

Software Requirements Specification

Social Networking for Medical Professionals

1 Introduction

1.1 Purpose

The web application is intended only for medical professionals where they can share their questions, comments, pictures related to the medical field, share their experiences related to treating an individual without unveiling the patient's information.

This document is meant to delineate the features of the web application, to serve as a guide to the developers on one hand and a software validation document for the prospective client on the other.

1.2 Scope

We describe what features are in the scope of the software and what is not in the scope of the software to be developed.

1.2.1 In-Scope:

- A. Managing the information of Users, which will include Name, E-mail Address, Profile Picture and a Short Bio.
- B. Segregating the Content based on keywords and creating tags for each of them, which will help in ease of searching.
- C. Suggesting Colleagues to users who are closely related in terms of medical field work.
- D. For creating Posts, the software will provide some templates which will ease the work of the user.
- E. Suggesting professionals to answer types of questions.
- F. User Authentication in order to avoid misuse of the information present in the website.

1.2.2 Out-Scope:

- A. Patient-Doctor interaction or Patient-Patient interaction.
- B. Doctor certification is dependent on MCI (Medical Council of India) i.e. we don't perform any verification, we depend only on government records.

1.3 Definitions, Acronyms and Abbreviations

1.3.1 Acronyms and Abbreviations

- A. MCI: Medical Council of India
- B. SRS: Software Requirement Specification
- C. WWW: World Wide Web
- D. GUI: Graphical User Interface

1.3.2 Definitions

- A. **Question:** A set of text/images/videos* that requires a discussion or resolution, which is asked by one user to others.
- B. **Answer:** Like a question, except answer should be affiliated with some question.

- C. **Profile:** A set of securities, basically user information template.
- D. **Tag:** It's a medical term that has some significance to a question.
- E. **User Points:** It is the value that quantifies User trust, in other words we believe users with higher points more than lower points.
- F. **Post Points:** It is the popularity of question, i.e. to show trending posts.
- G. **Post:** A post includes questions, answers, polls, comments.
- H. **Colleague:** Like friends who follow each other, and all the posted items will be visible to each other.

1.4 References

Appendix A: Calculation of Post Points and User Points.

Appendix B: User Screens.

1.5 Overview

The rest of this SRS is organized as follows:

- Section 2 gives an overall description of the software. It gives what level of proficiency is expected of the user, some general constraints while making the software and some assumptions and dependencies that are assumed.
- Section 3 gives specific requirements which the software is expected to deliver. Functional requirements are given by various use cases. Some performance requirements and design constraints are also given.
- Section 4 gives some possible future extensions of the system.

2 Overall Description

2.1 Product Perspective

The web application is aimed towards providing a platform for medical professionals who are experts in various fields, to share their experiences, comments, answers, questions. The application will be a mix of stack overflow and social networking websites. The application should be user friendly and reliable for the above purpose.

The software will be a web application which will be accessible using the internet.

2.2 Product Functions

The web application should support the following use cases:

Class of Use Case	Use Case	Description of Use Case
Accessing	Accessing the web application	Creating a web session
Authorization	Sign In	Signing In into the account
	Change Password	Changing the user password
	Forgot Password	Password Recovery

Profiles	Create Profile	Creating a User Profile
	Edit Profile	Editing a User Profile
	Delete Profile	Deleting a User Profile
Posts	Add Post	Adding a new Post <ul style="list-style-type: none"> • Question • Answer • Comment • Polls
	Edit Post	Editing a Post written by the User
	Delete Post	Deleting a Post written by the User
Tags	Create Tag	Creating a new Tag
	Edit Tag	Editing an existing Tag
	User Personalization	Content based personalization
Colleagues	Sending Request	Sending a request to make another user as a Colleague
	Accepting/Rejecting Request	Accepting/Rejecting a request of another user to make a Colleague
	Removing Colleague	Removing a user as a Colleague
Computations	Post Points	Computing the popularity of a Post
	User Points	Computing the trust of a User
Machine Learning & Data Mining	Content Filtering	Filtering based on post content and tags
	User Personalization	Content personalized based on User interests
Notifications	Post Related	On Posted Threads
	Voting/Polling	Voting/Polling Questions

	Shared by Colleagues	Colleague creates a new Post
--	----------------------	------------------------------

2.3 User Characteristics

- A. The user is assumed to be familiar with the medical jargon and can comprehend the question and provide a clear answer.
- B. The user should have a professional medical background so that they can add useful information.

