

**UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA**



PROYECTO DE CURSO

**SISTEMA WEB DE CONTROL DE PEDIDOS Y VENTA
CASO: EMPRESA DE HUEVOS "GALLINA FELIZ"**

INTEGRANTES:	• Alvarado Silva Oscar Ignacio	3
	• Choque Huayta Carla Shecid	12
	• Cocarico Nina Omar Anton	13
	• Gutierrez Morales Betzabe	24
	• Villca Palma Jhessica Cora	53

DOCENTE: Lic. Gladys Vargas

MATERIA: Ingeniería de Software I

LA PAZ - BOLIVIA

2025

ÍNDICE

CAPÍTULO I

MARCO INTRODUCTORIO

1. ANTECEDENTES.....	4
1.1. ANTECEDENTES DE LA INSTITUCIÓN.....	4
2. PLANTEAMIENTO DEL PROBLEMA.....	4
2.1. PROBLEMA CENTRAL.....	4
2.2. PROBLEMAS SECUNDARIOS.....	5
2.2.1. Dificultades en la gestión de pedidos:.....	5
2.2.2. Control de inventario limitado:.....	5
2.2.3. Generación de reportes limitada:.....	5
2.2.4. Falta de visibilidad sobre el desempeño de los clientes:.....	5
3. DEFINICIÓN DEL OBJETIVO.....	6
3.1. OBJETIVO GENERAL.....	6
3.2. OBJETIVOS ESPECÍFICOS.....	6
4. JUSTIFICACIÓN.....	7
4.1. JUSTIFICACIÓN ECONÓMICA.....	7
4.2. JUSTIFICACIÓN SOCIAL.....	7
4.3. JUSTIFICACIÓN TECNOLÓGICA.....	7

CAPÍTULO II

MARCO TEÓRICO

5. METODOLOGÍA.....	8
6. INGENIERÍA DE SOFTWARE.....	9
6.1. ¿QUÉ ES INGENIERÍA DE SOFTWARE?.....	9
6.2. CICLO DE VIDA DEL SOFTWARE.....	10
6.3. MODELOS DE PROCESO DE SOFTWARE.....	11
6.4. MODELOS DE DESARROLLO DE SOFTWARE.....	11
6.5. MÉTODOS DE DESARROLLO ÁGIL.....	12
6.6. METODOLOGÍA SCRUM.....	13
6.6.1. OBJETIVOS.....	13
6.6.2. CARACTERÍSTICAS.....	13
6.6.3. ELEMENTOS.....	14
6.6.4. EVENTOS.....	15
6.6.5. ESTRUCTURA.....	15
6.6.6. DISCIPLINA.....	16
6.6.7. TAREAS.....	16
6.7. CARACTERÍSTICAS DE UNA APLICACIÓN WEB.....	16
6.8. PROPUESTA DE VALOR : OPTIMIZACIÓN DE RUTAS.....	17

6.8.1. TEORÍA DEL MODELO MATEMÁTICO DE DIJKSTRA.....	17
6.8.2. MODELO MATEMÁTICO DE DIJKSTRA.....	17
6.8.3. APLICACIÓN A UN SISTEMA DE VENTAS Y PEDIDOS DE HUEVOS.....	19
6.8.4. VENTAJAS DE APLICAR DIJKSTRA EN ESTE SISTEMA.....	20
6.8.5. LIMITACIONES.....	20

CAPÍTULO III

MARCO APlicativo

Metodología Scrum por Fases.....	21
Adaptación al Contexto de "Gallina Feliz".....	22

ANEXOS

ANEXOS.....	24
MODELO ENTIDAD RELACIÓN.....	24
MODELO RELACIONAL.....	24
DICCCIONARIO DE DATOS (USANDO EL GESTOR POSTGRES).....	25
LISTA DE TAREAS PARA LAS FASES DE CASCADA.....	36
PREGUNTAS DE OBSERVACIÓN Y ENTREVISTA.....	37
CALENDARIZACIÓN PARA EL DESARROLLO DEL SISTEMA (SCRUM).....	42
DIAGRAMAS UML.....	42
• Diagrama de Componentes.....	43
• Diagrama de Actividad.....	44
• Diagrama Casos de Uso.....	45
◦ Diagrama de Casos de Uso General.....	45
◦ Diagrama de Casos de Uso Gestión de Usuario.....	45
◦ Diagrama de Casos de Uso Gestión de Productos.....	45
◦ Diagrama de Casos de Uso Gestión de Pedidos.....	45
◦ Diagrama de Casos de Uso Gestión para Realizar una Venta.....	46
• Diagrama de Interacción.....	46
• Diagrama de secuencia.....	47
TAREAS DE MANTENIMIENTO Y PRUEBA.....	47
Mantenimiento Adaptativo.....	47
Mantenimiento Correctivo.....	48
Mantenimiento Perfectivo.....	49
Mantenimiento Preventivo.....	50
Pruebas Funcionales.....	50
Pruebas No Funcionales.....	51
Pruebas de Mantenimiento.....	52
Pruebas Automatizadas.....	52
Pruebas Manuales.....	52

Ejemplos Específicos.....	53
MAPEO DEL SISTEMA.....	53
PANTALLAS MUERTAS DEL SISTEMA.....	55

CAPÍTULO I

MARCO INTRODUCTORIO.

1. ANTECEDENTES

1.1. ANTECEDENTES DE LA INSTITUCIÓN

La microempresa distribuidora "Gallina Feliz", con más de ocho años de experiencia en el mercado de La Paz y El Alto, se ha consolidado como un proveedor confiable de huevos al por mayor. Su compromiso con la calidad y la atención al cliente le ha permitido establecer una sólida base de consumidores, desde pequeñas tiendas hasta grandes supermercados.

Sin embargo, a pesar de su éxito, "Gallina Feliz" enfrenta desafíos en la gestión de sus operaciones diarias. Actualmente, la empresa depende en gran medida de métodos manuales para llevar el control de sus inventarios, procesar pedidos y generar reportes. Esta dependencia ha generado ineficiencias en la gestión, dificultando la toma de decisiones oportunas y la adaptación a las demandas cambiantes del mercado.

2. PLANTEAMIENTO DEL PROBLEMA

2.1. PROBLEMA CENTRAL

La microempresa "Gallina Feliz" enfrenta dificultades significativas en la gestión de sus operaciones de venta y distribución de huevos debido a la falta de un sistema informático integrado. Actualmente, los procesos de toma de pedidos, control de inventario y seguimiento de clientes se realizan de

manera manual, lo que genera ineficiencias, errores y una limitada capacidad para tomar decisiones basadas en datos precisos.

2.2. PROBLEMAS SECUNDARIOS

2.2.1. Dificultades en la gestión de pedidos:

- 2.2.1.1. Procesamiento manual de pedidos, lo que genera retrasos en la entrega.
- 2.2.1.2. Pérdida de información y errores en la toma de pedidos.
- 2.2.1.3. Dificultad para realizar seguimiento del estado de los pedidos.

2.2.2. Control de inventario limitado:

- 2.2.2.1. Dificultad para llevar un registro preciso de los niveles de inventario.
- 2.2.2.2. Riesgo de faltantes o excedentes de producto.
- 2.2.2.3. Dificultad para realizar pronósticos de demanda.

2.2.3. Generación de reportes limitada:

- 2.2.3.1. Dificultad para generar reportes de ventas, clientes y productos de manera rápida y precisa.
- 2.2.3.2. Limitada capacidad para analizar datos históricos y tomar decisiones basadas en evidencia.

2.2.4. Falta de visibilidad sobre el desempeño de los clientes:

- 2.2.4.1. Dificultad para identificar a los clientes más rentables y segmentar el mercado.
- 2.2.4.2. Pérdida de oportunidades de venta cruzada y upselling.

3. DEFINICIÓN DEL OBJETIVO

3.1. OBJETIVO GENERAL

Desarrollar e implementar un sistema de información que permita a "Gallina Feliz" automatizar y optimizar sus procesos de venta y distribución, mejorando la eficiencia, la precisión y la toma de decisiones.

3.2. OBJETIVOS ESPECÍFICOS

- 3.2.1. Automatizar la toma y procesamiento de pedidos:** Implementar un sistema en línea o móvil que permita a los clientes realizar pedidos de manera fácil y rápida.
- 3.2.2. Gestionar el inventario de manera eficiente:** Implementar un sistema de gestión de inventario que permita llevar un control preciso de los niveles de stock, realizar seguimiento de las entradas y salidas de productos y generar alertas de reposición.
- 3.2.3. Mejorar la gestión de clientes:** Implementar un CRM para almacenar información de los clientes, historial de compras y preferencias, y facilitar la segmentación y personalización de las ofertas.
- 3.2.4. Generar reportes personalizados:** Desarrollar reportes que permitan analizar las ventas por producto, cliente y región, identificar tendencias y evaluar el desempeño de la empresa.
- 3.2.5. Facilitar la colaboración entre los diferentes departamentos:** Integrar el sistema con otros sistemas existentes, como contabilidad y finanzas, para mejorar la comunicación y la colaboración entre los diferentes departamentos.

4. JUSTIFICACIÓN

4.1. JUSTIFICACIÓN ECONÓMICA

La implementación de un sistema de gestión de pedidos y ventas permitirá a "Gallina Feliz" reducir costos operativos, mejorar la eficiencia en la gestión de inventarios y aumentar la rentabilidad. Además, al mejorar la satisfacción del cliente y fidelizar a los clientes existentes, se generarán mayores ingresos a largo plazo.

4.2. JUSTIFICACIÓN SOCIAL

El sistema beneficiará a los empleados de "Gallina Feliz" al automatizar tareas repetitivas y liberar tiempo para que puedan enfocarse en actividades de mayor valor agregado. Además, permitirá mejorar las condiciones de trabajo y aumentar la satisfacción laboral.

4.3. JUSTIFICACIÓN TECNOLÓGICA

La tecnología actual ofrece una amplia gama de soluciones para la gestión de pequeñas y medianas empresas, como sistemas de gestión de relaciones con los clientes (CRM), sistemas de gestión de inventario (ERP) y plataformas de comercio electrónico. La implementación de un sistema de gestión integral permitirá a "Gallina Feliz" aprovechar las ventajas de las nuevas tecnologías y mejorar su competitividad en el mercado.

CAPÍTULO II

MARCO TEÓRICO.

5. METODOLOGÍA

El método de investigación empleado en este proyecto será el método científico, que nos ayudará a descubrir propiedades del objeto de estudio. El tipo de investigación empleado en este proyecto será la investigación descriptiva, la cual ayudará a obtener información relevante y de resultados claros y precisos.

La metodología de ingeniería de software a partir de la cual se desarrollará el presente proyecto será SCRUM, una metodología ágil que permite una alta adaptabilidad a las necesidades específicas de un proyecto. Las fases son:

- **Fase de Planificación del Sprint**, donde se define el propósito, objetivos y las tareas del proyecto.
- **Fase de Ejecución del Sprint**, en esta fase se realiza el desarrollo o codificación del sistema en incrementos.
- **Fase de Revisión del Sprint**, donde se entrega el producto al cliente y se evalúa la funcionalidad y desempeño del entregable.
- **Fase de Retrospectiva del Sprint**, en la cual se identifican las mejoras para futuros sprints y se ajusta el proceso de desarrollo.

La herramienta de modelado de datos basada en UML (UWE) consistirá en aplicar la metodología de análisis y diseño orientado a objetos al desarrollo de aplicaciones web. Usando UML, podemos ver una página como un objeto, lo que nos permitirá establecer los métodos y atributos de una página web.

Las tecnologías que se utilizarán serán:

- **Servidor Apache.**
- **PHP**, un lenguaje de programación diseñado originalmente para la creación de páginas web dinámicas.
- **MySQL**, para la construcción de la base de datos, permitiendo crear y administrar la base de datos, los usuarios y los permisos, así como diseñar y ejecutar las instrucciones SQL.

6. INGENIERÍA DE SOFTWARE

6.1. ¿QUÉ ES INGENIERÍA DE SOFTWARE?

Según El IEEE ingeniería de software se define como, Ingeniería es la aplicación de un método sistemático, estructurado y cuantificable a estructuras, máquinas, productos, sistemas o procesos. Ingeniería del software es la aplicación de un método sistemático, estructurado y cuantificable al desarrollo, operación y mantenimiento de software.

Según Bauer, 1972 define ingeniería de software como, la ingeniería de software es el establecimiento y uso de sólidos principios de ingeniería y buenas prácticas de gestión, así como la evolución de herramientas y métodos aplicables y su uso cuando sea apropiado para obtener, dentro de las limitaciones de recursos existentes, software que sea de alta calidad en un sentido explícitamente definido.

Para su mayor comprensión, la ingeniería de software es una disciplina de la ingeniería que comprende todos aspectos de la producción de software desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento, en los productos se aplican teorías, métodos y herramientas donde sean convenientes, pero las utilizan de forma selectiva y siempre tratando de descubrir soluciones a los problemas, aun cuando no existan teorías y métodos aplicables para resolver.

6.2. CICLO DE VIDA DEL SOFTWARE

El software no solo comprende los procesos técnicos de desarrollo de software, sino también con actividades tales como la gestión de proyectos de software y el desarrollo de 13 herramientas, métodos y teorías de apoyo a la producción de software, con estas características es una forma de producir software de alta calidad.

