Nelson Mandela Metropolitan University

for tomorrow

Department of Computing Sciences

Designing a Proxemic Natural User Interface to Support Information Sharing among Co-Located Mobile Devices

Timothy Lee Son

Supervisor: Prof. J. L. Wesson

Co-Supervisor: Dr D. Vogts

Submitted in fulfilment of the requirements for the degree of Magister Commercii in Computer Science and Information Systems at Nelson Mandela Metropolitan University

April 2016

Acknowledgements

I would like to thank God, for giving me the strength and courage to endure the challenges encountered during this research. I am grateful to my family and friends for their patience, encouragement, and devoted support throughout this research. I would also like to express my sincere gratitude to my supervisors, Prof. Janet Wesson and Dr Dieter Vogts, for their invaluable assistance, reinforcement, and support they provided throughout this research. Their positivity, remarkable dedication, and attention to detail is also greatly appreciated.

Thank you to Mr Nicholas Barfknecht and Mr Grant Eastwood for their scientific advice and sharing of knowledge, with whom various insightful discussions and suggestions occurred, which proved to be instrumental in completing this research. To Mr Gareth McEwan and Mr George Mujuru, who provided their statistical knowledge in presenting the results of the experiments and evaluations conducted throughout this research, your expertise is much appreciated. I am also grateful to Mrs Teniele Lee Son and Dr Lynette Barnard, who have given their time to comprehensively critique and proofread this dissertation. Special thanks to Mr Uvedale Tristam for his proofreading, which has also significantly improved the composition of this dissertation. To the technical checker, Dr Simone Beets, her expertise, attention to detail, and thorough feedback is greatly appreciated.

I would like to thank the Telkom Centre of Excellence for providing the necessary research funding and equipment to complete this research. Finally, I appreciate the National Research Foundation, for their financial assistance towards this research.

Summary

Existing information sharing methods used by mobile devices require the user to repeat a series of steps to share one or more selected files with another individual, where the entire process is repeated for sharing the same file(s) with multiple individuals. Due to constant advancements in mobile computing, mobile devices are able to provide new, more intuitive, and easier solutions to sharing information.

Natural User Interfaces (NUIs) primarily focus on the reuse of existing knowledge (from other applications or activities) or human abilities (such as touch, speech, and gestures) to provide a more accurate and usable solution to existing human-computer interaction (HCI) systems. The interaction techniques of NUIs have transformed these human abilities. The main research objective was to design a proxemic NUI to provide an accurate and usable solution to support information sharing among co-located mobile devices.

The development of MotionShare supported multiple devices to share information simultaneously using NUI interaction techniques. An initial calibration setup allowed MotionShare to calculate the approximate positions and orientations of every device in the environment. Novel NUI interaction techniques were implemented because of the known positions of these devices.

MotionShare was evaluated using two evaluation techniques, namely analytical and experimental. The results showed device positioning to have a mean precision, trueness, and recall of 72.21%, 91.39%, and 71.63% respectively. The results showed MotionShare gestures to have a recall of 90.50% and 100.00% for the point gesture and the touch gesture respectively. The experimental technique consisted of a pilot study (formative evaluation) and a usability evaluation (summative evaluation). The results of the usability evaluation showed high user satisfaction and statistical analysis, which revealed MotionShare to achieve the main research objective. These results also showed that participants preferred the touch gesture to the point gesture, but expressed both gestures can be utilised for the tasks of MotionShare.

Keywords: Information Sharing, Natural User Interfaces, Indoor positioning, Gesture-Based Interaction, Proxemics, Mobile Computing

Declaration

I, Timothy Lee Son, hereby declare the dissertation, ***Designing a Proxemic Natural User Interface to Support Information Sharing Among Co-Located Mobile Devices***, for Magister Commercii in Computer Science and Information Systems is my own independent work. All sources used or quoted have been indicated and acknowledged by means of complete references. This dissertation has not been previously submitted for assessment to any other university or completion of any other qualification.

Timothy Lee Son

Table of Contents

List of Figures

List of Tables

Glossary

Abbreviation	**Full Definition**
AP	Access Point
API	Application Programming Interface
CLI	Command Line Interface
dBm	Decibel-Milliwatt
DSR	Design Science Research
GPS	Global Positioning Systems
GUI	Graphical User Interface
HCI	Human-Computer Interaction
IDE	Integrated Development Environment
ML	Machine Learning
NFC	Near Field Communication
NMMU	Nelson Mandela Metropolitan University
NUI	Natural User Interface
REC-H	Human Research Ethics Committee
RSSI	Received Signal Strength Indicator
SDK	Software Development Kit
SSID	Service Set Identifier
QR	Quick Response
Wi-Fi	Wireless Fidelity
WIMP	Windows, Icons, Menus, and Pointing devices
XML	Extensible Markup Language

Chapter 1: Introduction

1.1 Background

The need for information sharing arises from the constant communication between users and their mobile devices (Michahelles, 2011). In recent years, the world has witnessed a rapid growth in the computational and data storage capabilities of mobile devices. This has led to an increased information sharing rate between users and their mobile devices. Current information sharing methods on mobile devices are readily available but require a manual process, which can be cumbersome, depending on the amount of information to be transferred and the number of recipients. Therefore, these methods can become time-consuming and ineffective.

According to an eMarketer (2014) report, the number of global smartphone users will surpass two billion by the year 2016. In 2015, an excess of 1.91 billion smartphone users indicated the use of smartphone devices on a daily basis, which is more than one-quarter of the global population. Mobile devices have become an integral part of people's lifestyle, silently operating in the background, to support their work, social interaction, leisure, and collaboration. This is a result of the constant interaction between people and their mobile devices.

Since the emergence of Natural User Interfaces (NUIs), traditional methods of interacting with mobile devices are being replaced with more intuitive NUI interaction techniques. The term *intuitive* implies the ease of understanding a concept without any conscious reasoning (Britton, Setchi, & Marsh, 2013). NUIs facilitate many different touch gestures, such as tap, point, swipe, and drag (Oh, Robinson, & Lee, 2013), which have changed the landscape of human-computer interaction (HCI) forever.

NUIs utilise interaction techniques which are natural and intuitive to the user, making user actions easier to perform, and specific tasks less time-consuming to complete (Oh et al., 2013). NUIs show great potential in defining a new generation of interactive computing (Seow, Morrison, Wixon, & Jacucci, 2010).

Proxemics involves the study of cultural, behavioural, and sociological aspects between individuals and their devices (Dingler, Funk, & Alt, 2015). For this research, proxemics encompasses the distance between the devices, orientation of each device, and the movement of the devices (distance and orientation change).

A NUI can include any of the following interaction techniques:

- Touch and optionally the use of a stylus;
- Gestures;
- Speech; and
- Proxemics.

Since the introduction of NUIs, various users have shown interest in this field, more specifically in gesture interaction techniques and their many potential areas of use (Koueider, 2013). Prominent research in the domain provides a glimpse into the complex and interesting nature of NUIs, which illustrates the broad range of topics incorporating NUIs and highlighting their benefits of use (Lamberti, Sanna, Carlevaris, & Demartini, 2015; Seow et al., 2010; Wigdor & Wixon, 2011).

A *co-located* environment is a gathering of users and their mobile devices in an informal or formal meeting (Heikkinen & Porras, 2013). This environment can take place indoors or outdoors, with users being in close proximity to each other. Proximity of users allows for NUI interaction techniques to support information sharing among co-located mobile devices. For this research, co-located means the users and their mobile devices are in close proximity to each other (indoors or outdoors) but users are not allowed to move around (for example, seated or standing around a table).

Ilić (2010) define *pose* as *"In computer vision and robotics, an object's pose refers to the combination of position and orientation of the object"*. For this research, pose means the location and orientation of a mobile device.

1.2 Research Relevance

As mobile devices are rapidly evolving in their sensory, computational, and data storage capabilities, the need arises for a more intuitive method of sharing information among co-located mobile devices. Existing information sharing methods do not capitalise on the advanced capabilities of mobile devices to provide a more accurate and usable solution for co-located information sharing, such as a proxemic NUI. These methods only use standard communication technologies to share files, which only show the mobile devices are within range to receive the files. As a result, there is a lack of context with the actual locating of the devices.

Preliminary research yielded minimal evidence in the application of NUIs to support information sharing among co-located mobile devices. Therefore, this research focuses on improving co-located information sharing through the development of a proxemic NUI and the relevant interaction techniques.

1.3 Problem Statement

Existing information sharing methods among co-located mobile devices are inefficient and time-consuming (Rukzio, Broll, Leichtenstern, & Schmidt, 2007; Stephanidis & Antona, 2013). Although these methods allow users to share information in a co-located environment, they do have limitations. One of these limitations is the actual information sharing process itself. The following steps are involved in the typical process:

1. The selection of the file(s) to be shared;
2. Enabling the respective mobile sharing technology;
3. Scanning of devices in the immediate area;

4. Authenticating the relationship between two devices through a form of pairing; and
5. Sending the file(s) to the recipient device, whereby the recipient is required to accept the sharing request before sharing of file(s) even occurs.

This process is repeated for each file to be shared as well as every new device with which the user would like to share information. Currently, the information sharing process is time-consuming, inefficient, and tedious.

1.4 Aim of Research

The aim of this research is:

> ***To design a proxemic Natural User Interface to provide an accurate and usable solution to support information sharing among co-located mobile devices.***

1.5 Scope and Constraints

The focus of this research is to determine how a proxemic NUI can be designed to support information sharing among co-located mobile devices. Various NUI interaction techniques exist; however, not all of them are applicable to the information sharing process. The interaction techniques need to be selected based on their accuracy and usability in supporting co-located information sharing. User acceptance and technical feasibility of these interactions also need to be considered. Due to time constraints, the field of trust and security issues for mobile information sharing is excluded from the scope of the research. Despite this, the design artefact considered and implemented a basic form of encryption for the privacy and security of sharing information.

1.6 Ethical Considerations

Since participants were required to evaluate the design of the proxemic NUI, ethical clearance was required for their involvement in this research. Consent forms were given to participants for both the focus group and usability evaluation. An explanation of this research was given and it was emphasised that participants had the right to withdraw from the focus group or evaluation at any point. The Nelson Mandela Metropolitan University (NMMU) Human Research Ethics Committee (REC-H) approved the ethical clearance. The ethical clearance number for this research is H15-SCI-CSS-004 (Appendix A).

1.7 Research Design

The research design illustrates the direct relationship between the research questions and research objectives. The research methodology together with the relevant research strategies are discussed in this section.

1.7.1 Research Questions

This research was guided through addressing the primary research question. The primary research question was:

> ***How can a proxemic Natural User Interface be designed to provide an accurate and usable solution to support information sharing among co-located mobile devices?***

The following secondary questions were formulated to answer the primary research question:

- RQ_1. What are the shortcomings of existing information sharing methods currently used by mobile devices?
- RQ_2. What are the benefits and shortcomings of existing NUI interaction techniques for information sharing?
- RQ_3. How should the relative pose for co-located mobile devices be calculated?
- RQ_4. How should NUI interaction techniques be designed to support information sharing among co-located mobile devices?

RQ_5. How can a proxemic prototype NUI be developed to support information sharing among co-located mobile devices?

RQ_6. How accurate and usable is the proxemic prototype NUI in supporting information sharing among co-located mobile devices?

The research presented in this dissertation was conducted to answer these research questions. Each research question has a corresponding research objective.

1.7.2 Research Objectives

The primary research objective of this research was:

> ***To design a proxemic Natural User Interface to provide an accurate and usable solution to support information sharing among co-located mobile devices.***

The following secondary objectives were identified to achieve the main research objective:

RO_1. To identify the shortcomings of existing information sharing methods currently used by mobile devices (Chapter 2).

RO_2. To identify the benefits and shortcomings of existing NUI interaction techniques for information sharing (Chapter 3).

RO_3. To determine how the relative pose can be calculated for co-located mobile devices (Chapter 4).

RO_4. To determine how NUI interactions techniques can be designed to support information sharing among co-located mobile devices (Chapter 5).

RO_5. To develop a proxemic prototype NUI to support information sharing among co-located mobile devices (Chapter 5).

RO_6. To evaluate the accuracy and usability of the proxemic prototype NUI in supporting information sharing among co-located mobile devices (Chapter 6).

The research presented in this dissertation was conducted to achieve these research objectives. Each research objective was formulated to answer a research question. Both the research questions and research objectives were addressed with the selected research methodology to be discussed in the next section.

1.7.3 Research Methodology

A research methodology is a collection of principles, practices, and procedures applied to an existing problem in a particular field of research (Peffers, Tuunanen, Rothenberger, & Chatterjee, 2008). In the process of solving this problem, the selected methodology is critical as it maps out the entire research work and provides credibility to the research. The selected research methodology for this research was the Design Science Research (DSR) methodology, which is discussed in the following section (Section 1.7.3.1). The following research strategies were used within the DSR methodology activities and cycles:

- Literature Study;
- Focus Groups;
- Prototyping; and
- Experiments.

These research strategies were used to address the identified research questions and research objectives. The next section covers the DSR methodology.

1.7.3.1 Design Science Research Methodology

The DSR methodology provides specific guidelines for evaluation and iteration of research projects. Many research disciplines use the DSR methodology; however, it originates from the research field of information systems as a constructive research paradigm, primarily focused on developing new and innovative artefacts (Rodríguez, Kuvaja, & Oivo, 2014). In the rigorous process of designing and using these artefacts, new scientific knowledge is created. Thus, the DSR methodology is heavily dependent on the complementary research cycles between design and behavioural science, whereby the development of new artefacts relies on existing theories and contributes new knowledge, which is artefact related (Hevner, March, Park, & Ram, 2004).

Hevner et al. (2004) defines the *DSR methodology* as a scientific study to develop a solution to a problem, whereby *"knowledge and understanding of a problem domain and its solution are achieved in the building and application of a design artefact"*. The DSR methodology is comprised of three main components, namely environment, DSR, and knowledge base (Figure 1.1). These components are guided by the three DSR methodology cycles which are: Relevance, Design, and Rigor (Hevner, 2007).

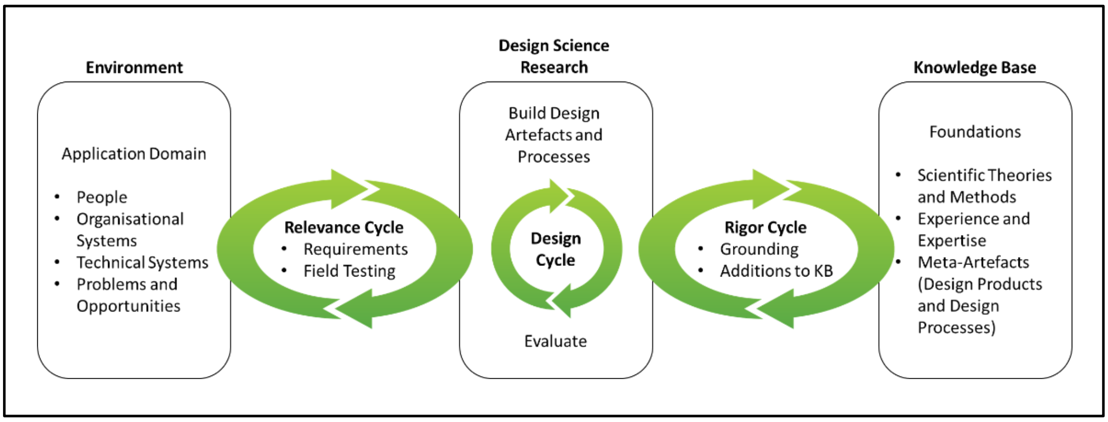


Figure 1.1: The DSR Methodology Cycles (Hevner, 2007)

The ***environment*** provides the research relevance through the specific needs of the application domain. An application domain comprises people, organisational systems, and technical systems, which interact and work towards a common objective. The DSR methodology typically commences by identifying and representing the problems and opportunities in an actual application environment. Thus, the Relevance Cycle initiates the DSR methodology with an application context, which not only provides the research requirements (problem identification) as inputs, but also defines the acceptance criteria against which the research results can be evaluated.

The ***knowledge base*** essentially provides rigor to the research. The rigor is based on the application of the knowledge base, which includes scientific theories and engineering methods, experiences and expertise in the application, and existing artefacts (meta-artefacts) and processes discovered in the application domain. This knowledge base provides the theoretical foundation for a rigorous DSR methodology to be conducted in the Rigor Cycle. The design artefact should expand existing research foundations. Moreover, rigorous research procedures must be used in the development and evaluation of the design artefact. Thus, the Rigor Cycle is able to provide existing knowledge to the research problem, which ensures the artefact is new and innovative. According to Hevner (2007), the research rigor in the DSR methodology is predicated on *"the appropriate selection and application of the theories and methods to construct and evaluate the design artefact"*.

The ***DSR*** component is where the artefact is built during the research process, as shown in Figure 1.1. The Design Cycle is the core cycle of any DSR methodology project. This cycle illustrates the iterative nature of the process, whereby the artefacts are built and evaluated in cycles, which rely on existing knowledge or theories. Design artefacts are validated through empirical procedures *"where specific objectives are set, appropriate data is gathered and analysed, and conclusions are drawn"* (Hevner et al., 2004).

Multiple iterations of the Design Cycle in the DSR methodology are performed until a satisfactory design is achieved, and only then are contributions outputted into the Rigor Cycle and the Relevance Cycle. Therefore, multiple iterations of the Relevance Cycle and Rigor Cycle are also performed. Another example of multiple iterations is when additional requirements are discovered during the Design Cycle that cause another iteration of the Relevance Cycle and investigated using the existing knowledge base from within the Rigor Cycle. This knowledge provided by the Rigor Cycle for the research problem is then used within the Design Cycle.

The DSR methodology can be used as a framework for conducting research based on Design Science (Figure 1.2), which involves the performance of the following activities:

1. ***Identify Problem and Motivate:*** Define specific research problem and justification of a solution;

2. ***Define Objectives of a Solution:*** Inferring the solution objectives derived from the problem definition and knowledge;
3. ***Design and Development:*** Involves creating the artefact solution;
4. ***Demonstration:*** Demonstrating the artefact's efficacy to solve the defined problem;
5. ***Evaluation:*** Observe and measure if and/or how well the artefact supports a solution to the defined problem, by comparing the solution objectives to actual observed results from the artefact in the demonstration phase; and
6. ***Communication:*** Communicates the importance of the problem, the artefact, its utility and novelty, rigor of its design, and its effectiveness to relevant audiences.

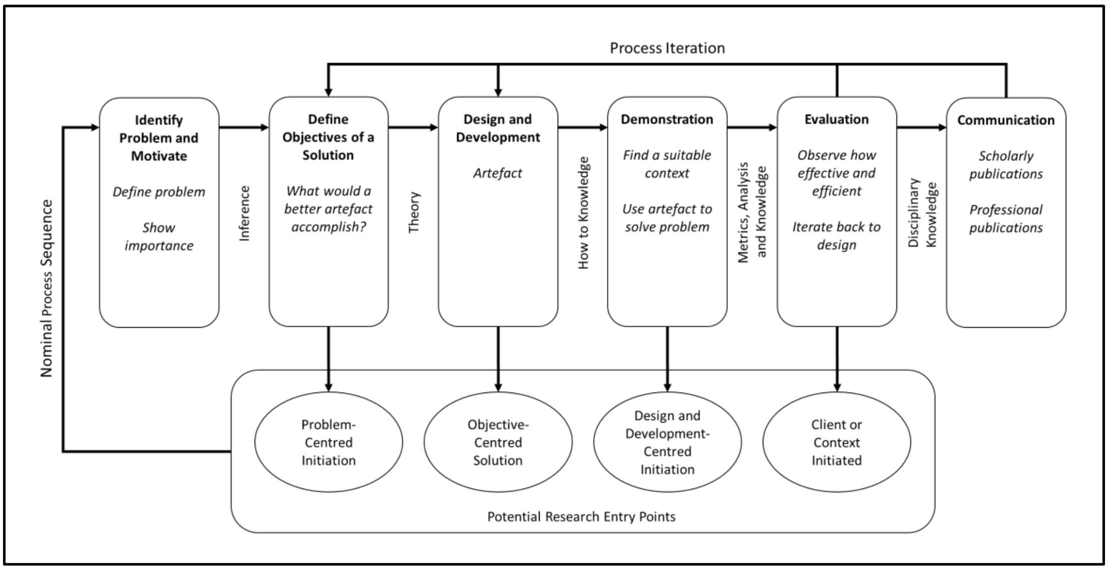


Figure 1.2: The DSR Methodology Process Model Adapted from Peffers et al. (2008)

Peffers et al. (2008) mentions that a distinguishing feature of the DSR methodology process model is identifying that the research can be initiated from various contexts, namely Problem-Centred Initiation, Objective-Centred Solution, Design and Development-Centred Initiation, and Client or Context Initiated. This model can start in the corresponding activity of the nominal process sequence shown.

Johannesson and Perjons (2014) discusses the activities to be performed in the DSR methodology (Figure 1.3). Firstly, the problem is defined and then the artefact is identified and outlined, which is used to solve the explicated problem and define important requirements. The artefact must then be created. Once the artefact is designed and developed, the artefact needs to be demonstrated to show its usability. Evaluation of the artefact follows demonstration in order to determine the effectiveness with which the artefact solved the explicated problem. This is similar to the process model of the DSR methodology (Figure 1.2) proposed by Peffers et al. (2008).

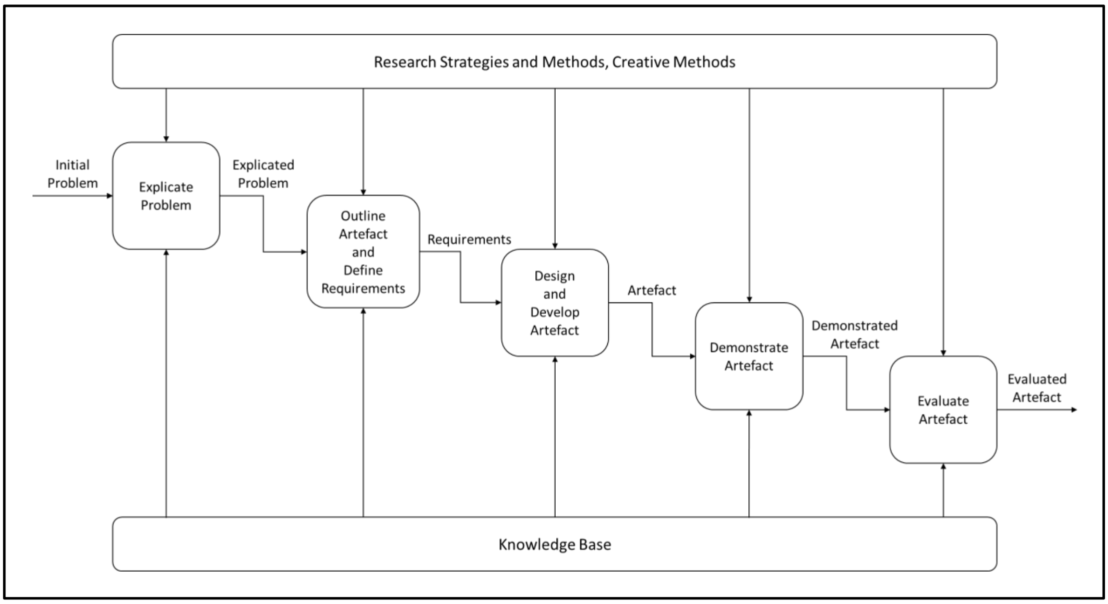


Figure 1.3: An Overview of the DSR Methodology Activities (Johannesson & Perjons, 2014)

Hevner et al. (2004) broadly defines the DSR methodology outputs as:

- ***Constructs*** (vocabulary and symbols) support the definition and communication of problems and solutions;
- ***Models*** (abstractions and representations) use constructs to represent the problem design and its solution space by matching the problem and solution elements;
- ***Methods*** (algorithms and practices) define the solution to the problem by explicitly defining the process through algorithms to textual descriptions of *"best practice"* approaches; and
- ***Instantiations*** (implemented and prototype systems) demonstrate feasibility of how constructs, models, or methods can be implemented and the artefact's relevance to solving the defined problem.

Vaishnavi and Kuechler (2015) summarises the DSR methodology outputs by Hevner et al. (2004), which are potentially obtained from research using the DSR methodology. However, Vaishnavi and Kuechler (2015) identify an additional output called Better Theories (Table 1.1).

Table 1.1: DSR Methodology Outputs (Vaishnavi & Kuechler, 2015)

Outputs	Description
Constructs	The conceptual vocabulary of a domain
Models	A set of propositions or statements expressing relationships between constructs
Methods	A set of steps used to perform a task, such as the how-to knowledge
Instantiations	The operationalisation of constructs, models, and methods
Better Theories	Artefact construction as analogous to experimental natural science, coupled with reflection and abstraction

DSR literature (Hevner & Chatterjee, 2010; Hevner et al., 2004; Indulska & Recker, 2008; Offermann, Blom, Schönherr, & Bub, 2010) identifies seven guidelines to be followed when conducting research using the DSR methodology. The fundamental principle of the DSR methodology, which is derived from these guidelines, is the importance of knowledge and understanding of the research problem to develop an artefact. Therefore, the DSR methodology is the rigorous process of designing an innovative and meaningful artefact (guideline one) to solve a defined problem (guideline two). A meaningful artefact in essence must display utility for the defined problem. Hence, the artefact must be subjected to a thorough evaluation, which is crucial to the DSR methodology (guideline three). Research contributions only occur when the defined problem is solved in a more effective and efficient manner than existing solutions (guideline four). The artefact is defined through the application of rigorous methods in both the construction and evaluation of the artefact (guideline five). The overall DSR methodology incorporates a search process as a problem is identified and investigated until an effective solution is presented (guideline six). Lastly, the effective communication of results to appropriate audiences must occur (guideline seven). Table 1.2 summarises these guidelines.

Table 1.2: Guidelines for the DSR Methodology (Hevner & Chatterjee, 2010)

Guideline	Description
Guideline 1: Design as an Artefact	DSR must produce a viable artefact in the form of a construct, a model, or an instantiation
Guideline 2: Problem Relevance	The objective of DSR is to develop technology-based solutions to important and relevant business problems
Guideline 3: Design Evaluation	The utility, quality, and efficacy of a design artefact must be rigorously demonstrated via well-executed evaluation methods
Guideline 4: Research Contributions	Effective DSR must provide clear and verifiable contributions in the areas of the design artefact, design foundations, and/or design methodologies
Guideline 5: Research Rigor	DSR relies upon the application of rigorous methods in both the construction and evaluation of the design artefact
Guideline 6: Design as a Search Process	The search for an effective artefact requires utilising available means to reach desired ends while satisfying laws in the problem environment
Guideline 7: Communication of Research	DSR must be presented effectively both to technology-oriented as well as management-oriented audiences

The DSR methodology guidelines address each of the key activities (Figure 1.3) and research strategies in the DSR methodology cycles (Figure 1.1). These guidelines facilitate good DSR practice and ensure each activity of the DSR methodology produces usable deliverables (Peffers, Rothenberger, Tuunanen, & Vaezi, 2012).

1.7.3.2 Literature Study

Literature studies are used to provide a comprehensive, critical, and contextualised synthesis of a topic in a specific research area (Hofstee, 2006). The literature study strategy is used to conduct the first and the second activities in the DSR methodology, namely ***Explicate Problem*** and ***Outline Artefact and Define Requirements***. For this research, a literature study was used to identify and discuss the shortcomings of existing information sharing methods used by mobile devices. This study was also used to introduce the definition and description of information sharing. An investigation was conducted into the existing process of information sharing with specific focus given to the field of mobile computing and the requirements to share information. Existing information sharing systems were reviewed and their shortcomings discussed.

A literature study was also used to determine if NUIs can support co-located information sharing and identify the most suitable NUI interaction techniques. Existing NUI systems were reviewed to show how various NUI interaction techniques are implemented and used by users.

These studies were used as the theoretical foundation to identify the requirements of the research by using the knowledge base of the problem domain (information sharing) and the suggested solution (NUIs and their interaction techniques).

1.7.3.3 Focus Groups

Literature has various definitions of focus groups that describes them as interactive and social events, collective activities, or organised discussions (Cohen & Crabtree, 2006; Krueger & Casey, 2014; Stewart & Shamdasani, 2014). Berkowitz (2015) define a *focus group* as:

> *"A focus group is a group of individuals selected and assembled by researchers to discuss their perceptions, opinions, beliefs, and attitudes towards a specific topic."*

Focus group participants are encouraged to not only express their own opinions, but also respond to opinions of other participants and questions presented by the moderator. Therefore, these groups prove to be an effective data collection strategy to gather in-depth qualitative data regarding participants' thoughts and opinions on a specific topic.

Focus groups were used to conduct the third and fourth activities in the DSR methodology, namely ***Design and Develop Artefact*** and ***Demonstrate Artefact***. In this research, focus groups were conducted to identify and determine the feasibility and user acceptance of NUI interaction techniques to be used to complete specific tasks with the design artefact.

1.7.3.4 Prototyping

Prototyping involves the creation of software prototypes, which are tested and reviewed by testers (Lim, Stolterman, & Tenenberg, 2008). A prototype is used to express research in a useful way (Olivier, 2009). Using prototypes allows for the testing of prototypes early on and enables the implementation of modifications before the final prototype is completed. Prototyping requires user involvement and permits the users to see and interact with a prototype allowing them to provide better and more extensive feedback and specifications. Gube (2013) identify and discuss the importance of prototyping in the design process of the artefact (Table 1.3). Prototyping was used to address the third and fourth activities in the DSR methodology, namely ***Design and Develop Artefact*** and ***Demonstrate Artefact***. This method was also used to determine whether a proxemic prototype NUI can be used to support co-located information sharing among mobile devices.

Table 1.3: Prototyping Importance (Gube, 2013)

Importance	Description
Find Design Issues Early	Conceptualised artefacts may appear to be an awesome concept, however is later discovered to be terrible when visually displayed on paper or screen
Iterate More Quickly on a Design Concept	Prototype creation allows for fast improvements to a design concept, allowing for iterative revision and refinement to the artefact
Gather Design Feedback Better	Individuals are able to quickly and conveniently understand the concept of the artefact through the prototype and potentially provide alternative solutions
Prototypes can be a Presentational Tool	Prototyping is an effective method in visually representing the concept of the artefact
Be Able to Perform User Testing Early On	Prototyping allows for user testing to be perform at the start of the design process, instead of at the end
Prototypes Encourage Collaboration	Involvement of individuals in the early stages of designing the artefact is achieved through prototypes
Prototypes Give You a Visual Guide to the Finished Product	It is much easier to produce an artefact when a solid concept of the end result is shown

1.7.3.5 Experiments

An experiment is conducted to assess a theory or to examine the outcome of a given intervention (Hofstee, 2006). Initial experiments of the Bluetooth Received Signal Strength Indicator (RSSI) were conducted to determine whether it can be used as an accurate indicator of distance between co-located mobile devices. The quantitative data captured from these experiments was analysed. This research assessed how accurate and usable the proxemic prototype NUI was in supporting information sharing among co-located mobile devices. The experimental strategy addresses the fourth and fifth activities in the DSR methodology, namely ***Demonstrate Artefact*** and ***Evaluate Artefact***.

A pilot study is used to determine if the design artefact and procedure is ready for the usability evaluation to be conducted (Van Teijlingen & Hundley, 2001). The pilot study is also used to ensure the statistical and analytical procedures for evaluating the data are correct for a research study.

Another experiment was the usability evaluation of the design artefact, which involved evaluating the proxemic prototype NUI in supporting information sharing among co-located mobile devices. Performance metrics and post-test questionnaires were used to determine the efficiency, effectiveness, and satisfaction of the artefact. Quantitative and qualitative data were recorded and analysed. The results of the evaluation were presented and conclusions drawn from this research. Two NUI interaction techniques were selected and compared against each other to determine which technique was the most preferred by participants.

1.7.3.6 Application of the Research Methodology

This research was focused on determining whether a proxemic NUI can be designed to provide an accurate and usable solution to support information sharing among co-located mobile devices (Section 1.7.1). Therefore, the creation of a design artefact was required to demonstrate the feasibility of NUI interaction techniques in solving the defined problem.

The DSR methodology was discussed and identified to be the most appropriate methodology for the purposes of this research. This research applied the three components of the DSR methodology, namely environment, DSR, and knowledge base, along with the corresponding Relevance, Design, and Rigor Cycles (Hevner, 2007). Peffers et al. (2008) identified the DSR methodology activities to be performed in each component and cycle. An adapted version of the DSR methodology cycles and activities was used for this research (Figure 1.4).

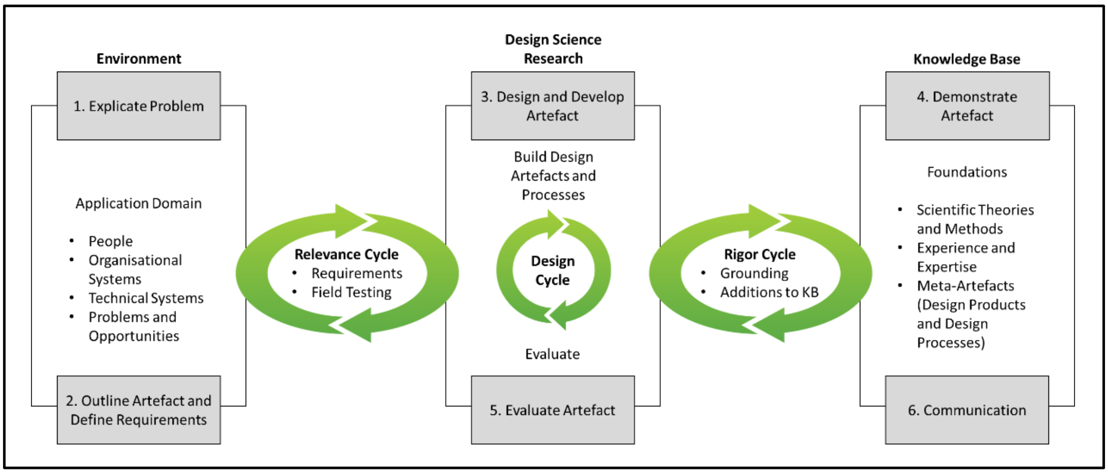


Figure 1.4: Combined DSR Cycles with the DSR Activities Adapted from Hevner (2007), Johannesson and Perjons (2014), and Peffers et al. (2008)

1.7.3.7 Summary

The research questions (Section 1.7.1) were introduced, followed by the research objectives (Section 1.7.2). Several secondary questions were formulated to address the primary question. The secondary research objectives were identified to achieve the primary research objective. The DSR methodology was selected as an appropriate methodology for this research because it requires the development of a proxemic prototype NUI to support information sharing among co-located mobile devices. The development of this prototype directly correlates with the development of an artefact to solve the identified problem, which is one of the core activities in the DSR methodology, namely ***Design and Develop Artefact***.

The following research strategies were identified:

- Literature study (Section 1.7.3.2);

- Focus groups (Section 1.7.3.3);
- Prototyping (Section 1.7.3.4); and
- Experiments (Section 1.7.3.5).

These research strategies were used throughout the different DSR methodology activities and cycles. The application of the research methodology (Section 1.7.3.6) outlined an adapted version of the DSR methodology with the corresponding activities to be performed in each component and cycle (Figure 1.4).

Table 1.4 summarises the research questions and research objectives that are outlined in the dissertation chapters, and these were addressed using the corresponding research strategies.

Table 1.4: Summary of Research Questions, Objectives, Methods, and Chapters

No.	Research Question	Research Objective	Research Strategy	Dissertation Chapter
1	What are the shortcomings of existing information sharing methods currently used by mobile devices?	To identify the shortcomings of existing information sharing methods currently used by mobile devices.	Literature Study	Chapter 2
2	What are the benefits and shortcomings of existing NUI interaction techniques for information sharing?	To identify the benefits and shortcomings of existing NUI interaction techniques for information sharing.	Literature Study	Chapter 3
3	How should the relative pose for co-located mobile devices be calculated?	To determine how the relative pose can be calculated for co-located mobile devices.	Literature Study and Experiments	Chapter 4
4	How should NUI interaction techniques be designed to support information sharing among co-located mobile devices?	To determine how NUI interaction techniques can be designed to support information sharing among co-located mobile devices.	Prototyping	Chapter 5
5	How can a proxemic prototype NUI be developed to support information sharing among co-located mobile devices?	To develop a proxemic prototype NUI to support information sharing among co-located mobile devices.	Focus Groups and Prototyping	Chapter 5
6	How accurate and usable is the proxemic prototype NUI in supporting information sharing among co-located mobile devices?	To evaluate the accuracy and usability of the proxemic prototype NUI in supporting information sharing among co-located mobile devices.	Experiments	Chapter 6

1.8 Dissertation Outline

The dissertation is structured according to the DSR methodology identified and discussed in the previous section. A narrative descriptive form of the dissertation structure is outlined by presenting the contents of the seven chapters for this research. These chapters are combined with the adapted DSR methodology from Hevner (2007), Johannesson and Perjons (2014), and Peffers et al. (2008), as shown in Figure 1.5.

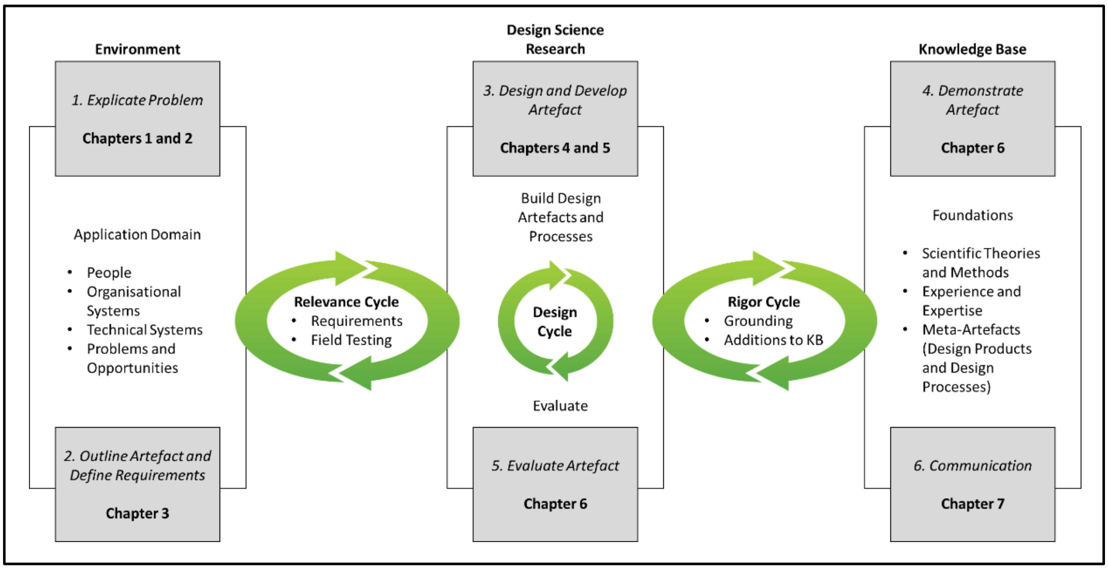


Figure 1.5: Dissertation Chapters Combined with the Adapted DSR Methodology

Figure 1.6 presents the flow of the seven dissertation chapters with their corresponding DSR methodology activities and guidelines. In this figure, the research questions, research objectives, and research strategies are mapped to each chapter. The dissertation is structured as follows:

Chapter 2: Information Sharing

Chapter 2 is based on the first activity in the DSR methodology, namely ***Explicate Problem***. This chapter provides an in-depth literature study for the first knowledge base (information sharing) by introducing and discussing information sharing. The concept of information sharing is defined and described. Existing information sharing methods used by mobile devices are identified and their shortcomings discussed. An investigation is conducted to determine the process used by mobile devices to share information, as well as the requirements to share this information. Existing information sharing systems are identified and their shortcomings discussed.

Chapter 3: Natural User Interfaces

Chapter 3 provides an in-depth literature study for the second knowledge base (NUIs). The second activity in the DSR methodology, ***Outline Artefact and Define Requirements***, is performed through investigating NUIs. The requirements of the design artefact are derived from analysing the problem domain and proposing a suitable solution. The NUI definition is explored and the various existing NUI interaction techniques are discussed. Existing NUI systems are identified and reviewed to demonstrate the usability of these techniques and determine their suitability in supporting co-located information sharing among mobile devices.

Chapter 4: Positioning and Communication Design

Chapter 4 details the design and implementation of the various prototypes to be integrated into the design artefact named MotionShare. This chapter addresses ***Design and Develop Artefact*** which is the third activity in the DSR methodology. Existing positioning techniques are identified and their shortcomings highlighted. Experiments are conducted to identify a potential solution to the defined problem. The hardware and software tools used in the design and implementation of the MotionShare are discussed.

Chapter 5: Design and Implementation of Gestures

Chapter 5 continues with the third activity in the DSR methodology, namely ***Design and Develop Artefact*** which is implemented in two phases. In the first phase, focus groups are conducted to identify which NUI interaction techniques are feasible and accepted by users in completing specific tasks of the information sharing process. The results of these groups are aggregated and used for the design and implementation of several prototypes. In the second phase, these prototypes are shown to the focus groups to evaluate user acceptance and preference. The results of the second phase are aggregated and compared to the existing knowledge base of NUIs addressed in Chapter 3. A decision was made to select two NUI interaction techniques based on this comparison. The two techniques are implemented into MotionShare.

Chapter 6: Evaluation

Chapter 6 discusses the evaluation of the design artefact, which addresses the fourth and fifth activities in the DSR methodology, namely ***Demonstrate Artefact*** and ***Evaluate Artefact***. MotionShare is subjected to two evaluation techniques, namely analytical and experimental. The analytical evaluation technique evaluates the precision, trueness, and recall for device positioning and MotionShare gestures. The experimental evaluation technique comprises of two evaluations, namely formative and summative. The formative evaluation consisted a pilot study. The summative evaluation is where MotionShare is subjected to a summative usability test, whereby suitable participants are selected and usability data is collected and analysed. The results are measured against metrics, namely efficiency, effectiveness, and satisfaction. The summative usability test also evaluates the artefact's efficacy to solve the defined problem.

Chapter 7: Conclusions

Chapter 7 completes the dissertation by presenting the conclusions drawn from this research using the sixth activity in the DSR methodology, which is defined by Peffers et al. (2008) as ***Communication***. The achievement of the research objectives is examined. Theoretical and practical research contributions are highlighted. Limitations and problems experienced are discussed together with ideas for future research.

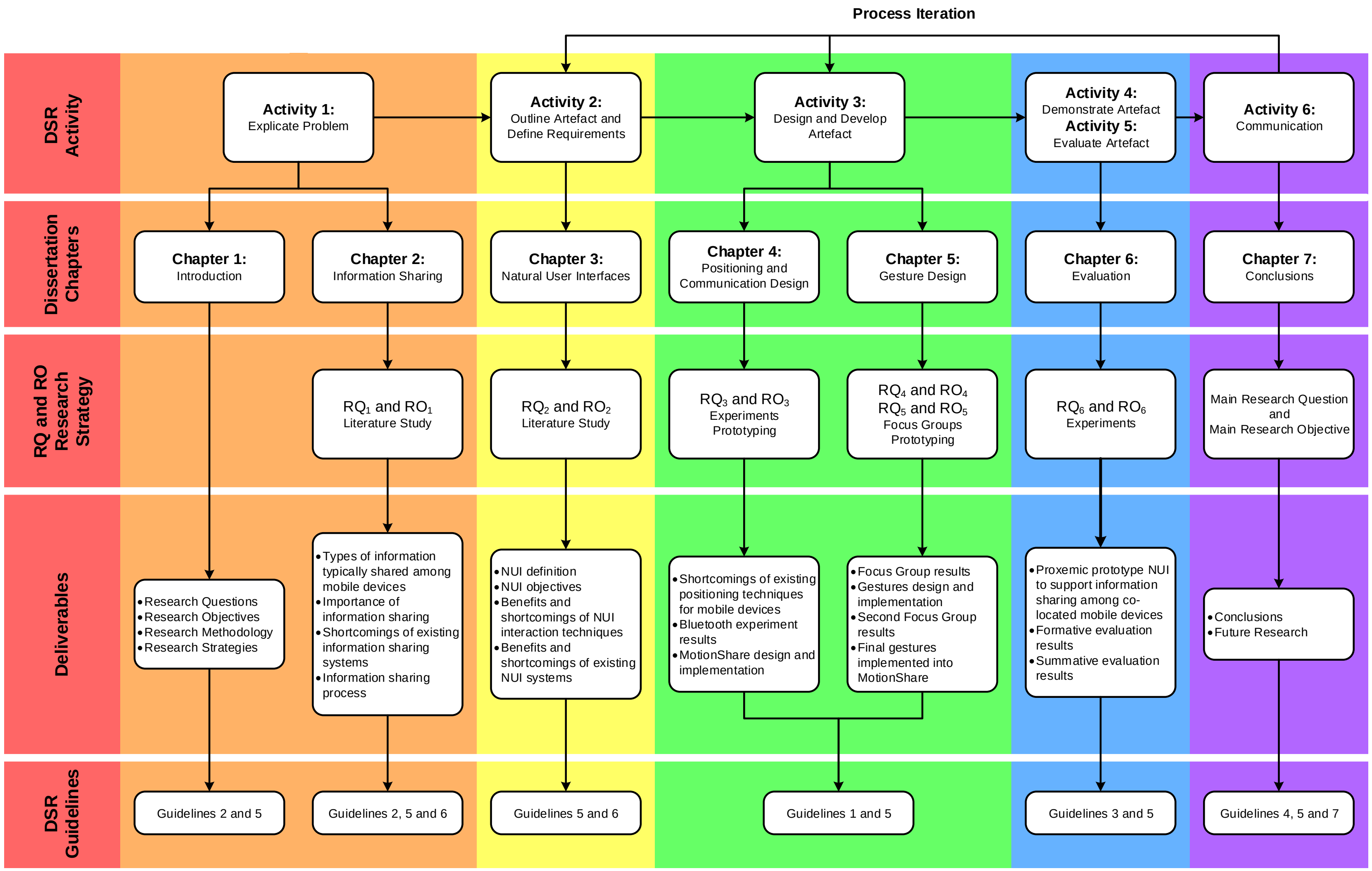


Figure 1.6: Dissertation Structure Combined with DSR Methodology

Chapter 2: Information Sharing

2.1 Introduction

The previous chapter provided an overview of the research by identifying the defined problem, scope and constraints, and ethical considerations. The research design employed throughout the research and dissertation structure was discussed.

The first activity in the DSR methodology, namely ***Explicate Problem***, is continued from the previous chapter and performed within the Relevance Cycle (Figure 2.1), to address the first research question (Section 1.7.1) identified:

> *"RQ$_1$. What are the shortcomings of existing information sharing methods currently being used by mobile devices?"*

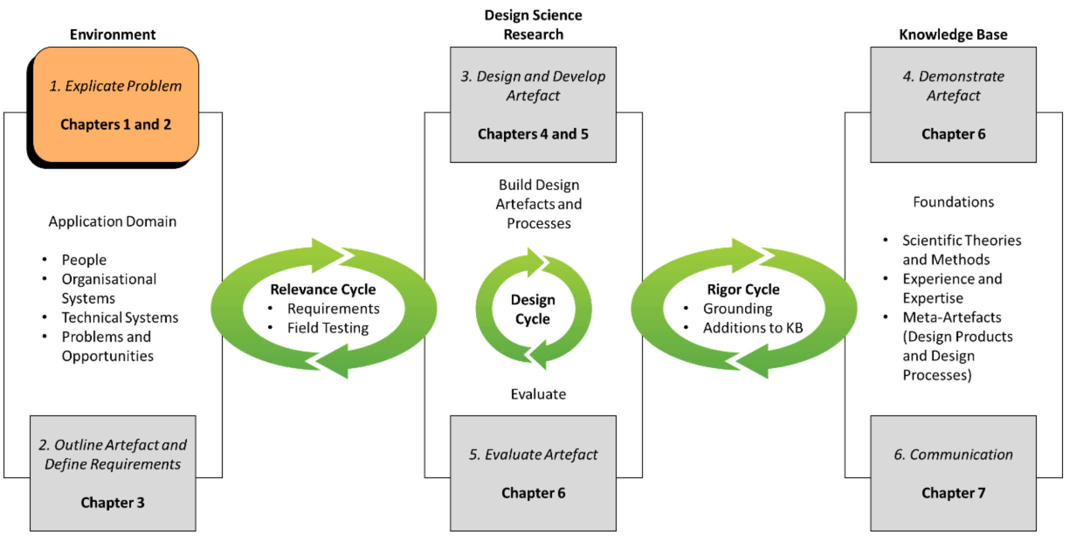


Figure 2.1: Chapter 2 Position in the Adapted DSR Methodology

The above question was answered by conducting an in-depth analysis of existing information sharing methods in the form of a literature study. The literature study investigates the existing methods available to mobile device users when they want to share co-located information.

Firstly, the term *information sharing* is defined and the various types of information typically shared among mobile devices are identified. The importance of information sharing is discussed. Existing information sharing methods are reviewed, which provide a greater understanding and insight into the problem domain of information sharing. An analysis of existing systems is conducted to identify the types of systems developed, which can be used as a theoretical reference basis for the design and implementation of MotionShare. Figure 2.2 outlines the structure of this chapter.

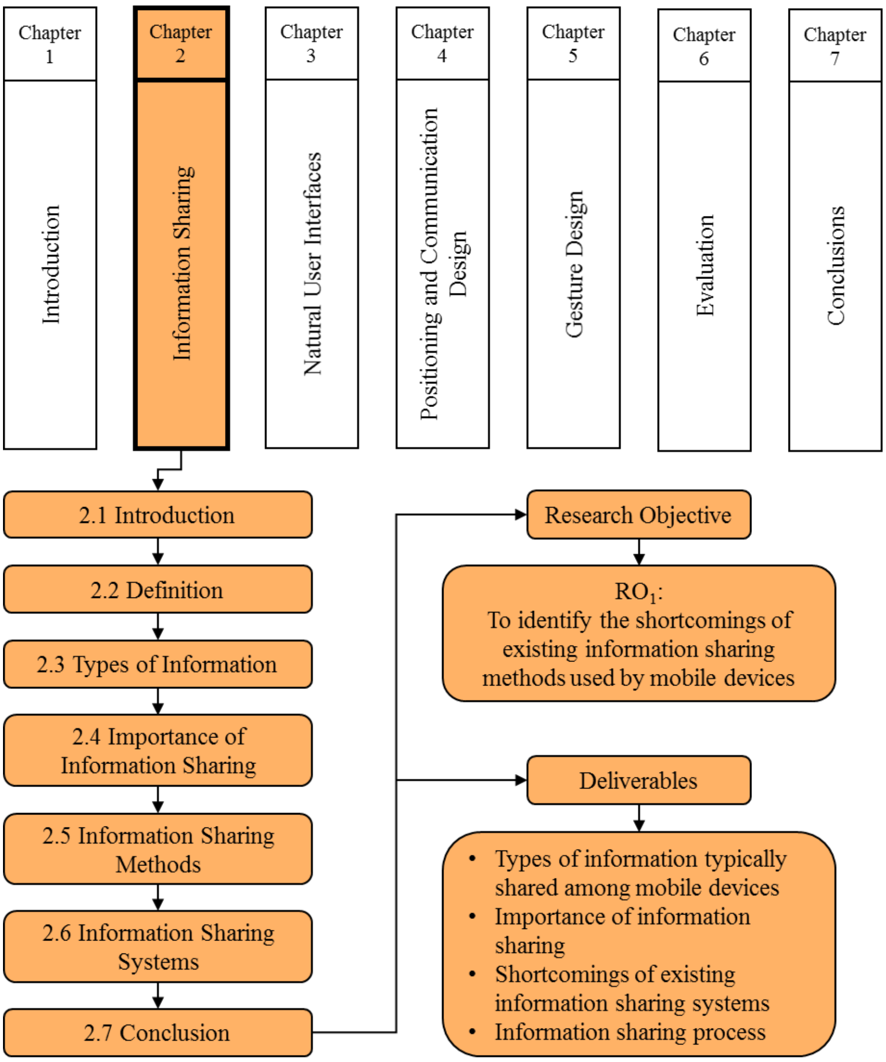


Figure 2.2: Chapter 2 Structure

2.2 Definition

Information sharing is defined as a central activity whereby any information, in its natural, electronic, or other form, is transferred between organisations or individuals by any means of communication or transference (Mesmer-Magnus & DeChurch, 2009). The following are examples of how information is commonly shared:

- Information shared by individuals that occurs in all social media websites (Facebook, Twitter, and LinkedIn) and video sharing websites (YouTube, NetFlix, and Twitch);
- Information shared by organisations such as a RSS feed of the Apple or Samsung website for people to stay informed of their latest news release on their products; and
- Information shared between technologies (firmware or software) such as identifying the IP addresses of available networks or location of mobile devices in a surrounding area.

The different ways of how information is shared is also dependent on the type of information. Thus, the types of information commonly shared on mobile devices are discussed in the next section.

2.3 Types of Information

The technological progression of mobile devices, such as smartphones, tablets, laptops, and other handheld devices, has enabled these devices to store and share information (Zhang, Van Den Berg, Madhani, Dutta, & Mohanti, 2007). This information can include user location, audio, video, and orientation, which is made possible through the increasing data storage capabilities and the number of embedded sensors in mobile devices. Table 2.1 is an extract of a report with the total number of smartphones that store different types of information. The percentage values represent the total number of smartphone users which participated in this report. These values are not exclusive because smartphones can store more than one information type shown in this report. Although 50% of smartphones are used to primarily store contact numbers of people, the types of information most commonly shared among mobile device users are documents, videos, music, and images (Kaspersky Lab, 2012).

Table 2.1: Information Stored on Smartphones Adapted from Kaspersky Lab (2012)

Information Stored on Device	Total number of Smartphones
Personal email messages	33.00%
Photos, videos, or music created by you	33.00%
Files for personal use (documents, spreadsheets)	16.00%
Photos, videos, or music created by other people	26.00%
Address book or phone contacts	50.00%
Files for work use (documents, spreadsheets)	13.00%
Work email messages	20.00%
Passwords or account details	17.00%
Coursework, study materials	10.00%
Banking details	12.00%
E-books	16.00%
None of these	17.00%

2.4 Importance of Information Sharing

Information sharing remains one of the most basic activities of cooperation, coordination, collaboration, and communication between various entities, including individuals, organisations, and technologies (Amir, 2013; Saab, Orendovici, Van Gorp, Maitland, & Maldonado, 2008; Wang, 2013). Figure 2.3 illustrates the importance of information sharing (Gava, De Mesquita Spinola, Tonini, & Medina, 2012). For example, individuals cooperate by coordinating with each other, which supports information sharing. The coordinating of individuals enables collaboration, which also supports information sharing. Collaboration presupposes communication because without individuals communicating, collaboration between individuals cannot occur. Communication of individuals also supports information sharing. Communication provides elements for coordination to occur. Therefore, information sharing occurs in every facet of people's daily lives due to the constant communication between people.

Thus, mobile technologies are ubiquitous and play an increasingly integral role in various fields, such as medicine, computers, and education (Barkhuus & Polichar, 2011). In the context of education, academic lives of university students are influenced by mobile devices, such as smartphones, tablets, laptops, and e-book readers (Chen & Denoyelles, 2013). These devices allow users instant connectivity to the world through the Internet, elevating access to information and encouraging interactivity with other individuals. Furthermore, these students not only consume information, but also create and share it.

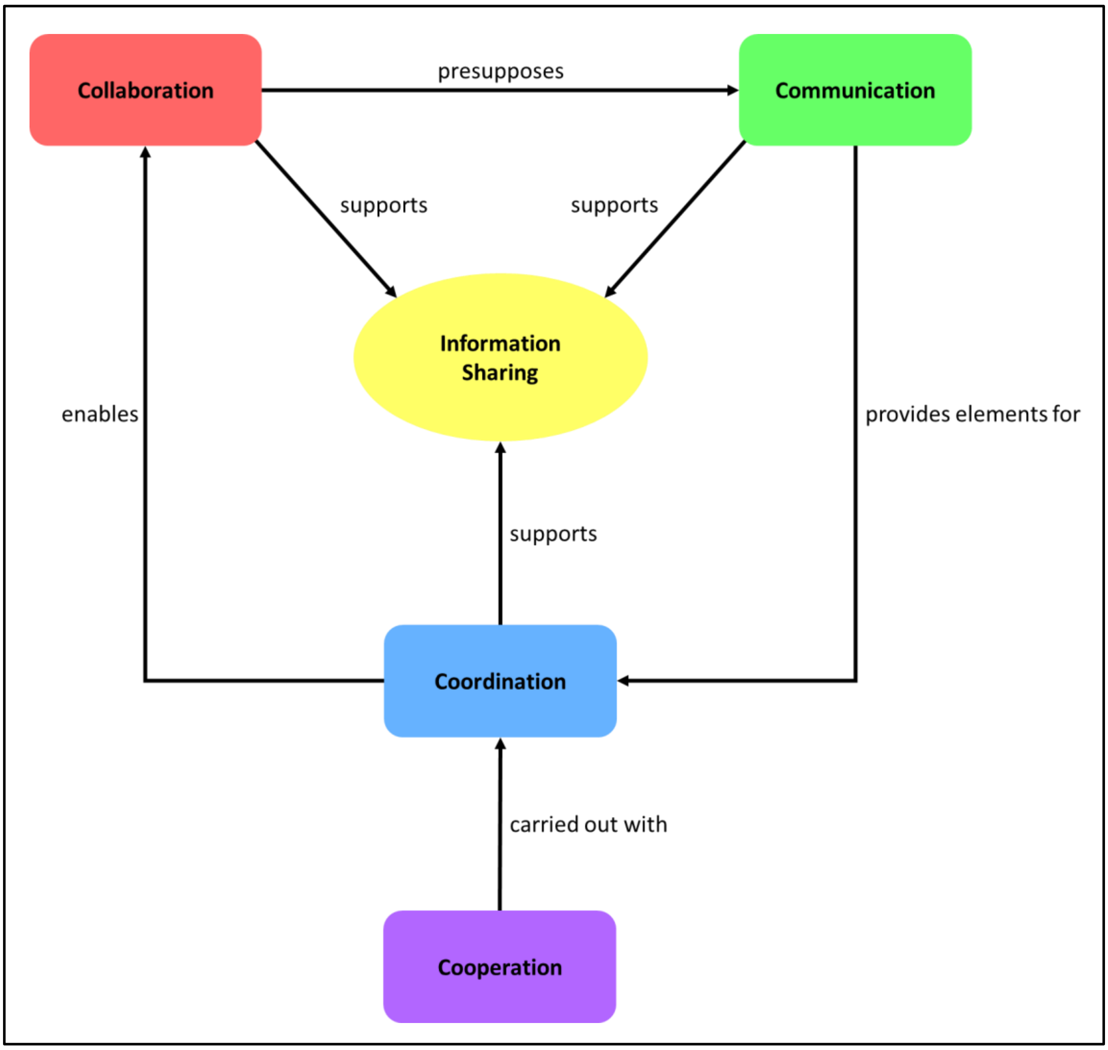


Figure 2.3: Information Sharing Importance Model adapted from Gava et al. (2012)

2.5 Information Sharing Methods

Numerous information sharing methods exist because people frequently communicate and coordinate with each other. These methods are either of a natural or digital nature. The following are examples of natural information sharing methods:

- A physical handover of photos to another individual such as a family member showing off their newly born baby photos;
- Exchanging of contact details between individuals such as business cards physically swopped between two people;
- During a social or business meeting, an agenda is given out to the attendees present; and
- Sending a postcard to an individual.

Natural information sharing methods still exist; however, the rapid progression of computing technology has resulted in these methods being digitised to take advantage of the existing knowledge of how they naturally work. The idea is to transform the digital counterparts to be easily usable and quickly accepted by individuals. The following are examples of how this transformation is achieved:

- Sending files such as images, videos, music, or documents can be done as email attachments or transferred between devices via, for example, flash drives, Bluetooth, or Dropbox;
- Contact details can be exchanged either verbally and the information being captured by the mobile device user or by the communicator capturing the information directly onto the device (input entry);
- Meeting agendas can be downloaded from a secure location onto the user's mobile device to be read; and
- Digital postcards are available to be sent to individuals and received as a link to a website.

Transformation of natural information sharing methods is predominantly present in almost all forms of technology. The next logical step in the evolution of information sharing in the digital domain is to utilise NUI interaction techniques to develop more natural and intuitive sharing methods for users.

Digital information sharing methods can be classified as either *distributed* or *co-located.* Typically, distributed information sharing occurs in virtual distributed teams, which are supported by the advancements in information technology (Gupta, Mattarelli, Seshasai, & Broschak, 2009). *Virtual distributed teams* refer to groups of individuals who are geographically and organisationally dispersed, and potentially function over different time zones (Lilian, 2014).

Co-located information sharing occurs in every facet of people's daily lives, from exchanging contact information with various people at a conference, to a group of people socialising at a restaurant. Co-located information sharing occurs when individuals are located in the same environment, such as a room (Singleton, 2014). As a result, information sharing occurs more frequently because individuals are face-to-face with one another. Kahai (2008) identify individuals in a co-located environment to possibly know each other or most likely having a shared context, such as working for the same organisation, studying at the same university, or living in the same city, all of which promote and facilitate communication in the form of social interaction.

