**PRODUCT DESCRIPTION**

# Pet Care

# Product Scope

An app that enables pet owners to manage their pet’s grooming and health, by providing them with all the required information in one place, such as symptoms of common ailments, possible ailments their pet might be suffering from, prevention of common ailments, as well as treatments available (based on the symptoms input by the user), along with the recommendations to visit the nearby vet in an emergency and store all medical records that can be instantly accessed.

The app also come with a wearable IoT device that is attached to a pet’s collar. Beside collecting location data, the device also have motion, heart rate, and breathing sensors, as well as microphones and camera. These functions help owners keep track on their pets’ physical activity, or understanding the pets’ emotion and the store owner get the pet’s profile when the pet owner register the store owner‘s services

More, the app included a Pet Social Network developed to allow pet owners to interact with, share details and updates related to their pets with pet care providers and other pet owners. Pet owners can also find the best pet store which is selling a variety of items such as pet food, grooming products like shampoos and brushes, litter boxes, bedding, and toys.

**Key objectives:**

**Unauthorized user:**

* Register
* Login (Email)
* Get back forgotten password

**Authorized user:**

* Login
* Logout
* Change profile information

**Pet Owner:**

* Add a new pet
* Remove a pet
* Update pet profile
* Check pet status in real time
* Find pet on the map.
* Book an appointment with services for a pet
* Cancel an appointment
* Check store/services provider information
* Get notification when an appointment is near
* Add/Remove product to cart
* Use 3rd party payment for online payment.
* View order/book history
* Add store/product to favorite list.
* Review the service’s store/store’s product
* Give feedback of the app.
* Post a status to the pet’s social network
* Like the other post
* Comment in the post.
* Share the post.
* Add friend with other pet owner
* Chat with other pet owner

**Store Owner:**

* Add a service.
* Update the service.
* Delete a service.
* Add a product
* Update product information
* Delete a product
* View customer’s review.
* View orders/upcoming pet appointments
* Get notification when an appointment or an order is created.

**Administrator:**

* View the app’s feedback
* Install, update, tune, diagnose both internal and third-party applications.
* View income of the app

# Stakeholders

|  |  |  |
| --- | --- | --- |
| Customer Class | Expectation | Represented By |
| Pet owner | This includes whoever will take a product and use them operationally | User who has use the app and has pet |
| Admin | They are critical to keeping the applications your organization relies on running | Application Administrators aren't developers and they're not user. |
| ****Store owner**** | This is a catch-all and includes those that provide the pet’s services and product | Any pet store, vet. |
|  |  |  |

# Functional Requirements

|  |  |  |
| --- | --- | --- |
| Requirement # | Description | Related Requirements |
| UR1 | Users can create information about their pet profiles (weight/ kind of pet/…). |  |
| UR2 | Users can book an appointment for their pet base on the service. |  |
| UR3 | Users can pay the services online |  |
| UR4 | Users can check the store owner’s products |  |
| UR5 | Users can rate and comments about the store’s service (Check pet’s health, Pet spa and a lot more ) and products |  |
| UR6 | Users can find the nearest store on the map |  |
| UR7 | Users can track their pet location through the collar | UR1 |
| UR8 | Users can keep track on their pets’ status(weight, hunger, health condition,…) through the collar. | UR1 |
| UR9 | Users can make friend and chat with other users |  |
| UR10 | Users can chat with store owner | StOR5 |
| AR1 | Manage store owner’s list |  |
| AR2 | Manage the ratings and comments of the service(Check pet’s health, Pet spa and a lot more ) of the store owner | UR5 |
| AR3 | Chat with Users and Store owner when there is a problem | UR10, StOR5 |
| StOR1 | Store owners can create the services (Check pet’s health, Pet spa and a lot more ) their store offers. |  |
| StOR2 | Store owners can edit or delete the services (Check pet’s health, Pet spa and a lot more )that their store already has on the system. |  |
| StOR3 | Store owners can create a new product to showcase |  |
| StOR4 | Store owners can edit or delete the products that their store already has on the system. |  |
| StOR5 | Store owners can chat with user | UR10 |
| StOR6 | Store owners can receive the status and information about the pet that the pet owner register with | UR1 |

# Non-functional Requirements

## Targeted Platforms

- The application will be deployed both on the mobile and web platform.

- The web platform is built on ReactJS framework for front-end and .NET Core for back-end.

- The mobile platform is built on Flutter framework.

## Performance

* The system must process and respond to the data flows quickly and accurately and shall take no longer than a few seconds to appear on the screen.
* The system must process and respond to errors, including expected errors and unexpected errors.
* The system must be able to handle and accommodate a high number of books and users.

