Cadence Health

Project Plan

Mobile Application

Version 1.1, 07/10/14

Contents

1. Statement of Purpose and Scope 3

2. Development Risks and Management 4

2.1. Risk Management 4

2.2. Risk Identification 5

2.3. Risk Analysis 5

2.4. Risk Planning 6

2.5. Risk Monitoring 8

2.6. Change Log 9

3. Schedule 9

3.1. Work Breakdown 9

3.2. Gantt Chart 10

4. Project Resources 12

4.1. Group Resources 12

4.2. Hardware 12

4.3. Software 13

5. Organisation 13

5.1. Group Roles and Responsibilities 13

5.2. Reviews, Audits and Testing 14

6. Tracking 14

6.1. Version Control 14

6.2. Tracking Progress 15

6.3. Communication 15

6.4. Quality Control 15

# Statement of Purpose and Scope

The purpose of this project plan is to provide a detailed document of the project expectations. It provides an overview of the project to help identify the objectives, analyse parameters, provide task allocation, and develop schedules.

The scope of this report is to present a detailed project plan of the design and development of the healthy food application for Cadence Health. The task is to provide the client an application for the smartphone market. Currently dieting has been used by many people as a means to lose weight. Research has shown that dieting yields negative results on your overall health in the long run. As a result the goal of Cadence Health is to start a shift away from dieting and calorie counting, to move towards healthier eating.

The project is being undertaken to raise awareness of this issue and promote the shift to a healthier lifestyle. This will be achieved over the use of functions and tools available through the application. This allows users to view their meals, ingredients and view an ORAC rating to help maintain a healthy lifestyle.

In order for project to be successful, the project plan will cover five categories:

- Risk Planning and Management: Identifying the risk and providing a risk rating through the use of a risk matrix. Providing a thorough planning of risk with the aim to mitigate.

- Project Schedule: Outlines the tasks that need to be completed, which resources will be used to complete the project and the timeframes in which the tasks need to done.

- Project Resource: Detailing of the available resources for the project, this includes the people, hardware and software

- Organisation Roles: Detailing the allocated roles and responsibilities of the stakeholders of the project.

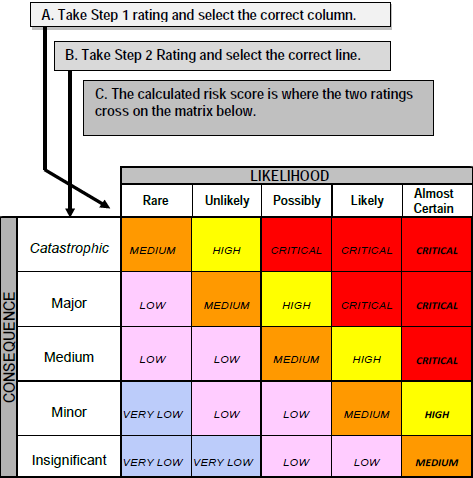
- Project Tracking: This includes how the project will be managed and viewed in regards to ensuring and maintaining a quality application.

# Development Risks and Management

# Risk Management

We are applying the four step process to the Risk Management strategy, which is often the go to for organisations, both large and small. The four steps are Identify, Analyse, Plan, and Monitor. Each step is taken for a specific purpose to ensure the risks to the project and the client’s organisation are mitigated as best as possible.

The Risk Matrix below shows the ratings given to risks, for both the Likelihood and possible Consequences. The aim of our risk management is to have all risks moved to as low a rating as possible. The risks with the highest ratings, even after their mitigations should be monitored carefully, and if possible eliminated completely.



# Risk Identification

Below are the risks that are associated with the project. They may change as the project evolves, so a change log will also be included to update these as needed.

Technology risks

1. Version controls somehow fail and work is lost.
2. Software resources do not function as expected
3. Competitor releases similar application
4. Phone breaking as result of testing

People Risks

1. Loose contact with team members
2. Loose contact with project sponsor
3. Team members cannot attend meetings
4. Conflict between team members
5. Conflict between team and sponsor

Organisational Risks

1. Unable to meet at normal location
2. Other subjects require work to be conducted at same time
3. Loss of phone during testing
4. Test users may not meet when needed

