Bhavan Patel

Manpreet Singh

Umaimah Syed

Amna Tiwana

Brandon Nguyen

**Phase 2**

Software Requirements Specification

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Revision** | **Description** | **Author** |
| 06/12/2020 | 1.0 | Initial Version | Team |
| 06/13/2020 | 1.1 | Added more Content | Team |
| 06/14/2020 | 1.2 | Made the flow chart | Manpreet Singh |
| 06/14/2020 | 1.2 | UML Diagram Design | Team |
| 06/15/2020 | 1.3 | SRS Document Added more content | Team |
| 6/16/2020 | 1.4 | SRS Finished First Draft | Team |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

Assignment #2 – Template 1

1. Purpose iii
   1. Scope 4
   2. Definitions, Acronyms, Abbreviations 4
   3. References 4
   4. Overview 4
2. Overall Description 4
   1. Product Perspective 4
   2. Product Architecture 6
   3. Product Functionality/Features 6
   4. Constraints 6
   5. Assumptions and Dependencies 6
3. Specific Requirements 7
   1. Functional Requirements 7
      1. Common Requirements: 7
      2. \_\_Account\_\_\_ Module Requirements: 7
      3. \_\_Add Account/Login\_\_\_ Module Requirements: 7
      4. \_Chat Box\_\_\_\_ Module Requirements: 7
      5. \_Listen and Receive\_\_\_\_ Module Requirements: 7
      6. \_Delete\_\_\_\_ Module Requirements: 7
   2. External Interface Requirements 7
   3. Internal Interface Requirements 8
4. Non-Functional Requirements 8
   1. Security and Privacy Requirements 8
   2. Environmental Requirements 8
   3. Performance Requirements 8

# Purpose

Purpose of our program, multiuser communication system. Provides a service for its users allowing them to communicate thew the internet one on one chat.

## Scope

Our program will is simple fast way to communicate

## Definitions, Acronyms, Abbreviations

IP-Internet Protocol

## References

Use Case Specification Document – Step 2 in assignment description

UML Use Case Diagrams Document – Step 3 in assignment description

Class Diagrams – Step 5 in assignment description

Sequence Diagrams – Step 6 in assignment description

## Overview

Our Program allows users to communicate one on one over the internet.

# Overall Description

## Product Perspective

Actor-User/client

Actor-Developer

Use Case ID: {User}

Use Case Name: {Anyone Who is going to be using our chat service}

Relevant Requirements: \* {User needs a machine that can install software

correctly }

Primary Actor: {User }

Pre-conditions: {Need Valid Email and set valid password, and Stable

internet connection}  
 Post-conditions: {Log out, save chat for user, close the program

completion of this use. Effects on other systems and

actors may also be described.}

Basic Flow or Main Scenario: {

1.User enters the login info

2. Server checks info is valid return the check is true

3. Allowing user access to the chat room and account details

4.Lookup Friends to Add

5.System Finds Friend and Connects user

with friend

6.Allows access to chat

7.Log out

8.System Saves Data and chat

}

Extensions or Alternate Flows: {

1. User Enters Login Info Incorrectly

2. Server Checks info and returns False

3. User enters wrong information 3 times in

row

4.System Prompts user to reset password

5.User enters new password

6.system updates password

7.User enters new login info

8 . System Checks Info returns true

9.repeat steps in main scenario 3 onwards }

Exceptions: { Incorrect Email,

Incorrect password

User not found,

Email address is taken,

user offline.}

Use Case ID: {Developer}  
 Use Case Name: {Has access to code, allowed to make changes to

program and update as needed.}

Relevant Requirements: \* {Also has designed the program in way that

none developers can’t access it. Also be able to code the program and all features listed correctly}

Primary Actor: {Developer }

Pre-conditions: {Has access to perform updates to code using IDE.

Post-conditions: {Check and update program if needed}

Basic Flow or Main Scenario: { To Update or edit code

1. Use IDE to change or update code

2.system update server

3.Close IDE

Extensions or Alternate Flows: { Abusive User that needs to be Taken off

the Platform Revoke access

1. Access Data Base and Find Username

2.Send email to user their access has been

revoked

3. Update data base take note this email is

put under revoked

4.delete account from data base

5.Finish update

Exceptions: {fail to update, If OS is updated to new version our

software might be out of date, data can get corrupted on server end}

## Product Architecture

Account Major class, Add Account/Login class, Chat Box Class, Delete Class, Listen /Receive\_Text Class

## Product Functionality/Features

-Login Logout

-reset password

-Connect With friends -Look up using email or username

-Allows save chat

-Delete Account

-send and receive messages

-Administer Can Suspend Account

## Constraints

-No Group chats

-No multiple windows for chat

-Only Text messaging sending and receiving allowed

-Need Valid Email address from user to create account

## Assumptions and Dependencies

List appropriate assumptions

-Need Valid Email address from user to create account

-Limited server recourses

-User will install the software on his machine

# Specific Requirements

## Functional Requirements

### Common Requirements:

-Internet access

-Need A account

-Need Operating System

### \_\_Account\_\_\_ Module Requirements:

-This is main Class that holds all the user information things like email address and passwords along with chat history.

-Needs Email Address

-Needs passwords

-Needs User ID

Validates All the information entered during login for user returns true or false only

### \_\_Add Account/Login\_\_\_ Module Requirements:

-Sign Up user Need Valid Email

-Password- minimum of six to 20 letters and numbers allowed

-Prompt user if Information is incorrect

### \_Chat Box\_\_\_\_ Module Requirements:

-Need one Friend to Chat

-Need able to communicate with server

-Need to able to search for first Friend and Add Friend to Chat

-Correct Format of Text for example a string

### \_Listen and Receive\_\_\_\_ Module Requirements:

-Connect Two IP address

-Need to be able to communicate with user

-Correct Format of Text for example a string

### \_Delete\_\_\_\_ Module Requirements:

-Needs A account

## External Interface Requirements

N/A

## Internal Interface Requirements

We need at least two user accounts for our platform to work as attended fully. Need access to the server at all times for most of the key functions to work correctly.

# Non-Functional Requirements

## Security and Privacy Requirements

-Message needs to encrypt

-Password data entry is obscured

## Environmental Requirements

N/A

## Performance Requirements

Windows OS and Mac OS any machine that can run java

**Flow Chart**

A close up of a map

Description automatically generated

**UML Diagram**

**A screenshot of a map

Description automatically generated**

**Project Schedule**

**A screenshot of a social media post

Description automatically generated**

**Roles & Responsibilities**

**Server Side:**

Team members: Manpreet Singh, Umaimah Syed, Bhavan Patel

Roles and responsibilities: .

• Work on the sever code

• Some of the GUI interface

• Find the potential security loopholes.

**Client Side:**

Team Members: Amna Tiwana, Brandon Nguyen

Roles and responsibilities:

• Work on the client-side code.

• Implement on the user interface.

• Update the appropriate version of the code with the sever side developer.

**Project Status Report**

Group Progress:

-Completed with the phase 1 and phase 2 requirements.

-Assigning roles and responsibilities to group members.

-Brainstormed for the ideas of the project

-Setup the git hub repository and got familiar with the IDE.

-Decided the communication platform to update the group members on daily basis.

Individual Progress: