

Pontificia Universidad Católica del Ecuador

Facultad De Ingeniería

Escuela de Sistemas



TEMA:

DISEÑO Y DESARROLLO DE UNA APLICACIÓN WEB QUE PERMITA ADMINISTRAR
CLIENTES, INVENTARIO Y PEDIDOS. CASO DE ESTUDIO: ARTEMISA.

AUTOR:

DAVID ANDRÉS BARRENO MONTESDEOCA

DIRECTOR:

JORGE ALARCÓN, Mtr

TRABAJO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS DE
INFORMACIÓN

QUITO, AGOSTO 2022

DEDICATORIA

Dedico este trabajo a mis padres Paco (Q.E.P.D.) y Lily por brindarme su sacrificio, amor incondicional y sobre todo su cálida compañía a lo largo de todo mi crecimiento personal. También le dedico este trabajo a mi hermano Francisco y a mi gatita Mía, que siempre se encuentran presentes. Por último, dedico este trabajo a todas las personas que se han involucrado en mi vida, ya sea brindándome conocimientos o su tiempo.

AGRADECIMIENTO

Quiero agradecer infinitamente a mi familia, pero especialmente a mi madre, que ha sacrificado gran parte de su vida por dármele todo. A mis compañeros de la universidad, que siempre se encuentran disponibles cuando uno los necesita, ya sea para jugar o para aprender juntos. También quiero agradecer a mis profesores, especialmente al Mgtr. Jorge Alarcón, quienes me han apoyado durante este proceso de aprendizaje, inculcándonos las bases fundamentales para desarrollarnos en el ámbito profesional. A la Pontificia Universidad Católica del Ecuador por la formación brindada.

RESUMEN

El presente trabajo de titulación presenta el diseño y desarrollo del aplicativo web enfocado a la automatización de procesos de la empresa Artemisa Moda EC. La empresa Artemisa fue fundada por Adrián Calvopiña en el año 2017 en la ciudad de Quito, Ecuador. Se encuentra ubicada en San Isidro del Inca y su principal actividad económica es la fabricación y venta de productos textiles con precios competitivos en el territorio ecuatoriano. A lo largo de su funcionamiento, realizaban sus procesos principales de formas anticuadas, por ejemplo, anotando sus pedidos en hojas de Excel o simplemente en un cuaderno. Es así como nació la necesidad de la empresa de desarrollar un aplicativo personalizado con el objetivo de ser más competitivos en el mercado y automatizar sus operaciones diarias. Como resultado del desarrollo y mediante el uso de la metodología de programación extrema (XP) se obtuvo un aplicativo que utiliza tecnologías y herramientas novedosas, como lo son Angular, Firebase y Bootstrap, que aportaron con librerías, marcos de trabajo y servicios para la construcción de la aplicación en un corto tiempo.

ÍNDICE DE FIGURAS, GRÁFICOS Y TABLAS

ÍNDICE DE FIGURAS

Figura 1 Flujo del Modelo-Vista-Controlador	18
Figura 2 Flujo de la arquitectura de microservicios	19
Figura 3 Caso de uso: Nivel general	40
Figura 4 Caso de uso: Siguiete nivel, F1 Administrar clientes.....	41
Figura 5 Caso de uso: Nivel detallado, F1.1 Ingresar clientes.....	41
Figura 6 Caso de uso: Nivel detallado, F1.2 Modificar clientes	43
Figura 7 Caso de uso: Nivel detallado, F1.3 Eliminar clientes.....	44
Figura 8 Caso de uso: Nivel detallado, F1.4.1 Consulta general.....	45
Figura 9 Caso de uso: Nivel detallado, F1.4.2 Consulta por parámetros	47
Figura 10 Caso de uso: Siguiete nivel, F2 Administrar categorías	48
Figura 11 Caso de uso: Nivel detallado, F2.1 Ingresar categorías.....	49
Figura 12 Caso de uso: Nivel detallado, F2.2 Modificar categorías	50
Figura 13 Caso de uso: Nivel detallado, F2.3 Eliminar categorías	52
Figura 14 Caso de uso: Nivel detallado, F2.4.1 Consulta general.....	53
Figura 15 Caso de uso: Nivel detallado, F2.4.2 Consulta por parámetros	54
Figura 16 Caso de uso: Siguiete nivel, F3 Administrar inventario	56
Figura 17 Caso de uso: Nivel detallado, F3.1 Ingresar inventario	56
Figura 18 Caso de uso: Nivel detallado, F3.2 Modificar inventario.....	57
Figura 19 Caso de uso: Nivel detallado, F3.3 Eliminar inventario	59
Figura 20 Caso de uso: Nivel detallado, F3.4.1 Consulta general.....	61
Figura 21 Caso de uso: Nivel detallado, F3.4.2 Consulta por parámetros	62
Figura 22 Caso de uso: Siguiete nivel, F4 Administrar pedidos	63

Figura 23 Caso de uso: Nivel detallado, F4.1 Ingresar pedidos	64
Figura 24 Caso de uso: Nivel detallado, F4.2 Modificar pedidos.....	66
Figura 25 Caso de uso: Nivel detallado, F4.3 Eliminar pedidos	67
Figura 26 Caso de uso: Nivel detallado, F4.4.1 Consulta general.....	69
Figura 27 Caso de uso: Nivel detallado, F4.4.2 Consulta por parámetros	70
Figura 28 Caso de uso: Siguiete nivel, F5 Administrar empleados	72
Figura 29 Caso de uso: Nivel detallado, F5.1 Ingresar empleados	72
Figura 30 Caso de uso: Nivel detallado, F5.2 Modificar empleados.....	74
Figura 31 Caso de uso: Nivel detallado, F5.3 Eliminar empleados	75
Figura 32 Caso de uso: Nivel detallado, F5.4.1 Consulta general.....	77
Figura 33 Caso de uso: Nivel detallado, F5.4.2 Consulta por parámetros	78
Figura 34 Ejemplo de estándar de codificación adoptado	80
Figura 35 Resultados de pruebas Karma de la aplicación.....	106

ÍNDICE DE TABLAS

Tabla 1 Puntaje de metodologías de desarrollo	27
Tabla 2 Planes de precios de Firebase.....	30
Tabla 3 Puntuación de Frameworks o servicios para Backend.....	32
Tabla 4 Puntuaciones de Frameworks para el Frontend.....	34
Tabla 5 Puntuaciones de sistemas de gestión de base de datos	37
Tabla 6 Excepciones Caso de Uso: Ingresar Clientes	42
Tabla 7 Excepciones Caso de Uso: Modificar Clientes	44
Tabla 8 Excepciones Caso de Uso: Eliminar Clientes.....	45
Tabla 9 Excepciones Caso de Uso: Consulta general de clientes.....	46
Tabla 10 Excepciones Caso de Uso: Consulta por parámetros de clientes	48
Tabla 11 Excepciones Caso de Uso: Ingresar categorías	50
Tabla 12 Excepciones Caso de Uso: Modificar categorías	51
Tabla 13 Excepciones Caso de Uso: Eliminar categorías.....	53
Tabla 14 Excepciones Caso de Uso: Consulta general de categorías	54
Tabla 15 Excepciones Caso de Uso: Consulta por parámetros de categorías	55
Tabla 16 Excepciones Caso de Uso: Ingresar inventario.....	57
Tabla 17 Excepciones Caso de Uso: Modificar inventario	59
Tabla 18 Excepciones Caso de Uso: Eliminar inventario	60
Tabla 19 Excepciones Caso de Uso: Consulta general de inventario	62
Tabla 20 Excepciones Caso de Uso: Consulta por parámetros de inventario.....	63
Tabla 21 Excepciones Caso de Uso: Ingresar pedidos.....	65
Tabla 22 Excepciones Caso de Uso: Modificar pedidos	67
Tabla 23 Excepciones Caso de Uso: Eliminar pedidos	68
Tabla 24 Excepciones Caso de Uso: Consulta general de pedidos	70
Tabla 25 Excepciones Caso de Uso: Consulta por parámetros de pedidos.....	71

Tabla 26 Excepciones Caso de Uso: Ingresar empleados.....	73
Tabla 27 Excepciones Caso de Uso: Modificar empleados	75
Tabla 28 Excepciones Caso de Uso: Eliminar empleado.....	76
Tabla 29 Excepciones Caso de Uso: Consulta general de empleados	78
Tabla 30 Excepciones Caso de Uso: Consulta por parámetros de empleados.....	79
Tabla 31 Resumen de estados de Iteración 1	82
Tabla 32 Caso de Prueba de Aceptación HU1_P01	82
Tabla 33 Caso de Prueba de Aceptación HU1_P02	83
Tabla 34 Caso de Prueba de Aceptación HU1_P03	84
Tabla 35 Caso de Prueba de Aceptación HU1_P04	84
Tabla 36 Caso de Prueba de Aceptación HU1_P05	85
Tabla 37 Caso de Prueba de Aceptación HU1_P06	86
Tabla 38 Resumen de estados de Iteración 2	87
Tabla 39 Caso de Prueba de Aceptación HU2_P01	87
Tabla 40 Caso de Prueba de Aceptación HU2_P02	88
Tabla 41 Caso de Prueba de Aceptación HU2_P03	89
Tabla 42 Caso de Prueba de Aceptación HU2_P04	89
Tabla 43 Caso de Prueba de Aceptación HU3_P01	90
Tabla 44 Caso de Prueba de Aceptación HU3_P02	91
Tabla 45 Caso de Prueba de Aceptación HU3_P03	92
Tabla 46 Caso de Prueba de Aceptación HU3_P04	92
Tabla 47 Caso de Prueba de Aceptación HU3_P05	93
Tabla 48 Caso de Prueba de Aceptación HU3_P06	94
Tabla 49 Resumen de estados de Iteración 3	95
Tabla 50 Caso de Prueba de Aceptación HU4_P01	96
Tabla 51 Caso de Prueba de Aceptación HU4_P02	96

Tabla 52 Caso de Prueba de Aceptación HU4_P03	97
Tabla 53 Caso de Prueba de Aceptación HU4_P04	98
Tabla 54 Caso de Prueba de Aceptación HU5_P01	99
Tabla 55 Caso de Prueba de Aceptación HU5_P02	99
Tabla 56 Caso de Prueba de Aceptación HU5_P03	100
Tabla 57 Caso de Prueba de Aceptación HU5_P04	101
Tabla 58 Caso de Prueba de Aceptación HU5_P05	102
Tabla 59 Resumen de estados de Iteración 4	103
Tabla 60 Caso de Prueba de Aceptación HU6_P01	103
Tabla 61 Caso de Prueba de Aceptación HU7_P01	104
Tabla 62 Caso de Prueba de Aceptación HU8_P01	104

ÍNDICE DE CONTENIDOS

ÍNDICE DE FIGURAS, GRÁFICOS Y TABLAS	I
ÍNDICE DE FIGURAS	I
ÍNDICE DE TABLAS.....	III
CAPÍTULO I: INTRODUCCIÓN.....	9
1. INTRODUCCIÓN.....	9
1.1. Justificación.....	9
1.2. Planteamiento del problema.....	10
1.3. Objetivo General	11
1.4. Objetivos Específicos	11
1.5. Antecedentes	11
1.6. Alcance	13
CAPÍTULO II: FUNDAMENTACIÓN TEÓRICA	15
2. Marco Teórico.....	15
2.1. ¿Qué es un proceso y cuáles son los beneficios de automatizarlos?.....	15
2.2. Aplicaciones Web	15
2.3. Páginas web adaptable	15
2.4. Arquitecturas de Software	16
2.5. Frameworks	19
2.6. Sistema gestor de bases de datos (SGBD)	21
2.7. Plataforma de WhatsApp para negocios.....	21
CAPÍTULO III: HERRAMIENTAS Y METODOLOGÍA	23

3.	Herramientas y metodología para el desarrollo de la aplicación	23
3.1.	Comparación y selección de la metodología para el desarrollo de la aplicación ..	23
3.2.	Comparación y selección de los Frameworks para el desarrollo web	28
3.3.	Comparación y selección del sistema de gestión de base de datos para el desarrollo de la aplicación	35
3.3.1.	MySQL.....	35
3.3.2.	MariaDB.....	35
3.3.3.	Firebase Firestore	36
CAPÍTULO IV: ANÁLISIS Y DISEÑO		38
4.	Análisis y diseño.....	38
4.1.	Historias de usuario	38
4.2.	Requerimientos funcionales	39
4.3.	Requerimientos no funcionales	39
4.4.	Casos de uso	40
CAPÍTULO V: Desarrollo		80
5.	Desarrollo del sistema.....	80
5.1.	Estándares de programación	80
5.2.	Organización de directorios.....	81
5.3.	Iteraciones.....	82
5.3.1.	Iteración 1.....	82
5.3.2.	Iteración 2.....	87
5.3.3.	Iteración 3.....	95

5.3.4. Iteración 4.....	103
5.4. Jasmine.....	105
CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES	107
6. Conclusiones y recomendaciones	107
6.1. Conclusiones.....	107
6.2. Recomendaciones	108
BIBLIOGRAFÍA	109
GLOSARIO DE TÉRMINOS	112
ANEXOS.....	113
Anexo A. Acta de Reunión	113
Anexo C. Historia de Usuario 2.....	114
Anexo D. Historia de Usuario 3.....	115
Anexo E. Historia de Usuario 4	115
Anexo F. Historia de Usuario 5	116
Anexo G. Historia de Usuario 6.....	117
Anexo H. Historia de Usuario 7	117
Anexo I. Historia de Usuario 8	118
Anexo J. Certificado de recepción del aplicativo web	119
Anexo K. Captura de pantalla de la página de inicio de sesión	120
Anexo L. Captura de pantalla del Dashboard.....	121

CAPÍTULO I: INTRODUCCIÓN

1. INTRODUCCIÓN

1.1. Justificación

La empresa Artemisa, localizada en Quito, tiene 5 años en el mercado nacional dedicándose a la industria de la moda. Entre sus principales actividades se encuentra la producción y venta de ropa y otros productos textiles para una gran variedad de clientes y empresas. Durante su periodo de actividad comercial, la empresa no ha encontrado una forma “sencilla” de llevar el registro de sus clientes, pedidos e inventario. Las soluciones por las que ha optado la empresa, no le han resultado de utilidad, ya que la empresa busca una herramienta personalizada para su negocio, que pueda ser utilizada en cualquier lugar y que se ajuste con sus necesidades.

Las aplicaciones web se encuentran mejorando día a día, brindando la posibilidad de implementar funcionalidades complejas como lo son los chats, o ventajas frente a las aplicaciones tradicionales en términos de acceso, interacción, disponibilidad, velocidad y un mejor control en ellas (Novak, 2018). Cabe recalcar que las aplicaciones web no requieren ser programadas para un sistema operativo en específico, por lo que son más versátiles en el desarrollo. Es así que empresas como Microsoft han optado por parar el desarrollo de aplicaciones de escritorio (como Microsoft Teams en Linux), con el objetivo de desarrollarlas en la web como aplicaciones web progresivas.

Tomando en cuenta lo antes mencionado, se puede afirmar que el presente trabajo de titulación permitió desarrollar una aplicación web que solventa las necesidades del negocio y le brinda una ventaja competitiva en la industria de la moda mediante el uso de tecnología novedosa. Además, cabe mencionar que el desarrollo supuso un reto que permitió explorar nuevas experiencias, herramientas y alimentar conceptos personales.

1.2. Planteamiento del problema

Actualmente, Artemisa no cuenta con una herramienta informática que les permita llevar de forma simple un registro de sus clientes, inventario y pedidos. El hecho de no mantener un inventario adecuado ha significado pérdidas económicas que, según palabras del dueño, representan hasta el 50% de las ganancias. El no tener un control ha facilitado a sus colaboradores llevarse productos del local y ha provocado que en ciertos pedidos se tengan que fabricar más unidades.

Otro problema detectado es la pérdida de datos, mismos que en muchas ocasiones han sido borrados por error o simplemente han desaparecido al perder la infraestructura de la empresa. Esto les ha complicado la administración y el cumplimiento de sus pedidos, llegando a un servicio que ha llegado a ser ineficiente en pocas ocasiones. Los pedidos no se realizan únicamente en el local físico, también se hacen en el domicilio del cliente, por lo que necesitan llevar un cuaderno para tomarlos y después, registrarlos en la computadora de la empresa. El contar con un almacenamiento en la nube, permite mantener los registros sin importar si se pierde el hardware.

Al no contar con una solución tecnológica, se les dificulta mucho el hecho de querer contactarse o comunicarse con sus clientes, ya que deben buscar en sus registros manualmente, perdiendo una gran cantidad de tiempo y dejando de ser eficientes. También han buscado diversas formas de obtener los datos de sus clientes para realizar publicidad o un seguimiento de sus pedidos, pero cada vez que lo requieren, tienen que buscar primero sus clientes y luego los pedidos que han realizado.

Sobre la base de lo presentado, se plantea la siguiente pregunta principal:

- ¿Cómo se puede automatizar los procesos administrativos de clientes, de inventario y de pedidos de la empresa Artemisa?

Y las preguntas secundarias:

- ¿Cuáles son los requerimientos funcionales y no funcionales para el desarrollo de este sistema?
- ¿Qué tipo de metodología se adapta al desarrollo del sistema?
- ¿Qué tipo de interfaz debe tener este sistema y desde qué dispositivos se podrá acceder al mismo?

1.3. Objetivo General

Diseñar y desarrollar una aplicación web que permita administrar clientes, inventario y pedidos.

1.4. Objetivos Específicos

- Realizar el levantamiento requerimientos funcionales y no funcionales correspondientes al desarrollo de la aplicación web.
- Identificar la metodología que se adapte de mejor forma en el desarrollo de la herramienta informática.
- Diseñar y desarrollar una interfaz web y móvil que permita la conexión desde diversos dispositivos.

1.5. Antecedentes

La empresa Artemisa cuyo dueño es Adrián Calvopiña se fundó en el año 2017 en la ciudad de Quito, Ecuador. El señor Calvopiña ya se encontraba al tanto del funcionamiento de una empresa que se enfoca en la industria textil debido a que antes de fundarla, él se dedicaba a la producción y venta de uniformes para el Colegio Militar “Eloy Alfaro”. Esto le permitió conocer proveedores y el proceso principal en la fabricación de uniformes, pero por decisión propia decidió fundar su propia empresa.

Artemisa inició sus actividades vendiendo uniformes de colegios, y gracias a su progreso, han ampliado su catálogo agregando sublimados y material publicitario para campañas políticas y de empresas, como Movistar. La empresa tiene como mercado objetivo el consumidor ecuatoriano que busca productos textiles con precios competitivos. Durante su periodo de actividad, siempre han buscado establecer precios competitivos para sobresalir en el mercado y ganar clientes recurrentes.

En cuanto a la misión y visión de la empresa, Adrián (2022) afirma que “nuestra misión es llevar productos textiles a precios cómodos a todo el territorio ecuatoriano para promover el producto nacional”. Además, Adrián (2022) afirma que “nuestra visión es afianzarnos como una empresa ecuatoriana con productos asequibles y de calidad para el consumidor”.

La empresa Artemisa almacenaba todos sus registros en hojas de Excel y en documentos de forma física. Esta manera de guardar sus datos desemboca en un gran inconveniente para la empresa, ya que sus registros se encuentran estrictamente vinculados con el correcto funcionamiento de su equipo central y también en la ética de su personal. Cabe mencionar que a la empresa se le complica mucho el realizar una búsqueda de clientes y de sus pedidos, afectando en gran parte a la eficiencia de sus trabajadores, ya que, en la mayoría de los casos, entorpece el cumplimiento de sus órdenes. Basándonos en estos antecedentes, se identifica que la empresa Artemisa tiene la necesidad de automatizar sus procesos. Para destacar esta necesidad, en los siguientes párrafos se explorarán casos en los que la automatización de procesos ha brindado resultados positivos para los objetivos propuestos.

