SOFT4902

Graduation Design Project

FEYZİYE SCHOOLS FOUNDATION

IŞIK UNIVERSITY

Project Title

PET ADOPTION WEBSITE

By SELİN KAROL/20SOFT1023, TAYLAN TANYERİ/19SOFT1010

REQUIREMENTS ANALYSIS DOCUMENT

1. Introduction

The purpose of this section is to provide a brief overview of the project and the reasons for its development, its scope, and references.

1.1. Purpose of the System

This project is being developed in order to underline the importance of the animals that are ill or disabled. To ensure animal welfare, we want to develop a platform so that people can get to know these pets better. We want to draw more attention to these pets because it is very difficult or impossible for them to meet their own needs. We wanted to create a separate platform to raise awareness for them. We specifically focus on these pets, as they have a less chance of being adopted and reaching their forever homes. As a result of that, we designed a special platform that people can learn, donate & adopt to make an impact.

1.1.1 Responsibilities

- Providing a reliable platform for adopters and pets,
- Securing user data and preservation of information,
- Enabling verified pets to be shared,
- Ensuring that only verified users can perform some tasks including adoption,

- Creating a profile for users that takes into account the special needs of disabled or sick pets,
- Enabling users to share pets in a way that other users can see,
- Providing appropriate adoption conditions, taking care of pets and accessibility of correct information regarding the medical history of pets and potential user' eligibility,
- Interacting with potential adoption families and monitoring the adoption process,
- Raising awareness in society.

1.2. Scope of the System

Our project introduces a website platform designed to facilitate the adoption of disabled or sick pets. This platform aims to facilitate the process of animal shelters or individual users to confide in trustworthy and willing people that will take care of the pets in need, and to increase the chances of animal-loving individuals to offer these special animals a loving home. Also, we invite those who wish to support our pets with donations, bringing public awareness by promoting responsible pet ownership.

1.3. Objectives and Success Criteria of the Project

Finishing the project, the project must achieve the criteria listed belove to be classified as successful:

- Accurate Matches: The platform must accurately match disabled or sick pets with suitable adoption families with special needs. These matches should take into account the needs and circumstances of both the animal and the adopting family.
- Adoption Process: The adoption process should include a process that evaluates the suitability of the candidate adoption family. Before adoption, it must be ensured that the animal has a home that suits its quality of life and needs.
- Functional Website: The website should be responsive and user-friendly, reaching the UX goals with an interface that is easy to learn, use and navigate.
- User Experience and Accessibility: Users should be able to easily access and use the website, enter their personal status and view animal profiles. The interface must be user-friendly and comply with accessibility standards.
- Security and Privacy: User information must be securely protected, processed in accordance with privacy policies, and user security must be ensured. Legal contracts, terms and services licenses must be required when signing up.
- Social Participation and Awareness: The platform should increase awareness in society about disabled or sick pets, raise users' awareness on this issue and create solidarity among animal

lovers. The pages such as About Us and Donations will direct users to related links and pages so that people can learn about the stories of some pets and support them financially.

1.4. Definitions, Acronyms, and Abbreviations

- ✓ Admin: The user who is responsible for the overall management of the system or platform. Admin usually has the authority to control the system, manage users, and regulate access rights of other users.
- ✓ Adoption Application: The formal request submitted by a user to adopt a specific pet.
- ✓ Adoption Centers: These places, also known as animal shelters or adoption centers, aim to find a temporary or permanent home for abandoned or stray animals. Adoption centers take animals into care and work to adopt them to suitable families.
- ✓ Caregiver/Caretaker: Person who takes care of a disabled pet that is up for adoption.
- ✓ Pet Adoption Website (PAW): A platform designed to facilitate the adoption of disabled/ill pets, promote pet adoption and encourage people to support these pets financially and emotionally, and raise public awareness.
- ✓ Pets: Animals that are kept, cared for and loved by humans. Animals such as cats, dogs, birds, with special conditions are considered pets.
- ✓ Potential User: The name given to people who are interested in using the system or platform but are not yet registered. Potential users can visit the system for trial purposes.
- ✓ Registered User: The name given to users who have completed their registration to the system and can log in. These users can log in to the system and perform certain operations. The name given to people who are interested in using the system or platform but are not yet registered. Potential users can visit the system for trial purposes.
- ✓ Questionnaire: A form of questions regarding people's personal lives, interests, disabilities, pet preferences. The questionnaire aims at finding pets from the pet pool and making it easy to access suitable pets from various options.
- ✓ Verification: Admin's approval of a registered user, stating that the user may proceed with adoption applications or upload pets to be adopted to the system.
- ✓ Verified User: The name given to users who have gone through the authentication process and whose real identity has been verified. Such users can often perform security-sensitive operations.

1.5. Overview

The comprehensive guidelines for the new movie promotion system are provided in the remainder of this publication. It is set up as follows:

- Section 2: In this section, we define existing animal adoption platforms. The section is about the current landscape predominantly catering to general users, lacking specific focus on sick or disabled animals. This limitation makes it hard for special animals to find suitable homes, leading to an inherent problem in the current system. Reliability issues persist for users interested in adopting sick or disabled animals, as the presence of fake profiles decreases trust within the platform.
- Section 3: We provide detailed Use Cases and Scenarios to illustrate user interactions, ensuring a user-friendly experience. Additionally, we specify "Non-Functional Requirements" to set quality standards. "The Use Case Model" and "Object Model" visually represent system interactions and structures, helping in understanding data organization. We present a "Dynamic Model" showcasing system behavior over time. "Detailed User Interface Designs" are included for intuitive navigation.
- Section 4: This section determines various additional analysis components critical to understanding the project. It may competitor analysis, risk assessment, Each element is carefully examined, providing valuable insights for decision-making processes.
- Section 5: Glossary explains terms and jargon used within the project. It provides clear definitions for these terms, ensuring a common understanding among all stakeholders involved in the project.
- Section 6: References section includes a list of all the sources, materials, and documents referred to during the analysis and planning phases of the project.

2. Current System

Existing animal adoption sites are generally geared towards general users and do not specifically address sick or disabled animals. This can make it difficult for special animals to find their families. In the current system, there may be a reliability problem for users who want to adopt sick or disabled animals. Fake users and animals can mislead real people in need, negatively impacting efforts to build a trustworthy platform. Competing systems generally do not specifically address sick or disabled animals. This is one of the shortcomings of competing sites, because these animals also need loving families.

3. Proposed System

The suggested technology proposes a system to match pets with possible adopters via identifying adopters, presenting pet information, raising awareness of donations to adoption shelters and to improve animal welfare. Our project aims at closing this gap by creating a platform that includes animals with special needs. Our new system aims at providing a reliable and transparent platform for users who want to adopt sick or disabled animals by including their medical history reports and needs. While we block fake users through our user verification processes, we aim to set an example for the society by showing that adopted animals live happy lives through regular updates. This approach emphasizes our sensitivity to animals with special needs and our social contribution.

3.1. Overview

The project connects potential adopters and their future companions based on compatibility traits by providing a user-friendly platform and special features to reach this goal. It includes different features for different user types. To ensure responsible and reliable pet ownership, the system has some constraints on visitors. While everyone can see the listed pets on the system, take the compatibility questionnaire and view the educational and fundraising links, registered users may view the locations of the pets, their medical history and special care guide. By creating user-profiles, users become members of the website adoption community and start saving pets they have an interest in. They may start their adoption application or register pets if they have received a verification from administration. They may also get notified to keep track of their ongoing adoption or informed when someone requests to start an adoption process with the animal they have listed. The admins on the other hand, may add/delete listed pets and user-profiles to ensure safety and preservation of data. Other than that, there is a pet database and a user database to save the information of the users and pets to be adopted, updated regularly to ensure maintainability and access to correct information to prevent complexities. The personal data of users will be protected by legal actions and contracts. A security team will be managing the preservation of data to protect rights of people and animals by using such security measures. Continuous monitoring of the performance and interactions is necessary to improve user experience.

