Software Project Management Plan

Team 4

October 28, 2019

Team Members

Karim Bachir
Sahas Senapathige
Sabrina Er Raqioui
Valentin Toth
Riccardo Montresor

Document Control

Change History

Revision	Change Date	Description of changes
V1.0	06/12/2019	Initial release
V2.0	03/01/2020	Final release

Document Owners

Senapathige Sahas and Toth Mihai Valentin are responsible for developing and maintaining this document.

Table of Contents

Overview

1.1 Purpose and Scope

Allow students and teachers to use an in-house messaging app for the exchange of information related to the school or external activities.

The application is required to provide a simple interface for sending messages.
As mentioned in the objectives section above, the delivered product should demonstrate the ability to provide proper communication end-to-end and broadcast users.

1.2 Goals and Objectives

Objectives: the general objective is to provide an alternative way to exchange information.
It is expected:

- an interface developed with JFrame that allows access to the user;
- simple and intuitive operation.
- simple connection to other chat members

1.3 Project Deliverables

Date	Deliverable
08/10/2019	Subdivisions of tasks within the group and client initialization
25/10/2019	Chat project introduction
15/11/2019	Added code of client and server
	Technical Prototype
18/11/2019	Finishing developing graphic part
29/11/2019	Developed code of the client: added all packages
01/12/2019	New GUI. Rivisited the first version.
04/12/2019	Implemented chat group
12/12/2019	Improved structure of code
20/12/2019	Code ready to be presented and released

1.4 Assumptions and Constraints

1.4.1 Assumptions

1. The client should connect to the server.
2. The client should interact with the server to exchange messages in a public and private chat Interfacing the app with a proxy server and third-party website is feasible.
3. The client should have a graphical interface

1.4.2 Constraints

Constraints:

1. Use TCP protocol.
2. Client must connect with every TCP server.
3. Generate the packages wanted by the server.

2 Startup Plan

2.1 Team Organization

Role	Actor(s)	Responsibility
Project Manager	Er Raqioui. Bachir	Call team meetings, coordinate communications within group, coordinate communications outside group, break out tasks, assign them to teammates
Developer	Bachir	Develop software based on requirement and architect specifications
Programmer	Bachir, Montresor, Senapathige, Er Raqioui, Toth	Program to requirement and architect specifications
Tester	Bachir, Montresor, Senapathige, Er Raqioui, Toth	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	Bachir	Specify overall internal workings of application
Requirement Engineer	Bachir, Montresor, Senapathige, Er Raqioui, Toth	Outline and document project dependencies and requirements. This includes internal and external dependencies.

2.2 Project Communications

Event	Information	Audience	Format	Frequency
Team Meeting	Task status: completed since last meeting & planned for next; obstacles encountered; change requests in process	All team members	Informal meetings during class hours; Formal meetings as needed; Updates and problems via TeamSpeak or Discord	As needed
Project Status Report	Review finished items, status of prototype; review any problems, schedule slippage, programming issues	All team members, customer	Message on Whatsapp or notes in GitHub's commit	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 as time allows.

2.4 Tools

- Programming & Markup Languages – Java, XML
- Operating System – Windows 8.1/10
- Version Control – all work products will be stored in an SVN repository
- Development Tools – Netbeans
- Presentation – Google Presentation

3 Control Plan

3.1 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
10/07/2010	*Milestone: Technical Prototype Complete*
10/07/2010	5-Minute Status Report
10/28/2010	Manager's Briefing
10/28/2010	*Milestone: Iteration #1 Complete*
11/11/2010	*Milestone: Test Report Complete*
11/16/2010	Inspection
11/18/2010	*Milestone: Iteration #2 Complete*
12/07/2010	*Milestone: Product Released*
12/09/2010	Final Presentations

3.2 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 SVN repository running on a central server (GitHub).
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 SVN 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 a message to the rest of the team describing the change, reason for the change.

(2) if no one responds to the group within 2 hours 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.

(3) if anyone does object to the change, the reason for objecting will be discussed at a meeting on TeamSpeak or Discord 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.
(4) if a change takes place, the initiator must collaborate with the project manager to update the schedule

4 Supporting Process Plans

4.1 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	ManageMyID website changes HTML layout	Likely	Moderate	Moderate	Avoid: Breaking the parsing engine out into a separately hosted server allows for fixes to occur in one place, instantly for all users, and with minimal intrusion.
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 ManageMyID website	Unlikely	Minor	Low	Avoid: Instead of using a feed from ManageMyID, depend upon logging into the site, and pull relevant information from HTML fields.

4.2 Test Plan

The test plan defines the items that will be tested, methods for testing.

4.3 Product Acceptance Plan

At the conclusion of each iteration, the prototype created will tested to ensure it meets the requirements of that iteration. An environment such as the iPhone simulator may be used to test functionality in lieu of the app being loaded on an actual smartphone.

For the final iteration, product acceptance testing will ensure that the prototype functions as expected with a user's data.