CS221 Group 04

Project Plan

Authors	msn, eme3, ejc1, stb10, jof2
Config Ref	SE_N04_MAN_PLAN_01
Date	Monday, 31st October 2011
Version	1.0
Status	Released

Address	Department of Computer Science
	Aberystwyth University
	Ceredigion
	SY23 3DB

CS221 Group 04

Project Plan Contents

1 INTRODUCTION

1.1 Purpose of this Document

The purpose of this document is to show how we have translated the client's requirement specification, a pirates-meets-complex mathematics educational application, into a set of basic objectives and milestones.
The document will describe the various interactions between components of the application and the visual appearance of the user interface that will be displayed to the end user.
This document will also provide a Gantt chart showing the project's major tasks and associated milestones along with a probabilistic risk analysis, describing the main issues that may be encountered through the development of the systems and how those risks can be mitigated.

1.2 Scope

This document should take in to account the details of the group project.
This document includes an overview of the proposed system, including our choice of platforms, high-level architecture and a description of target users. The document also contains; a use case diagram giving an overview of how the system's various 'actors' will be expected to interact with each other, screenshots and a description of the user interface design and how the application will interact with the end user; a Gantt chart which displays the start and end dates for the main tasks of the project and finally, a risk analysis which highlights possible issues the project may encounter.

The document is to be read by the client to ensure all of their requirements have been correctly translated by the team.

1.3 Objectives

The main objectives are to show our initial plan, including user interface design, user interaction, system interaction, a timeline of all major objectives and any problems that are suspect to occur during the process. The goals of this document are as follows;

1.3.1 Provide an overview of the proposed system, what technologies it will use and who it is for,
1.3.2 Display the major components of the system and how they will interact with each other,
1.3.3 Display a basic model of how the user interface will look and interact with the end user,
1.3.4 Provide a description of the project's main milestones and,
1.3.5 Identify possible issues with the development of the system and how to mitigate them.

2 OVERVIEW

Our proposed system is an Android based, social, treasure hunting game, designed to teach children about complex numbers and improve their general mathematics skills.

2.1 Platforms and High Level Architecture

We intend to use the following platforms for the system;

2.1.1 Android

The client stated specifically that the system is to be deployed to Android phones, we will be using the Android platform.

2.1.2 PHP

On the server side, to handle communication between the phone and the database server.
PHP is available on a vast majority of servers and is easy to develop for when compared to other systems.
PHP is also taught to second year students, so the opportunity to gain skill through a real application is valuable.

2.1.3 MySQL

MySQL is one of the most widely used database platforms and will be used to store all application user information.
The platform has simple command sets and is very quick to deploy and manage.
phpMyAdmin will also be installed to allow database management via a graphical interface.

The high level architecture consists of the following elements and game mechanics;

2.1.4 Map

The map will be displayed using the Google Maps API as the universally understood motions of scrolling and zooming are readily available to use via this API. If desired by the client, we can then tile over the world map with sand and other images to give the map a more "Piratey" appearance.
The map will show clues, users and other points of interest. The user's total treasure and the current clue is also on display.

2.1.5 Clues and Treasure

The game orientates around the concept of clues and treasure.
Users will find clues that need to be deciphered (using the basics of complex numbers) to find treasure.
Users are also able to place their own treasure, and a clue to that treasure will be generated for other users to find.
To avoid unfair play, the distance from which the user can place a clue to treasure they have buried will be limited.
On the map, a scroll icon will indicate a clue has been buried.
On the map, a "X" icon will indicate where treasure has been buried, providing the user has the clue for that treasure.
The phone's vibrate function will give a vague indication as to where the treasure is, by adjusting vibration feedback.

When a clue is found it is viewable at the bottom of the screen until it is solved.
Additionally, a line from the user to the clue will be drawn should they wish to return to the point they found it.

2.1.6 Internet Connectivity

Users require connection to the game server to allow them to bury and dig treasure.
Upon loss of connectivity, the user will be informed that game actions will not be possible until they reestablish connection.

2.1.7 Menu

Within the game map screen, a menu is made available to the user via the Android "Settings/Menu" button.
This provides access to four options;

Mateys - Display all User's Friends, ordered by treasure total.
Treasure - Display User's Hoard and each item of buried treasure.
Help - Offers a help screen or tutorial.
Quit - Prompt the user to confirm log out and quit the application.

