Department of Computing Sciences

Using Natural User Interfaces to Support Synchronous Distributed Collaborative Work

Timothy Kyle Potgieter

Supervisor: Dr Dieter Vogts

Submitted in fulfilment of the requirements for the degree Magister Scientiae in the Faculty of Science at the Nelson Mandela Metropolitan University

December 2014

Declaration

I, Timothy Potgieter, hereby declare that the dissertation for the degree Magister Scientiae is my own work and that it has not previously been submitted for assessment or completion of any postgraduate qualification to another University or for another qualification.

Timothy Potgieter

Acknowledgements

I would like to thank the following people and organizations for their support in the completion of this project:

- Jesus Christ, my all-sufficient Saviour, who has carried me through to the end of this project.
- My beautiful wife, Roberta. Words cannot express my love and appreciation for you. The love, support and understanding you have shown me over the past two years have been amazing. You are truly wonderful and I love you.
- My mom and brother, who still have no idea what my project is about, and the rest of my family. Thank you so much for all your love and support throughout all my years of education.
- My supervisor, Dr Dieter Vogts, for his invaluable advice, guidance and belief in me throughout the project.
- All the staff in the NMMU CS Department for all their valuable help and feedback they gave over the last six years.
- My fellow colleagues D, Amanzi, Michael, Blessing, Dany and Cainos. Each one of you has helped me through this project in one way or another. You guys are awesome so why are you panicking? Just go home. See you guys on Saturday 9am sharp!
- The NMMU Telkom Centre of Excellence for their financial assistance, without which this research would not be possible.
- The thirty individuals who willingly took part in the user study.
- Danie Venter from the NMMU Unit for Statistical Consultation for his assistance with the preparation of the statistical results of the evaluation.
- Hans van de Groenendaal from Engineer IT magazine, for his interest in the project and the article he published about it.

Summary

Synchronous Distributed Collaborative Work (SDCW) occurs when group members work together at the same time from different places together to achieve a common goal. Effective SDCW requires good communication, continuous coordination and shared information among group members. SDCW is possible because of groupware, a class of computer software systems that supports group work. Shared-workspace groupware systems are systems that provide a common workspace that aims to replicate aspects of a physical workspace that is shared among group members in a co-located environment. Shared-workspace groupware systems have failed to provide the same degree of coordination and awareness among distributed group members that exists in co-located groups owing to unintuitive interaction techniques that these systems have incorporated.

Natural User Interfaces (NUIs) focus on reusing natural human abilities such as touch, speech, gestures and proximity awareness to allow intuitive human-computer interaction. These interaction techniques could provide solutions to the existing issues of groupware systems by breaking down the barrier between people and technology created by the interaction techniques currently utilised. The aim of this research was to investigate how NUI interaction techniques could be used to effectively support SDCW. An architecture for such a shared-workspace groupware system was proposed and a prototype, called GroupAware, was designed and developed based on this architecture. GroupAware allows multiple users from distributed locations to simultaneously view and annotate text documents, and create graphic designs in a shared workspace. Documents are represented as visual objects that can be manipulated through touch gestures. Group coordination and awareness is maintained through document updates via immediate workspace synchronization, user action tracking via user labels and user availability identification via basic proxemic interaction. Members can effectively communicate via audio and video conferencing.

A user study was conducted to evaluate GroupAware and determine whether NUI interaction techniques effectively supported SDCW. Ten groups of three members each participated in the study. High levels of performance, user satisfaction and collaboration demonstrated that GroupAware was an effective groupware system that was easy to learn and use, and effectively supported group work in terms of communication, coordination and information sharing. Participants gave highly positive comments about the system that further supported the results. The successful implementation of GroupAware and the positive results obtained from the user evaluation provides evidence that NUI interaction techniques can effectively support SDCW.

Keywords: Synchronous Distributed Collaborative Work, Natural User Interfaces, Groupware, Computer Supported Cooperative Work, Human-Computer Interaction

Table of Contents

Declaration i

Acknowledgements ii

Summary iii

Table of Contents iv

List of Figures x

List of Tables xii

Glossary xiv

Chapter 1: Introduction 1

1.1 Background 1

1.2 Problem Identification 3

1.3 Aim of Research 3

1.4 Research Questions 3

1.5 Research Objectives 4

1.6 Scope and Constraints 4

1.7 Research Methodology 5

1.7.1 Design Science Research 5

1.7.2 Literature Review 7

1.7.3 Prototyping 7

1.7.4 Experiment 7

1.8 Dissertation Structure 8

Chapter 2: Collaboration 10

2.1 Introduction 10

2.2 Defining Collaboration 10

2.3 Computer Supported Cooperative Work 12

2.3.1 Communication 13

2.3.2 Coordination 14

2.3.3 Information Sharing 16
2.3.4 Time 17
2.3.4.1 Synchronous 17
2.3.4.2 Asynchronous 18
2.3.5 Space 18
2.3.5.1 Co-located 18
2.3.5.2 Distributed 19
2.3.6 Groupware 21
2.4 Requirements of SDCW in a Shared Workspace 23
2.4.1 Mechanics of Collaboration 24
2.4.2 Functional Requirements 25
2.4.3 Non-functional Requirements 26
2.5 Conclusions 27
Chapter 3: Natural User Interfaces 28
3.1 Introduction 28
3.2 Defining NUIs 28
3.3 NUI Interaction Techniques 29
3.3.1 Touch and Stylus 30
3.3.2 Speech 31
3.3.3 In-air Gestures 32
3.3.4 Proxemics 33
3.3.5 NUI Devices 34
3.3.5.1 Stationary Devices 35
3.3.5.2 Mobile Devices 35
3.4 Related Work 36
3.4.1 Synchronous Co-located Groupware Systems 36
3.4.2 Synchronous Distributed Groupware Systems 39
3.4.3 Discussion 43
3.4.3.1 Synchronous Co-located Groupware Systems 43

3.4.3.2 Synchronous Distributed Groupware Systems 45
3.5 Mapping SDCW Requirements onto NUI interaction 47
3.6 Conclusions 48
Chapter 4: Design and Implementation.. 50
4.1 Introduction 50
4.2 Development Methodology 50
4.3 Application Domain.. 51
4.4 Design.. 51
4.4.1 System Architecture Overview 52
4.4.2 Application Architecture 53
4.4.3 Distribution Architecture 54
4.4.4 Model Design 56
4.4.4.1 Functionality 56
4.4.4.2 Data 59
4.4.4.3 Shared Model.. 60
4.4.5 View and Controller Design 61
4.4.5.1 Workspace 61
4.4.5.2 User Labels.. 62
4.4.5.3 Custom Widget Design.. 63
4.4.5.4 Main Menu Widget.. 64
4.4.5.5 Document Widget.. 65
4.4.5.6 Text Viewer and Drawing Widgets.. 66
4.4.5.7 History Widget 69
4.4.5.8 User Widget 69
4.4.6 Detailed System Architecture 70
4.5 Implementation 72
4.5.1 Implementation Tools.. 72
4.5.1.1 Hardware.. 72
4.5.1.2 Programming Language and Environment 73
4.5.1.3 Surface 2.0 SDK 74
4.5.1.4 Thriple 74

4.5.1.5 Groupware Toolkit for C# ... 74
4.5.1.6 VLCDotNet ... 74
4.5.1.7 Emgu CV ... 75
4.5.2 Prototypes ... 75
4.5.2.1 Workspace ... 75
4.5.2.2 Distribution ... 77
4.5.2.3 Communication ... 78
4.5.2.4 Proxemics ... 79
4.5.2.5 Prototype Integration ... 79
4.5.3 Functionality ... 81
4.6 Conclusions ... 82
Chapter 5: Evaluation ... 83
5.1 Introduction ... 83
5.2 Evaluation Objectives ... 83
5.3 Evaluation Techniques ... 83
5.3.1 Related Work ... 84
5.3.1.1 Usability ... 86
5.3.1.2 Collaboration ... 87
5.3.2 Extant Systems' Review ... 89
5.3.3 Extant Systems' Review Conclusions ... 90
5.4 User Study ... 92
5.4.1 Experimental Design ... 92
5.4.2 Data Collection Methods and Instruments ... 92
5.4.3 Metrics ... 93
5.4.4 Location ... 94
5.4.5 Participants ... 94
5.4.6 Tasks ... 95
5.4.7 Questionnaires ... 95
5.4.8 Statistics ... 96
5.4.9 Experimental Procedure ... 96

5.5 Results 97
5.5.1 Demographics 97
5.5.2 Usability Results 99
5.5.2.1 Performance Results 100
5.5.2.2 User Satisfaction Results 102
5.5.2.3 General Usability Comments 107
5.5.3 Collaboration Results 109
5.5.3.1 Verbal and Practical Contributions Results 109
5.5.3.2 User Perceptions Results 113
5.5.3.3 General Collaboration Comments 116
5.5.3.4 Laughter Results 117
5.6 Discussion 118
5.7 Conclusions 120
Chapter 6: Conclusions 122
6.1 Introduction 122
6.2 Research Achievements 122
6.3 Limitations and Problems Faced 124
6.3.1 Implementation Problems 125
6.3.2 Evaluation Problems 126
6.4 Research Contributions 126
6.4.1 Theoretical Contributions 127
6.4.2 Practical Contributions 128
6.5 Future Research 129
References 131
Appendix A: Research Ethics Approval 139
Appendix B: Scenario and Task List 140
Appendix C: Biographical Questionnaire 142
Appendix D: Post-test Questionnaire 143
Appendix E: User Study Presentation 147

Appendix F: Written Information **148**

Appendix G: Consent Form **150**

List of Figures

Figure 1.1: Design Science Research paradigm (Hevner, 2007) 5
Figure 3.1: Typical touch gestures (Villamor, Willis, & Wroblewski, 2010) 30
Figure 3.2: Proxemic zones (Marquardt & Greenberg, 2012) 33
Figure 3.3: Proxemic dimensions (Marquardt & Greenberg, 2012) 34
Figure 3.4: Stationary NUI devices (Luderschmidt, 2013; Sebit LLC, 2014) 35
Figure 3.5: Mobile NUI devices (CompareHero, 2013; Sony Mobile Communications Inc., 2014) 36
Figure 3.6: The DeskPiles interactive workspace (Milic-Frayling et al., 2010) 36
Figure 3.7: Code Space meeting environment (Bragdon et al., 2011) 37
Figure 3.8: The NiCE Discussion Room (Haller et al., 2010) 38
Figure 3.9: CollaGIM workspace (Ditta et al., 2013) 39
Figure 3.10: A C-Slate workstation (Izadi et al., 2007) 40
Figure 3.11: A Remote Multi-touch Collaborative Environment (Ardaiz et al., 2010) 41
Figure 3.12: Three's Company collaborative environment (Tang et al., 2010) 42
Figure 3.13: Distributed photo-sharing system testing KinectArms toolkit (Genest et al., 2013) 43
Figure 4.1: System architecture overview 53
Figure 4.2: Typical interactions of the MVC components 54
Figure 4.3: Centralized Mixer with Broadcaster distribution architecture (Graham et al., 2006) 55
Figure 4.4: UI design of the workspace upon start-up of GroupAware 61
Figure 4.5: Active Users Panel 62
Figure 4.6: Example of divider tabs on a physical document 62
Figure 4.7: User labels 63
Figure 4.8: General design of a custom GroupAware widget 64
Figure 4.9: UI design of the Main Menu widget 64
Figure 4.10: Menu items of Main Menu widget 65
Figure 4.11: UI design of the Document widget (graphic) 65
Figure 4.12: Document widget design aspects 66
Figure 4.13: UI design of the Text Viewer widget 67
Figure 4.14: UI design of the Drawing widget with a completed sketch 68
Figure 4.15: Real-time drawing synchronization 68
Figure 4.16: UI design of the History widget 69
Figure 4.17: UI design of the User widget 70
Figure 4.18: Detailed system architecture of GroupAware 71

Figure 4.19: The ASUS EEE Slate 72
Figure 4.20: NMMU Telkom/CoE multi-touch tabletop in vertical position 73
Figure 5.1: The eight key evaluation areas of groupware system evaluation (Ramage, 1999) 85
Figure 5.2: The relationships among the four key group evaluation attributes (Araujo et al., 2004) .. 85
Figure 5.3: Participant demographics (n=30) 98
Figure 5.4: Group demographics (n=10) 99
Figure 5.5: Total completion rate for participant per individual task (n=30) 100
Figure 5.6: Total completion rate for groups per group task (n=10) 100
Figure 5.7: Time-on-task for the slowest and fastest groups as well as the average (n=10) 101
Figure 5.8: Cumulative team task times for fastest, slowest and average (n=10) 102
Figure 5.9: User satisfaction results per PSSUQ section (n=30) 103
Figure 5.10: Mean 7-Point Likert scale ratings for the Usability section (n=30) 104
Figure 5.11: Mean 7-Point Likert scale ratings for the Information Quality section (n=30) 105
Figure 5.12: Mean 7-Point Likert scale ratings for the Interface Quality section (n=30) 106
Figure 5.13: Proportion of speaking per group (n=10) 109
Figure 5.14: Average proportion of speaking 110
Figure 5.15: Proportion of practical contributions per group (n=10) 111
Figure 5.16: Average proportion of practical contributions 111
Figure 5.17: Proportion of all contributions per group (n=10) 112
Figure 5.18: Average proportion for all contributions 113
Figure 5.19: User perceptions results per Collaboration section (n=30) 114
Figure 5.20: Mean 7-Point Likert scale ratings for the Communication section (n=30) 114
Figure 5.21: Mean 7-point Likert scale ratings for the Coordination section (n=30) 115
Figure 5.22: Mean 7-Point Likert scale ratings for the Information Sharing section (n=30) 116
Figure 5.23: Total amount of laughter per group (n=10) 118
Figure 6.1: System architecture of GroupAware 127
Figure 6.2: Screenshot of a GroupAware client's workspace 129

List of Tables

Table 1.1: Design Science Research guidelines (Hevner & Chatterjee, 2010) 6
Table 1.2: The structure of the dissertation 8
Table 2.1: Past and present elements of workspace awareness (Gutwin & Greenberg, 2004) 15
Table 2.2: Examples of group tasks (Gutwin & Greenberg, 2002; Lanubile, 2009) 16
Table 2.3: Descriptions and examples of groupware system categories 21
Table 2.4: Johansen's time-space matrix (Johansen, 1988) 22
Table 2.5: Classification of groupware systems (Penichet et al., 2007) 22
Table 2.6: Functional requirements of SDCW in a shared workspace 25
Table 2.7: Non-functional requirements of SDCW in a shared workspace 26
Table 3.1: The devices and interaction techniques used in synchronous co-located groupware systems 44
Table 3.2: The devices and interaction techniques used in shared-workspace groupware systems ... 45
Table 3.3: Proposed mapping of SDCW requirements onto NUI interaction techniques 47
Table 4.1: The satisfying of the functional requirements 58
Table 4.2: Data channels used to transport messages 60
Table 4.3: Informal workspace test results 76
Table 4.4: Testing the audio and video delay (in seconds) for various transcoding options 78
Table 4.5: Results from expert review 80
Table 4.6: The implementation of the functional requirements 81
Table 5.1: Measures for evaluating the collaborative level of groupware systems (Araujo et al., 2004) 88
Table 5.2: Measures for evaluating groupware systems and how they can be captured (Mattsson, 2011) 88
Table 5.3: A summary of the extant groupware systems' evaluation techniques 90
Table 5.4: A summary of the data collection methods and instruments 93
Table 5.5: A summary of the metrics and how they were used 94
Table 5.6: Interpretation intervals for Cronbach's alphas 102
Table 5.7: Cronbach's alpha coefficients and interpretation for each PSSUQ section (n=30) 103
Table 5.8: Positive Usability themes and frequencies identified in user comments (n=30) 107
Table 5.9: Negative Usability themes and frequencies identified in user comments (n=30) 108
Table 5.10: Cronbach's alpha coefficients and interpretation for each Collaboration section (n=30) 113

Table 5.11: Positive Collaboration themes and frequencies identified in user comments (n=30) 116
Table 5.12: Negative Collaboration themes and frequencies identified in user comments (n=30)... 117

Glossary

3D	Three Dimensional
ACM	Association for Computing Machinery
API	Application Programing Interface
AUP	Agile Unified Process
CCCM	Concurrency Control and Consistency Maintenance
CIF	Common Industry Format
CLI	Command Line Interface
CMC	Computer-Mediated Communication
CoE	Centre of Excellence
CS	Computing Sciences
CSCL	Computer Supported Collaborative Learning
CSCW	Computer Supported Cooperative Work
CV	Computer Vision
DMS	Document Management System
DSR	Design Science Research
FTIR	Frustrated Total Internal Reflection
GDSS	Group Decision Support System
GIM	Group Information Management
GT	Groupware Toolkit
GUI	Graphical User Interface
HCI	Human-Computer Interaction
HMVC	Hierarchical Model-View-Controller
IDE	Integrated Development Environment
IM	Instant Messaging
IT	Information Technology
LCD	Liquid Crystal Display
MMO	Massively Multiplayer Online Game
MVA	Model-View-Adapter
MVC	Model-View-Controller
MVP	Model-View-Presenter
MVVM	Model-View-ViewModel
NMMU	Nelson Mandela Metropolitan University

NUI	Natural User Interface
PDF	Portable Document Format
PSSUQ	Post Study System Usability Questionnaire
RMCE	Remote Multi-touch Collaborative Environment
RTCE	Real-Time Collaborative Editor
SDCW	Synchronous Distributed Collaborative Work
SDK	Software Development Kit
TCP	Transmission Control Protocol
UI	User Interface
VC	Videoconferencing
VLC	VideoLAN Client
VoIP	Voice over Internet Protocol
WIMP	Windows Icons Menus Pointers
WMS	Workflow Management System
WPF	Windows Presentation Foundation
XAML	Extensible Application Markup Language

Chapter 1: Introduction

1.1 Background

Collaborative work, commonly referred to as collaboration, occurs when members of a group work together to achieve a common goal. Collaboration can occur synchronously or asynchronously, that is, at the same time or at different times. Effective synchronous collaboration relies on good communication, continuous coordination and shared information among group members. Furthermore, collaboration can occur in co-located or distributed environments, that is, in the same physical location or from different physical locations. Synchronous collaboration occurs most naturally among groups in a co-located environment. Communication, coordination and information sharing is effortless when group members are in the same physical location and is often taken for granted (Lanubile, 2009; Tee, Greenberg, & Gutwin, 2009).

Synchronous Distributed Collaborative Work (SDCW) can be defined as group members working together at the same time from different places. SDCW is possible because of groupware, a class of computer software systems that utilises computer networks to support group work (Penichet, Marin, Gallud, Lozano, & Tesoriero, 2007). Shared-workspace groupware systems provide a common workspace in which distributed members can create, see and manipulate work artefacts (Gutwin & Greenberg, 2004). The shared workspace aims to replicate aspects of the physical workspace that is shared among group members in a co-located environment (Tang et al., 2010).

The physical distance between group members creates challenges for SDCW, particularly in the area of coordination. Coordination involves the awareness of others and their tasks. Successful group coordination reduces effort, errors and duplication and without it collaboration is awkward, inefficient and clumsy (Gutwin & Greenberg, 2004). Coordination is difficult in distributed groups because group members are not naturally aware of who is around and what others are doing in the shared workspace. There are only a few examples where shared-workspace groupware systems have succeeded in the real world. This is not because there is no market for these systems, since similar synchronous groupware systems such as Instant Messaging (IM) and Massively Multiplayer Online games (MMOs) have proven to be extremely popular among the general public (Gutwin et al., 2008).

There are a number of possible reasons for the low adoption rate of shared-workspace groupware systems (Gutwin et al., 2008). One that is explored in this research is that the systems fail to capture important coordination information and present that information to the group. The limited awareness information found in groupware systems can possibly be linked to the interaction techniques that are implemented by the system. Interaction techniques are ways in which a user can interact with a computer, consisting of all hardware and software elements. Interacting with technology generates only a fraction of the awareness information that is available in a co-located workspace. The majority of existing groupware systems, thus far, have been designed and developed within the Windows, Icons, Menus and Pointers (WIMP) interaction paradigm. The WIMP interaction paradigm comprises limited and unnatural interaction techniques. Distributed group members are, therefore, not able to naturally and seamlessly interact with existing groupware systems. Groupware systems that require too much of the user's attention or are difficult to use, hinder collaboration. Thus, technologies and interaction techniques that have not previously been used in shared-workspace groupware systems are needed to solve this problem.

Natural User Interfaces (NUIs) are the next generation of user interfaces in the progressing field of Human-Computer Interaction (HCI). NUIs focus on reusing natural human skills and abilities such as touch, speech, gestures and proximity awareness to allow intuitive user interaction with technology. These interaction techniques could provide solutions to the problems of existing shared-workspace groupware systems by breaking down the barrier between people and technology created by current interaction techniques (Microsoft News Center, 2010). The technology would then be perceived as unobtrusive because it does not require the user's continuous attention or a large amount of cognitive resources. This gives users to ability to appropriately focus their attention on working together as a group.

Touch has recently become a popular interaction technique that has been adopted by many computing devices, especially mobile devices such as smartphones and tablets. Over the last decade there has been a greater adoption of the use of mobile devices rather than the typical desktop computers, which implement WIMP interaction, due to the advantages that mobility provides (Chittaro, 2006). Previous disadvantages of mobile devices such as performance, connectivity and storage are becoming less of an issue due to improvements in technology. Mobile devices, such as tablets, provide an intuitive means to interact with the device through the use of touch interaction.

NUI interaction techniques have been incorporated in the latest synchronous co-located groupware such as Code Space, CollaGIM and the NiCE Discussion Room, yielding positive results, indicating that NUIs can support group work (Bragdon et al., 2011; Ditta et al., 2013; Haller et al., 2010). This research aims to determine whether NUI interaction techniques can be applied to effectively support SDCW. In order to achieve this, a shared-workspace groupware system that incorporates NUI interaction techniques must be evaluated in terms of the usability and collaboration.

1.2 Problem Identification

The following problem statement identifies the problem addressed by this research:

Existing shared-workspace groupware systems do not provide effective support for SDCW.

The problem statement indicates that SDCW lacks effective support from existing shared-workspace groupware systems and highlights a need for further research in this area. This research aims to investigate the identified problem and propose a suitable solution.

1.3 Aim of Research

The goal that the research aims to achieve is the following:

To investigate and evaluate the use of NUI interaction techniques to support SDCW.

Incorporating NUI interaction techniques into shared-workspace groupware systems is proposed as a solution to effectively support SDCW. Several research questions and objectives are identified to guide the research in successfully accomplishing the goal.

1.4 Research Questions

The primary research question to be answered is the following:

How can the use of NUI interaction techniques effectively support SDCW?

The following secondary research questions are identified in order to answer the primary question:

- RQ1. What is SDCW and what are the requirements thereof?
- RQ2. What are the limitations of existing groupware systems?
- RQ3. What are the existing techniques of NUI interaction?
- RQ4. How can NUI interaction be incorporated into shared-workspace groupware systems to address the existing limitations and problems?
- RQ5. How can a shared-workspace groupware prototype implementing NUI interaction be designed and developed to support SDCW?
- RQ6. How effectively does the developed prototype support SDCW?
- RQ7. What are the research contributions and what future research should be carried out to improve NUI interaction techniques for SDCW?

The research presented in this dissertation will be conducted in a way that answers these research questions. Each research question has a corresponding research objective.

1.5 Research Objectives

The primary objective of the research is to investigate and evaluate the use of NUI interaction techniques to support SDCW. The following secondary research objectives are identified in order to successfully achieve the primary research objective:

RO1. To define and discuss the concept of SDCW and investigate the requirements thereof.

RO2. To identify the limitations of existing groupware systems.

RO3. To define and discuss existing NUI interaction techniques.

RO4. To determine how NUI interaction techniques can be incorporated into shared-workspace groupware systems to address the existing limitations.

RO5. To design and develop a shared-workspace groupware prototype that implements NUI interaction techniques to support SDCW.

RO6. To determine how effectively NUIs support SDCW through an evaluation of the prototype.

RO7. To identify the research contributions and make recommendations for future research in order to improve the support for SDCW using NUI interaction techniques.

The research presented in this dissertation will be conducted in a way that achieves these research objectives. The research will be subject to the following constraints.

1.6 Scope and Constraints

The scope of this research will be limited to collaborative work in a synchronous, distributed environment. Consequently, the focus will be on groupware systems that support real-time collaboration for a small group of co-workers who are separated by distance. This will be further concentrated to groupware systems that provide a shared visual workspace to support collaborative tasks that involve generating ideas and constructing digital artefacts. Emphasis will be placed on group coordination and awareness that occurs when performing these tasks using a groupware system.

Existing NUI interaction techniques will be adapted to support SDCW and new techniques will not be developed. The NUI techniques that will be considered are multi-touch and proxemic interaction. Touch interaction will be the primary input by means of a multi-touch enabled device. Basic proxemic interaction will be the secondary input by means of a webcam. A microphone will capture the speech of a user, but no speech recognition will take place; the voice input will simply be propagated to other group members. No other sources of input technology such as a physical keyboard and mouse will be used.

1.7 Research Methodology

A research methodology is a system of principles, practices and procedures applied to a specific branch of knowledge to contribute to the understanding of a phenomenon (Peffers, Tuunanen, Rothenberger, & Chatterjee, 2007). The Design Science Research (DSR) methodology will be applied in this research. The characteristics of DSR are discussed in the next section (Section 1.7.1). Within the DSR methodology, three research techniques will be used to address the research questions presented in Section 1.4, namely Literature Review (Section 1.7.2), Prototyping (Section 1.7.3) and Experiment (Section 1.7.4).

1.7.1 Design Science Research

The DSR paradigm is a research methodology that seeks to expand the capabilities of humans and organizations by creating new and innovative artefacts. It creates and evaluates Information Technology (IT) artefacts intended to solve identified problems. Such artefacts can be represented as formal logic, rigorous mathematics or software systems (Hevner et al., 2004). The DSR methodology is used in this research because it seeks to understand a phenomenon (using NUIs to support SDCW) through the design of an artefact (groupware system).

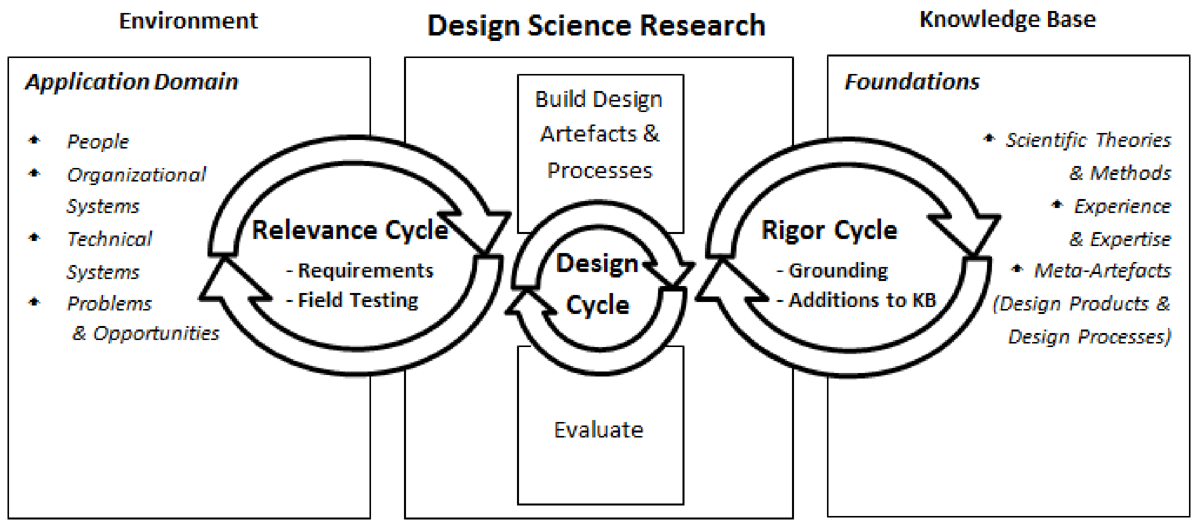


Figure 1.1: Design Science Research paradigm (Hevner, 2007)

Design Science Research comprises an Environment, a Knowledge Base, and three cycles, namely the Relevance, Rigor and Design cycles (see Figure 1.1). The Environment is the context in which the problem or opportunity resides i.e. the application domain. It also includes the people affected by the problem or opportunity, and the corresponding organizational and technical systems. The Knowledge Base provides the foundation and background of the problem via existing theories and methods along with domain experience and expertise.

The Relevance cycle inputs requirements from the Environment into the research and later introduces the research artefacts into environmental field testing. The Rigor cycle provides grounding from the Knowledge Base for the research and later adds the new knowledge generated by the research to the growing Knowledge Base. The Design cycle enables the iterative construction and evaluation of design artefacts and processes (Hevner, 2007).

When conducting research using the DSR methodology, a set of seven guidelines should be followed (Hevner & Chatterjee, 2010). These guidelines are presented in Table 1.1.

Table 1.1: Design Science Research guidelines (Hevner & Chatterjee, 2010)

Guideline	Description
Guideline 1: Design as an Artefact	DSR must provide a viable artefact in the form of a construct, a model, method or 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, foundations or 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 to both technology-oriented and management-oriented audiences.

Furthermore, research incorporating the DSR methodology must undergo six phases, namely (Peffers et al., 2007):

1. Identify the Problem: This is where the problem is identified and motivated.
2. Define Requirements: In this phase the problem is investigated further and the requirements of an artefact are discussed.
3. Design and Develop Artefact: This is when possible solutions to the problem are designed. Both design and development of the prototype occur in this phase.

4. Demonstrate Artefact: This phase involves the illustration of the artefact's effectiveness in solving the identified problem.
5. Evaluate Artefact: This phase indicates how well the artefact solves the problem.
6. Communicate Findings: The phase provides the communication of the problem, artefact and other findings resulting from the research.

These DSR phases will be covered using three existing research methods, namely Literature Review, Prototyping and Experiment. These methods will be discussed in the following three sections.

1.7.2 Literature Review

Literature reviews are used to offer a summary of a topic in a specific research area (Hofstee, 2009). The literature review will provide grounding and identify requirements for the research by drawing from the vast knowledge base found in the Computer Supported Cooperative Work (CSCW) field of research. The review will be used to investigate existing characteristics regarding collaboration with specific attention given to SDCW and the requirements thereof. Existing groupware systems will be identified and their shortcomings will be discussed. NUI interaction techniques that may be suitable to address the problems of shared-workspace groupware systems will be identified. The literature review will be used as a basis for the design and implementation of a groupware prototype.

1.7.3 Prototyping

A prototype can be used to express research in a useful way (Olivier, 2009). A prototyping, or proof-of-concept, method will be used to build a design artefact. The prototype will incorporate NUI interaction techniques in an attempt to alleviate the problems identified with existing shared-workspace groupware systems. Incremental prototyping will be used to separate the prototype into components, each of which will be iteratively designed, developed and evaluated. The prototype will be used to determine whether NUIs can effectively support SDCW by means of an experiment.

1.7.4 Experiment

An experiment can be conducted to assess a theory or to examine the outcome of an intervention (Hofstee, 2009). An experiment will be conducted to evaluate the effectiveness of the design artefact. The experiment will involve a usability evaluation based on existing SDCW evaluation methods. Performance and satisfaction metrics will be used to evaluate the prototype. Additionally, measures will be identified and used to determine the degree to which collaboration is supported by the prototype. Quantitative and qualitative data will be captured and analysed. The results of the evaluation will be presented, from which the conclusions of the research will be drawn.

1.8 Dissertation Structure

This dissertation will be structured according to the selected research methodology identified in the previous section. The dissertation will consist of six chapters, in which the research questions will be answered. Table 1.2 presents the structure of the dissertation and identifies the DSR phase involved, research questions answered and research method used in each chapter.

Table 1.2: The structure of the dissertation

Chapter	DSR Phase	Research Questions Answered	Research Method
Chapter 1: Introduction	Phase 1		
Chapter 2: Collaboration Literature Review	Phase 2	RQ1. What is SDCW and what are the requirements thereof? RQ2. What are the limitations of existing groupware systems?	Literature Review
Chapter 3: NUI Literature Review	Phase 2	RQ3. What are the existing techniques of NUI interaction? RQ4. How can NUI interaction be incorporated into shared-workspace groupware systems to address the existing limitations and problems?	Literature Review
Chapter 4: Design and Implementation	Phase 3 and Phase 4	RQ5. How can NUI interaction be incorporated into shared-workspace groupware systems to address the existing limitations and problems?	Prototyping
Chapter 5: Evaluation	Phase 5	RQ6. How effectively does the developed prototype support SDCW?	Experiment
Chapter 6: Conclusions	Phase 6	RQ7. What are the research contributions and what future research should be carried out to improve NUI interaction techniques for SDCW?	Critical Reflection

Design Science Research

Chapter 1 served as an introductory chapter to the research and covered Phase 1 of DSR. This chapter presented a brief background of the research topic to provide the context and significance of the research. The problem statement was highlighted and used to determine the aim of the research and identify the research questions and objectives. The scope and constraints of the research were identified. The research methodology to be applied throughout the research was discussed in this chapter.

The literature review, which involves Phase 2 of DSR, will be covered in Chapters 2 and 3. In Chapter 2, the plethora of literature found on CSCW will be explored to describe the concept of collaboration. The characteristics and requirements of collaboration will be discussed with specific attention given to synchronous distributed collaborative environments and situations. The advantages and disadvantages of existing groupware systems will then be discussed and analysed to determine how effectively these systems support SDCW.

NUIs will be discussed in Chapter 3. Various NUI interaction techniques will be introduced and discussed. The advantages and disadvantages of these interaction techniques will be identified and their possible application in shared-workspace groupware systems will be discussed. Furthermore, current and related literature will be examined to understand how NUI interaction can be used to support SDCW.

Chapter 4 will discuss the design and implementation of a shared-workspace groupware prototype that will incorporate NUI interaction techniques. This chapter will cover Phase 3 and phase 4 of DSR. The design of the prototype will be based on the requirements of SDCW identified in the previous chapters. An architecture incorporating NUI interaction techniques to support SDCW will be proposed, followed by the design of the prototype in terms of its functionality, data, User Interface (UI), and interaction techniques. The implementation of the prototype will be based on the proposed design. The chapter will then describe the details of the implementation in terms of the tools that were used, the issues that were faced and the functionality that was developed.

The experiment, which involves Phase 5 of DSR, will be discussed in Chapter 5. Firstly, the objectives of the evaluation will be clearly defined. A brief review of existing groupware evaluation techniques will then be conducted and the appropriate techniques selected. The evaluation design will be described in terms of the data collection methods and instruments used, the participant sample collected and the test procedure followed. The results of the experiment will then be presented and analysed.

Chapter 6 will conclude the dissertation. In this final chapter, the project will be critically examined to determine whether the research objectives were achieved. The final phase of DSR will be covered in this chapter. The limitations of the research and problems faced during development will be documented. The theoretical and practical research contributions will be presented. Finally, recommendations for future work in this area of research will be made.

Chapter 2: Collaboration

2.1 Introduction

This chapter presents the first of two Literature Reviews and delves deeper into the characteristics and components of collaboration, specifically focussing on SDCW. This chapter involves the second phase of the DSR methodology, namely Define Requirements. It answers the first two research questions identified in Chapter 1, namely:

RQ1. What is SDCW and what are the requirements thereof?

RQ2. What are the limitations of existing shared-workspace groupware systems?

The above questions will be answered by means of a Literature Review, in which research that is relevant and current will be reviewed, and the key points extracted and classified. Firstly, collaboration and collaborative software will be defined and described in detail by reviewing literature from CSCW's rich body of knowledge (Sections 2.2 and 2.3). The requirements of a SDCW software system will be presented (Section 2.4) followed by the key findings of the Literature Review, which will conclude the chapter (Section 2.5).

2.2 Defining Collaboration

In Chapter 1, collaboration was briefly defined as members of a group working together to achieve a shared goal. This definition only scratches the surface of the concept of collaboration, which is a topic that has been studied for a long time. Thus, this section will review past and present literature in order to investigate the meaning of collaboration.

In a study of the construction of shared knowledge in collaborative problem solving, collaboration was defined in the following way (Roschelle & Teasley, 1995, p. 70):

> *"the mutual engagement of participants in a coordinated effort to solve a problem together".*

This definition highlights and emphasizes essential qualities of collaboration such as the shared interaction and communication of people (mutual engagement of participants), the synergy and synchronization among the people (coordinated effort), and the performance of shared work (solve a problem together).

Collaboration involves two types of work (Gutwin & Greenberg, 2000). Firstly, the actual execution of the task needs to take place. This work is essentially no different for a group than it is for an individual, since the same actions still have to transpire in order for the task to be accomplished. The advantage of collaboration, however, is that the work is shared among group members. Shared perspectives, knowledge, skills and resources help to accomplish the work more efficiently and improve the quality of work (Belcher, 2011). Secondly, the effort of working together is work in and of itself. This work involves actions and interactions that groups have to perform in order to solve a shared problem. These actions and interactions include establishing a common understanding of the problem, making decisions about solutions to the problem and maintaining coordination among group members throughout the solving of the problem.

A term that is often used synonymously with collaboration is cooperation. Although these terms are similar, a distinction between them can be made. Cooperation occurs when there is a division of work, in which each cooperative member is responsible for some portion of the task. Collaboration, on the other hand, involves members working together on the same task, rather than on separate portions of the task (Dillenbourg, Baker, Blaye, & O'Malley, 1996).

The difference between collaboration and cooperation, however, is not clear-cut because collaboration may involve cooperation by means of spontaneous task delegation and division of work among members in order to accomplish the shared goal. This has resulted in the classification of three different collaborative styles, namely (Morris & Winograd, 2004):

- Serial: Group members all work together on the same aspect of a task.
- Parallel: Group members work on different sub-tasks.
- Assembly-line: Group members work on different stages of a task.

The first collaborative style involves no division of work, whereas the others do, but none of these styles is more or less collaborative than the others. Some styles, however, may be more effective than others for a particular task (Morris & Winograd, 2004).

Collaboration is present in numerous areas such as trade, communities, military, business, education, music, entertainment, medicine and science. Technology has had a significant impact on the support of collaboration in these areas (ACM, 2014). This can be seen from the computer-related research fields that exist, namely Computer-Mediated Communication (CMC), Computer Supported Collaborative Learning (CSCL) and CSCW. CSCW, in particular, is a rich body of knowledge comprising literature related to understanding group work and using technology to facilitate that work. CSCW will be discussed in the next section.

2.3 Computer Supported Cooperative Work

In the mid-1980s, a multidisciplinary workshop was organized by Paul Cashman and Irene Greif. Many people from different fields came together to discuss how computers could help people work together. The workshop resulted in the field of research known as Computer Supported Cooperative Work (CSCW), which has been active for over three decades (Penichet et al., 2007).

CSCW is the study of groups of people using computer technologies to support and facilitate their work. CSCW investigates the ways and contexts in which people collaborate and how computer technology can effectively support that collaboration. Furthermore, CSCW studies the impact that computers have on collaborative work and provides guidelines for the design and implementation of computer software for different collaborative environments.

In a researcher's report of his experiences at the 2nd Association for Computing Machinery (ACM) Conference on CSCW, a definition of CSCW was given as "a new research field focused on the role of the computer in group work" (Greenberg, 1989, p. 49). The researcher stated that CSCW was technology-driven, motivated by technical studies of computer hardware and software, and socially-driven, motivated by studies of group interaction. The social and technological drivers of CSCW make it an interdisciplinary research field because it can be applied in a number of fields. This can be verified by observing the various sessions held at the 2014 ACM Conference on CSCW, which included (ACM, 2014):

> *The Office; Building Communities and Relationships; Family; Work in Hospitals; Personal Health Management; Craft, Repair and DIY; Social Media; Social-Tech and Well-Being; Shopping and Collecting; Romance; Mobile Apps; Collaborative Search and Sharing; Collaborative Software Development; Leadership; Volunteering and Doing Good; Gaming.*

A more recent definition of CSCW was stated as "CSCW is a research field where the role played by individuals as members of groups is fundamental. The human being is … considered as a being embedded into the society, where he works and interacts" (Penichet et al., 2007, p. 237). The term groupware went hand-in-hand with the definition of CSCW and was defined as "software that supports group processes" (Penichet et al., 2007, p. 237). These definitions correspond to the earlier definition of CSCW in that the social driver is groups of people and the technological driver is groupware.

Penichet et al. (2007) went on to present a new classification for groupware based on five characteristics of CSCW, namely Communication, Coordination, Information Sharing, Time and Space (see Section 2.3.6). In terms of the social aspect, these characteristics describe how, when and where groups can collaborate. In terms of the technological aspect of CSCW, the characteristics describe the type of collaboration and environment that groupware systems can support. The five characteristics will be elaborated on in the following five sections (Sections 2.3.1 - 2.3.5). The discussion on CSCW will conclude with a classification of existing groupware systems (Section 2.3.6).

2.3.1 Communication

Communication is the process of exchanging information with one another, typically via a common set of symbols (Penichet et al., 2007). Communication can be formal or informal, occurring in planned or impromptu interaction (Lanubile, 2009). When communicating, people convey meaning through the actual words that are spoken as well as cues found within their speech and body language. These cues are critical to the correct interpretation of the information intending to be exchanged. There are two categories of cues, namely verbal and non-verbal cues. Below is a list of examples of these cues (Greenberg, 2002):

Examples of verbal cues:

- Volume: How loudly or softly the person is speaking.
- Clarity: How clearly the person is speaking.
- Tone/Inflection: How the pitch of the words changes whilst the person is speaking.
- Pauses: How many pauses occur and when they occur whilst the person is speaking.

Examples of non-verbal cues:

- Gesture: Movement of the head, body or limbs of the speaker.
- Posture: Body language of the speaker.
- Gaze: Where the speaker is looking.
- Eye Contact: Whether the speaker is looking into the eyes of those to whom he is speaking.
- Facial Expression: The shape of eyes, mouth, etc. of the speaker.
- Proxemics: The space between the speaker and those to whom he is speaking.

Tee, Greenberg and Gutwin discussed casual interaction as a part of communication in collaboration (Tee et al., 2009). They defined casual interaction as “the brief, unplanned meetings that commonly occur during the day between co-located people with shared interests” and said that it is a catalyst for collaborative work (Tee et al., 2009, p. 677). Casual interaction involves group members entering into casual conversation and exchanging knowledge and information, which ultimately leads to work being accomplished.

Communication is an essential component of collaborative work because it allows each member to express their perspectives and ideas of the problem, thereby creating a common ground of understanding among collaborators. This common understanding builds trust and enables the group to effectively and efficiently accomplish the task at hand (Tang et al., 2010). Discussing, negotiating and decision-making are some practical examples of how group members communicate during collaborative work. The next characteristic to be discussed is Coordination.

2.3.2 Coordination

Coordination is the process of synchronizing members, tasks and efforts in order to achieve a common goal (Lanubile, 2009; Penichet et al., 2007). Coordination is an internal, cognitive process that occurs mainly in the minds of each group member and is constantly updated as collaboration continues. The existence and maintenance of group coordination is essential to collaborative work (Gutwin & Greenberg, 2004).

Awareness is an integral part of coordination in collaborative work. Awareness can be formally defined as the knowledge that is created through interaction between an agent and its environment. In essence, awareness is to know what is going on in a particular environment. Environments change over time, so awareness must be maintained. People maintain their awareness by observing and interacting with the environment. Typically the overall goal is not just to maintain awareness, but to complete a task in the environment (Gutwin & Greenberg, 2004).

Gutwin and Greenberg coined the term workspace awareness for a specific type of awareness. They defined workspace awareness as “the up-to-the-moment understanding of another person’s interaction with the shared workspace” (Gutwin & Greenberg, 2004, p. 182). The concept can be further understood through the following characteristics:

- Workspace awareness is an understanding of people in the workspace rather than just of the workspace itself.
- It is limited to events happening inside the workspace.
- The physical nature of the workspace itself influences group coordination.

There are many elements involved in workspace awareness. Each element has been categorized and coupled with a specific question to enhance its meaning (see Table 2.1). Tee, Greenberg and Gutwin did further research on workspace awareness, in which they divided it into two types of awareness, namely (Tee et al., 2009):

- Interpersonal Awareness: The understanding of who is around, what they are doing, and whether or not they are available for conversation and collaboration.

- Artefact Awareness: A group member's up-to-the-moment knowledge of the artefacts and tools that other members are using as they do their work.

Table 2.1: Past and present elements of workspace awareness (Gutwin & Greenberg, 2004)

	Present		Past	
Category	**Element**	**Question**	**Element**	**Question**
Who?	Presence Identity Authorship	Is anyone in the workspace? Who is that? Who is doing that?	Presence History	Who was here, and when?
What?	Action Intention Artefact	What are they doing? What is that action part of? What are they working on?	Action History	What has a member been doing?
Where?	Location Gaze View Reach	Where are they working? Where are they looking? How much can they see? How far can they reach?	Location History	Where has a member been?
How?	-	-	Action History Artefact History	How did that action occur? How did this artefact get to this state?
When?	-	-	Event History	When did that event occur?

Interpersonal awareness focuses on the people in the workspace and artefact awareness focuses on the artefacts with which they are working. These types of awareness are essential for the coordination of group members' work and they help people initiate casual interaction, inform knowledge exchange, and build social relationships (Tee et al., 2009).

Coordination is an essential component of collaborative work because group members need to be aware of each other and their tasks in order to effectively and efficiently accomplish the shared goal. It must be maintained in order for group members to remain synchronized with one another and be aware of the changes that occur during collaboration. This is achieved through certain coordinative activities, most of which occur naturally and without much conscious effort.

Coordination includes activities such as using environmental cues to establish a common ground of understanding within the workspace, determining the availability of group members by identifying who is around and what they are doing, monitoring the state of artefacts in the shared workspace, observing other people's gestures and what they are referring to, and organizing the workspace for efficient updates on overall progress. The next characteristic to be discussed is Information Sharing.

