

UNIVERSIDAD DEL VALLE DE GUATEMALA

Ingeniería de Software 1, sección 30



TAREA INVESTIGATIVA: FRAMEWORKS

VENTAS E INVENTARIO DIVINO SEAS

Sofia Garcia - 22210

Julio Garcia Salas - 22076

Joaquin Campos - 22155

Juan Fernando Menéndez - 17444

Juan Pablo Solis - 22102

GUATEMALA, marzo de 2024

Índice

Índice.....	2
Resumen.....	2
Introducción.....	3
Design Thinking.....	4
Prototipo Inicial.....	4
Observaciones de entrevistas a usuarios.....	6
Lista de cambios por hacer al nuevo prototipo:.....	6
Prototipo Final.....	7
Observaciones de entrevistas a usuarios.....	11
Análisis.....	12
Lista de Requisitos Funcionales.....	12
Requisitos Funcionales para los Dueños.....	12
Requisitos Funcionales para la Gerente de Inventario, Envíos y Ventas.....	12
Requisitos Funcionales para el Auditor.....	13
Requisitos Funcionales para el Equipo de Atención al Cliente.....	13
Requisitos Funcionales Generales.....	13
Diagrama de Clases.....	14
Descripción de clases.....	14
Diagrama de Paquetes.....	17
Descripción de paquetes.....	17
Diagrama de Clases Persistentes.....	18
Selección del tipo de almacenamiento.....	18
Diagrama Entidad-Relación.....	19
Diseño.....	19
Estimaciones.....	19
Investigación de tecnologías.....	20
Selección de tecnologías.....	21
Selección de Lenguaje.....	22
Informe de gestión.....	23
Desglose de tareas.....	23
Informe grupal.....	24

Resumen

Este documento aborda el desarrollo de un sistema e-commerce integral para Divino Seas, una marca de renombre en el mercado de bikinis, que enfrenta desafíos operativos debido a su dependencia de procesos manuales y sistemas no automatizados. Estos desafíos incluyen sobreventas, errores humanos, insatisfacción del cliente y vulnerabilidades en la seguridad, que limitan su eficiencia operativa y capacidad de satisfacer las demandas del mercado de manera efectiva. El objetivo principal es superar estos retos operativos y apoyar el crecimiento sostenido de la empresa mediante la implementación de soluciones

tecnológicas innovadoras. Esto se logrará a través del diseño y desarrollo de una plataforma online que no solo facilite las compras de los clientes sino que también provea un portal exclusivo para el personal, enfocado en la gestión unificada de inventarios, ventas, usuarios y reportes. El análisis exhaustivo contemplará la ideación de prototipos, especificación de requisitos funcionales, desarrollo backend, estructuración de paquetes y definición de la estrategia de persistencia de datos. Se seleccionarán las tecnologías más adecuadas basadas en los requisitos no funcionales y las tendencias actuales en desarrollo de software, con el fin de posicionar a Divino Seas a la vanguardia de la innovación digital.

Introducción

En la intersección de la innovación y la tradición, Divino Seas ha ido creciendo exponencialmente como una marca emblemática en el competitivo mercado de bikinis, comprometida con la calidad excepcional y el diseño vanguardista. Sin embargo, a pesar de su reconocimiento y éxito, la empresa enfrenta desafíos críticos en su operación diaria, marcados por la dependencia de procesos manuales y sistemas no automatizados. Esta realidad ha causado una serie de complicaciones operativas, tales como sobreventas, errores humanos, insatisfacción del cliente y vulnerabilidades en la seguridad, que obstaculizan su capacidad para satisfacer eficientemente las demandas del mercado.

Este documento se propone abordar estas problemáticas a través de la concepción e implementación de soluciones tecnológicas innovadoras. Con el objetivo de superar los retos operativos y respaldar el crecimiento sostenido de Divino Seas, se delineará el diseño y desarrollo de un sistema e-commerce integral. Esta plataforma no solo facilitará las compras online directamente a los clientes, sino que también incluirá un portal exclusivo para el personal, diseñado para la gestión unificada de inventarios, ventas, usuarios y reportes.

Para alcanzar este fin, se realizará un análisis exhaustivo que abarcará desde la ideación de prototipos para las interfaces de usuario hasta la especificación detallada de los requisitos funcionales del sistema. Se contemplarán aspectos clave del desarrollo backend, incluyendo la elaboración de un diagrama de clases, una descripción precisa de estas clases y la estructuración de paquetes que aseguren una cohesión elevada y un acoplamiento reducido. Además, se definirá la estrategia de persistencia de datos, optando por la solución de almacenamiento más adecuada basada en un análisis riguroso de las necesidades específicas del sistema.

En la sección de diseño, seleccionaremos las tecnologías más apropiadas para la implementación del proyecto, guiándonos por los requisitos no funcionales y las tendencias actuales en el desarrollo de software. Este enfoque integral nos ayudará a encaminar el proyecto en la línea correcta para superar las limitaciones operativas actuales de la empresa.

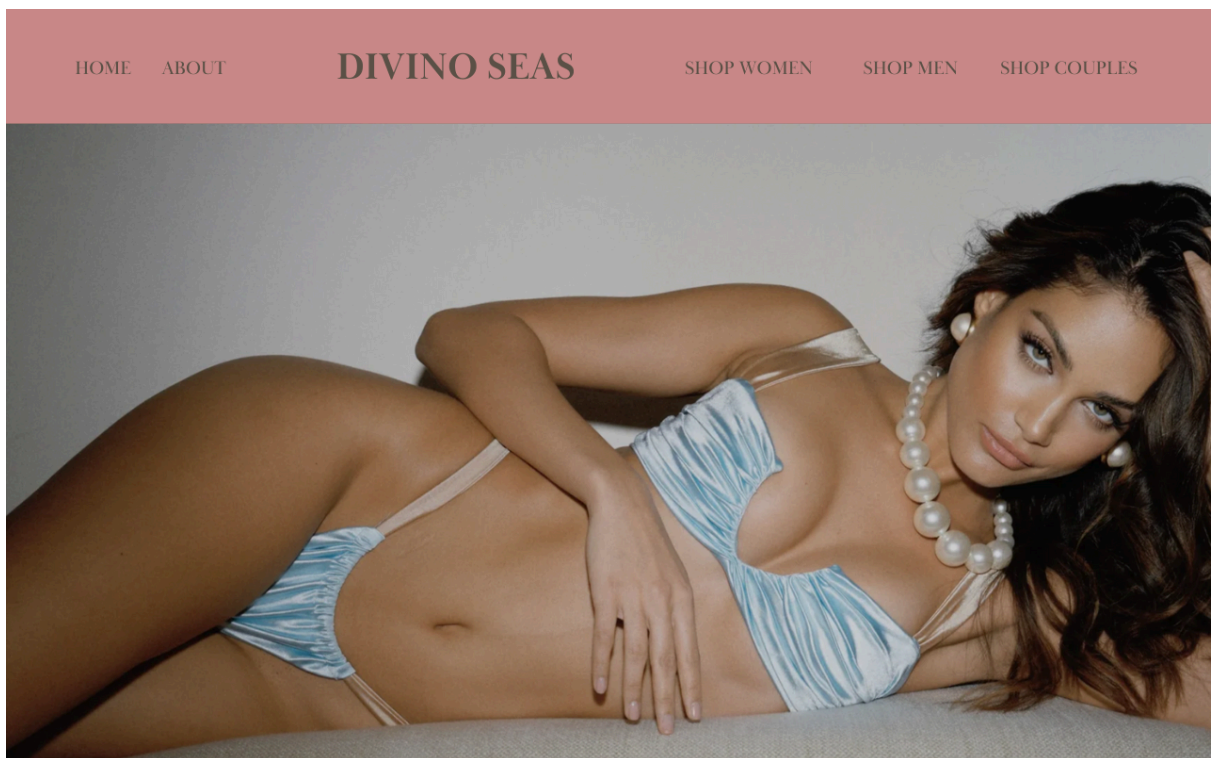
Design Thinking

Prototipo Inicial

Link al canva:

https://www.canva.com/design/DAF_d-VanIM/Mq7sPcveyKsBIAQ1O-svLg/edit?utm_content=DAF_d-VanIM&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton



Página web:



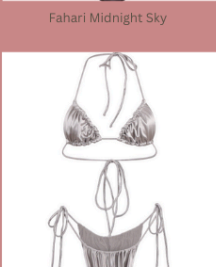
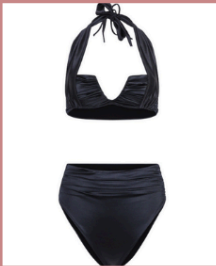
HOMEABOUT

DIVINO SEAS

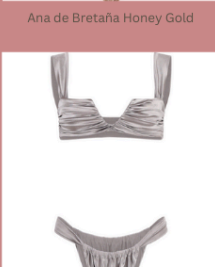
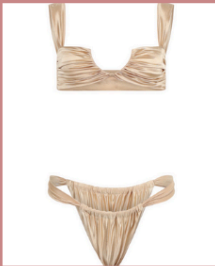
SHOP WOMENSHOP MENSHOP COUPLES



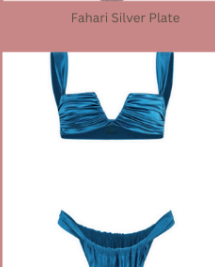

Ana de Bretaña Blue Atoll





Fahari Midnight Sky




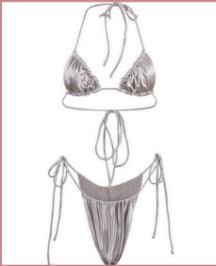
Ana de Bretaña Honey Gold



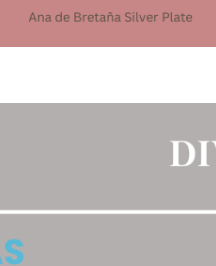
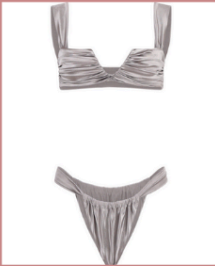
Fahari Silver Plate



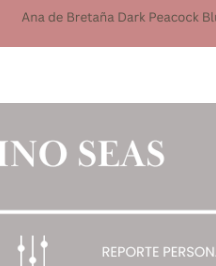
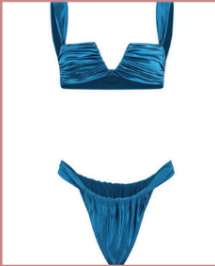
Ana de Bretaña Pastel Blue/Honey Gold



Inika Silver Plate



Ana de Bretaña Silver Plate



Ana de Bretaña Dark Peacock Blue

Página para dueños y trabajadores de la empresa:

InicioInventarioVentasAjustes

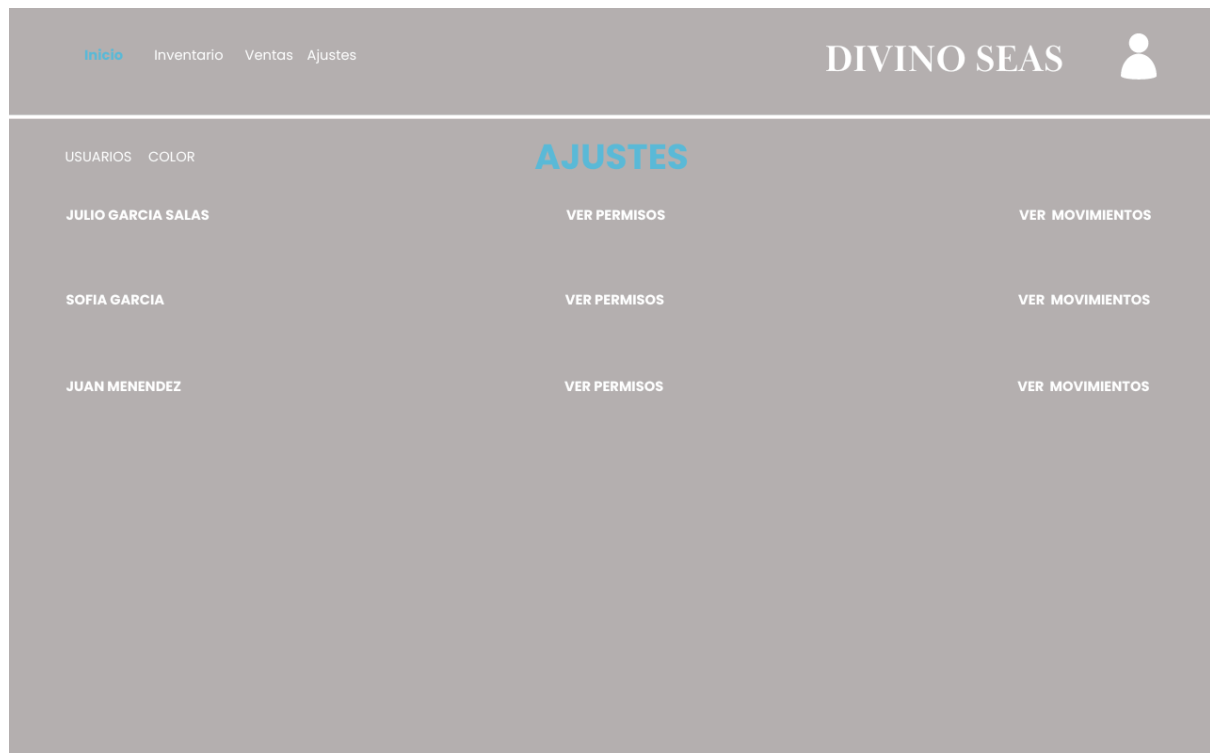
DIVINO SEAS

TODOENVIADOPENDIENTE

VENTAS

REPORTE PERSONALIZADO

ORDEN #1320	MARIA SOSA	Q. 1052	13/03/24
ORDEN #1320	MARIA SOSA	Q. 1052	13/03/24
ORDEN #1320	MARIA SOSA	Q. 1052	13/03/24
ORDEN #1320	MARIA SOSA	Q. 1052	13/03/24



Observaciones de entrevistas a usuarios

- “Divino Seas es una marca que da vibras de old money, el prototipo de la página web no representa eso.”
- “No ví que tuviera para buscar items y ver el carrito de compras. Toda página web de compras tiene eso, deberían de agregarlo.”
- “Los colores están demasiado llamativos como para lo que representa la marca.”
- “No dice contacto ¿cómo se supone que los clientes que no sepan que tienen redes sociales los pueda encontrar más fácilmente?”
- “Debe de decir el precio abajo del nombre del bikini”
- “Me parece que en vez de tener todo junto en la parte de inventario, solo se vean las existencias y para agregar o quitar se pueda presionar el elemento y ya poder editar todo en una página alterna.”
- “Tienen que poder editar los precios en el mismo lugar que editan las existencias.”
- “¿Cómo se vería el reporte de ventas?”

Lista de cambios por hacer al nuevo prototipo:

- **Revisar la estética de la página web:** Asegurarse de que el diseño y los colores reflejen una vibra de "old money". Usar colores más sobrios y una tipografía elegante que evoque exclusividad y sofisticación.


- **Agregar función de búsqueda y carrito de compras:** Es esencial que los usuarios puedan buscar productos específicos y ver su carrito de compras fácilmente. Podrían ser solo íconos representativos.
- **Modificar la paleta de colores:** Cambiar los colores utilizados por una paleta más reservada para la página de compras y acorde con la imagen de marca. Los colores pastel suaves, los tonos terrosos o una combinación monocromática pueden ser más apropiados. También no quedaría mal cambiar la de la página de empleados para que se pueda apreciar mejor todo.
- **Incluir información de contacto y acceso a redes sociales:** Añadir una sección de contacto claramente visible en el sitio, que podría incluir un formulario de contacto, dirección de correo electrónico y enlaces a las redes sociales.
- **Mostrar el precio de los productos debajo del nombre:** Asegura que el precio de cada bikini (o cualquier otro producto) esté claramente visible debajo del nombre del producto y los clientes también tomen sus decisiones a partir de ello, esto para que no tengan que estar preguntando o presionando la imagen para poder ver sus precios.
- **Reorganizar la gestión de inventario:** En lugar de mostrar todas las opciones de gestión de inventario juntas, permitir que los usuarios autorizados seleccionen un producto para ver más detalles y editar la información (como existencias y precios) en una página aparte. Esto hará que la administración sea más intuitiva.
- **Definir la presentación del reporte de ventas:** Considerar cómo se visualizará y accederá al reporte de venta. Podría ser un panel de administración donde se muestren gráficas y estadísticas clave sobre las ventas y que por aparte, dependiendo qué escoja la persona que desea ver, también pueda ver una tabla con las ventas. Que se muestre la información de fechas específicas en las que quiera buscar.


Prototipo Final


Link al canva:

https://www.canva.com/design/DAF_dNQZ0Yw/3BJLKne92wbZxC1q11ZSg/edit?utm_content=DAF_dNQZ0Yw&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton


Página Web nueva:


[HOME](#)[ABOUT](#)[CONTACT](#)




[SHOP WOMEN](#)[SHOP MEN](#)[SHOP COUPLES](#)



[HOME](#)[ABOUT](#)[CONTACT](#)



[SHOP WOMEN](#)[SHOP MEN](#)[SHOP COUPLES](#)

 <p>Ana de Breaña Blue Atoll Q. 688</p>	 <p>Fahari Midnight Sky Q. 701</p>	 <p>Ana de Breaña Honey Gold Q. 688</p>	 <p>Fahari Silver Plate Q. 701</p>
 <p>Ana de Breaña Pastel Blue/Honey Gold Q. 688</p>	 <p>Inika Silver Plate Q. 605</p>	 <p>Ana de Breaña Silver Plate Q. 688</p>	 <p>Ana de Breaña Dark Peacock Blue Q. 688</p>

Página para dueños y trabajadores de la empresa:

[Inicio](#)[Inventario](#)[Ventas](#)[Ajustes](#)





AGREGAR NUEVO ESTILO

INVENTARIO

REPORTE DE ENTRADAS Y SALIDAS

Ana de Bretaña Blue atoll TOP



7 XS

8 S

5 M

3 L

+

Ana de Bretaña Blue atoll BOTTOM



7 XS

8 S

5 M

3 L

+

[Inicio](#)[Inventario](#)[Ventas](#)[Ajustes](#)





AGREGAR NUEVO ESTILO

INVENTARIO

REPORTE DE ENTRADAS Y SALIDAS

Ana de Bretaña Blue atoll TOP



PRECIO

Q. 419.44

XS	7	↑↓
S	8	↑↓
M	5	↑↓
L	3	↑↓

Inicio

Inventario

Ventas

Ajustes

DIVINO SEAS

Q

TODO

ENVIADO

PENDIENTE

VENTAS

REPORTE PERSONALIZADO

ORDEN #1320

MARIA SOSA

Q. 1052

13/03/24

ORDEN #1320

MARIA SOSA

Q. 1052

13/03/24

ORDEN #1320

MARIA SOSA

Q. 1052

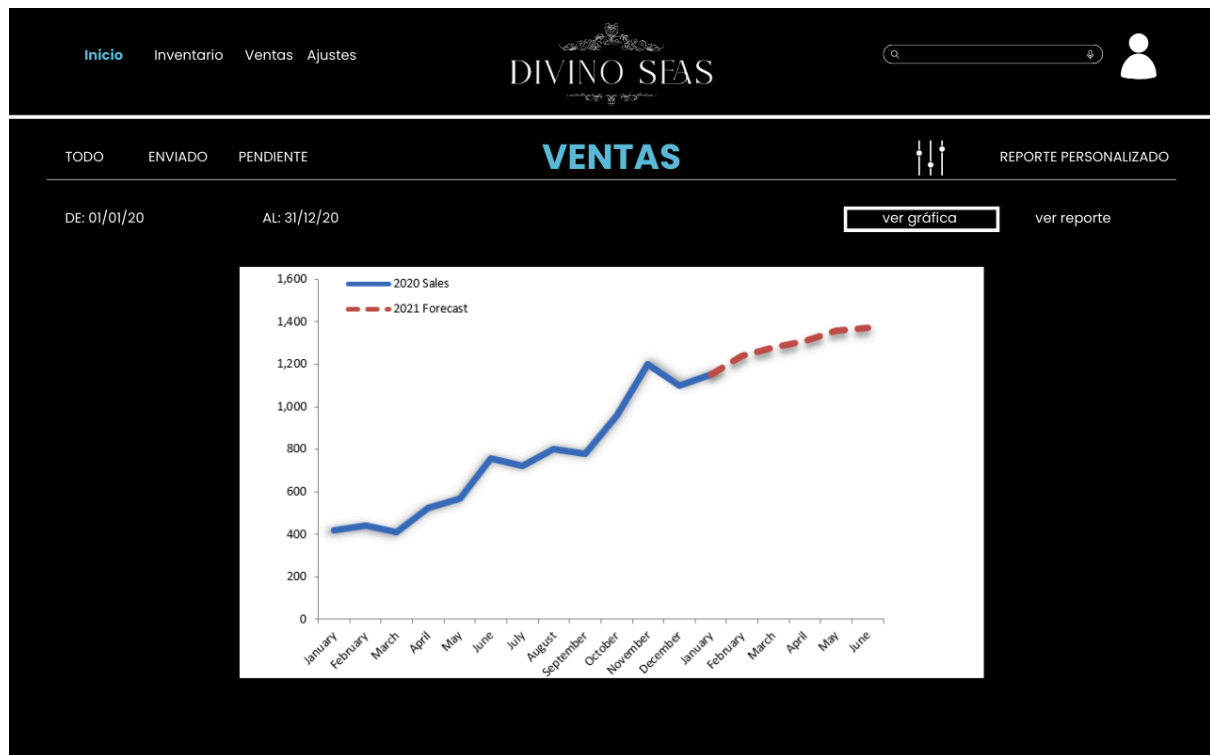
13/03/24

ORDEN #1320

MARIA SOSA

Q. 1052

13/03/24



Inicio Inventario Ventas Ajustes

DIVINO SEAS

AJUSTES

USUARIOS	COLOR	
JULIO GARCIA SALAS	VER PERMISOS	VER MOVIMIENTOS
SOFIA GARCIA	VER PERMISOS	VER MOVIMIENTOS
JUAN MENENDEZ	VER PERMISOS	VER MOVIMIENTOS

Observaciones de entrevistas a usuarios

- “Este prototipo está muchísimo mejor, representa mejor la imagen de cómo es la empresa en realidad.”
- “La gama de colores para cada página individual está muy linda”
- “No le cambiaría nada”

- “Muy bien hecho y es fácil de entender lo que está pasando”

Análisis

Lista de Requisitos Funcionales

Requisitos Funcionales para los Dueños

- RF1: Visualización de Dashboard Integral
- Detalle: El sistema debe proporcionar un dashboard que muestre en tiempo real el rendimiento de ventas, el estado del inventario en todas las plataformas y los KPIs críticos para permitir a los dueños evaluar la salud del negocio y tomar decisiones estratégicas informadas.
- RF2: Reportes de Análisis de Tendencias
- Detalle: El sistema debe generar reportes que analicen tendencias de ventas e inventario a lo largo del tiempo, ayudando a los dueños a planificar estrategias futuras.
- RF3: Alertas de KPIs
- Detalle: Configuración de alertas personalizables cuando ciertos KPIs críticos alcanzan umbrales predefinidos, permitiendo a los dueños tomar acciones correctivas de manera oportuna.

Requisitos Funcionales para la Gerente de Inventario, Envíos y Ventas

- RF1: Gestión Eficiente del Inventario
- Detalle: El sistema debe permitir al gerente gestionar el inventario de manera eficiente, incluyendo la capacidad de añadir, actualizar y eliminar productos, así como coordinar los envíos sin errores.
- RF2: Visualización de Ventas en Tiempo Real
- Detalle: Debe existir una funcionalidad que permita al gerente tener una visión clara y actualizada de las ventas en tiempo real a través de todos los canales de venta.
- RF3: Registro de Envíos
- Detalle: El sistema debe permitir al gerente registrar y rastrear el estado de los envíos, incluyendo información de seguimiento y fechas estimadas de entrega.
- RF5: Análisis de Demanda de Productos
- Detalle: Herramientas analíticas para prever la demanda de productos específicos, facilitando la planificación de inventario y promociones

Requisitos Funcionales para el Auditor

- RF1: Acceso a Registros Detallados
- Detalle: El sistema debe proporcionar al auditor acceso a registros detallados del inventario y ventas, que puedan ser fácilmente filtrados y verificados contra los estándares de cumplimiento.
- RF2: Registro de Auditorías
- Detalle: El sistema debe mantener un registro de todas las auditorías realizadas, incluyendo fecha, auditor responsable y hallazgos.
- RF3: Comparativas de Inventario
- Detalle: Herramientas para comparar registros de inventario físico contra registros del sistema, identificando discrepancias automáticamente.

Requisitos Funcionales para el Equipo de Atención al Cliente

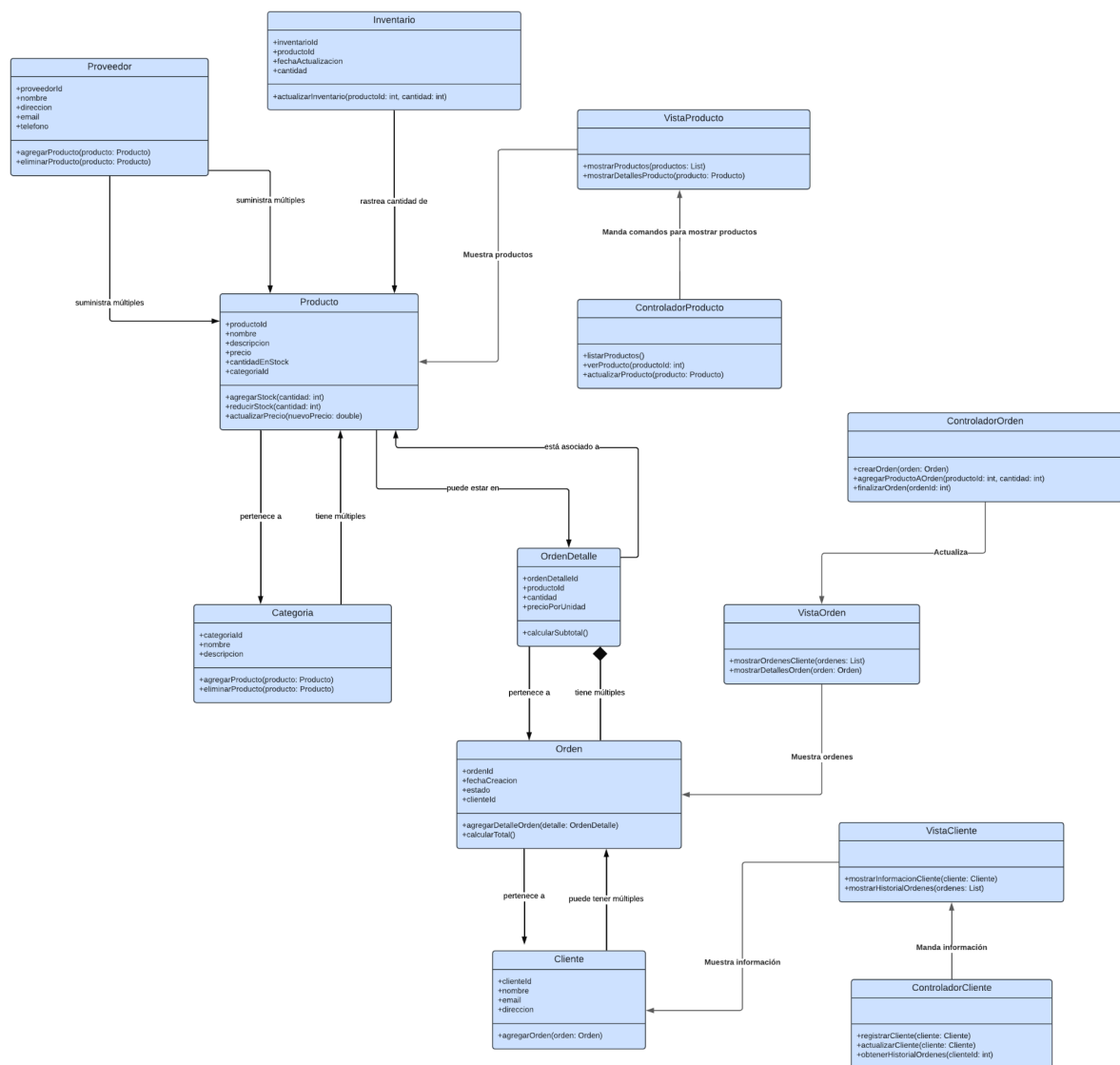
- RF1: Interfaz de Consulta de Inventario
- Detalle: El sistema debe ofrecer al equipo de atención al cliente una interfaz que proporcione información actualizada y precisa del inventario para responder rápidamente a las consultas de los clientes.
- RF2: Historial de Interacciones del Cliente
- Detalle: Acceso al historial completo de interacciones, compras y consultas de cada cliente para proporcionar un servicio personalizado.
- RF3: Gestión de Devoluciones y Cambios
- Detalle: Funcionalidad para procesar devoluciones y cambios de productos, incluyendo ajustes automáticos en el inventario y reembolsos.

Requisitos Funcionales Generales

- RF1: Integración con Plataformas de Pago
- Detalle: El sistema debe integrarse con diversas plataformas de pago para procesar transacciones de manera segura y eficiente.
- RF2: Acceso Multidispositivo
- Detalle: Diseño responsivo y adaptativo que asegure un acceso y funcionamiento óptimo desde cualquier dispositivo.
- RF3: Personalización de la Interfaz de Usuario
- Detalle: Permitir a los usuarios personalizar aspectos de la interfaz de usuario según sus preferencias y necesidades.
- RF5: Gestión de Permisos de Usuario

- Detalle: Sistema de gestión de permisos que permita asignar roles y accesos específicos a diferentes usuarios, asegurando que solo tengan acceso a la información y funcionalidades pertinentes a su rol.
- RF6: Copias de Seguridad Automáticas
- Detalle: El sistema debe realizar copias de seguridad automáticas de todos los datos críticos para prevenir pérdidas de información.

Diagrama de Clases



Descripción de clases

Modelo

Producto

- Descripción: Representa cada artículo o bien que se ofrece a la venta. Almacena información detallada como el nombre, descripción, precio y cantidad en stock. Es fundamental para operaciones de inventario y ventas.

Categoría

- Descripción: Agrupa productos similares o relacionados bajo una clasificación común, facilitando su búsqueda y gestión. Cada producto está asociado a una o más categorías.

Orden

- Descripción: Representa una transacción de venta, incluyendo los detalles de los productos comprados, la fecha de creación y el estado de la orden (por ejemplo, completada, pendiente). Relaciona clientes con los productos que adquieren.

OrdenDetalle

- Descripción: Detalla cada producto dentro de una orden, incluyendo la cantidad y el precio por unidad. Permite calcular subtotales por producto y contribuye al total de la orden.

Cliente

- Descripción: Almacena información sobre los clientes, como nombre, email y dirección. Es clave para gestionar las relaciones con los clientes y su historial de compras.

Proveedor

- Descripción: Contiene información sobre los proveedores de los productos, incluyendo nombre, dirección y contacto. Esencial para la gestión de inventario y órdenes de compra.

Inventario

- Descripción: Registra los niveles actuales de stock para cada producto, facilitando la gestión del inventario y la planificación de compras a proveedores.

Vista

VistaProducto

- Descripción: Presenta la información de los productos al usuario, incluyendo listados de productos y detalles específicos de cada uno. Interactúa con el usuario final a través de la interfaz gráfica.

VistaOrden

- Descripción: Muestra información relacionada con las órdenes, como el historial de órdenes de un cliente y los detalles de órdenes específicas.

VistaCliente

- Descripción: Proporciona una interfaz para visualizar y gestionar la información del cliente, incluyendo datos personales y el historial de compras.

Controlador

ControladorProducto

- Descripción: Maneja las interacciones del usuario relacionadas con los productos, como listar productos disponibles, ver detalles de un producto específico y actualizar la información del producto.

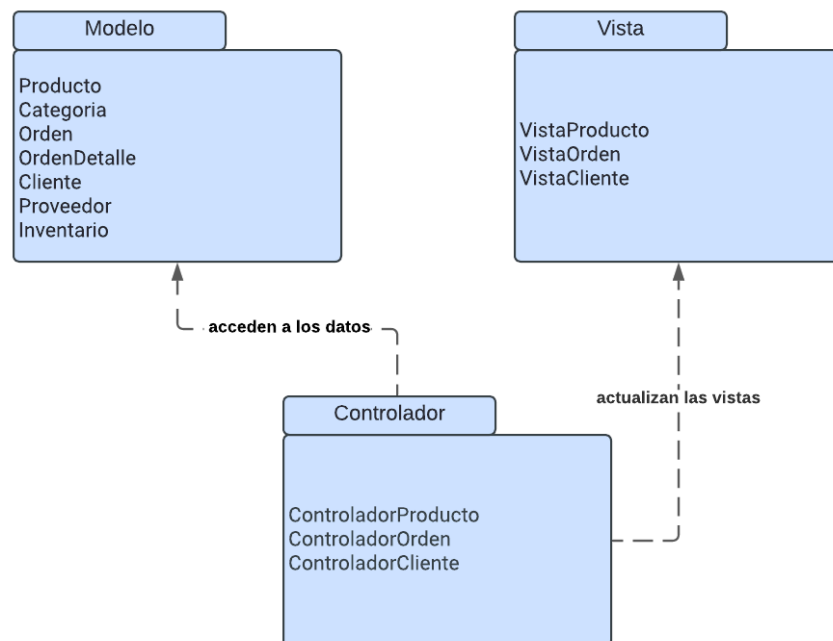
ControladorOrden

- Descripción: Gestiona la creación y modificación de órdenes, incluyendo la adición de productos a una orden, actualización del estado de la orden y visualización de detalles de la orden.

ControladorCliente

- Descripción: Responsable de las operaciones relacionadas con los clientes, como registrar nuevos clientes, actualizar información del cliente y recuperar el historial de compras.

Diagrama de Paquetes



Descripción de paquetes

Paquete Modelo

- Descripción: Contiene todas las clases que representan los datos y la lógica de negocio del sistema. Este paquete es el núcleo del sistema, manteniendo la información y el estado.

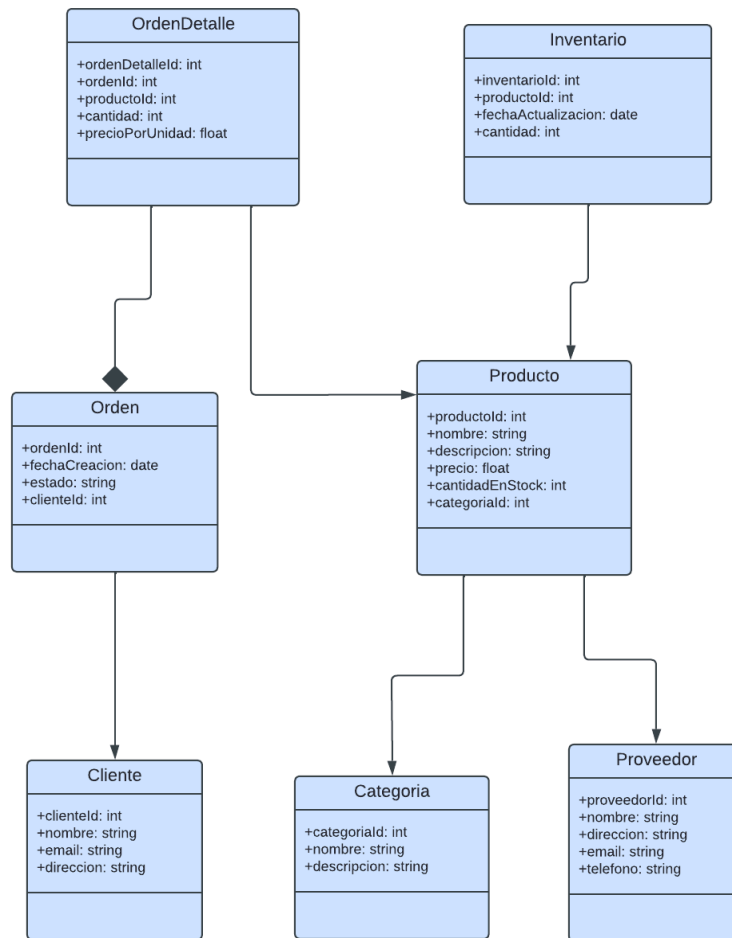
Paquete Vista

- Descripción: Agrupa las clases responsables de presentar la información al usuario, siguiendo el principio de separación de la lógica de negocio de su representación visual.

Paquete Controlador

- Descripción: Contiene las clases que manejan la interacción entre el usuario (a través de la Vista) y los datos del Modelo. Los controladores procesan las entradas del usuario, manipulan los modelos y actualizan las vistas.

Diagrama de Clases Persistentes



Selección del tipo de almacenamiento

Para este proyecto se optó por utilizar una base de datos relacional, por los siguientes motivos:

Estructura de Datos Compleja y Relacionada

El sistema maneja datos que están estrechamente relacionados, como productos, categorías, órdenes y clientes. Una base de datos relacional facilita la modelación de estas relaciones a través de claves foráneas y garantiza la integridad referencial.

Transacciones

Las operaciones de venta y gestión de inventario requieren transacciones que puedan ser atómicas, consistentes, aisladas y duraderas (ACID). Las bases de datos relacionales son fuertes en el manejo de transacciones ACID, asegurando la fiabilidad de las operaciones críticas.

Consultas Complejas

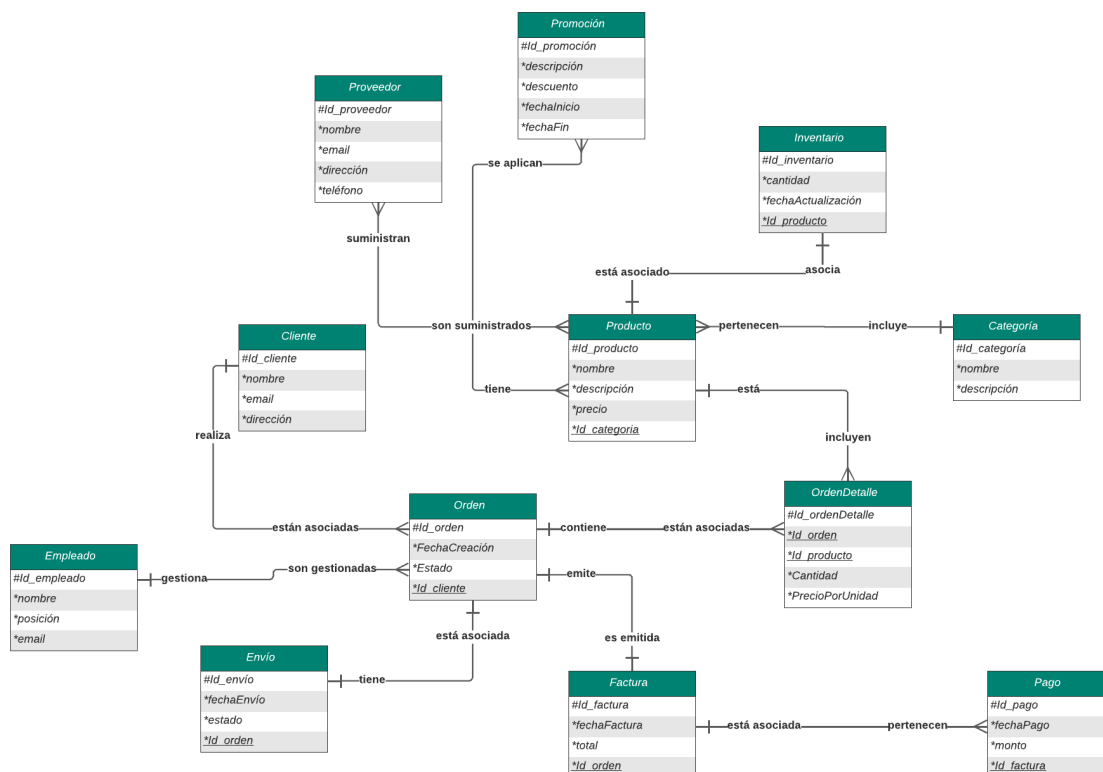
La necesidad de generar reportes detallados y realizar análisis de datos (por ejemplo, ventas por categoría, rendimiento de productos, etc.) se beneficia del poderoso sistema de consultas SQL que ofrecen las bases de datos relacionales.

Escalabilidad y Mantenibilidad

Aunque las bases de datos no relacionales pueden ofrecer ventajas en términos de escalabilidad horizontal, las bases de datos relacionales modernas han avanzado mucho en este aspecto, ofreciendo soluciones híbridas y opciones de escalabilidad que pueden satisfacer las necesidades del sistema.

Luego de tener esto claro, se escogió PostgreSQL para el almacenamiento de programa debido a su corta curva de aprendizaje y lo mucho que puede ofrecer en grandes proyectos.

Diagrama Entidad-Relación



Diseño

Estimaciones

- Cantidad de usuarios: se considera una cantidad de **+11000** usuarios totales esto sin tomar en cuenta las licencias, el valor se obtuvo de los clientes actuales de Divino

Seas. Se debe tomar en cuenta también por las licencias este número se puede ver limitado.

- Tiempo promedio: como se calculó en los requisitos no funcionales se tiene un aproximado de respuesta de 2 a 3 segundos, este puede variar, teniendo un estimado máximo de 5 segundos.
- Usuarios simultáneos: se calcula entre los 50 máximo pero se debe tomar en cuenta que de serlo necesario podríamos incluir licencias, de manera que este pueda aumentar.
- Escalabilidad del sistema: El total de espacio de disco duro se puede calcular sumando: el espacio total de dependencias y recursos estáticos, el total de espacio de código fuente, tamaño total de las bases de datos, memoria dinámica, por lo que se aproxima un total de 20 GB máximo, no obstante puede variar.
- Disponibilidad: Consideramos que la disponibilidad de información se verá limitada para cada usuario, de manera que la disponibilidad será dependiente como se mostró en los prototipos.
- Confiabilidad: Como todo proyecto se esperan los menores fallos, no obstante, se depende de la información ingresada.

Investigacion de tecnologías

- **React:** Se caracteriza por su enfoque en componentes reutilizables, lo que permite a los desarrolladores construir aplicaciones complejas a partir de pequeñas piezas independientes y reutilizables de código. React introduce el concepto de Virtual DOM, una representación ligera del DOM (Document Object Model) que mejora significativamente el rendimiento de las aplicaciones al minimizar las actualizaciones en el DOM real del navegador. Esta característica permite a React determinar de manera inteligente cuándo y qué partes de la interfaz de usuario necesitan ser actualizadas, reduciendo así el tiempo y los recursos necesarios para renderizar cambios en la pantalla.
- **AngularJs:** AngularJS se destaca por introducir el concepto de two-way data binding, lo que significa que los cambios en el modelo de datos automáticamente se reflejan en la interfaz de usuario y viceversa, facilitando así la sincronización entre el modelo y la vista sin necesidad de escribir código adicional. Además, AngularJS promueve la estructura MVC (Modelo-Vista-Controlador), ayudando a los desarrolladores a organizar y modularizar su código de manera más efectiva.
- **ExpressJs:** Express.js es un framework de desarrollo web para Node.js conocido por su flexibilidad y minimalismo. Diseñado para facilitar la creación de aplicaciones web y APIs de manera rápida y eficiente, Express simplifica el manejo de solicitudes y respuestas HTTP a través de su sistema de middleware, permitiendo a los desarrolladores personalizar el comportamiento de sus aplicaciones con gran precisión. Esta característica, junto con un enfoque en el rendimiento y la simplicidad, lo convierte en una elección popular entre los desarrolladores que buscan construir desde aplicaciones simples hasta complejos sistemas web. y llamar al siguiente middleware en la pila.

- **Flutter:** Flutter es un framework de código abierto el cual rápidamente ganó popularidad por su capacidad de compilar a código nativo, lo que permite a los desarrolladores construir aplicaciones de alto rendimiento y con una apariencia visualmente atractiva para iOS, Android, web y escritorio con una sola programación. Flutter se distingue por su motor de renderizado propio, el cual dibuja los widgets directamente en el canvas del dispositivo, ofreciendo una gran flexibilidad y control sobre el aspecto de la aplicación.

Selección de tecnologías

Basándonos en los requisitos no funcionales establecidos y las estimaciones de dimensiones del sistema, la elección de React y ExpressJS como tecnologías para el proyecto de Divino Seas parece adecuada por las siguientes razones:

1. Tiempo de respuesta: React es conocido por su rendimiento y capacidad para renderizar interfaces de usuario de manera eficiente, lo que contribuiría a cumplir con el requisito de un tiempo de respuesta menor a 2 segundos para consultas de inventario.
2. Disponibilidad del sistema: ExpressJS, como un framework de Node.js, es altamente escalable y puede manejar una carga considerable de usuarios simultáneos. Además, su naturaleza asincrónica facilita la implementación de estrategias de alta disponibilidad y tolerancia a fallos, contribuyendo al objetivo de lograr una disponibilidad del sistema del 99.9% mensualmente.
3. Adaptabilidad y Usabilidad: React permite crear interfaces de usuario altamente adaptables y responsivas, lo que garantizaría una experiencia de usuario óptima en una variedad de dispositivos y resoluciones de pantalla, cumpliendo así con los requisitos de adaptabilidad a diferentes tamaños de pantalla y dispositivos móviles, así como con las pruebas de usabilidad y diseño responsivo.
4. Escalabilidad y Mantenibilidad: Tanto React como ExpressJS son tecnologías ampliamente utilizadas y respaldadas por una gran comunidad de desarrolladores. Esto significa que hay una abundancia de recursos, documentación y herramientas disponibles para escalar el sistema y mantenerlo fácilmente en el futuro. Además, ExpressJS es compatible con estrategias de despliegue continuo, lo que facilitaría las actualizaciones del sistema sin tiempo de inactividad y la implementación de prácticas de mantenimiento continuo.
5. Integración y Restricciones: ExpressJS es altamente compatible con una variedad de sistemas y servicios externos, lo que facilitaría la integración con sistemas de pago en

línea existentes, cumpliendo así con el requisito de integración con sistemas de pago en línea. Además, la naturaleza flexible de React permitiría adaptarse fácilmente a requisitos específicos del mercado y culturales, como la adaptación cultural en el diseño para mercados específicos.

Selección de Lenguaje

JavaScript es el lenguaje de programación elegido para el desarrollo del proyecto Divino Seas, aprovechando principalmente los frameworks de desarrollo React y ExpressJS. A continuación, se presenta una descripción detallada de JavaScript, así como sus ventajas y desventajas, y las razones nos llevaron cómo equipo a seleccionarlo:

- Descripción de JavaScript:
 - JavaScript es un lenguaje de programación ampliamente utilizado en el desarrollo web para crear interactividad en las páginas web. Es un lenguaje de script del lado del cliente que se ejecuta en el navegador del usuario. Además, con el surgimiento de Node.js, JavaScript también se puede ejecutar del lado del servidor, lo que lo convierte en un lenguaje versátil y adecuado para el desarrollo full-stack.
- Ventajas de JavaScript:
 - **Amplia adopción:** JavaScript es uno de los lenguajes de programación más populares y ampliamente adoptados en el mundo del desarrollo web, lo que significa que hay una gran cantidad de recursos, bibliotecas y frameworks disponibles para su uso.
 - **React y ExpressJS:** React es un poderoso framework de JavaScript para la creación de interfaces de usuario interactivas, mientras que ExpressJS es un framework minimalista y flexible para el desarrollo de aplicaciones web del lado del servidor. La combinación de ambos, junto con JavaScript, permite un desarrollo coherente y eficiente tanto del lado del cliente como del servidor.
 - **Ecosistema robusto:** JavaScript cuenta con un ecosistema de desarrollo muy robusto, con una amplia gama de herramientas y bibliotecas que facilitan la construcción de aplicaciones web modernas y escalables.
 - **Asincronía:** La naturaleza asíncrona de JavaScript, especialmente en el entorno de Node.js, permite la ejecución de operaciones de entrada/salida de manera eficiente, lo que es crucial para aplicaciones web en tiempo real y de alto rendimiento.
 - **Flexibilidad:** JavaScript es un lenguaje flexible que permite a los desarrolladores adoptar diferentes estilos de programación, como programación orientada a objetos, funcional o basada en prototipos, lo que brinda libertad para abordar los desafíos de desarrollo de diversas maneras.
- Desventajas de JavaScript:

- **Compatibilidad entre navegadores:** Aunque ha mejorado significativamente en los últimos años, la compatibilidad entre navegadores aún puede ser un problema al desarrollar aplicaciones web complejas, especialmente al lidiar con versiones más antiguas de navegadores.
- **Gestión de errores:** JavaScript puede ser propenso a errores, especialmente en proyectos grandes y complejos. La falta de tipos estáticos puede llevar a errores difíciles de depurar y detectar.
- **Rendimiento:** Aunque JavaScript es bastante rápido, puede ser superado por lenguajes de programación de bajo nivel en términos de rendimiento en operaciones intensivas de CPU.
- Razones para la selección:
 - **Compatibilidad con React y Express JS:** JavaScript es el lenguaje nativo tanto de React como de ExpressJS. Esto garantiza una integración perfecta y un flujo de trabajo coherente entre el frontend y el backend del proyecto Divino Seas.
 - **Ecosistema amplio y maduro:** JavaScript ofrece un ecosistema de desarrollo maduro y bien establecido que facilita el desarrollo, la depuración y el mantenimiento del proyecto a lo largo del tiempo.
 - **Rendimiento y asincronía:** La asincronía de JavaScript, especialmente en Node.js, se alinea bien con los requisitos de rendimiento y escalabilidad del proyecto Divino Seas, lo que permite manejar grandes volúmenes de transacciones de manera eficiente.
 - **Flexibilidad y productividad:** JavaScript es un lenguaje flexible que permite a los desarrolladores trabajar de manera rápida y eficiente, lo que es crucial para cumplir con los plazos de entrega del proyecto y adaptarse a los requisitos cambiantes del cliente.

En resumen, JavaScript se seleccionó como la tecnología principal para el proyecto Divino Seas debido a su compatibilidad con React y ExpressJS, su amplio ecosistema de desarrollo, su rendimiento y asincronía, y su flexibilidad y productividad. Estas características hacen que JavaScript sea una opción sólida y adecuada para el desarrollo de aplicaciones web modernas y escalables.

Informe de gestión

Desglose de tareas

Tarea	Integrante
Resumen	SOFIA ISABEL GARCIA ORELLANA
Introducción	SOFIA ISABEL GARCIA ORELLANA
Prototipo inicial	SOFIA ISABEL GARCIA ORELLANA y JUAN PABLO SOLIS ALBIZUREZ
Entrevistas de prototipo	SOFIA ISABEL GARCIA ORELLANA y JUAN FERNANDO MENENDEZ GU...
Prototipo final	SOFIA ISABEL GARCIA ORELLANA
Entrevistas	SOFIA ISABEL GARCIA ORELLANA y JUAN FERNANDO MENENDEZ GU...
Requisitos funcionales	JULIO ANDRES GARCIA SALAS PE...
Diagrama de clases	JULIO ANDRES GARCIA SALAS PE... y JOSE JOAQUIN CAMPOS ENRIQUEZ
Diagrama de paquetes	JULIO ANDRES GARCIA SALAS PE... y JOSE JOAQUIN CAMPOS ENRIQUEZ
Diagrama de clases persistentes	JULIO ANDRES GARCIA SALAS PE...
Diagrama entidad relación	JULIO ANDRES GARCIA SALAS PE...
Estimaciones	JOSE JOAQUIN CAMPOS ENRIQUEZ
Investigación de tecnologías	JUAN FERNANDO MENENDEZ GU... y JUAN PABLO SOLIS ALBIZUREZ
Selección de tecnologías	JUAN PABLO SOLIS ALBIZUREZ y JUAN FERNANDO MENENDEZ GU...
Selección de lenguajes	JUAN PABLO SOLIS ALBIZUREZ y JUAN FERNANDO MENENDEZ GU...
Desglose de tareas	JOSE JOAQUIN CAMPOS ENRIQUEZ
Informe grupal	JOSE JOAQUIN CAMPOS ENRIQUEZ

Informe de gestión grupal

Formularios individuales en el github

- Investigación previa (10/3/2024 – 9:00pm/10:00pm)
Se trabajó en equipo para dividir la investigación.
- Requisitos funcionales (13/3/2024 - 10:00pm/11:00pm)
Se encargó un compañero de manera que no hubieran requisitos repetidos y se pudiera optimizar trabajo.
- Elaboración de diseño(13/3/2024 - 10:00pm/11:00pm)
Se trabajaba paralelo entre compañeros para poder trabajar rápido cosas que podían hacerse individualmente.
- Entrevistas (14/03/2024 - 3:00pm/4:00pm)
Se trabajó más individual para ahorrar manos de manera que fue efectivo y la información se compartió con el grupo.
- Prototipos (14/03/2024 - 4:00pm/4:30pm)
En los prototipos se profundizó lo máximo posible de manera que se pudiera tener una idea de como se quería el producto final, según lo descrito en las entrevistas anteriores, este fue realizado de manera individual para poder después presentar diseños y demás a usuarios en entrevistas y obtener el prototipo final.
- Diagrama de clases (14/03/2024 - 11:30pm/12:30pm)
El trabajo fue en equipo en la elaboración de los diagramas de clases, de manera que nos pudiéramos empezar a tener ideas generales de la estructura, poder hablar de las mismas y estar de acuerdo o en desacuerdo, asimismo, se hizo en grupo para evitar dejar pasar cosas importantes.
- Diagrama de paquetes (14/03/2024 - 12:40 am/1:00 am)
La ventaja de trabajar estos diagramas en grupo fue que al tener previamente el diagrama de clases se pudo hacer una muy buena relación entre ambas con ayuda de los integrantes.
- Diagrama de clases persistentes(14/03/2024 - 1:10am/1:50am)
Al igual que los diagramas anteriores, este se trabajó en parejas de manera que se pudiera tomar en cuenta la opinión de varios, esto porque define la estructura del proyecto interno.