IT314 - Software Engineering

Lab 6

Educational Networking Tool for College Students

Group - 29

Group Members:

- 202001150 - Devdeep Shetranjiwala
- 202001173 - Harsh Patel
- 202001270 - Naisarg Shah
- 202001412 - Zeel Bhanderi
- 202001424 - Kaushal Patel
- 202001431 - Shrey Shah
- 202001441 - Priyanshi Parmar
- 202001449 - Achyut Shah
- 202001452 - Krish Patel
- 202001461 - Harsh Agheda
- 202001462 - Om Patel

Domain Analysis Model

A Domain Analysis Model will include the following components:

1. Domain Concepts: The key concepts within an educational networking tool might include users, resources (posts), and discussions. Each of these concepts would have attributes and relationships that define their characteristics and interactions.

2. Domain functions: The primary functions of an educational networking tool would include creating and managing user accounts, creating and managing posts, participating in discussions, reviewing posts, following users, and searching resources. Each of these functions would have rules and constraints that govern their behavior.

3. Domain Data: The data used within an educational networking tool would include user information, resources (posts) information, and users' connections. Each type of data would have specific rules and relationships that apply to it.

4. Domain Context: The broader context of an educational networking tool would include external factors such as privacy regulations, user friendliness, etc.

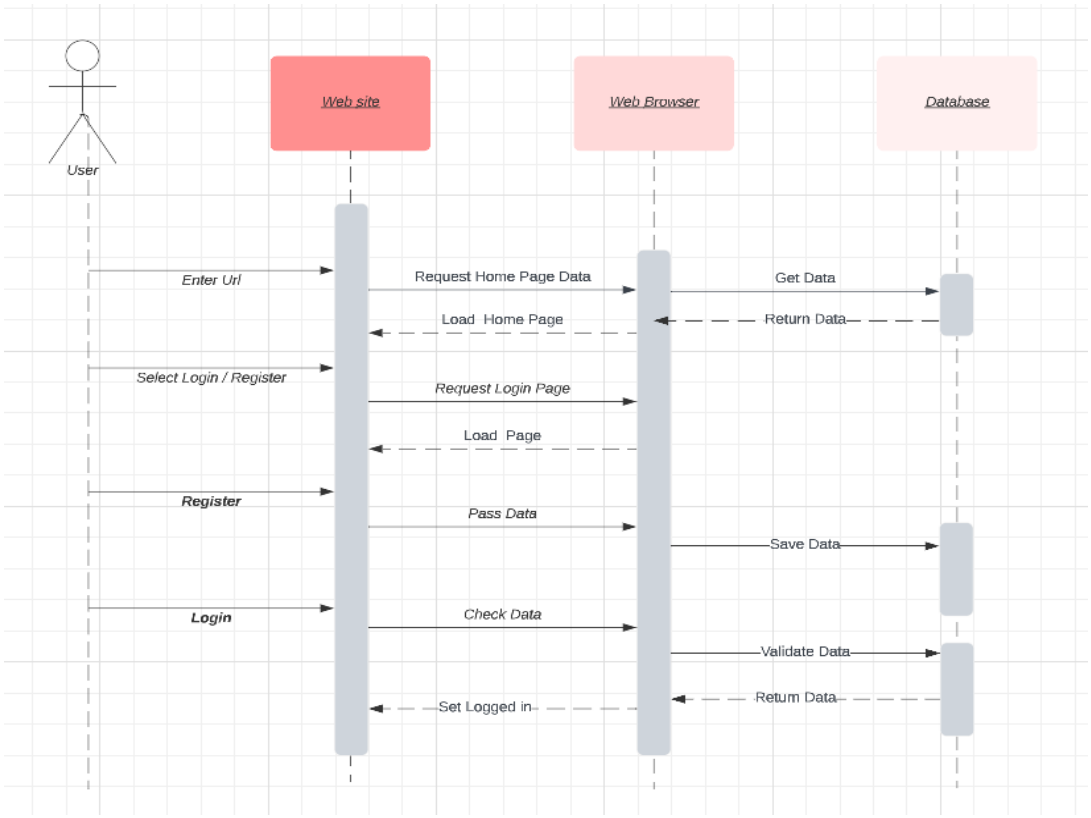
Identifying Boundary, Entity & Control Object

1. Boundary Objects: Should be the interface of the system through which the user interacts with the system. In our case, it will be a login/sign up form which will redirect the user to the home page on logging in so that the user can perform various functionalities.

