

Software Engineering Group Project Project Plan

Author: Christopher Edwards; Douglas Gardner; Luke Horwood; Jostein Kristiansen; Ben Rainbow; Ashley Smith; James Woodside

Config Ref: SE_02_PL_01

Date: 2013-11-07

Version: 0.3

Status: Draft

Department of Computer Science
Aberystwyth University
Aberystwyth
Ceredigion
SY23 3DB
Copyright © Contributors, 2013

CONTENTS

CONTENTS	2
TABLE OF FIGURES.....	3
1. INTRODUCTION	4
2. PURPOSE OF THIS DOCUMENT	4
2.1 Scope.....	4
2.2 Objectives.....	4
3. OVERVIEW	5
3.1 Technologies	5
3.1.1 Client	5
3.1.2 Android	5
3.1.3 Server.....	5
3.1.3.1 PHP	5
3.1.3.2 MySQL	5
3.1.3.3 Apache	5
3.1.3.4 Linux	5
3.2 Architecture.....	5
3.2.1 Client	5
3.2.2 Map.....	6
3.2.3 Photos	6
3.2.4 Internet Connectivity	6
3.3 Target Users	6
4. USE CASE DIAGRAM	7
5. USER INTERFACE DESIGN	9
5.1 GUI design	9
5.2 Start Screen	9
5.3 Displaying all existing routes Screen	10
5.4 Create new route screen	10
5.5 View map screen	10
5.6 create new point of interest screen	10
5.7 Review submission screen	11
5.8 Cancel upload screen.....	11
5.9 Upload confirmation screen	11
5.10 Help Screen	11
6. GANTT CHART	12
7. RISK ANALYSIS	13
7.1 Constant Risks.....	13
7.2 Risks related to documentation	13
7.3 Risks related to development and delivery.....	14
7.4 Risks related to the usage of the program	14
8. REFERENCES	17
DOCUMENT HISTORY	18

TABLE OF FIGURES

Figure 1: Architecture Diagram	6
Figure 2: Use Case Diagram FR notation specifies that the task is a functional requirement and is needed in order to produce the specified output from the application.	7
Figure 3: Diagram showing the Start Screen	9
Figure 4: The Display All Routes screen.....	10
Figure 5: Create New Route screen	10
Figure 6: Create New PoI screen	10
Figure 7: The View Map screen	10
Figure 8: The Review Route screen.....	10
Figure 9: The Cancel Upload screen.....	11
Figure 10: The Upload Confirmation screen	11
Figure 11: The Help screen(s)	11

1. INTRODUCTION

The below content is an overview of the proposed route planning system developed for the use on mobile devices running the [Android](#) Operating system.

This document covers all the aspects relating to the high level overview and abstract data types required in order to be able to create the final product.

2. PURPOSE OF THIS DOCUMENT

The purpose of this document is to show how we have created a basic set of objectives based on the client's requirements for a walking tour creator application.

The document will show the basic navigation and visual appearance of the applications user interface.

A Gantt chart will be provided, this will show the tasks involved during the process of the group project, including the milestones linked to each task.

Along with this will be a risk analysis, which will show possible problems that may be occur during the development of the system, it will also show how these risks can be avoided.

2.1 Scope

This document will take into account the client's requirements for the group project.

Included in the document will be an overview of the system, detailing the high-level architecture, choice of platform and the target users of the proposed system. Use Case diagrams will be included to show interactions of user's with the proposed system. Designs of the user interface will be included to give an idea of how the application will look and give details on navigation. A Gantt chart will be included to show the task's involved in the systems development and the milestones of the development. Finally a risk analysis will be included to give detail on potential problems we could face during the systems development, and how we may avoid or solve these problems.

2.2 Objectives

The main objectives of this document are to:

1. Provide an overview of the system, giving detail of the target audience and the technologies that will be used
2. Show the main interactions of the user's and the system
3. Show the basic navigation of the system and give an idea of how the finished product will look
4. Give a time line of all tasks that need to be completed
5. Identify any problems that may arise from the production of this system.

3. OVERVIEW

3.1 Technologies

The following technologies (and platform) shall be used in this system:

3.1.1 Client

3.1.2 Android

The client's requirements specifically stated that the application shall be designed for Android phones. The target Android platform will be Android version 4.2 (Jelly Bean (1)) but will be backwards-compatible with Android 2.3 (Gingerbread (1)).

Android is a free operating system for mobile phones (1), in which the applications are written in Java (2). Many members of the development team are literate at Java programming, making Android a platform conducive to creating high-quality applications, such as the one sought by the client.

