



**COMSATS University Islamabad (CUI)**

**Software Requirement Specification  
(SRS DOCUMENT)**

**for**

**FelineConnect**

Version 1.0

***By***

**Saifullah CIIT/SP21-BSE-104/ISB**

**Talha Ziaullah CIIT/SP20-BSE-094/ISB**

***Supervisor***

**Mr. Zahid Anwar**

***Bachelor of Science in Software Engineering (2021-2025)***

## Revision History

Name	Date	Reason for Changes	Version

## Application Evaluation History

<b>Comments (by committee)</b> *include the ones given at scope time both in doc and presentation	<b>Action Taken</b>
Add new module/features in the project.	New modoule and features added in the project.
Figures caption missing	Figure caption added.
Grammar and Formatting issues.	Resolved.

**Supervised by**

**Mr. Zahid Anwar**

Signature\_\_\_\_\_

# Table of Contents

<b>1. Introduction.....</b>	<b>5</b>
1.1 Purpose.....	5
1.2 Scope.....	5
1.3 Modules.....	6
1.3.1 Module 1: User Authentication and Profile Management.....	6
1.3.2 Module 2: Cat Breed Detection.....	6
1.3.3 Module 3: Online Store.....	6
1.3.4 Module 4: Search Filters and Notifications.....	7
1.3.5 Module 5: Payment Processing.....	7
1.3.6 Module 6: AI Chatbot.....	7
1.3.7 Module 7: Admin Panel.....	8
1.3.8 Module 8: Statistical Analytics.....	8
1.4 Overview.....	8
<b>2. Overall Description.....</b>	<b>9</b>
2.1 Product Perspective.....	9
2.2 User classes and characteristics.....	10
2.3 Operating Environment.....	10
2.4 Design and Implementation Constraints.....	10
<b>3. Requirement Identifying Technique.....</b>	<b>11</b>
3.1 Use case Diagram.....	11
3.2 Detail Use Case.....	14
3.2.1 Module 1: User Authentication and Profile Management.....	14
3.2.2 Module 2: Cat Breed Detection.....	19
3.2.3 Module 3: Online Store.....	22
3.2.4 Module 4: Search Filters and Notifications.....	24
3.2.5 Module 5: Payment Processing.....	28
3.2.6 Module 6: AI ChatBot.....	29
3.2.7 Module 7: Admin Panel.....	31
3.2.8 Module 8: Statistical Analytics.....	37
3.3 Event-Response Table.....	41
<b>4. Functional Requirements.....</b>	<b>42</b>
4.1 Module 1: User Authentication and Profile Management.....	42
4.2 Module 2: Cat Breed Detection.....	58
4.3 Module 3: Online Store.....	70
4.4 Module 4: Search Filters and Notifications.....	77
4.5 Module 5: Payment Processing.....	78
4.6 Module 6: AI Chatbot.....	88
4.7 Module 7: Admin Panel.....	93
4.8 Module 8: Statistical Analytics.....	97
<b>5. Non-Functional Requirements.....</b>	<b>102</b>
5.1 NFR-1: Reliability.....	102
5.2 NFR-2: Usability.....	102
5.3 NFR-3: Performance.....	103
5.4 NFR-4: Portability.....	103
<b>6. External Interface Requirements.....</b>	<b>104</b>
6.1 User Interfaces Requirements.....	104
6.2 Software interfaces.....	106
6.3 Hardware Interfaces.....	107
6.4 Communications interfaces.....	107
<b>7. References.....</b>	<b>107</b>
<b>8. Plagiarism.....</b>	<b>108</b>

# 1. Introduction

The Software Requirements Specification (SRS) for the cat breed recognition system with an integrated online store, named FelineConnect, aims to provide a comprehensive overview of the project's objectives, scope, modules, and overall structure. This document outlines the requirements and functionalities of FelineConnect, addressing the challenges faced by cat owners in accurately identifying their pets' breeds and sourcing relevant products.

The project's scope encompasses the development of a centralized platform that leverages deep learning models like ResNet-50 for precise breed identification and offers a user-friendly interface for seamless cat food purchases. FelineConnect's key objectives include accurate breed identification, user engagement through a feedback loop, secure online transactions, and accessibility across different devices.

The modules of FelineConnect cover user authentication, cat breed detection using ResNet-50, an online store for buying cat and food products, search filter and notification, payment processing, AI chatbot, statistical analytics and admin panel. This introduction serves as a foundation for understanding the detailed requirements and functionalities outlined in the subsequent sections of the SRS for FelineConnect.

## 1.1 Purpose

The purpose of this Software Requirements Specification (SRS) document is to outline the requirements and specifications for the development of FelineConnect, a comprehensive cat breed recognition system with an integrated online store. It contains details on how the system works, its use cases, functional and non-functional requirements. It also provides an insight into the functionalities and limitations of the system.

## 1.2 Scope

The scope of FelineConnect encompasses the development of a sophisticated cat breed recognition system with an integrated online store. This software aims to address the challenges faced by cat owners in accurately identifying their pets' breeds and sourcing relevant products. The software aligns with user and project goals by providing accurate breed identification, creating a user-friendly interface, implementing an integrated online store, optimizing transactions, and ensuring accessibility across different devices.

Major features include deep learning-based breed identification, image upload for instant results, a user-friendly interface for seamless navigation, an integrated online store for cat food purchases, and a feedback loop for continuous improvement in breed recognition accuracy.

## **1.3 Modules**

### **1.3.1 Module 1: User Authentication and Profile Management**

**FE-1:** Users can create an account by providing personalized information.

**FE-2:** Registered user can access the account by providing information such as email and password.

**FE-3:** Registered user can add their names, profile photo, and bio information.

**FE-4:** Registered user can permanently delete his account.

**FE-5:** Registered user can change existing password.

**FE-6:** Registered user can use forgot password option to recover account access via email.

### **1.3.2 Module 2: Cat Breed Detection**

**FE-1:** Users can upload images of their cats for breed identification.

**FE-2:** Split dataset into training and testing sets for model training.

**FE-3:** Integrate ResNet-50 architecture for accurate cat breed recognition.

**FE-4:** Transfer learning on a pre-trained ResNet-50 model.

**FE-5:** Fine-tune the model on a dataset of domestic cat breeds.

**FE-6:** Display detected cat breed along with confidence score.

**FE-7:** Allow users to provide feedback on the accuracy of breed recognition.

### **1.3.3 Module 3: Online Store**

**FE-1:** Registered users can buy different cat breeds from the store.

**FE-2:** Registered users can buy cat food items from the store.

**FE-3:** Users can add Cat and food items to the shopping cart.

**FE-4:** Users can remove cat and food items from the cart.

### **1.3.4 Module 4: Search Filters and Notifications**

**FE-1:** Implement a search bar for users to find cat breeds within the application.

**FE-2:** Allow users to refine search results using filters like breed type, food category, price range.

**FE-3:** Enable users to sort search results by relevance, price (ascending or descending).

**FE-4:** Allow users to search for specific cat food items.

**FE-5:** Send notifications to users about new products, or order status updates (e.g., order confirmation, delivery).

**FE-6:** Alert users about upcoming maintenance or updates to the app.

### **1.3.5 Module 5: Payment Processing**

**FE-1:** Facilitate cart checkout.

**FE-2:** Integrate a secure payment gateway.

**FE-3:** Confirm orders and provide order details.

**FE-4:** Enable order cancellation or modification within a specific timeframe.

**FE-5:** Provide a review and rating system for purchased food items.

### **1.3.6 Module 6: AI Chatbot**

**FE-1:** Allow users to submit queries or questions.

**FE-2:** Implement a chatbot for quick responses.

**FE-3:** Users can ask questions, provide commands, or engage in conversations with the chatbot.

**FE-4:** The chatbot would be able to generate relevant responses based on the user's input.

**FE-5:** The chatbot would be trained on custom data to provide user with their desired information.

### **1.3.7 Module 7: Admin Panel**

**FE-1:** Allow admin to add, modify, or remove user accounts.

**FE-2:** View and manage cat food inventory in the store.

**FE-3:** Allow the admin to view and manage orders.

**FE-4:** Admin can change delivery status of product to pending or delivered successfully.

**FE-5:** View and manage user feedback on model predictions.

### **1.3.8 Module 8: Statistical Analytics**

**FE-1:** Generate reports on the most popular cat breeds sold in the store.

**FE-2:** Track the performance of individual products in terms of sales, revenue, and customer ratings.

**FE-3:** Implement real-time data dashboards for quick insights into sales performance.

**FE-4:** Monitor inventory levels, track stockouts, and generate alerts for low stock or overstock situations.

## **1.4 Overview**

In the following section of the document, we will explain the overall description of our system, the requirements identifying techniques, also its external interface requirements. It contains an overall knowledge of our system. User classes, use-case diagrams and detailed use cases are also listed in this part. It also includes functional and non-functional requirements of FelineConnect and all the internal and external requirements.

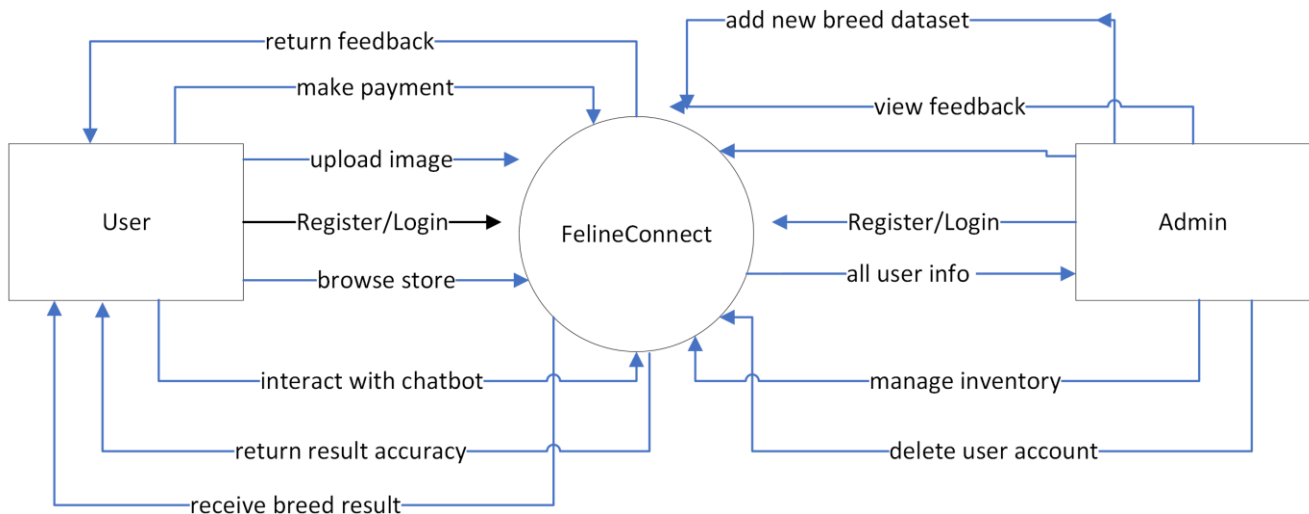


## 2. Overall Description

This section includes a high-level overview of the product and the environment in which it will be used, the anticipated users, and known constraints, assumptions, and dependencies.

### 2.1 Product Perspective

FelineConnect is an innovative solution that represents the next step in cat breed recognition and online shopping for cat-related products. It is an entirely new product that combines advanced deep learning and computer vision techniques with an integrated online store to provide a seamless experience for cat owners. FelineConnect is designed to be the central hub for cat owners, offering precise breed identification, convenient cat food purchases, user-friendly interface, and continuous improvement through feedback loops.



**Figure 2.1: FelineConnect Context Diagram**

## 2.2 User classes and characteristics

**Table 1 : User classes and characteristics**

User Class	Description
<b>Admin</b>	Admins in FelineConnect have administrative privileges and responsibilities. They can manage user accounts, including Users. Admins ensure smooth operation of the platform, address user issues, manage payments, and handle feedback from users, maintaining the platform's functionality.
<b>User</b>	Cat owner are central users of FelineConnect, utilizing the platform to identify their cat's breed accurately and purchase relevant cat food products. They can upload images of their cats, receive instant breed identification results, explore the online store, add items to the cart, make secure payments, and provide feedback on their experiences, contributing to continuous improvement.

## 2.3 Operating Environment

**OE-1:** The system will be developed using localhost fast server REST API to interact with python machine learning capabilities.

**OE-2:** The system “FelineConnect” shall operate on a variety of web browsers including (Opera, Google chrome, Microsoft Edge) and on window operating systems.

## 2.4 Design and Implementation Constraints

**CON-1:** User must have an active internet connection for system to run.

**CON-2:** It is assumed that the user using the system has a basic knowledge of how to run a software.

**CON-3:** The GUI of FelineConnect will be available exclusively in English.

**CON-4:** The user interface of the system shall be implemented using React.

### 3. Requirement Identifying Technique

Following are the requirements identifying techniques.

#### 3.1 Use case Diagram

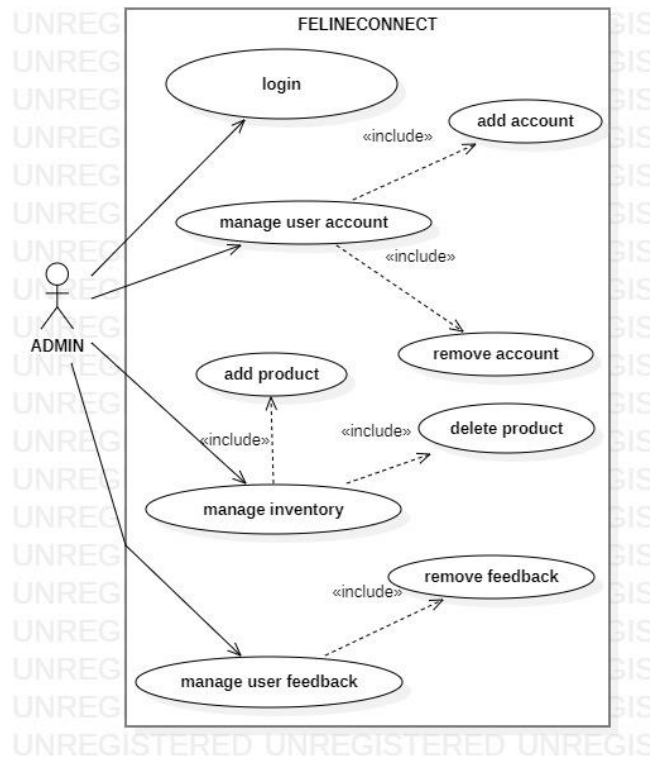
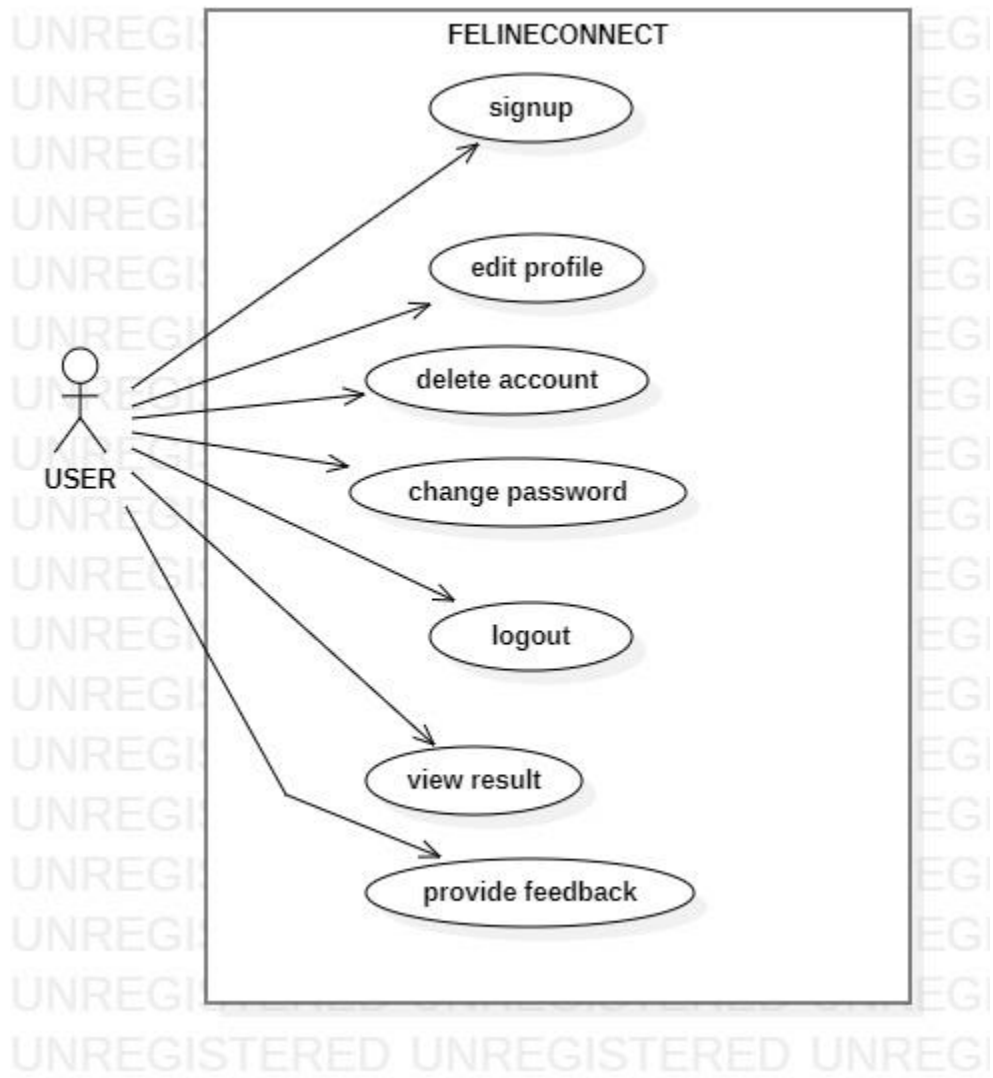
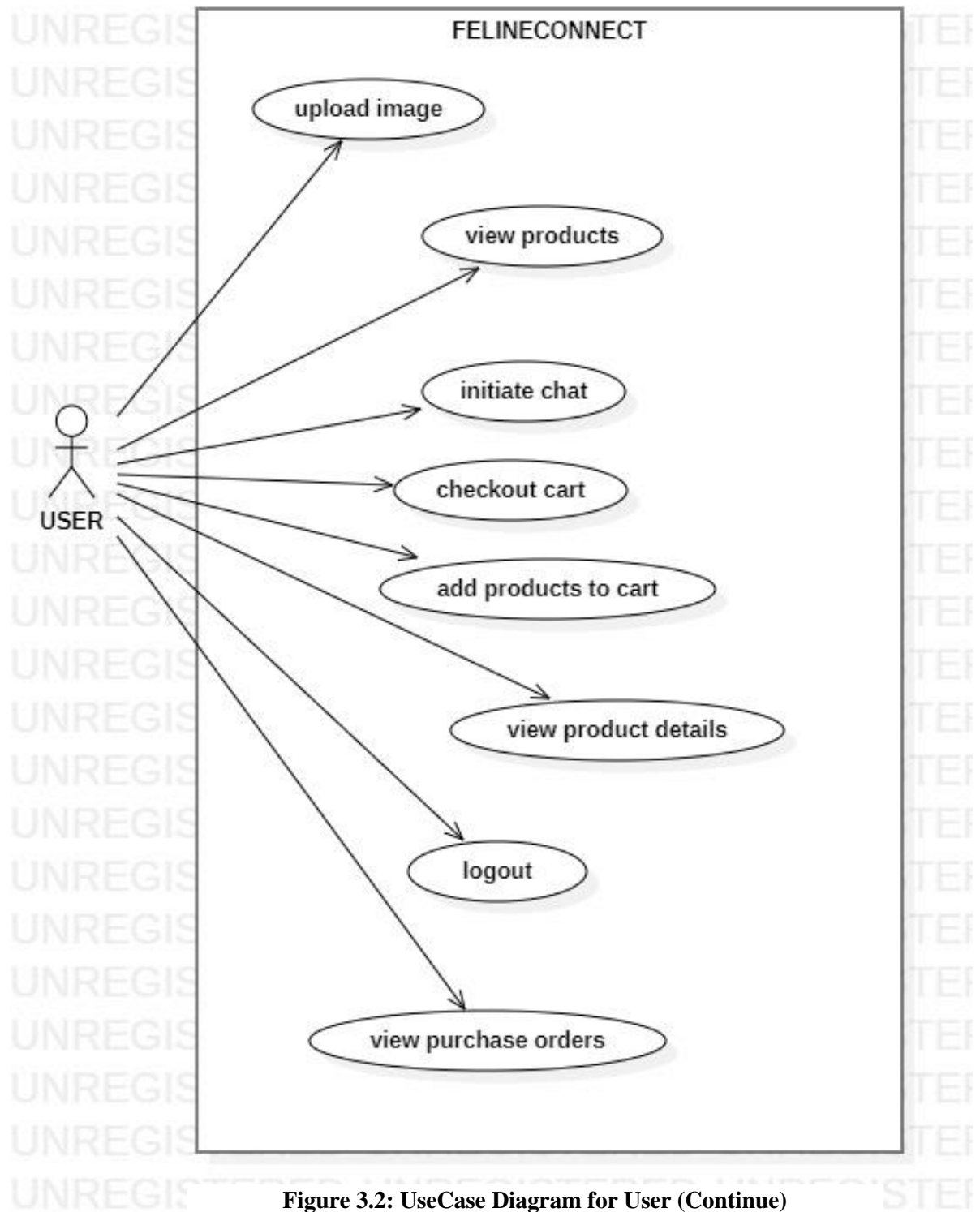


