MXit Web Applications:
The Design of a Web Interface to facilitate the creation of MXit services by teachers

Jared Cameron Baboo

BBXJAR001

Supervised by Gary Marsden

November 16, 2009

Abstract

The Mobile instant messaging applicaiton MXit has been shown to be successfully used as a tool to provide learners with educational resources. With Mxit allowing developers the ability to contribute clients and content it is now possible to create various types of content for the MXit service. This project aims to investigate a means to tap into teachers expertise to allow the development of more learning services for Mxit.

To achieve this a web based interface was developed that allowed teachers to create and supply content to MXit effectively empowering teachers to create custom clients and services for Mxi. This was achieve without forcing teachers to gain the technical competence normally needed to create such services. This Facilitated the development of services for MXit by teachers who have little to no programming experience.

Acknowledgments

I would like to thank our project supervisor Gary Marsden, Masters Students Ian Saunder,Sean Packham,Mathew Slade and Graeme Smith, the Teachers of the respective schools, the members of the 2009 Computer Science Honours class and proffesor Edwin Blake for encouraging me to continue with the course. A special thanks to all the friends who supported me when I was hospitalise Jessica Gordon, Lorin Zaal and Poovendren Govender.

Contents

Abstract i

Acknowledgements ii

Table of Contents iii

1 Introduction **1**

1.1 Problem Outline 1

1.2 Proposed Solution 1

1.2.1 System Objectives 1

1.3 Aim of Research 2

2 Background **3**

2.1 Mxit 3

2.2 End User Programming 3

2.3 User Centred Design 4

2.3.1 Ethnography 4

2.3.2 Participatory Design 4

2.3.3 Interaction Design 4

2.4 Dynamic Websites 5

2.5 Usability 5

2.6 Evaluation 5

3 Design **7**

3.1 Methodology 7

3.2 Participants 7

3.3 Ethical Considerations 8

3.4 Ethnography 8

3.5 Constraints 9

3.6 Usability Goals 9

3.7 Low Fidelity Prototype 10

3.7.1 Setting 10

3.7.2 Session Goals 11

3.7.3 Procedure 11

3.8 Results . . . 11
3.8.1 Issues . . . 13
3.9 Proposed Applications . . . 13
3.9.1 Course Management Tool . . . 13
3.9.2 Messaging service . . . 14
3.9.3 Revision service . . . 14
3.9.4 School Blog . . . 14
3.10 Usage scenario . . . 14
3.11 Design of Intermediary Representation . . . 15
3.12 High Fidelity Prototype . . . 16
3.13 First Iteration . . . 16
3.13.1 User Feedback . . . 18
3.13.2 Initial Heuristic Evaluation . . . 18
3.14 Second Iteration . . . 19
3.15 Results of Redesign . . . 20

4 Implementation **22**
4.1 System Overview . . . 22
4.2 Development . . . 22

5 Summarative Evaluation **25**
5.1 Usability Testing . . . 25
5.1.1 Participants . . . 25
5.1.2 Setting . . . 25
5.1.3 System and Tools . . . 25
5.1.4 Aims . . . 25
5.1.5 Procedure . . . 26
5.1.6 Expected Outcomes . . . 26
5.2 Expert Evaluation . . . 27
5.2.1 Participants . . . 27
5.2.2 Setting . . . 27
5.2.3 Aims . . . 27
5.2.4 Procedure . . . 27

5.2.5 Expected Outcomes 28

6 Conclusion **29**

6.1 Conclusion 29

6.2 Implications 29

7 Future Works **30**

A Interview Questions **35**

B Planned Tasks **36**

C Generalised IR **38**

Samples Questions from Questionaire for User Interaction Satisfaction **38**

List of Figures

1 System Strucutre and inital plan 12

2 Paper screen 13

3 System Use Case diagram 15

4 Iteration one example 17

5 Iteration one popup 18

6 Heuristic feedback 19

7 Improvements image one:Initial Screen 20

8 improvements image two: Screen naming 21

9 Improvements image three: Colour and Improved tree structure 21

10 MVC model 23

11 Interative model 24

1 Introduction

1.1 Problem Outline

As mobile devices become more and more prominent it becomes understandable why a mobile applications such as MXit could become as popular as it has[28]. With the popular Mobile application MXit allowing developers access to their messaging API it is clear that the development of custom MXit clients is encouraged[10]. However this only allows developers to create these custom clients and excludes individuals such as teachers who may have lower technical expertise. By giving teachers the ability to create custom clients and services one could tap into their expertise to facilitate the development of a mobile learning service for MXit. We seek to develop a means to allow allows teachers to create custom clients for MXit that offer services to MXit users. We aim to empower teachers by giving them the ability to define new content or services for MXit.

1.2 Proposed Solution

In order to determine how to develop an interface which allows teachers to create custom MXit clients and services we propose the development of a web based interface. We propose to design this interface by following human computer interaction principles and following a user centred design methodologies. using these methodologies and following these principles we aim to gain an understanding of the user and the content they would wish to place on MXit. Gaining insights into the user would allow us to better design an interface the user is comfortable with using while gaining an insight into the types of content would help define the required functionality of the system.The proposed interface should produce an intermediary representation of the teachers input which could be used to produce a MXit client that could connect to the MXit cloud and offer services to MXit users. MXit users then could add the created client as a contact giving them access to the services and information as specified by the teachers.

1.2.1 System Objectives

- Empower teachers with the ability to define and create m-learning content
- Offer a means for teachers to use mobile devices to assist learners with their school

work

- Make use of an existing mobile application to facilitate the development of learning tools
- Allow learners to use technology they are familiar with to help with their school work

1.3 Aim of Research

We aim to investigate how to create an interface that allows teachers to create clients for MXit which can be accessed by high school learners. We will investigate the types of content teachers would expect to be able to create as well as the potential uses teachers. To this end experimentation focused on the interface that was designed aiming to understand how to design a usable interface for the design of content for the mobile instant messaging application MXit.

2 Background

2.1 Mxit

MXit is an popular instant messaging application for mobile devices and computers[28, 3]. With the release of the MXit messaging API developers have gained the ability to create custom clients for MXit users. The benefit of the release of the API is that clients that can offer various services to MXit users can now be developed. One such service is a project called Dr. Math .This project provides a learning environment focused around mathematics using MXit. This has been shown to produce some successful results indicating that mobile devices can effectively be used as a learning medium[4, 3, 5].

An issue the release of the MXit API has for the development of services for MXit is that it requires technical competence with developing software. However teachers who may benefit from specifying their own custom clients may not possess this competence.

2.2 End User Programming

Research indicates that there is an occurrence of end users creating custom solutions that suit their specific needs. Indications are that this may be inevitable as the use of computers increases implying that developers should attempt to make this as effortless an event for end users as possible[7, 25, 19].

Methods and tools used to facilitate this include the use of analogy in the design of end user programming interfaces[25]. This would allow one item to behave in the same way as another one. Another possible tool is the use of a visual interface. There are indications that visual interfaces can facilitate end user programming however they may too prove a hinderence[19].

Visual programming interfaces may be a means to facilitate the development of applications by non programmers. This is achieved by annotating the programming process or by using a visual representation[19, 26]. An issue with this is that the developed interface may be domain specific limiting it's ability for reuse [26]. Also the manner in which

information is visualise may be inappropriate for the task at hand. However should the visualisation be well suited to the given task it can improve user comprehension and efficiency. Also indications are that when programming or developing mental models are used by developers which could be facilitated by the use of a visualisation [23].

2.3 User Centred Design

2.3.1 Ethnography

Ethnography is the descriptive tool which is used to study people and societies through observations [12]. This tool is useful as it allows the study of a situation allowing interactions to be noted and requirements to be extrapolated [12].

2.3.2 Participatory Design

To facilitate user centred design participatory design aims to be as democratic as possible. Participatory design aims to facilitate interaction between developers, designers and stakeholders. This is beneficial as it facilitates knowledge gathering for the developers and empowers users by allowing them to make design decisions [18]. Issues arise with this methodology as it may not be possible to have all stakeholders present and the tools or methods used may be inappropriate or disruptive [15]. An example of a tool used in participatory design is PICTIVE. This combines low tech objects and high tech recording tools[17].

