**Software Requirements Specifications**

**Version 1.1**

**Project Management App**

**Team A**

**CSC-354**

**Fall 2015**



09/29/2015

Author: Jennifer Li

Submitted To: Dr. Joo Tan

**TABLE OF CONTENTS**

Revision History…………………………………………………………………………………..ii

1.0 Introduction…………...…………………………….…...…………………………….……....1

* 1. Product Perspective……………………………………………………………………1
  2. Purpose………………………………………………………………………………...1
  3. Scope of Project……………………………………………………………………….1
  4. Definitions, acronyms, and abbreviations……………………………………………..1
  5. References……………………………………………………………………………..2

2.0 Project Description…………………………………………………………………………….3

2.1 System Overview……………………………………………………………………...4

2.2 Overview of Non-Functional Requirements…………………………………………..5

2.2.1 List of Non-Functional Requirements………………………………………5

2.3 Overview of Functional Requirements………………………………………………..5

2.3.1 Use Cases of Functional Requirements……………………………………..6

F.R.1: Create User Account Use Case………………………………...…..7

F.R.2: Login Use Case……………………………………………….........8

F.R.3: Create Project Use Case……………………………………………9

F.R.4: Add Members Use Case…………………………………………..10

F.R.5: Remove Members Use Case……………………………………...11

F.R.6: Assign Task Use Case……………………………………….........12

F.R.7: Submit Task Use Case…………………………………………....13

F.R.8: Review Task Use Case……………………………………………14

F.R.9: View Project Progress Use Case……………………………...…..15

F.R.10: View Member Progress Use Case…………………………….…16

F.R.11: Member’s Progress Use Case…………………………………...17

F.R.12: Accepting Project Invitation Use Case……………..…………...18

F.R.13: View Due Date Use Case………………………………………..19

F.R.14: Messaging Use Case…………………………………………….20

F.R.15: Reminders Use Case…………………………..……………...…21

F.R.16: Logout Use Case………………………………………………...22

**REREVISION HISTORY**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Author | Description | Date |
| 1.0 | Jennifer Li | I created the first draft. | 09/22/2015 |
| 1.1 | Tyler Mariano | I made grammatical changes and cosmetic fixes. | 09/29/2015 |

**1.0 INTRODUCTION**

This document provides a detailed description of the Project Management App that will be developed. Within the document is a list non-functional and functional requirements. In addition to that, this document will also provide a set of use cases that will describe the interactions between the system and the user.

**1.1 Project Perspective**

After soliciting requirements from the client, the Project Management App was requested to be mobile to help improve the overall management of projects. This mobile based Project Management App’s intended purpose is to create a seamless management level system that will save time, reduce the workload of the manager, and organizes the overall project by the usage of tasks, charts, and dates.

**1.2 Purpose**

This SRS document will describe in detail about the requirements of the Project Management App. In this document, the reader will understand the purpose and the functionality of the Project Management App; by the description of components, the interactions between the user and the app interfaces, features and design.

**1.3 Scope of Project**

The Project Management App, will be compatible with mobile devices such as android phones and tablets. This mobile app will consist of a database to provide the following functionalities. In this app the user can create an account and join or create a project. As the leader of the project, the user can assign tasks, add members and or remove members, review tasks and etc. As the member of the project, the user can accept project invitation, submit tasks, view project progress, view their own progress and etc. With these functionalities in this app, the leader can organize and prioritize the tasks in a centralized location.