2.4 Principal Actors

The principal actor in this application will be the Medical Professionals who will be interacting with one another.

2.5 General Constrains

- A. For working with the application requires Internet connection.
- B. The users should be medical professionals, Non-medical professionals are not allowed to manipulate content.

2.6 Assumptions and Dependencies

- A. Working of the application is dependent on the availability of Internet connection.
- B. The user after verification is assumed to be a genuine medical professional.
- C. Initially the users are assumed to be of equal ranks and afterwards there will be different ranks provided based on the content added, shared.
- D. Information obtained from government is trusted.

3 Specific Requirements

3.1 Functional Requirements

We describe the functional requirements by giving various use cases.

3.1.1 Use Case Related to Accessing

3.1.1.1 Accessing

- i. **Primary Actor:** User/Public.
- ii. **Pre-Condition:** Internet connection available.
- iii. **Main Scenario:** The user accesses the web application by going to the website.
- iv. **Alternate Scenario:**
 - a. **Network Failure:** Browser Specific.

3.1.2 Use Case Related to Authorization

3.1.2.1 Login

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The user must have a Profile.
- iii. **Main Scenario:**
 - a. The user is prompted for Username and Password.
 - b. After the user has entered the credentials, the system verifies the given credentials.

- c. The home page of the user is shown.
- iv. **Alternate Scenario:**
 - a. **Invalid Credentials:** Let the user try for some more tries.
 - i. If a User with the given Username exists, inform that User about this login.
 - ii. Tell that the given credentials are invalid.

3.1.2.2 *Change Password*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must be logged in.
- iii. **Main Scenario:**
 - a. User initiates the password change.
 - b. User is prompted to enter the:
 - i. Old Password
 - ii. New Password
 - iii. Retype New Password
 - c. Application does authentication.
 - d. New Password will be registered for the User account.
- iv. **Alternate Scenario:**
 - a. **Wrong Old Password:** Tell the user that the old password is invalid.
 - b. **Invalid New Password:** Tell the user that the given new password doesn't satisfy the required conditions.
 - c. **Passwords don't Match:** Tell the user that the given passwords don't match.

3.1.2.3 *Forgot Password*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must have a Profile.
- iii. **Main Scenario:**
 - a. User initiates the password recovery.
 - b. User is prompted to enter their E-mail address which was used to create the profile.
 - c. Application sends a password recovery form to the mail.
 - d. The user is prompted to enter a new password and retype the password.
 - e. New Password will be registered for the User account.
- iv. **Alternate Scenario:**
 - a. **Mail ID not found:** Tell the user that no profile was created with this mail id.
 - b. **Password Fails:**
 - i. **Invalid New Password:** Tell the user that the given new password doesn't satisfy the required conditions.
 - ii. **Passwords don't Match:** Tell the user that the given passwords don't match.

3.1.3 Use Case Related to Profiles

3.1.3.1 *Create Profile*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must be logged in.
- iii. **Main Scenario:**
 - a. User initiates the create profile functionality.

- b. System asks the user for the profile name.
 - c. User enters the profile name.
 - d. An empty profile is created.
- iv. **Alternate Scenario:**
 - a. **Profile with same name exists:**
 - i. User is asked to enter a different Username.
 - ii. User enters a different Username.
 - iii. Empty Profile gets created.

3.1.3.2 *Rename Profile*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must be logged in.
- iii. **Main Scenario:**
 - a. User initiates the rename profile functionality.
 - b. System asks the user for a new profile name.
 - c. User enters a new profile name.
 - d. Profile gets renamed.
- iv. **Alternate Scenario:**
 - a. **Previous Profile name doesn't exist:** Renaming fails, the error message is displayed.
 - b. **Profile with same name exists:** Renaming fails, the error message is displayed.

