PropEase - Real Estate Management System

Weekly Progress Report – Sprint 3

(29 September – 5 October 2025)

—--

Summary of Tasks Completed

This sprint focused on refining and expanding the **User Module** by introducing personalized features and strengthening backend integration.
Core goals included creating the **User Profile Page** and **Favorites functionality.**

Frontend Development

- **User Profile Page:**
  - Developed UI for displaying and updating user information (name, email, contact number).
  - Added editable fields and success/error prompts on save.
  - Integrated profile update API with form validation.
- **Favorites (Saved Properties):**
  - Added "Save Property" button on property cards and detail pages.
  - Created "My Favorites" section in the user dashboard.
  - Implemented state management for adding/removing favorites using Context API.
- **UX Enhancements:**
  - Improved navigation flow and added toast notifications for key actions (login, save, update).
  - Enhanced mobile responsiveness of search and property detail components.

Backend Development (Node.js + MongoDB)

- **User Profile APIs:**
  - Developed GET and PUT endpoints for retrieving and updating user details.
  - Added authentication middleware for secure access to user-specific routes.

- **Favorites Management APIs:**
  - Created endpoints for adding, retrieving, and removing favorite properties.
  - Updated User schema to include an array of saved property IDs.
  - Integrated with frontend components successfully.

- **Data Validation & Error Handling:**
  - Added robust error handling for invalid tokens and missing fields.
  - Implemented Joi validation for incoming requests.

Documentation

- Updated **API documentation (Swagger)** for user and favorites endpoints.

Roadblocks / Challenges Faced

Issue	Description	Resolution
API Authorization Error	Some protected routes returned unauthorized responses even for valid tokens.	Adjusted middleware order in Express routes and revalidated token structure.
Favorites Sync Delay	UI state didn't instantly reflect updated favorite items after API response.	Used Context for instant UI updates post-response.
Profile Data Persistence	Profile changes weren't reflecting immediately.	Added frontend re-fetch on successful profile update.
Slow Rendering on Low-End Devices	Property listings and map view took longer to load.	Deferred heavy components and introduced lazy loading.

Upcoming Tasks

The next sprint will focus on **finalizing the User Module** and conducting the **mid-semester demo preparation**.

Planned Tasks:

- **Frontend:**
    - Integrate **Google Maps API** for location-based property visualization.
    - Add **EMI Calculator** page and link to property details.
    - Complete UI polishing and final bug fixes.

- **Backend:**
    - Integrate **Map-based property search API**.
    - Begin **Chat feature backend structure** using Socket.io.