COLLEGE CODE:[8223]

COLLEGE NAME:[Vandayar Engineering College]

DEPARTMENT:[Computer Science And Engineering]

STUDENT NM ID: 3EB40010185B9B4BFD14530102F3457A

ROLL NO:822323104013

DATE:[26.09.2025]

Completed the project named as

Phase-2

TECHNOLOGY PROJECT NAME:USER AUTHENDICATION SYSTEM

SUBMITTED BY, NAME:[KEERTHIKA.M]

]MOBILE NO:[7010613062]

1.TECH STACK SELECTION

A suitable technology stack is essential for building a secure and scalable User Authentication System. The selection of technologies is based on ease of development, security support, and future scalability.

Frontend

HTML5: For creating the structure of the user interface.

CSS3: For styling the pages and making them visually appealing.

JavaScript: For handling client-side validation and dynamic interactions.

Bootstrap: For responsive design so that the system works well on desktop and mobile.

Backend

Python: A simple yet powerful programming language, well-suited for handling authentication.

Flask (or Django):

Flask is lightweight and good for small projects.

Django is more feature-rich and provides built-in authentication and admin tools.

Database

SQLite: Suitable for small-scale, lightweight systems. Easy to integrate and does not require separate installation.MySQL: A scalable and reliable database for larger projects with many users.

Security

Password Hashing: SHA-256 or bcrypt hashing is used to ensure that plain-text passwords are never stored.

Session Management: Secure cookies or JWT (JSON Web Tokens) to maintain user login state.

Validation: Strong input validation to prevent SQL injection, XSS, and other vulnerabilities.

2. UI Structure / API Schema Design

User Interface Screens:

1. Login Page – Users enter username and password.

2. Registration Page – New users register with email, username, and password

3. Dashboard – Accessible only after successful login.4. Password Reset Page – Allows users to reset their password securely.

API Schema Design:

POST /register → Registers a new user into the system.

POST /login → Authenticates a user based on credentials.

POST /reset-password → Resets user password after verification.

GET /user/{id} → Fetches user profile information.

3. Data Handling Approach

All user credentials are stored securely in the database.

Passwords are hashed using SHA-256 or bcrypt.

Data validation ensures that empty fields or weak passwords are not accepted.

User sessions are maintained securely using tokens or session IDs.

Sensitive data such as passwords are never transmitted in plain text.

.

4.Modules in the System:

UI Module – User-facing interface for login, registration, and dashboard.

Authentication Module – Handles login, registration, logout, and password reset.

Database Module – Stores and manages user records.

Security Module – Implements hashing and password verification.

Session Module – Ensures that only authenticated users access the system.

5. Basic Flow Diagram

Step-by-Step Flow:

1. User registers → Details stored in database.

2. User tries to log in → Credentials verified against stored data.

3. If invalid → Error message displayed.

4. If valid → User redirected to dashboard.5. User logs out → Session terminated.