



**Universidad  
Tecnológica  
del Perú**

## **FACULTAD DE INGENIERÍA**

**INGENIERÍA DE SISTEMAS E INFORMÁTICA Y SOFTWARE**

### **Integrador I: Sistemas Software**

**“SISTEMA DE CONTROL DE INVENTARIO DE LA JOYERÍA  
YOURSHINE.PE”**

### **ESTUDIANTES:**

DÍAZ SÁENZ FABIANA SOFÍA	U22211183
GERONIMO ORMEÑO FABIANA	U22101458
YARIHUAMAN RAMÍREZ DANIELA	U20307632

### **CICLO:**

2024 - 1

Lima, Perú

# Índice

<b>INTRODUCCIÓN .....</b>	<b>5</b>
<b>1. CAPÍTULO 1 – ASPECTOS GENERALES.....</b>	<b>6</b>
1.1. Descripción del problema: .....	6
1.2. Definición de objetivos: .....	13
<b>1.2.1. Objetivo principal: .....</b>	<b>13</b>
<b>1.2.2. Objetivos específicos:.....</b>	<b>13</b>
1.3. Alcances y Limitaciones: .....	14
<b>1.3.1. Alcances: .....</b>	<b>14</b>
<b>1.3.2. Limitaciones:.....</b>	<b>16</b>
1.4. Justificación.....	17
1.5. Estado del arte:.....	17
<b>2. CAPÍTULO 2 – MARCO TEÓRICO .....</b>	<b>21</b>
2.1. Marco teórico de programación orientada a objetos .....	21
<b>2.1.1. Metodología: .....</b>	<b>21</b>
2.1.2. Modelo: .....	27
<b>2.1.3. Patrones GUI:.....</b>	<b>29</b>
<b>2.1.4. Patrones de desarrollo:.....</b>	<b>34</b>
<b>2.1.5. Lenguajes de programación o frameworks: .....</b>	<b>37</b>
<b>2.1.6. Gestor de base de datos: .....</b>	<b>46</b>
<b>2.1.7. Herramientas adicionales: .....</b>	<b>49</b>
2.2. Marco teórico de las formulas y/o sistemas:.....	55
<b>3. CAPÍTULO 3 – DESARROLLO DE LA SOLUCIÓN .....</b>	<b>59</b>
3.1. Prototipos .....	59
3.2. Diagramas .....	61
<b>3.2.1. Modelo de negocios .....</b>	<b>61</b>
<b>3.2.2. Diagrama clases .....</b>	<b>62</b>
<b>3.2.3. Diagrama de entidad – relación .....</b>	<b>63</b>
<b>3.2.4. Diccionario de datos .....</b>	<b>64</b>
<b>BIBLIOGRAFÍA.....</b>	<b>65</b>

## INDICE DE FIGURAS

Figura 1: Diagrama del árbol.....	7
Figura 2: Estructura de lista de stock disponible en Oddo .....	18
Figura 3. Procesos Scrum. Frechina, A. ....	19
Figura 4. Product Backlog .....	23
Figura 5. Stakeholder. ....	25
Figura 6. Pantalla Modales.....	30
Figura 7. Barra de navegación.....	31
Figura 8. Dashboard .....	32
Figura 9. Captcha .....	33
Figura 10. Cards .....	33
Figura 11. MVC.....	35
Figura 12. JavaScript .....	38
Figura 13. HTML5 .....	41
Figura 14. NodeJS.....	43
Figura 15. Bootstrap 5.....	44
Figura 16. React .....	45
Figura 17. MySQL .....	46
Figura 18. Balsamiq.....	49
Figura 19. Bizagi .....	51
Figura 20. Draw.io .....	53
Figura 21. Canva.....	55
Figura 22. Login .....	59
Figura 23. Dashboard .....	60
Figura 24. Inventario .....	60
Figura 25. Modelo de negocios.....	61
Figura 26. Diagrama de clases.....	62
Figura 27. Diagrama de entidad relación.....	63

## INDICE DE TABLAS/CUADROS

<b>Tabla 1:</b> Información para graficar el indicador de calidad de los pedidos generados. ....	8
<b>Tabla 2:</b> Información para graficar el indicador de entregas perfectamente recibidas .....	9
<b>Tabla 3:</b> Información para graficar el indicador de entregas perfectamente recibidas .....	10
<b>Tabla 4:</b> Información para graficar el indicador de entregas perfecto. ....	12
Tabla 5: Requerimientos funcionales.....	14
Tabla 6: Requerimientos funcionales.....	15
Tabla 7: Requerimientos No funcionales .....	16
<b>Tabla 8:</b> Cuadro comparativo .....	26

## INDICE DE GRAFICOS

Gráfico 1: Indicador de calidad de pedidos generados.....	9
Grafico 2: Indicador de entregas perfectamente recibidas .....	10
Gráfico 3: Indicador de transporte .....	11
Gráfico 4: Indicador de entregas a tiempo .....	12

## INDICE DE ANEXOS

Anexo 1. Lean Canvas .....	66
----------------------------	----

## INTRODUCCIÓN

En los últimos años, la pandemia ha impulsado a muchas personas a emprender, generando una fuerte competencia en diversos sectores, incluido el de la joyería. En este contexto, "yourshine.pe", una empresa especializada en la venta de aretes de plata, ha experimentado una serie de desafíos relacionados con la gestión de su inventario. El principal problema es la acumulación de productos de baja demanda, lo que no solo aumenta los costos de almacenamiento, sino que también eleva el riesgo de desvalorización, afectando la rentabilidad y sostenibilidad del negocio a largo plazo.

El origen de este proyecto surge de la necesidad de solucionar estos problemas de inventario, que no son únicos de "yourshine.pe", sino que afectan a muchas empresas del sector. La idea de desarrollar una aplicación de escritorio que optimice el control de inventario tiene como objetivo evitar la acumulación de productos poco demandados y asegurar una rotación eficiente, ajustada a las tendencias del mercado. Anteriormente, la empresa ha utilizado métodos tradicionales para gestionar su inventario, pero estos se han mostrado ineficaces frente a los cambios rápidos en las preferencias de los consumidores y el crecimiento de la competencia.

El proyecto se propone diseñar una herramienta tecnológica que controle el inventario, además de permitir realizar previsiones de demanda basadas en datos de ventas y preferencias del cliente. Este enfoque mejorará la eficiencia operativa y permitirá a "yourshine.pe" competir de manera más efectiva en el mercado actual, donde la agilidad y la precisión en la gestión de inventarios son claves. El desarrollo de este sistema responde a la necesidad urgente de muchas empresas de adaptarse a los cambios en los hábitos de consumo, y su importancia radica en que puede convertirse en un modelo para otros emprendimientos del sector que enfrenten desafíos similares. La optimización del inventario no solo impacta en los costos operativos, sino que también puede marcar la diferencia entre el éxito y el fracaso de un negocio en un entorno tan competitivo.

## 1. CAPÍTULO 1 – ASPECTOS GENERALES

### 1.1. Descripción del problema:

La joyería "yourshine.pe," es una empresa especializada en la venta de aretes de plata, enfrenta desafíos importantes en la gestión de su inventario, lo que impacta negativamente tanto en su operatividad como en su rentabilidad. Uno de los principales problemas es la acumulación de productos menos demandados, lo que aumenta el riesgo de desvalorización del inventario.

En primer lugar, la recepción de productos inadecuada por parte del personal de inventario se presenta como un problema debido a la falta de un proceso estructurado, lo que puede resultar en errores en la cantidad o calidad de los aretes recibidos. Esto genera discrepancias entre lo que se recibe y lo que se registra, dificultando el control inicial del inventario.

Además, hay un registro ineficiente de productos, ya que la información sobre los nuevos artículos no se está registrando de manera precisa en el sistema. Esto afecta la capacidad de rastrear el inventario y provoca confusión sobre la disponibilidad de productos. La falta de clasificación y filtrado adecuado por parte del Gerente de inventario de los productos no son gestionados adecuadamente según su rotación, dificultando la identificación de aquellos que tienen baja demanda, resultando en acumulaciones de productos menos demandados.

La gestión de inventario en tiempo real limitada impide tomar decisiones informadas sobre reabastecimiento y ventas, afectando las transacciones al no contar con información precisa sobre la disponibilidad de productos. Asimismo, las decisiones de reabastecimiento deficientes surgen de la falta de datos históricos adecuados, lo que puede llevar a un exceso o falta de stock, afectando directamente la satisfacción del cliente y las ventas.

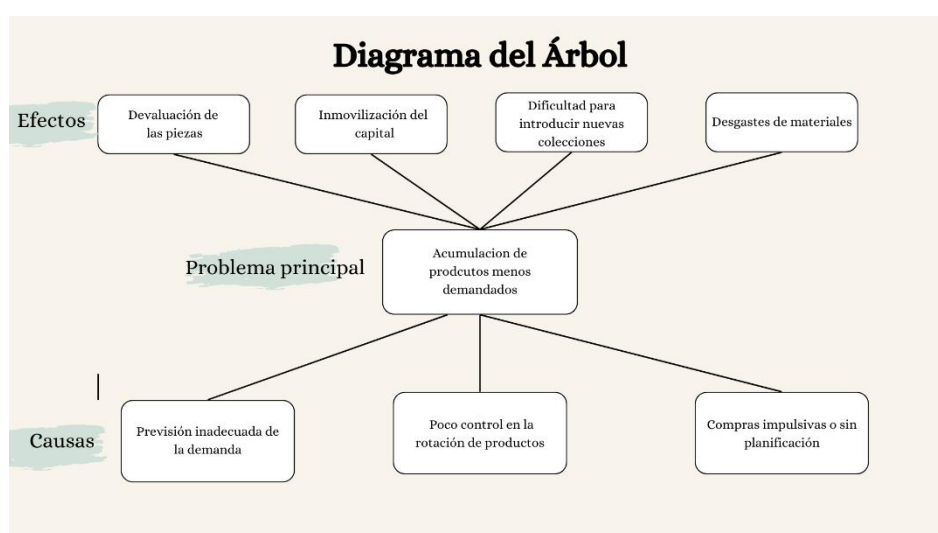
Por otro lado, las auditorías físicas difíciles son consecuencia de la falta de un proceso robusto para realizarlas, lo que puede resultar en

discrepancias significativas entre el inventario registrado y el real. Esto disminuye la confianza en los datos del inventario y puede llevar a pérdidas financieras. Finalmente, la generación de informes por parte del personal de ventas es insuficiente limita la capacidad de la dirección para tomar decisiones estratégicas informadas, ya que la incapacidad para generar informes detallados sobre la rotación de productos y rentabilidad dificulta la identificación de tendencias de mercado y oportunidades de mejora.

Estos problemas no solo afectan las ventas a corto plazo, sino que también representan una amenaza significativa para la sostenibilidad económica a largo plazo de "yourshine.pe". Con el fin de abordar estos desafíos y mejorar la eficiencia operativa, se propone el desarrollo de una aplicación de escritorio diseñada para optimizar el control de inventario. Esta solución se enfocará en gestionar la disponibilidad y rotación de productos, asegurando que la empresa pueda mantener un inventario equilibrado y alineado con la demanda del mercado.

Diagrama del árbol:

Figura 1: Diagrama del árbol



**Nota. Elaboración propia**

Problema central:

El problema central que enfrenta "yourshine.pe" es la acumulación de inventario de productos menos demandados.

Indicadores de Gestión Logísticos (KPI):

**Indicador de calidad de los pedidos generados:**

- Objetivo: Control de la calidad de pedidos generados.
- Definición: Número y porcentaje de compras generadas sin información adicional.
- Periodicidad: Este indicador se calcula cada mes.
- Fuente de información: Se solicitó la información a los encargados de la empresa.

**Gráfico del indicador**

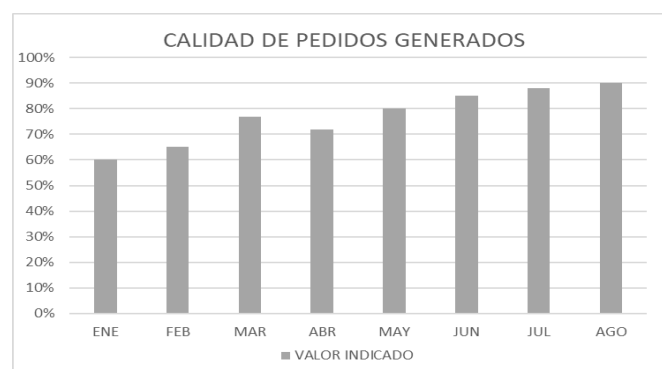
**Tabla 1:** Información para graficar el indicador de calidad de los pedidos generados.

**Indicador:** Calidad de pedidos generados

INFORMACIÓN A INGRESAR			
MES	PEDIDOS GENERADOS SIN PROBLEMAS	TOTAL DE PEDIDOS GENERADOS	VALOR INDICAD O
ENE	45	75	60%
FEB	52	80	65%
MAR	54	70	77%
ABR	72	98	72%
MAY	84	105	80%
JUN	95	112	85%
JUL	101	115	88%
AGO	106	120	90%
Año:2024			



**Gráfico 1:** Indicador de calidad de pedidos generados



**Indicador de entregas perfectamente recibidas**

- Objetivo: Controlar la calidad de los pedidos recibidos y la puntualidad de la entrega de los proveedores.
- Definición: Número y porcentaje de productos que no cumplen con las especificaciones de calidad.
- Periodicidad: Este indicador se calcula cada mes.
- Fuente de información: Se solicitó la información a los encargados de la empresa.

