



Proyecto Integrador

GRUPO- 09



WESPED



**Certified Tech
Developer**
The Ultimate Degree

El equipo de Desarrollo



Alejandra Marín
Infra-Back-BBDD



Andrés Fajardo
Testing-Back-Front-BDD



Dina Perez
Testing-Front



Indira Réquiz
BDD-Front -Testing



Juan Basualdo
Front



Romina Vilatta .M
Front



Pablo Alarcón
Back

Proyecto Integrador

Nos dieron un proyecto

Nuestro proyecto integrador para culminar el primer año de CTD es una web de reservas de hoteles

Hicimos un manual de marca

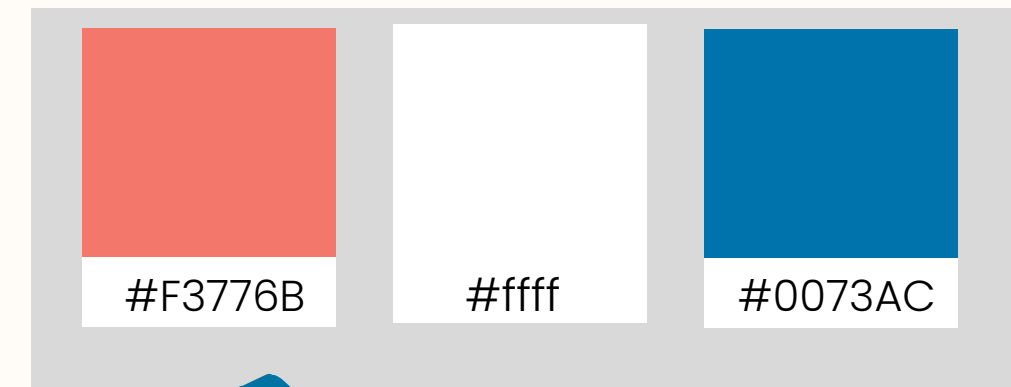
Aa

QUICKSAND

ABCDEFGHIJKLMNOPQRSTUVWXYZ
TUVWXYZabcdefghijklmnopqrstuvwxyz1234567890
!@#\$%^&*()

Creamos una marca

Wesped es nuestro producto.
Su nombre remite a la palabra Huésped, "aquel que recibe y es recibido".



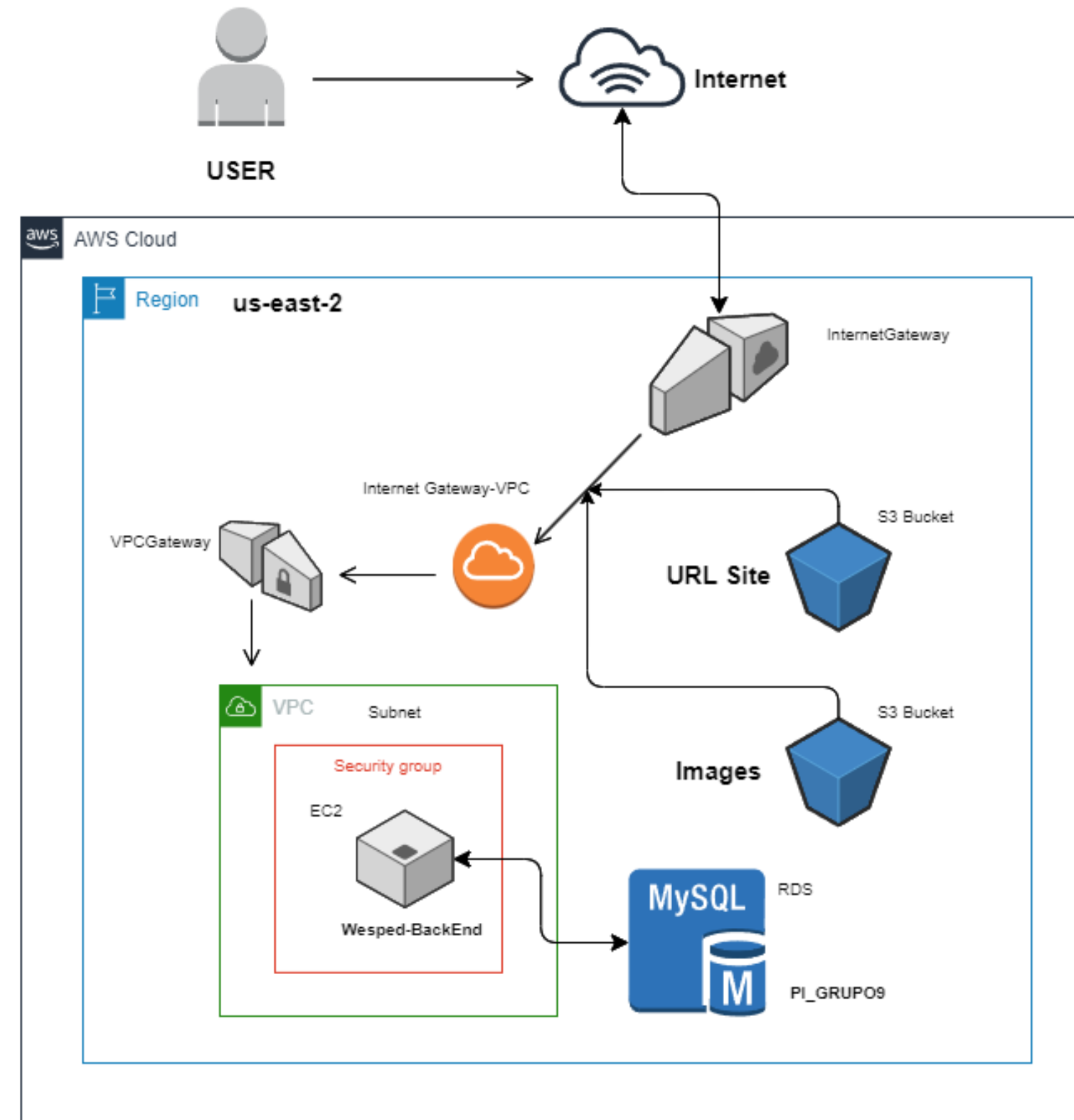
Nos planteamos un objetivo

Queremos acercar a nuestros clientes a los lugares dónde serán recibidos como en casa.