Para producir un software de alta calidad el desarrollo exige un enfoque secuencial a lo largo de su vida. Abarca las siguientes actividades:

- Análisis del Sistema: El Software es siempre parte de un sistema mayor, por tanto se comienza estableciendo las entidades, roles, funciones etc. de los que intervienen en el sistema, se identifican los requisitos del sistema.
- Análisis de Requisitos: Proceso de recopilación de los requisitos específicamente del software. El analista debe comprender el ámbito de la información, la función, el rendimiento y las interfaces del software.
- Diseño: Traduce los requisitos en una representación de software que pueda ser codificada.
- Codificación: Traducción del diseño en código fuente escrito en un lenguaje de programación.
- Prueba: Verificación de que las funciones del software producen los resultados que realmente se requieren.
- Mantenimiento: El mantenimiento aplica cada uno de los pasos precedentes para implementar los cambios que con el tiempo indudablemente sufrirá el software.

6.3. MODELOS DE PROCESO DE SOFTWARE

Los Modelos de Proceso de Software se los puede definir como una descripción simplificada de un proceso del software que presenta una visión de ese proceso. Los modelos de proceso de software, pueden incluir actividades que son parte de los procesos, productos de software y el papel de las personas involucradas en la ingeniería de software. La mayor parte de los modelos de procesos de software se basa en uno de los tres modelos generales o paradigmas de desarrollo de software:

- Enfoque en cascada: Considera las actividades anteriores y las representa con fases de procesos separados, tales como la especificación de requerimientos, el diseño de software, la implementación, las pruebas etc.
- Desarrollo iterativo: Este enfoque entrelaza las actividades de especificación, desarrollo y validación. Un sistema inicial se desarrolla rápidamente a partir de especificaciones muy abstractas.
- Modelo Prototipado: Un modelo sirve de prototipo para la construcción del sistema final.
- Transformación Formal: Un modelo matemático del sistema se transforma formalmente en la implementación.
- Desarrollo basado en Reutilización: El sistema es ensamblado a partir de componentes existentes.

6.4. MODELOS DE DESARROLLO DE SOFTWARE

Una parte importante de la ingeniería de software es el desarrollo de metodologías. Donde se mencionan: Metodologías Estructurada, Metodologías Orientada a Objetos, Metodologías Tradicionales y Metodología Ágiles.

- Metodología Estructurada: Las metodologías estructuradas se basan en la estructuración y descomposición funcional de problemas en unidades más

pequeñas interrelacionadas entre sí. Representan los procesos, flujos y estructuras de datos, de una manera jerárquica y ven el sistema como entradas-proceso-salidas.

- Metodología Orientada a Objetos: Las metodologías orientadas a objetos modelan el sistema examinando el dominio del problema como un conjunto de objetos que interactúan entre sí.
- Metodología Tradicional: Las metodologías tradicionales imponen una disciplina de trabajo sobre el proceso de desarrollo del software. Se centran especialmente en el control del proceso, con definición de roles, actividades, artefactos, herramientas y notaciones para el modelado y documentación detallada. Las metodologías tradicionales no se adaptan adecuadamente a los cambios.
- Metodología Ágil: Los procesos ágiles trabajan con requisitos desconocidos o variables. Si no existen requisitos estables, no existe una gran posibilidad de tener un diseño estable y de seguir un proceso totalmente planificado, que no vaya a variar ni en tiempo ni en dinero.

6.5. MÉTODOS DE DESARROLLO ÁGIL

Los métodos de desarrollo ágil trabajan con requisitos desconocidos o variables. Por tanto, el objetivo de los equipos es desarrollar software deberán responder rápidamente a los cambios que puedan surgir a lo largo del proyecto.

Se pretendía ofrecer una alternativa a los procesos de desarrollo de software tradicionales, caracterizados por ser rígidos y dirigidos por la documentación que se genera en cada una de las actividades desarrolladas.

A continuación se describe algunos de los métodos de desarrollo ágil:

- XP, Programación Extrema: Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo.
- SCRUM: Es una metodología ágil que consiste en asignación de tareas diarias basado en reuniones rápidas y control de la evolución de los procesos. Se puede llevar un seguimiento de las tareas y saber en qué puntos existe debilidad.
- OpenUP: Es un proceso unificado ligero, iterativo, incremental, extensible y ágil. Tiene como objetivo disminuir los riesgos con una identificación temprana; y es aplicable a un conjunto amplio de plataformas y aplicaciones de desarrollo. Este proceso de desarrollo unificado está basado en Rational Unified Process (RUP).

6.6. METODOLOGÍA SCRUM

6.6.1. OBJETIVOS.

Scrum busca mejorar la productividad del equipo de desarrollo mediante la entrega incremental de productos de valor. El objetivo principal es gestionar el desarrollo de proyectos complejos mediante una estructura ágil, transparente y colaborativa, permitiendo ajustes rápidos y continuos.

6.6.2. CARACTERÍSTICAS.

1. Iterativo e incremental: El trabajo se divide en ciclos cortos llamados sprints (generalmente de 1 a 4 semanas).
2. Autogestionado: El equipo decide cómo organizar su trabajo.

3. Enfoque en el cliente: Se busca entregar valor al cliente en cada sprint.
4. Transparencia: Todos los involucrados deben tener visibilidad sobre el progreso del proyecto.
5. Adaptabilidad: Se hacen ajustes durante el proceso para responder a cambios.
6. Mejora continua: Se busca la mejora constante tanto del proceso como del producto.

6.6.3. ELEMENTOS.

ROLES:

- Product Owner (PO): Responsable de definir y priorizar las características del producto (backlog).
- Scrum Master: Facilita el proceso, elimina obstáculos y asegura el cumplimiento de las prácticas de Scrum.
- Equipo de Desarrollo: Son los encargados de construir el producto, autoorganizados y multifuncionales.

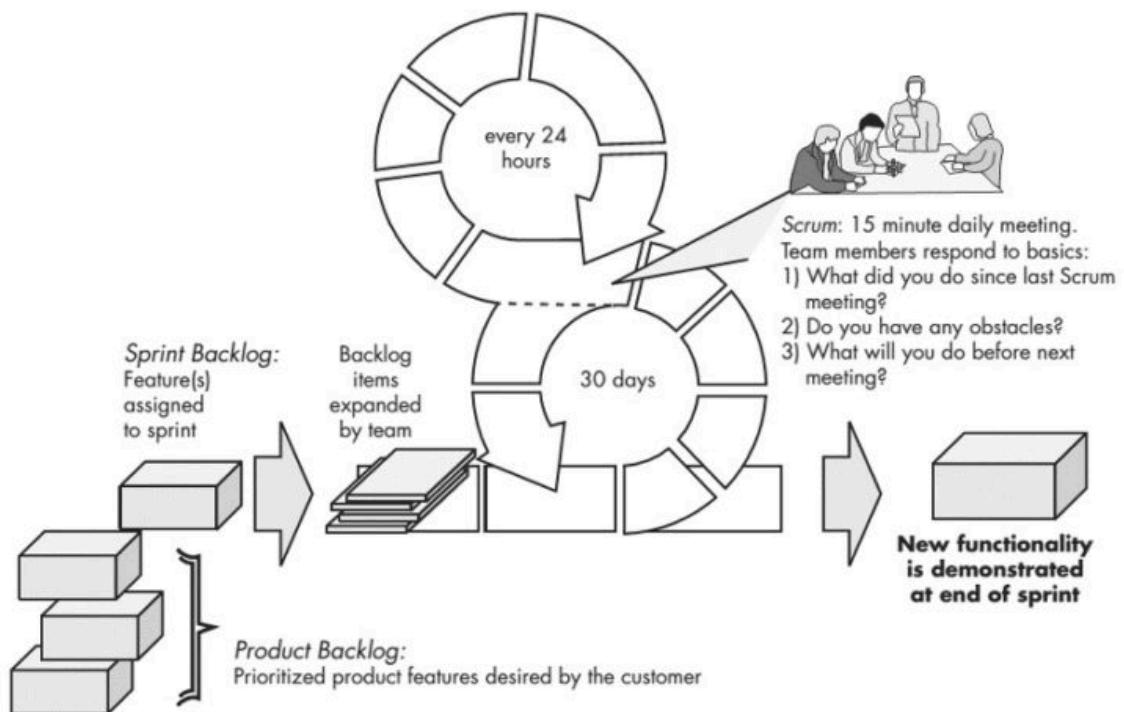
ARTEFACTOS:

- Product Backlog: Lista priorizada de todo lo que se debe hacer en el proyecto.
- Sprint Backlog: Tareas seleccionadas del Product Backlog que se realizan en el sprint actual.
- Incremento: Resultado tangible del trabajo completado durante el sprint.

6.6.4. EVENTOS.

- Sprint: Periodo de tiempo limitado para desarrollar un incremento del producto.
- Sprint Planning: Reunión al inicio del sprint para planificar qué se va a hacer.
- Daily Scrum: Reunión diaria de 15 minutos para sincronizar actividades.
- Sprint Review: Revisión al final del sprint para mostrar el trabajo realizado y recibir retroalimentación.
- Sprint Retrospective: Reflexión al final del sprint sobre lo que funcionó y lo que no, con miras a mejorar el proceso.

6.6.5. ESTRUCTURA.



- Sprint: El ciclo de trabajo principal que dura entre 1 y 4 semanas.
- Backlogs: El Product Backlog (lista priorizada del producto) y el Sprint Backlog (tareas del sprint actual).

- Reuniones periódicas: Sprint Planning, Daily Scrum, Sprint Review y Sprint Retrospective.

6.6.6. DISCIPLINA.

Scrum requiere disciplina para seguir los principios de la metodología ágil. La autogestión del equipo, la transparencia en los procesos, la comunicación constante y la mejora continua son esenciales. Cada miembro debe estar comprometido con los objetivos del equipo y trabajar de manera colaborativa.

6.6.7. TAREAS.

- Planificación: Definir las tareas a realizar durante el sprint.
- Desarrollo: Trabajar en las tareas del Sprint Backlog.
- Revisión: Presentar los resultados al Product Owner y a los stakeholders.
- Mejora: Reflexionar sobre lo que se puede mejorar y aplicar ajustes en el siguiente sprint.

En resumen, Scrum es un marco de trabajo ágil centrado en entregar valor mediante la colaboración constante, la autogestión y la adaptabilidad, con el fin de optimizar el proceso de desarrollo de productos y asegurar la satisfacción del cliente.

6.7. CARACTERÍSTICAS DE UNA APLICACIÓN WEB

Las Aplicaciones Web tienen una serie de rasgos comunes que diferencian a unas de otras:

- Desde el punto de vista del usuario, se ha universalizado su accesibilidad: Actualmente un usuario experto y un usuario con habilidad limitada en el uso de aplicaciones informáticas acceden al mismo tipo de aplicación. Aún más,

el número y tipo de usuario de las Aplicaciones Web no siempre es predecible, lo que obliga a tener el concepto de facilidad de uso aún más presente que en otros tipos de aplicaciones.

- Desde el punto de vista de la plataforma se realiza un uso intensivo de la red y la conexión se establece desde distintos tipos de dispositivos de acceso.
- Desde el punto de vista de la información, asistimos en la actualidad a una disponibilidad global de fuentes heterogéneas de información, estructurada y no estructurada, pertenecientes a distintos dominios y que colaboran en el cumplimiento de los objetivos de la aplicación.

6.8. PROPUESTA DE VALOR : OPTIMIZACIÓN DE RUTAS

6.8.1. TEORÍA DEL MODELO MATEMÁTICO DE DIJKSTRA

El algoritmo de Dijkstra es un algoritmo para encontrar el camino más corto entre un nodo inicial y los demás nodos en un grafo ponderado con pesos no negativos en las aristas. En este caso, los nodos representan las localidades de los clientes (y el almacén), y las aristas serían las rutas entre esos puntos, con los pesos representando distancias o tiempos de entrega.

6.8.2. MODELO MATEMÁTICO DE DIJKSTRA

1. Definición del Grafo

Se define un grafo dirigido $G = (V, E)$ donde:

- V es el conjunto de nodos (representando las localidades o puntos de entrega: almacén y clientes).
- E es el conjunto de aristas o conexiones entre nodos (representando las rutas de entrega entre localidades).

- Cada arista $e = (u, v)$ tiene un peso $w(u, v)$ que representa la distancia o tiempo de entrega entre los nodos u y v

2. Variables

- $d(v)$: Distancia mínima desde el nodo de inicio hasta el nodo v .
Inicialmente, $d(v) = \infty$ para todos los nodos v excepto para el nodo inicial s , donde $d(s) = 0$.
- $P(v)$: Nodo predecesor de v en el camino más corto desde el nodo inicial.
- Q : Conjunto de nodos no procesados (en la cola de prioridad).

3. Pasos del Algoritmo de Dijkstra

- Inicialización: Establecer $d(s) = 0$ para el nodo de inicio (almacén) y $d(v) = \infty$ para todos los demás nodos.
- Colocar todos los nodos v en un conjunto Q .
- Mientras Q no esté vacío:
 - Seleccionar el nodo $u \in Q$ con la menor distancia $d(u)$.
 - Eliminar u de Q .
 - Para cada vecino v de u que aún esté en Q , actualizar $d(v)$ y $P(v)$ si se encuentra una ruta más corta:

$$d(v) = \min(d(v), d(u) + w(u, v))$$

Si $d(v)$ cambia, actualizar el predecesor $P(v) = u$.

- El proceso continúa hasta que se han procesado todos los nodos.

