

Phase Two

For CPSY301



November 10, 2023

Authors:

Samantha bogen

Rolan ho

lisa kamdem

Dorian Laycock

Markus Luthi

Contents

[Project Selection and Preliminary Analysis 2](#_Toc150625043)

[About the Organization: 2](#_Toc150625044)

[Current System Description: 3](#_Toc150625045)

[Proposed Solution: 3](#_Toc150625046)

[Use Case Diagram and Use Case Descriptions 4](#_Toc150625047)

[Use Case Diagram for Foster Database: 4](#_Toc150625048)

[Use Case Diagram for Digital Form: 6](#_Toc150625049)

[Use Case Diagram for Waitlist: 7](#_Toc150625050)

[Revised Use Case Diagram and Use Case Descriptions in extended format 7](#_Toc150625051)

[Use Case Diagram for Searching a Pet: 8](#_Toc150625052)

[Use Case Diagram for applying to be a foster: 12](#_Toc150625053)

[Use Case Diagram for reviewing a supplier: 16](#_Toc150625054)

[Use Case Diagram for reporting a supplier: 19](#_Toc150625055)

[Preliminary User Interface Design: 24](#_Toc150625056)

[Pet foster Questionnaire: 25](#_Toc150625057)

[Saved Search (supplied by client, modified by us) 26](#_Toc150625058)

[Foster Contact (supplied by client, modified by us) 27](#_Toc150625059)

[Report Fraud Form (supplied by client) 28](#_Toc150625060)

[Data Storage and Persistence: 29](#_Toc150625061)

[ERD: 29](#_Toc150625062)

[Class Diagram: 30](#_Toc150625063)

[System Architecture and Patterns 30](#_Toc150625064)

[Appendix 31](#_Toc150625065)

[Team Constitution 31](#_Toc150625066)

[Team Curious Comets Members 31](#_Toc150625067)

[Statement of Team Goals 32](#_Toc150625068)

[Intellectual Property Statement 32](#_Toc150625069)

[Team Member Roles 32](#_Toc150625070)

[Division of Labour 32](#_Toc150625071)

[Group Expectations 32](#_Toc150625072)

[Plans for Resolving Conflict 32](#_Toc150625073)

[Plan if Conflict can not be Resolved 33](#_Toc150625074)

[Summary 33](#_Toc150625075)

# Project Selection and Preliminary Analysis

Project Title: PetStarz Integrated Management System

Project Sponsor: April Shymko, Founder and CEO of "PetStarz"

## About the Organization:

PetStarz is an innovative startup with a laser focus on transforming the pet industry through a groundbreaking Software-as-a-Service (SaaS) online platform. The organization's central mission is to bring about positive, systemic change by offering a comprehensive, digital solution that addresses multiple facets of pet ownership. Specifically, PetStarz has identified four domains that are integral to its operational framework:

1. Ethical Pet Sourcing: The platform seeks to facilitate an informed and responsible choice for potential pet owners by listing pets sourced through ethical means.

2. Supplier Verification: Acting as a bridge between users and a network of verified suppliers—which includes breeders, shelters, and rescue organizations—the platform emphasizes credibility and humane practices.

3. Fraudulent Practice Reporting: Committed to maintaining a high level of integrity within the industry, PetStarz incorporates a mechanism allowing users to report unethical or fraudulent activities. This acts as a self-regulating feature to uphold standards within the platform.

4. Information Dissemination: A cornerstone of the PetStarz platform is its dedicated space for educational content and research. This aims to ensure that pet owners are well-informed and equipped with the knowledge required for responsible pet care.

Designed to be intuitive, user-friendly, and reliable, the platform includes the core values of trust, ethical sourcing, pet welfare, user empowerment, and community support. The ultimate vision is not just to offer a service, but to cultivate a harmonious ecosystem that emphasizes the welfare of pets while empowering users. This aligns with the organization's long-term strategic objectives and offers a scalable solution adaptable to future industry challenges and user needs.

## Current System Description:

As of now, PetStarz is in a foundational stage, operating on a basic digital platform that serves as a preliminary step towards achieving its ambitious and visionary goals. This platform is designed to become a one-stop solution for a local community of pet enthusiasts. However, it's important to note that several crucial elements are still in the planning or conceptual phase, yet concrete implementation. While the platform provides a skeleton structure that aligns with the company's vision of revolutionizing pet ownership and sourcing, it currently falls short in offering the full range of features and functionalities envisioned. The platform lays the groundwork but hasn't yet reached its full potential, which is essential for meeting the diverse and specific needs of pet lovers.

## Proposed Solution:

The "PetStarz Integrated Management System" will include the following features:

1. Unified Pet Listings: In this project, one of the primary features is to build an all-inclusive database covering a variety of pets. This includes canines, felines, and other more exotic animals for adoption or foster. The aim is to provide an advanced and user-friendly search functionality that allows users to sift through listings based on specific criteria like species, breed, age, and location. The addition of a 'Foster Database' extends the scope of the system by listing pets available for temporary care, thus adding an alternative option for users who may not be looking for a long-term commitment but still want to engage with the pet community.

2. Supplier Network: The next step in this project is to develop a Supplier Network feature. This will facilitate a direct link between end-users and a vast, validated network of suppliers, which include breeders, rescue organizations, and shelters. Rigorous validation protocols will be built into the platform to ensure that all suppliers meet the standard of ethical behavior and care for the animals. Additionally, this feature will allow shelters and breeders to create, customize, and manage their own profile pages on the platform, giving users multiple touchpoints to gather information.

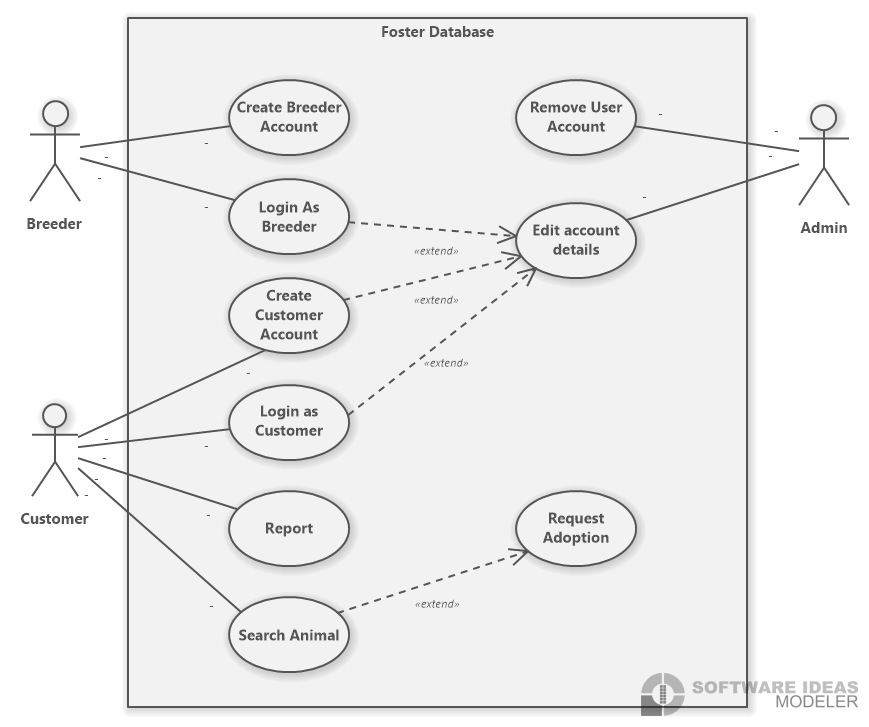
3. Fraud Reporting Mechanism with Digital Web Form: A key feature of the project is a Fraud Reporting Mechanism, where a digital web form on the platform allows users to report unethical or suspicious activities. This form simplifies the reporting process by collecting details like the complaint's nature and involved parties. It enhances the platform's commitment to ethical sourcing and integrity. This feature aims to promote transparency and trust within the pet adoption ecosystem, ensuring a safer environment for all users.

4. Research Hub: This part will not only serve as a repository of curated articles and guidelines but also feature a Q&A database. The latter is designed to function as an interactive FAQ platform that supports user queries. Users can post their questions regarding pet laws, ethical supplier selection, or general pet care, which could be addressed by experts in the field like veterinarians. This feature enhances the platform's usability, offering immediate, reliable information for users seeking to make informed decisions related to pet ownership.

