



THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato

Research Commons

<http://researchcommons.waikato.ac.nz/>

Research Commons at the University of Waikato

Copyright Statement:

The digital copy of this thesis is protected by the Copyright Act 1994 (New Zealand).

The thesis may be consulted by you, provided you comply with the provisions of the Act and the following conditions of use:

- Any use you make of these documents or images must be for research or private study purposes only, and you may not make them available to any other person.
- Authors control the copyright of their thesis. You will recognise the author's right to be identified as the author of the thesis, and due acknowledgement will be made to the author where appropriate.
- You will obtain the author's permission before publishing any material from the thesis.

A LOCATION-BASED AUDIO BOOK READER

A THESIS SUBMITTED

of

Master of Science in Computer Science

at

The University of Waikato

by

SANHONG PENG



Department of Computer Science

Hamilton, New Zealand

©September 2012

Abstract

The ability to display text on mobile devices has been progressively improved during the last decade with the growth in use of mobile devices. Recently, the e-book reader has been incorporated into travel tourism. In this project we design, implement and evaluate a location-based mobile e-book reader that provides users with both text and audio information.

This project has two goals. The first goal is to build a rich location-based travel information provider system as a standalone mobile application without using web browsing functions. The second goal is to provide a communication between the mobile application and a digital library of e-book collections.

Experimental results show that the mobile application, during elected events, has provided a special e-book reading experience to participants. The majority of participants liked the system especially with regards to interface design and functionality. Over half the participants felt the location-based audio e-book reader system was usable and interesting while travelling.

Acknowledgements

Foremost I want to thank my supervisor Dr Annika Hinze for her help and generous support throughout this project. She contributed many of the ideas in this project and without her constant guidance and encouragement, I would not have been able to finish this thesis.

Many thanks to my co-supervisor, David Bainbridge and the digital library group led by him, for providing me with much appreciated support throughout this thesis. Thanks to the Department of Computer Science who offered me a graduate assistant position. I would like to thank all the people in the Department of Computer Science, especially the Associated Professor Sally Jo Cunningham, the Research Fellow Ralf Heese, and the members of ISDB method group. I also would like to thank Sonja Gardiner for proofreading the draft of the thesis.

Many thanks to my husband James Liao. Especially while he was extremely busy with his big project, he always found the time to do lots of housework and provided consistent support for my thesis. I would like to give special thanks to my five-year-old daughter and three-year-old little boy; particularly to my daughter who has been in after-school care for three months since I started working on this project. I promised to fulfill her wish; “Take me home before 5:30 pm” as soon as I am able to.

Many thanks to all leaders of WildWing company for providing me with a mobile software developer role; I find the knowledge I've gained on the job to be invaluable in this thesis. Thanks to various co-workers who asked occasionally

how the work was going.

Finally I would like to thank my parents and my sisters in China for their endless love and support. Nobody could believe how important it is for me to hear their voices on the telephone.

Contents

1	Introduction	1
1.1	E-Book Reader	1
1.2	Problems	2
1.3	Illustrative Application Scenario	2
1.4	General Requirements	4
1.5	Thesis Structure	4
2	Related Work	6
2.1	Greenstone3	6
2.1.1	Design Features of Greenstone3	7
2.1.2	Implementation of Greenstone3	8
2.1.3	Conclusion	12
2.2	TIP	13
2.2.1	TIP Scenario	13
2.2.2	Design Features of TIP	15
2.2.3	Implementation of TIP	16
2.2.4	Conclusion	18
2.3	Audio Books desktop application	18
2.3.1	Audio Books desktop application scenario	19
2.3.2	Design Features of Audio Book desktop application	19
2.3.3	Implementation of Audio Books Application	21
2.3.4	Conclusion	23
2.4	The HistoryMap system	24
2.4.1	Scenario of the HistoryMap system	25
2.4.2	Design Features of HistoryMap System	26

2.4.3	Implementation of the HistoryMap system	27
2.4.4	Conclusion	29
2.5	The Tipple Application	29
2.5.1	Scenario of the Tipple Application	29
2.5.2	Design Features of the Tipple Application	30
2.5.3	Implementation of the Tipple Application	31
2.5.4	Conclusion	33
2.6	Summary	33
3	Requirements and Design of TER System	35
3.1	Requirements and Analysis	36
3.2	Design of the TER System Architecture	38
3.3	Design of Greenstone Digital Library	40
3.4	Communication between server and client	41
3.5	User Experience Design	41
3.5.1	Layout Design Issues	42
3.5.1.1	Marker on the Map	42
3.5.1.2	Book Section Order	42
3.5.1.3	User Travel Path Layout	43
3.5.2	Navigation Routes	43
3.5.3	User Interaction	43
3.5.4	Android Permissions	44
3.6	Summary	44
4	Interface of the TER system	45
4.1	Web Based Digital Library Collection Interface	46
4.2	Map Interface	48
4.3	Audio Player Interface	50
4.4	Reader Interface	52
4.5	Camera System	55
4.6	Multiple Books Version	58
4.7	User Preference Setting Interface	60
4.7.1	Show Scale Bar on the Map	60

4.7.2	Centre Map on GPS Location	61
4.7.3	Show Sight on the Map	62
4.7.4	Show Travel Path	62
4.7.5	Audio Player Preference Setting	63
4.7.6	Book Selection	64
4.8	Summary	64
5	Implementation	66
5.1	Components of TER system	66
5.1.1	DL Server	67
5.1.2	Location System	68
5.1.3	Audio System	68
5.1.4	Reader System	70
5.1.5	Camera System	70
5.1.6	Map Activity	71
5.1.7	Custom Overlay System	72
5.2	System Environment	74
5.3	Building DL Collections for TER	75
5.4	OpenStreetMap for TER	77
5.4.1	About OpenStreetMap	78
5.4.2	Generating the Binary Map File	78
5.5	Interface Implementation	79
5.5.1	Using Layout Design Tools	80
5.5.2	Using drawable resource	81
5.5.3	Using XML resources	81
5.5.4	Setting Layout in Running Time	81
5.6	Persistent Data in TER system	82
5.6.1	Shared Preferences	83
5.6.2	XML resource	83
5.6.3	External Storage	84
5.6.4	Internal Storage	84
5.6.5	Server Storage	85
5.7	Algorithm and Data structure	86

5.7.1	TER System Data Structure	86
5.7.2	TER System Algorithms	89
5.8	TER System Micro Optimizations	92
5.8.1	Reducing Number of Objects	93
5.8.2	Optimizing Loop Syntax	94
5.8.3	Handling Runtime Changes	95
5.8.4	Loading XML Data	95
5.8.5	Avoiding Memory Leaks	97
5.9	Multiple Thread in TER system	98
5.9.1	Background Process in TER System	99
5.9.1.1	Using Handlers	99
5.9.1.2	Using AsyncTask Class	100
5.9.2	Concurrent processes in TER system	100
5.10	Application Permission Setting	102
5.11	Summary	104
6	Case Study: Station of the Cross	105
6.1	Introduce 2012 Station of the Cross	105
6.2	Collect Data	106
6.2.1	Tools to collect GPS data	107
6.2.2	Measure the Physical Place	107
6.2.3	Collect Audio Files	109
6.2.4	Collect Text Files	109
6.3	Adaptation of the TER system	110
6.4	Group Interview/Observe	115
6.5	Release/Install software in the public	117
6.5.1	Non Market .apk Installation	117
6.5.1.1	Run through File Manager/Installer	118
6.5.1.2	Run through USB driver	118
6.5.1.3	Intall the USB Driver	119
6.5.2	Experience of Non Market .apk Installation	120
6.6	Testing	122
6.7	Feedback on the software	122

6.8	Summary	125
7	Evaluation	128
7.1	Objective of the Evaluation	129
7.2	Method	131
7.2.1	Questionnaire Method	131
7.2.2	SWOT Method	133
7.3	Participants	134
7.4	Analysis and Result	134
7.4.1	Participants Background	134
7.4.2	Checking its Usability	136
7.4.2.1	Checking the Efficiency	138
7.4.2.2	Checking the Controllability	139
7.4.2.3	Checking the Learnability	139
7.4.2.4	Checking the Helpfulness	141
7.4.2.5	Checking the Satisfaction	141
7.4.3	Strength, Weakness, Opportunities and Threats	142
7.5	Recommendations	142
8	Conclusions	146
8.1	Summary	146
8.2	Main Contributions	147
8.3	Future Work	148
A	Bibliography	151
B	Ethics Consent Approval for Paper Study	155
C	Participant Information Sheet for Paper Study	158
D	Application for Approval Outline of Research Study	161
E	Location based audio book reader questionnaire	166
F	The Poetry event Introduction for the Case Study	171

G User Study Data Collection	174
H The Stations software introduction	175

List of Figures

2.1	A simple stand-alone site, From <i>The design of Greenstone 3: An agent based dynamic digital library</i> by Don, K. Author, Bainbridge, D. Author and Witten, I. Author, Retrieved from website: http://www.greenstone.org/manuals/g3design.pdf . Reprinted with permission.	9
2.2	Greenstone3 Librarian Interface (Gli)	11
2.3	Greenstone 3 Design Panel Interface	12
2.4	Greenstone 3 Format Panel Interface	13
2.5	Browsing the Subject hierarchy in “Travel in New Zealand” . . .	14
2.6	TIP Architecture, From <i>The challenge of creating cooperating mobile service: Experiences and lessons learned</i> by Hinze, A. Author and Buchanan, G. Author, 2009, Retrieved from website: http://dl.acm.org/citation.cfm?id=1151723 . Reprinted with permission	17
2.7	Components of Audio Books system within TIP and Greenstone, From “Making your digital library mobile with location-triggered audio books”, by Hinze, A. and Bainbridge, D, 2011, <i>Proc Workshop on Books</i> , p. 1-4. Reprinted with permission. . .	22
2.8	Audio server architecture, From “Making your digital library mobile with location-triggered audio books”, by Hinze, A. and Bainbridge, D, 2011, <i>Proc Workshop on Books</i> , p. 1-4. Reprinted with permission.	24

2.9	Architecture of HistoryMap.From “Searching and browsing in a digital library of historical maps and newspaper”, by Jones,S.Author,Jones,M.Author,Barr,M and Keengan,T.T.Author and Frances,K.Author,2005, <i>Journal of Digital Information</i> .Reprinted with permission	28
2.10	Components of the Audio Book Service within TIP and Greenstone. Adapter from “Making your digital library mobile with location-triggered audio books”,by Hinze,A. and Bainbridge,D, 2011, <i>Proc Workshop on Books</i> , p. 1-4. Reprinted with permission.	31
2.11	Screen shots for this Tipple system.From “Making your digital library mobile with location-triggered audio books”,by Hinze,A. and Bainbridge,D, 2011, <i>Proc Workshop on Books</i> , p. 1-4. Reprinted with permission.	32
3.1	Build GPS Location and Location-based index	38
4.1	Greenstone Librarian Interface)	46
4.2	Travel in New Zealand Collection Home page	46
4.3	Books in the Collection	47
4.4	Search based on the GPS	48
4.5	Pictures of the Paradise Garden	49
4.6	Paradise Garden Collection Content	50
4.7	Map Interface	51
4.8	Display Location on the Map	51
4.9	Map Location and Travel Route	52
4.10	Audio Player Interface	52
4.11	The Stations Reader Interface	53
4.12	The Garden Reader Interface	53
4.13	The Auckland Poems Reader Interface	54
4.14	The Auckland Poems Reader Interface via Touching Icons	55
4.15	Pictures loaded and viewed	56
4.16	Scrolled view of POP UP Window	57
4.17	Scrolled view of POP UP Window	57

4.18	Camera System in TER system	58
4.19	Multiple Books Loaded From DL	59
4.20	Map Icon Sets for Book A	59
4.21	Map Icon Sets for Book B	59
4.22	Various Books on the Map	60
4.23	Android Preference Screen	61
4.24	Show Scale Bar on the Map	61
4.25	Centre Map on GPS Location	62
4.26	Show Sights on the Map	63
4.27	Show Your Trail on the Map	63
4.28	Audio Player Preference Setting	64
4.29	Centre Map on GPS Location	65
5.1	TER system components	67
5.2	Location System Components	69
5.3	Audio System Components	69
5.4	Reader System Components	70
5.5	Camera System Components	71
5.6	Android Activity Life Circle Diagram	72
5.7	Custom Overlay System UML Diagram	73
5.8	Building the collection	77
5.9	Hamilton Binary Map File	79
5.10	Auckland Binary Map File	80
5.11	Widget views in XML	82
5.12	Location sample stored in HashSet	88
5.13	Meta Data sample stored in HashMap	88
5.14	Audio Files sample stored in LinkedList	90
5.15	Pythagorean Theorem in TER system	91
5.16	Algorithm in Audio Player Reminder subsystem	92
5.17	Loop Optimization example	94
5.18	Web-based XML Data in DL Example	96
5.19	Synchronized example in TER system	101
5.20	Permission Required in TER system	102

6.1	2012 the Station of Cross.From <i>stations of the Cross</i> , by the Stations organization Author,2012,Retrieved from webiste: http://www.stations.org.nz/start.php with permission.	106
6.2	The Stations Collection	107
6.3	Final Version of the Station GPS data	109
6.4	The Stations Audio Files	110
6.5	Audio Download.From <i>stations of the Cross</i> , by the Stations organization Author,2012,Retrieved from webiste: http://www.stations.org.nz/start.php .Reprinted with permission.	111
6.6	The Stations Data in XML format	112
6.7	Comparison between TER system and the Stations	113
6.8	The Overview of the Architecture	114
6.9	The Stations	115
6.10	The Stations Digital Library Collection	117
6.11	The OEM driver download URL, <i>OEM USB driver</i> ,by Android Developers Author,2012, Retrieved from http://developer.android.com/tools/extras/oem-usb.html . Reprinted with permission	120
6.12	Usage Time	124
6.13	Usage share of Android version	126
6.14	Comparison of audio and text presentation	126
6.15	Further development suggestion	127
7.1	Auckland City.From <i>Auckland City Council</i> ,by City Council Author,2012,Retrieved from website: http://www.aucklandcity.govt.nz/whatson/events/livingro with permission.	130
7.2	Poetry on the Pavement	131
7.3	Queen Street in Auckland Stories in DL	132
7.4	Participants Age Group	135
7.5	Use of books, audio books and others while travelling	136
7.6	Comparison of usage of books while travelling	136
7.7	The Poetry DL and XML	137
7.8	Overall Result of Evaluation	138
7.9	Checking the Efficiency	139

7.10	Checking the Controllability	140
7.11	Checking the Learnability	140
7.12	Checking the Helpfulness	141
7.13	Checking the Satisfaction	142
7.14	SWOT Evaluation Report	144
7.15	Recommendations for the TER System	145

List of Tables

2.1	Chrystal and James Interests.	14
2.2	Evaluation results of related work	33
6.1	TER Installation in the Public	125

Chapter 1

Introduction

An E-book reader displays book content typically on mobile devices. This thesis works on a specialised travel e-book reader system that provides location-based audio content to its users. The system stores the electronic books in its digital library. It then displays lists of books on mobile devices to the users while travelling and it provides audio functions for accessing these books.

1.1 E-Book Reader

E-Book reader is the shortened name of “electronic book reader”. An e-book reader is a mobile electronic device that is designed primarily for reading digital books (Wikipedia,2012a). Any device which can display text on the screen can act as an e-book reader.

Several E-Book Reader devices aim to closely replicate the visual look of reading a physical book. Newer e-book readers, such as Kindle Fire, provides some additional functionality, such as web browsing multimedia content (ITBusinessEdge 2012).

1.2 Problems

E-Book readers may be suitable in the tourism context due to their portability. There has been some research into the Travel E-book reader. Specialised travel E-book readers present book content specifically related to the travel location and is triggered by the travellers' location. Selected travel e-book readers are described in Chapter 2. None of these existing location-based e-book readers customise their book collections. The books were mostly sourced from existing websites or a local data. The size of local data is limited to the storage capacity of the mobile device. Using an existing website as a resource is not a guarantee as websites may be updates or removed. My project takes a different approach: storing the books in a digital library (as explained in the following scenario).

1.3 Illustrative Application Scenario

In this thesis I describe the design of a location-based tourism e-book reader for audio content (TER). The following section describes an example scenario which is used throughout this paper to illustrate application requirements. The scenario is divided into steps for easier reference later in the paper.

Consider the following scenario for two tourists James and Chrystal who visit Hamilton in New Zealand. They are currently in the Hamilton Gardens. Both have an Android TER application on their android phone. In this scenario both visitors will spend time together.

Step 1: Before they start their travels, Chrystal and James have been equipped with an Android TER application and Greenstone digital library. James is interested in the plants. Chrystal is interested in the plants and the architecture.

Step 2: When James and Chrystal reach the Hamilton Gardens, there is a "chirp chirp" sound from their mobile device. As they look at their phone, the TER system presents to them the name and location details of Hamilton Paradise Garden, which links to e-books. All book chapters are numbered in

order on the map.

Step 3: Both click on the button “View Books”, then TER retrieves from the Greenstone digital library a list of books based on their location and interests. A list of books appears on the screen. James chooses the “Hamilton Garden Design” chapter from one of the books, while Chrystal chooses “English Garden in Paradise Garden Collection” from another book. They both then click on the button “play audio” and start listening to the stories, which are in MP3 format, while they are walk around in the English Garden.

Step 4: When they are closed to the “Italian Renaissance Garden” collection the “chirp chirp” sounds again. This time they see a new list of books or chapters related to the Italian Garden. This time both of them choose one chapter, then click on “Play audio” and the “Show Text” button. They can now listen to the story and view the text simultaneously. Chrystal clicks the volume panel to adjust the volume.

Step 5: When they touch the map marker of the “Maori Garden”, which was not fully open yet, they found they could read a story about it even though they were far away from the Maori Garden, and they were able to view a picture of it.

Step 6: After they finished visiting the Paradise Collection, they clicked on the “Show User Path”, and selected all books on the TER. The TER suggest to both tourists to visit another plant collection in the Hamilton Gardens.

Step 7: James and Chrystal would like to review what the gardens looked like. They touched on the map marker, and the stories and the pictures which were taken or loaded from the Server were displayed accross the map.

1.4 General Requirements

Taking the sample scenario from Section 1.3, a very basic functional requirements analysis was undertaken to establish the features and characteristics of the software to be developed. More details will be discussed in Chapter 3.

R1: Connect to digital library The digital books and information includes audio books and images which will be loaded from the digital library (DL). The Greenstone Digital Library has to support searching by location. This will allow the application to load a list of books and images from the server to achieve Step 3, Step 5 and Step 6 in Section 1.3.

R2: Run on a mobile device This application explores the role of audio books for mobile users. It is expected to provide tourist information to mobile users. This will allow user to travel around with TER system and achieve all steps described in Section 1.3.

R3: Support location-based system This mobile application offers location-based information based on the scenario explained in Section 1.3 Rather than sitting in an office or connecting to internet users can travel around and connect to the digital library.

R4: Provide real text-to-speech function In order to listen to the audio files described in Step 3 in Section 1.3, this mobile application provides a real text-to-speech function, which can convert text to speech.

R5: Play audio books This service will let users select related audio books or chapters based on their current location. It gives users control over the audio playback explained in Step 3 and Step 4.

1.5 Thesis Structure

We start from research on existing related work, including digital library collections, existing travel e-book readers, and location based systems. These are described in Chapter 2. Then we address our solution to designing the

new travel e-book reader application (TER system). Chapter 3 focuses on the requirements and design of the new system. Following that, the design implementation experience is described in Chapter 4. Chapter 5 shows screen shots of the TER system. The results of the TER system are explained in Chapter 6 and Chapter 7. Chapter 6 describes a study with the event “The Stations of the Cross” in April 2012. Chapter 7 is based on the event “Pavement of the Poetry” in June 2012.

Chapter 2

Related Work

In this chapter we introduce existing audio book service applications. As these applications are related to Greenstone 3 and TIP, both systems are introduced first in Section 2.1 and 2.2. In Section 2.3 we look at one audio application that runs on both desktop and GPS simulation components, after which we describe the HistoryMap system which is built on the Greenstone Digital Library in Section 2.4. In Section 2.5 we introduce the Tipple application, which runs on Android Phones. Finally, we conduct an evaluation of these applications depending on our basic functional requirements.

2.1 Greenstone3

This section describes Greenstone3 Digital Library software; an open source system for the construction and presentation of information collections (Greenstone 2011). A collection of information comprises many documents; up to several thousands. A digital library may include many different collections and each collection has a different presentation. These collections can be maintainable, searchable, and browsable.

Greenstone3 is the latest version of Greenstone and it is aimed at improving the dynamic nature of the Greenstone toolkit; in terms of how it organises content and how it provides services. It is a fully controlled management tool for users to build a collection. For example; a user can customise its metadata

and its format. In this project a digital library collection built on Greenstone3 has been added with its necessary GPS metadata and special format on demand. Also, the Greenstone Team works in the Digital Library of the University of Waikato. They have already run the Greenstone Digital Library on the Android Tablet and now the research group is working on running it on Android mobile devices. This will support the application running smoother on Android devices in terms of speed and cost.

2.1.1 Design Features of Greenstone3

Greenstone digital software has been widely employed by many users to build digital library systems. These digital library systems can be run on any Windows computer (or Linux) and provide a web browser interface to a local intranet or internet. From users different aims of the various projects and the developers different conceptions of what a digital library is, the following design features of Greenstone3 were drawn out in Don et al (2002).

F1: Backwards compatibility Greenstone3 needs to retain the existing system's strength by ensuring it is backwards compatible.

F2: Levels of customization Greenstone3 will provide different levels of customisation. This is targeted at the various purposes of different groups of people, such as content developers, collection editors, and workflow designers. Each group of people may have different requirements in building their digital library. For example, a workflow designer may want the system to function in a novel way, while the collection editor wants it to function in a standard way.

F3: Software modularity Code modularisation is essential to facilitate development and long term management of the software.

F4: Service based A set of services is provided through modularity of function, such as ResourceRetrieve Service, TextQuery Service, MetadataRetrieve Service, and ClassifierBrowse Service.

F5: Distributed architecture A rich digital library can be accessed in an intranet or the internet via the addition of an open protocol.

F6: Future compatibility Due to digital libraries long term function, the old collections need to be presented by updated versions of the system.

F7: Dynamic Users can add new documents and remove the documents while a repository remains online. The system also allows the user to adjust the presentation issues and add services at runtime.

F8: Computer environment integration Full integration makes the digital library become a seamless component of individual users work environment.

2.1.2 Implementation of Greenstone3

We now describe the Greenstone3 architecture. For portability reasons, Greenstone3 is written in Java and XML-encoded messages were used to communicate between agents.

Figure 2.1 shows the architecture of Greenstone3. A typical basic Greenstone3 digital library is made of up a back end (site), coupled to a front end called a receptionist. A receptionist provides the user interface, which allows users to send queries to the system. The MessageRouter (MR) agent acts as a main bridge between receptionist and any collection or service.

The digital library back end in Figure 2.1 contains two collections: “demo” and “foo”. A collection is a group of documents with a uniform means of access. Inside the system is a cluster of services related to the specific set of data they work on. All functions in the system are called services. In Figure 1, demo collection contains three agents: Search, Retrieve and Browse, which provide four services to the user. “Search agent” provides a query service, “Retrieve agent” provides a document retrieve service and a metadata retrieval service, “Browse” provides a metadata browsing service.

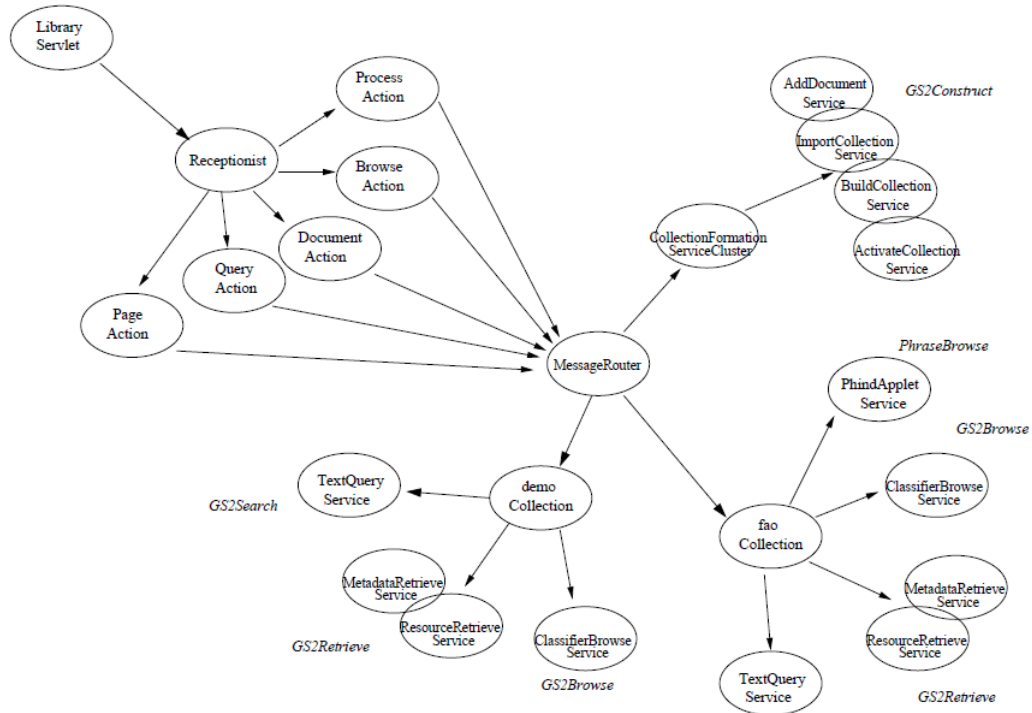


Figure 2.1: A simple stand-alone site, From *The design of Greenstone 3: An agent based dynamic digital library* by Don, K. Author, Bainbridge, D. Author and Witten, I. Author, Retrieved from website: <http://www.greenstone.org/manuals/g3design.pdf>. Reprinted with permission.

In Figure 2.1 we see that foo has a service cluster of four collection-formation services. A service cluster is a group of services which are related to each other. It provides four services: “AddDocuments” which allows the system to add a document to a collection, “ImportCollection” imports all documents related to the system, “BuildCollection” is a service that builds all indexes and browsing structures that are associated to the collection, and “ActivateCollection” makes a new collection active and visible to users in runtime. All these services are concerned with creating a collection and grouping it together into a cluster of collection-formation services.

The figuration in Figure 2.1 is a very simple example of a digital library structure. In fact, there may be many sites with many collections and it also may be run on a different computers. Different sites may need to communicate with each other using predefined protocols or XML. The following description

elaborates on this structure.

Modular structure

To meet the modularisation Design Features (F3 and F4), Greenstone3 utilises independent modules called agents, which communicate via a single method in which both input (XMLin) and output (XMLout) are expressed in XML. Therefore the XML specifications can be modified locally and communication will proceed effectively according to the new scheme, provided only that all affected modules are altered appropriately.

If agents are on different computers, they can communicate through SOAP (Simple Object Access Protocol) or other possible protocols as well. For example; if the agent is an Android powered device, they can talk through HTTP protocol.

Interacting with the user

The users send a request to the MessageRouter module by clicking on a URL or submitting a Web Form. This request is intercepted by a servlet that invoked the Receptionist module. The Receptionist creates a XML message based on the input and passes the interpreted message into the system through MR. The Receptionist includes a Java class for each action. These classes decode the message in the URL to determine what information must be requested and send it through MR. A single action normally involves several different requests. For example, The “PageAction” may involve the following queries: 1) a list of its collections, 2) for each collection it may query its metadata such as title and icon.

Greenstone3 Librarian Interface Greenstone Digital Library provides the Greenstone Librarian interface (See Figure 2.2) and allows users to drag and drop the files into the collection and build it immediately. Greenstone 3 can

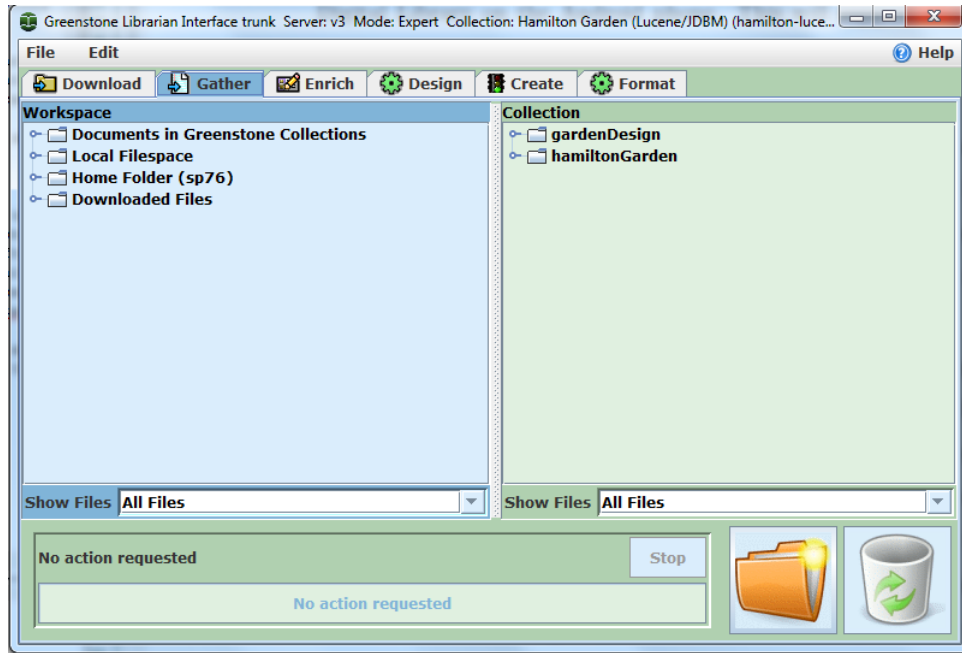


Figure 2.2: Greenstone3 Librarian Interface (Gli)

enrich the files with additional information called metadata. The metadata can be created, edited, assigned and retrieved.

Greenstone 3 Design Panel (See Figure 2.3) allows users to add Document Plug-ins to process different files, such as “imagePlugin” used for image collection and “MP3Plugin” used for audio collection. Adding a plug-in is extremely easy for the user, by choosing one from the drop down list. The Format Panel (See Figure 2.4) can help users alter the display of information, such as metadata in the web pages, on the fly as they are needed. Users can specify the metadata to be retrieved by applying a format command. Users can specify a default format for a particular component, or add a new format command. By clicking on the “BuildCollection” the collection is built and can be viewed immediately.

The research has built three types of collections: image, audio and text. Additional specific metadata which may be useful to the collection for the research and navigation purposes, such as longitude, latitude and title, have been added to the documents for further process. These metadata are then retrieved and manipulated in the application.

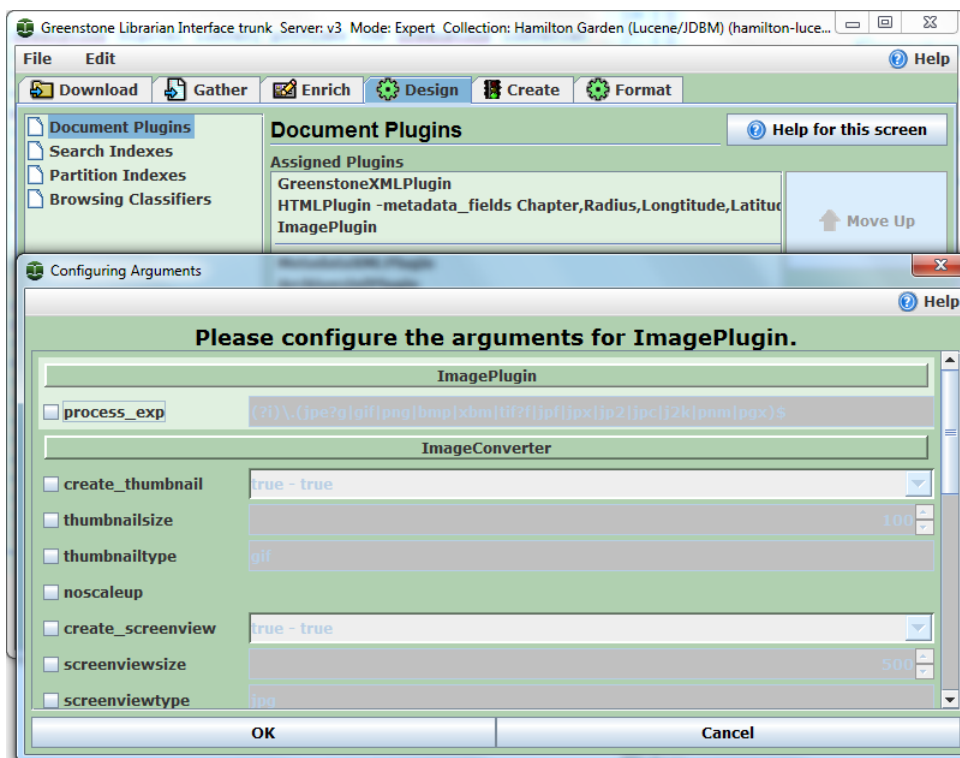


Figure 2.3: Greenstone 3 Design Panel Interface

For example; in collection “Travel in New Zealand”, we input keywords “Hamilton Garden” as the query and clicked on the SEARCH button to submit the query. It then displayed a list of names of documents which match this query (See Figure 2.5). This triggered the TextQuery service through Search agent. If we search by GPS location then all metadata can be retrieved and manipulated via HTTP.

2.1.3 Conclusion

Greenstone3 was used to build a digital library collection to store all books and images for the TER system based on discussion above. The collection customised its own metadata for searching by the location identified with longitude, latitude.

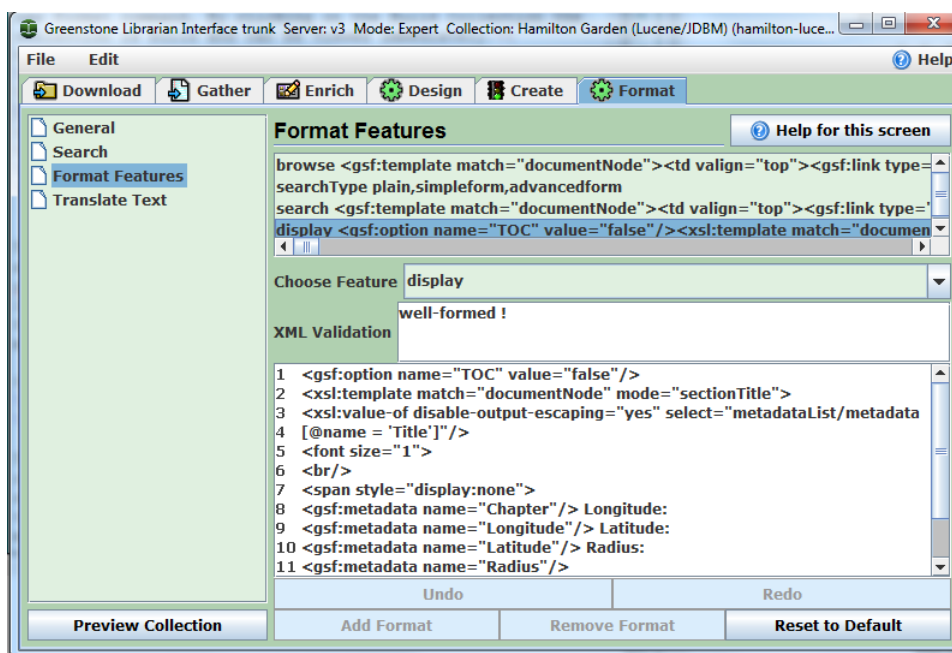


Figure 2.4: Greenstone 3 Format Panel Interface

2.2 TIP

This section introduces the concept and design for TIP, a personalised Tourist Information Provider, described in Hinze et al., 2006. The TIP system and TER system are both location-based tourist information provider systems. This section looks at this previous existing system.

TIP follows the approach that advanced tourist information systems should deliver semantically-rich information about sights rather than static travel information. The TIP system provides details of groups of sights based on the common features of these sights. Furthermore, a TIP system provides travel information based on users' personal information such as users' travel history and recommendations.

2.2.1 TIP Scenario

Let us introduce a scenario in which the application requirements can be illustrated. Two tourists, Chrystal and James are currently sight-seeing together at the Hamilton Gardens. Their background and interests are shown in the

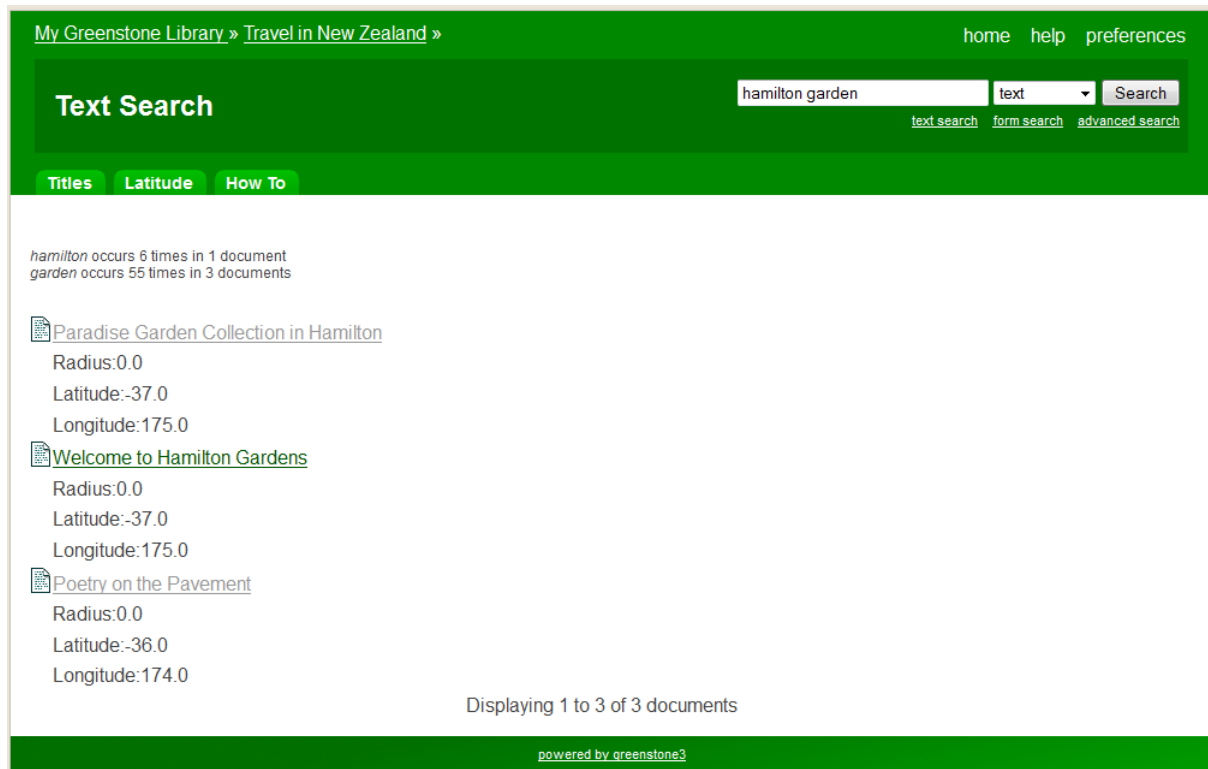


Figure 2.5: Browsing the Subject hierarchy in “Travel in New Zealand”

Table 2.1: Chrystal and James Interests.

