

Sheltr

System Requirements

Matthew Fante

INFO-C451: System Implementation (Spring 2024)

Table Of Contents

CUSTOMER PROBLEM STATEMENTS & SYSTEM REQUIREMENTS3

PROBLEM STATEMENT3

FUNCTIONAL REQUIREMENTS3

NON-FUNCTIONAL REQUIREMENTS5

SYSTEM SEQUENCE AND ACTIVITY DIAGRAMS6

USER INTERFACE SPECIFICATION7

PROJECT PLAN.....7

REFERENCE.....6

GLOSSARY OF TERMS6

Customer Problem Statements & System Requirements

Problem Statement

Adopting a pet should be a joyful journey, not a communication maze. Animal shelter employees struggle with overwhelmed inboxes and limited hours, while potential pet parents face dead ends and delayed connections. Sheltr transforms the experience. Our user-friendly app brings shelters and adopters face-to-digital-face, enabling instant communication, streamlined scheduling, and effortless meet-and-greets. With Sheltr, finding your furry forever friend is just a tap away, leaving frustration behind and paving the way for happy pets and pet parents.

Functional Requirements

Requirement #	Priority	Impact Group	Description
REQ-01	1 - HIGH	All Users	All users should be able to browse animal profiles based on various filters like breed, size, age, energy level, etc.
REQ-02	2 - MEDIUM	All Users	All users should be able to view detailed animal profiles featuring photos, personality descriptions, and additional information.
REQ-03	1 - HIGH	All Users	All users should be able to create an account with a valid email address.
REQ-04	3 - LOW	Shelter Staff, System Administrators	Shelter accounts should undergo an approval process before posting adoption profiles.
REQ-05	2 - MEDIUM	Shelter Staff	Shelter staff members should be able to create detailed profiles for each animal, including photos, personality descriptions, adoptability criteria, and vet records.
REQ-06	2 - MEDIUM	Shelter Staff	Shelter staff members should be able to edit and update animal profiles as needed.
REQ-07	1 - HIGH	Shelter Staff	Shelter staff members should be able to mark animals as available or adopted.
REQ-08	2 - MEDIUM	Shelter Staff	Shelter staff members should be able to view and process meet-and-greet requests from potential adopters.
REQ-09	2 - MEDIUM	Shelter Staff	Shelter staff members should be able to approve or decline meet-and-greet requests.
REQ-10	3 - LOW	Shelter Staff	Shelter staff members should be able to view a schedule of upcoming meet-and-greet appointments within a built-in calendar.

Matthew Fante

INFO-C451: System Implementation (Spring 2024)

03/24/24

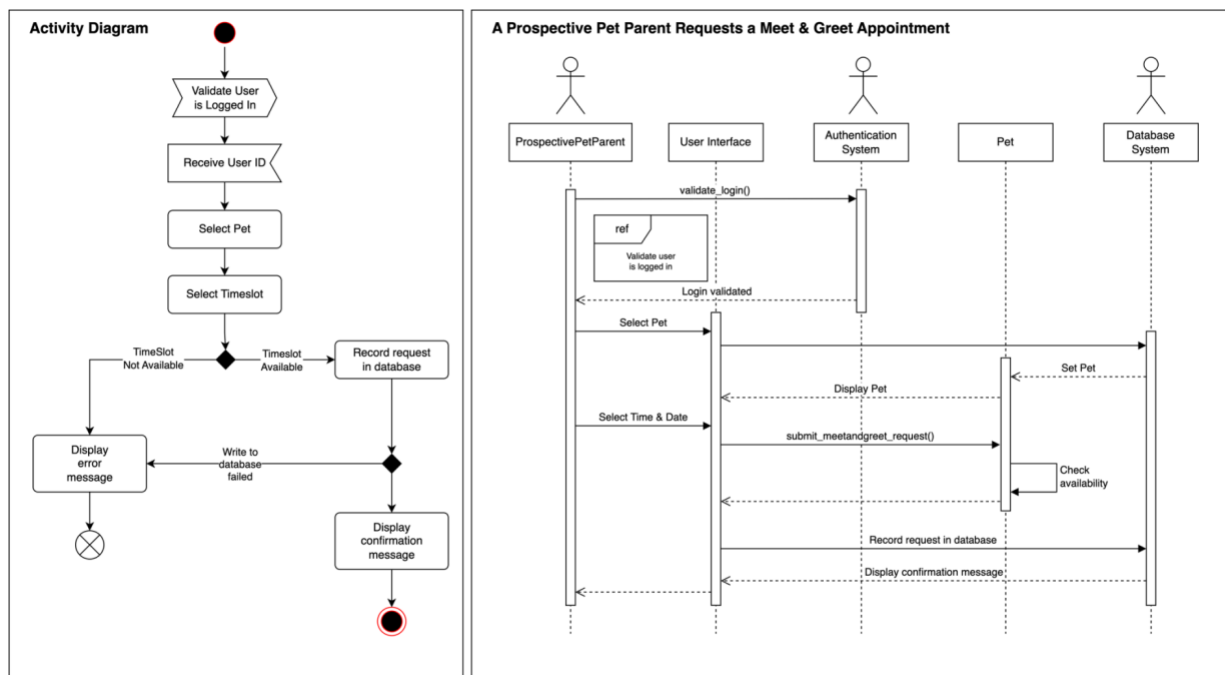
REQ-11	2 - MEDIUM	Shelter Staff	Shelter staff members should be able to view requester contact information (phone number or email address).
REQ-12	1 - HIGH	Shelter Staff	Shelter staff members should be able to update shelter information, including contact details, hours of operation, and adoption procedures.
REQ-13	3 - LOW	Prospective Pet Parents	Prospective pet parents should be able to mark favorite animals for easy reference.
REQ-14	2 - MEDIUM	Prospective Pet Parents	Prospective pet parents should be able to submit meet-and-greet requests directly from animal profiles with preferred dates and times.
REQ-15	2 - MEDIUM	Prospective Pet Parents	Prospective pet parents should be able to track the status of submitted requests and receive notifications of approvals or declines.
REQ-16	3 - LOW	Prospective Pet Parents	Prospective pet parents should be able to create and manage user profiles, including basic information and adoption preferences.
REQ-17	2 - MEDIUM	Prospective Pet Parents	Prospective pet parents should be able to update contact information and notification settings.
REQ-18	4 - OPTIONAL	All Users	All users should be able to access educational resources about pet care and adoption.
REQ-19	4 - OPTIONAL	All Users	All users should be able to read and leave reviews for shelters.
REQ-20	4 - OPTIONAL	All Users	All users should be able to share adopted pet stories and photos with the community.
REQ-21	4 - OPTIONAL	All Users	All users should be able to communicate directly with shelter staff within the app (initially, direct contact beyond approval/disapproval will be outside the app).