INFRAESTRUCTURA

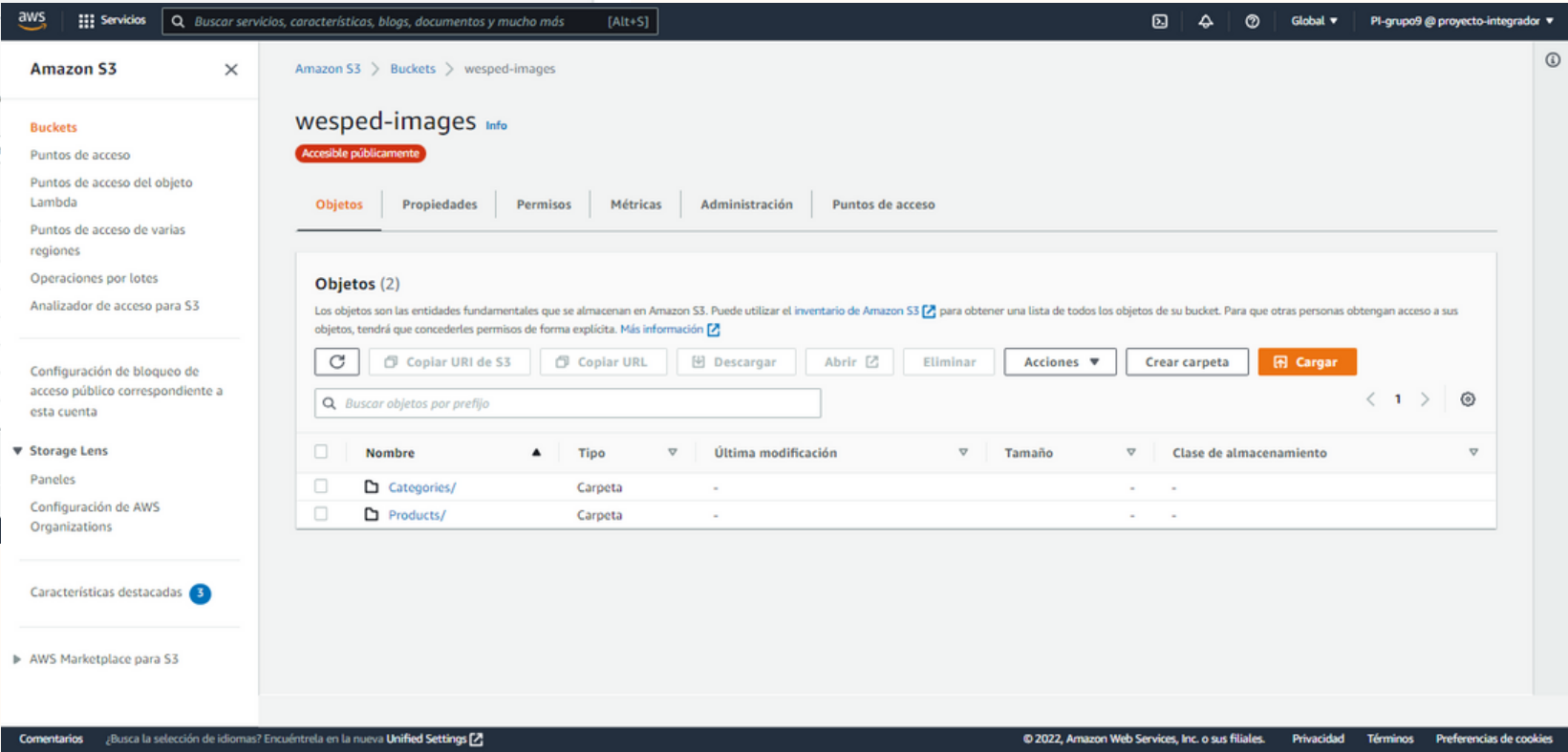
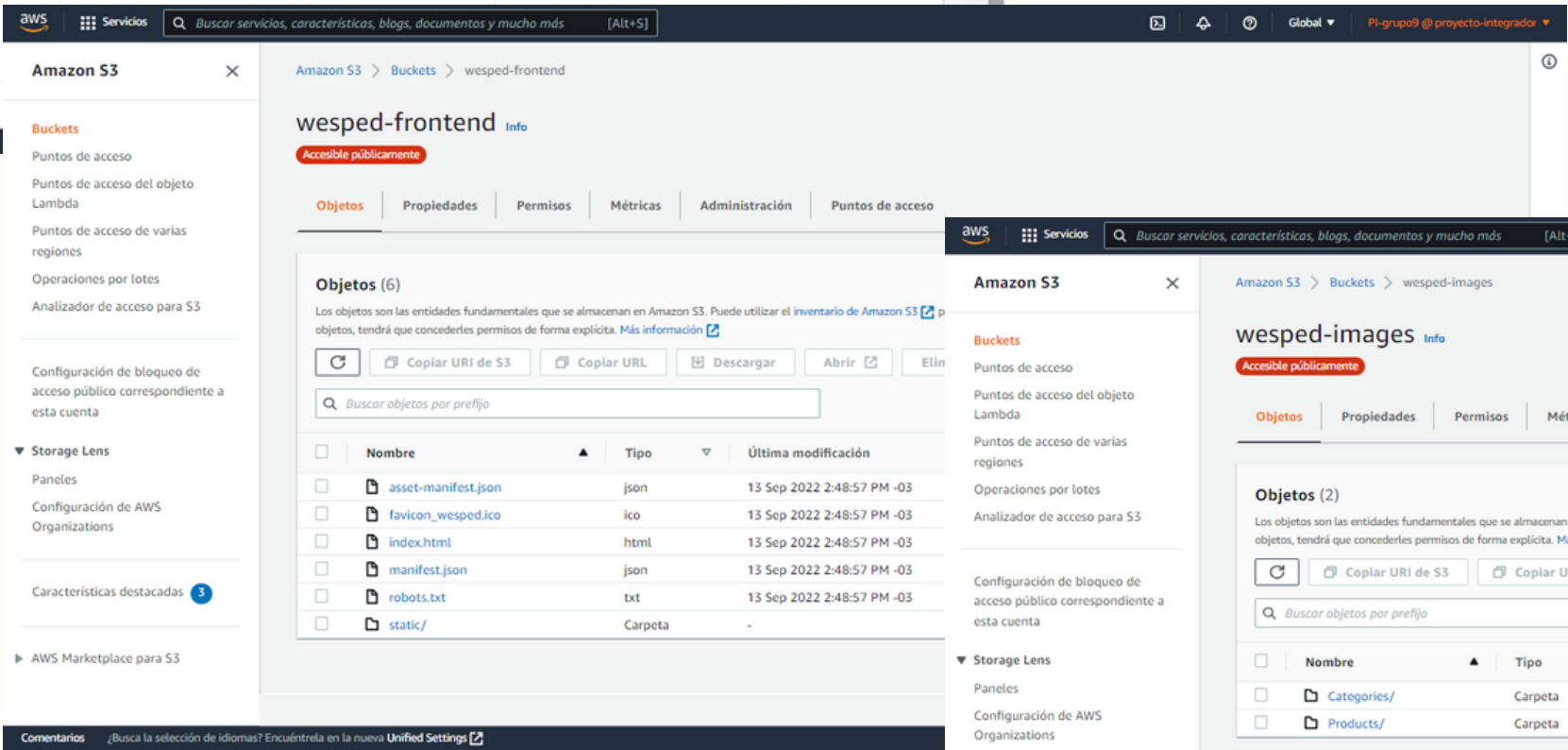
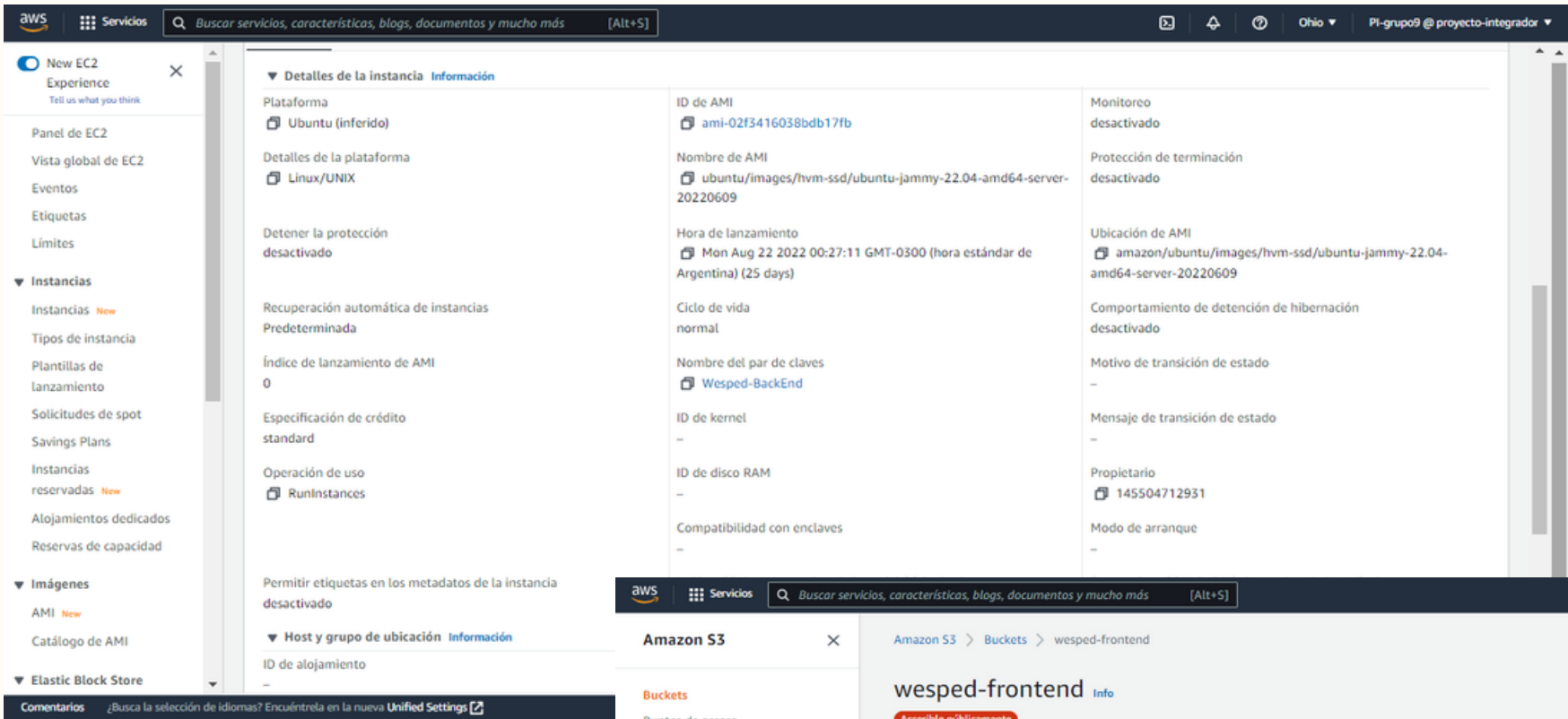