En el trabajo Importancia del uso de sistemas de información en la automatización de historiales clínicos, una revisión sistemática se presenta como resultados que la automatización de procesos realizados influyó positivamente en el funcionamiento, la eficiencia y la calidad de los procesos del sistema de salud (Preciado Rodríguez et al., 2021). Es decir, la automatización

de procesos puede aportar en cualquier área, ayudando a que los proyectos y las empresas puedan cumplir con sus objetivos de mejor forma.

Tomando en cuenta el trabajo en el cual se analizó la implementación de un sistema automatizado en la atención en la Municipalidad Provincial del Callao, se concluyó que la automatización de procesos permite que las diferentes organizaciones mejoren su funcionamiento, eficiencia y les brinda una ventaja competitiva en su campo (Farfán Jiménez, 2020). En este caso particular se puede constatar que un sistema automatizado es una herramienta necesaria que mejora el rendimiento de las empresas y organizaciones, brindando ventajas y facilidades a sus involucrados. Estas ventajas no son únicamente operacionales y funcionales, sino que también son económicas.

Sobre la base de los trabajos citados anteriormente, se evidencia que, en cualquier tipo de empresa la automatización de procesos es necesaria mejorar su eficiencia, efectividad y competitividad frente a otras en su área. Además, la automatización permite mejorar la relación con el cliente debido a que, al disminuir los tiempos en la ejecución de los procesos, y aumentar el rendimiento, este se encontrará en un ambiente más cómodo al solicitar un servicio. También es importante mencionar que, para lograr aportar positivamente, se requiere el uso de tecnología innovadora y de punta.

1.6. Alcance

El trabajo de titulación se considerará finalizado cuando se obtenga la carta de aceptación del cliente y se cumplan los siguientes módulos:

- **Administración de inventario:** Manejo de productos y stock.
- **Administración de clientes:** Control de los clientes, se podrán realizar consultas mediante las categorías de los clientes.
- **Administración de categorías:** Se podrán crear, modificar, consultar y eliminar categorías que podrán ser asignadas a los clientes.

- **Administración de pedidos:** Creación, modificación y visualización de pedidos. Estos pueden tener tres estados: pendientes, completados o cancelados.
- **Dashboard:** Visualización de un resumen general de los pedidos en curso y completados.
- **Facturación:** Emisión de notas de ventas que complementen el sistema de facturación que poseen actualmente.
- **Notificaciones mediante WhatsApp:** Alertas para cuando el stock de los productos sea bajo.
- **Roles de usuarios:**
 - **Administrador:** Acceso total a todo el sistema, podrá acceder al dashboard y a la administración de inventario, clientes, categorías y pedidos.
 - **Empleado:** Acceso a los pedidos e inventario.

CAPÍTULO II: FUNDAMENTACIÓN TEÓRICA

2. Marco Teórico

2.1. ¿Qué es un proceso y cuáles son los beneficios de automatizarlos?

Se denomina proceso a un conjunto de pasos o acciones que se ejecutan con el objetivo de lograr un fin establecido. El automatizarlos nos permite minimizar los costes de operación, errores humanos o de comunicación, reducir los tiempos de ejecución y tener un mejor control de sus pasos (Sandoval Ruiz, 2019). Es decir, el automatizar un proceso nos brinda una gran cantidad de beneficios que varían dependiendo del tipo de negocio. Entre sus ventajas podemos encontrar el aumento de la eficiencia y efectividad en la ejecución del proceso, y también la minimización de costes operativos.

2.2. Aplicaciones Web

En la actualidad, las aplicaciones web son una pieza clave para el mundo digital ya que permiten que las empresas crezcan y logren sus objetivos más rápido (Shetty et al., 2020). Las aplicaciones web se han vuelto populares debido a que solo requieren que el usuario tenga instalado un navegador para poder acceder a ellas. Esta es una característica clave porque permite que cualquier usuario pueda acceder desde una gran variedad de dispositivos inteligentes, sin la necesidad de tener que instalar la aplicación o requerir una gran cantidad de recursos.

2.3. Páginas web adaptable

Este tipo de páginas web se caracteriza principalmente por adaptar su contenido y estructura al dispositivo desde el cual se acceda (Vega & Fuertes, 2020). Los diseños adaptables solventan la necesidad del desarrollo orientado a múltiples terminales mediante la inclusión de estilos que permiten que el diseño se vaya adaptando con base en las dimensiones y resoluciones de las

pantallas. Es así que el objetivo principal que tienen las páginas web adaptables es verse correctamente en todos los dispositivos (computadoras, teléfonos y tablets).

Para hacer uso del diseño responsivo, es necesario importar etiquetas que indiquen que nuestra página será visualizada en todo tipo de dispositivo, y el ancho deberá depender de la pantalla de este. Luego, mediante el uso de propiedades en CSS, dimensiones en porcentajes o ir limitándolas a un ancho y largo máximo dependiendo de la resolución del dispositivo. No es una tarea sencilla, puesto que requiere de experiencia y conocimiento de muchas propiedades de CSS y el uso de HTML, pero con la práctica todo es posible. La tarea más complicada en la incrustación de estilos para que nuestra página se vuelva adaptable es la definición de una estructura que nos sirva como base para el desarrollo. Pero esto puede ser solventado utilizando marcos de trabajo como lo son Bootstrap o W3.CSS, los cuales ya incorporan en sus bibliotecas una gran variedad de plantillas para ser recicladas o rediseñadas.

2.4. Arquitecturas de Software

En el mundo del internet elegir una arquitectura que se ajuste al desarrollo de aplicaciones es una tarea difícil puesto a que existe una gran variedad de tecnologías disponibles (Shetty et al., 2020). Una arquitectura de software proporciona modelos y estándares que permiten identificar cómo va a funcionar y cómo será la interacción de nuestra aplicación. Además, nos brindan una idea general del proyecto para saber qué necesitaremos y cómo desarrollarlo. En la selección de la arquitectura de software, debemos tener en cuenta la forma en la que se repartirá la carga de cómputo, la disponibilidad y usabilidad.

2.4.1. Tres Capas

La arquitectura de tres capas establece que la aplicación debe ser separada en tres: el nivel de presentación, el nivel de aplicación y el nivel de datos. Cada una de sus capas tiene una infraestructura propia, permitiendo que se distribuya de mejor forma la carga. Al encontrarse

separada, cada uno de sus niveles actúa de forma independiente, pero se complementan entre sí. Su división nos permite desarrollar todas sus capas a la vez y escalarlas de forma individual dependiendo de los requerimientos y el uso.

A continuación, una breve explicación de los niveles de la arquitectura:

- **Nivel de presentación:** Corresponde a la interfaz del usuario, permitiendo la comunicación y la interacción con el mismo. En el caso de aplicaciones web, se ejecuta desde un navegador y se encuentra desarrollada en HTML y CSS.
- **Nivel de aplicación:** Es el nivel lógico donde se ejecuta todo el giro del negocio. Aquí se encuentra el procesamiento y la funcionalidad de la aplicación.
- **Nivel de datos:** Es el nivel donde se guarda y administra toda la información procesada en el nivel de aplicación.

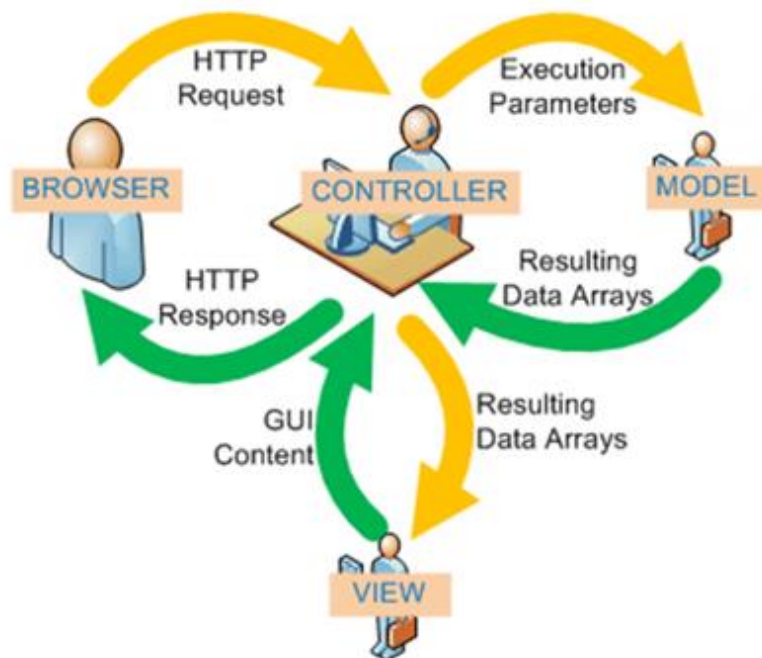
2.4.2. Modelo-Vista-Controlador (MVC)

La arquitectura modelo-vista-controlador (MVC) se caracteriza principalmente por ser utilizada para implementar interfaces de usuario, datos y lógica de control (Mozilla, 2022). Como su nombre ya lo indica, las capas en las que se divide este patrón son las siguientes:

- **Modelo:** Aquí se encuentra toda la lógica del negocio y los datos.
- **Vista:** Parte que verá el usuario, es decir, es la interfaz del usuario. Aquí el usuario podrá interactuar, ya sea completando formularios o visualizando resultados.
- **Controlador:** Se encarga de manejar todas las instrucciones o llamadas que recibe del modelo y de la vista, ya sea procesando los datos y enviando resultados.

Figura 1

Flujo del Modelo-Vista-Controlador



Nota. La figura muestra el flujo de interacción en el MVC. El usuario es el que interactúa con la interfaz, el controlador recibe la solicitud y luego procede a acceder al modelo realizando la instrucción. Finalmente, el controlador envía los resultados a la vista y la presenta al usuario. Tomado de Flujo MVC, por Universidad de Alicante, 2022, UA (<https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html>).

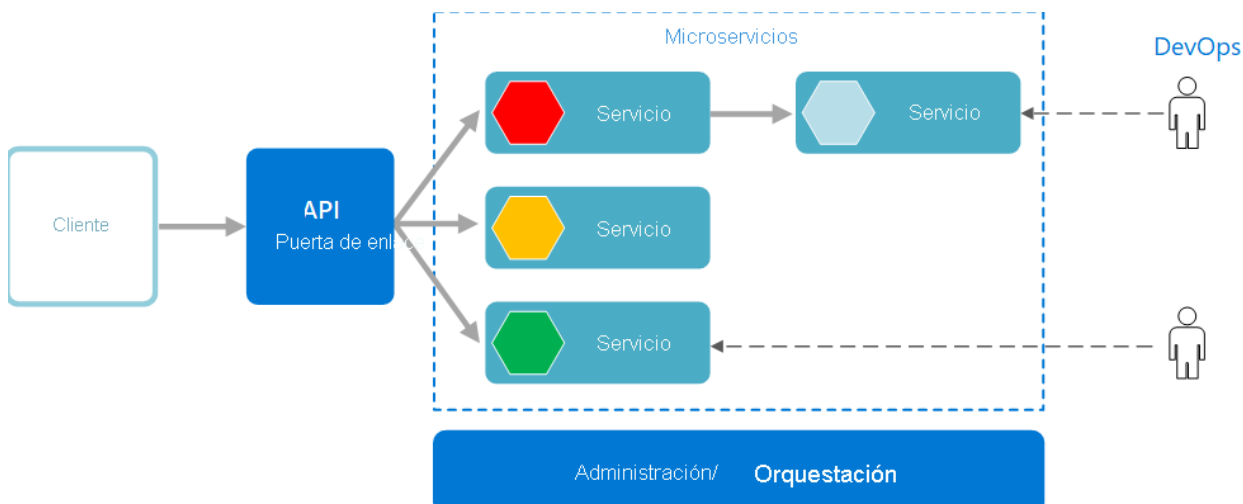
2.4.3. Microservicios

La arquitectura de microservicios, como su nombre ya lo indica, se caracteriza por tener una colección de varios servicios que funcionan de forma autónoma (Microsoft, 2022). Su ventaja principal radica en que cada uno de los servicios es integrado y no importa el lenguaje de programación con el que se encuentren desarrollados, ya que son independientes. Su comunicación se realiza mediante API establecidas y por lo general llevan una autenticación para evitar ser vulneradas o atacadas. También cabe recalcar que, al ser independientes, pueden ser

actualizados sin afectar el funcionamiento de la aplicación que los consuma, siempre y cuando se mantenga la misma estructura de respuesta.

Figura 2

Flujo de la arquitectura de microservicios



Nota. La figura muestra el flujo de interacción en la arquitectura de microservicios. El cliente hace una llamada a las API y estas se encargan de consumir cada servicio. Tomado de Estilo de Arquitectura de Microservicios, por Microsoft, 2022, Microsoft Learn (<https://learn.microsoft.com/es-es/azure/architecture/guide/architecture-styles/microservices>).

2.5. Frameworks

Un marco de trabajo o framework es una biblioteca y clases que proporcionan una gran variedad de funcionalidades con el objetivo de que los requisitos fundamentales para las aplicaciones ya se encuentren configurados (Shetty et al., 2020). El uso de marcos de trabajos facilita el desarrollo de una aplicación, ya que en su estructura incluye una rica variedad de librerías que incluyen funcionalidades que podemos reciclar. Gracias a ello, el desarrollo de aplicaciones se puede realizar en un tiempo mucho más corto y con una estructura optimizada.

2.5.1. Frameworks Front-end

Se denomina Front-end a la parte de una aplicación web en la que interactúan los usuarios, aquí se incluye todos los elementos como imágenes, botones, textos y mucho más (Shetty et al., 2020). El Front-end corresponde al diseño que ve al usuario y a su interacción, se caracteriza principalmente por cargar en el dispositivo del usuario y consumir sus recursos. Esta parte de la aplicación suele ser desarrollada en los lenguajes HTML, CSS y JavaScript, permitiendo darle estilo a la página y también añadir funcionalidades y validaciones.

Entre los Frameworks más populares para Front-end en el 2022 podemos encontrar a Angular de Google y React de Meta ya que cuentan con un gran soporte por parte de la comunidad y facilitan muchas de las tareas que van desde el enrutamiento de las páginas hasta el desarrollo de componentes. También, es importante mencionar que, por lo general, al trabajar con Frameworks de Front-end, se suele utilizar una gran variedad de librerías o frameworks dentro de los mismos, como puede ser el caso de Bootstrap que es una biblioteca multiplataforma que proporciona plantillas de diseño ya predefinidas y pueden ser modificadas sin ningún problema.

2.5.2. Frameworks Back-end

El Back-end como su nombre lo indica, es la parte trasera de la aplicación en donde se ejecutan la mayoría de las conexiones, ya sea con la base de datos o con otras API para su consumo. En su creación se encuentran innumerables actividades, ya que debemos proteger las API contra ataques, autenticar usuarios, gestionar peticiones y enviar respuestas de forma correcta (Shetty et al., 2020). Para el desarrollo del Back-end, existen muchos frameworks de código abierto que podemos encontrar navegando por internet, pero es de suma importancia elegir uno correctamente ya que aquí se realizarán la mayor parte de solicitudes por parte del usuario y el consumo de la base de datos en base a sus solicitudes. Entre los principales factores a tener en cuenta en la selección podemos encontrar: nuestro conocimiento del lenguaje de programación que este maneja,

2.6. Sistema gestor de bases de datos (SGBD)

Para entender el concepto de un gestor de base de datos, primero abordaremos el concepto principal de una base de datos. Se denomina base de datos a toda herramienta que permite la recopilación y organización de información, estas suelen surgir a partir de listas en hojas de cálculo o simplemente programas de texto (Microsoft, s.f.). En este proyecto, una base de datos permite que todos los datos que se generen en el aplicativo web se guarden para que luego puedan ser utilizados en consultas que faciliten los objetivos del negocio.

Se conoce como sistema gestor de base de datos a los programas que se presentan al usuario final para que pueda manejar el acceso y la información que se encuentra en una base de datos. Es el componente más importante en un sistema de base de datos, sin él, los usuarios no podrían introducir, consultar o interactuar con los datos (Ionos, 2020). Existen varios SGBD pero en su elección será importante determinar cuál será el modelo de base de datos con el que trabajaremos. Entre los principales modelos encontramos los siguientes:

- **Bases de datos relacionales:** Ideal para bases estructuradas, se divide en tablas, que poseen columnas y filas. Mantienen su relación mediante claves primarias y foráneas. Su lenguaje de dominio es el SQL.
- **Bases de datos no relacionales:** Sus datos son semiestructurados o no estructurados, no poseen un lenguaje de dominio y tampoco estándares, por lo cual, su uso suele variar dependiendo del SGBD.
- **Bases de datos orientadas a objetos:** Como su nombre lo indica, su información es representada mediante objetos.

2.7. Plataforma de WhatsApp para negocios

La plataforma de WhatsApp empresarial ofrece la capacidad de conectarse con sus clientes y personal a las medianas y grandes empresas (WhatsApp, 2022). Es una herramienta sumamente útil ya que nos permite enviar alertas, iniciar conversaciones y brindar un servicio

personalizado de forma simple e inmediata. Su costo varía dependiendo de las necesidades y del tamaño de las empresas, pero cuenta con un plan básico que permite realizar 1000 conversaciones cada mes, resultando ser una gran oportunidad para los negocios pequeños, permitiendo que hagan uso de ella sin necesidad de actualizar su plan a uno de pago.

3. Herramientas y metodología para el desarrollo de la aplicación

3.1. Comparación y selección de la metodología para el desarrollo de la aplicación

Una metodología de software nos proporciona un conjunto de técnicas y pasos que nos ayudan en el desarrollo de software. Para seleccionar una metodología de desarrollo, debemos tener en cuenta varios factores importantes para su construcción, primero tenemos el tiempo y luego el número de miembros que participaran en el desarrollo. Dependiendo de estos parámetros, el desarrollo podrá tomar una metodología ágil, tradicional o híbrida. En el caso del presente proyecto, no se contaba con una gran cantidad de tiempo, por lo cual, se otorgó prioridad a las metodologías ágiles.

3.1.1. Programación Extrema (XP)

La programación extrema es una metodología ágil utilizada para el desarrollo de proyectos cortos que se encuentran centrados en la velocidad y simplicidad de los ciclos (Raeburn, 2022). Esta metodología al igual que otras ágiles trabaja con procesos iterativos para centrarse en un desarrollo rápido y con una buena comunicación con el cliente.

XP posee los siguientes 5 valores:

- **Comunicación:** XP posee una comunicación rápida tanto con el equipo de trabajo, como con el cliente, siempre buscando sugerencias y aportes de los demás.
- **Simplicidad:** Se busca el proceso más simple y efectivo en el desarrollo.
- **Retroalimentación:** Considera importante los comentarios de los demás, para producir un trabajo con mejores resultados. Se complementa con la comunicación ya que todo debe ser inmediato.

- **Valor o Coraje:** Es importante aceptar la responsabilidad que se tiene durante el desarrollo.
- **Respeto:** XP motiva a todos a comunicarse con respeto con los miembros del equipo y con el cliente, para mantener un ambiente amigable en el desarrollo.

Por otro lado, la programación extrema posee 5 etapas fundamentales en su desarrollo:

- **Planificación:** Se determina que los requerimientos por parte del cliente son viables, se comienza a asignar la carga de trabajo al equipo, se planifica y se establecen plazos de entrega.
- **Diseño:** Se realiza un diseño simple y por lo general se utilizan tarjetas de clase, responsabilidad y colaboración (CRC). Permiten que el equipo aprecie los objetos, también sepa quién está a cargo de cada parte.
- **Codificación:** Como su nombre lo indica, aquí se realiza la programación utilizando estándares y trabajando en conjunto.
- **Pruebas:** Se realizan pruebas unitarias y de integración continua. Luego se realiza un proceso iterativo (se vuelve a la planificación) para ir dándole forma al proyecto en conjunto con el cliente.
- **Lanzamiento:** Se realiza la implantación del proyecto.

La integración de sus valores con las etapas de desarrollo convierte a XP en una metodología ideal para el desarrollo de proyectos cortos, obteniendo resultados de forma rápida que cumplan con las expectativas del cliente.