2.1.8. Login and Avatar

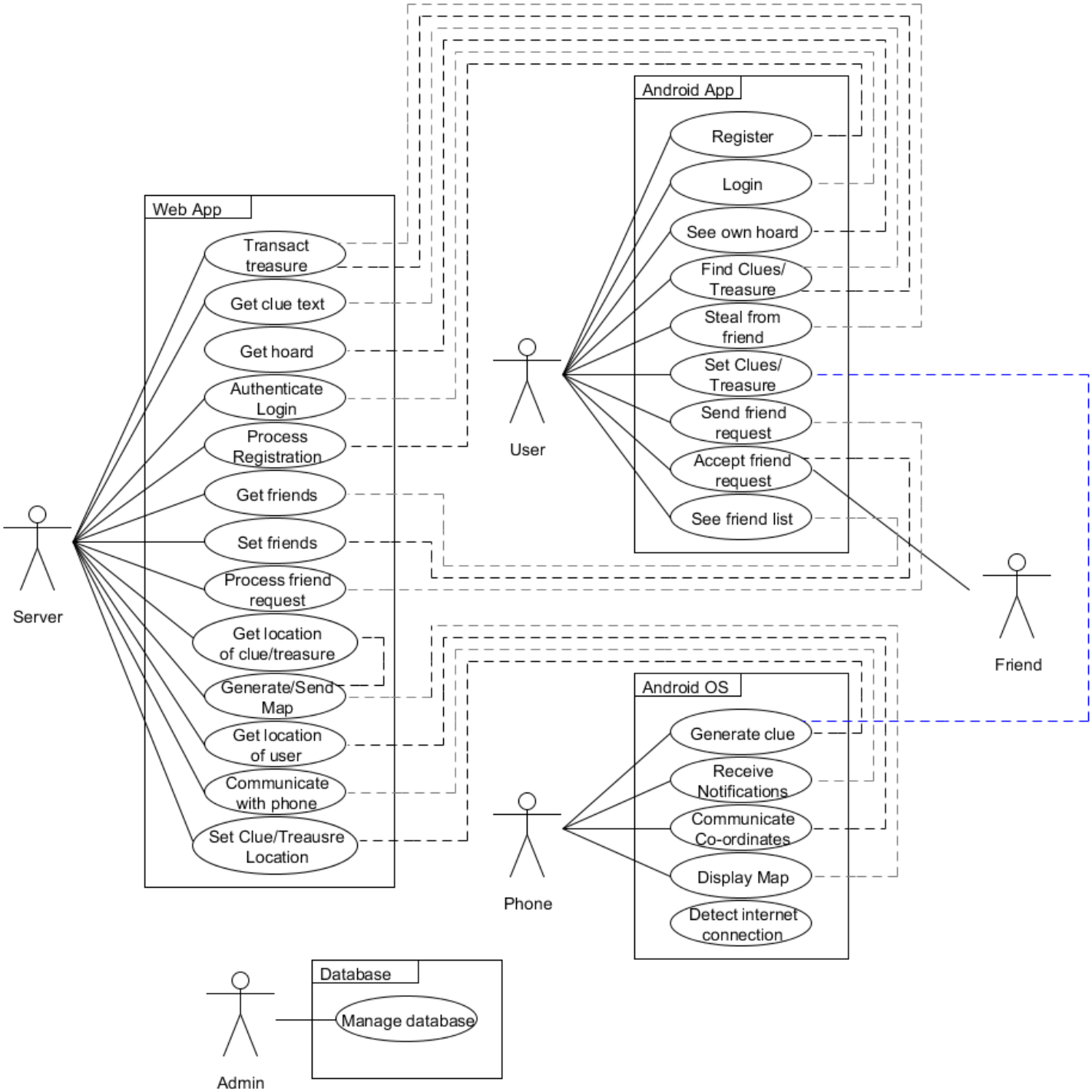
User login data will be stored on the database server, credentials are required to log in to the account via the app.
Upon registration, a User can pick from a selection of default avatars to represent their pirate character.

