

# Group 10 Project Plan

*November 6, 2013*

**SE.10.D1**

Version: 1.0  
Status: Draft

Contributor Name	Role
Daniel Clark	Project Lead
Charles Newey	Deputy Project Lead
Mark Lewis	QA Manager
Ashley Iles	Android Developer
Kenny Packer	Android Developer
Stephen McFarlane	Web Developer

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

©Copyright Group 10, 2013

# Contents

<b>1</b>	<b>INTRODUCTION</b>	<b>3</b>
1.1	Purpose of This Document . . . . .	3
1.2	Scope . . . . .	3
1.3	Objectives . . . . .	3
<b>2</b>	<b>SYSTEM OVERVIEW</b>	<b>4</b>
2.1	Platforms and High Level Architecture . . . . .	4
2.2	Target Users . . . . .	4
2.3	Client and Server Structure . . . . .	4
<b>3</b>	<b>GANTT CHART</b>	<b>5</b>
<b>4</b>	<b>RISK ANALYSIS</b>	<b>6</b>
4.1	Ongoing Tasks . . . . .	6
4.2	Documentation . . . . .	6
4.3	Software Development . . . . .	7
4.4	Risk Grade and Recommended Action Key . . . . .	7
<b>5</b>	<b>USE CASES AND SCENARIOS</b>	<b>7</b>
5.1	Overview of Use Cases . . . . .	7
5.1.1	Core Functionality . . . . .	7
5.1.2	Extended Functionality . . . . .	7
5.2	Android - Core Functionality . . . . .	8
5.2.1	Create Route . . . . .	8
5.2.2	Record Route . . . . .	9
5.2.3	Complete Route . . . . .	10
5.2.4	System . . . . .	11
5.3	Android - Extended Functionality . . . . .	12
5.3.1	Create Route . . . . .	12
5.3.2	Record Route . . . . .	12
5.3.3	Complete Route . . . . .	13
5.3.4	System . . . . .	13
5.4	Website - Core Functionality . . . . .	14
5.4.1	System . . . . .	14
5.5	Website - Extended Functionality . . . . .	15
5.5.1	System . . . . .	15
<b>6</b>	<b>REFERENCES</b>	<b>16</b>

# 1 INTRODUCTION

## 1.1 Purpose of This Document

The purpose of this document is to show that we have met the outlined objectives specified by the client. The main objective is as follows: "Walking Tour Creator (WTC) is a computer-based system to compile data about a walking tour, and structure it in a database that can be used by a mobile application which guides people around the walk, showing them places of interest along the way. WTC will be able to build a number of related walks, showing the location and details of places of interest along with photos, audio, video, about each place. All textual data will be stored in English."

This document will show we have managed to simplify these into a set of manageable goals. Gantt charts and GitHub's contribution analysis functionality will be used in conjunction to show and monitor the progress of the project's major tasks and other milestones. We are also implementing a risk analysis system that should alert us to the various problems that we may face whilst completing our objectives, and also allow us to mitigate the risks involved.

## 1.2 Scope

This document should take into account the specifications of the project. This document includes an overview of our proposed systems - which will encompass our choice of platforms, some high-level architectures and a description of prospective users. The document also contains; a use case diagram giving an overview of how the app and the web systems will interact, mock-ups and descriptions of the UI design and how both applications will interact with the user, a Gantt chart which displays the start and end dates for the main milestones, and a risk analysis for the project.

## 1.3 Objectives

These main objectives are to show our initial plans for the project. These goals include an overview of our proposed system, a set of use case diagrams, user interface designs, a Gantt chart, and a risk analysis.

- Produce an overview of our proposed system, which matches our client's specifications.
- Produce a set of detailed use case diagrams to define the interactions with the system components and users and provide them with a concise explanation, along with example usage scenarios.
- Create a UI for the Android and web systems we will employ, with a succinct explanation for our chosen design.
- Employ the usage of a Gantt chart, to document our expected progress as a team.
- Highlight aspects of our project plan which may cause us difficulties in completing our assigned work in a timely manner.