Diseño de Infraestructura

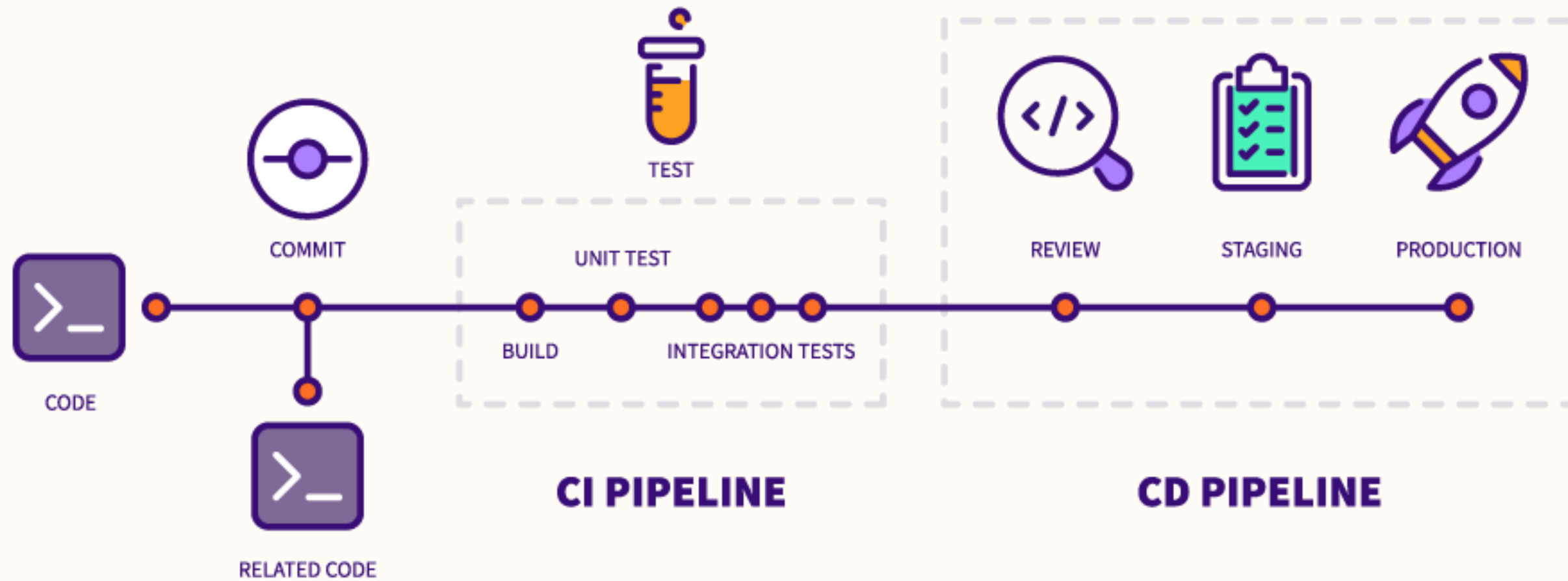


Levantamiento de infraestructura en AWS


Se crearon dos Bucket s3 para el front y las imágenes, y una instacia EC2 para el backend en AWS



