**INFORME ULTIMO SPRINT**

****

**PRESENTADO POR:**

ALEJANDRO FERNÁNDEZ VITONCÓ

WILLIAM RODRIGO DÍAZ MACA

MANUELA MENESES ERAZO

JHOSSEF NICOLÁS CONSTAÍN

JULIAN FELIPE PIAMBA VIVEROS

GUILLERMO ALBERTO PEÑA MUÑOZ

GERALDINE IVONNE CARLOSAMA

BAIRON CAMILO PALECHOR

**PRESENTADO A:**

JULIO ARIEL HURTADO

SANTIAGO HYUN DORADO

ALEX TORRES

**UNIVERSIDAD DEL CAUCA**

**FACULTAD DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES**

**INGENIERÍA DE SISTEMAS**

**JULIO 2023, POPAYÁN, CAUCA**

**INDICE**

Resumen. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 3

Notas para tener en cuenta. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 4

Arquitectura del proyecto. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 5

Decisiones de diseño más relevantes. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 6

Conclusión. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 7

Referencias. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 8

**RESUMEN.**

El presente proyecto tiene como objetivo el desarrollo de una aplicación back-end para la empresa Colombo-Argentina Digital Asociados SA, con el fin de lanzar al mercado una plataforma de comercio electrónico que compita con plataformas reconocidas como Mercado Libre. Esta aplicación permitirá a los usuarios comprar, vender y distribuir productos de forma segura, rápida y fácil de usar.

Se abordarán los requisitos funcionales y no funcionales de la aplicación, incluyendo la gestión de usuarios, la funcionalidad para vendedores y compradores, así como el proceso de entrega de productos. Esto implicará la creación de usuarios con roles específicos, la administración de productos por parte de los vendedores, la búsqueda y selección de productos por parte de los usuarios, el control del carrito de compra y el proceso de check out para los usuarios registrados. Además, se implementará un sistema para los entregadores o repartidores encargados de la entrega de los productos.

El objetivo final es proporcionar una plataforma integral que cumpla con las necesidades de los usuarios y permita a la empresa competir exitosamente en el mercado del comercio electrónico.

**NOTAS PARA TENER EN CUENTA EN IMPLEMENTACION.**

* Requisitos del proyecto: Explica los requisitos funcionales y no funcionales establecidos para la aplicación. Esto incluiría la necesidad de un sistema de gestión de productos, la capacidad de realizar operaciones CRUD y la capacidad de búsqueda por palabras que pueden ser el nombre o la descripción, también el hecho de poder manejar ciertos filtros en la búsqueda como lo son la categoría, el precio o la ubicación.
* Necesidad de funcionalidades: Destaca la importancia de tener un sistema de gestión de productos completo. Menciona que el CRUD de productos es esencial para permitir a los usuarios crear, ver, actualizar y eliminar productos de manera eficiente. Además, resalta cómo la función de búsqueda por nombre o descripción mejora la experiencia del usuario al permitirles encontrar rápidamente los productos que desean.
* Experiencia del usuario: Enfatiza que la implementación de un sistema CRUD de productos y una función de búsqueda mejora la experiencia del usuario al brindarles la capacidad de interactuar de manera intuitiva y eficiente con la aplicación. Al permitirles administrar y buscar productos fácilmente, estás facilitando el uso de la plataforma y aumentando la satisfacción del usuario.
* Eficiencia operativa: Resalta cómo el sistema de gestión de productos y la función de búsqueda mejoran la eficiencia operativa interna. Al permitir a los administradores y empleados gestionar los productos de manera efectiva y encontrar información rápidamente, se agilizan las tareas diarias y se reduce el tiempo empleado en la gestión manual de productos.
* Análisis de mercado: Puedes respaldar tus decisiones mencionando la importancia de estas funcionalidades en aplicaciones similares a Mercado Libre. Realiza un análisis comparativo de otras plataformas de comercio electrónico exitosas y resalta cómo la implementación de un sistema CRUD de productos y una función de búsqueda es una práctica común y exitosa en la industria.
* Beneficios a largo plazo: Explica cómo las decisiones tomadas en relación al CRUD de productos y la función de búsqueda sientan las bases para futuras mejoras y expansiones del sistema. Destaca la escalabilidad y flexibilidad de estas funcionalidades, permitiendo que la aplicación evolucione y se adapte a las necesidades cambiantes del mercado y de los usuarios.

