

INSTITUTO SUPERIOR TECNOLÓGICO

“SAN GABRIEL”



**ESPECIALIDAD INFORMÁTICA
PROYECTO DE TITULACIÓN**

PREVIO A LA OBTENCIÓN DEL TÍTULO DE:

TECNÓLOGO EN INFORMÁTICA MENCIÓN ANÁLISIS EN SISTEMAS

TEMA

IMPLEMENTACIÓN DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE
SERVICIOS TÉCNICOS A DOMICILIO Y CONTROL DE INVENTARIO STOCK
DE INSUMOS Y MATERIALES, UTILIZANDO EL FRAMEWORK PHP
LARAVEL Y GESTOR DE BASE DE DATOS MYSQL, EN EL CENTRO
COMERCIAL MACHADO, UBICADO EN LA CIUDAD DE RIOBAMBA EN EL

PERIODO 2021

AUTOR:

XAVIER PATRICIO BEJARANO GAVILANES

TUTOR:

ING. LUIS FREIRE S.

RIOBAMBA - ECUADOR

OCTUBRE, 2022

CERTIFICACIÓN

Certifico que el Sr. Xavier Patricio Bejarano Gavilanes, con el N° de Cédula 060419351-6 ha elaborado bajo mi Asesoría el Trabajo de Investigación titulado:

IMPLEMENTACIÓN DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE SERVICIOS TÉCNICOS A DOMICILIO Y CONTROL DE INVENTARIO STOCK DE INSUMOS Y MATERIALES, UTILIZANDO EL FRAMEWORK PHP LARAVEL Y GESTOR DE BASE DE DATOS MYSQL, EN EL CENTRO COMERCIAL MACHADO, UBICADO EN LA CIUDAD DE RIOBAMBA EN EL PERIODO 2021.

Por tanto, autorizo la presentación para la calificación respectiva.

Ing. Luis Freire S.

TUTOR DE TESIS

“El presente Trabajo de Investigación constituye un requisito previo para la obtención del Título de **Tecnólogo en Informática mención Análisis de Sistema**”



“Yo, Xavier Patricio Bejarano Gavilanes con N° de Cédula 060419351-6, declaro que la investigación es absolutamente original, autentica, personal y los resultados y conclusiones a los que se han llegado es de mi absoluta responsabilidad.”

Xavier Patricio Bejarano Gavilanes

INSTITUTO SUPERIOR TECNOLÓGICO
“SAN GABRIEL”
ESPECIALIDAD INFORMÁTICA
TRABAJO DE INVESTIGACIÓN
PREVIA A LA OBTENCION DEL TÍTULO DE:
TECNÓLOGO EN INFORMÁTICA MENCION ANÁLISIS DE SISTEMAS

TÍTULO:

IMPLEMENTACIÓN DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE SERVICIOS TÉCNICOS A DOMICILIO Y CONTROL DE INVENTARIO STOCK DE INSUMOS Y MATERIALES, UTILIZANDO EL FRAMEWORK PHP LARAVEL Y GESTOR DE BASE DE DATOS MYSQL, EN EL CENTRO COMERCIAL MACHADO, UBICADO EN LA CIUDAD DE RIOBAMBA EN EL PERIODO 2021.

APROBADO

ASESOR DE TESIS DE GRADO

PRESIDENTE DEL TRIBUNAL

MIEMBRO DEL TRIBUNAL

MIEMBRO DEL TRIBUNAL

FIRMAS DE RESPONSABILIDAD

NOMBRES	FECHA	FIRMA
DIRECTOR DE TESIS
PRESIDENTE DEL TRIBUNAL
MIEMBRO DEL TRIBUNAL
MIEMBRO DEL TRIBUNAL
NOTA FINAL DE LA TESIS		
.....		

DEDICATORIA

Dedico este trabajo a Dios, mi esposa Daniela y mi hijo Ethan por todo el apoyo brindado en toda mi trayectoria estudiantil.

Xavier Patricio

AGRADECIMIENTO

Un agradecimiento a Dios y a los profesores del Instituto Superior Tecnológico “San Gabriel”, quienes con su entrega y sapiencia guiaron mi camino para la culminación de mi carrera estudiantil.

Xavier Patricio

ABREVIATURAS

POO: Programación Orientada a Objetos

APP: Application

PHP: Hypertext Pre-Processor

SQL: Structured Query Language

WWW: World Wide Web

HTML: Hyper Text Markup Language

CSS: Cascading Style Sheets

URL: Uniform Resource Locator

OS: Operating System

DBMS: Data Base Management System

AMD: Advanced Micro Devices

ASCII: American Standard Code for Information Interchange

CPU: Central Processing Unit

RAM: Random Access Memory

DDL: Data Definition Language

DMA: Direct Memory Access

DRAM: Dynamic RAM

FTP: File Transfer Protocol

GB: Giga Bytes

GLOSARIO DE TÉRMINOS

Laravel: Es un framework de aplicación web con una sintaxis precisa y distinguida.

Web App: Es un programa de aplicación que se almacena en un servidor remoto y se entrega a través de Internet a través de una interfaz de navegador.

Framework: Es una herramienta que proporciona componentes listos para usar o soluciones que se personalizan para acelerar el desarrollo.

React: es una biblioteca de JavaScript declarativa, eficiente y flexible para crear interfaces de usuario.

Tailwind: Es básicamente un framework utilitario CSS de primer orden, sirve para crear una interfaz de usuario personalizada rápida.

Servidor: Es un programa o dispositivo informático que proporciona un servicio a otro programa informático y a su usuario, también conocido como cliente.

Cliente: Es una pieza de hardware o software de computadora que accede a un servicio puesto a disposición por un servidor como parte del modelo cliente-servidor de redes informáticas.

Web Server: Es un programa de computadora que sirve páginas o archivos HTML solicitados. En este caso, un navegador web actúa como cliente.

Application Server: Es un programa en una computadora en una red distribuida que proporciona la lógica comercial para un programa de aplicación.

MySQL: es un sistema de gestión de bases de datos relacionales (RDBMS) de código abierto. Su nombre es una combinación de "My", el nombre de My, la hija del cofundador Michael Widenius, y "SQL", la abreviatura de Structured Query Language.

TABLA DE CONTENIDO

CERTIFICACIÓN.....	II
FIRMAS DE RESPONSABILIDAD	V
DEDICATORIA.....	VI
AGRADECIMIENTO	VII
ABREVIATURAS	VIII
GLOSARIO DE TÉRMINOS	IX
ÍNDICE DE TABLAS.....	XIII
ÍNDICE DE GRÁFICOS	XIV
INTRODUCCIÓN.....	XVI
RESUMEN.....	XVIII
SUMMARY	XIX
CAPÍTULO I.....	1
1. MARCO REFERENCIAL	1
1.1. Formulación general del proyecto de tesis	2
1.1.1. Antecedentes del problema de investigación	2
1.1.2. Datos informativos	3
1.2. Definición del problema.....	3
1.2.1. Justificación.....	3
1.2.2. Objetivos	4
1.2.2.1. Objetivo general	4
1.2.2.2. Objetivos específicos.....	4
CAPÍTULO II.....	5
2. MARCO TEORICO	5
2.1. Aplicación web.....	6
2.2. Php.....	6

2.2.1.	Características y ventajas de php.....	8
2.2.2.	Composer.....	8
2.3.	Framework Laravel	9
2.3.1.	Características y ventajas de Laravel	10
2.3.2.	Configuración de Laravel.....	10
2.3.3.	Parametrización inicial de Laravel	13
2.3.4.	Configuración basada en entorno	13
2.3.5.	Tipos de variables de entorno.....	14
2.3.6.	Configuración de caché	15
2.3.7.	Directorio.....	15
2.3.7.1.	El directorio de aplicaciones	16
2.3.7.2.	Directorio de configuración.....	16
2.3.7.3.	Directorio de la base de datos.....	16
2.3.8.	Despliegue	17
2.3.8.1.	Requisitos del servidor	17
2.3.8.2.	Configuración del servidor con nginx	17
2.4.	MySQL.....	18
2.4.1.	Características y ventajas de MySQL	19
2.4.2.	Compatibilidad entre MySQL y Php.....	19
2.5.	React JS	20
2.5.1.	Características y ventajas de React	21
2.5.2.	Frontend con React.....	22
2.5.3.	Configuración de React.....	22
2.6.	Tailwind CSS	23
2.6.1.	Características y ventajas de Tailwind	24
2.7.	Visual Studio Code.....	24
	CAPÍTULO III	26

3.	ANÁLISIS Y DISEÑO	26
3.1.	Recolección de la información	27
3.1.1.	Alcance	27
3.1.2.	Análisis	28
3.1.3.	Estudio de factibilidad.....	28
3.1.3.1.	Factibilidad operativa	29
3.1.3.2.	Factibilidad técnica.....	29
3.1.3.3.	Factibilidad económica.....	30
3.1.3.4.	Factibilidad legal	30
3.1.4.	Análisis de los requerimientos	31
3.1.4.1.	Requerimientos funcionales del sistema	31
3.1.4.2.	Requerimientos no funcionales del sistema	31
3.1.5.	Definición de casos de uso	32
3.2.	Diseño.....	37
3.2.1.	Diseño conceptual	37
3.2.2.	Modelo relacional.....	37
3.2.3.	Diccionario de datos.....	38
3.2.4.	Diseño de interfaces.....	40
	CAPÍTULO IV	43
4.	IMPLEMENTACIÓN DE LA APLICACIÓN WEB	43
4.1.	Configuración de las herramientas de desarrollo	44
4.2.	Arquitectura del sistema.....	45
4.3.	Implementación del aplicativo web.....	46
4.3.1.	Definición de módulos	46
4.3.2.	Implementación de la aplicación.....	47
4.4.	Pruebas de aplicación	47
4.5.	Capacitación al personal.....	48

4.6. Mantenimiento de la aplicación	48
CONCLUSIONES.....	49
RECOMENDACIONES	50
BIBLIOGRAFÍA	51
ANEXOS.....	53
ANEXO 1.....	54
MANUAL DEL USUARIO	54
ANEXO 2.....	66
MANUAL DEL TÉCNICO	66

ÍNDICE DE TABLAS

Tabla N° 1 Características y ventajas PHP.....	10
Tabla N° 2 Características y ventajas Laravel.....	12
Tabla N° 3 Variables de entorno	17
Tabla N° 4 Características y ventajas MySQL.....	22
Tabla N° 5 Tipos de datos numéricos.....	23
Tabla N° 6 Tipos de datos fecha.....	24
Tabla N° 7 Tipos de datos texto	25
Tabla N° 8 Compatibilidad MySQL - PhP.....	26
Tabla N° 9 Características y ventajas React.....	27
Tabla N° 10 Características y ventajas Tailwind	31
Tabla N° 11 Pruebas de aplicación.....	57

ÍNDICE DE GRÁFICOS

Gráfico N° 1 Logo Php.....	6
Gráfico N° 2 Logo Composer.....	8
Gráfico N° 3 Logo Laravel.....	9
Gráfico N° 4 Logo MySQL.....	18
Gráfico N° 5 Logo React.....	20
Gráfico N° 6 Logo Tailwind.....	23
Gráfico N° 7 Logo Visual Code	24
Gráfico N° 8 Ingreso al sistema.....	32
Gráfico N° 9 Nuevo usuario	33
Gráfico N° 10 Inventario	34
Gráfico N° 11 Orden de servicio	34
Gráfico N° 12 Reporte - Ordenes de Ingreso	35
Gráfico N° 13 Reporte - Ordenes de Salida	36
Gráfico N° 14 Reporte - Servicios.....	36
Gráfico N° 15 Diseño Conceptual	37
Gráfico N° 16 Modelo Relacional	37
Gráfico N° 17 Diccionario de Datos 1.....	38
Gráfico N° 18 Diccionario de Datos 2.....	39
Gráfico N° 19 Diccionario de Datos 3.....	40
Gráfico N° 20 Interfaz Login.....	40
Gráfico N° 21 Interfaz Dashboard.....	41
Gráfico N° 22 Interfaz Usuarios.....	41
Gráfico N° 23 Interfaz Inventario.....	42
Gráfico N° 24 Interfaz Servicios	42
Gráfico N° 25 Instalación Xampp	44
Gráfico N° 26 Entorno Xampp.....	44
Gráfico N° 27 Confirmación Xampp.....	45
Gráfico N° 28 Panel de control Xampp.....	45
Gráfico N° 29 Arquitectura de aplicación	46
Gráfico N° 30 Módulos de aplicación	47
Gráfico N° 31 Guía ingreso al sistema	58
Gráfico N° 32 Guía acceso a Dashboard.....	59

Gráfico N° 33 Guía acceso a menú de usuario.....	59
Gráfico N° 34 Guía acceso a añadir usuario.....	60
Gráfico N° 35 Guía acceso a inventario	60
Gráfico N° 36 Guía añadir material e insumo	61
Gráfico N° 37 Guía menú equipos	62
Gráfico N° 38 Guía añadir equipos	62
Gráfico N° 39 Guía menú servicios.....	63
Gráfico N° 40 Guía añadir orden de reparación	63
Gráfico N° 41 Guía añadir informe técnico.....	64
Gráfico N° 42 Guía reportes.....	65
Gráfico N° 43 Guía modificar perfil.....	65

INTRODUCCIÓN

El Centro Comercial Machado desde 1985 viene brindando a la colectividad riobambeña productos y servicios de alta calidad, con el paso del tiempo y ante los nuevos desafíos tecnológicos tiene la necesidad de automatizar y gestionar de una mejor manera sus procesos administrativos; en este sentido la empresa trabaja con programas contables para la gestión de su facturación, sin embargo; dichos programas no cubren la gestión de servicios técnicos a domicilio o el inventario stock de insumos y materiales. Al no existir un software adecuado que gestione los servicios técnicos a domicilio e inventario stock de insumos y materiales se han encontrado los siguientes problemas: confusión al momento de generar ordenes de servicio, pérdida de información y falta de reportes.

Ante esta problemática se propone al Centro Comercial Machado la “Implementación de una aplicación web para la gestión de servicios técnicos a domicilio y control de inventario stock de insumos y materiales, utilizando el Framework PHP Laravel y gestor de base de datos MySQL”, con lo cual se podrá: gestionar cada orden de servicio a domicilio de una manera ágil y organizada, tener archivada toda la información inherente a cada servicio técnico, generar reportes y controlar eficazmente el inventario stock de insumos y materiales.

La motivación inherente para la realización del presente proyecto nace de la predisposición que el Centro Comercial Machado ha demostrado para con el ponente y la institución, cuando mediante convenios de cooperación ha facilitado el proceso de prácticas pre profesionales.

Para alcanzar lo propuesto, se plantea como objetivo general: Implementar una aplicación web para la gestión de servicios técnicos a domicilio y control de inventario stock de

insumos y materiales, utilizando el framework PHP Laravel y gestor de base de datos MySQL.