Android is the leading operating system for mobile phones, with almost 80% market share in the second quarter of 2013, with 187.4 million shipments (3). This makes it appealing over iOS, the operating system used by Apple's iPhones. Another compelling reason to use Android over iOS is that the development of Android applications is free: iOS licensing requires a minimum of a USD \$99 *per annum* fee (4).

3.1.3 Server

3.1.3.1 *PHP*

PHP shall be used on the server to handle communication between the Android device and the database, as well as being used to create the website end-users access to view past walks.

PHP is widely used within server development, with usage on approximately 2.1 million devices (5), and is taught to second-year students (6), giving us an opportunity to gain skills through its application.

PHP has the benefit of being a pre-processor, meaning that PHP does not need its own service. It is also easy to learn, and easy to compose for, allowing for rapid development.

3.1.3.2 *MySQL*

MySQL will be used for our database, to store all the user information from the application.

MySQL is a time-proven application used widely. It is well supported by the other server technologies we will use (in this case, PHP) (7), and is easy to administrate with graphical tools such as phpMyAdmin (8).

3.1.3.3 *Apache*

The Apache HTTP server shall be used in conjunction with PHP to create our web application. It is the most popular HTTP server in use today, with over 50% market penetration (9).

3.1.3.4 *Linux*

The "LAMP" stack is a common server application bundle (10), and will be used to create our server application. As such, we will be developing our server with a GNU/Linux operating system in mind.

3.2 Architecture

The high-level architecture will consist of the following elements:

3.2.1 Client

This describes the Android application. This will be shown in the screens described below in the user interface designs.

3.2.2 Map

The map will be displayed through the use of the Google [Maps API](#), this allows us to have scrolling and zooming readily available for use. The map will show user's the route of the walk, and also allow us to show points of interest.

3.2.3 Photos

Users will be able to add Photo's to points of interest within the application. GPS coordinates will tie a photo to a specific location.

3.2.4 Internet Connectivity

Users will require an internet connection to be able to upload a saved walk to the server. If the user fails to connect or connection is lost then they will be informed with a notification, they will not be able to upload to the server until the regain connection.

3.3 Target Users

As specified in the requirements

specification, the software is to be used by second year computer science students. The design of this

system has taken into account the users knowledge of computer systems. We have tried to make the system as intuitive as we can, with the fewest user actions required.

