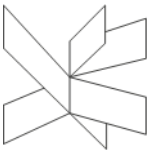


VIAPets - Analysis Document

Summary

The VIAPets company owner asked for a system specialized in handling pet and owner details regarding his combined pet shop and kennel business in Northern Europe. The pet shop hosts a variety of pets such as cats, dogs, fish, birds, and rodents, along with a mixed group labeled as “various”. The kennel’s maximum capacity at the moment is 10 pets, which only holds cats, dogs and birds.

Each pet can be distinguished by their own set of characteristics (favourite food, bite status etc.) The system records the duration of their stay plus the associated kennel charges. As part of the system, bookings and pet availability should be accessible online, along with pictures and prices.

It is essential that the setup is crafted using Java, relying on files instead of blending with a database. This approach ensures simplicity and aligns with the owner’s preference for a straightforward, non-database-dependent setup. There is no need for it to handle payments or manage user logins. Additionally, the owner wants the flexibility to update kennel capacity and booking prices as needed in the future.

By addressing these needs, the system will provide an easy-to-use, efficient tool for managing both the pet shop and kennel operations while meeting the owner's specific requirements.

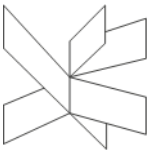
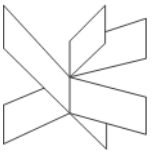
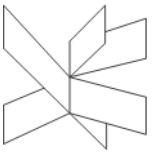


Table of contents



Functional requirements

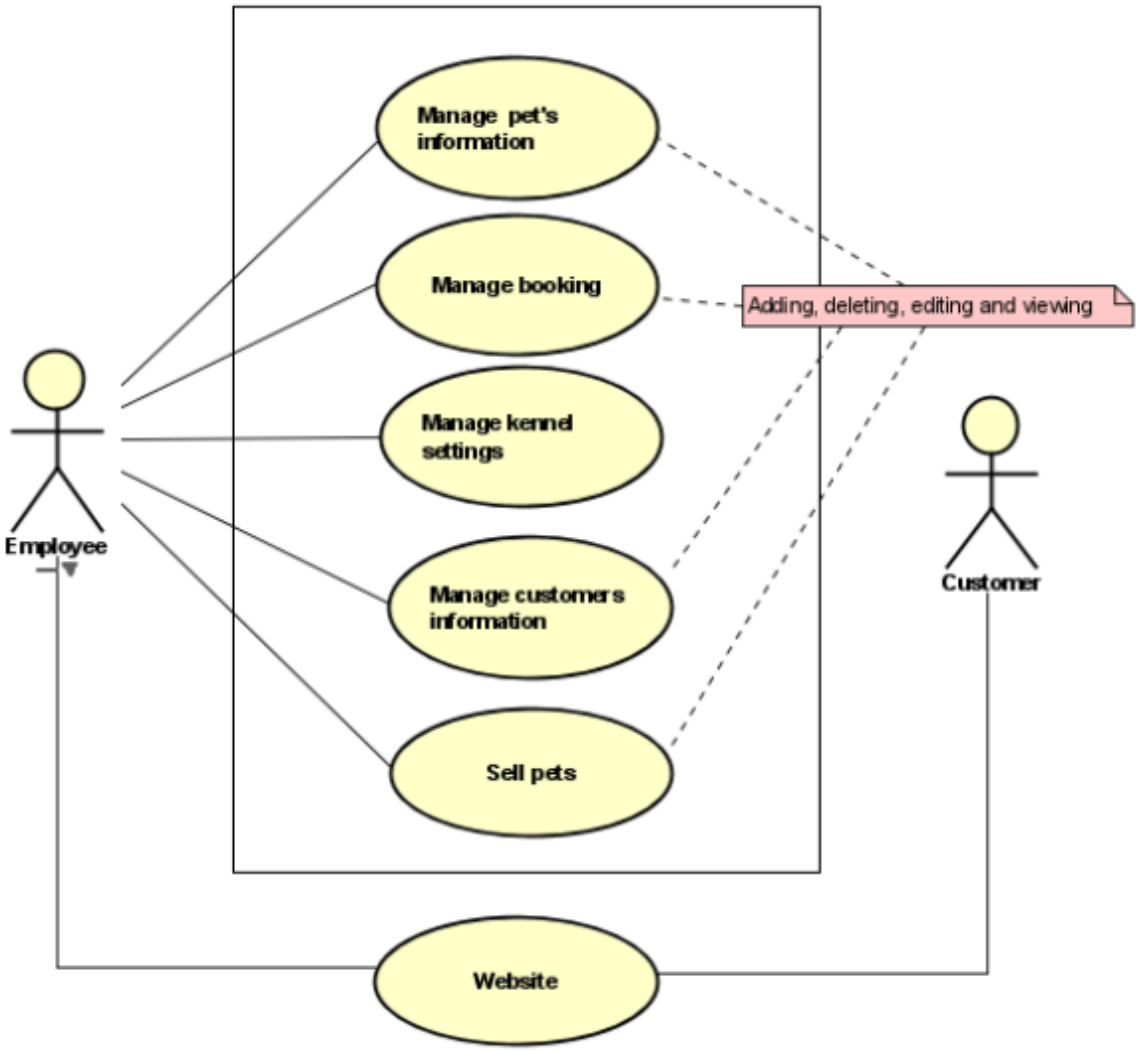
1. As an employee, I want to add pets information to the system so that I can have all information in one place.
2. As an employee, I want to view the details of pets in the system in order to ensure that the pets are in good condition.
3. As an employee, I want to update or edit the pets information in the system so the data is maintained accurately.
4. As an employee, I want to be able to sell pets from the store so that I can change the availability of pets in the store (mark sold).
5. As an employee, I want to register a new customer with a full name and a unique phone number so that a customer can be found by its data.
6. As an employee, I want to add information about customers so that I can find them later in the system.
7. As an employee, I want to view customers' details so that I can quickly access information for customers.
8. As an employee, I want to adopt pets from people who cannot take care of their pets so that the shop can have more pets to sell.
9. As an employee, I want to book a kennel slot so that I can house a pet in that spot.
10. As an employee, I want to access the kennel's availability information and update the available spots for booking so customers receive accurate information.
11. As a customer, I want to view pet's information on the website in order to look for a pet to buy.
12. As a customer, I want to view kennel information on the website in order to check if there is a free slot for the pet in the kennel.

