

Software Para Gestión Administración Taller de Bicicletas Chamán Bikes

UNAD – Zona Occidente

Sebastián Martínez López

Esteban Rojas Barrientos

Asesor

Giovanni López Molina

Universidad Nacional Abierta y a Distancia - UNAD

Escuela de Ciencias Básicas, Tecnología e Ingeniería ECBTI

Ingeniería De Sistemas

2025

Agradecimientos

Queremos expresar nuestro más sincero agradecimiento a Dios, quien ha sido nuestra orientación y pilar de fortaleza durante este proceso. También extendemos nuestro agradecimiento al ingeniero Marco Antonio López Ospina por haber aceptado nuestra propuesta de trabajo de grado y por su constante apoyo y orientación. Su confianza en nuestras ideas nos motivó a dar lo mejor de nosotros y a profundizar en nuestra investigación.

Agradecemos al ingeniero Giovanni López Molina por la aprobación de nuestro trabajo principal. Sus valiosas retroalimentaciones y recomendaciones han sido esenciales para el progreso y la optimización de este proyecto.

Además, queremos reconocer a la licenciada Claudia Lorena Betancur por su invaluable ayuda en la gestión de la revisión. Su apoyo y dedicación facilitaron el proceso y fueron clave para alcanzar este logro.

Gracias a todos ustedes por contribuir de manera significativa a nuestra formación académica.

Resumen

El proyecto consiste en el desarrollo de un sitio web utilizando tecnologías como PHP, MySQL, HTML, CSS, JavaScript y Bootstrap para un taller de bicicletas llamado Chaman Bikes ubicado en el municipio de Guarne. Actualmente, el taller no cuenta con un software que permita organizar y visualizar de manera detallada todas las ventas de productos de bicicletas, así como los mantenimientos y reparaciones realizadas. Para optimizar la administración de Chaman Bikes, se sugiere la creación de una página web que facilite su manejo y que lleve una documentación de todas las operaciones efectuadas en el taller. A través del uso de PHP y MySQL, se implementará una base de datos para almacenar la información de los clientes, productos, servicios y ventas. Además, se utilizará HTML y CSS para diseñar una interacción sencilla y agradable de manejar, donde los usuarios podrán realizar consultas, realizar compras de productos, agendar mantenimientos y reparaciones, así como visualizar el historial de transacciones realizadas.

La implementación de Bootstrap permitirá que el sitio web sea responsivo, es decir, se adapte a diferentes dispositivos como móviles y tablets, lo que brindará una mejor experiencia de usuario. Este proyecto busca solucionar la falta de un software que permita organizar y gestionar de manera eficiente todas las ventas, mantenimientos y reparaciones realizadas en el taller Chaman Bikes. A través del desarrollo de un sitio web con las tecnologías mencionadas, se espera mejorar la productividad y eficiencia del taller, así como ofrecer un mejor servicio a los clientes.

Palabras Clave. Mantenimiento, Bicicletas, Problema, Automatización, Ventas.

Abstract

The project consists of the development of a website using technologies such as PHP, MySQL, HTML, CSS, JavaScript and Bootstrap for a bicycle workshop called Chaman Bikes located in the municipality of Guarne. Currently, the workshop does not have software that allows organizing and viewing in detail all sales of bicycle products, as well as maintenance and repairs carried out. With the aim of improving the management of Chaman Bikes, the creation of a website is proposed that allows keeping a record of all transactions carried out in the workshop. Through the use of PHP and MySQL, a database will be implemented to store customer, product, service and sales information. In addition, HTML and CSS will be used to design a friendly and easy-to-use interface, where users will be able to make inquiries, make product purchases, schedule maintenance and repairs, as well as view the history of transactions carried out. The implementation of Bootstrap will allow the website to be responsive, that is, it adapts to different devices such as mobile phones and tablets, which will provide a better user experience. This project seeks to solve the lack of software that allows efficiently organizing and managing all sales, maintenance and repairs carried out in the Chaman Bikes workshop. Through the development of a website with the aforementioned technologies, it is expected to improve the productivity and efficiency of the workshop, as well as offer better service to customers.

Keywords. Maintenance, Bicycles, Problems, Automation, Sales.

Tabla de Contenido

| | |
|--|----|
| Introducción | 14 |
| Base Teórica del Objeto de Investigación | 15 |
| Marco Teórico y Conceptual | 15 |
| Introducción a la Gestión y Control de Talleres de Bicicletas..... | 19 |
| Definición de un Taller de Bicicletas | 20 |
| Importancia de la Bicicleta | 20 |
| Mantenimiento a las Bicicletas y su Importancia..... | 21 |
| Concepto de Administración del Taller de Bicicletas..... | 22 |
| Importancia de una Buena Administración del Taller | 23 |
| Reglas Para Mejorar la Administración del Taller | 23 |
| Rol del Administrador del Taller Chaman Bikes..... | 26 |
| Descripción del Administrador | 26 |
| Funciones del Administrador | 27 |
| Generalidades Acerca de Php | 28 |
| Introducción a Php | 28 |
| Php 8 | 28 |
| Características de Php | 29 |
| Diseño de Aplicaciones Web Con Php | 32 |
| Interacción Con la Base de Datos | 34 |
| Acceso a Datos | 35 |
| Componentes de Php | 36 |
| Sistemas Cliente – Servidor | 39 |

| | |
|--|----|
| Definición..... | 39 |
| Arquitecturas Cliente / Servidor..... | 40 |
| Bases de Datos. | 41 |
| Concepto de Bases de Datos | 41 |
| Microsoft Sql Server | 42 |
| Sql..... | 43 |
| Descripción y Análisis de la Situación Problema..... | 48 |
| Breve Descripción del Taller Chaman Bikes | 48 |
| Misión..... | 49 |
| Visión | 49 |
| Valores de Chaman Bikes | 50 |
| Análisis de Resultados de Encuestas para Directivos y Empleados. | 51 |
| Evaluación De Los Hallazgos Del Sondeo Realizado a los Clientes..... | 59 |
| Software Para Gestión Administración Taller De Bicicletas Chamán Bikes | 64 |
| Justificación | 64 |
| Objetivo General | 66 |
| Objetivos Específicos..... | 66 |
| Impacto..... | 66 |
| Planteamiento del Problema..... | 67 |
| Diseño del Metodología..... | 69 |
| Tipo de Investigación | 69 |
| Metodología | 69 |
| Vida Útil Del Software. | 70 |

| | |
|---|----|
| Aplicación de Cada Etapa del Modelo en Cascada..... | 72 |
| Análisis de Requisitos | 72 |
| Funciones | 73 |
| Limitaciones..... | 74 |
| Módulos | 75 |
| Modulo Inventarios | 75 |
| Modulo Facturación | 76 |
| Consultas | 76 |
| Requisitos de Rendimiento. | 77 |
| Hardware | 77 |
| Software | 78 |
| Seguridad | 78 |
| Diseño. | 79 |
| Backlog..... | 79 |
| Requisitos Funcionales | 80 |
| Requisitos no Funcionales | 81 |
| Diagrama de Casos de Usos..... | 83 |
| Casos de Usos..... | 83 |
| Relaciones Entre Casos de Usos | 75 |
| Mockups..... | 88 |
| Logueo de Usuarios..... | 88 |
| Página Principal..... | 89 |
| Servicios Realizados | 90 |

| | |
|---|-----|
| Productos | 91 |
| Reportes..... | 92 |
| Información General | 93 |
| Prototipo de la Base de Datos | 94 |
| Diccionario de Datos..... | 95 |
| Modelo Entidad Relación..... | 110 |
| Codificación | 110 |
| Modelo de Tres Capas..... | 111 |
| Mantenimiento | 112 |
| Prueba de Funcionamiento con Jmeter | 112 |
| Anexos | 115 |
| Manual de Usuario | 115 |
| Script de la Base de Datos..... | 145 |
| Glosario..... | 150 |
| Conclusiones y Recomendaciones | 154 |
| Recomendaciones..... | 154 |
| Conclusiones | 155 |
| Referencias Bibliográficas | 158 |

Lista de Tablas

| | |
|--|----|
| Tabla 1 <i>Encuesta 1</i> | 51 |
| Tabla 2 <i>Encuesta 2</i> | 52 |
| Tabla 3 <i>Encuesta 3</i> | 53 |
| Tabla 4 <i>Encuesta 4</i> | 54 |
| Tabla 5 <i>Encuesta 5</i> | 55 |
| Tabla 6 <i>Encuesta 6</i> | 56 |
| Tabla 7 <i>Encuesta 7</i> | 57 |
| Tabla 8 <i>Encuesta 8</i> | 58 |
| Tabla 9 <i>Encuesta 9</i> | 59 |
| Tabla 10 <i>Encuesta 10</i> | 60 |
| Tabla 11 <i>Encuesta 11</i> | 61 |
| Tabla 12 <i>Encuesta 12</i> | 62 |
| Tabla 13 <i>Encuesta 13</i> | 63 |
| Tabla 14 <i>Diseño de Entrada y Salida</i> | 73 |
| Tabla 15 <i>Requisitos Básicos del Hardware</i> | 77 |
| Tabla 16 <i>Requisitos Básicos del Software</i> | 78 |
| Tabla 17 <i>Requisitos Básicos del Software en la Versión Cliente</i> | 78 |
| Tabla 18 <i>Requerimientos Funcionales</i> | 80 |
| Tabla 19 <i>Requerimientos no Funcionales</i> | 81 |
| Tabla 20 <i>Obtener Información del Cliente</i> | 96 |
| Tabla 21 <i>Obtener Detalles de Producto</i> | 97 |
| Tabla 22 <i>Comprobar Inventario</i> | 98 |

| | |
|---|-----|
| Tabla 23 <i>Factura del Proveedor</i> | 99 |
| Tabla 24 <i>Pedido al Proveedor</i> | 100 |
| Tabla 25 <i>Procesar Detalles de Inventario</i> | 101 |
| Tabla 26 <i>Generar Reportes de Reorden</i> | 102 |
| Tabla 27 <i>Elaborar Informes e Artículos Disponible</i> | 103 |
| Tabla 28 <i>Compras</i> | 104 |
| Tabla 29 <i>Detalle Factura</i> | 105 |
| Tabla 30 <i>Mantenimiento Preventivo</i> | 106 |
| Tabla 32 <i>Facturas</i> | 107 |
| Tabla 33 <i>Detalle Factura</i> | 108 |
| Tabla 34 <i>Ventas</i> | 109 |
| Tabla 35 <i>Usuarios</i> | 109 |

Lista de Figuras

| | |
|--|-----|
| Figura 1 <i>Diagrama Espina de Pez Planteamiento del Problema</i> | 68 |
| Figura 2 <i>Fases del Modelo en Cascada</i> | 72 |
| Figura 3 <i>Diagrama de Casos de Usos</i> | 82 |
| Figura 4 <i>Relación de los Casos de Usos</i> | 83 |
| Figura 5 <i>Diagrama General</i> | 85 |
| Figura 6 <i>Despachar Pedido</i> | 85 |
| Figura 7 <i>Despachar Pedido</i> | 86 |
| Figura 8 <i>Gestionar Inventario</i> | 86 |
| Figura 9 <i>Generar Reportes de Inventario</i> | 87 |
| Figura 10 <i>Logueo de Usuarios</i> | 88 |
| Figura 11 <i>Página Principal</i> | 89 |
| Figura 12 <i>Servicios Realizados</i> | 90 |
| Figura 13 <i>Servicios Realizados</i> | 91 |
| Figura 14 <i>Reportes</i> | 92 |
| Figura 15 <i>Información General</i> | 93 |
| Figura 16 <i>Prototipo MER</i> | 94 |
| Figura 17 <i>Prototipo Base de Datos</i> | 94 |
| Figura 18 <i>Modelo de Tres Capas</i> | 111 |
| Figura 19 <i>Verificación del Servidor del Programa</i> | 113 |
| Figura 20 <i>Carga Exitosa De Los Hilos</i> | 113 |
| Figura 21 <i>Descripción De Las Cargas</i> | 114 |
| Figura 22 <i>Index</i> | 115 |

| | |
|--|-----|
| Figura 23 <i>Productos Disponibles</i> | 115 |
| Figura 24 <i>Login</i> | 119 |
| Figura 25 <i>Panel de Control</i> | 122 |
| Figura 26 <i>Código de la Conexión a la Base de Datos</i> | 123 |
| Figura 27 <i>Ejemplo Consulta en Base de Datos</i> | 123 |
| Figura 28 <i>Código Verificación del Stock</i> | 124 |
| Figura 29 <i>Tabla para Almacenar Datos</i> | 124 |
| Figura 30 <i>Panel de los Productos</i> | 125 |
| Figura 31 <i>Agregar Datos</i> | 128 |
| Figura 32 <i>Panel de Compras</i> | 131 |
| Figura 33 <i>Agregar Compras</i> | 134 |
| Figura 34 <i>Agregar Factura</i> | 136 |
| Figura 35 <i>Lista de Mantenimiento</i> | 139 |
| Figura 36 <i>Panel de Reportes</i> | 139 |
| Figura 37 <i>Panel Tipos de Usuarios</i> | 140 |
| Figura 38 <i>Conexión a la Base de Datos</i> | 141 |
| Figura 39 <i>Aplicación de Bootstrap</i> | 143 |
| Figura 40 <i>Aplicación de los Estilos CSS</i> | 143 |
| Figura 41 <i>Verificar la Entradas</i> | 144 |
| Figura 42 <i>Verificar la Cantidad de los Productos</i> | 144 |
| Figura 43 <i>Script 1</i> | 145 |

Autoría

Este trabajo de grado ha sido elaborado en su totalidad por Sebastián Martínez López y Esteban Rojas Barrientos. A lo largo del proceso, hemos trabajado en conjunto para investigar, analizar y desarrollar las ideas presentadas en este documento. Nuestro enfoque se ha centrado en lo que nos ha permitido profundizar en aspectos fundamentales y contribuir con nuevas perspectivas en el campo de estudio.

Cada sección de este trabajo refleja nuestra dedicación y esfuerzo, así como nuestra capacidad para colaborar efectivamente en la solución de problemas y la exploración de respuestas y soluciones innovadoras. Estamos orgullosos del resultado final, que es un testimonio de nuestra labor conjunta y del conocimiento adquirido a lo largo de nuestra formación académica.

Introducción

Las bicicletas se han consolidado como uno de los métodos de transporte más populares debido a su practicidad, beneficios medioambientales y accesibilidad para personas de todas las edades. No solo se utilizan como medio de transporte diario, sino también para actividades recreativas y laborales. Esta versatilidad hace que el mantenimiento adecuado de las bicicletas sea esencial para asegurar su buen funcionamiento, seguridad y durabilidad. Al igual que cualquier otro vehículo, las bicicletas requieren reparaciones y ajustes periódicos, los cuales deben ser realizados por profesionales capacitados. Estos técnicos deben poseer un conocimiento profundo de los componentes clave de la bicicleta, como la reparación de pinchazos, el ajuste de frenos, dirección, transmisión y otros aspectos mecánicos fundamentales para su correcto desempeño.

Este trabajo de grado tiene como objetivo explorar las mejores prácticas en el mantenimiento de bicicletas, en el texto se abordarán textualmente las definiciones más importantes, las tecnologías implementadas para el correcto funcionamiento del software además de las encuestas realizadas a los usuarios del taller. A través de los objetivos se dictaminarán los tipos de pruebas y los aspectos más relevantes en el manejo específico de la aplicación y se analizarán las características más importantes como el mantenimiento, diseño aplicación del modelo de tres capas teniendo en cuenta los casos de usos y las necesidades funcionales como las no funcionales.

Base Teórica del Objeto de Investigación

Marco Teórico y Conceptual

Taller de bicicletas llamado Chaman Bikes ubicado en el municipio de Guarne. Actualmente carece de un software para organizar y visualizar de manera detallada las ventas, mantenimientos y reparaciones.

Objetivo

Mejorar la gestión del taller Chaman Bikes mediante la creación de un sitio web.

Permitir llevar una documentación de todas las operaciones efectuadas en el taller.

Tecnologías para Utilizar

Php. Para el desarrollo del back-end y la lógica de negocio.

- **MySQL.** Para la implementación de una base de datos que almacene la información relevante.
- **Html y Css.** Para el diseño de una interacción amigable y fácil de usar.
- **Bootstrap** Para hacer el sitio web responsive, adaptándose a diferentes dispositivos.
- **JavaScript.** Para dar funcionalidad, validaciones y características específicas a la estructura HTML

Funcionalidades del Sitio Web

- **Registro de Clientes.** Permite la gestión de datos de los clientes, que incluyen nombres y apellidos, dirección, contacto, etc.
- **Registro de Productos.** Permite la gestión de datos sobre los productos comercializados en el taller.

- **Registro de Servicios.** Permite la gestión de información de los mantenimientos y reparaciones realizadas en el taller.
- **Registro de Ventas.** Permite la gestión de información de todas las transacciones de venta de productos.
- **Consultas.** Permite a los usuarios realizar consultas sobre productos, servicios y ventas.
- **Compras de Productos.** Permitir a los usuarios realizar compras de productos a través del sitio web.
- **Agendamiento de Mantenimientos y Reparaciones.** Permite a los usuarios agendar mantenimientos y reparaciones.

Beneficios del Proyecto

- Mejora la productividad y eficiencia del taller Chaman Bikes.
- Ofrece un mejor servicio a los clientes.
- Facilita la organización y visualización detallada de todas las transacciones.

En resumen, el proyecto consiste en el desarrollo de un sitio web utilizando tecnologías como PHP, MySQL, HTML, CSS, JavaScript y Bootstrap para el taller de bicicletas Chaman Bikes. El objetivo es mejorar la gestión del taller mediante un software que permita organizar y visualizar todas las ventas, mantenimientos y reparaciones realizadas.

La implementación de un sistema basado en base de datos y una interacción amigable y responsable aseguran una mejor experiencia de usuario y beneficios tanto para el taller como para sus clientes.

En este marco teórico vamos a explorar los conceptos fundamentales necesarios para desarrollar un sitio web para un taller de bicicletas utilizando PHP como lenguaje de

programación principal. PHP es un lenguaje de programación popular, frecuentemente empleado en el desarrollo de sitios web, que proporciona una extensa variedad de características y una extensa variedad de funciones para la creación de aplicaciones web dinámicas y robustas.

Taller de Bicicletas Conceptos Clave

Antes de comenzar el desarrollo del sitio web, es importante comprender los conceptos clave relacionados con un taller de bicicletas. Algunos de ellos pueden incluir.

Servicios del taller. Enumerar los diferentes servicios que ofrece el taller, como reparaciones, mantenimiento, ajustes, personalización, etc.

Catálogo de bicicletas y productos para repuestos. Mostrar una lista de las bicicletas y repuestos disponibles en el taller, incluyendo marcas, modelos, características técnicas y precios.

Desarrollo del Sitio Web con PHP

Una vez que se han comprendido los conceptos clave, podemos comenzar a desarrollar el sitio web del taller de bicicletas utilizando PHP. Algunos puntos importantes a tener en cuenta incluyen.

Diseñar la estructura y arquitectura del sitio web. Definir las páginas necesarias y establecer la navegación entre ellas.

Hacer uso de un administrador de bases de datos compatible con PHP. Utilizar un sistema de gestión de bases de datos como MySQL para almacenar la información del catálogo de bicicletas, los clientes, las reservas, etc.

Desarrollar formularios HTML. Crear formularios para que los clientes puedan introducir sus datos personales, realizar reservas, pedidos, etc.

Lógica de negocio. Implementar la lógica del sitio web utilizando PHP. Esto puede incluir el procesamiento y validación de formularios, consultas a la base de datos, realización de cálculos y la generación de respuestas dinámicas.

Plataforma de usuario. Diseñar una interfaz intuitiva y atractiva utilizando HTML, CSS y, posiblemente, JavaScript para mejorar la usabilidad y la interactividad del sitio web.

Conclusión

En este marco teórico, hemos explorado los conceptos clave necesarios para desarrollar un sitio web para un taller de bicicletas utilizando PHP. Desde la comprensión de los servicios del taller y el catálogo de bicicletas, hasta la implementación de formularios y la lógica de negocio, estos elementos forman la base para la creación de un sitio web funcional y atractivo. Con un enfoque sólido y una planificación adecuada, se puede desarrollar un sitio web de éxito para un taller de bicicletas utilizando PHP.

Como se Aporta al Crecimiento del Sector

- **Modernización tecnológica.** La implementación de un sitio web permite al taller adaptarse a las nuevas tecnologías y evolucionar hacia una gestión digital de sus transacciones. Esto contribuye a la modernización del sector tecnológico del municipio.
- **Eficiencia operativa.** El uso de un software de gestión permitirá al taller organizar de forma más eficiente las ventas, mantenimientos y reparaciones realizadas. Esta herramienta optimiza las tareas operativas, facilita el trabajo del personal y agiliza los procesos internos, lo que se traduce en un aumento significativo de la productividad y en una mejor atención al cliente.
- **Mejora en la experiencia del cliente.** A través de la creación de una página web con una interacción sencilla y amigable para el usuario, los clientes podrán realizar consultas,

compras y agendar mantenimientos de manera rápida y sencilla. Esto mejora la experiencia del cliente y contribuye a su fidelización.

- **Accesibilidad y disponibilidad.** Al contar con un sitio web responsive, el taller podrá ser utilizado desde cualquier dispositivo con acceso a internet (móviles, tablets, ordenadores, etc.). Esto aumenta la disponibilidad de los servicios del taller y facilita el acceso a la información tanto para los clientes como para los empleados.
- **Impulso al emprendimiento y desarrollo local.** Este proyecto de grado demuestra la habilidad y creatividad de los jóvenes para innovar en el municipio de Guarne en el sector tecnológico. Mediante la implementación de soluciones tecnológicas para mejorar los negocios locales, se promueve el emprendimiento y el desarrollo económico local.

En resumen, este proyecto aporta al sector tecnológico del municipio de Guarne una mayor modernización, eficiencia y digitalización de los negocios locales, así como una mejor experiencia y servicio para los clientes. Además, impulsa el desarrollo económico local y promueve el emprendimiento dentro de la comunidad.

Introducción a la Gestión y Control de Talleres de Bicicletas

Gómez (2023) señala que el uso de la bicicleta ha aumentado, no solo como medio de transporte ecológico, sino también como herramienta recreativa y deportiva. Este crecimiento ha generado una mayor demanda de servicios de mantenimiento y reparación, lo que ha convertido la gestión moderna de estos servicios en un asunto de interés. Esta gestión no solo se centra en la satisfacción del cliente y la calidad del servicio, sino también en la administración eficiente de inventarios, recursos humanos y financieros. Las decisiones adecuadas permiten una entrega rápida y oportuna que satisface las necesidades del cliente, garantizando la sostenibilidad del negocio. Este documento abarca las mejores prácticas de gestión para un taller de bicicletas,

describiendo enfoques sistemáticos para la toma de pedidos, la gobernanza comercial y la innovación en la prestación del servicio, con el objetivo de contribuir a la rentabilidad y sostenibilidad del negocio en un entorno dinámico.

Definición de un Taller de Bicicletas

Un taller de bicicletas es un establecimiento especializado en la reparación, mantenimiento y personalización de bicicletas de diversos tipos. de montaña, de carretera, urbanas y eléctricas. El principal objetivo de estos servicios es garantizar que los ciclistas disfruten de una experiencia segura y placentera con sus bicicletas, ofreciéndoles una amplia gama de soporte técnico.

Los talleres se encargan de reparar partes de la bicicleta como frenos, cambios de marcha y otros componentes esenciales. Además, suelen ofrecer planes de mantenimiento preventivo que aseguran el correcto funcionamiento de la bicicleta a lo largo del tiempo. Muchos ciclistas también buscan mejorar el rendimiento de sus bicicletas añadiendo accesorios o componentes específicos, como luces, guardabarros, portabultos, entre otros. Estos servicios ayudan a adaptar las bicicletas a las necesidades y preferencias personales de los ciclistas (Martínez & Rodríguez, 2023).

Importancia de la Bicicleta

El uso de la bicicleta es beneficioso en múltiples aspectos. Es un medio de transporte económico, ya que no requiere combustible y tiene costos de mantenimiento relativamente bajos en comparación con otros vehículos. Además, es ecológico, al reducir las emisiones de gases contaminantes y disminuir la contaminación acústica, lo que contribuye a un entorno urbano más saludable. En las ciudades, la bicicleta es una alternativa rápida para evitar el tráfico y llegar más rápidamente al destino, convirtiéndose en una opción de transporte eficiente.

Desde el punto de vista saludable, la bicicleta fomenta un estilo de vida activo, mejorando la condición física y reduciendo riesgos asociados a enfermedades cardiovasculares. También promueve la cohesión social, ya que las actividades y eventos ciclistas reúnen a las personas, creando una comunidad activa. En resumen, la bicicleta no solo es una forma de transporte, sino que mejora la calidad de vida, favorece el bienestar personal y contribuye a la preservación del medio ambiente.

Mantenimiento a las Bicicletas y su Importancia

El mantenimiento adecuado de las bicicletas es esencial para garantizar su buen funcionamiento, prolongar su vida útil y asegurar la seguridad del ciclista. Este proceso incluye una variedad de acciones, desde tareas simples como la limpieza y lubricación, hasta revisiones más detalladas de componentes cruciales como frenos, ruedas y cambios.

Un mantenimiento regular mejora la eficiencia de la bicicleta, proporcionando una conducción más suave y segura. Mantener los frenos y el sistema de cambios en buen estado es crucial para garantizar un rendimiento óptimo, especialmente en condiciones de tráfico o en terrenos difíciles. Además, detectar y solucionar problemas menores de forma temprana previene reparaciones costosas y asegura que la bicicleta se mantenga en buen estado de funcionamiento.

El mantenimiento constante de las bicicletas también promueve una cultura de respeto por el medio ambiente y por los recursos, fomentando hábitos responsables y sostenibles entre los ciclistas. Además, contribuye a que el ciclismo sea una opción de transporte más segura, económica y ecológica.

Concepto de Administración del Taller de Bicicletas

Mantener la bicicleta en buen estado es importante para que funcione bien, dure más y lo mantenga seguro mientras conduce. Este proceso implica una variedad de acciones, desde simples tareas de mantenimiento como limpieza y engrase hasta comprobaciones detalladas de las piezas de la máquina. El mantenimiento regular es un gran obstáculo para problemas como averías en los frenos o daños en los neumáticos, manteniendo a los conductores seguros y evitando accidentes.

La importancia del mantenimiento radica en varios aspectos. En primer lugar, mejora la eficiencia de la bicicleta, mejorando la experiencia de conducción al permitir un viaje más suave y seguro. Un buen freno y una buena transmisión facilitan los cambios de marcha, algo importante en colinas o carreteras urbanas concurridas. Además, el mantenimiento regular contribuye a la economía del ciclista. Si detectas y solucionas los pequeños problemas a tiempo, no se convertirán en problemas costosos. Otro aspecto importante es la seguridad. Una bicicleta en mal estado puede ser peligrosa. Es importante revisarla y mantenerla periódicamente para mantener a todos seguros.

El mantenimiento de las bicicletas promueve una mentalidad de preservación y estima por este modo de viajar ecológico. Al mantener sus bicicletas, los ciclistas promueven una forma de vida más responsable y ecológica. El cuidado regular de las bicicletas hace que andar en bicicleta sea mejor, más seguro, más económico y ecológico (Gómez, 2023).

Importancia de una Buena Administración del Taller

La importancia de una buena gestión de una tienda de bicicletas radica en su capacidad para optimizar el rendimiento general del negocio, lo que aporta muchos beneficios a la tienda y a sus clientes. Una gestión eficaz garantiza uso lógico de los recursos, la reducción de gastos, la reducción de gastos y el incremento de ganancias. Significa Gestión Eficiente La gestión de inventario le permite tener siempre las piezas y herramientas que necesita, acortando los periodos de espera y optimizando la experiencia del consumidor.

Reglas para Mejorar la Administración del Taller

Para mejorar la gestión del comercio de bicicletas, es importante establecer procedimientos estandarizados para asegurar la consistencia y la eficacia en todos los procesos, desde la recepción de las bicicletas hasta la entrega. Mantener un inventario organizado y mantener registros estrictos de las piezas y herramientas disponibles garantiza la disponibilidad de los materiales necesarios, siempre disponibles, mientras que las habilidades técnicas continuas de los empleados y la capacitación en servicio al cliente ayudan a crear un ambiente de trabajo positivo y mejorar la calidad del servicio. La implementación de un sistema para la gestión de citas ayuda a optimizar el tiempo y reducir los tiempos de espera, y promueve la comunicación abierta entre los empleados, lo que facilita la resolución de problemas. Las revisiones de desempeño ayudan a detectar áreas para mejorar, mientras que las estrategias de marketing efectivas y los análisis financieros periódicos garantizan la viabilidad y rentabilidad a largo plazo de sus seminarios. En conjunto, estos métodos no solo optimizan el trabajo del taller, sino que también contribuyen a lograr resultados satisfactorios para proporcionar experiencia a los clientes (Pérez & López, 2024).

Definir un Programa de Revisiones por Áreas

El plan de revisión en el taller significa encontrar puntos importantes como el trabajo básico de la bicicleta, ciertas piezas, la seguridad y la apariencia, entre otros aspectos. Es importante revisar las cosas con frecuencia, como cada mes o cada tres meses, y también antes de la época de mayor actividad del año. Revisar cosas como frenos, ruedas y piezas para asegurarse de que estén todas bien. Es importante tener una capacitación para seguir las reglas establecidas, además de que es buena idea realizar revisiones constantes y la recopilación de opiniones de los clientes sobre el servicio también ayudará a mejorar el programa. El programa debe revisarse y cambiarse con frecuencia para descubrir qué puede ser mejor y cambiar la frecuencia con la que se revisa y realiza según lo que aprendamos. Este programa mantendrá bien las bicicletas y hará felices a los clientes (García & Fernández, 2024).

Establecer los Aspectos a Evaluar en Cada Sector

En la mecánica general, se deben revisar el cuadro en busca de grietas o corrosión, la transmisión para verificar el estado de la cadena, piñones y plato, y los frenos para comprobar el funcionamiento de las palancas y el desgaste de las pastillas. En los componentes específicos, es importante verificar las ruedas para asegurarse de que estén alineadas y que los radios tengan la tensión adecuada, así como revisar el estado de la suspensión, buscando fugas de aceite y comprobando su funcionamiento. En cuanto a la seguridad, se debe inspeccionar el sistema de dirección, la fijación de la tija y el sillín, y el estado de los pedales. Para la estética, es recomendable mantener la limpieza general de la bicicleta, inspeccionar daños en la pintura y verificar el estado de los accesorios como reflectores y timbres. En la revisión de neumáticos, se debe asegurar que estén correctamente inflados y revisar el desgaste y la presencia de cortes. Además, en el sistema de iluminación, es crucial comprobar que todas las luces funcionen

correctamente y que las baterías estén en buen estado. Finalmente, en el sistema de transmisión, se debe asegurar que el cambio de marchas funcione sin problemas y verificar la tensión y el desgaste de la cadena. Realizar estas revisiones de manera sistemática contribuirá a mantener las bicicletas en óptimas condiciones y garantizar la seguridad de los ciclistas (Sánchez & Torres, 2024).