4. Resultado Final

- Al finalizar, para cada nodo v , se tendrá:
 - $d(v)$: La distancia más corta desde el nodo inicial hasta v .
 - $P(v)$: El nodo predecesor de v , que puede ser utilizado para reconstruir el camino más corto.

6.8.3. APLICACIÓN A UN SISTEMA DE VENTAS Y PEDIDOS DE HUEVOS

En el contexto de la **distribuidora de huevos** "Doña Chela", el modelo matemático de Dijkstra se puede aplicar de la siguiente forma:

1. **Nodos:**

- El **almacén** y los **clientes** de la distribuidora son los nodos

$$V = \{almacén, cliente1, cliente2, \dots, clienten\}$$

2. **Aristas:**

- Las **rutas de entrega** entre los puntos son las aristas

$$E = \{(almacen, cliente1), (almacen, cliente2), \dots, (cliente1, cliente2), \dots\}$$

- El peso $w(u, v)$ de cada arista puede ser la **distancia** en kilómetros o el **tiempo estimado de entrega** en horas.

2. **Objetivo:**

- El **objetivo** es encontrar la **ruta más corta** desde el almacén hasta los diferentes puntos de entrega. Si hay múltiples rutas para cubrir diferentes clientes, se pueden calcular las rutas más cortas a cada uno de ellos y **minimizar los tiempos de entrega**.

3. **Optimización:**

- En un contexto más complejo, se puede usar Dijkstra para **optimizar el trayecto** de entrega a cada cliente, considerando si las rutas son **lineales** (un solo vehículo) o si se deben optimizar múltiples **vehículos de entrega** (utilizando algoritmos más avanzados como el **Problema de Rutas de Vehículos (VRP)**).

6.8.4. VENTAJAS DE APLICAR DIJKSTRA EN ESTE SISTEMA

- **Optimización de la logística:** Reducir los tiempos de entrega y la distancia recorrida por los vehículos.
- **Reducción de costos:** Menor consumo de combustible y ahorro de tiempo.
- **Simplicidad:** Dijkstra es relativamente simple de implementar y no requiere demasiados recursos computacionales.

6.8.5. LIMITACIONES

- **Problemas de múltiples vehículos:** Dijkstra es más adecuado para calcular el camino más corto entre dos nodos, pero no resuelve directamente problemas de rutas múltiples o combinatorias como el **Problema del Viajante de Comercio (TSP)** o el **VRP**.
- **Condiciones dinámicas:** Si las rutas cambian constantemente (por ejemplo, por tráfico o nuevas rutas), Dijkstra puede necesitar ser recalculado.

En resumen, el algoritmo de Dijkstra es un excelente punto de partida para la optimización de las rutas de entrega en el sistema de ventas y pedidos de huevos, ya que puede proporcionar las distancias más cortas entre los puntos, facilitando la planificación de rutas eficientes para el administrador.

CAPÍTULO III MARCO APlicativo.

Metodología Scrum por Fases

Scrum se estructura en ciclos cortos llamados *sprints*, que suelen durar entre dos y cuatro semanas. Cada sprint implica una serie de eventos y actividades que conducen a la entrega de un incremento de producto funcional.

Fases de Scrum

1. Planificación del Sprint:

- **Definición del objetivo del Sprint:** El equipo se reúne para determinar qué funcionalidades se desarrollarán durante el próximo Sprint, basándose en el Product Backlog (lista de requisitos priorizados).
- **Estimación del trabajo:** Cada tarea se estima en términos de esfuerzo, utilizando una unidad de medida como puntos de historia.
- **Creación del Sprint Backlog:** Se crea una lista detallada de las tareas que se realizarán durante el Sprint.

2. Reuniones Diarias (Daily Scrum):

- **Reunión breve y diaria:** El equipo se reúne durante 15 minutos para sincronizarse, revisar el progreso y abordar cualquier obstáculo.
- **Formato:** ¿Qué hice ayer? ¿Qué haré hoy? ¿Qué impedimentos tengo?

3. Desarrollo:

- **Trabajo en equipo:** Los miembros del equipo trabajan de forma colaborativa para completar las tareas del Sprint.
- **Adaptación:** El equipo puede ajustar el Sprint Backlog si surgen nuevos requerimientos o si se encuentran obstáculos.

4. Revisión del Sprint:

- **Demostración del incremento:** El equipo presenta el trabajo realizado durante el Sprint al Product Owner y a los stakeholders.

- **Recopilación de feedback:** Se obtiene feedback sobre el incremento y se identifican áreas de mejora.

5. Retrospectiva:

- **Reflexión:** El equipo se reúne para reflexionar sobre lo que ha funcionado bien, lo que podría mejorar y qué acciones se tomarán para el próximo Sprint.

Roles Clave en Scrum

- **Product Owner:** Responsable de la visión del producto y de priorizar el Product Backlog.
- **Scrum Master:** Facilita el proceso Scrum, elimina obstáculos y asegura que el equipo siga las prácticas de Scrum.
- **Equipo de Desarrollo:** Realiza el trabajo de desarrollo del producto.

Artefactos de Scrum

- **Product Backlog:** Lista ordenada de todas las características, funcionalidades y mejoras que se desean en el producto.
- **Sprint Backlog:** Subconjunto del Product Backlog que se selecciona para un Sprint específico.
- **Incremento:** El resultado tangible de cada Sprint, que se suma a los incrementos anteriores.

Adaptación al Contexto de "Gallina Feliz"

Al aplicar Scrum al desarrollo del sistema de administración para "Gallina Feliz", el Product Backlog podría incluir elementos como:

- Módulo de gestión de pedidos
- Módulo de inventario

- Módulo de clientes
- Informes personalizados
- Integración con sistemas de pago
- Interfaz de usuario intuitiva

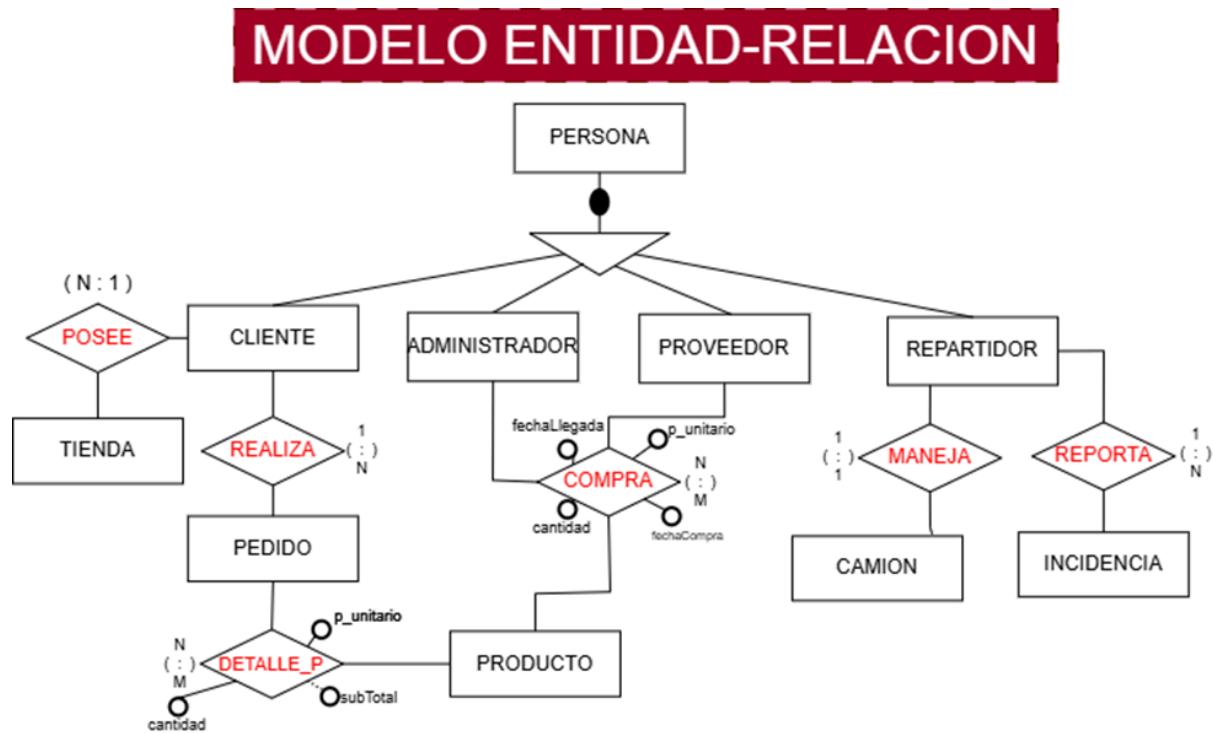
Cada Sprint se centraría en desarrollar un conjunto de funcionalidades específicas, permitiendo una entrega gradual y la posibilidad de obtener feedback temprano.

Beneficios de utilizar Scrum:

- **Mayor flexibilidad:** Permite adaptarse a los cambios y a las nuevas necesidades del negocio.
- **Mayor transparencia:** El progreso del proyecto es visible para todos los involucrados.
- **Mayor calidad:** El enfoque en la entrega incremental permite identificar y corregir errores de manera temprana.
- **Mayor satisfacción del cliente:** El cliente puede ver el progreso del producto de forma regular y proporcionar feedback.

ANEXOS

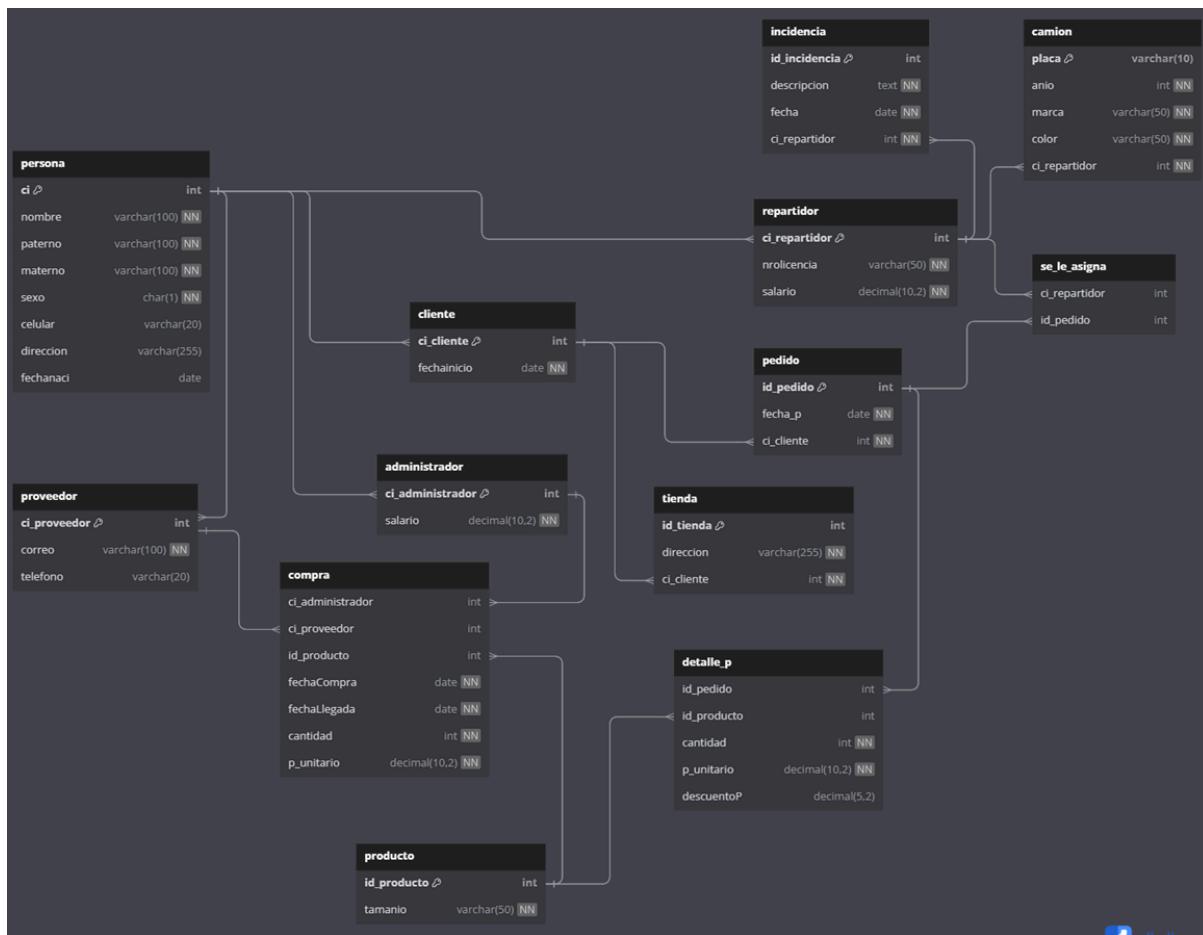
MODELO ENTIDAD RELACIÓN



MODELO RELACIONAL

MODELO RELACIONAL

PERSONA(ci, nombre, paterno, materno, sexo, celular, dirección, fechanaci)
CLIENTE(ci_cliente, antigüedad)
REPARTIDOR(ci_repartidor, nrolicencia, salario, horarioLaboral, placa)
ADMINISTRADOR(ci_administrador, salario)
PROVEEDOR(ci_proveedor, correo, telefono)
TIENDA(id_tienda, dirección, ci_cliente)
DETALLE_P(id_pedido, id_producto, cantidad, p_unitario, descuentoP)
PEDIDO(id_pedido, fecha_p, ci_cliente)
PRODUCTO(id_producto, tamaño)
CAMION(placa, anio, marca, color, ci_repartidor)
INCIDENCIA(id_incidencia, descripción, fecha, ci_repartidor)
COMPRA(ci_administrador, ci_proveedor, id_producto, fechaCompra, fechaLlegada, cantidad, p_unitario)
SE_LE_ASIGNA(ci_repartidor, id_pedido)



DICCIONARIO DE DATOS (USANDO EL GESTOR POSTGRES)

TABLA ADMINISTRADOR

administrador								
General Columns Advanced Constraints Partitions Parameters Security SQL								
Inherited from table(s)		Select to inherit from...						
Columns								
	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default	
	ci_administrador	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
	salario	numeric	10	2	<input checked="" type="checkbox"/>	<input type="checkbox"/>		

TABLA CAMION

camion

General Columns Advanced Constraints Partitions Parameters Security SQL

Inherited from table(s)

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	placa	character varying	10		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	anio	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	marca	character varying	50		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	color	character varying	50		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	ci_repartidor	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	

TABLA CLIENTE

cliente

General Columns Advanced Constraints Partitions Parameters Security SQL

Inherited from table(s)

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	ci_cliente	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	fechainicio	date			<input checked="" type="checkbox"/>	<input type="checkbox"/>	

TABLA COMPRA

compra

General Columns Advanced Constraints Partitions Parameters Security SQL

Inherited from table(s)

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	ci_administrador	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	ci_proveedor	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	id_producto	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	fechacompra	date			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	fechallegada	date			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	cantidad	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	p_unitario	numeric	10	2	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

TABLA DETALLE PEDIDO

detalle_p

General Columns Advanced Constraints Partitions Parameters Security SQL

Inherited from table(s) Select to inherit from...

