Project Group 05
Project Management Plan

Authors: srr11; hac22; bmo;
wia2; wjl3; njv1
Config Ref.: SE_05_PM_01
Date: 06/11/2013
Version: 1.1
Status: Release

Department of Computer Science
Aberystwyth University
Aberystwyth, Ceredigion
SY23 3DB

Contents

1. Introduction

1.1. Purpose of this document

The purpose of this document is to show how the project group has decided to carry out the client's requirement specification for a walking tour application as a set of objectives and milestones.

1.2. Scope

This document contains the details of the group project. It contains the group's choice of platforms and high level architecture as well as justification for the choices made. The Use-Case diagram is included showing how different users will interact with the different parts of the system as well as screenshots and descriptions of the GUI. The proposed Gantt chart is part of this document, detailing what the group will be working on in the process of development and in what timeframes.

1.3. Objectives

The objectives of this document are as follows:

1.3.1. To describe the overview of the proposed system;

1.3.2. To describe how the main components of the system will interact with each other;

1.3.3. To present the base user interface and describe how the user will interact with it;

1.3.4. To provide a list of the project milestones;

1.3.5. To provide a list of all tasks that need to be completed on the project and their anticipated timeframe in the form of a Gantt chart;

1.3.6. To list possible issues the team might encounter during development in the form of risk analysis.

2. Overview

The proposed system is a walking tour android application that allows people to “record” walks they make through GPS coordinates and add information and pictures at points they find something interesting. The walks will be available to view on a website for everyone.

2.1. Platforms and high level architecture

2.1.1. Android

The platform has been specified by the client in the project guideline.

2.1.2. IDE

We are using the Eclipse IDE with the ADT plugin because that is the IDE the team is most familiar with. We took into account Android Studio but reached the conclusion that Eclipse was more stable provided a better user interface.

2.1.3. Android Mapping API

We are using the Google Maps for Android API for the app because it comes as part of the Android SKD and gives the user a full screen map to view, which the programmer can overlay with their own controls.

2.1.4. PHP

The research showed that PHP is capable of processing easily JSON files, which we will be using, also it is available on most servers and is currently taught in one of the second year modules.

2.1.5. JSON

JSON would be the best data set to use for the tours. This is largely due to it being significantly lighter in weight than XML and how easy it is to process in PHP.

2.1.6. Web side mapping API

Because of its wide variety of browser support, features and simplicity, the Leaflet API stood out as exactly what the project group would need to use.

