OLIMPIADA NACIONAL DE ETP 2024

INSTANCIA ISTITUCIONAL

PROGRAMACION

* ESCUELA DE EDUCACION TECNICA N°2
* MAR DEL PLATA
* BUENOS AIRES
* CUE DE LA ESCUELA: 060593300
* PROF RESPONSABLE: GABRIEL PIMENTEL
* PROFESORADO: ANALISTA DE SISTEMA
* CARGO DOCENTE: PROFESOR
* CORREO ELECTRONICO: [PIMENTEL.@ICLOUD.COM](mailto:PIMENTEL.@ICLOUD.COM)
* TELEFONO:2235510343
* ESTUDIANTES:
* TRAVERSO MELANIE
* CHOQUE LAUTARO
* LUCIANO DIBATISTA
* PAOLA PEROTTO
* ESPECIALIDAD PROGRAMACION
* CICLO SUPERIOR
* 7°MO AÑO
* Documento de documentación

**INDICE**

1. **RESUMEN DEL PROYECTO**
2. **REQUISITOS**

**. REQUISITOS FUNCIONALES**

**. REQUISITOS NO FUNCIONALES**

1. **DISEÑO Y ARQUITECTURA**

**. ESTRUCTURA DE LA PAGINA WEB**

**. WIREFRAMES Y MOCKUPS**

**. TECNOLOGIAS UTILIZADAS**

1. **DESARROLLO**

**. ENTORNO DE DESARROLLO**

**. ESTRUCTURA DEL CODIGO**

1. **DIAGRAMAS**
2. **IMPLEMENTACION**

**. GUIA DE INSTALACION**

1. **EXPERIENCIA DE LOS ALUMNOS**
2. **CITAS BIBLIOGRAFICAS**
3. **DISTRIBUCUION DE TAREAS**

**Descripción General**

La aplicación web "Carrito de Compra" está diseñada para facilitar el proceso de compra en línea para una compañía líder en la comercialización de productos para la práctica de deportes al aire libre. La web tiene dos tipos principales de usuarios: clientes y personal de ventas.

**Propósito:**

**- Para los Clientes:** Proporcionar una plataforma intuitiva y segura donde los clientes puedan consultar el catálogo de productos, seleccionar los artículos que desean comprar, y gestionar sus pedidos de manera eficiente.

**- Para el Personal de Ventas:** Facilitar la administración de productos y pedidos, gestionar la facturación, y llevar un control sobre los pedidos pendientes y entregados.

**Problema que Resuelve:**

**1. Facilita el Proceso de Compra:** Permite a los clientes consultar productos y realizar compras en línea, reduciendo la necesidad de interacciones físicas y mejorando la accesibilidad para los clientes que prefieren comprar desde casa.

2**. Optimiza la Gestión de Pedidos:** Proporciona una interfaz para que el personal de ventas administre los productos, verifique el estado de los pedidos, y realice entregas de manera más organizada y eficiente.

3**. Mejora la Visibilidad de Productos**: Al ofrecer una lista de productos en línea, la empresa puede llegar a un mayor número de clientes potenciales, mostrando productos detallados y actualizados en tiempo real.

**4. Gestión Eficiente de Inventario:** Permite al personal de ventas actualizar fácilmente la información del inventario y realizar un seguimiento de los productos y pedidos, reduciendo los errores manuales y mejorando la precisión.

**Objetivos del Proyecto**

**1. Desarrollar una Plataforma de Comercio Electrónico Funcional:**

- Crear una aplicación web que permita a los clientes visualizar y seleccionar productos, armar un carrito de compras y realizar pedidos con facilidad.

**2. Implementar un Sistema de Gestión de Pedidos:**

- Desarrollar funcionalidades que permitan al personal de ventas administrar productos, verificar y gestionar pedidos, registrar entregas, y actualizar el estado de las facturas.

**3. Garantizar una Experiencia de Usuario Intuitiva:**

- Diseñar una interfaz de usuario que sea fácil de navegar tanto para los clientes como para el personal de ventas, con una experiencia de compra fluida y sin fricciones.

**4. Integrar un Sistema de Validación y Seguridad:**

- Implementar medidas de seguridad para proteger la información del cliente y garantizar transacciones seguras, incluyendo validación de usuario y manejo seguro de datos de pago.

**5. Optimizar el Proceso de Administración Interna:**

- Facilitar al personal de ventas la gestión del inventario y pedidos, permitiendo una actualización rápida de la información y una visualización clara del estado de las transacciones.

**6. Proporcionar Herramientas de Soporte y Capacitación:**

- Desarrollar manuales de uso, tutoriales y cursos de capacitación para asegurar que tanto el personal de ventas como los clientes puedan utilizar la aplicación de manera efectiva.

**7. Garantizar la Escalabilidad y Mantenimiento:**

- Diseñar la aplicación para que sea escalable y fácil de mantener, permitiendo futuras actualizaciones y adaptaciones según las necesidades del negocio y el crecimiento de la base de clientes.

**Requisitos Funcionales**

**1. Registro y Autenticación de Usuarios**

**- Registro de Nuevos Clientes:** Los clientes deben poder registrarse en la aplicación proporcionando información básica como nombre, correo electrónico, y contraseña.

**- Inicio de Sesión:** Los usuarios registrados deben poder iniciar sesión usando su correo electrónico y contraseña.

**- Recuperación de Contraseña:** Implementar una funcionalidad para que los usuarios puedan recuperar o restablecer su contraseña en caso de olvido. ( A TERMINAR)

**2. Gestión del Carrito de Compras**

**- Agregar Productos al Carrito:** Los clientes deben poder agregar productos a su carrito de compras desde la página de productos.

**- Modificar Cantidades:** Permitir a los clientes modificar la cantidad de cada producto en el carrito.

**- Eliminar Productos:** Ofrecer la opción de eliminar productos del carrito antes de finalizar la compra. ( SOLO DE LA BASE DE DATOS)

**- Ver Resumen del Carrito:** Mostrar un resumen del carrito de compras con los productos seleccionados, cantidades y precios.

**3. Proceso de Compra**

**- Registro de Pedido**: Permitir a los clientes registrar sus pedidos, generando un número de pedido único.

**- Gestión de Direcciones de Entrega:** Los clientes deben poder ingresar y gestionar sus direcciones de entrega. A TERMINAR

**- Opciones de Pago:** Integrar métodos de pago seguros para procesar transacciones, como tarjetas de crédito/débito y otras herramientas de pago en línea.

**- Confirmación de Pedido:** Proporcionar una confirmación del pedido y enviar notificaciones por correo electrónico con los detalles del pedido y el estado.

**4. Gestión de Productos**

**- Carga de Productos:** El personal de ventas debe poder agregar nuevos productos al catálogo, incluyendo detalles como código de producto, descripción, y precio. ( SOLO DE LA BASE DE DATOS)

**- Actualización de Productos:** Permitir la actualización de la información de productos existentes.

**- Eliminación de Productos**: Ofrecer la opción de eliminar productos del catálogo si es necesario.

**5. Gestión de Pedidos**

**- Consulta de Pedidos:** El personal de ventas debe poder consultar la lista de pedidos pendientes y ver los detalles de cada uno.

**- Actualizar Estado de Pedidos:** Permitir al personal actualizar el estado de los pedidos (e.g., en proceso, entregado, cancelado).

**- Registro de Entregas:** Permitir registrar y gestionar la entrega de pedidos.

**6. Facturación y Reportes**

**- Visualización de Facturas:** Permitir al personal de ventas consultar las facturas pendientes y ordenarlas por fecha o cliente.

**- Generación de Reportes:** Ofrecer herramientas para generar reportes sobre ventas, inventario y estado de cuentas.

**7. Búsqueda y Filtros**

**- Búsqueda de Productos:** Implementar una funcionalidad de búsqueda para encontrar productos por nombre, código o categoría.

**- Filtros Avanzados:** Permitir a los usuarios filtrar productos por precio, categoría, y otros atributos relevantes.

**8. Soporte y Ayuda**

**- Manuales y Tutoriales:** Proveer documentación y tutoriales para ayudar a los usuarios a navegar y utilizar la aplicación.

**- Formulario de Contacto:** Incluir un formulario de contacto para consultas o soporte adicional.

**Requisitos No Funcionales**

**1. Rendimiento**

**- Tiempo de Respuesta:** La aplicación debe tener tiempos de respuesta rápidos para la carga de páginas y procesamiento de transacciones, con un tiempo de carga de menos de 2 segundos para las páginas principales.

**- Capacidad de Manejo de Carga:** La aplicación debe soportar un número alto de usuarios simultáneos y transacciones sin degradación significativa en el rendimiento.

**2. Seguridad**

**- Protección de Datos**: Implementar cifrado de datos tanto en tránsito (SSL/TLS) como en reposo para proteger la información personal y de pago de los usuarios.

**- Autenticación Segura:** Utilizar métodos de autenticación seguros, como autenticación de dos factores (2FA) para el acceso a cuentas administrativas.

**- Protección Contra Ataques:** Asegurar que la aplicación esté protegida contra ataques comunes como SQL Injection, Cross-Site Scripting (XSS), y Cross-Site Request Forgery (CSRF).

**3. Escalabilidad**

**- Escalabilidad Vertical y Horizontal**: Diseñar la arquitectura de la aplicación para permitir la escalabilidad tanto vertical (mejoras en el hardware del servidor) como horizontal (adición de más servidores).

**- Modularidad:** Implementar una arquitectura modular para facilitar la adición de nuevas funcionalidades y la integración de sistemas adicionales en el futuro.

**4. Accesibilidad**

**- Cumplimiento con Normas de Accesibilidad:** Asegurar que la aplicación cumpla con las pautas de accesibilidad web (WCAG) para garantizar que sea usable por personas con discapacidades.

**- Compatibilidad con Navegadores:** Garantizar que la aplicación sea compatible con los navegadores web más utilizados (Chrome, Firefox, Safari, Edge) y versiones recientes de estos navegadores.

**- Diseño Responsive:** La interfaz de usuario debe ser completamente responsive, adaptándose a diferentes tamaños de pantalla y dispositivos, incluidos móviles y tablets.

**Estructura de la Página Web: "Carrito de Compra"**

La estructura de la página web "Carrito de Compra" está diseñada para proporcionar una experiencia de usuario intuitiva tanto para clientes como para el personal de ventas.

**1. Arquitectura General del Sitio**

**1.1. Capas del Sistema**

- Capa Cliente (Frontend):

- Interfaz de usuario accesible desde navegadores web.

- Diseño responsivo para adaptarse a diferentes dispositivos (desktop, tablet, móvil).

- Capa Servidor (Backend):

- Procesa la lógica de negocio, maneja las solicitudes del cliente, y se comunica con la base de datos.

- Implementa la seguridad y validación de datos.

- Capa de Base de Datos:

- Almacena información de productos, usuarios, pedidos, y otras entidades relevantes.

- Proporciona acceso eficiente y seguro a los datos.

**2. Estructura de Navegación**

**2.1. Página de Inicio**

- Componentes:

- Barra de Navegación Principal: Accesos rápidos a secciones importantes como "Inicio", "Productos", "Mi Cuenta", "Carrito", "Soporte".

- Banner Promocional: Destaca ofertas especiales o novedades.

- Destacados: Productos recomendados o destacados.

- Footer: Información de contacto, enlaces a redes sociales, políticas de privacidad, y términos de servicio.

**2.2. Página de Productos**

- Componentes:

- Barra de Búsqueda: Permite a los usuarios buscar productos por nombre, código, o categoría.

- Filtros Avanzados: Opciones para filtrar productos por precio, categoría, y otros atributos.

- Lista de Productos: Muestra productos con información básica (imagen, nombre, precio, y botón para agregar al carrito).

- Paginación: Navegar por múltiples páginas de productos.

**2.3. Página de Detalle del Producto**

- Componentes:

- Imágenes del Producto: Galería de imágenes del producto.

- Descripción Detallada: Información completa sobre el producto.

- Opciones de Compra: Selector de cantidad y botón para agregar al carrito.

- Opiniones y Valoraciones: Sección para reseñas y calificaciones de usuarios.

**2.4. Página de Carrito de Compras**

- Componentes:

- Resumen del Carrito: Lista de productos añadidos, con opciones para modificar cantidades o eliminar artículos.

- Resumen de Costos: Subtotal, impuestos, y costo total estimado.

- Botón de Pago: Proceder a la página de pago para finalizar la compra.

- Código de Descuento: Campo para aplicar códigos promocionales.

**2.5. Página de Pago**

- Componentes:

- Información de Entrega: Formulario para ingresar o seleccionar una dirección de entrega.

- Opciones de Pago: Formulario para ingresar información de pago (tarjeta de crédito/débito, otros métodos).

- Confirmación del Pedido: Resumen de la compra antes de la confirmación final.

**2.6. Página de Mi Cuenta**

- Componentes:

- Perfil del Usuario: Información personal del usuario, como nombre, correo electrónico, y dirección.

- Historial de Pedidos: Lista de pedidos anteriores con detalles y estado.

- Gestión de Direcciones: Opción para agregar, editar o eliminar direcciones de envío.

- Cambio de Contraseña: Opción para actualizar la contraseña del usuario.

**2.7. Página de Gestión para el Personal de Ventas**

- Componentes:

- Panel de Productos: Opciones para agregar, editar o eliminar productos del catálogo.

- Gestión de Pedidos: Lista de pedidos pendientes y su estado, con opciones para actualizar el estado y registrar entregas.

- Facturación: Acceso a las facturas y estado de cuenta.

**2.8. Página de Soporte**

- Componentes:

- Formulario de Contacto: Permite a los usuarios enviar consultas o reportar problemas.

- Preguntas Frecuentes (FAQ): Respuestas a preguntas comunes sobre el uso del sitio y la compra de productos.

**3. Principales Componentes**

**3.1. Barra de Navegación (Header)**

- Incluye enlaces a las principales secciones del sitio y elementos de acceso rápido como el icono del carrito y el enlace a la cuenta del usuario.

**3.2. Footer**

- Contiene enlaces de interés como contacto, políticas de privacidad, términos de servicio, y redes sociales.

**3.3. Formularios de Entrada**

- Formatos de registro, inicio de sesión, y entrada de datos para pedidos y pago.

**3.4. Botones de Acción**

- Botones para agregar productos al carrito, realizar pagos, y confirmar pedidos.

**3.5. Sistema de Notificaciones**

- Mensajes emergentes o notificaciones por correo electrónico para confirmar acciones como la creación de pedidos o cambios en el estado del pedido.

**3.6. Paneles de Administración**

- Herramientas para que el personal de ventas gestione productos, pedidos, y facturación.

**WIREFRAMES Y MOCKUPS**

****

****

**TECNOLOGIAS USADAS:**

HTML ,PHP, CSS ,Y NODE.

