IT644 Web Services and SOA

Project Proposal Document

Project Title

PropEase – Real Estate Management System

Project Description

PropEase is a comprehensive real estate web platform built on the MERN (MongoDB, Express.js, React.js, Node.js) stack that enables users to discover, list, rent, and purchase properties seamlessly. The platform offers a modern, scalable, and secure architecture ensuring smooth user experience across all devices.

Key features include:

- **Smart Property Search** – Advanced filters for location, budget, property type, amenities, and furnishing status.
- **Map-Based Browsing** – Integrated with Google Maps API to display property locations with interactive markers.
- **Property Management Dashboard** – Separate dashboards for owners, agents, and admins to manage listings, leads, and approvals.
- **User Authentication & Authorization** – Implemented using **JWT-based authentication** with role-based access control (RBAC).
- **Real-Time Chat System** – WebSocket-based messaging for instant communication between buyers, sellers, and agents.
- **Appointment Scheduling** – Calendar integration to book property visits.
- **EMI Calculator** – Built-in financial planning tool for estimating home loan installments.
- **360° Virtual Property Tours** – Supports immersive viewing experience using WebGL/Three.js.

- ❖ **Image & Document Upload** – Owners can upload property images and legal documents using Cloudinary/AWS S3 integration.
- ❖ **Notifications System** – Email and push notifications for updates on property visits, offers, and approvals.
- ❖ **Admin Panel** – Manage users, verify owners, approve property listings, and monitor platform activity.
- ❖ **Secure Payment Gateway** – Integration with **Stripe/Razorpay** for advance payments, booking fees, or premium listings.
- ❖ **Responsive Design & SEO Optimization** – Fully responsive UI for mobile and desktop with SEO-friendly meta tags and schema for property listings.

Motivation

The real estate industry faces major challenges such as lack of transparency, fraudulent property listings, and inefficient property discovery processes. Buyers struggle to find genuine properties that match their budget and preferences, while sellers face difficulty in reaching the right audience without costly intermediaries. Additionally, traditional property platforms lack AI-driven insights, virtual tours, and secure verification systems, making the process time-consuming and risky.

With increasing digital adoption and remote buying trends, there is a growing demand for an all-in-one online platform that provides trust, convenience, and intelligent decision-making tools. A modern, scalable solution built on the MERN stack can deliver a responsive, secure, and feature-rich experience that benefits buyers, sellers, and agents alike.

Problem Statement

The current real estate platforms suffer from the following issues:

1. **Trust Deficit –** Lack of proper owner verification leads to fake listings and fraud.
2. **Limited Search Experience –** Basic search features without smart filters or map-based browsing make property discovery inefficient.
3. **Poor Remote Experience –** Absence of 360° virtual tours and online scheduling makes remote property exploration difficult.

4. **Communication Gaps –** No integrated real-time chat for buyers, sellers, and agents.
5. **Financial Planning Limitations –** Lack of built-in EMI calculators and financial insights forces users to rely on external tools.
6. **Scalability & Security Issues –** Existing systems are not optimized for high concurrency, secure authentication, and role-based access.

Therefore, there is a need for a secure, scalable, and intelligent real estate management platform that:

- Ensures trust through owner verification.
- Offers AI-powered price estimation for transparency.
- Provides virtual tours and scheduling tools for convenience.
- Delivers real-time communication and smart property search.
- Supports financial planning within the platform.
- Is fully responsive and built on a modern web stack (MERN) for scalability and performance.

Target Users

- **Buyers/Renters –** Search and book properties.
- **Owner –** List and manage properties.
- **Admins –** Verify owners, approve listings, manage users.

Use Cases

- Search properties with filters & map view.
- List properties with images and details.

- Owner verification.
- 360° virtual tours.
- Schedule property visits.
- Real-time chat with owners/agents.
- EMI calculation and financial planning.
- Admin approval and user management.