2.3.3 Interaction Design

Interaction design focuses on people and their interactions with systems. Through studying peoples interactions with objects and systems an understanding of issues is elicited which can guide the further design of more usable interfaces [8, 6]. This has been shown to be an effective method of improving interfaces by it's use in the Jam-O-Drum Interactive Music System [2]. This further shows that interaction design can be used to generate alternative results based on requirements that can be extrapolated from observations [6].

2.4 Dynamic Websites

A web application can be defined as a system based on World Wide Web Consortium (W3C) defined standards and technologies that offers services and content through a interface such ad web browser[14]. Some technologies outlined by the W3C include HTML(Hypertext Markup language), XML(Extensible Markup Language), Javascript(A client side scripting language) , PHP (Hypertext PreProcessing) and Cascading Style Sheets(CSS)[11].

Web applications can create dynamic content on the fly from sources of data which could include user input. To facilitate the development of more interactive and dynamic web applications combinations of Cascading Style Sheets, Javascript(client side script) and Document Object Model(DOM) can be used[22].

2.5 Usability

To facilitate the design of a usable system there are many heuristics or guides a developer can use. The aim of these are to provide a suitably usable interface to facilitate user interactions. A usable interface is one that provides the user with a positive experience. Tools that designers can use include affordances which are the way objects appear to encourage interaction[13, 16]. Another too are heuristics such as Nielsen's ten heuristics for interface design. This heuristic outlines ten guides which could facilitate the design of a usable interface [20]. Another example of tools which can be used to facilitate the design of a usable interface is the visual information seeking mantra. This further defines a guide to design visual interfaces based on discoveries of visual information system [1].

2.6 Evaluation

Various methods and techniques exist for evaluating intrefaces which take into account participants as well as the setting of th evaluation.Evaluations can be classified as being either formative or summarative. Formative evaluations are used during the development of an interface to ensure that the users needs are continually met. Summarative evaluations are completed on the final product to assess the success of a product. Depending on the participants who partake in the evaluation different methods and results may be collected. When users evaluate the system quantitative data about the performance of the system, errors and users opinions are collected. To achieve this users would be asked to complete

a set task in a controlled environment. Expert evaluators can also be used to gather qualitative data about perceived errors and possible solutions [13, 24].

3 Design

3.1 Methodology

In order to design the intended solution an agile software development methodology, SCRUM, was decided upon. This methodology allowed for the development to take place following an iterative and incremental life-cycle[27].An iterative and incremental development cycle is used to facilitate the addition and evaluation of features as the need or desire for these features emerges [27]. The design of the interface also followed the User Centred Design methodologies. User centred design was used to gain a better understanding of the teacher and their tasks so as translate this into an effective solution. This is achieved by involving the user, in this case teachers, early in the design process.

3.2 Participants

The participants of this study were teachers from two schools in the athlone area of cape town. From the ethnographic study their technical skills ranged from minimal technical skills to advanced user. Participants also taught a myriad of subjects from life skills to mathematics and accounting. Two different schools were used in the study to allow for diverse groups of teachers to contribute towards the project.

Introduction to the school was through the principal for one school and a teacher for another. This indicated both a top down and bottom up approach with the Principle interested in the project to drive school improvement and the teacher interested in the project as a means to assist learners. The school whose principal had encouraged the project handed over to the Life Sciences teacher to act as liaison to facilitate teacher involvement. The top down approach (initiating contact through the principal) appeared to be undesired by the teachers who appeared somewhat reluctant to partake in the project compared to the bottom up approach (being driven by the teachers).

3.3 Ethical Considerations

Ethical issues that arise in the design of this system is initially the voluntary assistance of the teachers. It was ensured that participation in the design of the interface was voluntary and teachers were not forced to partake in any activities they felt uncomfortable with.

Items that arose as possible issues with the use of the system was the number of students who had access to a phone and used MXit. Should not all children have access to the content and services created the use of this system as a school specific tool could be limited as the lack of access for some students could create a culture of exclusion within the schools. Also as there is some negative stigma attached to MXit some learners and their parents may be opposed to the use of MXit and be disinclined to install MXit. This should serve as a reminder that MXit is a application that is installed voluntarily and all contacts are added voluntarily. This implies that even though teachers may create content it would be up to the students to take the initiative and access the materials.

Further issues arise with the specific content that is placed on MXit through this system. This system has the potential for abuse through the creation of content that is inappropriate or abusive such as being insulting towards a person or group of peoples. This implies that content has to be monitored to ensure only appropriate content is placed on MXit also this would be done to ensure the system does not become abused.

Teachers were informed that the MXit application was a voluntary application and that learners should not be pressured into installing MXit in order to use these services. Teachers were also made aware that the services would be accessible to all MXit users who would need to add the service as a contact. MXit also supports safe use of it's software and is somewhat restrictive about the type of content added[28, 10].

3.4 Ethnography

To gain an initial understanding of the teacher an ethnographic study was performed on the teachers work environment. This was augmented by group discussions and interviews with individual teachers. The ethnographic study indicated that teacher are methodical

individuals who work in a social and co-operative environment. From this study, the interviews and discussion initial constraints on the solution were discovered as well as the initial uses and expectations of the teachers.

3.5 Constraints

Time Teachers are constrained by time meaning that the solution must make their lives easier and not take up too much time citing the time it would take to create a client as too taxing on them. This implies that the interface must be easy to use and efficient.

Technological experience The teachers appear to have a range of skills with some teachers being technologically component while others lacking confidence in their technological skill.

Screen Size of Target Device While the creation of the actual content would be done on a regular computer screen care must be taken to remind teachers that the target device on which the content would be displayed was much smaller than the interface they were using.

Suitable Subjects From the discussion and interviews it appears that some subjects are not very well suited for an m-learning environment. Languages may prove difficult to provide revision tools for as they may require extensive spelling and grammar facilities. Subjects such as accounting which require large amounts of data to be processed by the students also could have difficulty in using any revision tolls we provide.

3.6 Usability Goals

As this section of the project focused on interface design usability becomes an issue that would require attention. Some goals that would be targeted[24].

effectivness The primary goal of this system are to be able to produce a working MXit client that has functionality and content as specified by the teachers through the interface.

Develop an intuitive interface Should the interface that is developed not be intuitive or easy for teachers to use they effectively may not be able to use it and the interface loses it's effectiveness.

Develop an Easy to learn interface As this solution presents the teachers with new ways of thinking the interface should be easy to learn and understand so that the process of creating new clients is not an arduous task.

Develop an interface that is memorable Should a teacher create or modify a custom client in the future the interface should be designed in a way that Promotes recognition so that minimal time is spent having to relearn how to perform tasks.

Develop an efficient interface Teachers are extremely busy limiting the time they have to create custom MXit clients therefore the implemented solution should allow teachers to create a client as fast as possible.

Safety A goal of this system would be to reduce errors but also not infringe on the teachers or learners in any way. Mxit encourages safety and safe use of MXit through non disclosure of personal information and care must be taken to ensure this in the system.

3.7 Low Fidelity Prototype

In order to design the interface for the proposed solution a low fidelity paper prototype was developed. Through the design of this paper prototype design ideas and usage scenarios were generated and refined. In order to facilitate the development of this prototype and the generation of ideas the session to develop the prototype was done in a familiar and comfortable environment for the teachers.

3.7.1 Setting

To increase comfort and ease for the participants(Teachers) the session was held at their respective schools. In order to represent the computer screen a A3 page was used with sticky notes and pencil drawings used to represent interface elements. These were used as as they allowed the prototype to be easily changed as ideas were generated.