2.2 Description of Target Users

As defined in the requirement specification given to us by the client, our proposed system will be aimed at young, primary school children, with the approximate age of 10 years. Whilst designing the user interface for the system, we have to be careful that the application is not only simple to use to avoid frustration and confusion, but it must also be interesting and lively to engage users.

3 USE CASE

3.1 Use Case Diagram



3.2 Use Case Descriptions

USER / FRIEND	
Register	User will be able to register for a game account via the Android app.
Login	User should be able to log in to the application providing valid, registered credentials.
See own hoard	User will be able to see how much treasure they have in both their hoard and at buried locations
Find clues/treasure	User must be able to find clues on the map, read those clues, and find the treasure those clues lead to.
Steal from friend	Users must be able to steal treasure from a Friend if they appear within a certain distance on the map.
Set clues/treasure	A User must be able to set both treasure and clues to that treasure at locations on the map.
Send friend request	The User must be able to send a friend request given the entry of another User's e-mail address.
Accept friend request	A Friend of that User must be able to receive and accept an incoming friend request.
See friend list	User must be able to see a list of their Friends, ordered by how much treasure they hold.

ADMINISTRATOR	
Manage Database	Administrator must be able to log in to the phpMyAdmin facility and manage the MySQL database.

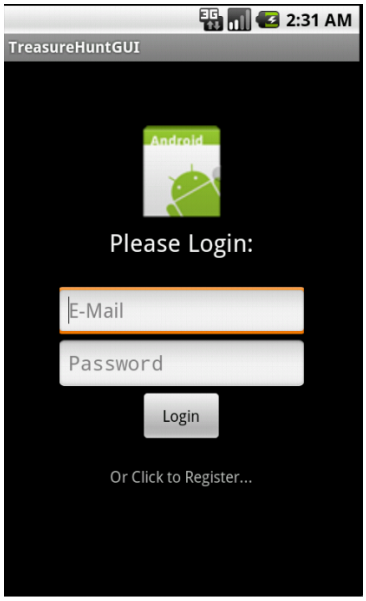
PHONE	
Generate Clue	The User's Phone must be able to generate the clue for a User to set a clue location.
Receive Notifications	The User's Phone will receive notifications from the Server, such as the location of clues and treasure, or that some of their treasure has been stolen by a Friend, as well as friend request notifications.
Communicate Co-Ordinates	The Phone will send the current GPS co-ordinates of the User to the Server, to detect which clues, treasure burial spots and Friends are nearby. This feature is paramount to the executing of the application.
Display Map	The Phone will display the world map to the User with any points of interest marked with the necessary symbols, for example the User's current location should display a pointer or avatar.
Detect Connectivity	The Phone will check for Internet and GPS connectivity, if either is lost, a message should be displayed to the User.

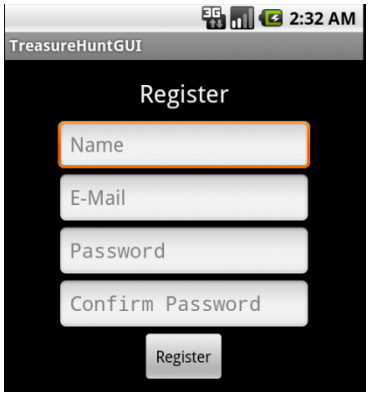
SERVER	
Transact Treasure	Server must be able to transfer treasure from one User to another in the event of a steal operation. Server must be able to transfer treasure hidden on the game map to the User who digs it up.
Get Clue Text	Server will need to send the text of a clue a User has found to the User's Phone for display.
Get Hoard	The Server will send info about the User's treasure in his or her hoard and buried locations for display.
Authenticate Login	The Server will allow a User to log in to the application via their Phone providing valid credentials. This will check the database for the existence of the User and the validity of their password.
Process Registration	Server will be able to add a new User to the database after a new User enters their data via Phone.
Get Friends	Server will send information about a User's friends to the User's Phone for display.
Set Friends	Following the acceptance of a friend request, the Server will set two User's as Friends.
Process Friend Request	Server must be able to send a friend request from one User to another, given a valid e-mail address.
Get Clue/Treasure	The Server must be able to make database requests to find the locations of clues and treasure. Information about these map features will be sent to a User's Phone if they are within distance.
Generate/Send Map	Server must be able to fetch the world map (possible via Google Servers and Maps API). Server will have to send the locations of map features to a User's Phone when they are in range.
Get Location of User	Server must be able to request the location of a playing User, via their Phone.
Communicate with Phone	Server must be able to send notifications to and receive location data and commands from the Phone of a currently playing User.
Set Clue/Treasure	Server will enable User's to set clues and treasure locations via the interface on their Phone. The Server will need to add these clues and treasure locations to the database for retrieval.