Finalmente, se indica que el trabajo de investigación se desarrolla en cinco capítulos que se describen a continuación:

El capítulo primero, contiene el marco referencial, donde se describen los antecedentes y la información inherente a la empresa, así como también se determina el planteamiento del problema, la justificación y los objetivos del trabajo de investigación, aspectos que proporcionaron el entendimiento de las necesidades de la empresa y el direccionamiento de la investigación.

En el capítulo segundo, se desarrolla el marco teórico que contiene conceptos esenciales sobre las tecnologías del proyecto: PHP, Laravel, MySQL, React y Tailwind; herramientas que permitieron trabajar en el backend y frontend de la app web.

En el capítulo tercero, se desarrolla el análisis y diseño de la aplicación web, así como también la funcionalidad de sus diferentes módulos, en este apartado se describe la recolección de datos, factibilidad, casos de uso, requerimientos de funcionalidad y la delineación de la interfaz; logrando una aplicación web intuitiva para el usuario.

El capítulo cuarto, contiene el proceso de implementación de la aplicación web, donde se determinan aspectos como la configuración de las tecnologías, la arquitectura de la aplicación, definición de los módulos y el desarrollo e implementación de aplicativo, culminando con las pruebas de funcionamiento, capacitación al personal y mantenimiento.

En el capítulo quinto, se describen las conclusiones y recomendaciones del proyecto de acuerdo a los objetivos planteados que han sido alcanzados, seguido de dos anexos: el manual de usuario y el manual técnico.

RESUMEN

Se implementó una app web para el manejo de servicios técnicos a domicilio y control de inventario stock de insumos y materiales, utilizando el framework PHP Laravel y gestor de base de datos MySQL en el centro comercial machado, ubicado en la ciudad de Riobamba, en el periodo 2021. El proceso metodológico de desarrollo de software, partió del análisis de las necesidades de la empresa, para lo cual se realizaron entrevistas al encargado de TIC de la empresa Ing. Fabián Machado, sobre la base de sus resultados se propone la implementación de una app web para el manejo de servicios técnicos a domicilio e inventario stock de insumos y materiales. El framework para el aplicativo fue Laravel, el backend fue programado con PHP, y en el frontend se usó React y Tailwind CSS. Para el almacenamiento de datos se utilizó el gestor de base de datos MySQL. Una vez desarrollada la aplicación web se realizaron pruebas de verificación, se brindó una capacitación a los diferentes usuarios y se dio parámetros necesarios para el mantenimiento de la app web. Los resultados conseguidos implementando la aplicación web fueron: automatizar el proceso de gestión de servicios técnicos a domicilio, control y seguimiento del inventario stock de insumos y materiales, división por roles para administradores, técnicos y clientes, generación de base de datos de equipos, generación de historiales de reparación mediante filtros de búsqueda globales, generación de espacios de intercambio virtual entre el cliente y la empresa, actualización de estados de los diferentes servicios técnicos, generación de comprobantes de entrada y de salida de equipos y generación de reportes de servicios.

SUMMARY

A web app was implemented for the management of home technical services and inventory control stock of supplies and materials, using the PHP Laravel framework and MySQL database manager in the Machado shopping center, located in the city of Riobamba, in the period 2021. The methodological process of software development, started from the analysis of the needs of the company, for which interviews were conducted with the ICT manager of the company Ing. Fabián Machado, based on their results, the implementation of a web app for managing technical services at home and stock inventory of supplies and materials. The framework for the application was Laravel, the backend was programmed with PHP, and the frontend used React and Tailwind CSS. For data storage, the MySQL database manager was used. Once the web application was developed, verification tests were carried out, training was provided to the different users and the necessary parameters were given for the maintenance of the web app. The results achieved by implementing the web application were: automate the management process of technical services at home, control and monitoring of inventory stock of supplies and materials, division by roles for administrators, technicians and customers, generation of equipment database, generation of repair histories through global search filters, generation of virtual exchange spaces between the client and the company, updating of the status of the different technical services, generation of entry and exit receipts for equipment and generation of service reports.

CAPÍTULO I

1. MARCO REFERENCIAL

1.1. Formulación general del proyecto de tesis

1.1.1. Antecedentes del problema de investigación

En la ciudad de Riobamba, el Centro Comercial Machado, apertura sus puertas en 1985, primero con la comercialización de electrodomésticos y luego con la comercialización de tecnología. Cada década brindo a la empresa el aprendizaje necesario para gestionar sus procesos administrativos, desde los primeros años existen registros exactos de contabilidad y gestión de servicios técnicos; dichos registros sirvieron de base para la futura implementación de tecnologías, donde se pasó de anotar en libretas de papel a manejar hojas de cálculo y software informático.

Pensando siempre en la innovación y dinámica empresarial, el Centro Comercial Machado ha llevado en orden su contabilidad con la ayuda de programas informáticos de alta calidad; sin embargo, ha descuidado la gestión de servicios técnicos a domicilio e inventario stock. Con la ausencia de un software adecuado que agencie los servicios técnicos a domicilio e inventario stock de insumos y materiales se han encontrado los siguientes inconvenientes: desorden al momento de generar ordenes de servicio, mala gestión de información, imposibilidad de generar reportes, pérdidas económicas en inventario stock de insumos y materiales utilizados para los servicios técnicos realizados.

A nivel local y nacional existen programas que apoyan a los laboratorios de servicio técnicos en empresas como SystemMarket, IDC, INTCOMEX y CARTIMEX, pero son programas ineficientes pues no cuentan con un módulo de control de inventario, no se enfocan en los servicios técnicos a domicilio y su aplicación es de escritorio; al mismo tiempo las alternativas que existen en el mercado a nivel mundial como: SGtaller, ERP, Gestioo, Persat, Synchronteam son programas con licencia uso con costos muy elevados.

1.1.2. Datos informativos

Misión

Comercial Machado tiene como misión brindar a la ciudadanía en general los mejores productos que existen en el mercado, cuidando siempre su calidad enmarcada en normas internacionales; además de los mejores costos posibles cuidando siempre el bolsillo de nuestros clientes.

Visión

Comercial Machado tiene como visión ser pionera en la comercialización de productos no solo a nivel local, sino también a nivel nacional, estando presentes en cada una de las 24 provincias del Ecuador, dando facilidades de acceso y comodidad a nuestros clientes y promoviendo el desarrollo económico y el empleo.

1.2. Definición del problema

1.2.1. Justificación

El presente proyecto de investigación es viable por ser una propuesta innovadora que solventarán todos los problemas suscitados en el área de servicios técnicos a domicilio del Centro Comercial Machado. La aplicación automatizará el proceso de gestión de servicios técnicos a domicilio e inventario stock de insumos y materiales, será un software de autor con código original y creado desde cero, minimizará costos de implementación ya que se encontrará alojada en la nube, será accesible ya que se podrá utilizar desde cualquier parte del mundo solo necesitando una conexión de internet y un navegador, no será necesario tener una instalación previa en el equipo cliente, tendrá una gestión por roles para administradores, técnicos y clientes, generará una de base de datos de equipos, usuarios y servicios, dispondrá de historiales de reparación mediante filtros de búsqueda globales, establecerá espacios de intercambio virtual entre el cliente y la empresa,

actualizará los estados de los diferentes servicios técnicos y brindará reportes de entrada, salida y coste de servicios.

Se justifica el uso del Framework PHP Laravel y gestor de base de datos MySQL, por requerimiento específico del Ing. Fabián Machado jefe del área informática de la empresa, pues este departamento del Comercial Machado tiene experiencia en el uso de Laravel y MySQL y se encargara de futuros mantenimientos y actualizaciones del software.

1.2.2. Objetivos

1.2.2.1. Objetivo general

- Implementar una aplicación web para la gestión de servicios técnicos a domicilio y control de inventario stock de insumos y materiales, utilizando el framework PHP Laravel y gestor de base de datos MySQL, en el Centro Comercial Machado, ubicado en la ciudad de Riobamba en el periodo 2021.

1.2.2.2. Objetivos específicos

- Investigar tecnologías: Framework PHP Laravel, MySQL, React Js y Tailwind CSS.
- Identificar las necesidades y requerimientos en el proceso de gestión de servicios técnicos a domicilio he inventario stock de insumos y materiales del Centro Comercial Machado.
- Desarrollar, e implementar el aplicativo web para la gestión de servicios técnicos a domicilio he inventario stock de insumos y materiales en base a los requerimientos.

CAPÍTULO II

2. MARCO TEORICO

2.1. Aplicación web

Se denomina aplicación web a las herramientas que los usuarios pueden usar web a través de una red de Internet o de una intranet accediendo a un equipo servidor a través de un navegador.

Las aplicaciones se han transformado en pieza fundamental de la vida cotidiana gracias a su uso en Tablet y smartphones, también hay algunas aplicaciones competentes para ordenadores de escritorio. Mientras que una aplicación nativa es un tipo de software que se adecúa a una determinada plataforma, una web app tiene lugar en el navegador web y guarda algunas diferencias con las llamadas aplicaciones nativas. (ICTEA, 2020).

Son notorias las aplicaciones web debido a la libertad del sistema operativo, así como a la desventaja para mantener aplicaciones web y actualizar sin instalar ni distribuir programas a miles de beneficiarios potenciales.

Es sustancial aludir que una página Web contiene elementos que admiten una comunicación dinámica entre la información y el usuario. Esto permite que el usuario tenga acceso a los datos de un modo interactivo, debido a que expresará la página a cada una de sus acciones, como muestra: enviar y rellenar formularios, participar en juegos y tener acceso a gestores de base de datos de todo tipo. (IONOS España S.L.U., 2018)

2.2. Php

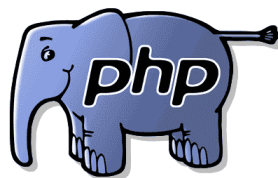


Gráfico No. 1 Logo PhP

Tomado de: <https://www.gettyimages.es>

PHP parte del acrónimo Hypertext Preprocessor, este código se describe como un lenguaje de 'scripting' de código abierto y de propósito general que está principalmente

pensado para el desarrollo web y que es capaz de ser embebido en páginas HTML. Su sintaxis recurre a Java, C, y Perl, por lo que es fácil de aprender. El principal objetivo de este tipo de código es consentir a los programadores web escribir rápidamente y dinámica páginas web generadas; aunque PHP no solo se limita a este uso. (Cowburn, 2018)

Ejemplo de código en PHP:

```
-<!DOCTYPE html>
-<html>
  -<head>
    -<title>Ejemplo PHP</title>
  -</head>
  -<body>

    -<?php
      -echo "¡Hola mundo!";
    -?>
  -</body>
-</html>
```

En lugar de utilizar varios comandos para mostrar HTML (como en Perl o C), las páginas de PHP contienen HTML con código alojado que hace "algo" (en este caso, mostrar "¡Hola mundo!"). El código de PHP está contenido entre las etiquetas especiales de comienzo y final <?php y ?> que admiten entrar y salir del "modo PHP".

Lo que diferencia a PHP de tecnologías del lado del equipo cliente como código de tipo Javascript es que la programación es hecha en el servidor, creando HTML y enviándolo al cliente. El cliente cogerá el resultante de aplicar el script, aunque no será posible acceder al subyacente código. Puede ser arreglado el servidor web incluso para que procese la totalidad de ficheros HTML con PHP, por lo que no existe la posibilidad de que los usuarios puedan saber o modificar la codificación interna.

2.2.1. Características y ventajas de php

Tabla No. 1 Características y ventajas PHP

CARACTERÍSTICA	VENTAJA
Lenguaje Libre	PHP al ser un lenguaje libre posee licencia abierta y una gran comunidad de soporte en línea.
Multiplataforma	Gran compatibilidad y adaptabilidad con los diferentes sistemas operativos.
Embebido HTML	Al permitir la introducción de código en otro lenguaje como HTML, PHP se vuelve más ágil y potente.
Sinergia con varios tipos de base de datos	Al trabajar del lado del servidor PHP tiene entre sus ventajas la compatibilidad y fácil adaptación a diferentes tipos de bases de datos como MySQL.
Escalable	Fameworks como Laravel permiten al Lenguaje Php generar proyectos modulares y de fácil ampliación.
Fácil aprendizaje	Sentencias, Variables, Tipo de datos, etc.; son declaradas de manera fácil en el Lenguaje ya que se aplican conocimientos de programación básica.

Elaborado por: Xavier Bejarano

2.2.2. Composer



Gráfico No. 2 Logo Composer
Tomado de: <https://www.gettyimages.es>

Un manejador de paquetes para PHP es la definición de Composer, este a su vez proporciona un modelo para descargar, administrar, e instalar librerías y dependencias. Le permite describir de qué bibliotecas depende el proyecto y las mismas serán instaladas dentro de cada proyecto por separado, Composer es la ideal respuesta cuando desarrollamos complejos proyectos que estriban de fuentes múltiples de instalación. Composer facilita el trabajo pues descarga de forma automática todas las dependencias

necesarias para no tener que realizar este proceso de forma manual. (Lashutko & Koziel, 2019)

2.3. Framework Laravel



Gráfico No. 3 Logo Laravel

Tomado de: <https://www.gettyimages.es>

Laravel es un Framework de aplicación web con una sintaxis formal y distinguida. Un marco web (Framework) brinda una estructura y un punto de partida para crear aplicaciones robustas, esto permite al programador enfocarse en crear algo increíble mientras el framework se preocupa por los detalles. (Otwell, 2021)

Laravel ofrece características poderosas como una capa de abstracción de base de datos formal, una inyección de dependencia completa, colas y trabajos programados, pruebas unitarias y de integración, etc. (Otwell, 2021)

Laravel es un framework que puede crecer con el programador, ya sea que el mismo sea novato en PHP, diseño web o tenga años de experiencia, Laravel ayuda al desarrollador a dar sus primeros pasos y lo impulsa a medida que lleva su experiencia al siguiente nivel.

2.3.1. Características y ventajas de Laravel

Tabla No. 2 Características y ventajas Laravel

CARACTERÍSTICA	VENTAJA
Framework progresivo	Laravel crece con el programador. Si recién se está dando los primeros pasos en el desarrollo web, la amplia biblioteca de documentación, tutoriales y guías en video de Laravel ayudará al programador a aprender el uso del framework sin sentirse abrumado. Si se es un desarrollador sénior, Laravel brinda herramientas sólidas para inyección de dependencias, colas, test unitarios y eventos en tiempo actual. Optimizado esta Laravel para desarrollar profesionales aplicaciones web, estando lista también para gestionar cargas empresariales de trabajo.
Framework escalable	Laravel es un framework escalable. Gracias a la naturaleza compatible con el escalado de PHP y el integrado soporte de Laravel para aplicaciones como Redis que usan caché distribuidos, el escalado horizontal con Laravel es sencillo. Con Laravel existen desarrolladas muchísimas aplicaciones web que se han encumbrado sencillamente para gestionar cientos de millones de solicitudes mensuales. Las plataformas como Laravel Vapor permiten ejecutar la aplicación web a una escala casi ilimitada en la última tecnología sin servidor de AWS.
Framework comunitario	Laravel combina un ecosistema con los mejores paquetes de PHP para brindar el marco disponible más robusto y fácil de usar para desarrolladores. Además, muchos desarrolladores de varias partes del mundo han apoyado con Laravel.