5. Community Engagement: It is a significant aspect of this project. Features such as interactive forums and discussion boards are planned. A 'Lost & Found' section will provide a community-driven platform to help locate lost pets. A dedicated 'Events Page' will also be designed to allow stakeholders like shelters and breeders to announce upcoming events, thus facilitating community interactions and knowledge-sharing.

# Use Case Diagram and Use Case Descriptions

### Use Case Diagram for Foster Database:



The breeder will create a breeder account with a username and password so they can added as a breeder to the database.

The breeder after creating an account can login as a breeder with the correct username and password, can later on edit their account details if necessary.

The customer will create a customer account with a username and password so they can added as a customer to the database.

The customer after creating an account can login as a customer with the correct username and password, can later on edit their account details if necessary.

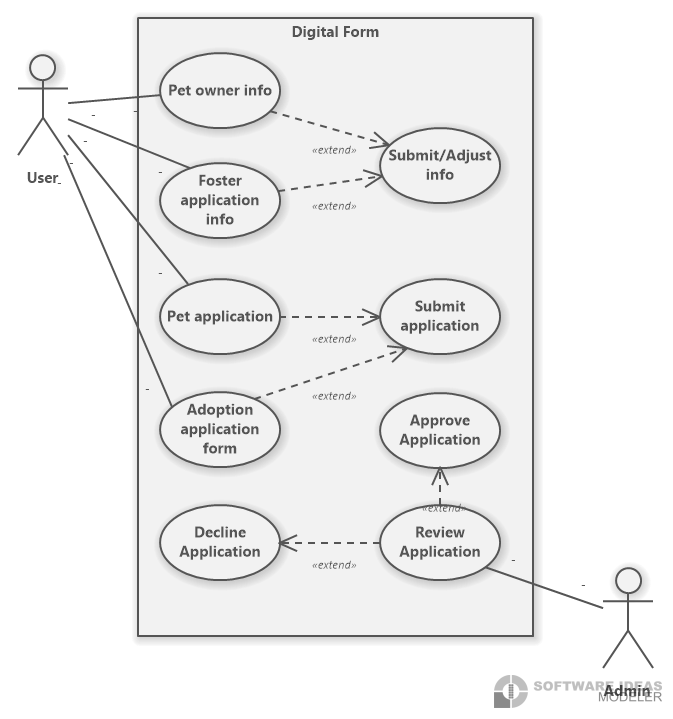
The customer can report if they feel like an account is fake or giving out fake information about the information.

The customer can also search animals and look at all the animal that PetStarz have available and if they want to adopt can request to adopt.

The admin can remove a user account if they have been report or a fake account from the database.

The admin can edit account details from all the customers and breeders in the database.

### Use Case Diagram for Digital Form:



The user will need to enter the required information in the pet owner information form and can then adjust the information and submit the form once they have all the required information, if the user is already a pet owner.

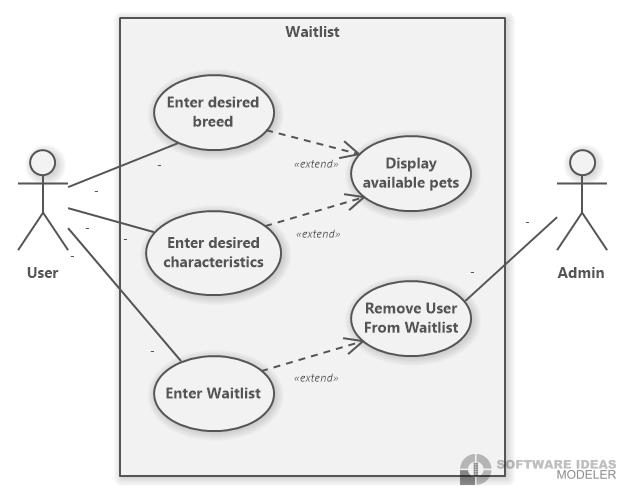
The user will need to enter the required information in the foster application information form and can then adjust the information and submit the form once they have all the required information, if the user wants to become an applicant for fostering.

The user will need to enter the required information in the pet application form and submit the form once they have all the required information, if they want to apply for a pet.

The user will need to enter the required information in the adoption application information form and then adjust the information and submit the form once they have all the required information, if the user wants to become an applicant for adoption.

The admin will review all the application that have been submitted and can either approve to application or decline it.

### Use Case Diagram for Waitlist:



The user can enter the desired breed they are looking for and then the website will display all the available pets once the user clicks enter.

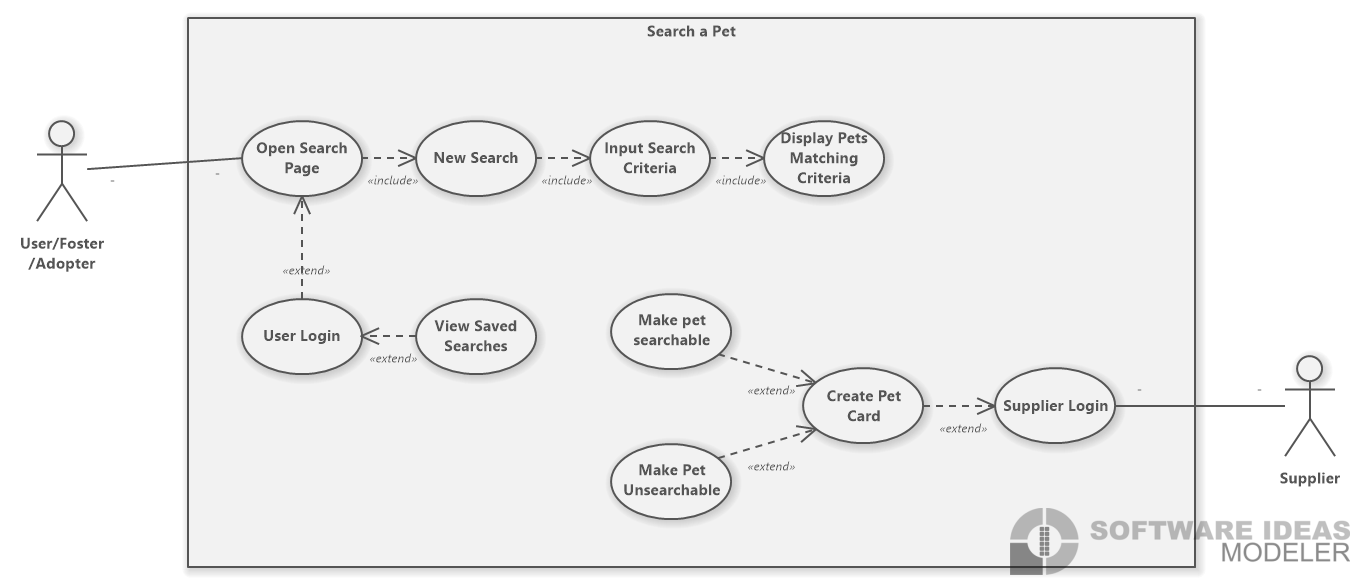
The user can enter the desired characteristics they are looking for and then the website will display all the available pets once the user clicks enter.

The user can enter in the waitlist and later on can remove themselves from the waitlist if necessary.

The admin can remove any user on the waitlist if necessary.

