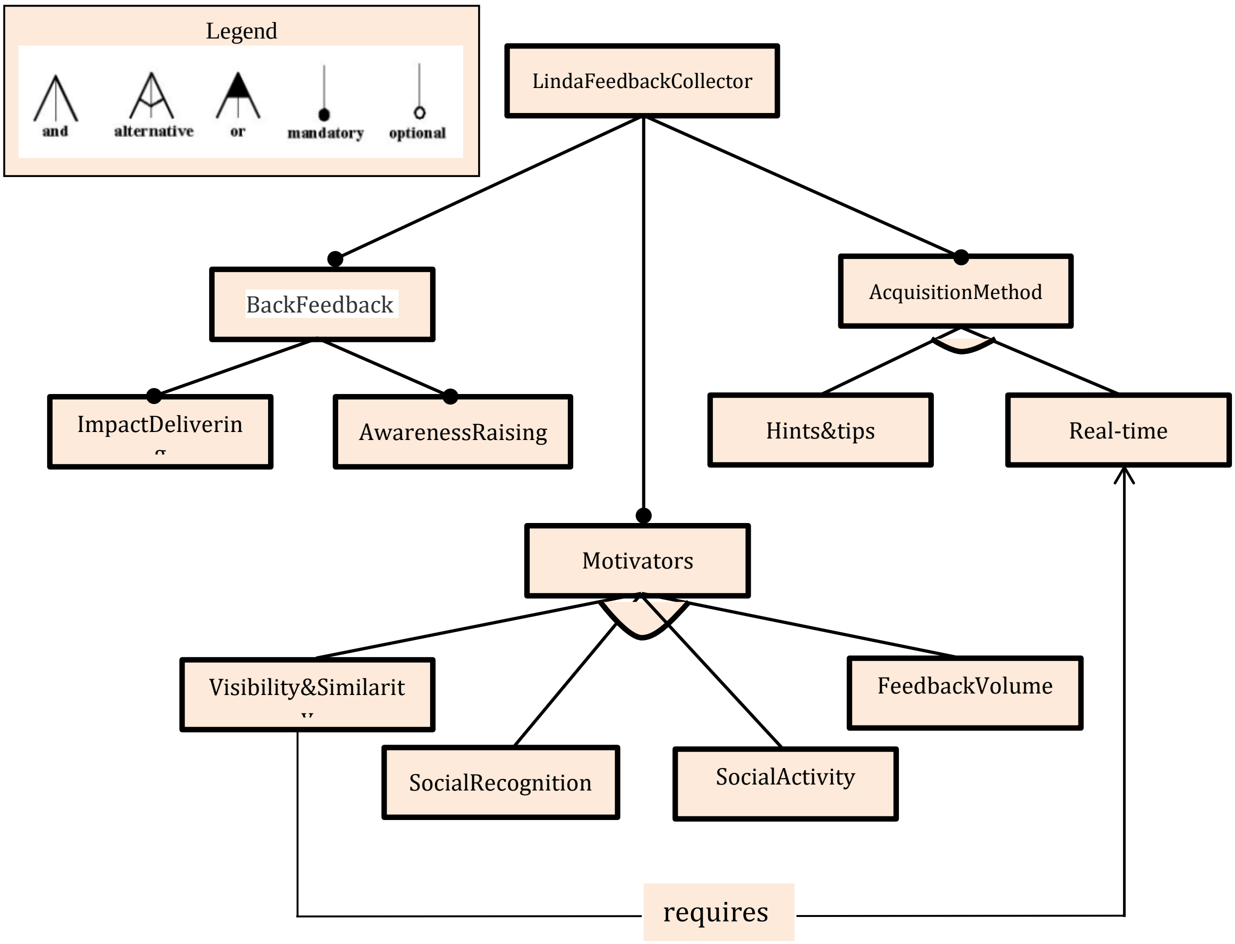

Linda Agent
positive impact on IT services&raising others awareness
or
Real-time method used
Hint&tip method is used
trigger real-time feedback request
trigger hint&tip feedback request
social activity in place
Response rate

Example of potential scenario-based goal model yielded from previous scenarios

Use-case (Linda's interactions with the system)

