

# Reflection on Uni-Stay Project Development – Individual Experiences & Growth

An expanded reflection highlighting everyone's experience and challenges, team growth and our next plans.

**Prepared by:** Group WO2 Members

**Project Phase:** Frontend Structure Complete | Backend & Database Integration Upcoming

**Date Due:** 14 May 2025

---

## 1. Sanele Trueman Zondi

**Role:** Project Leader & Frontend Developer

### Experience:

- As the team lead, I was responsible for aligning the team's progress with project goals, delegating tasks, and ensuring smooth communication.
- On the technical side, I led the frontend architecture, focusing on role-based UI (tenant vs. landlord dashboards), page routing, and access control.
- I implemented a JavaScript-based logic system that switches content based on user roles, enhancing the user experience.
- During mobile usability testing, I observed how our layout collapsed on smaller devices, which was a turning point for me to embrace mobile-first responsive design.
- I adopted a modular JavaScript structure with clear comments, which made it easier for others to understand and debug my code.

### Challenges:

- One of the biggest obstacles was ensuring that user roles persisted correctly, even after page reloads. This led to several redesigns and testing.
  - Our initial designs were desktop-focused; during mobile testing, major layout issues emerged, leading me to rework many interface components.
  - Balancing leadership responsibilities while actively developing tested my time management skills.
- 

## 2. Boitumelo Chantel Xaba

**Role:** Content & Documentation Lead, CSS Assistant and Secretary

### Experience:

- Helped shape the tone and clarity, button labels, tooltips, error messages, and onboarding prompts.

- Assisted with CSS styling, especially in ensuring text legibility, consistent button behavior, and form alignment.
- Took charge of creating and maintaining the documentation system, including naming conventions, content style guide, and dev notes.
- Introduced a content review system that ensured copy was approved before features were merged.
- Created a documentation style guide covering tone, terminology, and formatting rules.
- Coordinated with both developers and designers to make sure copy fit the layout and purpose.

#### **Challenges:**

- Difficulty persisting role-based views without a backend.
  - Frontend-only booking system limited by lack of real-time data validation.
  - UI inconsistencies on smaller screens and across browsers.
  - Often had to rewrite content that developers used as “temporary placeholders” because no one replaced it.
  - Designing the style guide sparked debates about branding, tone (formal vs. friendly), and consistency resolving this required careful collaboration.
  - Content was seen as a “final polish” rather than a core part of UX, which led to misalignment until the review system was enforced.
- 

### **3. Emmanuel Posholi Posholi**

**Role:** Frontend Developer

#### **Experience:**

- Transitioning from backend development to frontend was a steep learning curve. I had to familiarize myself with HTML, CSS and JS, event handling, and real-time UI feedback.
- I built the property search feature, which required handling filters, location-based queries, and dynamic list rendering.
- I implemented form validation systems (e.g., input checks, real-time feedback), refining them based on test feedback and UX reviews.
- I learned the importance of small UX details, like color indicators for password strength, and how they build user trust.
- I began exploring debouncing and throttling techniques to optimize performance for data-heavy features.

#### **Challenges:**

- My first version of the search function was resource-intensive, causing performance drops when listing more than 20 properties.
  - Adapting to frontend optimization strategies was new to me I had to research and apply quickly.
  - Ensuring that validation was not just functional, but also visually informative, pushed me to collaborate more with the design team.
- 

#### **4. Rendani Modika**

**Role:** Interactive Features Developer

**Experience:**

- Focused on building booking system logic including date range selection, availability validation, and calendar interface behavior.
- Took ownership of the messaging system that enables landlords and tenants to chat securely.
- Developed a habit of writing inline documentation and comments for complex logic, which helped other developers integrate with my code.

**Challenges:**

- Initially overlooked complex booking cases, such as:
    - Booking that spans across months.
    - Daylight saving adjustments (timezone shifts).
    - Double bookings or overlaps.
  - The chat interface had UI/UX bugs like message bubbles not aligning correctly, which made it difficult to scale for different screen sizes.
  - Testing date logic manually was exhausting; I had to build my own test harness to simulate different input scenarios.
- 

#### **5. Sisonke Mhlana**

**Role:** UI/UX Designer & Developer

**Experience:**

- Responsible for visual consistency, branding, and responsiveness across all devices.
- Created the design system, choosing fonts, color schemes, button styles, and layout templates.
- Introduced animations and transitions that improved interactivity (e.g., hover states, modal transitions, loading indicators).

- Developed fallback strategies for unsupported CSS properties to ensure compatibility across Chrome, Firefox, Safari, and some mobile browsers.

### **Challenges:**

- Ran into issues with CSS Grid not rendering consistently on IOS and other mobile browsers.
  - Our early animation scripts caused frame drops on low-end phones; I had to learn about GPU acceleration and performance budgets.
  - Creating a visually rich UI while staying lightweight and accessible was an ongoing balancing act.
- 

## **6. Zukhanye Anele Mene**

**Role:** Secretary and Quality Assurance Lead

### **Experience:**

- Developed a manual testing workflow that included cross-device and cross-browser testing.
- Created an accessibility checklist to ensure color contrast, screen reader support, and keyboard navigation were properly implemented.
- Reported dozens of UI/UX issues that would have otherwise gone unnoticed, improving the polish and inclusiveness of the product.
- Introduced the idea of “empathy testing”—trying to use the app like someone with low vision, or on a slow network.

### **Challenges:**

- Many early builds lacked alt text, ARIA labels, and had poor color contrast.
  - Discovered layout bugs that only appeared on iPhones running older iOS versions—these were difficult to debug due to limited test devices.
  - Faced resistance initially when trying to push for QA inclusion during development, rather than just pre-launch.
- 

## **Team-Wide Growth & Lessons**

### **Growth:**

- Our daily standups evolved from mere status updates into valuable discussion and problem-solving sessions.
- Embraced cross-discipline collaboration: developers began to understand design constraints, designers learned about code structure, and everyone respected QA.
- Developed a shared documentation hub that made onboarding and debugging easier.

- We discovered the importance of planning backend architecture early, even during frontend-only phases.
- We developed code review habits that improved code quality, caught bugs earlier, and encouraged team learning.

### Challenges:

- Building the Website without a backend initially led to workarounds that broke once the app scaled.
- We faced state management problems (like role switching and data persistence) due to missing architectural planning.

---

## Planned Next Phase (Backend & Database Integration)

We plan to implement:

- **Backend Technologies:** PHP with MySQL.
- **Database Tables:**
  - Users (ID, role, contact info)
  - Properties (ID, landlord ID, title, images, price)
  - Bookings (tenant ID, property ID, dates, status)
  - Messages (from, to, content, timestamp)
- **Features to Build:**
  - Secure login/registration with role-based access
  - Real-time bookings with availability checking
  - Message storage and notifications.
  - Landlord upload and tenant request systems
- **Future Improvements**
  - Build a component-based frontend for scalability.
  - Integrate backend earlier for better state management.
  - Set up automated testing pipelines for both UI and logic.
  - Strengthen documentation (Boitumelo leading this with markdown templates, guides, and CSS notes).
  - Implement performance tracking and accessibility audits.

While our Uni-Stay Website is still in development, the work completed has laid a strong foundation. Each one of us has grown not only in technical skill but also in teamwork and adaptability. We're now better prepared to enter the backend phase with a shared understanding of both our user needs and technical goals.