2.1.7. Drupal

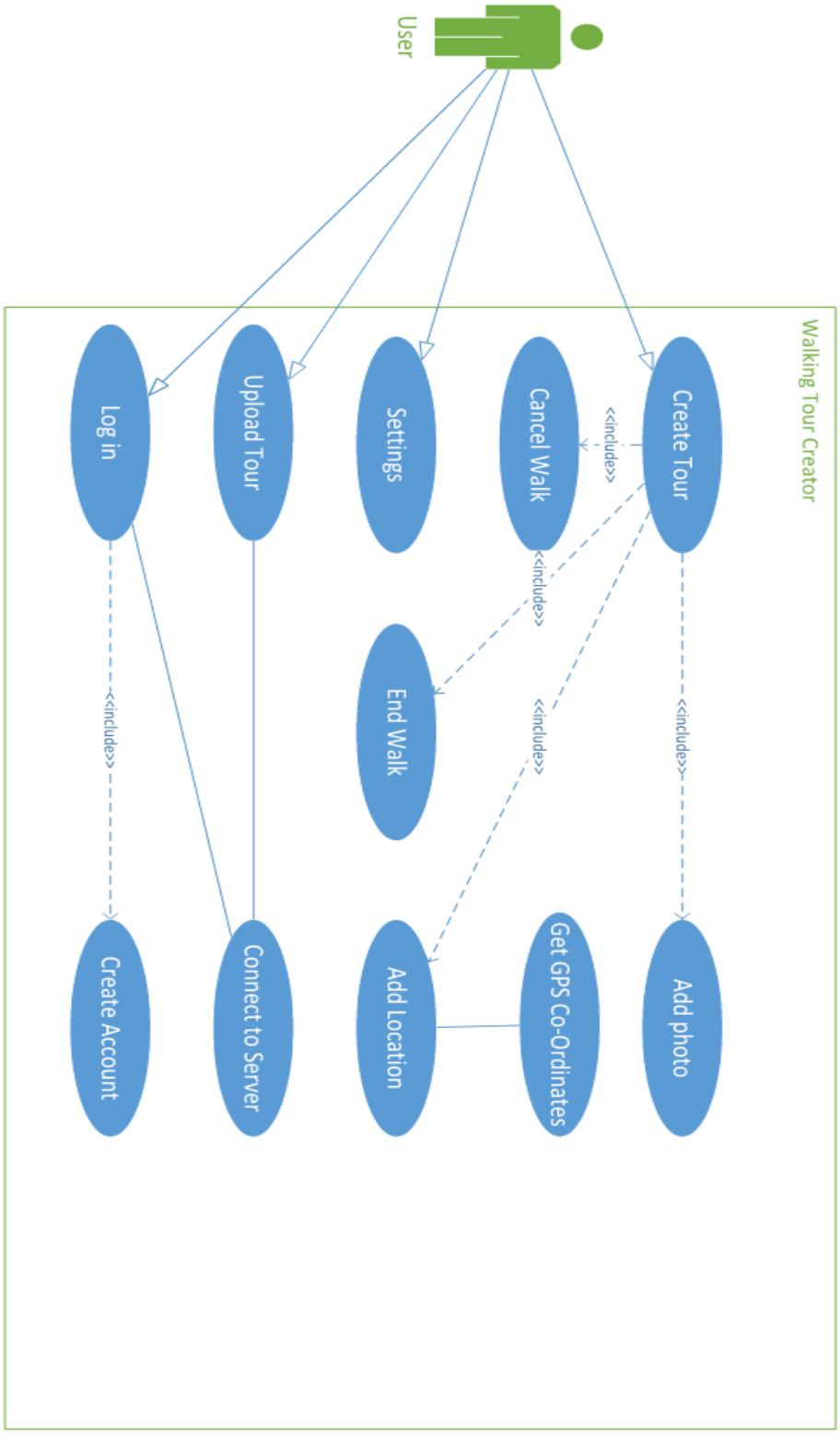
Drupal provides features such as an audit trail and database replication which will prove useful. It is also the CMS most familiar to the team and it will allow for the implementation of a user based system.

2.2. Target user base

The client has suggested second year computer science students as target base, but the app is not targeted specifically at that user group. Its purpose can be different for different users thus making it usable in a variety of ways and the actual system has to be easy to use by most age/background groups.

3. Use case

3.1. Android Use Case



3.2. Descriptions of the Android Use Case

3.2.1. Create Tour:

This will allow the user to create a new walking tour, regardless of logging in to our servers or not the user will be asked to select a title for their tour and a short description (as minimal) before starting the tour, during the tour they can attach photos of local scenery and the like with a short description of the photo.

3.2.1.1. Add Photo:

This will use the built-in camera app on the android device to take a photo for the user to add to their walk.

3.2.1.2. Add Location:

This will get called periodically to allow for an accurate walking tour to be created.

3.2.1.3. End Walk:

When the user presses to end the tour, they will be given a summary of the tour, and be asked to fill in the missing long description (if they did not do so before creating the tour), they will then be prompted to see if they want to have the walk saved locally or uploaded straight away.

3.2.2. Settings:

This is where the user will give their preferences for different in app options, such as the upload option, if a user is concerned about their data limits they can choose to only upload over Wi-Fi.