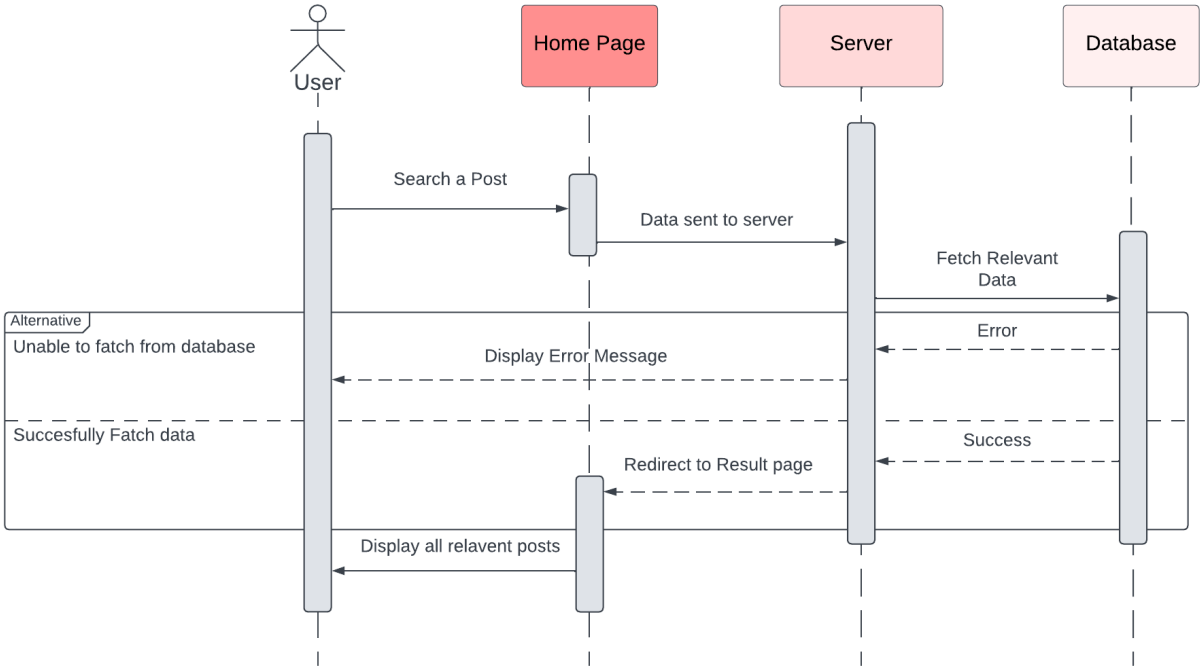
2. Entities: They are responsible for managing the persistent data and performing operations on that data. These include user database and posts database.

3. Control Objects: Control objects represent the system's control logic, which governs the flow of information and interactions between the boundary and entity objects. It includes a set of rules governing how users log in to the system, how the required data will be displayed to the user, how users can post new content, how users can interact with his connections, etc

