University of Concepción
Faculty of Engineering

FINAL PROJECT:
PET STORE

Professor : Zheng Li
Student : Juan Fecci

Index.

Objectives

> “The golden rule: can you make a change to a service and deploy it by itself without changing anything else?”
> — Sam Newman, Building Microservices

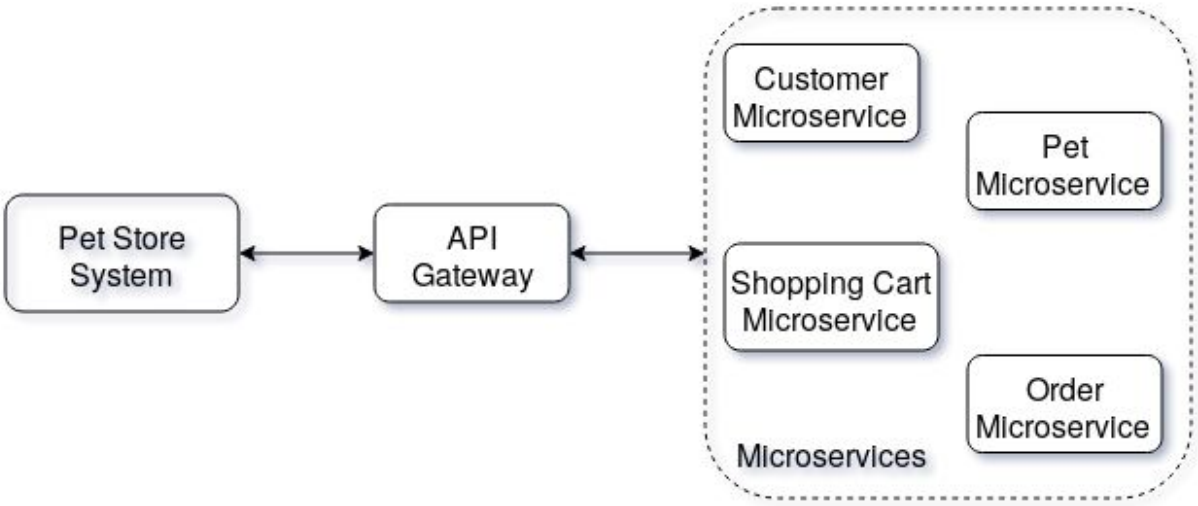
The objective of this work is to implement a pet sales system called "Pet Store" using the different microservices developed during the class "Introduction to Microservices Architecture" using the Api Gateway pattern.

The Pet store system meets the following objectives:

- Sign in and sign out of accounts
- Submit a catalog of available pets sorted by categories
- Implementation of a shopping cart
- Realization of purchase orders
- List of orders made by the user

System Model

The structure of the system used in the Pet Store system can be seen in the following diagram.



Description

The pet store system is developed in Python using Flask. In addition, is implemented in Gcloud so that it is available online

Using the same language, within the same system is included the "API Gateway" to be able to make the communication between the microservices

