Rental & Service Management System Software Design

CSCI-P465/565 (Software Engineering I)

Project Team - 19

Ashutosh Gupta
Bharath Veldi
Carter Novak
Darrion Shack
Nithin Madhusudhana

---

1. Product Overview

- Motivated by the founder's own experience, the Rental & Services Management System project seeks to streamline the moving-in process for both in-state and overseas students.
- The concept fills in significant holes in the market for rental platforms by concentrating on apartment rentals and related services. The system, which was created especially for Indiana University of Bloomington students, provides comprehensive property listings that include vital details like the distance from supermarkets and campus amenities. Students no longer have to visit various platforms during the housing hunt because to the centralization of this data.
- Additionally, the initiative gives students access to multimedia elements, map views, user evaluations, and other resources that help them make educated decisions. It also makes it easier to acquire moving-in services like utilities setup and transportation. Through features like messaging and chat, the system fosters a sense of community, enhancing the overall experience for international students during their crucial transition period.

2. System Requirements

- **Operating Environment:** The system must operate seamlessly in web environments, catering to users accessing it from different devices and platforms.
- **Responsive Web Application:** It's designed as a web application using modern frameworks like React, with backend elements aligning with the MERN stack (MongoDB, Express.js, React, Node.js).

- **Database:** MongoDB is used for storing data like apartment listings, user profiles, and transactional data.
- **Authentication and Security:** Firebase is integrated for user authentication, including Multi-Factor Authentication (MFA).
- **Client-Side (Frontend):** Developed using React, providing the user interface for interacting with the application.
- **Server-Side (Backend):** Built with Node.js and Express.js, handling requests and managing external authentication through Firebase.

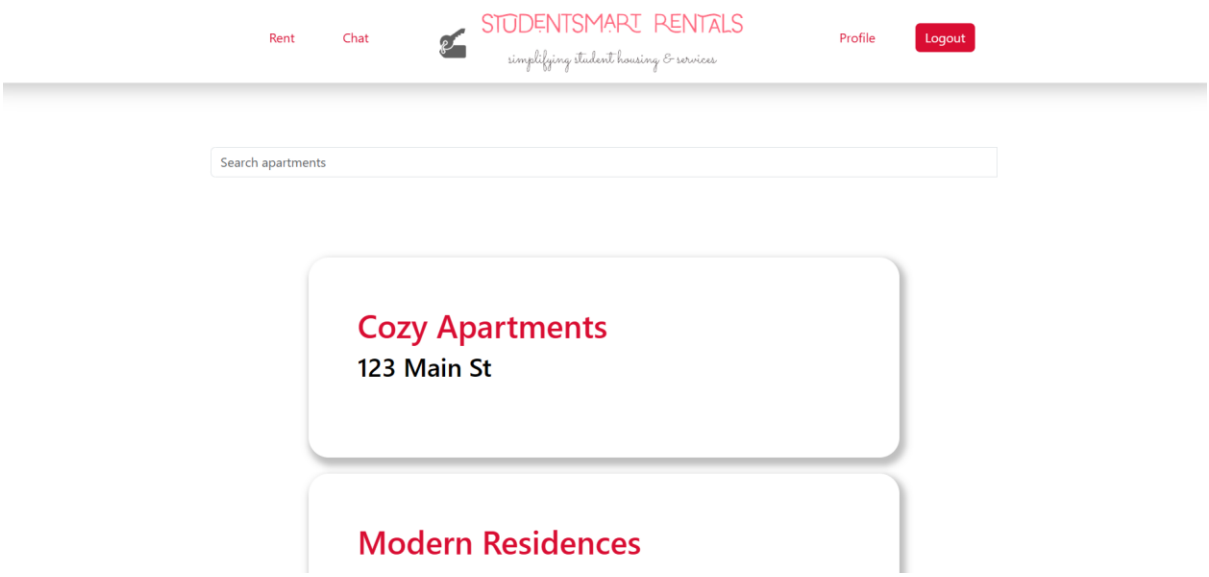
3. Product Installation

- **Initial Setup:** The software is a web-based application, thus it does not require a traditional installation process on a client machine. It is accessed through a web browser.
- **Dependencies:** Ensure that the browser is up to date to support the latest web technologies used in the application.
- **Software Support Packages:** For any backend setup or modification, familiarity with the MERN stack is required. Installation of Node.js, MongoDB, and other related technologies on the server-side may be necessary.
- **Python Dependency:** If there is a Python dependency (not explicitly mentioned in the documents), you can refer to Python's installation website for detailed installation instructions.