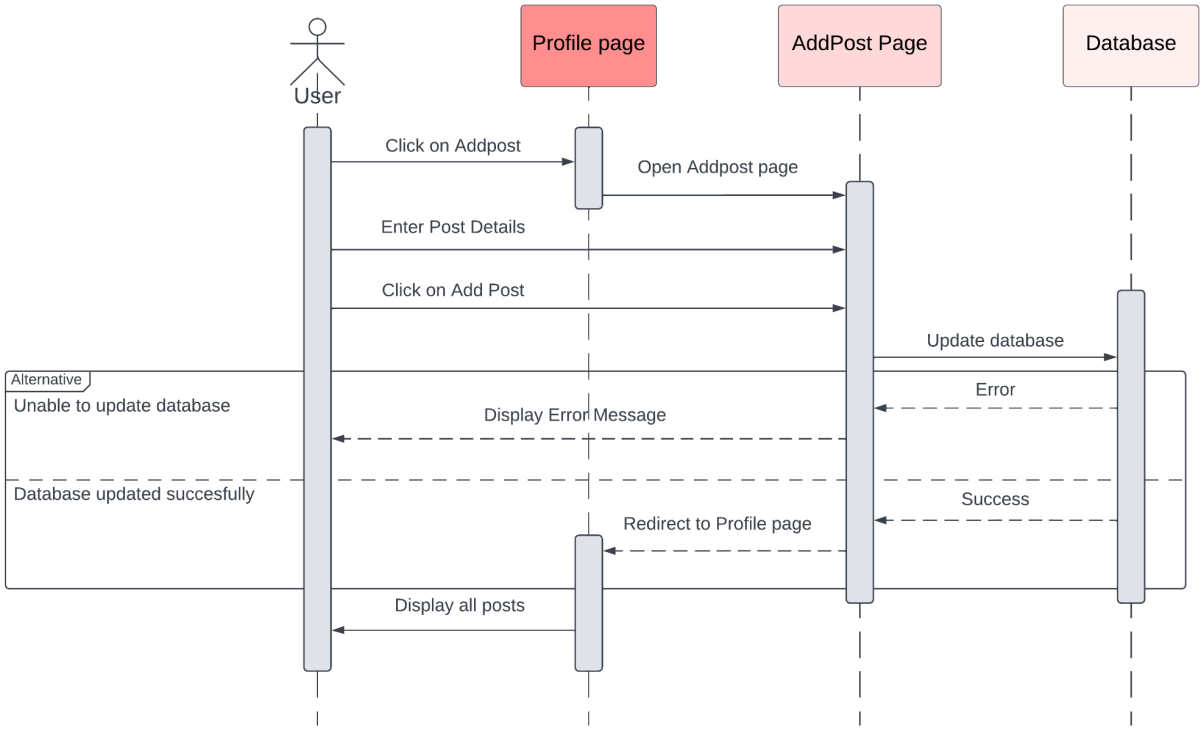
Sequence Diagram - Register & Login



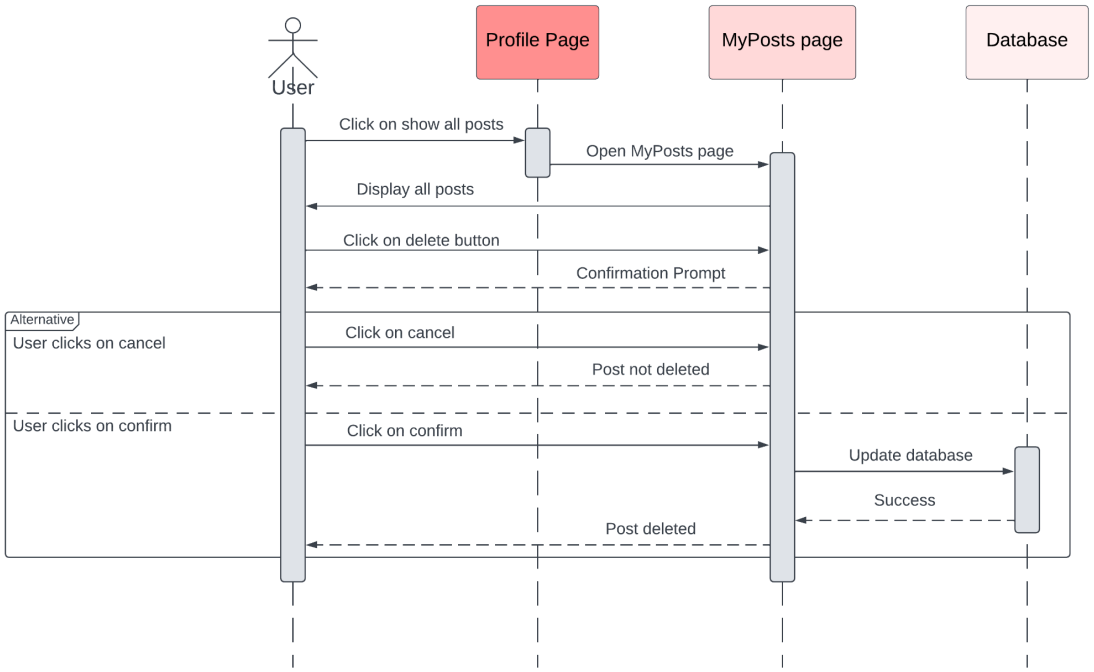
Sequence Diagram - Search



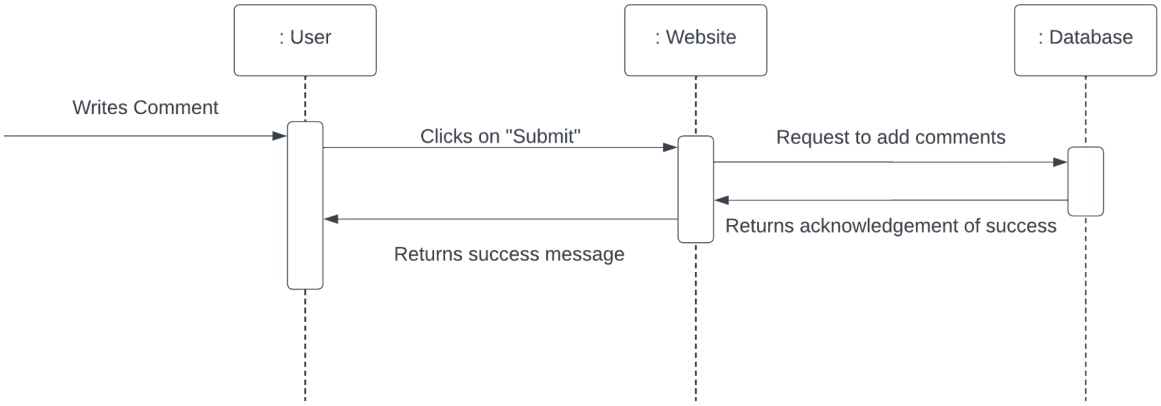
Sequence Diagram - Add Post



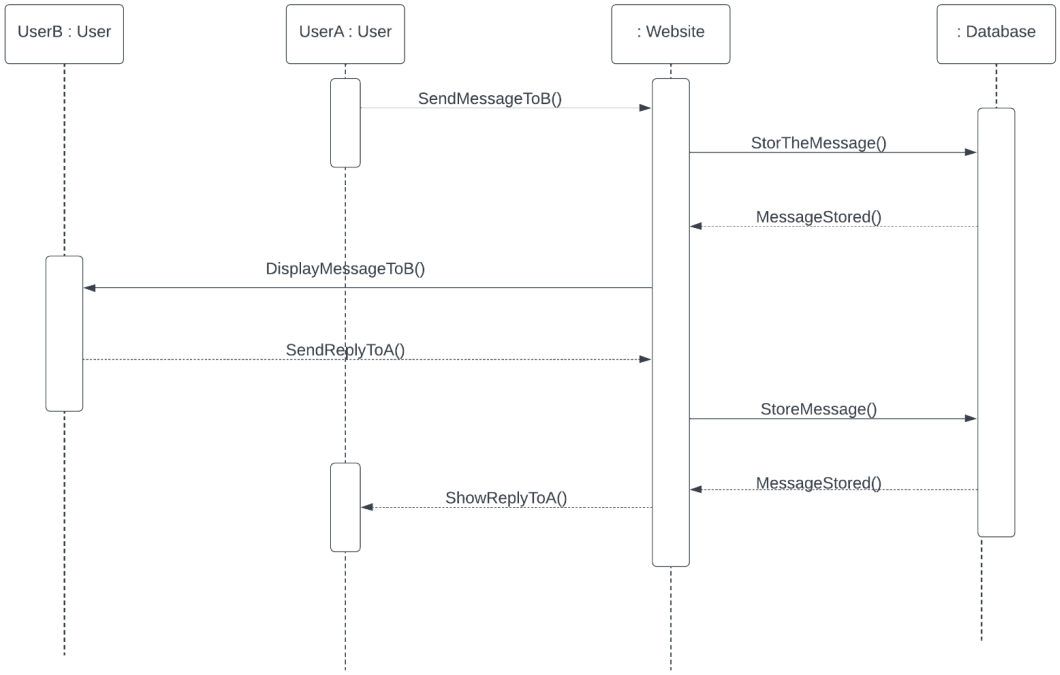
Sequence Diagram - Delete Post



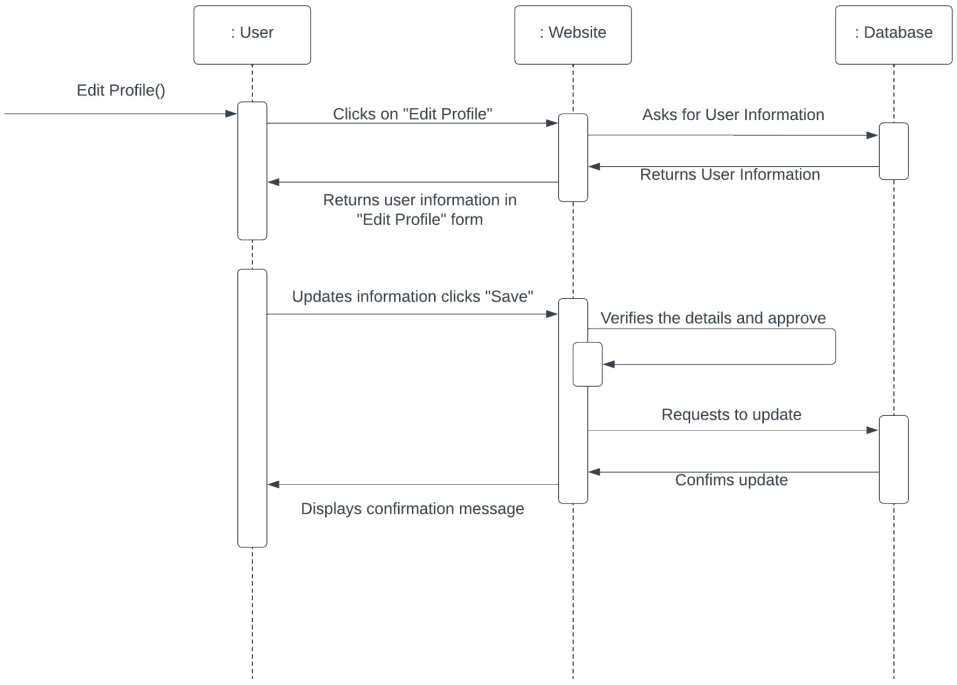
Sequence Diagram - Comment on Post



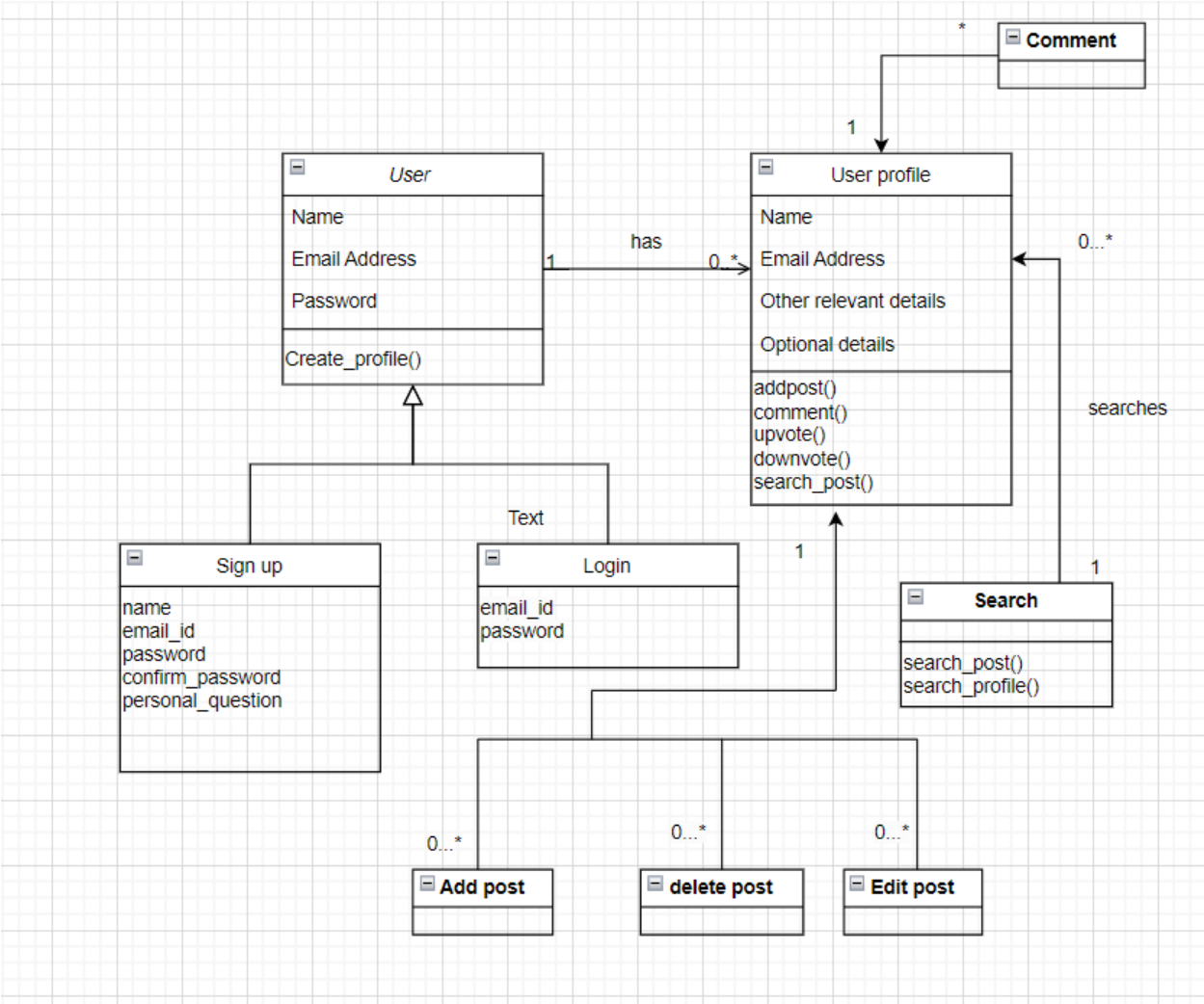
Sequence Diagram - Chat with User



Sequence Diagram - Update Profile



Class Diagram



Identifying Design Goals

Functionality: The networking tool must have a distinct set of features and capabilities that cater to the requirements of the users. Users should be able to make profiles, do searches, join groups and communities, exchange content, and interact with other users.

Usability: The tool should have a simple and intuitive UI that is easy to use. It should be created with the target audience's demands in mind, with a focus on usability and user experience.

Scalability: When more users sign up and utilize the platform, the tool should be built to manage massive amounts of data and traffic. It must be scalable both horizontally and vertically to accommodate future development.