Deploy en AWS



Build

✓ backend_build_job 

✓ frontend_build_job 

Deploy

✓ backend_deploy_job 

✓ frontend_deploy_job 

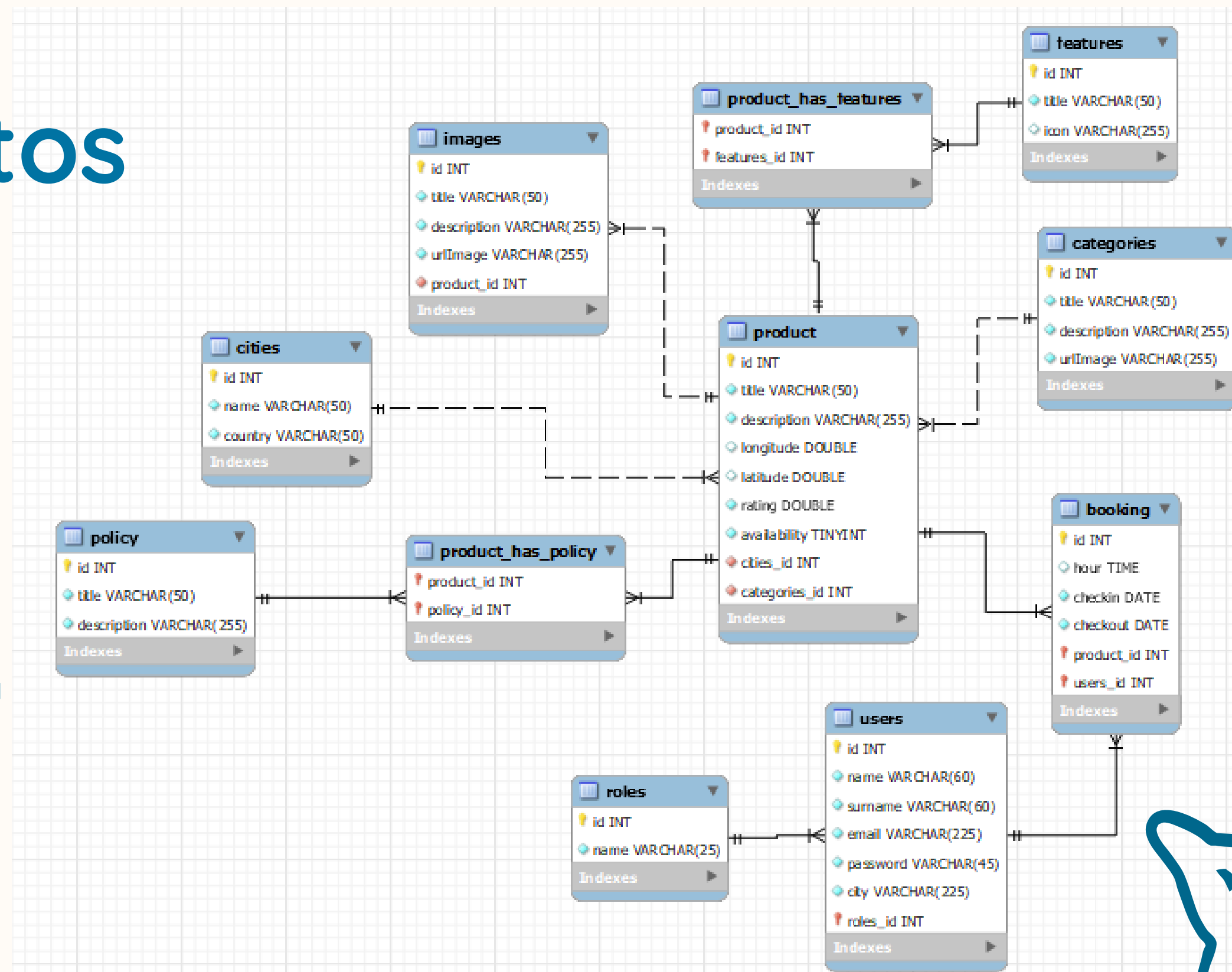
BASE DE DATOS



Nuestra Base de Datos

Se creó una BBDD relacional con SQL.

Su estructura está centrada en el producto o lugar de alojamiento con sus características y relaciona estos a los usuarios, gracias a las reservas que estos generen.

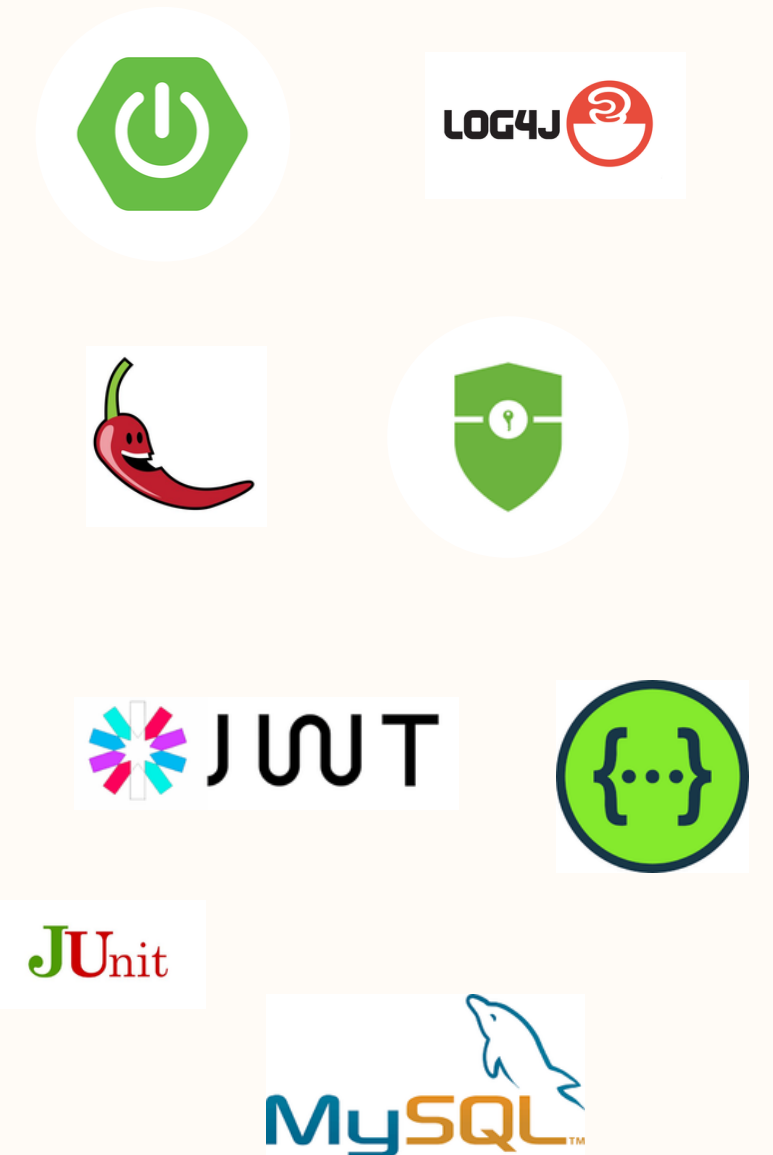
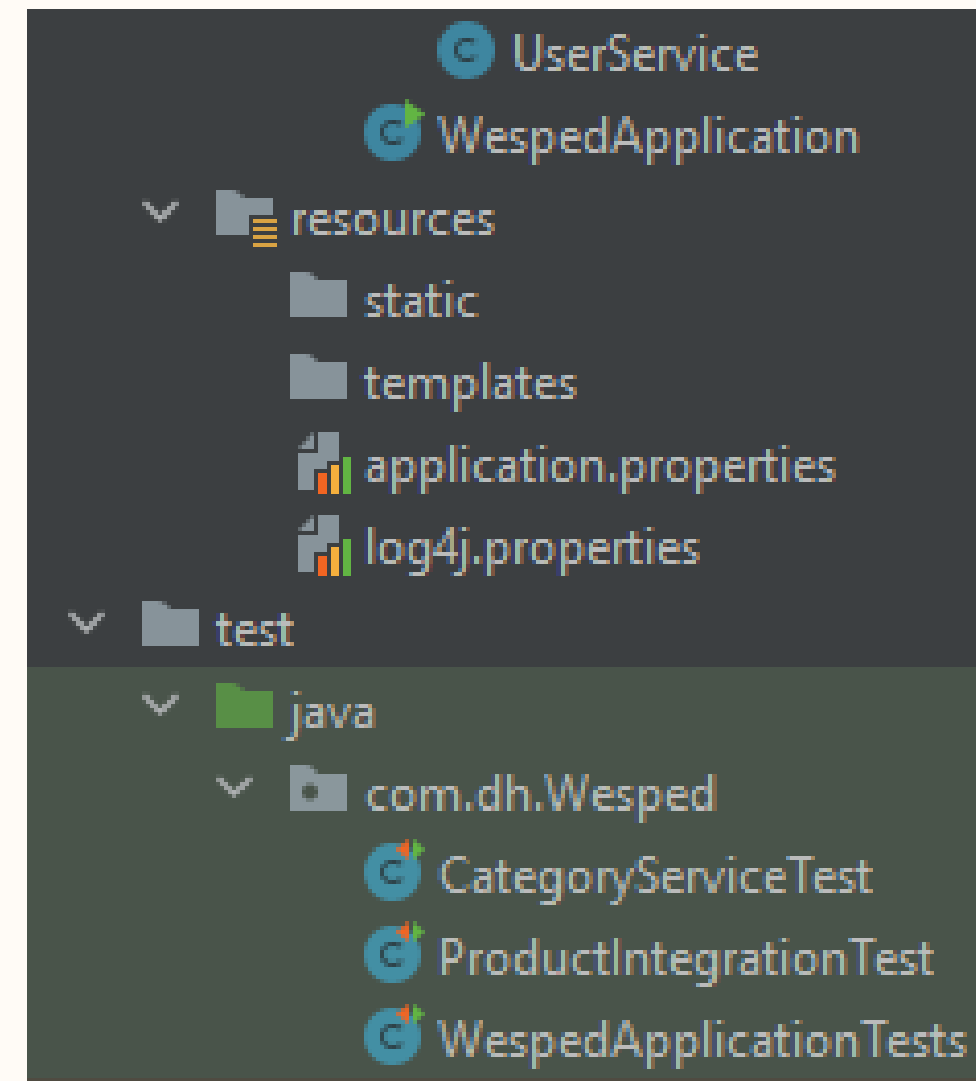
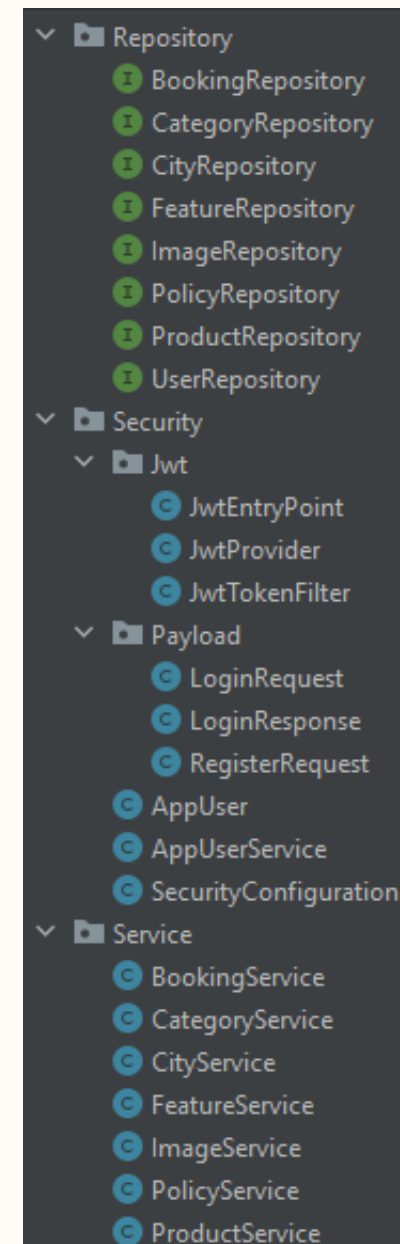
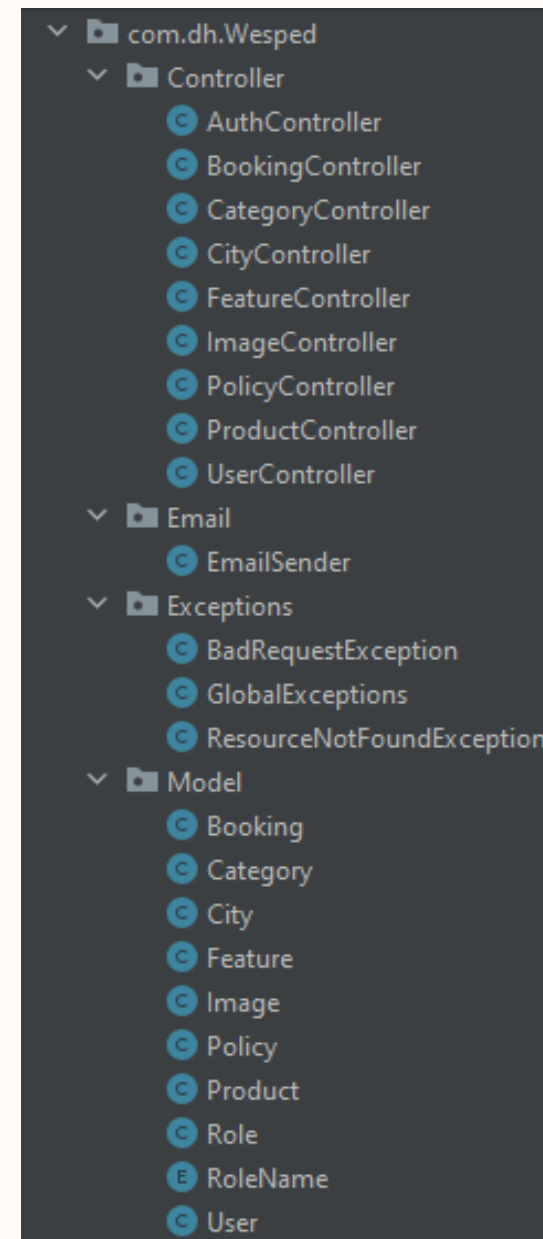