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	id_pedido	integer	<input type="button" value=" ▾"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	id_producto	integer	<input type="button" value=" ▾"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	cantidad	integer	<input type="button" value=" ▾"/>		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	p_unitario	numeric	<input type="button" value=" ▾"/>	10	2	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	descuentop	numeric	<input type="button" value=" ▾"/>	5	2	<input type="checkbox"/>	<input type="checkbox"/>
							0

TABLA INCIDENCIA

incidencia

General Columns Advanced Constraints Partitions Parameters Security SQL

Inherited from table(s) Select to inherit from...

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	id_incidencia	integer	<input type="button" value=" ▾"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	descripcion	text	<input type="button" value=" ▾"/>		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	fecha	date	<input type="button" value=" ▾"/>		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	ci_repartidor	integer	<input type="button" value=" ▾"/>		<input checked="" type="checkbox"/>	<input type="checkbox"/>	

TABLA PEDIDO

pedido

General Columns Advanced Constraints Partitions Parameters Security SQL

Inherited from table(s) Select to inherit from...

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	id_pedido	integer	<input type="button" value=" ▾"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	fecha_p	date	<input type="button" value=" ▾"/>		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	ci_cliente	integer	<input type="button" value=" ▾"/>		<input checked="" type="checkbox"/>	<input type="checkbox"/>	

TABLA PERSONA

persona

General Columns Advanced Constraints Partitions Parameters Security SQL

Inherited from table(s)

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	ci	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	nombre	character varying	100		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	paterno	character varying	100		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	materno	character varying	100		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	sexo	character	1		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	celular	character varying	20		<input type="checkbox"/>	<input type="checkbox"/>	
	direccion	character varying	255		<input type="checkbox"/>	<input type="checkbox"/>	
	fechanaci	date			<input type="checkbox"/>	<input type="checkbox"/>	

TABLA PRODUCTO

producto

General Columns Advanced Constraints Partitions Parameters Security SQL

Inherited from table(s)

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	id_producto	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	tamano	character varying	50		<input checked="" type="checkbox"/>	<input type="checkbox"/>	

TABLA PROVEEDOR

proveedor

General Columns Advanced Constraints Partitions Parameters Security SQL

Inherited from table(s)

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	ci_proveedor	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	correo	character varying	100		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	telefono	character varying	20		<input type="checkbox"/>	<input type="checkbox"/>	

TABLA REPARTIDOR

repartidor

General Columns Advanced Constraints Partitions Parameters Security SQL

Inherited from table(s) Select to inherit from...

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	ci_repartidor	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	nrolicencia	character varying	50		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	salario	numeric	10	2	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

TABLA TIENDA

tienda

General Columns Advanced Constraints Partitions Parameters Security SQL

Inherited from table(s) Select to inherit from...

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	id_tienda	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	direccion	character varying	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	ci_cliente	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	

TABLA SE LE ASIGNA

se_le_asigna

General Columns Advanced Constraints Partitions Parameters Security SQL

Inherited from table(s) Select to inherit from...

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	ci_repartidor	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	id_pedido	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Tabla: persona

Campo	Tipo	PK/FK	Descripción	Ejemplo
ci	int	PK	Identificador único de la persona Número de 6 a 10 dígitos , not null	12345678

nombre	varchar(100)	Ninguno	Primer nombre de la persona Cadena de letras , not null	Juan
paterno	varchar(100)	Ninguno	Apellido paterno Cadena de letras , not null	Pérez
materno	varchar(100)	Ninguno	Apellido materno Cadena de letras , not null	Gómez
sexo	char(1)	Ninguno	Género de la persona , not null Puede ser “M” o “F”	M
celular	varchar(20)	Ninguno	Número de celular Cadena numérica de 8 dígitos	789456123
direccion	varchar(255)	Ninguno	Dirección de residencia , not null Cadena alfanumérica	Av. Siempre Viva
fechanaci	date	Ninguno	Fecha de nacimiento , not null Formato “aaaa-mm-dd”	1990-05-15

Tabla: cliente

Campo	Tipo	PK/FK	Descripción	Ejemplo
ci_cliente	int	PK, FK	Relaciona a la persona que es cliente , not null 8-10 dígitos	12345678
fechainicio	date	Ninguno	Fecha en la que se registró como cliente , not null Formato “aaaa-mm-dd”	2023-01-10

Tabla: administrador

Campo	Tipo	PK/FK	Descripción	Ejemplo
ci_administrador	int	PK, FK	Identificador único del administrador , not null 8-10 digitos	98765432
salario	decimal(10,2)	Ninguno	Salario del administrador , not null	3500.50

Tabla: proveedor

Campo	Tipo	PK/FK	Descripción	Ejemplo
ci_proveedor	int	PK	Identificador único del proveedor , not null 8-10 digitos	56789012
correo	varchar(100)	Ninguno	Correo electrónico del proveedor , not null Cadena alfanumérica, debe contener "@"	abc@ejemplo.com
telefono	varchar(20)	Ninguno	Número telefónico del proveedor , not null Número de 8 dígitos	765432109

Tabla: compra

Campo	Tipo	PK/FK	Descripción	Ejemplo
ci_administrador	int	FK	Identificador del administrador que compra , not null 8-10 digitos	98765432

ci_proveedor	int	FK	Identificador del proveedor , not null 8-10 digitos	56789012
id_producto	int	FK	Identificador del producto , not null	101
fechaCompra	date	Ninguno	Fecha en que se realizó la compra , not null Formato “aaaa-mm-dd”	2023-02-15
fechaLlegada	date	Ninguno	Fecha estimada de llegada , not null Formato “aaaa-mm-dd”	2023-02-20
cantidad	int	Ninguno	Cantidad de productos comprados , not null Número mínimo “1”	50
p_unitario	decimal(10,2)	Ninguno	Precio unitario del producto , not null	25.00

Tabla: producto

Campo	Tipo	PK/FK	Descripción	Ejemplo
id_producto	int	PK	Identificador único del producto , not null	101
tamaño	varchar(50)	Ninguno	Tamaño o presentación del producto , not null Posibles opciones: “Pequeño”, “Mediano”, “Grande”	Mediano

Tabla: tienda

Campo	Tipo	PK/FK	Descripción	Ejemplo

id_tienda	int	PK	Identificador único de la tienda , not null	1
direccion	varchar(255)	Ninguno	Dirección de la tienda , not null Cadena alfanumérica	Av. Bolívar
ci_cliente	int	FK	Cliente asociado a la tienda , not null 8-10 digitos	12345678

Tabla: detalle_p

Campo	Tipo	PK/FK	Descripción	Ejemplo
id_pedido	int	FK	Pedido al que pertenece el detalle, not null	1001
id_producto	int	FK	Producto asociado al detalle, not null	101
cantidad	int	Ninguno	Cantidad de producto, not null Número mínimo: "1"	5
p_unitario	decimal(10,2)	Ninguno	Precio unitario, not null	25.00
descuentoP	decimal(5,2)	Ninguno	Descuento aplicado	5.00

Tabla: pedido

Campo	Tipo	PK/FK	Descripción	Ejemplo
id_pedido	int	PK	Identificador único del pedido, not null	1001
fecha_p	date	Ninguno	Fecha en la que se realizó el pedido, not null	2023-03-01

			Formato “aaaa-mm-dd”	
ci_cliente	int	FK	Cliente asociado al pedido, not null 8-10 digitos	12345678

Tabla: incidencia

Campo	Tipo	PK/FK	Descripción	Ejemplo
id_incidencia	int	PK	Identificador único de la incidencia, not null	2001
descripcion	varchar(255)	Ninguno	Detalle de la incidencia	Daño en producto
fecha	date	Ninguno	Fecha en la que ocurrió la incidencia , not null Formato “aaaa-mm-dd”	2023-04-10
ci_repartidor	int	FK	Repartidor relacionado, not null 8-10 digitos	87654321

Tabla: repartidor

Campo	Tipo	PK/FK	Descripción	Ejemplo
ci_repartidor	int	PK, FK	Identificador único del repartidor, not null 8-10 digitos	87654321
nrolicencia	varchar(50)	Ninguno	Número de licencia del repartidor , not null	ABC12345

salario	decimal(10,2)	Ninguno	Salario del repartidor , not null	1500.00
---------	---------------	---------	-----------------------------------	---------

Tabla: camion

Campo	Tipo	PK/FK	Descripción	Ejemplo
placa	varchar(10)	PK	Placa única del camión , not null Cadena alfanumerica de 10 campos	1234ABC
anio	int	Ninguno	Año del modelo del camión , not null	2020
marca	varchar(50)	Ninguno	Marca del camión	Toyota
color	varchar(50)	Ninguno	Color del camión	Blanco
ci_repartidor	int	FK	Repartidor asignado al camión 8-10 digitos	87654321

Tabla: se_le_asigna

Campo	Tipo	PK/FK	Descripción	Ejemplo
ci_repartidor	int	FK	Repartidor asignado 8-10 digitos	87654321
id_pedido	int	FK	Pedido asignado, not null	1001

LISTA DE TAREAS PARA LAS FASES DE CASCADA

1. FASE COMUNICACIÓN.

- a. Revisión inicial con el gerente de ventas Lic. Marcelo Quiroga, encargado de inventario y ventas de la empresa “Gallina Feliz”.
- b. Observación del entorno de trabajo para identificación de funcionalidades esenciales.
- c. Encuesta al personal de ventas sobre el procedimiento del área de ventas.
- d. Análisis del equipamiento necesario para la creación del sistema.
- e. Modelado de la solución para la presentación al usuario destinado y posterior verificación.
- f. Documentar correcciones hechas durante la verificación para su aprobación.

2. FASE PLANEACIÓN.

- a. Prestación de objetivos y costos.
- b. Calendarizar el transcurso del desarrollo del sistema, más fechas claves para las fases del proyecto.
- c. Proposición de opiniones sobre las herramientas a usar en el sistema (software).
- d. Identificar riesgos potenciales y planeación de un plan de mitigación y contingencia.

3. FASE MODELADO.

- a. Diseño del prototipo de la página web en base a pantallas muertas.
- b. Modelado de fluujogramas.
- c. Modelado de diagramas.
- d. Definición de MVC como patrón de diseño.

4. FASE CONSTRUCCIÓN.

- a. Desarrollo de la base de datos y pruebas de consulta a casas módulo.

- b. desarrollo página web, incluido las altas, bajas y ediciones, junto con las notificaciones.
- c. Verificación de componentes uno por uno.
- d. Documentación del código.

5. FASE DESPLIEGUE

- a. Entrega del sistema
- b. Verificación por parte del cliente.
- c. Capacitación a los usuarios sobre el manejo del sistema.

PREGUNTAS DE OBSERVACIÓN Y ENTREVISTA

FICHA DE OBSERVACIÓN

Nombre de la Empresa:

Ubicación:

Observador: Fecha:

Hora Inicio: Hora Fin:

TIPOS Y CLASIFICACIÓN		RESPUESTA
1	¿Cuántos tipos de huevos manejan(tamaño, color, orgánicos, etc)?	
2	¿Cómo se clasifican los huevos antes de la distribución?	
CALIDAD		RESPUESTA
3	¿Qué hacer con los huevos dañados o no aptos para la venta?	
4	¿Cómo verifican la calidad de los huevos antes de entregarlos a los clientes?	
DISTRIBUCIÓN		RESPUESTA
5	¿Cómo se organizan los huevos para su transporte(empaque, cajas, etc)?	

TIPOS Y CLASIFICACIÓN		RESPUESTA
6	¿Qué cantidad promedio se distribuye por cliente o pedido?	
DEMANDA		RESPUESTA
7	¿Qué tipos de huevos tienen mayor demanda?	
8	¿Hay temporada donde la demanda varía significativamente?	
PRECIOS		RESPUESTA
9	¿Cómo se determinan los precios según el tipo y la calidad del huevo?	
10	¿Varían los precios por volumen o cliente?	