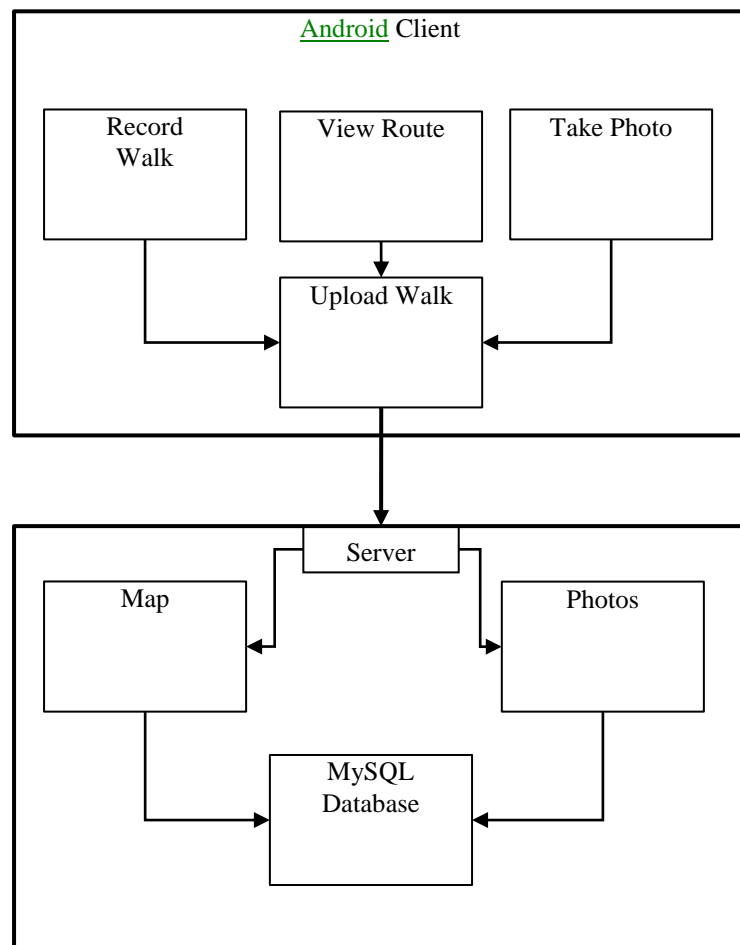


Figure 1: Architecture Diagram

4. USE CASE DIAGRAM

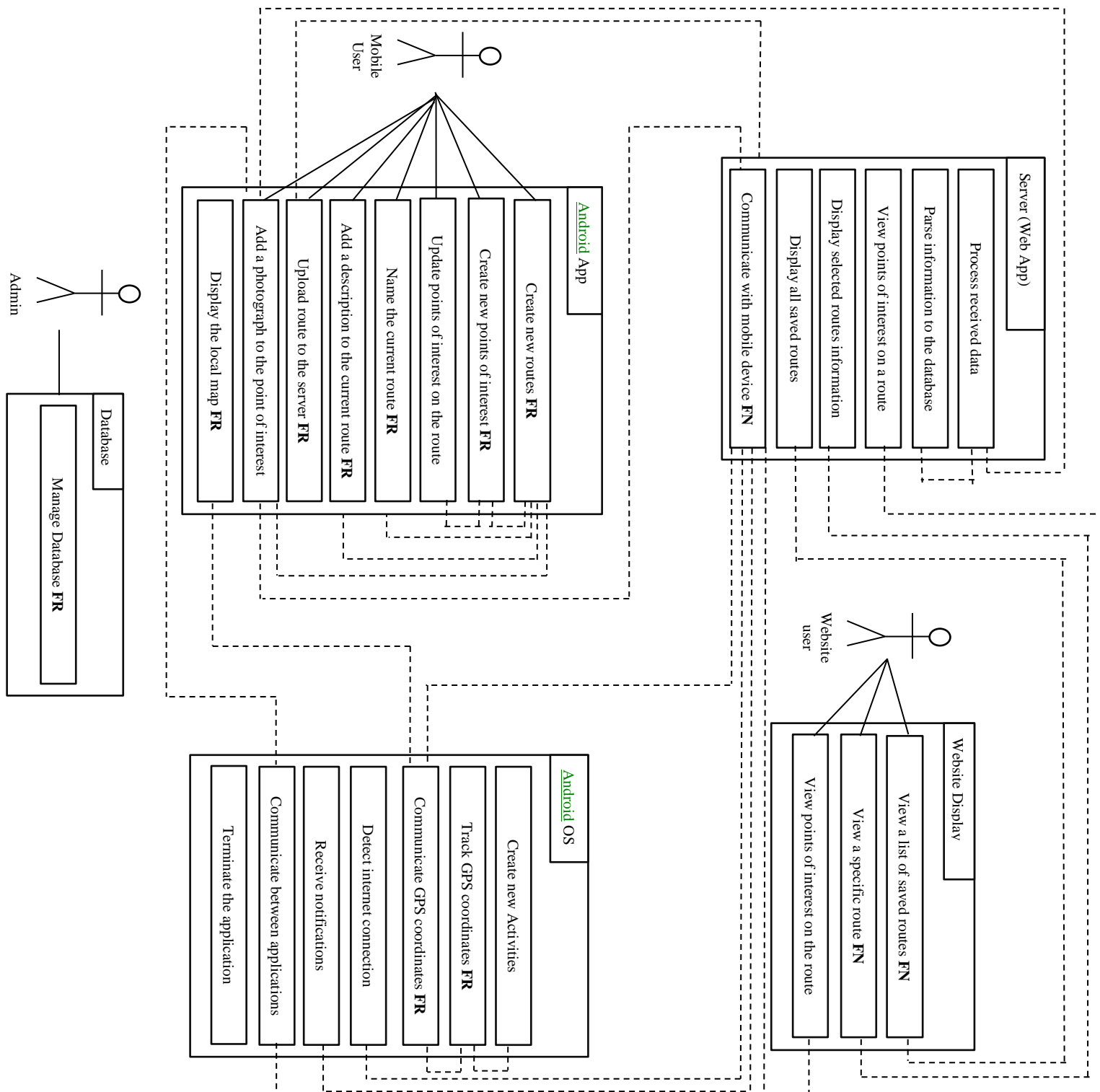


Figure 2: Use Case Diagram

FR notation specifies that the task is a functional requirement and is needed in order to produce the specified output from the application.

Web Application	
Process received data	The server will parse the data received from the mobile device. The server must be able to handle this data efficiently and decide the best course of action.
Parse information to Database	The server must be able to save the information in the correct format within the database.
View points of interest on a route	The server will can access a list of all points of interests for a route. This can then be displayed to the user at a later stage.
Display a routes information	The server can access all the information assigned to a specific route This can then be displayed to the user at a later stage.
Display all saved routes	The server can access the database which stores all the currently saved routes.
Communicate with mobile device	The server must be able to communicate with the Android mobile device. The communication will allow users to upload new routes to the database for display on a created website.