3.1.2. Scrum

Scrum es una metodología ágil que permite a los equipos trabajar en conjunto abordando problemas complejos a la vez que se mantiene una comunicación con el cliente (Martins, Qué es Scrum y cómo aplicarlo en gestión de proyectos, 2022). Permite desarrollar proyectos bien

estructurados ya que se maneja mediante iteraciones y reuniones cortas para mejorar el trabajo de forma continua.

La metodología Scrum cuenta con tres pilares fundamentales:

- **Artefactos**

- **Product Backlog:** Hace referencia a todas las tareas que todavía no se han terminado.
- **Sprint Backlog:** Es el trabajo que se tiene que desarrollar en los sprint, pero todavía no se ha terminado.
- **Incremento del producto:** Es lo que se presenta al final de cada sprint en el sprint review.

- **Eventos**

- **Sprint:** Son iteraciones sucesivas en las cuales se establecen objetivos para ser presentados en las siguientes reuniones.
- **Sprint Planning:** Aquí se definen los objetivos del Sprint y se establecen los trabajos que serán desarrollados del producto backlog.
- **Objetivo del Sprint:** Se determina que se logrará con el desarrollo de las tareas.
- **Scrums:** Son reuniones cortas que pueden ser diarias o semanales. Sirven para entender en dónde nos encontramos y si existe algún inconveniente.
- **Revisión de Sprints:** Se presenta y se revisa el trabajo de todo el equipo. Aquí se cambian los estados de las tareas.
- **Retrospectiva de Sprint:** Se realiza una retroalimentación del sprint para ver cómo podemos mejorar.

- **Roles**

- **Responsable del producto:** Es el encargado de comunicarse con el cliente, transmite sus opiniones y lleva el producto backlog.

- **Scrum Máster:** Se encarga de manejar y llevar los eventos de Scrum.
- **Equipo de Scrum:** Son los participantes en el proyecto, es decir los desarrolladores.

3.1.3. Kanban

Kanban es conocida por ser una metodología que colabora a los equipos en encontrar un equilibrio entre lo que deben hacer y la disponibilidad de los miembros (Martins, 2022). Su nombre viene de la palabra japonesa Kanban, que significa tarjeta. Permite visualizar y gestionar todos los trabajos que se tengan que desarrollar mediante un tablero. Otorgando una gran visión general y facilitando el desarrollo de proyectos cortos. Además, Kanban cuenta con cuatro principios básicos los cuales son los siguientes:

- Principios de gestión del cambio:
 - Comenzar con lo que hacemos ahora.
 - Aceptar retroalimentaciones incrementales y progresivas.
 - Fomentar el liderazgo.
- Principios de prestación de servicios:
 - Centrarse en las necesidades del cliente.
 - Administrar el trabajo, no los trabajadores.
 - Revisar regularmente el tablero.

Los principios mencionados nos permiten fomentar el trabajo teniendo una vista macro de lo que queremos realizar. Esto es realmente importante ya que al ser una metodología un poco más simple que otras, es posible desarrollar sin tener tanta documentación, pero conservando aspectos prácticos. Ahora veremos las prácticas de la metodología Kanban:

- **Visualizar el trabajo:** En Kanban se utilizan tarjetas, por lo cual, podemos ir visualizando en que parte del tablero se encuentra cada una de estas. Es esencial para determinar

que tareas se encuentran atascadas y cuales requieren un esfuerzo extra para ser terminadas.

- **Limitar el trabajo en progreso:** Es necesario establecer un límite de tareas en curso para evitar quedarnos atrapados en un cuello de botella. Permitirá que vayamos completando nuestros trabajos en proceso sin inconvenientes.
- **Gestionar el flujo de trabajo:** Se complementa con los principios anteriores, ya que resulta de suma importancia identificar el estado de cada tarea para lograr terminar el desarrollo en lo planificado.
- **Implementar políticas:** Es normal en trabajos en equipo, siempre tenemos que incentivar a todos a participar.
- **Implementar comentarios:** Kanban le da prioridad a la opinión y retroalimentación del equipo y del cliente.
- **Mejorar con retroalimentaciones y experimentando:** Se buscan formas de mejorar continuamente, tomando en cuenta las retroalimentaciones colectivas y la experiencia generada en el desarrollo

Tabla 1

Puntaje de metodologías de desarrollo

	Programación Extrema	Scrum	Kanban
Conocimiento	6	10	7
Eficiencia en proyectos cortos	10	10	9
Facilidad	8	7	9
Flexibilidad	10	10	10
Simplicidad	10	8	10

Tiempo de desarrollo	10	10	10
Validaciones	10	8	7
Trabajo individual	9	6	10
Total	73	69	72

Nota. La tabla muestra la evaluación personal con respecto a cada metodología de desarrollo.

Elaboración propia.

Tomando en cuenta la tabla comparativa presentada, se seleccionó XP como metodología de desarrollo ya que obtuvo una puntuación de 73, seguida se encuentra Kanban con un total de 72 y en último lugar scrum con 69 puntos.

3.2. Comparación y selección de los Frameworks para el desarrollo web

3.2.1. Comparación y selección del Framework o servicio para el Back-end

3.2.1.1. Laravel

Laravel es un framework enfocado a el desarrollo web, con estructuras limpias y códigos simples y elegantes (Laravel, s.f.). Puede ser utilizado tanto para el desarrollo Full-stack, pero dado que existen mejores marcos de trabajo en el Front-end, el uso que se le suele dar a Laravel es en el Back-end. En su estructura encontramos herramientas útiles para la interacción con bases de datos mediante su asignación objeto-relacional “Eloquent”, que permite realizar las funciones de un CRUD de forma inmediata y sin complicaciones. También posee una autenticación que brinda una capa extra de seguridad a nuestro Back-end, mediante estándares de seguridad para la autorización, como lo es el OAuth 2.0.

El lenguaje de programación de Laravel es PHP, y requiere de “Composer” para manejar sus dependencias. El hecho de que se encuentre desarrollado en PHP es una ventaja frente a otros

marcos de trabajos, ya que, gracias al apoyo de la comunidad, existe una gran variedad de librerías escritas en este lenguaje que pueden ser implementadas mediante comandos. También cabe destacar que, al ser un lenguaje reconocido y utilizado por grandes empresas, como por ejemplo Meta, Disney y Twitch, Laravel recibe constantes soportes en cuanto a integración de servicios y también cuenta con muchos tutoriales en internet.

3.2.1.2. Django

Django es un marco de trabajo que permite el desarrollo rápido, sencillo y escalable de sitios web (Mozilla, 2022). Su lenguaje de programación es Python y al igual que Laravel, también puede ser utilizado para el desarrollo del Front-end. Pero a diferencia de este, Django utiliza una arquitectura modelo-plantilla-vista, que es muy similar al modelo-vista-controlador, pero en este caso, la vista para a ser la plantilla y el controlador la vista.

Además, Django también proporciona autenticaciones, formularios y validaciones, facilitando la interacción con el usuario mediante su plantilla y realizando un procesamiento en la vista únicamente cuando se valide de forma correcta cada uno de los campos. Debido a que debemos utilizar mapas de URL para asignar funciones específicas a las llamadas, la programación de API en este framework se vuelve más complicada que Laravel.

3.2.1.3. Firebase

Firebase es una plataforma de Google orientada al desarrollo de aplicativos Backend-como-Servicio contando con una serie de herramientas como una base de datos en tiempo real, almacenamiento en la nube, autenticación y otros (Flutter, 2022). La plataforma Firebase nos permite desarrollar y escalar con base en las necesidades de nuestra empresa y las solicitudes de los usuarios. Su integración se da mediante la instalación de librerías en el Front-End, para la parte del cliente, o también se lo puede hacer en un Back-end para obtener permisos de

administrador. Es por ello por lo que, el lenguaje variará dependiendo de la tecnología que utilicemos, y para comunicarnos con este servicio lo haremos mediante métodos ya establecidos.

Tabla 2

Planes de precios de Firebase

Productos	Sin Costo (Plan Spark)	Prepago (Plan Blaze)
Autenticación		
Autenticación telefónica: EE. UU., India y Canadá	10,000 por mes	\$0.01 por verificación
Autenticación telefónica: Todos los demás países	10,000 por mes	\$0.06 por verificación.
Plataforma de Identidad	50k/mes	Sin costo hasta 50k, luego aplica una tarifa mínima dependiendo del tráfico.
Cloud Firestore		
Datos almacenados	1 GiB en total	Sin costo hasta 1 GiB en total Luego, pagas \$0.108 por GiB adicional
Salida de red	10 GiB por mes	Sin costo hasta 10 GiB por mes. Luego se aplican tarifas dependiendo del tráfico mensual.
Operaciones de escritura de documentos	20,000 operaciones de escritura por día	Sin costo hasta 20,000 operaciones de escrituras. Luego se aplican tarifas

		dependiendo del tráfico mensual.
Operaciones de lectura de documentos	50,000 operaciones de lectura por día	Sin costo hasta 50,000 operaciones de lectura. Luego se aplican tarifas dependiendo del tráfico mensual.
Eliminaciones de documentos	20,000 eliminaciones por día	Sin costo hasta 20,000 operaciones de eliminación. Luego se aplican tarifas dependiendo del tráfico mensual.
Almacenamiento en la nube		
GB almacenados	5 GB	\$0.026 por GB
GB descargados	1 GB por día	\$0.12 por GB
Operaciones de carga	20,000 por día	\$0.05 por cada 10,000
Operaciones de descarga	50,000 por día	\$0.004 por cada 10,000

Nota. Recuperado de Planes de precios, por Firebase 2022, Firebase Pricing (<https://firebase.google.com/pricing?hl=es>).

Si observamos la tabla anteriormente mostrada, podemos notar que el plan Spark de Firebase, otorga una gran cantidad de beneficios que pueden ser utilizados de forma gratuita, obviamente, dependiendo de la empresa. En este caso, Artemisa cuenta con 9 empleados, por lo que, utilizar este plan, permitiría a Artemisa consumir este servicio en la nube sin la necesidad de pagar hasta que crezca.

Tabla 3*Puntuación de Frameworks o servicios para Backend*

	Laravel	Django	Firestore
Conocimiento del Framework o servicio	9	5	9
Conocimiento del Lenguaje de Programación	10	8	10
Comunidad	10	9	9
Estandarización	10	10	10
Flexibilidad	8	7	9
Costo	6	6	10
Documentación	10	10	9
Facilidad de uso	10	8	9
Curva de aprendizaje	9	7	10
Actualización	10	10	10
Seguridad	10	10	10
Total	102	90	105

Nota. La tabla muestra la evaluación personal con respecto a cada framework o servicio para back-end. Elaboración propia.

Tomando en cuenta la tabla comparativa presentada, se seleccionó Firestore como servicio de Back-end para el desarrollo del aplicativo con un total de 105 puntos. En segundo lugar, tenemos a Laravel con 102 puntos y finalmente Django con 90. La ventaja que sobresale frente a las demás es el costo, ya que Firestore puede ser utilizado por el momento sin necesidad de pagar por un hosting y además, puede ser implementada utilizando métodos de sus librerías.

3.2.2. Comparación y selección del Framework para el Front-end

3.2.2.1. Angular

Se conoce como Angular al framework de diseño y desarrollo de aplicaciones que se encuentra orientado a páginas eficientes y sofisticadas (Angular, 2022). Los lenguajes de programación de Angular son TypeScript, AngularJS, HTML, SCSS y CSS. Mediante la integración de ellos podemos generar páginas veloces, optimizadas y seguras. Angular utiliza la arquitectura MVC, integrando llamadas a API mediante servicios, y consumiéndolas mediante componentes que pueden ser reciclados. A diferencia de muchos otros marcos de trabajo de Front-end, Angular no se encuentra optimizado para los posicionamientos en buscadores debido a que toda la aplicación se ejecuta en una simple página, y mediante enrutamientos, va cambiando los componentes.

Una de las grandes ventajas de Angular es el uso de guardianes que se activan mediante llamadas y nos permiten realizar validaciones antes de que carguen los componentes. También cabe recalcar, que su mayor uso se da en aplicaciones web que son interactivas, permitiéndonos llegar a hacer cambios en tiempo real sin la necesidad de programar instrucciones complejas en JavaScript para realizarlos. Al ser un Framework de código abierto, su documentación es amplia y recibe un gran apoyo por parte de la comunidad, lo cual lo convierte en una de las principales opciones para el desarrollo.

3.2.2.2. React

React es una librería de JavaScript que se utiliza principalmente para construir interfaces simples e interactivas (React, 2022). Al igual que Angular, React se basa en crear y reciclar componentes que pueden ir cambiando en tiempo real. Su curva de aprendizaje es más sencilla que Angular y Vue, pero React se encuentra enfocado más a proyectos complejos. Su lenguaje de programación es JavaScript, y al igual que Angular, debe ejecutarse con NodeJS.

3.2.2.3. Vue.js

Vue es un ramework progresivo incremental que se utiliza en la construcción de interfaces de usuarios, enfocándose únicamente en la capa de visualización (Vue.js, 2022). Puede ser utilizado para desarrollar aplicaciones de una sola página web al igual que Angular, o simplemente utilizarse de forma tradicional con múltiples páginas. La curva de aprendizaje de Vue es fácil, y posee una amplia comunidad y soporte. La forma de trabajar con formularios y validaciones es mucho más compleja que con Angular y React, pero su actualización en tiempo real resulta más simple.

Se encuentra orientado a empresas de todos los tamaños y al desarrollo de todo tipo de proyectos ya que su rendimiento es mayor al de Angular y React. Además, Vue se programa en Javascript y TypeScript, por lo que no tiene un lenguaje propio, pero si incluye una serie de atributos que son agregados al código HTML para realizar interacciones con el usuario.

Tabla 4

Puntuaciones de Frameworks para el Frontend

	Angular	React	Vue.js
Conocimiento del Framework	10	6	5
Conocimiento del Lenguaje de Programación	10	10	10
Comunidad	10	9	9
Flexibilidad	8	7	9
Documentación	10	10	9
Facilidad de uso	10	8	7
Curva de aprendizaje	9	10	6

Actualización	10	10	9
Seguridad	10	10	8
Total	87	80	72

Nota. La tabla muestra la evaluación personal con respecto a cada framework para front-end. Elaboración propia.

Como se puede observar en la tabla anterior, la calificación más alta la tiene Angular con un total de 87 puntos, seguido de React con 80 y Vue.js con 72 puntos. Es por ello por lo que el Framework de Front-end que se utilizó en el desarrollo es Angular.

3.3. Comparación y selección del sistema de gestión de base de datos para el desarrollo de la aplicación

3.3.1. MySQL

MySQL es un sistema de gestión de bases de datos relacionales de código abierto y rico en funciones y características similares a otros SGBD como PostgreSQL (León Soberón, 2020). Se encuentra desarrollado por Oracle y usa tablas, columnas y filas para almacenar la información. En su estructura entran las llaves primarias para asignar un identificador a sus filas y las tablas se relacionan con otras mediante claves foráneas. Su curva de aprendizaje es sencilla, no es tan compleja, pero sí es importante que entendamos su lenguaje de dominio específico que es el lenguaje consulta estructurada (SQL).

3.3.2. MariaDB

MariaDB Server es una variante de MySQL y es conocido por ser uno de los sistemas de gestión de bases de datos relacionales más utilizados en el mundo, con usuarios como WordPress y Google (MariaDB Foundation, 2022). Su instalación es realmente simple, puede ser de forma individual o mediante distribuciones de Apache como XAMPP. Al ser un sucesor de MySQL, este sistema gestor es altamente compatible con el mismo, por lo que, las instrucciones

son muy similares y en algunos casos son las mismas. Puede implementar columnas JSON mediante un formato LONGTEXT, que permiten almacenar objetos o datos no estructurados.

Como experiencia propia, MariaDB es uno de los mejores SGBD, pero suelen existir casos en los que da problemas en la ejecución de su servidor, llegando a ser necesario realizar una serie de pasos para recuperar las bases de datos y reestablecer el servicio. Tiene una alta compatibilidad por defecto con muchos marcos de trabajo de Back-end como lo es Laravel, y sus instrucciones no son complejas.

3.3.3. Firebase Firestore

Como ya se mencionó anteriormente, Firebase Firestore es un servicio que viene integrado en el Back-end como servicio de Firebase. Lo interesante en este tipo de base de datos es que es no relacional y nos permite realizar consultas y escuchar en tiempo real. Además, Firebase nos brinda una capa de seguridad mediante reglas, las cuales deben ser establecidas en base a los documentos que ocupemos, y aquí se ocupan condicionales y variables en base a la sesión del usuario, su identificador o correo.

Firestore, al ser no relacional, se maneja mediante colecciones, documentos y campos. Lo interesante es que podemos ir creando múltiples colecciones dentro de un documento en lugar de agregar campos. Es importante mencionar que en sus campos podemos utilizar los siguientes tipos de datos: cadenas, números, booleanos, mapas, arreglos, nulos, tiempos, puntos geográficos y referencias.

Tabla 5*Puntuaciones de sistemas de gestión de base de datos*

	MySQL	MariaDB	Firestore
Conocimiento del motor	9	9	9
Escalabilidad	9	9	8
Comunidad	9	9	10
Flexibilidad	8	9	10
Documentación	9	9	10
Facilidad de uso	9	9	10
Curva de aprendizaje	9	9	10
Actualización	8	8	10
Seguridad	9	8	10
Total	79	79	87

Nota. La tabla muestra la evaluación personal con respecto a cada sistema de gestión de base de datos. Elaboración propia.

En consideración a la tabla anterior, se utilizó Firestore ya que obtuvo una puntuación total de 87, seguida de MariaDB y MySQL con un puntaje similar. El resultado resulta ventajoso ya que anteriormente en la selección de framework o servicio para el back-end, se seleccionó a Firebase, con lo cual, ya estaríamos complementando este servicio.

4. Análisis y diseño

4.1. Historias de usuario

Tomando en cuenta las historias de usuarios que se encuentran en los Anexos B al I, con la encargada del proyecto (Ing. María José Calvopiña), se las enlista de la siguiente forma:

- **Administrador**
 - Inicio de sesión con usuario y contraseña.
 - Dashboard.
 - Administración de clientes.
 - Administración de inventario.
 - Administración de categorías.
 - Administración de pedidos.
 - Administración de empleados.
 - Emisión de notas de venta que complementen el sistema de facturación que poseen actualmente.
 - Notificaciones mediante WhatsApp en función al stock de los productos.

- **Empleado**
 - Inicio de sesión con usuario y contraseña.
 - Dashboard.
 - Administración de pedidos.
 - Administración de inventario.
 - Administración de clientes.
 - Emisión de notas de venta que complementen el sistema de facturación que poseen actualmente.

4.2. Requerimientos funcionales

- F0 Ingreso al sistema.
- F1 Administrar clientes.
- F2 Administrar categorías.
- F3 Administrar inventario.
- F4 Administrar pedidos.
- F5 Administrar empleados

4.3. Requerimientos no funcionales

A continuación, se presentan los requerimientos no funcionales que se desarrollaron en el aplicativo web:

- El aplicativo debe permitir el acceso a los usuarios dependiendo de su rango, ya sea administrador o empleado.
- El aplicativo deberá validar los campos de los usuarios en tiempo real, es decir, mientras un usuario se encuentra ingresando los datos, el sistema presentará los errores en caso de que existan.
- La interfaz de usuario debe ser simple.
- En cada acción, el aplicativo deberá contar con alertas para mantener informado al usuario.
- El aplicativo debe contar con servicios que puedan ser consumidos por otros componentes.
- El aplicativo autenticará a los usuarios y dependiendo de su estado (ya sea ingresado o no), lo redireccionará a la pantalla de inicio o a iniciar sesión.

