

SYNOPSIS ***MINOR PROJECT***

Room Bridge

A Room Finder with Chat & AI Based Ranking

(ES-451: Minor Project - Dissertation)

*Submitted in partial fulfillment of the requirements
for the award of the degree of*

Bachelor of Technology **Computer Science & Engineering**



Supervisors:
Ms. Apurva Jain
(Assistant Professor)

Submitted by:
Akshat Jain (00115607223)
Raghav Dwivedi (01015602722)
Pankaj Singh (04315602722)
Khushal (00715607223)

Department of Computer Science and Engineering
Dr. Akhilesh Das Gupta Institute of Professional Studies
(Formerly ADGITM)

FC-26, SHASTRI PARK, NEW DELHI

Affiliated to



GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY

Sector - 16C Dwarka, Delhi - 110075, India

2022-26

Table of Contents

Particulars	Page No
Title of the Project	
1. Introduction including Problem Statement	01
2. Objectives and Scope of the Project	02
3. Tools / Platform/Methodology	03
4. Hardware and Software Requirement specifications	05
5. Testing Technologies used	07
6 . Significance of the Project	09

Introduction including Problem Statement

Finding a suitable roommate or rental space has become a major challenge in today's urban and semi-urban environments. Students and professionals who migrate to cities often struggle with limited options, lack of verified listings, and the absence of a safe and reliable platform. Current solutions mainly focus on providing availability of rooms but fail to address the deeper issues of verification, compatibility, and secure communication.

The absence of a verification process exposes tenants to risks of fraud, while landlords face difficulties in finding genuine tenants who can be trusted with their property. Moreover, compatibility between roommates is often overlooked, which leads to conflicts and dissatisfaction in shared living arrangements. The lack of AI-driven personalization means that users are left to manually filter through countless listings without intelligent recommendations.

This project, Room Bridge, seeks to address these challenges by creating a secure, AI-enhanced housing and roommate platform. By combining verification mechanisms, AI-based ranking systems, and consent-driven communication features, the system aims to provide a trusted environment that benefits both tenants and landlords.

In addition, Room Bridge will integrate behavioral and lifestyle profiling to match users with compatible roommates based on interests, habits, and preferences. A transparent review and rating system will promote accountability and trust within the community. Real-time chat support, dispute resolution mechanisms, and in-app guidance will further streamline the user experience. The platform also intends to partner with educational institutions and corporate hubs to ensure a consistent flow of reliable users. Ultimately, Room Bridge envisions transforming the fragmented rental market into a structured, efficient, and user-friendly ecosystem.

Objectives and Scope of the Project

The primary objective of **Room Bridge** is to build a secure and **verified roommate** and rental platform that ensures trust and personalization for its users. The system will integrate **AI-based ranking mechanisms** that evaluate properties and roommate suggestions by combining star ratings with sentiment analysis of user reviews. A consent-based chat system will enable safe communication between tenants and landlords, while a review system will provide transparency and accountability.

The scope of the project includes students and working professionals in urban and semi-urban areas who require affordable and trustworthy housing solutions. Landlords will also benefit from a platform that verifies users at an admin level and reduces the risks of fraud. By leveraging AI-driven personalization, users will receive ranked suggestions for housing and roommates, ensuring better matches and higher satisfaction. In the long run, Room Bridge has the potential to evolve into a nationwide trusted ecosystem for rentals and shared living.

To further enhance its utility, the platform will incorporate location-based filters, budget preferences, and lifestyle compatibility scores, allowing users to narrow down options that best suit their unique needs. Automated lease generation and document verification will simplify the onboarding process for both parties. The platform also plans to introduce features like roommate agreements, rent-splitting tools, and emergency contact integrations. With continuous feedback loops and AI model retraining, Room Bridge aims to stay adaptive and improve its recommendation accuracy over time.