Elaborado por: Xavier Bejarano

2.3.2. Configuración de Laravel

Antes de crear una nueva aplicación Laravel en Windows, se recomienda instalar Docker Desktop. A continuación, se debe asegurar que el Subsistema de Windows para Linux 2 (WSL2) esté instalado y habilitado. WSL permite elaborar en Linux ejecutables binarios de nativa forma en Windows 10. Se puede encontrar información sobre cómo instalar y habilitar WSL2 en la documentación del entorno de desarrollo de Microsoft. (Otwell, 2021)

Se debe iniciar Windows Terminal y comenzar una nueva sesión de terminal para el sistema operativo WSL2 Linux. A continuación, se puede usar un comando de terminal simple para crear un nuevo proyecto de Laravel. Por ejemplo, para crear una nueva aplicación Laravel en un directorio llamado "aplicación-ejemplo", se puede ejecutar el siguiente comando en su terminal:

```
curl -s https://laravel.build/example-app | bash
```

Se creará dentro del directorio del programa Laravel cuando se ejecuten los comandos descritos.

Una vez creado el proyecto, se puede navegar hasta el directorio de la aplicación e iniciar Laravel Sail. Laravel Sail proporciona una interfaz de línea de comandos simple para interactuar con la configuración predeterminada de Docker de Laravel:

```
cd example-app  
./vendor/bin/sail up
```

La primera vez que se ejecute el comando Navegar hacia arriba, los contenedores de aplicaciones de Navegar se crearán en la máquina. Esto podría tomar varios minutos. Una vez que se hayan iniciado los contenedores Docker de la aplicación, puede acceder a la aplicación en el navegador web en: <http://localhost>

Para modificar los archivos de la aplicación Laravel que se crearon dentro de la instalación de WSL2 se recomienda usar el editor de código de Visual Studio de Microsoft y su extensión propia para el desarrollo remoto.

Se tiene que asegurar colocar el directorio bin de proveedores de todo el sistema de Composer en el \$PATH para que el sistema pueda ubicar el ejecutable Laravel. Este directorio existe en diferentes ubicaciones según el sistema operativo; sin embargo, algunas ubicaciones comunes incluyen:

- macOS: `$HOME/.composer/vendor/bin`
- Windows: `%USERPROFILE%\AppData\Roaming\Composer\vendor\bin`
- GNU / Linux Distributions: `$HOME/.config/composer/vendor/bin` or `$HOME/.composer/vendor/bin`.

Para mayor comodidad, el instalador de Laravel también puede crear un repositorio Git para el nuevo proyecto. Para crear un repositorio de Git, se pasa el indicador `--git` al crear un nuevo proyecto:

```
laravel new example-app --git
```

Este comando inicializará un nuevo repositorio de Git para el proyecto y confirmará automáticamente el esqueleto base de Laravel. El indicador de git asume que se ha instalado y configurado correctamente Git. También se puede usar la bandera `--branch` para establecer el nombre de la rama inicial:

```
laravel new example-app --git --branch="main"
```

En lugar de usar el indicador `--git`, también se puede usar el indicador `--github` para crear un repositorio de Git y también crear un repositorio privado correspondiente en GitHub:

```
laravel new example-app --github
```

El repositorio creado estará disponible en `https://github.com/<your-account>/example-app`. El indicador de github asume que se ha instalado correctamente la CLI de GitHub y está autenticado con GitHub. Además, se debe tener git instalado y configurado correctamente. Si es necesario, es posible pasar indicadores adicionales compatibles con la CLI de GitHub:

```
laravel new example-app --github="--public"
```

2.3.3. Parametrización inicial de Laravel

Se guardan en el directorio de configuración la totalidad de los archivos de configuración de Laravel. Cada opción está documentada, así que se puede revisar los archivos para familiarizarse con las opciones disponibles. (Otwell, 2021)

Laravel casi no necesita configuración adicional lista para usar. Sin embargo, es posible que se desee revisar el archivo `config/app.php` y su documentación; este archivo contiene varias opciones, como la zona horaria y la configuración regional, que es posible que se desee cambiar según la aplicación.

Desde la raíz del "directorio web" Laravel siempre debe ejecutarse, configurado siempre para el servidor web. Ejecutar una aplicación Laravel no se debe intentar desde un subdirectorio del "directorio web". Intentar hacerlo podría exponer archivos confidenciales que existen dentro de la aplicación. (Otwell, 2021)

2.3.4. Configuración basada en entorno

Dado que muchos de los valores de las opciones de configuración de Laravel pueden variar dependiendo de si la aplicación se ejecuta en una computadora local o en un servidor web de producción, muchos valores de configuración importantes se definen utilizando el archivo `.env` que existe en la raíz de la aplicación. (Otwell, 2021)

El archivo `.env` no debe comprometerse con el control de fuente de la aplicación, ya que cada desarrollador/servidor que use el programa podría solicitar una configuración diferente de entorno. Por consiguiente, esto se convertiría en un riesgo de seguridad en caso de que un husmeador obtenga acceso al repositorio de control de código fuente, ya que quedaría expuesta cualquier credencial personal.

Suele ser útil tener diferentes valores de configuración según el ambiente en el que se ejecuta el programa. Es posible, por ejemplo, que se desee utilizar localmente un

controlador de caché diferente al que utiliza en el servidor de producción. Laravel utiliza la biblioteca PHP DotEnv. En una nueva instalación del framework Laravel, el directorio raíz del programa contendrá un archivo `.env.example` que define muchas variables de entorno comunes. Durante el proceso de instalación de Laravel, este archivo automáticamente se clonará a `.env`. (Otwell, 2021)

El archivo `.env` predeterminado de Laravel contiene algunos valores de configuración comunes que pueden diferir en función de si su aplicación se ejecuta localmente o en un servidor web de producción. Estos valores luego se recuperan de varios archivos de configuración de Laravel dentro del directorio de configuración usando la función `env` de Laravel.

Si con un equipo se está desarrollando, es posible que se apetezca continuar incluyendo un archivo `.env.example` con el programa. Al ubicar valores de marcador de posición en el file de configuración de ejemplo, pueden ver claramente, otros desarrolladores del equipo, las variables de entorno que se necesitan para ejecutar el programa.

2.3.5. Tipos de variables de entorno

Todas las variables en los archivos `.env` generalmente como cadenas se analizan, por lo que se han desarrollado algunos valores discretos para autorizarle devolver una más amplia gama de tipos desde la función `env()`:

Tabla No. 3 Variables de entorno

.env Value	env() Value
true	(bool) true
(true)	(bool) true
false	(bool) false
(false)	(bool) false
empty	(string) "
(empty)	(string) "
null	(null) null
(null)	(null) null

Elaborado por: Xavier Bejarano

Si se necesita precisar una variable de entorno con un dato que contiene espacios, se puede hacer encerrando el valor entre comillas dobles:

```
APP_NAME="My Application"
```

2.3.6. Configuración de caché

Para aumentar la velocidad de la aplicación, se debe almacenar en caché todos los files de parametrización en un solo file empleando el comando `config:cache` Artisan. Realizando este procedimiento se podrá combinar todas las opciones de parametrización para la aplicación en un solo archivo que el marco puede cargar rápidamente. (Otwell, 2021)

Por lo general, se debe ejecutar el comando `php artisan config:cache` como parte del proceso de implementación de producción.

2.3.7. Directorio

Laravel está destinada a proporcionar un excelente punto de partida para aplicaciones grandes y pequeñas. Pero Laravel deja en libertad del desarrollador organizar la aplicación como fuese conveniente. Laravel casi no impone restricciones sobre dónde se

encuentra una clase determinada, siempre que Composer pueda cargar automáticamente la clase.

2.3.7.1. El directorio de aplicaciones

La mayor parte de la aplicación se encuentra en el directorio de aplicaciones. Por defecto, dicho directorio tiene un espacio de identificadores en el programa y Composer lo ejecuta automáticamente usando el estándar PSR-4. (Otwell, 2021)

El directorio de la aplicación recoge una variedad de adicionales directorios como Providers, Console y Http. Medite en los directorios Http y Console como si proporcionarán una API en la médula de su programa. Tanto la CLI como el protocolo HTTP son componentes para interactuar con el programa, pero en realidad no contienen el razonamiento del programa. Dicho de otra manera, son dos formas de formular comandos al programa. El directorio Console contiene todas las instrucciones de Artisan, mientras que el directorio Http recoge los controladores, el middleware y las solicitudes.

2.3.7.2. Directorio de configuración

Como su nombre lo indica, recoge la totalidad de los archivos de configuración del programa. Se recomienda leer todos estos archivos y habituarse con todas las disponibles opciones. (Otwell, 2021)

2.3.7.3. Directorio de la base de datos

Co lo que respecta a la base de datos, el directorio recoge las fábricas de modelos, las migraciones de la base de datos, y la sintaxis fuente. Es opcional, utilizar este directorio para almacenar una base de datos de tipo SQLite. (Otwell, 2021)

2.3.8. Despliegue

Para implementar la aplicación Laravel en producción, hay algunas cosas importantes que se pueden hacer para asegurarse de que la aplicación se ejecute de la manera más eficiente posible.

2.3.8.1. Requisitos del servidor

Laravel posee algunos requisitos para funcionar en el sistema. Se debe asegurar de que el servidor web tenga la siguiente versión y extensiones mínimas de PHP:

- PHP \geq 8.0
- BCMath PHP Extension
- Ctype PHP Extension
- cURL PHP Extension
- DOM PHP Extension
- Fileinfo PHP Extension
- JSON PHP Extension
- Mbstring PHP Extension
- OpenSSL PHP Extension
- PCRE PHP Extension
- PDO PHP Extension
- Tokenizer PHP Extension
- XML PHP Extension

2.3.8.2. Configuración del servidor con nginx

Nginx o motor x se define como un servidor proxy inverso y HTTP, un servidor proxy de correo y un servidor proxy TCP/UDP genérico, escrito originalmente por Igor Sysoev. Durante mucho tiempo, se ha estado ejecutando en muchos sitios robustos, incluidos

Yandex, Mail.Ru, VK y Rambler. Según Netcraft, nginx sirvió o actuó como proxy en el 22,16% de los sitios más concurridos en enero de 2022. Estas son algunas de las historias de éxito: Dropbox, Netflix, Wordpress.com, FastMail.FM. (F5 Networks, 2021)

En un servidor que ejecuta Nginx, se puede usar el siguiente archivo de configuración como punto de partida para configurar el servidor web. Lo más probable es que este archivo deba personalizarse según la configuración del servidor. Si se desea ayuda para administrar el servidor, se debe considerar usar un servicio de implementación y administración de servidor de Laravel propio, como Laravel Forge. (Otwell, 2021)

Se tiene que asegurar que el servidor web dirija todas las solicitudes al archivo public/index.php de la aplicación. Nunca se debe intentar mover el archivo index.php a la raíz del proyecto, ya que exponer la aplicación desde la raíz del proyecto pondrá en peligro muchos archivos de configuración confidenciales y los revelará ante Internet pública. Aquí un ejemplo de la configuración en Nginx:

```
server {
    listen 80;
    listen [::]:80;
    server_name example.com;
    root /srv/example.com/public;

    add_header X-Frame-Options "SAMEORIGIN";
    add_header X-Content-Type-Options "nosniff";

    index index.php;
```

2.4. MySQL



Gráfico No. 4 Logo MySQL

Tomado de: <https://www.gettyimages.es>

El software MySQL ofrece un servidor de base de datos SQL (Structured Query Language) muy rápido, multiproceso, multiusuario y sólido. MySQL Server está diseñado para sistemas de producción de carga pesada y de misión crítica, así como para integrarse en software de implementación masiva. (Oracle Corporation, 2021)

2.4.1. Características y ventajas de MySQL

Tabla No. 4 Características y ventajas MySQL

CARACTERÍSTICA	VENTAJA
Seguridad	Esgrime un sistema de privilegios muy seguro, el control se hace basado en host. MySQL utiliza seguridad basada en listas de control de acceso (ACL) para todas las conexiones, consultas y otras operaciones que los usuarios pueden intentar realizar. También hay soporte para conexiones encriptadas SSL entre clientes y servidores MySQL.
Escalabilidad y Límites	Manipula base de datos robustas. Su uso se amplía a más de 50 millones de registros. Se estima que algunos usuarios han usado el servidor MySQL con más de 60,000 tablas y cerca de 5 mil millones de filas. Son permitidos hasta 64 índices por tabla. Cada índice obtiene construir de 1 a 16 columnas.
Conectividad	Los clientes consiguen conectarse al servidor MySQL usando TCP/IP sobre cualquier plataforma. Aplicaciones cliente que utilicen Open Database Connectivity obtienen soporte del conector/ODBC (MyODBC). programas cliente java que usan JDBC obtienen soporte de la interface conector/J.

Elaborado por: Xavier Bejarano

2.4.2. Compatibilidad entre MySQL y Php

Tabla No. 8 Compatibilidad MySQL - PhP

CARACTERÍSTICAS	MYSQL	PHP
Dispone una amplia cantidad de documentación y ayuda en línea.	✓	✓
Velocidad para realizar las operaciones	✓	✓
Bajo costo para el desarrollo de proyectos	✓	✓
Facilidad de configuración e instalación	✓	✓

Fuente: (Oracle Corporation, 2021)

2.5. React JS



Gráfico No. 5 Logo React

Tomado de: <https://www.gettyimages.es>

ReactJS es una biblioteca de código abierto que se utiliza para crear interfaces de usuario, explícitamente para aplicaciones de una sola página. ReactJS permite a los ingenieros de software crear una aplicación web inmensa que puede usar datos y puede cambiar después de un tiempo sin recargar la página. En este sentido, React tiene un cálculo de diferenciación inteligente que utiliza para recuperar en su centro DOM lo que debería recuperarse mientras mantiene todo lo demás sin garantías. La utilización de piezas reutilizables proporciona un método sencillo para construir una aplicación. La brillante idea de React también hace que la configuración de la interfaz de usuario sea confiable y quita un gran peso de encima a los programadores para que puedan concentrarse en límites más grandes y razonamiento comercial. Brinda una rica variedad de bibliotecas entre las cuales los clientes pueden elegir realizar una tarea específica. (Rawat & Mahajan, 2020)

Las estrategias Lifecycle y React Hooks son otras características importantes que manejan la organización de los eventos que se llaman durante el ciclo de vida de un segmento.

