

# Fake Stack Overflow Specifications

Term Programming Project

CSE 316 – Fundamentals of Software Development

Fall, 2023

Assigned: Monday, 11/13/2023

Due: **Monday, 12/11/2023**

## Introduction

This document describes the specification of the Fake Stack Overflow Application. This application is inspired by the actual [stackoverflow.com](https://stackoverflow.com). The application will be developed using the [React](https://reactjs.org/) framework for the front end. For the backend, the application will use [MongoDB](https://www.mongodb.com/), a non-relational database. The web server will be in [Nodejs](https://nodejs.org/en/) using the [Express](https://expressjs.com/) framework for server-side routing. The application secures sensitive data such as passwords using the [bcrypt](https://www.npmjs.com/package/bcrypt) library in Node.

The specification is organized as a collection of anecdotal user stories. Please read each use case carefully and take note of each component (actors, pre-conditions, post-conditions, story, and exceptions).

## Use Cases

<b>Use-Case number</b>	1
<b>Use-Case Name</b>	Create Account
<b>Actors</b>	Unregistered user
<b>Preconditions</b>	The user has a working internet connection, is viewing the welcome page, and is not logged in.
<b>Postconditions</b>	A new account is created for the user with their username, a secret password, and an account name (email).
<b>Story</b>	The user arrives at the welcome page of the

	<p>application. The page shows options to register as a new user, login as an existing user, or continue as a guest user. They select the register as a new user option. On selecting this option, they enter their username, email, a secret password, and a password verification. The user presses the SignUp button. This saves the information in a database. They are then directed to a Login page.</p>
<b>Exceptions</b>	<p>No two users can create an account with the same email. The email should have a valid form. The typed password should not contain the username or the email id. Nicely styled feedback must be presented to the user if the account could not be created due to the above reasons or any other reason.</p>

<b>Use-Case number</b>	2
<b>Use-Case Name</b>	Login
<b>Actors</b>	Registered user
<b>Preconditions</b>	The user has a working internet connection, is viewing the application page, and is not logged in.
<b>Postconditions</b>	Login successful or failed.
<b>Story</b>	The user arrives at the welcome page of the application. This page has options to register as a new user, login as an existing user, or continue as a guest user. They select the login as an existing

	user. On selecting this option, they are asked to login with their email and password. Upon pressing the login button, they are taken to the home page if login is successful.
<b>Exceptions</b>	The user enters an unregistered email or an incorrect password then the application should report back appropriate feedback to the user on the same page.

<b>Use-Case number</b>	3
<b>Use-Case Name</b>	Logout of account
<b>Actors</b>	Registered user
<b>Preconditions</b>	The user has a working internet connection, has loaded the application page, and is currently logged in.
<b>Postconditions</b>	The user logs out of the application.
<b>Story</b>	The user is at a page with a logout button. They click on the logout button and the user is taken back to the welcome page if log out was successful.
<b>Exceptions</b>	The application should give appropriate feedback to the user if the log out failed.

<b>Use-Case number</b>	4
<b>Use-Case Name</b>	Home Page
<b>Actors</b>	Guest user
<b>Preconditions</b>	The user has a working internet connection, is viewing the home page, and is not logged in.
<b>Postconditions</b>	The home page.
<b>Story</b>	<p>The user has entered the home page as a guest user.</p> <p>The home page displays a menu with options to view all questions, view all tags, and a search box to search for questions and answers. Further, the user should be able to view the questions in <i>Newest</i>, <i>Active</i>, and <i>Unanswered</i> order. The meaning of these options are the same as in HW1-3. Refer to those documents.</p> <p>By default, the home page shows a list of all questions in <i>Newest</i> order. For each question in the list, it displays the question title, question summary, the list of associated tags, the no. of times it has been viewed and voted, the no. of answers it has, the username of the questioner and the date it was posted. The format of the date and the username is the same as in HW1-3. The question title is a link which when pressed shows the answers page.</p> <p>The page displays 5 questions at a time. The list of questions is scrollable to fit all the 5 questions if</p>

	they overflow. For more than 5 questions 2 buttons – <i>next</i> , and <i>prev</i> are displayed at the bottom of the list. The <i>next</i> button displays the next 5 questions, <i>prev</i> displays the previous 5. Pressing button <i>next</i> should display the first 5 questions when the last 5 questions are shown. However, when displaying the first 5, <i>prev</i> is disabled. Both buttons are outside the scrollable list of questions.
<b>Exceptions</b>	The application should show an appropriate error message to the user in case of a system failure or a communication failure. In case of an error, the user should be given a way to go back to the welcome page and restart.