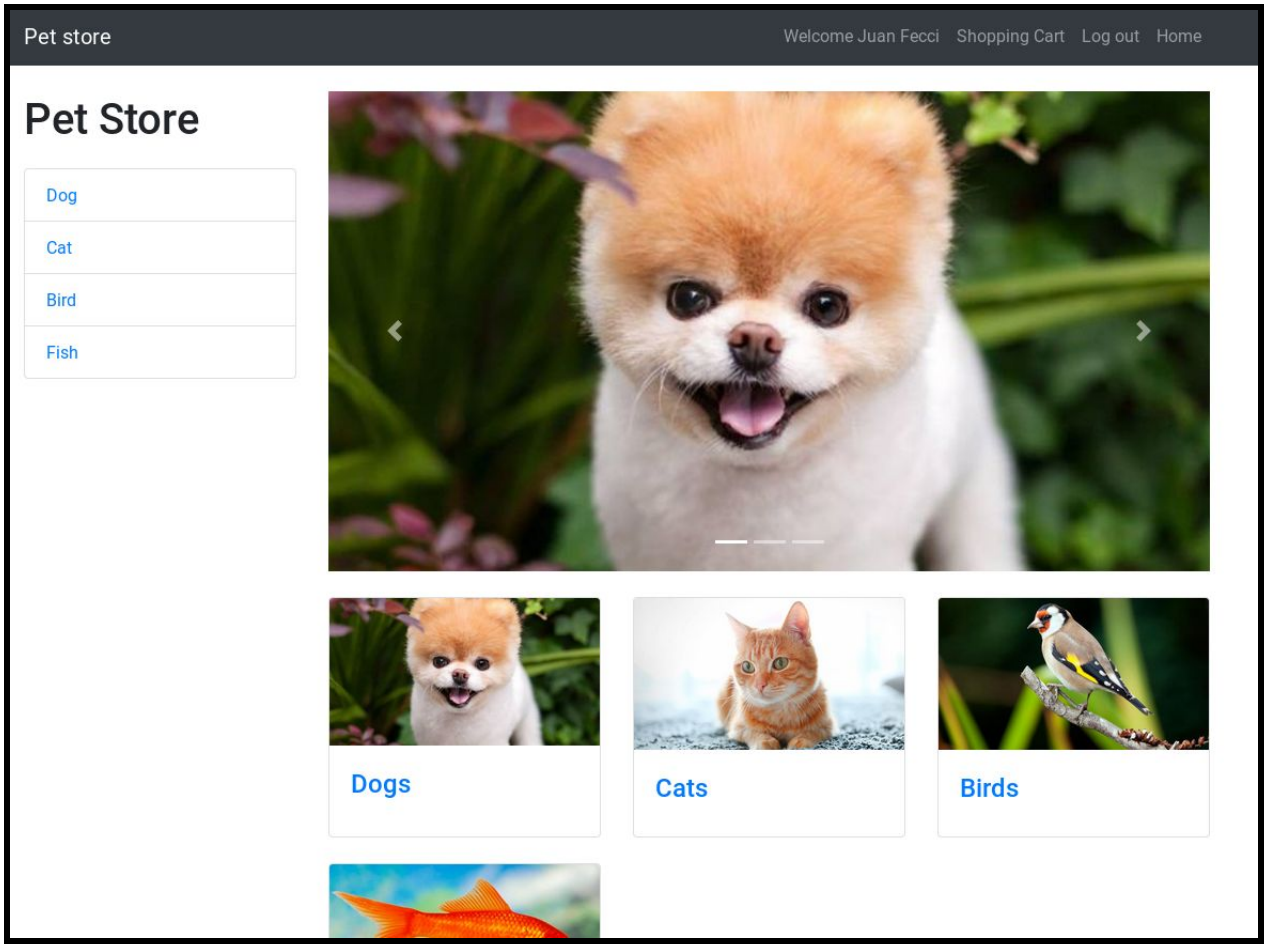
The details of the microservices are the following

- Customer Microservice: Includes the management of system accounts. The api corresponds to http://petstorecustomer.appspot.com/
- Pet Microservice: Includes pets available through the store. The api corresponds to http://practiceiv-on-gcloud.appspot.com/
- Shopping Cart Microservice: It is dedicated to the management of the shopping cart. The api corresponds to http://petstorecart.appspot.com/
- Order Microservice: It is dedicated to the management of the orders made by the user. The api corresponds to http://order-by-fecci.appspot.com

Description of the site

The Pet Store system presents a series of views that meet the established objectives, which can be seen at https://petstorefecci.appspot.com. The views with the following:

Home Page



The homepage of the system includes the 4 categories of species offered by the store. In the upper part there is a bar that will be present in all the pages of the system. This presents the following options:

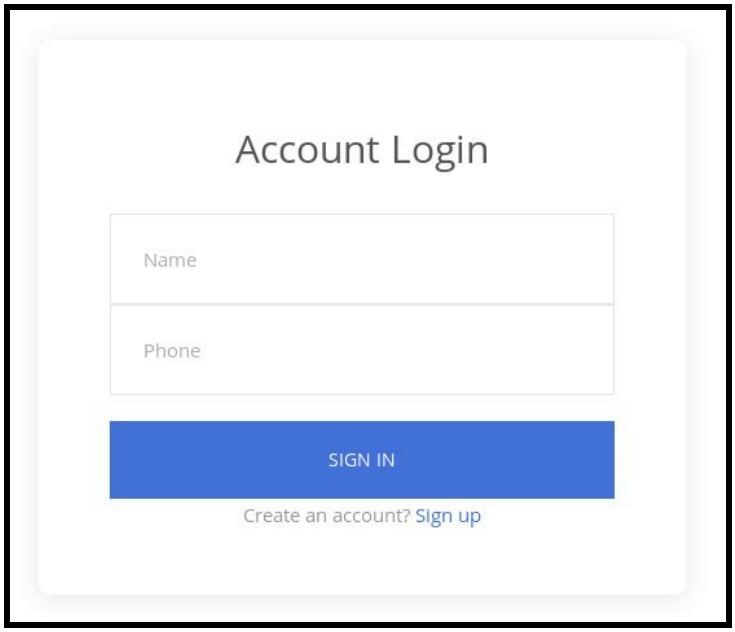
If an account was not entered:

- Login: Allows the entry of an account into the system.
- Register: Allows the registration of an account to the system.
- Home: Redirects to the home page.

If you entered with an account:

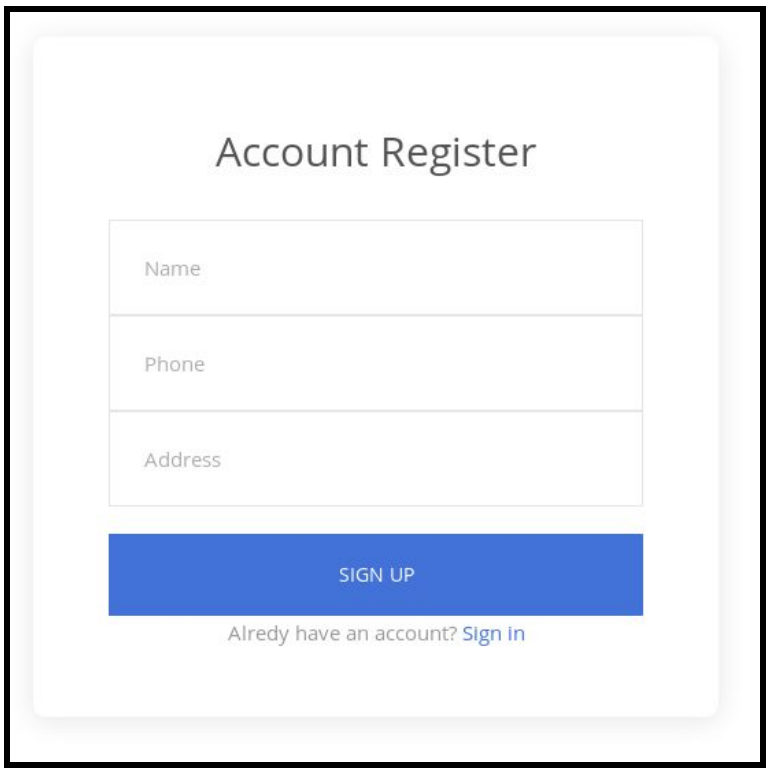
- Welcome [name]: Allows you to see the details of the account and the orders made.
- Shopping Cart: Allows you to observe the user's shopping cart.
- Log out: Allows logging out of the account.
- Home: Redirects to the home page.