3.2. Requirements

High-level functionality of the system focuses on providing a user-friendly platform which is interactive and informative for individuals so that they can engage with other users, contribute to the aim of exploring listed pets and finding a connection. The functional requirements of such, can be listed but not limited to as:

1. Forming user profiles with personal information,
2. Logging in to & out of accounts,
3. Taking the matching algorithm questions,
4. Matching with pets according to the results,
5. Seeing listed pets,
6. Viewing pet information,
7. Viewing nearby pet locations on map for registered users,

8. Favoring listed pets for registered users,
9. Adding new pets for registered and verified users,
10. Deleting their own pets for registered and verified users,
11. Editing their own pets for registered and verified users,
12. Starting adoption process for registered and verified users,
13. Getting notifications if a user starts adoption process,

3.2.1. Use Cases

Table 1. Register User

Use Case Name:	Register User
Actor:	Visitor
Summary:	To keep the data of a person, a new user is registered with a username and password.
Basic Flow:	1. It starts with register order. 2. The system requests a username and password. 3. The user inputs the necessary data. 4. The system determines whether the username is already in use. 5. The email address and name are requested by the system. 6. The user inputs the necessary data. 7. All of this data is stored in the system. 8. A login session is initiated by the system.
Alternative Flows:	Step 4: The use case returns to step 2 and the system displays a notification if the username is identical to an already-existing username. Step 5: The use case stays in step 5 and a notification is shown if the user doesn't enter a needed field.

Entry Conditions:	None
Exit Conditions:	The user's account is created, and the user is logged in. The database of the website contains the user's profile data.
Priority	Low

Table 2. Login User

Use Case Name:	Login User
Actor:	Registered User
Summary:	A user logins account.
Basic Flow:	1. User opens the website. 2. User clicks on login. 3. User enters username and password. 4. User logs in.
Alternative Flows:	Step 3: The use case stays in step 3 if user enters wrong username or password. User must either register or provide correct info.
Entry Conditions:	User has an account.

Exit Conditions:	User logs in. User is requested to try again.
Priority	Low

Table 3. View Pets

Use Case Name:	View Pets
Actor:	Users and visitors
Summary:	Users and visitors see pet list.
Basic Flow:	1. User or visitor opens the website. 2. User logs in. 3. User clicks on pet ads. 4. User sees pet lists and their names. 5. User may favor or unfavor a pet by clicking the heart button.
Alternative Flows:	Step 2: Visitor cannot login. Visitor clicks on pet ads and sees pet lists and names. Visitor cannot favor or unfavor a pet. Registration is required. Step 4: User may filter pets that he/she wishes to look for. Then user sees pet lists that are filtered.
Entry Conditions:	For user, user has an account. For visitor, none.
Exit Conditions:	User sees pets and favors/unfavors them. Visitor sees pets.

Priority	High

Table 4. Site Navigation

Use Case Name:	Site Navigation
Actor:	User or visitor
Summary:	User or visitor clicks on Donation or About Us sections.
Basic Flow:	1. User or visitor opens the website. 2. User or visitor clicks on Donation on the navigation bar. 3. Donation page opens. 4. User or visitor sees donation links and related information.
Alternative Flows:	Step 2: User or visitor may click on About Us on the navigation bar. About Us page opens. User or visitor sees related information.
Entry Conditions:	None
Exit Conditions:	User or visitor sees Donation page and/or About Us page.
Priority	Low

Table 5. Take Questionnaire

Use Case Name:	Take Questionnaire

Actor:	User or visitor
Summary:	User/Visitor takes a questionnaire to see the eligible pets for himself/herself.
Basic Flow:	1. User/Visitor opens the website. 2. User/Visitor clicks on the questionnaire. 3. Questionnaire opens. 4. User/Visitor answers the questionnaire. 5. Questionnaire list some eligible pets based on user answers. 6. User may favor the pets. Visitor cannot.
Alternative Flows:	Step 4: User/Visitor stops the questionnaire mid-way. The answers are lost. Step 6: Visitor tries to favor pets. An error message occurs: This feature requires registration.
Entry Conditions:	For user, login. For visitor, none.
Exit Conditions:	User/Visitor sees eligible pets. User may favor.
Priority	High

Table 6. Add Pet

Use Case Name:	Add Pet
Actor:	User

Summary:	User adds a pet advertisement.
Basic Flow:	1. User opens the website. 2. User clicks on add pet button. 3. User fills in required information and adds pet pictures. 4. User submits the advertisement. 5. Advertisement takes place.
Alternative Flows:	Step 4: User stops the advertisement mid-way. The information is lost.
Entry Conditions:	User is logged in and verified by admin.
Exit Conditions:	System displays the pet in the pet list.
Priority	High

Table 7. Remove/Edit Pet

Use Case Name:	Remove/Edit Pet
Actor:	User
Summary:	User removes/edits his/her pet advertisement.

Basic Flow:	1. User opens the website. 2. User clicks on edit pet button. 3. User changes information. 4. User submits the advertisement. 5. Advertisement is changed.
Alternative Flows:	Step 2: User clicks on delete button. Pet is deleted. Step 3: User stops the change mid-way. The information is lost.
Entry Conditions:	User is logged in and verified by admin.
Exit Conditions:	System displays the pet in the pet list or removes the pet.
Priority	High

Table 8. Edit User Profile

Use Case Name:	Edit User Profile
Actor:	User
Summary:	User edits/deletes his/her profile.
Basic Flow:	1. User opens the website. 2. User clicks on edit profile button. 3. User changes information. 4. User submits the changes.

	5. Profile is changed.
Alternative Flows:	Step 2: User clicks on delete button. Profile is deleted. Step 3: User stops the change mid-way. The information is lost.
Entry Conditions:	User is logged in.
Exit Conditions:	System saves changed information or removes the user.
Priority	Medium

Table 9. See User Profile

Use Case Name:	See User Profile
Actor:	User/Visitor
Summary:	User/Visitor views a user profile by clicking.
Basic Flow:	1. User/Visitor opens the website. 2. User/Visitor clicks on a user profile. 3. Profile is displayed by the system.
Alternative Flows:	None
Entry Conditions:	None

Exit Conditions:	User/Visitor sees the profile.
Priority	Medium

Table 10. Start Adoption Process

Use Case Name:	Start Adoption Process
Actor:	User
Summary:	User starts adoption process by sending a request to the pet's caregiver.
Basic Flow:	1. User opens the website. 2. User clicks on a pet. 3. Pet page opens. 4. User sends request. 5. A request reaches the caregiver. 6. Caregiver contacts user and proceeds with the process.
Alternative Flows:	Step 4: User cancels request. Step 5: Caregiver declines request.
Entry Conditions:	User is logged in and verified by admin. Caregiver is verified by the admin.
Exit Conditions:	Notification sent to caregiver.
Priority	High

Table 11. Admin Login

Use Case Name:	Admin Login
Actor:	Admin
Summary:	Admin logs in the system.
Basic Flow:	1. Admin opens the website. 2. Admin clicks on login. 3. Admin types down the admin username and password. 4. System directs admin to admin's panel.
Alternative Flows:	Step 3: Admin types down the wrong username or password. System requires to retype.
Entry Conditions:	Admin has the admin username and password.
Exit Conditions:	Admin is directed to admin's panel.
Priority	Medium

Table 12. Admin's Verification

Use Case Name:	Admin's Verification
Actor:	Admin
Summary:	Admin verifies or deletes users.