2.3.3 Information Sharing

Information sharing occurs when information such as books, documents and diagrams are used by a group to achieve work. Computers support information sharing by enabling data, documents and other digital media to be sent from one group member to another (Penichet et al., 2007). This can be achieved by the sending of digital media from one group member's personal workspace to one or many other group members' personal workspace. Another way information sharing can occur is through a shared visual workspace, which is a common digital environment for group members to access information as well as generate and manipulate work artefacts (Gutwin & Greenberg, 2004).

Information sharing is an essential component of collaborative work because it allows all group members to have the same information to work with and discuss. Furthermore, a shared workspace provides easy access to shared information, simplifies communication about work artefacts, coordinates activity through visual means, helps maintain awareness of others, and allows group members to carry out joint work on tasks (Gutwin et al., 2008).

Four classes of tasks that can occur in groups exist, namely (McGrath, 1984):

- Generation: The generation of creative ideas and plans of action.
- Choice: The solving of problems and decision-making.
- Negotiation: The resolving of conflicting viewpoints and interests.
- Execution: The performance and accomplishment of actual work.

The Choice and Negotiation classes are more communicative in nature and are not discussed further. The Generation and Execution classes, however, involve more appropriate types of information sharing tasks. In particular, these tasks involve the design, construction, organization and exploration of artefacts in a shared workspace. Examples of these tasks are presented in Table 2.2 (Gutwin & Greenberg, 2002; Lanubile, 2009).

Table 2.2: Examples of group tasks (Gutwin & Greenberg, 2002; Lanubile, 2009)

Class	Task	Examples
Generation	Design	Brainstorming, Drawing
	Organization	Arranging, Ordering, Sorting
Execution	Construction	Create, Read, Update, Delete, Integrate
	Exploration	Searching, Filtering

It is important for information sharing activities to be simple and intuitive so that the information can be shared quickly and seamlessly between group members. If information sharing is slow and complex, it will frustrate the members, which could have a negative effect on collaborative work. The next characteristic to be discussed is Time.

2.3.4 Time

The time characteristic indicates the temporal situation of collaboration. There are three types of temporal situations, namely synchronous, asynchronous or both synchronous and asynchronous. The first two situations will be discussed in more detail. The third situation is simply when both of the first two can occur.

2.3.4.1 Synchronous

Synchronous is a term used when multiple events, actions or tasks are occurring simultaneously, i.e. at the same time. Collaborative work that is done synchronously is referred to as real-time or synchronous collaboration. There are several advantages of synchronous collaboration. Some of these advantages include (University of Wisconsin–Madison, 2007):

- Immediacy: Information can be sent and received right away, which presents a more natural, efficient way of communicating and working. The sense of immediacy is more likely to solicit a timely response from people.
- Interactivity: There is a high volume of interactions among group members during synchronous collaboration, which can be very effective in a well-constructed team.

Some disadvantages of synchronous collaboration include the following (University of Wisconsin–Madison, 2007):

- Flexibility: Synchronous collaboration is not the most flexible form of collaboration because all the parties involved must be ready and willing to collaborate at any given moment. If this is not the case, then the session will not go as well as it could.
- Personalities: Not everyone does well with this kind of collaboration, particularly people who like to think over what they want to communicate. People with this personality may struggle with the pressure of synchronous collaboration and not give their best.

From the above it can be said that synchronous collaboration involves highly interactive group meetings where all members are working on a common goal simultaneously. Group members must always be ready to act and react, which could increase productivity, but may be demanding for group members with specific personalities. The converse of synchronous is asynchronous, which is discussed next.

2.3.4.2 Asynchronous

Asynchronous is a term used when multiple events, actions or tasks occur at different times. Collaborative work that is done asynchronously is referred to as asynchronous collaboration. There are some key advantages to asynchronous collaboration, namely (University of Wisconsin–Madison, 2007):

- Flexibility: Collaborators can receive the information when it is most convenient for them. There is less pressure to act on the information or respond immediately. They have time to digest the information and put it in the proper context and perspective.
- Ubiquitous: Many asynchronous collaboration tools, such as email, are pervasive. Most working people have an email account and thus asynchronous collaboration is always a feasible option.

The disadvantages of asynchronous collaboration are (University of Wisconsin–Madison, 2007):

- Immediacy: The lack of pressure can make the recipients slow to respond and sometimes co-workers wait hours, days, and even weeks to get a response or feedback. This means that information can be out of date by the time someone views it.
- Interactivity: There is less interaction among group members, which can cause less communication and trust among members, and ultimately lower productivity.

From the above it can be said that asynchronous collaboration is a ubiquitous form of work in which group members perform tasks at different times. There are no direct interactions among members increasing flexibility, but potentially reducing productivity. The next and final characteristic to be discussed is Space.

2.3.5 Space

The space characteristic indicates the geographical situation of collaboration, of which there are two types, namely co-located or distributed. Co-located and distributed collaboration will be discussed in more detail, specifically identifying the advantages and disadvantages of each with respect to the first three characteristics discussed in the previous sections.

2.3.5.1 Co-located

Co-located collaboration occurs when all the collaborators are in the same geographical location, e.g. a face-to-face meeting. The advantages of co-located collaboration, in terms of the three main characteristics, are the following:

1. Communication: The most natural form of communication among people is that of face-to-face communication. Communicative cues are the verbal and non-verbal hints that help convey meaning. Correctly interpreting these cues is most likely to occur in co-located settings. This allows for effective communication among groups members, which greatly increases the chance of effective collaboration (Greenberg, 2002).
2. Coordination: Interpersonal, artefact, and workspace awareness are a natural part of co-located work environments. Therefore, group coordination is intuitive and easy to maintain when people work in face-to-face settings. This promotes effective collaborative work (Gutwin & Greenberg, 2004; Tee et al., 2009).
3. Information Sharing: In co-located collaborative environments, shared workspaces may include paper media such as Post-its, paper documents, whiteboards and flipcharts as well as digital devices such as projectors, laptops, desktop computers, smart phones, tablets and digital tabletops. These tools, information sources and devices, as well as the artefacts created from them, can be easily shared and integrated with other group members and their work (Haller et al., 2010).

The disadvantages of co-located collaboration include the following (Rosenthal & Finger, 2006):

- In order to work, everyone must be in the same place.
- Scheduling and attending meetings can require a lot of time, money and energy.
- If the meeting place is not dedicated, work must be unpacked and packed up before and after every meeting.

Co-located collaboration is an effective form of collaboration. In many cases, however, co-located group meetings are not practical or possible because of globalization. Organizations that outsource, have offshore employees or clients, or are distributed across multiple cities, countries or continents, require another way to collaborate.

2.3.5.2 Distributed

Distributed collaboration occurs when all the members of a group are in geographically distinct locations. This form of collaboration was established because of technological advancements such as the Internet and electronic media transfer (Krauß et al., 2009). The advantages of distributed collaboration include the following:

- Groups are able to work together over long distances and are not restricted to meetings in the same place.

- There are more opportunities for experts to join project groups where their knowledge can be best used. Teams can be formed based on subject and expertise, without the constraint of physical proximity of group members.
- The time and costs associated with transportation to physically bring together group members from different geographic locations can be substantially higher than the cost of a distributed collaboration system.

Distributed collaboration does, however, come with some significant disadvantages, owing to the distance between group members, which impacts the first three characteristics of collaboration:

1. Communication: There is an increased chance of miscommunication among distributed group members because there is less rich, direct communication. Frequent misunderstandings could result in frustration and even failure (Lanubile, 2009). Distributed collaboration also offers fewer opportunities for casual interaction because of the separation. This has a negative impact on productivity because casual interaction leads to collaborative work (Greenberg, 2001; Tee et al., 2009).
2. Coordination: There is a lack of awareness information among distributed group members, which is problematic because without good awareness, the ease and naturalness of collaboration is lost. This makes distributed collaboration awkward, inefficient and clumsy compared to co-located work (Gutwin & Greenberg, 2004). Also, the lack of awareness forces group members to put a large amount of effort into coordinating their interaction. This effort is a problem because when collaboration is too rigid and planned, many spontaneous interactions don’t occur. Thus, distributed groups are potentially missing out on valuable opportunities for collaboration (Tee et al., 2009).
3. Information Sharing: There is less efficient sharing of information among distributed group members because the workspace is limited to digital information only. Any physical information, such as printed documents or books, would have to be owned by each member or scanned and sent to others in order to be shared. Furthermore, a computer network, such as a Local Area Network (LAN), Wide Area Network (WAN) or the Internet, is required to maintain the connection among group members during a distributed collaborative session. Any problems that arise with these networks can obstruct and even halt collaborative work.

The above difficulties show that there is a greater need for support in distributed collaboration than for co-located collaboration. That is why distributed collaboration is the main focus of CSCW research as well as this research. Many distributed groupware systems have been developed in an attempt to overcome these difficulties. The concept of groupware and the corresponding systems will now be discussed.

2.3.6 Groupware

CSCW supports collaborative groups through groupware, a class of software systems developed to assist work among group members. Groupware is a fusion of words "group" and "software". Groups can make use of groupware systems to help them communicate, maintain group coordination and share information during their work, which can occur at the same or different times, from the same or different places.

Many distributed groupware systems have been developed that support different aspects and environments of collaboration. Ten categories of distributed groupware systems will be described and classified, namely Document Management Systems (DMS), Electronic mail (Email), Forums, Group Decision Support Systems (GDSS), Instant Messaging (IM), Real-time Collaborative Editors (RTCE), Shared Whiteboards, Videoconferencing (VC), Voice over Internet Protocol (VoIP) and Workflow Management Systems (WMS) (see Table 2.3).

Table 2.3: Descriptions and examples of groupware system categories

Category	Description	Example
DMS	A system used to store and manage electronic documents.	SharePoint (Microsoft, 2014b)
Email	A tool used to exchange digital messages from a one person to one or more recipients.	Gmail (Google, 2014a)
Forums	An online message board where people can discuss topics, or threads, in the form of posted messages.	Stack Overflow (Stack Exchange, 2014)
GDSS	A web system designed to assist users in making decisions by facilitating group collaboration.	MeetingWorks (Tangient, 2014)
IM	An online chat tool with real-time text transmission over a network.	WhatsApp (WhatsApp Inc, 2014)
RTCE	A system that allows several people to simultaneously edit a computer file using different computers.	Google Drive (Google Docs) (Google, 2014b)
Shared Whiteboards	A collaborative tool that provides groups with an interactive workspace in which members can simultaneously create sketches and annotations to facilitate group work.	RealtimeBoard (RealtimeBoard Inc., 2014)
VC	A distributed group meeting that offers simultaneous two-way video and audio transmission over the network.	Skype (Microsoft, 2014c)
VoIP	A voice call, similar to a telephone call, that offers real-time audio transmission over a network.	TeamSpeak (TeamSpeak Systems, 2014)
WMS	A software system for the set-up, performance and monitoring of a defined sequence of tasks.	KiSSFLOW (KiSSFLOW, 2014)

A simple way to classify these groupware categories is based on Johansen's time-space matrix (see Table 2.4). Some categories involve systems that can be used in co-located and distributed collaborative environments and are therefore placed in more than one class.

Table 2.4: Johansen's time-space matrix (Johansen, 1988)

		Time	
		Same time (Synchronous)	**Different time (Asynchronous)**
Space	**Same place (Co-located)**	GDSS, Shared Whiteboards	DMS, Forums
	Different place (Distributed)	GDSS, IM, RTCE, Shared Whiteboards, VC, VoIP	DMS, Email, Forums, WMS

Table 2.4 shows the classification of ten categories of groupware systems according to time and space. Classification using these two characteristics results in four classes of collaboration, namely synchronous co-located, asynchronous co-located, synchronous distributed and asynchronous distributed. This research focuses on Synchronous Distributed Collaborative Work (SDCW).

Groupware systems have become more and more complex in terms of their capabilities. It is, therefore, not enough to use only the time-space matrix for groupware classification. Groupware can be classified according to the CSCW characteristics that they support (Penichet et al., 2007). This method provides a clear indication of each groupware system's functions and purpose. Table 2.5 shows the classification method applied to the synchronous distributed groupware categories.

Table 2.5: Classification of groupware systems (Penichet et al., 2007)

Groupware Category	**CSCW Characteristics**		
	Communicate	**Coordinate**	**Information Sharing**
GDSS	✓	✓	✓
IM	✓		
RTCE	✓	✓	✓
Shared Whiteboards			✓
VC	✓		✓
VoIP	✓		
Percentage	***83%***	***33%***	***66%***

In Table 2.5, each groupware category is classified according to communication, coordination, information sharing, time and space. A tick is placed in the corresponding cell if a particular category supports the characteristic. The percentages of each characteristic supported are given in the last row of Table 2.5 and three major observations can be made.

Firstly, 83% of the groupware system categories provide support for communication. This shows that most groupware systems provide support for communication. IM, VoIP and VC are all types of synchronous CMC, which is any communication among two or more people that occurs through the use of electronic devices connected via a network. These devices provide immediate communication by means of text, audio and video channels. Secondly, 66% of the groupware system categories provide support for information sharing. This shows that most groupware systems provide support for information sharing. GDSS, RTCE, shared whiteboards and even VC provide a shared workspace in which to work and share information.

Lastly, only two of the six groupware system categories provide support for coordination. This exposes a problem for existing groupware because coordination is an important aspect of SDCW. GDSS and RTCE, which are typically web-based systems, provide support for all three CSCW characteristics. Web-based interaction is primarily governed by the WIMP interaction paradigm (Krauß et al., 2009). GDSS facilitates and focuses on group decision-making and RTCE facilitates and focuses on collaborative document editing.

For both GDSS and RTCE, communication is typically supported by text-based channels and other CMC systems are required to support enhanced communication. For both system categories, information sharing includes a shared visual workspace. For GDSS, the workspace is a synchronized webpage providing access to shared information. For RTCE, the workspace is the actual document being edited and so only one document can be shared at a time. For both system categories coordination support is limited to the scope and nature of the system. GDSS coordination involves information that aids consensus such as real-time shared lists and voting results. RTCE coordination support involves information that aids simultaneous document editing such as who is working on the document and where they are currently editing.

2.4 Requirements of SDCW in a Shared Workspace

This section will define the requirements of SDCW in a shared workspace. Firstly, the mechanics of collaboration will be discussed (Section 2.4.1), followed by identifying the functional requirements (Section 2.4.2) and non-functional requirements (Section 2.4.3) of a shared-workspace groupware system.

2.4.1 Mechanics of Collaboration

Researchers, who studied group work in SDCW contexts, have found common actions and interactions that occur among group members, over and above the actual work they perform. These actions and interactions are called the mechanics of collaboration. There are seven mechanics of collaboration which are the following (Gutwin & Greenberg, 2000):

1. Intentional communication: Group members explicitly communicate with each other by verbal or written means. In a shared visual workspace, the workspace and the artefacts themselves are central to facilitating explicit communication.
2. Consequential communication: In addition to intentional communication, people also pick up information that is unintentionally given off by others as they go about their activities. This type of communication is important for smooth group interaction.
3. Coordination of action: People organize their actions in a shared workspace such that they do not conflict with others. Shared resources require turn-taking, and some tasks require that actions happen in a particular order. People also learn to predict one another's actions and use those predictions to make the group more effective.
4. Planning: Some planning activities happen repeatedly inside the shared workspace. For example, people divide and re-divide the task as they work, reserve specific areas of the shared workspace for their use, or consider various courses of action.
5. Monitoring: Many of the other mechanics of collaboration rely on the ability to monitor and gather information about others in the workspace. Much of this information is workspace awareness information (see Section 2.3.2).
6. Assistance: Group members provide help to one another when it is needed. Assistance may be opportunistic and informal, where the situation makes it easy for one person to help another, or it may be explicitly requested.
7. Protection: One may inadvertently alter or destroy another's work. Thus, collaborators must monitor their own work, noticing what effects others' actions could have and taking actions to prevent certain activities.

The mechanics of collaboration are important in shared-workspace groupware systems, in which group tasks involve objects and artefacts in a visual workspace. In these systems, if group members are unable to communicate effectively about the task, or intuitively coordinate their actions, performance and satisfaction will most likely be affected negatively. Although appropriate support for the mechanics of collaboration will not guarantee a groupware system's success, failure to support them will diminish the level of collaborative support and cause the system to be of no use.

2.4.2 Functional Requirements

An effective shared-workspace groupware system for SDCW must support communication, coordination and information sharing among group members. Therefore, the fundamental requirements of SDCW are to provide a shared visual workspace for conducting work and sharing information, enable communication among group members, and support group coordination and awareness during collaborative work.

Based on the mechanics of collaboration discussed in the previous section, and the collaborative tasks identified in Section 2.3.3, twelve functional requirements of SDCW in a shared workspace have been identified and presented in Table 2.6.

Table 2.6: Functional requirements of SDCW in a shared workspace

FR#	Requirement	CSCW Characteristics	Collaboration Mechanism
FR1.	Provide access to a shared visual workspace.	Info. Sharing	-
FR2.	Enable access to and annotation of shared information documents.	Info. Sharing	-
FR3.	Enable simultaneous creation, modification and deletion of work artefacts.	Info. Sharing	-
FR4.	Enable manipulation and organization of workspace items.	Info. Sharing	-
FR5.	Enable intentional communication.	Communication	Intentional com.
FR6.	Enable consequential communication.	Communication	Consequential com.
FR7.	Keep users updated on each other's actions.	Coordination	Coordination of action
FR8.	Keep a log of all the actions of the users.	Coordination	Coordination of action
FR9.	Enable division of workload amongst users.	Coordination	Planning
FR10.	Enable identification of the availability of users.	Coordination	Monitoring
FR11.	Enable monitoring of a user's actions.	Coordination	Monitoring
FR12.	Enable assistance amongst users.	Coordination	Assistance

The first four functional requirements involve the sharing of information among group members. These include providing a shared visual workspace that allows manipulation and organization of, and gives access to shared documents and work artefacts. Documents can be simultaneously viewed as sources of information and artefacts can be simultaneously created and modified in the workspace by group members.

The next two requirements involve intentional and consequential communication among group members. Communication is an integral part of collaboration and distributed group members must be able to communicate with each other in order to express their ideas, ask questions or give feedback. Additionally, members should be able to communicate through their actions in the workspace.

The final six requirements involve coordination among members. Group coordination is supported by workspace awareness, which requires group members to be continuously aware of each other's current and past actions in the workspace. This is done by allowing members to monitor each other's availability and actions. Group coordination also requires that group members can plan their work and help each other when necessary.

Protection was the only collaboration mechanism that was omitted from the requirements. It was decided that for the purposes of this research, all work would belong to the group and any work that is accidently destroyed is the loss of the entire group. The motivation for this decision was that providing protection would limit the collaborative nature of the system and promote individual work rather than group work. The non-functional requirements will now be discussed.

2.4.3 Non-functional Requirements

The non-functional requirements of SDCW in a shared workspace are presented in Table 2.7. These requirements are critical to the success of a system supporting SDCW. A shared-workspace groupware system that provides all the appropriate functionality, but fails to be intuitive, seamless and flexible, will fail because collaboration will be inefficient and ineffective.

Table 2.7: Non-functional requirements of SDCW in a shared workspace

Number	Requirement	Description
NFR1.	Intuitive	A shared-workspace groupware system should be intuitive and easy to use, promoting learnability and memorability.
NFR2.	Seamless	All the various characteristics of a shared-workspace groupware system should be integrated into one unified interface.
NFR3.	Flexible	A shared-workspace groupware system should be flexible enough to facilitate the collaborative styles and preferences of different groups.

These three non-functional requirements involve the UI and interaction of a shared-workspace groupware system. The shared workspace of a groupware system supporting SDCW is required to be easy and intuitive to interact with. Performing system functions must be easy to learn and remember so that the system is unobtrusive and does not require a group member's continuous attention (Seifried, Jetter, Haller, & Reiterer, 2011). This gives members the ability to focus their attention on working together as a group on the shared task. The workspace must be seamless by enabling communication, coordination and information sharing functionality all in one logical, consistent and aesthetically pleasing UI. A shared-workspace groupware system must also be flexible because groups are unique (Araujo, Santoro, & Borges, 2004). Different groups work together in different ways and a shared-workspace groupware system must be able to cater for various collaborative styles and preferences.

2.5 Conclusions

In this chapter, a review of relevant literature in the field of CSCW was conducted in order to define and describe the concept of collaboration and groupware, with specific attention given to SDCW. Five characteristics of CSCW were defined, namely Communication, Coordination, Information Sharing, Time and Space. These characteristics were used to classify existing categories of groupware systems.

Coordination was found to be of particular importance for SDCW. Research showed that awareness was fundamental to coordination among distributed group members. It was found, however, that many existing distributed groupware systems do not provide sufficient support for coordination, whereas other characteristics, namely communication and information sharing, were facilitated and focused on.

The concept of the mechanics of collaboration was defined as the common actions and interactions that occur among group members in a shared workspace environment. Seven mechanics of collaboration exist, namely explicit and consequential communication, coordination of action, planning, monitoring, assistance and protection. These mechanisms are important to shared-workspace groupware systems and failure to support them will diminish the level of collaborative support and cause the system to be of no use.

The key requirements of SDCW were identified, namely providing a shared visual workspace, the sharing of information, enabling communication among group members, and supporting group coordination and awareness during the collaborative work. Twelve functional requirements of a shared-workspace groupware were identified based on the mechanics of collaboration. Three non-functional requirements were also identified, namely intuitiveness, seamlessness and flexibility. The next chapter will review the interaction techniques of NUIs that could be incorporated into a shared-workspace groupware system as a solution to the issues of existing groupware systems.

Chapter 3: Natural User Interfaces

3.1 Introduction

This chapter presents the second Literature Review, which investigates NUIs. This chapter identifies and discusses the interaction techniques of NUIs and answers the third and fourth research question discussed in Chapter 1, namely:

RQ3. What are the existing techniques of NUI interaction?

RQ4. How can NUI interaction be incorporated into shared-workspace groupware systems to address the existing limitations and problems?

The above questions will be answered by means of a Literature Review, in which relevant and current research will be explored. Firstly, NUIs will be defined in detail (Section 3.2), and then an in-depth discussion of NUI interaction techniques will be presented (Section 3.3). This will be followed by an examination of existing groupware systems that incorporate NUI interaction (Section 3.4). The requirements of SDCW will be mapped onto NUI interaction techniques (Section 3.5) and the chapter will conclude with the key findings of the Literature Review (Section 3.6).

3.2 Defining NUIs

Human-Computer Interaction (HCI) began with Command Line Interfaces (CLIs), in which interaction with technology could only be achieved by inputting and executing particular text-based commands using a keyboard. HCI then progressed to the currently established Graphical User Interface (GUI), in which Windows, Icons and Menus are displayed on a monitor, and a special Pointing device is used to interact with them (WIMP) (Wigdor & Wixon, 2011).

NUIs are a new generation of interfaces, in which HCI is more natural and seamless than before. These new interfaces are moving away from the traditional input methods as they have been found to be mental and physical barriers between the user and the technology (Microsoft News Center, 2010). NUIs are the next generation of interfaces and provide a new way of thinking about how we interact with computing devices. NUIs provide a quick and enjoyable path from novice user to experienced user because they are focused on creating a more natural human-computer relationship than in the past, based on the user's preferences, capabilities and context (Wigdor & Wixon, 2011).

A formal definition of a NUI is the following (Blake, 2012, p. 4):

> *"A natural user interface is a user interface designed to use natural human behaviours for interacting directly with content".*

This definition reveals three important aspects of NUIs:

1. NUIs are designed: They require careful thought and planning in advance to make sure the interactions are suitable for the user, the content and the context.
2. NUIs focus on natural human behaviours: Their primary interaction techniques make use of abilities, skills and behaviours that are intuitive to humans such as touching, writing, speaking and gesturing.
3. NUIs allow users to directly interact with content: They focus on the content that is being displayed and invite direct interaction with it.

Eventually, NUIs will succeed GUIs as the main interface type (Blake, 2012). This is because NUIs have several advantages over GUIs, namely:

- The development of new input devices make NUIs capable of more flexible interaction techniques than GUIs, which are limited to the keyboard and mouse.
- NUIs are more natural to use and learn than GUIs because they exploit abilities, skills and behaviours that humans have naturally acquired.

The above statements can be further substantiated by looking at the history of HCI. CLIs, which were the main interface type of the time, became a specialized interface type and the predecessors of GUIs because GUIs have advantages over them. GUIs are more capable, flexible, usable and learnable than CLIs. These advantages are identical to those that NUIs now have over GUIs. Therefore, it can be concluded that NUIs are the continuation of the historical trend and will be the successors of GUIs (Blake, 2012).

The definitions and conclusions show the true potential of NUIs and the possible support for SDCW. Through careful design, focus on natural human behaviours and incorporation of appropriate NUI interaction techniques, a shared-workspace groupware system could be developed that effectively supports SDCW. The various techniques of NUI interaction will now be discussed.

3.3 NUI Interaction Techniques

An interaction technique is the fusion of input and output, consisting of all software and hardware elements, that provides a way for a computer user to accomplish a task (Hinckley, Jacob, & Ware, 2004). NUIs allow people to interact with technology by means of intuitive interaction techniques such as touch and stylus, speech recognition, in-air gestures and proxemics. In this section, these interaction techniques will be discussed in terms of what they are, how they work, and their advantages and disadvantages. The discussion will begin with touch and stylus (Section 3.3.1), followed by speech (Section 3.3.2). In-air gestures (Section 3.3.3) and proxemics (Section 3.3.4) will then be discussed. Finally, NUI devices that incorporate these interaction techniques will be identified (Section 3.3.5).

3.3.1 Touch and Stylus

Touch is the action of bringing a bodily part into direct contact with another entity, usually in order to interact, perceive, understand or appreciate through the tactile sense. Specifically, we use our fingers to touch. This important human ability is naturally developed in the very early stages of life. NUIs implement touch as a form of direct interaction with content (Blake, 2012). From the touch interaction perspective, the human finger can be seen as a natural input device. A stylus can be used as an alternate input device for touch interaction. A stylus is a pen-like device that allows the user to write or draw, which are common human skills. It can also provide the user with greater accuracy for content selection (Wigdor & Wixon, 2011).

Touch has recently become a popular interaction technique that has been adopted by many computing devices, especially mobile devices such as smartphones and tablets (Apple, 2013; Samsung, 2013). Touch interaction has advanced from the recognition of a single touch, to identifying more than fifty simultaneous touch points, the pressure of the touch and even recognition of other devices and objects (Microsoft PixelSense, 2012). The rise in prominence of these touch technologies has motivated researchers to investigate effective touch gestures.

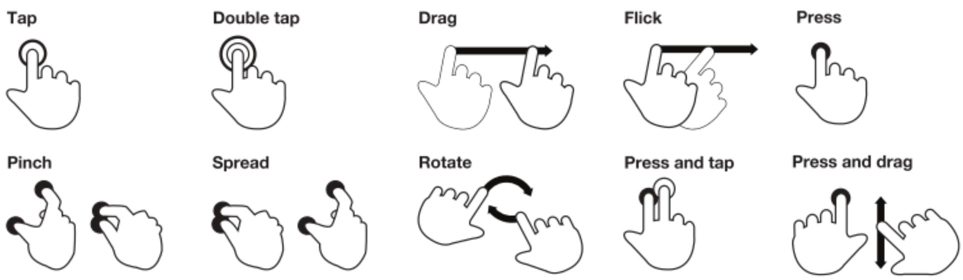


Figure 3.1: Typical touch gestures (Villamor, Willis, & Wroblewski, 2010)

Figure 3.1 shows the typical touch gestures that can occur during touch interaction. These include single touch gestures, namely single and double tap, drag, flick and press, and multi-touch gestures, namely pinch, spread, rotate, press and tap, and press and drag. Although some patterns have emerged to guide what functionality these gestures should support, such as tap for selection and pinch and spread for zooming, they are far from standard.

Touch interaction is not without its flaws. There are four limitations of touch interaction and technology, namely (Hampton, 2011):

- Feedback: While interacting with the touch technology, the user's hand and fingertips may occlude content being displayed and deprive a user of visual feedback. Furthermore, virtual buttons, i.e. "soft" buttons used in touch interaction, lack the tactile feedback of their physical counterpart.
- Precision: Users may have difficulty interacting with small items on the touch display because of the size of the human finger, which can cause user frustration. This is particularly prevalent when using touch technology with a small form factor such as mobile phones.
- Fatigue: Extended periods of use of touch technology may cause user fatigue. This is particularly prevalent when using vertical displays since users have nothing on which to rest their hands or wrists.
- Cost: Technology supporting touch interaction is generally costly.

Much research has been conducted to alleviate the limitations of touch interaction. Visual techniques have been implemented to improve feedback of touch interaction such as Ripples and Phosphor (Baudisch et al., 2006; Wigdor et al., 2009). A number of methods have arisen to enhance the precision and accuracy of touch technology. Devices with larger touch displays such as tablets have been developed to allow higher precision (Apple, 2013). Specific techniques and design guidelines aimed at improving precision include adding a fixed cursor offset, enlarging the target area and providing on-screen widgets to help with selection (Benko, Wilson, & Baudisch, 2006). Limited research has been conducted on fatigue associated with touch interaction, although smaller, tilted and horizontal displays can reduce user fatigue (Wigdor, Penn, Ryall, Esenther, & Shen, 2007). The cost of touch technology, although still relatively high, has reduced significantly in recent years owing to the popularity of touch technology and the research and development of low-cost touch-sensing techniques such as Frustrated Total Internal Reflection (FTIR) and Scanning FTIR (Han, 2005; Moeller & Kerne, 2010). These techniques have even initiated the building of DIY multi-touch tabletops (Castle, 2009).

3.3.2 Speech

Speech is a fundamental human skill used to communicate or express thoughts by means of spoken words. Speech is a natural communication tool among those with a common language. NUIs that implement speech interaction have the ability to capture, recognize and respond to the user's spoken commands. Speech recognition is accomplished using a microphone as an input device and a speech-recognition algorithm.

Speech interaction is advantageous when the user's hands are occupied or when the user is not focused on the interface, e.g. while driving (Tchankue, Wesson, & Vogts, 2010). Speech can also be used to input text by implementing speech-to-text systems, which eliminate the need to type on a keyboard (Hearst, 2011).

There are four main limitations of speech interaction, namely (Artman, 2010):

- Accuracy: Speech-recognition algorithms must be trained to recognise different languages and dialects. The training of these algorithms takes a long time because a lot of data is needed. Under-trained algorithms or speech input that differs from the algorithm's training data may result in inaccurate recognition.
- Fatigue: Extended periods of use of speech interaction can cause vocal strain.
- Environment: Speech interaction cannot be used in noisy environments, which limits its use in everyday life.
- Social acceptance: Speaking to computing devices is not yet socially acceptable. Public speech recognition inaccuracies can make a user feel embarrassed or awkward.

Speech recognition has advanced rapidly in recent times owing to the large data repositories being generated by mobile phone usage. It will continue to be improved because much research is being conducted to overcome the above limitations. It is a possibility that interacting with computing devices via speech will be the norm in the near future (Hearst, 2011).

3.3.3 In-air Gestures

In-air gestures, hereafter referred to as gestures, are a natural part of life. Every day, various forms of gestures are used for communication or interaction among people. NUIs that implement gesture interaction enable technology to recognize and respond to user gestures. Gestures have been defined as (Mitra & Acharya, 2007, p. 311):

> *"…expressive, meaningful body motions involving physical movements of the fingers, hands, arms, head, face or body with the intent of conveying meaningful information or interacting with the environment".*

Furthermore, gestures have been categorized into three types based on what part of the body is used (Mitra & Acharya, 2007):

1. Hand and Arm: Such as hand poses, pointing or sign language.
2. Head and Face: Such as nodding, smiling or winking.
3. Full Body: Such as walking, jumping or dancing.

The advantages of gesture interaction are that it provides a simple and easy to use interface, makes interaction between humans and computers more natural and gives people a new and enjoyable experience. There are three main limitations of gesture interaction, namely (Yan & Aimaiti, 2011):

- Fatigue: Extended periods of performing gestures can cause user fatigue.
- Misinterpretation: Body movements not intended for interaction can cause unintentional gesture commands to be performed.
- Limited Area of Activity: Camera-based gesture recognition systems require the user to be within view of the camera to perform gestures.

Despite these limitations, gesture interaction has become a widespread interaction technique and has been developed in various application areas, such as sign language recognition systems, navigation systems, medical research, gaming and augmented reality applications (Yan & Aimaiti, 2011). Gesture interaction has made its way into the latest developments in televisions such as the Samsung Smart TV, mobile devices such as the Samsung Galaxy S5 and gaming consoles such as the Xbox One (Bill Hughes, 2014; Microsoft, 2014e; Samsung, 2014).

Gestures have, therefore become an accepted form of interaction with technology. Furthermore, low-cost, programmable, 3D input sensors devices such as the Kinect and Leap Motion have given rise to many high-precision hand, face and full-body gesture-based applications (Microsoft, 2012; Motion, 2014). The last interaction technique, namely Proxemics will now be discussed.

3.3.4 Proxemics

Proxemics is the implicit knowledge and interpretation of spatial relationships. It is the theory of how people perceive, interpret and use distance, posture and orientation in order to mediate their relations to others. Proxemic theory correlates physical distance with social distance by defining four proxemic zones (see Figure 3.2) (Marquardt & Greenberg, 2012).

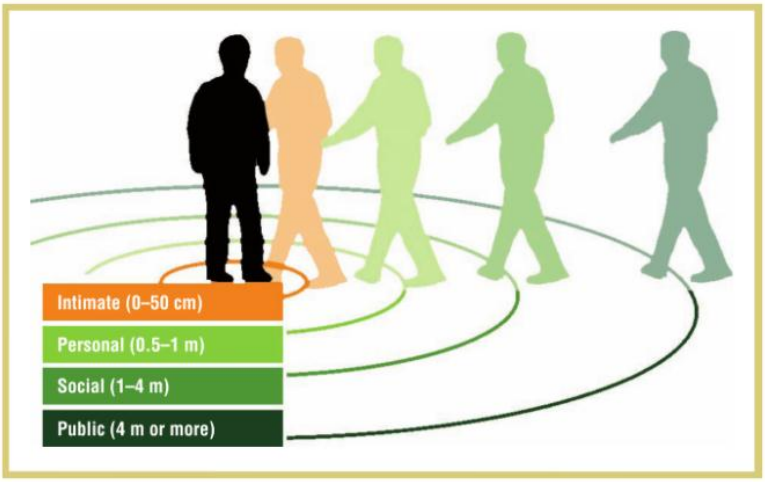


Figure 3.2: Proxemic zones (Marquardt & Greenberg, 2012)

There are five proxemic dimensions that are essential to defining proxemic relationships, namely (see Figure 3.3):

1. Distance: Absolute or relative distance between other entities.
2. Orientation: Which absolute or relative direction an entity is facing.
3. Movement: Changes of position and orientation of an entity over time.
4. Identity: Unique description of an entity.
5. Location: The physical environment and context in which an entity resides.

(a) distance *(b) orientation* *(c) movement* *(d) identity* *(e) location*

Figure 3.3: Proxemic dimensions (Marquardt & Greenberg, 2012)

The advantages of proxemic interaction include:

- Seamless interaction with technology.
- The technology disappears and become interweaved with our everyday lives.

The disadvantages of proxemic interaction include:

- Challenges in properly designing proxemic interaction systems.
- Increased privacy risks.

Over the past five years, researchers have been actively studying proxemic interaction (Ballendat, Marquardt, & Greenberg, 2010; Ledo & Greenberg, 2013; Marquardt & Greenberg, 2012). This research, along with the continuous trend of technological advancement, indicates that proxemic interaction will soon become prevalent in everyday life. This can already be seen in recent developments such as the Samsung Smart TV, smartphones, and Windows 8 boasting facial recognition capabilities and the Kinect enabling 3D motion tracking (Bill Hughes, 2014; Microsoft, 2014e; Samsung, 2014). These developments allow systems to capture and utilise the proxemic dimensions, namely distance, orientation, movement, identity and location.

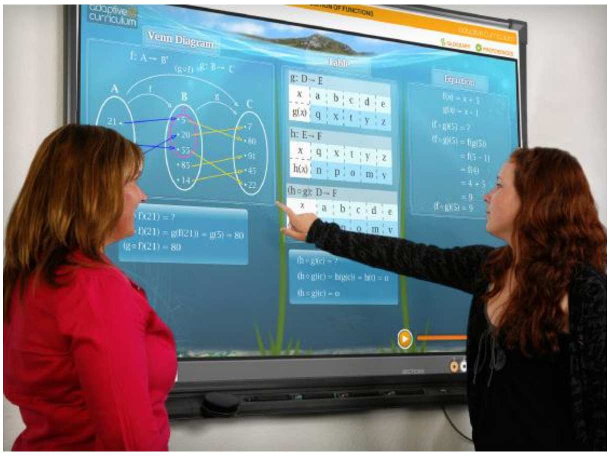
3.3.5 NUI Devices

A NUI device is defined here as a computing device that must be interacted with via NUI interaction techniques, such as touch, speech and gestures. These interaction techniques are built into the core design of the device. Devices that match these criteria include multi-touch tabletops, interactive whiteboards, smartphones and tablets. These devices can be grouped into stationary and mobile devices which will be discussed in the following two sections.

3.3.5.1 Stationary Devices

Stationary devices are devices that are confined to a specific location because of their size and thus do not support mobility. Stationary NUI devices include multi-touch tabletops and interactive whiteboards. A multi-touch tabletop is a large interactive display that incorporates touch interaction (see Figure 3.4a). Multi-touch tabletops allow up to four users to simultaneously interact with the device. Multi-touch tabletops are typically placed in a horizontal position although they may be placed in an upright position dependant on the particular model. An interactive whiteboard is a large enhanced display that is connected to a computer and allows for viewing, stylus or touch input, and collaboration by multiple users (see Figure 3.4b). Interactive white boards are typically mounted vertically on a wall. These are often used to aid collocated collaborative meetings.

(a) Multi-touch tabletop



(b) Interactive whiteboard

Figure 3.4: Stationary NUI devices (Luderschmidt, 2013; Sebit LLC, 2014)

3.3.5.2 Mobile Devices

Mobile devices are small, handheld devices that support mobility. Mobile NUI devices include smartphones and tablet computers. A smartphone is a compact, but powerful mobile phone with an operating system that allows various applications to be installed and run (see Figure 3.5a). Smartphones are the smallest form of touch devices and are therefore the most portable. The small screen, however, limits the amount of information that can be displayed at one time. A tablet computer, or simply tablet, is a lightweight mobile computer equipped with sensors, including a touchscreen, microphone, accelerometer and one or more cameras (see Figure 3.5b). Tablet devices offer a larger screen size, compared to smartphones, while maintaining the portability factor. This provides a real estate advantage to display more information.

(a) Smartphones *(b) Tablet computer*

Figure 3.5: Mobile NUI devices (CompareHero, 2013; Sony Mobile Communications Inc., 2014)

3.4 Related Work

In the previous chapter, existing groupware systems were classified in terms of the CSCW characteristics to highlight their strengths and weaknesses (see Section 2.3.6). In this section, the latest research developments of groupware systems will be investigated, with specific attention on those that incorporate NUI interaction techniques. Synchronous co-located (Section 3.4.1) and distributed (Section 3.4.2) groupware will be presented and discussed to see how NUIs have been incorporated to support collaboration in small groups.

3.4.1 Synchronous Co-located Groupware Systems

DeskPiles is a collaborative system prototype that supports digital information management across multiple devices in a multi-user, co-located environment (see Figure 3.6). The DeskPiles prototype was developed to support research that aims to facilitate the sharing and consolidation of knowledge in collaborative settings. A tile-based, zoomable, distributed user interface enables multiple users to collaboratively organize, annotate, cross-link and transfer digital resources using touch and stylus interaction in a co-located interactive workspace. The workspace is a meeting room consisting of a Microsoft Surface, an interactive wall display, and tablets (Milic-Frayling et al., 2010).

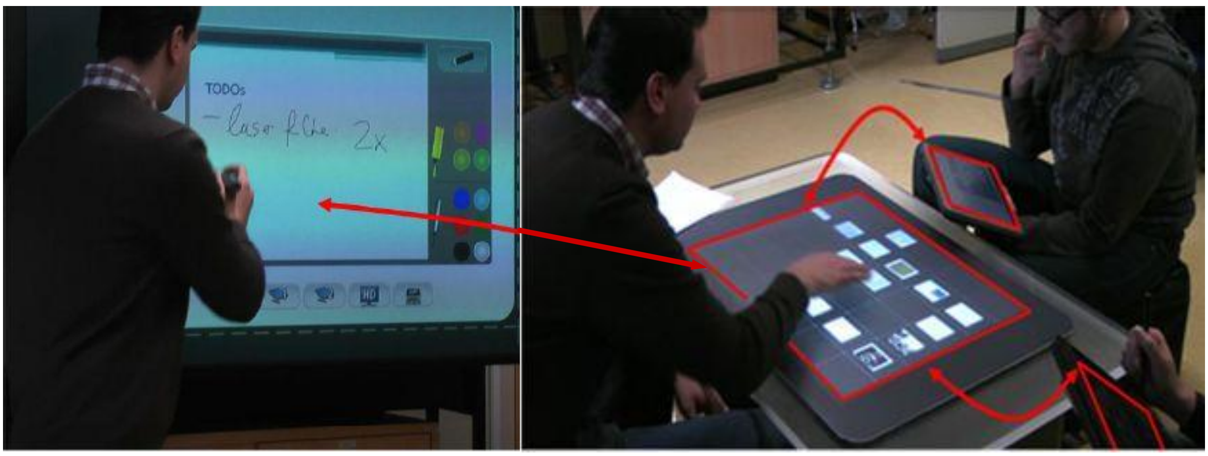


Figure 3.6: The DeskPiles interactive workspace (Milic-Frayling et al., 2010)

Code Space is a system prototype that implements touch and in-air gesture hybrid interactions to support co-located, small-group developer meetings by allowing equality in the access, control and sharing of information among group members across multiple personal devices and public displays (see Figure 3.7). The meeting space includes a shared, multi-touch wall display that provides different modes of interaction based on how many presenters and audience members are present. Mobile touch technologies such as smartphones and tablet PCs can be used to interact and share information with the wall display and the personal devices of other members. In-air gestures and hybrid interactions include pointing and manipulating with the arm, and pointing, manipulating, annotating and sharing with the arm and smartphone. These interactions, as well as user locations and movements, are captured by Microsoft Kinect sensors. Results of a pilot evaluation of Code Space indicated that interacting from a distance and sharing information across devices using natural in-air gestures and touch interactions could effectively support co-located developer meetings (Bragdon et al., 2011).

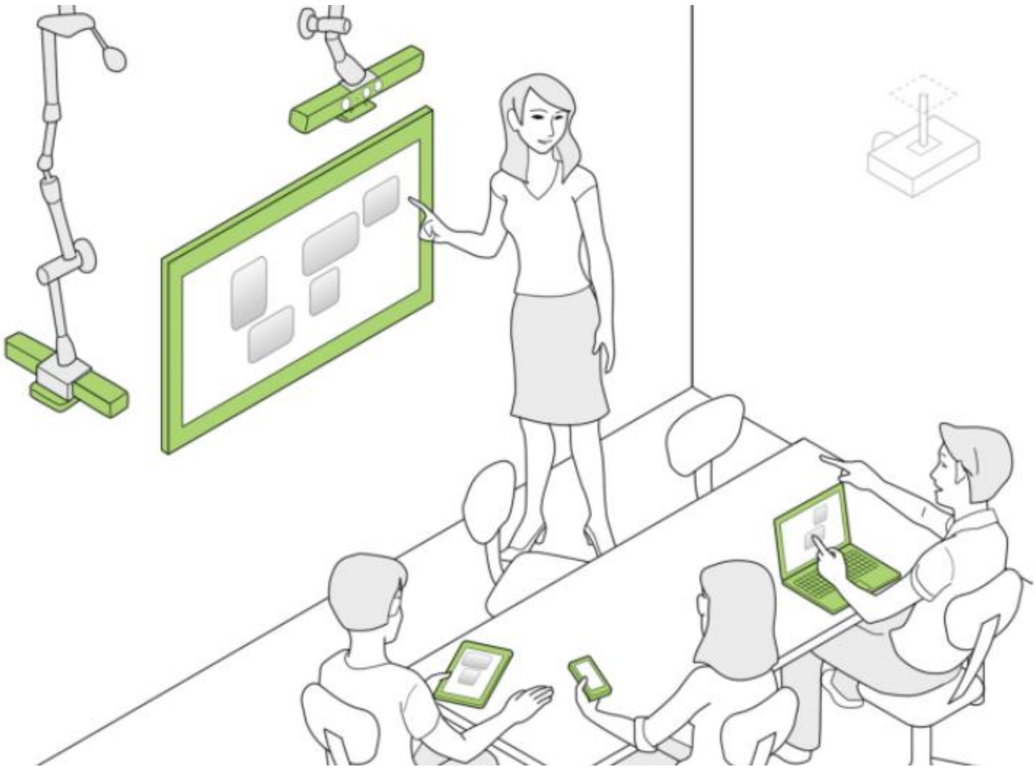


Figure 3.7: Code Space meeting environment (Bragdon et al., 2011)

The Natural User Interfaces for Collaborative Environments (NiCE) Discussion Room is a co-located collaborative meeting room design, in which an intuitive pen-based interface integrates digital and paper tools into a cohesive system that facilitates group work (see Figure 3.8). Group members can integrate their personal workspaces in the room to support open, active discussions and seamless content creation and sharing. The NiCE Discussion Room includes a large enhanced whiteboard that runs the NiCE sketching application, enabling users to draw original content or annotate and manipulate content imported from other paper and laptop interfaces in the room. Other tools and devices include pens, paper, personal laptops and specially designed furniture.

Users responded positively during the evaluation of the system and the results showed some distinct advantages in supporting co-located group meetings, such as the support of individual and group work, as well as the transitions between them, and a variety of collaborative styles, owing to the flexible interface design (Haller et al., 2010).