Use Case Name	Supply Feedback	
Trigging Event	Calling for a password change	
Brief Description	When a student requests a password change a feedback request is sent to the student after addressing the request asking for their overall satisfaction with the service provided.	
Actors	Linda	
Preconditions	Actor requests a password change.	
Post conditions	Actor receives feedback requests and takes actions.	
Flow of events	**Actors**	**system response**
	Linda	**To Linda**
	Call the IT automated service. Ask for a password change. Receives a response from the provider. Receives a pop up from IT asking for feedback after logging in to her PC. Clicks on the link to the feedback form. Gives feedback and clicks submit.	1 Answers Linda's call. Ask verification questions. Take action. Sends a pop up feedback requests after Linda's logs in. 6.1 send occasional updates about the impact of the given feedback on the service.
Exception conditions	5.1 if Linda does not click on the link, a reminder pop up is shown by the feedback system the next time she logs in. 5.2 if Linda does not continue the feedback form and clicks submit, a reminder pop up is shown the next time she logs in.	

Feature Model



Example 2:

Persona

Jack

Profile: Privacy fanatic and generous
Age:35
Gender: Male
Job: Researcher
Socially affected to give feedback: Yes
Culture Suitability: Middle Eastern-like

Statement: *"I think emails are good if you want someone to actually sit down and write a couple of sentences about how they feel about your service popups and other 'push' mechanisms intrude & interrupt flow."*

Goals: Impact the software with his feedback + raising others awareness about the used software + being socially recognised.

Behaviour to feedback:

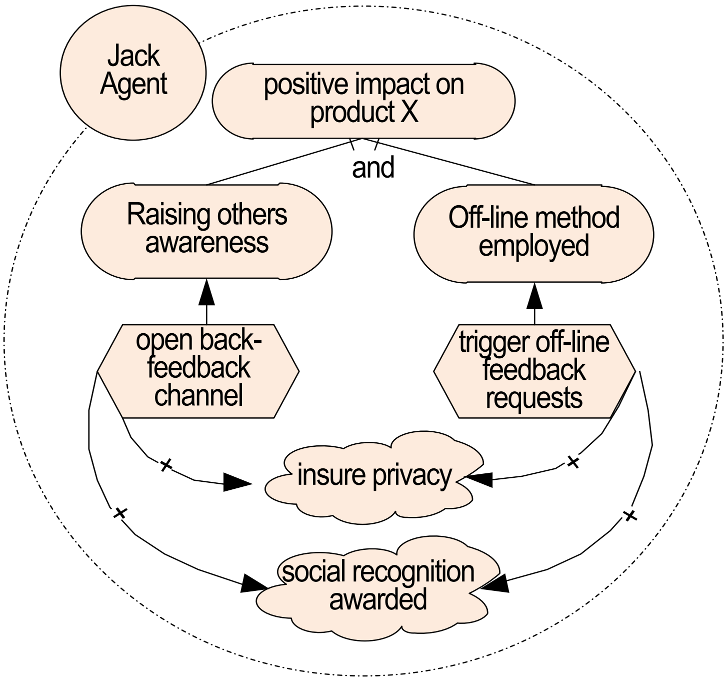
Jack as a researcher spends most of his time on the computer working on his research as well as networking with other researchers. **[Motivation]** Jack believes in the power feedback in general and its positive impact. He is a very positive person towards feedback requests and reminders coming from software application.
[Method] However, he prefers to be asked for feedback in an offline way (i.e. through emails or text messages).
[Discouragement] He believes online feedback request (i.e. popups) could somehow be intruding and interrupting especially when he is working on his research and deeply thinking.
[Privacy] In addition, Jack is always concerned about his privacy and therefore he does not accept to implicitly collect feedback from him (i.e. tracking his usage of the software). **[Motivation]** In addition, Jack is a socially motivated feedback provider and his willingness to give feedback is positively influenced by one or more of the following social factors:

- **Social recognition:** He likes to be socially recognized for his valuable and trustworthy feedback which he believes could help others and raise the social awareness about the software in use.
- **Volume of already given feedback:** He gets enthusiastic to give feedback when there is high number of feedbacks already given on a software. This means to jack the software is popular and deserves his feedback.
- **Visibility of other users' feedback:** Jack also gets more interested to give feedback if he is able to see other users' feedback on the software first and then having the option to accept/reject to give feedback.