Basic Flow:	1. Admin opens the website. 2. Admin checks new users and their profile information. 3. Admin verifies users. 4. System sends a notification to verified users.
Alternative Flows:	Step 3: Admin does not verify users. System sends a notification why user is not approved. Step 3: Admin deletes user because of irrelevant information. User profile is deleted.
Entry Conditions:	Admin logs in. There are non-verified users.
Exit Conditions:	Admin does user verification.
Priority	High

Table 13. Admin Pet Deletion

Use Case Name:	Admin Pet Deletion
Actor:	Admin
Summary:	Admin deletes pets.
Basic Flow:	1. Admin opens the website. 2. Admin checks pet profile information. 3. Admin deletes pet. 4. System sends a notification to caregiver.

Alternative Flows:	None
Entry Conditions:	Admin logs in. There are pet profiles.
Exit Conditions:	Admin deletes pet.
Priority	High

3.2.2. Scenarios

1. A woman is looking for a cat. She prefers to have a disabled one because she thinks those cats do not have good survival skills to live outside. Therefore, she wants one in order to provide a safe space for it. She does not have the time to take the cat to a vet regularly, so she prefers to have a cat without a chronical illness, but rather a physical disability, like blindness. She filters the search option and find cats. She starts adoption process for one by sending a notification to caretakers.

2. A veterinarian has clients that have ill pets. She thinks she is eligible for taking care of one of the pets, as her job is the treatment of pets. As a result, she specifically searches for a pet that is chronically ill. She downloads some documents of different pets and decides on which one she wants to adopt. As she is a verified user already, she starts adoption. But she realizes she started for the wrong pet, so she cancels.

3. There is an old, retired man who is missing a leg. He has seen some cats or dogs that have the same situation with him. He wants to feel connected to an animal like that. So, he wants to have a playful cat or dog in order to show they can both have fun and play with each other no matter what their physical disability is. He takes the questionnaire based on these opinions and sees his options. He favors some.

4. A girl takes a questionnaire to see pets that may be suitable for her. After taking the questionnaire, she specifically favors one pet that she liked, after checking its medical record and photographs. She wishes to have him, so she sign-ups for the website and provides information to become a verified user.

5. After a while, a family of a pet with a special situation becomes unable to care for their pet due to various reasons, such as economic conditions or time, and they log in to our website to find a better

home for their pet. After completing the necessary steps and giving information, they wait for potential adopters.

6. A boy, who is looking for a suitable family for his pet, rejects a family by thinking that they are not suitable after contacting the family and pursues it with other candidates.

7. Admin checks newly registered users and sees one with proper information and necessary documentation, so he/she verifies the user by sending a verification message.

8. Admin receives a request from a user which is about getting verified. After checking the information, the identification number does not match with user's first name and last name. This is a security problem and Admin is allowed to deny this user. After rejecting the request user gets a notification about this process and is warned if he/she types the correct identification number

9. A woman successfully gives her disabled parrot up for adoption. After the process she deletes the pet profile, admin checks it later.

10. There is a caretaker in a facility that is responsible for putting pets up for adoption in various sites. He realizes that he made a mistake, and he mixed the reports of 2 pets. He goes to the pet page, clicks edit button and correct himself by uploading the correct reports.

3.3. Nonfunctional Requirements

- Usability

  The website should have a user-friendly interface. Elements such as menus, buttons, and forms should be quickly understandable and usable by users.

  The website should be in different languages. For reaching out to other potential users.

  An effective search function should be provided to allow users to quickly find the content they want.

- Reliability

  Quick response to requests made to the website improves user experience. This can be achieved by optimizing server response times.

  User data collected on the website must be stored securely and necessary privacy policies must be applied.

  Errors occurring on the website (for example, 404 errors) should be directed in a user-friendly manner. Additionally, the source of these errors should be identified and corrected.

  It matters how long the website is online. Ideally, the website should be available 24/7 without interruption.

- Performance

Large size images or media files can negatively impact page load times. It is important to compress and optimize these files.

It is important how fast the pages of the website or application load. Fast loading can make users less likely to abandon the site.

Network resources used on the website (for example, CSS and JavaScript files) should be minimized. This can shorten page load times by reducing data transfer.

Page conversion rates show how long users spend on the website and which pages they return to. This data can help determine which pages need focus to improve performance.

Monitoring and correcting errors that occur on the website can improve performance. When the source of errors is identified and resolved, the site becomes more stable.

- Supportability

Detailed and up-to-date documentation regarding the product or service should be provided. Using this documentation, users can solve their own problems or customize the product to suit their needs.

A support system should be provided where users can report problems they experience and receive technical assistance. Communication such as email, phone, live chat will be provided.

- Implementation

The platform should use related tools and technologies in terms of web development. Such tools and technologies may be classified as programming languages as Python and Flask framework for backend, SQLite for data integrity and database formation; JavaScript and HTML tags & React for frontend, CSS and Bootstrap for UI and design & formatting; Visual Studio Code IDE for implementing the components.

Use of version control systems such as Github repositories are also encouraged. Bug tracking, error handling and testing are some of the essential functions while developing the platform.

There may be a certain budget, time and human resource constraints to implement the project. These restrictions are important for the project to be completed on time and within the specified budget.

- External Interface

Well-defined APIs must interact with interfaces, users, services and databases. They must be secured in case of cyber-attacks and data preservation. Unauthorized access will be prohibited via authentication mechanisms.

System should support data exchange formats to ensure data sharing between website components.

- Packaging

The practice of using modules in Python and such tools help us organize the packages based on functionality. This improves code organization and maintaining the code.

- Legal

The website must follow rules and take legal action to prevent some activities. The regulations when it comes to data protection and privacy must be applied for safety. Copyrights and pet rights must be issued. A contract securing the adopters' and adoptees' rights must be accepted in terms and conditions via all parties. The liabilities must be defined.

- Other constraints

Our site aims to raise awareness in society and ensure that disabled or ill pets find permanent, loving homes that can improve their lives. In this way, we aim to both improve the quality of life of pets and make a positive contribution to the lives of adopting families. We are required to comply with certain standards of social responsibility in order to respect the rights of employees, potential adopters, pets and follow fair trade practices and create a positive impact on society.

By this project, people who have disabilities will have the chance to meet pets that have a similar experience within life. Disabled pets will also meet people who are willing to take care of themselves, which decreases the rate of retraction when it comes to adopting, as people acknowledge the illness or the difficulty a pet is facing beforehand.

The system must adhere to regulatory constraints related to the welfare of animals. These may include compliance with animal protection laws, ethical treatment of animals.

3.4. System Models

The process of creating abstract representations of a system, each model offering a unique viewpoint or perspective, is known as system modeling, which are usually use case models and class diagrams. In this section, object model and dynamic model are introduced.