2.6 Mobile Information Sharing Technologies

The following information sharing technologies are available on most mobile devices:

- ***Bluetooth*** can be used to wirelessly share information between mobile devices in the immediate area and mobile devices generally have integrated Bluetooth hardware (Bluetooth SIG, 2015). This solution may seem like a solution that would work across mobile platforms; however, while Android supports Bluetooth information sharing, Apple's iPhone does not.
- ***Wi-Fi*** supports information sharing among mobile devices when these devices can connect to a network resource such as the Internet via a Wi-Fi access point (AP). Wi-Fi coverage allows mobile devices to communicate

over the network or directly with each other using Wi-Fi Direct (Kiruthika, Smita, & Dhanashree, 2009).

- ***Cloud storage services*** such as Dropbox is an online service which stores any information for a user and is accessible from any mobile device (Dropbox Inc., 2015). Using the Dropbox application on a mobile device, a user can upload information to his account and then share a link to the information so that the receiver can download it directly from the sender's Dropbox account.
- ***Email*** remains one of the most reliable methods of sharing information (Beal, 2015). A user selects the email application on his mobile device, attach the file(s), and sends the email to the other individual's email address, whereby the receiver can download the email and attachment(s) on his mobile device.
- ***Near Field Communication*** (NFC) supports information sharing by holding two mobile devices closely together, where both contain embedded NFC chips (Gummeson, Priyantha, Ganesan, Thrasher, & Zhang, 2013). Information is shared via NFC using Android Beam which notifies the individual to wirelessly *"beam"* the selected information to the other device.

Existing information sharing systems use one or more of these methods to share information among mobile devices. The next section discusses existing information sharing systems.

2.7 Information Sharing Systems

Mobile applications developed to support information sharing are identified and discussed. These applications were identified based on their relevance to the problem of information sharing. An investigation into these applications provided insight and an understanding into the complexity of information sharing, as well as possible application functionality, which could be incorporated into the design artefact.

2.7.1 Feem

FeePerfect (2015) developed the Feem application to support chat and local file transfers using Wi-Fi, without Internet. This application supports multiple platforms, including iOS, Android, Windows Desktop, Mac OSX, Linux, and Windows Phone 8. As Feem uses Wi-Fi, it allows for an easy and seamless method of cross-platform information sharing. The local file transfers typically involve pictures, music, videos, or documents. The local file transference using Wi-Fi and the different file types are relevant to this research.

Feem displays the menu screen when the navigation drawer is selected, which displays an items list (Figure 2.4). The Wi-Fi network name to which the mobile device is connected is always displayed next to the navigation drawer. *Peers* displays the fragment containing other devices connected on the Wi-Fi using Feem, which are classified as your peers. The different file types located on the mobile device are categorised accordingly (as photos, music, videos, applications,

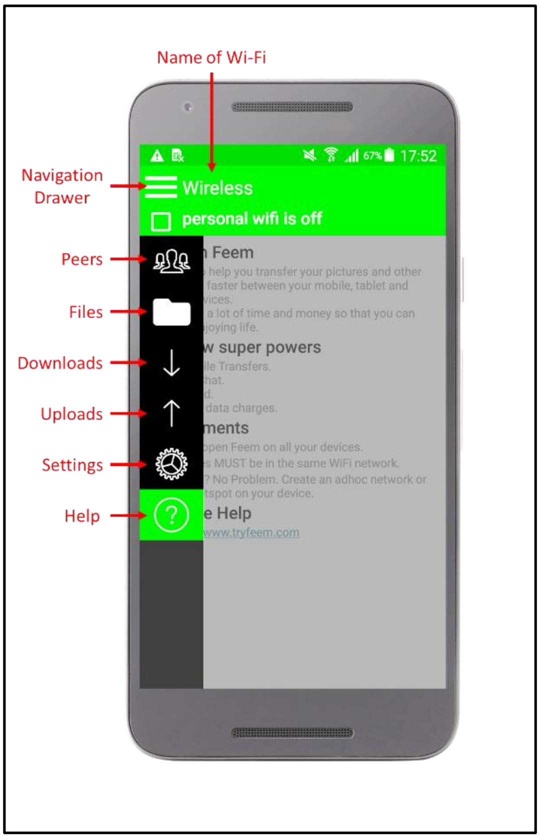


Figure 2.4: Feem Menu Screen

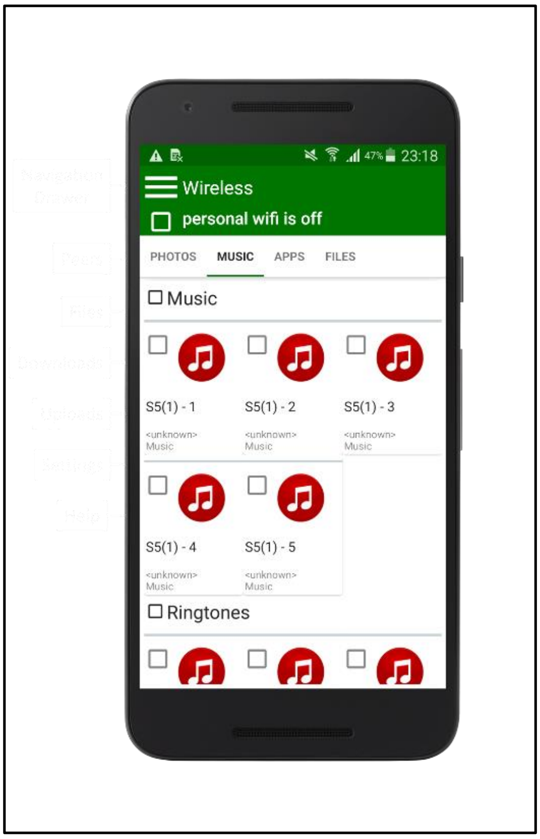


Figure 2.5: Feem File Structure

and files) and shown when *Files* is selected (Figure 2.5).

Downloads and *Uploads* displays the various files to be downloaded and uploaded respectively. The user also has the option to pause or cancel one or multiple downloads or uploads. *Settings* allows the Feem user to change several options, namely display name, avatar, checking the box for automatic downloads to occur without user confirmation, and the download location of the files. *Help* provides basic information on the application functionality and requirements. The notable shortcoming of Feem is the manual selection of files to be shared with the receiving device.

2.7.2 Share Link

ASUSTeK Computer Inc. (2015a) developed ZenUI which is a front-end touch interface for Android smartphones and tablets. ZenUI typically comes with preloaded applications, one of which is the Share Link application. ASUSTeK Computer Inc. (2015b) developed Share Link which supports various types of file transfers, such as sharing multimedia files and applications, quickly and seamlessly. This application allows the user to do all these functions by simply selecting *Send File* or *Receive File*, without Internet connectivity. The process to transmit files using Share Link is:

1. Select the *Send Files* button;
2. Select the file type such as *Music* (Figure 2.6) and then select the *Share Selected Files* button;
3. Select the *Send* button, which starts the scanning of devices; and
4. Once the receiver device is detected, select the receiver device icon to transmit the file (Figure 2.7).

Share Link provides a well-designed and intuitive interface according to the user reviews (ASUSTeK Computer Inc., 2015b). Nevertheless, the selection of files and scanning for a receiving device can become tedious when faced with the introduction of new receivers with whom to share. Another problem arose during the investigation of Share Link, where it became apparent that the device was unable to multi-task without interrupting the transfer process. The process to receive files using Share Link is also identified:

1. Select the *Receive File* button; and
2. Once the file is received, then the download progress bar is displayed.

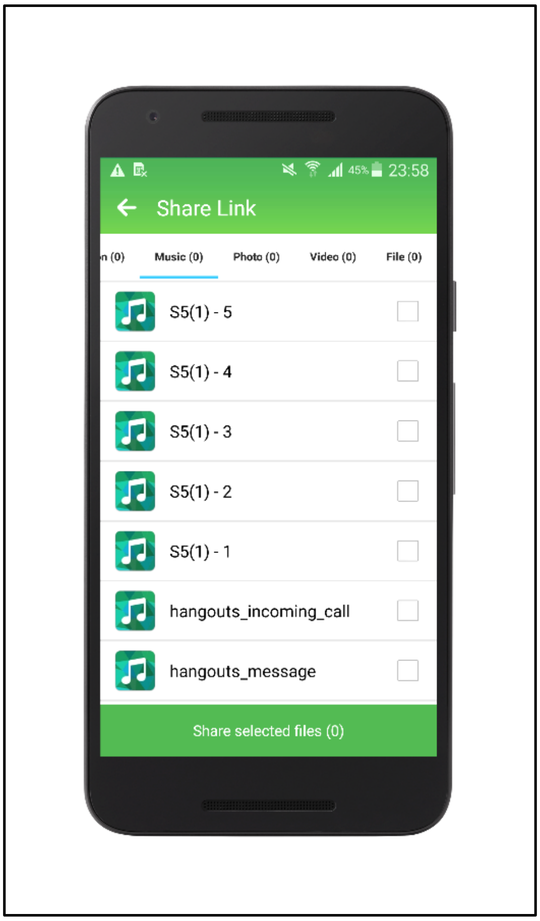


Figure 2.6: Share Link File Structure

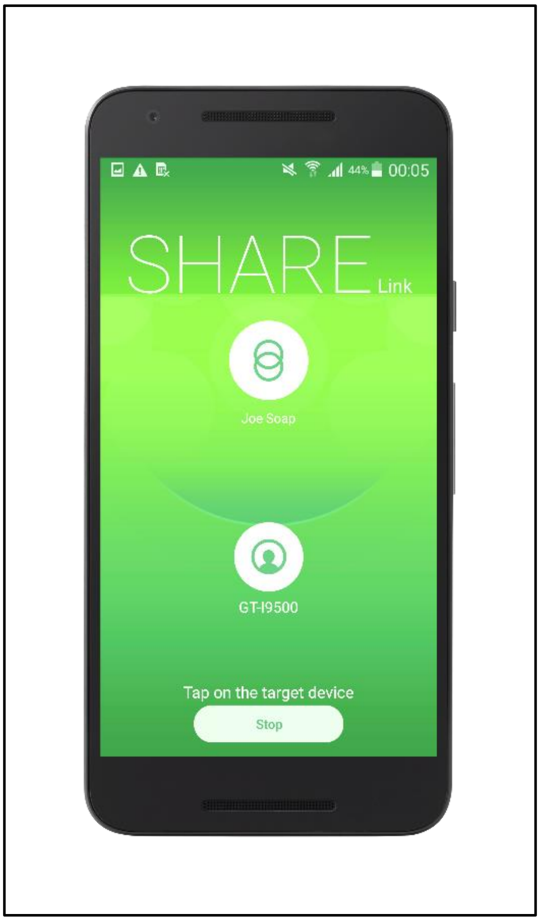


Figure 2.7: Scanning and Detecting Devices

2.7.3 SuperBeam

LiveQoS (2015) developed the SuperBeam application, which takes full advantage of the range of wireless technologies available to mobile devices, including NFC, Wi-Fi Direct, and quick response (QR) Code. It supports various types of information sharing. SuperBeam, similar to other existing information sharing applications, classifies the information stored on the mobile device per category in a list. After selecting the information to be shared, the sharing device has the option to share using QR Codes, NFC, or Wi-Fi Direct. Similarly, the receiving device has the option to receive information using the same communication technologies. The receiving device can scan the QR Code using the built-in QR Scanner. After successful scanning, the downloading of files commences.

One of SuperBeam's most distinguishable features is their *"blazing fast transfer speeds"*. This statement was tested through numerous file transfers (involving files of different types and sizes) to determine the transfer speed using SuperBeam. The speed was influenced by many factors; however, tests revealed speeds between 20 and 40 Mbps were achieved in most scenarios. Some tests with high-end devices, such as Samsung Galaxy S5 and S6, yielded even greater speeds of approximately 75 Mbps.

The SuperBeam PRO provides users with the following additional functionalities:

- Send and receive files to or from your desktop through SuperBeam for PC;
- Send entire folders and preserve their hierarchy;
- Share files with multiple devices simultaneously; and
- Pairing of devices using manual sharing key.

The manual sharing key was also identified to be another alternative sharing method using SuperBeam, which is only available in SuperBeam PRO. Selecting the *Web* tab, an image was displayed along with the relevant instructions on how to use Wi-Fi Direct (Figure 2.8). This image was deceptive as it gave the notion of SuperBeam being able to determine the physical locations of devices in the immediate area, which was proven to be false. This was achieved through visual confirmation of the devices in the area.

The uniform resource locator (URL) displayed in Figure 2.8 can be used by other mobile devices to download the intended files using a file downloader application, such as FileMaster (Shenzhen Youmi Information Technology Co. Ltd, 2015). The URL is entered into FileMaster and the files are directly downloaded onto the iOS device. This is an example of cross-platform information sharing using SuperBeam.

This application is an improvement on the traditional information sharing applications available on mobile devices (such as Bluetooth, Email, and Wi-Fi Direct) and offers the user a better information sharing method. SuperBeam provides users with a more detailed file hierarchy when selecting the file(s) to share and different sharing methods, all of which provide faster transfer speeds than the traditional methods.

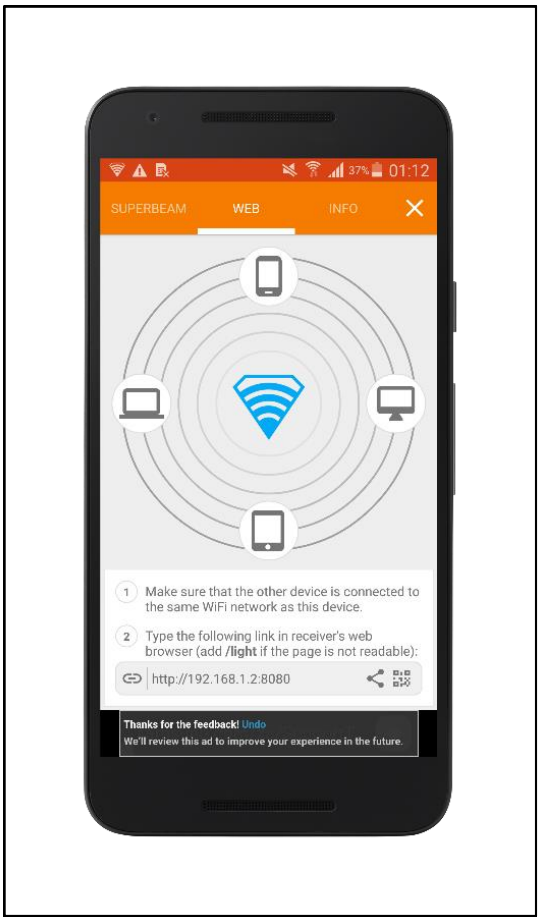


Figure 2.8: SuperBeam Wi-Fi Direct

2.7.4 Xender

Xender supports users in transferring multiple file types, namely documents, music, pictures, videos, and applications, with only a few selections (Anmobi Inc., 2015a). Xender also supports cross-platform transference and sharing between Android, iOS, PC, and MAC. The main screen of this application displays the various classifications of files into their respective types, where each type is uniquely displayed (Figure 2.9). The *Gallery* tab displays thumbnails of each image, whereas *Apps* showed tiles of the applications installed with each tile containing image, name, and size. The *Files* tab contains groupings of files by their type, which can be expanded by selecting a particular group to show the individual files belonging to that group.

Selecting the red "+" button reveals the options of Xender (Figure 2.9). One of the mobile devices is required to create a group by selecting the *Create Group* button. Only after the successful creation of the group, can the other device users select the red "+" button and select the *Join Group* button. This button initiates the scanning of available groups to join in the immediate area, whereby detected groups are displayed within the animated radar image (Figure 2.10). The group icon gives the perception that it represents the physical location of the device, which hosts the group. This was proven to be wrong because the device group was located somewhere else and the screen representation was elsewhere. Selecting the icon of the intended group allows for both devices to be automatically connected by means of a private Wi-Fi network.

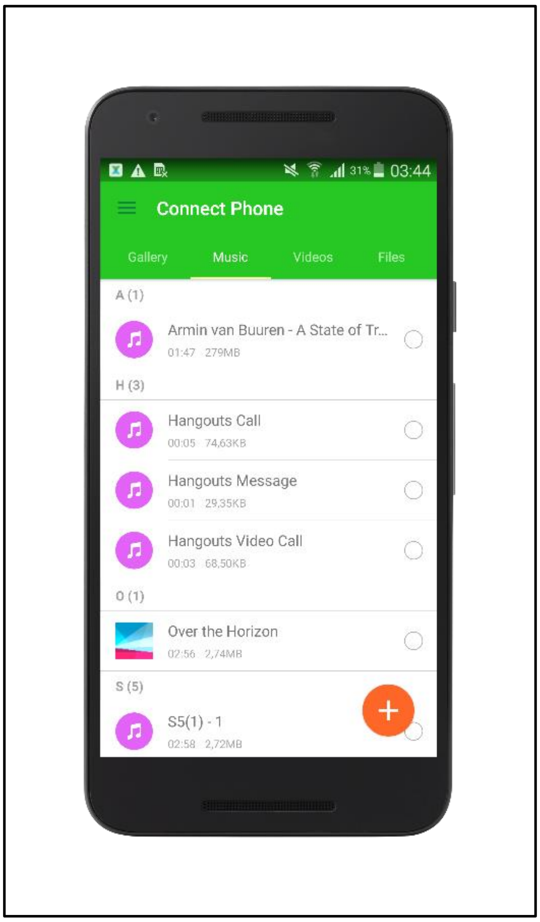


Figure 2.9: Xender Main Screen

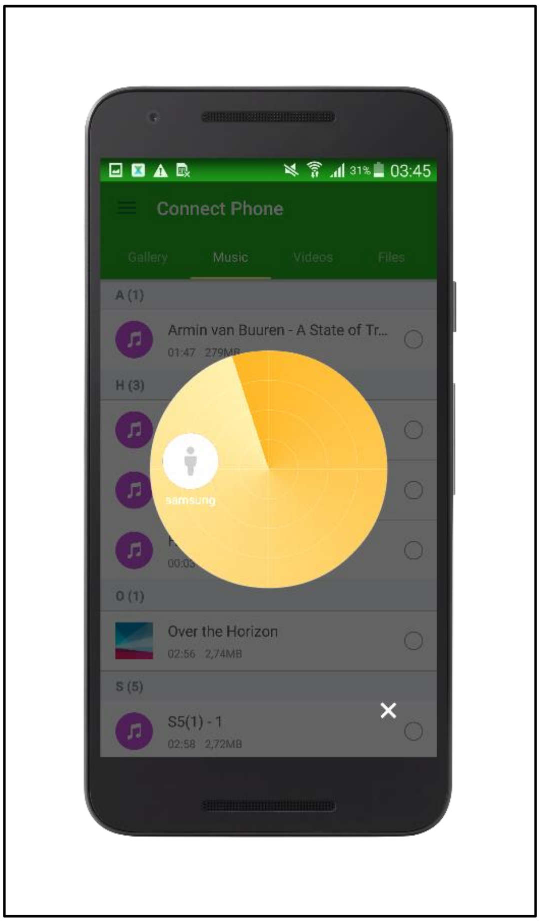


Figure 2.10: Xender Scanning Groups

When only smartphones are used, Xender supports group sharing of up to four devices (Anmobi Inc., 2015a). After a device has joined the group, information sharing can commence. To select the files, a single press is required. The files are transferred when the *Send(9)* button is selected (Figure 2.11). Xender's transferring screen is displayed on both the sending and receiving devices. Another sharing mechanism using Xender is the ability to turn on the *Shake to Send* option, which allows the mobile device to be shaken to initiate the file transfer.

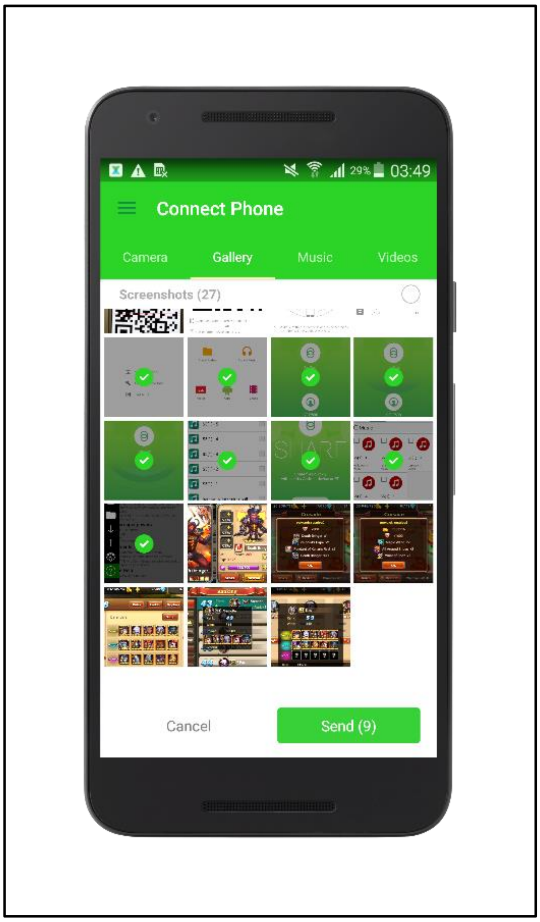


Figure 2.11: Xender File Selection

A long press on an individual file in Xender prompts additional menu options to be displayed. The *Slide* option allows for the selected file to be shared by swiping it to the right, regardless of the location of the receiving device in relation to the sending device. This was found to be deceptive and not as intuitive as one was led to believe because the YouTube demonstration placed two devices next to each other, whereby the right swipe to the receiving device on the right side suggested some form of determining device positioning (Anmobi Inc., 2015b). This was discovered when the two devices were placed in different positions. A shortcoming of Xender is the automatic connection to any group detected, which suggests these groups are insecure and vulnerable.

2.7.5 Information Sharing Process

The information sharing process was identified and derived from the investigation into existing information sharing applications, namely Feem (Section 2.7.1), Share Link (Section 2.7.2), SuperBeam (Section 2.7.3), and Xender (Section 2.7.4). The information sharing process typically consisted of the following steps, as shown in Figure 2.12. None of the existing applications reviewed allowed for the recipient(s) to be selected first as the traditional ways of sharing information, such as Bluetooth, Wi-Fi, and NFC, always selected the file(s) followed by the selection of recipient(s).

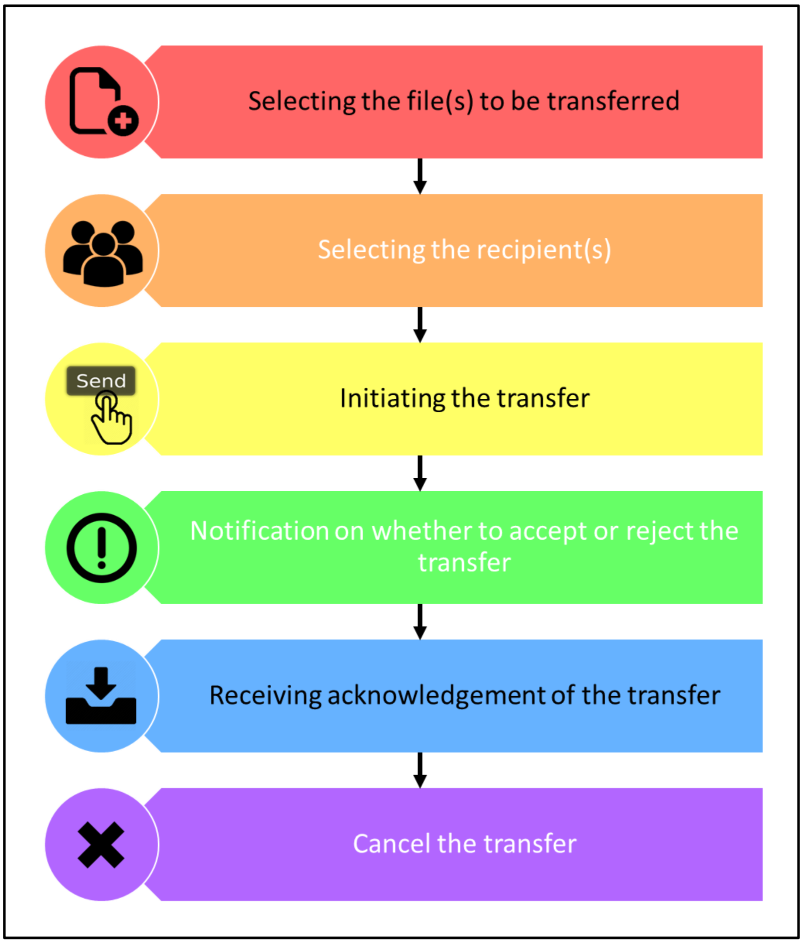


Figure 2.12: Typical Information Sharing Process

2.7.6 Summary

Several existing mobile applications were identified and discussed, which supported co-located information sharing. These applications were selected based on the defined problem (information sharing) and their characteristics. This discussion allowed for the common types of information sharing to be identified, namely images, music, videos, and documents. Through the discussion of these applications, the information sharing process was derived and will be used as the functional requirements in the development of MotionShare. Table 2.2 summarises the applications with their respective communication technologies used and the types of information sharing supported.

Table 2.2: Summary of Applications

Application Name	Communication Technology	Types of Information Sharing	File Selection	User Selection	Shortcomings
Feem	Wi-Fi	Photos, Music, Videos, Applications, and Files	• Single touch for selecting file(s)	• Single touch for selecting user(s) through a text list	• Manual file selection and traversing through a list of ambiguous or default device names
Share Link	Wi-Fi	Application, Music, Photo, Video, and File	• Single touch for selecting file(s)	• Single touch for selecting user(s) by icon	• Manual file selection and selecting receiver device icons with non-descriptive names • Unable to multi-task during the transfer process
SuperBeam	NFC, Wi-Fi Direct, QR Code, and Manual Sharing Key	Files and Folders, Audio and Music, Photos, Applications, Videos, Documents, and Contacts	• Single touch for selecting of file(s) • Select the button for select all functionality	• Receiving device scans the QR Code displayed on sending device • Transfer using NFC by holding devices close together • Receiving device enters URL into a file downloader application	• Manual file selection • Receiving device(s) need to scan QR Code displayed on sending device • Multiple device sharing using NFC is tedious with repetitive holding of devices • Multiple device sharing with Wi-Fi Direct requires the devices to enter URL to download file(s)
Xender	Wi-Fi	Applications, Camera, Gallery, Music, Videos, Files, and Storage	• Single touch for selecting file(s)	• Single touch for selecting user(s) through icon list • Initiate the transfer, either through *Shake to Send* option, *Slide* option, or select *Send* button	• Manual file selection • Traversing through a list • Slide option misleading • Automatic connection to any group detected

2.8 Conclusion

This chapter followed the first activity within the Relevance Cycle of the DSR methodology (Section 1.7.3.1), namely ***Explicate Problem***.

The definition of information sharing contributed to the explication and definition of the problem to be solved. Kaspersky Lab (2012) identified numerous types of information stored on different devices, but only information pertaining to smartphones was of interest since the research focus was on information sharing among co-located mobile devices. The information identified to be most commonly shared was reaffirmed through the analysis of existing applications used to support information sharing (Section 2.7).

Information sharing was identified to be important as it occurs frequently throughout every facet of life and continues to become more important as technology becomes more ubiquitous and integral. Taylor (2013) stated that *"In this day and age, almost every single person uses technology in some way, shape or form to make life easier"*. Thus, humanity has become dependent on technology. The information sharing importance model (Figure 2.3) can be applied to almost every aspect of life. Information sharing methods present the premise for natural information sharing as the foundational basis for digital information sharing methods. These digital information sharing methods are used within various domains.

Several mobile applications developed for information sharing in a co-located environment were identified and discussed. From these applications, the information sharing process was derived and will be used as functional requirements for the artefact to solve the defined problem. The shortcomings of existing information sharing methods were identified to be a manual file selection process that typically involved traversing through a list of ambiguous devices names.

This chapter achieved the first research objective (Section 1.7.2):

> *"RO1. To identify the shortcomings of existing information sharing methods currently used by mobile devices."*

The next chapter presents a literature study of NUIs, which includes the background, definition, objectives, and interaction techniques of NUIs. The relevant techniques are identified, which could potentially provide an accurate and usable solution to the research problem. Existing systems using a NUI and interaction techniques are also discussed.

Chapter 3: Natural User Interfaces

3.1 Introduction

NUIs are typically present in technologies, which allow users to perform natural movements to control the application or manipulation of on-screen content (Yao, Fernando, & Wang, 2012). Chapter 2 identified and discussed existing applications for information sharing using mobile devices. This led to the identification of the information sharing process of these applications, which will be used to determine whether a proxemic NUI can be used to provide an accurate and usable solution to the problem statement (Section 1.3).

The chapter addresses ***Outline Artefact and Define Requirements***, which is the second activity in the DSR methodology (Section 1.7.3.1) that is performed within the Relevance Cycle (Figure 3.1).

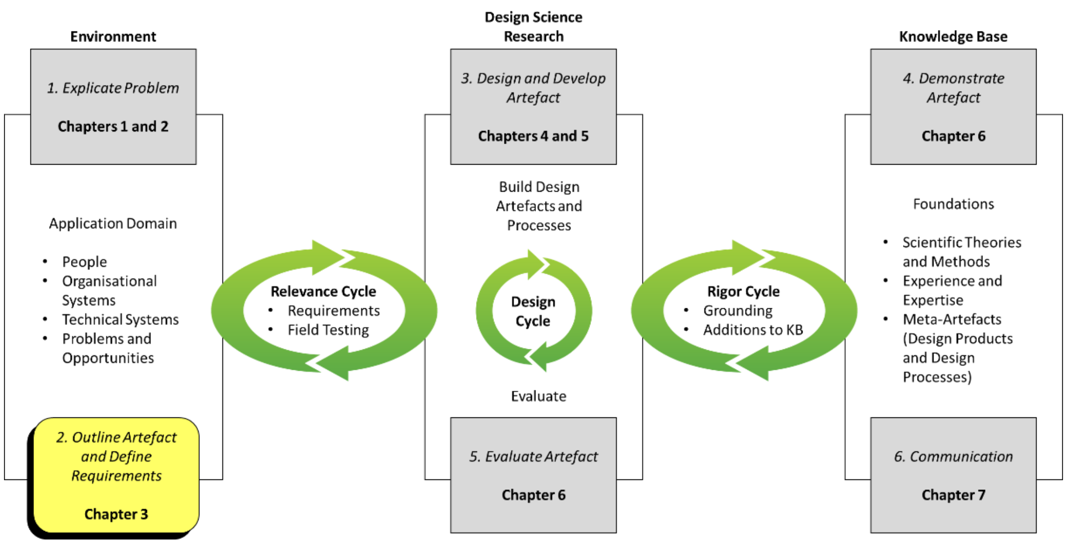


Figure 3.1: Chapter 3 Position in the Adapted DSR Methodology

This chapter covers the second research question (Section 1.7.1) identified:

> *"RQ_2. What are the benefits and shortcomings of existing NUI interaction techniques for information sharing?"*

This research question was addressed by investigating the background of NUIs. A review of NUI definitions and the primary objectives of a NUI are discussed. NUIs can consist of one or more interaction techniques, which are individually identified and discussed. The relevant interaction techniques are then analysed for their suitability, efficiency, and effectiveness to support information sharing among co-located mobile devices. Figure 3.2 illustrates the structure of this chapter.

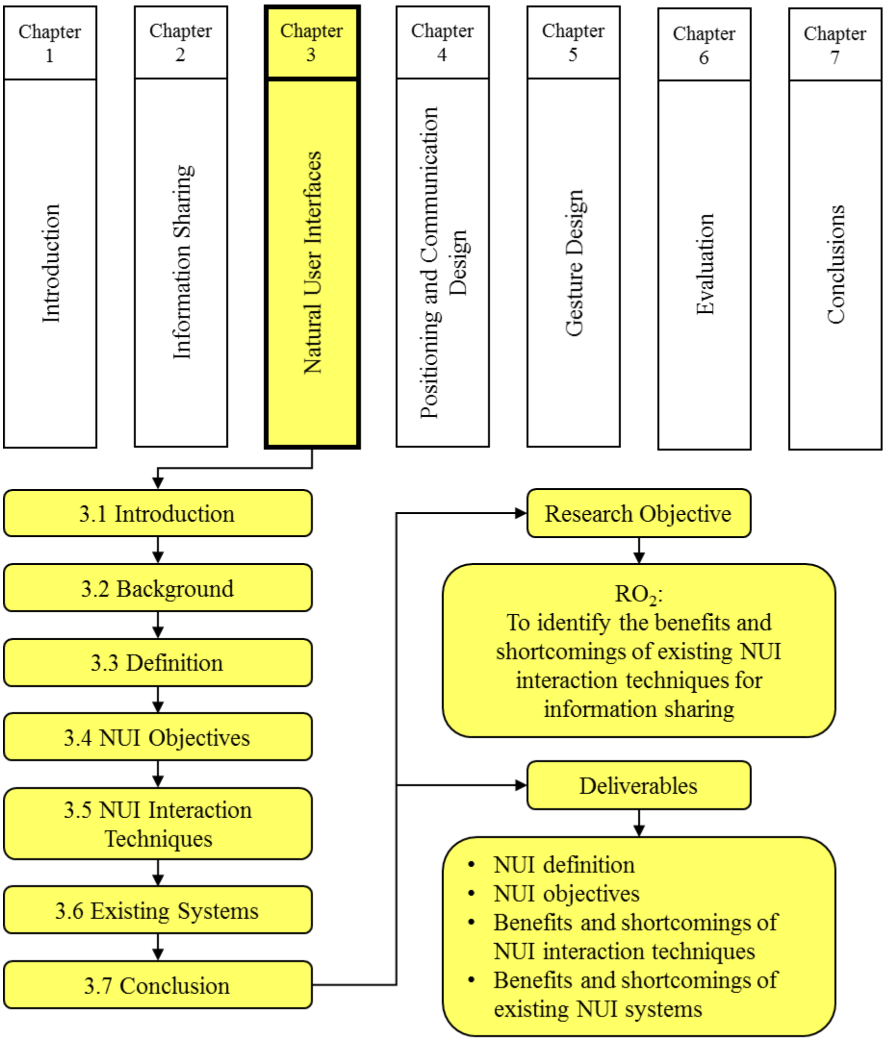


Figure 3.2: Chapter 3 Structure

3.2 Background

> *"The number of transistors, which can be placed inexpensively on an integrated circuit, will double approximately every two years."* - Moore (1970)

This quote is known as Moore's Law (Moore, 1970) and is an example that accurately describes the technological growth of computing and its adoption into various aspects of people's daily lives. Due to the increased processing power of computers as well as the decreased size and cost thereof, new form factors were created (smartphones), new platforms have evolved (the Internet), new infrastructures have become widely available and accessible (GPS), and new application families (document processing, image creation, modification and sharing) have flourished. All of these trends have caused the access to computing technology to continually grow as the number of people interacting with computers increases. Furthermore, as sharing information has become easier to do, it has changed the way people work, play, and interact with one another.

While the increase in computing power has occurred in a continuous manner, the same cannot be said about the evolution of interfaces between humans and computers. Interfaces have gone through different phases (Figure 3.3) which started with typing commands in a command line interface (CLI) and followed by the current establishment of a graphical user interface (GUI). Another phase in interfaces and the potential evolution in computing, is the NUI (Chong, 2013; Heikkinen & Porras, 2013).

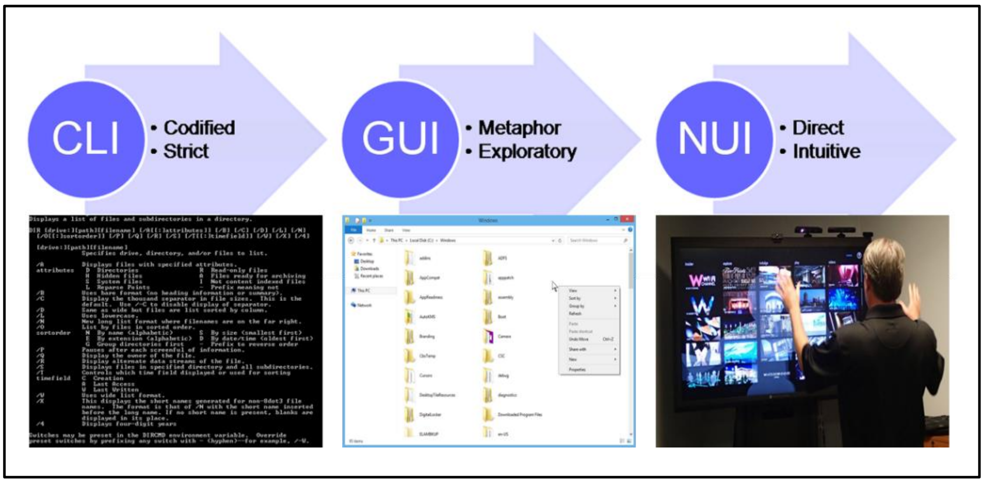


Figure 3.3: The Evolution of Interfaces (Heikkinen & Porras, 2013)

This is supported by Microsoft (2010), who identified NUIs to be the new generation of HCI. Microsoft (2010) explain that touch, motion detection, and speech recognition are components of an emerging field of computing commonly known as NUIs. These humanistic methods of communication with computing devices are now a reality from a statement made five years ago. Furthermore, Microsoft (2010) expect that everyone would enjoy using the technology in various ways, which are more suitable to the person, location, task, social context, and mood.

3.3 Definition

Recent developments in user-input technologies are changing the existing methods of interaction with computing devices (Câmara, 2011). The symbolic input devices, namely the mouse and keyboard, are used to provide data and control signals to PCs or laptops. These input devices are being replaced by touch and motion-based interfaces, increasingly known as NUIs.

Blake (2013) believes that NUIs were the next generation of interfaces to follow after GUIs. People are able to interact with NUIs using various interaction techniques, which includes multi-touch, gesture recognition, speech, eye tracking, and proxemics. NUIs are able to utilise more interaction options than previous computing interfaces, but NUIs are about more than just the computing device itself. NUIs present a new way of thinking regarding how people are able to interact with the content displayed on the devices.

The term *content* is used over any specific computing interface because the primary focus and desire of users is to obtain accessibility to their data, derive useful knowledge and learn from their information, or have the ability to interact with the content irrespective of the person's situation. This suggests different situations may potentially require different interface types. The expectation is to design NUIs to focus on content, irrespective of interfaces using multi-touch displays, 3D cameras, and voice detection. A popular example is the Microsoft

Kinect. Content is the only common link between any arbitrary interface techniques (Blake, 2013).

A universally acceptable NUI definition does not currently exist in literature. There are numerous existing definitions of NUI (Ballmer, 2011; Gantenbein, 2011; Wigdor & Wixon, 2011), and the term *NUI* is used to describe a wide variety of technology and systems being developed. Although there is no general consensus on the interpretation and understanding of NUIs, there are similarities identified in the various definitions that exist. Steinberg (2012) define *NUIs* as:

> *"Natural User Interfaces are interfaces, which are natural to use. This means the user is able to use the interface with little to no training. This is important as it reduces the cost of using the software, you do not need to train all the users to use it. It also means the users will enjoy using the interface."*

The most popular NUI definition in NUI literature is provided by Blake (2013), which is:

> *"A natural user interface is a user interface designed to reuse existing skills for interacting appropriately with content."*

This definition by Blake (2013) reveal three important aspects about NUIs, which are:

- ***NUIs are designed***
  The design of NUIs is a comprehensive process requiring planning and preparation. This design takes into consideration the NUI interactions appropriate to the user, task, device, and environmental context in which these interactions occur.
- ***NUIs reuse existing skills***
  NUIs focus on the aspect of being natural and in addition to this, the design is centred on human capabilities. Computing power and input technology have progressed to the point where designers are able to use existing non-computing skills and natural behaviours. NUIs support these skills and behaviours, by supporting HCI using intuitive actions, such as touching, gesturing, writing, and talking. Furthermore, interfaces are designed for

users to easily understand them through metaphors, which draw from their previous experiences of real world interactions.

The interface metaphors used within a NUI are in contrast to the metaphors used within GUI applications. A *GUI* is defined to be an interface primarily consisting of graphical representations. A GUI encompasses various interface types; however, almost all GUIs contain a specific set of interaction metaphors such as Windows, Icons, Menus, and Pointing devices (WIMP). The WIMP metaphors are useful and familiar to almost every person who uses a computing device, because people have been exposed to using GUIs for a long time.

NUIs most often refer to new input technologies such as touch, gesture detection, and voice recognition; however, a NUI is not about a specific input device or technique. NUIs are able to take advantage of any interface technology provided the interaction style focuses primarily on reusing existing skills.

- ***NUIs have appropriate interaction with content***

  NUIs make use of existing skills and natural behaviours to allow for the most appropriate user interaction technique with the content. Thus, the focus of NUI interactions is on the content and the appropriate technique.

Ballmer (2011) reported that Microsoft have invested a substantial amount of resources (time and money) towards the initial work for defining the field of NUIs. One notion was that technology would no longer be the driving force regarding the ways user interaction with content occurred. Instead, other factors, such as content, context, and activities would decide what interactions are the most *"natural"*. Another notion repeated throughout NUI literature is that appropriate interactions are essential in providing *"natural"* experiences, although only few examples of this are presented (Krummelbein & Nuur, 2013).

Feinzaig (2013) introduced a model which goes beyond a single interaction technique and presented individuals with a basic framework for selecting the appropriate method of interaction for any given activity (Figure 3.4). Existing interaction techniques with computing devices include everything from manipulating a mouse and keyboard, to touching, speaking, and gesturing. The model classifies these techniques according to screen proximity (namely *far* and *near*) and ease of learning ease of technology (namely *learned* and *natural*). An example of how to use this model is the following: in order to perform touch, the individual has to be located *near* the device screen, yet an individual can be several metres away when using gestures (*far*). Similarly, in considering how much time is required to learn a new technology, an older generation of technology typically took longer to learn (typing lessons or early CLIs), when compared to the newer generation which requires significantly less learning time (touch screens). The combination of these two ideas - proximity and ease of use – created this model (Figure 3.4), which Feinzaig (2013) identified as:

> *"It enables us to better envision where certain natural computing technologies play a role now and where they could grow in the future."*

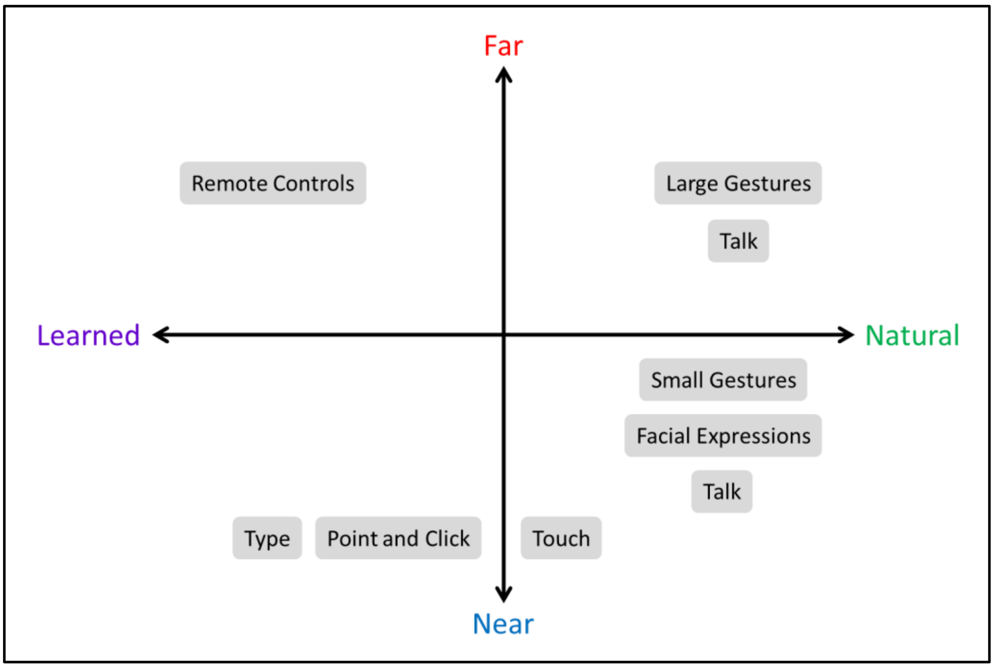


Figure 3.4: Natural Interactions Model (Feinzaig, 2013)

3.4 NUI Objectives

The investigation into the NUI field identified numerous definitions from literature (Ballmer, 2011; Blake, 2013; Feinzaig, 2013; Gantenbein, 2011; Krummelbein & Nuur, 2013; Steinberg, 2012; Wigdor & Wixon, 2011), which included the objectives of NUIs. These definitions do not promote one objective over the others.

3.4.1 Natural Interactions

NUI literature (Ballmer, 2011; Blake, 2013; Feinzaig, 2013; Gantenbein, 2011; Krummelbein & Nuur, 2013; Steinberg, 2012; Wigdor & Wixon, 2011) presents several predominant approaches to the NUI field. One of these approaches is that NUIs can be based on natural interactions. It is important to note that while NUIs can be based on natural interactions, these interactions are not restricted to this particular domain. Therefore, it is crucial to establish and emphasize the difference between these two terms, namely *NUIs* and *natural interaction*.

Natural interaction is defined as a user experience objective, which is not exclusive to NUIs, but has been researched extensively in the majority of all interaction fields (Tavares, Medeiros, De Castro, & Dos Anjos, 2013). Thus, it can be said *"a natural interaction is the effect of transparent interfaces, which are based on previous knowledge, and where the users feel like they are interacting directly with the content"* (Wendt, 2013). Therefore, the first objective of a NUI is based on the definition of a natural interaction, whereby the familiarity of the content and context ensures that the interactions can be understood by users.

3.4.2 Content First

Valli (2008) and Jain et al. (2011) explained the notion of *"creating transparent interfaces"*. This notion was extended by Wigdor and Wixon (2011), who mentioned creating enjoyable and user immersive experiences using NUIs. Blake (2013) discusses that the content should be the central focus and the supporting technology should be ubiquitous, thereby facilitating the content and user interaction in the best possible way. This discussion was aligned with Valli (2008), Jain et al. (2011), and Wigdor and Wixon (2011), who identified this NUI objective. Therefore, the second objective of a NUI is primarily content centric, whereby the NUI should not be considered a tool for content interaction but should instead facilitate the content.

3.4.3 From Context to Natural

NUI literature (Blake, 2013; Valli, 2008; Wigdor & Wixon, 2011) identify that the appropriate interaction technique can only be selected when the context is properly understood. A contextual understanding is essential when designing a NUI to support a specific context. Therefore, the third objective of a NUI is the importance of context, where understanding the context ensures the appropriate interaction techniques are selected and used.

3.4.4 Cognitive Load

Blake (2013) present the premise that reusing the skills already possessed by users reduces the cognitive load, thus freeing up the mental capacity to both comprehend the interactions and remember them for the next time the user must interact with the NUI. The cognitive load is not primarily focused on creating simple interfaces, but rather focused on directing the users' cognitive ability to be better utilised in a short time period, which is *"achieved by designing improved learning paths utilising germane load"* (Wigdor & Wixon, 2011). Therefore, the fourth objective of a NUI is to reduce the cognitive load experienced by users when interacting with a NUI.

3.4.5 Summary

Several NUI objectives were identified and discussed (Table 3.1). *Natural Interactions* is a crucial component to the success of NUIs. *Content First* emphasised how the design of NUIs needs to be content centric. Literature identifies the selection of the relevant interaction techniques as being determined through the understanding of the context (*From Context to Natural*). *Cognitive Load* addressed designing NUIs to reduce the cognitive load experienced by users.

Table 3.1: NUI Objectives

NUI Objectives	Description
Natural Interactions	The success of NUIs is dependent on understanding natural interactions
Content First	The primary focus of NUIs should be the content and the supporting technology should be pervasive
From Context to Natural	The appropriateness of NUI interaction techniques is context dependent
Cognitive Load	Using NUIs can reduce the cognitive load experienced by users

3.5 NUI Interaction Techniques

Continual advancements in the field of computing technology have provided potential opportunities for creating the next generation of UIs, which are more intuitive and interactive than ever before (Kaushik & Jain, 2014). NUIs can provide a more natural HCI by supporting users with various interaction techniques, such as touch and stylus, gestures speech, and proxemics. These techniques are discussed, which includes a definition, an explanation of how they function, and highlights their benefits and limitations.

3.5.1 Touch and Stylus

> *"To bring a bodily part into contact with the system especially so as to perceive through the tactile sense."* - Lightfoot (2010)

This description can be translated into the context of NUIs as using a bodily part to directly contact with the NUI through tactile sensing to either interact with, appreciate, or understand it. Humans have an innate ability to touch, developed in the early stages of infancy. According to Koueider (2013), Steve Jobs once said:

> *"We are going to use the best pointing device in the world. We are going to use a pointing device we are all born with – born with ten of them. We're going to use our fingers."*

He was right because people are more inclined to use the device with which they are most comfortable and connected to, as opposed to one which invokes an alienated response. As a person grows older and through repetition, this tendency automatically matures. Therefore, NUIs support natural movements performed by users, such as touch, to communicate with the application or manipulate information displayed on the screen (Blake, 2013).

An alternative input device is the stylus. A stylus is a small pen-shaped instrument used to support the user in performing numerous actions, such as writing, selecting, or drawing, on a computer screen or mobile device. Touchscreen devices allow users to place the stylus on the screen surface to draw or make selections by tapping the stylus on the screen. These pen-like input devices provide enhanced accuracy for content selection and handwriting (Wigdor & Wixon, 2011).

Typically, users of mobile devices use their fingers to directly interact with the information on their screens (Yao et al., 2012). In recent years, the advancement of mobile devices (smartphones and tablets) has resulted in the standard interaction with these devices being touch-based (Apple Inc., 2015; Microsoft, 2015c; Samsung, 2015). Touch interaction has also advanced from single-touch to multi-touch detection. Multi-touch is the ability of the interface, typically a touchscreen, to recognise multiple points of contact with the interface (Alvarez, Brown, & Nussbaum, 2011).

Figure 3.5 presents the core touch gestures classified into two types, namely single-touch and multi-touch (Bank, 2014). These gestures are implemented in various applications and are already associated with specific actions, such as single tap for selection and pinch for zooming. Some gestures, but not all, are grouped according to basic, object-related, navigating, and drawing actions (Villamor, Willis, Wroblewski, & Fulton, 2011).

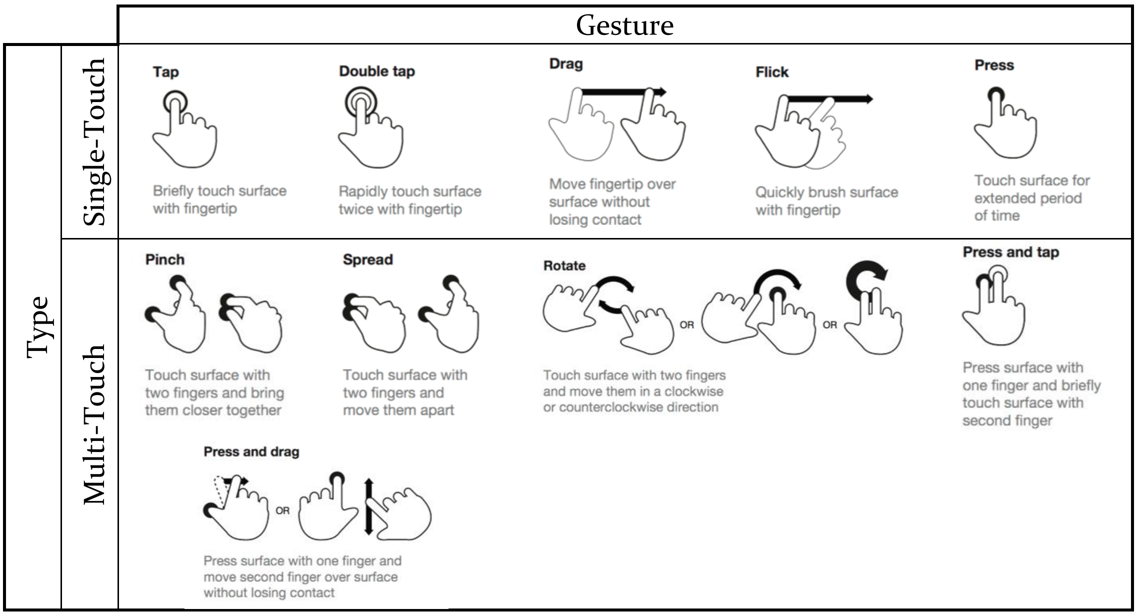


Figure 3.5: Core Touch Gestures (Bank, 2014)

Kin, Agrawala, and DeRose (2009) identifies the following benefits of touch interaction:

- Selecting from a list of choices using touch interaction requires little thinking and is a form of direct manipulation, which is easy to learn;
- Touch interaction is the fastest pointing method;
- Individuals have easier hand-eye coordination with touch screens than with the mouse and keyboard;
- No extra workspace is required when using touch; and
- Intuitive tactile response is unparalleled.

Although touch interaction may seem natural or easily learnt, there are several shortcomings (Budiu, 2015):

- The users' hands may potentially obscure the screen itself, which prevents any visual feedback from being seen by the user;
- Virtual keyboards with their small buttons and lack of tactile response are notoriously inaccurate and frustrating to use;
- Touch gestures can be misinterpreted by the system and lead to incorrect responses, which may quickly become annoying to the user;

- Screens need to be installed at a lower position and tilted to reduce arm fatigue; and
- Technology supporting touch interaction is more costly.

Due to the popularity and norm of touch interaction in mobile devices, research has been conducted to eliminate these drawbacks. Improved visual techniques are used to improve the feedback received from touch interaction. These techniques are authentic motions, responsive interactions, meaningful transitions, and delightful details. An example of a system providing responsive surface interaction when a point of contact is made using the device screen is Touch Ripple. A Touch Ripple indicates where and when a touch input occurs and the system acknowledges that the touch input was received (Google Inc., 2015f).

Design principles are used to create hierarchy, meaning, and focus to immerse the user in the experience, and improve precision through better designed virtual components and layouts (Google Inc., 2015e). In recent years, costs of touch interaction have reduced significantly because of the technological progression of mobile devices as well as the research and development into lowering these costs. Gestures is the next interaction technique which is discussed in the next section.

3.5.2 Gestures

Billinghurst et al. (2014) defines a *gesture* as:

> *"A gesture is a motion of the body that contains information. Waving goodbye is a gesture. Pressing a key on a keyboard is not a gesture because the motion of a finger on its way to hitting a key is neither observed nor significant."*

Gestures refer to any motion involving physical movements of a user's body, for example, hands or fingers. Gestures have a strong presence in various computing devices, such as mobile phones, tablets, laptops, and navigation systems (Loureiro & Rodrigues, 2011). Gestures provide a natural, direct, and intuitive way of interacting with a computing device, allowing easier HCI for all types of users, including the elderly (Hollinworth & Hwang, 2011).

There are two types of gestures, namely touch and in-air gestures (Chen, Schwarz, Harrison, Mankoff, & Hudson, 2014). Touch gestures are mainly found in touch screen interfaces where the user performs a specific gesture to achieve a certain system response. These gestures are classified with touch interaction, which was discussed earlier (Section 3.5.1). In-air gestures are any movements of the user's body that are recognised by the system without touching the screen (Agrawal et al., 2011). In-air gestures are prone to several limitations (Bratitsis & Kandroudi, 2014; Song et al., 2014), namely:

- The user has no implicit intention of performing a gesture, however the system has detected it (false positive error);
- The user believes the gesture was performed, because it was actually performed, however the system failed to detect it (false negative error);
- Performance of in-air gestures can lead to user fatigue;
- In-air gestures are only recognised in a specific detection area, dependent on the hardware constraints and placement of the camera;
- Social acceptability with regard to people seeing in-air gestures being performed; and
- In-air gestures are context dependent.

Ho and Weng (2013) identifies the increased prevalence of devices to facilitate and utilise in-air gestures, which is evident with commercially available products, such as Leap Motion (Leap Motion Inc., 2015), Intel RealSense Camera (Intel Corporation, 2015), and Microsoft Kinect (Microsoft, 2015b). However, these products are additional hardware typically used in conjunction with desktop computers and not mobile devices.

Typically, gestures are used in various input tasks, such as the navigation of maps or images, data entry, cursor control and item selection, and verification of user identity (Kamal, Li, & Lank, 2014; Ruiz, Li, & Lank, 2011). There is no doubt gestures play a crucial role in people's daily lives, specifically when communicating with other people (Curccurullo, Francese, Murad, Passero, & Tucci, 2012). One of the benefits of using gestures is its natural feel because gestures naturally accompany speech interaction. A major limitation of gestures is the gesture recognition. If the gesture technology is located in an environment where the background is not highly contrasted to the objects, people, and ambient lighting conditions, then the gesture recognition is vulnerable to inaccurate responses. Another limitation is the system's inability to recognise the start and end points of a gesture when the user is performing continuous movements (Rautaray & Agrawal, 2015).

Gesture interaction is considered an advanced and natural form of interaction with successful applications in several fields, with the greatest focus on game applications. Although this technique is intuitive to use, it has not replaced the more traditional interaction techniques like the keyboard and mouse. Continual research and advancements in this field show great promise and it has become an accepted form of HCI, which is evident in smart TVs, smartphones, and gaming platforms, such as Microsoft Xbox 360 (Microsoft, 2015d) and Nintendo Wii U GamePad (Nintendo, 2015). Speech is the next interaction technique which is discussed in the next section.

3.5.3 Speech

Speech is the most natural form of communication and is transparent to users. Individuals are typically taught to speak a language at an early age (Wagner, Malisz, & Kopp, 2014). As such, speech is comfortable and fast to perform. Existing NUIs use speech interaction to capture, identify, and understand the user's intent based on the words spoken (Hung & Gonzalez, 2013). Speech interaction is facilitated through the use of a microphone, which is used as the input device, and speech recognition algorithms interpret the input received (Wang, Furui, & Juang, 2013). The prevalence of speech interaction has highlighted the benefits of use in various fields, namely driving (Tchankue, Wesson, & Vogts, 2014), speech-to-text systems (Kushalnagar, Behm, Kelstone, & Ali, 2015), and healthcare (Rudzicz, Wang, Begum, & Mihailidis, 2015).

Nirjon, Dickerson, Stankovic, Shen, and Jiang (2013) identified the following limitations in speech interaction:

- ***Inaccuracy and Slowness:*** Most individuals are not able to type as fast as they can speak. In theory, this should make speech recognition algorithms faster than data entry typing; however, they are not because the systems struggle to recognise speech correctly.
- ***Vocal Strain and Fatigue:*** Individuals may need to speak louder than normal in order for the speech recognition to capture the user's voice.
- ***Environmental Factors:*** Noisy environments can generally cause speech recognition algorithms to fail.
- ***Interpretation of Accents:*** Individual's pronunciation of words differ because of various dialects and regional accents that cause speech recognition algorithms to fail.

The advancements of speech interaction is a result of the research conducted to overcome these limitations. Multi-modal HCI is the future technology (Tiwary & Siddiqui, 2012). Therefore, significant research efforts are being conducted into advancing speech interaction to overcome these limitations. The last interaction technique, namely Proxemics is discussed in the next section.

3.5.4 Proxemics

Dingler et al. (2015) defines *proxemics* as:

> *"Proxemics is a theory about people's understanding and use of interpersonal distances to mediate their interactions with other people."*

Thus, proxemic interaction is the interaction of a system with various entities, namely people, mobile devices, and non-digital objects, based on the five proxemic interaction dimensions for ubiquitous computing. Marquardt (2013) identified these dimensions (Figure 3.6) as:

- ***Distance*** is the discrete length of space measured between entities;
- ***Orientation*** is either continuous (pitch, roll, or yaw angle of one entity relative to another) or discrete (facing direction);

- ***Movement*** captures the distance and orientation of an entity over time;
- ***Identity*** is the unique description of an entity; and
- ***Location*** describes the physical context of residing entity.

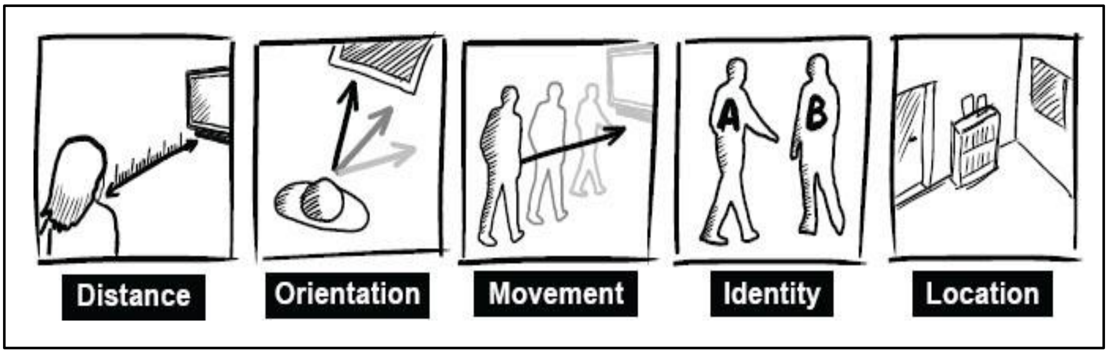


Figure 3.6: Five Dimensions of Proxemic Interaction (Marquardt, 2013)

This research covers distance and orientation, whereby the distance between devices and their orientations were considered. *"Proxemics correlates people's physical distance to social distance"* (Sørensen, Kristensen, Kjeldskov, & Skov, 2013), which means that specific interaction types occurs in different proxemic zones. Sørensen et al. (2013) classify these zones in Figure 3.7.