4. Product Operation

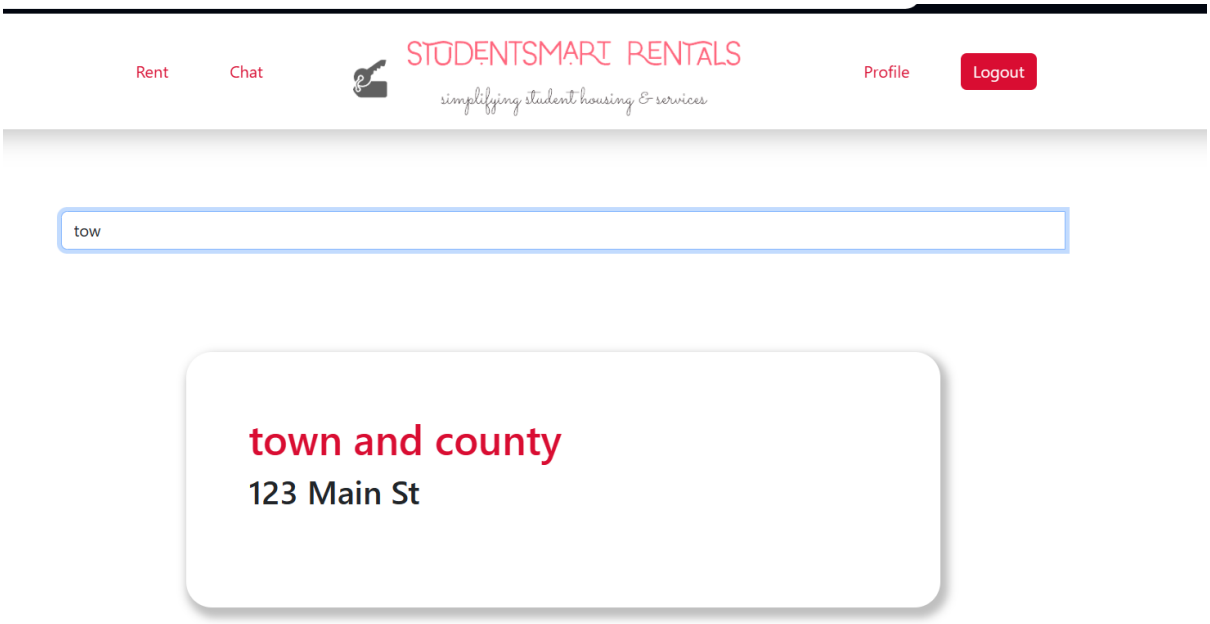
1. **Public Interface and User Registration:**

- The application provides a user-friendly public interface accessible without login. It displays available apartments for rent and other relevant items.
- Secure user registration and login system is in place. This likely includes standard user authentication as well as a secure system for password management and data encryption standards.



