Sprint 0 Submission - PetFinder (Team Sunrise Inc)

Team Information and Introduction

Team Name:
Sunrise Inc

Project Name:
PetFinder

Team Members:
Ojaswi Subedi, Tanvi Biswal, Kimberley Juarez, Anjali Fernando, Bibek Pandey

Overview

Problem Context:
Pet owners face stress and delays when a pet goes missing. Information about pets and contacts is often scattered, causing slow response times.

Goal:
PetFinder provides a local-first mobile app that stores pet info and generates alerts (social post, email, SMS, phone script, flyer) with one tap.

Requirements

Functional Requirements (User Stories)

- R1 Pet Profile: Create/update pet profiles with photo and details (local storage).
- R2 Emergency Contacts: Add/edit vet, shelter, police contacts with one-tap call/email/SMS.
- R3 Missing Pet Alert: Generate social media post, email/SMS draft, and phone script instantly.
- R4 Regular Update Prompts: Local notifications reminding owners to update info/photos.
- R5 Photo Gallery: Upload/view multiple pet photos locally.
- R6 Recommendations: Static safety tips and search strategies.
- R7 Toxic Foods Infographics: Quick visual guides for dangerous foods.

Non-Functional Requirements

- NF1 Security: All data stored locally; optional OS-level lock (FaceID/Passcode).

- NF2 Reliability: Core features work offline (profiles, contacts, alerts).
- NF3 Usability: ≤3 taps to reach any function; emergency-first button on home.
- NF4 Accessibility: Support large text, high contrast, screen reader labels.
- NF5 Cross-Platform: Runs consistently on iOS 14+ and Android 10+.
- NF6 Accuracy: Generated alerts match stored details exactly.

Design

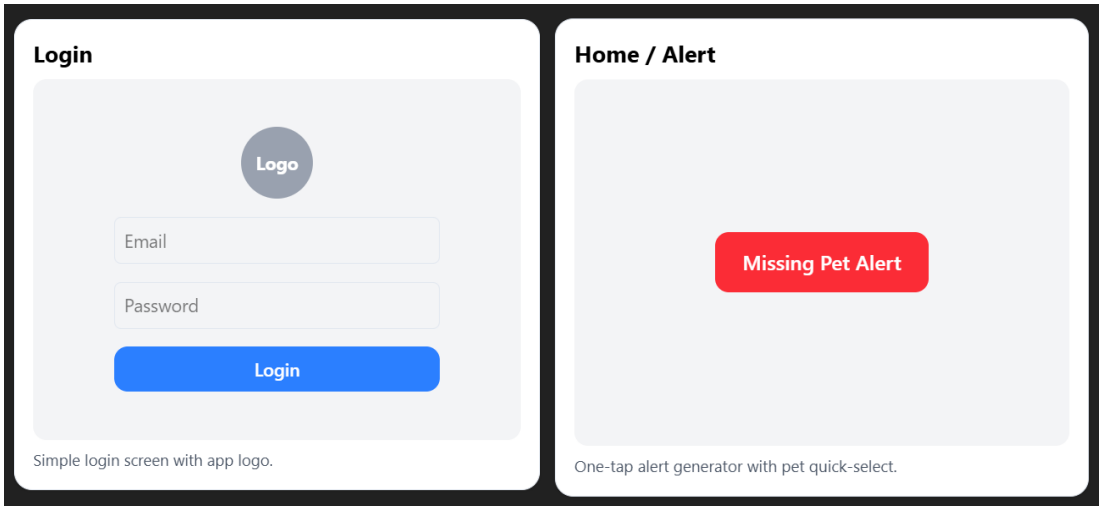
The app uses a tab-based navigation system:

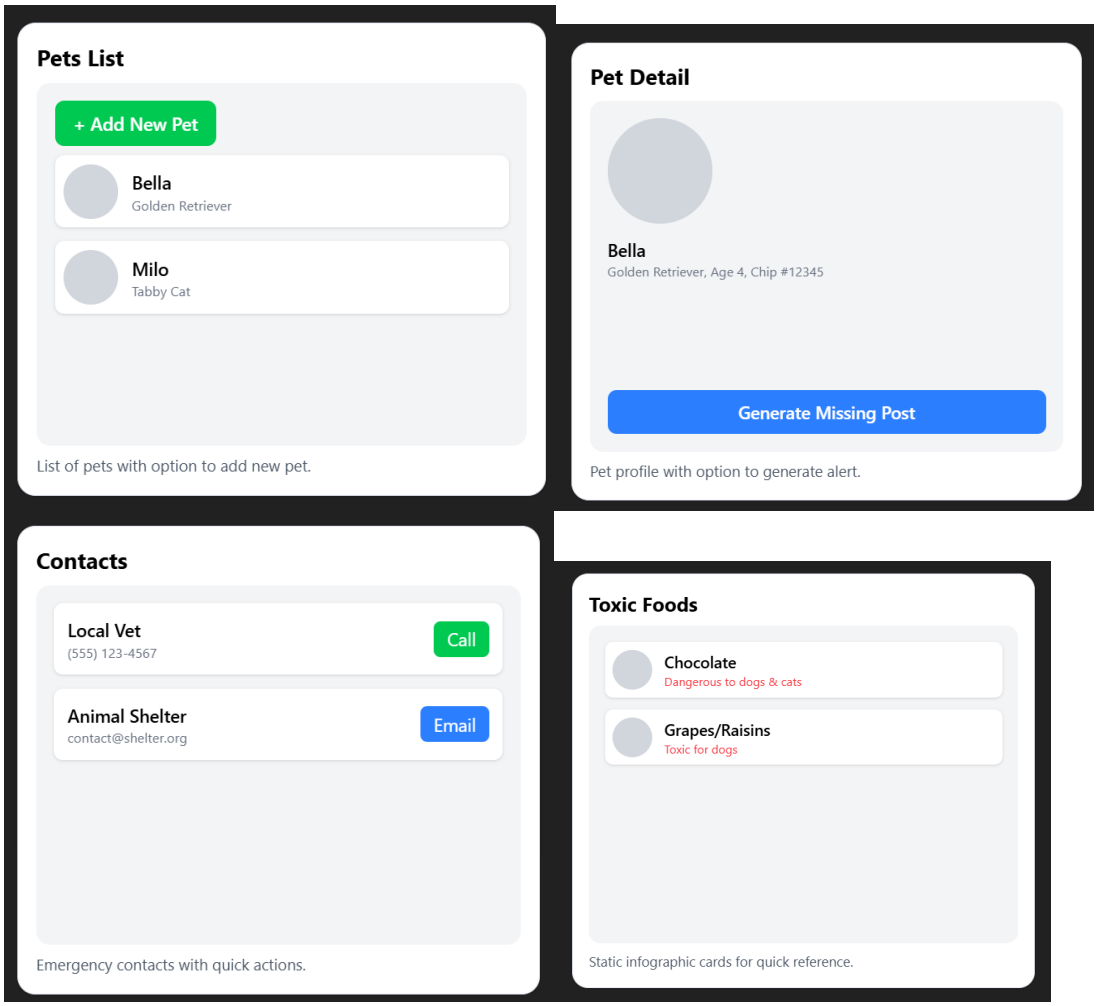
- Home: Big Missing Pet Alert button
- Pets: List of pets, access to detail screen
- Pet Detail: Shows info + generate alert option
- Contacts: Emergency contacts with quick actions
- Toxic Foods: Static infographic cards

Wireframes

Wireframes have been prepared for core screens.

Design rationale:
We chose a local-first design for simplicity, offline reliability, and reduced complexity versus cloud solutions.





Reflection Report

Accomplishments

We finalized the project scope as a local-first MVP, completed the requirements document, designed wireframes, and set up a GitHub and Trello for ease of collaboration. Roles and tasks were assigned that suited each team member best.

What Went Well

The team collaborated effectively during class meetings and quickly aligned on focusing on an offline MVP. Ideas for features and requirements were shared openly and documented clearly. Communication was handled effectively, and any problems were resolved smoothly.

What Could Be Improved

One area where our team should improve in is time management. Rather than waiting until the day of the assignment, we should start a few days early, giving us more time to think of

ideas and complete the task without feeling pressured. This will also allow us the time to find solutions to any problems that may arise.

Plans for Sprint 1

Sprint 1 will focus on mainly on design elements:

- Class Diagram
- Sequence Diagram
- State Diagram
- Refining Wireframes

Rubric Reflection

What went well in the presentation?

None of the team members talked over each other, and each team member had his or her own project section to talk about.

What can be improved in the presentation?

The team should decide beforehand who will speak on his or her topic first. The scope of the should also be defined.

Other Notes

...