Elegir un Área Simultáneamente para su Revisión

En esta área, es fundamental revisar varios componentes esenciales para asegurar el correcto desempeño de la bicicleta. Primero, se debe inspeccionar el cuadro en busca de grietas, corrosión o deformaciones, ya que esto puede comprometer la seguridad del ciclista. Luego, se debe verificar la transmisión, asegurándose de que la cadena esté en buen estado, así como los piñones y el plato; esto incluye comprobar que el cambio de marchas funcione de manera suave y eficiente. Por último, es esencial revisar los frenos. se debe comprobar el funcionamiento de las palancas, el desgaste de las pastillas y el estado de los cables, ya que un buen sistema de frenos es crucial para la seguridad.

Evaluar las Modificaciones Durante un Periodo de Tiempo, Realizar Ajustes y Correcciones

Para evaluar los cambios en un programa de revisiones por áreas en un local de reparaciones de bicicletas, es fundamental establecer indicadores de desempeño, como el número de bicicletas revisadas y la satisfacción del cliente. Durante un periodo determinado, se deben recopilar datos sobre hallazgos y reparaciones, analizando tendencias para identificar áreas problemáticas. Reuniones periódicas con el equipo permitirán discutir los resultados y proponer mejoras. Basado en este análisis, se deben realizar ajustes específicos, como modificar listas de verificación o cambiar la frecuencia de revisiones en áreas críticas. Tras implementar estos cambios, es crucial hacer un seguimiento para evaluar su efectividad y repetir el ciclo de

evaluación y ajuste, garantizando así la mejora continua del programa. Al adoptar este enfoque cíclico de evaluación y ajuste, se puede mejorar continuamente el programa de revisiones, asegurando que las bicicletas se mantengan en óptimas condiciones y que los clientes se complazcan con el servicio que han obtenido.

Registrar los Procedimientos de Ejecución de Cada Tarea que Resultaron en los Mejores

Resultados

Primero, se deben identificar todas las tareas realizadas en cada área de revisión, como la inspección del cuadro o el ajuste de frenos. A continuación, se ofrece una descripción detallada de cada procedimiento, incluyendo los pasos a seguir, las herramientas necesarias y el tiempo estimado para completarlas. Es importante documentar los resultados obtenidos, especificando qué funcionó bien y qué no, así como registrar las mejoras implementadas en base a la retroalimentación. Además, se deben crear guías de capacitación para asegurar que todo el personal esté familiarizado con los procedimientos más efectivos. Finalmente, se recomienda establecer un calendario para revisar y actualizar la documentación regularmente, asegurando que refleje las prácticas más recientes y efectivas, y que esté disponible en un formato accesible para el equipo.

Rol del Administrador del Taller Chaman Bikes

Descripción del Administrador

El administrador de un local de reparaciones de bicicletas es el individuo encargado de la administración y dirección del establecimiento. Su presencia es fundamental para el funcionamiento del taller, creando un ambiente en el que se facilitan las interacciones entre el personal y los clientes. Esta figura representa el punto de conexión entre diferentes elementos del negocio, contribuyendo a su desarrollo y adaptación a las necesidades del mercado. Su enfoque

se centra en el bienestar del local e intensificar las relaciones con la comunidad de ciclistas para consolidar las relaciones con la comunidad de ciclistas.

Funciones del Administrador

Las funciones más básicas del administrador son las siguientes

- Supervisar y coordinar al equipo de mecánicos y personal de atención al cliente.
- Resolver consultas y quejas de los clientes, garantizando una experiencia positiva.
- Monitorizar y gestionar el stock de piezas y suministros necesarios para las reparaciones.
- Elaborar presupuestos y mantener un seguimiento de los costos e ingresos del taller.
- Desarrollar tácticas de marketing para captar nuevos consumidores y fidelizar a los existentes.
- Organizar y facilitar la formación continua del equipo en técnicas de reparación y atención al cliente.
- Hay que asegurar que las herramientas y equipos de trabajo estén en buen estado y sean seguros de usar.
- Garantizar que el local cumpla con las regulaciones de salud, protección y medio ambiente.
- Realizar evaluaciones periódicas del desempeño del personal, ofreciendo retroalimentación y reconocimiento.

Generalidades Acerca de Php

Introducción a Php

Es un lenguaje de programación, principalmente empleado para el desarrollo de sitios web. Permite a los desarrolladores crear sitios web dinámicos y aplicaciones interactivas. PHP se puede integrar fácilmente con HTML, lo que facilita la creación de contenido dinámico. Además, es compatible con varias bases de datos, lo que lo convierte en una opción popular para la gestión de datos en línea. Gracias a su amplia comunidad, PHP cuenta con muchos recursos, bibliotecas y frameworks que ayudan a simplificar el desarrollo. Su flexibilidad y facilidad de uso lo han posicionado como uno de los lenguajes más usados en la creación de aplicaciones web en todo el mundo (PHP.net, 2024).

Php 8

Php 8, lanzado en noviembre de 2020, representa una evolución significativa respecto a sus versiones anteriores, no solo por sus mejoras sintácticas, sino también por el impacto directo que tiene en el rendimiento del desarrollo web moderno. Una de las innovaciones más destacadas es el compilador JIT (Just-In-Time), que permite ejecutar código más rápidamente al compilarlo en tiempo real. Si bien JIT no siempre mejora el rendimiento en aplicaciones web tradicionales como lo han señalado diversos benchmarks—, su inclusión sienta las bases para futuros desarrollos más intensivos en cálculos, acercando PHP a lenguajes como Java o C# en términos de velocidad de ejecución.

Otro avance relevante es la incorporación de los tipos de unión, que aportan mayor flexibilidad sin sacrificar el control de tipos, facilitando el diseño de funciones más robustas y seguras. A su vez, la introducción de atributos (similar a anotaciones en otros lenguajes como Java) mejora la legibilidad del código y la organización de metadatos, ofreciendo ventajas

significativas en frameworks modernos como Symfony o Laravel. Finalmente, el operador nullsafe resuelve de forma elegante el manejo de valores nulos, una fuente común de errores en versiones anteriores.

En conjunto, estas mejoras no solo responden a necesidades técnicas, sino que también reflejan una madurez en el lenguaje, alineándolo con estándares actuales de programación. Desde un enfoque académico, estas transformaciones pueden analizarse como parte de una tendencia más amplia hacia la tipificación fuerte, la eficiencia en ejecución y la escritura de código más limpio, que son pilares de la ingeniería de software contemporánea. (PHP.net, 2024).

Características de Php

- Es un lenguaje de programación de código abierto y gratuito, lo que facilita su uso a los desarrolladores acceder al código fuente, modificarlo y adaptarlo a sus necesidades específicas. Esta característica fomenta la colaboración y la innovación, ya que una gran comunidad de desarrolladores contribuye constantemente a su mejora.
- Tiene una alta compatibilidad con diversos sistemas para la administración de bases de datos, tales como MySQL, PostgreSQL y SQLite. Esta habilidad simplifica la elaboración de aplicaciones web que necesitan guardar, recuperar y manipular datos, permitiendo a los desarrolladores construir sistemas robustos y dinámicos.
- Una de las ventajas más notables de PHP es su capacidad para integrarse con facilidad con HTML. Esto posibilita que los programadores incrusten directamente código PHP en archivos HTML, creando contenido dinámico que puede cambiar en función de la interacción del usuario, sin la necesidad de múltiples archivos o scripts complicados.

- Es compatible con diversas plataformas, incluidos sistemas operativos como Windows, Linux y macOS. Esta versatilidad permite a los programadores trabajar en diferentes entornos y facilita la implementación de aplicaciones en servidores web variados.
- Dispone de una extensa y dinámica comunidad de programadores que aportan a su progreso, lo que se refleja en una gran cantidad de documentación, foros de debate y recursos en línea que simplifican el aprendizaje y la solución de problemas, permitiendo que los desarrolladores emergentes puedan obtener ayuda de manera sencilla.
- Existen numerosos frameworks y bibliotecas disponibles para PHP, como Laravel, Symfony y CodeIgniter. Estos frameworks proporcionan estructuras predefinidas y herramientas que agilizan la construcción de aplicación en lugar de construir desde cero.
- Ofrece funciones integradas para manejar sesiones y cookies de manera sencilla, la autenticación. Esto es crucial para aplicaciones web que requieren un seguimiento de los usuarios y su actividad.
- Aunque la seguridad de una aplicación depende de su implementación, PHP ofrece diversas características y prácticas recomendadas que ayudan a proteger las aplicaciones, como la validación de entradas y mecanismos para prevenir inyecciones SQL. Con el uso adecuado, estas herramientas pueden contribuir significativamente a la seguridad de las aplicaciones web. (PHP.net, 2024).

Beneficios de Php

- La estructura de Php es intuitiva y similar a los demás lenguajes de programación populares como C y Perl, lo que facilita su aprendizaje para nuevos desarrolladores. Esto lo hace una alternativa atractiva para los novatos en la programación de aplicaciones Web.
- Gracias a su fácil usabilidad y a la disponibilidad de frameworks, PHP permite un desarrollo más rápido de aplicaciones web. Los desarrolladores pueden construir prototipos y aplicaciones completas en menos tiempo, lo que es especialmente beneficioso en entornos de trabajo ágil.
- Con la introducción de características como el compilador JIT en PHP 8, el rendimiento del lenguaje ha mejorado considerablemente. Esto significa que las aplicaciones desarrolladas en PHP pueden ejecutar procesos complejos de manera más rápida y eficiente, ofreciendo una mejor experiencia al usuario.
- Al ser de código abierto y contar con una amplia gama de herramientas y recursos gratuitos, PHP puede ayudar a reducir los costos de desarrollo. Las empresas pueden utilizar PHP para crear aplicaciones potentes sin incurrir en altos gastos de licencias de software.
- PHP es un lenguaje versátil que se ajusta con facilidad a diversas clases de proyectos, desde simples sitios web hasta aplicaciones web complejas. Esta flexibilidad lo hace una elección perfecta para una extensa gama de aplicaciones y necesidades de desarrollo.
- PHP facilita la creación y consumo de APIs, lo que es esencial en la era actual, donde muchas aplicaciones requieren interoperabilidad y conexión con otros servicios. Esto permite a los desarrolladores construir aplicaciones modernas que se integren fácilmente con otras plataformas y servicios.

- PHP es capaz de manejar aplicaciones de gran tamaño y puede escalar para cubrir el aumento de la demanda. Las empresas pueden comenzar con aplicaciones más pequeñas y expandirse a medida que crecen, sin necesidad de cambiar de tecnología. (Php.net, 2024).

Diseño de Aplicaciones Web con Php

La creación de aplicaciones web utilizando Php requiere de una serie de aspectos que abarcan desde el diseño inicial hasta la implementación y conservación de la aplicación. A continuación, se detallan las fases y elementos esenciales a considerar durante el proceso. (PHP.net, 2024).

Planificación y Requisitos. Antes de comenzar a codificar, es fundamental definir los objetivos de la aplicación y comprender las necesidades de los usuarios. Esto incluye realizar entrevistas, encuestas o estudios de mercado para recopilar información sobre las funcionalidades requeridas, el público objetivo y las expectativas de rendimiento. Documentar estos requisitos ayudará a guiar el desarrollo y asegurará que se satisfagan las necesidades del usuario. (DesarrolladoresWeb.com, 2024).

Arquitectura de la Aplicación. Una vez que se han definido los requisitos, el siguiente paso es diseñar la arquitectura de la aplicación. Esto implica decidir cómo se estructurará el código, cómo se manejarán las bases de datos y cómo se comunicará la aplicación con el servidor. Se deben considerar patrones de diseño como MVC (Modelo-Vista-Controlador), que separa la lógica de la aplicación, la presentación y la gestión de datos, lo que facilita el mantenimiento y la escalabilidad. (DesarrolladoresWeb.com, 2024).

Elección de Tecnologías y Herramientas. Además de Php, es importante elegir otras tecnologías que complementen el desarrollo. Esto puede incluir sistemas de gestión de bases de datos como MySQL o PostgreSQL, así como frameworks de Php como Laravel o Symfony, que ofrecen herramientas y funcionalidades que aceleran el desarrollo. También es recomendable considerar el uso de HTML, CSS y JavaScript para el diseño del front-end y la interacción con el usuario.

Desarrollo del Back-End. El desarrollo del back-end en Php implica la creación de scripts que gestionen la lógica de la aplicación. Esto incluye la conexión a la base de datos, la validación de entradas del usuario y la implementación de funciones de negocio. Es importante seguir las mejores prácticas de programación, como la utilización de funciones y clases reutilizables, para garantizar un código limpio y mantenible (Tutsplus.com, 2024).

Diseño del Front-End. El diseño del front-end se centra en la creación de una interacción de usuario funcional y atractiva. Haciendo uso de HTML, CSS y JavaScript., se debe garantizar que la aplicación sea intuitiva y accesible. También se puede considerar el uso de frameworks de CSS como Bootstrap para facilitar el diseño responsivo, asegurando que la aplicación se vea bien en dispositivos de diferentes tamaños (Tutsplus.com, 2024).

Pruebas y Validación. Antes de lanzar la aplicación, es crucial realizar pruebas exhaustivas para detectar y corregir errores. Esto abarca evaluaciones unitarias, evaluaciones de integración y evaluaciones de usuario. Se debe verificar que todas las funcionalidades trabajen como se esperaba y que la aplicación sea segura, evitando vulnerabilidades comunes como inyecciones SQL y ataques XSS.

Despliegue y Mantenimiento. Una vez que la aplicación ha sido probada y está lista para su lanzamiento, se debe implementar en un servidor web. Es importante elegir un entorno de alojamiento adecuado que soporte Php y el sistema de gestión de bases de datos utilizado. Después del despliegue, el mantenimiento regular es esencial para asegurar que la aplicación siga siendo segura, eficiente y relevante. Esto incluye actualizaciones de seguridad, mejoras de rendimiento y la implementación de nuevas características basadas en la retroalimentación de los usuarios.

Documentación. Finalmente, es crucial documentar todo el proceso de desarrollo. Esto incluye la documentación del código, guías de usuario y manuales de mantenimiento. La documentación ayuda a otros desarrolladores a entender la aplicación y facilita el proceso de incorporación de nuevos miembros al equipo.

Interacción con Base de Datos

La interacción con bases de datos en Php es fundamental para desarrollar aplicaciones web dinámicas y funcionales. El primer paso para esta interacción es establecer una conexión, para lo cual PHP ofrece diversas extensiones, siendo MySQLi y PDO (Php Data Objects) las más comunes. MySQLi proporciona una interacción mejorada para bases de datos MySQL, mientras que PDO permite trabajar con múltiples tipos de bases de información. Tras la configuración de la conexión, es posible realizar consultas SQL para realizar operaciones de selección, adición, actualización y supresión de datos. Por ejemplo, una consulta puede emplearse para recuperar datos de una tabla, o emplear sentencias preparadas para insertar o actualizar datos de manera segura, previniendo inyecciones SQL. Es crucial manejar adecuadamente los errores al interactuar con la base de datos, utilizando las herramientas que PHP ofrece para gestionar excepciones y detectar problemas en las consultas. Aunque PHP cierra automáticamente las

conexiones al finalizar el script, es recomendable cerrarlas manualmente en aplicaciones más complejas para liberar recursos de manera eficiente. En conjunto, la interacción con bases de datos en PHP permite a los desarrolladores crear aplicaciones robustas y eficientes que gestionan y presentan datos de manera efectiva (W3Schools, 2024).

Acceso a Datos

El acceso a datos en PHP es un elemento crucial en la creación de aplicaciones web, facilitando la interacción con bases de datos para guardar, recuperar y manejar datos. PHP posibilita la vinculación a varias bases de datos, destacando MySQL como una de las más habituales. (W3Schools, 2024).

Conexión a Bases de Datos

PHP ofrece varias extensiones para conectarse a bases de datos, como MySQLi y PDO (PHP Data Objects). MySQLi es específico para MySQL y ofrece una interacción mejorada, mientras que PDO proporciona una forma más flexible de trabajar con diferentes sistemas de gestión de bases de datos. La conexión generalmente implica especificar el servidor, el nombre de usuario, la contraseña y la base de datos (W3Schools, 2024).

Consultas y Manejo de Resultados. Una vez establecida la conexión, se pueden realizar consultas SQL para recuperar datos. PHP permite ejecutar consultas y manejar los resultados de manera sencilla, proporcionando métodos para obtener datos en formatos asociativos o numéricos. Esto facilita la presentación de información en las aplicaciones web (W3Schools, 2024).

Operaciones de Modificación. Además de consultas de selección, PHP facilita la ejecución de operaciones de adición, actualización y supresión de datos. Estas operaciones son esenciales para preservar la integridad y pertinencia de los datos guardados. Las sentencias preparadas se utilizan a menudo para mejorar la seguridad y prevenir inyecciones SQL (W3Schools, 2024).

Manejo de Errores. Implementar un manejo adecuado de errores es importante al trabajar con bases de datos. PHP proporciona mecanismos para detectar y gestionar problemas en las consultas, lo que asiste a los programadores en la detección y rectificación eficaz de errores. (W3Schools, 2024).

Cierre de Conexiones. Aunque PHP cierra automáticamente las conexiones al finalizar el script, es recomendable cerrarlas manualmente en aplicaciones más complejas para liberar recursos de manera adecuada (W3Schools, 2024).

Componentes de PHP

PHP es un lenguaje de codificación muy empleado en el desarrollo de sitios web, y su arquitectura se compone de varios elementos fundamentales que facilitan la creación de aplicaciones dinámicas y eficientes. A continuación, se describen los componentes clave de PHP.

- **Motor de Ejecución.** El motor de ejecución de PHP es el núcleo del lenguaje. Se encarga de interpretar el código PHP y ejecutar las instrucciones, transformando el código en HTML que se envía al navegador del usuario. Este motor permite la ejecución de scripts en el servidor, lo que resulta en contenido dinámico en las páginas web.

- **Extensiones.** PHP tiene una arquitectura modular que permite agregar funcionalidades mediante extensiones. Estas extensiones pueden ofrecer soporte para diferentes bases de datos, servicios web, manipulación de imágenes, y mucho más. Algunas de las extensiones más comunes incluyen.

- **MySQLi.** Permite la conexión y manipulación de bases de datos MySQL.

- **PDO (PHP Data Objects).** Ofrece una interacción más flexible para trabajar con múltiples bases de datos.

- **cURL.** Facilita la realización de solicitudes a otros servidores a través de diversos protocolos, como HTTP y FTP.

- **Funciones Integradas.** PHP incluye una amplia gama de funciones integradas que permiten realizar diversas operaciones sin necesidad de escribir código adicional. Estas funciones abarcan tareas como manipulación de cadenas, gestión de fechas y horas, tratamiento de arrays, y operaciones matemáticas, entre otras.

Bibliotecas y Framework

Según TutsPlus. (2024) para simplificar y acelerar el desarrollo, existen numerosas bibliotecas y frameworks disponibles para PHP. Estas herramientas ofrecen estructuras y componentes predefinidos que facilitan la creación de aplicaciones complejas. Algunos de los frameworks más populares son.

- **Laravel.** Conocido por su sintaxis elegante y características avanzadas como el enrutamiento y la gestión de bases de datos.
- **Symfony.** Ofrece una gran flexibilidad y es ideal para aplicaciones empresariales complejas.
- **CodeIgniter.** Un framework ligero que permite un desarrollo rápido y eficiente.

Sistema de Plantillas

Los sistemas de plantillas son componentes que permiten separar la lógica de negocio del diseño de la interacción de usuario. Esto facilita el mantenimiento del código y mejora la legibilidad. Algunos sistemas de plantillas populares son.

- **Smarty.** Permite crear plantillas HTML que se integran con PHP de manera eficiente.
- **Twig.** Un motor de plantillas moderno que se utiliza frecuentemente en aplicaciones Symfony.

Manejo de Errores y Excepciones

PHP dispone de un sólido sistema de gestión de errores y excepciones que facilita a los programadores la administración de problemas de forma eficaz. Esto abarca la opción de modificar la manera en se corrigen los errores y realizar excepciones personalizadas para situaciones específicas.

- **API y Servicios Web.** PHP facilita la creación y consumo de APIs y servicios web, lo que permite la integración con otras aplicaciones y plataformas. Esto es esencial para el desarrollo de aplicaciones modernas que requieren interoperabilidad.

- **Gestión de Sesiones y Cookies**

PHP incluye funciones integradas para gestionar sesiones y cookies, lo que permite mantener el estado de las aplicaciones web y manejar la autenticación de usuarios de manera sencilla y segura.

Sistemas Cliente – Servidor

Definición

Los sistemas cliente-servidor representan una arquitectura clave en la informática moderna, estructurando la interacción entre proveedores de recursos o servicios, conocidos como "servidores", y los solicitantes de estos recursos, denominados "clientes". Este modelo es omnipresente en aplicaciones web, bases de datos y redes, y constituye la base para la mayoría de las interacciones de datos en línea. En un sistema cliente-servidor, el cliente realiza peticiones a través de una interfaz, como una aplicación o un dispositivo que interactúa directamente con el usuario, mientras que el servidor, una máquina que aloja y gestiona los recursos, procesa dichas solicitudes y envía las respuestas adecuadas.

Desde una perspectiva técnica, uno de los aspectos más notables de este modelo es la separación de responsabilidades, lo que permite que los clientes se enfoquen en la interfaz de usuario y la experiencia, mientras que los servidores gestionan el procesamiento intensivo de datos y la administración de recursos. Esta división mejora la escalabilidad y la eficiencia en la gestión de grandes volúmenes de información, permitiendo que los servidores manejen múltiples peticiones simultáneamente de manera eficiente. Sin embargo, también presenta ciertos desafíos,

como la dependencia de la conectividad de red, lo que significa que si la red experimenta latencia o fallos, la interacción cliente-servidor se ve afectada.

En comparación con otras arquitecturas, como el modelo peer-to-peer (P2P), donde los nodos actúan tanto como clientes como servidores, el modelo cliente-servidor ofrece una mayor centralización y control sobre los recursos y la seguridad, aunque puede ser menos flexible. En entornos empresariales o aplicaciones a gran escala, el modelo cliente-servidor sigue siendo preferido debido a su capacidad para garantizar seguridad, control de acceso y mantenimiento centralizado de datos. Sin embargo, a medida que las necesidades de escalabilidad y distribución aumentan, modelos como la arquitectura basada en microservicios están ganando popularidad, ya que permiten una distribución más granular de tareas entre servidores especializados (Techopedia, 2024).

Arquitecturas Cliente / Servidor

Los clientes transmiten peticiones a los servidores, quienes gestionan estas peticiones y proporcionan respuestas. La estructura cliente-servidor reparte las responsabilidades de una aplicación entre dos entidades: el cliente, que requiere servicios, y el servidor, que suministra dichos servicios. Esta separación permite distribuir las tareas y optimizar el uso de recursos, lo que resulta en un sistema más eficiente y escalable (RedHat, 2024).

Los Componentes de la Arquitectura Son

Cliente. Es la parte de la aplicación que interactúa con el usuario. Puede ser un navegador web, una aplicación de escritorio o una aplicación móvil. Los clientes envían solicitudes al servidor y muestran los resultados recibidos.

Servidor. Es la parte que maneja las solicitudes de los clientes y proporciona los recursos necesarios. Puede ser un servidor de aplicaciones, un servidor de bases de datos o un servidor web. El servidor procesa las solicitudes, accede a los datos requeridos y devuelve las respuestas al cliente.

Red. La interacción entre clientes y servidores se lleva a cabo mediante el uso de una red, que puede ser una red local (LAN) o internet. Esta red es esencial para el intercambio de datos y la conectividad entre los distintos componentes.

Arquitectura Cliente-Servidor de Dos Capas. Este es el modelo más básico, donde el cliente se comunica directamente con el servidor. El cliente envía solicitudes y el servidor responde, lo que puede resultar en una alta carga en el servidor si muchos clientes acceden simultáneamente.

Arquitectura Cliente-Servidor de Tres Capas. En este esquema, se incorpora una capa intermedia denominada o "capa de lógica de aplicación". Esta capa gestiona la lógica de negocio y actúa como intermediario entre el cliente y el servidor, lo que mejora la escalabilidad y el mantenimiento.

Bases de datos

Concepto de Bases de datos

Una base de datos es un sistema organizado que posibilita el almacenamiento, ordenamiento y manejo eficaz de grandes cantidades de información, simplificando el acceso y la gestión de los datos. Incluye tablas que estructuran los datos en filas y columnas, en las que cada fila simboliza un registro singular y cada columna un atributo de dicho registro. Hay diversas clases de bases de datos, tales como las relacionales, que emplean un modelo fundamentado en tablas y SQL para administrar información, y las noSQL, que gestionan datos

no estructurados o semiestructurados. Las bases de datos resultan fundamentales en varias aplicaciones, desde sistemas de administración de empresas hasta redes sociales, dado que ofrecen funciones fundamentales como almacenamiento, acceso seguro, integridad y manejo de datos. Esto las hace una herramienta esencial en el desarrollo de software y en la toma de decisiones basadas en información. Además, aseguran que únicamente usuarios con autorización tengan acceso a la información, salvaguardando de esta manera la privacidad y la seguridad de los datos. (DatabaseGuide, 2024).

Microsoft SQL Server

SQL Server, desarrollado por Microsoft, es un sistema de gestión de bases de datos relacionales (RDBMS) que se destaca por su capacidad para almacenar, gestionar y recuperar información de manera eficiente y segura. A lo largo de los años, SQL Server ha evolucionado significativamente, incorporando características avanzadas que mejoran tanto el rendimiento como la fiabilidad en entornos de bases de datos empresariales. Su amplia gama de herramientas y funciones facilita la ejecución de consultas complejas, la gestión de transacciones y la garantía de integridad de los datos mediante mecanismos robustos de seguridad.

Un aspecto fundamental que distingue a SQL Server es su capacidad de integración con otros productos del ecosistema Microsoft, lo que lo convierte en una solución altamente eficiente para empresas que ya utilizan plataformas como Azure, Power BI o .NET. Además, su capacidad para manejar grandes volúmenes de datos, junto con sus opciones avanzadas de escalabilidad y alta disponibilidad, lo hace ideal para aplicaciones empresariales que requieren rendimiento constante y mínimo tiempo de inactividad. Sin embargo, en comparación con otras soluciones como MySQL o PostgreSQL, SQL Server se caracteriza por ser una opción propietaria, lo que

puede implicar mayores costos de licenciamiento y una curva de aprendizaje más pronunciada debido a su interfaz más compleja y su configuración avanzada.

Desde una perspectiva crítica, la adopción de SQL Server en entornos de grandes datos o en proyectos que exigen alta disponibilidad, como el análisis de datos en tiempo real, es especialmente ventajosa, ya que proporciona un conjunto completo de herramientas de análisis y reporting. En contraposición, en contextos más pequeños o proyectos que demandan soluciones de código abierto, otras alternativas podrían ofrecer un mejor rendimiento a un costo menor (Microsoft Docs, 2024).

SQL

SQL facilita la ejecución de diversas operaciones, tales como la generación de tablas, la introducción de datos, la actualización y supresión de registros, además de la realización de consultas complejas que pueden fusionar datos de varias tablas. Su sintaxis es sencilla e intuitiva, lo que simplifica su asimilación en diversos contextos. Adicionalmente, SQL es compatible con la mayoría de los sistemas de administración de bases de datos, tales como MySQL, PostgreSQL, Oracle y SQL Server, lo que lo establece como un estándar en el sector. (Microsoft Docs, 2024).

Introducción a SQL Server y a las Bases de Datos Relacionales

SQL Server es un sistema de administración de bases de datos relacionales (RDBMS) creado por Microsoft, que facilita el almacenamiento, recuperación y manejo eficaz de datos. SQL Server, diseñado para proporcionar un rendimiento y seguridad superiores, es frecuentemente empleado en ambientes corporativos para la creación de aplicaciones, la administración de datos y la realización de análisis de datos. Su estructura se fundamenta en el

modelo relacional, que estructura la información en tablas interconectadas, lo que simplifica el acceso y manejo de datos a través del uso de SQL (Structured Query Language).

Las bases de datos relacionales son una técnica de organización de datos que utiliza un modelo tabular. Cada tabla contiene filas y columnas, donde las filas representan registros únicos y las columnas representan las particularidades de estos registros. Esta configuración permite establecer conexiones entre diferentes tablas a través de claves primarias y foráneas, lo que mejora la integridad y eficiencia en el almacenamiento de datos. Las bases de datos relacionales utilizan el idioma SQL para realizar funciones como la creación de tablas, la inserción de información, la actualización de registros y las consultas complejas. Esto permite a los usuarios adquirir información de gran valor y tomar decisiones fundamentadas en datos.. (TechNet, 2024).

Cliente Nativo de SQL

El cliente original de SQL Server, a menudo denominado SQL Server Management Studio (SSMS), es un instrumento de gestión y desarrollo que facilita a los usuarios una interacción eficaz con instancias de SQL Server y bases de datos relacionales. Esta herramienta ofrece una interacción visual intuitiva y funciones poderosas para administrar bases de datos, realizar consultas SQL y ejecutar labores de mantenimiento. (Microsoft Docs, 2024).

Características de SQL Server

Teniendo en cuenta la información de (Microsoft Docs, 2024) SQL Server es un sistema de administración de bases de datos relacionales robusto y adaptable, que proporciona una extensa variedad de funciones diseñadas para mejorar la eficiencia, la seguridad y la escalabilidad en el manejo de datos. A continuación, se destacan algunas de sus características más importantes.

- SQL Server asegura la integridad de la información a través de transacciones que satisfacen las características ACID (Atomicidad, Consistencia, Aislamiento, Durabilidad). Esto garantiza que las operaciones se lleven a cabo de forma fiable, incluso en situaciones de errores.
- Utiliza SQL para la creación, consulta y manipulación de datos, permitiendo a los usuarios realizar operaciones complejas de manera sencilla y efectiva.
- Los procedimientos almacenados permiten encapsular la lógica de negocio en la base de datos, lo que mejora el rendimiento al reducir el tráfico de red y facilita la reutilización de código.
- SQL Server ofrece múltiples capas de seguridad, incluyendo autenticación, autorización, cifrado de datos y auditoría, protegiendo la información sensible y asegurando que solo los usuarios autorizados tengan acceso.
- Con características como Always On Availability Groups, replicación y restauración point-in-time, SQL Server asegura que los datos estén siempre disponibles y que se puedan recuperar rápidamente en caso de fallos.
- SQL Server está concebido para gestionar grandes cantidades de información y usuarios simultáneos, facilitando la escalabilidad tanto en el ámbito vertical como horizontal para ajustarse a las demandas de expansión de las organizaciones.
- Ofrece capacidades de análisis de datos mediante la integración con herramientas como los Servicios de Análisis SQL Server (SSAS) y los Servicios de Reportes SQL Server (SSRS), simplifican la elaboración de informes y el estudio de datos.
- SQL Server soporta tipos de datos espaciales y datos no estructurados, lo que permite trabajar con información geográfica y documentos, ampliando su versatilidad en diferentes aplicaciones.

- Incluye herramientas y funciones integradas para implementar modelos de inteligencia artificial y aprendizaje automático directamente dentro de la base de datos, facilitando el análisis avanzado.
- SQL Server Management Studio (SSMS) proporciona una interacción gráfica intuitiva que facilita la administración, desarrollo y consulta de bases de datos, lo que mejora la experiencia del usuario.

SQL Server y el Modelo Cliente/ Servidor

SQL Server se basa en el modelo cliente/servidor, que organiza la interacción entre los usuarios (clientes) y el sistema de gestión de bases de datos (servidor). En este modelo, el cliente, que puede ser una aplicación o una interacción gráfica como SQL Server Management Studio, envía consultas SQL al servidor, que gestiona y almacena los datos. La comunicación se realiza a través de protocolos como TCP/IP, utilizando el Protocolo de TDS para transmitir datos de manera eficiente. Este enfoque centraliza el almacenamiento de información, lo que simplifica la administración, mejora la seguridad y facilita el mantenimiento. Sin embargo, también presenta desafíos, como la dependencia de la red y la carga en el servidor con múltiples clientes concurrentes (Microsoft Docs, 2024).

Ventajas de los Clientes

Interacción Amigable. Proporcionan interfaces gráficas que facilitan la interacción del usuario.

Descentralización. Permiten parte del procesamiento en el cliente, aliviando la carga del servidor.

Flexibilidad. Se pueden actualizar o reemplazar fácilmente sin afectar la infraestructura del servidor.

Acceso Local. Pueden ejecutar aplicaciones localmente, útil en casos de conectividad limitada.

Trabajo Offline. Permiten realizar tareas sin conexión y sincronizar datos posteriormente.

Costos Reducidos. Generalmente, son más económicos de implementar que sistemas **completamente centralizados.**

Personalización. Los usuarios pueden ajustar su entorno de trabajo según sus preferencias.

Ventajas del Servidor

Interacción Amigable. Facilitan la interacción del usuario con aplicaciones mediante interfaces gráficas intuitivas.

Descentralización del Procesamiento. Permiten realizar parte del procesamiento en el cliente, aliviando la carga del servidor.

Flexibilidad. Pueden actualizarse o reemplazarse fácilmente sin afectar la infraestructura del servidor.

Acceso Local a Aplicaciones. Pueden ejecutar aplicaciones localmente, útil en situaciones de conectividad limitada.

Trabajo Offline. Permiten realizar tareas sin conexión y sincronizar datos posteriormente.

Descripción y Análisis de la Situación Problema

Breve descripción del Taller Chaman Bikes

Chaman Bikes es un taller de bicicletas situado en Guarne, especializado en la gestión preventiva y correctiva, además de la venta de artículos y accesorios de ciclismo. Este taller se dedica a garantizar que cada bicicleta que recibe atención esté en condiciones óptimas, proporcionando a los ciclistas una experiencia segura y placentera.

Además de su compromiso con la calidad y la atención al detalle, Chaman Bikes promueve una cultura de movilidad sostenible. La empresa valora la bicicleta como un medio sano y bueno para el medio ambiente, y se esfuerza por educar a la comunidad sobre su mantenimiento y uso responsable.

La tienda de Chaman Bikes ofrece una amplia gama de productos, desde componentes de alta calidad hasta ropa y accesorios, diseñados para mejorar la experiencia del ciclista.

Misión

En Chaman Bikes, nuestra misión es ser el taller de referencia en Guarne para todos los entusiastas del ciclismo. Nos especializamos en ofrecer servicios de mantenimiento preventivo y correctivo, asegurando que cada bicicleta que pasa por nuestras manos funcione de manera óptima y segura. Creemos que el ciclismo no solo es un medio de transporte, sino una forma de vida que promueve la salud, la sostenibilidad y el bienestar comunitario. Nos dedicamos a proporcionar un servicio de alta calidad y personalizado, satisfaciendo las demandas particulares de cada cliente y su bicicleta. Además, ofrecemos una amplia variedad y calidad de artículos y accesorios de ciclismo que complementan la experiencia de nuestros ciclistas, desde componentes hasta indumentaria, todo seleccionado cuidadosamente para garantizar la mejor experiencia.

Visión

En Chaman Bikes, aspiramos a ser el taller líder en Guarne, reconocido por brindar servicios de mantenimiento preventivo y correctivo de alta calidad para bicicletas. Nos proyectamos como un espacio donde cada ciclista encuentre atención personalizada, seguridad y confianza, promoviendo al mismo tiempo la movilidad sostenible y el bienestar comunitario. Buscamos destacarnos por ofrecer artículos y accesorios de ciclismo seleccionados con responsabilidad, fomentando un estilo de vida activo y respetuoso con el entorno. Visualizamos un entorno acogedor donde se cultive el amor por el ciclismo y se inspire a nuevas generaciones a vivir con pasión y compromiso por la aventura.

Valores de Chaman Bikes

- **Compañerismo.** Fomentar relaciones solidarias y de apoyo entre todos.
- **Honestidad.** Mantener transparencia en todas nuestras interacciones.
- **Amabilidad.** Tratar a todos con respeto y consideración.
- **Motivación.** Fomentar un ambiente positivo y estimulante.
- **Adaptabilidad.** Ser flexibles ante los cambios y desafíos.
- **Nobleza.** Actuar con dignidad y principios éticos en todas las situaciones.
- **Bondad.** Actuar con generosidad y consideración hacia los demás.
- **Integridad.** Actuar con ética y coherencia.
- **Creatividad.** Fomentar la innovación en nuestros servicios.
- **Empatía.** Comprender y valorar las experiencias de los demás.
- **Solidaridad.** Apoyar a la comunidad y a los ciclistas.
- **Benevolencia.** Actuar con generosidad y consideración hacia los demás.
- **Inspiración.** Motivar a otros a disfrutar del ciclismo y la aventura.
- **Compromiso.** Dedicación a la satisfacción y bienestar de nuestros clientes.
- **Inclusión.** Promover un ambiente donde todos se sientan bienvenidos.
- **Cuidado.** Prestar atención a los detalles en nuestros servicios y productos.
- **Lealtad.** Compromiso con nuestros clientes y la comunidad.
- **Empatía.** Comprender y valorar las experiencias de los demás.
- **Trabajo en Equipo.** Colaborar para alcanzar objetivos comunes.
- **Adaptabilidad.** Ser flexibles ante los cambios y desafíos.
- **Solidaridad.** Apoyar a la comunidad ciclista y fomentar un espíritu de unión.

Análisis de Resultados de Encuestas Para Directivos y Empleados

¿Con qué Regularidad Utiliza el Computador en el Taller?

Tabla 1

Encuesta 1

| Opciones | Respuestas | Porcentaje |
|----------|------------|------------|
| Siempre | 1 | 20% |
| A veces | 2 | 40% |
| Rara vez | 2 | 40% |
| Nunca | 0 | 0% |
| Total | 5 | 100% |

Interpretación

El resultado indica que la computadora se utiliza constantemente en las tareas diarias de los empleados del taller. Por lo tanto, se convierte en un método esencial para el buen funcionamiento de la empresa.

¿Cuál es la Mayor Dificultad que se Manifiesta en la Actualidad en el Taller?

Tabla 2

Encuesta 2

| Opciones | Respuestas | Porcentaje |
|------------------------------|-------------------|-------------------|
| Información mal administrada | 4 | 80% |
| Falta de Personal | 1 | 20% |
| Falta de Equipamiento | 0 | 0% |
| Total | 5 | 100% |

Interpretación

Los administradores y empleados del taller creen que el mayor problema que enfrenta actualmente el taller es la información mal administrada, ya que deben manejar esta información manualmente, con frecuencia corriendo el riesgo de perderla.

¿Cómo Usted Registra los Movimientos y Operaciones del Taller?

Tabla 3

Encuesta 3

| Opciones | Respuestas | Porcentaje |
|-----------------|-------------------|-------------------|
| Computador | 1 | 20% |
| Papel | 4 | 80% |
| Otros | 0 | 0% |
| Total | 5 | 100% |

Interpretación

Según las respuestas, el 20% de los empleados registran sus datos a través de computadoras, mientras que el 80% por ciento lo hace en papel, lo que no garantiza la seguridad y la confidencialidad de los datos, y su información está expuesta a todos. Por lo tanto, se debe regularizar el uso de computadoras en todas las áreas de la empresa y con la participación de todos los empleados.

¿Ha Experimentado la Pérdida de Datos Relevantes para el Taller?

Tabla 4

Encuesta 4

| Opciones | Respuestas | Porcentaje |
|-----------------|-------------------|-------------------|
| Si | 4 | 80% |
| No | 1 | 40% |
| Total | 5 | 100% |

Interpretación

Un alto porcentaje de los empleados ha perdido datos importantes del taller, lo que demuestra que sabemos cómo guardar, manipular y proteger la información. Esto representa un grave problema para el desempeño de las actividades del taller, ya que nos permite manipular los datos cuando sea necesario y con quién sea. no reflejando los datos reales del negocio de esta manera.

¿Qué Piensa que Sería la Opción más Adecuada para Potenciar la Atención a los Usuarios?

Tabla 5

Encuesta 5

| Opciones | Respuesta | Porcentaje |
|--------------------------|------------------|-------------------|
| Automatizar los procesos | 4 | 80% |
| Contratar más personal | 1 | 20% |
| Comprar más maquinas | 1 | 20% |
| Ninguna | 0 | 0% |
| Total | 5 | 100% |

Interpretación

El mayor porcentaje de resultados indica que la automatización de procesos mejoraría significativamente el desarrollo de las actividades y la atención al cliente, además de permitir el mantenimiento de información segura y confiable para que el usuario pueda acceder a datos verídicos que muestren la situación real del taller cuando lo requiera.

*¿Piensa Usted que la Puesta en Marcha de un Sistema de Computación Potencia el
Desempeño del taller?*

Tabla 6

Encuesta 6

| Opciones | Respuesta | Porcentaje |
|-----------------|------------------|-------------------|
| Si | 5 | 100% |
| No | 0 | 0% |
| Total | 5 | 100% |

Interpretación

La mayoría de los entrevistados concuerdan en que la implementación de un sistema informático potencia el desempeño de una compañía y contribuye al procesamiento de la información, por lo que consideran imprescindible y urgente disponer de un software que asista en la gestión y supervisión de la información del taller.

En qué Equipos Informáticos Considera Usted que Debería Funcionar un Software en el Taller?

Tabla 7

Encuesta 7

| Opciones | Respuesta | Porcentaje |
|----------------------------------|------------------|-------------------|
| Los que posee actualmente | 4 | 80% |
| Los que poseen más actualización | 1 | 20% |
| Nuevos de última tecnología | 0 | 0% |
| Ninguno | 0 | 0% |
| Total | 5 | 100% |

Interpretación

Según el 80% de los participantes en la encuesta, si es viable implementar un software para el taller, se debe hacerlo en los equipos que ya tienen para que la inversión no sea significativa pero el rendimiento de los trabajadores mejore y, por lo tanto, la atención al cliente sea adecuada.

*¿Mantiene un Control Adecuado de los Ingresos y Egresos de Bienes del Taller?***Tabla 8***Encuesta 8*

| Opciones | Respuesta | Porcentaje |
|-----------------|------------------|-------------------|
| Si | 5 | 100% |
| No | 0 | 0% |
| Total | 5 | 100% |

Interpretación

Todas las personas que participaron en la encuesta están de acuerdo en que el control de ingresos y salidas de mercancías de la empresa es insuficiente.

Evaluación de los Hallazgos del Sondeo Realizado a los Clientes

¿Con qué Regularidad Acude al Taller?

Tabla 9

Encuesta 9

| Opciones | Respuesta | Porcentaje |
|----------|-----------|------------|
| Siempre | 72 | 72% |
| A veces | 20 | 20% |
| Rara vez | 8 | 8% |
| Nunca | 0 | 0% |
| Total | 100 | 100% |

Interpretación

La mayoría de las personas que visitan el taller siempre regresan, lo que demuestra que el servicio mecánico es de excelente calidad y que sus clientes confían en el taller.

¿Considera Usted que la Atención en el Taller es?

Tabla 10

Encuesta 10

| Opciones | Respuesta | Porcentaje |
|--------------------|------------------|-------------------|
| Buena | 90 | 90% |
| Mala | 2 | 2% |
| Buena, pero demora | 8 | 8% |
| Ninguno | 0 | 0% |
| Total | 100 | 100% |

Interpretación

Los clientes que visitan el taller dicen que la atención es buena pero demorada. Después de hacer una observación, se descubrió que la demora se produce en los procesos de cobro y salida de mercancías.

¿Considera Usted que la Mercancía del Taller es?

Tabla 11

Encuesta 11

| Opciones | Respuesta | Porcentaje |
|-----------------|------------------|-------------------|
| Excelente | 82 | 82% |
| Buena | 15 | 15% |
| Regular | 3 | 3% |
| Deficiente | 0 | 0% |
| Total | 100 | 100% |

Interpretación

Las respuestas muestran que la mayoría de los clientes encuestados están satisfechos con el estado de los bienes que ofrece la empresa. Es importante destacar que los clientes encuestados en su mayoría han utilizado los servicios del taller durante al menos seis meses.

¿Cuánto Tiempo le Lleva Hacer un Pago por un Producto o Servicio en el Taller?

Tabla 12

Encuesta 12

| Opciones | Respuesta | Porcentaje |
|-----------------|------------------|-------------------|
| 3 a 5 | 76 | 76% |
| 5 a 8 | 18 | 18% |
| > 8 | 6 | 6% |
| Total | 100 | 100% |

Interpretación

Se espera que con la implementación del sistema se reduzca significativamente el tiempo de espera que muchos clientes estiman para pagar un servicio o comprar un artículo entre 5 y 8 minutos en promedio.

¿Cuál Piensa que Sería la Opción más Adecuada para Optimizar el Tiempo en el Servicio a los Clientes?

Tabla 13

Encuesta 13

| Opciones | Respuesta | Porcentaje |
|--------------------------|------------------|-------------------|
| Automatizar los procesos | 82 | 82% |
| Contratar más personal | 18 | 18% |
| Ninguna | 0 | 0% |
| TOTAL | 100 | 100% |

Interpretación

El mayor porcentaje que indica el resultado indica que la automatización de procesos mejoraría significativamente el desarrollo de actividades y la atención al cliente.

Conclusión de las interpretaciones

Los hallazgos de la encuesta señalan que el ordenador es un instrumento crucial para las tareas cotidianas del taller, sin embargo, la administración manual de la información enfrenta diversas dificultades, tales como la pérdida de información relevante y la ausencia de protección en los datos. La mayoría de los trabajadores y gestores están de acuerdo en la puesta en marcha de un sistema de computación mejoraría significativamente la eficiencia de las operaciones internas y la atención al cliente, reduciendo tiempos de espera en áreas como el cobro y la salida de mercancías. Además, la encuesta revela que, aunque el servicio al cliente es de calidad y los clientes confían en el taller, existen áreas de mejora en la automatización de procesos.

La mayoría de los encuestados considera viable implementar un software en los equipos existentes para mejorar la administración de la información y optimizar el rendimiento del taller sin realizar una gran inversión.

Software Para Gestión Administración Taller de Bicicletas Chamán Bikes

Justificación

El problema identificado es la falta de sistematización de los procesos operativos y administrativos del taller Chaman Bikes. La utilización de minutas, talonarios, facturas, acuerdos, cotizaciones e inventarios en papel ha generado diversos problemas, como pérdidas de materia prima debido al manejo deficiente del stock y falta de precisión en las facturas, lo que ha ocasionado pérdidas económicas para la empresa. Estos inconvenientes han llevado a la pérdida de clientes y proveedores, así como a retrasos en el despacho del producto terminado, impactando negativamente en la operación de la empresa. Realizar un sistema de organización automatizado que permita llevar a cabo de manera eficiente los procedimientos administrativos y operativos del taller. Mejorar la precisión en el manejo del inventario y minimizar las pérdidas de materia prima. Optimizar la emisión de facturas para evitar pérdidas económicas. Generar una interacción amigable y fácil de usar para los empleados del taller, permitiéndoles realizar consultas, compras de productos, agendar mantenimientos y reparaciones, y visualizar el historial de transacciones realizadas.

Se debe analizar detalladamente la viabilidad de la implementación de un sistema de gestión automatizado. Se deben considerar aspectos técnicos, económicos y sociales. En el caso de Chaman Bikes, se ha identificado que no cuentan actualmente con un presupuesto suficiente para adquirir un software, lo que podría representar un obstáculo financiero. Sin embargo, se puede investigar alternativas más económicas, como el uso de software de código abierto o la contratación de servicios de desarrollo a medida. Es relevante implementar un sistema de gestión automatizado en Chaman Bikes debido a los problemas y dificultades generados por el manejo manual de los procesos. Investigaciones y análisis previos en otras empresas del sector han

demostrado los beneficios de la automatización en términos de eficiencia, precisión y satisfacción del cliente. Asimismo, se pueden presentar estadísticas de los beneficios obtenidos en otras empresas que han implementado sistemas similares. La implementación de un sistema de gestión automatizado en Chaman Bikes permitirá mejorar la eficiencia y precisión en los procesos operativos y administrativos. Esto reducirá las pérdidas de materia prima y las pérdidas económicas asociadas, así como los retrasos en el despacho del producto terminado. Además, la interacción amigable y fácil de usar mejorará la satisfacción del cliente, permitiéndoles realizar consultas, compras y agendamientos con mayor facilidad. Como beneficios indirectos, se espera una mayor retención de clientes y proveedores, así como optimización en la reputación e imagen del taller.

Se propone implementar un sistema de gestión automatizado que incluya módulos para la gestión del inventario, la emisión de facturas, la gestión de contratos y cotizaciones, y la agenda de mantenimientos y reparaciones. Se establecerán plazos y actividades para la implementación del sistema, así como los recursos necesarios, incluyendo la contratación de servicios de desarrollo a medida si es necesario. Se estimará un presupuesto y se buscarán opciones de financiamiento, como líneas de crédito o programas de apoyo a las Mipymes.

La integración de un sistema de integración automatizado en Chaman Bikes es necesaria para resolver los problemas y dificultades generados por el manejo manual de los procesos. Los beneficios esperados incluyen una mayor eficiencia, precisión y satisfacción del cliente, así como una mejora en la retención de clientes y proveedores. A pesar de las limitaciones financieras, se buscarán alternativas viables y se desarrollará un plan de medidas para ejecutar el proyecto.

Objetivo General

Desarrollar un aplicativo web de inventario y mantenimientos para el taller de bicicletas Chaman Bikes mediante el uso de PHP y MySQL.

Objetivos específicos

Identificar el estado actual del registro de entradas y salidas del taller de bicicletas Chaman Bikes en el municipio de Guarne.

Diseñar una aplicación web, mediante el uso de tecnologías como PHP, MySQL, HTML, CSS, JavaScript y Bootstrap.

Implementar un sistema de mantenimientos e inventarios que permita realizar un seguimiento de todas las bicicletas y sus respectivas características (marca, datos del cliente, color, etc.).

Realizar pruebas de funcionamiento del software con apache Jmeter para analizar la carga del rendimiento general.

Impacto

Crear un software de bicicletas en el municipio de Guarne podría tener un impacto significativo en diversas áreas. En primer lugar, fomentaría el transporte sostenible, reduciendo la dependencia de vehículos motorizados y disminuyendo la contaminación del aire. Esto, a su vez, contribuiría a mejorar la salud pública al incentivar a los ciudadanos a adoptar un estilo de vida más activo. El software también podría identificar áreas que requieren una mejor infraestructura ciclista, como carriles exclusivos y estacionamientos seguros, beneficiando a los ciclistas y promoviendo una movilidad más eficiente. Además, aprovechando el desarrollo de las nuevas rutas ciclistas y la apertura de la nueva carretera en el sector del estado, se podría decir que va a impactar de manera positiva la economía local y promoviendo la cultura, vías urbanas y

naturaleza del sector rural del municipio de Guarne. Al mejorar la conectividad y el acceso a servicios, se facilitaría la movilidad de los residentes, especialmente en zonas con transporte público limitado. La implementación de este software también fomentaría la interacción social al crear espacios para eventos y actividades grupales. Asimismo, la recopilación de datos sobre el uso de las bicicletas permitiría a las autoridades locales tomar decisiones informadas sobre planificación urbana y movilidad. Por último, incluir módulos de educación sobre seguridad vial ayudaría a crear un entorno más seguro para todos, mientras que la creación del software podría inspirar otros proyectos tecnológicos en la región, generando nuevas oportunidades de empleo y desarrollo.

Planteamiento del problema

La compañía Chaman Bikes, que lleva más de tres años en el municipio de Guarne, se enfoca en el mantenimiento, distribución, adquisición y comercialización de productos relacionados con bicicletas. No obstante, los procesos administrativos y operacionales de la empresa se llevan a cabo de manera manual, utilizando minutas, hojas de registro, facturas, contratos, cotizaciones e inventarios en papel.

Esta metodología de trabajo ha provocado varias dificultades en los años recientes. Además, no se cuenta con una interacción amigable y fácil de usar para que los empleados del taller puedan realizar consultas, realizar compras de productos, agendar mantenimientos y reparaciones, así como visualizar el historial de transacciones realizadas. Esto genera una menor satisfacción del cliente, ya que no se ofrece un servicio ágil y eficiente.

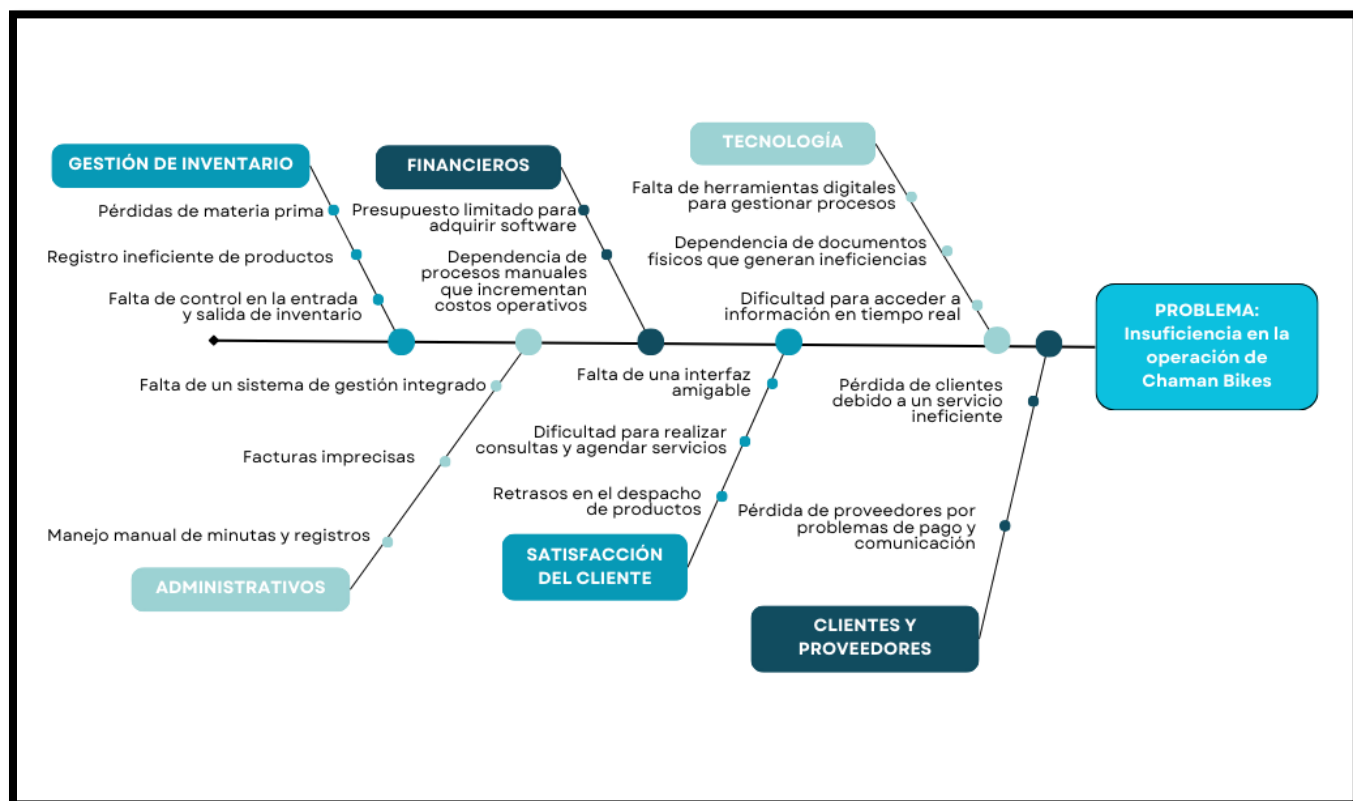
Primero, se han observado pérdidas de materia prima a causa de una gestión inadecuada del inventario. Además, la inexactitud en las facturas ha provocado pérdidas financieras para la compañía. Estos inconvenientes han provocado la pérdida de clientes y proveedores, además de

demoras en la entrega del producto final. Estos problemas han afectado de manera adversa el funcionamiento del taller.

Al ejecutar un análisis del estado actual del taller, se ha planteado la interrogante de ¿cuál es la deficiencia en el funcionamiento? ¿Cuál es el motivo de la ausencia de sistematización en sus procesos, ya sean operativos o administrativos? Es importante destacar que, al ser una PYME, Chaman Bikes no posee el presupuesto adecuado para comprar un software que facilite la gestión eficaz de sus procesos. Dado lo anterior, el taller ha tenido el impulso de pedir una aplicación que les permita aliviar los inconvenientes y retos previamente mencionados.

Figura 1

Diagrama espina de pez planteamiento del problema



Diseño de la Metodología

Tipo de Investigación

El diseño metodológico para el desarrollo de Chaman Bikes, un software de mantenimiento y venta de artículos de bicicletas se fundamenta en una investigación descriptiva y aplicada. El objetivo general es crear una solución que optimice la gestión de mantenimiento y la venta de productos relacionados, mejorando la experiencia del usuario.

El procedimiento inicia con un estudio de requerimientos a través de encuentros con los involucrados y métodos de recopilación de datos, tales como cuestionarios y entrevistas, con el objetivo de detectar las demandas específicas de los usuarios. Luego, se realizará el diseño de la arquitectura del software, la interfaz de usuario y la base de datos. En la fase de codificación, se activarán módulos y se realizarán ensayos unitarios para garantizar la calidad del código. Luego, se realizarán evaluaciones funcionales, de usabilidad y de carga para confirmar el funcionamiento adecuado del sistema.

Metodología

La metodología para el desarrollo de Chaman Bikes se fundamenta en un enfoque ágil, empleando el marco de trabajo Scrum para garantizar flexibilidad y adaptación a lo largo del proceso. Iniciaremos con un análisis de requisitos, donde realizaremos reuniones con interesados, como mecánicos y usuarios, así como encuestas y entrevistas para recopilar información sobre sus necesidades específicas. Posteriormente, se procederá al diseño del software, creando prototipos de la interacción y definiendo la arquitectura del sistema, asegurando que estos diseños se revisen y validen con los usuarios para alinear expectativas.

La realización del trabajo se hará en sprints de dos a cuatro semanas, permitiendo una implementación gradual de los módulos del software. Al final de cada sprint, se realizarán

reuniones de revisión y demostración para presentar los avances y recoger feedback, lo que facilitará realizar ajustes y mejoras rápidamente. Además, se realizarán pruebas continuas, incluyendo pruebas unitarias, funcionales y de usabilidad, para asegurar que cada componente cumpla con los estándares de calidad.

Vida útil del software

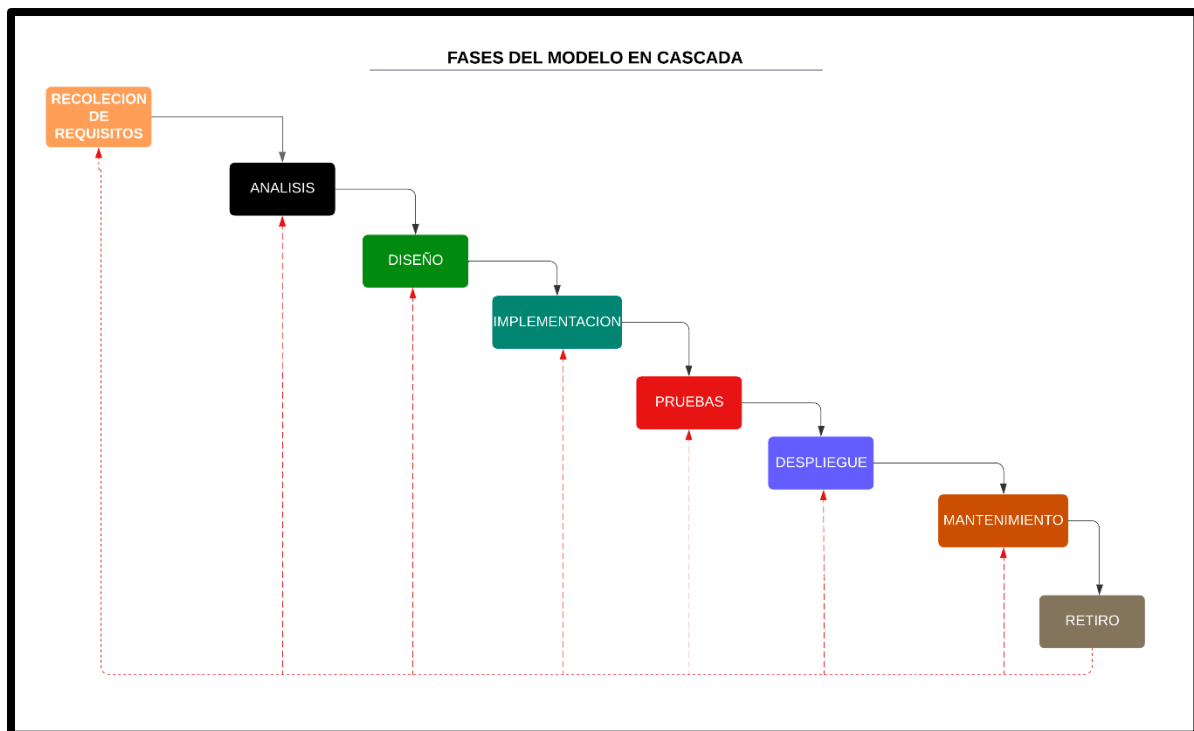
La vida útil del software para un sistema de mantenimiento de bicicletas utilizando un modelo en cascada que es secuencial, lo que significa que cada etapa debe finalizarse antes de avanzar a la próxima. Esto puede ser beneficioso para proyectos con requisitos bien definidos, pero también puede ser menos flexible ante cambios en las necesidades del cliente. El modelo se podría aplicar de la siguiente manera.

- **Recolección de Requisitos.** Se identifican y documentan las necesidades del usuario, como gestión de reservas y seguimiento de reparaciones.
- **Análisis.** Se analizan los requisitos para crear un documento que defina el funcionamiento del sistema, incluyendo diagramas de flujo.
- **Diseño.** Se elabora la estructura del programa, la plataforma de usuario y la base de datos, creando prototipos si es necesario.
- **Implementación.** Los desarrolladores programan el software siguiendo las especificaciones del diseño, incluyendo todas las funcionalidades necesarias.
- **Pruebas.** Se llevan a cabo ensayos rigurosos para identificar y rectificar fallos, asegurando que todas las funcionalidades funcionen correctamente.
- **Despliegue.** Se instala el software se emplea en el entorno de producción y se capacita a los usuarios sobre su uso.