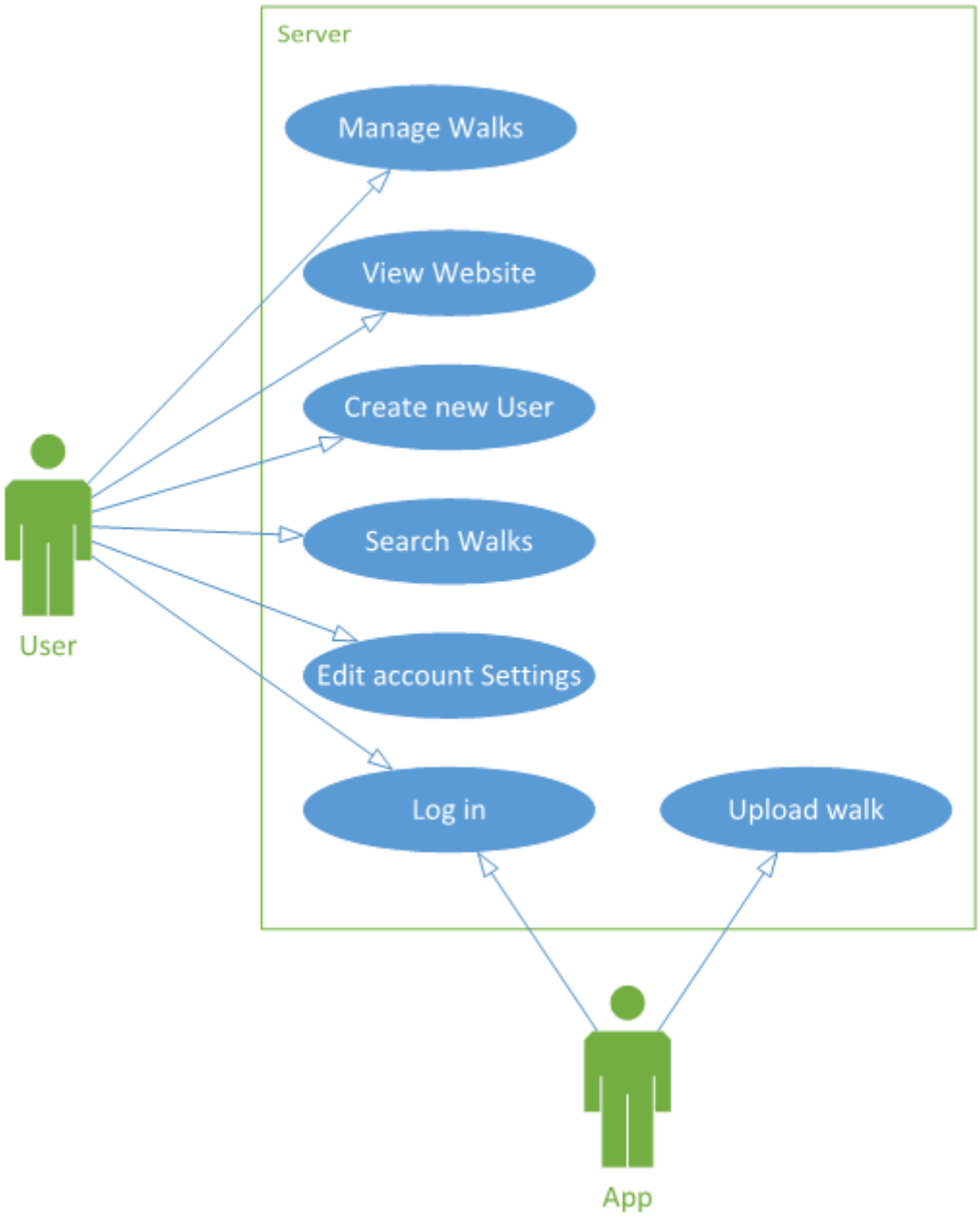
3.2.2.1. Log in:

This is where members of the website can sign in to have the walks they create attributed to them. Possibly the user could edit the notes on their walks from previous uploads although this isn't in the specification.

3.2.2.2. Upload Tour:

If the user has chosen to save their tours locally on the device, this option will allow the user to select which tours they wish to upload to the website.

3.3. Web Side Use Case



3.4. Description of the Web Side Use Case

3.4.1. View Website

Displays our walking tour viewer homepage.

3.4.2. Database Dependent Use Cases

3.4.2.1. Manage Walks

Allows the user the ability to edit/delete previous walks on their profile.

3.4.2.2. View Tour:

This will display a map for the user to view with "pins" in it that have pictures attached along with the notes associated with the walk.

3.4.2.3. Create New User:

Creates a new user account, minimum requirements are e-mail address, username, and password. Other data can be held, for example in an extended profile.

3.4.2.4. Log in:

This connects the user with the database to allow them to view/edit their previous walks.

3.4.2.4.1. Edit Account Settings:

This is where the user will change preferences on their account for the website.