Figure 3.1: UseCase Diagram for Admin



**Figure 3.2: UseCase Diagram for User**



**Figure 3.2: UseCase Diagram for User (Continue)**

## 3.2 Detail Use Case

### 3.2.1 Module 1: User Authentication and Profile Management

#### UC-1: User Signup

<b>Use Case ID:</b>	UC-1
<b>Use Case Name:</b>	User Signup
<b>Actors:</b>	User
<b>Description:</b>	The user will enter their full name, email, password, city, address and phone number during the signup process.
<b>Trigger:</b>	The User presses the Register Button.
<b>Preconditions:</b>	PRE-1. The user must access the signup page. PRE-2. The user should not be logged in.
<b>Postconditions:</b>	POST-1. A Success Message will be displayed, indicating that the user has successfully signed up. POST-2. The user will be navigated to the homepage.
<b>Normal Flow:</b>	<ol style="list-style-type: none"><li>1. System displays the signup form to be filled by the user.</li><li>2. Enter the full name.</li><li>3. Enter a valid email address.</li><li>4. Enter a secure password.</li><li>5. Enter City.</li><li>6. Enter Address.</li><li>7. Enter valid Phone number.</li><li>8. Press the Register button.</li><li>9. A Success Message will be displayed, and the user is taken to the homepage.</li></ol>
<b>Alternative Flows:</b>	<ol style="list-style-type: none"><li>1. The User enters the first name.</li><li>2. The User enters the last name.</li><li>3. The User enters a valid password.</li><li>4. The User presses the register button.</li><li>5. An Error Message will be shown, indicating a lack of network connectivity.</li></ol>
<b>Exceptions:</b>	<ol style="list-style-type: none"><li>1. The User enters incorrect or incomplete first name, last name, or email.</li><li>2. The email provided is already registered.</li><li>3. The password is not secure enough.</li></ol>
<b>Business Rules:</b>	N/A
<b>Assumptions:</b>	<ol style="list-style-type: none"><li>1. The User should understand basic English language to fill out the signup form.</li><li>2. An active internet connection is required to complete the signup process.</li></ol>

## UC-2: Login

<b>Use Case ID:</b>	UC-2
<b>Use Case Name:</b>	Login
<b>Actors:</b>	User
<b>Description:</b>	The user will enter their email, password to login to the system.
<b>Trigger:</b>	The Client presses the Login Button.
<b>Preconditions:</b>	N/A.
<b>Post conditions:</b>	POST-1. User will be logged in to the system POST-2. System redirects the user to homepage
<b>Normal Flow:</b>	<ol style="list-style-type: none"><li>1. System displays the login form.</li><li>2. Enter the email.</li><li>3. Enter the password.</li><li>4. Click the log in button.</li><li>5. User will be logged in to the system.</li><li>6. Success message will be displayed.</li><li>7. The user will be redirected to the homepage.</li></ol>
<b>Alternative Flows:</b>	<ol style="list-style-type: none"><li>1. Enter the email.</li><li>2. Click login button.</li><li>3. Error message will be displayed.</li></ol>
<b>Exceptions:</b>	The user enters incorrect email or password.
<b>Business Rules:</b>	BR-1: all the mandatory fields are required to be filled with correct data.
<b>Assumptions:</b>	The user has already signed up

## UC-3: Signup with google

<b>Use Case ID:</b>	UC-3
<b>Use Case Name:</b>	Signup with Google
<b>Actors:</b>	User
<b>Description:</b>	User can login to the system using third party authentication with Google
<b>Trigger:</b>	Users presses the Sign up with Google icon button.
<b>Preconditions:</b>	N/A
<b>Post conditions:</b>	POST-1. Users will be logged in/ registered into the system. POST-2. The user will be redirected to the homepage.

<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. The system displays the signup form</li> <li>2. User clicks the Sign up with Google icon button.</li> <li>3. The system redirects the user to Google's authentication page.</li> <li>4. User enters their Google credentials.</li> <li>5. User is redirected back to the system.</li> <li>6. The system retrieves user information from Google.</li> <li>7. The user is logged in/registered, and the system may prompt the user to complete their profile</li> </ol>
<b>Alternative Flows:</b>	<ol style="list-style-type: none"> <li>1. User has already signed into Google on browser</li> <li>2. User will click Sign up with Google icon button on the system.</li> <li>3. User will select his account</li> <li>4. User will be logged in/ registered</li> </ol>
<b>Exceptions:</b>	The user enters an incorrect email or password.
<b>Business Rules:</b>	N/A
<b>Assumptions:</b>	<ol style="list-style-type: none"> <li>1. The user has a valid Google account.</li> <li>2. The user is willing to grant access to their Google account.</li> <li>3. The user is on the signup page of the system.</li> <li>4. The user is not already logged into the system.</li> </ol>

#### UC-4: Update user credentials

<b>Use Case ID:</b>	UC-4
<b>Use Case Name:</b>	Update User Credentials
<b>Actors:</b>	User.
<b>Description:</b>	The user be able to update their profiles.
<b>Trigger:</b>	The user presses the update button in the user profile
<b>Preconditions:</b>	PRE-1. The User has logged in to the system.
<b>Post conditions:</b>	POST-1. The User will have successfully updated their credentials.
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. User is logged into the system.</li> <li>2. User navigates to settings.</li> <li>3. User selects "Personal Information."</li> <li>4. User is presented with the option to change their: <ul style="list-style-type: none"> <li>• Name</li> <li>• Address</li> <li>• Phone Number</li> <li>• Password</li> </ul> </li> <li>5. User makes the desired changes.</li> <li>6. User clicks "Save Changes."</li> <li>7. The credentials are updated in the system.</li> </ol>
<b>Alternative Flows:</b>	<ol style="list-style-type: none"> <li>1. User is logged into the system.</li> <li>2. User navigates to settings.</li> <li>3. User selects "Personal Information."</li> <li>4. User is presented with the option to change their:</li> </ol>



	5. Name 6. Address 7. Phone Number 8. Password 9. User makes the desired changes. 10. User clicks "Save Changes." 11. Error message is generated
<b>Exceptions:</b>	The user enters any incorrect data.
<b>Business Rules:</b>	N/A
<b>Assumptions:</b>	1. The user has logged in. 2. Internet connect is available

#### UC-5: Forget password

<b>Use Case ID:</b>	UC-5
<b>Use Case Name:</b>	Forgot Password
<b>Actors:</b>	User
<b>Description:</b>	If the user forgets their password, then they can reset the password to a new one
<b>Trigger:</b>	The user presses the forgot password button
<b>Preconditions:</b>	PRE-1. User account already exist.
<b>Post conditions:</b>	POST-1. Password of the user account will be successfully reset.
<b>Normal Flow:</b>	1. The user will click Forgot Password button 2. The user will provide the email 3. The user will get a recovery link on the email 4. The user will click on the recovery email 5. The user will provide a new password 6. The user will provide confirmation of new password 7. The user will press Save Changes button.
<b>Alternative Flows:</b>	N/A

#### UC-6: Delete user account

<b>Use Case ID:</b>	UC-6
<b>Use Case Name:</b>	Delete User Account
<b>Actors:</b>	User
<b>Description:</b>	User can Delete their account.
<b>Trigger:</b>	The user presses the Delete Account button
<b>Preconditions:</b>	PRE-1. The user has logged in to the account
<b>Post conditions:</b>	POST-1. User account will be successfully deleted POST-2. The user will be redirected to the homepage

<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. The user will login the system</li> <li>2. The user will go to Settings</li> <li>3. The user will select Delete Account Button</li> <li>4. A box will appear the user will enter the account password.</li> <li>5. User will click on Delete button.</li> <li>6. A confirmation Dialogue Box will appear.</li> <li>7. User will select Confirm button.</li> <li>8. The user account has been deleted.</li> <li>9. User will be redirected to the signup page</li> </ol>
<b>Alternative Flows :</b>	<ol style="list-style-type: none"> <li>1. The user will login the system</li> <li>2. The user will go to their profile</li> <li>3. The user will select Delete Account Button</li> <li>4. A box will appear the user will enter incorrect account password.</li> <li>5. User will click on Delete button.</li> <li>6. A message box will appear displaying incorrect password.</li> <li>7. The account will not be deleted</li> </ol>
<b>Exceptions:</b>	
<b>Business Rules:</b>	N/A
<b>Assumptions:</b>	The user has logged in to the system

#### UC-7: Sign-out

<b>Use Case ID:</b>	UC-7
<b>Use Case Name:</b>	Sign out
<b>Actors:</b>	User
<b>Description:</b>	User can log out of the system
<b>Trigger:</b>	Users presses the Log out button
<b>Preconditions:</b>	PRE-1. User is logged in to the system
<b>Post conditions:</b>	POST-1. Users will be logged out of the system
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. User clicks the logout button</li> <li>2. User is logged out</li> <li>3. User is redirected to the homepage</li> </ol>
<b>Alternative Flows:</b>	N/A
<b>Exceptions:</b>	
<b>Business Rules:</b>	N/A
<b>Assumptions:</b>	<ol style="list-style-type: none"> <li>1. The user has already logged in</li> <li>2. The user has working internet connection</li> </ol>

### 3.2.2 Module 2: Cat Breed Detection

#### UC-8: Upload Cat Images for Breed Recognition

<b>Use Case ID</b>	<b>UC-8</b>
<b>Use Case Name</b>	Upload Cat Images for Breed Recognition
<b>Actors</b>	User
<b>Description</b>	Users can upload images of their cats to FelineConnect for breed recognition.
<b>Trigger</b>	User wants to upload cat images for breed identification.
<b>Preconditions</b>	User is logged into their FelineConnect account.
<b>Postconditions</b>	Cat images are successfully uploaded for breed recognition.
<b>Normal Flow</b>	<ol style="list-style-type: none"><li>1. User navigates to the "Image Submission" section.</li><li>2. User selects the option to upload cat images.</li><li>3. User chooses one or more images from their device.</li><li>4. User confirms the image selection and submission. System processes and stores the uploaded images.</li></ol>
<b>Exceptions</b>	Invalid image format, size limit exceeded, or incomplete submission.
<b>Business Rules</b>	N/A
<b>Assumptions</b>	Users have access to their device's storage for image selection.

#### Integrate ResNet-50 for Cat Breed Recognition

<b>Use Case ID</b>	<b>UC-9</b>
<b>Use Case Name</b>	Integrate ResNet-50 for Cat Breed Recognition
<b>Actors</b>	System
<b>Description</b>	System integrates ResNet-50 architecture for accurate cat breed recognition.
<b>Trigger</b>	System initiates integration of ResNet-50 for breed recognition.
<b>Preconditions</b>	ResNet-50 architecture is available for integration.
<b>Postconditions</b>	ResNet-50 is successfully integrated for breed recognition.
<b>Normal Flow</b>	<ol style="list-style-type: none"><li>1. System prepares for integrating ResNet-50 architecture.</li><li>2. System configures the necessary environment for ResNet-50 integration.</li><li>3. System downloads the pre-trained ResNet-50 model.</li><li>4. System integrates ResNet-50 architecture into the FelineConnect system.</li></ol>

<b>Exceptions</b>	None
<b>Business Rules</b>	ResNet-50 integration enhances accuracy in breed recognition.
<b>Assumptions</b>	Sufficient computational resources are available for model integration and training.

### Transfer Learning with ResNet-50 Model

<b>Use Case ID</b>	<b>UC-10</b>
<b>Use Case Name</b>	Perform Transfer Learning with ResNet-50 Model
<b>Actors</b>	System
<b>Description</b>	System performs transfer learning on a pre-trained ResNet-50 model.
<b>Trigger</b>	Pre-trained ResNet-50 model is available for transfer learning.
<b>Preconditions</b>	Pre-trained ResNet-50 model is downloaded and ready for transfer learning.
<b>Postconditions</b>	Transfer learning is successfully applied to the ResNet-50 model.
<b>Normal Flow</b>	<ol style="list-style-type: none"> <li>1. System prepares the pre-trained ResNet-50 model for transfer learning.</li> <li>2. System defines the transfer learning strategy and objectives.</li> <li>3. System executes transfer learning on the pre-trained model using a dataset of domestic cat breeds.</li> </ol>
<b>Exceptions</b>	None
<b>Business Rules</b>	Transfer learning optimizes the ResNet-50 model for cat breed recognition.
<b>Assumptions</b>	Sufficient data and computational resources are available for transfer learning.

### Fine-tune ResNet-50 Model on Cat Breed Dataset

<b>Use Case ID</b>	<b>UC-11</b>
<b>Use Case Name</b>	Perform Fine-tuning on ResNet-50 Model with Cat Breed Dataset
<b>Actors</b>	System
<b>Description</b>	System fine-tunes the ResNet-50 model using a dataset of domestic cat breeds.
<b>Trigger</b>	System initiates fine-tuning of ResNet-50 model on a specific dataset.
<b>Preconditions</b>	ResNet-50 architecture is integrated and ready for fine-tuning.
<b>Postconditions</b>	ResNet-50 model is fine-tuned for improved accuracy in cat breed recognition.
<b>Normal Flow</b>	<ol style="list-style-type: none"> <li>1. System prepares the dataset of domestic cat breeds for fine-tuning.</li> <li>2. System defines the fine-tuning parameters and objectives.</li> <li>3. System executes fine-tuning of the ResNet-50 model using the prepared dataset.</li> </ol>
<b>Exceptions</b>	None
<b>Business Rules</b>	Fine-tuning improves the ResNet-50 model's accuracy and performance on cat breed recognition tasks.
<b>Assumptions</b>	Sufficient computational resources and quality dataset are available for fine-tuning.

### Display Detected Cat Breed with Confidence Score

<b>Use Case ID</b>	<b>UC-14</b>
<b>Use Case Name</b>	Display Detected Cat Breed with Confidence Score
<b>Actors</b>	System
<b>Description</b>	System displays the detected cat breed along with the confidence score.
<b>Trigger</b>	Cat breed recognition is successfully completed.
<b>Preconditions</b>	Cat breed recognition algorithm produces results.
<b>Postconditions</b>	Detected cat breed and confidence score are displayed to the user.
<b>Normal Flow</b>	1. System receives the results from the breed recognition algorithm. 2. System formats and presents the detected cat breed and confidence score on the user interface.
<b>Exceptions</b>	None
<b>Business Rules</b>	Displaying confidence scores helps users assess the accuracy of breed recognition.
<b>Assumptions</b>	Breed recognition algorithm provides accurate results with confidence scores.

### Provide Visual Representation of Key Features

<b>Use Case ID</b>	<b>UC-15</b>
<b>Use Case Name</b>	Provide Visual Representation of Key Features
<b>Actors</b>	System
<b>Description</b>	System provides a visual representation of key features contributing to breed identification.
<b>Trigger</b>	User requests detailed information about breed identification.
<b>Preconditions</b>	Detailed feature analysis data is available.
<b>Postconditions</b>	Visual representation of key features is displayed to the user.
<b>Normal Flow</b>	1. User requests detailed information about the detected cat breed. 2. System retrieves and processes key feature data. 3. System generates a visual representation (e.g., heatmap, feature map) of key features contributing to breed identification.
<b>Exceptions</b>	Key feature data not available or cannot be processed.
<b>Business Rules</b>	Visual representations aid users in understanding the basis of breed identification.
<b>Assumptions</b>	Sufficient data and algorithms are available for generating visual representations of key features.