Tools / Platform/Methodology

The Room Bridge system leverages a carefully selected set of modern tools, platforms, and methodologies to ensure a secure, scalable, and user-friendly housing and roommate solution. The integration of advanced technologies such as AI, robust backend frameworks, and a responsive frontend ensures a seamless experience for both tenants and landlords. Below is an overview of the tools, platforms, and methodologies adopted:

1. Tools and Libraries

- **React.js (Vite):** Used for building the frontend user interface, offering component-based development, fast rendering, and responsive design for enhanced user experience in browsing listings and managing profiles.
- **Tailwind CSS / Material UI:** Utility-first CSS framework and component library used to design a modern, responsive, and visually appealing interface with minimal effort
- **Spring Boot:** A Java-based framework used to handle the core backend business logic, user authentication, secure API handling, and integration with the database.
- **Node.js:** A runtime environment used for real-time communication features such as chat, ensuring low-latency and scalable messaging between tenants and landlords.
- **MySQL:** A relational database used for structured data storage, including user profiles, room details, transaction records, and verification documents. Its ACID compliance ensures reliability and security in managing critical data.
- **Python (AI/ML):** Python with libraries such as scikit-learn, TensorFlow, and PyTorch is used to implement machine learning models for ranking properties based on ratings, reviews, and sentiment analysis.
- **GitHub:** Used for source code management, collaboration, and version control across multiple contributors.

2. Platform

- **Local Development Environment:** Initial development and testing take place on local machines with React.js, Spring Boot, Node.js, and MySQL installed.
- **AWS Cloud (Deployment):** AWS services such as EC2, S3, and RDS are used for deploying the application, storing user-uploaded documents, and managing cloud-based database operations.
- **GitHub:** Version control and team collaboration are handled via cloud repositories to maintain structured development workflows.

3. Methodology

- The methodology adopted for Room Bridge combines standard software engineering practices with AI-driven personalization:

- **User Registration & Verification:** Users register on the platform and upload identity documents, which are verified at the admin level to ensure authenticity.
- **Room Listing:** Landlords and tenants create detailed listings with room descriptions, images, and preferences for compatibility.
- **AI Recommendation Engine:** Using natural language processing and sentiment analysis, the system ranks rooms and roommates by combining user star ratings with analysis of comments and feedback.
- **Mutual Consent Chat:** Node.js enables a real-time chat feature, but conversations are only initiated after mutual consent, ensuring privacy and security
- **Landlord–Tenant Interaction & Reviews:** The system supports reviews, ratings, and direct interactions, fostering transparency and accountability.

4. JavaScript Libraries / Frameworks

- **React.js:** For building the responsive frontend.
- **Node.js:** For backend services that require real-time processing.
- **Express.js:** For lightweight APIs alongside Node.js chat services.
- **Tailwind CSS:** For building custom, responsive UI without writing lengthy CSS.

5. Database

- **MySQL:** Used for structured data such as user profiles, listings, verification documents, and financial transactions.

6. Version Control and Collaboration

- **Git:** For managing source code versions.
- **GitHub:** For repository hosting, collaborative workflows, and issue tracking.

Additional Tools

- **JMeter:** For load testing and performance evaluation of the platform.
- **Postman:** For API testing and debugging.
- **Docker:** For packaging and deploying backend services and AI models across different environments.

Hardware and Software Requirement specifications

To ensure the successful development, deployment, and maintenance of the Room Bridge platform, a well-defined set of hardware and software resources is required. These specifications have been chosen to balance performance, scalability, and cost-effectiveness while supporting AI-driven features, real-time chat, and secure data handling.

Hardware Requirements

1. Processor (CPU):

- Minimum: Intel Core i5 (8th Gen) or AMD Ryzen 5 equivalent.
- Recommended: Intel Core i7/Ryzen 7 or higher for handling multiple concurrent user requests, backend computations, and secure data transactions.

2. Memory (RAM):

- Minimum: 8 GB RAM to support local development, testing environments, and database operations.
- Recommended: 16 GB or higher for running backend services, AI models, and handling large datasets efficiently.

3. Storage:

- Minimum: 500 GB SSD for faster boot-up, data access, and application processing.
- Additional Storage: HDD or extended SSD space may be required depending on the size of user-uploaded documents, room images, AI model data, and logs.

Software Requirements

1. Operating System:

- Compatible with Windows 10 or higher, macOS, or Linux distributions such as Ubuntu 20.04 LTS or later.
- Linux-based environments are preferred for deployment due to their stability and efficiency in handling server workloads.

2. Programming Languages:

- Java (JDK 17 or higher): Used for backend development with Spring Boot.
- JavaScript (ES6+): For frontend (React.js, TailwindCSS) and Node.js-based chat service.

- Python 3.8+: For AI/ML modules such as property ranking, sentiment analysis, and compatibility checks.