**Gráfico del indicador**

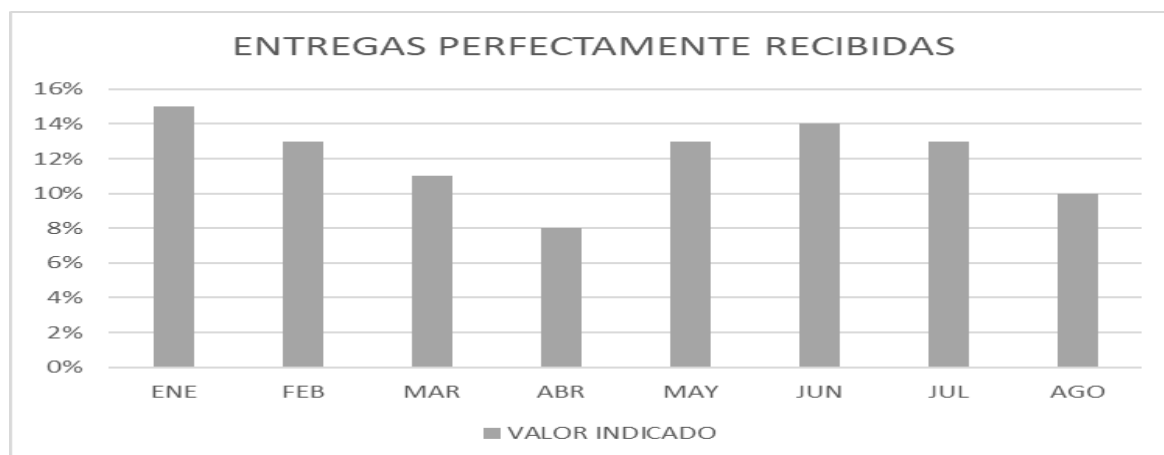
**Tabla 2:** Información para graficar el indicador de entregas perfectamente recibidas

**Indicador:** Entregas perfectamente recibidas

INFORMACIÓN A INGRESAR			
MES	PEDIDOS RECHAZADOS	TOTAL ORDENES DE COMPRAS	VALOR INDICADO
ENE	3	20	15%
FEB	2	16	13%
MAR	3	18	11%
ABR	2	25	8%
MAY	4	30	13%
JUN	3	21	14%
JUL	2	15	13%

<b>AGO</b>	2	20	10%
<b>Año:2024</b>			

**Grafico 2:** Indicador de entregas perfectamente recibidas



### Indicador de transporte

- Objetivo: Controlar el costo de transporte respecto a las ventas de la empresa.
- Definición: Consiste en el rubro respecto a las ventas
- Periodicidad: Este indicador se calcula cada mes.
- Fuente de información: Se solicitó la información a los encargados de la empresa.

### Gráfico del indicador

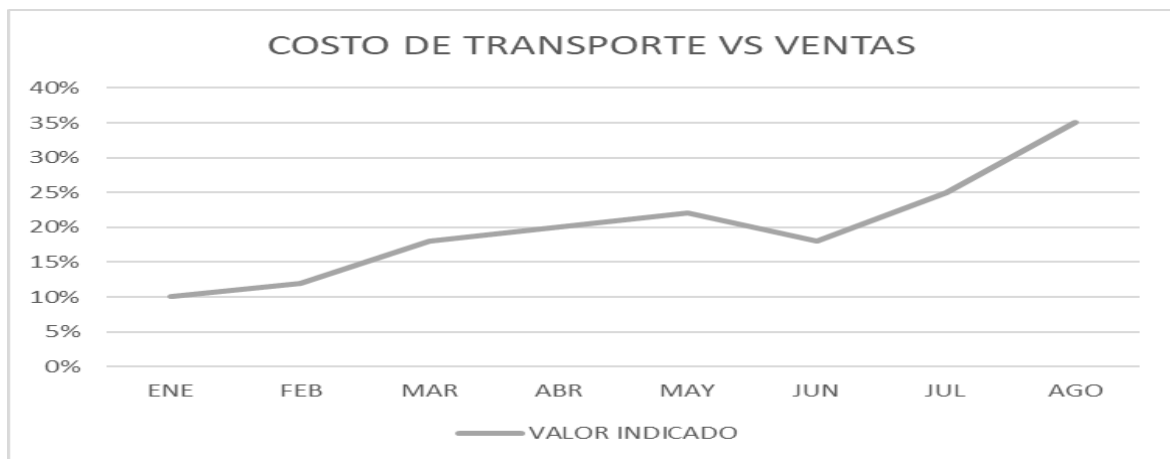
**Tabla 3:** Información para graficar el indicador de entregas perfectamente recibidas

**Indicador:** Costo de transporte

MES	INFORMACIÓN A INGRESAR		
	COSTOS TOTAL TRANSPORTE	TOTAL VALOR VENTAS	DE DE VALOR INDICADO
ENE	325	3250	10%
FEB	420	3500	12%

<b>MAR</b>	630	4000	18%
<b>ABR</b>	960	4800	20%
<b>MAY</b>	990	4500	22%
<b>JUN</b>	1104	5020	18%
<b>JUL</b>	1100	5000	25%
<b>AGO</b>	1925	5500	35%
<b>AÑO: 2024</b>			

**Gráfico 3:** Indicador de transporte



### **Indicador de entregas a tiempo**

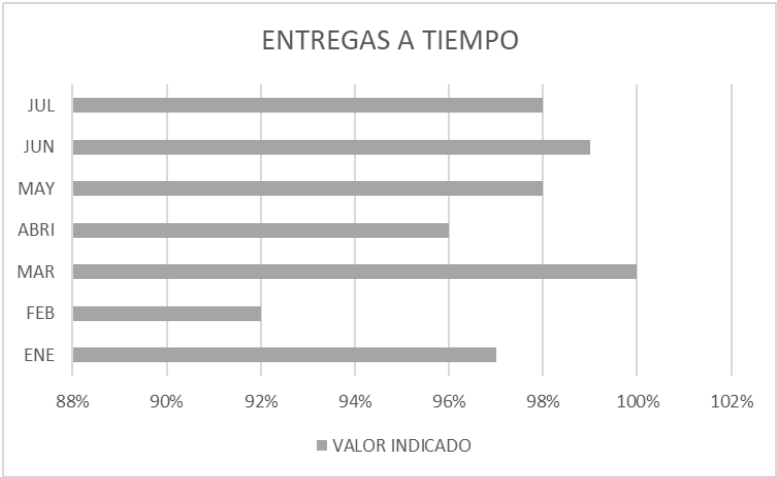
- Objetivo: Controlar el nivel de cumplimiento de las entregas.
- Definición: Este indicador mide el nivel de cumplimiento de la empresa.
- Periodicidad: Este indicador se calcula cada mes.
- Fuente de información: Se solicitó la información a los encargados de la empresa.

**Gráfico del indicador**

**Tabla 4:** Información para graficar el indicador de entregas perfecto.

**Indicador:** Entregas perfectas

INFORMACIÓN A INGRESAR			
MES	ENTREGADOS A TIEMPO	TOTAL PEDIDOS ENTREGADOS	VALOR INDICADO
ENE	58	60	97%
FEB	60	65	92%
MAR	70	70	100%
ABRI	95	98	96%
MAY	98	100	98%
JUN	104	105	99%
JUL	108	110	98%
AÑO:			
2024			



**Gráfico 4:** Indicador de entregas a tiempo

## **Indicador de Producción e inventarios**

### ▪ Rotación de inventario

- Definición: Mide cuántas veces se ha vendido y reemplazado el inventario en un periodo determinado.
- Objetivo: Evaluar la eficiencia de la empresa para vender productos antes de que se desvaloricen
- Calculo:

$$\text{Rotación de inventario} = \frac{\text{Costo de bienes vendidos}}{\text{Inventario promedio}}$$

- Periodicidad: Este indicador se calcula cada mes.
- Fuente de información: Solicitar la información a los encargados de la empresa.

### 1.2. Definición de objetivos:

#### 1.2.1. Objetivo principal:

Mejorar la acumulación de productos menos demandados a través del desarrollo de un sistema de control de inventario.

#### 1.2.2. Objetivos específicos:

- Implementar un módulo que permita conocer las tendencias y preferencias de los clientes para realizar previsiones de demanda más precisas.
- Mantener la información actualizada sobre las necesidades reales del mercado y sobre los productos que se están vendiendo, antes de realizar compras para evitar la acumulación de productos no demandados.

- Crear un módulo de seguimiento constante y detallado de la rotación de productos para identificar rápidamente los productos de alta y baja rotación.

### 1.3. Alcances y Limitaciones:

#### 1.3.1. Alcances:

La aplicación contará con funcionalidades para analizar las ventas pasadas, emitir alertas de reposición, y gestionar el inventario según categorías, con el objetivo de optimizar los niveles de stock y evitar la acumulación de productos con menor demanda.

Especificación de requerimientos funcionales	
Requerimiento funcional N°:	Nombre:
RQF01	Ingreso de la cantidad de inventario
Tipo:	Prioridad:
Jefe de inventarios	Alta
Descripción	
Debe permitir modificar el inventario cuando se añadan o retiren productos, con actualizaciones automáticas en tiempo real.	

**Tabla 5:** Requerimientos funcionales

<b>Especificación de requerimientos funcionales</b>	
Requerimiento funcional N°:	Nombre:
RQF02	Clasificación de productos
Tipo:	Prioridad:
Jefe de inventario /Vendedor	Alta
Descripción	
Los productos deben poder clasificarse según categorías alta demanda o baja demanda para facilitar su gestión.	

**Tabla 6:** Requerimientos funcionales

<b>Especificación de requerimientos no funcionales</b>	
Requerimiento funcional N°:	Nombre:
RQNOF01	Rendimiento de la aplicación
Tipo:	Prioridad:
Desarrollador/Jefe de pruebas	Alta
Descripción	
El sistema debe ser capaz de manejar grandes volúmenes de datos de inventario sin que el rendimiento se vea afectado. Las consultas y actualizaciones deben ser rápidas (menos de 2 segundos de respuesta)	
Manejo de errores:	
El sistema debe manejar las consultas y actualizaciones fallidas mediante mensajes de error claros, sin interrumpir el flujo de trabajo del usuario.	
Criterios de aceptación:	

El sistema debe procesar una consulta de inventario con más de 10,000 productos en menos de 2 segundos.

El sistema debe actualizar la cantidad de stock de un producto en menos de 1 segundo.

**Tabla 7:** Requerimientos No funcionales

### 1.3.2. Limitaciones:

- La implementación de la aplicación podría requerir un periodo de adaptación para los empleados, lo que podría retrasar la plena efectividad del sistema y generar resistencia al cambio.
- El desarrollo e implementación de la aplicación puede representar una inversión significativa, que podría ser un reto financiero para la empresa si no se ven los beneficios a corto plazo.
- La necesidad de actualizaciones continuas y mantenimiento de la aplicación podría requerir recursos adicionales, tanto en términos de tiempo como de dinero, lo que podría no estar contemplado en el presupuesto inicial.
- El periodo para desarrollar la aplicación será de 15 semanas lo que reduce el alcance del proyecto, la profundidad de las funcionalidades a implementar, el tiempo disponible para pruebas y ajustes.
- El sistema de control de inventario, al ser diseñado para escritorio, limita la accesibilidad a usuarios que necesiten acceder al sistema desde dispositivos móviles o ubicaciones fuera de la oficina, lo que podría afectar la



#### 1.4. Justificación

La adopción de un sistema integral de gestión de inventario en "yourshine.pe," una joyería especializada en la venta de aretes de plata, es fundamental para abordar los desafíos que enfrenta en la administración de sus productos. A lo largo de sus operaciones diarias, se han identificado problemas recurrentes, como la acumulación de productos menos demandados, que han impactado negativamente tanto en la rentabilidad como en la eficiencia operativa de la empresa.

Estos problemas han generado una inmovilización de capital significativa, ya que los recursos que podrían utilizarse en productos de mayor demanda están atados a inventarios que se mueven lentamente. Además, la falta de un control de inventario limita la capacidad de "yourshine.pe" para responder a la demanda del mercado de manera oportuna, lo que puede resultar en oportunidades de venta perdidas y una disminución en la satisfacción del cliente.

En respuesta a estos desafíos, la implementación de un sistema de control de inventario se presenta como una solución necesaria y urgente. Este sistema permitirá una gestión más eficiente del inventario, asegurando que los productos estén disponibles cuando los clientes los requieran, lo que mejorará la experiencia del cliente y fortalecerá la posición competitiva de la empresa. Además, ayudará a liberar capital inmovilizado y a evitar pérdidas financieras asociadas con el exceso de stock y la obsolescencia de productos, contribuyendo a la sostenibilidad y crecimiento a largo plazo de "yourshine.pe". Esta iniciativa no sólo optimizará la operatividad, sino que también establecerá una base sólida para el futuro éxito del negocio.

#### 1.5. Estado del arte:

Para crear un sistema que ayude a manejar el inventario de una joyería, necesitamos buenas ideas y ejemplos a seguir. En este contexto, hemos decidido utilizar como referencia sistemas ERP reconocidos, como Odoo.

Estos sistemas son muy buenos para seguir la pista de cada producto y controlar cuántos hay en la tienda.

Usar estos ejemplos nos ayuda a entender cómo hacer nuestro propio sistema, que sea fuerte y funcione bien para una joyería. En este proyecto, vamos a ver cómo podemos usar cosas importantes como números únicos para cada joya y grupos de joyas parecidas.

### 1.5.1. Odoo

Odoo es un sistema de gestión empresarial (ERP) de código abierto, reconocido por su versatilidad y capacidad para integrar diversas áreas operativas de una empresa, incluyendo la gestión de inventarios. En una joyería, donde cada pieza puede ser única, la trazabilidad es fundamental. Odoo facilita esto mediante el uso de números de serie y lotes, permitiendo un seguimiento detallado de cada artículo desde su entrada hasta su venta o devolución.

Implementar un sistema de trazabilidad similar desde cero, inspirado en Odoo, permite a una joyería controlar con precisión su inventario. Asignando identificadores únicos a cada pieza, se asegura un rastreo completo de su ciclo de vida, mejorando la eficiencia operativa y protegiendo activos valiosos

**Figura 2:** Estructura de lista de stock disponible en Oddo

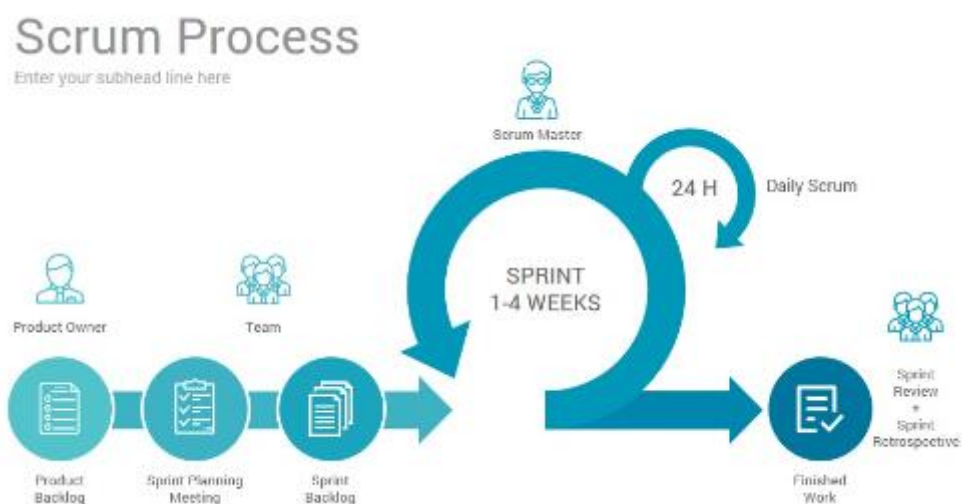
Product	Location	On Hand	Forecast	Route	Min	Max	To Order	Actions
[E-COM06] Corner Desk ...	WH/Stock	4.00	-1.00		0.00	0.00	1.00	Order Once Automate Snooze
[E-COM09] Large Desk	WH/Stock	1.00	-4.00		0.00	0.00	4.00	Order Once Automate Snooze
[FURN_9001] Flipover	WH/Stock	5.00	-6.00		0.00	0.00	6.00	Order Once Automate Snooze
[FURN_9666] Table	WH/Stock	2.00	-1.00		0.00	0.00	1.00	Order Once Automate Snooze
[FURN_7777] Office Chair	WH/Stock/Asse...	4.00	4.00	Buy	5.00	10.00	6.00	Order Once Automate Snooze
[FURN_8888] Office Lamp	WH/Stock/Asse...	8.00	8.00		10.00	10.00	2.00	Order Once
[FURN_8900] Drawer Black	WH/Stock/Asse...	12.00	12.00	Manufacture	0.00	0.00	0.00	
[FURN_9001] Flipover	WH/Stock/Asse...	5.00	5.00		0.00	0.00	0.00	Snooze
[E-COM06] Corner Desk ...	WH/Stock/Flat P...	4.00	4.00	Manufacture	0.00	0.00	0.00	Snooze
[E-COM09] Large Desk	WH/Stock/Flat P...	1.00	1.00		2.00	2.00	1.00	Order Once
[FURN_9666] Table	WH/Stock/Flat P...	2.00	2.00	Manufacture	5.00	10.00	8.00	Order Once Automate Snooze

**Nota.Odoo**

### 1.5.2. Aplicación de escritorio para gestión e inventario de la compañía JOYERÍA DIAMOND RC.

Tomaremos como referencia el proyecto previamente desarrollado para el control de inventarios de la compañía JOYERÍA DIAMOND RC., el cual fue llevado a cabo empleando la metodología ágil Scrum. Este modelo de referencia es crucial, ya que Scrum facilita una administración eficaz del proyecto, con entregas iterativas y un fuerte énfasis en la cooperación dentro del equipo. Al seguir este enfoque, intentaremos replicar las mejores prácticas observadas, como la capacidad de adaptarse rápidamente a los cambios y la entrega continua, garantizando que nuestro proceso de desarrollo sea organizado y eficiente.

