# Garv Singh Rawlot

Extension:

## Basic Extension 1: User Page

For this extension, I created a new ejs file named user-info.ejs. It basically takes input from the user.ejs page and uses queries to the database written in the main.js file to output a table of information about the user selected including their topics.
To do this, I first added a POST route in the main.js file for the user-info.ejs file that stores the input from the user.ejs file in a var ‘term’. It then runs a query that checks if the user is a member of a topic and renders the page “user-info” with the data collected in the query. The file ‘user-info’ uses this data to display all the rows in a table using a loop that goes through all the elements.
This extension takes the username as input when you click it and displays that user’s:

- User ID
- Firstname
- Surname
- Username
- Country
- Topic Title

## Basic Extension 2: Topic Page

This extension is similar to the one before. For this, I made a new file named ‘topic-info.ejs’. It takes input from the topic.ejs page and uses queries written in the main.js file to output information as a table. Although, it prints the topic title and description separately.
To do this, I first added a POST route in the main.js file for the topic-info.ejs file that stores the input from the topic.ejs file in a var ‘term’. It then runs a query that looks for all the users associated with the topic and adds them to the datalist. Then, the file ‘topic-info.ejs’ uses this data to display all the rows in a table using a loop that goes through all the elements.
This extension takes the title of the topic as input when you click it and displays:

- Topic_title
- Topic_description
- User_ID
- Username
- Firstname
- Surname
- Country

Out of these, the bottom 5 are presented in rows and the top 2 are presented on the top as headlines to minimize repetition and add to the app’s usability.

# Basic Extension 3: Styling

To style the page, I took the help of bootstrap. I added the online stylesheet and the min.js bundle available online to each views page in the web application. After that, I started using the basic pre-defined classes to make my tables and fonts look better. I also used bootstrap to create a minimal navigation bar for the web app. In particular, I have:

- Changed the font style, font colour and font behaviours for some of the fonts where necessary
- Added a better-to-read dark colour scheme that stands out, is easier to read and good to look at.
- Added a navigation bar with hover features that change the colour of the page link when active.
- Added responsiveness to the website by making it good for phones as well.

I also kept the code aesthetically pleasing by using vscode extensions like prettier. This gave the code a more uniform look.