3.4.2.5. Search Walks:

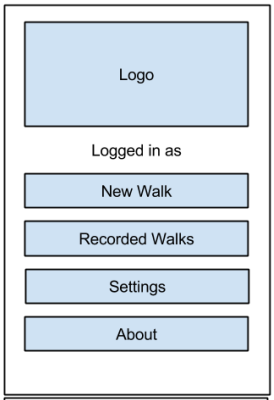
Searches based on keyword/location and delivers the top “x” amount of walks.

3.4.2.6. Receive Tour:

Takes a MIME file sent from the app, decodes it and stores the information in a SQL database.

4. Android User Interface Design

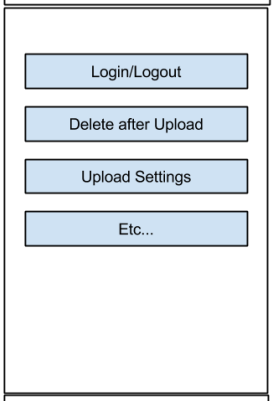
The following wireframes are initial concept for the design and will be modified accordingly with time.



4.1. Main Screen

On the main screen the user can see whether he is logged in or not. The start button will take the user to the preparation screen for a recording where they can add descriptions and a title for the walk.

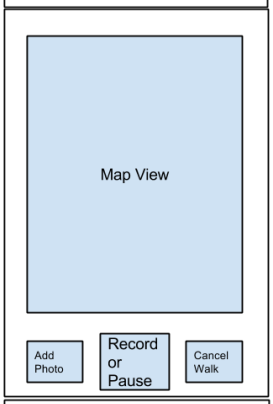
The settings and Info buttons take the user to the specified sub-screen.



4.2. Settings screen

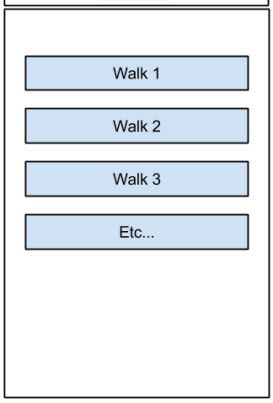
The settings screen provides the user with the option to log in if they have not already. For the time being we have made the option to delete the walk data after upload is complete or not.

The upload settings give the user a choice, whether they would like to use their mobile internet connection as well as Wi-Fi to upload walks to the database online.



4.3. Map screen

On the map screen the user will see his position on the map and the path that he has already walked on. They can use the add photo button to select/take a photo and add a description to it. The cancel walk will exit the walk without saving it.