2.5.1. Características y ventajas de React

Tabla No. 9 Características y ventajas React

CARACTERÍSTICA	VENTAJA
Librería clara y dinámica	React crea una interfaz de usuario inteligente y dinámica para sitios y aplicaciones portátiles. Además, crea perspectivas básicas para cada estado en la aplicación, React actualizará y entregará los datos de manera efectiva cuando cambie cualquier información. Las perspectivas definitivas hacen que el código sea más discernible y más fácil de depurar.
Curva de aprendizaje corta y fácil	La naturaleza directa y no compleja de ReactJS permite que se adapte rápidamente a la estructura. La comprensibilidad es probablemente la mejor cualidad de React. Es efectivamente comprensible incluso para las personas que son nuevas en él. Si bien los diferentes sistemas requieren aprender numerosas ideas sobre la estructura en sí, React sin duda hace que el programador intuya el camino. Por lo tanto, los niveles iniciales de habilidad en la estructura sin duda pueden refinarse sin impedimentos ni complejidades.
Enlace de datos unidireccional	ReactJS se presenta con un flujo de datos unidireccional entre los estados y las capas en una aplicación. Esto significa que los datos pueden fluir en una sola dirección entre los estados y las capas de la aplicación. Por otro lado, en el enlace de datos bidireccional como Angular, si se cambia un modelo, la vista también cambiará y viceversa. Las ventajas de la restricción de información de una sola dirección brindan un mejor control en toda la aplicación. Si el flujo de datos se dirige hacia otro camino, en ese punto, se requiere más trabajo. Es porque los componentes no deben cambiar y la información dentro de ellos no se puede cambiar. Flux en ReactJS proporciona un estilo comparable de restricción de información, que es una opción en contraste con el controlador de vista de modelo (MVC) normal.
JSX	JSX se puede considerar mejor como una estructura de lenguaje ampliada que se asemeja a HTML. Es básicamente similar a una mezcla de JavaScript y XML. JSX crea secciones de React, los cuadrados de estructura de las UI de React. Esto permite determinar los componentes DOM antes que los componentes directamente dentro de los registros de JavaScript. Lo que implica que la lógica detrás de las partes y las imágenes son generales.

Elaborado por: Xavier Bejarano

2.5.2. Frontend con React

React es una conocida biblioteca de JavaScript front-end de código abierto creada por Facebook. React tiene una amplia popularidad entre las comunidades de desarrolladores debido a su simplicidad y su proceso de desarrollo simple pero efectivo. Está instando a la formación de partes de interfaz de usuario reutilizables, que exhiben información que progresa después de un tiempo. Se actualiza rápidamente a través de la representación de las piezas valiosas para la lectura de cada estado y realiza los cambios de información dentro de la aplicación. (Rawat & Mahajan, 2020)

Llegó ReactJS y alteró la forma en que se crean las aplicaciones web. En ReactJS, cada componente se ocupa de su estado y los convierte en las IU. Con esta idea de componentes en lugar de diseños en JavaScript, se puede pasar mucha información sin mucho esfuerzo a la aplicación y, en este sentido, ayuda a mantener el estado fuera del DOM. En cualquier caso, React no es una estructura MVC como parece ser Angular. Se sirve como una biblioteca principal. Es decir, ReactJS excluye administradores de estado, conmutadores y directores de API en la biblioteca central. Puede parecer un impedimento para los ingenieros de React, pero esto es simplemente lo mejor para crear un sitio, ya que el código es simple con todas las partes y otras cosas. React llama la atención del usuario para la creación de interfaces pues los datos se obtienen del componente principal de una sección o están contenidos en la parte misma.

2.5.3. Configuración de React

React se puede instalar en varias plataformas como MAC OS, MS WINDOW o UNIX. Antes de instalar React, debe tener NODE y NPM preinstalados en su computadora. Las dos formas en que puede incluir reaccionar en su aplicación se mencionan a continuación:

- Instalación automática de ReactJS: El proceso programado debe utilizarse cuando necesitamos hacer otra aplicación en React:

```
- npm install -g create-react-app create-react-app my-app cd my-app npm start
```

- Instalación manual de ReactJS: el ciclo de establecimiento manual debe utilizarse cuando necesitemos utilizar React en nuestra aplicación actual. Requisitos: Webpack: es un paquete, con la ayuda de este podemos crear código medido y hace un montón de la multitud aparente de módulos, lo que amplía la exhibición de nuestra aplicación. Babel: es un compilador que ayuda a ejecutar el código JavaScript actual (como ES6) en el programa más experimentado. A raíz de la introducción de Webpack y Babel, necesitamos instalar el módulo react y react-dom: - npm install react react-dom --save Después de estas instalaciones, podemos crear componentes y renderizarlos también. (Rawat & Mahajan, 2020).

2.6. Tailwind CSS



Gráfico No. 6 Logo Tailwind

Tomado de: <https://www.gettyimages.es>

TailwindCSS es un framework de CSS que brinda al programador un desarrollo de código ágil que se pueda indexar con facilidad en el código HTML, también ayuda en los flujos de desarrollo que permiten optimizar mucho el peso y la organización del código CSS. (Roberto & Vidal, 2021)

En la actualidad existen diversas metodologías CSS que los desarrolladores web aplican como filosofía relacional entre CSS y HTML. A menudo, los desarrolladores deben aplicar correctamente estas metodologías para mantener la coherencia dentro del código base. En comparación con las metodologías, los marcos CSS brindan clases CSS preestablecidas a la base de código para ofrecer un diseño de interfaz de usuario consistente, pero, aun así, para avanzar y usar el diseño del marco, a menudo se necesita

ejecutar una metodología CSS consistente para manejar los desafíos comunes de CSS. El framework (Marco) Tailwind CSS, presenta su propia metodología de funcionamiento con clases de utilidad predefinidas. (Klimm, 2021).

2.6.1. Características y ventajas de Tailwind

Tabla No. 10 Características y ventajas Tailwind

CARACTERÍSTICA	VENTAJA
Personalización y trazabilidad	Las clases de utilidad en Tailwind CSS se generan en función de un archivo de configuración de JavaScript. Esto permite configurar propiedades específicas y consistentes. El uso de estas clases generadas es fácil de aprender debido a la buena documentación.
Consistencia	Tailwind CSS requiere que los desarrolladores usen las utilidades exactas para lograr los objetivos de estilo. A diferencia de otras metodologías, como BEM1, el desarrollador no puede desviarse de las especificaciones, ya que entonces no se percibiría ningún efecto. Con BEM, un desarrollador sin experiencia tiene más posibilidades de desviarse de las convenciones de nomenclatura. Esto obliga a los desarrolladores a lidiar con el aprendizaje de las convenciones CSS de Tailwind de manera regular, lo que conduce a conexiones más sólidas entre equipos dentro del sistema de diseño.
Curva de aprendizaje y experiencia altos	Tailwind CSS tiene un alto nivel de satisfacción para los desarrolladores, lo que promueve la actividad de las interconexiones entre equipos multidisciplinarios de desarrollo, así como también la motivación necesaria para mejorar el framework.

Elaborado por: Xavier Bejarano

2.7. Visual Studio Code



Gráfico No. 7 Logo Visual Studio Code

Tomado de: <https://www.gettyimages.es>

Visual Studio Code es un editor de código fuente ligero pero potente que se ejecuta en su escritorio y está disponible para Windows, macOS y Linux. Tiene agregado soporte para Node.js, JavaScript, TypeScript; ofrece también un amplio ecosistema de extensiones para tiempos de ejecución y otros lenguajes (como C++, C#, Java, Python, PHP, Go, .NET). (Microsoft, 2021)

VS Code viene con un diseño simple e intuitivo que maximiza el espacio provisto para el editor mientras deja un amplio espacio para navegar y acceder al contexto completo de su carpeta o proyecto. La pantalla de usuario contiene cinco aspectos importantes:

- Editor: es el segmento principal para editar sus files. Es posible abrir tantos editores como se necesite, uno junto al otro, horizontal y verticalmente.
- Barra lateral: contiene diferentes vistas, como el Explorador, para ayudarlo mientras trabaja en su proyecto.
- Barra de estado: información sobre el proyecto abierto y los archivos que edita.
- Barra de actividad: ubicada en el extremo izquierdo, le permite cambiar entre vistas y le brinda indicadores adicionales específicos del contexto, como la cantidad de cambios salientes cuando Git está habilitado.
- Paneles: es posible mostrar diferentes paneles por debajo de la sección del editor para obtener información de depuración o salida, advertencias y errores, o un integrado terminal. También, el panel tiene la capacidad de moverse hacia la derecha para lograr más espacio vertical.

Cada vez que inicia VS Code, se abre en el mismo estado en que se encontraba la última vez que se cerró. La carpeta, el diseño y los archivos abiertos se conservan.

CAPÍTULO III

3. ANÁLISIS Y DISEÑO

3.1. Recolección de la información

En el Centro Comercial Machado, ubicado en el cantón Riobamba, se ve la necesidad de crear una aplicación web que permita automatizar la gestión de servicios técnicos a domicilio he inventario stock de insumos y materiales; ya que actualmente dicho procedimiento se lo está registrando en papel, lo que ha generado pérdida de datos, inconsistencias en inventario y demás inconvenientes que perjudican el normal desarrollo de los servicios técnicos a domicilio que realiza la empresa.

La entrevista con el Ing. Fabián Machado, encargado del área de TIC's del Centro Comercial Machado, confirma lo expuesto anteriormente y da como resultado la necesidad de implementar una aplicación web que permita automatizar la gestión de servicios técnicos a domicilio he inventario stock de insumos y materiales. De la misma manera, en coordinación con el Ing. Fabián Machado, se realiza un análisis de los requerimientos para empezar con el proceso de creación de la aplicación web.

3.1.1. Alcance

En el Centro Comercial Machado se brindan servicios técnicos a domicilio y se utiliza un inventario stock de insumos y materiales por lo que el presente proyecto cubrirá los requerimientos que conciernen al proceso de gestión de dichos servicios e inventario.

La aplicación web se implementará en un equipo suministrado por el Centro Comercial Machado, el equipo operará como hosting y estará enlazado a la red interna de la empresa.

Las opciones básicas que tendrá la aplicación web serán:

- Acceso al sistema o Login.
- Registro de datos de los usuarios: Administradores, Técnicos y Clientes.
- Registro de órdenes de servicio.
- Registro de inventario stock de insumos y materiales.

- Reportes.

3.1.2. Análisis

Para el desarrollo de este proyecto a continuación se detalla la importancia que tendrá la implementación de una app web para el manejo de servicios técnicos a domicilio y control de inventario stock de insumos y materiales, utilizando el Framework PHP Laravel y gestor de base de datos MySQL, en el centro comercial Machado, localizado en la ciudad de Riobamba en el periodo 2021; el mismo que presenta como antecedentes pérdida de documentación y consumo excesivo de tiempo en la gestión del documental manual de los servicios técnicos y control de inventario stock, se desglosa a continuación un análisis de las funciones principales de la aplicación web:

- Ingreso al sistema mediante usuario y contraseña que serán validados para dar su acceso.
- Almacenamiento en una base de datos donde será resguardada toda la información.
- Impresión de reportes.
- Ingreso de usuarios, inventario y ordenes de servicio.

3.1.3. Estudio de factibilidad

El presente Estudio de Factibilidad consiste en comprobar la viabilidad y rentabilidad de la implementación de una app web para el manejo de servicios técnicos a domicilio y control de inventario stock de insumos y materiales, utilizando el Framework PHP Laravel y gestor de base de datos MySQL, en el centro comercial Machado, ubicado en la ciudad de Riobamba, ofreciendo un servicio diferenciador enfocado a disminuir la dificultad de la empresa para gestionar sus servicios técnicos a domicilio. Para la realización de este estudio fue necesario hacer una investigación de campo, observando el entorno para conocer las necesidades y dificultades identificadas, proyectando las siguientes mejores:

- Paso de un sistema manual a un sistema automático de gestión.
- Optimización de tiempo en procesos.
- Almacenamiento de datos de forma segura.
- Mejora en el manejo de inventario stock de insumos y materiales.

3.1.3.1. Factibilidad operativa

Para valorar la factibilidad operativa del proyecto es necesario tener en cuenta la administración del aplicativo como una operación central y permanente de todo el proceso, su manejo adecuado es un factor fundamental para el funcionamiento exitoso de la aplicación. En este sentido es necesario el dominio de sistemas y el manejo del aplicativo como tal por parte de administradores y técnicos, consiguientemente es preciso dar a conocer la nueva modalidad de gestión mediante a la aplicación web a los clientes del Centro Comercial Machado; para ello se requiere una pequeña descripción o leyenda en las ordenes de servicio con las instrucciones o pasos que deberá seguir el cliente para dar seguimiento a su servicio.

El desarrollo de la aplicación en cuanto a operatividad es factible debido a que puede ser manejado por el personal administrativo - técnico y clientes del Centro Comercial Machado, lo cual representa una ventaja para su desarrollo y manejo.

3.1.3.2. Factibilidad técnica

Requerimientos en Hardware:

- Laptop: Intel Core i5-4210U 1.70GHz (c/TB 2.70 GHz)
- Pantalla LED 15.6" HD (1600x900)
- 4GB RAM / Disco Duro 500GB
- DVD+RW,

Requerimientos en Software:

- Windows 8.1 o superior
- PhP 7.1
- Laravel 8
- MySQL 8
- Xampp 8.1
- Visual Studio Code 1.71

La factibilidad técnica es favorable para la implementación de la app web para el manejo de servicios técnicos a domicilio, teniendo en cuenta que se cuenta con todos los requerimientos en software y hardware.

3.1.3.3. Factibilidad económica

- **Online**

Económicamente el proyecto es factible ya que el proponente utilizará tecnologías bajo licencia Open Source (Código Abierto), lo que permitirá costos mínimos tanto en el desarrollo como en la implementación de la app web; dichos valores junto con la estructura de hardware requerida serán cubiertos por el proponente y auspiciante del proyecto.

3.1.3.4. Factibilidad legal

El presente proyecto es factible legalmente ya que las tecnologías usadas están amparadas bajo la categorización Open Source. Las licencias de código abierto son autorizaciones que plasman la definición de código abierto; en síntesis, autorizan que el software se utilice, comparta y modifique libremente. Una licencia tiene que pasar por el proceso de verificación de licencias de la Iniciativa de Código Abierto, para ser admitida por la Iniciativa de Código Abierto u OSI.

3.1.4. Análisis de los requerimientos

3.1.4.1. Requerimientos funcionales del sistema

Los parámetros de requerimientos entregados para la creación del sistema se definieron por el Ing. Fabián Machado, encargado del departamento de TIC de la empresa; estableciéndose los siguientes:

- Creación, Modificación y Eliminación de Usuarios.
- Creación, Modificación y Eliminación de Inventario Stock de Insumos y Materiales.
- Creación, Modificación y Eliminación de Ordenes de Servicio.
- Reportes: Ordenes de Ingreso o Reparación, Ordenes de Salida, Reportes de Servicios.