Login Page



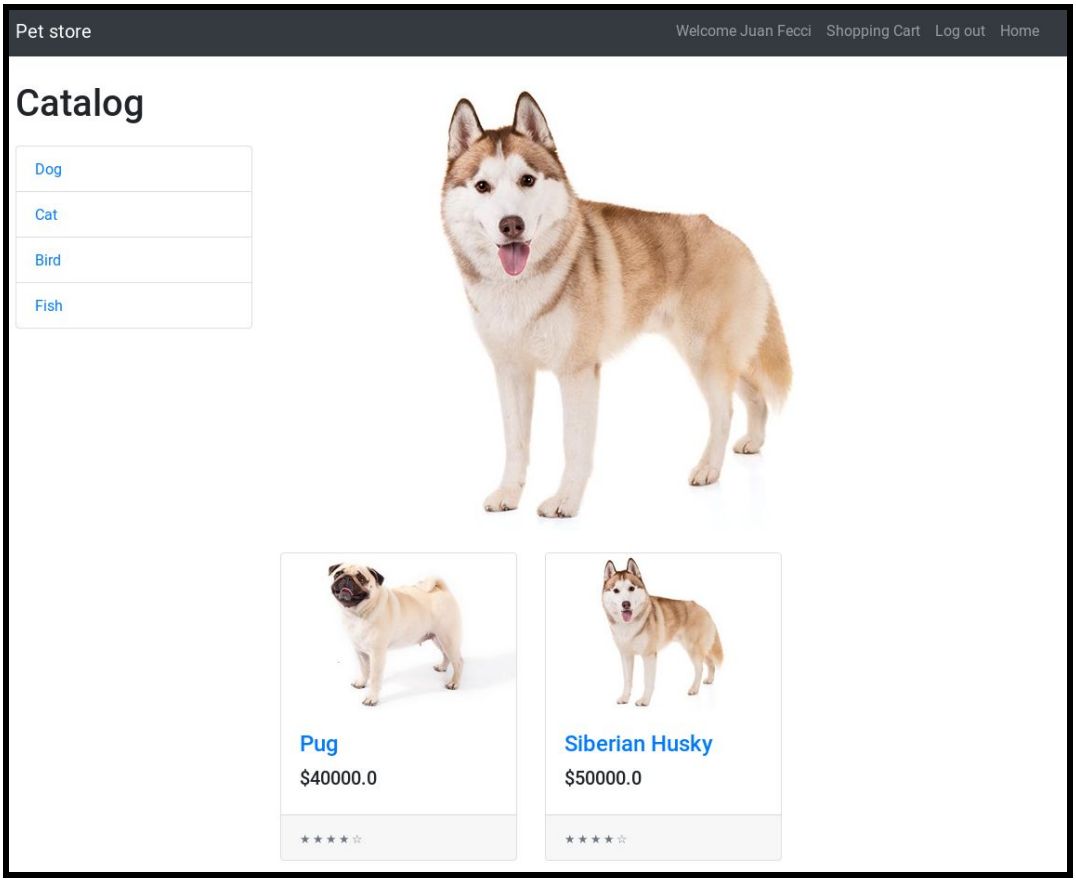
Corresponds to the page to be able to enter with an account to the system using the name and the number phone

