



**COLLEGE CODE : 9623**

**COLLEGE NAME : Amrita College Of Engineering And Technology**

**DEPARTMENT : Computer Science and Engineering**

**STUDENT NMID – 2A81889A479217DBC564038EFF6EDBDB**

**ROLL NO : 962323104110**

**DATE : 25.10.2025**

**Completed the project named as**

**Phase\_05\_Technology**

**PROJECT NAME : Login Authentication System**

**SUBMITTED BY,**

**NAME : SWEATHA M L**

**MOBILE NO : 9345768379**

## Project Overview:

- In today's digital age, online platforms require secure user authentication to protect sensitive information.
- Our Login Authentication System aims to provide a secure, efficient, and user-friendly way for users to register, log in, and manage sessions safely.

This system ensures:

- Passwords are hashed using bcrypt for data security.
- Authentication is handled using JWT (JSON Web Tokens).- Users can reset forgotten passwords through email.
- An admin panel allows administrators to monitor and manage users.-

Security features such as input validation, rate limiting, and account lockout protect against brute-force and injection attacks.

Tech Stack:

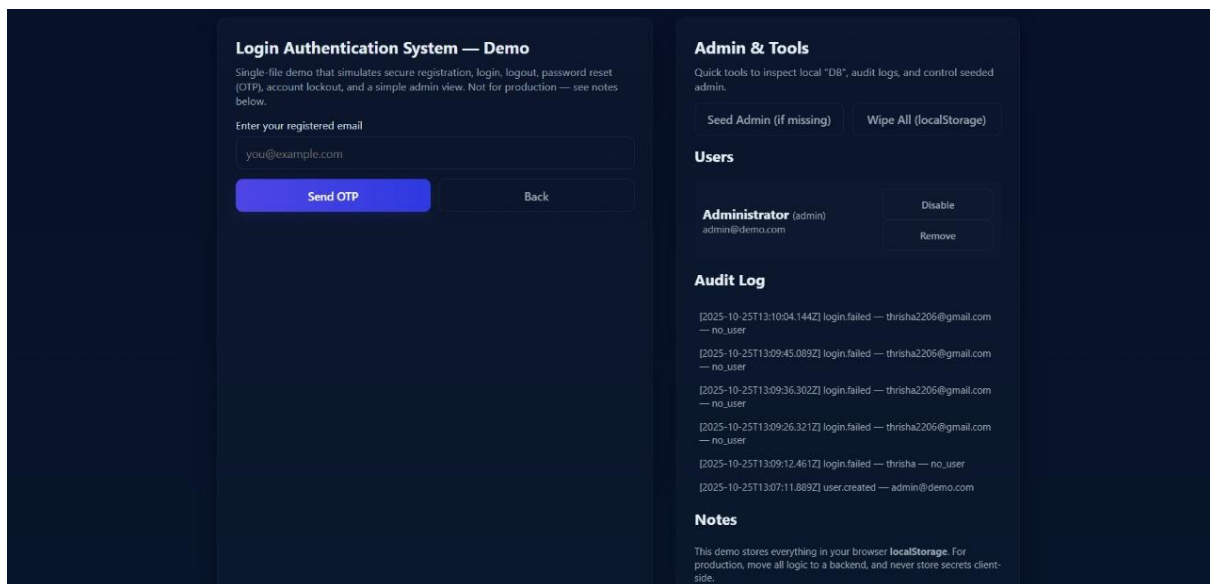
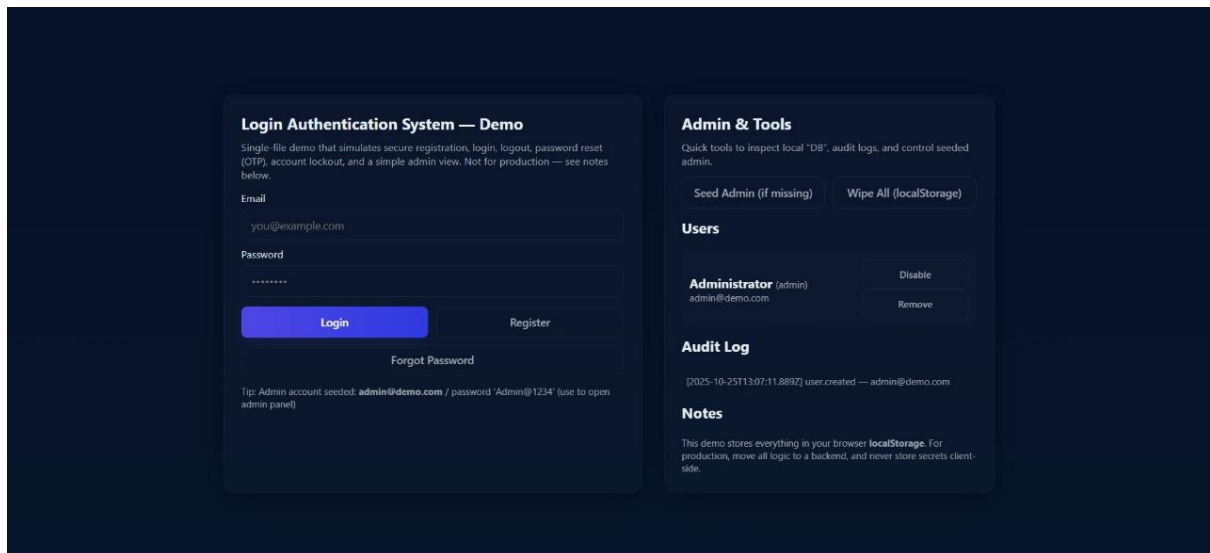
- Frontend: React.js (or HTML/CSS/JS prototype)
- Backend: Node.js with Express.js
- Database: MongoDB (Mongoose ORM)
- Styling: Tailwind CSS / Bootstrap
- Version Control & Hosting: Git + GitHub / GitHub Pages

