

Wellie Pet – Sprint 2 Technical Documentation

1. Database Schema and Relationships

We are using firebase to store user's info and pet-related data(in the future).

- User: Stores user profile info (e.g., name, account number, passwords).
- PetStatus: Tracks virtual pet mood, health level, and state.(In the future)
- HealthLog: Logs daily health metrics (steps, sleep, hydration).

Relationships:

- One-to-one: User → PetStatus
- One-to-many: User → HealthLog

Future Plan:

- Integrate Firebase for cloud sync and multi-device support.

2. APIs Used and Integration

- OpenWeatherMap API:
 - Used to fetch real-time weather data.
 - Integrated with Retrofit and Moshi converter.
- Health Connect API:
 - Used to access user's health data: steps, sleep, and hydration.
 - Supports permissions through the Accompanist library.

3. Code Organization and Project Structure

- Modular folders for clear separation of concerns:
 - /ui/mobile: Phone UI screens using Compose.
 - /ui/wear: Watch-specific UI using Wear Compose.
 - /data/model: Data models like User, PetStatus, HealthLog.
 - /data/repository: Logic for accessing Room and APIs.
 - /data/source: Weather and Health Connect sources.
 - /domain/usecase: Encapsulated business logic.
 - /navigation: AppNavHost and screen routing.
 - /utils: Utility functions, constants, and extension methods.

4. Notable Challenges and Solutions

- UI Scaling on Watch:
 - Problem: Mobile layouts didn't fit small Wear OS screens.
- Bug:
 - Might be a redirect bug after login, takes a long time to load.