

MANUAL TÉCNICO PAN DE DIOS

JAVIER ENRIQUE CAMPOS TORRES

FICHA 2627060

INSTRUCTOR

ANDRÉS RUBIANO CUCARÍAN

ANÁLISIS Y DESARROLLO DE SOFTWARE

CENTRO DE SERVICIOS FINANCIEROS

2024

CONTENIDO

Introducción.....	3
Taxonomía.....	6
Descripción del sistema de información.....	7
Diseño técnico del sistema de información	8
Esquema o modelo de requerimientos.....	9
Software base del sistema y pre- requisitos.....	10
Componentes y estándares	11
Modelos de datos	12
Despliegue y configuración de los componentes del sistema	14
Organización de los componentes.....	16
Instalación.....	20
Resolución de problemas.....	23

INTRODUCCIÓN

Este manual técnico está diseñado para proporcionar una guía completa y detallada sobre la estructura, funcionalidades y mantenimiento de la aplicación móvil desarrollada para el registro de servicios y pedidos a domicilio. Este documento es una herramienta esencial para los desarrolladores, administradores del sistema y personal de soporte técnico, asegurando que comprendan profundamente el funcionamiento interno de la aplicación y puedan gestionarla de manera eficaz. El propósito de este manual es ofrecer un recurso exhaustivo que abarca todos los aspectos técnicos de la aplicación, incluyendo su arquitectura, configuración, operación y resolución de problemas. Al seguir las instrucciones y directrices aquí presentadas, el personal técnico podrá:

- Comprender la arquitectura y componentes clave de la aplicación.
- Realizar instalaciones y configuraciones iniciales.
- Gestionar y mantener la aplicación en entornos de desarrollo y producción.
- Solucionar problemas y realizar diagnósticos eficientes.
- Implementar actualizaciones y mejoras de manera segura.

ALCANCE DEL MANUAL

Este manual cubre una amplia gama de temas relacionados con la aplicación, tales como:

- Descripción de la Arquitectura del Sistema:

Una visión general de la estructura de la aplicación, incluyendo su backend, frontend, bases de datos y servicios auxiliares.

- Requisitos del Sistema:

Detalles sobre los requisitos de hardware y software necesarios para ejecutar la aplicación.

- Procedimientos de Instalación y Configuración:

Instrucciones paso a paso para instalar y configurar la aplicación en distintos entornos.

- Gestión y Mantenimiento:

Directrices para la administración diaria de la aplicación, incluyendo copias de seguridad, monitoreo y optimización del rendimiento.

- Solución de Problemas:

Métodos y técnicas para diagnosticar y resolver problemas comunes que puedan surgir durante la operación de la aplicación.

- Actualizaciones y Mejoras:

Procedimientos para aplicar parches, actualizar componentes y añadir nuevas funcionalidades sin afectar la continuidad del servicio.

AUDIENCIA DEL MANUAL

Este manual está dirigido a:

- Desarrolladores de Software:

Personal encargado del desarrollo y mantenimiento del código fuente de la aplicación.

- Administradores del Sistema:

Responsables de la gestión de los servidores y la infraestructura sobre la cual se ejecuta la aplicación.

- Ingenieros de Soporte Técnico:

Profesionales encargados de proporcionar soporte y resolver problemas técnicos relacionados con la aplicación.

- Consultores de IT:

Especialistas externos que puedan estar involucrados en la implementación y optimización del sistema.

ESTRUCTURA DEL MANUAL

El manual se organiza en capítulos y secciones para facilitar su uso y referencia. Cada capítulo aborda un área específica del sistema, proporcionando tanto teoría como práctica para asegurar una comprensión completa. Los capítulos principales incluyen:

1. Visión General del Sistema
2. Requisitos del Sistema
3. Instalación y Configuración
4. Arquitectura y Componentes del Sistema
5. Operación y Mantenimiento
6. Solución de Problemas
7. Actualizaciones y Mejoras
8. Anexos y Referencias

TAXONOMÍA

• Tecnologías Utilizadas:

- Lenguajes de Programación: HTML - PHP - React (JavaScript).
- Base de Datos: MySQL.
- Servidor Local: XAMPP (Apache).

• Componentes Funcionales:

- Interfaz de Usuario: Frontend desarrollado con React para una experiencia de usuario dinámica y receptiva.
- Lógica de Negocio: Backend desarrollado en HTML y PHP para procesamiento de datos y lógica empresarial.
- Base de Datos: Estructura de datos definida en MySQL para almacenar información, inventarios, usuarios, etc.
- Servidor: Configuración y gestión del servidor local mediante XAMPP.
- Gestión de entradas y salidas de productos en inventario.
- Generación de informes de calidad y análisis de datos.
- Gestión de usuarios y permisos de acceso.
- Mantenimiento y administración del sistema.

• Usuarios:

- Administradores: Responsables de la configuración y gestión del sistema.
- Usuarios Finales: Encargados de ingresar datos de calidad y gestionar inventarios.

• Entorno de Desarrollo:

- Desarrollo de Frontend: React, HTML, CSS, JavaScript.
- Desarrollo de Backend: PHP.
- Base de Datos: MySQL.
- Herramientas de Desarrollo: XAMPP para la creación de un servidor local y el ambiente de desarrollo.

• Entorno de Implementación:

- Servidor Local: XAMPP.
- Sistema Operativo: Windows, Linux, macOS (dependiendo de la preferencia del usuario).

DESCRIPCIÓN DEL SISTEMA DE INFORMACIÓN

Pan de Dios es un software para registro de servicios y pedidos a domicilio con una variedad de funcionalidades diseñadas para abordar los desafíos cotidianos que enfrentan las panaderías, aprovechando al máximo las últimas tecnologías en una aplicación web intuitiva. Está desarrollado en HTML, utilizando el paradigma de programación Orientada a objetos (POO) y una arquitectura Modelo-Vista-Controlador (MVC). Se ejecuta en un servidor local proporcionado por la herramienta XAMPP, donde se almacena la base de datos MySQL y se administra con la ayuda de un phpmyadmin, con el propósito de simplificar los servicios y domicilios en una panadería, su enfoque se centra en la optimización de procesos, la mejora de la eficiencia y el acceso a tecnología avanzada para todos los usuarios.

DISEÑO TÉCNICO DEL SISTEMA DE INFORMACIÓN

El diseño técnico del Sistema de Información de Pan de Dios es una amalgama de tecnologías de vanguardia y prácticas de desarrollo centradas en la eficiencia, la usabilidad y la escalabilidad. Este sistema, destinado a revolucionar la gestión de procesos, se basa en un enfoque integral que abarca desde la arquitectura del software hasta la experiencia del usuario final.

Arquitectura del Software

Pan de Dios se estructura sobre una arquitectura de tres capas, que separa claramente la presentación, la lógica de negocio y el almacenamiento de datos. En la capa de presentación, se emplea React para crear una interfaz de usuario moderna e interactiva que garantiza una experiencia fluida al usuario. La capa intermedia, donde reside la lógica de negocio, se implementa en PHP, aprovechando su versatilidad y rendimiento para procesar datos y ejecutar operaciones empresariales. Por último, en la capa de datos, se utiliza MySQL para el almacenamiento seguro y eficiente de la información relevante para el control de calidad y la gestión de procesos.

Seguridad y Escalabilidad

La seguridad de los datos es una prioridad fundamental en el diseño de Pan de Dios, se implementan medidas de seguridad robustas, como la encriptación de datos y el control de acceso basado en roles, para proteger la integridad y confidencialidad de la información sensible. Además, el diseño del sistema se concibe con la escalabilidad en mente, permitiendo su adaptación a medida que las necesidades del negocio evolucionan y crecen.

Entorno de desarrollo y despliegue

Pan de Dios se desarrolla en un entorno local utilizando XAMPP, que proporciona un servidor web Apache, un servidor de base de datos MySQL y los intérpretes de PHP y Perl. Esta configuración ofrece un ambiente de desarrollo cohesivo y fácil de configurar, que permite a los desarrolladores trabajar de manera eficiente en todas las fases del ciclo de vida del software. Una vez completado, el sistema puede ser desplegado en un servidor local o en la nube, según los requisitos y preferencias del cliente.

ESQUEMA O MODELO DE REQUERIMIENTOS

• **Requerimientos Funcionales:**

- El sistema debe permitir el logueo, consultar productos y realizar pedidos.
- Debe proporcionar herramientas de análisis y visualización de datos para facilitar la interpretación de la información registrada.
- Debe mantener un registro actualizado de las existencias de productos disponibles en el inventario.
- Debe facilitar la gestión de productos y clientes.

• **Requerimientos No Funcionales:**

Seguridad:

- El sistema debe garantizar la seguridad y confidencialidad de los datos almacenados, utilizando técnicas de encriptación y control de acceso.
- Debe tener mecanismos de respaldo y recuperación de datos para protegerse contra posibles pérdidas de información.

Usabilidad:

- La interfaz de usuario debe ser intuitiva y fácil de usar, permitiendo una navegación fluida y eficiente para todos los usuarios.
- El sistema debe ser compatible con diferentes dispositivos y navegadores web, garantizando una experiencia consistente en todos los entornos.

Rendimiento:

- El sistema debe ser capaz de manejar grandes volúmenes de datos y transacciones simultáneas sin degradación del rendimiento.
- Debe tener tiempos de respuesta rápidos para garantizar una experiencia de usuario ágil y sin demoras.

SOFTWARE BASE DEL SISTEMA Y PRE-REQUISITOS

El desarrollo del sistema Pan de Dios se fundamenta en una combinación de tecnologías modernas que permiten un rendimiento óptimo, una experiencia de usuario excepcional y una gestión eficaz de los datos. A continuación, se detalla el software base utilizado y los prerequisites necesarios para su instalación y funcionamiento adecuado:

Frontend con React: Una biblioteca de JavaScript de código abierto, para desarrollar la interfaz de usuario dinámica y altamente interactiva. React proporciona una estructura eficiente para construir componentes reutilizables y gestionar el estado de la aplicación de manera efectiva.

API en PHP: La lógica empresarial y el procesamiento de datos en Pan de Dios se implementan mediante una API (Interfaz de Programación de Aplicaciones) desarrollada en PHP que es un lenguaje de programación ampliamente utilizado en el desarrollo web que ofrece velocidad, versatilidad y compatibilidad con diversos sistemas de gestión de bases de datos.

Base de Datos MySQL: Pan de Dios almacena y gestiona todos los datos de los productos y los usuarios en una base de datos MySQL. MySQL es un sistema de gestión de bases de datos relacional de código abierto ampliamente utilizado, conocido por su fiabilidad, rendimiento y escalabilidad.

XAMPP como Servidor Local: Para facilitar el desarrollo y la prueba del sistema en un entorno local, Pan de Dios requiere la instalación previa de XAMPP que es una solución de servidor web de código abierto que incluye Apache, MySQL y PHP, proporcionando un entorno de desarrollo completo y listo para usar en sistemas Windows, Linux o macOS.

Requerimientos de Plataforma

Hardware:

- El sistema debe ser compatible con hardware estándar de computadoras personales, incluyendo procesadores modernos y memoria suficiente para ejecutar las aplicaciones de manera eficiente.

Software:

- Debe ser compatible con el sistema operativo Windows, Linux o macOS.
- Requiere la instalación previa de XAMPP para configurar el entorno de servidor local con Apache, MySQL y PHP.

COMPONENTES Y ESTÁNDARES

La sección de componentes y estándares describe los elementos fundamentales del software, así como las prácticas y convenciones adoptadas para garantizar su coherencia y eficacia en el desarrollo y mantenimiento del sistema. A continuación, se detallan los principales componentes y estándares utilizados.

Componentes de React:

- Pan de Dios se basa en la biblioteca de JavaScript, React para la creación de su interfaz de usuario. Esta biblioteca permite la construcción de componentes reutilizables y la gestión eficiente del estado de la aplicación, facilitando el desarrollo de una interfaz interactiva y dinámica para los usuarios.

Node.js y Gestor de Paquetes npm:

- Node.js se utiliza en Pan de Dios para ejecutar el entorno de desarrollo y para manejar las distintas dependencias y bibliotecas de React. npm (Node Package Manager) se emplea como gestor de paquetes para la instalación y gestión de las dependencias del proyecto.

Estándares de Codificación:

- Se siguen las convenciones y estándares de codificación recomendados por la comunidad de React y Node.js para garantizar la legibilidad y mantenibilidad del código. Esto incluye el uso consistente de nombres de variables, la estructura de directorios y la organización del código fuente.

Herramientas de Desarrollo:

- Se utilizan diversas herramientas de desarrollo para facilitar el proceso de desarrollo y pruebas. Esto incluye herramientas de compilación como Webpack para empaquetar los archivos de origen, así como herramientas de depuración y prueba para garantizar la calidad del software.

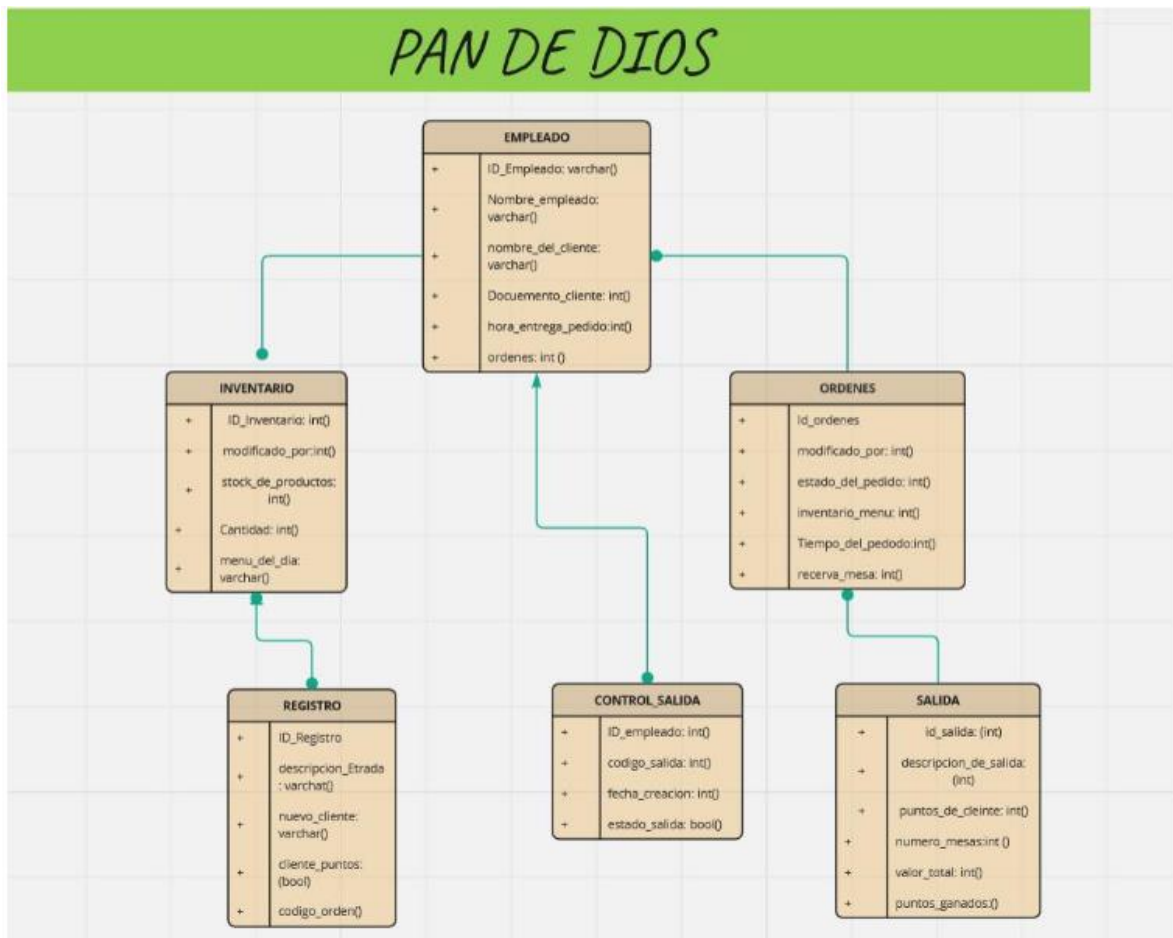
Servidor Local con XAMPP:

- Para ejecutar Pan de Dios en un entorno local, se requiere el uso de XAMPP, que proporciona un servidor web Apache, un servidor de base de datos MySQL y los intérpretes de HTML y PHP. XAMPP simplifica la configuración del entorno de desarrollo y permite probar el software en un entorno controlado antes de su implementación en producción.

MODELOS DE DATOS

Este modelo abarca desde la definición de las entidades y sus relaciones hasta la estructura física de la base de datos. A continuación, se describen cada uno de los componentes del modelo de datos de Pan de Dios:

Modelo Entidad-Relación (ER):



El modelo entidad-relación de Pan de Dios define las entidades principales y sus relaciones dentro del sistema. Las entidades pueden incluir usuarios, productos, clientes, entre otros. Las relaciones entre estas entidades se representan mediante asociaciones, como la relación entre un usuario y sus permisos de acceso, o la relación entre un producto y un cliente.

Modelo de Bases de Datos:



El modelo de bases de datos de Pan de Dios describe la estructura física de la base de datos, incluyendo la definición de tablas, columnas, índices y restricciones. Se utiliza MySQL como sistema de gestión de bases de datos relacional, y se implementan prácticas de normalización para garantizar la integridad y eficiencia de los datos almacenados. Se pueden definir vistas, procedimientos almacenados y disparadores según sea necesario para mejorar el rendimiento y la funcionalidad del sistema.

Funcionalidad y servicios ofrecidos:

Pan de Dios ofrece una amplia gama de funcionalidades y servicios específicamente diseñados para ofrecer productos y realizar servicios a domicilio.

DESPLIEGUE Y CONFIGURACIÓN DE LOS COMPONENTES DEL SISTEMA

El despliegue y la configuración de los componentes se lleva a cabo siguiendo un proceso planificado que garantiza una implementación exitosa y un funcionamiento óptimo del sistema. A continuación, se describe el proceso de despliegue y configuración de los principales componentes:

Preparación del Entorno:

- Antes de iniciar el despliegue, es importante preparar el entorno de producción adecuado. Esto incluye asegurarse de que los servidores estén configurados correctamente, con los requisitos de hardware y software necesarios para ejecutar Pan de Dios de manera eficiente.

Instalación de Servidores y Software Base:

- Se procede a instalar y configurar los servidores necesarios para el funcionamiento de Pan de Dios, incluyendo el servidor web Apache, el servidor de base de datos MySQL y el servidor de aplicaciones Node.js; Además, se instalan las dependencias del sistema operativo y se realiza cualquier configuración adicional según las necesidades del entorno.

Configuración de la Base de Datos:

Se crea una nueva base de datos en el servidor MySQL y se configuran las tablas y relaciones necesarias según el modelo de datos de Pan de Dios. Esto incluye la definición de tablas para usuarios y productos, así como la configuración de llaves primarias y foráneas.

Implementación de la Aplicación:

Se procede a la implementación de la aplicación Pan de Dios en el servidor web. Esto implica copiar los archivos de la aplicación (incluyendo archivos estáticos, como HTML, CSS y JavaScript, así como archivos del servidor, como PHP y Node.js) en el directorio raíz del servidor web y configurar cualquier archivo de configuración necesario, como archivos de entorno para configurar variables de entorno.

Configuración de Variables de Entorno:

Se configuran las variables de entorno necesarias para la aplicación, como la configuración de la base de datos (nombre de usuario, contraseña, nombre de la base de datos), claves de API, URL de servicios externos, entre otros. Estas variables de entorno se utilizan para personalizar la configuración de la aplicación según el entorno de implementación (desarrollo, pruebas, producción).

Pruebas y Validación:

Una vez completada la configuración, se realizan pruebas exhaustivas para verificar que todos los componentes del sistema funcionen correctamente. Esto incluye pruebas de integración, pruebas de rendimiento y pruebas de carga para garantizar que el sistema pueda manejar la carga esperada y proporcionar una experiencia de usuario óptima.

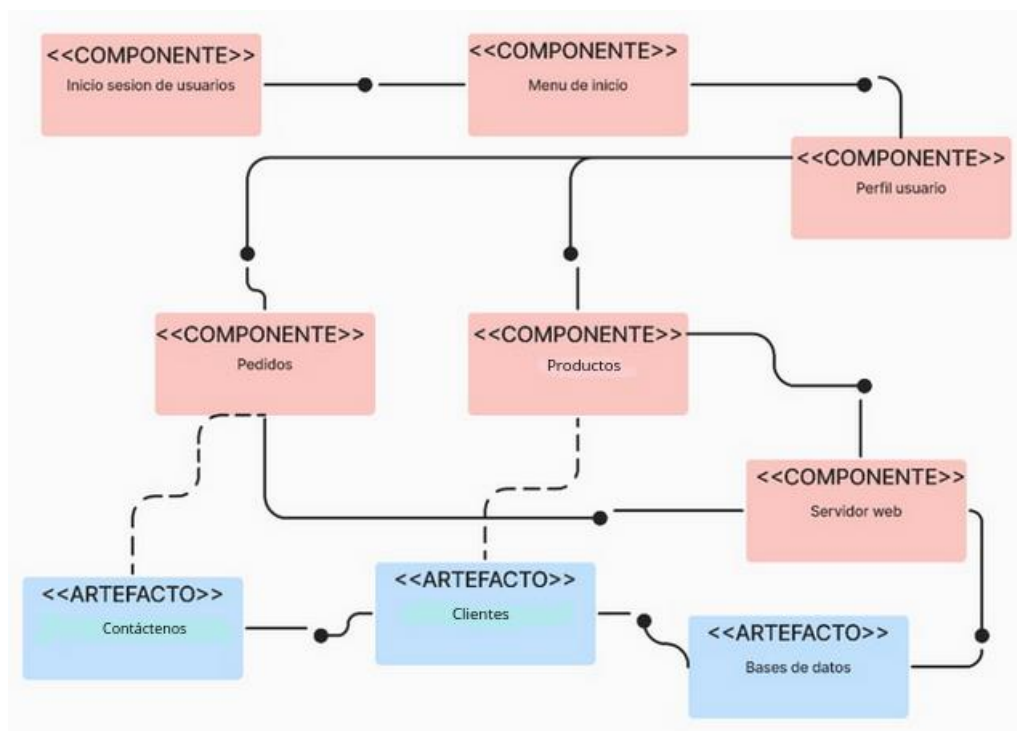
Monitoreo y Mantenimiento:

Después del despliegue inicial, se establece un proceso de monitoreo continuo para supervisar el rendimiento y la disponibilidad del sistema. Se implementan medidas de seguridad y se realizan actualizaciones periódicas para garantizar la estabilidad y la seguridad del sistema a lo largo del tiempo.

ORGANIZACIÓN DE LOS COMPONENTES

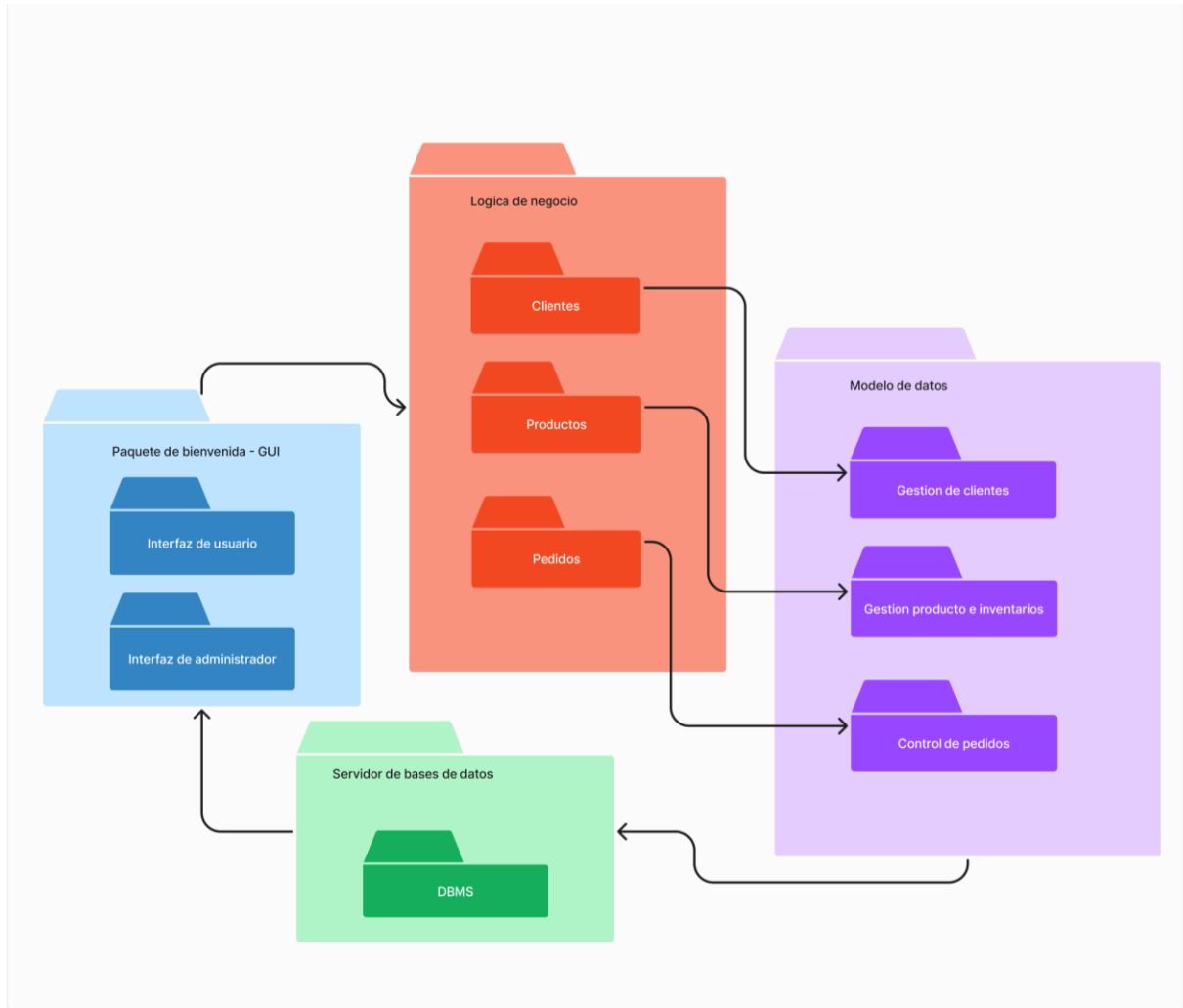
La organización de los componentes de Pan de Dios se basa en un enfoque modular y estructurado que facilita el desarrollo, la comprensión y el mantenimiento del sistema. A continuación, se describe la disposición de los componentes, junto con los diagramas que ilustran su arquitectura y distribución:

Diagrama de Componentes:



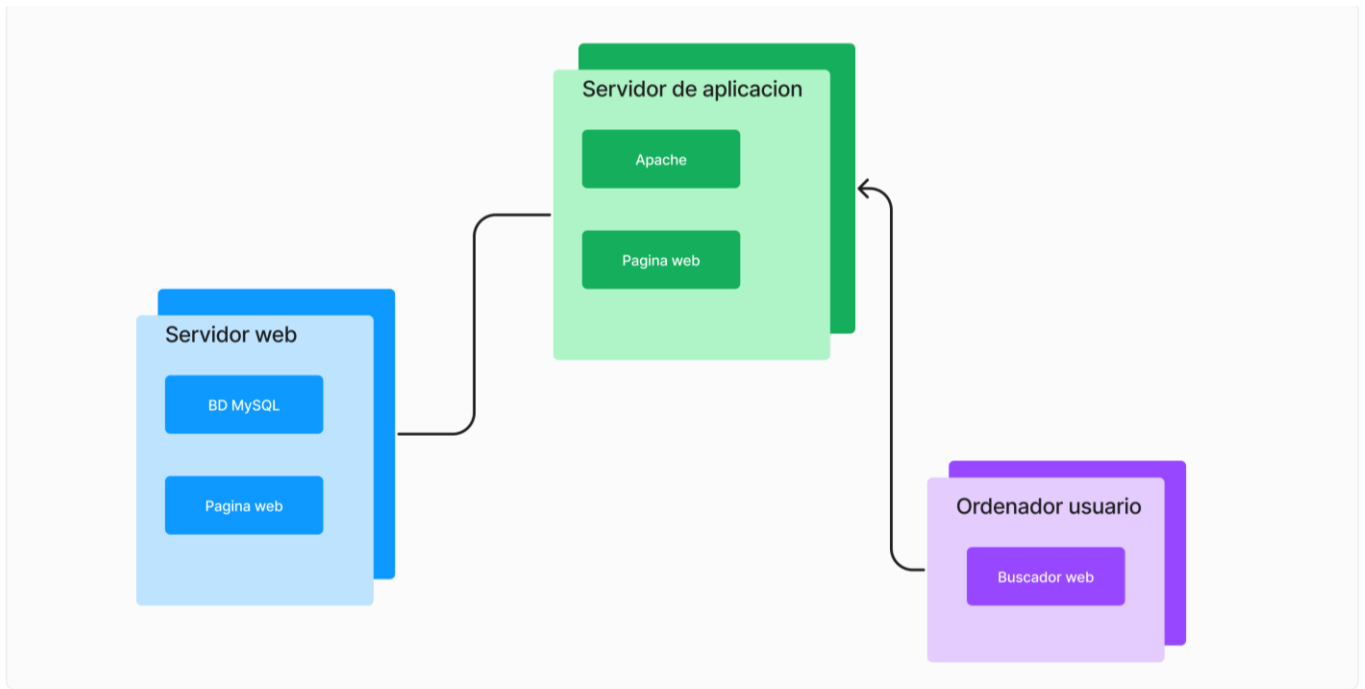
El diagrama de componentes de Pan de Dios representa las diferentes partes del sistema y cómo se relacionan entre sí.

Diagrama de Paquetes:



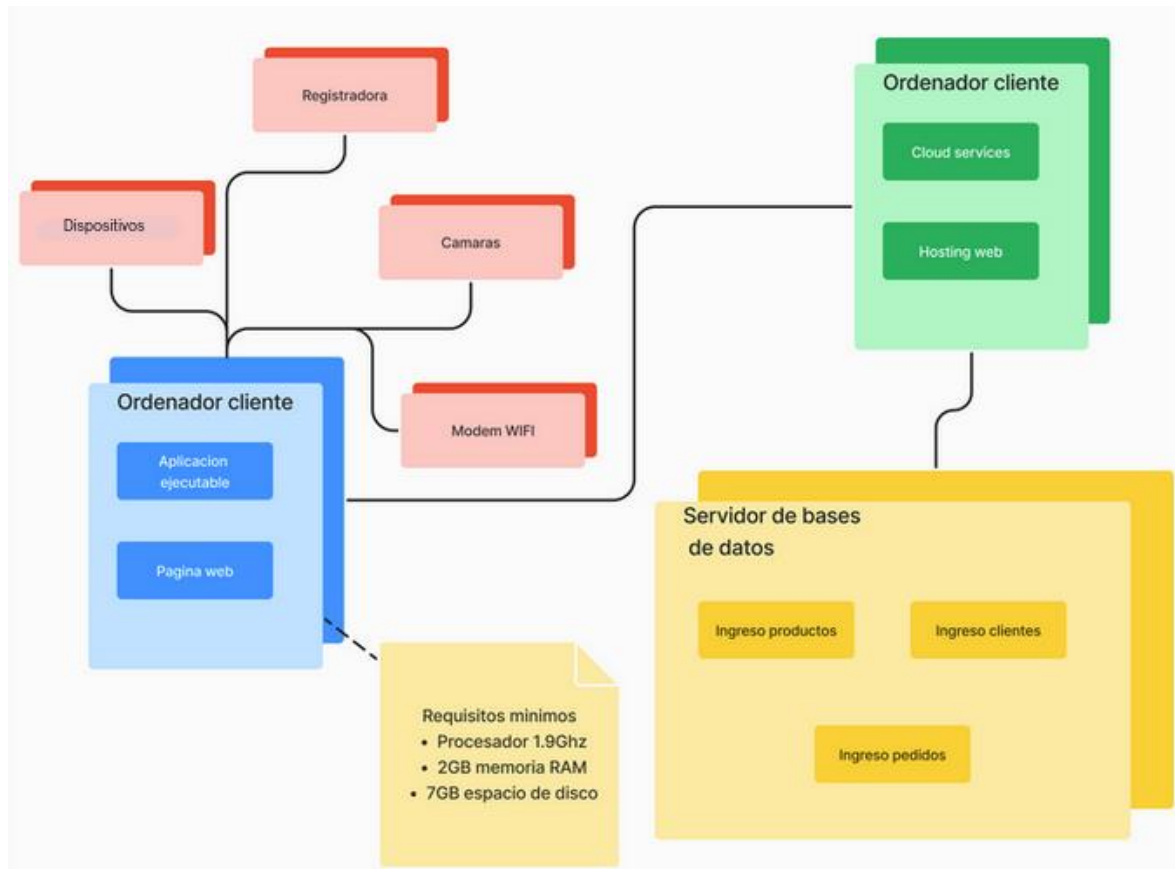
El diagrama de paquetes de Pan de Dios organiza los componentes del sistema en grupos lógicos o paquetes que representan funcionalidades relacionadas. Este diagrama ayuda a visualizar la organización modular del sistema y cómo se distribuyen las responsabilidades entre los diferentes componentes.

Diagrama de Nodos:



El diagrama de nodos de Pan de Dios muestra la infraestructura física en la que se ejecuta el sistema, incluyendo servidores, dispositivos de red y servicios externos. Este diagrama ayuda a entender cómo se distribuyen los componentes del sistema en el entorno de implementación y cómo se comunican entre sí a través de la red.

Diagrama de Despliegue:



El diagrama de despliegue de Pan de Dios describe cómo se distribuyen los componentes del sistema en el entorno de producción. Esto incluye la asignación de componentes a servidores físicos o virtuales, así como la configuración de redes y servicios externos. Este diagrama proporciona una visión general de la infraestructura de despliegue y cómo se implementa el sistema en un entorno operativo real.

INSTALACIÓN

Pre-requisitos de Instalación:

Antes de instalar Pan de Dios, asegúrese de cumplir con los siguientes prerequisites:

- Sistema Operativo compatible: Windows, Linux o macOS.
- Espacio suficiente en el disco duro para la instalación de XAMPP y el software Pan de Dios.
- Conexión a Internet para descargar los componentes necesarios.
- Permiso de administrador para instalar software en el sistema.
- Conocimientos básicos de administración de servidores y desarrollo web para configurar y mantener el entorno de desarrollo.

A continuación, se detallan los pasos para instalar Pan de Dios en un entorno local utilizando XAMPP como servidor web. Antes de comenzar, asegúrese de tener XAMPP instalado en su sistema y una conexión a Internet activa.

Clonar el Repositorio de GitHub:

- Abra una terminal o línea de comandos.
- Ejecute el siguiente comando para clonar el repositorio de GitHub de Pan de Dios:

```
git clone https://github.com/ADSOJAVIERCAMPOS/pandedios-react.git
```

Configurar XAMPP:

- Inicie XAMPP y asegúrese de que los servicios de Apache y MySQL estén activos.
- Abra el panel de control de XAMPP y haga clic en "Admin" junto a "Apache" para abrir el navegador y verificar que el servidor esté funcionando correctamente.

Configurar la Base de Datos:

- Abra un navegador web y vaya a <http://localhost/phpmyadmin/>.
- Cree una nueva base de datos llamada "pandedios" e importe el archivo pandedios.sql

Configurar la API PHP:

- En el directorio raíz de PandeDios, navegue hasta la carpeta "api" y abra el archivo "config.php" en un editor de texto.
- Actualice las credenciales de la base de datos con el nombre de usuario y contraseña de MySQL.

Instalar Dependencias:

- Abra una terminal y navegue hasta el directorio raíz de Pan de Dios (donde se encuentra el archivo "package.json").
- Ejecute el siguiente comando para instalar las dependencias de Node.js:

```
npm install node
```

Compilar y Ejecutar la Aplicación:

- Una vez completada la instalación de las dependencias, ejecute el siguiente comando para compilar la aplicación:

```
npm run build
```

Copiar Archivos al Directorio de XAMPP:

- Copie todos los archivos y carpetas generados en la carpeta "build" (dentro del directorio raíz de Pan de Dios) al directorio "htdocs" de XAMPP. Esto suele estar ubicado en "C:\xampp\htdocs" en el sistemas Windows.

Acceder a Pan de Dios:

- Abra un navegador web y vaya a http://localhost/nombre_de_la_carpeta (donde "nombre_de_la_carpeta" es el nombre de la carpeta donde copió los archivos de Pan de Dios).

RESOLUCIÓN DE PROBLEMAS

La resolución de problemas en Pan de Dios es un proceso fundamental para garantizar su funcionamiento óptimo y la satisfacción del usuario. A continuación, se describe un enfoque estructurado para identificar, diagnosticar y resolver problemas de manera eficiente:

Identificación del Problema:

- El primer paso en la resolución de problemas es identificar claramente el problema o la situación anómala. Esto puede implicar la recopilación de información sobre los síntomas reportados por los usuarios, como errores de la aplicación, mal funcionamiento de funciones específicas o pérdida de datos.

Reproducción del Problema:

- Una vez identificado el problema, es importante intentar reproducirlo en un entorno controlado. Esto puede implicar la recreación de los pasos o acciones que llevaron al problema, utilizando datos de prueba o escenarios simulados. La capacidad de reproducir el problema es crucial para comprender su causa raíz y encontrar una solución efectiva.

Análisis y Diagnóstico:

- Con el problema reproducido, se procede a realizar un análisis detallado para identificar la causa raíz. Esto puede implicar la revisión del código fuente, la inspección de registros de errores, el monitoreo del rendimiento del sistema y la realización de pruebas adicionales para descartar posibles causas.

Resolución del Problema:

- Una vez identificada la causa raíz, se desarrolla y aplica una solución adecuada para corregir el problema. Esto puede implicar la modificación del código fuente, la actualización de bibliotecas o dependencias, la configuración de parámetros de sistema, o la implementación de parches de software. Es importante realizar pruebas exhaustivas para verificar que la solución propuesta resuelve efectivamente el problema sin introducir nuevos errores.

Verificación y Validación:

- Después de aplicar la solución, se realiza una verificación exhaustiva para asegurarse de que el problema se haya resuelto correctamente. Esto puede implicar pruebas adicionales para confirmar que la funcionalidad afectada ahora funciona como se espera, así como la revisión de informes de errores para asegurarse de que el problema ya no se reproduce.

Documentación y Seguimiento:

- Una vez resuelto el problema, es importante documentar detalladamente la causa, la solución y los pasos realizados durante el proceso de resolución de problemas. Esto facilitará la referencia futura y proporcionará información valiosa para resolver problemas similares en el futuro. Además, se puede establecer un seguimiento para monitorear la estabilidad del sistema y verificar que el problema no vuelva a ocurrir en el futuro.

Comunicación con los Usuarios:

- Durante todo el proceso de resolución de problemas, es importante mantener una comunicación abierta y transparente con los usuarios afectados. Se deben proporcionar actualizaciones periódicas sobre el estado del problema y la progresión de la resolución, así como instrucciones claras sobre cualquier acción que los usuarios necesiten realizar.