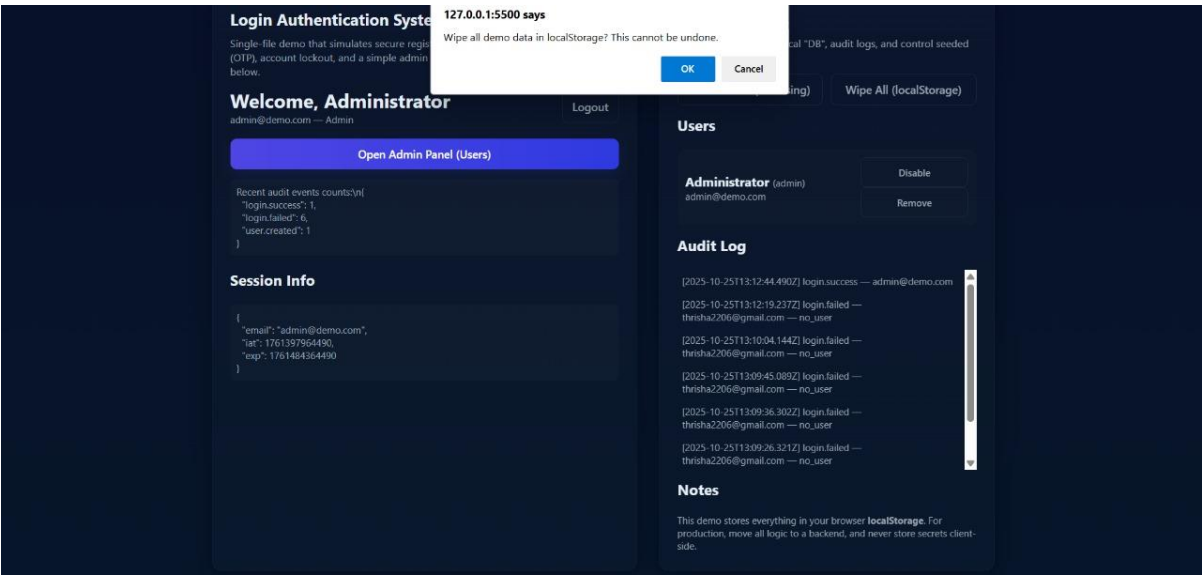
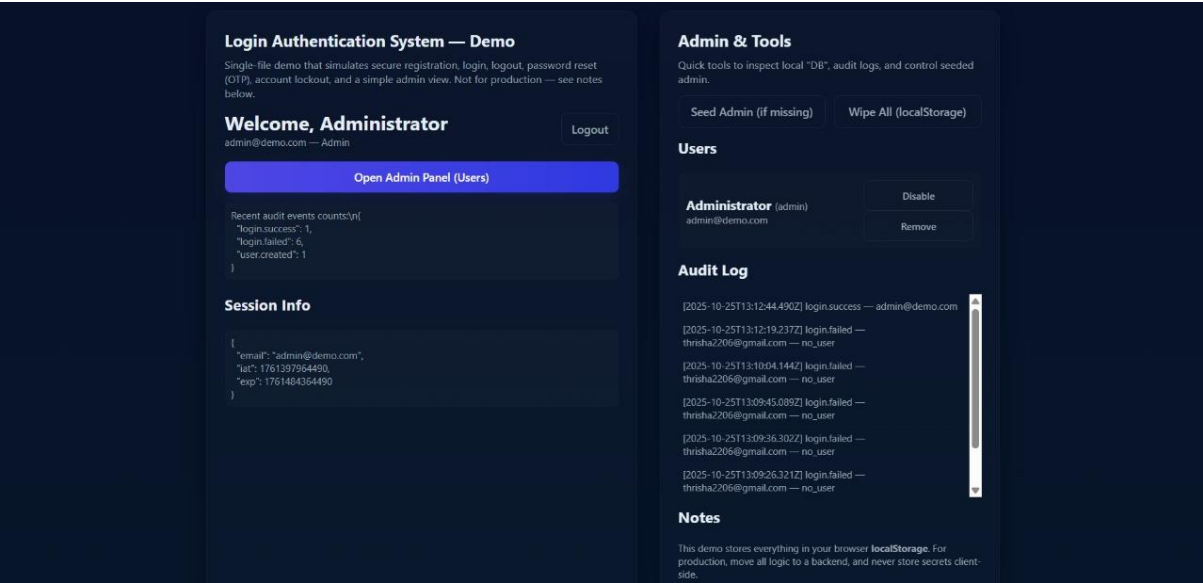
# Project Report :