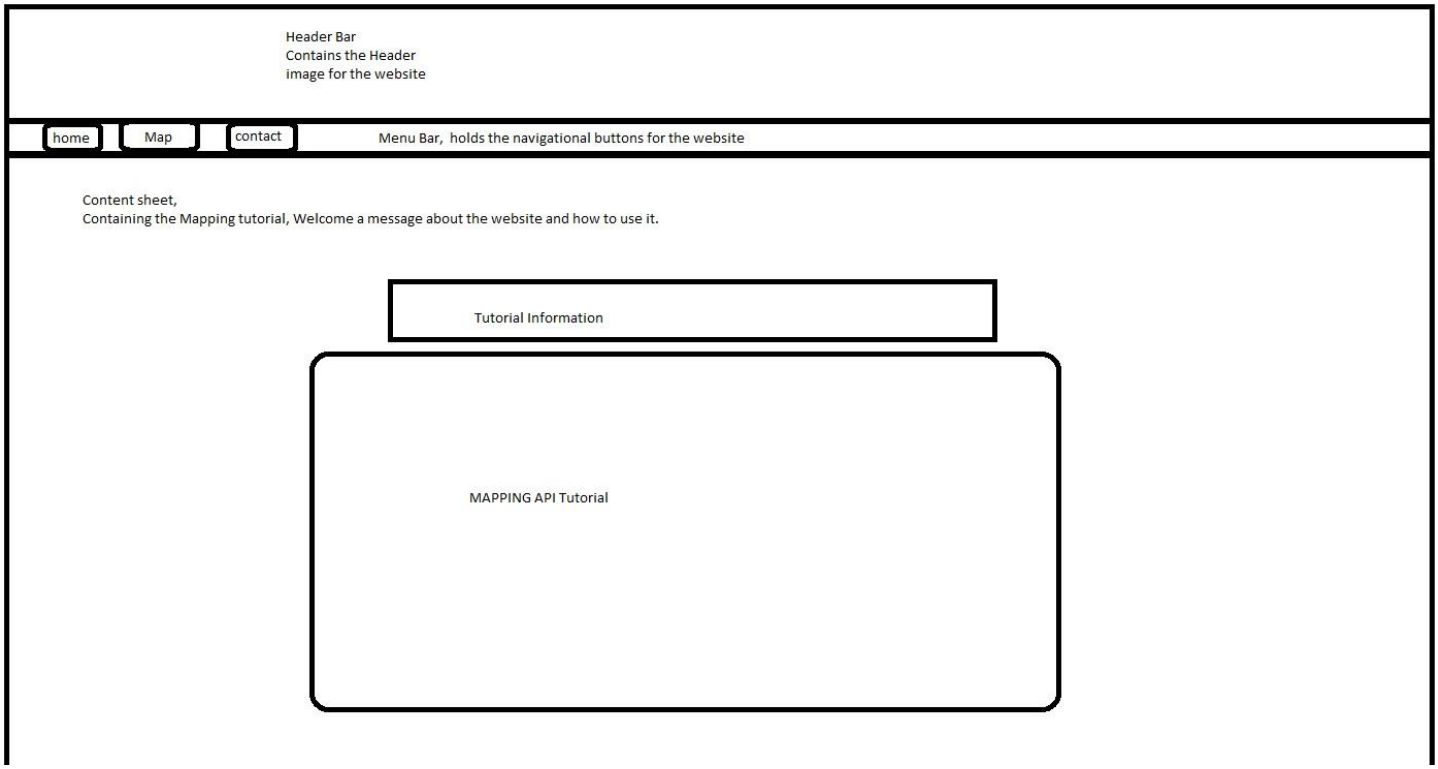


4.4. Recorded walks screen

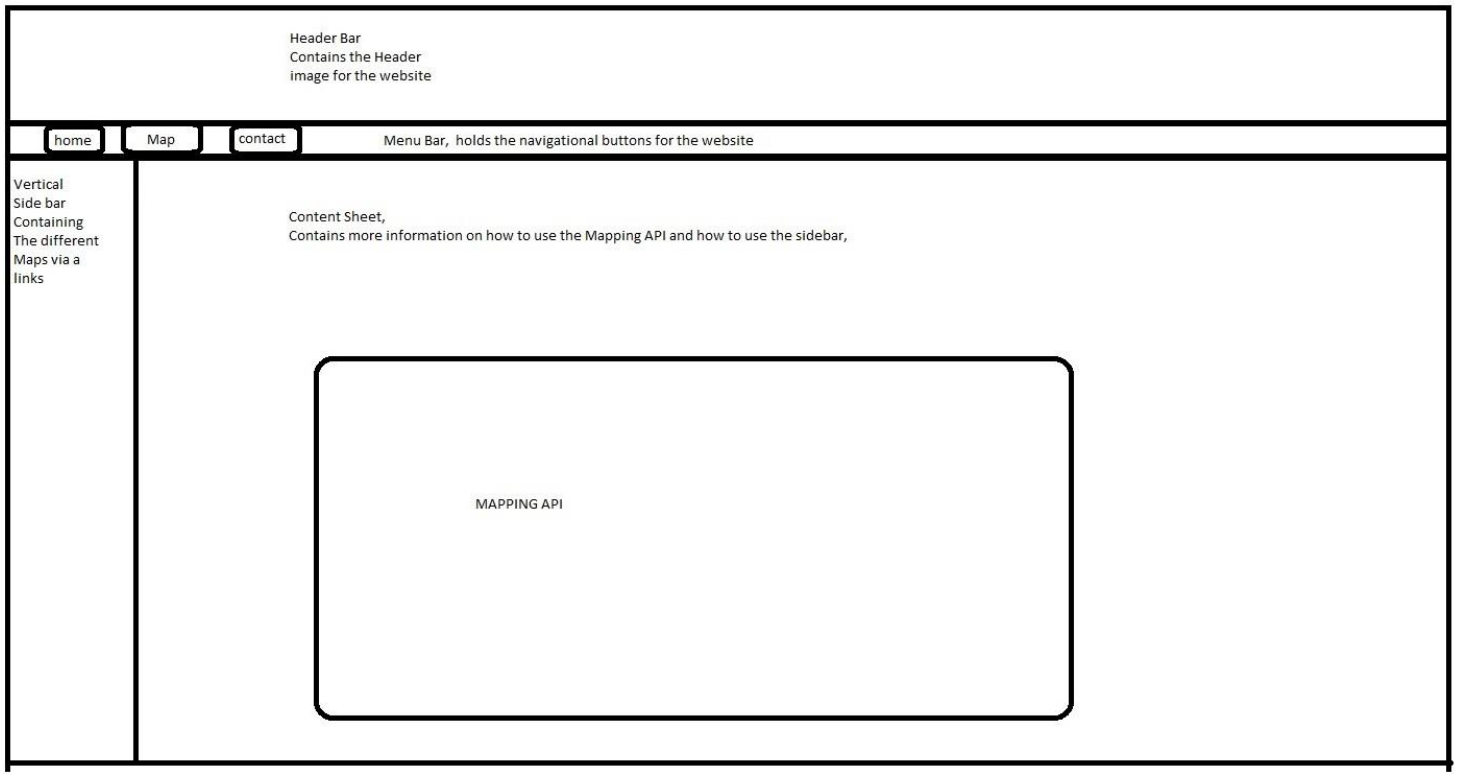
On the recorded walks screen the user can see a list of previous walks they have recorded.