3.1.3.3 *Delete Profile*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must be logged in.
- iii. **Main Scenario:**
 - a. User initiates the delete profile functionality.
 - b. System asks the user for the name of the Profile.
 - c. Profile gets deleted.
- iv. **Alternate Scenario:**
 - a. **Profile name doesn't exist:** Deletion fails, the error message is displayed.

3.1.4 Use Case Related to Posts

3.1.4.1 *Adding a Question*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must be logged in.
- iii. **Main Scenario:**
 - a. User initiates the add question functionality.
 - b. System prompts the user for the following details:
 - i. Question (basically markdown):
 - 1. Text
 - 2. Images
 - ii. (ii) Tags:
 - 1. At least a fixed number of tags must be selected
 - 2. User can add tags.
 - c. A new question post request is submitted and will be created.

- iv. **Alternate Scenario:**
 - a. **Empty Question:** New post request is not submitted.
 - b. User tag creation is validated, then only added to the system.

3.1.4.2 *Adding an Answer*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must be logged in and selected a question.
- iii. **Main Scenario:**
 - a. User initiates the add answer functionality.
 - b. System prompts the user for the following details:
 - Answer (basically markdown):
 1. Text
 2. Images
 - c. New answer is saved and posted.
- iv. **Alternate Scenario:**
 - a. **Empty Answer:** Answer is not submitted

3.1.4.3 *Adding a Comment*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must be logged in and selected a question/answer/comment.
- iii. **Main Scenario:**
 - a. User initiates the add comment functionality.
 - b. System prompts the user for the following details:
 - i. Answer (basically markdown):
 1. Text
 2. Images
 - c. New comment is saved and posted.
- iv. **Alternate Scenario:**
 - a. **Empty Comment:** Comment is not submitted

3.1.4.4 *Adding a Poll*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must be logged in and satisfies some criteria.
- iii. **Main Scenario:**
 - a. User initiates the add poll functionality.
 - b. System prompts the user for the following details:
 - Poll (basically markdown):
 1. Text
 2. Images
 3. Options
 4. End time of Poll
 - c. New Poll is saved and posted.
- iv. **Alternate Scenario:**
 - a. **Empty Poll:** Poll with empty options or empty text are not posted.
 - b. **Submitting same poll as others:** Same polls are not encouraged.
 - An automatic check will be conducted on every poll to match similarity.

- In special cases handed to manual investigators.
- c. Criteria on whom to give poll access is still vague but it depends on user voting.

3.1.4.5 *Editing a Post*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must be logged in.
- iii. **Main Scenario:**
 - a. User selects their post.
 - b. User initiates the edit post functionality.
 - c. System asks the user to enter new values after validating.
 - d. The modified post is saved.
- iv. **Alternate Scenario:**
 - a. **Empty Edits:** New post request is not submitted.
 - b. User tag creation is validated, then only added to the system.

3.1.5 Use Cases Related to Tags

3.1.5.1 *Create Tags*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must be logged in.
- iii. **Main Scenario:**
 - a. User selects add question, there is selects create tags.
 - b. System checks all the preexisted tags.
 - c. If tag is not present its creation request is accepted and:
 - Automatically verified against some data base.
 - Manually verified by few selected random users with special criteria.
 - d. Then tag is successfully created.
- iv. **Alternate Scenario:**
 - a. **Empty Tag:** Empty tag request is not accepted.

3.1.5.2 *Edit Tags*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must be logged in and met special criteria.
- iii. **Main Scenario:**
 - a. User selects tag.
 - b. Sends an edit suggestion to person who created tag and few other anonymous users.
 - c. If it is valid request its accepted.
 - d. Then tag is edited successfully.
- iv. **Alternate Scenario:**
 - a. **Empty Tag:** Empty tag request is not accepted.

3.1.5.3 *Content based personalization*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must be logged in.
- iii. **Main Scenario:**
 - a. User selects few interested words in profile.
 - b. With those words we select a tag that were similar.

- c. Based on those tags we try personalizing user questions.
 - d. The main idea is to use content-based filtering.
- iv. **Alternate Scenario:**
 - a. All user personalized questions were not only based on tags.