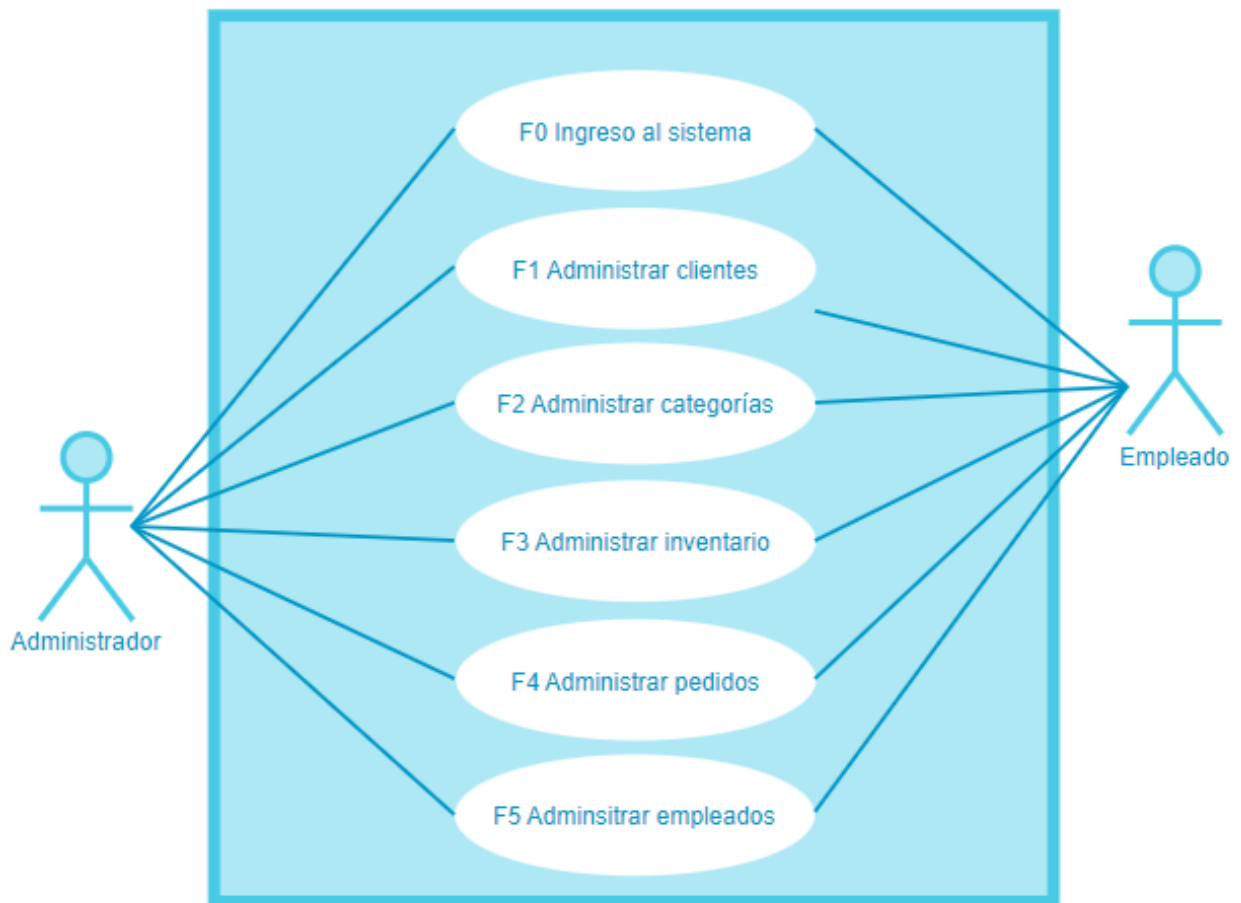
4.4. Casos de uso

A continuación, se presentarán los distintos requerimientos funcionales con sus respectivos casos de uso:

4.4.1. Nivel general

Figura 3

Caso de uso: Nivel general

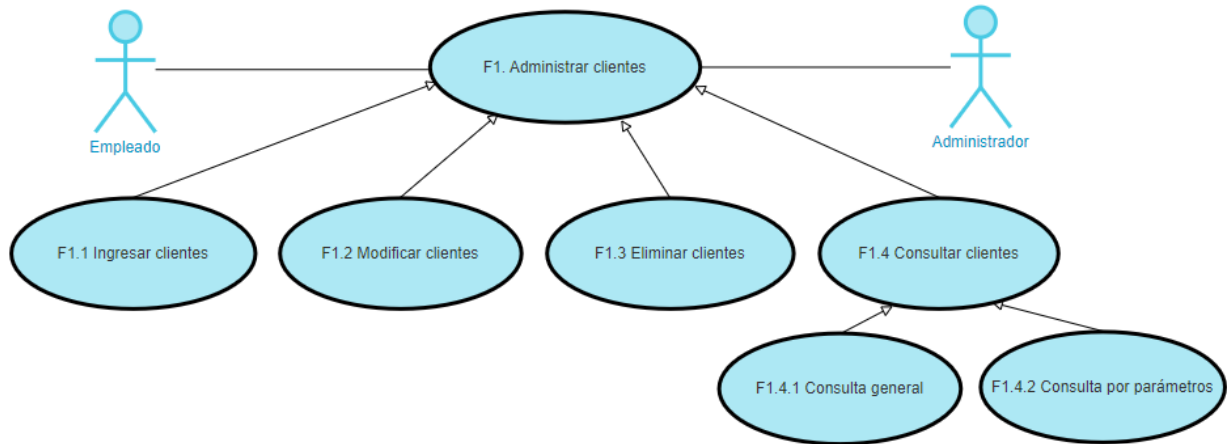


Nota. Elaboración propia.

4.4.2. F1. Administrar clientes

Figura 4

Caso de uso: Siguiente nivel, F1 Administrar clientes



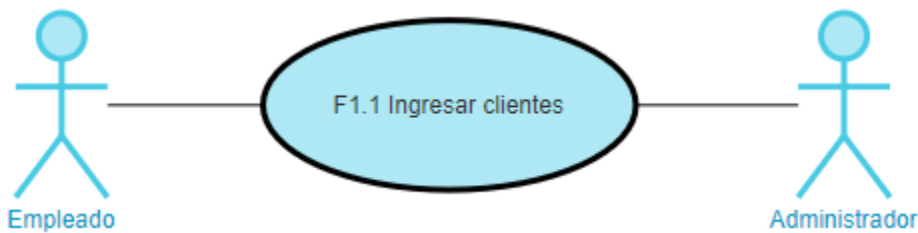
Nota. Elaboración propia.

Caso de uso: Nivel de detalle

- **F1.1 Ingresar clientes**

Figura 5

Caso de uso: Nivel detallado, F1.1 Ingresar clientes



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán ingresar un cliente al sistema.

Actores: Administrador y empleado.

Flujo principal

1. El actor selecciona la opción “Clientes” del “Menú principal”.
2. El sistema despliega las opciones de “Clientes”.
3. El actor selecciona la opción “Ingresar Cliente”.
4. Se despliega la interfaz de “Ingresar Cliente”.
5. El actor ingresa los datos del cliente.
6. El actor presiona el botón “Registrar”.
7. El sistema registra los datos del cliente en la base de datos (E1, E2).
8. El sistema muestra el mensaje de éxito o error de la acción.

Flujo alternativo

9. Si los datos del cliente ya existen, dirigirse a Modificar clientes (F1.2).

Excepciones

Tabla 6

Excepciones Caso de Uso: Ingresar clientes

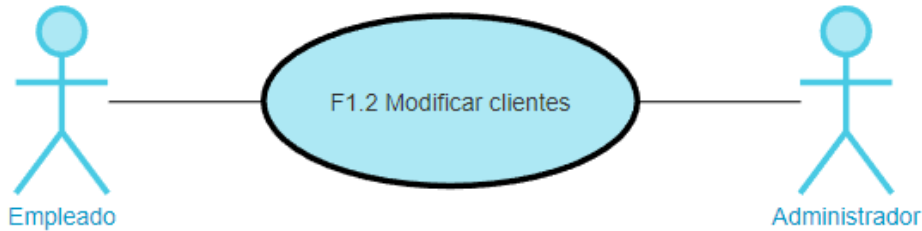
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD
E2	Cliente existente	Verificar los datos del cliente

Nota. Elaboración propia.

- **F1.2** Modificar clientes

Figura 6

Caso de uso: Nivel detallado, F1.2 Modificar clientes



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán modificar un cliente del sistema.

Actores: Administrador y empleado.

Flujo principal

1. El actor selecciona la opción "Clientes" del "Menú principal".
2. El sistema despliega las opciones de "Clientes".
3. El actor selecciona la opción "Administrar Clientes".
4. El sistema carga la lista de clientes registrados (E1).
5. El sistema muestra la interfaz de los clientes.
6. El actor selecciona el cliente que desea modificar.
7. El actor presiona el botón "Modificar".
8. El sistema muestra la interfaz de modificar con los datos del cliente seleccionado.
9. El actor modifica los datos del cliente.
10. El actor presiona el botón "Actualizar".
11. El sistema actualiza los datos del cliente en la base de datos (E1).

12. El sistema muestra el mensaje de éxito o error de la acción.

Flujo alterno

9. El actor decide no actualizar al cliente y cierra la ventana de modificación.

Excepciones

Tabla 7

Excepciones Caso de Uso: Modificar clientes

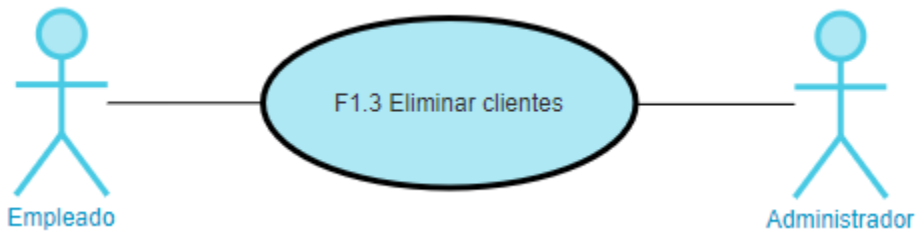
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F1.3** Eliminar clientes

Figura 7

Caso de uso: Nivel detallado, F1.3 Eliminar clientes



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán eliminar un cliente del sistema.

Actores: Administrador y empleado.

Flujo principal

1. El actor selecciona la opción “Clientes” del “Menú principal”.

2. El sistema despliega las opciones de “Clientes”.
3. El actor selecciona la opción “Administrar Clientes”.
4. El sistema carga la lista de clientes registrados (E1).
5. El sistema muestra la interfaz de los clientes.
6. El actor selecciona el cliente que desea eliminar.
7. El actor presiona el botón “Eliminar”.
8. El sistema muestra una alerta para confirmar la eliminación del usuario.
9. El actor presiona el botón “Confirmar”.
10. El sistema elimina los datos del cliente de la base de datos (E1).
11. El sistema muestra el mensaje de éxito o error de la acción.

Flujo alternativo

9. El actor presiona el botón “Cancelar”.
10. El sistema cierra la ventana de confirmación.

Excepciones

Tabla 8

Excepciones Caso de Uso: Eliminar clientes

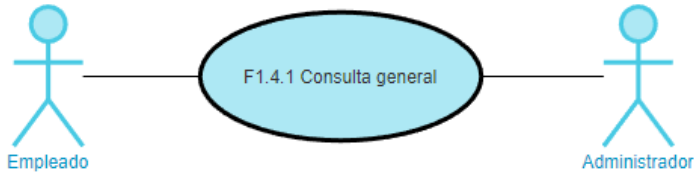
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F1.4.1** Consulta general

Figura 8

Caso de uso: Nivel detallado, F1.4.1 Consulta general



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán visualizar los clientes registrados.

Actores: Administrador y empleado.

Flujo principal

1. El actor selecciona la opción “Clientes” del “Menú principal”.
2. El sistema despliega las opciones de “Clientes”.
3. El actor selecciona la opción “Administrar Clientes”.
4. El sistema carga la lista de clientes registrados (E1).
5. El sistema muestra la interfaz de los clientes.

Flujo alternativo

6. En caso de querer modificar o eliminar un cliente, ver caso de uso F1.2 Modificar Clientes o F1.3 Eliminar Clientes.

Excepciones

Tabla 9

Excepciones Caso de Uso: Consulta general de clientes

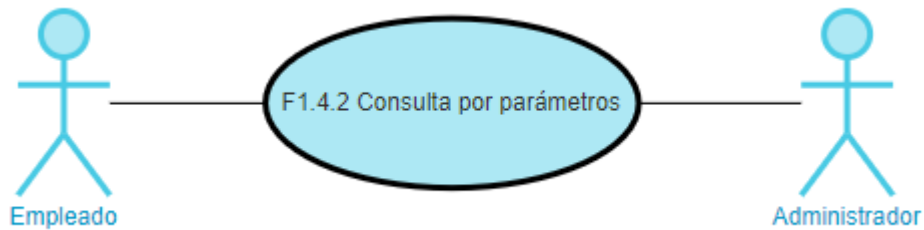
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F1.4.2** Consulta por parámetros

Figura 9

Caso de uso: Nivel detallado, F1.4.2 Consulta por parámetros



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán visualizar y filtrar por parámetros a los clientes registrados.

Actores: Administrador y empleado.

Flujo principal

1. El actor selecciona la opción “Clientes” del “Menú principal”.
2. El sistema despliega las opciones de “Clientes”.
3. El actor selecciona la opción “Administrar Clientes”.
4. El sistema carga la lista de clientes registrados (E1).
5. El sistema muestra la interfaz de los clientes.
6. El actor selecciona el campo a filtrar.
7. El actor escribe el texto a buscar.
8. El sistema filtra los resultados.
9. El sistema muestra el resultado de la búsqueda.

Flujo alterno

10. En caso de querer modificar o eliminar un cliente, ver caso de uso F1.2 Modificar Clientes o F1.3 Eliminar Clientes.

Excepciones

Tabla 10

Excepciones Caso de Uso: Consulta por parámetros de clientes

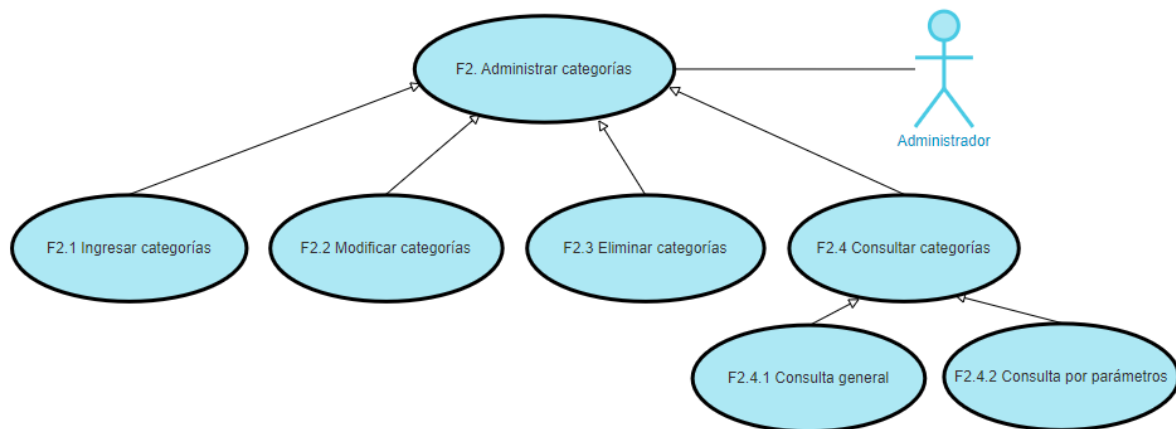
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

4.4.3. F2. Administrar categorías

Figura 10

Caso de uso: Siguiete nivel, F2 Administrar categorías



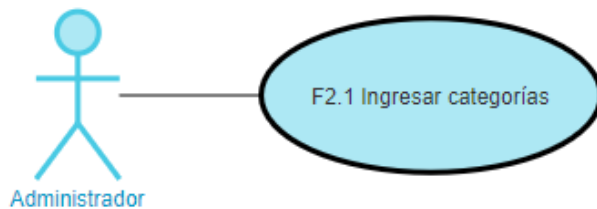
Nota. Elaboración propia.

Caso de uso: Nivel de detalle

- **F2.1** Ingresar categorías

Figura 11

Caso de uso: Nivel detallado, F2.1 Ingresar categorías



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán ingresar una categoría al sistema.

Actores: Administrador.

Flujo principal

1. El actor selecciona la opción “Categorías” del “Menú principal”.
2. El sistema despliega las opciones de “Categorías”.
3. El actor selecciona la opción “Ingresar Categoría”.
4. Se despliega la interfaz de “Ingresar Categoría”.
5. El actor ingresa el nombre de la categoría.
6. El actor presiona el botón “Registrar”.
7. El sistema registra los datos de la categoría en la base de datos (E1, E2).
8. El sistema muestra el mensaje de éxito o error de la acción.

Flujo alternativo

9. Si los datos de la categoría ya existen, dirigirse a Modificar categorías (F2.2).

Excepciones

Tabla 11

Excepciones Caso de Uso: Ingresar categorías

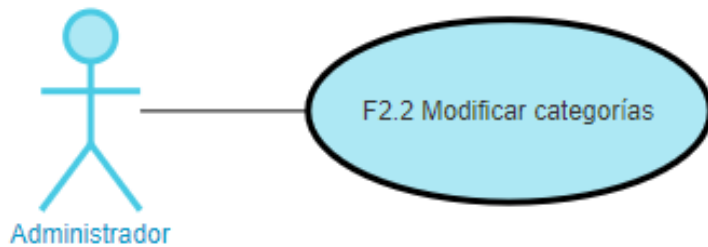
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD
E2	Categoría existente	Verificar los datos de la categoría

Nota. Elaboración propia.

- **F2.2** Modificar categorías

Figura 12

Caso de uso: Nivel detallado, F2.2 Modificar categorías



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán modificar una categoría del sistema.

Actores: Administrador.

Flujo principal

1. El actor selecciona la opción “Categorías” del “Menú principal”.
2. El sistema despliega las opciones de “Categorías”.

3. El actor selecciona la opción “Administrar Categorías”.
4. El sistema carga la lista de categorías registradas (E1).
5. El sistema muestra la interfaz de las categorías.
6. El actor selecciona la categoría que desea modificar.
7. El actor presiona el botón “Modificar”.
8. El sistema muestra la interfaz de modificar con los datos de la categoría seleccionada.
9. El actor modifica los datos de la categoría.
10. El actor presiona el botón “Actualizar”.
11. El sistema actualiza los datos de la categoría en la base de datos (E1).
12. El sistema muestra el mensaje de éxito o error de la acción.

Flujo alternativo

9. El actor decide no actualizar la información y cierra la ventana de modificación.

Excepciones

Tabla 12

Excepciones Caso de Uso: Modificar categorías

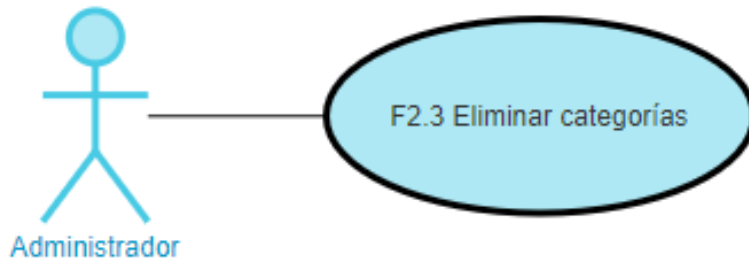
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F2.3** Eliminar categorías

Figura 13

Caso de uso: Nivel detallado, F2.3 Eliminar categorías



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán eliminar una categoría del sistema.

Actores: Administrador.

Flujo principal

1. El actor selecciona la opción “Categorías” del “Menú principal”.
2. El sistema despliega las opciones de “Categorías”.
3. El actor selecciona la opción “Administrar Categorías”.
4. El sistema carga la lista de categorías registradas (E1).
5. El sistema muestra la interfaz de las categorías.
6. El actor selecciona la categoría que desea eliminar.
7. El actor presiona el botón “Eliminar”.
8. El sistema muestra una alerta para confirmar la eliminación de la categoría.
9. El actor presiona el botón “Confirmar”.
10. El sistema elimina los datos de la categoría de la base de datos (E1).
11. El sistema muestra el mensaje de éxito o error de la acción.