FICHA DE OBSERVACIÓN (RESPUESTA)

Nombre de la Empresa: Gallina Feliz.....

Ubicación: La Paz y El Alto.....

Observador: Marcos Joel Machicado Lopez..... Fecha: 22/04/25.....

Hora Inicio: 12:00..... Hora Fin: 12:30.....

TIPOS Y CLASIFICACIÓN		RESPUESTA
1	¿Cuántos tipos de huevos manejan(tamaño, color, orgánicos, etc)?	Actualmente manejan tres tipos principales: huevos medianos, grandes y pequeños.
2	¿Cómo se clasifican los huevos antes de la distribución?	Los huevos se clasifican manualmente por tamaño, color y calidad, utilizando inspección visual.
CALIDAD		RESPUESTA
3	¿Qué hacer con los huevos dañados o no aptos para la venta?	Los huevos dañados se descartan y, en algunos casos, se destinan para uso interno o se venden a bajo costo a empresas que elaboran alimentos procesados.
4	¿Cómo verifican la calidad de los huevos antes de entregarlos a los clientes?	Se realiza una inspección manual para detectar grietas o irregularidades y verificar frescura mediante el peso y color de la cáscara.
DISTRIBUCIÓN		RESPUESTA
5	¿Cómo se organizan los huevos para su transporte(empaque, cajas, etc)?	Los huevos se empaquetan en bandejas de cartón de 30 unidades, que luego se colocan en

TIPOS Y CLASIFICACIÓN		RESPUESTA
		cajas para facilitar su transporte y evitar daños.
6	¿Qué cantidad promedio se distribuye por cliente o pedido?	Por cliente, se distribuyen en promedio 50 a 200 bandejas, dependiendo de si es una tienda pequeña o un supermercado.
DEMANDA		RESPUESTA
7	¿Qué tipos de huevos tienen mayor demanda?	Los huevos grandes tienen la mayor demanda, especialmente en supermercados.
8	¿Hay temporada donde la demanda varía significativamente?	Sí, durante las festividades y la época de fin de año la demanda aumenta hasta un 30%.
PRECIOS		RESPUESTA
9	¿Cómo se determinan los precios según el tipo y la calidad del huevo?	Los precios se calculan considerando el costo de producción, el tipo de huevo y las condiciones del mercado.
10	¿Varían los precios por volumen o cliente?	Sí, los clientes que compran al por mayor reciben descuentos escalonados según la cantidad adquirida.

ENCUESTA DEL DESARROLLO DE SISTEMA DE VENTAS Y PEDIDOS

Nombre de la Empresa:

Observador: Fecha:

Cargo que Ocupa en la Empresa:

Hora Inicio: Hora Fin:

PREGUNTAS		RESPUESTA
1	¿Cuál es el proceso actual de ventas de la empresa?	
2	¿Qué herramientas o sistemas utilizan actualmente para gestionar ventas?	
3	¿Qué funcionalidades consideras esenciales para el sistema?	
4	¿Qué tipo de reportes y análisis necesitas obtener del sistema?	

5	¿Cuántos usuarios utilizarán el sistema y qué funciones cumplen dentro de la empresa?	
6	¿Qué presupuesto tienen asignado para el desarrollo e implementación del sistema?	
7	¿Cómo se maneja el inventario en relación a las ventas?	
8	¿Qué métricas usan para medir el éxito de la empresa?	
9	¿Qué estándares o normas deben ser aplicados en el sistema?	
10	¿Qué tecnologías tienen disponible actualmente en la distribuidora (computadora, red, internet)?	

ENCUESTA DEL DESARROLLO DE SISTEMA DE VENTAS Y PEDIDOS **(RESPUESTA)**

Nombre de la Empresa: Gallina Feliz.....

Observador: Marcos Joel Machicado Lopez..... Fecha: 22/04/25.....

Cargo que ocupa en la Empresa: Administrador.....

Hora Inicio: 12:30..... Hora Fin: 13:00.....

PREGUNTAS		RESPUESTA
1	¿Cuál es el proceso actual de ventas de la empresa?	Las ventas se realizan de forma manual. Los clientes hacen pedidos por teléfono o en persona, y se registran en cuadernos físicos.
2	¿Qué herramientas o sistemas utilizan actualmente para gestionar ventas?	Principalmente utilizan hojas de cálculo y registros manuales en papel.
3	¿Qué funcionalidades consideras esenciales para el sistema?	Gestión de pedidos, control de inventario, generación de facturas, seguimiento de clientes y reportes automáticos.
4	¿Qué tipo de reportes y análisis necesitas obtener del sistema?	Reportes de ventas diarias, mensuales y anuales, análisis de productos más vendidos, y control del inventario.

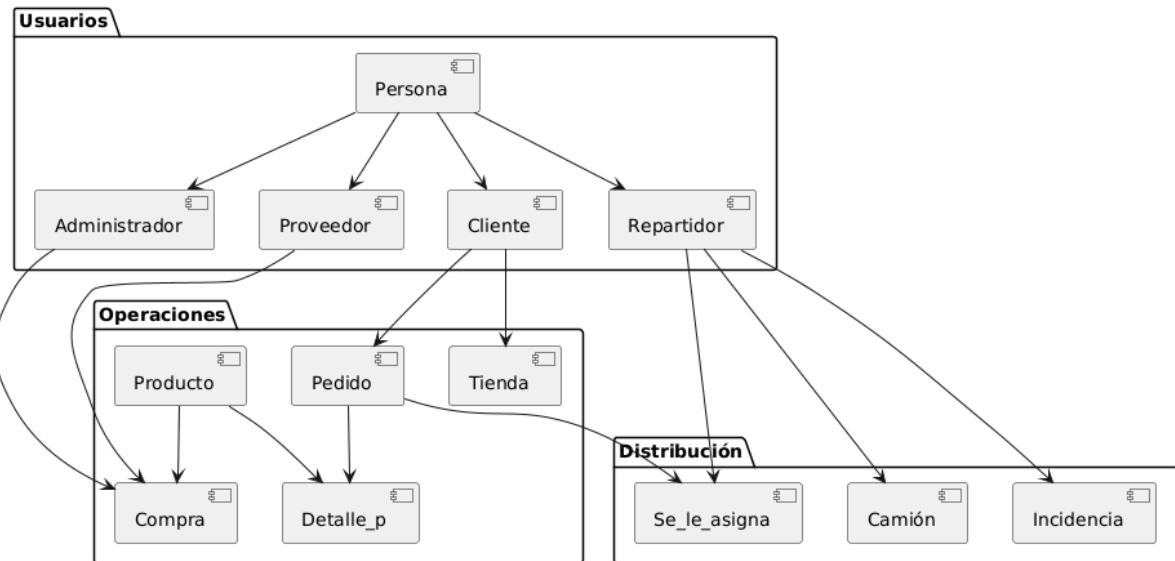
5	¿Cuántos usuarios utilizarán el sistema y qué funciones cumplen dentro de la empresa?	Se estima que cinco usuarios lo utilizarán, incluyendo dos encargados de ventas, un gerente y dos responsables de inventario.
6	¿Qué presupuesto tienen asignado para el desarrollo e implementación del sistema?	Aproximadamente entre \$3,000 y \$5,000.
7	¿Cómo se maneja el inventario en relación a las ventas?	El inventario se actualiza manualmente después de cada venta, lo que suele causar retrasos y errores.
8	¿Qué métricas usan para medir el éxito de la empresa?	Ventas mensuales, número de clientes recurrentes y volúmenes de pedidos.
9	¿Qué estándares o normas deben ser aplicados en el sistema?	Cumplimiento con la normativa fiscal boliviana y estándares de calidad en el manejo de datos.
10	¿Qué tecnologías tienen disponible actualmente en la distribuidora (computadora, red, internet)?	Cuentan con computadoras básicas, conexión a internet y una red local sencilla.

CALENDARIZACIÓN PARA EL DESARROLLO DEL SISTEMA (SCRUM)

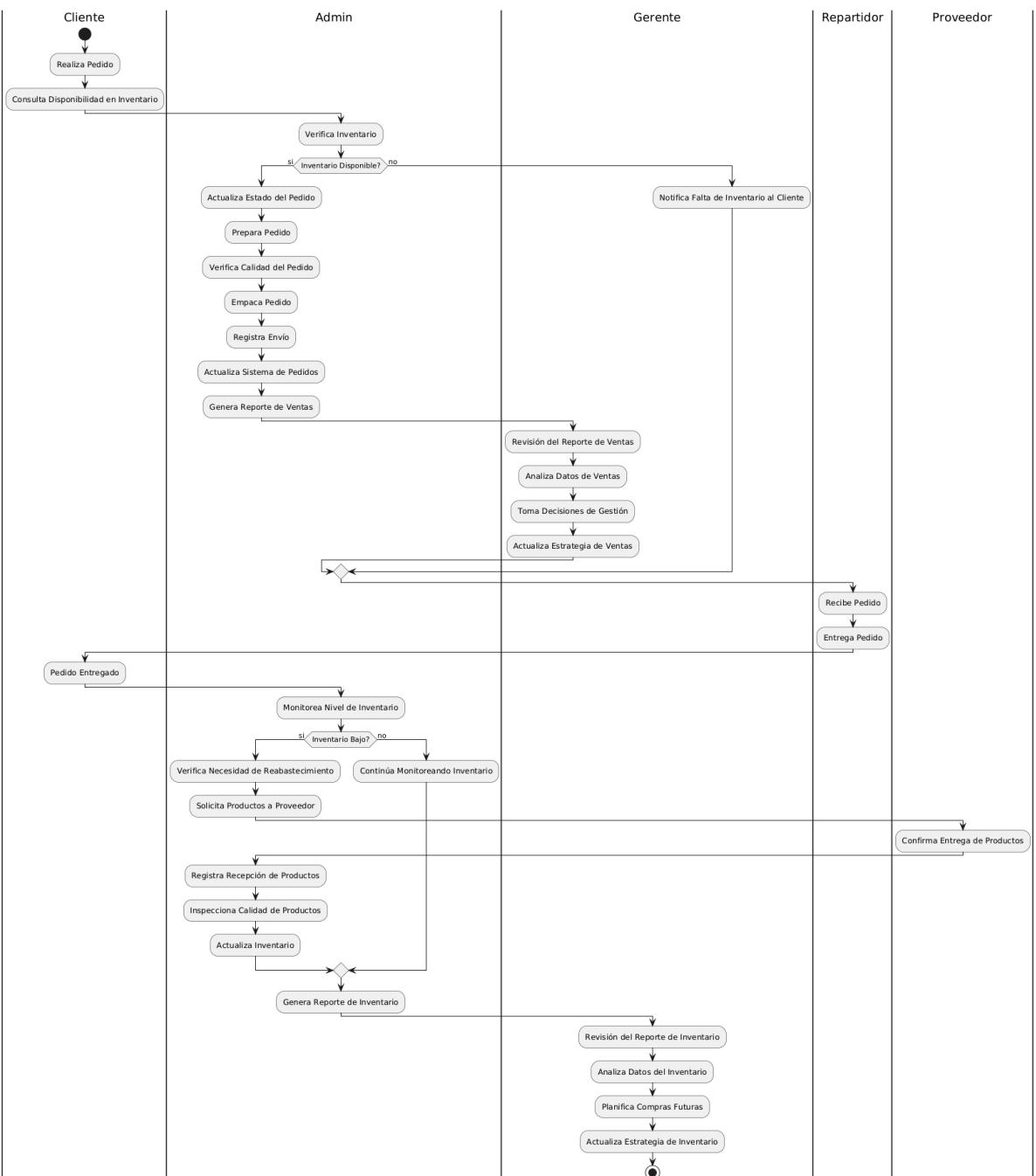
Sprints		FEB	MAR	
	Tablero Sprint 1			
✓ CS-1 Análisis y Modelado Inicial	FINALIZADA	<div style="width: 50%; background-color: #6A5ACD2; height: 10px;"></div>		
✓ CS-6 Reunión con el Product Owner Lic. Marcelo Quiroga) para entender...	FINALIZADA	<div style="width: 50%; background-color: #6A5ACD2; height: 10px;"></div>		
✓ CS-7 Observación del entorno y encuesta al personal de venta...	FINALIZADA	<div style="width: 50%; background-color: #6A5ACD2; height: 10px;"></div>		
✓ CS-8 Análisis de equipamiento necesario para el sistema.	FINALIZADA	<div style="width: 50%; background-color: #6A5ACD2; height: 10px;"></div>		
✓ CS-9 Modelado inicial: Diagramas de caso de uso, secuencia y entidad-r...	FINALIZADA	<div style="width: 50%; background-color: #6A5ACD2; height: 10px;"></div>		
✓ CS-10 Presentación del modelo al Product Owner para verificación.	FINALIZADA	<div style="width: 50%; background-color: #6A5ACD2; height: 10px;"></div>		
✓ CS-11 Documentación de correcciones y ajustes al modelo.	FINALIZADA	<div style="width: 50%; background-color: #6A5ACD2; height: 10px;"></div>		
✓ CS-12 Diseño inicial de prototipos de pantalla.	FINALIZADA	<div style="width: 50%; background-color: #6A5ACD2; height: 10px;"></div>		
✓ CS-13 Sprint Review y Retrospective.	FINALIZADA	<div style="width: 50%; background-color: #6A5ACD2; height: 10px;"></div>		
✓ CS-2 Diseño del Sistema	FINALIZADA	<div style="width: 50%; background-color: #2ECC71; height: 10px;"></div>		
✓ CS-14 Diseño completo de la base de datos y creación de tablas iniciale...	FINALIZADA	<div style="width: 50%; background-color: #2ECC71; height: 10px;"></div>		
✓ CS-15 Validación de queries básicos con datos de prueba.	FINALIZADA	<div style="width: 50%; background-color: #2ECC71; height: 10px;"></div>		
✓ CS-16 Refinar prototipos de pantallas según retroalimentación del Produ...	FINALIZADA	<div style="width: 50%; background-color: #2ECC71; height: 10px;"></div>		
✓ CS-17 Definir MVC como patrón de diseño y estructurar componentes inl...	FINALIZADA	<div style="width: 50%; background-color: #2ECC71; height: 10px;"></div>		
✓ CS-18 Verificación de base de datos y prototipos finales. E	FINALIZADA	<div style="width: 50%; background-color: #2ECC71; height: 10px;"></div>		
✓ CS-19 Sprint Review y Retrospective.	FINALIZADA	<div style="width: 50%; background-color: #2ECC71; height: 10px;"></div>		
✓ CS-3 Desarrollo del Módulo de Inventario y Ventas	FINALIZADA	<div style="width: 50%; background-color: #3498DB; height: 10px;"></div>		
✓ CS-20 Desarrollar CRUD completo para inventarios y ventas.	FINALIZADA	<div style="width: 50%; background-color: #3498DB; height: 10px;"></div>		
✓ CS-21 Implementar notificaciones automáticas tras ventas....	FINALIZADA	<div style="width: 50%; background-color: #3498DB; height: 10px;"></div>		
✓ CS-22 Pruebas unitarias de cada módulo (inventarios y venta...	FINALIZADA	<div style="width: 50%; background-color: #3498DB; height: 10px;"></div>		
✓ CS-23 Corrección de errores identificados durante pruebas. D	FINALIZADA	<div style="width: 50%; background-color: #3498DB; height: 10px;"></div>		
✓ CS-24 Sprint Review y Retrospective.	FINALIZADA	<div style="width: 50%; background-color: #3498DB; height: 10px;"></div>		
✓ CS-4 Desarrollo de Reportes y Gestión de Roles	FINALIZADA	<div style="width: 50%; background-color: #F0E68C; height: 10px;"></div>		
✓ CS-25 Crear funcionalidad para reportes diarios y mensuale...	TAREAS PO...	<div style="width: 50%; background-color: #F0E68C; height: 10px;"></div>		
✓ CS-26 Implementar gestión de roles y permisos en el sistem...	TAREAS PO...	<div style="width: 50%; background-color: #F0E68C; height: 10px;"></div>		
✓ CS-27 Pruebas de funcionalidad de reportes y gestión de role...	TAREAS PO...	<div style="width: 50%; background-color: #F0E68C; height: 10px;"></div>		
✓ CS-28 Ajustes a funcionalidades según retroalimentación.	TAREAS PO...	<div style="width: 50%; background-color: #F0E68C; height: 10px;"></div>		
✓ CS-29 Sprint Review y Retrospectiv...	TAREAS PO...	<div style="width: 50%; background-color: #F0E68C; height: 10px;"></div>		
✓ CS-5 Pruebas Finales y Despliegue	FINALIZADA	<div style="width: 50%; background-color: #E74C3C; height: 10px;"></div>		
✓ CS-30 Pruebas integrales del sistema complet...	TAREAS PO...	<div style="width: 50%; background-color: #E74C3C; height: 10px;"></div>		
✓ CS-31 Corrección de errores críticos.	TAREAS PO...	<div style="width: 50%; background-color: #E74C3C; height: 10px;"></div>		
✓ CS-32 Documentación final del sistema (manual de usuario, diccionario...	TAREAS PO...	<div style="width: 50%; background-color: #E74C3C; height: 10px;"></div>		
✓ CS-33 Capacitación del personal sobre el manejo del sistem...	TAREAS PO...	<div style="width: 50%; background-color: #E74C3C; height: 10px;"></div>		
✓ CS-34 Despliegue y revisión final con el Product Own...	TAREAS PO...	<div style="width: 50%; background-color: #E74C3C; height: 10px;"></div>		