Figure 3.8: The NiCE Discussion Room (Haller et al., 2010)

CollaGIM is a groupware system that supports Group Information Management (GIM) in a synchronous co-located collaborative environment (Ditta et al., 2013). CollaGIM naturally and effectively facilitates multi-user GIM by means of a multi-touch tabletop, which is capable of recognising 32 simultaneous touch points (see Figure 3.9). The system, therefore, allows up to four members to work together at the same time. Groups members can collaborative search, organize, manipulate and share their personal information using CollaGIM. The user evaluation results of CollaGIM were positive. Results showed that the system effectively and efficiently supported GIM tasks for pairs of users. Furthermore, users reported that CollaGIM was easy and enjoyable to use and that they quickly learnt how to share their information and become productive.

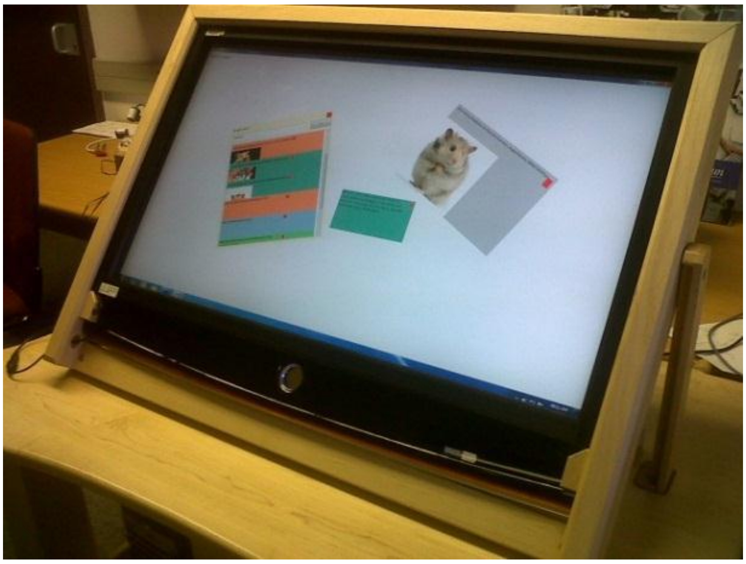


Figure 3.9: CollaGIM workspace (Ditta et al., 2013)

3.4.2 Synchronous Distributed Groupware Systems

Collaborative Slate (C-Slate) is a shared-workspace groupware system that was designed to improve synchronous distributed collaboration on horizontal surfaces. C-Slate supports collaborative reviewing and annotation of shared electronic documents among geographically separate group members. Each member's workstation consists of a horizontally mounted, stylus-enabled tablet display for accessing the shared workspace, a stereo camera for enhanced interaction, and a secondary display and webcam for audio and video conferencing (see Figure 3.10). Workstations are connected to each other across a network. Upon connection, an audio and video feed is established and a fully synchronized digital workspace is provided. Within the workspace, media items such as images, video, web pages and documents can be opened, reviewed and annotated.

C-Slate utilises a stereo camera, i.e. a camera with two or more lenses with a separate image sensor for each lens, and a real-time computer vision and machine learning technique for capturing and recognising a user's hands and physical objects placed over the display. The captured images of hands and objects are transmitted over the network to other C- Slates and visually overlaid onto the workspace, which provides a virtual embodiment of the user's hands as well as image sharing of physical objects such as written notes, drawings or game pieces. Furthermore, the recognition system is able to distinguish a user's hand from objects. Hand pose recognition enables multi-touch interaction on the surface and object recognition enables interaction via physical objects such as documents, stationery or mobile devices. Initial user feedback on the system was positive. Users said that C-Slate offers a natural way of collaborating remotely. For example, at a glance users can see if a remote user is interacting, writing or pointing to something on the screen. Users were enthusiastic about the multiple interaction techniques (Izadi et al., 2007).

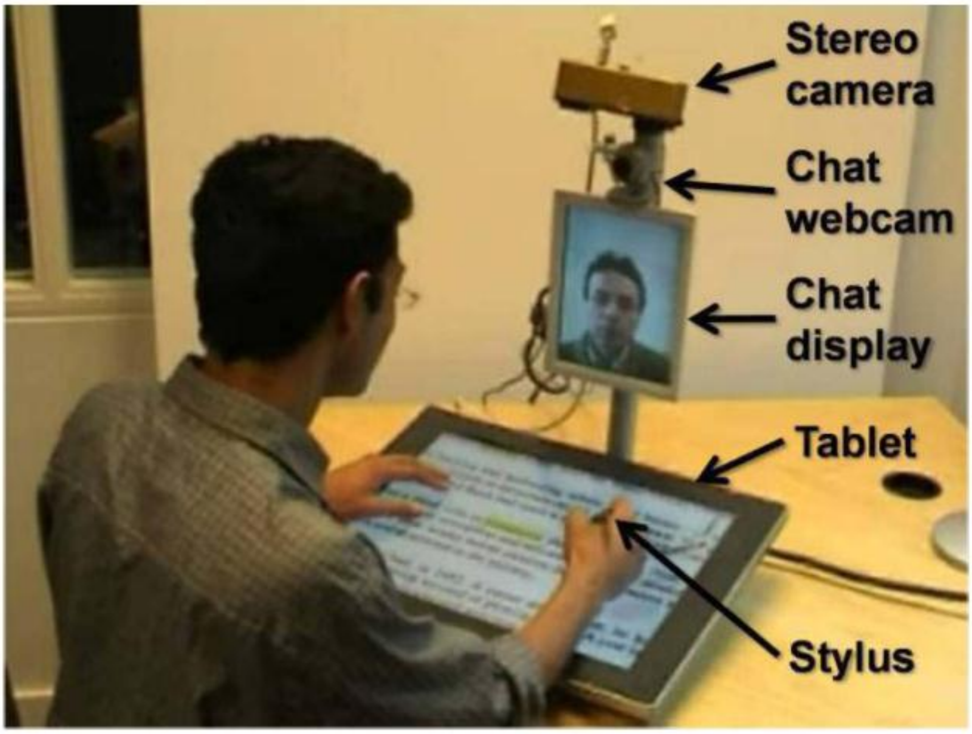


Figure 3.10: A C-Slate workstation (Izadi et al., 2007)

Ardaiz, Arroyo, Righi, Galimany, & Blat (2010) presented a Remote Multi-touch Collaborative Environment (RMCE) solution to support synchronous, distributed collaboration using multi-touch interaction within a shared virtual collaborative environment (see Figure 3.11). The RMCE system integrates videoconferencing with shared immersive spaces and multi-touch interaction to support collaborative discussion, manipulation and organization of shared objects using multi-touch interaction within a shared virtual collaborative 2D or 3D environment. A vertical multi-touch surface provides access to the shared workspace and a webcam provides videoconferencing capabilities. Awareness mechanisms such as fingerprints and finger-rays are built into the system. In a 2D environment, coloured fingerprints are used to represent a user's fingers, as they interact in the workspace. In a 3D environment, a finger-ray starts to grow when a user touches the screen and ends when it intersects with an object or virtual wall.

Preliminary results indicate that after adjusting to the virtual shared space, users are able to collaboratively manipulate shared objects with multi-touch interfaces. Users perceived communication to be natural when using the embedded videoconferencing channel, although shared artefacts often obstructed the video. Furthermore, fingerprints and finger-rays gave system feedback to the local user and activity awareness to the remote user. Finger-rays, however, proved to be unintuitive and difficult to use.

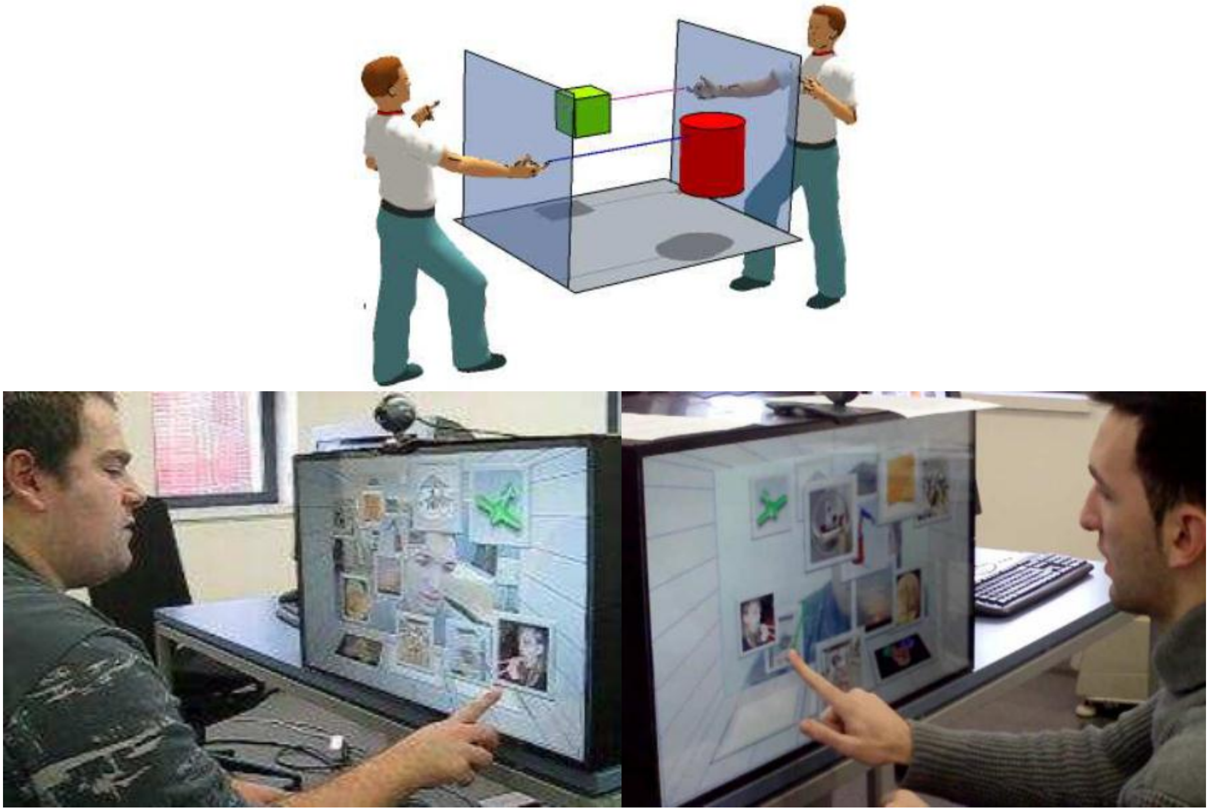


Figure 3.11: A Remote Multi-touch Collaborative Environment (Ardaiz et al., 2010)

Three’s Company is a groupware system designed to support three-way, real-time, distributed collaboration over a shared visual workspace (Tang et al., 2010). The system supports discussion, manipulation and organization of shared work artefacts. The system’s environment consists of three geographically separate workstations. Each remote workstation consists of a multi-touch tabletop surface as a collaborative workspace and a physical surrogate for each remote group member to enable group communication. Each surrogate includes a camera, speaker, microphone and LCD monitor. These are carefully aligned to appropriately represent a group member (see Figure 3.12). Additional cameras above the tabletops capture “shadows” of the users’ arms as they move over the workspace. A technique called trace pearls tracks each point of contact for each user with a trail that fades after two seconds. The arm shadows and traces are transmitted and overlaid onto the workspaces of the other group members, providing virtual embodiment for group awareness and coordination.

The evaluation of Three’s Company investigated the effects of varying two types of configurations during real-time distributed collaboration around a shared tabletop workspace, namely spatial and communication configurations. Two types of tasks were given to evaluation participants, namely single-orientation tasks, which involved text, and orientation-free tasks.

In the study of spatial configuration, same-side and around-the-table configurations were employed. The same-side configuration involved all distributed members effectively sitting in each other's laps, whereas the around-the-table configuration involved members sitting at different ends of the table. Results showed that 75% of users preferred the same-side configuration because all users have the same perspective of the shared workspace. Same-side did, however, increase simultaneous attempts to manipulate the same object and occasionally made user identification difficult. The around-the-table configuration enabled natural user identification and reduced manipulation conflicts, but caused readability issues in single-orientation tasks.

In the study of communication configurations, the presence and absence of communication channels was varied resulting in three conditions, namely Audio + arm Shadows (A+S), Audio + Video (A+V) and Audio + Video + arm Shadows (A+V+S). The results showed that more than 70% of the users preferred the A+S configuration. The audio feed was perceived to be the most important because it allowed conversation, which forms the basis of real-time collaboration. The presence of the arm shadows was found to reduce confusion and promote awareness of group members' activities. The video feed was rarely used while completing tasks and was thus not a necessity. However, the video feed was used between tasks and improved the group's experience.

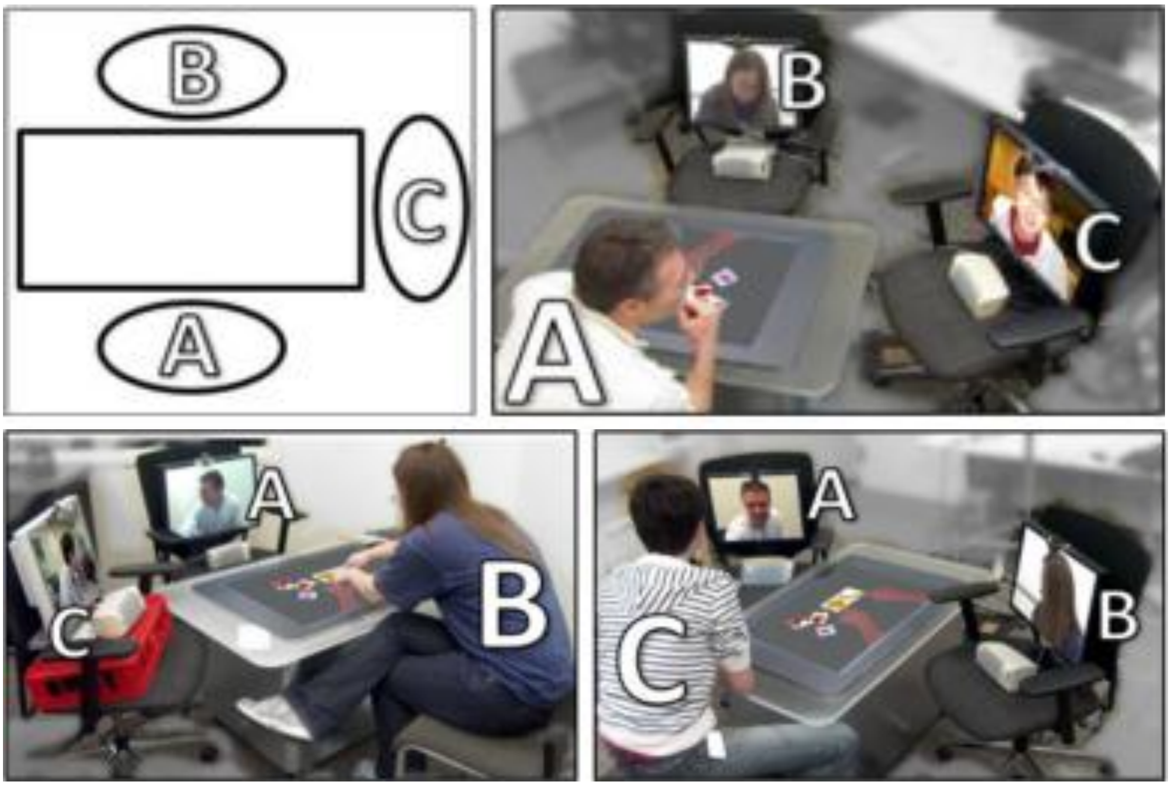


Figure 3.12: Three's Company collaborative environment (Tang et al., 2010)

KinectArms is a developer toolkit that simplifies the capture of distributed tabletop gestures and the display of those gestures through arm embodiments (Genest, Gutwin, Tang, Kalyn, & Ivkovic, 2013). As group members work in a distributed tabletop environment, KinectArms captures images of their arms by means of an overhead Kinect and transmits the images into the workspace (see Figure 3.13). The toolkit provides built-in effects to show height, to improve visibility and to provide movement traces.

Although it is not a groupware system itself, KinectArms enables designers to add rich arm embodiments to their systems without undue cost or development effort, greatly improving the expressiveness and usability of distributed tabletop groupware. Analytical evaluation of the KinectArms toolkit confirmed that it can be quickly and easily integrated with distributed groupware systems that comprise a multi-touch tabletop such as the distributed photo-sharing system shown in Figure 3.13. Furthermore, results showed that it provides expressive gestures with good performance and that it is extensible, allowing additional effects to be developed and added to the toolkit.

Figure 3.13: Distributed photo-sharing system testing KinectArms toolkit (Genest et al., 2013)

3.4.3 Discussion

This section discusses the groupware systems that were reviewed in the previous two sections. A summary of the system components will be given in terms of the devices and interaction techniques used and the functionality provided. The co-located systems (Section 3.4.3.1) will be discussed, followed by the distributed systems (Section 3.4.3.2).

3.4.3.1 Synchronous Co-located Groupware Systems

The devices and interaction techniques used and the functionality supported in each of the systems are summarised in Table 3.1.

Table 3.1: The devices and interaction techniques used in synchronous co-located groupware systems

System	Devices (H=Horizontal, V=Vertical)	Interaction Techniques	Functionality
DeskPiles	Multi-touch tabletop (H) Interactive whiteboard Tablets	Touch Stylus	Link Organize Annotate Manipulate Share
Code Space	Multi-touch surface (V) Smartphones Tablets Microsoft Kinect sensors	Touch In-air gesture Proxemics	Point Annotate Manipulate Share
The NiCE Discussion Room	Interactive whiteboard Anoto pens and paper Personal laptops	Stylus Pen and paper	Sketch Annotate Manipulate Share
CollaGIM	Multi-touch tabletop (H)	Touch	Search Organize Annotate Manipulate Share

From Table 3.1 it is clear that a device that is common to three of the four co-located collaborative systems is a large multi-touch display. Personal mobile technologies, such as smartphones, tablets and laptops, are also pervasive. Other devices include interactive whiteboard, Microsoft Kinect sensors and enhanced pen and paper. Touch or stylus interaction is the primary forms of interaction in all of the systems. Other techniques include in-air gestures and proxemic interaction. Common functionality supported by all the reviewed systems includes annotating, manipulating and sharing digital content. Other functions include linking, organizing, pointing and sketching.

From the positive results obtained by the user evaluations of the reviewed systems, it can be said that NUIs have been successfully incorporated into co-located groupware systems. The system reviews suggest that a general co-located collaborative system that incorporates NUI interaction is one in which advanced technological devices are setup in a meeting room to seamlessly support task work among group members. Key tasks include annotating, manipulating and sharing. Supporting these tasks is the main focus of co-located groupware systems because communication and coordination among group members are naturally mediated by the members themselves. Distributed groupware systems, however, require group communication and coordination to be computer-mediated. The latest groupware systems supporting SDCW while now be discussed.

3.4.3.2 Synchronous Distributed Groupware Systems

The devices and interaction techniques used and the functionality supported in each of the above systems are summarised in Table 3.2.

Table 3.2: The devices and interaction techniques used in shared-workspace groupware systems

System	Devices (H=Horizontal, V=Vertical)	Interaction Techniques	Functionality
C-Slate	A stylus-enabled tablet (H) Chat webcam and display Overhead stereo camera	Stylus Touch In-air gesture Object recognition	Audio and video conferencing Hand and object image overlays Review and annotate shared electronic documents
RMCE	Multi-touch surface (V) Webcam	Touch	Audio and video conferencing Fingerprints and finger-rays Manipulate and organize shared objects
Three's Company	Multi-touch tabletop (H) Surrogates (LCD, camera, speaker and microphone) Overhead camera	Touch In-air gesture	Audio and video conferencing Arm shadows and trace pearls Manipulate and organize shared objects
KinectArms Test System	Multi-touch tabletop (H) Overhead Kinect	Touch In-air gesture	Remote arm embodiments (with height indicators, enhanced visibility and identification, and motion traces) Manipulate and organize shared objects

From Table 3.2 it is clear that the key devices found in the reviewed groupware systems are multi-touch displays, overhead cameras and videoconferencing equipment, e.g. webcams. Two of the four systems use an additional display for videoconferencing, whereas the other system embeds the video into the main workspace display. Touch is the primary form of interaction in all four of the systems. Another dominant interaction technique was that of in-air gestures. Other forms of interaction were stylus interaction and object recognition. Functionality supported by all except one of the reviewed systems includes videoconferencing for audio and video communication. All systems supported awareness mechanisms for group coordination, and task work in a shared visual workspace. The manner in which videoconferencing is supported is the same for all systems. The tasks supported in most of the system is manipulating and organizing shared objects. A common theme in the awareness mechanisms for horizontal devices was observed. Overhead cameras were used to capture and transmit images of the group members' hands and arms as they worked over the workspace.

The synchronous distributed groupware system reviews provided insight as to what devices are used to support SDCW and how NUI interaction techniques have been incorporated. Furthermore, it was seen what functionality was used to support SDCW in terms of communication, coordination and information sharing in a shared workspace. The benefits and limitations of the reviewed systems in terms of the devices, interaction techniques and functionality, as well as the implications they have on this research will now be presented.

Firstly, three out of the four used multi-touch tabletops. This does not seem like a practical solution since only one user is working on an inherently multi-user device. Scalability is an issue in terms of cost since each participant requires a multi-touch tabletop. C-Slate incorporated a tablet device, which was regarded to be a much more practical and realistic option for a synchronous distributed groupware system. Secondly, the touch interaction technique was unanimous among the reviewed systems. This was for good reason because touch provides simple, direct and natural interaction with the workspace. Touch interaction, therefore, on tablet devices was chosen.

The functionality indicated that videoconferencing was the communication implementation method of choice. This seemed appropriate since videoconferencing is a widespread method for CMC. It was found in Three's Company, however, that the video feed was not of such importance during SDCW. Surrogates, therefore, seemed to be a substantial effort to preserve spatial relationships among group members. Built-in videoconferencing, as in the RMCE solution, seemed to be a good choice.

The functionality also indicated that overhead cameras and hand and arms embodiments were the coordination implementation method of choice. Although these methods seemed to provide good awareness of user presence and actions in the workspace, the setup is cumbersome and is impractical in terms of scalability. Each member requires an overhead camera that is compatible with the system, which could cost a lot. Furthermore, the actual embodiment technique would most likely not scale past three members without the workspace becoming cluttered with arms. A more practical, lightweight setup is needed. Basic proxemic interaction was chosen as a potential solution since limited research has been conducted on proxemics in SDCW.

Lastly, the functionality indicated that manipulating and organizing was the information sharing tasks implementation of choice. Systems that only supported these tasks seem rather trivial and game-like. C-Slate and some of the co-located systems implemented document reviewing and annotating, and sketching which seemed like more realistic collaborative tasks.

Interestingly, no systems implemented speech interaction. Group members' speech was captured, but it was simply propagated and outputted to all other members. It is clear though that speech was not used to interact with the systems because group members needed to communicate with each other. In summary, a combination of touch and proxemic interaction was selected to provide information sharing and coordination in a shared workspace. Videoconferencing built into the workspace was chosen as the method of providing communication among group members. Finally, the tasks that were chosen to be supported were viewing and annotating a document, and sketching graphic designs.

3.5 Mapping SDCW Requirements onto NUI interaction

Based on the review of existing shared-workspace groupware systems that incorporate NUI interaction techniques, touch and proxemic interaction were chosen as appropriate techniques to support SDCW. A mapping of the functional requirements of SDCW in a shared-workspace (see Table 2.6 in Section 2.4.2) onto NUI interaction techniques is proposed and is presented in Table 3.3.

Table 3.3: Proposed mapping of SDCW requirements onto NUI interaction techniques

Number	Requirement	Description
FR1.	Provide access to a shared visual workspace.	NUI system running on a multi-touch device that is connected to a network.
FR2.	Enable access to and annotation of shared information documents.	Touch interaction with documents.
FR3.	Enable simultaneous creation, modification and deletion of work artefacts.	Touch interaction with work artefacts.
FR4.	Enable manipulation and organization of workspace items.	Touch interaction to manipulate shared work and information present as visual objects in the workspace.
FR5.	Enable intentional communication.	Built-in audio and video conferencing.
FR6.	Enable consequential communication.	Proxemic interaction and synchronization of workspace.
FR7.	Keep users updated on each other's actions.	Proxemic interaction, synchronization of workspace and storing history logs of user actions.
FR8.	Keep a log of all the actions of the users.	Storing of history logs.
FR9.	Enable division of workload amongst users.	Shared workspace and audio conferencing.
FR10.	Enable identification of the availability of users.	Proxemic interaction and synchronization of workspace.
FR11.	Enable monitoring of a user's actions.	Synchronization of workspace.
FR12.	Enable assistance amongst users.	Simultaneous editing of shared work.

From Table 3.3, it can be seen that group members are provided with a shared visual workspace by means of a NUI system running on a multi-touch device that is connected to a network. The workspace can be interacted with via touch and proxemic interaction. The multi-touch device can display visual objects representing workspace items such as shared information documents and work artefacts. These objects can be manipulated using touch interaction. To maintain intuitive interaction with the objects, they should be able to be dragged, rotated and resized with one or two fingers. Touch interaction can be used to give members access to shared information documents, which can be viewed and annotated via touch gestures such as tap to open, drag to annotate and pinch and spread to zoom in and out. Touch interaction is also used to give members access to work artefacts, which can be created, modified and deleted via touch gestures.

Proxemic interaction can be used to enable consequential communication, keep users updated on each other's actions and enable identification of the availability of users. This can be done by the NUI system monitoring the distance and orientation of group members, relative to the device on which the workspace is running. For example, if a group member moves or faces away from the device for a certain period of time, the system can identify that the member is not available for collaboration. The system can then convey that information to the rest of the group, thereby keeping the group members up-to-date on each other's movements and enabling members to identify each other's availability. Proxemic interaction is, therefore, a form of consequential communication because simply by moving or facing away from the device, information is communicated to the rest of the group. The same applies for moving or facing towards the device.

3.6 Conclusions

A literature review was conducted to define NUIs. NUIs provide natural ways to interact with computing devices through various interaction techniques. Four types of natural interaction were defined and their advantages and disadvantages discussed. These techniques were touch and stylus, speech, in-air gestures and proxemic interaction. Devices that support NUI interaction techniques were discussed in terms of stationary and mobile technology.

In order to see how these NUI interaction techniques and devices are used, a review of related synchronous co-located and distributed shared-workspace groupware systems was conducted. The devices, interaction techniques and tasks supported by each system were discussed in order to select the appropriate interaction techniques for this research.

Multi-touch and proxemic interaction techniques were chosen and a mapping of the SDCW requirements onto these techniques was proposed. A NUI system running on a multi-touch device connected to a network could provide support for the access to a share visual workspace. Touch interaction could provide intuitive interaction with visual objects representing information documents and work artefacts. Proxemic interaction could be used to enable consequential communication, keep users updated on each other's actions and enable identification of the availability of users. The next chapter will propose the design and implementation of a prototype system to be used for testing of these interaction techniques. The prototype will then be evaluated to determine the effectiveness of the proposed techniques.

Chapter 4: Design and Implementation

4.1 Introduction

The preceding two chapters have investigated current research related to SDCW, groupware, and NUI interaction techniques. Chapter 2 identified the typical tasks performed using a groupware system, which were used to determine the functional requirements of SDCW in a shared-workspace. Chapter 3 included a discussion of NUI interaction techniques, on to which the functional requirements were mapped.

In this chapter a shared-workspace groupware system called GroupAware is proposed. This chapter involves the third and fourth phases of the DSR methodology, namely Design and Develop Artefact and Demonstrate Artefact. This chapter begins by identifying the development methodology that was used to design and implement the system (Section 4.2). The application domain is then briefly discussed (Section 4.3), followed by an in-depth discussion of the design of GroupAware (Section 4.4). GroupAware is designed to support the requirements of a groupware system using appropriate NUI interaction techniques. The details of the implementation of the prototype are then discussed in detail (Section 4.5) and the chapter concludes with a general discussion of the design and implementation of the GroupAware prototype (Section 4.6).

4.2 Development Methodology

Incremental prototyping was used to iteratively build the design artefact, namely GroupAware. Incremental prototyping occurs when the overall design solution is partitioned into smaller, independent components, called prototypes (Higher National Computing, 2007; Phillips, 1997). Each prototype is iteratively designed, developed and evaluated and then integrated into the final product. This process formed the Design cycle found in DSR as discussed in Section 1.7.

GroupAware was made up of four prototypes, namely Workspace, Distribution, Communication and Proxemics. The Workspace prototype involved the UI and functionality of the shared visual workspace. The Distribution prototype involved the characteristics of the distribution architecture such as using suitable protocols and message types, and handling synchronization and concurrency issues. The Communication prototype involved the audio and video conferencing among group members. The Proxemics prototype involved automating user availability by means of face detection methods. These prototypes are discussed further in Section 4.5.2. The integration of all four prototypes constituted the overall design artefact. This artefact was used to determine whether NUIs can effectively support SDCW by means of a user study, which is discussed in the next chapter.

4.3 Application Domain

In the third year of a Bachelor's Degree in the Department of Computing Sciences (CS) at Nelson Mandela Metropolitan University (NMMU), students are required to complete a year-long project working in teams of two to four members. The project involves developing a software solution to a particular problem. During this project, teams go through the four phases of the Agile Unified Process (AUP) methodology, namely Inception, Elaboration, Construction and Transition (Ambler, 2006). In the Elaboration phase, teams are required to create design artefacts for their software, such as UI designs, based on the requirements they have documented in the Requirements Document.

Team brainstorming sessions are crucial during this phase of development. Brainstorming is an informal and highly interactive session in which ideas are generated and artefacts are designed (Böhmer, Saponas, & Teevan, 2013). Brainstorming sessions typically occur in co-located environments (Rosenthal & Finger, 2006). Scheduling and attending co-located meetings, however, can be difficult for students and the option of working from home could be useful. Therefore, the application domain was chosen to be a small group of physically distributed members working on a software development project. More specifically, the group is moving from the requirements phase (Inception) to the design phase (Elaboration). Consequently, a typical group meeting would include tasks such as creating a system logo and UI designs. Groups would refer to their existing Requirements Document as a source of information, possibly making notes or correcting errors in the document. This scenario is generalizable to many software development projects and is thus the chosen application domain for this research.

4.4 Design

The first stage of the design process was identifying and establishing the functional requirements of a shared-workspace groupware system (see Section 2.4.2). The second stage was to create an architecture for such a system, upon which the design of the system could be based. In this section, an overview of the proposed system architecture is presented (Section 4.4.1). The system architecture comprises an application and distribution architecture. The application architecture (Section 4.4.2) and the distribution architecture (Section 4.4.3) are presented and discussed. The design of GroupAware is presented in terms of the functionality, data, UI and interactions (Sections 4.4.4 and 4.4.5). The design section concludes with a detailed look at the system architecture (Section 4.4.6).

4.4.1 System Architecture Overview

In this section, a system architecture is proposed that supports the requirements of SDCW in a shared-workspace that were discussed in Section 2.4. The architecture outlines the dependencies between each component of the system as well as how the information should flow within the system. A SDCW environment involves three components, namely the device (hardware), groupware system (software) and the network. The focus in this research is the design of a groupware system, which runs on a device that is connected to a network.

In general, a shared-workspace groupware system is required to allow group members to effectively work together in real-time from distributed locations. More specifically, the first functional requirement of SDCW that was established in Section 2.4 is stated below:

FR1: *Provide access to a shared visual workspace*

This functional requirement is the foundation of the entire system architecture. Four important concepts are communicated through this requirement, namely:

- Workspace: The system must provide a space in which users can work.
- Visual: The system must visually present the workspace to the users.
- Access: The system must allow user interaction with the visual workspace.
- Shared: The system must provide multiple users with simultaneous access to the visual workspace.

The first three concepts relate to the software application to be designed. An application architecture must be defined to provide a framework for the interactions of these concepts. The workspace concept involves the application logic, which defines the data and functionality of the application. The visual concept involves the user interface, which represents the state of the workspace. The access concept involves the input handling, which controls the user interaction of the system. These concepts correspond perfectly with the components of the well-established Model-View-Controller (MVC) architectural pattern (Reenskaug, 1979). Thus, the application architecture will be based on the MVC pattern, which will be discussed in the next section.

The last concept, namely shared, relates to the manner in which the workspace will be shared among distributed group members. A distribution architecture must be defined to provide a framework for the connectivity of the workspace over a network. It must be carefully designed in order to effectively and efficiently connect and synchronize the workspace among group members.

There are five successful distribution architectures that exist, namely Centralized Core with Thick Client, Generic Thin Client, Centralized Mixer with Broadcaster, Replicated Input Broadcasting and Replicated State Synchronization (Graham, Phillips, & Wolfe, 2006). Based on a comparison of these distribution architectures, the Centralized Mixer with Broadcaster was chosen, which comprises a server that broadcasts received data to all connected clients. The architecture and the motivations for choosing it are discussed in Section 4.4.3.

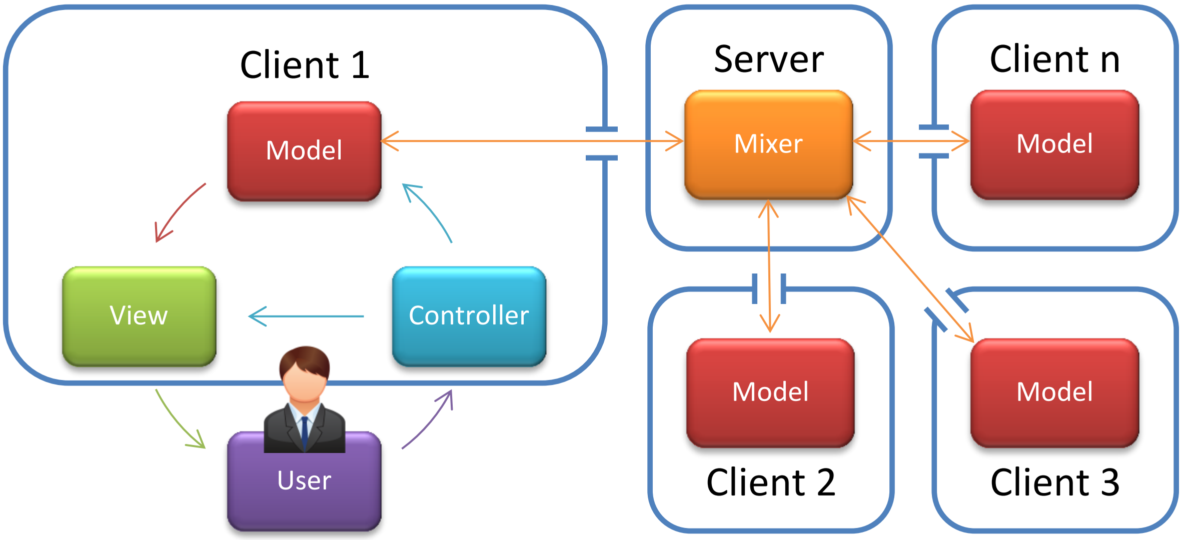


Figure 4.1: System architecture overview

Figure 4.1 presents an overview of the proposed system architecture. Client 1 shows the MVC components within the software application, with which the user interacts. The model is kept synchronized with all the other clients' models through a network communication channel that allows state changes to be sent and received to and from the server. The server acts as a broadcaster, sending all the data it receives to all other clients. Only the models of Clients 2, 3 and n are shown for the sake of brevity. The application and distribution architecture are discussed in detail in the following sections.

4.4.2 Application Architecture

The proposed application architecture that will be employed is based on the Model-View-Controller (MVC) software architectural pattern, which was first introduced in the 1970s (Reenskaug, 1979). The MVC architectural pattern has subsequently evolved giving rise to other patterns such as Hierarchical Model-View-Controller (HMVC), Model-View-Adapter (MVA), Model-View-Presenter (MVP), Model-View-ViewModel (MVVM) (Fowler, 2006). The core MVC pattern was chosen because of personal preference and previous experience with it.

MVC is an architectural pattern that divides a software application into three interconnected components. These components separate the internal representations of information from the way that it is presented to and manipulated by the user. The central component of MVC, the model, defines the functionality of the application in terms of its problem domain. The model directly manages the application's data and logic. The view is the visual representation of the information in the model, in the same way that a graph visually represents a set of information. The third component, the controller, accepts user input and converts it to commands for the model or view.

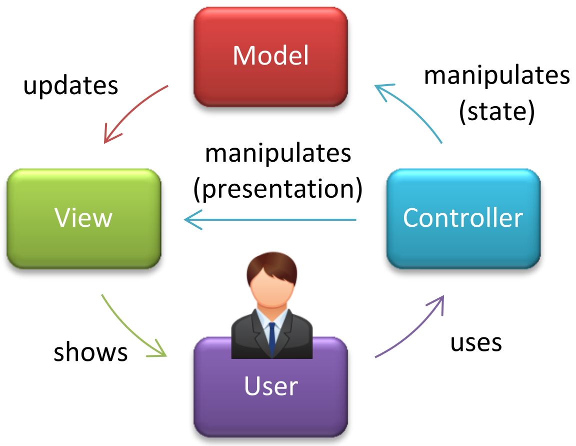


Figure 4.2: Typical interactions of the MVC components

In addition to dividing an application into three components, the MVC pattern defines the interactions between these components. Figure 4.2 shows the typical interactions of the MVC components of an application and the way a user interacts with the application. The process begins with the user interacting with the application by means of the controller. The controller can send commands to the model to manipulate the model's state (e.g. a user editing a document). As soon as there has been a change in its state, the model updates the view. The controller can also send commands directly to the view to manipulate the view's presentation of the model (e.g. a user scrolling through a document). Lastly, the user sees the changes on the updated view. The distribution architecture will now be discussed.

4.4.3 Distribution Architecture

The proposed distribution architecture that will be employed is called the Centralized Mixer with Broadcaster (Graham et al., 2006). This architecture involves situations where input data from session members needs to be broadcast to all other group members. Data from members may need to be mixed together to create a unified data stream, such as voice data in Skype (Microsoft, 2014c). The data may be identical for all members or customized for individual members or groups of members.

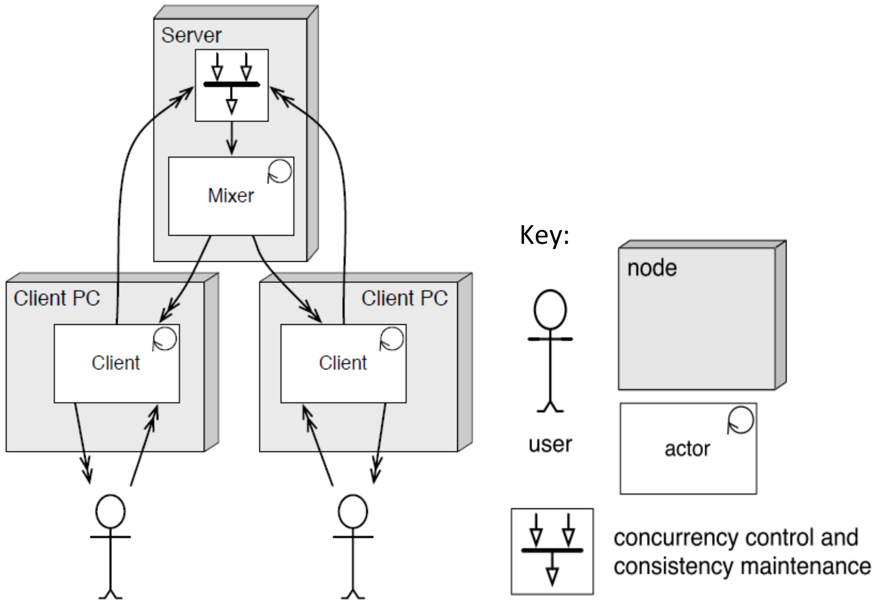


Figure 4.3: Centralized Mixer with Broadcaster distribution architecture (Graham et al., 2006)

As Figure 4.3 indicates, each client application is able to send data to the server. The server is typically located on one of the clients' devices. The server's Concurrency Control and Consistency Maintenance (CCCM) unit receives client data. The CCCM is responsible for resolving any conflicts between client operations, and ensuring all operations are reliable and correctly ordered. The data is passed to the mixer, which, if necessary, mixes the data and then broadcasts one or more streams of data to all clients. The clients then update themselves accordingly.

An advantage of the Centralized Mixer with Broadcaster architecture is the reduction of bandwidth consumption. The architecture reduces bandwidth consumption since each client communicates with only one other node, rather than requiring each peer to communicate with each other peer. This aids scalability because the use of bandwidth grows linearly with the number of clients, not quadratically. Reducing bandwidth is a key advantage in shared-workspace groupware systems because these systems require frequent workspace updates and are thus bandwidth intensive. Scalability becomes important if the number of distributed group members increases, which is possible because the distributed nature of the group allows people from anywhere in the world to join.

Another advantage of the architecture is the increased control in the trade-off between fidelity and feedthrough time. Fidelity is the degree to which a client's workspace is an up-to-date representation of the shared workspace. Feedthrough time involves the time from a user performing an action to other users seeing the result. The architecture allows the selective reduction of fidelity in order to improve feedthrough time. Clients can be grouped by their available bandwidth and processing time. Different qualities of stream (e.g. varying frame rate in video and application sharing or frequency range in voice transmission) can be provided to each group.

The main disadvantage of the architecture is that the server represents a single point of failure. If the server crashes, the collaborative session cannot continue. This could have a negative effect on system availability, which is the percentage of time that the groupware system is up and available for use. The advantages of this architecture are deemed to outweigh this disadvantage and thus the architecture was chosen as the distribution architecture of GroupAware. The design of GroupAware's model will now be discussed.

4.4.4 Model Design

As mentioned in Section 4.4.2, the model is the central component in the MVC architectural pattern. It defines functionality of the application in terms of its problem domain and manages the data stored by the application. This section discusses the design of GroupAware's model by defining the functionality (Section 4.4.4.1), the data (Section 4.4.4.2) and the manner in which the model is shared across all clients (Section 4.4.4.3).

4.4.4.1 Functionality

The functionality of GroupAware is based on the functional requirements established in Section 2.4.2. The main functionality that is required of a shared-workspace groupware system is the following:

- Provide shared workspace.
- Manipulate and organize work artefacts.
- View text documents.
- Create graphic documents.
- Enable communication.
- Provide workspace awareness.

Since group members are distributed they will each have their own physical workspace. The physical workspace will contain a primary computing device with multi-touch capabilities, a microphone and a webcam. The device will be running GroupAware and give members access to the shared visual workspace, through which they can access all the functionality of GroupAware. The microphone and webcam will enable communication among group members.

The workspace will be presented as a multi-touch interface that will contain widgets. A widget is an interactive visual element that represents or provides control of data in the workspace. These widgets will effectively represent shared documents and work artefacts. They can be manipulated and organized within the workspace via touch gestures such as drag, rotate, pinch and spread.

Group members will be able to simultaneously access text documents to use as shared information sources. These text documents will be read-only because multi-touch interaction is not well-suited for heavy text editing. Group members will be allowed to make annotations on the text document similar to a Portable Document Format (PDF) file (Adobe Systems, 2014a). Members will be able to create new graphic documents that can be simultaneously modified. Members will have access to drawing tools such as pencil, eraser and line colour and thickness. These documents can be removed from the workspace and loaded into the workspace at a later stage, or deleted permanently.

Each group member's interactions with the text and graphic documents will be immediately sent to the other members, thereby keeping all members updated of each other's actions within the workspace. This allows group members to maintain awareness and monitor each other's work. Thus, the artefacts themselves enable consequential communication. All actions are also stored in a history log that can be viewed by all group members at any time during the collaborative session.

GroupAware will enable natural communication among group members via an audio and video feed. The audio feed is automatically established among group members when they join the session and is constantly active. Each group member will be able to hear all other members. Group members can use the audio feed to divide the workload amongst each other, have discussions about their work and ask others for assistance.

Research has shown that, although an audio feed is vital, a video feed is not necessary for effective collaboration in shared-workspace groupware (Tang et al., 2010). This is because group members are focused on the information in the workspace and do not need to look at each other while working. It was found, however, that the video made users feel as if they were part of a group and they used the video feed while they were discussing things about the task. Thus, a constant video feed will not be provided, but members will be given the option of enabling and disabling the video feed.

GroupAware will implement basic proxemic interaction, which will determine whether a group member is within physical interaction distance to the device on which GroupAware is running and whether the user is facing the device. GroupAware will therefore track two proxemic dimensions of the group member, namely Distance and Orientation. This function will help group members identify each other's availability. As long as the user is close to the device and is facing the display, their status will be labelled as "Available". If, however, the user is far from the device or is not facing the display for a certain amount of time, the user's status will be labelled as "Away". If a user is close to the device and faces the display after being labelled as “Away”, the user’s status will again be labelled as “Available”. As previously mentioned, GroupAware will store all the actions performed by other group members. Upon the user’s return, GroupAware will open the history log showing all the actions stored during the time that the user was away, which the member can then quickly browse through to see what was missed and continue working seamlessly with the group.

The following table shows how the above functionality satisfies the functional requirements of a share-workspace groupware system.

Table 4.1: The satisfying of the functional requirements

FR#	Functional Requirement	✓	Quote
FR1.	Provide access to a shared visual workspace.	✓	“The device will be running GroupAware and give members access to the shared visual workspace”.
FR2.	Enable access to and annotation of shared information documents.	✓	“Group members will be able to simultaneously access text documents to use as shared information sources”.
FR3.	Enable simultaneous creation, modification and deletion of work artefacts.	✓	“Members will be to able create new graphic documents that can be simultaneously modified”.
FR4.	Enable manipulation and organization of workspace items.	✓	“The workspace will be presented as a multi-touch interface that will contain widgets” that “can be manipulated and organized within the workspace”.
FR5.	Enable intentional communication.	✓	“GroupAware will enable natural communication among group members via an audio and video feed”.
FR6.	Enable consequential communication.	✓	“The artefacts themselves enable consequential communication”.
FR7.	Keep users updated on each other’s actions.	✓	“Each group member’s interactions with the text and graphic documents will be immediately sent to the other members”.
FR8.	Keep a log of all the actions of the users.	✓	“All actions are also stored in a history log”.
FR9.	Enable division of workload amongst users.	✓	“Group members can use the audio feed to divide the workload amongst each other”.
FR10.	Enable identification of the availability of users.	✓	“GroupAware will implement basic proxemic interaction” that “will help group members identify each other’s availability”.
FR11.	Enable monitoring of a user’s actions.	✓	“Each group member’s interactions with the text and graphic documents will be immediately sent to the other members” which “allows group members to … monitor each other’s work”.
FR12.	Enable assistance amongst users.	✓	“Group members can use the audio feed to … ask others for assistance”.