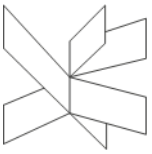


Non-functional requirements

13. The system must be coded in java in IntelliJ IDEA Community Edition 2024
14. The system must store the date in some files, not databases.
15. The system does not have to handle payments or manage bookings online.
16. The system does not have to include login and password features.
17. The total number of pets in the kennel is limited to 10.

Use case diagram



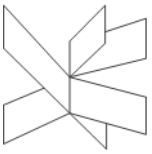


Use case description

<table>
<tr><td>Use case</td><td>Manage pet's information</td></tr>
<tr><td>Summary</td><td>Adding, removing, editing and viewing the pets' traits and availability</td></tr>
<tr><td>Actor</td><td>Employee</td></tr>
<tr><td>Precondition</td><td>Scenario A: The pet must not already exist in the system
Scenario B: The pet to be deleted must exist in the system and the pet must not have any active bookings or ongoing sales linked to it.
Scenario C: The pet to be edited must exist in the system and the employee wants to change something about it.
Scenario D: There must be at least one pet record in the system.</td></tr>
<tr><td>Postcondition</td><td>Scenario A: A new pet record is created in the system with all the provided details
Scenario B: The pet record is removed from the system
Scenario C: The pet record is updated with the new information provided
Scenario D: The system displays the selected pet's details without modifying any data.

Details are changed and stored inside a file.</td></tr>
<tr><td>Base sequence</td><td>1. System lists all pets and their information.

