

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

DESARROLLO DE PLATAFORMA WEB Y MÓVIL PARA EL PROCESO DE ADMINISTRACIÓN DE PRODUCTOS DE LA EMPRESA CONO SUPERIOR

TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN ANÁLISIS DE SISTEMAS INFORMÁTICOS

David Sebastian Curipoma Landeta

david.curipoma@epn.edu.ec

Cristian Eduardo Guamba Muñoz

cristian.guamba@epn.edu.ec

DIRECTOR: ING. JUAN PABLO ZALDUMBIDE PROAÑO, MSC.

juan.zaldumbide@epn.edu.ec

CODIRECTOR: PhD. MYRIAM GUADALUPE PEÑAFIEL AGUILAR

myriam.penafiel@epn.edu.ec

Quito, septiembre 2021

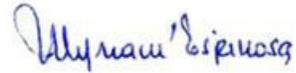
CERTIFICACIÓN

Certificamos que el presente trabajo fue desarrollado por el Sr. Guamba Muñoz Cristian Eduardo y el Sr. Curipoma Landeta David Sebastian como requerimiento parcial a la obtención del título de TECNOLOGO EN ANÁLISIS DE SISTEMAS INFORMÁTICOS, bajo nuestra supervisión:



Juan Pablo Zaldumbide

DIRECTOR(A) DEL
PROYECTO



Myriam Guadalupe Peñafiel

CODIRECTOR(A) DEL PROYECTO

DECLARACIÓN

Nosotros Curipoma Landeta David Sebastian con CI: 1718165523 y Guamba Muñoz Cristian Eduardo con CI: 1750024067 declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y que hemos consultado las referencias bibliográficas que se incluyen en este documento.

Sin perjuicio de los derechos reconocidos en el primer párrafo del artículo 144 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación – COESC-, somos titulares de la obra en mención y otorgamos una licencia gratuita, intransferible y no exclusiva de uso con fines académicos a la Escuela Politécnica Nacional.

Entregamos toda la información técnica pertinente, en caso de que hubiese una explotación comercial de la obra por parte de la EPN, se negociará los porcentajes de los beneficios conforme lo establece la normativa nacional vigente.

ÍNDICE DE CONTENIDOS

1. INTRODUCCIÓN.....	1
1.1. Objetivo General.....	2
1.2. Objetivos específicos.....	2
1.3. Alcance.....	3
2. METODOLOGÍA.....	4
2.1. Metodología de Desarrollo.....	4
2.1.1. Roles.....	4
2.1.2. Artefactos.....	5
2.2. Diseño de interfaces (mockups).....	6
2.2.1. Herramienta usada para el diseño.....	6
2.3. Diseño de la arquitectura.....	7
2.3.1. Patrón arquitectónico.....	8
2.3.2. Sistema web.....	8
2.3.3. Aplicación Móvil.....	9
2.4. Herramientas de desarrollo.....	10
2.4.1. API.....	10
2.4.2. Sistema Web.....	10
2.4.3. Aplicación Móvil.....	10
3. RESULTADOS Y DISCUSIÓN.....	12
3.1. Sprint 0. Acondicionamiento del ambiente de trabajo.....	12
3.2. Sprint 1.....	12
3.2.1. Creación del esquema de base de datos.....	12
3.2.2. Esquemmatización de las interfaces de usuario.....	14
3.2.3. Instanciar base de datos en un servidor en la nube.....	14
3.3. Sprint 2 (API).....	14

3.3.1.	Crear las migraciones para las entidades	15
3.3.2.	Crear <i>seeders</i> para todas las entidades	15
3.3.3.	Crear los controladores	15
3.3.4.	Implementar JWT para la autenticación de usuarios.....	15
3.3.5.	Establecer las relaciones entre las entidades	15
3.3.6.	Implementar la subida de imágenes al servidor	16
3.3.7.	Generar rutas	16
3.4.	Sprint 3 (Cliente Web)	16
3.4.1.	Implementar el módulo de Inicio de sesión	16
3.4.2.	Implementar el módulo de clientes	17
3.4.3.	Implementar el módulo de empleados	17
3.4.4.	Implementar el módulo de productos.....	17
3.4.5.	Implementar el módulo de producción	17
3.4.6.	Implementar el módulo de pedidos	17
3.5.	Sprint 4 (Cliente Móvil)	22
3.5.1.	Implementar el módulo de Inicio de sesión.....	22
3.5.2.	Visualizar el listado de clientes.....	23
3.5.3.	Visualizar el listado de empleados y su producción.....	23
3.5.4.	Implementar el módulo de productos.....	23
3.5.5.	Implementar el módulo de pedidos	23
3.6.	Sprint 5 (Publicación y pruebas)	28
3.6.1.	Desplegar la API y los clientes web y móvil	28
3.6.2.	Realizar pruebas de aceptación	30
3.6.3.	Realizar pruebas de compatibilidad en cliente web.....	31
3.6.4.	Realizar pruebas de compatibilidad en cliente móvil.....	31
3.6.5.	Realizar pruebas de carga y estrés en API Rest.....	32
4.	CONCLUSIONES Y RECOMENDACIONES	34

4.1.	Conclusiones	34
4.2.	Recomendaciones	34
5.	REFERENCIAS BIBLIOGRÁFICAS	35
6.	ANEXOS.....	36
6.1.	Manual Técnico	36
6.2.	Manual de Usuario.....	36
6.3.	Manual de Instalación.....	36

ÍNDICE DE FIGURAS

Fig. 1: Inicio de sesión para el cliente web	6
Fig. 2: Inicio de sesión para el cliente móvil	7
Fig. 3: Arquitectura API Rest.....	8
Fig. 4: Arquitectura cliente web	9
Fig. 5: Arquitectura cliente móvil	9
Fig. 6: Esquema de base de datos	13
Fig. 7: Recurso de base de datos MySQL en Heroku	14
Fig. 8: Cliente web - Página de inicio para usuario no autenticado.....	18
Fig. 9: Cliente web - Página de inicio para usuario administrador.....	19
Fig. 10: Cliente web - Página de inicio para usuario cliente	19
Fig. 11: Cliente web - Página de inicio de sesión cliente web.....	20
Fig. 12: Cliente web - Listado de pedidos y filtro por estados	20
Fig. 13: Cliente web - Detalle de una orden	21
Fig. 14: Cliente web - Listado de usuarios.....	21
Fig. 15: Cliente web - Registro de usuarios.....	22
Fig. 16: Cliente móvil - Página de inicio de sesión.....	24
Fig. 17: Cliente móvil - Menú administrador	24
Fig. 18: Cliente móvil - Menú cliente.....	25
Fig. 19: Cliente móvil – Página de lista de clientes	25
Fig. 20: Cliente móvil – Página lista de empleados.....	26
Fig. 21: Cliente móvil – Producción empleado.....	26
Fig. 22: Cliente móvil - Catálogo de productos	27
Fig. 23: Cliente móvil - Pedidos administrador.....	27
Fig. 24: Cliente móvil - Pedidos cliente	28
Fig. 25: Despliegue API Rest.....	29
Fig. 26: Cliente web publicado mediante Heroku	30
Fig. 27: Ejemplo prueba de compatibilidad cliente web.....	31
Fig. 28: Catálogo de productos en Sony Xperia L2.....	32
Fig. 29: Resultados prueba de carga.....	32
Fig. 30: Resultados prueba de estrés.....	33

ÍNDICE DE TABLAS

TABLA I: Asignación de roles.....	4
TABLA II: Ejemplo de Historia de Usuario	5
TABLA III: Herramientas API.....	10
TABLA IV: Herramientas Cliente Web	10
TABLA V: Herramientas Cliente Móvil	10
TABLA VI: Ejemplo de prueba de aceptación	30

RESUMEN

Este proyecto corresponde al desarrollo de un cliente móvil y un cliente web que permiten, principalmente, la generación y administración de pedidos de una empresa de distribución de conos y similares para heladería o repostería, administración de producción de empleados y la administración de productos. Ambos clientes interactúan con el origen de datos mediante un servicio web de tipo *API Rest*.

Para el desarrollo de este proyecto se ha utilizado la metodología *Scrum*, pues permite una mejor organización, especialmente en proyectos compartidos con más de un desarrollador. El enfoque por entregas iterativas permite un mejor seguimiento de las etapas de desarrollo, las cuales se detallan en la sección Resultados y Discusión de este documento.

En la primera sección se describe el problema y la solución que justifican el desarrollo de los dos sistemas para administración de pedidos, junto con el objetivo principal del proyecto, sus objetivos específicos y el alcance.

En la sección dos se detallan los artificios definidos por la metodología *Scrum*, junto con la arquitectura de la solución y del origen de datos a usar.

En la sección tres se desglosan las diferentes etapas del desarrollo de este proyecto, adicional a los resultados posteriores obtenidos de la etapa de pruebas.

Finalmente, en la sección cuatro se encuentran las conclusiones y recomendaciones encontradas durante el desarrollo del proyecto y su culminación.

PALABRAS CLAVE: *Scrum, Administración de pedidos, API REST*

ABSTRACT

The development of this project corresponds to the development of a mobile and a web client that mostly allows the generation and administration of orders for a company that distributes cones and similars, for ice cream or pastry shops, employee production management and the administration of products. Both clients interact with the data source through a rest API web service.

For the development of this project, Scrum methodology was chosen, as it allows a better organization, especially in projects shared with more than one developer. The iterative deliverables approach allows better monitoring of the development stages, which are detailed in the Results and Discussion section of this document.

The first section describes the problem and solution that justify the development of the two systems for orders management, along with the main objective of the project, its specific objectives and scope.

In section two the artifices defined by the Scrum methodology are detailed, along with the architecture of the solution and the data source to be used.

Section three breaks down the different stages of the development of this project, in addition to the subsequent results obtained from the testing stage.

Finally, section four contains the conclusions and recommendations found during the development of the project and its completion.

KEYWORDS: *Scrum, Order Management, REST API*

1. INTRODUCCIÓN

La automatización de procesos en la era actual es uno de los hitos que muchas empresas buscan, con el objetivo de reducir costos, aumentar productividad, reducir errores y tener un control de resultados en tiempo real.

Hoy en día en el mercado, son cada vez más las empresas que optan por automatizar procesos, con el fin de ser mucho más rentables y eficaces [1].

Por motivo de la pandemia ocasionada debido al brote de la COVID-19 [2], la digitalización de los procesos y procedimientos de una empresa se ha convertido en una necesidad para la evolución de un negocio. Muchos de estos procesos se pueden digitalizar para optimizarlos, esto por medio de herramientas digitales y recursos tecnológicos que ayudan a mejorar la capacidad operativa de la empresa [3].

Según la Comisión Económica para América Latina y el Caribe (CEPAL) [4], “el comercio sin soporte de papel”, es considerado como un potenciador considerable para la incorporación de las pequeñas y medianas empresas (PYMES) en el comercio electrónico y la digitalización de sus procesos. Esto debido a la eficacia, el mejoramiento de la productividad y la competitividad que la actualización tecnológica brinda a los negocios, lo que se traduce en ahorros en términos de costos y tiempo. Por ello los procesos actuales con los cuales se realiza la comunicación con los clientes y la administración de productos en la empresa Cono Superior, pueden ser automatizados, por lo que se propone el desarrollo de una aplicación web y otra móvil, para ayudar a la empresa a cumplir estas tareas.

Actualmente la empresa “Cono Superior”, la cual ofrece productos de galleta tales como conos, tulipanes, barquillos, entre otros, para distintas heladerías en la ciudad de Quito, maneja sus pedidos, productos y clientes, de una manera tradicional. La administración de pedidos y de los clientes, se lo hace a través de llamadas telefónicas, mensajes de texto o mensajes por medio de herramientas de comunicación como lo es “WhatsApp”. Estas formas de comunicación se han venido utilizando desde hace años atrás cuando la cantidad de clientes o heladerías, a los cuales se realizaba la recepción de dichos pedidos, no era muy amplia.

En una conversación que se ha realizado con la gerente general de la empresa [5], se nos ha informado que la administración de solicitudes de pedidos como se lo ha venido llevando

hasta la actualidad, no es muy optima por diferentes motivos, por lo cual en múltiples ocasiones la solicitud de un pedido nunca se pudo establecer.

De igual manera, la forma de administrar los pedidos que ofrece la empresa, se la realiza registrando esta información en un cuaderno. Esta no es una actividad que contribuya con la productividad y genera dificultades, ya que mantener ordenado un listado de ordenes en una empresa o negocio, es uno de los principios fundamentales para tener un control y gestión eficiente de los procesos de almacenaje y entrega de productos [6]. Adicionalmente, la empresa ofrece una amplia variedad de productos, por lo cual la forma de administrar los mismos es más compleja.

Con los avances tecnológicos actuales, surgen nuevas herramientas que facilitan la vida y ahorran mucho tiempo del que normalmente se invierte en desarrollar alguna actividad o tarea específica [7]. Hoy en día las plataformas web son una solución que adoptan muchas empresas, ya que posibilita la ejecución de diversas tareas en un mismo lugar a través de internet. Por ello, se pretende desarrollar una plataforma web y móvil para el proceso de administración de pedidos de la empresa “Cono Superior”, optimizando la comunicación con el cliente y la administración de pedidos por parte de la empresa.

Las plataformas web y móvil están dirigidas a todo el público que quieran conocer acerca de la empresa y lo que esta ofrece, pero en especial, se encuentran dirigidas a sus clientes tales como heladerías o negocios independientes, que llevan años trabajando con la misma, brindando una nueva forma de comunicación y administración de pedidos, por parte de la empresa hacia el cliente.

1.1. Objetivo General

Desarrollar la plataforma web y aplicación móvil para el proceso de administración de productos de la empresa “Cono Superior”.

1.2. Objetivos específicos

- Levantar los requerimientos correspondientes de acuerdo con las necesidades de la empresa.
- Diseñar la arquitectura de la solución basada en los requerimientos.
- Implementar la plataforma web y móvil en función del diseño.
- Realizar las pruebas en la plataforma web y móvil.

1.3. Alcance

El alcance de este proyecto integrador contempla el desarrollo de un cliente web y otro móvil, orientados a la administración de pedidos y control de producción de empleados, brindar comunicación directa entre el cliente y la empresa, y dar información sobre el negocio.

La aplicación web muestra información de interés sobre la empresa, permite que el usuario inicie sesión, que se registren productos, clientes, empleados, que los clientes seleccionen productos, los añadan a un pedido y reciban notificaciones del estado de su pedido. El administrador también recibe notificaciones por cada pedido nuevo.

El cliente móvil es la versión concentrada del cliente web, ya que se enfoca en la administración de productos y clientes de una forma más versátil, dividiendo al usuario en cliente y administrador; los cuales pueden iniciar sesión, al igual que realizar solicitudes de pedidos ver el estado de las mismas, además de visualizar el catálogo de productos, el listado de empleados y la producción de los mismos

2. METODOLOGÍA

Este proyecto se ha realizado mediante la metodología ágil *Scrum*, la cual se describe a continuación.

2.1. Metodología de Desarrollo

Scrum es un método ágil de trabajo y un conjunto de herramientas adaptable, iterativo, rápido, flexible y eficaz, diseñado para ofrecer un valor considerable en forma rápida a lo largo del proyecto. También garantiza transparencia en la comunicación y crea un ambiente de responsabilidad colectiva y de progreso continuo [8].

2.1.1. Roles

En *Scrum* se consideran tres roles centrales, los cuales intervienen obligatoriamente cuando se trabaja con *Scrum* para la creación de un proyecto [8].

Estos roles son:

Product Owner: Es la persona que establece el valor empresarial del proyecto mediante el planteamiento de los requisitos, representando las intenciones del cliente [8].

Scrum Master: Es quien se asegura de que la metodología *Scrum* se aplique correctamente y de que los miembros del equipo de trabajo no tengan impedimentos para trabajar en el proyecto. Es quien conoce más de la metodología [8].

Scrum Team: Se refiere al equipo de desarrollo que crea los entregables y debe interpretar correctamente los requisitos expresados por el *Product Owner* [8].

Asignación de roles

Los roles para la creación del proyecto se muestran a continuación en la **TABLA I**.

TABLA I: Asignación de roles

Roles	Miembros
<i>Product Owner</i>	Ing. Sandra Vásquez (Gerente General)
<i>Scrum Master</i>	Ing. Juan Pablo Zaldumbide. Msc
<i>Scrum Team</i>	David Curipoma, Cristian Guamba

2.1.2. Artefactos

Los artefactos en *Scrum* son elementos establecidos para garantizar transparencia en el proceso de desarrollo, permitiendo un mejor entendimiento de las actividades planificadas y evitando problemas de comunicación [9].

Recopilación de requerimientos

Para poder establecer las distintas fases del proceso de desarrollo, primero es necesario recopilar los requerimientos del *Product Owner* mediante la etapa de recopilación de requerimientos.

Historias de Usuario

Las Historias de Usuario son los requerimientos descritos desde la perspectiva del *Product Owner* que buscan aclarar la razón de ser de dicho requerimiento [10]. Un ejemplo de su representación puede ser el mostrado en la **TABLA II**.

TABLA II: Ejemplo de Historia de Usuario

HISTORIA DE USUARIO		
Identificador: HU01	Usuario: Cliente, Administrador	
Nombre de la historia: Iniciar sesión		
Prioridad en Negocio (Alta/Medio/Baja): Alta	Riesgo en Desarrollo (Alta/Medio/Baja): Medio	
Iteración asignada: 1		
Responsable(es): David Curipoma, Cristian Guamba		
Descripción: El usuario debe ser capaz de ingresar al sistema con sus credenciales otorgadas, el sistema concede los permisos de interacción y visualización de información dependiendo de su rol de usuario, que puede ser "cliente" o "administrador".		
Observación: Solo los usuarios previamente registrados en la base de datos tienen acceso al sistema y pueden visualizar o interactuar con la información de este, dependiendo de su rol.		

Product Backlog

Es una lista de todos los requerimientos junto con detalles que ayudan a establecer prioridades y cuyos elementos pueden cambiar de lugar durante el desarrollo. Estos detalles son el contenido, la disponibilidad y la orientación del requerimiento [9].

Sprint Backlog

Es un conjunto de elementos derivado del *Product Backlog*. En él se distinguen los *sprints* que posee el proyecto. Esta lista consta de predicciones de entrega del conjunto de tareas que se van a realizar durante cada *sprint* [9].

2.2. Diseño de interfaces (mockups)

En esta sección se muestran los *mockups* de los clientes web y móvil.

2.2.1. Herramienta usada para el diseño

Los mockups se han diseñado con la herramienta Figma, los resultados se muestran a continuación.

Cliente Web

El cliente web permite que los usuarios de tipo administrador y cliente inicien sesión.

En la **Fig. 1** se puede observar la interfaz diseñada para el inicio de sesión de los usuarios para el cliente web.