- **Mantenimiento.** Tras la implementación, el software se adentra en la fase de mantenimiento, donde se realizan las actividades correspondientes. actualizaciones y mejoras acorde a las demandas de los usuarios y se solucionan los fallos identificados.
- **Retiro.** Finalmente, cuando el software ya no es necesario o ha sido reemplazado por una versión más avanzada, se procede a su retiro. Esto puede incluir la desinstalación y la exportación de información relevante a un nuevo sistema si es necesario.

Figura 2

Diagrama espina de pez planteamiento del problema



Aplicación de Cada Etapa del Modelo en Cascada

El modelo en cascada es uno de los enfoques más tradicionales para el desarrollo de software, caracterizado por su estructura lineal y secuencial, en la que cada fase del proyecto (análisis, diseño, implementación, pruebas y mantenimiento) debe completarse antes de pasar a la siguiente. Aunque este enfoque ha sido ampliamente utilizado en proyectos de gran escala y con requisitos bien definidos, no está exento de críticas, especialmente en cuanto a su inflexibilidad para adaptarse a cambios a medida que avanza el desarrollo.

Sin embargo, se ha optado por el modelo en cascada para el desarrollo del software en este proyecto debido a su estructura clara y su facilidad de gestión en proyectos con requisitos estables y bien definidos desde el inicio. Este enfoque es particularmente adecuado cuando se busca garantizar que todas las fases del desarrollo se completen de manera exhaustiva antes de proceder a la siguiente, lo cual es esencial para proyectos donde los errores en fases iniciales pueden tener un impacto significativo en las etapas posteriores.

Análisis de Requisitos

El análisis es fundamental para poder empezar a desarrollar el software que cumpla con las necesidades del taller de bicicletas. El análisis de requerimientos se podría aplicar de la siguiente manera.

Identificación del Personal del Sistema

- **Propietario.** Desea un sistema que administre los datos de los clientes, el inventario y las órdenes laborales, además de los diversos tipos de servicios que brinda el taller como el mantenimiento preventivo y correctivo.
- **Mecánicos.** Necesitarán acceso fácil a los registros de reparación y las especificaciones de las bicicletas.

Diseño de Entrada y Salida

Tabla 14

Diseño de Entrada y Salida

| | |
|---|---|
| Cliente | Realiza el pedido, se envía y se entrega la factura. |
| Proveedor | Una persona que entrega productos al negocio. |
| Factura | Es un papel que se entrega al cliente como prueba de que ha realizado una compra. |
| Pantalla de impresión de informes | Los diferentes informes de inventario se implementan. |
| Monitor de última actualización de productos nuevos | El proveedor envía nuevos productos y artículos al sistema. |

Recolección de Requisitos

- **Entrevistas.** Habla con los propietarios y empleados para entender sus necesidades.
- **Cuestionarios.** Diseña encuestas para recopilar información de manera más estructurada.
- **Observación.** Observa el flujo de trabajo en el taller para identificar áreas de mejora.

Funciones

Implementa un proceso sencillo que permita a los usuarios registrarse con facilidad, proporcionando solo la información básica necesaria. Esto incluirá opciones de verificación para asegurar la autenticidad de las cuentas y la protección de datos personales.

- Proporciona una visualización clara y renovada en tiempo real acerca de la disponibilidad de bicicletas y otros equipos en el inventario. Esto permitirá a los usuarios

conocer al instante qué opciones están disponibles para alquilar. Además, se incluirán filtros para facilitar la búsqueda según preferencias específicas.

- Establece un sistema para documentar todos los servicios de mantenimiento realizados en cada bicicleta. Esto incluirá un seguimiento de las intervenciones, el estado de cada bicicleta. De esta manera, se garantizará que todas las bicicletas se mantengan en condiciones óptimas.

- Permite actualizar y cargar las fotos de las bicicletas que se encuentran o estuvieron en mantenimiento.

- Permite la generación de informes periódicos en Excel, que resumen el estado del inventario, incluyendo bicicletas disponibles, en mantenimiento y requeridas. Esto ayudará en la toma de decisiones sobre compras y mantenimiento.

Limitaciones

- El acceso al sistema estará restringido exclusivamente al súper administrador, quien tendrá la autoridad para gestionar todas las funciones del software. Esto incluye la capacidad de añadir, editar o eliminar usuarios y supervisar todas las operaciones del sistema.

- Como el taller carece de personal formado en la gestión de bases de datos, el sistema debe ser diseñado para ser intuitivo y sencillo de utilizar. Es necesario establecer una interacción fácil de usar que habilite a los usuarios sin habilidades técnicas a ejecutar tareas elementales, como documentar mantenimientos y comunicar problemas.

- El negocio dispone de escaso personal para la gestión y cuidado del sistema por lo que es fundamental que el sistema sea eficiente y automatizado en la medida de lo posible. Esto implica diseñar flujos de trabajo que minimicen la carga de trabajo del personal

- La ausencia de un soporte técnico robusto puede dificultar la resolución rápida de problemas. Esto puede llevar a tiempos de inactividad prolongados y a una experiencia insatisfactoria para los usuarios.
- Plazos ajustados pueden llevar a una reducción en la calidad del software o a la omisión de requisitos importantes

Módulos

Módulo de Inventario

- El módulo de inventario se diseñará con una interacción de usuario que incluya un panel de control para visualizar el estado general del inventario. Este panel mostrará información esencial, como el total de bicicletas, las que están en mantenimiento, las disponibles y las fuera de servicio. El menú de navegación permitirá un acceso rápido a secciones clave.
- Para gestionar el historial de mantenimiento, se creará una sección donde se ingresarán los detalles de cada mantenimiento realizado. Esto incluirá el ID de la bicicleta, la fecha del mantenimiento, el tipo de servicio, los detalles de las reparaciones y las piezas reemplazadas.
- Para la actualización del estado del inventario de accesorios, se facilitará una lista completa de todos los artículos registrados, permitiendo a los administradores filtrar y buscar rápidamente según el tipo de accesorio o el estado del inventario. Al seleccionar un accesorio, se podrá actualizar su cantidad en stock, ajustar el precio y añadir comentarios sobre su estado actual.
- Administrar productos con un proveedor específico asociado y una línea de productos específica. Para el proceso de facturación y los controles internos, como los

inventarios físicos, es esencial tener actualizados los productos, la existencia, la descripción y los precios.

Módulo de Facturación

- Emitir facturas de productos si el producto no está exento de impuestos, debe aplicarse el rebajo de productos.
- El sistema debe verificar que el código de cada vendedor y cliente sea único cuando se registran o cuando alguien se convierte en un nuevo cliente.
- Este sistema permite a las personas tomar decisiones inteligentes con información actualizada y precisa.

Consultas

- **Todos los accesorios disponibles.** Recupera todos los accesorios que están disponibles.
- **Accesorios por tipo.** Permite filtrar los accesorios según un tipo específico, como cascos, luces, candados, etc.
- **Accesorios con bajo stock.** Devuelve una lista de accesorios que están por debajo de un umbral específico, identificando qué artículos necesitan ser reabastecidos aplicando un formato específico.
- **Cambios de un accesorio.** Proporciona un registro de todas las modificaciones realizadas en un accesorio específico, permitiendo seguir su historial.
- **Compras recientes.** Obtiene todas las solicitudes ejecutadas en el último mes.
- **Consulta de accesorios con mayor venta.** Devuelve una lista de los accesorios más vendidos, ayudando a identificar tendencias en la demanda.

- **Consulta de ingresos por ventas de accesorios.** Calcula los ingresos totales generados por las ventas de accesorios en el último mes.
- **Consulta de accesorios en pedido.** Visualiza todas las órdenes de compra que están pendientes de recibir, facilitando el seguimiento de las entregas.
- **Lista de facturas generales.** Una lista de todas las facturas que ha emitido el taller.

Requisitos de Rendimiento

El sistema se construirá utilizando herramientas de última generación como PHP y SQL Server, soportará grandes flujos de información y proporcionará un entorno cliente-servidor trabajando en una intranet, lo que garantizará un rendimiento óptimo.

Hardware

Para que la aplicación se pueda desarrollar de manera correcta se necesitan los siguientes elementos pertenecientes al hardware.

Un servidor que tenga las siguientes especificaciones.

Tabla 15

Requisitos básicos del Hardware

| Elemento | Descripción |
|-----------------|------------------------|
| Procesador | Intel (R) Core (TM) I5 |
| Memoria RAM | 8 GB |
| Disco Duro | SSD 256 GB |
| Tarjeta de red | 1000 Mbps |

Los computadores clientes deben tener las mismas características

Software

Servidor

Tabla 16

Requisitos Básicos del Software

| Programa | Versión |
|------------------------------|---------------------|
| Sistema Operativo de Windows | 10 Pro |
| Xamp | 8.0.30 / PHP 8.0.30 |
| Google Chrome | 129.0.6668.70/.71 |
| Sql server | 202217 del 2022 |

Tabla 17

Requisitos básicos del Software en la versión cliente

| Programa | Versión |
|------------------------------|-------------------|
| Sistema Operativo de Windows | 10 pro |
| Google Chrome | 129.0.6668.70/.71 |

Seguridad

- Se implementa un sistema de regulación de acceso fundamentado en roles es fundamental. Esto significa que solo el súper administrador y otros usuarios autorizados tienen acceso completo al sistema. Otros empleados tendrán permisos limitados según su trabajo, lo que garantiza que la información confidencial no esté disponible para todos.

- Respaldo regular de la base de datos y de la información crítica. Los respaldos deben almacenarse en un lugar seguro y ser accesibles solo para usuarios autorizados. Esto garantiza la recuperación de datos en caso de pérdida o corrupción.
- Formar a los trabajadores en medidas de seguridad, tales como la gestión segura de contraseñas y la detección de emails dudosos. Es fundamental la sensibilización del personal para evitar ataques de ingeniería social.
- Llevar a cabo ensayos de seguridad regulares, tales como auditorías y pruebas de penetración, para detectar debilidades y valorar la capacidad del sistema para resistir ataques. Esto contribuye a reducir los riesgos antes de que sean aprovechados.

Diseño

La fase de diseño de la aplicación del taller implica entender las necesidades de los clientes, crear un prototipo de la interacción y definir las funcionalidades clave, como la venta de productos y la gestión de citas para mantenimiento. Es importante planificar la estructura del sistema, asegurando que la base de datos maneje bien la información de clientes y productos. También se debe pensar en proteger los datos y en la integración de métodos de pago. Además, se deben documentar las decisiones tomadas y planificar una estrategia de lanzamiento para atraer usuarios.

Backlog

El backlog de "Chaman Bikes" es una lista priorizada de tareas y requisitos que deben implementarse en el software de gestión del taller. Incluye tanto las funcionalidades que se desean desarrollar como las mejoras y correcciones que deben realizarse.

Requerimientos Funcionales

Tabla 18

Requerimientos funcionales

| ID | Requerimiento Funcional | Descripción |
|-----------|---------------------------------------|---|
| RF-001 | Gestión de Clientes | El sistema debe habilitar el registro, modificación y supresión de datos de los clientes. |
| RF-002 | Registro de Bicicletas | Permitir registrar las bicicletas de los clientes, incluyendo serial, marca y color. |
| RF-003 | Programación de Mantenimientos | Permitir a los clientes programar citas para mantenimiento de bicicletas, especificando fecha y hora. |
| RF-004 | Gestión de Servicios de Mantenimiento | Registrar y listar los diferentes servicios de mantenimiento ofrecidos, con descripciones y precios. |
| RF-005 | Ventas de Artículos | Permitir la venta de artículos relacionados con bicicletas, con opción de agregar al carrito. |
| RF-006 | Procesar Pagos | Implementar un sistema para procesar pagos mediante tarjeta de crédito, débito y efectivo. |
| RF-007 | Generación de Reportes | Generar reportes de ventas, servicios realizados y clientes atendidos en un período específico. |
| RF-008 | Gestión de Inventario | Mantener un inventario actualizado de los artículos en venta, con alertas para bajo stock. |
| RF-010 | Acceso de Administrador | Proveer acceso administrativo para gestionar el sistema, incluyendo usuarios y permisos. |

Requisitos No Funcionales

Tabla 19

Requerimientos No Funcionales

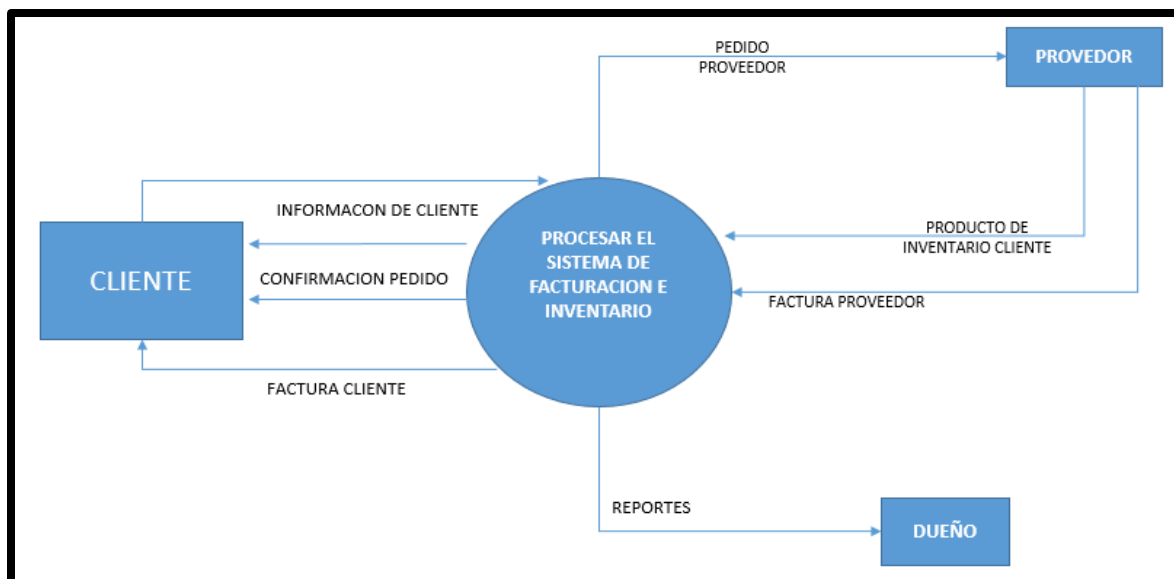
| ID | Requisito No Funcionales | Descripción |
|-----------|---------------------------------|--|
| RNF-001 | Rendimiento | El sistema debe tener la capacidad de gestionar al menos 100 usuarios simultáneos sin disminuir el desempeño. |
| RNF-002 | Seguridad | Es necesario establecer acciones de seguridad para almacenar la información de los clientes y las transacciones. |
| RNF-003 | Usabilidad | La interacción debe ser sencilla e intuitiva, facilitando a los usuarios la finalización de tareas en menos de 3 clics. |
| RNF-004 | Escalabilidad | El sistema debe tener la capacidad de escalabilidad para resistir un incremento en la carga de usuarios y datos sin requerir una reorganización. |
| RNF-005 | Mantenibilidad | El código debe estar registrado y organizado para simplificar actualizaciones y mantenimiento futuros. |
| RNF-006 | Compatibilidad | Es necesario que sea compatible con navegadores web actuales y aparatos móviles. |
| RNF-007 | Disponibilidad | El sistema debe mantenerse operativo al menos el 99.5% del tiempo, notificando a los usuarios los periodos de inactividad programados. |
| RNF-008 | Accesibilidad | Debe cumplir con las normas de accesibilidad para garantizar que personas con discapacidad puedan utilizar el sistema. |
| RNF-009 | Tiempo de Carga | El tiempo para que el sistema responda a las acciones de los usuarios no debe exceder los 2 segundos. |

Diagrama de Casos de Usos

Se aplicaría para identificar y definir las relaciones entre las diferentes acciones y el programa. Por ejemplo, un cliente podría buscar productos y realizar compras a través de una interacción amigable, mientras que también podría gestionar citas para el mantenimiento de su bicicleta y ver su historial de compras. El administrador tendría acceso a funciones para actualizar el inventario y gestionar las cuentas de los usuarios, garantizando que la tienda esté siempre al día. Por otro lado, el mecánico podría registrar los servicios de mantenimiento realizados.

Figura 3

Diagrama de Casos de Usos

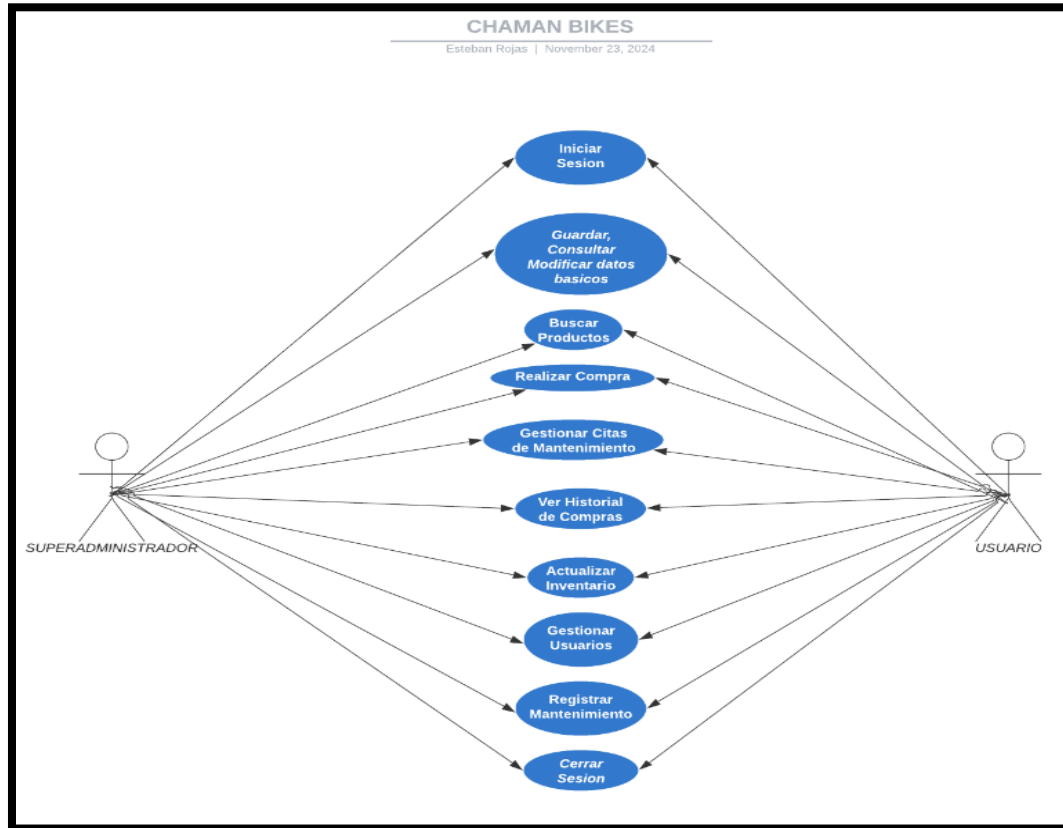


Casos de Uso

- **Buscar Productos.** El cliente puede buscar bicicletas y accesorios.
- **Realizar Compra.** El cliente selecciona productos y procede al pago.
- **Gestionar Citas de Mantenimiento.** El cliente programa citas para mantenimiento de la bicicleta.
- **Ver Historial de Compras.** El cliente puede ver sus compras anteriores.
- **Actualizar Inventario.** El administrador añade o elimina productos del inventario.
- **Gestionar Usuarios.** El administrador puede crear, editar o eliminar cuentas de clientes.
- **Registrar Mantenimiento.** El mecánico registra el mantenimiento realizado en las bicicletas.

Figura 4

Relación de los Casos de Usos



Relaciones Entre Casos de Uso

Includes. La relación "Includes" en casos de uso se utiliza para indicar que un caso de uso principal incorpora otro. Por ejemplo, al programar mantenimiento, el cliente debe registrar la bicicleta si no lo ha hecho previamente. Al realizar mantenimiento, el técnico consulta el historial de la bicicleta. Al generar factura, es necesario registrar el mantenimiento realizado. Esta relación permite reutilizar casos de uso, mejorar el modularidad y facilitar la comprensión del sistema.

Extends. La relación "Extends" se utiliza en casos de uso para indicar que un caso de uso opcional puede añadir acciones a otro caso de uso base en situaciones específicas. Por ejemplo, el caso de uso realizar mantenimiento puede extenderse para incluir enviar notificación al cliente si se detectan problemas adicionales. Otro ejemplo sería que generar factura puede extenderse para aplicar descuentos si el cliente es miembro. Esta relación permite flexibilidad y personalización, añadiendo funcionalidades sin modificar el flujo principal.

Figura 5

Diagrama General

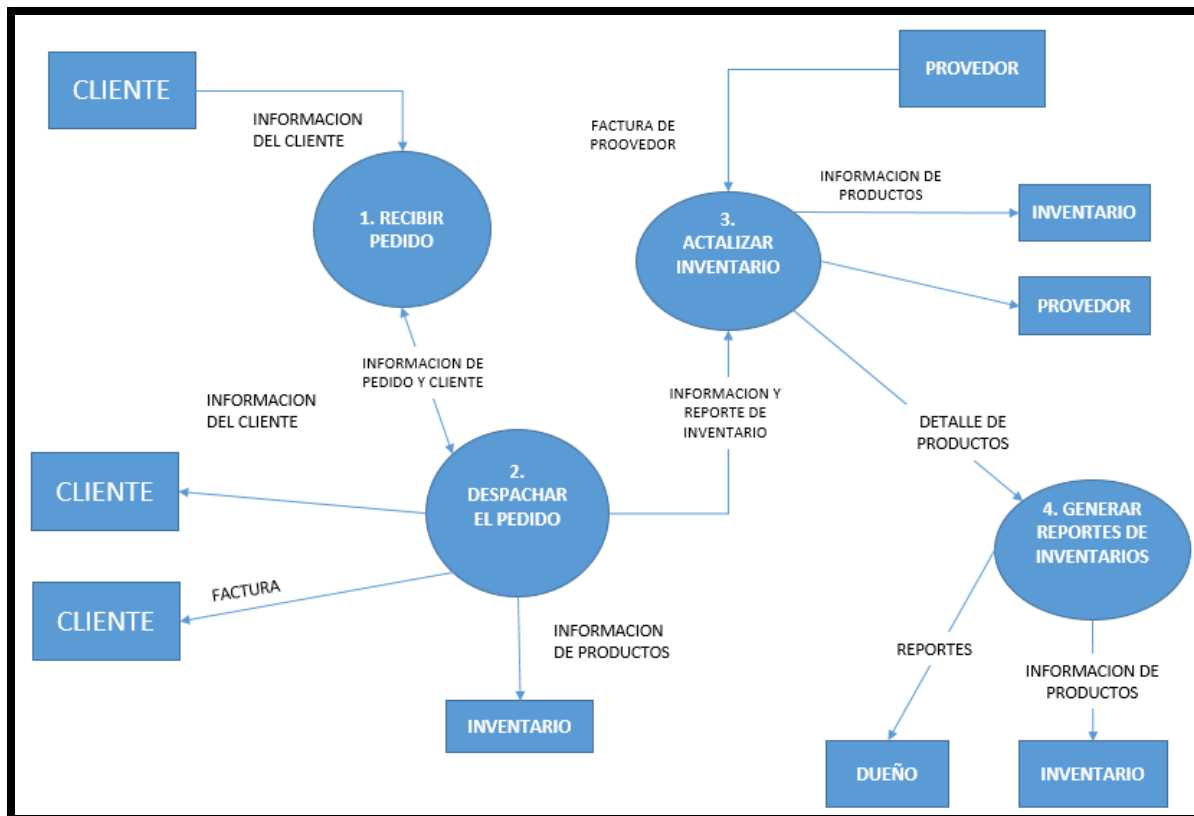


Figura 6

Despachar Pedido

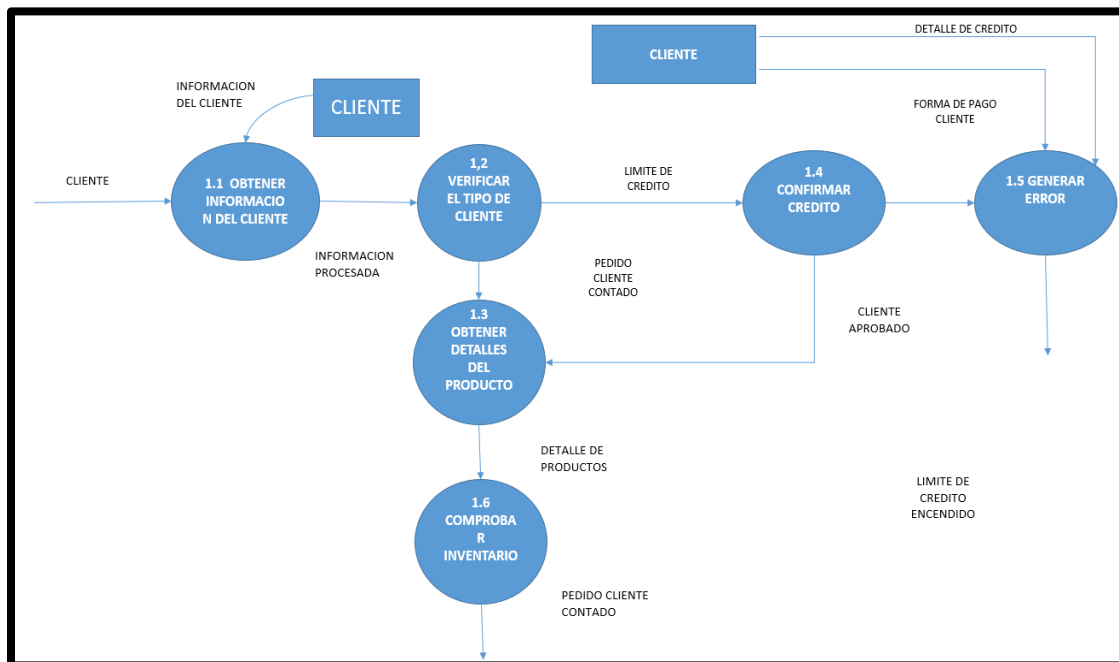


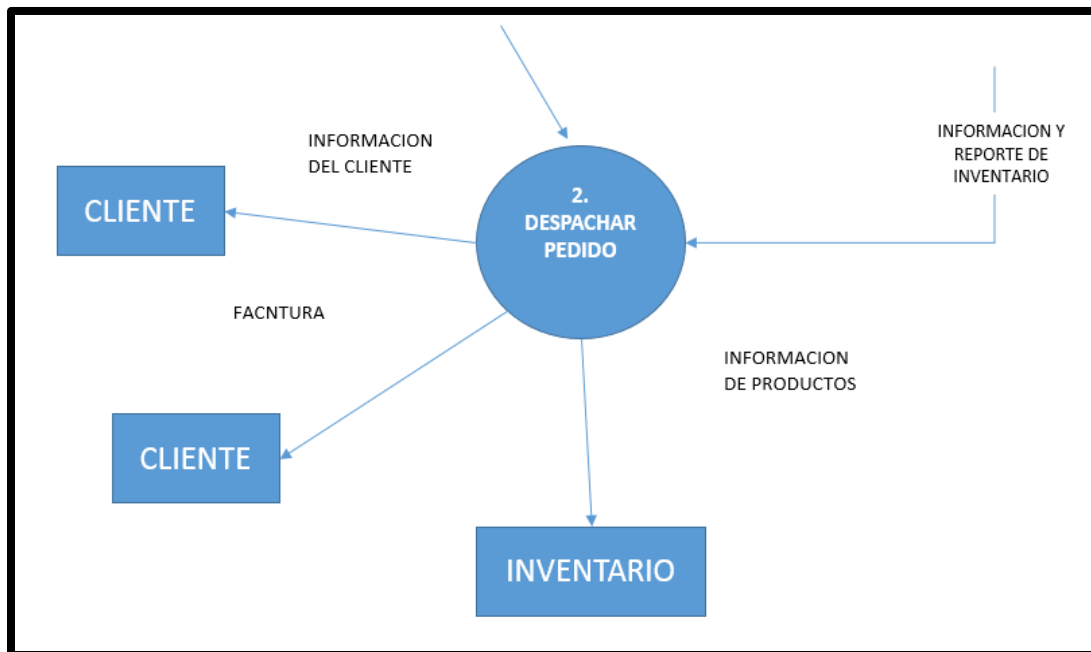
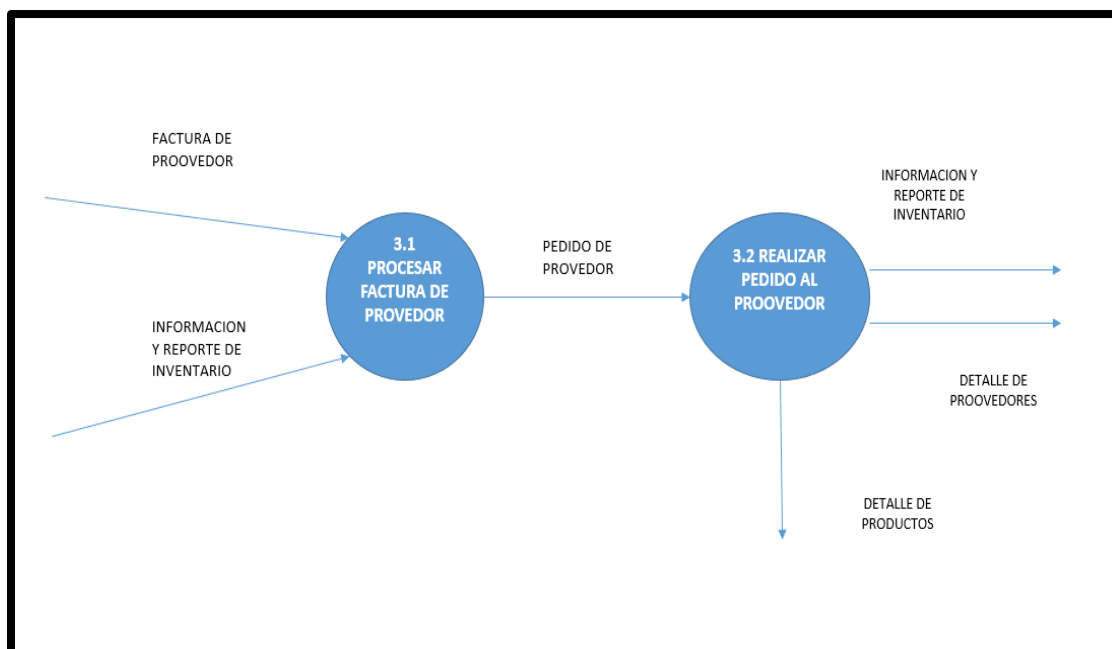
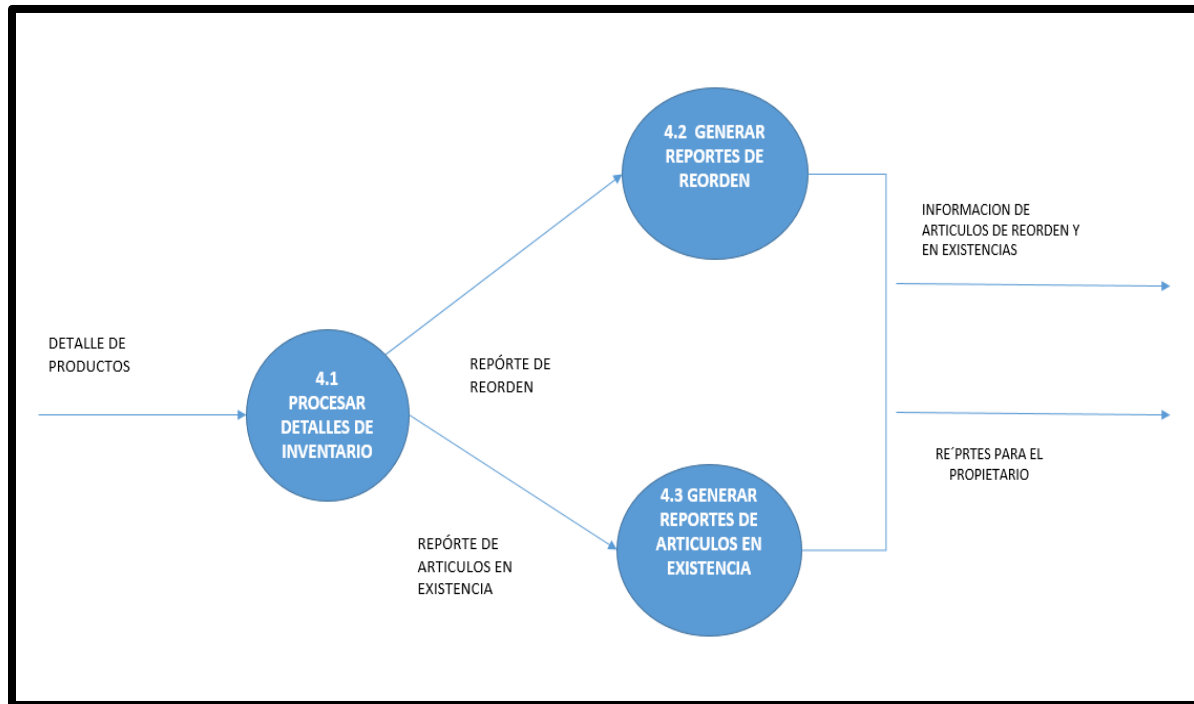
Figura 7*Despachar Pedido***Figura 8***Gestionar Inventario*