# Revised Use Case Diagram and Use Case Descriptions in extended format

### Use Case Diagram for Searching a Pet:



|  |  |
| --- | --- |
| Name | Open Search Page |
| Actors | User/Foster/Adopter |
| Limitation |  |
| Precondition | Must logged in to open search page |
| Postcondition | Can accessed search page |
| Normal flow | Logged in directs way to search page |
| Alt flow | Cannot view search page, must log in to view search page |

|  |  |
| --- | --- |
| Name | User Login |
| Actors | User/foster/adopter |
| Limitation | Cannot search for a pet. Must have a created account to be able to apply to become a foster |
| Precondition | The user must have an existing account |
| Postcondition | The user can have access to apply to the Forster |
| Normal flow | The user clicks the sign-in button  enters their email/password  If the information is correct they can now log in |
| Alt flow | 1. The user doesn’t have an account. 2. They must create one 3. They enter the wrong password. They have 3 tries, after that, they must reset their password |

|  |  |
| --- | --- |
| Name | View Saved Searches |
| Actors | User/foster/adopter |
| Limitation |  |
| Precondition | Must have an existing account and have existing saved searched |
| Postcondition | View the saved searched list |
| Normal flow | Must be logged in, direct themselves to their profile, click an saved search, view all their saved search |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | New Search |
| Actors | User/foster/adopter |
| Limitation |  |
| Precondition | Must be on the search page |
| Postcondition | Display pet’s matching the criteria of the search query. |
| Normal flow | Navigate to website, navigate to search, select criteria, click search, view all pets meeting criteria |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Input search criteria |
| Actors | User/foster/adopter |
| Limitation | Not all animals meet the criteria being searched |
| Precondition | Logged in and open the search page a click on new search |
| Postcondition | User can now input the necessary criteria they are looking for |
| Normal flow | Logged in, in the search page, click new search and input the criteria they are looking for |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Display pets matching criteria |
| Actors | User/foster/adopter |
| Limitation | Not all pets meet the criteria |
| Precondition | A pet with the meeting criteria must exist |
| Postcondition | The user is able to see the pet with the matching criteria inputted |
| Normal flow | User logs in goes to the search page enters the pet criteria they are looking for and the website will display the matching pets with those inputted criteria |
| Alt flow |  |

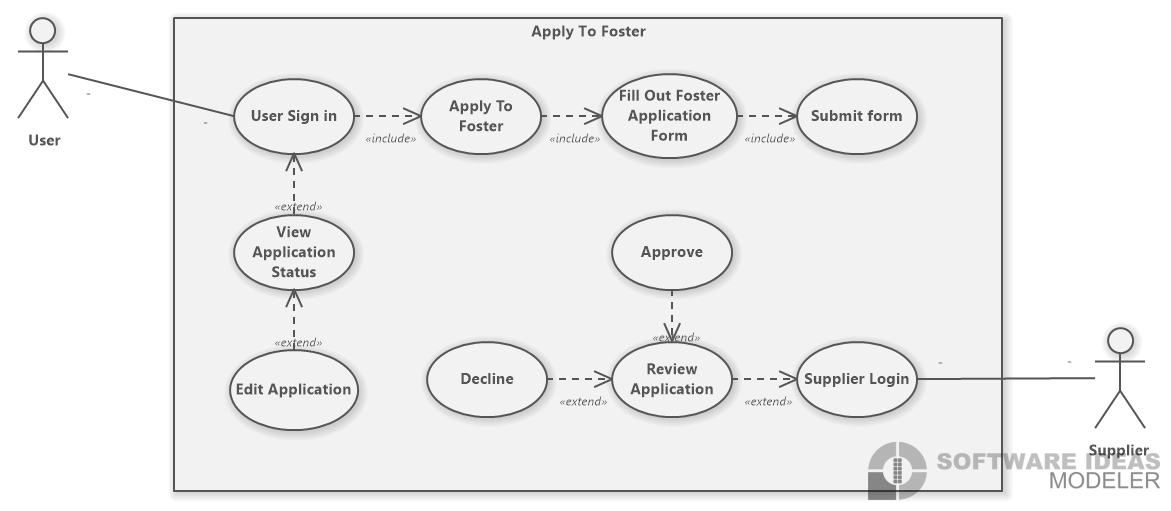
|  |  |
| --- | --- |
| Name | Supplier Login |
| Actors | supplier |
| Limitation |  |
| Precondition | The supplier must have an existing account |
| Postcondition | The supplier can have access to all the forms that have been submitted |
| Normal flow | 1. The supplier clicks the sign-in button 2. enters their email/password 3. If the information is correct they can now log in |
| Alt flow | 1. The supplier doesn’t have an account. They must create one 2. They enter the wrong password. They have 3 tries, after that, they must reset their password |

|  |  |
| --- | --- |
| Name | create pet card |
| Actors | Supplier |
| Limitation |  |
| Precondition | A pet needs to existed in the system |
| Postcondition | The animal has now a pet card |
| Normal flow | supplier login create new pet card, fill out the pet card form enter all requirements necessary and then |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Make Pet searchable |
| Actors | Supplier |
| Limitation |  |
| Precondition | A pet must have an existing pet card |
| Postcondition | A pet is now searchable for people to see the pet |
| Normal flow | supplier logs in goes to necessary pet and makes the pet’s card searchable |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Make pet unsearchable |
| Actors | Supplier |
| Limitation |  |
| Precondition | A pet must have an existing pet card |
| Postcondition | A pet is now unsearchable for people to not see the pet |
| Normal flow | supplier logs in goes to necessary pet and makes the pet’s card unsearchable |
| Alt flow |  |

### Use Case Diagram for applying to be a foster:



|  |  |
| --- | --- |
| Name | User sign in |
| Actors | User |
| Limitation | Cannot accept/decline their application. Must have a created account to be able to apply to become a foster |
| Precondition | The user must have an existing account |
| Postcondition | The user can have access to apply to the Foster |
| Normal flow | The user clicks the sign-in button  enters their email/password  If the information is correct they can now log in |
| Alt flow | 1. The user doesn’t have an account. They must create one 2. They enter the wrong password. They have 3 tries, after that, they must reset their password |

|  |  |
| --- | --- |
| Name | Apply to foster |
| Actors | User |
| Limitation | Only members/ people who have an account can apply to foster |
| Precondition | Must have an account |
| Postcondition | Can have access to the application form |
| Normal flow | The signed-in user clicks on the apply to foster |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Fill out the foster Application form |
| Actors | User |
| Limitation | Only members/ people who have an account can fill out this form |
| Precondition | Must have an account and click on the Apply Foster button |
| Postcondition | The user's application has been sent out |
| Normal flow | The user enters all the necessary requirement information for the application form |
| Alt flow | 1. The user doesn't know the answer to some of the questions in the form 2. the form saves the already imputed information 3. The user can send an email to ask for clarification and come back to it later on |

|  |  |
| --- | --- |
| Name | Submit form |
| Actors | User |
| Limitation |  |
| Precondition | All the required questions in the form should be filled out |
| Postcondition | The user’s application will be sent out to the supplier to view |
| Normal flow | 1. The user answers all the required information for the foster form 2. Click the submit button |
| Alt flow | 1. The user clicks the submit button and the user forgot an a question so the pages shows which question the forgot |

|  |  |
| --- | --- |
| Name | View application status |
| Actors | User |
| Limitation |  |
| Precondition | An application must have been made |
| Postcondition | Being able to see if your application has been approved or declined |
| Normal flow | After making an application and submitting it. The user can later go into their account and see the status of their application |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Edit application |
| Actors | user |
| Limitation |  |
| Precondition | An application form must have been submitted |
| Postcondition | New/edit information has been made to the form |
| Normal flow | After submitting the application form they can later log-in to their account and edit their application |
| Alt flow |  |

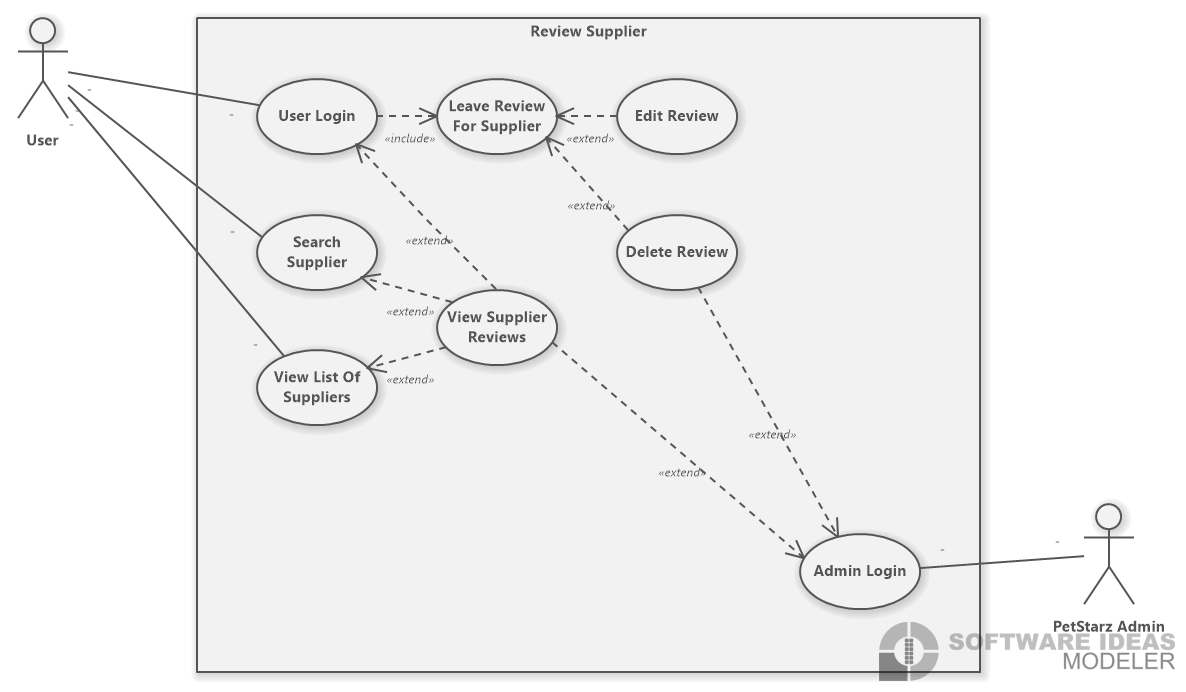
|  |  |
| --- | --- |
| Name | Supplier Login |
| Actors | supplier |
| Limitation |  |
| Precondition | The supplier must have an existing account |
| Postcondition | The supplier can have access to all the forms that have been submitted |
| Normal flow | 1. The user clicks the sign-in button 2. enters their email/password 3. If the information is correct they can now log in |
| Alt flow | 1. The supplier doesn’t have an account. They must create one 2. They enter the wrong password. They have 3 tries, after that, they must reset their password |