**ARQUITECTURA DEL PROYECTO.**

**Por qué usar arquitectura hexagonal con microservicios y no con arquitectura monolítica.**

Consideramos que tenemos unos puntos razonables y debidamente justificados en la forma de trabajar su arquitectura:

1. Modularidad y desacoplamiento: Decimos que en la arquitectura de microservicios permite dividir la aplicación en componentes más pequeños y autónomos, conocidos como microservicios. Cada microservicio se enfoca en una función específica, como la gestión de usuarios, productos, compras y entregas. Esto ofrece una mayor modularidad y facilita el desarrollo, la implementación y el mantenimiento de cada componente de manera independiente. Además, si se requiere escalar una función en particular, solo es necesario escalar el microservicio correspondiente, en lugar de toda la aplicación.
2. Escalabilidad y rendimiento: Con la arquitectura de microservicios, es más sencillo escalar y distribuir el sistema de forma horizontal. Cada microservicio puede escalarse de manera independiente según las necesidades de carga o demanda, lo que permite gestionar eficientemente un crecimiento en el número de usuarios o transacciones sin afectar el rendimiento general del sistema. Esto proporciona flexibilidad y capacidad para manejar altos volúmenes de tráfico.
3. Aislamiento de errores y tolerancia a fallos: En una arquitectura monolítica, un fallo en un componente puede afectar a todo el sistema. Sin embargo, en una arquitectura de microservicios, los errores son más localizados. Si un microservicio falla, los demás pueden seguir funcionando sin interrupciones, lo que mejora la tolerancia a fallos y la disponibilidad general del sistema.
4. Tecnologías y equipos especializados: Con microservicios, es posible utilizar diferentes tecnologías y herramientas para cada componente, según sea necesario. Esto permite seleccionar las tecnologías más adecuadas para cada microservicio y aprovechar las habilidades y especialidades de los equipos de desarrollo. Además, al ser componentes más pequeños y específicos, es más fácil para los equipos gestionar, mantener y actualizar cada microservicio de manera individual.

Podemos concluir que la arquitectura hexagonal con microservicios ofrece una solución flexible, escalable y robusta para este proyecto, permitiendo un desarrollo modular, una mejor gestión de la carga, una mayor tolerancia a fallos y la utilización eficiente de recursos tecnológicos y humanos. Esta arquitectura proporciona los fundamentos necesarios para construir una aplicación de comercio electrónico segura, eficiente y capaz de adaptarse a las necesidades cambiantes del mercado.

**DECISIONES DE DISEÑO MAS RELEVANTES**

Al dividir el proyecto en secciones basadas en los microservicios, cada compañero se hizo responsable de desarrollar y tomar decisiones específicas dentro de su área asignada. Cada microservicio (Usuario/Persona, Producto, Compra y Delivery) fue abordado individualmente por un miembro del equipo, quien asumió la responsabilidad de su desarrollo y toma de decisiones relacionadas. A través de la colaboración y la comunicación entre los compañeros, se aseguró que las decisiones tomadas en cada microservicio fueran coherentes y estuvieran alineadas con los objetivos generales del proyecto. Esta división permitió un enfoque más especializado y eficiente en el desarrollo de cada componente del sistema, maximizando la calidad y la eficacia del trabajo realizado en conjunto.

Las decisiones clave tomadas para la implementación en el proyecto fueron las siguientes:

1. Arquitectura de microservicios: Se optó por dividir la aplicación en microservicios independientes para cada funcionalidad (Usuario/Persona, Producto, Compra, Delivery). Esto permitió una estructura modular y escalable del sistema.
2. Independencia de los microservicios: Cada microservicio fue desarrollado de manera independiente, lo que facilitó la colaboración entre los miembros del equipo y permitió un avance más eficiente en paralelo.
3. Enfoque de puertos y adaptadores: Se utilizó un enfoque flexible de puertos y adaptadores para organizar y estructurar el código de los microservicios, lo que permitió una mayor flexibilidad y facilidad en el intercambio de componentes.
4. Bases de datos separadas: Se decidió que cada microservicio tuviera su propia base de datos, lo que garantizó una gestión y escalabilidad eficiente de los datos relacionados con cada funcionalidad.

Adjuntamos tabla de distribución de tareas.



**CONCLUSIÓN.**

El proyecto no solo permitió desarrollar una aplicación funcional, sino que también brindó la oportunidad de adquirir conocimientos y habilidades en el diseño de arquitecturas escalables, la implementación de microservicios, la seguridad de datos y el trabajo colaborativo en un entorno de desarrollo de software.

**REFERENCIAS.**

1. https://www.adictosaltrabajo.com/2017/01/12/comunicacion-entre-microservicios-compartimos-codigo/
2. https://medium.com/@edusalguero/arquitectura-hexagonal-59834bb44b7f
3. https://chat.openai.com/
4. https://www.kranio.io/blog/microservicios-que-son-como-funcionan-y-mejores-practicas-de-uso
5. https://www.youtube.com/results?search\_query=arquitectura+hexagonal+java