Persons	Types of Sights	Information Topics	Travelling History
Chrystal	Paradise Garden	Architecture	First visit to Hamilton
James	Paradise Garden	History	Repeated visit to Hamilton

Table 2.1. They are interested in the same of type of sights but with different information topics. It is Chrystal’s first time to Hamilton, but for James it’s a repeat visit to Hamilton Gardens. Both of them are equipped with the client programmer of TIP —our Tourist Information Provider (Hinze et al, 2006) —and they have already set up user profiles, such as sights and information topics they are interested in through the TIP programmer.

When both of them reach the Hamilton Gardens, they push the INFORMATION button and then receive information about sights which are near to their current location and based on their interests; Chinese Scholars Garden

and English Flower Garden. When they are about to enter the English Flower Garden they push the INFORMATION button again. They are now provided with different information; Chrystal received detailed information about the garden as well as specific information regarding the architecture of the garden. James was presented with brief general information about the garden as well as elaborate information regarding historical facts. The provided tourist information is based on their interest and their travel history.

When they go to visit the Chinese Scholars Garden, they are provided with general or specific (more expert knowledge) information and facts about the topics of architecture and history, respectively. Additionally, Chrystal was given a comparison between the architectures of the Chinese Scholars Garden and the English Flower Garden. Finally the TIP client suggests they visit additional gardens in Hamilton Gardens; the Japanese Garden of Contemplation, American Modernist Garden, Italian Renaissance gardens and Indian Char Bagh gardens.

2.2.2 Design Features of TIP

Hinze et al(2006) identified six design features, which we can link to the above scenario.

F1: User profiles and Information filters The system delivers tourist information based on user interest. The user's profile; such as the user's visit history and user's interest has to be defined and saved.

F2: Location-based information delivery The TIP needs to identify the user's location and range of location, therefore the user's location has to be stored.

F3: Information delivery based on hierarchical information structure Depending on the users personality information, the system provides different detail levels of information. In the above scenario James is presented with specific information (more expert knowledge) and Chrystal was presented with general information about the gardens, due their visit

history.

F4: Clusters of sights The system groups the sights in semantic cluster.

The sights can be grouped based on a common attribute, such as a category of place. In our scenario, all the gardens located at the square “Hamilton Gardens” forms a group of gardens. So Chrystal can be provided with joint architecture information about the Chinese Garden and English Flowers Garden.

F5: Information delivery based on User History The system doesn't provide repeated information to revisiting users. Instead more specific information is provided. Users should receive a different level of information if they revisit the same place. In our scenario James is given specific information because he revisits Hamilton and his visit history has already been stored in the database.

F6: Personalized recommendations based on context The system should

suggest visiting further sights which match the user's context and interests, for example; user's location, user's interest and user's visit history.

In our scenario the system recommended that Chrystal and James visit a Cathedral in Berlin because they visited a number of gardens. The recommended sights are nearby, and are related to their interest.

2.2.3 Implementation of TIP

This section discusses the implementation of TIP 2.0 system (Hinze et al., 2006). Figure 2.6 shows the architecture of TIP 2.0. The TIP server requires clients for the delivery of sight information and client interfaces for registration and profile definition. It provides standard browsers for desktop and mobile TIP browsers for a mobile interface. Services cooperating with TIP may require thin clients or thick clients. Thin clients may use the mobile TIP browser or a standard browser to present the information. Thick clients are application specific; they operate in their own runtime environment. Communication between the server and clients are managed via TIP/IP. Most interactions are

performed via the TIP server since only the server controls the interface definition.

The basic communication between the server and thin clients or between thin

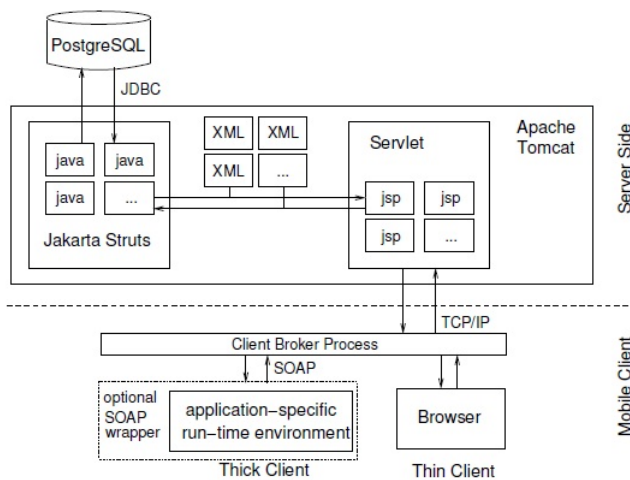


Figure 2.6: TIP Architecture, From *The challenge of creating cooperating mobile service: Experiences and lessons learned* by Hinze, A. Author and Buchanan, G. Author, 2009, Retrieved from website: <http://dl.acm.org/citation.cfm?id=1151723>. Reprinted with permission

clients is performed via TCP/IP using HTTP. Among this connection a TIP client broker is employed using the following set up; each TIP service acts as an independent process, and the client broker process is utilised for the management of processes.

The basic communication with the thick client is more complicated by the limitation of the protocols on the PocketPC platform. Furthermore, the thick client must support service modularity and cooperation both within the framework (TIP) and to services outside the framework, such as the Zoomable Map Services (Jones et al., 2005).

In TIP 2.0 the Zoomable map service is extended by adding another service; the Sight Display Module (SDM). The SDM support both the geographical positioning of halos and the indication of different semantics (e.g., “home” and “other place”). Semantic information may be either displayed explicitly

or communicated to the user through subtle cues such as the use of different shapes or colors. This service can be performed between two processes without server interaction. Thus, handheld devices have sufficient storage to cache simple sight information intelligently within a local context.

2.2.4 Conclusion

The TIP application runs on both the server and client sides and the Desktop runs as a mobile interface. There are a few bridges involved in the system, for example; The server and client communicates using a TIP client, or, the zoomable map service is extended by adding SDM. All these can be achieved in a simple way, with modern improved mobile features. A TER system can be run on a server side mobile application if the Greenstone3 is successfully run on a mobile device, even though they run separately, the communication between server and client is much simpler than the TIP application. In comparison to the TIP application, the TER system will be a standalone mobile application to communicate with the digital library without using its browser.

2.3 Audio Books desktop application

The Audio Books desktop application, which was developed by Xin Gao at the University of Waikato (UoW) in 2007 (Gao,2007), explores the use of audio sources in a mobile tourist information provider system. It uses audio and text-to-speech functions to make digital library books available in a tourist location context. This system and the TER system have some similar features, such as providing a text-to-speech function, using digital library books and running a location triggered system.

The Audio Books desktop application, which was developed by Xin Gao at the University of Waikato (UoW) in 2007, explores the use of audio sources in a mobile tourist information provider system. It uses audio and text-to-speech functions to make digital library books available in a tourist location context. This system and the TER system have some similar features, such as providing

a text-to-speech function, using digital library books and running a location triggered system.

2.3.1 Audio Books desktop application scenario

The following describes the audio book desktop application scenario in Gao (2007). This scenario takes place in the University of Waikato (UoW), and is running on an emulator on the desktop.

A visitor, James, comes to the Library of the UoW. The Audio Icon reminds him that he has already arrived at the Library and that the stories related to the Library are available. When James presses LISTEN TO THE AUDIO button, the screen displays a list of books available based on users location and profiles. When a user selects a particular book the general information of this book such as author and publisher will be displayed. Once the user selects a chapter, the Audio Control Panel will pop up. Through this Audio Control Panel the user has capability to control the audio sound. When James pressed the PLAY TEXT TO SPEECH button, he could then read the text on the screen.

2.3.2 Design Features of Audio Book desktop application

Gao (2007) identified 13 main design features to this Audio Books desktop application. These 13 requirements can be divided into three parts: Audio Icon Functional Requirements, Audio Books Requirements and General Requirements based on their contents.

1. Audio Icon Design Features

F1: Providing basic Audio Icon Service When users enter a range of a specified area the Audio Icon will sound and remind users. Therefore the distance between users and the object needs to be identified.

F2: Distinguishing the source The type of sound related to the place which users are interested in. Different objects should have specific sounds and identify the space by listening to the sound.

F3: Controlling Volume Provide the user with interactions to control the volume, such as turn on/off audio and the ability to adjust the volume according to the particular environment.

2. Audio Books System Design Features

F4: Displaying Audio Books List Users can view a list of possible audio books available before they start listening to a particular audio book.

F5: Playing Chapter-based Books All chapters are ordered in a book. Users are able to know the order of the chapter related to the current location; as well as recognise other chapters and related location.

F6: Text-to-speech When books are loaded from the Digital Library based on the users profile and location, the system will provide text-to-speech software which can convert the text to voice.

F7: Text-to-speech and Text Display In addition to listening to the books, users can read the text through this function. Users can make the decision between listening to the spoken book, reading the text or do both at the same time.

F8: Audio for this Book Audio books have two different models: MP3 file and read by text-to-speech software. MP3 files are stored independently from text; if users would like to read the text of MP3 file or listen to text-to-speech reading, the hyperlink for this audio is provided via linked to the Digital Library. Users can then switch this MP3 file model to text-to-speech file model.

3. General Design Features

F9: Anti Overlap When another object comes into a user's sight or users want to switch from current audio reading to another, users can choose to interrupt the current book reading.

F10: Audio Control This is similar to the common audio player. Users have general audio control abilities such as play forward, play back, and volume adjust.

F11: User Parameters A user can set up some basic parameters, such as audio volume or accepting interruption. For example, accept interrupt give the user the ability to choose whether the notice for new information is displayed or not. These user parameters will be stored in the database.

F12: Audio Books Control Availability Audio books are not automatically played. There will be a dialog box to ask users for their permission to play an audio book.

F13: Signaled place based on users' interest Users only get audio books available based on users interest stored in their profile. Therefore, users are not inundated with large volumes of information from the system.

2.3.3 Implementation of Audio Books Application

Figure 2.7 shows the architecture of the components of the Audio Book Service within TIP and Greenstone. Components previously used are shown in gray. This system runs on a GPS simulation component. When the GPS simulation detects a new object, the recommendation component identifies the location's range.

A list of recommended new locations including the description, place name, and GPS coordinates provided by the recommendation component are sent to the Notifier. The Notifier forwards these location details to the Recommendations Information Manager, Audio Icon Manager, and Audio Books Manager. The Audio Icon Manager will direct the Audio Server to send an audio stream to the Audio receiver on the client side based on the Audio Setting. The Au-

dio Client on the client side will check regularly with the Recommendation Information Manager for new location information. It also checks with the Audio Display Manager to identify the audio that is currently being played. The Audio Display Manager will receive from the Audio Server, the information about the audio currently playing. The Audio Books Manager provides a list of available books (from Greenstone via TIP/Greenstone) to the client side. Once it receives the client selected chapter or is ordered by the Travel Plan Component, it then sends these chapters to the Audio Server. The Audio Books Control Panel is able to provide an interface to users for audio controlling.

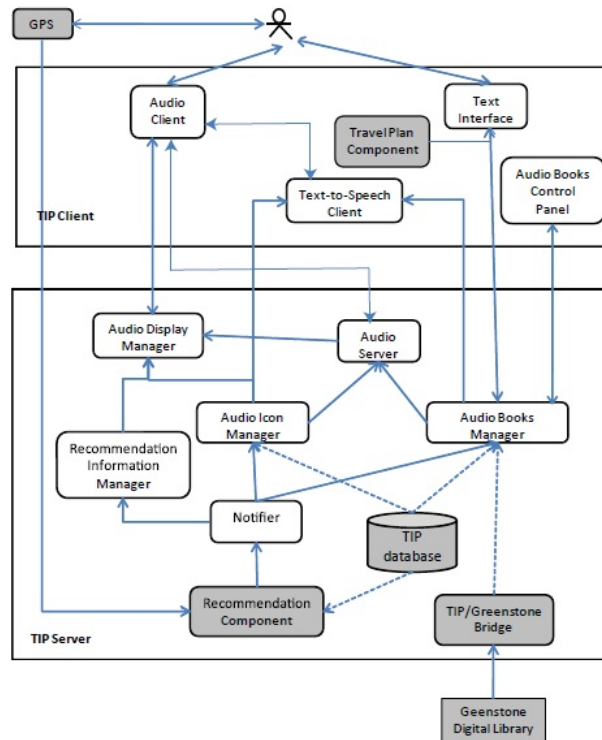


Figure 2.7: Components of Audio Books system within TIP and Greenstone, From “Making your digital library mobile with location-triggered audio books”, by Hinze, A. and Bainbridge, D., 2011, *Proc Workshop on Books*, p. 1-4. Reprinted with permission.

Both the Audio Book Manager and the Audio Icon Manager may send text to the text-to-speech component to be read out. At this stage only the name is

read out through text-to-speech component.

Based on the design of the Audio icon and Audio Books Service, it requires much functionality to complete the task in order to play audios in many different ways. The Audio Icon service needs to play more than one audio file at the same time and the playing audio needs to be changed once the location changes. This requires the system to be capable of mixing streams of audio into one before sending it to the client rather than opening more audio streams on the client side. The Audio Server does this job.

Figure 2.8 shows the architectures of the Audio Server. When users turn on the Audio Receiver, it sends a request with the user's ID to the Audio Server through Audio Sender. This request asks for the current status of the playlist of the Audio Server component. These request details are then passed to the Mix Controller. The Mix Controller is the central controller of Audio Server; the Audio File Loader, the Audio Sender, Remote Control Receiver and Interface with the Audio Icon and Audio Books Service cooperate with the Mix Controller and Mixer to complete the tasks described in the above paragraph. The Mixer loads the file from the Audio File Loader onto a specific length of data, then mixes these data streams and carries the mixed data to the Audio Sender in one single data package. The Audio sender will then send this data package to the Audio Receiver as response to the previous requests.

2.3.4 Conclusion

The Audio book desktop application runs on the desktop, even though it can run on a mobile emulator; however it does rely on the browser. The TER system with built-in audio book on the real mobile device will bring significant benefits while travelling. Additionally, the Audio book desktop application didnt connect to digital library and was therefore not efficient enough while travelling.

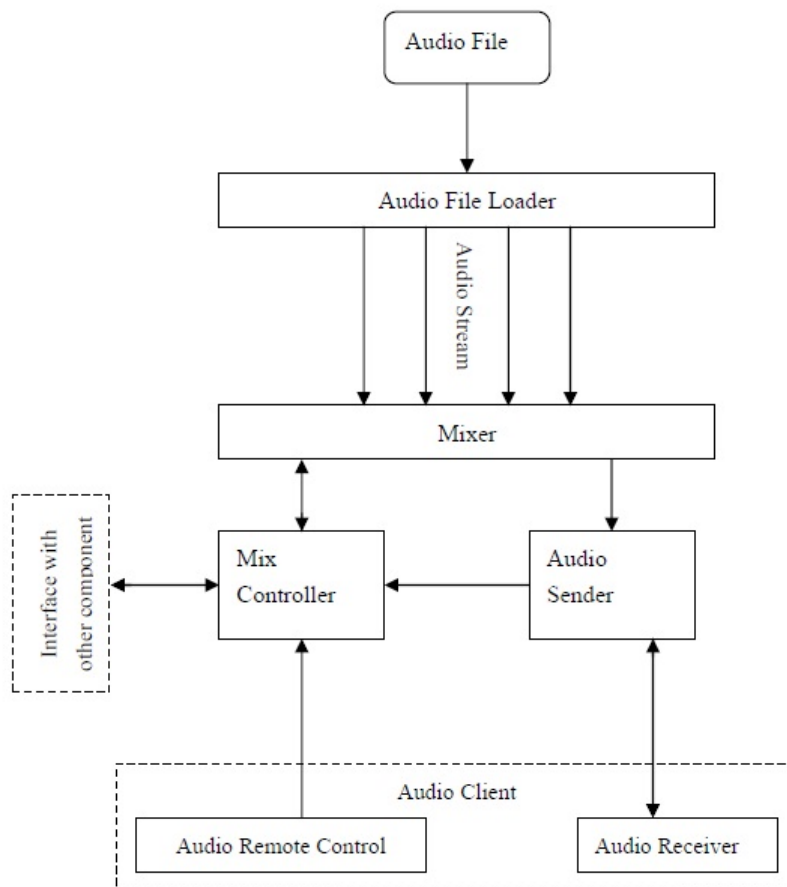


Figure 2.8: Audio server architecture, From “Making your digital library mobile with location-triggered audio books”, by Hinze, A. and Bainbridge, D., 2011, *Proc Workshop on Books*, p. 1-4. Reprinted with permission.

2.4 The HistoryMap system

This section introduces the HistoryMap system that supports searching and browsing through information between both collections of historical maps and Niupepa newspapers collection (Jones et al., 2005). Jones et al (2005) describes how a place name can be searched across both two digitised collections, and also how the newspapers text can be dynamically configured to include hyperlinks to maps containing given locations. The HistoryMap system worked on the location search rather than text search; this will provide good experiences of how to address the location search features and location triggered function built in TER system.

In this section we first describe the scenario of the HistoryMap system. We then describe the HistoryMap System requirements derived from Jones (2005) and then present the underlying implementation.

2.4.1 Scenario of the HistoryMap system

The HistoryMap user interface consists of two components: the first interface (Accessing Map Interface) which facilitates searching and browsing of a set of digitised maps, and the second (Accessing Historical newspaper Interface) which supports searching and browsing access to the Niupepa collection via the standard NZDL-provided interface.

Assume a user, James, is interested in the history of a location. James can search and browse newspapers about his interested location in the Niupepa collection. Firstly, James will access the Nuipepe collection and elect to search directly for places by entering query keywords “Kerepehi 1961” into the query box and select the SEARCH ALL operation. The list of related newspaper titles is displayed on the screen. He can then read the commentary by clicking on the link of the newspaper.

Next, James wants to view the image of the interested location in a given year. This time he accesses the Accessing Map interface and searches for the image by entering the explicit place name. The display is then updated to show a more of detailed map of the selected geographic area. Each thumbnail represents a map covering the geographic area within which the search location is found.

Then James wants to search across these two collections when he selects the Interact With Map operation on the Accessing Historical newspaper interface. The place name he entered in the query box is issued to the Map collection over the network. Hence the newspaper component display is updated with small map-link icons, giving the side-by-side access to related maps and the

newspaper. Further integration is achieved by manipulating the content of the newspaper documents. Maps related to the search location are inserted into newspaper documents dynamically by clicking a small map-link icon.

2.4.2 Design Features of HistoryMap System

The HistoryMap system contains two interfaces: Accessing Map and Accessing historical newspaper. Although these two components can integrate to provide cross-searching between the map collection and Niupepa collection, users can solely interact with one of them. The following nine requirements are drawn out to match the scenario presented in session 2.4.1.

1. Accessing Maps

F1: Supports users in either searching or browsing activities Users can use direct search by putting place name into a query box, or click on a high-level overview map to indicate a particular location of interest.

F2: A list of map thumbnails is displayed in a chronological year order

When users click on the particular location, a list of map thumbnails with the year maps were created is displayed in chronological order. These thumbnails represent maps which cover the geographical area within which the location can be found.

F3: Maps can be navigated Users are presented with navigational arrow that allow them to browse adjacent regions to the north, south, east or west.

F4: Maps have been rendered at different levels of detail Users can see high-level overview map, or zoom in/out the map to access different resolution images.

F5: Searched by location which defined by latitude and longitude As a result users can search a contemporary place on historical maps on which they are not explicitly marked.

2. Accessing Historical newspapers

F6: Emulates the standard HTML web interface to the Niupepa collection

Via the Niupepa interface, users can browse the newspaper collection by providing the place names.

3. Integrating maps and newspapers

F7: Provides the cross-collection searching in both collections

Users can either enter a place name or click on the map. The place names are issued to other collections (Niupepa collection or newspaper collection), and the system responds to the updated newspaper or map.

F8: Reconfigure the newspaper content

Users can update the newspaper by inserting the corresponding map from the map collection.

F9: Users control over integration

The HistoryMap system provides the user with control and flexibility with respect to such annotations. For example, the user can choose to use cross-collection access between two collections, or use either of two interfaces independently. Also, users can choose to turn the link insertion on or off.

2.4.3 Implementation of the HistoryMap system

Figure 2.9 represents the architecture of this system. Both the New Zealand place name database and map database supports searching and browsing activities in the system. The format for this data is ASCII text files and they are stored on disk as Java objects representing the run-time data structures that support query access.

Each location in the database has the following attributes: Location ID, name of location, easting, northing, type of location district, latitude and longitude. The Place name database is stored as a hash table on disk, the location acts as a key. The data associated with the location is stored as an array of strings. Each map has the following characters: image file name, latitude and longitude of the top left corner, latitude and longitude of the bottom right corner, map

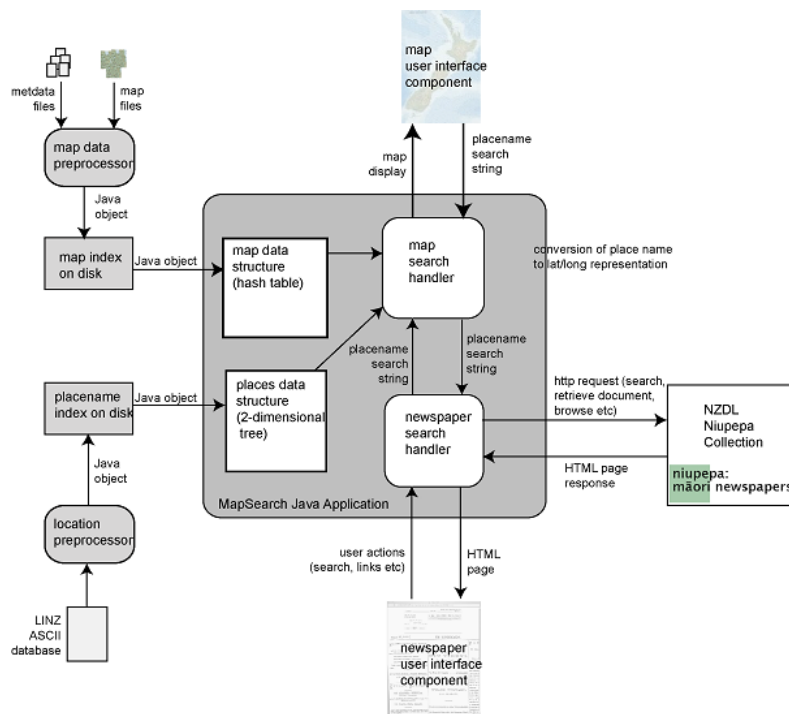


Figure 2.9: Architecture of HistoryMap. From “Searching and browsing in a digital library of historical maps and newspaper”, by Jones, S. Author, Jones, M. Author, Barr, M. Author and Keengan, T. T. Author and Frances, K. Author, 2005, *Journal of Digital Information*. Reprinted with permission

orientation, detail level and year of creation. The Map database is stored as a two-dimensional binary tree on disk. Left or right node is determined by comparison between the current tree node’s discriminator and the corresponding value of the node of interest. For range searching, or finding locations within a geographical area; apply the $R + \log N$ comparisons, where R is the number of points to find and N is the number of points in the tree.

Links to a related map are inserted into newspaper text dynamically. Once the newspaper text has been retrieved from the NZDL Niupepa collection, the newspaper text in HTML format is parsed to identify any text segments that match the location in the place name database. If the system finds the matched segments in the newspaper text, the HTML content is modified to insert links to relevant holdings in the map collection. Finally these links to the related map are provided to users. Users have the ability to turn the link insertion on or off (Figure 2.9).

2.4.4 Conclusion

HistoryMap system does not provide travel information while travelling. But HistoryMap system does provide a search function based on the location rather than a place name. A location triggered system can be more reliable than searching by the place name while travelling. This idea was used in the TER system. The TER system added longitude and latitude data onto its database to provide a search function as well.

2.5 The Tipple Application

This section details the design features and implementation of Tipple (Hinze et al, 2011), the name of the existing Android application. The existing Tipple Application is designed to make digital books available in a visitor's location context. The books are sourced from the digital library and can either be audio books or synthesised using text-to-speech. First we look at the scenario of Tipple, followed by the design features and implementation. TER system expanded the Tipple Application based on TER system requirements.

2.5.1 Scenario of the Tipple Application

Imagine a visitor, James, visits the Hamilton Gardens. Installed on his Android phone is Tipple, loaded with his profile and interests, which includes garden design and planting. Walking from Hamilton Gardens to the Paradise Garden collection, the "chirp chirp" sound made from his android phone alerts him to an area of interest nearby Hamilton Gardens a collection of world garden designs. When James checks his phone, he can zoom in or out on the map to see the details of this location such as; name, longitude and latitude. James decided to visit the paradise garden by using the Map. Once there he selected a chapter from the listing of books and chapters. He can now listen to the

speaking story while he is wandering around.

2.5.2 Design Features of the Tipple Application

This section introduces the design features of the Tipple application. The digital books include both audio books in MP3 and books in text stored locally.

F1: Playing Chapter-based Books (MP3) Chapter-based books are collected in chronological order. This service lets users know the order of the chapter and the location of other chapters in book.

F2: Text-to-speech Books from the local file need to be read to users. Books are loaded based on users location and profile. Books loaded with text-speech software will synthesise to voice.

F3: Text Display Users can choose to listen to the book, read the book or combine both together at the same time. Tipple supports displaying the text function on users' android phone's screen.

F4: Anti-Overlap When a new object is coming, users can hear the "chirp chirp" sound. Users can then choose whether interrupt the current book.

F5: User Profiles Users are able to set up their profiles with their interests. Tipple will respond to the users profiles based on users' current location and their interests.

F6: User Preferences Users are available to set up their preferences. For example, they can select "text-to-speech", or "text-display". If they choose "text-to-speech", they can start listening to the story once the books are available.

F7: Embedded Map system Tipple uses MapsForge, an open source library with a GoogleMap's inspired API that works with Open Street Map (OSM). Tipple implements the map as static files on the phone so the entire Tipple application can work without internet connection.

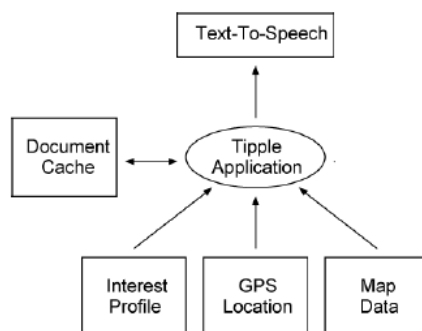


Figure 2.10: Components of the Audio Book Service within TIP and Greenstone. Adapter from “Making your digital library mobile with location-triggered audio books”, by Hinze, A. and Bainbridge, D., 2011, *Proc Workshop on Books*, p. 1-4. Reprinted with permission.

2.5.3 Implementation of the Tipple Application

Figure 2.10 shows the overview of the software architecture of Tipple application.

When the client sends the information to local data, such as where the user is and what potential places of interest might be nearby, the related information will be retrieved from the local data and stored in the document cache on the mobile devices. These documents are accessed locally as needed when the users move around.

For the map system, we use the MapsForge (mapsforge,2012). We store the map information as static files on the phone to guarantee the entire Tipple application will work without internet connection. Location mark-up was inserted using GPS coordinates as the centre of a circular area. Because all online maps provided satellite views that were either not available at the desired resolution or too complex to be used (with many trees, flower beds and lawns that were hard to distinguish), a raster map was created for the user study using Open Street Map.

Android phones come equipped with a Text-to-Speech API, which is used to get the Tipple speak out once it has the text information from its document cache and the location details from GPS.

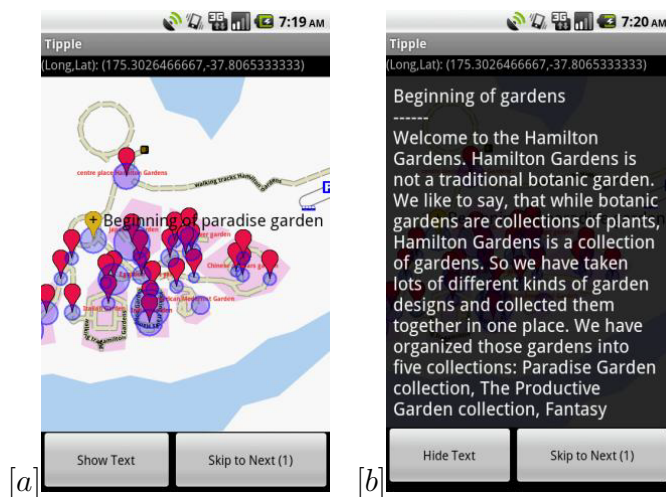


Figure 2.11: Screen shots for this Tipple system. From “Making your digital library mobile with location-triggered audio books”, by Hinze, A. and Bainbridge, D, 2011, *Proc Workshop on Books*, p. 1-4. Reprinted with permission.

The screen shots in Figure 2.11 are taken from an example of a user traveling around the Hamilton Gardens in New Zealand. Consider the scenario of this system in Section 2.5.1, and imagine the user, James, is interested in the Architecture. Tipple has been equipped on his Android phone. James travels around the Hamilton Gardens. When James has passed through the first place of interest; the Paradise Gardens, the mobile phone sounds “chirp chirp” to advise something of interest based on his prescribed interests. James has a number of options; Audio only mode, Text only mode, Audio and Text mode. He chose not to play the audio commentary just yet, and continues on to the second location of interest. This is the point the screen shot shown in Figure 2.15 [a] has been taken. From here James can start to listen to the audio while walking around until he arrives at the third area of interest. Again, he will hear the “chirp chirp” sound and he can choose interrupt the current audio by clicking the “Skip to Next” button once or continue play the current audio.

Text only mode has two buttons; “Show/Hide” text button for displaying the text of the related story, while “Skip to Next” allows users to skip to the next location in the queue. Figure 2.11; screen shot (b) is taken from the same point of James’ tour as before. This mode alerts user when user has arrived at the location of interest. There is also an Audio and Text mode, which is not described here.

2.5.4 Conclusion

The Tipple Application combines location and book together, and implements the audio player features. These have been expanded onto TER system. But the Tipple application has narrowed its usage onto one static book stored locally. While travelling, users can’t choose books based on their interests and requirements.

2.6 Summary

To sum up, these five applications are successfully run based on their specific requirements, but are not efficient enough according to our five basic requirements discussed in Section 1.4.

Table 2.2: Evaluation results of related work

Section	Apps Name	R1	R2	R3	R4	R5
2.1	GreenStone 3	+	-	-	-	-
2.2	TIP 2.0	-	+-	+	-	-
2.3	Audio book	+	-	-	+-	+
2.4	History Map	+	-	+	-	-
2.5	Tipple	-	+	+	+	-

R1:Connect to DL, R2:Mobile Devices, R3:Location Based, R4: Text to speech, R5:Audio books. (Legend:support of Requirements) + Yes, - No, +- Partially

Table 2.2 summarises their advantages and disadvantages according to the five requirements R1 to R5 discussed in Section 1.2. Greenstone3 is an open source

software used to build a digital library collection. It does not match requirements R2, R3, R4 and R5. HistoryMap system is a location based system which connects to the digital library collection, but it does not target on the tours system. The remaining three applications aim to provide travel information in different ways, but none of them built their customised digital library. Tipple stored a static data locally, TIP and Audio Book Apps communicated with their server through internet. TIP and Audio Book Application were able to run on the mobile device using the mobile device browser. Tipple application run on a real mobile device. Among these five applications, Tipple and Audio Book Application both implemented the text-to-speech function and supported audio books functions.

Chapter 3

Requirements and Design of TER System

Travelers are most likely to be away from their normal routine, out of their usual network and maybe even outside of their standard rate plan. In the age of the smartphone, there are things to undertake when travelling, such as location, GPS, maps and pictures etc., so the traveler gets a rich experience from the mobile device.

Using Android devices with extra software means that people can turn their smartphone into great travel companions. Instead of just a phone, users will be carrying a mobile digital library, an E-Book reader, a travel guide, a note taker and much more.

This research focuses on providing useful tourism travel information while using the mobile system, and building a *Travel E-Book Reader* with Digital Library application (TER). The TER system builds on and extends concepts from both previous Tipple (Section 2.5) and Greenstone (Section 2.1). It runs on the Android phone. The Android platform is not largely different to other cell phones since they all have GPS and Camera functions.

This section focuses on requirements of the TER system and then discusses the architecture and interface design of the TER system in terms of mobile

application. Finally it explains the communication issues between the server and clients.

3.1 Requirements and Analysis

Taking the sample scenario from Section 1.4, and based on the previous work in the area (Chapter 2), a functional requirements analysis was undertaken to establish the features and characteristics of the software to be developed. The first five requirements are similar to the general requirements introduced in the Section 1.2. In addition to these five statements, there are more specific requirements drawn from the related work experience and case study in the event “The Stations (See Chapter 7). From the experience of the Tipple application study (Section 2.5), R6 and R7 describe usage of photographs in the application. From the related work of TIP (Section 2.2), Tipple Application (Section 2.4) and audio books application on desktop (Section 2.3), R10 allows users to choose a book from a list of books. R11 and R12 are about how to present book information on the screen. R8 and R9 come from user evaluation in the event The Stations. Once connected to the server the R13 issues are required in general.

R1: Connect to digital library There are a few different digital library collections which will be run on the server, for example text collection, and image collection. These web based XML metadata will be transferred to the mobile devices over mobile network.

R2: Run on a mobile devices This application explores the role of audio books for mobile users. It will use mobile GPS features, mobile camera and touch screen to enrich the mobile services.

R3: Support location-based system This mobile application offers location-based information. All audio books and other services such as taking pictures will be based on location.

- R4: Provide text-to-speech and mp3 player function** To reduce the limitation of mobile devices, (such as screen size, limited input options, and distracted attention) this mobile application provides real text-to-speech function, which can convert text to speech. The Media Player also supports mp3 files.
- R5: Play audio books** This service lets users select related audio books or chapters based on their current location. It gives users control over the audio playback.
- R6: View pictures** This service allows users to view pictures related to the content of each chapter. The pictures are stored in Digital Library using Greenstone Version 3. This requirement comes from experiences of the case study in Tipple Application.
- R7: Take pictures** Users can take pictures for some chapters which do not have related images in the digital library collection. These pictures can be saved and viewed later.
- R8: Read content** This service allows users to read content for other places in their own time, not only for the place they are currently in. Also, the text has to be readable in a friendly design.
- R9: Touch interaction** All buttons and icons are effective when users touch the screen. All the text area is available to scroll up/down.
- R10: Choose book** Users are able to choose one or more books to read or listen to while visiting a particular place. Other related text or images can be updated as well.
- R11: Switch on/off travel path** This service allows users to switch the travel path suggested by this electronic guide on or off. Several paths are clear and easy for users to follow.
- R12: Order book based on the location** This location-triggered system provides book order based on the location. This order can guide users as

to the way to travel around. The map has to show users current situation.

R13: Minimise download data The study considers two ways to run it: One is running on the server on which the Greenstone server has been installed in the mobile device. The other way is to run as a client which connects to the Greenstone server. In both situations data transfer over mobile network may be slower than desktop, and data is likely to be more expensive (in second situation), so the application needs to minimise the requirements to load data over the network.

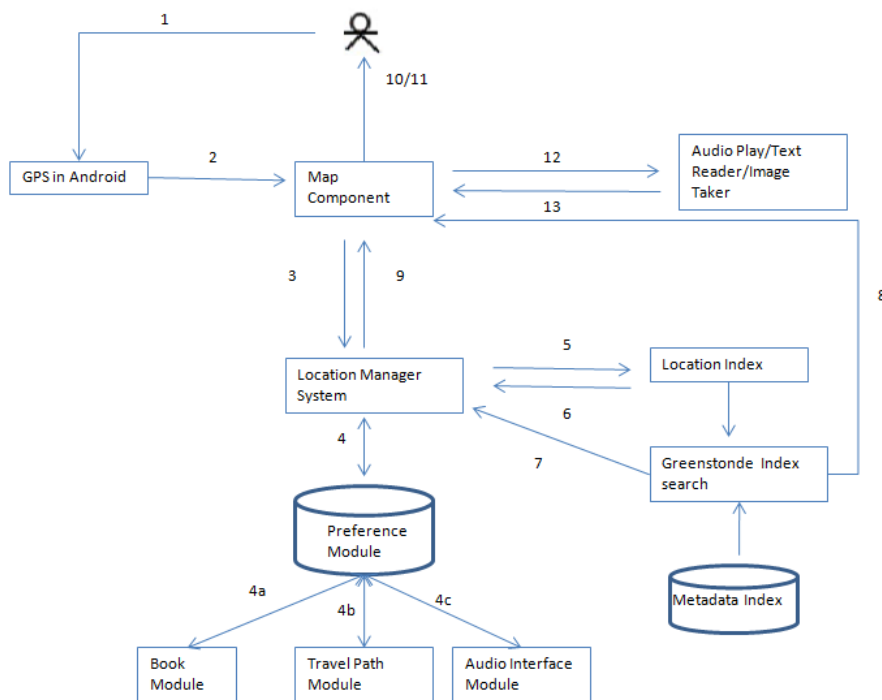


Figure 3.1: Build GPS Location and Location-based index

3.2 Design of the TER System Architecture

Figure 3.1 provides an overview of the architecture of the TER system. The TER system contains a Map Component, Location Manager System, Audio Player system, Image Taking System, User Preference Module, and Digital Library system. All these components interact with the map and deliver travel

information to the end user. This section describes the Travel E-Book Reader procedures step by step, as numbered in Figure 3.1.

- (1) While users walk around, the GPS location will dynamically identify the users location details.
- (2) The GPS data is sent to the Map component. On the mobile screen, the map is displayed with the user location.
- (3) The GPS location details will be sent to the Preference Module.
- (4) The Preference Module accesses the user profile database (to retrieve the users interest).
- (4a,4b,4c) The Preference Module contains the Book Module, Travel Path Module, and Audio Interface Module. The Book Module allows users to choose books to be read or listened to. The Travel Path Module provides users with an electronic guide for their travel on the map. The Audio Player Module gives users flexibility to choose to read, listen, or both.
- (5) The Preference Module sends: (a) the GPS to the location index, (b) the user profile information to the Greenstone index.
- (6) The location index identifies books and book chapter IDs that are related to the surrounding location of the user and sends them back to the reminder.
- (7) The greenstone index identifies books and chapters that match the user profile and sends them to the Location Manager System.
- (8) The results of step 5 and 6 are combined in an intersection, finding all books and chapters IDs at the user location that match the user profile. These IDs are used to retrieve the metadata of the books and chapters from the metadata index.
- (9) The Location Manager System sends the locations and the metadata to the map.