**Figura 3.** Procesos Scrum. Frechina, A.

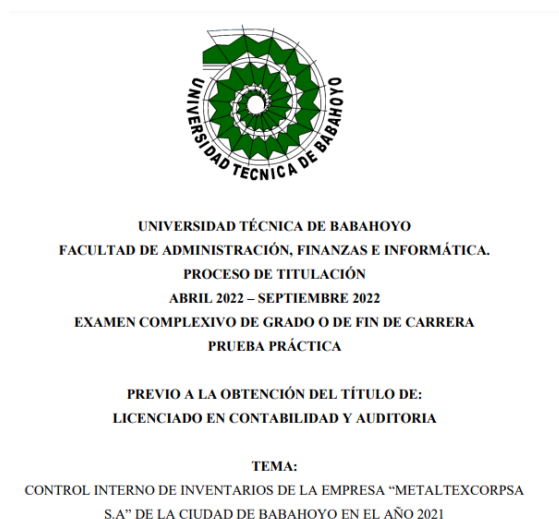


**Nota.** Frechina, A. (2018). Metodología Scrum ¿Qué es?

### 1.5.3. Control interno de inventarios de la empresa “METALTEXCORPSA S.A” de la ciudad de Babahoyo en el año 2021.

Después de revisar todo el documento, nos ha parecido muy completo y valioso para orientar el desarrollo de nuestro proyecto. No obstante, creemos que existe la oportunidad de mejorarlo aún más. Sería beneficioso considerar la implementación de un sistema de control de inventario que cuente con una interfaz intuitiva y fácil de manejar para cualquier usuario, independientemente de su nivel de experiencia técnica. Esto no solo facilita la gestión diaria de la mercancía, sino que también reduciría el margen de error y optimizaría el flujo de trabajo dentro de la empresa. Además, un sistema así permitiría a los empleados adaptarse rápidamente a su uso, incrementando la eficiencia operativa desde el primer día de su implementación. Por lo tanto, nos encargaremos de desarrollar un sistema eficiente que cumpla con estos requisitos.

#### **Figura 4. Artículo titulado: Control interno de inventarios de la Empresa “Metaltexcorspa SA” de la ciudad de Babahoyo en el año 2021**



**Nota. Repositorio de la Universidad Técnica de Babahoyo. Control interno de inventarios de la Empresa “Metaltexcorspa SA” de la ciudad de Babahoyo en el año 2021**

## 2. CAPÍTULO 2 – MARCO TEÓRICO

### 2.1. Marco teórico de programación orientada a objetos

#### 2.1.1. Metodología:

Para llevar a cabo este proyecto, empleamos una metodología ágil debido a la rapidez y flexibilidad que ofrece en su desarrollo. Esta metodología facilita la descomposición del proyecto en partes más pequeñas, que se completan y entregan en períodos breves, lo cual mejora la organización y facilita el trabajo en equipo.

Dentro de las metodologías ágiles, optamos por Scrum, que adopta un enfoque de desarrollo incremental. En Scrum, el proyecto se divide en etapas cortas llamadas sprints, lo que permite realizar ajustes rápidamente y aplicar cambios sólo en la parte afectada, en un tiempo reducido.

El ciclo de vida de Scrum se caracteriza por ser repetitivo, con etapas que se repiten para añadir nuevas funcionalidades al producto. Estas iteraciones cortas y ágiles permiten entregas parciales que se validan con el usuario, asegurando que el producto cumpla con los requisitos deseados en cada fase del desarrollo.

Las fases que conforman son:

#### **a) Inicio:**

En la fase de inicio, se lleva a cabo el estudio y análisis del proyecto para identificar las necesidades básicas del sprint. En las metodologías ágiles, un sprint es un mini-proyecto de hasta un mes que se conecta con otros mini-proyectos para alcanzar los objetivos generales y específicos del proyecto principal.

Durante esta fase, se abordan preguntas clave como:

- ¿Qué quiero?
- ¿Cómo lo quiero?
- ¿Cuándo lo quiero?

Además, Scrum prioriza la formación de equipos pequeños, de entre 3 y 5 personas, para facilitar la fluidez de las ideas y fomentar la creatividad.

Entre los primeros pasos de Scrum se incluyen los siguientes procesos:

1. Crear la visión del proyecto.
2. Identificar a los Scrum Masters y a los stakeholders.
3. Formar equipos Scrum.
4. Desarrollar épicas.
5. Crear backlogs o listas de requisitos, priorizando el producto.
6. Planificar el lanzamiento.

En esta fase inicial, se realizan reuniones con la empresa para definir los requisitos más importantes. También se ajustan los tiempos de disponibilidad del administrador y se abordan los problemas que impiden que la joyería alcance los resultados esperados.

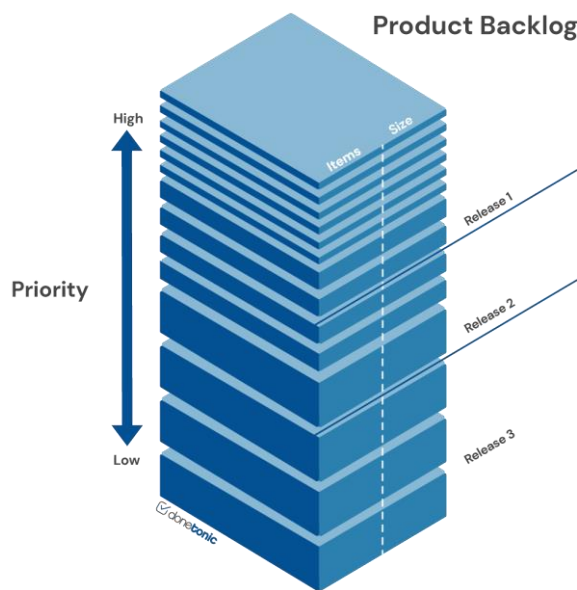
## **b) Planificación y estimación:**

En la fase de Planificación y Estimación, el equipo descompone los ítems del Product Backlog en tareas específicas y alcanzables. Esta fase es crucial para asegurar que cada tarea sea clara y factible dentro del tiempo asignado al sprint. Se estiman los tiempos y recursos necesarios para cada tarea, facilitando la gestión de expectativas y la asignación eficiente de recursos.

El equipo establece el Sprint Goal, un objetivo concreto para el sprint, como "Desarrollar e implementar el módulo para analizar las tendencias

de los clientes". La planificación abarca la asignación de responsabilidades a los miembros del equipo de acuerdo con sus habilidades y capacidades, promoviendo la colaboración y garantizando que todos comprendan sus roles durante el sprint. Esta fase configura un plan detallado que orientará el trabajo del equipo durante el sprint, asegurando que se enfoquen en los aspectos más críticos y mantengan el rumbo hacia los objetivos establecidos.

**Figura 4.** Product Backlog.



### **C) Implementación:**

En la fase de Implementación, el equipo se enfoca en llevar a cabo las tareas planificadas en el Sprint Backlog, orientadas a mejorar la acumulación de productos menos demandados mediante el desarrollo de un sistema de control de inventario. Esta etapa es fundamental para desarrollar e implementar efectivamente funcionalidades clave como los módulos para el análisis de datos de ventas y tendencias, el sistema para evaluar las necesidades del mercado, y el mecanismo para seguir la rotación de productos. Estos desarrollos están diseñados para cumplir con los objetivos específicos del proyecto, que incluyen la

implementación de un sistema de análisis de tendencias y preferencias de clientes para realizar previsiones de demanda más precisas, evaluar las necesidades reales del mercado antes de realizar nuevas adquisiciones, y crear un seguimiento detallado de la rotación de productos para identificar rápidamente los de alta y baja rotación.

Durante esta fase, se realizan entregas parciales y se revisa continuamente el progreso para asegurar que se está alcanzando el Sprint Goal. El enfoque iterativo de la implementación permite al equipo ajustar el trabajo basado en los resultados obtenidos y el feedback recibido. Este método asegura que cualquier problema o desviación se aborde de manera oportuna, manteniendo el desarrollo del sistema alineado con los objetivos del proyecto y garantizando una gestión de inventario más eficiente y ajustada a las necesidades reales del mercado.

#### **d) Revisión y Retrospectiva:**

Al final de cada sprint, en la fase de Revisión y Retrospectiva, se realiza una Sprint Review para presentar a los stakeholders el trabajo completado, incluyendo los desarrollos del sistema de control de inventario. Durante esta revisión, se demuestra el progreso en funcionalidades clave como el análisis de tendencias de clientes y el seguimiento de productos, y se obtiene retroalimentación para asegurar que las soluciones cumplen con los requisitos del cliente. La revisión también permite verificar que el sistema está alineado con los objetivos específicos del proyecto, como mejorar la previsión de demanda, evaluar las necesidades del mercado y gestionar la rotación de productos.

Posteriormente, se lleva a cabo una Sprint Retrospective, donde el equipo reflexiona sobre el sprint. Se evalúa qué aspectos del trabajo fueron exitosos, cuáles presentaron problemas y qué se puede mejorar en futuros sprints. Esta fase es crucial para la mejora continua, ya que permite al equipo ajustar el proceso, optimizar la eficiencia y resolver cualquier inconveniente surgido durante el sprint. Al incorporar los



aprendizajes de la retrospectiva, el equipo puede refinar el sistema y sus procesos, garantizando una mejor adaptación a las necesidades del mercado y una gestión más eficaz del inventario en los siguientes ciclos de desarrollo.

**Figura 5.** Stakeholder.



#### **e) Lanzamiento:**

Finalmente, en la fase de lanzamiento, se entrega el producto completo o las partes funcionales desarrolladas al cliente o usuario final. Esta fase es crucial para asegurar que el sistema de control de inventario, diseñado para mejorar la acumulación de productos menos demandados, esté completamente funcional y cumpla con los requisitos y expectativas establecidas. El sistema incluye funcionalidades clave como el análisis de tendencias y preferencias de clientes, la evaluación de necesidades del mercado, y el seguimiento de la rotación de productos.

Se realiza una revisión final para confirmar que todas estas funcionalidades están operativas y efectivas. Cualquier ajuste necesario se implementa para garantizar el buen funcionamiento del sistema, asegurando que la herramienta proporcionada optimice la gestión de inventario, alinee las compras con la demanda real y reduzca la

acumulación de productos no deseados. El lanzamiento marca el cierre del ciclo de desarrollo y ofrece al cliente una solución que no solo mejora la eficiencia en la gestión de inventario, sino que también alinea las adquisiciones con las necesidades reales del mercado, cumpliendo así los objetivos específicos del proyecto.

Tabla de comparación:

Aspecto	Scrum	Kanban	XP(Extreme Programming)	Lean
Ciclo de vida	Sprints de 2 a 4 semanas con entregas funcionales.	Flujo continuo sin sprints.	Iteraciones breves de 1 a 2 semanas.	Enfoque en minimizar desperdicios.
Enfoque de roles	Roles específicos: Product Owner, Scrum Master, equipo.	Sin roles formales; equipos autodirigidos.	Equipo cohesivo con roles definidos.	Sin roles formales; se adapta a necesidades.
Adaptabilidad	Muy adaptable, cambios al final de cada sprint.	Flexible, permite cambios en cualquier momento.	Flexible, enfocado en prácticas continuas.	Fomenta flexibilidad y optimización.
Planificación	Sprints con objetivos específicos.	Sin planificación estructurada; prioriza urgencias.	Similar a Scrum, enfoque en pruebas.	Enfoque en optimización y maximizar valor.
Motivo de elección para el proyecto	Claridad, adaptabilidad y entrega continua.	No fue elegido por falta de estructura.	Valioso por la calidad del software.	No elegido por optimización insuficiente.

**Tabla 8:** Cuadro comparativo

### 2.1.2. Modelo:

#### **Modelo Entidad-Relación**

Un diagrama entidad-relación (ERD) es un tipo de diagrama que muestra cómo las entidades (personas, objetos o conceptos) se relacionan entre sí dentro de un sistema. Estos diagramas se usan principalmente para diseñar o analizar bases de datos relacionales en áreas como ingeniería de software, sistemas de información y educación. Los ERD emplean símbolos como rectángulos, diamantes, óvalos y líneas para representar entidades, relaciones y atributos. Además, reflejan la estructura gramatical al usar entidades como sustantivos y relaciones como verbos. Los diagramas ER están relacionados con los diagramas de estructura de datos (DSD), que se enfocan en las relaciones dentro de las entidades, y a menudo se combinan con diagramas de flujo de datos (DFD), los cuales muestran cómo fluye la información en sistemas o procesos.

Componentes de un diagrama ERD:

Los diagramas ER se componen de entidades, relaciones y atributos. También representan la cardinalidad, que define las relaciones en términos de números.