Security: The tool should be developed with security in mind, with measures in place to guard user information, avoid unauthorized access, and lower the possibility of cyberattacks.

Compatibility: The tool should work with a variety of hardware and operating systems, browsers, and platforms, including PC and mobile devices.

Maintenance: The tool should have a robust infrastructure and well-documented codebase, and it should be simple to manage and update. Additionally, it should be versatile and adaptable in order to add new features and functions as required.

High Level System Design

Architecture

Our project is typically a web application that uses a client-server architecture. In this architecture, the client (web browser) sends requests to the server, which processes those requests and sends responses back to the client.

When a user accesses the application, the client sends a request for the web page to the server. The server processes the request, retrieves the necessary data from a database or other sources, and generates an HTML response that is sent back to the client. The client then renders the HTML and presents it to the user.

When a user interacts with our application, such as submitting a post or comment, the client sends a request to the server to perform the action. The server then processes the request, updates the appropriate data, and sends a response back to the client.

Thus, our application follows the client-server architecture where the client communicates with the server over HTTP or HTTPS protocols, and the server responds to the requests and manages the data.

In a client-server architecture, there are three tiers:

Presentation tier (also known as the client tier): This tier includes the user interface (UI) components that the user interacts with. In our educational networking app, the presentation tier would include the web browser used to access the application.

Application tier (also known as the middle tier or logic tier): This tier includes the logic and business rules that govern the application's behavior. In educational networking apps, the application tier would include the server-side code that handles user requests, communicates with the database, and implements the application's features.

Data tier (also known as the storage tier): This tier includes the data storage and management components of the application. In educational networking app, the data tier would include the database where user data, posts, comments, and other information are stored.

Subsystems

Authentication subsystem: User authorisation and authentication are handled by this subsystem. It makes sure that only users with the proper permissions can use the application's functionalities by checking user credentials and permissions.

Post and comment subsystem: Users' posts and comments are created, edited, and deleted through this subsystem. It manages the database's storage and retrieval of this data.

Voting subsystem: This subsystem manages user voting on posts, including upvoting and downvoting.

Search subsystem: This subsystem handles user searches for specific posts or users.

Connections subsystem: This subsystem allows users to see whom they are connected to and manage their following and followers list.