Certain data must be captured, stored and managed within GroupAware in order for the functionality to be appropriately supported. These data requirements are discussed in the next section.

4.4.4.2 Data

In Chapter 1, the following definition of SDCW was given:

> *SDCW is when a **group of people** work together toward a common goal at the **same time from different locations**.*

This definition implies that the system needs to capture and store the data of each distributed group member. GroupAware stores this data in a class definition called User. The User class stores data that can help group members identify each other in the workspace such as a user's name, avatar and colour. It also stores data that can help group awareness and coordination among members, such as a user's status, the last time the user was seen, and the last document the user worked on.

From the same definition in Chapter 1, another important aspect can be highlighted:

> *SDCW is when a group of people **work together toward a common goal** at the same time from different locations.*

The application domain (see Section 4.3) contextualized the common goal with the following:

> *…teams are required to **create design artefacts** for their software, such as UI designs, **based on the requirements they have documented** in the Requirements Document.*

The above implies that an existing external text document will be used as a shared source of information to collaboratively create graphic design documents that will drive the project forward. Thus text and graphic documents are the two key work elements of GroupAware. Since these two elements are both documents, they share common data properties. The shared data is combined into a class definition called Document. The data stored by the Document class includes the name of the document, a list of users currently working on the document and the changes made on the document. The Document class also stores meta-data such as the author of the document, the time it was created, the user that last modified the document and the time of the modification.

All group-related actions performed by members are stored in a class definition called Command. The Command class stores the type of action, who performed the action, when the action was performed and any data relating to the action. Group-related actions include session actions, document actions and drawing actions. Session actions include Join and Leave. Document actions include Create, Delete, Open, Close, Save and Rename. Drawing actions include Draw, Erase, Undo and Redo. Drawing data is stored in a class definition called Stroke. The Stroke class stores data such as the points that make up a stroke and the type, colour and thickness of the stroke. Every action is stored within a history log, which members can view at any time during the collaborative session.

The data discussed above makes up the model of each client of GroupAware. This model is synchronized among all clients connected to a server via a network. Thus, all the clients together make up a shared model of GroupAware. The manner in which the model is shared is discussed in the following section.

4.4.4.3 Shared Model

The model of the shared workspace is constantly kept up-to-date for each group member's client application through a client-server, message-based transport protocol. The data stored by the client model is sent to the server via multiple data channels that transport messages of a defined type (see Table 4.2). These channels operate within the Transmission Control Protocol (TCP), which defines connection-oriented communication between the client and the server (Postel, 1981). TCP provides reliable and ordered delivery of messages. The server broadcasts all incoming messages to all connected clients, regardless of the message type and the channel upon which they were received. Clients are connected to the server on a single port, through which all channels operate.

Table 4.2: Data channels used to transport messages

Data Channel	Type	Description
Session	Session Message	A channel for session updates such as a client joining or leaving.
User	Object	A channel for user data updates such as user status changing.
Document	Object	A channel for document data updates such as creating or loading a document.
Command	String	A channel for synchronizing user actions such as opening or closing a document.
Drawing	Object	A channel for synchronizing drawing-related user actions such as drawing or erasing.

Table 4.2 shows the different data channels to be used in the shared model of GroupAware. These include Session, User, Document, Command and Drawing channels for updating and synchronizing the state of the workspace for each group member. This concludes the model design. The design of the view and controller will be discussed in the following section.

4.4.5 View and Controller Design

This section discusses the design of both the view and controller components of the application. The view involves the GroupAware's UI design and the controller involves the user interaction with GroupAware. There are three fundamental elements found within the GroupAware's UI design, namely the workspace, user labels and widgets. These elements will be discussed in terms of their visual (view) and interaction (controller) design in the following sections.

4.4.5.1 Workspace

The workspace is the bottommost UI element and it consists of a fullscreen interactive canvas with a dark background (see Figure 4.4). The fullscreen canvas gives the user the maximum space to work with and creates a fully-immersive experience. The dark background helps to reduce eye strain because a high-definition surface with bright colours and white space can cause fatigue of the user's eyes.

Figure 4.4 shows what the user sees when GroupAware is first started. The user is welcomed as a guest and given a simple starting instruction by the Main Menu widget, which is placed in the centre of the workspace. Tapping anywhere in the workspace instantly brings the Main Menu widget to the location of the tap. In order to join a collaborative session, the user must first create a profile. This involves choosing a username and avatar, and indicating handedness.

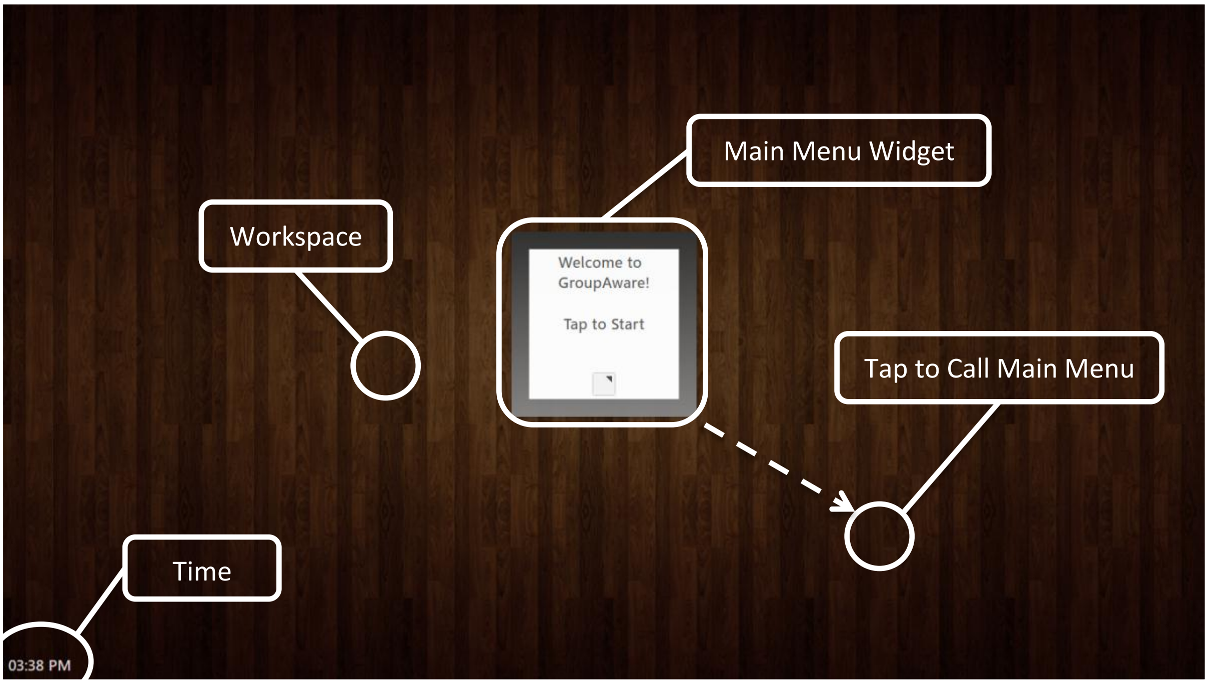
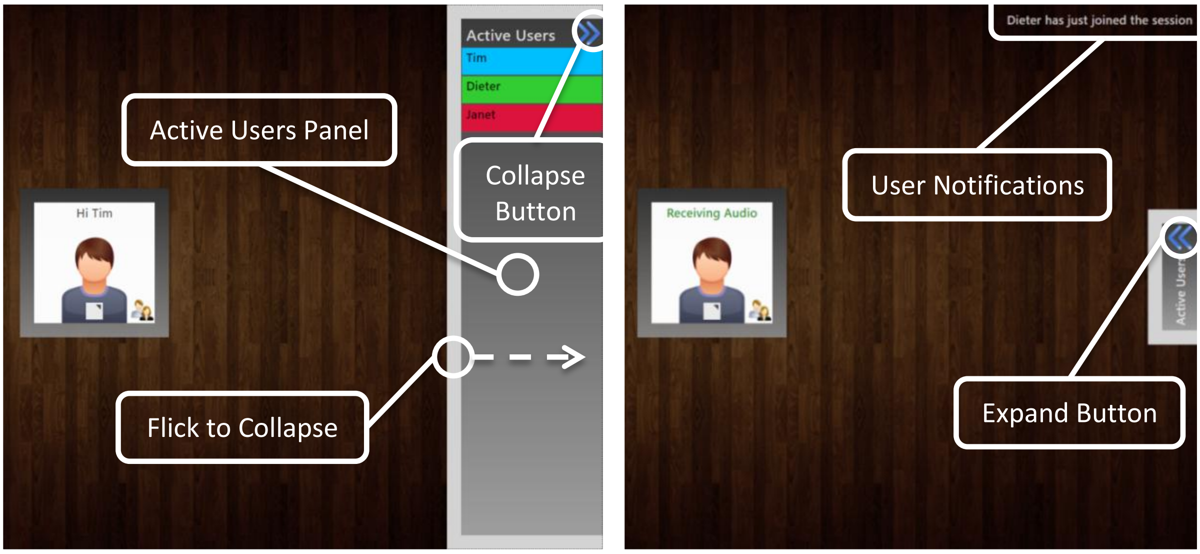


Figure 4.4: UI design of the workspace upon start-up of GroupAware



(a) After creating a profile and joining the session *(b) Collapsed panel and session notifications*

Figure 4.5: Active Users Panel

Once a user has created a profile and joined a collaborative session, a unique colour is automatically assigned to the user and a collapsible Active Users Panel indicates who else is currently in the session (see Figure 4.5a). The panel is displayed on the right or left hand side of the workspace, depending on the handedness of the user. The panel can be collapsed by either tapping the "Collapse" button or flicking the panel towards the side of the workspace. Figure 4.4b shows the collapsed Active Users Panel, which provides more space for the user to work with. When the panel is collapsed, session notifications are displayed in the top-right corner of the workspace.

The workspace moves away from conventional Windows interface elements such as WIMP and moves toward NUI interaction techniques. This is done by implementing custom interactive user labels and widgets with touch and proxemic interaction.

4.4.5.2 User Labels

A user label is a listbox item containing a user's name and colour. User labels are analogous in design to the divider tabs used in physical documents (see Figure 4.6).

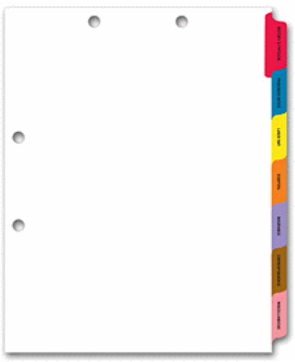
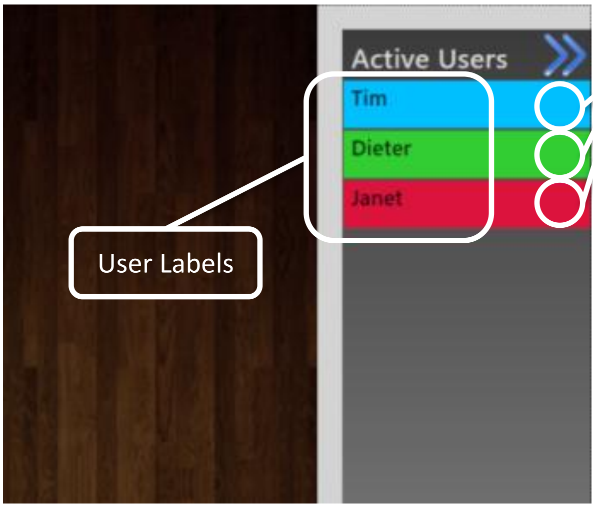


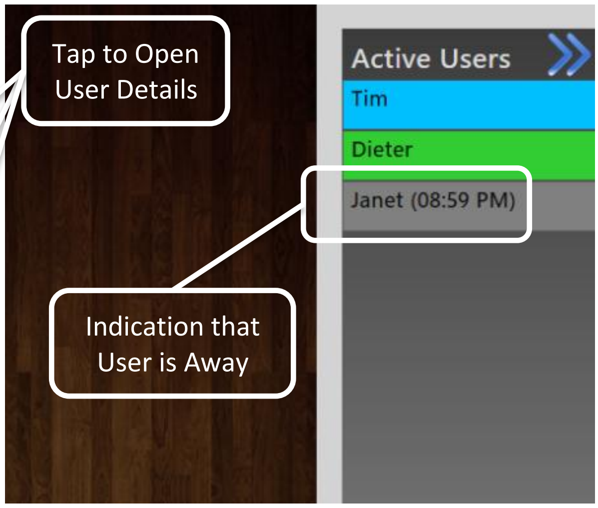
Figure 4.6: Example of divider tabs on a physical document

User labels are a lightweight awareness mechanism to help group members determine the availability of other members and where they are working at any given time. User labels are first presented in the Active Users Panel (see Figure 4.7a). Tapping a user label opens the details of that user (see Section 4.4.5.8).

A user label displaying a user's colour indicates that the user has joined the session and is available for collaboration. GroupAware incorporates basic proxemic interaction to determine user availability. GroupAware labels users as away when they physically move or face away from the device. If a user is away, the corresponding user label goes grey and displays the time when the user was last seen (see Figure 4.7b).



(a) The UI design of user labels



(b) When a user (Janet) is away

Figure 4.7: User labels

Additionally, user labels are presented in widgets such as the Document, Drawing and Text Viewer. The design of these and other important widgets will be discussed in the next sections.

4.4.5.3 Custom Widget Design

A widget is an interactive visual element that represents or provides control of data in the workspace. Widgets can be interacted with and manipulated by touch gestures such as tap, drag and rotate. Widgets are surrounded by a border to allow actions such as moving, rotating and resizing within the workspace. Within the border is the content of the widget that displays information or provides system functionality. Figure 4.8 illustrates the general design of a custom widget used within GroupAware.

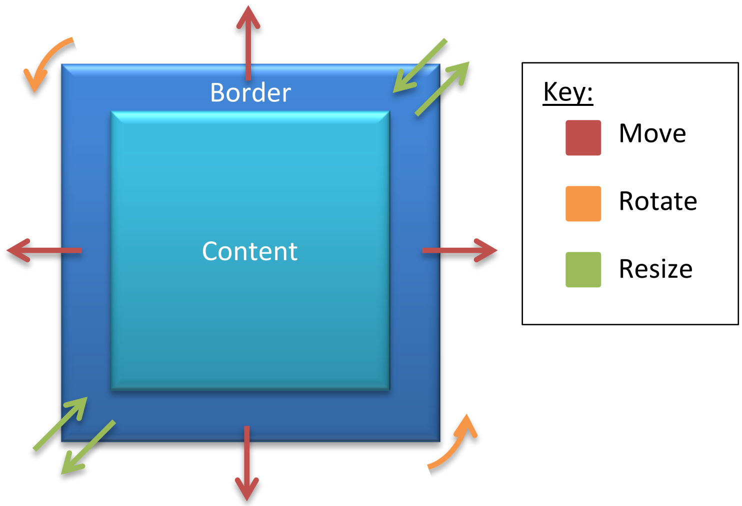


Figure 4.8: General design of a custom GroupAware widget

4.4.5.4 Main Menu Widget

The first of the widgets is the Main Menu widget, which gives the user control of the workspace (see Figure 4.9). There is only one instance of Main Menu widget in the workspace and it is easily distinguishable by its grey border. The content of the widget includes the username and avatar of the user, and a hierarchical menu control that contains the items of menu. Additionally, three visual indicators exist on the widget. The lock icon in the bottom-left corner indicates that the widget cannot be manipulated in the workspace. The image of a face appears in the centre when the user's face is detected by GroupAware. An icon appears in the bottom-right corner when the user has joined the session.

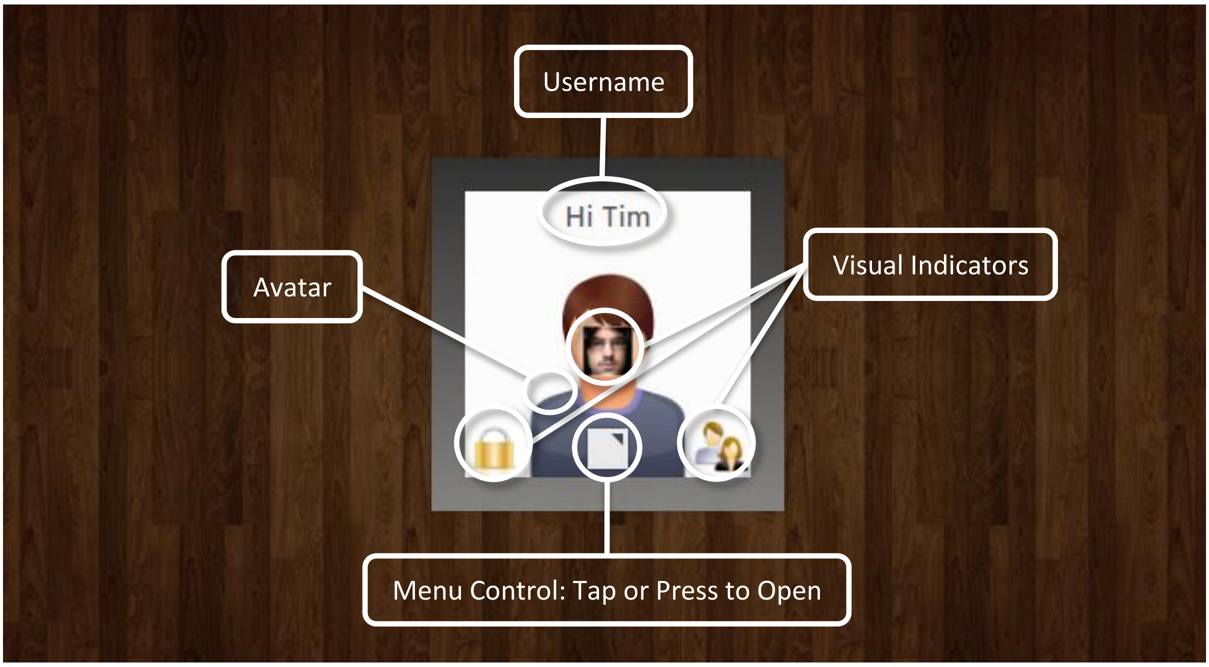
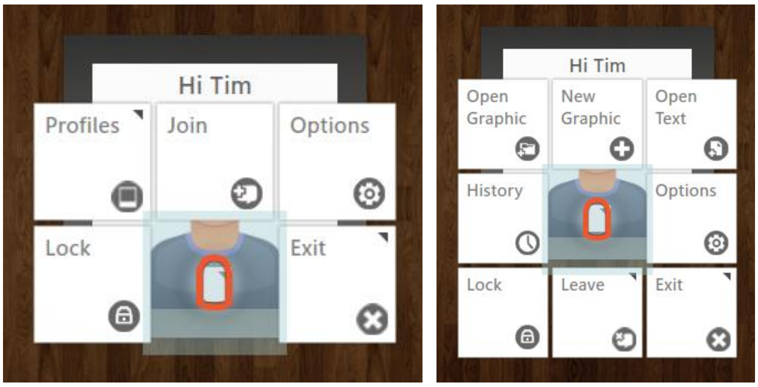


Figure 4.9: UI design of the Main Menu widget



(a) Before session joined *(b) After session joined*

Figure 4.10: Menu items of Main Menu widget

The hierarchical menu control is a standard for the Microsoft Surface and replaces the traditional GUI menu. The items of the menu are displayed when tapping or pressing down on the menu control. Each menu item is assigned to a specific command. The Main Menu widget's menu items that are available depend on whether the user has joined a session or not. Before joining a session, the user can create, change or delete a profile, and join a session (see Figure 4.10a). After joining, the user can open existing text and graphic documents, create new graphic documents, access the history of all user actions and leave the session (see Figure 4.10b). Menu items that are always available are the workspace options, locking the widget and exiting GroupAware.

4.4.5.5 Document Widget

The Document widget is a compact representation of a document that is open in the workspace that shows only the essential information. The colour of the border indicates which user created or loaded the document in the workspace. The content of the widget includes the name of the document and a More Information button at the top of the widget, a picture identifying the type of document in the centre, and a menu control at the bottom of the widget (see Figure 4.11).

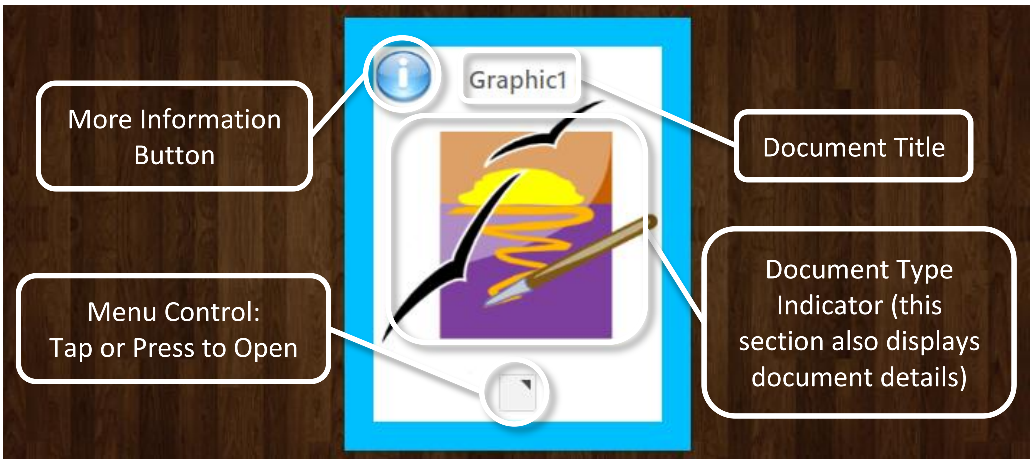
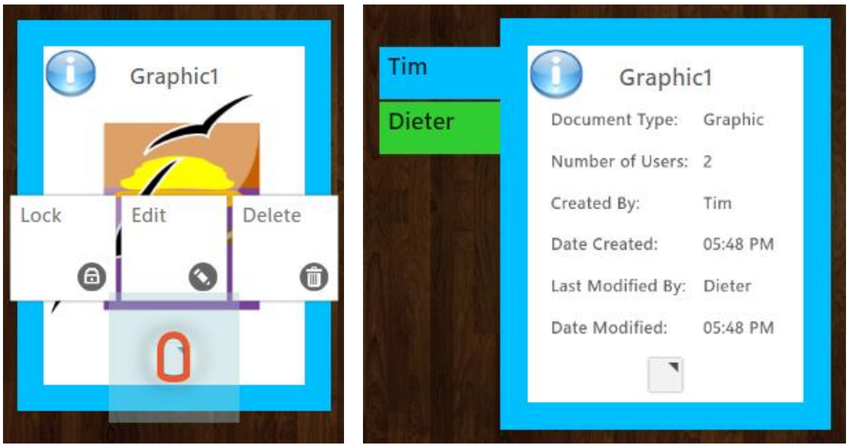


Figure 4.11: UI design of the Document widget (graphic)



(a) Menu items *(b) User labels and document information*

Figure 4.12: Document widget design aspects

The Document widget can represent either a graphic or text document. The design of a graphic Document widget is shown in Figures 4.11 and 4.12. A text document differs only in the picture in the centre of the widget. The menu items allow the user to lock the widget, and edit and delete the document (see Figure 4.12a). Additionally, the document can be opened for editing by double-tapping anywhere on the widget. When users open the document for editing, their user labels are added to top-left of the widget (see Figure 4.12b). To view the details of the document, users must tap the More Information button. The details are displayed in the centre of the widget (see Figure 4.12b).

4.4.5.6 Text Viewer and Drawing Widgets

Two important GroupAware widgets are the Text Viewer widget and the Drawing widget. These widgets are synchronized in real-time and allow simultaneous access and interaction from multiple group members. Both these widgets represent a type of document and thus share similar characteristics and functionality. The colour of the border indicates which user created or loaded the document in the workspace. Both widgets have a top menu bar that has the title of the document in the centre, Undo and Redo buttons on the left and Share, Minimize and Close buttons on the right. Furthermore, they have a bottom toolbar equipped with drawing/annotation tools. They do, of course, also have their differences.

The Text Viewer widget enables the viewing and annotating of text documents such as PDFs. Figure 4.13 shows the UI design of this widget. Annotations can be added and removed by using the annotation tools found in the toolbar. Annotations are given the same colour as the colour of the user that made them and are automatically saved. Changing the page can occur in three ways, namely tapping the page or user labels to the left and right of the document, tapping the

Previous/Next Page buttons in the far left/right of the toolbar, or performing a flick gesture on the page. Zoom In, Zoom Out and Reset Zoom buttons are provided to the right of the toolbar as alternatives to the pinch and spread gesture that can be performed on the page. When users are viewing the text document, their user labels are added to the left or right of the document according to their current page that they are viewing.

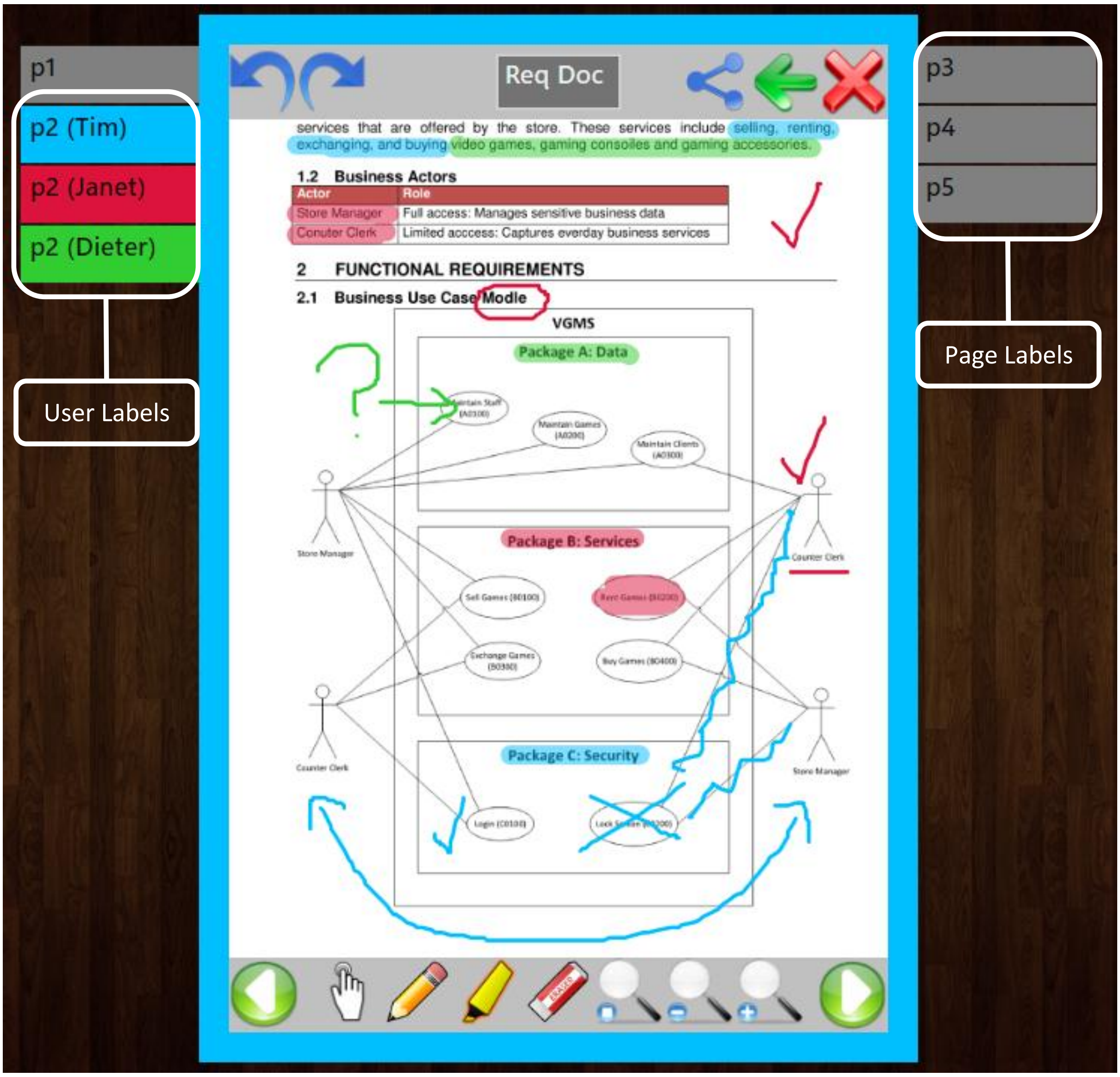


Figure 4.13: UI design of the Text Viewer widget

The Drawing widget enables the creation of collaborative graphic design work artefacts by means of an ink canvas and drawing tools (see Figure 4.14). Drawings can be saved by tapping the Save button in the far left of the menu bar. A save status indicator appears to the right of the graphic's title when any changes occur and disappears when the drawing is saved. Renaming the graphic is possible by tapping the title and inputting a new name using the on-screen keyboard that appears. When users open the graphic document, their user labels are added to the left of the widget.

Figure 4.14: UI design of the Drawing widget with a completed sketch

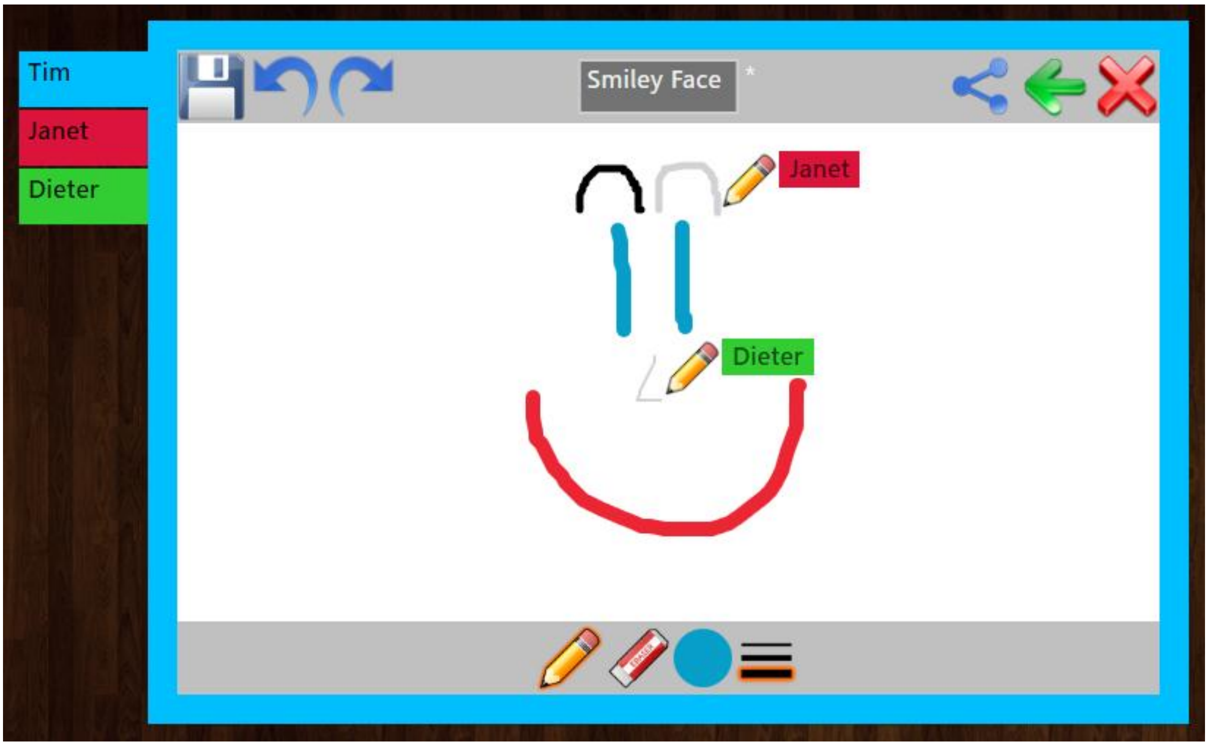


Figure 4.15: Real-time drawing synchronization

Furthermore, users can see their group members performing drawing actions in real-time (see Figure 4.15). This enables users to identify who is doing what at any given time, which helps with group coordination and awareness.

4.4.5.7 History Widget

The History widget allows group members to view all the actions that have occurred in the workspace. Figure 4.16 shows the design of this widget, which is essentially a scrollable list of history items. Each item indicates who performed the action, what the action was, in which document the action occurred (if applicable) and the time the action occurred. By double-tapping on an item that specifies a document, the corresponding document, if it still exists, will be opened in the workspace. The history items can be filtered by user, action, document and time by selecting an item and then selecting the filter criteria.

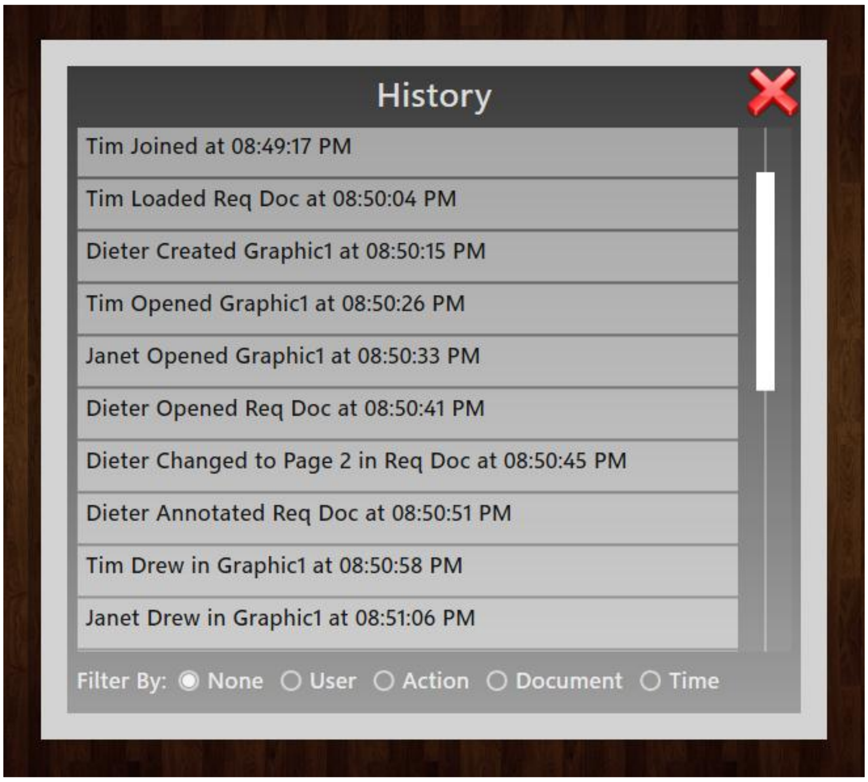


Figure 4.16: UI design of the History widget

Special use of the History widget occurs when a user goes away from the workspace. GroupAware specifically logs all the actions of the other members while the user is away. When the user returns, GroupAware automatically displays the History widget showing all the actions that the user had missed while away.

4.4.5.8 User Widget

The User widget gives more information about a particular user in the workspace and gives control of audio and video conferencing. Figure 4.17 shows the design of this widget. The border colour of the widget indicates which user's information is displayed. Users can adjust the volume of a group member's audio feed by dragging the volume slider left or right. Users can mute a member by tapping the Mute button. Users can enable the video feed of a group member by tapping the Video Call button (see Figure 4.17a). Tapping the same button again disables the video feed and displays the avatar of the user (see Figure 4.17b). Users can see more information about a group member by tapping the More Information button (see Figure 4.17c).

(a) Tim in video call view *(b) Dieter in voice call view* *(c) Janet in details view*

Figure 4.17: UI design of the User widget

4.4.6 Detailed System Architecture

Figure 4.18 illustrates the detailed system architecture of GroupAware. The distributed member interacts with the GroupAware client in two ways, namely Touch and Proxemic interaction. These two forms of interaction make up the Controller of the client. Touch and proxemic interaction input is received from the Touch Device and Proxemic Sensor and this information is sent to the corresponding Application Programing Interface (API). The Touch API makes use of gesture recognition to identify which gesture has been invoked, such as Tap, Drag, Flick, Press, Rotate, Pinch and Spread. The Proxemic API interprets the raw sensor data and updates the appropriate proxemic dimensions, namely Distance and Orientation.

Depending on the action of the user, the Controller manipulates the state of the Model or the presentation of the View. The Model involves the data, which stores the current workspace state and application logic that enables GroupAware's functionality. Any changes to the Model's state are immediately sent to the Server via multiple message-based data channels such as Session, User, Document, Command and Drawing. The Server manages and broadcasts all incoming messages to all connected clients, regardless of the message type and the channel upon which they were received.

Model state changes trigger the View to be updated. The View involves the workspace UI, which is comprised of custom designed widgets such as Main Menu, Document, Drawing and Text Viewer. The View shows the user the updated workspace model or any changes in its presentation.

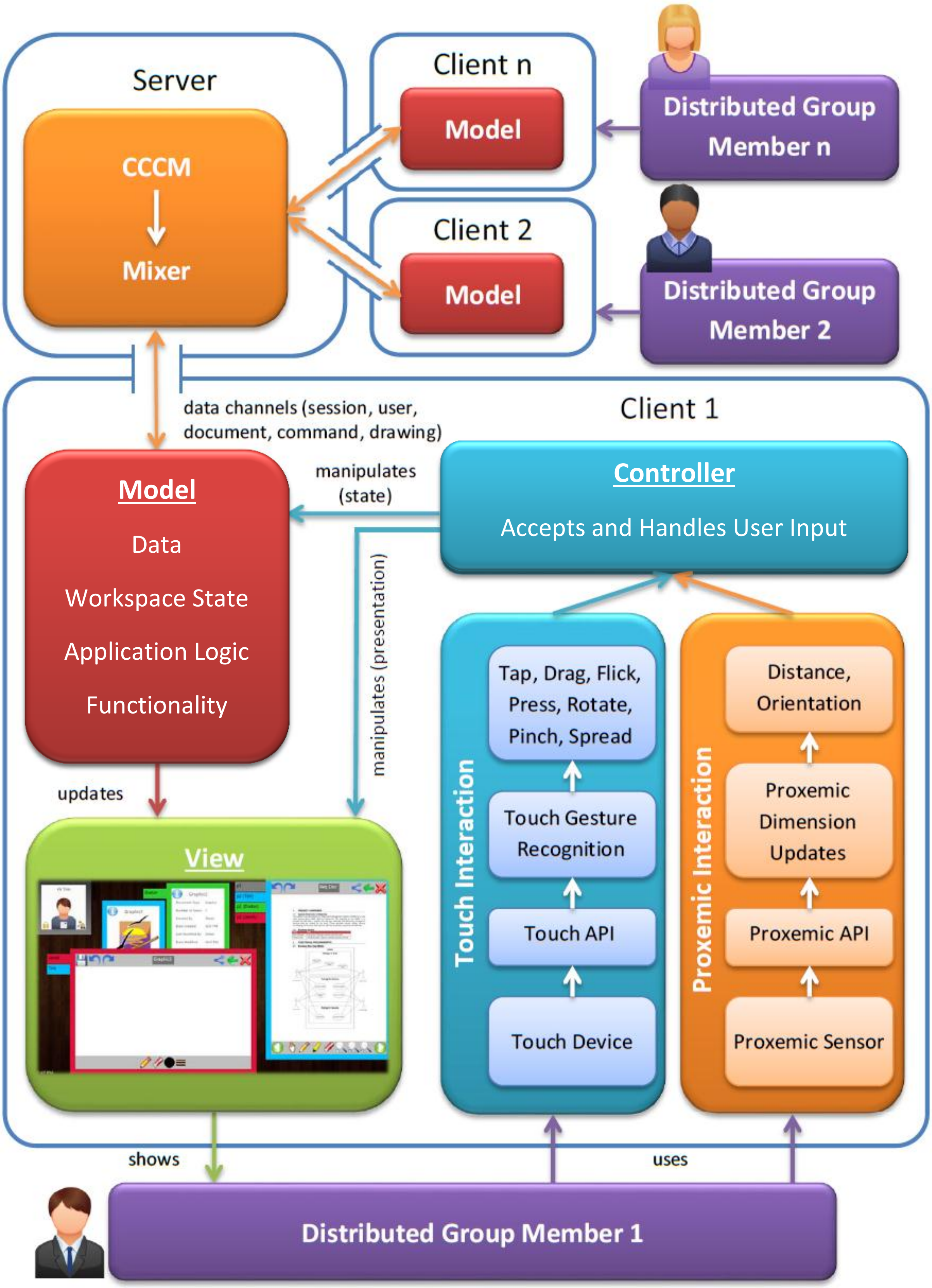


Figure 4.18: Detailed system architecture of GroupAware

4.5 Implementation

The previous section discussed the design of GroupAware as a general shared-workspace groupware system. This section discusses the implementation of the prototype. Firstly, the existing tools and libraries that were used during implementation are discussed (Section 4.5.1). Secondly, the prototype development and integration, and the challenges that were faced are discussed (Section 4.5.2). Finally, the actual functionality available in the GroupAware prototype is discussed (Section 4.5.3).

4.5.1 Implementation Tools

This section identifies the environment in which GroupAware was implemented. The environment consists of both hardware and software components. The hardware required for implementation is that of three multi-touch computing devices capable of handling high-quality graphics and continuous interaction. The computing devices must also include a microphone for audio capture and a webcam for face detection.

The software component requires a platform that is suitable for multi-touch application development. The amount of support available can help improve code quality and functionality and should therefore be considered. The development environment should be multi-touch supportive to allow for the implementation of custom widgets capable of supporting gesture interaction.

4.5.1.1 Hardware

The ASUS EEE Slate EP121 is a high-performance tablet boasting an Intel® Core™ i5 processor and 4GB of memory (see Figure 4.19). The slate runs Windows 7 operating system and has a 12.1" screen with a wide viewing angle. Input options include touch and stylus interaction. Additionally, it incorporates built-in stereo speakers, a 2MP front-facing camera and a digital array microphone.

(a) Digitizer Pen *(b) Leather Portfolio Case*

Figure 4.19: The ASUS EEE Slate

The NMMU Telkom/Centre of Excellence (CoE) multi-touch tabletop was built using a custom designed wooden structure that hosts a 42” LG Plasma television (see Figure 4.20). A multi-touch USB overlay developed by PQ Labs was fitted over the television (Labs, 2014). The multi-touch overlay is capable of recognising 32 simultaneous touch points. The television and overlay were connected to a high-end computer running Microsoft Windows 7. The tabletop display was designed to be used in both a vertical and horizontal setting. Other stop positions were available to allow the display to be positioned between the horizontal and vertical setting.

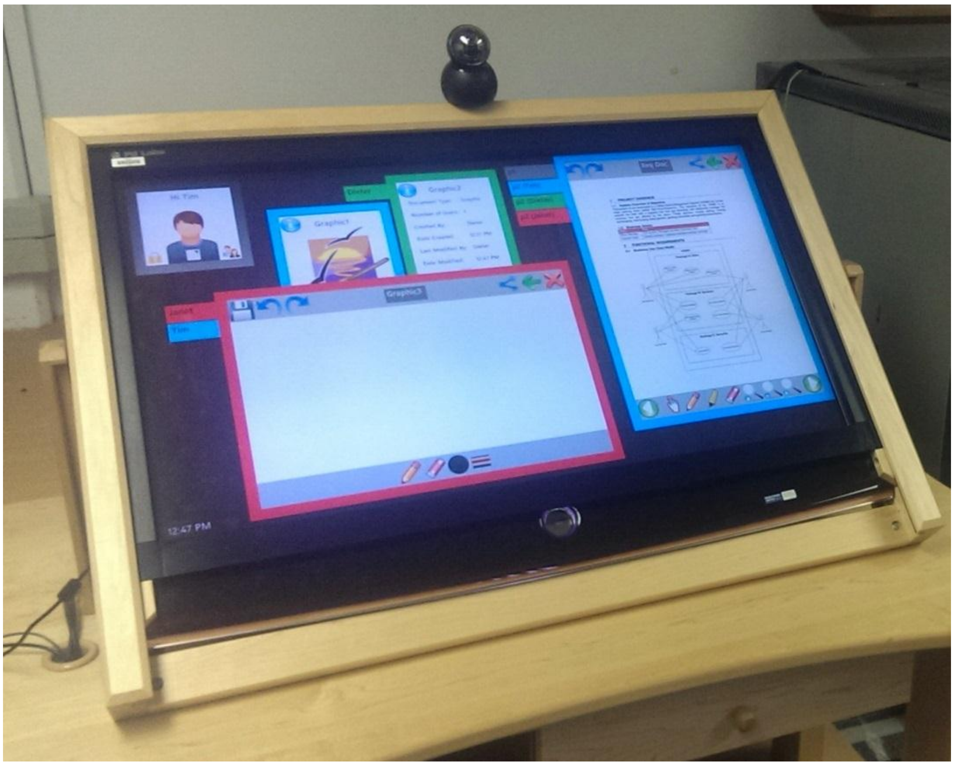


Figure 4.20: NMMU Telkom/CoE multi-touch tabletop in vertical position

4.5.1.2 Programming Language and Environment

C-Sharp (C#) is a simple, but powerful object-oriented programming language that enables rapid application development (Microsoft, 2014d). C# is an innovative language that is designed for building a variety of applications that run on Microsoft’s .NET Framework (Microsoft, 2014a). The .NET Framework class library provides access to many useful, well-designed classes that speed up the development cycle significantly. Furthermore, .NET has a large online following of support, tutorials and Software Development Kits (SDKs).