Flujo alternativo

9. El actor presiona el botón "Cancelar".
10. El sistema cierra la ventana de confirmación.

Excepciones

Tabla 13

Excepciones Caso de Uso: Eliminar categorías

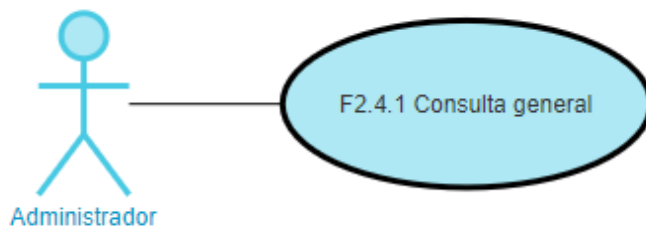
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F2.4.1** Consulta general

Figura 14

Caso de uso: Nivel detallado, F2.4.1 Consulta general



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán visualizar las categorías registradas.

Actores: Administrador.

Flujo principal

1. El actor selecciona la opción "Categorías" del "Menú principal".

2. El sistema despliega las opciones de “Categorías”.
3. El actor selecciona la opción “Administrar Categorías”.
4. El sistema carga la lista de categorías registradas (E1).
5. El sistema muestra la interfaz de las categorías.

Flujo alternativo

6. En caso de querer modificar o eliminar una categoría, ver caso de uso F2.2 Modificar categorías o F2.3 Eliminar categorías.

Excepciones

Tabla 14

Excepciones Caso de Uso: Consulta general de categorías

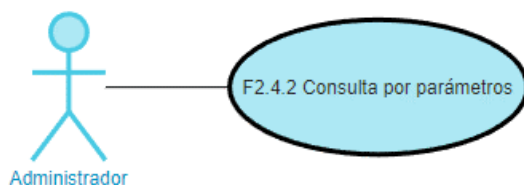
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F2.4.2** Consulta por parámetros

Figura 15

Caso de uso: Nivel detallado, F2.4.2 Consulta por parámetros



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán visualizar y filtrar por parámetros las categorías registradas.

Actores: Administrador.

Flujo principal

1. El actor selecciona la opción “Categorías” del “Menú principal”.
2. El sistema despliega las opciones de “Categorías”.
3. El actor selecciona la opción “Administrar Categorías”.
4. El sistema carga la lista de categorías registradas (E1).
5. El sistema muestra la interfaz de las categorías.
6. El actor selecciona el campo a filtrar.
7. El actor escribe el texto a buscar.
8. El sistema filtra los resultados.
9. El sistema muestra el resultado de la búsqueda.

Flujo alternativo

10. En caso de querer modificar o eliminar una categoría, ver caso de uso F2.2 Modificar categorías o F2.3 Eliminar categorías.

Excepciones

Tabla 15

Excepciones Caso de Uso: Consulta por parámetros de categorías

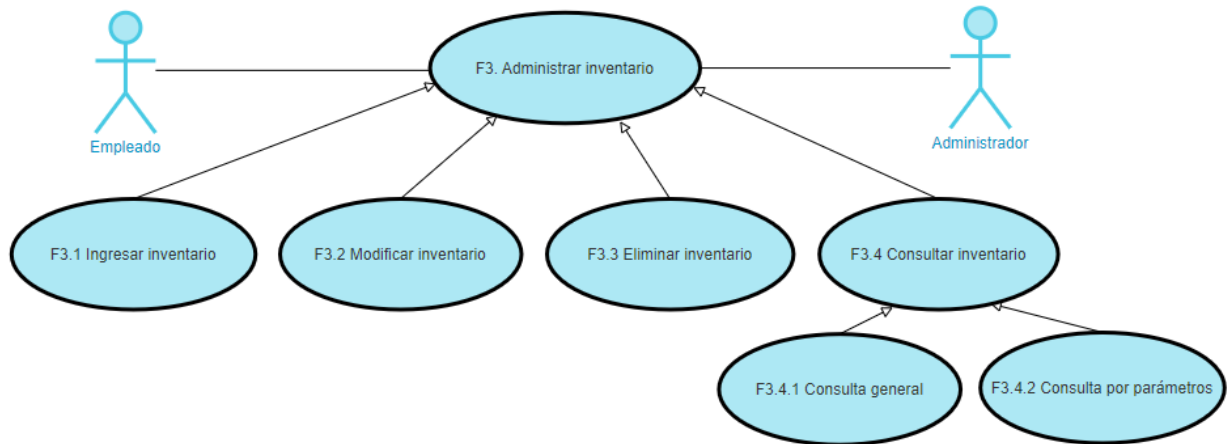
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

4.4.4. F3. Administrar inventario

Figura 16

Caso de uso: Siguiete nivel, F3 Administrar inventario



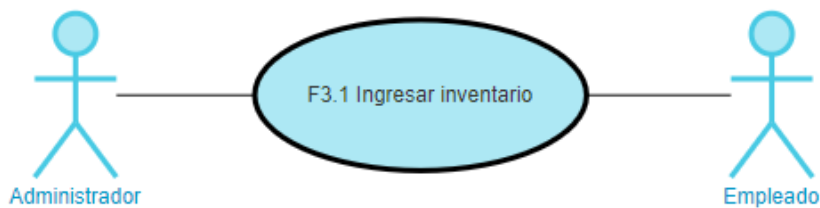
Nota. Elaboración propia.

Caso de uso: Nivel de detalle

- **F3.1** Ingresar inventario

Figura 17

Caso de uso: Nivel detallado, F3.1 Ingresar inventario



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán ingresar un producto del inventario al sistema.

Actores: Administrador y empleado.

Flujo principal

1. El actor selecciona la opción “Inventario” del “Menú principal”.
2. El sistema despliega las opciones de “Inventario”.
3. El actor selecciona la opción “Ingresar Producto”.
4. Se despliega la interfaz de “Ingresar Producto”.
5. El actor ingresa los datos del producto.
6. El actor presiona el botón “Registrar”.
7. El sistema registra los datos del producto en la base de datos (E1, E2).
8. El sistema muestra el mensaje de éxito o error de la acción.

Flujo alternativo

9. Si el producto ya existe, dirigirse a Modificar inventario (F1.2).

Excepciones

Tabla 16

Excepciones Caso de Uso: Ingresar inventario

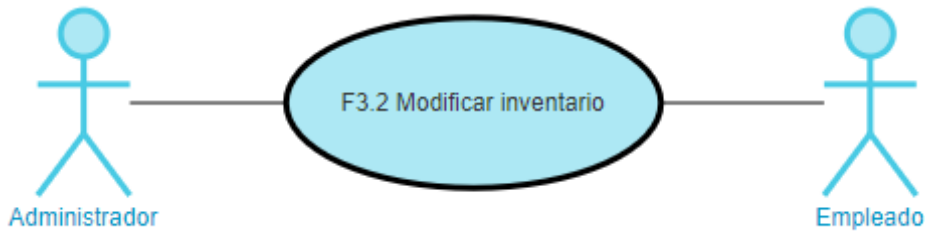
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD
E2	Producto existente	Verificar el producto

Nota. Elaboración propia.

- **F3.2** Modificar inventario

Figura 18

Caso de uso: Nivel detallado, F3.2 Modificar inventario



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán modificar un producto del inventario del sistema.

Actores: Administrador y empleado.

Flujo principal

1. El actor selecciona la opción "Inventario" del "Menú principal".
2. El sistema despliega las opciones de "Inventario".
3. El actor selecciona la opción "Administrar Inventario".
4. El sistema carga la lista de productos registrados (E1).
5. El sistema muestra la interfaz de los productos.
6. El actor selecciona el producto del inventario que desea modificar.
7. El actor presiona el botón "Modificar".
8. El sistema muestra la interfaz de modificar con los datos de producto del inventario seleccionado.
9. El actor modifica los datos del producto.
10. El actor presiona el botón "Actualizar".
11. El sistema actualiza los datos del producto del inventario en la base de datos (E1).
12. El sistema muestra el mensaje de éxito o error de la acción.

Flujo alternativo

7. Si el actor desea modificar el stock del producto del inventario, presionará el botón de añadir o reducir el stock.
9. El actor decide no actualizar la información y cierra la ventana de modificación.

Excepciones

Tabla 17

Excepciones Caso de Uso: Modificar inventario

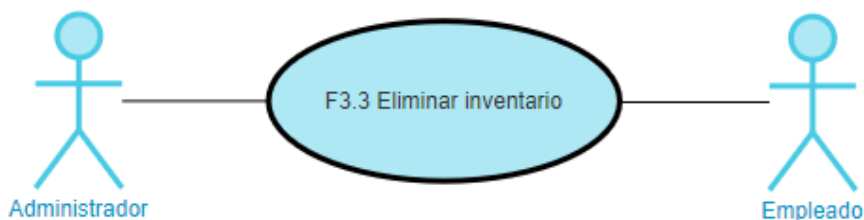
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F3.3** Eliminar inventario

Figura 19

Caso de uso: Nivel detallado, F3.3 Eliminar inventario



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán eliminar un producto del inventario del sistema.

Actores: Administrador y empleado.

Flujo principal

1. El actor selecciona la opción “Inventario” del “Menú principal”.
2. El sistema despliega las opciones de “Inventario”.
3. El actor selecciona la opción “Administrar Inventario”.
4. El sistema carga la lista productos del inventario registrados (E1).
5. El sistema muestra la interfaz del inventario.
6. El actor selecciona el producto del inventario que desea eliminar.
7. El actor presiona el botón “Eliminar”.
8. El sistema muestra una alerta para confirmar la eliminación del producto.
9. El actor presiona el botón “Confirmar”.
10. El sistema elimina los datos del producto de la base de datos (E1).
11. El sistema muestra el mensaje de éxito o error de la acción.

Flujo alterno

9. El actor presiona el botón “Cancelar”.
10. El sistema cierra la ventana de confirmación.

Excepciones

Tabla 18

Excepciones Caso de Uso: Eliminar inventario

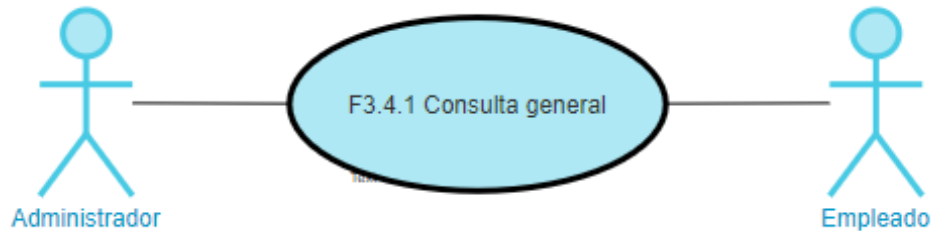
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F3.4.1** Consulta general

Figura 20

Caso de uso: Nivel detallado, F3.4.1 Consulta general



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán visualizar los productos del inventario registrados.

Actores: Administrador y empleado.

Flujo principal

1. El actor selecciona la opción "Inventario" del "Menú principal".
2. El sistema despliega las opciones de "Inventario".
3. El actor selecciona la opción "Administrar Inventario".
4. El sistema carga la lista de productos del inventario (E1).
5. El sistema muestra la interfaz del inventario.

Flujo alternativo

6. En caso de querer modificar o eliminar un producto del inventario, ver caso de uso F3.2 Modificar Inventario o F3.3 Eliminar Inventario.

Excepciones

Tabla 19

Excepciones Caso de Uso: Consulta general de inventario

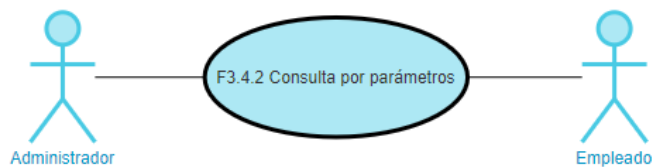
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F3.4.2** Consulta por parámetros

Figura 21

Caso de uso: Nivel detallado, F3.4.2 Consulta por parámetros



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán visualizar y filtrar por parámetros a los productos del inventario.

Actores: Administrador y empleado.

Flujo principal

1. El actor selecciona la opción “Inventario” del “Menú principal”.
2. El sistema despliega las opciones de “Inventario”.
3. El actor selecciona la opción “Administrar Inventario”.
4. El sistema carga la lista de productos del inventario (E1).
5. El sistema muestra la interfaz del inventario
6. El actor selecciona el campo a filtrar.

7. El actor escribe el texto a buscar.
8. El sistema filtra los resultados.
9. El sistema muestra el resultado de la búsqueda.

Flujo alternativo

10. En caso de querer modificar o eliminar un producto del inventario, ver caso de uso F3.2 Modificar Inventario o F3.3 Eliminar Inventario.

Excepciones

Tabla 20

Excepciones Caso de Uso: Consulta por parámetros de inventario

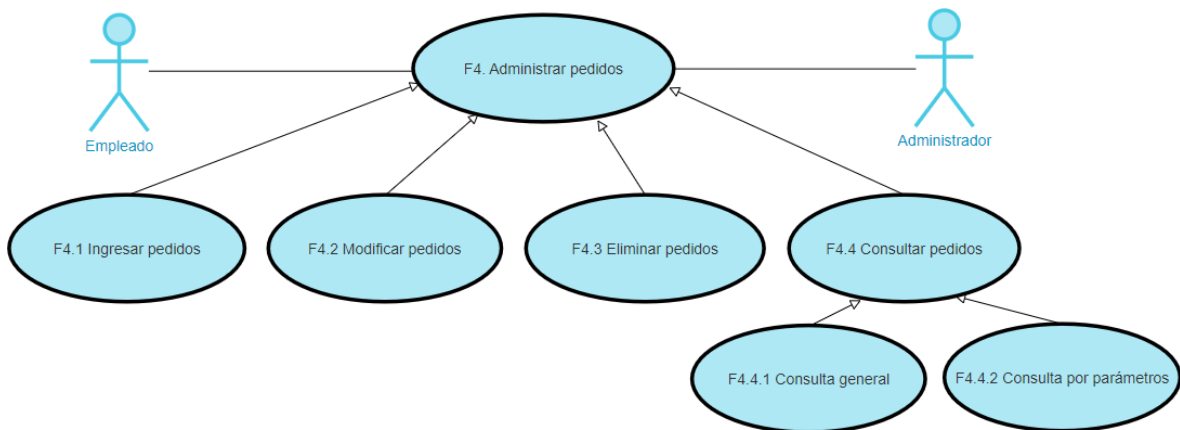
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

4.4.5. F4. Administrar pedidos

Figura 22

Caso de uso: Siguiete nivel, F4 Administrar pedidos



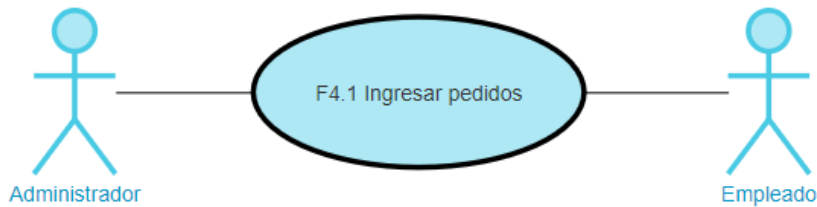
Nota. Elaboración propia.

Caso de uso: Nivel de detalle

- **F4.1** Ingresar pedidos

Figura 23

Caso de uso: Nivel detallado, F4.1 Ingresar pedidos



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán ingresar un pedido al sistema.

Actores: Administrador y empleado.

Flujo principal

1. El actor selecciona la opción "Pedidos" del "Menú principal".
2. El sistema despliega las opciones de "Pedidos".
3. El actor selecciona la opción "Ingresar Pedido".
4. Se despliega la interfaz de "Ingresar Pedido".
5. El actor ingresa los datos del cliente.
6. El actor presiona el botón buscar.
7. El sistema muestra la información del cliente buscado.
8. El actor presiona "Seleccionar Productos".
9. El sistema despliega la interfaz de "Seleccionar Productos".
10. El actor selecciona los productos.

11. El actor presiona “Listo”
12. El actor introduce la cantidad de cada producto.
13. El sistema presenta los valores de subtotales y total.
14. El actor ingresa la “información extra” del pedido.
15. El actor presiona el botón “Agregar Pedido”.
16. El sistema registra los datos del pedido en la base de datos (E1).
17. El sistema muestra el mensaje de éxito o error de la acción.

Flujo alternativo

7. Si el cliente no existe, dirigirse a Ingresar clientes (F1.1)
13. Si el actor introduce un stock mayor al actual del producto, se presentará un mensaje de error.
17. Si el pedido se encuentra en estado “Completado”, el sistema actualizará el stock de los productos.

Excepciones

Tabla 21

Excepciones Caso de Uso: Ingresar pedidos

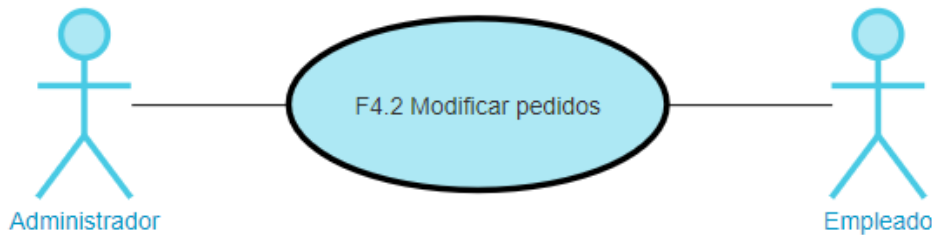
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F4.2** Modificar pedidos

Figura 24

Caso de uso: Nivel detallado, F4.2 Modificar pedidos



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán modificar un pedido del sistema.

Actores: Administrador y empleado.

Flujo principal

1. El actor selecciona la opción “Pedidos” del “Menú principal”.
2. El sistema despliega las opciones de “Pedidos”.
3. El actor selecciona la opción “Administrar Pedidos”.
4. El sistema carga la lista de pedidos registrados (E1).
5. El sistema muestra la interfaz de los pedidos.
6. El actor selecciona el pedido que desea modificar.
7. El actor presiona el botón “Modificar”.
8. El sistema muestra la interfaz de modificar con los datos del pedido seleccionado.
9. El actor modifica los datos deseados.
10. El actor presiona el botón “Actualizar”.
11. El sistema actualiza los datos del pedido en la base de datos (E1).

12. El sistema muestra el mensaje de éxito o error de la acción.

Flujo alterno

10. El actor cierra la ventana en caso de no querer modificar el pedido.

Excepciones

Tabla 22

Excepciones Caso de Uso: Modificar pedidos

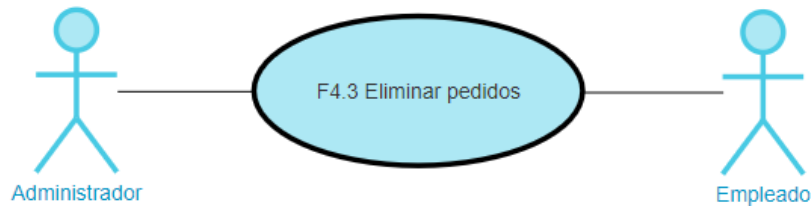
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F4.3** Eliminar pedidos

Figura 25

Caso de uso: Nivel detallado, F4.3 Eliminar pedidos



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán eliminar un pedido del sistema.

Actores: Administrador y empleado.

Flujo principal

1. El actor selecciona la opción "Pedidos" del "Menú principal".
2. El sistema despliega las opciones de "Pedidos".
3. El actor selecciona la opción "Administrar Pedidos".
4. El sistema carga la lista de pedidos registrados (E1).
5. El sistema muestra la interfaz de los pedidos.
6. El actor selecciona el pedido que desea eliminar.
7. El actor presiona el botón "Eliminar".
8. El sistema muestra una alerta para confirmar la eliminación del pedido.
9. El actor presiona el botón "Confirmar".
10. El sistema elimina los datos del pedido de la base de datos (E1).
11. El sistema muestra el mensaje de éxito o error de la acción.

