Software Project Management Plan

Team 2

5 Jenuary, 2020

Team Members

Bryan Imaralu, Giovanni Gottardi, Nicolò Salaorni, Edoardo Gruia

Change History

Revision	Change Date	Description of changes	Author
V1.0	05/01/'20	Initial release	Giovanni Gottardi

Document Storage

This document is stored in the project's SVN repository at: https://github.com/BryanImaralu/2020_5bi_team2_Imaralu

Document Owner

Bryan Imaralu is responsible for developing and maintaining this document.

1 Overview

1.1 Purpose and Scope

The Golden Boys Team is interested in creating a mobile application that will be useful for chatting with people by giving them the possibility to send and receive messages, send broadcast messages and more.

The chat app will provide users with the ability to log in to their account via the login page and see the messages received. A second function will allow users to view a list of people with whom the user has chatted.

The application will be created for use on a mobile device with an iOS and android platform, essentially all devices.

The user interface will be intuitive and simple to use, with 95% of new users being able to use the application without referencing the user manual.

1.2 Goals and Objectives

The overall objective is to offer a useful and simple app to use for messaging with people.

Project Goals:

1. Create an app that functions as expected and looks great.
2. Learn about software engineering and creating a mobile app.

Project Objectives:

1. Create an app that functions in a simple and intuitive manner.

1.3 Project Deliverables

Date	Deliverable
09/23/10	Requirements Specification
09/28/10	Project Plan
10/05/10	Iteration #1 Plan
10/07/10	Technical Prototype
10/14/10	Customer Approved UI Prototype
10/21/10	Architecture Document
10/28/10	Iteration #1 Complete
11/11/10	Test Report
11/18/10	Iteration #2 Complete
12/02/10	User Guide and System Administration Manual
12/07/10	Product Released

1.4 Assumptions and Constraints

1.4.1 Assumptions

1. Server is always active.

1.4.2 Constraints

Constraints:

1. The application will run on an every devices.

1.5 Success Criteria

A working app, easy-to-use that allows users to chat with people they want.

1.6 Definitions

Term	Definition
Actor	user or other software system that receives value from a user case.
Baselined	the work product has undergone a formal review and can only be changed through the prescribed change control procedures
Client or Customer	the person or organization for which this application is being built.
Developer	the person or organization developing the system, also sometimes called the supplier.
Project	activities that will lead to the production of the chat application. ** **
Chat Application	the product that is being described here; the software system specified in this document.
Scenario	one path through a user case
Stakeholder	anyone with an interest in the project and its outcomes. This includes clients, customers, users, developers, testers, managers and executives.
User	the person or persons who will actually interact with the chat application.
Use case	describes a goal-oriented interaction between the system and an actor. A use case may define several variants called scenarios that result in different paths through the use case and usually different outcomes.

2 Startup Plan

2.1 Team Organization

Role	Actor(s)	Responsibility
Project Manager	Bryan Imaralu	Call team meetings, coordinate communications within group, coordinate communications outside group, break out tasks, assign them to teammates
Developer	Bryan Imaralu, Giovanni Gottardi	Develop software based on requirement and architect specifications
Programmer	Edoardo Gruia, Nicolò Salaorni	Program to requirement and architect specifications
Tester	Giovanni Gottardi, Nicolò Salaorni	Write test cases, perform unit testing of test cases against incremental release of code, perform integrated testing of test cases against incremental release of code, report issues
Architect	Edoardo Gruia	Specify overall internal workings of application
Requirement Engineer	Bryan Imaralu	Outline and document project dependencies and requirements. This includes internal and external dependencies.

2.2 Project Communications

Event	Information	Audience	Format	Frequency
Team Meeting	Activity status: work in progress;obstacles encountered; manage incoming messages and broadcast lists	All team members	Informal meetings after class; Formal meetings as needed; Updates on application status problems when they occur	Once time a week
Project Status Report	Review finished items, status of prototype; review any problems, schedule slippage, programming issues	All team members, costumer	E-mail with information or In-person as customer sees fit	Iteration Closeout

2.3 Technical Process

An iterative and incremental development process is planned. Feedback will be used from each iteration to improve the next. The first iteration will focus on basic functionality of the application. Subsequent iterations will build upon that and incorporate more features.