Microsoft Visual Studio 2010 supports C# with a full-featured code editor, compiler, project templates and a powerful debugger (Microsoft, 2010). Visual Studio is a comprehensive Integrated Development Environment (IDE) that offers the complete .NET Framework and quality developer support. User interface design and programming is also supported extensively by Visual Studio. Windows Presentation Foundation (WPF) is a graphical design component of the .NET framework (Microsoft, 2009). WPF employs Extensible Application Markup Language (XAML), an XML-based language, to define and link various interface elements. The workspace and widgets discussed in Section 4.4.5 were developed using WPF within the Visual Studio environment.

The above reasons provided the justification for GroupAware to be implemented in Visual Studio using WPF for frontend graphics and C# as the backbone. Additionally, personal preference for and experience with these programming tools motivated the choice.

4.5.1.3 Surface 2.0 SDK

The Surface 2.0 SDK is a software development kit for, but not limited to, application development on Microsoft's custom-built tabletop device called Surface (Microsoft, 2011). The SDK is supported by the Visual Studio environment and provides developers with basic controls designed for multi-touch tabletop interaction. These controls can be integrated with WPF to create advanced multi-touch interfaces. The SDK also provides sample solutions written in C# with a WPF interface that demonstrate the key features. A Touch Simulator is included that can simulate finger touches using the mouse. This allows Surface applications to be written and tested on a standard input computer.

4.5.1.4 Thriple

In the design of GroupAware, the Document and User widgets enabled the viewing of more detailed information. To view this information, the widget could be "flipped", which required a 3D flip animation. Thriple is an open source library of three-dimensional (3D) controls and panels available for WPF applications (Smith, 2009). Thriple enabled quick and easy 3D animation for the widgets. Thriple was readily available for WPF and was downloadable with free sample projects, which were developed in C#.

4.5.1.5 Groupware Toolkit for C#

Groupware Toolkit (GT) is a software library for C# and the .NET Framework to simplify the development of real-time distributed groupware and to improve the performance of distributed applications (de Alwis, Gutwin, & Greenberg, 2009). GT handles many of the mundane aspects of network communication while still providing control over communication channels. The third version of GT was used to implement GroupAware's client-server architecture.

4.5.1.6 VLCDotNet

VideoLAN Client (VLC) is a free and open source cross-platform multimedia player and framework that plays multimedia files, DVDs, Audio CDs and various streaming protocols (VideoLAN

Organization, 2014b). VLCDotNet is a software library that provides developers with access to all the audio and video capabilities of the VLC media player (GitHub, 2014). VLCDotNet is available for multiple platforms, one of which is WPF. VLCDotNet was used to provide audio communication among the users of GroupAware.

4.5.1.7 Emgu CV

Open Source Computer Vision (OpenCV) is an image processing library (Itseez, 2014). It was designed especially for computational efficiency with a strong focus on real-time applications. It is written in optimized C/C++ and can take advantage of multi-core processing. Since the .NET Framework is an interpreted environment, it cannot directly call functions written in native C or C++ programming languages. Emgu CV is a cross-platform .Net wrapper for OpenCV that enables functionality of OpenCV to be called from .NET compatible languages such as C# (EmguCV, 2014). The Emgu CV libraries were used for face detection functionality in GroupAware.

4.5.2 Prototypes

GroupAware was developed by breaking the system up into separate components, called prototypes. Each prototype was designed, developed and evaluated separately in order to gain a good understanding about that specific component. The prototypes started off small and were iteratively developed into a fully functional component of GroupAware. The four prototypes of GroupAware are discussed here, namely Workspace, Distribution, Communication and Proxemics. These prototypes were then integrated and further developed into the final GroupAware system to be used for evaluation.

4.5.2.1 Workspace

This section describes the implementation of the workspace prototype. The prototype started as a Visual Studio project template for creating a Surface application, which is provided by the Surface 2.0 SDK. The prototype was iteratively developed on top of the template. The foundation of the workspace was chosen to be a ***ScatterView*** Surface control which enabled the free-form manipulation of widgets. Each widget was then designed, developed and tested in the workspace. Two significant widgets in the workspace are the Drawing widget and Text Viewer widget. The challenges faced with these widgets will be discussed as well as the solution that was implemented to overcome the challenges.

The Drawing widget made use of a ***SurfaceInkCanvas*** control which enables users to draw by simply dragging their fingers over the control. The ***SurfaceInkCanvas*** collects and renders drawing data which is represented by the ***Stroke*** class. The first challenge faced with this control was that the

Stroke class is not serializable, meaning it cannot be explicitly converted into a stream of bytes in order to transmit it over a network. The solution to this challenge was to create a custom serializable class to store the important stroke data that could be sent over the network.

The second challenge faced with the ***SurfaceInkCanvas*** control was that it did not provide direct access to the stroke data while the user was drawing, but only after they had completed the stroke. This made real-time drawing feedthrough a challenge. The solution to this challenge was to use the ***TouchDown***, ***TouchMove*** and ***TouchUp*** events of the control to capture and transmit stroke data to other users while drawing.

The Text Viewer widget enables multiple users to simultaneously view text documents. These documents were required to be read-only, much like PDF documents. The challenge with this widget is that Surface applications cannot explicitly render PDF documents. Attempts were made to implement the viewing of PDF documents by means of libraries such as Adobe PDF Reader and Debenu Quick PDF Library Lite (Adobe Systems, 2014b; Debenu, 2014). These attempts, however, were unsuccessful since the libraries were compatible only with Windows Forms and not WPF.

A possible solution was to use the ***WindowsFormsHost*** control, which allows Windows Forms controls to be used in WPF. This worked for attempts in a pure WPF application, but did not work properly in the workspace prototype, which was a Surface application. The design of the ***WindowsFormsHost*** control implied that it could not be rotated or resized when used as a child of the ***ScatterView*** control. A workaround to this challenge was to first export the PDF document as multiple images, one image per page. These images could then be successfully rendered and viewed in the workspace.

Once the major challenges were resolved, informal testing was done on these widgets by running the prototype in debug-mode with the Drawing and Text Viewer widgets in the workspace. The testing revealed the need to redesign certain aspects of the widgets and add more functionality (see Table 4.3).

Table 4.3: Informal workspace test results

Widget	Design changes	Functionality
Drawing	Make widget smaller	Save status indicator Undo and redo Colour wheel
Text Viewer	Reposition toolbar	Highlighter Flick gesture Jump to page

The Drawing widget was initially designed as a fullscreen canvas to allow for a large drawing surface, but this forced the user to close the Drawing widget in order to do anything else in the workspace such as view a text document. Thus a smaller, windowed, Drawing widget was designed so that users can multitask. Furthermore, additional functionality such as a save status indicator, undo and redo buttons, and a colour wheel was found to be necessary during the informal tests.

A design issue specific to the Text Viewer widget was that the toolbar was covering the text when zooming into the document. Thus, the toolbar was repositioned for better viewing of the document when zoomed in. Additional functionality such as a highlighter, flick gesture to change page and a jump to page function was found to be necessary during the informal tests.

4.5.2.2 Distribution

This section describes the implementation of the prototype that deals with the networking of GroupAware. A client-server distribution architecture was chosen in the design of GroupAware (see Section 4.4.3) and thus C# Socket programming was the implementation path at the beginning of the prototype. The complexity quickly increased as development continued and synchronization and concurrency issues began to arise. Help from existing external libraries was sought out.

A solution was found in GT (de Alwis et al., 2009). GT provided sample code including a simple server, called a Client-Repeater. The server implements a frequently-used pattern for groupware systems, which is to simply repeat all incoming messages to all connected clients. GT also provided all the necessary libraries to create a client application that can send and receive various types of messages on multiple data channels. After a few tweaks of the Client-Repeater and writing a simple client application, the framework of GroupAware's distribution architecture was set up.

To properly test the client and server, the client-side networking code was added to the workspace prototype discussed in the previous section. Testing was done on the shared workspace by running the client application on two devices connected by a network. One of the devices also ran the server application. The Drawing and Text Viewer widgets were added to the shared workspace and two users simultaneously interacted with them. The testing revealed some system errors in the simultaneous drawing of graphic documents and viewing of text documents. The Drawing widget caused the system to crash when one user drew more than one line at a time. The Text Viewer widget caused the system to crash when a user changed the page while another user was annotating that page.

4.5.2.3 Communication

This section describes the implementation of the prototype that deals with the live audio and video feed. The first challenge faced with this prototype was that C# does not have direct access to recording devices on the computer. Thus, existing external libraries had to be sought out in order to implement this functionality. The SDK for the VLC media player, called libVLC media framework, was found (VideoLAN Organization, 2014a). LibVLC can be embedded into an application to allow access to multimedia capabilities such as audio and video streaming. The problem with this SDK was that it was only available for Windows Forms and it did not work in Surface applications. Thus another solution had to be found.

VLCDotNet was found to be an existing VLC media framework that could be successfully used in Surface applications. The framework provided a sample application that was used as an initial prototype. The prototype was tested by running it on two computing devices connected by a network. The test was successful, but the audio and video was noticeably delayed. The network was analysed to see if it was the cause, but no problems were found. The cause of the delay was found to be related to the transcoding process that occurred before sending the audio and video. This took the prototype into a series of iterative tests in which different transcoding options were analysed.

Transcoding is the direct conversion of one encoding to another allowing video and audio to be sent across networks. Transcoding involves two aspects namely, the encapsulation and the coder-decoder (codec). Many different types of encapsulations and codecs exist, but only two of each are discussed further. Two encapsulations were tested, namely ***Raw*** and ***MPEG-TS***. Two audio codecs were tested namely ***MP3*** and ***MPGA***. Two video codecs were tested, namely ***H.264*** and ***MPEG-2***. The results of the various tests are given in Table 4.4.

Table 4.4: Testing the audio and video delay (in seconds) for various transcoding options

	Audio Only		**Audio and Video**	
	MP3	MPGA	H.264 + MP3	MPEG-2 + MP3
Raw	1.4	1.4	N/A	N/A
MPEG-TS	2	2	11	2.4

Raw encapsulation only applied to the audio tests because video cannot be transcoded using ***Raw*** encapsulation. ***Raw*** encapsulation proved to be better than ***MPEG-TS*** by 0.8 seconds. ***MP3*** and ***MPGA*** showed no difference in delays. The ***H.264*** and ***MP3*** codec had a large delay of 11 seconds. This is because ***H.264*** is a high quality video codec. The ***MPEG-2*** and ***MP3*** codec had a much lower delay than the previous codec at 2.4 seconds. The transcoding choices, therefore, were clear: ***Raw/MP3*** for audio only and ***MPEG-TS/MPEG-2 + MP3*** for audio and video.

4.5.2.4 Proxemics

This section describes the implementation of the prototype that deals with the proxemic interaction of GroupAware. Face detection was used to automate support for user availability among group members. Since Emgu CV is a popular Open-Source face detection software library, it was used to support proxemic interaction within GroupAware. Emgu CV was used to develop a simple WPF application with face detection functionality. This application was used to test the capabilities and limitations of the Emgu CV's face detection algorithm. A number of informal tests were conducted using a standard webcam as an input source. The tests subjected the algorithm to different faces, face orientations and background environments.

Results showed that the algorithm was capable of accurately detecting front-facing and profile faces. Furthermore, it could calculate how far away the face was from the webcam. Limitations of the algorithm include the orientation of the face and false positives. Deviations from looking straight at the camera occasionally caused a face not to be detected. When there were many objects in view of the webcam, the algorithm occasionally detected a face when one was not there. Lastly, the algorithm struggled to detect the faces of users with glasses, unconventional hairstyles or darker skin tone.

The limitations of the face detection algorithm highlighted a need for another technique to be used, in addition to face detection, for more accurate identification of user availability. Since the workspace involved touch interaction, user's touches would be considered when determining availability, i.e. if a face was not detected, but a touch was detected, the system would consider the user to be available.

4.5.2.5 Prototype Integration

Once all four prototypes had been designed, developed, tested and updated, they were integrated into one prototype. The prototype was further developed to satisfy all the requirements of the system.

An expert review of the final prototype was conducted to establish any design, implementation or usability issues. Two experts situated in two different venues used the prototype to collaboratively view, discuss and annotate a text document, and also create a graphic document. The results of the expert review are presented in Table 4.5.

There were six design issues found by the experts. It was found that the Main Menu widget required menu icons for easy distinguishing of functions. This design issue was fixed by adding menu icons. GroupAware lacked the ability to control the volume for the audio feed of the user. This issue was fixed by adding a volume control slider to the User widget. An expert wanted to open his document in the other expert's workspace, but was unable to do so. Subsequently, a Share button was added to the Text Viewer and Drawing widgets. While browsing the history log, an expert tried to open a document by tapping on a specific history item. This feature was added. Better colours for the user labels were chosen because the ones in use were difficult to distinguish. Experts did not require a permanent video feed as their attentions were focused on the tasks at hand.

Table 4.5: Results from expert review

Issue	Design	Implementation	Usability
Add menu icons to Main Menu widget.	✓		
Allow volume control for user audio feed.	✓		
Allow users to share a document with others.	✓		
Tapping on a history item opens the document in which the item occurred.	✓		
Choose better user label colours.	✓		
No need for permanent video feed.	✓		
Fix drawing quality loss.		✓	
Fix system crash when a user leaves the session.		✓	
Allow the Active Users Panel to be minimized.			✓
Distinguish between removing a document from the workspace and permanently deleting the document.			✓

Two implementation issues were found by the experts. Visible quality loss was identified during shared drawing tasks. A more effective drawing synchronization technique was developed in order to fix this issue. Just before the session ended, the system crashed. It was found that the crash occurred when an expert attempted to leave the session. This issue was addressed accordingly.

Finally, two usability issues were found by the experts. An expert wished to minimize the Active Users Panel to increase the size of the workspace, but was unable to do so. The panel was made collapsible by means of a button and a flick touch gesture. The experts were unable to distinguish between removing the document from the workspace and permanently deleting it. Separate buttons were added to the Document widget clearly marked for each purpose.

4.5.3 Functionality

This section discusses the implementation of the prototype with regard to the functional requirements of a shared-workspace groupware system, as identified in Section 2.4.2. Table 4.6 presents the functional requirements, whether it was successfully implemented (indicated by a tick) and how it was implemented.

Table 4.6: The implementation of the functional requirements

FR#	Functional Requirement	✓	Implementation
FR1.	Provide access to a shared visual workspace.	✓	When users join a GroupAware session, they have access to a shared visual workspace.
FR2.	Enable access to and annotation of shared information documents.	✓	Users can collaboratively view and annotate text documents using the Text Viewer widget.
FR3.	Enable simultaneous creation, modification and deletion of work artefacts.	✓	Users can simultaneously create, modify and delete graphic documents using the Main Menu, Drawing and Document widgets.
FR4.	Enable manipulation and organization of workspace items.	✓	Documents are represented as widgets in the workspace, which can be manipulated and organized using touch gestures.
FR5.	Enable intentional communication.	✓	Users can speak to each other via live audio feed.
FR6.	Enable consequential communication.	✓	Users can overhear conversations of others. User labels communicate what a user is working on. Face detection communicates user availability.
FR7.	Keep users updated on each other's actions.	✓	User actions such as joining the session, opening documents, and drawing are immediately synchronized in each user's workspace.
FR8.	Keep a log of all the actions of the users.	✓	A history log of all group-related user actions is stored and can be viewed at any time.
FR9.	Enable division of workload amongst users.	✓	Workload can be divided among users at a high level by discussion via the audio feed.
FR10.	Enable identification of the availability of users.	✓	Face detection identifies the availability of other users as available or away.
FR11.	Enable monitoring of a user's actions.	✓	Users can monitor each other's actions in the shared workspace because all group-related user actions are synchronized in real-time.
FR12.	Enable assistance amongst users.	✓	Users can help each other verbally, via the audio feed, or practically, by simultaneously working on a document.

4.6 Conclusions

This chapter presented the design and implementation of GroupAware, an SDCW groupware prototype using NUI interaction techniques. The development methodology and application domain of GroupAware were discussed (Sections 4.2 and 4.3 respectively). Section 4.4 included an in-depth discussion of the design of GroupAware. A system architecture based on the requirements of a share-workspace groupware system was proposed. The architecture comprised an application architecture and a distribution architecture. The design discussion described the architecture, data, functions, user interface and interactions of GroupAware.

Section 4.5 discussed the actual implementation of GroupAware. The manner in which GroupAware was implemented to address the functional requirements was explained. The tools used to implement the prototype were discussed. The four sub-prototypes that made up the final prototype were discussed in terms of the challenges faced and how they were overcome. The actual functionality implemented was described.

The next chapter, Chapter 5, describes the evaluation of GroupAware. The evaluation will provide insight into the effectiveness of the design and implementation of GroupAware and help determine whether NUI interaction techniques can effectively support SDCW.

Chapter 5: Evaluation

5.1 Introduction

In Chapter 4, the design and implementation of a groupware system for SDCW, called GroupAware, was discussed. This chapter presents a detailed description of the evaluation of GroupAware and involves the fifth phase of the DSR methodology, namely Evaluate Artefact. This chapter answers the sixth research question discussed in Chapter 1, namely:

RQ6. How effectively does the developed prototype support SDCW?

Therefore, the purpose of this chapter is to determine if SDCW can be supported using NUI interaction techniques. This will be confirmed by conducting a user study of GroupAware to determine the effectiveness, efficiency, collaborative support and user satisfaction of the prototype. The results of the user study are analysed and presented to validate the design of GroupAware.

This chapter begins by identifying the objectives of the evaluation (Section 5.2). The evaluation methods, found through a review of relevant literature, are then discussed (Section 5.3), followed by a discussion of the user study design (Section 5.4). The results (Section 5.5) and analysis discussion (Section 5.6) thereof are presented, followed by the conclusions of the chapter (Section 5.7).

5.2 Evaluation Objectives

The primary objective of this evaluation is to determine whether the aim of this research was achieved by obtaining and analysing empirical data from a user study. More specifically, it was to determine how effectively the NUI interaction techniques incorporated by the GroupAware prototype supported SDCW, particularly in the area of group coordination. Positive results from the evaluation would suggest that the proposed architecture, design and interaction techniques of GroupAware effectively support SDCW. Secondary objectives were to evaluate the usability and the perceived user satisfaction of the prototype.

5.3 Evaluation Techniques

In this section, relevant literature regarding the evaluation of SDCW groupware and NUI systems will be reviewed in order to find suitable techniques for evaluating a system like GroupAware. GroupAware involves two aspects, namely the synchronous distributed collaborative tasks and the natural user interactions implemented to perform these tasks. Thus, both these aspects must be evaluated using appropriate techniques to understand the full effectiveness of the system.

5.3.1 Related Work

The evaluation of groupware systems is an important issue in the field of CSCW. However, many groupware systems seem to be poorly evaluated (Antunes, Herskovic, Ochoa, & Pino, 2012). One reason for this is that there is no consensus on the methodology to be adopted in order to perform groupware evaluations.

A survey on the most frequently used evaluation methodologies was presented (Pinelle & Gutwin, 2000). The survey was based on the works published at the main CSCW conferences. Interestingly, it was observed that nearly one third of the presented groupware systems were not evaluated in a formal way. In terms of the evaluations that were conducted, no common approaches, methodologies or techniques were found and many of them were dependent mainly on the researchers' interests or the practical adequateness for a specific setting (Greenberg & Buxton, 2008; Inkpen, DiMicco, Scott, & Mandryk, 2004).

Another reason for the poor evaluation of groupware is that evaluating groupware systems is a very difficult undertaking since the success of a collaborative system depends on multiple factors, including the group characteristics and dynamics, the social and organizational context in which it is inserted, and the positive and negative effects of technology on the group's tasks and processes. Ideally, a single groupware evaluation method should cover the individual, group and organizational domains, assessing whether or not the system is successful in the combination of those realms (Antunes et al., 2012).

Eight key areas that can be examined in a groupware system evaluation have been identified (see Figure 5.1). It is suggested that the following eight key questions could be asked as a starting point to the evaluation (Ramage, 1999):

1. Does it work? (functionality)
2. Does it work well enough? (efficacy)
3. Is it workable with? (usability)
4. Does it follow the standards laid down by various bodies? (standards)
5. What does it do to those who work with it? (individual effects)
6. What does it do to their work? (group effects)
7. What does it do to those they work with and for? (organizational effects)
8. What does it do to the world beyond work? (societal effects)

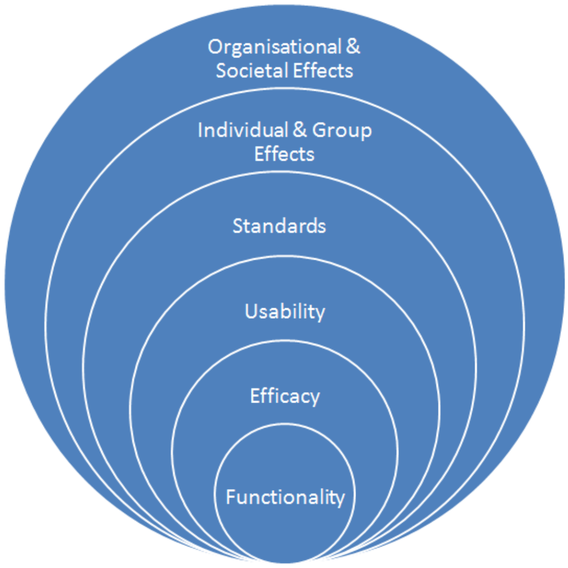


Figure 5.1: The eight key evaluation areas of groupware system evaluation (Ramage, 1999)

The CSCW Lab is a conceptual framework for applying evaluation methodologies in the context of groupware research (Araujo et al., 2004). The framework identified and addressed the major attributes of groupware evaluation, namely Group Context, Usability, Collaboration, and Cultural Impact. Each dimension is a step of the evaluation process, which consists of characterizing the group and work context, measuring usability strengths and weaknesses, determining the collaboration capabilities, and studying the impact of the application over time. The relationships these attributes have with each other are illustrated in Figure 5.2.

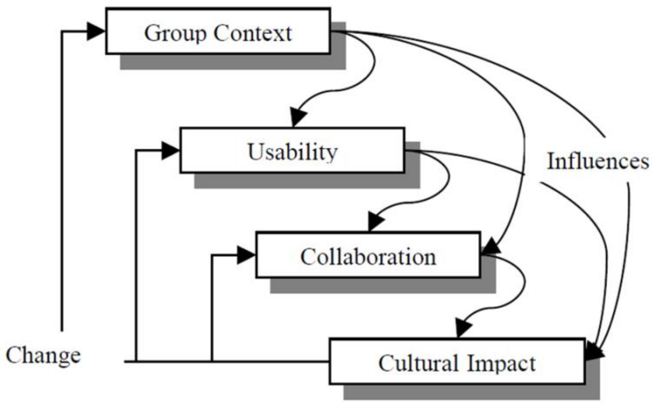


Figure 5.2: The relationships among the four key group evaluation attributes (Araujo et al., 2004)

From the nature and scope of this research, two of the four attributes were deemed appropriate for the evaluation of GroupAware. These attributes are Usability and Collaboration. Group Context and Cultural Impact require gathering a large sample of diverse groups and conducting longitudinal field studies of a system deployed within an organization. Since this research is constrained by time, these two attributes were not suitable for evaluation. Usability is discussed in the next section.

5.3.1.1 Usability

Usability is a quality attribute that assesses how easy a system is to use, learn and remember (Jakob Nielsen, 2012). Additionally, usability of a system determines its performance and user satisfaction. Evaluating groupware usability is a complex task since it involves the evaluation of how a user can use the system as well as group interaction. Usability evaluation techniques for single-user applications have been adapted to the evaluation of groupware systems (Araujo et al., 2004). Gutwin and Greenberg presented the following definition of groupware usability based on the mechanics of collaboration (see Section 2.4.1) (Gutwin & Greenberg, 2000, p. 100):

> *"We define groupware usability as the degree to which a groupware system supports the mechanics of collaboration for a particular set of users and a particular set of tasks".*

The above definition makes the assumption that the system is already usable for a single user. The authors proposed a conceptual framework to evaluate collaboration in groupware that uses a shared workspace. The proposal is to assess the mechanics of collaboration according to three metrics, namely:

- Effectiveness: This considers whether the activity was successfully completed, and the number and severity of errors made during that activity. A usable groupware system will not prevent the mechanics of collaboration from taking place, and will not cause group members to make undue errors in those activities.
- Efficiency: This considers the resources, such as time or effort, required to carry out the activity. A usable groupware system will allow the activities of collaboration to proceed with less time and effort than a system with usability problems.
- Satisfaction: This considers whether the group members are reasonably happy with the processes and outcomes of each of the activities of collaboration.

Well-established evaluation methods that can be used to evaluate the usability of a system according to the above metrics include the following (Baker, Greenberg, & Gutwin, 2001; Gutwin & Greenberg, 2000; Pinelle & Gutwin, 2002):

- Heuristic Evaluation (HE): Several evaluators, i.e. experts, examine an interface and judge its compliance with recognized usability principles called heuristics. On average, 3-5 experts will find 75-80% of all usability problems (Jakob Nielsen, 1994). HE is a quick and easy evaluation method to apply because no end users are required, only a few experts are needed and heuristics are typically well documented and are therefore easy to learn and apply. Issues to consider in HE of groupware are the quality of the heuristics and how well the experts can use them to identify problems in groupware usability.

- Walkthroughs: This involves an inspector, who begins the walkthrough with a realistic and detailed task and user description, a list of correct user actions and the interface to evaluate. The inspector then 'walks through' the task using the interface and considers whether the user would perform the correct action. Advantages of walkthroughs are that they are low cost in terms of time and money since they do not need a working prototype or end users, and they provide contextual information through the task and user descriptions. Issues to consider in groupware walkthroughs are creating and managing multiple user and task descriptions concurrently, specifying appropriate group tasks and dealing with the high degree of variability that can occur among groups.
- Observational user study: This is done by observing how users perform predefined tasks on a system in a laboratory setting. Observation can be done in person or via a video recording. Advantages of observational user studies include evaluating specific aspects of the system through the control of the tasks and the environment and finding many usability issues from various end users. Issues to consider in an observational user study of groupware are finding suitable groups of users and the time to set up and observe multiple locations. Furthermore, the number of users observed is important. More than half of 29 groupware system evaluations involved 30 or less users, which is equivalent to 15 groups of two or 10 groups of three (Mattsson, 2011). Therefore, 10-15 groups of users are acceptable.
- User questionnaires: This involves evaluation through questionnaires that are completed by the system users. This is typically done after observational user testing, with the participants completing a standard or custom questionnaire to express their subjective opinions of the system. An advantage of user questionnaires is that valuable perceptions, comments and opinions can be expressed by users. Issues to consider are the number and quality of the questions and the way the questions are worded, which must increase understanding and decrease bias.

The benefits and limitations of the above evaluation methods must be carefully considered when conducting a usability evaluation on a shared-workspace groupware system. The next groupware evaluation attribute, namely Collaboration, will be discussed in the following section.

5.3.1.2 Collaboration

When designing and constructing a groupware system, the aim is to support collaboration. If it is not possible to determine if this collaboration really occurred, the system loses its relevance and cannot be validated. The level of collaboration in groupware can be determined by a number of qualitative and quantitative measures (see Table 5.1).

Table 5.1: Measures for evaluating the collaborative level of groupware systems (Araujo et al., 2004)

Criteria	Measure	Type
Communication	Number of exchanged messages. Quality of exchanged messages.	Quantitative Qualitative
Contribution	Number of contributions in the creation of a collective product. Quality of contributions in the creation of a collective product.	Quantitative Qualitative
Coordination and awareness	Engagement in a process definition. Tasks' performance. Mutual understanding of group members and tasks.	Qualitative Quantitative Qualitative

Measuring collaboration also involves subjective metrics, which involves people's opinions and feelings. People generally have a good sense of whether collaboration is occurring among the group that they are in. By introducing instruments such as questionnaires or direct observation, evaluators can be aware of participants' satisfaction and have an indication about the collaboration that occurs among group members (Araujo et al., 2004).

Mattsson (2011) identified and tested the validity of six quantitative measures that can be used to evaluate groupware systems. These measures include Speaking, Monitoring, Looking at task description, Sketching, Erasing and Laughing. Each measure's operationalization and unit of measurement is presented in Table 5.2.

Table 5.2: Measures for evaluating groupware systems and how they can be captured (Mattsson, 2011)

Measure	Operationalization	Unit of Measurement
Speaking	Start: Participant says something to the other participant. End: Directly after utterance.	Frequency count
Monitoring	Start: Participant is watching the other participants or the workspace without doing anything else. End: Participant starts doing something.	Seconds
Looking at task description	Start: Participant is reading the task list. End: Participant starts doing something else.	Seconds
Sketching	Start: Participant is sketching something in the workspace. End: Participant starts doing something else.	Seconds or Frequency count
Erasing	Start: Participant engages with the erasing tool. End: Participant disengages with the erasing tool.	Seconds or Frequency count
Laughing	Start: Participant laughs or giggles. End: Directly after laughter.	Frequency count

Speaking, sketching and erasing are considered important measures for evaluating collaboration. Speaking involves discussion, negotiation, sense- and decision-making, or task delegation, which are all forms of verbal collaborative contribution and thus are indicators of collaboration. In a shared-sketch environment, sketching and erasing are forms of practical contribution to the joint task and are thus indicators of collaboration. It goes without saying that the verbal and practical contributions are only counted when they are performed in the context of work.

Monitoring, looking at task description and laughter are considered to be optional measures for evaluating collaboration because they are highly dependent on the type of tasks given to the participants and the personalities of the participants. These measures could, however, highlight system usability issues and user satisfaction. Excessive monitoring and looking at the task list could reveal a participant’s confusion and laughing could reveal a good sense of group synergy or participant’s enjoyment of the system.

5.3.2 Extant Systems’ Review

A study of three related groupware systems was conducted in order to identify and analyse the evaluation techniques that were used. One co-located and two distributed groupware systems were reviewed. All three systems involved synchronous collaboration within a shared workspace. In this review, a brief description of each system is given followed by identifying the number and grouping of participants, as well as the metrics, measures, data collection methods and instruments that were used in the evaluation.

Ardaiz et al. presented an application framework aimed to support real-time collaboration within a RMCE (Ardaiz et al., 2010). Eight groups of two participants each were used in a preliminary evaluation of their system. The system was evaluated in terms of how effective and efficient the system was in providing support for real-time collaboration in a distributed environment. The test moderators were present to observe the users’ interaction during the sessions, which were followed by a post-test discussion for user feedback. The pairs of participants were given two tasks including playing a letter writing game, which involved drawing letters in the workspace, and collaboratively creating a storyboard, which involved selecting and ordering given objects to create a visual story.

Tang et al. presented the design of a system called Three’s Company that aimed to support three-way distributed collaboration over a shared visual workspace (Tang et al., 2010). Four groups of three participants each were used to evaluate qualitative measures of activity such as examining users’ interactions (collaboration), tracking their problems (effectiveness), and identifying interesting patterns of use (user satisfaction). The data collection methods and instruments used for the evaluation were system logging of users' interactions with the workspace, video recording of each session for observation, and post-test questionnaires for user feedback. Participant groups were given two tasks including collaboratively recreating logos from a set of variously-shaped tiles and organizing text-based items into agreed-upon groupings. Both tasks required simultaneous selecting and organizing of objects in the workspace.

Ditta et al. presented CollaGIM, a co-located collaborative system supporting Group Information Management (GIM) on a multi-touch tabletop (Ditta et al., 2013). Fifteen groups of two participants each were used to evaluate the support for GIM that CollaGIM provides in terms of effectiveness, efficiency, collaboration, and user satisfaction. The data collection methods and instruments used for the evaluation were system logging of user interaction, pre-test and post-test questionnaires, and observation by the test moderator. The pairs of participants were given the primary task of collaboratively creating a magazine article comprising vibrant images and expressive text. To achieve the tasks, groups were required to search, select, share, sort and organize text and graphic documents in the shared workspace.

5.3.3 Extant Systems' Review Conclusions

From the review of relevant literature, it can be seen that the evaluation of groupware systems is both important and difficult. Evaluation involves many different aspects and thus the objectives of the evaluation must be well-defined. The choice of appropriate participant groups, metrics and data collection methods and instruments is crucial to a successful evaluation and these choices have to be based on the objectives of the evaluation.

Table 5.3: A summary of the extant groupware systems' evaluation techniques

System	Participants	Metrics	Methods & Instruments	Tasks
RMCE	4 groups of 2 (n = 8)	Usability* Collaboration	User observation* Post-test discussion*	Writing/drawing Selecting and organizing
Three's Company	4 groups of 3 (n = 12)	Usability Collaboration	System logging User observation (Video) Post-test questionnaire	Selecting and organizing
CollaGIM	15 groups of 2 (n = 30)	Usability Collaboration	System logging User observation Post-test questionnaire	Searching, selecting, sharing, sorting, and organizing

**Since it was a preliminary user study, this was not explicitly stated, but it can be gathered from the results and discussions thereof*

Table 5.3 provides a summary of the evaluation techniques used in the reviewed groupware systems. The extant systems review found that a small to medium number of groups of two or three participants were used to evaluate the systems based on usability, which included effectiveness, efficiency and user satisfaction, and collaboration. Two of the three systems included mechanisms to log the users' interaction with the system during the experiment. All systems involved user observation either directly or via video recording and post-test user feedback either via questionnaires or discussions. Tasks common to all system evaluations were the selecting and organizing of visual objects in the shared workspace.

The design choices for this evaluation were based on the objectives of the evaluation (see Section 5.2), the reviewed literature, the common themes in related system evaluations and the application domain of this research (see Section 4.3). Usability and Collaboration were chosen as attributes for the evaluation of GroupAware based on the evaluation objectives. Effectiveness, efficiency and user satisfaction were chosen to be the usability metrics because these are core usability metrics. Verbal contributions (speaking) and practical contributions (sketching and erasing) were chosen to be the collaboration metrics. The perceptions of the participants of the three CSCW characteristics, namely Communication, Coordination and Information Sharing were used to measure the collaboration level, with particular emphasis of Coordination. The collaborative metrics were chosen because they were found to be important in determining the level of collaboration. Additionally, laughter was chosen as a measure of collaboration because *"laughter is used as a signal for being part of a group — it signals acceptance and positive interactions with others"* (Camazine et al., 2003, p. 18).

An observation user study and user questionnaires were chosen as the usability methods for the evaluation of GroupAware based on its advantages and popularity in extant groupware system evaluations. A sample size of thirty participants was chosen to be satisfactory based on literature. A group size of three was chosen because it best represented the dynamics of a small group while still being practically manageable. Peer-to-peer collaboration (groups of two) was considered inadequate in representing the dynamics of a small group, and larger groups (four or more) would exponentially increase the complexity of the study (Tang et al., 2010).

The primary task given to participants was based on the application domain of this research, which was chosen to replicate the real-world use of the groupware system. The task was to collaboratively design aspects of a software system including a logo and initial UI designs. To accomplish this task, groups were required to create graphic documents and make use of a shared information source in the form of a text document. Thus, multiple documents need to be created, selected and organized in a shared visual workspace, which corresponds to the collaborative tasks found in the review of extant system evaluations.

5.4 User Study

This section describes the experiment that was conducted on the design artefact, GroupAware. The design of the experiment is first outlined (Section 5.4.1). The various elements of the experiment are then discussed, such as data collection (Section 5.4.2), participants (Section 5.4.5) and the tasks required to be performed (Section 5.4.6). This section concludes with the details of the whole procedure (Section 5.4.9). Ethics approval for this experiment was sought from the NMMU Ethics Clearance Committee, who granted approval with the reference number H14-SCI-CSS-004 (see Appendix A).

5.4.1 Experimental Design

A user study was conducted to evaluate GroupAware. Ten groups of three participants each used the prototype to complete a set of given tasks within the context of a defined scenario (see Appendix B). The experiment took place in three different laboratories within the NMMU CS Department. Each participant was given a designated venue, which was equipped with the necessary hardware and software. Before the experiment began, a brief demonstration of how to use GroupAware was given. Participants were each given a task list and taken to their designated venues. Participants were not given any specific instructions about how to work together or delegate tasks, but they were encouraged to work together as a group. During the experiment, data was collected by the system, which logged users' actions, and video cameras, which recorded each participant's interaction with the system and one another. After the experiment, participants were asked to complete a post-test questionnaire.

5.4.2 Data Collection Methods and Instruments

Various data were collected for research purposes using appropriate and valid data collection techniques throughout the experiment. The data collection methods and instruments that were used in the experiment are given below:

- System Logs: GroupAware implemented ways to collect data. System logging was used to log the actions of the participants in order to capture performance metrics, such as task completion and time-on-task, as well as collaborative metrics, such as number of practical contributions. Practical contributions were logged when users performed actions within the workspace that contributed towards the task at hand, such as creating and loading documents, and sketching and erasing.
- User Observation: A video camera was used to capture audio and video footage of each participant during the group's collaborative work session. This footage was thoroughly

analysed by playing back the footage of all three members in a synchronized manner to capture the required data and study the interaction between the participants and the system. The raw data was digitally stored and interesting observations were noted.

- Subjective Feedback: A biographical questionnaire was used to capture the biographical and previous experience information of the participants. A post-test questionnaire was used to capture user satisfaction metrics such as the users' perceived usability of the system and their perceived support of the SDCW that the system provides. A section for open-ended comments was also provided, which encouraged participants to give qualitative feedback.

Table 5.4: A summary of the data collection methods and instruments

Method	Instrument(s)
System Logs	Log file Saved stroke data
User Observation	Video recording
Subjective Feedback	Biographical questionnaire Post-test questionnaire

Table 5.4 presents a summary of the data collection methods used in the evaluation of GroupAware and the corresponding instruments. These methods and instruments enable raw data to be captured based on the defined metrics, which are discussed in the next section.

5.4.3 Metrics

The data collected from the experiments using the methods and instruments described in the previous section allowed for the usability and level of collaboration attributes of GroupAware to be evaluated. The usability metrics that were used are effectiveness, efficiency and user satisfaction. The collaborative metrics that were used are verbal and practical contributions, user perceptions and laughter. These metrics and how they were recorded are presented below and a summary is given in Table 5.5:

- Effectiveness: The task completion rate, i.e. the proportion of tasks successfully completed, was recorded for each task per group.
- Efficiency: The time-on-task, i.e. the time in minutes taken to complete a task, was recorded for each task per group.
- User Satisfaction: The post-test questionnaire was used to capture subjective satisfaction data via a 7-point Likert scale rating system and an open-ended comments section.
- Verbal Contributions: The video recording of each group member was analysed to record the number of times that a member spoke during the collaborative session.

- Practical Contributions: The system logged all interactions that each group member had with the system. The log was analysed to record the number of times each member performed task-related contributions such as drawing and erasing.
- User Perceptions: The post-test questionnaire was used to capture user perceptions about collaboration via a 7-point Likert scale rating system and an open-ended comments section.
- Laughter: The video recording of each group member was analysed to record the number of times that a member laughed during the collaborative session.

Table 5.5: A summary of the metrics and how they were used

Attribute	Metric	Operationalization	Unit of Measurement
Usability	Effectiveness	Task success	Yes/No
	Efficiency	Time-on-task	Minutes
	User satisfaction	Post-test questionnaire	7-point Likert scale Open-ended comments
Collaboration	Verbal contributions	User observation	Frequency count
	Practical contributions	System logging	Frequency count
	User perceptions	Post-test questionnaire	7-point Likert scale Open-ended comments
	Laughter	User observation	Frequency count

5.4.4 Location

The experiment took place in the laboratories of the CS Department at NNMU. Three laboratories were used, namely the CoE, the ACI and the Usability laboratory. Each location had a workstation setup for a participant. The workstation included a device to run the GroupAware prototype and a video camera to capture audio and video for later observation.

5.4.5 Participants

Three important factors were considered in the selection of participants for the experiment, namely participant profile, the group composition and the required sample size of participants. These factors with regard to the user study are discussed below:

- Profile: The participant profile for the user study is intended to approximate the typical end users of GroupAware. Firstly, shared-workspace groupware systems are designed for groups that are working together on a common goal. Thus, colleagues and classmates, who are often also friends, tasked with a group project or assignment would be a candidate group to use GroupAware. Furthermore, distributed groupware systems are usually used by intermediate to expert computer users because they most likely use computers in their everyday life and enjoy using technology to aid their work. Novice computer users would rather meet together in the same place because they feel they would be more comfortable and productive. Lastly, GroupAware is more likely to be used by the younger generation, for example students, because they have grown up with technology and are familiar with technology as a part of life. Participants were required to be computer literate and be over the age of 18. Therefore, a sample from the NMMU CS Department was well justified.

- Group Composition: Since GroupAware is a collaborative system, it was necessary to have participants working in groups during the evaluation. Each group consisted of three participants. Each group's members were expected to have a pre-existing relationship or at least be briefly acquainted prior to the study, as this best represents the typical use of a groupware system.
- Sample Size: Ten groups of three members each participated in the study, adding up to a total of thirty participants. User satisfaction was assessed from individual participant responses (n=30), while performance and collaboration were assessed per group (n=10).

5.4.6 Tasks

The participant groups were provided with a task list (see Appendix B) that described the scenario as well as a list of tasks that had to be completed in order to achieve the overall goal of the scenario. Each group member was provided with an almost identical copy of the task list, but they were instructed that they needed to work together to achieve the list of tasks. That is, the group needs to decide during the session who had which role in the group and who was to perform which tasks.

The only difference among the users' task lists was the placement of what was called the Secret Task. This task was given its name because the group members were not explicitly aware of when other members would have to perform the task. The task involved the user moving away from the device for a short time and was intended to represent the real-life event of a user suddenly being distracted or having to leave with the intention of returning, for example getting a phone call or going to the restroom.

5.4.7 Questionnaires

Two questionnaires were given to each of the participants, namely a biographical questionnaire (see Appendix C) and a post-test questionnaire (see Appendix D). The biographical questionnaire was based on the Common Industry Format (CIF) for usability testing and was used to collect demographic and experience details for each participant (Bevan, 1999). This questionnaire was completed during the Secret Task.

The post-test questionnaire included three sections, namely Section A, B and C. Section A included the standard Post Study System Usability Questionnaire (PSSUQ) which consists of 19 statements paired with 7-point Likert scales, which used the antonyms "Strongly Disagree" and "Strongly Agree" at either end of the scale (Lewis, 2002). Section B included additional statements paired with 7-point Likert scales to rate the system in terms of the perceived support for collaboration. Section C included five open-ended questions regarding the best and worst aspects of the system, general comments about the system usability and collaboration, and suggested improvements for the system. The questionnaire was given to participants at the end of the experiment.

5.4.8 Statistics

The data collected from each session was captured into a Microsoft Excel worksheet for analysis. The data was stored per individual. The NMMU Unit for Statistical Consultation provided advice and assistance with deriving the statistics. Excel's built-in functionality was used to produce descriptive statistics, such as the mean, median and mode, to provide sound analysis of the data and to observe general patterns and trends.

5.4.9 Experimental Procedure

Participants were found in NMMU's CS Department. The 3rd year CS students were identified as the best candidates for participation since they were already in groups for their final year project. To collect the sample, the test moderator gave a brief presentation during a 3rd year CS lecture (see Appendix E). Additionally, an email was sent out to all 3rd year CS students asking if they would like to participate. Only three 3rd year groups responded, which was not enough for statistical analysis. The invitation to participate was extended to the CS post-graduate students, who formed groups for the study.

Prior to the day of the user study, participants had received a written document via email. This document outlined the background, rationale and scenario of the user study (see Appendix F). The participants were advised to read through this document carefully in order to understand what the system was and how they would use it. They were also advised to read it through as a group so that a common understanding among the group could be established.

On the day of the study of a particular group, an introductory meeting was held in the CoE laboratory. A consent form was required to be signed by each participant in order to verify that participation was voluntary (see Appendix G). A verbal presentation was then given to the group explaining the basic proceedings of the study and demonstrating the system functionality. After the presentation, participants were assigned to a venue, given their task list and taken to their respective venues. The moderator then started the system and the video recording in each venue. Participants were given a few minutes to familiarize themselves with the system, thereafter the study commenced. Once the group had completed their tasks, each participant was asked to complete a post-test questionnaire. Once all the participants had completed their questionnaires, everyone met back in the CoE laboratory where they were thanked and dismissed.

5.5 Results

This section presents and discusses the results of the user study of GroupAware described in the previous section. The section begins with a discussion of the demographics of participants and the groups used within the study (Section 5.5.1). The usability (Section 5.5.2) and collaboration (Section 5.5.3) results are then presented and discussed in terms of the metrics used.

5.5.1 Demographics

Figure 5.3 presents the participant demographic results obtained from the biographical questionnaire. These results were based on a sample size of thirty participants. Figure 5.3 indicates that the majority of the participants were part of the 21-29 years age group which was expected due to the participants being selected within a university department. Only one participant was aged between 30 and 39 years. The majority of the participants were male with only 10% being female. 83% of the participants had a right dominant hand. Participants were asked whether they suffered from any form of colour blindness and the results showed that only one of the participants had such a condition. Observation revealed that this had no significant effect on the results. The results showed that 50% of the participants had a certified postgraduate degree, 30% had a matriculation certificate and 20% had a bachelor's degree.

All participants were computer literate with 57% of the participants having been exposed to computers for more than ten years, 30% between six and nine years, and only 13% for three to five years. 67% of participants considered themselves experts in the use of computers, and the rest felt that they were intermediate computer users. 93% of participants reported that they had used a multi-touch device and therefore the majority was familiar with touch interfaces. More specifically, 44% of those who had used a multi-touch device had previous experience with tablets, 36% with smartphones, and only 20% with multi-touch tabletops. Just over half the participants reported that they had used a groupware system. More specifically, 55% of those who had used a groupware system had previous experience with a synchronous distributed document editor called Google Docs, 15% with a synchronous co-located groupware system called CollaGIM, and 30% with various asynchronous distributed groupware systems.