3.4.1. Use Case Model

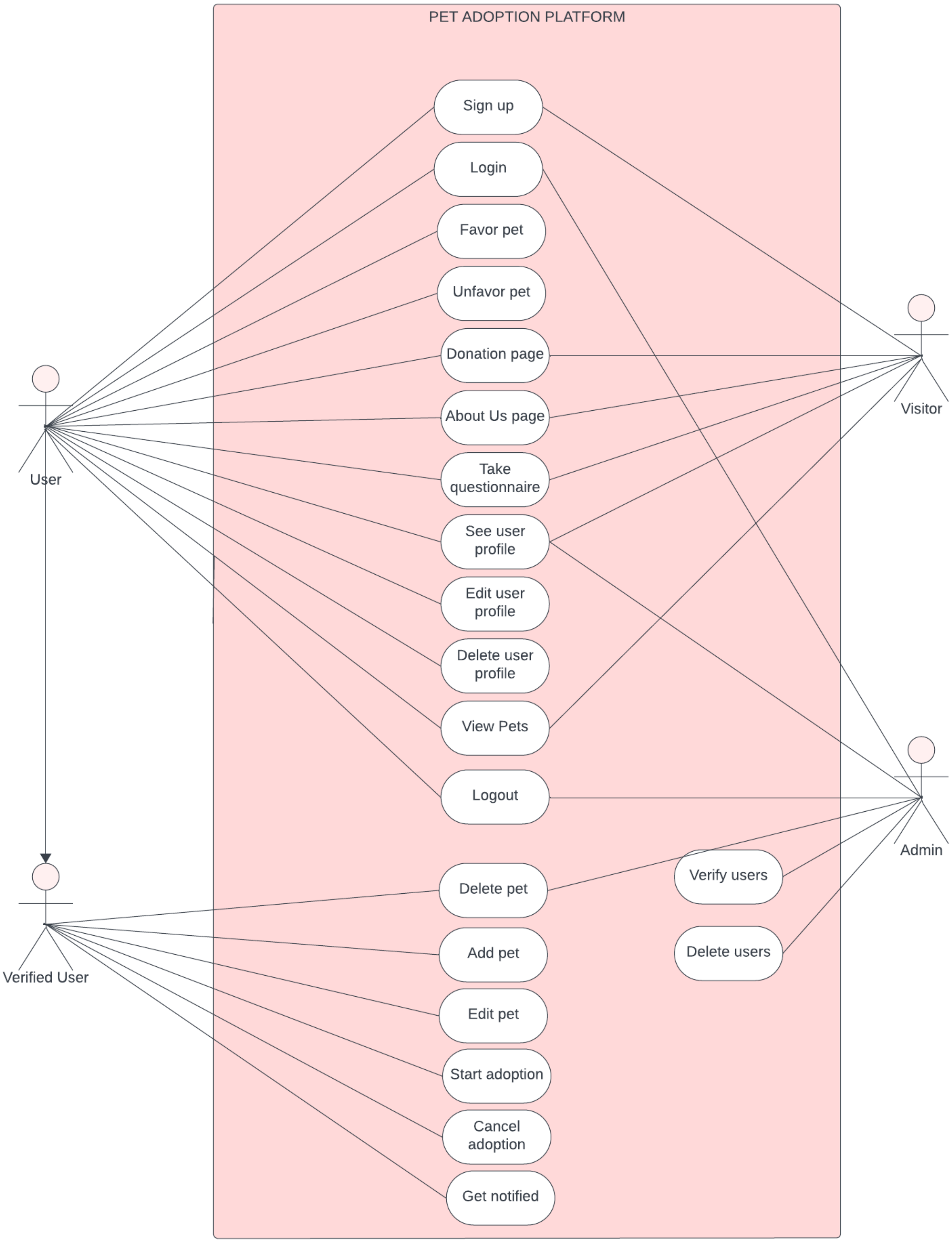


Figure 1 Use Case Model of Pet Adoption Website.

3.4.2. Object Model

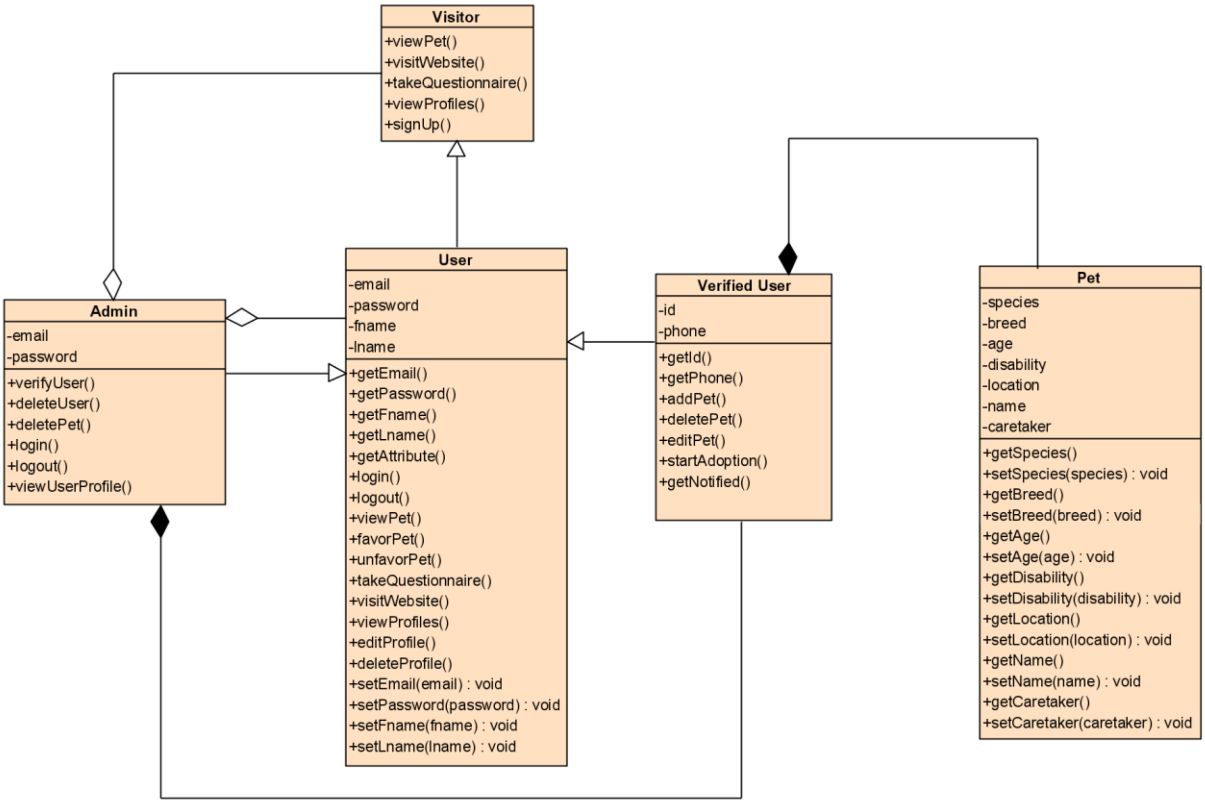


Figure 2 Class Diagram of Pet Adoption Website.