Estimation Risks

1. Estimated times needed are too short
2. Workload underestimated

# Risk Analysis

The risk analysis table below is a result of looking carefully at the risks to work out the Likelihood of the risk and the worst case Impact. From this it is then given a rating, which is dependent on Likelihood and the potential impact.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Risk no. | Risk Description | Likelihood | Impact | Rating |
| 1 | Version controls somehow fail and work is lost | Rare | Catastrophic | Medium |
| 2 | Software resources do not function as expected | Unlikely | Major | Medium |
| 3 | Competitor releases similar application | Rare | Medium | Low |
| 4 | Phone breaking as result of testing | Unlikely | Catastrophic | Medium |
| 5 | Loose contact with team members | Rare | Major | Low |
| 6 | Loose contact with project sponsor | Rare | Catastrophic | Medium |
| 7 | Team members cannot attend meetings | Possible | Medium | Medium |
| 8 | Conflict between team members | Possible | Major | High |
| 9 | Conflict between team and sponsor | Possible | Major | High |
| 10 | Unable to meet at normal location | Possible | Minor | Low |
| 11 | Other subjects require work to be conducted at same time | Likely | Minor | Medium |
| 12 | Loss of phone during testing | Rare | Catastrophic | Medium |
| 13 | Test users may not meet when needed | Possible | Minor | High |
| 14 | Estimated times needed are too short | Possible | Major | High |
| 15 | Workload underestimated | Possible | Major | High |

# Risk Planning

The risk strategy table below is ranked in order of rating. The aim is to mitigate each risk as much as possible, so that the rating is reduced to an acceptable level. For it to be acceptable the rating must be below no higher than Medium, except in the rare case a high rating may be allowed, but must be more carefully monitored.

|  |  |  |  |
| --- | --- | --- | --- |
| Risk no. | Initial rating | Strategy | Rating after mitigation |
| 1 | Medium | Have multiple backups saved in different locations | Low |
| 2 | Medium | Try multiple software vendors, so we can test if they work or not | low |
| 3 | Low | Keep all information within team, and do not share IP | Medium |
| 4 | Medium | Thoroughly test application on emulator before real device testing | low |
| 5 | Low | Have multiple ways to communicate, e.g social media, email etc. | low |
| 6 | Medium | Work out issues before they become major. If needed work with staff to resolve the problem | low |
| 7 | Medium | Arrange multiple meetings/ times. Make sure team can make at least one of them per week | low |
| 8 | High | Resolve situations within team, if not able to solve consult staff for more help and to notify them about the situation | Medium |
| 9 | High | Work with sponsor as much as possible, and continuously communicate with them to ensure things are working | Low |
| 10 | Low | Be flexible with meeting locations and times | Low |
| 11 | Medium | Schedule work to be done bit by bit, as to avoid large amounts of work piling up | low |
| 12 | Medium | Ensure phone is used for testing in secure area. Do not leave it unattended, and put someone in charge of it | low |
| 13 | High | Be flexible with meeting times and locations and have a best fit approach to meetings | low |
| 14 | High | Carefully work out times needed. Update and keep track of time needed to complete tasks and update as you go | medium |
| 15 | High | Share workload, and if needed take up slack as needed. Track all work done and update timesheets to gain better overview of project | medium |

# Risk Monitoring

There are a number of indicators that will help see and monitor each risk. If these are monitored carefully and updated, the risks can be managed more easily and better mitigated, where possible.

All risks will be monitored closely and checked on regularly, but risks with a residual rating of medium will have special attention paid to them. Some of the risks cannot be monitored easily due to their nature, but the mitigations will allow the team to better plan for them and have a better way to react if they eventuate.

|  |  |
| --- | --- |
| Risk type | Indicators |
| Technology | * Computers are not able to cope with workload * Minimum requirements aren’t met for certain applications * Slow or poor performance |
| People | * Low moral * Conflict over trivial issues * Low communication |
| Organisational | * Feedback is poor initially * Feedback does not change, even after faults are fixed |
| Estimation | * Time taken always more than time estimated * Schedule is always pushed back, without prior need |

# Change Log

|  |  |  |  |
| --- | --- | --- | --- |
| Risk no. | Description/ Initial rating | Strategy | Rating after mitigation |
| 16 |  |  |  |
| 17 |  |  |  |

# Schedule

This section outlines each of the tasks that are to be completed for the application to be finished. The schedule also outlines the approximate start and end dates of each task.

