5CCS2SEG Lab Project

Jaman Salique – 1625442

David Valley – 1132032

Jordan Watson - 1616076

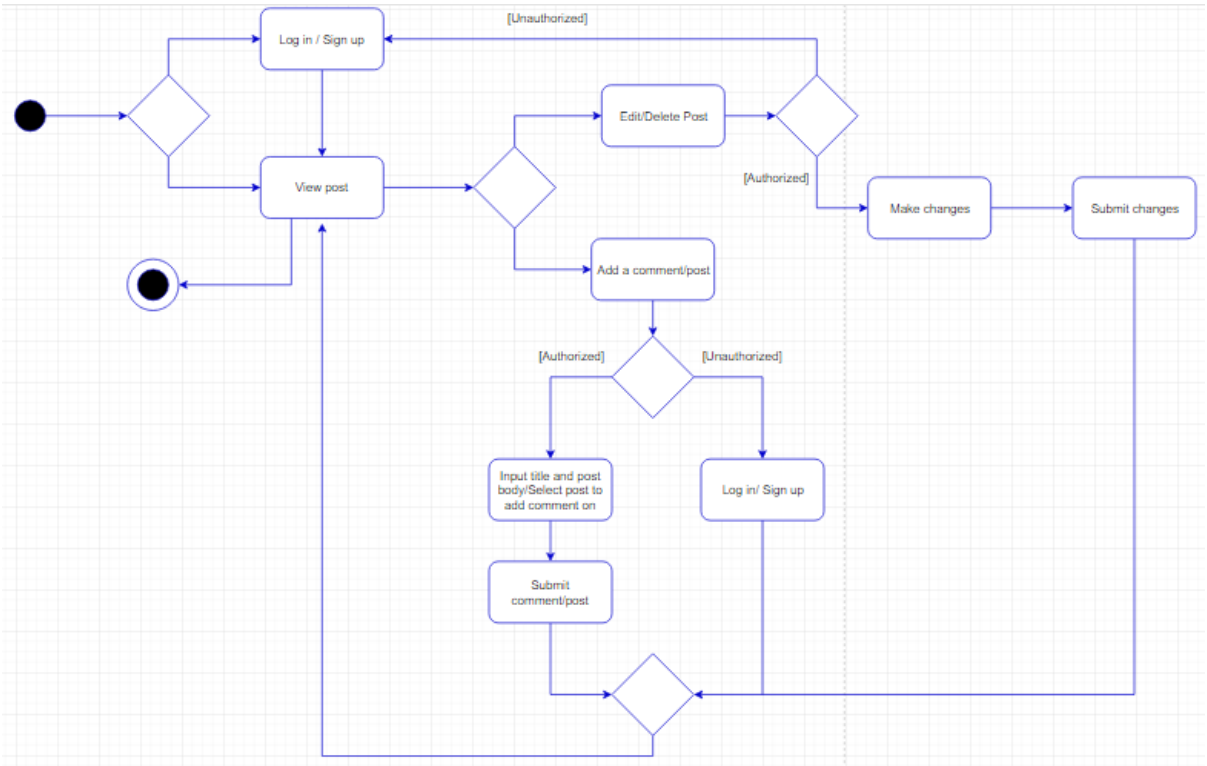
Kai Rycroft – 1618808

Adeike Christopher Kobena Showunmi - 1540594

In this project we developed a web-based question answering forum. The application is aimed at King's College London students who have questions about their course and can be answered by anyone, students and staff alike. Our application provides a means to post a question, to read questions and answers to questions and to post an answer/comment to an existing question. Also our application has the means to identify the author of a question or answer, and finally the identity of a user is verified.

Activity Diagram

To demonstrate a basic way of using our application a visual representation is best for this. We made an activity diagram to show the different activities and possible routes the user can make when using our application.



As you can see in the activity diagram above you can see the different possible routes a user can take when using the application. For example if the user is currently logged out and they try to Edit/Delete a post then they are redirected to the login page where they can sign up or log in, however if they were already logged in then the user can carry on making changes to a post or they can delete the post they made.

