BOSTON UNIVERSITY

Project Status Report 3

Team1
CS-673 (Software Engineering)

Fall 2024

By

Sergio Khalil

Yuhang Zhang

Asma Asiri

Brad Nissenbaum

Shivaang Kumar

Team Meeting Minutes:

Date: 10/22/2024

Attendees:

- Sergio Khalil
- Yuhang Zhang
- Asma Asiri
- Brad Nissenbaum
- Shivaang Kumar

Discussion Points:

- The importance of responsive design and how CSS is structured to ensure usability across devices.
- Ensuring that tenant status changes and other updates are dynamically reflected across components in real time.
- The architecture and modular design of components like AgentDashboard and AgentClientPage allow for code reusability and easier maintenance.
- How can the front end be integrated with back-end services or APIs to securely and efficiently fetch, update, and manage tenant data?
- Discussing the approach to testing individual components and ensuring data consistency across the application.
- Considerations for maintaining the application include handling edge cases, such as tenant information discrepancies and payment updates.

Capture Decision:

- Implementing context-based state management to maintain the consistency of tenant data across components ensures that updates to tenant information, such as changes in complaint status, are reflected in both the AgentDashboard and AgentClientPage.
- The decision to separate possible and current tenants into two tables improves the workflow for accepting and tracking tenants based on their status.
- Implementing editable fields for payment and status information directly on the AgentClientPage makes it easy for agents to update tenant data without navigating away from the page.

Action items:

- Ensure tenant data is consistently managed and shared across all components.
- Integrate the Context API fully to efficiently manage tenant status updates, payment information, and compliance status.
- Maintain two distinct tables for potential tenants and current tenants within the Agent Dashboard.
- Add clear action buttons (accept, edit, delete) for better workflow management, and ensure seamless transitions between tables based on tenant status.
- Develop and optimize editable forms on the Agent Client Page for tenant details, compliance status, and payment information.
- Build and test logic that automatically updates the compliance status based on payment due dates to mark tenants as "Payment Late" when necessary.
- Highlight such changes visually using CSS (e.g., red for late status) to improve agent response.
- Test and validate that changes are retained when the agent closes and reopens the platform.
- Conduct thorough testing to ensure that tenant status changes and information updates are accurately reflected between the Agent Dashboard and Agent Client Page.

Team Contribution:

Asma Asiri:I developed an agent-client platform using React.js and CSS, focusing on clean, functional user interactions. I built the AgentDashboard to manage potential and current tenants, including selecting, deleting, and accepting tenants with context-based state management to ensure updates are consistent across pages. I styled the dashboard with modern, responsive layouts and hover effects for better usability. On the AgentClientPage, I enabled agents to view and edit tenant details, including payment information and complaint status, with interactive dropdowns and automatic updates based on due dates. I ensured that the tenant details, housing info, and payment history were clearly presented using well-structured CSS for easy navigation.

Sergio Khalil: I implemented a feature that allows users to upload documents directly within a task. These documents are securely stored in AWS S3 to ensure better data protection and compliance. In addition, I integrated functionality in the backend that stores a reference to each document associated with a specific task, ensuring easy retrieval and management of files without storing them on the server.

Yuhang Zhang: I focused on the early planning and design of rental software projects. I analyzed key requirements, including price comparison and enhanced user experience for easy rentals. Based on these findings, I drafted wireframes for property listing pages and ensured the design was responsive. In addition, I researched front-end frameworks suitable for implementing these features.

Brad Nissenbaum: Led discussions with team about design needs and helped shape what features needed to be implemented moving forward. Reviewed designs and shared suggestions with team.

Shivaang Kumar: I contributed to enhancing the user interface and improving the design of the client-facing pages in Figma, focusing on creating responsive, user-friendly components. I worked on converting these designs into functional React components for key pages like the login and applicant info forms. To facilitate team collaboration and simplify the development process, I also helped develop a simplified Entity Relationship Diagram to clarify relationships between key components, making it easier to build the backend and frontend features.

- GitHub group project repo:

GitHub repository: https://github.com/sergioBU2024/CS673Team1-Real-Estate-Checklist-Tracker