Project Report

Diego Moscoso

CMPT 353 – Winter 2023

Design Report

Database Design

For the database I decided to use 3 tables. One table will keep record of users registered in the app, another for the messages send to the server, and one for the channels created.

The users table will hold the following values for each user: user_id (integer), user_name (string) and user_password (string). The app uses this table to match the user_name and user_password passed through the log in feature and then give access to other pages keeping the account number for future features that the user may interact with. For now, the user cannot change its name or password. The account number/id is given when the user is created. There is one admin user with user_name(admin) and user_password(admin).

The messages table will hold the messages of all the users in all the chatrooms. This table will hold the following values: message_id (integer), message_text (string) max characters 80, channel_id (integer), user_id(integer), message_ref_id(integer), message_likes(integer), message_dislikes(integer). For now, a reply is a message with an origin other than empty.

The channels table will hold the channels available in the service. This table will hold the following values: channel_id(integer), channel_name(string) and member_id(integer). The member_id may be useful later to show only channels that the sign in user created or is part of, for now all channels are available for all registered users.

React Design

This app has a total of 4 possible pages to visit if user is admin otherwise, only 3 pages are possible to visit for the regular user. The regular user can create a new channel and give it a name. Inside each channel the user can send a message, reply to an existing message and react by like or dislike to existing messages. Admin user is capable of all the regular user actions and, delete users, delete channels, and delete specific messages. NOTE: If the admin deletes a user or deletes a channel all the messages related to the user, or the channel are also deleted.

Three pages are available to visit for the regular user. Landing, here a user can log in or register. Channels, here the user can view available channels, select a channel, and create a new channel. Chatroom, here the user can view past messages, post new messages, reply to messages, react with a like or dislike to a message.

If a non-registered user tries to visit any page, the page will reload and change to the landing page. The page does not hold values in the sessionStorage, localStorage or cookies, if user refreshes the browser the session will end, and the page will load at Landing.

App Testing

Database Testing

The server and the queries done to the database were tested by working with MySQL Workbench. Every query done was displayed in the workbench making easy to spot any error when making any request.

React Testing

The client was tested using both types of accounts. The followed process was to create five different users.

Every user would try to create a new channel. Select the channel. Post a new message. If another channel exists, select the channel, and reply and react to one message. Log out, log in again and repeat steps.

Once every user has posted at least five messages, react, and reply to other five we change accounts to the admin.

Once again, we test the regular capabilities, create channels, create messages, react and reply. Once done that part, admin would move around the pages and delete one message from every channel. Repeat until all channels have 2 messages. Now delete 2 channels. With the help of MySQL Workbench, review deleted channels and messages of said channels. At last, delete all the users and check the database again. The messages remaining should be of the available channels of still registered users.

Link to Video Presentation

https://usask.cloud.panopto.eu/Panopto/Pages/Viewer.aspx?id=4d25849f-abcd-41e1-ab28-afdd00319a42