### Allow Users to Provide Feedback on Accuracy

<b>Use Case ID</b>	<b>UC-16</b>
<b>UseCase Name</b>	Allow Users to Provide Feedback on Accuracy

<b>Actors</b>	User
<b>Description</b>	System allows users to provide feedback on the accuracy of breed recognition.
<b>Trigger</b>	User interacts with the breed detection result interface.
<b>Preconditions</b>	Breed detection results are displayed to the user.
<b>Postconditions</b>	User feedback on accuracy is recorded and processed.
<b>Normal Flow</b>	<ol style="list-style-type: none"> <li>1. User views the detected cat breed and confidence score.</li> <li>2. System provides an option for the user to provide feedback on the accuracy of the detected breed.</li> <li>3. User submits feedback (e.g., rating, comment) through the interface.</li> <li>4. System records and processes the user feedback for analysis.</li> </ol>
<b>Exceptions</b>	User does not provide feedback.
<b>Business Rules</b>	User feedback helps improve the accuracy of breed recognition algorithms.
<b>Assumptions</b>	User interface includes an interactive feedback mechanism for accuracy assessment.

### 3.2.3 Module 3: Online Store

#### Buy cat food items

<b>Use Case ID:</b>	<b>UC-17</b>
<b>Use Case Name:</b>	Buy Cat Food Items
<b>Actors:</b>	Primary Actor: User Secondary Actor: Online Store System
<b>Description:</b>	Users can purchase cat food items from the online store.
<b>Trigger:</b>	User indicates the desire to buy cat food items.
<b>Preconditions:</b>	PRE-1. User is logged into the online store. PRE-2. User has selected cat food items to purchase.
<b>Postconditions:</b>	POST-1. Purchased cat food items are added to the user's shopping cart. POST-2. User can proceed to checkout or continue shopping.
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. User browses cat food items in the online store.</li> <li>2. User adds selected cat food items to the shopping cart.</li> <li>3. User reviews the shopping cart and proceeds to checkout.</li> <li>4. Online store system confirms the purchase and updates the shopping cart.</li> <li>5. User completes the payment process.</li> </ol>
<b>Alternative Flows:</b>	Ex-1. If the selected cat food items are out of stock, the online store system notifies the user and removes them from the cart. Ex-2. If the payment process fails, the user is prompted to retry or use a different payment method.
<b>Business Rules:</b>	BR-1. Cat food items must be in stock to be added to the shopping cart. BR-2. Payment must be successfully processed before completing the purchase.
<b>Assumptions:</b>	Assume that users have registered accounts and payment methods set up in the online store system.

### Add food items to the shopping cart

<b>Use Case ID:</b>	<b>UC-18</b>
<b>Use Case Name:</b>	Add Items to Shopping Cart
<b>Actors:</b>	Primary Actor: User Secondary Actor: Online Store System
<b>Description:</b>	Users can add cat food items to the shopping cart for purchase.
<b>Trigger:</b>	User selects cat food items and chooses to add them to the cart.
<b>Preconditions:</b>	PRE-1. User is logged into the online store. PRE-2. Cat food items are available for purchase.
<b>Postconditions:</b>	POST-1. Selected cat food items are added to the shopping cart. POST-2. User can proceed to checkout or continue shopping.
<b>Normal Flow:</b>	1. User selects cat food items to add to the shopping cart. 2. Online store system adds selected items to the cart. 3. User reviews the shopping cart.
<b>Alternative Flows:</b>	None
<b>Exceptions:</b>	Ex-1. If the selected cat food items are out of stock, the online store system notifies the user and prevents adding them to the cart.
<b>Business Rules:</b>	BR-1. Cat food items must be in stock to be added to the shopping cart.

### Remove food items from the cart

<b>Use Case ID:</b>	<b>UC-19</b>
<b>Use Case Name:</b>	Remove Items from Shopping Cart
<b>Actors:</b>	Primary Actor: User Secondary Actor: Online Store System
<b>Description:</b>	Users can remove cat food items from the shopping cart before completing the purchase.
<b>Trigger:</b>	User selects cat food items in the cart and chooses to remove them.
<b>Preconditions:</b>	PRE-1. User is logged into the online store. PRE-2. Cat food items are in the shopping cart.
<b>Postconditions:</b>	POST-1. Selected cat food items are removed from the shopping cart. POST-2. User can continue shopping or proceed to checkout.
<b>Normal Flow:</b>	1. User views the shopping cart with selected cat food items. 2. User selects items to remove from the cart. 3. Online store system updates the cart accordingly.
<b>Alternative Flows:</b>	None
<b>Exceptions:</b>	None
<b>Business Rules:</b>	BR-1. User can only remove items that are currently in the shopping cart.

### Provide a review for purchased food items

<b>Use Case ID:</b>	<b>UC-20</b>
<b>Use Case Name:</b>	Review and Rate Purchased Items
<b>Actors:</b>	Primary Actor: User Secondary Actor: Online Store System
<b>Description:</b>	Users can review and rate cat food items after purchasing them from the online store.
<b>Trigger:</b>	User completes the purchase of cat food items.
<b>Preconditions:</b>	PRE-1. User is logged into the online store. PRE-2. User has purchased cat food items.
<b>Postconditions:</b>	POST-1. User submits a review and rating for the purchased items. POST-2. Online store system records the review and rating for the items.
<b>Normal Flow:</b>	1. User completes the purchase of cat food items. 2. Online store system prompts user to review and rate the purchased items. 3. User provides a review and rating.
<b>Alternative Flows:</b>	None
<b>Exceptions:</b>	None
<b>Business Rules:</b>	BR-1. Users can only review and rate items they have purchased.

### 3.2.4 Module 4: Search Filters and Notifications

#### FE-1: Implement a Search Bar for Cat Breeds

<b>Use Case ID</b>	<b>UC-21</b>
<b>Use Case Name</b>	Implement Search Bar for Cat Breeds
<b>Actors</b>	User
<b>Description</b>	Users can use a search bar to find specific cat breeds within the application.
<b>Trigger</b>	User wants to search for a particular cat breed.
<b>Preconditions</b>	User is logged into their FelineConnect account.
<b>Postconditions</b>	Search results for the specified breed are displayed.
<b>Normal Flow</b>	1. User navigates to the search bar section.2. User types the name of the cat breed into the search bar.3. User submits the search query.4. System processes the query and retrieves matching results.5. Search results are displayed to the user.
<b>Exceptions</b>	No results found for the search query.
<b>Business Rules</b>	Search should be responsive and provide relevant results quickly.
<b>Assumptions</b>	The system has a comprehensive database of cat breeds.



### Refine Search Results Using Filters

Use Case ID	UC-22
Use Case Name	Refine Search Results Using Filters
Actors	User
Description	Users can refine their search results using various filters like breed type, food category, and price range.
Trigger	User wants to narrow down search results.
Preconditions	User has initiated a search query.
Postconditions	Filtered search results are displayed.
Normal Flow	1. User views the search results page.2. User selects desired filters (breed type, food category, price range).3. User applies the selected filters.4. System processes the filter criteria.5. Filtered results are displayed to the user.
Exceptions	No results match the applied filters.
Business Rules	Filters should be clear and easy to use.
Assumptions	The system provides a variety of relevant filters for user selection.

### FE-3: Sort Search Results by Relevance or Price

Use Case ID	UC-23
Use Case Name	Sort Search Results by Relevance or Price
Actors	User
Description	Users can sort search results by relevance or price (ascending or descending).
Trigger	User wants to organize search results for easier viewing.
Preconditions	User has search results displayed.
Postconditions	Sorted search results are displayed.
Normal Flow	1. User views the search results page.2. User selects the sorting criteria (relevance, price ascending, price descending).3. User applies the sorting criteria.4. System processes the sorting request.5. Sorted results are displayed to the user.
Exceptions	None
Business Rules	Sorting options should be clear and intuitive.
Assumptions	The system has capabilities to sort data efficiently.

### FE-4: Search for Specific Cat Food Items

Use Case ID	UC-24
Use Case Name	Search for Specific Cat Food Items
Actors	User
Description	Users can search for specific cat food items using the search bar.
Trigger	User wants to find a specific cat food item.
Preconditions	User is logged into their FelineConnect account.
Postconditions	Search results for the specified cat food item are displayed.

<b>Normal Flow</b>	1. User navigates to the search bar section.2. User types the name of the cat food item into the search bar.3. User submits the search query.4. System processes the query and retrieves matching results.5. Search results are displayed to the user.
<b>Exceptions</b>	No results found for the search query.
<b>Business Rules</b>	Search should be responsive and provide relevant results quickly.
<b>Assumptions</b>	The system has a comprehensive database of cat food items.

#### FE-5: Send Notifications About New Products or Order Status

<b>Use Case ID</b>	<b>UC-25</b>
<b>Use Case Name</b>	Send Notifications About New Products or Order Status
<b>Actors</b>	System, User
<b>Description</b>	The system sends notifications to users about new products or updates on their order status.
<b>Trigger</b>	A new product is added or there is an update in order status.
<b>Preconditions</b>	User has enabled notifications.
<b>Postconditions</b>	User receives a notification.
<b>Normal Flow</b>	1. A new product is added to the store or an order status is updated.2. System generates a notification.3. System sends the notification to the user's device.4. User receives and views the notification.
<b>Exceptions</b>	Notification delivery failure.
<b>Business Rules</b>	Notifications should be timely and relevant.
<b>Assumptions</b>	Users have notifications enabled and their devices are connected to the internet.

#### FE-6: Alert Users About Maintenance or App Updates

<b>Use Case ID</b>	<b>UC-26</b>
<b>Use Case Name</b>	Alert Users About Maintenance or App Updates
<b>Actors</b>	System, User
<b>Description</b>	The system alerts users about upcoming maintenance or updates to the app.
<b>Trigger</b>	Scheduled maintenance or new app updates are planned.
<b>Preconditions</b>	User has enabled notifications.
<b>Postconditions</b>	User receives an alert notification.
<b>Normal Flow</b>	1. Maintenance or app update is scheduled.2. System generates an alert notification.3. System sends the alert notification to the user's device.4. User receives and views the alert notification.
<b>Exceptions</b>	Notification delivery failure.
<b>Business Rules</b>	Alerts should be sent with sufficient notice.
<b>Assumptions</b>	Users have notifications enabled and their devices are connected to the internet.

#### FE-7: Save User Preferences for Notifications

<b>Use Case ID</b>	<b>UC-27</b>
--------------------	--------------

<b>Use Case Name</b>	Save User Preferences for Notifications
<b>Actors</b>	User, System
<b>Description</b>	Users can save their preferences for receiving different types of notifications.
<b>Trigger</b>	User wants to customize their notification settings.
<b>Preconditions</b>	User is logged into their FelineConnect account.
<b>Postconditions</b>	User preferences for notifications are saved.
<b>Normal Flow</b>	1. User navigates to the notification settings section.2. User selects their preferred notification types (e.g., product updates, order status, maintenance alerts).3. User saves the selected preferences.4. System records the preferences in the user's profile.5. System sends notifications based on saved preferences.
<b>Exceptions</b>	Preferences not saved due to system error.
<b>Business Rules</b>	User preferences should be easy to update and save.
<b>Assumptions</b>	The system can handle and store individual user preferences.

#### FE-8: View Notification History

<b>Use Case ID</b>	<b>UC-28</b>
<b>Use Case Name</b>	View Notification History
<b>Actors</b>	User
<b>Description</b>	Users can view their notification history within the application.
<b>Trigger</b>	User wants to review past notifications.
<b>Preconditions</b>	User is logged into their FelineConnect account.
<b>Postconditions</b>	User views a list of past notifications.
<b>Normal Flow</b>	1. User navigates to the notification history section.2. System retrieves past notifications from the database.3. Past notifications are displayed to the user.<4. User reviews the notification history.
<b>Exceptions</b>	No past notifications found.
<b>Business Rules</b>	Notification history should be displayed in a user-friendly manner.
<b>Assumptions</b>	The system stores and maintains a log of all notifications.

#### FE-9: Implement Push Notifications

<b>Use Case ID</b>	<b>UC-29</b>
<b>Use Case Name</b>	Implement Push Notifications
<b>Actors</b>	System, User
<b>Description</b>	The system sends push notifications to users' devices about new updates or alerts.
<b>Trigger</b>	A new update or alert is generated by the system.
<b>Preconditions</b>	User has enabled push notifications.
<b>Postconditions</b>	User receives a push notification.
<b>Normal Flow</b>	1. A new update or alert is generated by the system.2. System creates a push notification. 3. System sends the push notification to the user's device.4. User receives and views the push notification.
<b>Exceptions</b>	Push notification delivery failure.

<b>Business Rules</b>	Push notifications should be immediate and relevant.
<b>Assumptions</b>	Users have push notifications enabled and their devices are connected to the internet.

### FE-10: Provide In-App Notifications

<b>Use Case ID</b>	<b>UC-30</b>
<b>Use Case Name</b>	Provide In-App Notifications
<b>Actors</b>	System, User
<b>Description</b>	The system provides in-app notifications for users about updates, new products, and alerts.
<b>Trigger</b>	A new update or alert is generated by the system.
<b>Preconditions</b>	User is logged into their FelineConnect account.
<b>Postconditions</b>	User sees an in-app notification.
<b>Normal Flow</b>	1. A new update or alert is generated by the system.2. System creates an in-app notification.3. System displays the in-app notification to the user.4. User sees and interacts with the in-app notification.
<b>Exceptions</b>	None
<b>Business Rules</b>	In-app notifications should be prominent and actionable.
<b>Assumptions</b>	Users regularly interact with the app and can see notifications.

## 3.2.5 Module 5: Payment Processing

### Facilitate cart checkout

<b>Use Case ID:</b>	<b>UC-30</b>
<b>Use Case Name:</b>	Cart Checkout
<b>Actors:</b>	Primary Actor: User Secondary Actor: Payment System
<b>Description:</b>	Users can proceed with the checkout process to complete their purchase.
<b>Trigger:</b>	User indicates the desire to checkout the items in the shopping cart.
<b>Preconditions:</b>	PRE-1. User is logged into the online store. PRE-2. User has items in the shopping cart.
<b>Postconditions:</b>	POST-1. User completes the checkout process and receives confirmation of the order.
<b>Normal Flow:</b>	1. User views the items in the shopping cart and selects checkout. 2. Online store system prompts user for shipping and payment information. 3. User enters shipping details and selects payment method. 4. Online store system processes payment and confirms the order.
<b>Alternative Flows:</b>	None
<b>Exceptions:</b>	Ex-1. If payment authorization fails, user is prompted to retry or use a different payment method.

<b>Business Rules:</b>	BR-1. User must provide valid shipping and payment information to complete checkout.
------------------------	--

### Integrate a secure payment gateway

<b>Use Case ID:</b>	<b>UC-31</b>
<b>Use Case Name:</b>	Secure Payment Integration
<b>Actors:</b>	Primary Actor: System Secondary Actor: Payment Gateway
<b>Description:</b>	Online store system integrates with a secure payment gateway for processing payments.
<b>Trigger:</b>	User initiates payment during checkout process.
<b>Preconditions:</b>	PRE-1. User has completed the checkout process.
<b>Postconditions:</b>	POST-1. Payment gateway confirms payment authorization.
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. Online store system sends payment request to the payment gateway.</li> <li>2. Payment gateway processes payment authorization.</li> <li>3. Payment gateway sends confirmation to online store system.</li> </ol>
<b>Alternative Flows:</b>	None
<b>Exceptions:</b>	Ex-1. If payment authorization fails, online store system notifies user and prompts for alternative payment method.
<b>Business Rules:</b>	BR-1. Payment gateway must be integrated securely to ensure transaction security.

### Confirm orders and provide order details

<b>Use Case ID:</b>	<b>UC-32</b>
<b>Use Case Name:</b>	Order Confirmation and Details
<b>Actors:</b>	Primary Actor: System Secondary Actor: User
<b>Description:</b>	Online store system confirms user orders and provides order details.
<b>Trigger:</b>	Payment authorization is successful.
<b>Preconditions:</b>	PRE-1. User has completed the checkout process. PRE-2. Payment authorization is successful.
<b>Postconditions:</b>	POST-1. User receives confirmation of the order and order details.
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. Online store system generates order confirmation with order details.</li> <li>2. User receives order confirmation via email or on the website.</li> </ol>
<b>Alternative Flows:</b>	None
<b>Exceptions:</b>	None
<b>Business Rules:</b>	BR-1. Order confirmation must include detailed order information such as items purchased, shipping details, and payment summary.

## 3.2.6 Module 6: AI ChatBot

## Initiate Chat Session

<b>Use Case ID:</b>	<b>UC-33</b>
<b>Use Case Name:</b>	Initiate Chat
<b>Actors:</b>	User
<b>Trigger:</b>	User initiates a conversation with the AI chatbot.
<b>Preconditions:</b>	PRE-1. The user must be logged in to the system
<b>Post conditions:</b>	None
<b>Normal Flow:</b>	<ol style="list-style-type: none"><li>1. User initiates a conversation with the AI chatbot</li><li>2. User asks questions, provides commands, or engages in conversation.</li><li>3. The system processes the users input and generates response</li></ol>
<b>Alternative Flows</b>	N/A
<b>Exceptions:</b>	N/A
<b>Business Rules:</b>	BR-1: The chatbot should provide accurate, up-to-date information from authoritative sources. BR-2: User interactions with the chatbot are logged and may be used for analysis and system improvement
<b>Assumptions:</b>	<ol style="list-style-type: none"><li>1. The User must understand basic English language.</li><li>2. Internet connection is available.</li></ol>

## Close a chat session

<b>Use Case ID:</b>	<b>UC-34</b>
<b>Use Case Name:</b>	Close a chat session
<b>Actors:</b>	User
<b>Trigger:</b>	User wants to end the chat session with the chatbot
<b>Preconditions:</b>	PRE-1. User is currently engaged in a chat session
<b>Post conditions:</b>	POST-1. Chat session is successfully closed
<b>Normal Flow:</b>	<ol style="list-style-type: none"><li>1. User initiates the request to close the chat session by expressing their intent to end the conversation.</li><li>2. The chatbot ask for confirmation of the closing of chat session</li><li>3. The User confirm to close the chat session</li><li>4. The chatbot closes the chat session.</li></ol>
<b>Alternative Flows</b>	
<b>Exceptions:</b>	N/A
<b>Business Rules:</b>	BR-1: The chatbot should provide accurate, up-to-date information from authoritative sources.

	BR-2: User interactions with the chatbot are logged and may be used for analysis and system improvement
<b>Assumptions:</b>	3. The User must understand basic English language. 4. Internet connection is available.