DIAGRAMAS UML

- Diagrama de Componentes

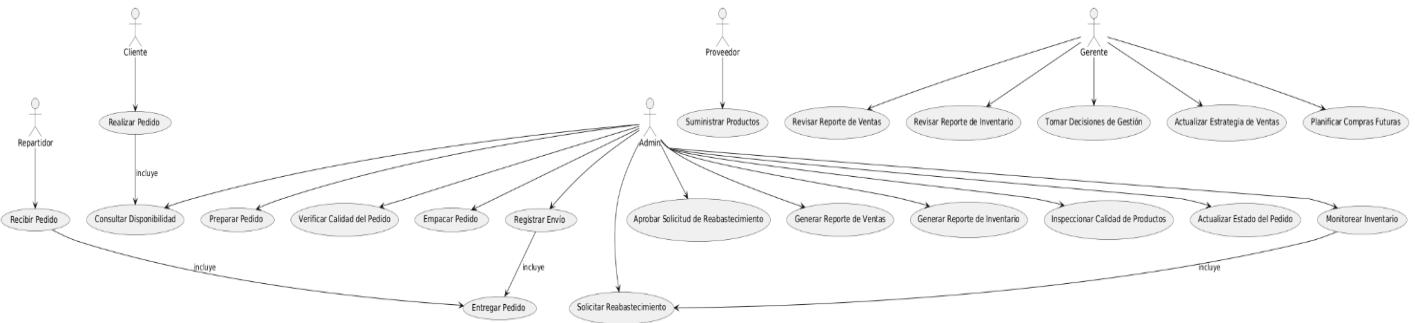


● Diagrama de Actividad

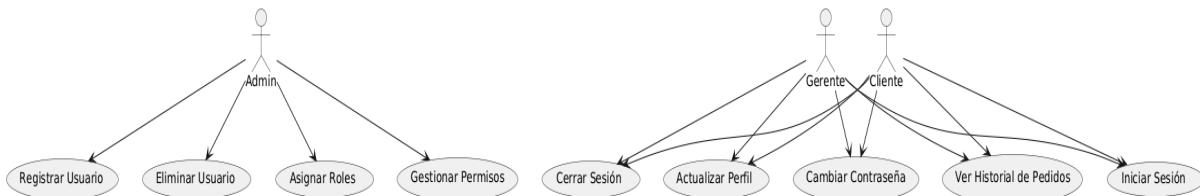


● Diagrama Casos de Uso

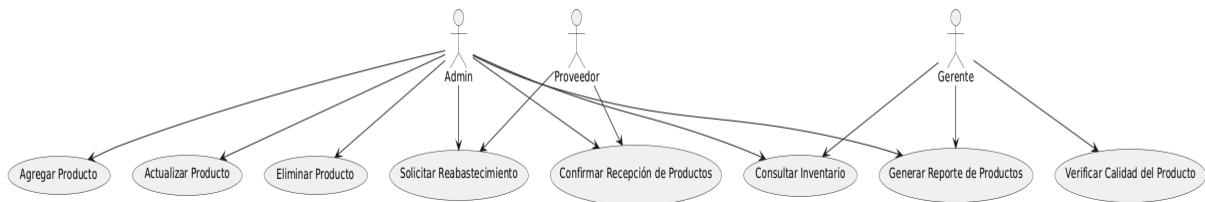
○ Diagrama de Casos de Uso General



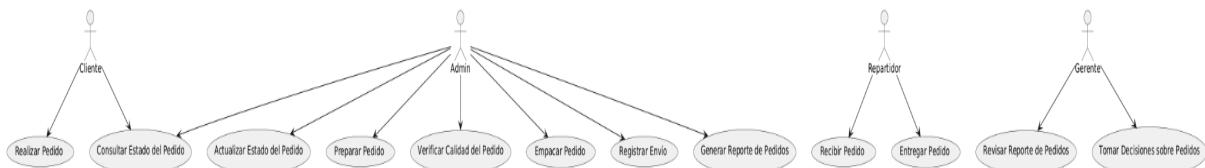
○ Diagrama de Casos de Uso Gestión de Usuario



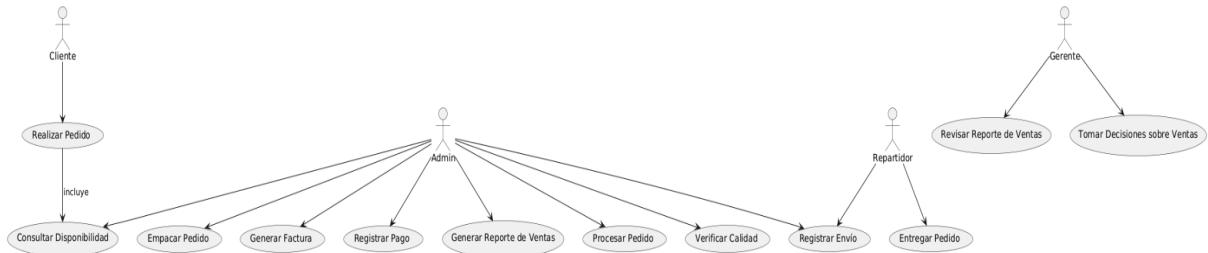
○ Diagrama de Casos de Uso Gestión de Productos



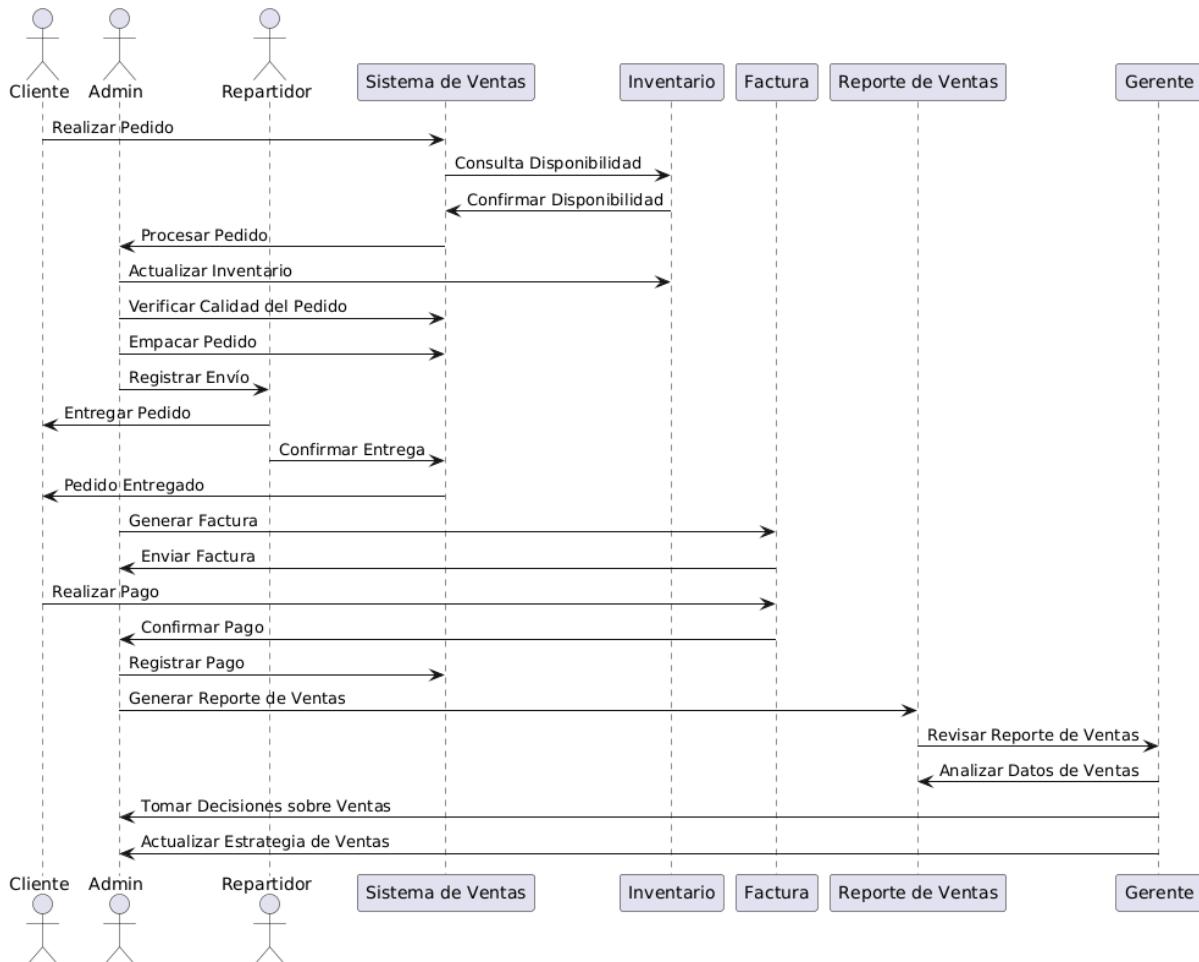
○ Diagrama de Casos de Uso Gestión de Pedidos



