



Requirements Document

Project: FurSure

Table of Contents

<u>DOCUMENT CHANGE HISTORY</u>	<u>2</u>
<u>1 OVERVIEW</u>	<u>2</u>
1.1 ASSUMPTIONS	2
1.2 CONSTRAINTS	3
1.3 DEPENDENCIES	3
1.4 DOCUMENT REFERENCES	4
1.5 ACRONYMS AND DEFINITIONS	4
1.6 STAKEHOLDERS	4
<u>2 FUNCTIONAL REQUIREMENTS</u>	<u>5</u>
2.1 REQUIREMENT DESCRIPTIONS	5
2.2 SCREEN MOCKUPS	6
<u>3 NON-FUNCTIONAL REQUIREMENTS</u>	<u>7</u>
3.1 PERFORMANCE REQUIREMENTS	7
3.2 LOAD REQUIREMENTS	7

Document Change History

Revision Number	Revision Date	Revision Description
0.1	02/09/2025	Initial draft of SRD
1.0	2/15/2025	Final draft of SRD

1 Overview

According to the Cornell Feline Health Center, maintaining accurate medical records for pets is essential for their long-term well-being. Vaccination schedules, medication reminders, and routine health checkups are crucial in preventing disease and ensuring a higher quality of life for pets. However, many pet owners struggle with organizing and accessing this vital information when needed.

To address this challenge, we developed a streamlined and comprehensive solution for pet health management. By integrating features such as vaccination tracking, medical record storage, and reminders for vet visits and medications, this solution makes pet care easier and more efficient. Additionally, with the ability to sync with smart pet devices like automatic feeders and litter monitoring systems, this solution helps pet owners take a proactive approach to their cats' health. As we like to say, don't just be sure—be FurSure, with the ultimate tool for pet health tracking and management.

1.1 Assumptions

The development of this software is based on the following assumptions, which have been sorted according to their category:

Users

- Users will have internet access for most features except vaccine record access, which can be viewed offline.
- Users will own one or more cats and require a way to track their pet's health data.
- Users will have mobile devices as the initial release will be mobile-only, with a web version planned in the future.
- Users will be able to log in with a username and password, with biometric login as an optional feature.

Secondary Users (Veterinarians)

- Vets will not input data directly but will be able to send update messages to users.
- Vets won't have direct access to pet records but can provide recommendations through the system.

Technology

- Automatic feeders utilized by users will support microchip-based identification.
- Storage will be cloud-based or local, depending on what's most cost-effective and practical

Security & Access

- 2FA will only be required when logging in from an unrecognized device.
- Vaccine records will be the only information accessible offline; all other features require an internet connection.

1.2 Constraints

The constraints for this software are everchanging and are subject to modifications as follows:

1. Platform Constraint: The initial release will be mobile-only, with a web version planned for later development.
2. Integration Constraint: The app must integrate with third-party APIs for Litter-Robot and automatic feeders that use microchip-based identification, acknowledging that these APIs may change or become unavailable without notice.
3. Security & Privacy Constraint: The system must comply with data protection regulations, as these laws regulate how pet *owners*' personal and health-related data is collected, stored, and shared, ensuring that their information is protected and cannot be used without their consent.
4. User Access Constraint: The maximum number of users per account is limited to four.
5. Storage Constraint: The system must support unlimited pet entries per user while ensuring scalable storage options (cloud or local, based on feasibility).

1.3 Dependencies

As we develop this software further, we are more than likely going to reconsider our dependencies on whether we are going to keep or remove some. So far, this is our current dependencies that we are focusing on:

Litter-Robot API (pylitterbot)	An unofficial Python package that supports Litter-Robot 3, Litter-Robot 4, and Feeder-Robot. Since the official API is not open-source, this third-party solution enables control over these devices but may change without notice.
--	---

Automatic Feeder API (Tuya Smart Pet Feeder Solution)	A third-party API solution that enables feeding schedules, manual feeding, and consumption tracking for micro-chip based pet feeders.
Cloud/Local Storage (Google Cloud Storage)	A scalable cloud-based object storage service to securely store and retrieve pet health data with high availability.
Push Notification Service (OneSignal)	A push notification service supporting mobile push, web push, SMS, and email notifications to alert users about vaccination dates, vet visits, and feeding schedules.
Mobile Platform Compatibility (Android & iOS SDKs)	Official development kits for ensuring compatibility with Android and iOS devices, providing a seamless cross-platform user experience.

