ANONYMOUS CONFESSION APP – FULL STACK APPLICATION

A PROJECT REPORT

Submitted by

Rohit Kumar (UID: 23BCS13184)

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE ENGINEERING

Chandigarh University

JULY – 2025

ABSTRACT

The Anonymous Confession App is a full-stack web application that enables users to share their thoughts, feelings, or confessions anonymously. The system ensures user privacy, seamless interaction, and scalability using modern web technologies — ReactJS for the frontend, Node.js with Express for the backend, and MongoDB for database management. The platform includes key modules for posting, reporting, and moderating confessions. A primary focus was placed on user anonymity, data security, and an intuitive interface. RESTful APIs and JWT authentication ensure secure, efficient communication between client and server.

CHAPTER 1: INTRODUCTION

The Anonymous Confession App is designed to provide individuals with a platform to express their emotions and secrets freely without fear of exposure or judgment. By integrating secure full-stack technologies, the app maintains a balance between anonymity, security, and simplicity.

1.1 Problem Statement

Many social media users hesitate to share personal thoughts due to privacy concerns or fear of social judgment. Existing platforms either compromise anonymity or lack moderation mechanisms. This application provides a secure and anonymous environment to express emotions safely, with an admin panel ensuring proper content management.

1.2 Objectives

1. Develop a secure and scalable full-stack web application for anonymous confession sharing.
2. Implement JWT authentication and role-based access for admins.
3. Create a responsive and user-friendly interface using ReactJS.
4. Enable real-time confession posting and moderation.
5. Maintain secure data storage and prevent data breaches.

1.3 Scope

The application provides a centralized platform for anonymous communication. It can be expanded to include mobile versions, AI-driven content moderation, and community-based confession spaces for institutions or organizations.

CHAPTER 2: LITERATURE REVIEW

2.1 Existing Systems

Several anonymous platforms, such as Whisper and Secret, enable anonymous messaging but suffer from poor data protection and moderation. They often lack accountability and privacy safeguards. The proposed system improves upon these limitations by integrating robust backend architecture, encrypted tokens, and a user-reporting feature to maintain platform integrity.

2.2 Technologies Used

Frontend: ReactJS, HTML5, CSS3, Bootstrap
Backend: Node.js, Express.js
Database: MongoDB
Tools: Postman, VS Code
Authentication: JWT (JSON Web Token)
Version Control: Git & GitHub

CHAPTER 3: SYSTEM DESIGN AND ARCHITECTURE

3.1 Architectural Overview

The system follows a three-tier architecture:

1. Presentation Layer: ReactJS frontend for UI/UX interactions.

2. Application Layer: Node.js with Express.js handles logic, authentication, and API routing.

3. Data Layer: MongoDB manages storage of confessions, reports, and users.

3.2 Database Design

The MongoDB database uses collections such as:

• Users – stores credentials and tokens.

• Confessions – holds user posts, timestamps, and likes.

• Reports – contains flagged posts for admin review.

Relationships between collections ensure smooth moderation and retrieval.

3.3 Data Flow

1. User posts a confession via the frontend.
2. Request is sent to backend using RESTful APIs.
3. Backend authenticates user (if required) and stores data in MongoDB.
4. Data is fetched back and displayed on the frontend dynamically.

CHAPTER 4: IMPLEMENTATION AND RESULTS

4.1 Backend Modules

1. **Authentication Module:** Handles login and admin authentication using JWT.
2. **Confession Module:** Enables CRUD operations for confessions.
3. **Report Module:** Allows users to flag inappropriate confessions.
4. **Admin Module:** Manages reports and deletes flagged content.

4.2 Frontend Modules

- **Home Page:** Displays a live feed of confessions.
- **Post Page:** Allows anonymous submissions.
- **Report Button:** Lets users report harmful posts.
- **Admin Dashboard:** Provides admin control over reports.

4.3 Testing

Testing was conducted using Postman for API validation, and manual testing for UI responsiveness. Backend endpoints were verified using Mocha and Chai libraries. The integrated system performed reliably across modules, ensuring consistent data flow between client and server.

CHAPTER 5: CONCLUSION AND FUTURE ENHANCEMENTS

The Anonymous Confession App demonstrates how full-stack development can produce a secure, private, and interactive platform for open communication. It successfully combines confidentiality with scalability and offers a framework that can be expanded for real-world deployment.

5.1 Future Enhancements

- AI-based content filtering and moderation.
- Integration of mobile application (Android/iOS).
- Dark mode and customizable UI themes.
- Real-time chat features for anonymous replies.
- Cloud deployment using AWS or Render for better scalability.

APPENDIX A: USER MANUAL

1. Start the backend server using **npm start**.
2. Launch the ReactJS frontend using **npm start**.
3. Register or log in as a user.
4. Post confessions anonymously.
5. Admin can review and manage reports from the dashboard.

APPENDIX B: ACHIEVEMENTS AND CERTIFICATIONS

• Successfully developed and deployed a full-stack web application.
• Implemented secure JWT-based authentication.
• Integrated RESTful APIs with a ReactJS frontend.
• Deployed backend using Node.js and MongoDB.
• Achieved hands-on expertise in full-stack development (MERN Stack).