API-BASED SIGNUP/LOGIN SYSTEM

An API-based login/signup system is used by web applications to manage user authentication and authorization through Application Programming Interfaces (APIs). This system allows users to create accounts, log in securely, and access protected resources or perform actions within the application.

Signup/login systems are crucial components of modern web applications, providing a secure and efficient means for users to access and interact with online services.

Details about the System

1. HTTP Endpoint :

The code defines HTTP endpoints (**“/signup”, “/login”, “/home/<username>”**) to handle user registration, login, and home page rendering.

These endpoints can receive HTTP requests from clients and return appropriate responses.

1. HTTP Methods:

The endpoints support both GET and POST HTTP methods, allowing clients to interact with the server to perform signup, login, and access home page functionalities.

1. Data Transfer:

The system communicates with clients by transferring data via HTTP requests and responses.

During user registration and login, form data is sent to the server, and responses containing HTML templates are returned to render pages.

1. Authentication and Authorization:

The system implements authentication by verifying user credentials (email and password) against those stored in a database. If authentication is successful, the server redirects the user to the home page, demonstrating authorization to access protected resources.

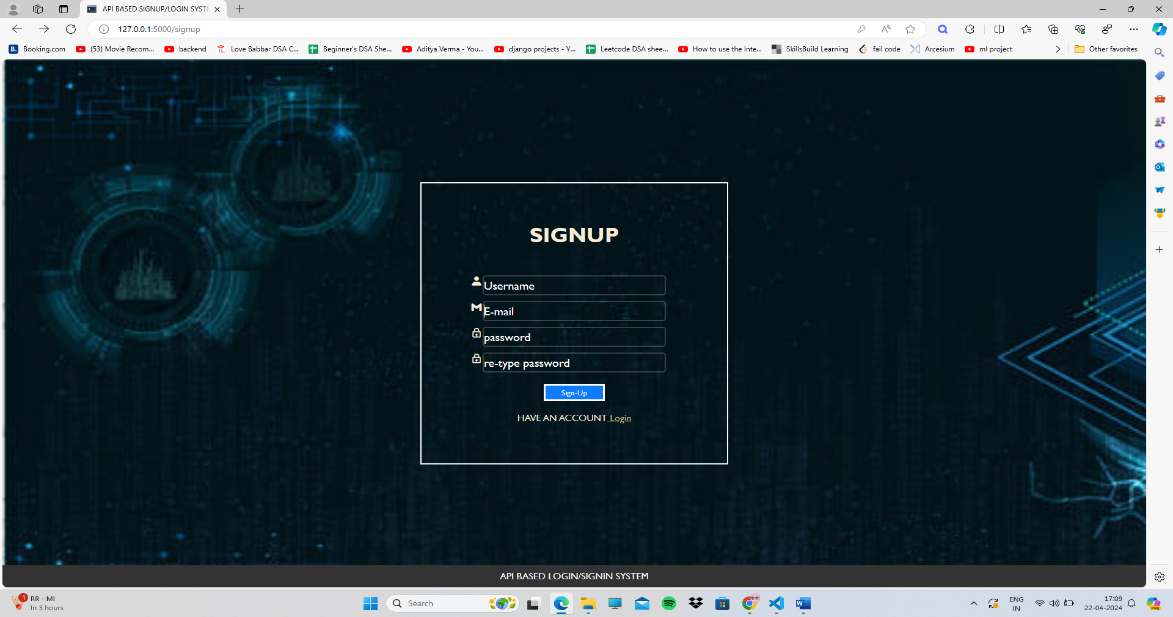
1. Data Storage and Retrieval:

User data, including usernames, email addresses, and hashed passwords, is stored and retrieved from the SQLite database “signupdata.db”. This database serves as the persistent storage for user information.