Scope

PropEase is a **web-based real estate platform** built on the **MERN stack**, enabling property **search, listing, buying, and renting** with advanced features like **owner verification** and **virtual tours**. It serves **buyers, renters, owners, agents, and admins** by offering **secure transactions, real-time communication, and smart decision-making tools**.

High-Level Features

- Smart Search & Filters (location, budget, type)
- Map-Based Property Browsing
- 360° Virtual Property Tours
- Real-Time Chat Between Users
- Appointment Scheduling for Property Visits
- EMI Calculator & Financial Planning
- Admin Panel for Approvals & Management
- Secure Authentication & Role-Based Access
- Notifications & Alerts (Email/SMS/Push)
- Responsive & Scalable Architecture

Technology Stack

- **Frontend:** React.js, Redux/Context API, Tailwind CSS
- **Backend:** Node.js, Express.js
- **Database:** MongoDB (Mongoose)
- **Auth & Security:** JWT, bcrypt.js
- **File Storage:** Multer + Cloudinary / AWS S3
- **APIs:** Google Maps API, Socket.io (chat), Stripe/Razorpay (payments)

Roles and responsibilities

❖ Team Lead – Chandresh Thakkar

Oversees the project, manages the team, ensures timely delivery, and handles critical decisions.

❖ Frontend Developers – Kunal Agarwal, Vitrag Shah

Build responsive UI using React.js, integrate APIs, and implement features like property search, maps, and chat.

❖ Backend Developers – Anuj Patel, Abhi Andani, Ankush Bhattacherjea

Develop REST APIs with Node.js & Express, manage database operations in MongoDB, handle authentication, and integrate third-party services.

❖ UI/UX Designer – Amit Patel

Designs user-friendly layouts, ensures good user experience, and creates wireframes for features like search, listing, and virtual tours.

❖ DevOps Engineer – Kirtan Pithadiya

Handles deployment (Vercel, Heroku), CI/CD pipelines, monitors performance, and ensures system scalability and security.

❖ Technical Writer – Harsh Wadhwani

Prepares project documentation, user manuals, API documentation, and creates technical content for reports and presentations.

Timeline with milestones

Sprint	Timeframe	Description of Work
Sprint 1	**Aug 20 – Aug 26**	**Project Proposal: Define scope, features, tech stack, roles, and timeline. Deliverable: Proposal Document.**
Sprint 2	**Aug 27 – Sep 07**	**Requirements Gathering: Functional & non-functional requirements, user stories, UI wireframes (draft). Deliverable: Requirements Specification.**
Sprint 3	**Sep 08 – Sep 14**	**System Design: Architecture, ERD, component diagrams, API contracts, final wireframes. Deliverables: Design Document + Gantt Chart.**
Sprint 4	**Sep 15 – Sep 28**	**Initial Development Setup: MERN stack setup, Git repo, CI/CD config. Start core modules (auth, user roles). Weekly progress reports begin.**
Sprint 5	**Sep 29 – Oct 14**	**Core Features Development: Property listing module, search filters, Google Maps API integration (basic). Mid-Semester Demo prep. Deliverable: Partial working demo.**

Sprint 6	**Oct 15 – Oct 28**	**Advanced Features: Chat system, scheduling visits, EMI calculator. Continue weekly reports.**
Sprint 7	**Oct 29 – Nov 04**	**Testing Prep: Unit tests, integration tests, DB population with sample data, bug log creation. Deliverable: Test Plan + Test Cases.**
Sprint 8	**Nov 05 – Nov 15**	**Final Development & Deployment: AI price estimator, 360° property tours, security configs, hosting (Heroku/Vercel). Deliverables: Final App + Deployment URL.**
Sprint 9	**Nov 16 – Nov 18**	**Documentation & Presentation: Final report (15–25 pages), project slides, poster, walkthrough video. Deliverables: Final Report + Presentation.**
Sprint 10	**Nov 19, 2025**	**Peer Review & Contribution Report: Submit self/peer assessments. Final closing sprint.**