1. **Entidad:** Se refiere a algo que puede ser identificado, como una persona, objeto, concepto o evento, y que tiene datos asociados. Las entidades son representadas como sustantivos y suelen mostrarse en rectángulos.

- **Tipo de entidad:** Es un grupo de cosas definibles, como "estudiantes" o "autos", mientras que la entidad es un ejemplo específico de ese grupo.

- **Conjunto de entidades:** Define un grupo en un momento dado, como "estudiantes inscritos el primer día". Un ejemplo sería los autos registrados en una región.
- **Categorías de entidades:** Pueden ser fuertes (definidas por sus propios atributos), débiles (no se pueden definir por sí solas) o asociativas (relacionan diferentes entidades).
- **Claves de entidad:** Son atributos que identifican de forma única a una entidad. Se dividen en:
  - **Súper clave:** Conjunto de atributos que definen una entidad.
  - **Clave candidata:** Súper clave mínima, con el menor número de atributos.
  - **Clave primaria:** Clave candidata seleccionada para identificar la entidad.
  - **Clave extranjera:** Conecta una entidad con otra, estableciendo relaciones.

2. **Relación:** Describe cómo las entidades se asocian entre sí, representando una interacción entre ellas, como "un estudiante se inscribe en un curso". Se muestran como diamantes o etiquetas en las líneas de conexión.

- **Relación recursiva:** Ocurre cuando una entidad se relaciona consigo misma.

3. **Atributo:** Característica de una entidad, mostrada como un óvalo.

- **Atributo descriptivo:** Propiedad de una relación.
- **Categorías de atributos:** Pueden ser simples (no divisibles), compuestos (pueden desglosarse), derivados (calculados a partir de otros), de valor único (solo un valor) o de valores múltiples (varios valores).

### 2.1.3. Patrones GUI:

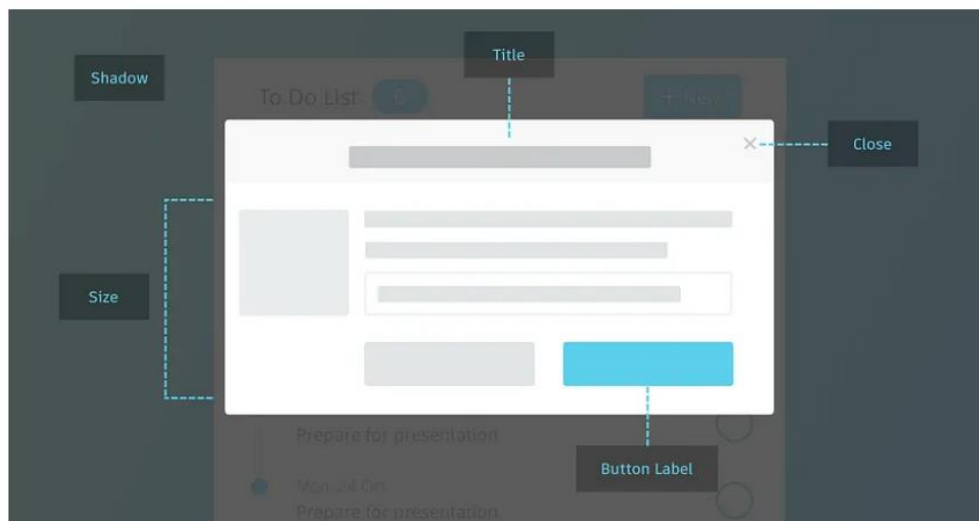
Los patrones de diseño de la interfaz de usuario son fundamentales para crear interfaces que sean robustas y fáciles de usar. Actúan como guías para diseñar experiencias de usuario que sean intuitivas, consistentes y eficientes. Estos patrones no solo se centran en el aspecto visual, sino que también abordan problemas comunes de diseño para mejorar la usabilidad y accesibilidad. Al utilizar patrones probados, los diseñadores pueden desarrollar interfaces que resulten familiares para los usuarios, lo que facilita el aprendizaje y aumenta la satisfacción.

- Pantallas modales:

Son un patrón de diseño que se usa para posicionar generalmente mensajes o alertas sobre la pantalla principal de un sitio. Su visualización, crea un estado donde se desactiva la pantalla principal hasta que el usuario interactúe y responda al modal.

Las pantallas modales permiten enfocar la atención del usuario en información o acciones específicas, interrumpir temporalmente la interacción con la pantalla principal para completar una acción o responder a un mensaje importante, facilitar la comunicación mostrando mensajes, alertas o formularios directamente en la pantalla principal, reforzar la experiencia del usuario al presentar opciones y datos contextuales sin necesidad de navegar fuera de la página actual, y mantener el contexto para que el usuario pueda continuar con la tarea en la pantalla principal después de cerrar el modal, sin perder el progreso.

**Figura 6.** Pantalla Modales

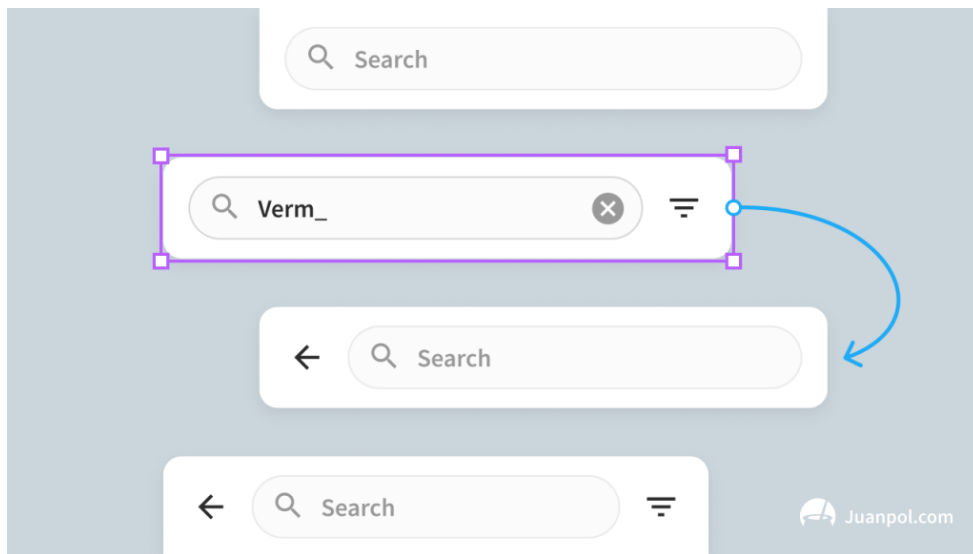


**Nota. Patrón GUI – Modales**

- Barra de navegación:

La barra de navegación es un patrón de diseño de la interfaz de usuario que facilita el acceso a las principales secciones de una aplicación o sitio web. Generalmente ubicada en la parte superior o lateral de la interfaz, debe ser clara y coherente, permitiendo a los usuarios encontrar rápidamente lo que buscan. Su diseño debe ser consistente en toda la aplicación para reducir la curva de aprendizaje y mejorar la experiencia del usuario. Además, debe adaptarse a diferentes tamaños de pantalla y dispositivos, y destacarse visualmente para ser fácilmente identificable. La interactividad, como cambios de color o animaciones, también es clave para proporcionar retroalimentación cuando los usuarios interactúan con la barra.

**Figura 7.** Barra de navegación

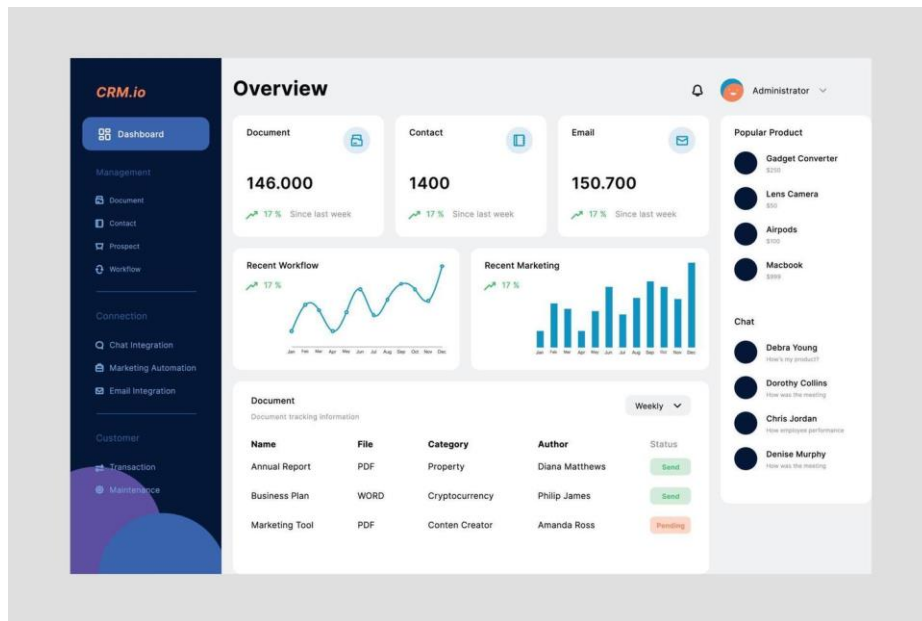


**Nota. Patrón GUI – Barras de búsqueda**

- Dashboard:

Es un panel de control que ofrece una vista consolidada de información clave y datos relevantes en una sola pantalla. Utiliza gráficos, tablas y otros elementos visuales para representar datos de manera clara y comprensible, permitiendo a los usuarios monitorear y analizar información de forma rápida. Incluye elementos interactivos como filtros y menús para personalizar y explorar los datos, y presenta la información más importante de manera prominente. Además, permite la personalización del contenido y debe ser accesible y fácil de usar, con actualizaciones en tiempo real para proporcionar la información más reciente.

**Figura 8. Dashboard**



### **Nota. Patrón GUI - Dashboard**

- **Captcha:**

Es un patrón de diseño de interfaz de usuario utilizado para garantizar que una interacción en línea sea realizada por un ser humano y no por un bot. Se integra en formularios y otras áreas críticas para prevenir el acceso automatizado no deseado. El captcha requiere que los usuarios completen tareas que son fáciles para los humanos, pero difíciles para los algoritmos, como identificar texto distorsionado o seleccionar imágenes específicas. Debe ser accesible y fácil de usar, con alternativas para personas con discapacidades. Su objetivo es proteger la seguridad de los sistemas en línea al distinguir entre usuarios reales y automatizados.



**Figura 9. Captcha**

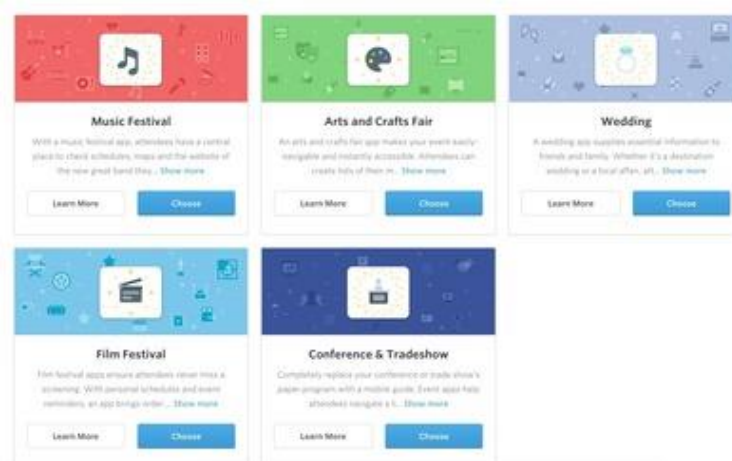


**Nota. Patrón GUI – Captcha**

- **Cards**

Una card es un componente con un patrón de diseño de interfaz de usuario (UI) que agrupa información relacionada en un contenedor de tamaño variable que se asemeja visualmente a una tarjeta. Las Cards se crearon como un patrón de interfaz modular que funciona bien una amplia variedad de pantallas y resoluciones. Su objetivo principal es la de mostrar la información suficiente como para conseguir que el usuario la toque o la haga clic para ver más detalles y seguir consumiendo contenido.

**Figura 10. Cards**



## **Nota. Patrón GUI - Cards**

### 2.1.4. Patrones de desarrollo:

#### **MVC:**

El patrón Modelo-Vista-Controlador (MVC) es una forma de organizar el código en aplicaciones de software que separa la lógica de negocio, la interfaz de usuario y el control de las interacciones del usuario. Esto permite un diseño más limpio, mantenible y escalable.

- **Modelo (Model):**

Es responsable de gestionar los datos y las reglas de negocio. Se encarga de obtener, almacenar y procesar la información, por ejemplo, la gestión del inventario en una base de datos. El modelo no sabe nada sobre la interfaz de usuario ni cómo se presentan los datos.

- **Vista (View):**

Representa la interfaz gráfica o la forma en la que los datos se presentan al usuario. Es lo que el usuario ve y con lo que interactúa. La vista solo muestra los datos que recibe del modelo y no contiene lógica de negocio.

- **Controlador (Controller):**

Actúa como intermediario entre el modelo y la vista. Responde a las acciones del usuario (como hacer clic en un botón o enviar un formulario) y se encarga de actualizar el modelo o la vista.

#### **Funcionamiento:**

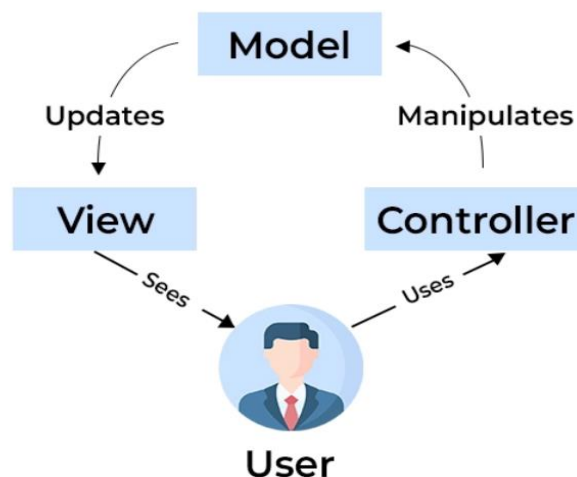
El usuario interactúa con la vista (por ejemplo, selecciona un producto).

- Esa interacción es capturada por el controlador, que decide qué hacer (consultar o actualizar datos).
- El controlador pide información al modelo o lo actualiza.
- El modelo responde al controlador con los datos solicitados.
- El controlador actualiza la vista para reflejar los cambios y mostrar los datos al usuario.

Ventajas del MVC:

- El diseño del código es más limpio y organizado.
- La lógica de negocio está separada de la interfaz gráfica, lo que permite modificar la interfaz sin tocar el núcleo del sistema.
- Cambios en una parte (modelo, vista o controlador) no afectan a las otras.
- Este patrón es muy usado en el desarrollo de aplicaciones web y de escritorio porque permite gestionar de manera eficiente el flujo de datos y las interacciones del usuario.