3.4.3. Dynamic Model

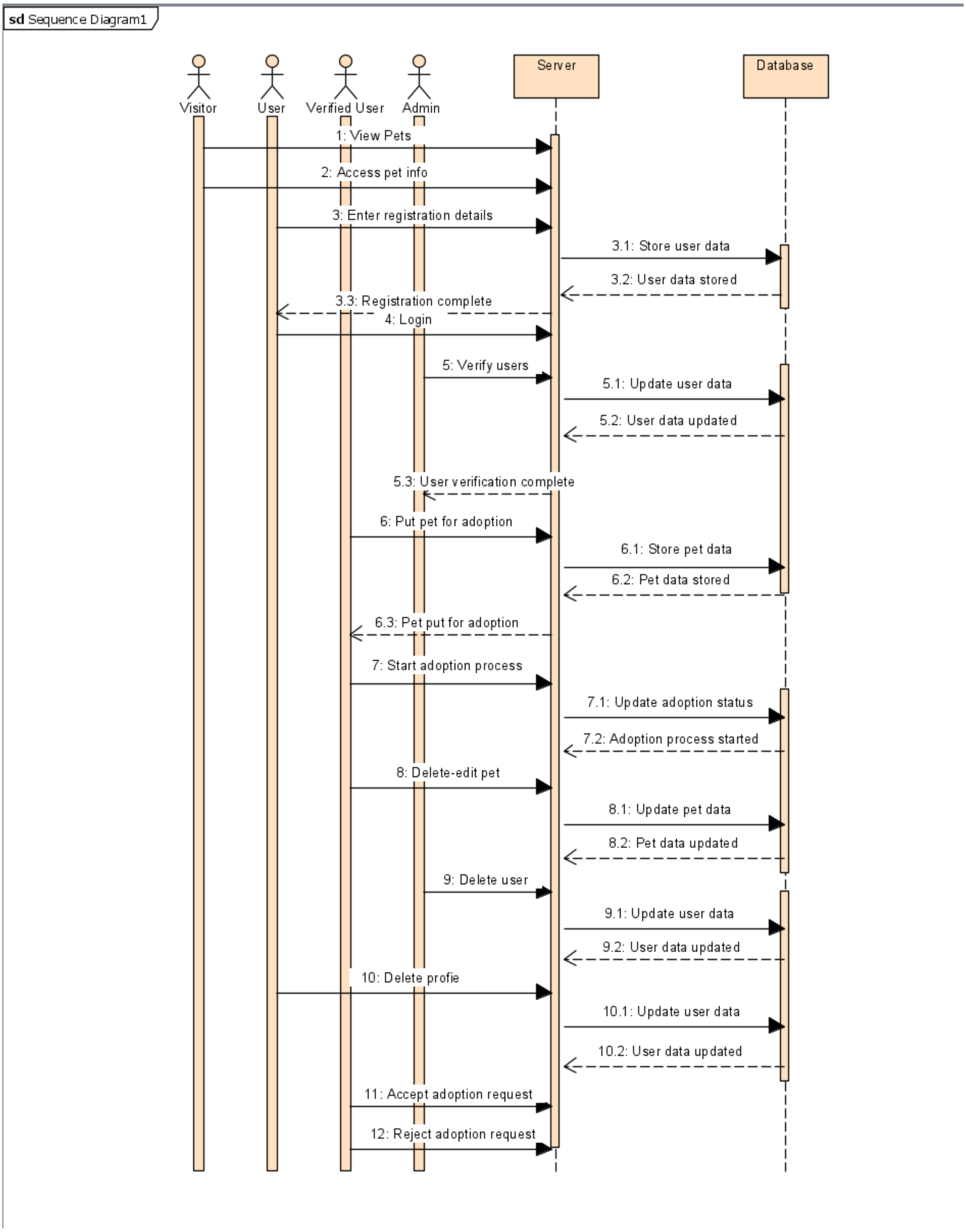


Figure 3 Sequence Diagram for Pet Adoption Website.

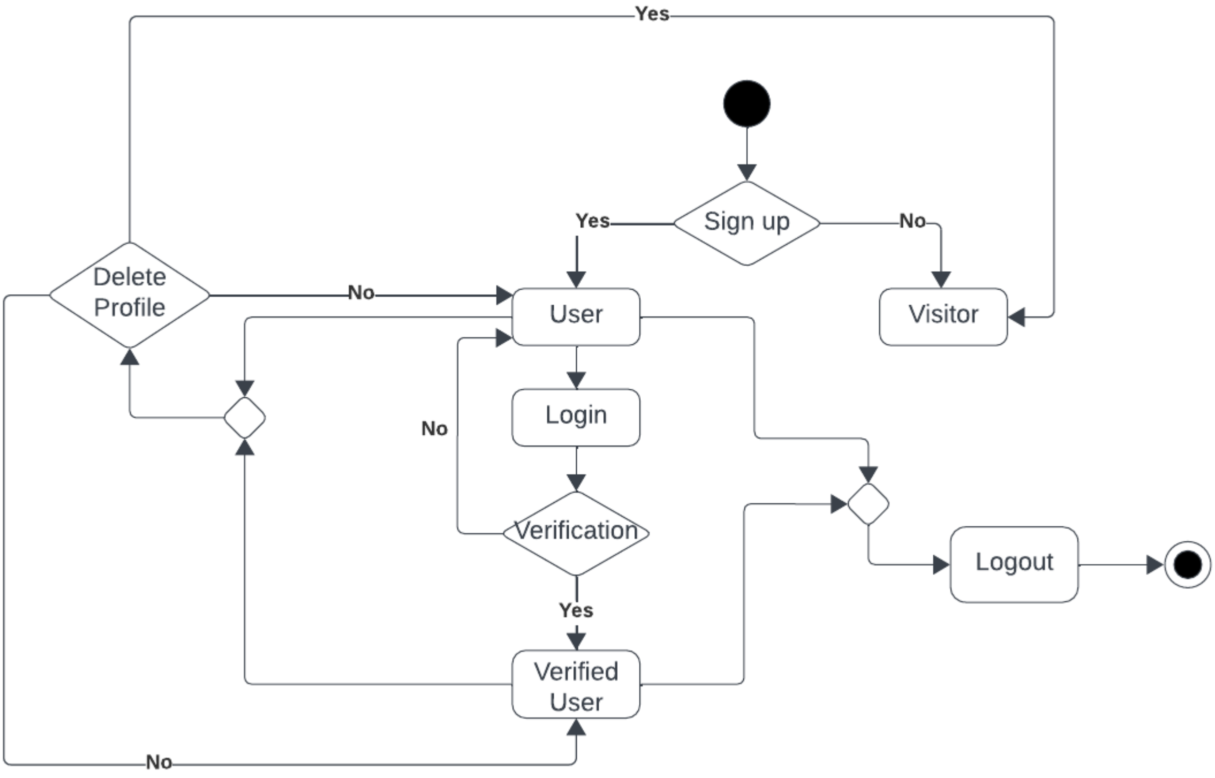


Figure 4 State Machine Diagram for Users.

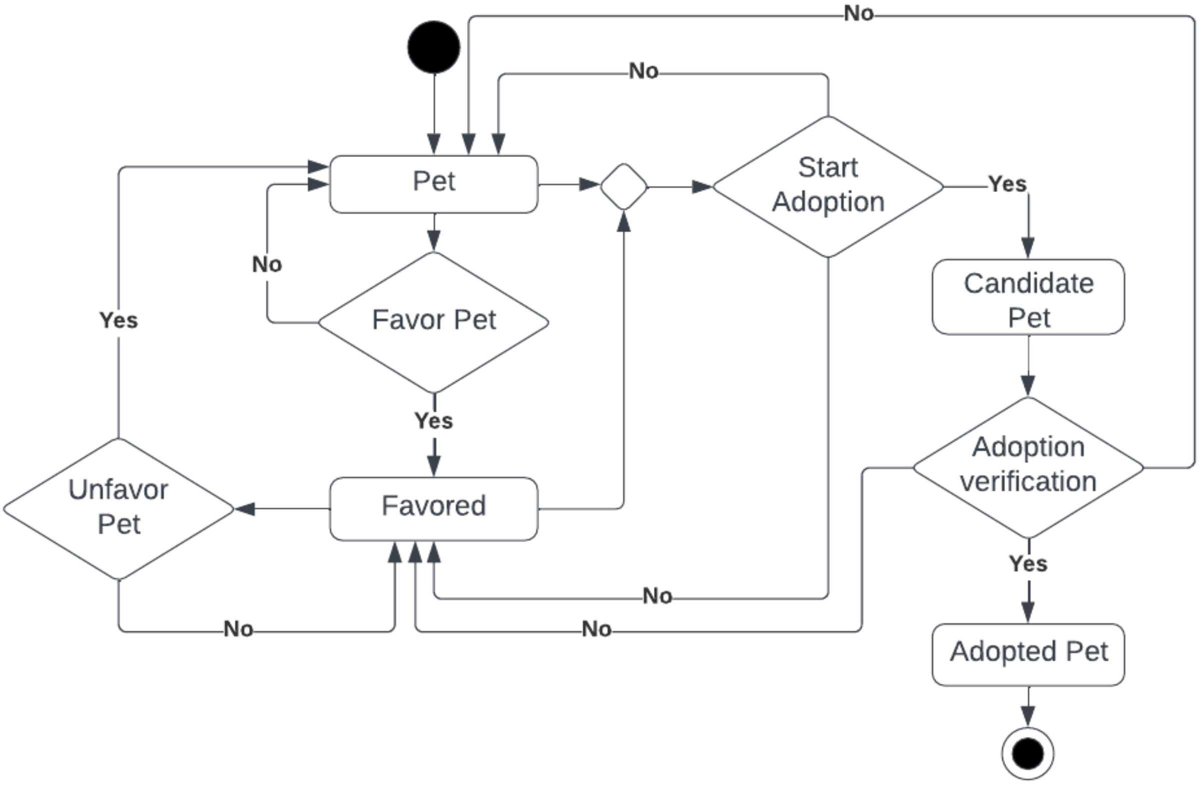


Figure 5 State Machine Diagram for Pets.