Web User	
View a list of all stored routes	The website will display all routes currently listed in the database to the user.
View the information for a specific list	The website will allow the user to select a specific list. The specific information for the selected list will be returned to the user.
View the points of interest for a specified route	For any selected route the user will be able to view specific points of interest for said route.

Database Administrator	
Manage the database system	The administrator must be able to log into the database administrator facility to manage the database and collected data.

Mobile user	
Create new routes	The user of the Android application will be able to create new routes which will record their GPS coordinates and any desired points of interest.
Create points of interest on a route	The user of the application will be able to add points of interest with their current GPS coordinates and add it to the route.
Update points of interest on a route	The user can modify and or update the points of interest on the route.
Create a name for the route	A name can be specified for the route which can be used to identify the route at a later time period.
Add a description to the route	A description can be added to the route which will be displayed on the website.
Add a photograph to the point of interest	A photograph can be retrieved from the devices camera, which can then be applied to the point of interest as a visual aid.
Upload the completed route to the server	Once the application user is happy with the route they can upload it to the server which will handle the request and store the information in the database.

Android Operating System	
Create new activities	The operating systems needs to creative multiple activities and display these accordingly to the application user. The operating must also handle the user interaction as user interface display.
Track GPS coordinates	The operating system will handle the tracking of the GPS coordinates. This will track the user's current position on their route for use in path drawing and point of interest positioning. The operating system must also terminate the GPS tracking when the user requests.
Communicate GPS coordinates	The GPS coordinates must be communicated through the infrastructure. The mapping system must know about the GPS coordinates in order to display the correct local map and the application itself must know about the coordinates in order to effectively plan the route.
Display the location map	Based upon the GPS coordinates the local area map will be displayed to the user.
Detect internet connection	The Operating System will check for a suitable connection point. This is required to upload information to the server .
Communicate between applications	The Operating System must handle communication between applications efficiently. If the user requests use of the camera then this request will be dealt with.
Terminate the application	Once the user exits the application the application must in fact terminate.