Flujo alterno

11. El actor presiona el botón "Cancelar".
12. El sistema cierra la ventana de confirmación.

Excepciones

Tabla 23

Excepciones Caso de Uso: Eliminar pedidos

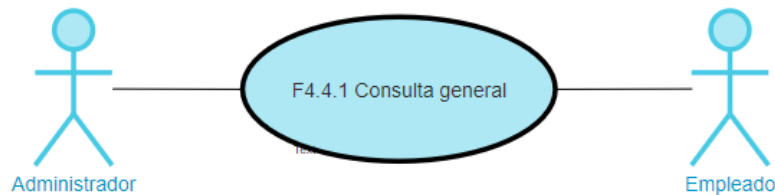
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F4.4.1** Consulta general

Figura 26

Caso de uso: Nivel detallado, F4.4.1 Consulta general



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán visualizar los pedidos registrados.

Actores: Administrador y empleado.

Flujo principal

7. El actor selecciona la opción "Pedidos" del "Menú principal".
8. El sistema despliega las opciones de "Pedidos".
9. El actor selecciona la opción "Administrar Pedidos".
10. El sistema carga la lista de pedidos registrados (E1).
11. El sistema muestra la interfaz de los pedidos.

Flujo alternativo

12. En caso de querer modificar o eliminar un pedido, ver caso de uso F4.2 Modificar Pedidos o F4.3 Eliminar Pedidos.

Excepciones

Tabla 24

Excepciones Caso de Uso: Consulta general de pedidos

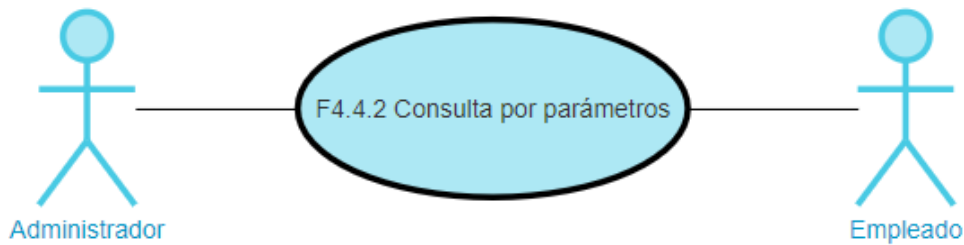
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F4.4.2** Consulta por parámetros

Figura 27

Caso de uso: Nivel detallado, F4.4.2 Consulta por parámetros



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán visualizar y filtrar por parámetros a los pedidos registrados.

Actores: Administrador y empleado.

Flujo principal

1. El actor selecciona la opción "Pedidos" del "Menú principal".
2. El sistema despliega las opciones de "Pedidos".
3. El actor selecciona la opción "Administrar Pedidos".
4. El sistema carga la lista de pedidos registrados (E1).
5. El sistema muestra la interfaz de los pedidos

6. El actor selecciona el campo a filtrar.
7. El actor escribe el texto a buscar.
8. El sistema filtra los resultados.
9. El sistema muestra el resultado de la búsqueda.

Flujo alternativo

10. En caso de querer modificar o eliminar un pedido, ver caso de uso F4.2 Modificar Pedidos o F4.3 Eliminar Pedidos.

Excepciones

Tabla 25

Excepciones Caso de Uso: Consulta por parámetros de pedidos

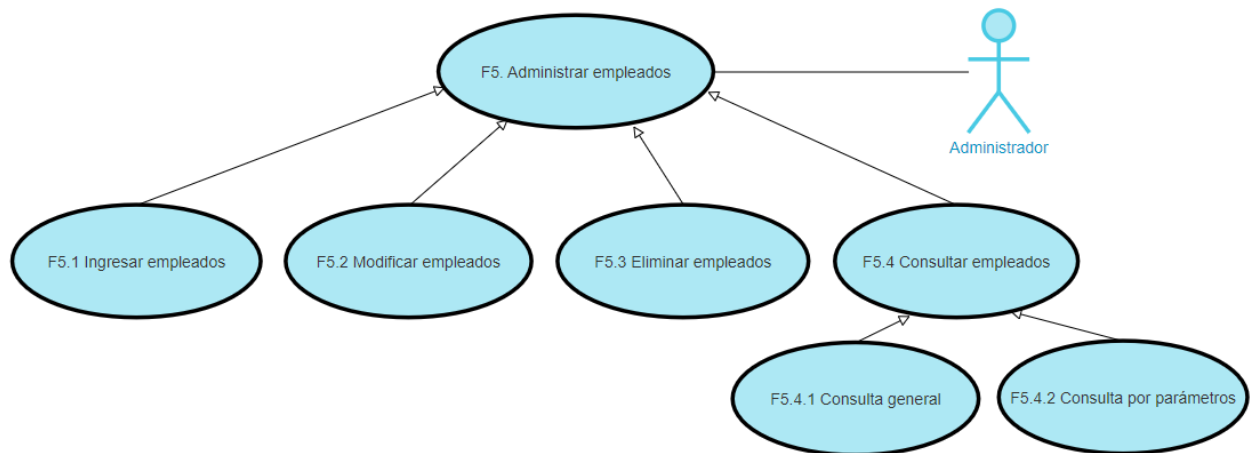
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

4.4.6. F5. Administrar empleados

Figura 28

Caso de uso: Siguiete nivel, F5 Administrar empleados



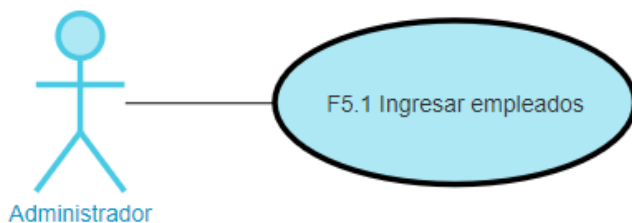
Nota. Elaboración propia.

Caso de uso: Nivel de detalle

- **F5.1** Ingresar empleados

Figura 29

Caso de uso: Nivel detallado, F5.1 Ingresar empleados



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán ingresar un empleado al sistema.

Actores: Administrador.

Flujo principal

1. El actor selecciona la opción “Empleados” del “Menú principal”.
2. El sistema despliega las opciones de “Empleados”.
3. El actor selecciona la opción “Ingresar Empleado”.
4. Se despliega la interfaz de “Ingresar Empleado”.
5. El actor ingresa los datos del empleado.
6. El actor presiona el botón “Registrar”.
7. El sistema registra los datos del empleado en la base de datos (E1, E2).
8. El sistema muestra el mensaje de éxito o error de la acción.

Flujo alternativo

9. Si los datos del empleado ya existen, dirigirse a Modificar Empleados (F5.2).

Excepciones

Tabla 26

Excepciones Caso de Uso: Ingresar empleados

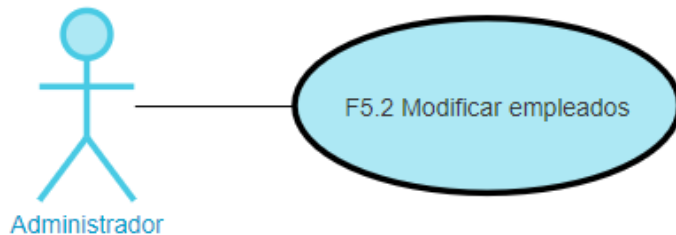
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD
E2	Empleado existente	Verificar los datos del empleado

Nota. Elaboración propia.

- **F5.2** Modificar empleados

Figura 30

Caso de uso: Nivel detallado, F5.2 Modificar empleados



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán modificar un empleado del sistema.

Actores: Administrador.

Flujo principal

1. El actor selecciona la opción “Empleados” del “Menú principal”.
2. El sistema despliega las opciones de “Empleados”.
3. El actor selecciona la opción “Administrar Empleados”.
4. El sistema carga la lista de empleados registrados (E1).
5. El sistema muestra la interfaz de los empleados.
6. El actor selecciona el empleado que desea modificar.
7. El actor presiona el botón “Modificar”.
8. El sistema muestra la interfaz de modificar con los datos del empleado seleccionado.
9. El actor modifica los datos del empleado.
10. El actor presiona el botón “Actualizar”.
11. El sistema actualiza los datos del empleado en la base de datos (E1).

12. El sistema muestra el mensaje de éxito o error de la acción.

Flujo alterno

13. Si los datos del empleado ya existen, dirigirse a Modificar Empleados (F5.2).

Excepciones

Tabla 27

Excepciones Caso de Uso: Modificar empleados

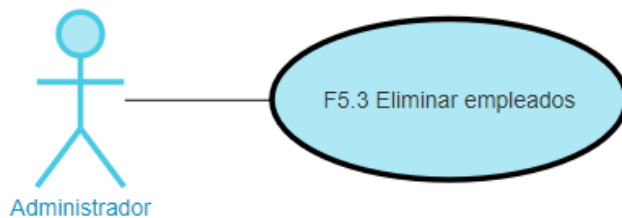
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F5.3** Eliminar empleados

Figura 31

Caso de uso: Nivel detallado, F5.3 Eliminar empleados



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán eliminar un empleado del sistema.

Actores: Administrador.

Flujo principal

1. El actor selecciona la opción "Empleados" del "Menú principal".

2. El sistema despliega las opciones de “Empleados”.
3. El actor selecciona la opción “Administrar Empleados”.
4. El sistema carga la lista de empleados registrados (E1).
5. El sistema muestra la interfaz de los empleados.
6. El actor selecciona el empleado que desea eliminar.
7. El actor presiona el botón “Eliminar”.
8. El sistema muestra una alerta para confirmar la eliminación del empleado.
9. El actor presiona el botón “Confirmar”.
10. El sistema elimina los datos del empleado de la base de datos (E1).
11. El sistema muestra el mensaje de éxito o error de la acción.

Flujo alterno

13. El actor presiona el botón “Cancelar”.
14. El sistema cierra la ventana de confirmación.

Excepciones

Tabla 28

Excepciones Caso de Uso: Eliminar empleado

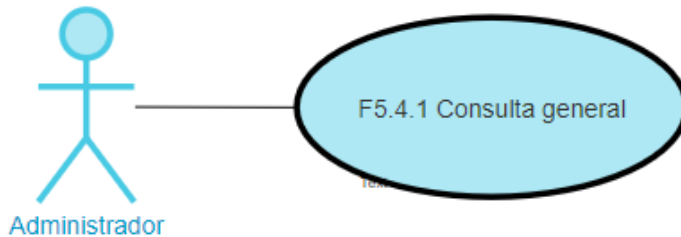
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F5.4.1** Consulta general

Figura 32

Caso de uso: Nivel detallado, F5.4.1 Consulta general



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán visualizar los empleados registrados.

Actores: Administrador.

Flujo principal

1. El actor selecciona la opción “Empleados” del “Menú principal”.
2. El sistema despliega las opciones de “Empleados”.
3. El actor selecciona la opción “Administrar Empleados”.
4. El sistema carga la lista de empleados registrados (E1).
5. El sistema muestra la interfaz de los empleados.

Flujo alternativo

6. En caso de querer modificar o eliminar un empleado, ver caso de uso F5.2 Modificar empleados o F5.3 Eliminar empleados.

Excepciones

Tabla 29

Excepciones Caso de Uso: Consulta general de empleados

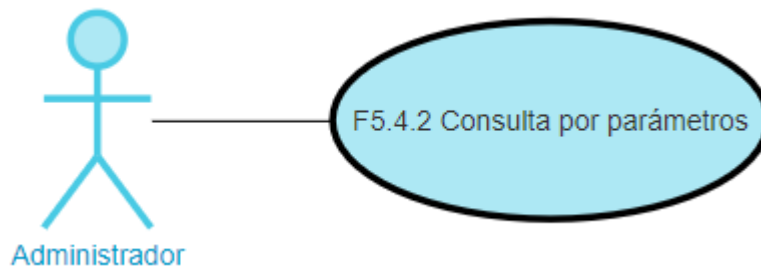
Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

- **F5.4.2** Consulta por parámetros

Figura 33

Caso de uso: Nivel detallado, F5.4.2 Consulta por parámetros



Nota. Elaboración propia.

Descripción: A través de este caso de uso los actores podrán visualizar y filtrar por parámetros a los empleados registrados.

Actores: Administrador.

Flujo principal

1. El actor selecciona la opción "Empleados" del "Menú principal".
2. El sistema despliega las opciones de "Empleados".
3. El actor selecciona la opción "Administrar Empleados".
4. El sistema carga la lista de empleados registrados (E1).
5. El sistema muestra la interfaz de los empleados.

6. El actor selecciona el campo a filtrar.
7. El actor escribe el texto a buscar.
8. El sistema filtra los resultados.
9. El sistema muestra el resultado de la búsqueda.

Flujo alternativo

10. En caso de querer modificar o eliminar un empleado, ver caso de uso F5.2 Modificar Empleados o F5.3 Eliminar Empleados.

Excepciones

Tabla 30

Excepciones Caso de Uso: Consulta por parámetros de empleados

Código	Causa	Sugerencia de solución
E1	Problemas con BDD	Notificar al Administrador de BDD

Nota. Elaboración propia.

5. Desarrollo del sistema

5.1. Estándares de programación

En la programación se utilizó el estilo camelCase en la escritura de variables y funciones ya que redujo el esfuerzo de lectura y aportó de forma positiva en la apariencia del código. En el siguiente gráfico se presenta un ejemplo del estilo de programación adoptado.

Figura 34

Ejemplo de estándar de codificación adoptado

```
src > app > pages > login > login.page.ts > ...
1 import { Component, OnInit } from '@angular/core';
2 import { FormBuilder, FormGroup, Validators } from '@angular/forms';
3 import { ModalController } from '@ionic/angular';
4 import { ForgotPasswordPage } from 'src/app/modals/forgot-password/forgot-password.page';
5 import { AlertService } from 'src/app/services/alert/alert.service';
6 import { AuthService } from 'src/app/services/auth/auth.service';
7
8 @Component({
9   selector: 'app-login',
10  templateUrl: './login.page.html',
11  styleUrls: ['./login.page.scss'],
12 })
13 export class LoginPage implements OnInit {
14
15   public loginForm: FormGroup;
16
17   constructor(private fb: FormBuilder, private alertService: AlertService, private authService: AuthService,
18     this.createLoginForm());
19 }
20
21 ngOnInit() { }
22
23 login() { //Validates login form
24   var email = this.loginForm.value['email'];
25   var password = this.loginForm.value['password'];
26   if(this.loginForm.valid){
27     this.authService.logIn(email, password);
28   }else{
29     this.alertService.toast({ icon: 'error', title: '¡Ha ocurrido un error!', text: 'Debes completar todos
30     this.loginForm.markAllAsTouched();
31   }
32 }
33
34 createLoginForm() { //Generates login form
35   this.loginForm = this.fb.group({
36     "email": [ "", [Validators.required, Validators.email]],
37     "password": [ "", [Validators.required]]
38   });
39 }
40
```

Nota. Elaboración propia.

5.2. Organización de directorios

Debido a que se utilizó un framework que se caracteriza principalmente por ser modular, resultó de suma importancia definir los directorios principales para la separación de cada página, componente, servicio, modal y guardianes de autenticación. Es así como se lo realizó de la siguiente forma:

- **Páginas**
 - **Ubicación:** app/pages
 - **Contenido por página:** Plantillas (HTML), módulos y rutas (ts), hojas de estilos preprocesadas (SCSS) y componentes de pruebas (spec.ts).
- **Componentes**
 - **Ubicación:** app/components
 - **Contenido por página:** Plantillas (HTML), módulo (ts), hojas de estilos preprocesadas (SCSS) y componentes de pruebas (spec.ts).
- **Servicios:**
 - **Ubicación:** app/services
 - **Contenido por página:** Código fuente (service.ts) y componentes de pruebas (service.spec.ts).
- **Modales:**
 - **Ubicación:** app/modals
 - **Contenido por página:** Plantillas (HTML), módulos y rutas (ts), hojas de estilos preprocesadas (SCSS) y componentes de pruebas (spec.ts).
- **Guardianes de autenticación:**
 - **Ubicación:** app/guards
 - **Contenido por página:** Código fuente (guard.ts) y componente de pruebas (guard.spec.ts).

5.3. Iteraciones

A continuación, se presentan los resultados obtenidos en cada iteración:

5.3.1. Iteración 1

Tabla 31

Resumen de estados de Iteración 1

Historia de Usuario	Tarea	Estado
HU1	F1.1 Ingresar clientes	Aprobado
	F1.2 Modificar clientes	Aprobado
	F1.3 Eliminar clientes	Aprobado
	F1.4.1 Consulta general	Aprobado
	F1.4.2 Consulta por parámetros	Aprobado

Nota. Elaboración propia.

Tabla 32

Caso de Prueba de Aceptación HU1_P01

Caso de Prueba de Aceptación			
Código	HU1_P01	Historia de Usuario	HU1
Nombre	Introducir una cédula incorrecta y comprobar que se indique el error.		
Descripción	Se probará que se validen las cédulas de forma correcta.		
Condiciones de ejecución	<ul style="list-style-type: none">• Usuario con permisos de administrador o empleado.		
Entrada / Pasos de ejecución	<ul style="list-style-type: none">• Se accederá a la ventana de Ingreso de Clientes.• Se introducirá una cédula inválida.		
Resultado esperado	Se indica el error.		

Evaluación de la prueba	Prueba satisfactoria.
--------------------------------	-----------------------

Nota. Elaboración propia.

Tabla 33

Caso de Prueba de Aceptación HU1_P02

Caso de Prueba de Aceptación			
Código	HU1_P02	Historia de Usuario	HU1
Nombre	Introducir un usuario ya existente, y comprobar que el sistema no permita agregarlo nuevamente.		
Descripción	Se probará que no es posible agregar clientes con el mismo documento de identidad.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. • Información de un cliente existente. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Ingreso de Clientes. • Se selecciona el tipo de documento “Cédula”. • Se introducirá la cédula de un cliente existente. • Se completarán los demás campos. • Se presionará el botón “Registrar”. 		
Resultado esperado	Se presenta el mensaje de error.		
Evaluación de la prueba	Prueba satisfactoria.		

Nota. Elaboración propia.

Tabla 34*Caso de Prueba de Aceptación HU1_P03*

Caso de Prueba de Aceptación			
Código	HU1_P03	Historia de Usuario	HU1
Nombre	Dejar todos los campos vacíos y comprobar que existan validaciones.		
Descripción	Se probará que las validaciones se encuentren funcionando.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Ingreso de Clientes. • Se presionará el botón "Registrar" 		
Resultado esperado	Se presenta el mensaje de error y se indican los errores.		
Evaluación de la prueba	Prueba satisfactoria.		

Nota. Elaboración propia.**Tabla 35***Caso de Prueba de Aceptación HU1_P04*

Caso de Prueba de Aceptación			
Código	HU1_P04	Historia de Usuario	HU1
Nombre	Introducir todos los datos correctos y comprobar que el sistema registre la información.		
Descripción	Se comprobará que se puedan ingresar nuevos clientes satisfactoriamente.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. 		

Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Ingreso de Clientes. • Se completarán todos los campos de forma satisfactoria. • Se presionará el botón “Registrar”
Resultado esperado	Se registrará la información del cliente en la base de datos y se presentará el mensaje de buen trabajo.
Evaluación de la prueba	Prueba satisfactoria.

Nota. Elaboración propia.

Tabla 36

Caso de Prueba de Aceptación HU1_P05

Caso de Prueba de Aceptación			
Código	HU1_P05	Historia de Usuario	HU1
Nombre	Modificar un cliente y comprobar que se actualice.		
Descripción	Se comprobará que se puedan ingresar nuevos clientes satisfactoriamente.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. • Cliente registrado. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Administrar Clientes. • Se seleccionará un usuario al azar. • Se presionará el botón con ícono de modificar. • Se modificará la información del usuario. • Se presionará el botón “Actualizar” 		
Resultado esperado	Se actualizará la información del cliente en la base de datos y se presentará el mensaje de buen trabajo.		

Evaluación de la prueba	Prueba satisfactoria.
--------------------------------	-----------------------

Nota. Elaboración propia.