**1.4 Definitions, Acronyms, and Abbreviations**

In this document there will be terms and abbreviations that will be used. The table down below defines all the terms and abbreviations that will be used throughout the SRS.

|  |  |
| --- | --- |
| Term | Definition |
| SRS | Software Requirements Specification |
| App | Application and or System |
| User | Project Leader and or Project Member |

**1.5 References**

The following documents may be referenced by this document:

**2.0 PROJECT DESCRIPTION**

This section, will provide an overview of the functionality of the Project Management App, functionality such as user interaction with the app.

|  |  |
| --- | --- |
| User Level | Available Functionality |
| Project Leader | * The project leader can add members. * The project leader can remove members. * The project leader can assign tasks to the members. * The project leader can review each task. * The project leader can check each member’s individual task progress. * The project leader can set reminders. |
| Project Member | * The project member can submit assigned task. * The project member can join projects upon receiving an invite from the project leader. * The project member can view their task’s due date via calendar view. * The project member can view their own progress of the assigned task. |

General user overview: User’s computer skills will range from technological illiterate to technological savvy, therefore the app shall accommodate the weakest computer skill level. This will be accomplished by:

* Creating an app with a simple user interface
* Clear and simple directions for all of the functionalities for both types of users
* Using a uniform design for all the interfaces used by the user.

**2.1 System Overview**

This figure displays the system environment with the two actors and the interactive application. All actors can interact with the Project Management App.

Project Management App : Android Based Managing Application with DB

Project Management Application

Database

**2.2 Overview of Non-Functional Requirements**

The Project Management App has three persistent qualities and constrains that will be revised over and over again with the release of new versions or this App.

**2.2.1 List of Non-Functional Requirements**

The non-functional requirements of the Project Management App are listed below.

|  |  |
| --- | --- |
| Requirement | Type |
| The App must not be down for more than 1 min per day. | Reliability |
| The App must be designed with simplicity in mind. | Usability |
| The App must be able to support multi platforms that runs on Android. | Scalability |

**2.3 Overview of Functional Requirements**

The Project Management App has lot of functional requirements that define the specifications of the behaviors of the functions. The purpose of this is to identify the possible interactions between the types of users, the Project Management App and its database.

The functional requirements of the Project Management App are listed below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Functional  Requirements | Phase | Legacy | Requirement | Priority Level |
| F.R.1.0 |  | X | User shall be able to create an account | Necessary |
| F.R.2.0 |  | X | User shall be able to log-in. | Necessary |
| F.R.3.0 |  |  | User shall be able to create a project. | Necessary |
| F.R.4.0 |  | X | Project leader shall be able to add members to the created project. | Necessary |
| F.R.5.0 |  | X | Project leader shall be able to remove members to the created project. | Necessary |
| F.R.6.0 |  |  | Project leader shall be able to assign task to the project members. | Necessary |
| F.R.7.0 |  | X | Project members shall be able to submit completed task for review. | Necessary |
| F.R.8.0 |  | X | Project leader shall be able to review the submitted task. | Necessary |
| F.R.9.0 |  |  | User shall be able to view the project’s overall progress. | Necessary |
| F.R.10.0 |  |  | Project leader shall be able to view each project members’ task progress. | Necessary |
| F.R.11.0 |  |  | Project members shall be able to view their assigned task progress. | Necessary |
| F.R.12.0 |  | X | User shall be able to accept project innovations from project leaders. | Necessary |
| F.R.13.0 |  |  | User shall be able to view due dates of all project tasks. | Necessary |
| F.R.14.0 |  |  | User shall be able to message each user through the app. | Necessary |
| F.R.15.0 |  |  | Project leader shall be able to set reminders for all project members. | Necessary |
| F.R.16.0 |  | X | User shall be able to logout. | Necessary |

**F.R.1.0:** **Create User Account Use Case**

|  |  |
| --- | --- |
| Use Case Name: | Create User Account |
| Related Requirements: | F.R.10 |
| Scenario: | User creates an account on the app. |
| Triggered Event: | New user wants to set up an account in the app. |
| Brief Description: | The user creates an account by entering basic information such as, email, password, confirm password, phone number, biography and etc. |
| Actors: | User. |
| Stakeholders: | Project Leader and or Project Member |
| Pre-Conditions: | Create an account subsystem must be available.  Email validation service must be available.  Phone validation service must be available. |
| Post-Conditions: | User account must be created and saved.  Email information must be validated.  Phone information must be validated.  Password must be saved. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. User indicates a desire to create a user account and enters the basic prompted information. | 1.1 System creates a new user.  1.2 System prompts for user email address. | | 2. User enters email address. | 2.1 System creates email address  2.2 System verifies email address.  2.3 System prompts for user phone number. | | 3. User enters phone number. | 3.1 System creates user account.  3.2 System verifies phone number.  3.3 System associates user, email address, phone, and account.  3.4 System return valid user account details. | |
| Exception Conditions: | 1. Basic user information is incomplete.  2. Invalid email address.  3. Invalid phone number.  4. Mixed matched password. |