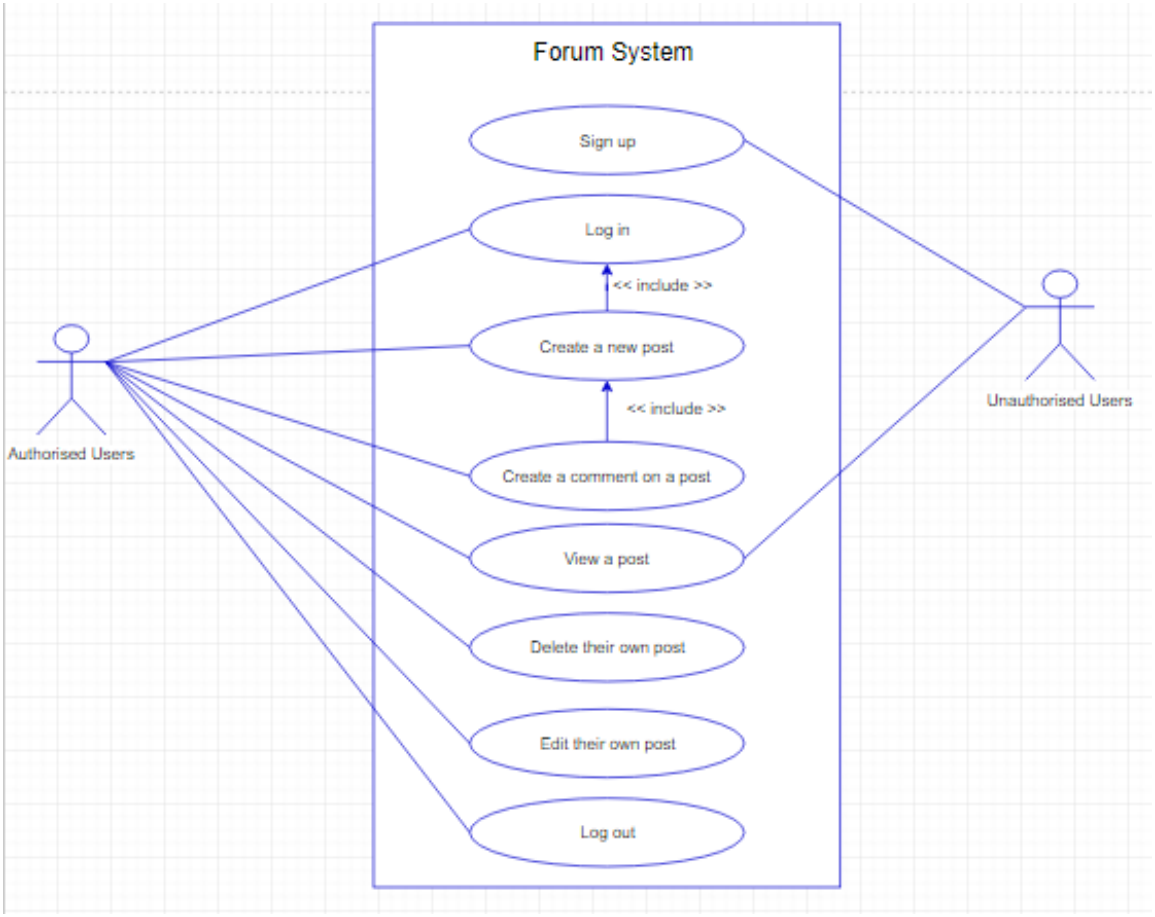
Use cases

To show interactions between different actors in our application we created a use case diagram which shows the different actors and the different operations they can do.