3.1.4.2. Requerimientos no funcionales del sistema

- **Disponibilidad:** El aplicativo web estará disponible las 24 horas del día los 365 días del año, al 100% de su funcionalidad.
- **Escalabilidad:** El aplicativo web podrá ser modificado para futuras actualizaciones y adiciones de módulos adicional.
- **Seguridad:** Los datos del aplicativo web estarán cifrados y almacenados bajo las parametrizaciones de seguridad de MySQL.
- **Eficiencia:** La app web para el manejo de servicios técnicos a domicilio e inventario stock de insumos y materiales, superará en todos los aspectos al procedimiento manual que actualmente tiene el Centro Comercial Machado.
- **Funcionalidad:** El aplicativo web cumplirá con todos los parámetros de funcionalidad pedidos por el Ing. Fabián Machado.

3.1.5. Definición de casos de uso

Actores:

- **Administradores:** usuario con control total del aplicativo web.
- **Técnicos:** usuario con control parcial en base a roles.
- **Cliente:** usuario con acceso estrictamente informativo.

Caso de uso para ingreso al sistema:

- **Objetivo:** Permitir la validación de los diferentes usuarios.

Flujo Principal

1. Ingreso al Sistema

2. Presenta Menú

Flujo Secundario

- 1.1 Ingreso de Credenciales
- 1.2 Consulta en Base de Datos
- 1.3 Entrega Resultados

- 2.1 Acceso Permitido
- 2.2 Acceso Negado

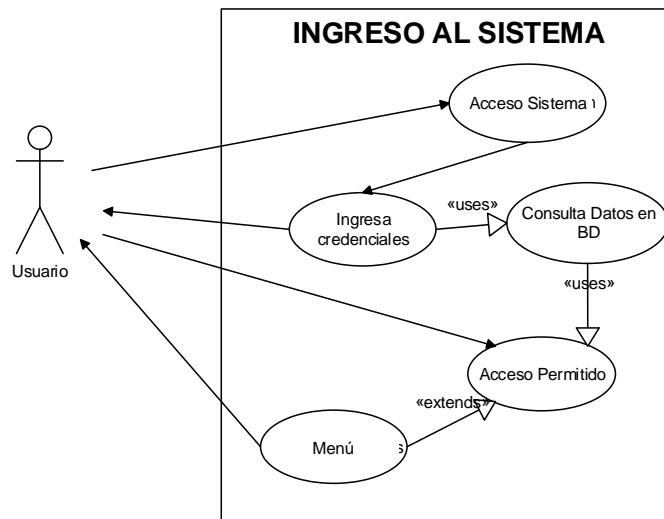


Gráfico No. 8 Ingreso al sistema

Elaborado por: Xavier Bejarano

Caso de uso para ingreso de nuevo usuario

- **Objetivo:** Crear y admitir a nuevos usuarios el ingreso al sistema

Flujo Principal

1. Ingreso nuevo usuario

2. Presenta Página Principal

Flujo Secundario

1.1 Ingreso de Información
1.2 Validar Datos
1.3 Almacenar en Base de datos

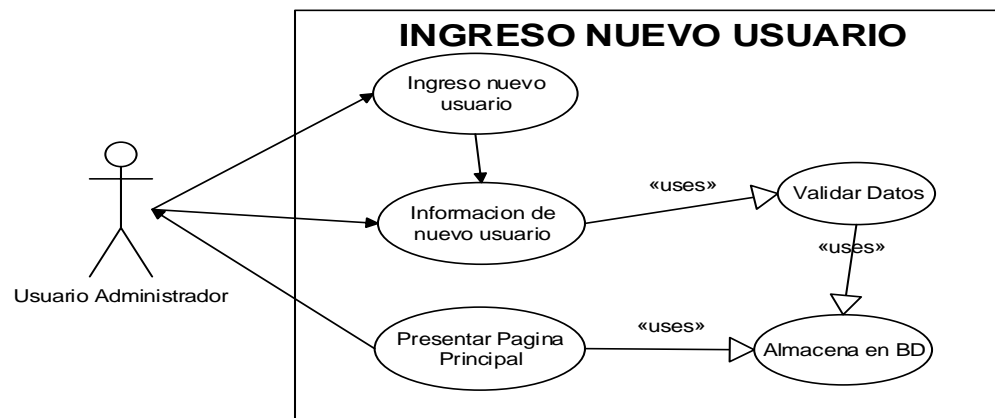


Gráfico No. 9 Nuevo usuario

Elaborado por: Xavier Bejarano

Caso de uso para ingreso de inventario stock de insumos y materiales

- **Objetivo:** Realizar el ingreso de inventario stock de insumos y materiales.

Flujo Principal

1. Ingreso inventario

2. Presenta Página Inventario

Flujo Secundario

1.1 Ingreso de Información
1.2 Validar Datos
1.3 Almacenar en Base de datos

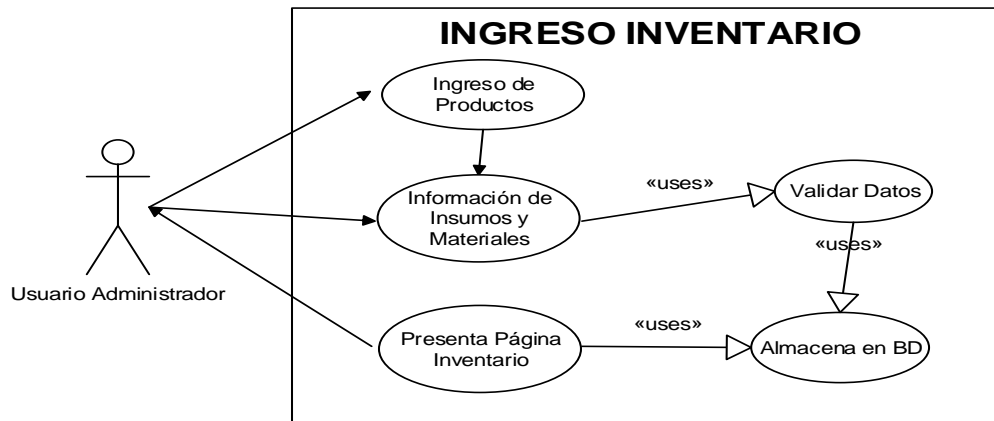


Gráfico No. 10 Inventario

Elaborado por: Xavier Bejarano

Caso de uso para ingreso de orden de servicio técnico

- **Objetivo:** Realizar el ingreso de orden de servicio técnico.

Flujo Principal

3. Ingreso de orden de servicio técnico

Flujo Secundario

- 3.1 Ingreso de Información
- 3.2 Validar Datos
- 3.3 Almacenar en Base de datos

4. Presenta Página Servicios

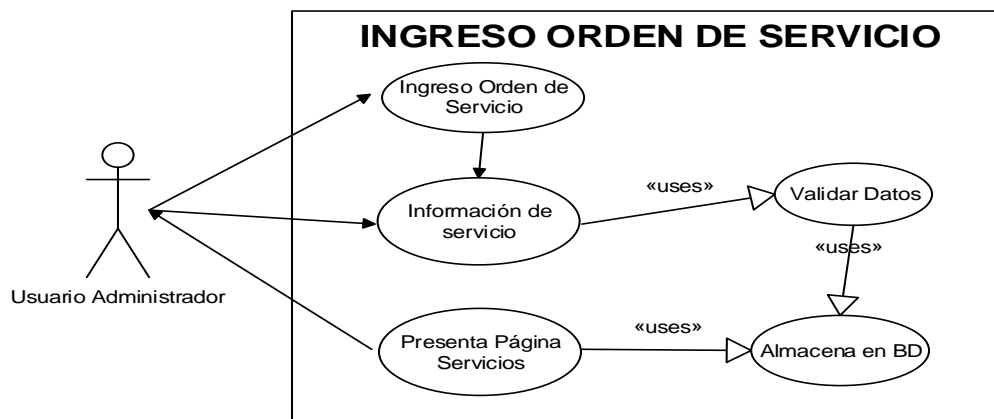


Gráfico No. 11 Orden de Servicio

Elaborado por: Xavier Bejarano

Caso de Uso de Reportes - Ordenes de Ingreso

- **Objetivo:** Permite al usuario la impresión de reportes de Ordenes de Ingreso

Flujo Principal

1. Selección de Opción

Flujo Secundario

- 1.1 Verificar Solicitud
 - 1.2 Verifica en BD
 - 1.3 Genera documento Solicitado
2. Imprime

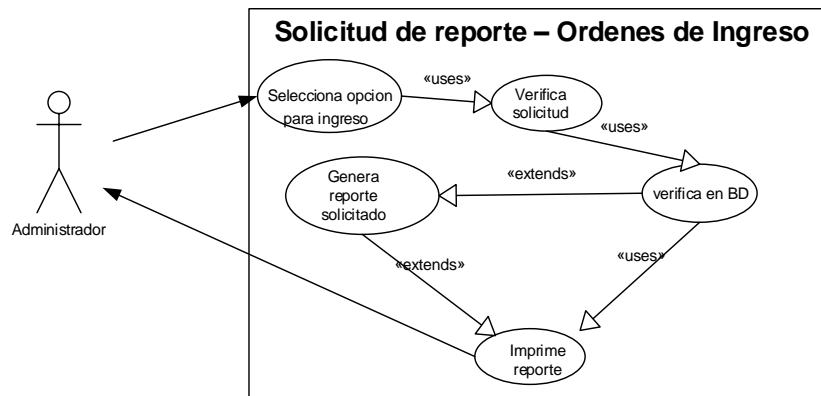


Gráfico No. 12 Ordenes de Ingreso

Elaborado por: Xavier Bejarano

Caso de Uso de Reportes - Ordenes de Salida

- **Objetivo:** Permite al usuario la impresión de reportes de Ordenes de Salida

Flujo Principal

3. Selección de Opción

Flujo Secundario

- 3.1 Verificar Solicitud
 - 3.2 Verifica en BD
 - 3.3 Genera documento Solicitado
4. Imprime

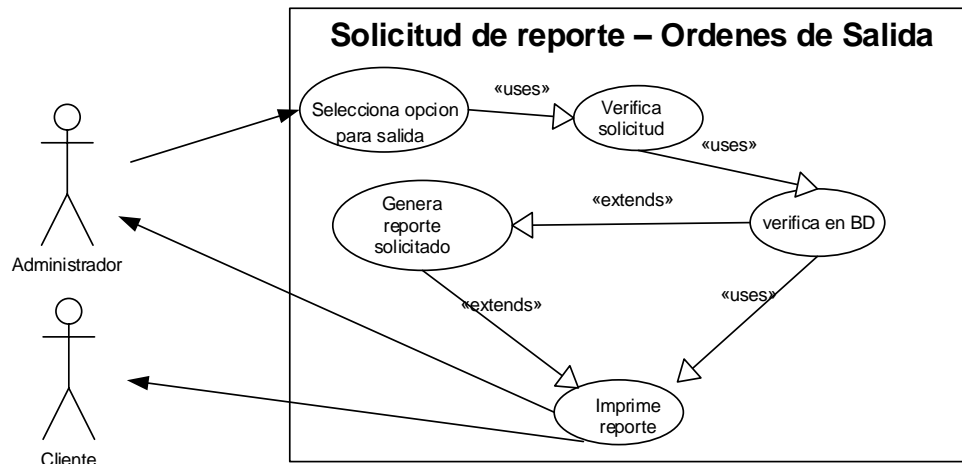


Gráfico No. 13 Ordenes de Salida

Elaborado por: Xavier Bejarano

Caso de Uso de Reportes - Servicios

- **Objetivo:** Permite al usuario la impresión de reportes de Servicios

Flujo Principal

Flujo Secundario

5. Selección de Opción

5.1 Verificar Solicitud

5.2 Verifica en BD

5.3 Genera documento Solicitado

6. Imprime

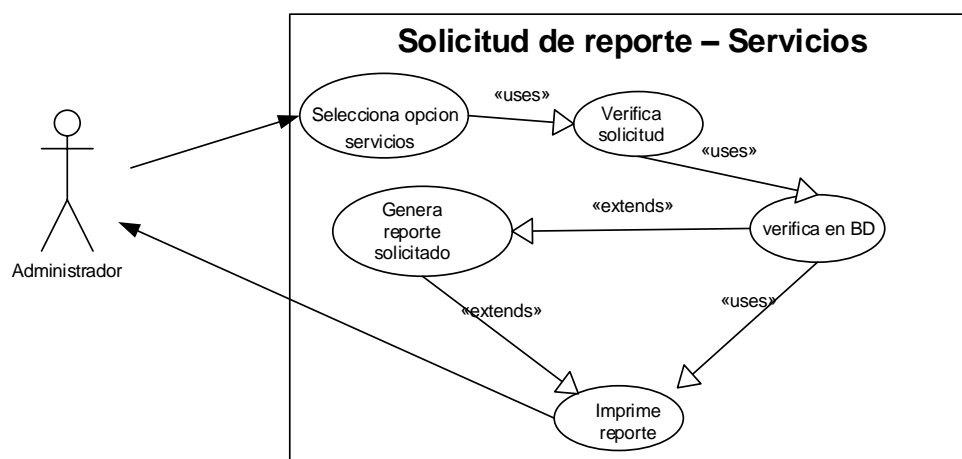


Gráfico No. 14 Reporte Servicios

Elaborado por: Xavier Bejarano

3.2. Diseño

3.2.1. Diseño conceptual

Diagrama de componentes:

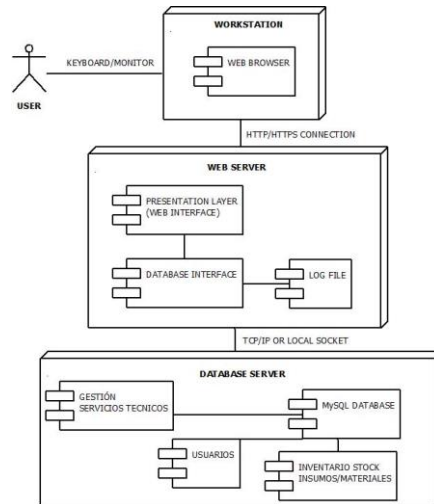


Gráfico No. 15 Diseño Conceptual

Elaborado por: Xavier Bejarano

3.2.2. Modelo relacional

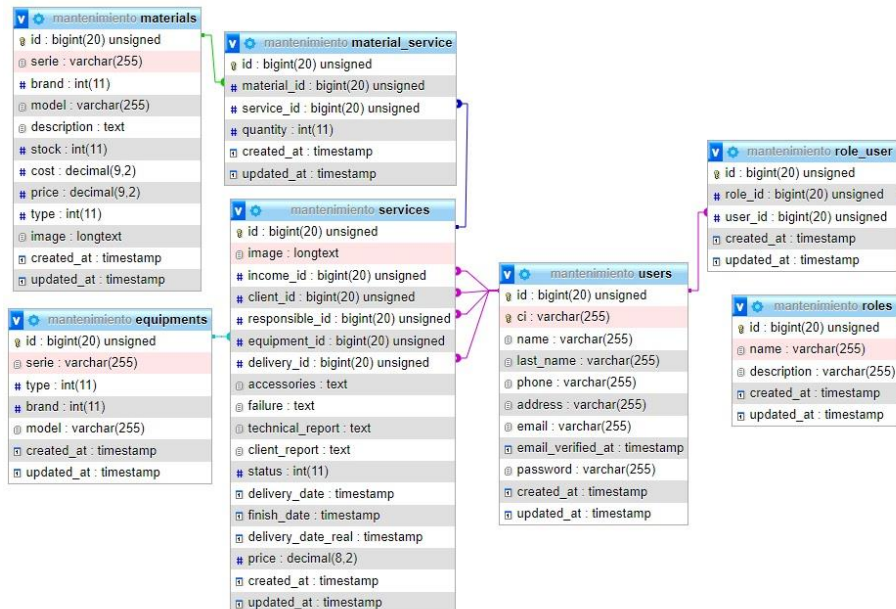


Gráfico No. 16 Modelo Relacional

Elaborado por: Xavier Bejarano

3.2.3. Diccionario de datos

equipments

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Media type
id (<i>Primaria</i>)	bigint(20)	No				
serie	varchar(255)	No				
type	int(11)	No				
brand	int(11)	No				
model	varchar(255)	No				
created_at	timestamp	Sí	NULL			
updated_at	timestamp	Sí	NULL			

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	id	0	A	No	

materials

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Media type
id (<i>Primaria</i>)	bigint(20)	No				
serie	varchar(255)	No				
brand	int(11)	No				
model	varchar(255)	No				
description	text	Sí	NULL			
stock	int(11)	No				
cost	decimal(9,2)	No				
price	decimal(9,2)	No				
type	int(11)	No				
image	longtext	Sí	NULL			
created_at	timestamp	Sí	NULL			
updated_at	timestamp	Sí	NULL			

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	id	0	A	No	

material_service

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Media type
id (<i>Primaria</i>)	bigint(20)	No				
material_id	bigint(20)	No		materials -> id		
service_id	bigint(20)	No		services -> id		
quantity	int(11)	No				
created_at	timestamp	Sí	NULL			
updated_at	timestamp	Sí	NULL			

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	id	0	A	No	
material_service_material_id_foreign	BTREE	No	No	material_id	0	A	No	
material_service_service_id_foreign	BTREE	No	No	service_id	0	A	No	

migrations

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Media type
id (<i>Primaria</i>)	int(10)	No				
migration	varchar(255)	No				
batch	int(11)	No				

Gráfico No. 17 Diccionario de datos 1

Elaborado por: Xavier Bejarano

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	id	8	A	No	

personal_access_tokens

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Media type
id (<i>Primaria</i>)	bigint(20)	No				
tokenable_type	varchar(255)	No				
tokenable_id	bigint(20)	No				
name	varchar(255)	No				
token	varchar(64)	No				
abilities	text	Sí	NULL			
last_used_at	timestamp	Sí	NULL			
created_at	timestamp	Sí	NULL			
updated_at	timestamp	Sí	NULL			

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	id	0	A	No	
personal_access_tokens_token_unique	BTREE	Sí	No	token	0	A	No	
personal_access_tokens_tokenable_type_tokenable_id_index	BTREE	No	No	tokenable_type	0	A	No	
				tokenable_id	0	A	No	

roles

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Media type
id (<i>Primaria</i>)	bigint(20)	No				
name	varchar(255)	No				
description	varchar(255)	Sí	NULL			
created_at	timestamp	Sí	NULL			
updated_at	timestamp	Sí	NULL			

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	id	3	A	No	

role_user

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Media type
id (<i>Primaria</i>)	bigint(20)	No				
role_id	bigint(20)	No		roles -> id		
user_id	bigint(20)	No		users -> id		
created_at	timestamp	Sí	NULL			
updated_at	timestamp	Sí	NULL			

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	id	0	A	No	
role_user_role_id_foreign	BTREE	No	No	role_id	0	A	No	
role_user_user_id_foreign	BTREE	No	No	user_id	0	A	No	

services

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Media type
id (<i>Primaria</i>)	bigint(20)	No				
image	longtext	Sí	NULL			
income_id	bigint(20)	No		users -> id		
client_id	bigint(20)	No		users -> id		
responsible_id	bigint(20)	No		users -> id		
equipment_id	bigint(20)	No		equipments -> id		

Gráfico No. 18 Diccionario de datos 2

Elaborado por: Xavier Bejarano

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	id	0	A	No	
services_income_id_foreign	BTREE	No	No	income_id	0	A	No	
services_client_id_foreign	BTREE	No	No	client_id	0	A	No	
services_responsible_id_foreign	BTREE	No	No	responsible_id	0	A	No	
services_equipment_id_foreign	BTREE	No	No	equipment_id	0	A	No	
services_delivery_id_foreign	BTREE	No	No	delivery_id	0	A	Sí	

users

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Media type
id (<i>Primaria</i>)	bigint(20)	No				
ci	varchar(255)	No				
name	varchar(255)	Sí	NULL			
last_name	varchar(255)	Sí	NULL			
phone	varchar(255)	Sí	NULL			
address	varchar(255)	Sí	NULL			
email	varchar(255)	Sí	NULL			
email_verified_at	timestamp	Sí	NULL			
password	varchar(255)	No				
created_at	timestamp	Sí	NULL			
updated_at	timestamp	Sí	NULL			

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	id	0	A	No	
users_ci_unique	BTREE	Sí	No	ci	0	A	No	

Gráfico No. 19 Diccionario de datos 3

Elaborado por: Xavier Bejarano

3.2.4. Diseño de interfaces

Acceso al sistema

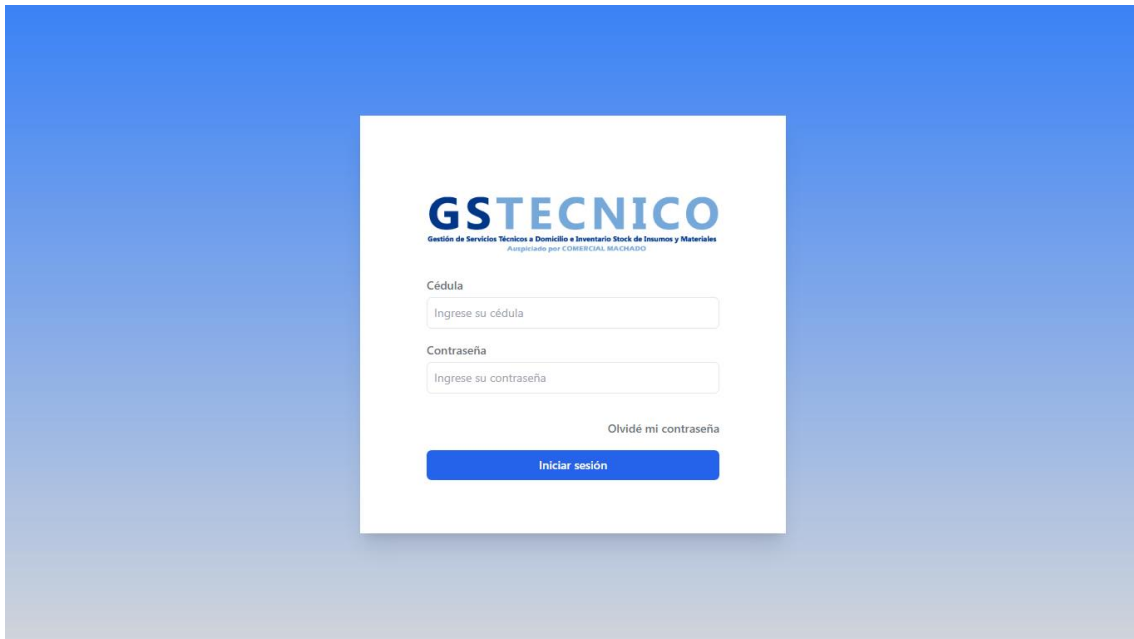


Gráfico No. 20 Interfaz Login

Elaborado por: Xavier Bejarano

Dashboard Principal

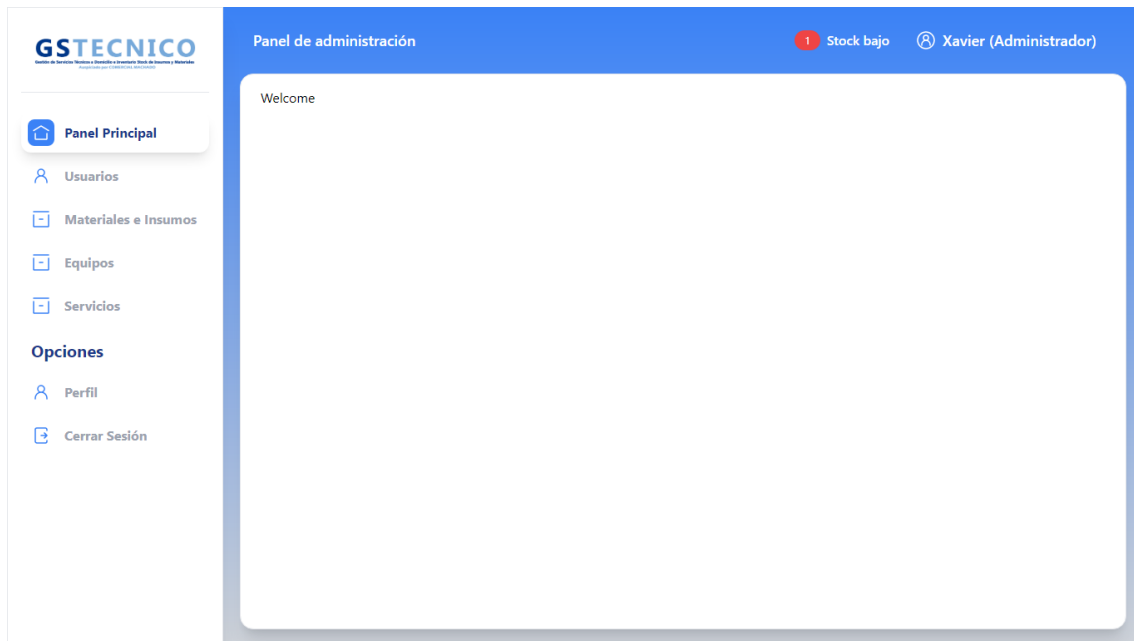


Gráfico No. 21 Interfaz Dashboard
Elaborado por: Xavier Bejarano

Usuarios

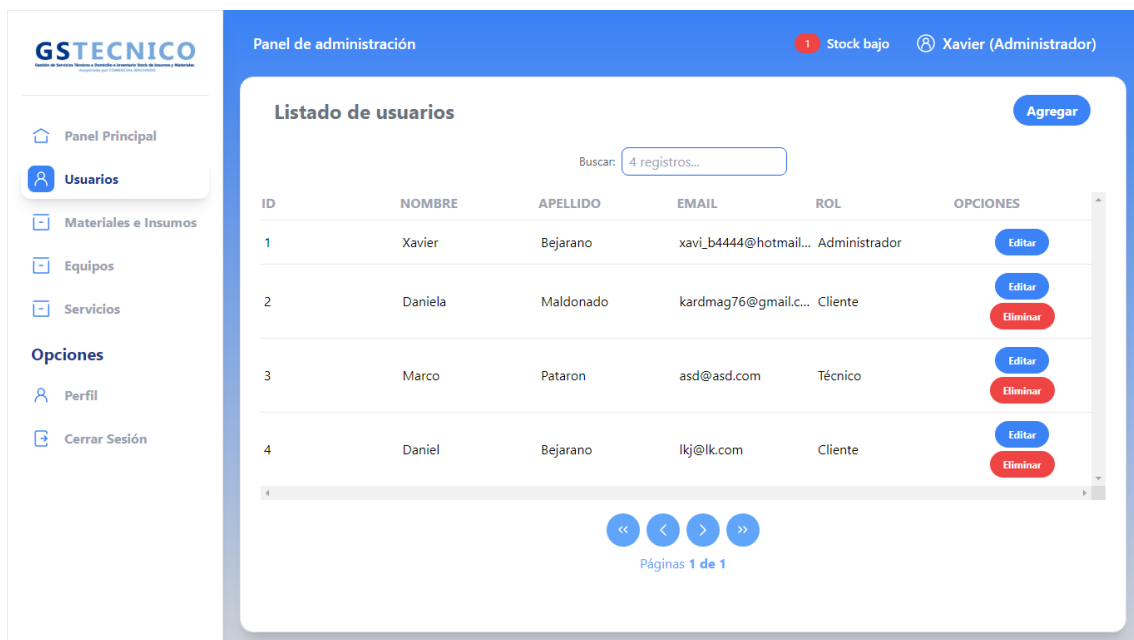


Gráfico No. 22 Interfaz Usuarios
Elaborado por: Xavier Bejarano

Inventario

Panel de administración 1 Stock bajo Xavier (Administrador)

Listado de Materiales e Insumos

Buscar: 1 registros...

ID	SERIE	MARCA	MODELO	DESCRIPCIÓN TIPO	STOCK	PRECIO	OPCIONES
1	056465	Western Digi...	wd500	Disco Mecán... Repuesto	4	\$80.56	Editar Eliminar

Páginas 1 de 1

Gráfico No. 23 Interfaz Inventario

Elaborado por: Xavier Bejarano

Servicios

Panel de administración 1 Stock bajo Xavier (Administrador)

Listado de servicios

Buscar: 3 registros...

