Software Requirements Specification

for

CHATTY

Version 1.0

Prepared by Charles Bonoan and Arvin Shertukde

23 February 2021

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this document is to create an online chatroom application where users can find other users who have the same interests as one another and be able to connect and chat about those interests.

1.2 Document Conventions

The font we use in our document is Open Sans with a default size of 12, the headings are bolded and use a size of 14, and the section headings are not bolded but are size 20.

1.3 Intended Audience and Reading Suggestions

This project is intended for anyone who has access to internet and able to use a browser that will support our application i.e. Google Chrome, Firefox, etc.

1.4 Project Scope

The purpose of the web application is to create multiple chat rooms that allow people to talk with others who have similar tastes. We will have the user create a guest username that they can use to talk with others.

2. Overall Description

2.1 Product Perspective

Chatty will be developed for CSUB students who want to chat with other students who may find the same interest as one anothers. During this time of quarantine, college students, especially first years, will find it difficult to make new friends and create connections. Chatty hopes to lessen the strain of finding connections through their college years.

2.2 Product Functions

The first major product function is to create a webpage that will have multiple chat rooms. These chatrooms will have to be labeled for their different purposes, for example one chat room could be for discussing movies while another could be for discussing video games.

2.3 User Classes and Characteristics

Users of our software will be able to choose from a list of chat rooms and be able to connect and chat with anyone in said chat room. Each chat room will be based on a specific interest so that each user can find connections with people who have the same interests.

2.4 Operating Environment

Operating environment for the chatroom is shown below:

- Operating Systems: any operating system that supports a browser(ex. Windows, Linux, etc.)
- Browser run application(ex. Google Chrome, Windows Edge, Internet Explorer, Firefox, etc.)

2.5 Design and Implementation Constraints

Since our application will be developed on CSUB's Odin server that is managed by the Computer Science department, we need to make sure we are developing our application in a way that does not overwhelm the Odin server when people are using the application.

2.6 User Documentation

There will be a support page available so that the user can submit tickets if they are having problems. There will also be a page to guide the user on how to use the chat rooms.

2.7 Assumptions and Dependencies

Since we intend for this application to be used for just CSUB students, opur users are expected to sign in using their student ID. Users can also choose a username to be displayed when they are chatting in one of the chat rooms.

3. External Interface Requirements

3.1 User Interfaces

- Front-end software: HTML5, CSS3, JS, Bootstrap (CSS framework)
- Back-end software: Python 2.17.6
    - This version of Python is what is supported on CSUB's Odin server, meaning whatever library we decide to use, we need to make sure it is compatible with this version of Python.

3.2 Software Interfaces

Software Used	Description
Browser	We are going to develop our application to where it is able to run on all modern and common browsers such as Google Chrome and Firefox.
Python Flask	Flask is a Python web framework that will help us display our web content using its collection of modules and libraries.

3.3 Communication Interfaces

Client based interface where we will have a front end application, for now we do not plan to have any backend features. The application will also support all browsers, and users will be able to interact with one another through the chatrooms.

4. System Features

4.1 Chatroom List

Instead of being constrained to having one chat room to join, users can choose from several chat rooms, each having a different personality of users in it. Users can also choose to back out from a chatroom and choose to join another if they wish to.

4.2 Language Filter

We want users to be comfortable when in a chatroom and speaking to people, so we have implemented a way to scan chat messages for inappropriate words or phrases that may seem harmful and offensive to other students. That way everyone can have a safe environment to create new connections.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

It goes without saying that this application is going to be a real-time chat application meaning that users will be able to see new messages in real time. We also need to develop this app in a way that it will not take up a lot of resources from the Odin server in terms of memory, so we will be deleting chat room messages every new day.

5.2 Safety Requirements

For safety, we will not allow non CSUB users to be able to access the web application, in order to guarantee that the people who are chatting with others will not have alternative motives.

5.3 Security Requirements

We will implement a way for students to authenticate themselves in order to use the application. This way no one known to CSUB and possible have malicious intent to use the chat application.

5.4 Software Quality Attributes

- Availability: people associated with CSUB who have access to a browser
- Correctness: The user's message should be sent to the correct chatroom and other users in the chatroom should receive the messages
- Maintainability: The server admins should make sure all chat rooms are functioning and are on topic
- Usability: The chatrooms can be used as much as users want to be able to talk to others