- (10) The map displays the locations and data to the user (icon, chapter or map).
- (11) If a user comes close the location of a chapter, the map will notify the user by changing colours on the screen and using a special sound such as chirp chirp.
- (12) The user selects a place on the map, which then displays the content and image.
- (13) The chapter is played as an audio book (and/or shown in a dialogue), or displays the related image (loading the content/image from Greenstone (step 13)).
- (14) The user can take pictures at the place, where there is not a related picture loaded from Digital library, and can then save and view it.

3.3 Design of Greenstone Digital Library

Assume the TER system has been installed in the users mobile phone. The system detects the users location by using the GPS function built in to the mobile system, then the places of interest are automatically sent to the Digital Library. The system requires the book information, such as book name, section name, section content.

The Digital Library will be built in the Greenstone software version 3. It will consist of text and images, which will be loaded to the mobile devices. Within the Greenstone Digital Library, the GPS data, including the longitude, latitude and radius of the places, will be inserted into metadata and onto the collection. The Digital Library will allow the GPS index search features. All these metadata, such as book name, section name, section content, and along with the GPS data, will be transferred onto the mobile device.

The TER system will explore different book usage on the mobile devices, so the Digital Library will consist of several books and images. Each book must contain more than one section. Those sections will then link to the places and present to the mobile user.

3.4 Communication between server and client

The TER system will communicate with the Greenstone Digital library server. The Greenstone software has been run more recently on Android tablets. This makes it possible to run a completely self-sufficient version of the tourist information system on the phone. Currently the TER system runs on Android mobile devices and Android users act as clients.

Figure 3.4 shows the communication between the server and clients is via the HTTP protocol over the mobile internet connection. Even when the Greenstone Digital Library component is installed on the phone, communication is still carried over HTTP as this simplifies the software design. HTTP protocol can transfer the web based XML data between servers and clients, but it can be expensive and can slow down if the connection is open all the time. The communication issue will be an important issue in the TER system.

3.5 User Experience Design

The TER system targets mobile users, so the most important step toward Android application is to consider its accessibility and usability. There are some issues which arise when developing applications on mobile devices, such as screen size, touch interaction, slower speed, (compared to desktop in general) and other user activities such as rotation of the phone.

The following section will discuss application layout design, navigation design and user interaction design of the TER system, such as Map layout, Icon layout, Book Order layout, User Travel Path layout, Image/Text View layout,

User Preference layout and User Interaction layout.

3.5.1 Layout Design Issues

The mobile screen size limits how much information can be displayed in a readable manner. The TER will combine map and other layout components together to deliver the message to end users.

3.5.1.1 Marker on the Map

When starting to design the interface of the mobile location service application, the design of the graphical marker on the map stands out as a critical part of the whole design interface. Each marker not only provides geographic context, but also links to deeper sets of data. So markers serve as a gateway for users to receive deeper information.

Specifically the marker design needs to pin-point to an exact location on the map, and present an easy touch target for users to effectively access the deeper information. The implementation will show the particular marker icon. Also, there will be different markers for different purposes. For example, a marker for the current place, and markers for book content from different books. All markers should follow the above rules to match the best look of the mobile devices.

3.5.1.2 Book Section Order

Initially Open Street Map has already displayed the street name and place name. The map with different colors and words makes the background colourful, so the mobile screen isnt suitable to write each section name with the order in a traditional content table. The limited screen size and the map background requires less words on it. It requires a friendly visual design with limited space

to describe the book section order, and book name.

3.5.1.3 User Travel Path Layout

Wherever the traveler goes they may need to know their current location and the trip details. Traditionally the travel book guide shows them on the map, and has text about the trip, but there isnt a way to mix them together dynamically when travelers move around. Mobile devices have this advantage over the book, so the study utilises the GPS function and map function to draw the travel path onto it. Travelers may follow various trips so the travel path has to identify the different trip based on user preference.

3.5.2 Navigation Routes

The TER system will always present the map on the screen, but it will at times need to deliver messages to users. How does the user navigate to other views via the cellphone? For example, the user may like to read the story, change their preferences setting, etc. The mobile devices provide some special features to interact with the phone. For example, the mobile devices provide menu functions, arrow up/down functions, move the screen, touch on the screen and long press on the screen. These features might be useful to provide an easy way to navigate to another view.

3.5.3 User Interaction

On smart phones interactions are likely to be touch-based: no mouse over/enter capability for tool tips, trigger event etc. So design of the mobile application needs to consider that all clickable targets need to be obvious, and it must have buttons/icons which can be touched in the application, sufficiently large to enable accurate selection.

Unlike a desktop application, mobile devices support different events, such as long press events and onTap events, and the way navigate to other screen is quite special, via menu features. In terms of rotation, the mobile devices are easily rotated and the rotation can impact the running application. Hence the mobile application should be aware of the interaction between users and the mobile device.

3.5.4 Android Permissions

Mobile applications can be a privacy nightmare via the permission setting. For example, some mobile applications with internet connection permission can upload a users personal data, such as private photos, contact data and calendar data, and send it to the remote server without the users knowledge or consent. Some services with the read phone call permission setting can cost users money and steal users account information. So it is reasonable Android application permissions requests are sometimes so invasive that it could scare a lot of users away. Development of mobile applications should be aware of the permissions setting, and try to limit any cost for the end user.

There will be more mobile application design issues encountered during the implementation stage, such as mobile memory size, and mobile storage. These issues will be discussed in more detail in Chapter 5.

3.6 Summary

All these design issues are carefully considered through the implementation phase. Following the architecture design, the TER system has been developed and tested. All Android design concepts were taken account during the implementation and evaluation stages. Following the design, the Digital Library was built successfully and fully implemented into the TER system.

Chapter 4

Interface of the TER system

As described in the chapter entitled Implementation, The TER system mainly consists of seven components. Within these seven components, the Location System is invisible but is shown on the Map System. This chapter introduces the TER system interface so it mainly focuses on the six visual components: Digital Library Server System, Map System, Audio Player, E-Book Reader, Camera System and User Preference System.

The Digital Library Server System provides all necessary data to be run on mobile devices. The Audio Player allows the user to listen to the audio and control the audio system. The E-Book Reader provides readable content on the screen. The Map System is the basic, yet the most important, interface of the TER system. The other systems can efficiently function based on the location given on the map. The Camera System relies on the mobile device hardware and allows the user to take photographs of visited places and save them. Mobile users can set their preferences through the User Preference System and then change the system settings and functions. Each part is not independent as they rely on each other to integrate with the TER system.

This section gives a visual description of the TER system. In the case where the TER system has been adapted for standalone applications, which communicate with local XML files, the interface and structure of these XML files will be discussed in the case study and implementation sections.

```

C:\Windows\system32\cmd.exe - gli
C:\Program Files\Java\jdk1.6.0_26\bin;C:\Program Files\Android\android-sdk\tools;C:\apache-ant-1.8.2\bin;C:\Program Files\QuickTime\QTSystem\;C:\Program Files\CollabNet\Subversion Client;C:\Program Files\PHP\;C:\Program Files\MikTeX 2.8\miktex\bin;C:\Python26\Scripts;C:\Python26\;C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\Program Files\Common Files\Roxio Shared\DLLShared\;C:\Program Files\Common Files\Roxio Shared\OEM\DLLShared\;C:\Program Files\Common Files\Roxio Shared\OEM\12.0\DLLShared\;C:\Program Files\Roxio\OEM\AudioCore\;c:\Program Files\Microsoft SQL Server\90\Tools\bin\;C:\Program Files\Java\jdk1.6.0_26\bin;C:\Program Files\Android\android-sdk\tools;C:\apache-ant-1.8.2\bin;C:\Program Files\QuickTime\QTSystem\;C:\Program Files\Android\android-sdk\platform-tools

Setting environment for using Microsoft Visual Studio 2008 x86 tools.

C:\Users\sp76>cd gs3-svn/gli
C:\Users\sp76\gs3-svn\gli>gli

Greenstone Librarian Interface (GLI)
Copyright (C) 2008, New Zealand Digital Library Project, University Of Waikato
GLI comes with ABSOLUTELY NO WARRANTY; for details see LICENSE.txt
This is free software, and you are welcome to redistribute it

```

Figure 4.1: Greenstone Librarian Interface)

4.1 Web Based Digital Library Collection Interface

The TER system uses Greenstone version 3 to build the Digital Library collection. The Greenstone Librarian Interface (GLI) runs on Apache Tomcat, which starts the Digital Library server. Figure 4.1 shows the GLI running interface.

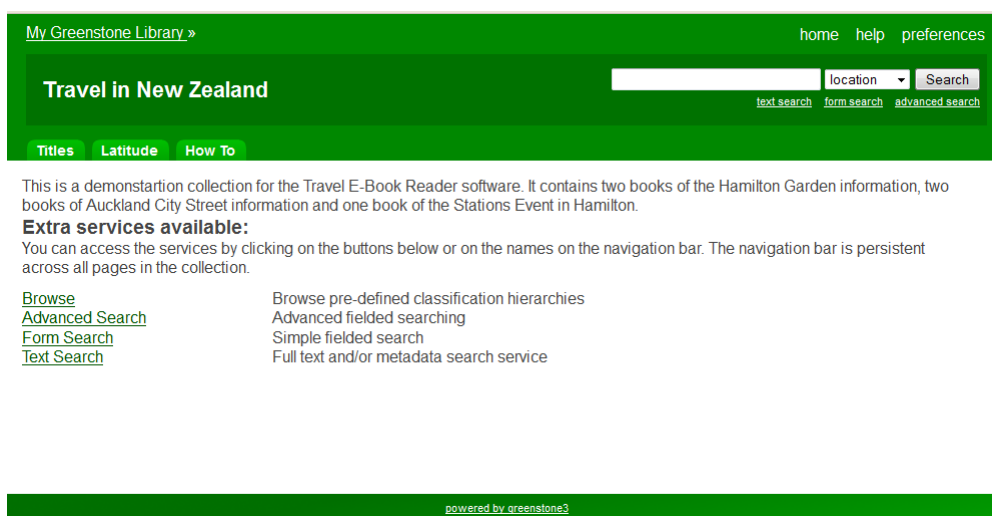


Figure 4.2: Travel in New Zealand Collection Home page

The TER has been tested in real life with real data. This case study will be discussed in Section 5 and Section 6. All the text data was input into one collection called Travel in New Zealand. This was a demonstration collection for the TER software. It contains two books of Hamilton Gardens information used in the implementation stage, two books of Auckland city street information used for the Auckland Poetry on the Pavement event and one book of the Stations used in 2012 Stations Event in Hamilton. These five books are Paradise Garden Collection in Hamilton, Welcome to Hamilton Gardens, Poetry on Auckland City, Queen Street in Auckland and The Stations. The content of the books Paradise Garden Collection and Welcome to Hamilton Gardens is mainly from Hamilton Gardens website. The content of the book The Stations, is adapted from The Stations website and The Stations booklet. The content of the book Poetry on Auckland City is from the Poetry on the Pavement event advertised on the Auckland City Council website (Auckland, 2005). The Queen Street in Auckland book content is extracted from the Wikipedia website (Wikipedia, 2012).

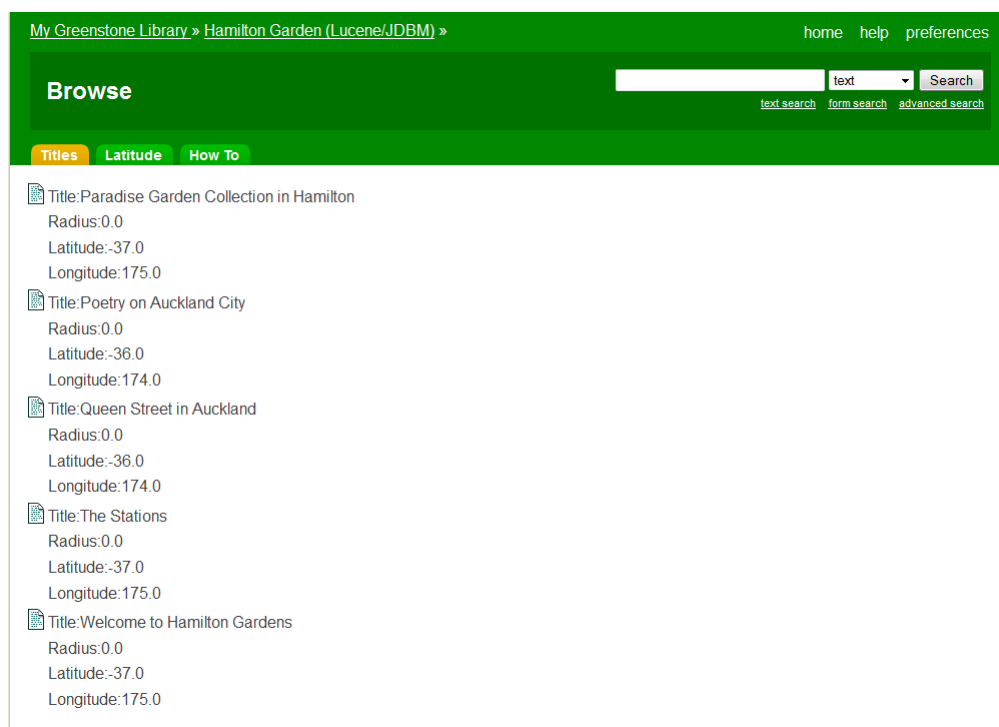


Figure 4.3: Books in the Collection

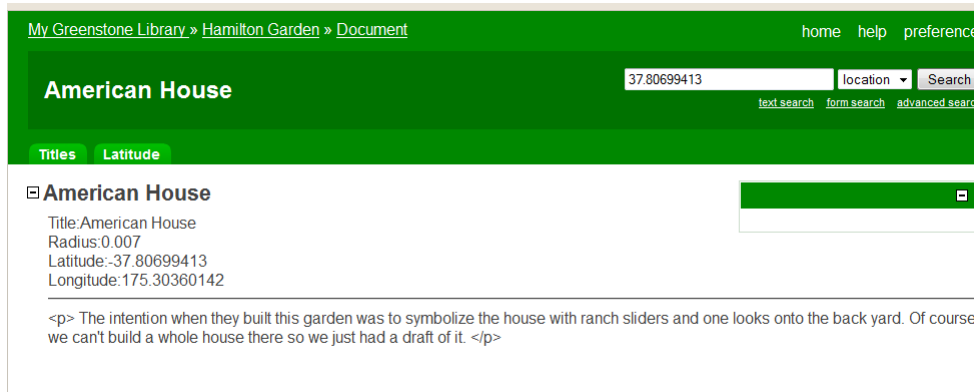


Figure 4.4: Search based on the GPS

Figure 4.2 shows the home page of the Hamilton Gardens collection, which was opened on the browser, Firefox version 13.0. This page introduces the collections purpose and usage. Figure 4.3 presents the five book titles. All book icons are clickable and link to details of the selected book. Figure 4.6 shows the book.

Here are the chapter and content details for the Paradise Garden Collection. The top right hand side is the content of the book. This allows users to navigate to the selected chapter. The search by location feature is used. Figure 4.4 is the result of the search for location 37.80699413. One image collection has been build up for TER system shown on Figure 4.5. This collection contains images, which can be downloaded onto a client based on the image name. These pictures are from from the Hamilton Gardens website (Hamilton, 2011).

4.2 Map Interface

The maps used in the TER system are produced by Open Street Map. The map handles all user input and touch, such as moving and zooming. For example, Figure 4.7 is the magnified map and only displays the main path. Figure 4.7 b shows more specific information, such as street and place names and place icons on the map after zooming in. These off line maps are stored locally to address internet retrieval issues.

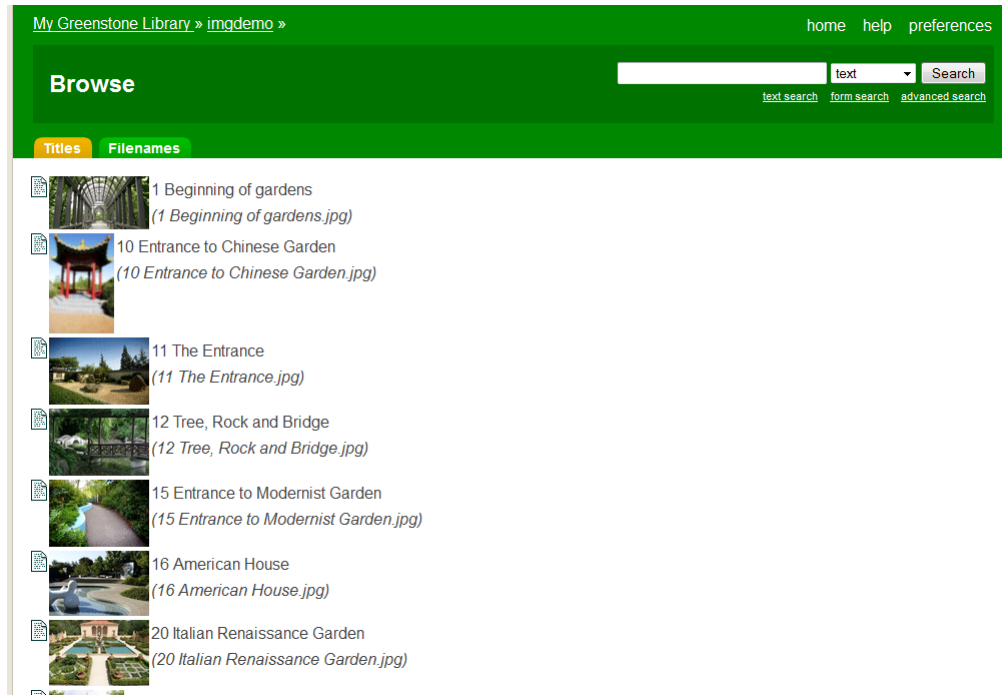


Figure 4.5: Pictures of the Paradise Garden

Once the user turns on the internet collection, the book and each sections content will be loaded from the server. The map will then change to the following interface shown in Figure 4.8.

Assume user is walking in the Hamilton Garden. While he continue to the Paradise Garden Collection, The Android phone equipped with the TER reminder system will remind him and the mobile screen will be displayed as the below Figure 4.9. The icon (Number 1 in this case)with lighter pink color, which display the user' current location on the map, has been changed to darked red color, and this section's name "Beginning of gardens", which related to the entrance of the Paradise Garden Collection, shown on the map.

All map icons show the stories available to read or listen to, based on the marked locations. All numbers represent the order of the book sections. Picture b is the magnified version of Picture a in Figure 4.8. This assumes the user is walking in Hamilton Gardens. While they continue on to the Paradise Garden Collection, the android phone equipped with the TER reminder system will notify them and the mobile screen will be displayed as in Figure 4.9.

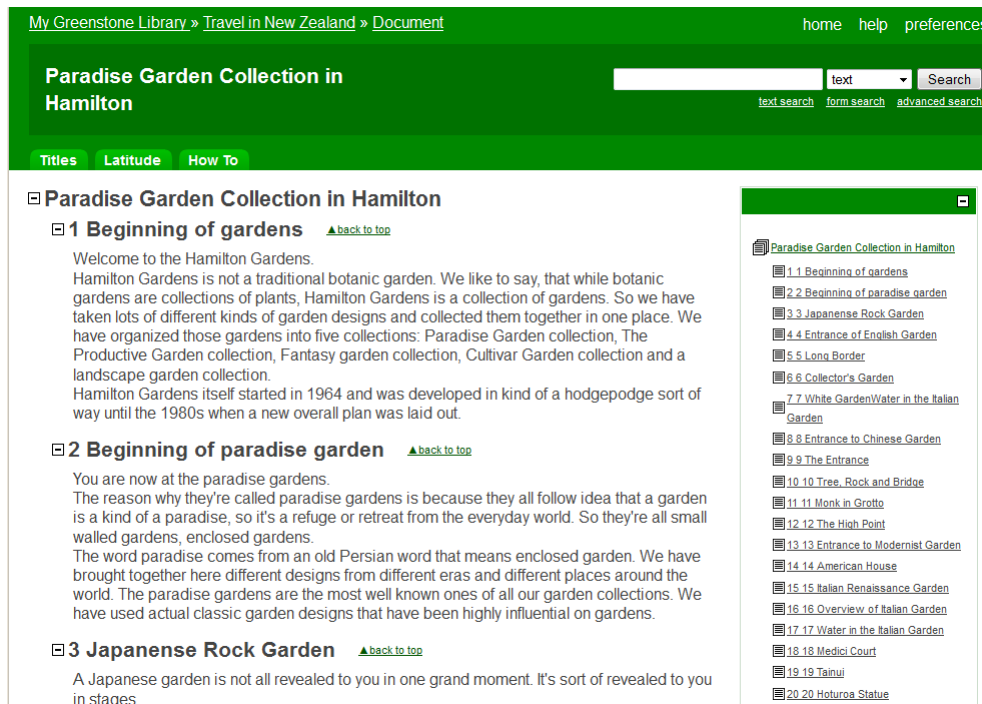


Figure 4.6: Paradise Garden Collection Content

The icon, number one in this case, with the lighter pink color, displays the users current location on the map. This has been changed to a darker red color. This sections name, Beginning of Gardens, relates to the entrance of the Paradise Garden Collection, shown on the map.

When the user turns on the travel route, the map draws lines between different places and shows the user where the next spot is and the relationship of all possible travel places. This interface, (Figure 4.9 [b]) mainly designed for users who would like to follow the book order to travel around in Hamilton Gardens, Figure 4.9 [c] also shows the travel path for visitors in the Auckland Poems on the Pavement event.

4.3 Audio Player Interface

The TER system allows users to listen to the audio files from a selected section. This feature works through the Audio Player Interface.

The Audio Player Interface contains three main buttons: Play/Pause, Show

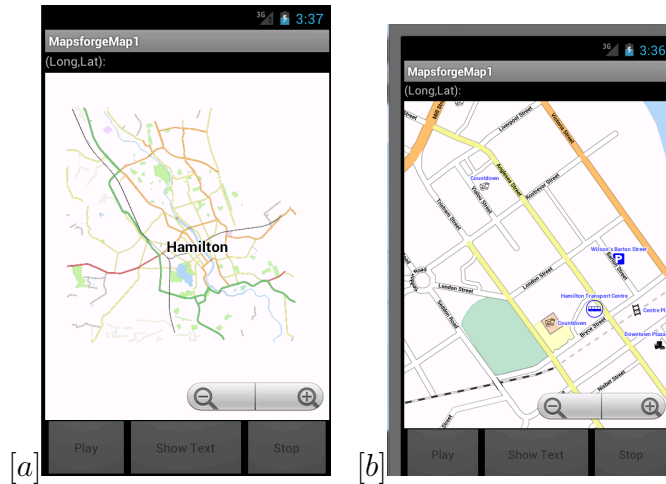


Figure 4.7: Map Interface

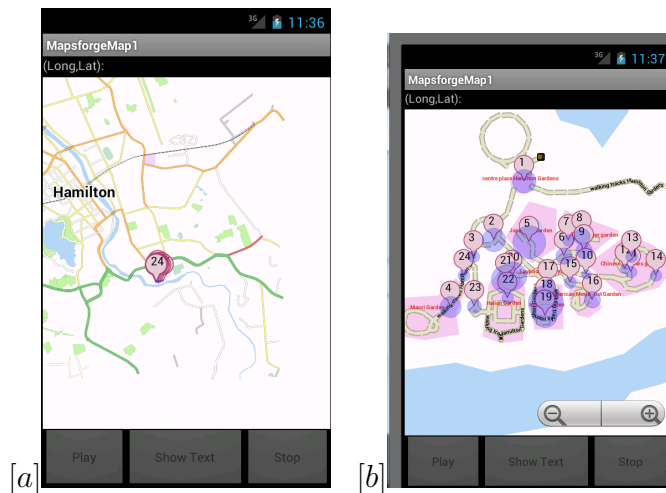


Figure 4.8: Display Location on the Map

Text/Hide Text and Stop/Skip to next. These button labels dynamically change. Initially all buttons are greyed out (Figure 4.10 A) and only activated when users reach the range of each place. When the TER reminder system sounds “chirp chirp”, the first button changes to Play, and the stop button is enabled on the bottom of the map (Figure 4.10 B). When the user touches the play button then the play button label turn into Pause (Figure 4.10C). Figure 4.10 D shows when the user presses the Pause button and Show Text button. In that situation, users hear the TER reminder “chirp chirp”, ignore it, and continue to the next place. They hear the TER reminder chirp chirp again and press the Stop button, which turns into Skip to Next (1) (Fig-

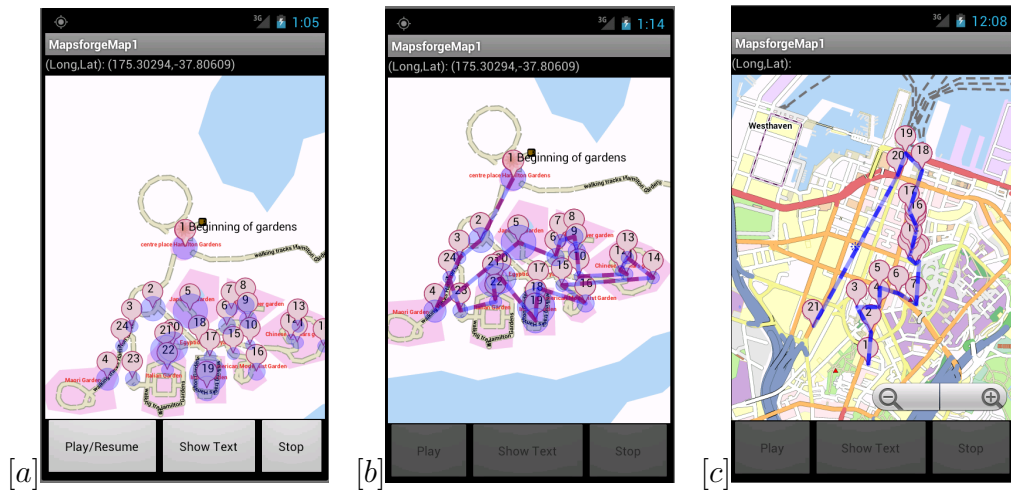


Figure 4.9: Map Location and Travel Route

ure 4.10 E). Assuming the user ignores it again and continues to the next place and hears the reminder “chirp chirp” a third time, then Skip to Next (1) turns into Skip to Next (2)(Figure 4.10 F).

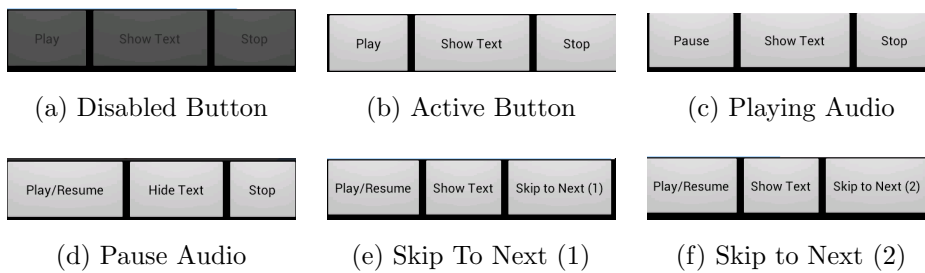


Figure 4.10: Audio Player Interface

4.4 Reader Interface

The TER system gives users flexible choices in using the Digital Library collection. Either reading or listening is free to be chosen according to the user’s requirements. The TER system provides two exiting actions to access text content. One of the actions is to press the “Show Text” button.

When users walk into the place and the TER reminder system shows them the story available for reading or listening to, they can read the content by

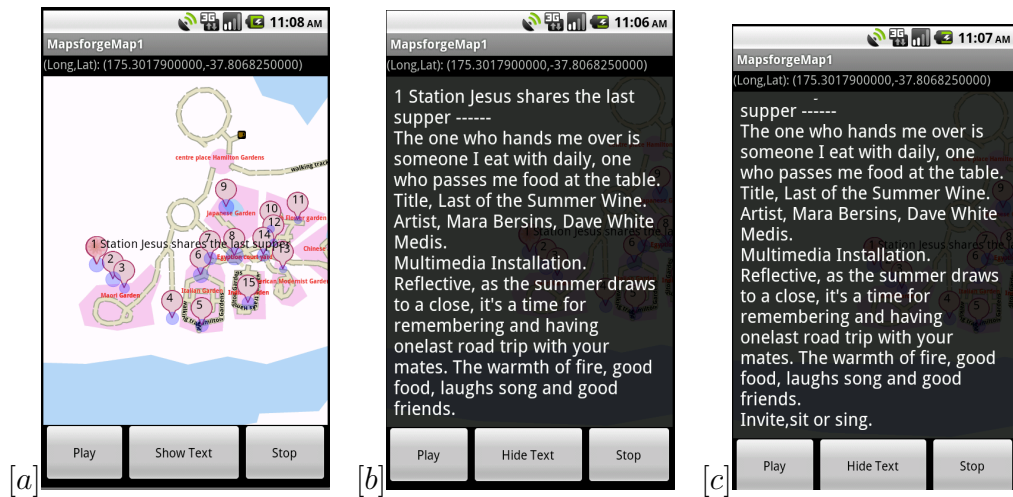


Figure 4.11: The Stations Reader Interface

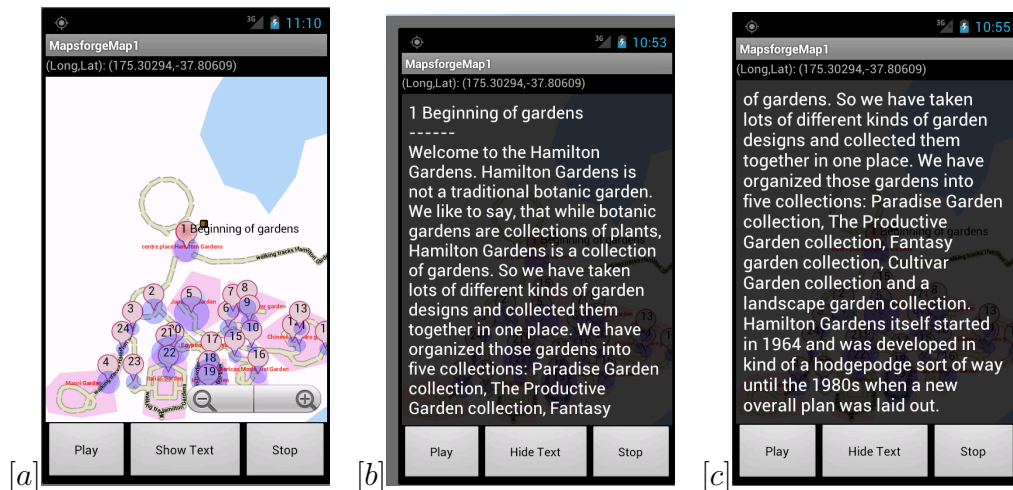


Figure 4.12: The Garden Reader Interface

pressing the Show Text button. The content, based on the user’s current location and pre-loaded from the Digital Library can be displayed on the mobile screen. Depending on the materials, the content shown on the screen can have a different format. Figures 4.13 is the poems reader interface, Figure 4.12 is information about Hamilton Gardens. Figure 4.11 is the explanation of The Stations event. Each screen shot contains three pictures: [a], b) and [c]. All pictures are [a] shown when the user is close to the place. The other two pictures are shown when users press on the “Show Text” button. Picture [c] is a scrolled down view of picture [c].

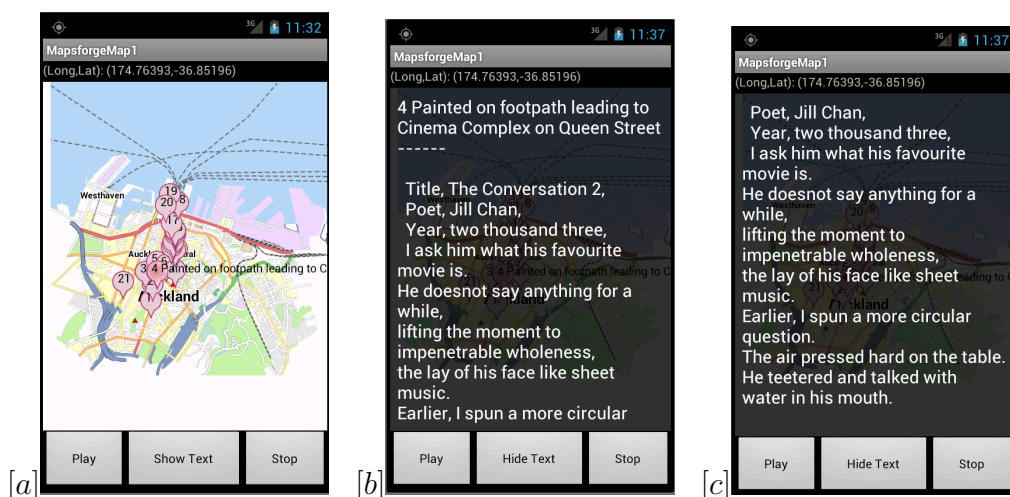


Figure 4.13: The Auckland Poems Reader Interface

Poems have more break lines within several paragraphs. The Stations list a few important messages about the art, such as artist, materials used and the bible story of the event. The garden information simply explains the information about each paradise garden.

Another action is to touch icons on the map. Users are able to read all contents in their own time by touching the map icon. The related book content, preloaded from the Digital Library, will be displayed on the screen once the user touches the icon. This way, the content displayed is slightly different when pressing the button.

Figure 4.14 shows the poems displayed when touching on the map icon with number four. Picture [a] does not show any highlighted icon on the map. Users can touch any icons on the map. In this case, the user touches on the icon with number “4”, then Picture [b] is displayed on the map. Picture [c] shows a version of the picture seen when the user scrolls down. [b].

When comparing Figure 4.13 and Figure 4.14, the text format are different in terms of break lines, paragraph and font size. Figure 4.14 is more accurate when compared with the original poem format. When the user touches the

OK button on the screen, this text reader will disappear and change back to the picture [a].

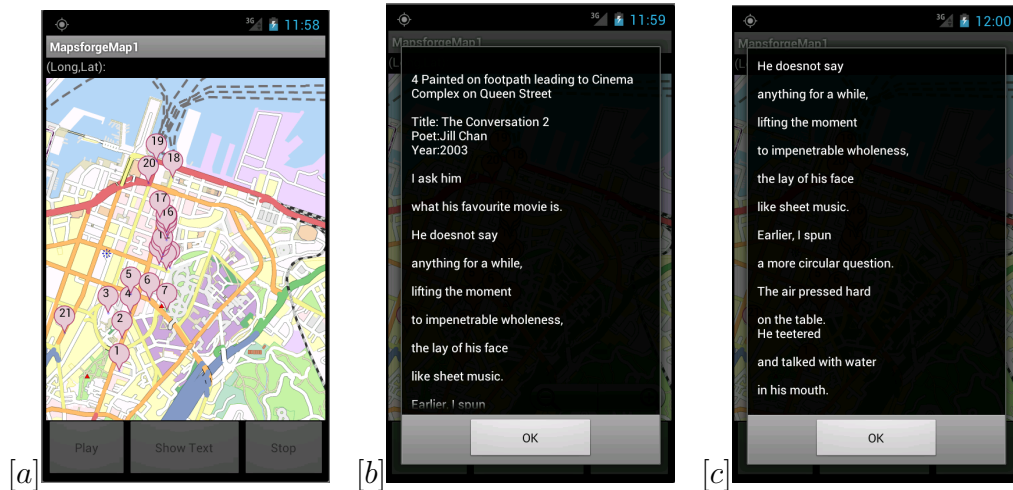


Figure 4.14: The Auckland Poems Reader Interface via Touching Icons

4.5 Camera System

The TER system allows users to view pictures which can be downloaded from the Digital Library server. When users touch on the map marker, the pictures, which can provide visual information of the stories or explanation, will be shown on the pop up window. The pop up window can contains pictures, text and buttons. If the pictures are available from the server, then there will pictures, text on the center and one button labelled “OK” on the bottom. This study has built the image collection for the Hamilton Garden. The Hamilton Garden books contains twenty-five chapters in its book, and thirteen pictures. Figures 4.15 shows when user touched on the map icon with number 5, 20 and 22, then the related information including the pictures, titles, stories will be displayed. In this case, when users scrolled down the whole stories will be available(Figure 4.16).

The TER system allows users to view pictures, which can be downloaded from

the Digital Library server. When users touch on a map marker, the pictures, which provide visual information or explanation of the stories, will be shown in the pop-up window. The pop-up window contains pictures, text and buttons. If the pictures are available from the server, then there will be pictures and text in the centre, with one button labelled OK on the bottom.

This study has built the image collection for Hamilton Gardens. The Hamilton Gardens book contains twenty-five chapters and thirteen pictures. Figure 4.15 shows what happens when the user touches the map icon with number 5, 20 and 22. The related information including the pictures, titles, stories will be displayed. In this case, when users scroll down, the full stories will be available (Figure 4.16).

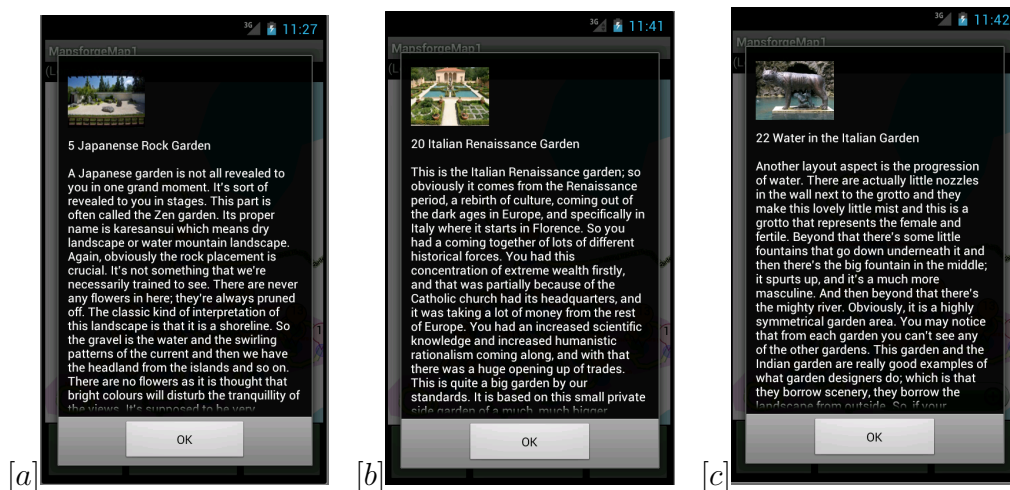


Figure 4.15: Pictures loaded and viewed

The TER system provides camera features as well. When there are no pictures available from the server, the POP UP window will contain the button labelled Take Photo at the bottom and provide camera functions. Where the story is very long, the button is visible when the user scrolls down. Figure 4.17 shows screen shots of both long and short stories.

When the user presses the Take Photo button, the TER system will launch the camera function. The following figure (Figure 4.18) shows the camera system

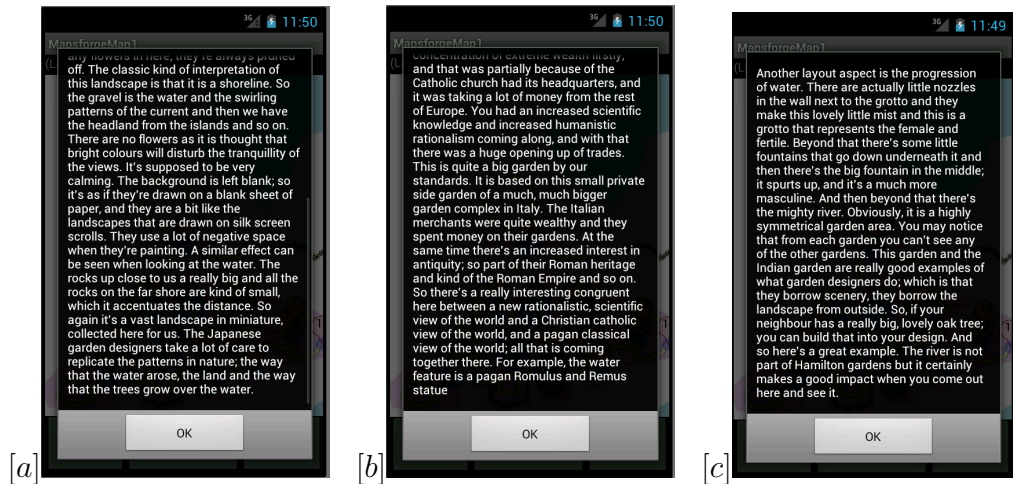


Figure 4.16: Scrolled view of POP UP Window

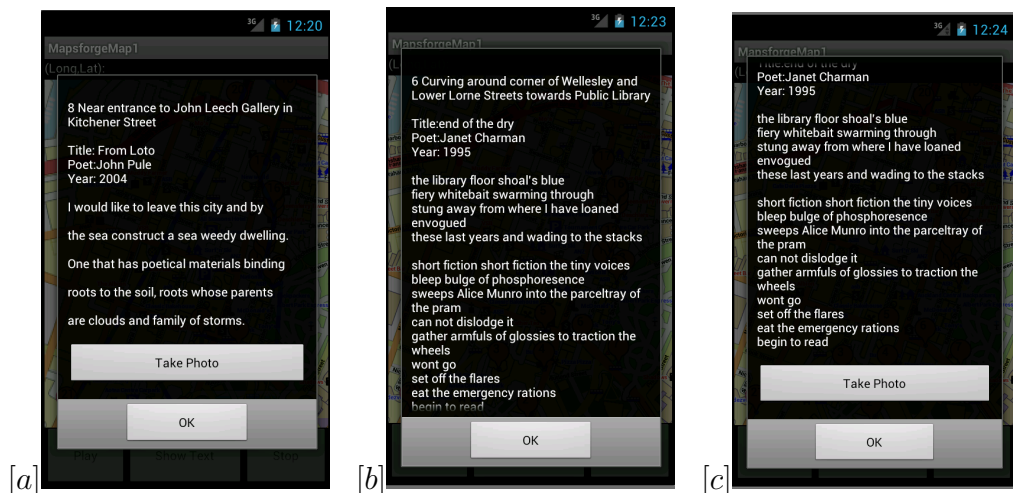


Figure 4.17: Scrolled view of POP UP Window

running on the Galaxy Nexus phone during the Poetry event (See Chapter 7) in Auckland.

Figure 4.18 [a] shows the Take Photo button to the user. When the user presses this button, the android will activate its camera system. The feature for taking pictures, [b], shows the camera lens and scene on the emulator screen. There are two signs on the bottom: the circle icon means taking picture and the sign x means cancel the camera function. [c], the right tick on the bottom, shows the current image has been taken. When users click on the “x” sign on the camera interface, the TER system will switch back to screen [a] and cancels

the camera function. It will not save the pictures. Only when the user clicks on the tick sign of the camera system, the image can be saved and viewed as shown in figure [d].



Figure 4.18: Camera System in TER system

4.6 Multiple Books Version

The TER system connect to the Digital Library collection(DL), and the DL collection has more than one book at the nearby place. The TER system allows users to listen/read multiple books. Once user select multiple books on the nearby place, these book information will be displayed using different map icons. Figure 4.19 shows two books preloaded from DL server, both books are all talked about the Hamilton Garden, one book contains 24 sections and the other contains two sections. Figure [a] and [c] only shows the sections of the book available on the map, and Figure [b] and [d] shows both book sections and travel path for these two books respectively.

The TER system connects to the Digital Library collection (DL), and the DL collection has more than one book of nearby places. The TER system allows users to listen to or read multiple books. Once the user selects multiple books on the nearby place, the book information will be displayed using different map icons. Figure 4.20 shows two books preloaded from the DL server. Both books discuss the Hamilton Gardens. One book contains 24 sections and the other contains two sections. Figure [a] and [c] show the only sections of the book available on the map. Figures [b] and [d] show both book sections and

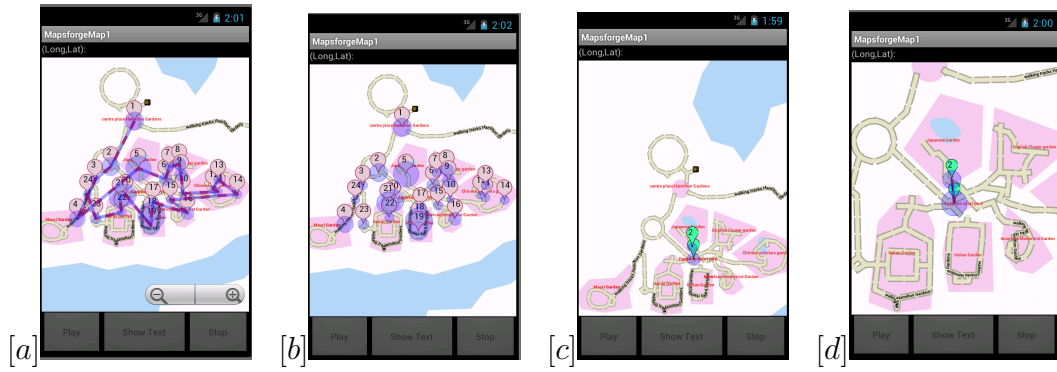


Figure 4.19: Multiple Books Loaded From DL

travel paths for these two books respectively.

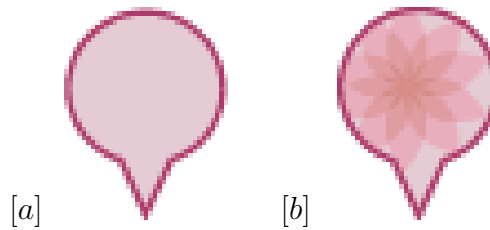


Figure 4.20: Map Icon Sets for Book A

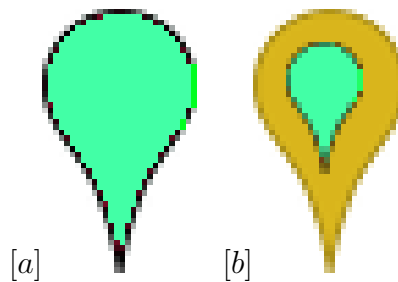


Figure 4.21: Map Icon Sets for Book B

When the user selects multiple books, the TER system will provide information for the books selected. Figure 4.22 [a] shows two books loaded from the DL server and [b] shows both the book sections and the travel paths on the map. Multiple books are differentiated visually by map icons and lines between those icons.

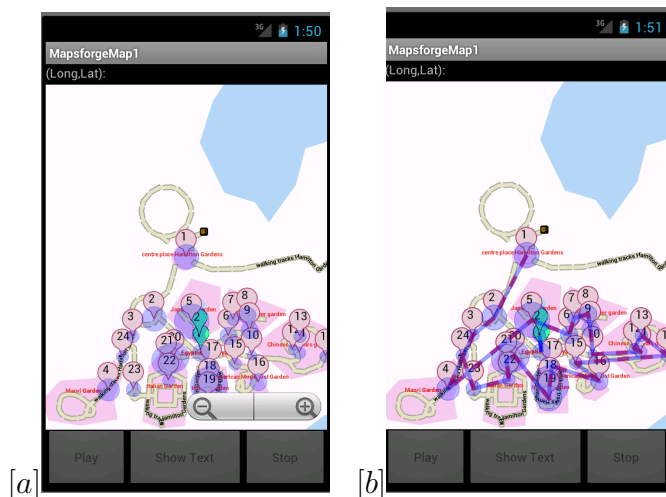


Figure 4.22: Various Books on the Map

4.7 User Preference Setting Interface

The TER system provides an easy way for users to set up their preferences. The preference setting includes the display scale bar on the map. The centre of the map on the screen shows all map icons, user trails, change audio settings, and selected books. As an android user, the common way is to go to the preference setting menu by click on the android menu item. However, it depends on the mobile devices used. For example, Galaxy Nexus users need to click on the three vertical dots on the bottom right of the menu bar. Galaxy S users just click on the menu then the preference setting menu will pop up. Figure 4.23 [a] will display on the screen for the user to check or change their preference settings. Once users select “Preference” then the preference menu (Figure 4.23 [b]) is shown and allows the user to check or update their preference settings. This section will explain each preference setting.

4.7.1 Show Scale Bar on the Map

The map in the TER system comes with an integrated scale bar. Once the user selects the scale bar, the scale information contained in the Open Street Map will be retrieved and displayed on the left bottom of the map screen (Figure 4.24).

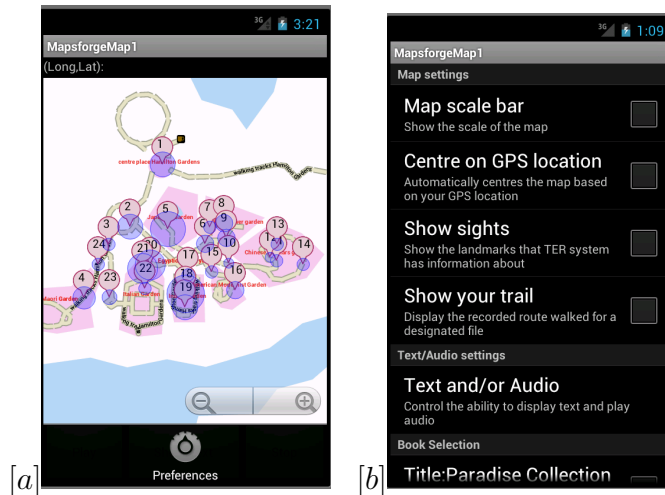


Figure 4.23: Android Preference Screen

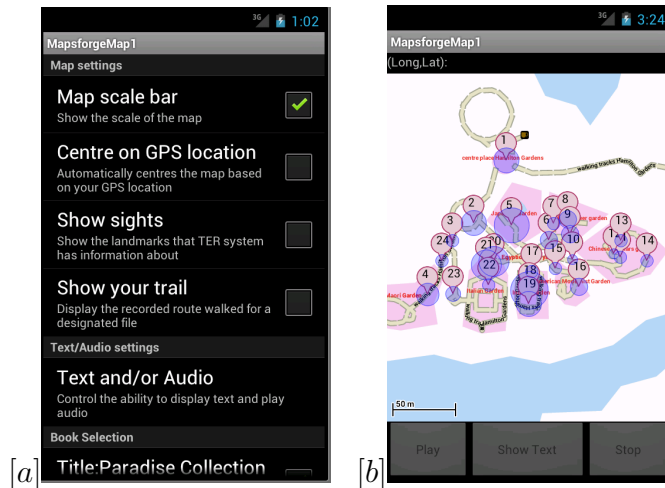


Figure 4.24: Show Scale Bar on the Map

4.7.2 Centre Map on GPS Location

The map will automatically centre itself based on the location, once the user switches on this option. On users' phone the Green Circle shows users' current location, where user go, the map will always be in the center and the current location will always be visible by using this option. Figure 4.25 was taken from Auckland in the event "The Pavement" (See Chapter 7). The green circle on the users' phone (with the a black circle which used to highlighted the green circle) showed where users were. When a user walked to other place the green circle moved following users travel path and the map appeared on the screen

was changed.

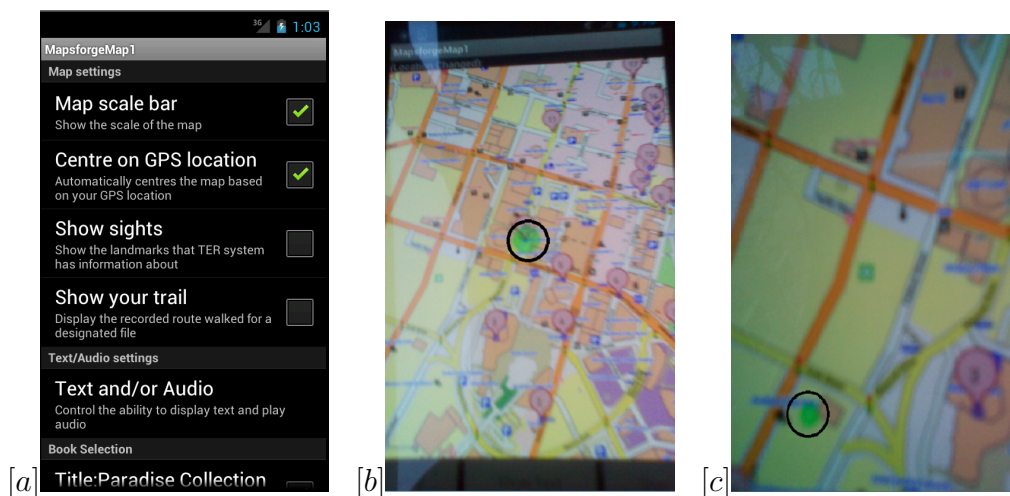


Figure 4.25: Centre Map on GPS Location

4.7.3 Show Sight on the Map

The user can turn the Show Sights function on and off using this check box on the preference menu. This option allows the user to control whether they show or hide all landmarks that the TER system has information about. Figure 4.26 [a] shows all landmarks from the TER system. Figure 4.26 [b] shows the map without the landmarks.

4.7.4 Show Travel Path

Android users can check their travel any time by turning on the option “Show your trail” from the preferences menu. This will show the travel path between each landmark. Each landmark connects to two other landmarks, except the start place and the end place. All these landmarks are in the order provided in the book from the TER system retrieved from the DL server.

Assuming the DL server talks about the book, then all landmarks will be in the same order as they are in the book. Figure 4.27 retrieved the book, Paradise

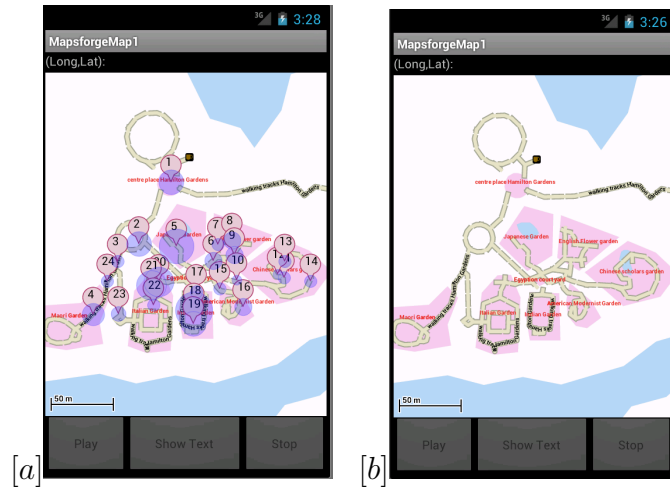


Figure 4.26: Show Sights on the Map

Garden Collection in Hamilton from the DL server. Those twenty-four places mentioned in the book are numbered based on the order they appear in the sections of the books. The landmarks labelled 1 and 24 are only connected to another landmark because they are the start and end points.

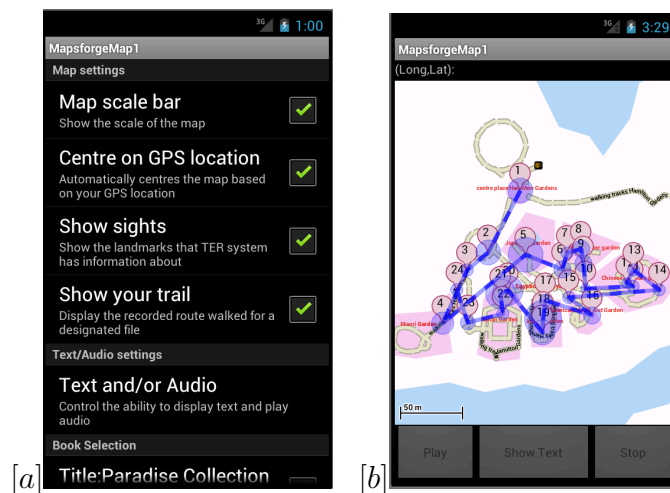


Figure 4.27: Show Your Trail on the Map

4.7.5 Audio Player Preference Setting

The TER system provides choices for listening and reading while users travel around. This feature is controlled by the user when the Audio Player Prefer-

ence settings are changed. Text and/or Audio has four options displayed as a group of radios in the Figure 4.28: Text Only (Figure [b]), Text with Audio (Figure [c]), Audio Only (Figure [d]), Audio with Text (Figure [e]). These four options do exactly what their labels say. For example, Text Only will allow users to only read text by adding two buttons, Show Text and Stop on the map. Text with Audio, will support either the display text or play audio by adding three buttons, Show Text, Play and Stop in turn on the map screen.

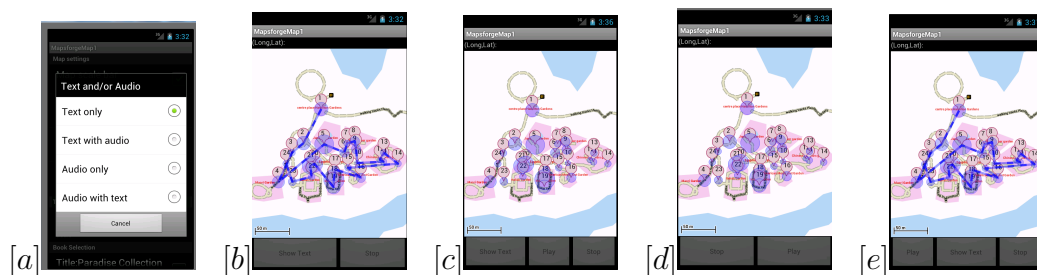


Figure 4.28: Audio Player Preference Setting

4.7.6 Book Selection

The TER system derives information from the DL server, which may contain more than one book of nearby the places to the user. The user can choose the book by using the Book Selection setting. The titles and summaries of the book content are displayed in this block. These books can be selected by checking one or more check boxes on the preference menu. Figure 4.29 [b] is shown when the user selects two books and shows the user trail [a] at the same time. During the running time, the user is free to change the book settings to rearrange the landmarks drawn on the map.

4.8 Summary

The TER system generates the interface dynamically so android users can change their preference settings to rearrange the TER system interface and

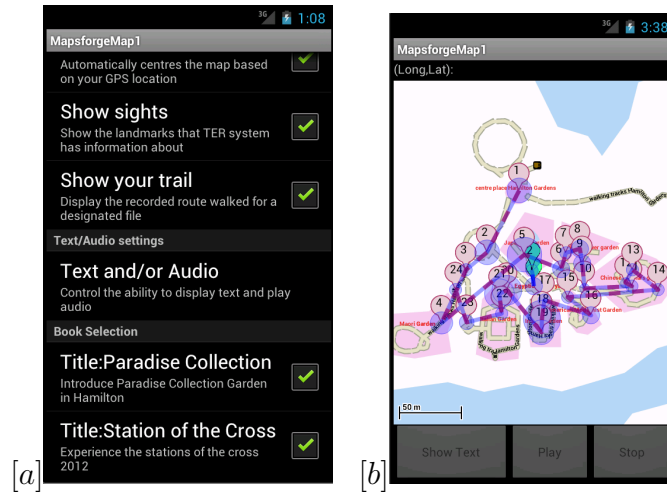


Figure 4.29: Centre Map on GPS Location

function at any time. For example, the user can change their audio setting or book at the very beginning when they start the application, in the middle when they have already read a few chapters, or at the end. All settings and interfaces can be refreshed immediately and efficiently. The map can be zoomed in or out, moved and touched. Android users can rotate the camera setting and it does not affect the TER system. The TER system runs smoothly and remains stable with this interface.

Chapter 5

Implementation

The task of this implementation was to convert the research result and design concepts from Chapter 2 into a usable software product. The TER system used Greenstone Version 3 to build the DL collection and used Java tools to develop software on android devices. Firstly, this chapter will introduce the environment of the TER system followed by a description of the internal structure of TER system. Finally, it will explain different aspects of the implementation strategy.

5.1 Components of TER system

The TER system contains seven components described in Figure 5.1. The DL Server System stores the digital collection details, including book collections and image collections accessed by the TER system. The Location System mainly listens to locations and retrieves information from the server. The Camera System provides the camera function. The Reader System allows the user to read information in their own time. The Audio System plays the audio files. The forge-map library has been imported and includes several classes for the most frequently used types of overlays. To implement a custom overlay in the TER system, the Custom Overlay System builds its own overlay and extends some of the existing overlay classes. These six components interact with the Map Activities Component. The Map Activities Component is a central control system. All information is retrieved and transferred from the

Map Activities. This section will introduce each of the components.

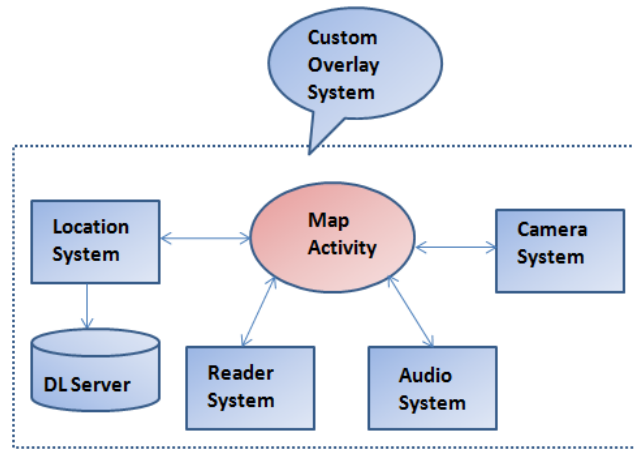


Figure 5.1: TER system components

5.1.1 DL Server

The DL server in the TER contains two collections, introduced in Section 4.1. These collections contain the book content and related images for testing the android application. With an android device equipped with the TER and an Internet connection, it will first load the books and images from the DL server and information stored in the document cache. Then the TER system accesses these data locally. Using the document cache to store data reduces the cost and improves the speed of performance. The procedure for building the DL server will be introduced in more details in Section 5.3.

The DL server was built with Greenstone 3 and located on the server named ORESUND at the lab FG2.01 at the Waikato University. Computers are provided for masters level students to undertake research at the University of Waikato. Due to security issues, all computers located on campus are protected by a firewall. This causes problems in testing the Internet connection on a real device. All Internet connection testing was monitored by the emulator and the debug tools setting on the server.

Greenstone 3 is a complete redesign and reimplementa-tion of Greenstone 2. Greenstone 2 is mainly written in C and Greenstone 3 is written in Java. Greenstone 3 is structured as a network of independent modules that communicate using XML. It runs in a distributed fashion and can be spread across different servers if necessary. It can be downloaded from Greenstone (Greenstone, 2011). The TER system development used SVN to update the Greenstone 3 version by installing Apache Ant and a subversion client together. During the implementation stage the Greenstone server was updated four times via the Subversion Control.

5.1.2 Location System

The TER system starts from the Location System (see Figure 5.2), utilising the GPS function of the android device. The Location System receives GPS information about the travellers locations. The latitude and longitude details are sent to the Location Manager. Based on this location information, the Digital Library server finds the appropriate book in the collection. Through the Web Based XML handler class the XML data can be transferred and stored onto the Location Instance in cache. Now the Location System holds all necessary book details for later usage. Alternatively, the Location Manager will use the Image Loader class to download images in the background. Once all images have been loaded to local devices, the Internet connection is closed. Now all text information and images are available to be used locally to save Internet data transmission expense.

5.1.3 Audio System

The Audio System (Figure 5.3) is triggered by the Location System. The Location Listener Class listens to the GPS system and dynamically updates the location using GPS details. Once the GPS details are shown in the region of the place which matches the GPS in the Location Instances, the Location Manager will retrieve the related information from the Location Instances, then call the Text-To- Speech API which is built into the android to generate the

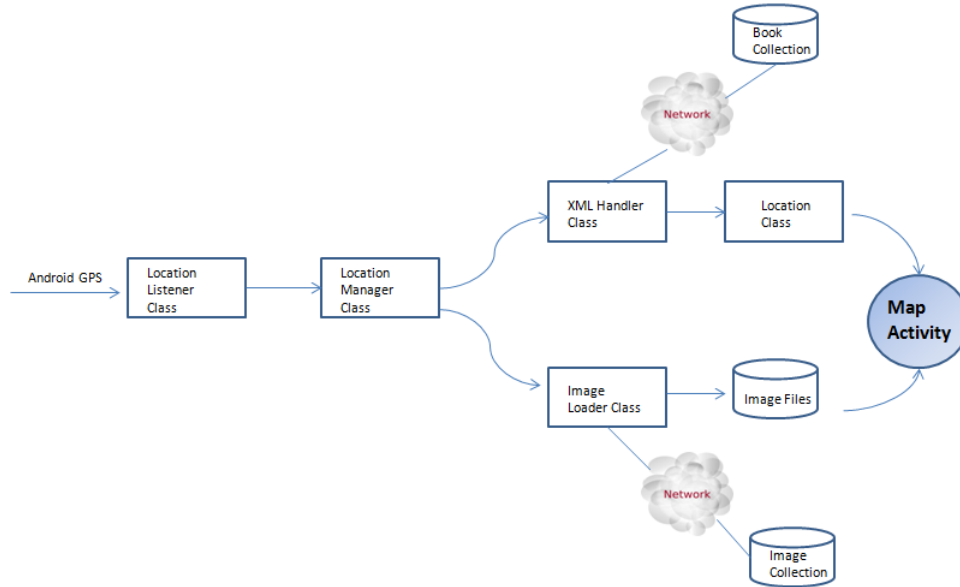


Figure 5.2: Location System Components

Audio mp3 file. It is then stored locally. The files are held ready to play when travellers press the PLAY button via the Audio Player System interface. The Audio Player System has the standard audio player functions for the user. All mp3 or WAV files have been saved for fast access. The Audio System queues all audio files that have not yet been played. The last one will be always play first and the first entry will play last.

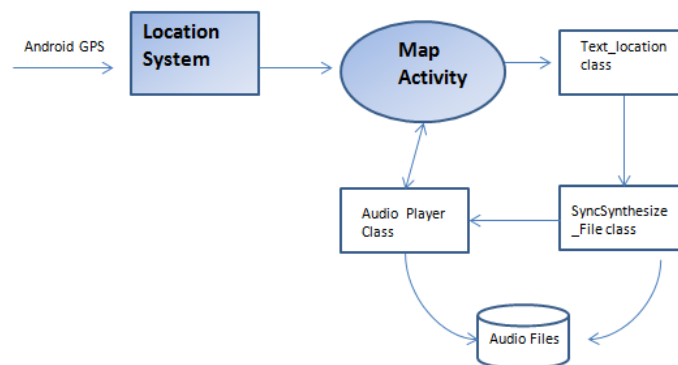


Figure 5.3: Audio System Components

5.1.4 Reader System

The Reader System displays text retrieved from the Location Instance using a customized format. All text will be always shown with its section number and its title on the first line, followed with a break line by the main content. The text content stored in the Location Instance is different from the content to be used for audio. Therefore the TER system has two different text formats (Figure 5.4). The reason for this is that the Text-To-Speech API can only generate the sound for standard letters and words. Other elements of formatting would result in errors. For example, the Text-to-Speech API cannot generate appropriate sound for markup language and dates. Here `< br />` means a break line in markup language but the Text-to-Speech API will translate it as “less than br before slash greater than” or the year 1994 will be pronounced as “one, nine, nine, four”. So the Reader System contains both text content and format mark up language, and the Audio System stores words that are recognized for Text-to-Speech API.

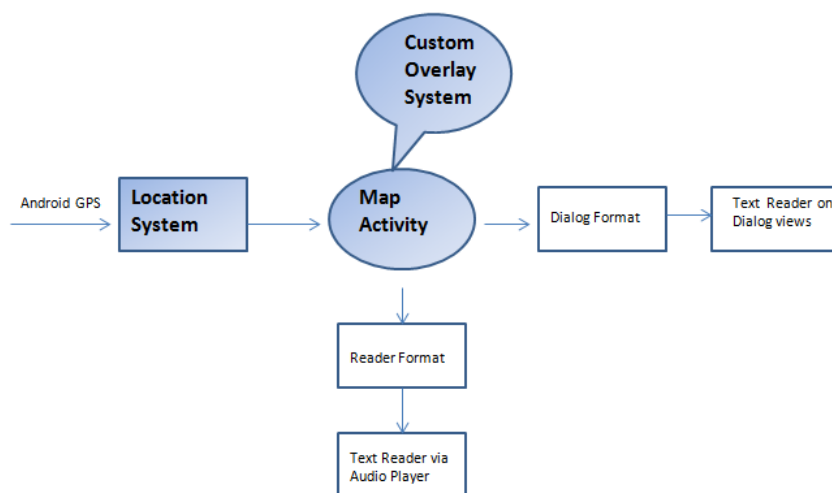


Figure 5.4: Reader System Components

5.1.5 Camera System

Each overlay object derived from the maps forge library class is displayed as a map marker on the screen. Each overlay object contains the necessary details retrieved from the Location Instance. When users click on the button, Take Picture, the Camera System (Figure 5.5) will be called via the listener class.

The location data will also be passed to the Camera System. The Camera System uses android media to capture the image. As images take up a lot of memory in Android system, before saving the image it has to be scaled down based on an appropriate matrix calculation method. When saving an image to a local device, the image is compressed into a PNG file. All images, either preloaded from the server or taken by then user, can be displayed based on their GPS details. Viewing an image class allows the user to view an image while touching the balloon icon on the map.

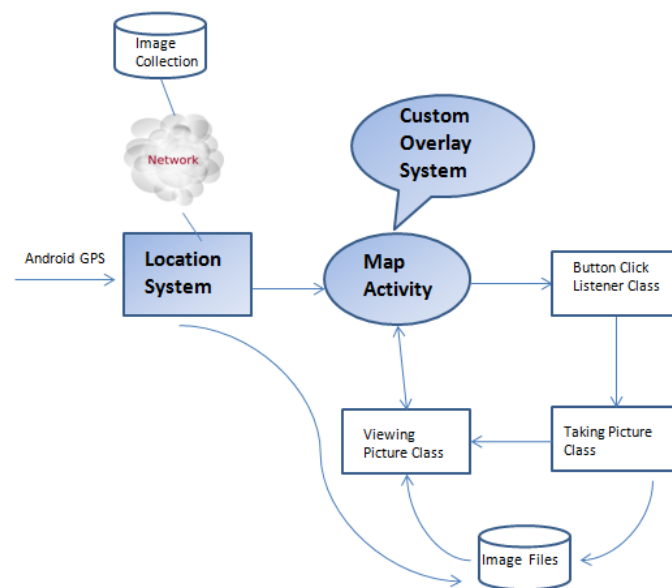


Figure 5.5: Camera System Components

5.1.6 Map Activity

The TER system always presents the map to the end users. All activities are triggered via the map screen, such as a touch the map icon to trigger the Reader System or Camera System. The Map Activity component is central to user interaction. The Audio Player is activated through the Audio Player interface on the bottom of the map screen.

Android activity in the system is managed as an activity stack. When a new activity is started, it is placed on the top of the stack and becomes the running activity. The diagram (see Figure 5.6) shows the life cycle of the activities. The greyed ovals are major states the activity can be in. The square rectangles

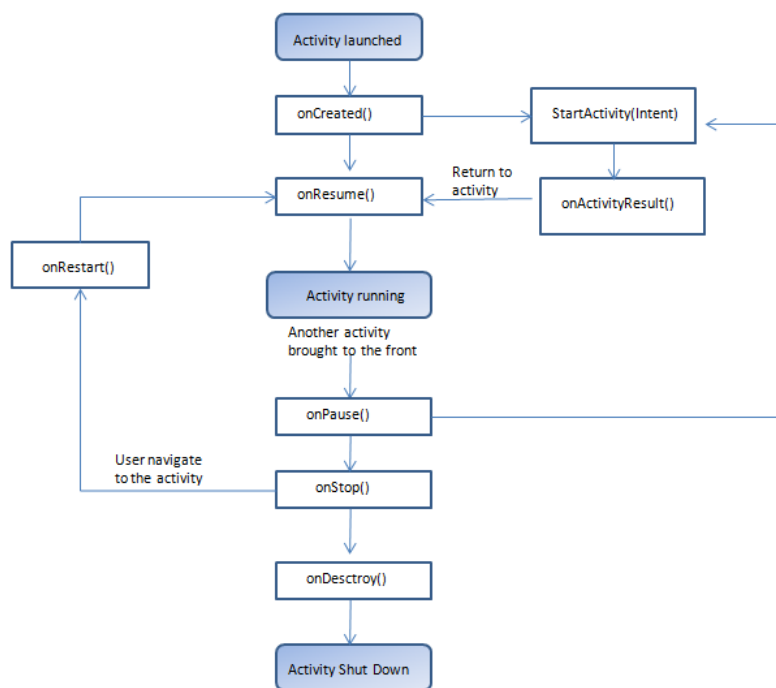


Figure 5.6: Android Activity Life Circle Diagram

represent the callback method when the activity is moved between states. The TER system utilises the activity life cycle to interact with the user. For example, when the map is firstly presented to the user, the running activity is the Map Activity. When users select the preference menu, the preference activity is put on the top of the stack. When the user selects some result from the preference activity, it moves back to the previous activity so the Map Activity is brought to the front of the stack (Android,2012). The TER system uses this loop to allow users to interact with the map. The map is dynamically drawn based on the result of user interaction.

5.1.7 Custom Overlay System

The Custom Overlay System is about how to use the overlay API from the maps forge-map library on the android device. Overlays can be used to display geographical data on top of the map. The maps forge-map library includes several classes for the most frequently used types of overlays. For example, the route overlay is a special overlay to display a sequence of way node. The circle

overlay is to display a series of circles on top of the map. The TER system call methods from the circle overlay and route overlay classes to build a route and circle on top of the map. Except for this, the TER system also builds its own customized overlay.

The class diagram (Figure 5.7) shows the class structure in the Custom Over-

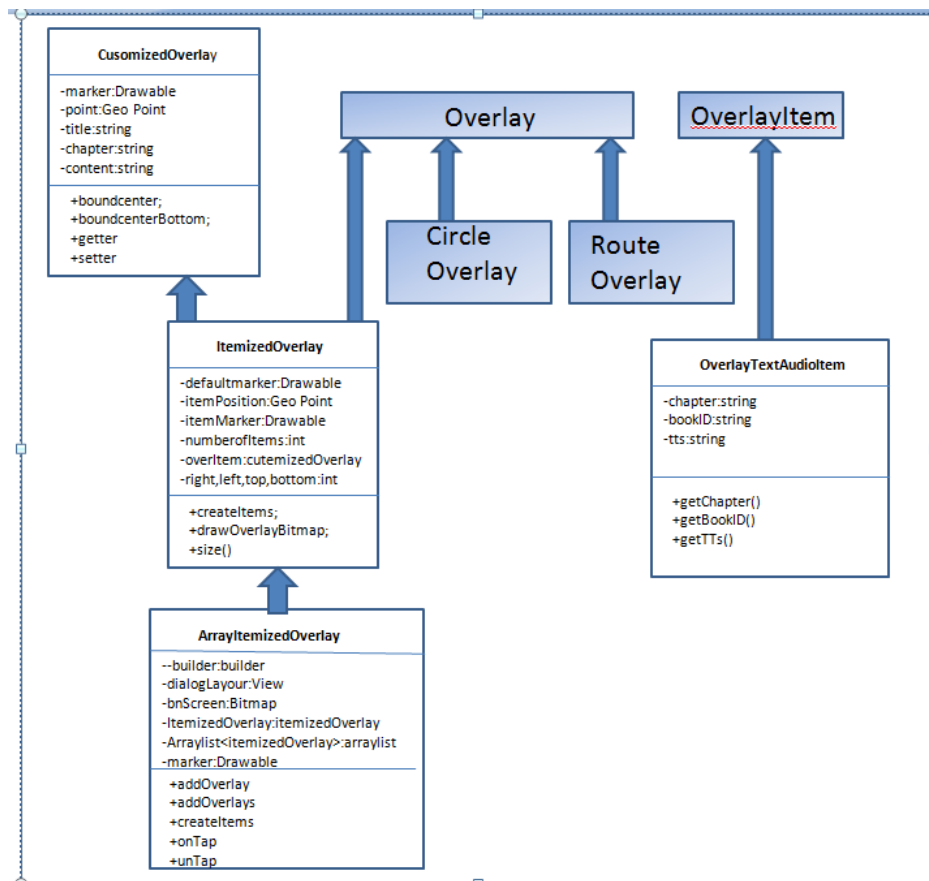


Figure 5.7: Custom Overlay System UML Diagram

lay System. All classes marked by a greyed square are existing classes derived from the maps forge-map library. All UML class diagrams are customized overlay classes used in the TER system. These classes list most attributes and methods and describe its main features. The Custom Overlay System is designed to display each section number, each section title, several markers, and several routes based on the book selection. It also customises the dialog layout when users interact with the map.

5.2 System Environment

The TER system has been set up as a developing environment for both android applications and the Greenstone Digital library server on Windows. Because android application development on Mac OSX and Linus seems more trouble-free than on Windows, the Windows operating system was chosen to be part of the research. This section will introduce the environment setting for the TER system.

Android applications can be set up on Windows OS, Mac OS and Linux. The TER system was based on Windows XP. Basically android applications need Eclipse Classic or Eclipse IDE for Java Developers, Android SDK Starter Package and an Android Development Tools(ADT) Plugin (Android,2012).

Eclipse Classic or Eclipse IDE can be downloaded from the Eclipse website (Eclipse, 2011). If the TER system is going to use an ADT plugin, the Eclipse IDE should be 3.6 or greater. The TER system uses the Eclipse Classic 3.6.2.

The android SDK provides developers the API libraries and developer tools necessary to build, test, and debug applications for android. The Android SDK Starter Package used in the TER system is composed of modular packages that can be downloaded separately using the android SDK Manager. For example, when the SDK Tools are updated or a new version of the android platform is released, we can run SDK Manager to quickly download the updated version to the environment. (Android, 2012).