Figura 9*Generar Reportes de Inventario*

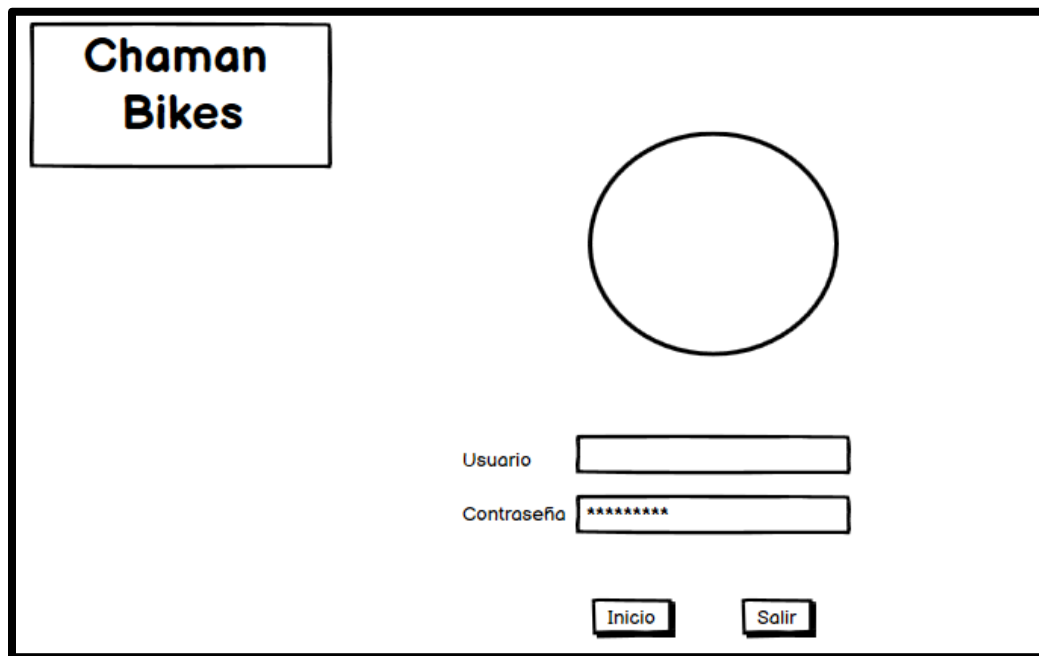
MockUps

Logueo de Usuarios

El sistema de logueo de usuarios deja que los usuarios ingresen a la plataforma de forma segura a través de la adición de su nombre de usuario y contraseña.

Figura 10

Logueo de usuarios



The mockup shows a login interface for 'Chaman Bikes'. It features a logo in the top left corner, a large empty circle in the center, and two input fields for 'Usuario' and 'Contraseña'. Below the input fields are two buttons labeled 'Inicio' and 'Salir'.

**Chaman
Bikes**

Usuario

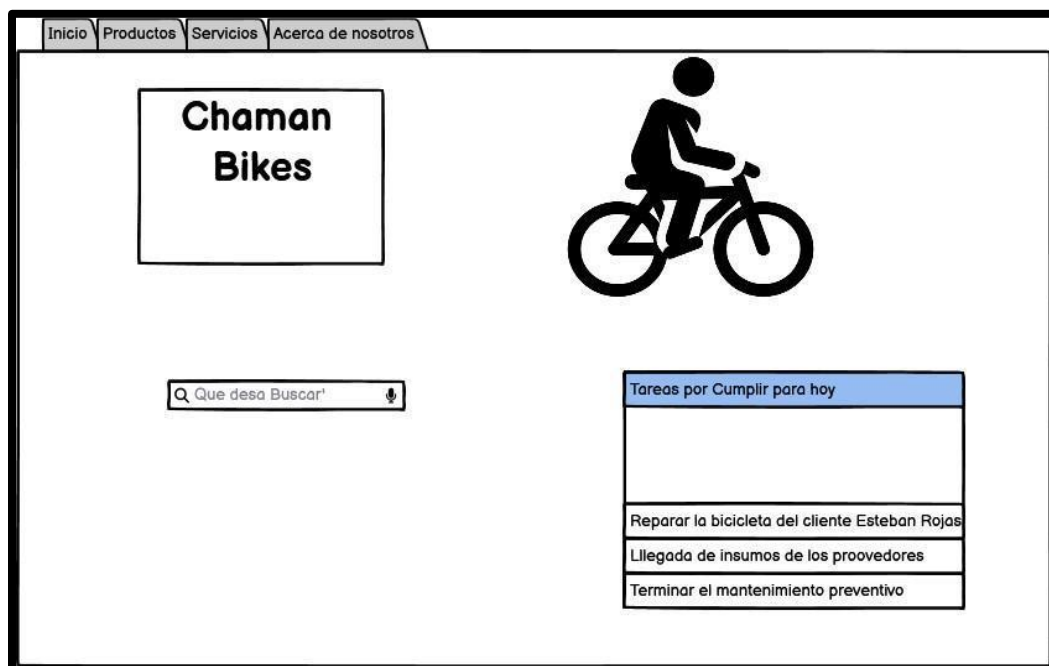
Contraseña

Página Principal

Desde aquí, los usuarios pueden acceder a funciones clave, como la consulta de productos, la programación de mantenimientos y la visualización del historial de transacciones.

Figura 11

Página Principal



Servicios Realizados

La pestaña de servicios realizados ofrece a los usuarios un historial de todos los servicios que han realizado, como mantenimientos y reparaciones. Cada registro incluye detalles como la fecha del servicio, el tipo de trabajo realizado y el costo asociado. Esta sección permite a los usuarios revisar su historial de servicios de manera fácil y rápida además de que se puede agregar, modificar o eliminar uno o varios servicios.

Figura 12

Servicios Realizados

The screenshot shows a web interface with a navigation bar at the top containing 'Inicio', 'Productos', 'Servicios', and 'Acerca de nosotros'. The main content area is titled 'Servicios Realizados' and contains a table with the following data:

| Articulo | Age | Nickname | Precio |
|-------------------------------------|-----|----------|-------------------------------------|
| Giacomo Guilizzoni Founder & CEO | 40 | Peldi | <input type="radio"/> |
| Marco Botton Tuttofare | 38 | | <input checked="" type="checkbox"/> |
| Mariah Maclachlan Better Half | 41 | Patata | <input type="checkbox"/> |
| Valerie Liberty Head Chef | :) | Val | <input checked="" type="checkbox"/> |
| Data Grid Docs | | | <input type="checkbox"/> |
| | | | |
| | | | |
| | | | |
| | | | |

Below the table are three circular icons: a plus sign for 'Agregar', a crossed-out circle for 'Eliminar', and a pencil for 'Modificar'. To the right of these icons are two buttons: 'Aceptar' and 'Cancelar'.

Productos

La pestaña de productos presenta un catálogo completo de los artículos disponibles para la venta, incluyendo accesorios, repuestos y componentes para bicicletas. Cada producto cuenta con una imagen, descripción detallada, precio y opciones de compra. Los usuarios pueden filtrar y buscar productos por categorías o características específicas, facilitando la navegación y la selección.

Figura 13

Panel de Productos

| Articulo | Age | Nickname | Precio |
|-------------------------------------|-----|----------|-------------------------------------|
| Giacomo Guilizzoni Founder & CEO | 40 | Peldi | <input type="radio"/> |
| Marco Botton Tuttofare | 38 | | <input checked="" type="checkbox"/> |
| Mariah Maciachian Better Half | 41 | Patata | <input type="checkbox"/> |
| Valerie Liberty Head Chef | :) | Val | <input checked="" type="checkbox"/> |
| Data Grid Docs | | | <input type="checkbox"/> |
| | | | |
| | | | |
| | | | |

Reportes

Los usuarios pueden generar reportes sobre el historial de servicios realizados, transacciones de compra y ventas, así como el estado del inventario. Cada reporte ofrece datos analíticos, gráficos y estadísticas que ayudan a los usuarios a comprender mejor el rendimiento del taller y la toma de decisiones.

Figura 14

Reportes

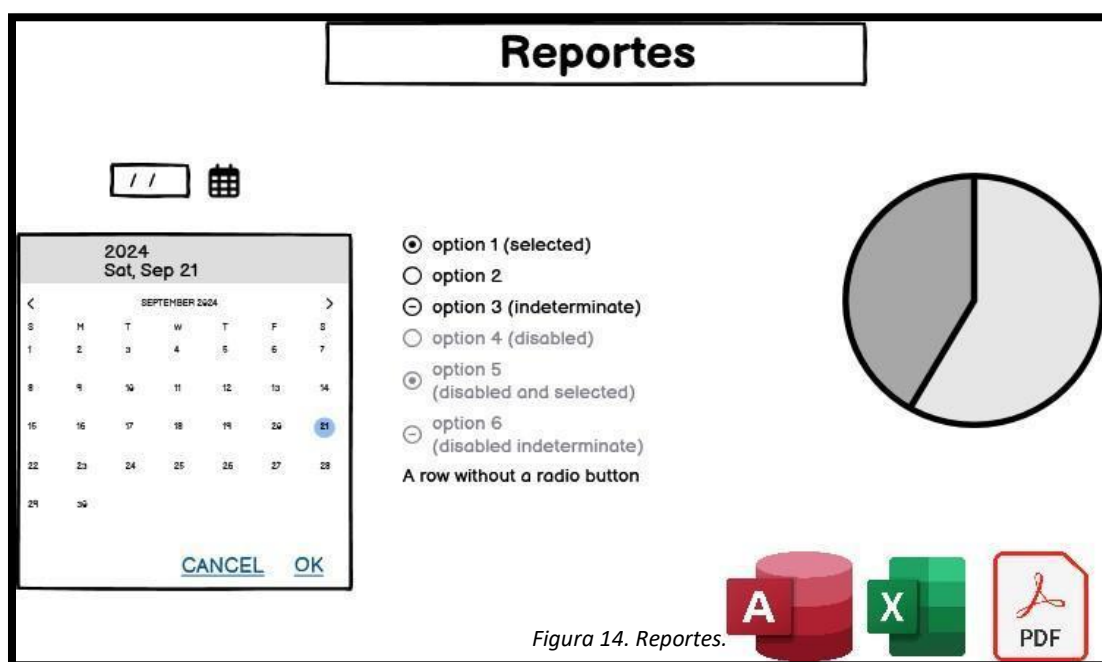


Figura 14. Reportes.

Información General

Esta sección tiene como objetivo informar a los usuarios sobre la identidad de la empresa, su dedicación hacia la excelencia y el servicio al cliente y su compromiso con la calidad, así como las políticas de servicio y atención. Además, puede incluir enlaces a redes sociales y formas de contacto, facilitando así la interacción entre la compañía y sus consumidores.

Figura 15

Informacion General



Diseño del Prototipo de la Base de Datos

Figura 16

Prototipo Modelo Entidad Relación

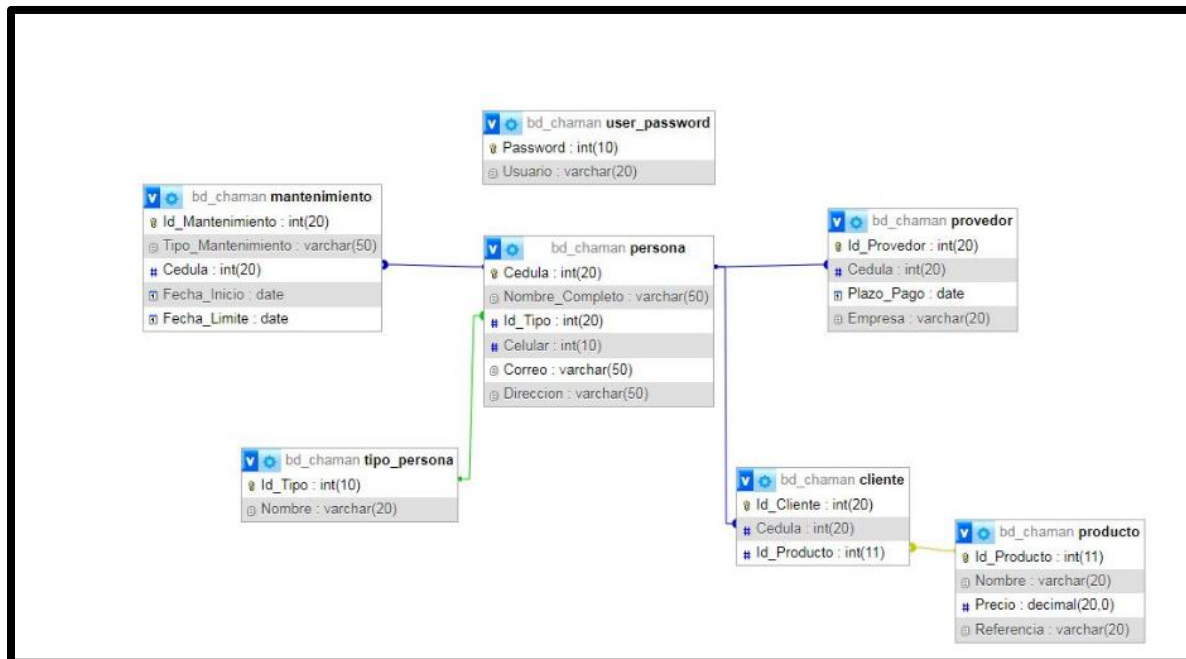


Figura 17

Prototipo Base de Datos

El software muestra un listado de tablas con las siguientes columnas: Tabla, Acción, Filas, Tipo, Cotejamiento, Tamaño y Residuo a depurar.

| Tabla | Acción | Filas | Tipo | Cotejamiento | Tamaño | Residuo a depurar |
|-----------------|--|----------|---------------|---------------------------|-----------------|-------------------|
| cliente | Examinar, Estructura, Buscar, Insertar, Vaciar, Eliminar | 0 | InnoDB | utf8mb4_general_ci | 48.0 KB | - |
| mantenimiento | Examinar, Estructura, Buscar, Insertar, Vaciar, Eliminar | 0 | InnoDB | utf8mb4_general_ci | 32.0 KB | - |
| persona | Examinar, Estructura, Buscar, Insertar, Vaciar, Eliminar | 0 | InnoDB | utf8mb4_general_ci | 32.0 KB | - |
| producto | Examinar, Estructura, Buscar, Insertar, Vaciar, Eliminar | 0 | InnoDB | utf8mb4_general_ci | 16.0 KB | - |
| proveedor | Examinar, Estructura, Buscar, Insertar, Vaciar, Eliminar | 0 | InnoDB | utf8mb4_general_ci | 32.0 KB | - |
| tipo_persona | Examinar, Estructura, Buscar, Insertar, Vaciar, Eliminar | 0 | InnoDB | utf8mb4_general_ci | 16.0 KB | - |
| user_password | Examinar, Estructura, Buscar, Insertar, Vaciar, Eliminar | 0 | InnoDB | utf8mb4_general_ci | 16.0 KB | - |
| 7 tablas | Número de filas | 0 | InnoDB | utf8mb4_general_ci | 192.0 KB | 0 B |

En la parte inferior, se muestra un formulario para crear una nueva tabla:

Nombre de la tabla:

Número de columnas:

Crear

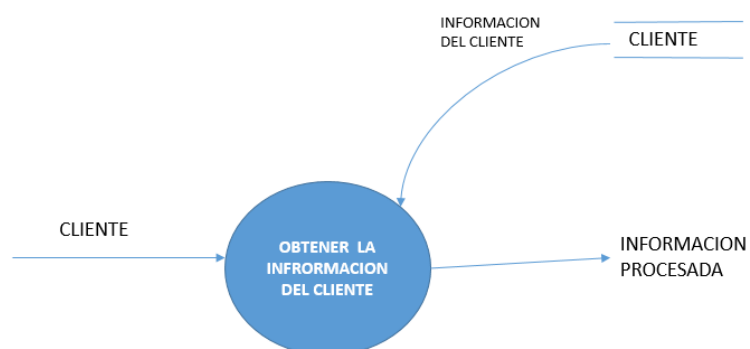
Diccionario de Datos

Un diccionario de datos es una recopilación o repositorio de significados, descripciones y características de los subsistemas de información que deberán ser utilizados en un sistema como, por ejemplo, tablas, campos y relaciones que se utilicen en un sistema de base de datos. Esta información es relevante para los desarrolladores o los integrantes del equipo de trabajo; gracias al diccionario de datos, todos los participantes en el desarrollo de un sistema comparten una concepción clara y uniforme de los datos que se encuentran en el programa, facilitando la administración y el mantenimiento de este. En el software Chaman Bikes un diccionario de datos incluye definiciones para las entidades Bicicleta Usuario Mantenimiento y Proveedor ya que para una entidad Bicicleta define el diccionario de datos un campo ID_Bicicleta, que es un identificador único de cada bicicleta y es un número entero, otro campo Modelo que es una cadena de texto que define el modelo de la bicicleta y otros detalles. También incluiría la definición de cómo se relacionan estos datos, por ejemplo, tal Usuario puede tener varias Bicicletas o el Mantenimiento está asociado a una bicicleta.

Por tanto, el diccionario de datos permite no solo familiarizarse con las definiciones de los datos involucrados en el sistema sino también para que los desarrolladores y los usuarios de este puedan conocer cómo interaccionan los datos entre diferentes subsistemas de los datos y aquellos sistemas dependientes de los elementos que se han definido, favoreciendo la calidad e incluso la eficacia de desarrollo y el mantenimiento del sistema.

Tabla 20*Obtener Información del Cliente*

| | |
|---------------|---------------------------------|
| Nombre | Obtener Información del Cliente |
|---------------|---------------------------------|

**Grafico**

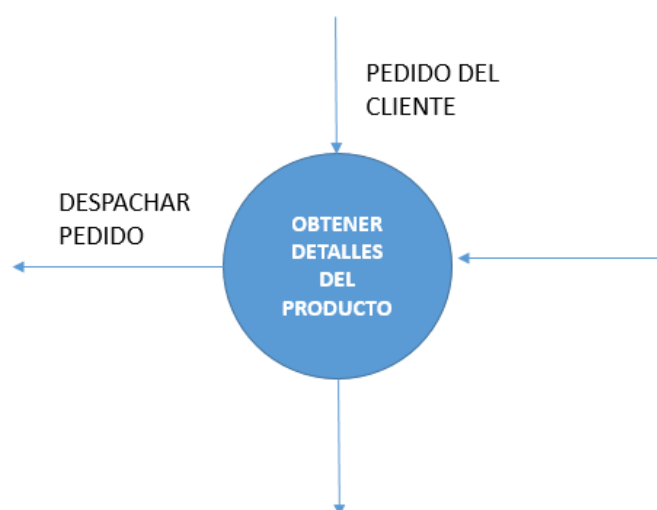
Descripción Validar las características del cliente

Flujo de Ingresos Archivo Cliente

Flujo de Egreso Información Procesada

Tabla 21*Obtener Informacion del Producto*

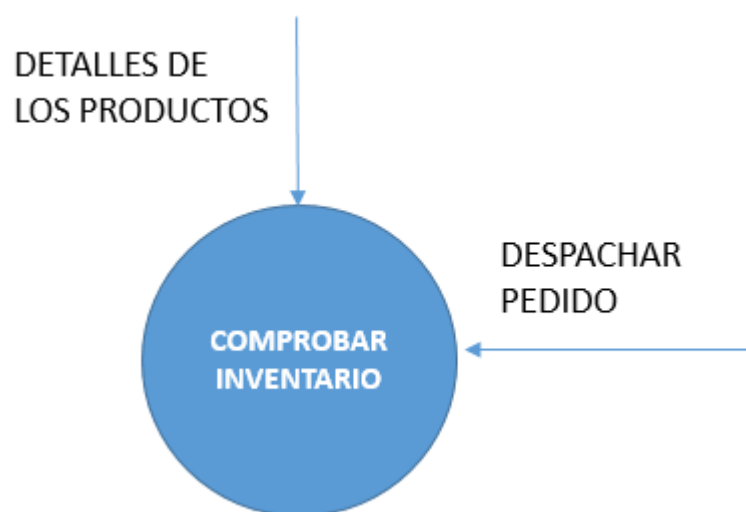
| Nombre | Obtener detalles del Producto |
|--------|-------------------------------|
|--------|-------------------------------|

**Grafico**

| | |
|--------------------------|--|
| Descripción | Analiza el trámite del producto que tiene el cliente |
| Flujo de Ingresos | Pedido de cliente |
| Flujo de Egreso | Despachar pedido. |

Tabla 22*Comprobar Inventario*

| Nombre | Comprobar Inventario |
|--------|----------------------|
|--------|----------------------|

**Grafico**

| | |
|--------------------------|---|
| | Este sucede desde el almacenaje de los productos hasta el |
| Descripción | stock. |
| Flujo de Ingresos | Detalle de productos |
| Flujo de Egreso | Información de productos. |

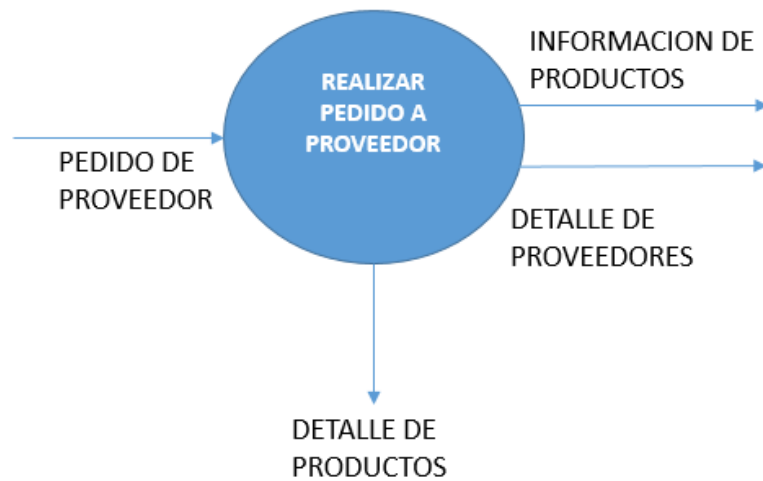
Tabla 23

Procesar Factura del Proveedor

| Nombre | Procesar Factura del Proveedor |
|--------------------------|--|
| Grafico | <pre> graph TD A[INGRESAR PRODUCTOS] --> B((PROCESAR LA FACTURA DE PROVEEDOR)) B --> C[PRODUCTOS DE INVENTARIO] B --> D[PEDIDO PROVEDOR] </pre> |
| Descripción | Los productos del inventario que provienen del proveedor son introducidos. |
| Flujo de Ingresos | Archivo de la factura de proveedor |
| Flujo de Egreso | Pedido proveedor, productos de inventario. |

Tabla 24*Realizar Pedido a Proveedor*

| Nombre | Realizar Pedido a proveedor |
|--------|-----------------------------|
|--------|-----------------------------|

**Grafico****Descripción**

Efectúa pedido al proveedor y maneja datos de productos.

Flujo de Ingresos

Archivo factura de proveedor

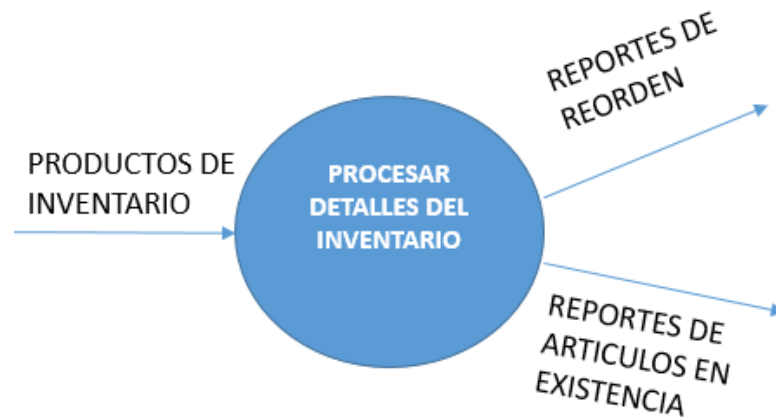
Detalle productos, información de productos y detalle de proveedores
Detalle productos, información de productos y

Flujo de Egreso

detalle de proveedores

Tabla 25*Procesar Detalles de Inventario*

| Nombre | Procesar Detalles de Inventario |
|--------|---------------------------------|
|--------|---------------------------------|

**Grafico**

Es un procedimiento en el que se recolectan la información

Descripción

necesaria y se comprueban los costos y la disponibilidad.

Flujo de Ingresos

Productos de inventarios.

Flujo de Egreso

Reportes de orden y reportes de artículos en existencia

Tabla 26*Procesar Detalles de Inventario*

| | |
|---------------|-----------------------------|
| Nombre | Generar Reportes de Reorden |
|---------------|-----------------------------|

**Grafico****Descripción**

Se tramitan y se genera la ficha de la mercancía en reorden.

Flujo de ingresos

Artículos en reorden

Flujo de Egreso

Información de artículos de reorden.

Tabla 27*Generar Reportes de Artículos en Existencia*

| | |
|---------------|--|
| Nombre | Generar Reportes de Artículos en Existencia. |
|---------------|--|

**Grafico**

Se produce la información actual sobre los productos

Descripción presentes en el taller.

Flujo de ingresos Reportes.

Flujo de Egreso Información de artículos en existencia.