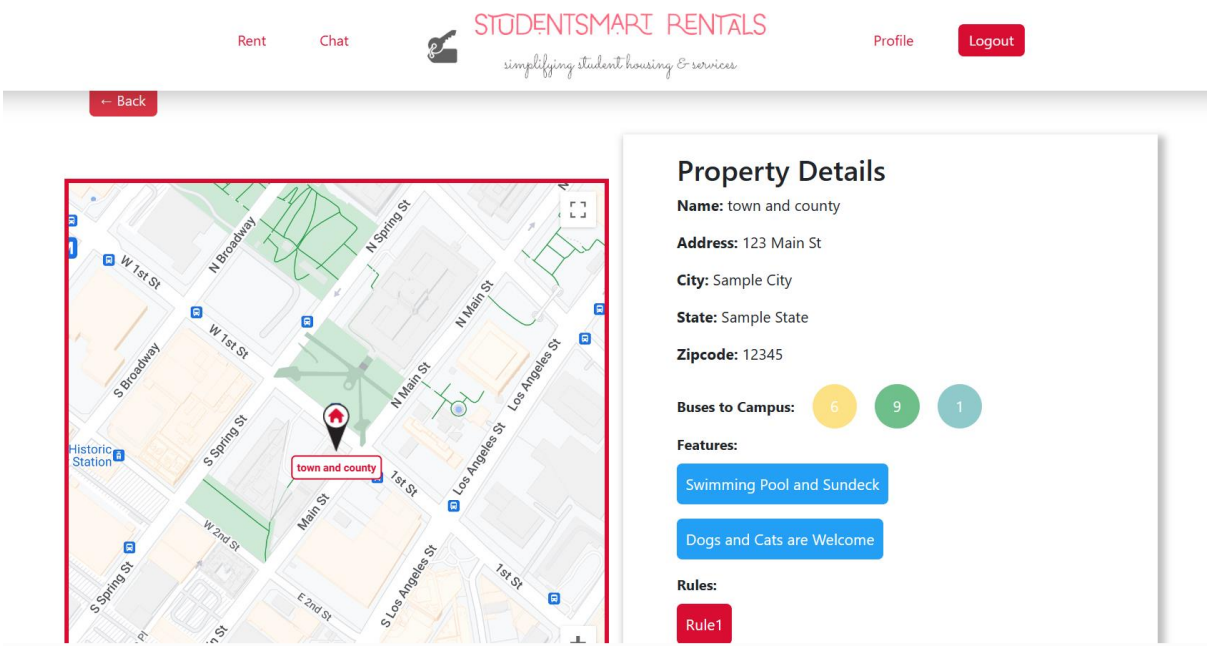
2. **Apartment Listing and Search Functionality:**

- Users can browse through apartment listings, which feature detailed profiles of each property.
- A search functionality is available, allowing users to perform searches based on different queries and filters. The SearchService class manages this, storing search queries, filters, and results, and displaying them on the user interface.