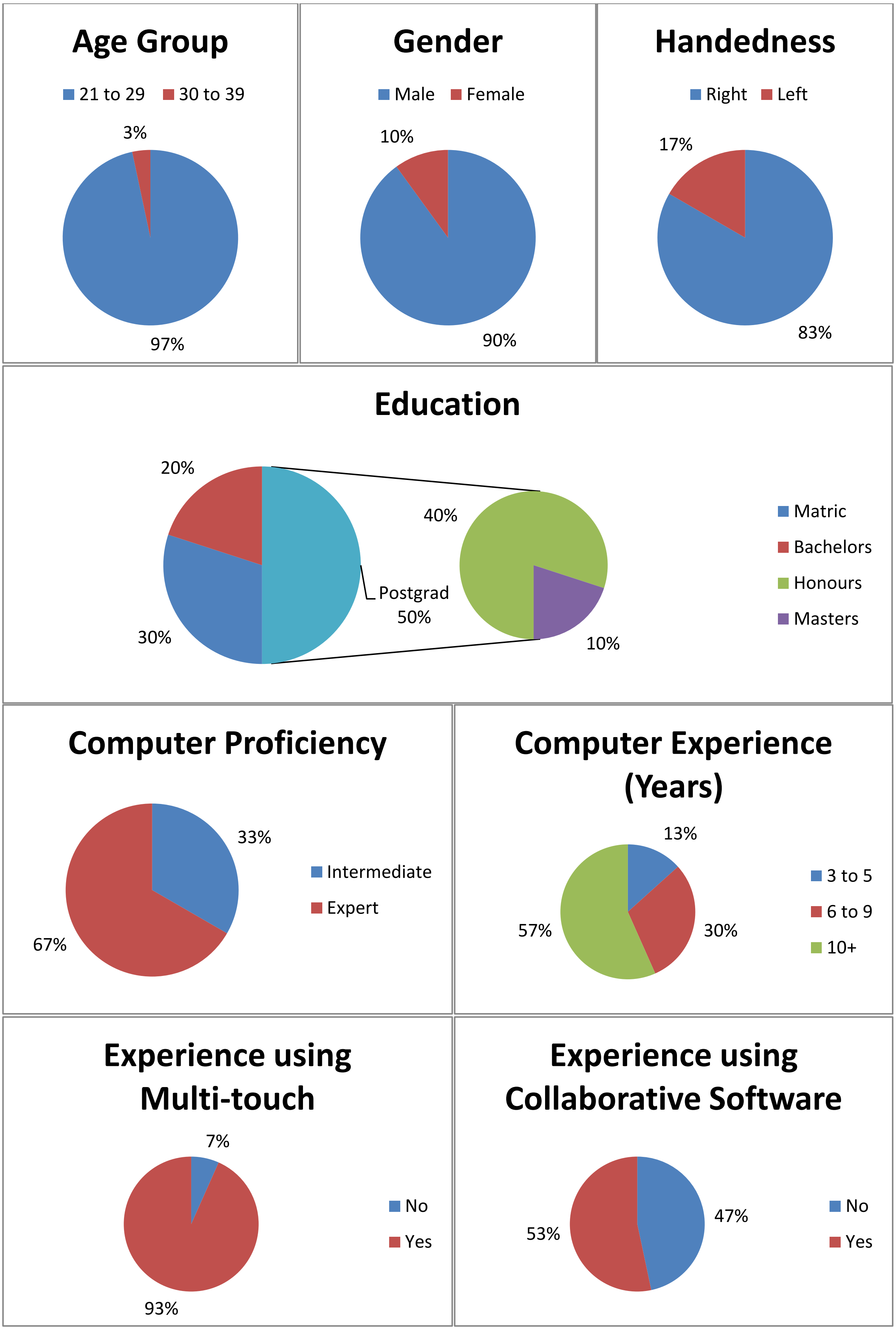


Figure 5.3: Participant demographics (n=30)

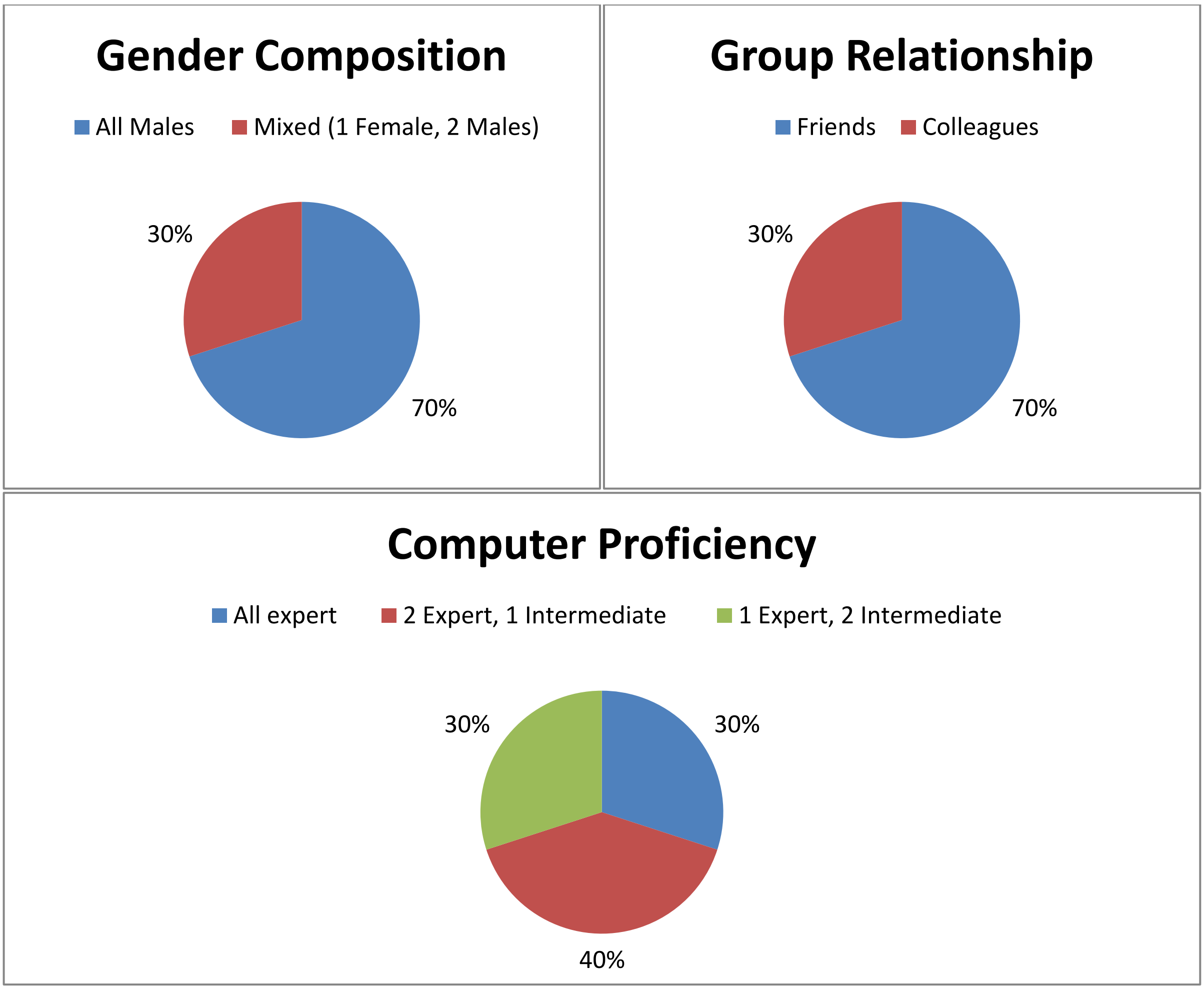


Figure 5.4: Group demographics (n=10)

Figure 5.4 presents the group demographic information based on three different criteria, namely gender composition, group relationship and computer proficiency. These results were based on a sample size of ten groups. It was found that 70% of the groups consisted of all male participants and only 30% had a mixture of both genders, consisting of one female and two males. All groups had the same education level, with 70% of groups being good friends, while 30% were acquainted only as colleagues. Lastly, computer expertise amongst groups was quite evenly split with 30% of groups consisting of all expert users, 30% consisting of one expert and two intermediate users, and 40% consisting of two experts and one intermediate user. The next section presents the usability results.

5.5.2 Usability Results

This section presents and discusses the usability results of GroupAware. Firstly, the results of the participants' performance are given in terms of the effectiveness and the efficiency (Section 5.5.2.1). The user satisfaction results are then given (Section 5.5.2.2). The section concludes with the general usability comments of the participants (Section 5.5.2.3).

5.5.2.1 Performance Results

Quantitative metrics were considered for the performance of the groups using GroupAware, namely effectiveness and efficiency. Task completion rate was used to measure effectiveness and time-on-task was used to measure efficiency. The performance results are presented in the following figures.

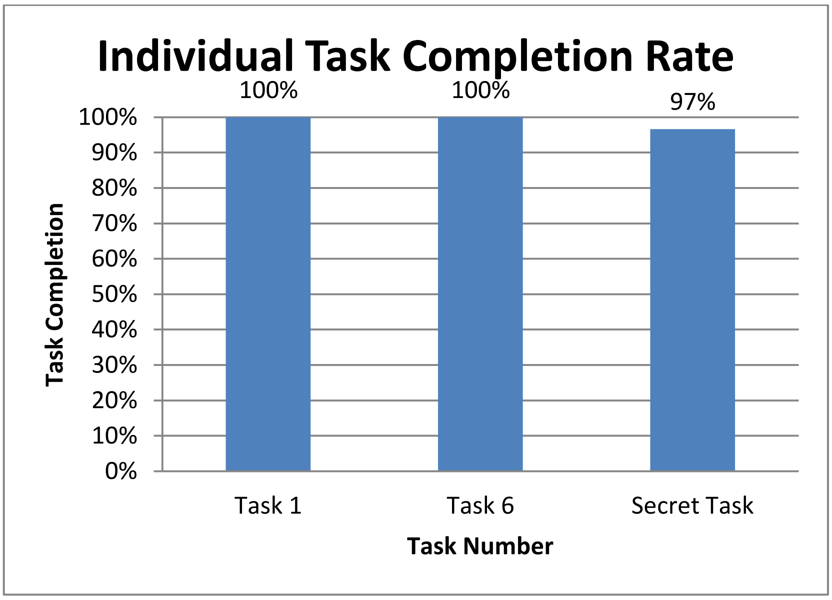


Figure 5.5: Total completion rate for participant per individual task (n=30)

Individual tasks were the tasks that the participants completed on their own. The sample size is therefore thirty. Figure 5.5 shows that the individual tasks were achieved successfully with Tasks 1 and 6 both having a 100% success rate and the Secret Task having a 97% success rate. This was due to one participant not being able to complete the Secret Task.

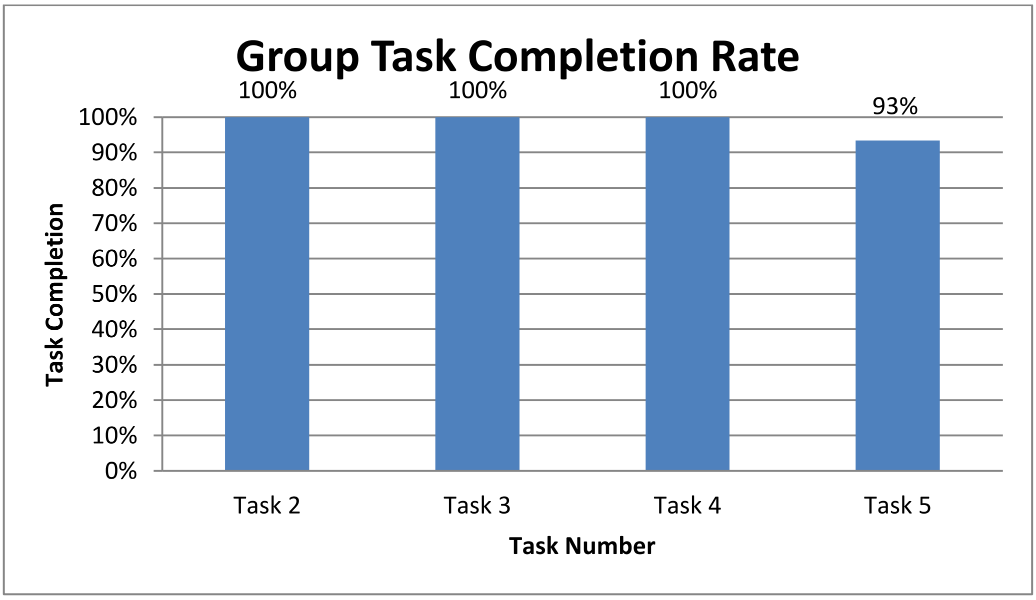


Figure 5.6: Total completion rate for groups per group task (n=10)

Group tasks were the main tasks in the user study performed together by the group members. The sample size for these results is therefore ten. Figure 5.6 shows that Tasks 2, 3 and 4 had a 100% success rate. Task 5 had a 93% completion rate, which was due to one group partially completing the task.

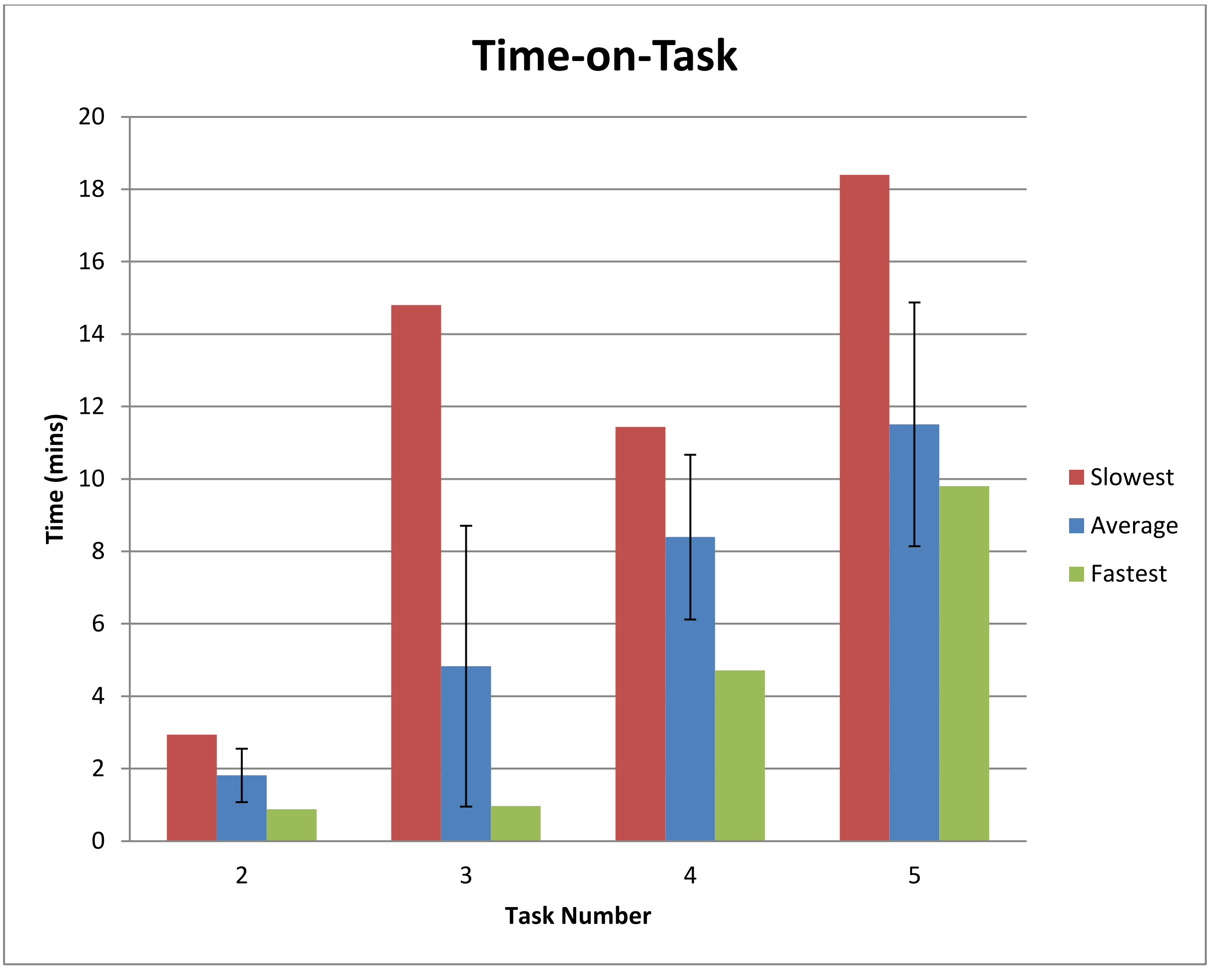


Figure 5.7: Time-on-task for the slowest and fastest groups as well as the average (n=10)

The tasks were designed to increase in difficulty as the user study progressed. This can be seen in Figure 5.7, in which the average task time increased from the first group task to the last. Task 3 has a very large range and consequently a large standard deviation. The reason for this can be linked to the nature of the task, which was to design a logo. Some groups came up with a quick and simple design and therefore finished the task quickly. Other groups either designed intricate logos or were simply enjoying drawing their logo.

Task 5 involved creating three consistent User Interface designs. The slowest group for Task 5 was a lot slower than the majority of the groups. The reason for this was identified through observation of the group during the task. They were found to be having disagreements about the look of their UI designs and took time to reach consensus.

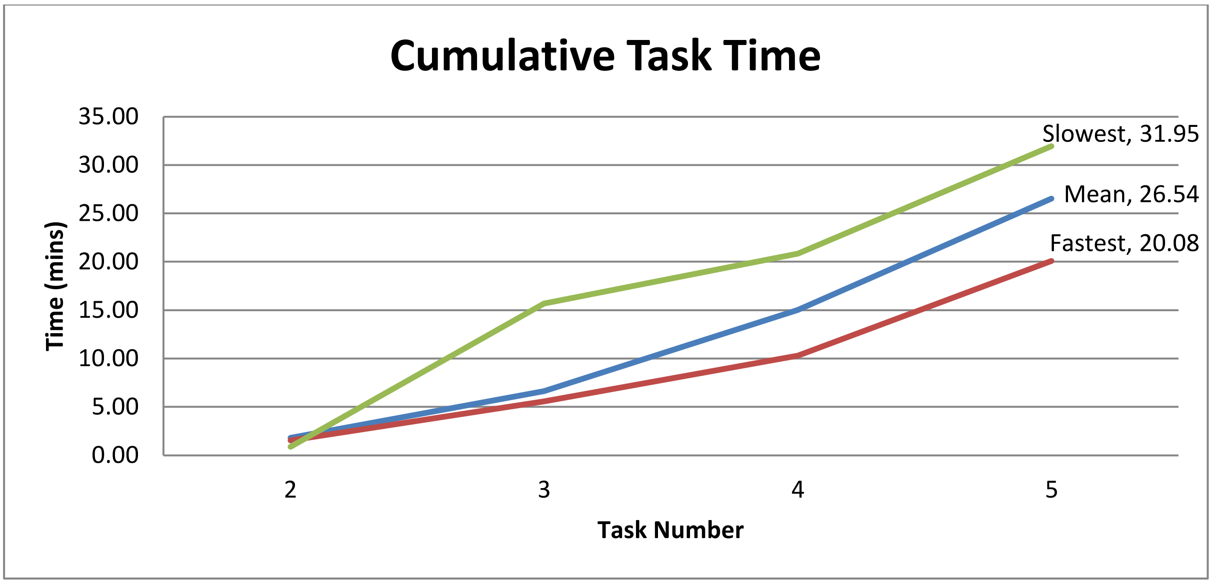


Figure 5.8: Cumulative team task times for fastest, slowest and average (n=10)

Figure 5.8 presents a line graph showing the cumulative task time of the fastest group, slowest group and the average. The fastest group took twenty minutes to complete all the group tasks, the slowest group took approximately thirty two minutes and the average across all groups was twenty six and a half minutes. Considering the scenario, the tasks that were performed and the fact that only 25% of the participants had previously used synchronous distributed groupware, it can be said that the overall task times were very reasonable.

5.5.2.2 User Satisfaction Results

This section presents and discusses the user satisfaction results based on Section A of the post-test questionnaire, namely Overall Satisfaction, which comprises the PSSUQ. Cronbach's alpha was used to test the reliability of the PSSUQ. Cronbach's alpha is the coefficient of internal consistency, that is, how well a set of items measures a single aspect. It is typically used as an estimate of the reliability of a psychometric test, such as a questionnaire, for a sample of participants (Cronbach, 1951). A guideline for interpreting internal consistency using Cronbach's alpha can be seen in Table 5.6.

Table 5.6: Interpretation intervals for Cronbach's alphas

Cronbach's Alpha Interval	Interpretation
$\alpha \geq 0.8$	Excellent consistency
$0.7 \leq \alpha \leq 0.79$	Good consistency
$0.6 \leq \alpha \leq 0.69$	Acceptable consistency
$0.5 \leq \alpha \leq 0.59$	Poor consistency
$\alpha < 0.5$	Unacceptable consistency

Cronbach's alpha coefficient was calculated for the whole PSSUQ as well as the three sub-sections found within the PSSUQ, namely System Usability, Information Quality and Interface Quality. Table 5.7 shows the alphas and the corresponding interpretation for each section.

Table 5.7: Cronbach's alpha coefficients and interpretation for each PSSUQ section (n=30)

Section	Cronbach's Alpha - Interpretation
A. PSSUQ	0.86 – Excellent consistency
A1. System Usability	0.83 – Excellent consistency
A2. Information Quality	0.86 – Excellent consistency
A3. Interface Quality	0.68 – Acceptable consistency

It can be seen from Table 5.7 that the internal consistency of the PSSUQ is excellent. The System Usability and Information Quality sections both have excellent consistency. The consistency of the Interface Quality section is acceptable. Thus, all the results from the PSSUQ can be considered as reliable.

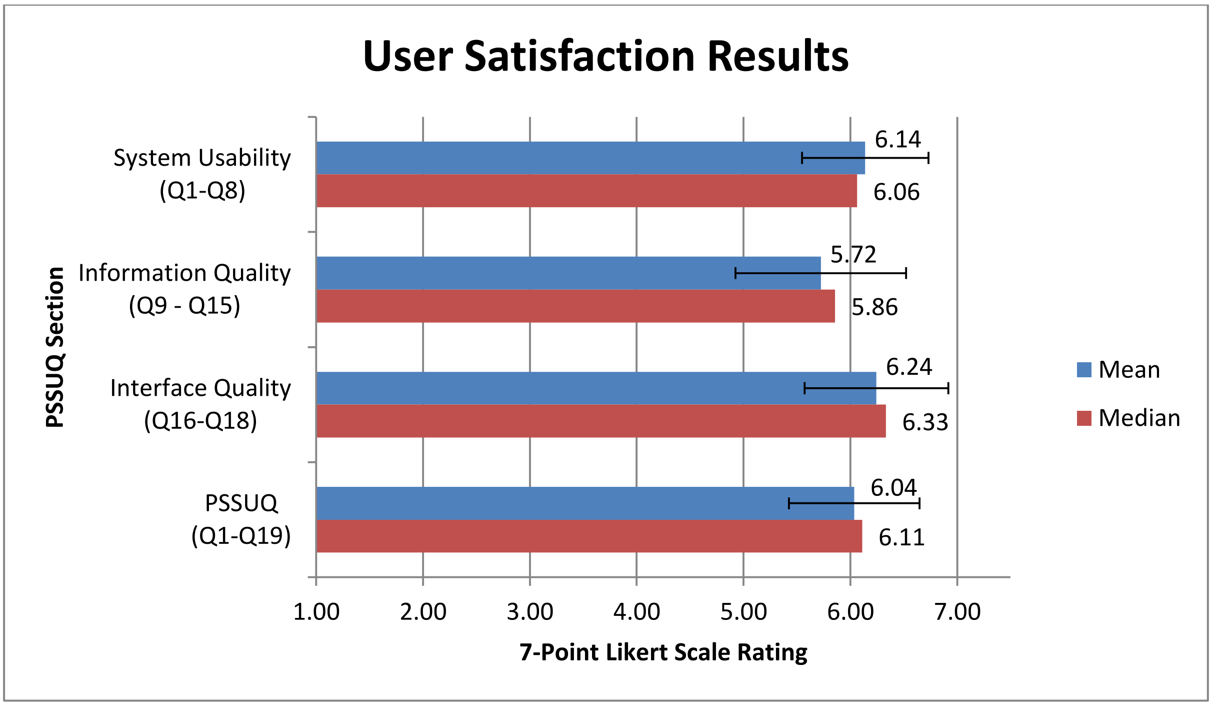


Figure 5.9: User satisfaction results per PSSUQ section (n=30)

The user satisfaction results of the PSSUQ and its sections are presented in Figure 5.9. The sample size for these and subsequent results is thirty since each member completed the post-test questionnaire. The mean, median and standard deviation are shown for each range of questions. These results are very encouraging as three of the four sections have averages above 6 with reasonably low standard deviations of around 0.5.

Figure 5.9 reveals that the Information Quality section has an average rating of less than 6. Although still relativity high at 5.72, this does show room for improvement for the quality of information. This matter is further addressed in Figure 5.11.

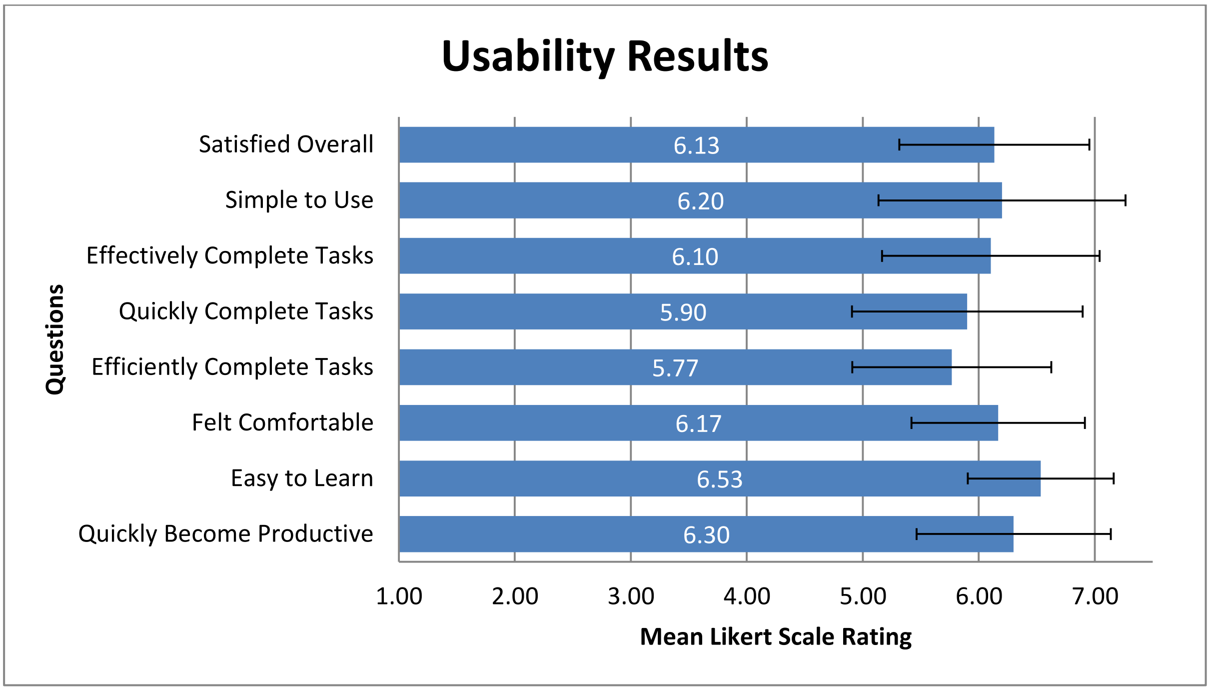


Figure 5.10: Mean 7-Point Likert scale ratings for the Usability section (n=30)

Figure 5.10 shows the average ratings and standard deviations for the first eight questions of the PSSUQ section, which correspond to the usability of the system. Questions 4 and 5 were rated the lowest on average and both have quite a high standard deviation. Both these questions have to do with efficiency and are quoted below:

Q4: *"I was able to complete the tasks and scenarios quickly using this system."*

Q5: *"I was able to efficiently complete the tasks and scenarios using this system."*

Although the ratings of 5.9 and 5.77 respectively are not a major cause for concern, they do highlight the fact that users might have felt that they were not as efficient as they could have been. Users may have been thinking that they could complete the tasks quicker if their group was co-located or they were working alone. Alternatively, they might have needed to get more comfortable with the system before feeling that they could work efficiently. Future user studies could be done to confirm these notions.

The two highest rated questions, with average ratings of 6.53 and 6.3 respectively, were the following:

Q7: *"It was easy to learn to use this system."*

Q8: *"I believe I could become productive quickly using this system."*

It was expected that the system be easy to learn since it incorporates NUI interaction techniques. Question 7 confirms that users perceived that the system's learnability is good. Users also perceived that they became productive quickly using the system, again indicating that they quickly learnt how to use it and could achieve the given tasks.

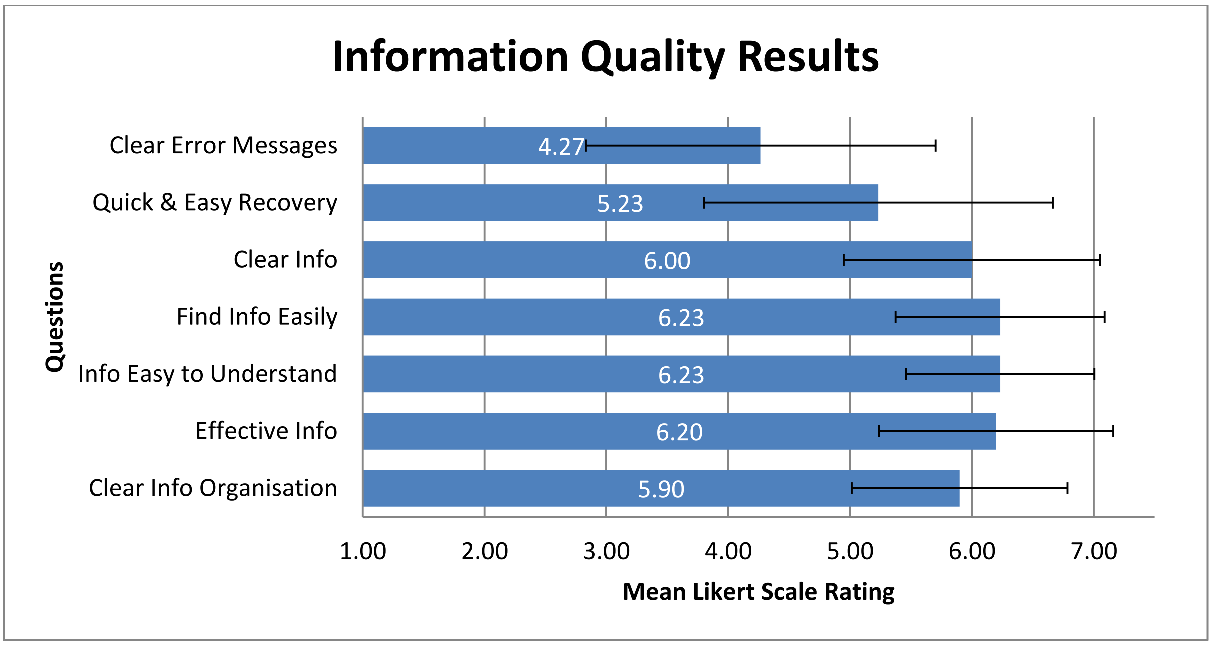


Figure 5.11: Mean 7-Point Likert scale ratings for the Information Quality section (n=30)

Figure 5.11 shows the average ratings and standard deviations for the next seven questions of the PSSUQ section, which correspond to the quality of information provided by the system. Questions 9 and 10 were rated the lowest on average and both have a high standard deviation. Both these questions have to do with errors and are quoted below:

Q9: *"The system gave error messages that clearly told me how to fix problems."*

Q10: *"Whenever I made a mistake using the system, I could recover easily and quickly."*

These were the two lowest rated questions in the whole questionnaire and thus raise concerns as to why this is so. After careful analysis it was found that users perceived system crashes as their mistakes, although this was not the case. Since the system is a prototype of a multi-user application, bugs were naturally present in the code, and the system would occasionally crash, requiring a system restart. Users seemed to perceive a crash as their fault and thus responded accordingly when these two questions were asked.

The two highest rated questions, both having an average rating of 6.23, were the following:

Q12: *"It was easy to find the information I needed."*

Q13: *"The information provided for the system was easy to understand."*

The results of the two questions above show that participants perceived the information to be easy to find and understand. This suggests that the information was structured in a simple and logical manner. The information structure enabled participants to easily navigate the system and comprehend the information.

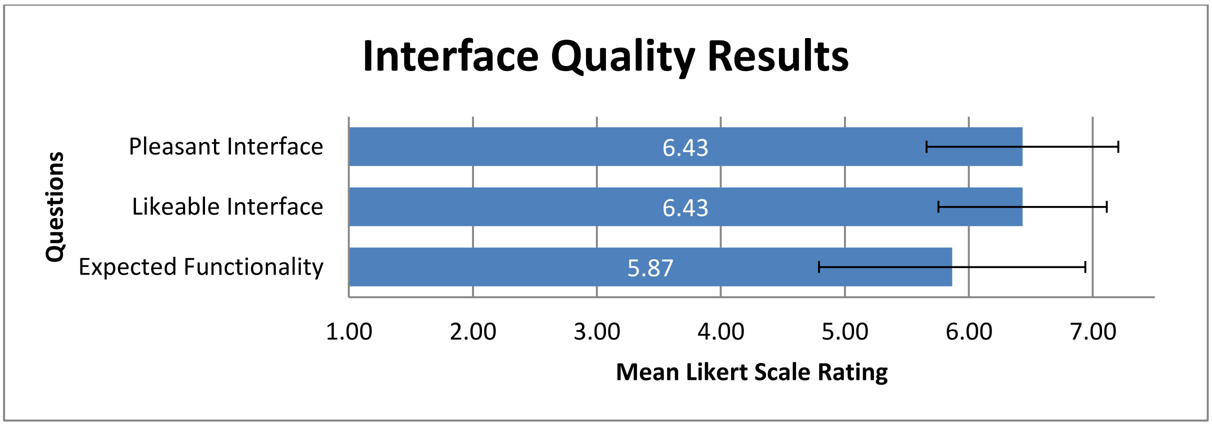


Figure 5.12: Mean 7-Point Likert scale ratings for the Interface Quality section (n=30)

Figure 5.12 shows the average ratings and standard deviations for the next three questions of the PSSUQ section, which correspond to the quality of the system's interface. Question 18 was rated the lowest on average in this section and had a high standard deviation. It reads as follows:

Q18: *"This system has all the functions and capabilities I expect it to have."*

The high variation in this question may be due to the fact that many users did not know what functions and capabilities to expect. Another reason may be that users wanted more functionality from the prototype. Nevertheless, the rating is still relatively high at 5.87.

The two remaining questions in the section were rated high, both having an average rating of 6.43. The high average ratings of these two questions indicate that the implemented interface was perceived to be user-friendly and aesthetically pleasing. The questions are quoted below:

Q16: *"The interface of this system was pleasant."*

Q17: *"I liked using the interface of this system."*

5.5.2.3 General Usability Comments

This section presents and discusses the results of the participants' general comments regarding the usability of the system found in Section C of the post-test questionnaire. These results were used to verify the quantitative results presented in the previous section. This verification process is known as methods triangulation (Patton, 1999). The Discussion Section (see Section 5.6) discusses the harmony between the results.

The comments were first divided into positive and negative comments and then grouped into common themes. The number of users that expressed an opinion regarding each theme was recorded. A condition that was used during thematic grouping was that users could express their opinion about a theme only once. For example, if a participant indicated that the system was "easy to use" in the best aspects section and then went on to say the system was "simple to use" in the general usability section, the frequency of the "easy to use" theme would be increased by one. Furthermore, the video footage from the user study was thoroughly analysed for interesting activity and usage patterns in order to compare with and find further reasons for the responses from the questionnaire. This is another case of methods triangulation.

Table 5.8: Positive Usability themes and frequencies identified in user comments (n=30)

Category	Theme	Freq	%	Example
Performance	Easy to use	21	70%	"System [is] very easy to use". "Simple to use".
	Effective/ Productive	16	53%	"A very effective tool to collaborate with group members". "Can make work productive".
	Learnability/Intuitive	13	43%	"The system is very intuitive". "Easy to learn and remember".
User Satisfaction	Fun/System is good	19	63%	"Very fun to use". "It was excellent".
	Good UI	10	33%	"Navigation was easy as items are laid out logically". "UI is clear and icons can be easily identified".

From Table 5.8 it is clear that the general usability comments from participants were overwhelmingly positive. The strongest theme was the system's ease of use with 70% of participants saying that the system was "simple to use". 63% of participants said that it was "very fun to use" the system. Just over half the participants said that they could work effectively and be productive when using the system. Slightly fewer than half the participants appreciated the intuitiveness and learnability of the system. 33% of participants commented positively on the user interface with one participant in particular saying that "navigation was easy as items are laid out logically".

Participants were also asked to report on any negative aspects about the system's usability. Table 5.9 shows the thematic grouping of those comments. The highest negative comment, which a third of participants reported, had to do with the system crashing. A crash required a system restart, which affected all participants in the group and took a few minutes. Thus, groups that encountered a crash felt it was the worst part of the evaluation. This, however, is not a cause for concern since it is a prototype evaluation for research purposes.

Table 5.9: Negative Usability themes and frequencies identified in user comments (n=30)

Category	Theme	Freq	%	Example
Functions	System crashes	10	33%	"The system crashed".
	Drawing tools	2	7%	"Drawing was tough, need to add more aids" .
Interface	Clutter	7	23%	"Screen can become crowded at times when many documents are open".
	Feedback	4	13%	"The save button did not give me feedback".
	Lack of WIMP patterns	1	3%	"Didn't follow your traditional way of doing certain things".
Hardware	Touch accuracy	4	13%	"Hardware limitations make it difficult to accurately draw/select".
	Touch sensitivity	2	7%	"Touch sensitivity is too high".

The next highest negative aspect was workspace clutter, which was reported by 23% of participants. Concerning clutter, one participant said that the "screen can become crowded at times when many documents are open". These comments came mostly from those participants working on the tablet device. In Task 5, each participant had to create their own graphic document. From the video footage it was observed the workspace became crowded when participants using the tablet opened all three graphics.

13% of participants said that system feedback and touch accuracy was the most negative aspect. Some of the participants that experienced clutter also experienced lack of system feedback. This was because system feedback was displayed on the Main Menu widget and when the workspace was cluttered, the Main Menu widget could not be seen. 7% of participants commented negatively on the drawing tools, accuracy of user detection and touch sensitivity. Two participants felt that "drawing was tough" and better drawing tools were needed in order to create better designs. The last negative comment, made by only one participant, was the lack of traditional WIMP interaction patterns. This was in some ways a positive comment since moving away from traditional WIMP interaction and towards NUI interaction techniques was a priority of this research.

5.5.3 Collaboration Results

This section presents and discusses the collaboration results of GroupAware. Firstly, the results of the verbal and practical contributions are given (Section 5.5.3.1). The user perceptions results are then given (Section 5.5.3.2), followed by the general collaboration comments of the participants (Section 5.5.3.3). The section concludes with the laughter results (Section 5.5.3.4).

5.5.3.1 Verbal and Practical Contributions Results

Two types of contributions to shared work were considered, namely verbal and practical contributions. Verbal contributions are equivalent to any form of speaking relating to the shared task. Practical contributions involve task actions which include creating or loading a document and sketching, annotating or erasing in a document. The following results show the proportions of verbal and practical contributions made by each group member for each group as well as the average proportions across all groups. The verbal and practical contributions that were captured are first discussed separately, followed by discussion of the combination of both types of contribution.

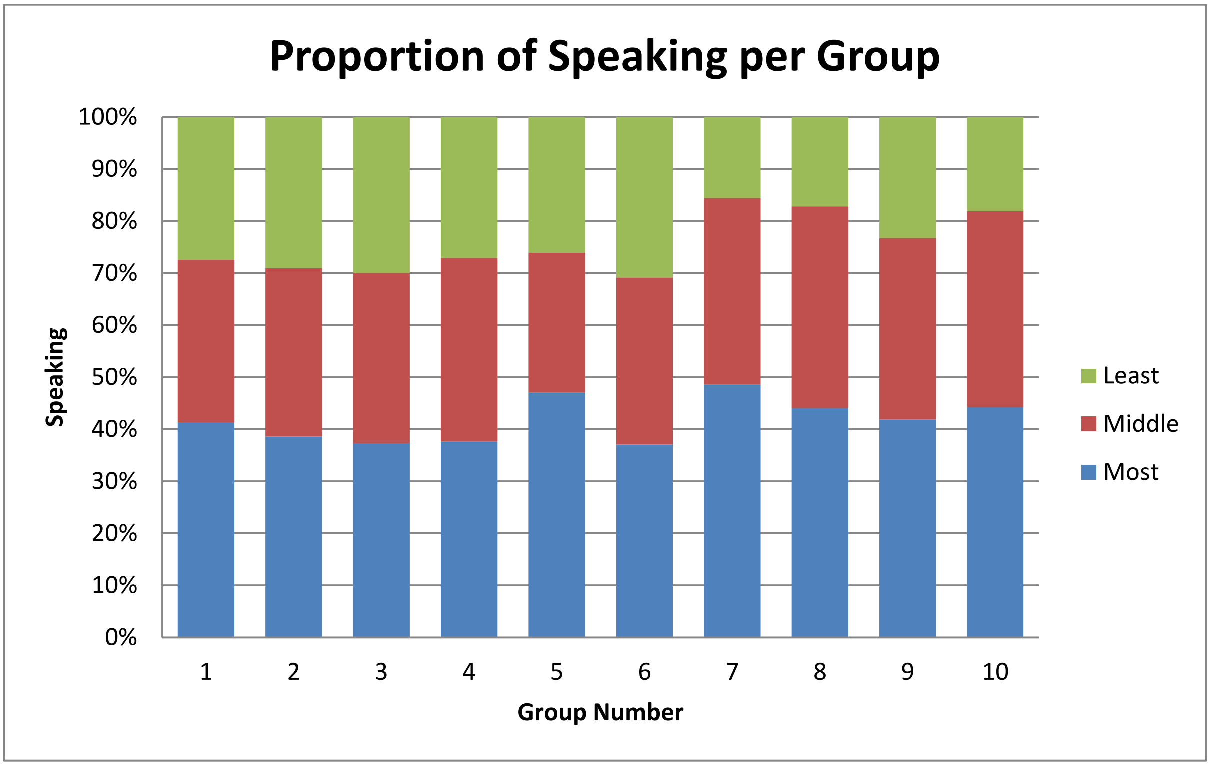


Figure 5.13: Proportion of speaking per group (n=10)

Figure 5.13 shows the verbal contributions, or speaking, by each group member for each group. Speaking was grouped and ordered by the member with the most occurrences to the member with the least occurrences and represented as a 100% stacked column graph. Each column represents 100% of a group’s contribution. The bottom section of the bar represents the highest contributing group member and the top section represents the lowest contributing member.

If all three sections are equal, it follows that all three members contributed equally, i.e. a three-way 33% or 33-33-33 split. In an ideal world, collaboration among peers would result in a 33-33-33 split since all members should be able to have their say. In reality, however, this split is not guaranteed owning to human factors such as personality, social skills and technological aptitude. Therefore, an equal amount of speaking per member indicates a high level of collaboration, but an unequal split does not necessarily indicate a low level of collaboration.

From Figure 5.13 it can be seen that there is a relatively even proportion of verbal contributions among group members per group. Groups 5 and 7 show that one participant in the group dominated the speaking. The video footage showed that Group 5 and 7 had natural leaders in the group that guided the group through the tasks. Group 7 also showed that one participant in particular did not speak very much. By observation, it seemed personality could account for this.

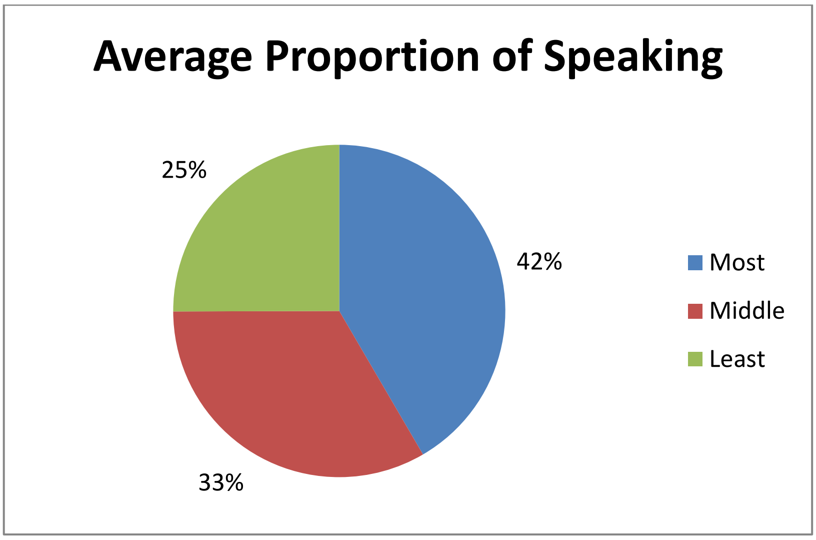


Figure 5.14: Average proportion of speaking

Figure 5.14 shows the average proportion of speaking across all groups. The average proportion is a 25-33-42 split, which is relatively close to the ideal split of 33-33-33. This implies that on average group members were able to express their ideas and opinions about the shared tasks through verbal communication relatively equally. It follows that groups could coordinate themselves effectively through verbal communication when using the system.