BACKEND



API

Herramientas utilizadas

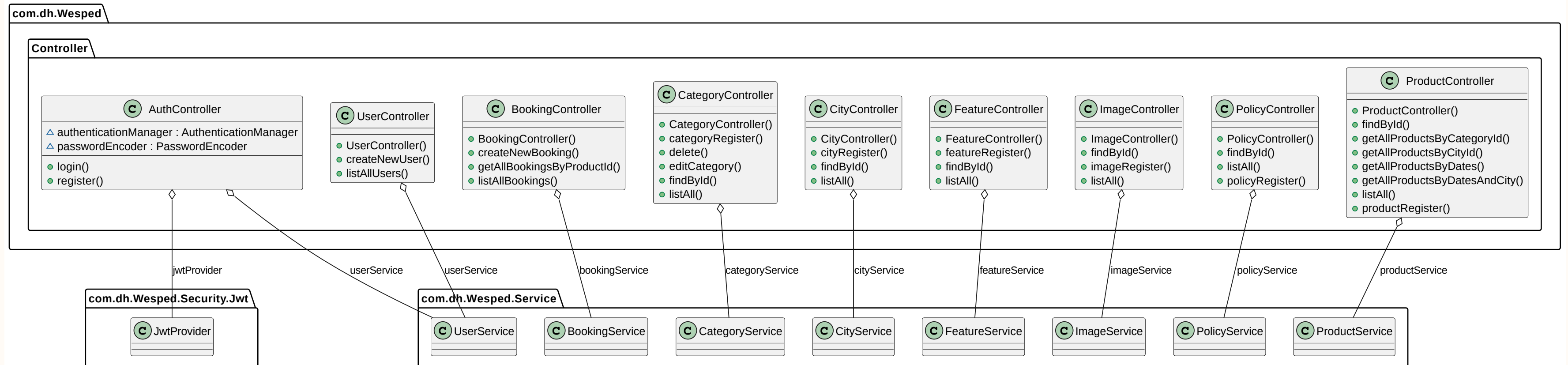


Se creó una API con java, y se implementó sprint boot, spring security, mySQL y JWT.



UML

CONTROLLER's Class Diagram



Documentamos nuestra API

Utilizamos Swagger para
documentar los endpoints
del proyecto.



Swagger Supported by SMARTBEAR /v3/api-docs Explore

Wesped API 1.0 OAS3
[/v3/api-docs](#)
Proyecto Integrador - Equipo 9 - Certified Tech Developer - Digital House [2022]

Servers

product-controller ^

- GET /products ✓
- PUT /products ✓
- POST /products ✓
- GET /products/{id} ✓
- GET /products/random ✓
- GET /products/city/{cityId} ✓
- GET /products/category/{categoryId} ✓
- GET /products/booking/{checkinDate}/{checkoutDate} ✓
- GET /products/booking/{checkinDate}/{checkoutDate}/{cityId} ✓