## **2 SYSTEM OVERVIEW**

### **2.1 Platforms and High Level Architecture**

Sample text

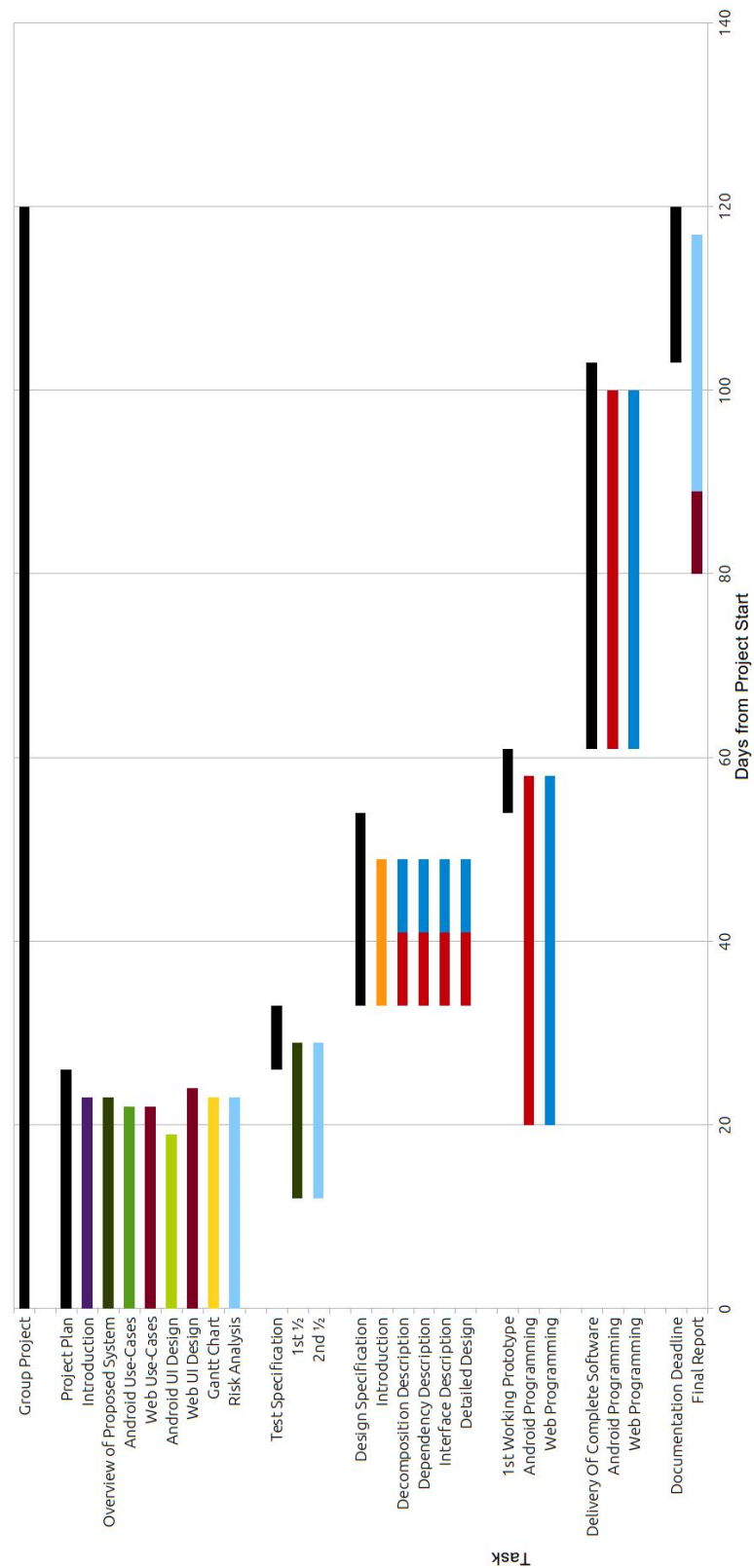
### **2.2 Target Users**

Sample text

### **2.3 Client and Server Structure**

Sample text Include stuff about protocol, yada yada

### 3 GANTT CHART



## 4 RISK ANALYSIS

### 4.1 Ongoing Tasks

Risk Event	L	M	Risk	Mitigation
Team member absence	0.6	0.3	0.18	All team members to regularly check emails and the agreed online resources for meeting times. Being unaware of meetings is not a valid excuse. If a team member is unable to attend a meeting, the project lead (Daniel) must be notified as soon as they know they can't attend.
Project Lead absence	0.6	0.5	0.30	Meeting to go ahead as planned with Charlie (Deputy Project Lead) taking the meeting.
QA Manager absence	0.6	0.3	0.18	Meeting to go ahead as planned with Steve (Deputy QA Manager).
Unable to contact team member	0.3	0.8	0.24	Ensure that all team members regularly check emails and other agreed online resources, as well as checking meeting minutes so they are aware of any outstanding tasks/actions. Persistently being unreliable with result in a warning, and further action if necessary; e.g. carding or role reallocation.
Git failure	0.3	1.0	0.3	All work to be backed up regularly in several places in case of human error or Git failure.
Major illness or unexpected circumstances	0.5	0.9	0.45	Team members to be notified as soon as possible, in case any urgent tasks need to be re-assigned or completed by another team member.
Git conflict	0.3	0.9	0.27	All team members are to have read the information on the project wiki on Git conflicts. If a conflict is encountered, then it must be resolved immediately. If a conflict cannot be resolved easily, then an appropriate team member (Git expert/project lead) must be notified and the conflict must be resolved. Try to ensure an even task allocation, to avoid multiple team members working on the same code simultaneously.

### 4.2 Documentation

Risk Event	L	M	Risk	Mitigation
Late submission	0.4	0.8	0.32	Deadlines for documentation to be brought forward to ensure that any future problems encountered will come to light within a reasonable time frame. If team members run into any problems, they are to alert the group so that a solution can be issued.