Android offers Android Development Tools (ADT) for the Eclipse IDE. This plugin is designed to provide a powerful, integrated environment for android applications. It extends the capabilities of Eclipse to let developers quickly build an application UI, debug the application and export the application for distribution. The ADT plugin can be downloaded from the Android Developer website (Android, 2012).

The TER system is a maps forge-map based mobile application in which the maps forge-map 0.2.0 library is imported. The maps forge project (maps-forge,2011b) was initiated in 2008 in Berlin. It provides a free and open source, offline vector map library for Android. It allows developers to easily create new Open Street Map based applications by using its API. The API includes solutions for map rendering, route planning and navigation, map overlays and more.

During the implementation stage, the application was frequently tested in the emulator, especially to avoid Internet connection issues. Two emulators frequently used in the TER system are Android Platform 2.2 version, API level 8 and Android platform 4.0, API 14. Both of them support SD cards GPS and cameras. The emulator that runs on Windows is slightly different from on the Mac OS and Linux (Backx, 2009) in terms of the SD card. For example, if Android users want to load image files, they will insert the SD card into the phone. The emulator running on Mac OS and Linux allows users to create folders directly and insert files into the folder. But the emulator running on Windows does not allow developers to create folders or insert files directly onto the SD card, so the emulator has to be set up with a simulated SD card first. Then developers use the Eclipse DDMS tools to copy and delete files from SD card of the emulator.

5.3 Building DL Collections for TER

Two digital library collections have been built for the TER system to test. This section will focus on the steps to build the collection. Here, the steps to build the Greenstone 3 Digital Library collections is introduced.

- (1) Start the greenstone librarian Interface (Figure 5.8 (a))
- (2) Start a new collection within the librarian interface by clicking on File New. Give the collection name as Hamilton Garden (Figure 5.8 (b))

- (3) Click on the Gather panel, and drag the prepared Hamilton Garden Web HTML file from the left hand side and drop it on the right (Figure 5.8 (c)).
- (4) Click on the Design panel, and click on the Document Plugins, select HTML plugin, add longitude, latitude and Radius to the meta data (Figure 5.8 (d)).
- (5) Under the Search Indexes, click on the New Indexes panel a and add ex.Longitude, Radius and ex.latitude onto the search indexes (Figure 5.8 (e)).
- (6) Click on the Format Panel, and click on the Search button add Location as Display text onto the collection search interface, and change the Browsing Classifiers features, such as ass longitude, radius and latitude to the search index (Figure 5.8 (f)).
- (7) Under Format Panel click on the Format Features add the customized format onto the web page (Figure 5.8 (g)).

All sections within one book are listed on one web page so the HTML file uses the Greenstone 3 specified tag `<sections>` to hold each section. The tag `<descriptions>` holds all metadata necessary for the particular collection, such as bookID, longitude, latitude, radius, chapter number and chapter name. Inside the HTML file we insert line breaks when it needed. For example the book collection for poems contains a lot of break lines according to its space requirements.

In case the TER system is run without an Internet connection, these connection details will be transferred to the XML file for processing. Inside the XML file this metadata includes longitude, latitude, radius, book ID and section number. Each section name is related to the corresponding section content.

The image collection (Figure 5.1) sends 13 images available to be downloaded to local devices. Each image is associated with the places mentioned in the

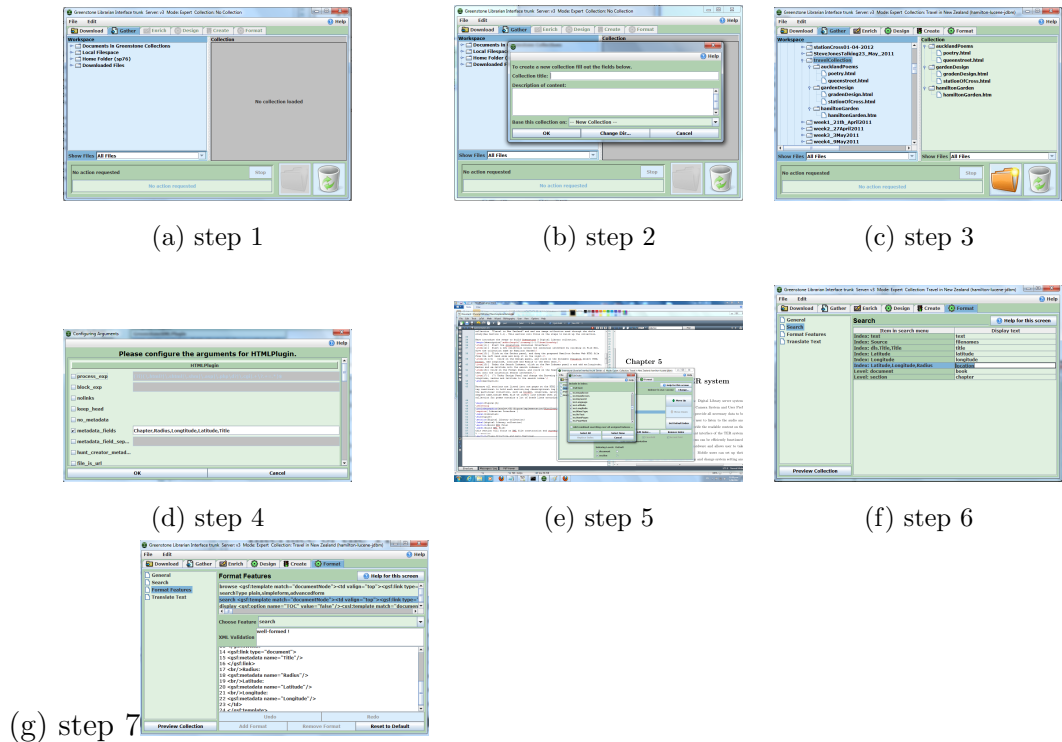


Figure 5.8: Building the collection

Travel in New Zealand collection.

5.4 OpenStreetMap for TER

The TER system uses maps forge binary map files based on OpenStreetMap. OpenStreetMap is free to use for re-distributing data for commercial and non-commercial purposes (openstreetmap,2012). The maps forge library map file format is designed for map rendering on mobile devices. However, the OpenStreetMap data has to be converted into a binary map file, which is needed to display maps with maps forge-based applications. This section introduces OpenStreetMap first, followed by the procedure of creating a new map.

5.4.1 About OpenStreetMap

OpenStreetMap can be downloaded free from the OpenStreetMap website. OpenStreetMap (OSM) is a collaborative project to create a free editable map of the world. It was founded in July 2004. In November 2011, OpenStreetMap cleared 500,000 registered users (openstreetmap,2012).

A wide variety of programs use OpenStreetMap data to display maps on mobile devices. Nearly all phones that can use a Java platform can make the OpenStreetMap available, such as android, iPhone, Blackberry. The TER system uses the offline maps in an android location-based service (openstreetmap,2012).

OpenStreetMap uses a topological data structure with some major elements. The TER system uses nodes and relation elements contained in OSM maps. Nodes are points with a geographic position. Positions are stored as a pair of latitude and longitude coordinates. Relations are groups of nodes, ways and other relations that can be assigned certain properties. The TER system can read GPS data from a map and shows relationships on the map (Wikipedia, 2012b).

Figure 5.9 [a] shows part of Hamilton Binary Map file read by a maps forge-based application. Figure 5.9 [b] is the zoomed-in part for Hamilton Garden Place. Figure 5.10 shows an Auckland map and Auckland City Central Map read by a maps forge-based application.

5.4.2 Generating the Binary Map File

The TER system is a maps forge-based application, which has imported the maps forge- map library. The maps forge binary map file format is designed for map rendering on mobile devices. It allows for efficient storage of OpenStreetMap data, fast tile-based access and the filtering of map objects by zoom level.

The map file consists of several sub-files, each storing the map objects for a

different zoom interval. Zoom intervals are non-overlapping groups of consecutive zoom levels. Each zoom interval is represented by a single member of the group, the so-called base zoom level (mapsforge, 2011a).

This section introduces the procedure for generating a maps forge binary map file from OSM. It will require a map-writer plugin and osmosis software. The maps forge map-writer plugin allows the conversion of OpenStreetMap data into the maps forge binary map file. To do this, firstly install the latest version of osmosis software from the website (Osmosis, 2012). Secondly install the map-writer plugin (mapsforge,2011b). Finally run the osmosis command to convert the OpenStreetMap to the binary map file.

The maps forge library has been updated regularly. Each version of the maps forge-map library may have different requirements for rendering the binary map file. For example the maps forge-map library 0.2.4 is used to render a version 2.0 binary map file, and maps forge-map library 0.3.0 is use to render the 3.0 binary map file (MapFileWriterOsmosis, 2011).



Figure 5.9: Hamilton Binary Map File

5.5 Interface Implementation

The TER system is made up of a series of pages called views. These views contain important content, such as a text view containing the current loca-

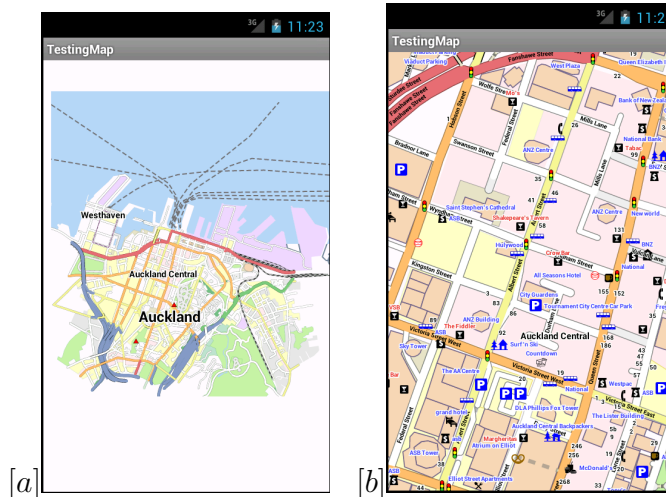


Figure 5.10: Auckland Binary Map File

tion GPS details, a map view showing the map and other important landmark details and button views providing the Audio player interface to users. Users change their preference settings via a list view with a number of check boxes and radio buttons. This view design has been achieved in the TER system by using android layout design tools or background code. The android layout design tools provide XML to build the interface. All views can be written in XML and used in the activity. Also the TER system dynamically sets or modifies the layout design during the running time.

5.5.1 Using Layout Design Tools

The android development environment provides two basic design tools for the developer to design their layout easily and nicely: Widget views and Android Development Toolkit (ADT) for Eclipse.

The majority of the widget views in the android application can be constructed with these widget views. This consists of simple widget-like buttons, text views, image views and menus. The TER system combines simple widgets into more complex components, such as a list of radio buttons, check boxes and dialogue views that include image, text and buttons.

The ADT plugin for Eclipse significantly speeds up Android layout design. By using the ADT tools the layout can be designed by using XML files or a graphic user interface (GUI). Android developers can visualize beforehand what a layout will look like. The android operating system will inflate the XML file into the visual layout that the end user actually sees.

5.5.2 Using drawable resource

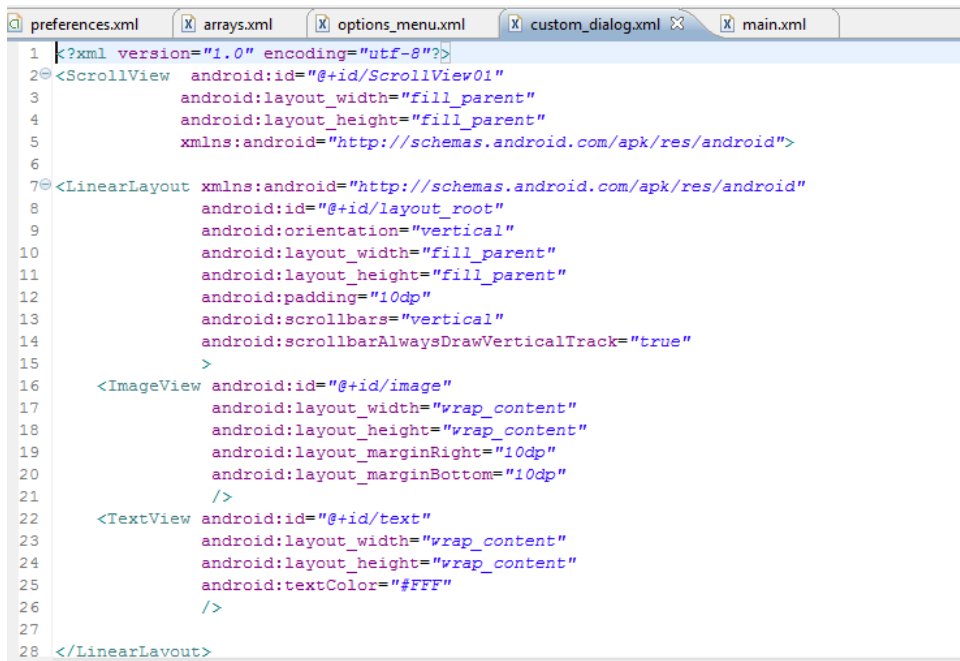
Based on the mobile design issues discussed in Section 3.5, the TER system uses customised icons for the TER system. These icons are imported into the Android application resources. When the user comes to the place, the marker will be replaced by a different icon imported from its drawable resources. The TER can also show different book information by using these map markers on the map.

5.5.3 Using XML resources

The TER system uses the five XML files to build its main view page. Figure 5.11 shows one XML file used in the TER system. This XML file will be called when users press on the map marker and the dialogue views that include image, text and buttons will pop up.

5.5.4 Setting Layout in Running Time

Android developers also can dynamically set or modify their layout design during running time. Throughout the programming, these initial settings can be updated based on user interaction. For example, in Section 4.3 all audio player interfaces will be dynamically changed and it will be done via code. Also, on the map all overlay items are redrawn during running time described in Section 4.6. All map markers and travel paths are redrawn in running time

A screenshot of an IDE window showing several XML files: preferences.xml, arrays.xml, options_menu.xml, custom_dialog.xml, and main.xml. The main.xml file is open and displays the following XML code:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <ScrollView android:id="@+id/ScrollView01"
3     android:layout_width="fill_parent"
4     android:layout_height="fill_parent"
5     xmlns:android="http://schemas.android.com/apk/res/android">
6
7 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
8     android:id="@+id/layout_root"
9     android:orientation="vertical"
10    android:layout_width="fill_parent"
11    android:layout_height="fill_parent"
12    android:padding="10dp"
13    android:scrollbars="vertical"
14    android:scrollbarAlwaysDrawVerticalTrack="true"
15    >
16    <ImageView android:id="@+id/image"
17        android:layout_width="wrap_content"
18        android:layout_height="wrap_content"
19        android:layout_marginRight="10dp"
20        android:layout_marginBottom="10dp"
21    />
22    <TextView android:id="@+id/text"
23        android:layout_width="wrap_content"
24        android:layout_height="wrap_content"
25        android:textColor="#FFF"
26    />
27
28 </LinearLayout>
```

Figure 5.11: Widget views in XML

described in Section 4.7. All of these are done in the Java source code.

5.6 Persistent Data in TER system

The TER system needs to transfer data across various activities. For example, when users select a book, this book information has to be sent to the Greenstone server to retrieve the related book data to the local devices. Then these related data have to be sent to the maps forge library to draw its corresponding map marker. Also when the user changes their idea to select a different book, the book information has to be updated and the map has to be redrawn dynamically. No matter where the data has been changed, they need to be persistent for the application. TER system stores persistent application data in several ways. The solution will depend on the specific needs.

5.6.1 Shared Preferences

The shared preferences class provides a general framework that allows applications to save and retrieve consistent key-values of primitive data types. The TER system creates user preferences and provides an activity framework for the application to create user preferences, which will be automatically consistent by using the shared preferences class.

The preference activity allows users to interact with the application. In the TER system, whenever users change their book selection, the audio player interface setting and show user travel path selection data, will be dynamically written to the system, based on user interaction. During running time, the android activity will keep track of this data modification and always store their new values in the system. Then the TER system generates the map and map interface based on these settings. These data will be permanently stored in the system until the user modifies it again.

Android applications can also pass preference data between different activities. All the preference data is saved onto the XML resource and retrieved across the application. For example, the TER system has to pass the overlay item details to the picture taking activity. So using the android application bundle can pass multiple data to other activities. Through the bundle, all data can be retrieved at the destination class using this method.

5.6.2 XML resource

The TER system also uses the XML resource file to store data. Some global data are stored here for easy maintenance. For example, all preferences setting data including lists of book names, lists of radio buttons for audio player interface setting and show travel path checked boxes are stored here. This information stored in XML resources files, then called in the application.

5.6.3 External Storage

Android-compatible devices support a shared external storage to use and save files, such as an SD card. Files saved to the external storage are world-readable and can be modified by the user when they enable USB mass storage to transfer files on a computer. During the case study (Section 6 and Section 7), the TER system was adapted into a standalone application, which runs without an Internet connection. The data initially stored on the server were transferred onto the SD card as XML files.

Before working with external storage, the application has to get permission to access the SD card. This method is called the `getExternalStorageState()` to check whether the media is available (Android,2012).

The TER system also creates and writes image files to external storage. After images are preloaded from the DL server, these images are written into bitmaps onto the android SD card.

5.6.4 Internal Storage

The TER system saves files directly on the device's internal storage. By default, files saved to the internal storage are private to the application and other applications cannot access them. When the user uninstalls the application, these files are removed.

The TER system provides access to its raw asset. The Asset Manager class presents a lower-level API that allows application to open and read raw files that have been bundled with the application as a simple stream of bytes. All map files and audio reminder mp3 files in the TER system are pre-stored in the assets folder. The assets folder will be distributed along with the android APK file, which contains all the raw files needed for the application.

The TER system uses the Asset Manager class and `OpenFileOutput` to read

data stored in the assets folder. The system will check the internal storage directory first, then open an asset using Access Streaming mode, and write the data onto the external storage for fast access later.

5.6.5 Server Storage

The TER System uses the network connection when needed to store and retrieve data on its web-based services. The `java.net`, `Apache.http` and `Android.net` package can all help with network access. The TER system has tested three of them; Android version 2.0, Android version 3.0 and Android 4.0 (Emulator).

Package `org.apache.http.client` is the API for client-side HTTP communication and entry point to the `HTTPClient` module (Android,2012). `DefaultHTTPClient` is the default implementation of an HTTP client. `Android.net` that helps with network access beyond the normal `java.net`.

`AndroidHTTPClient` is the implementation of the Apache `DefaultHTTPClient` that is configured with reasonable default settings and registered schemes for android. The `java.net.URLConnection` is the super class of all classes that represent a communications link between the application and a URL.

During research and testing, the `DefaultHTTPClient` class in the Apache package was found not to support the Android 4.0 version. The `AndroidHTTPClient` class in `Android.net` package could support both Android 3.0 and higher. Test results shows the `URLConnection` in `java.net` package supported various android platforms including version 2.0, 3.0 and 4.0. So the TER system used the `java.net` package to connect to the server and retrieve data.

5.7 Algorithm and Data structure

Data structures provide a means to manage huge amounts of data efficiently. Usually efficient data structures are a key to designing efficient algorithms. Both data structures and algorithms are key organising factors in software design. This section will focus on the explanation of data structure and algorithms involved in the TER system.

5.7.1 TER System Data Structure

To overview the data structure, there are some frequently used data structures in the TER system. Like an array data structure stores a number of elements of the same type in a specific order, they are accessed using an integer to specify which element is required. For example, this uses an array to hold GEO points of all landmarks shown on the map, then draws a travel path between these landmarks. An enum type is a type whose fields consist of a fixed set of constants. One example is to store audio player interface options. There are four options here, like TEXT ONLY, TEXT PLUS AUDIO, AUDIO ONLY, AUDIO PLUS TEXT. These values are a fixed set of constants, so it is better to use an enum type data structure.

An object contains a number of data fields, and also a number of program code fragments for accessing or modifying them. In the TER system, the details of each chapter loaded from the DL server such as content, latitude, longitude and chapter numbers, are stored in an object (Name it Location) implemented in a customized class.

The TER system also uses Java collections to handle the collection of objects. A collection is similar to an array except that they have advanced features and can change their size dynamically. There are so many types of collections. They allow developers to complete work more quickly and turn in more readable/maintainable code. Based on the application's need, the TER system has

already used the following collections framework.

- HashMap
- HashSet
- ArrayList
- LinkedList

A HashSet is a collection that cannot contain duplicate elements. Not only must these elements be unique, but while they are in the set, each element must not be modified. It creates a collection that uses a hash table for storage. It does not guarantee that the order will remain constant over time. The TER system uses the HashSet to store previous visited locations and current locations. It is possible there is more than one place needed to be highlighted on the map. The HashSet can store these place details and remove them based on need. These places are not allowed to be duplicated and their order is not important (Oracle,2011).

In Figure 5.12 [a] when a user is walking close to the place marked as A, B, and C, these three nearby places will be highlighted on the map. In this case the HashSet will have three objects in it. The greyed circle B means the user has already visited and its details are stored in another HashSet, which is used to record places that have been visited. The TER system will keep track of these visited places and change the content based on this kind of value. Of course this kind of data does not allow duplicated data. Once the user continues to walk to the next place, these dataset values will change in the system. Figure [2] the hash- set for the current location will be null. The HashSet for the previous visited places will still be the same as Figure [a].

The HashMap class can add keys and value put (key, value) pair elements. It permits null key and value. The key and value of the HashMap can be produced by the Set Interface and Map interface through the Iterator interface. When web-based XML data are loaded from the DL server, all metadata tag

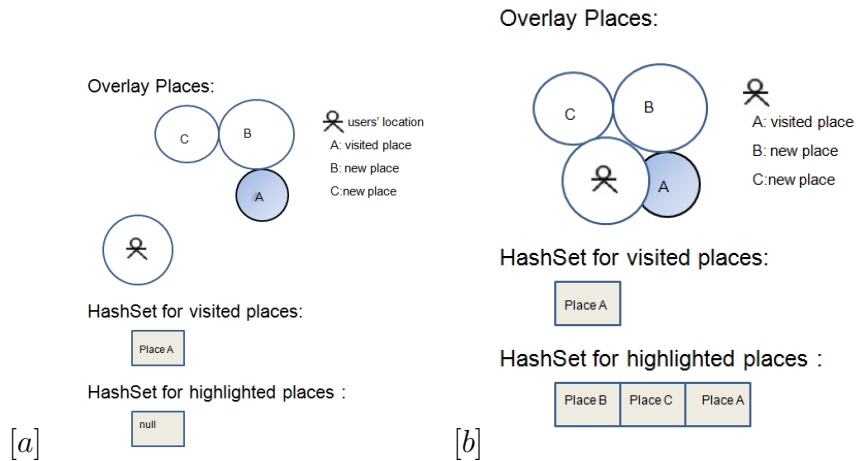


Figure 5.12: Location sample stored in HashSet

and their content elements can be loaded. These pair elements can be easily saved and retrieved by using the HashMap.

In Figure 5.13 the name “Chapter” in the metadata will be a key and “2”

Metadata:

```

- <metadataList>
  <metadata name="Chapter">2</metadata>
  <metadata name="Radius">0.01</metadata>
  <metadata name="Latitude">-37.806534</metadata>
  <metadata name="Longitude">175.30264764</metadata>
  <metadata name="Title">2 Beginning of paradise garden</metadata>
</metadataList>
    
```

HashMap:

Key:Chapter Value: 2	Key:Radius Value:0.01	Key:Latitude Value: -37.806534	Key:Longitude Value: 175.30264764	Key:Title Value: Beging of paradise garden
-------------------------	--------------------------	-----------------------------------	--------------------------------------	---

Figure 5.13: Meta Data sample stored in HashMap

in the content will be a value in the HashMap structure, and these data will be retrieved using the key to access its values. Also the TER system uses the HashMap to store pair elements for highlighted overlay items on demand.

A list is an ordered Collection (sometimes called a sequence). The TER system uses two types of list interface: LinkedList and ArrayList. LinkedList allows for constant-time insertions or removals, but only sequential access of

elements. In other words, you can walk the list forwards or backwards, but grabbing an element in the middle takes time proportional to the size of the list. An ArrayList, on the other hand, allows random access, so you can grab any element in constant time. However, ArrayLists take up as much memory as is allocated for the capacity, regardless of whether elements have actually been added (Oracle,2011).

The Audio Player stores all audio files in a queue by using LinkedList. Due to its small size and fast access through the first one in the queue, the first one always plays first. In Figure 5.14 the Text-to-Speech API generates the audio files. These audio files are stored in the linkedlist. The Audio Player Processor retrieves the first audio file in the linkedlist. The linkedlist used here only stores the audio files that are ready to play. Once complete, the audio files will be removed from the first position in the linkedlist. The linkedlist makes it fast retrieve and delete.

The ArrayList is the most frequently used collection in the system. A large amount of data need to be stored in the ArrayList because they are a non-fixed size, random access or index search. For example, the ArrayList of Location to hold all location objects, the ArrayList of overlay items to hold overlay details to be drawn on the map, the ArrayList of travel route data to hold the travel route node to be drawn lines between them and the ArrayList of HashMap to store all HashMap loaded from web based XML to be transferred into the location object.

5.7.2 TER System Algorithms

Based on the data structure, the TER system determines its algorithms. One example is to calculate the distance between two geographical locations for identifying when to remind users whether the new story is available. Also it is possible that there is some overlap between stories in nearby locations.

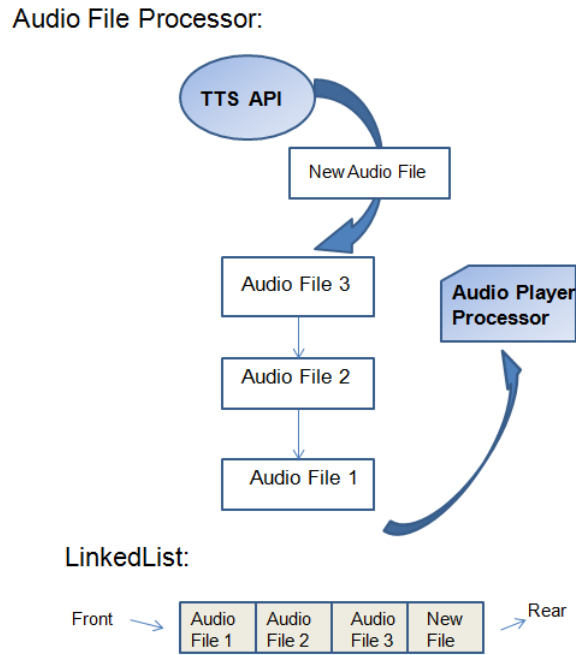


Figure 5.14: Audio Files sample stored in LinkedList

The TER system imports the mapsforge projection package to translate between the pixel coordinate system on the screen and geographical points on the earth. Then it uses the appropriate algorithm to manage the reminder system. In the Audio Player System there is a reminder subsystem which reminds people of the story available to listen to or read when users come to a certain area. The Reminder System will also tell the Map System to redraw or remove the highlighted map marker from the screen. First we need the right method to calculate the distance between two points on the map, then we calculate the working range of the reminder system. If the point is out of range, the reminder system stays silent. Only when the point is inside the range, can the reminder system wake up and function.

Figure 5.15 describes the algorithm for distance calculation. There are two points, A with longitude and latitude (a_1, b_1) and point B with a_2 and b_2 . The distance between two points is based on the Pythagorean theorem. When we know two sides of the right triangle, we will use the Pythagorean theorem. So the distance between A and B is the square root sum of $(a_2 - a_1)$ and $(b_2 - b_1)$.

To control distance flexibility, we import the controller which sets up a fixed value to control the distance between A and B.

Based on the Pythagorean theorem the boundary distance is radius of A plus radius of B plus controller. Then draw a circle with the radius of this boundary. Any point that comes inside this circle the reminder system will wake up and work, otherwise it will keep silent. In 5.16, B1 is outside of the circle so the reminder system will keep silent. B2 is inside the circle so the reminder will sound “Chirp Chirp” to remind traveller and send information to the mapsforge library to redraw highlighted marker on the screen. Once the point B2 moves out of the circle area, the reminder system will communicate with the mapsforge to remove the highlighted marker on the screen.

Another algorithm used in the TER system is used to keep track of the au-

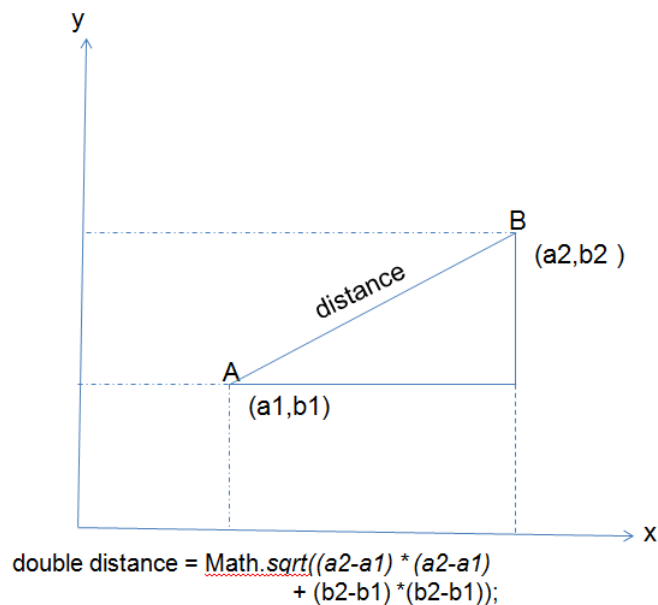


Figure 5.15: Pythagorean Theorem in TER system

dio file system. As introduced in this section, all audio files are stored in a linkedlist. The Audio Player system always retrieves or removes the first file from the linkedlist. Each place originally contains three audio files: one file for beginning, like the title, one file for its contents and one file is for the end. Three files are stored in linkedlist. So every time the title file will be played

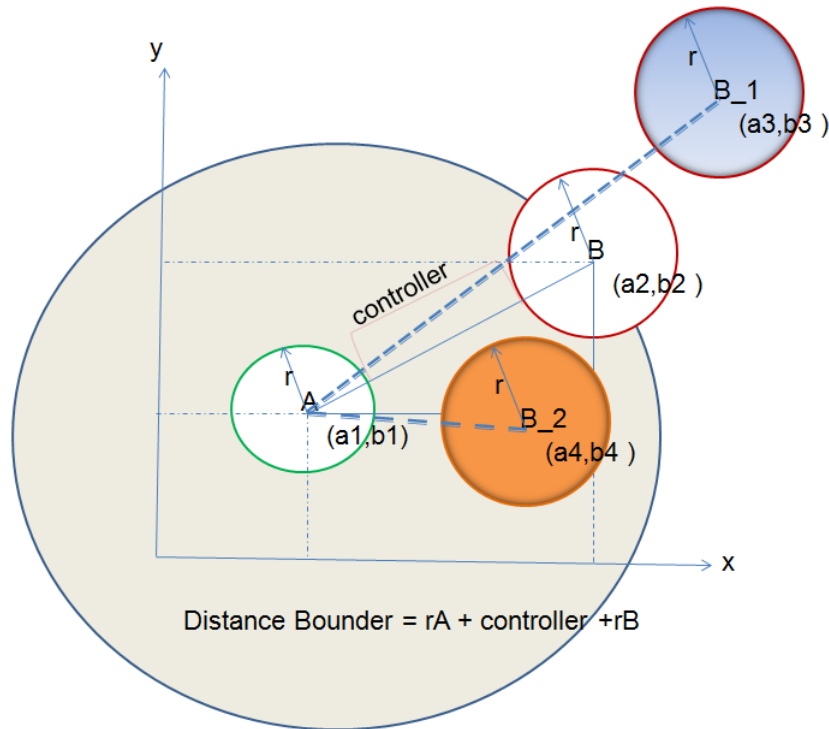


Figure 5.16: Algorithm in Audio Player Reminder subsystem

first. Then the system uses the modular method to keep track of the playing procedure. This algorithm and its data structure make the programme run more efficiently and fast.

5.8 TER System Micro Optimizations

An android user will encounter situations which can be a challenge to the application. Some examples of situations are described below.

- An Android application will run on a mobile device with limited computing power and storage, and constrained battery life. Battery life is important to users, and the android application might drain users battery.
- Android users can have lots of opportunities to change the mobile device configuration such as rotation, phone call state and language setting. All those changes might destroy the application.

- In an android application it is easy to accidentally “leak” memory and resources. Even though the application looks fine in the device, the Debug tools show there is still an error message related to memory issues.

This section introduces some strategies used in the TER system to optimise its performance. These micro optimisations focus on writing efficient code for running on multiple platforms.

5.8.1 Reducing Number of Objects

Object creation is never free. A general rule is reusing the objects can make allocation cheaper, but allocating memory is always more expensive than not allocating memory. Fewer objects created means less-frequent garbage collection. During the test stage we found if we created more objects the application always ran out of memory. Then the TER system reduced the object memory allocation to get rid of the error message.

Some examples of things that can help to reduce the memory allocation in TER system are as follows:

- Create reusable objects to reduce the number of objects.
- Use Clear() method to clear data and reuse the objects.
- Force the Garbage to collect the object and release the memory.
- Reduce the number of temporary objects.
- Using single one-dimension array to hold data.
- Within Stop() method always clear all data in all used objects.
- Always close the input/output stream when finished reading/writing.
- Always close the internet collection when not needed.

5.8.2 Optimizing Loop Syntax

The TER system uses the optimization for loop syntax. The TER system frequently uses the collection to store the item and always loop through it. So it is quite important to write a more efficient loop structure for higher performance. For example, the TER system uses an array list of HashMap to hold all key-value paired data transferred from the web-based XML file. When looping via the array list, each item will be saved as one location object. The length of the arraylist will be reused for overlay items on the map. Figure 5.17 shows a different loop syntax which functions the same way and returns the same result but with different performance. [b] and [c] have better performance than [a]. [c] is the fastest loop. Some tips used in TER system are as follows.

```
[a]
ArrayList<location> locations= new ArrayList<Location>();
...
for (int i = 0; i < location_hashmap_list.size(); i++) {
    HashMap<String, String> location_hashmap =
    location_hashmap_list.get(i);
    Location location = new Location(location_hashmap);
    locations.add(location);
}
location_hashmap_list.clear();
return locations;
```

```
[b]
ArrayList<location> locations= new ArrayList<Location>();int
length=location_hashmap_list.size();
...
for (int i = 0; i < length; i++) {
    HashMap<String, String> location_hashmap =
location_hashmap_list.get(i);
    location = new Location(location_hashmap);
    locations.add(location);
}
location_hashmap_list.clear();
return locations;
```

```
[c]
ArrayList<location> locations= new ArrayList<Location>();
int length=location_hashmap_list.size();
...
for (HashMap<String, String>
location_hashmap:location_hashmap_list){
    Location location = new Location(location_hashmap);
    locations.add(location);
}
location_hashmap_list.clear();
return locations;
```

Figure 5.17: Loop Optimization example

- With an ArrayList, use a hand-written counted loop. This will reduce the cost of getting the array length once for every iteration through the loop.
- Make the Arraylist reusable and make data global in one class to reduce

the amount of object creation.

- Using enhancers for each loop. The enhanced loop allows developers to iterate through a collection without having to create an iterator or without having to calculate beginning and end conditions for a counter variable.

5.8.3 Handling Runtime Changes

During runtime the android configurations can be changed, such as screen orientation, keyboard availability and phone call state and language setting.

When these changes occur, the android application restarts running activity. The restart activity will call on Destroy() followed by onCreate(). During the restart stage significant amounts of data can be costly and the application might be destroyed. The TER system declares that the activity handles the configuration change itself, which prevents the system from restarting the activity.

The TER system declares the activity handles a configuration change such as rotation or keyboard availability in its manifest file. Then when these configurations change, the TER system will not restart the activity. Instead, the activity will receive a call to make ConfigurationChanges() to handle these configurations. All activity resource objects are updated and reset on the screen based on the new configuration.

5.8.4 Loading XML Data

XML is widely used in the developer community as a tool for passing, manipulating, storing and organising information. Greenstone server data can be accessed as web-based XML data. Figure 5.18 shows the XML data for Chapter 1 in the book “Paradise Garden” in the “Travel in New Zealand” collection. All necessary information is stored in the XML elements. Then the

parser will access these XML elements. The parser can be thought of as an interface to an XML document, exposing its contents through a well-specified API. Ideally a good parser should be fast, space efficient, rich in functionality and be easy to use.

In reality, none of the main parsers have all these features at the same time.