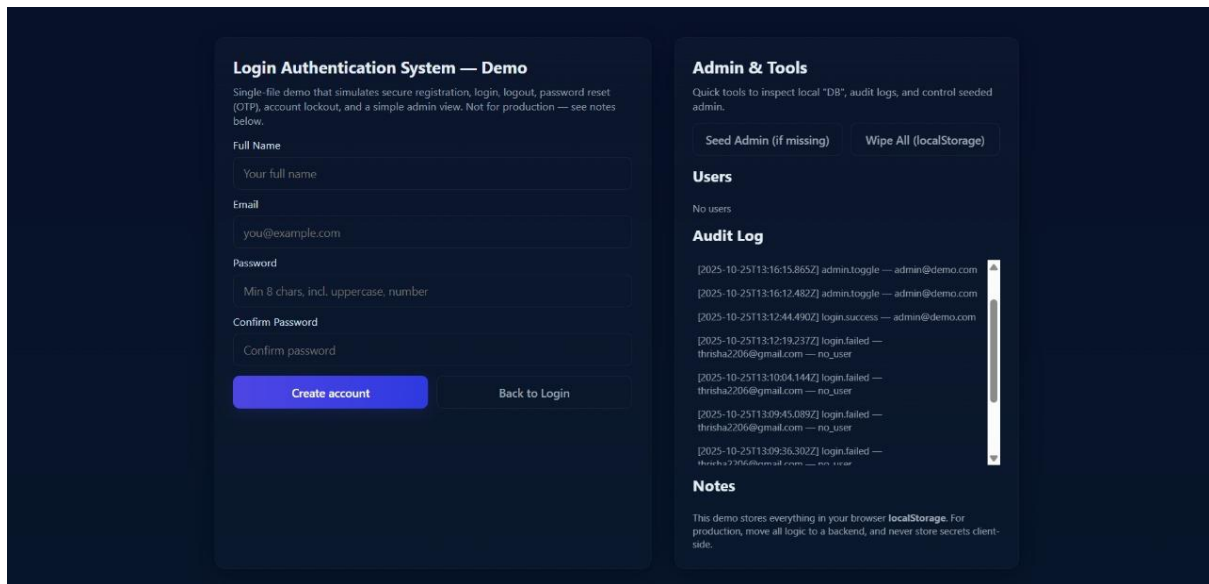
The project was developed in multiple phases:

- Phase 1 – Problem Understanding & Requirements We identified the problem of insecure password-based systems and defined user requirements.
- Phase 2 – Solution Design & Architecture Designed the architecture with RESTful APIs, frontend UI, and database schema using MongoDB.
- Phase 3 – Implementation Built the frontend and backend, integrated bcrypt, JWT, and implemented secure routes.
- Phase 4 – Enhancement & Deployment Added Forgot Password, 2FA, Role-Based Dashboard, and deployed the system online.
- Phase 5 – Testing & Final Verification Performed Unit, Integration, and Security Testing to ensure complete system stability

# Screenshots:







## Challenges & Solutions:

### Challenge:

- Difficulty setting up secure password storage
- Session expiration & token validation issues
- Handling repeated failed login attempts
- Managing frontend and backend communication

### Solution Applied:

- Implemented bcrypt hashing to safely store passwords
- Switched to JWT-based stateless authentication
- Added account lockout after 5 invalid attempts
- Used RESTful API architecture and CORS policy setup

- UI responsiveness across devices
- Applied Tailwind CSS/Bootstrap for consistent styling
- Email service for password reset
- Configured Nodemailer with fallback for testing
- Deployment configuration errors
- Resolved with environment variables and GitHub CI/CD

## **Github Link:**

GitHub Repo Link: **<https://github.com/mlsweatha/login-authentication-system>**

Deployed Link: <https://login-authentication-system-y37k.vercel.app/>

“All the project phases (Phase 1 to Phase 5) have been uploaded to the GitHub repository mentioned above.”