Inadequate quality submission	0.4	1.0	0.4	Documents to be checked by either of the QA managers or project leaders before submitting. If team members need help then they should ask the rest of the team for help.
Human error	0.5	0.2	0.1	All documents to be checked for spelling, grammar, logic, and clerical errors by the creator of each document and at least one other team member.
Loss of documentation	0.4	1.0	0.4	Documentation to be stored and versioned on Git and backed up individually by the document's creator. Each individual is responsible for their own documentation.

### 4.3 Software Development

Risk Event	L	M	Risk	Mitigation
Project behind schedule	0.5	1.0	0.5	Development to begin as soon as possible. Charlie (lead developer) to delegate any appropriate outstanding tasks. Regular feedback to be given by Daniel (project lead) to ensure schedule is adhered to.
Parts of the project missing/incomplete	0.4	1.0	0.4	Daniel (project lead) and Charlie (lead developer) to delegate programming tasks appropriately. Constant testing by QA managers and project leaders to check if code is of a sufficient quality.

### 4.4 Risk Grade and Recommended Action Key

Risk Grade	Less than negligible	Negligible	Acute	Severe	Critical	Catastrophic
Risk score	< 0.2	0.2 - 0.39	0.4 - 0.59	0.6 - 0.79	0.8 - 0.99	1.0
Action	Tolerate	Tolerate	Tolerate or treat	Treat	Transfer	Terminate

## 5 USE CASES AND SCENARIOS

### 5.1 Overview of Use Cases

#### 5.1.1 Core Functionality

The "core functionality" sections represent the highest priority features for the project, and will be the bare minimum functionality that the project will contain.

#### 5.1.2 Extended Functionality

The "extended functionality" sections represent the lower priority features, that will be implemented within the project if time permits.

## 5.2 Android - Core Functionality

### 5.2.1 Create Route

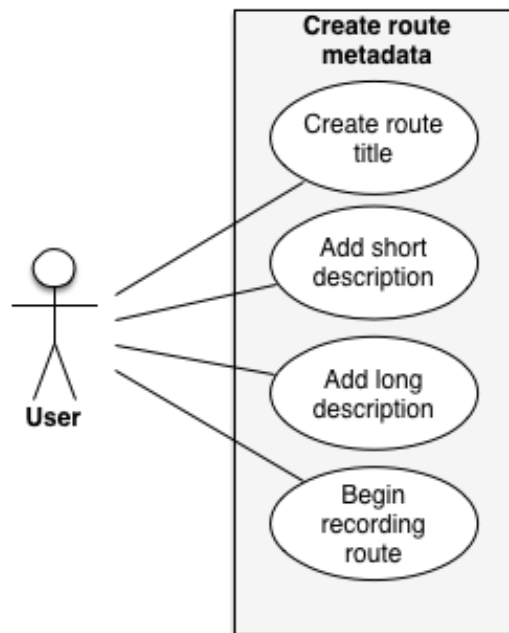


Figure 1: A user wants to create a new walk. The user will progress from the main screen to the "create route" screen, and fill in the route title, a short description (a tagline), and a long description. From there, the user can progress to recording the route.