5. Website User Interface Design

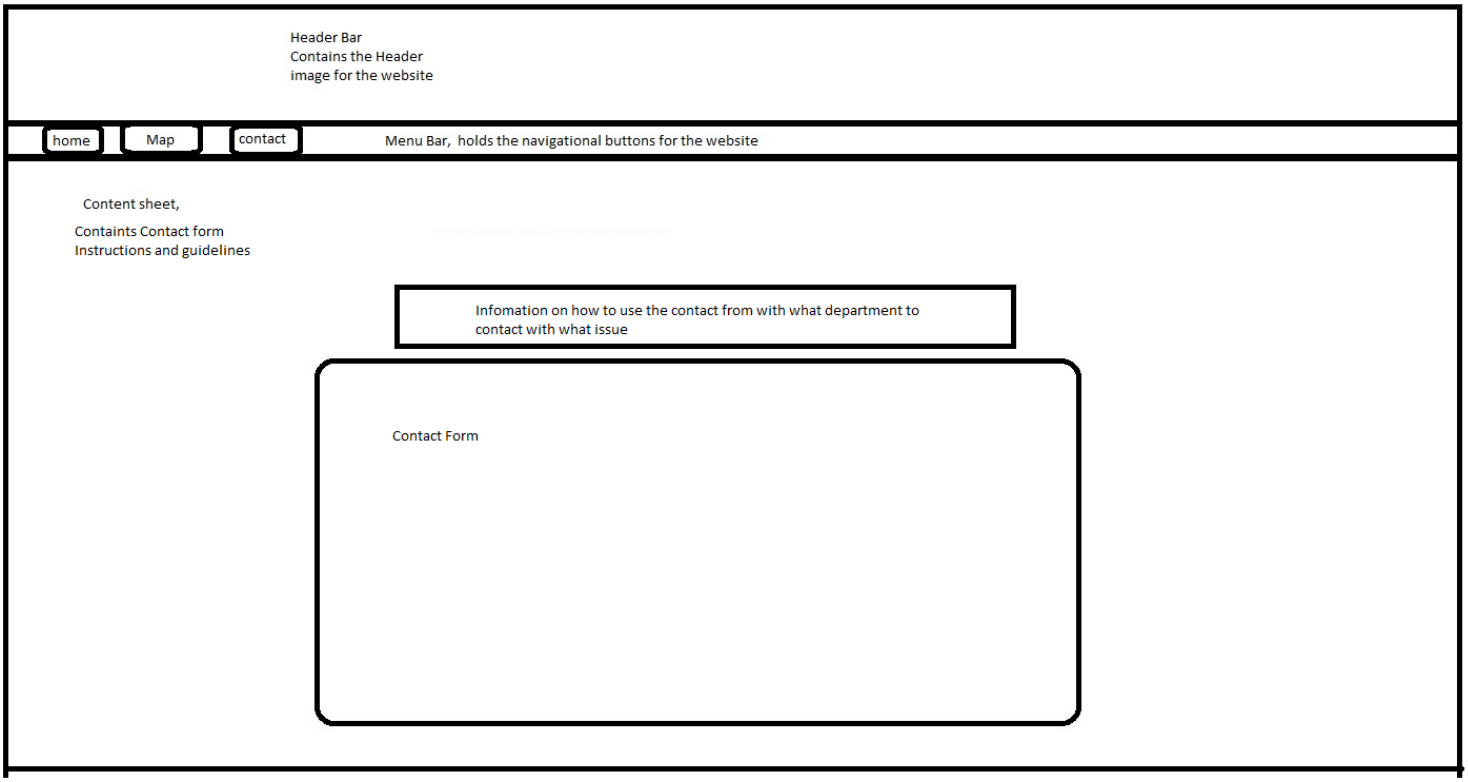
5.1. Home Page



5.2. Map page



5.3. Contact page



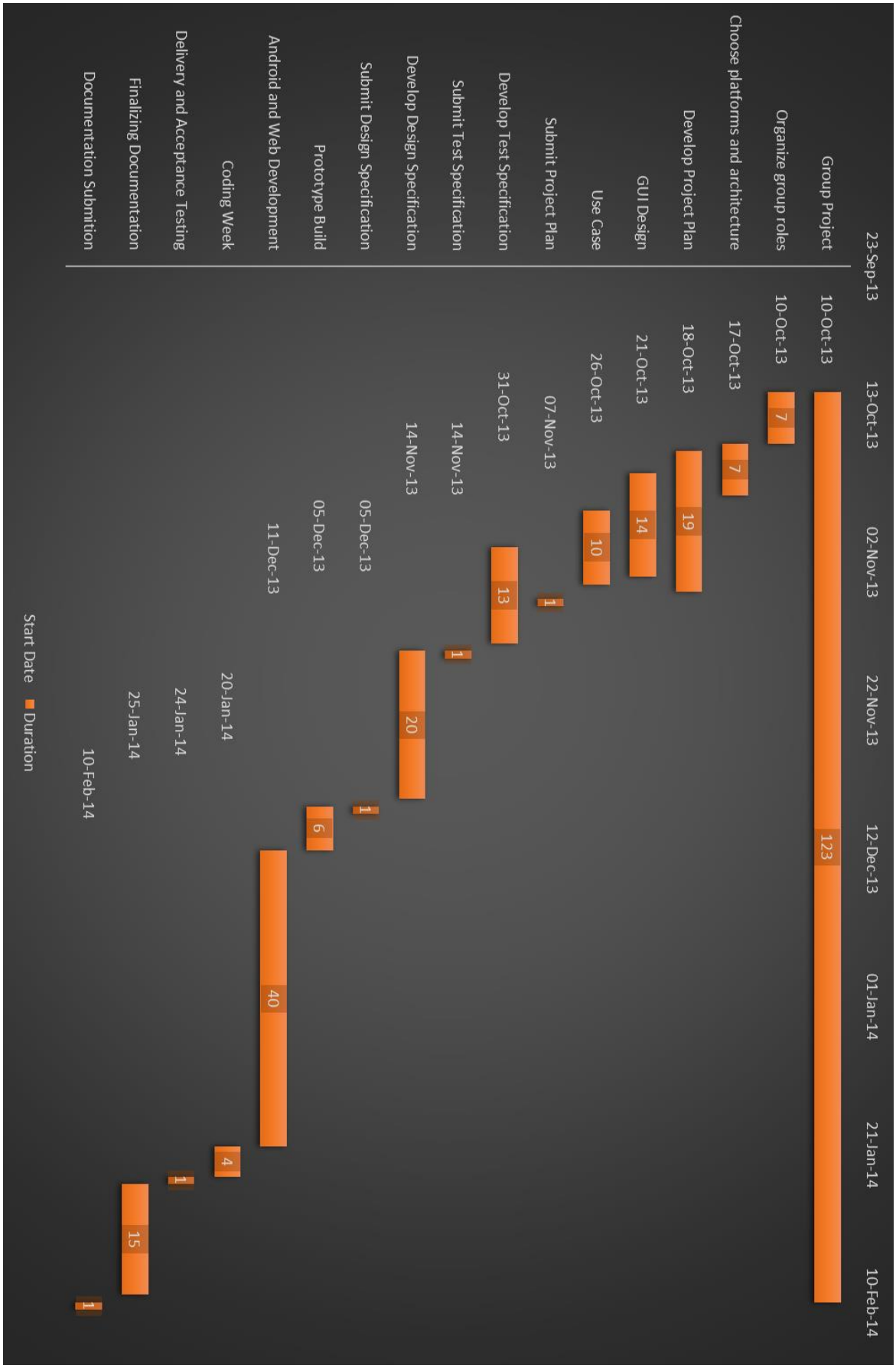
5.4. Overview

The site will consist of 3 main pages. The home page, where the user can log in, view a tutorial on how to use the website and navigate further.

The map page, where a user can load and view a walk that is in the database.

The contact page where the project group's contacts are listed so that the user can give feedback to us about the site.

6. Gant Chart


23-Sep-13
13-Oct-13
02-Nov-13
22-Nov-13
12-Dec-13
01-Jan-14
21-Jan-14
10-Feb-14
Group Project
Organize group roles
Choose platforms and architecture
Develop Project Plan
GUI Design
Use Case
Submit Project Plan
Develop Test Specification
Submit Test Specification
Develop Design Specification
Submit Design Specification
Prototype Build
Android and Web Development
Coding Week
Delivery and Acceptance Testing
Finalizing Documentation
Documentation Submition
10-Oct-13
10-Oct-13
17-Oct-13
18-Oct-13
21-Oct-13
26-Oct-13
07-Nov-13
31-Oct-13
14-Nov-13
14-Nov-13
05-Dec-13
05-Dec-13
11-Dec-13
20-Jan-14
24-Jan-14
25-Jan-14
10-Feb-14
123
7
7
19
14
10
1
13
1
20
1
6
40
4
1
15
1
Start Date
Duration

7. Risk Assessment

Risk Event	Risk Level	Mitigation
Absence of team member	2	All meetings are announced with enough time before them for everyone to make room in their schedules. If someone cannot make it, they need to appologise and look through the minutes of the meeting.
Loss of contact	4	Members need to be in constant contact throughout the project. To avoid such a scenraio, all members have exchanged all possible contact information.