### 3.2.7 Module 7: Admin Panel

#### View Dashboard

<b>Use Case ID:</b>	<b>UC-35</b>
<b>Use Case Name:</b>	View dashboard
<b>Actors:</b>	Admin
<b>Description:</b>	The admin will be able to view dashboard
<b>Trigger:</b>	The user pressed Dashboard option from sidebar.
<b>Preconditions:</b>	N/A
<b>Post conditions:</b>	POST-1. Admin successfully views the dashboard
<b>Normal Flow:</b>	1. Admin will Login. 2. Admin will navigate to dashboard from the sidebar. 3. The system will display the dashboard to the admin.
<b>Alternative Flows</b>	N/A
<b>Exceptions:</b>	N/A
<b>Business Rules:</b>	N/A
<b>Assumptions:</b>	1. The admin should understand basic English language to fill out the signup form. 2. The admin is signed up. 3. The admin should have an active internet connection.

#### Add another admin

<b>Use Case ID:</b>	<b>UC-36</b>
<b>Use Case Name:</b>	Add another Admin
<b>Actors:</b>	Admin
<b>Description:</b>	The admin will be able to add another admin
<b>Trigger:</b>	The admin presses the Register Button.
<b>Preconditions:</b>	N/A
<b>Post conditions:</b>	POST-1. A Success Message will be shown which means admin is successfully signed in and they will be navigated to the homepage.

	POST-2. An Error Message will be shown if there is something wrong in the fields.
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. Admin will Login.</li> <li>2. Admin will navigate to add Another Admin button from the sidebar.</li> <li>3. The admin will enter the first name</li> <li>4. The admin will enter the last name</li> <li>5. The admin will enter the email</li> <li>6. The admin will enter the password.</li> <li>7. Press the register button</li> <li>8. A Success Message will be shown.</li> </ol>
<b>Alternative Flows</b>	<ol style="list-style-type: none"> <li>1. The admin will enter the first name</li> <li>2. The admin will enter the last name</li> <li>3. The admin will enter his password.</li> <li>4. The admin will press the register button</li> <li>5. An Error Message will be shown.</li> </ol>
<b>Exceptions:</b>	The admin enters incorrect first name, last name, email.
<b>Business Rules:</b>	N/A
<b>Assumptions:</b>	<ol style="list-style-type: none"> <li>1. The admin should understand basic English language to fill out the signup form</li> <li>2. The admin should have an active internet connection.</li> </ol>

## Admin login

<b>Use Case ID:</b>	<b>UC-37</b>
<b>Use Case Name:</b>	Admin Login
<b>Actors:</b>	Admin
<b>Description:</b>	The admin will be required to enter their email, password to login to the system.
<b>Trigger:</b>	The admin presses the Login Button.
<b>Preconditions:</b>	N/A
<b>Post conditions:</b>	POST-1. Admin will be logged in to the system
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. System displays the login form to be filled by the user.</li> <li>2. Enter the email.</li> <li>3. Enter the password.</li> <li>4. Press the register button.</li> <li>5. A success message is displayed.</li> <li>6. User will be logged in to the system</li> <li>7. The admin is redirected to the dashboard</li> </ol>
<b>Alternative Flows</b>	N/A
<b>Exceptions:</b>	The admin enters incorrect email or password.



<b>Business Rules:</b>	N/A
<b>Assumptions:</b>	The admin has already signed up

### View specific user information

<b>Use Case ID:</b>	<b>UC-38</b>
<b>Use Case Name:</b>	View Specific User Information
<b>Actors:</b>	Admin
<b>Description:</b>	The admin will be able to view the user information like name, emails, feedbacks etc.
<b>Trigger:</b>	The admin presses the View Details button
<b>Preconditions:</b>	N/A
<b>Post conditions:</b>	POST-1. Admin will be able to view the specific user information
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. Admin logs in</li> <li>2. Admin selects the View Users menu from the navbar.</li> <li>3. List of all the users will be displayed.</li> <li>4. The admin will click on a specific user details button.</li> <li>5. All the users are displayed.</li> </ol>
<b>Alternative Flows</b>	N/A
<b>Exceptions:</b>	No user account exists
<b>Business Rules:</b>	N/A
<b>Assumptions:</b>	The admin has already logged in

### View all users list

<b>Use Case ID:</b>	<b>UC-39</b>
<b>Use Case Name:</b>	View All Users List
<b>Actors:</b>	Admin
<b>Description:</b>	The admin will be able to view the list of all users in the system.
<b>Trigger:</b>	The admin presses the View Users button from the side menu bar.
<b>Preconditions:</b>	N/A
<b>Post conditions:</b>	POST-1. Admin will be able to view the list of all users.
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. Admin logs in</li> <li>2. Admin selects the View Users menu from the navbar.</li> <li>3. List of all the users will be displayed.</li> </ol>

<b>Alternative Flows</b>	N/A
<b>Exceptions:</b>	N/A
<b>Business Rules:</b>	N/A
<b>Assumptions:</b>	1. The admin has already logged in. 2. Some users already exist.

### Delete user account

<b>Use Case ID:</b>	<b>UC-40</b>
<b>Use Case Name:</b>	Delete a user account
<b>Actors:</b>	Admin
<b>Description:</b>	The admin will be able to delete a user account.
<b>Trigger:</b>	The admin presses the View Users button from the side menu bar.
<b>Preconditions:</b>	N/A
<b>Post conditions:</b>	POST-1. Admin will be able to delete a specific user account.
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. Admin logs in</li> <li>2. Admin selects the View Users menu from the navbar.</li> <li>3. List of all the users will be displayed.</li> <li>4. The admin clicks on view button.</li> <li>5. The page containing the information of the user will be displayed.</li> <li>6. The admin clicks on the delete user account.</li> <li>7. A confirmation Dialogue Box will appear.</li> <li>8. admin will select Confirm button.</li> <li>9. A success message will be displayed.</li> </ol>
<b>Alternative Flows</b>	N/A
<b>Exceptions:</b>	N/A
<b>Business Rules:</b>	N/A
<b>Assumptions:</b>	1. The admin has already logged in. 2. Some users already exist.

### View all feedbacks list

<b>Use Case ID:</b>	<b>UC-41</b>
<b>Use Case Name:</b>	View All Feedback List
<b>Actors:</b>	Admin

<b>Description:</b>	The admin will be able to view the list of all user feedbacks.
<b>Trigger:</b>	The admin presses the View Feedback button Side menu.
<b>Preconditions:</b>	N/A
<b>Post conditions:</b>	POST-1. Admin will be able to view the list of all users feedbacks.
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. Admin logs in</li> <li>2. Admin selects the View Feedbacks menu from the side menu.</li> <li>3. List of all the feedbacks will be displayed.</li> </ol>
<b>Alternative Flows</b>	N/A
<b>Exceptions:</b>	N/A
<b>Business Rules:</b>	N/A
<b>Assumptions:</b>	<ol style="list-style-type: none"> <li>1. The admin has already logged in.</li> <li>2. Some feedbacks already exist.</li> </ol>

#### View User Feedback description

<b>Use Case ID:</b>	<b>UC-42</b>
<b>Use Case Name:</b>	View User Feedback Description
<b>Actors:</b>	Admin
<b>Description:</b>	The admin will be able to view the user feedback description.
<b>Trigger:</b>	The admin presses the View Details button
<b>Preconditions:</b>	N/A
<b>Post conditions:</b>	POST-1. Admin will be able to view the user feedback description
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. Admin logs in</li> <li>2. Admin selects the View Feedback button from the sidebar.</li> <li>3. List of all the feedbacks will be displayed.</li> <li>4. The admin will click on a specific user feedback details button.</li> <li>5. The details of the user feedback are displayed.</li> </ol>
<b>Alternative Flows</b>	N/A
<b>Exceptions:</b>	N/A
<b>Business Rules:</b>	N/A
<b>Assumptions:</b>	The admin has already logged in

## Sign-out

<b>Use Case ID:</b>	<b>UC-43</b>
<b>Use Case Name:</b>	Sign out
<b>Actors:</b>	Admin
<b>Description:</b>	Admin can log out of the system
<b>Trigger:</b>	Admin presses the Log out button
<b>Preconditions:</b>	PRE-1. Admin is logged in to the system
<b>Post conditions:</b>	POST-1. Admin will be logged out of the system
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. Admin logs in</li> <li>2. Admin clicks the logout button</li> <li>3. Admin is logged out</li> <li>4. Admin is redirected to the homepage</li> </ol>
<b>Alternative Flows</b>	N/A
<b>Exceptions:</b>	
<b>Business Rules:</b>	N/A
<b>Assumptions:</b>	<ol style="list-style-type: none"> <li>1. The admin has already logged in</li> <li>2. The admin has working internet connection</li> </ol>

## Manage cat food inventory in the store

<b>Use Case ID:</b>	<b>UC-44</b>
<b>Use Case Name:</b>	Manage Cat Food Inventory
<b>Actors:</b>	Primary Actor: Admin Secondary Actor: System
<b>Description:</b>	Admin can view and manage cat food inventory in the store to ensure availability and update stock levels.
<b>Trigger:</b>	Admin accesses the inventory management interface.
<b>Preconditions:</b>	PRE-1. Admin is logged into the system. PRE-2. Cat food inventory data is available.
<b>Postconditions:</b>	POST-1. Admin updates inventory levels, adds new products, or removes outdated items as needed.
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. Admin logs into the system and navigates to the inventory management section.</li> <li>2. Admin views the current inventory status and stock levels.</li> <li>3. Admin makes changes to inventory (e.g., add products, update quantities).</li> </ol>
<b>Alternative Flows:</b>	Ex-1. If an item is out of stock, Admin can place orders with suppliers or update expected arrival dates.
<b>Exceptions:</b>	None
<b>Business Rules:</b>	BR-1. Admin should regularly update inventory data to reflect current stock levels. BR-2. Admin actions should align with store policies and product availability.

### 3.2.8 Module 8: Statistical Analytics

#### FE-1: Generate Reports on Most Popular Cat Breeds Sold

Use Case ID	UC-45
Use Case Name	Generate Reports on Most Popular Cat Breeds Sold
Actors	Admin, System
Description	System generates reports on the most popular cat breeds sold in the store.
Trigger	Admin requests a report on popular cat breeds.
Preconditions	Sales data is available and up-to-date.
Postconditions	Report on popular cat breeds is generated and accessible to the admin.
Normal Flow	1. Admin navigates to the analytics section.2. Admin selects the option to generate a report on popular cat breeds.3. System processes the sales data.4. System generates the report highlighting the most popular cat breeds.5. Admin reviews the report.
Exceptions	Insufficient sales data to generate a report.
Business Rules	Reports should be accurate and reflect the latest sales data.
Assumptions	The system has comprehensive sales data for analysis.

#### FE-2: Track Performance of Individual Products

Use Case ID	UC-46
Use Case Name	Track Performance of Individual Products
Actors	Admin, System
Description	System tracks the performance of individual products in terms of sales, revenue, and customer ratings.
Trigger	Admin wants to analyze the performance of a specific product.
Preconditions	Product performance data is available.
Postconditions	Performance data for individual products is displayed.
Normal Flow	1. Admin navigates to the product performance section.2. Admin selects a product to analyze.3. System retrieves performance data for the selected product.4. System displays data on sales, revenue, and customer ratings.5. Admin reviews the performance metrics.
Exceptions	No performance data available for the selected product.
Business Rules	Performance metrics should be up-to-date and accurate.
Assumptions	The system has historical data for individual products.

#### FE-3: Implement Real-Time Data Dashboards

Use Case ID	UC-47
-------------	-------

Use Case Name	Implement Real-Time Data Dashboards
Actors	Admin, System
Description	System provides real-time data dashboards for quick insights into sales performance.
Trigger	Admin wants to view real-time sales data.
Preconditions	Real-time data collection is enabled.
Postconditions	Real-time data dashboards are displayed to the admin.
Normal Flow	1. Admin navigates to the real-time dashboard section.2. System retrieves and displays real-time sales data.3. Admin reviews the real-time data on the dashboard.4. System continuously updates the dashboard with new data.
Exceptions	Real-time data retrieval failure.
Business Rules	Dashboards should be updated continuously and reflect the latest data.
Assumptions	The system can collect and process real-time sales data efficiently.

#### FE-4: Monitor Inventory Levels

<b>Use Case ID</b>	<b>UC-48</b>
Use Case Name	Monitor Inventory Levels
Actors	Admin, System
Description	System monitors inventory levels, tracks stockouts, and generates alerts for low stock or overstock situations.
Trigger	Inventory levels reach critical thresholds.
Preconditions	Inventory tracking is enabled.
Postconditions	Alerts for low stock or overstock situations are generated and sent to the admin.
Normal Flow	1. System continuously monitors inventory levels.2. System identifies low stock or overstock situations.3. System generates alerts for these situations.4. Admin receives and reviews the alerts.5. Admin takes necessary actions to address the inventory issues.
Exceptions	Inventory tracking system failure.
Business Rules	Inventory monitoring should be accurate and proactive.
Assumptions	The system can accurately track and report inventory levels.

#### FE-5: Generate Monthly Sales Reports

<b>Use Case ID</b>	<b>UC-49</b>
Use Case Name	Generate Monthly Sales Reports
Actors	Admin, System
Description	System generates monthly sales reports to provide insights into the store's performance.
Trigger	End of the month.
Preconditions	Sales data for the month is complete.

Postconditions	Monthly sales report is generated and available for review.
Normal Flow	1. System identifies the end of the month.2. System compiles sales data for the month.3. System generates the monthly sales report.4. Admin reviews the sales report.
Exceptions	Incomplete sales data for the month.
Business Rules	Reports should be comprehensive and accurate.
Assumptions	The system can compile and analyze monthly sales data efficiently.

#### FE-6: Provide Yearly Sales Analysis

<b>Use Case ID</b>	<b>UC-50</b>
Use Case Name	Provide Yearly Sales Analysis
Actors	Admin, System
Description	System provides a detailed yearly sales analysis, including trends and growth rates.
Trigger	End of the year.
Preconditions	Sales data for the year is complete.
Postconditions	Yearly sales analysis report is generated and available for review.
Normal Flow	1. System identifies the end of the year.2. System compiles sales data for the year.3. System analyzes trends and growth rates.4. System generates the yearly sales analysis report.5. Admin reviews the yearly sales analysis.
Exceptions	Incomplete sales data for the year.
Business Rules	Analysis should include trends, growth rates, and other relevant metrics.
Assumptions	The system can compile and analyze yearly sales data effectively.

#### FE-7: Generate Custom Sales Reports

<b>Use Case ID</b>	<b>UC-51</b>
Use Case Name	Generate Custom Sales Reports
Actors	Admin, System
Description	Admin can generate custom sales reports based on specific criteria such as date range, product category, and region.
Trigger	Admin requests a custom sales report.
Preconditions	Sales data for the selected criteria is available.
Postconditions	Custom sales report is generated and available for review.
Normal Flow	1. Admin navigates to the custom reports section.2. Admin specifies criteria for the custom report.3. System processes the request based on the selected criteria.4. System generates the custom sales report.5. Admin reviews the custom sales report.
Exceptions	No data available for the selected criteria.
Business Rules	Custom reports should be flexible and tailored to admin needs.
Assumptions	The system can process and generate reports based on various criteria.

**FE-8: Analyze Customer Feedback Trends**

<b>Use Case ID</b>	<b>UC-52</b>
Use Case Name	Analyze Customer Feedback Trends
Actors	Admin, System
Description	System analyzes customer feedback trends to identify areas for improvement.
Trigger	Admin requests an analysis of customer feedback.
Preconditions	Customer feedback data is available.
Postconditions	Feedback trends report is generated and available for review.
Normal Flow	1. Admin navigates to the feedback analysis section.2. Admin requests an analysis of customer feedback.3. System compiles and analyzes feedback data.4. System generates a report highlighting feedback trends.5. Admin reviews the feedback trends report.
Exceptions	Insufficient feedback data for meaningful analysis.
Business Rules	Analysis should provide actionable insights for improvement.
Assumptions	The system has comprehensive feedback data for analysis.



### 3.3 Event-Response Table

Event	System State	Response
User uploads cat image for breed recognition	System is ready to process image for analysis	<ol style="list-style-type: none"><li>1. Preprocess the uploaded image to standardize pixel values and sizes.</li><li>2. Validate and normalize input images to enhance model accuracy.</li><li>3. Initiate breed recognition using ResNet-50.</li></ol>
Breed recognition process completes	Breed identification result is available	Display the detected cat breed along with the confidence score. Provide visual representation of key features contributing to breed identification.
User provides feedback on breed accuracy	Feedback submission interface is active	<ol style="list-style-type: none"><li>1. Collect user feedback on the accuracy of breed recognition.</li><li>2. Analyze feedback data for continuous model improvement.</li></ol>
User adds cat food items to shopping cart	Online store interface is accessible	Add selected cat food items to the shopping cart.
User removes cat food items from cart	Shopping cart contains selected items	Remove specified cat food items from the shopping cart.
User proceeds to checkout	Shopping cart contains selected items	<ol style="list-style-type: none"><li>1. Facilitate the checkout process.</li><li>2. Integrate a secure payment gateway for transaction processing.</li><li>3. Confirm order details and provide order confirmation to the user.</li></ol>
User interacts with chatbot for assistance	Chatbot interface is active	<ol style="list-style-type: none"><li>1. Allow users to submit queries, questions, or commands to the chatbot.</li><li>2. Implement quick responses based on user input.</li><li>3. Provide relevant information based on user's queries.</li></ol>

## 4. Functional Requirements

### 4.1 Module 1: User Authentication and Profile Management

**Table 1: Customer.Signup**

<b>Identifier</b>	FR-1
<b>Title</b>	Customer.Signup
<b>Requirement</b>	The system shall allow a customer to sign up.
<b>Source</b>	Team Member 1
<b>Rationale</b>	To allow a customer to make an account and signup for the website.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 2: Customer.Signup.Duplicate**

<b>Identifier</b>	FR-2
<b>Title</b>	Customer.Signup.Duplicate
<b>Requirement</b>	The system shall confirm that customer is not already a register user.
<b>Source</b>	Team Member 2
<b>Rationale</b>	To ensure system does not accept signup on email linked to an existing account.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 3: Customer.Signup.Form**

<b>Identifier</b>	FR-3
<b>Title</b>	Customer.Signup.Form
<b>Requirement</b>	The system shall require the necessary information for signup process.
<b>Source</b>	Team Member 1

<b>Rationale</b>	This requirement ensures that the system collects all necessary information for customer signup.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 4: Customer.Signup.Form.Empty**

<b>Identifier</b>	FR-4
<b>Title</b>	Customer.Signup.Form.Empty
<b>Requirement</b>	The system shall display an alert if any of the form fields are left empty.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This requirement ensures that customers provide all necessary information for registration.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 5: Customer.Signup.Cancel**

<b>Identifier</b>	FR-5
<b>Title</b>	Customer.Signup.Cancel
<b>Requirement</b>	The system shall terminate the signup process if the customer cancels.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This requirement ensures that the customer signup process is stopped if the customer cancels at any point.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 6: Customer.Signup.Form.Email.Standard**

<b>Identifier</b>	FR-6
<b>Title</b>	Customer.Signup.Form.Email.Standard
<b>Requirement</b>	The system shall display an alert if the entered email does not adhere to the standard format.
<b>Source</b>	Team Member 1

<b>Rationale</b>	This requirement ensures that customers provide a valid email address for registration.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 7: Customer.Signup.Form.Password.Standard**

<b>Identifier</b>	FR-7
<b>Title</b>	Customer.Signup.Form.Password.Standard
<b>Requirement</b>	The system shall display an alert if the entered password does not adhere to the standard format.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This requirement ensures that customers provide a strong password for their account.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 8: Customer.Signup.Email.Duplicate.Registered**

<b>Identifier</b>	FR-8
<b>Title</b>	Customer.Signup.Email.Duplicate.Registered
<b>Requirement</b>	The system shall display an alert if the entered email is already registered for another account.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Customer should be notified if the email is already a registered account.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-1.2
<b>Priority</b>	High

**Table 9: Customer.Signup.Email.Duplicate.Registered.Redirect.Login**

<b>Identifier</b>	FR-9
<b>Title</b>	Customer.Signup.Email.Duplicate.Registered.Redirect.Login
<b>Requirement</b>	The system shall redirect the customer to the login page and terminate the use case if the email is already registered.