Our application does the following:

- Users can create a new post only if they are logged in
- Users can edit or delete posts they made only if they are logged in
- Users can add comments/answers to a post only if they are logged in
- Users can view posts and comments even if they are not logged in
- Users can create a new account by signing up
- Users can log out

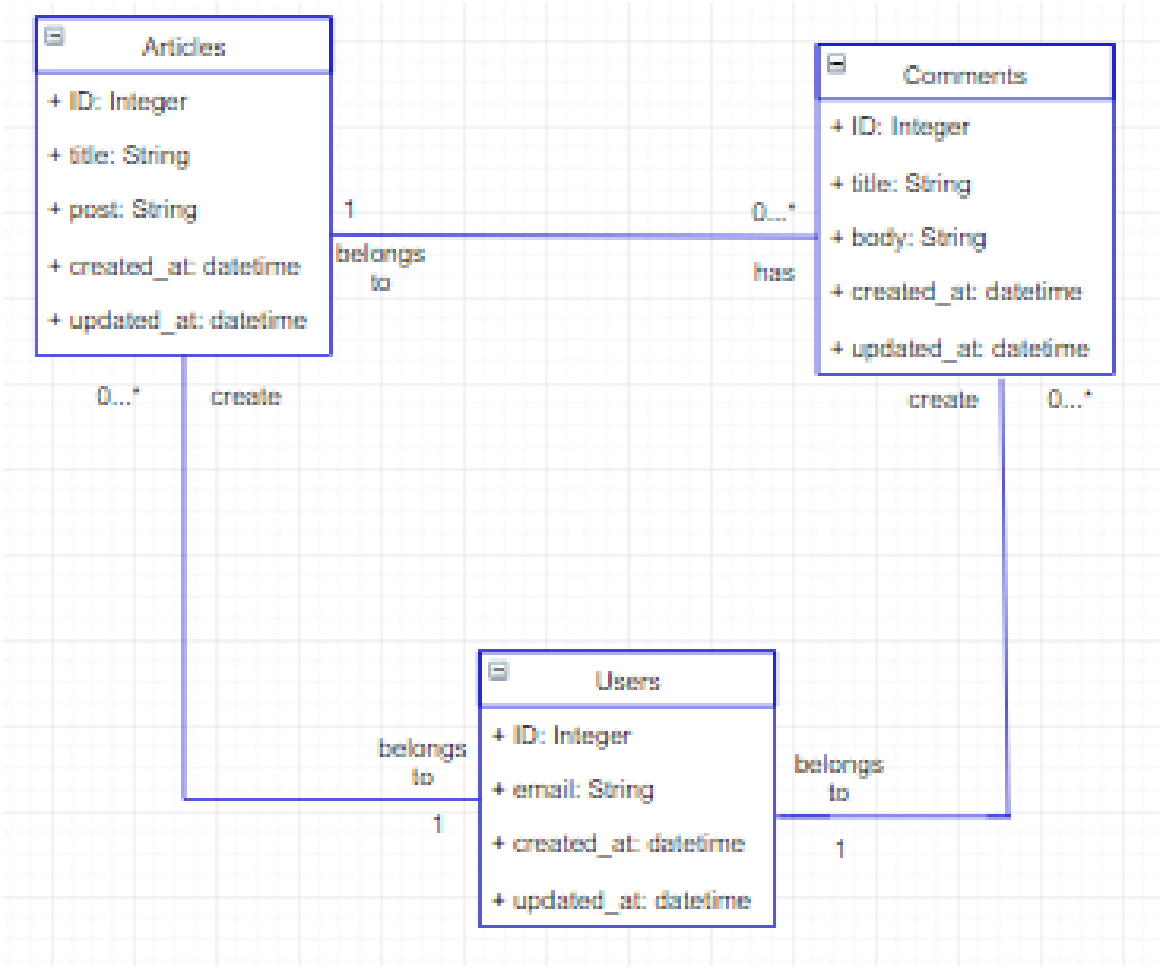
The use case diagram below shows the above:



In the use case diagram above there are two different actors an authorised user and unauthorised user. An unauthorised user is someone who has not signed up yet (their details are not in the database) and therefore can only view a post and sign up, whereas the authorised user can have access to more operations like: creating a post, deleting and editing the posts they made, logging out, logging in and creating comments on posts.