Tabla 28*Compras*

| Compras | | | | | |
|----------------|-----------------|--------------|-------------|---------------|-------------------------------------|
| Llave | Nombre | Campo | Tipo | Tamaño | Descripción |
| PK | ID compras | id | Int | | almacena el Código de las compras |
| FK | ID Productos | idproducto | Int | | Almacena el Código de cada producto |
| | Fecha de compra | fecha compra | Date Time | | Almacena la fecha de compra |
| | Cantidad | cantidad | Int | | Almacena la cantidad del producto |
| | Precio | precio | Doble | | Almacena el precio |

Tabla 29*Detalle Factura*

| Detalle Factura | | | | | |
|------------------------|----------------|--------------|-------------|---------------|--|
| Llave | Nombre | Campo | Tipo | Tamaño | Descripción |
| PK | Cedula Código | id | INT | | almacena el Código del detalle de la factura |
| FK | Factura ID del | idfactura | INT | | almacena el Código de las facturas |
| FK | producto | idproducto | INT | | Almacena el Código de cada producto |
| | Cantidad | Cantidad | INT | | Almacena la cantidad del producto |
| | Precio | Precio | INT | | Almacena el precio |

Tabla 30*Mantenimiento Preventivo*

| Detalle Mantenimiento Preventivo | | | | | |
|---|---------------|--------------|-------------|---------------|------------------------------------|
| Llave | Nombre | Campo | Tipo | Tamaño | Descripción |
| | ID del | | | | almacena el Código del |
| PK | mantenimiento | id | INT | | mantenimiento preventivo |
| FK | ID Factura | idmt | INT | | almacena el Código de las facturas |
| | | | | | Almacena el Código de cada |
| FK | ID producto | idproducto | INT | | producto |
| | Cantidad | Cantidad | INT | | Almacena la cantidad del producto |
| | Precio | Precio | INT | | Almacena el precio |

Tabla 30*Mantenimiento Preventivo*

| Mantenimiento Preventivo | | | | | |
|---------------------------------|--------------------|--------------|-------------|----------------------|-----------------------|
| Llave | Nombre | Campo | Tipo | Tamaño | Descripción |
| Pk | Cedula | Cedula | Varchar | 50 | Almacena la cedula |
| | Nombre | Nombre | Varchar | 50 | Almacena el nombre |
| | Apellido | Apellido | Varchar | 50 | Almacena el apellido |
| | | | | | Almacena el nombre |
| | Teléfono | Teléfono | Int | | de la ciudad |
| | Celular | Celular | Int | | Almacena el celular |
| | | | | | Almacena la |
| | Dirección | Dirección | Varchar | 100 | dirección |
| | | | | | Almacena el correo |
| | Correo | Correo | Varchar | 100 | electrónico |
| | | | | | Almacena el serial de |
| | Serial Bicicleta | Serialbici | Varchar | 20 | la bicicleta |
| | | | | | Almacena la marca |
| | Marca Bicicleta | Marcabici | Varchar | 100 | de la bicicleta |
| | | | | Almacena el color de | |
| Color Bicicleta | Colorbici | Varchar | 50 | la bicicleta | |
| | | | | Almacena la fecha el | |
| Fecha de Inicio | Fechaentrada | Date | | inicio del | |
| | | | | mantenimiento | |
| Fecha de | | | | Almacena la fecha | |
| Vencimiento | Fechasalida | Date | | del fin del | |
| | | | | mantenimiento | |
| Disponibilidad de | | | | Almacena | |
| Almacenamiento | Diasalmacenamiento | | | disponibilidad del | |
| | | | Int | mantenimiento | |

Tabla 31*Facturas*

| Facturas | | | | | |
|-----------------|----------------------|------------------|-------------|---------------|---|
| Llave | Nombre | Campo | Tipo | Tamaño | Descripción |
| PK | Código de la factura | idfactura | INT | | Almacena el Código del mantenimiento preventivo |
| | Razón social | razonsocial | Varchar | 150 | Almacena la razón social |
| | Teléfono | Telefono | Varchar | 20 | Almacena el Teléfono |
| | Celular | celular | Varchar | 30 | Almacena el número del celular |
| | Ciudad | ciudad | Varchar | 100 | Almacena el nombre de la ciudad |
| | Dirección | direccion | Varchar | 100 | Almacena la dirección |
| | Fecha de elaboración | fechaelaboracion | DATE | | Almacena la fecha de la elaboración |
| | Forma de pago | formapago | Varchar | 50 | Almacena la forma de pago |
| | contacto | contacto | Varchar | 100 | Almacena el contacto del cliente |
| | Nombre del cliente | nombrecliente | Varchar | 100 | Almacena el nombre del cliente |
| | Correo | correo | Varchar | 100 | Almacena el correo electrónico |
| | Valor de la factura | valorfactura | INT | | Almacena el valor de la factura |

Tabla 32*Productos*

| Productos | | | | | |
|------------------|--------------------|--------------|-------------|---------------|---------------------------------------|
| Llave | Nombre | Campo | Tipo | Tamaño | Descripción |
| | Código | | | | Almacena el Código del |
| PK | Producto | idproducto | INT | | producto |
| | Nombre | nombre | Varchar | 100 | Almacena el nombre |
| | Descripción | descripcion | Varchar | 255 | Almacena la descripción |
| | Cantidad | cantidad | INT | | Almacena la cantidad |
| | Precio | precio | INT | | Almacena el precio |
| | Precio de venta | precioventa | INT | | Almacena el precio de venta |
| | Imagen | imagen | Varchar | 100 | Almacena la información de la foto |

Tabla 34*Ventas*

| Ventas | | | | | |
|---------------|--------------------|--------------|-------------|---------------|---------------------------------|
| Llave | Nombre | Campo | Tipo | Tamaño | Descripción |
| PK | Código de la venta | id | INT | | Almacena el Código de la venta |
| FK | Código Producto | idproducto | INT | | Almacena el Código del producto |
| | Fecha de venta | fechaventa | DATE | | Almacena la fecha de la venta |
| | Cantidad | cantidad | INT | | Almacena la cantidad |
| | Precio | precio | INT | | Almacena el precio total |

Tabla 35*Usuarios*

| Usuarios | | | | | |
|-----------------|----------------------------|--------------|-------------|---------------|---------------------------------|
| Llave | Nombre | Campo | Tipo | Tamaño | Descripción |
| PK | Identificación del usuario | IdUsuario | INT | | Almacena el Código del usuario |
| | Contraseña | Pass | Varchar | 200 | Almacena la contraseña |
| | Usuario | UserName | Varchar | 40 | Almacena el usuario |
| | Nivel del usuario | Nivel | Varchar | 30 | Almacena el nivel de usuario |
| | Pin de verificación | Pin | Varchar | 30 | Almacena el pin de verificación |
| | Nombre | Nombre | Varchar | 60 | Almacena el nombre del usuario |
| | Correo | Email | Varchar | 100 | Almacena el correo electrónico |

Modelo Entidad Relación

En un software de mantenimiento preventivo para bicicletas, se puede aplicar el Modelo Entidad-Relación (MER) para estructurar la base de datos. Las entidades clave incluyen Bicicleta, que tendría atributos como ID_Bicicleta, modelo y estado; Mantenimiento, con ID_Mantenimiento, fecha, tipo y una referencia a la bicicleta; Usuario, que contendría nombre y contacto; y Proveedor de Servicio, con información sobre el proveedor. Las relaciones son fundamentales. una bicicleta puede tener múltiples mantenimientos, estableciendo una relación de uno a muchos entre Bicicleta y Mantenimiento. Cada mantenimiento está vinculado a un proveedor específico, creando otra relación similar. También, un usuario puede poseer varias bicicletas, lo que refuerza la relación de uno a muchos.

Codificación

La codificación del software Chaman Bikes, diseñado para el mantenimiento de bicicletas, se basa en un modelo de tres capas que optimiza su construcción y funcionalidad. En la capa de presentación, se desarrolla una plataforma fácil de entender que autoriza a los usuarios gestionar manera fácil sus citas y acceder a información sobre el estado de sus bicicletas. La fase lógica de negocio, donde se implementan las consignas y procesos fundamentales, se encarga de gestionar el historial de mantenimiento, calcular costos y programar alertas para servicios necesarios.

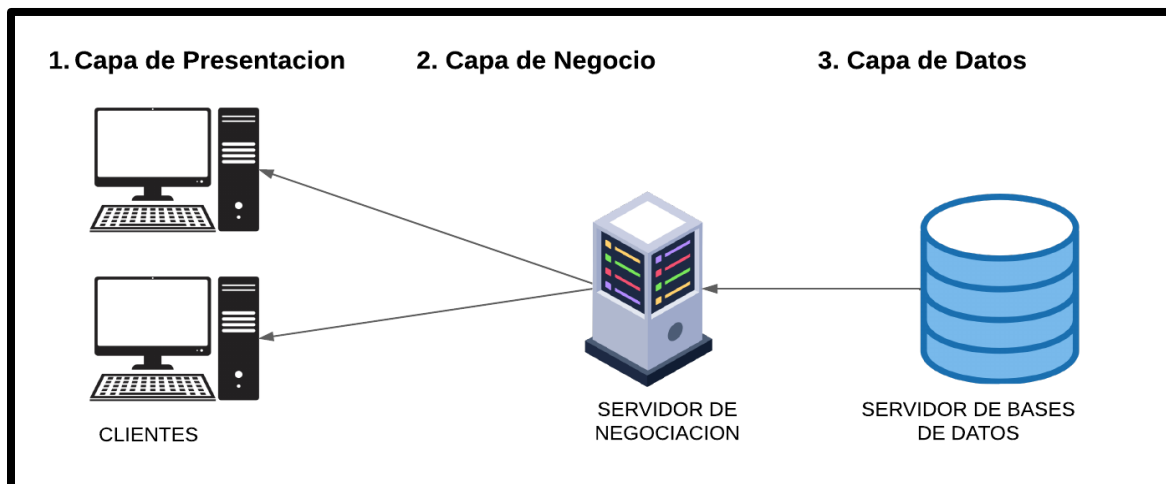
Modelo de Tres Capas

En el software Chaman Bikes, la arquitectura de tres niveles se segmenta en tres niveles vinculados que distinguen la presentación, la lógica empresarial y el acceso a datos, lo que facilita su desarrollo y mantenimiento.

- **Capa de Presentación.** Esta está a cargo la interacción de usuario y la experiencia del cliente. Aquí se desarrollan las pantallas y elementos visuales que permiten a los ciclistas y mecánicos interactuar con el sistema, programar citas y consultar el historial de mantenimiento. Su diseño se enfoca en ser intuitivo y accesible.
- **Capa de Lógica de Negocio.** En esta capa se gestionan todas las reglas y procesos del sistema. Se implementan funciones que controlan la lógica del mantenimiento, como la programación de servicios, la gestión de inventarios y la generación de informes. Esta capa es fundamental para garantizar que los trabajos se realicen de manera coherente y eficiente.
- **Capa de Datos.** Esta capa se ocupa de guardar y recuperar la información. Utiliza bases de datos para gestionar datos sobre bicicletas, piezas y registros de servicio. Se encarga de cuidar la integridad y el amparo de los datos, permitiendo un acceso rápido y confiable.

Figura 18

Modelo de Tres (3) Capas



Mantenimiento

El mantenimiento adecuado de las bicicletas es fundamental para asegurar la seguridad, el desempeño y la durabilidad de estos vehículos. Este proceso realiza inspecciones y ajustes estándar, como inspecciones de frenos, lubricantes de cadenas e inspecciones de neumáticos para identificar problemas antes de que se agraven. Al adoptar un programa de mantenimiento regular, los ciclistas no sólo mejoran su seguridad al evitar accidentes, sino que también ahorran costos a largo plazo al evitar reparaciones importantes. y se fomenta el carácter, los viajes sostenibles son una inversión inteligente

Prueba de Funcionamiento con Jmeter

Para dar cumplimiento con el cuarto objetivo las pruebas en el software Chaman Bikes son esenciales para garantizar su calidad y funcionamiento adecuado en el mantenimiento de bicicletas.

Se realiza la prueba de carga con Jmeter En las pruebas de carga realizadas con JMeter, se configuraron 4 hilos de carga para simular la interacción de cuatro usuarios concurrentes con el sistema. Cada hilo representa a un usuario virtual que ejecuta las mismas acciones definidas en el plan de prueba, como realizar solicitudes HTTP o completar formularios. Al aumentar el número de hilos, se evalúa cómo responde el sistema bajo un pequeño pero significativo número de usuarios simultáneos, lo que promueve la identificación de posibles fallas o cuellos de botella de rendimiento cuando se presenta una carga moderada. Esta configuración permite obtener una visión general de la capacidad del sistema para manejar

- Se ingresa el protocolo de la aplicación teniendo en cuenta el nombre del servidor y se aplica el método GET

Figura 19

Verificación del Servidor del Programa

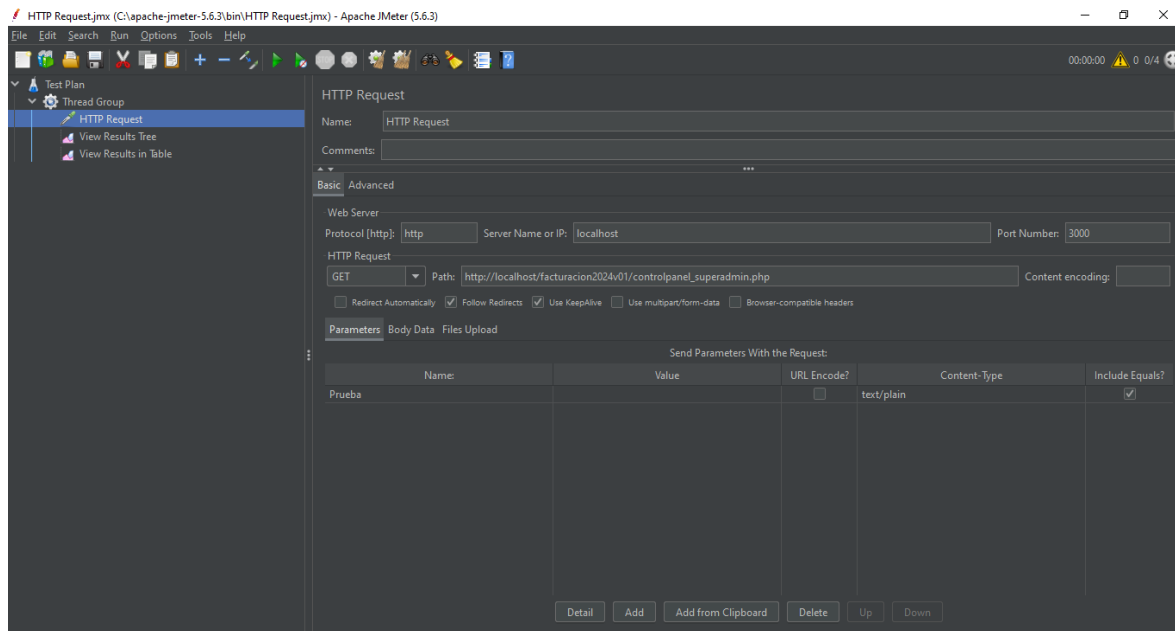


Figura 20

Carga Exitosa de los Cuatro (4) Hilos Aplicados en el Programa De Verificación

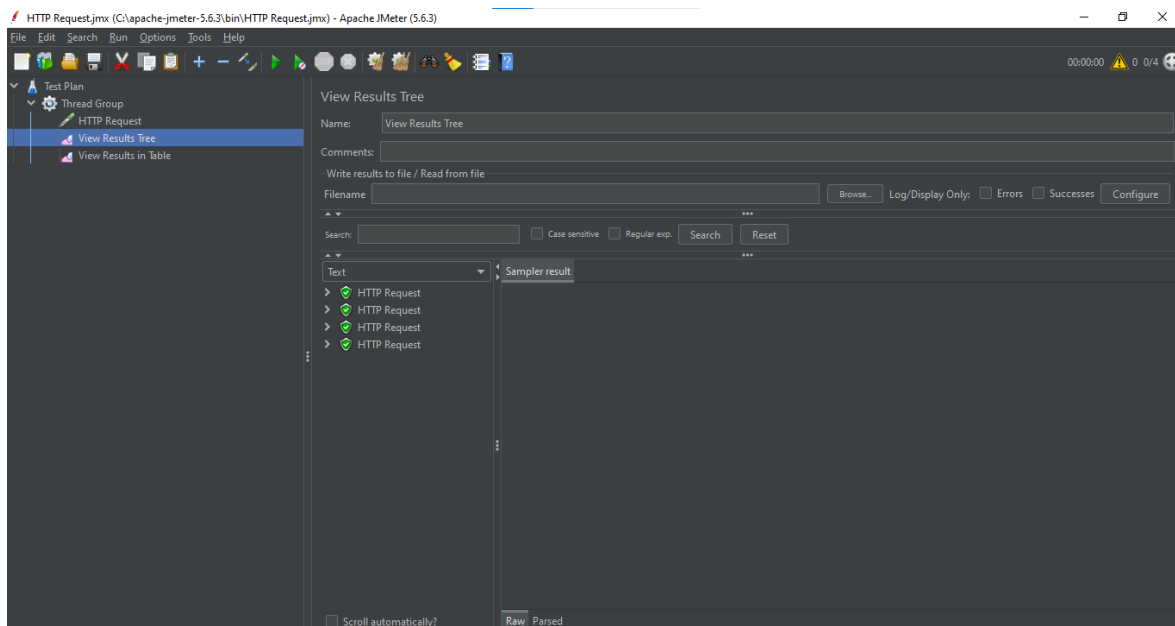


Figura 21

Descripción Específica del Éxito de Carga e Cada Hilo

View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename: Browse... Log/Display Only: Errors Successes

| Sample # | Start Time | Thread Name | Label | Sample Time... | Status | Bytes | Sent Bytes | Latency | Connect Time... |
|----------|--------------|-------------------|--------------|----------------|--------|-------|------------|---------|-----------------|
| 1 | 10:33:35.751 | Thread Group 1... | HTTP Request | 17 | ✓ | 46452 | 302 | 12 | 1 |
| 2 | 10:33:36.006 | Thread Group 1... | HTTP Request | 19 | ✓ | 46452 | 302 | 14 | 1 |
| 3 | 10:33:36.251 | Thread Group 1... | HTTP Request | 53 | ✓ | 46452 | 302 | 25 | 1 |
| 4 | 10:33:36.499 | Thread Group 1... | HTTP Request | 55 | ✓ | 46452 | 302 | 29 | 0 |

Scroll automatically? Child samples? No of Samples: 4 Latest Sample: 55 Average: 50 Deviation: 10

Anexos

Manual de usuario

Index.Php

Figura 22

Página de Inicio

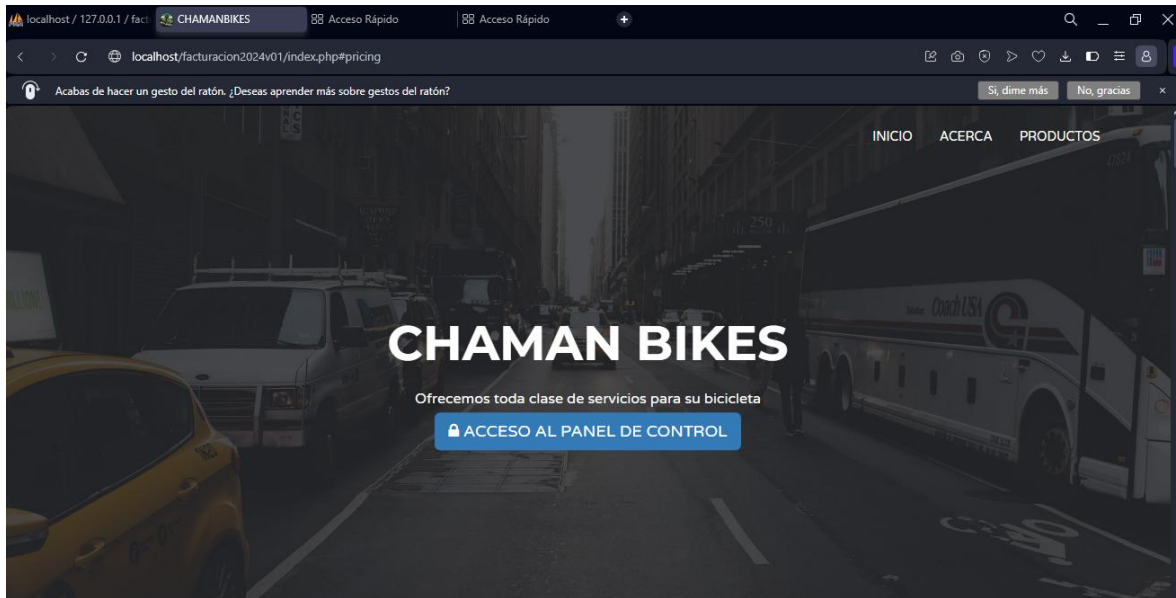
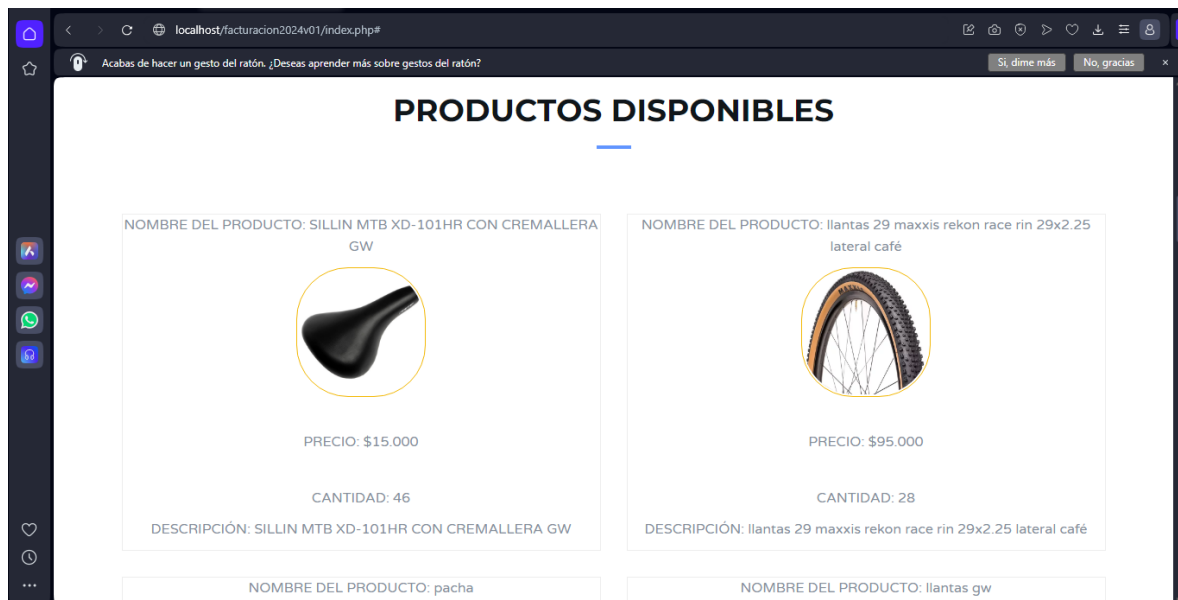


Figura 23

Productos Disponibles



Informe Explicativo de la Página index.php del Proyecto Chaman Bikes

La página index.php de Chaman Bikes es la página principal del programa de manejo de inventarios y mantenimientos de bicicletas. Su diseño se enfoca en dar una experiencia amigable y funcional tanto para los usuarios administradores como para los clientes. A continuación, se describen sus principales actividades y utilidades.

Conexión a la Base de Datos

- La página comienza estableciendo una interconexión con la información mediante el archivo cnx.php, asegurando que se puedan realizar operaciones de lectura sobre los productos disponibles.
- Se ejecuta una consulta SQL para obtener todos los artículos en la base de datos, lo que permite que la información esté siempre actualizada.

Diseño y Estructura Responsiva

- Utiliza Bootstrap y fuentes de Google para un diseño moderno y adaptable, garantizando que la página se visualice correctamente en dispositivos móviles y de escritorio.
- Se incluye una imagen de fondo con una superposición que mejora la estética visual de la página.

Navegación Intuitiva

- La barra de navegación permite acceder fácilmente a diferentes secciones de la página, como "Inicio", "Acerca" y "Productos".
- Se proporciona un enlace al panel de control, facilitando el acceso para los administradores.

Sección de Bienvenida

- En la parte superior, se presenta un mensaje de bienvenida que destaca los servicios ofrecidos por Chaman Bikes, atrayendo la atención del usuario.

Descripción del Sitio

- La sección "Acerca" proporciona información relevante sobre las características del sistema, incluyendo.

- Completamente Personalizable. Se destaca que el usuario administrador puede modificar la configuración del sistema según sus necesidades.

- En los tipos de usuarios se mencionan que hay varios tipos de usuarios que pueden manipular el sistema, lo que permite una gestión más organizada.

- Las Páginas responsivas resalta a la plataforma para que sea accesible desde cualquier dispositivo.

Listado de Productos Disponibles

- La sección "Productos Disponibles" presenta los artículos en la base de datos, mostrando.

- El nombre del producto indica el nombre de cada producto.

- La imagen del producto proporciona una visualización del producto para mejorar la presentación.

- Precio y Cantidad. Expone el precio de venta y la cantidad disponible de cada producto, con una notificación que indica si el producto está agotado.

- Descripción. Proporciona información adicional sobre cada producto.

Pie de Página

- Incluye enlaces a redes sociales, lo que permite a los usuarios conectarse con la empresa a través de diferentes plataformas.
- Se proporciona información de derechos de autor, destacando a los creadores del sitio.

Interactividad y Scripts

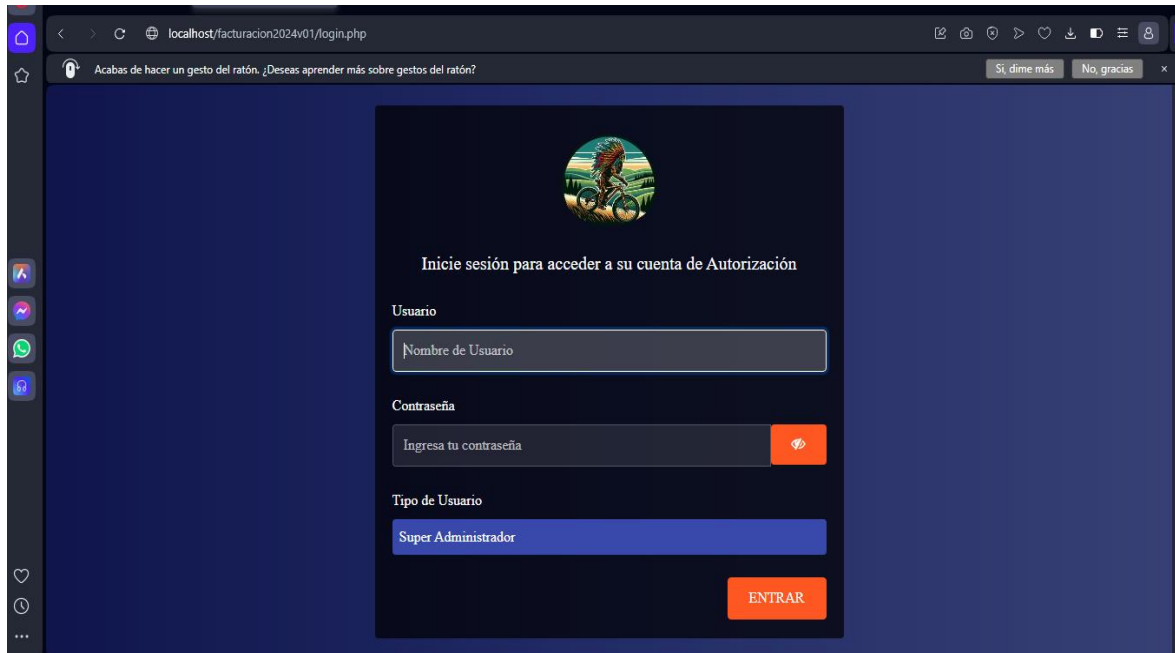
- La página incluye scripts de jQuery y Bootstrap para mejorar la interactividad y la funcionalidad, como animaciones y controles de navegación.

Conclusión. La página `index.php` de Chaman Bikes es un punto de entrada bien estructurado y funcional para los usuarios del sistema de gestión de bicicletas. Ofrece una navegación intuitiva, un diseño responsivo y un acceso fácil a información clave sobre productos y servicios, todo lo cual contribuye a una experiencia de usuario positiva y eficiente.

Login.php

Figura 24

Login



La Importancia del Login en Sistemas de Inventario y Gestión

El módulo de inicio de sesión o login es una de las partes más fundamentales en cualquier sistema de información, como el que hemos desarrollado para Chaman Bikes. A continuación, detallo por qué un sistema de login es esencial para proyectos de esta naturaleza.

Seguridad de la Información

El login es la primera defensa para salvaguardar la información de la aplicación. En aplicaciones que manejan información sensible como inventarios, historial de mantenimientos y datos de clientes, es crítico garantizar que únicamente los usuarios con autorización tengan acceso a estos datos. Sin un sistema de autenticación, la información estaría expuesta a cualquier persona, lo que pondría en riesgo la seguridad de la empresa.

Control de Acceso por Roles

Un sistema de login permite diferenciar entre los distintos tipos de usuarios.

- **Superadministradores.** Con acceso total a todas las funcionalidades, como la gestión del inventario, mantenimientos y usuarios.
- **Administradores.** Que pueden realizar tareas específicas, pero con ciertas limitaciones.
- **Invitados.** Con acceso más restringido, ideal para visualizar ciertos datos, pero sin permisos para modificarlos.

Esto permite gestionar de manera eficiente las responsabilidades dentro del sistema, asegurando que cada usuario solo pueda hacer lo que le corresponde, minimizando riesgos de errores o accesos indebidos.

Trazabilidad y Auditoría

Cada vez que un usuario inicia sesión, el sistema puede registrar esta actividad, lo que permite llevar un control de quién accede a qué recursos y cuándo. Esto es clave para auditorías o cuando se necesita investigar cambios realizados en el sistema, como modificaciones en el inventario o en las órdenes de mantenimiento.

Experiencia de Usuario Personalizada

Un sistema con inicio de sesión permite ofrecer una experiencia adaptada a cada tipo de usuario. Los administradores pueden ver paneles de control específicos con métricas de rendimiento, mientras que un técnico de mantenimiento puede ver un resumen de las bicicletas que necesita reparar. Esta personalización hace el sistema más eficiente y fácil de usar.

Cumplimiento Normativo

En algunos sectores, es obligatorio tener controles de acceso que limiten quién puede ver y modificar datos específicos. Un sistema de login asegura que cumplamos con leyes de protección de datos personales, cumpliendo la normativa (en el caso de ciertos países), lo cual es crucial si el sistema llega a manejar información sensible de clientes o empleados.

Prevención de Ataques

Implementar un sistema de login con buenas prácticas (como políticas de contraseñas seguras y mecanismos de bloqueo tras varios intentos fallidos) ayuda a prevenir ataques maliciosos como el phishing o la fuerza bruta, que buscan vulnerar las cuentas de usuarios para acceder al sistema.

Conclusión. Una estructura de login no es solo una método de seguridad para permitir el ingreso de los usuarios al sistema; es un componente primordial que garantiza la protección, el control y el rendimiento en la gestión de los medios del taller. Implementar un inicio de sesión sólido es una práctica primordial en cualquier programa que maneje datos sensibles o complejos, como lo es el sistema de inventario y mantenimientos de Chaman Bikes.

Controlpanel_admin

Figura 25

Panel de Control

| Id.Producto | Nombre del Producto | Cantidad | Precio | Imagen |
|-------------|---------------------|----------|--------|--------|
| 95 | herramientas | 30 | 40000 | |
| 94 | reloj | 34 | 12000 | |
| 93 | tensor | 40 | 15000 | |
| 92 | sillin | 47 | 18000 | |
| 91 | rin | 100 | 14000 | |

El panel de control en la página de "ChamanBikes", como proyecto de grado, es la interacción de administración que permite gestionar el inventario, las compras, las facturas y el estado de los productos en stock. Este panel está construido en PHP, utilizando Bootstrap para el diseño visual y MySQL como base de datos para guardar los datos del inventario y otros datos de la tienda.

Conexión a la Base de Datos

Se establecen las conexiones a la base de datos y se incluye la información de los usuarios para la administración.

Figura 26

Código de la Conexión a la Base De Datos

```
<?php require_once('Connections/cnx.php'); ?>
<?php require_once('Connections/infousers.php'); ?>
```

Consulta de Productos Recientes

- Se realiza una consulta SQL que selecciona los últimos 10 productos agregados al inventario, ordenados por su ID de manera descendente.

Figura 27

Ejemplo Consulta en Base de Datos

```
$query_RstProductos = "SELECT * FROM productos ORDER BY idproducto DESC LIMIT 10";
```

Función Stock Bajo (\$idp)

- Esta función determina el nivel de stock de un producto y asigna una clase CSS basada en la cantidad de unidades disponibles.

- btn-danger si el stock es menor o igual a 3 (indica peligro o falta de stock).
- btn-warning si el stock está entre 4 y 5 (advertencia).
- btn-success si el stock es mayor a 5 (suficiente stock).

Figura 28

Código Verificación del Stock

```
function stockbajo($idp) {
    // Lógica para verificar el nivel de stock
}
```

Figura 28. Código verificación del Stock.

Interacción del Administrador

- El diseño de la interacción está basado en Bootstrap, con tarjetas de resumen que muestran el número de productos, compras, facturas y productos con stock bajo.
- Las tarjetas utilizan iconos de FontAwesome y colores para diferenciar la información, haciendo el panel visualmente claro y atractivo.

Tablas de Datos

- Los últimos productos agregados se muestran en una tabla. Cada fila contiene.
- El ID del producto.
- El nombre del producto.
- La cantidad disponible, resaltada con colores según el stock.
- El precio del producto.
- Un enlace a la imagen del producto.

Figura 29

Tabla para Almacenar Datos

```
<table id="tabla" class="table table-striped ...">
  <!-- Código de la tabla -->
</table>
```

Scripts y Bibliotecas Externas

El panel incluye varias bibliotecas externas, como Chart.js, DataTables, y Highcharts para mostrar gráficos interactivos y tablas dinámicas que facilitan la visualización y administración de la información.

Uso de graficas

- Se integran gráficos interactivos para mostrar datos estadísticos, como ventas mensuales o productos más vendidos, utilizando las bibliotecas **Highcharts** y **Morris.js**.

Este panel proporciona una solución eficaz para gestionar y supervisar el taller de mantenimiento y venta de bicicletas, permitiendo a los administradores mantener el control sobre los productos, el stock y las transacciones.