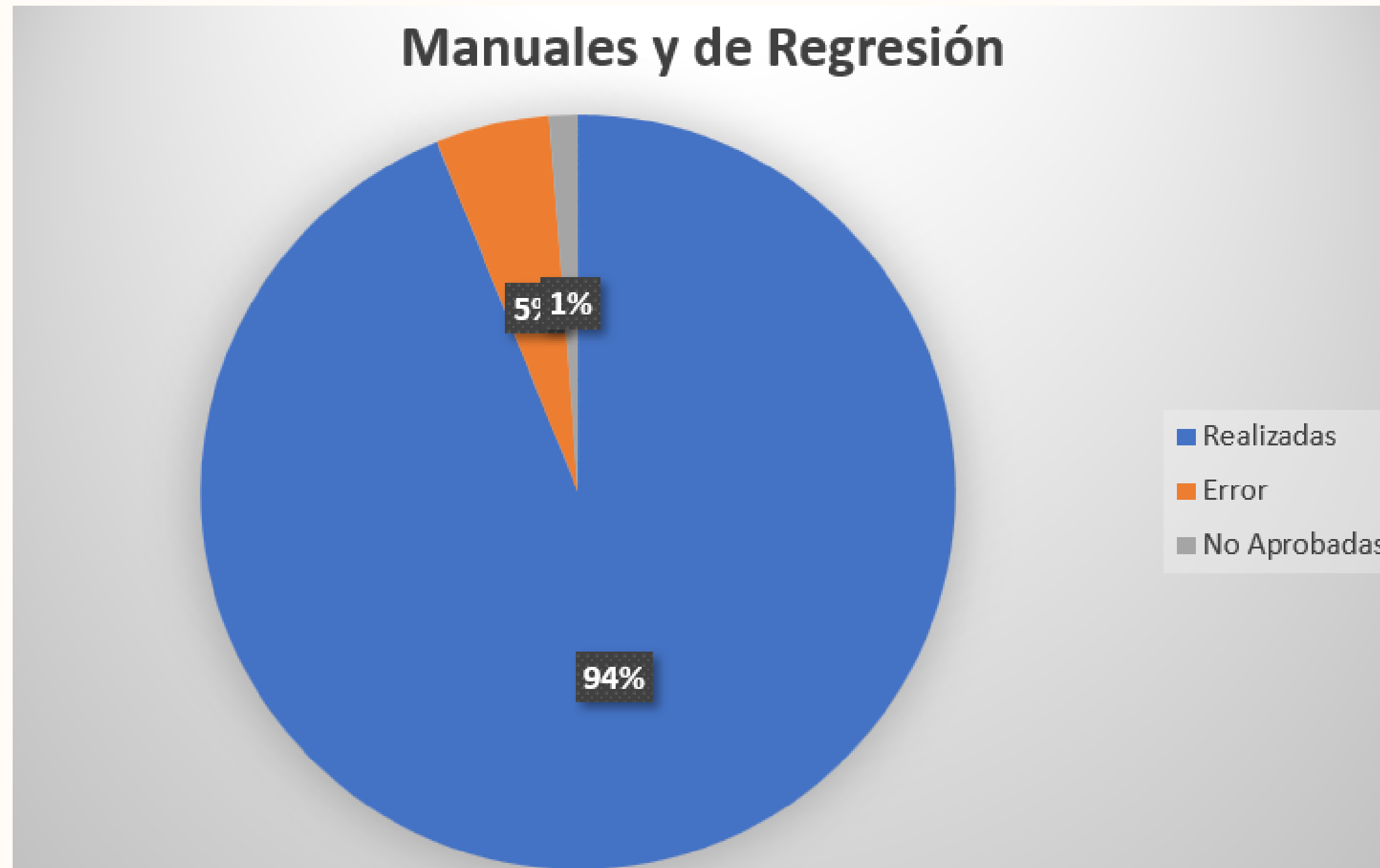
category-controller ^

- GET /categories ✓

TESTING



PRUEBAS MANUALES

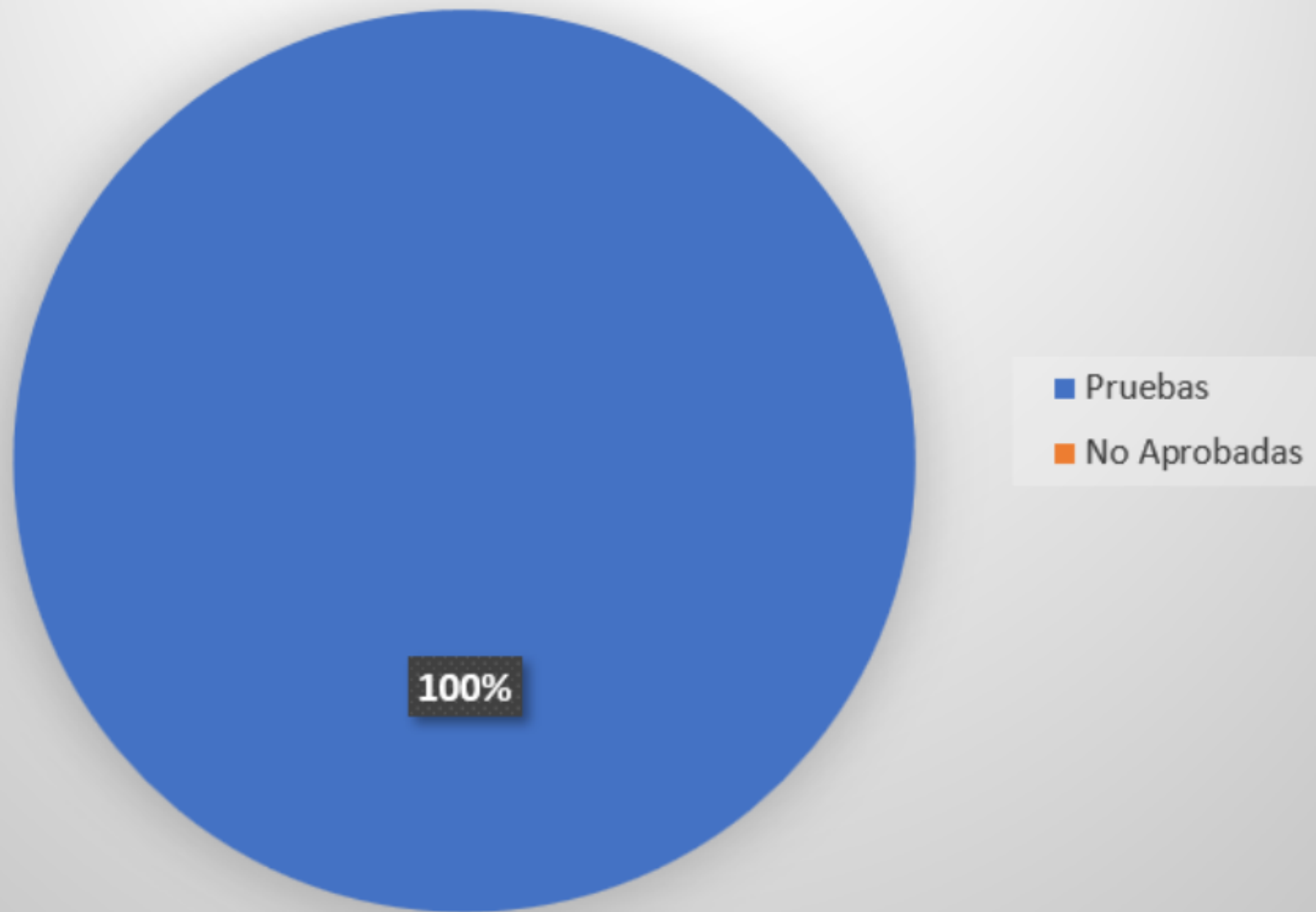


PRUEBAS EXPLORATORIAS

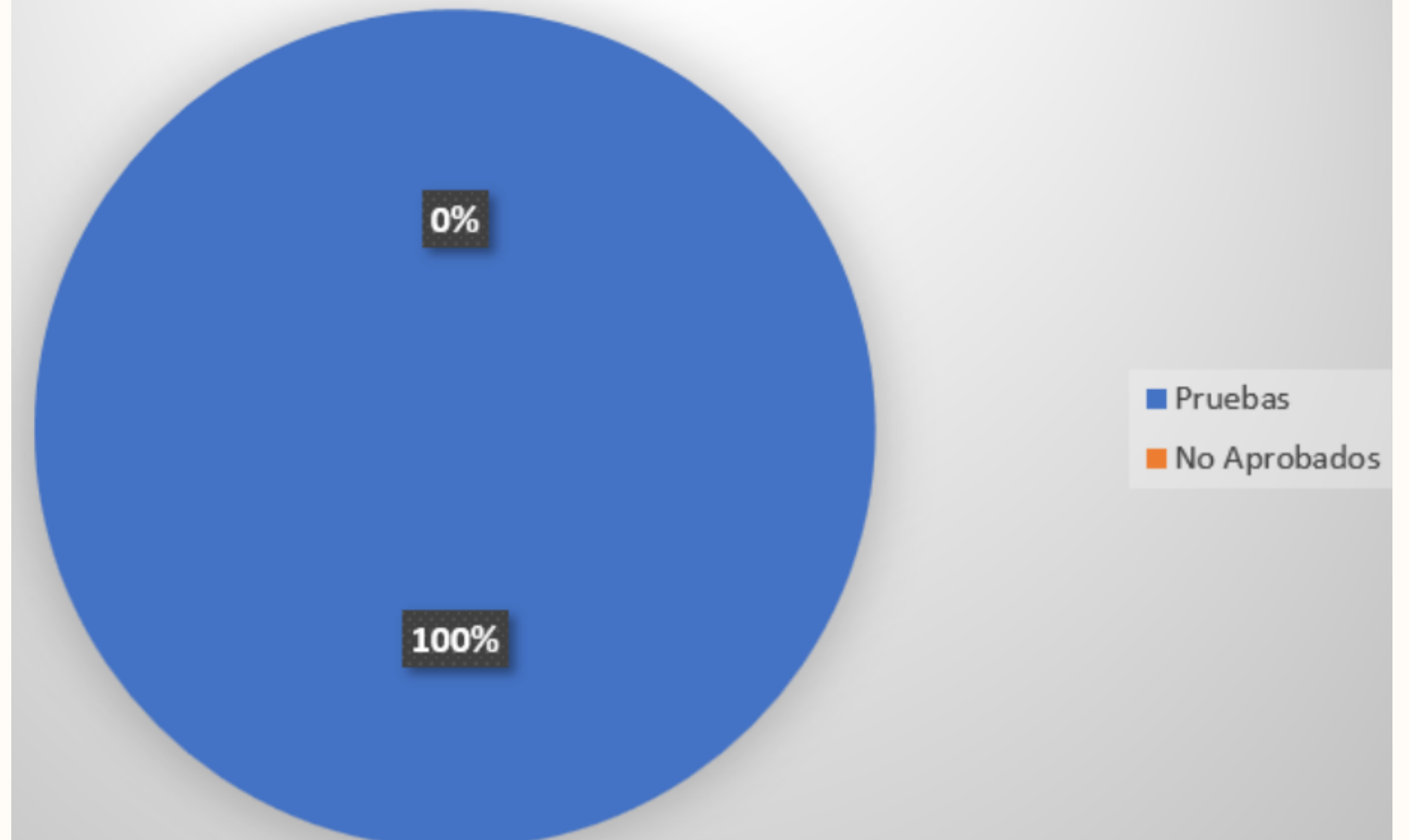


PRUEBAS AUTOMATIZADAS