3.7.2 Session Goals

- Determine the technical ability of the given sample of teachers
- Introduce concepts around MXit ,mobile learning and MXit bots to teachers.
- gain an understanding of the type of content teachers are likely to place on MXit through the creation of the custom clients
- Generate ideas for the design of the interface
- Determine what expected feedback and assistance was expected from the user

3.7.3 Procedure

As the session would introduce new concepts to teachers involving MXit and mobile learning an introductory session was arranged to answer any initial questions about these concepts. As the introduced concepts were new to the participants and the computer literacy levels varied a question and answer approach was used to determine functionality and guide the session.

In order to facilitate the generation of ideas and at the request of the participants an example site was shown displaying possible available functionality of a website. To give participants an idea of the possible functionality and to limit influence any custom site would have on the design a jquery user interface demo site was shown to the participants to demonstrate the possible functionality a website could contain. From this ideas were generated and tested using the paper prototype. This facilitated the generation of ideas. By involving the users in this process it was hoped that the users would gain a sense of ownership of the resultant system as they would have actively taken part in the design of the interface.

3.8 Results

Through this session an understanding of user requirements were gained as well as proposed applications of the system. Paper prototypes produce were used as a guide to produce high

level prototypes as well as to gain understanding on how a user might interact with the system.

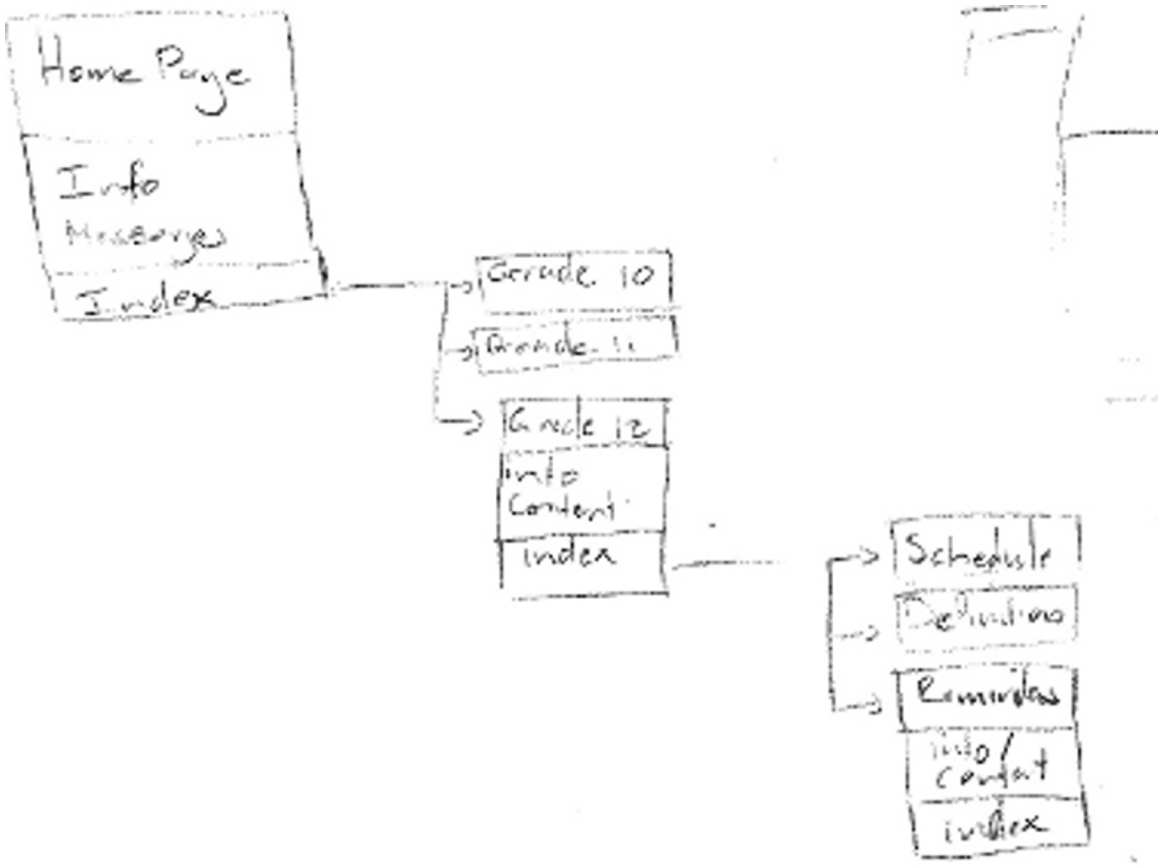

Figure 1: A pencil and paper example of the overall system as teachers initally envisioned

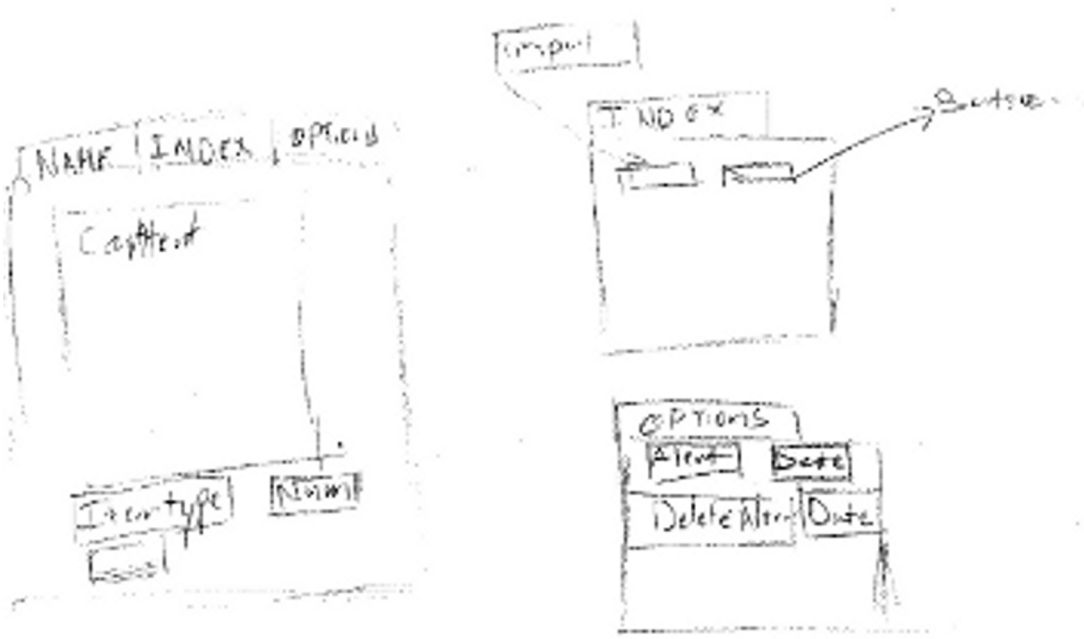

Figure 2: A pencil and paper example of the screens envisiaged

3.8.1 Issues

An issue that later arose as a result of this session was a lack of focus on affordances as this was overlooked by the development team. This resulted in an initial design that lacked effective affordances that are easily understood by teachers limiting the effectiveness of the design.

3.9 Proposed Applications

3.9.1 Course Management Tool

Teachers envisaged the use of this service to manage courses and sports timetables for the students using MXit to remind students of scheduled class tests, sporting and schooling events. Teachers also envisioned using this technology to set homework and give students access to online materials.

3.9.2 Messaging service

Teachers envisioned further uses of the technology as a messaging tool to learners and their parents. Acting as a reminder system for learners alerting them to upcoming events such as tests and homework due dates as well as an informative system to the learners parents allowing them to know when the learners had homework assigned to them.

3.9.3 Revision service

Teachers also saw benefit in the project in providing learners with a revision tool allowing them to practice or revise sections of school work. This would be in the form of an assessment feature which learners could access. These assessments would allow the learner to revise a set of work through the completion of a set of tests or by looking up some definitions.

3.9.4 School Blog

Teachers also saw the possibility of using this system as a blog to inform learners and their parents of activities. This could serve a similar function as the messaging applications but in the sense of also recording past happenings similar to a newsletter or be used on a more personal level allowing teachres to create personal blogs which learners could follow.