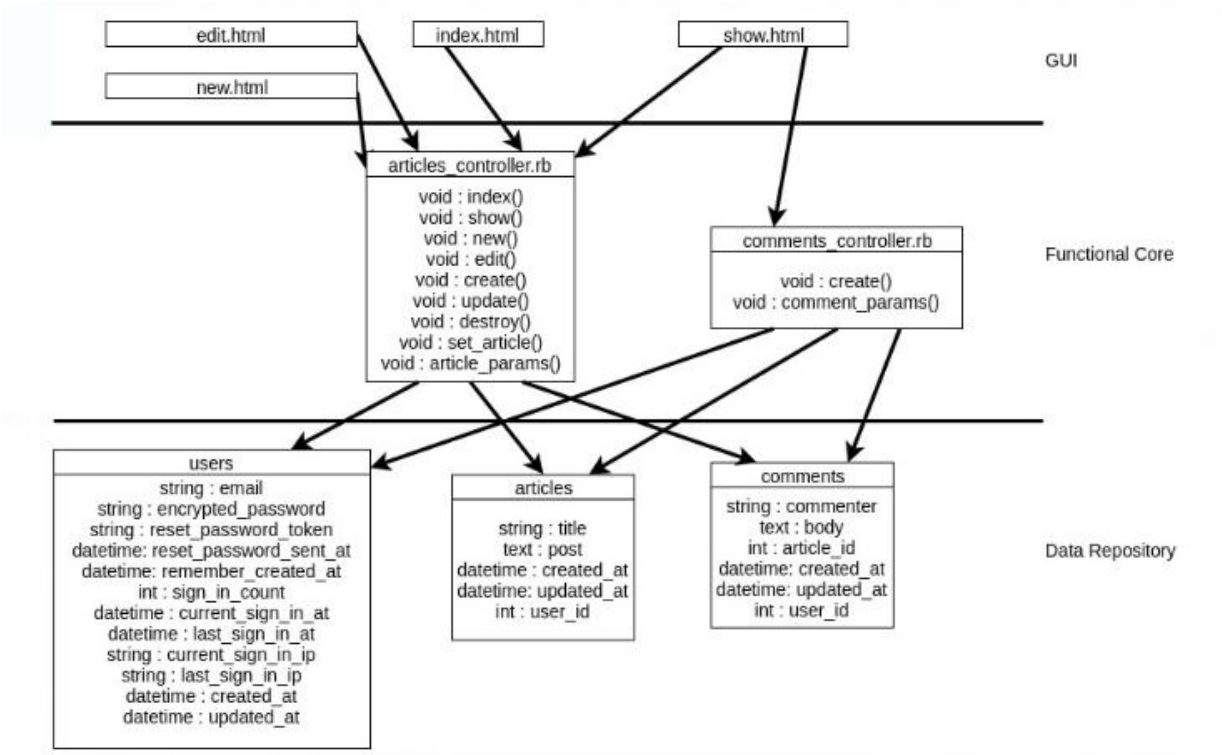
Class Structure

To help show the class structure of our application we decided to create a class diagram. Our classes are Users, Articles (the posts), Comments. To help save effort and time we used the Devise gem which gives us a sign in / sign out functionality. The Devise gem also defines database tables for the user which was extremely helpful. The class diagram below shows how each class relates to each other:



Architecture Diagram

We made an architecture diagram to show the different sub components and the dependencies between them. This helps form modules within our system. The architecture diagram below shows this.



As you can see in the diagram above there are 3 sections GUI, Functional core and Data repository. The GUI section contains all view files (html files), the functional core contains the files that contain the functionality for the application like the controller files and finally the data repository contains the databases. The diagram shows how the sections interact with each other, the arrows in the diagram evaluate this. The functional core uses the databases and the GUI uses the functional core.