### 5.2.2 Record Route

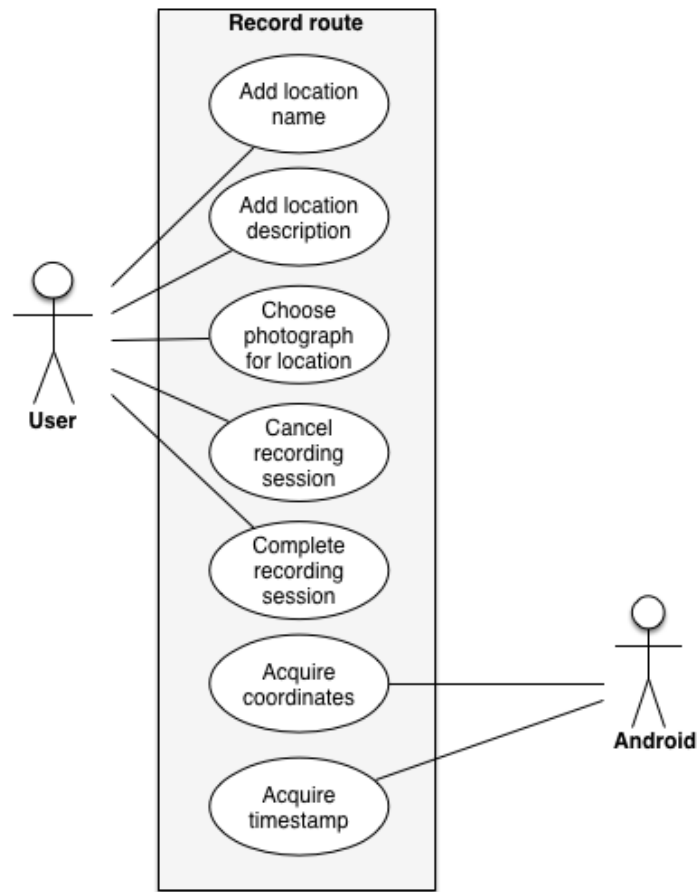


Figure 2: At each location (or waypoint) that the user chooses, they will retrieve their phone and enter the name and a description for the location. A photo can also be chosen from the gallery or taken using the camera - and then added to the waypoint. From this screen, the user can choose to complete the recording session, or cancel it entirely. The Android device will automate the task of fetching GPS coordinates and attaching a timestamp to each entry.