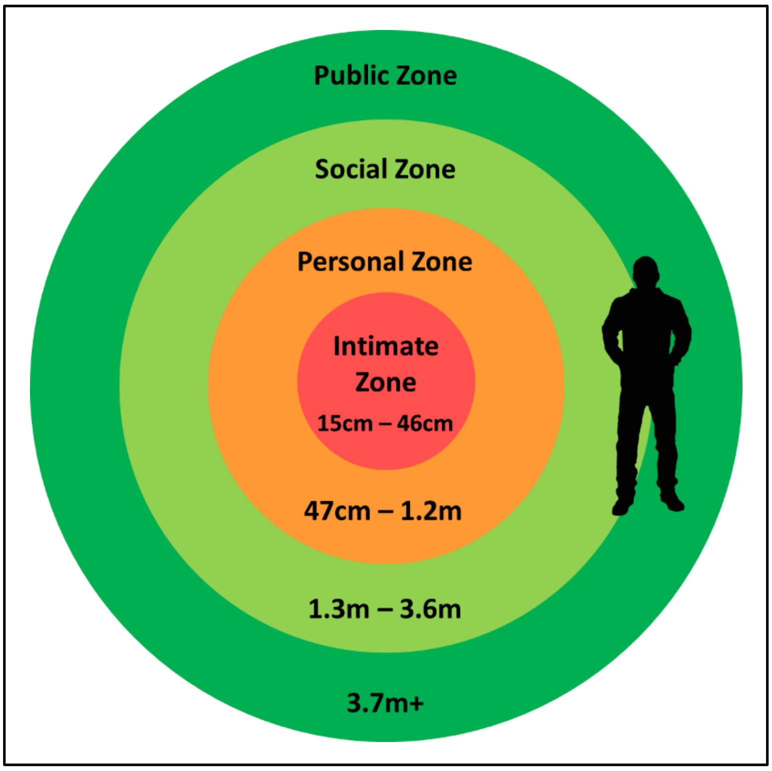


Figure 3.7: Proxemic Zones (Sørensen et al., 2013)

Proxemic interaction technology is aware of the user's presence, position, movement, and identity, and takes these attributes as implicit inputs for interaction. A proxemic interactive display can instantaneously provide context specific information to the user in a seamless manner whereby technology is ubiquitous. Proxemic interaction can become problematic in a high-density environment, whereby unexpected breaches (from the system and other users) in a user's personal space can lead to negative reactions.

Ubiquitous computing is gradually becoming a reality with people using various computing devices to communicate with each other. These computing devices include computers, smartphones, tablets, smart watches and other digital intelligent devices that have created a ubiquitous society. The continuous development of different interaction techniques suggest their future prevalence in everyday life. This can already be seen in existing devices such as smart TVs, smartphones, and Microsoft PixelSense (Microsoft, 2015a). All of these devices use one or more of the proxemic dimensions, namely distance, orientation, movement, identity, and location, to provide enhanced system functionality.

3.5.5 Summary

Several interaction techniques associated with NUIs were discussed. These techniques were touch and stylus, gestures, speech, and proxemics. The prevalence of touch interaction is evident in mobile devices, such as smartphones and tablets, and as a result, the design artefact is expected to implement some form of touch interaction. Gestures were identified to provide a natural, direct, and intuitive way of HCI, either through touch or in-air gestures. Touch gestures were typically present in touch screen interfaces, such as smartphones, and classified as touch interaction. *In-air gestures* were defined as any movements performed by the user without physically touching the device screen. These in-air gestures are deemed irrelevant to this research based on the empirical evidence from literature suggesting that in-air gestures are not socially acceptable and context appropriate. Speech interaction was identified as the most natural form of communication; however, it is not relevant to the defined problem due to the limitations identified. Proxemics utilise the user's presence, position, movement, and identity for system interaction. This is potentially useful when users are changing the orientation of their mobile devices to share information, such as facing the recipient device(s).

3.6 Existing Systems

Several NUI systems are available which use different interaction techniques depending on the context. These systems are identified and discussed to show how NUI interaction techniques are used in supporting the specific context of each system. This section provides an insight and understanding into which interaction techniques should be applied to the defined problem.

3.6.1 MobiSurf

MobiSurf (Seifert et al., 2012) was designed to facilitate various types of collaboration by providing seamless integration of mobile devices and a shared interactive surface. The following interactions used with MobiSurf are:

- Placing the mobile device on the surface;
- Using the mobile device's camera to detect its location relative to the surface; and
- Detecting dragging gestures across the displays.

The MobiSurf concept was developed using guidelines provided by various related systems (CoSearch, WebSurface, and SearchTogether). MobiSurf consists of two components, which are the mobile devices and the two web browsing applications (Figure 3.8). The web browser application was designed to operate on the shared surface, which displays numerous browser windows and is moved using the corresponding handle located at the top of the window (Figure 3.9). These windows support touch interaction with the content, virtual keyboard for data entry, and components, such as links, buttons, or scrolling.

Figure 3.8: MobiSurf on Shared Surface and Mobile devices (Seifert et al., 2012)

Figure 3.9: Shared Browser Application on Interactive Surface (Seifert et al., 2012)

The web browser application on the mobile devices was designed for the Android platform. This application allows the transfer of web pages with the surface and other mobile devices by transferring the URLs of the respective web pages. When a MobiSurf user wants to transfer a web page from their mobile device (Figure 3.10(a)) to the surface, the user must touch the surface with his device at the desired point (Figure 3.10(b)). MobiSurf detects the touch interaction on the surface performed by the mobile device user. The URL of the web page is transferred via Wi-Fi and then displayed on the surface (Figure 3.10(c)).

(a) (b) (c)

Figure 3.10: Transfering Information from a Mobile Device to the Surface (Seifert et al., 2012)

MobiSurf also allows mobile device users to transfer web pages directly to other mobile devices. The sender device is required to display the web page that the user wants to share (Figure 3.11(a)). The receiver device displays the home screen. To share the web page, both devices are held close together (Figure 3.11(b)) because the web page is transferred using NFC. The receiver device displays the web page received (Figure 3.11(c)).

(a) (b) (c)

Figure 3.11: Transferring Information between Mobile Devices (Seifert et al., 2012)

PhoneTouch (Schmidt, Chehimi, Rukzio, & Gellersen, 2010) was used within MobiSurf to provide a quick interaction style for transferring information from a mobile device to the surface and vice versa. PhoneTouch is based on using the touch interaction technique. To transfer the information with PhoneTouch, the mobile device user simply touches the surface with his device (*dropping them down*) and the selected information is then transferred and displayed at the initial point of the touch. Similarly, PhoneTouch allows mobile device users to transfer information from the surface to their mobile device by performing a touch interaction (*picking them up*) with their device at the desired object.

3.6.2 Gesture On

Gesture On (Lü & Li, 2015) was designed to enable the user to draw a pre-defined gesture on the mobile device screen in standby mode, which means the user can draw the gesture before the screen is turned on. Depending on the gesture drawn, the system responds by directly bringing up the desired item on the screen, thereby bypassing all the additional steps typically involved in mobile access. These additional steps are:

- Device is woken up by selecting the power button (Figure 3.12(a));
- Device is unlocked by authenticating ownership, such as swipe, pattern, pin, or password (Figure 3.12(b));
- Device is unlocked and home screen displayed (Figure 3.12(c)); and
- The desired application or functionality can be searched for on the device (Figure 3.12(d)).

(a) (b) (c) (d)

Figure 3.12: Screenshots of Gesture On (Lü & Li, 2015)

These additional steps are eliminated with the use of Gesture On. For example, this application can allow the user to make a phone call by drawing a gesture on the blank screen (device is in standby mode). Gesture On activates the device from standby mode, authenticates the user, and then performs a search on the device based on the gesture completed when the user uses Gesture Search.

Gesture Search (Li, 2010) allows quick access to information on the mobile device (contacts, applications, bookmarks and music) through gesture-based interaction. Gesture On returns a list of potential results of the performed gesture, whereby the user can select the friend's name from these results and immediately perform a phone call action. The detailed processing flow is presented in Figure 3.13.

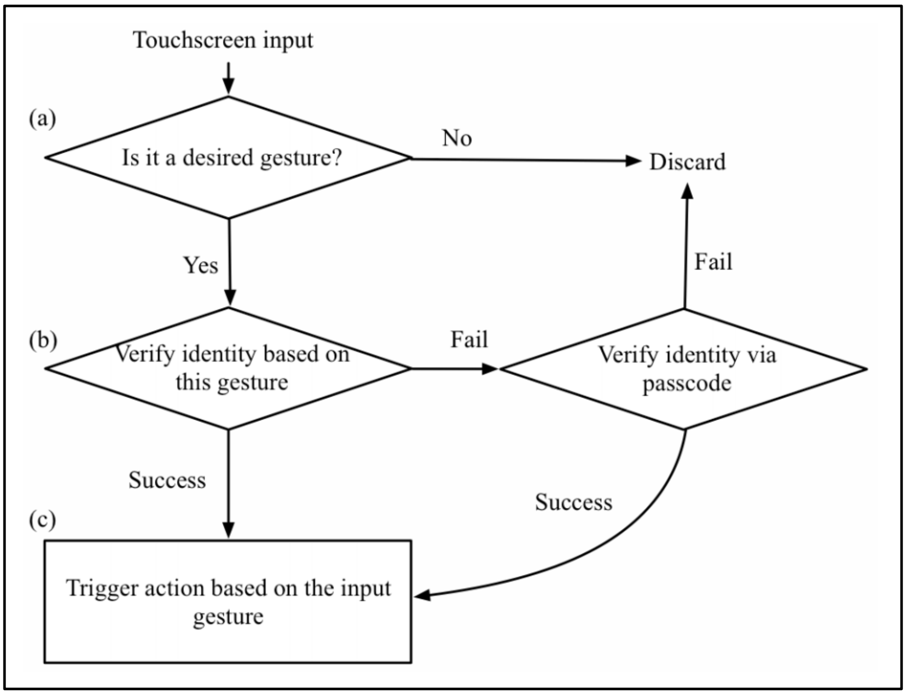


Figure 3.13: Processing Flow of Gesture On (Lü & Li, 2015)

The two core elements of this flow are the filtering out of accidental touch interactions and verifying user identity based on the gesture performed. Firstly, Gesture On determines whether the touch gesture was accidental or not. In the event that it was accidental, the system ignores the gesture. If the event was intentional, then Gesture On wakens the device and processes the gesture. Gesture On verifies if the input gesture is from the same user by authenticating it. If the authentication process fails, the unlock option typically used by the user (swipe, pattern, pin, or password) is shown. After successful authentication, the system launches Gesture Search with the entered gesture as a search query.

3.6.3 AirLink

AirLink (Chen, Ashbrook, Goel, Lee, & Patel, 2014) is an application that facilitates information sharing among multiple devices through the user performing in-air gestures (Figure 3.14). Information is shared when the user waves their hand from one device to another. This application measures the Doppler shift that is caused by hand motions using the built-in microphone and speaker in mobile devices. The Doppler shift is when a shift in the signal frequency occurs, which was caused by the hand motion.

Figure 3.14: Information Sharing Using AirLink (Chen et al., 2014)

In-air gesture recognition by measuring the Doppler shift was first demonstrated in SoundWave (Gupta, Morris, Pattel, & Tan, 2012). Although both applications are similar in concept, the objectives and algorithms are different. SoundWave was designed to operate only on a single laptop to detect gestures, whereas AirLink was designed for multiple device interaction. AirLink combines all the signals from the devices and assigns a code word which is used to identify the various combinations of hand gestures (Figure 3.15). By combining the hand gestures and Doppler shift detected from multiple devices, AirLink is able to recognise the origin, destination, and direction of a hand gesture. *T* represents a movement *towards* a device, *A* represents *away*, and *X* represents *towards then away*. Therefore, performing a specific hand gesture is seen as a sequence of movements referred as a code word. Figure 3.15 illustrates the gestures and corresponding code words which is possible with only three devices.

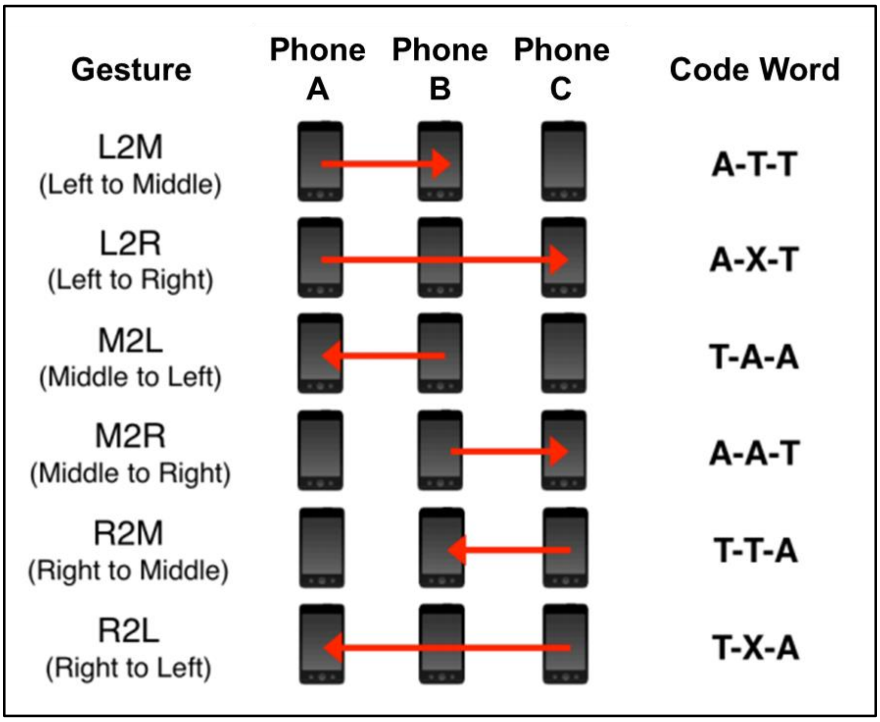


Figure 3.15: Code Word Assigned to Each Gesture in AirLink (Chen et al., 2014)

Every device using AirLink generates an ultrasonic pilot tone from its speakers. To share information the user simply moves his hand from the initiating device to the receiving device. The hand motion reflects the ultrasound emitted from the speakers, thereby causing a Doppler shift. AirLink was robust in a 3-phone scenario and discussed the introduction of more devices, which meant longer corresponding code words would be required and assigned to the additional devices. AirLink does not know the actual relative positions of the devices, but relies on the characters in the code word to send information to a specific device.

3.6.4 Flick

Flick (Ydangle Apps, 2013) is a mobile application supporting information sharing among multiple devices on different platforms. This application supports the sharing of documents, videos, music, images, and contacts. Flick can be installed onto Android devices, iOS devices, Macs, and Windows PCs. Information is shared by performing a flick gesture on the device screen. In order for the information to be transferred in Flick, all the devices must have Flick installed and need to be connected to a Wi-Fi network. Flick displays a welcome dialogue and gives the user the option to enter a device name.

The main screen of Flick consists of three components, namely a top, middle, and bottom (Figure 3.16). All the Flick devices in the immediate area are detected and listed in a horizontal scrollable tab list (top), which displays the device name, connected status, and computer icon. The device currently selected is highlighted to show this is the receiving device. The middle component is a blank space, which displays the selected information to be shared (Figure 3.17). After each file share, the file is automatically deleted from this space to reduce clutter. This functionality can be turned off in the options. The bottom component of Flick presents the various items available. When the user wants to share information to another device, the user must scroll through the horizontal tab list to the intended device name.

The intended device must be selected and is highlighted to show it is now the receiving device. Initial expectations of Flick was that the gesture performed was directional, which meant flicking in the direction of the intended receiver. However, the flick gesture is only performed upwards to initiate the information sharing process regardless of the receiving device location, for example, positioned to the left or right. When a Flick user closes the application, the corresponding device name is removed from the list after the update time. This update time is set in the Options screen with different time intervals, ranging from one minute to one hour.

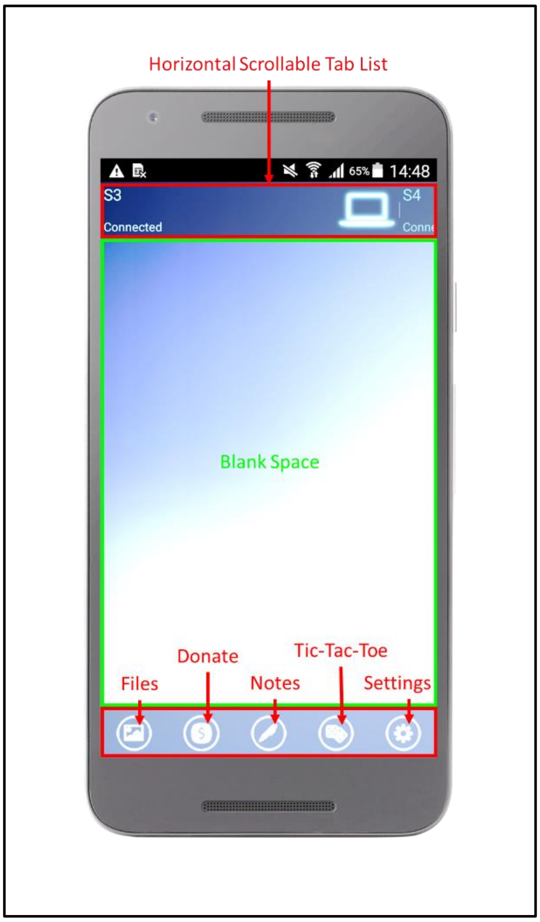


Figure 3.16: Flick Main Screen

Figure 3.17: Selected File Displayed

3.6.5 Zapya

Zapya is a tool used to facilitate cross-platform information transfer and sharing (DewMobile Inc., 2015). Zapya supports the transfer of various information types, namely applications, images, videos, audio recordings, and files. People are able to use Zapya on Android devices, iOS devices, Windows Phones, Macs, and PCs. The main screen of Zapya, similar to existing information sharing systems, presents the file structure where all the information stored on the device is classified according to different tabs and displayed in a list. The navigation drawer displays a list of items, an opportunity to change the avatar and the display name.

Pressing the floating action button, which is located on the bottom right of the main screen, displays four options, namely:

- ***Help*** displays a slide show of how to create and search for a group;
- ***Create Group*** creates a private Wi-Fi hotspot, which can be password encrypted;
- ***Search and Join*** searches for all available groups and randomly displays the avatar on the radar (Figure 3.18); and
- ***Go Back*** displays the Zapya main screen.

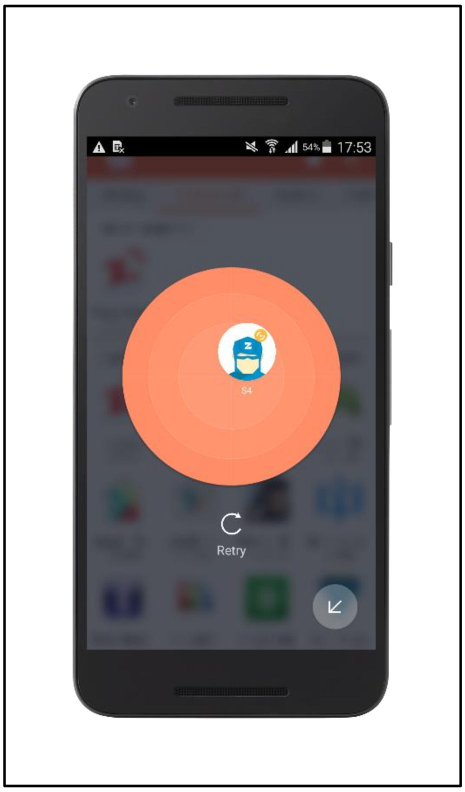


Figure 3.18: Searching for Available Groups

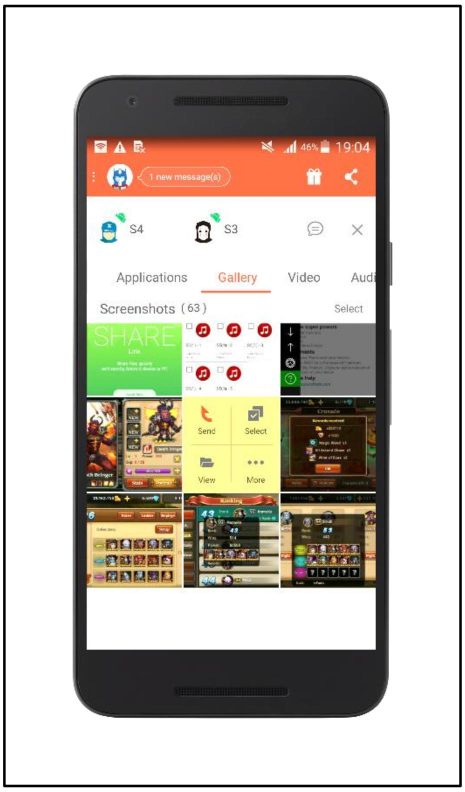


Figure 3.19: Zapya Four Tile Option Overlay

After the search is completed for available groups, the user can join a group by selecting the group avatar (Figure 3.18). Once the user has joined the group, all the group members' avatars and names are displayed near the top of the screen. Each member's avatar is accompanied by another icon to show device type. The *speech bubble* icon represents the instant messaging where all group members can communicate similar to WhatsApp. The *speech bubble* glows when new messages are received to notify the user. The *cross* icon represents the exiting from the group. Selecting a file causes Zapya to display a four tile option overlay (Figure 3.19). A long press and holding a file displays an instructional message for the user to perform the flick gesture. Performing the flick gesture shares the file to all group members. Zapya users can drag the file to the receiving device's avatar. The *history* tab displays the history of previous information shares. The *history* tab also shows the progress of existing information sharing.

3.6.6 Summary

Several NUI systems using different interaction techniques were discussed. MobiSurf (Section 3.6.1) facilitates interaction between mobile devices and a shared interactive surface. Information is shared between these device types through touch interaction. In order to share information from mobile devices to the shared surface, the user is required to touch the surface using his device, which is supported by PhoneTouch. The secondary option to share information is holding the devices close together and using NFC.

Gesture On (Section 3.6.2) was designed for users to draw touch gestures on the screen in standby mode. Users are able to perform touch gestures and Gesture On responds before the device screen is turned on. The processing flow of Gesture On (Figure 3.13) is appropriate as MotionShare needs to identify the gestures performed by the user and respond accordingly. AirLink (Section 3.6.3) supports information sharing among multiple mobile devices through in-air gestures. A user performing in-air gestures is detected by AirLink through the measurement of the Doppler shift. The Doppler shift caused by in-air gestures are measured using the embedded microphone and speaker in mobile devices.

Flick (Section 3.6.4) supports information across different platforms using touch gestures. Flicking the selected information towards the top of the screen initiates the transfer process. The receiving device is selected through a horizontal scrollable list located at the top of Flick. Zapya (Section 3.6.5) supports cross-platform information sharing and transferring of different file types. Zapya shares information using two methods. The first method is the long press and holding the selected information and the device with the same hand. The device is flicked with the user's wrist and information is shared. The second method involves using a long press and dragging the selected information towards the receiving avatar displayed at the top of Zapya. Table 3.2 summarises the NUI systems with the respective interaction techniques used.

Table 3.2: Summary of NUI Systems

Name	Interaction Technique	Benefits	Shortcomings
MobiSurf	• Touch interaction: To initiate information sharing, the device is physically touched with the receiving device's screen • Touch gestures: For interacting with the displayed content • Proxemic interaction: Holding the devices close together allows for information to be shared using NFC	• MobiSurf provides an environment where a user can seamlessly switch between individual and group work • MobiSurf allows easy sharing of information between devices using touch interaction	• MobiSurf only allows for the collaboration of two users and their mobile devices with a shared interactive surface • To transfer information from a mobile device to the shared surface, the device has to touch the surface
Gesture On	• Touch gestures: Drawing specific gestures to trigger a specific system response	• Gesture On is functional even when the mobile device is in standby mode • Touch gesture eliminates the additional steps required when performing a specific action	• Gesture On only uses touch gestures, where the possibility of in-air gestures can be introduced • Gesture On struggles to distinguish between symbols which are similar in shape
AirLink	• In-air gestures: Information is shared when a user waves his hand from one device to another	• Sharing information using in-air gestures • AirLink only requires the built-in microphone and speakers on mobile devices (no additional hardware required)	• AirLink only functional when devices are placed linearly • Code words assigned to each device are hardcoded • AirLink code words become more complex when more devices are introduced
Flick	• Touch gestures: Flicking the information towards the top of the screen causes the information to be shared	• Flick operates on multiple platforms • Information is shared by performing a flick gesture	• The flick gesture performed in Flick is directed towards the top of the screen and not necessarily the direction in which the receiving device is located • Flick only allows for single file sharing (no support for multiple file sharing)
Zapya	• Touch interaction: Long press and holding the selected file(s) to share and flicking your wrist while holding the device • Touch gesture: Long press and dragging the file to the receiving avatar	• Zapya shares information to all users in group with a flick gesture • Single user sharing is performed with a long press and drag action	• Zapya does not support multiple selective user sharing • Group admin unable to kick and/or prevent unauthorised users from joining

3.7 Conclusion

This chapter addressed the second activity, ***Outline Artefact and Define Requirements***, within the Relevance Cycle of the DSR methodology (Section 1.7.3.1). This chapter conducted a literature study to answer the second research question (Section 1.7.1):

> *"RQ_2. What are the benefits and shortcomings of existing NUI interaction techniques for information sharing?"*

Literature shows NUI to be the next phase in the evolution of HCI. An investigation into the NUI definition yielded the fact there is no universally accepted definition. The understanding and definition of a NUI is crucial in the context of this research. Through the investigation of different definitions of NUIs, different objectives of NUIs were identified.

NUIs are typically represented using various interaction techniques, which include touch, gestures, speech, and proxemics. The different interaction techniques were discussed which showed how NUIs could provide a more natural interaction that is more intuitive and interactive than previous interfaces.

An analysis of existing systems showed that the NUI interaction techniques are present in mobile computing and predominantly represented by touch interaction. The second interaction technique identified which is appropriate for information sharing among co-located mobile devices is gesture-based interaction. The lack of accurate positioning information was evident in the existing systems, which led to the interactions performed in existing systems being inappropriate and somewhat deceptive. For example, when a user performed a flick gesture towards the top of the device screen, it was expected it to mean that the receiving device was located somewhere in front of the sending device. However, this was not the case as the receiving device could be located to the left, right, or behind the user sending the information.

This chapter accomplished the second research objective (Section 1.7.2):

> "RO_2. *To identify the benefits and shortcomings of existing NUI interaction techniques for information sharing."*

The following chapter addresses the design and implementation of MotionShare. The chapter also discusses the development methodology for MotionShare that is used in the next DSR activity and DSR cycle.

Chapter 4: Positioning and Communication Design

4.1 Introduction

The previous two chapters discussed the literature studies on the fields of information sharing (Chapter 2) and NUIs (Chapter 3). These literature study chapters formed the theoretical foundation for the design and implementation of various prototypes, which focused on different aspects of the MotionShare. In Chapter 2, shortcomings of existing information sharing methods used by mobile devices were identified and the functional requirements of the information sharing process were discussed. In Chapter 3, important aspects of NUIs and the benefits they provide for addressing the shortcomings of existing information sharing methods were identified. Existing NUI interaction techniques in related domains were identified and reviewed to determine their feasibility in addressing the second research question (Section 1.7.1).

This chapter addresses the third activity in the DSR methodology (Section 1.7.3.1), namely ***Design and Develop Artefact***, which is performed within the Design Cycle (Figure 4.1). This chapter addresses the third research question (Section 1.7.1) identified:

> *"RQ_3. How should the relative pose for co-located mobile devices be calculated?"*

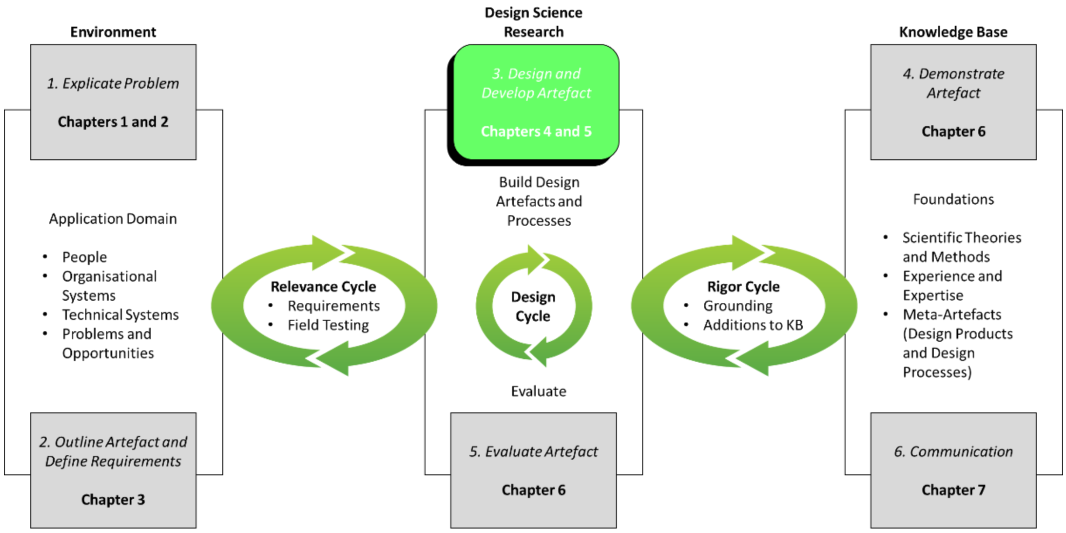


Figure 4.1: Chapter 4 Position in the Adapted DSR Methodology

This chapter documents the design and implementation of various prototypes in the development methodology section (Section 4.2). Each prototype demonstrates a different aspect of functionality to be used in the final design artefact named MotionShare. MotionShare is a mobile application which uses a proxemic NUI designed to support information sharing among co-located mobile devices. The implementation tools used are discussed in Section 4.6. Figure 4.2 shows the structure of this chapter.

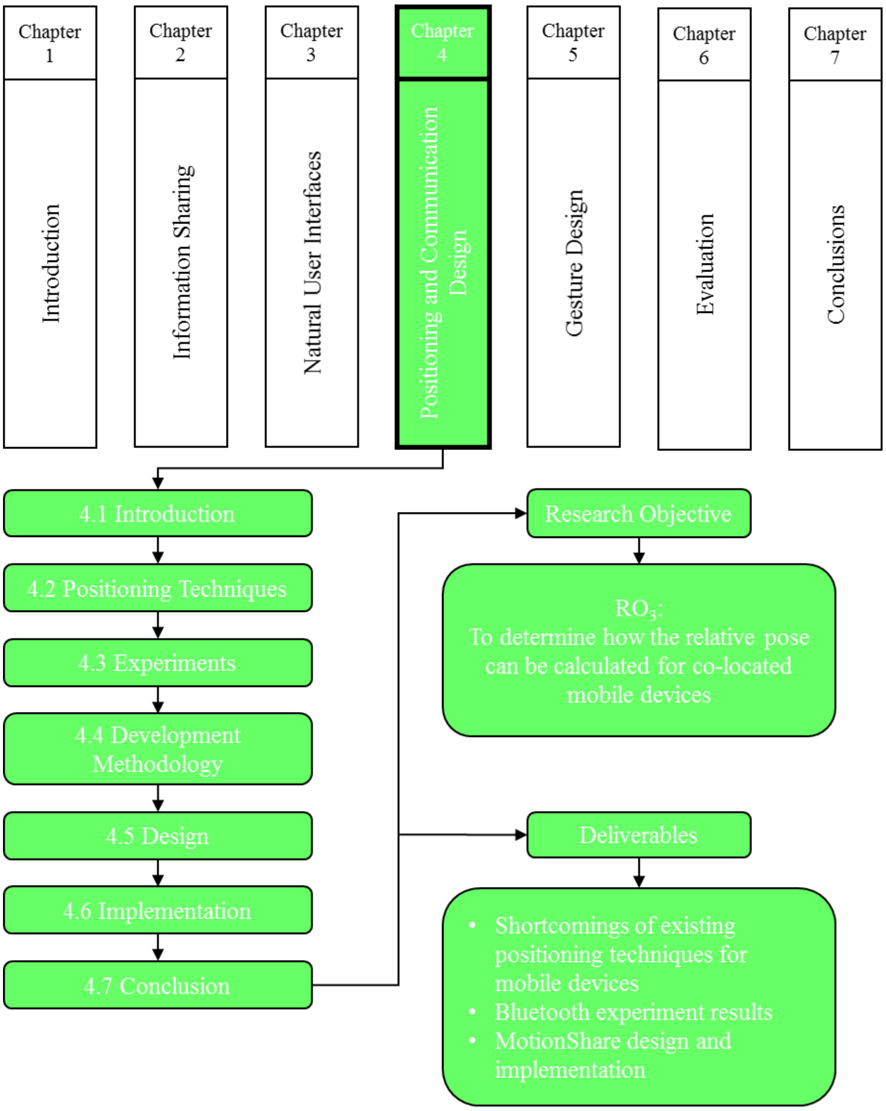


Figure 4.2: Chapter 4 Structure

4.2 Positioning Techniques

For NUI interaction techniques to be used in information sharing among co-located mobile devices, position and orientation information of each device relative to one another is required. Thus, positioning techniques are crucial in providing enhanced multi-device communication and supporting NUI interactions.

Several existing techniques to determine the position of a mobile device were identified (Graf, 2012). These techniques are global positioning systems (GPS), Wi-Fi positioning, cell tower triangulation, infrared, ultrasound, Bluetooth, and even radio-frequency identification (RFID) tags. Figure 4.3 provides an overview of these positioning techniques (Linnhoff-Popien, Marcus, & Küpper, 2010; Linnhoff-Popien, Marcus, & Schönfeld, 2015). Most of the techniques identified are limited in communication range, resource intensive, and applicable to mobile devices.

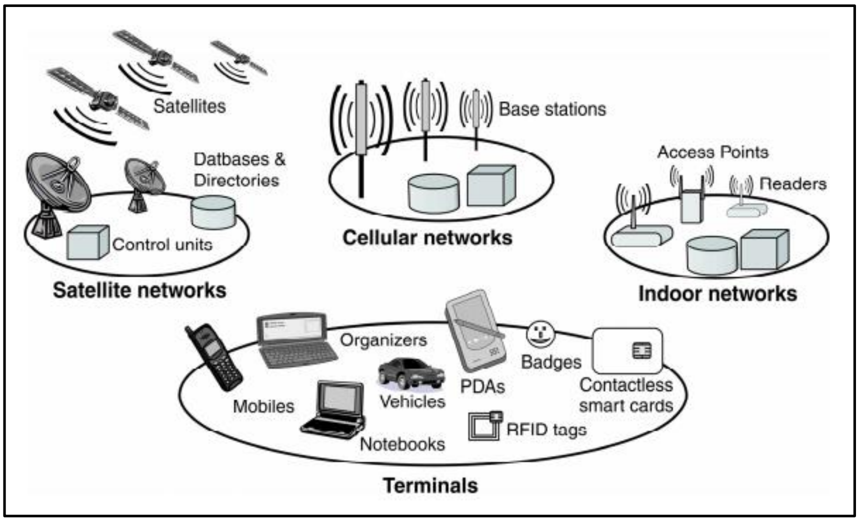


Figure 4.3: An Overview of Existing Positioning Techniques (Linnhoff-Popien et al., 2010, 2015)

Smartphones can use different techniques to locate their own position. Hightower and Borriello (2001) described several criteria to measure the techniques presented in Table 4.1.

Table 4.1: Criteria to Measure Positioning Techniques (Hightower & Borriello, 2001)

Criteria	Description
Accuracy	The difference between the true position and the calculated position.
Precision	The closeness of a number of position values to their mean value.
Power Consumption	The amount of electrical power required to determine the position measured in Watts per second.
Latency	The time needed to obtain the device position. Generally known as TTFF (time to first fix) measured in seconds.
Availability	Not all positioning techniques are available in every situation.

Three of the most common positioning techniques, namely GPS, Cell Tower Triangulation, and Wi-Fi Positioning are now discussed.

4.2.1 Global Positioning System

Most of the existing mobile devices rely on GPS as the standard technique to compute their geographical location in real-time (Paek, Kim, & Govindan, 2010). GPS is renowned as a worldwide satellite-based navigation system providing location and time information, irrespective of weather conditions, to almost anywhere in the world.

GPS on smartphones has a 95.00% success rate in calculating their location to within a distance of 5-15 metres (Zandbergen, 2014). GPS uses up to 32 satellites to determine the position of a mobile station (Huber, 2011). These satellites transmit two types of signals, namely Precise Positioning Service (PPS) and Standard Positioning Service (SPS). PPS is restricted to the United States military because it is encrypted (Graf, 2012). SPS is freely available for public usage and accessed using a GPS receiver.

One of the major disadvantages of using GPS is that it is limited by indoor environments and city buildings. This is because GPS receivers cannot receive the required line of sight to the satellite when indoors due to the fact that the line of sight to satellites is essential because it is broken by obstructions, such as walls.

4.2.2 Cell Tower Triangulation

Other positioning techniques such as Wi-Fi based location and cell tower triangulation are, however, receiving more interest because they consume less energy than GPS and are available in areas where GPS is not. The second positioning technique is cell tower triangulation. Telecommunication companies operate cellular networks and provide almost worldwide coverage. Every network is comprised of fixed cell towers, called base transceiver stations (Oriyano & Doherty, 2015). Cellular networks are divided into various cells, which have a different radius coverage depending on the number of cells in the network. As a user moves through the various cells, his mobile device will automatically connect to the nearest base station with the best signal strength available.

The cell identification (cell ID) refers to the use of a particular base station's location, which is used in cell tower triangulation. The accuracy of this technique depends on the cell size covered by the cell tower. Cell sizes can be as large as 35 kilometres in radius in 2G networks and slightly less in 3G networks due to technical constraints. To obtain a more accurate reading on the device location, another technique known as timing advance is used (eTutorials.org, 2015). Timing advance measures the time taken for signals to travel from the mobile device to the cell tower. Thus, the device can be located in a circle around the cell tower. The accuracy of the position of the device is improved significantly with the introduction of multiple cell towers. In Figure 4.4, the intersection of circles covered by each cell tower reduces the possible location area of the mobile device (Neilson, 2013). Device location is represented by a crosshair symbol.

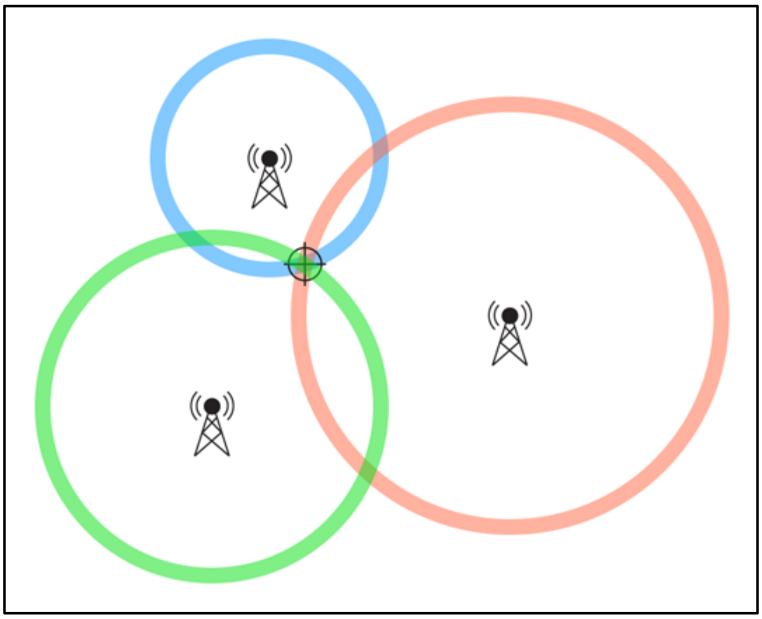


Figure 4.4: Overview of Cell Tower Triangulation (Neilson, 2013)

When the mobile device is within range of at least three cell towers, algorithms that are more complex can be used to better determine device position, which utilise the Time Difference of Arrival (TDOA) or Angle of Arrival (AOA). Küpper (2005), Wang et al. (2008), and Arigela et al. (2013) provide a detailed discussion of these techniques. Triangulation is measured in metres with a precision of between 50-150 metres (Hightower, 2011). Research into cell tower triangulation to improve the accuracy is being conducted in various parts of the world (Ahern, Davis, King, Naaman, & Nair, 2006; Yang, Varshavsky, Liu, Chen, & Gruteser, 2010).

4.2.3 Wi-Fi Positioning

Various technologies have been developed to address the issue of indoor positioning not being supported by GPS. Wi-Fi positioning is comparable to cell tower triangulation. The cell ID algorithm in cell tower triangulation is also used in the Wi-Fi Positioning technique. Instead of cell towers, the location information of each Wi-Fi Access Point (AP) is stored in a database and is used in determining the position of the mobile device. This information is gathered through wardriving with crowdsourcing by companies such as Google.

Wardriving is defined as the act of searching for Wi-Fi networks by an individual in a moving vehicle, using a mobile device (Mitchell, 2015). *Crowdsourcing* is defined as the process of obtaining needed services, ideas, or content (usually online) from a large group of individuals (Crowdsourcing LLC, 2015). Typically, Wi-Fi networks only have an average range of 60 metres, which results in this method providing a precision within this range.

Cities around the world have increased the number of APs located to the point where a mobile device is almost always in range of a Wi-Fi network. This allows for the use of different techniques, such as cell of origin (COO), to determine the device's position. Wi-Fi positioning allows a user to obtain a position fix of their location by sending a request, in the form of a web service, to the database with all the location information of the APs. Therefore, the shortcoming of Wi-Fi positioning is that it is only functional with the location of multiple APs in the immediate area.

4.2.4 Summary

In Section 1.1, all of the existing positioning techniques discussed have different benefits and shortcomings. GPS is highly accurate (95.00%) and positioning is within metres, but it is weak in energy consumption and is unable to perform within indoor environments. Although cell tower triangulation is energy efficient and available almost everywhere in the world, it still lacks the accuracy and precision provided by GPS (accuracy within metres). Wi-Fi positioning is best suited for an indoor environment; however, it requires multiple APs to be located nearby to enable it to work, and is only able to provide a device position to within an accuracy of 60 metres. Table 4.2 compares the positioning techniques using the criteria identified (Table 4.1).

Table 4.2: Comparison of Positioning Techniques

Positioning Technique	Accuracy	Precision	Power Consumption	Latency	Availability
GPS	95.00%	5-15 metres	High	Unknown	Does not work indoors
Cell Tower Triangulation	Lower than GPS and depends on cell tower coverage	50-150 metres	Low	Unknown	Almost everywhere with cell towers
Wi-Fi Positioning	Unknown	+-60 metres	Low	Unknown	Limited to Wi-Fi AP coverage

All the discussed techniques provided coarse-grained granularity, and ideally for this research, a more fine-grained technique is required. Therefore, experiments were conducted to determine if a more fine-grained solution is available. The next section discusses these experiments and their results.

4.3 Experiments

Experiments were conducted involving the use of Bluetooth technology for mobile devices. The experiments used the following smartphone models:

- 2 x Samsung Galaxy S III;
- 2 x Samsung Galaxy S4; and
- 2 x Samsung Galaxy S5.

The primary objective of these experiments was to determine whether the Bluetooth RSSI can be used as an accurate indicator of distance among mobile devices. The next section discusses hypotheses formulated to achieve the primary objective.

4.3.1 Hypotheses

Table 4.3 presents the objectives of these experiments that address the different hypotheses.

Table 4.3: Hypotheses for the Different Experiments

No.	Hypothesis
$H_{1,0}$	If the same device model is used, then the RSSI values will be same
$H_{1,1}$	If the same device model is used, then the RSSI values will be different
$H_{2,0}$	Changing the orientation of the mobile device will not influence the RSSI values
$H_{2,1}$	Changing the orientation of the mobile device will influence the RSSI values
$H_{3,0}$	Changes in battery level of the device will not influence the RSSI values
$H_{3,1}$	Changes in battery level of the device will influence the RSSI values
$H_{4,0}$	If devices are placed in different environments , then the RSSI values are not different
$H_{4,1}$	If devices are placed in different environments , then the RSSI values are different

The first hypothesis states that the use of the same device would result in the same RSSI values being displayed. If the results of this experiment rejected this hypothesis, it would give better insight and understanding into what factors influence Bluetooth RSSI. Each experiment conducted followed the same procedure (Section 4.3.2) and addressed one of the hypotheses presented in Table 4.3.

The first experiment used two Galaxy S III smartphones. Similarly, Galaxy S4 and S5 smartphones were subjected to the same process. The second experiment used the two Galaxy S4 smartphones, changing the orientations of the devices for every possible combination. To establish if a difference existed using different models, the process was repeated with the two Galaxy S III and the two Galaxy S5 smartphones.

The third experiment was similar to the first experiment with regard to the same device models used, but the difference was the battery levels between the devices. This was to address the third hypothesis, to determine if the battery level of a device influenced the RSSI value. Lastly, the fourth experiment used the Galaxy S5 smartphones and performed the procedure in different environments. Only Galaxy S5 smartphones were used and all other variables were also kept constant. This allowed for the effects of the different environments to be compared to determine whether the different environments would yield different RSSI values of the devices.

4.3.2 Procedure

Each experiment only utilised two smartphone devices at a time and included the Bluetooth scanning of these devices, which were placed at different distance increments. The distance increments commenced at 25cm and went up to 200cm. These increments were considered to be a fair representation of the distances that users would be seated or standing apart from each other in a co-located environment. It was decided that the experiments would be limited to only two metres for a small testing environment despite the fact that Bluetooth is well-known to have a range of up to 30 metres. The prototype Bluetooth RSSI was developed, installed, and used on each device to determine the RSSI values (Figure 4.5). The RSSI values displayed on each device were recorded by means of data logging and observation. Each experiment was repeated several times to ensure a sufficient data sample could be obtained to develop a more accurate model for the selected machine learning (ML) algorithm.

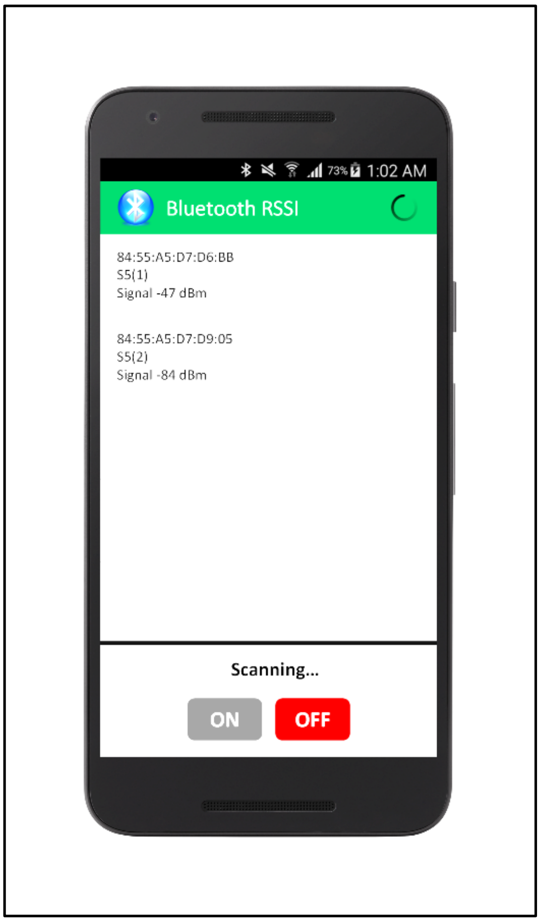


Figure 4.5: Screenshot of the Prototype Bluetooth RSSI

4.3.3 Results

The results generated from the experiments address the hypotheses that were identified (Table 4.3). The statistical analysis of these results and the visualisations thereof are presented.

4.3.3.1 Data Analysis

The data gathered from the various experiments conducted were aggregated according to the hypothesis, which the particular experiment addressed. The raw data was obtained from data logging and observations served as a checking mechanism to ensure data integrity and accuracy.

Several experiments were conducted on the collected data. Table 4.4 shows the performance of various ML algorithms in classifying distance based on the Bluetooth RSSI.

Table 4.4: Classification of Distances for Mobile Devices

ML Algorithms	Overall Classified (%)
IB1	81.25
IBk	80.25
LMT	77.42
KStar	75.25
J48	72.17
J48graft	69.17
Multilayer Perceptron	63.50
Bayesian Network	57.25
Naïve Bayes	56.42
SMO	52.73

The IB1 classifier had the highest accuracy in correctly classifying the instances. Therefore, the ML algorithm used on the results of the Bluetooth experiment was the IB1 classifier. The IB1 algorithm (Devasena, 2013) is an instance-based nearest neighbour classifier. It uses normalised Euclidean distance to determine the training instance closest to the given test instance, and predicts the same class as this training instance. The 10-fold cross-validation was used with the IB1. This meant that the dataset was split into 10 equal parts (folds). Using the 10-fold cross-validation meant that 90% of the dataset was used for the training (and 10% for testing) in each fold test.

Confusion matrices (Data School, 2014) are used in ML to visualise the performance of a specific algorithm. Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. The value at each intersection between a column and row represents the number of predictions classified. The ideal scenario is to have the value only appear in the *"diagonal"*.

In Table 4.5, the distance was successfully classified as 25cm (100%). IB1 correctly classified 50cm (92%) and misclassified it as 25cm (4%). The 75cm distance was correctly classified 92% of the time, with a misclassification of 3% as 200cm. The 100cm distance was correctly classified 67% of the time, with 23% misclassified as

200cm. IB1 struggled to classify 125cm with only 50% accuracy. IB1 incorrectly classified 125cm as 150cm (14%) and 175cm (30%). This classifier had no issues with 150cm (90%) and only misclassified 4% as 175cm. The 175cm distance was correctly classified 66% of the time while 12% misclassified as 125cm. IB1 correctly classified 200cm (76%) and misclassified it as 100cm (12%). Overall, the IB1 algorithm correctly classified instances with an accuracy of 81.25%, which is an acceptable rate.

Table 4.5: Confusion Matrix for the IB1 Classifier (%)

	a	b	c	d	e	f	g	h	Classification
a	**100.00**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**a = 25cm**
b	4.00	**92.00**	2.00	2.00	0.00	0.00	0.00	0.00	**b = 50cm**
c	0.00	2.00	**92.00**	2.00	1.00	0.00	0.00	3.00	**c = 75cm**
d	0.00	0.00	3.00	**67.00**	5.00	2.00	0.00	23.00	**d = 100cm**
e	0.00	0.00	2.00	4.00	**50.00**	14.00	30.00	0.00	**e = 125cm**
f	0.00	0.00	0.00	2.00	3.00	**90.00**	4.00	1.00	**f = 150cm**
g	0.00	0.00	0.00	7.00	12.00	5.00	**66.00**	10.00	**g = 175cm**
h	0.00	0.00	4.00	12.00	0.00	4.00	4.00	**76.00**	**h = 200cm**

4.3.3.2 Statistical Analysis

The aggregated results were tabulated according to device used, distance increment, RSSI value, and the corresponding device with its RSSI value. Tabulation of the data allowed for the information to be efficiently and effectively examined in providing overall comparisons between the variables. Various statistical analyses were applied to the results, as shown in Table 4.6. The analysed data was used to create different data visualisations, which are now discussed and shown in the next section.

Table 4.6: Statistical Analysis to Various Experiment Results in dBm (n = 90)

No.	Experiment	Hypothesis No.	Mean	Median	Mode	SD	Min	Max
1	Galaxy S III	1	-73	-77	-80	13,94	-42	-96
2	Galaxy S4	1	-75	-79	-73	14,02	-45	-96
3	Galaxy S5	1	-75	-75	-76	12,73	-41	-100
4	Orientation (↑, ↑)	2	-73	-77	-80	13,94	-42	-96
5	Orientation (↑, →)	2	-73	-75	-66	12,03	-47	-98
6	Orientation (↑, ↓)	2	-73	-75	-73	13,25	-44	-97
7	Orientation (↑, ←)	2	-75	-77	-86	12,10	-50	-93
8	Orientation (→, →)	2	-74	-76	-76	11,24	-51	-94
9	Orientation (→, ↓)	2	-73	-76	-79	12,55	-42	-98
10	Orientation (→, ←)	2	-74	-75	-74	11,30	-52	-100
11	Orientation (←, →)	2	-74	-79	-80	12,20	-45	-91
12	Battery Level (100%)	3	-73	-77	-80	13,94	-42	-96
13	Battery Level (75%)	3	-74	-77	-79	15,02	-42	-100
14	Battery Level (50%)	3	-75	-78	-65	15,38	-41	-100
15	Battery Level (25%)	3	-74	-76	-83	15,07	-43	-104
16	Environment 1	4	-60	-61	-73	11,79	-35	-76
17	Environment 2	4	-67	-68	-64	8,76	-47	-83
18	Environment 3	4	-73	-77	-80	13,94	-42	-96

4.3.3.3 Graphical Representation

Data visualisations were generated for the results obtained from the experiments conducted. These visualisations represent the behaviour of Bluetooth RSSI with different phone models (Figure 4.6), at different battery levels (Figure 4.7), and in different environments (Figure 4.8). Graphical representation of these results allows for the findings to be used quickly. For example, a quick comparison to determine the most optimal environment.

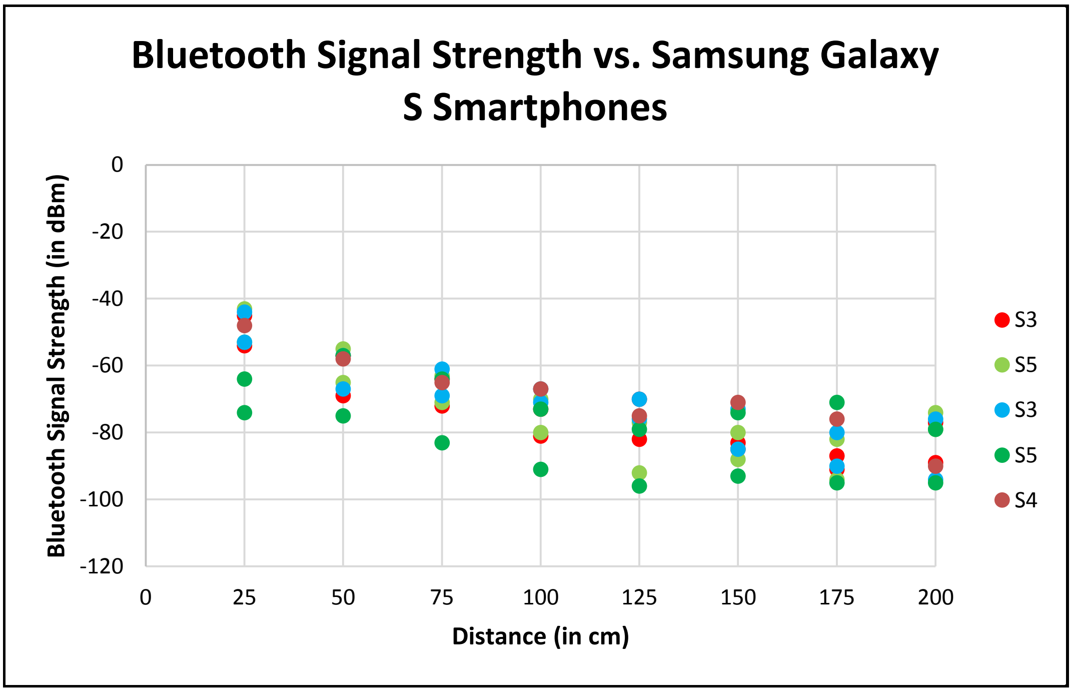


Figure 4.6: Bluetooth RSSI with different Samsung Galaxy S Models

In Figure 4.6, the null hypothesis ($H_{1,0}$) was rejected because the same device models, for example S3 did not have the same RSSI values, and therefore the alternative first hypothesis was accepted ($H_{1,1}$). Similarly, the null hypothesis ($H_{2,0}$) was rejected as the RSSI values did differ slightly (informal t-tests performed resulted in no significant difference) and the alternative second hypothesis was accepted ($H_{2,1}$).

The null hypothesis ($H_{3,0}$)was rejected as changes in battery levels showed the RSSI values were different (Figure 4.7). Therefore, the alternative third hypothesis was accepted ($H_{3,1}$). Battery levels were considered an influencing factor in the design of MotionShare because the t-tests performed revealed no significant difference.

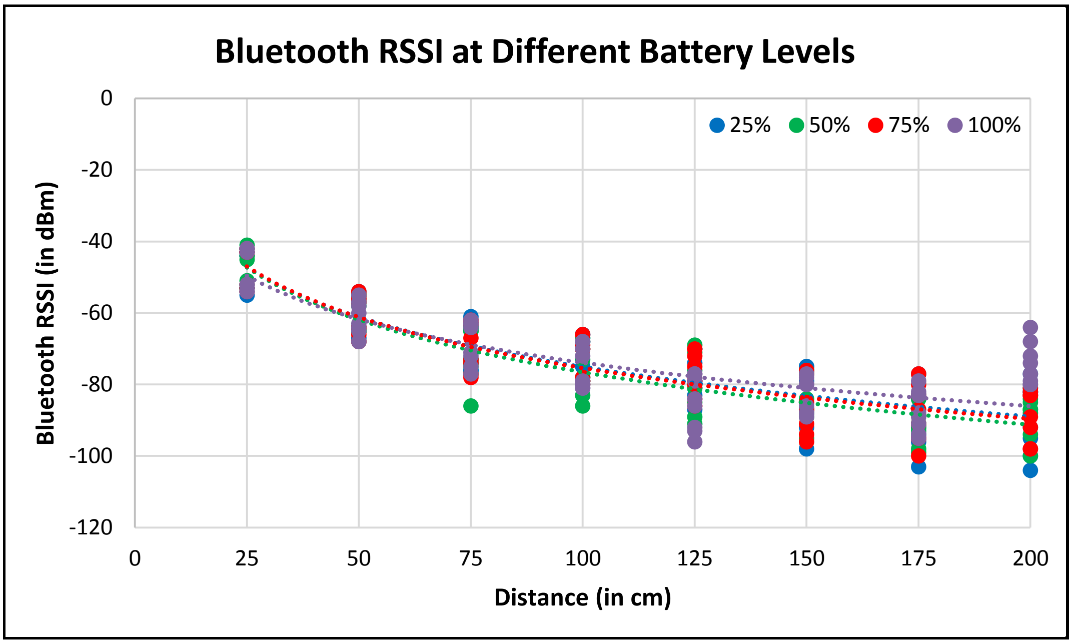


Figure 4.7: Bluetooth RSSI at Different Battery Levels

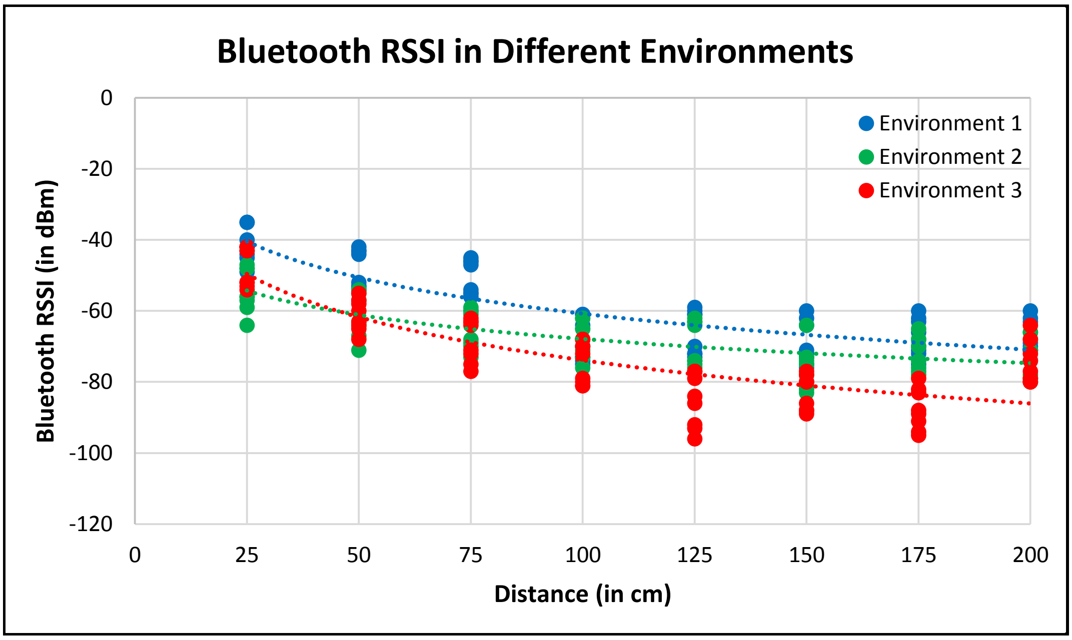


Figure 4.8: Bluetooth RSSI in Different Environments

In Figure 4.8, different environments affected the RSSI values of the mobile devices. The null hypothesis ($H_{4,0}$) was rejected and the alternative fourth hypothesis was accepted ($H_{4,1}$). The t-tests performed showed the different environments to be significantly different. Therefore, the environment least adversely affecting the RSSI values was selected when evaluating MotionShare.