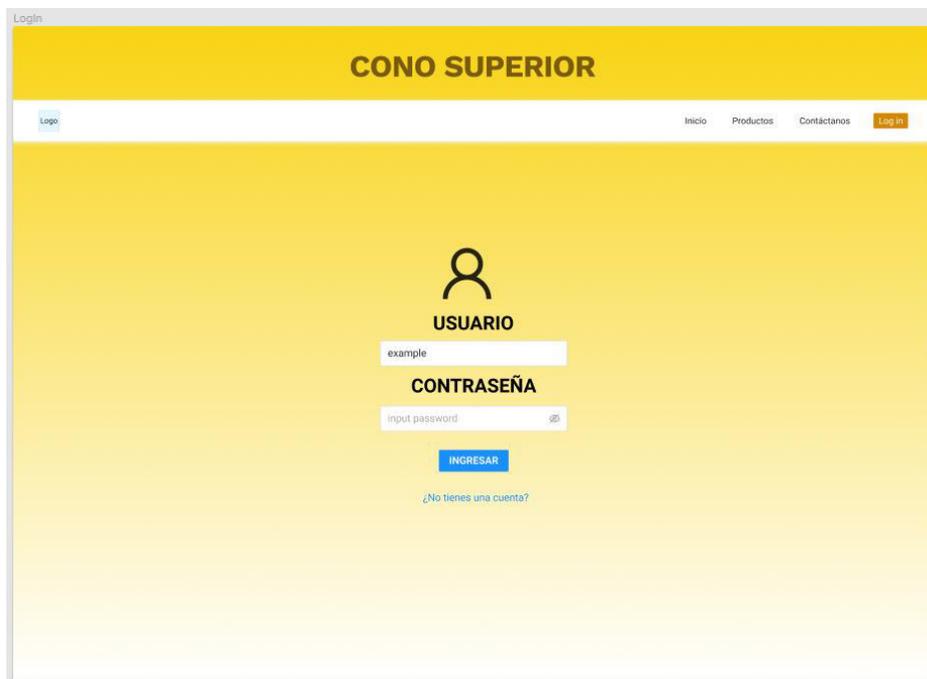


Fig. 1: Inicio de sesión para el cliente web

El diseño completo se puede encontrar en el manual técnico del proyecto.

Ciente móvil

De igual forma, el cliente móvil permite que los usuarios de tipo administrador y cliente inicien sesión.

En la **Fig. 2** se puede observar la interfaz diseñada para el inicio de sesión de los usuarios para el cliente móvil.



Fig. 2: Inicio de sesión para el cliente móvil

El diseño completo se puede encontrar en el manual técnico del proyecto.

2.3. Diseño de la arquitectura

En esta sección se muestran las arquitecturas elegidas para ser implementadas en el proyecto.

2.3.1. Patrón arquitectónico

El patrón utilizado es el de una arquitectura API Rest (ver

Fig. 3), debido a que es necesaria la disponibilidad de un servicio para que sea consumido por dos clientes, el web y el móvil, permitiendo finalmente la interacción con la base de datos [11].

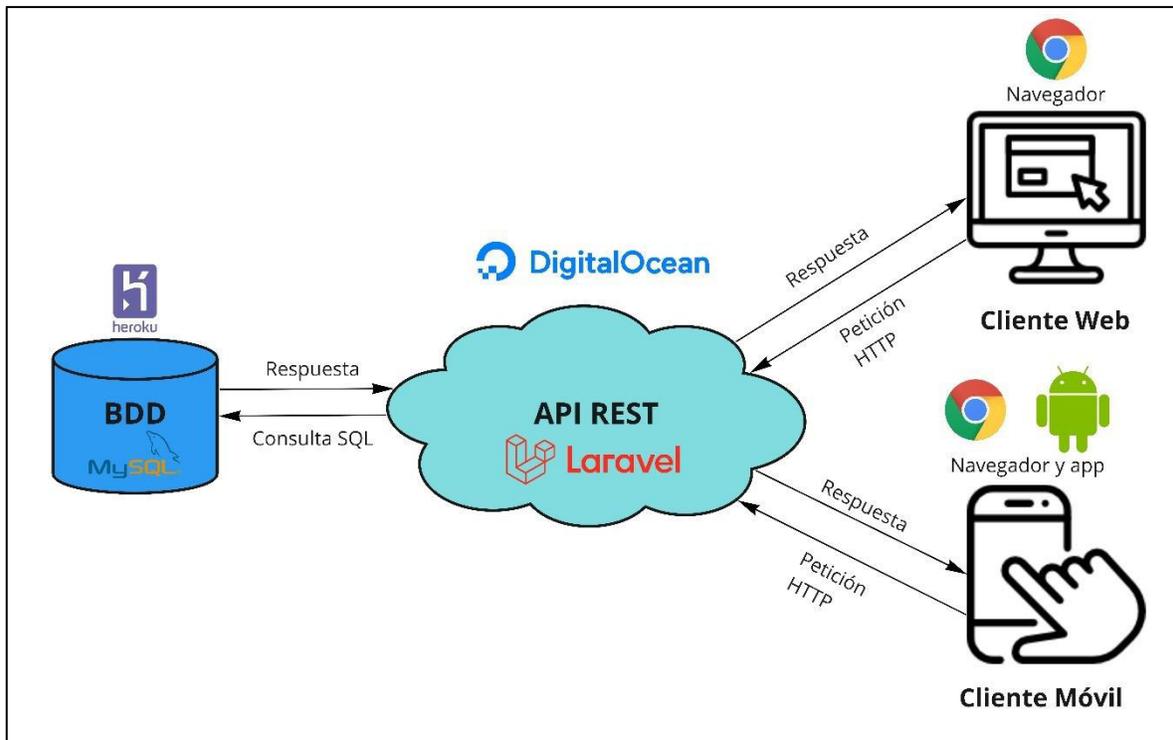


Fig. 3: Arquitectura API Rest

2.3.2. Sistema web

La arquitectura del cliente web es la que se presenta en la **Fig. 4** a continuación.

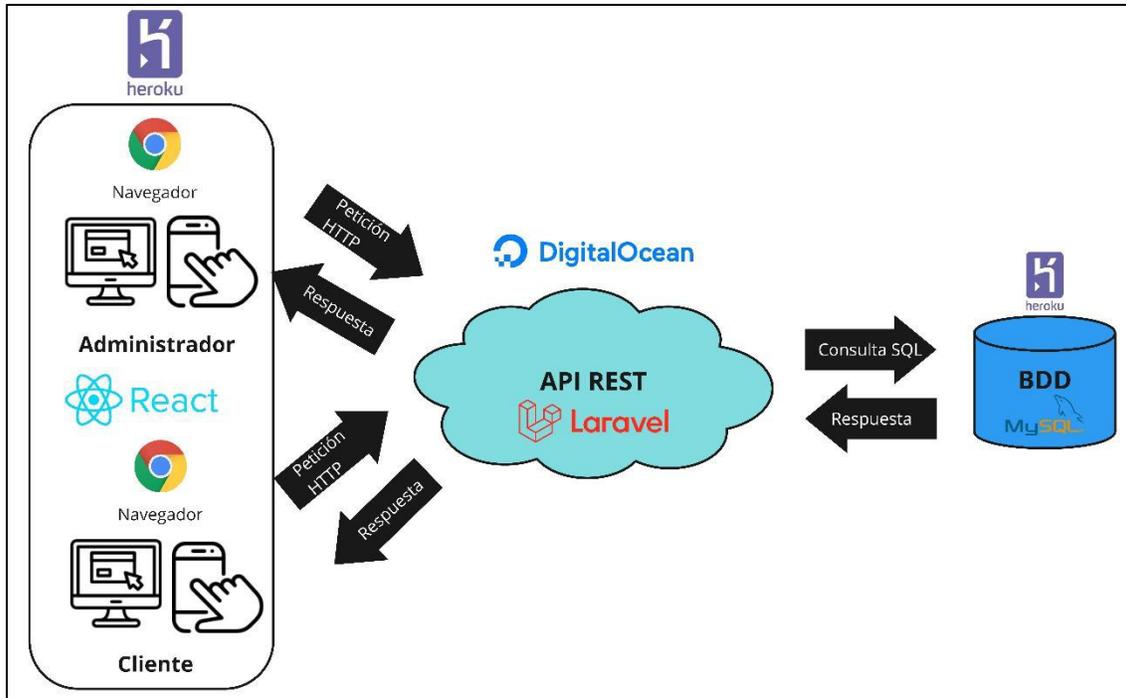


Fig. 4: Arquitectura cliente web

2.3.3. Aplicación Móvil

La arquitectura del cliente web es la que se presenta en la Fig. 5 a continuación.

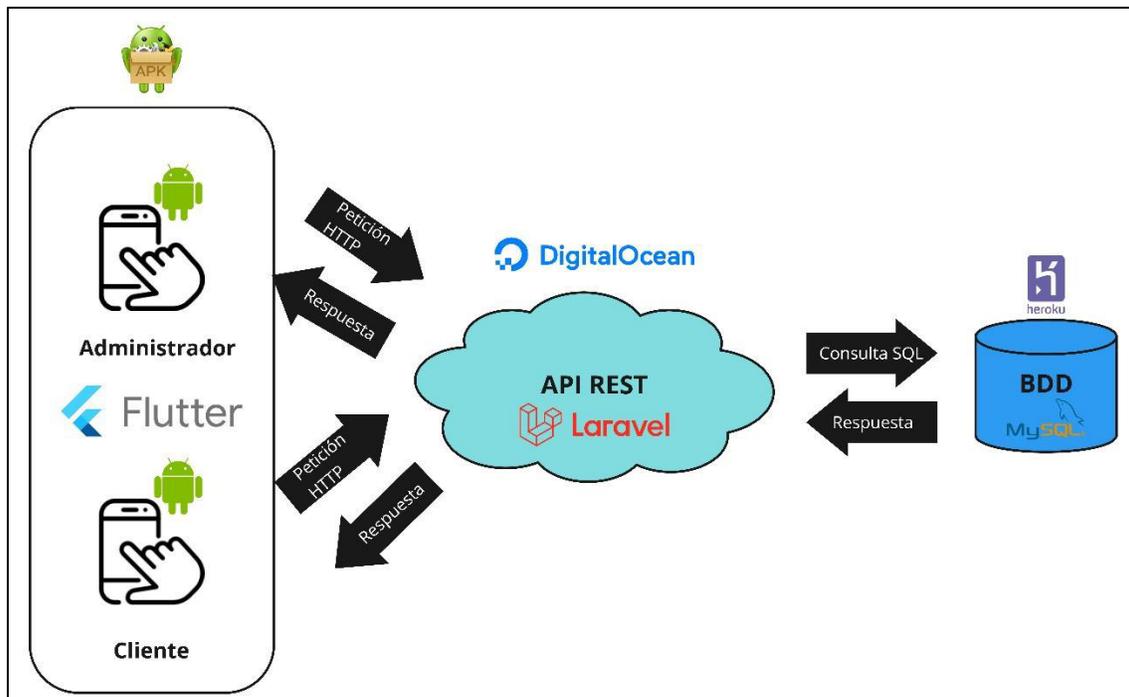


Fig. 5: Arquitectura cliente móvil

2.4. Herramientas de desarrollo

En este apartado se presentan las herramientas que se han utilizado para el desarrollo de este proyecto.