**Figura 11. MVC.**



## **DAO:**

El Data Access Object es un patrón de diseño estructural que proporciona una abstracción a la lógica de acceso a la base de datos, separando la lógica de negocios de la lógica de persistencia. Este patrón se encarga de gestionar todas las operaciones relacionadas con la base de datos, como crear, leer, actualizar y eliminar datos también conocido CRUD.

El propósito principal del DAO es proporcionar una interfaz uniforme para acceder a los datos sin exponer los detalles internos de la base de datos, lo cual permite que los desarrolladores trabajen con objetos del negocio sin preocuparse por cómo se gestionan en la base de datos.

- **Ventajas del Patrón DAO:**
  - Separación de responsabilidades: Permite mantener el código más limpio y modular, facilitando la separación de la lógica de negocio de la lógica de acceso a datos.
  - Reutilización: Los métodos para acceder a la base de datos pueden ser reutilizados en diferentes partes de la aplicación.
  - Mantenibilidad: Cambios en la base de datos, no requieren cambios en la lógica de negocios, solo en la implementación del DAO. se gestionan en la base de datos.
- **Ejemplo:**

### **Problemática**

Una de las grandes problemáticas al momento de acceder a los datos, es que la implementación y formato de la información puede variar según la fuente de los datos, además, implementar la lógica de acceso a datos en la capa de lógica de negocio puede ser un gran problema, pues tendríamos que lidiar con la lógica de negocio en sí, más la implementación para acceder a los datos, adicional, si tenemos múltiples fuentes de datos o estas pueden variar, tendríamos que implementar las diferentes lógicas para acceder las

diferentes fuentes de datos, como podrían ser: bases de datos relacionales, No SQL, XML, archivos planos, servicios web o REST, etc).

Un programador inexperto o con pocos conocimientos sobre arquitectura podría optar por crear en un solo método la lógica de negocio y la lógica de acceso a datos, creando un método muy difícil de entender y mantener.

### **Solución**

El patrón Data Access Object (DAO) propone separar por completo la lógica de negocio de la lógica para acceder a los datos, de esta forma, el DAO proporcionará los métodos necesarios para insertar, actualizar, borrar y consultar la información; por otra parte, la capa de negocio solo se preocupa por lógica de negocio y utiliza el DAO para interactuar con la fuente de datos.

#### **2.1.5. Lenguajes de programación o frameworks:**

- **JavaScript:**

Es un lenguaje de programación ligero y versátil que se utiliza principalmente para añadir interactividad y dinamismo a las páginas web. Sus programas, conocidos como scripts, se integran directamente en el código HTML y se ejecutan en el navegador web. Estos scripts permiten que elementos en la página respondan a eventos, como cambiar el aspecto de un botón al pasar el ratón sobre él o abrir una ventana nueva al hacer clic en un enlace.

Desarrollado por Netscape, JavaScript tiene sus raíces en el lenguaje Java, aunque con diferencias clave. Mientras que Java es un lenguaje de programación completo que permite crear aplicaciones independientes que pueden ejecutarse fuera del

navegador, JavaScript está diseñado específicamente para funcionar dentro de las páginas HTML. Aunque Java también se puede utilizar en páginas web mediante applets, JavaScript se distingue por su integración directa y su enfoque en la interacción en el entorno web.

Ventajas:

- JavaScript permite añadir interactividad a las páginas web sin necesidad de recargar la página completa.
- Los scripts de JavaScript se ejecutan directamente en el navegador del usuario, lo que reduce la necesidad de procesamiento en el servidor.
- JavaScript se integra de manera fluida con HTML y CSS, permitiendo manipular la estructura y el estilo de las páginas web de manera dinámica.
- Se puede utilizar en el servidor a través de Node.js, lo que permite desarrollar aplicaciones completas con un solo lenguaje de programación.

**Figura 12.** JavaScript.

```
<HTML>
<HEAD>
<TITLE>Ejemplo01.htm</TITLE>

<SCRIPT LANGUAGE="JavaScript">
  //Visualizar un mensaje de bienvenida
  alert("¡Bienvenido a nuestra página!");
</SCRIPT>

</HEAD>
<BODY>
<a href='Ejemplo02.html'>Ir al siguiente ejemplo...</a>
</BODY>
</HTML>
```

**Nota.** Ejemplo simple aplicando javascript

- HTML 5:

Es la quinta versión del lenguaje de marcado HTML (HyperText Markup Language), se utiliza para estructurar y presentar contenido en la web. Introdujo nuevas etiquetas, capacidades multimedia, y APIs que permiten una mayor interacción con los usuarios sin depender de plugins externos.

- Características principales de HTML5:

A. Nuevas etiquetas semánticas: Mejoran la estructura y semántica del código, facilitando la comprensión tanto para los navegadores como para los desarrolladores:

- `<header>`: Representa el encabezado de un documento o sección.
- `<footer>`: Define el pie de página.
- `<article>`: Define contenido independiente, como una entrada de blog.
- `<section>`: Representa una sección genérica.
- `<aside>`: Contenido relacionado pero separado, como barras laterales.

B. Soporte multimedia nativo: Permite la inserción de multimedia directamente en las páginas web sin necesidad de plugins:

- `<video>`: Para incrustar videos.
- `<audio>`: Para insertar contenido de audio.

C. Capacidades gráficas:

- `<canvas>`: Un elemento que se puede usar para crear gráficos interactivos, juegos o incluso animaciones.
- SVG (Scalable Vector Graphics): Soporte nativo para gráficos vectoriales escalables.

D. Formularios mejorados: Mejoras en la creación de formularios, con nuevos tipos de input que simplifican la validación de datos:

- Nuevos tipos de campos de entrada como `<input type="email">`, `<input type="date">`, `<input type="number">`, y más.
- Soporte para atributos como `required`, `placeholder`, y `autofocus`, que mejoran la usabilidad y experiencia del usuario.

E. APIs de HTML5: una serie de APIs (Interfaz de Programación de Aplicaciones) permiten a los desarrolladores crear aplicaciones web más interactivas y avanzadas:

- Geolocation API: Permite acceder a la ubicación geográfica del usuario.
- Local Storage: Ofrece almacenamiento local en el navegador sin necesidad de cookies, permitiendo que las aplicaciones web almacenen datos de forma persistente.
- Web Workers: Permite ejecutar scripts en segundo plano sin bloquear la interfaz de usuario.
- WebSockets: Proporciona una conexión bidireccional en tiempo real entre el servidor y el navegador, ideal para aplicaciones en tiempo real.

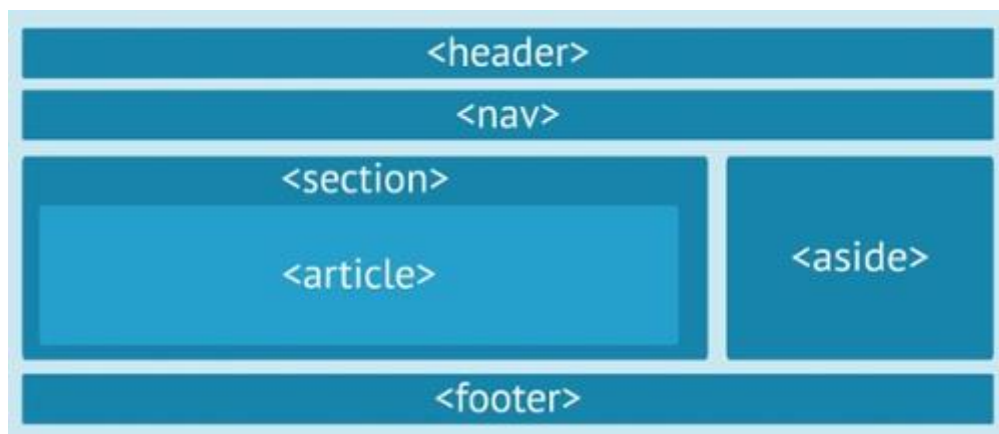
F. Compatibilidad con dispositivos móviles: Las aplicaciones web responsivas pueden ajustar el diseño automáticamente según el tamaño de la pantalla.



### Ventajas de HTML5:

- Las nuevas etiquetas semánticas facilitan que los motores de búsqueda y las tecnologías de asistencia comprendan el contenido de una página.
- Facilita la creación de aplicaciones web interactivas y avanzadas sin depender de tecnologías adicionales.
- Compatibilidad con dispositivos móviles
- Mejora el rendimiento al reducir la necesidad de plugins y tecnologías adicionales.

Figura 13. HTML5



### Nota. Estructura básica de una página con HTML

- **Node.js:**

Es un entorno de ejecución de JavaScript basado en el motor V8 de Google Chrome, que permite ejecutar código JavaScript en el servidor. Es decir, permite a los desarrolladores utilizar un solo lenguaje para ambas partes de una aplicación web: frontend y backend.

### **Características principales:**

- Utiliza un modelo de Entrada/Salida, puede manejar múltiples operaciones de manera eficiente sin esperar a que cada una se complete
- Usa un sistema de bucle de eventos para manejar múltiples solicitudes concurrentes. Esto lo hace más eficiente en términos de recursos en comparación con otros servidores web que usan hilos múltiples.
- El procesamiento de solicitudes es extremadamente rápido, lo convierte en una excelente opción para aplicaciones que requieren una alta capacidad de respuesta.
- Esto permite a los desarrolladores integrar rápidamente módulos de terceros para gestionar tareas comunes como conexiones a bases de datos, autenticación, gestión de archivos, entre otros.
- Compatible con múltiples sistemas operativos, incluidos Linux, Windows y macOS, lo que permite a los desarrolladores ejecutar sus aplicaciones en cualquier entorno.

### **Funcionamiento:**

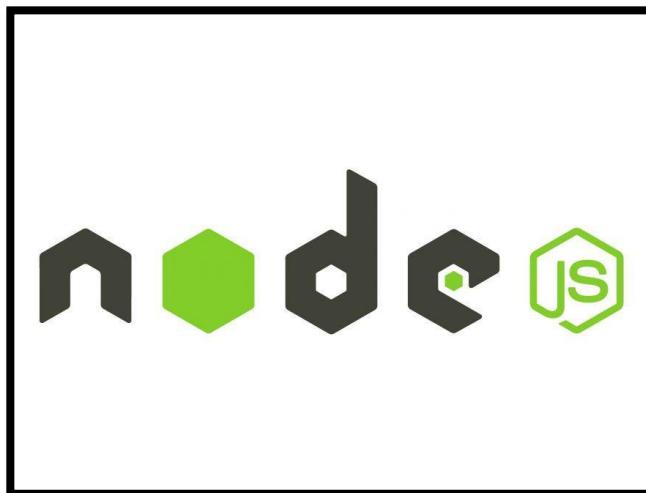
- Servidor HTTP: Puede utilizarse para crear servidores web sin necesidad de un software adicional. Puedes manejar las solicitudes HTTP directamente.

- Asincronía y Callback: Diseñado para manejar tareas de manera asincrónica. Esto significa que, en lugar de esperar a que una operación termine sigue ejecutando otras tareas.

#### **Ventajas de Node.js:**

- Debido a su naturaleza asincrónica y no bloqueante, Node.js puede manejar grandes volúmenes de conexiones concurrentes.
- El motor V8 de JavaScript optimiza el rendimiento, lo que hace que Node.js sea rápido.
- El ecosistema de npm proporciona acceso a miles de bibliotecas que facilitan el desarrollo rápido de aplicaciones.
- Permite a los desarrolladores utilizar JavaScript tanto en el cliente como en el servidor.

**Figura 14.** NodeJS



**Nota.** Logo de nodejs

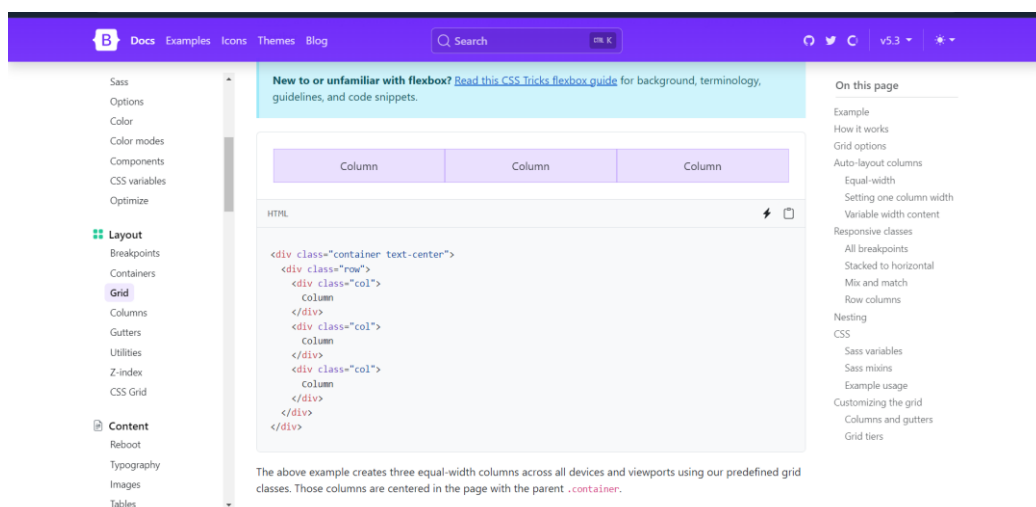
- **Bootstrap 5:**

Es un framework que permite crear sitios web y aplicaciones web responsivas y modernas de manera rápida y eficiente. Incluye una colección de herramientas de diseño como componentes de interfaz de usuario, sistemas de rejilla, y estilos CSS predefinidos, lo que facilita el desarrollo web.

### **Ventajas de Bootstrap 5:**

- Con clases y componentes predefinidos, se puede construir sitios y aplicaciones rápidamente sin preocuparte por escribir CSS desde cero.
- Facilita la creación de sitios web que se adapten automáticamente a cualquier dispositivo.
- Bootstrap tiene una enorme comunidad de desarrolladores, lo que significa que hay mucha documentación, tutoriales y soluciones disponibles.
- Funciona bien con los navegadores modernos como Chrome, Firefox, Safari, y Edge.

**Figura 15.** Bootstrap 5



**Nota. Estructura de cómo se puede implementar un grid con Bootstrap**

- **React:**

Es una biblioteca de JavaScript para desarrollar interfaces de usuario (UI) interactivas y dinámicas. Se enfoca principalmente en la vista dentro del patrón de diseño Modelo-Vista-Controlador (MVC), lo que permite a los desarrolladores crear componentes reutilizables y manejar de manera eficiente la interacción del usuario y el estado de la interfaz.