|  |  |
| --- | --- |
| Name | Review Application |
| Actors | supplier |
| Limitation |  |
| Precondition | Must log in as a supplier |
| Postcondition | Can access all the forms that have been submitted to become a foster. |
| Normal flow | 1. The supplier logs in 2. Navigates to review application |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Approve Application |
| Actors | Supplier |
| Limitation |  |
| Precondition | An application must be submitted and reviewed |
| Postcondition | The application was approved |
| Normal flow | the supplier reads over the application and decides that the user will be a good foster applicant |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Decline Application |
| Actors | Supplier |
| Limitation |  |
| Precondition | An application must be submitted and reviewed |
| Postcondition | The application was declined |
| Normal flow | the supplier reads over the application and decides that the user will not be a good foster applicant |
| Alt flow |  |

### Use Case Diagram for reviewing a supplier:



|  |  |
| --- | --- |
| Name | User sign in |
| Actors | User |
| Limitation | Can only leave review, search suppliers, view list of suppliers and delete/edit their own reviews |
| Precondition | The user must have an existing account |
| Postcondition | The user can have access to the list of suppliers, search a supplier, leave a review or edit an already created review |
| Normal flow | The user clicks the sign-in button  enters their email/password  If the information is correct they can now log in |
| Alt flow | 1. The user doesn’t have an account. 2. They must create one 3. They enter the wrong password. They have 3 tries, after that, they must reset their password |

|  |  |
| --- | --- |
| Name | Search Supplier |
| Actors | User |
| Limitation |  |
| Precondition | User must be logged in |
| Postcondition | Will be able to search for a supplier |
| Normal flow | After logging in th user can search for the supplier they are looking for |
| Alt flow | The supplier the user is looking for doesn’t exist. Will show an “User does not exist error, try again” |

|  |  |
| --- | --- |
| Name | View List of supplier |
| Actors | User |
| Limitation |  |
| Precondition | Enter in the supplier names in the search supplier |
| Postcondition | A list of the matching supplier will display |
| Normal flow | The use enters in the name of the supplier hey are looking for, and the lis will adjust as they type in the name to see if any supplier with that name exists |
| Alt flow | The supplier the user is looking for doesn’t exist. Will show an “User does not exist error, try again”. The view list will be empty |

|  |  |
| --- | --- |
| Name | Leave review For supplier |
| Actors | User |
| Limitation | Only members/people will accounts can leave a review |
| Precondition | Will be looking at the a suppliers profile |
| Postcondition | User can write a review for the a supplier |
| Normal flow | The user logged into their account, selects a supplier they want to leave a review for enters their review and clicks post |
| Alt flow |  |

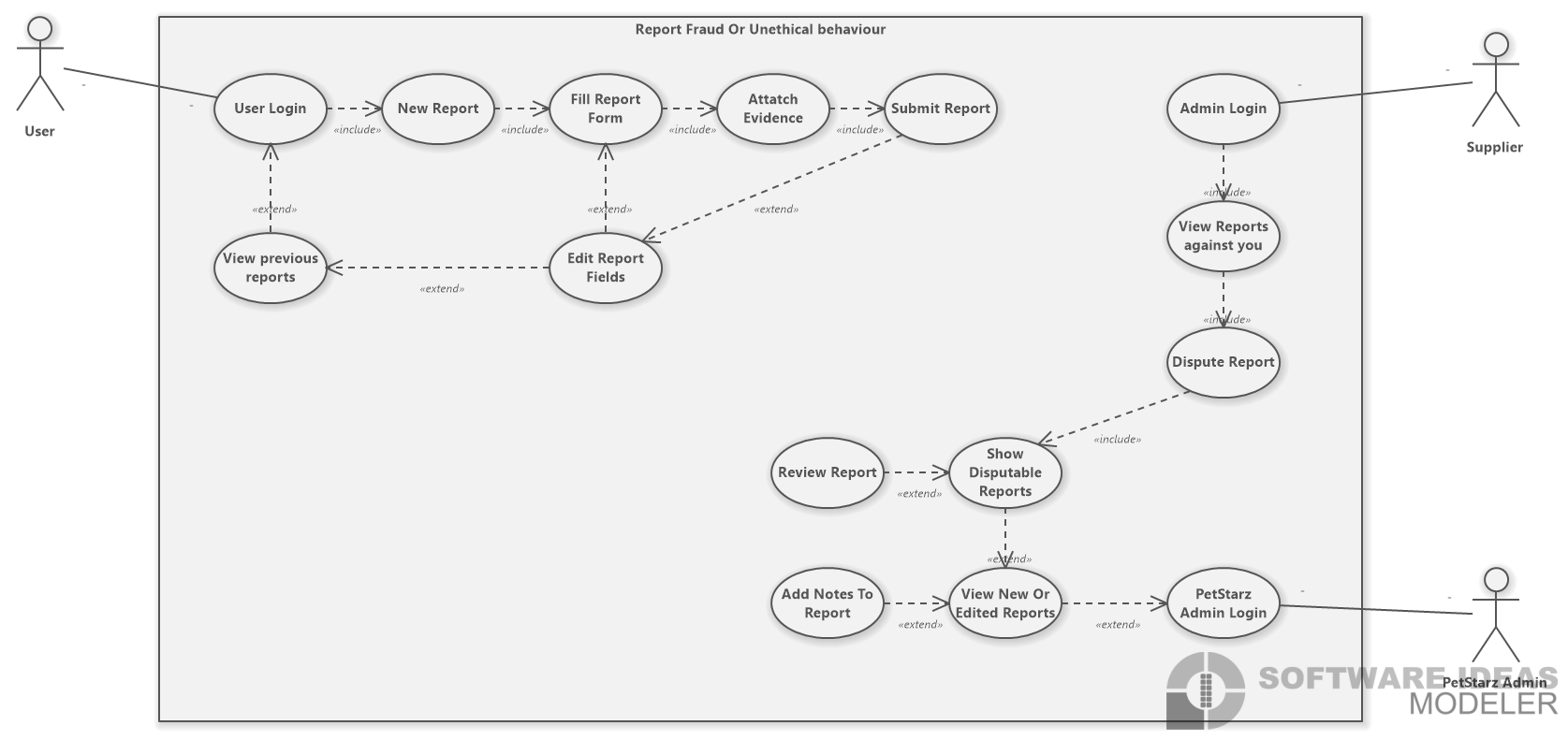
|  |  |
| --- | --- |
| Name | View supplier reviews |
| Actors | User, admin |
| Limitation |  |
| Precondition | Previous reviews must be made to the selected supplier |
| Postcondition | Will be able to view supplier reviews |
| Normal flow | The user selected a supplier they want to view their review  The admin can view all the review made by all users and see all the reviews left for each supplier |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Edit review |
| Actors | User |
| Limitation | Cannot edit other people’s reviews |
| Precondition | Must have an already made review |
| Postcondition | Can edit a review |
| Normal flow | User is logged in, goes to profile, clicks on review, direct themselves to the necessary review, presses the edit button and can edit the review they created |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Delete review |
| Actors | User, admin |
| Limitation | The user cannot delete other people’s reviews. |
| Precondition | Must have an already existing review |
| Postcondition | Can delete a review |
| Normal flow | User is logged in, goes to profile, clicks on review, direct themselves to the necessary review, presses the delete button  Admin is logged in, click on the review and can delete any review posted |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Admin Login |
| Actors | Admin |
| Limitation |  |
| Precondition | The admin must have an existing account |
| Postcondition | The admin can have access to view supplier reviews, delete reviews |
| Normal flow | The user clicks the sign-in button  enters their email/password  If the information is correct they can now log in |
| Alt flow | 1. The user doesn’t have an account. 2. They must create one 3. They enter the wrong password. They have 3 tries, after that, they must reset their password |