# Work Breakdown

Deliverable 1: Feasibility Study (Monday 18/08/14)

Deliverable 2: Project Plan and Requirements Document (Monday 08/09/14)

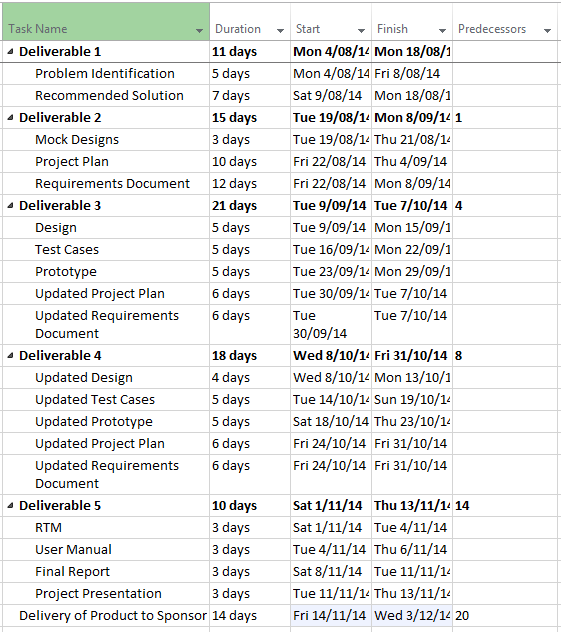
Deliverable 3: Updated Project Plan, Updated Requirements Document, Design, Test Cases, Prototype (Tuesday 07/10/14)

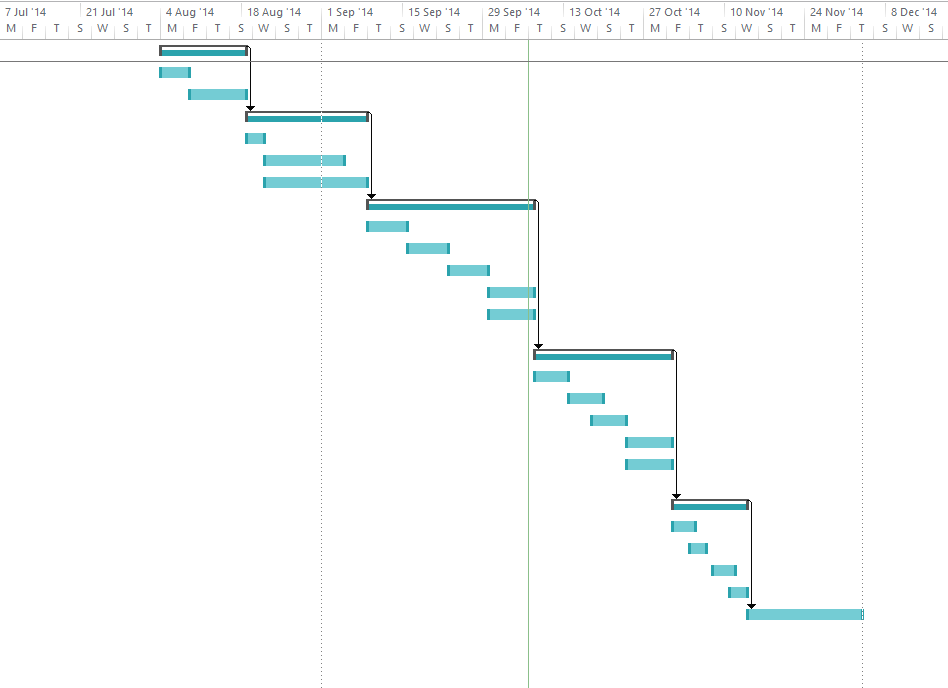
Deliverable 4: Updated Project Plan, Updated Requirements Document, Updated Design, Updated Test Cases, Updated Prototype (Friday 31/10/14)

Deliverable 5: Project Presentation and Software Demonstration, RTM, User Manual and Final Report (Thursday 13/11/14)

Final Deliverable: Delivery of Product to Sponsor (Week 13 or 14)

# Gantt Chart





# Project Resources

# Group Resources

Every member in the team plays a key part and everyone has input to the project, both as a whole and individually. The key person for the success of the project is the Sponsor Leanne, who has ultimate say on whether or not the project is successful. She also has the special skills with nutrition and health science, which none of the team members have.

The team also has other students, Lecturers and other university staff, whose knowledge may be of benefit to the team.

Everyone in the team has the same resources in terms of hardware and software, and will use them to do their individual parts of the project and it as a whole.

# Hardware

The following hardware will be used during the documentation and creation of the application.

Desktop PC- used to work at both home and university, to do both documentation and coding. The specifications of each team member’s computers vary, but are a combination of the latest version of Windows and Apple’s OS X.

Laptop - Used to do work during meetings and again work on the writing the application itself and its documentation

Android Smartphone - used to test prototypes of the application for bugs and issues not seen in the emulator.