1.4 Document References

“Best Mobile Push Notification Software.” *OneSignal*, onesignal.com/mobile-push.

“Choosing and Caring for Your New Cat.” *Cornell University College of Veterinary Medicine*, www.vet.cornell.edu/departments-centers-and-institutes/cornell-feline-health-center/health-information/feline-health-topics/choosing-and-caring-your-new-cat.

Natekspencer. “GitHub - Natekspencer/Pylitterbot: Python Package for Controlling a Whisker Connected Self-cleaning Litter Boxes and Feeders.” *GitHub*, github.com/natekspencer/pylitterbot?utm_source=Pet%20Camera-TuyaOS-Tuya%20Developer&utm_medium=referral&utm_campaign=Pet%20feeder%20features

1.5 Acronyms and Definitions

Term / Acronym	Definition
2FA	Two-factor authentication – a form of multi-factor authentication, is a security process in which users provide two different authentication factors to verify themselves.
API	Application Programming Interface – a software intermediary that allows two applications to talk to each other. They are an accessible way to extract and share data within and across the internet.
SDK	Software Development Kit – a set of tools and resources that allows developers to create software applications for a specific platform, operating system, or device.

1.6 Stakeholders

Name	Role	Phone	Email
Binky Barnes	Product Owner	571-555-1963	elwoodcitybully@gmail.com
Frodo Bagel	Marketing and Outreach Coordinator	202-555-1414	frodo.bagel@ringaround.com
Aragorn to be Wild	Quality Assurance & Testing Lead	406-555-1248	aragorn@rangermail.com
Lego Lass	User Representative	703-555-0201	legolass@elfshot.com
Gimli Locks	Lead Software Engineer	415-555-7845	gimli@dwarvensecurity.com

2 Functional Requirements

The following table lists the key functionalities of the system, grouped by purpose and providing an overview of essential requirements:

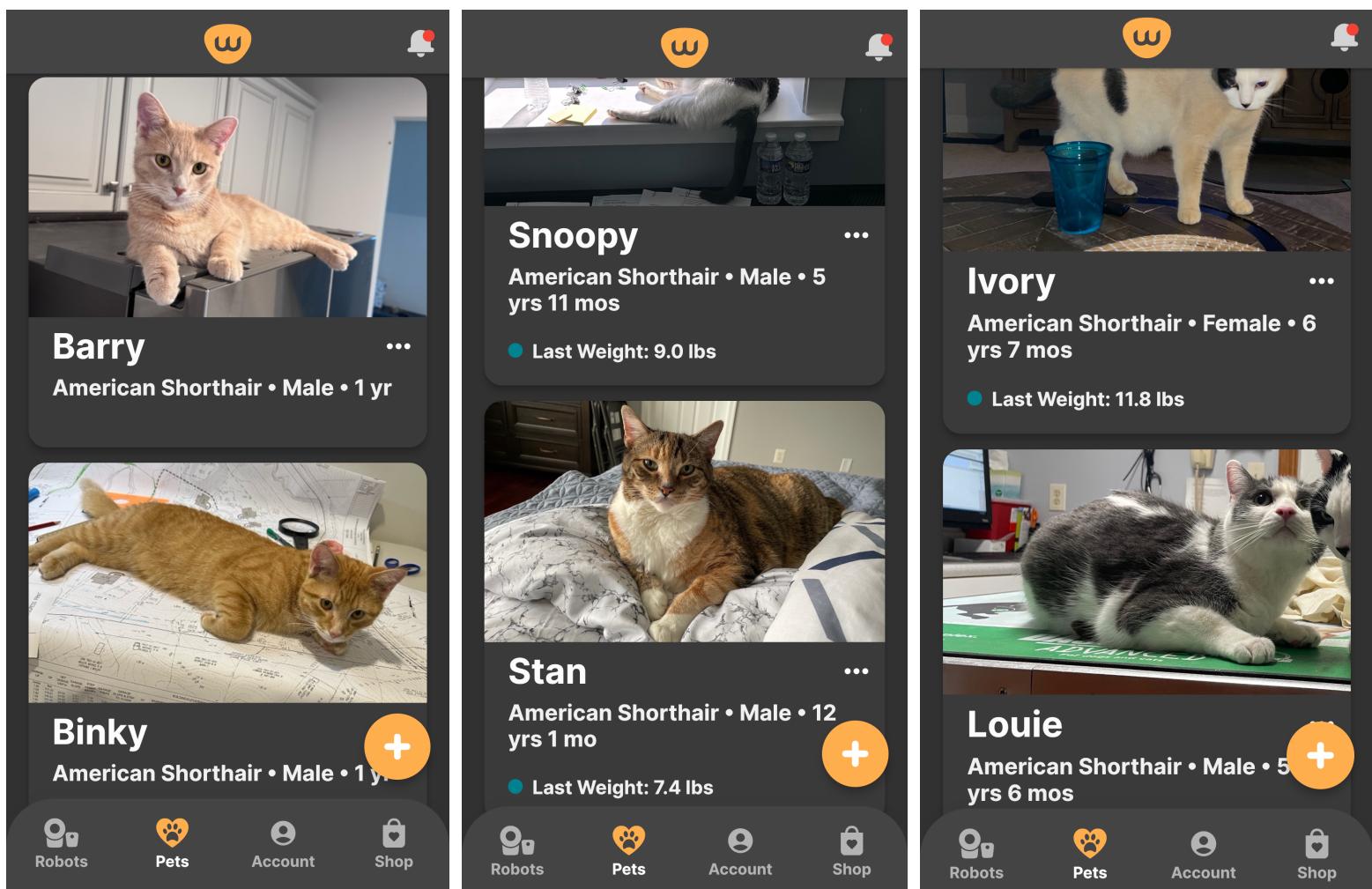
2.1 Requirement Descriptions

Requirement Number	Requirement Description
2.1	Users must be able to register, log in, and access their accounts securely.
2.2	Users must be able to create edits and view, pet profiles, including health and diet.
2.3	The system must send reminders for vaccinations, visits, and medication schedules.
2.4	The app must integrate with third-party API's.
2.4.1	Integration with Litter-Robot API (third-party)
2.4.2	Integration with smart-feeder API (third-party)
2.5	Users must be able to view feeding and waste trends through device integration.
2.6	The app must allow multiple users (up to 4) to manage shared pet profiles.
2.7	Vaccine records must be accessible offline; all other features require Internet access.

2.2 Screen Mockups

This section will include visual representations of the app, including:

- Dashboard (health stats, reminders, feeding schedules)
- Pet Profile Page (detailed health and diet)
- Integration Settings (linking devices)
- Notifications (alerts and reminders)
- Vet Communication Panel (messages from veterinarians)



This is how multiple pet profiles are viewed in the Whisker app (which houses the Litter-Robot app).

It will look similar to this as this will be integrated into the FurSure app

3 Non-Functional Requirements

3.1 Performance Requirements

The system must meet the following performance criteria:

Requirement Number	Requirement Description
3.1.1	Application should load within 2 seconds on standard mobile devices.
3.1.2	Data retrieval (e.g., viewing pet records, retrieving feeding history) should not exceed 3 seconds .
3.1.3	App should support real-time notifications for reminders and vet messages.
3.1.4	System should handle multiple users managing multiple pets without performance degradation.

3.2 Load Requirements

The system must be able to handle the following: recurrent

Requirement Number	Requirement Description
3.2.1	Up to 25 pet records per user without affecting response time.
3.2.2	Up to 1000 concurrent users without significant lag.
3.2.3	Up to 1,000 API calls per minute for real-time synchronization with Litter-Robot and automatic feeders.
3.2.3	Scalability to accommodate increased users and pets as the application grows.