### Use Case Diagram for reporting a supplier:



|  |  |
| --- | --- |
| Name | User login |
| Actors | User |
| Limitation |  |
| Precondition | The user must have an existing account |
| Postcondition | The user can have access to all the forms that have been submitted |
| Normal flow | 1. The user clicks the sign-in button 2. enters their email/password 3. If the information is correct they can now log in |
| Alt flow | 1. The user doesn’t have an account. They must create one 2. They enter the wrong password. They have 3 tries, after that, they must reset their password |

|  |  |
| --- | --- |
| Name | View previous reports |
| Actors | User |
| Limitation |  |
| Precondition | Must be logged in their account must have a existing reports |
| Postcondition | The user can view previous report |
| Normal flow | User logs in goes to report and can now view previous reports |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | New report |
| Actors | User |
| Limitation |  |
| Precondition | Must be logged in, in their account |
| Postcondition | Can create a new report |
| Normal flow | User logs in in their account, goes to profile and clicks new report |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Fill report form |
| Actors | User |
| Limitation |  |
| Precondition | Clicked on new report button |
| Postcondition | The form will be filled out with the necessary requirements |
| Normal flow | After login in, clicking new report the user will need to answer/fill out the report form with all the information they have |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Edit report fields |
| Actors | User |
| Limitation |  |
| Precondition | A report previous report must of been created |
| Postcondition | Can now edit their report they previous cerated |
| Normal flow | The user logged in, goes to profile and clicks reports and clicks edit button. Can now edit the report they had |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Attach evidence |
| Actors | User |
| Limitation |  |
| Precondition | A report must be made/created already |
| Postcondition | Can attach evidence to the report the created |
| Normal flow | User logged in, went to their profile click on report and clicks on add evidence, with the evidences the user has that they want to but in their report |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Submit report |
| Actors | User |
| Limitation |  |
| Precondition | the user entered the appropriate information in the report with the evidence |
| Postcondition | The user can submit report |
| Normal flow | After the logged in user filled out the report and added the appropriate evidence they can click the submit report |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Supplier Login |
| Actors | Supplier |
| Limitation |  |
| Precondition | The supplier must have an existing account |
| Postcondition | The supplier can have access to all the forms that have been submitted |
| Normal flow | 1. The supplier clicks the sign-in button 2. enters their email/password 3. If the information is correct they can now log in |
| Alt flow | 1. The supplier doesn’t have an account. They must create one 2. They enter the wrong password. They have 3 tries, after that, they must reset their password |

|  |  |
| --- | --- |
| Name | View Reports against you |
| Actors | Supplier |
| Limitation | Can only view their own reports |
| Precondition | Must have existing report against them |
| Postcondition | Can view the reports against them |
| Normal flow | After they have logged into their supplier account, can got into their profile, click reports, click on view reports against me |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Dispute report |
| Actors | Supplier |
| Limitation | Can only dispute reports made against them. |
| Precondition | A report must exist against them |
| Postcondition | A notification will be sent to a PetStarz administrator for review. |
| Normal flow | Notification is sent to supply that a report has been made against them, the supplier views the report, navigates to a “dispute report” form, fills the form out and submits it. |
| Alt flow | A supplier may choose not to dispute a report. |

|  |  |
| --- | --- |
| Name | Show disputed reports |
| Actors | Supplier |
| Limitation | Can only see reports they’ve disputed |
| Precondition | A disputed report must exist |
| Postcondition | Can view the reports against them |
| Normal flow | After they have logged into their supplier account, can got into their profile, click reports then click on view disputed reports |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | PetStraz admin login |
| Actors | PetStraz admin |
| Limitation |  |
| Precondition | The PetStraz admin must have an existing account |
| Postcondition | The PetStraz admin can have access to all the forms that have been submitted |
| Normal flow | 1. The PetStraz admin clicks the sign-in button 2. enters their email/password 3. If the information is correct they can now log in |
| Alt flow | 1. The PetStraz admin doesn’t have an account. They must create one 2. They enter the wrong password. They have 3 tries, after that, they must reset their password |

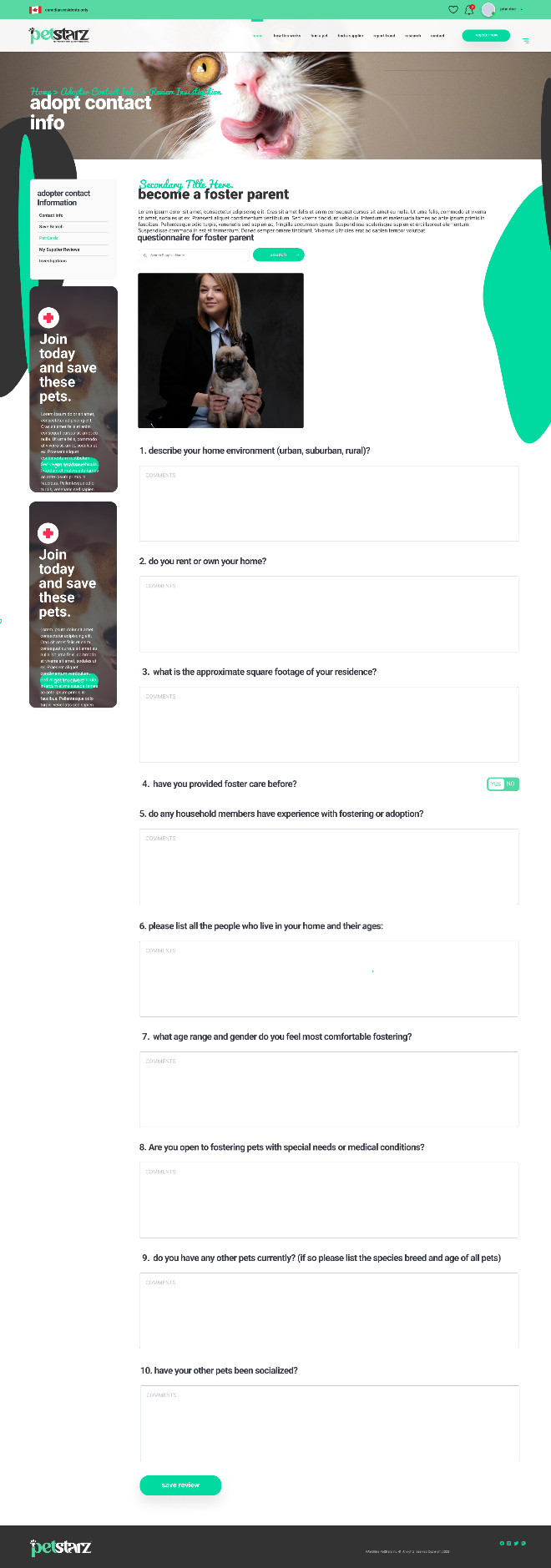
|  |  |
| --- | --- |
| Name | View new or edited reports |
| Actors | PetStraz admin |
| Limitation |  |
| Precondition | Must be logged in as PetStarz admin, |
| Postcondition | Can view or edit all the report that have been made to a supplier |
| Normal flow | The petStarz admin is logged in, goes to profile and clicks on view reports and if they want to edit they can click the edit button on the report they wish to edit |
| Alt flow |  |

|  |  |
| --- | --- |
| Name | Add notes to report |
| Actors | PetStraz admin |
| Limitation |  |
| Precondition | Must be logged in as PetStarz admin, |
| Postcondition | Can add notes to a report |
| Normal flow | The petStarz admin is logged in, goes to profile and clicks on the appropriate report and can add notes on to the report |
| Alt flow |  |