Panel Productos

Productos_control.php

Figura 30

Panel de los Productos

The screenshot displays the 'Lista de Productos' (Product List) interface. The main content area shows a table with the following data:

| idproducto | Nombre del Producto | Cantidad | Precio de Compra | Precio de Venta | Imagen | Comprar | Modificar | Eliminar |
|------------|---------------------|----------|------------------|-----------------|--------|---------|-----------|----------|
| 92 | sillin | 47 | \$18.000 | \$30.000 | | | | |
| 93 | tensor | 40 | \$15.000 | \$30.000 | | | | |
| 87 | zapatilla deportiva | 20 | \$100.000 | \$220.000 | | | | |

The interface also features a sidebar with navigation options, a search bar, and pagination controls showing 'Mostrando registros del 21 al 23 de un total de 23 registros'.

Conexiones y Consultas a la Bases de Datos

Se incluyen archivos externos para establecer el enlace con la base de datos (cnx.php, infousers.php) y el manejo de contadores (contadores.php).

Se selecciona la base de datos y se realiza una consulta para obtener todos los productos ordenados por nombre, que son almacenados en un array asociativo.

El número total de productos obtenidos se guarda en \$totalRows_RstProductos.

Eliminación de Productos

Si se recibe un parámetro idproducto a través de GET, se ejecuta una consulta DELETE para eliminar el producto correspondiente. Después de eliminarlo, la página redirige automáticamente a productos_control.php para actualizar la vista.

Funciones Adicionales

Eliminar. Verifica si un producto tiene dependencias en la tabla de compras y deshabilita el botón de eliminación si es el caso.

Ver entradas y ver salidas. Estas funciones verifican si hay compras o ventas relacionadas con un producto, habilitando o deshabilitando los botones correspondientes.

Ver stock. Comprueba si el producto tiene stock disponible y ajusta el estilo del botón según la disponibilidad pintar fila.

Cambia el color de la fila del producto si la cantidad es igual a cero o stock bajo. Clasifica los productos en tres categorías de acuerdo con su nivel de stock (bajo, medio, alto) y asigna un color a los botones en consecuencia.

HTML y Bootstrap para el Frontend

La estructura HTML utiliza Bootstrap para crear un diseño responsivo y amigable con el usuario.

Incluye un menú de navegación superior y una barra lateral izquierda para facilitar la navegación.

En la tabla de productos, se utilizan varios iconos y estilos CSS personalizados. Por ejemplo, la imagen del producto se muestra en forma de un círculo con bordes estilizados.

Si el usuario es administrador, aparecen botones adicionales para modificar, eliminar y gestionar productos.

DataTables

La tabla de productos utiliza la librería DataTables para agregar funcionalidad como la búsqueda, paginación y filtrado dinámico.

El script de configuración de DataTables se personaliza para adaptarse al idioma español y definir el número de registros mostrados por página.

CSS y Scripts

Se incluyen hojas de estilo para darle forma al botón de imagen del producto y al modal de Bootstrap.

Se incluyen scripts externos como bootbox.js para manejar la confirmación de eliminación de productos, así como el uso de DataTables para la tabla.

Descripción del Funcionamiento

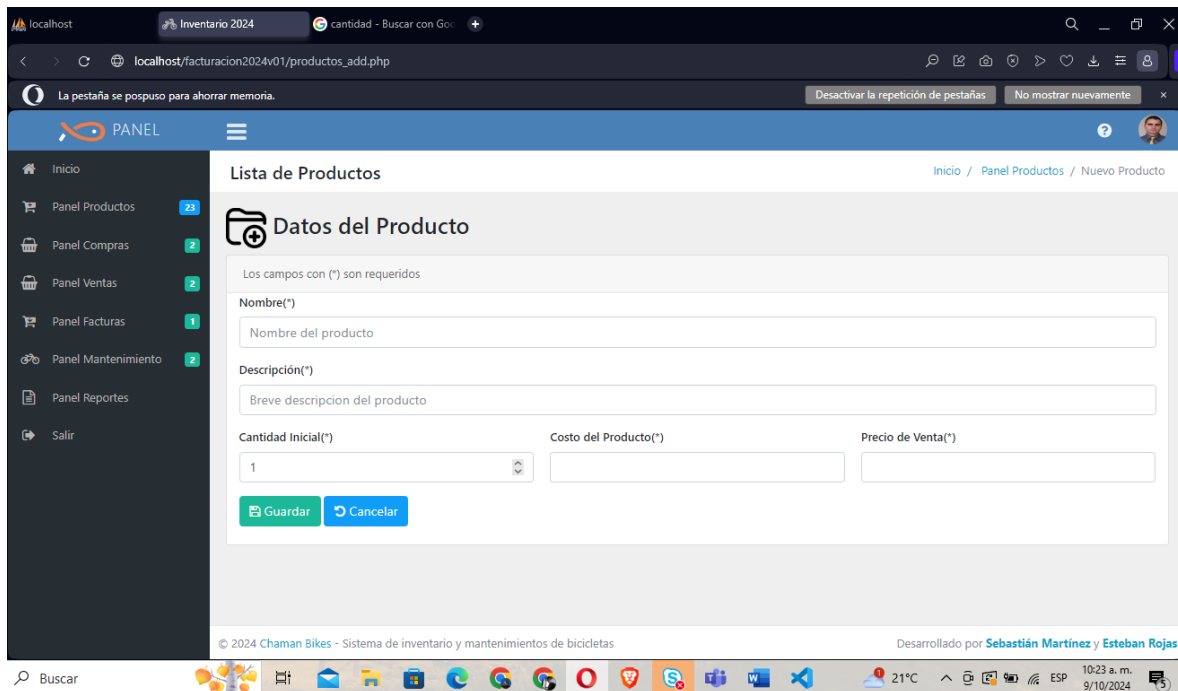
Esta sección de gestión está diseñada para facilitar la administración de los productos de un inventario. Los administradores pueden agregar nuevos productos, modificar los existentes o eliminarlos si no tienen dependencias (compras o ventas relacionadas). Además, se puede verificar el stock y categorizar los productos con poco inventario, mientras que los usuarios regulares pueden consultar la lista sin capacidad de modificarla.

Este tipo de página web es común en sistemas de gestión de inventarios para negocios o tiendas, proporcionando una interacción eficiente y organizada que facilita las operaciones diarias.

Agregar productos

Figura 31

Agregar Datos



The screenshot shows a web browser window displaying a web application interface. The browser's address bar shows the URL `localhost/facturacion2024v01/productos_add.php`. The application has a dark sidebar on the left with a menu containing items like 'Inicio', 'Panel Productos', 'Panel Compras', 'Panel Ventas', 'Panel Facturas', 'Panel Mantenimiento', 'Panel Reportes', and 'Salir'. The main content area is titled 'Lista de Productos' and 'Datos del Producto'. It contains a form with the following fields: 'Nombre(*)', 'Descripción(*)', 'Cantidad Inicial(*)' (with a value of 1), 'Costo del Producto(*)', and 'Precio de Venta(*)'. There are 'Guardar' and 'Cancelar' buttons at the bottom of the form. The footer of the application indicates '© 2024 Chaman Bikes - Sistema de inventario y mantenimientos de bicicletas' and 'Desarrollado por Sebastián Martínez y Esteban Rojas'. The Windows taskbar at the bottom shows the system tray with a search bar, taskbar icons, and system information like '21°C' and '10:23 a.m. 9/10/2024'.

El módulo "Agregar Producto" es una funcionalidad clave dentro del sistema de inventario de Chaman Bikes. Su propósito es permitir a los administradores o empleados del negocio agregar nuevos artículos añadidos a la base de datos de manera sencilla y estructurada.

Formulario de Registro de Productos

El formulario está dividido en varias secciones, cada una solicitando información crítica para el manejo de productos.

Campos obligatorios, como el Nombre del Producto, Descripción, Cantidad Inicial, Costo del Producto, y Precio de Venta, aseguran que todos los datos necesarios para el control del inventario sean recolectados de manera eficiente.

Se utilizan validaciones en tiempo real mediante la librería Parsley.js, la cual ayuda a evitar errores al ingresar datos incorrectos o incompletos.

Interacción de Usuario

El diseño es limpio y sencillo, utilizando Bootstrap para crear una interacción responsiva, lo que facilita su uso tanto en móviles como en equipos de escritorio.

El ícono de un carrito de compra junto al título "Datos del Producto" mejora la experiencia visual y hace que el propósito del formulario sea claro para el usuario.

Botones de Acción

Se incluyen dos botones importantes.

Guardar. Al presionar este botón, el sistema valida la información y envía los datos al archivo `productos_save.php` para que sean procesados y guardados en la base de datos.

Cancelar. Permite al usuario volver a la pantalla anterior sin realizar ninguna acción.

Validación de Datos

Se asegura que todos los campos obligatorios sean completados antes de enviar el formulario. Esto ayuda a reducir errores y mejora la integridad de los datos ingresados.

Importancia del Módulo

Este módulo es esencial para mantener un inventario actualizado y gestionado eficientemente en el sistema de Chaman Bikes. Permite.

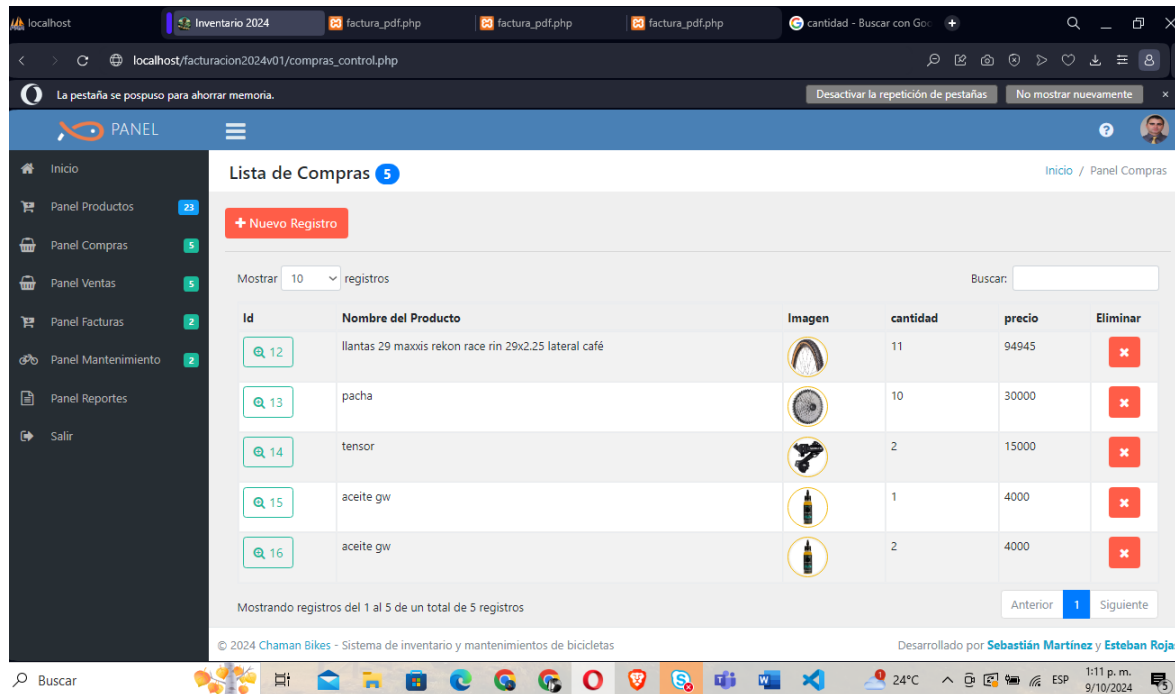
- Controlar las cantidades de productos disponibles.
- Definir precios de venta y costo, lo cual es fundamental para el análisis financiero.
- Mejorar la precisión y la rapidez en la administración de nuevos productos, agilizando las operaciones del negocio.

Esta funcionalidad es parte vital del sistema de inventario, ya que, sin un registro adecuado de los productos, sería imposible gestionar correctamente las existencias y los precios.

Panel compras compras_control.php

Figura 32

Panel de Compras



Informe. Página "Control Compras" del Sistema Chaman Bikes

Descripción General

La página "Control Compras" forma parte del proyecto Chaman Bikes, que gestiona el inventario, mantenimientos y facturación de un taller de bicicletas. Este módulo específico permite administrar las compras de productos registrados en el sistema, facilitando el control del stock disponible en el taller.

Objetivos del Módulo

El módulo tiene como principal objetivo proporcionar una herramienta para gestionar la compra de nuevos inventarios de productos que ya se encuentran registrados en la base de datos de del programa. Con ello se busca.

- Monitorear el stock de productos a través de compras registradas.

- Visualizar y administrar detalles de compras como. nombre del producto, imagen, cantidad adquirida y precio de compra.
- Facilitar la eliminación de compras en caso de errores o productos no disponibles.

Características Principales

Consulta de Compras. El sistema recupera los datos de las compras realizadas, integrando información de la tabla compras y la tabla productos, como el nombre del producto, imagen, fecha de compra, cantidad y precio.

Eliminación de Registros. Permite eliminar compras específicas a través de un enlace con el parámetro idproducto, lo que elimina el registro de la tabla de compras.

Interacción Amigable. Emplea el marco de trabajo Bootstrap para un diseño responsivo y contemporáneo, que se ajusta a diferentes aparatos.

DataTables. Mejora la presentación de la tabla de compras, permitiendo la paginación, búsqueda y ordenación de los registros de manera dinámica.

Seguridad y Control de Acceso

Sólo los administradores tienen acceso a la eliminación de registros o la posibilidad de registrar nuevos productos. Los usuarios sin permisos administrativos no tienen acceso a estas funciones.

Flujo de Trabajo

- Al ingresar a la página, el sistema efectúa un requerimiento a la base de datos para obtener las compras registradas.
- Se presenta una tabla que muestra todos los productos comprados, incluyendo una imagen de referencia, cantidad y precio.

- Si el usuario tiene permisos de administrador, puede eliminar una compra específica o añadir un nuevo registro desde un botón directo en la interacción.

Beneficios

- Control eficiente del stock. Permite saber en todo momento qué productos se han comprado y su disponibilidad.

- Fácil gestión de inventarios. El personal administrativo puede fácilmente eliminar registros erróneos o añadir nuevas compras para mantener el inventario actualizado.

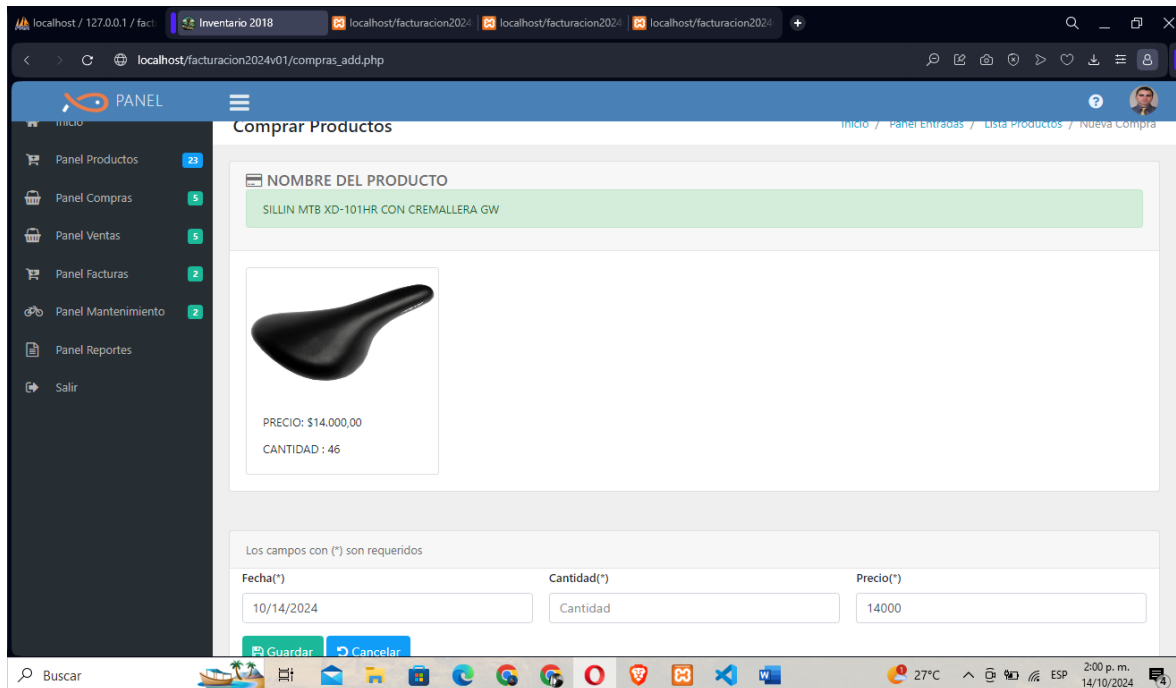
- Interacción intuitiva. Al estar diseñada con Bootstrap y DataTables, ofrece una temática de usuario rápida y fácil en su uso.

Este módulo es parte fundamental para el control de inventarios de Chaman Bikes, asegurando un seguimiento claro y preciso de las compras que se realizan para mantener el stock de productos del taller.

Compras_add

Figura 33

Agregar Compras



La página "Agregar Compras" del proyecto Chaman Bikes, Desarrollado en PHP

MySQL y Bootstrap, permite al administrador añadir nuevas compras de productos al sistema de inventario, actualizando su stock de manera eficiente. A continuación, se describe su funcionamiento.

Conexión a la Base de Datos

Se conecta a la base de datos mediante los archivos `cnx.php` e `infousers.php`.

La página recibe el `idproducto` a través de la URL (`$_GET['idproducto']`), y se realiza una consulta para recibir los datos completa del producto desde la tabla `productos`, como nombre, imagen, precio y cantidad disponible.

Interacción de Usuario

La página está diseñada con Bootstrap, asegurando una estructura responsiva y moderna.

Muestra detalles del producto seleccionado como el nombre, imagen, precio y cantidad actual. Estos datos se muestran en un card con un diseño visual claro.

Un breadcrumb permite una navegación fácil entre diferentes secciones, como el panel de control o la lista de productos.

Formulario para Agregar Stock

El administrador puede registrar una nueva compra ingresando la fecha de la compra, la cantidad adquirida y el precio del producto.

El campo de fecha utiliza un calendario desplegable implementado con la librería daterangepicker para seleccionar la fecha de la compra.

Los datos del formulario se envían mediante POST al archivo `entradas_save.php` para su procesamiento.

Validación y Envío de Datos

El formulario requiere la validación de campos obligatorios como la fecha, la cantidad y el precio.

Al enviar el formulario, se almacenan datos adicionales ocultos, como el `idproducto`, para asegurar que los datos se guarden correctamente en relación con el producto seleccionado.

Scripts y Estilo

Se incluyen varios archivos de JavaScript y CSS para mejorar la experiencia de usuario, como la librería jQuery y moment.js para manejar fechas.

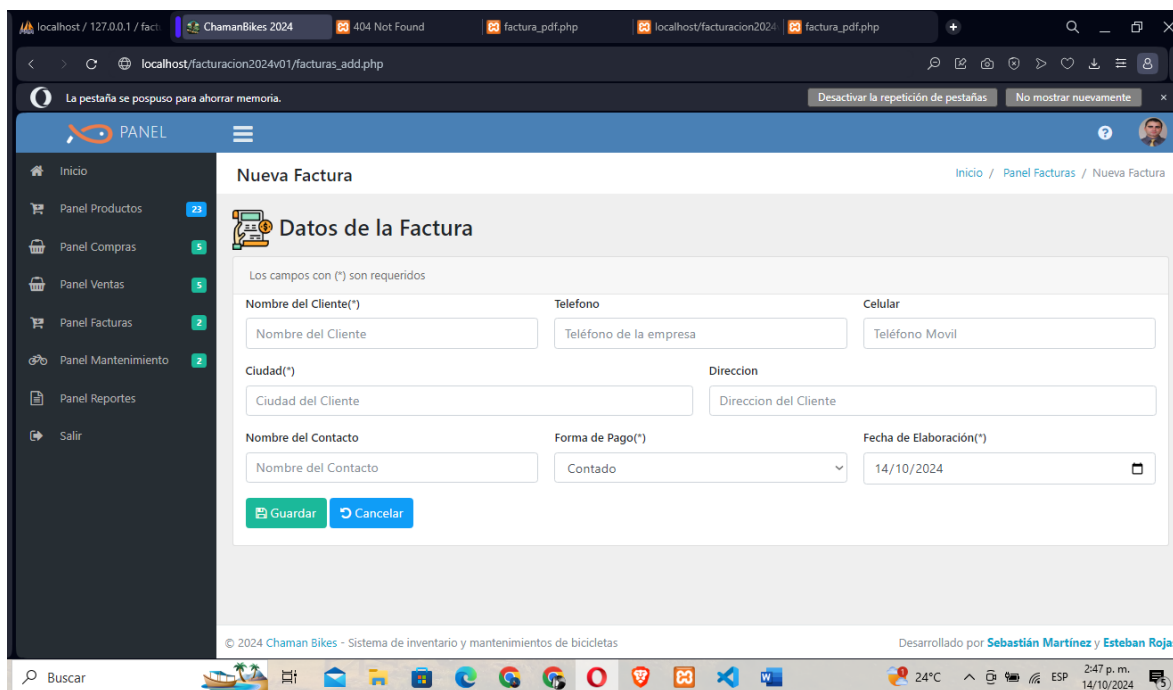
Los archivos CSS de Bootstrap y Font Awesome son utilizados para el diseño visual.

En resumen, esta página facilita el ingreso de nuevas compras y la actualización del stock de productos, proporcionando una interacción intuitiva y bien estructurada para el administrador del sistema.

Facturas add

Figura 34

Agregar Factura



The screenshot shows a web browser window displaying the 'Nueva Factura' (New Invoice) form. The browser's address bar shows the URL 'localhost/facturacion2024v01/facturas_add.php'. The page has a dark blue sidebar with a 'PANEL' header and a navigation menu including 'Inicio', 'Panel Productos', 'Panel Compras', 'Panel Ventas', 'Panel Facturas', 'Panel Mantenimiento', 'Panel Reportes', and 'Salir'. The main content area is titled 'Nueva Factura' and contains a form titled 'Datos de la Factura'. The form includes several input fields: 'Nombre del Cliente(*)', 'Telefono', 'Celular', 'Ciudad(*)', 'Direccion', 'Nombre del Contacto', 'Forma de Pago(*)', and 'Fecha de Elaboración(*)'. There are also 'Guardar' and 'Cancelar' buttons at the bottom of the form. The footer of the page indicates '© 2024 Chaman Bikes - Sistema de inventario y mantenimientos de bicicletas' and 'Desarrollado por Sebastián Martínez y Esteban Rojas'. The browser's taskbar at the bottom shows the system date and time as 2:47 p.m. on 14/10/2024.

La página "Agregar Factura" en el proyecto Chaman Bikes, desarrollado en PHP, MySQL y Bootstrap, permite al administrador registrar nuevas facturas para los clientes del sistema. Su propósito principal es recolectar información relevante sobre las ventas realizadas y almacenarlas en la base de datos para su posterior gestión. Se detalla su funcionamiento.

Conexión a la Base de Datos

El código se conecta a la base de datos utilizando el archivo `cnx.php` para gestionar las operaciones de facturación.

También incluye `infousers.php` para manejar los datos de los usuarios y `contadores.php` para manejar los contadores de las facturas.

Interacción de Usuario

La página está diseñada utilizando Bootstrap, lo que asegura una estructura visual agradable y adaptable.

Se presenta un encabezado que indica que el administrador está en la sección de "Nueva Factura", y se incluye una imagen representativa relacionada con facturas.

Los breadcrumbs facilitan la navegación, permitiendo regresar fácilmente a otras áreas del sistema, como el panel de control o la lista de facturas.

Formulario para la Creación de Facturas

El formulario recolecta información esencial del cliente, como nombre, teléfono, ciudad y dirección, así como la fecha de elaboración y la forma de pago (contado, crédito, cheque).

Los campos obligatorios están marcados con asteriscos (*) y validados mediante la librería Parsley.js, lo que asegura que los datos requeridos sean ingresados correctamente antes de enviar el formulario.

La fecha de elaboración se inicializa automáticamente con la fecha actual (`$Fecha=date('Y-m-d');`), mejorando la precisión de los registros.

Validación y Procesamiento de Datos

Al enviar el formulario, los datos se envían a la página facturas_save.php para su procesamiento y almacenamiento en la base de datos.

El formulario cuenta con validación incorporada mediante Parsley.js, que asegura que los campos obligatorios se completen antes de que los datos se envíen.

Scripts y estilo.

El formulario diseñado para ser sencillo e intuitivo de manejar.

Se incluyen varios scripts para mejorar la experiencia del usuario, incluyendo validación avanzada y el uso de SweetAlert para mostrar alertas visuales.

En resumen, esta página facilita el registro de nuevas facturas en el sistema, asegurando que toda la información relevante del cliente y de la venta se capture de manera eficiente y precisa. La validación en el frontend asegura que los datos ingresados sean correctos antes de ser procesados y guardados en la base de datos.

Panel Mantenimientos

Figura 35

Lista de Mantenimientos

La pestaña se pospuso para ahorrar memoria. Desactivar la repetición de pestañas No mostrar nuevamente

PANEL

Inicio

Panel Productos 23

Panel Compras 5

Panel Ventas 5

Panel Facturas 2

Panel Mantenimiento 6

Panel Reportes

Salir

Lista de Mantenimientos 6 Inicio / panel Productos

Datos del Mantenimiento

+ Nuevo Registro

Mostrar 10 registros Buscar:

| ID | Cedula | Nombre | Apellido | Celular | Serial | Fecha Entrada | Fecha Salida | Tipo Mantenimiento | Foto | Modificar | Eliminar |
|----|------------|-----------|----------|------------|--------|---------------------|---------------------|-------------------------|------|-----------|----------|
| 2 | 1035919796 | Sebastian | Martinez | 2147483647 | AAABB | 2024-10-14 16:01:00 | 2024-10-14 00:00:00 | General | | | |
| 3 | 70750760 | MARIO | MENDIETA | 2147483647 | 12345 | 2023-10-13 00:00:00 | 2023-10-23 00:00:00 | general | | | |
| 4 | 70750761 | MARIO | MENDIETA | 2147483647 | 12345 | 2023-10-13 00:00:00 | 2023-10-23 00:00:00 | general | | | |
| 8 | 1035919795 | Sebastian | Martinez | 2147483647 | AAAxx | 2024-10-14 14:00:00 | 2024-10-14 12:00:00 | General | | | |
| 9 | 1035919795 | Sebastian | Martinez | 2147483647 | AAASS | 2024-10-14 17:00:00 | En Mantenimiento | General - grasa premium | | | |

Panel reportes

Figura 36

Panel de Reportes

La pestaña se pospuso para ahorrar memoria. Desactivar la repetición de pestañas No mostrar nuevamente

PANEL

Inicio

Panel Productos 23

Panel Compras 5

Panel Ventas 5

Panel Facturas 2

Panel Mantenimiento 6

Panel Reportes

Salir

Lista de Reportes Inicio / Panel Reportes

REPORTES

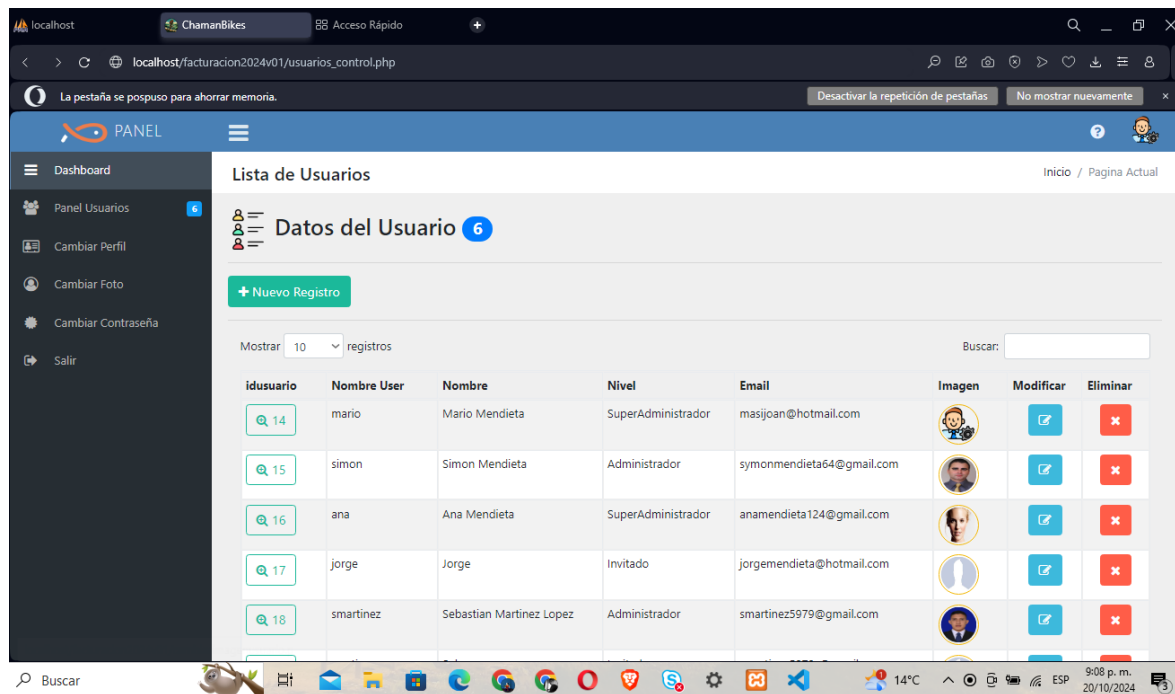
| Reporte | Descarga |
|------------------------|----------|
| Lista De Productos | |
| Lista De Entradas | |
| Lista De Salidas | |
| Lista De Facturas | |
| Lista Detalle Facturas | |
| Lista De Mantenimiento | |

© 2024 Chaman Bikes - Sistema de inventario y mantenimientos de bicicletas Desarrollado por Sebastián Martínez y Esteban Rojas

Superadministrador Usuarios_Control.Php

Figura 37

Panel Tipo de Usuarios



La página "Usuarios Control" del proyecto Chaman Bikes está diseñada para la administración de usuarios del sistema, enfocada en el superadministrador. Su funcionalidad principal es permitir la creación, modificación, eliminación y búsqueda de usuarios en la base de datos, todo gestionado a través de una interacción desarrollada en PHP, Bootstrap y MySQL.

La página muestra una lista de usuarios con información relevante como nombre, nivel, correo electrónico e imagen de perfil, con opciones para editar o eliminar cada registro. Además, incluye un botón para agregar nuevos usuarios. Para eliminar un usuario, se realiza una confirmación previa, evitando acciones accidentales. La imagen de perfil de cada usuario puede ser visualizada o cambiada mediante un enlace directo. El sistema está equipado con un buscador y utiliza tablas dinámicas con paginación y filtros, mejorando la gestión de grandes volúmenes de datos.

