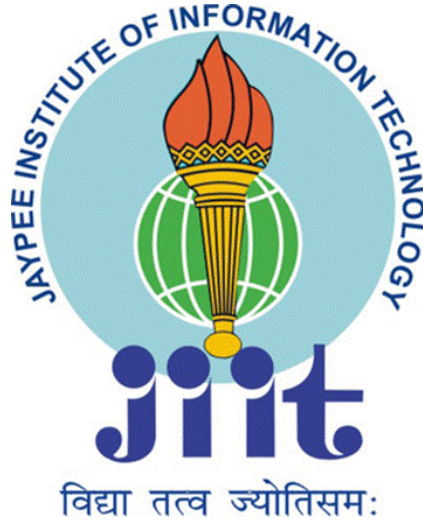


**JAYPEE INSTITUTE OF INFORMATION
TECHNOLOGY, NOIDA**



COMPUTER NETWORKS AND IOT LAB

B.TECH, EVEN SEMESTER 2025

TITLE OF PROJECT:

**Secure File Sharing Web Application using FTP,HTTP
(MERN)**

Submitted By:	Submitted To:
Phalak Bhatnagar 22103023 B1 Aayushmaan Mittal 22103029 B1 Vishal 22103028 B1	Dr. Meenal Jain Dr. Aastha Maheshwari

Acknowledgement

We would like to express our heartfelt gratitude to Dr. Meenal Jain and Dr. Aastha Maheshwari for their constant support, guidance, and encouragement throughout the duration of this project. Their valuable insights and constructive feedback were instrumental in helping us understand the subject matter more deeply and complete the work effectively.

We are thankful for the learning opportunities and mentorship provided by them, which have significantly contributed to our academic and professional development.

Problem Statement

In an increasingly interconnected digital world, the secure transmission of data is critical. Traditional file-sharing methods often lack encryption, authentication, or reliability, exposing sensitive data to interception, unauthorized access, or corruption. Moreover, many file transfer systems do not provide a user-friendly interface or compatibility with network protocols like FTP, limiting usability across diverse platforms. This project addresses these challenges by building a secure, web-based file sharing platform that ensures confidentiality, integrity, and accessibility, leveraging modern web technologies and network communication protocols.

Objective

The primary objective of this project is to design and develop a secure, efficient, and user-friendly file sharing system using modern web technologies and networking concepts. The system aims to enable seamless upload, storage, and retrieval of files over a network, supporting direct file transfers between users on the same network environment.

Through this implementation, we seek to demonstrate the integration of frontend and backend technologies (using the MERN stack) with essential networking protocols like FTP, thereby creating a full-stack application that not only supports high-level user interactions but also adheres to core principles of data communication and resource sharing. This project also explores how such systems can be scaled and adapted to IoT and edge computing use cases, setting the foundation for secure file exchanges in distributed environments.

Features

1. **User Authentication**

Secure login and signup functionality using JWT-based authentication.

Passwords securely hashed before storage.

2. **Encrypted File Upload and Download**

Files are encrypted using AES-256-CBC before being stored on the server.

Decryption occurs securely during download for authorized users.

3. **FTP Support**

File transfer through FTP protocol for traditional network clients.

4. **File Management Dashboard**

View, delete, and manage uploaded files in a clean, organized interface.

Technology Stack

Frontend

- **React.js**: Component-based UI for dynamic rendering.
- **Next.js**: React framework used for server-side rendering and routing.
- **Tailwind CSS + shadcn/ui**: For responsive and modern design elements.

Backend

- **Node.js**: JavaScript runtime for building the server-side logic.
- **Express.js**: Web application framework for creating REST APIs.

Database

- **MongoDB**: NoSQL database for storing user credentials and file metadata.