```
-<documentNode docType="hierarchy" nodeID="HASH2fe900889d0aaf08b59fca.1" nodeType="leaf">
- <metadataList>
  <metadata name="Chapter">1</metadata>
  <metadata name="Radius">0.01</metadata>
  <metadata name="Latitude">-37.80608675</metadata>
  <metadata name="Longitude">175.3029395</metadata>
  <metadata name="Title">1 Beginning of gardens</metadata>
</metadataList>
- <nodeContent>
  Welcome to the Hamilton Gardens. <br /> Hamilton Gardens is not a traditional botanic garden. We like to say, that while botanic gardens are
  designs and collected them together in one place. We have organized those gardens into five collections: Paradise Garden collection, The Prod
  collection.<br /> Hamilton Gardens itself started in 1964 and was developed in kind of a hodgepodge sort of way until the 1980s when a new
</nodeContent>
```

Figure 5.18: Web-based XML Data in DL Example

DOM parser and SAX parser are two major parser APIs. DOM stands for Document Object Model and is rich in functionality. SAX stands for Simple API for XML parsing and is much more space efficient. The reason for this is that the two parsers work differently. The DOM specification defines a tree-based approach to navigating an XML document. It processes XML data and creates an object-oriented hierarchical representation of the document. The SAX specification defines an event-based approach whereby parser scan through XML data, calling handler functions to find the text nodes, text attributes, and text content.

The TER system runs on mobile devices and XML file located on the server, so speed and space are the two most important issues. Also, the TER system does not need to load the entire XML structure. The SAX parser was chosen to read the web-based XML file in the TER system. During running time the SAX parser only loads a small part of XML file in its memory, so it does not require much memory. It speeds up the system and saves a lot of memory space.

Once the SAX parser loads the XML file, the information is saved into a document cache. The Internet connection is closed. The design of the document

cache is to save cost and bandwidth. After loading, the TER system only accesses its data storage locally.

5.8.5 Avoiding Memory Leaks

The mobile device has limited memory size. The android device can also use a lot of applications in the memory space available. The application can run out of memory easily. That is because of term context issues for a lot of operations within android development. Therefore memory space is one of the most important issues for mobile applications. The following list shows some examples of memory issues encountered during the implementation stage.

- Running on the Emulator, setting the Heap size as 16 by default, the application always takes much more heap size than its default setting.
- Using Eclipse, if using the default setting such as Xms 256m and Xmx 1024m, it will always show the “out of memory ” message.
- When taking a photo, each photo takes up a lot of memory space. When taking a second photo it always runs out of the memory.
- When the user selects books more than four times, the application crashes due to large memory space taken.
- When the user exits the application and comes back again, the android dialogue box pops up to force the user to close the application due to memory still being occupied by the application.
- When the user listens to the first few chapters loaded from the server, it might crash when the next chapter is selected and show a memory error message.
- When the user switches the “Travel Path” on and off a few times, the TER system takes a lot of spaces to redraw the map and path several times, then it is crashes easily due to running out of the memory.

There are many different reasons which cause memory leakage. The way to figure out what causes the leaks, is to go into the adb shell and keep track of the memory allocation and figure it out from there. In short, the following things can be done to avoid running out of memory.

- All activities should have the same life cycle.
- keep long process activity run in the background.
- One can also use context activity instead of the context application.
- It is better to use the static inner class that has a weak reference to outer class.
- The garbage collector does not release the memory spaces properly.
- When the activity is destroyed and recreated, make sure the object can be dismissed and the memory allocation can be released.
- Always scale down and compress the image before saving it. Images take significant memory space.

5.9 Multiple Thread in TER system

When an Android Application is launched, the system creates a thread of execution for the application, called “main” or “UI thread”. This thread is in charge of dispatching events to the appropriate user interface widgets and is the thread in which the application interacts with components from the Android UI tool kit. All components then run in the same process and are instantiated in the UI thread. An android application has a single thread running model, which means if the UI thread is blocked for more than five seconds, the android will force that with an “Application Not Responding” dialogue. However, it is very important to deal with multiple threads in android (Lars,2012). There are mainly two ways used in the TER system. These are described below: one is using synchronized statements to allow concurrent processing, the other is running threads in background.

5.9.1 Background Process in TER System

If the programmer does not use any concurrency constructs, all code of an android application runs in its main thread. If the application performs a long lasting operation, such as loading images from the Internet, the UI thread will block it until the corresponding code has finished.

To provide a good user experience, all potentially slow running operations in an android application should run asynchronously. There are several ways to run threads asynchronously, such as Handler and AsyncTask used in the TER system.

5.9.1.1 Using Handlers

The Handler class can update the user interface. A Handler provides methods for receiving instances of the Message or Runnable class. Inside the TER System it uses a SAX parser content handler to receive instances of the Message class.

As introduced in section 6.8.4, the SAX parser was chosen to pass the XML file onto the system. Using the SAX parser, we must define an event-driven API that will respond to each element as it is received. Inside the SAX parser class, the content handler is the main interface that handles each of the incoming events from the XML. The order of events in this interface is very important. For example, all of an element's content will appear in order, between the startElement event and the corresponding endElement event. When each XML element is coming, the different handler will be called to read all XML until it reaches the end.

When an opening tag such as <longitude > is reached, the startElement handler will be called. When a closing tag </longitude > is reached, the endElement handler receives notification of the end of an element. When the SAX parser reaches the closing tag, it calls character() method to get the content

that is between the opening tag and closing tags. That is the way the content handler works in reading XML files, and all these are running asynchronously.

5.9.1.2 Using AsyncTask Class

AsyncTask class allows the programmer to perform asynchronous work on the user interface. It has its own handlers, which allows it to do something behind the UI thread (Android,2012). It can publish the results as work progresses. The AsyncTask class encapsulates the creation of Threads and Handlers.

When the user starts running the TER system, the map displays on the mobile screen. That is the UI thread running. In the background, the system is loading images from the DL server using Asynctask class. An AsyncTask is started via the execute method and this method use is called the doInBackground method. The doIn- Background method contains the coding instruction. This connects to the DL server, and loads images. Once the image is loaded, the AsyncnTask frame called onPostExecute(), which contain code to scale down image, writes a bitmap out onto the local device. Now the UI thread is not blocked no matter the size and time of loading processing.

5.9.2 Concurrent processes in TER system

During the operation of the TER system, many threads do need to run concurrently. For example, when the TER system is drawing more than one overlap item on the screen, there is more likely more than one landmark to be drawn on the screen while UI thread is running. Another obvious example is when users are playing the mp3 files while the UI thread is running. The TER system frequently uses the synchronized statement where the application utilises more than one thread.

Synchronized statements ensure the method will be accessed one thread at a time to prevent deadlock occurring. A synchronized statement contains a syn-

chronized block, within which objects are placed with methods that are to be synchronized (Android,2012). Calls to the methods contained in the synchronized block happen only after a thread has acquired the lock for the object or class referenced in the synchronized statement.

Whenever a thread enters into the synchronized method, it acquires a lock.

[a]

```

/**
 * Adds the given item to the Overlay.
 *
 * @param overlayItem
 * the item that should be added to the Overlay.
 */
public synchronized void addOverlay(OverlayMarkerItem overlayItem) {
    this.overlayMarkerItems.add(overlayItem);
    populate();
}

```

[b]

```

synchronized (checkOverlayItem) {
    // make sure that the current item has a position
    if (checkOverlayItem.getPoint() == null) {
        continue;
    }
    checkItemPoint = projection.toPixels(
        checkOverlayItem.getPoint(), checkItemPoint);
    // check if the translation to pixel coordinates has failed
    if (checkItemPoint == null) {
        continue;
    }
    checkMarkerBounds = this.defaultMarker.getBounds();
    // calculate the bounding box of the marker
    checkLeft = checkItemPoint.x + checkMarkerBounds.left;
    checkRight = checkItemPoint.x + checkMarkerBounds.right;
    checkTop = checkItemPoint.y + checkMarkerBounds.top;
    checkBottom = checkItemPoint.y + checkMarkerBounds.bottom;

    // check if the event position is within the bounds of the
    // marker
    if (checkRight >= eventPosition.x
        && checkLeft <= eventPosition.x
        && checkBottom >= eventPosition.y
        && checkTop <= eventPosition.y) {

        return onTap(i, mapView);
    }
}

```

Figure 5.19: Synchronized example in TER system

Whenever it leaves this synchronized method, it releases the lock. The Figures 5.19 shows two examples of synchronized statements in the TER system.

Figure [a] use a synchronised method to control access to the addOverlays() method. Figure [b] uses the synchronized block, which contains the object and method to control the access to the object and the method inside the block. Basically these synchronized objects such as checkOvelayItem on the map can only be accessed once the thread has the lock. overlayItem can be added only when the thread is locked, thus the synchronized statement prevents two threads from calling checkOvelayItem object (Figure [b]) and adOverlays (Figure [a]) method concurrently.

5.10 Application Permission Setting

By default a basic android application has no permission to perform any operations that would access specific pieces of data or impact other applications, such as reading or writing application files, using users camera systems or performing network access. To make use of protected features of the android device, a programmer can declare the permissions required by the application in the AndroidManifest.xml file.

Permissions requested by the application is granted to it by the package installer during installing time. Mobile users choose whether to grant or not.

Figure 5.20 is the permission required in Android Manifest XML file. The TER system keeps all necessary permissions for running the application. The information collected is only for the purpose of running the application. The TER system does not collect personal information, nor does it give the information to third parties.

The following explain each permission setting in the TER system.

```
<uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<uses-permission android:name="android.permission.VIBRATE"/>
<uses-permission
android:name="android.permission.ACCESS_MOCK_LOCATION" />
<uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission
android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.CAMERA" />
```

Figure 5.20: Permission Required in TER system

- ACCESS COARSE LOCATION - this permission will allow the TER system to access coarse location and keep track where users are, such as the WiFi network-based location.

- ACCESS FINE LOCATION - this permission will allow the TER system to access fine locations such as the GPS location. This setting is almost identical to the above coarse location permission, except that it is more precise when tracking the location.
- ACCESS MOCK LOCATION - this permission will allow the TER system to create mock location providers for testing. It is mainly used during the testing stage in the laboratory.
- WRITE EXTERNAL STORAGE - this permission will allow the TER system to write to external storage, such as image files loaded from the DL server and then saved onto the devices SD card.
- INTERNET - this permission will allow the TER system to open network sockets. For example the TER system needs to load collection data from the DL server.
- ACCESS NETWORK STATE - this permission will allow the TER system to check the network connection. The TER system needs to determine the Internet connection information before it attempts to connect to the DL server, such as connected status, Internet connection type by WiFi or mobile 3G.
- READ PHONE STATE - this permission will allow the TER system read only access to the phone state. The TER system uses this permission to get the phone ID for all tracking purpose during the image loading procedure, such as pausing the application if the phone is ringing.
- CAMERA - this permission will allow the TER system to be able to access the camera device. The TER system uses the camera device to allow users to take pictures.
- VIBRATE - this allows the TER system to access the vibrator. When the TER system reminds the system to wake up, it will provide a notification by vibrating at a certain time.

All these permissions are necessary for the application to run, and it makes sure it is not harmful for users to run the TER application.

5.11 Summary

During the implementation, the TER System has been tested on both the emulator and various versions of android devices including Android platform 2.0, 3.0 and 4.0.

The TER system has been adapted to different versions during the evaluation, so the code is slightly different. This will be covered in the chapter on Evaluation and Case study. All of them follow the above described implementation strategies.

Chapter 6

Case Study: Station of the Cross

Station of the Cross is an art exhibition held every Easter in Hamilton (the Stations, 2012). The exhibition is part of the Easter festival and engages viewers with a story that is an important part of our cultural heritage.

During the 2012 Station of the Cross, we explored the use of our audio guide project for the Stations Exhibition. In the 2012 Station Exhibition, the adaptation of TER system, which is named the Stations E-Book Reader in this section, has been released to the public and evaluated during the event. This section brings up the whole procedure for this particular case study.

6.1 Introduce 2012 Station of the Cross

In 2012, Stations of the Cross exhibited for the eighth year, a series of works from local artists between 20:00 and 22:00 from 1 April 2012 to 7 April 2012 (Figure 6.1). The exhibition was visited by several thousands of people in the Hamilton Paradise Gardens.

In 2012 the Stations held 15 events in 15 locations. These 15 events were ordered based on the story. Generally speaking, it acted as a story book with

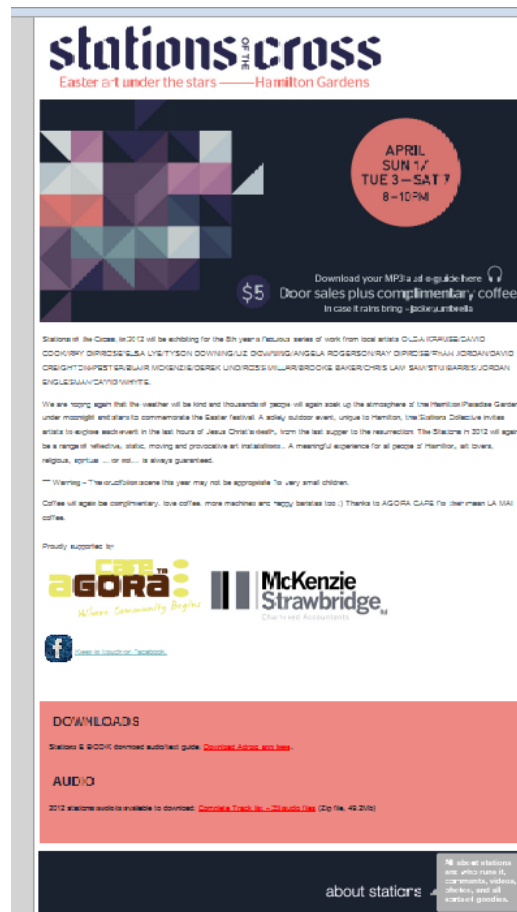


Figure 6.1: 2012 the Station of Cross. From *stations of the Cross*, by the Stations organization Author, 2012, Retrieved from website: <http://www.stations.org.nz/start.php>. Reprinted with permission.

chapters in chronological order. All these 15 events are displayed on the Travel in New Zealand Digital Library collection (Figure 6.10).

The Stations E-Book Reader retrieved the information from the Stations Book, and played the audio for each event. It allowed users to listen to the narrative of the Biblical story, and then hear each artist comment on their exhibit. Users could download the application from the Station website (the Stations, 2012), or were able to have it installed onsite.

6.2 Collect Data

We started by collecting data correctly. The GPS data presented pairs of longitude and latitude coordinates for each event on the Station story. Secondly,

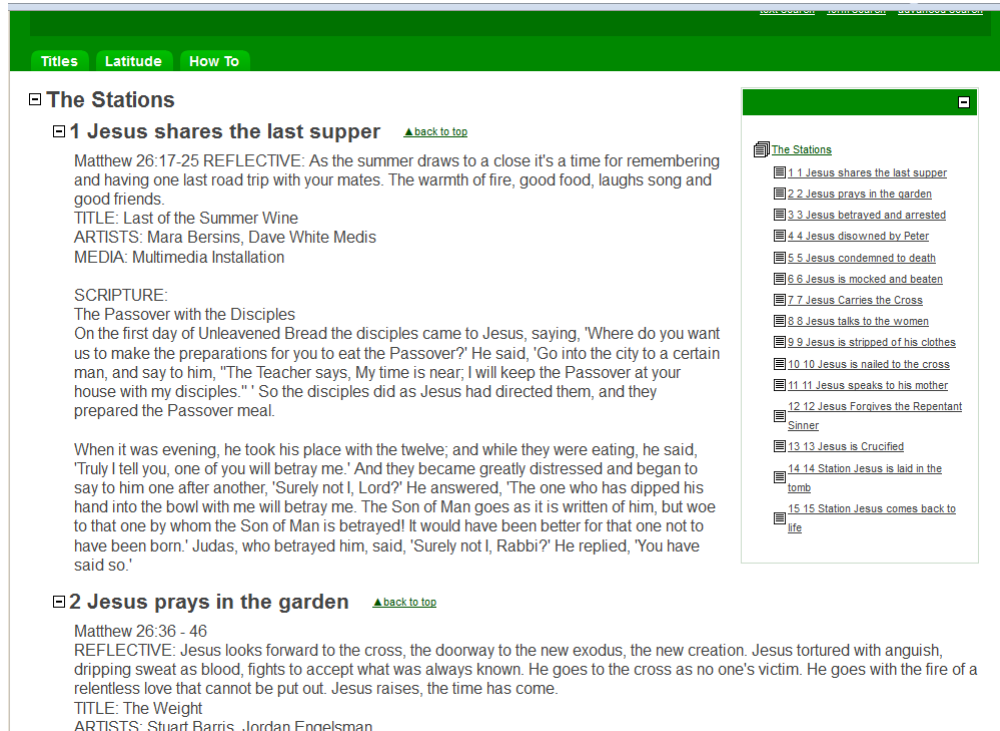


Figure 6.2: The Stations Collection

we collected audio data from the Artists and the Bible to match the 2012 events. This section introduces how the GPS data and Stations event data were collected.

6.2.1 Tools to collect GPS data

We collected the geographical data with pairs of longitude and latitude coordinates by using built-in mobile GPS. In theory, the GPS mainly refers to the particular point of each event. For each event the artist set up their arts in the areas. People walked around the event and viewed these arts, listened to the music or story or participated events.

6.2.2 Measure the Physical Place

The true centre of the Gardens was not exactly the centre of the arts. The application considered the place center and radius of the region and then prompted visitors to start listening to or reading the related story. We col-

lected GPS data four times and through testing we found the place center related to the arts size and type, physical location characters, and the region of the crowd of the people.

For example, if the material used by the artist was a big electronic picture on the wall, and in front of the pictures there was a fence or other barrier preventing people getting too close to the pictures, the place center was not the place where the picture was.

Secondly, the physical location characters were very important issues as well. Such as the location of Stations 2 and Stations 3; both of them had very similar GPS data with different deeps. Stations 2 had the latitude and longitude values of -37.80693246 and 175.3019421, Station 3 had the values of -37.80699838 and 175.302044. Station 2 was positioned above the Station 3, while Station 3 was placed in a tunnel. Initially, both of these two Stations had a radius of three meters. During the test we found when people walk through the Station 2, then the reminder would pick up the Station 3 first, which was inconvenient even though users had the ability to skip to it.

Thirdly, how far ahead people could start to listen to or read the story? The application took into account of the radius of each place and the next nearest place. If the Station was far away from another Station event, the bigger radius worked more efficiently. If certain events were very close, such as station 10, station 12, station 13, and station 14, we found the smaller radius worked better. But, if the radius was much smaller, the reminder wouldnt pick up at the location when the user stood outside of that region. We tested the application whenever we had new GPS data collected (Figure 6.3), then combined the above issues together and found the better way to match what the user expected.

We tested the application whenever we had new GPS data collected (Figure 6.3), then combined the above issues together and found the better way to match the user expected.

the Stations GPS Data in 2012 Hamilton Garden			
Station Event	Longitude	Latitude	Radius
1	175.3017902	-37.80682631	6m
2	175.3019421	-37.80693246	6m
3	175.302044	-37.80699838	4m
4	175.302481	-37.807213	5m
5	175.3027454	-37.8072655	5m
6	175.3027371	-37.80690259	6m
7	175.302821	-37.80678595	5m
8	175.3030401	-37.80676875	7m
9	175.3029641	-37.80641861	6m
10	175.303369	-37.80657412	3m
11	175.3036183	-37.80650854	4m
12	175.3033986	-37.80666975	3m
13	175.3034847	-37.80686098	3m
14	175.3033072	-37.80676067	5m
15	175.3031677	-37.80710399	5m

Figure 6.3: Final Version of the Station GPS data

6.2.3 Collect Audio Files

We focused on combining the simple version of Bible story and the artist speaking together. The simple version of Bible story and Art information would be displayed on the users Android phone and the user could hear the story first followed by the Artist talking about their arts via the Android equipped with Travel E-Book Reader application.

Each event originally had two Audio files in WAV format; one was a simplified story from Bible and one was the artist speaking about their motion and work. We combined the two WAV Audio files together, and then used the free Online Audio Converter tools to convert WAV file to MP3 files (Figure 6.4). So people could download the MP3 file onto their smart phones or MP3 Players to listen from the Station webpage shown on Figure 6.5. The Application could play either WAV or MP3 but MP3 files and could be played in a MP3 Player.

6.2.4 Collect Text Files

The story content of the 15 events in the 2012 Station Exhibition was given in PDF format. We converted the PDF format to XML format in order to be retrieved by the application. Unfortunately, simply copying the text content from PDF to the XML file editor didnt work. The file needed to be converted to UTF-8. Figure 6.6 shows one example of Station XML format. These text































 1 Station Jesus shares the last supper-text-TTS.mp3	Station 1	Stations of the Cross
 1 Station Jesus shares the last supper-title-TTS.mp3		
 2 Stations Jesus prays in the garden-text-TTS.mp3	Station 2	Stations of the Cross
 2 Stations Jesus prays in the garden-title-TTS.mp3		
 3 Station Jesus betrayed and arrested-text-TTS.mp3	Station 3	Stations of the Cross
 3 Station Jesus betrayed and arrested-title-TTS.mp3		
 4 Jesus disowned by Peter-text-TTS.mp3	Station 4	Stations of the Cross
 4 Jesus disowned by Peter-title-TTS.mp3		
 5 Station Jesus condemned to death-text-TTS.mp3	Station 5	Stations of the Cross
 5 Station Jesus condemned to death-title-TTS.mp3		
 6 Station Jesus is mocked and beaten-text-TTS.mp3	Station 6	Stations of the Cross
 6 Station Jesus is mocked and beaten-title-TTS.mp3		
 7 Station Jesus Carries the Cross-text-TTS.mp3	Station 7	Stations of the Cross
 7 Station Jesus Carries the Cross-title-TTS.mp3		
 8 Station Jesus talks to the women-text-TTS.mp3	Station 8	Stations of the Cross
 8 Station Jesus talks to the women-title-TTS.mp3		
 9 Station Jesus is stripped of his clothes-text-TTS.mp3	Station 9	Stations of the Cross
 9 Station Jesus is stripped of his clothes-title-TTS.mp3		
 10 Station Jesus is nailed to the cross-text-TTS.mp3	Station 10	Stations of the Cross
 10 Station Jesus is nailed to the cross-title-TTS.mp3		
 11 Station Jesus speaks to his mother-text-TTS.mp3	Station 11	Stations of the Cross
 11 Station Jesus speaks to his mother-title-TTS.mp3		
 12 Station Jesus Forgives the Repentant Sinner-text-TTS.mp3	Station 12	Stations of the Cross
 12 Station Jesus Forgives the Repentant Sinner-title-TTS.mp3		
 13 Station Jesus is Crucified-text-TTS.mp3	Station 13	Stations of the Cross
 13 Station Jesus is Crucified-title-TTS.mp3		
 14 Station Jesus is laid in the tomb-text-TTS.mp3	Station 14	Stations of the Cross
 14 Station Jesus is laid in the tomb-title-TTS.mp3		
 15 Station Jesus comes back to life-text-TTS.mp3	Station 15	Stations of the Cross
 15 Station Jesus comes back to life-title-TTS.mp3		

Figure 6.4: The Stations Audio Files

files were text to be displayed on Androids screen when the user ran the Station E-Book Reader application.

6.3 Adaptation of the TER system

Due to the following issues, the Station E-Book reader had adapted to the event and it was different from the TER system (Chapter 5).

First of all, the Greenstone running on the Android application is still on testing period, so the Travel E-Book runs standalone on Android devices. Secondly, because of quite a few mountains and valleys around the Hamilton Gardens, the phone frequently lost internet connectivity, and the Station Travel E-Book couldnt connect to the Digital library. Thirdly, the 2012 Station all happened in a tight location, so the Station Travel E-Book Reader did not need the Travel Path advisor. All numbers were clearly identified in the order

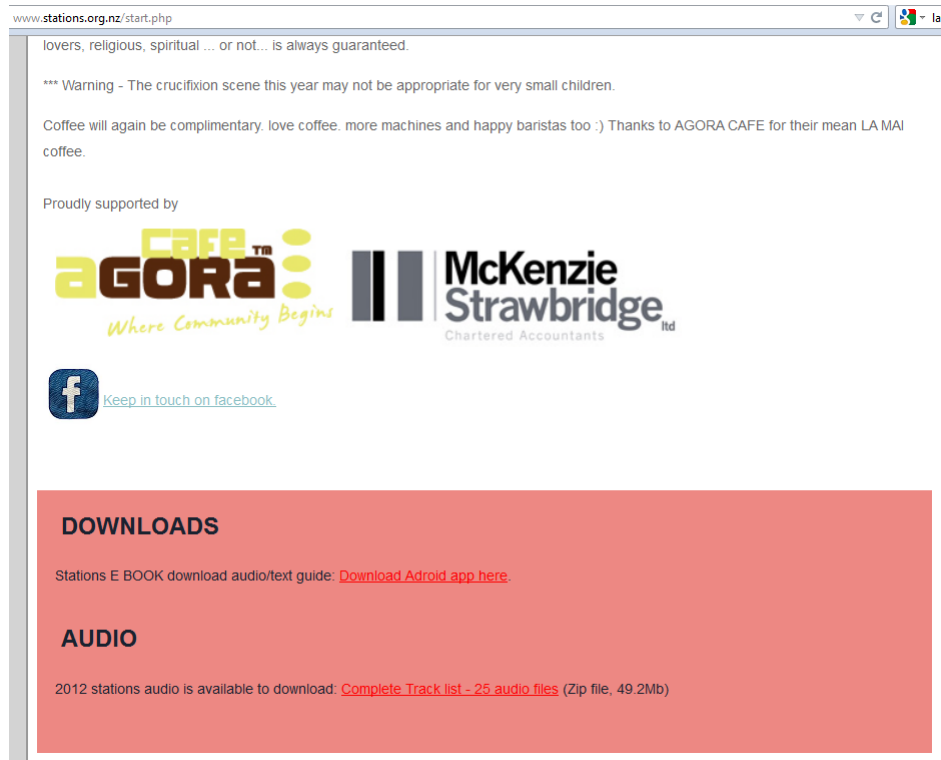


Figure 6.5: Audio Download. From *stations of the Cross*, by the Stations organization Author, 2012, Retrieved from webiste: <http://www.stations.org.nz/start.php>. Reprinted with permission.

of each event. Finally all artists works were not able to be loaded on the phone or other media place. The pictures of artists work was not displayed on the screen and visitors were not allowed to take photos, so the taking image function was disabled in the Station application. Figure 6.7 shows the differences in terms of architecture between the Station E-Book reader (in grey area) and the TER system (in both grey and white area) (Chapter 5). Figure 6.8 shows the overview of the software architecture of the Stations E-Book reader system for the Stations. This section describes Travel E-Book Reader procedures step by step numbered in the Figure 6.8.

- (1) While users walk around, the GPS Location will dynamically identify users location details.
- (2) The GPS data is sent to the Map component. On the mobile screen, the map is displayed with the user location.

```
<location>
  <latitude>-37.80693246</latitude>
  <longitude>175.30194209</longitude>
  <radius>0.006</radius>
  <chapter>2</chapter>
  <title>2 Stations Jesus prays in the garden</title>
  <text>
    Matthew 26:36 - 46
    REFLECTIVE: Jesus looks forward to the cross, the doorway to
    the new exodus, the new creation. Jesus tortured with anguish,
    dripping sweat as blood, fights to accept what was always known.
    He goes to the cross as no one's victim. He goes with the fire of
    a relentless love that cannot be put out. Jesus raises, the time
    has come.
    TITLE: The Weight
    ARTISTS: Stuart Barris, Jordan Engelsman
    MATERIALS: Wooden door, steel, found object
    DIMENSIONS: 2000 x 820 mm

    SCRIPTURE:.
    Jesus Prays in Gethsemane.