### 5.2.3 Complete Route

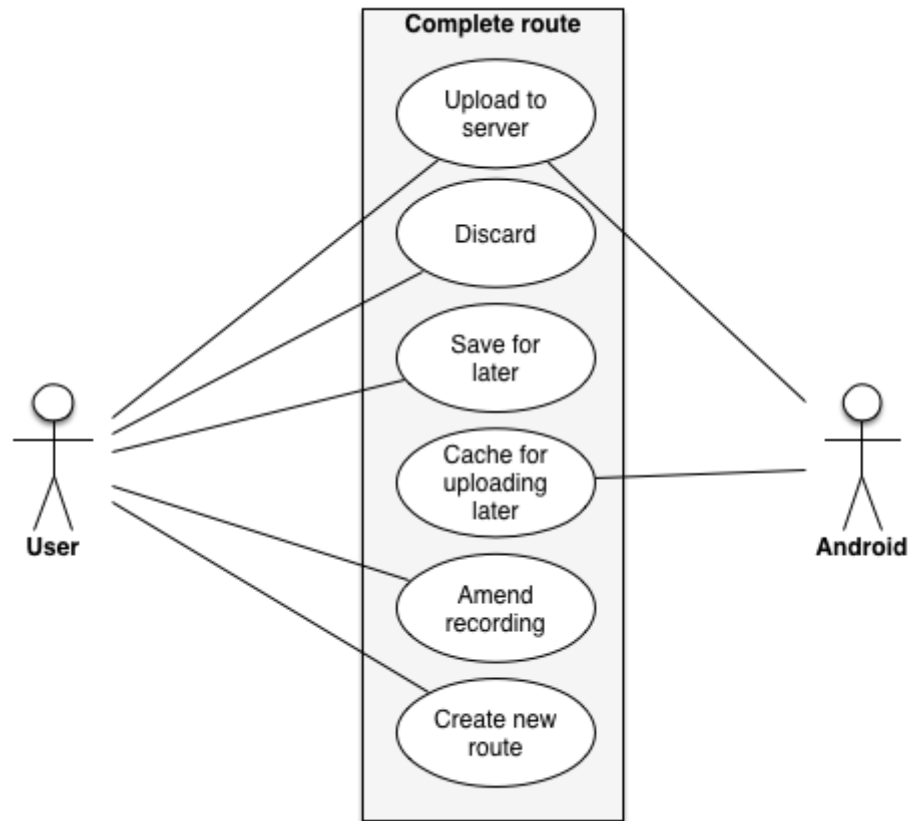


Figure 3: After the user has entered all of the locations that they wish to add and elected to complete the recording, they will be presented with several choices. They can upload the changes to the server (this will be handled by the Android system), saving the walk for later, discarding the walk entirely, or amending the recording - in case a mistake was made. The Android system will also be able to cache the recording for later upload (in the case of no signal, or other upload errors). The user will also be able to create a new route from this screen.

### 5.2.4 System

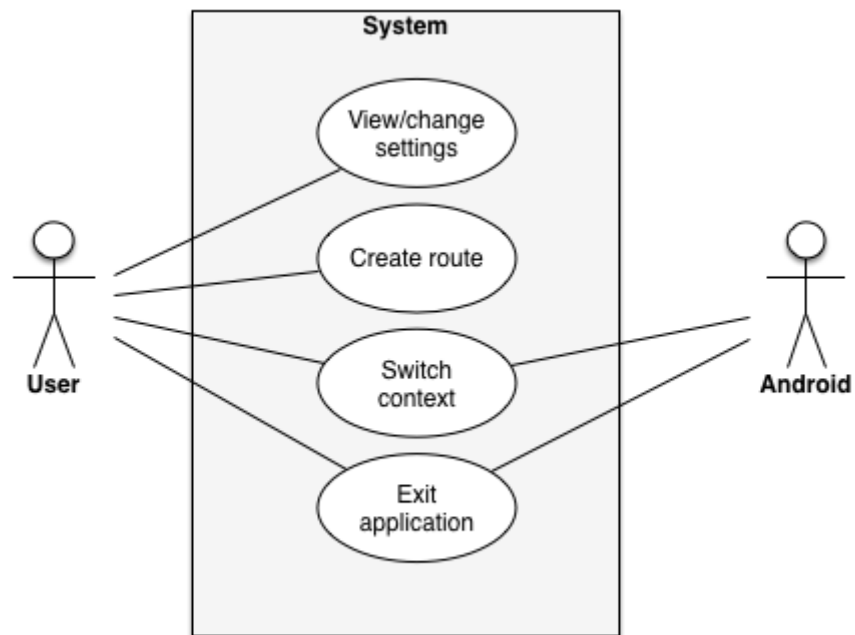


Figure 4: The user will be able to view or change their default settings and create a route from this screen. The application will also be able to handle context switching (i.e. switching to another app) and the application will be able to handle exits safely.

## 5.3 Android - Extended Functionality

### 5.3.1 Create Route

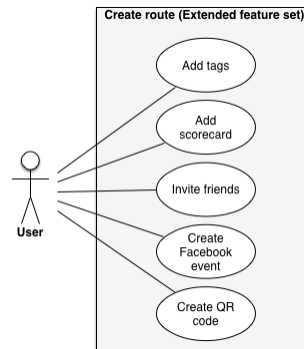


Figure 5: The user will be able to 'tag' their walk(s) with keywords to aid searching on the website. The user will be able to create a Facebook event (through a Facebook app), and invite friends to such an event. Another function available would be to generate a QR code to the Facebook event URL, to help the user share the event effectively.