**Ventajas de React:**

- Al dividir la UI en componentes reutilizables, facilita el desarrollo y mantenimiento de aplicaciones complejas.
- Permite un flujo de trabajo más rápido y ágil debido a su naturaleza declarativa y su ecosistema.
- El uso del Virtual DOM y su enfoque basado en componentes lo hace extremadamente rápido y eficiente.
- Mejora la legibilidad del código, permitiendo que la lógica de la interfaz y el diseño estén en el mismo lugar.
- Tiene muchos recursos, bibliotecas y herramientas disponibles.

**Figura 16.** React



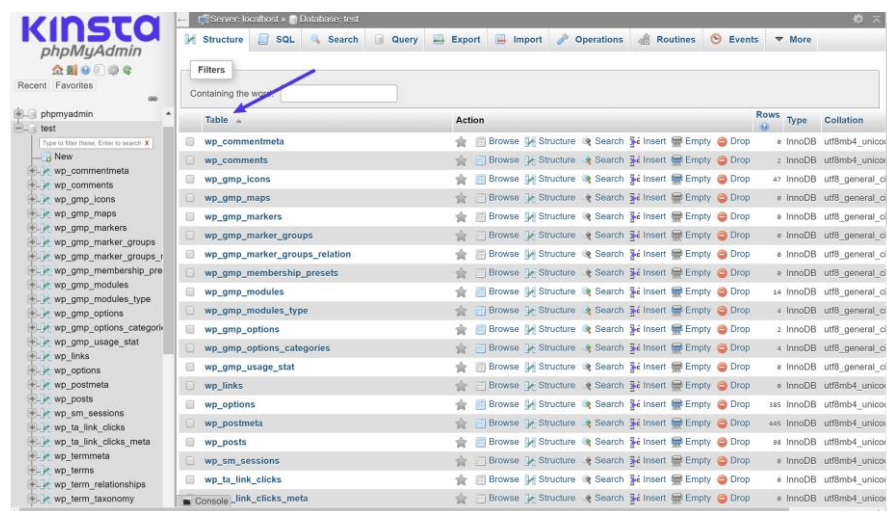
**Nota.** Logo de React

### 2.1.6. Gestor de base de datos:

#### MySQL:

MySQL es un sistema de gestión de bases de datos relacional (RDBMS) de código abierto, desarrollado originalmente por MySQL AB y actualmente propiedad de Oracle Corporation. Es uno de los sistemas de bases de datos más populares del mundo, especialmente para aplicaciones web.

Figura 17. MySQL



#### **Nota. Diseño de base datos con MySQL**

Características principales:

- **Relacional:** Utiliza tablas para almacenar datos y establecer relaciones entre ellas.
- **Open Source:** Disponible gratuitamente bajo la licencia GPL, aunque también existe una versión comercial.
- **Multiplataforma:** Funciona en diversos sistemas operativos como Windows, Linux, macOS, etc.
- **Escalabilidad:** Puede manejar bases de datos desde pequeñas hasta muy grandes (terabytes de información).

- **Velocidad:** Ofrece un alto rendimiento en la ejecución de consultas.
- **Seguridad:** Proporciona un sistema sólido de privilegios y contraseñas.
- **Compatibilidad:** Cumple con el estándar SQL (Structured Query Language).

Operaciones CRUD:

- CREATE: Insertar nuevos datos
- READ: Consultar datos existentes
- UPDATE: Modificar datos existentes
- DELETE: Eliminar datos

Lenguaje SQL en MySQL:

MySQL utiliza SQL para realizar operaciones.

Algunos comandos:

SELECT: Para consultar datos

INSERT: Para insertar nuevos registros

UPDATE: Para modificar registros existentes

DELETE: Para eliminar registros

CREATE TABLE: Para crear nuevas tablas

ALTER TABLE: Para modificar la estructura de una tabla

DROP TABLE: Para eliminar una tabla

Herramientas:

1. **MySQL Workbench:** Herramienta gráfica oficial para diseño y administración de bases de datos.
2. **phpMyAdmin:** Herramienta web popular para administrar bases de datos MySQL.
3. **Command Line Client:** Interfaz de línea de comandos para interactuar con MySQL.

Integración con Lenguajes de programación:

MySQL se puede integrar fácilmente con varios lenguajes de programación como PHP, Python, Java, Node.js, etc., a través de conectores específicos.

Ventajas:

- Alta velocidad en la ejecución de consultas
- Bajo costo en requerimientos para elaborar bases de datos
- Facilidad de configuración e instalación
- Soporte para múltiples sistemas operativos
- Baja probabilidad de corromper datos

Desventajas:

- Para grandes volúmenes de datos, puede ser menos eficiente que otros RDBMS
- Algunas características avanzadas no están disponibles en la versión gratuita



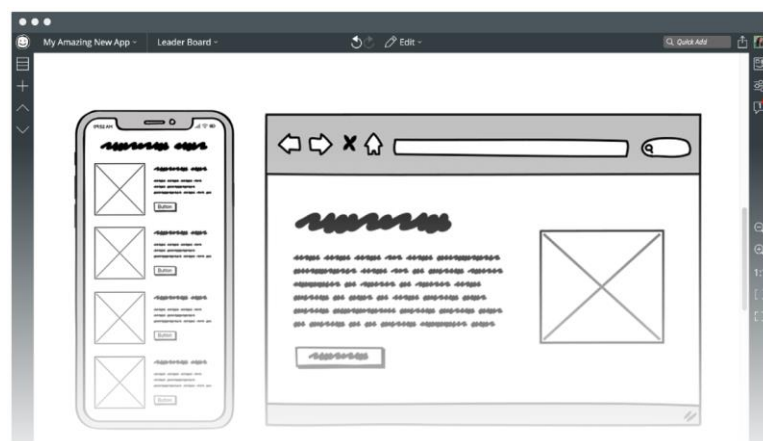
### 2.1.7. Herramientas adicionales:

- **Balsamiq:**

Balsamiq es una plataforma en línea centrada en el diseño de interfaces de usuario, conocida por su simplicidad y enfoque en la creación rápida de prototipos. A través de Balsamiq, es posible desarrollar wireframes para sitios web y aplicaciones, proporcionando una representación visual básica de la estructura de una interfaz sin entrar en detalles de diseño final. Esto facilita la planificación y validación temprana de ideas antes de pasar a etapas más avanzadas del proceso de diseño.

Una de las características más valoradas de Balsamiq es su enfoque en la creación rápida y eficiente de maquetas. Su interfaz intuitiva permite a diseñadores, desarrolladores y partes interesadas colaborar fácilmente, promoviendo una comunicación clara y asegurando que todas las partes involucradas puedan contribuir al proceso de creación. Además, Balsamiq fomenta la colaboración permitiendo la retroalimentación y la revisión en tiempo real. Con un catálogo de componentes predefinidos y la capacidad de arrastrar y soltar elementos, la plataforma es ideal para la creación de prototipos de baja fidelidad, lo que la convierte en una herramienta esencial para equipos de desarrollo que buscan validar ideas rápidamente y con pocos recursos.

**Figura 18.** Balsamiq



- **Bizagi:**

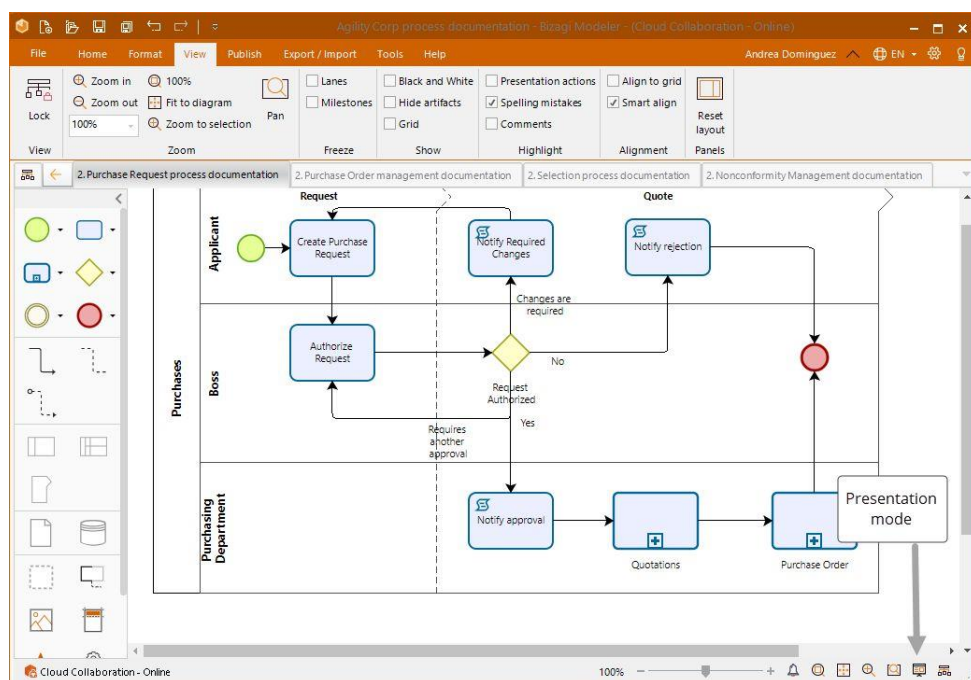
Es una plataforma de automatización de procesos empresariales de bajo código, diseñada para mejorar la colaboración entre personas, sistemas, robots e información mediante la automatización inteligente de procesos de negocio. Bizagi facilita la gestión de flujos de trabajo y tareas recurrentes dentro de las organizaciones, permitiendo una mayor eficiencia operativa. La plataforma se caracteriza por su enfoque preconfigurado, el cual soporta la automatización de procesos como la incorporación de empleados, solicitudes de licencia, cuentas por pagar y contratación.

Bizagi está compuesto por tres productos clave: Bizagi Modeler, Bizagi Studio y Bizagi Engine, que funcionan de manera complementaria:

- Bizagi Modeler: es una herramienta que permite modelar, documentar y analizar procesos empresariales utilizando la notación estándar BPMN (Business Process Model and Notation). Esto ayuda a las organizaciones a crear mapas visuales y detallados de sus procesos.
- Bizagi Studio: permite a las organizaciones diseñar y automatizar procesos mediante un entorno de desarrollo de bajo código, facilitando la creación de aplicaciones de procesos sin necesidad de contar con una amplia experiencia en programación.
- Bizagi Engine: es el motor que ejecuta los procesos diseñados, disponible tanto en entornos locales como en la nube. Esto permite a las organizaciones ejecutar y monitorizar sus procesos desde cualquier lugar, en cualquier momento.

La plataforma Bizagi ayuda a las empresas a automatizar procesos críticos, mejorar la asignación de tareas y optimizar la comunicación entre equipos. Además, permite integraciones con otros sistemas empresariales, lo que la convierte en una solución flexible y escalable. También incluye herramientas de análisis que facilitan la toma de decisiones basada en datos, contribuyendo a la eficiencia operativa y la mejora continua.

**Figura 19.** Bizagi



- **draw.io:**

Es una herramienta de diagramación en línea que permite a los usuarios crear una amplia variedad de diagramas, como diagramas de flujo, mapas mentales, diagramas de red, diagramas UML (Unified Modeling Language), organigramas y otros tipos de representaciones visuales. Es ampliamente utilizada tanto en entornos educativos como empresariales para planificar, visualizar, y comunicar ideas de manera clara y estructurada. draw.io es conocido por ser una solución fácil de usar y accesible, ya que está disponible como una aplicación web

gratuita y como una extensión para plataformas de almacenamiento en la nube, como Google Drive, OneDrive y GitHub.

- Características:

Una de las principales características de draw.io es su compatibilidad con una amplia gama de navegadores, ya que es una aplicación web basada en el lenguaje de programación JavaScript y se distribuye bajo la licencia Apache 2.0. Su facilidad de uso se debe en parte a su sistema de arrastrar y soltar ("drag and drop"), que permite a los usuarios mover formas, imágenes, flechas y otros elementos visuales de manera sencilla y rápida. Además, ofrece la posibilidad de crear colecciones de diagramas y personalizar imágenes para usarlas en los diseños.

La arquitectura de draw.io admite la creación de diagramas profesionales, administrativos y académicos, y su disponibilidad en línea permite a los usuarios almacenar su trabajo en servicios en la nube como Google Drive, OneDrive o en almacenamiento local. También ofrece una versión offline que puede descargarse para trabajar sin conexión en sistemas operativos como Windows, Mac OS, Linux, Chrome OS, entre otros.

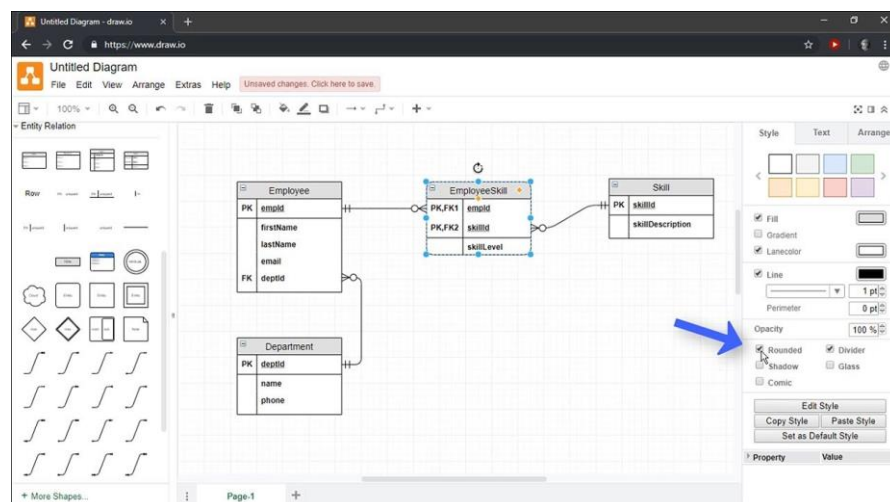
La interfaz de draw.io es simple y bien organizada, lo que facilita su uso. Gracias a su diseño minimalista y a su rápida navegación, los usuarios pueden trabajar de forma fluida sin complicaciones visuales.

- Funcionalidades:

Permite crear una amplia variedad de diagramas y cuenta con una función de diseño automático que ayuda a crear bibliotecas personalizadas de elementos. La herramienta también ofrece acceso a una gran selección de bibliotecas que incluyen cientos de formas y elementos visuales. Además, facilita la inclusión de metadatos en los diagramas, lo que permite agregar información adicional relevante.

Una de las características más innovadoras de draw.io es la capacidad de integrar los diagramas en puntos de intersección o "confluencia", y también incluye una herramienta de búsqueda avanzada que facilita localizar diagramas específicos dentro del sistema. Los diagramas creados pueden registrarse en un historial de versiones dentro de la opción "confluencia", lo que permite realizar un seguimiento de los cambios.