Security

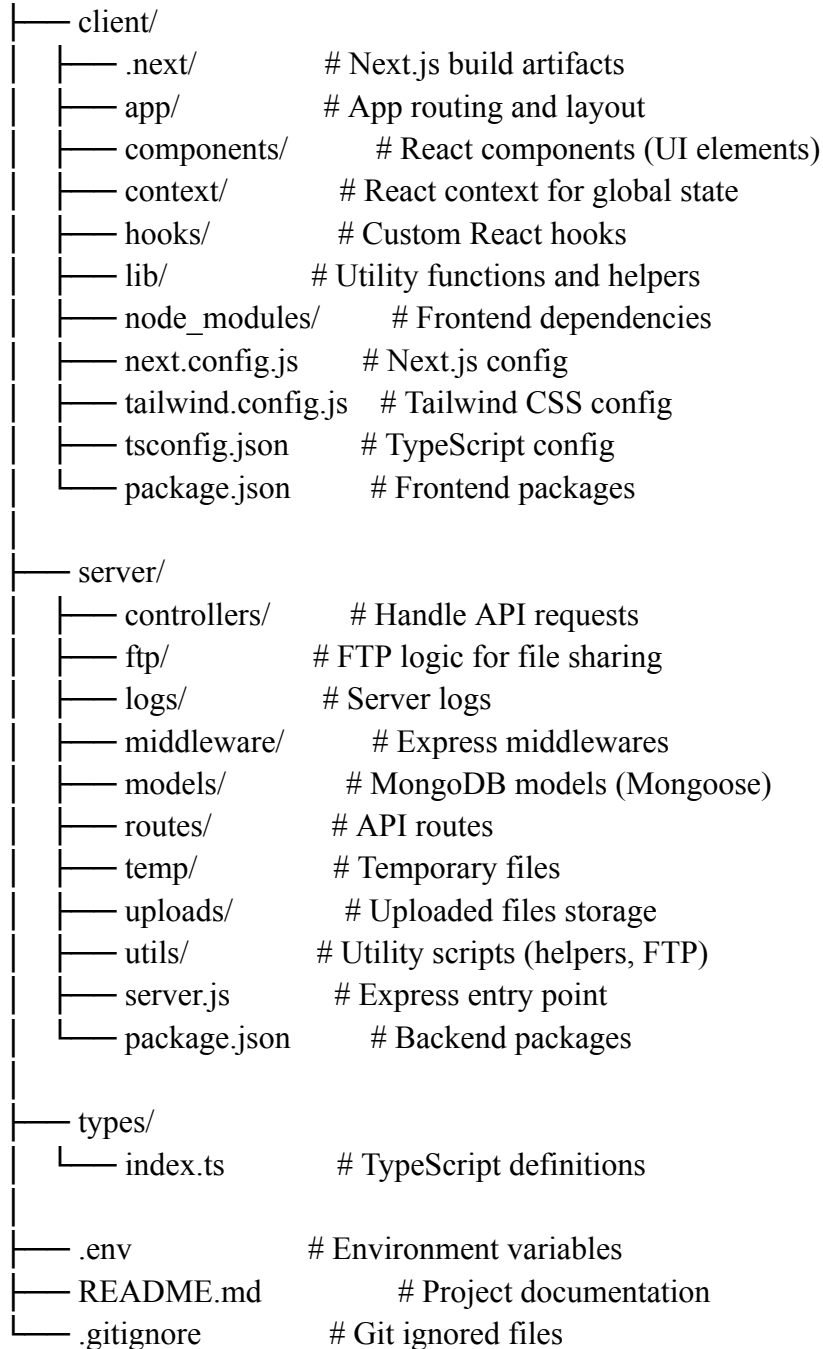
- **JWT (JSON Web Tokens)**: For secure and stateless user authentication.
- **AES-256-CBC**: Advanced encryption standard for file security.
- **Environment Variables**: Secure storage of secrets and keys.