Scenarios

- **Scenario 1:** Jack requests a software installation by calling the IT desk. The IT Desk addresses his request and then shows him a pop up asking for his feedback right after he logs into his PC. The feedback request is to evaluate his experience and satisfaction with the recent provided IT service. The pop up does not indicate how Jack's feedback is going to be used and whether his identity is going to be anonymised. In addition, Jack does not appreciate pop ups so his willingness decreases to give feedback and he ignores the request.
- **Scenario 2:** Jack requests a software installation by calling the IT desk. The IT Desk addresses his request and then sends him an email asking for his feedback right after providing the service. The feedback request is to evaluate his experience and satisfaction with the recent provided IT service. The email shows the how his feedback is going to raise others awareness about the service and it also gives him the option to configure his privacy preferences (e.g. exposing his identity to his private social group). Jack's willingness increases and responds to

Goal Model

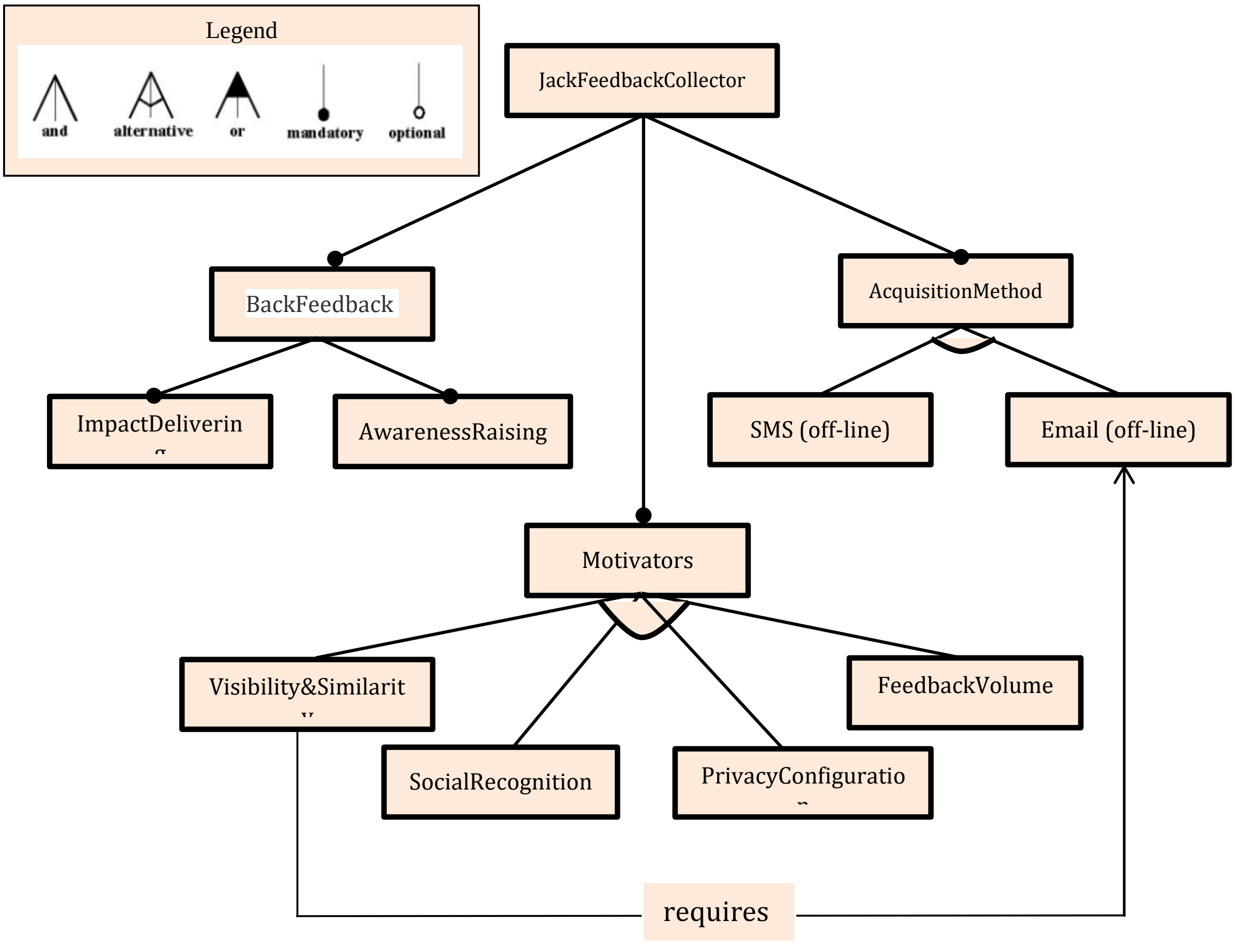

Jack Agent
positive impact on product X
and
Raising others awareness
Off-line method employed
open back-feedback channel
trigger off-line feedback requests
insure privacy
social recognition awarded

Example of potential scenario-based goal model yielded from previous scenarios

Use-case (Linda's interactions with the system)

Use Case Name	Supply Feedback	
Trigging Event	Requesting a software installation	
Brief Description	When a student requests a software installation a feedback request is sent to the student after addressing the request asking for their overall satisfaction with the service provided.	
Actors	Jack	
Preconditions	Actor requests a software installation.	
Post conditions	Actor receives feedback requests and takes actions.	
Flow of events	**Actors**	**Feedback system response**
	Jack	**to Jack**
	Call the IT automated service. Ask for a software installation. Receives a response from the provider. Receives an email from IT asking for feedback after addressing his request. Clicks on the link to the feedback form. Gives feedback and clicks submit.	Answers Jack's call. Ask verification questions. Take action. Sends an offline feedback requests by email. Includes a link to the feedback form in the email. Explains how the collected feedback is going to be used. Assures privacy options are configurable by Jack. 6.1 sends occasional updates about the impact of the given feedback on product x.
Exception conditions	5.1 if Jack does not click on the link, a reminder email is sent by the feedback system. 6.1 if Jack does not continue the feedback form and clicks submit, a reminder email is sent.	