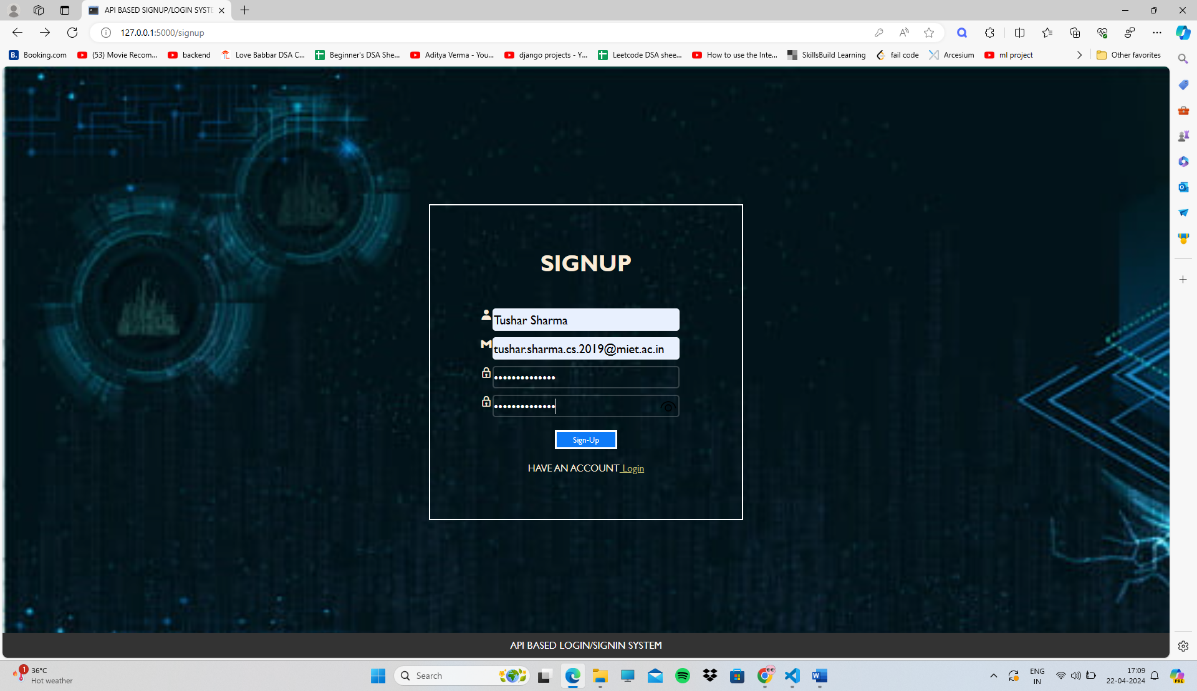
1. Error Handling:

The system includes error-handling mechanisms to deal with invalid requests, missing fields, password format violations, and existing username/email conflicts. It returns appropriate error responses (with HTTP status codes) to inform clients about encountered issues.

1. USER REGISTRATION :
   * + New users register using the “/signup” endpoint.



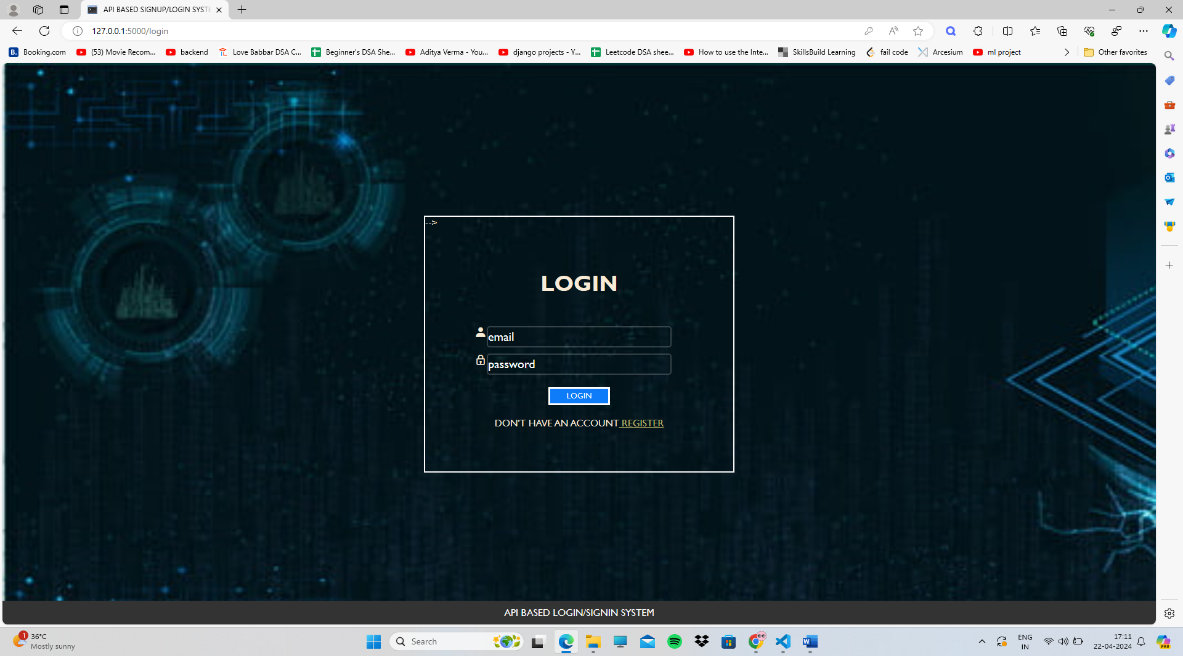
* + - Users provide credentials through the signup form and the method used here is “Post”.