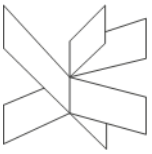
<table><tr><td>Scenario A. : ADD:</td><td>A.1. Add a new pet to the list.
A.2. Add new pets information (ID,name, species, color, gender, age, name, notes).
A.3. Add extra information IF they are cats/dogs - the breed and breeder.
A.3. Add extra information IF they are fish - type, whether or not they hunt other fish and whether they live in saltwater or freshwater.
A.4. Add extra information IF</td></tr></table></td></tr>
</table>



	<table><tr><td></td><td>they are birds - type and what they like to eat. A.5. Add extra information for rodents - kind and if they bite. A.6. Add a category named “various” for the rest of the pets (pets not mentioned above). [ALT3.6] A.7. Add price.</td></tr></table> <table><tr><td>Scenario B. : DELETE:</td><td>B.1. Select an existing pet from the list. B.2. Remove the pet</td></tr></table> <table><tr><td>Scenario C. : EDIT:</td><td>C.1. Select an existing pet from the list. C.2. Change the pets information (ID, name, species, color, gender, age, name, notes and extra details based on the type of pet)</td></tr></table> <table><tr><td>Scenario D. : VIEW:</td><td>D.1. Select an existing pet from the list. D.2. View information.</td></tr></table>
Alternate sequence (branch or exception)	[*ALT0] The process can be canceled in all steps with user interaction. Use case ends.
Note	There should be information that separates pets in the shop from the pets in the kennel. This Use case covers requirements: 1,2,3,4,9

<table>
<tr><td>Use case</td><td><h2>Manage booking</h2></td></tr>
<tr><td>Summary</td><td>Adding, viewing, editing and removing information about booking in the kennel</td></tr>
<tr><td>Actor</td><td>Employee</td></tr>
<tr><td>Precondition</td><td>Scenario A: Kennel spot is available and the customer must exist in the system.
Scenario B: The customer booked a kennel spot and employee wants to view the information
C: The customer booked a kennel spot and employee wants to edit the information
D: The customer booked a kennel spot and employee wants to remove the information</td></tr>
<tr><td>Postcondition</td><td>Details are added or changed and stored inside a file.</td></tr>
<tr><td>Base sequence</td><td>1. System lists all kennel bookings (if there are any) and information about them.

<table><tr><td>Scenario A. ADD</td><td>A.1. Add a new booking to the list by entering customerID, petID, start date and end date.[ALT1]</td></tr></table>
<table><tr><td>Scenario B. VIEW</td><td>B.1. Select an existing booking from the list.
B.2. View information.</td></tr></table>
<table><tr><td>Scenario C. EDIT</td><td>C.1. Select an existing booking from the list.
C.2. Change customer information by entering different customerID, change pet information by entering different petID, change start date and end date.</td></tr></table>
<table><tr><td>Scenario D. DELETE</td><td>D.1. Select an existing booking from the list.
D.2. Remove booking.</td></tr></table>
2. System closes the list.
3. Use case ends.</td></tr>
<tr><td>Alternate sequence (branch or exception)</td><td>[*ALT0]
The process can be canceled in all steps with user interaction. Use case ends.
[ALT1] If the pets or the customer does not exist, add a pet by going</td></tr>
</table>



	to scenario A in Manage pets' information or adding a customer by going to scenario A in Manage customers' information.
Note	This Use case covers requirements: 1, 2, 3, 5, 6, 7, 9

Use case	Manage kennel settings
Summary	Changing the maximum kennel slot and booking price for the day
Actor	Employee
Precondition	Expanding the kennel slots or changing the booking price
Postcondition	Settings are changed and stored inside files.
Base sequence	1. System displays current settings. 2. Change maximum kennel slots or/and booking price for a day. 3. System closes the list. 4. Use case ends.
Alternate sequence (branch or exception)	[*ALT0] The process can be canceled in all steps with user interaction. Use case ends..
Note	This Use case covers requirements: 10

Use case	Manage customers information
Summary	Adding, viewing, editing and removing information about customers
Actor	Employee
Precondition	Scenario A: The customer must not already exist in the system Scenario B: The customer to be deleted must exist in the system. Scenario C: The customer to be edited must exist in the system and the customer changed some information. Scenario D: There must be at least one customer record in the system.

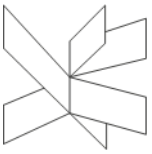
<table>
<tr><td>Postcondition</td><td>Scenario A: A new customer record is created in the system
Scenario B: The sale to be deleted must exist in the system.
Scenario C: The customer's record is updated with the new information provided.
Scenario D: The system displays the selected customer’s details without modifying any data.</td></tr>
<tr><td>Base sequence</td><td>1. System lists all customers (if there are any) and information about them.