3.1.6 Use Cases Related to Colleagues

3.1.6.1 *Sending Request to become a colleague of another User*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must be logged in.
- iii. **Main Scenario:**
 - a. User first searches for the other Username and then selects them.
 - b. User then sends the request to the other user by selecting the option provided.
 - c. The system then verifies whether the user is there or not and send the request to the other user, once this is successful it then displays success message.

3.1.6.2 *Accepting/Rejecting the Request from a User*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must be logged in.
- iii. **Main Scenario:**
 - a. User after receiving a request will have the option to accept or ignore the request.
 - b. If the user chooses to accept the request. A message/notification will be sent to the sender that the recipient has accepted the request.
 - c. If the user chooses to ignore the request. Then the request will be deleted from the request list of the user. The sender will not be notified of this action. After some time, the request expired notification will be sent to the sender.

3.1.6.3 *Removing a Colleague*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** The User must be logged in.
- iii. **Main Scenario:**
 - a. User will have the option to remove an existing colleague by selecting the remove colleague option.
 - b. After selecting the option, the colleague will be removed from the list, but all the shared content will not be removed. No notification will be sent to the removed colleague.

3.1.7 Use Cases Related to Computations

3.1.7.1 *Compute Post Points*

- i. **Primary Actor:** Application.
- ii. **Pre-Condition:** The post should be posted by a genuine user.
- iii. **Main Scenario:** Depending on the upvotes and downvotes the post will be given some points and will be ranked among other posts with similar tags.

3.1.7.2 *Computation Status of the User*

- i. **Primary Actor:** Application.
- ii. **Pre-Condition:** User is listed in the application.

- iii. **Main Scenario:** Depending on the activity/ content involved in the application like sharing, commenting, posting questions/experiences, answering questions. The user will be awarded points and will be ranked based on the user activities in the application.

3.1.8 Use Cases Related to Data Mining

3.1.8.1 *Post Related*

- i. **Primary Actor:** Application.
- ii. **Pre-Condition:** User is signed up.
- iii. **Main Scenario:**
 - a. Posts, questions will be recommended to the user based on his discipline of work.

3.1.9 Use Cases Related to Notifications

3.1.9.1 *Post Related*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** User is logged in.
- iii. **Main Scenario:**
 - a. User initiates the follow post functionality for the post.
 - b. The system sends notification when there is an update like comments/likes in the specific post.

3.1.9.2 *Voting or Polling*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** User is logged in.
- iii. **Main Scenario:**
 - a. User initiates the follow the poll/vote functionality
 - b. System sends the stats of the following poll or voting created.

3.1.9.3 *Shared by Colleagues*

- i. **Primary Actor:** User.
- ii. **Pre-Condition:** User is logged in.
- iii. **Main Scenario:**
 - a. User follows the colleagues whom they like or want to know their updates.
 - b. When that specific user following person creates a post, the system sends a notification.

3.2 Performance Requirements

- A. Should run on any machine capable of running Chrome/Firefox or any web browser.
- B. Good, stable internet connection for correct operation.

3.3 Design Constraints

- A. **Scalability:** The application should support heavy user interaction with the application. It should support hundreds of online users concurrently.
- B. **Fault Tolerance:** Data should not become corrupted in case of system crash or power failure. The data should be eventually consistent.
- C. **Responsiveness:** The application should be responsive and there should be minimal delay while operating with the GUI.
- D. **Secure:** The information of the users should not be compromised.

4 Future Extensions

- A. Adding video types, i.e. mp3, oog, etc. to post.
- B. Showing adds to annoy users.

5 Appendix

5.1 User Points and Post Points

5.1.1 User Points

Based On:

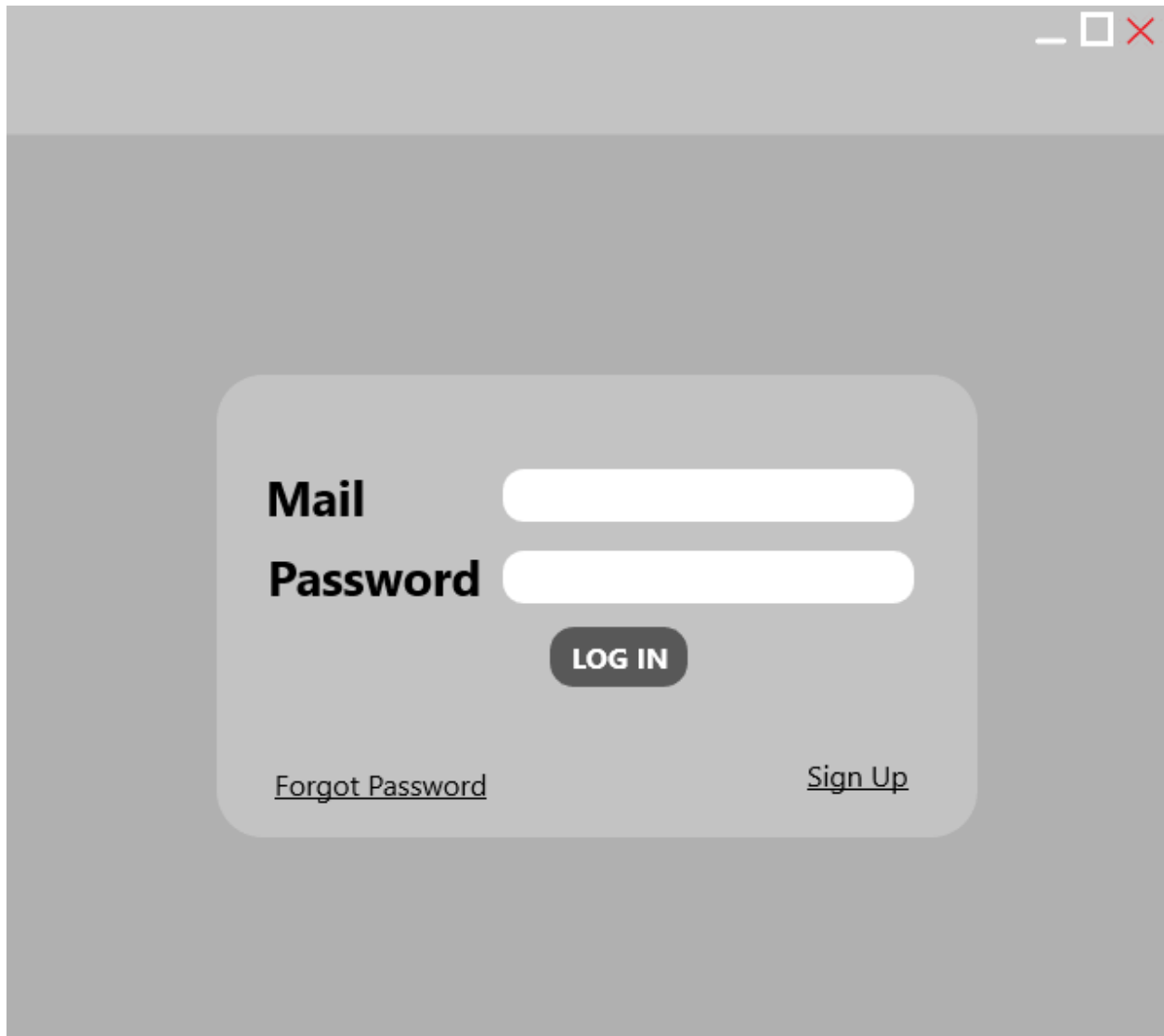
- upvotes/downvotes given directly to user
- Upvotes/Downvotes given to questions /comments/Polls/Answers by this user
- Usage of user
- Soon a mathematical equation will be decided that quantifies based on above parameters

5.1.2 Post Points

Based On:

- upvotes/downvotes given directly to post
- Upvotes/Downvotes given to post by whom i.e. user with more user points have more influence on post points
- Visits to post
- Posted by whom
- Soon a mathematical equation will be decided that quantifies based on above parameters

5.2 User Screens



A mockup of a login page within a window. The window has a light gray title bar with standard OS controls (minimize, maximize, close) in the top right corner. The main content area is a solid medium gray. Centered on the screen is a light gray rounded rectangle containing the login form. The form includes labels for 'Mail' and 'Password' in bold black text, each followed by a white rounded rectangular input field. Below the password field is a dark gray rounded button with the text 'LOG IN' in white. At the bottom of the rounded rectangle are two links: '[Forgot Password](#)' on the left and '[Sign Up](#)' on the right, both in a smaller black font.

