



**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 - 2

Lima, Perú

# Índice

## Contenido

<b>INTRODUCCIÓN</b>	4
<b>1. CAPÍTULO 1 – ASPECTOS GENERALES</b>	5
1.1. Descripción del problema:	5
1.2. Definición de objetivos:	15
<b>1.2.1. Objetivo principal:</b>	<b>15</b>
<b>1.2.2. Objetivos específicos:</b>	<b>15</b>
1.3. Alcances y Limitaciones:	16
<b>1.3.1. Alcances:</b>	<b>16</b>
<b>1.3.2. Limitaciones:</b>	<b>24</b>
1.4. Justificación:	25
1.5. Estado del arte:	26
<b>2. CAPÍTULO 2 – MARCO TEÓRICO</b>	30
2.1. Marco teórico de programación orientada a objetos	30
<b>2.1.1. Metodología:</b>	<b>30</b>
2.1.2. Modelo:	35
<b>2.1.3. Patrones GUI:</b>	<b>36</b>
<b>2.1.4. Patrones de desarrollo:</b>	<b>41</b>
<b>2.1.5. Lenguajes de programación o frameworks:</b>	<b>45</b>
<b>2.1.6. Gestor de base de datos:</b>	<b>53</b>
<b>2.1.7. Herramientas adicionales:</b>	<b>55</b>
2.2. Marco teórico de las fórmulas y/o sistemas:	60
<b>3. CAPÍTULO 3 – DESARROLLO DE LA SOLUCIÓN</b>	64
3.1. Prototipos	64
3.2. Diagramas	66
<b>3.2.1. Modelo de negocios</b>	<b>66</b>
<b>3.2.2. Diagrama clases</b>	<b>67</b>
<b>3.2.3. Diagrama de entidad – relación</b>	<b>68</b>
<b>3.2.4. Diccionario de datos</b>	<b>68</b>
<b>4. CAPÍTULO 4 – RESULTADOS</b>	70
4.1. Resultados de encuesta	70
4.2. Presupuesto	76
<b>CONCLUSIONES</b>	77
<b>RECOMENDACIONES</b>	79

<b>BIBLIOGRAFÍA.....</b>	<b>80</b>
<b>ANEXOS .....</b>	<b>84</b>

## 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 un sistema web 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 inadecuados 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.

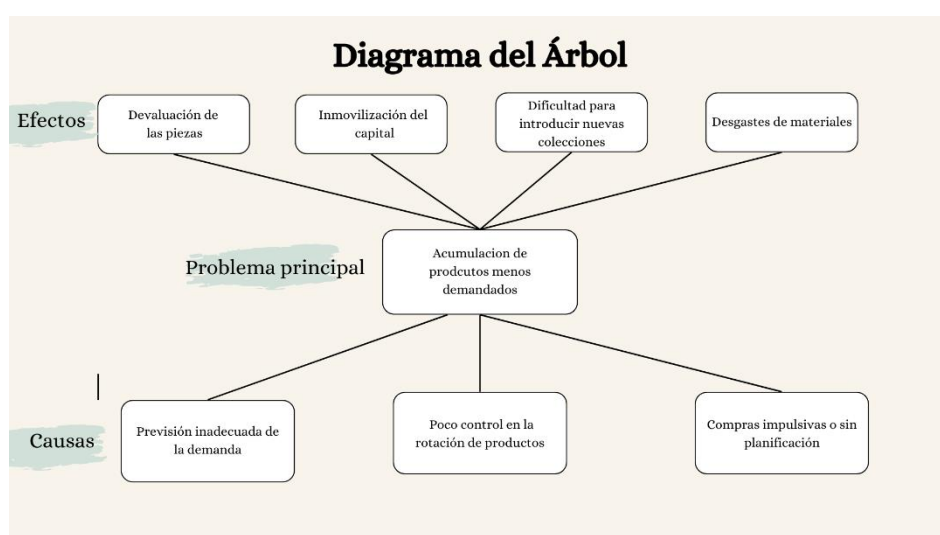
La gestión de inventario en tiempo real limitada impide tomar decisiones informadas sobre reabastecimiento y ventas, afectando las transacciones al no contar con información precisa sobre la disponibilidad de productos. Asimismo, las decisiones de reabastecimiento deficientes surgen de la falta de datos 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 y 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 un sistema web diseñado 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:

### **Indicadores de Gestión:**

#### **Calidad de los pedidos generados:**

**Definición:** Este indicador mide el porcentaje de pedidos generados correctamente, evaluando si cumplen con los criterios de precisión y coherencia en la información, como la cantidad solicitada, especificaciones del producto, y detalles de entrega. La calidad de los pedidos es fundamental para evitar errores en la preparación y distribución, mejorando la satisfacción del cliente.

#### **Fórmula:**

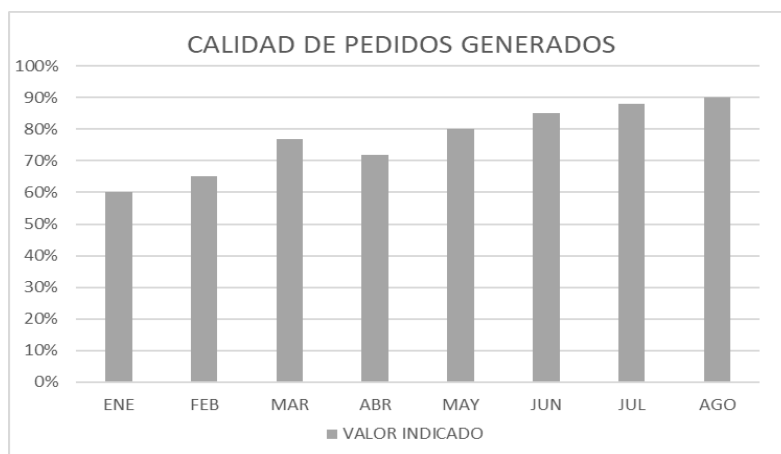
$$\frac{\text{Número de pedidos generados correctamente}}{\text{Número total de pedidos generados}} \times 100$$

**Tabla 1:** Tabla de tasa de calidad de pedidos generados

	INFORMACIÓN A INGRESAR		
MES	PEDIDOS GENERADOS SIN PROBLEMAS	TOTAL DE PEDIDOS GENERADOS	VALOR INDICADO
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			

Fuente: Elaboración propia

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



Fuente: Elaboración propia

### Entregas perfectamente recibidas:

**Definición:** Este indicador evalúa la precisión y satisfacción de las entregas desde el punto de vista del cliente. Una entrega perfectamente recibida significa que el pedido llegó a tiempo, en la cantidad correcta, sin daños, y con toda la documentación necesaria. Esto se calcula en



función del número total de entregas y las que fueron recibidas sin problemas reportados.

**Fórmula:**

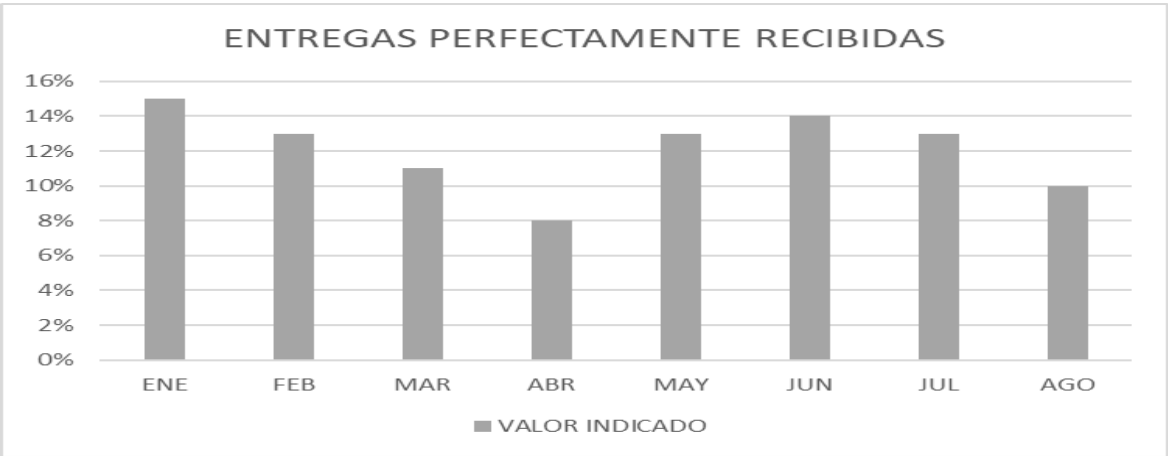
$$\frac{\text{Número de pedidos rechazados}}{\text{Número total de entregas}} \times 100$$

**Tabla 2:** Tasa de entregas perfectamente recibidas

	INFORMACIÓN A INGRESAR		
MES	PEDIDOS RECHAZADOS	TOTAL DE 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%
AGO	2	20	10%
Año:2024			

Fuente: Elaboración propia

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



Fuente: Elaboración propia

### Índice de Devoluciones por Defectos

**Definición:** Este indicador mide el porcentaje de aretes devueltos por defectos en comparación con el total de aretes vendidos. Ayuda a identificar y evaluar problemas de calidad en los productos, permitiendo detectar patrones de defectos y tomar acciones para mejorar la calidad y reducir las devoluciones.

**Fórmula:**

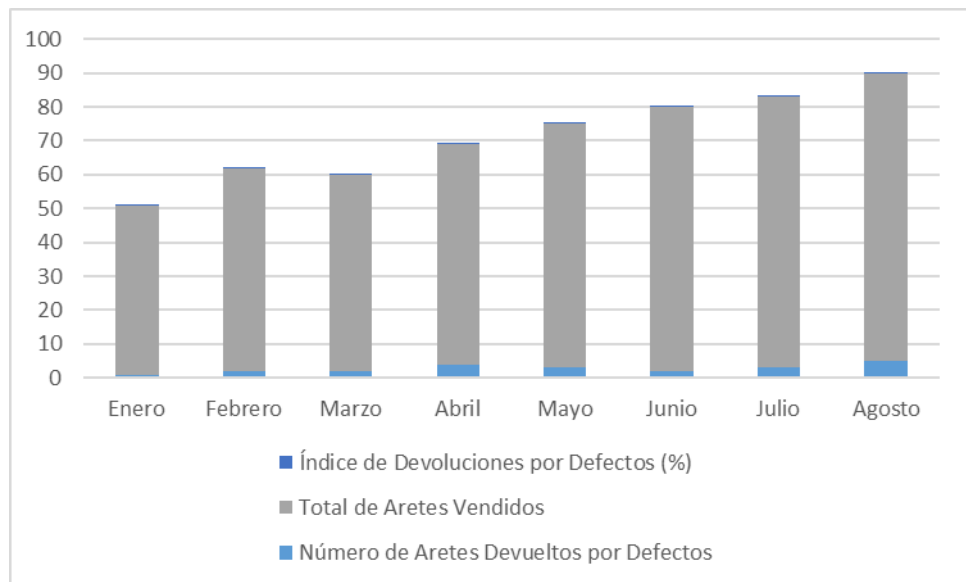
$$\frac{\text{Número de Aretes Devueltos por Defectos}}{\text{Total de aretes vendidos}} \times 100$$

**Tabla 3: Índice de devoluciones por defecto**

Mes	Número de Aretes Devueltos por Defectos	Total de Aretes Vendidos	Índice de Devoluciones por Defectos (%)
Enero	1	50	2.00%
Febrero	2	60	3.30%
Marzo	2	58	3.45%
Abril	4	65	6.15%
Mayo	3	72	4.10%
Junio	2	78	1.87%
Julio	3	80	2.56%
Agosto	5	85	5.88%

Fuente: Elaboración propia

Gráfico 3: Índice de Devoluciones por Defectos



Fuente: Elaboración propia

### Indicadores de utilización:

#### Índice de rotación de aretes:

**Definición:** Este indicador mide cuántas veces el inventario de aretes se ha renovado en un periodo específico. Es útil para gestionar la reposición de productos, ya que permite entender la velocidad con la que se venden los aretes y ayuda a planificar el inventario para evitar tanto excesos como faltantes.

#### Formula:

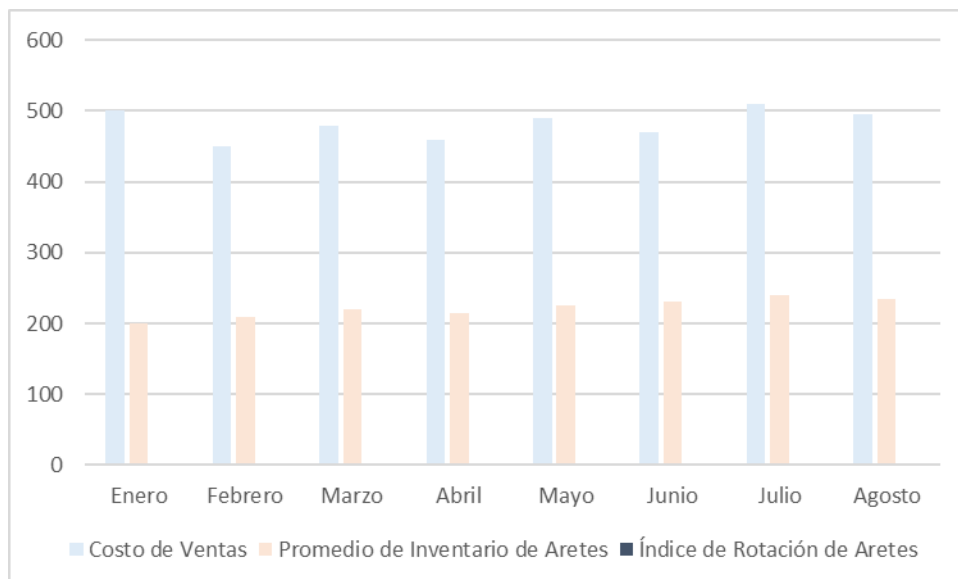
$$\frac{\text{Costo de ventas}}{\text{Promedio de inventario de aretes}}$$

Tabla 4: Índice de rotación de artes

Mes	Costo de Ventas	Promedio de Inventario de Aretes	Índice de Rotación de Aretes
Enero	500	200	2.5
Febrero	450	210	2.14
Marzo	480	220	2.18
Abril	460	215	2.14
Mayo	490	225	2.18
Junio	470	230	2.04
Julio	510	240	2.13
Agosto	495	235	2.11

Fuente: Elaboración propia

Gráfico 4: Índice de rotación de artes



Fuente: Elaboración propia

### Tasa de disponibilidad de modelos

**Definición:** Mide la disponibilidad de los diferentes modelos de aretes, lo cual es clave para asegurar variedad y satisfacer la demanda.

**Formula:**

$$\frac{\text{Modelo de aretes disponibles}}{\text{Total de modelos de aretes}} \times 100$$

**Indicadores de productividad**

**Tasa de productos obsoletos**

**Definición:** Este indicador mide el porcentaje de aretes que han permanecido en inventario por más de seis meses sin venderse. Es fundamental para identificar productos que pueden estar generando pérdidas por devaluación.