3.10 Usage scenario

Mrs Smith is a life sciences teacher at a school in the souther suburbs of cape town south africa.To help learners in their studies Mrs. Smith researches and finds that they can create learning content similar to Dr. Math on MXit using MXit Web Applications. She prepares the content she wishes to place on MXit creating assessments that she might set for revision work along with answers to the questions in the assessment. She also prepares sets of definitions which the classes she teaches cover or are expected to know. She also prepares any advice that she thinks might benefit learners in their studies. She then takes the prepared material and goes to the staff room or computer room at school and using the schools computers goes to the MWA website. She transfers the prepared materials to the web based interface and clicks submit this produces a confirm prompt indicating that a application has successfully been created. The interface produces a client hosted on a

server which Mrs. Smith informs her classes about allowing interested learners to add the created client as a contact.

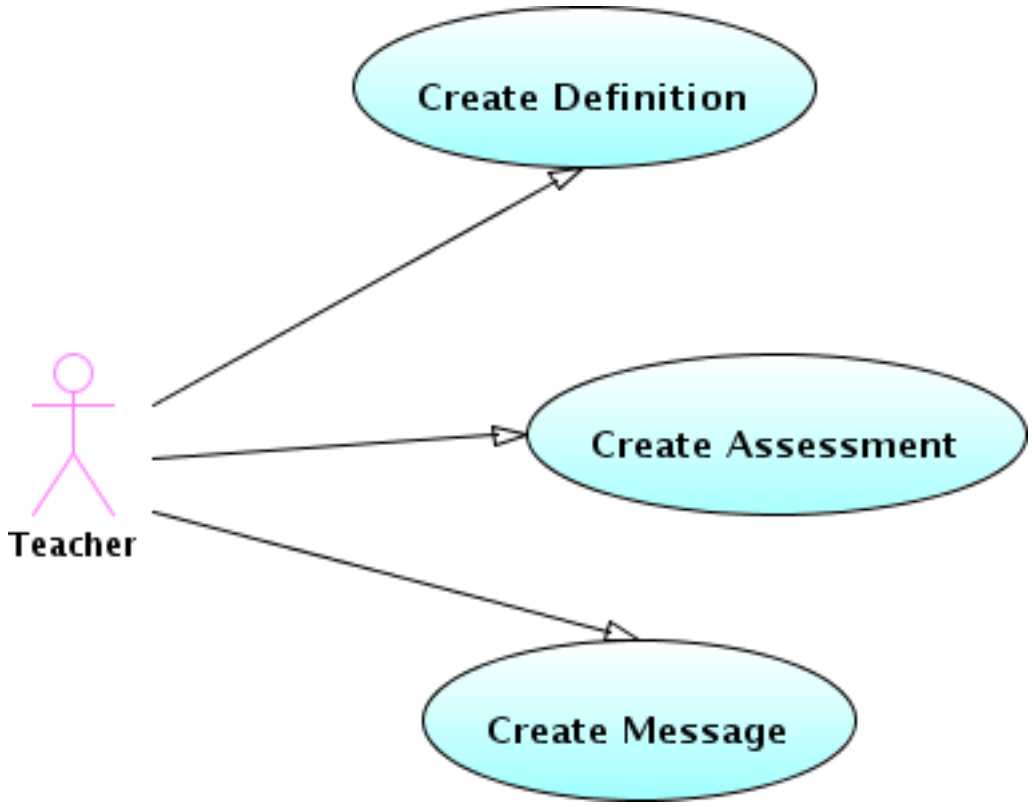

Figure 3: A use case diagram representing possible uses of the system

3.11 Design of Intermediary Representation

Once the requirements of the software had become understood through the paper prototyping session an intermediary representation was designed. This was designed to use a well formed markup language to facilitate future recall of previously created content. This representation would be primary point of communication between the two developed subsystems and as discussion between developers during the design of the IR ensured that both parties were aware of the specifications of the IR.

3.12 High Fidelity Prototype

Following the development of the low fidelity prototype a high fidelity prototype was implemented. This prototype was an implementation of the low fidelity prototype that encompassed as much of the desired user requirements as time allowed. This was implemented following an incremental development life-cycle so as to allow for changing requirements.

3.13 First Iteration

The priority of the first iteration was to produce a functioning interface that allowed the user to construct the desired functionality of the clients. As the aim of the project was to investigate how to design an interface that allowed the creation of custom MXit clients the initial focus of the implementation was the interface allowing the creation of new content. To achieve this the interface was implemented with an aim to be as simple and easy to use as possible.

The first iteration produce a web page that allowed a teacher to add various content to a MXit bot. The interface allowed for the generation of three primary types of screens which would be presented to a learner through MXit. The three screens were classified as being a Generic MXit Screen, Assessment Screen and Definition Screen. From these primary screens secondary screens could be created. The Generic MXit Screen could spawn other primary screens while the Assessment Screen could be used to generate Question type screens and the Definition Screen Allowed the creation of Definitions or formula.

To facilitate the facilitate the development of a user friendly interface the high fidelity prototype was implemented with Nielsen's ten heuristics in mind. This iteration experimented with the use of pop up and hover over features to investigate the User reaction to these. These were investigated to act as accelerators and help guides to facilitate user interactions. As a result of earlier ethnographic study insight was gained which helped determine the correct terminology to use to facilitate user interactions.

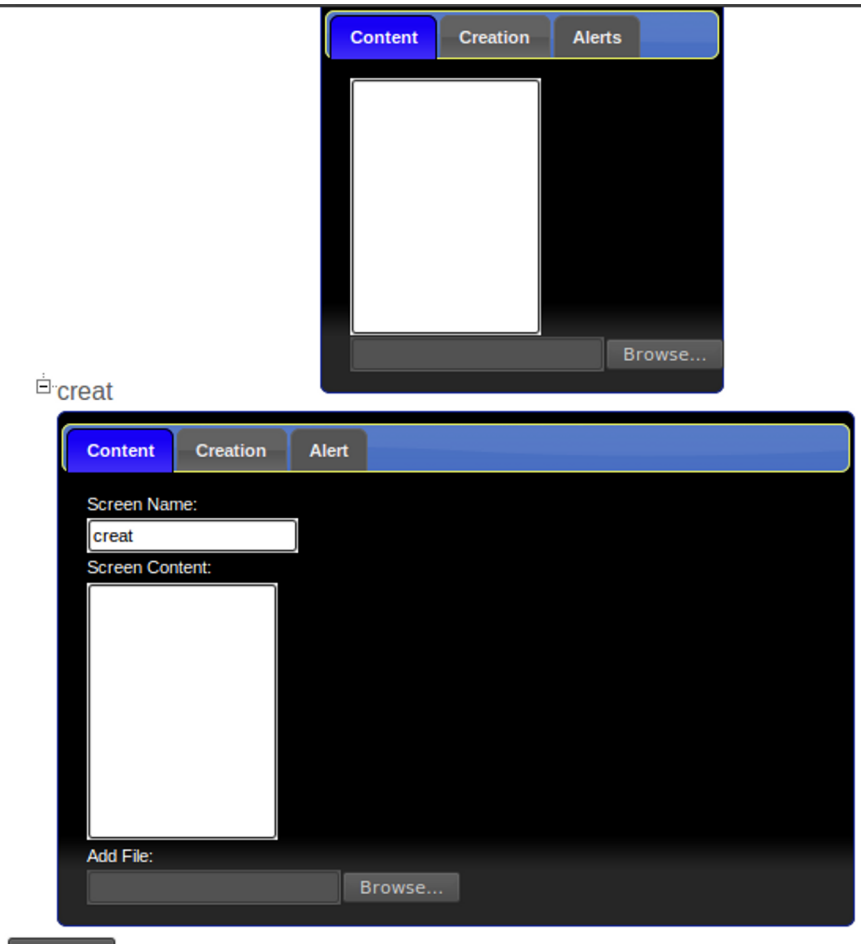

Figure 4: An example from iteration one showing an inefiecient use of screen space.

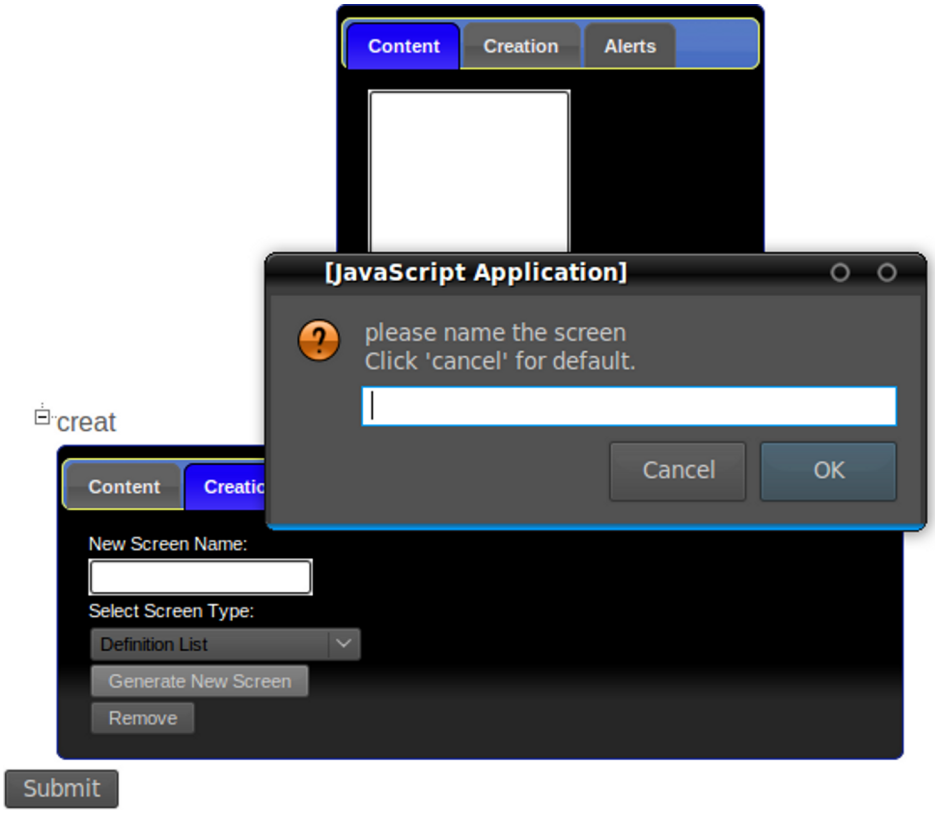

Figure 5: An example of popups used during iteration one

3.13.1 User Feedback

Since teachers formed an active component in the design of the interface they were kept informed as to progress of the implementation. As the implementation of the high fidelity prototype proceeded the teachers were allowed to interact with the prototype in order to determine issues and to influence design decisions. From the teachers interactions with the system an initial problem which arose was the correct use of terminology as well as the ease of use of the first high fidelity prototype. Another issue which arose was a sense of uncertainty of the user when using the interface however it was unclear as to specific causes of uncertainty.

3.13.2 Initial Heuristic Evaluation

To ensure adherence to Nielsen's heuristic's an initial evaluation was completed by an expert evaluator. The secondary purpose of this evaluation was to uncover any hidden issues and test usability of the system by individuals not associated with the design and

development of the system.

The result of the this evaluation indicated numerous errors in design. Specifically the design did not allow individuals without prior knowledge of the system ie. people who did not partake in the initial design an easy or intutitive experience as the purpose or goal of each section was initially unclear. These indicated a failure to fully and effectively incorporate the guidelines as specified by Nielsen. Another issue that this evaluation brought to light was the overlooking of effective implementations of affordances. As a result of this evaluation further understanding was gained as to the cause of uneasiness or uncertainty as observed when the teachers interacted with the system. Recomended by the evaluator was the use of a guide or wizard style interface to facilitate the development of services by new users.

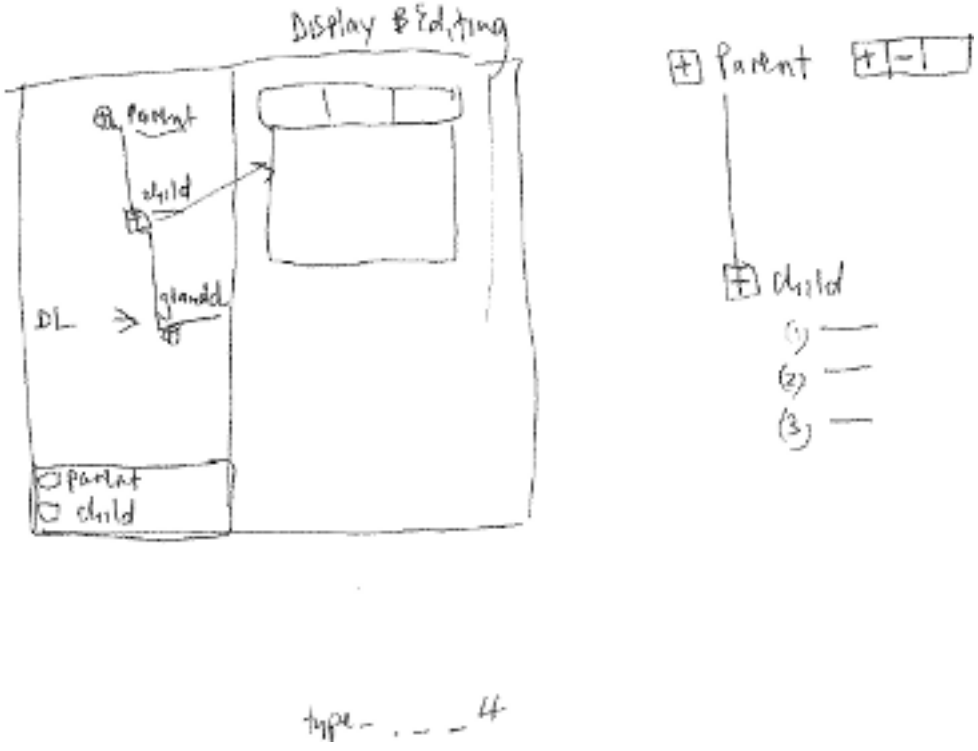

Figure 6: A simple sketch of suggestions from expert evaluators

3.14 Second Iteration

Due to the errors uncovered through the first prototype the second iteration involved a reimplementation of the paper prototype incorporating features tested within the first prototype as well as focusing on the incorporation of affordances and adhering to heuristics.

In this iteration the visual information seeking mantra was followed so as to facilitate the development of a more user friendly interface.

In order to rectify issues which arose during the first iteration and improve overall user experience improvements were made to the interface to:

- Reduce possible errors
- improve client side validation
- focus on and improve affordances
- add guidance for the creation of new screens
- increase adherence to heuristics
- adhere to visual information seeking mantra