5. USER INTERFACE DESIGN

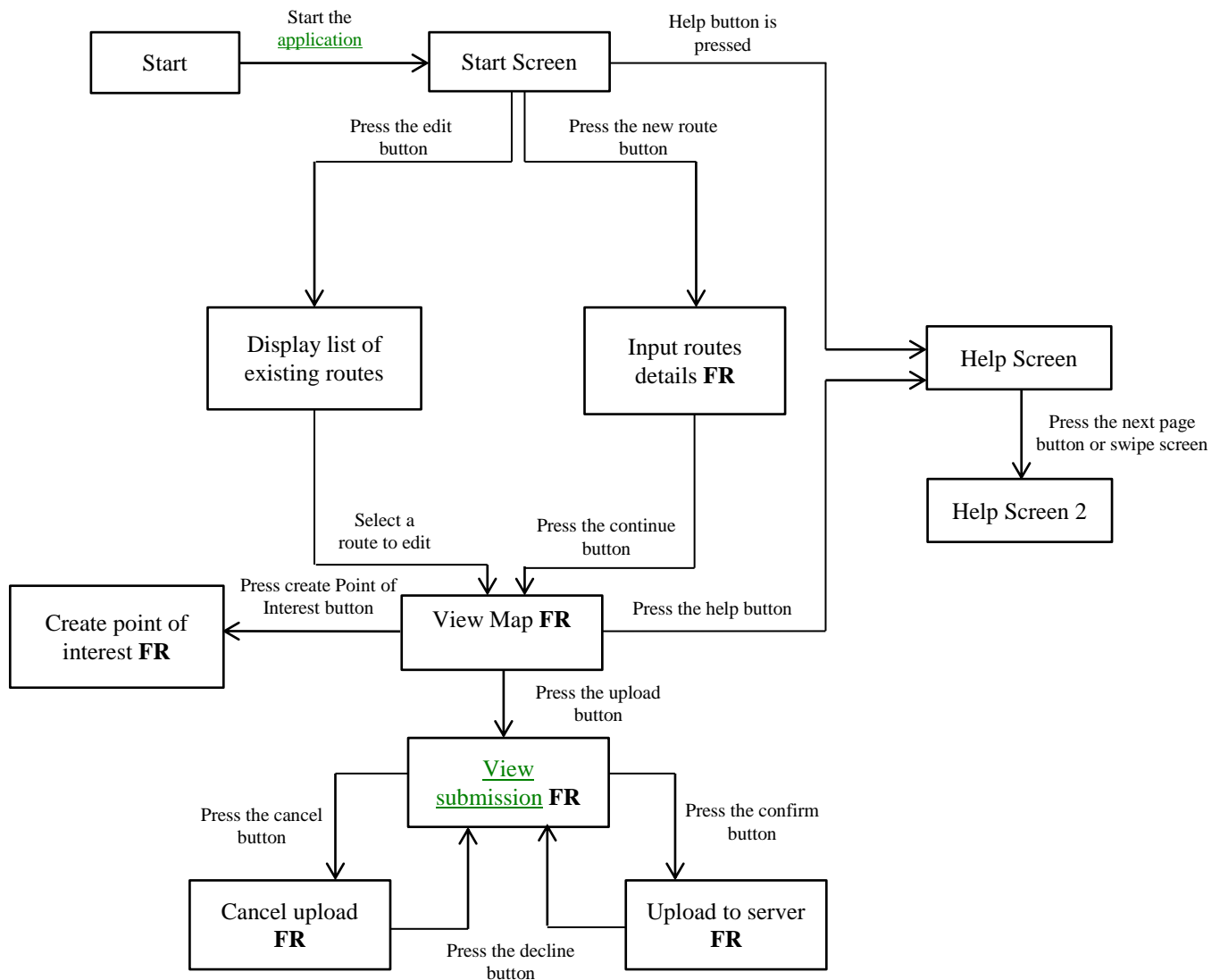


Figure 3:Flow Chart of the screen navigation

5.1 GUI design

Below are the designs for each element in the Graphical User Interface. These are simple representations of the proposed design for the Android application. These are designed to give a rough ideal of the finished applications look, but are by no means finalized designs.

5.2 Start Screen

This is the screen that will greet users when they load the app. It has the name of the app and the logo, along with two buttons. The first button, 'New route' allows the user to start mapping a new walk and takes the user to the “Create new route” screen. The second button allows the user to modify existing routes and takes the user to the “Display all existing routes” interface.

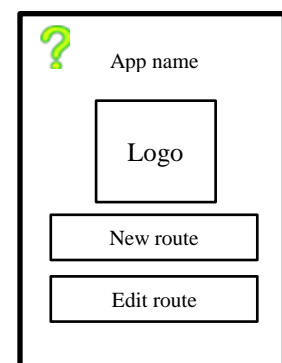


Figure 4: Diagram showing the Start Screen

Figure 5: The Display All Routes screen

5.3 Displaying all existing routes Screen

This screen contains a list of all available routes that the user can modify. The user can scroll through the existing routes and select one they wish to make changes to by pressing the “Edit” button. The number of existing routes is placed above the list of routes.

5.4 Create new route screen

This screen allows the user to enter the details for their walk and has three text fields: “Route name”, “Short description” and “Walk details”. After inputting the required details, the user can then press the “Begin tracking” button to proceed to the map screen.

Figure 6: Create New Route screen

5.5 View map screen

This is the screen where users will create their walks. It has a button allowing the user to create a new point of interest, which takes them to the “Create new point of interest” screen. It also has “Help” and “Upload” buttons. The help button takes the user to the help interface where they can troubleshoot problems that they may be encountering. The upload button takes the user to the “Review submission” screen before sending the finished walk to the server and adding it to the list of editable walks. There are also two counters for the length (time taken) of the walk and the number of locations featured within the walk. A map is placed within the centre of the screen, which displays the points of interest that have already been placed on the map.

5.6 Create new point of interest screen

Figure 7: Create New PoI screen

The PoI creation screen allows the user to designate locations that

should be featured in the walk. There are two fields: “Name”, where the user can input a name for the current location, and “Short description”, where the user can write a short summary of the featured location. The user can also take pictures of the current point of interest using their phones camera by pressing the “Take

Figure 8: The View Map screen

Figure 9: The Review Route screen

photo” button and then attach the photo to the walk from the phone's memory. The “Add to route” button then adds the location to the map and takes the user back to the view map screen.

5.7 Review submission screen

The review submission screen allows the user to check that all details of the route are correct in order to stamp out any errors. The name, short description and walk details are displayed and there are two text fields allowing the user to modify the description and the details of the walk. Below the text fields, there is a scrollable box containing a list of all the points of interest within the walk. At the bottom of the screen, there are two buttons: cancel and confirm. Cancel takes the user back to the cancel upload screen and the confirm button takes the user to the upload confirmation screen.