**Fórmula:**

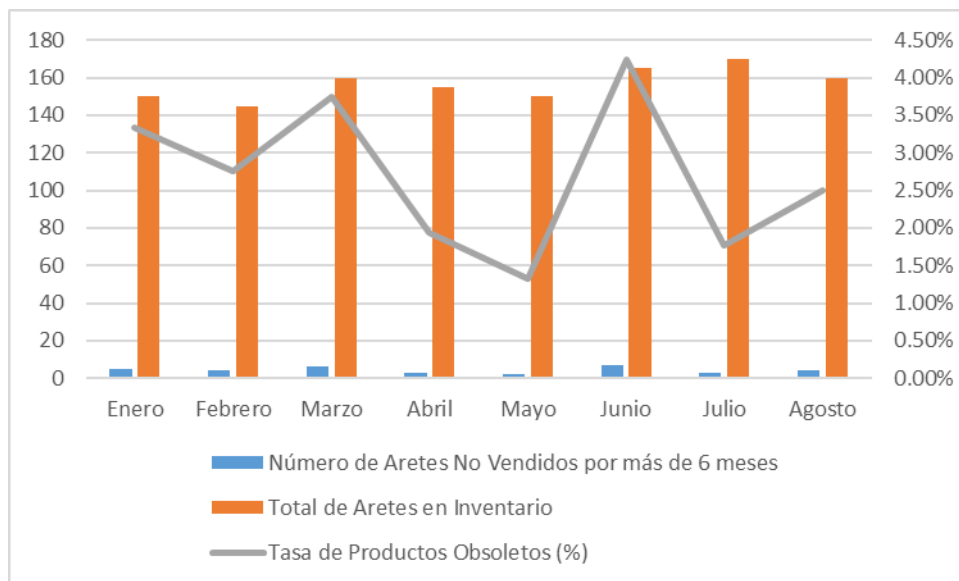
$$\frac{\text{Número de Aretes No Vendidos por más de 6 meses}}{\text{Total de aretes en inventario}}$$

Tabla 5: Tasa de productos obsoletos

Mes	Número de Aretes No Vendidos por más de 6 meses	Total de Aretes en Inventario	Tasa de Productos Obsoletos (%)
Enero	5	150	3.33%
Febrero	4	145	2.76%
Marzo	6	160	3.75%
Abril	3	155	1.94%
Mayo	2	150	1.33%
Junio	7	165	4.24%
Julio	3	170	1.76%
Agosto	4	160	2.50%

Fuente: Elaboración propia

Gráfico 5: Tasa de productos obsoletos



Fuente: Elaboración propia

## Índice de popularidad

**Definición:** Mide la demanda de cada diseño, permitiendo ajustar la oferta y producción a los productos con mayor aceptación.

**Fórmula:**

$$\frac{\text{Ventas de un modelo de arete}}{\text{Total de ventas de aretes}} \times 100$$

## Rotación de inventario

**Definición:** Mide cuántas veces se ha vendido y reemplazado el inventario en un periodo determinado.

**Fórmula:**

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

## Tiempo de reposición

**Definición:** Mide el tiempo promedio necesario para reponer un producto en inventario, optimizando la eficiencia operativa.

**Fórmula:**

$$\frac{\text{Tiempo total de reposición}}{\text{Número de reposiciones}}$$

## Cumplimiento de plazos de reposición

**Definición:** Mide el porcentaje de reposiciones que se cumplen dentro del tiempo establecido, asegurando un flujo adecuado de inventario.

**Fórmula:**

$$\frac{\text{Número de reposiciones a tiempo}}{\text{Total de reposiciones}} \times 100$$

### 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.

Tabla 7. Leyenda de prioridad

<b>Leyenda de Prioridad</b>	
<b>Prioridad</b>	<b>Descripción</b>
<b>Alta</b>	Requerimientos críticos para el funcionamiento básico y necesario del sistema. Estos requerimientos deben implementarse primero.
<b>Media</b>	Requerimientos importantes, pero no esenciales para el funcionamiento básico. Pueden desarrollarse después de los de prioridad alta
<b>Baja</b>	Requerimientos opcionales o de mejora, que pueden posponerse sin afectar las funcionalidades principales del sistema



Tabla 8

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.	

**Fuente: Elaboración propia**

Tabla 9

Especificación de requerimientos funcionales	
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.	

**Fuente: Elaboración propia**

**Tabla 10**

<b>Especificación de requerimientos funcionales</b>	
Requerimiento funcional N°:	Nombre:
RQF03	Búsqueda de Productos
Tipo:	Prioridad:
Vendedor/Inventario	Alta
Descripción	
Debe permitir buscar productos específicos por nombre o categoría en el inventario para agilizar el proceso de consulta.	

**Fuente: Elaboración propia**

**Tabla 11**

<b>Especificación de requerimientos funcionales</b>	
Requerimiento funcional N°:	Nombre:
RQF04	Generación de Reportes
Tipo:	Prioridad:
Gerente de Inventario	Media
Descripción	
El sistema debe generar reportes mensuales de ventas y movimientos de productos para análisis de inventario.	

**Fuente: Elaboración propia**

**Tabla 12**

<b>Especificación de requerimientos funcionales</b>	
Requerimiento funcional N°:	Nombre:
RQF05	Notificación de Bajo Stock
Tipo:	Prioridad:
Jefe de Inventarios	Alta
Descripción	
El sistema debe notificar automáticamente cuando el inventario de un producto esté por debajo de un umbral definido.	

**Fuente: Elaboración propia**

**Tabla 13**

<b>Especificación de requerimientos funcionales</b>	
Requerimiento funcional N°:	Nombre:
RQF06	Gestión de Usuarios
Tipo:	Prioridad:
Administrador	Alta
Descripción	
Debe permitir crear, editar y eliminar cuentas de usuario con diferentes niveles de acceso y permisos.	

**Fuente: Elaboración propia**

**Tabla 14**

<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.	

**Fuente: Elaboración propia**

**Tabla 15**

<b>Especificación de requerimientos no funcionales</b>	
Requerimiento funcional N°:	Nombre:

RQNOF02	Seguridad de Datos
Tipo:	Prioridad:
Desarrollador/Jefe de pruebas	Alta
Descripción	
El sistema debe utilizar protocolos de seguridad (como cifrado de datos sensibles) para evitar accesos no autorizados	
Manejo de errores:	
Si falla la verificación de seguridad, se debe notificar al administrador y registrar el intento no autorizado.	
Criterios de aceptación:	
Los datos sensibles deben estar encriptados y sólo accesibles por usuarios autorizados.	

**Fuente: Elaboración propia**

**Tabla 16**

Especificación de requerimientos no funcionales	
Requerimiento funcional N°:	Nombre:
RQNOF03	Usabilidad
Tipo:	Prioridad:
Desarrollador/Jefe de pruebas	Alta
Descripción	
La interfaz debe ser intuitiva y fácil de usar, permitiendo que los usuarios realicen tareas sin necesidad de capacitación extensa.	

Manejo de errores:
Mensajes claros al usuario para orientar en caso de campos obligatorios incompletos o acciones inválidas.
Criterios de aceptación:
Los usuarios deben poder navegar todas las funciones principales sin instrucciones adicionales.

**Fuente: Elaboración propia**

**Tabla 17**

<b>Especificación de requerimientos no funcionales</b>	
Requerimiento funcional N°:	Nombre:
RQNOF04	Tolerancia a Fallos
Tipo:	Prioridad:
Desarrollador/Administrador	Alta
Descripción	
El sistema debe seguir funcionando correctamente a pesar de fallos menores o problemas inesperados, como caídas de red momentáneas.	
Manejo de errores:	
Guardado automático de datos en caso de fallo de red y reconexión sin pérdida de información	
Criterios de aceptación:	
En caso de fallo de red, los datos no deben perderse y el sistema debe reintentar la conexión automáticamente hasta restablecerla.	

**Fuente: Elaboración propia**

**Tabla 18**

<b>Especificación de requerimientos no funcionales</b>	
Requerimiento funcional N°:	Nombre:
RQNOF05	Escalabilidad
Tipo:	Prioridad:
Desarrollador	Media
Descripción	
El sistema debe permitir añadir nuevos módulos o funcionalidades sin comprometer el rendimiento o requerir grandes modificaciones	
Manejo de errores:	
Si la ampliación afecta el rendimiento, debe sugerirse una revisión para optimización	
Criterios de aceptación:	
El sistema debe admitir la integración de nuevos módulos sin reducción notable de rendimiento en pruebas de carga.	

**Fuente: Elaboración propia**

**Tabla 19**

<b>Especificación de requerimientos no funcionales</b>	
Requerimiento funcional N°:	Nombre:
RQNOF06	Manejo de Errores
Tipo:	Prioridad:
Desarrollador/Jefe de pruebas	Alta

Descripción
El sistema debe mostrar mensajes de error claros y específicos cuando ocurra un fallo, sin interrumpir el flujo de trabajo del usuario. Los errores críticos deben registrarse para diagnóstico.
Manejo de errores:
Mensajes de error detallados para fallos comunes, e información de contacto para soporte técnico en errores críticos.
Criterios de aceptación:
El sistema debe mostrar mensajes de error específicos y legibles para cada error, y registrar errores críticos para futura revisión.

Fuente: Elaboración propia

#### 1.3.2. Limitaciones:

- La implementación del sistema web 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 del sistema web 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 del sistema web 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 el sistema web 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 estar limitado a la web, podría depender de una conexión estable a internet para su uso completo, lo que podría restringir el acceso o la funcionalidad en ubicaciones con conectividad limitada, afectando la capacidad de gestionar inventarios en tiempo real desde cualquier lugar.

#### 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**

The screenshot shows the 'Inventory' section of the Oddo application. The top navigation bar includes 'Inventory', 'Overview', 'Operations', 'Products', 'Reporting', and 'Configuration'. Below this, there are tabs for 'New', 'Order', 'Snooze', 'Order To Max', and 'Replenishment'. A '3 selected' indicator and an 'Actions' button are also visible. The main table displays a list of products with columns for 'Product', 'Location', 'On Hand', 'Forecast', 'Route', 'Min ...', 'Max...', and 'To Order'. The table is filtered by 'Physical Locations' and 'WH/Stock'. The 'TRIGGER' section on the left shows 'All', 'Auto', and 'Manual' options. The table data includes products like '[E-COM06] Corner Desk ...', '[E-COM09] Large Desk', '[FURN\_9001] Flipover', '[FURN\_9666] Table', '[FURN\_7777] Office Chair', and '[FURN\_8888] Office Lamp'.

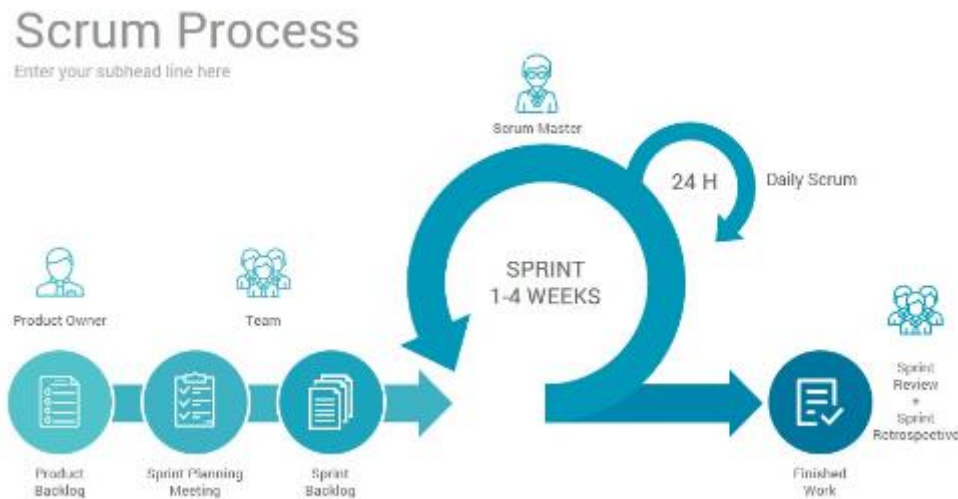
Product	Location	On Hand	Forecast	Route	Min ...	Max...	To Order
[E-COM06] Corner Desk ...	WH/Stock	4.00	-1.00		0.00	0.00	1.00
[E-COM09] Large Desk	WH/Stock	1.00	-4.00		0.00	0.00	4.00
[FURN_9001] Flipover	WH/Stock	5.00	-6.00		0.00	0.00	6.00
[FURN_9666] Table	WH/Stock	2.00	-1.00		0.00	0.00	1.00
[FURN_7777]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

### Nota.Oddo

#### 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 “Metaltexcorpsa SA” de la ciudad de Babahoyo en el año 2021**



UNIVERSIDAD TÉCNICA DE BABAHOYO  
FACULTAD DE ADMINISTRACIÓN, FINANZAS E INFORMÁTICA.  
PROCESO DE TITULACIÓN  
ABRIL 2022 – SEPTIEMBRE 2022  
EXAMEN COMPLEXIVO DE GRADO O DE FIN DE CARRERA  
PRUEBA PRÁCTICA

PREVIO A LA OBTENCIÓN DEL TÍTULO DE:  
LICENCIADO EN CONTABILIDAD Y AUDITORIA

TEMA:  
CONTROL INTERNO DE INVENTARIOS DE LA EMPRESA “METALTEXCORPSA  
S.A” DE LA CIUDAD DE BABAHOYO EN EL AÑO 2021

**Nota. Repositorio de la Universidad Técnica de Babahoyo. Control interno de inventarios de la Empresa “Metaltexcorpsa 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 de la metodología Scrum se reparten en 16 procesos o tareas, que a su vez se resumen en 5 pasos o etapas de implementación:

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

La primera fase se encarga de estudiar y analizar el proyecto identificando las necesidades básicas del sprint.

En el contexto de las metodologías ágiles, un sprint es un mini-proyecto con una duración no mayor a un mes que se interconecta con otros mini-proyectos para dirigirnos a los objetivos generales y específicos del proyecto general.

Las preguntas a hacer en la fase de inicio son:

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

La metodología Scrum da preferencia a la formación de equipos pequeños de mínimo 3 y máximo 5 personas, pues se facilita la fluidez de las ideas y se aporta creatividad al grupo.

Entre los primeros pasos de Scrum, tenemos 6 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.

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

La segunda fase de Scrum incluye normalmente los siguientes pasos:

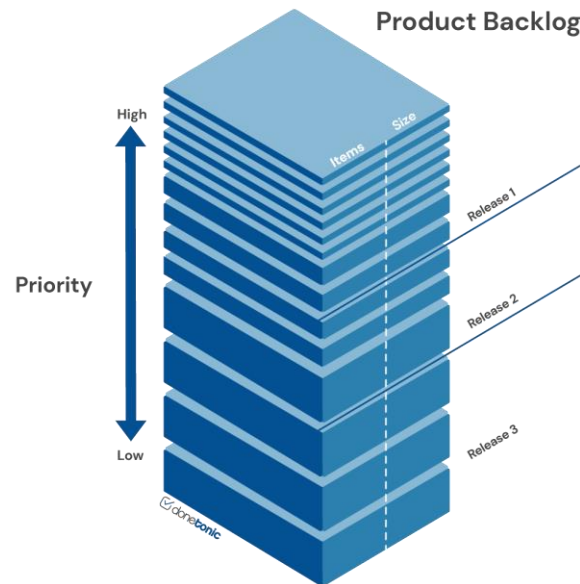
1. Crear, estimar y comprometer historias de usuario.
2. Identificar y estimar tareas.
3. Crear el sprint backlog o iteración de tareas.

La clave para llevar una buena administración de los proyectos es hacer una planificación y estimación del sprint, lo que te ayudará a establecer metas fijas y a cumplir con los plazos.

Tal vez esta sea la fase más importante del proyecto, pues si eres el Master Scrum tendrás que delegar las tareas correspondientes a cada grupo y hacer las estimaciones de tiempos de entrega, así como crear una lista ordenada para clasificar el trabajo según su prioridad.

Por ello, es necesario que utilices un Software para Scrum que facilite la asignación de tareas y “haga rodar” el trabajo en equipo.

**Figura 4. Product Backlog.**



### **C) Implementación:**

Al llegar a la tercera de las 5 fases de Scrum, nos topamos con la implementación del proyecto.

Es decir, la sala de reuniones donde se discute el sprint y se explora cómo optimizar el trabajo de cada grupo Scrum para darle forma definitiva al proyecto.

En la implementación se cumple con los siguientes procesos:

1. Crear entregables.
2. Realizar daily stand-up.
3. Refinanciamiento del backlog priorizado del producto.

En la fase de implementación o desarrollo no deberían hacerse cambios innecesarios de última hora (se supone que para evitarlo existe una fase de planificación).



Aún así, si necesitas hacer un movimiento que será clave para el éxito del sprint, no dudes en proceder.

#### d) Revisión y Retrospectiva:

Una vez que ya todo está maquetado e implementado, deberás hacer la revisión del proceso, que no es más que la autocrítica o evaluación interna del grupo respecto a su propio trabajo.

Es importante sumar opiniones constructivas y aportar soluciones viables.

Entre los pasos más importantes para realizar en esta fase tenemos:

1. Demostrar y validar el sprint.
2. Retrospectiva del sprint.

**Figura 5. Stakeholder.**



#### e) Lanzamiento:

La última de las fases del método Scrum es el lanzamiento.

Con esto nos referimos al desenlace del proyecto y entrega del producto, donde deberías cumplir con 2 únicas tareas que son:

1. Enviar entregables.
2. Enviar retrospectiva del proyecto.

Tabla 20. Cuadro comparativo

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.

Fuente: Elaboración propia

### 2.1.2. Modelo:

#### **Modelo Entidad-Relación**

Un ERD (Diagrama Entidad-Relación) es una representación gráfica que muestra las conexiones entre entidades en un mismo sistema. Estas pueden hacer referencia tanto a personas como a objetos o conceptos.

Estos modelos de entidad-relación se emplean de forma frecuente en la planificación y corrección de bases de datos relacionales en áreas como la ingeniería de software, sistemas de información empresarial, educación e investigación.

Elementos 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:**

Las entidades representan personas, objetos o conceptos que contienen información.

Dentro del modelo entidad-relación, se distinguen dos categorías: la entidad fuerte y la entidad débil.

#### **2. Relación:** Son conexiones que se crean entre un grupo de entidades.

Se dividen en tres categorías:

- **Relaciones binarias:** implican a dos entidades.
- **Relaciones ternarias:** involucran a tres entidades.
- **Relaciones n-arias:** abarcan n entidades.

#### **3. Atributo:** Son componentes que, en su conjunto, constituyen una entidad dentro del modelo entidad-relación. Es fácil concebirlos como las características que describen una entidad.

Cada grupo de entidades posee un conjunto de atributos asignados. Estos pueden categorizarse en clases:

- Simples o atómicos: color, edad, estatura...
- Compuestos: calle, número...
- De valor único: fecha de nacimiento, cédula...
- Multivalorados: números de teléfono, nombres de los hijos...
- Derivados: edad calculada a partir de la fecha de nacimiento o departamento a partir del municipio de residencia.

### 2.1.3. Patrones GUI:

Los Patrones de diseño o Design patterns son una parte imprescindible dentro de un design system. Estos patrones son elementos o agrupaciones de componentes que podemos ver repetidamente en un producto digital.

Podríamos decir que un patrón de diseño es una solución estandarizada a un “problema” o estado que se repite a lo largo del diseño de una interfaz o producto.

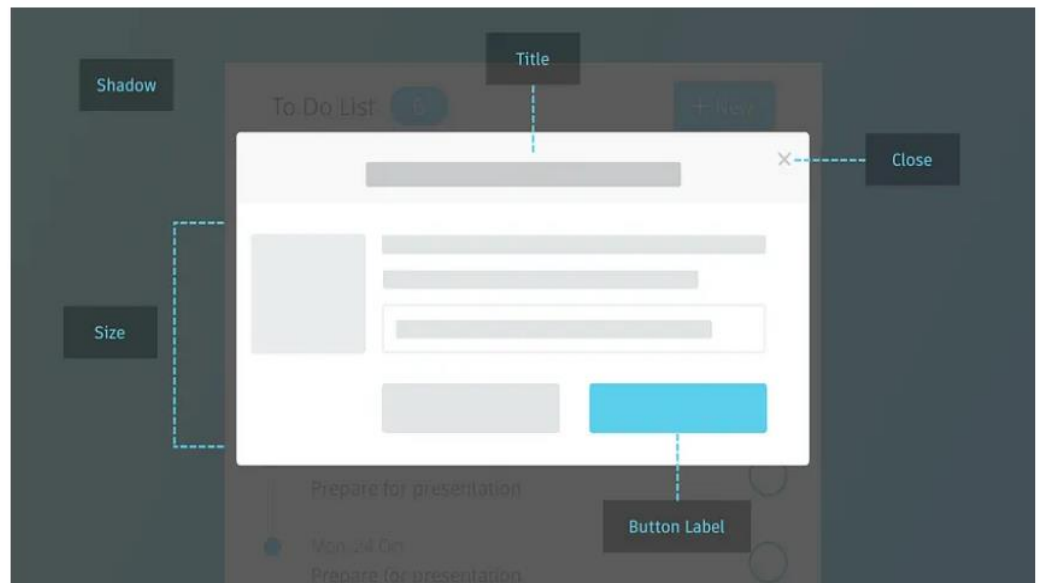
- Pantallas modales:

Son cuadros que aparecen sobre la página, bloqueando todas las funciones para concentrar el foco en una acción particular. Esta es su característica diferenciadora, le piden al usuario a realizar una acción. Por esto, normalmente usan llamados a la acción o preguntas como “¿Quieres borrar esta imagen?” o “Suscríbete”.

Las ventanas modales se activan cuando el usuario hace clic en un llamado o presiona un botón. Cuando son bien usadas,

previenen a los usuarios de errores y facilitan acciones al permitir realizarlas sin salir de la página.

**Figura 6. Pantalla Modales**



**Nota. Patrón GUI – Modales**

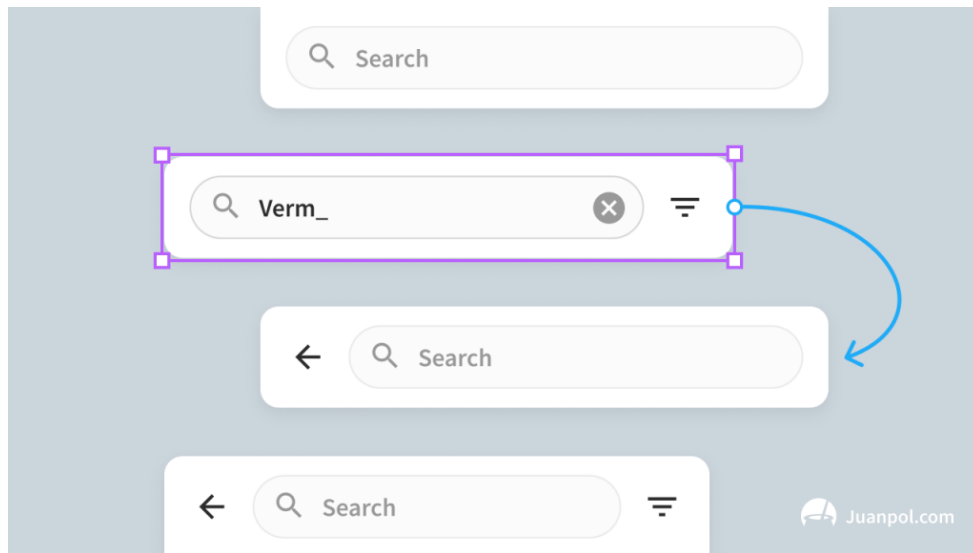
- Barra de navegación:

La barra de navegación es un elemento de interfaz de usuario que permite acceder a herramientas de navegación tanto unificadas como específicas del producto.

Las herramientas de navegación unificadas son las que se pueden encontrar en muchos productos de Autodesk. Las herramientas de navegación específicas del producto son exclusivas de un producto. La barra de navegación flota encima y a lo largo de uno de los laterales del área de dibujo actual.

Las herramientas de navegación se inician pulsando uno de los botones de la barra de navegación o seleccionando una de las herramientas en la lista que aparece al hacer clic en la parte más pequeña de un botón de división.

**Figura 7. Barra de navegación**



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

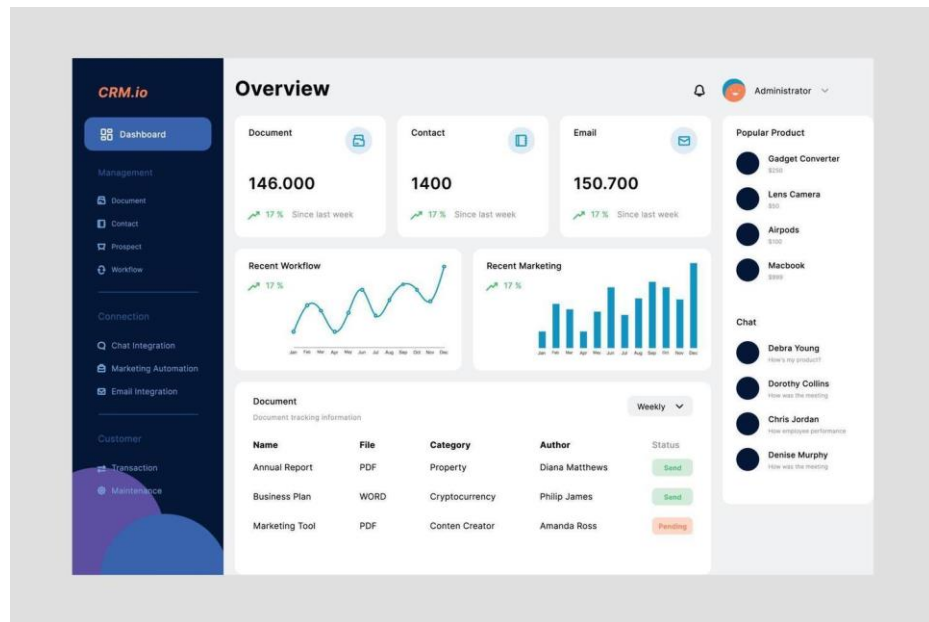
- **Dashboard:**

El dashboard o también conocido como panel de información o panel de gestión, es una interfaz gráfica que ayuda a los usuarios a visualizar indicadores clave de desempeño o KPI y métricas para la toma de decisiones que llevarán a la empresa a lograr sus objetivos. Por esta razón, muchos profesionales lo ven como un informe de progreso.

Básicamente, reúne una gran cantidad de datos que se encuentran disponibles en la organización, transformándolos en KPI, con tablas y gráficos

Los dashboards siempre deben mostrarse en forma de presentación visual en una sola pantalla del panel de control, lo que evita las barras de desplazamiento y facilita la realización de comparaciones y el seguimiento de los objetivos.

**Figura 8. Dashboard**



### **Nota. Patrón GUI - Dashboard**

- Captcha:

Un CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart: test de Turing público y automático para distinguir a los ordenadores de los humanos) es un tipo de medida de seguridad conocido como autenticación pregunta-respuesta. Un CAPTCHA te ayuda a protegerte del spam y del descifrado de contraseñas pidiéndote que completes una simple prueba que demuestre que eres humano y no un ordenador que intenta acceder a una cuenta protegida con contraseña.

**Figura 9. Captcha**



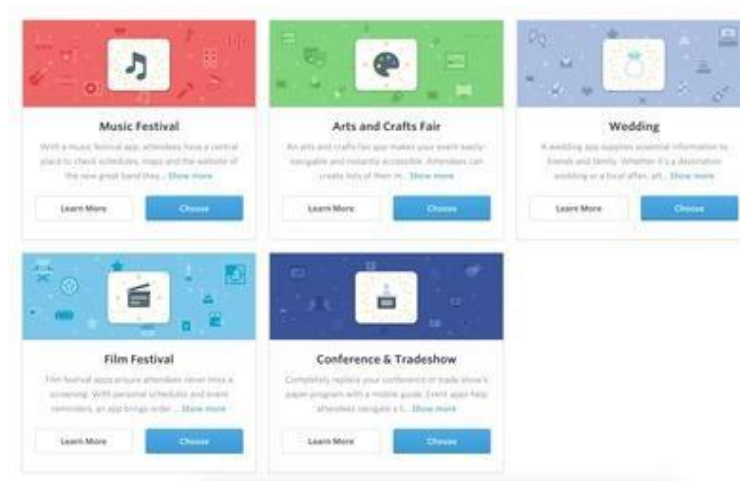
**Nota. Patrón GUI – Captcha**

- **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 en 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:**

MVC era inicialmente un patrón arquitectural, un modelo o guía que expresaba cómo organizar y estructurar los componentes de un sistema software, sus responsabilidades y las relaciones existentes entre cada uno de ellos.

Su nombre, MVC, parte de las iniciales de Modelo-Vista-Controlador (Model-View-Controller, en inglés), que son las capas o grupos de componentes en los que organizaremos nuestras aplicaciones bajo este paradigma.

- **Modelo (Model):**

En la capa Modelo encontraremos siempre una representación de los datos del dominio, es decir, aquellas entidades que nos servirán para almacenar información del sistema que estamos desarrollando. Por ejemplo, si estamos desarrollando una aplicación de facturación, en el modelo existirán las clases Factura, Cliente o Proveedor, entre otras.

- **Vista (View):**

Los componentes de la Vista son los responsables de generar la interfaz de nuestra aplicación, es decir, de componer las pantallas, páginas, o cualquier tipo de resultado utilizable por el usuario o cliente del sistema. De hecho, suele decirse que la Vista es una representación del estado del Modelo en un momento concreto y en el contexto de una acción determinada.

- **Controlador (Controller):**

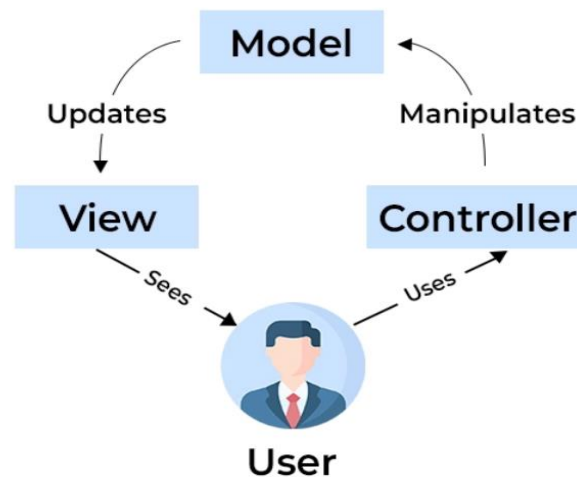
La misión principal de los componentes incluidos en el Controlador es actuar como intermediarios entre el usuario y el sistema. Serán capaces de capturar las acciones de éste

sobre la Vista, como puede ser la pulsación de un botón o la selección de una opción de menú, interpretarlas y actuar en función de ellas. Por ejemplo, retornando al usuario una nueva vista que represente el estado actual del sistema, o invocando a acciones definidas en el Modelo para consultar o actualizar información.

**Ventajas del MVC:** El uso del patrón MVC ofrece múltiples ventajas sobre otras maneras de desarrollar aplicaciones con interfaz de usuario, y en especial para la Web. Sin entrar en detalles aquí, por que la extensión del artículo ya es grande, comentaré a continuación algunas de ellas:

- La clara separación de responsabilidades impuesta por el uso del patrón MVC hace que los componentes de nuestras aplicaciones tengan sus misiones bien definidas. Por lo tanto, nuestros sistemas serán más limpios, simples, más fácilmente mantenibles y, a la postre, más robustos.
- Múltiples vistas a partir del mismo modelo, pudiendo reaprovechar mucho mejor los desarrollos y asegurando consistencia entre ellas.
- Mayor velocidad de desarrollo en equipo, que es consecuencia de lo anterior, ya que al estar separado en tres partes tan diferenciadas, diferentes programadores pueden ocuparse de cada parte en paralelo. Esto la hace ideal para el desarrollo de aplicaciones grandes.
- Facilidad para realización de pruebas unitarias.

**Figura 11. MVC.**



### **DAO:**

El patrón Data Access Object (DAO) pretende principalmente independizar la aplicación de la forma de acceder a la base de datos, o cualquier otro tipo de repositorio de datos. Para ello se centraliza el código relativo al acceso al repositorio de datos en las clases llamadas DAO. Fuera de las clases DAO no debe haber ningún tipo de código que acceda al repositorio de datos.

- Ventajas del Patrón DAO:
  - Modificar el API de acceso: Se podría cambiar el acceso a la base de datos de usar JDBC a usar Hibernate y sólo habría que modificar las clases DAO no afectando al resto de la aplicación.
  - Modificar el repositorio de datos: Sería posible que el acceso a los usuarios se hiciera mediante LDAP a un servicio de directorio en vez de estar dichos usuarios en una base de datos relacional. Como en el caso contrario sólo sería necesario cambiar las clases DAO y el resto de la aplicación no sería necesaria modificarla.
  - Implementación de triggers o listeners: Al estar todo el código centralizado en las clases DAO podríamos fácilmente implementar políticas de seguridad en el acceso al repositorio de datos. Mientras

que, en caso de no usarlo, sería imposible hacerlo ya que cualquiera podría acceder a los datos sin la obligación de pasar por dicha política de seguridad.

- 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**

Dado lo anterior, 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.

Un error común al implementar este patrón es no utilizar Entidades y en su lugar, regresar los objetos que regresan las mismas API's de las fuentes de datos, ya que esto obliga al BusinessObject tener una dependencia con

estas librerías, además, si la fuente de datos cambia, también cambiarán los tipos de datos, lo que provocaría una afectación directa al BusinessObject.

## **DTO:**

DTO o mejor dicho Data Transfer Object es un patrón de diseño, ya que si bien tiene efectos en la composición del proyecto añadiendo nuevas clases y carpetas o contenedores según lo definamos, no califica por sí sólo como un patrón de arquitectura ya que estos son esquemas generales que proporcionan pautas para la estructura y organización de sistemas de software a nivel macro.

El concepto es sencillo y poderoso a la vez, DTO nos dice que debemos crear una serie de objetos con propiedades que haga de capa intermedia entre la entidad (clase que se mapea directamente hacia la base de datos) y el agente que lo consuma: el cual puede ser por ejemplo una vista o podría ser directamente un cliente si hablamos de un API.

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

- **JavaScript:**

JavaScript (JS) es un lenguaje de programación ligero, interpretado, o compilado justo-a-tiempo (just-in-time) con funciones de primera clase. Si bien es más conocido como un lenguaje de scripting (secuencias de comandos) para páginas web, y es usado en muchos entornos fuera del navegador, tal como Node.js, Apache CouchDB y Adobe Acrobat JavaScript es un lenguaje de programación basada en prototipos, multiparadigma, de un solo hilo, dinámico, con soporte para programación orientada a objetos, imperativa y declarativa (por ejemplo programación funcional). Lee más en acerca de JavaScript.

## Ventajas:

- **Velocidad** - JavaScript tiende a ser muy rápido porque a menudo se ejecuta inmediatamente en el navegador. Entonces mientras no requiera recursos externos, JavaScript no tiene permitido retrasarse por llamados del servidor backend.
- **Simplicidad** - La sintaxis de JavaScript está inspirada por Java y es relativamente sencillo de aprender comparado a otros lenguajes de programación populares como C++.
- **Popularidad** - JavaScript esta por todas partes de la web, y con la llegada de Node.js, se ha incrementado su uso en backend. Hay incontables recursos para aprender JavaScript. Tanto StackOverflow como GitHub muestran un creciente número de proyectos que usan JavaScript, y la popularidad que ha alcanzado en los recientes años se espera que siga creciendo.
- **Compatibilidad**- A diferencia de PHP u otros lenguajes scripting, JavaScript puede ser usado en cualquier página web. JavaScript puede ser usado en diferentes tipos de aplicaciones gracias al soporte en otros lenguajes como Pearl y PHP.

**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:

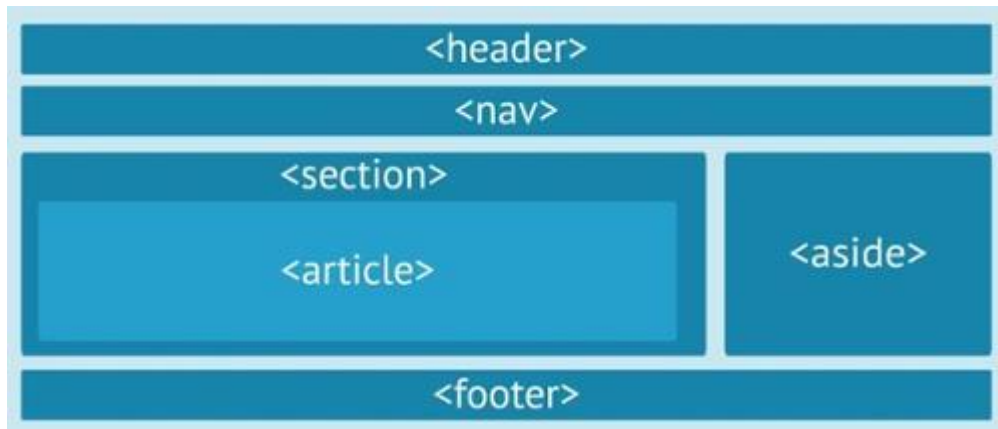
El HTML5 es un estándar que sirve para definir la estructura, el diseño y el contenido de una página web. En realidad, se trata de un código, un lenguaje (HTML) que define los contenidos que forman parte de una página web, como imágenes, texto, vídeos, juegos, marcos, estilo de letra, etc.

- Características principales de HTML5:

- A. Hay nuevas etiquetas semánticas que facilitan la estructura de los documentos HTML y que reemplazarán a etiquetas de versiones anteriores como `<div>`, que ayudaban a identificar cada bloque en las páginas web. Esta es la principal evolución con respecto al HTML4, ya que este tipo de etiquetas permite que la página web sea indexada con mayor exactitud por los buscadores. Estas nuevas etiquetas ayudan también a definir mejor la estructura y los componentes de la web y brindan información sobre su contenido y relevancia. El HTML5 diferencia entre un artículo, sección, vídeo, encabezado, etc.
- B. Existe la posibilidad de incluir audio y vídeo a través de las etiquetas multimedia.
- C. Se integran mejor los gráficos vectoriales escalables (SVG) en lugar de los genéricos que se incluían hasta ahora, como `<object>`, además de incluir nuevos elementos que nos permiten dibujar en él a través de la etiqueta `<canvas>`.
- D. Se redefinen, modifican o estandarizan ciertos elementos que se repiten en la mayoría de las páginas web.

- E. Se pueden incluir fórmulas matemáticas a través de MathML.

**Figura 13. HTML5**



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

- **Node.js:**

Node JS es un entorno de ejecución open source orientado a eventos y de un solo hilo, que además es de código abierto y multiplataforma. Funciona gracias a un motor de ejecución de Javascript originalmente creado para usarse en Google Chrome como servicio web.

**Características principales:**

- No es un lenguaje de programación, sino un entorno para trabajar con lenguajes de programación como JavaScript.
- Node JS está escrito en C, C++ y JavaScript, que son tipos de lenguajes de programación bastante conocidos y fáciles de usar.



- Su sistema monohilo es escalable ya que permite manejar un gran número de conexiones a la vez sin que se afecte el rendimiento del servidor.
- Este entorno es muy veloz gracias a su sistema de hilos sin bloqueo.
- Node JS tiene a su disposición un amplio conjunto de paquetes de código abierto conocido como NPM; este es un ecosistema de más de 1 millón de paquetes.
- Al ser multiplataforma puede ser utilizado en diferentes sistemas operativos como Windows, Linux o Mac.

### **Funcionamiento:**

Node JS se caracteriza por tener un modelo de funcionamiento asíncrono y orientado a eventos y a bucle de eventos, pudiendo manejar múltiples clientes a la vez. Para entender cómo esto se traduce, hay que compararlo con el método de funcionamiento de sistemas multihilo en lenguajes como Javascript.

Un modelo síncrono puede ejecutar varias tareas a la vez, creando un hilo o subproceso para cada una de ellas. Es decir, cada una se ejecuta por separado y se cierra cuando la solicitud ha tenido respuesta.

El problema de este modelo es que mientras más solicitudes se gestionan más subprocesos o hilos se crean, lo que hace mayor consumo de RAM.

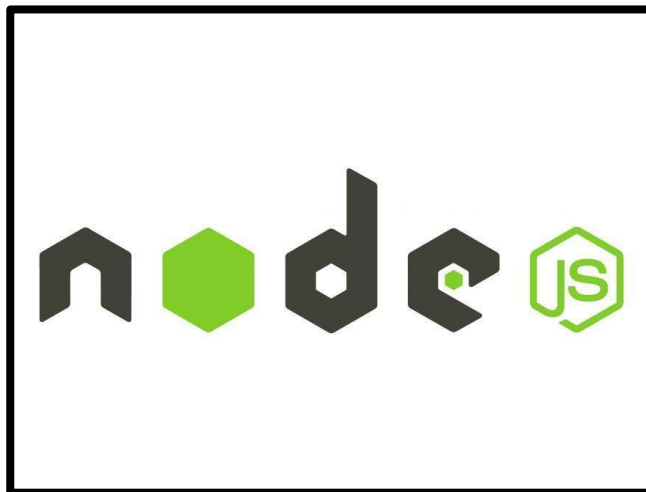
Node JS también es capaz de atender un sinnúmero de solicitudes, pero con la particularidad de tener un pool de hilos limitado.

Esto gracias a su sistema de loop o bucle, que a fines prácticos lo que hace es esperar indefinidamente las peticiones al servidor. En este bucle de eventos se crea una especie de cola con todas las peticiones que van llegando y se procesan todas a la vez.

No tiene que completarse la tarea 1 para que se ejecute la tarea 2, sino que todas se procesan de forma paralela en un proceso que no termina.

Esto permite un sistema de trabajo más eficiente, rápido y con menos gasto de recursos. Además, ya veremos que gracias a este método Node JS es especialmente ventajoso para ciertos tipos de aplicaciones en entornos de servidor. Se puede usar para código tanto back end como front end.

**Figura 14. NodeJS**



**Nota. Logo de nodejs**

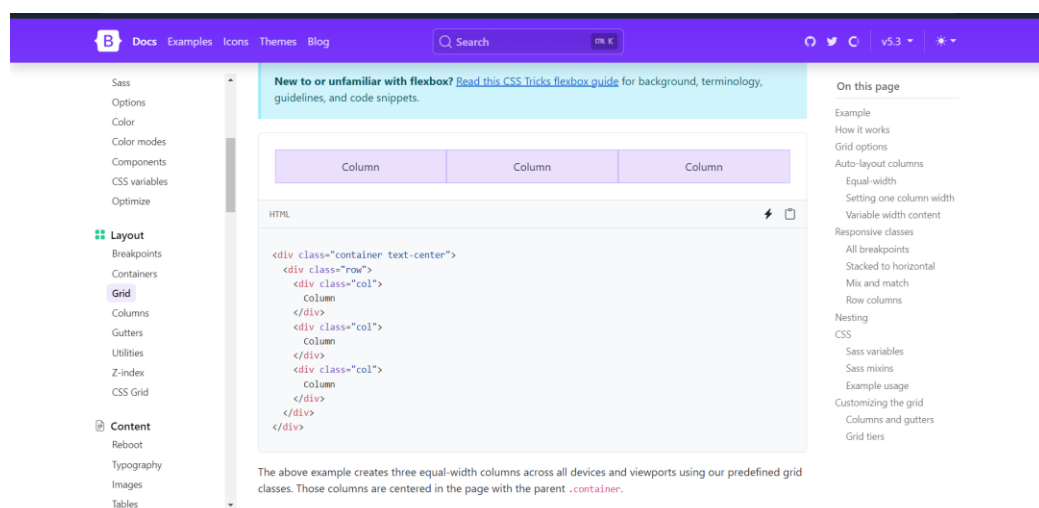
- **Bootstrap 5:**

Bootstrap es un framework front-end que proporciona estructuras de CSS para la creación de sitios web y aplicaciones responsivas de forma rápida y simple. Además, puede lidiar con sitios web de desktop y páginas de dispositivos móviles de la misma forma.

Originalmente, Bootstrap fue desarrollado para Twitter por un grupo de desarrolladores liderados por Mark Otto y Jacob Thornton. Tornandose una de las estructuras de front-end y proyectos de código abierto más populares del mundo.

Antes de ser una estructura de código fuente abierto, Bootstrap era conocido como Twitter Blueprint. Después de algunos meses de desarrollo, Twitter realizó su primer Hack Week: el proyecto ganó una gran popularidad cuando desarrolladores de todos los niveles de habilidad comenzaron a utilizar el framework sin ninguna orientación externa. Tras el evento, ello sirvió como guía de estilo para el desarrollo de herramientas internas en la empresa por más de un año antes de su lanzamiento tornarse publico.

**Figura 15. Bootstrap 5**



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

- **React:**

ReactJS es una de las librerías más populares de JavaScript para el desarrollo de aplicaciones móviles y web. Creada por Facebook, React contiene una colección de fragmentos de código JavaScript reutilizables utilizados para crear interfaces de usuario (UI) llamadas componentes.

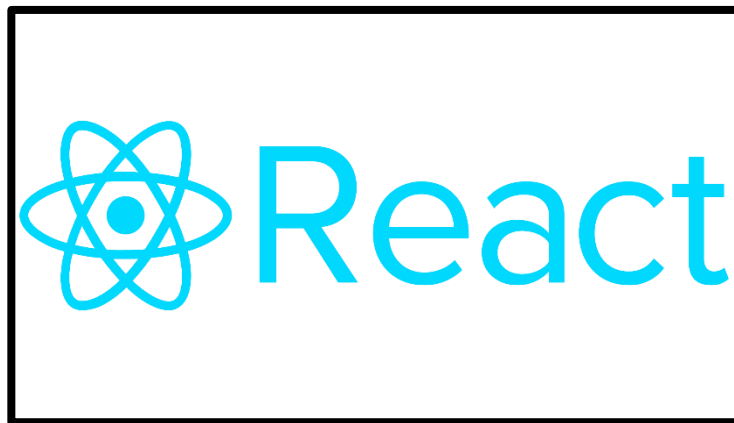
Es importante señalar que ReactJS no es un framework de JavaScript. Esto porque sólo es responsable de renderizar los componentes de la capa de vista de una aplicación. React es una alternativa a frameworks como Angular y Vue, que permiten crear funciones complejas.

### **Ventajas de React:**

- **Fácil de usar:** Los desarrolladores con conocimientos de JavaScript pueden aprender a utilizar React en muy poco tiempo, ya que se basa en JavaScript plano y en un enfoque basado en componentes. Es posible empezar a desarrollar aplicaciones basadas en la web con React tras solo un par de días de estudio.
- **Admite componentes Java reutilizables:** React te permite reutilizar componentes que han sido desarrollados en otras aplicaciones. Como React JS es de código abierto, es posible pre-construir componentes, lo que reduce el tiempo de desarrollo de aplicaciones web complejas.
- **Facilita la redacción de los componentes:** Gracias a la integración de JSX, es más fácil escribir componentes de React: los usuarios pueden crear objetos de JavaScript combinados con tipografía y etiquetas HTML. JSX también

simplifica la renderización de múltiples funciones, lo que mantiene el código sencillo sin reducir las capacidades de la aplicación. Aunque JSX no es la extensión sintáctica más popular, ha demostrado su eficacia en el desarrollo de componentes especiales y aplicaciones dinámicas.

**Figura 16. React**



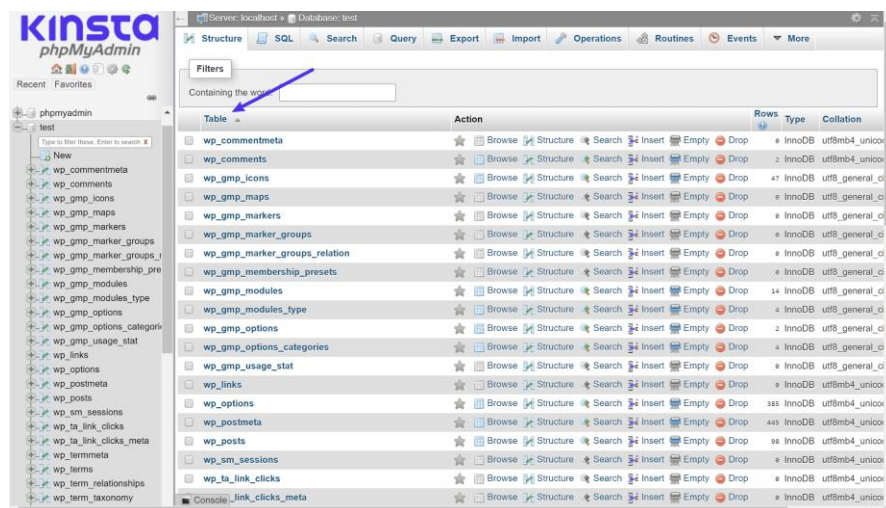
**Nota. Logo de React**

#### 2.1.6. Gestor de base de datos:

##### **MySQL:**

MySQL es la base de datos de código abierto más popular del mercado. Según DB-Engines, MySQL se clasifica como la segunda base de datos más popular, detrás de Oracle Database. MySQL potencia muchas de las aplicaciones más accesibles, como Facebook, Twitter, Netflix, Uber, Airbnb, Shopify y Booking.com.

**Figura 17. MySQL**



### Nota. Diseño de base datos con MySQL

Ventajas principales: MySQL es rápido, confiable, ampliable y fácil de utilizar. Originalmente fue desarrollado para manejar bases de datos grandes rápidamente y se ha utilizado en entornos de producción altamente exigentes durante muchos años.

Las principales ventajas de MySQL incluyen

- **Facilidad de uso:** los desarrolladores pueden instalar MySQL en minutos y la base de datos es fácil de gestionar.
- **Confiabilidad:** MySQL es una de las bases de datos más maduras y utilizadas. Lleva más de 25 años probándose en una amplia variedad de casos, incluso en muchas de las mayores empresas del mundo. Las organizaciones utilizan MySQL para ejecutar aplicaciones clave para el negocio debido a su confiabilidad.
- **Escalabilidad:** MySQL se amplía para satisfacer las demandas de las aplicaciones más accesibles. La arquitectura de replicación nativa

de MySQL permite a organizaciones como Facebook escalar aplicaciones para admitir miles de millones de usuarios.

- **Desempeño:** HeatWave MySQL es más rápido y menos costoso que otros servicios de bases de datos, como lo demuestran varias referencias estándar del sector, como TPC-H, TPC-DS y CH-benCHmark.

#### 2.1.7. Herramientas adicionales:

- **Balsamiq:**

Una de las mejores herramientas para crear prototipos, bocetos o wireframes es Balsamiq Mockups. Este programa tiene varias ventajas:

- Es un programa de escritorio, es decir, solo tienes que registrarte para poder empezar a utilizarlo sin ningún tipo de descarga.
- Programado en Flex y AIR por lo que crea muy rápido Wireframes.
- Interfaz fácil de usar: como se ha creado con AIR es instalable tanto en Windows como Linux y Mac OS X.
- Te permite escoger entre un montón de objetos prediseñados como: barras de estado, menús, barras de progreso, etc. Además, te permite exportar el diseño que realices en PNG, PDF e incluso al portapapeles.
- Permite incrustar tus diseños en tu página web o en informes de errores. Puedes compartir tus diseños a través de un sistema muy parecido a Dropbox. Balsamiq funciona como cualquier aplicación normal, te permite arrastrar, soltar, guardar mockups en un archivo, copiar y pegar, deshacer, etc.

**Figura 18. Balsamiq**



Fuente: <https://balsamiq.com/>

- **Bizagi:**

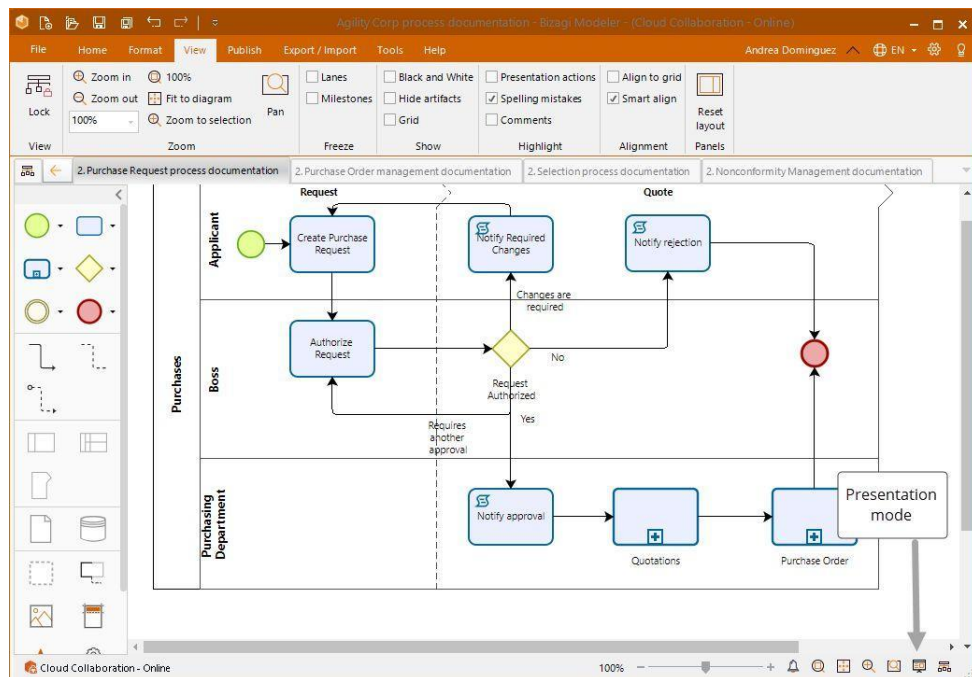
Es un modelador de procesos que permite representar de forma esquemática todas las actividades y decisiones que se toman en el negocio. Con una interfaz que recuerda a Microsoft Office, BizAgi Process Modeler cumple con el estándar BPMN (Business Process Management Notation).

Una vez hayas finalizado la representación del flujo de trabajo, la aplicación puede documentar los proyectos de forma automática a partir de la información que se haya incluido en los esquemas.

Con el Modelador Bizagi, podrás hacer diagramas y documentar sus procesos de la manera más eficiente y buscando fomentar la colaboración en tu organización. El primer paso que tendrás que dar para mejorar la eficiencia operacional de una organización, consistirá en definir claramente los procesos. El Modelador de Procesos BPMN Bizagi, te permitirá diagramar y documentar sus procesos de la manera más rápida y fácil posible.



**Figura 19. Bizagi**



Fuente: Aplicación de bizagi

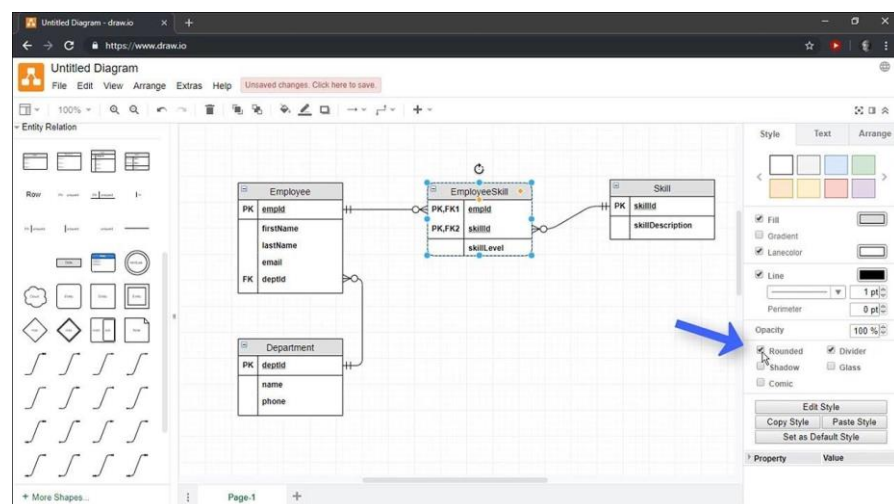
- **draw.io:**

Draw.io es una herramienta de creación y edición de diagramas libre que permite la integración con diversas plataformas. El software consiste en una aplicación web realizada mayoritariamente en JavaScript y licenciada con Apache v2, la cual funciona en una amplia gama de navegadores y permite la creación de diagramas, contando con modelos para diversos tipos como pueden ser diagramas UML, esquemas de red, flujogramas, diagramas de ingeniería y electrónica, mapas conceptuales, diagramas de Venn... También permite crear colecciones de diagramas e imágenes personalizados para utilizar en los diagrama.

Se trata de una aplicación sencilla de usar mediante la cuál se pueden dibujar diagramas profesionales y que cumple

con la mayor parte de las necesidades que puede tener cualquier empresa, profesional o administración a la hora de diseñar diagramas de cualquier tipo. Esta herramienta está disponible para trabajar en línea pudiendo guardar los diagramas en Google Drive, OneDrive o en local. Adicionalmente, la aplicación web también está disponible para el trabajo sin conexión mediante la descarga previa de la misma para su uso en diversas plataformas como Chrome OS, GNU/Linux, Mac OS y Windows.

**Figura 20. Draw.io**



Fuente: <https://app.diagrams.net/>

- **Canva:**

Canva es una web de diseño gráfico y composición de imágenes para la comunicación fundada en 2012, y que ofrece herramientas online para crear tus propios diseños, tanto si son para ocio como si son profesionales. Su método es el de ofrecer un servicio freemium, que puedes utilizar de forma gratuita, pero con la alternativa de pagar para obtener opciones avanzadas.

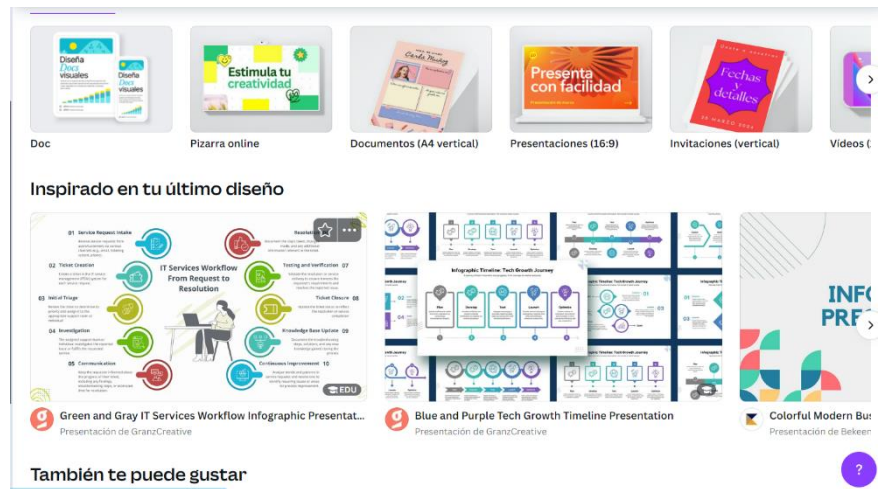
- **Características:**

1. **Equipos y colaboración:** Canva te permite crear equipos para que puedas trabajar en diseños juntos en

tiempo real. Puedes tener hasta 20 equipos a la vez, incluso con una cuenta gratuita de Canva. La comunicación con los miembros del equipo es fluida, ya que puede publicar comentarios, etiquetar miembros específicos y dejar comentarios y sugerencias.

2. **Plantillas:** La gran ventaja de Canva es que nos permite utilizar plantillas ya hechas por lo que no tenemos que saber de diseño ni tener ningún programa especializado para crearlas desde cero. Hay plantillas para prácticamente todo lo que necesites. Más de 250.000 plantillas gratis ya preparadas para que cambies el texto o la imagen. Ideas de negocio online marketing digital
3. **Editor y creador de videos:** El creador y editor de videos son algunos de los aspectos más destacados de Canva. Esta herramienta promete videos de alta calidad, incluso si no tienes experiencia en el área. Con su sencilla herramienta de arrastrar y soltar, puedes crear videos cortos utilizando la biblioteca del software de plantillas de video prefabricadas. También tienes acceso a videos de archivo gratuitos, arte GIF animado y pistas de audio de alta calidad.
4. **Recorte de fotografías:** Con la función de recorte de fotos de Canva, puedes ajustar el enfoque o alterar la forma de las imágenes para que se ajusten al resto de tu diseño. La herramienta de recorte de Canva es inteligente, ya que alinea automáticamente las imágenes con los otros elementos de tu diseño después de recortarlos.

**Figura 21. Canva**



Fuente: <https://www.canva.com/>

## 2.2. Marco teórico de las fórmulas 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.

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

### 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.

- Calidad de los pedidos generados:

Este indicador mide el porcentaje de pedidos generados correctamente, evaluando si cumplen con los criterios de precisión y coherencia en la información, como la cantidad solicitada, especificaciones del producto, y detalles de entrega. La calidad de los pedidos es fundamental para evitar errores en la preparación y distribución, mejorando la satisfacción del cliente.

Número de pedidos generados correctamente/ (Número total de pedidos generados) ×100

- Entregas perfectamente recibidas:

Este indicador evalúa la precisión y satisfacción de las entregas desde el punto de vista del cliente. Una entrega perfectamente recibida significa que el pedido llegó a tiempo, en la cantidad correcta, sin daños, y con toda la documentación necesaria. Esto se calcula en función del número total de entregas y las que fueron recibidas sin problemas reportados.

(Número de pedidos rechazados) / (Número total de entregas) ×100

- Índice de Devoluciones por Defectos

Este indicador mide el porcentaje de aretes devueltos por defectos en comparación con el total de aretes vendidos. Ayuda a identificar y evaluar problemas de calidad en los productos, permitiendo detectar patrones de defectos y tomar acciones para mejorar la calidad y reducir las devoluciones.

Número de Aretes Devueltos por Defectos/ (Total de aretes vendidos) ×100

### **Indicadores de Utilización:**

Los indicadores de utilización miden el grado en que se están utilizando los recursos disponibles en una organización, como maquinaria, equipos y mano de obra. Estos indicadores ayudan a evaluar si los recursos se están empleando de manera eficiente y efectiva, y permiten identificar áreas donde se puede mejorar la utilización para aumentar la capacidad y reducir costos.