2.4.1. API

En la **TABLA III** se observan las herramientas usadas para el desarrollo de la API (*Application Programming Interface*).

TABLA III: Herramientas API

Herramienta	Justificación
Laravel	Al usar Laravel junto con Eloquent, es posible realizar una API para comunicar clientes con la base de datos, de manera fácil y rápida, ofreciendo una interacción simplificada entre las entidades que intervienen en el <i>backend</i> .

2.4.2. Sistema Web

En la **TABLA IV** se observan las herramientas usadas para desarrollar el Cliente Web.

TABLA IV: Herramientas Cliente Web

Herramienta	Justificación
React	Ya que el cliente web cuenta con varios módulos, los cuales a su vez poseen varios componentes visuales, se ha decidido utilizar React, pues esta herramienta permite la carga de componentes de una forma optimizada, permitiendo controlar que elementos deben ser procesados y mostrados, y cuáles no [12].

2.4.3. Aplicación Móvil

En la **TABLA V** se observan las herramientas usadas para desarrollar el Cliente Móvil.

TABLA V: Herramientas Cliente Móvil

Herramienta	Justificación
Flutter	Gracias a que este kit de herramientas es compatible tanto con Android como con IOS, simplifica el trabajo al momento de programar, ya que solo se crea una única base de código con la cual se elabora todo el cliente móvil para consumir recursos de

	nuestra API Rest; además es importante recalcar su versatilidad y facilidad de implementación [13].
--	---

3. RESULTADOS Y DISCUSIÓN

A continuación, se muestra el desarrollo correspondiente a los *Sprints* planificados para el desarrollo del proyecto integrador.

3.1. Sprint 0. Acondicionamiento del ambiente de trabajo

En el *Sprint 0* se planificaron todas las tareas para establecer un ambiente de desarrollo con todas las herramientas necesarias.

Dentro de este *Sprint* se establecieron las siguientes tareas:

- Reunión con el cliente.
- Recopilar los requerimientos funcionales y no funcionales.
- Realizar las historias de usuario.
- Preparar el ambiente de trabajo.

3.2. Sprint 1

En el *Sprint 1* se realizaron las actividades correspondientes a la creación del diagrama de base de datos y la esquematización de las interfaces de usuario.

Los objetivos principales de este *Sprint* se listan a continuación:

- Creación del esquema de base de datos.
- Esquematización de las interfaces de usuario.
- Instanciar base de datos en un servidor en la nube

Resultados:

3.2.1. Creación del esquema de base de datos.

Para la creación del esquema de base de datos se ha realizado un diagrama MER (Modelo entidad-relación) en la aplicación PowerDesigner con las entidades y relaciones necesarias.

El esquema resultante se muestra a continuación en la **Fig. 6**.

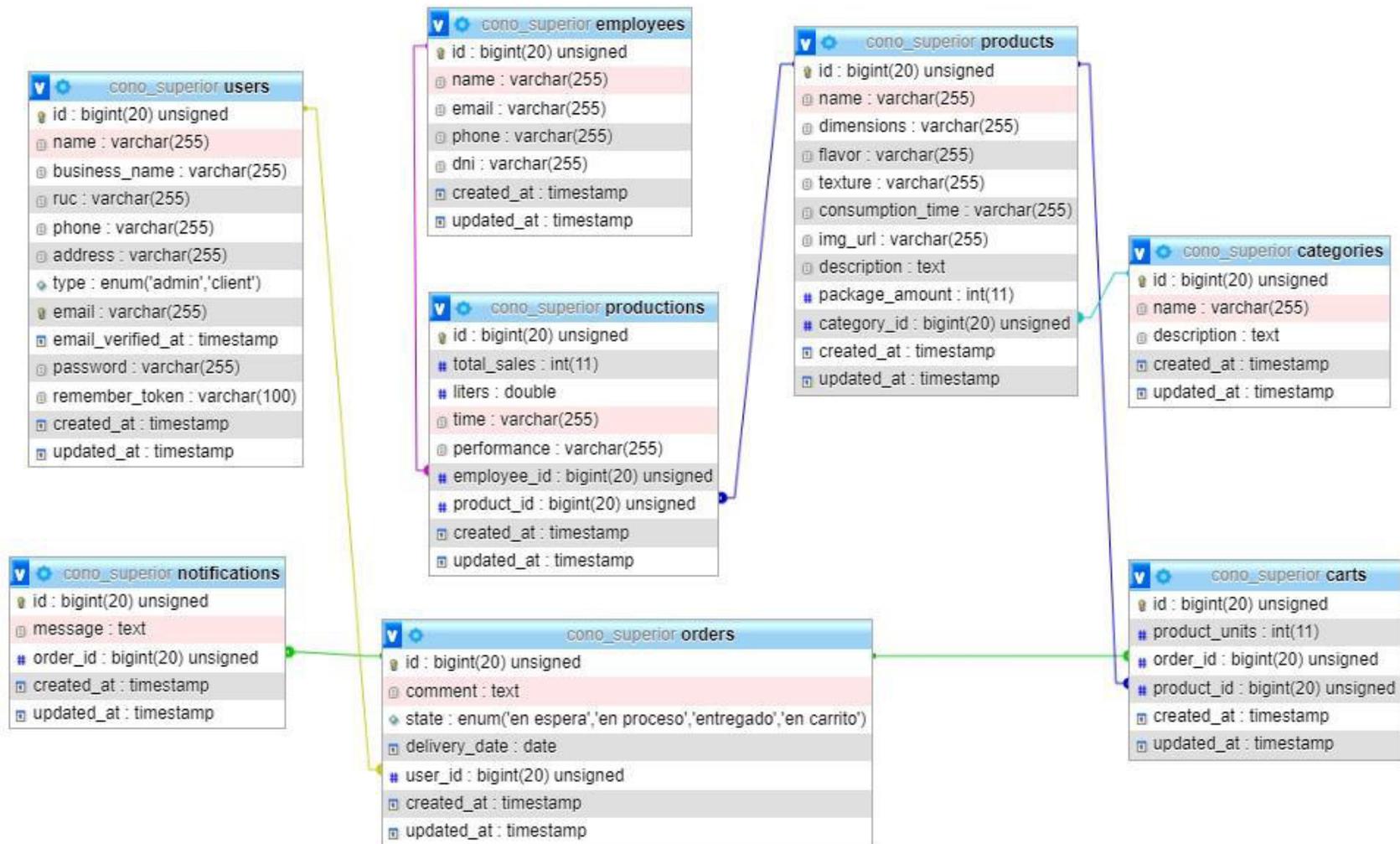


Fig. 6: Esquema de base de datos

3.2.2. Esquematzación de las interfaces de usuario.

Para esquematizar las interfaces de usuario de los clientes web y móvil, se ha utilizado la herramienta de diseño Figma.

El diseño final se puede observar en el manual técnico de este proyecto.

3.2.3. Instanciar base de datos en un servidor en la nube

Mediante la plataforma como servicio Heroku, se crea una instancia de una base de datos para el proyecto con un máximo de 3500 consultas por hora.

En la **Fig. 7** se puede observar el recurso creado en el panel de control de Heroku.

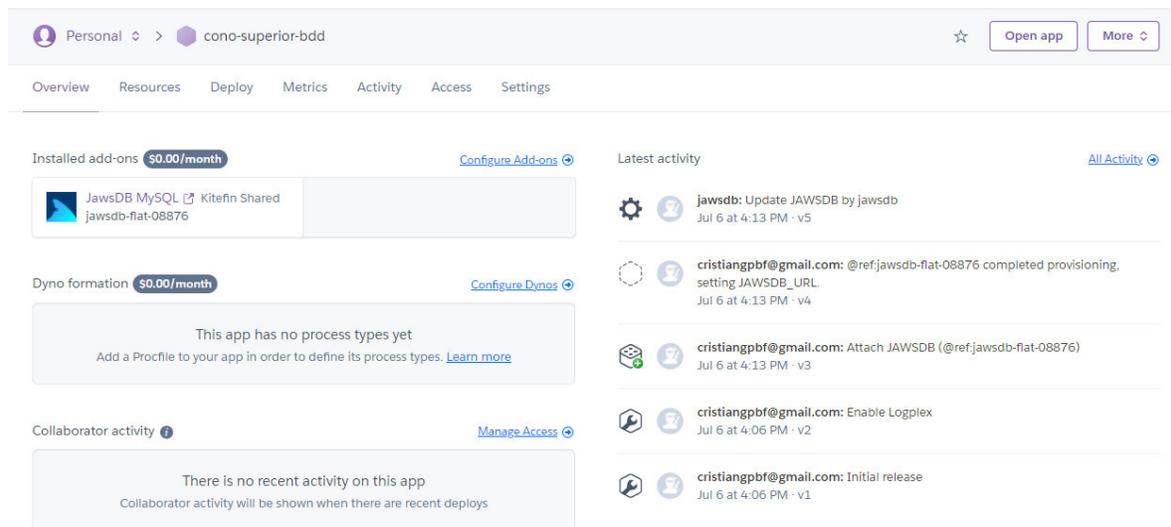


Fig. 7: Recurso de base de datos MySQL en Heroku

3.3. Sprint 2 (API)

En el *Sprint 2* se realizan las actividades correspondientes a la creación de la API Rest para la comunicación con la base de datos.

Objetivos:

- Crear las migraciones para las entidades
- Crear *seeders* para todas las entidades
- Crear los controladores
- Implementar JWT para la autenticación de usuarios
- Establecer las relaciones entre las entidades
- Implementar la subida de imágenes al servidor

- Generar rutas
- Implementar el envío de correos

Resultados:

3.3.1. Crear las migraciones para las entidades

Para este objetivo se crearon todas las migraciones para establecer la estructura de cada tabla necesaria en la base de datos, esto mediante el ORM (*Object Relational Mapper*) Eloquent incluido junto con la instalación de Laravel.