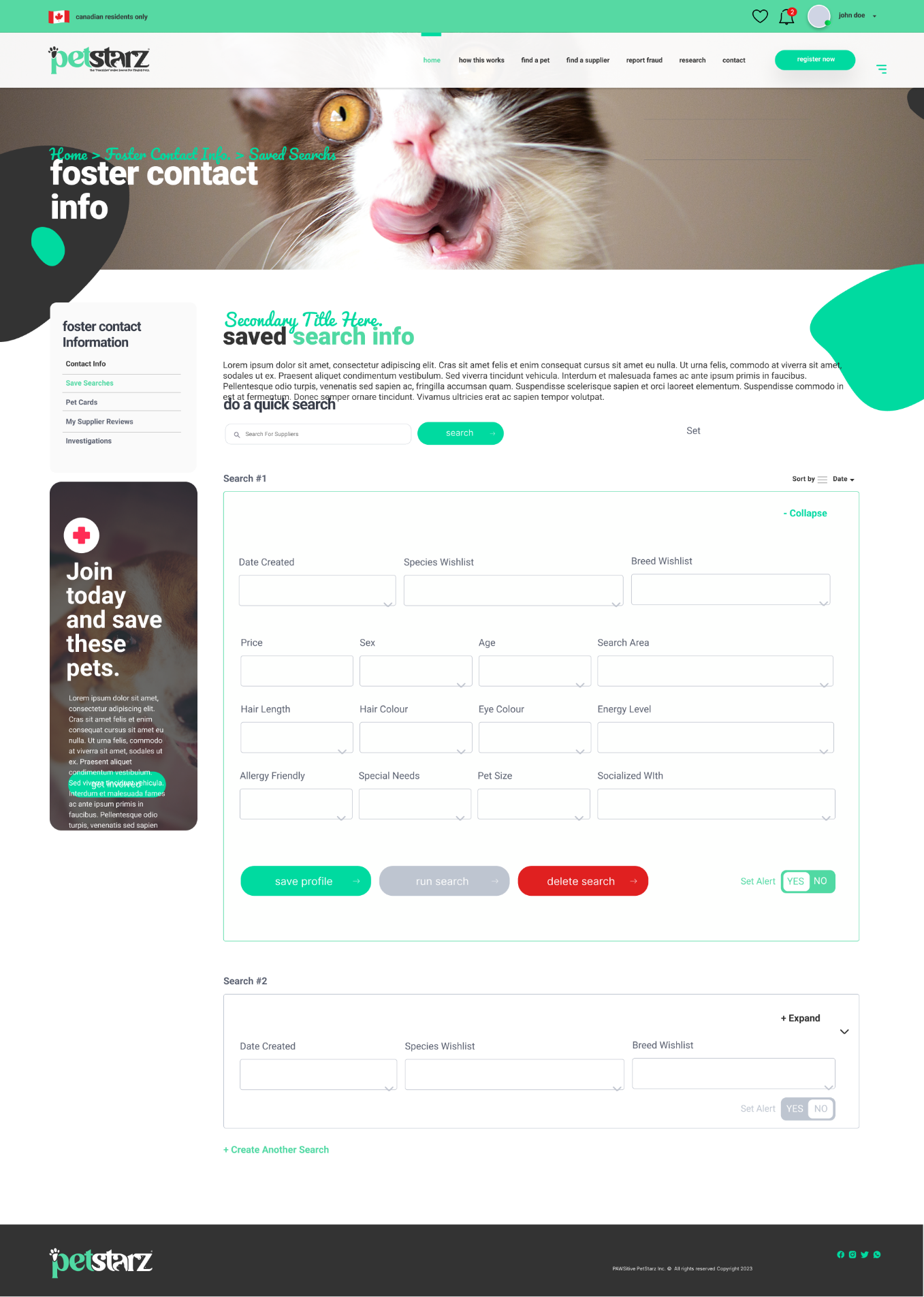
# Preliminary User Interface Design:

Because our client had already designed almost all of the user interface for this project before anything else, we worked with what was given to us to build the sections that were missing. The fostering side of this application is nearly identical to the adoption side (which has already been created), with some minor differences that included a questionnaire for potential fosters to fill out and submit before they could be considered by suppliers. This was what we focused on building, along with altering the previously created adoption pages to work with the fostering side of the application.

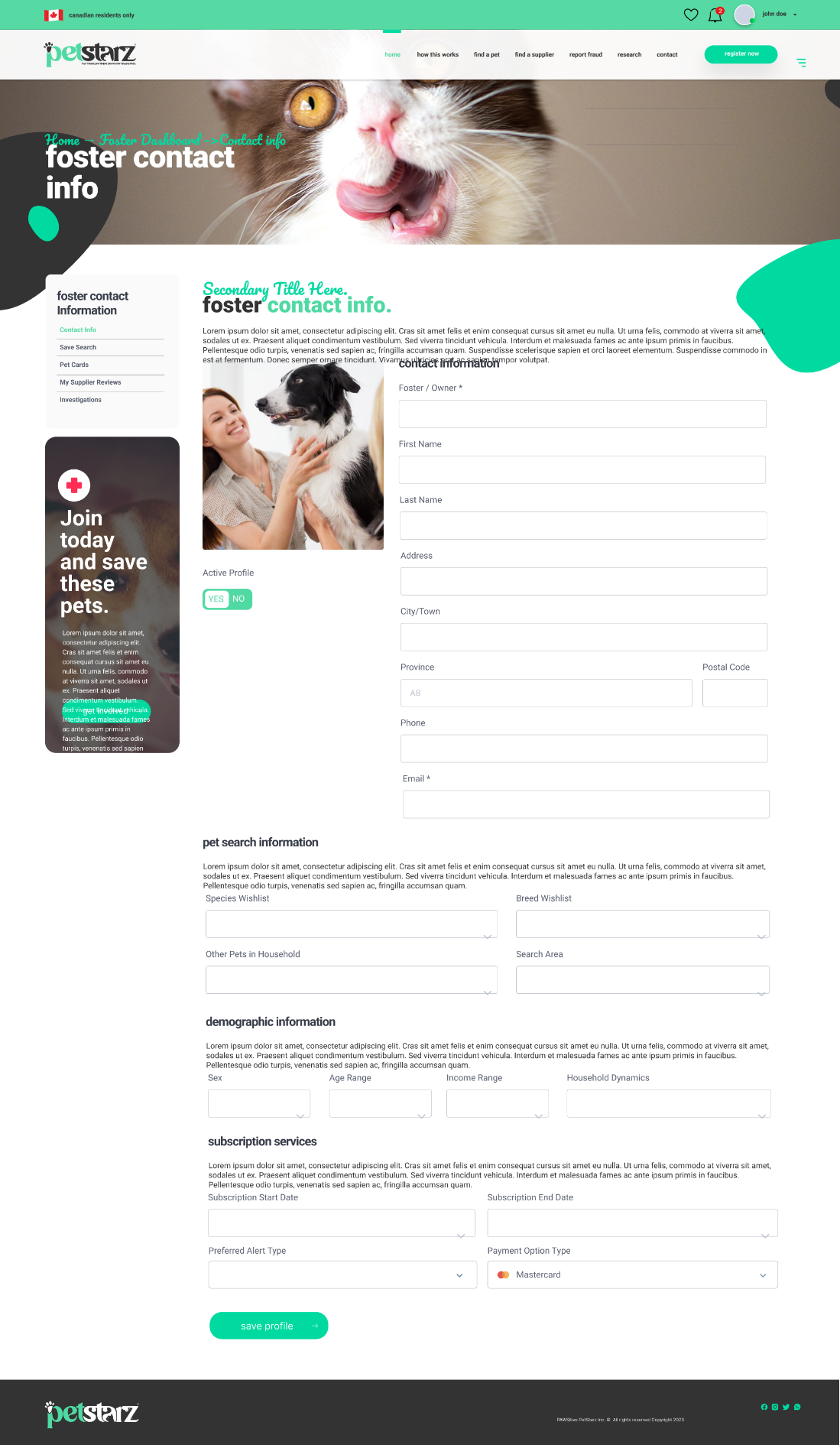
## Pet foster Questionnaire:



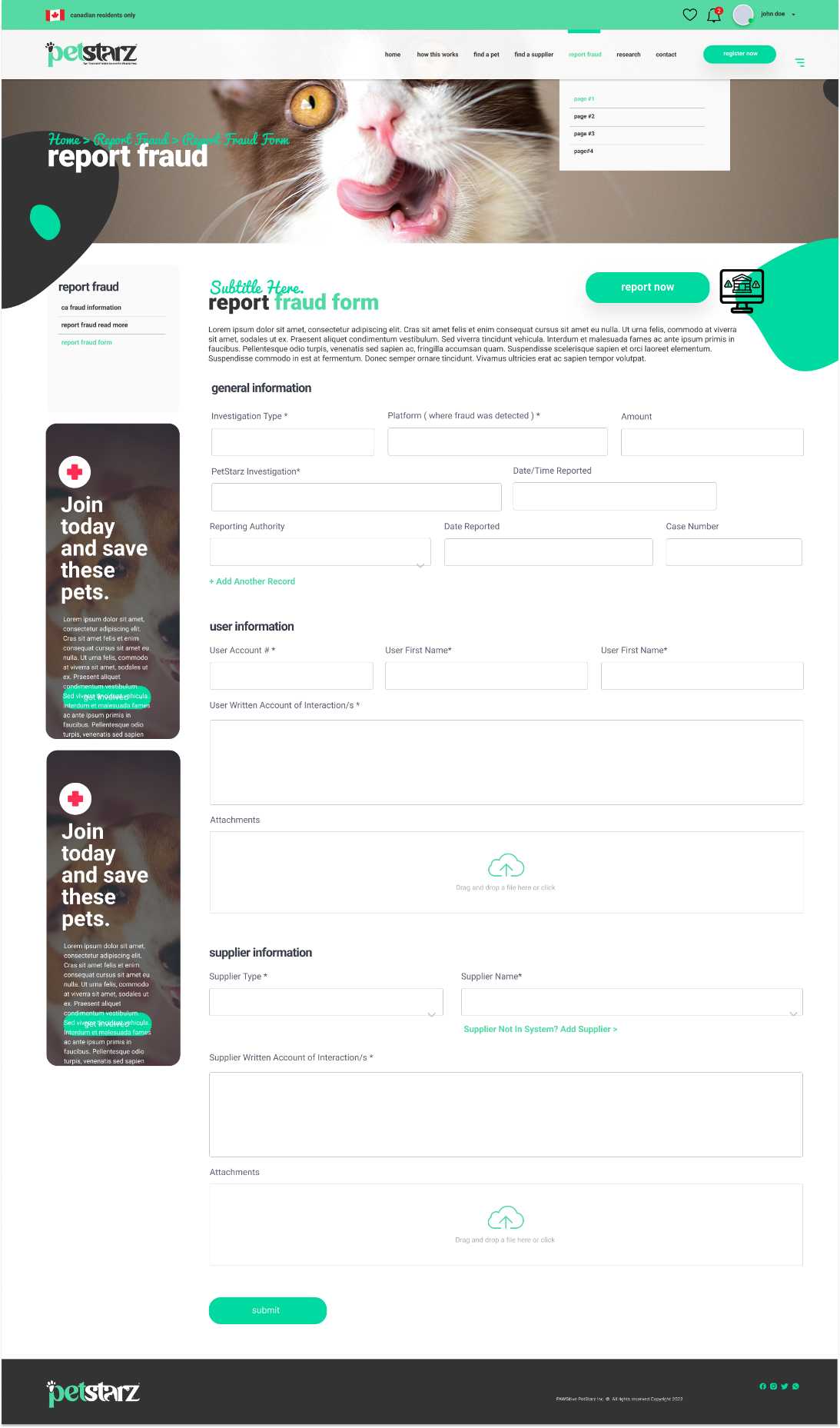
## Saved Search (supplied by client, modified by us)



## Foster Contact (supplied by client, modified by us)



## Report Fraud Form (supplied by client)



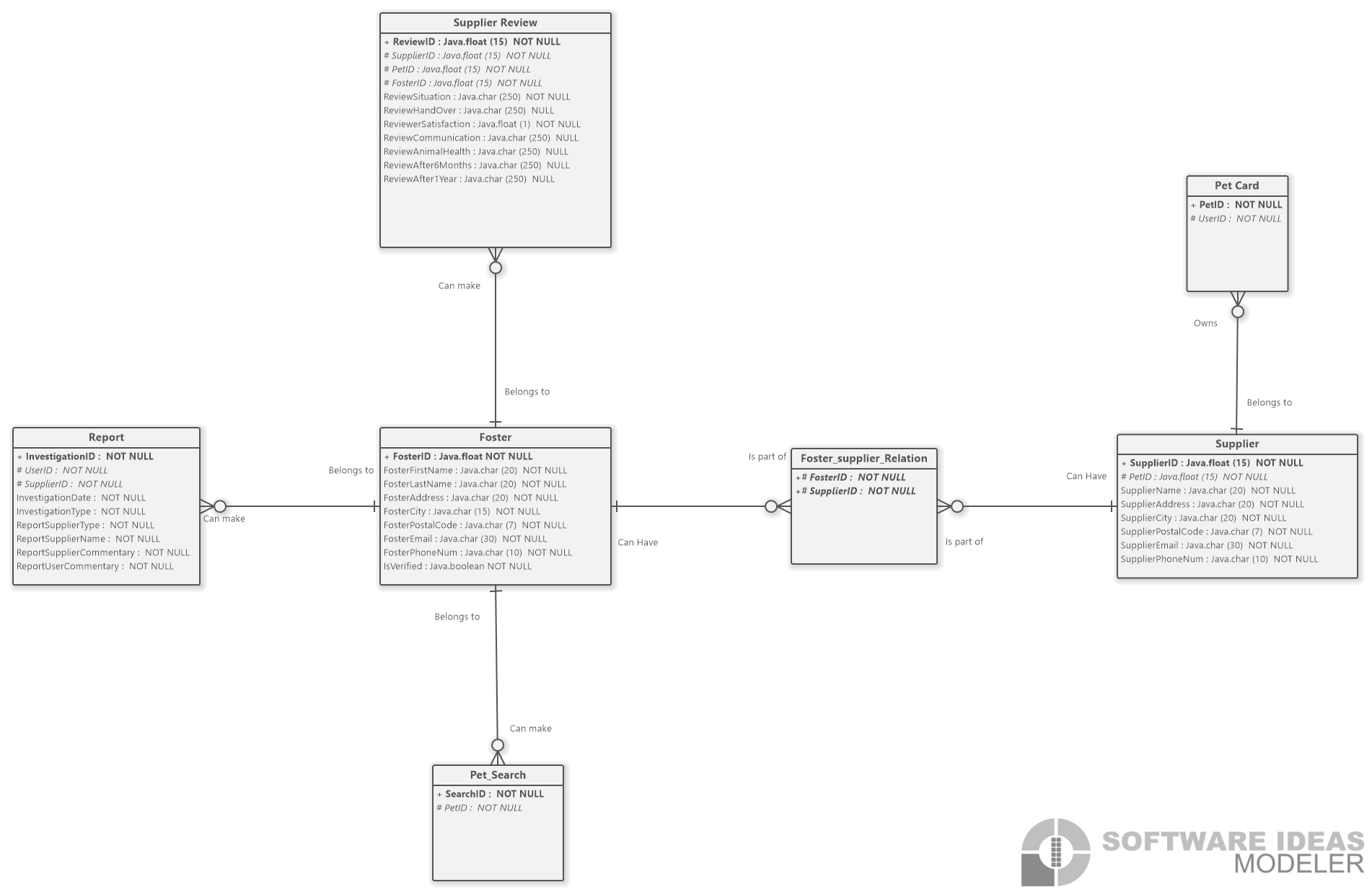
# Data Storage and Persistence:

While this project does require data storage, Our Client has already decided to use MongoDB as their database provider. According to our research, MongoDB is a NoSQL database. This means everything is stored in JSON-like documents. This means that they do not require traditional tables and rows.

