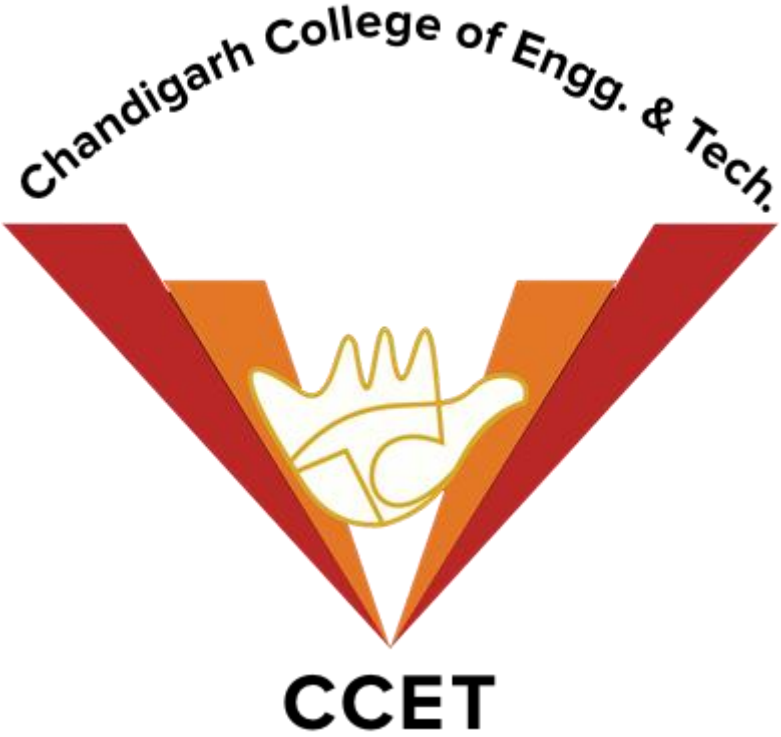
MAJOR PROJECT SYNOPSIS

SHYAM ESTATE



Submitted to:

Dr. Santosh Kumar Yadav Lecturer

CSE Department

Submitted by:

Eklavya(9508)

Aryan rathi(9505)

6th semester CSE

1. Introduction

1.1 Purpose

The *Shyam Estate* project aims to create a modern real estate platform that connects property buyers, sellers, and agents through an efficient and interactive web application. The system simplifies property search, listing management, and communication while delivering a seamless and scalable digital experience.

1.2 Scope

The system includes:

- **Customer Side**: Property browsing, filtering, and inquiry features.
- **Agent/Seller Side**: Property listing, management, and customer inquiry handling.
- **Admin Panel**: Monitoring users, properties, and platform statistics.

1.3 Technologies Used

- **Frontend**: React.js (JavaScript, JSX)
- **Backend**: Node.js with Express.js
- **Database**: MongoDB (Cloud Hosted / Local)
- **Other Technologies**: JWT (Authentication), Redux (State Management), Cloudinary (Image Uploads)
- **Server Environment**: Node.js server with RESTful API architecture

---

2. System Overview

2.1 Project Structure

The project is structured into:

- **Frontend Application** (`/client` folder)
  - React application with routing and component-based structure
  - Customer and Agent/Seller interface
- **Backend Server** (`/server` folder)
  - Node.js Express API for handling business logic
  - JWT Authentication for users and agents
- **Database**
  - MongoDB database storing users, properties, and inquiries

---

3. Functional Requirements

3.1 Customer Side

3.1.1 Registration and Profile Management

- Customers sign up/login via React forms.
- Secure authentication using JWT tokens.
- Manage saved properties and inquiries.

3.1.2 Property Search and Inquiry

- Browse properties with dynamic search filters (location, price, type).
- Send inquiries directly to property owners/agents.

3.2 Agent/Seller Side

3.2.1 Authentication

- Secure agent registration/login.
- Separate agent dashboard for property management.

3.2.2 Property Management

- Create, edit, and delete property listings.
- Upload property images (stored on Cloudinary).
- View inquiries from potential customers.

3.3 Admin Panel

3.3.1 Manage Listings and Users

- Approve, reject, or delete property listings.
- Manage users (customers, agents) – block/unblock accounts.

3.3.2 Platform Analytics

- View statistics on property uploads, user growth, and activity trends.

---

4. Database Design

4.1 Database Structure (MongoDB - shyamEstateDB)

Collections:

- **Users** (UserID, Name, Email, PasswordHash, Role)
- **Properties** (PropertyID, AgentID, Title, Description, Price, Location, Images, Status)
- **Inquiries** (InquiryID, CustomerID, PropertyID, Message, Timestamp)

4.2 Security Features

- Password hashing with bcrypt.
- Input validation and sanitation using Express middlewares.
- JWT-based session management and role-based access control.

---

5. System Implementation & Deployment

5.1 Setup Instructions

1. Clone the repository.
2. Install dependencies (`npm install`) separately in `/client` and `/server`.
3. Set up `.env` files with MongoDB URI, JWT Secret, and Cloudinary credentials.
4. Run the backend server (`npm run server`) and frontend (`npm start`) concurrently.

5.2 Example Accounts for Testing

- Customer: Email: testuser@shyamestate.com | Password: test123
- Agent: Email: agent@shyamestate.com | Password: agent123

---

6. User Interface & Responsiveness

- Built with responsive design principles.
- Fully mobile-friendly (using CSS Flexbox/Grid and React libraries like Material-UI/Bootstrap).

---

7. Error Handling & Logging

- API errors are handled via custom middleware in Express.
- Frontend displays user-friendly error messages.
- Logging critical server-side errors for monitoring and debugging.

---

8. Future Enhancements

- Real-time chat between buyers and sellers (using WebSockets).
- Payment gateway integration for premium listings (Stripe, Razorpay).
- AI/ML-powered property recommendations.
- Mobile App development using React Native.
- Multi-language support for global users.

---

9. Conclusion

The *Shyam Estate* project, built on the powerful MERN stack, offers a scalable, secure, and dynamic solution for real estate operations. It enhances user experience, simplifies real estate transactions, and supports business growth through an intelligent and interactive web platform.