3.4.4. User Interface—Navigational Paths and Screen Mock-ups

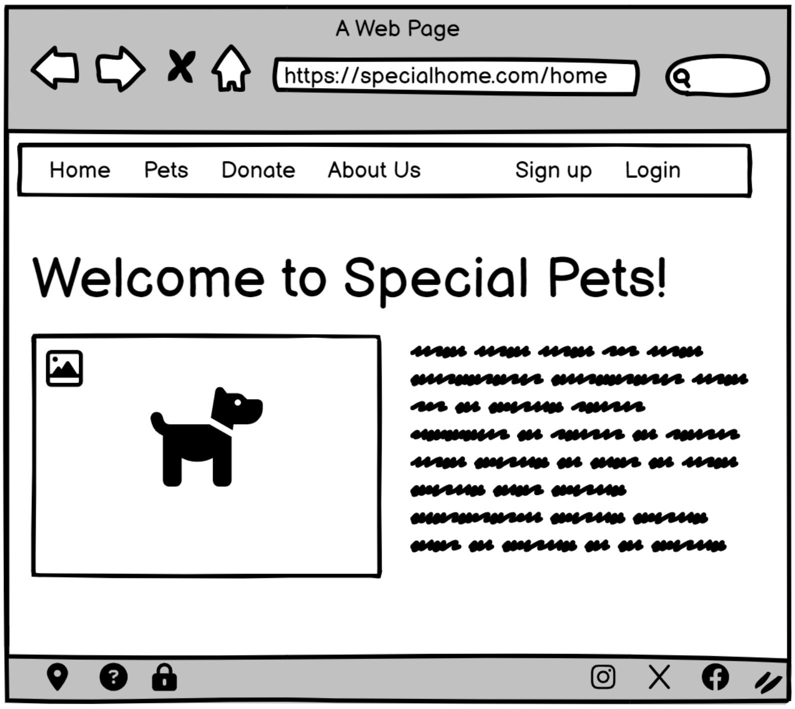


Figure 6 Prototype of Pet Adoption Homepage.

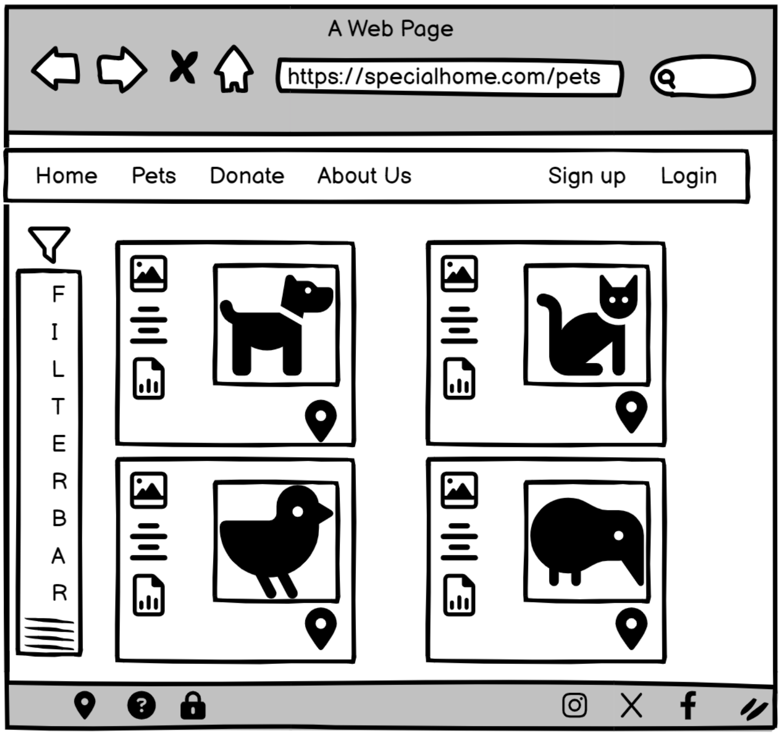


Figure 7 Prototype of Pet Lists.

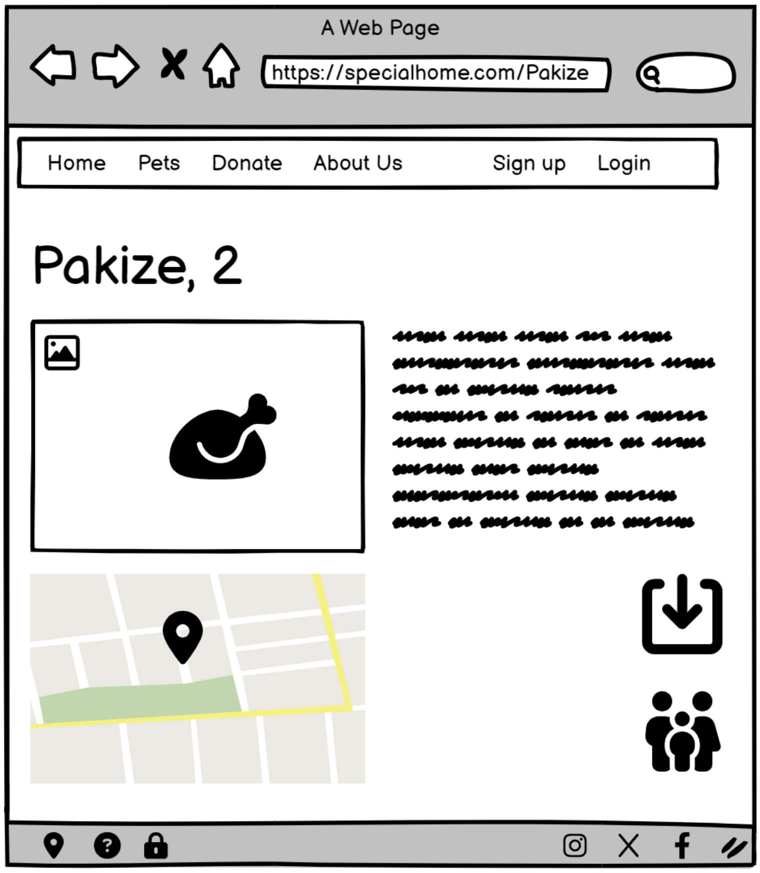


Figure 8 Prototype of Pet Page.