Figure 5.15 shows the practical contributions by each group member for each group. Practical contributions are grouped, ordered and represented in the same way as the verbal contributions. Furthermore, the same principles in terms of an equal split among group members apply to practical contributions as those to verbal contributions.

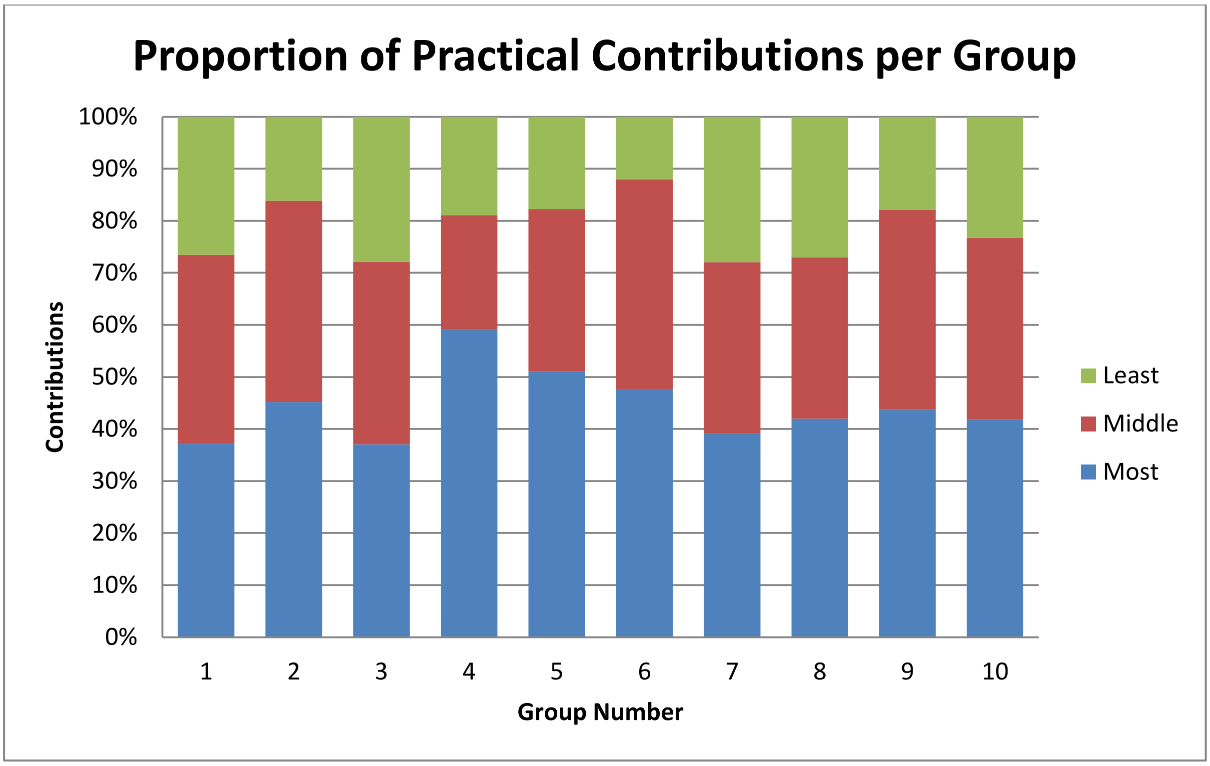


Figure 5.15: Proportion of practical contributions per group (n=10)

From Figure 5.15 it can be seen that there is slightly more variation in the proportion of practical contributions among group members per group when compared to verbal contributions. Each group member in each group did, however, practically contribute something to the shared tasks. Group 4 shows that one participant in the group dominated the practical contributions. Observation showed that the participant took the lead on the design of the logo in Task 3. Groups 2 and 6 both show that one participant did not practically contribute very much. From the video footage it was seen that in cases, the participant wasn't very comfortable with the technology.

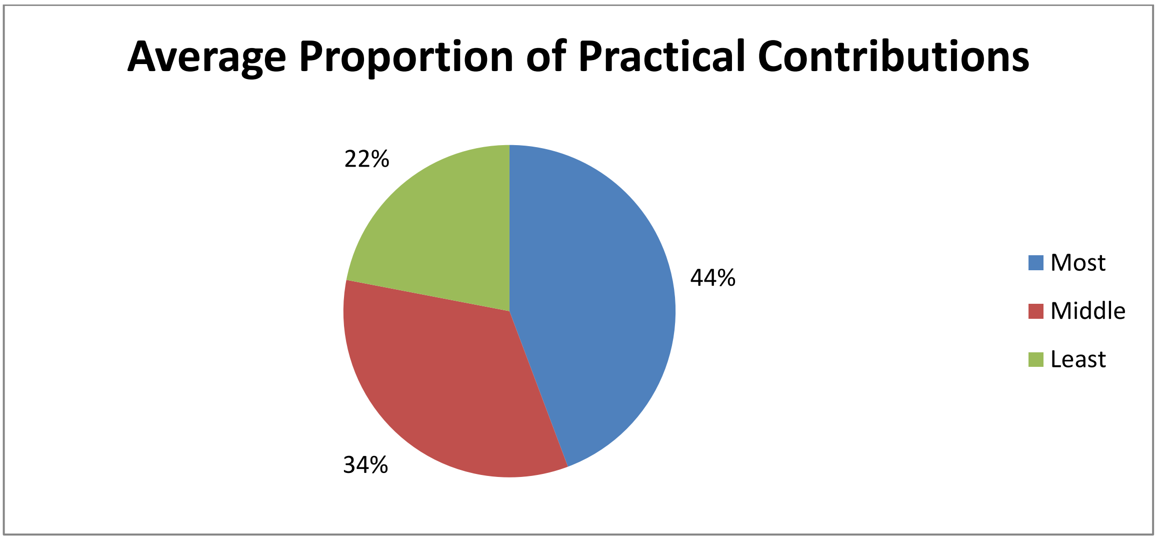


Figure 5.16: Average proportion of practical contributions

Figure 5.16 shows the average proportion of practical contributions across all groups. The average proportion is a 22-34-44 split, which is also relatively close to the ideal split of 33-33-33. This implies that on average group members were able to express their ideas about the shared tasks through practical contributions relatively equally. It follows that group members could work together effectively to achieve a common goal when using the system.

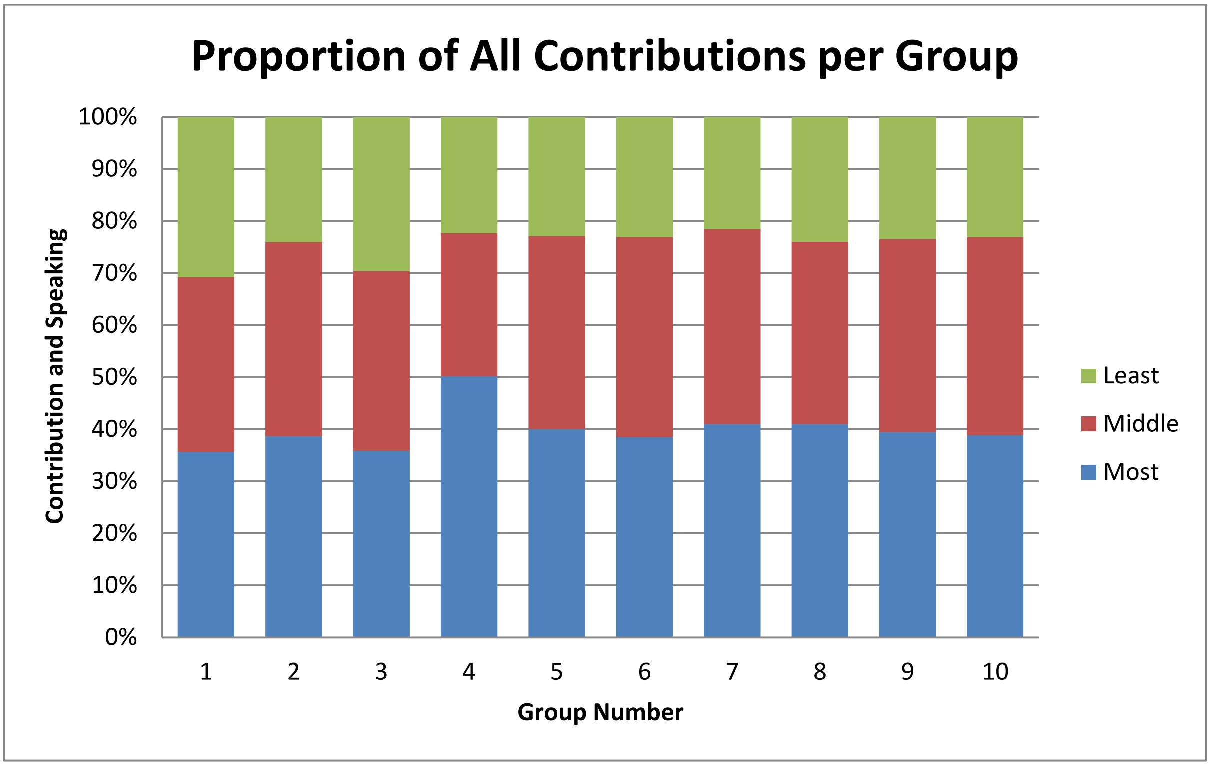


Figure 5.17: Proportion of all contributions per group (n=10)

Figure 5.17 shows the combination of both verbal and practical contributions by each group member for each group. The combination of both types of contributions are grouped, ordered and represented in the same way as before. It was discussed previously that for verbal and practical contributions independently, an unequal split in contribution did not necessarily indicate a low level of collaboration which can be seen in the case where a member may take or be given a leadership role in which that member speaks more whereas the others contribute to the task practically.

For all contributions, however, collaboration among peers is required to result in a relatively equal split since all group members must add value both verbally and practically in order to work together as a group. One group member doing everything or one member doing nothing is an indication of bad collaboration. Thus, an unequal split would indicate that the group did not work well together and that collaboration was hindered in the area of communication, coordination or information sharing.

From Figures 5.17 it can be seen that there is a relatively even proportion of combined contributions among group members per group. Group 4 shows that one participant in the group contributed notably more than the rest. This was largely owing to his many practical contributions in the design of the logo in Task 3.

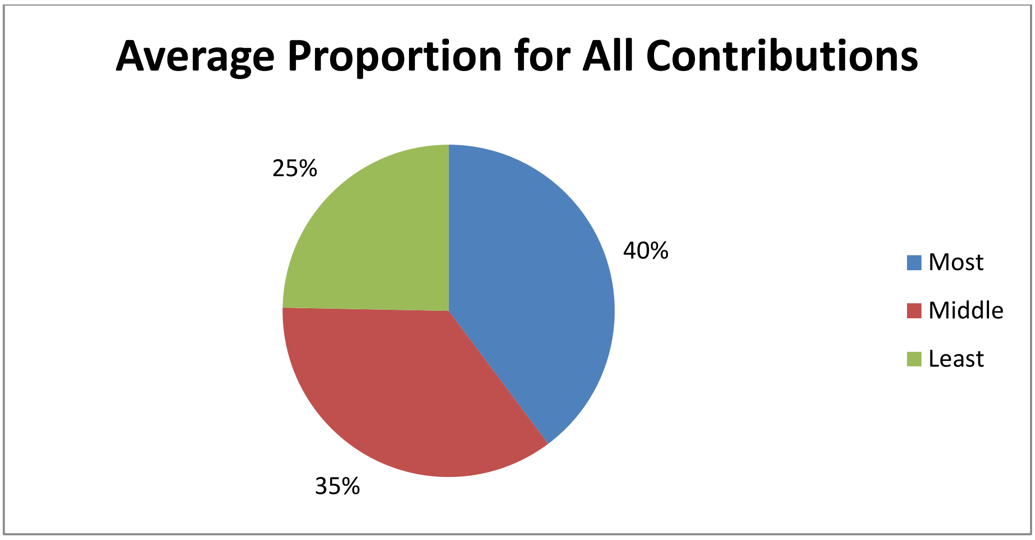


Figure 5.18: Average proportion for all contributions

Figure 5.18 shows the average proportion of verbal and practical contributions across all groups. The average proportion is a 25-35-40 split, which is the closest to the ideal split of 33-33-33. This implies that on average group members were able to verbally and practically contribute to the shared task relatively equally. Thus, it follows that effective collaboration was indeed achieved among group members when using the system.

5.5.3.2 User Perceptions Results

This section presents and discusses the results of participants' perceptions of collaboration when using GroupAware. These results are based on Section B of the post-test questionnaire, namely Collaboration. Cronbach's alpha was used to test the reliability of the Collaboration section as well as its sub-sections (see Table 5.10).

Table 5.10: Cronbach's alpha coefficients and interpretation for each Collaboration section (n=30)

Cronbach's Alpha Interval	Interpretation
B. Collaboration	0.74 – Good consistency
B1. Communication	0.85 – Excellent consistency
B2. Coordination	0.62 – Acceptable consistency
B3. Information Sharing	0.78 – Good consistency

It can be seen from Table 5.10 that the internal consistency of the Collaboration section is good. The Communication section has excellent consistency and the Coordination section has acceptable consistency. The consistency of the Information Sharing section is good. Thus, all the results from the Collaboration section can be considered as reliable.

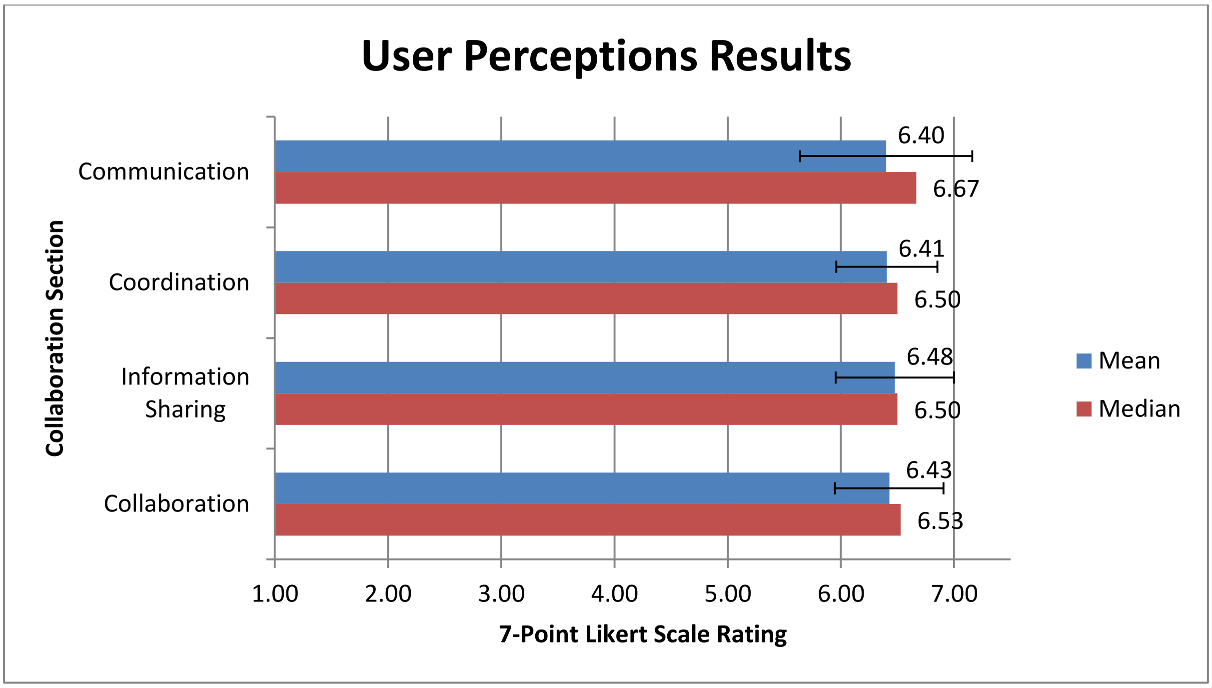


Figure 5.19: User perceptions results per Collaboration section (n=30)

Figure 5.19 shows the descriptive statistics of the Collaboration sections, namely Communication, Coordination and Information Sharing. Figure 5.19 reveals that the participants had a positive perception of the level of collaborative support that the system provides. All the sub-sections had an average rating greater or equal to 6.4 with low standard deviations. This is an encouraging result as the user study aimed to determine the system's effectiveness in supporting collaborative work.

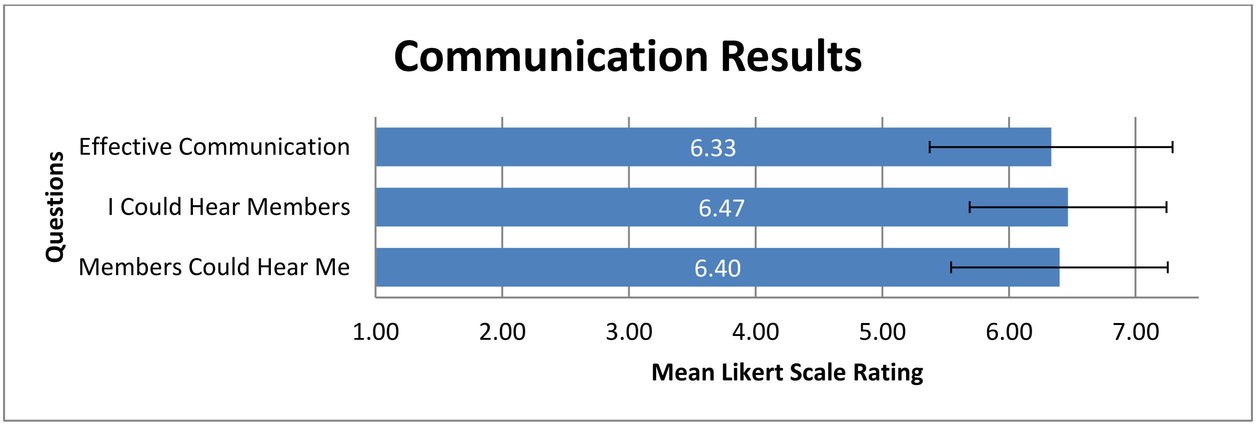


Figure 5.20: Mean 7-Point Likert scale ratings for the Communication section (n=30)

Figure 5.20 shows the average ratings and standard deviations for the first three questions of the Collaboration section, which correspond to the system's support for communication among group members. The results show average ratings of 6.33 and above, which indicates that participants perceived that there was good communication among the group. The highest rated question, which also had the lowest standard deviation, was Question 2 and is quoted below:

Q2: *"I could clearly hear what my group members were saying."*

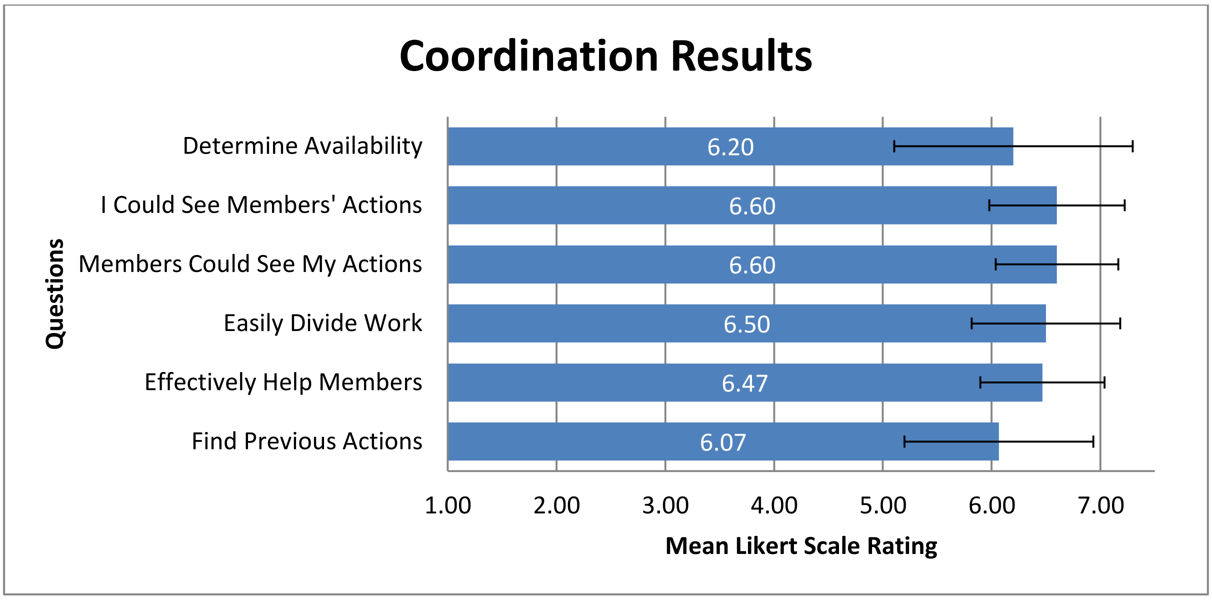


Figure 5.21: Mean 7-point Likert scale ratings for the Coordination section (n=30)

Figure 5.21 shows the average ratings and standard deviations for the next six questions of the Collaboration section, which correspond to the system's support for coordination among group members. The results show average ratings all above 6, indicating that participants perceived that their group was well coordinated and aware of each other during collaboration. The two highest rated questions, both averaging 6.6, were questions 5 and 6 and are quoted below:

Q5: *"I was able to see what my group members were doing."*

Q6: *"My group members were able to see what I was doing."*

Figure 5.22 shows the average ratings and standard deviations for the last three questions of the Collaboration section, which correspond to the system's support for information sharing among group members. The results show average ratings of 6.43 and above, which indicates that participants perceived that there was sharing of information among the group. The highest rated question, which also had the lowest standard deviation, was Question 10 and is quoted below:

Q10: *"I was able to effectively access the shared work."*

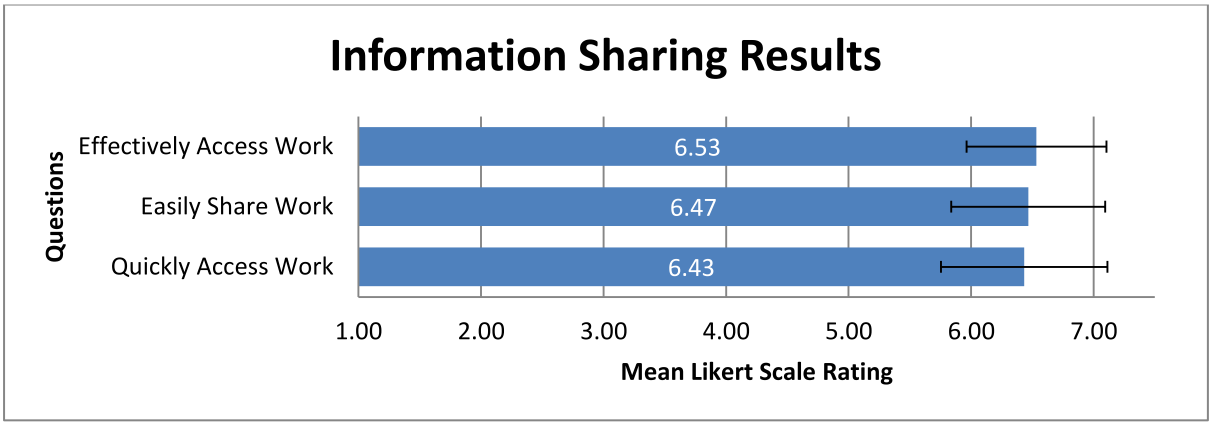


Figure 5.22: Mean 7-Point Likert scale ratings for the Information Sharing section (n=30)

5.5.3.3 General Collaboration Comments

This section presents and discusses the results of the participants' general comments regarding the collaboration of the system found in Section C of the post-test questionnaire. Similar to Section 5.5.2.3, the methods triangulation process was adopted here to seek out consistencies between the quantitative results presented in the previous section and the qualitative results presented in this.

The comments were first divided into positive and negative comments and then grouped into common themes. The number of users that expressed an opinion regarding each theme was recorded. The same condition was applied during thematic grouping as the one applied for the general usability comments. Furthermore, the methods triangulation was again utilised by thoroughly analysing the user study video footage for interesting activity and usage patterns in order to compare with and find further reasons for the responses from the questionnaire.

Table 5.11: Positive Collaboration themes and frequencies identified in user comments (n=30)

Category	Theme	Freq	%	Example
Collaboration	Supports collaboration	21	70%	"I could easily collaborate with my group members". "Great collaborative ability".
	Audio feed	14	47%	"The voice made it easier to collaborate". "Really nice that we could talk to each other".
	Simultaneous document editing	13	43%	"Being able to work on documents simultaneously with members".
	Able to see others' work	10	33%	"Effortless to know what my partners were doing". "[I] could see what was being done by others".
	User identification	3	10%	"Clear identification of users". "Different colours so I can see who is who".
	User availability	2	7%	"[I] could see if [my group members] were available".
	"What I Missed" feature	2	7%	"[I liked] being told what has happened during a break".

From Table 5.11 it is clear that the general comments from participants regarding collaboration were overwhelmingly positive. 70% of participants agreed that the system had “great collaborative ability”. The large percentage of participants commenting positively on the collaborative ability of GroupAware is a very encouraging result. Almost half the participants appreciated the live audio feed saying that it was “really nice that we could talk to each other” and that hearing each other’s “voice made it easier to collaborate”.

43% of participants stated that the best aspect of the system was the ability to work with others on the same document at the same time. The high frequency of this theme is encouraging because it is the core functionality of GroupAware. 33% of participants commented positively on the ability to see what others are doing. Other positive, but less frequent comments, were regarding the colour-coding of active users, the ability to determine user availability and the automatic logging of group members’ actions while away.

Participants were also asked to report on any negative aspects about the system’s usability. Table 5.12 shows the thematic grouping of those comments. In the category of Communication, four participants reported occasional delays in the audio feed. It was discovered that the audio delays were caused by high external usage of the network during the evaluation sessions. This is why only a few member experience delays. In terms of coordination, two participants reported negatively on the accuracy of the user detection saying that the “system would think I’m away, but it was just not picking me up properly”. This was as a result of the face detection functionality and improvements to the software library used would eradicate this issue.

Table 5.12: Negative Collaboration themes and frequencies identified in user comments (n=30)

Category	Theme	Freq	%	Example
Communication	Delay in audio feed	4	13%	“A bit of lag with my sound”.
Coordination	User detection accuracy	2	7%	“System would think I’m away, but it was just not picking me up properly”.

5.5.3.4 Laughter Results

This section presents and discusses the results obtained from the participants’ laughter. Figure 5.23 presents the total amount of laughter recorded for each group. It is important to note that laughter in a group could not only indicate enjoyment, but also confusion. Nervous laughter can occur when someone does not understand or is in a difficult situation (Mattsson, 2011). From the video footage, however, it was observed that participants only laughed genuinely when using the system.

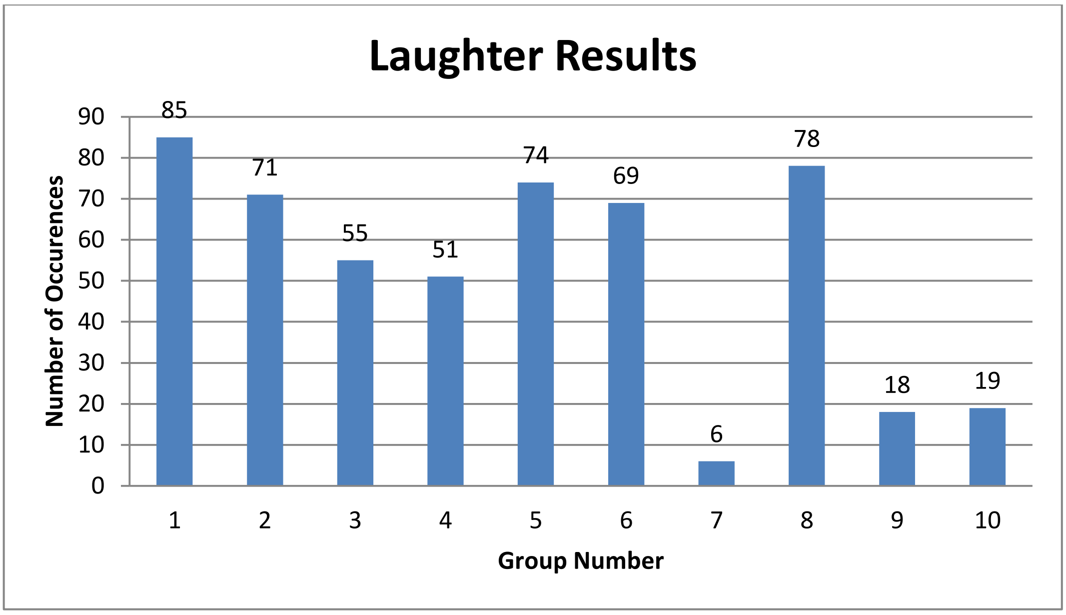


Figure 5.23: Total amount of laughter per group (n=10)

Figure 5.23 reveals that seven of the ten groups laughed a total of 50 or more times, with an average of 53 occurrences per group. Group 1 laughed the most with 85 occurrences and Group 7 laughed the least with 6 occurrences, which is considerably less than the average. From the video footage, Group 1 seemed to really enjoy the live feedback when others sketched and Group 7 seemed to take the tasks more seriously.

Although concrete conclusions cannot be made from the laughter results, some logical suggestions can be made. Earlier, it was shown that the group sessions took an average of 26 minutes. Thus, average laughter of 53 times per group means that groups averaged two laughs per minute. Although, there is no rule that governs what constitutes a lot of laughter, a group laughing twice a minute seems to have some significance. At best, the laughter results suggest that participants felt as if they part of a group and collaboration occurred. At worst, participants were simply enjoying the experience of using the system.

5.6 Discussion

From the usability and collaboration results, it is clear that the effectiveness of and response to GroupAware was extremely positive. In terms of performance, the participants could effectively and efficiently use the system to collaboratively complete the tasks given to them. A 98% overall task completion rate was achieved with an average session time of 26 minutes. A major positive theme from the general usability comments, with which 53% of participants agreed, was that participants could be effective and productive when using GroupAware. One participant in particular mentioned that GroupAware "can make work productive". This positive theme supports the performance results that were obtained.

User satisfaction results were high. Participants gave good ratings of the system in the PSSUQ, which averaged 6 out of 7 overall. More specifically, the PSSUQ section consisted of three sub-sections, namely Usability, Information Quality and Interface Quality. These sections scored average ratings of 6.14, 5.72 and 6.24 respectively as well as medians of 6.06, 5.86 and 6.33 respectively. The highest average rating obtained in the PSSUQ section was 6.53, for a question concerned with the learnability of the system. The next highest average ratings obtained was 6.43, for two questions relating to the interface of the system.

These high question ratings demonstrate that the NUI interaction techniques were incorporated correctly because NUIs are expected to be easy and enjoyable to use and have good learnability since they are focused on fostering natural user interaction. Four major positive themes from the general usability comments include "easy to use", "fun to use", "learnability" and "good UI". 70%, 63%, 43% and 33% of participants respectively commented on the four themes. These positive themes support the user satisfaction results that were obtained.

Some negative usability themes from the general comments were obtained. The main theme was clutter, which caused obstruction of system feedback. Other negative themes mentioned by participants included hardware issues, such as touch accuracy and sensitivity, and difficulty drawing. These negative themes, however, had low frequencies and were greatly outweighed by the positive ones.

Contribution results showed that on average all members of the group contributed fairly equally both verbally, via conversation, and practically, via task actions such as sketching. This implies good system support of collaboration. A major positive theme from the general collaboration comments, with which 70% of participants agreed, was that participants could easily collaborate as a group when using GroupAware. One participant in particular mentioned that GroupAware has "great collaborative ability". This positive theme supports the contribution results that were obtained.

Section B of the post-test questionnaire, namely Collaboration, evaluated how the participants perceived the level of collaboration that GroupAware supports. Overall, this section scored an average rating of 6.43 out of 7. More specifically, the Collaboration section consisted of three sub-sections that corresponded to the three main CSCW characteristics, namely Communication, Coordination and Information Sharing. These sections scored average ratings of 6.4, 6.41 and 6.48 respectively as well as medians of 6.67, 6.5 and 6.5 respectively. The highest average rating obtained in the Collaboration section was 6.6, for two questions concerned with the coordination aspect of the system. The next highest average ratings obtained was 6.53, for a question relating to the access to a shared workspace that the system provides.

These high question ratings demonstrate that participants perceived the level of collaboration in terms of all three characteristics to be high. More specifically, the question with the highest average rating came from the Coordination section, which was expected since the research placed its emphasis on providing good group coordination and awareness. Three positive themes from the general collaboration comments include "able to see others' work", "user identification" and "user availability". 33%, 10% and 7% of participants respectively commented on the three themes. These positive themes support the user perception results that were obtained.

Laughter results indicated that groups laughed a lot during the session, which could mean two things: either that there was synergy among the groups or that participants were enjoying themselves while using the system. The contribution and user perception results indicate that both of these possibilities were true. Thus the laughter results supported the collaboration results.

Two negative collaboration themes from the general comments were obtained. These were "delay in audio feed" and "user detection accuracy". These themes were found to be caused by network and software library limitations and not by the system itself. Therefore, no substantial negative themes concerning collaboration were found.

From all the above results it can be concluded that the GroupAware, a groupware system incorporating NUI interactions techniques, provides effective support for SDCW in terms of communication, coordination and information sharing. This answers the sixth research question, namely:

> *RQ6. How effectively does the developed prototype support SDCW?*

5.7 Conclusions

This chapter described the evaluation of GroupAware with the primary objective of achieving the aim of research identified in Chapter 1. Furthermore, secondary objectives of the evaluation included determining GroupAware's usability and collaboration support. An investigation into groupware evaluation techniques and extant systems' evaluations was conducted and it was concluded that a user observational study combined with user questionnaires would be employed for the evaluation of GroupAware.

The experimental design of the evaluation was discussed in Section 5.4. The evaluation of GroupAware involved thirty participants assembled into ten groups of three members each. Each group member was provided with a task list, which contained the scenario description as well as the tasks to be completed. The groups were required to complete the given tasks using GroupAware. Each participant completed a biographical and a post-test questionnaire. The internal consistency of the post-test questionnaire was estimated using Cronbach's alpha to make sure the questionnaire, including each sub-section, was reliable so that the results could be trusted.

The results of the evaluation were presented, analysed and discussed in Sections 5.5 and 5.6. The results indicated that GroupAware obtained high levels of usability and collaboration. Positive results of performance and user satisfaction demonstrated that GroupAware is an effective groupware system that is simple and enjoyable to use and is highly learnable. Positive results of verbal and practical contributions and user perceptions of collaborative support demonstrated that GroupAware supported collaboration in terms of communication, coordination and information sharing. Furthermore, participants gave an abundance of positive subjective feedback about the system in the general comments section of the post-test questionnaire that greatly outweighed the negative comments. The positive comments provided further support of the results.

The positive usability results indicated that the NUI interaction techniques were incorporated correctly into the groupware prototype and the positive collaboration results confirmed that the techniques effectively supported SDCW. The next chapter concludes this dissertation by identifying the contributions made by this research. In addition, several points are discussed and presented for possible future work.

Chapter 6: Conclusions

6.1 Introduction

This chapter will conclude the dissertation by revisiting the original objectives of this research to determine whether they were achieved. This chapter involves the final phase of DSR, namely Communicate Findings. This chapter answers the last research question discussed in Chapter 1, namely:

RQ7. What are the research contributions and what future research should be carried out to improve NUI interaction techniques for SDCW?

The chapter begins with a discussion of the achievements that have been attained through this research (Section 6.2). The known limitations of the research and the problems that were faced along the way are then discussed (Section 6.3). The contributions made by this research are then presented in terms of both the theoretical and practical contributions (Section 6.4). The chapter concludes with recommendations of future work (Section 6.5).

6.2 Research Achievements

The research identified that existing shared-workspace groupware systems do not provide effective support for SDCW. The goal that this research aimed to achieve articulated in the following primary research objective:

To investigate and evaluate the use of Natural User Interface (NUI) interaction techniques to support Synchronous Distributed Collaborative Work (SDCW).

The following objectives were derived in order to fulfil the aim of this research:

- RO1. To define and discuss the concept of SDCW and investigate the requirements thereof (Chapter 2).
- RO2. To identify the limitations of existing groupware systems (Chapter 2).
- RO3. To define and discuss existing NUI interaction techniques (Chapter 3).
- RO4. To determine how NUI interaction techniques can be incorporated into shared-workspace groupware systems to address the existing limitations (Chapter 3).
- RO5. To design and develop a shared-workspace groupware prototype that incorporates NUI interaction techniques to support SDCW (Chapter 4).

RO6. To determine how effectively NUIs support SDCW through an evaluation of the prototype (Chapter 5).

RO7. To identify the research contributions and make recommendations for future research in order to improve the support for SDCW using NUI interaction techniques (Chapter 6).

DSR was used to design and evaluate a solution to address the problem identified. DSR consists of six phases to support the creation of a solution to the problem identified. The six phases were used to address each of the research objectives identified above to design and evaluate a solution to the problem.

Chapter 2 achieved the first two research objectives by defining and discussing the concept of SDCW, investigating the functional and non-functional requirements thereof, and identifying the limitations of existing groupware systems. Chapter 2 involved the second phase of the DSR methodology, which is the requirements definition phase. A literature review was conducted in order to achieve the objectives. Relevant literature found in CSCW's rich body of knowledge that comprises more than three decades of research was reviewed in order to define the concept of collaboration, with specific attention given to SDCW. Five characteristics of CSCW were defined, namely communication, coordination, information sharing, time and space. An in-depth discussion of these characteristics gave insight into the meaning of collaboration and SDCW. Existing groupware systems were then identified, classified and discussed in order to highlight the benefits and limitations. The benefits included that communication and information sharing was well-supported and the limitation was that coordination was a problem in shared-workspace groupware systems. The mechanics of collaboration were defined and discussed, upon which the functional requirements of a shared-workspace groupware system were based. The non-functional requirements were identified and presented.

Chapter 3 achieved the next two research objectives by defining and discussing existing NUI interaction techniques, and determining how those techniques can be incorporated into a shared-workspace groupware system to address the existing limitations and problems. A second literature review was conducted to define NUIs. Four types of natural interaction, including touch, speech, in-air gestures and proxemics, were defined and their advantages and disadvantages discussed. In order to see how these NUI interaction techniques have been used, a review of related synchronous co-located and distributed groupware systems was conducted. The devices, interaction techniques and tasks supported by each system were discussed in order to select the appropriate interaction techniques for this research. Multi-touch and proxemics interaction techniques were chosen and a mapping of the SDCW requirements onto these techniques was proposed.

Chapter 4 achieved the fifth research objective by designing and developing a shared-workspace groupware prototype called GroupAware that incorporates NUI interaction techniques to support SDCW. Chapter 4 involved the third and fourth phases of the DSR methodology, in which the artefact is designed, developed and demonstrated. The development methodology and application domain of GroupAware were discussed. A system architecture based on the requirements of a shared-workspace groupware system was proposed. The architecture comprised an application architecture and a distribution architecture. An in-depth discussion of the design of GroupAware was given in terms of the functionality, data, UI and user interactions. The manner in which GroupAware was implemented to address the functional requirements was explained. The tools used to implement the prototype were discussed. The four prototypes that were integrated to form GroupAware were discussed in terms of the challenges faced and how they were overcome. Finally, the actual functionality implemented was described.

Chapter 5 achieved the sixth research objective by determining how effectively NUIs support SDCW through an evaluation of GroupAware. Chapter 5 involved the fifth phase of the DSR methodology, which is the artefact evaluation phase. The objectives of the evaluation of GroupAware were described. An investigation into groupware evaluation techniques and extant systems evaluations was conducted and it was concluded that a user observational study combined with user questionnaires would be employed for the evaluation of GroupAware. The experimental design of the evaluation was discussed. The evaluation of GroupAware involved ten groups of three members each, who were required to complete scenario-based tasks using GroupAware. The results of the evaluation were presented and discussed. The results indicated that GroupAware obtained high levels of usability and collaboration. Furthermore, participants gave an abundance of positive general comments that greatly outweighed the negative comments. The positive comments further supported the results. The positive results confirmed that the incorporated NUI interaction techniques effectively supported SDCW.

6.3 Limitations and Problems Faced

This section highlights the limitations of the research and the problems that were encountered during implementation and evaluation of GroupAware. The following two research limitations were identified:

1. The prototype was limited to being deployed on multi-touch devices running Windows 7 operating system. The system, therefore, cannot be used with Linux, Android and Apple devices.

2. The interaction techniques and architecture were only tried and tested in the GroupAware prototype. More benefits may be identified if the proposed NUI interaction techniques are incorporated into other related systems.

Several problems were encountered during the implementation (Section 6.3.1) and evaluation (Section 6.3.2) of the GroupAware prototype. These problems included both hardware and software related problems. These problems and how they were overcome are discussed in this section, starting with the implementation problems.

6.3.1 Implementation Problems

The problems faced during implementation included application debugging issues and software library incompatibilities. Like any software application, GroupAware had to be debugged frequently to find and reduce bugs in the code. For a groupware system, however, the complexity of the task is greatly increased because of three factors including code complexity, multiple users and the network. Writing a groupware system is more complex than writing a single-user system. Consequently, debugging a groupware system is more complex than debugging a single-user system. To solve this issue, a groupware toolkit called GT was used to simplify the implementation of GroupAware. The debugging was therefore simplified.

Multiple users are expected to interact with the system simultaneously over a network. The GroupAware was expected to be stable for groups of three members. The ideal conditions to have in order to debug the code are three people running GroupAware on three different devices connected by a stable network. These conditions were not always met during implementation, which made debugging difficult and in some cases impossible. Two solutions to this problem were implemented. The first was to run GroupAware three times on the same device, connecting to the "localhost", switching between clients and performing the necessary actions. The issues with the first solution were the difficultly in testing simultaneous actions and the additional effort required to constantly switch between the clients. The second solution was to create the ideal conditions. The issue with the second solution was finding the people and devices to use.

Various Open-Source software libraries were used in the implementation of GroupAware and some software library incompatibility issues arose during the implementation of GroupAware. Software libraries were required to be compatible with C#, WPF and Surface applications. There were two main occurrences of the incompatibility issue. Firstly, when it was attempted to render a PDF file in a Surface application and also when it was attempted to create the live audio and video feed. Attempts were made to render a PDF file using Debenu Quick PDF Library Lite and Adobe PDF

Reader. These attempts, however, were unsuccessful since the libraries were incompatible with WPF. No software library supporting C#, WPF and Surface applications could be found. A workaround to this challenge was to first export the PDF file as multiple standard images, one image per page. These images could then be successfully rendered in the Surface application.

The SDK for the VLC media player, called libVLC media framework, was found. LibVLC can be embedded into an application to allow access to multimedia capabilities such as audio and video streaming. The problem with this SDK was that it was incompatible with WPF and Surface applications. A solution was found in the VLCDotNet library, which was compatible with WPF and Surface applications and provided the same functionality as libVLC.

6.3.2 Evaluation Problems

The problems faced during evaluation included hardware and network related problems. The hardware used in the evaluation of GroupAware had some inherent issues. The first hardware issue was that of accessibility. Three Windows 7 multi-touch mobile devices were needed, but only two were available. An additional multi-touch device was needed. To solve this problem NMMU Telkom/CoE mutli-touch tabletop was used during evaluation.

The second hardware issue related to the NMMU Telkom/CoE mutli-touch tabletop. The device is a custom-built tabletop comprising of a television and a multi-touch sensor overlay and it had two issues, namely touch sensitivity and touch accuracy. The sensor overlay occasionally became highly sensitive and sensed touches from participants' sleeves or task lists. Furthermore, the distance between the television screen and the sensor overlay is significant enough to make touching inaccurate when looking at the tabletop from an angle.

NMMU CS Department's network was used to connect the devices during the evaluation. This caused a problem because NMMU CS Department's network usage increases during the lunch break (12:05 – 13:05) since all students are free to use the network. Participant groups that were scheduled for an evaluation during this time experienced network delay, which mainly affected the audio feed. To solve this problem, groups were, as far as possible, scheduled before and after this time.

6.4 Research Contributions

The contributions of this research can be divided into both a theoretical (Section 6.3.1) and a practical (Section 6.3.2) contribution. These contributions are discussed separately in this section, beginning with the theoretical contribution.