**F.R.2.0:** **Login Use Case**

|  |  |
| --- | --- |
| Use Case Name: | Login |
| Related Requirements: | F.R.2.0 |
| Scenario: | User logs into the app. |
| Triggered Event: | The user opens the app and navigates to the login interface. |
| Brief Description: | The user must enter their full email address as the username, and must enter the password to their account. Then user must submit the information. |
| Actors: | User. |
| Stakeholders: | Project Leader and or Project Member |
| Pre-Conditions: | User must be created in the database.  User must have created a password.  Enter password must match the password in the database. |
| Post-Conditions: | User will gain access to the app and functionality. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. User enters email address, password, and submits the information. | 1.1 Database is search for the email address.  1.2 If found, database record for the user is retrieved.  1.3 Password from database record is compared to the entered password from the user.  1.4 If password matched, session is created. | |
| Exception Conditions: | 1. Invalid username match with the database.  2. Invalid password match with the database.  3. Use account information is not in the database. |

**F.R.3.0:** **Create Project Use Case**

|  |  |
| --- | --- |
| Use Case Name: | Create Project |
| Related Requirements | F.R.3.0 |
| Scenario: | User has logged in, they will be able to create a project. |
| Triggered Event: | User will need to manage and organize a project. |
| Brief Description: | The user will be able to create a project by providing information such as, name of project, due date, description, and members. |
| Actors: | User. |
| Stakeholders: | Project Leader. |
| Pre-Conditions: | A project that needs to be managed.  User must be logged in. |
| Post-Conditions: | The project that needs to be managed will be initialized.  The user will automatically become the project leader. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. User will enter the Name of Project, Due Date, Description, and members. | 1.1 A connection is made to the database.  1.2 All of the information is stored in to the database. | | 2. User clicks “Create Project” | 2.1 The system will periodically check the information stored into the database. | |
| Exception Conditions: | 1. Project information entered is incomplete. |

**F.R.4.0:** **Add Members Use Case**

|  |  |
| --- | --- |
| Use Case Name: | Add Members |
| Related Requirements | F.R.4.0 |
| Scenario: | The project leader wishes to add a member to the project |
| Triggered Event: | The added user meets the project leader’s need for the certain position. |
| Brief Description: | The project leader will be able to add members to the project by searching the user by username. After reading the biography, if credentials are met the leader will select the user as a possible member to the project. |
| Actors: | Project Leader. |
| Stakeholders: | Project Leader and Project Members. |
| Pre-Conditions: | User must be logged in.  User must create a project.  User must be in the database of the app. |
| Post-Conditions: | A user will be prompted if they would like to be a member of this project. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. Project Leader will click the add a member button. | 1.1 A connection is made to the database. | | 2. Project Leader will search through the results. | 2.1 The system will display all users of the app. | | 3. Project Leader will choose a user to add as a member. | 3.1 The system will send a confirmation to the leader.  3.2 The system will prompt the chosen user an invitation. | |
| Exception Conditions: | 1. User not in the database. |

**F.R.5.0:** **Remove Members Use Case**

|  |  |
| --- | --- |
| Use Case Name: | Remove Member |
| Related Requirements | F.R.5.0 |
| Scenario: | The Project Leader wishes to remove a member from the |
| Triggered Event: | An event that causes a member of the project to not be able to complete the assigned task. Examples: illness, termination of contract, misconduct and or etc. |
| Brief Description: | The Project Leader will be able to functionally remove the Project Member form the project by resigning the assigned task to the member and deleting the user from the project. |
| Actors: | Project Leader. |
| Stakeholders: | Project Leader and Project Member. |
| Pre-Conditions: | Project Leader must be logged in.  Project Member must be a member of the project. |
| Post-Conditions: | A Project Member will be deleted from the project. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. Project Leader will click the remove a member button. | 1.1 The appropriate SQL statement will execute deletion from the specific project.  1.2 Confirmation will be display of the completion of the deletion of the ex-Project Member. | |
| Exception Conditions: | 1. User a Project Member of the project. |