3.3.2. Crear *seeders* para todas las entidades

Se crearon todos los *seeders* necesarios para tener datos ficticios ingresados en las tablas, una vez creada la instancia de la base de datos. Los *seeders* se han creado mediante la librería Faker para PHP.

3.3.3. Crear los controladores

En los controladores se establece la lógica de los métodos a utilizar por cada entidad. En cada controlador se crean los métodos relacionados a cada petición HTTP, es decir, las peticiones GET, POST, PUT, DELETE, junto con variaciones para definir cada *endpoint* que será consumido por los clientes web y móvil.

3.3.4. Implementar JWT para la autenticación de usuarios

Con este objetivo se busca establecer una comunicación segura entre los clientes y la base de datos. Esto mediante el uso de Json Web Tokens (JWT) que permiten proteger las rutas establecidas en los *endpoints* para que solo unas pocas sean accesibles sin la necesidad de que el usuario esté autenticado, mientras que otras rutas son protegidas y requieren de un token válido.

3.3.5. Establecer las relaciones entre las entidades

En este punto se establecen las relaciones entre las entidades creadas anteriormente para que se reflejen en la base de datos mediante las migraciones. De igual forma se establecen relaciones lógicas para la API, de esta forma se facilita la interpretación de las relaciones entre las entidades para su uso en los controladores y en los *endpoints*.

3.3.6. Implementar la subida de imágenes al servidor

El módulo de productos necesita subir imágenes para cada registro, por lo que se ha añadido un enlace simbólico al directorio *storage* del proyecto Laravel en el directorio *public*. Las imágenes se pueden enviar como un objeto de tipo archivo al servidor, y se valida que el mismo sea de tipo imagen.

3.3.7. Generar rutas

Las rutas en el proyecto Laravel definen como se va a acceder a los recursos del sistema. Se definen los métodos HTTP que acepta esta ruta, y los métodos del controlador a los que va a acceder cuando se llame a esta ruta.

Se sitúa a cada una de estas rutas dentro o fuera del *middleware* correspondiente a JWT, de esta forma se puede establecer que rutas son accesibles sin un *token* y cuales no.

3.4. Sprint 3 (Cliente Web)

En el *Sprint 3* se realizan las actividades correspondientes a la creación del cliente web.

Objetivos:

- Implementar el módulo de Inicio de sesión
- Implementar el módulo de clientes
- Implementar el módulo de empleados
- Implementar el módulo de productos
- Implementar el módulo de producción
- Implementar el módulo de pedidos

3.4.1. Implementar el módulo de Inicio de sesión

El usuario puede iniciar sesión en el sistema mediante un correo y contraseña, estos datos se envían a la API la cual valida esta información y devuelve un token de autorización que permite al usuario realizar ciertas acciones disponibles solamente para el usuario autenticado.

La contraseña del usuario se almacena cifrada en la base de datos.

El sistema maneja dos tipos de usuario, el usuario administrador y el usuario cliente, cada uno con un panel de opciones principal que muestra los módulos a los que puede acceder el usuario. Ver la **Fig. 9** y la **Fig. 10**.

3.4.2. Implementar el módulo de clientes

El módulo de clientes permite al usuario listar todos los usuarios registrados en el sistema, registrar nuevos clientes, ver los detalles de cada usuario, cambiar la contraseña, editar la información y eliminar el usuario. El sistema no permite la eliminación de un usuario con pedidos existentes.

Al momento de registrar un nuevo usuario en el sistema, el usuario administrador podrá crear más usuarios con el tipo de usuario Administrador.

Si el usuario listado es el administrador no se permite su eliminación.

3.4.3. Implementar el módulo de empleados

El módulo de empleados permite al usuario administrador listar todos los empleados registrados en el sistema, añadir nuevos empleados, ver la producción de este empleado en específico, editar la información de cada empleado, eliminar un empleado.

El sistema no permite la eliminación de empleados con producciones registradas.

3.4.4. Implementar el módulo de productos

El módulo de productos permite al usuario administrador listar todos los productos, crear productos y editarlos. Si el usuario es un cliente, el sistema le permite listar los productos, ver su información de forma más detallada y añadir productos al carro de compras.

Un usuario no autenticado puede ver todos los productos, sin las opciones exclusivas de usuario autenticado y su tipo de usuario.

3.4.5. Implementar el módulo de producción

El módulo de producción permite al usuario administrador ver la producción de cada empleado. Puede registrar una nueva producción para el empleado y eliminar los registros de producción.

3.4.6. Implementar el módulo de pedidos

El módulo de pedidos permite al usuario administrador listar las órdenes y filtrarlas por su estado, ver el detalle de la orden, cambiar el estado de la orden, asignar o cambiar la fecha estimada de entrega y eliminar los pedidos.

Para los usuarios de tipo cliente, el módulo de pedidos permite consultar todos los pedidos del cliente, filtrarlos por estado, eliminarlo y consultar los detalles de la orden. Los pedidos no pueden ser editados una vez que el estado ha cambiado a "en proceso" o "entregado".

En este hito también se ha desarrollado la funcionalidad correspondiente a la selección de productos por parte de un usuario de tipo cliente, y la posibilidad de añadirlos a un carrito de compras, para posteriormente realizar un pedido.

Resultados:

A continuación, se observan las pantallas resultantes en este *sprint*.

En la **Fig. 8** se muestra la pantalla de inicio para un usuario no autenticado.



Fig. 8: Cliente web - Página de inicio para usuario no autenticado

Una vez el usuario se autentica en el sistema, la pantalla pasara a verse como en la **Fig. 9** si el tipo de usuario es administrador, y como en la **Fig. 10** si el tipo de usuario es un cliente.



Fig. 9: Cliente web - Página de inicio para usuario administrador

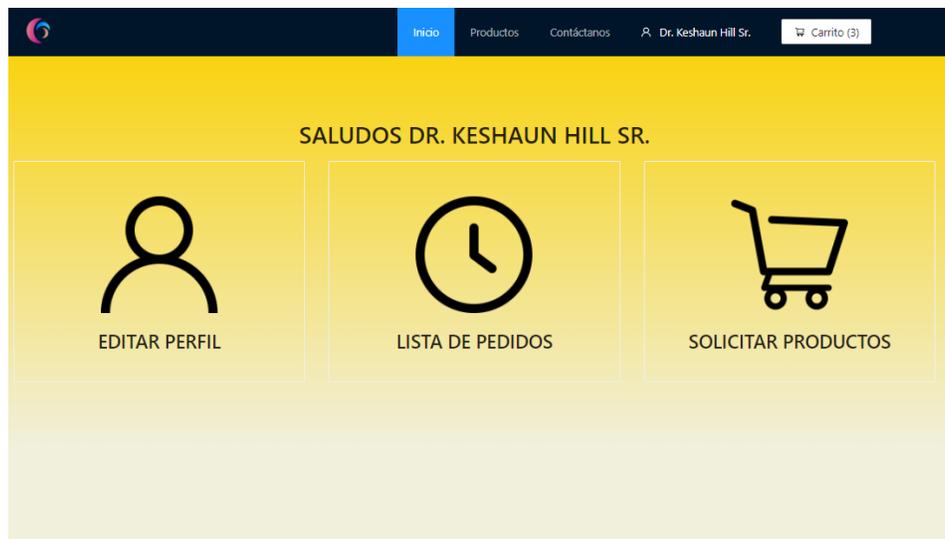


Fig. 10: Cliente web - Página de inicio para usuario cliente

Para iniciar sesión se muestra la pantalla de la **Fig. 11**.

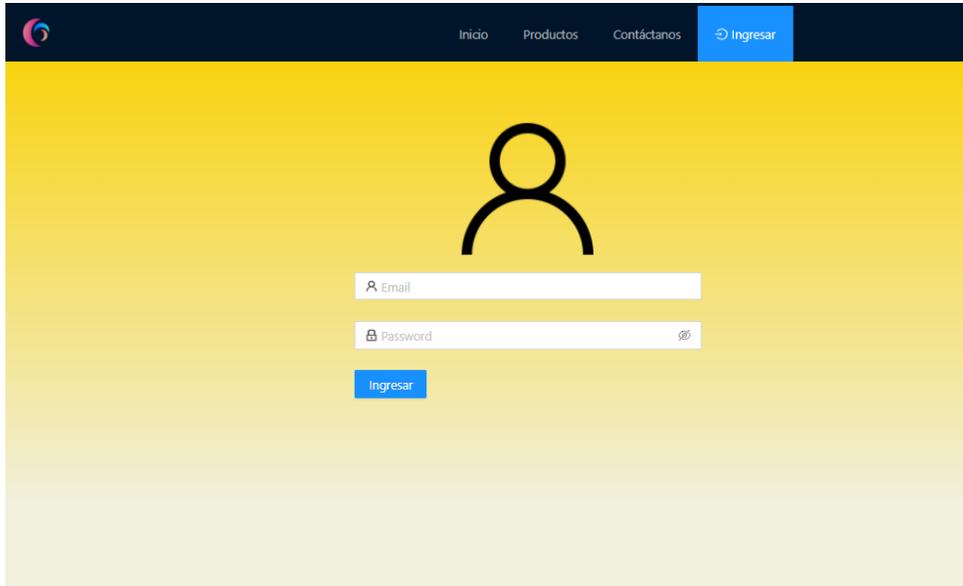


Fig. 11: Cliente web - Página de inicio de sesión cliente web

El módulo Pedidos del usuario administrador se ve como en la **Fig. 12**. En la **Fig. 13** se observa el detalle de un pedido.