4 UI DESIGN

The following images cover the initial concept of the application.
Note that placeholder Android logo images are placed wherever a sprite is to be designed.
These mockups were created in the Android SDK, allowing for re-use later when coding and further design begins.

4.1 Login and Registration



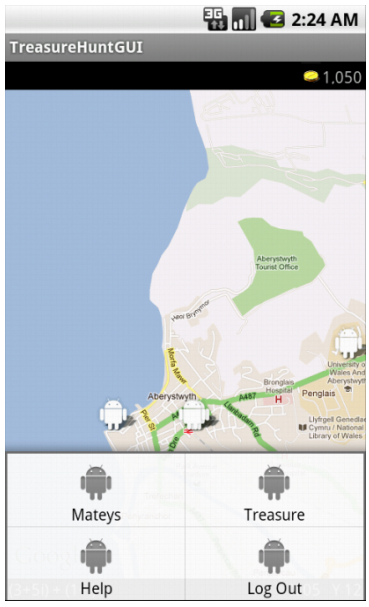


The login screen is the first view to be presented to the application user, inviting them to access the game by entering credentials for the e-mail address and password fields that are to be validated by the server upon the push of the Login button.

New users are invited to click the link below which will take them to a rudimentary Registration view. Here they will be able to fill in the data required for them to gain an account such as Name, E-mail address and password.
Further fields may be added if required by further design development or client request.
Provisions for an avatar selection screen have also been made, where the user is prompted to pick an image from a series of stock pirate theme images.
Clicking the Register button will send the information entered to the server to be added to the database.

The spec makes no requirement for a "Remember Me" or Forgotten Password feature. The spec also makes no requirement for validating a registrants e-mail address, therefore, upon registration, the game's main map will load.

4.2 Main Map View



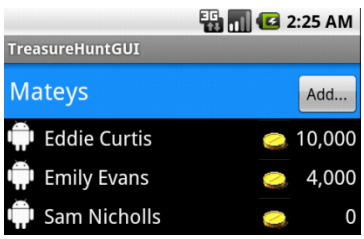
The main map screen is where all the game mechanics will be displayed, using the Settings/Menu button on the Android device will also launch a menu to allow the user to enter the application's two other views, detailed below.
Help and log out are also reachable via this menu.

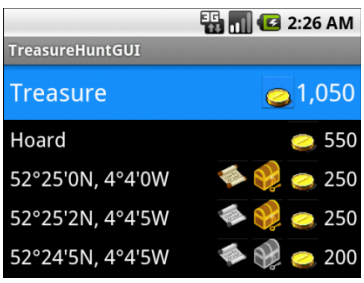
Note here the use of the Google Maps API. If the user so wishes, we can also tile over the map with sand/island imagery to enhance the game's theme.

The map view will show the user's current location on the world map, any nearby clues or friends and if they are within distance, the treasure for their current clue.

The top right of the application displays their current full treasure total, whilst the bar on the bottom displays the current clue and how many X and Y units they are away from the origin of the clue.