Tabla 37

Caso de Prueba de Aceptación HU1_P06

Caso de Prueba de Aceptación			
Código	HU1_P06	Historia de Usuario	HU1
Nombre	Eliminar un cliente.		
Descripción	Se comprobará que se puedan eliminar clientes satisfactoriamente.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. • Cliente registrado. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Administrar Clientes. • Se seleccionará un usuario al azar. • Se presionará el botón con ícono de eliminar. • Se presionará el botón “Confirmar” 		
Resultado esperado	Se eliminará la información del cliente en la base de datos y se presentará el mensaje de confirmación.		
Evaluación de la prueba	Prueba satisfactoria.		

Nota. Elaboración propia.

5.3.2. Iteración 2

Tabla 38

Resumen de estados de Iteración 2

Historia de Usuario	Tarea	Estado
HU2	F2.1 Ingresar categorías	Aprobado
	F2.2 Modificar categorías	Aprobado
	F2.3 Eliminar categorías	Aprobado
	F2.4.1 Consulta general	Aprobado
	F2.4.2 Consulta por parámetros	Aprobado
HU3	F3.1 Ingresar pedidos	Aprobado
	F3.2 Modificar pedidos	Aprobado
	F3.3 Eliminar pedidos	Aprobado
	F3.4.1 Consulta general	Aprobado
	F3.4.2 Consulta por parámetros	Aprobado

Nota. Elaboración propia.

Tabla 39

Caso de Prueba de Aceptación HU2_P01

Caso de Prueba de Aceptación			
Código	HU2_P01	Historia de Usuario	HU2
Nombre	Dejar todos los campos vacíos y comprobar que existan validaciones.		
Descripción	Se comprobará que las validaciones funcionen correctamente.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador. 		

Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Ingresar Categoría. • Se presionará el botón “Registrar”
Resultado esperado	Se presenta el mensaje de error y se indican los errores.
Evaluación de la prueba	Prueba satisfactoria.

Nota. Elaboración propia.

Tabla 40

Caso de Prueba de Aceptación HU2_P02

Caso de Prueba de Aceptación			
Código	HU2_P02	Historia de Usuario	HU2
Nombre	Introducir todos los datos correctos y comprobar que el sistema registre la información.		
Descripción	Se comprobará que se puedan ingresar nuevas categorías satisfactoriamente.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Ingresar Categoría. • Se ingresa el nombre de la nueva categoría. • Se presionará el botón “Registrar” 		
Resultado esperado	Se registrará la información de la categoría en la base de datos y se presentará el mensaje de buen trabajo.		
Evaluación	Prueba satisfactoria.		

Nota. Elaboración propia.

Tabla 41*Caso de Prueba de Aceptación HU2_P03*

Caso de Prueba de Aceptación			
Código	HU2_P03	Historia de Usuario	HU2
Nombre	Modificar una categoría y comprobar que se actualice.		
Descripción	Se comprobará que se pueda actualizar una categoría satisfactoriamente.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador. • Categoría registrada. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Administrar Categorías. • Se seleccionará una categoría al azar. • Se presionará el botón con ícono de modificar. • Se modificará el nombre de la categoría. • Se presionará el botón “Actualizar”. 		
Resultado esperado	Se actualizará la información de la categoría en la base de datos y se presentará el mensaje de buen trabajo.		
Evaluación de la prueba	Prueba satisfactoria.		

Nota. Elaboración propia.**Tabla 42***Caso de Prueba de Aceptación HU2_P04*

Caso de Prueba de Aceptación			
Código	HU2_P04	Historia de Usuario	HU2
Nombre	Eliminar una categoría.		
Descripción	Se comprobará que se pueda eliminar una categoría satisfactoriamente.		

Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador. • Categoría registrada.
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Administrar Categorías. • Se seleccionará una categoría al azar. • Se presionará el botón con ícono de eliminar. • Se presionará el botón “Confirmar”.
Resultado esperado	Se eliminará la información de la categoría en la base de datos y se presentará el mensaje de buen trabajo.
Evaluación de la prueba	Prueba satisfactoria.

Nota. Elaboración propia.

Tabla 43

Caso de Prueba de Aceptación HU3_P01

Caso de Prueba de Aceptación			
Código	HU3_P01	Historia de Usuario	HU3
Nombre	Introducir la identificación de un cliente existente y comprobar que el sistema cargue la información.		
Descripción	Se comprobará que se pueda cargar la información de un cliente existente mediante su búsqueda.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. • Cliente existente. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Ingresar Pedido. • Se seleccionará el tipo de documento “Cédula”. • Se ingresará el documento de un cliente existente. 		

	<ul style="list-style-type: none"> • Se presionará sobre el botón “Buscar”.
Resultado esperado	Se presenta el cliente encontrado y se cargará la información del mismo.
Evaluación de la prueba	Prueba satisfactoria.

Nota. Elaboración propia.

Tabla 44

Caso de Prueba de Aceptación HU3_P02

Caso de Prueba de Aceptación			
Código	HU3_P02	Historia de Usuario	HU3
Nombre	Introducir la identificación de un nuevo cliente y verificar que se pueda agregarlo.		
Descripción	Se comprobará que se puedan ingresar nuevos clientes desde la interfaz de pedidos.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Ingresar Pedido. • Se seleccionará el tipo de documento “Cédula”. • Se ingresará una cédula que no exista en el sistema. • Se presionará sobre el botón buscar. • El sistema presentará el mensaje de “Cliente no encontrado”. • Se presionará sobre el botón de “+”. • Se completará todos los campos satisfactoriamente. • Se presionará el botón “Registrar” 		

Resultado esperado	Se registrará la información del cliente en la base de datos y se cargará la información en el pedido.
Evaluación de la prueba	Prueba satisfactoria.

Nota. Elaboración propia.

Tabla 45

Caso de Prueba de Aceptación HU3_P03

Caso de Prueba de Aceptación			
Código	HU3_P03	Historia de Usuario	HU3
Nombre	Dejar todos los campos vacíos y comprobar que existan validaciones.		
Descripción	Se comprobará que las validaciones funcionen correctamente.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Ingresar Pedido. • Se presionará el botón “Agregar Pedido” 		
Resultado esperado	Se presenta el mensaje de error y se indican los errores.		
Evaluación de la prueba	Prueba satisfactoria.		

Nota. Elaboración propia.

Tabla 46

Caso de Prueba de Aceptación HU3_P04

Caso de Prueba de Aceptación			
Código	HU3_P04	Historia de Usuario	HU3

Nombre	Introducir todos los datos correctos y comprobar que el sistema registre la información.
Descripción	Se comprobará que se puedan ingresar nuevas categorías satisfactoriamente.
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. • Datos de un cliente.
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Ingresar Pedido. • Se completarán todos los campos correctamente. • Se presionará el botón “Agregar Pedido”
Resultado esperado	Se registrará la información del pedido en la base de datos y se presentará el mensaje de buen trabajo.
Evaluación de la prueba	Prueba satisfactoria.

Nota. Elaboración propia.

Tabla 47

Caso de Prueba de Aceptación HU3_P05

Caso de Prueba de Aceptación			
Código	HU3_P05	Historia de Usuario	HU3
Nombre	Validar que el stock de productos se actualice si el pedido se encuentra completado.		
Descripción	Se comprobará el stock del producto se actualiza si este cambia de estado a “Completado”.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. • Pedido pendiente registrado. 		

Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Administrar Pedidos. • Se seleccionará un pedido pendiente al azar. • Se presionará sobre el botón con ícono de “Modificar”. • Se modificará el estado a “Completado” • Se presionará el botón “Actualizar” • Se verifica que el stock de los productos haya cambiado.
Resultado esperado	Se actualizará la información del pedido y sus productos en la base de datos, y se presentará el mensaje de buen trabajo.
Evaluación de la prueba	Prueba satisfactoria.

Nota. Elaboración propia.

Tabla 48

Caso de Prueba de Aceptación HU3_P06

Caso de Prueba de Aceptación			
Código	HU3_P06	Historia de Usuario	HU3
Nombre	Eliminar un pedido.		
Descripción	Se comprobará que se puedan eliminar pedidos.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. • Pedido registrado. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Administrar Pedidos. • Se seleccionará un pedido pendiente al azar. • Se presionará sobre el botón con ícono de “eliminar”. • Se presionará sobre el botón “Confirmar”. 		

Resultado esperado	Se eliminará la información del pedido y se presentará el mensaje de buen trabajo.
Evaluación de la prueba	Prueba satisfactoria.

Nota. Elaboración propia.

5.3.3. Iteración 3

Tabla 49

Resumen de estados de Iteración 3

Historia de Usuario	Tarea	Estado
HU4	F4.1 Ingresar inventario	Aprobado
	F4.2 Modificar inventario	Aprobado
	F4.3 Eliminar inventario	Aprobado
	F4.4.1 Consulta general	Aprobado
	F4.4.2 Consulta por parámetros	Aprobado
HU5	F5.1 Ingresar empleados	Aprobado
	F5.2 Modificar empleados	Aprobado
	F5.3 Eliminar empleados	Aprobado
	F5.4.1 Consulta general	Aprobado
	F5.4.2 Consulta por parámetros	Aprobado

Nota. Elaboración propia.

Tabla 50*Caso de Prueba de Aceptación HU4_P01*

Caso de Prueba de Aceptación			
Código	HU4_P01	Historia de Usuario	HU4
Nombre	Dejar todos los campos vacíos y comprobar que existan validaciones.		
Descripción	Se comprobará que las validaciones funcionen correctamente.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Ingresar Producto. • Se presionará el botón "Registrar" 		
Resultado esperado	Se presenta el mensaje de error y se indican los errores.		
Evaluación de la prueba	Prueba satisfactoria.		

Nota. Elaboración propia.**Tabla 51***Caso de Prueba de Aceptación HU4_P02*

Caso de Prueba de Aceptación			
Código	HU4_P02	Historia de Usuario	HU4
Nombre	Introducir todos los datos correctos y comprobar que el sistema registre la información.		
Descripción	Se comprobará que se puedan ingresar nuevas categorías satisfactoriamente.		

Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado.
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Ingresar Producto. • Se completarán todos los campos correctamente. • Se presionará el botón “Registrar”
Resultado esperado	Se registrará la información del producto en la base de datos y se presentará el mensaje de buen trabajo.
Evaluación de la prueba	Prueba satisfactoria.

Nota. Elaboración propia.

Tabla 52

Caso de Prueba de Aceptación HU4_P03

Caso de Prueba de Aceptación			
Código	HU4_P03	Historia de Usuario	HU4
Nombre	Actualizar un producto.		
Descripción	Se comprobará que se puedan actualizar una categoría satisfactoriamente.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Administrar Inventario. • Se seleccionará un producto al azar. • Se presionará el botón con ícono de modificar. • Se modificarán los datos del producto. • Se presionará el botón “Actualizar”. 		

Resultado esperado	Se actualizará la información del producto en la base de datos y se presentará el mensaje de buen trabajo.
Evaluación de la prueba	Prueba satisfactoria.

Nota. Elaboración propia.

Tabla 53

Caso de Prueba de Aceptación HU4_P04

Caso de Prueba de Aceptación			
Código	HU4_P04	Historia de Usuario	HU4
Nombre	Buscar un producto mediante su SKU.		
Descripción	Se comprobará que se puedan actualizar una categoría satisfactoriamente.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. • Producto registrado. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Administrar Inventario. • Se seleccionará el filtro "SKU". • Se ingresará el SKU de un producto a buscar. 		
Resultado esperado	Se presentará el producto buscado.		
Evaluación de la prueba	Prueba satisfactoria.		

Nota. Elaboración propia.

Tabla 54*Caso de Prueba de Aceptación HU5_P01*

Caso de Prueba de Aceptación			
Código	HU5_P01	Historia de Usuario	HU5
Nombre	Introducir una cédula incorrecta y comprobar que se indique el error.		
Descripción	Se probará que se validen las cédulas de forma correcta.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Ingreso de Empleados. • Se selecciona el tipo de documento "Cédula". • Se introducirá una cédula inválida. 		
Resultado esperado	Se indica el error.		
Evaluación de la prueba	Prueba satisfactoria.		

Nota. Elaboración propia.**Tabla 55***Caso de Prueba de Aceptación HU5_P02*

Caso de Prueba de Aceptación			
Código	HU5_P02	Historia de Usuario	HU5
Nombre	Introducir un empleado ya existente, y comprobar que el sistema no permita agregarlo nuevamente.		
Descripción	Se probará que no es posible agregar empleados con el mismo documento de identidad.		

Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador. • Información de un empleado existente.
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Ingreso de Empleado. • Se selecciona el tipo de documento “Cédula”. • Se introducirá la cédula de un empleado existente. • Se completarán los demás campos. • Se presionará el botón “Registrar”.
Resultado esperado	Se presenta el mensaje de error.
Evaluación de la prueba	Prueba satisfactoria.

Nota. Elaboración propia.

Tabla 56

Caso de Prueba de Aceptación HU5_P03

Caso de Prueba de Aceptación			
Código	HU5_P03	Historia de Usuario	HU5
Nombre	Dejar todos los campos vacíos y comprobar que existan validaciones.		
Descripción	Se probará que las validaciones se encuentren funcionando.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Ingreso de Empleado. • Se presionará el botón “Registrar” 		
Resultado esperado	Se presenta el mensaje de error y se indican los errores.		

Evaluación de la prueba	Prueba satisfactoria.
--------------------------------	-----------------------

Nota. Elaboración propia.

Tabla 57

Caso de Prueba de Aceptación HU5_P04

Caso de Prueba de Aceptación			
Código	HU5_P04	Historia de Usuario	HU5
Nombre	Introducir todos los datos correctos y comprobar que el sistema registre la información.		
Descripción	Se comprobará que se puedan ingresar nuevos empleados satisfactoriamente.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Ingreso de Empleado. • Se completarán todos los campos de forma satisfactoria. • Se presionará el botón “Registrar” 		
Resultado esperado	Se registrará la información del empleado en la base de datos y se presentará el mensaje de buen trabajo.		
Evaluación de la prueba	Prueba satisfactoria.		

Nota. Elaboración propia.

Tabla 58*Caso de Prueba de Aceptación HU5_P05*

Caso de Prueba de Aceptación			
Código	HU5_P05	Historia de Usuario	HU5
Nombre	Eliminar un empleado.		
Descripción	Se comprobará que se puedan eliminar empleados satisfactoriamente.		
Condiciones de ejecución	<ul style="list-style-type: none">• Usuario con permisos de administrador o empleado.• Empleado registrado.		
Entrada / Pasos de ejecución	<ul style="list-style-type: none">• Se accederá a la ventana de Administrar Empleados.• Se seleccionará un empleado al azar.• Se presionará el botón con ícono de eliminar.• Se presionará el botón “Confirmar”		
Resultado esperado	Se eliminará la información del empleado en la base de datos y se presentará el mensaje de confirmación.		
Evaluación de la prueba	Prueba satisfactoria.		

Nota. Elaboración propia.

5.3.4. Iteración 4

Tabla 59

Resumen de estados de Iteración 4

Historia de Usuario	Tarea	Estado
HU6	Notificaciones mediante WhatsApp	Aprobado
HU7	Emisión de notas de venta.	Aprobado
HU8	Dashboard	Aprobado

Nota. Elaboración propia.

Tabla 60

Caso de Prueba de Aceptación HU6_P01

Caso de Prueba de Aceptación			
Código	HU6_P01	Historia de Usuario	HU6
Nombre	Actualizar el stock de un producto (bajo el mínimo) y comprobar que funcionen las alertas.		
Descripción	Se comprobará que las alertas mediante WhatsApp funcionen.		
Condiciones de ejecución	<ul style="list-style-type: none">• Usuario con permisos de administrador o empleado.• Producto registrado.		
Entrada / Pasos de ejecución	<ul style="list-style-type: none">• Se accederá a la ventana de Administrar Inventario.• Se seleccionará un producto al azar.• Se presionará sobre el botón de “Remover Stock (-)”.• Se reducirá su stock bajo el mínimo.• Se completarán los campos faltantes.• Se presionará el botón “Actualizar”.		

Resultado esperado	Se registrará la información en la base de datos, se presentará el mensaje de buen trabajo y se recibirá la alerta en WhatsApp.
Evaluación de la prueba	Prueba satisfactoria.

Nota. Elaboración propia.

Tabla 61

Caso de Prueba de Aceptación HU7_P01

Caso de Prueba de Aceptación			
Código	HU7_P01	Historia de Usuario	HU7
Nombre	Obtener la nota de venta de un pedido.		
Descripción	Se comprobará que los trabajadores puedan obtener una ayuda mediante un documento que tenga la información del pedido.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. • Pedido completado o pendiente. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Administrar Pedidos. • Se seleccionará un producto completado o pendiente al azar. • Se presionará el botón que contiene el icono de documento. 		
Resultado esperado	Se presentará la nota de venta del pedido seleccionado.		
Evaluación de la prueba	Prueba satisfactoria.		

Nota. Elaboración propia.

Tabla 62

Caso de Prueba de Aceptación HU8_P01

Caso de Prueba de Aceptación			
Código	HU8_P01	Historia de Usuario	HU8
Nombre	Comprobar que se presente una gráfica que muestre el número de pedidos con respecto al tiempo.		
Descripción	Se comprobará que se presente correctamente el resumen de forma general.		
Condiciones de ejecución	<ul style="list-style-type: none"> • Usuario con permisos de administrador o empleado. • Pedidos registrados. • Productos registrados. • Clientes registrados. 		
Entrada / Pasos de ejecución	<ul style="list-style-type: none"> • Se accederá a la ventana de Dashboard. • Se comprobará que la gráfica presentada sea correcta. 		
Resultado esperado	Se presentará una gráfica correcta.		
Evaluación de la prueba	Prueba satisfactoria.		

Nota. Elaboración propia.

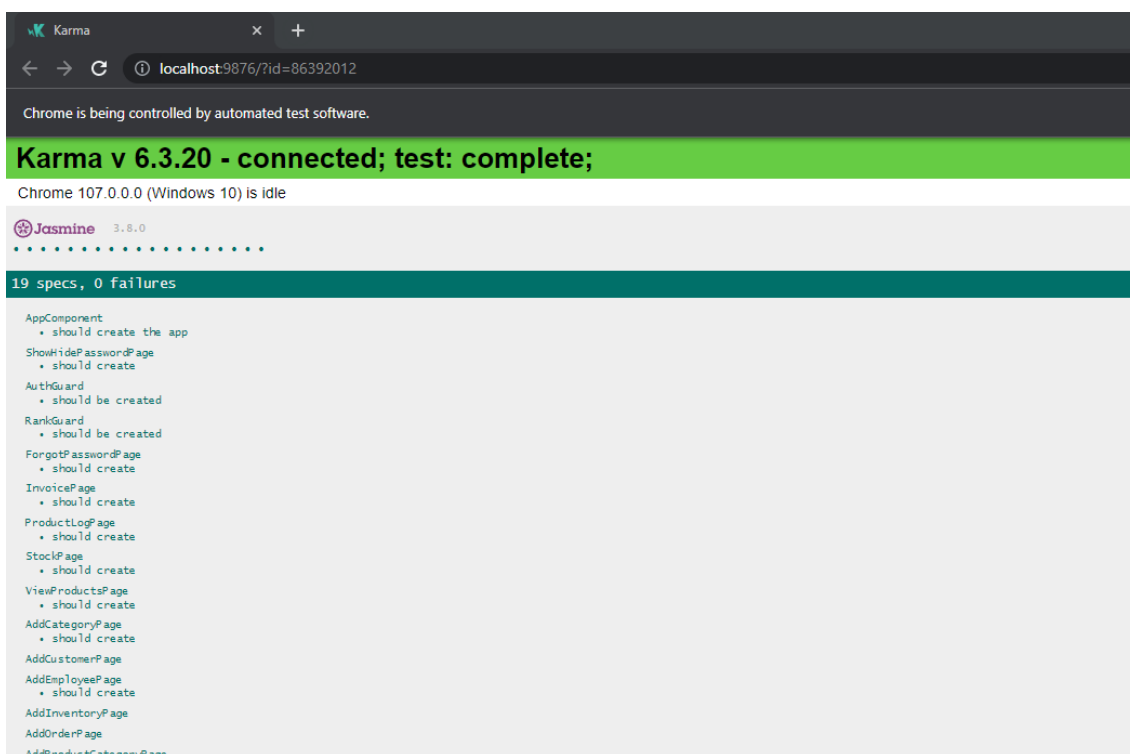
5.4. Jasmine

Jasmine es un marco de desarrollo que se encuentra orientado a poner en prueba el comportamiento del código en JavaScript de forma rápida y sencilla (Jasmine, 2022). Al utilizar e instalar Angular, la consola nos instala directamente todo lo necesario para poder utilizar el marco de pruebas "Jasmine". Mediante el comando "ng test", el cual debe ser ejecutado en la consola y en el directorio de la aplicación, realizaremos las pruebas de unidad, entrando en un modo de observación y evaluación ejecutando las pruebas de karma. En el presente trabajo se

utilizó Jasmine para validar que todos los componentes de la aplicación se creen y ejecuten con éxito. Esto es de suma importancia ya que, si un módulo no funciona, puede afectar el rendimiento de la aplicación sin que nos demos cuenta. A continuación, se adjunta una foto de las pruebas realizadas mediante este “framework” de pruebas.