<b>Source</b>	Team Member 1
<b>Rationale</b>	If the customer email already links to an account, the system shall redirect to login page.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-1.2
<b>Priority</b>	High

**Table 10: Customer.Signup.Create.Account**

<b>Identifier</b>	FR-10
<b>Title</b>	Customer.Signup.Create.Account
<b>Requirement</b>	The system shall create a customer account with the provided information.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This requirement ensures that the customer registration process is completed successfully.
<b>Business Rule</b>	The customer shall be at least 18 years old to register.
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 11: Customer.Signup.Create.Account.Failed**

<b>Identifier</b>	FR-11
<b>Title</b>	Customer.Signup.Create.Account.Failed
<b>Requirement</b>	The system shall notify customer for any errors in signup process regarding customer data.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This requirement allows the customer to correct any errors made in the registration process
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 12: Customer.Signup.Success**

<b>Identifier</b>	FR-12
<b>Title</b>	Customer.Signup.Success

<b>Requirement</b>	The system shall direct the customer to the homepage after successful registration.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Once signup is successful the customer account is created, customer is logged in and redirected to homepage
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-1.10
<b>Priority</b>	High

**Table 13: Signup.Google**

<b>Identifier</b>	FR-13
<b>Title</b>	Signup.Google
<b>Requirement</b>	The system shall have an option to sign up with google.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This requirement provides customers with an alternative path for registration using their Google account for a seamless account creation.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 14 Signup.Google.Permission**

<b>Identifier</b>	FR-14
<b>Title</b>	Signup.Google.Permission
<b>Requirement</b>	The system shall redirect the customer to Google's permission page after clicking the Signup with Google button.
<b>Source</b>	Team Member 1
<b>Rationale</b>	The system directs the customer to give permission to the system to connect with google account.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-1.13
<b>Priority</b>	High

**Table 15: Signup.Google.Permission.No**

<b>Identifier</b>	FR-15
-------------------	-------

<b>Title</b>	Signup.Google.Permission.No
<b>Requirement</b>	The system shall redirect the customer to the manual signup page if they deny access to their Google account.
<b>Source</b>	Team Member 1
<b>Rationale</b>	If the google account access is not provided, then the customer can signup manually
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-1.13
<b>Priority</b>	High

**Table 16:Signup.Google.Account.Link**

<b>Identifier</b>	FR-16
<b>Title</b>	Signup.Google.Account.Link
<b>Requirement</b>	The system shall create a customer account using the Google account information and direct the customer to the homepage.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Once the google account access is provided by customer system will make an account with the google information and allow customer system access.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 17: Customer.Login**

<b>Identifier</b>	FR-17
<b>Title</b>	Customer.Login
<b>Requirement</b>	The system shall allow customers to login using their email and password.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Customer with accounts can login using their email and password.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 18: Customer.Account.Active**

<b>Identifier</b>	FR-18
<b>Title</b>	Customer.Account.Active
<b>Requirement</b>	The system shall confirm that customer have an active account.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This requirement is necessary to ensure that only customers with active accounts can log in to the system and perform actions that are specific to their account.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 19: Customer.Account.Inactive**

<b>Identifier</b>	FR-19
<b>Title</b>	Customer.Account.Inactive
<b>Requirement</b>	The system shall prompt the customer that their account has been deactivated.
<b>Source</b>	Team Member 2
<b>Rationale</b>	If admin bans a customer, then they cannot be logged in to the system
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 20: Customer.Login.Form.Empty**

<b>Identifier</b>	FR-20
<b>Title</b>	Customer.Login.Form.Empty
<b>Requirement</b>	The system shall display an alert message when the customer leaves any form field empty on login page.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Email and password are mandatory to proceed with the login process.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High



**Table 21: Customer.Login.Cancel**

<b>Identifier</b>	FR-21
<b>Title</b>	Customer.Login.Cancel
<b>Requirement</b>	The system shall terminate the use case if the customer cancels.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This requirement is necessary to provide customers with the ability to cancel the login process if they change their mind or encounter issues, and to prevent any unintended actions or errors.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 22: Customer.Login.Email.Standard**

<b>Identifier</b>	FR-22
<b>Title</b>	Customer.Login.Email.Standard
<b>Requirement</b>	The system shall display an alert message when the customer enters an invalid email format while logging in.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This requirement is necessary to ensure that customers provide valid email addresses when logging in.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 23: Customer.Login.No.Account**

<b>Identifier</b>	FR-23
<b>Title</b>	Customer.Login.No.Account
<b>Requirement</b>	The system shall display an alert message when the customer's email does not link to any registered account.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This requirement is necessary to prevent unauthorized access to accounts by ensuring that only email addresses that linked to registered accounts are allowed to log in.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A

<b>Priority</b>	High
-----------------	------

**Table 24: Customer.Login.Success**

<b>Identifier</b>	FR-24
<b>Title</b>	Customer.Login.Success
<b>Requirement</b>	The system shall redirect the customer to the homepage upon successful login.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Customers are directed to homepage after successful login process.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-18,FR-22
<b>Priority</b>	High

**Table 25: Customer.Login.Google**

<b>Identifier</b>	FR-25
<b>Title</b>	Customer.Login.Google
<b>Requirement</b>	The system shall provide a "Login with Google" option for customers to log in using their Google account.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This requirement is necessary to provide customers with an alternative and convenient way to log in to their accounts using their Google account.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 26: Customer.Login.Google.Permission**

<b>Identifier</b>	FR-26
<b>Title</b>	Customer.Login.Google.Permission
<b>Requirement</b>	The system shall display a pop-up asking for permission to access the customer's Google account when the customer clicks the "Login with Google" button.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This requirement is necessary to ensure that customers are aware of and consent to the system's access to their Google account information, and to prevent unauthorized access to their account information.

<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 27: Customer.Login.Google.Permission.Denied**

<b>Identifier</b>	FR-27
<b>Title</b>	Customer.Login.Google.Permission.Denied
<b>Requirement</b>	The system shall display an alert message when the customer denies access to their Google account while logging in and direct to homepage.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This requirement is necessary to provide customers with a clear and visible notification if they deny access to their Google account information
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 28: Customer.Login.Google.Success**

<b>Identifier</b>	FR-28
<b>Title</b>	Customer.Login.Google.Success
<b>Requirement</b>	The system shall redirect the customer to the homepage upon successful login using Google.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Customer is logged in and redirected to homepage after successful link to google account.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-24
<b>Priority</b>	High

**Table 29: Customer.Authenticate**

<b>Identifier</b>	FR-29
<b>Title</b>	Customer.Authenticate
<b>Requirement</b>	The system shall validate the customer's entered credentials
<b>Source</b>	Team Member 1

<b>Rationale</b>	The system verifies that the entered credentials are valid and exist in the system.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 30: Customer.Credentials.Invalid**

<b>Identifier</b>	FR-30
<b>Title</b>	Customer.Credentials.Invalid
<b>Requirement</b>	The system shall prompt the customer if the given credentials are wrong.
<b>Source</b>	Team Member 1
<b>Rationale</b>	The system should tell the customer that the given credentials are wrong
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-29
<b>Priority</b>	High

**Table31: Customer.Ban.Check**

<b>Identifier</b>	FR-31
<b>Title</b>	Customer.Ban.Check
<b>Requirement</b>	The system shall check whether the customer is banned before granting access to the system.
<b>Source</b>	Team Member 1
<b>Rationale</b>	The system checks whether the customer is banned or not. Banned Customer should not be allowed to the system.
<b>Business Rule</b>	BR-1: Banned customers are not allowed to use the system.
<b>Dependencies</b>	FR-29
<b>Priority</b>	High

**Table 32: Customer.Authenticate.SystemError**

<b>Identifier</b>	FR-32
<b>Title</b>	Customer.Authenticate.SystemError
<b>Requirement</b>	The system shall prompt the customer if the system is down.
<b>Source</b>	Team Member 1

<b>Rationale</b>	The system should respond appropriately if the system is down or under maintenance.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 33: Sign.out**

<b>Identifier</b>	FR-33
<b>Title</b>	Sign.out
<b>Requirement</b>	The users will be able to sign out of their application on clicking sign out
<b>Source</b>	Team Member 1
<b>Rationale</b>	To allow users to sign out of the application so they cannot access the app anymore
<b>Business Rule (if required)</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 34: Select.Profile**

<b>Identifier</b>	FR-34
<b>Title</b>	Select.Profile
<b>Requirement</b>	The users will be able to view their profile and edit their personal information by selecting the personal information button in the settings
<b>Source</b>	Talha Ziaullah
<b>Rationale</b>	To empower users to access and update their personal information as needed
<b>Business Rule (if required)</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 35: Select.Personal.Information**

<b>Identifier</b>	FR-35
<b>Title</b>	Select.Personal.Information

<b>Requirement</b>	The users will be able to view the personal information by select the personal information button
<b>Source</b>	Team Member 1
<b>Rationale</b>	To view user's personal information so they can update it if they want
<b>Business Rule (if required)</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 36: Select.Update.Button**

<b>Identifier</b>	FR-36
<b>Title</b>	Select.Update.Button
<b>Requirement</b>	The users will be able to update the personal information by select the update button within their profile settings
<b>Source</b>	Team Member 1
<b>Rationale</b>	To allow the user to update the personal information if they want to
<b>Business Rule (if required)</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 37: Change.Username**

<b>Identifier</b>	FR-37
<b>Title</b>	Change.Username
<b>Requirement</b>	The users will be able to change their email to a new one
<b>Source</b>	Team Member 1
<b>Rationale</b>	To allow users to update their email address as needed.
<b>Business Rule (if required)</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 38: Change.Address**

<b>Identifier</b>	FR-38
<b>Title</b>	Change.Address
<b>Requirement</b>	The users will be able to change their address
<b>Source</b>	Team Member 1
<b>Rationale</b>	To change their address to a new one
<b>Business Rule (if required)</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 39: Change.Phone.Number**

<b>Identifier</b>	FR-39
<b>Title</b>	Change.Phone.Number
<b>Requirement</b>	The users will be able to change their phone number
<b>Source</b>	Team Member 1
<b>Rationale</b>	To change their phone number to a new one
<b>Business Rule (if required)</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 40: Save.Changes**

<b>Identifier</b>	FR-40
<b>Title</b>	Save.Changes
<b>Requirement</b>	Users shall be able to save their changes by clicking the “Save Changes” button in the profile settings. This action will update their profile information with the newly provided details.
<b>Source</b>	Team Member 1
<b>Rationale</b>	To allow users to persist the changes made to their profile information.
<b>Business Rule (if required)</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 41: Forget.Password**

<b>Identifier</b>	FR-41
<b>Title</b>	Forget.Password
<b>Requirement</b>	The users will be able to reset the password by clicking on Forgot password button
<b>Source</b>	Team Member 1
<b>Rationale</b>	To reset their password in case they have forgotten one
<b>Business Rule (if required)</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 42: Verification.Link**

<b>Identifier</b>	FR-42
<b>Title</b>	Verification.Link
<b>Requirement</b>	The users will be able to click the verification link that is sent to the provided email address
<b>Source</b>	Team Member 1
<b>Rationale</b>	To generate a link that will reset the password of that user
<b>Business Rule (if required)</b>	N/A
<b>Dependencies</b>	FR-41
<b>Priority</b>	High

**Table 43: Enter.New.Password**

<b>Identifier</b>	FR-43
<b>Title</b>	Enter new password
<b>Requirement</b>	The users will be able to enter a new password
<b>Source</b>	Team Member 1
<b>Rationale</b>	To allow authorized access to the application
<b>Business Rule (if required)</b>	N/A
<b>Dependencies</b>	FR-41
<b>Priority</b>	High



**Table 44: Enter.New.Confirm.Password**

<b>Identifier</b>	FR-44
<b>Title</b>	Enter.New.Confirmation.Password
<b>Requirement</b>	The users will be able to confirm the new password
<b>Source</b>	Team Member 1
<b>Rationale</b>	To ensure no mistyping of password occurs
<b>Business Rule (if required)</b>	N/A
<b>Dependencies</b>	FR-43
<b>Priority</b>	High

**Table 45: Delete.Account**

<b>Identifier</b>	FR-45
<b>Title</b>	Delete.Account
<b>Requirement</b>	The user shall be able to initiate the process of deleting their account by clicking the “Delete Account” button within their profile settings.
<b>Source</b>	Team Member 1
<b>Rationale</b>	To provide users with the option to permanently remove their account from the application if they no longer wish to use it.
<b>Business Rule (if required)</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Low

**Table 46: Confirm.Deletion**

<b>Identifier</b>	FR-46
<b>Title</b>	Confirm.Deletion
<b>Requirement</b>	User shall be required to confirm the account deletion process by clicking “Yes” on the confirmation dialogue box. The extra step is added to minimize the risk of accidental account deletion.
<b>Source</b>	Team Member 1

<b>Rationale</b>	To safeguard users against accidental account deletion by introducing a confirmation step.
<b>Business Rule (if required)</b>	N/A
<b>Dependencies</b>	FR-45
<b>Priority</b>	Medium

## 4.2 Module 2: Cat Breed Detection

**Table 47: Image.Upload.Functionality**

<b>Identifier</b>	FR-47
<b>Title</b>	Image.Upload.Functionality
<b>Requirement</b>	Users shall be able to upload images of their cats for breed recognition. The system shall process the uploaded image to identify the breed of the cat.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This requirement enables users to utilize the system for identifying the breed of their cats through image submission, enhancing user experience and engagement.
<b>Business Rule</b>	The uploaded images must be clear and of sufficient quality for accurate breed recognition
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 48: Error Handling for Unsupported Image Formats**

<b>Identifier</b>	FR-48
<b>Title</b>	Error Handling for Unsupported Image Formats.
<b>Requirement</b>	If a user attempts to upload an image in an unsupported format, the system shall display an error message indicating the acceptable formats (e.g., JPEG, PNG).
<b>Source</b>	Team Member 2
<b>Rationale</b>	This ensures users are informed about the acceptable image formats, reducing frustration and guiding them to provide the correct input.
<b>Business Rule</b>	Users must be notified of the supported image formats to ensure successful upload.

<b>Dependencies</b>	FR-47
<b>Priority</b>	High

**Table 49: Resizing and Normalizing Images**

<b>Identifier</b>	FR-49
<b>Title</b>	Resizing and Normalizing Images
<b>Requirement</b>	System shall, upon receiving input images, automatically resize them to a standardized dimension to ensure uniformity in image size across the dataset.
<b>Source</b>	Team Member 2
<b>Rationale</b>	To ensure consistent input data for processing and analysis of images.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-47
<b>Priority</b>	Medium

**Table 50: Image.Submission.Confirmation**

<b>Identifier</b>	FR-50
<b>Title</b>	Image.Submission.Confirmation
<b>Requirement</b>	Upon successful image upload, the system shall display a confirmation message to the user.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Providing confirmation reassures users that their submission was successful, enhancing their confidence in the system.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-47
<b>Priority</b>	Low

**Table 51: Camera.Access**

<b>Identifier</b>	FR-51
<b>Title</b>	Camera.Access
<b>Requirement</b>	The system must ask for camera access to the customer.
<b>Source</b>	Team Member 2

<b>Rationale</b>	This requirement ensures that the system requests permission from the customer to access their device's camera, which is necessary for accessing of image.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 52: Standardization.Pixel.Values.Sizes**

<b>Identifier</b>	FR-52
<b>Title</b>	Standardization.Pixel.Values.Sizes
<b>Requirement</b>	The system shall preprocess images submitted by users to standardize pixel values and sizes. This process involves ensuring that all images have consistent pixel values and sizes, regardless of their original format or resolution.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Standardizing pixel values and sizes ensures that images processed by the system are uniform, facilitating easier analysis, comparison, and manipulation.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 53: Handling.Image.Formats**

<b>Identifier</b>	FR-53
<b>Title</b>	Handling.Image.Formats
<b>Requirement</b>	The system shall support various image formats (e.g., JPEG, PNG, GIF) for submission within the Image Submission module.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Supporting multiple image formats enhances user flexibility and convenience, allowing them to submit images in their preferred format.
<b>Business Rule</b>	Users are allowed to submit images in any of the supported formats specified by the system.
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

--	--

**Table 54: Image.Quality.Assurance**

<b>Identifier</b>	FR-54
<b>Title</b>	Image.Quality.Assurance
<b>Requirement</b>	The system shall perform quality checks on submitted images to ensure they meet predefined quality standards.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Implementing quality assurance measures helps prevent the submission of low-quality or corrupted images, thus maintaining data accuracy and reliability.
<b>Business Rule</b>	Images failing to meet the quality standards will not be accepted for further processing within the system.
<b>Dependencies</b>	FR-52
<b>Priority</b>	Medium

**Table 55: Image Validation and Normalization**

<b>Identifier</b>	FR-55
<b>Title</b>	Image.Validation.Normalization
<b>Requirement</b>	The Image Submission module shall validate and normalize input images to enhance model accuracy.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Validating and normalizing input images is crucial for ensuring that the model receives consistent and reliable data, which ultimately enhances its accuracy in image recognition tasks.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 56: Format.Compatibility.Check**

<b>Identifier</b>	FR-56
<b>Title</b>	Format.Compatibility.Check

<b>Requirement</b>	The Image Submission module shall verify the format compatibility of input images. Supported formats include JPEG, PNG, and GIF.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Ensuring format compatibility helps prevent processing errors and maintains the integrity of the image data throughout the submission process.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 57: Splitting.Dataset.for.Model.Training**

<b>Identifier</b>	FR-57
<b>Title</b>	Splitting Dataset for Model Training
<b>Requirement</b>	The system shall split the dataset into training and testing sets for model training. The splitting process should ensure that a certain percentage of data is allocated for training and the remaining for testing.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Splitting the dataset into training and testing sets is crucial for evaluating the performance of machine learning models accurately. It allows the model to learn patterns from the training data and assess its generalization ability on unseen data.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 58: Integration of ResNet-50 Architecture**

<b>Identifier</b>	FR-58
<b>Title</b>	Integration of ResNet-50 Architecture
<b>Requirement</b>	The system shall integrate the ResNet-50 architecture for accurate cat breed recognition.
<b>Source</b>	Team Member 1
<b>Rationale</b>	The integration of ResNet-50 architecture is essential to leverage its state-of-the-art deep learning capabilities, ensuring accurate identification of cat breeds within the FelineConnect system.