ID	FECHA DE INGRESO	SERIE DE EQUIPO	CLIENTE	ESTADO	RESPONSABLE	OPCIONES
3	2022-09-09	DCN15T	Daniel Bejarano	Trabajo terminado	Marco	Editar Informe Comprobante Eliminar
2	2022-09-09	KWRHMT	Daniela Maldona...	Sin Revisar	Xavier	Editar Informe Comprobante Eliminar
1	2022-09-09	53PHSF	Daniela Maldona...	Trabajo terminado	Marco	Editar Informe Comprobante Eliminar

Gráfico No. 24 Interfaz Servicios

Elaborado por: Xavier Bejarano

CAPÍTULO IV

4. IMPLEMENTACIÓN DE LA APLICACIÓN WEB

4.1. Configuración de las herramientas de desarrollo

Se demandan servicios web para desplegar el entorno base del aplicativo, en este sentido se procederá a instalar de XAMPP

1. Se descarga el instalador desde la web:

<https://www.apachefriends.org/es/download.html>

2. Se ejecuta el instalador:

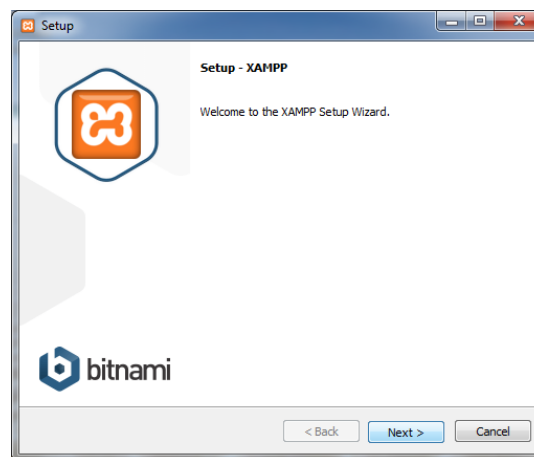


Gráfico No. 25 Instalación Xampp

Elaborado por: Xavier Bejarano

3. Se configura el entorno de Xampp:

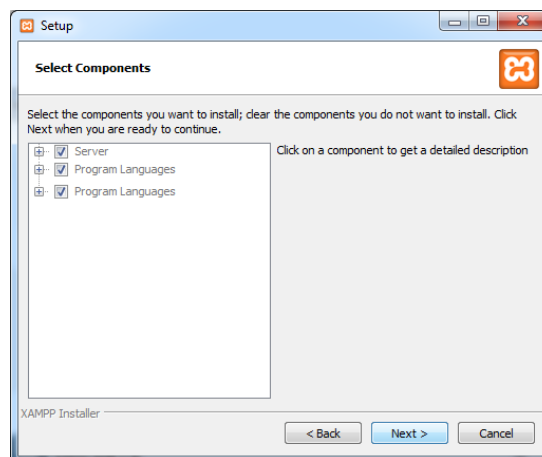


Gráfico No. 26 Entorno Xampp

Elaborado por: Xavier Bejarano

4. A continuación, se muestra la pantalla de confirmación:

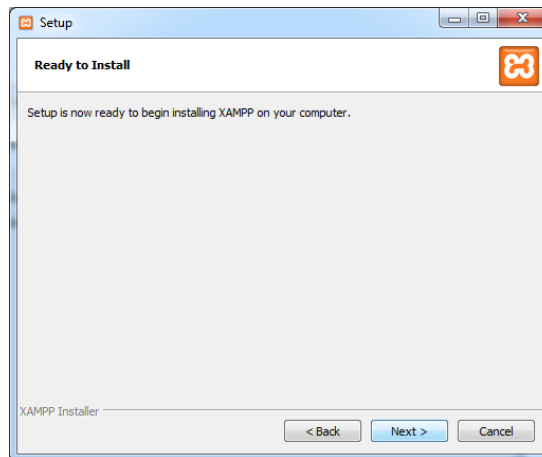


Gráfico No. 27 Confirmación Xampp

Elaborado por: Xavier Bejarano

5. Culminada la instalación se puede inicializar los servicios web.

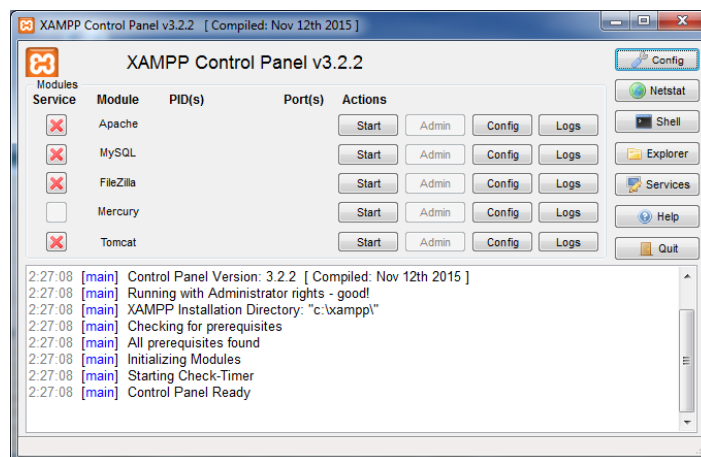


Gráfico No. 28 Panel de control Xampp

Elaborado por: Xavier Bejarano

4.2. Arquitectura del sistema

El modelo aplicado para el desarrollo del aplicativo web es la arquitectura cliente-servidor, esta arquitectura es una estructura de aplicación distribuida que divide tareas o cargas de trabajo entre los proveedores de un servicio o recurso, denominados equipos servidores, y los peticionarios del servicio, denominados equipos clientes. Por lo general, los servidores y los clientes interactúan a través de una red independiente informática de hardware, pero tanto el servidor como el cliente pueden habitar en el mismo sistema. Un

solo host de equipo servidor puede ejecutar uno o más aplicaciones de servidor, que cooperan sus recursos con los clientes. Un solo cliente usualmente no comparte ninguno de sus recursos, más puede solicitar servicio o contenido de un servidor. Los equipos clientes, en consecuencia, inicializan sesiones de interacción con los equipos servidores, que aguardan las solicitudes entrantes.

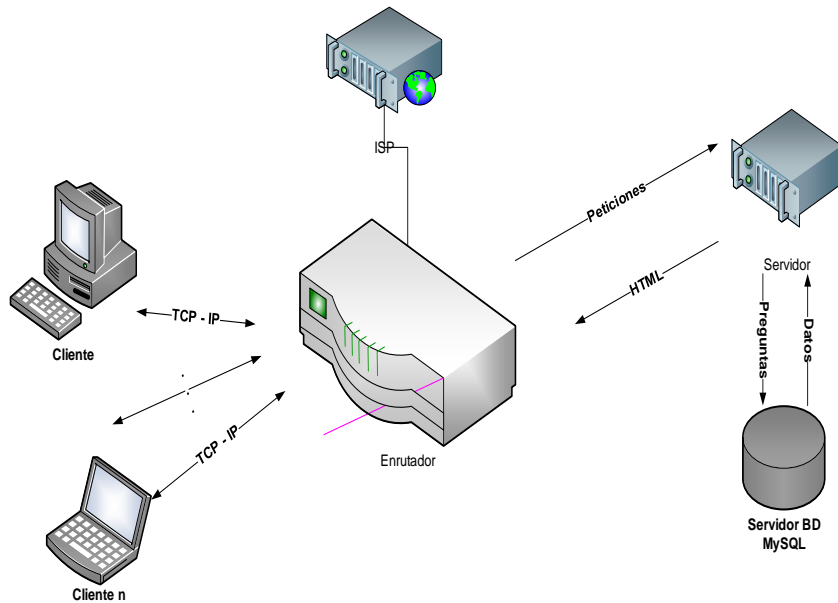


Gráfico No. 29 Arquitectura de aplicación

Elaborado por: Xavier Bejarano

El aplicativo web de gestión de servicios técnicos a domicilio e inventario stock de insumos y materiales reside en el servidor, los clientes mediante la infraestructura de red acceden al servidor para realizar consultas o requerimientos.

4.3. Implementación del aplicativo web

4.3.1. Definición de módulos

Los módulos que conforman la aplicación web se encuentran en varios componentes con relación y navegabilidad directa.

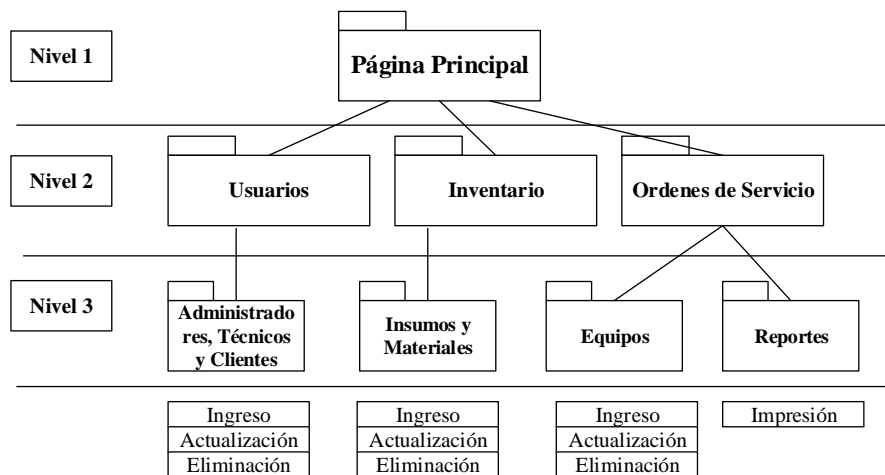


Gráfico No. 30 Módulos de aplicación

Elaborado por: Xavier Bejarano

4.3.2. Implementación de la aplicación

Se desarrolla el aplicativo y se realiza la implementación.

4.4. Pruebas de aplicación

Para evidenciar la funcionalidad del sistema se ejecutaron las siguientes pruebas:

Tabla No. 11 Pruebas de aplicación

PRUEBAS	SI	NO
Presenta página informativa	✓	
Ingresar, modificar y eliminar usuarios	✓	
Ingresar, modificar y eliminar inventario	✓	
Ingresar, modificar y eliminar ordenes de servicio	✓	
Permitir acceso a usuarios	✓	
Funcionalidad		
Tabletas	✓	
Navegadores Chrome, Firefox y Edge	✓	
Ingreso sin errores a la BD	✓	
Conexión de Base de Datos	✓	
Validación de cedulas	✓	
Generación de Solicitud de Datos	✓	
Reducción de tiempo al generar reportes	✓	

Elaborado por: Xavier Bejarano

4.5. Capacitación al personal

- **Personal**

Impartidor: Xavier Bejarano, desarrollador de la aplicación

Destinatarios: Personal administrativo y técnico del Centro Comercial Machado

- **Equipos y herramientas tecnológicas**

Los equipos y herramientas tecnológicas necesarias para la ejecución de la capacitación fueron:

Equipos electrónicos

- Computador
- Proyector de pantalla

Herramientas tecnológicas

- Prezi (Presentación)
- Chrome
- Internet

- **Tiempo**

Tiempo estimado: 1 mes

4.6. Mantenimiento de la aplicación

Es recomendable realizar un mantenimiento semestral del aplicativo web con la finalidad de verificar las últimas actualizaciones de las tecnologías usadas en la realización del proyecto; también se deben verificar los diferentes procesos internos y optimizar la base de datos.

El diseño, desarrollo e implementación del aplicativo web está sujeta a los requerimientos del usuario, el mantenimiento es responsabilidad exclusiva del auspiciante del proyecto; así como también, las modificaciones y adiciones de nuevos módulos en el aplicativo web.

CONCLUSIONES

- El framework Laravel es una de las tecnologías más robustas y dinámicas que existen para la creación de aplicaciones web, su licencia open source facilita el manejo del framework sin ninguna restricción; además que es compatible con el lenguaje de programación PHP y el gestor de base de datos MYSQL. Todas estas características hicieron que el presente proyecto sea escalable y progresivo, condiciones indispensables para una aplicación web profesional.
- Planificar un análisis de requerimientos adecuado, mediante entrevistas con administradores y contacto directo con los diferentes usuarios involucrados, fue uno de los pilares fundamentales sobre el que se cimiento el desarrollo del presente proyecto, dando como resultado una aplicación web a la altura de las necesidades del Centro Comercial Machado.
- El desarrollo e implementación de una aplicación web permitió, al Centro Comercial Machado, optimizar los diferentes procesos implicados al momento de gestionar un servicio técnico a domicilio e inventario stock de insumos y materiales, ya que los administradores y técnicos pueden fácilmente llevar un registro confiable de todos sus servicios e inventario.

RECOMENDACIONES

- Se recomienda a los estudiantes del Instituto Superior Tecnológico “San Gabriel”, se capaciten constantemente en las diferentes tecnologías para desarrollo de aplicaciones web; ya que las mismas están en continua actualización.
- Se recomienda al Centro Comercial Machado, realizar un seguimiento al uso del aplicativo web GSTECNICO, con la finalidad de mejorar y optimizar los requerimientos funcionales en una segunda versión del aplicativo.
- Se recomienda al Instituto Superior Tecnológico “San Gabriel”, continuar impulsando este tipo de proyectos en el área específica de desarrollo de aplicaciones web, ya que en la actualidad se ha podido observar gran demanda de este tipo de tecnología.

BIBLIOGRAFÍA

Blog Neosoft Sistemas. 2020. “¿Qué Es Una Aplicación Web? – Blog Neosoft Sistemas.”

Retrieved June 12, 2020 (<https://www.neosoft.es/blog/que-es-una-aplicacion-web/>).

Codedimension. (20 de Noviembre de 2020). *¿Qué es y para que sirve un sitio web?*

Obtenido de <https://www.codedimension.com.ar/noticias-sobre-tecnologia/noticias/que-es-y-para-que-sirve-un-sitio-web/1>

Cowburn, P. (08 de Enero de 2019). *Manual de PHP*. Obtenido de PHP Group:

<https://www.php.net/manual/es/index.php>

Cuesta, M. (11 de Marzo de 2020). *Trabajando con objetos*. Obtenido de

https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Trabajando_con_objetos

F5 Networks. (01 de 01 de 2021). *NGINX*. Obtenido de <https://nginx.org/>

ICTEA. 2020. “¿Qué Es Una Aplicación Web? - Base de Conocimientos - ICTEA.” 1.

IONOS España S.L.U. 2018. “Web App y sus tipos - IONOS.”

Retrieved June 12, 2020 (<https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/que-es-una-web-app-y-que-clases-hay/>).

Klimm, M. C. (01 de 01 de 2021). *Technology Arts Sciences TH Köln*. Obtenido de

<https://epb.bibl.th-koeln.de/frontdoor/index/index/docId/1666>

Lashutko, Y., & Koziel, B. (Junio de 2019). *Universidad Complutense de Madrid*.

Obtenido de https://eprints.ucm.es/id/eprint/56604/1/1138365865-461700_YEVHENII_LASHUTKO_181-MyBulletin_3940146_547473299.pdf

Lopez, Miguel. 2018. “Definición de Aplicación Web - Miguel Angel López WEB.” 16.

Retrieved June 12, 2020 (<http://mialtoweb.es/definicion-de-aplicacion-web/>).

López, D. A. (2020). *Programación Orientada a Objetos I*. Ávila: Publicaciones de la Universidad Católica de Ávila.

Microsoft (2021). Visual Studio Code. Obtenido de VS Code:

<https://code.visualstudio.com/>

Oracle Corporation. (01 de 01 de 2021). *MySQL*. Obtenido de <https://www.mysql.com/>

Otwell, T. (01 de 01 de 2021). *Laravel*. Obtenido de Laravel LLC:

<https://laravel.com/docs>

Perez, M. (15 de Marzo de 2019). *ventajas y desventajas de MySQL, oracle, visual foxpro y access*. Obtenido de <https://mape309site.wordpress.com/2017/11/15/ventajas-y-desventajas-de-mysql-oracle-visual-foxpro-y-access/>

Rawat, P., & Mahajan, A. N. (2020). ReactJS: A Modern Web Development Framework. *Internationalist Journalist of Innovative Research, Science and Technology*, 1-5.

Roberto, J., & Vidal, G. (01 de 01 de 2021). *Universidad Politécnica de Valencia*. Obtenido de <https://riunet.upv.es/handle/10251/172200#>

ANEXOS

ANEXO 1.