Figura 35

Resultados de pruebas Karma de la aplicación.



Nota. El gráfico representa resultados satisfactorios obtenidos al ejecutar las pruebas karma mediante el uso de Jasmine en la aplicación. Elaboración propia.

CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES

6. Conclusiones y recomendaciones

6.1. Conclusiones

- El levantamiento de requerimientos funcionales y no funcionales de manera adecuada permitió un desarrollo eficiente de la aplicación al usar historias de usuario de la metodología XP, por lo que se obtuvo como resultado un producto elaborado en un tiempo relativamente corto.
- La correcta selección de una metodología mediante de la valoración de sus características aportó positivamente en el desarrollo del proyecto. La aplicación web fue desarrollada utilizando la metodología ágil programación extrema (XP), gracias a su ciclo de vida basado en iteraciones, se obtuvo un aplicativo que cumple con las expectativas del cliente.
- El uso de frameworks en el Front-end para el desarrollo de la aplicación web, aportó con librerías que incorporan clases, funciones y estilos que fueron consumidos en los diferentes componentes del proyecto. Las herramientas seleccionadas con la ayuda de distintas comparativas permiten que la aplicación pueda ser abierta desde cualquier tipo de dispositivo a través de un navegador web.
- El utilizar un Back-end como servicio (BaaS), en lugar de desarrollar un sistema propio, permitió cumplir con el cronograma dado que nos hubiera tomado más tiempo el desarrollar un Back-end por cuenta propia, por lo que, al momento de realizar un proyecto similar, es importante analizar todas las herramientas que se tiene con el objetivo de seleccionar la más conveniente.

6.2. Recomendaciones

- Se recomienda utilizar un Back-end como servicio cuando se cuente con tiempos cortos de implementación y se busque optimizar el código. Un ejemplo de ello puede ser el utilizado en el desarrollo de la aplicación, el cual es Firebase, ya que incluye una gran variedad de métodos que pueden ser consumidos para el registro de datos, validación de cuentas y subidas de archivos a la nube.
- Para líneas futuras, se recomienda la evaluación de un nuevo framework en el Back-end y también en la selección del motor gestor de base de datos, para realizar una comparativa de rendimientos y tiempos de desarrollo.
- Un módulo para tener en cuenta es la implementación del sistema de facturación electrónica, que no se tomó en cuenta en el presente proyecto debido a los plazos y las necesidades del negocio.

BIBLIOGRAFÍA

- Angular. (2022, Septiembre 25). *Introducción a la Documentación de Angular*.
<https://docs.angular.lat/docs>
- Calvopiña, A. (2022, Septiembre 14). Historia de Artemisa. (D. Barreno, Entrevistador)
- Farfán Jiménez, J. A. (2020, Diciembre 31). La implementación de un sistema automatizado reduce los tiempos de atención en los procesos aplicables a la ventanilla única de turismo en la Municipalidad Provincial del Callao. *Industrial Data*, 1-8.
<https://www.redalyc.org/journal/816/81665362003/81665362003.pdf>
- Firebase. (2022, Septiembre 26). *Firebase Pricing*. Google:
<https://firebase.google.com/pricing?hl=es>
- Flutter. (2022, Septiembre 25). *Firebase*. Flutter: <https://esflutter.dev/docs/development/data-and-backend/firebase>
- Ionos. (2020, Marzo 16). *SGBD: Introducción al sistema gestor de base de datos*. Digital Guide Ionos: <https://www.ionos.es/digitalguide/hosting/cuestiones-tecnicas/sistema-gestor-de-base-de-datos-sgbd/>
- Jasmine. (2022, Noviembre 4). *Jasmine Documentation*. <https://jasmine.github.io/>
- Laravel. (s.f.). *The PHP Framework for Web Artisans*. Laravel: <https://laravel.com/>
- León Soberón, J. J. (2020). *Análisis comparativo de sistemas gestores de bases de datos postgresql y mysql en procesos crud*. Chiclayo.
- MariaDB Foundation. (2022, Septiembre 25). *MariaDB Server*. MariaDB: <https://mariadb.org/es/>

- Martins, J. (2022, Julio 18). *¿Qué es la metodología Kanban y cómo funciona?* Asana:
<https://asana.com/es/resources/what-is-kanban>
- Martins, J. (2022, Agosto 17). *Qué es Scrum y cómo aplicarlo en gestión de proyectos.* Asana:
<https://asana.com/es/resources/what-is-scrum>
- Microsoft. (2022, Septiembre 18). *Estilo de arquitectura de microservicios.*
<https://learn.microsoft.com/es-es/azure/architecture/guide/architecture-styles/microservices>
- Microsoft. (s.f.). *Conceptos básicos sobre bases de datos.* <https://support.microsoft.com/es-es/office/conceptos-b%C3%A1sicos-sobre-bases-de-datos-a849ac16-07c7-4a31-9948-3c8c94a7c204>
- Mozilla. (2022, 23 Septiembre). *Introducción a Django.* Aprende sobre desarrollo web en MDN:
<https://developer.mozilla.org/es/docs/Learn/Server-side/Django/Introduction>
- Mozilla. (2022, Septiembre 17). *MVC.* Glosario:
<https://developer.mozilla.org/es/docs/Glossary/MVC>
- Novak, C. (2018, Abril 18). *App vs Web App.* Mozilla Explains: <https://blog.mozilla.org/en/internet-culture/mozilla-explains/no-judgment-digital-definitions-app-vs-web-app/>
- Preciado Rodríguez, A. J., Valles Coral, M. A., & Lévano Rodríguez, D. (2021). *Importancia del uso de sistemas de información en la.* Revista Cubana de Informática Médica.
- Raeburn, A. (2022, Mayo 16). *¿Qué es la programación extrema (XP)?* Asana:
<https://asana.com/es/resources/extreme-programming-xp>
- React. (2022, 25 Septiembre). *Una biblioteca de JavaScript para construir interfaces de usuario.*
<https://es.reactjs.org/>

Sandoval Ruiz, D. E. (2019, Diciembre 30). *Automatización de los procesos de las empresas a través de Software*. <https://fundacionavenir.net/revista/index.php/avenir/article/view/87>

Shetty, J., Dash, D., Kumar, A., & C, G. (2020). Review Paper on Web Frameworks, Databases and Web Stacks. *International Research Journal of Engineering and Technology (IRJET)*, 1-5.

Universidad de Alicante. (2022, Septiembre 17). *Modelo Vista Controlador (MVC)*. Universitat D'Alcant: <https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html>

Vega, S., & Fuertes, J. L. (2020, Junio). *Sitio web "responsive" con acceso a información remota*. <https://oa.upm.es/63125/>

Vue.js. (2022, Septiembre 25). *Introducción*. Vue.js: <https://es.vuejs.org/v2/guide/>

WhatsApp. (2022, Agosto 25). *WhatsApp Business Platform*. https://developers.facebook.com/docs/whatsapp/?locale=es_LA

GLOSARIO DE TÉRMINOS

A

API

Interfaz de programación de aplicaciones, es un conjunto de funciones y procedimientos que permiten la comunicación e integración de distintas aplicaciones., 20

aplicaciones web progresivas

Son aplicaciones que cargan a través de la web y pueden ser ejecutadas desde cualquier plataforma que use un navegador compatible, 9

B

BaaS

Back-end como servicio, 104

BDD

bases de datos, 42, 50, 57, 64, 71

C

código abierto

Software que puede ser utilizado, modificado y redistribuido libremente., 20

Composer

Sistema que provee estándares necesarios para la gestión de librerías de PHP, 28

CRUD

Acrónimo de crear, leer, actualizar y borrar, 28

F

Full-stack

Desarrollo tanto de front-end, como de back-end, 28

K

karma

es un corredor de pruebas creado por el equipo de AngularJS, 102



N

NodeJS

Entorno de ejecución multiplataforma, 33

ANEXOS

Anexo A. Acta de Reunión

Quito, 18 de agosto de 2022	
ACTA N° 1	
Reunión informativa sobre el trabajo de titulación "DISEÑO Y DESARROLLO DE UNA APLICACIÓN WEB QUE PERMITA ADMINISTRAR CLIENTES, INVENTARIO Y PEDIDOS. CASO DE ESTUDIO: ARTEMISA."	
La presente acta muestra el desarrollo de la reunión llevada a cabo el 18 de agosto de 2022, en la cual se presentaron los siguientes ítems:	
<ol style="list-style-type: none">1. Presentación del trabajo.2. Contextualización de la empresa.3. Explicación de los procesos y necesidades.4. Otros.	
De esta forma, se obtuvieron los datos necesarios para el desarrollo del trabajo mediante la explicación de la Ingeniera María José Calvopiña, representante de proyectos de Artemisa.	
 _____ Ing. María José Calvopiña Representante y dueña de Artemisa	 _____ David Barreno M. Estudiante de Ingeniería en Sistemas de Información

Anexo B. Historia de Usuario 1

Historia de Usuario	
ID: HU1	Usuario: Administrador y empleado
Iteración asignada: 1	Prioridad: Alta
Nombre: Administrar a los clientes	
Descripción: Quiero que los administradores y empleados puedan agregar, modificar y quitar viejos y nuevos clientes, también poder consultarlos mediante su número de	

identificación, nombre o correo para obtener de forma más rápida toda la información. Además, quiero que mis clientes puedan ser clasificados en una categoría simple que yo pueda agregar, ya sea uniformes, ropa, etc.

Pruebas de aceptación
<ul style="list-style-type: none"> • Introducir una cédula incorrecta y comprobar que se indique el error. • Introducir un usuario ya existente al agregar, y comprobar que el sistema no permita agregarlo nuevamente. • Dejar todos los campos vacíos y comprobar que existan validaciones. • Introducir todos los datos correctos y comprobar que el sistema registre la información. • Modificar un cliente y comprobar que se actualice. • Eliminar un cliente.

Anexo C. Historia de Usuario 2

Historia de Usuario	
ID: HU2	Usuario: Administrador
Iteración asignada: 2	Prioridad: Media
Nombre: Administrar categorías de los clientes	
<p>Descripción:</p> <p>Quiero que los administradores puedan agregar, modificar y eliminar categorías de clientes, para que puedan ser utilizadas en la parte de manejar clientes. Solo es necesario el nombre de la categoría.</p>	

Pruebas de aceptación
<ul style="list-style-type: none"> • Dejar todos los campos vacíos y comprobar que existan validaciones. • Introducir todos los datos correctos y comprobar que el sistema registre la información. • Modificar una categoría y comprobar que se actualice. • Eliminar una categoría.

Anexo D. Historia de Usuario 3

Historia de Usuario	
ID: HU3	Usuario: Administrador y empleado
Iteración asignada: 2	Prioridad: Alta
Nombre: Administrar pedidos	
Descripción: Quiero que los administradores y empleados puedan agregar nuevos pedidos, en estos podrán ingresar el documento del cliente y si existe, se cargará la información, caso contrario se tendrá que agregar uno nuevo. Ellos podrán modificar el estado de los pedidos pendientes, y podrán borrar cualquier pedido sin importar su estado. En los pedidos se podrá seleccionar los productos y luego indicar la cantidad para que carguen los subtotales y los totales. El stock de los productos se deberá actualizar cuando el pedido se encuentre en estado "Completado".	

Pruebas de aceptación
<ul style="list-style-type: none">• Introducir la identificación de un cliente existente y comprobar que el sistema cargue la información.• Introducir la identificación de un nuevo cliente y verificar que se pueda agregarlo.• Dejar los campos vacíos y comprobar que existan validaciones.• Introducir todos los datos correctos y comprobar que el sistema registre la información.• Validar que el stock de productos se actualice si el pedido se encuentra completado.• Eliminar un pedido.

Anexo E. Historia de Usuario 4

Historia de Usuario	
ID: HU4	Usuario: Administrador y empleado
Iteración asignada: 3	Prioridad: Alta

Nombre: Administrar el inventario
<p>Descripción:</p> <p>Quiero que los administradores y empleados puedan agregar, modificar y quitar productos. Podrán modificar el stock del producto, ya sea añadiendo más unidades o quitando y podrán ingresar el costo total o una breve descripción para tener un mejor manejo de inventario. En cada producto será necesario subir una imagen de este para tener la referencia. Además, se deberá incluir un stock mínimo y medio para que se pinten los productos dependiendo del stock actual.</p>

Pruebas de aceptación
<ul style="list-style-type: none"> • Dejar todos los campos vacíos y comprobar que existan validaciones. • Introducir todos los datos correctos y comprobar que el sistema registre la información. • Actualizar un producto. • Buscar un producto mediante su SKU.

Anexo F. Historia de Usuario 5

Historia de Usuario	
ID: HU5	Usuario: Administrador
Iteración asignada: 3	Prioridad: Alta
Nombre: Administrar a los empleados	
<p>Descripción:</p> <p>Quiero que los administradores puedan registrar nuevos empleados, pero también modificar y eliminar a los que ya se encuentren registrados. Se deberá agregar una forma de generar una contraseña y también se incluirá la selección del rango del empleado en su registro y actualización. También quiero que se puedan consultar a los empleados mediante su identificación, nombre o apellido.</p>	

Pruebas de aceptación
<ul style="list-style-type: none"> • Introducir una cédula incorrecta y comprobar que se indique el error.

- Introducir un empleado ya existente al agregar, y comprobar que el sistema no permita agregarlo nuevamente.
- Dejar todos los campos vacíos y comprobar que existan validaciones.
- Introducir todos los datos correctos y comprobar que el sistema registre la información.
- Eliminar un empleado.

Anexo G. Historia de Usuario 6

Historia de Usuario	
ID: HU6	Usuario: Administrador
Iteración asignada: 4	Prioridad: Baja
Nombre: Notificaciones mediante WhatsApp	
Descripción: Quiero que me lleguen alertas a mi WhatsApp en función al stock del producto, siempre y cuando sea igual o menor al stock mínimo.	

Pruebas de aceptación
<ul style="list-style-type: none"> • Actualizar el stock de un producto (bajo el mínimo) y comprobar que funcionen las alertas.

Anexo H. Historia de Usuario 7

Historia de Usuario	
ID: HU7	Usuario: Administrador y empleado
Iteración asignada: 4	Prioridad: Media
Nombre: Emisión de notas de venta.	
Descripción: Quiero que los administradores y empleados puedan tener una ayuda mediante un documento que tenga la información del pedido (simulando una nota de venta), esto servirá únicamente para completar las notas de venta de forma manual.	

Pruebas de aceptación
<ul style="list-style-type: none"> • Obtener la nota de venta de un pedido.

Anexo I. Historia de Usuario 8

Historia de Usuario	
ID: HU8	Usuario: Administrador y empleado
Iteración asignada: 4	Prioridad: Alta
Nombre: Dashboard	
<p>Descripción:</p> <p>Quiero que los administradores y empleados puedan ver un conteo de los productos con stock bajo, medio y normal, también de los clientes nuevos en el mes, año y el total, y finalmente los pedidos, ya sean pendientes, completados, cancelados y el total. También quiero poder ver una pequeña gráfica con los pedidos que se han registrado por días.</p>	

Pruebas de aceptación
<ul style="list-style-type: none"> • Comprobar que se presente una gráfica que muestre el número de pedidos con respecto al tiempo.

Anexo J. Certificado de recepción del aplicativo web

Quito, 04 de agosto de 2022



ARTEMISA MODA EC

*San Isidro del Inca – N53E y E19
Quito – Ecuador*

CERTIFICADO

Yo, Ing. María José Calvopiña Orellana con C.I. 1725094146 como representante de Artemisa Moda EC.

Certifico,

Que el estudiante DAVID ANDRÉS BARRENO MONTESDEOCA con C.I. 1719169904 de la carrera de Ingeniería en Sistemas de Información de la Pontificia Universidad Católica del Ecuador, ha culminado satisfactoriamente el aplicativo web bajo los requerimientos y especificaciones descritas con anterioridad. Aprovecho la presente para felicitar y agradecer por el trabajo entregado.

Ing. María José Calvopiña Orellana
Representante de Artemisa

Anexo K. Captura de pantalla de la página de inicio de sesión



¿Olvidaste tu contraseña? Haz click [aquí](#).

Anexo L. Captura de pantalla del Dashboard

Dashboard
Resumen general de todos los pedidos y clientes.

Selecciona el rango de fechas
8/1/2022 - 11/15/2022
MM/DD/YYYY - MM/DD/YYYY

Últimos 7 días Este mes Último mes Últimos 3 meses Últimos 6 meses Último año

Pedidos

1 Pendientes	2 Completados	1 Cancelados	4 Total
-----------------	------------------	-----------------	------------

Nro. de Pedidos

Fecha	Nro. de Pedidos
2022-08-01	0
2022-08-02	0
2022-08-03	0
2022-08-04	0
2022-08-05	0
2022-08-06	0
2022-08-07	0
2022-08-08	0
2022-08-09	0
2022-08-10	0
2022-08-11	0
2022-08-12	0
2022-08-13	0
2022-08-14	0
2022-08-15	0
2022-08-16	0
2022-08-17	0
2022-08-18	0
2022-08-19	0
2022-08-20	0
2022-08-21	0
2022-08-22	0
2022-08-23	0
2022-08-24	0
2022-08-25	0
2022-08-26	0
2022-08-27	0
2022-08-28	0
2022-08-29	0
2022-08-30	0
2022-08-31	0
2022-09-01	0
2022-09-02	0
2022-09-03	0
2022-09-04	0
2022-09-05	0
2022-09-06	0
2022-09-07	0
2022-09-08	0
2022-09-09	0
2022-09-10	0
2022-09-11	0
2022-09-12	0
2022-09-13	0
2022-09-14	0
2022-09-15	0
2022-09-16	0
2022-09-17	0
2022-09-18	0
2022-09-19	0
2022-09-20	0
2022-09-21	0
2022-09-22	0
2022-09-23	0
2022-09-24	0
2022-09-25	0
2022-09-26	0
2022-09-27	0
2022-09-28	0
2022-09-29	0
2022-09-30	0
2022-10-01	0
2022-10-02	0
2022-10-03	0
2022-10-04	0
2022-10-05	0
2022-10-06	0
2022-10-07	0
2022-10-08	0
2022-10-09	0
2022-10-10	0
2022-10-11	0
2022-10-12	0
2022-10-13	0
2022-10-14	0
2022-10-15	0
2022-10-16	0
2022-10-17	0
2022-10-18	0
2022-10-19	0
2022-10-20	0
2022-10-21	0
2022-10-22	0
2022-10-23	0
2022-10-24	0
2022-10-25	0
2022-10-26	0
2022-10-27	0
2022-10-28	0
2022-10-29	0
2022-10-30	0
2022-10-31	0
2022-11-01	0
2022-11-02	0
2022-11-03	0
2022-11-04	0
2022-11-05	0
2022-11-06	0
2022-11-07	0
2022-11-08	0
2022-11-09	0
2022-11-10	0
2022-11-11	0
2022-11-12	0
2022-11-13	0
2022-11-14	0
2022-11-15	4