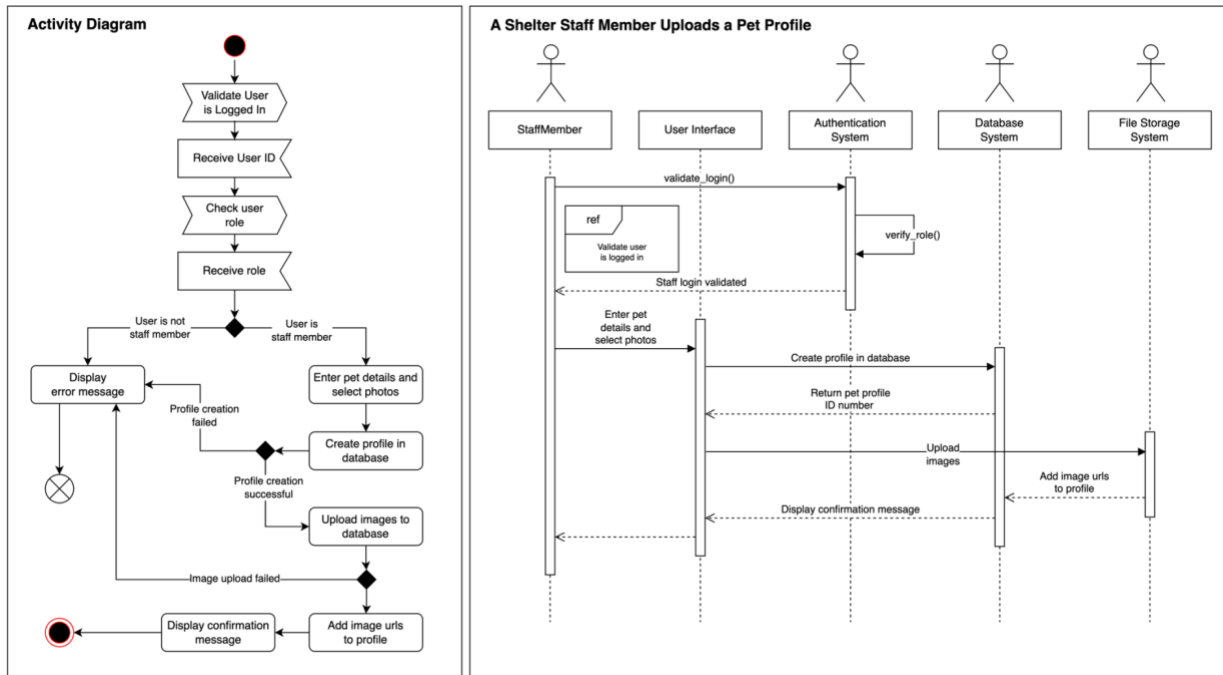
Non-Functional Requirements

Requirement #	Category	Subcategory	Priority	Description
REQ-22	Functionality	Completeness	1 - HIGH	The system should meet all core functionalities for each user role (guests, administrators, shelter staff, prospective pet parents), as specified in this document.
REQ-23	Functionality	Compatibility	2 - MEDIUM	The system's behavior and functionalities should be consistent across all platforms and devices.
REQ-24	Functionality	Security	1 - HIGH	User data, pet information, and system access should be protected with proper authentication, authorization, and encryption measures.
REQ-25	Functionality	Interoperability	4 - OPTIONAL	The system might be able to integrate with other/existing shelter databases or apis where needed.
REQ-26	Usability	User Interface (UI)	2 - MEDIUM	The app interface should be intuitive, user. Friendly, and visually appealing for to all users.
REQ-27	Usability	User Experience (UX)	2 - MEDIUM	The system should be easy to navigate and learn, offering a smooth and enjoyable experience for all users.
REQ-28	Usability	Accessibility	1 - HIGH	The app should be accessible to users with disabilities, complying with accessibility standards.
REQ-29	Usability	Help And Support	3 - LOW	Clear and easily accessible help documentation and support should be available in the app.
REQ-30	Reliability	Availability	1 - HIGH	The system should be highly available with minimal downtime. Users should be able to access the system whenever needed.
REQ-31	Reliability	Fault Tolerance	4 - OPTIONAL	The system should be resilient to errors and failures. There should be mechanisms for automatic recovery and data backups.
REQ-11	Reliability	Performance	1 - HIGH	The app should have fast loading times and responsiveness, avoiding lags or delays that hinder user experience.