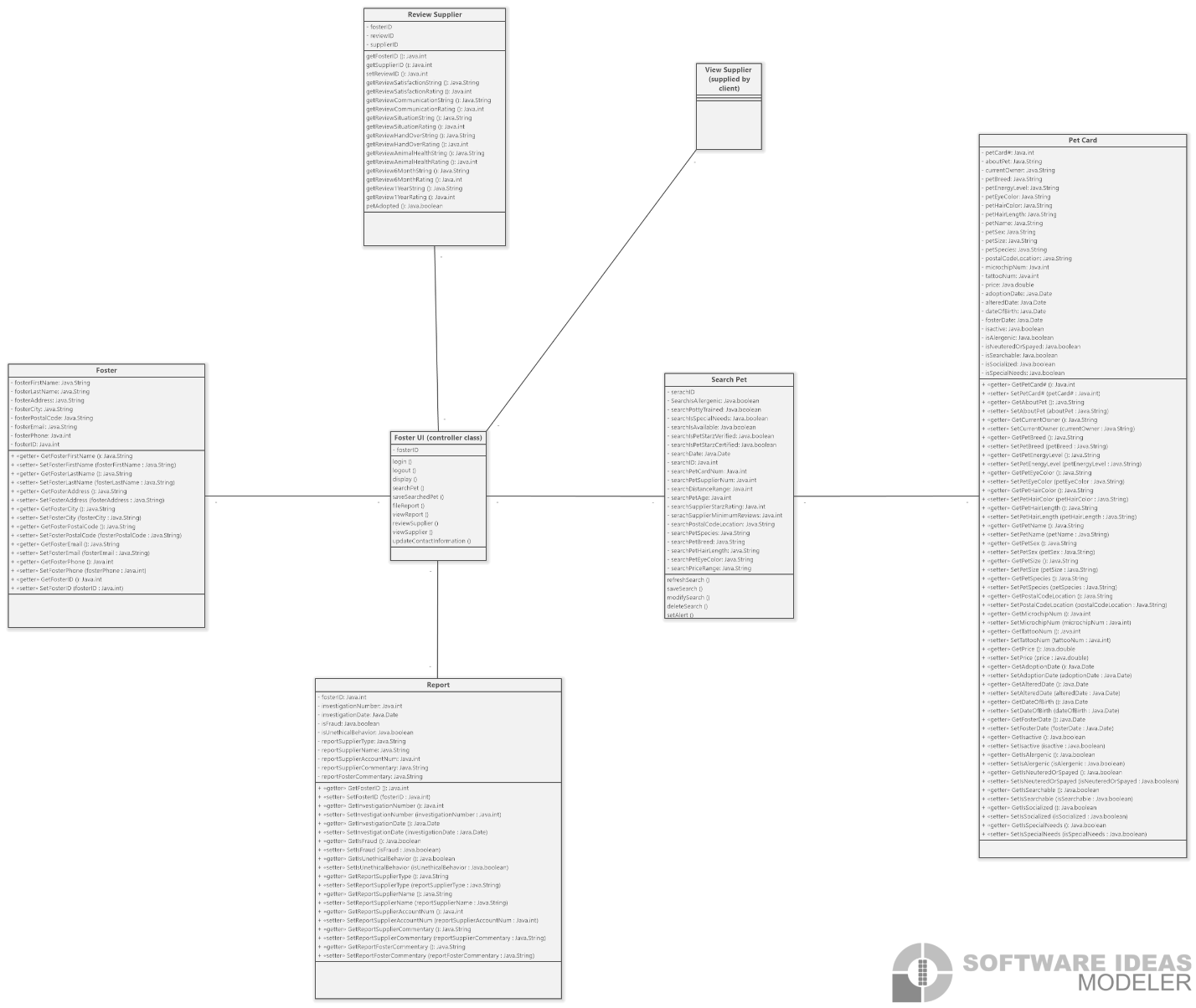
This style of database is very well suited for this project, as it has strengths when dealing with web-development and mobile app development. It is very flexible and has a huge amount of scalability, which is especially important for a project of this size.

Because our documentation requires an ERD, we created one to show how these items would be related in a traditional database, but may be required to translate this ERD into another form of documentation that suits a NoSQL database better.

# ERD:



# Class Diagram:



# System Architecture and Patterns

System architectures and design patterns are both important aspects of Software development. They serve as blueprints for designing the system, solving different problems that arise during the development. It will help insure the development runs smoothly.

System architectures are for the overall structure of the system. It covers how the components within the system should interact with each other but does not define any of the components.

Design patterns are way to describe how the relationships between classes or objects. Design patterns can be categorized into three types: creational, structural or behavioural. They serve as a template for how to structure the classes of the project, solving a design problem.

For out project we will be using the adapter pattern, Microkernel, because what we are building must exist within a larger project it makes the most sense to make sure what we are building is adaptable and can be moved around. We want to make sure that what we are developing can take in further requirements from our sponsor. The Microkernel pattern allows the group members to work individually.

For the creation of our project we will be implementing a prototype design pattern, because it will allow us to create a skeleton of our project so we can see what we like and what works early on. It will allow our group to clearly see what needs to be completed. It will also allow us some flexibility with what gets implemented and since we are working within a larger project it will allow us to change aspects that don’t end up aligning with our sponsor’s vision.

When choosing architectures and design for our project we mainly focused on how to solve the problem of keeping our project flexible to fit into the larger scope of what our sponsor is trying to build.

# Appendix

## Team Constitution

### Team Curious Comets Members

Samantha Bogen

Has previously attended Capilano University for Studio Art, giving them a keen eye for layout and design. They have worked a variety of fast paced customer service jobs that require attention to detail and being able to problem solve in a fast paced environment. They are great at pattern recognition and critical thinking.

Lisa Kamdem

Previously attended Thompson River University where she learned Java programming. Working as a hostess has taught her the value of patience. Lisa is great at being able to break down ideas and explain them in simple ways that anyone can understand.

Dorian Laycock

Went to Red River College for cabinet making where he learned about CNC programming. Being a cabinet maker taught him the importance of taking care at each step of a project to ensure the result is accurate. He is great at keeping his focus and paying attention to the small details.

Markus Luthi

Has acquired a Red Seal Pipe Fitting ticket through SAIT, where he learned how to follow technical drawings and the importance documentation. Working as a pipe fitter taught him a lot about the importance of safety and how to adapt to a situation on the fly. He learned how to work with and manage a group. Markus understands the importance of leadership and keeping a project on track.

Rolan Ho

Attended Plymouth University for Civil Engineering, where he studied Project and Risk Management. Working as a Civil Engineer, Rolan worked on large scale infrastructure projects in Hong Kong where he had to make sure he was looking at the whole picture. He is good at taking in the entire scope of a project and making sure that everything is being considered.

### Statement of Team Goals

As a group we hope to have clear and open communication to help us to solve any problems that arise as a group. We want to finish each phase of the project on time and keep ourselves on a set schedule so we never fall behind. We want to ensure that all Stakeholders are satisfied with the end result of our project.

### Intellectual Property Statement

Refer to Sait policy AC.2.11

### Team Member Roles

Markus is leading the team

Lisa is making schedules for the team to follow, and keeping us on track for deadline.

Samantha is ensuring that all documents follow APA format.

Dorian is editing and formatting the final document.

Rolan is ensuring the document meets the report guidelines.

### Division of Labour

All tasks will be divided equally between the group and when that can not be accomplished the tasks will be done in a rotation so that no one is consistently doing all the work. The group will have weekly meetings to ensure that everyone feels that the work load is being divided fairly.

### Group Expectations

The group will keep in contact over discord and email and expect responses to be within 24 hours. The group will meet weekly during class times to discuss the project and the progress everyone has made.

### Plans for Resolving Conflict

1. If a team member fails to attend a meeting the group will reach out to understand the circumstances and figure out ways that the situation can be avoided in the future. If there is a second meeting that is not attended that as a strike against them.
2. If a team member fails to meet a deadline, the group will ask the teacher to not give them any marks on the assignment and it will count as a strike against them.
3. If a team member is not meeting the standards of the other members and not contributing the same amount to the projects the group will inform the instructor and ask that they receive a lower mark and it will count as a strike against them.
4. If a team member fails to communicate, the group will reach out to make sure that nothing is wrong and offer any sort of support that they can to ensure that is doesn’t happen again. If it is a continuous pattern without any reason given then the group will bring the matter to the teacher and it will count as a strike against them.
5. If there is different levels of commitment to the project the group will discuss it as a whole and make sure everyone is communicating their expectations properly.
6. If there is a personality conflict between members than someone will mediate between the two until a middle ground has been reached.
7. If someone feels that they are doing more work than everyone else the group will discuss how they can redistribute the workload so it is more even.
8. If a team member is doing less work than the other members than the group will discuss how they can redistribute the workload so it is more even.

### Plan if Conflict can not be Resolved

If a team member reaches three strikes against them, showing a pattern, then the group will ask the instructor to intervene and remove them from the group.

### Summary

The group hopes to collaborate in a fair and equitable way so that all labour is divided equally. We want everyone feels like they have a chance to contribute and be heard. We hope that with our varying views we will be able to come together to build something that meets all of out clients needs and truly impresses them.