Networking Protocols

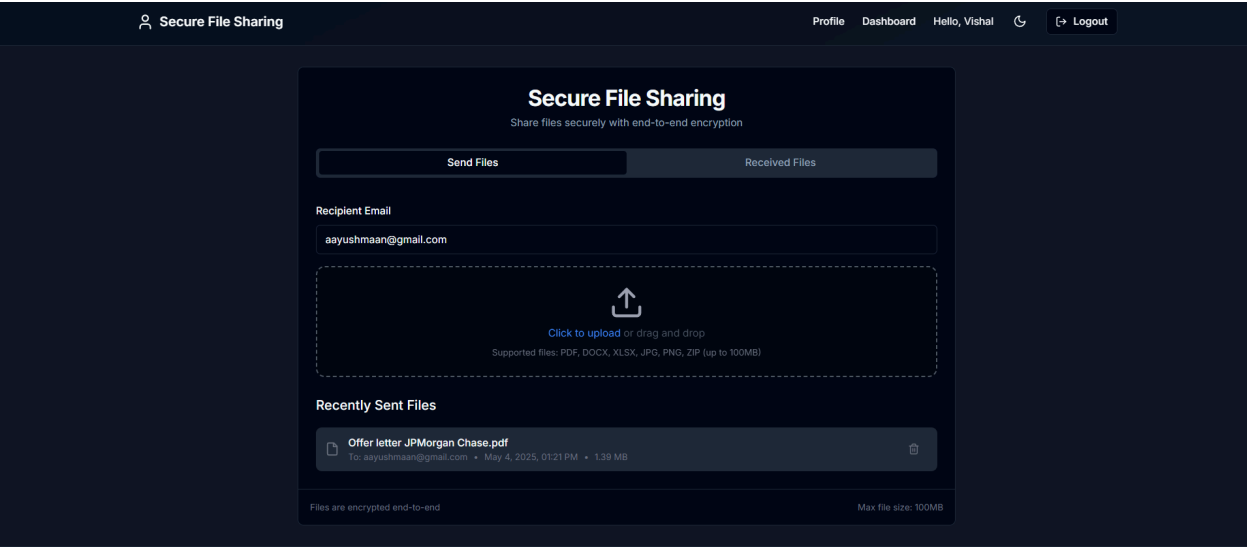
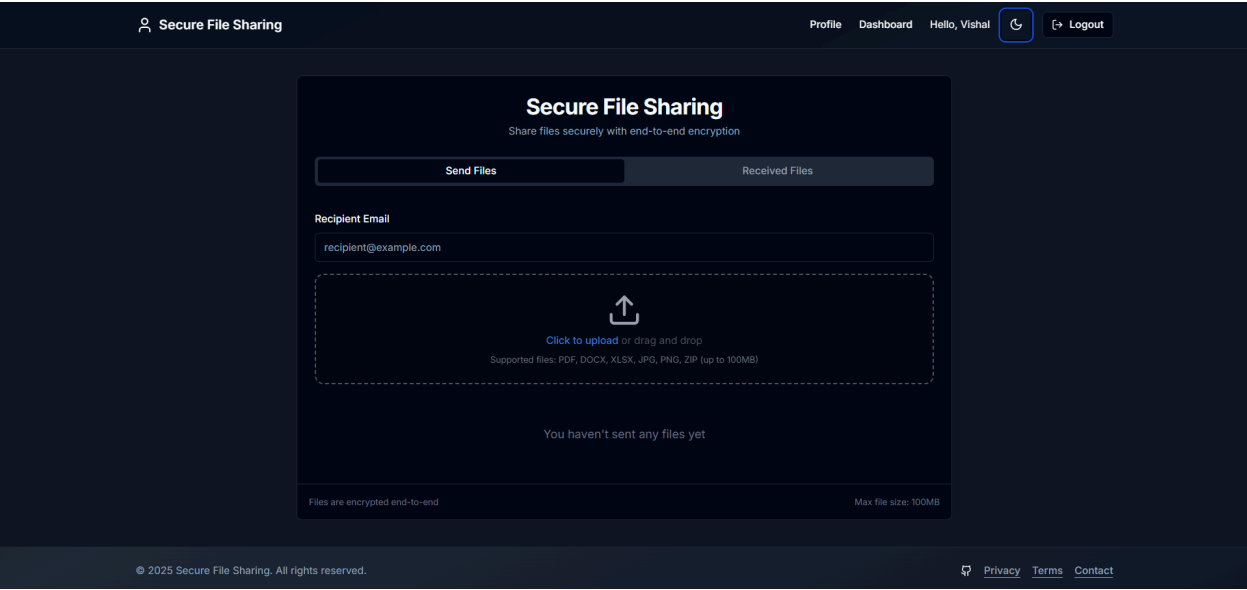
- **HTTP/HTTPS**: For secure client-server communication.
- **FTP (File Transfer Protocol)**: For direct file exchange between client and server.