<b>Use-Case number</b>	5
<b>Use-Case Name</b>	Home Page
<b>Actors</b>	Registered user
<b>Preconditions</b>	The user has a working internet connection, is viewing the home page, and is logged in.
<b>Postconditions</b>	The contents of the home page or the welcome page if there is an error.
<b>Story</b>	The home page displays a menu with options to view all questions, view all tags, and a search box to search for questions and answers. Further, the user should be able to view the questions in

	<p><i>Newest</i>, <i>Active</i>, and <i>Unanswered</i> order. The meaning of these options are the same as in HW1-3. Since the user is logged in, the home page has an option to add a new question.</p> <p>By default, the home page shows a list of all questions in the database in <i>Newest</i> order. For each question in the list it displays, the question title, question summary, the list of associated tags, the no. of times it has been viewed and voted, the no. of answers it has, the username of the questioner and the date it was posted. The format of the date and the username is the same as in HW1-3. The question title is a link which when clicked shows the answers page.</p> <p>The page displays 5 questions at a time. The list of questions is scrollable to fit all the 5 questions if they overflow. For more than 5 questions 2 buttons – <i>next</i>, and <i>prev</i> are displayed at the bottom of the list. The <i>next</i> button displays the next 5 questions, <i>prev</i> displays the previous 5. Pressing button <i>next</i> should display the first 5 questions when the last 5 questions are shown. However, when displaying the first 5, <i>prev</i> is disabled. Both buttons are outside the scrollable list of questions.</p>
<b>Exceptions</b>	<p>The application should show an appropriate error message to the user in case of a system failure or a communication failure. In case of an error, the user should be given a way to go back to the welcome page and restart the application.</p>

<b>Use-Case number</b>	6
<b>Use-Case Name</b>	Searching
<b>Actors</b>	Registered user, Guest user.
<b>Preconditions</b>	The user has a working internet connection, and is viewing the home page.
<b>Postconditions</b>	The user is shown a set of questions that match the search criteria.
<b>Story</b>	<p>In the search box, the user enters search text. The constraints for valid search text are the same as HW1-3. The default display order is <i>Newest</i>. Users should be able to change the display order by selecting <i>Active</i> or <i>Unanswered</i>.</p> <p>The search results are displayed 5 at a time with buttons to view the next and previous as described previously.</p>
<b>Exceptions</b>	Show “no results found” in case of search failure.

<b>Use-Case number</b>	7
<b>Use-Case Name</b>	All Tags
<b>Actors</b>	Registered user, Guest user.
<b>Preconditions</b>	The user has a working internet connection, and is viewing the home page.

<b>Postconditions</b>	Shows all tags in the database.
<b>Story</b>	When the user selects the <i>Tags</i> option from the menu, a set of all tag names is displayed to the user. Each tag name in the set is a link and shows the no. of questions associated with the tag. Upon clicking a link, the set of questions associated with the tag name is displayed.
<b>Exceptions</b>	Show an appropriate message if there is a system or communication failure below the horizontal menu. The message should be stylized so the message is visible clearly to the user.

<b>Use-Case number</b>	8
<b>Use-Case Name</b>	New Question
<b>Actors</b>	Registered user.
<b>Preconditions</b>	The user has a working internet connection, is logged in, and has clicked on the add new question option in the home page.
<b>Postconditions</b>	The new question created by the user is added to the database.
<b>Story</b>	<p>The user sees a form to add a new question. The form has the following fields:</p> <ul style="list-style-type: none"> <li>- A field to enter the question title (max. 50 characters).</li> <li>- A field to enter the question summary (max. 140 characters).</li> </ul>



	<ul style="list-style-type: none"> <li>- A field to enter the question text.</li> <li>- A field to enter tag names. Tag names are separated by whitespace. A new tag name can only be created by a user with at least 50 reputation points.</li> <li>- A button to create a new question.</li> </ul> <p>Pressing the new question button, saves the question to the database and displays the home page, which displays the new question along with existing questions. If the questions are displayed in <i>Newest</i> order then the new question is the first question displayed.</p>
<b>Exceptions</b>	An appropriate error message should be displayed if any specified constraint is not met.

<b>Use-Case number</b>	9
<b>Use-Case Name</b>	Answers
<b>Actors</b>	Guest user.
<b>Preconditions</b>	The user has a working internet connection and has clicked on a question in the home page as a guest.
<b>Postconditions</b>	Show the question details and all its answers.
<b>Story</b>	<p>The user has entered the Answers page and sees a menu described in the home page use case.</p> <p>The page displays the question title, the total no.</p>

