**UNIVERSIDAD DEL VALLE DE GUATEMALA**

Logotipo

Descripción generada automáticamente

**Corte #3**

BRANDON JAVIER REYES MORALES - 22992

CARLOS ALBERTO VALLADARES GUERRA - 221164

GUSTAVO ADOLFO CRUZ BARDALES- 22779

PAULA REBECA BARILLAS ALVAREZ - 22764

RODRIGO ALFONSO MANSILLA DUBÓN - 22611

Catedrático: Lynette García Pérez

Ingeniería de Software

Guatemala, 2024

1. **Resumen**
2. **Introducción**
3. **Design Thinking: Prototipos**
   1. **Versiones**
   2. **Cambios**
4. **Análisis**
   1. **Requisitos Funcionales**

|  |  |
| --- | --- |
| Historia | Requisito |
| Emprendedor | * Sistema De Gestión de inventarios que actualice el stock después de cada transacción. * Análisis de ventas en tiempo real para identificar tendencias y ajustar estrategias * Pánel de control para el seguimiento de pedidos y gestión de devoluciones. |
| Vendedor Casual | * Lista de productos con plantillas y asistencia. * Opciones de configuración de privacidad avanzadas. * Sistema de Valoraciones y reseñas. * Autenticación de dos factores para acceder al área de gestión de ventas. |
| Comprador | * Búsqueda Avanzada por medio de filtros por categoría, precio y ubicación del vendedor. * Pasarela de pago seguro con múltiples opciones de pago. * Sistema de garantía de compra para protección contra fraudes o productos no entregados. * Acceso a historial de compras detallado para el seguimiento de los gastos. |
| Emprendedor con volumen de ventas alto | * Integración con plataformas de pago con soporte transacciones a gran escala. * Sistema de detección de fraude y alertas en tiempo real. * Herramientas de CRM para gestionar relaciones con clientes y personalización de ofertas. * Funcionalidades de reporte y análisis financiero para el seguimiento del flujo de caja y la rentabilidad. |
| Usuario en situaciones de riesgo | * Opciones de pago anónimas o cifradas para la protección de la identidad y los datos financieros. * Sistema de revisión de seguridad de la cuenta con monitoreo de actividad inusual. * Opción para establecer límites de compra para prevenir el uso indebido en caso de robo de cuenta. |
| Usuario con conocimiento tecnológico limitado | * Interfaz de usuario intuitiva con navegación simple y diseño responsive. * Soporte al cliente mediante chat en vivo, chatbots , teléfono, y correo electrónico con guías paso a paso. * Tutoriales en vídeo y FAQs para las funciones más comunes de la plataforma. |
| Usuario Internacional | * Soporte multilingüe, atención al cliente, y documentación de ayuda. * Conversión automática de monedas basada en la ubicación del usuario. |
| Repartidor | * Funcionalidad de selección de pedidos por ubicación para repartidores. * Sistema de gestión de itinerarios que optimice las rutas de entrega basadas en la geolocalización, datos en tiempo real y prioridad de los pedidos. * Sistema de calificación y feedback de los usuarios para mejorar la calidad del servicio de entrega. * Notificaciones en tiempo real sobre nuevos pedidos disponibles acorde a la ubicación y preferencias del repartidor. * Herramientas de seguimiento en tiempo real para los clientes, permitiendo a los repartidores actualizar el estado de la entrega. |

* 1. **Backend**
     1. **Diagrama de Clases**
     2. **Descripción de las clases**
     3. **Diagrama de PaquetesDiagrama

        Descripción generada automáticamente**
     4. **Descripción de paquetes y componentes**
        1. **Paquete Productos:** Este paquete contiene elementos que están relacionados con los productos dentro del sistema. Incluye el nombre del producto, descripción, fotografía, categoría y orden.
        2. **Paquete GUI:** Es el paquete de la Interfaz Gráfica de Usuario, con ventanas para categorías, órdenes, billetera, transacciones, recomendaciones y forma de pago.
        3. **Paquete DB:** Este parece ser el paquete de la base de datos, donde se maneja la persistencia y eliminación de datos.
        4. **Paquete Billetera**: Contiene elementos relacionados con el método de pago.
        5. **Paquete de Utilidad:** Este paquete incluye utilidades generales del sistema como comprador, vendedor y mensajero.
  2. **Persistencia**
     1. **Diagrama de clases persistentes**
     2. **Tipo de almacenamiento**
     3. **Diagrama entidad relación**