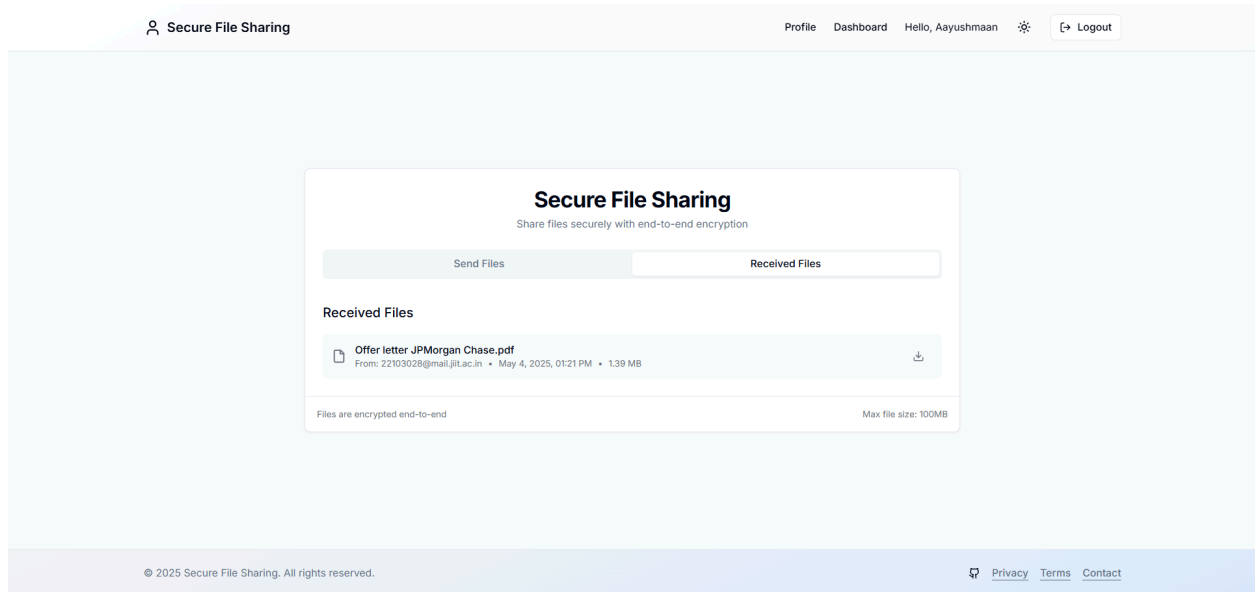
Project Structure

MERN-FILE-SHARE/



Project Images





Future Scope

1. Authentication and Role Management: Adding user authentication (OAuth or JWT) with role-based access control for secure file operations.
2. Real-time File Transfer Logs: Implementing WebSockets to provide real-time notifications and logs during upload/download.
3. IoT Device Integration: Allowing file uploads from IoT-enabled sensors or devices, especially for edge computing applications.
4. Cloud Storage Integration: Connecting the system with cloud storage providers like AWS S3 or Google Drive for persistent and scalable storage.
5. Virus Scanning and File Integrity: Integrating antivirus scanning APIs and checksum-based file integrity checks.

Conclusion

The MERN File Share project successfully demonstrates a practical application of computer networking by enabling secure and efficient file sharing across devices using the MERN (MongoDB, Express, React, Node.js) stack. With the integration of FTP protocol and a modern frontend interface, the system offers users a seamless way to upload, share, and download files over the network. The modular architecture and clean separation of client-server logic make it scalable and maintainable.