- Índice de rotación de aretes:

Este indicador mide cuántas veces el inventario de aretes se ha renovado en un periodo específico. Es útil para gestionar la reposición de productos, ya que permite entender la velocidad con la que se venden los aretes y ayuda a planificar el inventario para evitar tanto excesos como faltantes.

$$(\text{Costo de ventas})/(\text{Promedio de inventario de aretes})$$

- Tasa de disponibilidad de modelos

Mide la disponibilidad de los diferentes modelos de aretes, lo cual es clave para asegurar variedad y satisfacer la demanda.

$$(\text{Modelo de aretes disponibles})/(\text{Total de modelos de aretes}) \times 100$$

### **Indicadores de Productividad:**

Los indicadores de productividad evalúan la relación entre la producción obtenida y los recursos utilizados para alcanzarla. Se utilizan para medir la eficiencia operativa y se expresan comúnmente como la cantidad de output (producción) por unidad de input (recursos). Estos indicadores son cruciales para identificar mejoras en los procesos productivos y maximizar el rendimiento de los recursos.

- Índice de popularidad

Mide la demanda de cada diseño, permitiendo ajustar la oferta y producción a los productos con mayor aceptación.

$$(\text{Ventas de un modelo de arete})/(\text{Total de ventas de aretes}) \times 100$$

- Rotación de inventario

Mide cuántas veces se ha vendido y reemplazado el inventario en un periodo determinado.

$$\text{Rotación de inventario} = (\text{Costo de bienes vendidos})/(\text{Inventario promedio})$$

- Tiempo de reposición

Mide el tiempo promedio necesario para reponer un producto en inventario, optimizando la eficiencia operativa.

$$(\text{Tiempo total de reposición})/(\text{Número de reposiciones})$$

- Cumplimiento de plazos de reposición

Mide el porcentaje de reposiciones que se cumplen dentro del tiempo establecido, asegurando un flujo adecuado de inventario.

$$(\text{Número de reposiciones a tiempo})/(\text{Total de reposiciones}) \times 100$$

Tasa de productos obsoletos

Este indicador mide el porcentaje de aretes que han permanecido en inventario por más de seis meses sin venderse. Es fundamental para identificar productos que pueden estar generando pérdidas por devaluación

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

#### 3.1. Prototipos



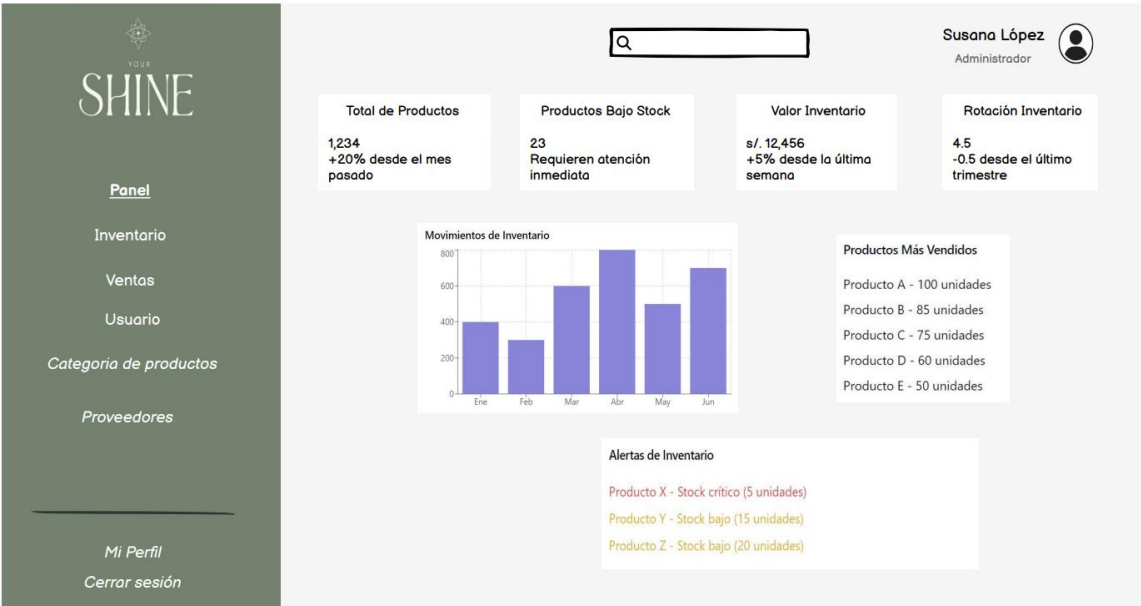
**Figura 22. Login**



The login form is centered on a light cream background with a dark olive green border. At the top is a geometric logo consisting of a star-like shape with a central cross. Below the logo, the text "YOUR SHINE" is displayed in a serif font, with "YOUR" in a smaller size above "SHINE". The form includes two input fields: "Usuario:" followed by a rectangular text box, and "Contraseña:" followed by a rectangular text box. Below these is a reCAPTCHA checkbox labeled "I'm not a robot" with a small square icon, and a reCAPTCHA logo with the text "reCAPTCHA Privacy - Terms". A dark green button with the text "Acceder" in white is positioned below the reCAPTCHA. At the bottom, a link "¿Olvidaste tu contraseña?" is displayed in a smaller, lighter green font.

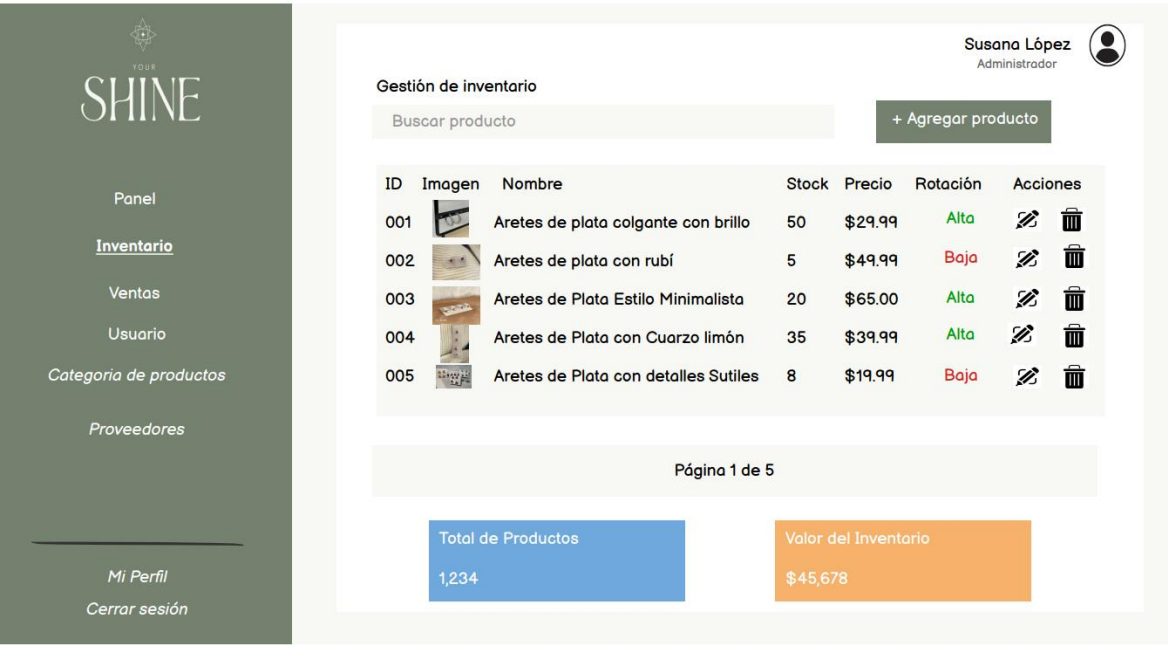
Fuente: Elaboración propia

Figura 23. Dashboard



Fuente: Elaboración propia

Figura 24. Inventario

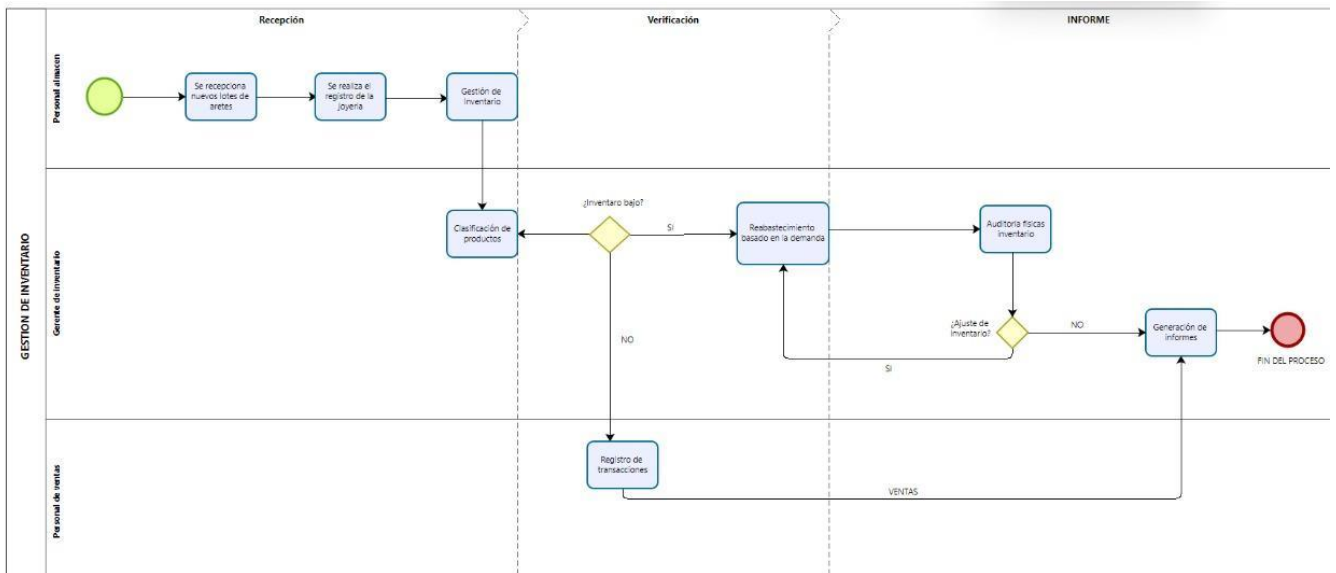


Fuente: Elaboración propia

3.2. Diagramas

3.2.1. Modelo de negocios

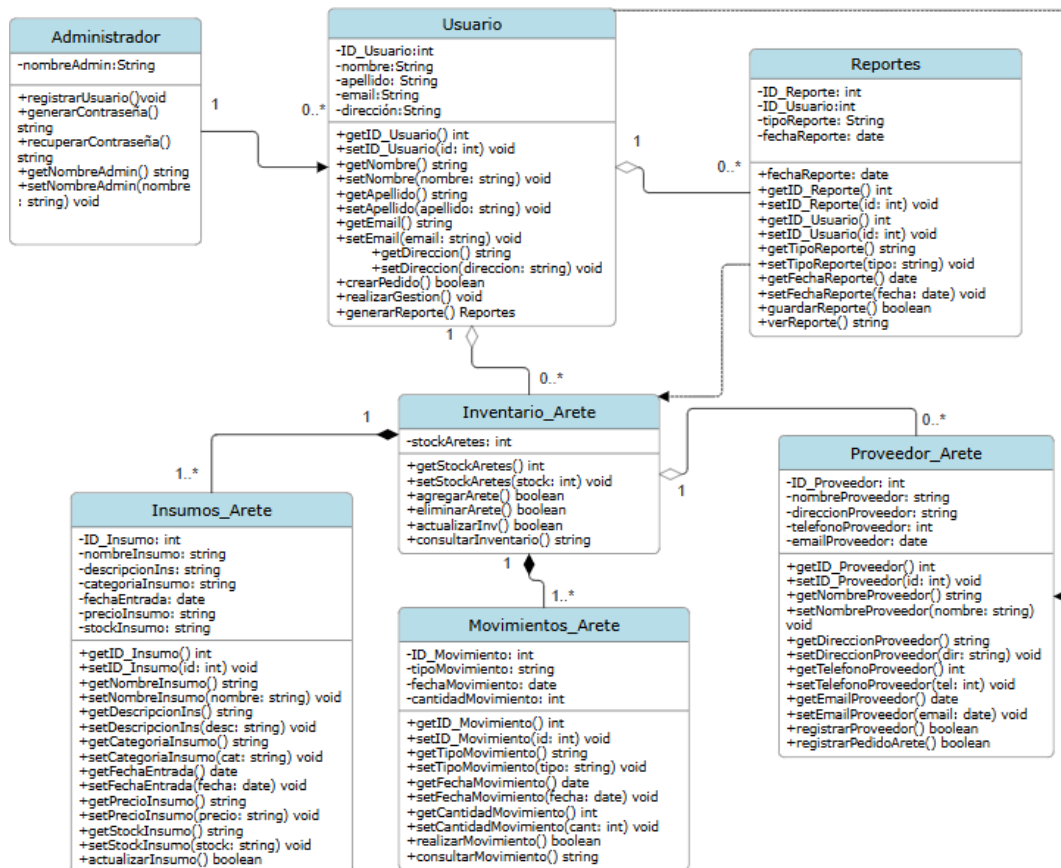
Figura 24. Modelo de negocios



Fuente: Elaboración propia

### 3.2.2. Diagrama clases

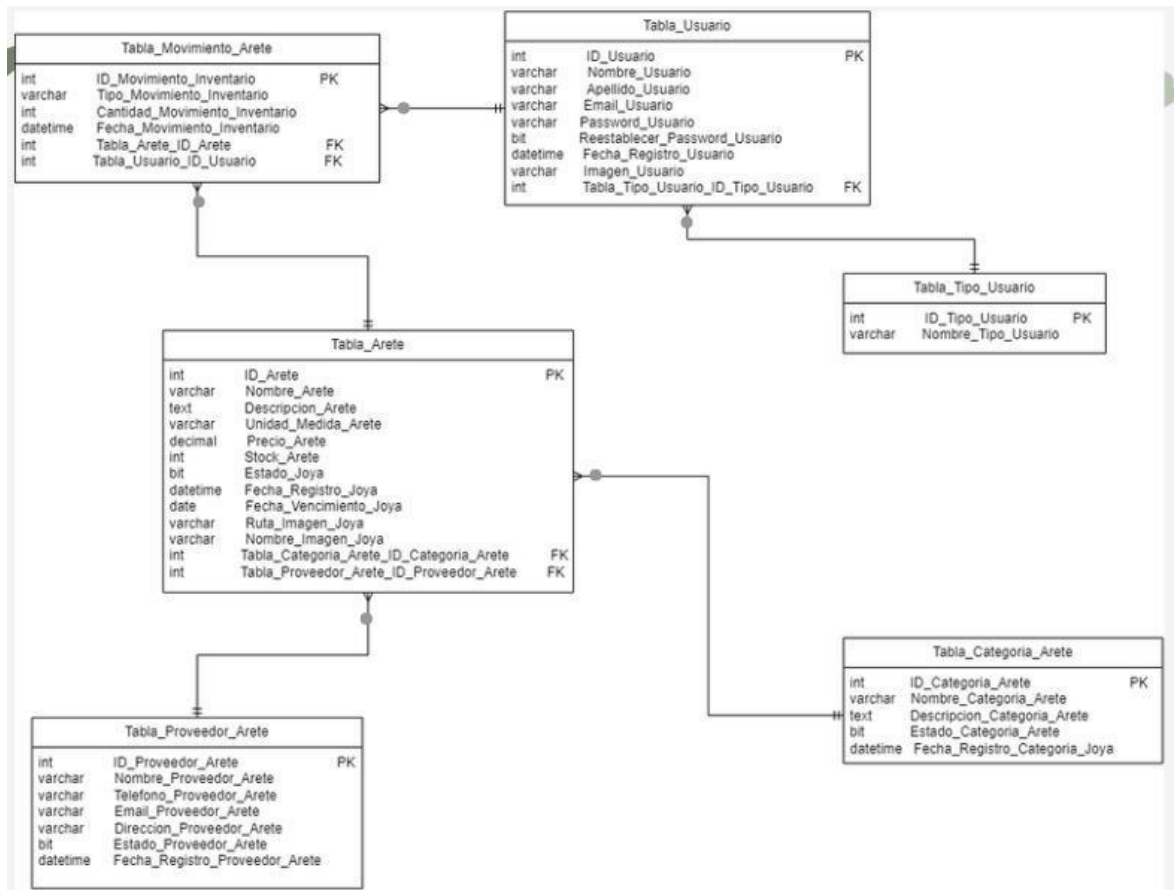
Figura 25. Diagrama de clases



Fuente: Elaboración propia

### 3.2.3. Diagrama de entidad – relación

Figura 26. Diagrama de entidad relación



Fuente: Elaboración propia

### 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
FK	Imagen_Usuario	VARCHAR	255	Ruta de la imagen de perfil del usuario
	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

## 4. CAPÍTULO 4 – RESULTADOS

### 4.1. Resultados de encuesta

#### **Funcionabilidad**

1. ¿El sistema cumple con todos los requisitos funcionales establecidos al inicio del proyecto, tales como la gestión de inventario y la previsión de demanda?