All the results from the experiments were aggregated into a single visualisation to illustrate the relationship between Bluetooth RSSI and distance. The experiments showed an inverse relationship exists between these two variables (Figure 4.9).

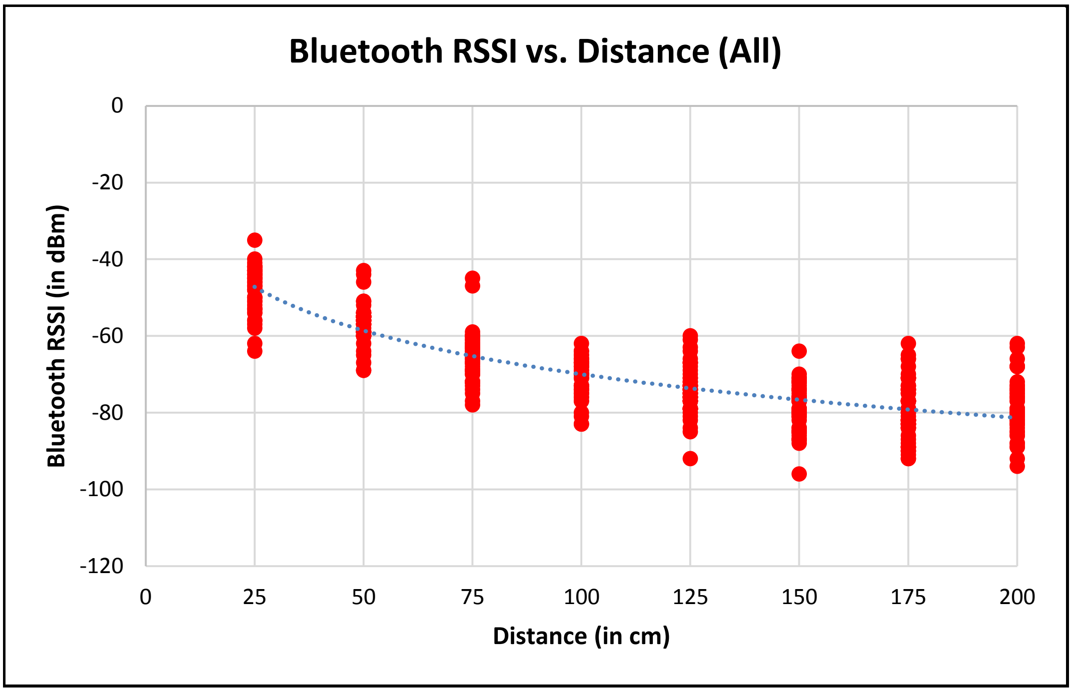


Figure 4.9: Bluetooth RSSI vs. Distance Relationship

4.3.4 Conclusions

Through the experiments conducted, the identified hypotheses (Table 4.3) were addressed. The null hypothesis was rejected because the use of the same device models did not display the same RSSI values at different distance increments (Figure 4.6) and the alternative first hypothesis was accepted. As hypothesised, changing the orientation of the mobile device increased the RSSI values, which validated hypothesis two. The effect was however minimal, which is evident from the statistical analysis performed (Table 4.6) and graphical representation (Figure 4.9). There was better performance in RSSI values when the devices were fully charged, but similarly to the orientation experiments, the values were hardly noticeable (Figure 4.7). In Figure 4.7, the trend line of the 100% battery level is barely visible above the others because the RSSI values are so closely clustered together. Despite the fact that the difference is marginal, the third alternative hypothesis was accepted and the null hypothesis rejected. The last hypothesis regarding the RSSI values being different in other environments was shown (Figure 4.8). The different environments were shown to be significantly different from the t-tests performed. Bluetooth RSSI values of mobile devices were the best in environment 1. In conclusion, these RSSI values provide more fine-grained positioning of devices than existing techniques.

4.4 Development Methodology

A prototype is a rudimentary working model of a product, typically built for demonstration, and forms part of the development methodology (Butter, Quintana, & Valenzuela, 2014). A basic version of the design artefact is built, tested, and then reworked until it is deemed acceptable. Incremental prototyping is selected to iteratively build the artefact, named MotionShare. *Incremental prototyping* is defined as *"the building of the design artefact by building individual prototypes"* (Mirchandani, 2000). Towards the end of this process, these prototypes are merged into an overall design. There are several benefits of using incremental prototyping (Higher National Computing, 2010; Kumar, 2015):

- The software component is generated quickly and early;
- Prototypes support the identifying and addressing of issues at the initial stages;
- Ambiguity is eliminated by improving the understanding of the functional requirements of the design artefact;
- Prototyping ensures the design artefact does what it is supposed to – not what the developer thinks it ought to do; and
- It is easier to test and debug during a smaller iteration.

MotionShare is the design artefact comprising of several prototypes, each of which focuses on a different aspect of functionality. Figure 4.10 depicts the iterative nature of the development (Guida, Lamperti, & Zanella, 2013), starting with the first iteration of ProtoFile addressing the first functional requirement (Section 2.7.5). The second iteration includes the ProtoFile and adds to the next functional requirement in ProtoWiFiAP. In the N^{th} iteration, the design artefact is complete and ready. Hence, the iterative process in the development of MotionShare is in accordance with the activity, ***Design and Develop Artefact***, and the Design Cycle found within the DSR methodology (Section 1.7.3.1).

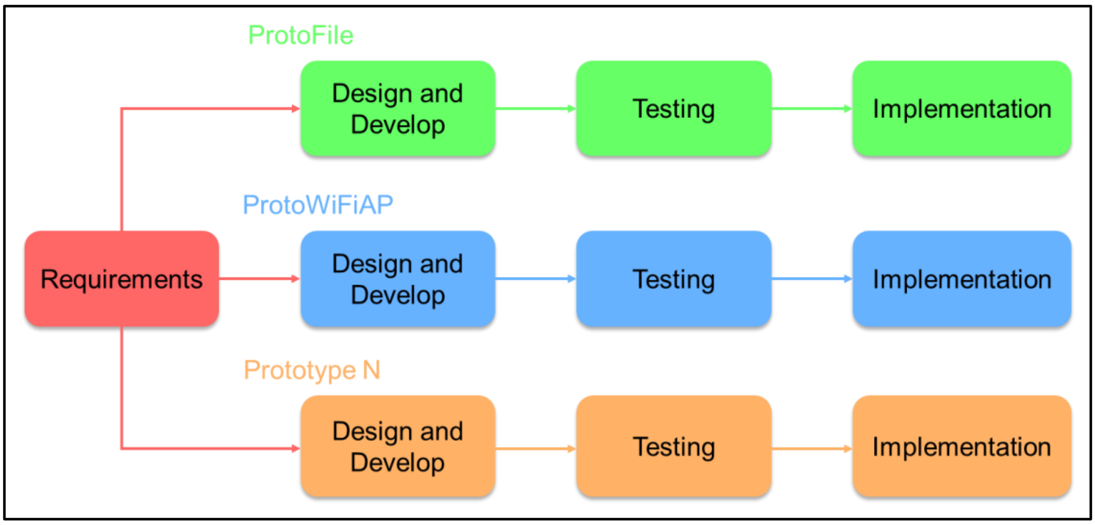


Figure 4.10: Incremental Prototyping Model (Guida et al., 2013)

4.5 Design

In order to design MotionShare, the functional requirements had to be identified and understood. The information sharing process (Section 2.7.5) formed the basis of the expected functionalities of MotionShare. These functionalities are as follows:

- Selecting the file(s) to be shared;
- Selecting the recipient(s);
- Initiating the transfer;
- Notification on whether to accept or reject the transfer;
- Receiving acknowledgment that the transfer is complete; and
- Cancel the transfer.

The design of MotionShare is described to support the information sharing process (Figure 2.12). The next section discusses the prototypes which address the expected MotionShare functionalities.

4.5.1 Prototypes

MotionShare comprises of the following prototypes:

- ***ProtoFile*** retrieves the names and locations of all files stored on a mobile device and classifies them according to the different information types typically shared among mobile devices (Section 2.3).
- ***ProtoWiFiAP*** focuses on the development of a private and secure Wi-Fi hotspot to support information sharing among co-located mobile devices.
- ***ProtoBluetooth*** involves the enabling of the Bluetooth communication technology and scanning of devices in the immediate area. The results of the scan are processed and only the relevant information to be utilised later is displayed.
- ***ProtoCompass*** provides the mobile device with an orientation value in relation to the earth's magnetic field. As a result, the device always knows the direction of magnetic north.
- ***ProtoMap*** focuses on processing the information captured from the initial calibration of the mobile devices. Upon completion of the calculations, a map of all the devices is displayed.
- ***ProtoGesture*** involves the implementation of several existing gestures obtained from literature and the focus groups.

Figure 4.11 presents the integration of the discussed prototypes into the design artefact named MotionShare.

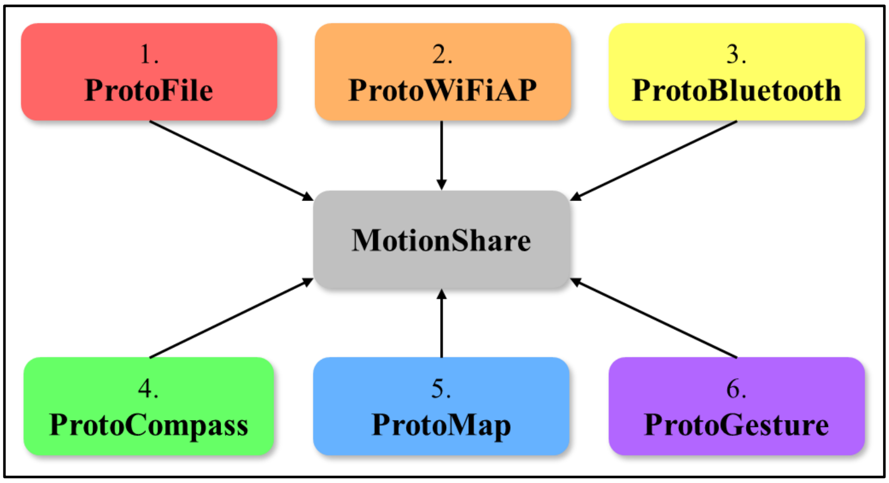


Figure 4.11: Integration of Prototypes into MotionShare

4.5.2 Data

MotionShare needs to capture and store the data of every co-located mobile device in the environment using a *CustomDevice* class. The *CustomDevice* class stores data used to help identify devices in the environment such as device name, Bluetooth RSSI, approximate distance, server bearing, client bearing, and colour. The architecture design of MotionShare needs to support simultaneous information sharing between devices. Furthermore, the information shared among and received among mobile devices are stored locally on the devices.

4.5.3 Architecture

MotionShare is an Android application, which needs to support information sharing among co-located devices, and therefore a client-server architecture is required. Typically, this type of architecture on an Android platform is a network architecture involving mobile devices operating as clients and a desktop computer operating as a server (Baotić, 2014). The server facilitates the communication between the various clients and itself. This communication occurs by means of an active Internet connection or over Wi-Fi (Figure 4.12).

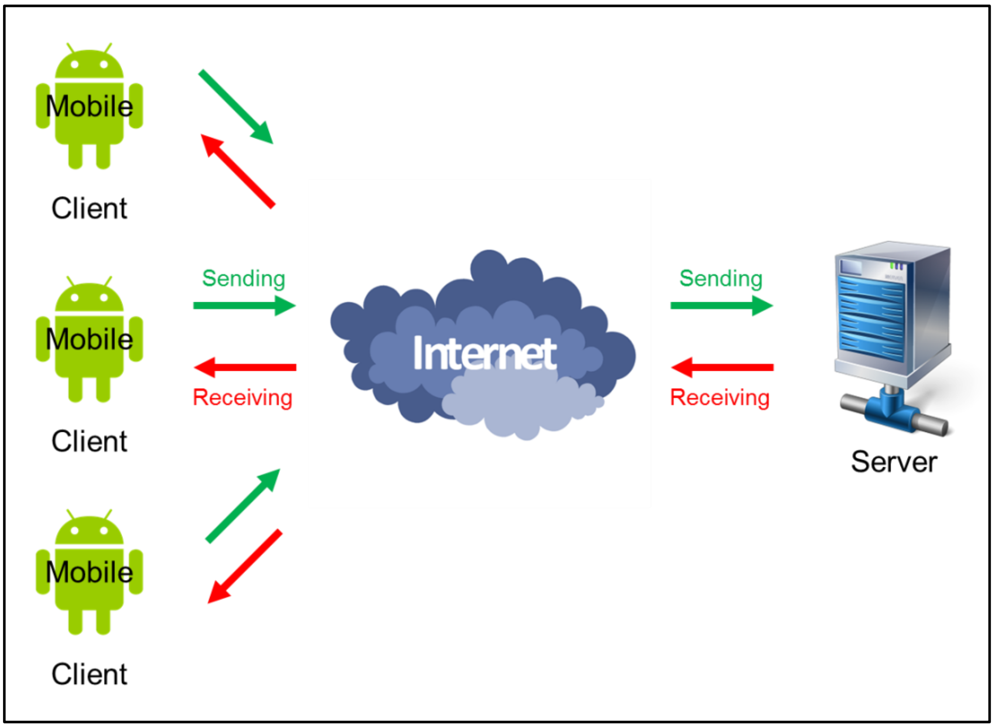


Figure 4.12: A Mobile Client-Server Model via the Internet Adapted from Baotić (2014)

The constant communication between the mobile clients and the server can result in potentially costly mobile data bandwidth costs. In order to reduce these costs, the traditional client-server architecture was modified. The desktop computer operating as a server was replaced by another mobile device. The decision to replace the computer, enabled any mobile device to become the server or a client.

Figure 4.13 illustrates the proposed client-server model. Devices A, C, and D are clients and device B is the server. The communication between these devices are supported by the private and secure Wi-Fi hotspot created by the server device. The hotspot is protected by WPA2-PSK (AES), which is the most secure wireless encryption option currently available. WPA2-PSK (AES) is the Wi-Fi Protected Access 2 Pre-Shared Key Advanced Encryption Standard (Juniper Networks, 2015). Typically, the PSK is the encrypted password for the Wi-Fi network. WPA2-PSK uses WPA2, the latest available Wi-Fi encryption standard, and the latest AES encryption protocol (Hoffman, 2014). This architecture design supports information sharing among co-located mobile devices without consuming mobile data bandwidth. All the devices in this proposed design are able to send and receive information.

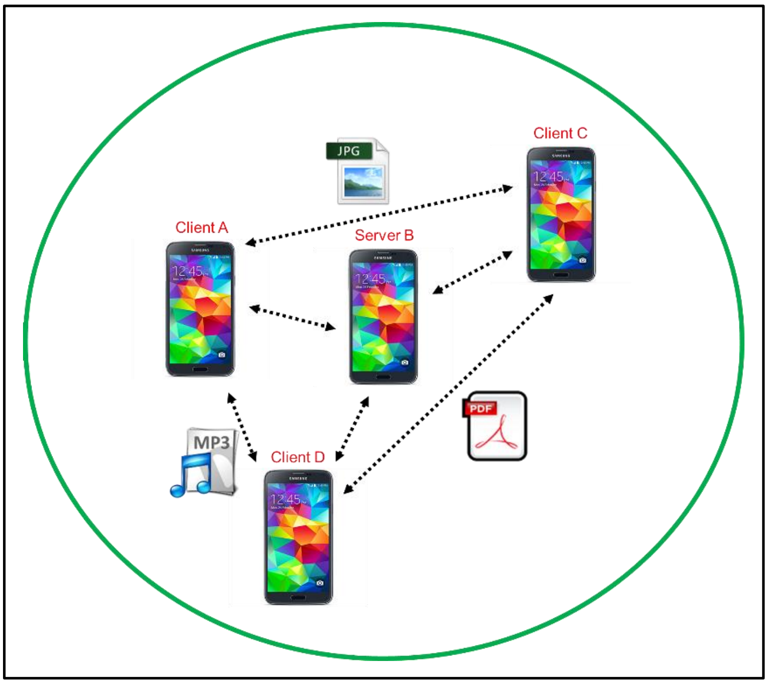


Figure 4.13: The Proposed Mobile Client-Server Model

4.6 Implementation

The design of MotionShare was outlined in Section 4.5. The existing tools involved in the implementation of the prototypes and MotionShare are discussed (Section 4.6.1). The development of the prototypes and how they were integrated to form MotionShare are also discussed (Section 4.6.2).

4.6.1 Implementation Tools

The environment required for this research consists of hardware and software components. The hardware component required mobile multi-touch devices with the following requirements:

- Advanced computing processing capabilities;
- Data storage capabilities similar to a desktop computer;
- Embedded sensors within the device;
- Bluetooth capabilities; and
- Wi-Fi connectivity.

Therefore, smartphone devices were selected for this research based on these requirements discussed above. The software component required a platform to support mobile design and development. The resources available for a particular platform can significantly improve the code quality and reduce the time for designs to be iterated and tested. The number of active users on a particular platform was also considered. Furthermore, the platform should support the various NUI interactions used in this research.

4.6.1.1 Hardware

This research aims to design a proxemic NUI to provide an accurate and usable solution to support information sharing among co-located mobile devices. Therefore, several smartphones were involved in the design and implementation of MotionShare. These devices varied in model types, which were the Samsung Galaxy S III, Samsung Galaxy S4, and Samsung Galaxy S5 (Figure 4.14).

Figure 4.14: Samsung Galaxy S III, Samsung Galaxy S4, and Samsung Galaxy S5 Smartphones (Samsung, 2012, 2013, 2014)

The Samsung Galaxy S III is a multi-touch, slate-format smartphone designed, developed, and marketed by Samsung (2012). The Galaxy S III runs Android 4.3 "Jelly Bean" operating system and has a 4.8" (122mm) display screen with HD Super AMOLED (720 x 1280) resolution (Figure 4.14).

This device has various data inputs, some of which were used in the implementation of MotionShare. The data inputs available on the Galaxy S III are presented in Table 4.7.

Table 4.7: Available Data Inputs on the Galaxy S III (Samsung, 2012)

No.	Data Inputs	Used
1	Multi-Touch Capacitive	✓
2	Touchscreen	✓
3	3 Push Buttons	✓
4	Assisted Global Positioning System (A-GPS)	✗
5	GLONASS	✗
6	Barometer	✗
7	Gyroscope	✓
8	Accelerometer	✓
9	Magnetometer	✓

The Samsung Galaxy S4 is an Android smartphone produced by Samsung (2013), which runs Android 5.0.1 "Lollipop" operating system. This device is the successor to the Galaxy S III and has a similar design; however, the Galaxy S4 emphasises the software functionalities by taking advantage of its sophisticated hardware capabilities. The Galaxy S4's display is larger than its predecessor, with a 5" (127mm), 1080p PenTile RGBG Super AMOLED screen (Figure 4.14). Similarly, the data inputs available on the Galaxy S4 were used within MotionShare and tabulated in Table 4.8.

Table 4.8: Available Data Inputs on the Galaxy S4 (Samsung, 2013)

No.	Data Inputs	Used
1	Accelerometer	✓
2	Barometer	✗
3	Gesture Sensor	✓
4	GPS	✗
5	GLONASS	✗
6	Gyroscope	✓
7	Hall Effect Sensor	✗
8	Hygrometer	✗
9	Magnetometer	✓
10	Proximity Sensor	✗
11	RGB	✗
12	Light Sensor	✗
13	Thermometer	✗

The Samsung Galaxy S5 is also an Android smartphone produced by Samsung (2014) and runs Android 5.0.1 "Lollipop" operating system. This device provides a more refined user experience, additional security features, expanded health features, and an improved camera. The Galaxy S5 boosts a 5.1" (130mm) 1080p Super AMOLED panel, which is only slightly larger than the Galaxy S4, and allows for the automatic brightness and gamut adjustments (Figure 4.14). Data inputs available on the Galaxy S5 have improved from the previous versions (Table 4.9).

Table 4.9: Available Data Inputs on the Galaxy S5 (Samsung, 2014)

No.	Data Inputs	Used
1	Fingerprint Recognition	✗
2	Heart Rate Sensor	✗
3	Motion Coprocessor	✗
4	Accelerometer	✓
5	Gesture Sensor	✓
6	Gyroscope	✓
7	Proximity Sensor	✗
8	Magnetometer	✓
9	Barometer	✗
10	Hall Effect Sensor	✗
11	Magnetic Sensor	✗
12	RGB Ambient Light	✗
13	Infrared (IR) LED Sensor	✗

In Table 4.10, a comparison of the Samsung Galaxy S smartphones is presented. The computational capabilities of all three Galaxy S models allow for any device in the environment to become the server device and perform all the required computations during the initial setup.

Table 4.10: Specifications Comparison of the Samsung Galaxy S Smartphones

	Galaxy S5	Galaxy S4	Galaxy S III
Manufacturer	Samsung	Samsung	Samsung
Display	5.1" FHD Super AMOLED (1920x1080, 432ppi)	5" FHD Super AMOLED (1920x1080, 441ppi)	4.8" HD Super AMOLED (1280x720, 306ppi)
CPU	2.5GHz Quad-Core Krait 400	1.6GHz Quad-Core Cortex-A15	1.4Ghz Quad-Core Cortex-A9
RAM	2GB	2GB	1GB
Storage	16GB/32GB + MicroSD Slot	16GB/32GB/64GB + MicroSD Slot	16GB/32GB + MicroSD Slot
Camera	16MP Rear, 2MP Front	13MP Rear, 2MP Front	8MP Rear, 1.9MP Front
Battery	2800maH	2600maH	2100maH
Connectivity	Wi-Fi: 802.11 a/b/g/n/ac	Wi-Fi: 802.11 a/b/g/n/ac	Wi-Fi: 802.11 a/b/g/n
	Wi-Fi Direct	Wi-Fi Direct	Wi-Fi Direct
	Wi-Fi Hotspot	Wi-Fi Hotspot	Wi-Fi Hotspot
	GPS/GLONASS	GPS/GLONASS	GPS/GLONASS
	NFC	NFC	NFC
	Bluetooth 4.0 BLE	Bluetooth 4.0	Bluetooth 4.0
Dimensions	142.0 x 72.5 x 8.1mm	136.6 x 69.8 x 7.9mm	136.6 x 70.6 x 8.6mm
Weight	145g	130g	133g

4.6.1.2 Programming Language and Environment

Java is the computer programming language used in the development of MotionShare, which is described by Gosling et al. (2015) as:

> *"The Java® programming language is a general-purpose, concurrent, class-based, object-oriented language."*

Gosling et al. (2015) also describes Java to be a relatively simple language in which many programmers could potentially become proficient. The primary reason behind this statement is the surplus of Java resources available to support the learning process, including websites, tutorials, books, and classes. Another reason is that Java is currently one of the most human-readable languages that exists, which means an individual with zero programming knowledge or experience can often look at some Java code and have at least an idea of what the code is supposed to do (Conder & Darcey, 2010).

The decision to use three Galaxy S smartphone models (Section 4.6.1.1), namely S III, S4, and S5, directly resulted in the selection of Java. Android applications are typically developed in Java using the Android software development kit (SDK). The Android SDK provides all the application programming interface (API) libraries and tools required to build an Android application. Therefore, it is important to select an appropriate integrated development environment (IDE) in which to write and build applications. Several IDEs exist for Java Android development such as Eclipse, IntelliJ IDEA, and Android Studio, which are among the most popular ones used (Eye Internet Ltd., 2014).

Android Studio is the official IDE for Android application development, primarily based on IntelliJ IDEA (Google Inc., 2015a). Android Studio incorporates all the functionality of IntelliJ IDEA, as well as the following additional capabilities:

- Flexible Gradle-based build system combines the best features from other build systems that allows the developer to write his own script in Java;
- Build variants and multiple Android application package (APK) file generation;
- Code templates to help you build common application features;
- Rich layout editor with support for drag and drop theme editing; and
- Android Lint tool is static code analysis tool that is used to identify performance, usability, and version compatibility issues.

Android development has a substantial number of resources available such as forums in which to post questions, tutorials, and libraries to be utilised. The development of Android applications is more than just using Java. It requires an understanding of three areas:

- How the Android user interface (UI) is constructed;
- How to use extensible markup language (XML); and
- How to access the various Android sub-systems.

In Android Studio, the project files are displayed in the Android project view. This view shows a flattened version of a project's structure, which provides quick access to the key source files of the project and helps one to work with the Gradle-based build system of Android Studio. The characteristics of the Android project view (Google Inc., 2015a) are:

- Shows the most important source directories at the top level of the module hierarchy;
- The build files for all modules are combined into a common folder;
- Each module's manifest file is combined into a common folder;
- Resource files from all Gradle source sets are displayed; and
- Resource files for different orientations, screen types, and locales are combined into a single group per resource type.

The various capabilities provided by using this IDE have been discussed. Sufficient justification was provided for MotionShare to be implemented in Android Studio. The next section discusses the implementation of the prototypes.

4.6.2 Prototypes

MotionShare was implemented by breaking down the individual requirements of this artefact into sub-applications, which then became prototypes (Figure 4.11). Each prototype was individually designed, developed, and evaluated to provide a better understanding of the specific requirements. Figure 4.15 illustrates the development process for an Android application, which was used for each prototype (Google Inc., 2015b).

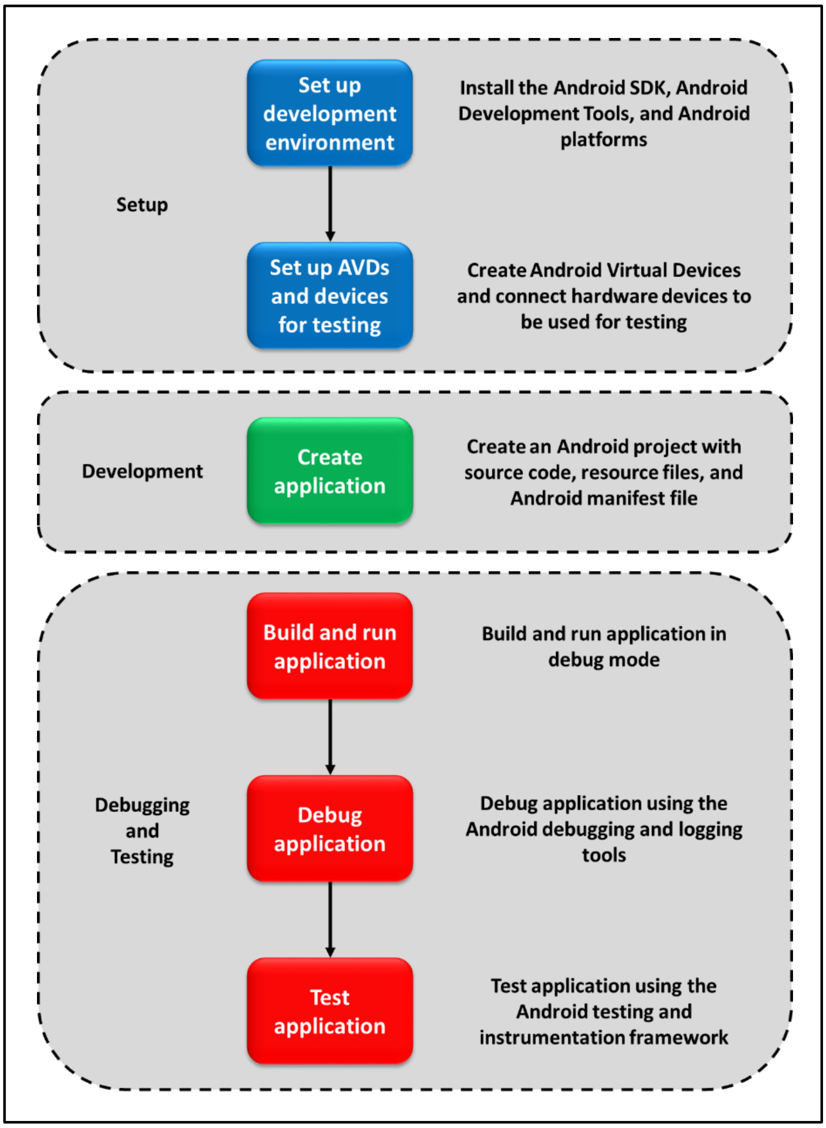


Figure 4.15: The Development Process for Android Applications (Google Inc., 2015b)

Implementation of MotionShare began with one prototype that developed into a functional component which could later be integrated into the artefact. The six prototypes of MotionShare, namely ProtoWiFiAP, ProtoBluetooth, ProtoFile, ProtoCompass, ProtoMap, and ProtoGesture, are individually discussed in the following subsections. A discussion of the integration of the prototypes into MotionShare to be used in the evaluation phase of this research concludes this section.

4.6.2.1 ProtoFile

ProtoFile deals with the location of different files in various storage locations on the mobile device based on file type. Once the files are located, they are displayed in a list (Figure 4.16) with the same hierarchical structure as existing information sharing systems (Section 2.7). File selection was implemented according to the existing information sharing systems reviewed (Section 2.7).

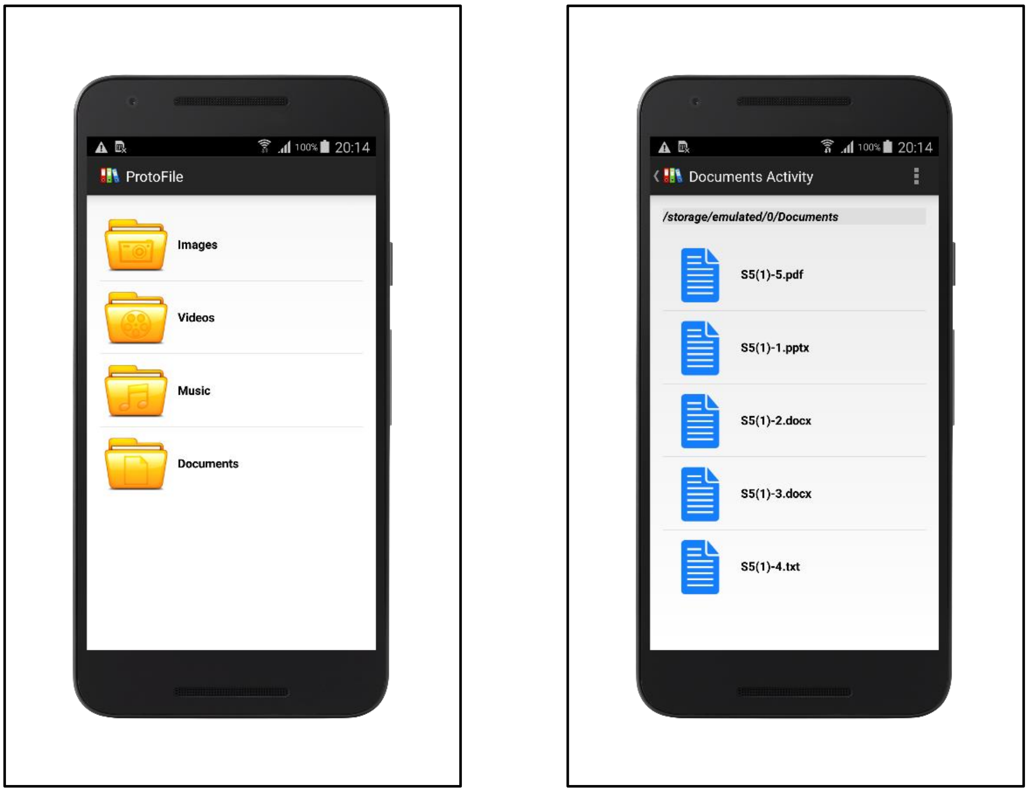
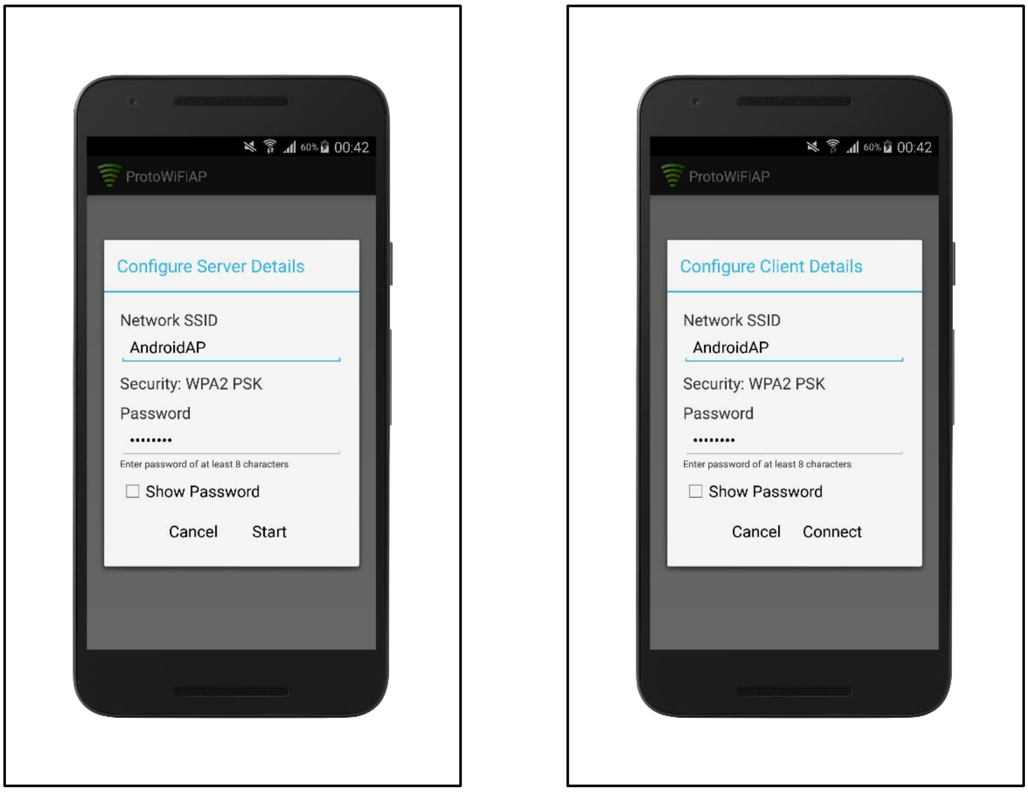


Figure 4.16: Screenshots of ProtoFile

4.6.2.2 ProtoWiFiAP

ProtoWiFiAP was developed to provide the user with the option of being either a client or a server. The objective of the server was to create a private and secure Wi-Fi hotspot whereby client devices could join the password protected network. The server can determine the network service set identifier (SSID) and the password (Figure 4.17(a)). A SSID is the public name of the wireless network to which the client wants to connect. The client requires the input of the network SSID and encrypted password (Figure 4.17(b)).



(a) (b)

Figure 4.17: Screenshots of ProtoWiFiAP

ProtoWiFiAP demonstrates the ability to support device communication using a Wi-Fi hotspot by turning a smartphone into a Wi-Fi AP. The Wi-Fi AP is typically a computing device that allows devices to connect wirelessly to a network using Wi-Fi (Figure 4.17). The Wi-Fi AP is differentiated from a hotspot, which refers to the physical space where Wi-Fi coverage exists. The communication between mobile devices across the Wi-Fi hotspot is based on sockets. Oracle (2015) define a *socket* as:

> *"A socket is one endpoint of a two-way communication link between two devices running on the same network."*

Both the client and server communicate back and forth through the socket according to an agreed-upon protocol (Arackal, 2014).

Figure 4.18: Wi-Fi Hotspot Diagram

4.6.2.3 ProtoBluetooth

Existing positioning techniques for mobile devices are too coarse-grained for the NUI interactions to utilise in a co-located environment, which is the context of this research (Section 4.2.4). Bluetooth experiments were conducted in order to determine the feasibility of using Bluetooth RSSI as an accurate indicator of distance between mobile devices. Results of the experiments showed that these RSSI values could approximate distance with a more accurate result than existing positioning techniques, specifically to within centimetre granularity. ProtoBluetooth allows the user to enable Bluetooth from within the application itself and to scan for devices in the immediate area using this technology. Results from the scan were processed and only the relevant information was displayed in a list (Figure 4.19).

The inverse relationship identified from the experiments was used in computing the approximate distance between mobile devices. It is important to note that the Bluetooth RSSI value tends to fluctuate considerably due to external factors that influence RSSI (Socha, 2015), such as absorption, interference, and/or diffraction.

Initially, a rule-based system was considered, but due to the variable nature of RSSI influenced by external factors, it was not considered viable and an alternative approach based on ML was used. ML is derived from the study of pattern recognition and computational learning theory in artificial intelligence (Shahriar, Kamruzzaman, & Beecham, 2014). ML algorithms typically function by constructing a model based on observed data, called training data, from which it learns and is able to make data-driven predictions or decisions, rather than following a set of hardcoded instructions. Thus, ProtoBluetooth was able to perform the computation of the approximate distance of the mobile devices (Figure 4.19).

Figure 4.19: Screenshot of ProtoBluetooth

An issue with using the Bluetooth RSSI is that it only allows for the computation of an approximate distance and provides no indication of the direction in which the devices are located. Therefore, a need arises to identify a possible position technique using the RSSI value and provide additional information to make this value a viable option for indoor positioning of co-located mobile devices.

4.6.2.4 ProtoCompass

ProtoCompass utilises embedded sensors in mobile devices. The selected hardware was identified as Android-powered devices (Section 4.6.1.1), which have embedded sensors to measure motion, orientation, and various environmental conditions. These sensors have the capacity to provide raw data with high precision and accuracy for monitoring three-dimensional device positioning. The Android platform by Google Inc. (2015c) supports three broad classifications of sensors, namely motion, environmental, and position.

Motion Sensors measure acceleration forces and rotational forces along three axes, namely x, y, and z. The Android platform provides access to several Motion Sensors, such as the accelerometer, gravity sensor, gyroscope, linear acceleration sensor, and rotation vector sensor (Google Inc., 2015c). Table 4.11 provides a summary of these sensors.

Table 4.11: Motion Sensors Supported on the Android Platform

Sensor	Description	Units of Measure
TYPE_ACCELEROMETER	Acceleration force along x, y, and z axis	m/s^2
TYPE_GRAVITY	Force of gravity along the x, y, and z axis	m/s^2
TYPE_GYROSCOPE	Rate of rotation around the x, y, and z axis	rad/s
TYPE_LINEAR_ACCELERATION	Acceleration force along the x, y, and z axis	m/s^2
TYPE_ROTATION_VECTOR	Rotation vector component along the x axis (x * sin(θ/2)) Rotation vector component along the y axis (y * sin(θ/2)) Rotation vector component along the z axis (z * sin(θ/2)) Scalar component of the rotation vector ((cos(θ/2))	Unitless
TYPE_STEP_COUNTER	Number of steps taken by the user since the last reboot while the sensor was activated	Steps

Environmental Sensors measure the various environmental properties, such as relative ambient humidity, illuminance, ambient pressure, and ambient temperature. These sensors are not applicable to this research.

Position Sensors measure the physical position of a device. These sensors are the geomagnetic field sensor and orientation sensor. Android also provides a proximity sensor to determine how close the face of a device is to an object. Table 4.12 provides a summary of the Position Sensors supported by Android (Google Inc., 2015d).

Table 4.12: Position Sensors Supported on the Android Platform

Sensors	Description	Units of Measure
TYPE_MAGNETIC_FIELD	Geomagnetic field strength along the x, y, and z axis	μT
TYPE_ORIENTATION	Azimuth: angle around the z-axis Pitch: angle around the x-axis Roll: angle around the y-axis	Degrees
TYPE_PROXIMITY	Distance from object (only near and far values - how far away a person's head is from the face of the mobile device, for example, when a user makes or receives a call)	cm

Motion and Position Sensors allow for interaction techniques to be designed, developed, and used, which capitalise on the hand-held nature of mobile devices. Sensor-based interaction techniques provide users with a natural and intuitive way to interact with their devices.

ProtoCompass involved the design and development of a digital compass (Figure 4.20). The Motion (Table 4.11) and Position (Table 4.12) Sensors embedded in mobile devices were utilised in this prototype. These sensors are the accelerometer, magnetometer, and gyroscope. An *accelerometer* (Aviv, Sapp, Blaze, & Smith, 2012; Lee, 2015b) is an embedded sensor in a mobile device used to measure the acceleration forces on all three physical axes (x, y, and z) and can determine the device's physical position. A *magnetometer* (Zhang & Sawchuk,

2012) is a magnetic sensor embedded in a mobile device to determine the heading of the device, provided the user is holding it parallel to the ground.

Similarly, a gyroscope (Thomason & Wang, 2012) is an embedded sensor that provides an additional dimension to the information supplied by the accelerometer by measuring the rotation or twist of the device. The *gyroscope* measures the angular rotational velocity of a device. Unlike the accelerometer, the gyroscope is not affected by gravity. It can be concluded that the accelerometer and gyroscope measure the rate of change differently. In practice, this means that an accelerometer will measure the directional movement of a device, but will not be able to accurately resolve its lateral orientation or tilt during this movement accurately, without the use of the gyroscope which would provide the additional information (Kratz, Rohs, & Essl, 2013).

ProtoCompass initially displays a calibration message to ensure the three steps are performed to calibrate the embedded sensors (Figure 4.20). These steps did affect the behaviour of the compass and, to some extent, improve the accuracy. The compass was developed based on the real-time sensor data parsed through the sensors, which resulted in the compass constantly moving from the substantial data received. The sensor data provided by these sensors proved to be changing at a considerable rate, similar to the behaviour of Bluetooth RSSI. The solution was to poll the sensor values over a time interval of two seconds and only extract those values, which are useful to this context and eliminate the unnecessary noise. The issue of the sensor data varying considerably was resolved by applying a Low-Pass Filter.

A *Low-Pass Filter* (Lee, 2014, 2015a) is a smoothing algorithm that smooths the sensor values by filtering out high-frequency noise and *"passes"* low-frequency or slowly varying changes. As a result, a more stable compass is displayed and the orientation changes are smoother. The accuracy of the compass was compared to existing compass applications on the Google Play Store and it was found to be similar (informal testing).

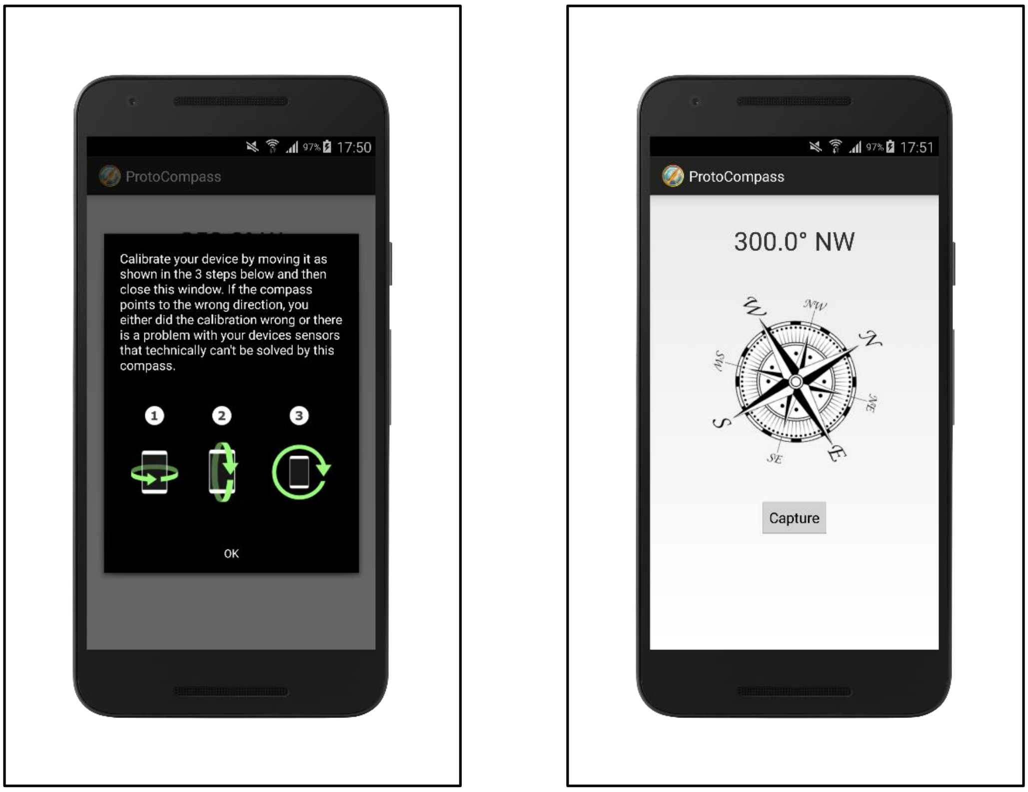


Figure 4.20: Screenshots of ProtoCompass

Figure 4.21 illustrates how multiple sensors were combined to create sensor fusion. Sensor fusion is the combination of different sensors, which results in the new data being more useful than sensor data of each individual sensor. The sensor data from the accelerometer, magnetometer, and gyroscope are combined through the Low-Pass Filter and used to provide an enhanced compass.

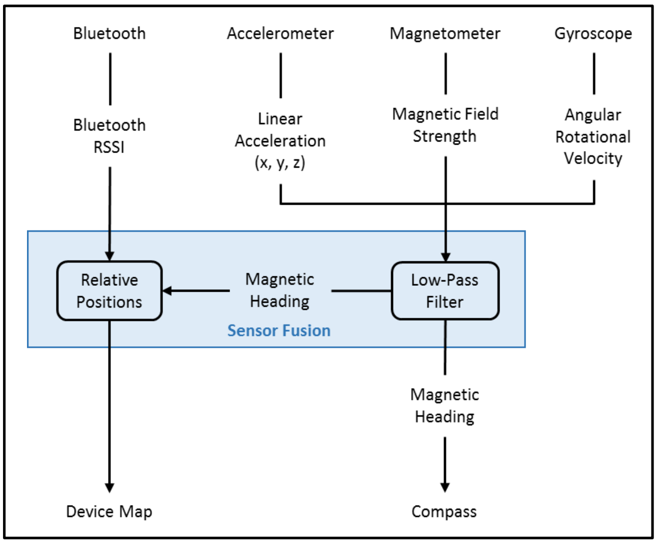


Figure 4.21: Sensor Fusion by Combining Output from Multiple Sensors

4.6.2.5 ProtoMap

ProtoMap computes a map based on the information obtained from ProtoBluetooth. Figure 4.22 shows poses of several devices which are randomly placed on a table. In Figure 4.23, screenshots of each device screen relative to its pose is shown. Device A, which acted as the server, was pointed in the direction of Device B (Figure 4.23(a)). Device B was a client and was pointed towards Device C (Figure 4.23(b)). Figure 4.23(c) shows what map is displayed on Device C as it is facing away from all the other devices in the environment. Device D was oriented in the direction of both Devices A and C, with Device B positioned to the left of Device D (Figure 4.23(d)). Each device is represented by a uniquely coloured dot and labelled with the device name. The dot displayed in the centre of the map represents the pose of the subject device with the relative positions of other devices displayed around it.

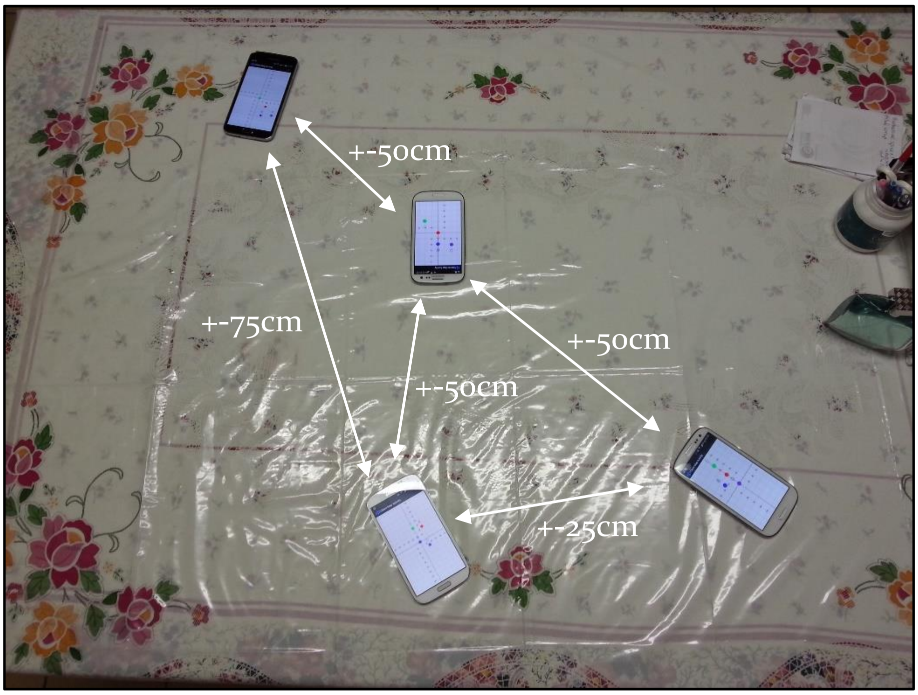


Figure 4.22: Screenshot of the Devices' Poses

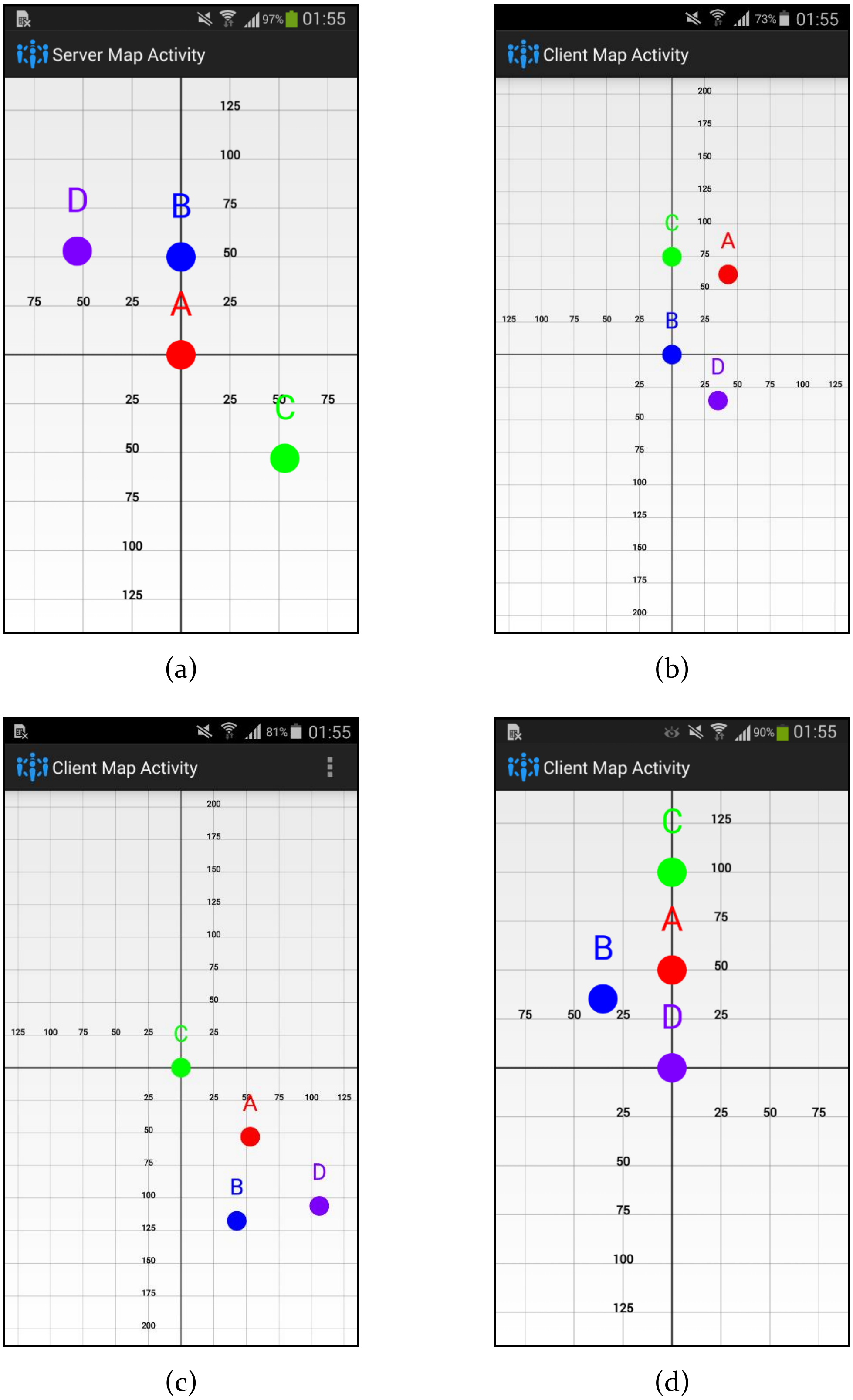


Figure 4.23: Screenshots of ProtoMap on the Different Devices

4.6.2.6 ProtoGesture

ProtoGesture is discussed in depth in the following chapter together with the gestures to be potentially used. The design of the gestures considered the mobile device's context of use. To ensure the appropriateness of the gestures, focus groups were conducted to determine which gestures people were currently using with their mobile devices and what gestures people expected to use for specific tasks in MotionShare. Furthermore, the focus groups ensured the gesture set was socially and contextually acceptable. Section 4.6.2.7 discusses the MotionShare architecture

4.6.2.7 MotionShare Architecture

Upon completion of all six prototypes in terms of design, development, testing, and updating, they were integrated into MotionShare, to satisfy the requirements of the application. Figure 4.24 illustrates the MotionShare architecture. The *GUI* layer contained ProtoGesture and ProtoMap. ProtoGesture involved the NUI interaction techniques which are discussed in Chapter 5. ProtoMap managed the computation and display of the device dots on map. The *Information Sharing* layer consisted of ProtoFile and ProtoWiFiAP, to handle the file selection process and the sharing of information among mobile devices respectively. In the *Proxemic (Pose Calculations)* layer, ProtoBluetooth was used to determine the distances between mobile devices (position), whereas ProtoCompass used sensor fusion with a Low-Pass Filter to smooth the sensor data (orientation).

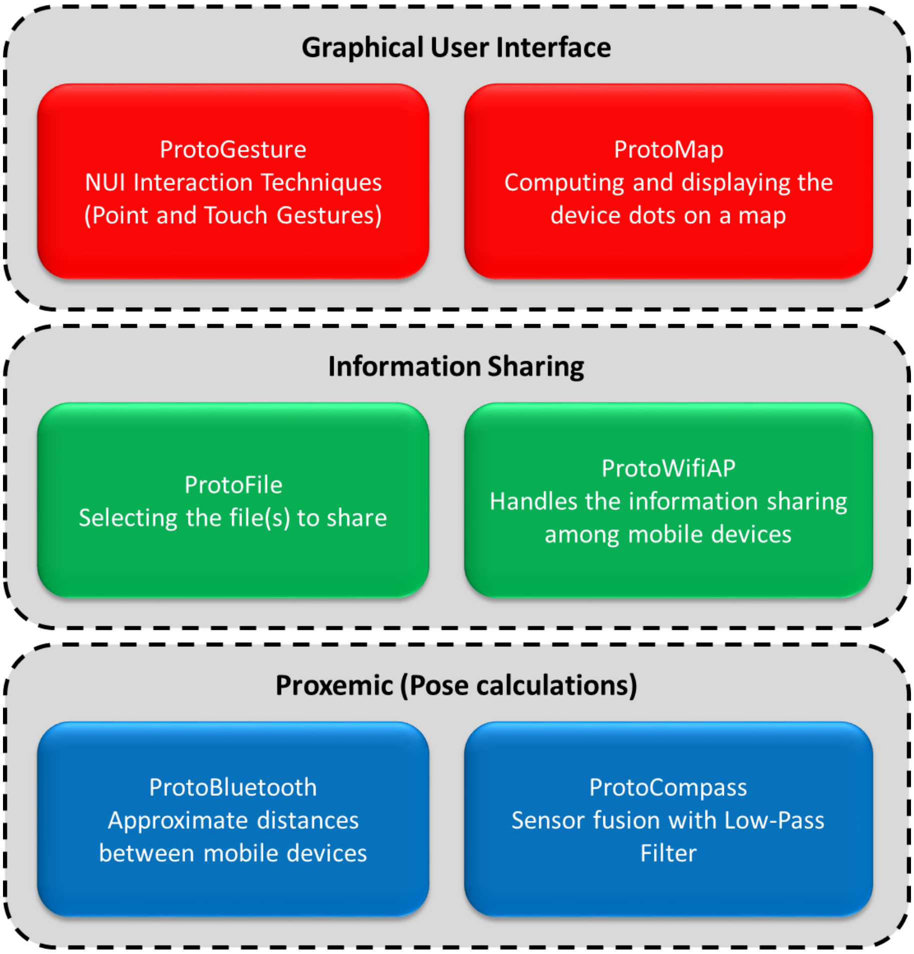


Figure 4.24: MotionShare Architecture

4.7 Conclusion

This chapter followed the third activity, ***Design and Develop Artefact***, within the Design Cycle of the DSR methodology (Section 1.7.3.1). The experiments conducted and prototyping were used to address the third research question (Section 1.7.1) of this chapter:

> *"RQ3. How should the relative pose for co-located mobile devices be calculated?"*

The design and implementation of MotionShare was discussed according to the development methodology described (Section 4.4) and adhered to. The methodology mentioned the design and development of various prototypes. Each prototype focused on and demonstrated a core functionality to be integrated into the design artefact. The information sharing process identified in Chapter 2 formed the basis of the expected functionality, which helped in the design process. A full paper on this research was accepted and presented at the Southern Africa Telecommunications Networks and Applications Conference 2015 (Lee Son, Wesson, & Vogts, 2015), which supported the feasibility of the artefact. Contents of this paper were discussed throughout this chapter. The design and development of MotionShare is an important aspect of the DSR methodology (Section 1.7.3.1).

The positioning techniques section (Section 4.2) identified the appropriate existing techniques with their accuracy and granularity. This was followed by a discussion on the experiments conducted and results obtained (Section 4.3). Section 4.6 described the implementation of MotionShare, which included the tools used, the various prototypes, and integration of the prototypes into MotionShare.

The main deliverables of this Design Cycle were the design of the application architecture and the implementation of the various prototypes which were integrated into MotionShare.

This chapter achieved the third research objective (Section 1.7.2):

> *"RO_3. To determine how the relative pose can be calculated for co-located mobile devices."*

The next chapter discusses the results of the focus groups conducted, which determine the design and implementation of gestures.

Chapter 5: Gesture Design

5.1 Introduction

The design and implementation of MotionShare was discussed in Chapter 4. Several prototypes were developed and then integrated into MotionShare. This chapter continues from Chapter 4 and also addresses the third activity in the DSR methodology (Section 1.7.3.1) that is performed within the Design Cycle (Figure 5.1), namely ***Design and Develop Artefact***. This chapter addresses two research questions (Section 1.7.1) identified:

> "*RQ4. How should NUI interaction techniques be designed to support information sharing among co-located mobile devices?*
>
> *RQ5. How can a proxemic prototype NUI be developed to support information among co-located mobile devices?"*

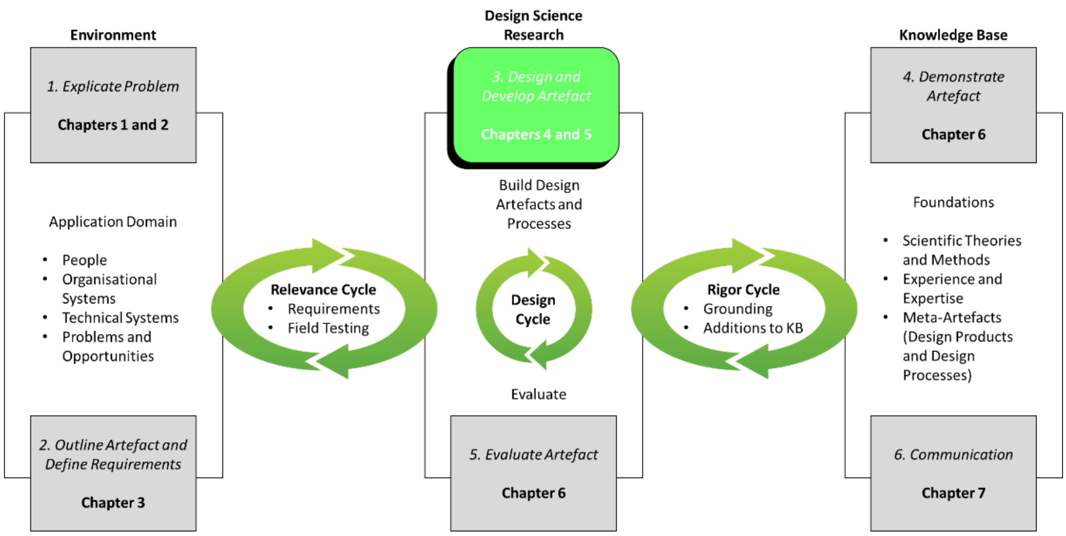


Figure 5.1: Chapter 5 Position in the Adapted DSR Methodology

These research questions are answered by discussing the design and implementation of the gestures to be used within MotionShare. MotionShare involves the use of gesture-based interactions to support information sharing among co-located mobile devices. ProtoGesture was designed and implemented to test the feasibility of the gestures as well as user acceptance of the gestures. Initial user acceptance of the gestures was established from multiple focus groups conducted. Figure 5.2 illustrates the structure of this chapter.