Feature Model



Interview on the Client Point of View of PAFA

Session Opening

At first, Interviewee will be given a sheet of information that explains how the session will go, what they are expected to do, how we will use the information obtained during the session and how they can contact me for further information. In addition, a consent form will also be given to each participant to assure the interviews are ethically undertaken. Afterward, the interviewee will be introduced to the seven personas for feedback acquisitions and an explanation about them will be given. Then a brief overview of how to use those personas for the design of feedback acquisition (PAFA) will be given and illustrated by showing two examples on Linda and Jack personas. The interviewee will be then ready to be taken through the interview questions.

Objectives of the study

To assess the impact of using and adopting PAFA for the design of feedback acquisition in real case studies. The assessment will come from the client (Graduate school and IT desk representatives at BU) point of view as a confirmatory and complementary phase to the previous phase of PAFA evaluation.

Introduction

Thank you for helping me out. I'm interested in investigating how the adoption of our proposed method for the design of feedback acquisition PAFA (as previously explained) could impact you current process for the collection of students' feedback (e.g. after an IT incident is being addressed). With you permission I would like to record the session so that I can review it later for analysis and research purposes only.

Do you have any questions regarding study aim, the information letter or the consent form?

Do you agree to volunteer?

With your permission, I'm going to begin recording now. Remember that there is no wrong answer and every bit of information you can give me is helpful.

Interview Protocol

Time of Interview:
Date:
Department name:
Interviewee Name:
Interviewee role in the department:
Years of experience:

Interview questions:

Based on the previous examples, how do you think PAFA could impact –if adopted- your current feedback collection process?

Please elaborate on your answer

Please explain why do you think this impact can result when adopting PAFA?

On scale of 1to 5, please rate the potential usefulness of PAFA to your current system:

●	●	●	●	●
Strongly Useful	Useful	Neutral	Useless	Strongly Useless

Please elaborate on your answer

On scale of 1to 5, please rate the easiness to adopt PAFA to update your current system:

●	●	●	●	●
Very Straightforward	Straightforward	Neutral	Difficult	Very Difficult

Please elaborate on your answer

Can you think of any challenges that you could face if adopting PAFA?