    Then Jesus went with them to a place called Gethsemane; and he
    said to his disciples, 'Sit here while I go over there and pray.'
    He took with him Peter and the two sons of Zebedee, and began to
    be grieved and agitated. Then he said to them, 'I am deeply
    grieved, even to death; remain here, and stay awake with me.' And
    going a little farther, he threw himself on the ground and
    prayed, 'My Father, if it is possible, let this cup pass from me;
    yet not what I want but what you want.' Then he came to the
    disciples and found them sleeping; and he said to Peter, 'So,
    could you not stay awake with me one hour? Stay awake and pray
    that you may not come into the time of trial; the spirit indeed
    is willing, but the flesh is weak.' Again he went away for the
    second time and prayed, 'My Father, if this cannot pass unless I
    drink it, your will be done.' Again he came and found them
    sleeping, for their eyes were heavy. So leaving them again, he
    went away and prayed for the third time, saying the same words.
    Then he came to the disciples and said to them, 'Are you still
    sleeping and taking your rest? See, the hour is at hand, and the
    Son of Man is betrayed into the hands of sinners. Get up, let us
    be going. See, my betrayer is at hand.' .
  </text>
</location>
```

Figure 6.6: The Stations Data in XML format

- (3) The GPS location details will be sent to the Preference Module.
- (4) The Preference Module accesses the user profile database (to retrieve the users interest).
- (5) The Preference Module contains Audio Interface Module. The Audio Player Module gives the user flexibility to choose to read, listen or both together. The Preference Module sends the user profile information to the Map Location Manager System.
- (6) The location identifies books and book chapters that relate to the surrounding location of the user and sends them back to the Location System Manager.

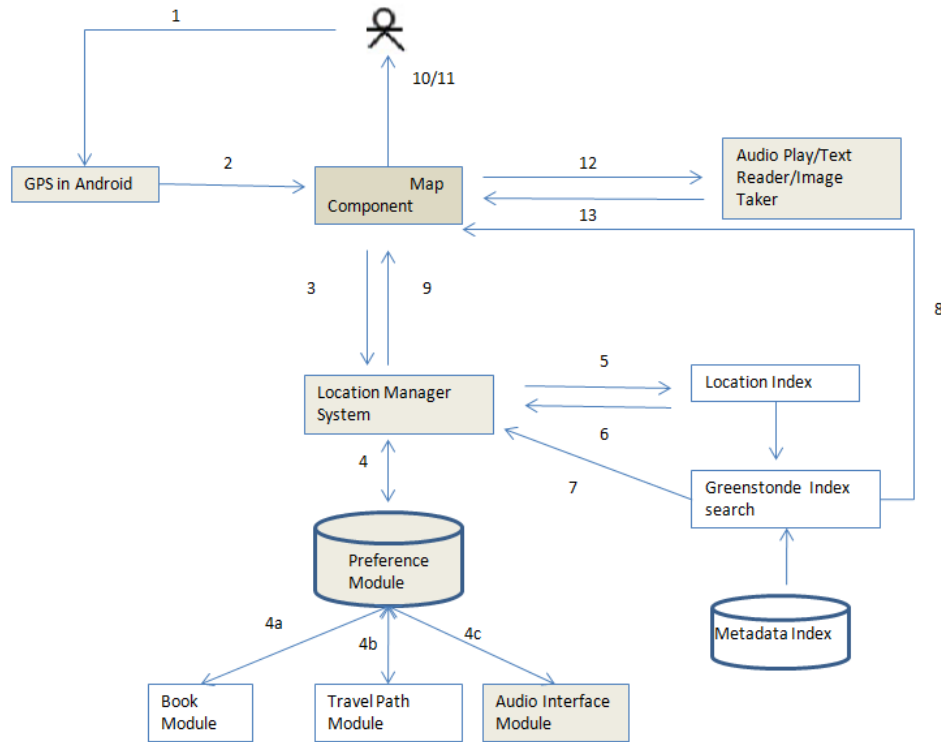


Figure 6.7: Comparison between TER system and the Stations

- (7) The location system sends all chapter numbers and location back to the Map Component.
- (8) The map component displays the locations and book order on the map by using marker and chapter number.
- (9) If a user comes close to the location of a chapter, the map will notify the user by changing colours on screen and using special sound such as chirp chirp
- (10) The chapter is played as an audio book and/or shown as readable text. It allows users to play existing MP3 files, or, if there is no MP3 file it will use the Text-To-Speech application to generate the MP3 files.
- (11) The user can choose play/pause/skip to the other chapters.

The Stations E-Book Reader used offline maps and stored all audio files and text files locally. The maps used was same with the TER system, which was

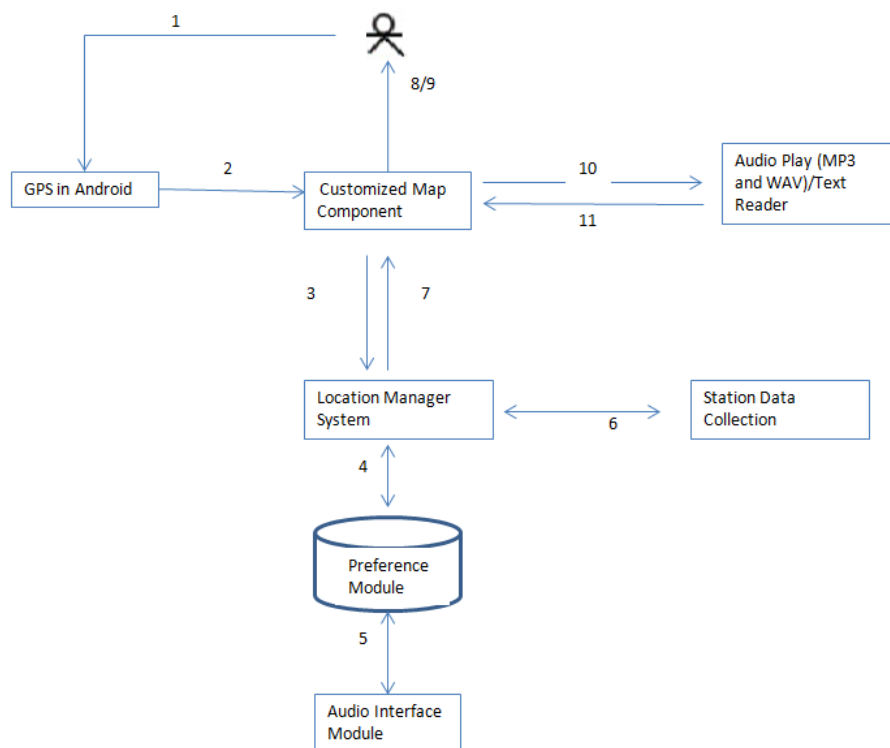


Figure 6.8: The Overview of the Architecture

introduced in Section 5.4. Location markers were inserted using GPS coordinates as the centre of circular areas. Numbers in the Marker are inserted to arrange the order of each event along with the location.

The Station Travel E-Book Reader combined the Text-to-Speech API and Media Player together. Text-to-Speaker was used when there wasn't existing MP3 files to generate the sound. For example, the names of all 15 events were spoken by the Text-to-Speech API, while the Bible story and artists' speech was pre-recorded by people. The Station Travel E-Book Reader called the Media system to play and control the sound. Due to large content of some artists' commentary, the readable text was a shortened version of the audio sound.

Figure 6.9 shows screen shots taken from an example of a user visiting the 2012 Station in the Hamilton Public Gardens.

The user walks through the 15 events in an organised order as prompted by



Figure 6.9: The Stations

the Android phone, which is equipped with the Station E-Book Reader application. When user has passed through the location of Station 1 (the first event) (Figure 6.9 [b]), the phone sounds a “Chirp Chirp” and the marker on the map had been changed slightly, these sounds and markers notify him of something of interest to him, the user can have a few options to play it: Audio only mode, Text only mode, Audio and Text mode. He chose Audio and Text mode. He also wanted to read the text while listening to the Bible story and artist talking; by pressing the button “Show Text” he can read the simplified story version and information on the art works. The screen shot in Figure 6.9 [c] was taken from that position. When he went to Station 2 the reminder sounds again and this time he didn’t want to play the audio file, but rather just continue to Station 3. Now he played the Skip To Next button to listen to the latest audio files which was Station 5. All audio files would be stored in the queue; the top one being the latest one. Where the walkway was slippery he could focus on the path while listening to the Audio Files.

6.4 Group Interview/Observe

Before the Station started the Station speaker recited the poems in front of the visitors for around 5 minutes while people stood in a long queue to enter.

At the end the Station speaker introduced the Station research group and the Station E-Book Reader briefly. Then one of us started to distribute the poster and questionnaire to the visitors.

We did oral group interviews and observed nearly seven hundred people on Tuesday, Wednesday, Thursday and Friday of that week. This interview mainly identified the number of Android users, and people interested in the Travel E-Book Reader onsite.

We divided the whole queue into small groups of seven to nine people and talked with each group while we held the Android phone with the Station E-Book running interface and invited visitors to try the Station E-Book Reader. Some of the questions raised were; “Do you have Android phone?” “Would you like to try out this Station E-Book Reader while walking in the dark area?” “Would you like to listen to the Artist talk tonight for the Stations?”

A major portion of the 2012 Stations visitors were adults of various nationalities. Most of them didnt bring any phone with them; only a few people brought Android phones and all of these Android users were pleased to install the Station E-Book Reader. During the installation we observed and interviewed Android user information such as ages, sex, nationalities and experiences. We mainly focused on the users experience with downloading Android .apk files. More details of these issues will be detailed in the next section.

Through our observation, we found that four out of 11 Android users were males, one of 11 was young male (he came with their parents and sisters), four out of 11 were older adults and six out of 11 were young people. All of them were English speakers with the exception of the young male who was Arabic. All of them tried to install the Station E-Book Reader onto their phones with the help of the research group.

6.5 Release/Install software in the public

The Station E-Book Reader had been released to the public using the release model while the 2012 Stations Exhibition started. The simplified version of the Stations collection had been put onto the Digital Library website (Figure 6.10). All this information was introduced in the Station research poster (see Appendix poster). This section focused on non-Market .apk installation challenges and how to install .apk in the public without internet connection.

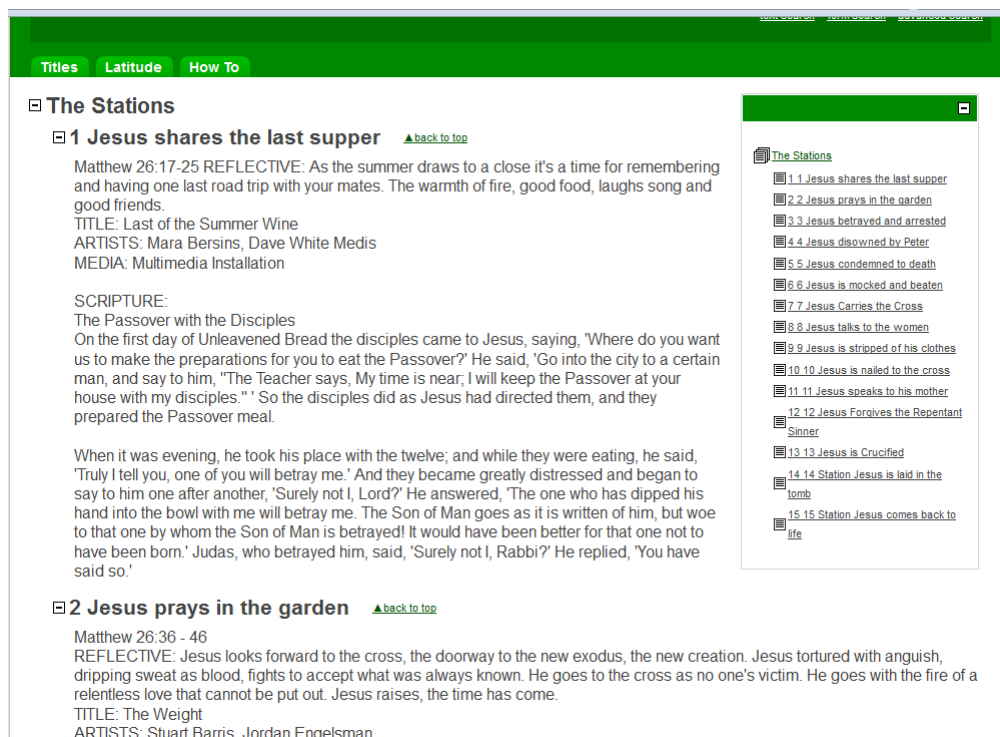


Figure 6.10: The Stations Digital Library Collection

6.5.1 Non Market .apk Installation

All installable Android applications use the .apk file type format. A file with .apk extension is an Android Package file and can be installed on any Android powered device. While the Android Market can download and install the application automatically, the Station E-Book Reader is non-Market .apk application.

Unlike Windows, Mac OS X or the Linux Desktop operating system, where user can simply double click to run the installer, Android does not include a File Manager in stock form.

6.5.1.1 Run through File Manager/Installer

Generally there are three different ways to install non-Market .apk applications onto Android devices (Android,2012).

- (1) Install .apk Application with an APK Installer or File Manager.
- (2) Push Install APK via Android SDK on computer.
- (3) Install from Computer without Android SDK with Droid Explorer.

All of these methods need a prerequisite, such as an APK Installer or File Manager for Method 1 and Method 2 and Method 3 needs the proper the Google USB driver to be installed on the device in order for Windows to recognise the device.

If the Android device doesn't have Installer or File Manager, users download the Installer or File Manager free from the Android Market with their Google Account.

6.5.1.2 Run through USB driver

If the Android Application is developed on Mac OS X or Linux, then it doesn't need to install a USB driver to test the application. The Station was developed on Windows, and therefore requires the installation of the appropriate USB driver to communicate with the devices .

Google USB Driver

The Google USB driver is a downloadable component for the Android SDK, available from the SDK Manager. Users can then go to Android developer

website (Android,2012) to download the proper USB driver. The Google USB driver is only available for the following devices, and the The Galaxy Nexus driver was distributed by Samsung. Android devices using Google USB driver:

- ADP1 / T-Mobile G1
- Verizon Droid
- Nexus One
- Nexus S.

OEM USB Driver

All other Android powered devices use original equipment manufacturers (OEM) driver. Figure 6.11 provides links to the web sites for several original equipment manufacturers (OEMs), where users can download the appropriate USB driver for your device (Android, 2012).

6.5.1.3 Intall the USB Driver

Once we downloaded the appropriate USB driver to the Android device, we launched the driver and installed it. The following describes the install of the USB driver via Windows XP steps by steps.

- (1) Go to Device Manager on the computer.
- (2) Click Properties - Update Driver - Browse - Let me Pick - Computer - Have Disk - Browse then selected the driver and download the file.

This section introduced two ways to install and run the .apk file via the Windows: File Manger System or other Installer, USB Driver. The USB Driver consists of two types: Google USB and OEM USB. Different USB Drivers work for different versions of Android powered devices. The install then downloaded the USB Driver to run the .apk file locally. Form File Manager System users would be able to view and touch the .apk icon on the main menu (Android,2012).

OEM Drivers

Note: If your device is one of the Android Developer Phones (purchased from the Google Play publisher site), a Nexus One, or a Nexus S, then you instead of an OEM driver. The Galaxy Nexus driver, however, is distributed by [Samsung](#) (listed as model SCH-I515).

OEM	Driver URL
Acer	http://www.acer.com/worldwide/support/mobile.html
ALCATEL ONE TOUCH	http://www.alcatel-mobilephones.com/global/Android-Downloads
Asus	http://support.asus.com/download/
Dell	http://support.dell.com/support/downloads/index.aspx?c=us&cs=19&l=en&s=dhs&-ck=anavml
Foxconn	http://drivers.cmcs.com.tw/
Fujitsu	http://www.fmworld.net/product/phone/sp/android/develop/
Fujitsu Toshiba	http://www.fmworld.net/product/phone/sp/android/develop/
Garmin-Asus	https://www.garminasus.com/en_US/support/pcsync/
Hisense	http://app.hismarttv.com/dss/resourcecontent.do?method=viewResourceDetail&resourceId=16&type=5
HTC	http://www.htc.com Click on the support tab to select your products/device. Different regions will have different links.
Huawei	http://www.huaweidevice.com/worldwide/downloadCenter.do?method=index
Intel	http://www.intel.com/software/android
KT Tech	http://www.kttech.co.kr/cscenter/download05.asp for EV-S100 (Take)
Kyocera	http://www.kyocera-wireless.com/support/phone_drivers.htm
Lenevo	http://developer.lenovomm.com/developer/download.jsp
LGE	http://www.lg.com/us/mobile-phones/mobile-support/mobile-lg-mobile-phone-support.jsp
Motorola	http://developer.motorola.com/dorstools/ISR_Drivers/

Figure 6.11: The OEM driver download URL, *OEM USB driver*, by Android Developers Author, 2012, Retrieved from <http://developer.android.com/tools/extras/oem-usb.html>. Reprinted with permission

6.5.2 Experience of Non Market .apk Installation

Most visitors were unable to wait for installation before they entered. We couldnt install the appropriate USB Driver for users onsite due to time restraints. The two ways to install the .apk were either via their File Manager System or their Android USB Driver already available. Next, users could just touch the icon and run it. Alternatively, users have to connect to the internet to download File Installer or USB driver.

In order to copy the files from our computer to users Android devices, users are required to turn on the USB connection to allow transfer files between com-

puter and Android devices. To run the .apk file, Users have to change a few settings. Firstly, turn on “Unknown resource” through Application Setting menu. Secondly, turn on ”USB debugging” via the “Development” setting. And finally users need to turn on “Use GPS Satellites” through “Location and Security” setting. The Station .apk files are non-risky for all Android devices. So all of these setting don’t raise any risky for Android device, and cost nothing for users’ account. The most important thing is all procedures were clearly explained to the user before process.

We asked individual users’ permission to sign the form before the installation. We used the computer equipped with Android SDK and various cables to install the Station .apk. The various cables were used for different versions of Android device, such as version 2.3, version 4.0. By connecting to our computer via the right cable, we copied the Stations .apk onto users Android devices. The provided cables worked fine for all Android device except for one.

The computer was used to help users with the appropriate USB Driver to run the .apk file. Only one Android device was able to run it through this option.

Also, Android powered devices support different language setting, such as English, Chinese, and Arabic. The installer needed to have knowledge of Android device settings. All of the Android users were using the English setting except one, using Arabic. That was the one which failed to copy due to lack of knowledge about Android device setting. The owner was a school boy who was unable to find the location of the Development setting. He wasn’t familiar with the phone and we didnt have enough time to guide the boy before he entered.

Eventually nine out of 11 Android users had been installed successfully, but not all of them could view the icon of the Stations Application. Only three of them could run it by using the File Manager which was preinstalled on their device.

6.6 Testing

We observed and interviewed eight users using the Station E-Book Reader in the 2012.

Stations Exhibition. The methodology used was observation and interview. Four users were observed and interviewed after the testing and four users had a short interview afterwards in the Hamilton Gardens.

The Event took place between 20:00 and 22:00 in the evening. Even though each event location had a signed board to describe what it was, visitors found it hard to read due to the light and amount of people. Visitors couldn't really stop to read text on the board due to following visitors. The physical location of the Hamilton Garden consists of valley/hills, Visitors were aware of the uneven surface of the ground while visiting.

Before the users entered the Gate, we briefly explained what the application did, highlighting the Text with Audio function based on the particular environment of the Stations. Users then held the Android phone supplied by the department of Computer Science of the University of Waikato. We observed the journey of four users who followed the guide. Four users visited the Stations with the phone but without us. There were short interviews following the testing.

6.7 Feedback on the software

The Station .apk was available to be downloaded via the Stations website and Digital library website. These links were clearly explained in the Stations poster. We didn't keep track of the sites information, and the only way we could get feedback was via the phone or email. Until now we havent received

any feedback from the public regarding site access, download amount and the usage of the software. The Feedback described here is mainly based on the observation and interview through our testing. Due to time limitation the short interview was captured just after the testing and at the entrance of the Stations.

The interview questionnaire focused on the users' satisfaction of the Stations Application. All four questions were open questions and encouraged users to express their opinion. The first question focused on the interface design, the second question generally evaluated functionality of the software, question three and question four mainly investigated the usage of the software and improvement and the last question provided the user space to express based on their knowledge and experience.

Usage Time

Figure6.12 shows the overall length of time participants spend using the service. The whole journey lasted between 30 minutes and two hours, Most participants use the service for an average 40 minutes and two participants encountered the lack of battery power. One participant said it couldnt get GPS location signals from GPS Satellite Service due to the weather, though the service was still running.

Interface Usage Evaluation

The interfaces refer to the map, icon, number, title and preference setting interface, such as check box and radius button.

All participants (7 of 8) were satisfied with the map, location marker and number. All users appreciated the numbers which displayed the travel path of the event. None of users tried the Preference setting due to lack of knowledge of the software. A problem identified was that users werent aware that they

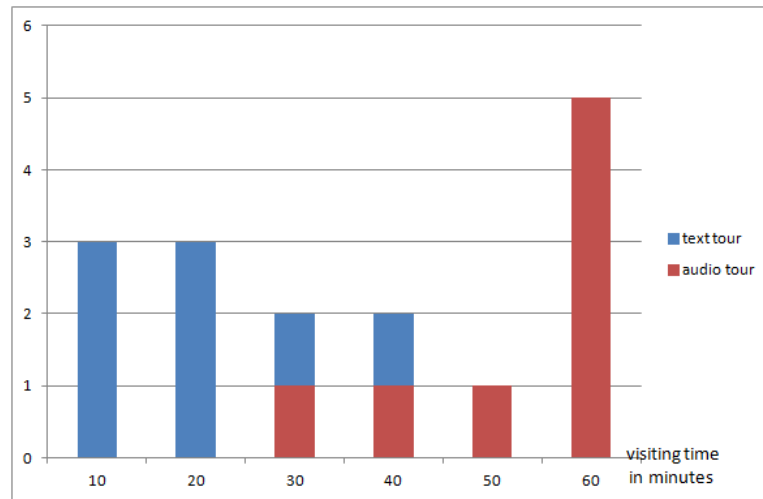


Figure 6.12: Usage Time

could change some interface settings. One user suggested it would be better to have start menu which introduced the event and the software.

Functionality Evaluation

All users were given the original audio setting with AUDIO WITH TEXT to start with. None of them utilised any other audio setting. Most participants (6 of 8) were satisfied with the audio sound. The audio reminder and audio information was generally described as being helpful and interesting. One user said “It was exciting to hear the artist talking when looking at their works”. One of the participants was not happy about listening to the audio for a long period with the earphone. Another problem we found was that some stories didn’t really match the station event, for example; when users walked closed to the Station 1 the map showed availability of Station 3. The majority of participants played and read the audio in order. One participant preferred to play/read the sound whenever they touched the location marker on the map. Most participants (7 of 8) were satisfied with the touch interaction, while one participant preferred to touch he location marker which would then pop up the content of that event.

6.8 Summary

Table 6.1 shows these 11 participants' (which were invited to install .apk file) information. They were people who were invited to install the TER system on their own devices. Most users were adults and Android devices were 2.1 version or higher. Two out of 11 devices were Galaxy Nexus, three Nexus, one IDEOS and the remaining six were different models of Android powered devices. All users who had been observed in the case study were provided the Galaxy Nexus by the Computer Science department of the Waikato University.

Table 6.1: TER Installation in the Public

Device Type	Galaxy Nexus	Google Nexus One	Google Nexus S	Huawei Ideos	Motorola	HTC Innovation	Others
Amount	2	3	1	1	1	1	2
Ages	36-50	20-35	36-50	36-50	20-35	<20	36-50
Gender	M	2:M,1:F	F	M	F	M	F
Stored	Yes	Yes	Yes	Yes	No	Yes	No
Running	No	No	OK	OK	No	OK	No

The framework API that an Android platform delivers is specified using API Level. Each Android platform version supports exactly one API Level, although support is implicit for all earlier API Levels. All Android applications need to specify the minimum API Level on which the application is able to run. The Stations application set a minimum of SDKVersion 8. All of Android users (11 users) were able to run the Stations application. Most Android devices to date run the older OS version 2.3.x Gingerbread in which API level is 9 to 10 (Figure 6.13).

Table 6.1 also summarises the installation of the software in the public. The last two rows identify only a few people (11 users) who brought their Android phone with the Stations Event. Nearly all the Android users were happy to install the free software for the event. The majority of users are adults with the exception of one school student. Four Android users were female. Most Android users (6 of 9) didn't have any experience with the installation of non

Version	Release date	API level	Distribution
4.0.x <i>Ice Cream Sandwich</i>	October 19, 2011	14-15	7.1%
3.x.x <i>Honeycomb</i>	February 22, 2011	11-13	2.7%
2.3.x <i>Gingerbread</i>	December 6, 2010	9-10	65%
2.2 <i>Froyo</i>	May 20, 2010	8	19.1%
2.0, 2.1 <i>Eclair</i>	October 26, 2009	7	5.2%
1.6 <i>Donut</i>	September 15, 2009	4	0.6%
1.5 <i>Cupcake</i>	April 30, 2009	3	0.3%

Figure 6.13: Usage share of Android version

market .apk file.

Figure 6.14 summarises the presentation preferences of the participants for



Figure 6.14: Comparison of audio and text presentation

this study. Eight users took both audio and text tour during the case study. The majority of audio participants (7 of 8) preferred the audio presentation over the text presentation, one of them would like to have another option which allows to read the content whenever they touch on the location marker on the map, so that the traveller won't be closed to the location and the story but will still be available to read. Another reason was that the reminder suddenly made the noise at the quiet area without earphone. Text tour participants (5 of 8) preferred the audio presentation over the text, two non-native speakers preferred to read the text in their own time and one participant said he would like to watch the artist's works without text file. The reason given was text wasn't comfortable to read in the dark area and small screen.

Figure 6.15 shows users' suggestions for further improvement. Allow users to play the audio in their own time: All audio files were initially queued in a

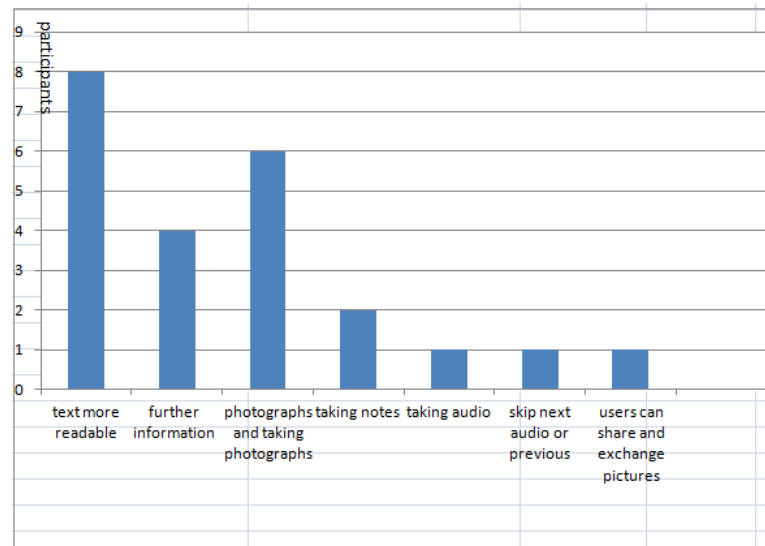


Figure 6.15: Further development suggestion

line, so the latest one always played first. Users said they would like to play in their own time.

Participants like the idea of more interaction involved in this service, such as taking pictures, noting their own feeling and, taking audio files and share information between different mobile users.

In short, as more and more techniques become available via the mobile devices, users want to increase the usage of their mobile device in order to enrich the service.

Chapter 7

Evaluation

This section describes the assessment and evaluation method of the TER system. There are two methodologies used in the study. One methodology used is to evaluate the technical merits of the created software, and the other is to measure the innovation and the new model used in the travel tours (UsabilityNet,2012).

- Questionnaire Method designed based on ISO 9241 part 11. this method is generally used to evaluate the usability of TER software product.
- The SWOT analysis. This method examines the strengths, weaknesses, opportunities and threats. It was developed to evaluate the new travel techs knowledge used in the tours area. It was used to evaluate the TER software against the SWOT features.

These analyses were performed by the project partners as well as by external evaluators. Participants were involved in the evaluation of Auckland and Waikato. The TER system has been adapted and tested in two events. One is “the Stations” event in April 2012 held in Hamilton Gardens. Chapter 6 introduces this event in more detail. Secondly, the TER system has been tested in the “Poetry on the Pavement” event on July 2012 in Auckland City. This event tested nearly all features in the TER system, so this section will mainly focus on the evaluation based on this event process.

7.1 Objective of the Evaluation

The TER system has been tested in the context of two real-world events held in Auckland and Hamilton. One is for the Stations event held in Hamilton Garden in April 2012. One was for the Poetry on the Pavement (Poetry event) held in Auckland city center in July 2012. Evaluations involved in these two events were slightly different. For example, the Stations involved installations and avoided pictures based on the requirements of its organisation. The Poetry event focused on the software product rather than the event. The Stations was introduced in Chapter 7, so this Chapter will mainly focus on the Poetry event.

In the Poetry event TER has learned experiences gained from the Stations event. For example, it has learned how to collect efficient GPS data, (introduced in the Chapter 6) which was used in the Poetry event to make the process more smooth and efficient. To focus on software product, the Poetry event TER system was not installed for any other mobile users. There was also some re-design and re-development involved in the TER system. For example:

- Functionality was added to the TER system which allowed a user to read the stories in their own time;
- A formatter was added to the TER system which formatted the text in a user friendly manner.

These were suggestions gained from the Stations event.

The Poetry event was the journey of poems which stretched from Karangahape Road down to the Auckland waterfront. All of the poems were composed by New Zealand poets and reflect on aspects of Auckland places, historic events or personal memories. It was advertised on the Auckland City Council website. All poems were available for travelers to download. Figure 7.1 shows the advertisement from the Auckland City Council website (Auckland,2005).

In the Poetry event the TER system built its DL collection based on these poems (Figure 7.2). For testing different books the TER system created another

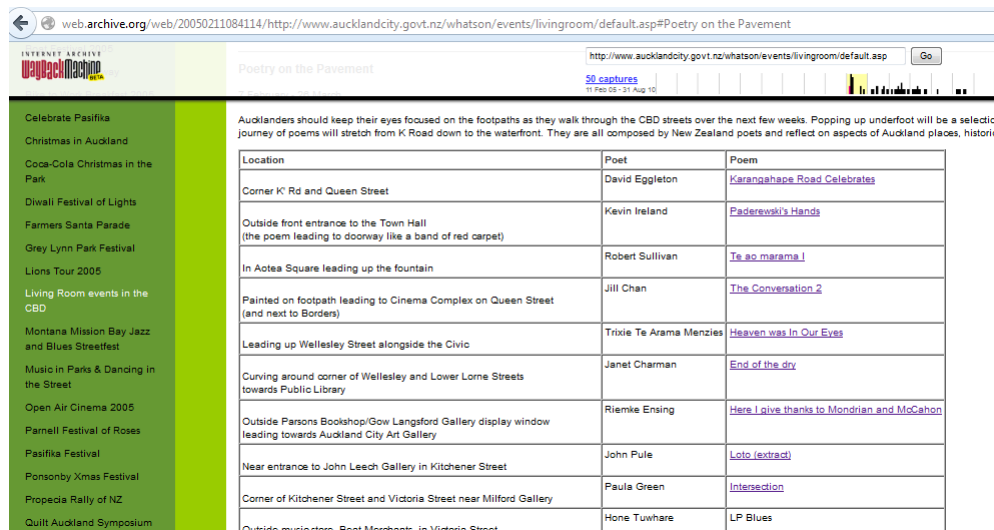


Figure 7.1: Auckland City. From *Auckland City Council*, by City Council Author, 2012, Retrieved from website: <http://www.aucklandcity.govt.nz/whatson/events/livingroom/default.asp>. Reprinted with permission.

book, which consists of an introduction to Auckland CBD streets (Figure 7.3). The content of this book was mainly from the Wikipedia Website, and was for research purposes. Participants were grabbed in the Auckland CBD and asked to follow the Pavement Poetry event on a mobile phone provided by the Computer Science department of the University of Waikato. Afterwards the researcher conducted an interview during which a questionnaire was filled in. This took between 5 and 30 minutes.

In the Pavement Poetry event TER system build its DL collection based on these poems (Figure 7.2). For testing different books TER system created another books which consists of introduction of Auckland CBD streets (Figure 7.3). This books contents are mainly from Wikipedia Website for research purpose. Participants were grabbed in the Auckland CBD city and run TER system to follow the Pavement Poetry event on a mobile phone provided by the Computer Science department in the University of Waikato. Afterwards the researcher conducted an interview during which a questionnaire were filled. This was taken between 5 minutes and 30 minutes.

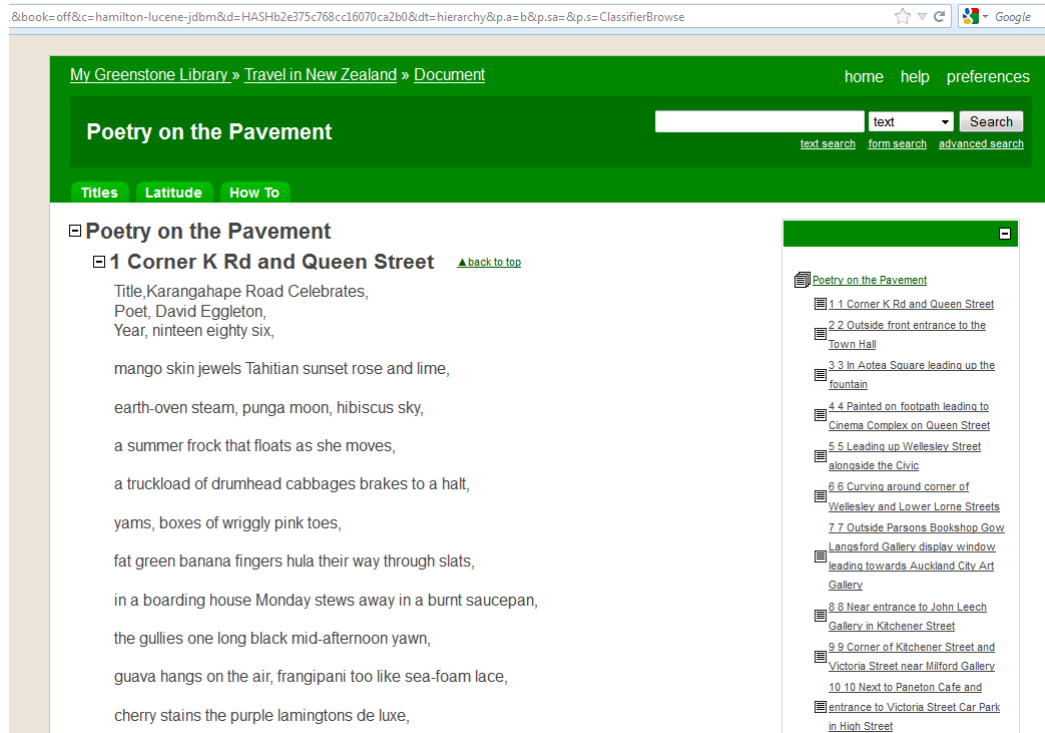


Figure 7.2: Poetry on the Pavement

7.2 Method

Two evaluation methods were involved in the evaluation study. One was questionnaire, with a design based on the ISO standard, and which contained 4 groups of questions. The second was using the SWOT method to evaluate the software. For the user study conducted in the Auckland CBD, most participants were walking around the city with some luggage. Therefore during the evaluation participants answered the questionnaire verbally, and the researcher filled in the questionnaire with their answers.

7.2.1 Questionnaire Method

The purpose of this method was to evaluate the usability and accessibility of the TER product. It asked participants a number of questions using a questionnaire which consisted of two parts:

- The first part asked participants general questions about themselves and relevant experiences while travelling.

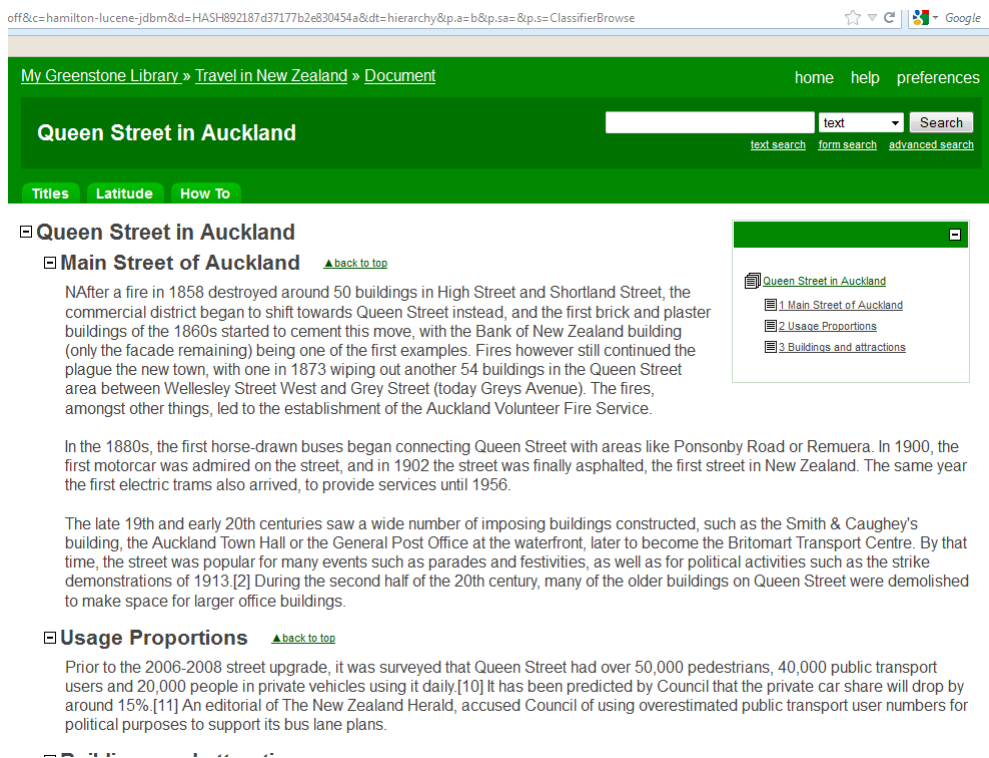


Figure 7.3: Queen Street in Auckland Stories in DL

- The second part was an evaluation of the services with the TER technique.

In ISO 9241Part 11, usability is defined by the quality of use: “The effectiveness, efficiency and satisfaction with which specified users achieve specified goals in particular environments” (Bevan et al 1995). Based on this standard, the questionnaire was designed to measure the TER system usability. It has five scales named Efficiency, Helpfulness, Controllability, Learnability, and Satisfaction. The questionnaire consisted of 25 statements. Thirteen of the 25 statements applied a rating on a five point rating scale, starting from 1 “predominantly disagree” through to 5 “predominantly agree”. The remaining 12 statements provided participants with the options “Did not like”, “OK”, or “Like a lot”. A further option (“feedback”) was provided for arbitrary answers.

There were five parts in the questionnaire. Part 1 consisted of general questions which asked participants about themselves. Part 2 was questions which contained seven statements on a five point rating scale starting from 1 (“predominantly disagree”) to 5 (“predominantly agree”). Part 3 contained 15

statements, each of which asked users whether they “did not like”, “like”, or “like a lot”. There was a feedback option below each item. This part evaluated participant experience of the TER software and it covered five aspects of usability:

- Efficiency of task, which focuses on the map, map markers, images, speed and text in the TER system;
- Helpfulness of task, which focuses on locations, chapter numbers and travel path issues;
- Controllability of task, which focuses on listening/reading, book selection and camera issues;
- Learnability of task, which focuses on audio interface, interface design and preference setting issues;

Part 4 contained 13 statements which provide a five rating scale starting from 1 (“predominantly disagree”) to 5 (“predominantly agree”). Due to time consumption, only four participants got through all the questions. The rest of the participants only picked up a few questions to answer. The records of this part have been considered in the SWOT analysis.

7.2.2 SWOT Method

SWOT analysis is a technique commonly used as an evaluation method to assist in identifying strategic issues. The predictive capabilities of the technique come about from the consideration of the system’s strengths and weaknesses in the overall system environment. The overall environment involves “users, tasks, equipment (hardware and software), and the environments which influences the interaction” (Bevan 1994). This method also identifies the opportunities and threats of the software system. These questions were listed in Part 5 of the questionnaire.

7.3 Participants

The Poetry event was held on the Auckland CBD streets. There were 20 participants involved in the study. Fifteen participants were invited from the public who were walking around on the city street. Five participants were invited with computer backgrounds. Fifteen participants were randomly selected from the public based on wide range of nationalities and age groups. The only requirements were that they were happy and free to participate in the study. All participants were provided research consent forms to sign first. This study was conducted over two different days. Thirteen people participated on a sunny day, and seven participated on a rainy day. During the test stage a smart phone was provided and the researcher followed the participants on the journey. The whole journey took more than one hour. Most of participants only spent an average time of half an hour on the journey. The travel path depended on the place where the participants were, and where they were going. All participants had at least listened or read 5 chapters of the book and experienced all of the features listed on the questionnaire.

7.4 Analysis and Result

All data collected was based on the questionnaire and interview, using the above two methods. This section analyses the data recorded according to the questions on the answer sheet.

7.4.1 Participants Background

Both the general questions and relevant questions allowed us to gain insight into the participants' backgrounds. From observation we found participants are from various nationalities, and half the participants were male. The evaluation tried to cover all age groups, but in fact focused on younger people and middle aged people. Figure 7.4 shows most of them belonged to younger and middle age groups. This happened because people of these age groups are more likely to walk in the street.

Each relevant question provides a rating between 1 and 5. The total points

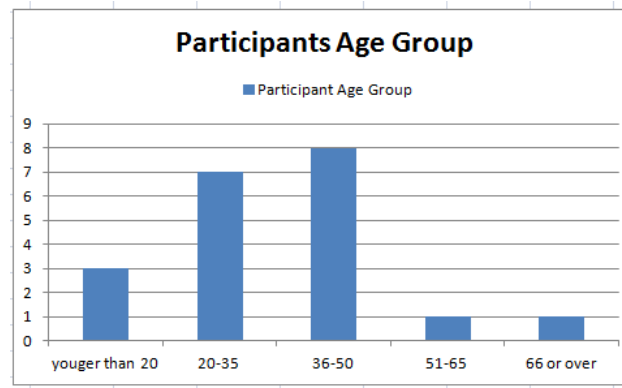


Figure 7.4: Participants Age Group

indicated in Figure 7.5 are made up of 20 users, and it shows how frequently people use material. People mostly use digital maps (total of 76, as shown in Figure 7.5) to help their travelling. There were two common ways to get the digital map found during the interview: loading from Google maps; downloading digital maps to their cell phone. The second largest point is the usage of guide book, which include booklets, physical maps and other guide book information (some participants did not specify what kind of guide books during the interview). From the survey people said this information was mostly from the AA shop, poster shops, information centers, travel agents or other services. Figure 7.5 also shows less usage for audio books while travelling. Others includes pre-visited friends, and comments and feedback of experienced travelers. Figure 7.6 compares the frequency (how often people use it) and percentage (how many people use it) of use of materials while travelling. When looking at the total point of each relevant question, we must take into account that the total point may result in both number of people who used and how often people uses. The total point for reading books is 59 which is lower than usage of guide books and digital maps, but number of people who use these three materials are very close. That means higher frequency of using the guide books and digital maps. The total point of travel applications is only 39 but it had been used by 40 percent of 20 participants.

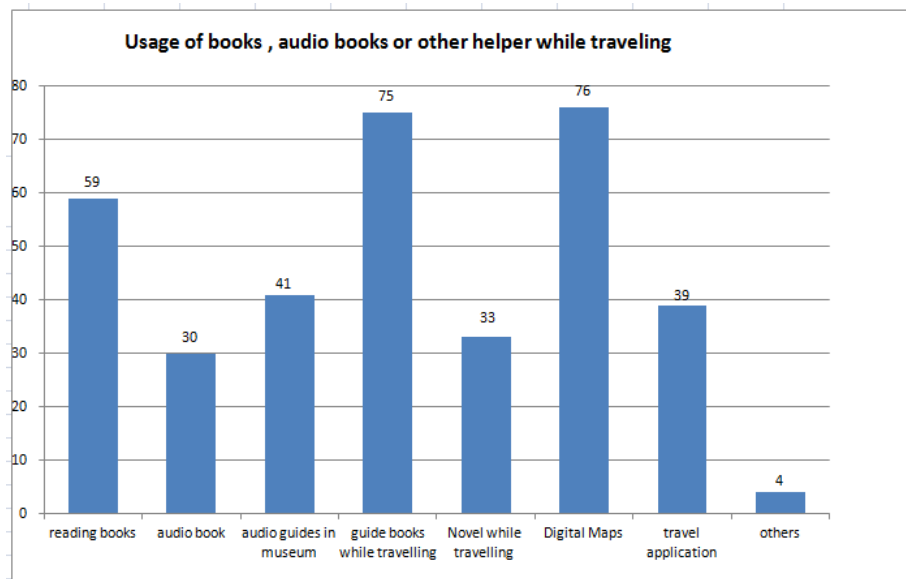


Figure 7.5: Use of books, audio books and others while travelling

Usage of books, audio books or other helper while travelling	Total point	Number of used participants
Reading books	59	17
audio book	30	6
audio guides in museum	41	9
guide books while travelling	75	18
Novel while travelling	33	9
Digital Maps	76	18
travel application	39	8
others		4

Figure 7.6: Comparison of usage of books while travelling

7.4.2 Checking its Usability

The usability evaluation contains 15 questions which are divided into five parts. Efficiency contains five questions, Helpfulness contains three, Controllability contains three, Learnability contains three, and Satisfaction contains one. Each question is provided with a feedback option.

Figure 7.8 shows the main score and the corresponding scale of usability for the task. The calculation method was involved in the data analysis. All questions were provided with three choices: “Did not Like”, “OK”, and “Like a Lot”. For the compare and measure software we transferred data into qualitative data. In order to reduce errors we used two ways to transfer the data. First the quantity data is transferred in turn into a 3 scale rating, scaled with 1, 2 and 3. The second is to change the rating scale from 1 (“Did not Like”),

3 (“It is OK”) and 5 (“Like a Lot”). Figures 7.7 [b] with Title “Graph 2” is based on this rating. Compared to each of two scale ratings the results are not significantly different. Therefore most analysis are based on the rating scale of 1, 2 and 3. Based on users answers we sum all point for each question to get its total point.

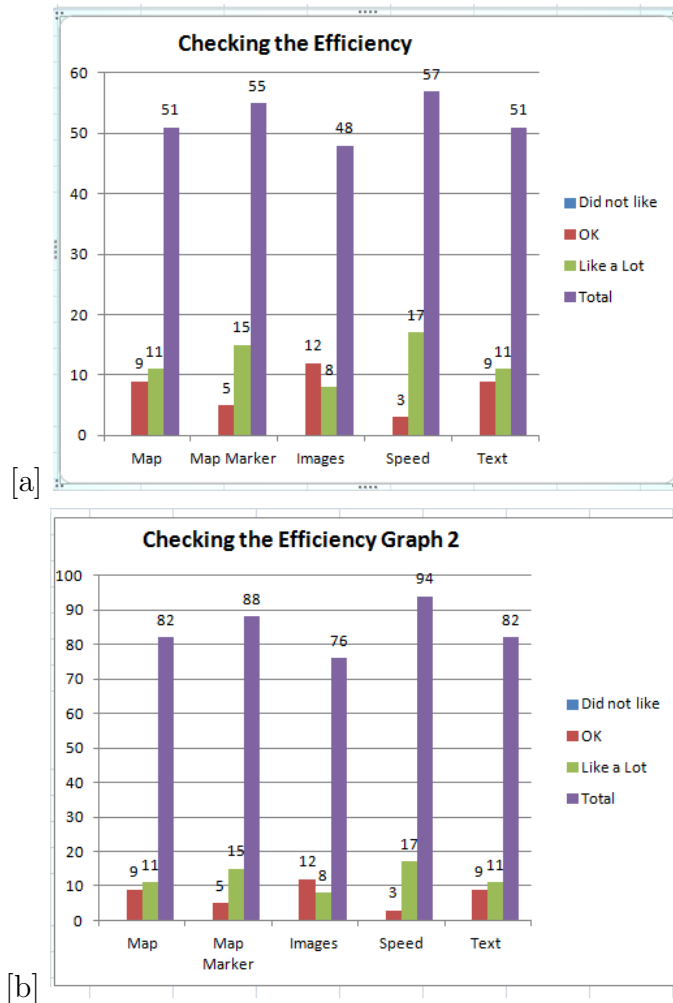


Figure 7.7: The Poetry DL and XML

In Figure 7.8 it is shown as “weight”. For each question the highest mark for the 20 participants is 60, which means all users choose “Like a Lot”. We calculate the rating by using actual total point divided by 60. The rating shows how much satisfaction there is by all participants for each question. The result shows the application is successful in terms of these five items: Efficiency, Helpfulness, Controllability, Learnability and Satisfaction. The application

is more satisfied in terms of Efficiency, Helpfulness and Controllability than Learnability and Satisfaction. The section below has an analysis of each of them.

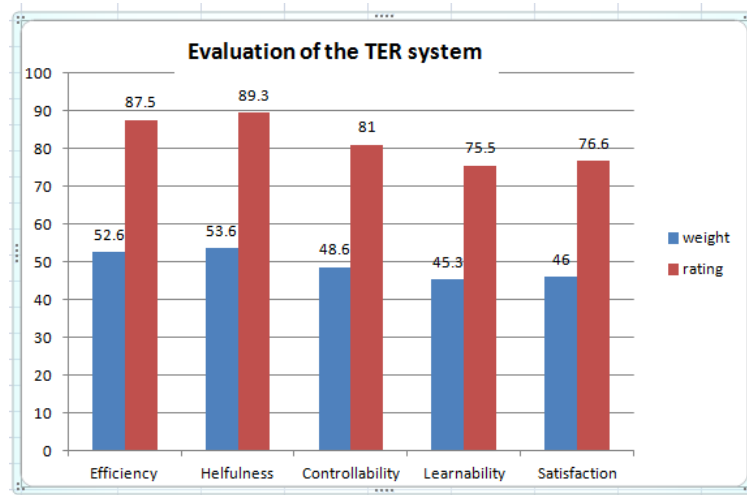


Figure 7.8: Overall Result of Evaluation

7.4.2.1 Checking the Efficiency

Figure 7.9 shows the result of checking the efficiency procedure. This part tested efficiency of using map, map marker, images, speed and text. The number of participants of each choice are shown in its related column, and total points are calculated based on its rating score. No participants chose “Did not Like” for all questions. The highest mark for each question is 60. The total point of each question is all over 50, so the percentage of each question is over 80 percent. The results of this procedure shows that the application in terms of efficiency is successful. There were a few participants which showed negative feedback of the item usage.

- The images are not clear to view.
- The images can not deleted.

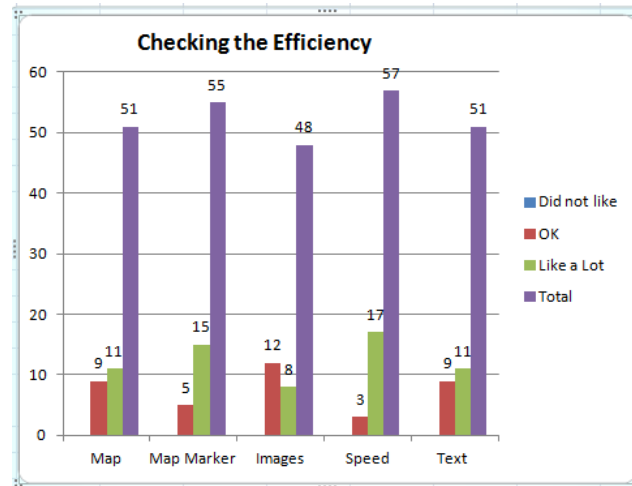


Figure 7.9: Checking the Efficiency

7.4.2.2 Checking the Controllability

Figure 7.10 shows the item selected for checking the controllability. It focuses on controllability to use the application and how easy it is to control the application. The number of participants of each choice are shown in its related column and total points calculated are based on its rating score. The result shows users are mostly satisfied with the controllability of reading and listening, and taking pictures or not. The lower record of the usage of book selection shows people are more likely to redesign this part. There are a few significant negative answers (four people did not like two items) for usage of listening/reading and book selection. These peoples reasons are described below.

- The reminder system interrupts my travelling.
- The number of selectable book are limited to two.
- There is not enough information about books, such as author, published year.

7.4.2.3 Checking the Learnability

Figure 7.11 shows the main score of learnability for the task. The number of participants of each choice are shown in its related column and total points were calculated based on its rating score. In this Figure the highest score (total

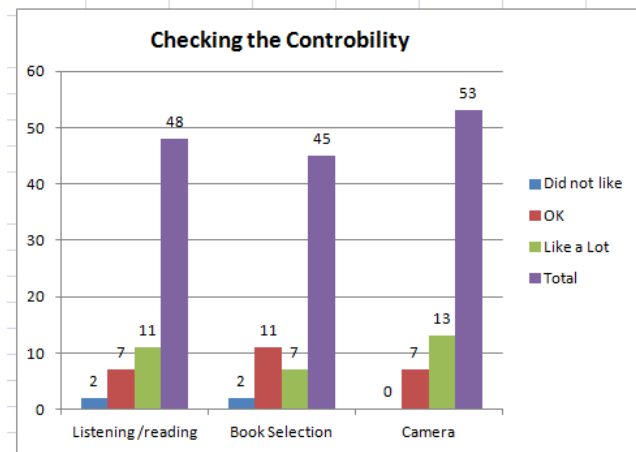


Figure 7.10: Checking the Controllability

point of 53) shows the interface setting is clean and meaningful. The total point of usage of Preference Setting is lower than other two in this group. No one said they liked to select anything from the preference button on the mobile device. One participant said “it was too many steps to be done to change their preference setting”. One Participant said there were at least three steps to be taken for changing the audio interface. Participants without mobile application usage experience said they would not know they could change the preferences without a guide.

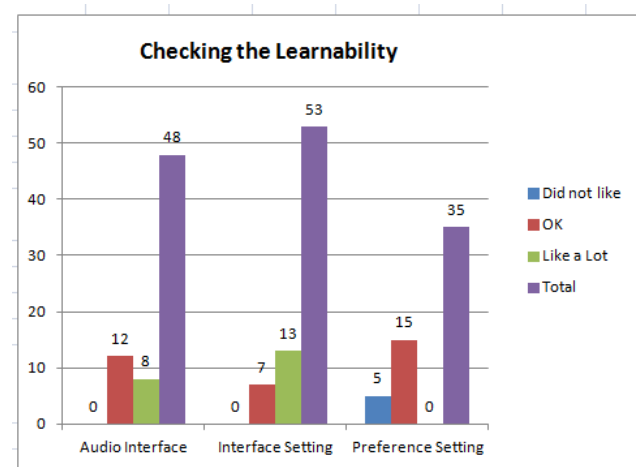


Figure 7.11: Checking the Learnability

7.4.2.4 Checking the Helpfulness