	<p>of answers, and the total no. of views (including the current view). Below this information, the page displays the question text, the set of tags, and the username of the user who asked the question along with the date and time the question was asked. It also displays the no. of votes the question has received. The set of comments for a question is listed below the question text and its tags set. (see later for <i>comments</i> use case.)</p> <p>Below the question details section, the page lists a scrollable set of all answers for the question. The most recent answer should appear first. Each answer has the answer text, the no. of votes received, and the username of the answer giver along with the date and time. It displays 5 answers at a time with two buttons – <i>next</i> and <i>prev</i> to display the next 5 answers and the previous 5 answers, respectively. The <i>next</i> button displays the first 5 when the last 5 answers are shown and the <i>prev</i> button is disabled when the first 5 answers are shown. The buttons should be outside the scrollable list of answers.</p> <p>Answers also have a set of comments. They should be listed below each answer.</p>
<b>Exceptions</b>	<p>An appropriate error message is displayed if the expected data could not be rendered due to system or communication failure.</p>

<b>Use-Case number</b>	10
<b>Use-Case Name</b>	Answers
<b>Actors</b>	Registered user.
<b>Preconditions</b>	The user has a working internet connection, is viewing the answers page for a particular question, and is logged in.
<b>Postconditions</b>	Show questions details and all its answers.
<b>Story</b>	<p>The registered user is viewing the Answers page and sees a menu described in the home page use case.</p> <p>The page displays the question title, the total no. of answers, the total no. of views (including the current view), and the total no. of votes. Below this information, the page displays the question text, its tags, and the username of the user who asked the question with the date and time the question was asked. The set of comments for a question is listed below the question text and its tags. (see later for <i>comments</i> use case.)</p> <p>Below the question details section, the page displays a scrollable list of all answers for the question. The most recent answer should appear first. Each answer has the answer text, the no. of votes received, and the username of the answer giver along with the date and time. It displays 5 answers at a time with two buttons – <i>next</i> and <i>prev</i> to display the next 5 answers and the previous 5 answers, respectively. The <i>next</i> button displays the</p>

	<p>first 5 when the last 5 answers are shown and the <i>prev</i> button is disabled when the first 5 answers are shown. The buttons should be outside the scrollable list of answers.</p> <p>A button to add new answers is displayed below all the answers, in a separate section. This button is outside the scrollable list of answers.</p> <p>Answers also have comments. They are listed below each answer.</p> <p>The question and each answer has options to upvote and downvote the question or answer. Upvoting increases the vote by 1 and downvoting decreases the vote by 1. Upvoting a question/answer increases the reputation of the corresponding user by 5. Downvoting a question/answer decreases the reputation of the corresponding user by 10. A user can vote if their reputation is 50 or higher.</p>
<b>Exceptions</b>	<p>An appropriate error message is displayed if any of the constraints above are violated or there is a database or a communication error.</p>

<b>Use-Case number</b>	11
<b>Use-Case Name</b>	Comments
<b>Actors</b>	Guest user.
<b>Preconditions</b>	The user has a working internet connection, and is seeing the answers page for a question as a guest.
<b>Postconditions</b>	Shows all comments.
<b>Story</b>	A question or an answer has comments. The most recent comment is displayed first. For both questions and answers, 3 comments are displayed at a time with the username of the commenters and the number of votes for the comment. Two buttons – <i>next</i> and <i>prev</i> display the next 3 comments and the previous 3 comments, respectively. The <i>next</i> button shows the first 3 comments when the last 3 comments are shown and the <i>prev</i> button is disabled when the first 3 comments are shown.
<b>Exceptions</b>	N/A

<b>Use-Case number</b>	12
<b>Use-Case Name</b>	Comments
<b>Actors</b>	Registered user.
<b>Preconditions</b>	The user has a working internet connection, is logged in, and is seeing the answers page.

<b>Postconditions</b>	Shows all comments.
<b>Story</b>	<p>For any set of comments, the most recent comment is displayed first. Three comments are displayed at a time with the username of the commenters and the number of votes the comment has received. Two buttons – <i>next</i> and <i>prev</i> display the next 3 comments and the previous 3 comments, respectively. The <i>next</i> button shows the first 3 comments when the last 3 comments are shown and the <i>prev</i> button is disabled when the first 3 comments are shown. An input field is shown below the comments to capture a new comment by the logged in user. The new comment is added to the database and displayed on pressing enter.</p> <p>A logged in user can upvote a comment if they find the comment useful. Upvoting a comment increments the votes of a comment by 1. A button to upvote a comment is displayed with a comment so logged in users can show their approval. Downvoting on comments is not allowed.</p>
<b>Exceptions</b>	<ul style="list-style-type: none"> <li>- If a new comment is more than 140 characters then do not add it to the database and display an appropriate error message to inform the user.</li> <li>- If a new comment is added by a user with less than 50 reputation, display an appropriate message to inform the user and reject the comment.</li> <li>- Voting on a comment has no reputation constraints.</li> </ul>

<b>Use-Case number</b>	13
<b>Use-Case Name</b>	New Answer
<b>Actors</b>	Registered user.
<b>Preconditions</b>	The user has a working internet connection, is seeing the new answers page, and is logged in.
<b>Postconditions</b>	The new answer posted by the user is added to the database.
<b>Story</b>	The user has entered the new question page. They see a menu as described in the home use case. A form with a text field is shown for the user to enter their answer and a button which when pressed saves the answer in the database. After posting the answer the user is taken back to the answers page for the question where they can see the new answer.
<b>Exceptions</b>	An appropriate error message should be displayed on this page if a database error or communication error happens.