## Availability

-The server shall be working 24 hours per day and 7 days per week if there is no unexpected critical bugs.

- If there is a crash, some function will be disable for a specific of time until the problem is fixed

## Reliability

- There is no requirement for system maintenance task from the user.

- Mean Time Between Failures (MTBF): more than 6 months.

- Maximum Bugs and Defect Rate: 0.5 bugs per thousand lines of code (0.5bugs/KLOC).

- Critical bugs:

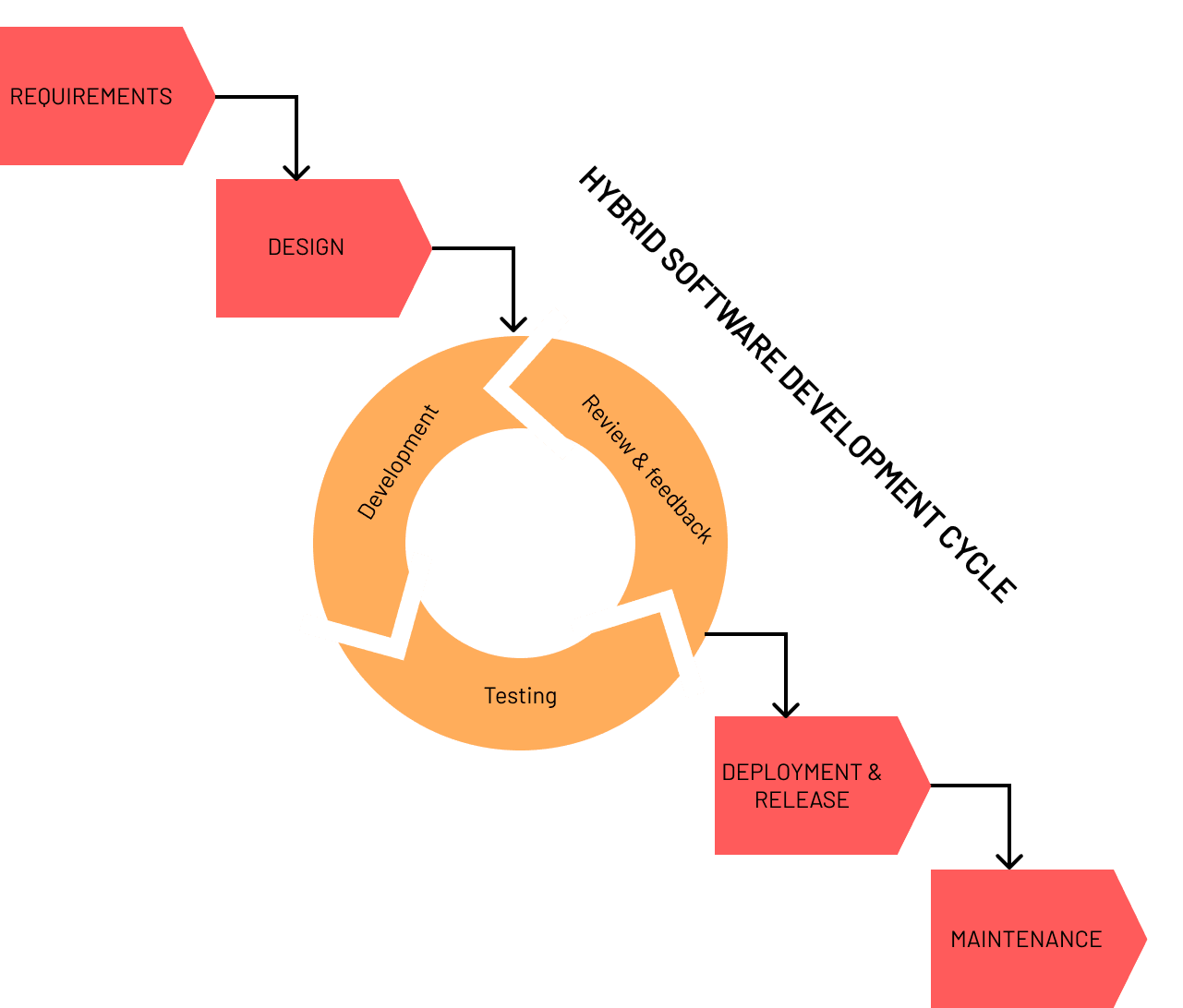
- Loss of data: not any.

- Server crash: probably if any 3rd party use is unavailable

# Release Method

- The project will be developed by the hybrid of Waterfall and Agile

- When we combine the two approaches into one methodology, it ends up looking like this:



- The project follows a lot of the traditional path of a waterfall project but also gives the stakeholders a chance to give feedback during a series of development iterations.

# Use Case Diagram