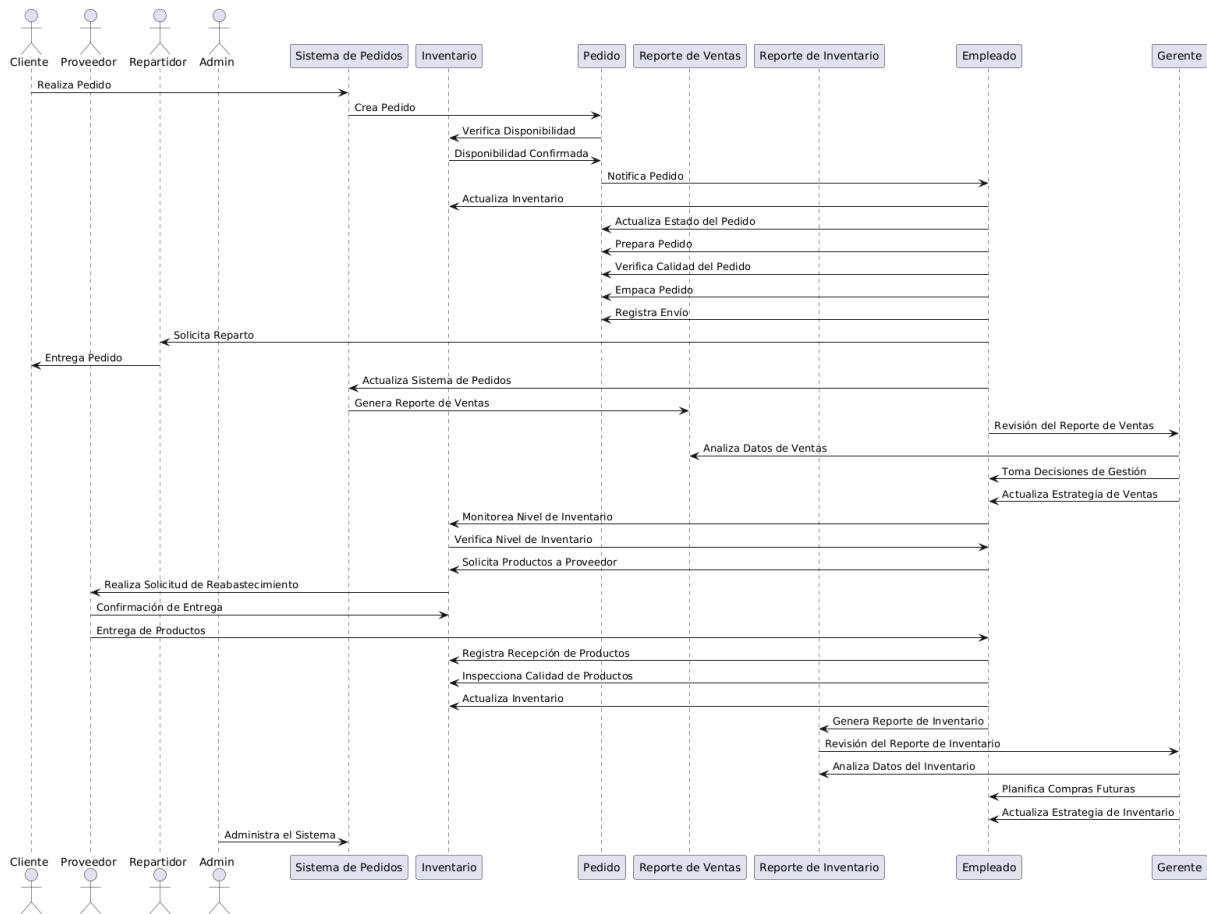
- Diagrama de Casos de Uso Gestión para Realizar una Venta



- Diagrama de Interacción



● Diagrama de secuencia



TAREAS DE MANTENIMIENTO Y PRUEBA

Mantenimiento Adaptativo

1. Integración de nuevos métodos de pago:

Dada la creciente demanda por pagos con criptomonedas en Bolivia, el sistema debe adaptarse para soportar estas transacciones. Esto implica:

- Implementar APIs de plataformas de criptomonedas reconocidas que operen en Bolivia, como Binance o Bitso, para procesar pagos.
- Modificar la interfaz de pago para incluir la opción de criptomonedas, mostrando los valores convertidos a bolivianos (Bs).

- Garantizar la claridad en las facturas, especificando el monto en criptomonedas y su equivalente en bolivianos según la tasa de cambio.

2. Expansión internacional:

Ante la posibilidad de expandir el negocio a países vecinos como Perú o Brasil, se debe adaptar el sistema para:

- Soportar los idiomas español y portugués, incluyendo la traducción de todos los textos en la interfaz.
- Manejar formatos de fecha (dd/mm/aaaa) y moneda locales (soles peruanos o reales brasileños).
- Verificar que el sistema cumpla con las regulaciones fiscales de cada país.

Mantenimiento Correctivo

1. Errores de facturación:

Si un cliente reporta un doble cargo por su compra, los pasos a seguir serán:

- Revisar el código del módulo de facturación para identificar el error que causa la duplicación de cargos.
- Corregir el error en el algoritmo responsable del procesamiento de pagos.
- Realizar pruebas exhaustivas para garantizar que el problema ha sido solucionado y no afecta otros procesos.

2. Vulnerabilidades de seguridad:

Si se detecta una vulnerabilidad que expone datos sensibles, las acciones incluyen:

- Aplicar actualizaciones de seguridad necesarias, como parches para cumplir con normativas bolivianas de protección de datos (Ley 164).
- Revisar y fortalecer el código afectado, utilizando herramientas de análisis de seguridad.
- Realizar pruebas de penetración para confirmar que el sistema está protegido contra ataques futuros.

Mantenimiento Perfectivo

1. Optimización de la búsqueda de productos:

Si los usuarios encuentran lenta la funcionalidad de búsqueda, se implementarán mejoras como:

- Uso de índices de búsqueda en la base de datos para acelerar las consultas.
- Rediseño del algoritmo de búsqueda para hacerlo más eficiente.
- Realización de pruebas de rendimiento para evaluar el impacto de los cambios en un entorno simulado de ventas en Bolivia.

2. Actualización de la interfaz de usuario:

Para mejorar la experiencia del usuario:

- Se rediseñará la interfaz del sistema basándose en retroalimentación de usuarios bolivianos.
- Se aplicarán principios de diseño intuitivo, como botones claros y flujos de navegación simplificados.
- Se realizarán pruebas de usabilidad para asegurar la aceptación de los cambios, incluyendo encuestas a usuarios locales.

Mantenimiento Preventivo

1. Realización de backups regulares:

Para prevenir pérdidas de datos:

- Se configurará un sistema de copias de seguridad automáticas de la base de datos cada noche.
- Los respaldos se almacenarán en servidores seguros ubicados en Bolivia o en la nube con soporte local.
- Se verificará periódicamente la integridad de los respaldos mediante pruebas de recuperación.

2. Monitoreo del rendimiento del sistema:

Para identificar posibles problemas antes de que afecten a los usuarios:

- Se implementarán herramientas de monitoreo como Prometheus o Nagios para alertar sobre aumentos inusuales en los tiempos de respuesta.
- Se realizará un análisis mensual de los datos de rendimiento.
- Se establecerá un protocolo de acción rápida en caso de detectar problemas que afecten al mercado boliviano.

PRUEBAS

Pruebas Funcionales

1. Pruebas de unidad:

Verificar que cada componente individual del sistema funciona correctamente. Estas pruebas se enfocan en módulos específicos como el cálculo de precios o la generación de facturas.

2. Pruebas de integración:

Asegurar que diferentes módulos o componentes funcionan bien juntos, como la integración entre el carrito de compras y el módulo de facturación.

3. Pruebas de sistema:

Validar el sistema completo para verificar que cumple con los requisitos especificados, simulando un proceso completo desde el registro del usuario hasta la entrega del pedido.

4. Pruebas de aceptación:

Confirmar que el sistema satisface las necesidades y expectativas del cliente, asegurando que todas las funcionalidades requeridas estén implementadas correctamente.

Pruebas No Funcionales

1. Pruebas de rendimiento:

Evaluar la rapidez y eficiencia del sistema bajo diferentes condiciones de carga, como manejar 1000 pedidos simultáneos.

2. Pruebas de seguridad:

Identificar vulnerabilidades en el sistema y asegurar que los datos están protegidos contra accesos no autorizados mediante pruebas de penetración.

3. Pruebas de usabilidad:

Evaluar cuán fácil es para los usuarios interactuar con el sistema, asegurándose de que la navegación sea intuitiva y los elementos visuales sean claros.

4. Pruebas de compatibilidad:

Verificar que el sistema funcione en diferentes dispositivos, navegadores y plataformas para garantizar una experiencia uniforme.

Pruebas de Mantenimiento

1. Pruebas de regresión:

Asegurar que las nuevas actualizaciones no afecten negativamente las funcionalidades existentes del sistema.

2. Pruebas de recuperación:

Verificar que el sistema pueda recuperarse correctamente de fallos o problemas, como una caída del servidor o un corte de energía.

Pruebas Automatizadas

1. Scripts de prueba:

Utilizar scripts automatizados para realizar pruebas repetitivas y validar que el sistema sigue funcionando después de cambios o actualizaciones.

2. Monitoreo continuo:

Implementar herramientas de monitoreo para detectar y alertar sobre problemas de manera proactiva, como tiempos de respuesta lentos o errores en el sistema.

Pruebas Manuales

1. Pruebas exploratorias:

Realizar pruebas manuales para descubrir problemas que no hayan sido identificados por pruebas automatizadas.

2. Pruebas de usuario final:

Permitir que usuarios reales utilicen el sistema y proporcionen feedback sobre su funcionamiento, identificando posibles áreas de mejora.

Ejemplos Específicos

1. Pruebas de facturación:

Verificar que todas las transacciones se registren y procesen correctamente, y que los clientes reciban facturas exactas.

2. Pruebas de catálogo:

Asegurarse de que los productos y sus descripciones se muestren correctamente en el catálogo del sistema.

3. Pruebas de búsqueda:

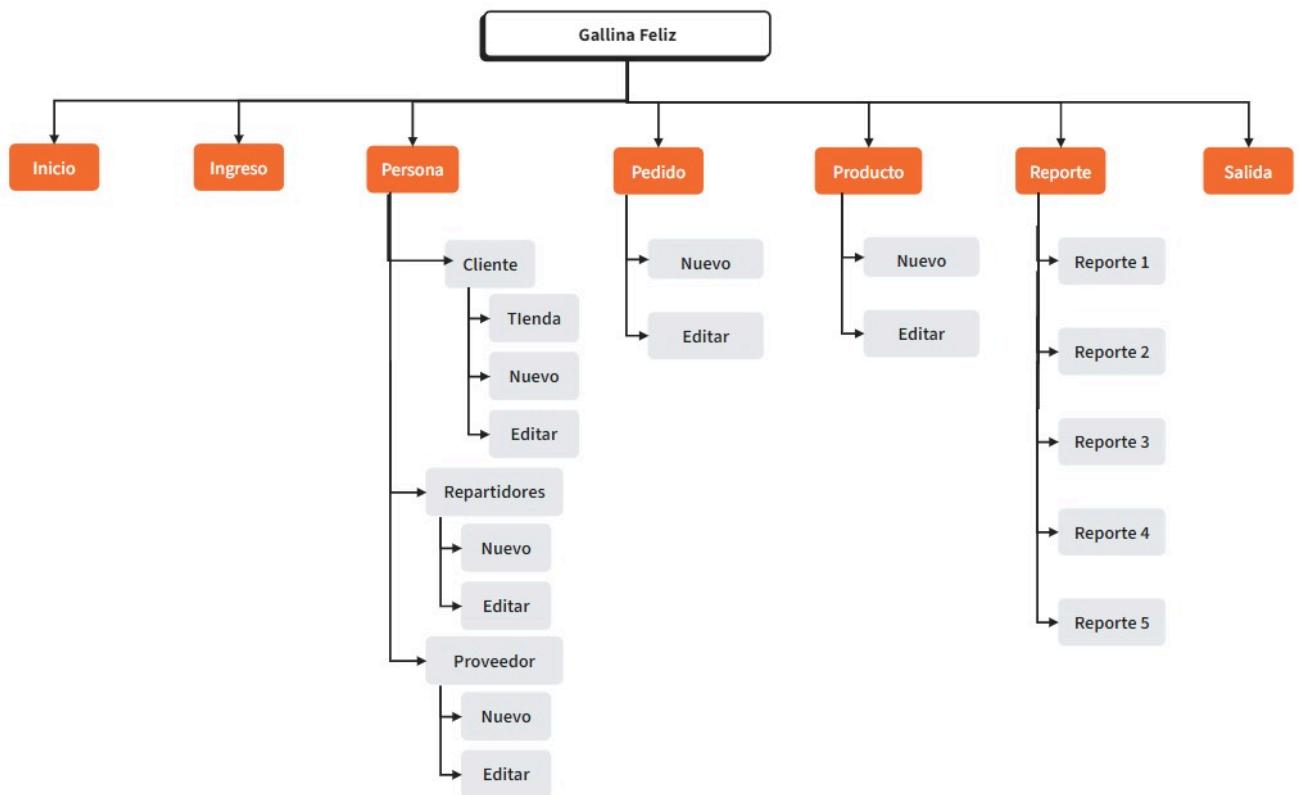
Evaluar que el buscador del sistema encuentre y muestre productos de manera eficiente y precisa.