<b>Use-Case number</b>	14
<b>Use-Case Name</b>	User Profile
<b>Actors</b>	Registered user.

<b>Preconditions</b>	The user has a working internet connection, is viewing the user profile page, and is logged in.
<b>Postconditions</b>	The user profile of the currently logged in user.
<b>Story</b>	<p>The page displays a menu as described in the home use case. In the main section of the page, the length of time the user has been a member of fake stack overflow and the reputation of the user is shown. Below this information, a set of question titles asked by the user is listed. Each question title is a link which when clicked shows the <i>New question</i> page. In this page the user can modify the existing question and post it again or delete it. The form displays the existing information for the question in appropriate fields. Note posting modifications is not considered a new question. Deleting a question will delete all answers and comments associated with it.</p> <p>The menu also shows links to view all tags created and all questions answered by the user. When a link is clicked, the corresponding set of tags and answered questions are listed.</p> <p>The set of tags are displayed in the same format as described in the <i>Tags</i> page. Each entry in the list of tags being displayed has an option for the user to delete or edit the tag. If a user deletes a tag, it will not be shown with a question. However, a tag can be edited or deleted only if it is not being used by any other user.</p> <p>The questions answered by the user are displayed in the same format as <i>Newest questions</i> in the</p>



	<p><i>Home</i> page. Clicking a question link shows the answers page for that question. Their answer/s for the question is displayed first followed by the rest in <i>Newest</i> order. A user can edit or delete the answer. If a user deletes an answer then all its votes and comments are also deleted. The changes should be reflected in the <i>home</i> page and <i>questions</i> page appropriately.</p>
<b>Exceptions</b>	<p>Display an appropriate message if the user has no questions, answers, or tags.</p> <p>Display an error message if a user performs an invalid action as defined in the user story.</p>

<b>Use-Case Name</b>	User Profile
<b>Actors</b>	Admin user.
<b>Preconditions</b>	The user has a working internet connection, is viewing the user profile page, and is logged in as admin.
<b>Postconditions</b>	The user profile of an admin user.
<b>Story</b>	<p>The page displays a menu as described in the home use case. In the main section of the page, the length of time the admin user has been a member of fake stack overflow and the reputation of the user is shown.</p> <p>The main section lists all users in the system. Each user is a link which when clicked takes the admin</p>

	to the corresponding user's page. An admin has the same read and write permissions as the user on the user's profile. Each user link has a delete button next to them. Clicking delete will remove the user and all their information from the system.
<b>Exceptions</b>	<p>Display an appropriate message if the system has no users.</p> <p>Display a warning message asking for confirmation to prevent an admin from accidentally deleting a user.</p>

## System Requirements

You must use the React framework along with CSS to design and implement the front end. You can reuse and extend the CSS and React components you created for the homework assignments. The **react app should run at localhost:3000**.

The Express web server should run on Node and should connect to a backend database. The **server should run at localhost:8000**.

For the backend, you must use MongoDB. The web server should connect to a local instance.

For MongoDB, the server-side script should connect to **mongodb://127.0.0.1:27017/fake\_so**. Your server should set up the backend documents based on the schema you have designed.

Include a script called *server/init.js* in your repository that will be run to create initial data in your schema. This script will be run using Node before starting the server. The initial data must contain the user profile for admin. The username and password for an admin user must be provided as the first argument to *server/init.js*. You must use these credentials to create a user profile for admin in the database.

You should provide a UML class diagram that documents the objects in your system and how they are related to each other. **Save the image as a PNG/JPG in the *image* directory of your repository.**

User passwords need to be stored securely in the database. Hence, **we should hash the passwords and store their hashes instead of the original passwords.** To this end, install and use the [bcrypt](#) library in Node.

Your application should be designed to handle requests from multiple users. Use the event loop and asynchronous features in JavaScript to enable this most efficiently.

## **The Repository**

The repository has a directory for **server** and **client**. The file **server/server.js** is the main launching point of the application server. The **client** directory hosts the front-end resources of your application.

**The *README.md* file should document your application and how to run it. This file must contain instructions on how to reproduce and run your application on another computer. It must also contain the**

**list of features implemented by each member of the team. Without this you will not receive credit.**

You are allowed to use any library you deem fit. Make sure they are part of **packages.json** so we can install it during evaluation.