### 5.3.2 Record Route

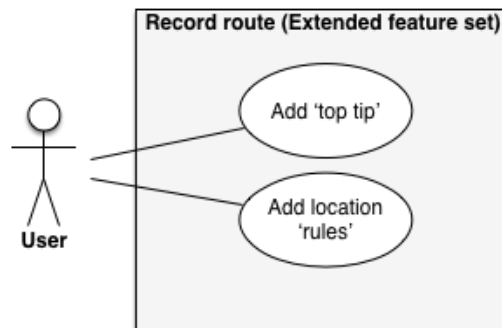


Figure 6: The user will be able to add a 'top tip' for every location; for example, a favourite drink or best place to sit. The user will also be able to choose 'rules' for each location for the followers of the walk - so as to allow the participants of the walk to play a game, or complete a challenge.

### 5.3.3 Complete Route

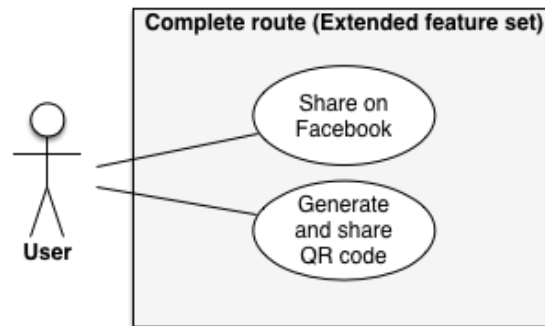


Figure 7: The user will be able to choose to share the create event on Facebook using a Facebook app, as well as sharing the QR code generated earlier.

### 5.3.4 System

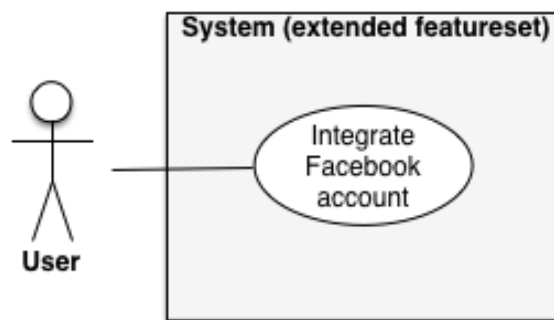


Figure 8: The user will be able to integrate the application with their Facebook account (using a Facebook app and OAuth authentication).