**F.R.6.0:** **Assign Task Use Case**

|  |  |
| --- | --- |
| Use Case Name: | Assign Task |
| Related Requirements | F.R.6.0 |
| Scenario: | The Project Leader wishes to assign task to a member of the project. |
| Triggered Event: | Created project have task that needs to be completed. |
| Brief Description: | The Project Leader will provide basic information about the task such as, description, dependency, due date, and etc. |
| Actors: | Project Leader. |
| Stakeholders: | Project Leader and Project Member. |
| Pre-Conditions: | Project Leader must be logged in.  Project has to be created.  Project must have member or members. |
| Post-Conditions: | Task will be assigned to a certain Project Member. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. Project Leader will click the project member’s username and provide the basic information and or document that are need to complete this task. | 1.1 The system will connect to the database.  1.2 The system will store all the entered information to the database. | |
| Exception Conditions: | 1. Project task information entered is incomplete. |

**F.R.7.0:** **Submit Task Use Case**

|  |  |
| --- | --- |
| Use Case Name: | Submit Task |
| Related Requirements | F.R.7.0 |
| Scenario: | The Project member has complete the task and wishes to submit to the Project Leader. |
| Triggered Event: | Project Member has finished all of the requirements that were need to complete the task. |
| Brief Description: | The Project Member will provide all document that will be submitted for review and any necessary comments about the task. |
| Actors: | Project Member. |
| Stakeholders: | Project Member and Project Leader. |
| Pre-Conditions: | Project Member must be logged in.  Project Member must have been assigned a task.  The completion of task must be before the due date. |
| Post-Conditions: | The task, with all of the documents and comments will be reviewed. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. Project Member will click the “Submit Task” button. | 1.1 The system will connect to the database. | | 2. Project Member will upload all document and comments | 2.1 The system will store all the entered information to the database.  2.2 The system will prompt the Project Leader about the task submission. | |
| Exception Conditions: | 1. Missed the due date. |

**F.R.8.0:** **Review Task Use Case**

|  |  |
| --- | --- |
| Use Case Name: | Review Task |
| Related Requirements | F.R.8.0 |
| Scenario: | The Project Leader seen the prompted message and wishes to review the submitted task. |
| Triggered Event: | The Leader has received the completed documents and comments on or before the set due date. |
| Brief Description: | The Project Leader will review the submitted documents with comments. If the task is done to the Project Leader’s liking, progress will be updated for the entire project and the specific member. Also unlock any dependent task that depends on the completion of the completed task. If the task is bot completed to the Project leaders liking they can re-assign the task. |
| Actors: | Project Leader. |
| Stakeholders: | Project Leader and Project Member |
| Pre-Conditions: | Project Leader must be logged in .  Task must be submitted.  Project Leader must be prompted. |
| Post-Conditions: | Update in the overall project progress  Update in the project member’s progress  Or resigned if not completed. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. Project Leader will click the “Review Task” button. | 1.1 The system will connect to the database. | | 2. Project Leader will download all document and comments. | 2.1 The system will retrieve all the entered information for this task from the database. | | 3. If complete both progress will be updated; if not task will be re-assigned | 3.1 Both progresses will be updated in the database.  3.2 –Refer to F.R.6.0 | |
| Exception Conditions: | 1. Must update the overall progress if completed properly.  2. Must update the project member’s progress if completed properly.  3. If not must re-assign task |