**DESARROLLO**

**Instalación de Herramientas Básicas**

**INSTALAR Node.js**: Node.js es esencial para trabajar con JavaScript y React.

**Instalar Servidor de PHP**: Puedes usar XAMPP, WAMP o MAMP, que incluyen Apache y PHP.

**Crear un Proyecto Node.js**: Creamos una carpeta para el proyecto

**Instalamos React y sus dependencias**…

**Realización de la base de datos en mariadb…**

**Unimos la base de datos a nuestro código en react…**

**Hicimos la Gestión de Dependencias…**

**ESTRUCTURA DEL CODIGO**

**|INE-NODE**

**├── node\_modules/**

**├── public/**

**│ ├── css/**

**│ │ ├── cosocss.css**

**│ │ ├── products.css**

**│ │ └── style.css**

**│ ├── img/**

**│ │ ├── zapatillas.jpg**

**│ │ ├── zapatillas2.jpg**

**│ │ └── zapatillas3.jpg**

**│ ├── js/**

**│ │ ├── carrito.js**

**│ │ └── productos.js**

**│ ├── carrito.html**

**│ ├── check\_session.php**

**│ ├── checkin.php**

**│ ├── index.html**

**│ ├── login.html**

**│ ├── logout.php**

**│ ├── productos.html**

**│ └── registro.html**

**├── olimpiadas.sql**

**├── package-lock.json**

**├── package.json**

**└── server.js**

**DIAGRAMAS**

**DIAGRAMA DE CONTEXTO**

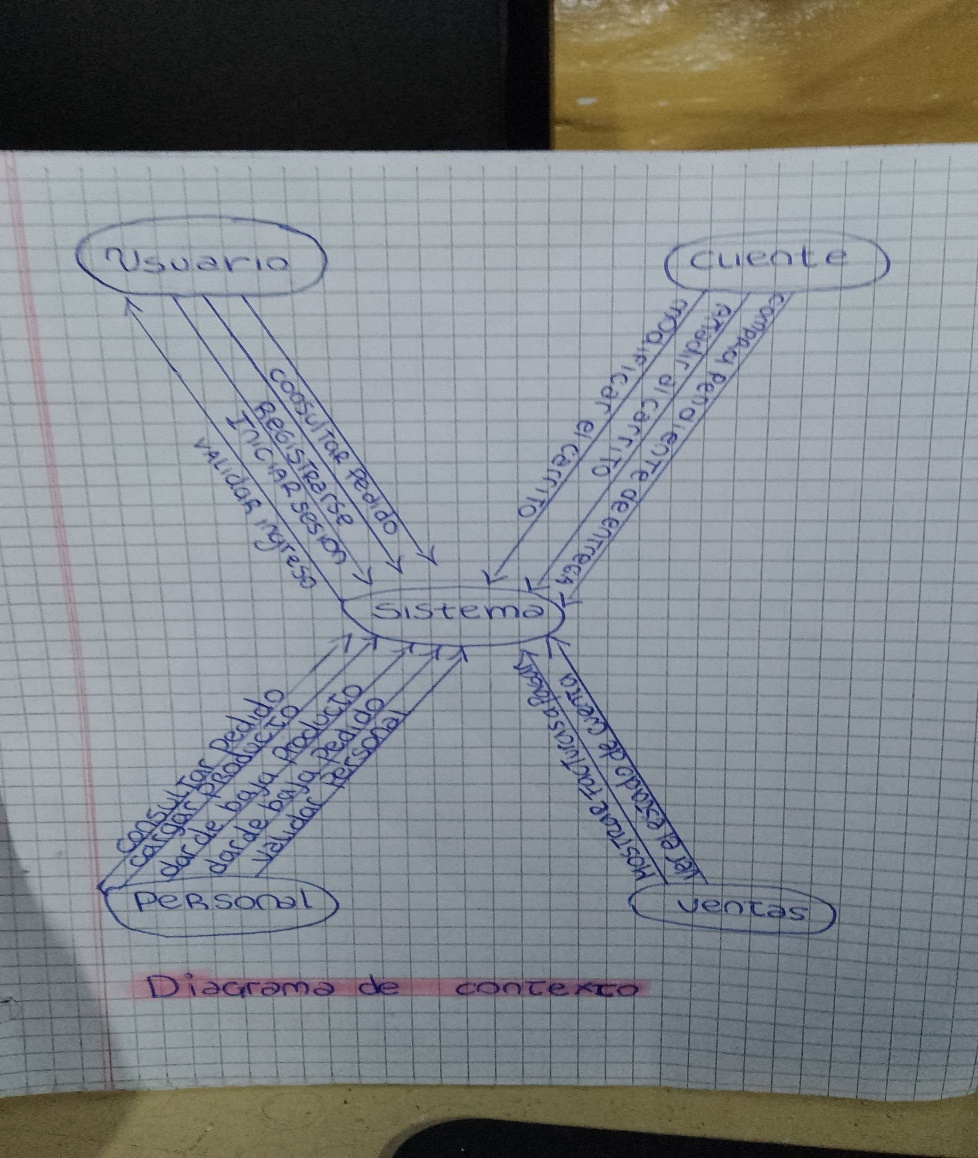
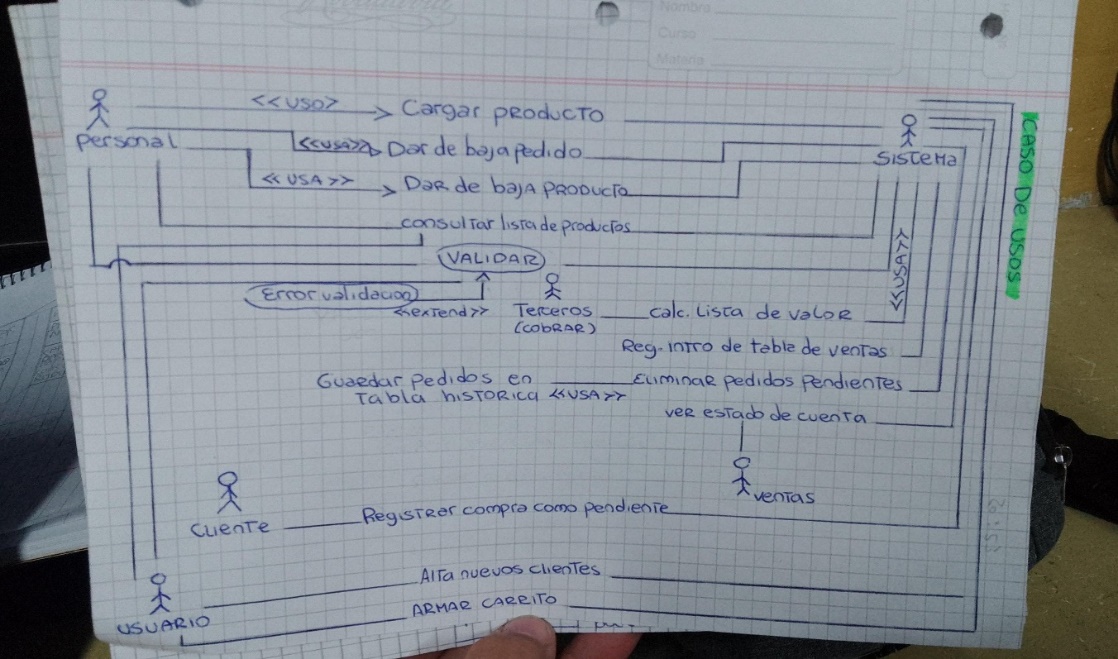
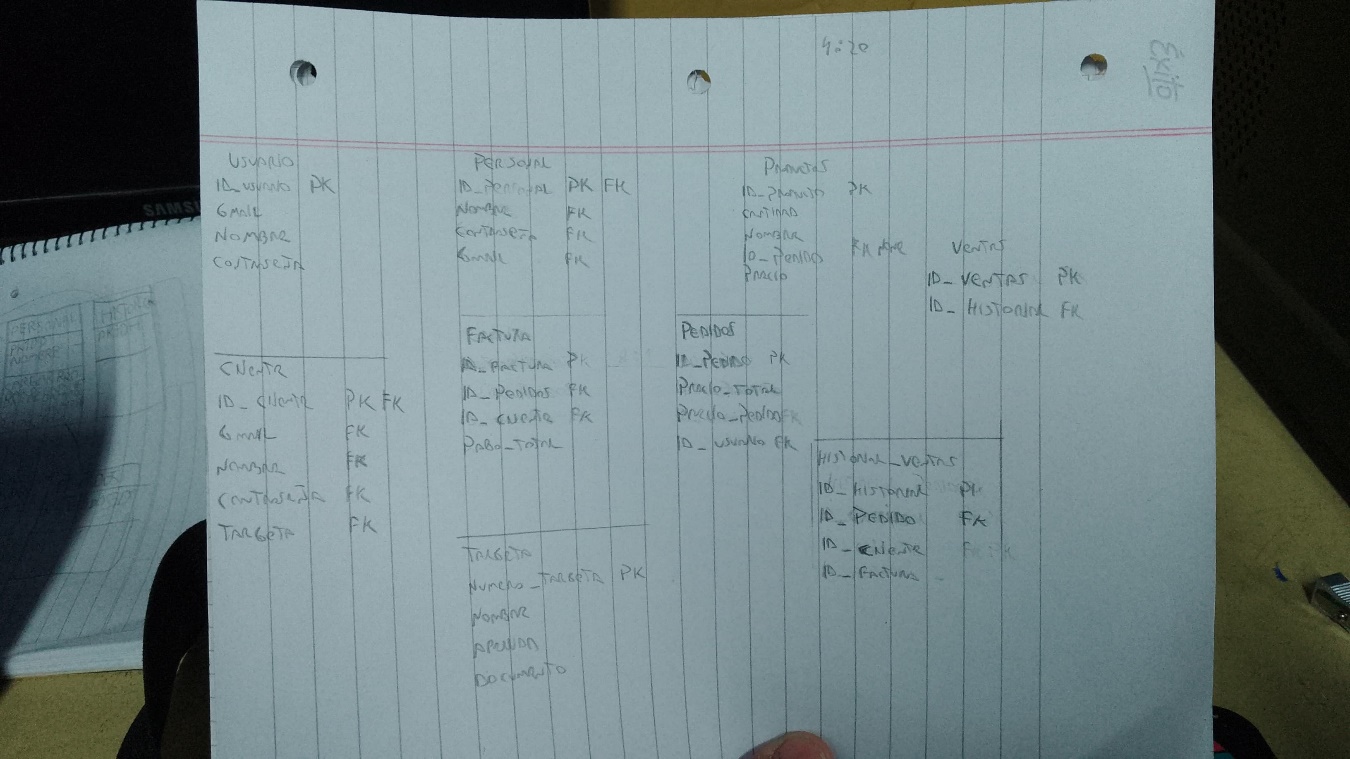