3.15 Results of Redesign

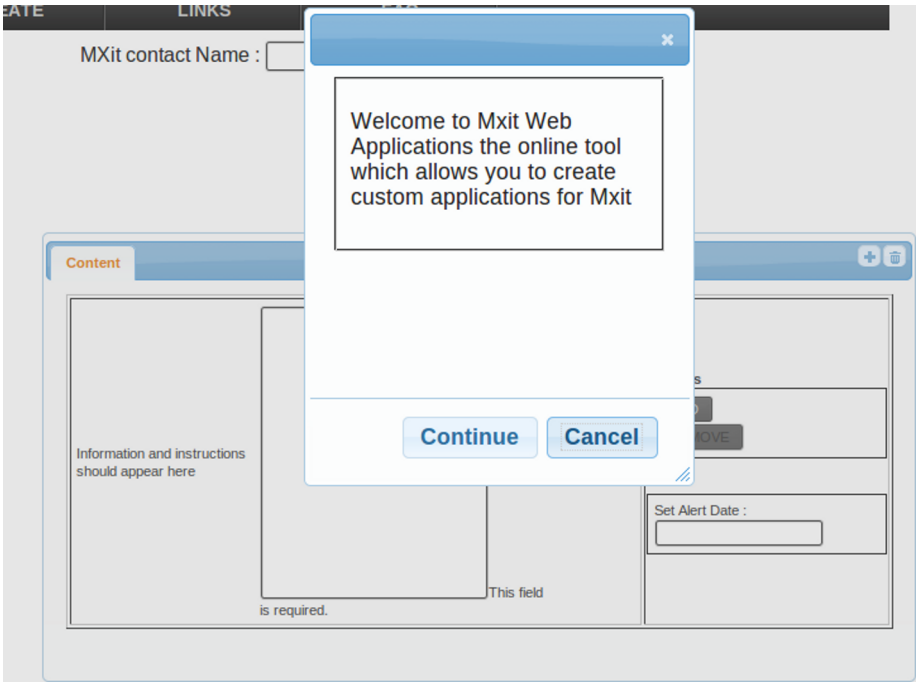

Figure 7: New Initial screen providing some initial information to the user

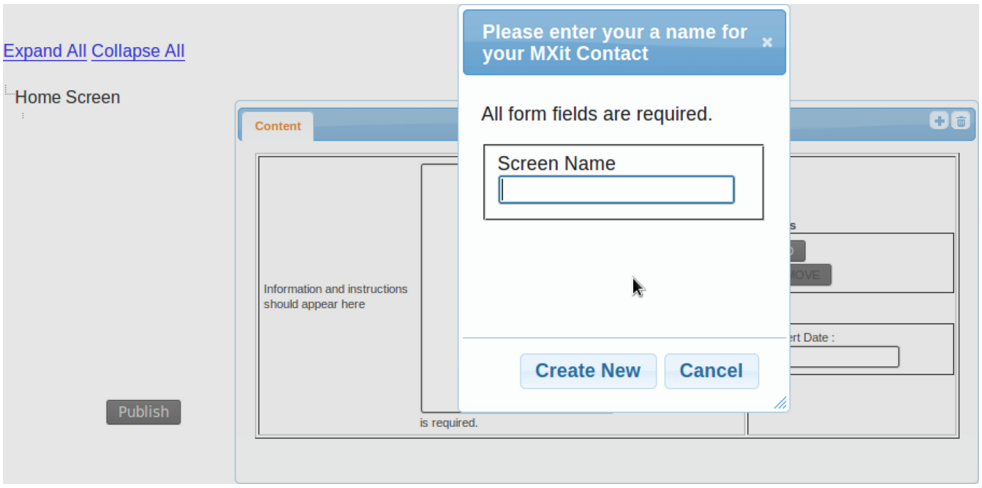

Figure 8: Improvements to guide user interactions

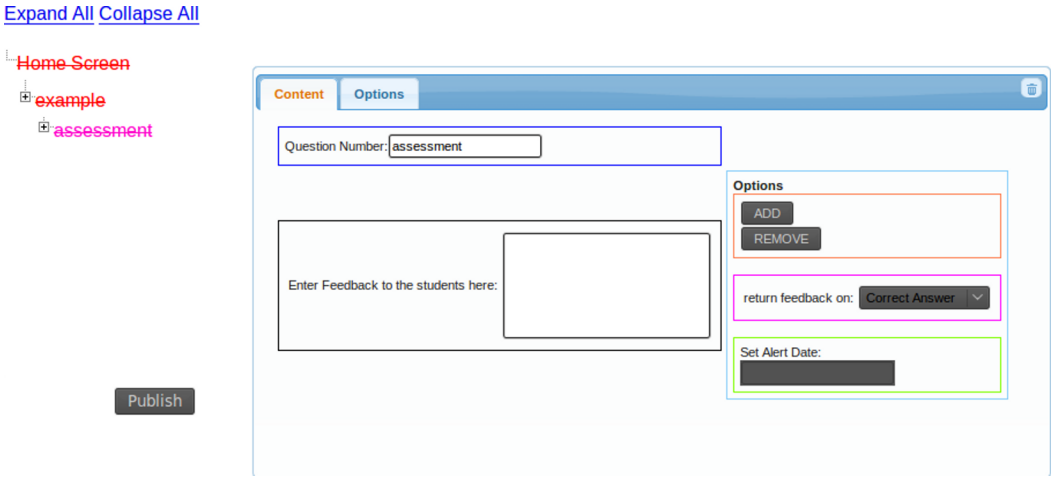

Figure 9: Colour was used to assist users in identifying types of screen while an improved structure and positioning of the tree menu.

4 Implementation

4.1 System Overview

The implemented solution contained two sub-systems, namely a web based front end and a server side back end. To facilitate communication between front end and back end an intermediary representation was designed which took the form of a well formed XML document. This document was then used by the back end to create a custom MXit client which could connect to the MXit cloud allowing MXit users to connect to the created client. The custom client would then be able to offer services as defined in the XML document to MXit users.

4.2 Development

Overview and Method

To facilitate the creation of a modular system adherence to the Model View Controler pattern was enforce. A modular implementation was important so as to minimise the impact of changes on various aspects of the system. The MVC pattern was chosen as it allowed for the creation of multiple interchangeable views which could interact with a common model while allowing for the model to be changed without effecting the view.

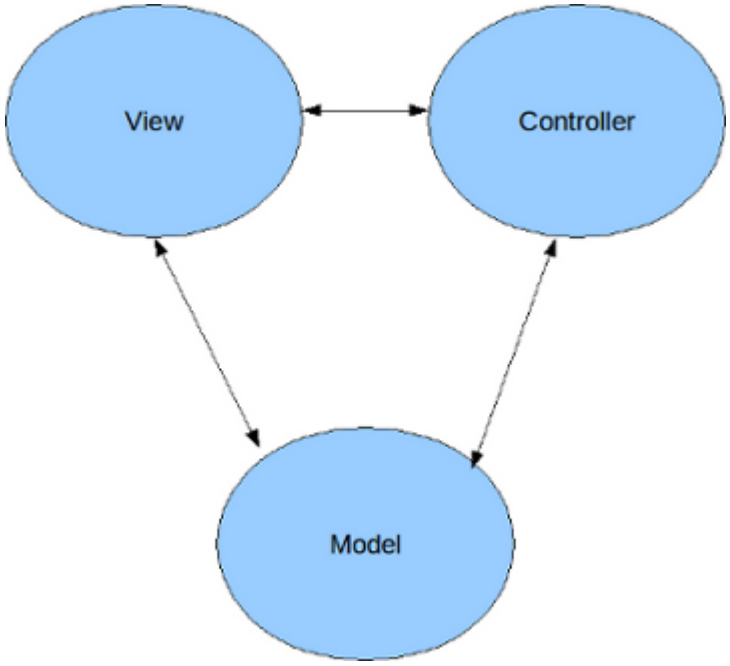

Figure 10: The MVC model was applied Using XML, javascript, html, Cascading style sheets and PHP.

To further facilitate development an agile software developmental life cycle was followed. This allowed for incremental development allowing core features to be developed first improving and adding to this as required features and functionality emerge.

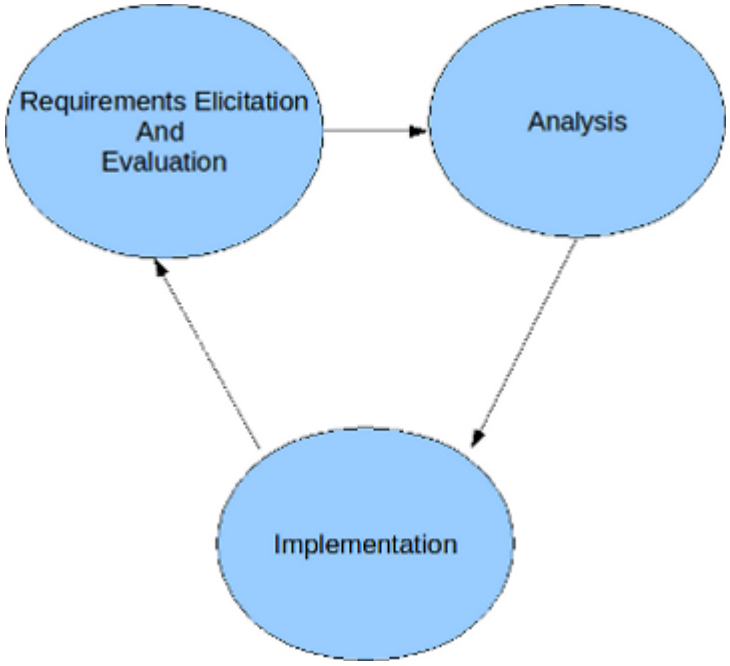

Figure 11: Interative model

Tools Used

The server used to host the website was a Ubuntu Linux desktop running an Apache 2 server with perl, MySQL and php modules enabled. As a development environment firefox with the fire bug plug-in was used along with the Screem html editor. The solution was implemented using Javascript and the JQuery library for client side scripting while php was used as the server side scripting language.To facilitate the implmentation of a user friendly interface cascading style sheets was used. Initial testing was done using a standards compliant browser with plans to ensure coherency across all browsers.