<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 59: Cat.Image.Processing**

<b>Identifier</b>	FR-59
<b>Title</b>	Cat.Image.Processing
<b>Requirement</b>	The system shall preprocess uploaded cat images before feeding them into the ResNet-50 architecture for breed recognition.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Preprocessing cat images ensures optimal input for the ResNet-50 architecture, enhancing the accuracy and speed of breed recognition.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-58
<b>Priority</b>	High

**Table 60: Feedback.Loop.Implementation**

<b>Identifier</b>	FR-60
<b>Title</b>	Feedback.Loop.Implementation
<b>Requirement</b>	The system shall implement a feedback loop mechanism for users to provide insights into the accuracy of breed recognition results.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Feedback from users facilitates continuous improvement of the breed recognition model, ensuring enhanced accuracy and user satisfaction over time.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 61: Integration of ResNet-50 Model for Transfer Learning**

<b>Identifier</b>	FR-61
<b>Title</b>	Integration of ResNet-50 Model for Transfer Learning
<b>Requirement</b>	The system shall integrate a pre-trained ResNet-50 model for transfer learning purposes.

<b>Source</b>	Team Member 1
<b>Rationale</b>	This requirement is essential to leverage the existing knowledge encoded in the ResNet-50 model for fine-tuning on specific cat breed recognition tasks, which will expedite the development process and improve model performance.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 62: Data Preparation for Transfer Learning**

<b>Identifier</b>	FR-62
<b>Title</b>	Data Preparation for Transfer Learning
<b>Requirement</b>	Prior to transfer learning, the system shall preprocess the input cat image data to ensure compatibility with the ResNet-50 model architecture.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Proper data preprocessing is crucial for effective transfer learning, as it ensures that the input data aligns with the requirements of the ResNet-50 model, thereby enhancing the efficiency and accuracy of the training process.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-61
<b>Priority</b>	High

**Table 63: Fine-Tuning ResNet-50 Model**

<b>Identifier</b>	FR-63
<b>Title</b>	Fine-Tuning ResNet-50 Model
<b>Requirement</b>	The system shall fine-tune the integrated ResNet-50 model using the preprocessed cat image dataset to adapt its weights and parameters for accurate breed recognition.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Fine-tuning the ResNet-50 model on cat image data is necessary to customize its features for the specific task of breed recognition, thereby improving the model's accuracy and performance in identifying different cat breeds.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-61



<b>Priority</b>	High
-----------------	------

**Table 64: Model Evaluation and Validation**

<b>Identifier</b>	FR-64
<b>Title</b>	Model Evaluation and Validation
<b>Requirement</b>	The system shall evaluate and validate the performance of the fine-tuned ResNet-50 model using appropriate metrics
<b>Source</b>	Team Member 2
<b>Rationale</b>	Model evaluation and validation are crucial steps to ensure the effectiveness and reliability of the integrated ResNet-50 model for breed recognition tasks, enabling informed decisions regarding its deployment in the FelineConnect system.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-63
<b>Priority</b>	High

**Table 65: Fine-tuning ResNet-50 Model on Domestic Cat Breeds Dataset**

<b>Identifier</b>	FR-65
<b>Title</b>	Fine-tuning ResNet-50 Model on Domestic Cat Breeds Dataset
<b>Requirement</b>	The system shall fine-tune the ResNet-50 deep learning model on a dataset comprising various domestic cat breeds.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Fine-tuning the ResNet-50 model specifically on a dataset of domestic cat breeds is essential to ensure accurate breed identification for the target user base of FelineConnect.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 66: Model Validation for Breed Recognition**

<b>Identifier</b>	FR-66
<b>Title</b>	Model Validation for Breed Recognition
<b>Requirement</b>	The system shall fine-tune the ResNet-50 deep learning model on a dataset comprising various domestic cat breeds.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Validating the model's accuracy with an independent test dataset ensures robustness and reliability in breed recognition, thereby enhancing user trust in the system's capabilities.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 67: User Feedback Integration**

<b>Identifier</b>	FR-67
<b>Title</b>	User Feedback Integration
<b>Requirement</b>	The system shall integrate a user feedback mechanism allowing users to provide insights into breed recognition accuracy.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Integrating a user feedback loop facilitates continuous improvement of the model's accuracy, ensuring enhanced performance over time and increased user satisfaction with breed identification results.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 67: Display detected cat breed along with confidence score.**

<b>Identifier</b>	FR-67
<b>Title</b>	Display detected cat breed along with confidence score.
<b>Requirement</b>	The system shall display the detected cat breed along with the confidence score after analyzing the uploaded cat image.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Users need to know the detected breed and the confidence level to trust the accuracy of the system's identification.

<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 68: Provide options for user feedback on breed detection accuracy.**

<b>Identifier</b>	FR-68
<b>Title</b>	Provide options for user feedback on breed detection accuracy.
<b>Requirement</b>	The system shall offer users the option to provide feedback on the accuracy of the detected cat breed, allowing them to indicate if the identification was correct or incorrect.
<b>Source</b>	Team Member 2
<b>Rationale</b>	User feedback is essential for continuous improvement of the breed detection model, enhancing its accuracy and reliability over time.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 69: Implement algorithm for determining confidence score.**

<b>Identifier</b>	FR-69
<b>Title</b>	Implement algorithm for determining confidence score.
<b>Requirement</b>	The system shall employ an algorithm to calculate the confidence score for each detected cat breed, based on the analysis of features extracted from the uploaded cat image.
<b>Source</b>	Team Member 2
<b>Rationale</b>	The confidence score provides users with insight into the reliability of the detected breed identification, enhancing trust in the system's capabilities.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 70: Allow users to view breed detection history.**

<b>Identifier</b>	FR-70
<b>Title</b>	Allow users to view breed detection history.

<b>Requirement</b>	The system shall enable users to access their breed detection history, displaying previously uploaded images along with the corresponding detected breeds and confidence scores.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Providing access to detection history allows users to track past identification results and monitor changes in accuracy over time.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 71: Visual Representation of Breed Identification Features**

<b>Identifier</b>	FR-71
<b>Title</b>	Visual Representation of Breed Identification Features
<b>Requirement</b>	The system shall display a visual representation of key features contributing to the identification of a cat breed based on uploaded images.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Providing users with a visual representation of key features contributing to breed identification enhances user understanding and confidence in the accuracy of the system's results.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 72: Instant Breed Identification Feedback**

<b>Identifier</b>	FR-72
<b>Title</b>	Instant Breed Identification Feedback
<b>Requirement</b>	Upon uploading an image, the system shall promptly generate and display the breed identification results, along with the visual representation of key features contributing to the identification.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Instant feedback enhances user experience by providing prompt results, contributing to user satisfaction and engagement with the platform.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A

<b>Priority</b>	High
-----------------	------

**Table 73: Customizable Visual Representation Settings**

<b>Identifier</b>	FR-73
<b>Title</b>	Customizable Visual Representation Settings
<b>Requirement</b>	The system shall allow users to customize the visual representation settings for breed identification features. Users shall have the option to adjust the transparency, color, and size of highlighted areas or overlays according to their preferences.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Providing customization options enhances user engagement and satisfaction by allowing users to tailor the visual representation to their preferences and needs.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 74: Breed Identification Result Sharing**

<b>Identifier</b>	FR-74
<b>Title</b>	Breed Identification Result Sharing
<b>Requirement</b>	The system shall provide functionality for users to share breed identification results via various social media platforms or email.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Enabling result sharing enhances user engagement and promotes platform visibility, allowing users to easily share their cat's breed identification with friends, family, or social media followers.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 75: User Feedback on Breed Recognition Accuracy**

<b>Identifier</b>	FR-75
<b>Title</b>	User Feedback on Breed Recognition Accuracy

<b>Requirement</b>	Users shall be able to provide feedback on the accuracy of breed recognition results generated by the system.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Continuous improvement of breed recognition accuracy is crucial for enhancing user satisfaction and system reliability.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 76: Feedback Analysis and Reporting**

<b>Identifier</b>	FR-76
<b>Title</b>	Feedback Analysis and Reporting
<b>Requirement</b>	The system shall analyze aggregated user feedback data periodically to identify trends, patterns, and areas for improvement in breed recognition accuracy.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Analyzing user feedback enables systematic evaluation of the system's performance and guides decision-making for future enhancements and optimizations.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-75
<b>Priority</b>	High

## 4.3 Module 3: Online Store

**Table 77: Product.Stock**

<b>Identifier</b>	FR-77
<b>Title</b>	Product.Stock
<b>Requirement</b>	The system shall check that the product is in stock.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Ensuring product availability to add in cart.
<b>Business Rule</b>	N/A

<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 78: Product.Stock.No**

<b>Identifier</b>	FR-78
<b>Title</b>	Product.Stock.No
<b>Requirement</b>	The system shall alter a message that the product is out of stock.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Providing message when product is out of stock.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-77
<b>Priority</b>	High

**Table 79: Product.Customer.LoggedIn**

<b>Identifier</b>	FR-79
<b>Title</b>	Product.Customer.LoggedIn
<b>Requirement</b>	The system shall check that that the customer is logged in.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Checking customer login status before adding a product to cart
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 80: Product.Customer.LoggedIn.No**

<b>Identifier</b>	FR-80
<b>Title</b>	Product.Customer.LoggedIn.No
<b>Requirement</b>	The system shall direct the customer to login page if the customer is not logged in.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Directing customer to login page if not logged in
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-79

<b>Priority</b>	High
-----------------	------

**Table 81: Product.Viewdetail**

<b>Identifier</b>	FR-81
<b>Title</b>	Product.Viewdetail
<b>Requirement</b>	The system shall display the product details on the product page
<b>Source</b>	Team Member 2
<b>Rationale</b>	Displaying product details to the customer.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 82: Product.Quantity**

<b>Identifier</b>	FR-82
<b>Title</b>	Product.Quantity
<b>Requirement</b>	The system shall allow the customer to set the quantity of the product they want to add to the cart.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Allowing customer to set quantity of the product they want to add
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 83: AddToCart.Button**

<b>Identifier</b>	FR-83
<b>Title</b>	AddToCart.Button
<b>Requirement</b>	The system shall enable the customer to press the "Add to Cart" button.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Enabling customer to add product to cart
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High



**Table 84: AddToCart**

<b>Identifier</b>	FR-84
<b>Title</b>	AddToCart
<b>Requirement</b>	The system shall add the selected product to the cart.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Adding selected product to the cart
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-82
<b>Priority</b>	High

**Table 85: AddToCart.Yes**

<b>Identifier</b>	FR-85
<b>Title</b>	AddToCart.Yes
<b>Requirement</b>	The system shall allow the customer to surf other product pages after adding a product to the cart.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Allowing customer to surf other product pages after adding a product to cart
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 86: AddToCart.Identical**

<b>Identifier</b>	FR-86
<b>Title</b>	AddToCart.Identical
<b>Requirement</b>	The system shall allow the customer to add multiple identical products to the cart.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Allowing customer to add multiple identical products to cart
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 87: Product.Quantity.Insufficaint**

<b>Identifier</b>	FR-87
<b>Title</b>	Product.Quantity.Insufficaint
<b>Requirement</b>	The system shall generate an alert if there is insufficient inventory to fulfill the required quantity of a product.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Alerting customer if required product quantity is not available
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-83
<b>Priority</b>	High

**Table 88: AddToCart.Limit**

<b>Identifier</b>	FR-88
<b>Title</b>	AddToCart.Limit
<b>Requirement</b>	The system shall limit a single product to be added to the cart with a maximum quantity of 10.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Limiting a single product to be added to cart with a maximum quantity of 10
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 89: Cart.Product**

<b>Identifier</b>	FR-89
<b>Title</b>	Cart.Product
<b>Requirement</b>	The system shall verify that user have at least one product in cart.
<b>Source</b>	Team Member 1
<b>Rationale</b>	The system ensures that the user has at least one product in their cart before deleting.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 90: Cart.Product.Empty**

<b>Identifier</b>	FR-90
<b>Title</b>	Cart.Product.Empty
<b>Requirement</b>	The system shall direct the user to home page for add to cart.
<b>Source</b>	Team Member 1
<b>Rationale</b>	The system redirects the user to the home page if they try to access the cart without any product added to it
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 91: Product.Cart.Delete**

<b>Identifier</b>	FR-91
<b>Title</b>	Product.Cart.Delete
<b>Requirement</b>	The system shall allow the customer to delete a single or more quantity of a product from the cart.
<b>Source</b>	Team Member 1
<b>Rationale</b>	The system allows the customer to delete one or more quantity of a product from the cart
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 92: Product.Cart.Delete.Recalculate**

<b>Identifier</b>	FR-92
<b>Title</b>	Product.Cart.Delete.Recalculate
<b>Requirement</b>	The system shall recalculate the cart amount after deleting a product from the cart.
<b>Source</b>	Team Member 1
<b>Rationale</b>	The system recalculates the cart amount after a product is deleted from the cart
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 93: Delete.Single.Quantity**

<b>Identifier</b>	FR-93
<b>Title</b>	Delete.Single.Quantity
<b>Requirement</b>	The system shall allow the customer to delete more than one quantity of a single product from the cart.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Customer may want to delete more than one quantity of a single product.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 94: Clear.Empty.Cart**

<b>Identifier</b>	FR-94
<b>Title</b>	Clear.Empty.Cart
<b>Requirement</b>	The system shall allow the customer to clear the complete cart.
<b>Source</b>	Team Member 2
<b>Rationale</b>	The system allows the customer to clear the complete cart.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 95: User Feedback on Purchased Cat Food Items**

<b>Identifier</b>	FR-95
<b>Title</b>	User Feedback on Purchased Cat Food Items
<b>Requirement</b>	After purchasing cat food items from the online store, users shall be prompted to provide feedback and ratings for the purchased items. This feedback will be collected and displayed on the product pages to assist other users in making informed purchasing decisions.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Incorporating a review and rating system for purchased cat food items enhances user engagement and satisfaction by facilitating peer-to-peer recommendations and fostering a sense of community within the platform.
<b>Business Rule</b>	N/A

<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 96: Display Product Reviews and Ratings**

<b>Identifier</b>	FR-96
<b>Title</b>	Display Product Reviews and Ratings
<b>Requirement</b>	The online store interface shall display user-generated reviews and ratings for each cat food item on the respective product pages.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Providing access to product reviews and ratings enhances the transparency and trustworthiness of the online store, empowering users to make informed purchasing decisions based on the experiences of others.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

## 4.4 Module 4: Search Filters and Notifications

**Table 97: Customer.Implement.search.bar**

<b>Identifier</b>	FR-97
<b>Title</b>	Customer.Implement.search.bar
<b>Requirement</b>	Implement a search bar for users to find cat breeds within the application.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This FR is required to allow customers to pay for their purchases and is necessary for the business to accept payments.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 98: Refine.Search.Result**

<b>Identifier</b>	FR-98
<b>Title</b>	Refine.Search.Result

<b>Requirement</b>	Allow users to refine search results using filters like breed type, food category, price range.
<b>Source</b>	Team Member 1
<b>Rationale</b>	By enabling users to refine search results with filters, we enhance the search experience, allowing users to narrow down their options and find the most relevant information more efficiently.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 99: Search.Specific.Cat**

<b>Identifier</b>	FR-99
<b>Title</b>	Search.Specific.Cat
<b>Requirement</b>	Allow users to search for specific cat food items.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Expanding the search functionality to include specific cat food items enhances the utility of the application, catering to the diverse needs of users interested in purchasing cat food products.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

## 4.5 Module 5: Payment Processing

**Table 100: Customer.Payment.Method**

<b>Identifier</b>	FR-100
<b>Title</b>	Customer.Payment.Method
<b>Requirement</b>	The system shall allow the customer to add a payment method during checkout.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This FR is required to allow customers to pay for their purchases and is necessary for the business to accept payments.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A

<b>Priority</b>	High
-----------------	------

**Table 101: Customer.Card.Detail**

<b>Identifier</b>	FR-101
<b>Title</b>	Customer.Card.Detail
<b>Requirement</b>	The system shall request the customer to provide the Card Number, Expiry Date, and CVC while adding a payment method.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This FR is necessary to ensure that the correct payment details are collected and processed during the checkout process, ensuring accurate and secure payments.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 102: Payment.CashOnDelivery**

<b>Identifier</b>	FR-102
<b>Title</b>	Paymnet.CashOnDelivery
<b>Requirement</b>	The system shall allow the customer to select Cash on Delivery as a payment method.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This FR is required to provide customers with a convenient and familiar payment method.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 103: Payment.Method.Future**

<b>Identifier</b>	FR-103
<b>Title</b>	Payment.Method.Future
<b>Requirement</b>	The system shall allow the customer to save a payment method for future purchases.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This FR is necessary to save time for the customer during future checkouts and provides a seamless checkout experience.