Registration Page



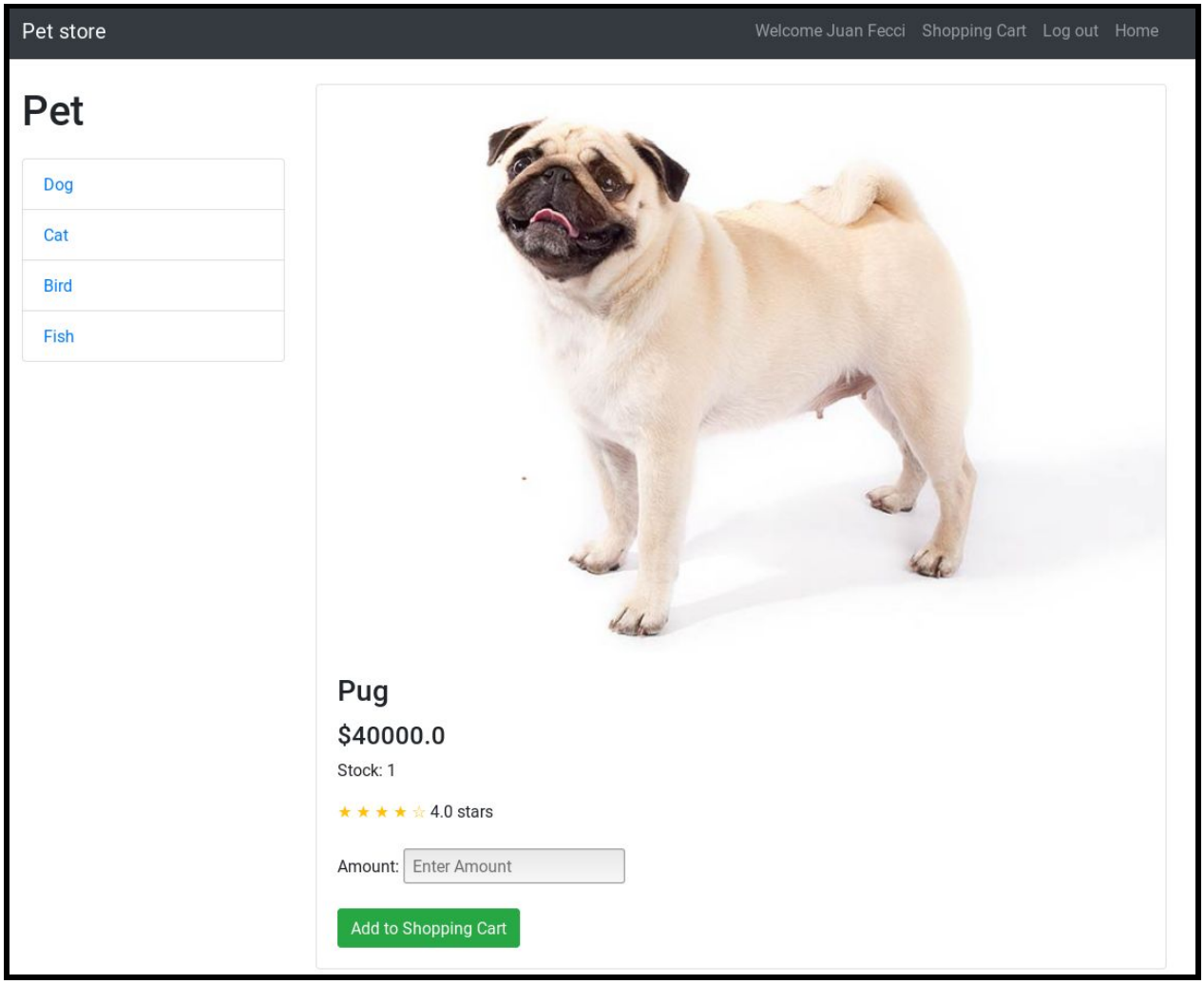
Corresponds to the page to be able to register with an account to the system using the name, number phone and address

Catalog Page



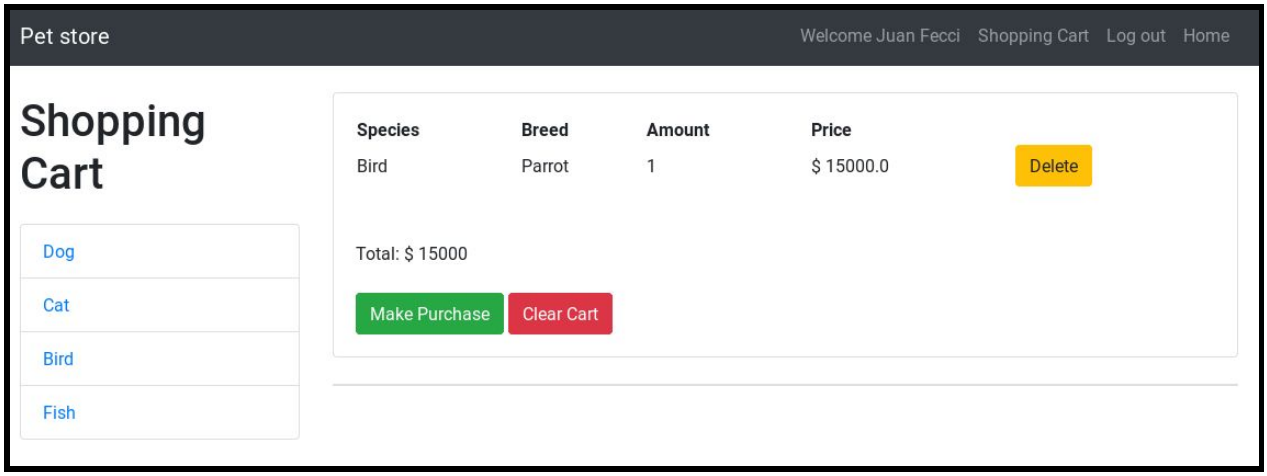
The catalog page of the selected species shows the different pets available in the store.