Figure 7.12 shows the results in terms of Helpfulness. The number of participants of each choice are shown in its related column, and total points were calculated based on its rating score. This part evaluated the helpfulness of the task. It estimated three items: location, chapter number, and travel path. The results show positive feedback in terms of helpfulness of the task. More than half the participants like the design of location, chapter number and travel path. None of participant choose “Did not Like” out of the three items. A few participants said they really appreciate the location triggered system, the presentation of book order (which used chapter number), and that the travel paths are simple and visible. One participant suggested that the travel path should stand out more on the map.

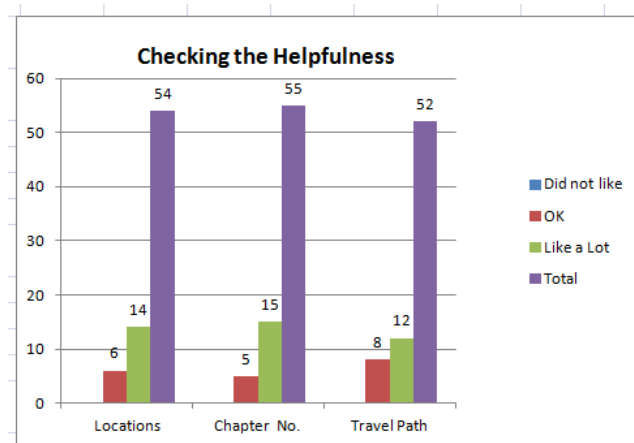


Figure 7.12: Checking the Helpfulness

7.4.2.5 Checking the Satisfaction

Figure 7.14 shows the estimate for checking the satisfaction of the task. It contains one item and 13 statements. Due to time consumption only four participants got through the 13 statements. The rest of the participants looked through and answered one or two of them. The figure only records the answers of the item “experience of all”. The number of participants of each choice are shown in its related column and total points were calculated based on its rating score.

Nineteen of the 20 participants were satisfied with this application in terms of satisfaction of “experience of all”. One second language speaker said he did not like the application because it is mainly using the English and either reading or listening is challenge for him while travelling.

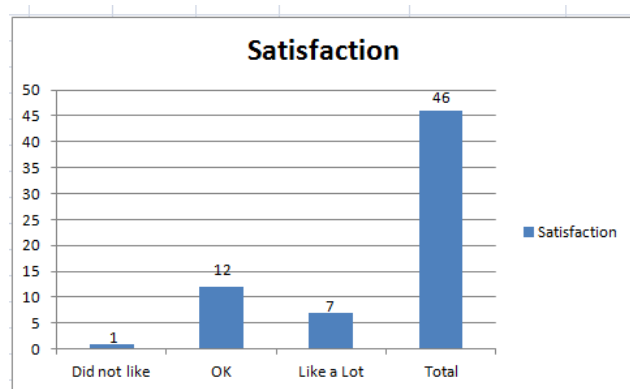


Figure 7.13: Checking the Satisfaction

7.4.3 Strength, Weakness, Opportunities and Threats

This part contains four main questions asking for the strengths, weaknesses, opportunities and threats. Figure 7.14 shows the SWOT analysis of the TER system.

7.5 Recommendations

TER tested in Auckland city in two days. The evaluation study has gained useful feedbacks discussed above. Also, over half participants provided their suggestion for future development.

To sum up, the TER system is successful in terms of usability and we believe that it should continue in the future. Below are listed the suggested changes or expansions to the TER system which would make the software more suitable and reliable while travelling.

- Add instruction or guide about how to run the app;

- Allows taking and viewing multiple images;
- Change the preference setting interface design and make it straightforward;
- Provide more book information to guide user how to choose books, such as type, author, published date.

In addition to the specific area evaluation, participants provided some suggestions for further work. Figure 6.15 lists a few recommendations made during the interview.

TER SYSTEM SWOT EVALUATION REPORT

EVALUATION CRITERIA	Strength	Weakness	Opportunities	Threats
Efficiency: Map, Speed, Text, Map Marker	<ul style="list-style-type: none"> Location triggered system is a good idea and it is interesting. Present all information in a visualized way. Text font is readable. Map Marker is easy to touch on. Speed is fast. 	Sometimes the Map Marker is disappear when zoom in or out few times.	Using mobile to provide travel information is a interesting idea.	If loaded from server, be aware of the amount of data transferred and cost.
Helpfulness: Location, Book Order, Travel Path	<ul style="list-style-type: none"> The current location is marked in the map and this enable traveler to identify where they are and how far to go the next place. Book order is clear, Travel path can help traveler to walk towards the target. 	In one book the color of travel path is too closed to the map.	It presents book, chapter, travel path and map together.	It is not suitable for long story to be presented on the small screen.
Controllability: Listening/Reading, Book Selection, Picture Taking	<ul style="list-style-type: none"> All these features work fine. It is nice for user to taking pictures to record their travelling. It is good for people to have choice between listening and reading. Also traveler can read in their own time. 	<ul style="list-style-type: none"> The reminder system sometimes interrupts travelling. A traveler does not have clue to choose books due to lack of book information. Images are not bigger enough. Multiple images are not available. 	Travelling with listening is convenient.	It is not comfortable to listen with the ear phone.
Learnability: Audio Interface, Main Screen Interface Setting, Preference Setting	<ul style="list-style-type: none"> Main screen is meaningful. Audio interface is easy to follow. 	<ul style="list-style-type: none"> The application takes a few steps to change preference. It is not straightforward. No instructions are given. 	Can dynamically generate sound.	<ul style="list-style-type: none"> The quality of sound sometimes sounds funny. The sound can come out in a quite area and interrupt others.

Figure 7.14: SWOT Evaluation Report

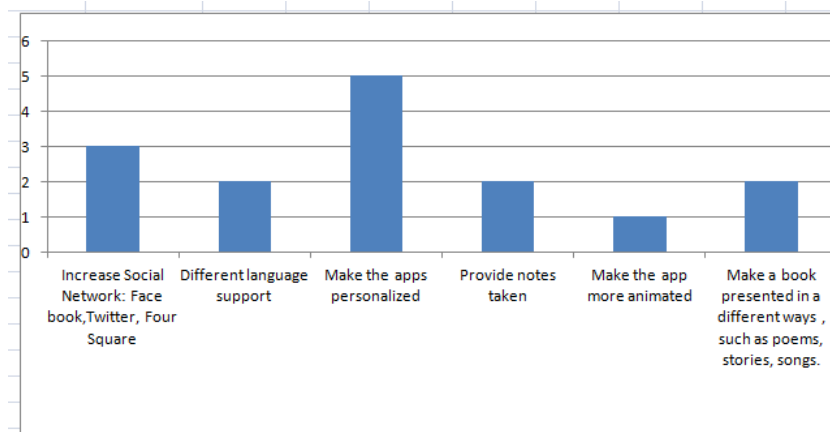


Figure 7.15: Recommendations for the TER System

Chapter 8

Conclusions

This thesis introduced a location based travel e-book reader system (TER system), and empirically evaluated the mobile software.

8.1 Summary

This thesis started from the introduction of this project and identified the five requirements of this project in Chapter 1. Chapter 2 described five related works previously done. We described a visual scenario to present their usage in the real world for each related work, and introduced the ideas underlying the construction of these works. From a practical point of view, the travel information provider system relies on new technology, such as a digital map and location detection. From an information retrieve system, people prefer to have controllability of the software, such as choices between listening to the audio while travelling and the ability to select a book of interest to guide their travels. These features have been partially introduced in all these five previous works, but none of them combined the five requirements together which was described in Chapter 1.

Based on the requirements in Chapter 1 and the research work described in Chapter 2, we started to think about our travel e-book reader system in Chapter 3. In the design phase we discussed both the ways of presenting information

and the users experience with small screen devices. Chapter 4 introduced the ideas and experiences while implementing the software, and explained the benefits of these experiences. To present the software, Chapter 5 provided running time screen shots. The experimental results for this mobile system produced in this thesis were presented in Chapter 6 and Chapter 7.

8.2 Main Contributions

Our experimental results show that travel e-book reader (TER) systems are very welcome while travelling. The location triggered audio book function provided a special experience while travelling. This is found in both events introduced in Chapter 6 and Chapter 7. The idea can be applied to travel in any new place, or to a special event held in a familiar place.

This thesis makes several contributions to research on mobile applications and travel applications. These contributions are:

- the use of mobile system and digital library. A single mobile device can retrieve data from a large collection.
- the use of the text to speech function on mobile devices. Text-to-speech function is not as perfect as human speech, but it can be improved.
- the use of a small screen to present large amounts of information. A reduced size mobile device can present large data in a special way.
- the use of mobile hardware to enrich the application function. A mobile device contains existing hardware, such as a camera, and this can be embedded into the mobile application to maximise the usage of mobile devices.
- the method of evaluating the products. The software has been used in two events which is introduced in Chapter 6 and Chapter 7. The software

was tested in these events and was found to provide a special and real experience of the products.

- the experience of collecting data. To collect GPS data is not simple, such as drag and drop the location using online tools.
- the experience of install software in the public.

8.3 Future Work

Some topics for future research wereraised from the issues addressed in this thesis. One question is the usage of internet; how much data people need to download, and at what cost. People questioned how much they can control in terms of speed and cost. This is a necessary and serious question which should be considered in future work.

Another question raised was around social networking. How can people share their travel experiences? People are keen to exchange their experience via the social network.

Making interaction with the mobile device more comfortable is another avenue for future research. We have applied options for the user to set their preferences. Participants complained that the procedure is not simple. Mobile devices provide multiple screens and navigate in different ways; how can it made simple and consistent? With the exception of common usage of multiple screens, a dialogue and list of menus, can we find a special way, such as the use of an audio file, to deliver the information?

This thesis introduces the ideas and products of a travel mobile application. There are many other similar travel applications. Some of them have been described in Chapter 2. The experimental results presented in this thesis show that, by using text-to-speech function and delivering information by sound and combining lots of information with the map, it is possible to provide rich travel

information while travelling.

Bibliography

Android. Handling runtime changes, 2012.

Appendix A

Bibliography

Bibliography

- Android. (2012). *Developers*, Retrieved 2011, from <http://developer.android.com/guide/topics/resources/runtime-changes.html>
- Apache. (2012). *Apache XML*, Retrieved 2011, from <http://xerces.apache.org/xerces-j/>
- Auckland. (2005). *Auckland City*, Retrieved 2012, from <http://web.archive.org/web/20050211084114/http://www.aucklandcity.govt.nz/whatson/events/livingroom/default.asp#Poetry%20on%20the%20Pavement>
- Backx, P. (2009). *Android Tutorial: Creating and Using an SD Card in the Emulator*, Retrieved 2011, from <http://www.streamhead.com/android-tutorial-sd-card/>
- Bevan, N. (1994). Usability measurement in context. *Behaviour and Information Technology*, 13, 132-145.
- Bevan, N. (1995.). Usability is Quality of Use. *Paper presented at the the 6th International Conference on Human Computer Interaction*, Yokohama. Retrieved from <http://www.usabilitynet.org/papers/usabis95.pdf>
- David, M. (2010). *SAX & DOM parsers*, Retrieved 2011, from <http://www.cs.nmsu.edu/~epontell/courses/XML/material/xmlparsers.html>
- Eclipse. (2012). *EclipseClassic 4.2 [Software]*. Eclipse Downloads Retrieved 5, Retrieved 2011, from <http://www.eclipse.org/downloads/>
- Franklin, S. (2010). *XML Parsers: DOM and SAX Put to the Test*, Retrieved 2011, from <http://www.devx.com/xml/Article/16922/1954>
- Gao, X. (2007). *A Service for audio icon and audio books in the mobile tourist information system (TIP) via the greenstone digital library*. Master's thesis, the University of Waikato, Hamilton, New Zealand. Retrieved from <http://researchcommons.waikato.ac.nz/handle/10289/2246>
- Greenstone. (2011). *Greenstone3*. Greenstone digital library software, Retrieved 2011, from <http://www.greenstone.org/developers>
- Hamilton. (2011). *Hamilton Garden*. Retrieved Sep, 2011, from <http://hamiltongardens.co.nz/>
- Harker, S. (1995). The development of ergonomics standards for software. [Applier Ergonomics]. *Applier Ergonomics*, Vol 26, (4), pp. 275-279.
- Hinze, A., Agnès, V., & George, B. (2009). Tip: Personalizing Information Delivery in a Tourist Information System *Information Texhnology & Tourism*, 11(3), pp.247-264.
- Hinze, A., & Bainbridge, D. (2011). Making your digital library mobile with location-triggered audio books. *Proc Workshop on Books, ebooks and Digital Documents*, 1-4.

Hinze, A., & Buchanan, G. (2006). The challenge of creating cooperating mobile services: experiences and lessons learned. *Paper presented at the the 29th Australasian Computer Science Conference, Australia*. Retrieved from <http://dl.acm.org/citation.cfm?id=1151723>

Hinze, A., Michel, Y., & Eschner, L. (2009). *Event-based communication for location-based service collaboration*. Retrieved from <http://crpit.com/confpapers/CRPITV92Hinze.pdf>

Hinze, A., & Osborn, W. (2007). *Location-based indexing for mobile context-aware access to a digital library*. Retrieved from <http://www.cs.waikato.ac.nz/pubs/wp/2007/uow-cs-wp-2007-04.pdf>

Hinze, A., & Voisard, A. (2003). *Location- and Time-Based Information Delivery in Tourism*, from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.3.5869>

IQapps. (2012). *About the Trip Journal Retrieved*. August, 2012, from <http://www.trip-journal.com/>

ITBusinessEdge. (2012). *Kindle Fire*. Retrieved June, 2012, from http://www.webopedia.com/TERM/K/kindle_fire.html

Jones, S., Jones, M., Barr, M., & Keegan, T. T. (2005). Searching and browsing in a digital library of historical maps and newspapers. *Journal of Digital Information*, Vol 6(2). Retrieved from <http://journals.tdl.org/jodi/article/view/68/71>

Katherine J. Don, Bainbridge, D., & Witten, I. H. (2002). *The design of Greenstone 3: An agent based dynamic digital library*. Retrieved from <http://www.greenstone.org/manuals/gs3design.pdf>

Lars, V. (2011, 2012,jan). *Android Threads, Handlers and AsyncTask*. Retrieved January, 2012, from <http://www.vogella.com/articles/AndroidPerformance/article.html>

MapFileWriterOsmosis. (2011). *MapFileWriterOsmosis*. Mapsforge, Retrieved 2011, from <http://code.google.com/p/mapsforge/source/browse/wiki/MapFileWriterOsmosis.wiki?r=987>

mapsforge. (2011a). *MapView.java*. Android Open Source, Retrieved 2011, from <http://www.java2s.com/Open-Source/Android/App/mapsforge/org/mapsforge/android/maps/MapView.java.htm>

mapsforge. (2011b). *free mapping and navigation tools*, Retrieved 2011, from <http://code.google.com/p/mapsforge/downloads/list?q=label:Featured>

Mu, T. (2012). *SpecificationBinaryMapFile*. Retrieved 2012, from <http://code.google.com/u/118427153554151536405/>

Nielsen. (April 2012). *The digital revolution: A look through the marketer's lens*. Retrieved May, 2012, from <http://na.ad-tech.com/sf/wp-content/uploads/DigitalConsumer.pdf>

openstreetmap. (2011). *openstreetmap*. The Free Wiki World Map, Retrieved 2011, from <http://www.openstreetmap.org/>

Oracle. (2011). *Java*. Retrieved October, 2011, from <http://www.oracle.com/technetwork/java/index.html>

Osmosis. (2012). *Osmosis*. Osmosis. Retrieved December, 2012, from <http://wiki.openstreetmap.org/wiki/Osmosis>

Project New Zealand Digital Library. (2012a). *greenstone3*, Retrieved 2011, from <http://www.greenstone.org/greenstone3-home>

Project New Zealand Digital Library. (2012b). *Tutorial exercises*, Retrieved 2011, from http://wiki.greenstone.org/wiki/index.php/Tutorial_exercises#Building_a_small_collection_of_HTML_files

Satya, K., Dave, M., & Sayed, H. Y. (2011). *Pro Android 2*. New York, US: Paul Manning.

Seffah, A., Donyaee, M., ., R. K., & Padda, H. K. (2006). Usability measurement and metrics: A consolidated model. *Software Qual Journal* (2006), 14, 159–178. doi: DOI 10.1007/s11219-006-7600-8

Segan, S. (March 10,2012). *The Top 10 Free Travel Apps*. Retrieved May, 2011, from <http://www.pcmag.com/article2/0,2817,2361139,00.asp>

SnapHop. (2012). *Mobile Marketing Statistics*. 2012 Retrieved June, 2012, from <http://snaphop.com/2012-mobile-marketing-statistics/>

theStations. (2012). *the Stations*. Retrieved 2012, from <http://www.stations.org.nz/start.php>

Thilo. (2012). *Overlay API*. Retrieved Sep, 2011, from <http://code.google.com/p/mapsforge/wiki/OverlayAPI>

UsabilityNet. (2012). Usability Net. Retrieved May, 2012, from http://www.usabilitynet.org/tools/r_international.htm

Wikipedia. (2012a). *E-book reader*. Retrieved e-book reader, Retrieved 2011, from http://en.wikipedia.org/wiki/E-book_reader

Wikipedia. (2012b). *OpenStreetMap*. Retrieved January, 2012, from <http://en.wikipedia.org/wiki/OpenStreetMap>

Witten, I., Bridge, D., & Nichols, D. (2010). *How to Build a Digital Library 2nd (Ed.)* . Retrieved from www.greenstone.org/howto

Appendix B

Ethics Consent Approval for Paper Study

**Application for approval under the
Ethical Conduct in Human Research
and Related Activities Regulations**



Ethics Committee, Faculty of Computing and Mathematical Sciences

Name: Casey Peng

Department: Computer Science

Email address: shanhong1005@126.com

Contact: 07 8384466 Ext 6017

This is an application for approval of: (please tick as many as apply)

- Research project involving human participants
- Course/Paper which involves student projects that collect data from human participants
- Undergraduate student project which involves data collection from human participants
- Master's degree research
- PhD research
- PhD research proposal to move from Conditional to Full enrolment

Supervisor's name: (if applicable) :Annika Hinze

Supervisor's approval (signature)

Project Title:

Location-based Audio Book Reader with Digital library

Is this research associated with an external grant or funding? Yes No

- I request approval for this research or related activity and attach all relevant documentation necessary for evaluation under the Ethical Conduct in Human Research and Related Activities Regulations.
- I am applying under section 10 of the Ethical Conduct in Human Research and Related Activities Regulations Large Random Sample Surveys and Similar Research Methodology and consequently there is no consent form in this application.
- I am applying under section 11 of the Ethical Conduct in Human Research and Related Activities Regulations Field Research and consequently there is no consent form in this application.
- I have read and complied with the Ethical Conduct in Human Research and Related Activities Regulations.

Principal Investigator's signature :

Ethics Committee Action

Should this application be referred to another delegated University Ethics Committee?

Yes No

Details: _____

Does this application also require approval from an external body (e.g. Northern Y Regional Ethics Committee)? See Appendix 5.

Yes No

Details: _____

- Approved Convenor's signature
- Approved with recommendations*
- Request modifications* *Reviewer's signature*
- Approved with modifications* *Reviewer's signature*
- Forward to University committee Date: _____
- Copy of approval letter to UNILink for research associated with external grants and contracts

Appendix C

Participant Information Sheet for Paper Study

Participant Information Sheet



Ethics Committee, Faculty of Computing and Mathematical Sciences

Project Title

Location-based Audio Book Reader with Digital library

Purpose

This study is part of a master research project conducted at the University of Waikato.

What is this research project about?

The project investigates the use of audio books while travelling.

What will you have to do and how long will it take?

Participants will go to the Waikato and Auckland region and use an audio guide on a mobile phone. Afterwards the researcher wants to conduct an interview during which a questionnaire will be filled. This should take no longer than 15 min. You will be asked to give consent prior to the interview.

What will happen to the information collected?

The information collected will be used by the researcher to write a research report and scientific publications. It is possible that articles and presentations may be the outcome of the research. Only the researchers will be privy to the notes taken. All data will be anonymised. No notes will be kept that allow the researcher to connect the notes taken with the participants. No participants will be named in the publications and every effort will be made to disguise their identity. If you wish to receive a summary of the study outcomes, your contact details will be taken but kept separate from the study data and only be used to send the summary. The contact data will be destroyed by the end of the study analysis.

Declaration to participants

If you take part in the study, you have the right to:

- Refuse to use the audio guide on their travel at any time and to withdraw from the study..
- Refuse to answer any particular question, and to withdraw from the study during the interview.
- Ask any further questions about the study that occurs to you during your participation.
- Be given access to a summary of findings from the study when it is concluded.

Who's responsible?

If you have any questions or concerns about the project, either now or in the future, please feel free to contact either:

Researcher:

Casey Peng
Computer Science
Shanhong1005@gmail.com

Supervisor:

Annika Hinze
Computer Science
hinze@cs.waikato.ac.nz
07 838 4052
Department of Computer Science
The University of Waikato
Private Bag 3105
Hamilton 3240
NEW ZEALAND

Research Consent Form



Ethics Committee, Faculty of Computing and Mathematical Sciences

Location-triggered E-Book Reader with Digital library

Consent Form for Participants

I have read the **Participant Information Sheet** for this study and have had the details of the study explained to me. My questions about the study have been answered to my satisfaction, and I understand that I may ask further questions at any time.

I also understand that I am free to withdraw from the study while using the electronic guide on the Android phone and during the interview or to decline to answer any particular questions in the study. I understand I can withdraw any information I have provided up until the researcher has commenced analysis on my data. I agree to provide information to the researchers under the conditions of confidentiality set out on the **Participant Information Sheet**.

I agree to participate in this study under the conditions set out in the **Participant Information Sheet**.

Signed: _____

Name: _____

Date: _____

I am interested in receiving a summary of the study outcome.

Contact details: _____

Researcher:

Casey Peng
Computer Science
Shanhong1005@gmail.com

Supervisor:

Annika Hinze
Computer Science
hinze@cs.waikato.ac.nz
07 838 4052
Department of Computer Science
The University of Waikato
Private Bag 3105
Hamilton 3240
NEW ZEALAND

Appendix D

Application for Approval

Outline of Research Study

**Application for Approval
Outline of Research or Related Activity
Ethics Committee, Faculty of Computing and Mathematical Sciences**



Note: add your project details to this document – do not delete any of the existing content

Details of Proposed Activity

1. Identify the project

Title of Project

Location-based Audio Book Reader with Digital library

1.1 Researcher(s) name and contact information

Researcher:

**Casey Peng
Computer Science
Shanhong1005@gmail.com**

1.2 Supervisor's name and contact information (if relevant)

Supervisor:

**Annika Hinze
Computer Science
hinze@cs.waikato.ac.nz
07 838 4052
Department of Computer Science
The University of Waikato
Private Bag 3105
Hamilton 3240
NEW ZEAL**

1.3 Anticipated date to begin data collection

28 May 2012

1.4 Does your application involve issues of health or disability with human participants? If so, please refer to the guidelines as to whether your application needs to be submitted to the Northern Y Regional Ethics Committee.

The application does not involve issues of health or disability with human participants.

2. Describe the research or related activity

2.1 Briefly outline what the project is about including your goals and anticipated benefits. Include links with a research programme, if relevant.

The study explores the participants' use of audio books, digital library, electronic guides and books during travels. The participants will be asked to use an electronic guides on the Computer Science Department's Android phone for the location in the Waikato region. Follow-up questions will analyse usability and affordance of the system, and explore in detail the reasoning for the usage during the study. The follow-up interview is done at the university.

2.2 Briefly outline your methods.

The study will be performed as a field study with participants using an electronic guide on an Android phone supplied by Computer Science Department in the Waikato or Auckland region, followed by semi-structured interviews. The exact location will depend on each participant's travel.

2.3 Describe plans to give participants information about the goals of the research or related activity.

Participants will receive an information sheet about the study. They will also have the opportunity to receive information about the outcome of the study in summarized form.

2.4 Identify the expected outputs of this research or related activity (e.g., reports, publications, presentations).

The researchers plan to publish the results in working papers, academic conferences and journals. The results will be presented at talks at these conferences or during visits to other universities. The results may also be used for collaboration with local tourist organizations and applications to funding bodies. The results of the research may be made available in summarized form to potential funding bodies. All publication and presentation of the results will be done in anonymised form.

2.5 Identify who is likely to see or hear reports or presentations arising from this research or related activity.

The likely viewer/reader/listener will include supervisor, lecturers of the Department of Computer Science, national and international researchers, tourism agencies and funding bodies.

2.6 Identify the physical location(s) for the research or related activity, the group or community to which your potential participants belong, and any private data or documents you will seek to access. Describe how you have access to the site, participants and data/documents. Identify how you obtain(ed) permission from relevant authorities/gatekeepers if appropriate and any conditions associated with access.

The participants will take the phone with the installed software to the Waikato and Auckland region. They will participate in a follow-up interview either at university or the travelled place. Participants will be student from the university, people visiting Waikato or Auckland region. Participants will be approached in public places and at the university. We will not access documents.

3. Obtain participants' informed consent without coercion

3.1 Describe how you will select participants (e.g., special criteria or characteristics) and how many will be involved.

People have to be willing to use the provided Android phone with the electronic guide. The study is planned to have 10 to 20 participants.

3.2 State clearly whether this is an application under section 10 of the Ethical Conduct in Human Research and Related Activities Regulations: Large Random Sample Surveys.

This is not an application under section 10 of the Ethical Conduct in Human Research and Related Activities Regulations.

3.3 Describe how you will invite them to participate.

Participant will be approached directly by the researcher (in public places and at university) and/or be contacted via email (at the university).

- 3.4 Show how you provide prospective participants with all information relevant to their decision to participate. Attach your participant information sheet, cover letter, or introduction script. See document on informed consent for recommended content. Information should include, but is not limited to:
- what you will ask them to do;
 - how to refuse to answer any particular question, or withdraw any information they have provided at any time before completion of data collection;
 - how and when to ask any further questions about the study or get more information.
 - the form in which the findings will be disseminated and how participants can access a summary of the findings from the study when it is concluded.

Participants will receive an information sheet and a consent form (as attached) before the study commences. The study will also be explained to them by the researchers.

- 3.5 Describe how you get their consent. (Attach a consent form if you use one).

Participants will be asked to sign the attached consent form.

- 3.6 Explain incentives and/or compulsion for participants to be involved in this study, including monetary payment, prizes, goods, services, or favours, either directly or indirectly.

No incentives will be used.

4. Minimise deception

If your research or related activity involves deception – this includes incomplete information to participants -- explain the rationale. Describe how and when you will provide full information or reveal the complete truth about the research or related activity including reasons for the deception.

The research does not involve deception.

5. Respect privacy and confidentiality

- 5.1 Explain how any publications and/or reports will have the participants' consent.

Publications and reports as described in section 2.4 will not contain any personal data of the participants. No further consent will be required beyond the signed consent sheet.

- 5.2 Explain how you will protect participants' identities (or why you will not).

Only the researchers will know participant's details. All data will be anonymised. If participants wish to receive a summary of the study outcomes, they will need to give contact details. However, these details will be separated from the notes. None of the contact details will be used in any publications; contact details will be destroyed by the end of study analysis.

- 5.3 Describe who will have access to the information/data collected from participants. Explain how you will protect or secure confidential information.

Data collected will be kept as paper records and as electronic summaries. After the summaries have been sent out, no confidential information is kept by the researchers. The collected data is anonymous and therefore does not need specific protection. All data will be kept for long term-archiving in the FCMS data archive.

6. Minimise harm to participants

'Harm' includes pain, stress, emotional distress, fatigue, embarrassment and exploitation.

6.1 Where participants risk change from participating in this research or related activity compared to their daily lives, identify that risk and explain how your procedures minimize the consequences.

There is no risk of harm for the participants.

6.2 Describe any way you are associated with participants that might influence the ethical appropriateness of you conducting this research or related activity – either favourably (e.g., same language or culture) or unfavourably (e.g., dependent relationships such as employer/employee, supervisor/worker, lecturer/student). As appropriate, describe the steps you will take to protect the participants.

If a dependent relationships exist between participants and one of the researchers, the other researcher will conduct the interview.

6.3 Describe any possible conflicts of interest and explain how you will protect participants' interests and maintain your objectivity.

There is no conflict of interest.

7. Exercise social and cultural sensitivity

7.1 Identify any areas in your research or related activity that are potentially sensitive, especially from participants' perspectives. Explain what you do to ensure your research or related activity procedures are sensitive (unlikely to be insensitive). Demonstrate familiarity with the culture as appropriate.

No areas of the research are identified as being particularly sensitive to the participants. However, participants will be able to refuse answering questions as laid out in the information sheet.

7.2 If the participants as a group differ from the researcher in ways relevant to the research or related activity, describe your procedures to ensure the research or related activity is culturally safe and non offensive for the participants.

The research is not offensive or culturally unsafe.

Appendix E

Location based audio book
reader questionnaire



Location-based audio book reader with Digital Library

1. Please indicate your gender :

Female: Male :

2. Please indicate your age group:

Younger than 20:

20 – 35:

36-50:

51-65:

66 or over:

3. Use of books and audio books during travels

	<i>Never</i>	<i>Rarely</i>	<i>Sometimes</i>	<i>Often</i>	<i>Always</i>
Reading/books	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Audio books	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Audio guides in museums	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Guide books while travelling	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Novels while travelling (ref to place?)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Digital Maps	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Travel Application	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

APPENDIX E. LOCATION BASED AUDIO BOOK READER QUESTIONNAIRE

Anything else they used

4. Usability and affordance of the Location-based E-Book Reader

- Provide feedback about the following issues

	<i>Did not like</i>	<i>OK</i>	<i>Like a lot</i>	<i>Feedback</i>
--	---------------------	-----------	-------------------	-----------------

Map	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
-----	--------------------------	--------------------------	--------------------------	--

Listening or Reading	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
----------------------	--------------------------	--------------------------	--------------------------	--

locations	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
-----------	--------------------------	--------------------------	--------------------------	--

Images	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
--------	--------------------------	--------------------------	--------------------------	--

Book selection	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
----------------	--------------------------	--------------------------	--------------------------	--

Audio interface	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
-----------------	--------------------------	--------------------------	--------------------------	--

Chapter number	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
----------------	--------------------------	--------------------------	--------------------------	--

Travel Path	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
-------------	--------------------------	--------------------------	--------------------------	--

Speed Issues	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
--------------	--------------------------	--------------------------	--------------------------	--

Text Format	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Interface design	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Camera issues	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Map Marker	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Preference setting	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Experience of all	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

5. Usage analysis of the Location-based E-Book Reader

Part 1: User Satisfaction Survey

On a scale of 1 to 5 where 1 is strongly disagree and 5 is strongly agree, How much do you agree with the following statements?

	Strongly Disagree					Strongly Agree
It was easy to start the program	1	2	3	4	5	
The screen layout is clear	1	2	3	4	5	
The Text are Readable	1	2	3	4	5	
The picture size are suitable	1	2	3	4	5	
The generated sound is clear	1	2	3	4	5	
Audio Sequences are of an adequate quality	1	2	3	4	5	
The book selection works	1	2	3	4	5	
The user path displays clearly and nicely	1	2	3	4	5	
Play/Pause, Show/Hide Text, Skip work fine	1	2	3	4	5	
Text/image comes up when touching the balloon	1	2	3	4	5	
Pictures can be taken and viewed	1	2	3	4	5	

APPENDIX E. LOCATION BASED AUDIO BOOK READER QUESTIONNAIRE

The image/text load fast	1	2	3	4	5
User have good experience of all	1	2	3	4	5

Part 2: Please give short answer to these questions which designed for evaluating software packages that are intended to be used.

- Would you like to use the software, e.g. using this application for travelling or event? Why /why not?

- One thing that was good

- One thing to change

- Other comments

Appendix F

The Poetry event Introduction for the Case Study

Introduce Poetry on the Pavement Event which advertised in Auckland City Council website.

Poetry on the Pavement

Aucklanders should keep their eyes focused on the footpaths as they walk through the CBD streets. Popping up underfoot will be a selection of poems which together tell a unique story of Auckland. The journey of poems will stretch from K Road down to the waterfront. They are all composed by New Zealand poets and reflect on aspects of Auckland places, historic events or personal memories. The following table shows the location, poets and poems of this special journey(Auckland City Council Website,2012).

Android Audio Book for the Poetry on the Pavement

At the University of Waikato we have developed an Android application showing a map of the Auckland City and containing audio files, text and images about the journey of poems. Walking through the CBD Street you will hear a chirp-sound indicating you can listen to the audio, read the poems, take pictures for the location, and switch to other books at that location.

Try it. Enjoy it...

You are invited to tryout this application and welcome to provide any feedback.

Location	Poet	Poem
Corner K Rd and Queen Street	David Eggleton	Karangahape Road Celebrates
Outside front entrance to the Town Hall (the poem leading to doorway like a band of red carpet)	Kevin Ireland	Paderewski's Hands
In Aotea Square leading up the fountain	Robert Sullivan	Te ao marama I
Painted on footpath leading to Cinema Complex on Queen Street (and next to Borders)	Jill Chan	The Conversation 2
Leading up Wellesley Street alongside the CMC	Trile Te Arama Menzies	Heaven was In Our Eyes
Curving around corner of Wellesley and Lower Lorne Streets towards Public Library	Janet Charman	End of the dry
Outside Parsons Bookshop/Gow Langford Gallery display window leading towards Auckland City Art Gallery	Rienke Ensing	Here I give thanks to Mondrian and McCahon
Near entrance to John Leech Gallery in Kitchener Street	John Pule	Loto (extract)
Corner of Kitchener Street and Victoria Street near Milford Gallery	Paula Green	Intersection
Outside music store, Beat Merchants, in Victoria Street	Hone Tuwhare	LP Blues
Next to Paneton Café and entrance to Victoria Street Car Park in High Street	Michael Morrissey	Advice on Auckland (extract)
On High Street near entrance to Little High Street and World shop	Jenny Bornholdt	Extract from book entitled Air Pocket
In Durham Lane near Elbow Bar	Arthur Baysting	Auckland, April
In Freyberg Place alongside fountain	Konal Helu Thaman	Weekend in Auckland
Outside Unity Books in High Street	C. K. Stead	Auckland (from Straw into Gold)
At entrance to Upper Vulcan Lane on High Street	Murray Edmond	From book entitled Piece of Work
At entrance to Lower Vulcan Lane	Anne Kennedy	First and last times
At Intersection on Queen Street and Shortland Street	Anna Jackson	The Madonna of the Traffic Lights
Outside Britomart entrance and old CPO	Sue Fitchett	I'm Writing you a Letter Mr Kipling
On Quay Street and the bottom lower Albert Street in front of two benches with a view of the harbour	Michele Leggott	From sequence entitled Circle
At Intersection of Fanshawe Street and Albert Street outside Ronnies Bakery	Bob Orr	Once I Felt the Summer Breeze
Outside Wah Lee's shop in Hobson Street	Denys Trussell	A Poem in the Shop of Wah Lee

(Auckland City Council Website,2012)

Appendix G

User Study Data Collection

Appendix H

The Stations software introduction

Audio for Stations of the Cross

At the University of Waikato we developed an application showing a map of the Hamilton Gardens and containing audio files and texts about all stations, e.g., scriptures and reflections on the shown art. Reaching a station you will hear a chirp-sound indicating that you can listen to the audio or read the text.



There are two options for you to participate.

- You have an Android phone: we invite you to try out this app.
- You have another phone: we can transfer the audio files onto your phone and you can play them when approaching a station.

Please give us feedback afterwards (via our little feedback flyer).

In Short

Scriptures and reflections as audio files are available as

- App if you own an Android phone
- MP3 files (if your phone has a mini-USB plug)

Contact

Dr. Annika Hinze
University of Waikato
Department of Computer Science
hinze@cs.waikato.ac.nz



Web

Shortened URL: <http://goo.gl/KoMOa>