## 5.4 Website - Core Functionality

### 5.4.1 System

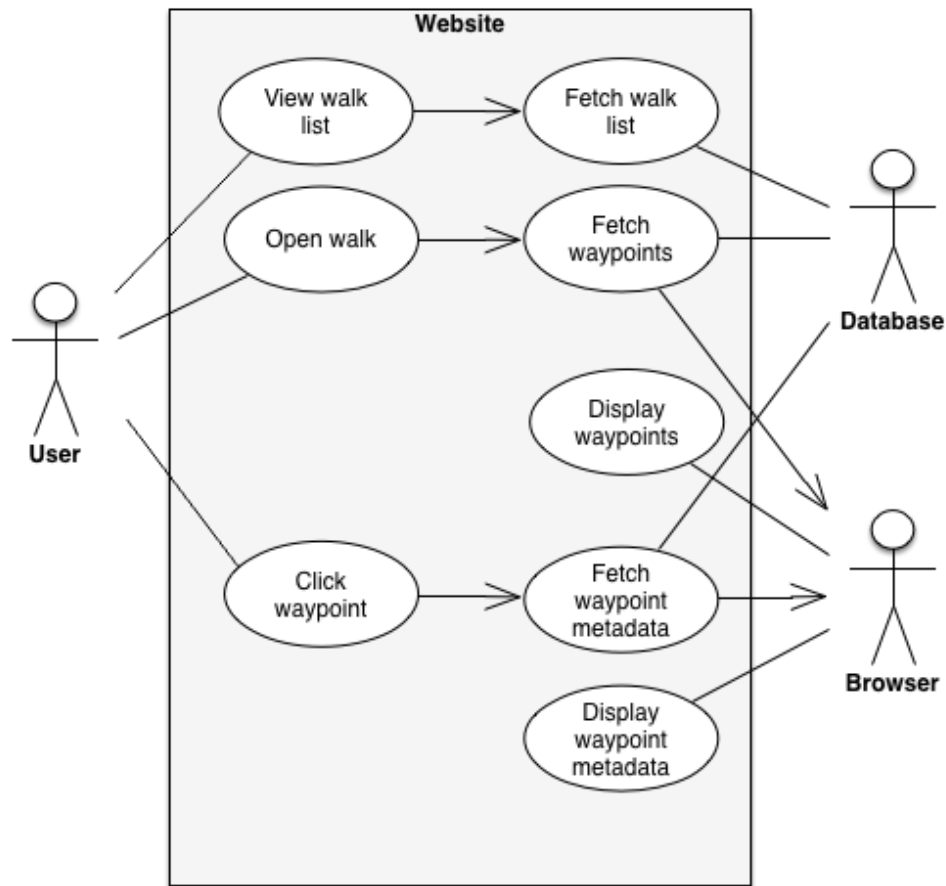


Figure 9: When the page loads, the user will be prompted with a list of walks to load. This will be achieved by the webpage querying the database - the database will provide the information and display it to the user in the browser. The user will be able to open a walk using the web interface, and this will request the walk's information from the database and cause the browser to display the information for each walk on a map. Upon the user clicking a waypoint on the map, the information for each waypoint will be fetched from the database and displayed onscreen.

## 5.5 Website - Extended Functionality

### 5.5.1 System

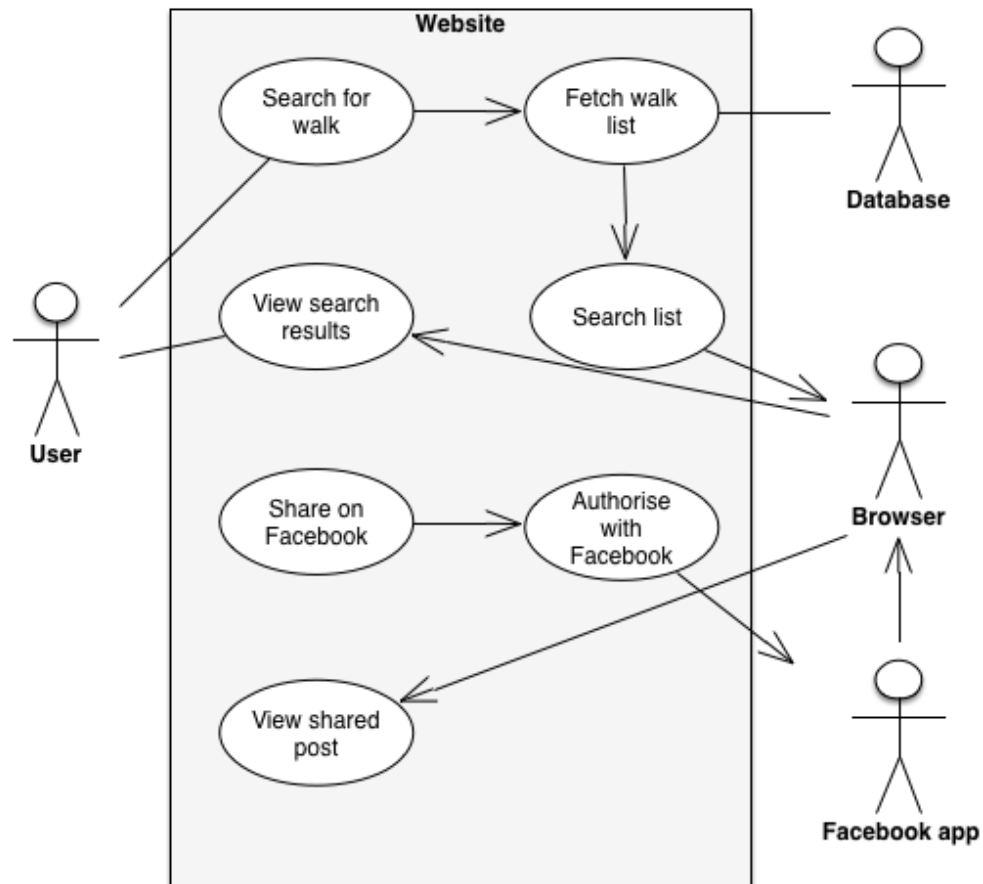


Figure 10: The user will be able to search for walks within the browser, which will search through the database and display the matching walks. The user will also be able to share walks on Facebook using a Facebook app.

## 6 REFERENCES

- [1] PAVLIC, T. Programming Assignment template for L<sup>A</sup>T<sub>E</sub>X.  
<http://www.latextemplates.com/template/programming-coding-assignment>.  
Accessed: Oct 23, 2013.