REQ-12	Performance	Scalability	2 - MEDIUM	The system should be scalable to accommodate a growing user base and increasing data volume without performance degradation.
REQ-13	Performance	Efficiency	3 - LOW	The system should utilize resources efficiently, avoiding unnecessary CPUor memory usage.

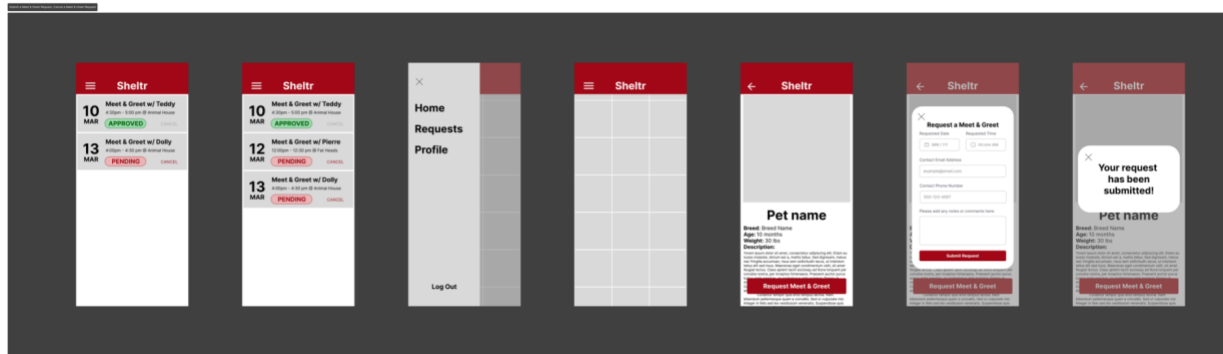
REQ-14	Performance	Response Time	2 - MEDIUM	The app should respond quickly to user actions and requests, meeting responsiveness expectations.
REQ-15	Supportability	Maintainability	3 - LOW	The system code should be well documented, modular, and easy to Understand for future maintenance and updates.
REQ-16	Supportability	Testability	1 - HIGH	The system should be designed with testability in mind. It should allow for thorough testing and debugging.
REQ-17	Supportability	Configurability	3 - LOW	The system should be configurable. The data model should be able to adapt to Specific needs of different shelters and environments.

System Sequence And Activity Diagrams





User Interface Specification



User Action 1 - Request A Meet & Greet:

1. Tap on a picture on the pet grid to navigate to the profile
2. Press the “request a meet & greet” button
3. Enter time, date, contact information, and any notes for the shelter
4. Press the “submit request” button
5. A success message will be displayed, which can be dismissed by pressing the x button or anywhere outside of the modal.

User Effort: MEDIUM

This action requires up **to nine taps** for a logged in user from the home screen (pet grid) to complete, assuming all form fields in the request are completed. There are two dropdown menus (date and time) and three text fields (email, phone, and notes) in this flow.

User Action 2 – Delete A Meet & Greet Request:

1. Tap on the “hamburger” menu icon to access the menu drawer

Matthew Fante

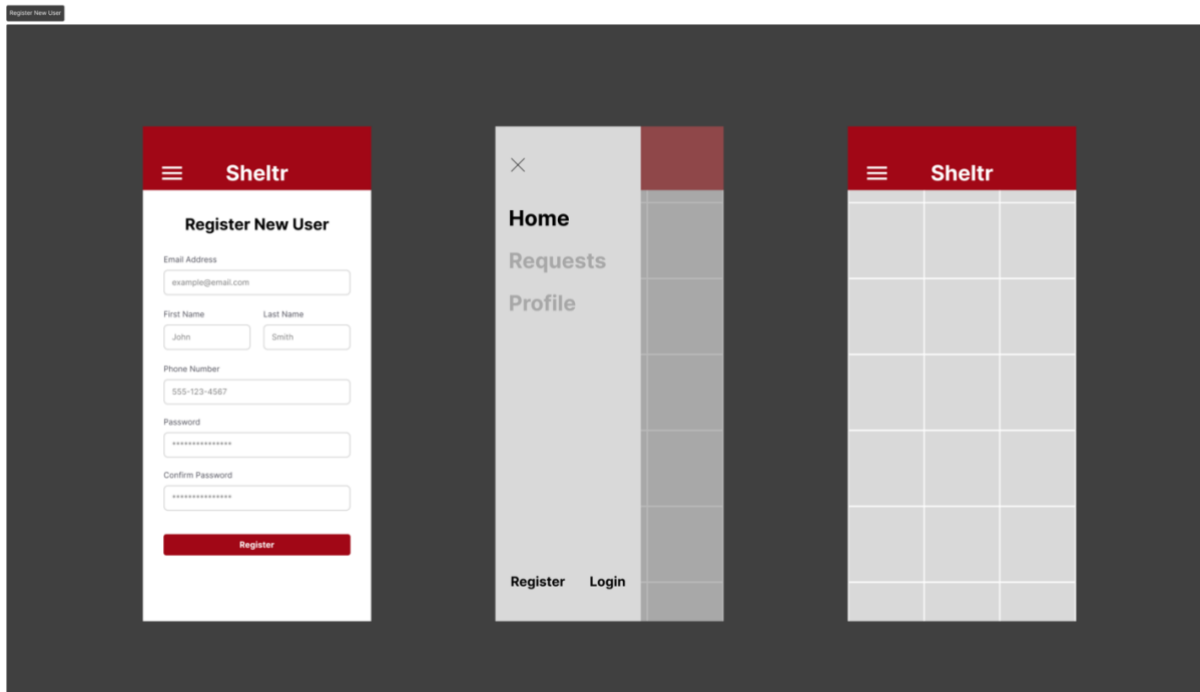
INFO-C451: System Implementation (Spring 2024)

03/24/24

2. Tap on “requests” to view all upcoming meet & greet requests
3. Press the “cancel” button on the corresponding request to cancel

User Effort: LOW

This action requires **three taps** for a logged-in user from the home screen (pet grid) to complete.



User Action 3 – Register A New Meeting:

1. Tap on the “hamburger” menu icon to access the menu drawer
2. Tap on “register” to navigate to the “register new user” form
3. Enter an email address, first name, last name, phone number, and confirmed password 4) press the “register” button
4. A successfully registered user will be routed back to the home screen as a logged-in user

User Effort: HIGH

This action requires **nine taps** for a guest user from the home screen (pet grid) to complete.



View a live interactive demo of the user interface prototype in Figma [here](#)

Traceability Matrix

#	Priority Weight	UC-01	UC-02	UC-03	UC-04	UC-05	UC-06	UC-07	UC-08	UC-09	UC-10	UC-11	UC-12	UC-13	UC-14	UC-15	UC-16	UC-19	UC-20	UC-21	UC-22	UC-23	UC-24	UC-25	UC-26	UC-27	UC-28
REQ-01	5	X																					X				X
REQ-02	3			X	X																		X				X
REQ-03	5		X															X	X	X		X					
REQ-04	1																X										
REQ-05	3										X								X		X		X			X	
REQ-06	3											X							X		X		X			X	
REQ-07	5												X										X				
REQ-08	3													X					X		X			X			
REQ-09	3													X					X		X			X			
REQ-10	1														X				X		X			X			
REQ-11	3																		X		X	X					
REQ-12	5															X			X		X				X	X	
REQ-13	1					X													X		X	X					
REQ-14	3				X		X												X		X			X			
REQ-15	3							X											X		X			X			
REQ-16	1								X										X		X	X					
REQ-17	3									X									X		X	X					
Max PW		5	5	3	3	1	3	3	1	3	3	3	5	3	1	5	1	5	5	5	5	5	5	3	5	5	5
Total PW		5	5	3	6	1	3	3	1	3	3	3	5	6	1	5	1	5	37	5	32	13	19	13	5	11	8

Use Cases

Use Case #	Description
UC-01	Browse adoptable pets: Explore animal profiles without signing up.
UC-02	Register user account: Sign up for an account to access additional features.
UC-03	Search for adoptable pets: Filter and browse animal profiles based on various criteria (species, breed, age, size, sex).
UC-04	View animal profiles: Access photos, descriptions, and additional information about each animal.
UC-05	Save favorite pets: Mark animals of interest for easy reference.
UC-06	Request a meet-and-greet: Initiate a request to meet a specific animal at the shelter.
UC-07	Track request status: View the status of submitted meet-and-greet requests.
UC-08	Update user profile: Edit personal information and pet preferences.
UC-09	Update contact information: Modify phone number or email address for communication.
UC-10	Create animal profiles: Add details about adoptable animals including photos, descriptions, and other details.