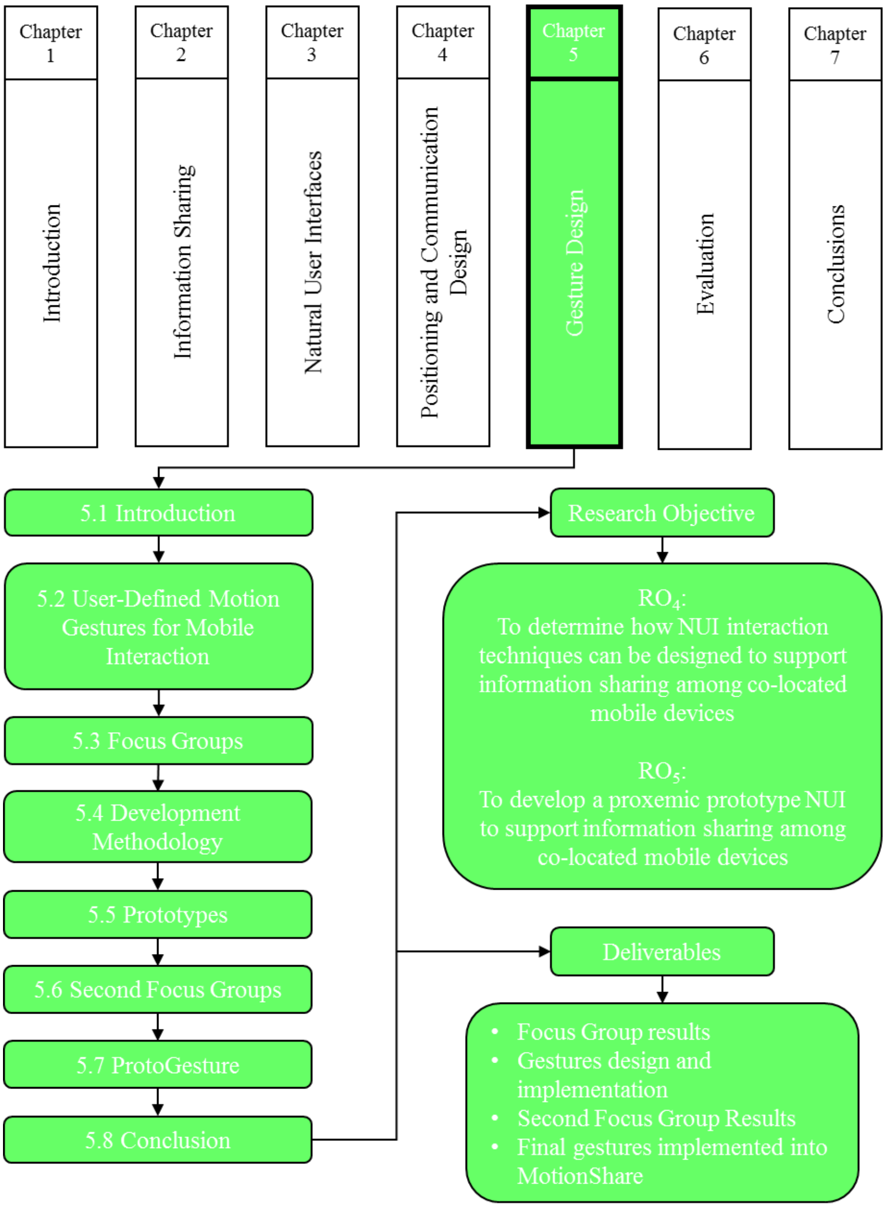


Figure 5.2: Chapter 5 Structure

5.2 User-Defined Motion Gestures

Ruiz et al. (2011) developed a list of user-defined gestures from the feedback elicited from 20 participants. These participants were required to design and perform a motion gesture with a smartphone device (a cause) to complete a specific task on the smartphone (an effect). Participants were presented with a task list, which consisted of 19 tasks (Table 5.1).

Table 5.1: List of Tasks (Ruiz et al., 2011)

Category	Sub-Category	Task Name
Action	System or Phone	Answer Call
		End Call
		Ignore Call
		Place Call
		Voice Search
	Application	Act on Selection
Navigation	System or Phone	Home Screen
		Application Switch Next
		Application Switch Previous
	Application	Next (Vertical)
		Previous (Vertical)
		Next (Horizontal)
		Previous (Horizontal)
		Pan Left and Right
		Pan Up and Down
		Zoom In and Out

The results gathered from the participants for each task were classified into two categories, namely action and navigation. For each task, similar gestures were grouped together and the largest size was then selected as the representative gesture for the specific task. There was no agreement among participants regarding the gestures for switching to another application, switching to a previous application, and acting on selection. As such, gestures for these tasks were excluded from the list of user-defined gestures. The resulting list of user-defined gestures is illustrated in Figure 5.3.

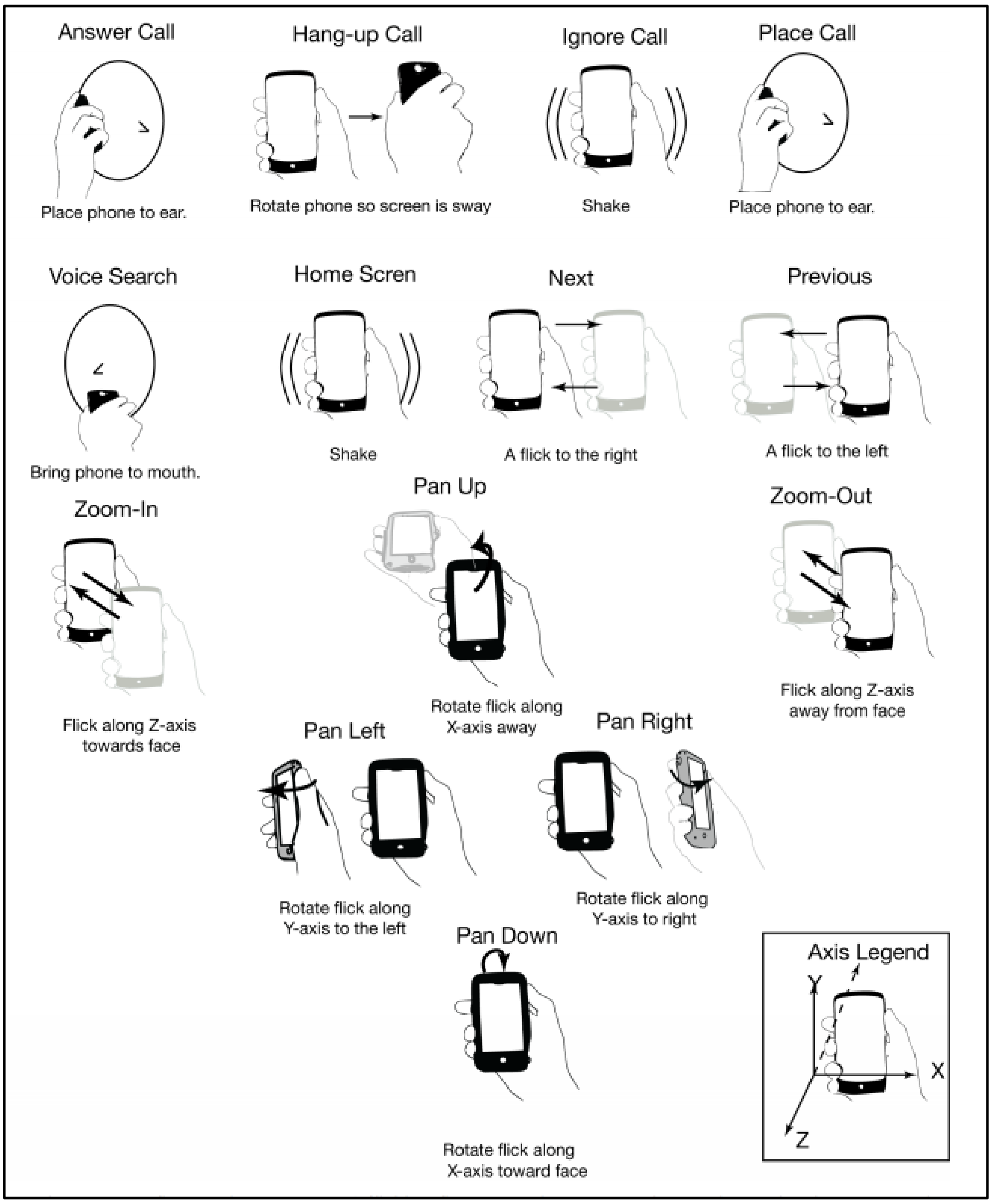


Figure 5.3: The User-Defined Motion List of Gestures (Ruiz et al., 2011)

The article involved participants designing and performing gestures with a mobile device to complete specific tasks. This article served as a basis for which the focus groups were conducted. The desired information that the research required is the appropriate gestures for the specific tasks performed using MotionShare.

5.3 Focus Groups

Focus groups (Section 1.7.3.3) were used in the DSR methodology. Two focus groups of four participants each were seated around a table. The placement of the participants was to simulate an actual co-located environment, whereby individuals would gather for a formal or social event.

5.3.1 Location

The focus groups were conducted in the Seminar Room at the NMMU South Campus Department of Computing Sciences. This venue was selected, because it allowed the moderator to conduct these groups in a controlled environment with no external factors to influence the participants.

5.3.2 Procedure

Participants were presented with consent forms, detailing all the necessary information, to fill in before the focus group could commence. Upon completion of these forms, the moderator provided the information verbally (Appendix F). The focus group was video recorded and observations during the session were written down to reaffirm what was discussed.

Table 5.2: Focus Group Questions (n=8)

Moniker	Questions
Single Select	What action would you perform to **select a file**?
Multiple Select	What action would you perform to **select multiple files**?
Single Share	What action would you perform to **share a file with another mobile device**?
Multiple Share	What action would you perform to **share a file with multiple devices**?
Initiate	What action would you perform to **initiate a file transfer**?
Accept	What action would you perform to **accept a file transfer**?
Reject	What action would you perform to **reject a file transfer**?
Cancel	What action would you perform to **cancel a file transfer**?

The primary objective of these groups was to identify a set of possible gestures, which would be used in performing the specified user tasks of MotionShare. The moderator presented several questions to the focus group (Table 5.2). These questions were based on the functional requirements of MotionShare (Section 4.5).

5.3.3 Results

The results of the two focus groups conducted were aggregated and categorised according to the questions presented by the moderator. Table 5.3 shows the results of these focus groups. Five participants (62.50%) indicated a long press action to initiate the file selection process, whereas the remaining three participants (37.50%) mentioned a long press with a hold and drag action of the selected file(s) into a basket icon or avatar icon representation of the recipient (*Single Select*). The second question (*Multiple Select*) presented to the focus groups was related to multiple file selection. Similarly, the same five participants (62.50%) indicated a long press to initiate the selection process, after which a single touch was required to select other files.

The other 37.50% of participants (n=3) suggested a custom touch gesture be drawn on the device screen to invoke a system response of selecting all the files. Participants responded to *Single Share* with suggestions of a swipe or flick gesture (50.00%), throw and catch concept (12.50%), and point device to recipient (37.50%). The throw and catch concept was based on the Bump app, which was discontinued by Google as the user experience was poor. Users complained of the application not registering the throw action (9 out of 10 times performed) and also the catch action (8 out of 10 times performed).

With regard to sharing among multiple devices (*Multiple Share*), two participants (25.00%) answered with multiple swipes or flicks to recipients. The other six participants (75.00%) preferred a single action for all and suggested moving the device in an arc-like manner or shape in the air to cover the intended recipients. These six participants (75.00%) also suggested an alternative option, which was to draw a custom gesture on the screen, for example, an arc to cover the intended recipients.

Initiating a file transfer (*Initiate*), 62.50% of participants (n=5) were accustomed to simply selecting a *Share* button. The remaining three participants (37.50%) suggested pointing the device and tilting it towards the intended recipient. These three participants (37.50%) felt this suggestion was similar to handing a document over to another individual.

Six participants (75.00%) preferred a simple dialog box containing a *Yes* and *No* button to either *Accept* or *Reject* a file transfer. Two participants (25.00%) suggested tilting the device towards themselves (*Accept*). These two participants (25.00%) suggested a concept of swinging the device like a pendulum to indicate a *No* action (*Reject*), similar to an individual shaking his head. When the question was posed on cancelling a file transfer (*Cancel*), all the participants (n=8) mentioned a cross icon displayed next to the progress of the file transfer(s) would suffice.

Table 5.3: Focus Groups Results (n=8)

Moniker	Responses
Single Select	• Long press to initiate the file selection process, typically evident in almost all mobile applications (n=5, 62.50%) • Long press, hold, and drag selected file(s) into a basket icon or designated target (n=3, 37.50%)
Multiple Select	• Long press to initiate the file selection process and single touch for multiple file selection after long press initiation (n=5, 62.50%) • Custom touch gesture to select all files (n=3, 37.50%)
Single Share	• Swipe or flick (n=4, 50.00%) • Throw (send and initiate) and catch (receive and accept). Participants referred to the Bump and Flock concept discontinued by Google (n=1, 12.50%) • Point device towards recipient (n=3, 37.50%)
Multiple Share	• Multiple swipes or flicks (n=2, 25.00%) • Moving device in an arc shape to cover intended recipients (n=6, 75.00%) • Draw custom gesture (arc) to share to all users in a selected quadrant (n=6, 75.00%)
Initiate	• Select a *Share* button (n=5, 62.50%) • To initiate the transfer, point the device towards the intended recipient or possibly tilt away (n=3, 37.50%)
Accept or Reject	• Simple dialog box with a *Yes* and *No* button to accept or reject the transfer (n=6, 75.00%) • To accept the transfer, pull the device towards yourself such as tilt towards (n=2, 25.00%) • Swing the device like a pendulum to indicate a *No*, like shaking your head. Cool concept suggested to be different from the standard *No* button (n=2, 25.00%)
Cancel	• Simple cross icon displayed next to the progress bar of each file transfer (n=8, 100.00%)

5.3.4 Discussion

The results of the two focus groups were aggregated (Table 5.3) and compared to the existing systems (Section 3.6) identified from the literature study. Table 5.4 presents the comparison between the focus groups results and the literature study. This table also presented the decision on the interaction technique to be implemented for MotionShare. *Single Select* and *Multiple Select* was decided to be a long press to initiate the file selection process into a single touch for multiple files, which was implemented in ProtoGesture.

The decision on sharing with a single device (*Single Share*) was to allow the user to point with the device towards the recipient. Multiple device sharing (*Multiple Share*) was decided to also use the point gesture, which would detect multiple devices in a 45 degree arc, and allow users to draw a custom gesture on the device screen by highlighting the recipients. The second option for multiple device sharing allows for selective multiple sharing as the user can decide which recipients to highlight or cover.

To initiate the file transfer using the device pointed at the recipient (*Initiate*), the *Share* button was decided. With regard to the custom touch gesture, initiating file transfer would occur when the user lifted his finger up from the device screen. Accepting or rejecting the file transfer (*Accept* or *Reject*) was removed, as existing systems (Section 3.6) showed that the devices which joined a private network had already made the decision to send and receive files. Therefore, the decision to remove the notification of accepting or rejecting an incoming file was based on evidence provided from literature. The example is also considered with large amounts of information sharing, since the constant notification to receive files can become potentially annoying and tedious. Similarly, the *Cancel* button for a file transfer (*Cancel*) was also removed as the files were automatically accepted.

Table 5.4: Comparison of Focus Groups Results, Literature Study, and Decision

Moniker	Focus Group	Literature Study	Decision
Single Select	• Long press to initiate file selection • Long press, hold, and drag	• Long press • Single touch (Feem, Share Link, SuperBeam, Xender)	• Long press to initiate file selection
Multiple Select	• Long press to initiate file selection with single touch for multiple file selection • Custom touch gesture	• Long press • Select All option (Feem, Share Link, SuperBeam, Xender)	• Long press to initiate file selection with single touch for multiple file selection • Custom touch gesture
Single Share	• Swipe or flick • Throw and catch • Point device	• Swipe or flick file (Flick) • Hold the file with long press and flick device (Zapya) • In-air wave towards recipient (AirLink)	• Point device • Draw custom gesture
Multiple Share	• Multiple swipes or flicks • Move device arc shape • Draw custom gesture	• Select *Broadcast* button (Flick, Feem, Share Link) • Long press and drag file to recipient avatars (Zapya) • Single touch to select the recipients (SuperBeam) • Multiple in-air waves towards recipients (AirLink)	• Orientate and point device to recipients • Draw custom gesture and highlight recipients
Initiate	• Select *Share* button • Point device and tilt away	• When dragging file to recipient and finger is lifted (Flick, Feem, Share Link, Zapya, SuperBeam) • In-air wave towards recipient (AirLink)	• When orientating and pointing to recipient, select *Share* button • When draw custom gesture, initiated when user lifts finger
Accept or Reject	• Dialog box with a *Yes* and *No* button to accept and reject the file respectively • Pull device towards yourself (tilt towards) to accept the file transfer • Swing device like pendulum to reject the file transfer	• Dialog box (Feem, Share Link, MobiSurf, Zapya) • No accept mechanism as it is automatic (Flick, SuperBeam, Xender, AirLink)	• Automatically accepts all incoming file transfers
Cancel	• Cross icon displayed next to progress bars of the file transfer(s)	• Cancel file transfer with the cross (Feem, Share Link, MobiSurf, Zapya) • No cancel mechanism (Flick, SuperBeam, Xender, AirLink)	• No cancel mechanism

5.4 Development Methodology

Prototyping was used to test the feasibility of the gestures (identified from the focus groups) to be implemented into MotionShare. (Section 4.4). The following four prototypes were created:

- ***ProtoTilt*** focuses on the detecting the movements of the mobile device and inferring a context;
- ***ProtoFileGesture*** involves the drawing of a gesture on the device screen, which represents a specific action performed on the system;
- ***ProtoTouch*** detects the touch gesture performed on the device screen and whether it highlights any area of the displayed devices; and
- ***ProtoPoint*** utilises the position information, which was displayed in ProtoMap (Section 4.6.2.5), by changing the positions of the device dots depending on the direction in which the device is pointed.

Each of these prototypes focused on a different gesture to be potentially used within MotionShare. The next section discusses the design and implementation of these prototypes.

5.5 Prototypes

The design and implementation of each of the four prototypes are discussed, which show how each prototype functions with the specific gesture performed.

5.5.1 ProtoTilt

Figure 5.4 shows the design of ProtoTilt. The user performs a tilt or flick gesture. The accelerometer in the mobile device processes this gesture into values, which are converted into the respective axes, namely x, y, and z. These values are the raw input data of ProtoTilt. To classify the user initiated gesture correctly, the raw input data needed to be smoothed. The smoothing of the raw data was performed to remove any noise, discrepancies, or outliers, and allow ProtoTilt to correctly classify the data. The classification based on the data received is then visually displayed on the mobile device.

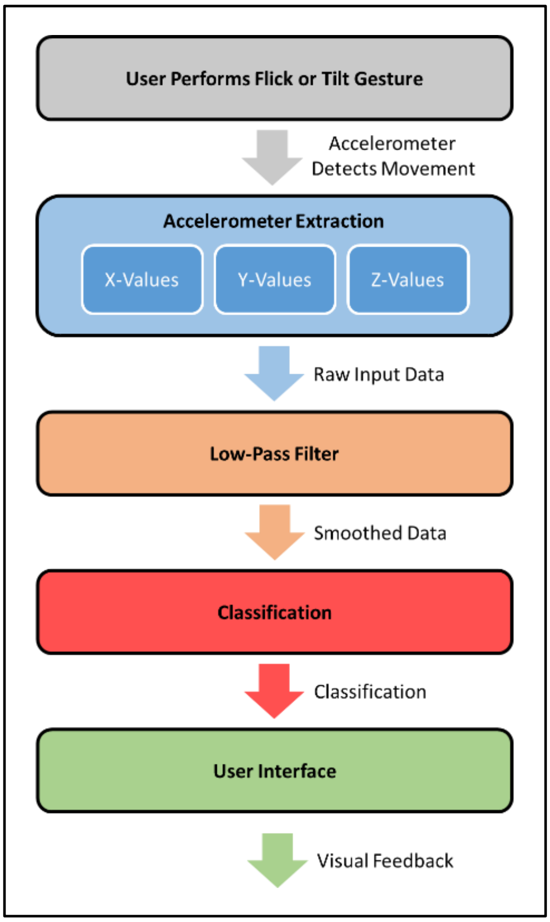


Figure 5.4: ProtoTilt Design

ProtoTilt uses the accelerometer sensor to measure the device movements performed by the user. The accelerometer sensor measures the linear movements in three dimensions, namely side-to-side, forward-and-back, and up-and-down (labelled x, y, and z respectively in Figure 5.5). Every device movement generates a data element, which contains the accelerometer reading in all three dimensions.

Tilting the device towards the left side along the x-axis causes ProtoTilt to display LEFT (Figure 5.6), whereas tilting to the right side causes ProtoTilt to display RIGHT. Similarly, tilting the device away from the user along the y-axis displays AWAY and tilting towards displays TOWARDS (Figure 5.7). When the device is held parallel to the ground, ProtoTilt displays FLAT. These device movements of

ProtoTilt were used to show how information could be shared to another device, depending on the location of the recipient.

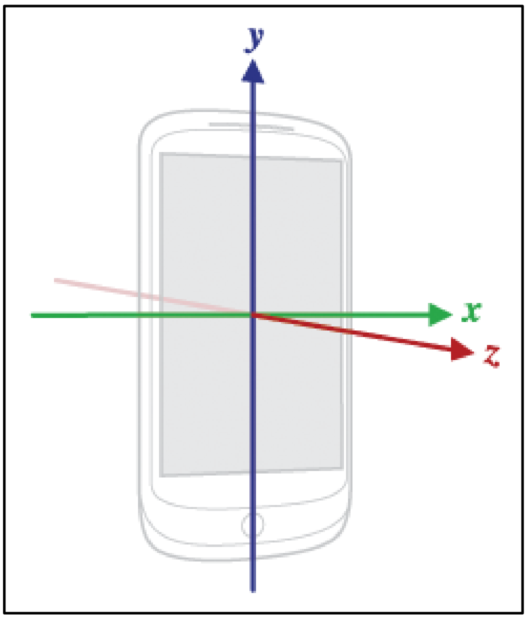


Figure 5.5: Accelerometer Axis of Measurement ProtoFileGesture (Google Inc., 2015g)

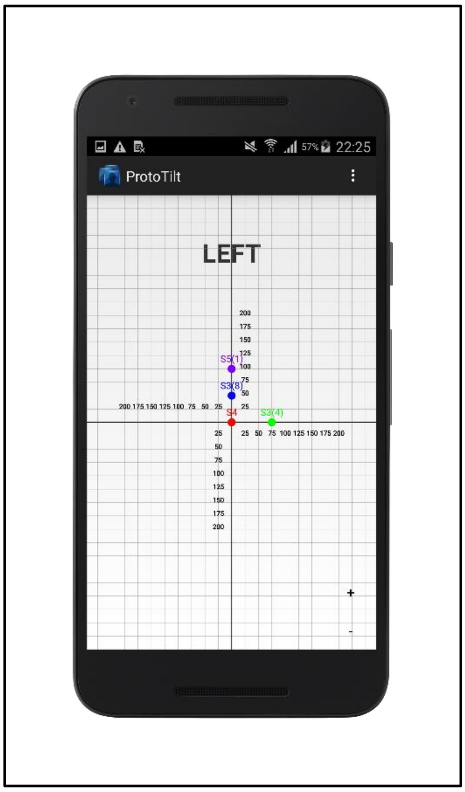


Figure 5.6: Device Tilted to the Left

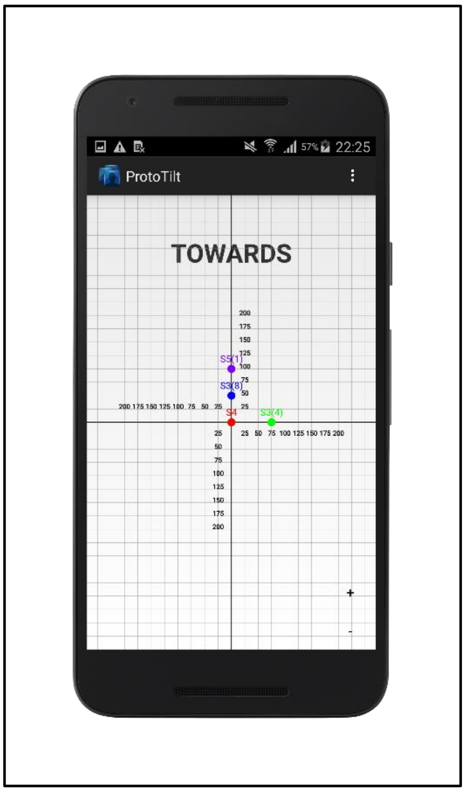


Figure 5.7: Device Tilted Towards the User

5.5.2 ProtoFileGesture

Figure 5.8 shows the design of ProtoFileGesture. The user performs a pre-defined touch gesture on the device screen. ProtoFileGesture detects the touch gesture and compares it to the existing gesture database of ProtoGesture. In the event the touch gesture performed is matched against a gesture in the gesture database, then ProtoFileGesture performs a specific system response, otherwise the gesture performed was not recognised and the relevant message is displayed.

For example, the gesture database contains a circle. If the user draws a circle on the device screen, ProtoFileGesture detects the circle and matches it against the gesture database. Upon successful gesture match, ProtoFileGesture selects all the files in the current directory because the circle gesture represents a *Select All* action. The matching threshold (confidence value) is used to decrease the amount of false positives of ProtoFileGesture.

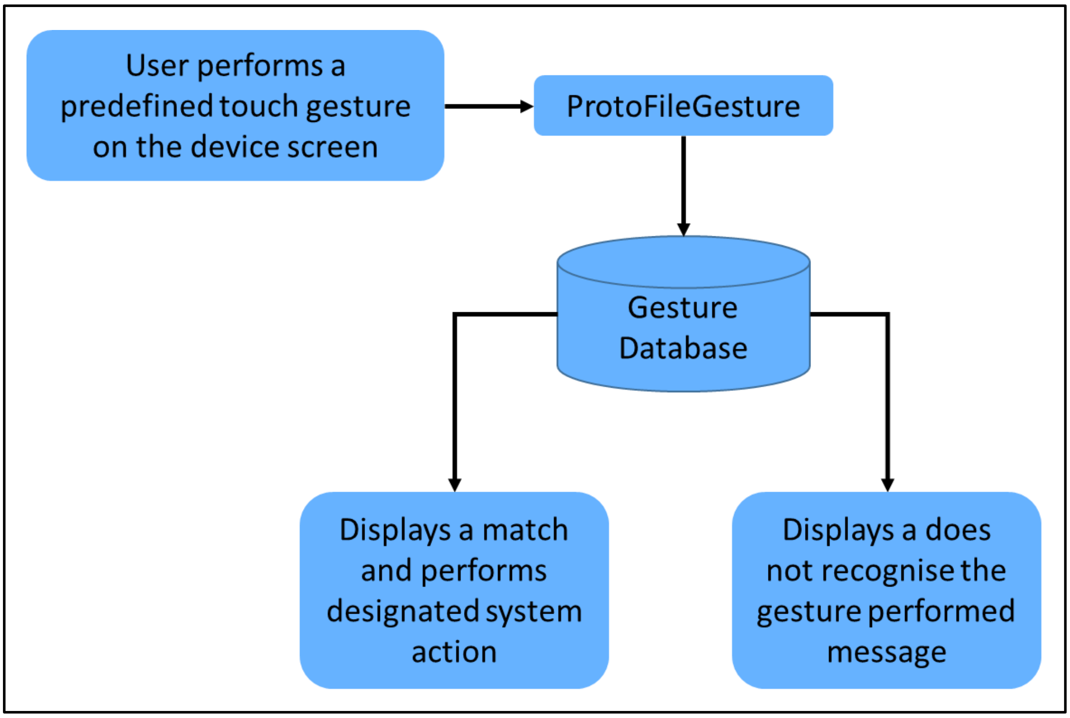
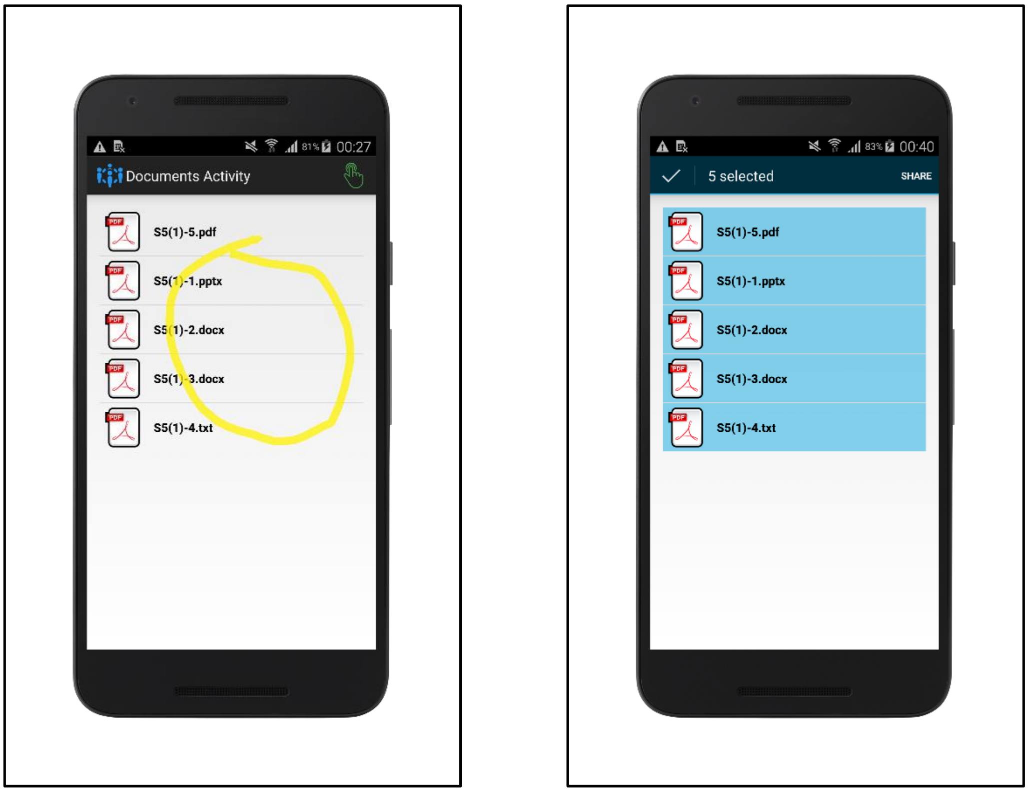


Figure 5.8: ProtoFileGesture Design

ProtoFileGesture focuses on the user performing a pre-defined touch gesture on the device screen to invoke a specific system response, which is to select all files in the current directory. ProtoFileGesture requires a gesture database to recognise a pre-defined touch gesture. The gesture database was generated by initially training the prototype on the specific gesture. After training the prototype, whenever the user performs the pre-defined gesture (Figure 5.9(a)), ProtoFileGesture responds by selecting all the files (Figure 5.9(b)).



(a) (b)

Figure 5.9: Screenshots of ProtoFileGesture

5.5.3 ProtoTouch

Figure 5.10 shows the design of ProtoTouch. ProtoTouch detects a touch event caused by the user's finger touching the device screen. The initial touch event is classified as a *TouchDown* and captures the x and y coordinates of the screen. As the user drags his finger across the screen, the continuous touch events are classified as *TouchMove* and the x and y coordinates are also captured. When the user decides to lift his finger from the device screen, the touch event is classified as a *TouchUp* and the last x and y coordinates are captured.

ProtoTouch processes all these x and y coordinates and determines whether the existing device dots displayed on the screen are highlighted by the user's touch gesture. To determine whether or not any of these devices dots were highlighted, a Circle to Circle Collision Detection algorithm was implemented. A Circle to Circle Collision Detection algorithm is a collision detection algorithm that uses the centre points of two circles and ensures the distance between these points are less than the sum of the two radii. ProtoTouch considers the possibility of the user highlighting an area next to the device dot (within the range of 2mm on the device screen) and therefore, still recognised the device dot as being highlighted.

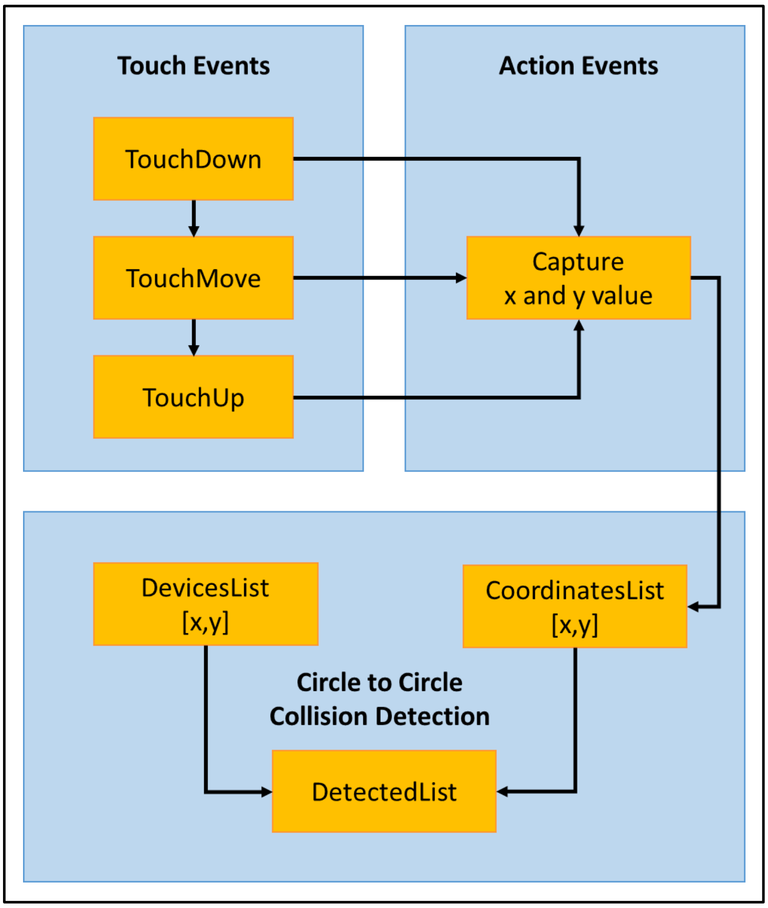


Figure 5.10: ProtoTouch Design

ProtoTouch involves detecting the gestures drawn on the device screen by the user's finger (Figure 5.11). The user highlights by touching the device screen with his finger and dragging across the intended device dots to share the selected file(s). When the user lifts his finger from the device screen, the selected file(s) are transferred to the highlighted device dots.

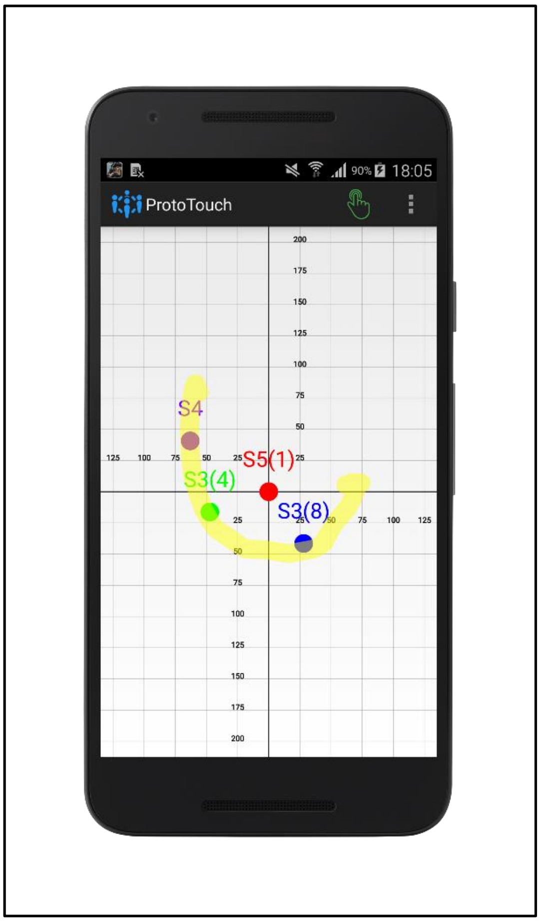


Figure 5.11: Screenshot of ProtoTouch

5.5.4 ProtoPoint

ProtoPoint uses the information acquired from the sensor fusion (Figure 4.21) in ProtoCompass (Section 4.6.2.4) and ProtoMap (Section 4.6.2.5). ProtoPoint uses this information to detect the orientation and direction of the device, and displays the device map accordingly.

ProtoPoint allows the user to point the mobile device in the direction of the intended recipient(s), by utilising the position information used to generate the device map in ProtoMap (Section 4.6.2.5). In Figure 5.12, ProtoPoint displays the device map when pointed towards a device, for example *S3(2)*. When the device is pointed in the direction of multiple devices, for example *S5(1)* and *S5(2)*, ProtoPoint re-calculates the device map based on the orientation of the device (Figure 5.13). Therefore, ProtoPoint has detected the change in device orientation and always shows the device dot(s) in front of this device on the positive y-axis.

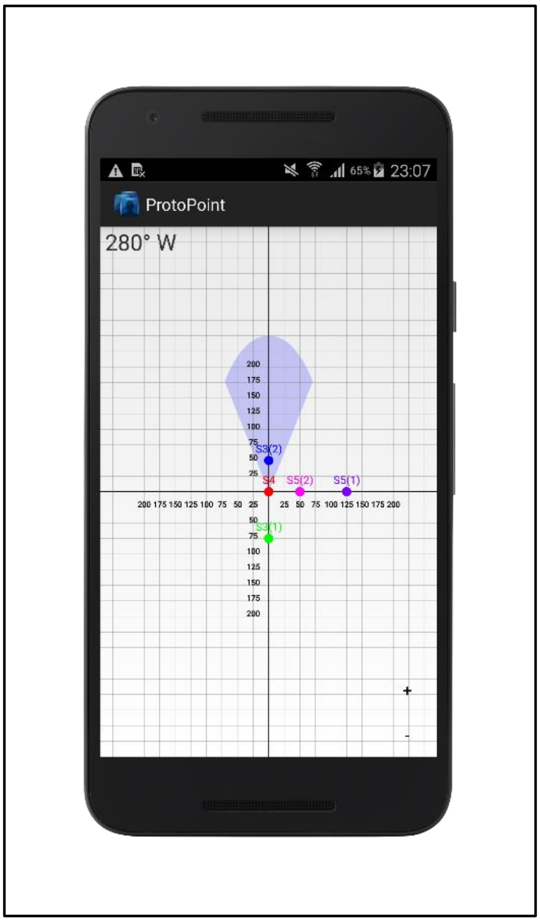


Figure 5.12: Device Pointed Towards One Device

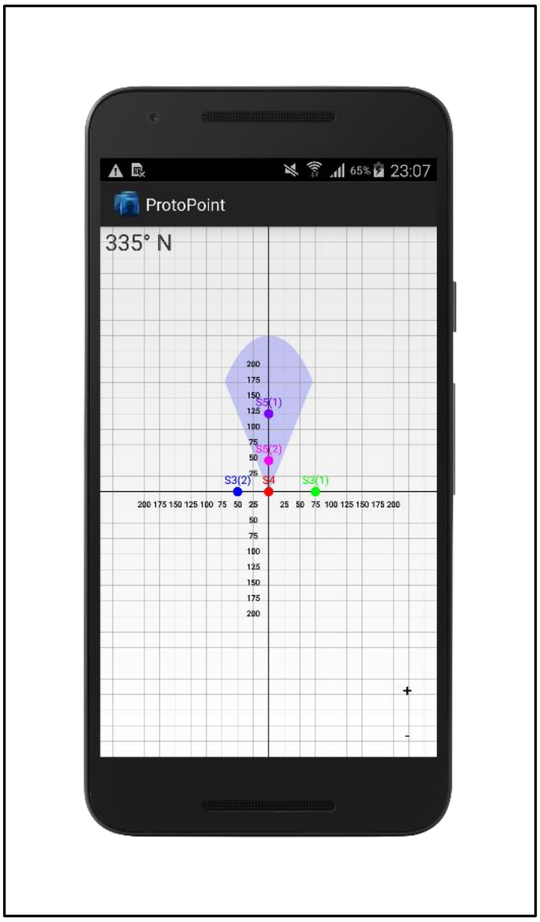


Figure 5.13: Device Pointed Towards Multiple Devices

5.5.5 Summary

The following four prototypes were discussed:

- ***ProtoTilt*** uses the accelerometer values to classify the user's gesture performed with the mobile device.
- ***ProtoFileGesture*** compares the touch gesture performed by the user against the gesture database, which contains pre-defined gestures for specific system actions. The touch gesture involved in ProtoFileGesture to perform a select all files action was drawing a circle on the device screen.
- ***ProtoTouch*** classifies the different touch events invoked by user's finger touching the device screen. When the user lifts his finger from highlighting any area on the device screen, ProtoTouch determines which device dots were highlighted to receive the selected files.

- ***ProtoPoint*** uses the information from the previous prototypes, namely ProtoCompass (Section 4.6.2.4) and ProtoMap (Section 4.6.2.5), to determine the orientation and direction of the device. Based on this information, ProtoPoint also determines the exact number and which devices it was pointed towards.

These prototypes discussed above were subjected to a second round of focus groups. The next section discusses the results of these groups conducted.

5.6 Second Focus Groups

The same two focus groups of four participants (Section 5.3) were subjected to the four prototypes developed (Section 5.5). The location of the focus groups remained the same and these were conducted in the Seminar Room at the NMMU South Campus Department of Computing Sciences (Section 5.3.1). The procedure of these focus groups differed slightly from the initial ones conducted as the participants were now presented with prototypes developed. Participants were given the freedom to experience how the prototypes functioned and then to decide which prototypes they preferred, which were to be incorporated into MotionShare. Participants were asked to give a rating out of 10 to indicate their satisfaction with each prototype.

Figure 5.14 shows the results of the second focus groups. ProtoTouch and ProtoPoint were rated the highest by the participants with mean ratings of 9.75 and 9.50 respectively. ProtoTilt had a mean rating of 7.88 among the participants. ProtoFileGesture (rating of 7.5) was rated the lowest by the participants because participants were accustomed to the traditional method of selecting files, which was a long press to initiate the file selection process, followed by single touch (Section 5.3.3). Although the decision was made to implement this traditional method of file selection, the novel idea of multiple file selection was still presented to participants to determine if their opinions would change. ProtoTouch and ProtoPoint were integrated to form ProtoGesture, based on the results of the second focus groups. The next section discusses ProtoGesture.

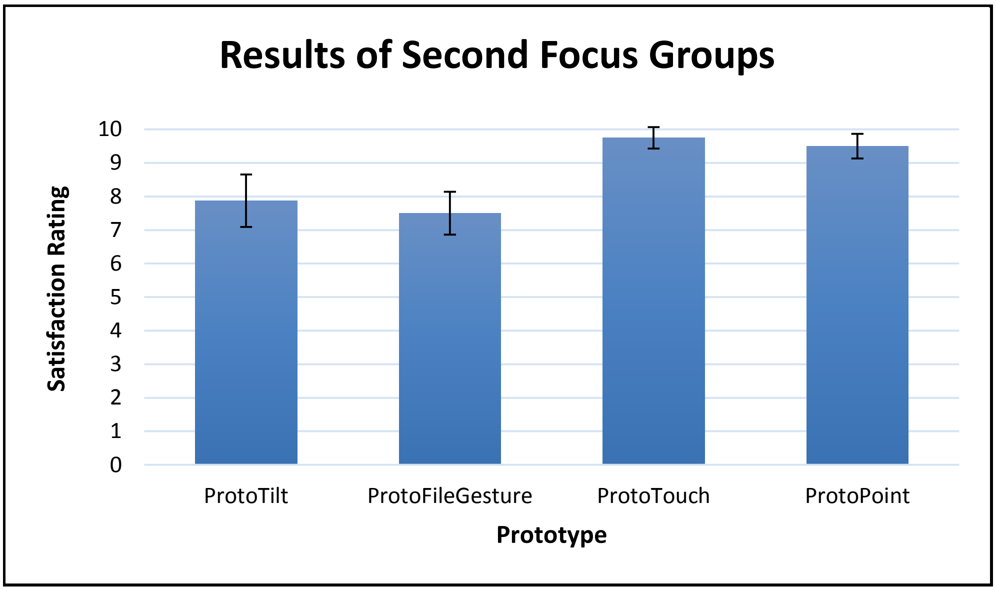
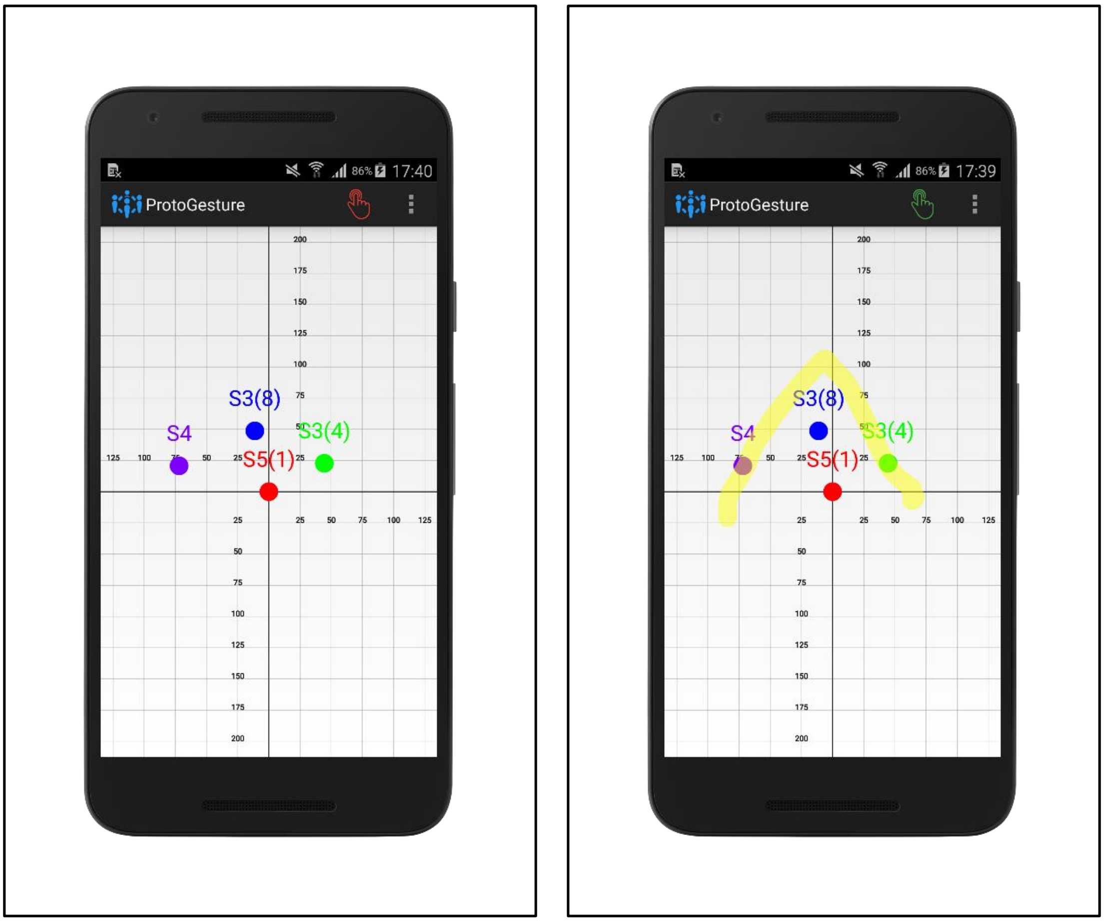


Figure 5.14: Results of Second Focus Groups

5.7 ProtoGesture

ProtoGesture incorporates both ProtoTouch and ProtoPoint. By default, the user can use the point gesture to share the selected file(s). The touch gesture can only be used when the user selects the hand gesture icon located on the top right. The red hand means the touch gesture is off (Figure 5.15(a)) and the point gesture is on (Figure 5.15(b)). The green hand means the touch gesture is on and the point gesture is off. When the green hand is displayed, the user can highlight the device dots to share the selected file(s) and the sharing process is initiated when the user lifts his finger up from the device screen (Figure 5.15(b)). ProtoGesture was the last prototype integrated into MotionShare.



(a) (b)

Figure 5.15: Screenshots of ProtoGesture

5.8 Conclusion

This chapter followed the third activity, ***Design and Develop Artefact***, within the Design Cycle of the DSR methodology (Section 1.7.3.1). The focus groups conducted and prototyping were used to address two research questions (Section 1.7.1) of this chapter:

> *"RQ4. How should NUI interaction techniques be designed to support information sharing among co-located mobile devices?*
>
> *RQ5. How can a proxemic prototype NUI be developed to support information sharing among co-located mobile devices?"*

Two focus groups were conducted to determine which gestures would be accepted by the participants. The results of the focus groups were aggregated and compared to the existing systems reviewed in the literature study chapters (Chapters 2 and 3). A decision was made on which gestures should be designed and implemented, which was justified by the comparison between the results of the focus group and existing systems. The selected gestures came in the form of the prototypes designed, namely ProtoTilt, ProtoFileGesture, ProtoTouch, and ProtoPoint. ProtoTouch and ProtoPoint were integrated into ProtoGesture, which was integrated into MotionShare. These two gestures, namely touch and point, were used for information sharing because they received the highest ratings (mean ratings of 9.75 and 9.50 respectively).

This chapter achieved two research objectives (Section 1.7.2):

> *"RO_4. To determine how NUI interactions techniques can be designed to support information sharing among co-located mobile devices.*
>
> *RO_5. To develop a proxemic prototype NUI to support information sharing among co-located mobile devices."*

These research objectives were achieved through the multiple prototypes developed and selection of the appropriate NUI interaction techniques based on the focus groups conducted and the review of existing systems from literature. The following chapter reports on the results of the evaluations conducted, which addresses the sixth research question (Section 1.7.1) and satisfies the sixth research objective (Section 1.7.2).

Chapter 6: Evaluation

6.1 Introduction

In Chapter 4, the positioning and communication design was presented. The gesture design integrated into MotionShare was discussed in Chapter 5. This chapter addresses the fourth and fifth activities in the DSR methodology (Section 1.7.3.1). The fourth activity, ***Demonstrate Artefact***, is performed within the Design Cycle and the fifth activity, ***Evaluate Artefact***, is performed within the Rigor Cycle (Figure 6.1). This chapter addresses the sixth research question (Section 1.7.1) identified:

> *"RQ6. How accurate and usable is the proxemic prototype NUI in supporting information sharing among co-located mobile devices?"*

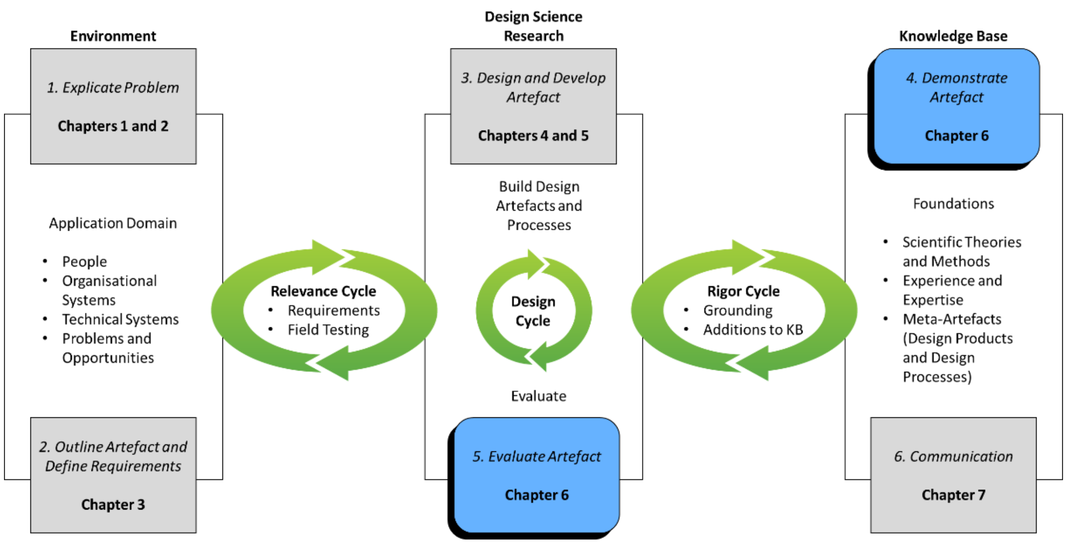


Figure 6.1: Chapter 6 Position in the Adapted DSR Methodology

This chapter discusses the research design and results of the usability evaluation conducted. MotionShare was evaluated to determine the accuracy and usability of the gestures implemented. The first evaluation was conducted to address the first evaluation objective, namely the accuracy of the distance indicator and gestures implemented, by measuring the results against specific metrics. A preliminary usability evaluation is described, which was used as an initial pilot study prior to the final usability evaluation. The results of the usability evaluation were analysed and presented, potential issues were identified, and appropriate changes to MotionShare were made to address these issues. Lastly, the design implication and recommendations conclude this chapter. Figure 6.2 presents the structure of this chapter.

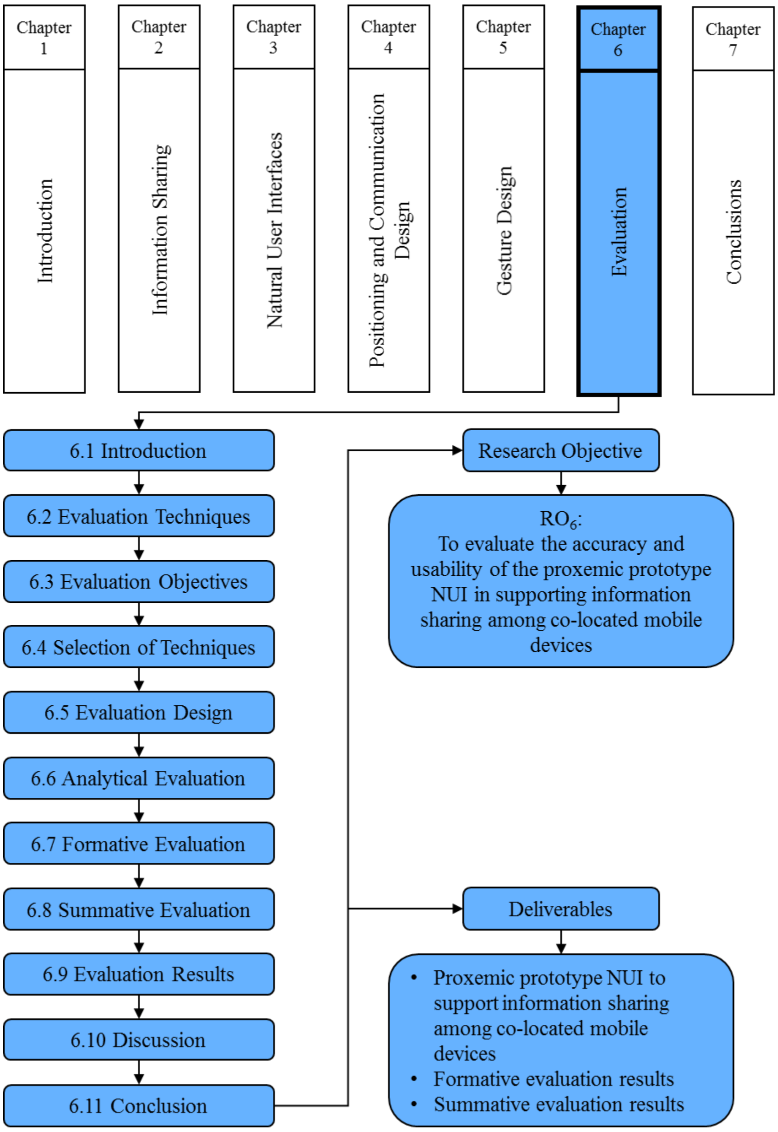


Figure 6.2: Chapter 6 Structure

6.2 Evaluation Techniques

To answer the sixth research question (Section 1.7.1), the correct evaluation techniques needed to be selected in order to evaluate the design artefact. Evaluation serves to provide evidence of a design artefact developed in the DSR methodology, which is functional and has achieved the purpose for which it was designed. Without proper evaluation, outcomes from the DSR methodology are in essence unsubstantiated assertions that the artefact achieves its design purpose.

Evaluating DSR artefacts is considered essential and is well documented in literature (Cleven, Gubler, & Hüner, 2009; Peffers et al., 2012). Rigorous evaluation methods need to be conducted for the utility, quality, and efficacy of a design artefact to be presented (Venable, Pries-Heje, & Baskerville, 2012).

Venable et al. (2012) mentions that scientific research requires evidence which is provided through rigorous evaluation. DSR literature (Johannesson & Perjons, 2012; Venable et al., 2012) characterises evaluation techniques along two dimensions: *ex ante* vs. *ex post* and *naturalistic* vs. *artificial*.

Existing DSR literature discusses a variety of techniques available to evaluate a design artefact (Ostrowski & Helfert, 2012); however, not all are appropriate for evaluating MotionShare. An investigation into DSR literature provided greater insight and understanding into which techniques should be selected for the purposes of evaluating MotionShare.

Johannesson and Perjons (2012) discusses the first dimension, namely ex ante and ex post evaluation. Ex ante evaluation, often called informed argument, is the approach of evaluating the artefact without it being used (evaluation based on design specifications), while ex post evaluation requires the artefact to be used (evaluation to occur after artefact implementation). An ex ante evaluation typically involves conducting interviews, whereby experts are involved to express their views on the artefact. The experts' views are based on the general knowledge and experience of similar artefacts and their applications of use. This type of evaluation strategy evaluates the artefact based on the researchers' reasoning and arguments presented. The researcher must present sufficient evidence regarding the artefact's ability to satisfy the defined requirements and to solve the explicated problem (the first two activities of the DSR methodology identified in Section 1.7.3.1). Ex ante evaluation is typically used when evaluating highly innovative artefacts. Ex post evaluation is used to evaluate the artefact's usability and determine whether it met the requirements initially identified.

Venable (2006) classify evaluation into two techniques, namely artificial and naturalistic (second dimension). An artificial evaluation technique involves evaluating the artefact in a rigorously controlled, simulated environment. The methods typically used in an artificial evaluation include:

- Criteria-based analysis;
- Theoretical arguments;
- Mathematical proofs;
- Field experiments;
- Laboratory experiments; and
- Simulations.

Naturalistic evaluation technique is where the artefact is placed in its real environment and tested on its performance. This technique is always empirical and includes several methods:

- Surveys;
- Action research;
- Field studies; and
- Case studies.

DSR literature (Hevner et al., 2004; Johannesson & Perjons, 2014) identifies several techniques available for evaluating the design artefact in the DSR methodology. These techniques are summarised in Table 6.1.

Table 6.1: Design Evaluation Techniques (Hevner et al., 2004; Johannesson & Perjons, 2014)

Techniques	Description
Observational	• Case Study: In depth examination of the artefact in a specific environment • Field Study: Monitor the artefact usage in multiple projects
Analytical	• Static Analysis: Examination of the artefact structure to identify static qualities such as complexity • Architecture Analysis: Examination of the artefact fitting into a technical architecture • Optimisation: Demonstrate inherent optimal properties of the artefact or provide optimality bounds on artefact behaviour • Dynamic Analysis: Examination of the artefact to identify dynamic qualities such as performance
Experimental	• Controlled Experiment: Examination of the artefact in a controlled environment to identify qualities such as usability • Simulation: Execution of the artefact using artificial data
Testing	• Functional (Black Box) Testing: Execution of the artefact interfaces to identify defects and failures • Structural (White Box) Testing: During the artefact implementation, perform coverage testing of a metric such as execution paths
Descriptive	• Informed Argument: Information usage from the knowledge base, such as relevant research, to construct a convincing argument for the artefact's utility • Scenarios: Demonstration of artefact's utility through well-designed scenarios

Petter et al. (2010) discusses the benefits of evaluation to be threefold:

1. Evaluation is essential to prove the artefact's feasibility and contributes to the existing body of knowledge (Vaishnavi & Kuechler, 2008).
2. Evaluation provides feedback (Hevner et al., 2004), which is used to identify the following:
    a. Whether the problem is well understood;
    b. If assumptions made are appropriate;
    c. If the process of the design artefact is acceptable; and
    d. If there are improvements to be made to the artefact.
3. Upon completion of the evaluation, social science research approaches can be used to theorise and explain why the artefact is functional or non-functional in a specific environment (March & Smith, 1995).