#	Cliente	entrega	Orden generada el	Ultima actualización	Opciones	
2	Dr. Fausto Rice	2021-08-24	2021-08-29	2021-08-29	[editar] [+] [eliminar]	
3	Dr. Toni Ullrich V	en proceso	2003-11-03	2021-08-29	2021-08-30	[editar] [+] [eliminar]
4	Leonora Shanahan	en espera	2014-04-13	2021-08-29	2021-08-29	[editar] [+] [eliminar]
5	Joy Fay	entregado	1999-08-18	2021-08-29	2021-08-29	[editar] [+] [eliminar]
6	Nikko Harber	en espera	1994-04-12	2021-08-29	2021-08-29	[editar] [+] [eliminar]
7	Major Ebert	entregado	1974-03-31	2021-08-29	2021-08-30	[editar] [+] [eliminar]
8	Lora Auer	en espera	1982-10-20	2021-08-29	2021-08-29	[editar] [+] [eliminar]
9	Dr. Brannon Gutkowski PhD	en espera	1990-10-24	2021-08-29	2021-08-29	[editar] [+] [eliminar]
10	Jan Kutch	en espera	2006-05-19	2021-08-29	2021-08-29	[editar] [+] [eliminar]

Fig. 12: Cliente web - Listado de pedidos y filtro por estados

The screenshot shows a web client interface with a dark blue header containing navigation links: Inicio, Productos, Contáctanos, and Administrador. The main content area has a yellow background and displays the following information:

Orden #2

Fecha de solicitud: 2021-08-29
 Fecha de entrega: 1980-04-24

Estado: en espera
 Mensaje del cliente: Aliquid qui molestias ex est fugit molestias nisi non. Rerum ipsa occaecati quia quisquam tempore provident eligendi temporibus.

Código Producto	Nombre Producto	Cantidad
7	producto7	500
13	producto13	163
12	producto12	179

Fig. 13: Cliente web - Detalle de una orden

En la Fig. 14 y la Fig. 15 se observan las pantallas del módulo de clientes.

The screenshot shows a web client interface with a dark blue header containing navigation links: Inicio, Productos, Contáctanos, and Administrador. The main content area has a yellow background and displays the following information:

LISTA DE USUARIOS

Registar nuevo usuario

	Nombre	Razón social	Identificación	Teléfono	Dirección	Tipo	Email	Opciones
+	Administrador	Cono Superior	00000000001	0999999999	Condores	admin	admin@prueba.com	
+	Dr. Keshawn Hill Sr.	Jacques Friesen V	4539202869712582	+17542819819	27739 Pearlle Point Apt. 870 Leolashire, HI 97811	client	geoffrey.zulauf@yahoo.com	
+	Dr. Fausto Rice III	Harrison Durgan	5295174086085541	1-641-302-6533	412 Cordie Heights New Dakota, SC 22088-4882	client	vincenza.towne@hotmail.com	
+	Dr. Toni Ullrich V	Julio Harris	4929494901896782	445.649.3072	150 Mathew Field Apt. 248 Hoegerton, AZ 68592	client	ncorwin@hotmail.com	
+	Leonora Shanahan	Dr. Teagan Rath MD	5522950194056523	804-256-7499	47423 Lucinda Way Apt. 362 West Aureliaville, VT 43945-9048	client	uwuckert@larkin.com	
+	Joy Fay	Dr. Judd Kutch	6011491234835432	+12396937721	23451 Carter Bridge East Danmouth, MO 34218-9565	client	joesph98@emser.org	
+	Nikko Harber	Prof. Lacy Douglas	4024007169442554	1-320-929-8566	856 Grimes Forest Apt. 994 South Telly, OK 51709-4102	client	bernice29@yahoo.com	
+	Major Ebert	Nya Brakus DVM	4916300098510653	231.779.2940	9315 Aliza Landing New Arielle, WV 20849	client	guillermo85@buckridge.info	
+	Lora Auer	Mrs. Nikki Frami	4539994684787670	(954) 719-6067	7233 Moore Underpass Karifort, WV 53802-4882	client	schneider.terrance@johns.com	
+	Dr. Brannon Gutkowski PhD	Cooper Gutkowski	2720202710748374	+1-925-796-7218	978 Morisette Park West Meganeville, KS 22208	client	oda.mann@gmail.com	
+	Jan Kutch	Billie Kutch	4556708291411	+14024443800	2775 Becker Junction Apt. 758 Rogahnside, MA 08273-0547	client	iiittle@hotmail.com	

Fig. 14: Cliente web - Listado de usuarios

The image shows a web registration form titled "Registro" on a yellow background. At the top, there is a dark blue navigation bar with a logo on the left and links for "Inicio", "Productos", "Contáctanos", and "Administrador" on the right. The form itself is centered and contains the following fields from top to bottom: a text input for "Nombre", a text input for "Nombre de la Empresa", a text input for "Número de ruc", a text input for "Número de teléfono", a text input for "Dirección", a dropdown menu for "Selección tipo de usuario", a text input for "Email", a password input for "Clave" with a toggle icon, and a second password input for "Confirma tu clave" with a toggle icon. A blue "Regístrate" button is located at the bottom of the form.

Fig. 15: Cliente web - Registro de usuarios

3.5. Sprint 4 (Cliente Móvil)

En el *Sprint 4* se realizaron las actividades correspondientes a la creación del cliente web.

Objetivos:

- Implementar el módulo de Inicio de sesión.
- Visualizar el listado de clientes.
- Visualizar el listado de empleados y su producción.
- Implementar el módulo de productos.
- Implementar el módulo de pedidos.

3.5.1. Implementar el módulo de Inicio de sesión.

El usuario puede iniciar sesión en el cliente móvil mediante un correo y una contraseña, estos datos son enviados a la API, la cual validará los mismos, en el caso de que el usuario exista, nos proporcionará un token de autorización, permitiendo al usuario realizar las diferentes acciones según el rol correspondiente.

La contraseña del usuario se almacena cifrada en la base de datos.

Los clientes manejan dos tipos de usuario, el usuario administrador y el usuario cliente, cada uno con un menú de opciones que muestra los módulos a los que puede acceder el usuario.

3.5.2. Visualizar el listado de clientes.

El administrador puede listar todos los clientes registrados en el sistema, donde se presenta la información más relevante de cada usuario. El cliente móvil no permite la eliminación, edición y agregación de un usuario. Ver **Fig. 19**.

3.5.3. Visualizar el listado de empleados y su producción.

Esta función permite al administrador, listar todos los empleados registrados en el sistema (**Fig. 20**), además de ver la producción registrada de cada empleado de manera individual (**Fig. 21**).

El cliente móvil no permite añadir, editar o eliminar esta información.

3.5.4. Implementar el módulo de productos

El módulo de productos permite al cliente listar todos los productos que ofrece la empresa, ver la información más relevante de los mismos y mostrar la imagen de cada producto de manera individual (**Fig. 22**). El cliente también puede añadir varios productos a un carrito de compras, para realizar la solicitud de un pedido con los productos seleccionados.

El cliente móvil no permite añadir, editar o eliminar productos.

3.5.5. Implementar el módulo de pedidos

El módulo de pedidos permite al usuario administrador listar las órdenes y filtrarlas por su estado, además de atender cada una de ellas, donde puede asignar una fecha de entrega y cambiar el estado de la orden a “en proceso” o “entregado”, también puede ver los productos solicitados en cada una de las órdenes con su información relevante (**Fig. 23**).

Para un usuario de tipo cliente, este módulo permite consultar los pedidos que se encuentran en estado “en espera”, dando la opción al cliente de editar o eliminar el pedido mientras el estado del mismo no cambie. Si el estado del pedido cambia a “en proceso” o “entregado” el cliente solo puede visualizar los detalles de su pedido (**Fig. 24**).

En este hito también se ha desarrollado la funcionalidad correspondiente a la selección de productos por parte de un usuario de tipo cliente, y la posibilidad de añadirlos a un carrito de compras, para posteriormente agregar un comentario si el cliente lo desea y solicitar un pedido.

Resultados:

A continuación, se observan las pantallas resultantes en este *sprint*.

En la **Fig. 16** se muestra la pantalla de inicio de sesión para los usuarios.

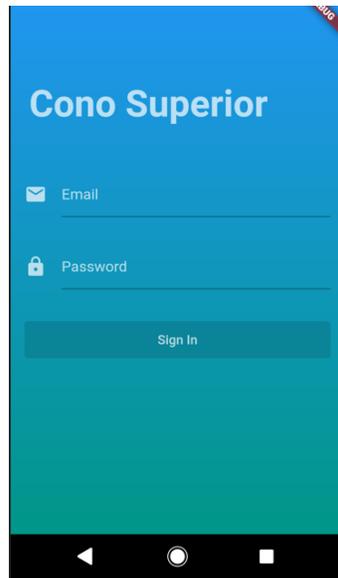


Fig. 16: Cliente móvil - Página de inicio de sesión

Una vez el usuario esta autenticado podrá acceder a los diferentes módulos según su rol, si el usuario es de tipo administrador podrá visualizar el menú como se muestra en la **Fig. 17**, y si el usuario es de tipo cliente muestra el menú como en la **Fig. 18**.

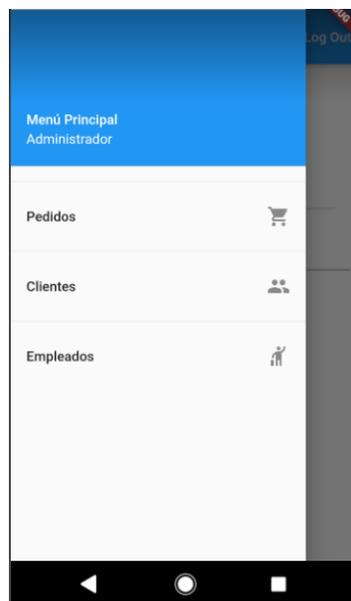


Fig. 17: Cliente móvil - Menú administrador

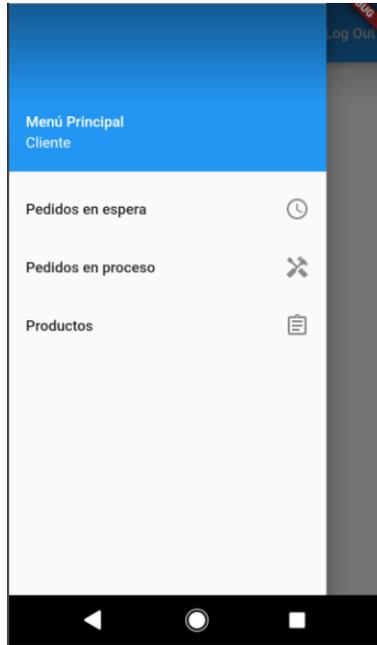


Fig. 18: Cliente móvil - Menú cliente

En el módulo de clientes se presenta la información, como se muestra en la **Fig. 19**.