5.8 Cancel upload screen

This screen displays the details of the walk and at the bottom asks for confirmation if the user wants to cancel the walk. There are two buttons at the bottom: confirm and decline. If they press decline, the user is taken back to the previous screen (review submission). If they press confirm, the walk is deleted.

Figure 10: The Cancel Upload screen

Figure 11: The Upload Confirmation screen

5.9 Upload confirmation screen

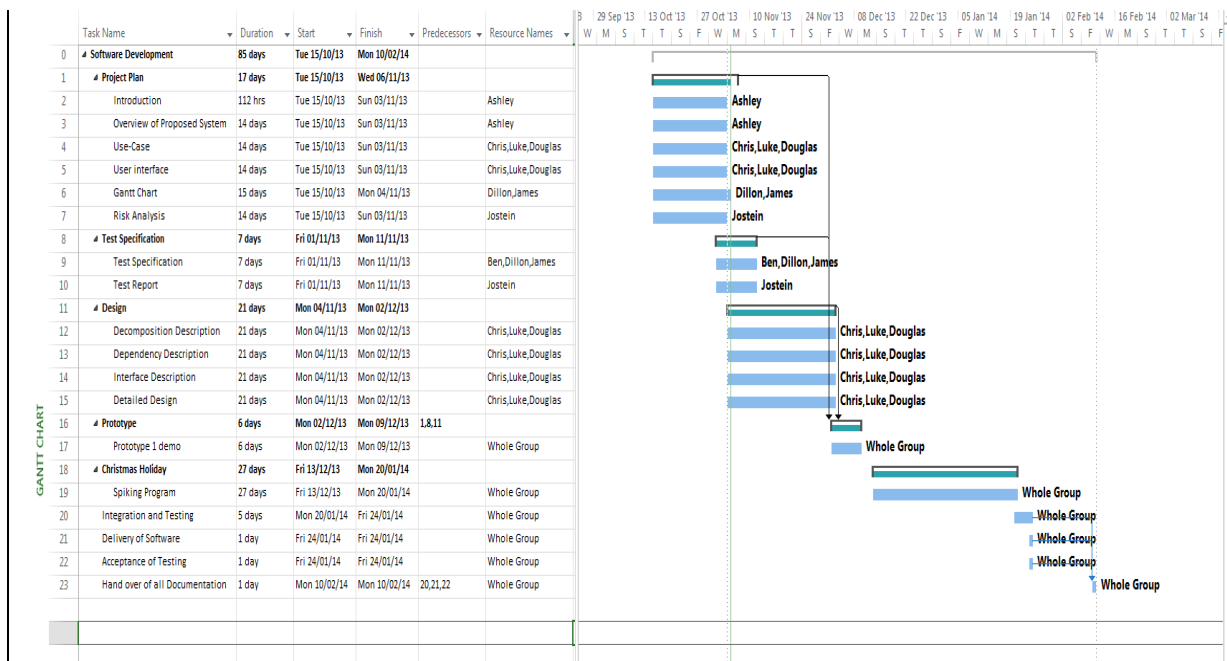
This screen displays the details of the walk and at the bottom asks for confirmation if the user wants to upload the walk to the server. There are two buttons at the bottom: confirm and decline. If they press confirm the walk is sent to the server and added to the list of walks available to edit. If they press decline, they are taken back to the review submission screen.

5.10 Help Screen

The helps screens will all be created using a consistent them and layout. Each help screen will display information on a specific topic (e.g. how to upload a route) and the user can navigate through these help pages using the “previous” and “next” buttons to either progress or back track through the pages. The screens themselves will display a basic text description of what the page is dedicated to (e.g. “How to upload:”) as well as providing a more detailed body of text regarding the contents of the help method itself.

Figure 11: The Help screen(s)

6. GANTT CHART



7. RISK ANALYSIS

7.1 Constant Risks

Task	Hazard	Risk level	How to deal with it
Scheduled meeting	Team member absent	low	Absent team member shall read the minutes of the missed meeting. Continued absence shall result in action being taken against the missing team member.
Scheduled meeting	Project leader absent	low	Meeting shall continue as normal, chaired by deputy leader.
Scheduled meeting	QA Manager absent	low	Meeting shall continue as normal; QA questions and decisions shall be handled by others in the QA team.
Scheduled meeting	Programmer absent	low	Meeting shall continue as normal; other programmers will raise any queries related to the program, on behalf of the missing programmer.
Informing the team whenever you complete a goal, and estimate how long it will take you to complete the next.	Lose team member contact	moderate	It is very important that every team member keeps updating the team on their progress via the group e-mail. If someone neglects their duty to do this, parts of documentation may become missing, and major setbacks in progress towards the next milestone may occur.
Handing work to the group	Local files lost	high	Team members shall ensure that copies of all work shall be backed <u>up</u> : either on the University file store or an external storage device. It is the sole responsibility of the team member to do so.
Handing over your work to the group	Illness or any other circumstance keeping you from completing your work	Moderate / high	If you ever get the feeling that you are unable to complete any/all parts of your assigned work in time for hand-in; Inform the rest of the group immediately. We all rely on each other to make sure our own work is completed within the planned time; if you are falling behind, you should inform the others that you are struggling, and we will help you complete your work any way we can.