UC-11	Manage animal profiles: Edit and update information for each animal.
UC-12	Mark availability: Indicate if an animal is available, adopted, or unavailable for adoption.
UC-13	Manage meet-and-greet requests: Review, approve or decline requests from potential adopters.
UC-14	View upcoming meet-and-greets: View upcoming scheduled meet-and-greets.
UC-15	Update shelter information: Edit contact details, hours of operation, and adoption procedures.
UC-16	Review and approve new shelter staff accounts: Verify information and grant access to the platform.
UC-17	Manage user accounts: Reset passwords and address user concerns.
UC-18	Remove inappropriate content: Ensure the platform maintains a safe and appropriate environment.
UC-19	User registration: Create a new user account using email address and password.
UC-20	User login: Authenticate users with email and password.
UC-21	Password reset: Initiate self-service password reset process
UC-22	Validate login: Maintain user sessions and authentication tokens.
UC-23	Store user data: Securely store user information (profiles, contact info, preferences, etc.) in the database.
UC-24	Store animal data: Store detailed information about each adoptable animal in the database.
UC-25	Store meet-and-greet requests: Manage requests from potential adopters, including details and status.
UC-26	Store shelter data: Maintain information about each participating shelter (contact info, procedures, etc.).
UC-27	Store animal photos: Upload and store high-quality images of adoptable animals.
UC-28	Serve images: Deliver animal photos efficiently to users within the app.

System Requirements

Requirement #	Priority	Description
REQ-01	5	All Users Should Be Able To Browse Animal Profiles Based On Various Filters Like Breed, Size, Age, Energy Level, Etc.

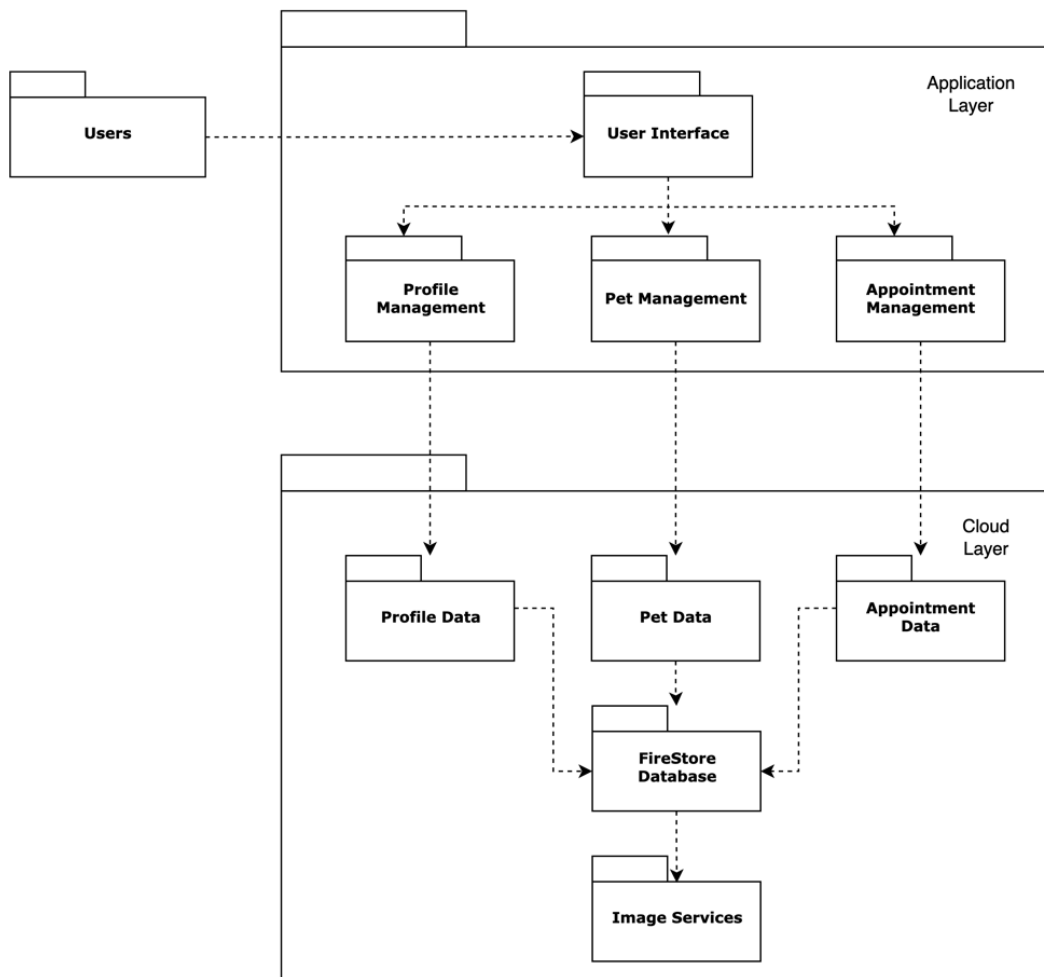
REQ-02	3	All Users Should Be Able To View Detailed Animal Profiles Featuring Photos, Personality Descriptions, And Additional Information.
REQ-03	5	All Users Should Be Able To Create An Account With A Valid Email Address.
REQ-04	1	Shelter Accounts Should Undergo An Approval Process Before Posting Adoption Profiles.
REQ-05	3	Shelter Staff Members Should Be Able To Create Detailed Profiles For Each Animal, Including Photos, Personality Descriptions, Adoptability Criteria, And Vet Records.
REQ-06	3	Shelter Staff Members Should Be Able To Edit And Update Animal Profiles As Needed.
REQ-07	5	Shelter Staff Members Should Be Able To Mark Animals As Available Or Adopted.
REQ-08	3	Shelter Staff Members Should Be Able To View And Process Meet-And-Greet Requests From Potential Adopters.
REQ-09	3	Shelter Staff Members Should Be Able To Approve Or Decline Meet-And-Greet Requests.
REQ-10	1	Shelter Staff Members Should Be Able To View A Schedule Of Upcoming Meet-And-Greet Appointments Within A Built-In Calendar.
REQ-11	3	Shelter Staff Members Should Be Able To View Requester Contact Information (Phone Number Or Email Address).
REQ-12	5	Shelter Staff Members Should Be Able To Update Shelter Information, Including Contact Details, Hours Of Operation, And Adoption Procedures.
REQ-13	1	Prospective Pet Parents Should Be Able To Mark Favorite Animals For Easy Reference.
REQ-14	3	Prospective Pet Parents Should Be Able To Submit Meet-Andgreet Requests Directly From Animal Profiles With Preferred Dates And Times.
REQ-15	3	Prospective Pet Parents Should Be Able To Track The Status Of Submitted Requests And Receive Notifications Of Approvals Or Declines.
REQ-16	1	Prospective Pet Parents Should Be Able To Create And Manage User Profiles, Including Basic Information And Adoption Preferences.
REQ-17	3	Prospective Pet Parents Should Be Able To Update Contact Information And Notification Settings.