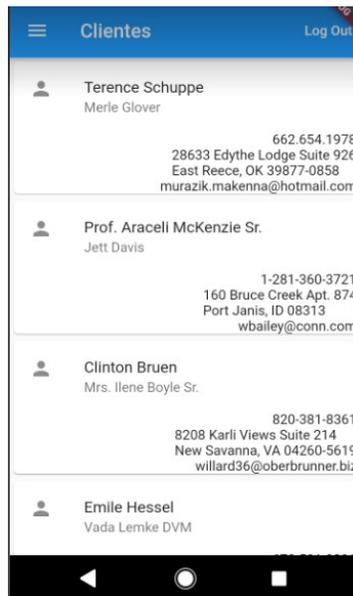


Fig. 19: Cliente móvil – Página de lista de clientes

El módulo de empleados y producción muestra la lista de empleados **Fig. 20**, además de la producción individual de los mismos **Fig. 21**.

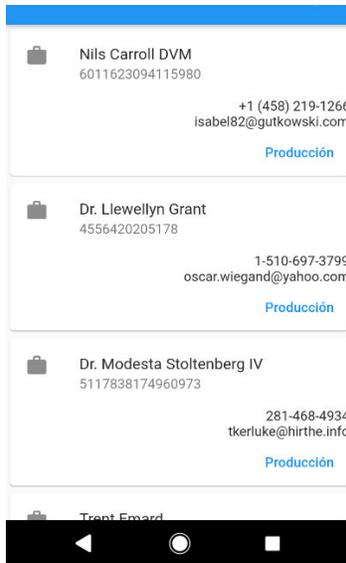


Fig. 20: Cliente móvil – Página lista de empleados



Fig. 21: Cliente móvil – Producción empleado

La **Fig. 22** muestra el módulo de productos presentados al cliente, los cuales pueden ser añadidos a un carrito de compras.



Fig. 22: Cliente móvil - Catálogo de productos

En la **Fig. 23** se muestra el módulo de pedidos presentado al administrador y la **Fig. 24** muestra el módulo de pedidos del cliente.



Fig. 23: Cliente móvil - Pedidos administrador

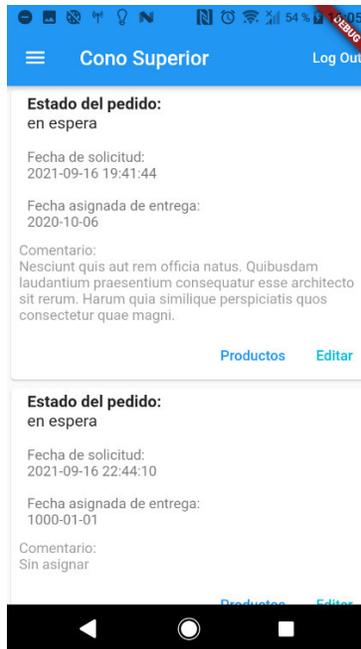


Fig. 24: Cliente móvil - Pedidos cliente

3.6. Sprint 5 (Publicación y pruebas)

En este *Sprint* se ha realizado el proceso de publicación para los componentes del proyecto, junto con las pruebas para cada uno de estos.

Objetivos:

- Desplegar la API y cliente
- Realizar pruebas de aceptación
- Realizar pruebas de compatibilidad en cliente web
- Realizar pruebas de compatibilidad en cliente móvil
- Realizar pruebas de carga y estrés en API Rest

3.6.1. Desplegar la API y los clientes web y móvil

Luego de haber finalizado la etapa de desarrollo, se procede a publicar cada componente del proyecto a un ambiente en línea, para posteriormente realizar las pruebas necesarias.

Despliegue API Rest

Para desplegar la API Rest, se ha utilizado el proveedor de servidores virtuales DigitalOcean, el cual permite un despliegue simplificado para servicios web mediante la creación de contenedores.

En la **Fig. 25** se observa el recurso desplegado mediante el panel de control de DigitalOcean.

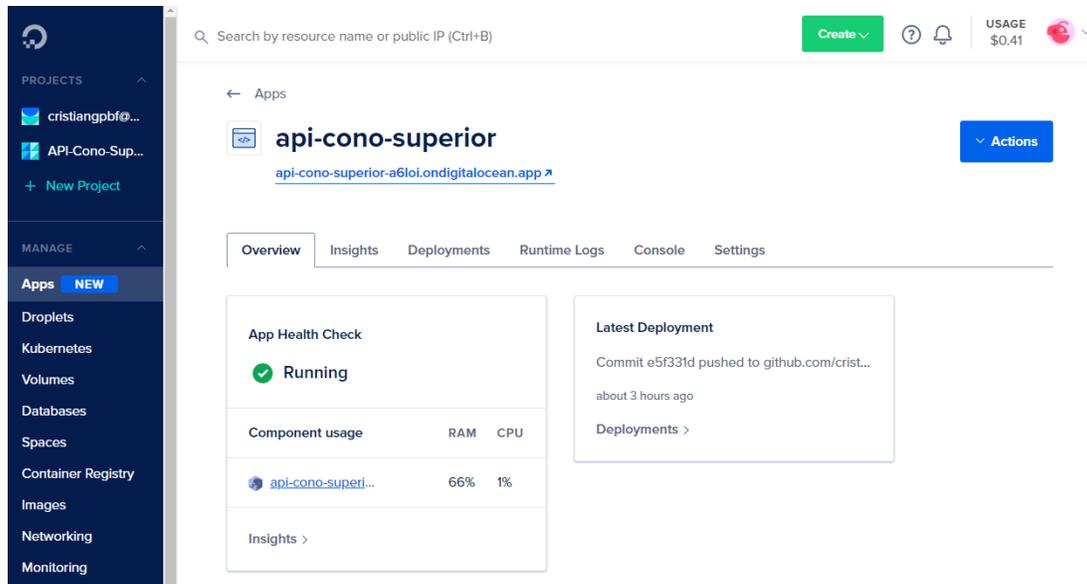


Fig. 25: Despliegue API Rest

Despliegue Cliente Web

Para desplegar el cliente web, se ha utilizado la plataforma web Heroku, en donde se enlaza el proyecto de GitHub directamente con el recurso generado en el panel de control de la plataforma. Esto permite un despliegue automatizado que se ejecuta cada vez que se realiza la acción *push* a la rama principal del proyecto.

En la **Fig. 26** se encuentra el recurso desplegado, accesible mediante el protocolo HTTPS.

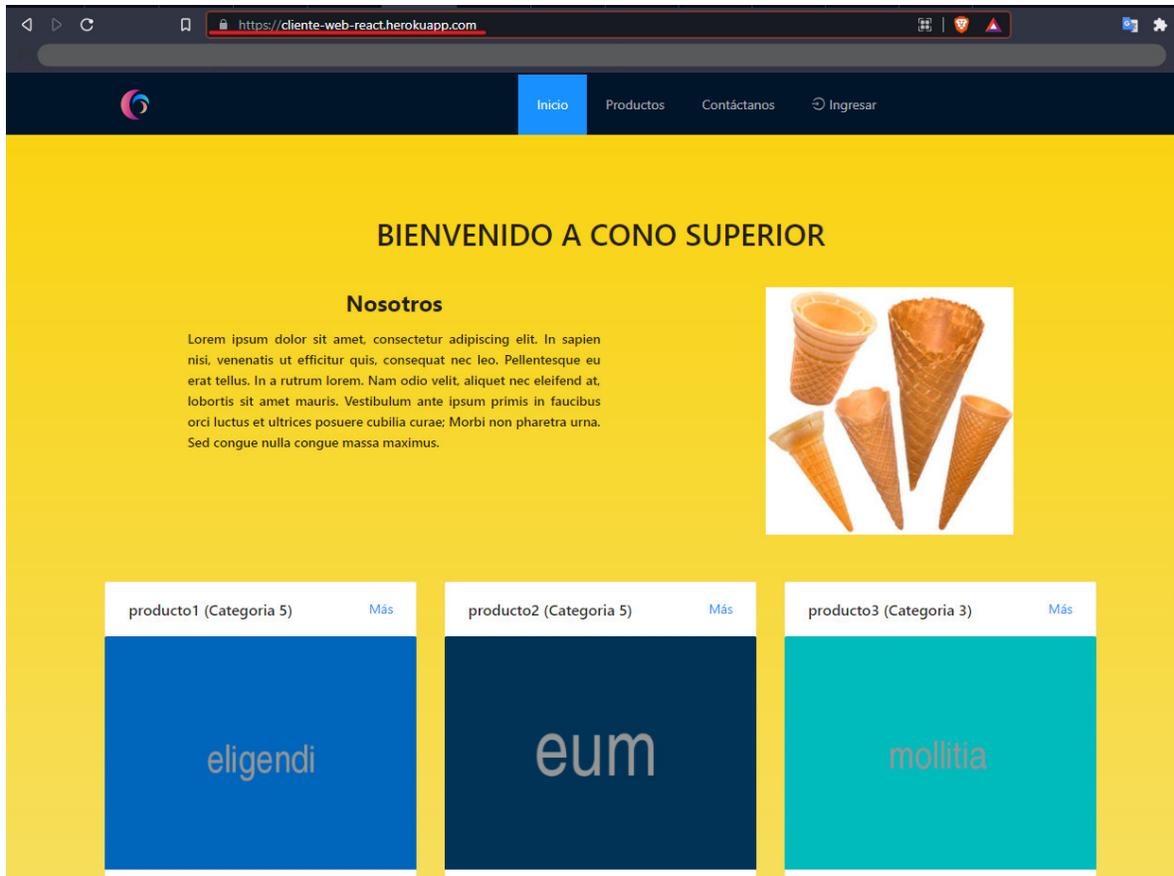


Fig. 26: Cliente web publicado mediante Heroku

3.6.2. Realizar pruebas de aceptación

Las pruebas de aceptación buscan validar el funcionamiento de los clientes en función de los requerimientos planteados en una instancia inicial del proyecto.