<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 104: Payment.Method.Once**

<b>Identifier</b>	FR-104
<b>Title</b>	Payment.Method.Once
<b>Requirement</b>	The system shall allow the customer to use a payment method once for the current purchase.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This FR provides flexibility to the customer to choose how they want to pay for their purchase.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 105: CashOnDelivery.Limit**

<b>Identifier</b>	FR-105
<b>Title</b>	CashOnDelivery.Limit
<b>Requirement</b>	The system shall not allow Cash on Delivery for orders above Rs. 15,000.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This FR is necessary to mitigate risks associated with high-value purchases and to ensure that payments are received securely.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 106: Customer.Card.Format**

<b>Identifier</b>	FR-106
<b>Title</b>	Customer.Card.Format
<b>Requirement</b>	The system shall display an alert if the Card Number entered by the customer is in the wrong format.
<b>Source</b>	Team Member 1



<b>Rationale</b>	This FR is necessary to ensure that customers enter valid payment details, and to avoid errors that could result in failed payments.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 107: Customer.Card.Expire**

<b>Identifier</b>	FR-107
<b>Title</b>	Customer.Card.Expire
<b>Requirement</b>	This FR is necessary to ensure that customers enter valid payment details, and to avoid errors that could result in failed payments.
<b>Source</b>	Team Member 1
<b>Rationale</b>	The system allows the customer to clear the complete cart.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 108: Edit.Payment.Method**

<b>Identifier</b>	FR-108
<b>Title</b>	Edit.Payment.Method
<b>Requirement</b>	The system shall allow the customer to edit their payment method.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This requirement enables the customer to modify their payment method.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 109: Payment.Method.Valid**

<b>Identifier</b>	FR-109
<b>Title</b>	Payment.Method.Valid
<b>Requirement</b>	The system shall check the customer has a valid payment method linked to the account.
<b>Source</b>	Team Member 1

<b>Rationale</b>	This requirement ensures that the customer has an existing payment method linked to their account before allowing them to edit it.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-105
<b>Priority</b>	High

**Table 110: Payment.Method.Valid.No**

<b>Identifier</b>	FR-110
<b>Title</b>	Payment.Method.Valid.No
<b>Requirement</b>	The system shall alter a message that user don't have a valid payment method linked.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This requirement provides feedback to the user if they attempt to edit a payment method that is not linked to their account.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-106
<b>Priority</b>	High

**Table 111: Payment.Method.Edit.New**

<b>Identifier</b>	FR-111
<b>Title</b>	Payment.Method.Edit.New
<b>Requirement</b>	The system shall require the customer to provide a new Card Number, Expiry Date, and CVC for the updated payment method.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This requirement ensures that the customer enters valid and updated payment details.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 112: Payment.Method.Edit.New.Save**

<b>Identifier</b>	FR-112
<b>Title</b>	Payment.Method.Edit.New.Save

<b>Requirement</b>	The system must save the new payment method and link it to the customer's account.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This requirement stores the updated payment details and links it to the customer's account for future use.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 113: Payment.Method.Edit.Card.Format**

<b>Identifier</b>	FR-113
<b>Title</b>	Payment.Method.Edit.Card.Format
<b>Requirement</b>	The system shall verify that the Card Number entered by the customer is in the correct format.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This requirement checks that the card number entered by the customer is in the correct format before saving it.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 114: Payment.Method.Edit.Card.Format.Wrong**

<b>Identifier</b>	FR-114
<b>Title</b>	Payment.Method.Edit.Card.Format.Wrong
<b>Requirement</b>	The system shall display an alert message if the Card Number entered by the customer is not in the correct format.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This requirement provides feedback to the user if they enter an incorrect card number format.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-81
<b>Priority</b>	High

**Table115: Payment.Method.Edit.Card.ExpiryDate**

<b>Identifier</b>	FR-115
<b>Title</b>	Payment.Method.Edit.Card.ExpiryDate
<b>Requirement</b>	The system shall verify that the Card Expiry Date entered by the customer is not in the past.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This requirement checks that the card expiry date entered by the customer is not in the past.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 116: Payment.Method.Edit.Card.ExpiryDate.Passed**

<b>Identifier</b>	FR-116
<b>Title</b>	Payment.Method.Edit.Card.ExpiryDate.Passed
<b>Requirement</b>	The system shall display an alert message if the Card Expiry Date entered by the customer is in the past.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This requirement provides feedback to the user if they enter an expired card expiry date.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 117: Checkout.Cart**

<b>Identifier</b>	FR-117
<b>Title</b>	Checkout.Cart
<b>Requirement</b>	The system shall not allow the checkout process if the customer cart is empty.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This ensures that orders cannot be placed without any items.
<b>Business Rule</b>	N/A

<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 118: Checkout.Cart.Product.List**

<b>Identifier</b>	FR-118
<b>Title</b>	Checkout.Cart.Product.List
<b>Requirement</b>	The system shall display the list of products in the customer's cart for confirmation during checkout
<b>Source</b>	Team Member 2
<b>Rationale</b>	This allows customers to verify their order before making a purchase.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 119: Checkout.Shipping.Option**

<b>Identifier</b>	FR-119
<b>Title</b>	Checkout.Shipping.Option
<b>Requirement</b>	The system shall allow the customer to select shipping options during checkout.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This allows customers to choose their preferred delivery method.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 120: Checkout.Confirm.Order**

<b>Identifier</b>	FR-120
<b>Title</b>	Checkout.Confirm.Order
<b>Requirement</b>	The system shall allow the customer to confirm the order during checkout.
<b>Source</b>	Team Member 1

<b>Rationale</b>	This allows customers to place their order and finalize their purchase.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 121: Checkout.Select.Payment.Method**

<b>Identifier</b>	FR-121
<b>Title</b>	Checkout.Select.Payment.Method
<b>Requirement</b>	The system shall allow the customer to select a payment method during checkout
<b>Source</b>	Team Member 2
<b>Rationale</b>	This allows customers to choose their preferred payment method.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 122: Checkout.Add.Payment.Method**

<b>Identifier</b>	FR-122
<b>Title</b>	Checkout.Add.Payment.Method
<b>Requirement</b>	The system shall allow the customer to add a payment method if N/A is available.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This allows customers to add or modify their payment information.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 123: Checkout.Proccess.Payment.Method**

<b>Identifier</b>	FR-123
<b>Title</b>	Checkout.Proccess.Payment.Method
<b>Requirement</b>	The system shall process the payment from the customer's payment method during checkout.
<b>Source</b>	Team Member 1

<b>Rationale</b>	This ensures that payment is collected securely and accurately.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 124: Checkout.Payment.Method.Not.Successful**

<b>Identifier</b>	FR-124
<b>Title</b>	Checkout.Payment.Method.Not.Successful
<b>Requirement</b>	The system shall display an alert if payment is not successful
<b>Source</b>	Team Member 2
<b>Rationale</b>	This informs the customer of a payment failure and allows them to take appropriate action.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 125: Checkout.Payment.Method.Retry**

<b>Identifier</b>	FR-125
<b>Title</b>	Checkout.Payment.Method.Retry
<b>Requirement</b>	The system shall allow the customer to retry payment if it fails.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This allows customers to try again if there was a payment failure.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-121
<b>Priority</b>	High

**Table 126: Checkout.Confirmation.Email**

<b>Identifier</b>	FR-126
<b>Title</b>	Checkout.Confirmation.Email
<b>Requirement</b>	The system shall send an email with the order summary to the customer after the checkout process is complete.
<b>Source</b>	Team Member 2

<b>Rationale</b>	This provides customers with a record of their purchase and allows them to track their order
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 127: Checkout.Notify.Admin**

<b>Identifier</b>	FR-127
<b>Title</b>	Checkout.Notify.Admin
<b>Requirement</b>	The system shall notify the admin about the new order.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This informs the admin that a new order has been placed and requires attention.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 128: Checkout.Stock**

<b>Identifier</b>	FR-128
<b>Title</b>	Checkout.Stock
<b>Requirement</b>	The system shall update the stock after the order is converted from the cart.
<b>Source</b>	Team Member 1
<b>Rationale</b>	This ensures that inventory levels are accurate and prevents overselling.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

## 4.6 Module 6: AI Chatbot

**Table 129: User Query Submission**

<b>Identifier</b>	FR-129
<b>Title</b>	User Query Submission





<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 132: Chatbot.Available**

<b>Identifier</b>	FR-132
<b>Title</b>	Chatbot.Available
<b>Requirement</b>	The system shall check that the Chatbot is available.
<b>Source</b>	Team Member 2
<b>Rationale</b>	To ensure that the Chatbot is available for use.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 133: Chatbot.Available.No**

<b>Identifier</b>	FR-133
<b>Title</b>	Chatbot.Available.No
<b>Requirement</b>	The system shall show an alert message if the Chatbot is not available.
<b>Source</b>	Team Member 2
<b>Rationale</b>	To ensure that customers is informed when the Chatbot is not available.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 134: Chatbot.Search.Field**

<b>Identifier</b>	FR-134
<b>Title</b>	Chatbot.Search.Field
<b>Requirement</b>	The system shall display a textbox for customers to enter their queries.
<b>Source</b>	User
<b>Rationale</b>	To enable customers to enter their queries, the system should display a textbox.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A

<b>Priority</b>	High
-----------------	------

**Table 135: Chatbot.Search.Response**

<b>Identifier</b>	FR-135
<b>Title</b>	Chatbot.Search.Response
<b>Requirement</b>	The system shall process the customer's query and generate a response.
<b>Source</b>	Team Member 2
<b>Rationale</b>	To provide responses to customer queries, the system should process the query and generate a response.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 136: Chatbot.UnableToUnderstand**

<b>Identifier</b>	FR-136
<b>Title</b>	Chatbot.UnableToUnderstand
<b>Requirement</b>	The system shall handle situations where the chatbot is unable to understand the customer's query by showing a message asking the customer to rephrase the question.
<b>Source</b>	Team Member 2
<b>Rationale</b>	To handle situations where the Chatbot is unable to understand a query, the system should ask the customer to rephrase the question.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 137: Chatbot.Cancel**

<b>Identifier</b>	FR-137
<b>Title</b>	Chatbot.Cancel
<b>Requirement</b>	The system shall terminate the process if the customer cancels interacting with the chatbot.
<b>Source</b>	Team Member 1
<b>Rationale</b>	To terminate the process when an customer cancels interacting with the Chatbot.

<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 138: Chatbot.Technical.Issue**

<b>Identifier</b>	FR-138
<b>Title</b>	Chatbot.Technical.Issue
<b>Requirement</b>	The system shall show an alert if any technical issues are encountered with chatbot.
<b>Source</b>	Team Member 2
<b>Rationale</b>	To notify the customer about the status of chatbot.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 139: Customer.Write.Message**

<b>Identifier</b>	FR-139
<b>Title</b>	Customer.Write.Message
<b>Requirement</b>	The system shall provide a text box for the customer to write a message to the admin.
<b>Source</b>	Team Member 2
<b>Rationale</b>	To allow customers to communicate with the admin and seek product-related queries.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 140: System.Alert.TextLimit**

<b>Identifier</b>	FR-140
<b>Title</b>	System.Alert.TextLimit
<b>Requirement</b>	The system shall alert the customer if the message text exceeds 255 characters.
<b>Source</b>	Team Member 2

<b>Rationale</b>	To ensure customers do not exceed the message character limit.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

## 4.7 Module 7: Admin Panel

**Table 141: Admin.Notification**

<b>Identifier</b>	FR-141
<b>Title</b>	Admin.Notification
<b>Requirement</b>	The system shall notify the admin of the customer message.
<b>Source</b>	Team Member 2
<b>Rationale</b>	To ensure the admin is aware of customer queries and can respond accordingly.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-136
<b>Priority</b>	High

**Table 142: Super.Admin.Login**

<b>Identifier</b>	FR-142
<b>Title</b>	Super.Admin.Login
<b>Requirement</b>	The system shall check that super admin is logged in to the system before adding a new admin.
<b>Source</b>	Team Member 2
<b>Rationale</b>	To ensure only authorized users can add admins
<b>Business Rule</b>	None
<b>Dependencies</b>	None
<b>Priority</b>	High

**Table 143: Super.Admin.Permission**

<b>Identifier</b>	FR-143
<b>Title</b>	Super.Admin.Permission
<b>Requirement</b>	The system shall check that the super admin has the permission to add a new admin.

<b>Source</b>	Team Member 2
<b>Rationale</b>	To maintain proper access control
<b>Business Rule</b>	None
<b>Dependencies</b>	None
<b>Priority</b>	High

**Table 144: New.Admin**

<b>Identifier</b>	FR-141
<b>Title</b>	New.Admin
<b>Requirement</b>	The system shall add the new Admin to the system when the Super Admin submits the information.
<b>Source</b>	Team Member 1
<b>Rationale</b>	To create the new admin account
<b>Business Rule (if required)</b>	None
<b>Dependencies</b>	None
<b>Priority</b>	High

**Table 145: Customer.Authenticate**

<b>Identifier</b>	FR-145
<b>Title</b>	Customer.Authenticate
<b>Requirement</b>	The system shall validate the Customer's entered credentials
<b>Source</b>	Team Member 1
<b>Rationale</b>	The system verifies that the entered credentials are valid and exist in the system.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 146: Admin.Authenticate.SystemError**

<b>Identifier</b>	FR-146
<b>Title</b>	Admin.Authenticate.SystemError
<b>Requirement</b>	The system shall prompt the Admin if the system is down.
<b>Source</b>	Team Member 1
<b>Rationale</b>	The system should respond appropriately if the system is down or under maintenance.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 147: Admin.Faq.View**

<b>Identifier</b>	FR-147
<b>Title</b>	Admin.Faq.View
<b>Requirement</b>	System shall allow the admin to view all faqs.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Faqs are a list of questions and answers that are visible to customers. Admin can view a list of all active faq that are shown on the customer side.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 148: Admin.Faq.View.Empty**

<b>Identifier</b>	FR-148
<b>Title</b>	Admin.Faq.View.Empty
<b>Requirement</b>	System shall notify the admin if there are no faqs to show.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Notifying the admin if the list of faqs are empty.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 149: Admin.Faq.Add.Privilege**

<b>Identifier</b>	FR-149
<b>Title</b>	Admin.Faq.Add.Privilege
<b>Requirement</b>	System shall verify that the admin has the right to manage FAQs.
<b>Source</b>	Team Member 2
<b>Rationale</b>	To restrict unauthorized access to FAQs.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 150: Admin.Faq.Add.Answer**

<b>Identifier</b>	FR-150
<b>Title</b>	Admin.Faq.Add.Answer
<b>Requirement</b>	System shall allow the admin to add an answer.
<b>Source</b>	Team Member 1
<b>Rationale</b>	The answer is the admin response to the frequently asked question available in the FAQ section.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	FR-146
<b>Priority</b>	High

**Table 151: Cat Food Inventory Management**

<b>Identifier</b>	FR-151
<b>Title</b>	Cat Food Inventory Management
<b>Requirement</b>	The admin shall be able to view and manage the cat food inventory in the online store, including adding new products, updating existing ones, and removing discontinued items.
<b>Source</b>	Team Member 1
<b>Rationale</b>	To ensure that the online store's inventory is accurately maintained, reflecting the availability of cat food products for purchase.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High



**Table 152: Admin.MonitorSystemHealth**

<b>Identifier</b>	FR-152
<b>Title</b>	Admin.MonitorSystemHealth
<b>Requirement</b>	System shall show an overview of the current system health status.
<b>Source</b>	Team Member 1
<b>Rationale</b>	System health status can be views by admin to know the current load on system, error rates, response times, and other customer requests to the system.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 153: Admin.MonitorSystemHealth.Cancel**

<b>Identifier</b>	FR-153
<b>Title</b>	Admin.MonitorSystemHealth.Cancel
<b>Requirement</b>	System shall allow admin to return to dashboard.
<b>Source</b>	Team Member 1
<b>Rationale</b>	Once admin has viewed system health they can then return to their dashboard for other activities.
<b>Business Rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

## 4.8 Module 8: Statistical Analytics

**Table 154: Generating.Reports**

<b>Identifier</b>	FR-154
<b>Title</b>	Generating.Reports

<b>Requirement</b>	The system should generate the statistical report based on the user-selected time range, providing relevant data such as charts, graphs, tables, and other relevant information.
<b>Source</b>	Team Member 2
<b>Rationale</b>	To allow the user to generate report in specific format
<b>Business rule</b>	Generate the accurate report
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 155: Data.selection**

<b>Identifier</b>	FR-155
<b>Title</b>	Data.selection
<b>Requirement</b>	The user should be able to select the type of data to be included in the report.
<b>Source</b>	Team Member 2
<b>Rationale</b>	Help system to show specific data such as for inventory, cattle, milk
<b>Business rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	High

**Table 156: Report.customization**

<b>Identifier</b>	FR-156
<b>Title</b>	Report.customization
<b>Requirement</b>	The user should be able to customize the report by choosing different colors, fonts, and layout options.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This will help the user to collect data in different options
<b>Business rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Low

**Table 157: User.access.control**

<b>Identifier</b>	FR-157
<b>Title</b>	User.access.control

<b>Requirement</b>	The app should have appropriate user access control mechanisms to ensure that only authorized users can generate reports and view sensitive data.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This will help the system to secure sensitive data.
<b>Business rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Low

**Table 158: Report.Sharing**

<b>Identifier</b>	FR-158
<b>Title</b>	Report.Sharing
<b>Requirement</b>	The app should allow users to share the reports with others via email, messaging, or social media.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This will help the system to share our report with our team members.
<b>Business rule</b>	Report sharing will be accurate and in best quality.
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 159: Report.Automation**

<b>Identifier</b>	FR-159
<b>Title</b>	Report.Automation
<b>Requirement</b>	The app should allow users to schedule the reports to be generated automatically at specific intervals, such as daily, weekly, or monthly, and send them to specific recipients.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This will help user to generate report automatically.
<b>Business rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 160: Graph.Types**

<b>Identifier</b>	FR-160
<b>Title</b>	Graph.Types
<b>Requirement</b>	The feature should allow users to select from a variety of graph types such as line graph, bar graph, pie chart that are relevant for the type of data being presented.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This will help user to generate report graph according to his circumstances
<b>Business rule</b>	Only select those Graph Types, which are in system
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 161: Data.Filtering**

<b>Identifier</b>	FR-161
<b>Title</b>	Data Filtering
<b>Requirement</b>	The feature should allow users to filter the data used to generate the graph, based on specific criteria such as date ranges or specific values.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This will help the user to generate a graph on specific criteria.
<b>Business rule</b>	Filter the data by given tags.
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

**Table 162: Exporting**

<b>Identifier</b>	FR-162
<b>Title</b>	Exporting
<b>Requirement</b>	The feature should allow users to export the graph in a variety of formats, such as PNG, JPEG, or PDF, for sharing or inclusion in other documents.
<b>Source</b>	Team Member 2
<b>Rationale</b>	This will help user to export graph in specific format.
<b>Business rule</b>	Export file only in given format.
<b>Dependencies</b>	FR-99
<b>Priority</b>	Medium

**Table 163: Graph.Interaction**

<b>Identifier</b>	FR-163
<b>Title</b>	Graph.Interaction
<b>Requirement</b>	Users should be able to interact with the graph by selecting data points, highlighting different sections, and toggling on and off certain variables
<b>Source</b>	Team Member 2
<b>Rationale</b>	This will help user to export graphs on specific circumstances.
<b>Business rule</b>	N/A
<b>Dependencies</b>	N/A
<b>Priority</b>	Medium

## 5. Non-Functional Requirements

### 5.1 NFR-1: Reliability

Identifier	NFR-1
Title	Reliability
Availability	System is online all the time 24/7 except in case of some update or error on server side.
Accuracy	The system will show up to date and accurate information to users. Exact accuracy rate can be given after implementation
Error Handling	If there is error processing the image, you just need to refresh the application and wait for 5-6 seconds, and application will run smooth again.