System Architecture & Design

Architectural Styles

Sheltr is a cross-platform mobile app, built using the Flutter framework and the Dart programming language. The system uses an event-driven, asynchronous architecture. The app's Flutter user interface is designed using a component/nested widget-based structure.

Identifying Subsystems



Mapping Subsystems to Hardware

- End users will be able to install the app on a variety of client devices (described under Hardware Requirements), while all backend services are managed in the cloud via Google FireBase services.
- The FireBase Console can be accessed from any modern web browser.

Persistent Data Storage

- The system will use the following persistent data objects:
 - o User
 - o Pet
 - o Profile
 - o Appointment (Meet & Greet Request)
- Data will be stored in Firestore Database as NoSQL tables.

Network Protocol

All backend services are managed in the cloud via Google Firebase. Google Firebase uses the REST and RPC APIs for network transport.

Global Control Flow

Execution Order

The app uses an event-driven model that allows multiple users to seamlessly interact with the system in an asynchronous fashion.

Time Dependency

All system actions happen in real-time. No batch processing is required.

Concurrency

N/A

Hardware Requirements

This app supports most modern mobile devices. All iPhones that support iOS 12 or higher are compatible. [Android hardware requirements are as follows:](#)

Requirement	Minimum	Recommended
x86_64 CPU Cores	4	8
Memory in GB	8	16
Display resolution in pixels	WXGA (1366 x 768)	FHD (1920 x 1080)

User Interface Design and Implementation

The general layout and structure of my UI prototype from Module 7 is fairly close to the system as it's been implemented, however there are two main UI components I have had to develop to meet the needs of my system:

- The approval interface for Meet & Greet requests
- The approval interface for users requesting their accounts be converted to shelter accounts (allowing them to add new adoptable pets to the feed)