- A. Siempre
- B. Casi siempre
- C. A veces
- D. Raramente
- E. Nunca

2. ¿Las funciones de búsqueda y filtrado dentro del sistema son efectivas?

- A. Muy efectivas
- B. Efectivas
- C. Moderadamente efectivas
- D. Poco efectivas
- E. Nada efectivas

3. ¿El Sistema registra correctamente las entradas y salidas de los productos?

- A. Totalmente de acuerdo
- B. Parcialmente de acuerdo
- C. Neutral
- D. Parcialmente en desacuerdo
- F. Totalmente en desacuerdo

#### **Usabilidad**

5. ¿La interfaz del sistema es intuitiva y fácil de usar para los usuarios encargados de gestionar el inventario?

- A. Totalmente de acuerdo

- B. De acuerdo
  - C. Neutral
  - D. En desacuerdo
  - E. Totalmente en desacuerdo
6. ¿Los reportes de ventas son fáciles de generar y comprende dentro del sistema?
- A. Totalmente de acuerdo
  - B. De acuerdo
  - C. Neutral
  - D. En desacuerdo
  - E. Totalmente en desacuerdo
7. ¿La búsqueda de productos específicos es rápida y precisa (por código, nombre o categoría)?
- A. Totalmente de acuerdo
  - B. De acuerdo
  - C. Neutral
  - D. En desacuerdo
  - E. Totalmente en desacuerdo
8. ¿Las alertas de bajo stock son visualmente prominentes dentro y no puedo pasarlas por alto?
- A. Totalmente visibles
  - B. Muy visibles
  - C. Moderadamente visible
  - D. Poco visible
  - E. Apenas visible
9. ¿Las etiquetas y descripciones en los campos del sistema son claras y comprensibles?
- A. Muy claras
  - B. Claras
  - C. Aceptables
  - D. Poco claras
  - E. Nada claras

10. ¿El rendimiento del sistema ha afectado su productividad de manera negativa?

- A. Nunca
- B. Rara vez
- C. Algunas veces
- D. A menudo
- E. Siempre

11. ¿Los reportes dentro del sistema se generan en un tiempo razonable?

- A. Siempre
- B. Casi siempre
- C. A veces
- D. Raramente
- E. Nunca

12. ¿El sistema procesa las actualizaciones de inventario en menos de 2 segundos?

- A. Siempre
- B. Casi siempre
- C. A veces
- D. Raramente
- E. Nunca

13. ¿Se experimentan caídas o errores frecuentes dentro del sistema?

- A. Siempre
- B. Casi siempre
- C. A veces
- D. Raramente
- E. Nunca

14. ¿El sistema opera con la misma rapidez independientemente de la cantidad de datos almacenados?

- A. Siempre
- B. Casi siempre
- C. A veces
- D. Raramente



E. Nunca

### **Confiabilidad**

15. ¿El sistema garantiza la consistencia y precisión de los datos del inventario después de cada actualización?

A. Totalmente de acuerdo

B. De acuerdo

C. Neutral

D. En desacuerdo

E. Totalmente en desacuerdo

16. ¿El sistema es capaz de detectar y corregir errores de manera automática?

A. Siempre

B. Casi siempre

C. A veces

D. Raramente

E. Nunca

17. ¿El sistema es estable durante largos periodos de uso sin presentar problemas de acceso o caídas?

A. Completamente de acuerdo

B. De acuerdo en su mayoría

C. Neutral

D. En desacuerdo en su mayoría

E. Totalmente en desacuerdo

18. ¿El sistema asegura la integridad y protección de los datos en caso de fallo o interrupción?

A. Siempre

B. Casi siempre

C. A veces

D. Raramente

E. Nunca

19. ¿El sistema cuenta con herramientas de monitoreo que alerten sobre posibles fallos antes de que ocurran?

- A. Siempre
- B. Casi siempre
- C. A veces
- D. Raramente
- E. Nunca

### **Portabilidad**

20. ¿El sistema funciona correctamente en diferentes navegadores web?

- A. Siempre
- B. Casi siempre
- C. A veces
- D. Raramente
- E. Nunca

21. ¿Ha experimentado problemas de compatibilidad al actualizar su sistema operativo?

- A. Sí
- B. No

22. ¿La visualización del sistema se adapta correctamente a diferentes tamaños de pantalla?

- A. Siempre
- B. Casi siempre
- C. A veces
- D. Raramente
- E. Nunca

23. ¿El sistema puede integrarse fácilmente con otros sistemas en diferentes entornos?

- A. Sí
- B. No

24. ¿El sistema mantiene la misma funcionalidad cuando se ejecuta en diferentes entornos con configuraciones predeterminadas?

- A. Siempre
- B. Casi siempre

- C. A veces
- D. Raramente
- E. Nunca

**Seguridad:**

25. ¿La información del inventario está adecuadamente protegida?

- A. Totalmente de acuerdo
- B. De acuerdo
- C. Neutral
- D. En desacuerdo
- E. Totalmente en desacuerdo

26. ¿Solo puedo acceder a las funciones asignadas a mi rol (vendedor, administrador)?

- A. Sí
- B. No

27. ¿Puedo ver claramente qué acciones están permitidas para mi rol actual?

- A. Totalmente de acuerdo
- B. De acuerdo
- C. Neutral
- D. En desacuerdo
- E. Totalmente en desacuerdo

28. ¿El sistema me impide modificar registros que no corresponden a mi nivel de autorización?

- A. Totalmente de acuerdo
- B. De acuerdo
- C. Neutral
- D. En desacuerdo
- E. Totalmente en desacuerdo

29. ¿Los datos de inventario están protegidos contra modificaciones no autorizadas?

- A. Siempre
- B. Casi siempre

- C. A veces
- D. Raramente

#### 4.2. Presupuesto

Presupuesto						
Categoría	Componente	Tipo de recursos	Cantidad	Tipo de unidad	Costo Unitario	Costo total
Mano de obra	Analista	Humano	6	Mes	4,000	24,000
	Desarrollador frontend	Humano	2	Mes	5,000	10,000
	Desarrollador backend	Humano	3	Mes	5,000	15,000
	Diseñador ux/ui	Humano	1	Mes	3,500	3,500
	Documentador	Humano	2	Mes	2,000	4,000
Materiales (Software/Hardware)	Equipo de desarrollo	Tecnológico	3	Unidad	4,000	12,000
	Licencia IDE	Licencia	3	Unidad	500	1,500
	Servidor de desarrollo	Tecnológico	1	Unidad	3,500	3,500
	Licencia Mysql	Licencia	1	Unidad	0	0
Capacitación	Capacitación usuario final	Servicio	1	Sesión	1,500	1,500
	Capacitación Técnica	Servicio	1	Sesión	3,500	3,500
Gastos	Servicios cloud	Servicio	6	Mes	500	300
	Internet	Servicio	6	Mes	180	900
	Energía	Servicio	6	Mes	100	600
Total						80,300

## CONCLUSIONES

- Al desarrollar el sistema de inventario para la joyería de aretes de plata, pudimos observar la importancia de entender las tendencias y preferencias de los clientes. La implementación del módulo de previsión de demanda se convierte en un recurso clave para la toma de decisiones, ya que permite a la empresa anticiparse a las necesidades del mercado. Gracias a este módulo, logramos optimizar la planificación del inventario, reducir la acumulación de productos de baja demanda y evitar la devaluación del inventario. Integrar un análisis de tendencias en el sistema de inventario no solo mejora la rentabilidad, sino que también contribuye a una mejor gestión de recursos, lo cual es fundamental para la sostenibilidad del negocio en el sector de la joyería.
- Al implementar el módulo para mantener la información actualizada sobre las necesidades reales del mercado y los productos que se están vendiendo, se pudo comprobar que una visión clara y actualizada de la demanda permite tomar decisiones más estratégicas sobre las compras. Este módulo nos ayuda a evitar la acumulación de productos de baja rotación o sin demanda, lo cual optimiza los recursos y reduce el riesgo de inmovilización de capital en inventario obsoleto. Mantener una visión real del mercado antes de realizar compras es crucial para la eficiencia operativa y la rentabilidad.
- En cuanto al módulo de seguimiento de rotación de productos, su desarrollo fue fundamental para identificar rápidamente los productos de alta y baja rotación. Este seguimiento detallado permite ajustar las estrategias de venta y mejorar la disponibilidad de productos populares, mientras se gestionan con precisión los artículos menos demandados. En resumen, contar con un monitoreo constante de la rotación de productos brinda una ventaja competitiva, ya que permite tomar decisiones rápidas y

efectivas que mejoran la eficiencia del inventario y contribuyen a satisfacer las expectativas del cliente.

## RECOMENDACIONES

- Actualmente, el sistema se enfoca en la gestión de inventario, pero integrar una función de control y gestión de compras centralizada ayudaría a automatizar la adquisición de productos en función de la demanda, evitando excesos de stock o falta de productos.
- Conectar el sistema con plataformas de venta en línea y redes sociales para obtener datos en tiempo real sobre las preferencias y comportamientos de los clientes. Esta integración permitiría un análisis más completo y ajustado al comportamiento actual de los consumidores.
- Agregar algoritmos de inteligencia artificial y machine learning para analizar grandes volúmenes de datos y predecir patrones de compra futuros, lo que permitiría optimizar aún más las previsiones de demanda y ajustar el inventario de manera proactiva.
- Implementar un módulo de recomendaciones que sugiera productos a los clientes en función de sus compras anteriores o tendencias de mercado actuales. Este sistema podría personalizar la experiencia del cliente, aumentando la posibilidad de venta cruzada y potenciando la fidelización de los clientes al ofrecerles productos alineados con sus gustos.

## 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>

Mancuzo, G. (2023). ▷ 5 fases de la metodología SCRum [5 etapas y 16 procesos].

<https://blog.comparasoftware.com/fases-metodologia-scrum/>

Universidad Europea. (2023, 17 noviembre). ¿Qué es un modelo de entidad-relación?

<https://universidadeuropea.com/blog/modelo-entidad-relacion/>

López, Ó. (2023, 10 diciembre). ¿Qué es un Patrón de Diseño? Formiux.

<https://formiux.com/que-es-un-patron-de-diseno/>



Ida, E. (2022, 31 marzo). ¿Cuál es la diferencia entre un modal, un pop up, un pop over y un light box? | IDA Chile. Blog IDA Chile | Estrategia Para el Éxito de Tu Negocio. <https://blog.ida.cl/disenio/ldiferencia-modal-pop-up-pop-over-light-box/>

Help. (s. f.). Acerca de la barra de navegación

<https://help.autodesk.com/view/ACD/2025/ESP/?guid=GUID-17225A0F-7A50-41B9-8404-7D415518DEFB>

Pires, R. (2022, 28 abril). Descubre qué es un dashboard y qué información debe contener. Rock Content - ES. <https://rockcontent.com/es/blog/dashboard/>

¿Qué es un CAPTCHA? - Ayuda de Administrador de Google Workspace. (s. f.). <https://support.google.com/a/answer/1217728?hl=es>

Del Prado, J. A. (2022, 6 septiembre). Qué es y cómo diseñar una Card. UXABLES | Blog.

<https://www.uxables.com/disenio-ux-ui/que-es-y-como-disenar-una-card/>

Aguilar, J. M. (s. f.). ¿Qué es el patrón MVC en programación y por qué es útil? - campusMVP.es. campusMVP.es.

<https://www.campusmvp.es/recursos/post/que-es-el-patron-mvc-en-programacion-y-por-que-es-util.aspx>

DAO [cursohibernate]. (s. f.).

<http://www.cursohibernate.es/doku.php?id=patrones:dao>

Data Access Object (DAO). (s. f.).

<https://reactiveprogramming.io/blog/es/patrones-arquitectonicos/dao>

Martínez, G. A. (2024, 1 abril). Qué es el Patrón de diseño DTO y por qué es imprescindible que lo apliques. Brave Developer.

<https://bravedeveloper.com/2021/11/20/que-es-el-patron-arquitectonico-dto-y-por-que-es-imprescindible-que-lo-apliques/>

JavaScript | MDN. (2023, 24 julio). MDN Web Docs. <https://developer.mozilla.org/es/docs/Web/JavaScript>

Zubikarai, S. (2021, 15 marzo). Ventajas y desventajas de JavaScript. freeCodeCamp.org.

<https://www.freecodecamp.org/espanol/news/ventajas-y-desventajas-de-javascript/>

\_ESIC Business & Marketing School. (s. f.). HTML5: qué es, características y cómo funciona. ESIC.

<https://www.esic.edu/rethink/tecnologia/html5-que-es-caracteristicas-y-como-funciona-c#:~:text=El%20HTML5%20es%20un%20est%C3%A1ndar.%2C%20estilo%20de%20letra%2C%20etc.>

Jesús. (2022, 3 septiembre). ¿Qué es Node JS? » Dongee. Tutoriales Dongee.

[https://www.dongee.com/tutoriales/que-es-node-js/?tm=tt&ap=gads&aaid=adaFnZvkSdpvp&gad\\_source=1&gclid=Cj0KCQjwsoe5BhDiARIsAOXVoUsBk744s8F5fvWZXQA4c5FpZCg8Ndg9C4KIKNV-LfyrynMav\\_NJQaAi3nEALw\\_wcB](https://www.dongee.com/tutoriales/que-es-node-js/?tm=tt&ap=gads&aaid=adaFnZvkSdpvp&gad_source=1&gclid=Cj0KCQjwsoe5BhDiARIsAOXVoUsBk744s8F5fvWZXQA4c5FpZCg8Ndg9C4KIKNV-LfyrynMav_NJQaAi3nEALw_wcB)

Latam, A. (2022, 29 agosto). Bootstrap ¿Qué es, cómo y cuando utilizar? Alura.