Pet page



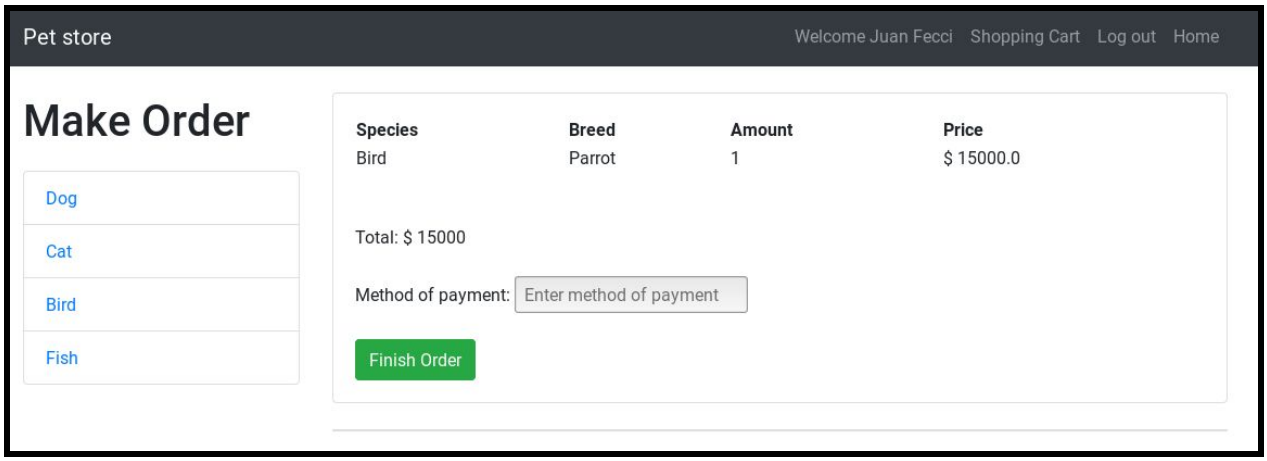
The page of the selected mascot shows the price and the stock of this and includes the option of entering the amount of pets needed by the user. Only a user who is logged in can make the purchase of the pet, for it must enter it to the shopping cart