**F.R.9.0:** **View Project Progress Use Case**

|  |  |
| --- | --- |
| Use Case Name: | View Project Progress |
| Related Requirements | F.R.9.0 |
| Scenario: | Project Leader and or Project Member wishes to view the progress of the overall progress of the project. |
| Triggered Event: | Possible trigger would be the want of viewing the progression of the project after the Project Leader updates it. |
| Brief Description: | Project Leader updates the task which will be represented in a Gantt chart. It will display the overall progress of the project by the percent of completion of project tasks. |
| Actors: | Project Member and Project Leader |
| Stakeholders: | Project Member and Project Leader |
| Pre-Conditions: | Both Project Member and Project Leader must be logged in.  Project must be created.  Project must have assigned task weather completed or not.  Project Leader must update or updated it. |
| Post-Conditions: | The progress display in Gantt chart format. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. Project Leader and or Project Leader will click the “View Project Progress” button. | 1.1 A connection made to the database.  1.2 The appropriate SQL statement will executed.  1.3 The results are displayed | |
| Exception Conditions: | 1. Faulty connection with database. |

**F.R.10.0:** **View Member Progress Use Case**

|  |  |
| --- | --- |
| Use Case Name: | View Member Progress |
| Related Requirements | F.R.10.0 |
| Scenario: | Project Leader wishes to view the progress of each member of the project. |
| Triggered Event: | The Project Leader needs to view the Project Members progress. |
| Brief Description: | Project Leader updates the task which will be represented in a Gantt chart. It will display the progress of each of the project members percent of completion of project tasks. |
| Actors: | Project Leader. |
| Stakeholders: | Project Leader. |
| Pre-Conditions: | Project Leader must be logged in.  Project must be created.  Project must have assigned task weather completed or not.  Project Leader must update or updated it. |
| Post-Conditions: | The progress display in Gantt chart format. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. Project Leader will click the “View Member Progress” button. | 1.1 A connection made to the database.  1.2 The appropriate SQL statement will executed.  1.3 The results are displayed | |
| Exception Conditions: | 1. Faulty connection with database. |

**F.R.11.0:** **Member’s Progress Use Case**

|  |  |
| --- | --- |
| Use Case Name: | Member’s Progress |
| Related Requirements | F.R.11.0 |
| Scenario: | Project Member wishes to view their own progress of task they completed for the project. |
| Triggered Event: | The Project Member would like to view their own progress. |
| Brief Description: | After the Project Leader updates the task which will be represented in a Gantt chart. It will display the progress of a Project Member’s percent of completion of project tasks. |
| Actors: | Project Member. |
| Stakeholders: | Project Member. |
| Pre-Conditions: | Project Member must be logged in.  Project must be created.  Project must have assigned task weather completed or not.  Project Leader must update or updated it. |
| Post-Conditions: | The progress display in Gantt chart format. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. Project Member will click the “View Member’s Progress” button. | 1.1 A connection made to the database.  1.2 The appropriate SQL statement will executed.  1.3 The results are displayed | |
| Exception Conditions: | 1. Faulty connection with database. |

**F.R.12.0:** **Accepting Project Invitation Use Case**

|  |  |
| --- | --- |
| Use Case Name: | Accepting Project Invitation |
| Related Requirements | F.R.12.0 |
| Scenario: | Users will receive prompted messaging asking if the user would like to join a Project. |
| Triggered Event: | Project Leader adds a certain user to the project, which prompts a message of invitation to the project. |
| Brief Description: | One invitation is accepted the user will become a Project Member. Which lets the Project Leader assign task to the newly added Project Member. |
| Actors: | Project Member. |
| Stakeholders: | Project Member. |
| Pre-Conditions: | Project Member must be logged in.  Project Leader must have sent an invitation to the project. |
| Post-Conditions: | Acceptance to be a Project Member.  If denied, the user will stay a user. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. Project Member will click the “Accept” button. Or the “Deny” button. | 1.1 If accepted connection made to the database to add the new member.  1.2 Confirmation message will be sent to the Project Leader.  1.3 If deny – system will not connect to the database. | |
| Exception Conditions: | 1. Faulty connection whether database, cell phone, and or internet.  2. Phone is out of power or turned off when message is sent. |

