COLLEGE CODE : 9623

COLLEGE NAME : Amrita College Of Engineering And Technology

DEPARTMENT : Computer Science and Engineering

STUDENT NM-ID : 2A81889A479217DBC564038EFF6EDBDB

ROLL NO : 962323104110

DATE : 10-09-2025

Completed the project named as Phase_01_ TECHNOLOGY

PROJECT NAME : Login Authentication System

SUBMITTED BY,

NAME : Sweatha ML

MOBILE NO : 93457 68379

Phase 1 - Problem Understanding & Requirements

Problem Statement

In the digital era, most web and mobile applications require users to create accounts and authenticate their identity before accessing services. Traditional username–password-based login systems are vulnerable to security threats such as password theft, brute-force attacks, phishing, and unauthorized access. Many users also tend to reuse weak passwords across multiple platforms, further increasing the risk of data breaches.

There is a need for a secure, efficient, and user-friendly login authentication system that ensures confidentiality of user credentials, prevents unauthorized access, and provides a reliable mechanism for identity verification. Such a system should implement best practices in authentication—like password hashing, encryption, session management, and optional multi-factor authentication—while also maintaining simplicity and ease of use for end users.

Users and Stakeholders

Users

1. End Users – People who register, log in, and use the application securely.

2. Administrators – Manage user accounts, reset passwords, monitor login attempts, and ensure security policies are enforced.

3. Developers – Maintain and enhance the authentication system, fix bugs, and add new security features.

Stakeholders

1. Application Owners / Organization – They are responsible for providing the service, ensuring security, and protecting user data.

2. Security Teams – Ensure compliance with security standards, monitor threats, and handle breaches.

3. Regulatory Authorities – Ensure that the system complies with data protection laws (like GDPR, IT Act, etc.).

4. Business Partners / Clients – Depend on the authentication system for secure transactions and trust in the platform.

User Stories

1. As an end user, I want to create an account using my email and password so that I can access the application securely.

2. As an end user, I want my password to be stored securely (hashed & encrypted) so that my credentials are not exposed if there is a data breach.

3. As an end user, I want to be able to reset my password via email/OTP so that I can regain access if I forget it.

4. As an end user, I want to enable two-factor authentication (2FA) so that my account is protected even if my password is compromised.

5. As an administrator, I want to monitor failed login attempts so that I can detect and prevent brute-force attacks.

6. As an administrator, I want to manage (create, disable, or delete) user accounts so that only authorized users have access.

7. As a developer, I want to implement session management so that users remain logged in securely without re-entering credentials frequently.

8. As a security team member, I want to generate logs of authentication activities so that I can audit suspicious behavior.

MVP Features

1. User Registration
Users can create an account with username/email and password.

2. Secure Login
Users can log in with valid credentials.
Passwords are stored securely using hashing (e.g., bcrypt, SHA-256 + salt).

3. Logout Functionality
Users can securely log out and end their session.

4. Password Reset
Users can reset forgotten passwords via email/OTP link.

5. Session Management
System maintains active sessions securely (with session tokens/cookies).

6. Admin Account
Administrator can log in separately to manage users.

7. Basic Security Features
Input validation (prevent SQL Injection, XSS).
Account lockout after multiple failed login attempts.

Wireframes/API Endpoint List

Wireframes

1. Login Page

Fields: Email/Username, Password

Buttons: “Login”, “Forgot Password?”, “Register”

2. Registration Page

Fields: Name, Email, Password, Confirm Password

Button: “Register”

3. Dashboard (After Login)

Welcome message

Logout button

4. Forgot Password Page

Field: Email/Username

Button: “Send Reset Link/OTP”

API Endpoint List

Endpoint	Method	Description
/api/register	POST	Register a new user with name, email and password.
/api/login	POST	Authenticate user with email/username and password, return a JWT/session token.
/api/logout	POST	Logout the user and invalidate the session/token.
/api/forgot-password	POST	Send password reset link/OTP to user's email.
/api/reset-password	PUT	Reset the user's password after verification.

Acceptance Criteria

1. User Registration

Users must be able to create an account with a unique email/username.
Password must be validated (minimum 8 characters, strong complexity).
System must store passwords securely (hashed, not plain text).
Duplicate accounts with the same email must not be allowed.

2. User Login

Users can log in using valid credentials.
Invalid login attempts must show an error message without exposing sensitive details.
After 3–5 failed attempts, the account should be temporarily locked.
A session token or JWT must be issued upon successful login.

3. Logout

Users must be able to log out successfully.
Session token/JWT should be invalidated after logout.

4. Password Reset

Users can request a password reset using a registered email.
Reset link/OTP should expire after a limited time (e.g., 10–15 minutes).
After reset, old password should no longer be valid.