University filestore/GitHub Downtime	3	The University filestore is reliable and so is GitHub, but in case of such an event relevant information will be sent via the group e-mail/social media. Everyone in the group will cease work and wait for further instructions on what exactly everyone is expected to work so that there is no duplicating or loss of information.
File corruption	4	The files are going to be backed up on the university filestore and on GitHub so as long as team members are concious about their duties to back up everything, the risk is not too high.
Illness or other unexpected circumstance	5	In the event of serious illness or other event preventing a team member to contribute, they should inform everyone else as soon as possible so that appropriate action can be taken.
Documentation errors	5	Every document, after it has been drafted, has to be presented to the whole team. After everyone has agreed on it's contents and the QA manager has accerted that it fits within standards, it is finished and submitted.
Not fuffiling the project timetable	5	Every task to be started as soon as possible allowing enough time for revision and redaction. Programming and web teams have to keep close contact with the project leader and QA manager to make sure they are working as efficiently as possible.
Parts of the impelementation missing or incomplete	4	The web and android teams have to document every part of the system they build and refer to the project leader and QA manager if any problems arise. The project lead and QA manager have to keep track of general progress and make sure all requirements are fufilled.

Risk Level	2	3	4	5	6
Consequences	Low	Low/Med	Med	Med/High	High

8. References

N/A

9. Document History

Version	CCF No.	Date	Changes Made to the Document	Changed by
1.0	N/A	06/11/2013	N/A – First release of project plan	srr11
1.1	N/A	06/11/2013	Changed the Web Side Use Case	bmo