**F.R.13.0:** **View Due Date Use Case**

|  |  |
| --- | --- |
| Use Case Name: | View Due Date |
| Related Requirements | F.R.13.0 |
| Scenario: | Project Leader and or Project Member wishes to view the upcoming due dates for the upcoming month or months. |
| Triggered Event: | Possible trigger, newly added members view upcoming due dates or add tasks to the overall project. |
| Brief Description: | This feature will contain a calendar view of all due dates. This can be viewed by all users involved in the project. |
| Actors: | Project Leader and or Project Member. |
| Stakeholders: | Project Leader and Project Member. |
| Pre-Conditions: | Both Project Members and Project Leader must be logged in.  Project must be created.  Project must have assigned task with set due dates. |
| Post-Conditions: | Display of a calendar with all upcoming due dates that are incorporated with the project. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. Project Leader and or Project Member will click the “View Due Dates” button. | 1.1 A connection made to the database.  1.2 The appropriate SQL statement will executed.  1.3 The results are displayed | |
| Exception Conditions: | 1. Faulty connection with database. |

**F.R.14.0:** **Messaging Use Case**

|  |  |
| --- | --- |
| Use Case Name: | Messaging |
| Related Requirements | F.R.14.0 |
| Scenario: | Project Leader and or Project Member wishes to contact the people involved in the project. |
| Triggered Event: | Possible trigger would be an issue with the assigned task or a simple question. |
| Brief Description: | The messaging functionality is for all users that are involved in the project. Generally used for questions, and or quick response to pressing matters. |
| Actors: | Project Leader and or Project Member. |
| Stakeholders: | Project Leader and Project Member. |
| Pre-Conditions: | Both Project Members and Project Leader must be logged in.  Project Leader and or Project Member must have a question about the project. |
| Post-Conditions: | Activation of chatting functionality.  Communication connection link set with sender and receiver.  Message sent to the receiver. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. Project Leader and or Project Member will click the “Message” button. | 1.1 A connection made to the database.  1.2 connect to the appropriate receiver.  1.3 The message reminder will be sent. | |
| Exception Conditions: | 1. Faulty connection whether database, cell phone, and or internet.  2. Phone is out of power or turned off when message is sent. |

**F.R.15.0:** **Reminders Use Case**

|  |  |
| --- | --- |
| Use Case Name: | Reminders |
| Related Requirements | F.R.15.0 |
| Scenario: | Project Leader and Project members will receive message reminders of due dates. |
| Triggered Event: | Approaching due dates. |
| Brief Description: | Project Leader will activate the reminder feature which sends notifications to both the Project Leader and the Project Members of the upcoming due dates. |
| Actors: | Project Leader. |
| Stakeholders: | Project Leader and Project Members |
| Pre-Conditions: | Project Leader must be logged in.  Project must be created.  Project must have assigned task with set due dates. |
| Post-Conditions: | Prompted messages about upcoming due dates. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. Project Leader will click the “Set Reminder” button. | 1.1 A connection made to the database.  1.2 The appropriate SQL statement will execute to retrieve all upcoming due dates.  1.3 The message reminder will be sent. | |
| Exception Conditions: | 1. Phone is out of power or turned off when message is sent. |

**F.R.16.0:** **Logout Use Case**

|  |  |
| --- | --- |
| Use Case Name: | Logout |
| Related Requirements: | F.R.16.0 |
| Scenario: | User logs out of the app. |
| Triggered Event: | The user closes out of the app or session times out. |
| Brief Description: | The user will click the logout button or be inactive for 30 minutes. |
| Actors: | User. |
| Stakeholders: | Project Leader and or Project Member. |
| Pre-Conditions: | User must be logged into the app. |
| Post-Conditions: | App will destroy the session. |
| Flow of Activities: | |  |  | | --- | --- | | Actor | System | | 1. User click logout button. | 1.1 Session is destroyed. | | 2. User is inactive for 30 minutes. | 2.1 Session is destroyed. | |
| Exception Conditions: |  |