Conexión A La Base De Datos

Figura 38

Conexión a la Base de Datos

```

1 <?php require_once('Connections/cnx.php'); ?>
2 <?php
3 // *** Validar la solicitud de inicio de sesión en este sitio.
4 //if (!isset($_SESSION))
5 //{
6 session_start();
7 // Establece la sesión como "autenticado" si se realiza el login correctamente
8 $_SESSION["autenticado"]="SI"; |
9 if (isset($_POST['Usuario'])) // Verifica si el formulario ha enviado el campo 'Usuario'
10 {
11     $loginUsername=$_POST['Usuario']; // Captura el nombre de usuario ingresado
12     // $password=sha1(md5($_POST['Clave'])); // Este código está comentado y no en uso
13     $password=$_POST['Password']; // Captura la contraseña ingresada
14     $TipoUser=$_POST['TipoUser']; // Captura el tipo de usuario
15     // echo $loginUsername; // Para depurar, imprime el nombre de usuario (comentado)
16     // echo $password; // Para depurar, imprime la contraseña (comentado)
17     // $MM_redirectLoginSuccess = "controlpanel_admin.php"; // Redirección en caso de éxito
18     // $MM_redirectLoginFailed = "noexiste.php"; // Redirección en caso de error en el login
19     // $MM_redirecttoReferrer = false; // Opción comentada
20
21     // Selecciona la base de datos
22     mysqli_select_db($cnx,$database_cnx);
23
24     // Consulta SQL para verificar las credenciales del usuario
25     // $loginRS_query="SELECT UserName, Pass FROM usuarios WHERE UserName='$loginUsername' AND Pass='$pas
26     // $loginRS_query="SELECT UserName, Pass FROM usuarios WHERE UserName='$loginUsername' AND Pass='$pas
27
28     // Consulta para verificar que coincidan el nombre de usuario y la contraseña
29     $loginRS_query="SELECT UserName, Pass FROM usuarios
30     WHERE (UserName='$loginUsername' AND Pass='$password')";

```

Este archivo es crucial, ya que define los parámetros necesarios para establecer una conexión con la base de datos en el servidor. Los valores establecidos en este archivo incluyen.

- **Hostname.** Indica el servidor de la base de datos, en esta situación, 'localhost', lo que indica que los datos se encuentran en el mismo servidor donde se encuentra la aplicación web.
- **Username.** Establece el identificador de usuario que puede acceder a la información. Aquí se utiliza 'root', el usuario predeterminado en entornos de desarrollo locales.
- **Password.** Es la contraseña correspondiente al usuario, en este caso está vacía, algo común en servidores de desarrollo locales, pero que debe ser asegurado en entornos de producción.

- **Database.** Se refiere al nombre de la base de datos específica a la que se relacionara el programa, que en este caso es 'facturas2023v01'.

El código también incluye una función para establecer el conjunto de caracteres de la conexión a 'utf8', lo que garantiza que los datos se manejen correctamente en términos de codificación, permitiendo el uso de caracteres especiales o acentuados.

La importancia de este archivo radica en que garantiza que la aplicación pueda interactuar con la base de datos de manera eficiente y segura. Sin una conexión establecida correctamente, la aplicación no podría acceder ni gestionar los datos necesarios para su funcionamiento.

En este fragmento de código, se realiza la conexión a la base de datos utilizando la configuración establecida en el archivo Connections/cnx.php. Esta conexión es esencial para poder interactuar con la base de datos y ejecutar consultas, ya que permite verificar si las credenciales del usuario ingresadas en el formulario coinciden con las almacenadas en la tabla de usuarios.

La validación se lleva a cabo mediante una consulta SQL que compara el nombre de usuario (UserName) y la clave (Pass) proporcionados por el usuario con los registros de la información. Si se encuentra una coincidencia, el sistema autentica al usuario y le concede acceso, lo que habilita el flujo de trabajo en el sistema.

El uso de conexiones seguras y correctas es fundamental para asegurar que la aplicación pueda gestionar los datos correctamente, manteniendo la integridad y disponibilidad de la información guardada en la base de datos.

Css Bootstrap

Figura 39

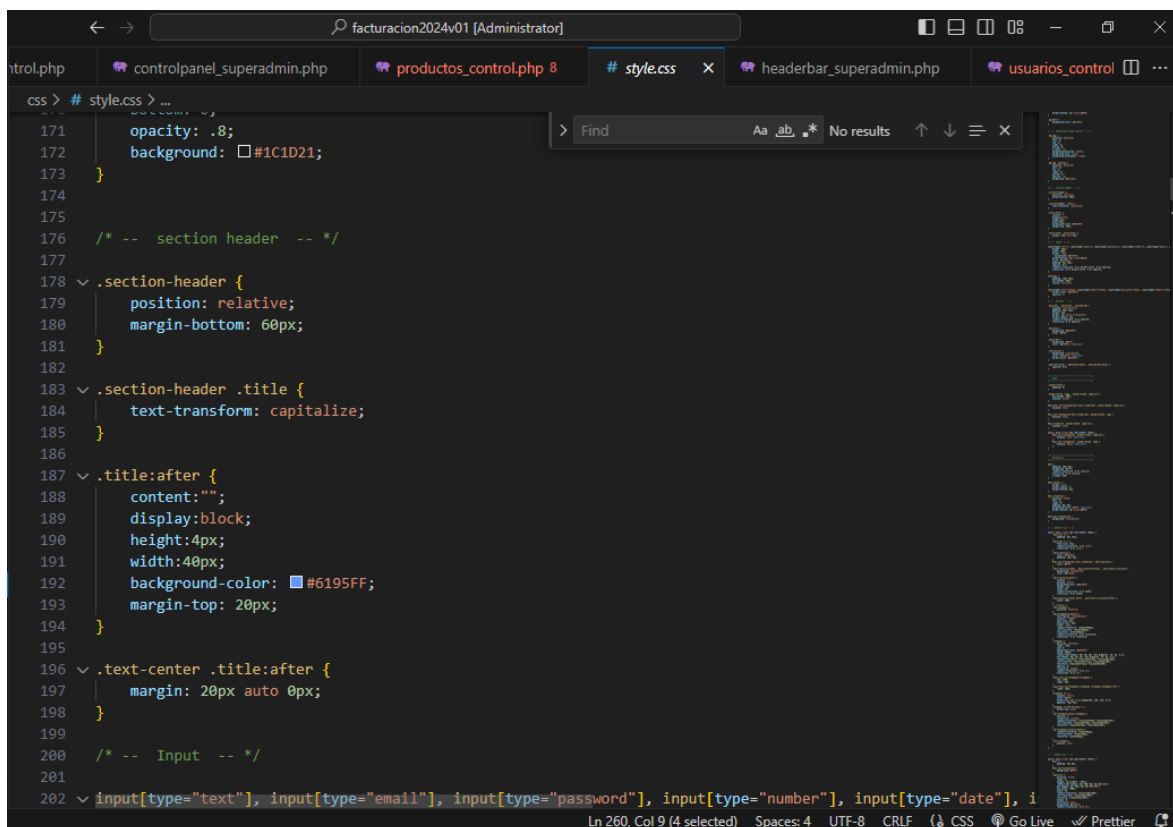
Aplicación de Bootstrap

```
<!-- Bootstrap CSS -->
<link href="assets/css/bootstrap.min.css" rel="stylesheet" type="text/css" />
<!-- Font Awesome CSS -->
<link href="assets/font-awesome/css/font-awesome.min.css" rel="stylesheet" type="text/css" />
<!-- Custom CSS -->
<link href="assets/css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="https://cdn.datatables.net/1.10.16/css/dataTables.bootstrap4
```

Style.css

Figura 40

Aplicación de Estilos con CSS



```
css > # style.css > ...
171 opacity: .8;
172 background: #1C1D21;
173 }
174
175
176 /* -- section header -- */
177
178 .section-header {
179   position: relative;
180   margin-bottom: 60px;
181 }
182
183 .section-header .title {
184   text-transform: capitalize;
185 }
186
187 .title:after {
188   content: "";
189   display: block;
190   height: 4px;
191   width: 40px;
192   background-color: #6195FF;
193   margin-top: 20px;
194 }
195
196 .text-center .title:after {
197   margin: 20px auto 0px;
198 }
199
200 /* -- Input -- */
201
202 input[type="text"], input[type="email"], input[type="password"], input[type="number"], input[type="date"], i
```

Funciones Personalizadas que Consultan la Base de Datos

En estas funciones se realizan consultas adicionales, como verificar las entradas y salidas de productos.

Figura 41

Verificar las Entradas

```
// Función para verificar entradas (compras) de un producto
function verentradas($id) {
    require('Connections/cnx.php');
    mysqli_select_db($cnx, $database_cnx);
```

Figura 42

Verificar la Cantidad de Productos

```
// Consulta para obtener la información del producto
$sql1 = "SELECT * FROM productos WHERE idproducto = $idp";
$RstResp1 = mysqli_query($cnx, $sql1) or die(mysql_error());
$row_RstResp1 = mysqli_fetch_assoc($RstResp1);

// Verifica la cantidad del producto
$Cantidad = $row_RstResp1['cantidad'];
$Color = '';

// Si la cantidad es 0, pinta la fila de rojo
if ($Cantidad == 0) {
    $Color = '#e51919';
}

return $Color;
```

Script de la Base de Datos

Figura 43

Script 1

```
-- Clave primaria: `id` (única para cada detalle de factura)
-- Relación con las tablas `facturas` y `productos`:
-- El campo `idfactura` es una clave foránea hacia `facturas(idfactura)`
-- El campo `idproducto` es una clave foránea hacia `productos(idproducto)`

CREATE TABLE detallefactura (
  id int NOT NULL AUTO_INCREMENT, -- Clave primaria
  idfactura int NOT NULL, -- Clave foránea hacia `facturas(idfactura)`
  idproducto int NOT NULL, -- Clave foránea hacia `productos(idproducto)`
  cantidad int NULL DEFAULT NULL,
  precio int NULL DEFAULT NULL,
  PRIMARY KEY (id) USING BTREE, -- Clave primaria
  CONSTRAINT fk_factura FOREIGN KEY (idfactura) REFERENCES facturas (idfactura), -- Relación con
  `facturas`
  CONSTRAINT fk_producto FOREIGN KEY (idproducto) REFERENCES productos (idproducto) --
  Relación con `productos`
) ENGINE = InnoDB;
```

Figura 43*Script 2*

```
-- Clave primaria: `idproducto` (única para cada producto)

CREATE TABLE productos (
  idproducto int NOT NULL AUTO_INCREMENT, -- Clave primaria
  nombre varchar(100) CHARACTER SET utf8 COLLATE utf8_bin NOT NULL,
  descripcion varchar(255) CHARACTER SET utf8 COLLATE utf8_bin NOT NULL,
  cantidad int NULL DEFAULT NULL,
  precio int NULL DEFAULT NULL,
  precioventa int NULL DEFAULT NULL,
  imagen varchar(100) CHARACTER SET utf8 COLLATE utf8_bin NULL DEFAULT NULL,
  PRIMARY KEY (idproducto) USING BTREE -- Clave primaria
) ENGINE = InnoDB;
```

Figura 44*Script 3*

```
-- Clave primaria: `idfactura` (única para cada factura)

CREATE TABLE facturas (
  idfactura int NOT NULL, -- Clave primaria
  razonsocial varchar(150) CHARACTER SET utf8 COLLATE utf8_bin NULL DEFAULT NULL,
  telefono varchar(20) CHARACTER SET utf8 COLLATE utf8_bin NULL DEFAULT NULL,
  celular varchar(30) CHARACTER SET utf8 COLLATE utf8_bin NULL DEFAULT NULL,
  ciudad varchar(100) CHARACTER SET utf8 COLLATE utf8_bin NULL DEFAULT NULL,
  direccion varchar(100) CHARACTER SET utf8 COLLATE utf8_bin NULL DEFAULT NULL,
  fechaelaboracion date NULL DEFAULT NULL,
  formapago varchar(50) CHARACTER SET utf8 COLLATE utf8_bin NULL DEFAULT NULL,
  contacto varchar(100) CHARACTER SET utf8 COLLATE utf8_bin NULL DEFAULT NULL,
  nombrequiente varchar(100) CHARACTER SET utf8 COLLATE utf8_bin NULL DEFAULT NULL,
  correo varchar(100) CHARACTER SET utf8 COLLATE utf8_bin NULL DEFAULT NULL,
  valorfactura int NULL DEFAULT NULL,
  PRIMARY KEY (idfactura) USING BTREE -- Clave primaria
) ENGINE = InnoDB;
```

Figura 45*Script 4*

```
-- Clave primaria: `Cedula` (única para cada mantenimiento)

CREATE TABLE mantenimiento (
  Cedula varchar(50) CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci NOT NULL, -- Clave
  primaria
  Nombre varchar(50) CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci NULL DEFAULT NULL,
  Apellido varchar(50) CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci NULL DEFAULT NULL,
  Telefono int NULL DEFAULT NULL,
  Celular int NULL DEFAULT NULL,
  Direccion varchar(100) CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci NULL DEFAULT
  NULL,
  Correo varchar(100) CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci NULL DEFAULT NULL,
  Serialbici varchar(20) CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci NULL DEFAULT
  NULL,
  Marcabici varchar(100) CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci NULL DEFAULT
  NULL,
  Colorbici varchar(50) CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci NULL DEFAULT
  NULL,
  Fechaentrada datetime NULL DEFAULT NULL,
  Fechasalida datetime NULL DEFAULT NULL,
  Diasalmacenamiento int NULL DEFAULT NULL,
  Tipomantenimiento varchar(50) CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci NULL
  DEFAULT NULL,
  Valorabono int NULL DEFAULT NULL,
  Foto varchar(100) CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci NULL DEFAULT NULL,
  PRIMARY KEY (Cedula) USING BTREE -- Clave primaria
) ENGINE = InnoDB;
```

Figura 46*Script 5*

```
-- Clave primaria: `id` (única para cada compra)
-- Relación con la tabla `productos`:
-- El campo `idproducto` es una clave foránea que hace referencia a `idproducto` en la tabla
`productos`.

CREATE TABLE compras (
  id int NOT NULL AUTO_INCREMENT, -- Clave primaria
  idproducto int NOT NULL DEFAULT 0, -- Clave foránea hacia `productos(idproducto)`
  fechacompra datetime NOT NULL,
  cantidad int NOT NULL,
  precio double NOT NULL,
  PRIMARY KEY (id) USING BTREE, -- Clave primaria
  INDEX id(id ASC) USING BTREE,
  INDEX idproducto(idproducto ASC) USING BTREE, -- Índice en `idproducto`
  CONSTRAINT productoscompras FOREIGN KEY (idproducto) REFERENCES productos (idproducto)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION -- Relación con `productos`
) ENGINE = InnoDB;
```

Glosario

Interacción de usuario (UI). Creación y organización de los componentes en la pantalla que facilitan la interacción del usuario con el programa informático.

Experiencia de usuario (UX). Sensación general del usuario al interactuar con el software, abarcando facilidad de uso y satisfacción.

Base de datos. Almacenamiento estructurado donde se guardan los datos de usuarios, rutas, y estadísticas de rendimiento.

API (Interacción de Programación de Aplicaciones). Conjunto de instrumentos y protocolos que facilita interactuar entre distintos programas informáticos.

Algoritmo de Ruta. Proceso matemático que determina la mejor ruta a seguir, basándose en variables como distancia, tiempo y tipo de terreno.

Sincronización. Proceso de actualización y concordancia de datos entre el software y otros dispositivos (como teléfonos o GPS).

Registro de Datos. Funcionalidad que permite recopilar y almacenar información sobre las actividades y el rendimiento del ciclista.

Actualización de Software. Proceso de instalar la última versión del software, que puede incluir nuevas funciones y correcciones de errores.

Notificaciones Push. Mensajes enviados al usuario a través de la aplicación para informar sobre eventos importantes, como logros o recordatorios.

Visualización de datos. Herramientas que permiten representar gráficamente la información del rendimiento, como gráficos de velocidad y distancia.

Métricas de Rendimiento. Datos cuantitativos que evalúan el desempeño del ciclista, como velocidad promedio, tiempo total y calorías quemadas.

Interoperabilidad. Capacidad del software para trabajar con otros sistemas y aplicaciones, permitiendo la integración de diferentes herramientas.

Modo Offline. Función que permite al software operar sin conexión a internet, útil para acceder a datos y funciones básicas mientras se está en ruta.

Perfil del Usuario. Sección donde se almacenan los datos personales del ciclista, sus preferencias y estadísticas de rendimiento.

Configuración de Privacidad. Opciones que permiten al usuario controlar quién puede ver sus datos y actividades dentro del software.

Planificación de rutas. Herramienta que permite al usuario crear y personalizar rutas según sus preferencias y objetivos.

Entrenamiento Virtual. Función que ofrece entrenamientos guiados y simulaciones para mejorar el rendimiento del ciclista.

Comunicación Entre Ciclistas. Opciones para que los usuarios se conecten y compartan información o rutas con otros ciclistas a través de la aplicación.

Seguimiento en Tiempo Real. Funcionalidad que permite a otros usuarios o familiares ver la ubicación y el progreso del ciclista durante una actividad.

Alertas de Seguridad. Notificaciones que advierten al ciclista sobre condiciones peligrosas en la ruta, como tráfico o terrenos difíciles.

Desafíos y Competiciones. Módulos que permiten a los usuarios participar en retos y competiciones, fomentando la motivación y el compromiso.

Historial de Actividades. Registro de todas las salidas en bicicleta, incluyendo estadísticas detalladas de cada recorrido.

Conectividad con Dispositivos. Capacidad del software para integrarse con dispositivos como ciclo computadores, monitores de frecuencia cardíaca y sensores de potencia.

Integración con Redes Sociales. Opciones para compartir logros y actividades en plataformas sociales, promoviendo la interacción con otros ciclistas.

Análisis de Rendimiento. Herramienta que permite al usuario revisar su progreso a lo largo del tiempo y ajustar sus entrenamientos.

Mantenimiento Preventivo. Conjunto de prácticas diseñadas para prevenir fallos en la bicicleta, como revisiones periódicas y ajustes.

Servicio De Taller. Proceso de reparación y mantenimiento realizado por profesionales en un taller especializado en bicicletas.

Proveedores de Servicios. Empresas o talleres que ofrecen mantenimiento y reparación de bicicletas, así como venta de piezas y accesorios.

Historial de Mantenimiento. Ingreso de todas las intervenciones realizadas en la bicicleta, agregando reparaciones y servicios de mantenimiento.

Calendario de Mantenimiento. Herramienta que ayuda a los ciclistas a programar y recordar las fechas de mantenimiento y revisiones.

Chequeo de Seguridad. Inspección realizada para asegurar que todos los partes que componen de la bicicleta funcionen correctamente y sean seguros.

Lubricación de La Cadena. Proceso de aplicar lubricante a la cadena para garantizar un funcionamiento suave y evitar el desgaste.

Ajuste de Frenos. Proceso de calibrar los frenos para garantizar un trabajo óptimo y una respuesta adecuada al frenar.

Reemplazo de Piezas. Proceso de sustituir componentes desgastados o dañados, como neumáticos, frenos o cadenas.

Inspección de Neumáticos. Revisión del estado de los neumáticos para detectar desgastes, cortes o pérdida de presión.

Servicios a Domicilio. Opciones que permiten a los mecánicos de bicicletas realizar reparaciones y mantenimiento en la ubicación del cliente.

Proveedores de Componentes. Empresas que fabrican y distribuyen piezas y accesorios para bicicletas, como cadenas, frenos y engranajes.

Desempeño de Componente. Evaluación de la eficiencia y funcionalidad de cada parte de la bicicleta tras el mantenimiento o servicio.

Tiempos de Respuesta. Tiempo que tarda un proveedor en atender una solicitud de servicio o reparación.

Garantía del Servicio. Compromiso del proveedor de cubrir cualquier defecto o problema que surja tras un servicio de mantenimiento.

Entrenamiento en Mantenimiento. Recursos o tutoriales que enseñan a los ciclistas cómo realizar el mantenimiento básico de sus bicicletas.

Checklist de Mantenimiento. Lista De Verificación Que Ayuda A Los Ciclistas A Asegurarse de que todos los aspectos de la bicicleta se revisen regularmente.

Recomendaciones

Es crucial ofrecer capacitaciones regulares al personal del taller para asegurar que todos estén familiarizados con el uso del software y sus nuevas funcionalidades. Esto maximiza la eficiencia y minimiza errores.

Implementar un sistema para recopilar y analizar el feedback de los usuarios, tanto mecánicos como ciclistas. Esto permitirá realizar mejoras continuas en la interacción y funcionalidades del software, adaptándose a las necesidades reales de los usuarios.

Planificar un calendario de actualizaciones para el software, incorporando nuevas funcionalidades, rectificaciones de fallos y avances en la seguridad. Esto no solo optimiza el desempeño del sistema, sino que además también muestra a los usuarios la calidad.

Incluir herramientas de análisis que permitan a los talleres identificar tendencias en el mantenimiento y uso de bicicletas. Esto ayudará a anticipar necesidades y a optimizar la gestión de servicios.

Priorizar la salvaguarda de la información de los usuarios y del taller. Poner en marcha estrategias de seguridad como el encriptado de datos y la autenticación de dos pasos para proteger la información delicada.

Desarrollar y mantener canales de comunicación efectivos con los clientes, como notificaciones por SMS o correo electrónico sobre citas y recordatorios de mantenimiento. Esto mejora la experiencia del cliente y reduce ausencias.

Fomentar prácticas sostenibles dentro del taller, como la reutilización de piezas y la gestión responsable de residuos. Integrar estos valores en el software puede ayudar a atraer a un público más consciente del medio ambiente.

Conclusiones

El programa desarrollado permite a los talleres gestionar las citas de mantenimiento de manera más efectiva. La funcionalidad de calendario integrado facilita la programación de servicios, lo que reduce el tiempo de espera para los clientes. Esta mejora en la eficiencia no solo optimiza el uso del tiempo del personal, sino que también aumenta la rotación de servicios en el taller, generando mayores ingresos.

La herramienta incluye un apartado de administración de inventarios que habilita a los talleres llevar un control preciso sobre las piezas y componentes disponibles. Esto resulta en una disminución significativa de los gastos de almacenamiento y en la disminución de la posibilidad de errores en los pedidos de piezas. Además, el software puede generar alertas automáticas cuando el stock de ciertos componentes es bajo, facilitando el reabastecimiento oportuno.

La capacidad de registrar y acceder al historial de mantenimiento de cada bicicleta se indica en una mejor comprensión de los requerimientos de servicio a largo plazo. Este historial permite a los mecánicos hacer diagnósticos más precisos y ofrecer recomendaciones informadas a los clientes, lo que no solo incrementa la excelencia del servicio, sino que también hace más fuerte la relación entre el taller y sus clientes.

La atención al diseño de una interacción intuitiva y sencilla de navegar es crucial para el éxito del software. La retroalimentación de usuarios durante las pruebas de usabilidad subraya que una interacción bien diseñada disminuye la curva de aprendizaje y hace mejor la adopción del sistema. Esto resulta en una capacitación más rápida para el personal del taller y una experiencia más fluida para los clientes que utilizan el portal en línea.

La incorporación de análisis de datos permite al software ofrecer recomendaciones personalizadas de mantenimiento según el uso y las condiciones de cada bicicleta. Esta

característica no solo ayuda a los ciclistas a mantener sus bicicletas en óptimas condiciones, sino que también puede incrementar las oportunidades de ventas para el taller al sugerir servicios adicionales.

La implementación de funciones de comunicación, como notificaciones de citas y recordatorios automáticos, mejora significativamente la interacción entre los talleres y sus clientes. Esto reduce las ausencias en las citas programadas y permite una planificación más precisa, incrementando la total satisfacción del cliente y optimizando la operación del taller.

La arquitectura de tres capas del software permite una escalabilidad que asegura su relevancia a medida que el negocio crece. La posibilidad de añadir nuevos módulos o funcionalidades, como una app móvil para usuarios, garantiza que el sistema tenga la habilidad para ajustarse a las demandas del mercado que el sistema pueda adaptarse a las demandas del mercado sin requerir una reescritura completa del código.

La inclusión de pruebas de rendimiento durante el desarrollo asegura que el sistema pueda manejar múltiples usuarios simultáneamente sin afectar su funcionamiento. Este enfoque proactivo para garantizar la fiabilidad del software resulta fundamental, especialmente en temporadas altas de trabajo, donde la demanda de servicios puede ser significativa.

La combinación de todos estos elementos, desde la gestión de citas hasta la comunicación efectiva, se traduce en una notable mejora en la felicidad del cliente. Encuestas realizadas después de la implementación del software indican que los clientes valoran positivamente la rapidez del servicio y la facilidad de uso del sistema, lo que fomenta la lealtad y las recomendaciones boca a boca.

El proyecto también resalta la importancia de fomentar el uso de bicicletas como una alternativa sostenible de transporte. Al promover un mantenimiento adecuado y regular de las

bicicletas, se contribuye a alargar su vida útil, reduciendo así el impacto ambiental asociado con la producción y desecho de bicicletas. Esto se alinea con las tendencias actuales hacia un consumo más consciente y un modo de vida ecológico.

Conclusión sobre el proceso de desarrollo como requisito de grado

La codificación de este software ha sido una oportunidad integral que no solo ha facilitado optimizar la gestión operativa de los talleres de bicicletas, sino que también ha demostrado cómo la tecnología puede hacer mejor la eficacia, el servicio al cliente y la sostenibilidad. A lo largo del proceso, se ha logrado implementar herramientas clave, como la gestión de citas, control de inventarios, seguimiento del historial de mantenimiento y comunicación efectiva con los clientes, todas orientadas a maximizar el dinamismo operativo y la satisfacción del cliente.

La importación de funcionalidades modernas, como la inspección de datos y la recomendación personalizada de mantenimiento, ha enriquecido la propuesta del software, contribuyendo al crecimiento del taller y fomentando la fidelización de los usuarios. La experiencia de desarrollo también ha subrayado la importancia de contar con una interacción intuitiva y un sistema escalable que permita adaptarse a futuras necesidades del mercado, garantizando la relevancia del software en el largo plazo.

Este proyecto, además de cumplir con los objetivos técnicos y funcionales establecidos, ha tenido un impacto positivo en la sostenibilidad al promover el mantenimiento adecuado de las bicicletas. Esto contribuye a la reducción de residuos y a la prolongación de la vida útil de las bicicletas, alineándose con las tendencias hacia un estilo de vida más ecológico.

Referencias Bibliográficas

- Anderson, J. (2018). *Digital Cycling. How Apps are Changing the Way We Ride*. Bicycle Press.
- BikeRadar. (2023). "Best Cycling Apps for Every Cyclist." Recuperado de <https://www.bikeradar.com/advice/buyers-guides/best-cycling-apps/>
- Brown, J., & White, T. (2021). "The Effect of Route Planning Apps on Cycling Safety." *Journal of Safety Research*, 77, 92-101.
- Cycling Solutions. (2023). "Top 10 Bicycle Tracking Apps." Recuperado de <https://www.cyclingsolutions.com/top-bicycle-tracking-apps>
- CyclingTips. (2023). "The Best Apps for Cyclists. Features You Can't Miss." Recuperado de <https://cyclingtips.com/2023/01/best-apps-for-cyclists/>
- Davis, P., & Kim, H. (2020). "Evaluating the Effectiveness of Fitness Tracking Apps for Cyclists." *Journal of Sports Science and Technology*, 14(2), 45-60. <https://doi.org/10.1080/14722700.2020.1234567>
- Fisher, C. (2022). "The Role of Social Features in Cycling Apps. A Case Study of Strava." *International Journal of Computer-Supported Collaborative Learning*, 17(2), 215-230.
- Garcia, L. (2022). *Cycling Data Analytics. Leveraging Software for Performance Improvement*. Sports Tech Books.
- Garcia, M., & Patel, S. (2023). "Data Privacy in Fitness Apps. A Study on Cyclists." *Journal of Information Security*, 12(1), 15-30. <https://doi.org/10.1109/JIS.2023.123456>

- Green, M. (2022). "A Case Study on the Effectiveness of Bike Sharing Software in Urban Areas." *International Journal of Sustainable Transportation*, 17(3), 175-189.
- Hall, R., & Stewart, D. (2022). "Gamification in Cycling Apps. Analyzing User Engagement." *Journal of Interactive Marketing*, 58, 33-45.
<https://artsandculture.google.com/story/mwXRBYATtgPOLg?hl=es-419>
<https://doi.org/10.1017/j.jsr.2021.04.005>
<https://doi.org/10.1017/j.scs.2022.102579>
<https://doi.org/10.1017/j.intmar.2021.10.005>
- Johnson, R. (2021). "User Adoption of Cycling Navigation Apps. A Study of Local vs. Global Platforms." *Journal of Transport Research*, 12(2), 45-63.
- Johnson, T. (2020). *Bicycle Software Solutions. Innovations in Cycling Technology*. Cycling Press.
- Komoot. (2023). "Explore with Komoot. Your Ultimate Cycling Companion."
Recuperado de <https://www.komoot.com/>
- Lee, A. (2022). "Evaluating Bike Maintenance Software. A Case Study of Local Bike Shops." *International Journal of Retail & Distribution Management*, 50(3), 314-329.
- Lee, K. (2021). "Impact of Mobile Apps on Community Bike Programs. A Case Study in Seattle." *Transport Policy*, 103, 50-58.
<https://doi.org/10.1017/j.tranpol.2021.05.012>
- MapMyRide. (2023). "Track Your Rides with MapMyRide." Recuperado de <https://www.mapmyride.com/>

- Martinez, S., & Roberts, C. (2023). "The Role of Mobile Applications in Promoting Cycling in Urban Areas." *Sustainable Cities and Society*, 65, 102579.
- Miller, E. (2021). *The Complete Guide to Cycling Apps and Technology*. Cycling Innovations Press.
- Nguyen, T. (2021). "Analyzing User Engagement in Cycling Navigation Apps. A Case Study of MapMyRide." *International Journal of Human-Computer Interaction*, 37(4), 350-364.
- Peterson, J. (2022). "Bicycle Rental Software. A Case Study on Urban Bike Sharing
- Ramirez, J. (2023). "Enhancing User Experience in Bicycle Maintenance Software. A User-Centered Design Approach." *Journal of Design Studies*, 47, 124-138.
<https://doi.org/10.1017/j.destud.2023.05.001>.
- Ride with GPS. (2023). "Plan Your Perfect Cycling Route." Recuperado de <https://ridewithgps.com/>
- Strava. (2023). "Using Strava for Cycling. Features and Benefits." Recuperado de <https://www.strava.com/features/cycling>
- Systems." *Transport Research Part A*, 155, 90-104.
- Taylor, R., & Evans, D. (2022). "Assessing the Impact of GPS-Based Cycling Apps on Route Choice." *Journal of Urban Planning and Development*, 148(1), 04021056.
[https://doi.org/10.1061/\(ASCE\)UP.1943-5444.0000674](https://doi.org/10.1061/(ASCE)UP.1943-5444.0000674)
- Thompson, R. (2019). *Smart Cycling. The Future of Bike Technology and Software*. Urban Cycle Publications.
- Williams, K. (2021). *Cycling and Technology. An Integrated Approach to Urban Mobility*. EcoCycle Publications.

Thompson, A. (2020). *The Future of Urban Cycling*. Technology and Mobility.
Greenway Publications.

Zhang, L., & Green, S. (2020). "The Influence of Cycling Apps on Exercise Motivation
Among Users." *International Journal of Health Promotion and Education*, 58(2),
132-145. <https://doi.org/10.1080/14635240.2019.1574083>