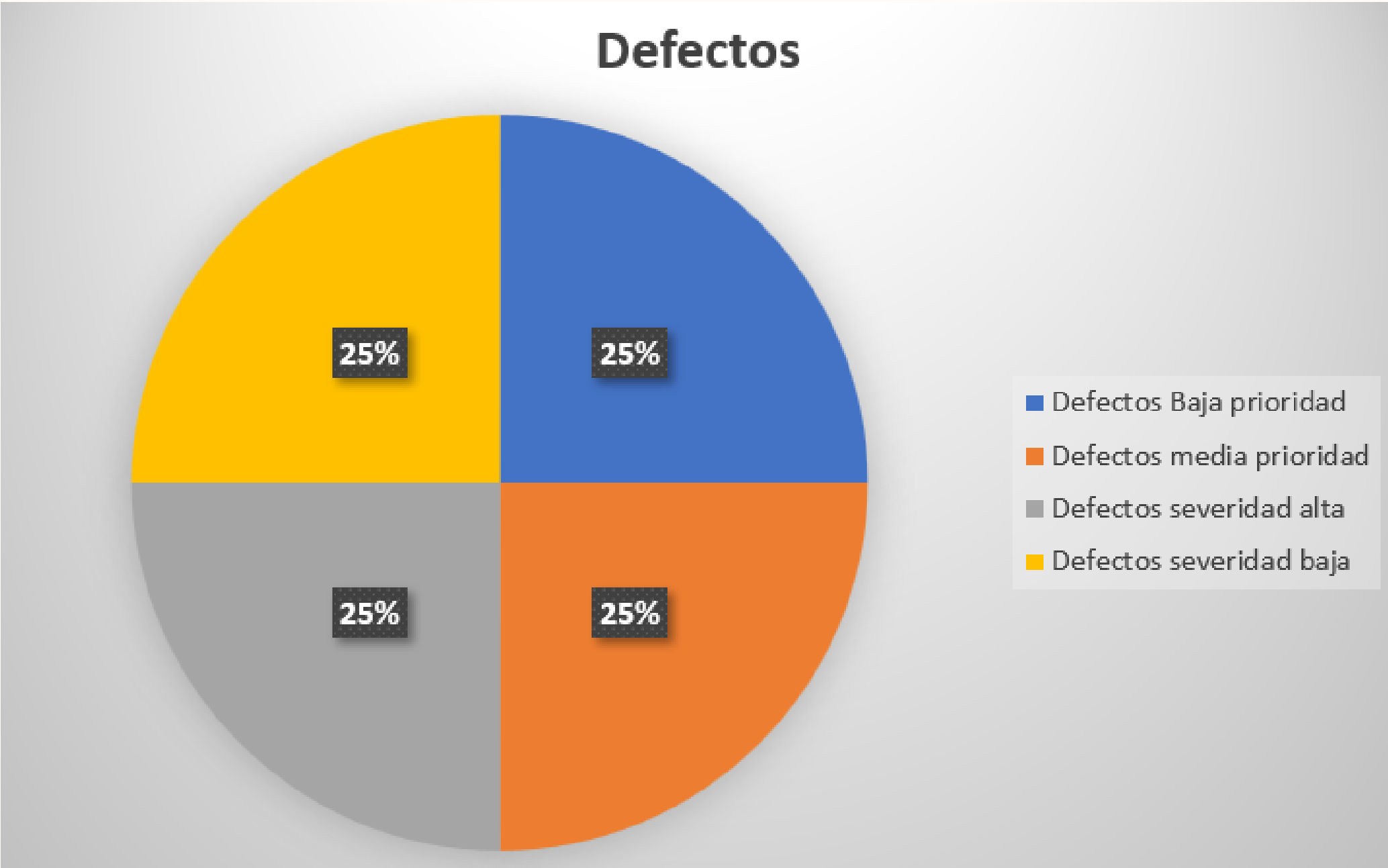
Pruebas Selenium



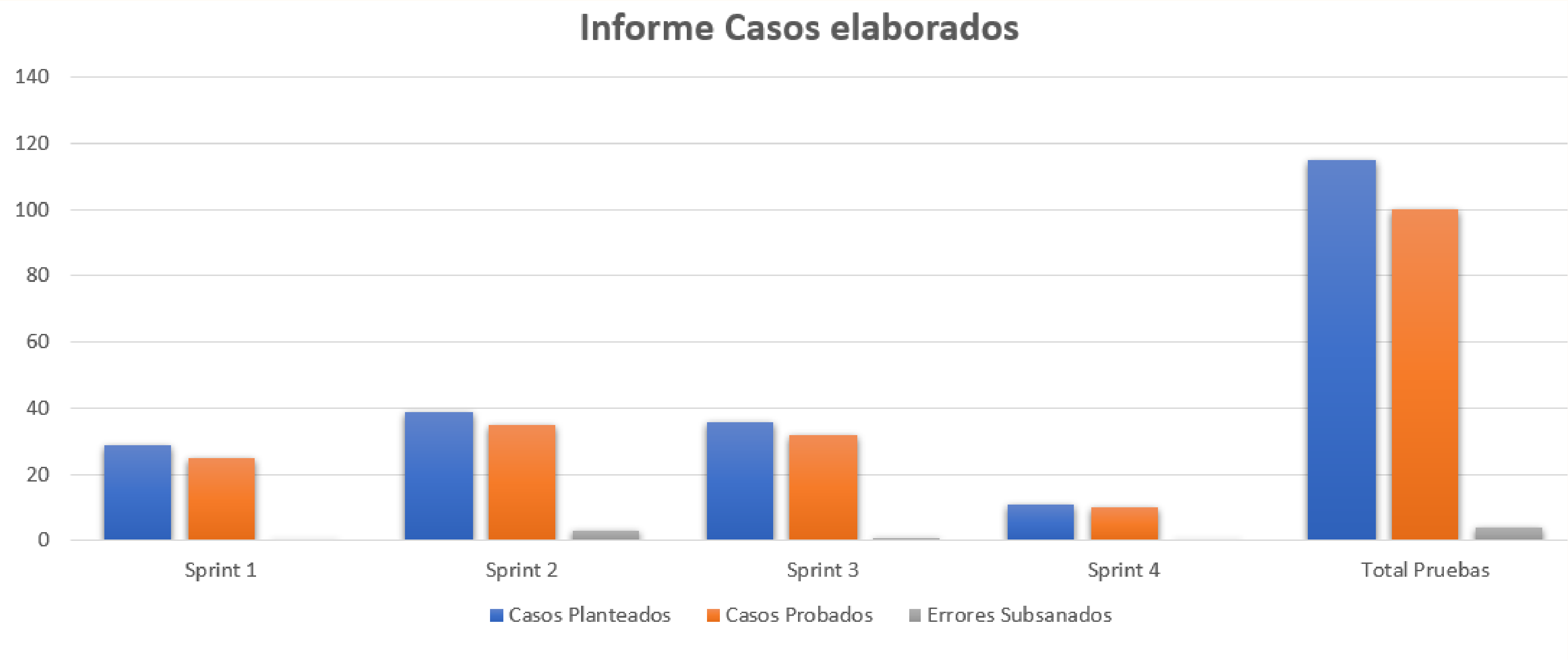
Postman



INFORME FINAL



INFORME FINAL

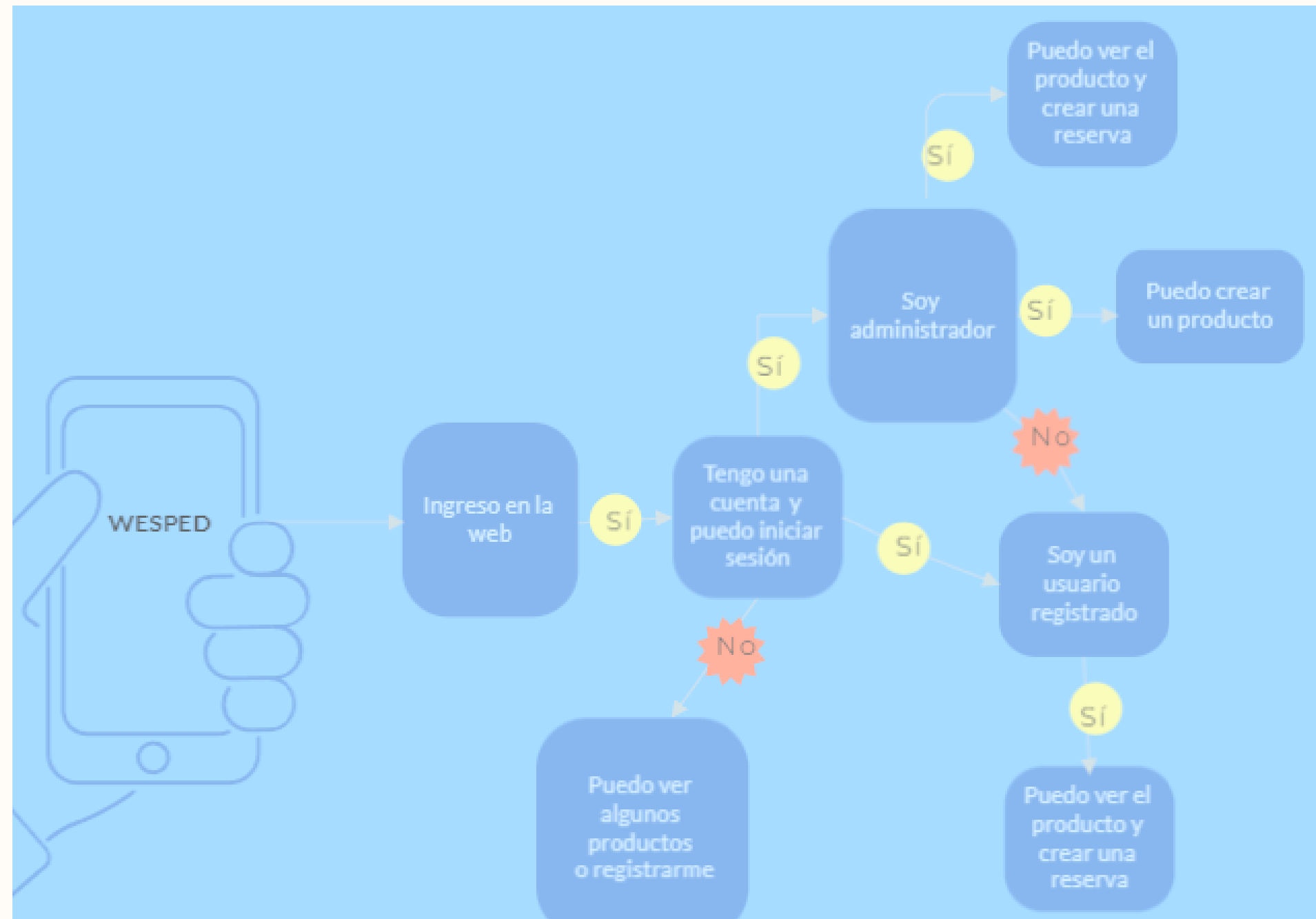


FRONTEND

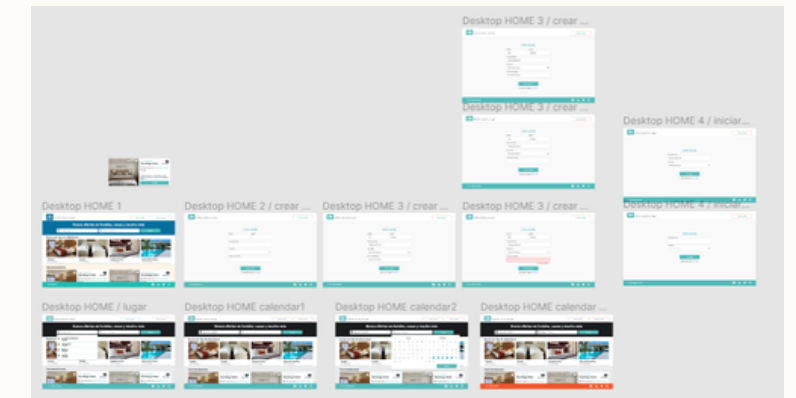
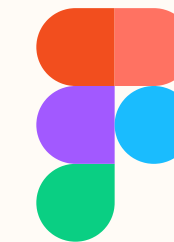


Creamos nuestro Front

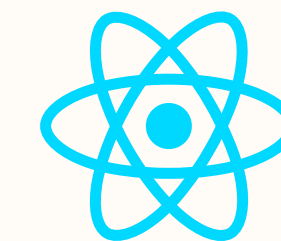
Diagrama de flujo de la app



Template del proyecto



Herramientas utilizadas



React





¡Gracias!



@WESPEDP