Matthew Fante
INFO-C451: System Implementation (Spring 2024)
03/16/24

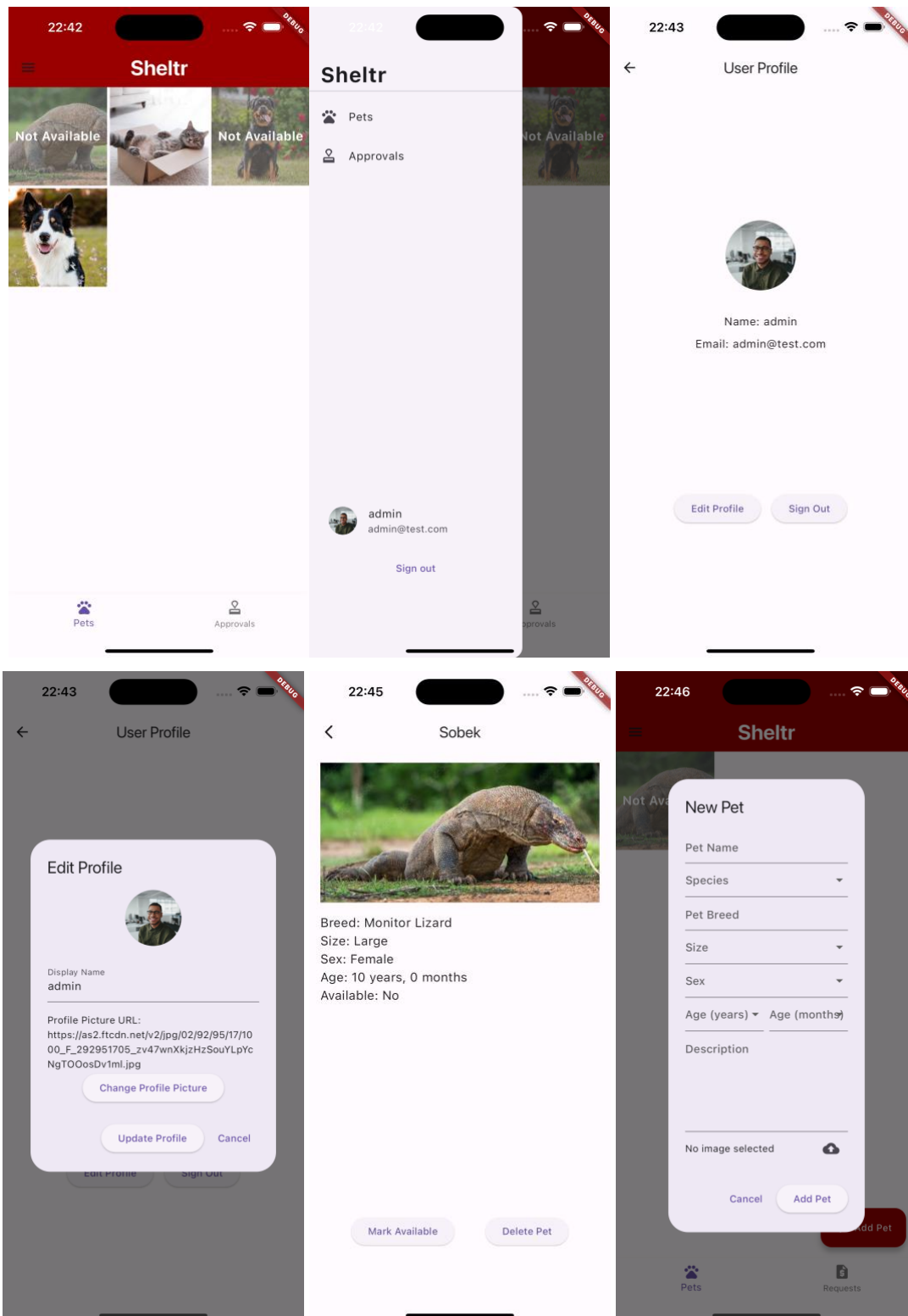
Initially, I envisioned one 'requests' interface that would perform different duties based on the user's role. In implementation, I created separate app routes to display users' Meet & Greet lists, shelters' Requests lists, and the admins' Approval lists, then conditionally give access to only the routes relevant to the logged in user's role.

The end result produces the effect I envisioned: I wanted to avoid there being a lot of unusable interface objects since users of different role types need access to different sections of the app.

On the next page, I've collected screenshots of several of the use cases I've implemented. I continue to refine the UI of the app to get closer to my prototype, as my main priority has been building logic and implementation.

Matthew Fante
INFO-C451: System Implementation (Spring 2024)
03/16/24

Screenshots



Design of Tests

Unit testing will be completed in an emulated iOS environment prior to final submission of the code repository. Test accounts for user, shelter, and admin roles are already created and have been used throughout the course of development. The following list is a revised set of unit tests based on the functional requirements planning assignment to ensure all use cases are tested.

General Tests

- Initialization and Navigation
 - Test app initialization and default route.
 - Test navigation to different screens (e.g., profile page, pets list).
- Error Handling
 - Test error scenarios (e.g., network failure, authentication failure).
 - Validate that appropriate error messages are displayed to the user.

Guest User Tests

- Browse Adoptable Pets
 - Test if guest users can view the list of adoptable pets.
 - Test if guest users can filter the list based on various criteria (e.g., species, breed).
- View Animal Profiles
 - Test if guest users can access individual pet profiles.
 - Test if pet profiles display correct information (e.g., photos, descriptions).
- Register User Account
 - Test user registration process with valid and invalid inputs.
 - Validate error handling for duplicate emails, invalid passwords, etc.