<table><tr><td>Scenario A. : ADD:</td><td>A.1. Add a new customer to the list.
A.2. Add new customer’s information (full name, phone number, email). [ALT1]</td></tr></table>
<table><tr><td>Scenario B. : DELETE:</td><td>B.1. Select an existing customer from the list.
B.2. Remove the customer from the list.</td></tr></table>
<table><tr><td>Scenario C. : EDIT:</td><td>C.1. Select an existing customer.
C.2. Change the customer’s information (full name, phone number, email)</td></tr></table>
<table><tr><td>Scenario D. : VIEW:</td><td>D.1. Select an existing customer..
D.2. View information.</td></tr></table></td></tr>
<tr><td>Alternate sequence (branch or exception)</td><td>[*ALT0] The process can be canceled in all steps with user interaction. Use case ends.
[ALT1] If the system detects that the customer’s data has already been added, it shows a message indicating that there is already a customer linked to the information inputted (full name and phone number).</td></tr>
</table>

<table>
<tr><td>Use case</td><td>Sell pets</td></tr>
<tr><td>Summary</td><td>Upon selling a pet, the pet's record is updated to show it is no longer available, with its status marked as sold. This update is applied in both the internal system and stored files, and the pet is removed from the list of available pets.</td></tr>
<tr><td>Actor</td><td>Employee</td></tr>
<tr><td>Precondition</td><td>Scenario A: The pet to be sold information must already be registered in the system and be available for sale. As well the customer's details must already be registered in the system.
Scenario B: The sale to be deleted must exist in the system.
Scenario C: The sale to be edited must exist in the system and the employee wants to edit the information.
Scenario D: There must be at least one sale recorded in the system.</td></tr>
<tr><td>Postcondition</td><td>Scenario A: A new sale record is created, and the pet is marked as "sold".
Scenario B: The sale record is deleted from the system, and the pet’s status may return to "not sold”.
Scenario C: The sale record is updated with the new information provided.
Scenario D: The system displays the details of the selected sale.

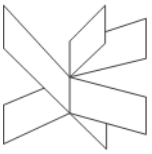
Changes have been made both in the system and in the file.</td></tr>
<tr><td>Base sequence</td><td>1. System lists all sales (if there are any) and information about them.

<table><tr><td>Scenario A. : ADD:</td><td>A.1. Add a new sale to the list.
A.2. Add new sale’s information (petID, customerID, salePrice).
[ALT1]</td></tr></table>
<table><tr><td>Scenario B. : DELETE:</td><td>B.1. Select an existing sale from the list.
B.2. Remove existing sale.</td></tr></table></td></tr>
</table>



	<table><tr><td>Scenario C. : EDIT:</td><td>C.1. Select an existing sale from the list. C.2. Change the sale's information (petID, customerID, salePrice)</td></tr></table> <table><tr><td>Scenario D. : VIEW:</td><td>D.1. Select an existing sale D.2. View information.</td></tr></table>
Alternate sequence (branch or exception)	[*ALT0] The process can be canceled in all steps with user interaction. Use case ends. [ALT1] If the system detects that the pet has already been marked as sold or if the pet is in the kennel, it shows an error message
Note	This Use case covers requirements : 2, 3, 4, 5, 6, 7, 8

Use case	Website
Summary	The customer sees pets for sale and kennels availability.
Actor	Employee, Customer.
Precondition	Pet and its information exist.
Postcondition	New booking for kennel, customer asks employee for a pet.
Base sequence	1. Gets into the kennel/shop page and sees the kennels availability or pets for sale. [ALT2] 2. There is an available spot at the kennel and the customer asks the employee for that spot. / The customer finds a pet they like and asks the customer. 3. The employee books the kennel for the customer.



Alternate sequence (branch or exception)	[ALT2] There are no free slots on the kennel so the user cannot book it successfully. / There are no more pets for sale so the customer cannot buy any pets.
Note	This Use case covers requirements : 11, 12

Relation between requirements and use cases

Use cases	Covered requirements
Manage pet's information	1,2,3,4,9
Manage booking	1, 2, 3, 5, 6, 7, 9
Manage kennel settings	10
Manage customers information	5,6,7
Sell pets	2, 3, 4, 5, 6, 7, 8
WEBSite	11, 12