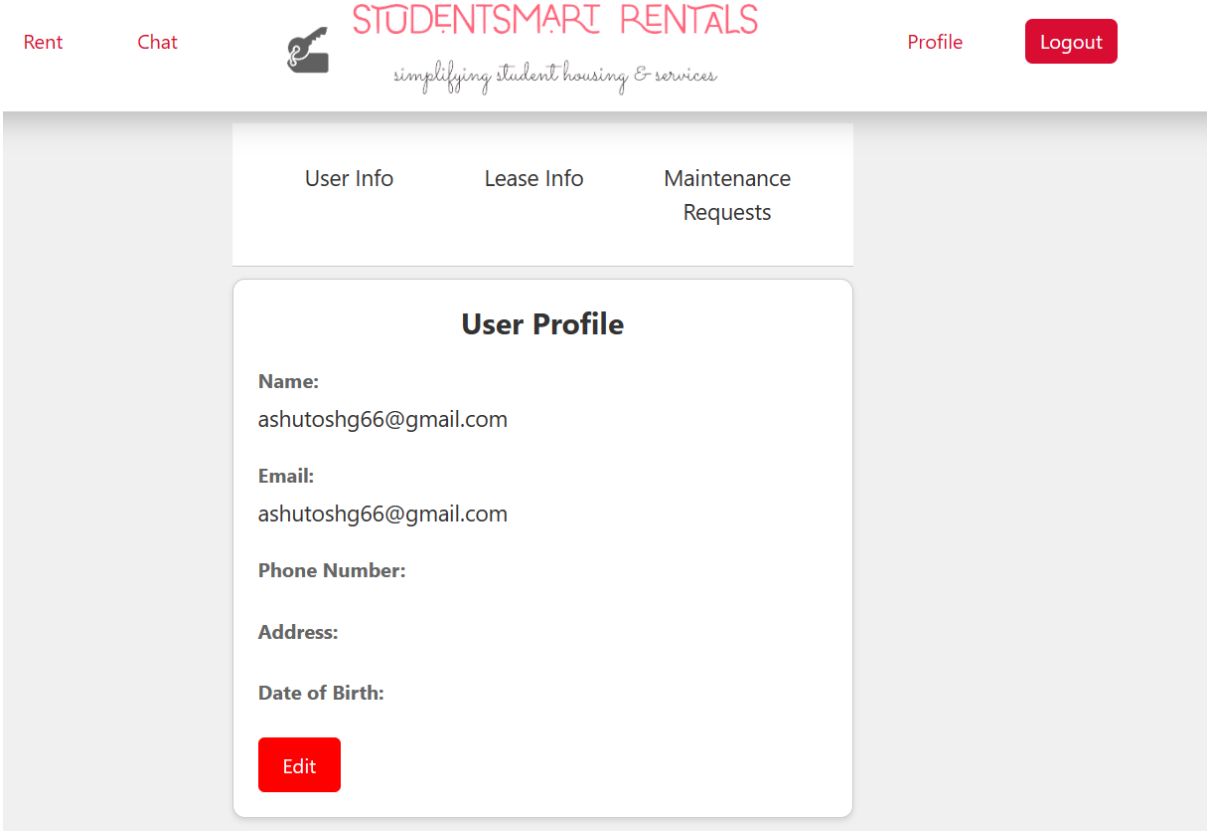
3. **Maps and Location Services:**

- The application integrates mapping services, likely using Google Maps or a similar API, to enhance the user experience by providing accurate property locations and proximity information.