7.2 Risks related to documentation

Task	Hazard	Risk level	How to deal with it
Handing over your documentation to the group	Late submission by some team member(s)	low	Internal deadlines shall be set prior to the final, external deadlines, ensuring that there is a buffer period allowing team members to assist if issues arise.
Handing over your documentation to the group	Parts of documentation missing or incomplete	moderate	All documents will be handed over to the QA manager for approval. Documents shall be easy to read, and in need of minimal editing by the QA team.
Handing over your documentation via GitHub	Git	low	We might not be able to edit any .docx files directly on git-hub, but we should be able to download your documentation and edit it on our own machines, then re-upload the edited version to Git-hub.

7.3 Risks related to development and delivery

Task	Hazard	Risk level	How to deal with it
Keeping up with the project timeline (Gantt chart)	Not being able to complete all tasks by the time planned on the Gantt chart	moderate	The programming team shall update the QA manager and project leader on their progress, as often as possible; allowing the swift resolution of any problems that arise.
Following the project specification	Some parts of the project are missing, incomplete, or not what the specification asks for.	moderate	<p>All points in the project specification shall be implemented, with final responsibility for this lying with the project leader, who shall ensure that the project follows the specification.</p> <p>Extensive testing of code shall take place to ensure that everything works as planned.</p> <p>The QA manager shall ensure that all documentation, code, and other work are both high-quality and comprehensive.</p>
Handing in the final version of the project	Downtime on, Blackboard / University Filestore / Git-hub / Aberystwyth University internet connection	moderate	<p>The team shall ensure that each stage of the project is ready to be submitted at least one day before the deadline. This will ensure that disruption from any downtime on Blackboard, the University file store, GitHub, or the University's Internet connection.</p> <p>Any downtime in these places is unlikely to last for longer than a day; so the project can be immediately submitted once services resume.</p>
Storing parts of the project on an external storage device that you carry around with you	You might lose your storage device, it could be stolen, or damaged by water	moderate	External storage shall not be carried around without reason, and additional backups should be made on a regular basis (e.g., daily at 6 p.m.)

7.4 Risks related to the usage of the program

Task	Hazard	Risk level	How to deal with it
Loading the map	No connection to the internet, or too slow to load the entire map	low	<p>If connection to the Internet is checked before the user gets the chance to start a new walk, one can make sure that no grey areas appear on the map.</p> <p>Alternatively, if a version of the map of Ceredigion is available offline, the team shall only have to worry about the GPS reading coordinates correctly.</p> <p>If the coordinates are correct, one should be able to switch from the offline-version of the map to the online when Internet connection is resumed.</p> <p>(This should require the app to check for a new version of the offline map regularly, for maintainability purposes)</p>
Displaying a list of existing routes	The map has changed after the route was made	low	<p>If the map is different from when the route was made, it is possible that the resultant route will end up showing a route that cross over train tracks and through residential houses.</p> <p>A possible response is to store the map along with a route. However, this may result in too much data being stored locally.</p>
Using the UI	The application is run without a touch screen.	low	No certified Android phone lacks a touch screen (11). The Android developers recommend that one assumes that all Android phones have a touch-screen, making this risk a very low priority.
Using the help-screen(s)	The help-screen is too descriptive, or does not cover all problems	low	The QA manager shall take oversight of all help-screens. Usability tests shall take place to ensure that the help screens are helpful.
Creating a new point of interest	Help screen is not available	moderate	<p>The developers shall ensure that the help screen is available.</p> <p>If a set of help-screens are only displayed in this portion of the program, the other set of help-screens will feel less large and easier to navigate.</p>
Viewing the map	The map is currently loading, and only displaying a grey area	low	<p>The user must be aware of any delays in loading the map.</p> <p>A small animation or textbox shall be added to inform the user that the map is loading; thereby avoiding any confusion as to why parts of the map are missing.</p>
Cancelling / confirming the submission	The confirm – decline buttons have switched places from where they were on the previous screen	low	<p>If one presses buttons without reading what they say, having the “cancel” / “confirm” buttons in different positions in the “Review submission screen”, “Cancel upload screen”, and “Upload confirmation screen”, could result in someone unintentionally deleting their walk.</p> <p>Designs will ensure that the ‘default’ option is always to the right of the ‘cancel’ button, or equivalent.</p>
Editing the walking tour before submitting it	Should we have a separate “edit” button, to prevent someone from	low	If it is made very clear that the user is editing the submission, by bringing up a new window when you click on any of the information in the submission, accidental deletion of information can