2.4 Tools

- Programming - Python
- Version Control – all work products will be stored in an repository

3 Work Plan

3.1 Release Plan

3.1.1 Plan By Feature

Iteration #1

Summary: Demonstrate fundamental architecture, and have authentication integrated into application.

Features / Deliverables	Estimated Effort	Actual Effort
Authentication	30	

Iteration #2

Summary: Includes the ability to also view broadcast chats in the app.

Features / Deliverables	Estimated Effort	Actual Effort
Broadcast chat	45	

Iteration #3

Summary: Allows and displays group chats.

Features / Deliverables	Estimated Effort	Actual Effort
Group chat	75	

3.2 Iteration Plans

3.2.1 First Iteration

The server interface works both with the iOS application and the Android one.

3.2.2 Mobile iOS application.

Mobile iOS application gives user the ability to see their chat.

3.2.3 Final Product

The server interfaces with the application. The mobile application interfaces with the server, giving users the ability to chat.

4 Control Plan

4.1 Monitoring and Control

The following list of dates includes formal reviews outside of the Communication Plan. Milestones are included to reference where the project is scheduled to stand as these reviews occur:

Date	Review / Milestone
10/07/2019	Milestone: Technical Prototype Complete
10/07/2019	5-Minute Status Report
10/28/2019	Manager's Briefing
10/28/2019	Milestone: Iteration #1 Complete
11/11/2019	Milestone: Test Report Complete
11/16/2019	Inspection
11/18/2019	Milestone: Iteration #2 Complete
12/07/2019	Milestone: Product Released
12/09/2019	Final Presentations

4.2 Configuration Management Plan

The following procedure is to be used when making changes to all baselined work products:

1. All project work products will be stored in a centralized repository running on a central server.
2. All baselined documents will have a Document Control section with a change history to track initialization and subsequent changes.
3. All project work products (documents, source code, test cases, program data, test data, etc) will be stored in the repository but not all will be under change control (subject to formal change control procedures.) Only the system requirements, project plan and source code will be baselined and under configuration control.
4. Items that are subject to change control will be considered baselined after a group review at the end of the initial document creation.
5. The change control procedure once a product is baselined is:

(1) anyone wanting to make a change to a baselined item sends an email to the rest of the team and project sponsor (i.e. Salaorni) describing the change, reason for the change, expected schedule impact, and time line for integrating the change.

(2) if no one responds to the group within 2 days with a reason for why the change request shouldn't be permitted, it will be considered accepted and the person proposing the change may proceed with the change.

1. (3)if anyone does object to the change, the reason for objecting will be discussed at a meeting where everyone is invited to attend and voice their opinion. At the end of the meeting a democratic vote will be held to decide whether or not the change should be allowed.
2. (4)if a change takes place, the initiator must collaborate with the project manager to update the schedule.

5 Supporting Process Plans

5.1 Risk Management Plan

Rank	Risk	Probability of Loss	Size of Loss	Risk Exposure	Response
1	Schedule / time line delivery	Likely	Major	High	Mitigate: Stick to the schedule.
2	Improve the communication with the server	Likely	Moderate	Moderate	Avoid: breakdown of the server engine.
3	Learning curve for new tools and technologies longer than expected	Unlikely	Moderate	Moderate	Buy Information: Begin working on a basic prototype early to test out fundamental programming concepts & knowledge
4	Data feed not available from the server	Unlikely	Minor	Low	Avoid: instead of using a feed from the server, it depends on access to the app and extracts the information requested from users who use the application.

5.2 Test Plan

The test plan defines the items that will be tested, methods for testing, and a schedule detailing the tasks, owners, and time line.

The test plan will be available in a separate document in the version control system at: https://github.com/BryanImaralu/2020_5bi_team2_Imaralu

5.3 Product Acceptance Plan

At the end of each iteration, the prototype created will be tested to ensure that it meets the requirements of that iteration. An environment like the iPhone simulator and the Android simulator can be used to test the functionality instead of the app loaded on a real smartphone.

For the final iteration, product acceptance tests will ensure that the prototype works as expected with user data.