Formative evaluation is a type of usability evaluation used to help the system design process by evaluating the system during the development cycle. This evaluation is typically performed iteratively with the objective of identifying and eliminating usability issues (Flagg, 2013). A summative evaluation is only performed when the design is completed. The summative evaluation is used to determine the success level of the system design in meeting a set of quantitative (performance and/or satisfaction) objectives (Stufflebeam & Coryn, 2014). All the evaluation techniques presented in Table 6.1 can be used in both formative and summative evaluations, depending on the research objectives.

6.3 Evaluation Objectives

MotionShare is a mobile application designed to support information sharing among co-located mobile devices. This application utilises a set of NUI interaction techniques, which allow the user to share information more intuitively than using existing methods. The primary objective of this evaluation was to determine the accuracy and usability when using MotionShare. In doing so, the evaluation was to answer the sixth research question (Section 1.7.1):

> *"How accurate and usable is the proxemic prototype NUI in supporting information sharing among co-located mobile devices?"*

Furthermore, the evaluation served to identify any potential issues with the design of MotionShare.

6.4 Selection of Techniques

The selection of evaluation techniques was based on aligning the evaluation objectives (Section 6.3) with the objectives of the fourth and fifth activities (***Demonstrate Artefact*** and ***Evaluate Artefact*** respectively) of the DSR methodology. Existing DSR literature outlined various techniques available, as discussed in Section 6.2. This research provided the theoretical background and knowledge required to select the appropriate techniques to evaluate MotionShare.

A combination of techniques and methods were used to evaluate MotionShare. Evaluating the calculated positions of the devices and the usability of the gestures used in MotionShare were considered important. Therefore, ex post evaluation (Section 6.2) was used to evaluate these two aspects of MotionShare with an analytical technique (Table 6.1). The analytical technique examines the accuracy levels of device positioning and gesture performance. Ex post evaluation (Section 6.2) was also used to determine the usability of MotionShare with an experimental technique (Table 6.1), which involved the examination of MotionShare in a controlled environment.

6.5 Evaluation Design

DSR literature (Cleven et al., 2009; Hevner et al., 2004; March & Smith, 1995; Vaishnavi & Kuechler, 2008; Venable, 2006) identifies four main design artefact types created by the DSR methodology:

- ***Constructs:*** vocabulary and symbols;
- ***Models:*** abstractions and representations;
- ***Methods:*** algorithms and practices; and
- ***Instantiations:*** implemented and prototype systems.

MotionShare is classified as an instantiation artefact type because it operationalises constructs, models, and methods (March & Smith, 1995). Evaluation of MotionShare is important as it addresses the fifth activity of the DSR methodology (Section 1.7.3.1), namely ***Evaluate Artefact***. Trochim (2006) define *evaluation* as:

> *"Evaluation is the systematic acquisition and assessment of information to provide useful feedback about the artefact."*

The evaluation design consisted of an ex post evaluation with two of the techniques identified being applied to MotionShare, namely analytical and experimental. The analytical evaluation technique was used to assess the accuracy of the positioning and gestures, whereas the experimental evaluation technique was used to determine the usability of the MotionShare (Section 6.4). The experimental evaluation technique is broken up into two types of evaluations conducted with MotionShare, namely formative (Section 6.7) and summative (Section 6.8). The evaluation design structure and process is illustrated in Figure 6.3. This figure reflects on the rigorous evaluation procedure performed and the iterative nature identified by DSR literature (Cleven et al., 2009; Peffers et al., 2012) as a core activity of the DSR methodology.

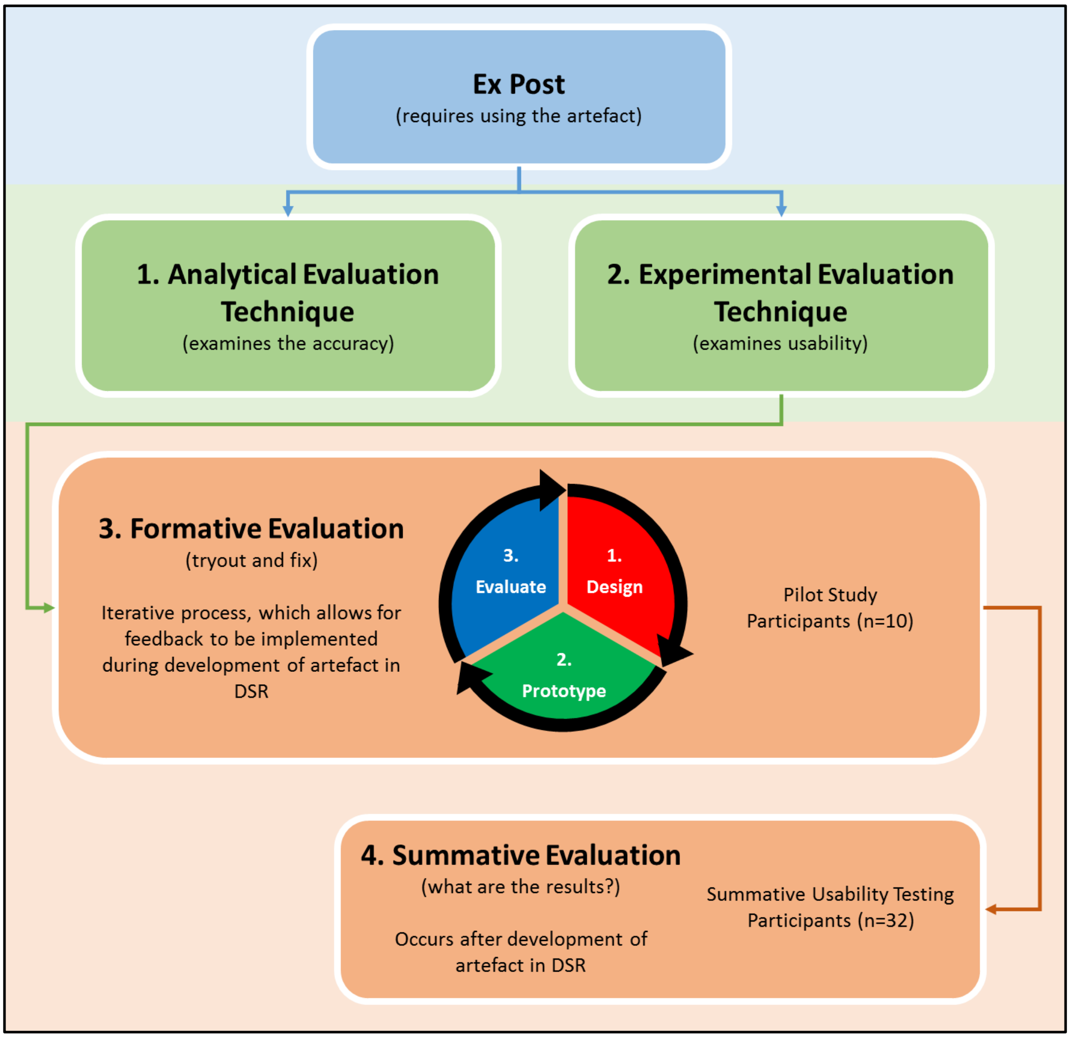


Figure 6.3: Evaluation Design Structure

6.6 Analytical Evaluation

An analytical evaluation technique was selected as the most suitable technique for assessing the accuracy and precision of device positioning and gesture performance for MotionShare. Therefore, this analytical evaluation had two objectives, namely device positioning and MotionShare gestures.

6.6.1 Device Positioning

The positioning of devices in an environment is an essential component of MotionShare. The information relating to devices' positions affects the behaviour and responsiveness of MotionShare. It is therefore important to assess the level of accuracy and precision of this information with an analytical evaluation.

6.6.2 MotionShare Gestures

The two types of gestures implemented in MotionShare, namely point and touch gestures, depend on the calculated information of the devices' positions. The accuracy of the MotionShare gestures are also important because they can potentially influence the usability and user experience of MotionShare. As such, the MotionShare gestures were also subjected to an analytical evaluation.

6.6.3 Metrics

The metrics used in the analytical evaluation were based on the two objectives identified, namely device positioning (Section 6.6.1) and MotionShare gestures (Section 6.6.2). The International Organisation for Standardisation (ISO) 5725-1 standard (International Organisation for Standardisation, 1994) define *accuracy* as:

> *"Accuracy is the closeness of agreement between a test result or measurement result and the accepted reference value or true value. When the term is applied to sets of measurements of the same measurand, it involves a component of random error and a component of systematic error. Thus, trueness and precision are used."*

This definition is interpreted as the level of measurement yielding true (no systematic errors) and consistent (no random error) results. Furthermore, MathsIsFun.com (2015) corroborates this definition with their own, more simplified definition:

> *"Accuracy is how close a measured value is to the actual (true) value."*

Figure 6.4 shows how the accuracy concept encompasses trueness and precision. In this figure, reference value represents the true value. This figure is substantiated by the ISO 5725-1 standard (International Organisation for Standardisation, 1994), which states that *"accuracy consists of trueness (proximity of measurement results to the true value) and precision (repeatability or reproducibility of the measurement)"*. Therefore, the metrics used in this evaluation were precision and trueness.

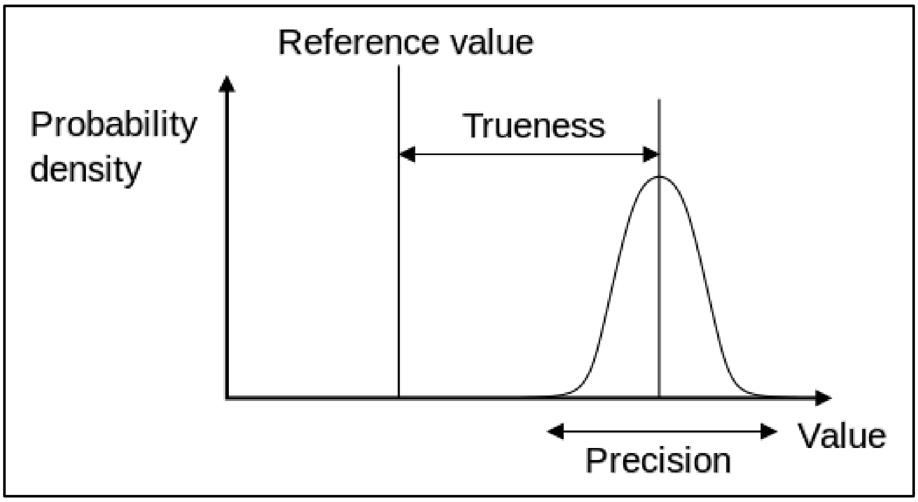


Figure 6.4: Accuracy Components (ASTM, 2011)

Sokolova and Lapalme (2009) identifies the equations used in the analytical evaluation to measure three metrics, namely precision, trueness, and recall. Each of these equations contain similar symbols:

- tp is the number of true positives;
- tn is the number of true negatives;
- fp is the number of false positives; and
- fn is the number of false negatives.

Sokolova and Lapalme (2009) explains the above terms as follows:

- ***True positives*** are the number of positive class examples correctly classified as positive;
- ***True negatives*** are the number of negative class examples correctly classified as negative;

- ***False positives*** are the number of positive class examples incorrectly classified as negative; and
- ***False negatives*** are the number of negative class examples incorrectly classified as positive.

These terms are better understood using the confusion matrix presented in Figure 6.5.

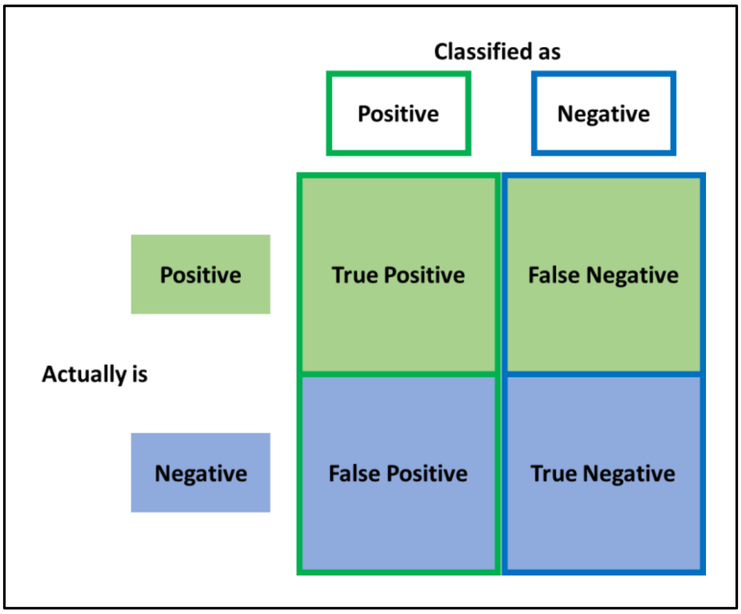


Figure 6.5: Confusion Matrix Adapted from Witten and Frank (2005)

6.6.3.1 Precision

The ISO 5725-1 standard define *precision* as the *"closeness of agreement among a set of results"* (International Organisation for Standardisation, 1994). ASTM (2011) define precision as the level of measurement yielding consistent results obtained under stipulated scenarios, namely repeatability (precision within the same scenario) and reproducibility (precision in different scenarios), as shown in Figure 6.4. Both of these precision definitions can be simplified by using MathsIsFun.com (2015), which interprets it as *"how close the measured values are to each other"*. Precision (Sokolova & Lapalme, 2009) is determined with Equation 6.1.

$$Precision = \frac{tp}{tp + fp} \times 100 \tag{6.1}$$

6.6.3.2 Trueness

The ISO 5725-1 standard define *trueness* as the *"closeness of agreement between the arithmetic mean of a large number of test results and the accepted reference value"* (International Organisation for Standardisation, 1994). A definition that is easier to understand is from ASTM (2011), who defines trueness as the *"closeness of agreement between the average value obtained from a large series of test results and the true value"*, as shown in Figure 6.4. Trueness (Sokolova & Lapalme, 2009) is determined with Equation 6.2.

$$Trueness = \frac{tp + tn}{tp + tn + fp + fn} \times 100 \tag{6.2}$$

6.6.3.3 Recall

Recall is typically referred to as *Sensitivity* or *True Positive Rate* (TPR), which focuses on the effectiveness of a classifier to correctly classify test results as positive values depending on the positive condition (Sokolova & Lapalme, 2009). Recall (Sokolova & Lapalme, 2009) is determined with Equation 6.3.

$$Recall = \frac{tp}{tp + fn} \times 100 \tag{6.3}$$

6.6.4 Procedure

The analytical evaluation technique was applied to both device positioning and MotionShare gestures. Bearing in mind that MotionShare was designed to facilitate information sharing among mobile devices in a co-located environment, it was not dependent on a specific location. As such, it can potentially be used in almost any environment depending on how adversely the sensor information is affected by external variables. These variables are absorption, interference, and/or diffraction (Section 4.6.2.3) and have the same influence on the embedded sensors. However, the analytical evaluation technique was applied to the co-located mobile devices in the same environment, which provided the least adversity levels. The same environment allowed the precision, trueness, and recall metrics to be effectively measured and compared with various scenarios for device positioning (Section 6.6.1) and MotionShare gestures (Section 6.6.2).

The mobile devices were placed at different distance increments in various combinational layouts to simulate a co-located environment whereby individuals would gather or meet. These scenarios allowed for the analytical evaluation technique to assess the accuracy and precision of the first objective, which is device positioning. Similarly, the procedure was also repeated for MotionShare gestures, which was the second objective of this evaluation.

6.6.5 Results

Section 6.6.3 discussed accuracy as comprising of two metrics, namely precision and trueness, which were used in the analytical evaluation. The results of the evaluation are discussed according to device positioning and the MotionShare gestures with the two metrics identified.

6.6.5.1 Device Positioning Results

The results of the analytical evaluation according to precision and trueness for device positioning of MotionShare are discussed. Table 6.2 shows the results of device positioning. Several mobile devices were placed at known different distance increments, which were used as the true values. The distance values calculated by MotionShare for every device were recorded and used as the predicted values. These predicted values were classified either as true positive, true negative, false positive, or false negative.

Table 6.2: Results of Device Positioning (n=8000)

		Predicted Values							
		25	**50**	**75**	**100**	**125**	**150**	**175**	**200**
True Values	**25**	956	44	0	0	0	0	0	0
	50	93	867	26	14	0	0	0	0
	75	0	123	812	23	14	28	0	0
	100	0	34	29	563	264	13	19	78
	125	0	0	52	228	529	137	43	11
	150	0	0	16	114	138	711	14	7
	175	0	0	0	16	34	248	655	47
	200	0	0	0	17	29	101	216	637

Table 6.3 presents the individual precision, trueness and recall results at the different distance increments in percentages. Equation 6.1 was used to measure precision as a percentage. The 25cm distance had the highest precision of 91.13% among the various tested distances, which suggests that MotionShare was precise at this distance. The lowest precision was 52.48% for the 125cm distance, which suggests that the external variables affected the Bluetooth RSSI. This effect during the 125cm distance is evident in Table 6.2, where MotionShare classified 125cm as 100cm for 228 instances and 150cm for 137 instances. The precision of MotionShare deviated significantly from 100cm to 175cm (52.48% ≤ precision ≤ 69.17%). MotionShare had a mean precision of 72.21%.

The trueness metric was measured as a percentage value using Equation 6.2. The highest trueness of MotionShare was at 25cm distance (98.29%) and closely followed by 50cm with 97.54%. The lowest trueness was 175cm with 78.23%. As the remaining distances were in a higher trueness range (88.13% ≤ trueness ≤ 96.11%), MotionShare demonstrated its overall effectiveness in classifying the distances among mobile devices. The average trueness of MotionShare was 91.39%.

The third metric measured for device positioning was recall. Equation 6.3 was used to measure recall as a percentage value. The highest recall was at 25cm with 95.60%. The lowest recall was at 125cm with 52.90%. The effectiveness of device positioning in MotionShare showed it struggled at this distance and also at 100cm (56.30%), which was interesting as the further distances of 150cm (71.10%), 175cm (65.50%), and 200cm (63.70%), all had significantly higher recall. The lower recall values could be attributed to the external variables affecting the Bluetooth RSSI, which caused MotionShare to misclassify the distances among devices. The mean recall for device positioning was 71.63%.

Table 6.3: Precision, Trueness, and Recall Results of Device Positioning (n=8000)

Distance (cm)	Precision (%)	Trueness (%)	Recall (%)
25	91.13	98.29	95.60
50	81.18	97.54	86.70
75	86.84	96.11	81.20

100	57.74	89.39	56.30
125	52.48	88.13	52.90
150	57.43	89.80	71.10
175	69.17	78.23	65.50
200	81.67	93.68	63.70
Mean	**72.21**	**91.39**	**71.63**

6.6.5.2 MotionShare Gestures Results

The results of the analytical evaluation according to recall for MotionShare gestures are discussed. Table 6.4 presents the results of MotionShare Gestures. Mobile devices were placed at known different distance increments. Equation 6.3 was used to measure recall as a percentage value. The touch gestures had the highest mean recall value (100.00%) and the point gestures had a mean value of 90.50% (n=362). The point and touch gestures both experienced no recall issues with the 25cm, 50cm, and 75cm distance increments as the recall was 100.00% (n=50 at each distance for each gesture). Point gestures had a recall of 92.00% (n=46) at the 100cm distance and 86.00% (n=43) for the 125cm distance. Performing the point gesture when devices were located 150cm apart presented a few issues with 88.00% recall (n=44). Devices at distance 175cm had a recall of 82.00% (n=41) when performing a point gesture. MotionShare struggled to recognise the orientation of the device at the 200cm distance with a recall of 76.00% (n=38). This suggested that the devices were placed in areas of the environment where the embedded sensors were more adversely affected. The touch gestures had a 100.00% recall at all the tested distance increments (n=400) because the gesture involved the highlighting of the device dots on the screen. This gesture was not affected by the external variables other than when computing the devices' positions. Point gestures had a mean recall value of 90.50% (n=362) because the devices were placed at different distances, which led to devices being placed in areas of the environment where external variables could more adversely affect the sensors. Point gesture is heavily dependent on the sensors when the device is oriented in the direction of the receiving device. If MotionShare does not detect the motion, then the device is not updated and the device points are no longer displayed correctly.

Table 6.4: Results of MotionShare Gestures (n=400)

Distance (cm)	25	50	75	100	125	150	175	200	Mean
Point Gesture (%)	100.00	100.00	100.00	92.00	86.00	88.00	82.00	76.00	90.50
Touch Gesture (%)	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00

6.6.5.3 Discussion

Device positioning and MotionShare gestures were subjected to an analytical evaluation. Device positioning was measured according to the precision, trueness, and recall metrics. MotionShare gestures, namely point and touch, were measured according to the recall metric. The procedure for both device positioning and MotionShare gestures was similar. Devices were placed at different distance increments ranging from 25cm to 200cm distance. In device positioning, the overall mean for precision, trueness, and recall was 72.21%, 91.39%, and 71.63% respectively. These values showed the adverse effect of external variables on MotionShare's device positioning throughout the various increasing distance increments. Both MotionShare gestures were also analytically evaluated. The point gesture had a mean value of 90.50%, whereas the touch gesture had a mean value of 100.00%. These mean values showed that the point gesture was dependent on the embedded sensors to detect orientation of the devices. Touch gesture was not dependent on sensors to be performed. MotionShare detected the touch gesture on the device screen and responded accordingly.

6.7 Formative Evaluation

A formative evaluation was conducted during the development stages of the artefact. This evaluation was iteratively performed with the primary objective of detecting and eliminating any potential usability issues (Flagg, 2013; User Experience Professionals' Association, 2012a). Formative evaluations can be categorised into several classifications (Evaluation Toolbox, 2010). *Proactive* and *Clarificative* are typically referred to as ex ante evaluations (Section 6.2), whereas

Interactive and *Monitoring* are ex post evaluations (Section 6.2). These classifications are summarised in Table 6.5.

Table 6.5: Formative Evaluation Classifications (Evaluation Toolbox, 2010)

Classification	Stage	Objective	Example
Proactive	Pre-Project Development	Comprehension or clarification on the necessity of the project	• Literature Review • Stakeholder Analysis • Problem or Solution Tree Analysis
Clarificative	Project Development	Establish theory of change on which project is constructed	• Logframe Matrix • Program Logic
Interactive	Project Implementation	Continual improvement on project design throughout implementation	• Informal Interview • Expert Reviews • Focus Group • Project Diary
Monitoring	Project Implementation	Ensure efficient and effective delivery of project activities	• Budget Tracking • Time Tracking • Questionnaire • Dartboard • Observation

Literature reviews (Chapters 2 and 3) were conducted to establish the theoretical foundation for the project requirements of this research (*Proactive*). Several focus groups (*Interactive*) were conducted to identify the most applicable gestures to be implemented into the tasks of MotionShare. The formative evaluation consisted of a pilot study, which served as preliminary testing to the summative evaluation.

Payne (2015) define a *pilot study* as a research study conducted before the formative evaluation commences. This study provides potentially valuable insight into missing elements of the procedure. Van Teijlingen and Hundley (2001) identifies four primary reasons for a pilot study:

1. Pilot study allows preliminary testing of the research hypotheses, which leads to the summative evaluation testing being performed on more precise hypotheses. This is because existing hypotheses could change or new hypotheses be presented based on the findings of the pilot study.

2. Pilot study provides the researcher with insight into potential concepts and approaches, which may not have been foreseen before this study, thereby potentially improving the results of the summative evaluation.
3. Pilot study ensures the statistical and analytical procedures for evaluating the data are correct before the summative evaluation is conducted.
4. Pilot study provides sufficient data to determine if the artefact and procedure is ready for the summative evaluation to be conducted.

These reasons formed the objectives of the pilot study, which was conducted. Participants of the pilot study are discussed in the next section.

6.7.1 Participants

Participants were selected based on purposive sampling to identify a representative sample, which represented the same population used in the summative evaluation and also best enabled the researcher to address the research questions (Maxwell, 2012). A total of 10 participants (eight males and two females) were involved in the evaluation because the ideal number of participants was identified to be in the region of 10 – 12, as previous studies have shown that this number range detects 80.00% of the possible usability issues of the evaluated system (Tullis & Albert, 2013). All the participants were experienced mobile device users, most of whom have used mobile devices for 5-6 hours on a daily basis for more than seven years. Figure 6.6 shows the user profile of mobile device types and age distribution.

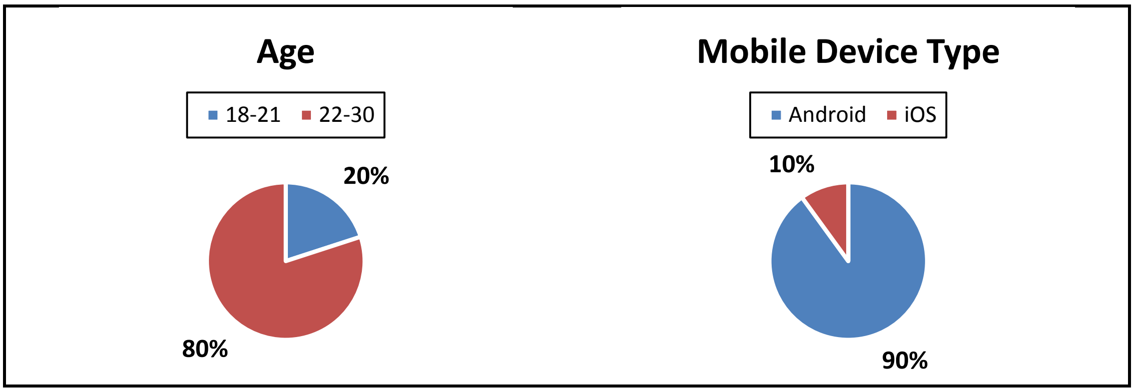


Figure 6.6: Pilot Study Participant Demographics (n=10)

6.7.2 Evaluation Metrics

The ISO 9241-11 standard (Mifsud, 2015) define *usability* as:

> *"The extent to which an application can be used by individuals to achieve specified objectives according to metric classifications, such as* ***effectiveness, efficiency, and satisfaction,*** *in a specified context of use."*

6.7.2.1 Effectiveness

Effectiveness is widely regarded to be the fundamental usability metric (Sergeev, 2010a). This metric can be measured according to the participants' ability to complete the task at hand, which is used to determine the task completion rate. To determine the task completion rate, each participant is assigned a binary value of "1" for task success or "0" for task failure. Therefore, task completion is a simple metric to be computed and easy to understand. Effectiveness (Sergeev, 2010a) can be determined with Equation 6.4, where:

- N is the number of tasks;
- R is the number of participants; and
- n_{ij} is the result of whether task i by participant j is successfully completed (n_{ij}=1) or failed to complete (n_{ij}=0).

$$Effectiveness = \frac{\sum_{j=1}^{R}\sum_{i=1}^{N} n_{ij}}{RN} \times 100\% \tag{6.4}$$

6.7.2.2 Efficiency

Participants' time taken to complete each task was recorded and used to measure efficiency (Tullis & Albert, 2013). The time taken to complete a task can be computed by using the start time and subtracting it from the end time as shown in Equation 6.5.

$$Task\ Time = End\ Time - Start\ Time \tag{6.5}$$

The overall relative efficiency involves the ratio of the time taken by participants who have completed the task in relation to the total time taken by all participants. Therefore, efficiency (Sergeev, 2010b) can be determined with Equation 6.6, where:

- N is the number of tasks;
- R is the number of participants;
- n_{ij} is the result of whether task i by participant j is successfully completed (n_{ij}=1) or failed to complete (n_{ij}=0); and
- t_{ij} is the time spent by participant j to complete task i even if the task is not successfully completed, in which case time is recorded until the participant has decided to quit the task.

$$Overall\ Relative\ Efficiency = \frac{\sum_{j=1}^{R}\sum_{i=1}^{N} n_{ij}t_{ij}}{\sum_{j=1}^{R}\sum_{i=1}^{N} t_{ij}} \times 100\% \qquad (6.6)$$

6.7.2.3 Satisfaction

The level of user acceptance and comfort experienced by participants was measured by providing participants with the Post-Study System Usability Questionnaire (PSSUQ), which is designed to perform an overall assessment of a system at the end of a usability evaluation (Tullis & Albert, 2013). The PSSUQ was selected for the following reasons:

- The PSSUQ is a standardised questionnaire designed to measure the users' perceived satisfaction, which is important to this research;
- Standardised questionnaires contain a specific set of statements, presented in a specified order using a specified format with specific rules for producing scores based on the respondents' answers;
- The PSSUQ has undergone psychometric qualification, including assessment of reliability, validity, and sensitivity;
- The PSSUQ allows for findings to be reported accurately and in detail using standardised metrics; and
- The PSSUQ statements can be subjected to factor analysis.

The PSSUQ consists of 19 statements and was analysed by calculating a mean value for each statement on the 7-point Likert-type scale of *"Strongly Disagree"* to *"Strongly Agree"* with *"N/A"* as option if necessary. The 7-point Likert-type scale was selected over the 5-point Likert-type scale because Lewis (2002) performed related analysis and identified *"the mean of 7-point scales correlated more strongly* *than the mean different of 5-point scales with the observed significance levels of t-tests"*. Furthermore, Lewis (2002) mention that existing versions of PSSUQ rather use a 7-point Likert-type scale as it allows for improved item-level and scale-level comparisons than 5-point Likert-type scales. According to the consulted statisticians, the mean scores of the 7-point Likert-type scale are classified according to the following ranges:

- ***Disagree*** $[1.00 \leq \mu < 3.57)$
- ***Neutral*** $[3.57 \leq \mu \leq 4.42]$
- ***Agree*** $(4.42 < \mu \leq 7.00]$

Furthermore, the mean satisfaction (Sergeev, 2010c) is determined with Equation 6.7, where:

- R is the number of participant responses;
- Q is the number of statements in the PSSUQ; and
- p_{ij} is the weight of the answer (1-7) for scenario i and participant j.

$$Average\ Satisfaction = \frac{\sum_{j=1}^{R}\sum_{i=1}^{Q}\frac{p_{ij}}{7}}{QR} \times 100\% \qquad (6.7)$$

6.7.3 Evaluation Instruments

The pilot study generated data which was collected and analysed to ensure that the statistical and analytical procedures were correct before the summative evaluation. The instruments used were based on the metrics identified in Section 6.7.2. Effectiveness was measured based on the participants' answers to the questions presented in the list of user tasks provided by the moderator to be completed by each participant (Appendix G). Participants were also observed to see if they experienced any difficulties in completing the list of user tasks as well as to determine whether they understood the instructions provided and the information displayed by MotionShare. Efficiency was measured based on the time taken to complete each task on the task list. Satisfaction was measured based on the use of a standardised user satisfaction questionnaire, which was provided to participants at the end of the evaluation.

This questionnaire was based on the PSSUQ, which is typically used to evaluate a system's usability (Tullis & Albert, 2013). This questionnaire (Appendix H) was modified to include two additional sections to the standard PSSUQ, namely biographical information and overview. The overview section of the questionnaire involved open-ended questions about the most positive aspect(s) and most negative aspect(s) of the system, as well as any additional comment(s) or recommendation(s). Participants were also required to complete a post-test questionnaire. This questionnaire was used to determine the type of gestures that the participants preferred, by measuring their satisfaction with the implemented MotionShare gestures.

6.7.4 User Tasks

Participants of the pilot study received a list of user tasks, which was used in the summative evaluation. The objective was to establish whether the tasks included in the list would evaluate every aspect of MotionShare, which was mapped according to the functional requirements thereof (Section 4.5). Each task included a question, which participants were required to answer. Table 6.6 presents the list of tasks.

Table 6.6: Summary of Pilot Study User Tasks

Task Moniker	Functionality
Single Point	Selecting a single file to be shared Share with a single user (point gesture)
Multiple Point	Selecting multiple files to be shared Share with multiple users (point gesture)
Single Touch	Selecting a single file to be shared Share with a single user (touch gesture)
Multiple Touch	Selecting multiple files to be shared Share with multiple users (touch gesture)
Receive	Receiving file(s)
Receive	Receiving file(s)
Receive	Receiving file(s)

6.7.5 Evaluation Procedure

The participants of the pilot study were identified in Section 6.7.1. Participants evaluated MotionShare as a group of five because the sixth research objective was *"to evaluate the accuracy and usability of the proxemic prototype NUI in supporting information sharing among co-located mobile devices"*. The evaluations occurred in the same environment, so the results could be compared and analysed. This procedure was almost identical to the summative evaluation procedure.

The moderator of the evaluation provided verbal information to the participants, which was the same as the summative evaluation (Appendix F). The smartphones were placed randomly around the table to simulate a co-located environment. A demonstration of MotionShare's calibration phase was shown and a list of user tasks was provided to the participants. The evaluation commenced when the participants invoked MotionShare on their assigned smartphones.

When the user tasks were completed by all the participants within the group, the participants were presented with a PSSUQ (Appendix H) to complete. A post-test questionnaire was also presented for participants to complete (Appendix I).

6.7.6 Results

The results of the evaluation were measured according to three metrics (Section 6.7.2), namely effectiveness, efficiency, and satisfaction.

Effectiveness

The answers obtained from the questions linked to the list of user tasks showed MotionShare to be effective in supporting participants in completing their tasks (Table 6.7). Almost all the participants' answers were correct; however, the two areas of concern related to sharing with a single user (point gesture) and receiving file(s). MotionShare did not detect the change in device orientation. Therefore, the share with a single user using a point gesture did not work. Similarly, the file was not received as the user accidently closed MotionShare, before the file share was complete.

Table 6.7: Pilot Study Effectiveness Results (n=10)

Task Moniker	Effectiveness (%)
Single Point	90.00
Multiple Point	100.00
Single Touch	100.00
Multiple Touch	100.00
Receive	100.00
Receive	90.00
Receive	100.00
Mean	97.14

Efficiency

The mean time taken per task was calculated which allowed for the fastest and slowest times to be identified (Figure 6.7). A comparison between the two NUI interaction techniques was conducted to see which interaction technique was more efficient. Ninety-five percent confidence intervals are shown for the mean values. The single point gesture had a mean of 46.60 seconds and the multiple point gesture had a mean of 44.60 seconds. The single touch gesture had a mean of 34.50 seconds and the multiple touch gesture had a mean of 32.20 seconds. Therefore, the point gesture required more time than the touch gesture.

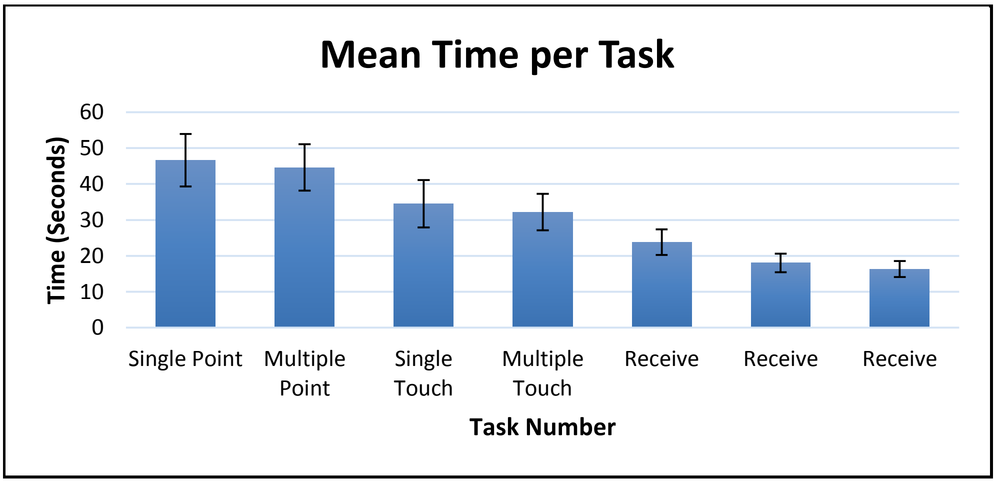


Figure 6.7: Pilot Study Mean Time per Task (n=10)

Satisfaction

The raw PSSUQ results of the pilot study with statistical analysis are included in Appendix J. Figure 6.8 shows the results from the PSSUQ classified according to factor analysis, which includes system use, information quality, interface quality, and overall. Ninety-five percent confidence intervals are shown for the mean values.

High user satisfaction was reported by the participants in all four PSSUQ categories. Mean ($6.31 \leq \mu \leq 6.45$) and median ($6.50 \leq \text{median} \leq 7.00$) ratings for all four categories were at least six. The participants experienced similar satisfaction levels as the standard deviation was less than one ($0.59 \leq \sigma \leq 0.86$). Considering the pilot study objectives (Section 6.7), these values meant that the MotionShare was ready for the summative evaluation. These values also provided evidence to support the benefits of designing a proxemic prototype NUI to support information sharing among co-located mobile devices.

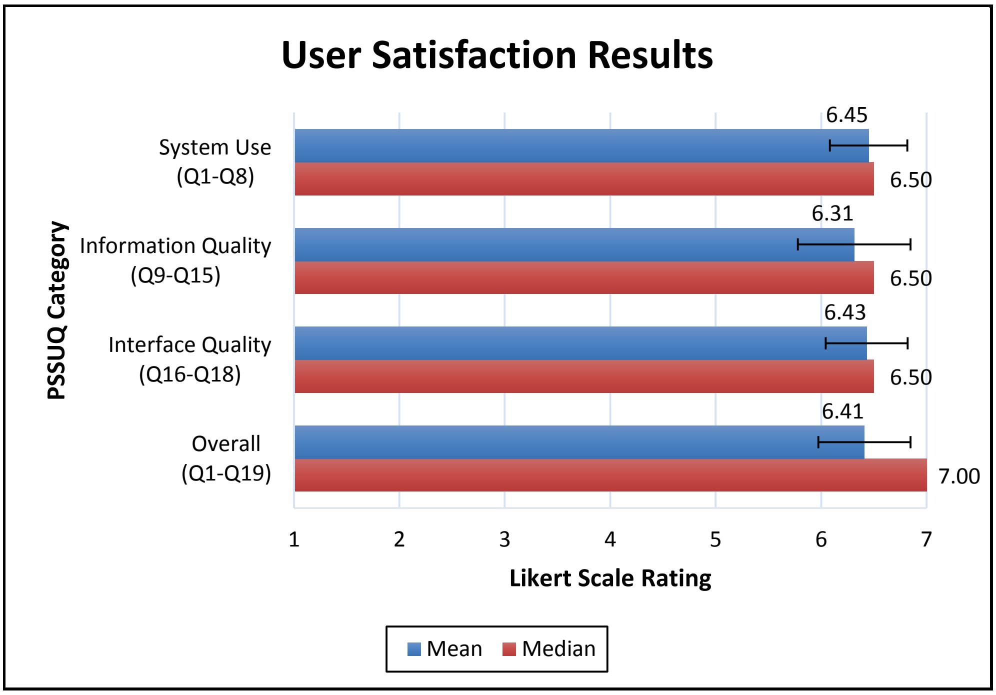


Figure 6.8: Pilot Study PSSUQ Results Classified by Factor Analysis (n=10)

Table 6.8 shows the results of the PSSUQ statements classified under the *System Use* category. This category related specifically to the system usability of MotionShare. The statistical functions performed allow for quantitative data to be collected and analysed. Furthermore, the statistical procedure was thoroughly tested to ensure it was correct, which was used in the summative evaluation.

Table 6.8: Pilot Study Results for System Use (n=10)

PSSUQ Statement	Mean	Classification	Median	Mode	Standard Deviation
1. Overall, I am satisfied with how easy it is to use this system	6.70	Agree	7.00	7.00	0.48
2. It was simple to use this system	6.60	Agree	7.00	7.00	0.52
3. I could effectively complete the tasks and scenarios using this system	6.10	Agree	6.00	6.00	0.74
4. I was able to complete the tasks and scenarios quickly using this system	6.40	Agree	6.00	6.00	0.52
5. I was able to efficiently complete the tasks and scenarios using this system	6.50	Agree	6.50	7.00	0.53
6. I felt comfortable using this system	6.40	Agree	6.00	6.00	0.52
7. It was easy to learn to use this system	6.70	Agree	7.00	7.00	0.48
8. I believe I could become productive quickly using this system	6.20	Agree	6.00	6.00	0.79

Participants were satisfied with the overall usage of the system (statement one) with their responses classified as *Agree* (μ=6.70). The simplicity in using MotionShare (statement two) was also rated favourably with a mean value of 6.60, which classified the participants' responses as *Agree*. Participants rated the learnability of MotionShare to be easy as their responses were classified as *Agree* (μ=6.70). Only MotionShare's effectiveness in completing the list of tasks had the lowest mean value (μ=6.10), which still classified the participants' responses as *Agree* (statement three). The remaining statements had mean values exceeding six ($6.20 \leq \mu \leq 6.50$), which meant they were all classified as *Agree*. The median ($6.00 \leq$ median ≤ 7.00) and mode ($6.00 \leq$ mode ≤ 7.00) values of the *System Use* statements were all above six, which meant that the participants' responses were classified as *Agree*. The low standard deviations ($0.48 \leq \sigma \leq 0.79$), which were all below one, suggested that all the participants felt the same way about the usability of MotionShare.

The second PSSUQ category (*Information Quality*) included a different set of statements (9-15), where Table 6.9 contains the results of these statements. Similarly, statistical and analytical procedures performed on *System Use* are repeated here. Seven of the eight statements had mean values exceeding six ($6.10 \leq \mu \leq 6.80$) with only one exception. Statement nine had the lowest mean value ($\mu=5.10$). General comments and user observations suggest that this was due to the simplistic design of MotionShare and very few opportunities for errors. The median ($5.00 \leq \text{median} \leq 7.00$) and mode ($4.00 \leq \text{mode} \leq 7.00$) values were both favourable (values exceeding six), which showed the participants' responses as *Agree* with the corresponding statements. Only statement nine was excluded from containing favourable median and mode values because of the general comments received and the user behaviour observed. Statement nine had a median value of five and mode value of four, which showed participants to *slightly agree* and *neutral* respectively on the 7-point Likert-type scale. The statements classified under *Information Quality* received similar ratings from the participants, where the standard deviations were less than one ($0.42 \leq \sigma \leq 1.10$). The largest standard deviation occurred in statement nine ($\sigma=1.10$).

Table 6.9: Pilot Study Results for Information Quality (n=10)

PSSUQ Statement	Mean	Classification	Median	Mode	Standard Deviation
9. The system gave error messages that clearly told me to fix problems	5.10	Agree	5.00	4.00	1.10
10. Whenever I made a mistake using the system, I could recover easily and quickly	6.30	Agree	6.50	7.00	0.95
11. The information, such as online help, on-screen messages, and other documentation, provided with this system was clear	6.80	Agree	7.00	7.00	0.42
12. It was easy to find the information I needed	6.50	Agree	6.50	7.00	0.53
13. The information provided for the system was easy to understand	6.70	Agree	7.00	7.00	0.48
14. The information was effective in helping me complete the tasks and scenarios	6.10	Agree	6.00	6.00	0.57
15. The organisation of information on the system screens was clear	6.70	Agree	7.00	7.00	0.48

Interface Quality also yielded favourable mean values ($6.00 \leq \mu \leq 6.70$), whereby the participants' responses to all three statements were classified as *Agree* (Table 6.10). Participants liked using the system interface as the corresponding PSSUQ statement received the highest mean value (μ=6.70). The statement: *"This system has all the functions and capabilities I expect it to have"*, had a mean value of six, which suggested that the expectations of participants were satisfied as their response was classified as *Agree*. From the general comments received regarding this statement, participants expressed the need for additional functionality of MotionShare, which can be pursued as future research and was not deemed feasible within the constraints of this research. The median (6.00 ≤ median ≤ 7.00) and mode (6.00 ≤ mode ≤ 7.00) values also exceeded the value six, with a low standard deviation ($0.48 \leq \sigma \leq 0.67$) of less than one. This low standard deviation suggests that the participants' responses were in close proximity to one another.

Table 6.10: Pilot Study Results for Interface Quality (n=10)

PSSUQ Statement	Mean	Classification	Median	Mode	Standard Deviation
16. The interface of this system was pleasant	6.60	Agree	7.00	7.00	0.52
17. I liked using the interface of this system	6.70	Agree	7.00	7.00	0.48
18. This system has all the functions and capabilities I expect it to have	6.00	Agree	6.00	6.00	0.67

The general feedback received from the open-ended questions was positive. Positive comments covered a diverse range of topics related to MotionShare, which included usefulness, simplistic design, easy to learn, detailed instructions provided, smooth responsive nature of the Compass, and the aesthetic appeal of the Device Map. Negative comments were restricted to minor usability issues and questionnaire options provided. These negative comments served to provide suggestions to improve the usability of the existing design of MotionShare, evaluation instruments, and evaluation procedure. The feedback received was applied to the relevant areas of concern prior to the commencement of the summative evaluation. It was encouraging to receive the participants' responses towards the potential applications and ideas for expanding this research for future work.

6.8 Summative Evaluation

A summative evaluation was conducted to obtain measures to establish a usability benchmark and compare results with the usability requirements of a design artefact (Stufflebeam & Coryn, 2014; User Experience Professionals' Association, 2012b). These requirements are typically task-based and should correspond to the artefact's requirements. Summative evaluations are typically used to validate the set of objectives (task completion and time taken on a task) and subjective characteristic (user satisfaction) of the design artefact. Therefore, the primary objective of a summative evaluation is to evaluate the design artefact through these defined measures. As discussed and identified in Section 6.4, ex post evaluation was used with a controlled experimental technique.

6.8.1 Participants

The ideal number of participants was identified to be in the region of 10 – 12 (Section 6.7.1). Although the number of participants exceeded the ideal range, it has been proven that a larger sample size provides a better confidence level with the results as well as a closer estimate to the true population values (Tullis & Albert, 2013). Furthermore, this number of participants was selected due to the fact that the research was not limited to qualitative data, but also involved quantitative data. Quantitative data is primarily aimed at statistics and this research required statistical analysis. A sample size of 30 is the minimum number for statistical significance (Campbell, 2011). Therefore, 32 participants was selected for this research.

6.8.2 Evaluation Metrics

The same evaluation metrics were used in both the pilot study and usability evaluation. These metrics are summarised as follows (Joo, 2010):

- ***Effectiveness:*** *"Accuracy and completeness with which users achieve specified goals."*
- ***Efficiency:*** *"Resources expended in relation to the accuracy and completeness with which users achieve goals."*
- ***Satisfaction:*** *"Freedom from discomfort, and positive attitudes towards the use of the product."*

Therefore, these metrics were measured in the usability evaluation. The evaluation instruments used in the usability evaluation are discussed in the next section.

6.8.3 Evaluation Instruments

The usability evaluation generated data, which required it to be collected and analysed for the purposes of this research. The evaluation instruments used were based on the metrics identified in Section 6.8.2. Table 6.11 presents a summary of these instruments used in the usability evaluation. The instrument column lists the evaluation instrument used, the data analysis column describes the corresponding data analysis performed with the specified instrument, and classification of the data collected is shown under the classification column.

Table 6.11: Summary of the Usability Evaluation Instruments

<table>
<tr><th>Instrument</th><th>Data Analysis</th><th>Classification</th></tr>
<tr><td>Consent Form</td><td>No analysis</td><td rowspan="2">Demographic</td></tr>
<tr><td>Biographical questionnaire</td><td>Quantitative statistical analysis (frequency)</td></tr>
<tr><td>Written information given to participants</td><td rowspan="3">No analysis</td><td rowspan="3">Instructions</td></tr>
<tr><td>Verbal information given to participants</td></tr>
<tr><td>User Task List</td></tr>
<tr><td>User Task List</td><td>Quantitative statistical analysis (mean, median, mode, standard deviation)</td><td rowspan="3">Evaluation</td></tr>
<tr><td>Post-Study System Usability Questionnaire</td><td>Quantitative statistical analysis (frequency, mean, median, mode, standard deviation)
Qualitative thematic analysis</td></tr>
<tr><td>Post-Test Questionnaire</td><td>Quantitative statistical analysis (mean, median, mode, standard deviation)</td></tr>
</table>

6.8.4 User Tasks

Participants received a list of user tasks, which required them to evaluate the suitability of both the point and touch gestures for the information sharing process (Appendix G). Each participant group was instructed to work together to complete the task list because it comprised of two roles, namely sender and receiver. Sender tasks were the point and touch gestures. The receiver tasks were the receiving of files sent by the sender. During the session, the participant who assumed the role of the sender would verbally tell the other participant who was the receiver to ensure they were looking at their own device to complete the receiver task. Nine user tasks were presented to the participants, ensuring a comprehensive coverage of the entire functionality of MotionShare. Each task included a question, which participants answered by completing the necessary steps of the particular task. Table 6.12 summarises the list of tasks.

Table 6.12: Summary of User Tasks

Task Moniker	Functionality
Single Point	Selecting a single file to be shared Share with a single user (point gesture)
Multiple Point	Selecting multiple files to be shared Share with multiple users (point gesture)
Single Touch	Selecting a single file to be shared Share with a single user (touch gesture)
Multiple Touch	Selecting multiple files to be shared Share with multiple users (touch gesture)
Receive	Receiving file(s)
Receive	Receiving file(s)
Receive	Receiving file(s)

6.8.5 Evaluation Procedure

Participants of the usability evaluation were identified in Section 6.8.1. All participants received an email notification containing the written documentation detailing the usability evaluation (Appendix E). Participants evaluated MotionShare as a group of four because individual participant evaluation was inappropriate due to the fact that MotionShare was designed and implemented to primarily determine if a proxemic NUI can support information sharing among co-located mobile devices.

The usability evaluations occurred in the same environment as the pilot study, which was the moderator's home. The selection of the environment was as a result of external variables at NMMU influencing the sensor data being read by the embedded sensors in the smartphones, which was verified during numerous informal tests. These variables were absorption, interference, and/or diffraction (Section 4.6.2.3) and had the same influence on the smartphone sensors as with the Bluetooth RSSI. During each of the informal tests which were conducted in different NMMU venues, one or all smartphones indicated the wrong sensor data.

The moderator of the usability evaluation provided participants with a consent form, which required the participants' signatures to ensure that the participation was voluntary (Appendix D). The consent form included the ethical clearance number, which was H15-SCI-CSS-004 (Section 1.6). Additional information was verbally presented to the participants, which outlined the evaluation procedure (Appendix F) as well as a demonstration of the application's calibration phase. Participants were individually allocated smartphones in the controlled environment and were provided with a list of user tasks (Appendix G) to complete. Counterbalancing was used to eliminate the learnability effect by changing the order of the tasks in which the gestures were used. The moderator randomly placed the smartphones around a table to simulate a co-located environment and initialised MotionShare. Participants were given a few minutes to familiarise themselves with the application and ask any questions regarding the usability evaluation. Thereafter, the moderator commenced the evaluation and the entire session was video recorded.

Upon completion of the user tasks by all participants within the group, each participant was required to complete a PSSUQ (Appendix H), which was used to perform an overall assessment of MotionShare at the end of the usability evaluation (Tullis & Albert, 2013). Finally, the participants were required to complete an additional post-test questionnaire to determine which NUI interaction technique they preferred and why (Appendix I).

6.9 Evaluation Results

The results of the evaluation were categorised into demographics, performance metrics (effectiveness and efficiency), and satisfaction metrics (satisfaction).

6.9.1 Demographics

Figure 6.9 presents the aggregated demographic results of all the participants, which were obtained from the biographical questionnaire (Appendix H). The participants were all students (undergraduate or post-graduate) from the NMMU Department of Computing Sciences. Since the proxemic NUI was designed to operate on a mobile platform, participants were required to have prior experience with mobile devices, which formed part of the participant profiling. Purposive sampling was used to derive a sample, as this would best enable the researcher to address the research questions (Maxwell, 2012).

In Figure 6.9, the gender composition of participants was 78.00% (n=25) male and 22.00% (n=7) female. The age distribution among the participants showed 31.00% (n=10) and 69.00% (n=22) belonged to the 18-21 and 22-30 years categories respectively. The number of hours spent on mobile devices by participants per a day was evenly distributed as 13.00% (n=4) indicated 1-2 hours, 28.00% (n=9) indicated 3-4 hours, 34.00% (n=11) indicated 5-6 hours, and 25.00% (n=8) indicated 7+ hours. In terms of whether participants had used a similar type of application before, 72.00% (n=23) said *No*, 12.00% (n=4) said *Maybe*, and 16.00% (n=5) said *Yes*. To be noted, is that in a follow up interview, participants who had indicated *Yes* were referring to the use of the Gmail or the WhatsApp application.

All participants (n=32) were mobile device proficient with over seven years of experience in using mobile devices. Most of the participants (91.00%) indicated they would use MotionShare in the foreseeable future when indoor positioning techniques are improved to allow for NUI interaction techniques to be utilised in the proper context. MotionShare therefore demonstrated the potential applications of use with regard to knowing the positions of smartphones in an indoor environment, as evident by the overwhelming majority of participants (97.00%). Most of the participants were Android users (87.00%) and only 13.00% (n=4) were iOS users.

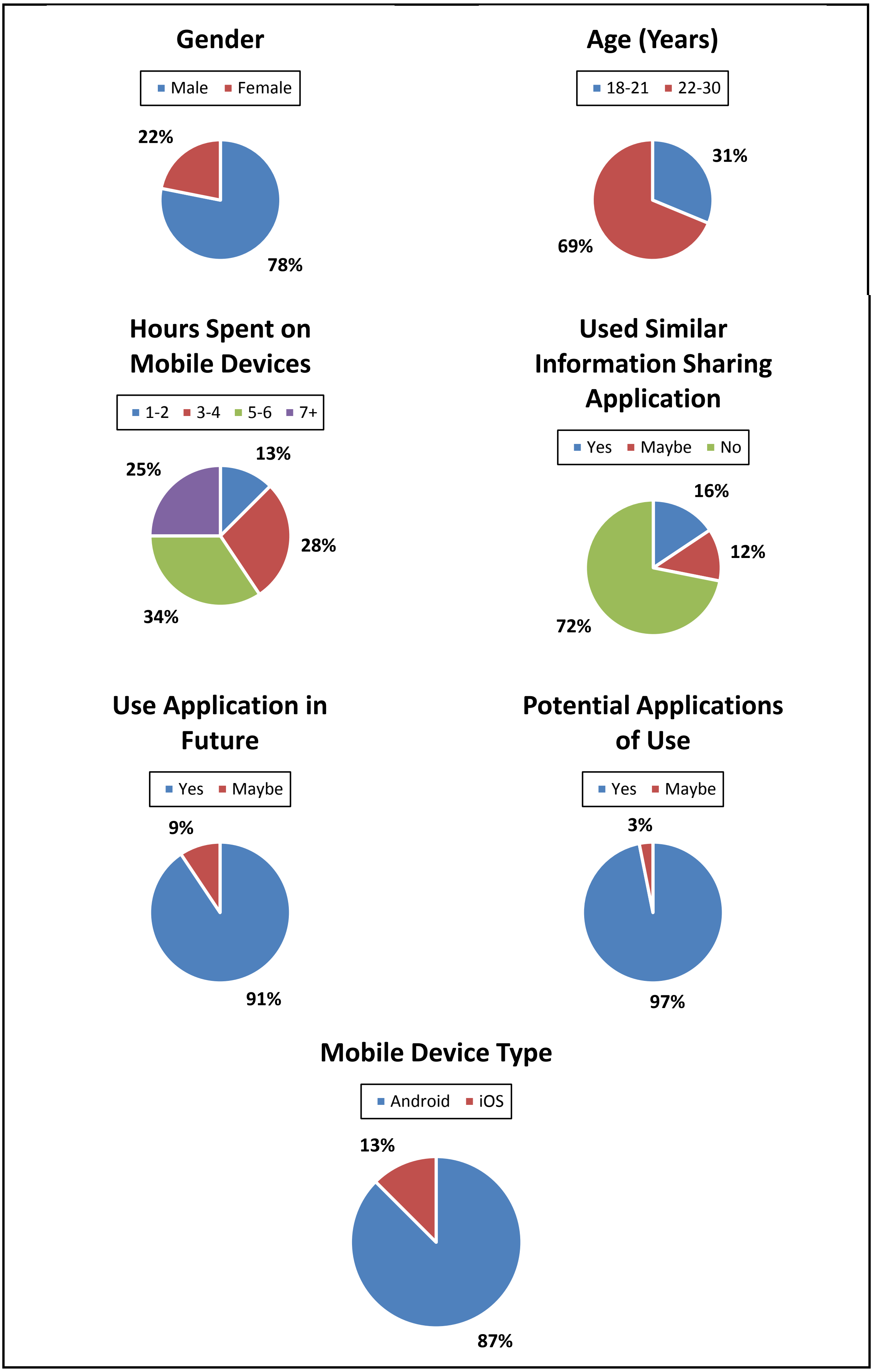


Figure 6.9: Participant Demographics (n = 32)

6.9.2 Performance Results

The results of the usability evaluation according to effectiveness and efficiency of the user tasks are discussed in this section.

6.9.2.1 Effectiveness

The effectiveness metric was discussed in Section 6.7.2 The moderator provided each participant with a list of user tasks, which required an answer to indicate task completion. Effectiveness was measured as a percentage value using Equation 6.4. Task completion can be calculated by means of two different methods: calculating the mean success rate of each task among all participants or calculating the mean success rate of each participant among all tasks. Figure 6.10 presents the individual task complete rate per task.

Almost all the tasks had 100.00% success, with the exception of Single Point and Receive. Single Point gesture had 97.50% because the device for one of the participants did not detect the point gesture performed. Receive had 95.00% which was a result of an odd occurrence of the files not showing up in the list of files on the receiver's side.

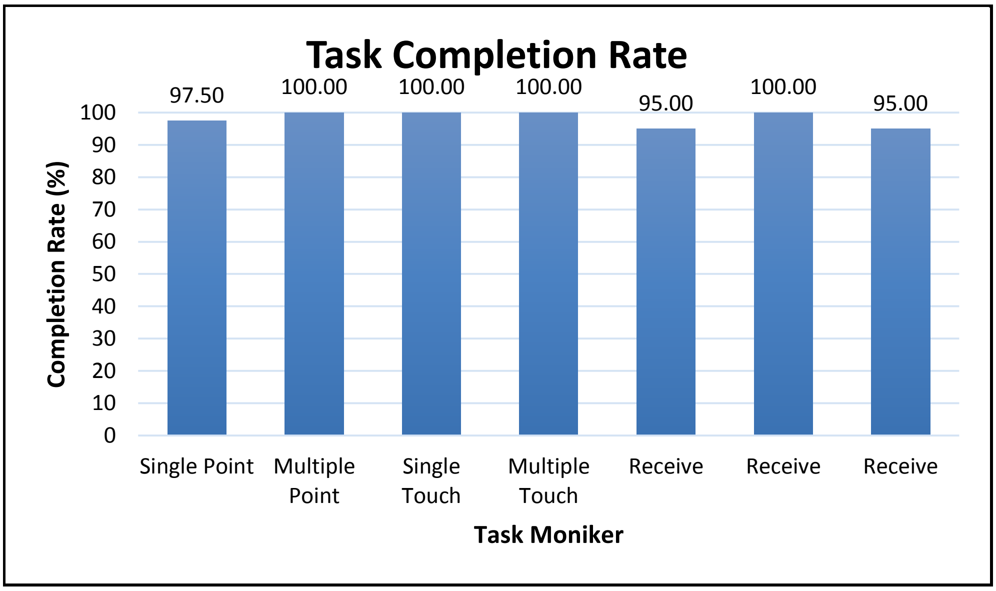


Figure 6.10: Task Completion Rate of Participants (n=32)

6.9.2.2 Efficiency

The time taken for each task was determined by using Equation 6.5. The time taken by the participant when reading the task instruction list and answering the task question was included in the task duration. Figure 6.11 shows the mean task times and 95.00% confidence intervals. The mean task times enabled easy identification of the tasks which participants struggled to complete. These tasks were investigated and evaluated further to identify potential usability issues with the application. Table 6.13 summarises the statistical analysis performed on the task times in seconds.