	accidentally changing some information?		<p>be avoided.</p> <p>This is a small problem, and probably not a huge deal as you can edit the walking-tour after submission as well, although it might look better.</p>
Cancelling any other parts of the program	As far as I can tell, there is no way to go back to the previous screen in: “Display existing routes”, “Create new route”, “create new point of interest”, or the “Help” screens.	moderate	<p>If one cannot cancel “create new route”, then one would have to first create a route, and then cancel the route, and finally going back to the start screen.</p> <p>If one cannot cancel “create new point of interest”, then one would be forced to create a new POI, and then delete it again when you are submitting the walking-tour.</p> <p>If there is no obvious way to exit the “Help-screen”, some users might get stuck in there, and forced to restart the entire program.</p>
Removing a Point of interest	There is no way to remove a POI in the “View map screen”	low	<p>Users should be able to remove any points of interest that they do not want from the map.</p> <p>However, being able to remove them whilst reviewing the submission may not be enough; for example, placing a PoI in a slightly incorrect location, and want to relocate it.</p> <p>(An example of this might be inaccurate GPS-coordinates.)</p>

8. REFERENCES

1. **Alliance, Open Handset.** [Android](http://www.openhandsetalliance.com/Android/overview.html) Overview. [Online] [Cited: 30 October 2013.] <http://www.openhandsetalliance.com/Android/overview.html>.
2. Application Fundamentals. [Android](http://developer.Android.com/guide/) Developers. [Online] [Cited: 30 October 2013.] <http://developer.Android.com/guide/>.
3. **Etherington, Darrell.** [Android](http://techcrunch.com/2013/08/07/Android-nears-80-market-share-in-global-smartphone-shipments-as-ios-and-blackberry-share-slides-per-idc/) Nears 80% Market Share In Global Smartphone Shipments, As iOS And BlackBerry Share Slides, Per IDC. *TechCrunch*. [Online] 7 August 2013. [Cited: 4 November 2013.] <http://techcrunch.com/2013/08/07/Android-nears-80-market-share-in-global-smartphone-shipments-as-ios-and-blackberry-share-slides-per-idc/>.
4. **Apple, Inc.** Choosing an iOS Developer Program. *Apple Developer*. [Online] [Cited: 30 October 2013.] <https://developer.apple.com/programs/start/ios>.
5. **The PHP Group.** Usage Stats for January 2013. *php.net*. [Online] January 2013. [Cited: 4 November 2013.] <http://php.net/usage.php>.
6. **Aberystwyth University.** Module Information: CS25010. [Online] [Cited: 4 November 2013.] <http://www.aber.ac.uk/en/modules/deptcurrent/?m=CS25010>.
7. **The PHP Group.** Technology Overview. *PHP Manual*. [Online] 1 November 2013. [Cited: 4 November 2013.] <http://us1.php.net/manual/en/mysqlinfo.terminology.php>.
8. About. *phpMyAdmin*. [Online] [Cited: 4 November 2013.] http://www.phpmyadmin.net/home_page/index.php.
9. June 2013 Web Server Survey. *Netcraft*. [Online] June 2013. [Cited: 4 November 2013.] <http://news.netcraft.com/archives/2013/06/06/june-2013-web-server-survey-3.html>.
10. What is a LAMP stack? *Stack Overflow*. [Online] 8 April 2012. [Cited: 4 November 2013.] <http://stackoverflow.com/questions/10060285/what-is-a-lamp-stack>.
11. Compatibility Program Overview. [Android](http://source.Android.com/compatibility/overview.html) Source. [Online] [Cited: 4 November 2013.] <http://source.Android.com/compatibility/overview.html>.

DOCUMENT HISTORY

<i>Version</i>	<i>CCF No.</i>	<i>Date</i>	<i>Changes made to document</i>	<i>Changed by</i>
0.1	Nil	2013-11-04	Severely copyedit and collation	dog2
0.2	Nil	2013-11-04	Added introduction and help screen info	che16