Software Requirements Specification

for

CS401 Final Project

Version 1.0

Prepared by

Group Name:

|  |  |  |
| --- | --- | --- |
| Luis Haro | Lhr004@gmail.com |  |
|  |  |  |

|  |  |
| --- | --- |
| Instructor: | Christopher Smith |
| Course: | CS 401 |
| Date: | 4/23/2020 |

Contents

Contents ii

Revisions ii

1 Introduction 1

1.1 Document Purpose 1

1.2 Product Scope 1

1.3 Intended Audience and Document Overview 1

1.4 Definitions, Acronyms and Abbreviations 1

1.5 Document Conventions 1

1.6 References and Acknowledgments 1

2 Overall Description 2

2.1 Product Overview 2

2.2 Product Functionality 2

2.3 Design and Implementation Constraints 2

2.4 Assumptions and Dependencies 3

3 Specific Requirements 4

3.1 External Interface Requirements 4

3.2 Functional Requirements 4

3.3 Use Case Model 5

4 Other Non-functional Requirements 6

4.1 Performance Requirements 6

4.2 Safety and Security Requirements 6

4.3 Software Quality Attributes 6

5 Other Requirements 6

Appendix A – Data Dictionary 7

Appendix B - Group Log 7

Revisions

| Version | Primary Author(s) | Description of Version | Date Completed |
| --- | --- | --- | --- |
| Draft Type and Number | Full Name | Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded. | 00/00/00 |

# 

# Introduction

## Document Purpose

This document will focus and, on the scope, requirements and specifications of the final project. The document will inform us of the principals of the project so students or instructor can have an idea of the whole project; this way, the members of the group can follow the requirement or make changes to it, in the same way other students can have a picture of the project and take over if it is necessary.

## Product Scope

This software project will create a new chat program based on Java, users can use this program to create room, login, create new accounts, send messages to the room as public or private.

## Intended Audience and Document Overview

This document is intended to be use for developers as a guide for requirements as that develop the software; in addition, it’s intended to inform users of what they can do with this new software. It can also be used as a guideline for students whom might have to work in this project in the future.

## Definitions, Acronyms and Abbreviations

UML - Unified Modeling language.

## Document Conventions

Use Arial font size 12 throughout the document for text. Use italics for comments. Document text should be single spaced and maintain the 1” margins found in this template. For Section and Subsection titles use size 14 Arial bold.

## References and Acknowledgments

Most of the information in this document can be found in out GitHub page:

**https://github.com/luisHaroDev/FinalProject-2.git**

Template of this document can be found at CS.GMU.EDU

# Overall Description

## Product Overview

A screenshot of a social media post

Description automatically generated

A screenshot of a cell phone

Description automatically generated

A screenshot of a social media post

Description automatically generated

## Product Functionality

This project will be a simple chat room / messaging system using a client-server model. The project will allow users on the client side to create an account, using a username and password, join a chat room, and send messages in that chat room.

The server will maintain each chat room and forward messages from clients in a chat room to all other clients in the same chat room. The server will use a database to maintain user accounts, chat rooms, and chat room message histories.

The client application will present controls to create an account, log in, select a chat room to chat with other people, view the chat room's message history once you have entered that specific chat room, and send messages. Also, be able to send private messages between two users. This chat will make a smoother communication for people. Broadcast general notifications, such as user logins and logouts.

.

## Design and Implementation Constraints

We will be using and array of classes to store data about users, rooms, messages, receiver and senders of messages and time. Assumptions and Dependencies

We will assume that our database is secure and not on site.

# Specific Requirements

## External Interface Requirements

### User Interfaces

This software will be user friendly and will use GUI interface to interact will the user

### Hardware Interfaces

The only hardware interface will be a computer and IDE with the available to run Java programs such as Eclipse, NetBeans, etc. and also a terminal

### Software Interfaces

Our programs will interface with our databases and software will interface with different classes and components within the project.

## Functional Requirements

The user should be able to created and new account, using a username and password and also be able to log-in using this data.

The Program should verify the user credentials and give access or denied it, a message should appear letting the user know why he/she was not allowed into the chat if credential do not match the database.

The database must have users’ credentials and message history store in a text file.

Users should be able to create chatrooms, join existing chat rooms, in this chat rooms the user will be able to send public or private messages to other users.

The User can read notification once he/she logged in.

## Use Case Model

User run program.

Click create new account, enter username and password, this is saved in database.

Click create a new chatroom, user enter a name for the new chat room.

User type a string into a string box and click send

Message appeared to everyone in the chat to see.

User click log-off and the program is closed.

# Other Non-functional Requirements

## Performance Requirements

Our software will try to use our data structure as efficient as possible. We will delete user and add new one in the array in the same index to save memory. time complexity for search will be keep simple since it is a same data file.

## Safety and Security Requirements

Not much security features will be use for the exception if the user it is in the database.

## Software Quality Attributes

The program should be easy to use and the code easy to read in order to allow team members to understand, it should have comments have explain variable, methods and attributes so other student can follow if it is necessary.

# Other Requirements

Appendix A- Group Log