Prospective Pet Parent Tests

- Save Favorite Pets
 - Test if users can add pets to a "favorites" list.
 - Test if the favorites list persists across sessions.
- Request a Meet-and-Greet
 - Test if users can request a meet-and-greet with a pet.
 - Validate error handling for invalid requests or when a pet is unavailable.
- Track Request Status
 - Test if users can view the status of meet-and-greet requests.
 - Test different request statuses (e.g., pending, approved, declined).
- Update User Profile
 - Test if users can update their personal information.
 - Validate that changes are reflected correctly in the database.
- Update Contact Information

- ~~○ Test if users can change their contact information (e.g., phone, email).~~
- ~~○ Test validation of contact information (e.g., correct email format).~~

Animal Shelter Staff Tests

- ~~● Create Animal Profiles~~
 - ~~○ Test if staff can create new animal profiles.~~
 - ~~○ Test mandatory fields and error handling for invalid data.~~
- ~~● Manage Animal Profiles~~
 - ~~○ Test if staff can update existing animal profiles.~~
 - ~~○ Test if changes are reflected in the user interface.~~
- ~~● Mark Availability~~
 - ~~○ Test if staff can change the availability status of a pet.~~
 - ~~○ Test how the change affects the pet's visibility to users.~~
- Manage Meet-and-Greet Requests
 - Test if staff can approve or decline meet-and-greet requests.
 - Test if users are notified of request status changes.
- View Upcoming Meet-and-Greets
 - Test if staff can view scheduled meet-and-greets.
 - Validate the accuracy of the meet-and-greet schedule.
- ~~● Update Shelter Information~~
 - ~~○ Test if staff can update shelter details (e.g., contact info, hours).~~
 - ~~○ Validate that changes are correctly stored in the database.~~

System Administrator Tests

- ~~● Review and Approve New Staff Accounts~~
 - ~~○ Test if admins can approve or deny new shelter staff accounts.~~
 - ~~○ Test error handling for invalid approval attempts.~~
- ~~● Manage User Accounts~~
 - ~~○ Test if admins can reset passwords or address user concerns.~~
 - ~~○ Validate correct error handling and successful account management.~~
- ~~● Remove Inappropriate Content~~
 - ~~○ Test if admins can identify and remove inappropriate content.~~
 - ~~○ Test validation to ensure only admins have this access.~~

Authentication System Tests

- ~~● User Registration~~
 - ~~○ Test user registration with various inputs (valid/invalid emails, passwords).~~
 - ~~○ Test edge cases like re-registration with the same email.~~
- ~~● User Login~~
 - ~~○ Test user authentication with correct and incorrect credentials.~~
 - ~~○ Validate error handling and session management.~~
- ~~● Password Reset~~

- ~~Test password reset process with valid and invalid emails.~~
 - ~~Validate that password reset emails are sent and processed correctly.~~
- ~~Validate Login~~
 - ~~Test session token validity and expiration.~~
 - ~~Test that users are redirected to the login page upon session expiration.~~

Database System Tests

- ~~Store User Data~~
 - ~~Test if user information is correctly stored in the database.~~
 - ~~Validate database schema constraints (e.g., unique keys, required fields).~~
- ~~Store Animal Data~~
 - ~~Test if animal profiles are stored correctly.~~
 - ~~Test database constraints and relationships among data.~~
- ~~Store Meet-and-Greet Requests~~
 - ~~Test if meet-and-greet requests are stored with correct data.~~
 - ~~Validate integrity and relationships in the database.~~
- ~~Store Shelter Data~~
 - ~~Test if shelter information is correctly stored and retrieved.~~
 - ~~Validate accuracy and integrity of stored data.~~

File (Image) Storage System Tests

- ~~Store Animal Photos~~
 - ~~Test image upload with various image types and sizes.~~
 - ~~Validate error handling for failed uploads and invalid images.~~
- ~~Serve Images~~
 - ~~Test if stored images are served quickly and accurately.~~
 - ~~Validate image caching and error handling for missing images.~~

Reference

Glossary Of Terms

- **User:** anyone using the sheltr app, including guests, prospective pet parents, shelter staff, and system administrators.
- **Guest:** unregistered user browsing animal profiles without an account.
- **Prospective pet parent:** a registered user interested in adopting an animal through sheltr.
- **Shelter staff:** authorized users representing an animal shelter on the sheltr.
- **System administrator:** user with ability to manage user accounts, data, and settings.

Matthew Fante

INFO-C451: System Implementation (Spring 2024)

03/16/24

- **Animal profile:** a detailed page showcasing information about an adoptable animal.
- **Meet-and-greet:** an arranged meeting between a prospective pet parent and a shelter representative regarding a specific animal.

Project Plan

Week #	Milestone	Complete?
3	Design class diagrams and begin drafting UI prototype in Figma	✓
4	Integrate google cloud services into project, validate authentication and data i/o services are successfully deployed	✓
5	Build initial menu and page scaffolding for app, connect backend providers to each contextual view (ensure authentication session carries over from one context to another)	✓
6		
7	Design widgets for pet profiles, parent profiles, implement “add new pet” and “edit pet” functionality, validate image storage is functioning as expected	✓
8		
9	Implement search and filter functionality for adoptable pets	✗
10		
11	Design widgets for and implement “meet and greet request”, “upcoming appointments”, and “meet and greet approval” screens	✓
12		
13	Implement user experience segmentation. Restrict functionalities available to the user based on their role as guest, shelter, adopter, or system administrator.	✓
14	Leave this time open to allow for any delays, potentially implement any of the optional features listed in the system requirements section	✓
15	Perform final testing, capture validation screenshots, assemble final report	✓
16	Present application and final report	✓

Note: Search and filter functionality has been delayed from initial release.