[https://www.aluracursos.com/blog/bootstrap-que-es-como-y-cuando-utilizar?utm\\_source=google&utm\\_medium=cpc&utm\\_campaign=AL\\_PR\\_F\\_Search\\_Pmax&gad\\_source=1&gclid=Cj0KCQjwsoe5BhDiARIsAOXVoUuqcjer55HTpbQokRcBjYTQrdZpYBV--RLCLmq1dcAopYLoe-SXGNsaAq-dEALw\\_wcB](https://www.aluracursos.com/blog/bootstrap-que-es-como-y-cuando-utilizar?utm_source=google&utm_medium=cpc&utm_campaign=AL_PR_F_Search_Pmax&gad_source=1&gclid=Cj0KCQjwsoe5BhDiARIsAOXVoUuqcjer55HTpbQokRcBjYTQrdZpYBV--RLCLmq1dcAopYLoe-SXGNsaAq-dEALw_wcB)

A, D., & A, D. (2023, 29 junio). Qué es React: definición, características y funcionamiento. Tutoriales Hostinger.

<https://www.hostinger.es/tutoriales/que-es-react>

¿Qué es MySQL? (s. f.). Oracle Perú.  
<https://www.oracle.com/pe/mysql/what-is-mysql/>

ISDI España, La Escuela de Negocios de la Era Digital. (2024, 9 octubre).  
Balsamiq la herramienta para hacer prototipos de proyectos. ISDI España.  
<https://www.isdi.education/es/blog/balsamiq-herramienta-para-realizar-prototipos-de-tus-proyectos>

BIZAGI MODELER – Edutic. (s. f.).  
<https://edutic.up.edu.pe/catalogo-software/bizagi-modeler/>

Draw.io – Mancomún. (s. f.).  
<https://mancomun.gal/es/solucion-tic/draw-io/>

Fernández, Y. (2023, 9 junio). Qué es Canva, cómo funciona y cómo usarlo para crear un diseño. Xataka.  
<https://www.xataka.com/basics/que-canva-como funciona-como-usarlo-para-crear-diseno>

Características de Canva. (s. f.). Issuu.  
[https://issuu.com/arielagarcia/docs/mimannual\\_ariela\\_garcia/s/16853956](https://issuu.com/arielagarcia/docs/mimannual_ariela_garcia/s/16853956)

Osaín, C. L. (s/f). Indicadores de Gestión. Ucipfg.com. Recuperado el 3 de octubre de 2024, de  
<https://www.ucipfg.com/Repositorio/MLGA/MLGA-03/semana2/indicadores-de-gestion.pdf>

## ANEXOS

- Anexo 1. Lean Canvas



- Anexo 2. Project Charter

PROJECT CHARTER			
Nombre del proyecto	Sistema de control de inventario de la joyeria yourshine.pe		
Fecha de inicio	15 de Agosto de 2024	Fecha de Fin	5 de diciembre de 2024
Gerente del proyecto	Fabiana Geronimo	Patrocinador del proyecto	Yourshine.pe
Propósito del proyecto			
Desarrollar un sistema de control de inventario para reducir la acumulación de productos de baja demanda, mejorar la eficiencia operativa y optimizar la rotación de stock basado en tendencias de mercado y ventas.			
Objetivos del proyecto		Alcance del proyecto	
1. Implementar un módulo de análisis de tendencias y preferencias 2. Mantener información actualizada del mercado y ventas 3. Desarrollar un módulo de seguimiento de rotación de productos		1. Ingreso de la cantidad de inventario 2. Clasificación de productos	
Supuestos		Riesgos y problemas	
1. Se espera que los empleados, especialmente aquellos en inventario y ventas, se adapten al nuevo sistema y participen activamente en su uso. 2. Se supone que la empresa proporcionará la capacitación necesaria a los empleados, asegurando que entiendan y manejen correctamente el sistema. 3. Se espera que haya soporte técnico disponible para resolver problemas técnicos durante y después de la implementación.		1. La falta de datos precisos limita la previsión de demanda. 2. Retrasos en la formación del personal pueden afectar la adopción del sistema. 3. El sistema podría necesitar actualizaciones frecuentes debido a fluctuaciones en la demanda.	
Presupuesto estimado	El presupuesto estimado de este proyecto es de s/. 80,300		
Hitos			
Hito	Fecha prevista de finalizacion		
Documentación de proyecto	1/11/2024		
Prototipo de proyecto	10/09/2024		
1era fase de desarrollo	2/11/2024		
2da fase de desarrollo	28/11/2024		
Testeo del programa	30/11/2024		
Equipo del proyecto		Partes interesadas	
Product Owner	Fabiana Geronimo	Patrocinadores	
Project Manager	Fabiana Díaz	Equipo de proyecto	
Desarrollador Líder	Daniela Yarihuaman	Usuario final	
Diseñador UX/UI	Fabiana Geronimo Fabiana Diaz Daniela Yarihuaman		

- Anexo 3. Diagrama Gantt

## Sistema de inventario

Joyería YourShine  
Grupo 4

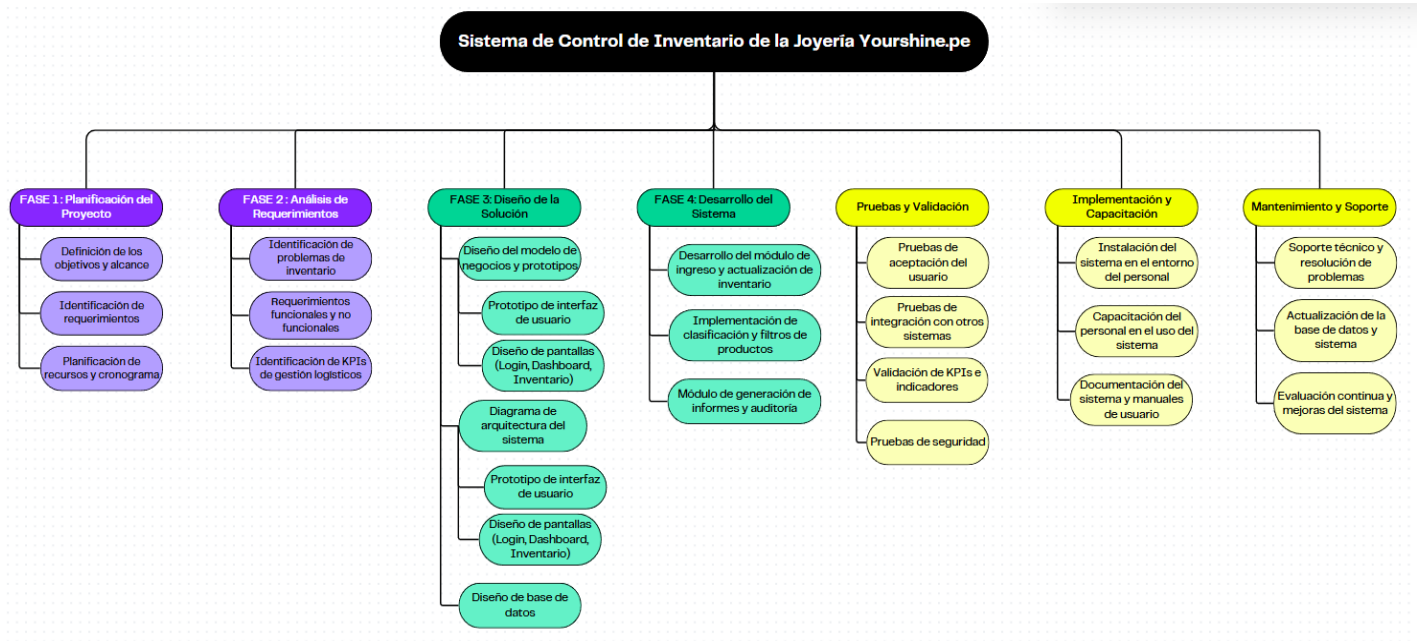
### Inicio del proyecto

jue, 2024-08-15

Semana para mostra

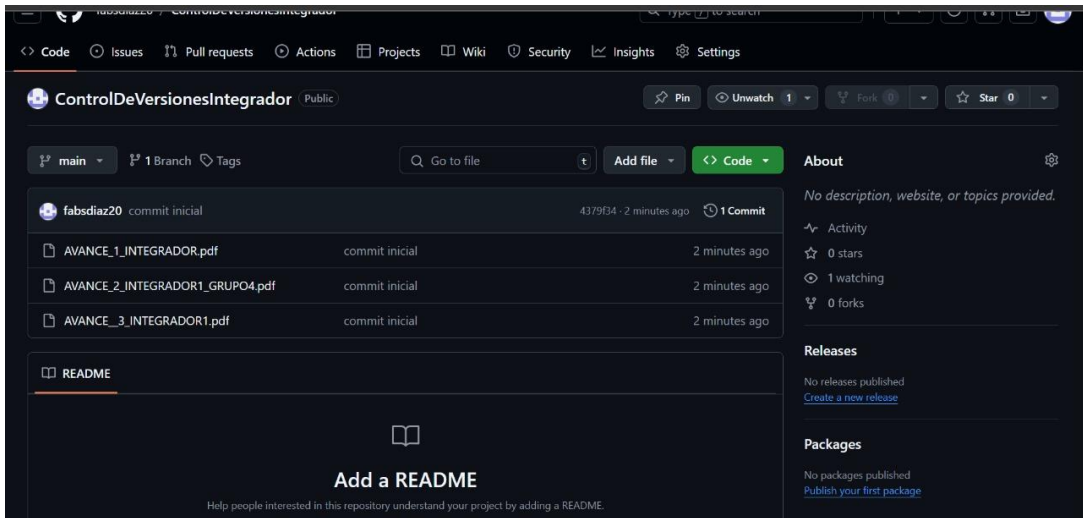
Semana para mostrar:				1																																																	
TAREA	ASIGNADO A	PROGRESO	INICIO	FIN																																																	
Avance del proyecto 1																																																					
Descripción del problema	Todos los integrantes	100%	15-8-24	22-8-24																																																	
Definir introduccion, problema y objetivos	Fabiana	100%	22-8-24	24-8-24																																																	
Definir alcances y limitaciones	Daniela	100%	24-8-24	28-8-24																																																	
Definir la justificación	Fabiana	100%	28-8-24	30-8-24																																																	
Realizar el estado de arte	Sofía	100%	30-8-24	1-9-24																																																	
Avance del proyecto 2																																																					
Definición de marco teórico	Fabiana-Sofia	100%	5-9-24	8-9-24																																																	
Definición de fórmulas	Fabiana	100%	8-9-24	10-9-24																																																	
Elaboración de prototipo balsam	Sofía	100%	10-9-24	13-9-24																																																	
Modelar base de datos	Daniela	100%	13-9-24	15-9-24																																																	
Elaborar los diagramas	Daniela	100%	13-9-24	16-9-24																																																	
Avance del proyecto 3																																																					
Desarrollo de encuesta	Daniela	100%	20-9-24	22-9-24																																																	
Desarrollo de presupuesto	Fabiana	100%	24-9-24	25-9-24																																																	
Realizar conclusiones	Fabiana	100%	25-9-24	29-9-24																																																	
Brindar recomendaciones	Daniela	100%	30-9-24	2-10-24																																																	
Presentar anexos	Sofía	100%	5-10-24	8-10-24																																																	
Desarrollo del sistema																																																					
Realizar estructura y diseño	Sofía	70%	30-9-24	31-9-24																																																	
Creacion de formularios	Fabiana	85%	1-10-24	2-10-24																																																	
Conexión a base de datos																																																					
Implementar modulos																																																					
Realizar logeo y register																																																					
Generar reportes																																																					
presentar control de versiones																																																					

- Anexo 4. Work Breakdown Structure - EDT

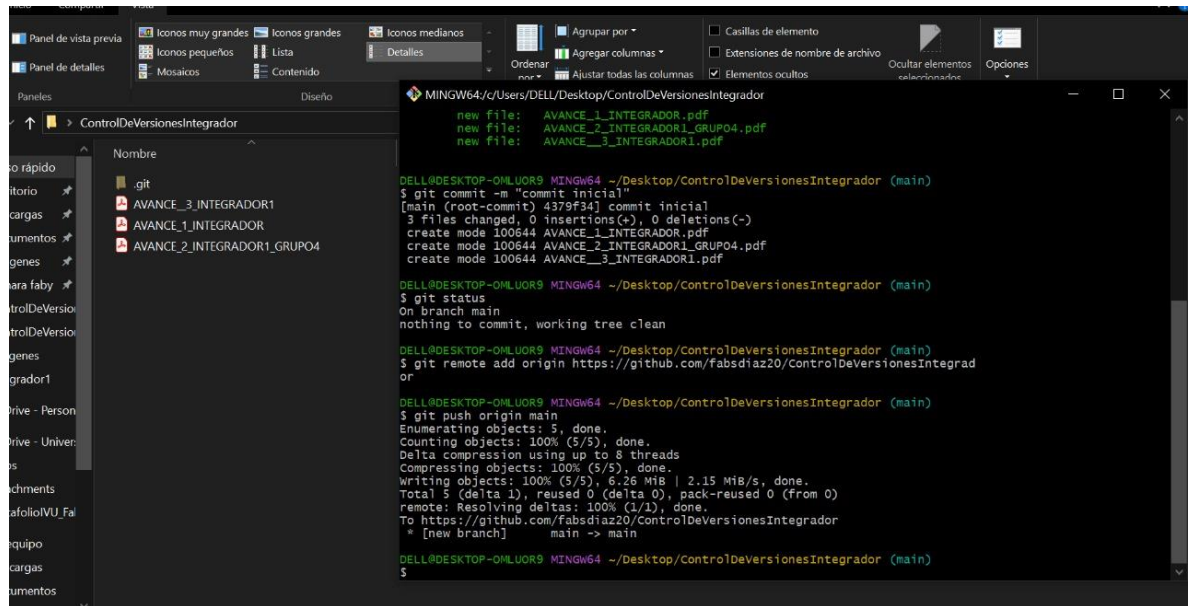


- Anexo 5. Control de Versiones: -Git & Github

Github



## Git



The screenshot shows a Windows File Explorer window on the left, displaying the contents of a folder named 'ControlDeVersionesIntegrador'. The folder contains a '.git' directory and three PDF files: 'AVANCE\_1\_INTEGRADOR', 'AVANCE\_2\_INTEGRADOR1\_GRUPO4', and 'AVANCE\_3\_INTEGRADOR1'. On the right, a terminal window titled 'MINGW64/c/Users/DELL/Desktop/ControlDeVersionesIntegrador' shows the following commands and output:

```
new file: AVANCE_1_INTEGRADOR.pdf
new file: AVANCE_2_INTEGRADOR1_GRUPO4.pdf
new file: AVANCE_3_INTEGRADOR1.pdf

DELL@DESKTOP-OMLUOR9 MINGW64 ~/Desktop/ControlDeVersionesIntegrador (main)
$ git commit -m "commit inicial"
[main (root-commit) 4379f34] commit inicial
3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 AVANCE_1_INTEGRADOR.pdf
create mode 100644 AVANCE_2_INTEGRADOR1_GRUPO4.pdf
create mode 100644 AVANCE_3_INTEGRADOR1.pdf

DELL@DESKTOP-OMLUOR9 MINGW64 ~/Desktop/ControlDeVersionesIntegrador (main)
$ git status
On branch main
nothing to commit, working tree clean

DELL@DESKTOP-OMLUOR9 MINGW64 ~/Desktop/ControlDeVersionesIntegrador (main)
$ git remote add origin https://github.com/fabsdiaz20/ControlDeVersionesIntegrador
DELL@DESKTOP-OMLUOR9 MINGW64 ~/Desktop/ControlDeVersionesIntegrador (main)
$ git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 6.26 MiB | 2.15 MiB/s, done.
Total 5 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/fabsdiaz20/ControlDeVersionesIntegrador
 * [new branch] main -> main

DELL@DESKTOP-OMLUOR9 MINGW64 ~/Desktop/ControlDeVersionesIntegrador (main)
$
```