1. **Diseño:**
   1. **Estimaciones:**
      1. ***Capacidad de manejo de transacciones:***Capacidad de procesar hasta 2,000 transacciones por segundo.
      2. ***Escalabilidad de Recursos:*** Automatización de escalado de recursos en menos de 5 minutos ante incrementos del 20% en la demanda.
      3. ***Satisfacción del usuario:*** Puntaje promedio de 4.5/5 en encuestas de satisfacción con usuarios.
      4. ***Tiempo promedio de completación de tareas comunes:*** Los usuarios nuevos completan tareas en menos de 2 minutos y medio.
      5. ***Índice de error y recuperación:*** 95% de recuperación en los errores y menos del 5% del índice de errores en tareas comunes.
      6. ***Detección de transacciones sospechosas****:* 99% las transacciones sospechosas se identifican y bloquean automáticamente.
      7. ***Tiempo medio de detección de vulnerabilidades:*** Menos de 24 horas para detección de vulnerabilidades.
      8. ***Tiempo medio de respuesta a incidentes de seguridad:*** Respuesta y mitigación en menos de 4 horas.
      9. ***Tiempo de carga de páginas:*** Tiempo de carga promedio inferior a 2 segundos en condiciones normales y menos de 4 segundos bajo carga alta.
      10. ***Procesamiento de transacciones:*** Tiempo de procesamiento y confirmación en mendos de 5 segundos.
      11. ***Cumplimiento de contraste de colores:*** 100% de cumplimiento en estándares de contraste de colores.
      12. ***Texto alternativo en elementos multimedia:*** 100% de contenido multimedia con texto alternativo adecuados.
   2. **Tecnologías**
      1. **Tecnologías Consideradas:**
         1. **Frontend:**
            1. **React**
            2. **React Native**
            3. **Vue JS**
         2. **Backend:**
            1. **Laravel**
            2. **Express**
            3. **Node.js**
         3. **DBM:**
            1. **PostgreSQL**
            2. **Mysql**
         4. **Fullstack:**
            1. **Meteor.js**
2. **Anexos**
   1. **Informe de Gestión**
      1. **Desglose Del proyecto**
         1. **Tareas:**
            1. **Preparación de Documentos**

**Resumen**

**Introducción**

* + - * 1. **Design Thinking**

**Prototipos**

**Elaborar versiones del prototipo**

**Documentar los cambios.**

* + - * 1. **Análisis**

**Requisitos Funcionales**

**Backend**

**Creación de diagrama de clases**

**Describir las clases con sus paquetes**

**Persistencia de datos**

**Elaborar diagrama de clases persistentes.**

**Seleccionar el tipo de almacenamiento de datos y justificación.**

**Diseño de diagrama Entidad-Relación y alternativas según el tipo de almacenamiento elegido.**

**Diseño**

**Selección de tecnología**

* + - 1. **Tareas por integrante;**
    1. **Informe de gestión LOGT**

|  |  |  |
| --- | --- | --- |
| Tarea | Integrante | Observaciones |
| Hoja de Presentación | Todos | Creación del documento y hoja de presentación. |
| Resumen | Todos | Síntesis clara y concisa del proyecto. |
| Introducción | Todos | Descripción completa y objetivos. |
| Prototipos | Gustavo Cruz, Carlos Valladares | Diseño y pruebas con usuarios, análisis de feedback. |
| Requisitos Funcionales y Análisis | Rodrigo Mansilla | Identificación precisa de requisitos, análisis detallado. |
| Backend y Persistencia de Datos | Paula Barillas, Brandon Reyes | Diseño técnico y selección de tecnologías de almacenamiento. |
| Selección de Tecnología | Rodrigo Mansilla | Investigación y justificación de las tecnologías elegidas. |
| Informe de Gestión y Desglose de Tareas | Rodrigo Mansilla | Organización del trabajo y seguimiento del progreso. |
| Diagrama Entidad-Relación | Paula Barillas | Diseño de la base de datos y relaciones. |
| Preparación de Documentación Final | Todos los miembros | Revisión final y ajustes. |
| Repositorio de GitHub | Todos | Configuración, manejo del repositorio y documentación. |