For deploying the application there will be additional hardware needed. To start with the application’s back end can be run locally on a desktop PC. But once it is deployed there must be a backend that is available to use from any location. It must be able to cope with demand and be efficient so that the application is smooth and fluid.

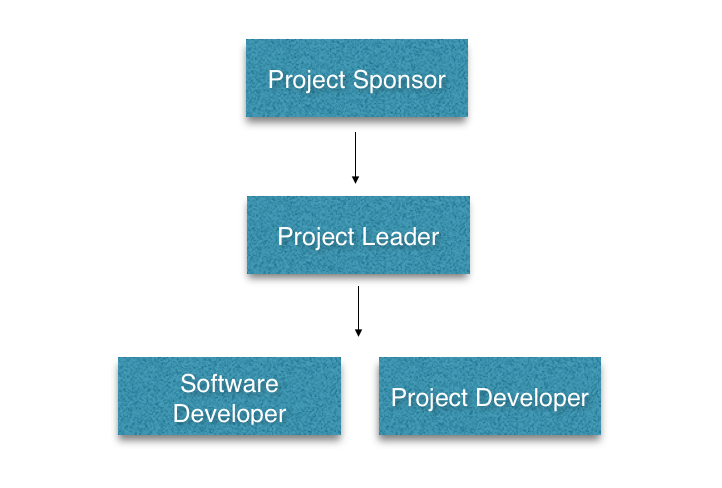
# Software

The following software will be used to develop and prototype the application:

* Microsoft Windows (7, 8, 8.1)
* OS X Mavericks (10.9)
* Titanium and it’s development environment
* Github
* Alloy
* Genymotion Android emulator
* Microsoft project
* Enterprise architect
* Sybase Power designer

# Organisation

# Group Roles and Responsibilities



Project Sponsor - Highest level of authority over the project

Project Leader - Brings all components together

Project Developer - Document and design application

Software Developer - Program application

|  |  |
| --- | --- |
| Role: | Responsibilities: |
| Project Sponsor  (Cadence Health: Leanne) | * Ultimate authority over the projects direction |
| Project Leader | * Final review and delivery of deliverables * Quality control |
| Project Developer | * Contribute to deliverables * Participate in team meetings * Gather requirements * Document application * Test application |
| Software Developer | * Model application * Code application * Participate in team meetings * Gather requirements * Test application |

# Reviews, Audits and Testing

Specifications for testing are to be outlined in the proceeding deliverable where a separate document solely dedicated to test specifications shall be produced.

Group meetings are held twice a week where each team member is responsible for reviewing and auditing other team members project work. No project work is presented to the client without first being reviewed by the entire team, then once all work is finalised the team leader shall oversee a final review.

# Tracking

Tracking for this project will be primarily achieved through the use of Github. Github allows project members to share project resources and documents between each other.

# Version Control

Github provides an interface to a version control system of our project documents and the application source code. Github shall be used as the main repository for all project related files.

Version control is very important in any project, and especially so for software projects. As it is important to contain the ability to revert to previous versions of the application should a problem arise.

All project documents will also be stored on Github. This provides an external environment to store and retrieve files, should any localised hard drive malfunctions occur.

# Tracking Progress

Github will also provide a way for the team to keep track of project progress and complete the tasks set out on the project timeline.

The project leader will analyse progress via the project gantt chart, then update relevant documents on Github so the rest of the team can acquire work as prompted.

# Communication

Communication between team members will be carried out through team meetings, Github, and Facebook.

Team Meetings - Team meetings will take place twice a week to ensure that all team members are both aware and understand the tasks allocated to them, but also that the work produced is up to the standards of the team.

Github - When an individual uploads or updates a file they have been working on, Github asks the user to make a note against their commit. This allows the team to view the progress of other team members.

Facebook - Facebook provides the ability to great groups. This group provides team members with the means to communicate privately on a medium readily available to all team members.

# Quality Control

Error Checking - Checking that all code produced complies and contains no errors. The team members responsible for writing the code will carry out the error checking.

Application Testing - The application will be tested in the final weeks of development to ensure that it meets the project requirements and that no functionality has been overlooked.

Documentation - Ensue that all supplied documentation covers all areas of the application developed, as well as the spelling and grammar.

Managing Change - If in the event the project changes during development, all team members as well as the client will be notified promptly. As the project team is using an agile approach to system design, the client will be provided with an incremental prototype of the system and supporting documentation.