### 5.2 NFR-2: Usability

Identifier	NFR-2
Title	Usability
Adaptability	The interface will have options that are easy to understand and naturally mapped. It will follow Design standards and conventions of commonly used applications so user can adapt easily
Learnability	A person with basic grasp on language can understand. 97% of users will be able to use efficiently after 10 minutes of initial use with 0.5% error rate.
Design	The styling will be clean, and options will be clearly readable. Font style will be clear
Guidance	A help option will direct user to detailed explanation of each aspect of product and application
Visibility	Notification messages will be made clearly visible to user in a seamless way that does not disturb their other activities.

### 5.3 NFR-3: Performance

<b>Identifier</b>	<b>NFR-3</b>
<b>Title</b>	<b>Performance</b>
<b>Response Time</b>	Our system is real-time, it will have fast response time. It will provide a response in max 2-3 minutes. The speed depends on network connection strength as well.
<b>Throughput</b>	The application can perform navigation with streaming or cleaning. The processes run in parallel.

### 5.4 NFR-4: Portability

<b>Identifier</b>	<b>NFR-4</b>
<b>Title</b>	<b>Portability</b>
<b>Platform independent</b>	Our system is tailored for web-based systems, ensuring compatibility across different platforms and responsive performance.

## 6. External Interface Requirements

### 6.1 User Interfaces Requirements

**UI-1:** The system's interface shall abide by vanilla CSS and Bootstrap UI design principles.

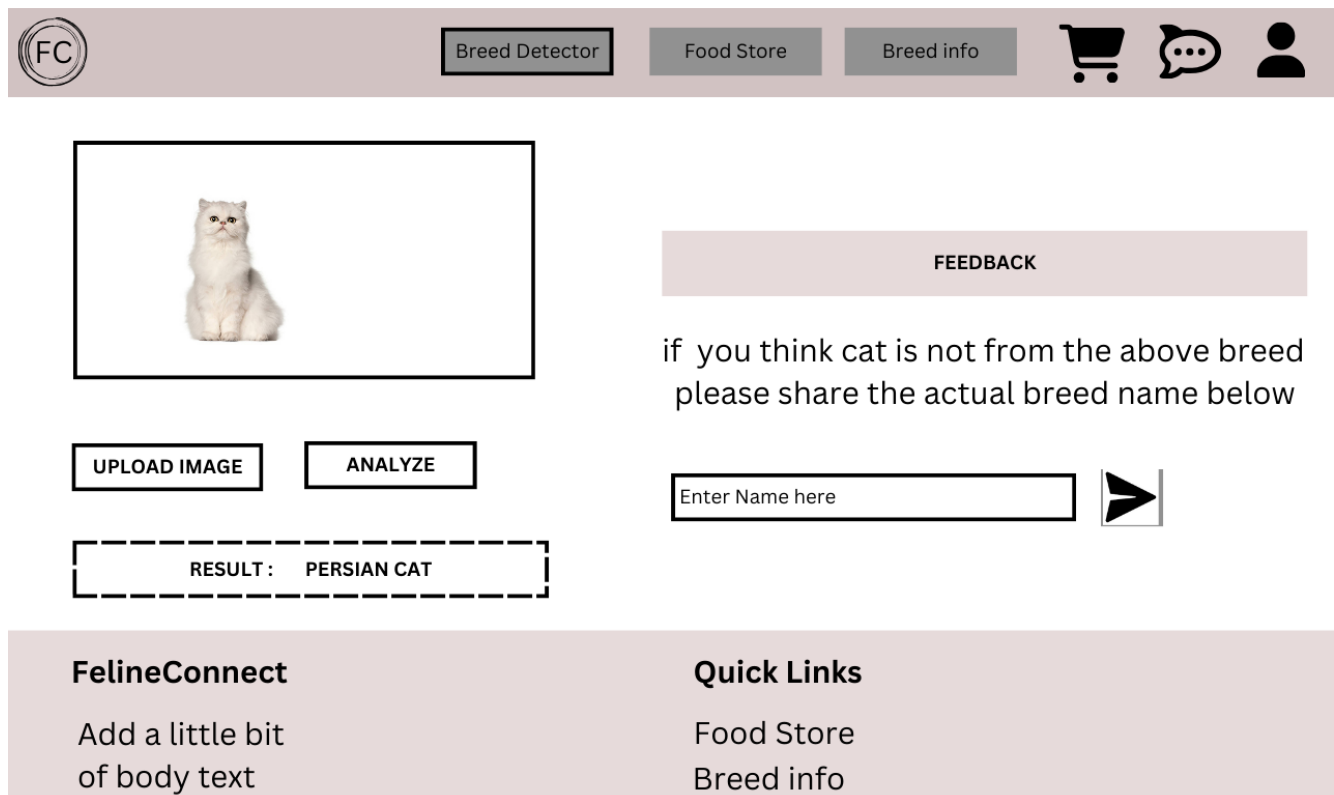
**UI-2:** The system shall use JPEG and PNG vector graphics for icons.

**UI-3:** Every popup, dialog, or page of the system shall let the user easily navigate to previous page, by providing backward navigation buttons.

**UI-4:** The system shall use a consistent coloring scheme for user interface components throughout the app.

**UI-5:** The size of UI elements shall be stated in screen size percentage.

**UI-6:** The system UI layout shall be highly responsive.



The mockup shows a web interface for 'FelineConnect'. At the top is a navigation bar with a circular 'FC' logo, three buttons labeled 'Breed Detector', 'Food Store', and 'Breed info', and three icons: a shopping cart, a speech bubble, and a user profile. The main content area is split into two columns. The left column features a large image placeholder containing a white cat, two buttons labeled 'UPLOAD IMAGE' and 'ANALYZE', and a dashed border box containing the text 'RESULT: PERSIAN CAT'. The right column has a light pink header labeled 'FEEDBACK', followed by the text 'if you think cat is not from the above breed please share the actual breed name below', a text input field with the placeholder 'Enter Name here', and a right-pointing arrow button. The footer consists of two sections: 'FelineConnect' with the text 'Add a little bit of body text' and 'Quick Links' with links to 'Food Store' and 'Breed info'.

**Figure 6.1.1: FelineConnect Breed Recognition Mockup**



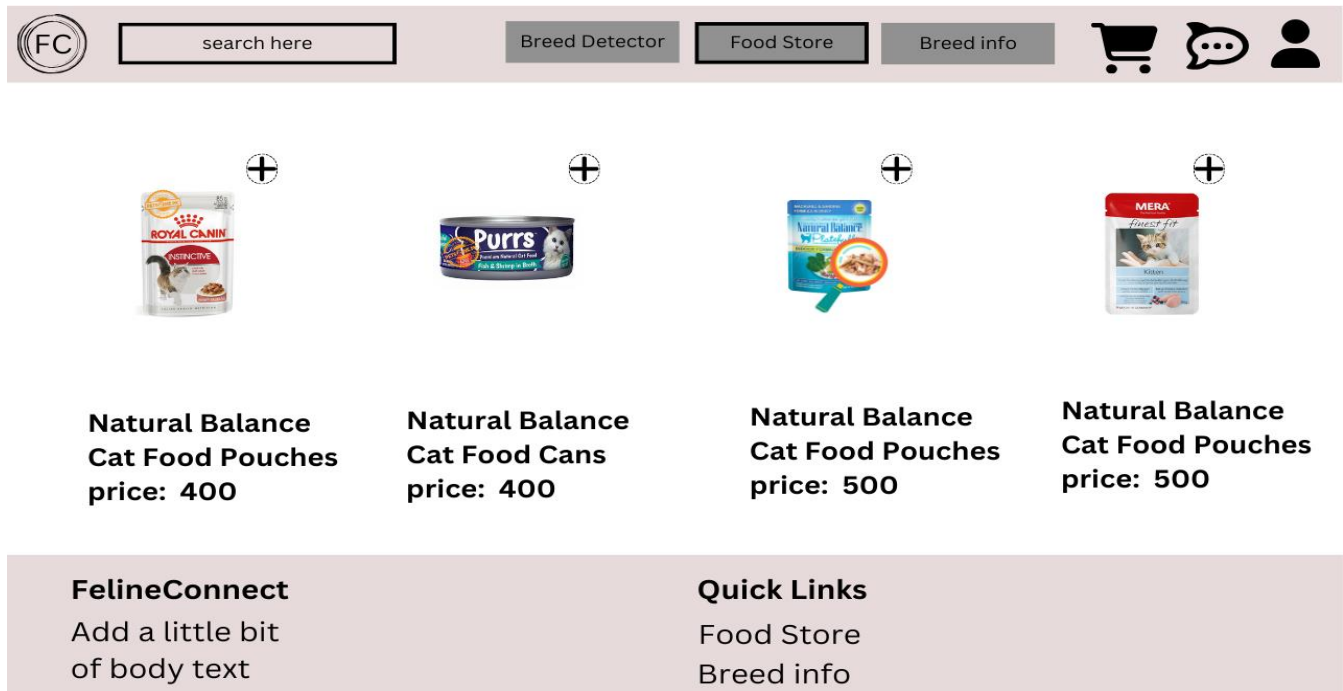


Figure 6.1.2: FelineConnect Food Store Mockup

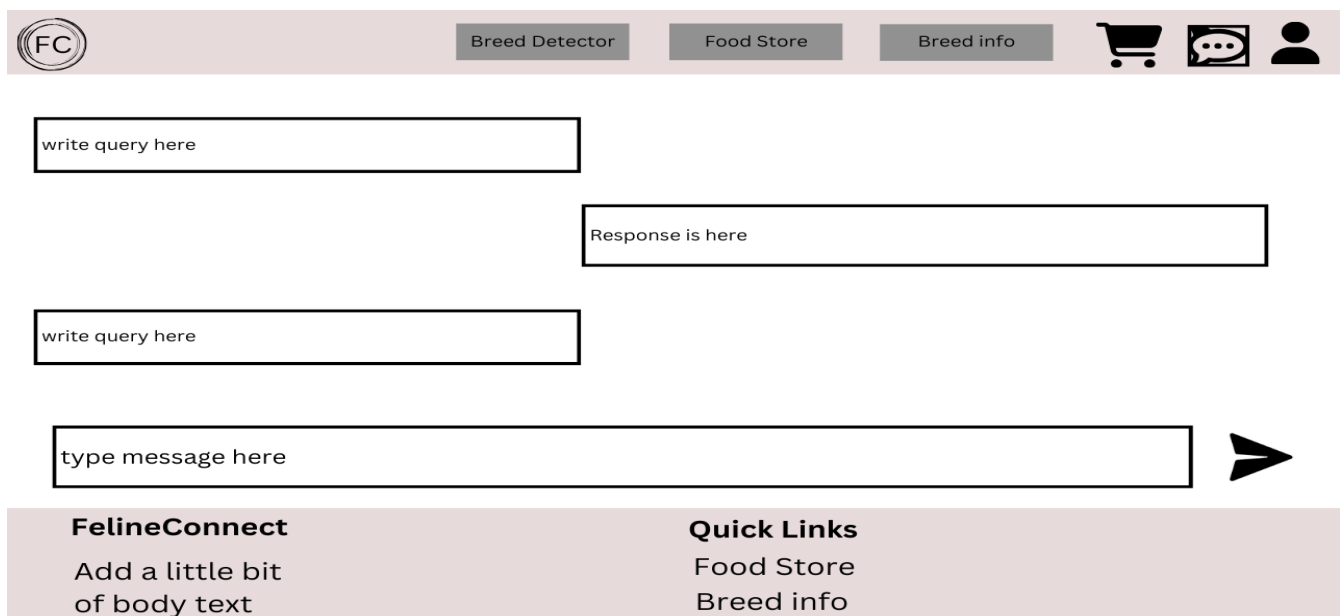
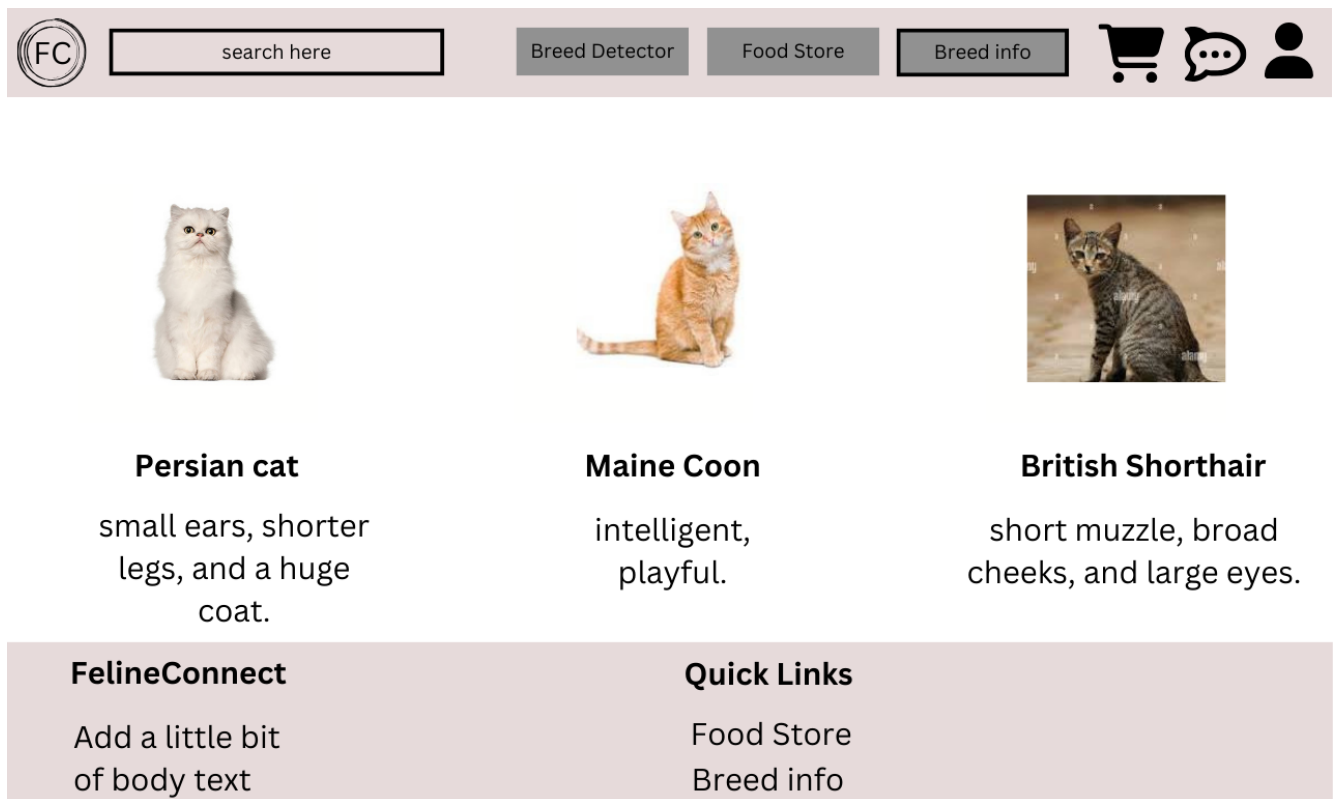


Figure 6.1.4: FelineConnect Chatbot Mockup



**Figure 6.1.3: FelineConnect Breed Info Mockup**

## 6.2 Software interfaces

<b>6.2.1</b>	Canva will be used for creating mockups of the application
<b>6.2.2</b>	React.js will be used for the front-end development
<b>6.2.3</b>	Node Js will be used for backend development
<b>6.2.4</b>	MySql will be used for database
<b>6.2.5</b>	Keras and TensorFlow will be used for model training

## 6.3 Hardware Interfaces

**HI-1:** As the application requires internet to run, all the components related to it can be considered as hardware components of the system such as modem, WAN, Ethernet etc.

## 6.4 Communications interfaces

6.4.1 The system will be sending verification links through emails.
6.4.2 Our system will use the HTTP protocol to communicate with node server.

## 7. References

- <https://ieeexplore.ieee.org/document/8672684>
- <https://universe.roboflow.com/diffran-nur-cahyo-egmuz/cat-breed-classification>
- [https://www.irjmets.com/uploadedfiles/paper/issue\\_5\\_may\\_2023/38683/final/fin\\_irjmets1683962083.pdf](https://www.irjmets.com/uploadedfiles/paper/issue_5_may_2023/38683/final/fin_irjmets1683962083.pdf)
- <https://core.ac.uk/download/pdf/552632009.pdf>
- <https://www.geeksforgeeks.org/cat-dog-classification-using-convolutional-neural-network-in-python/>
- <https://www.kaggle.com/datasets/timost1234/cat-individuals>
- <https://www.kaggle.com/datasets/yapwh1208/cats-breed-dataset>
- <https://www.kaggle.com/datasets/imspars/animal-breed-cats-and-dogs>

# 8. Plagiarism

Document Viewer

## Turnitin Originality Report

Processed on: 25-Mar-2024 9:44 PM PDT  
ID: 2331498998  
Word Count: 1709  
Submitted: 1

Document By Sardar Saifullah

Similarity Index

11%

Similarity by Source

Internet Sources:	7%
Publications:	1%
Student Papers:	11%

include quoted

include bibliography

excluding matches < 3 words

mode:

quickview (classic) report

print

refresh

download

5% match (student papers from 16-Apr-2023)

[Submitted to Lindenwood University on 2023-04-16](#)

3% match (student papers from 14-Nov-2022)

[Submitted to Higher Education Commission Pakistan on 2022-11-14](#)

1% match (student papers from 02-Oct-2018)

[Submitted to Higher Education Commission Pakistan on 2018-10-02](#)

1% match (student papers from 13-Apr-2019)

[Submitted to Higher Education Commission Pakistan on 2019-04-13](#)

1% match (student papers from 07-Jul-2023)

[Submitted to University of Essex on 2023-07-07](#)

<1% match (Internet from 24-Nov-2023)

<https://www.coursehero.com/file/143261309/Smart-Restaurant-scopedocx/>