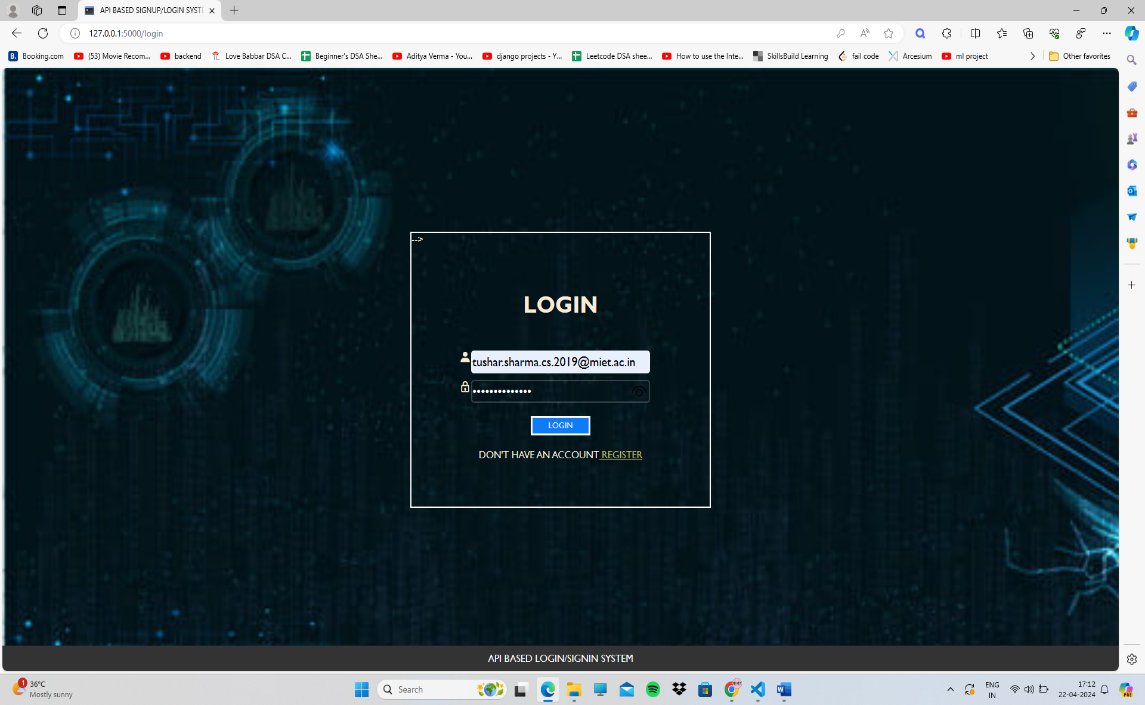
* + - User signup by providing the following details “Username”, “E-mail”, and “Password”.
    - User details are verified and stored in a database.

1. USER LOGIN :

* **User Login using the credentials i.e. “E-mail” and “Password” using the “/login” endpoint.**



* **The system verifies the user credentials and matches them with the details provided during the registration.**



**Importance of Signup/Login Systems**

**User Identification**: Signup/login systems serve as the primary means for identifying users within a web application. They allow users to create unique accounts, providing a digital identity within the system.

**Access Control**: These systems enable access control mechanisms, determining what resources and functionalities a user can interact with based on their authentication status and authorization level.

**Personalization**: By logging in, users can personalize their experience within the application, such as saving preferences, accessing personalized content, and maintaining a history of interactions.

**Security**: Signup/login systems play a crucial role in securing user data and protecting sensitive information. They often employ encryption techniques to safeguard user credentials during transmission and storage.

**Trust and Credibility**: A well-implemented signup/login system instills trust and credibility in users, assuring them that their information is protected and that they have control over their account.