3. Frameworks and Libraries:

- Spring Boot: For backend business logic, authentication, and API development.
- Node.js + Express.js: For real-time chat and lightweight API services.
- React.js (Vite): For building an interactive and dynamic frontend.
- TailwindCSS / Material UI: For modern, responsive user interface design.
- scikit-learn, TensorFlow, PyTorch: For AI/ML functionalities like ranking algorithms and sentiment analysis.

4. Database:

- MySQL (Primary): Used for structured data such as user details, room listings, and transactions.

5. Cloud Platforms and Deployment:

- AWS (Amazon Web Services): For deploying the platform, managing user traffic, and storing large datasets such as verification documents and room images. Services like EC2 (compute), S3 (storage), and RDS (database hosting) will be utilized.
- Docker: For containerization, ensuring consistent deployment across different environments.
- GitHub: For version control, repository hosting, and collaborative development.

6. Development and Testing Tools:

- Visual Studio Code / IntelliJ IDEA: IDEs used for coding, debugging, and integration.
- Postman: For REST API testing and backend integration validation.
- Jupyter Notebook / PyCharm: For developing and testing AI models.
- JMeter: For load and performance testing

Testing Technologies used

To ensure the accuracy, reliability, and overall functionality of the Room Bridge system, various testing tools and methodologies are employed. Specifically, the project leverages JUnit, Mocha/Chai, Postman, and JMeter for comprehensive testing of backend, frontend, and AI-driven modules.

1. Unit and Integration Testing:

- **JUnit (Java):** Used for backend unit testing in Spring Boot, validating core business logic, authentication flows, and API endpoints.
- **React Testing Library:** Focuses on frontend components such as room listings, profile forms, and chat windows, ensuring user interactions function correctly.
- **Simulated User Interactions:** Validates search filters, profile verification steps, chat initiation, and landlord–tenant interactions.

2. Database Testing:

- **MySQL Workbench:** Used to test schema integrity, relational constraints, and query performance for user data, listings, and transactions.
- **Data Validation Tests:** Ensure correct storage and retrieval of user profiles, room details, and verification records.

3. AI/ML Model Evaluation Metrics:

- **Accuracy:** Tests whether the AI ranking model correctly prioritizes rooms and roommates based on feedback and star ratings.
- **F1 Score & Confusion Matrix:** Helps balance precision and recall, especially for compatibility recommendations.
- **Sentiment Analysis Validation:** Confirms that user comments are classified correctly into positive, neutral, or negative sentiments.

4. Performance and Load Testing:

- **Apache JMeter:** Used to simulate high traffic, ensuring backend APIs and chat system can handle multiple concurrent users.
- **React Performance Profiling:** Ensures smooth rendering of room listings, chats, and dashboards under heavy load.

- **Latency Tests:** Measure response times in chat to ensure near real-time communication between tenants and landlords.

5. Security and Verification Testing:

- **JWT Authentication Testing:** Validates token generation, expiration, and misuse prevention.
- **Penetration Testing:** Simulates attack scenarios to test system resilience against SQL injection, XSS, and unauthorized access.
- **Document Verification Module Testing:** Ensures uploaded IDs and verification docs are securely processed and validated at the admin level.

Significance of the Project

The Room Bridge project addresses one of the most critical challenges faced by students, professionals, and landlords in urban and semi-urban areas—finding reliable, safe, and compatible housing and roommates. Unlike existing rental platforms that primarily focus on availability, Room Bridge emphasizes trust, verification, personalization, and communication, making it significantly more impactful.

Enhanced Trust & Safety

- By integrating admin-level document verification and secure authentication, the system minimizes fraud and ensures that both landlords and tenants are verified.
- This creates a safer rental ecosystem where users feel confident in their interactions.

AI-Driven Personalization

- The inclusion of AI-based ranking and compatibility recommendations ensures that users are not just matched by availability, but also by lifestyle preferences, reviews, and sentiment analysis.
- This reduces conflicts and improves roommate satisfaction.

Transparent Communication

- The mutual-consent chat system and direct landlord–tenant communication channel build trust and reduce misunderstandings.
- Tenants and landlords can negotiate, clarify, and finalize agreements without third-party manipulation.

Scalability & Future Expansion

- The project lays the foundation for a national-level housing and roommate ecosystem.
- Future integration of advanced features like AI chatbots, IoT-based smart home verification, and predictive rental analytics can further enhance its significance.