5 Summarative Evaluation

Due to unforeseen circumstances involving the health of the project member responsible for this section of work execution of the evaluations could not be completed. The following sections outline the planned actions that would have been taken to procure sufficient information to evaluate the implemented interface. Evaluations were design using the D.E.C.I.D.E. framework as described by Preec et al [24].

5.1 Usability Testing

5.1.1 Participants

As the targeted group for this project was teachers for this evaluation teachers who had been part of the design process as well as teachers at the schools who were interested in the project were invited to partake in the evaluation of the developed interface.

5.1.2 Setting

Testing for the evaluation was arranged to be done at the teachers schools. A controlled room would be made available to act as a lab for the duration of the testing. This would allow the venue for the testing to be controlled minimising the chance of disruptions which would effect any results obtained.

5.1.3 System and Tools

For the purposes of testing a windows based machine with the firefox web browser installed would be used. To facilitate testing the screen capture suite Camtasia would be used to record the testing sessions. This would allow user interactions with the system to be stored as well as act as a timing mechanism for the testing.

5.1.4 Aims

The goal of the testing was to determine if the interface developed was usable by teachers to create custom clients which offered services through MXit. A secondary goal of the evaluation was to determine difficulties users experienced in using the interface to indicate possible areas of improvement. Thirdly the testing would serve as a measure to gauge user

reaction to the system as a whole allowing teacher the opportunity to indicate what they liked and disliked about the system.

5.1.5 Procedure

To perform the evaluations a constructive interaction method was chosen. This would allow two teachers to work together to complete a set of tasks.At the start of the each session both participants would sign a waiver indicating voluntary participation in the experiment. After which details of the system and procedure of the testing would be explained to them. The user would then be given up to five minutes to familiarise themselves with the interface before being given a set of tasks to complete. The completion of these tasks would be recorded using Camtasia software which would act as a log to record the length of time it would take to complete the tasks. The user would be given fifteen to twenty five minutes to complete the tasks allowing for variable typing speed of the teachers. Once these tasks were completed the teachers would be asked to complete a written questionnaire about their experiences after which they would be asked a set of verbal open ended questions [21, 9]. Should it prove possible a group discussion with the participants would be held to discuss their experiences and opinions of the system.Should it prove possible a group discussion with the participants would be held to discuss their experiences and opinions of the system.

5.1.6 Expected Outcomes

As a result of the user testing quantitative data would be collected indicating the time a user would take to complete a set of task. As time was a constraining factor for teachers the goal would be to minimise this time. Also a system error count would be produce indicating the average number of errors the system produce through user interaction as well as a user error count indicating the number of mistakes a user made during their interaction of the system. System errors would indicate problems in the design or implementation of the interface which would have to be re-evaluated to check correctness while user errors indicate areas which should be improved should the errors prove significant. Further data would be collected through the analysis of a user questionnaire results. Qualitative data would be collected through the open ended questions and opportunistic group discussion which could indicate possible areas of improvments and extensions to the system.

5.2 Expert Evaluation

5.2.1 Participants

For the analytic evaluation members the ICT for development lab would act as expert evaluators. Specifically members with with experience around usability would be asked to act as experts for this evaluation.

5.2.2 Setting

The evaluations would be performed in a lab setting allowing for control over external variables. The system and tools used would be the same as for the user evaluation as previously mentioned.

5.2.3 Aims

The aim of this evaluation was to perform a heuristic evaluation on the interface to determine usability of the system. This would also serve as a guide to indicate possible areas of improvement. This evaluation would also serve to augment the results obtained from the user testing possibly showing an insight into the issues users might have indicated in their testing.

5.2.4 Procedure

Similar to the user testing the evaluators would first be asked to sign a waiver indicating voluntary participation in the experiment after which the nature of the evaluation would be explained to them. Unlike the user testing the expert evaluators would interact with the system individually so as to not influence their individual opinions. The evaluators would be given the same set of tasks to complete as the teachers but would not be timed and would be given more freedom to explore the interface. This would be done to give the evaluators an idea of the tasks users might try to complete while giving them to explore the interface to determine issues which might arise should a user get distracted from a given task. Evaluators would be asked to list and record any issues they encountered as they worked with the system in a notebook. To facilitate the identification of errors a copy of Nielsen's ten heuristics and brief description of the visual information seeking mantra would be provided to the evaluators. Oral feedback would be recorded through note taking

by the experiment administrator after the evaluator has completed the tasks or believes they had identified all probable issues and points of interest the evaluator would be asked for suggestions to make interaction with the interface more appealing and pleasing to the user.

5.2.5 Expected Outcomes

Through the execution of this evaluation qualitative data would be collected which would be used to better understand any significant results from an analysis of the user testing. The data collected from this evaluation could also be used to indicate possible difficulties users may experience and indicate areas of improvement and possible enhancements to the system. This data would be used to facilitate an understanding of why the interface succeeded or failed in delivering a pleasing user experience.

6 Conclusion

6.1 Conclusion

A system was successfully developed that allowed teachers to create custom clients which could offer services to MXit users. Investigations into the designing of an interface for such a system indicate that the creation of a generic tool usable by all subjects may prove difficult. This is due to the small screen size users may have difficulty in reading large amounts of text which may be required for some subjects.

It is possible to design a user friendly interface and through the use of guides and correct affordances achieve the goal of giving teachers the ability to create content for MXit. The difficulty discovered here was allowing the teachers to see themselves as creators of electronic resources and not only users of these resources. This could be facilitated through guiding of actions in order to achieve goals.

Indications from the teachers imply a willingness to create services for learners using technology learners would be familiar with. A major issue the teachers voice was the time to create the actual content indicating that efficiency and ease of use would be important aspects of an interface designed for teachers.

6.2 Implications

The implications of this system is a closing of the digital divide for learners as they would be able to access additional resources using low cost information communication technology. For teachers to this allows for the closing of the digital divide as the system empowers teachers by allowing them to create learning services for mobile devices. The system also encourages teachers to become more confident using technology and hopes to improve learner grades through the provision of learning facilities they otherwise would not have access to.

7 Future Works

Future work can be carried out to further investigate the addition of extra features and services. These include the addition of a feedback service to allow teachers to answer questions or view issues when they are available. This would be akin to posting a comment or sending a private message to the contact. Another feature which could be implemented would be offering a forum based service allowing teachers and learners to create a collaborative learning environment over MXit.

A possible research area is the use of MXit as a course management system similar to vula in schools. Also the effects of the implemented services could be a research area to better understand end user needs and facilitate the development of useful services. A further area of research would be investigating the MXit platform for alternative uses. Dr. math has implemented some simple games for educational purposes indicating the possibility of using MXit as a simple games platform.

References

[1] Ben, S. The eyes have it: A task by data type taxonomy for information visualizations. In *Proc. Of the 1996 IEEE Symposium on Visual Languages, IEEE Computer Society, Washington, DC* (1996), pp. 336–343.

[2] Blaine, T., and Perkis, T. The jam-o-drum interactive music system: a study in interaction design. In *DIS '00: Proceedings of the 3rd conference on Designing interactive systems* (New York, NY, USA, 2000), ACM, pp. 165–173.