4.3 Mateys and Treasure View



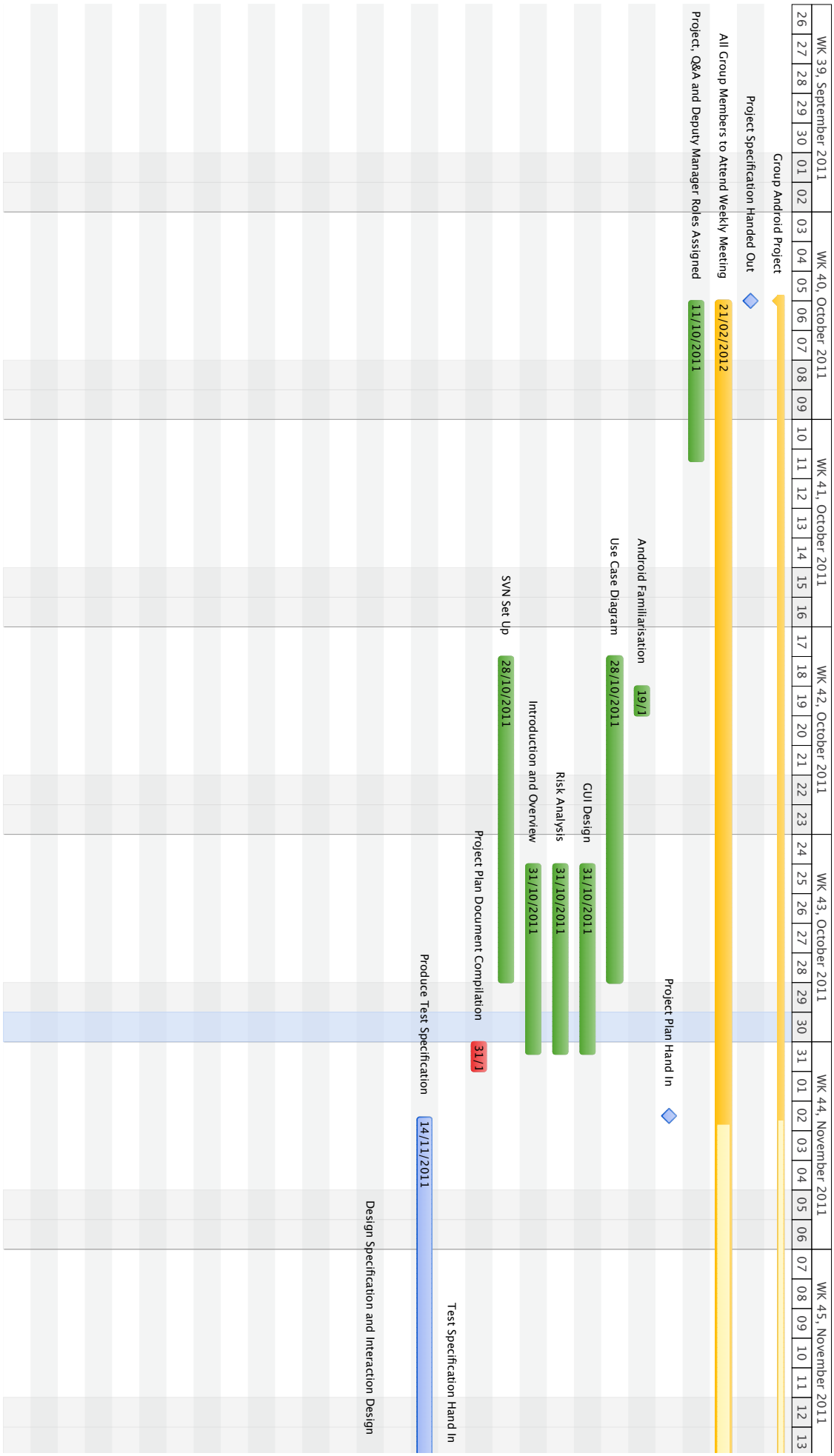


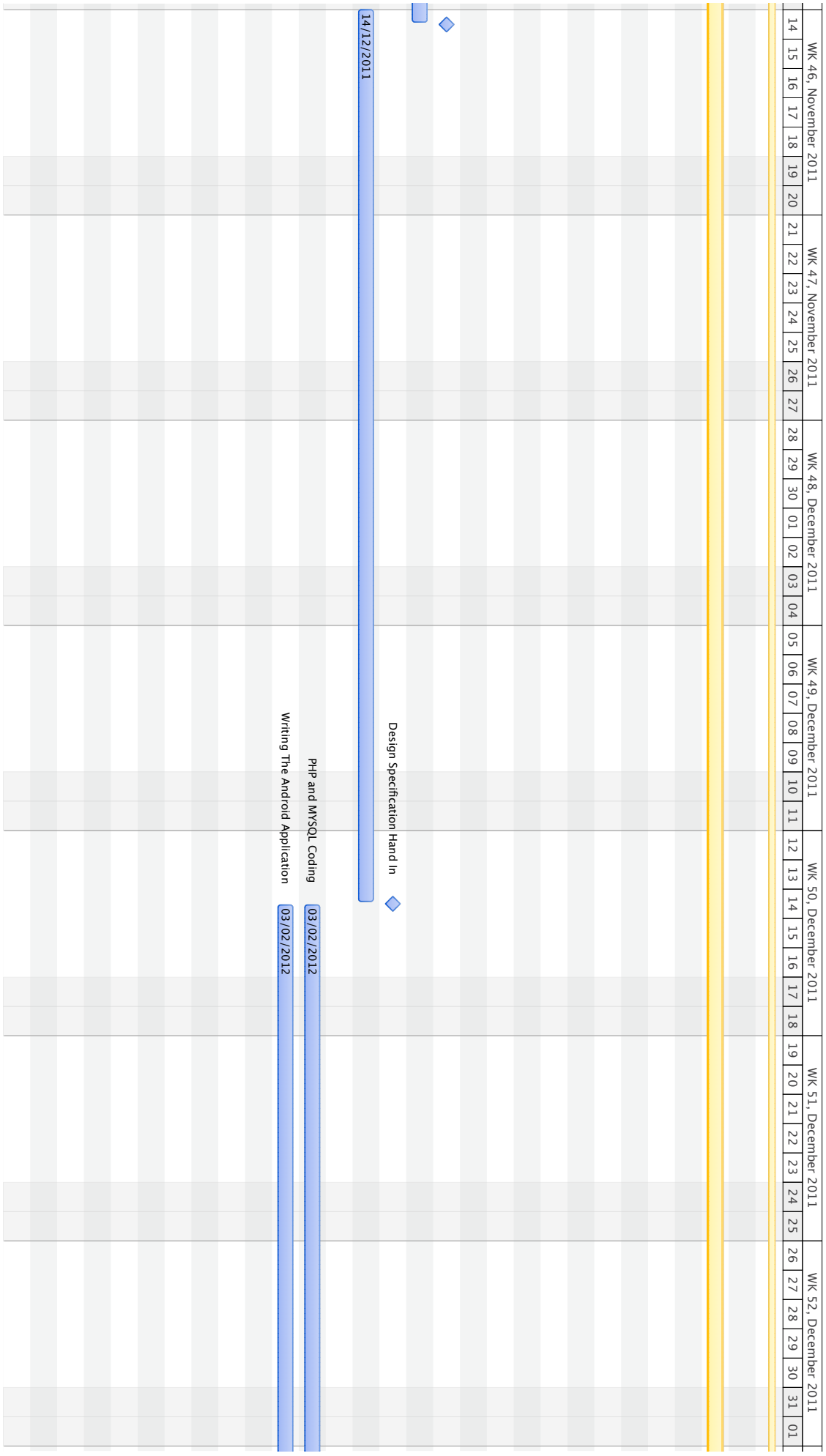
Two other views, Mateys and Treasure are accessible from the menu on the main map view.