**Figura 20.** Draw.io



- **Canva:**

Canva es una herramienta de diseño digital en línea que ofrece a sus usuarios la posibilidad de elaborar una extensa gama de creaciones visuales, desde diapositivas y contenido para medios sociales hasta afiches, tarjetas de invitación, emblemas corporativos y mucho más. Su popularidad abarca tanto a profesionales del diseño como a principiantes en el campo, gracias a su interfaz de uso sencillo y su amplio abanico de características. Canva simplifica el proceso de generar diseños de aspecto profesional sin requerir la instalación de programas adicionales, ya que funciona como una aplicación basada en la web y en dispositivos móviles, permitiendo su uso desde cualquier aparato con conexión a internet.

- **Características:**

Una de las principales características de Canva es su accesibilidad y facilidad de uso. La plataforma se basa en un sistema de "arrastrar y soltar" que permite a los usuarios seleccionar y mover elementos gráficos fácilmente dentro del diseño. Canva ofrece una vasta biblioteca de plantillas prediseñadas, imágenes, íconos, fuentes y otros recursos gráficos que los usuarios pueden personalizar para crear diseños únicos.

Además, Canva permite a los usuarios trabajar de manera colaborativa, facilitando la edición compartida de proyectos en tiempo real. Esto es especialmente útil en entornos educativos y empresariales donde varios usuarios pueden aportar ideas y realizar ajustes simultáneamente. Canva también ofrece una opción de almacenamiento en la nube, lo que permite guardar y acceder a los diseños desde cualquier lugar.

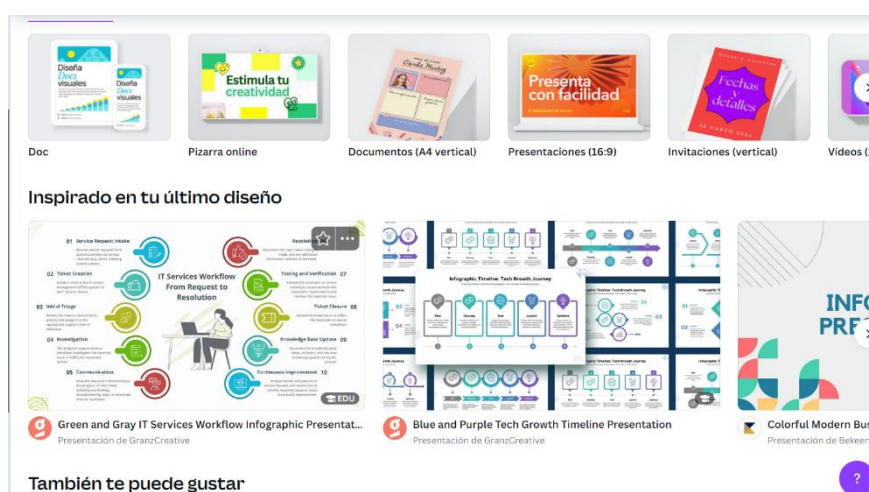
- **Funcionalidades:**

Canva permite la creación de una amplia variedad de diseños gráficos, que van desde publicaciones para redes

sociales y carteles hasta infografías y folletos. Su amplia biblioteca de plantillas y recursos visuales facilita la creación rápida de contenido, y sus opciones de personalización permiten ajustar los colores, fuentes y disposición de los elementos según las necesidades del usuario.

Además, Canva permite la creación de presentaciones y ofrece herramientas para animar elementos gráficos, lo que le da un aspecto más dinámico a los diseños. Una de las funcionalidades más apreciadas es la posibilidad de descargar los diseños en múltiples formatos, como PNG, JPG, PDF e incluso GIF, dependiendo de las necesidades del proyecto.

**Figura 21.** Canva



## 2.2. Marco teórico de las formulas y/o sistemas:

Los indicadores son medidas cuantificables que reflejan el desempeño de procesos, actividades o sistemas dentro de una organización. El desempeño, como concepto estratégico, abarca la eficiencia, productividad y rentabilidad, asociando los recursos utilizados con los objetivos alcanzados.

**Indicador:** Es una expresión matemática de lo que se quiere medir, con base en factores o variables claves y tienen un objetivo y cliente predefinido. Los indicadores de acuerdo a sus tipos (o referencias) pueden ser históricos, estándar, teóricos, por requerimiento de los usuarios, por lineamiento político, planificado, etc.

$$Indicador = \frac{a(unidad)}{b(unidad)}$$

### **Indicadores de gestión:**

Los indicadores de gestión son herramientas clave para medir y evaluar el desempeño de una organización en relación con sus objetivos estratégicos. Son expresiones cuantitativas que permiten evaluar el grado en el que se logran las metas establecidas, facilitando la toma de decisiones y el control de las actividades dentro de una empresa. En esencia, los indicadores ayudan a diagnosticar problemas, mejorar procesos y asegurar que la organización se mantenga alineada con su estrategia.

### **Importancia de los indicadores de gestión:**

- Sin indicadores, sería difícil saber si las actividades de una organización están siendo eficaces y eficientes. Estos permiten medir el rendimiento frente a las metas, identificar desviaciones y tomar medidas correctivas.
- Los indicadores proporcionan información valiosa que respalda las decisiones estratégicas, garantizando que estas se basen en datos reales y no en suposiciones.
- Ayudan a monitorear el progreso de la empresa a lo largo del tiempo, permitiendo ajustar las estrategias según sea necesario.



Los indicadores de gestión, en el control de inventario de joyería, son herramientas indispensables para medir, controlar y mejorar el desempeño organizacional. Proporcionan una base sólida para la toma de decisiones, la optimización de recursos y la alineación estratégica, permitiendo a la empresa adaptarse y prosperar en un entorno competitivo.

#### **Indicadores de cumplimiento:**

Con base en que el cumplimiento tiene que ver con la conclusión de una tarea. Los indicadores de cumplimiento están relacionados con las razones que indican el grado de consecución de tareas y/o trabajos.

Ejemplo: cumplimiento del programa de pedidos.

#### **Indicadores de evaluación:**

La evaluación tiene que ver con el rendimiento que se obtiene de una tarea, trabajo o proceso. Los indicadores de evaluación están relacionados con las razones y/o los métodos que ayudan a identificar nuestras fortalezas, debilidades y oportunidades de mejora.

#### **Indicadores de eficiencia:**

Teniendo en cuenta que eficiencia tiene que ver con la actitud y la capacidad para llevar a cabo un trabajo o una tarea con el mínimo de recursos. Los indicadores de eficiencia están relacionados con las razones que indican los recursos invertidos en la consecución de tareas y/o trabajos.

#### **Indicadores de eficacia:**

Eficaz tiene que ver con hacer efectivo un intento o propósito. Los indicadores de eficacia están relacionados con las razones que indican capacidad o acierto en la consecución de tareas y/o trabajos.

Para el control de inventario de joyería, los indicadores como la rotación de inventario, tasa de agotamiento de stock, precisión de previsión de demanda son importantes porque:

- Ayudan a mantener un balance óptimo entre satisfacer la demanda y minimizar el capital inmovilizado.
- Identifican tendencias ya que permiten reconocer patrones en la demanda de productos, facilitando la planificación.
- Ayudan a minimizar pérdidas por obsolescencia o exceso de inventario.
- Mejoran la Satisfacción del Cliente
- Aseguran la disponibilidad de productos.
- Contribuyen a la eficiencia operativa y financiera de la empresa.

Por ejemplo:

La rotación de inventario se calcula dividiendo el costo de las ventas por el promedio del inventario:

$$\text{Rotación de inventario} = \frac{\text{Costo de ventas}}{\text{Promedio del inventario}}$$

Supongamos que el costo anual de las ventas de aretes de plata es de s/ 12,000 y el valor promedio del inventario de aretes durante este mismo periodo es de s/ 3,000.

$$\text{Rotacion de inventario} = \frac{12,000}{3,000} = 4$$

Conclusión:

Esto indica que el inventario de aretes de plata se vendió y reemplazó cuatro veces durante el año. Si la empresa encuentra

que este número es bajo, podría buscar estrategias para aumentar las ventas y mejorar la eficiencia del inventario.

### 3. CAPÍTULO 3 – DESARROLLO DE LA SOLUCIÓN

#### 3.1. Prototipos

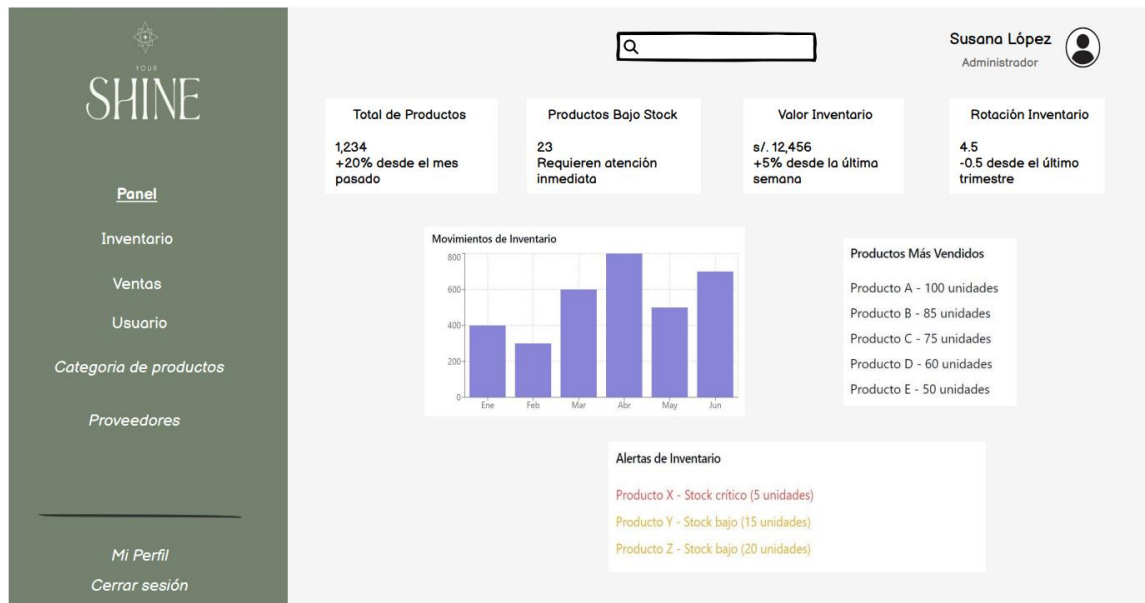
**Figura 22.** Login



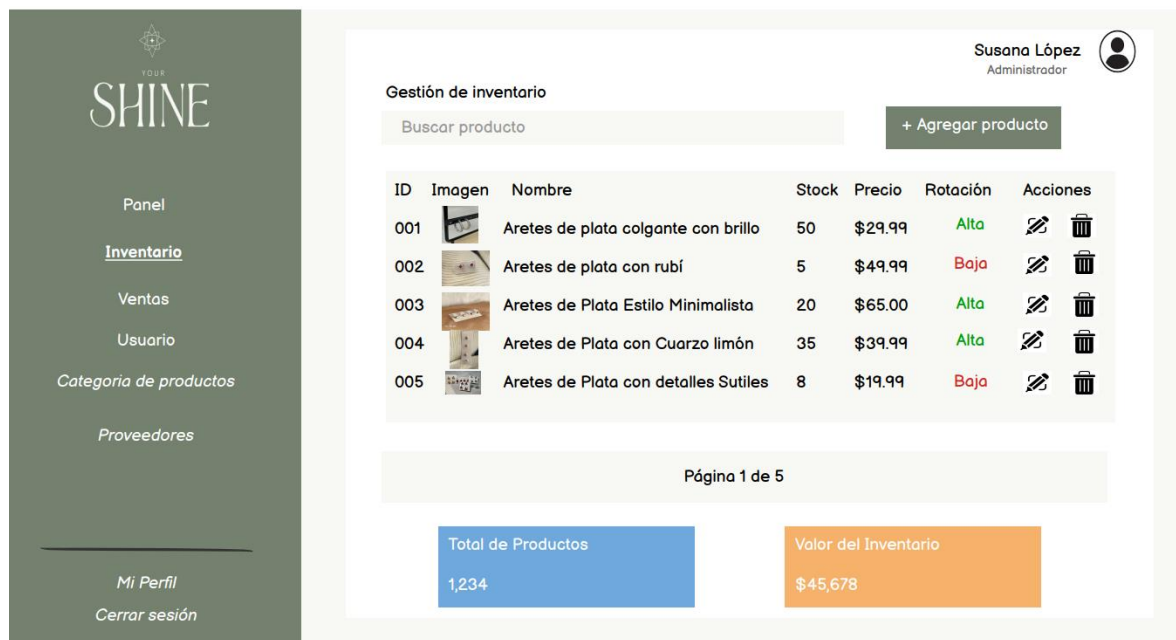
The image shows a login form for a system named "YOUR SHINE". At the top is a logo consisting of a stylized star or flower shape. Below the logo, the text "YOUR SHINE" is displayed in a large, serif font. The form includes two input fields: "Usuario:" and "Contraseña:". Below these fields is a reCAPTCHA checkbox with the text "I'm not a robot" and a small reCAPTCHA logo. A button labeled "Acceder" is positioned below the reCAPTCHA. At the bottom of the form, there is a link that says "¿Olvidaste tu contraseña?". The entire form is set against a light beige background, which is itself centered within a larger, dark olive-green rectangular frame.