DIAGRAMA DE CASOS DE USO



**DIAGRAMA DE ENTIDAD RELACION**



**GUIA DE INSTALACION**

PARA EL MISMO SE NECESITA NODE INSTALADO EN EL SISTEMA EL XAPM Y CARGAR LA BASE DE DATOS QUE SE PROPORCIONA

**EXPERIENCIA DE LOS ALUMNOS**

**Nuestra experiencia en el proyecto de 5 días fue una aventura emocionante y desafiante que nos puso a prueba como equipo. Los cuatro integrantes del grupo nos embarcamos en una tarea que requería ser completada en un plazo más corto de lo que estamos acostumbrados, lo que nos obligó a ser creativos y eficientes en nuestra planificación y ejecución.**

**Desde el principio, supimos que iba a ser un desafío, pero estábamos decididos a hacerlo funcionar. Nos reunimos para discutir nuestra estrategia y asignar tareas, y cada uno de nosotros se comprometió a hacer su parte. La comunicación fue clave en este proyecto, y nos aseguramos de mantenernos en contacto constante para asegurarnos de que todos estuviéramos en la misma página.**

**A medida que avanzábamos en el proyecto, nos dimos cuenta de que la gestión del tiempo era crucial. Teníamos que ser cuidadosos con cómo asignábamos nuestras horas y asegurarnos de que cada uno de nosotros estaba trabajando en las tareas más importantes. Fue un desafío, pero nos las arreglamos para hacer que funcionara.**

**Una de las cosas más valiosas que sacamos de esta experiencia fue la importancia del trabajo en equipo. Aprendimos a confiar en cada uno de nosotros y a apoyarnos mutuamente en momentos de necesidad.**

**A pesar de los desafíos, nos las arreglamos para completar el proyecto en el plazo establecido y entregar un resultado de alta calidad. Fue un logro increíble, y nos sentimos orgullosos de lo que habíamos logrado. Esta experiencia nos enseñó la importancia de la flexibilidad, la creatividad y la gestión efectiva del tiempo.**

**CITAS BIBLIOGRAFICAS**

[**https://es.stackoverflow.com/questions/159529/ejecutar-codigo-php-dentro-de-javascript**](https://es.stackoverflow.com/questions/159529/ejecutar-codigo-php-dentro-de-javascript)

[**https://www.hostinger.com.ar/tutoriales/como-alojar-pagina-web-en-servidor**](https://www.hostinger.com.ar/tutoriales/como-alojar-pagina-web-en-servidor)

**DISTRIBUCUION DE TAREAS**

**LOS DIGRAMAS LOS HICIMOS LAUTARO CHOQUE Y MELANIE TRAVERSO**

**LA PROGRAMACION DE LA PAGINA:**

**LA BASE DE DATOS LA HIZO MELANIE TRAVERSO**

**EL CSS FUE POR PARTE DE LAUTARO CHOQUE**

**PARTE DEL HTML FUE PRODUCIDO POR PAOLA PEROTTO**

**EL JS LO HIZO LUCIANO DIBATISTA**

**LA DOCUMENTACION LA HIZO MELANIE TRAVERSO**