6.4.1 Theoretical Contributions

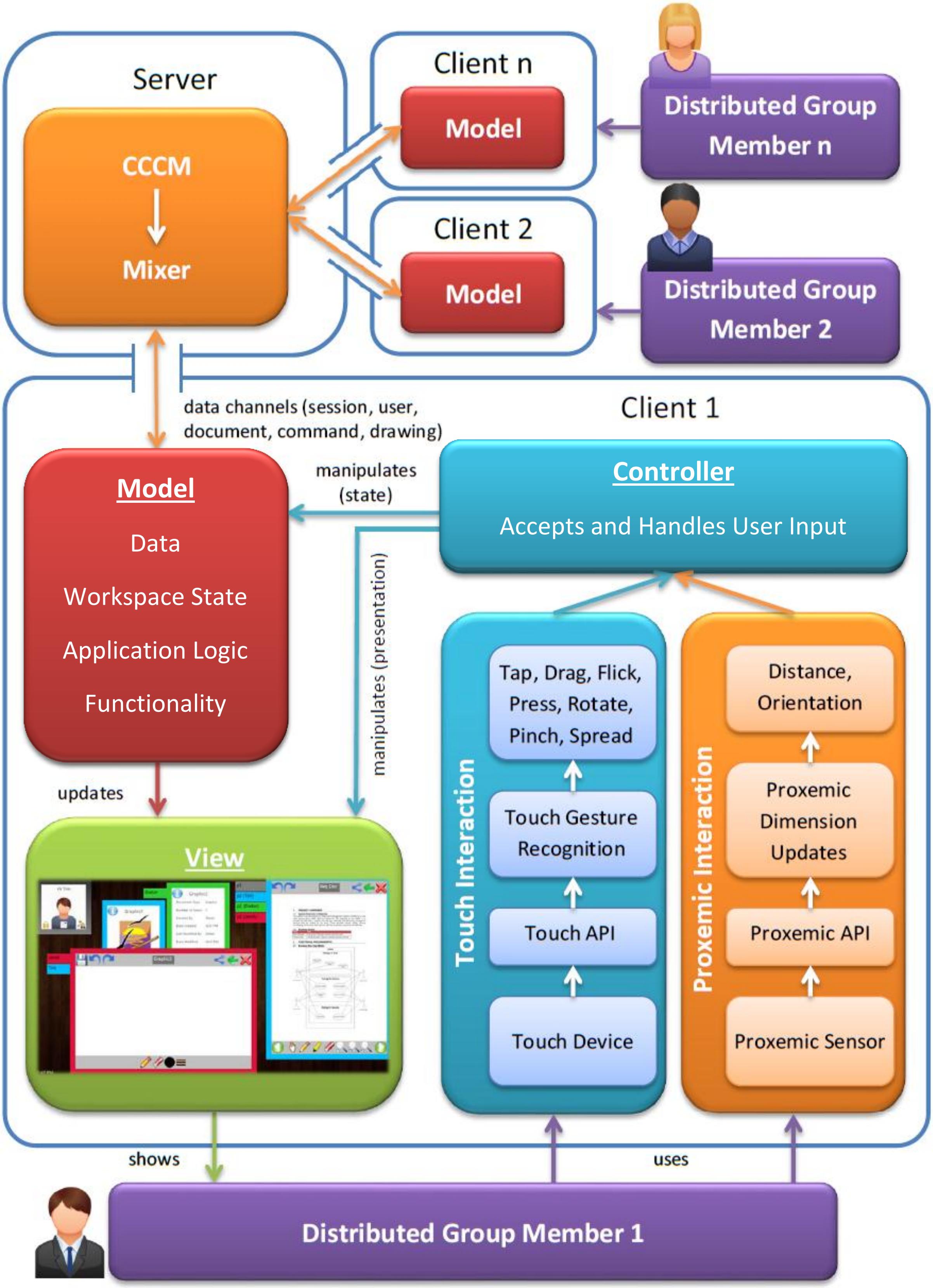


Figure 6.1: System architecture of GroupAware

The main theoretical contribution of this research project was demonstrating that NUI interaction techniques could be designed to effectively support SDCW. The evaluation results provided empirical evidence that by using NUI interaction techniques for SDCW, a highly effective and efficient SDCW environment could be established. The results showed that a shared-workspace groupware system incorporating NUI interaction techniques provides several benefits such as high levels of performance, user satisfaction, learnability, collaboration and enjoyment. With the implementation of GroupAware, it was established that integrating SDCW with NUI interaction techniques is feasible and could form a basis for other developers to apply these interaction techniques to similar groupware systems.

An architecture to support a shared-workspace groupware system was presented in Section 4.4.6 (see Figure 6.1). This architecture comprised the MVC architecture pattern which decouples the workspace state from the both the workspace UI and interaction, which increases flexibility and encourages code reuse. Furthermore, the architecture comprised the Centralized Mixer with Broadcaster distribution architecture which supports scalability and increases control of the trade-off between fidelity and feedthrough time (performance versus quality). The proposed architecture was used in the design of the prototype. The successful implementation and positive evaluation of GroupAware implies that the architecture effectively support SDCW. This architecture, therefore, could be employed in similar groupware systems that make use of NUI interaction techniques.

6.4.2 Practical Contributions

The main practical contribution of this research project was the design and implementation of GroupAware, a shared-workspace groupware prototype that incorporates NUI interaction techniques. GroupAware satisfies the requirements of SDCW by allowing groups to work together in real-time across distributed locations by means of a shared visual workspace. Text and graphic documents are represented as widgets in the workspace and can be manipulated via touch gestures. Tasks such as viewing and annotating text documents and creating graphical documents are supported by GroupAware. GroupAware allows communication though a constant live audio and an on-demand video feed. Group coordination and awareness is supported through basic proxemic interaction, real-time synchronization of workspace and storing history logs of user actions.

The results of the evaluation indicated that GroupAware obtained high levels of usability and collaboration. Positive results of performance and user satisfaction demonstrated that GroupAware is an effective groupware system that is simple and enjoyable to use and is highly learnable. Positive results of verbal and practical contributions and user perceptions of collaborative support demonstrated that GroupAware supported collaboration in terms of communication, coordination and information sharing. Furthermore, participants gave an abundance of positive subjective feedback about the system in the general comments section of the post-test questionnaire that greatly outweighed the negative comments. The positive comments provided further support of the results. The positive usability results indicated that the NUI interaction techniques were incorporated correctly into the groupware prototype and the positive collaboration results confirmed that the techniques effectively supported SDCW. Therefore, the prototype itself is a practical contribution. Figure 6.2 shows a screenshot of a GroupAware client's workspace during a collaborative session.

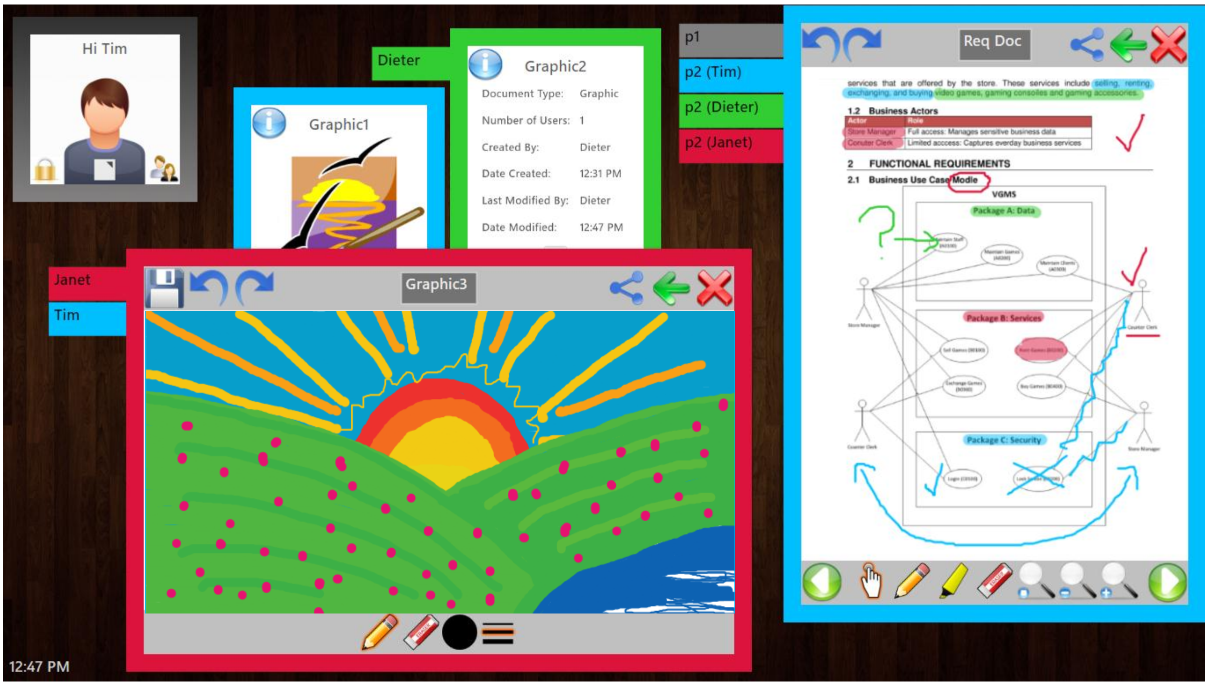


Figure 6.2: Screenshot of a GroupAware client's workspace

6.5 Future Research

Several opportunities for future work were identified based on the results of this research in terms of further development and evaluation. Firstly, improving GroupAware and re-evaluating it with a wider selection of participants, could provide more insight and conclusive results into the effectiveness of the architecture, the appropriateness of the NUI interaction techniques and the collaborative ability of the groupware system.

This research only focussed on small groups in synchronous, distributed collaborative environments. Future research could include extending the system to support mixed-presence groups. Mixed-presence groups include both co-located and distributed users. The workspace could be adjusted to cater for multiple simultaneous co-located users. This would enable a larger group to work together by means of multiple distributed groups for increased productivity.

GroupAware only incorporated multi-touch and basic proxemic interaction. More NUI interaction techniques could be incorporated and their effectiveness evaluated or even compared. In-air gestures, speech interaction and enhanced proxemic interaction could be incorporated into the collaborative workspace for more natural user interaction and improved collaborative support.

GroupAware was platform dependant, requiring Windows 7 in order to run optimally. GroupAware could therefore be ported on to other popular platforms such as Linux, Android and Apple. Furthermore, support for smartphones could be investigated. Smartphones are getting smarter every day, being able to support a variety of NUI interaction techniques such as in-air gestures, shaking and pointing. GroupAware could be adapted to support smartphone interaction techniques.

Finally, the evaluation was a scenario-based controlled lab experiment with ten very similar groups. Comparative studies with related systems or long-term field studies in an organization could be done with more participants and more diverse groups to provide more insight into the collaborative support that the system provides.

References

ACM. (2014). The 17th ACM Conference on Computer-Supported Cooperative Work & Social Computing. Retrieved from http://cscw.acm.org/2014/cscw2014_program.pdf

Adobe Systems. (2014a). About Adobe PDF. Retrieved from http://www.adobe.com/africa/products/acrobat/adobepdf.html

Adobe Systems. (2014b). Adobe PDF Reader. Retrieved from http://www.adobe.com/africa/products/reader.html

Ambler, S. (2006). The Agile Unified Process (AUP). Retrieved from http://www.ambysoft.com/unifiedprocess/agileUP.html

Antunes, P., Herskovic, V., Ochoa, S. F., & Pino, J. A. (2012). Structuring Dimensions for Collaborative Systems Evaluation. *ACM Computing Surveys*, *44*(2), 1 – 31.

Apple. (2013). Apple iPad. Retrieved from http://www.apple.com/ipad/overview/

Araujo, R. M. De, Santoro, F. M., & Borges, M. R. S. (2004). A conceptual framework for designing and conducting groupware evaluations. *International Journal of Computer Applications in Technology*, *19*(3/4), 139. doi:10.1504/IJCAT.2004.004043

Ardaiz, O., Arroyo, E., Righi, V., Galimany, O., & Blat, J. (2010). Virtual collaborative environments with distributed multitouch support. In *Proceedings of the 2nd ACM SIGCHI symposium on Engineering interactive computing systems - EICS '10* (p. 235). New York, New York, USA: ACM Press. doi:10.1145/1822018.1822055

Artman, J. (2010). The Disadvantages of Voice Recognition Software. Retrieved from http://www.ehow.com/list_6656013_disadvantages-voice-recognition-software.html

Baker, K., Greenberg, S., & Gutwin, C. (2001). Heuristic Evaluation of Groupware Based on the Mechanics of Collaboration. In M. R. Little & L. Nigay (Eds.), *Engineering for Human-Computer Interaction (8th IFIP International Conference, EHCI 2001, Toronto, Canada, May)* (Vol. 2254, pp. 123–139). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/3-540-45348-2

Ballendat, T., Marquardt, N., & Greenberg, S. (2010). Proxemic Interaction: Designing for a Proximity and Orientation-Aware Environment. In *ACM International Conference on Interactive Tabletops and Surfaces - ITS '10* (pp. 121–130). New York, New York, USA: ACM Press. doi:10.1145/1936652.1936676

Baudisch, P., Tan, D., Collomb, M., Robbins, D., Hinckley, K., Agrawala, M., … Ramos, G. (2006). Phosphor. In *Proceedings of the 19th annual ACM symposium on User interface software and technology - UIST '06* (pp. 169–178). New York, New York, USA: ACM Press. doi:10.1145/1166253.1166280

Belcher, L. M. (2011). Advantages and Disadvantages of Collaboration in the Workplace. Retrieved from http://smallbusiness.chron.com/advantages-disadvantages-collaboration-workplace-20965.html

Benko, H., Wilson, A. D., & Baudisch, P. (2006). Precise selection techniques for multi-touch screens. In *Proceedings of the SIGCHI conference on Human Factors in computing systems - CHI '06* (p. 1263). New York, New York, USA: ACM Press. doi:10.1145/1124772.1124963

Bevan, N. (1999). Common Industry Format Usability Tests. In *Proceedings of UPA '98* (pp. 1–6).

Bill Hughes. (2014). *Samsung Galaxy S5 For Dummies* (p. 352).

Blake, J. (2012). Introducing NUI concepts. In *Natural User Interfaces in .NET* (pp. 1–43).

Böhmer, M., Saponas, T. S., & Teevan, J. (2013). Smartphone Use Does Not Have to Be Rude: Making Phones a Collaborative Presence in Meetings. In *Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services - MobileHCI '13* (pp. 342–351).

Bragdon, A., DeLine, R., Hinckley, K., & Morris, M. R. (2011). Code Space: Touch + Air Gesture Hybrid Interactions for Supporting Developer Meetings. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces - ITS '11* (pp. 212–221). New York, New York, USA: ACM Press. doi:10.1145/2076354.2076393

Camazine, S., Deneubourg, J.-L., Franks, N. R., Sneyd, J., Theraulaz, G., & Bonabeau, E. (2003). *Self-Organization in Biological Systems* (p. 18).

Castle, A. (2009). Build Your Own Multitouch Surface Computer. Retrieved from http://www.maximumpc.com/article/features/maximum_pc_builds_a_multitouch_surface_computer

Chittaro, L. (2006). Visualizing Information on Mobile Devices. *Computer*, *39*(3), 40–45. doi:10.1109/MC.2006.109

CompareHero. (2013). Ultimate 5 Tips- Choose the Right Smartphone. Retrieved from http://www.comparehero.my/blog/ultimate-5-tips-choose-the-right-smartphone/

Cronbach, L. J. (1951). Coefficient Alpha and the Internal Structure of Tests. *Psychometrika*, *16*(3), 297–335.

De Alwis, B., Gutwin, C., & Greenberg, S. (2009). GT/SD: Performance and Simplicity in a Groupware Toolkit. In *Proceedings of the 1st ACM SIGCHI symposium on Engineering interactive computing systems - EICS '09* (pp. 265–274). New York, New York, USA: ACM Press. doi:10.1145/1570433.1570483

Debenu. (2014). Debenu Quick PDF Library Lite Overview. Retrieved from http://www.debenu.com/products/development/debenu-pdf-library-lite/

Dillenbourg, P., Baker, M. J., Blaye, A., & O'Malley, C. (1996). The evolution of research on collaborative learning. In E. Spada & P. Reiman (Eds.), *Learning in Humans and Machine: Towards an interdisciplinary learning science* (pp. 189–211). Oxford: Elsevier. Retrieved from http://telearn.archives-ouvertes.fr/docs/00/19/06/26/PDF/Evolution.pdf

Ditta, M. A., Wesson, J., & Cowley, L. (2013). CollaGIM: Supporting Group Information Management on a Multi - touch Surface. In *Proceedings of the 2013 SATNAC Conference* (pp. 199–204).

EmguCV. (2014). Main Page. Retrieved from http://www.emgu.com/wiki/index.php/Main_Page

Fowler, M. (2006). GUI Architectures. Retrieved from http://martinfowler.com/eaaDev/uiArchs.html

Genest, A. M., Gutwin, C., Tang, A., Kalyn, M., & Ivkovic, Z. (2013). KinectArms: a Toolkit for Capturing and Displaying Arm Embodiments in Distributed Tabletop Groupware. In *Proceedings of the 2013 conference on Computer supported cooperative work - CSCW '13* (pp. 157–166). New York, New York, USA: ACM Press. doi:10.1145/2441776.2441796

GitHub. (2014). VLCDotNet. Retrieved from https://github.com/ZeBobo5/Vlc.DotNet

Google. (2014a). Gmail - Email from Google. Retrieved from https://www.gmail.com/intl/en/mail/help/about.html

Google. (2014b). Overview of Google Docs - Drive Help. Retrieved from https://support.google.com/drive/answer/143206?hl=en-GB

Graham, T. C. N., Phillips, W. G., & Wolfe, C. (2006). Quality Analysis of Distribution Architectures for Synchronous Groupware. In *2006 International Conference on Collaborative Computing: Networking, Applications and Worksharing* (pp. 1–9). IEEE. doi:10.1109/COLCOM.2006.361870

Greenberg, S. (1989). The 1988 Conference on Computer-Supported Cooperative Work : Trip Report. *ACM SIGCHI Bulletin*, *20*(April), 49–55.

Greenberg, S. (2001). Supporting Casual Interaction between Intimate Collaborators. In *EHCI '01 Proceedings of the 8th IFIP International Conference on Engineering for Human-Computer Interaction* (Vol. 2254, pp. 3–4). Springer-Verlag London, UK ©2001. doi:10.1007/3-540-45348-2_2

Greenberg, S. (2002). *Real Time Distributed Collaboration* (pp. 1–7). Alberta, Canada.

Greenberg, S., & Buxton, B. (2008). Usability evaluation considered harmful (some of the time). In *Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI '08* (p. 111). New York, New York, USA: ACM Press. doi:10.1145/1357054.1357074

Gutwin, C., & Greenberg, S. (2000). The mechanics of collaboration: developing low cost usability evaluation methods for shared workspaces. In *Proceedings IEEE 9th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE 2000)* (pp. 98–103). IEEE Comput. Soc. doi:10.1109/ENABL.2000.883711

Gutwin, C., & Greenberg, S. (2002). A Descriptive Framework of Workspace Awareness for Real-Time Groupware. *Computer Supported Cooperative Work*, *11*(3), 411–446. doi:10.1023/A:1021271517844

Gutwin, C., & Greenberg, S. (2004). The Importance of Awareness for Team Cognition in Distributed Collaboration. In E. Salas & S. M. Fiore (Eds.), *Team Cognition: Understanding the Factors that Drive Process and Performance* (pp. 177–201). APA Press.

Gutwin, C., Greenberg, S., Blum, R., Dyck, J., Tee, K., & Mcewan, G. (2008). Supporting Informal Collaboration in Shared-Workspace Groupware. *Journal of Universal Computer Science*, *14*(9), 1411–1434.

Haller, M., Leitner, J., Seifried, T., Wallace, J. R., Scott, S. D., Richter, C., … Hunter, S. (2010). The NiCE Discussion Room: Integrating Paper and Digital Media to Support Co-Located Group Meetings. In *Proceedings of the 28th international conference on Human factors in computing systems - CHI '10* (pp. 609–618). New York, New York, USA: ACM Press. doi:10.1145/1753326.1753418

Hampton, T. J. (2011). The Touchscreen Revolution and Its Limitations. Retrieved from http://pledgingforchange.com/guest-blog-posts/the-touchscreen-revolution-and-its-limitations.php

Han, J. Y. (2005). Low-cost multi-touch sensing through frustrated total internal reflection. In *Proceedings of the 18th annual ACM symposium on User interface software and technology - UIST '05* (pp. 115–118). New York, New York, USA: ACM Press. doi:10.1145/1095034.1095054

Hearst, M. A. (2011, November 1). "Natural" search user interfaces. *Communications of the ACM*, *54*(11), 60–67. doi:10.1145/2018396.2018414

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 Science Research in Information Systems. In *Design Research in Information Systems* (Vol. 22, pp. 9–22). Boston, MA: Springer US. doi:10.1007/978-1-4419-5653-8

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly*, *28*(1), 75–105. doi:10.2307/249422

Higher National Computing. (2007). Incremental Prototyping. Retrieved from http://www.sqa.org.uk/e-learning/IMAuthoring01CD/page_09.htm

Hinckley, K., Jacob, R., & Ware, C. (2004). Input/Output Devices and Interaction Techniques. In A. B. Tucker (Ed.), *Computer Science Handbook* (Second Edi., pp. 506–537). Chapman & Hall/CRC.

Hofstee, E. (2009). *Constructing a Good Dissertation: A Practical Guide to Finishing a Master's, MBA or PhD on Schedule*. Johannesburg, South Africa: EPE.

Inkpen, K., DiMicco, J. M., Scott, S., & Mandryk, R. (2004). Methodology for Evaluating Collaboration Behaviour in Co-Located Environments. Chicago, IL. doi:10.1.1.59.738

Itseez. (2014). OpenCV. Retrieved from http://opencv.org/

Izadi, S., Agarwal, A., Criminisi, A., Winn, J., Blake, A., & Fitzgibbon, A. (2007). C-Slate: A Multi-Touch and Object Recognition System for Remote Collaboration using Horizontal Surfaces. In *Second Annual IEEE International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP'07)* (pp. 3–10). IEEE. doi:10.1109/TABLETOP.2007.34

Johansen, R. (1988). *GroupWare: Computer Support for Business Teams*. NY, USA: The Free Press New York. Retrieved from http://dl.acm.org/citation.cfm?id=542298

KiSSFLOW. (2014). Business Process Management Tool & Workflow Software. Retrieved from http://kissflow.com/

Krauß, M., Riege, K., Winter, M., Pemberton, L., Iais, F., Birlinghoven, S., & Augustin, S. (2009). Remote Hands-on Experience: Distributed Collaboration with Augmented Reality. In U. Cress, V. Dimitrova, & M. Specht (Eds.), *Learning in the Synergy of Multiple Disciplines* (Vol. 5794, pp. 226–239). Berlin: Springer-Verlag. doi:10.1007/978-3-642-04636-0_22

Labs, P. (2014). PQ Labs. Retrieved from http://multitouch.com/

Lanubile, F. (2009). Collaboration in Distributed Software Development. In A. De Lucia & F. Ferrucci (Eds.), *Software Engineering* (pp. 174–193). Berlin: Springer-Verlag. doi:10.1007/978-3-540-95888-8_7

Ledo, D., & Greenberg, S. (2013). Mobile Proxemic Awareness and Control : Exploring the Design Space for Interaction with a Single Appliance. In *Proceedings of the Annual SIGCHI Conference: Human Factors in Computing Systems - CHI '13*. Retrieved from http://hdl.handle.net/1880/49363

Lewis, J. (2002). Psychometric Evaluation of the PSSUQ Using Data from Five Years of Usability Studies. *International Journal of Human-Computer Interaction*, *14*(3), 463–488. doi:10.1207/S15327590IJHC143&4_11

Luderschmidt, J. (2013). The TwinTable – A Multi-Touch Tabletop System for Tangible Interaction. Retrieved from http://johannesluderschmidt.de/the-twintable-a-multi-touch-tabletop-system-for-tangible-interaction/2048/

Marquardt, N., & Greenberg, S. (2012). Informing the Design of Proxemic Interactions. *IEEE Pervasive Computing*, *11*(2), 14–23. doi:10.1109/MPRV.2012.15

Mattsson, S. (2011). *Measures of collaboration in CSCW*. University of Gothenburg.

McGrath, J. E. (1984). *Groups: Interaction and Performance*.

Microsoft. (2009). Windows Presentation Foundation. Retrieved from http://msdn.microsoft.com/en-us/library/ms754130%28v=vs.110%29.aspx

Microsoft. (2010). Visual Studio 2010. Retrieved from http://msdn.microsoft.com/en-us/library/dd831853%28v=vs.100%29.aspx

Microsoft. (2011). Surface 2.0 SDK. Retrieved from http://msdn.microsoft.com/en-us/library/ff727815.aspx

Microsoft. (2012). Kinect for Windows. Retrieved from http://www.microsoft.com/en-us/kinectforwindows/

Microsoft. (2014a). .Net at Build 2014. Retrieved from http://www.microsoft.com/net

Microsoft. (2014b). SharePoint – Team Collaboration Software Tools. Retrieved from http://products.office.com/en-us/sharepoint/collaboration

Microsoft. (2014c). Skype - Free internet calls and online cheap calls to phones and mobiles. Retrieved from www.skype.com

Microsoft. (2014d). Visual C#. Retrieved from http://msdn.microsoft.com/en-us/library/kx37x362.aspx

Microsoft. (2014e). Xbox One - Official Site. Retrieved from http://www.xbox.com/en-US/xbox-one#adrenalinejunkie

Microsoft News Center. (2010). Natural User Interfaces: Voice, Touch and Beyond. Retrieved from http://www.microsoft.com/en-us/news/features/2010/jan10/01-06cesnui.aspx

Microsoft PixelSense. (2012). Experience Things in a Whole New Way. Retrieved from http://www.microsoft.com/en-us/pixelsense/whatissurface.aspx

Milic-Frayling, N., Oleksik, G., Jones, R., Jetter, H.-C., Gerkin, J., Rieterer, H., & Baumberg, J. (2010). DeskPiles. Retrieved from http://research.microsoft.com/en-us/projects/deskpiles/

Mitra, S., & Acharya, T. (2007). Gesture Recognition: A Survey. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, *37*(3), 311–324. doi:10.1109/TSMCC.2007.893280

Moeller, J., & Kerne, A. (2010). Scanning FTIR: Unobtrusive Optoelectronic Multi-Touch Sensing through Waveguide Transmissivity Imaging. In *Proceedings of the fourth international conference on Tangible, embedded, and embodied interaction - TEI '10* (pp. 73–76). New York, New York, USA: ACM Press. doi:10.1145/1709886.1709900

Morris, M. R., & Winograd, T. (2004). Quantifying Collaboration on Computationally-Enhanced Tables. In *CSCW 2004 Workshop on Methodologies for Evaluating Collaboration Behaviour in Co-located Environments* (pp. 1–4).

Motion, L. (2014). Leap Motion - 3D Motion and Gesture Control for PC & Mac. Retrieved from https://www.leapmotion.com/product

Nielsen, J. (1994). Heuristic Evaluation. In J. Nielsen & R. Mack (Eds.), *Usability Inspection Methods* (pp. 25–62). New York: John Wiley and Sons.

Nielsen, J. (2012). Usability 101: Introduction to Usability. Retrieved from http://www.nngroup.com/articles/usability-101-introduction-to-usability/

Olivier, M. S. (2009). *Information Technology Research - A Practical Guide for Computer Science and Informatics (3rd ed)* (3rd ed.). Pretoria, South Africa: Van Schaik Publishers.

Patton, M. Q. (1999). Enhancing the Quality and Credibility of Qualitative Analysis. *Health Services Research*, *34*(5), 1189–1208.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, *24*(3), 45–77. doi:10.2753/MIS0742-1222240302

Penichet, V. M. R., Marin, I., Gallud, J. A., Lozano, M. D., & Tesoriero, R. (2007). A Classification Method for CSCW Systems. *Electronic Notes in Theoretical Computer Science*, *168*(Février), 237–247. doi:10.1016/j.entcs.2006.12.007

Phillips, R. (1997). 3.1.1 Incremental Prototyping. In *The Developer's Handbook to Interactive Multimedia* (pp. 37–38). Sterling, VA: Stylus Publishing. Retrieved from http://dl.acm.org/citation.cfm?id=522921

Pinelle, D., & Gutwin, C. (2000). A Review of Groupware Evaluations. In *WETICE '00 Proceedings of the 9th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises* (pp. 86–91). IEEE Computer Society Washington, DC, USA ©2000. Retrieved from http://dl.acm.org/citation.cfm?id=715503

Pinelle, D., & Gutwin, C. (2002). Groupware Walkthrough: Adding Context to Groupware Usability Evaluation. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '02* (pp. 455–462).

Postel, J. (1981). *Transmission Control Protocol*. Retrieved from http://tools.ietf.org/html/rfc793

Ramage, M. (1999). *The Learning Way: Evaluating Cooperative Systems*. Lancaster University. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.136.3869

RealtimeBoard Inc. (2014). Online whiteboard & online collaboration tool - RealtimeBoard. Retrieved from https://realtimeboard.com/

Reenskaug, T. (1979). Models-Views-Controllers. Xerox Parc.

Roschelle, J., & Teasley, S. D. (1995). The Construction of Shared Knowledge in Collaborative Problem Solving. In C. O'Malley (Ed.), *Computer Supported Collaborative Learning* (pp. 69–97). Berlin: Springe-Verlag. doi:10.1007/978-3-642-85098-1_5

Rosenthal, S., & Finger, S. (2006). Design Collaboration in a Distributed Environment. In *Proceedings. Frontiers in Education. 36th Annual Conference* (pp. 13–18). IEEE. doi:10.1109/FIE.2006.322529

Samsung. (2013). Samsung Galaxy S4. Retrieved from http://www.samsung.com/africa_en/consumer/mobile-phone/mobile-phone/smart-phone/GT-I9500ZWEXFA?pid=africa_en_home_thelatest_left1_galaxys4_20130426

Samsung. (2014). UA75ES9000R Series 9 Smart LED TV. Retrieved from http://www.samsung.com/za/consumer/tv-audio-video/television/led-tv/UA75ES9000RXXA

Sebit LLC. (2014). Interactive Whiteboards - Adaptive Curriculum. Retrieved from http://www.adaptivecurriculum.com/us/solutions/interactive-whiteboards.html

Seifried, T., Jetter, H., Haller, M., & Reiterer, H. (2011). Lessons Learned from the Design and Implementation of Distributed Post-WIMP User Interfaces. In J. A. Gallud, R. Tesoriero, & V. M. R. Penichet (Eds.), *Distributed User Interfaces* (pp. 95–102). London: Springer-Verlag. doi:10.1007/978-1-4471-2271-5_11

Smith, J. (2009). Introducing Thriple - A Library of 3D WPF Components. Retrieved from http://joshsmithonwpf.wordpress.com/2009/03/08/introducing-thriple-a-library-of-3d-wpf-components/

Sony Mobile Communications Inc. (2014). Xperia™ Tablet Z - Android Tablet - Sony Smartphones. Retrieved from http://www.sonymobile.com/global-en/products/tablets/xperia-tablet-z/

Stack Exchange. (2014). Stack Overflow. Retrieved from http://stackoverflow.com/

Tang, A., Pahud, M., Inkpen, K., Benko, H., Tang, J. C., & Buxton, B. (2010). Three's Company: Understanding Communication Channels in Three-way Distributed Collaboration. In *Proceedings of the 2010 ACM conference on Computer supported cooperative work - CSCW '10* (pp. 271–280). New York, New York, USA: ACM Press. doi:10.1145/1718918.1718969

Tangient. (2014). eCollaborating - Meetingworks. Retrieved from http://ecollaborating.wikispaces.com/Meetingworks.com

Tchankue, P., Wesson, J., & Vogts, D. (2010). MIMI : A Multimodal and Adaptive Interface for an In-Car Communication System. In *Proceedings of the 2010 SATNAC Conference* (pp. 55–60).

TeamSpeak Systems. (2014). TeamSpeak - Welcome to TeamSpeak. Retrieved from http://www.teamspeak.com/

Tee, K., Greenberg, S., & Gutwin, C. (2009). Artifact Awareness through Screen Sharing for Distributed Groups. *International Journal of Human-Computer Studies*, *67*(9), 677–702. doi:10.1016/j.ijhcs.2009.04.001

University of Wisconsin–Madison. (2007). Asynchronous vs Synchronous Communication. Retrieved from http://academictech.doit.wisc.edu/blend/facilitate/communicate

VideoLAN Organization. (2014a). libVLC. Retrieved from https://wiki.videolan.org/LibVLC/

VideoLAN Organization. (2014b). VLC Media Player. Retrieved from http://www.videolan.org/vlc/index.html

Villamor, B. C., Willis, D., & Wroblewski, L. (2010). Touch Gesture Reference Guide.

WhatsApp Inc. (2014). WhatsApp Home. Retrieved from http://www.whatsapp.com/

Wigdor, D., Penn, G., Ryall, K., Esenther, A., & Shen, C. (2007). Living with a Tabletop: Analysis and Observations of Long Term Office Use of a Multi-Touch Table. In *Second Annual IEEE International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP'07)* (pp. 60–67). Ieee. doi:10.1109/TABLETOP.2007.33

Wigdor, D., Williams, S., Cronin, M., Levy, R., White, K., Mazeev, M., & Benko, H. (2009). Ripples. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology - UIST '09* (pp. 3–12). New York, New York, USA: ACM Press. doi:10.1145/1622176.1622180

Wigdor, D., & Wixon, D. (2011). *Brave NUI World*. (R. Roumeliotis, Ed.) (pp. 3–227). Elsevier.

Yan, X., & Aimaiti, N. (2011). *Gesture-based interaction and implication for the future*.

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

Vice-Chairperson: Research Ethics Committee (Human)
Tel: +27 (0)41 504-2235

Ref: [H14-SCI-CSS-004/Approval]

Contact person: Mrs U Spies

24 June 2014

Dr D Vogts
Faculty of Science
Computing Scieœ
09-01-01F
South Campus

Dear Dr Vogts

USING NATURAL USER INTERFACES TO SUPPORT SYNCHRONOUS DISTRIBUTE COLLABORATIVE WORK

PRP: Dr D Vogts
PI: Mr T Potgieter

Your above-entitled application for ethics approval served at Research Ethics Committee (Human).

We take pleasure in informing you that the application was approved by the Committee.

The ethics clearance reference number is **H14-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

Prof CB Cilliers
Chairperson: Research Ethics Committee (Human)

cc: Department of Research Capacity Development
Faculty Officer: Science

Appendix B: Scenario and Task List

Scenario

You and your team were approached by a new retail video game store, called *Get-Your-Game-On*, to write a video game management system (VGMS) to be used in their store. The VGMS must effectively support the staff in managing the services offered by the store.

The video game store offers a variety of services including:

- Selling games
- Renting games
- Exchanging games
- Buying old games

The VGMS must support the above functionality and also be able to store, organize and manage all staff information, client accounts and merchandise details. For security purposes the system must have a password protected login function and it must allow users to lock the system.

Currently, you and your team have completed the Requirements document and must use it as an information source to continue into the Elaboration (Design) phase. For your upcoming weekly meeting with the Store Manager, you must design a logo for the store and create a few initial User Interface (UI) designs for the VGMS.

It's a Sunday and neither you nor your team feels like travelling to your usual meeting place. You all have access to a tablet and the Internet so you decide to work from home using GroupAware.

Task List

1. Start Up
   1.1. Create a new profile.
   1.2. Join the session.

2. Open Text Documents (1)
   2.1. Load and open the *Text Document* called "Team Tasks".
   2.2. Briefly read though the tasks and choose one member to cross them off as your team completes them.

3. Create Graphic Document (1)
   3.1. As a team, create one new *Graphic Document*.
   3.2. Secret Task: Go to the predetermined location and complete the biographical questionnaire. Then return to your device and complete the table at the end of the document.

3.3. Design a simple logo for the VGMS.
3.4. Rename the document to “Logo”.
3.5. Save and then close the document.

4. Open Text Documents (2)
    4.1. Load and open the *Text Document* called “Req Doc”.
    4.2. There are several spelling mistakes on each page. Choose a page (one member per page) and identify 3 mistakes by underlining, circling or highlighting them. When you are done, help your team members find the rest of the mistakes on their page*.

**Note: All annotations are automatically saved to the text document.*

5. Create Graphic Document (2)
    5.1. Navigate to the “Use Case Glossary”, found in the Req Doc, and choose a use case package (one member per package).
    5.2. Decide on a use case from your package to create an initial UI design for.
    5.3. Create a new *Graphic Document* and sketch a UI design for the use case.
    5.4. Make sure that there is consistency among all the UI designs.
    5.5. Rename the document to the same name as your use case chosen in 5.2.
    5.6. Save and then close the document.

6. Shutdown
    6.1. Disconnect from the server.
    6.2. Exit the application.

Secret Task

Identify the first 5 actions that occurred after you went away:

Number	**Action**		
	Who	**What**	**Where**
1.			
2.			
3.			
4.			
5.			

Appendix C: Biographical Questionnaire

(Please mark with an X where appropriate)

<table>
<tr><td>1.</td><td>Student Number</td><td colspan="20"></td></tr>
<tr><td>2.</td><td>Full name</td><td colspan="20"></td></tr>
<tr><td>3.</td><td>Age</td><td colspan="4">18 – 20</td><td colspan="4">21 – 29</td><td colspan="4">30 – 39</td><td colspan="4">40 – 49</td><td colspan="4">50+</td></tr>
<tr><td>4.</td><td>Gender</td><td colspan="10">Male</td><td colspan="10">Female</td></tr>
<tr><td>5.</td><td>Do you suffer from colour blindness?</td><td colspan="10">No</td><td colspan="10">Yes</td></tr>
<tr><td>6.</td><td>Computer proficiency</td><td colspan="7">Novice</td><td colspan="6">Intermediate</td><td colspan="7">Expert</td></tr>
<tr><td>7.</td><td>Computer experience (in years)</td><td colspan="5">0 – 2</td><td colspan="5">3 – 5</td><td colspan="5">6 – 9</td><td colspan="5">10+</td></tr>
<tr><td rowspan="2">8.</td><td>Have you used multi-touch hardware before?</td><td colspan="10">No</td><td colspan="10">Yes</td></tr>
<tr><td>If yes, which hardware?</td><td colspan="20"></td></tr>
<tr><td rowspan="2">9.</td><td>Have you used collaborative software before?</td><td colspan="10">No</td><td colspan="10">Yes</td></tr>
<tr><td>If yes, which software?</td><td colspan="20"></td></tr>
</table>

Appendix D: Post-test Questionnaire

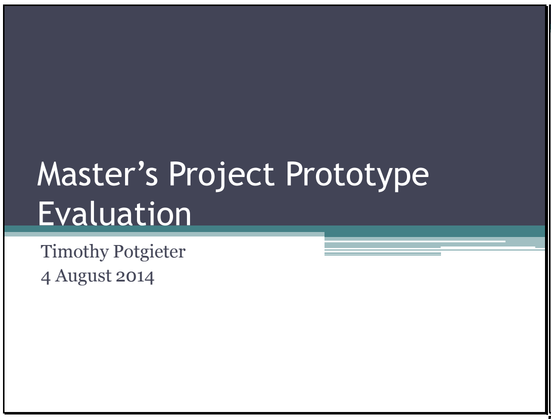
A. Overall Satisfaction								
1. Overall, I am satisfied with how easy it is to use this system.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
2. It was simple to use this system.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
3. I could effectively complete the tasks and scenarios using this system.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
4. I was able to complete the tasks and scenarios quickly using this system.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
5. I was able to efficiently complete the tasks and scenarios using this system.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
6. I felt comfortable using this system.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
7. It was easy to learn to use this system.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
8. I believe I could become productive quickly using this system.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
9. The system gave error messages that clearly told me how to fix problems.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
10. Whenever I made a mistake using the system, I could recover easily and quickly.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
11. The information provided with this system was clear.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree

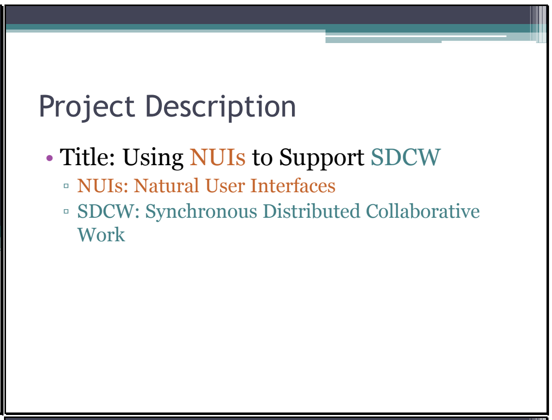
12. It was easy to find the information I needed.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
13. The information provided for the system was easy to understand.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
14. The information was effective in helping me complete the tasks and scenarios.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
15. The organization of information on the system screens was clear.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
16. The interface of this system was pleasant.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
17. I liked using the interface of this system.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
18. This system has all the functions and capabilities I expect it to have.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
19. Overall, I am satisfied with this system								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree

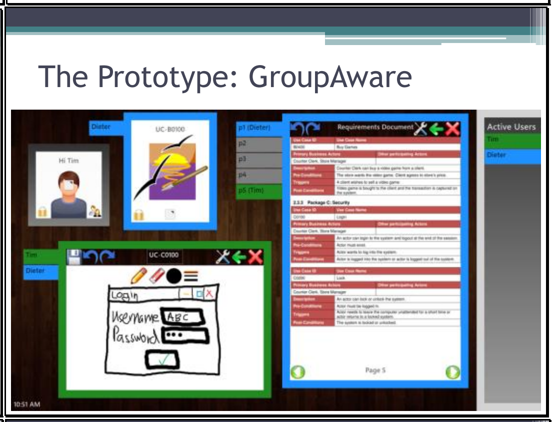
B. Collaboration								
a. Communication								
1. I could effectively communicate with my group members using this system.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
2. I could clearly hear what my group members were saying.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
3. My group members could clearly hear what I was saying.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
b. Coordination								
1. I was able to determine when my group members were available for collaboration.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
2. I was able to see what my group members were doing.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
3. My group members were able to see what I was doing.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
4. It was easy to divide work amongst my group members and I.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
5. I could effectively help my group members when they needed it.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
6. It was easy to find my group's previous actions.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
c. Information Sharing								
1. I was able to effectively access the shared work.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
2. I could easily share my work with the others.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree
3. If someone needed help, I was quickly able to access their work.								
Strongly disagree	1	2	3	4	5	6	7	Strongly agree

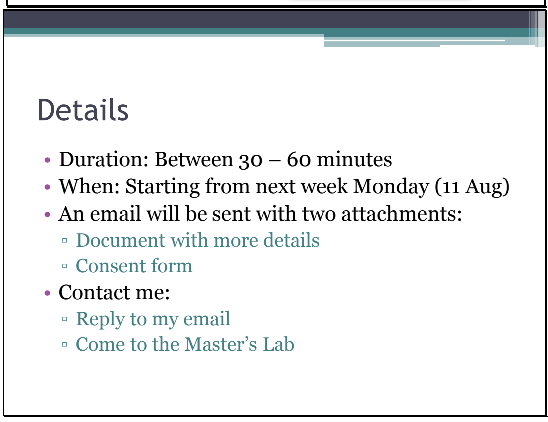
C. General Comments
Identify the best aspects of the system:
Identify the worst aspects of the system:
General comments about the usability of the system:
General comments about collaboration using the system:
Suggest any improvements to the system:

Appendix E: User Study Presentation









Appendix F: Written Information

Background

Synchronous Distributed Collaborative Work (SDCW) is when a group of people work together on a common goal at the same time from different locations. SDCW is possible because of groupware, a class of software systems that supports group work and processes. Developing groupware systems for SDCW is difficult because the physical distance between group members creates challenges in the three key areas of collaboration, namely coordination, communication and information sharing (Lanubile, 2009; Penichet et al., 2007).

The challenge of effectively coordinating group members and their tasks is of particular importance in SDCW because of the need for continuous group interaction. Existing SDCW groupware systems have implemented a number of effective techniques to overcome this challenge, but their low adoption rate of in the real world suggests that opportunities for further research still exist (Genest et al., 2013; Gutwin et al., 2008).

Natural User Interfaces (NUIs) are a recent development in the field of Human Computer Interaction (HCI). NUIs allow people to interact with technology in natural ways through touch, speech, in-air gestures and proxemics (Microsoft News Center, 2010). Subsequently, NUIs allow software systems to understand and respond to natural human interaction, which could provide new solutions to the challenges of group coordination faced by existing SDCW groupware systems.

This research aims to identify how effectively NUI interaction techniques can support SDCW. An NUI SDCW groupware prototype has been developed and will be evaluated in a user study. Below is a description of the scenario for the user study.

Scenario

You and your team were approached by a new retail video game store, called *Get-Your-Game-On*, to write a video game management system (VGMS) to be used in their store. The staff members will be using the system on the store's standard desktop computer. The VGMS must effectively support the staff in managing the services offered by the store.

Get-Your-Game-On offers a variety of services including:

- Selling – Sell games, consoles, and gaming accessories to their clients
- Renting – Rent out games and consoles to their clients for a certain time
- Exchanging – Exchange games with their clients
- Buying – Buy games from their clients

The VGMS must support the above functionality and also be able to store, organize and manage all staff information, client accounts and merchandise details, i.e. games, consoles and gaming accessories. For security purposes the system must have a password protected login function and it must allow users to lock the system, i.e. disable all functionality without logging the user out.

Currently, you and your team have completed the *Requirements* document and must use it as an information source to continue into the *Elaboration* phase. The essential outcomes of the Elaboration phase include the following:

1. Analysis Use Case Model
2. User Interface (UI) Designs
3. Detailed Use Case Narratives
4. Class Diagram
5. Database Design

It's a Sunday and you have a deadline for tomorrow. You and your team have to complete the first two of the above outcomes. Neither you nor your team has transport to your usual meeting place. You all have access to a tablet and the Internet so you decide to work from home using GroupAware.

Appendix G: Consent Form

NELSON MANDELA METROPOLITAN UNIVERSITY

INFORMATION AND INFORMED CONSENT FORM

RESEARCHER'S DETAILS	
Title of the research project	Using Natural User Interfaces to Support Synchronous Distributed Collaborative Work
Reference number	H14-SCI-CSS-004
Principal investigator	Timothy Potgieter
Contact telephone number (private numbers not advisable)	041 504 2094

A. DECLARATION BY OR ON BEHALF OF PARTICIPANT		Initial
I, the participant and the undersigned	(full names)	

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 Potgieter	
from	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 researchers are studying how Natural User Interface (NUI) interaction techniques can support synchronous distributed collaborative work (SDCW). The information will be used for research purposes.			
2.2	Procedures:	I understand that I am required to use a system to evaluate NUI interaction techniques for SDCW. I understand that an overhead camera will be recording the evaluation.			
2.3	Risks:	I understand that there are no risks involved in participating in this study.			
2.5	Confidentiality:	My identity will not be revealed in any discussion, description or scientific publications by the researchers.			
2.6	Access to findings:	Any new information or benefit that develops during the course of the study will be shared in the dissertation on the research, which will be available from the Computing Science Department.			
2.6	Voluntary participation / refusal / discontinuation:	My participation is voluntary	YES	NO	
		My decision whether or not to participate will in no way affect my present or future class marks / care / employment / lifestyle	TRUE	FALSE	

3.	No pressure was exerted on me to consent to participate 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.	

A.2 I HEREBY VOLUNTARILY CONSENT TO PARTICIPATE IN THE ABOVE-MENTIONED PROJECT:		
Signed/confirmed at PORT ELIZABETH	on	2014
Signature	Signature of witness:	
	Full name of witness:	