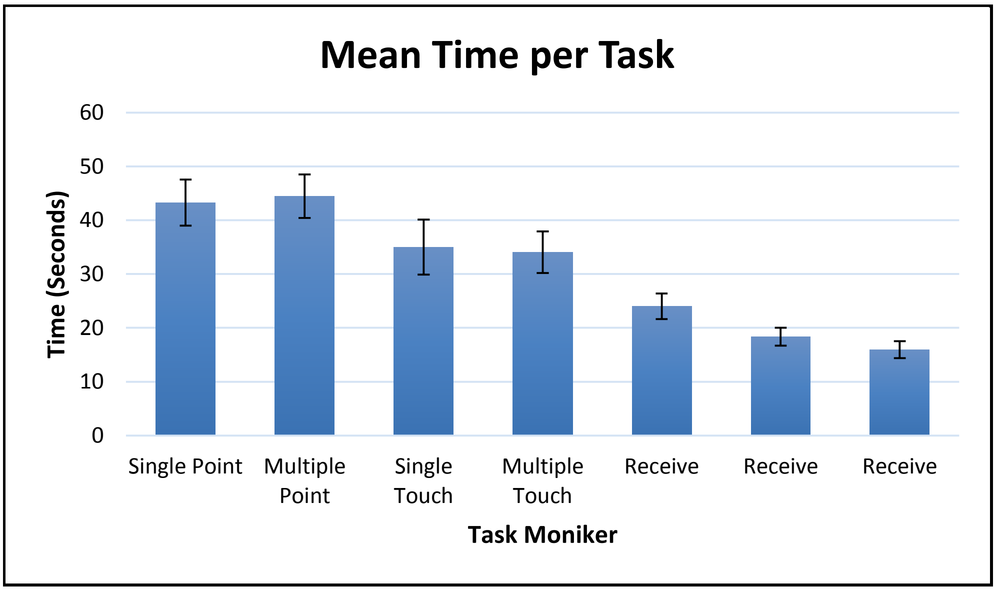


Figure 6.11: Mean Time per Task (n=32)

Table 6.13: Statistical Analysis for Tasks in Seconds (n=32)

Task Moniker	Mean	Median	Mode	Standard Deviation	Standard Error	Minimum	Maximum
Single Point	43.30	44.00	49.00	9.03	2.02	29.00	61.00
Multiple Point	44.45	46.00	48.00	8.48	1.90	27.00	55.00
Single Touch	35.00	34.50	45.00	10.82	2.42	19.00	58.00
Multiple Touch	34.05	32.50	30.00	8.51	1.90	20.00	52.00
Receive	24.00	24.00	27.00	5.40	1.21	15.00	34.00
Receive	18.35	19.50	20.00	3.76	0.08	12.00	25.00
Receive	15.95	15.50	15.00	3.66	0.82	10.00	24.00

6.9.3 Satisfaction Results

The user satisfaction results obtained from using the PSSUQ are discussed in this section (Appendix H). The PSSUQ scale scores were computed based on factor analysis and were classified into the following categories (Frughling & Lee, 2005; Herrero, Panetto, Meersman, & Dillon, 2012; Lewis, 1995):

- ***System Use:*** Mean of the participants' scores in statements 1-8.
- ***Information Quality:*** Mean of the participants' scores in statements 9-15.
- ***Interface Quality:*** Mean of the participants' scores in statements 16-18.
- ***Overall:*** Mean of the participants' scores in statements 1-19.

The raw PSSUQ results of the summative evaluation are included in Appendix K. Figure 6.12 presents the feedback from the PSSUQ classified according to these categories. Ninety-five percent confidence intervals are indicated for the mean values. Seven-point Likert-type scales were used in the quantitative sections of the PSSUQ, with the antonyms appearing at each end of this numerical range. This allows participants the option to select various points along the continuum of the designated word pairing, with "*Strongly Disagree*" and "*Strongly Agree*" as anchor points (Boyle & Schmierbach, 2015).

Various statistical functions were applied, namely mean (μ) and median, for each PSSUQ category (Figure 6.12). Ninety-five percent confidence intervals are indicated for the mean values. The mean of *System Use* for the evaluated application was 5.95 (on the 7-point Likert-type scale). The mean values of the *Interface Quality* and *Overall* categories were also high with 6.27 and 6.00 respectively. This suggested that the participants found the application to have an intuitive design and overall experience to be pleasant. The information presented by the application (*Information Quality*) had the least favourable mean (μ=5.90), which suggested the information provided by the application was insufficient or that there are potentially better methods of visually presenting the information. General comments and observations suggested this was a result of minor usability issues. The median value for all PSSUQ categories were the same (median=6.00). The standard deviations of these categories were all low ($0.79 \leq \sigma \leq 1.11$), which indicated that the participants' responses were in close proximity to the respective calculated means.

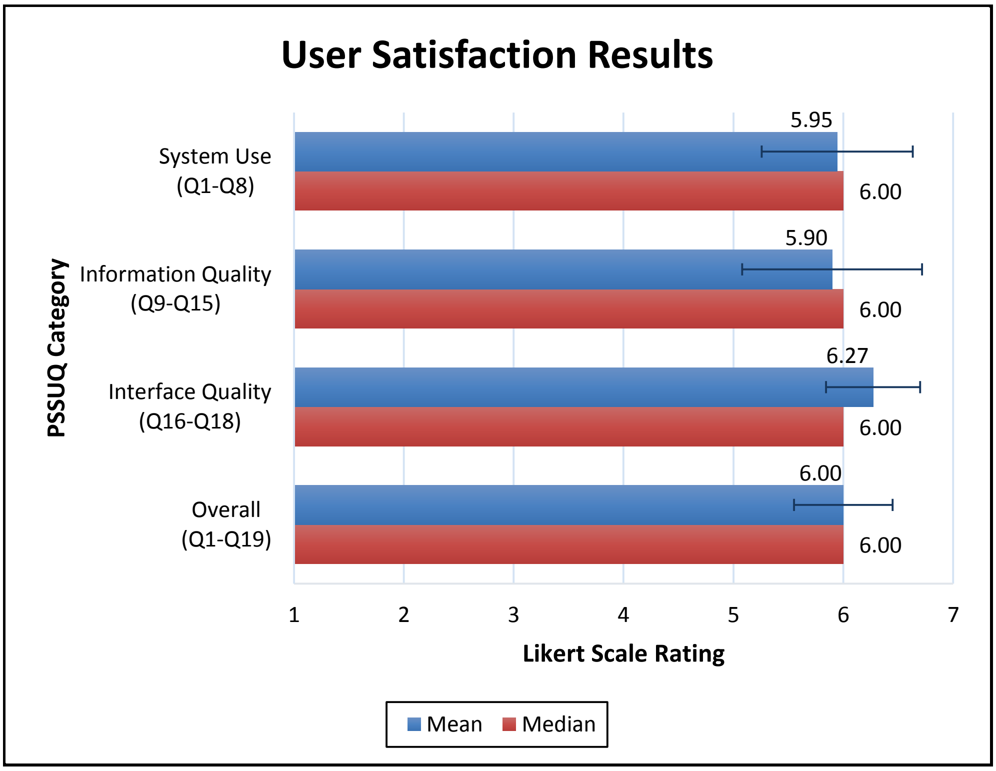


Figure 6.12: PSSUQ Results Classified by Factor Analysis (n=32)

The participants' questionnaires were aggregated and classified according to these categories. A popular technique for visualising the four PSSUQ categories is a radar chart (Tullis & Albert, 2013). In Figure 6.13, the shape illustrated in this chart indicates participants thought the application was well-designed in terms of the four PSSUQ categories. All four categories received high ratings. The category with the highest rating by participants was *Interface Quality* with a rating of 90.00%. The *Overall* and *System Use* categories were rated by participants with ratings of 86.00% and 85.00% respectively. *Information Quality* was rated the lowest by participants with a rating of 84.00%. According to Sauro and Lewis (2012), the rating for the *Overall* category (86.00%) indicated MotionShare's usability to be very good.

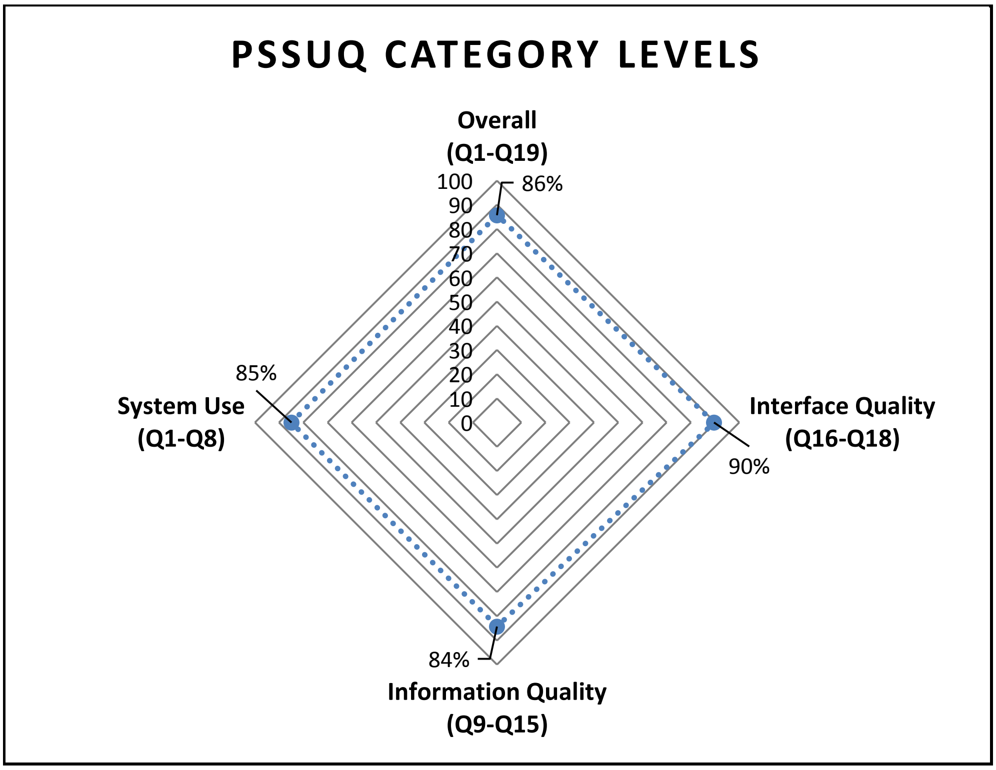


Figure 6.13: Participants' PSSUQ Ratings Based on Factor Analysis (n=32)

Cronbach's alpha (α) is a numerical coefficient of internal consistency, which means that it measures how closely a set of variables are as an individual group. Cronbach's alpha is also used to compute the average correlation of statements in a psychometric test, like the PSSUQ, to determine the reliability level (Tavakol & Dennick, 2011). For this research, Cronbach's alpha was computed to examine the internal consistency of the PSSUQ. George and Mallery (2013) discussed how the internal consistency using Cronbach's alpha should be interpreted, which is presented in Table 6.14.

Table 6.14: Cronbach's Alpha Interpretation (George & Mallery, 2013)

Cronbach's Alpha	Interpretation
α ≥ 0.9	Excellent consistency
0.9 > α ≥ 0.8	Good consistency
0.8 > α ≥ 0.7	Acceptable consistency
0.7 > α ≥ 0.6	Questionable consistency
0.6 > α ≥ 0.5	Poor consistency
0.5 > α	Unacceptable consistency

Cronbach's alpha was computed with Equation 6.8 for each PSSUQ category, namely *System Use*, *Information Quality*, *Interface Quality*, and *Overall*, where:

- K is the number of statements in the category
- $\sigma_{Y_i}^2$ is the variance of component i for the current sample
- σ_X^2 is the variance of the observed total scores

$$\alpha = \frac{K}{K-1}\left(1 - \frac{\sum_{i=1}^{K} \sigma_{Y_i}^2}{\sigma_X^2}\right) \tag{6.8}$$

Table 6.15 shows the Cronbach's alpha value with the corresponding interpretation for the four PSSUQ categories.

Table 6.15: Cronbach's Alpha on PSSUQ Categories (n=32)

PSSUQ Category	Cronbach's Alpha	Interpretation
System Use	0.88	Good consistency
Information Quality	0.85	Good consistency
Interface Quality	0.79	Acceptable consistency
Overall	0.92	Excellent consistency

Table 6.15 indicates the internal consistency of the PSSUQ was excellent as *Overall* had excellent consistency (α=0.92). The PSSUQ categories, namely *System Use* and

Information Quality both had good consistency (α=0.88 and 0.85 respectively). *Interface Quality* has the lowest consistency among the PSSUQ categories with acceptable consistency (α=0.79). Therefore, all the results derived from the PSSUQ can be considered to be reliable, as proven by Cronbach's alpha.

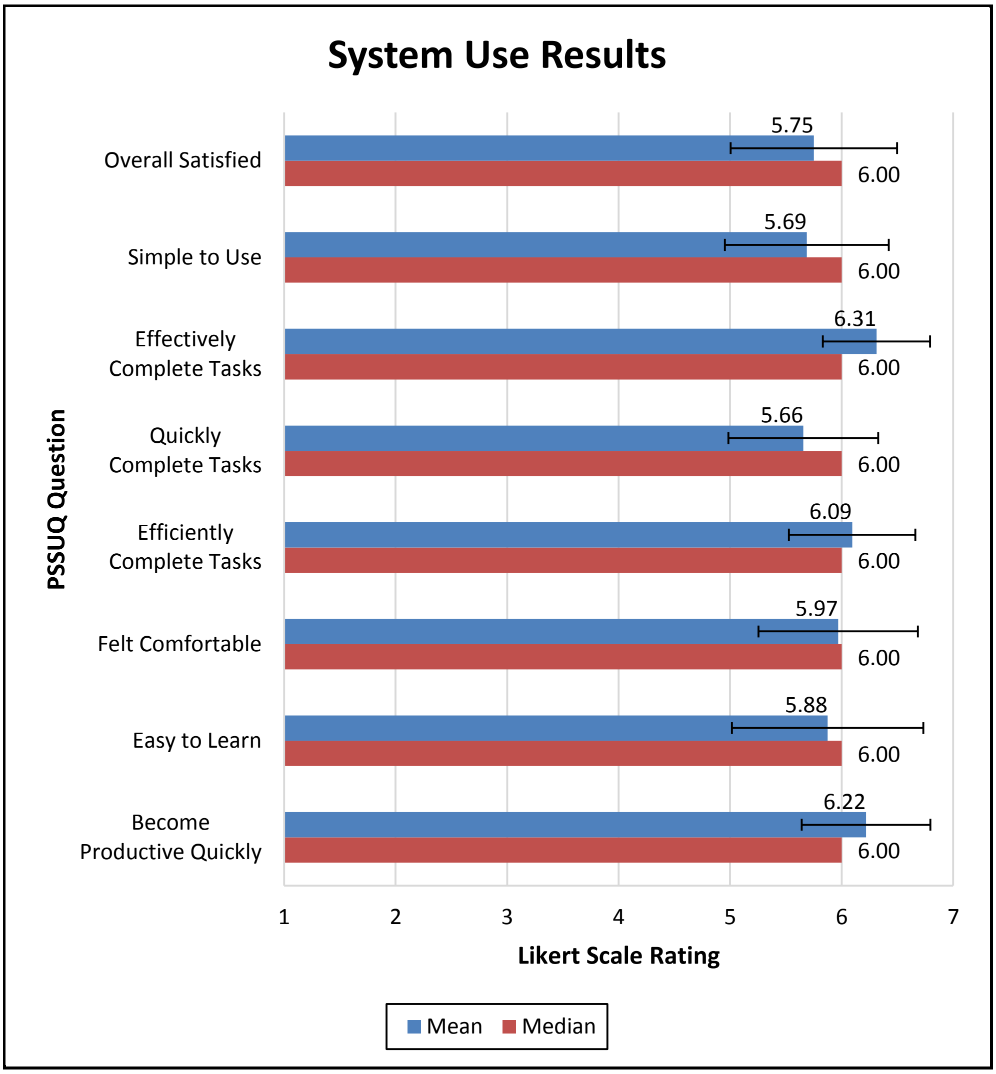


Figure 6.14: Statistical Analysis for System Use (n=32)

Figure 6.14 presents the statistical functions performed on the various PSSUQ statements classified under the *System Use*, which were identified earlier. These functions, namely mean, median, and standard deviation, allow for quantitative data to be collected and analysed. Ninety-five percent confidence intervals are indicated for the mean values. The lowest mean (μ=5.66) was received for the statement: *"I was able to complete the tasks and scenarios quickly using this system"*. Although the value is acceptable and above the average rating of four (7-

point Likert-type scale), it still suggests that MotionShare contains some minor usability issues.

MotionShare was expected to be usable as it utilises NUI interaction techniques, which is evident from the high mean, median, and mode values for the majority of the statements (rated between five and six). These statements had low standard deviations within the range of $0.69 \leq \sigma \leq 1.24$, which indicated that participants' experiences and responses were similar.

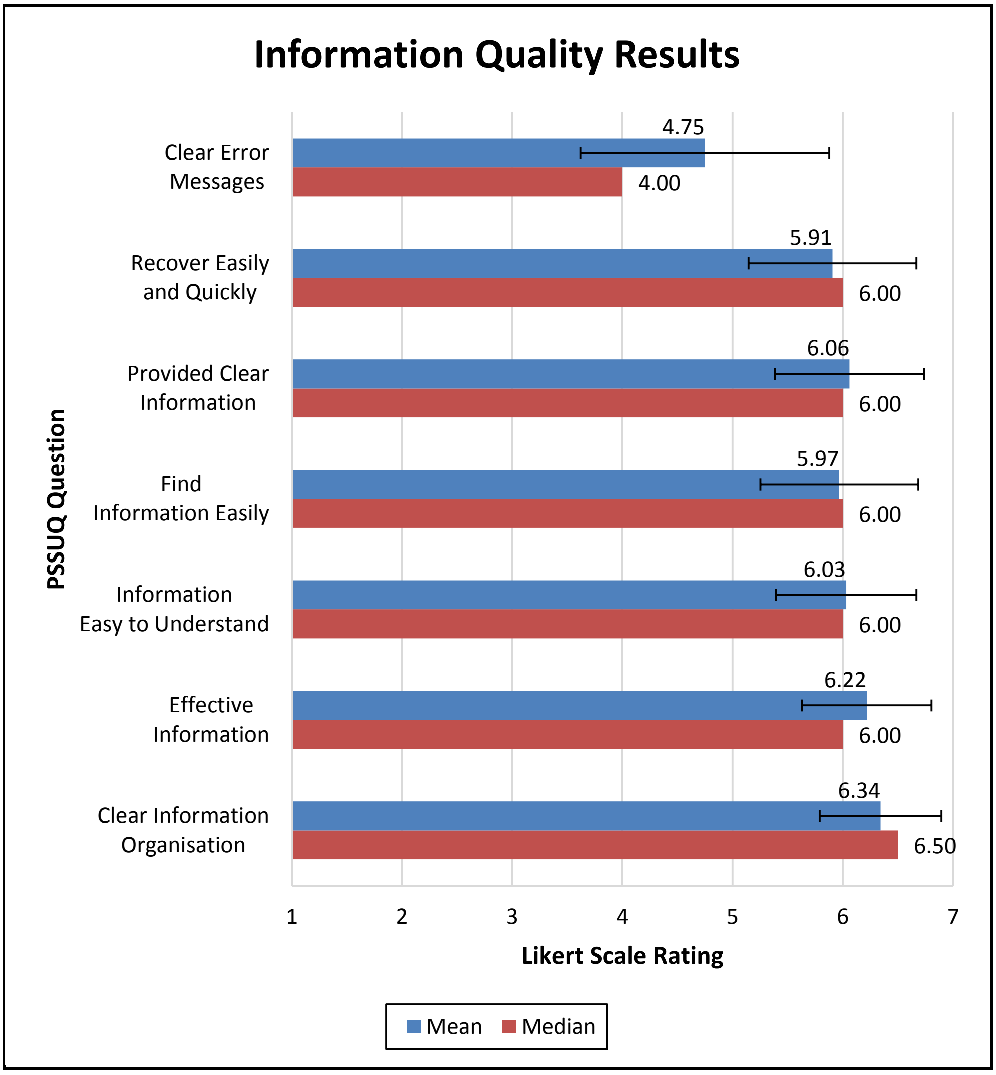


Figure 6.15: Statistical Analysis for Information Quality (n=32)

The next classification category of the PSSUQ is the *Information Quality*. Seven statements classified under this category are shown in Figure 6.15. Ninety-five percent confidence intervals are indicated for the mean values. The means of these statements were rated in the five to six range. There was one exception where statement nine had the lowest mean (μ=4.75), median (rated four), and mode (rated four). Statement nine was:

"The system gave error messages that clearly told me how to fix problems."

This was as a result of no error messages appearing during the usability evaluation of the application. Analysis of the results indicated that users experienced an odd occurrence of file(s) not being transferred to the intended recipient(s), which was easily overcome by initiating another file transfer.

Only one statement received the highest rating (mode=seven), closely followed by several other statements with a mode of six. Statement 15 was:

"The organisation of information on the system screens was clear."

The high rating of this statement indicates that the information organisation on the application was clear which resulted in a good user experience for the participants, which directly influenced the high rating of six given to the other statements. The rating of six indicates that the provided instructions were clear, easy to understand, and easy to follow.

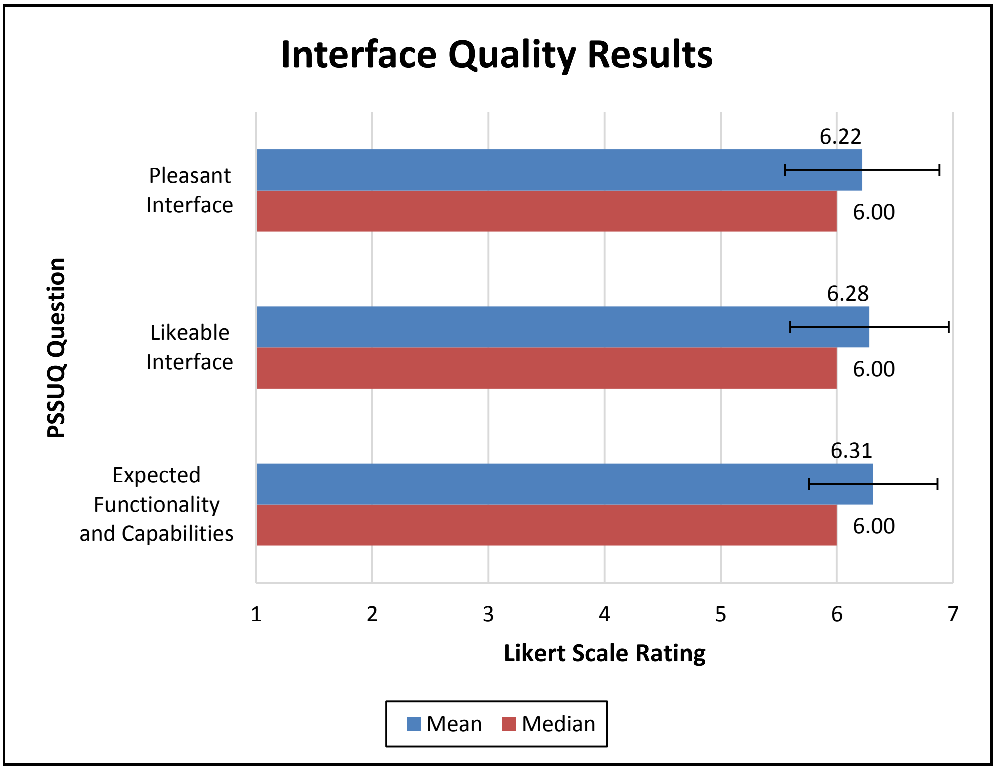


Figure 6.16: Statistical Analysis on Interface Quality (n=32)

Figure 6.16 illustrates the statistical functions performed on *Interface Quality* statements. Ninety-five percent confidence intervals are indicated for the mean values. All three statements had high mean values (μ=6.22, μ=6.28, and μ=6.31 respectively). This indicates that participants perceived the application interface to be highly usable and simple. The low standard deviation in the statements ($0.69 \leq \sigma \leq 0.85$) is as a result of participants having mobile device experience and their expectations of the application being satisfied.

The last section of the PSSUQ was the *Overview* section (Appendix H), which included open-ended questions. This section allowed the participants to list the positive and negative aspects of the application as well as any additional comments or recommendations. Thematic analysis is a qualitative data analysis method. Braun and Clarke (2006) defines *thematic analysis* as:

> *"A method for identifying, analysing, and reporting patterns or themes within data. It minimally organises and describes your data set in rich detail. Furthermore, it interprets various aspects of the research topic."*

Thematic analysis was applied to analyse the data generated from the *Overview* section in the PSSUQ, whereby themes were identified and discussed in various tables. Thematic analysis was applied to classify the participants' responses into various themes, whereby a frequency count (*f*) for each theme and corresponding percentage were computed. The category, theme, frequency count, percentage, and sample comments were tabulated according to the most positive aspects of MotionShare (Table 6.16.), the most negative aspects of MotionShare (Table 6.17), and additional comments or recommendations (Table 6.18). The key themes with the highest frequency counts are listed in each table.

In Table 6.16, the strongest theme was gesture detection (*f*=29) with 91.00% of participants expressing *"Intuitive ability to share data wirelessly by pointing the device to the intended target"*. This was followed by ease of use (*f*=23) with 72.00% commenting *"The system was easy to use"*. Effectiveness (*f*=20) was rated positively by 63.00% of participants stating *"Clear indication of devices on the map allowed for tasks to be completed"*. The map of devices theme (*f*=18) reflected 56% of participants mentioning the *"Map of devices is simple to understand"*. Exactly half of the participants (*f*=16) said the *"Compass is very responsive and accurate"*. Only 41.00% of participants (*f*=13) referred to the learnability of MotionShare as *"The system is intuitive"*.

Table 6.16: Key Themes in Positive Qualitative Feedback (n=32)

Category	Theme	Frequency Count (*f*)	%	Sample Comment
Performance	Ease of Use	23	72.00	*"The system was easy to use" "Simple to use"*
	Effectiveness	20	63.00	*"The system works amazingly well" "Clear indication of devices on the map allowed for tasks to be completed"*
	Learnability	13	41.00	*"The system is intuitive" "The system is easy to learn"*
User Satisfaction	Compass Functionality (Orientation)	16	50.00	*"Orientating the device and map adjustment according to user movement is good" "Compass is very responsive and accurate"*
	Map of Devices	18	56.00	*"Map of devices is simple to understand" "Map component provides a nice visualisation of the actual location of devices" "Locations of devices displayed on the map are fairly accurate"*
	Gesture Detection	29	91.00	*"Intuitive ability to share data wirelessly by pointing the device to the intended target" "Gesture detection of user touch made the application interactive"*

The most negative aspect of MotionShare was notification (*f*=12), where 38.00% of participants expressed the comment *"Insufficient feedback information when files have been selected or shared"*. The second most negative aspect was file share (*f*=8) with 25.00% of participants stating *"File sharing component is not as intuitive as it could be"*. This was followed by update (*f*=6), with 19.00% of participants commenting *"Odd occurrence of list of files not updated when files have been received"*. Both calibration (*f*=4) and simplicity (*f*=4) were rated negatively by 13.00% of participants who said *"In order for the system to be functional, the devices in the environment need to be calibrated"* and *"Simple design of GUI"* respectively.

Table 6.17: Key Themes in Negative Qualitative Feedback (n=32)

Category	Theme	Frequency Count (f)	%	Sample Comment
Functionality	File Share	8	25.00	*"File sharing component is not as intuitive as it could be"* *"Non-Android users may not be familiar with the select and hold to enable file selection mode"*
	Calibration	4	13.00	*"In order for the system to be functional, the devices in the environment need to be calibrated"*
User Interface	Notification	12	38.00	*"Insufficient feedback information when files have been selected or shared"* *"More system feedback needs to be provided by the system when performing tasks"*
	Update	6	19.00	*"Odd occurrence of list of files not updated when files have been received"*
	Simplicity	4	13.00	*"Simple design of GUI"*

A large number of additional comments or recommendations were made by the participants. The most frequent comment provided by 84.00% of participants was the theme *User Experience* (*f*=27) who said *"Overall functionality of the application and accuracy of the calibration used in sharing the files is impressive"*. This was closely followed by *Gesture Detection* (*f*=25) where 78.00% of participants stated *"Touch gesture for selective recipient sharing is cool"*. MotionShare's *Future Potential* was identified (*f*=17) by 53.00% of participants who stated *"A promising application with potential applications of commercial use"* and *"Computation of the device positions and knowing this information can lead to other potential* *applications of use"*. The theme *Additional* (*f*=11) was suggested with 34.00% of participants who mentioned *"Possible investigation into real-time tracking of device positioning as users move around in an environment"*. The theme *Gesture* (*f*=5) indicated 16.00% of participants who stated *"Additional gestures could be implemented"*. Only 9.00% of participants alluded to MotionShare's intuitiveness (*f*=3) who mentioned *"The use of this application relies on Android-specific experience"*.

Table 6.18: Key Themes in Additional Qualitative Feedback (n=32)

Category	Theme	Frequency Count (f)	%	Sample Comment
System	Future Potential	17	53.00	*"A promising application with potential applications of commercial use"* *"Computation of the device locations and knowing this information can lead to other potential applications of use"*
	Gesture	5	16.00	*"Additional gestures could be implemented"*
	Additional	11	34.00	*"Possible investigation into real-time tracking of device positioning as users move around in an environment"*
	Gesture Detection	25	78.00	*"Touch gesture for selective recipient sharing is cool"*
User Satisfaction	User Experience	27	84.00	*"Overall functionality of the application and accuracy of the calibration used in sharing the files is impressive"*
	Intuitiveness	3	9.00	*"The use of this application relies on Android-specific experience"*

The satisfaction results of the post-test questionnaires are displayed in Figure 6.17. Ninety-five percent confidence intervals are shown for the mean values. Participants preferred the touch gesture as a suitable and intuitive method of information sharing with co-located mobile devices (μ=5.75) as opposed to pointing the device in the direction of the intended recipients (point gesture). The point gesture was only preferred by 16.00% of participants (*f*=5). Question two of the post-test questionnaire indicated that the participants felt the touch gesture was slightly more complex than the point gesture (μ=4.13). The majority of participants (81.00%) found the touch gesture to be easier to use (μ=4.38) with only 19.00% preferring the point gesture instead (*f*=6). Participants expressed that both gestures (μ=4.19) required technical assistance. Question five asked the participants which technique they felt was more intuitive for information sharing and they responded with a slight preference towards point (μ=3.22) because they felt that pointing the device in the direction of the intended recipients was more intuitive than the touch gesture. Eighteen participants who favoured the point gesture (72.00%) selected the touch gesture as their preferred technique. The accuracy of these two gestures was rated in favour of the touch gesture (μ=5.59) by 91.00% of participants (*f*=29). Participants felt individuals would learn to use the touch gesture more quickly than the point gesture, by a slight margin (μ=4.13). This was as a result of participants expressing that both techniques are relatively easy to learn. Participants believed potential users would become quickly proficient using both gestures in order to share information among co-located mobile devices. Participants were equally divided (μ=4.03) on which technique was more cumbersome with 63.00% (*f*=20) of them remaining neutral (rated 4). Participant confidence in using the touch gesture (μ=5.59 and *f*=26) was a clear indication of them feeling more in control as the nature of this technique was direct. The low standard deviations for each question indicates participants had similar views on the NUI interaction techniques.

The results of the pilot study were compared to the results of the summative evaluation. The summative evaluation results were better than the pilot study as the larger sample size provided a better confidence level and a closer estimate to the true population values (Tullis & Albert, 2013).

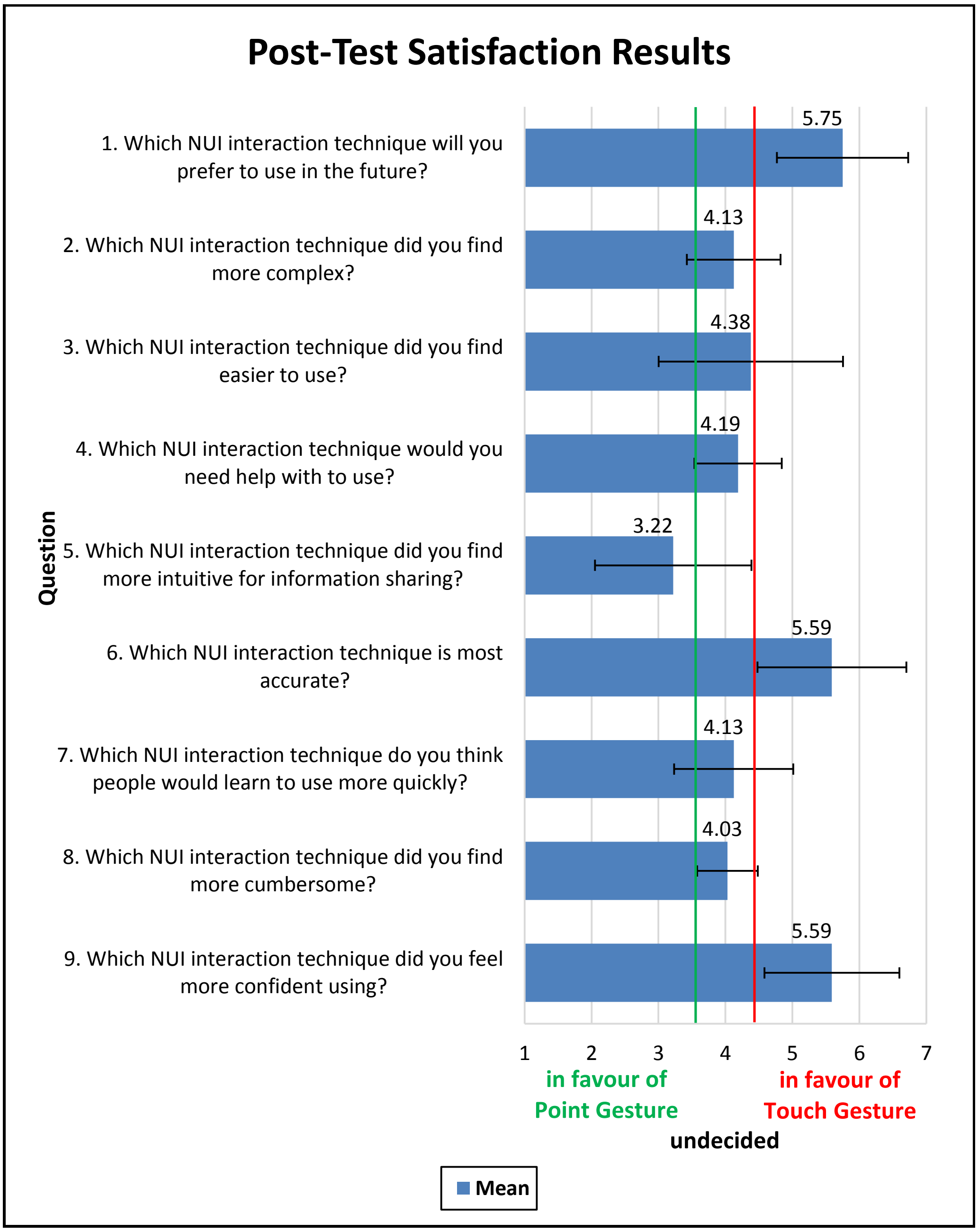


Figure 6.17: Post-Test Satisfaction Results (n=32)

6.10 Discussion

The overall evaluation was classified as ex post, which included the use of two techniques, namely analytical and experimental. The experimental evaluation technique was used in both a formative evaluation and a summative evaluation. The formative evaluation was iteratively performed in the form of expert reviews and a pilot study. The results of the pilot study indicated MotionShare was ready to be evaluated and the statistical and analytical procedures performed were correct. Upon completion of this evaluation, the summative evaluation commenced. The summative evaluation involved participants using both NUI interaction techniques to complete a list of tasks.

The general consensus was that the usability of MotionShare was good, as evident from the usability results (86.00% overall rating mentioned in Section 6.9.3). User satisfaction results were high, as the participants rated MotionShare with a mean value of 5.87 in the PSSUQ statements. The key themes in the positive qualitative feedback (Table 6.16) included:

1. *Gesture Detection* (91.00%).
2. *Ease of Use* (72.00%);
3. *Effectiveness* (63.00%);
4. *Map of Devices* (56.00%);
5. *Compass Orientation Functionality* (50.00%); and
6. *Learnability* (41.00%);

These themes were identified from the participants' comments with the strongest theme identified to be gesture detection (91.00%). The *Gesture Detection* theme showed that the participants were highly satisfied with the support of these NUI interaction techniques for information sharing process.

The negative aspects of MotionShare allowed for easy identification of potential usability issues to be solved, thereby improving the overall design (Table 6.17). The negative themes identified were:

1. *Simplicity* (13.00%);
2. *Calibration* (13.00%);
3. *Update* (19.00%);
4. *File Share* (25.00%); and
5. *Notification* (38.00%).

The most negative aspect was identified to be *Notification* (38.00%), where participants experienced insufficient system feedback and expected more feedback when performing tasks using MotionShare. The negative feedback received had lower frequency counts compared to the positive feedback.

Other themes emerged from the questions asking participants to make additional comments or suggestions (Table 6.18). These themes were:

1. *User Experience* (84.00%);
2. *Gesture Detection* (78.00%);
3. *Future Potential* (53.00%);
4. *Additional* (34.00%);
5. *Gesture* (16.00%); and
6. *Intuitiveness* (9.00%).

The notable comments were related to *User Experience* (84.00% of participants were impressed with overall functionality), G*esture Detection* (78.00% described the touch gesture for selective sharing to be cool), and *Future Potential* (53.00% believed the application showed great potential for various applications of use).

Lastly, the post-test questionnaire asked the participants which NUI interaction technique they preferred. Nine questions were asked to assess the level of satisfaction experienced using both techniques, namely the point gesture and the touch gesture. The results reflected that the participants strongly preferred the touch gesture (μ=5.75). Accuracy and confidence in using the touch gesture had favourable mean values of 5.59 and 5.59 respectively.

6.11 Conclusion

This chapter followed both activities, ***Demonstrate Artefact*** and ***Evaluate Artefact***, within the Design Cycle and Rigor Cycle of the DSR methodology (Section 1.7.3.1). This chapter addressed the sixth research question (Section 1.7.1):

> *"RQ6. How accurate and usable is the proxemic prototype NUI in supporting information sharing among co-located mobile devices?"*

The evaluation of DSR artefacts was identified as an essential activity of the DSR methodology. A naturalistic evaluation technique was selected whereby

MotionShare was placed in a real environment and its performance was evaluated. The methods used within the usability evaluation rigorously measured the performance metrics, namely effectiveness and efficiency, and the satisfaction metric.

The evaluation involved 32 participants who were divided in eight groups of four members each. The Cronbach's alpha coefficient was used to compute the internal consistency of the system usability section of the PSSUQ (α=0.92 was interpreted as excellent consistency) to ensure that the results derived were reliable.

The results indicated MotionShare's usability rating as 86.00%, which is regarded as very good, according to Sauro and Lewis (2012). The high ratings by participants in the respective classifications of the PSSUQ categories demonstrated MotionShare to be effective and efficient. The positive qualitative feedback received from the open-ended questions in the overview section of the PSSUQ allowed for thematic analysis to identify themes, which were used to describe MotionShare. The negative feedback received together with the additional comments or recommendations served to improve MotionShare.

The positive usability results of this evaluation provide empirical evidence that the design and implementation of a proxemic prototype NUI, in the form of MotionShare, can provide accurate and usable support for information sharing among co-located mobile devices. Therefore, this chapter has answered the sixth research objective (Section 1.7.2):

> *"RO6. To evaluate the accuracy and usability of the proxemic prototype NUI in supporting information sharing among co-located mobile devices."*

This dissertation is concluded in the following chapter. The achievement of the research objectives and the research contributions are presented. Limitations and problems experienced are also discussed. Lastly, opportunities for future research are mentioned.

Chapter 7: Conclusions

7.1 Introduction

This chapter concludes the dissertation by revisiting the original research questions and research objectives in order to determine whether they were answered and achieved. The chapter follows ***Communication***, the sixth activity of the DSR methodology (Figure 7.1), which is performed within the Rigor Cycle. The primary research question (Section 1.7.1) of this research was:

> ***"How can a proxemic Natural User Interface be designed to provide an accurate and usable solution to support information sharing among co-located mobile devices?"***

Similarly, the primary research objective (Section 1.7.2) of this research was:

> ***"To design a proxemic Natural User Interface to provide an accurate and usable solution to support information sharing among co-located mobile devices."***

The DSR methodology was used throughout this research in the development of the design artefact, named MotionShare. MotionShare was designed and implemented to evaluate whether a proxemic prototype NUI could be designed to support information sharing among co-located mobile devices. The evaluation results would determine whether MotionShare could achieve this objective. The theoretical and practical contributions of this research are highlighted. Limitations and problems experienced are described. Finally, possibilities and recommendations for future research are explored, which conclude the chapter. Figure 7.2 depicts the structure of this chapter.

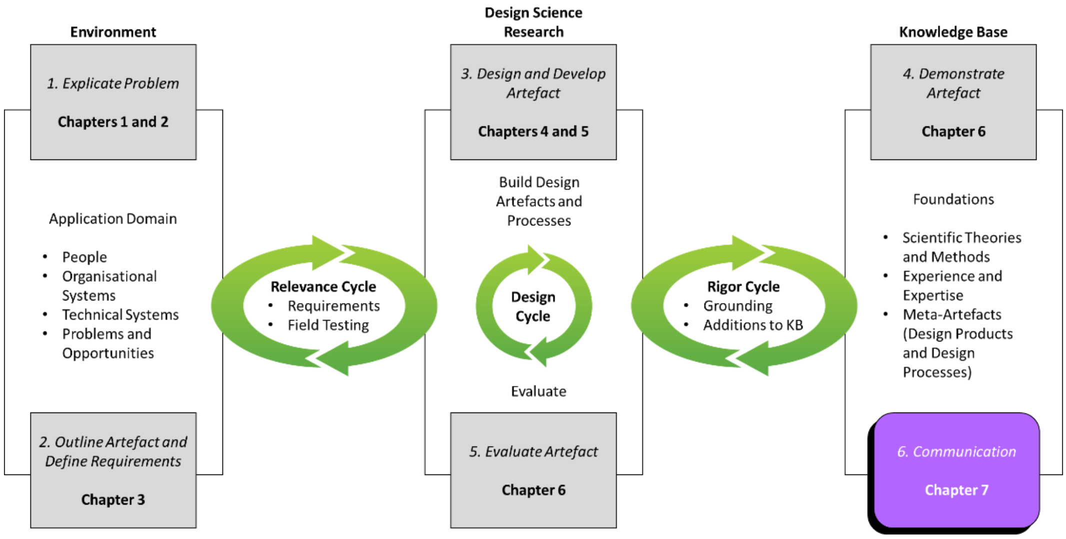


Figure 7.1: Chapter 7 Position in the DSR Methodology

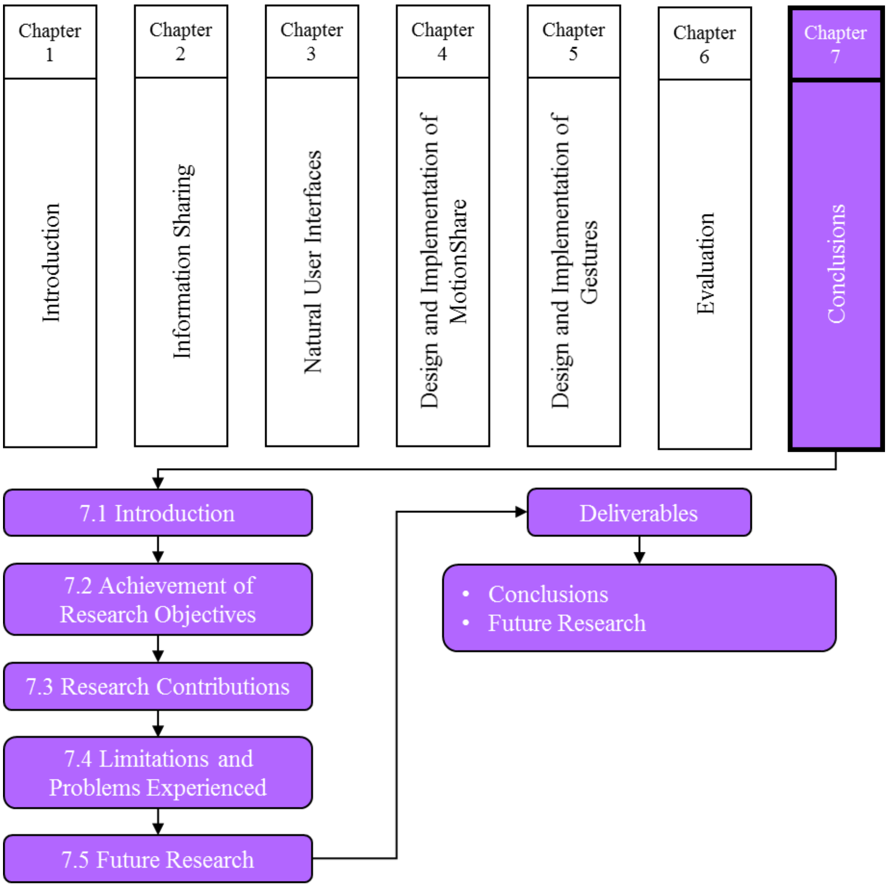


Figure 7.2: Chapter 7 Structure

7.2 Achievement of Research Objectives

This research indicated that existing applications do not fully utilise the accuracy and usability of NUI interaction techniques. Existing methods of information sharing require a manual, redundant, and tedious process in order to share information. The primary objective of this research (Section 1.7.2) was:

> ***"To design a proxemic Natural User Interface to provide an accurate and usable solution to support information sharing among co-located mobile devices."***

In order for the research to achieve this objective, the following objectives were derived (Section 1.7.2):

"RO1. To identify the shortcomings of existing information sharing methods currently used by mobile devices (Chapter 2).

RO2. To identify the benefits and shortcomings of existing NUI interaction techniques for information sharing (Chapter 3).

RO3. To determine how the relative pose can be calculated for co-located mobile devices (Chapter 4).

RO4. To determine how NUI interactions techniques can be designed to support information sharing among co-located mobile devices (Chapter 5).

RO5. To develop a proxemic prototype NUI to support information sharing among co-located mobile devices (Chapter 5).

RO6. To evaluate the accuracy and usability of the proxemic prototype NUI in supporting information sharing among co-located mobile devices (Chapter 6)."

The various phases, activities, guidelines, and cycles of the DSR methodology identified and discussed in Section 1.7.3.1, were followed and applied throughout this research. These aspects of the DSR methodology were used to address each of these research objectives identified above.

Chapter 2 addressed RO_1 by investigating the existing information sharing methods used by mobile devices. The field of information sharing was discussed to provide a better understanding and insight into the problem domain. The definition and importance of information sharing were outlined. The various information types that are typically shared among mobile devices were identified. Existing systems for information sharing among mobile devices were reviewed. A comparison of the different information sharing techniques indicated that existing techniques do not utilise a proxemic NUI and its associated interaction techniques. Therefore, the use of a proxemic NUI for information sharing among co-located mobile devices was identified as a potential solution to the issues and shortcomings of existing information sharing systems.

The next research objective, RO_2, was addressed in Chapter 3. Due to the complex nature of NUIs, the need to clarify and establish a universally acceptable definition of a NUI was required. Existing literature revealed numerous definitions of NUIs; however, all of these definitions were derived from a single definition of a NUI. A general NUI definition was adopted for the purposes of this research. Investigation into the different NUI definitions also resulted in the different characteristics of NUIs and the various NUI interaction techniques being identified.

NUIs provide several potential benefits of use, including its intuitive and direct nature to help individuals to complete tasks. The various interaction techniques were defined and their benefits and shortcomings discussed. This allowed for the selection of suitable techniques to support information sharing among co-located mobile devices. Existing NUI systems were reviewed, to show how these systems implemented and used NUI interaction techniques. The results of this analysis was considered in determining which interaction techniques were relevant for the purposes of this research.

Chapter 4 involved the design and implementation of MotionShare. Chapter 5 discussed the design and implementation of the gestures incorporated into MotionShare. Together, these two chapters achieved RO_3, RO_4, and RO_5. Existing positioning techniques for mobile devices were identified and their shortcomings highlighted. A more fine-grained solution was required as the granularity of existing techniques were too coarse-grained for the purposes of this research. Numerous experiments were conducted using Bluetooth to determine its feasibility as an accurate indicator of distance between co-located mobile devices. The results of these experiments showed that Bluetooth RSSI could be used together with sensor fusion to approximate the position of mobile devices in a co-located environment. The implementation of MotionShare resulted in several prototypes being developed and tested. Feedback received during informal testing resulted in iterations of these prototypes before they were integrated into MotionShare.

Chapter 5 involved the discussion of the design and implementation of the NUI gestures. The design of several potential NUI interaction techniques were outlined and implemented into a prototype, named ProtoGesture. Focus groups were conducted to determine which techniques the users thought could be used (intuitive) when performing specific information sharing tasks. Existing literature suggested various potential gestures. A comparison of the focus group results and the existing literature allowed for a decision to be made regarding which gestures could be implemented in MotionShare. Two gestures were incorporated into MotionShare, namely the point gesture and the touch gesture.

MotionShare was subjected to an ex post evaluation using two techniques, namely analytical and experimental. Chapter 6 detailed the entire evaluation design process to address RO_6. The analytical evaluation technique was used to assess the precision, trueness, and recall metrics of device positioning as well as the MotionShare gestures. The results of evaluation of the device positioning and MotionShare gestures revealed the precision, trueness, and recall to be good (Baeza-Yates & Manber, 2012). The experimental evaluation technique was broken up into two types of evaluations, namely formative and summative.

The formative evaluation was a pilot study. The results of the pilot study revealed minor issues with MotionShare, which were implemented to improve the overall usability of the artefact before the summative evaluation commenced.

A summative usability evaluation was conducted to evaluate the usability of the NUI interaction techniques implemented in MotionShare. The evaluation was based on the effectiveness, efficiency, and satisfaction metrics.

The evaluation of MotionShare involved 32 participants (eight groups of four) using mobile devices in a co-located environment. Quantitative feedback received indicated that MotionShare had high levels of usability. The qualitative feedback revealed more positive than negative comments. The results of the PSSUQ confirmed that the NUI interaction techniques incorporated were effective, efficient, and satisfactory in supporting information sharing among co-located mobile devices. Participants were also subjected to a post-test questionnaire to determine which NUI interaction technique was preferred by the participants. The touch gesture was the preferred choice among participants for the information sharing tasks of MotionShare. This was because they felt more in control with highlighting the device dots and it was not as dependent on the embedded sensors as the point gesture.

This section has demonstrated that all the research objectives were successfully achieved. The theoretical and practical contributions as a result of this research are highlighted in the following section.

7.3 Research Contributions

The contributions of this research are classified into two areas, either as theoretical or practical. The theoretical contributions involve the use of a proxemic NUI to support information sharing. The practical contributions involve the design and implementation of MotionShare.

7.3.1 Theoretical Contributions

A theoretical contribution is the use of Bluetooth as an indicator of distance for mobile devices to a fine level of granularity, which existing positioning techniques are not able to provide. Bluetooth was very sensitive to environmental factors. Bluetooth combined with sensor fusion allowed for an approximation of the positions of mobile devices in a co-located environment. This allowed for appropriate NUI interaction techniques to be implemented based on the calculated position information, which contributes to the research on indoor positioning for mobile devices.

Lastly, another theoretical contribution is demonstrating that NUI interaction techniques can be designed to intuitively and efficiently support information sharing among co-located mobile devices. The evaluation results provide empirical evidence, which indicate that NUI interaction techniques can be applied to information sharing for mobile devices in a co-located environment. The results reflect that the usage of these techniques resulted in high levels of effectiveness, efficiency, and satisfaction. MotionShare could potentially form the foundational basis for future development in similar applications when the position of co-located devices are known.

7.3.2 Practical Contributions

MotionShare was developed as a proof-of-concept application to demonstrate that a proxemic NUI can be designed to support information sharing among co-located mobile devices. MotionShare is an Android application, which allows the use of NUI interaction techniques to be performed when sharing information among co-located mobile devices. MotionShare users were allowed to share various file types, namely documents, images, music, and videos.

MotionShare demonstrates how two NUI interaction techniques can be used when the actual context has been established. *Actual context* refers to the application actually knowing the approximate positions of the devices in the environment and responding accordingly to the interaction technique performed. Logically, in the event where the interaction technique is directed towards no devices, information sharing will not occur. Existing information sharing systems which have some form of a NUI interaction technique, but without knowledge of actual context, are not capable of performing these techniques correctly. This is a result of the positions of devices not being known by these systems.

The results of the evaluation present the opportunity for other researchers to potentially use these results as a comparative source of reference when conducting similar research. These results indicate that MotionShare obtained high levels of effectiveness, efficiency, and satisfaction. The effectiveness metric revealed participants had no problem in completing the list of tasks using MotionShare. The efficiency metric indicated almost all participants were able to complete the tasks. High satisfaction levels were experienced by participants when using MotionShare.

7.4 Limitations and Problems Experienced

Several problems were encountered throughout this research. One of the problems was identified to be the existing positioning techniques for mobile devices, such as Wi-Fi, GPS, and Cell Tower Triangulation. These techniques are only able to determine the positions of mobile devices to within kilometres and metres, which was deemed to be too coarse-grained for the purposes of this research. Therefore, a need arose to find a potential solution to determine the positions of these devices to be more fine-grained than existing techniques.

The primary problem of this research was identified during the design and implementation cycle. Existing mobile systems that have implemented a NUI do not provide an actual context. This context is referred to as knowing the positions of co-located mobile devices to ensure the appropriate NUI interaction techniques could be efficiently utilised. Indoor positioning for mobile devices is a complex field and remains inaccurate and highly volatile. The instability of indoor positioning information is a result of the external variables, which influence the sensor data output by the various sensors embedded in mobile devices. Indoor positioning typically requires the use of external hardware, such as dongles attached to the device or cameras placed throughout an environment, to support the computations of the positions of the devices. Sensor data was identified to be the key component of this research, which utilised sensor fusion.

MotionShare depends on the accuracy of sensor fusion and Bluetooth RSSI. This dependency was shown to be an issue because external factors can influence the data transmitted by the embedded sensors and Bluetooth. Certain environments were shown to adversely affect the operations of MotionShare, which resulted in the initial calibrations of determining the device positions to be inaccurate and ineffective. The sensors embedded in mobile devices are not as accurate as expected and can be easily affected by external variables. It is possible that with the rapid development of mobile devices, embedded sensors will be improved and become more accurate and stable in the future.

7.5 Future Research

The additional qualitative feedback received from the evaluation indicated several opportunities for future research. A larger population sample of participants from a more diverse background could potentially provide greater insight and conclusive results into the efficiency, effectiveness, and satisfaction of MotionShare and the implemented NUI interaction techniques. A larger data set allows for more complex statistical analysis to be performed.

This research has shown that a proxemic NUI can be designed to dynamically track the positions of mobile devices in a co-located environment, which potentially allows for several other applications of use. The improved tracking of the positions could be achieved by the improvement of existing indoor positioning techniques or the development of new innovative techniques with improved accuracy and stability levels of sensor data transmitted by the embedded sensors in mobile devices.

MotionShare only incorporated two NUI interaction techniques and more techniques could be introduced and developed. This can be followed up with evaluating their effectiveness, efficiency, and satisfaction or comparing one technique with one another. The possibility of introducing in-air gestures and enhanced proxemic interaction could be implemented; however, literature has suggested that in-air gestures are distracting when conversing face to face with other individuals. The appropriateness of in-air gestures is dependent on the size of the gesture and the context of use.

MotionShare was developed on the Android platform and can thus only operate on Android mobile devices. MotionShare could potentially be developed for other mobile platforms. The technological advancement in mobile computing technology suggests that positioning techniques can become more accurate in the future. The sensor data can become more reliable as the embedded sensors in mobile devices also improve.

Another suggestion for future research is the introduction of an easier calibration phase with the mobile devices based on different environments. With mobile computing continually improving, an easier and possibly more accurate calibration can be developed in the future.

In conclusion, the research questions and objectives were successfully answered and achieved. The design of a proxemic NUI to support information sharing among co-located mobile devices was effectively implemented. The results of the rigorous evaluations indicated that the proxemic NUI successfully achieved its aims and objectives.

References

Agrawal, S., Constandache, I., Gaonkar, S., Choudhury, R. R., Caves, K., & DeRuyter, F. (2011). Using Mobile Phones to Write in Air. In *MobiSys 2011: Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services* (pp. 15–28). Bethesda, MD.

Ahern, S., Davis, M., King, S., Naaman, M., & Nair, R. (2006). Reliable, User-Contributed GSM Cell-Tower Positioning Using Context-Aware Photos. In *UbiComp 2006: Proceedings of the 8th International Conference on Ubiquitous Computing* (pp. 1–2). Orange County, CA.

Alvarez, C., Brown, C., & Nussbaum, M. (2011). Comparative Study of Netbooks and Tablet PCs for Fostering Face-To-Face Collaborative Learning. *Journal of Computers in Human Behavior*, *27*(2), 834–844.

Amir, O. (2013). Information Sharing for Care Coordination. In *AAMAS 2013: Proceedings of the 12th International Conference on Autonomous Agents and Multi-Agent Systems* (pp. 3059–3060). Richland, SC.

Anmobi Inc. (2015a). Xender - The Mobile File Transferring Application. Retrieved October 25, 2015, from http://www.xender.com/index.html

Anmobi Inc. (2015b). Xender: File Transfer, Sharing. Retrieved October 25, 2015, from https://play.google.com/store/apps/details?id=cn.xender&hl=en

Apple Inc. (2015). iPhone - Apple. Retrieved October 27, 2015, from http://www.apple.com/iphone/

Arackal, C. M. (2014). How Can Sockets in Android Be Explained Simply? Retrieved September 15, 2015, from http://www.quora.com/How-can-sockets-in-Android-be-explained-simply

Arigela, L., Veerendra, P., Anvesh, S., & Hanuman, K. (2013). Mobile Phone Tracking and Positioning Techniques. *International Journal of Innovative Research in Science, Engineering and Technology*, *2*(4), 906–913.

ASTM. (2011). Accuracy and Precision (1): ISO Definitions. Retrieved November 7, 2015, from http://www.astro.lu.se/Education/utb/ASTM11/lecture2.pdf

ASUSTeK Computer Inc. (2015a). ASUS ZenUI. Retrieved October 25, 2015, from http://www.zenui.com/

ASUSTeK Computer Inc. (2015b). Share Link - File Transfer. Retrieved October 25, 2015, from https://play.google.com/store/apps/details?id=com.asus.sharerim&hl=en

Aviv, A. J., Sapp, B., Blaze, M., & Smith, J. M. (2012). Practicality of Accelerometer Side Channels on Smartphones. In *ACSAC 2012: Proceedings of the 28th Annual Computer Security Applications Conference* (pp. 41–50). Orlando, FL.

Baeza-Yates, R., & Manber, U. (2012). *Computer Science: Research and Applications*. New York, NY: Springer.

Ballmer, S. (2011). CES 2010: A Transforming Trend - The Natural User Interface. Retrieved October 26, 2015, from http://www.huffingtonpost.com/steve-ballmer/ces-2010-a-transforming-t_b_416598.html

Bank, C. (2014). Gestures and Animations: The Pillars of Mobile Design. Retrieved October 27, 2015, from https://uxmag.com/articles/gestures-animations-the-pillars-of-mobile-design

Baotić, A. (2014). Simple Client-Server Syncing for Mobile Applications using Couchbase Mobile. Retrieved November 6, 2015, from https://infinum.co/the-capsized-eight/articles/server-client-syncing-for-mobile-apps-using-couchbase-mobile

Barkhuus, L., & Polichar, V. E. (2011). Empowerment through Seamfulness: Smartphones in Everyday Life. *Journal of Personal and Ubiquitous Computing*, *15*(6), 629–639.

Beal, V. (2015). What is Email? Retrieved December 1, 2015, from http://www.webopedia.com/TERM/E/e_mail.html

Berkowitz, B. (2015). Conducting Focus Groups. Retrieved October 21, 2015, from http://ctb.ku.edu/en/table-of-contents/assessment/assessing-community-needs-and-resources/conduct-focus-groups/main

Billinghurst, M., Piumsomboon, T., & Bai, H. (2014). Hands in Space: Gesture Interaction with Augmented-Reality Interfaces. *Journal of IEEE Computer Graphics and Applications*, *34*(1), 77–80.

Blake, J. (2013). *Natural User Interfaces in .NET*. Greenwich, CT: Manning Publications.

Bluetooth SIG. (2015). Bluetooth Technology Website. Retrieved December 1, 2015, from https://www.bluetooth.com/

Boyle, M., & Schmierbach, M. (2015). *Applied Communication Research Methods: Getting Started as a Researcher*. New York, NY: Routledge.

Bratitsis, T., & Kandroudi, M. (2014). Motion Sensor Technologies in Education. *EAI 2014: Journal of European Alliance for Innovation Endorsed Transactions on Serious Games*, *1*(2), 1–10.

Braun, V., & Clarke, V. (2006). Using Thematic Analysis in Psychology. *Journal of Quality Research in Psychology*, *3*(2), 77–101.

Britton, A., Setchi, R., & Marsh, A. (2013). Intuitive Interaction with Multi-Functional Mobile Interfaces. *Journal of King Saud University - Computer and Information Sciences*, *25*(2), 187–196.

Budiu, R. (2015). Mobile User Experience: Limitations and Strengths. Retrieved October 27, 2015, from http://www.nngroup.com/articles/mobile-ux/

Butter, M., Quintana, M., & Valenzuela, E. (2014). Incremental Prototyping Model for the Development of Educational Platforms: A Process of Design and Quality Standards. *Journal of Universal Computer Science*, *20*(10), 1–10.

Câmara, A. (2011). Natural User Interfaces. In *INTERACT 2011: Proceedings of the 13th IFIP TC 13 International Conference* (pp. 1–2). Lisbon, Portugal.

Campbell, J. (2011). Why is 30 the "Magic Number" for Sample Size? Retrieved October 3, 2015, from http://www.jedcampbell.com/?p=262

Chen, B., & Denoyelles, A. (2013). Exploring Students' Mobile Learning Practices in Higher Education. Retrieved October 24, 2015, from http://er.educause.edu/articles/2013/10/exploring-students-mobile-learning-practices-in-higher-education

Chen, K.-Y., Ashbrook, D., Goel, M., Lee, S.-H., & Patel, S. (2014). AirLink: Sharing Files between Multiple Devices Using In-Air Gestures. In *UbiComp 2014: Proceedings of the International Joint Conference on Pervasive and Ubiquitous Computing* (pp. 565–569). Seattle, WA.