A continuación, en la

TABLA VI se muestra un ejemplo del formato utilizado para las pruebas de aceptación de los clientes.

TABLA VI: Ejemplo de prueba de aceptación

Prueba N.- 20	Identificador: PHU20	Historia de usuario: 20
Nombre: Realizar solicitud de pedido		
Descripción: Los clientes web y móvil deben permitir que el usuario realice solicitudes de pedidos a la empresa Cono Superior.		
Pasos de ejecución:		
<ol style="list-style-type: none"> 1. Iniciar sesión como cliente 2. Dirigirse a la opción Carrito de compras 3. Verificar campos de llenado opcionales 4. Presionar el botón realizar pedido 		

Resultado deseado: El pedido ha sido realizado, ahora se encuentra en la sección de pedidos.

Resultado luego de la prueba: Satisfactorio

El detalle de todas las pruebas de aceptación se puede encontrar en el anexo Manual Técnico, sección siete.

3.6.3. Realizar pruebas de compatibilidad en cliente web

Las pruebas de compatibilidad buscan asegurar el correcto funcionamiento del cliente web en diferentes ambientes. Estas pruebas de compatibilidad se han realizado en cuatro navegadores webs diferentes, en dos sistemas operativos. Los navegadores elegidos fueron Firefox, Chrome, Edge y Opera, mientras que los sistemas operativos elegidos fueron Windows 10 para computadores personales y Android 11 para dispositivos móviles.

En la **Fig. 27** se observa la prueba realizada en el navegador Firefox en un computador personal con Windows 10.



Fig. 27: Ejemplo prueba de compatibilidad cliente web

El detalle de todas las pruebas de compatibilidad del cliente web se pueden encontrar en el anexo Manual Técnico, sección siete.

3.6.4. Realizar pruebas de compatibilidad en cliente móvil

Las pruebas de compatibilidad del cliente móvil permiten identificar si la aplicación funciona correctamente en distintos dispositivos o sistemas. La **Fig. 28** muestra los resultados

obtenidos de las pruebas de compatibilidad del cliente móvil, realizado en un dispositivo SONY Xperia L2 con un SO Android versión 7.1.1



Fig. 28: Catálogo de productos en Sony Xperia L2

3.6.5. Realizar pruebas de carga y estrés en API Rest

Para medir las capacidades de respuesta de la API REST en el servidor de DigitalOcean, se realizaron pruebas de carga y estrés mediante la herramienta Apache JMeter.

Para las pruebas de carga se realizaron 60 peticiones a cinco rutas, en periodos de subida de un segundo, en cinco bucles, resultando en 300 peticiones por ruta y 1500 peticiones en total (Fig. 29).

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Están...	% Error	Rendimiento
/categories	300	2812	2276	4185	383,30	0,00%	3,0/sec
/login admin	300	4462	2389	6380	1062,19	0,00%	2,9/sec
/login clients	300	5811	2762	6449	465,79	0,00%	2,9/sec
/products	300	5021	126	6470	1328,26	0,33%	3,1/sec
/products/#	300	3791	2393	5501	809,06	0,00%	3,0/sec
Total	1500	4379	126	6470	1355,10	0,07%	13,3/sec

Fig. 29: Resultados prueba de carga

Para las pruebas de estrés se ha realizado una carga total de 3500 peticiones en total distribuidas entre las cinco rutas elegidas para las pruebas de carga, esto debido a que el

plan elegido para el servidor de base de datos no permite exceder ese número de consultas para un solo usuario (**Fig. 30**).

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Están...	% Error	Rendimiento
/products	700	36124	561	124116	32167,41	4,14%	2,3/sec
/products/#	700	19421	274	52322	9467,57	2,00%	2,3/sec
/categories	700	18137	221	46772	7726,32	2,14%	2,3/sec
/login admin	700	20748	279	111178	11818,32	3,00%	2,1/sec
/login clients	700	21054	253	100236	11022,21	2,86%	2,1/sec
Total	3500	23097	221	124116	18235,93	2,83%	10,5/sec

Fig. 30: Resultados prueba de estrés

El detalle de las pruebas realizadas se puede encontrar en el anexo Manual Técnico, sección siete.

4. CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

- El cliente web cumple con todas las especificaciones del cliente, permitiendo realizar las actividades de administración de una manera cómoda desde un computador y disminuyendo el uso de medios físicos en papel para la persistencia de información relevante para el administrador.
- Los clientes controlan de manera adecuada el acceso de usuarios a los diferentes módulos y pantallas, en función de los permisos otorgados por la sesión existente.
- La realización del proyecto ha ocurrido de manera fluida y organizada gracias a la utilización de *Scrum*.
- El uso de la herramienta Heroku para el alojamiento en la nube de la base de datos y el cliente web, facilita mucho el proceso de despliegue de dichos componentes a un ambiente en la nube.
- El uso de la herramienta DigitalOcean para el alojamiento de la API, facilita mucho el proceso de despliegue de este componente, con un costo muy bajo.
- Las pruebas de estrés nos permiten identificar la ruta de la API con la mayor demanda de recursos.
- Se han logrado adecuar varios componentes del cliente web, para que este sea *responsive* y se muestre correctamente en un ambiente móvil

4.2. Recomendaciones

- Tener en mente la cantidad de interacciones que se puede hacer con la base de datos en Heroku por hora, pues el plan de entrada solo permite 3500 peticiones en una hora.
- Pese a que el cliente web se ha desarrollado para mostrarse en un ambiente móvil, para una mayor comodidad al momento de operar el sistema, se recomienda usar un ambiente de escritorio.
- Asegurarse de tener siempre al menos un usuario tipo administrador creado para el sistema. Desde este primer usuario administrador es posible seguir creando más usuarios para el sistema.

5. REFERENCIAS BIBLIOGRÁFICAS

- [1] «ISOTools,» [En línea]. Available: <https://www.isotools.org/2018/03/28/por-que-automatizar-los-procesos-en-tu-organizacion/>. [Último acceso: 29 Agosto 2020].
- [2] World Health Organization, «COVID-19: cronología de la actuación de la OMS,» 27 abril 2020. [En línea]. Available: <https://www.who.int/es/news/item/27-04-2020-who-timeline---covid-19>. [Último acceso: 06 junio 2021].
- [3] J. Ortega, «Josmarketing,» 31 Marzo 2020. [En línea]. Available: <https://josmarketing.es/que-es-la-digitalizacion-de-empresas/>. [Último acceso: 26 Febrero 2021].
- [4] CEPAL, «Boletín: Facilitación del comercio y el transporte en América Latina y el Caribe,» Octubre 2007. [En línea]. Available: https://repositorio.cepal.org/bitstream/handle/11362/36145/1/FAL254e_es.pdf. [Último acceso: 26 Febrero 2021].
- [5] I. S. Vásquez, Interviewee, *Conocimiento de la empresa*. [Entrevista]. 20 Junio 2020.
- [6] Samuel, «Mygestion,» 19 Octubre 2020. [En línea]. Available: <https://www.mygestion.com/blog/como-digitalizar-la-gestion-de-almacen>. [Último acceso: 26 Febrero 2020].
- [7] D. Estrada, «easycodigo,» [En línea]. Available: <https://easycodigo.com/ventajas-de-una-plataforma-web-para-tu-empresa/>. [Último acceso: 29 Agosto 2020].
- [8] I. VMEdU, Una guía para el Cuerpo de Conocimiento de Scrum, Arizona: SCRUMstudy, 2017.
- [9] Kaizenia, «¿Qué son los Artefactos de Scrum?,» [En línea]. Available: <https://kzi.mx/que-son-los-artefactos-de-scrum/>. [Último acceso: 10 julio 2021].
- [10] Scrum México, «Escribiendo Historias de Usuario,» 2 agosto 2018. [En línea]. Available: <https://scrum.mx/informate/historias-de-usuario>. [Último acceso: 20 julio 2021].
- [11] R. Sánchez, M. Pérez, R. Gómez, E. Sosa y R. Castillo, «Innovación y Desarrollo Tecnológico Revista Digital,» 2020 abril 1. [En línea]. Available: https://iydt.files.wordpress.com/2020/06/2_7_implementacion-de-patrones-de-diseno-en-arquitectura-api-rest.pdf. [Último acceso: 15 julio 2021].
- [12] Facebook Inc, «React,» 2021. [En línea]. Available: <https://es.reactjs.org/>. [Último acceso: 2021].
- [13] Flutter, «Flutter documentation,» 2021. [En línea]. Available: <https://flutter.dev/?gclid=ds&gclid=ds>. [Último acceso: 2021].

6. ANEXOS

6.1. Manual Técnico

- Levantamiento de requerimientos
- Historias de Usuario
- *Product Backlog*
- *Sprint Backlog*
- Diseño de interfaces
- Diseño de Base de Datos
- Pruebas

Enlace: <https://github.com/Cristiangpbf/Proyecto-Integrador-Curipoma-Guamba/tree/main/MANUAL%20T%C3%89CNICO>

6.2. Manual de Usuario

El video correspondiente a las instrucciones de uso y funcionalidades de este proyecto, se encuentran en el siguiente enlace externo:

Enlace: <https://drive.google.com/drive/folders/19vyMaevg0jW9HRfHa-wFjIQB0n-eohoy?usp=sharing>

6.3. Manual de Instalación

El proceso de instalación para el correcto funcionamiento de este proyecto se encuentra detallado en el archivo *README* del repositorio de GitHub utilizado en este proyecto, el cual se encuentra en el siguiente enlace:

Enlace: <https://github.com/Cristiangpbf/Proyecto-Integrador-Curipoma-Guamba>