4. Other Analysis Elements

4.1.Risks and Alternatives

Table 14. Risks

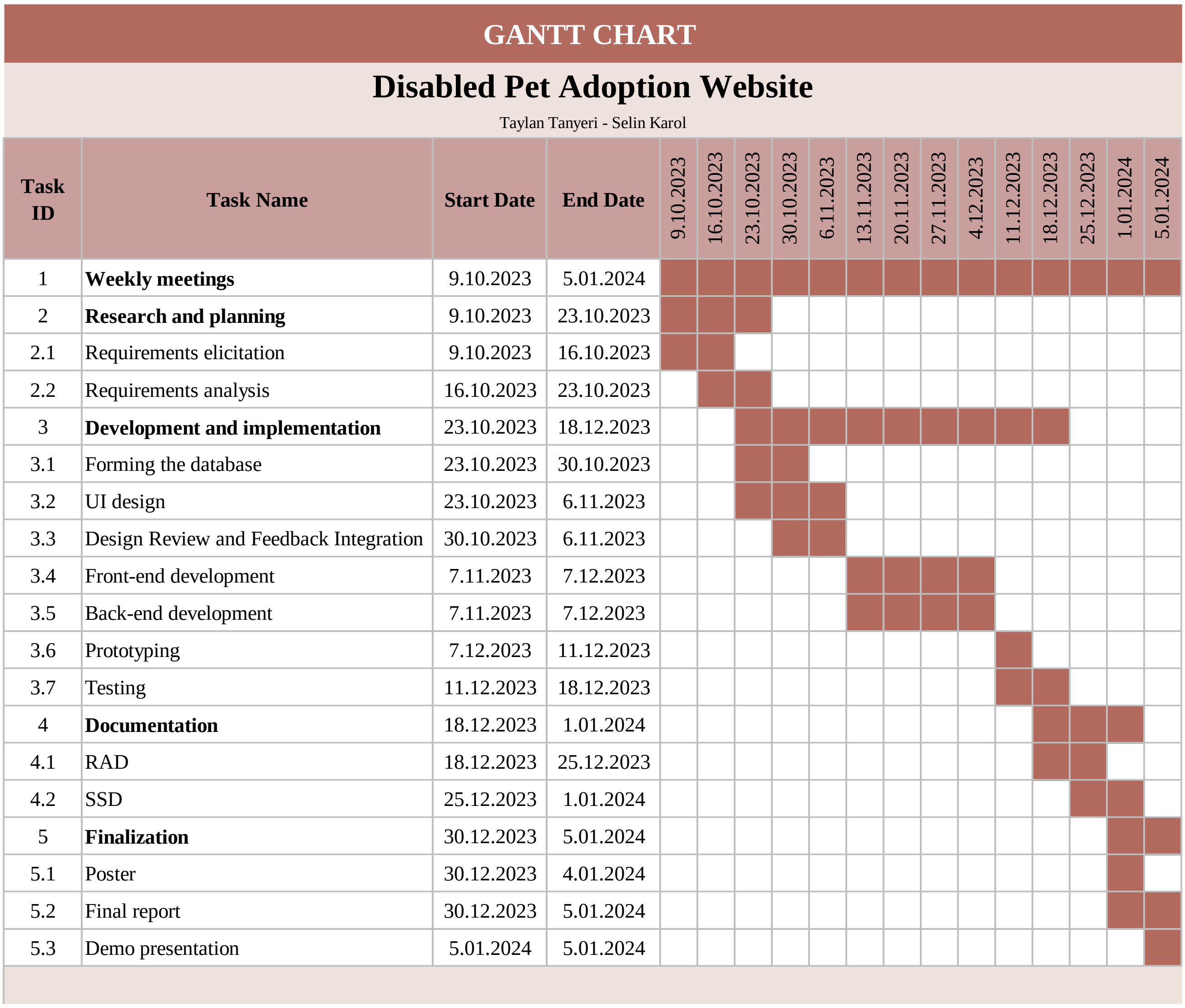
	Likelihood	Effect on the project	B Plan Summary
Risk 1: Mistreatment of animals by uncaring or irresponsible families	Medium	Endangering the trustworthiness of the site and decreasing the credibility of the platform.	Evaluating candidate families in more detail, conducting reference checks and arranging home visits.
Risk 2: Fake applications and information	High	Keeping the server occupied unnecessarily, hard time to access accurate information.	Strengthening authentication methods and using security software as well as requiring personal ID for verification.
Risk 3: Physical victimization or unintentional return of ill animals by families	High	Failing to achieve the goal of the project, potential damage to the project's reputation.	Establishing a help line for the families seeking help regarding the care and the treatment of the pets, developing comprehensive educational

			programs by getting in touch with related NGOs for awareness and promotion.
Risk 4: Cyber-attacks regarding the security of the site	Low	Compromised data integrity, site downtime.	Building a team of cybersecurity experts to ensure safeguarding data and information to prevent unauthorized access, preservation of data.
Risk 5: Potential harm coming from the animal	Low	Potential damage to the project's reputation, demotivation of adopters.	The system would not be held liable, but rather the facilities and pet owners will take full responsibility by legal contracts.
Risk 6: Utilization of the system	High	Conflicts due to not updated data and utilization problems.	The pets need to be confirmed monthly by the facilities and caretakers, regardless of their current situation. If a pet is adopted, it must be deleted.

4.2.Project Plan

Table 15. List of work packages

WP#	Work package title	Members involved
WP1	Research and Planning	Selin-Taylan
WP2	Development and Implementation	Selin-Taylan
WP3	Documentation	Selin-Taylan
WP4	Finalization	Selin-Taylan



GANTT CHART

Disabled Pet Adoption Website

Taylan Tanyeri - Selin Karol

Task ID	Task Name	Start Date	End Date	9.10.2023	16.10.2023	23.10.2023	30.10.2023	6.11.2023	13.11.2023	20.11.2023	27.11.2023	4.12.2023	11.12.2023	18.12.2023	25.12.2023	1.01.2024	5.01.2024
1	**Weekly meetings**	9.10.2023	5.01.2024	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
2	**Research and planning**	9.10.2023	23.10.2023	🟥	🟥	🟥											
2.1	Requirements elicitation	9.10.2023	16.10.2023	🟥	🟥												
2.2	Requirements analysis	16.10.2023	23.10.2023		🟥	🟥											
3	**Development and implementation**	23.10.2023	18.12.2023			🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥			
3.1	Forming the database	23.10.2023	30.10.2023			🟥	🟥										
3.2	UI design	23.10.2023	6.11.2023			🟥	🟥	🟥									
3.3	Design Review and Feedback Integration	30.10.2023	6.11.2023				🟥	🟥									
3.4	Front-end development	7.11.2023	7.12.2023						🟥	🟥	🟥	🟥					
3.5	Back-end development	7.11.2023	7.12.2023						🟥	🟥	🟥	🟥					
3.6	Prototyping	7.12.2023	11.12.2023										🟥				
3.7	Testing	11.12.2023	18.12.2023										🟥	🟥			
4	**Documentation**	18.12.2023	1.01.2024											🟥	🟥	🟥	
4.1	RAD	18.12.2023	25.12.2023											🟥	🟥		
4.2	SSD	25.12.2023	1.01.2024												🟥	🟥	
5	**Finalization**	30.12.2023	5.01.2024													🟥	🟥
5.1	Poster	30.12.2023	4.01.2024													🟥	
5.2	Final report	30.12.2023	5.01.2024													🟥	🟥
5.3	Demo presentation	5.01.2024	5.01.2024														🟥

Figure 9 Gantt Chart.

Table 16. Work Package 1

WP 1: *Research and Planning*			
Start date: *9.10.2023* **End date:** *23.10.2023*			
Leader:	*Selin Karol*	**Members involved:**	*Taylan Tanyeri*
Objectives: *Go over the already-existing projects to come up with some ideas regarding the requirements and features of the project. Fully understanding what is expected from this project and what needs to be added in order to make it unique and useful.*			
Tasks: ***Task 1.1 <Requirements Elicitation>:*** *Gathering requirements from potential users in order to decide on the necessary features of the project. To have a fully functional project with a good*			

success rate, it is important to satisfy the demands of the customers and provide an application to ensure what needs to be done can be achieved via using the project. ***Task 1.2 <Requirements Analysis>:*** *Analyzing the requirements that has been gathered from users and similar projects so that the newly constructed project will reach the user experience goals and be fit to the needs of the users. It is crucial to do the analysis correctly so that the project can be developed and implemented in a way that aligns with the users' expectations and requirements.*
Deliverables ***D1.1:*** *Gathering requirements* ***D1.2:*** *Understanding user needs and expectations* ***D1.3:*** *Deciding on the features* ***D1.4:*** *Forming a list of operations to be performed by the users to reach their goals* ***D1.5:*** *Coming up with solutions with respect to user problems and demands* ***D1.6:*** *Satisfaction of users*

Table 17. Work Package 2

WP 2: *Development and Implementation*			
Start date: *23.10.2023* **End date:** *18.12.2023*			
Leader:	*Taylan Tanyeri*	**Members involved:**	*Selin Karol*
Objectives: *By bearing in mind the requirements and user needs, deciding on the proper architecture, UI design and securing the system with some features. Furthermore, arranging components & tools to use, frameworks and libraries to download and implement, coding efficiently and cleanly. Controlling the version, testing and performance optimization by monitoring the maintenance.*			
Tasks: ***Task 1.1 <Forming the Database>:*** *Choosing a database management system and tool, designing the schema and uploading the related data to the database will save the pet information and introduce them to the system.* ***Task 1.2 <UI Design>:*** *To ensure a user-friendly application, needed features must take place along with an easily used program. Building prototypes and improving the interface with*			

feedback will contribute to this goal. Providing a responsive design is also a key when it comes to web development, allowing users to perform their actions without encountering UI based errors.