Chen, X. A., Schwarz, J., Harrison, C., Mankoff, J., & Hudson, S. E. (2014). Air+Touch: Interweaving Touch and In-Air Gestures. In *UIST 2014: Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (pp. 519–525). Honolulu, HI.

Chong, A. (2013). The Future of UX or UI: The Natural User Interface. Retrieved October 26, 2015, from http://blog.pivotal.io/labs/labs/the-future-of-uiux-the-natural-user-interface

Cleven, A., Gubler, P., & Hüner, K. M. (2009). Design Alternatives for the Evaluation of Design Science Research Artefacts. In *DESRIST 2009: Proceedings of the 4th International Conference on Design Research in Information Systems and TechnologyP* (pp. 1–8). Malvern, PA.

Cohen, D., & Crabtree, B. (2006). Qualitative Research Guidelines Project. Retrieved October 21, 2015, from http://www.qualres.org/HomeFocu-3647.html

Conder, S., & Darcey, L. (2010). Learn Java for Android Development: Introduction to Java. Retrieved September 14, 2015, from http://code.tutsplus.com/tutorials/learn-java-for-android-development-introduction-to-java--mobile-2604

Crowdsourcing LLC. (2015). Crowdsourcing and Crowdfunding - The Industry Website. Retrieved November 30, 2015, from http://www.crowdsourcing.org/

Curccurullo, S., Francese, R., Murad, S., Passero, I., & Tucci, M. (2012). A Gestural Approach to Presentation Exploiting Motion Capture Metaphors. In *AVI 2012: Proceedings of the International Working Conference on Advanced Visual Interfaces* (pp. 148–155). Capri, Italy.

Data School. (2014). Simple Guide to Confusion Matrix Terminology. Retrieved May 21, 2015, from http://www.dataschool.io/simple-guide-to-confusion-matrix-terminology/

Devasena, C. L. (2013). Classification of Incomplete Multivariate Datasets using Memory Based Classifiers – A Proficiency Evaluation. *International Journal of Advanced Trends Computer Science and Engineering*, *2*(1), 148–153.

DewMobile Inc. (2015). Zapya-Connecting 300,000,000 People Worldwide. Retrieved November 15, 2015, from http://www.izapya.com/

Dingler, T., Funk, M., & Alt, F. (2015). Interaction Proxemics: Combining Physical Spaces for Seamless Gesture Interaction. In *PerDis 2015: Proceedings of the 4th International Symposium on Pervasive Displays* (pp. 107–114). Saarbrücken, Germany.

Dropbox Inc. (2015). Dropbox. Retrieved December 1, 2015, from https://www.dropbox.com/

eMarketer. (2014). 2 Billion Consumers Worldwide to Get Smartphones by 2016. Retrieved November 1, 2015, from http://www.emarketer.com/Article/2-Billion-Consumers-Worldwide-Smartphones-by-2016/1011694

eTutorials.org. (2015). Mobile Positioning Techniques. Retrieved November 14, 2015, from http://etutorials.org/Mobile+devices/mobile+wireless+design/Part+Four+Beyond+Enterprise+Data/Chapter+17+Location-Based+Services/Mobile+Positioning+Techniques/

Evaluation Toolbox. (2010). Formative Evaluation. Retrieved October 8, 2015, from http://evaluationtoolbox.net.au/index.php?option=com_content&view=article&id=24&Itemid=125

Eye Internet Ltd. (2014). List of IDEs for Android App Development, Which is Best for You? Retrieved November 14, 2015, from http://tekeye.biz/2014/list-of-android-app-development-ides

FeePerfect. (2015). Feem - Wi-Fi File Transfers between Phones, Tablets and PCs. Retrieved October 25, 2015, from http://tryfeem.com/

Feinzaig, L. (2013). The Commerical Birth of Natural Computing. Retrieved October 27, 2015, from http://allthingsd.com/20130205/the-commercial-birth-of-natural-computing/

Flagg, B. N. (2013). *Formative Evaluation for Educational Technologies*. New York, NY: Routledge.

Frughling, A., & Lee, S. (2005). Assessing the Reliability, Validity and Adaptability of PSSUQ. In *AMCIS 2005: Proceedings of the 11th Americas Conference on Information Systems* (pp. 2394–2402). Omaha, NE.

Gantenbein, D. (2011). TechFest Focus: Natural User Interfaces. Retrieved October 26, 2015, from http://research.microsoft.com/en-us/news/features/techfest2011nui-030811.aspx

Gava, V. L., De Mesquita Spinola, M., Tonini, A. C., & Medina, J. C. (2012). The 3C Cooperation Model Applied to the Classical Requirement Analysis. *JISTEM 2012: Journal of Information Systems and Technology Management*, *9*(2), 235–264.

George, D., & Mallery, P. (2013). *IBM SPSS Statistics 21 Step by Step: A Simple Guide and Reference* (13th ed.). New York, NY: Pearson Higher Education.

Google Inc. (2015a). Android Studio Overview. Retrieved September 14, 2015, from http://developer.android.com/tools/studio/index.html

Google Inc. (2015b). Developer Workflow. Retrieved September 15, 2015, from https://developer.android.com/tools/workflow/index.html

Google Inc. (2015c). Motion Sensors. Retrieved September 20, 2015, from http://developer.android.com/guide/topics/sensors/sensors_motion.html

Google Inc. (2015d). Position Sensors. Retrieved September 20, 2015, from http://developer.android.com/guide/topics/sensors/sensors_position.html

Google Inc. (2015e). Principles - Layout. Retrieved October 27, 2015, from https://www.google.com/design/spec/layout/principles.html

Google Inc. (2015f). Responsive Interaction. Retrieved October 27, 2015, from https://www.google.com/design/spec/animation/responsive-interaction.html

Google Inc. (2015g). Sensors Overview. Retrieved September 20, 2015, from http://developer.android.com/guide/topics/sensors/sensors_overview.html

Gosling, J., Joy, B., Steele, G., Bracha, G., & Buckley, A. (2015). *The Java Language Specification*. Upper Saddle River, NJ: Addison-Wesley.

Graf, T. (2012). Power-Efficient Positioning Technologies for Mobile Devices. Retrieved December 1, 2015, from https://www.snet.tu-berlin.de/fileadmin/fg220/courses/WS1112/snet-project/power-efficient-positioning-technologies_graf.pdf

Gube, J. (2013). Why Prototyping is Essential to Your Design Process. Retrieved October 14, 2015, from http://designinstruct.com/web-design/prototyping-is-essential/

Guida, G., Lamperti, G., & Zanella, M. (2013). *Software Prototyping in Data and Knowledge Engineering*. Dordrecht, Netherlands: Springer.

Gummeson, J. J., Priyantha, B., Ganesan, D., Thrasher, D., & Zhang, P. (2013). EnGarde: Protecting the Mobile Phone from Malicious NFC Interactions. In *MobiSys 2013: Proceedings of the 11th International Conference on Mobile Systems, Applications, and Services* (pp. 445–458). Taipei, Taiwan.

Gupta, A., Mattarelli, E., Seshasai, S., & Broschak, J. (2009). Use of Collaborative Technologies and Knowledge Sharing in Co-Located and Distributed Teams: Towards the 24-h Knowledge Factory. *Journal of Strategic Information Systems*, *18*(3), 147–161.

Gupta, S., Morris, D., Pattel, S., & Tan, D. (2012). SoundWave: Using the Doppler Effect to Sense Gestures. In *CHI 2012: Proceedings of the SIGSCHI Conference on Human Factors in Computing Systems* (pp. 1911–1914). Austin, TX.

Heikkinen, K., & Porras, J. (2013). UIs - Past, Present and Future. Retrieved October 26, 2015, from http://www.wwrf.ch/files/wwrf/content/files/publications/outlook/WWRF_outlook_10.pdf

Herrero, P., Panetto, H., Meersman, R., & Dillon, T. (2012). *On the Move to Meaningful Internet Systems: OTM 2012 Workshops*. Berlin, Germany: Springer.

Hevner, A. R. (2007). A Three Cycle View of Design Science Research. *Scandinavian Journal of Information Systems, 19*(2), 87–92.

Hevner, A. R., & Chatterjee, S. (2010). *Design Research in Information Systems: Theory and Practice*. New York, NY: Springer.

Hevner, A. R., March, S., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. *Journal of Management Information Systems Quarterly, 28*(1), 75–105.

Higher National Computing. (2010). Incremental Prototyping. Retrieved September 13, 2015, from http://www.sqa.org.uk/e-learning/IMAuthoring01CD/page_09.htm

Hightower, J. (2011). Mobile Location Technologies. Retrieved November 14, 2015, from https://courses.cs.washington.edu/courses/csep590b/11wi/lectures/110124-location.pdf

Hightower, J., & Borriello, G. (2001). Location Systems for Ubiquitous Computing. *Journal of Computer, 34*(8), 57–66.

Ho, K., & Weng, H. (2013). Favoured Attributes of In-Air Gestures in the Home Environment. In *OzCHI 2013: Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration* (pp. 171–174). Adelaide, Australia.

Hoffman, C. (2014). Wi-Fi Security: Should You Use WPA2-AES, WPA2-TKIP, or Both? Retrieved September 14, 2015, from http://www.howtogeek.com/204697/wi-fi-security-should-you-use-wpa2-aes-wpa2-tkip-or-both/

Hofstee, E. (2006). *Constructing a Good Dissertation: A Practical Guide to Finishing a Master's, MBA or PhD on Schedule*. Johannesburg, South Africa: Exactica.

Hollinworth, N., & Hwang, F. (2011). Investigating Familiar Interactions to Help Older Adults Learn Computer Applications More Easily. In *BCS-HCI 2011: Proceedings of the 25th British Computer Society Conference on Human-Computer Interaction* (pp. 473–478). Newcastle Upon Tyne, United Kingdom.

Huber, D. (2011). Background Positioning for Mobile Devices - Android vs. iPhone. In *INFOCOM 2011: Proceedings of the 30th IEEE International Conference on Computer and Communications* (pp. 1–7). Shanghai, China.

Hung, V. C., & Gonzalez, A. J. (2013). Context-Centric Speech-Based Human-Computer Interaction. *International Journal of Intelligent Systems*, *28*(10), 1010–1037.

Ilić, S. (2010). Tracking and Detection in Computer Vision: Camera Models and Pose Estimation. Retrieved December 9, 2015, from http://campar.in.tum.de/twiki/pub/Chair/TeachingWs09MATDCV/camera_models_and_pose_estimation.pdf

Indulska, M., & Recker, J. (2008). Design Science in IS Research: A Literature Analysis. In *ANU 2008: 4th Biennial Australian National University Workshop on Information Systems Foundations* (pp. 1–13). Canberra, Australia.

Intel Corporation. (2015). Intel® RealSense™ Camera. Retrieved October 30, 2015, from http://www.intel.com/content/www/us/en/architecture-and-technology/realsense-depth-camera.html

International Organisation for Standardisation. (1994). ISO 5725-1:1994 Accuracy (Trueness and Precision) of Measurement Methods and Results. Retrieved November 7, 2015, from https://www.iso.org/obp/ui/#iso:std:iso:5725:-1:ed-1:v1:en

Jain, J., Lund, A., & Wixon, D. (2011). The Future of Natural User Interfaces. In *CHI EA 2011: Extended Abstracts on Human Factors in Computing Systems* (pp. 211–214). Vancouver, Canada.

Johannesson, P., & Perjons, E. (2012). *A Design Science Primer* (1st ed.). Lexington, KY: CreateSpace.

Johannesson, P., & Perjons, E. (2014). *An Introduction to Design Science* (1st ed.). Zug, Switzerland: Springer International.

Joo, S. (2010). How Are Usability Elements - Efficiency, Effectiveness, and Satisfaction - Correlated with Each Other in the Context of Digitial Libraries? In *ASIST 2010: Proceedings of the American Society for Information Science and Technology* (pp. 1–2). Pittsburgh, PA.

Juniper Networks. (2015). Understanding WPA-PSK and WPA2-PSK Authentication. Retrieved September 14, 2015, from http://www.juniper.net/techpubs/en_US/network-director1.1/topics/concept/wireless-wpa-psk-authentication.html

Kahai, S. (2008). Leading in Face-to-Face Versus Virtual Teams. Retrieved October 24, 2015, from http://www.leadingvirtually.com/leading-in-face-to-face-versus-virtual-teams/

Kamal, A., Li, Y., & Lank, E. (2014). Teaching Motion Gestures via Recognizer Feedback. In *IUI 2014: Proceedings of the 19th International Conference on Intelligent User Interfaces* (pp. 73–82). Haifa, Israel.

Kaspersky Lab. (2012). Digital Consumer's Online Trends and Risks. Retrieved September 24, 2015, from http://www.kaspersky.com/downloads/pdf/kaspersky_lab_consumer_survey_report_eng_final.pdf

Kaushik, M., & Jain, R. (2014). Natural User Interfaces: Trend in Virtual Interaction. *IJLTEMAS 2014: International Journal of Latest Technology in Engineering, Management, and Applied Science*, *3*(4), 141–143.

Kin, K., Agrawala, M., & DeRose, T. (2009). Determining the Benefits of Direct-Touch, Bimanual, and Multi-Finger Input on Multi-Touch Workstation. In *GI 2009: Proceedings of the 35th International Conference on Graphics Interface* (pp. 119–124). British Columbia, Canada.

Kiruthika, M., Smita, D., & Dhanashree, H. (2009). Sharing Internet Connection Through Wi-Fi Network. In *ICAC3 2009: Proceedings of the International Conference on Advances in Computing, Communication, and Control* (pp. 659–663). Mumbai, India.

Koueider, A. (2013). Gesture-Based Interaction: The Future of Input Technology. Retrieved October 28, 2015, from http://www.androidauthority.com/gesture-based-interaction-future-input-technology-206986/

Kratz, S., Rohs, M., & Essl, G. (2013). Combining Acceleration and Gyroscope Data for Motion Gesture Recognition using Classifiers with Dimensionality Constraints. In *IUI 2013: Proceedings of the ACM International Conference on Intelligent User Interfaces* (pp. 173–178). Santa Monica, CA.

Krueger, R. A., & Casey, M. A. (2014). *Focus Groups: A Practical Guide for Applied Research* (5th ed.). Thousand Oaks, CA: SAGE Publications.

Krummelbein, M., & Nuur, I. (2013). *The Myth of Natural User Interfaces : How Contextual Understanding and Existing Knowledge Lead Towards Natural Interaction*. Aarhus University.

Kumar, S. (2015). What is Prototype Model- Advantages, Disadvantages and When to Use It? Retrieved September 13, 2015, from http://istqbexamcertification.com/what-is-prototype-model-advantages-disadvantages-and-when-to-use-it/

Küpper, A. (2005). *Location-Based Services: Fundamentals and Operation*. Chichester, England: John Wiley & Sons.

Kushalnagar, R. S., Behm, G. W., Kelstone, A. W., & Ali, S. (2015). Tracked Speech-To-Text Display: Enhancing Accessibility and Readability of Real-Time Speech-To-Text. In *ASSETS 2015: Proceedings of the 17th International ACM SIGACCESS Conference on Computers and Accessibility* (pp. 223–230). Lisbon, Portugal.

Lamberti, F., Sanna, A., Carlevaris, G., & Demartini, C. (2015). Adding Pluggable and Personalised Natural Control Capabilities to Existing Applications. *Sensors 2015*, *15*(2), 2832–2859.

Leap Motion Inc. (2015). Leap Motion. Retrieved October 30, 2015, from https://www.leapmotion.com/

Lee Son, T., Wesson, J., & Vogts, D. (2015). An Environment to Support Gesture-Based Information Sharing Among Co-Located Mobile Devices. In *SATNAC 2015: Proceedings of the Southern Africa Telecommunication Networks and Applications Conference* (pp. 225–230). Stellenbosch, South Africa.

Lee, Y. J. (2014). Android Sensor Signal Processing using Acceleration Sensor for Smart Phone. In *ASTL 2014: Proceedings of the International Conference on Advanced Computer and Information Technology* (Vol. 51, pp. 188–191). Saipan, CNMI: Science and Engineering Research Support Society.

Lee, Y. J. (2015a). Detection of Movement and Shake Information using Android Sensor. In *ASTL 2015: Proceedings of International Workshops on Mechanical Engineering and Multimedia* (Vol. 90, pp. 52–56). Jeju Island, Korea: Science and Engineering Research Support Society.

Lee, Y. J. (2015b). New Interface Design using Accelerometer Sensor for Smartphone. *International Journal of Smart Home*, *9*(1), 1–12.

Lewis, J. R. (1995). IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use. *International Journal of Human-Computer Interaction*, *7*(1), 57–78.

Lewis, J. R. (2002). Psychometric Evaluation of the PSSUQ Using Data from Five Years of Usability Studies. *International Journal of Human-Computer Interaction*, *14*(3), 463–488.

Li, Y. (2010). Gesture Search: A Tool for Fast Mobile Data Access. In *UIST 2010: Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology* (pp. 87–96). New York, NY.

Lightfoot, A. (2010). *A Parallel of Words*. Bloomington, IN: AuthorHouse.

Lilian, S. C. (2014). Virtual Teams: Opportunities and Challenges for e-Leaders. *Procedia - Social and Behavioral Sciences*, *110*, 1251–1261.

Lim, Y.-K., Stolterman, E., & Tenenberg, J. (2008). The Anatomy of Prototypes: Prototypes as Filters, Prototypes as Manifestations of Design Ideas. *Journal of ACM Transactions on Computer-Human Interaction*, *15*(2), 1–27.

Linnhoff-Popien, C., Marcus, P., & Küpper, A. (2010). Positioning. Retrieved September 17, 2015, from http://idp-social-marketplace.googlecode.com/svn-history/r1635/trunk/Quellen/chapter_positioning.pdf

Linnhoff-Popien, C., Marcus, P., & Schönfeld, M. (2015). Outdoor Positioning Systems. Retrieved September 17, 2015, from http://www.mobile.ifi.lmu.de/wp-content/uploads/2015/03/02_gps.pdf

LiveQoS. (2015). SuperBeam | Quick and Easy File Sharing. Retrieved October 25, 2015, from http://superbe.am/

Loureiro, B., & Rodrigues, R. (2011). Multi-Touch as a Natural User Interface for Elders: A Survey. In *CISTI 2011: Proceedings of the 6th Iberian Conference on Information Systems and Technologies* (pp. 1–6). Chaves, Portugal.

Lü, H., & Li, Y. (2015). Gesture On: Enabling Always-On Touch Gestures for Fast Mobile Access from the Device Standby Mode. In *CHI 2015: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (pp. 3355–3364). New York, NY.

March, S. T., & Smith, G. F. (1995). Design and Natural Science Research on Information Technology. *Journal of Decision Support Systems*, *15*(4), 251–266.

Marquardt, N. (2013). Proxemic Interactions with and Around Digital Surfaces. In *ITS 2013: Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces* (pp. 493–494). St. Andrews, United Kingdom.

MathsIsFun.com. (2015). Accuracy and Precision. Retrieved November 7, 2015, from http://www.mathsisfun.com/accuracy-precision.html

Maxwell, J. A. (2012). *Qualitative Research Design: An Interactive Approach*. Thousand Oaks, CA: SAGE Publications.

Mesmer-Magnus, J. R., & DeChurch, L. A. (2009). Information Sharing and Team Performance: A Meta-Analysis. *Journal of Applied Psychology*, *94*(2), 535–546.

Michahelles, F. (2011). Mobile Devices - Present and Future Capabilities. Retrieved October 2, 2015, from http://www.im.ethz.ch/people/fmichahelles/talks/post_bern.pdf

Microsoft. (2010). Natural User Interface: Voice, Touch and Beyond. Retrieved October 26, 2015, from http://news.microsoft.com/2010/01/06/natural-user-interfaces-voice-touch-and-beyond/

Microsoft. (2015a). Experience Things in a Whole New Way. Retrieved October 28, 2015, from http://www.microsoft.com/en-us/pixelsense/whatissurface.aspx

Microsoft. (2015b). Kinect - Windows App Development. Retrieved October 30, 2015, from https://dev.windows.com/en-us/kinect

Microsoft. (2015c). Windows Phones. Retrieved October 27, 2015, from http://www.microsoft.com/en-us/windows/phones

Microsoft. (2015d). Xbox 360. Retrieved October 28, 2015, from http://www.xbox.com/en-ZA/xbox-360?xr=shellnav

Mifsud, J. (2015). Usability Metrics - A Guide to Quantify the Usability of Any System. Retrieved September 30, 2015, from http://usabilitygeek.com/usability-metrics-a-guide-to-quantify-system-usability/

Mirchandani, C. (2000). *Incremental Prototyping, a Model-Based Methodology for Optimal System Design*. College Park, MD: University of Maryland.

Mitchell, B. (2015). The Origin of Wardriving Wireless Networks. Retrieved September 17, 2015, from http://compnetworking.about.com/cs/wireless/g/bldef_wardrive.htm

Moore, G. E. (1970). Moore's Law. Retrieved October 26, 2015, from http://www.mooreslaw.org/

Neilson, S. (2013). Mobile Network Geolocation: Obtaining the Cell IDs and the Signal Strength of Surrounding Towers From a GSM Modem & Triangulating Device Location. Retrieved November 14, 2015, from http://www.neilson.co.za/mobile-network-geolocation-obtaining-the-cell-ids-the-signal-strength-of-surrounding-towers-from-a-gsm-modem/

Nintendo. (2015). Wii U GamePad. Retrieved October 28, 2015, from https://www.nintendo.co.za/Wii-U/Hardware-Features/Wii-U-GamePad/Wii-U-GamePad-663028.html

Nirjon, S., Dickerson, R., Stankovic, J., Shen, G., & Jiang, X. (2013). sMFCC: Exploiting Sparseness in Speech for Fast Acoustic Feature Extraction on Mobile Devices – A Feasibility Study. In *HotMobile 2013: Proceedings of the 14th Workshop on Mobile Computing Systems and Applications* (pp. 1–6). Jekyll Island, GA.

Offermann, P., Blom, S., Schönherr, M., & Bub, U. (2010). Artefact Types in Information Systems Design Science - A Literature Review. In *DESRIST 2010: Proceedings of the 5th International Conference on Design Research in Information Systems and Technology* (pp. 77–92). St. Gallen, Switzerland.

Oh, J., Robinson, H. R., & Lee, J. Y. (2013). Page Flipping vs. Clicking: The Impact of Naturally Mapped Interaction Technique on User Learning and Attitudes. *Journal of Computers in Human Behavior*, *29*(4), 1334–1341.

Olivier, M. S. (2009). *Information Technology Research - A Practical Guide for Computer Science and Informatics*. Pretoria, South Africa: Van Schaik.

Oracle. (2015). What Is a Socket? Retrieved September 15, 2015, from https://docs.oracle.com/javase/tutorial/networking/sockets/definition.html

Oriyano, S.-P., & Doherty, J. (2015). *Wireless and Mobile Device Security*. Burlington, MA: Jones and Bartlett.

Ostrowski, L., & Helfert, M. (2012). Design Science Evaluation – Example of Experimental Design. *Journal of Emerging Trends in Computing and Information Sciences*, *3*(9), 253–262.

Paek, J., Kim, J., & Govindan, R. (2010). Energy-Efficient Rate-Adaptive GPS-Based Positioning for Smartphones. In *MobiSys 2010: Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services* (pp. 299–314). San Francisco, CA.

Payne, T. (2015). What is a Pilot Study? Definition and Example. Retrieved October 8, 2015, from http://study.com/academy/lesson/what-is-a-pilot-study-definition-example.html

Peffers, K., Rothenberger, M., Tuunanen, T., & Vaezi, R. (2012). Design Science Research Evaluation. In *DESRIST 2012: Proceedings of the 7th International Conference on Design Research in Information Systems and Technology* (pp. 398–410). Las Vegas, NV.

Peffers, K., Tuunanen, T., Rothenberger, M., & Chatterjee, S. (2008). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, *24*(3), 45–77.

Petter, S., Khazanchi, D., & Murphy, J. D. (2010). A Design Science Based Evaluation Framework for Patterns. *Journal of Advances in Information Systems*, *41*(3), 9–26.

Rautaray, S., & Agrawal, A. (2015). Vision-Based Hand Gesture Recognition for Human-Computer Interaction: A Survey. *Journal of Articial Intelligence Review*, *43*(1), 1–54.

Rodríguez, P., Kuvaja, P., & Oivo, M. (2014). Lessons Learned on Applying Design Science for Bridging the Collaboration Gap between Industry and Academia in Empirical Software Engineering. In *CESI 2014: Proceedings of the 2nd International Workshop on Conducting Empirical Studies in Industry* (pp. 9–14). Hyderabad, India.

Rudzicz, F., Wang, R., Begum, M., & Mihailidis, A. (2015). Speech Interaction with Personal Assistive Robots Supporting Aging at Home for Individuals with Alzheimer's Disease. *TACCESS 2015: Journal of ACM Transactions on Accessible Computing*, *7*(2), 1–22.

Ruiz, J., Li, Y., & Lank, E. (2011). User-Defined Motion Gestures for Mobile Interaction. In *CHI 2011: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 197–206). Vancouver, Canada.

Rukzio, E., Broll, G., Leichtenstern, K., & Schmidt, A. (2007). Mobile Interaction with the Real World: An Evaluation and Comparison of Physical Mobile Interaction Techniques. In *AmI 2007: Proceedings of Ambient Intelligence European Conference* (pp. 1–18). Darmstadt, Germany.

Saab, D. J., Orendovici, R., Van Gorp, A., Maitland, C., & Maldonado, E. (2008). Building Global Bridges: Coordination Bodies for Improved Information Sharing among Humanitarian Relief Agencies. In *ISCRAM 2008: Proceedings of the 5th International Conference for Crisis Response and Management* (pp. 471–483). Washington, DC.

Samsung. (2012). Samsung Galaxy S3. Retrieved September 14, 2015, from http://www.samsung.com/global/galaxys3/

Samsung. (2013). Samsung Galaxy S4. Retrieved September 14, 2015, from http://www.samsung.com/global/microsite/galaxys4/

Samsung. (2014). Samsung Galaxy S5. Retrieved September 14, 2015, from http://www.samsung.com/global/microsite/galaxys5/

Samsung. (2015). Samsung Galaxy S6. Retrieved October 27, 2015, from http://www.samsung.com/za/consumer/mobile-devices/smartphones/galaxy-s/SM-G920FZDAXFA

Sauro, J., & Lewis, J. R. (2012). *Quantifying the User Experience: Practical Statistics for User Research*. Waltham, MA: Elsevier.

Schmidt, D., Chehimi, F., Rukzio, E., & Gellersen, H. (2010). PhoneTouch: A Technique for Direct Phone Interaction on Surfaces. In *UIST 2010: Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology* (pp. 13–16). New York, NY.

Seifert, J., Simeone, A. L., Schmidt, D., Reinartz, C., Holleis, P., Wagner, M., ... Rukzio, E. (2012). MobiSurf: Improving Co-Located Collaboration through Integrating Mobile Devices and Interactive Surfaces. In *ITS 2012: Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces* (pp. 51–60). Cambridge, MA.

Seow, S. C., Morrison, A., Wixon, D., & Jacucci, G. (2010). Natural User Interfaces: The Prospect and Challenge of Touch and Gestural Computing. In *CHI 2010: Proceedings of the 28th International Conference on Human Factors in Computing Systems* (pp. 4453–4456). Atlanta, GA.

Sergeev, A. (2010a). ISO-9241 Effectiveness Metric - Theory of Usability. Retrieved September 30, 2015, from http://ui-designer.net/usability/effectiveness.htm

Sergeev, A. (2010b). ISO-9241 Efficiency Metric - Theory of Usability. Retrieved September 30, 2015, from http://ui-designer.net/usability/efficiency.htm

Sergeev, A. (2010c). ISO-9241 Satisfaction Metric - Theory of Usability. Retrieved September 30, 2015, from http://ui-designer.net/usability/satisfaction.htm

Shahriar, S., Kamruzzaman, M., & Beecham, S. (2014). Multiple Resolution River Flow Time Series Modelling using Machine Learning Methods. In *MLSDA 2014: Proceedings of the 2nd Workshop on Machine Learning for Sensory Data Analysis* (pp. 1–5). Gold Coast, Australia.

Shenzhen Youmi Information Technology Co. Ltd. (2015). FileMaster - File Manager. Retrieved October 25, 2015, from https://itunes.apple.com/za/app/filemaster-file-manager-privacy/id582219355?mt=8

Singleton, A. (2014). Distributed Teams | Co-Located, Outsourced, Distributed. Retrieved October 24, 2015, from http://www.continuousagile.com/unblock/team_options.html

Socha, W. (2015). Broadcasting Power, RSSI, and Measured Power Explained. Retrieved September 19, 2015, from https://forums.estimote.com/t/broadcasting-power-rssi-and-measured-power-explained/1047

Sokolova, M., & Lapalme, G. (2009). A Systematic Analysis of Performance Measures for Classification Tasks. *International Journal of Information Processing and Management*, *45*(4), 427–437.

Song, J., Sörös, G., Pece, F., Fanello, S. R., Izadi, S., Keskin, C., & Hilliges, O. (2014). In-Air Gestures Around Unmodified Mobile Devices. In *UIST 2014: Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (pp. 319–329). Honolulu, HI.

Sørensen, H., Kristensen, M. G., Kjeldskov, J., & Skov, M. B. (2013). Proxemic Interaction in Multi-Room Music System. In *OzCHI 2013: Proceedings of the 25h Australian Computer-Human Interaction Conference on Augmentation, Application, Innovation, and Collaboration* (pp. 153–162). Adelaide, Australia.

Steinberg, G. (2012). Natural User Interfaces. In *ACM SIGCHI 2012: Proceedings of the Conference on Human Factors in Computing Systems* (pp. 1–4). Austin, TX.

Stephanidis, C., & Antona, M. (2013). *Universal Access in Human-Computer Interaction: Applications and Services for Quality of Life*. Berlin, Germany: Springer.

Stewart, D. W., & Shamdasani, P. N. (2014). *Focus Groups: Theory and Practice*. Thousand Oaks, CA: SAGE Publications.

Stufflebeam, D. L., & Coryn, C. L. S. (2014). *Evaluation Theory, Models, and Applications* (2nd ed.). San Francisco, CA: John Wiley and Son.

Tavakol, M., & Dennick, R. (2011). Making Sense of Cronbach's Alpha. *International Journal of Medical Education*, *2*, 53–55.

Tavares, T. A., Medeiros, A., De Castro, R., & Dos Anjos, E. (2013). The Use of Natural Interaction to Enrich the User Experience in Telemedicine Systems. In *HCI International 2013 - Posters' Extended Abstracts* (pp. 220–224). Las Vegas, NV: Springer.

Taylor, C. (2013). Could You Live Without Technology? Retrieved October 25, 2015, from http://reasondigital.com/advice-and-training/could-you-live-without-technology/

Tchankue, P., Wesson, J., & Vogts, D. (2014). *A Model for Mobile, Context-Aware In-Car Communication Systems to Reduce Driver Distraction*. Nelson Mandela Metropolitan University.

Thomason, J., & Wang, J. (2012). Exploring Multi-Dimensional Data on Mobile Devices with Single Hand Motion and Orientation Gestures. In *MobileHCI 2012: Proceedings of the 14th International Conference on Human-Computer Interaction with Mobile Devices and Services Companion* (pp. 173–176). San Francisco, CA.

Tiwary, U. S., & Siddiqui, T. J. (2012). *Speech, Image, and Language Processing for Human-Computer Interaction: Multi-Modal Advancements*. Hershey, PA: IGI Global.

Trochim, W. (2006). Introduction to Evaluation. Retrieved October 8, 2015, from http://www.socialresearchmethods.net/kb/intreval.php

Tullis, T., & Albert, W. (2013). *Measuring the User Experience: Collecting, Analysing, and Presenting Usability Metrics* (2nd ed.). Amsterdam, Netherlands: Elsevier.

User Experience Professionals' Association. (2012a). Formative Evaluation. Retrieved October 6, 2015, from http://www.usabilitybok.org/formative-evaluation

User Experience Professionals' Association. (2012b). Summative Evaluation. Retrieved October 6, 2015, from http://www.usabilitybok.org/summative-evaluation

Vaishnavi, V. K., & Kuechler, B. (2008). Design Science Research in Information Systems. Retrieved September 28, 2015, from http://desrist.org/desrist/content/design-science-research-in-information-systems.pdf

Vaishnavi, V. K., & Kuechler, W. (2015). *Design Science Research Methods and Patterns: Innovating Information and Communication Technology* (2nd ed.). Boca Raton, FL: CRC Press.

Valli, A. (2008). The Design of Natural Interaction. *Journal of Multimedia Tools and Applications*, *38*(3), 295–305.

Van Teijlingen, E., & Hundley, V. (2001). The Importance of Pilot Studies. Retrieved October 14, 2015, from http://sru.soc.surrey.ac.uk/SRU35.pdf

Venable, J. (2006). A Framework for Design Science Research Activities. In *Emerging Trends and Challenges in Information Technology Management* (pp. 184–187). Washington, DC.

Venable, J., Pries-Heje, J., & Baskerville, R. (2012). A Comprehensive Framework for Evaluation in Design Science Research. In *DESRIST 2012: Proceedings of the 7th International Conference on Design Research in Information Systems and Technology* (pp. 423–438). Las Vegas, NV.

Villamor, C., Willis, D., Wroblewski, L., & Fulton, K. (2011). Touch Gesture Reference Cards. Retrieved October 27, 2015, from http://static.lukew.com/TouchGestureCards.pdf

Wagner, P., Malisz, Z., & Kopp, S. (2014). Gesture and Speech in Interaction: An Overview. *Journal of Speech Communication*, *57*, 209–232.

Wang, J. (2013). *Information Systems and Modern Society: Social Change and Global Development*. Hershey, PA: IGI Global.

Wang, J.-F., Furui, S., & Juang, B.-H. (2013). *Real World Speech Processing*. Boston, MA: Springer.

Wang, S., Min, J., & Yi, B. K. (2008). Location-Based Services for Mobiles: Technologies and Standards. In *ICC 2008: Proceedings of the IEEE International Conference on Communication* (pp. 35–38). Beijing, China.

Wendt, T. (2013). Designing for Transparency and the Myth of the Modern Interface. Retrieved November 2, 2015, from https://uxmag.com/articles/designing-for-transparency-and-the-myth-of-the-modern-interface

Wigdor, D., & Wixon, D. (2011). *Brave NUI World: Designing Natural User Interfaces for Touch and Gesture*. Burlington, MA: Elsevier.

Witten, I. H., & Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques* (2nd ed.). San Francisco, CA: Elsevier.

Yang, J., Varshavsky, A., Liu, H., Chen, Y., & Gruteser, M. (2010). Accuracy Characterization of Cell Tower Localization. In *UbiComp 2010: Proceedings of the 12th International Conference on Ubiquitous Computing* (pp. 223–226). Copenhagen, Denmark.

Yao, J., Fernando, T., & Wang, H. (2012). A Multi-Touch Natural User Interface Framework. In *ICSAI 2012: Proceedings of the International Conference on Systems and Informatics* (pp. 499–504). Yantai, China.

Ydangle Apps. (2013). Welcome to Flick. Retrieved November 15, 2015, from http://getflick.io/

Zandbergen, P. (2014). How Accurate is the GPS on my Smart Phone? Retrieved May 13, 2015, from http://communityhealthmaps.nlm.nih.gov/2014/07/07/how-accurate-is-the-gps-on-my-smart-phone-part-2/

Zhang, M., & Sawchuk, A. A. (2012). A Preliminary Study of Sensing Appliance Usage for Human Activity Recognition using Mobile Magnetometer. In *UbiComp 2012: Proceedings of the 14th International Conference on Ubiquitous Computing* (pp. 745–748). Pittsburgh, PA.

Zhang, T., Van Den Berg, E., Madhani, S., Dutta, A., & Mohanti, S. (2007). Mobile Information Services Enabled By Mobile Publishing. *Journal of Wireless Networks*, *13*(5), 617–633.

Appendix A: Research Ethics Approval

Nelson Mandela Metropolitan University

for tomorrow

• PO Box 77000 • Nelson Mandela Metropolitan University
• Port Elizabeth • 6031 • South Africa • www.nmmu.ac.za

Chairperson: Research Ethics Committee (Human)
Tel: +27 (0)41 504-2235

Ref: [H15-SCI-CSS-004 /Approval]

Contact person: Mrs U Spies

21 July 2015

Prof J Wesson
Faculty: Science
Department: Computing Sciences
Embizweni Building, Room 02-13
South Campus

Dear Prof Wesson

DESIGNING A NATURAL USER INTERFACE TO SUPPORT INFORMATION SHARING AMONG CO-LOCATED MOBILE DEVICES

PRP: Prof J Wesson
PI: Mr T Lee Son

Your above-entitled application served at Research Ethics Committee (Human) for extension.

We take pleasure in informing you that the extension was approved by the Committee.

The ethics clearance reference number is **H15-SCI-CSS-004** and is valid for three years. Please inform the REC-H, via your faculty representative, if any changes (particularly in the methodology) occur during this time. An annual affirmation to the effect that the protocols in use are still those for which approval was granted, will be required from you. You will be reminded timeously of this responsibility, and will receive the necessary documentation well in advance of any deadline.

We wish you well with the project. Please inform your co-investigators of the outcome, and convey our best wishes.

Yours sincerely

CB Cilliers

Prof C Cilliers
Chairperson: Research Ethics Committee (Human)

cc: Department of Research Capacity Development
Faculty Officer: Science

Appendix B: Consent Form (Focus Group)

NELSON MANDELA METROPOLITAN UNIVERSITY
INFORMATION AND INFORMED CONSENT FORM

RESEARCHER'S DETAILS	
Title of the research project	Designing a Proxemic Natural User Interface to Support Information Sharing Among Co-Located Mobile Devices
Reference number	H15-SCI-CSS-004
Principal investigator	Timothy Lee Son
Contact telephone number (private numbers not advisable)	

A. DECLARATION BY OR ON BEHALF OF PARTICIPANT		Initial
I, the participant and the undersigned		

A.1 HEREBY CONFIRM AS FOLLOWS:		Initial
I, the participant, was invited to participate in the above-mentioned research project		
that is being undertaken by	Timothy Lee Son	
From the	Department of Computing Sciences	
of the Nelson Mandela Metropolitan University.		

THE FOLLOWING ASPECTS HAVE BEEN EXPLAINED TO ME, THE PARTICIPANT:					Initial
2.1	Aim:	The investigators are investigating and evaluating the usability of a proxemic natural user interface to support information sharing among co-located mobile devices. The information will be used to / for research purposes.			
2.2	Confidentiality:	My identity will not be revealed in any discussion, description or scientific publications by the investigators.			
2.3	Access to findings:	Any new information or benefit that develops during the course of the study will be shared as follows: published in papers and dissertation			
2.4	Voluntary participation / refusal / discontinuation:	My participation is voluntary in this usability evaluation	YES	NO	
		My decision on whether or not to participate will in no way affect my studies/employment at NMMU	TRUE	FALSE	

3.	No pressure was exerted on me to consent to participation and I understand that I may withdraw at any stage without penalisation.	
4.	Participation in this study will not result in any additional cost to myself.	
5.	Permission to record usability evaluation using a Dictaphone and/or video camera	

A.2 I HEREBY VOLUNTARILY CONSENT TO PARTICIPATE IN THE ABOVE-MENTIONED PROJECT:			
Signed/confirmed at		on	2015
Signature	Signature of witness:		
	Full name of witness:		

Appendix C: Focus Group Questions

Nelson Mandela Metropolitan University

for tomorrow

Focus Group Questions

1. What action would you perform to **select a file**?
2. What action would you perform to **select multiple files**?
3. What action would you perform to **share a file with another mobile device**?
4. What action would you perform to **share a file with multiple devices**?
5. What action would you perform to **initiate a file transfer**?
6. What action would you perform to **accept a file transfer**?
7. What action would you perform to **cancel a file transfer**?

Appendix D: Consent Form (Usability Evaluation)

NELSON MANDELA METROPOLITAN UNIVERSITY
INFORMATION AND INFORMED CONSENT FORM

RESEARCHER'S DETAILS	
Title of the research project	Designing a Proxemic Natural User Interface to Support Information Sharing Among Co-Located Mobile Devices
Reference number	H15-SCI-CSS-004
Principal investigator	Timothy Lee Son
Contact telephone number (private numbers not advisable)	

B. DECLARATION BY OR ON BEHALF OF PARTICIPANT		Initial
I, the participant and the undersigned		

A.1 HEREBY CONFIRM AS FOLLOWS:		Initial
I, the participant, was invited to participate in the above-mentioned research project		
that is being undertaken by	Timothy Lee Son	
From the	Department of Computing Sciences	
of the Nelson Mandela Metropolitan University.		

THE FOLLOWING ASPECTS HAVE BEEN EXPLAINED TO ME, THE PARTICIPANT:					Initial
2.1	Aim:	The investigators are investigating and evaluating the usability of a proxemic natural user interface to support information sharing among co-located mobile devices. The information will be used to / for research purposes.			
2.2	Confidentiality:	My identity will not be revealed in any discussion, description or scientific publications by the investigators.			
2.3	Access to findings:	Any new information or benefit that develops during the course of the study will be shared as follows: published in papers and dissertation			
2.4	Voluntary participation / refusal / discontinuation:	My participation is voluntary in this usability evaluation	YES	NO	
		My decision on whether or not to participate will in no way affect my studies/employment at NMMU	TRUE	FALSE	

3.	No pressure was exerted on me to consent to participation and I understand that I may withdraw at any stage without penalisation.	☐
4.	Participation in this study will not result in any additional cost to myself.	☐
5.	Permission to record usability evaluation using a Dictaphone and/or video camera	☐

A.2 I HEREBY VOLUNTARILY CONSENT TO PARTICIPATE IN THE ABOVE-MENTIONED PROJECT:		
Signed/confirmed at	on	2015
Signature	Signature of witness:	
	Full name of witness:	

Appendix E: Written Information Given to Participant

Written Information Given to Participant Prior to Participation

Dear Participant

You are being asked to participate in a research study. The researcher will provide you with the necessary information to assist you to understand the study and explain what would be expected of you as a participant. These guidelines would include the risks, benefits, and your rights as a study subject. Please feel free to ask the researcher to clarify anything that is not clear to you.

To participate, it will be required of you to provide a written consent that will include your signature, date, and initials to verify that you understand and agree to the conditions.

You have the right to query concerns regarding the study and please feel free to ask questions at any time. However, if at any time during the study, you wish to withdraw, you are welcome to do so. Immediately report any new problems during the study to the researcher. The researcher will be present throughout the full duration of the study.

The ethical integrity of the study has been approved by the Research Ethics Committee (Human) (REC-H) of Nelson Mandela Metropolitan University. The REC-H consists of a group of independent experts that has the responsibility to ensure that the rights and welfare of participants in the study are protected and that the studies are conducted in an ethical manner. Studies cannot be conducted without REC-H's approval. Queries with regard to your rights as a research subject can be directed to: *Research Ethics Committee (Human), Department of Research Capacity Development, PO Box 77000, Nelson Mandela Metropolitan University, Port Elizabeth, 6031.*

Your identity will remain confidential at all times. However, you might be referred to as “participant X”. The results of the research study may be presented at conference proceedings or journals. This informed consent statement has been prepared in compliance with current statutory guidelines.

Yours sincerely,

Timothy Lee Son

Research and Evaluator

Appendix F: Verbal Information Given to Participant

Verbal Information Given to Participant Prior to Participation

I, Timothy Lee Son, the Primary Investigator (PI) and Researcher will provide participants with a verbal introduction. The introduction will be given in English and will include:

- The participants' rights will be given to them, indicating that they are free to withdraw from the focus group or usability evaluation at any time.
- The purpose of the system, which the participants will evaluate as well as the purpose for the focus group or usability evaluation.
- Participants will be made aware that all the results from the focus group or usability evaluation will be used for academic purposes only.
- What is expected from the participants during the focus group or usability evaluation. This includes the signing of the consent forms (Appendix B or D), a verbal and written (Appendix E) introduction to the focus group and/or usability evaluation, completion of the biographical form and the Post-Study System Usability Questionnaire.
- The basic system functionality will be explained and participants will be given a chance to familiarise themselves with the system and the setup.
- Any questions the participants might have will be answered verbally by the PI.

Appendix G: User Tasks

September 2015

Participant ☐

MotionShare
User Tasks (Sender)

---

1. Selecting a single file to be shared
    1.1. Select the **folder** button on the Map screen
    1.2. Select the **Documents** button
    1.3. **Long press** to initiate file selection mode
    1.4. Select **any document** file to be shared
    1.5. Question: What is the **name** of the document file selected to be shared?
    Answer: ____

    Select the **Share** button after the document file has been selected

---

2. Share with a single user (point gesture)
    2.1 Orientate the device to be **pointing** in the direction of intended recipient
    2.2 Confirm the map has **readjusted** according to the device screen
    2.3 If the map does not correlate with the actual pointing of the device, then continue to move device until the screen has adjusted. Note the inaccuracy
    2.4 Select the **options** button on the Map screen
    2.5 Select the **Share** button
    Question: What is the **name** of the intended recipient?

    Answer: ____

---

3. Selecting multiple files to be shared
    3.1 Select the **folder** button on the Map screen
    3.2 Select the **Images** button
    3.3 **Long press** to initiate the file selection mode
    3.4 Select **any number** of image files to be shared
    3.5 Question: What are the **names** of the image files selected to be shared?
    Answer: ____

    Select the **Share** button after the image files have been selected

4. Share with multiple users (point gesture)
   4.1. Orientate the device to be **pointing** in the direction of the intended recipients
   4.2. Confirm the map has **readjusted** according to the device screen
   4.3. If the map does not correlate with the actual pointing of the device, then continue to move device until the screen has adjusted. Note the inaccuracy
   4.4. Select the **options** button on the Map screen
   4.5. Select the **Share** button
   Question: What are the **names** of the intended recipients?
   Answer: ____

5. Selecting a single file to be shared
   5.1. Select the **folder** button on the Map screen
   5.2. Select the **Images** button
   5.3. **Long press** to initiate file selection mode
   5.4. Select **any image** file to be shared
   5.5. Question: What is the **name** of the image file selected to be shared?
   Answer: ____
   5.6. Select the **Share** button after the image file has been selected

6. Share with a single user (touch gesture)
   6.1. Select the  button
   Note: Red = gesture detection mode is off and Green = gesture detection mode is on
   6.2. Draw on the screen using your finger (by **pressing down and dragging**) and highlighting the intended recipient
   6.3. Initiate file sending by **pressing up** or releasing your finger from the screen
   6.4. Question: What is the **name** of the intended recipient?
   Answer: ____

7. Selecting multiple files to be shared
   7.1. Select the **folder** button on the Map screen
   7.2. Select the **Documents** button
   7.3. **Long press** to initiate the file selection mode
   7.4. Select **any number** of document files to be shared
   7.5. Question: What are the **names** of the document files selected to be shared?
   Answer: ____
   Select the **Share** button after the image files have been selected

8. Share with multiple users (touch gesture)
   8.1. Select the  button
   Note: Red = gesture detection mode is off and Green = gesture detection mode is on
   8.2. Draw on the screen using your finger (by **pressing down and dragging**) and highlighting the intended recipients
   8.3. Initiate file sending by **pressing up** or releasing your finger from the screen
   8.4. Question: What is the **name** of the intended recipients?
   Answer: ____

September 2015 Participant ☐

MotionShare
User Tasks (Receiver)

1. Receiving file(s)
   1.1. Notification message will be **displayed** on receipt of file(s)
   1.2. Select the **folder** button on the Map screen
   1.3. Select the **file type** (e.g. Documents) option
   1.4. Confirm **new** addition of received file(s)
   1.5. Question: What is the **type** of file received and **name** of the received file(s)?
   (for example: documents; S5(1)-1.txt, S5(1)-2.docx)

   Answer: ____

   1.6. Select the hardware **back** button on the device

---

2. Receiving file(s)
   2.1. Question: What is the **type** of file received and **name** of the received file(s)?
   (for example: documents; S5(1)-1.txt, S5(1)-2.docx)

   Answer: ____

   2.2. Select the hardware **back** button on the device

---

3. Receiving file(s)
   3.1. Question: What is the **type** of file received and **name** of the received file(s)?
   (for example: documents; S5(1)-1.txt, S5(1)-2.docx)

   Answer: ____

   3.2. Select the hardware **back** button on the device

Appendix H: Post-Study System Usability Questionnaire

September 2015 Participant ☐

MotionShare
Post-Study System Usability Questionnaire

Section A: Biographical Information

Place X in the appropriate box

1. Gender

   ☐ Male ☐ Female

2. Age (in years)

   ☐ 18 - 21 ☐ 22 - 30 ☐ 31 - 40 ☐ 41+

3. How many hours do you spend a day on any mobile device? (e.g. smartphone, tablet, laptop, or PDA)

   ☐ 0 ☐ 1-2 ☐ 3-4 ☐ 5-6 ☐ 7+

4. How many years have you been using a mobile device?

   ☐ 0 ☐ 1-2 ☐ 3-4 ☐ 5-6 ☐ 7+

5. Have you used a similar information sharing application before?

   ☐ Yes ☐ Maybe ☐ No

6. Would you consider using this application in the future?

   ☐ Yes ☐ Maybe ☐ No

7. Do you see potential applications of use with this particular application in the future?

   ☐ Yes ☐ Maybe ☐ No

8. What type of mobile phone or platform do you own?

   ☐ Android ☐ Blackberry ☐ iOS ☐ Symbian

Section B: System Usability

Place X in the appropriate box

1 = Strongly Disagree, 2 = Disagree, 3 = Slightly Disagree, 4 = Neutral, 5 = Slightly Agree, 6 = Agree,
7 = Strongly Agree

		1	2	3	4	5	6	7	N/A
1.	**Overall, I am satisfied with how easy it is to use this system**								
2.	**It was simple to use this system**								
3.	**I could effectively complete the tasks and scenarios using this system**								
4.	**I was able to complete the tasks and scenarios quickly using this system**								
5.	**I was able to efficiently complete the tasks and scenarios using this system**								
6.	**I felt comfortable using this system**								
7.	**It was easy to learn to use this system**								
8.	**I believe I could become productive quickly using this system**								
9.	**The system gave error messages that clearly told me to fix problems**								
10.	**Whenever I made a mistake using the system, I could recover easily and quickly**								
11.	**The information, such as online help, on-screen messages, and other documentation, provided with this system was clear**								
12.	**It was easy to find the information I needed**								
13.	**The information provided for the system was easy to understand**								
14.	**The information was effective in helping me complete the tasks and scenarios**								
15.	**The organisation of information on the system screens was clear**								
16.	**The interface of this system was pleasant**								
17.	**I liked using the interface of this system**								
18.	**This system has all the functions and capabilities I expect it to have**								
19.	**Overall, I am satisfied with this system**								

Section C: Overview

1. Please provide the most **positive** aspect(s) of the system

2. Please provide the most **negative** aspect(s) of the system

3. Please provide any **additional** comment(s) or recommendation(s)

Appendix I: Post-Test Questionnaire

September 2015

Participant [ ]

MotionShare
Post-Test Questionnaire

Section A: General

Place X in the appropriate box

1. Which NUI interaction technique will you **prefer to use** in the future?

Point Gesture	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**Touch Gesture**

Reason

____

____

2. Which NUI interaction technique did you find **more complex**?

Point Gesture	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**Touch Gesture**

Reason

____

____

3. Which NUI interaction technique did you find **easier to use**?

Point Gesture	1	2	3	4	5	6	7	Touch Gesture

Reason

____

____

4. Which NUI interaction technique would you **need help with to use**?

Point Gesture	1	2	3	4	5	6	7	Touch Gesture

Reason

____

____

5. Which NUI interaction technique did you find **more intuitive** for information sharing?

Point Gesture	1	2	3	4	5	6	7	Touch Gesture

Reason

____

____

6. Which NUI interaction technique is **most accurate**?

Point Gesture	1	2	3	4	5	6	7	Touch Gesture

Reason

____

____

7. Which NUI interaction technique do you think people would **learn to use** more quickly?

Point Gesture	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**Touch Gesture**

Reason

____

____

8. Which NUI interaction technique did you find **more cumbersome**?

Point Gesture	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**Touch Gesture**

Reason

____

____

9. Which NUI interaction technique did you feel **more confident using**?

Point Gesture	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**Touch Gesture**

Reason

____

____

Appendix J: Pilot Study PSSUQ Raw Results

No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	Total
1	7.00	7.00	6.00	7.00	7.00	7.00	7.00	6.00	4.00	7.00	7.00	7.00	7.00	6.00	7.00	6.00	7.00	6.00	7.00	125.00
2	6.00	6.00	7.00	6.00	6.00	6.00	6.00	7.00	5.00	6.00	6.00	6.00	6.00	7.00	6.00	7.00	6.00	7.00	6.00	118.00
3	6.00	6.00	5.00	6.00	6.00	6.00	6.00	5.00	5.00	6.00	7.00	6.00	6.00	5.00	6.00	6.00	6.00	5.00	6.00	110.00
4	7.00	6.00	7.00	6.00	7.00	6.00	7.00	6.00	6.00	6.00	7.00	6.00	7.00	6.00	7.00	6.00	7.00	6.00	7.00	123.00
5	6.00	6.00	5.00	6.00	6.00	6.00	6.00	6.00	4.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00	111.00
6	7.00	7.00	6.00	7.00	7.00	6.00	7.00	7.00	6.00	7.00	7.00	6.00	7.00	6.00	7.00	7.00	7.00	6.00	7.00	127.00
7	7.00	7.00	6.00	6.00	6.00	7.00	7.00	6.00	4.00	4.00	7.00	7.00	7.00	6.00	7.00	7.00	7.00	6.00	7.00	121.00
8	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	133.00
9	7.00	7.00	6.00	6.00	6.00	6.00	7.00	5.00	4.00	7.00	7.00	7.00	7.00	6.00	7.00	7.00	7.00	5.00	7.00	121.00
10	7.00	7.00	6.00	7.00	7.00	7.00	7.00	7.00	6.00	7.00	7.00	7.00	7.00	6.00	7.00	7.00	7.00	6.00	7.00	129.00
Mean	6.70	6.60	6.10	6.40	6.50	6.40	6.70	6.20	5.10	6.30	6.80	6.50	6.70	6.10	6.70	6.60	6.70	6.00	6.70	121.80
Median	7.00	7.00	6.00	6.00	6.50	6.00	7.00	6.00	5.00	6.50	7.00	6.50	7.00	6.00	7.00	7.00	7.00	6.00	7.00	122.00
Mode	7.00	7.00	6.00	6.00	7.00	6.00	7.00	6.00	4.00	7.00	7.00	7.00	7.00	6.00	7.00	7.00	7.00	6.00	7.00	121.00
Standard Deviation	0.48	0.52	0.74	0.52	0.53	0.52	0.48	0.79	1.10	0.95	0.42	0.53	0.48	0.57	0.48	0.52	0.48	0.67	0.48	7.36

Appendix K: Summative Evaluation PSSUQ Raw Results

No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	Total
1	5.00	6.00	4.00	5.00	6.00	5.00	6.00	6.00	1.00	3.00	4.00	4.00	4.00	4.00	5.00	5.00	4.00	6.00	5.00	88.00
2	5.00	4.00	7.00	5.00	5.00	6.00	6.00	6.00	4.00	6.00	4.00	4.00	4.00	6.00	6.00	6.00	6.00	6.00	7.00	103.00
3	4.00	3.00	7.00	5.00	6.00	5.00	2.00	5.00	4.00	4.00	7.00	4.00	5.00	5.00	6.00	7.00	7.00	6.00	7.00	99.00
4	5.00	5.00	6.00	4.00	5.00	3.00	3.00	5.00	5.00	6.00	6.00	6.00	6.00	5.00	5.00	5.00	4.00	5.00	6.00	95.00
5	5.00	4.00	6.00	6.00	6.00	5.00	6.00	6.00	4.00	6.00	6.00	6.00	6.00	6.00	7.00	5.00	6.00	5.00	6.00	107.00
6	6.00	5.00	6.00	6.00	6.00	5.00	6.00	7.00	4.00	6.00	5.00	7.00	5.00	6.00	6.00	6.00	6.00	6.00	6.00	110.00
7	7.00	6.00	7.00	6.00	6.00	7.00	7.00	7.00	4.00	7.00	6.00	6.00	6.00	7.00	6.00	7.00	7.00	7.00	7.00	123.00
8	6.00	6.00	7.00	5.00	7.00	6.00	6.00	7.00	4.00	6.00	6.00	6.00	7.00	7.00	7.00	7.00	7.00	6.00	6.00	119.00
9	6.00	6.00	6.00	7.00	7.00	6.00	6.00	6.00	4.00	7.00	6.00	6.00	6.00	7.00	6.00	7.00	6.00	5.00	6.00	116.00
10	5.00	5.00	6.00	6.00	6.00	7.00	6.00	6.00	4.00	6.00	5.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	118.00
11	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	5.00	6.00	5.00	5.00	6.00	125.00
12	3.00	4.00	6.00	4.00	4.00	5.00	6.00	6.00	2.00	6.00	6.00	7.00	7.00	7.00	7.00	6.00	6.00	7.00	6.00	105.00
13	6.00	6.00	7.00	5.00	7.00	7.00	7.00	7.00	6.00	6.00	7.00	6.00	7.00	7.00	7.00	6.00	7.00	7.00	7.00	125.00
14	5.00	5.00	5.00	4.00	5.00	4.00	4.00	6.00	3.00	3.00	4.00	4.00	5.00	5.00	5.00	4.00	5.00	6.00	5.00	87.00
15	6.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00	4.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00	112.00
16	5.00	6.00	6.00	5.00	6.00	5.00	6.00	7.00	4.00	6.00	7.00	6.00	6.00	6.00	7.00	7.00	7.00	7.00	6.00	115.00
17	6.00	5.00	6.00	6.00	6.00	6.00	4.00	5.00	3.00	5.00	7.00	5.00	6.00	6.00	7.00	7.00	6.00	6.00	6.00	108.00
18	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	133.00
19	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	133.00
20	6.00	6.00	7.00	7.00	6.00	6.00	6.00	5.00	4.00	7.00	5.00	5.00	5.00	5.00	5.00	5.00	6.00	6.00	6.00	108.00
21	6.00	6.00	6.00	6.00	6.00	7.00	7.00	7.00	4.00	6.00	6.00	7.00	7.00	6.00	6.00	5.00	6.00	6.00	6.00	116.00

22	3.00	4.00	6.00	4.00	4.00	5.00	6.00	6.00	4.00	6.00	6.00	7.00	7.00	7.00	7.00	6.00	6.00	7.00	6.00	107.00
23	7.00	6.00	7.00	6.00	6.00	7.00	7.00	7.00	5.00	7.00	6.00	6.00	6.00	7.00	6.00	7.00	7.00	7.00	7.00	124.00
24	6.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00	5.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00	113.00
25	5.00	6.00	6.00	5.00	6.00	5.00	6.00	7.00	5.00	6.00	7.00	6.00	6.00	6.00	7.00	7.00	7.00	7.00	6.00	116.00
26	6.00	5.00	6.00	6.00	6.00	6.00	4.00	5.00	7.00	5.00	7.00	5.00	6.00	6.00	7.00	7.00	6.00	6.00	6.00	112.00
27	6.00	6.00	7.00	5.00	7.00	7.00	7.00	7.00	6.00	6.00	7.00	6.00	7.00	7.00	7.00	6.00	7.00	7.00	7.00	125.00
28	7.00	7.00	6.00	5.00	7.00	6.00	5.00	4.00	7.00	6.00	6.00	7.00	6.00	6.00	6.00	6.00	7.00	6.00	6.00	116.00
29	7.00	7.00	6.00	6.00	6.00	7.00	6.00	6.00	6.00	6.00	7.00	6.00	6.00	7.00	7.00	6.00	6.00	7.00	7.00	122.00
30	6.00	6.00	7.00	5.00	7.00	7.00	7.00	7.00	5.00	5.00	6.00	6.00	6.00	6.00	7.00	6.00	7.00	7.00	7.00	120.00
31	6.00	7.00	6.00	7.00	6.00	6.00	6.00	6.00	7.00	7.00	6.00	7.00	6.00	6.00	6.00	7.00	7.00	6.00	7.00	122.00
32	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	6.00	6.00	6.00	6.00	5.00	6.00	7.00	7.00	7.00	7.00	7.00	126.00
Mean	5.75	5.69	6.31	5.66	6.09	5.97	5.88	6.22	4.75	5.91	6.06	5.97	6.03	6.22	6.34	6.22	6.28	6.31	6.34	114.00
Median	6.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00	4.00	6.00	6.00	6.00	6.00	6.00	6.50	6.00	6.00	6.00	6.00	116.00
Mode	6.00	6.00	6.00	6.00	6.00	7.00	6.00	7.00	4.00	6.00	6.00	6.00	6.00	6.00	7.00	7.00	7.00	6.00	6.00	116.00
Standard Deviation	1.08	1.06	0.69	0.97	0.82	1.03	1.24	0.83	1.52	1.03	0.91	0.97	0.86	0.79	0.75	0.83	0.85	0.69	0.60	11.39