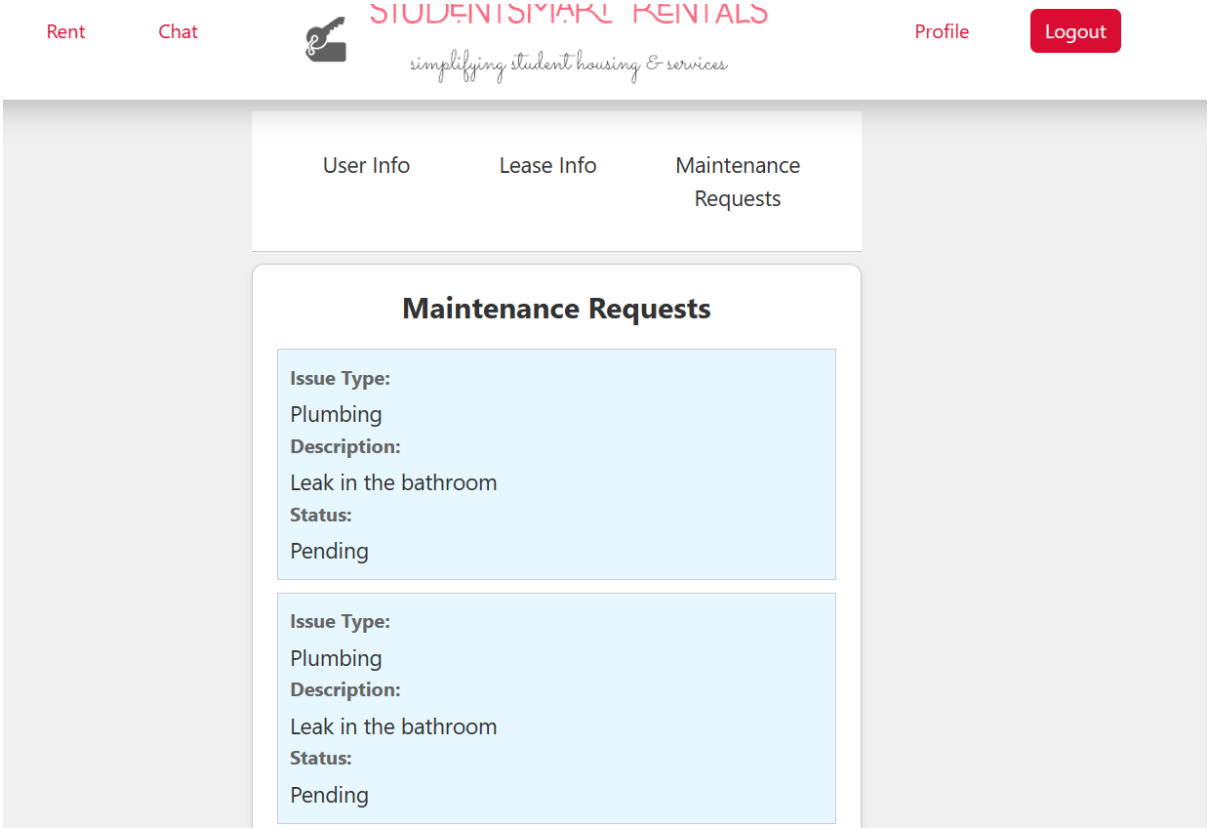
4. **User Profile Management:**

- Users have the capability to manage their profiles, including personal information, lease details, and maintenance requests. They can view and add new maintenance requests and edit their profile information.



5. **Maintenance Requests and Insurance Information:**

- Users can add maintenance requests and possibly access insurance information related to the rentals.

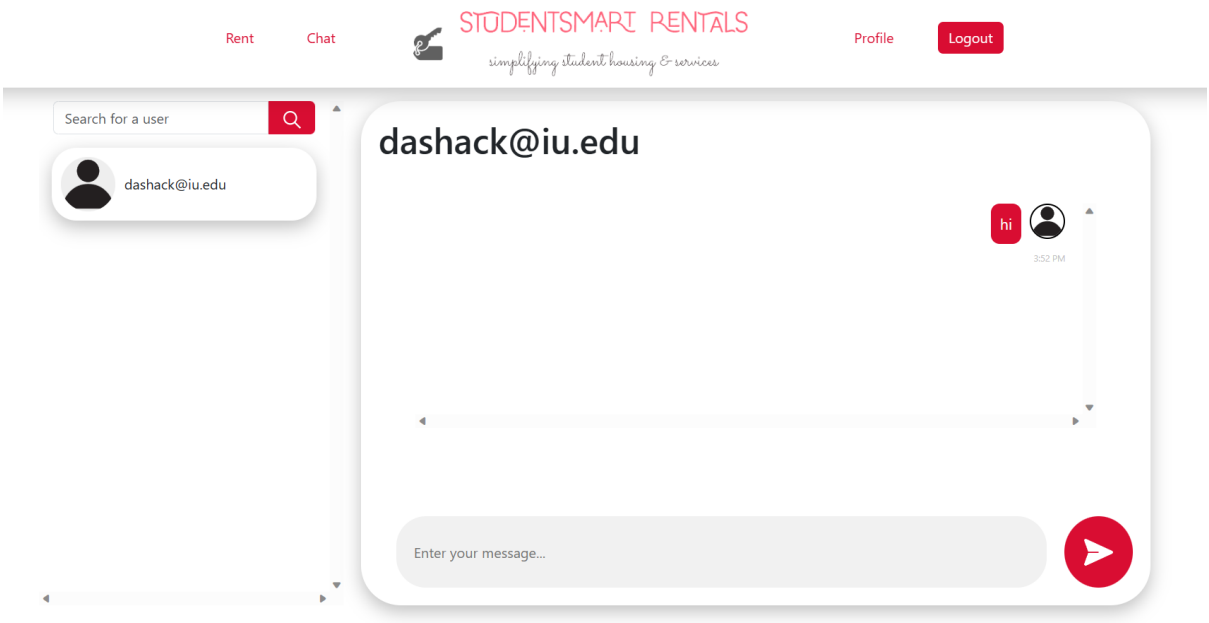


6. **User Interface and Accessibility:**

- The user interface adheres to established design principles, ensuring a consistent and user-friendly experience. The system is designed using HTML5, CSS3, and JavaScript, with a focus on data security and user-friendly navigation.

7. **User Chat:**

- Users can search for other users to chat with or find old chats. This functionality enhances the ability of users to connect with each other within the application
- The chat box enables users to send messages to the currently selected chat user in real-time, facilitating immediate communication and interaction



5. Known Defects

- Various components like Login, Sign Up, Reset Password, RentalsForm, Search, and Payment Gateway have specific error cases like invalid credentials, invalid input, data retrieval failures, and network or connectivity issues.

Revision History

Revision	Date	