**Nota.** Formulario para inicio de sesión

**Figura 23. Dashboard**



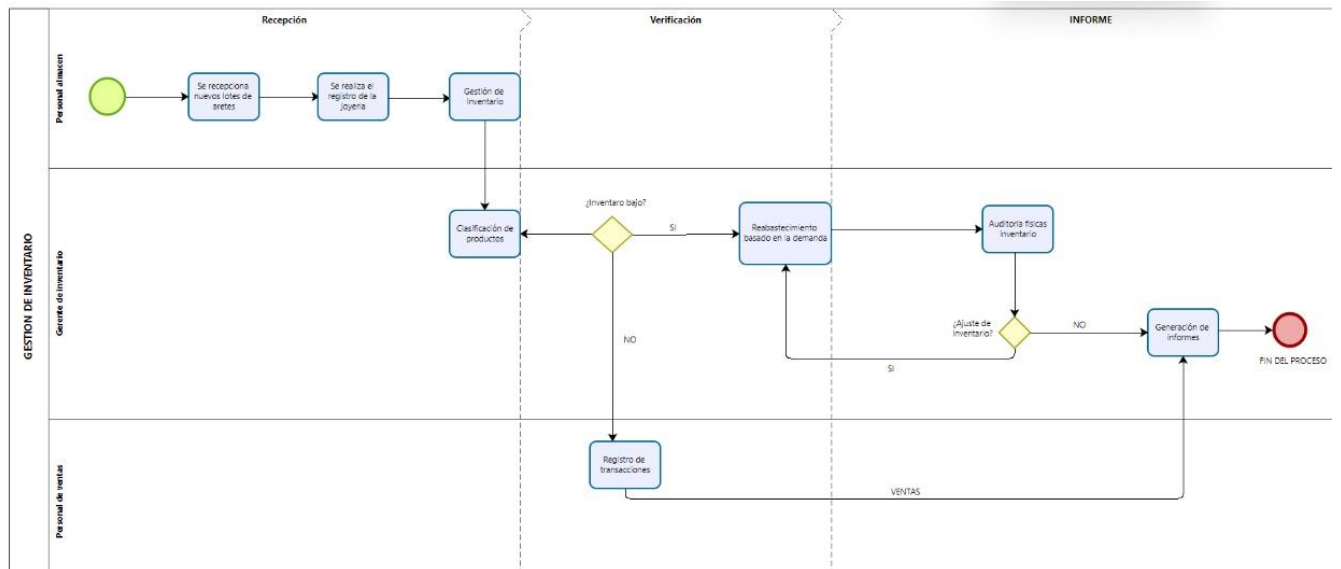
**Figura 24. Inventario**



## 3.2. Diagramas

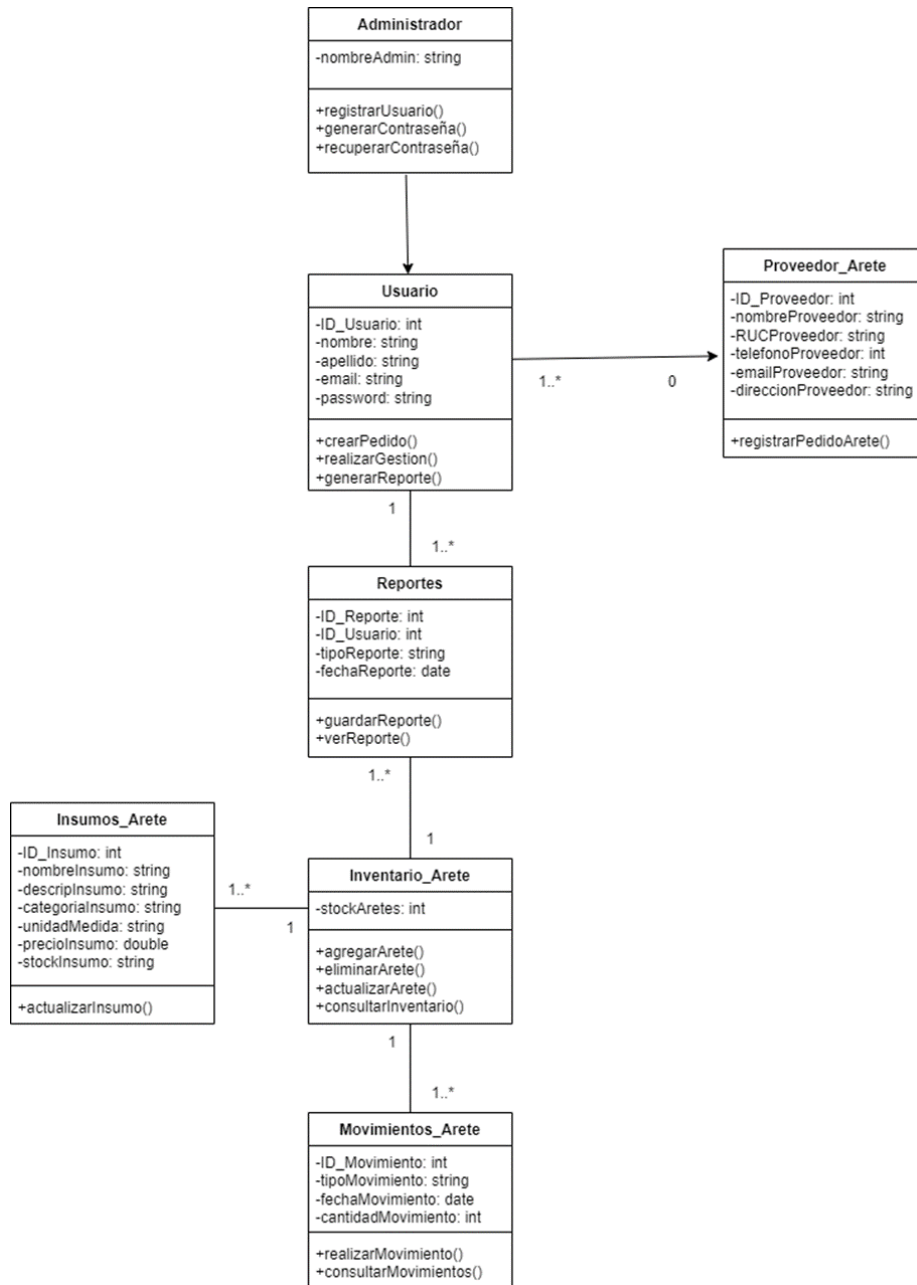
### 3.2.1. Modelo de negocios

Figura 25. Modelo de negocios



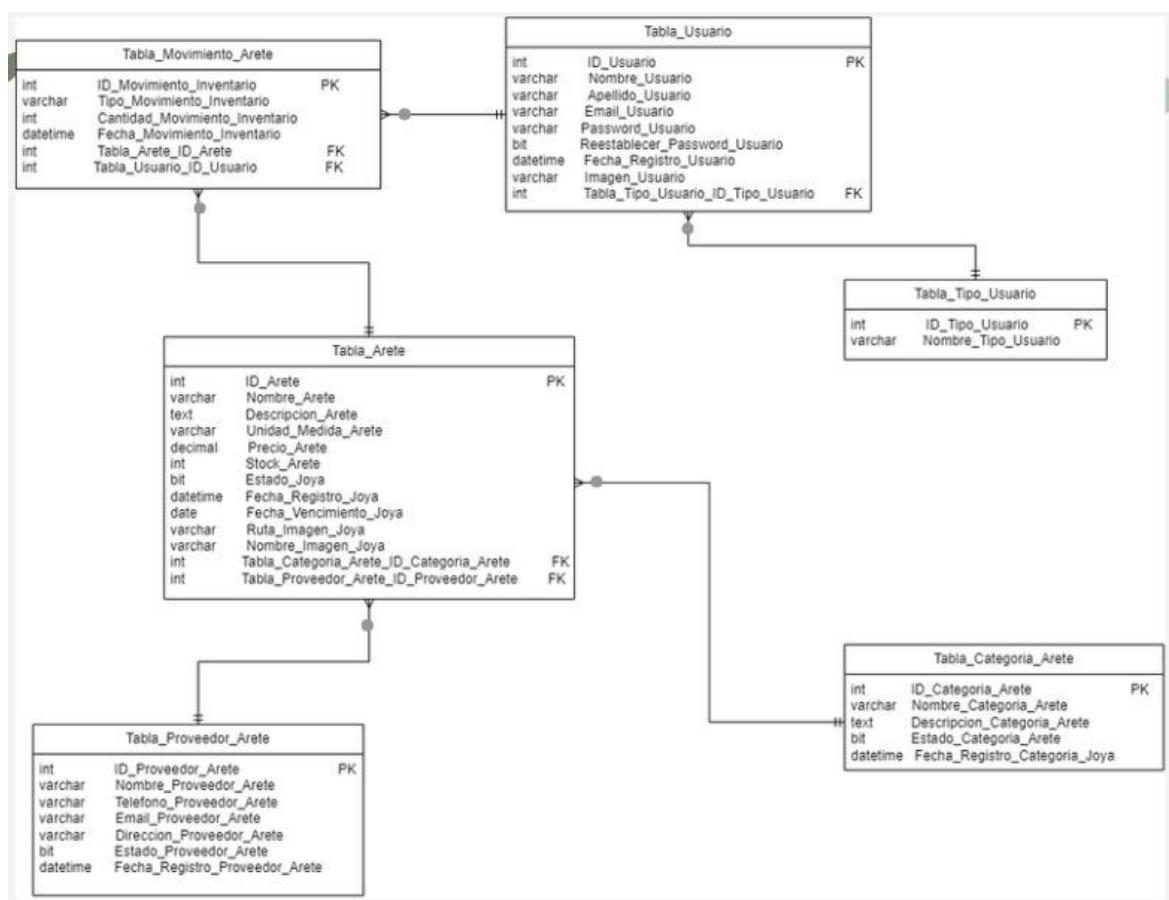
### 3.2.2. Diagrama clases

**Figura 26.** Diagrama de clases



### 3.2.3. Diagrama de entidad – relación

**Figura 27.** Diagrama de entidad relación



### 3.2.4. Diccionario de datos

#### Tabla\_Tipo\_Usuario

Llave	Campo	Tipo	Tamaño	Descripción
PK	ID_Tipo_Usuario	INT	-	Identificador único del tipo de usuario
	Nombre_Tipo_Usuario	VARCHAR	50	Nombre del tipo de usuario (ej. Administrador, Vendedor)

#### Tabla\_Usuario

Llave	Campo	Tipo	Tamaño	Descripción
PK	ID_Usuario	INT	-	Identificador único del usuario
	Nombre_Usuario	VARCHAR	50	Nombre del usuario
	Apellido_Usuario	VARCHAR	50	Apellido del usuario
	Email_Usuario	VARCHAR	100	Correo electrónico del usuario
	Password_Usuario	VARCHAR	125	Contraseña encriptada del usuario
	Reestablecer_Password_Usuario	BIT	1	Indica si el usuario debe reestablecer su contraseña
	Fecha_Registro_Usuario	DATETIME	-	Fecha y hora de registro del usuario
	Imagen_Usuario	VARCHAR	255	Ruta de la imagen de perfil del usuario
FK	Tabla_Tipo_Usuario_ID_Tipo_Usuario	INT	-	ID del tipo de usuario asociado

#### Tabla\_Categoria\_Arete

Llave	Campo	Tipo	Tamaño	Descripción
PK	ID_Categoria_Arete	INT	-	Identificador único de la categoría
	Nombre_Categoria_Arete	VARCHAR	50	Nombre de la categoría del arete
	Descripcion_Categoria_Arete	TEXT	-	Descripción detallada de la categoría
	Estado_Categoria_Arete	BIT	1	Estado activo/inactivo de la categoría
	Fecha_Registro_Categoria_Arete	DATETIME	-	Fecha y hora de registro de la categoría

#### Tabla\_Proveedor\_Arete

Llave	Campo	Tipo	Tamaño	Descripción
PK	ID_Proveedor_Arete	INT	-	Identificador único del proveedor
	Nombre_Proveedor_Arete	VARCHAR	100	Nombre del proveedor
	Telefono_Proveedor_Arete	VARCHAR	20	Número de teléfono del proveedor
	Email_Proveedor_Arete	VARCHAR	100	Correo electrónico del proveedor
	Direccion_Proveedor_Arete	VARCHAR	255	Dirección del proveedor
	Estado_Proveedor_Arete	BIT	1	Estado activo/inactivo del proveedor
	Fecha_Registro_Proveedor_Arete	DATETIME	-	Fecha y hora de registro del proveedor

#### Tabla\_Arete

Llave	Campo	Tipo	Tamaño	Descripción
PK	ID_Arete	INT	-	Identificador único para cada arete
	Nombre_Arete	VARCHAR	100	Nombre comercial del arete.
	Descripcion_Arete	TEXT	-	Detalles específicos del arete
	Unidad_Medida_Arete	VARCHAR	20	Medida
	Precio_Arete	DECIMAL	-10,2	Precio de venta del arete.
	Stock_Arete	INT	-	Cantidad de aretes disponibles.
	Estado_Joya	BIT	1	Indica si el arete está disponible para la venta.
	Fecha_Registro_Joya	DATETIME	-	Fecha y hora de registro del arete
	Fecha_Vencimiento_Joya	DATETIME	-	Fecha de vencimiento o revisión del arete
	Ruta_Imagen_Joya	VARCHAR	255	Ruta de la imagen del arete
	Nombre_Imagen_Joya	VARCHAR	255	Nombre del archivo de imagen del arete
FK	Tabla_Categoria_Arete_ID_Categoria_Arete	INT	-	ID de la categoría asociada
FK	Tabla_Proveedor_Arete_ID_Proveedor_Arete	INT	-	ID del proveedor asociado

#### Tabla\_Movimiento\_Arete

Llave	Campo	Tipo	Tamaño	Descripción
PK	ID_Movimiento_Inventario	INT	-	Identificador del movimiento.
	Tipo_Movimiento_Inventario	VARCHAR	20	Especifica si es un movimiento de entrada o salida
	Cantidad_Movimiento_Inventario	INT	-	Cantidad de aretes en el movimiento.
	Fecha_Movimiento_Inventario	DATETIME	-	Fecha del movimiento.
FK	Tabla_Arete_ID_Arete	INT	-	ID de la joya asociada al movimiento
FK	Tabla_Usuario_ID_Usuario	INT	-	ID del usuario que realizó el movimiento



## BIBLIOGRAFÍA

Serrato-Violeth, J. P., & Ramirez-Castillo, C. A. (2023). Aplicación de escritorio para gestión e inventario. Recuperado de: <https://repositorio.uniagustiniana.edu.co/handle/123456789/2456>

Villacis Aguachela, A. E. (2022). Control interno de inventarios de la Empresa "Metaltexcorpsa SA" de la ciudad de Babahoyo en el año 2021 (Bachelor's thesis, Babahoyo: UTB-FAFI. 2022). Recuperado de: <http://dspace.utb.edu.ec/handle/49000/12957>

Mora Garcia, L. A. (2009). INDICADORES DE LA GESTIÓN LOGÍSTICA KPI "Los indicadores claves del desempeño logístico". Recuperado de: [https://www.fesc.edu.co/portal/archivos/e\\_libros/logistica/ind\\_logistica.pdf](https://www.fesc.edu.co/portal/archivos/e_libros/logistica/ind_logistica.pdf)

Frechina, A. (2018). Metodología Scrum ¿Qué es? Recuperado de: <https://winred.es/management/metodologia-scrum-que-es/gmx-niv116-con24594.htm>

Pantoja, E. B. (2004). El patrón de diseño Modelo-Vista-Controlador (MVC) y su implementación en Java Swing. Acta Nova, 2(4), 493. <https://core.ac.uk/download/pdf/287301069.pdf>

Navarrete, T. (2006). El lenguaje JavaScript. Asignatura: Fundamentos de Cartografía i SIG. <https://books.google.es/books?hl=es&lr=&id=3x09sewjaHIC&oi=fnd&pg=PA3&dq=javascript&ots=YhJIOLyVvB&sig=FHVGJISuQuSERp6ZuNqinHFf45Y>

Osaín, C. L. (s/f). Indicadores de Gestión. Ucipfg.com. Recuperado el 3 de octubre de 2024, de

## Anexos

### Anexo 1. Lean Canvas