[3] Butgereit, L. Math on MXit: the medium is the message. *Plenary papers*, 107.

[4] Butgereit, L. Using Instant Messaging over GPRS to help with school work.

[5] Butgereit, L. Math on MXit: using MXit as a medium for mathematics education. In *Meraka INNOVATE Conference for Educators, CSIR, Pretoria* (2007), pp. 18–20.

[6] Coiera, E. Interaction design theory. *International journal of medical informatics 69*, 2-3 (2003), 205–222. note: Working Conference on Health Information Systems.

[7] Eisenberg, M., and Fischer, G. Programmable design environments: integrating end-user programming with domain-oriented assistance. In *CHI '94: Proceedings of the SIGCHI conference on Human factors in computing systems* (New York, NY, USA, 1994), ACM, pp. 431–437.

[8] Hallnos, L., and Redstrom, J. *Interaction Design: Foundations, Experiments.* Textile Research Centre, Swedish School of Textiles, Unversity College of Boros and Interactive Institute, Boros, Sweden, 2006.

[9] Harper, B., and Norman, K. Improving user satisfaction: The questionnaire for user interaction satisfaction version 5.5. In *Proceedings of the 1st Annual Mid-Atlantic Human Factors Conference* (1993), Citeseer, pp. 224–228.

[10] http://devzone.mxit.com/. MXit DevZone Website.

[11] http://www.w3schools.com/. World Wide Web Consortium tutorial pages.

[12] Hughes, J., King, V., Rodden, T., and Andersen, H. The role of ethnography in interactive systems design. *interactions 2*, 2 (1995), 56–65.

[13] Jones, M., and Marsden, G. *Mobile interaction design.* John Wiley & Sons, 2006.

[14] Kappel, G., Proll, B., Reich, S., and Retschitzegger, W. *Web engineering.* Wiley, 2006.

[15] Kensing, F., and Blomberg, J. Participatory design: Issues and concerns. *Computer Supported Cooperative Work: The Journal of Collaborative Computing 7*, 3/4 (1998), 167–185.

[16] Lidwell, W., Holden, K., and Butler, J. *Universal principles of design.* Rockport Publishers, 2003.

[17] Muller, M. J. Pictive—an exploration in participatory design. In *CHI '91: Proceedings of the SIGCHI conference on Human factors in computing systems* (New York, NY, USA, 1991), ACM, pp. 225–231.

[18] Muller, M. J., and Kuhn, S. Participatory design. *Commun.ACM 36*, 6 (1993), 24–28.

[19] Myers, B. A., Ko, A. J., and Burnett, M. M. Invited research overview: end-user programming. In *CHI '06: CHI '06 extended abstracts on Human factors in computing systems* (New York, NY, USA, 2006), ACM, pp. 75–80.

[20] Nielsen, J. Ten usability heuristics.

[21] Norman, K. QUIS: The Questionnaire for User Interaction Satisfaction. Tech. rep., Technical report, http://www. cs. umd. edu/hcil/quis, 1995.

[22] Parsons, D. *Dynamic Web Application Development Using XML and Java.* Cengage Learning EMEA, 2008.

[23] PETRE, M., and BLACKWELL, A. F. Mental imagery in program design and visual programming. *International Journal of Human-Computer Studies 51*, 1 (1999), 7–30.

[24] Preece, J., Rogers, Y., and Sharp, H. *Interaction design: beyond human-computer interaction.* John Wiley, 2002.

[25] Repenning, A., and Perrone, C. Programming by example: programming by analogous examples. *Commun.ACM 43*, 3 (2000), 90–97.

[26] WHITLEY, K. N. Visual programming languages and the empirical evidence for and against. *Journal of Visual Languages & Computing 8*, 1 (1997), 109–142.

[27] www.controlledchaos.com. SCRUM Website.

[28] www.mxitelifestyle.com. MXit Website.

Appendix

A Unstructured Post Evaluation Interview Guide

How do you feel?

What do you think?

What did you find difficult about the screen?

What did you like about the interface?/was there anything you liked?

What could be done to make the interface better?

What would you like to see done to the interface? what features would you like?

How do you feel about using the interface?

Would you want to use it again? Why or why not?

What would make you change your opinion if you don't want to use it again?

How do you see yourself using the interface and what for?

How would you feel about using or modifying work someone else had created with this system?

How much feedback would you want from the system? and how often would you want to access feedback

B Planned Tasks

Task one (Entering Information):

- Go to the interface and read the information box. Click OK when you feel ready. Give the contact the name EXCITE.
- In the home Screen Enter the following text:
  This contact is for the revision of school work by learners using MXit the grade covered is Matric.

Task three(Creation):

- create a new MXit Screen,name it matric and enter the following text:
  This section covers work completed during matric at schools in South Africa

Task four(Definitions List Creation):

- create a Definition List named Biology.

- Enter the following text in the definition list description:
  This contains definitions for some common words covered in biology during grade 11

Task five(Creating definitions):

- create a new word to define, example word one would be osmosis.
- Define osmosis to be the diffusion from a high concentration to a low concentration

Task Six(Creating Assessments)

- go back to the matrix screen and create a new assessment called quiz
- create an new single answer or terminology question with the following question.
- Question How many langauges does South Africa Have
- Answer Eleven
- Feedback - Eleven Languages

Task Seven (MCQ)

- go to the assessment/Quiz screen and create a multiple choice question
- Question: How many apples in a dozen bread rolls
- Answer A : 12
- Answer B: 6
- Answer C : none
- Choose the correct answer to be C.

C Generalised IR

```
<applicationName>
<type>
        <header>
                <name></name>
                <parent></parent>
        </header>
        <content>
                <DisplayName></DisplayName>
                <question></question>
                <Statements></Statements>
                <Answer></Answer>
                <correct><correct>
                <ScreenContent><ScreenContent>
        </content>
        <options>
                <AlertDate></AlertDate>
                <FeedbackOn></FeedbackOn>
                <RepeatUntilCorrect></RepeatUntilCorrect>
                <Mark></Mark>
        </options>
</type>
</applcationName>
```

Example From Questionaire

Identification number: _____
System code: _____
Age: _____
Gender: ____ **male**

____ **female**

PART 1: System Experience

1.1 How long have you worked on this system?

__ less than 1 hour
__ 1 hour to less than 1 day
__ 1 day to less than 1 week
__ 1 week to less than 1 month
__ 1 month to less than 6 months
__ 6 months to less than 1 year
__ 1 year to less than 2 years
__ 2 years to less than 3 years
__ 3 years or more

1.2 On the average, how much time do you spend per week on this system?

__ less than one hour
__ one to less than 4 hours
__ 4 to less than 10 hours
__ over 10 hours

PART 2: Past Experience

2.1 How many operating systems have you worked with?

__ none
__ 1
__ 2
__ 3-4
__ 5-6
__ more than 6

2.2 Of the following devices, software, and systems, check those that you have personally used and are familiar with:

__ computer terminal
__ personal computer
__ lap top computer
__ color monitor
__ touch screen
__ floppy drive
__ CD-ROM drive
__ keyboard
__ mouse
__ track ball
__ joy stick
__ pen based computing
__ graphics tablet
__ head mounted display
__ modems
__ scanners
__ word processor
__ graphics software
__ spreadsheet software
__ database software
__ computer games
__ voice recognition
__ video editing systems
__ CAD computer aided design
__ rapid prototyping systems
__ e-mail
__ internet

PART 3: Overall User Reactions

Please circle the numbers which most appropriately reflect your impressions about using this computer system. Not Applicable = NA.

3.1 Overall reactions to the system:	terrible 1 2 3 4 5 6 7 8 9 wonderful	NA
3.2	frustrating 1 2 3 4 5 6 7 8 9 satisfying	NA
3.3	dull 1 2 3 4 5 6 7 8 9 stimulating	NA
3.4	difficult 1 2 3 4 5 6 7 8 9 easy	NA
3.5	inadequate power 1 2 3 4 5 6 7 8 9 adequate power	NA
3.6	rigid 1 2 3 4 5 6 7 8 9 flexible	NA