Figure 5-1 Login Page

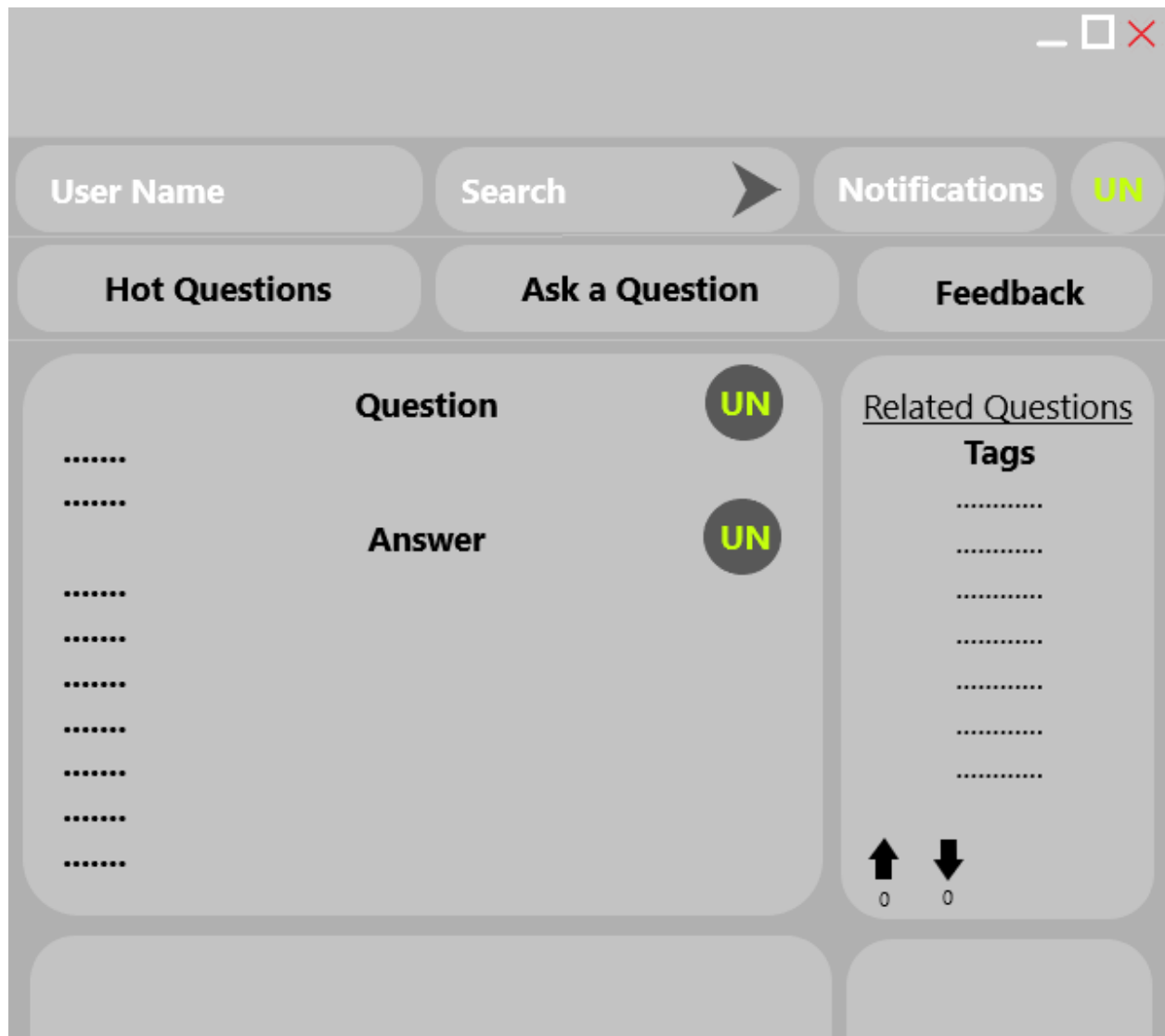


Figure 5-2 Dashboard