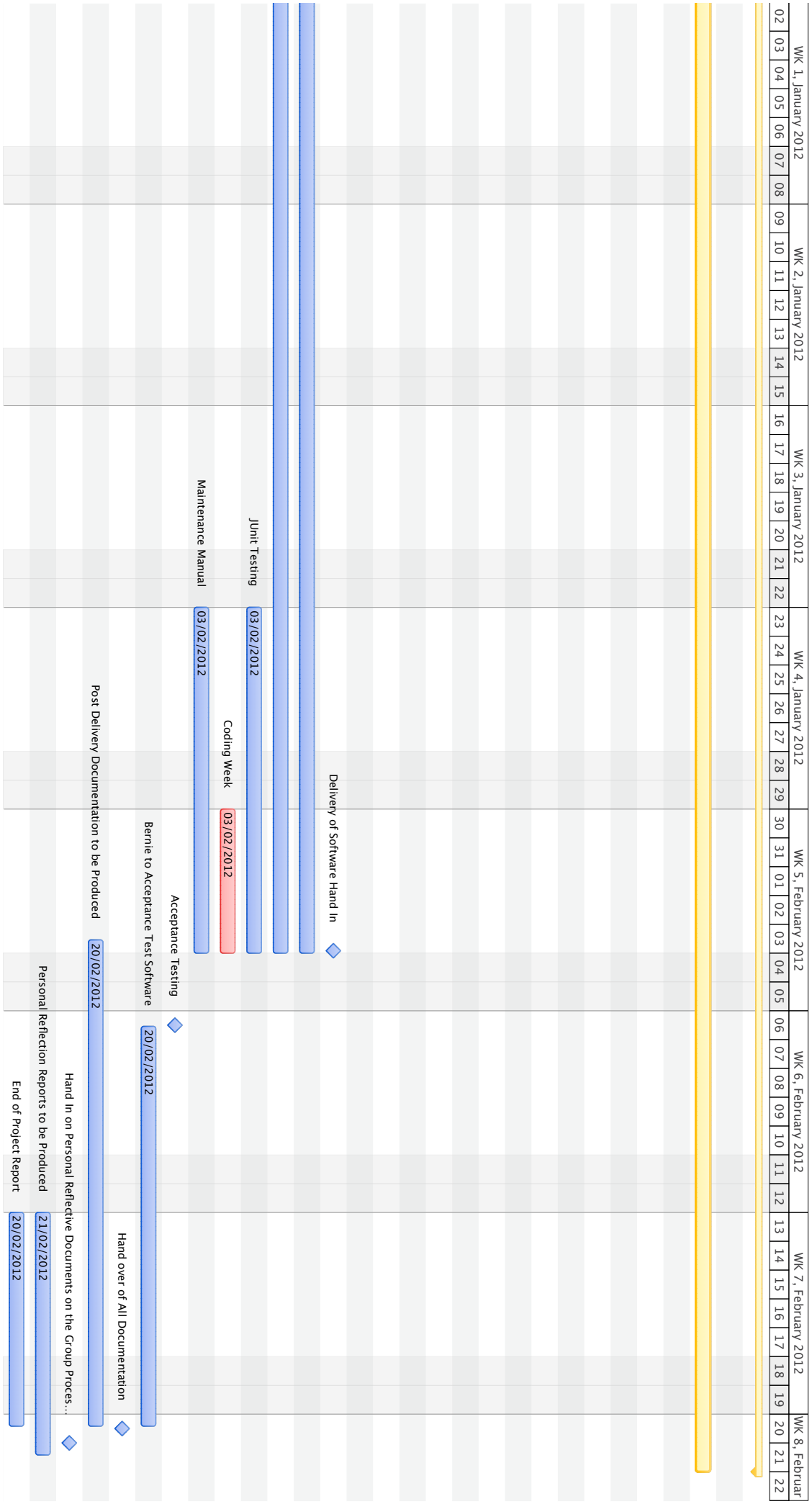
Mateys will display the user's list of friends in order of their total hoard, also displaying their avatar if one is set.

Treasure will display the user's current hoard and the locations of their buried treasure.
Icons indicate whether the clue or treasure has been found. The scroll icon represents the clue, if it is greyed out, another user has found it, in the same manner, the treasure chest represents the treasure itself, if this is greyed out, it has been found by another user and is no longer included in the user's total. It will remain on the list until a number of days pass.

5 GANTT CHART




WK 46, November 2011
14 15 16 17 18 19 20
WK 47, November 2011
21 22 23 24 25 26 27
WK 48, December 2011
28 29 30 01 02 03 04
WK 49, December 2011
05 06 07 08 09 10 11
WK 50, December 2011
12 13 14 15 16 17 18
WK 51, December 2011
19 20 21 22 23 24 25
WK 52, December 2011
26 27 28 29 30 31 01
14/12/2011
Design Specification Hand In
PHP and MYSQL Coding
03/02/2012
Writing The Android Application
03/02/2012


WK 1, January 2012
WK 2, January 2012
WK 3, January 2012
WK 4, January 2012
WK 5, February 2012
WK 6, February 2012
WK 7, February 2012
WK 8, Februar
Delivery of Software Hand In
JUnit Testing
03/02/2012
Coding Week
03/02/2012
Maintenance Manual
03/02/2012
Acceptance Testing
Bernie to Acceptance Test Software
20/02/2012
Post Delivery Documentation to be Produced
20/02/2012
Hand over of All Documentation
Hand In on Personal Reflective Documents on the Group Proces...
Personal Reflection Reports to be Produced
21/02/2012
End of Project Report
20/02/2012

6 RISK ANALYSIS

Group 04	**Project** "Treasure Hunt" (WT) **Client** Bernie Tiddeman (bpt)

PROJECT PROBABILISTIC RISK ASSESSMENT

Ongoing Risks Risk Event	L	M	Risk	Mitigation
Team member absence	0.6	0.3	**0.18**	All meetings are announced with plenty of time so being unaware of a meeting is not an excuse. All absences require an apology to be made and the missing team member is expected to read the minutes of any missed meetings to be up to speed at the following meeting. Unreliable team members will be warned then carded.
Project Leader absence	0.6	0.4	**0.24**	Meeting to continue as normal, chaired by the Deputy Project Leader.
QA Manager absence	0.6	0.4	**0.24**	Meeting to continue as normal, with QA questions and decisions to be made by the Deputy QA Leader.
Lose team member contact	0.3	0.7	**0.21**	Highly important for team members to stay in touch with the rest of the team via the group e-mail. Unreliable team members will be warned then carded.
SVN Repository Downtime	0.3	0.9	**0.27**	Team to be notified of changes to documents and files via the group e-mail. Team members are not to write to the same document or source file until diffs can be made when the SVN service returns to normal. Manual diffs can be made, but will require QA approval.
SVN Repository Failure	0.3	1.0	**0.30**	All work should be backed up at an alternate source to prevent work loss from any incorrect commands issued to the SVN repository system or the failure of the server that provides the service.
Illness or other unexpected circumstance	0.5	0.9	**0.45**	Team to be notified as soon as possible of any major illness or unexpected circumstances that would prevent work being completed or submitted so that the client and manager can also be notified in due time. Team members who do not notify the team of issues will be warned and carded.

Documentation Risk Event	L	M	Risk	Mitigation
Individual parts of document submitted late by team member	0.4	0.8	**0.32**	Deadlines for documentation to be set earlier than the officially mandated hand-in time to allow for any team member who may run in to issues to be supported. Team members are to alert the group e-mail as soon as an issue is suspected.
Individual parts of document incomplete or of low quality	0.4	1.0	**0.4**	Documents to be checked by entire team and verified by at least one QA authority before submission. Team members are expected to ask for help via the group e-mail if stuck or confused. Unreliable team members will be carded.
Human errors	0.6	0.2	**0.12**	Documents to be checked by entire team and verified by at least one QA authority before submission. Team members expected to check over their own work and use any tools necessary to validate quality.

Software Development and Delivery Risk Event	L	M	Risk	Mitigation
Slipping from Project Timeline	0.5	0.9	**0.45**	Coding to begin as soon as possible, iterating early and often. Programming and web teams to keep in contact with QA manager and Project Leader. If an issue arises, the rest of the team is to be informed immediately to begin further mitigation.
Parts of required implementation missing or incomplete	0.4	1.0	**0.4**	Project Leader to ensure tasks exist for all objectives required by the specification. Constant unit testing to ensure all functions are implemented. QA manger to ensure all work is satisfactory.
Feature Creep	0.6	0.8	**0.48**	Programming and web teams to ensure they keep to the tasks defined by the Project Leader. QA manager to ensure objectives are met and to keep teams in check. Communication between project team, client and manager must be well documented.
Implementation not working as expected by client	0.5	0.9	**0.45**	Communication between project team, client and manager must be well documented. Client to be kept up to date with project as much as possible to prevent expectation change. Testing to be a paramount part of software development, consider writing tests before code.
Client requirements change	0.5	0.9	**0.45**	Keep regular contact with client, do not be afraid to say things cannot be done with the given time.

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

REFERENCES

N/A.

DOCUMENT HISTORY

Version	CCF#	Date	Change Description	Changed By
1.0	N/A	2011-10-31	Compiled first release of Project Plan.	MSN