4. Pruebas de interfaz de usuario:

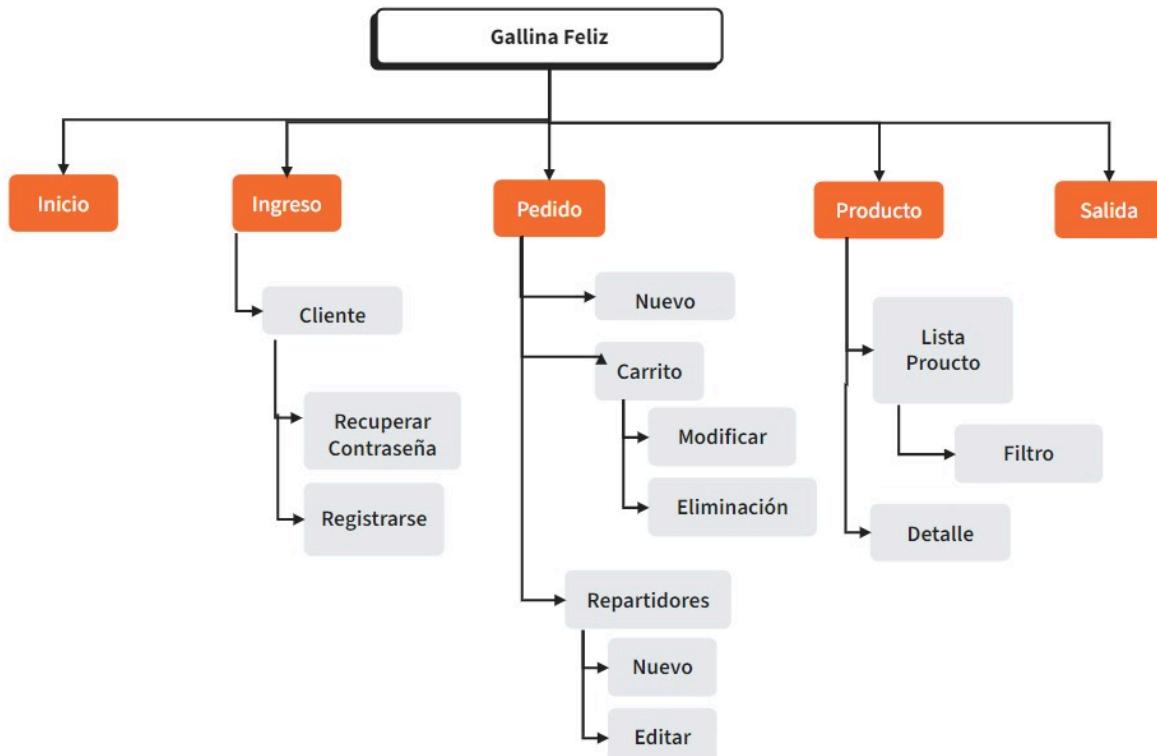
Comprobar que todos los elementos de la interfaz de usuario funcionan correctamente y son intuitivos para los usuarios.

MAPEO DEL SISTEMA

MAPEO DEL SISTEMA (ADMINISTRADOR)



MAPEO DEL SISTEMA (CLIENTE)



PANTALLAS MUERTAS DEL SISTEMA

SISTEMA DE GESTIÓN "DONA CHELA AL POR MAYOR"

Pepe Administrador

INICIO

- Personas
- Pedidos
- Productos
- Reportes
- Grafica

INFORMACION

Placeholder text: Lorem ipsum dolor sit amet consectetur adipisicing elit. Veritatis aliquam distinctio soluta sit laudantium, et rerum quidem, ipsum qui laboriosam consequatur quo error voluptate earum aspernatur commodi eos aliquid odio!

CONTACTOS

Placeholder text: Lorem ipsum dolor sit amet consectetur adipisicing elit. Veritatis aliquam distinctio soluta sit laudantium, et rerum quidem, ipsum qui laboriosam consequatur quo error voluptate earum aspernatur commodi eos aliquid odio!

SISTEMA DE GESTIÓN "DONA CHELA AL POR MAYOR"

Pepe Administrador

CLIENTES

CI	NOMBRE COMPLETO	DIRECCIÓN	CELULAR	ANTIGÜEDAD	TIENDAS	Opciones
10002	CACERES MAMANI MARIA	Av juan pablo segundo	456456	1 AÑOS		
22334455	Gonzalez Serrano Lucia	Calle Nueva 12	22233444	2 AÑOS		
87654321	Lopez Martinez Ana	Calle Falsa 456	987654321	3 AÑOS		
33445566	Martinez Alvarez Pedro	Av. Los Alamos 34	33444555	1 AÑOS		
1445	Mercedez Mamani Maria	Av juan pablo segundo	142555	0 AÑOS		
12345678	Perez Gonzalez Juan	Av. Siempre Viva 123	123456789	5 AÑOS		
44556677	Ramirez Cutierrez Marta	Calle Sol 56	444555666	1 AÑOS		

PROVEEDORES

OPCIONES

CI	NOMBRE	CORREO	TELÉFONO	DIRECCIÓN	OPCIONES
22334455	Gonzalez Serrano Lucia	proveedor2@example.com	987654321	Calle Nueva 12	
33445566	Martinez Alvarez Pedro	proveedor3@example.com	555123456	Av. Los Alamos 34	
55667788	Morales Fernandez Jose	proveedor5@example.com	777345678	Calle Luna 89	
44556677	Ramirez Gutierrez Marta	proveedor4@example.com	666789012	Calle Sol 56	
11223344	Sanchez Ramirez Carlos	proveedor1@example.com	123456789	Boulevard Libertad 789	

INGRESO DE PRODUCTOS

OPCIONES

ID	PRODUCTO	CANTIDAD	FECHA DE INGRESO	PROVEEDOR	OPCIONES
1001	Pequeño	100	Jan. 15, 2024	Carlos Sanchez Ramirez	
1002	Mediano	50	Feb. 18, 2024	Lucia Gonzalez Serrano	
1003	Grande	200	March 8, 2024	Pedro Martinez Alvarez	
1004	Extra Grande	75	April 4, 2024	Marta Ramirez Gutierrez	
1005	Familiar	150	May 22, 2024	Jose Morales Fernandez	

REPORTE 1

Reporte de todos los pedidos realizados por los clientes, los repartidores asignados a los pedidos con su respectivo camión, los productos incluidos en los pedidos, la tienda asociada, además de si en la entrega hubo alguna incidencia o no en dicho pedido del cliente.

OPCIONES

ID PEDIDO	CLIENTE	TIENDA	REPARTIDOR	PLACA DEL CAMION	PRODUCTO	CANTIDAD	PRECIO UNITARIO	DESCUENTO	TOTAL PAGO	HUBO INCIDENCIA
1	Juan Perez Gonzalez	Av. Central 100	Juan Perez Gonzalez	ABC1234	Pequeño	2	10.00	5.00	15.00	None
2	Ana Lopez Martinez	Calle Falsa 200	Ana Lopez Martinez	XYZ5678	Mediano	1	15.00	10.00	5.00	None
5	Marta Ramirez Gutierrez	Boulevard 500	Marta Ramirez Gutierrez	JKL7890	Familiar	1	25.00	10.00	15.00	None
8	Juan Perez Gonzalez	Av. Central 100	Marta Ramirez Gutierrez	JKL7890	Pequeño	250	25.00	10.00	6240.00	None
3	Lucia Gonzalez Serrano	Avenida Libertad 300	Lucia Gonzalez Serrano	DEF9012	Grande	3	12.00	0.00	36.00	None
9	Juan Perez Gonzalez	Av. Central 100	Pedro Martinez Alvarez	GHJ3456	Mediano	15	0.80	0.00	12.00	None
7	Juan Perez Gonzalez	Av. Central 100	Pedro Martinez Alvarez	GHJ3456	Mediano	250	25.00	10.00	6240.00	None
4	Pedro Martinez Alvarez	Calle Sol 400	Pedro Martinez Alvarez	GHJ3456	Extra Grande	2	20.00	5.00	35.00	None
6	Juan Perez Gonzalez	Av. Central 100	Ana Lopez Martinez	XYZ5678	Pequeño	50	25.00	10.00	1240.00	None

SISTEMA DE GESTIÓN
"DOÑA CHELA AL POR MAYOR"

Pepe Administrador

REPORTE 2

Este reporte detalla las compras de productos hechas por el administrador, incluyendo información del proveedor, el producto comprado, detalles de la compra realizada, el número de pedidos de dicho producto, número de clientes que compraron el producto y el número de incidencias del producto.

[Generar PDF](#)

ID COMPRA	ADMINISTRADOR	PROVEEDOR	PRODUCTO	CANTIDAD	PRECIO UNITARIO	TOTAL DE LA COMPRA	FECHA DE LA COMPRA	FECHA DE LA LLEGADA	NÚMERO DE PEDIDOS	NÚMERO CLIENTES	NÚMERO INCIDENCIAS
1001	Carlos Sanchez Ramirez	Carlos Sanchez Ramirez	Pequeño	100	25.50	2550.00	Jan. 10, 2024	Jan. 15, 2024	3	1	3
1002	Lucia Gonzalez Serrano	Lucia Gonzalez Serrano	Mediano	50	30.75	1537.50	Feb. 15, 2024	Feb. 18, 2024	3	2	3
1003	Pedro Martinez Alvarez	Pedro Martinez Alvarez	Grande	200	20.00	4000.00	March 5, 2024	March 8, 2024	1	1	1
1004	Marta Ramirez Gutierrez	Marta Ramirez Gutierrez	Extra Grande	75	15.80	1185.00	April 1, 2024	April 4, 2024	2	2	2
1005	Jose Morales Fernandez	Jose Morales Fernandez	Familiar	150	22.50	3375.00	May 20, 2024	May 22, 2024	1	1	1

ACCOUNT

[Help](#)

[Logout](#)

SISTEMA DE GESTIÓN
"DOÑA CHELA AL POR MAYOR"

Pepe Administrador

REPORTE 3

Reporte de los clientes, mostrando sus datos, número de tiendas que tienen, el número de pedidos que ha realizado, la media de descuento que se le ha dado, el número de repartidores que lo entiendieron, el producto que más pide (el primero), quien es el proveedor de ese producto.

[Generar PDF](#)

CI CLIENTE	NOMBRE CLIENTE	SEXO	CELULAR	EDAD	NÚMERO DE TIENDAS	NÚMERO DE PEDIDOS	PROMEDIO DE DESCUENTOS AL CLIENTE	NÚMERO DE REPARTIDORES QUE LO ATENDIERON	PRODUCTO MÁS COMPRADO POR EL CLIENTE	PROVEEDOR DEL PRODUCTO
12345678	Juan Perez Gonzalez	M	123456789	39	1	6	8	5	Pequeño	Carlos Sanchez Ramirez
87654321	Ana Lopez Martinez	F	987654321	34	1	1	10	1	Mediano	Carlos Sanchez Ramirez
22334455	Lucia Gonzalez Serrano	F	222333444	32	1	1	0	1	Grande	Carlos Sanchez Ramirez
33445566	Pedro Martinez Alvarez	M	333444555	36	1	1	5	1	Extra Grande	Carlos Sanchez Ramirez
44556677	Marta Ramirez Gutierrez	F	444555666	29	1	1	10	1	Familiar	Carlos Sanchez Ramirez
1445	Maria Mercedez Mamani	F	142555	14	0	0	0	0	None	Carlos Sanchez Ramirez
10002	MARIA CACERES MAMANI	F	456456	26	0	0	0	0	None	Carlos Sanchez Ramirez

ACCOUNT

[Help](#)

[Logout](#)

SISTEMA DE GESTIÓN
"DOÑA CHELA AL POR MAYOR"

Pepe Administrador

REPORTE 4

Reporte de los pedidos de mayor cantidad que se hicieron, junto al nombre de los clientes que la realizaron.

[Generar PDF](#)

ID PEDIDO	FECHA PEDIDO	CANTIDAD	TAMAÑO DEL HUEVO	DESCUENTO ASIGNADO	CI CLIENTE	NOMBRE DEL CLIENTE
7	May 7, 2024	250	Mediano	10.00	12345678	Juan Perez Gonzalez
8	May 7, 2024	250	Pequeño	10.00	12345678	Juan Perez Gonzalez

ACCOUNT

[Help](#)

[Logout](#)

SISTEMA DE GESTIÓN
"DONA CHELA AL POR MAYOR"

Pepe Administrador

MENU

- Inicio
- Personas
- Pedidos
- Productos
- Reportes
- Grafica

ACCOUNT

- Help
- Logout

REPORTE 5

Listado de todos los clientes con una antigüedad mayor a 3 años, el número de pedidos que realizaron, el número de tiendas a las que se le hacen sus envíos.

[Generar PDF](#)

CI	NOMBRE	PATERNO	MATERNO	CELULAR	ANTIGÜEDAD	NUMERO DE TIENDAS
12345678	Juan	Perez	Gonzalez	123456789	5	6
87654321	Ana	Lopez	Martinez	987654321	3	1

SISTEMA DE GESTIÓN
"DONA CHELA AL POR MAYOR"

Pepe Administrador

MENU

- Inicio
- Personas
- Pedidos
- Productos
- Reportes
- Grafica

ACCOUNT

- Help
- Logout

Número de ventas de cada mes del año 2024

Número de ventas de cada mes del año 2024 para ver los meses de más tendencia que tuvo la distribuidora

Descripción de la gráfica	Num ventas
Enero	1
Mayo	8
Junio	1