Task 1.3 <Design Review and Feedback Integration>: *In order for the project goals to meet user requirements, it is crucial to have feedback and continuous integration. This will facilitate maintaining the quality high. Design reviews will be used to revising, making adjustments to better the UX goals.*

Task 1.4 <Front-end Development>: *Process of creating interactive components that will be visible to the user side involves various steps. Coming up with proper web technologies and front-end frameworks like React or Vue are the responsibilities of the front end developers to ensure proper implementation of design, quality of UI and UX.*

Task 1.5 <Back-end Development>: *Choosing server technologies and programming languages, handling data and server management directly involves backend development. Routing and rendering is necessary between components and securing the data within the database concerns this development type.*

Task 1.6 <Prototyping>: *Prototyping is used for version control and observation of the features to be improved. Preliminary model of a product is created and tested so that developers can see the functionality and usability.*

Task 1.7 <Testing>: *In order to identify errors and defects, testing is used. Verification of the functionalities and fixing the defects is within the boundaries of testing & correcting.*

Deliverables

D1.1: *Database schema*

D1.2: *User interface design*

D1.3: *Front-end codebase*

D1.4: *Back-end codebase and API documentation*

D1.5: *Prototype models and mock-ups*

D1.6: *Test results*

D1.7: *User interface implementation*

D1.8: *Client-server authentication*

D1.9: *Cookie management*

Table 18. Work Package 3

WP 3: *Documentation*
Start date: *18.12.2023* **End date:** *1.1.2024*

Leader:	*Selin Karol*	**Members involved:**	*Taylan Tanyeri*
Objectives: *Keeping records of events, decisions, actions; listing the requirements and features, building a structural base to be followed and implemented throughout the process.*			
Tasks: ***Task 1.1 <RAD>:*** *Requirements Analysis Document is used for elicitation and analysis of the user demands, outlining what the software should do and the work product to be developed. It is a comprehensive record of the features and functions of a soon to be developed project. Explains what will be developed.* ***Task 1.2 <SSD>:*** *Software Design Document is a description of building a work product by meeting the technical requirements. Explains how a software product will be developed. The timeline, context and goals will be specified and act like a guide to developing the software.*			
Deliverables ***D1.1:*** *RAD Report* ***D1.2:*** *SSD Report*			

Table 19. Work Package 4

WP 4: *Finalization*			
Start date: *30.12.2023* **End date:** *5.1.2024*			
Leader:	*Taylan Tanyeri*	**Members involved:**	*Selin Karol*
Objectives: *Making the last touches of the presentation, preparing a final report based on the steps taken to implement the project and documentation and preparing a poster to briefly introduce the application to users.*			
Tasks: ***Task 1.1 <Poster>:*** *For the purpose of introduction of the project, a poster will be presented, summarizing the goal of the developed application.* ***Task 1.2 <Final Report>:*** *A documentation of the overall project will take place regarding the relevant information, including the design and development process, milestones achieved, challenges encountered, and lessons learned.* ***Task 1.3 <Demo Presentation>:*** *The presentation of the developed software, highlighting the software features, capabilities and the tools and third party libraries used will be acknowledged.*			

Deliverables

D1.1: *Poster*

D1.2: *Final Report*

5. Glossary

API: Also known as Application Programming Interface, mechanisms that enable software components to interact and communicate via some protocols.

Development: Process of creating application program with respect to Program Development Life Cycle, including analyzing, designing, coding, debugging and testing.

Documentation: Process of creating records of the features to do and that are done. Necessary for providing information and communication.

Framework: A structure used by developers to build a software. It provides a template for tasks to be handled.

Implementation: Code being written to develop the project.

Maintainability: Keeping something in existence, preserving. Working with changing requirements.

Non-Functional Requirements: Requirements describing how the software will perform activities, such as performance, external interface, design constraints etc.

Objectives: The outcomes or goals that the project needs to achieve.

Responsive Web Development: A technical development for a website to dynamically adapt to the width of a device in use.

Risks: Potential problems that may arise during the development. May be related to budget, implementation, testing etc.

Scope: Outlook, operation.

Testing: The process of checking whether the software is defect-free or not. It is necessary for high quality software development.

UI: Also known as User Interface, it is a tool to perform the communication between users and the product in order to achieve user goals.

Use Case: Written description of tasks that user will be performing.

UX: Also known as user experience, process of providing meaningful and desired experiences for users.

6. References

(1) Almeida, Fernando, and José Monteiro. "The Role of Responsive Design in Web Development." Webology 14.2 (2017).

(2) Almeida, Jason, et al. “Requirements Analysis Document.” Carnegie Mellon University, www.cs.cmu.edu/afs/cs/academic/class/15413-s99/www/ui/doc/RAD/UI_RAD.htm. Accessed 1 Nov. 2023.

(3) Assistant Secretary for Public Affairs. “Use Cases.” Usability.Gov, Department of Health and Human Services, 9 Oct. 2013, www.usability.gov/how-to-and-tools/methods/use-cases.html. Accessed 1 Nov. 2023

(4) Bittner, Kurt, and Ian Spence. “Use case modeling.” Addison-Wesley Professional, 2003.

(5) Chung, Lawrence. “Non-Functional Requirements.” The University of Texas at Dallas. https://www.utdallas.edu/. Accessed 1 Nov. 2023.

(6) Interaction Design Foundation - IxDF. “What is User Experience (UX) Design?” Interaction Design Foundation - IxDF. https://www.interaction-design.org/literature/topics/ux-design Accessed 1 Nov. 2023.

(7) McKay, Everett N. “UI is communication: How to design intuitive, user centered interfaces by focusing on effective communication.” Newnes, 2013.

(8) Monteiro, Tiago. “What Is a Framework? Software Frameworks Definition.” freeCodeCamp.Org, 27 Apr. 2023, www.freecodecamp.org/news/what-is-a-framework-software-frameworks-definition/. Accessed 1 Nov. 2023.

(9) “Programming Development Cycles.” California State University, Long Beach, home.csulb.edu/~pnguyen/cecs100/lecturenotes/Programming%20Development%20Cycles.docx . Accessed 1 Nov. 2023.

(10) “Rad Documentation Guide.” OGCIO, 16 Mar. 2015, www.ogcio.gov.hk/en/our_work/infrastructure/methodology/system_development/past_documents/rad/documentation_guide.html. Accessed 30 Oct. 2023.

(11)“Software Engineering: Risk Management.” GeeksforGeeks, 18 May 2023, www.geeksforgeeks.org/software-engineering-risk-management/. Accessed 30 Oct. 2023.

(12) Uddin, Azeem & Anand, Abhineet. Importance of Software Testing in the Process of Software Development. 2019. Accessed 30 Oct. 2023

(13) “What Is an API (Application Programming Interface)?” Amazon, aws.amazon.com/what-is/api/. Accessed 1 Nov. 2023.