Shopping Cart Page



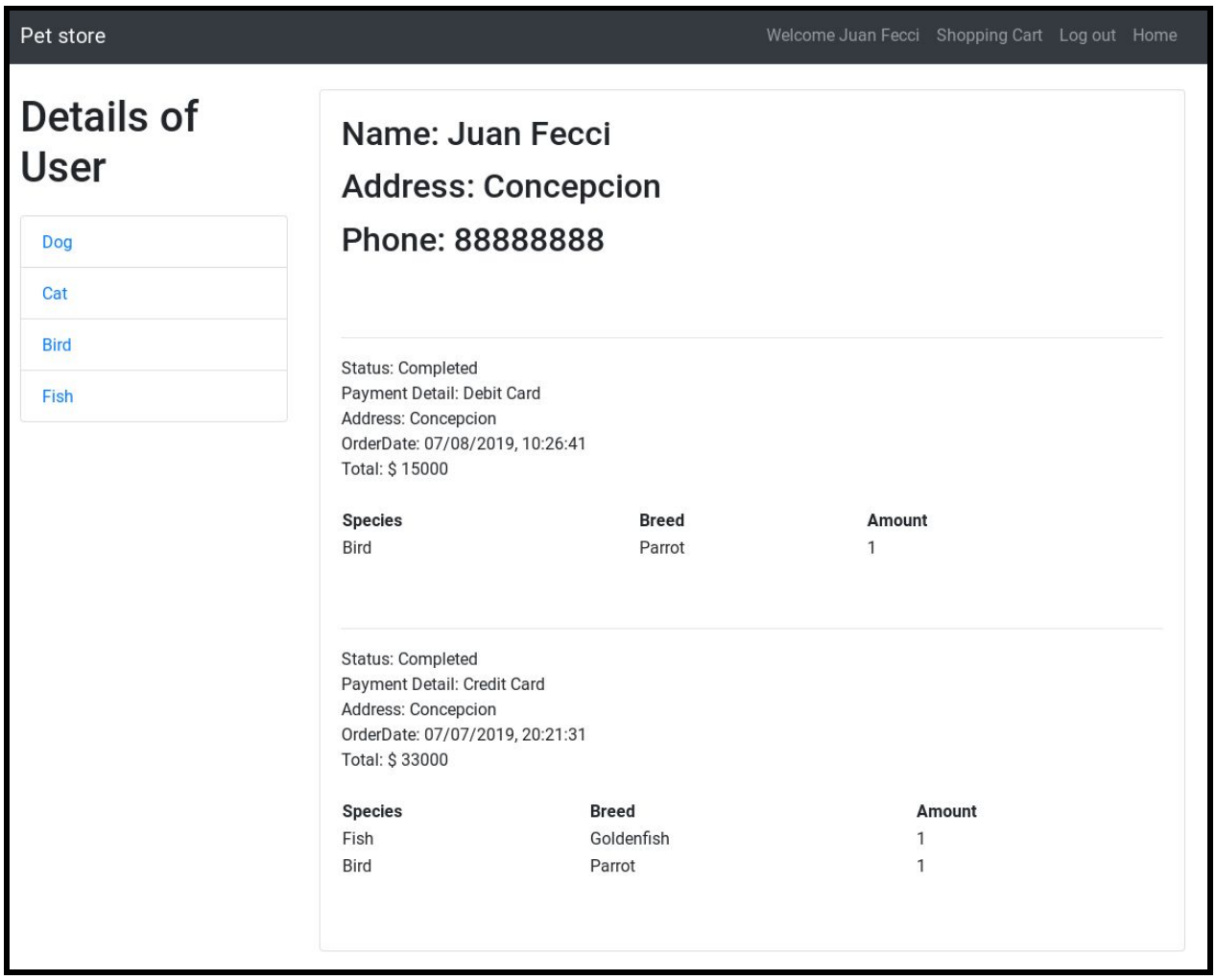
This page shows the shopping cart that the user has to then make the purchase of what was entered. It shows the detail of the pets added, the total to be paid and the option to make the purchase

Make order page



This page allows you to make a simulation of a payment by entering the payment method and finalizing the order.

Account Page



This page shows the details of the logued account, such as your name, address and telephone number, as well as the details of the orders already made.

Installation

To perform the installation correctly, it is necessary that the user have an account in Google and the system have python installed with the module virtualenv.

1. Download, install and initialize with your project the Google's Cloud SDK following the nexts instructions: https://cloud.google.com/sdk/docs/

2. Start a google cloud project using the following command

```
gcloud init
```

3. Acquire local credentials for authenticating with GCP services using this command:

```
gcloud auth application-default login
```

4. Clone the repository of the pet store system with this command:

```
git clone https://github.com/juanfecci/petstore.git
```

5. Install the dependencies of the system using the requirements.txt. If you want to create a virtual environment, use the following commands:

```
virtualenv -p python27 env
source env/bin/activate
pip install -t lib -r requirements.txt
```

Running the app

There are 3 ways for run the app:

Running on your local machine

1. With the virtual environment activated, start the local server with this command:

```
dev_appserver.py app.yaml
```

2. In your browser, enter the following address for use the system: http://localhost:8080

Deploying the app

1. Deploy with the next command

```
gcloud app deploy
```

2. In your browser, enter the following address for use the system: https://[projectName].appspot.com *replace [projectName] with the name of your project

Use the deployed app

You can access to the deployed app with this address:

```
https://petstorefecci.appspot.com
```

Conclusion

In this project, the pet store was successfully implemented using the different microservices available. It was possible to observe that with success the system was developed using the API gateway pattern.

It is important to emphasize that the development was easier thanks to the fact that the microservices of the functions of the system were already implemented, allowing the parallel development of the different microservices and demonstrating the benefits of the architecture