We recognized the need for businesses to understand their feedback acquisition more and hence we developed PAFA to meet, amongst other thing that objective", how would PAFA be helpful in that? For example (the goal model can show the users' intention to give feedback and whether their intentions are conflicted or in line with your business goals to collect feedback) (e.g. Sara (incentive seekers) could provide low quality feedback. This could go against your goal to collect high quality feedback)

Please elaborate on your answer

We recognize a need of business to cover or engage with different aspects in the design of feedback acquisition and see the whole picture, how would you see the link between the models I just showed you (E.g. moving from step 1 to step 4, in each phase a sharper and clearer view of users' requirements was observed)? and would you think that we need to have any specific considerations when moving between models so we help you more in having a complete and efficient design?

- ○ Yes
- ○ No

Please elaborate on your answer on each phase of PAFA

Generally speaking, what would you criticize about the adoption of PAFA to direct the design of feedback acquisition?

Do you have any other comments you would like to add?

Analysis:

A qualitative analysis will be conducted in which the audio recording is transcribed first and then initial exploration of the gathered data by reading the transcripts will be performed. Then data will be coded by labelling and segmenting the text and then the generated codes will be used to create themes by gathering similar codes together. In the final step the created themes will be compared and connected to each other to make a sensible story.

Participant Information Sheet

Evaluation workshop for PAFA method

You are invited to take part in this research project but before you decide it is important for you to understand why the research is being done and what it will involve. Please take time to read the following information carefully and discuss it with others if you wish. Ask us if there is anything that is not clear or if you would like more information. Take time to decide whether or not you wish to take part. You will be asked to sign on a consent form and at the end of the session you will be given a copy of this information sheet and a copy of the signed consent form.

I am a PhD student at BU and as part of my PhD thesis, I am conducting a research study under the supervision of Dr. Raian Ali. The purpose of the study is to evaluate a feedback acquisition design method we recently proposed (PAFA method).

As a participant of this study, you will take part as a software engineer in a workshop that will last for around 3-4 hours. You are expected to be active at working and engaging with other participants (if needed) in response to the questions raised by PAFA evaluation protocol which was given to you earlier.

Whilst there will be an immediate benefits for you at the individual level by participating in this study (a £40 Amazon voucher), it is hoped that this work will help improving the development process of feedback acquisition in software applications and might also open the gate for collaborative work with you in the future.

You should know that you are free to withdraw and discontinue your participation in this study at any time without it affecting any benefits that you are entitled to in any way. You do not have to give a reason. If you do decide to take part you will be given this information sheet to keep (and be asked to sign a consent form)

I may wish to digitally record the session or take pictures. However, information obtained in this study (audio recordings) will be kept strictly anonymous unless permission is given by the participants to the interviewer for a specific context. The results of this study will be used for research purposes only and presented collectively and no individual participants will be identified without their permissions and I may wish to quote your words directly in reports and publications resulting from this study.

Contact for further information

If you have any questions about this study, please contact me by email on *malmaliki@bournemouth.ac.uk* or contact my supervisor by email on *rali@bournemoth.ac.uk* . Any queries regarding Data Protection should be addressed to the Information Office at *information@bournemouth.ac.uk* .

Participant’s Consent Form

Study Title: Interview on Persona-based Feedback Acquisition

Researcher Information:
Malik AL Maliki, PhD researcher
Faculty of Science and Technology
Bournemouth University
malmaliki@bournemouth.ac.uk

Supervisor Information:
Raian Ali, Senior Lecturer
Faculty of Science and Technology
Bournemouth University
rali@bournemouth.ac.uk

Please Initial Here

I confirm that I have read and understood the participant information sheet for the above research project and have had the opportunity to ask questions.	
I understand that my participation is voluntary and that I am free to withdraw at any time, without giving reason and without there being any negative consequences. In addition, should I not wish to answer any particular question(s), complete a test or give a sample, I am free to decline.	
I give permission for the above named researcher to have access to my anonymised responses. I understand that my name will not be linked with the research materials, and I will not be identified or identifiable in the report or reports that result from the research.	
I agree to take part in the above research project.	
I would like to be sent a copy of the resulted report from this study. (Please print your email address ………………………………………………………………...)	
I confirm that I have received a £40 Amazon voucher as an incentive to take part in this study.	

______________________ ____________ ____________________________

Name of Participant Date Signature

______________________ ____________ ____________________________

Name of Researcher Date Signature