MANUAL DEL USUARIO

MANUAL DEL USUARIO

GSTECNICO

Gestión de Servicios Técnicos a Domicilio e Inventario Stock de Insumos y Materiales
Auspiciado por COMERCIAL MACHADO

APLICATIVO WEB

CENTRO COMERCIAL MACHADO

GSTECNICO

VERSIÓN 1.0

REALIZADO POR:

XAVIER PATRICIO BEJARANO GAVILANES

Derechos Reservados 2022

CONTENIDO

1	REQUERIMIENTOS.....	57
2	INGRESAR AL SISTEMA.....	57
3	USUARIOS DEL SISTEMA	64
4	FUNCIONALIDAD GENERAL	59
4.1	DASHBOARD	65
4.2	MENÚ USUARIO	66
4.3	MENÚ MATERIALES E INSUMOS	67
4.4	MENÚ EQUIPOS	69
4.5	MENÚ SERVICIOS.....	71
4.6	MENÚ REPORTES	74
4.6	PERFIL.....	75

1. REQUERIMIENTOS

- **Windows**

Sistema Operativo (32-bit and 64-bit)

Windows 8

Windows 10

Windows 11

Computador o tableta con procesador Pentium 4 o superior

2GB de RAM

- **Mac**

Sistema Operativo

macOS 10.15

macOS 11

macOS 12

Computador o tableta Mac con procesador Intel x86 o Apple silicon processor

2gb de RAM

- **Navegadores recomendados**

Google Chrome, Firefox y Edge

2. INGRESAR AL SISTEMA

Dentro de su navegador, teclee la siguiente dirección electrónica:

www.gstecnico.com

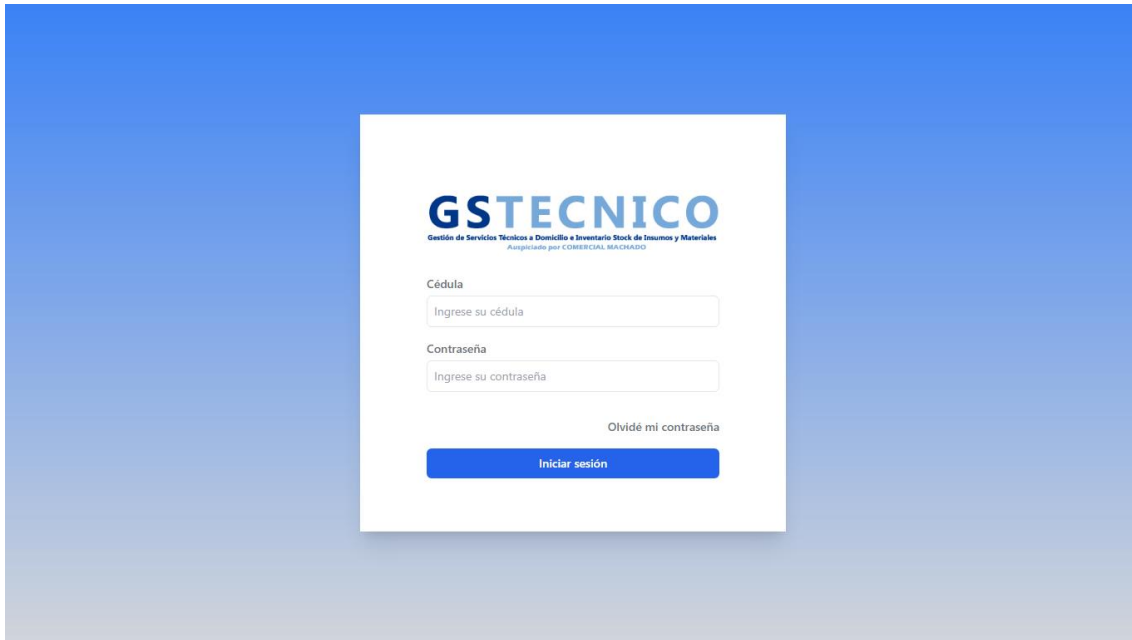


Gráfico No. 31 Guía ingreso al sistema

Elaborado por: Xavier Bejarano

Al ingresar a la web se encontrará con la ventana de Login, la misma cuenta con cuadros receptores de texto donde tendrá que ingresar sus credenciales: cédula de identidad y contraseña. Posteriormente deberá hacer click en el botón “Iniciar sesión” para acceder al menú principal o Dashboard.

7. USUARIOS DEL SISTEMA

- Administradores
- Técnicos
- Clientes

8. FUNCIONALIDAD GENERAL

4.1. Dashboard

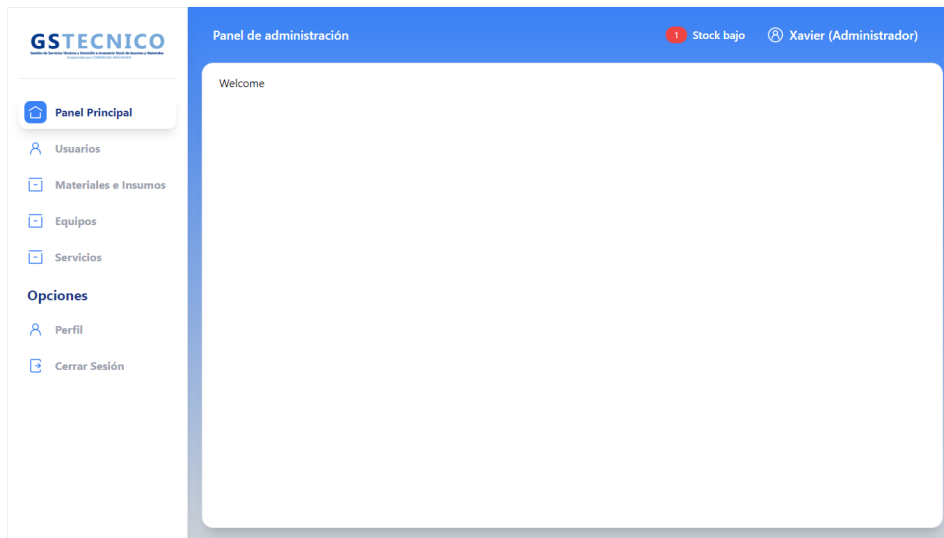


Gráfico No. 32 Guía acceso a Dashboard
Elaborado por: Xavier Bejarano

El Dashboard es el menú principal lateral que se va a mostrar en todas las pantallas después del ingreso al sistema, en dicho menú encontrará todas las opciones establecidas para cada usuario según su rol.

4.2. Menú Usuario

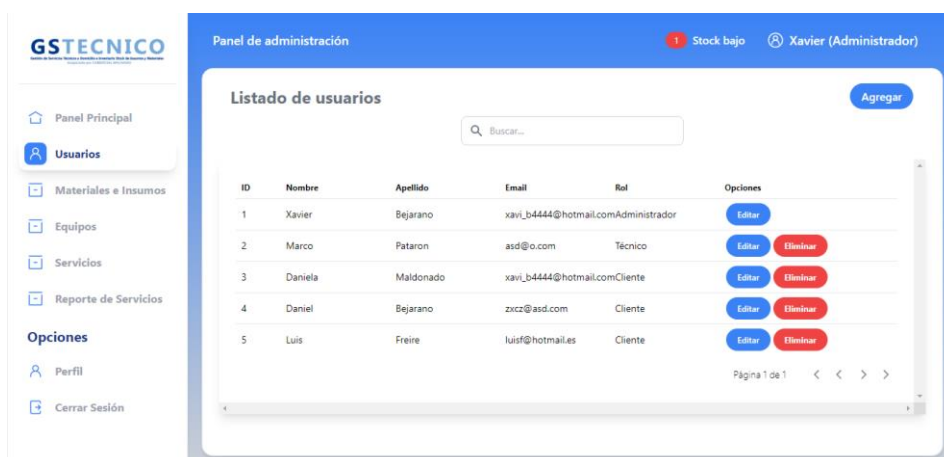


Gráfico No. 33 Guía acceso a menú de usuario
Elaborado por: Xavier Bejarano

En el menú usuario usted podrá encontrar el listado de los usuarios registrados en el sistema. Encontrará en este apartado datos del usuario como: id, nombre, apellido, email y rol; en la parte superior del listado encontrará una barra de búsqueda global que le permitirá acceder a cualquier dato del usuario de forma rápida.

The screenshot shows the 'Registrar Usuario' form within the GSTECHNICO administration panel. The form includes the following fields: NOMBRE (Nombre), APELLIDO (Apellido), CÉDULA (Cédula), TELÉFONO (Teléfono), EMAIL (Email), DIRECCIÓN (Dirección), ROL (Selección), and CONTRASERÍA (Contraseña). A 'Guardar' button is located in the top right corner of the form area.

Gráfico No. 34 Guía acceso a añadir usuario
Elaborado por: Xavier Bejarano

Haciendo click en el botón “Agregar” en la parte superior derecha, encontrará el apartado para agregar un nuevo usuario. Usted tendrá que llenar todos los datos del usuario, definir su rol y crear una contraseña, después, haciendo click en el botón “Guardar”, almacenará toda la información ingresada en el sistema.

4.3. Menú Materiales e Insumos

The screenshot shows the 'Listado de Materiales e Insumos' table within the GSTECHNICO administration panel. The table has a search bar and an 'Agregar' button. The table contains the following data:

ID	Serie	Marca	Modelo	Descripción	Tipo	Stock	Precio	Opciones
2	05646556	Hitachi	Memoria	Bgb	Repuesto	15	\$10.50	Editar Eliminar
3	23423408	Generico	Conector Sata rojo	Bus de datos	Repuesto	4	\$2.00	Editar Eliminar

Page 1 of 1

Gráfico No. 35 Guía acceso a inventario
Elaborado por: Xavier Bejarano

En el menú materiales e insumos usted podrá encontrar el listado de los materiales e insumos registrados en el sistema. Encontrará en este apartado datos de materiales e insumos como: id, serie, marca, modelo, descripción, tipo, stock y precio; en la parte superior del listado encontrará una barra de búsqueda global que le permitirá acceder a cualquier dato del material e insumo de forma rápida. Junto a cada insumo encontrará los botones: “Modificar” y “Eliminar”, que le permitirán modificar los datos del insumo o eliminarlo de su lista.

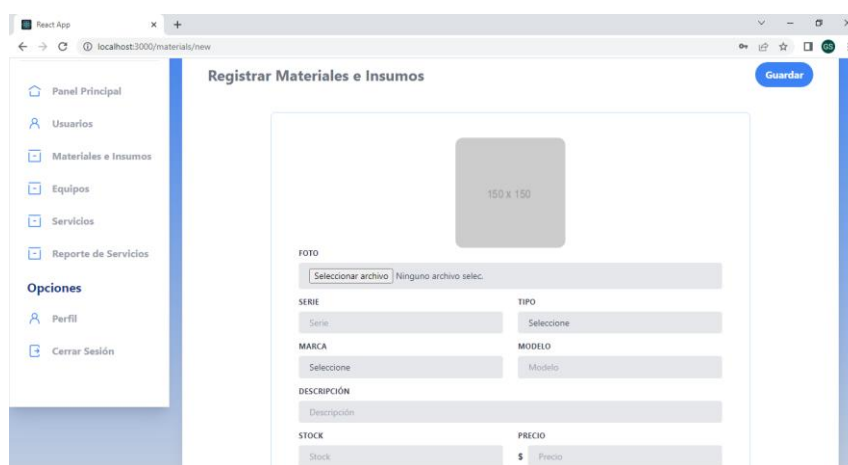


Gráfico No. 36 Guía añadir material e insumo

Elaborado por: Xavier Bejarano

En la parte superior derecha del menú materiales e insumos encontrará el botón “Agregar”, cuando haga click en el botón agregar encontrará el apartado para ingresar un nuevo material o insumo; deberá llenar todos los datos del mismo. En la parte superior encontrará el botón “Guardar”, haciendo click en el mismo almacenará la información ingresada al sistema.

4.4. Menú Equipos

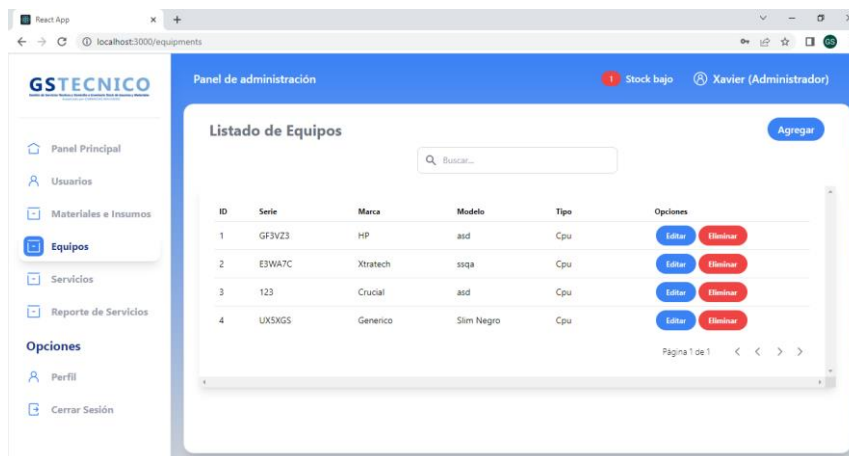


Gráfico No. 37 Guía menú equipos

Elaborado por: Xavier Bejarano

En el menú equipos usted podrá encontrar el listado de los equipos registrados en el sistema. Encontrará en este apartado datos de equipos como: id, serie, marca, modelo, tipo; en la parte superior del listado encontrará una barra de búsqueda global que le permitirá acceder a cualquier dato del equipo de forma rápida. Junto a cada equipo encontrará los botones: “Modificar” y “Eliminar”, que le permitirán modificar los datos del equipo o eliminarlo de su lista.

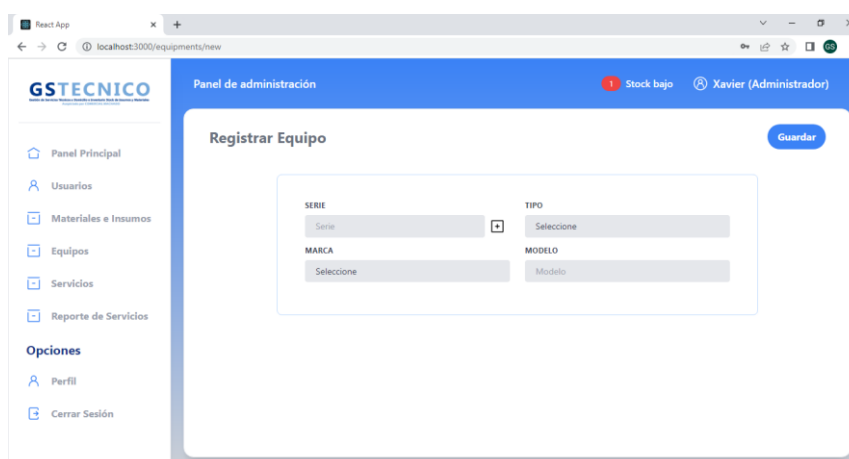


Gráfico No. 38 Guía añadir equipos

Elaborado por: Xavier Bejarano

En la parte superior derecha del menú equipos encontrará el botón “Agregar”, cuando haga click en el botón agregar encontrará el apartado para ingresar un nuevo equipo; deberá llenar todos los datos del mismo. En la parte superior encontrará el botón “Guardar”, haciendo click en el mismo almacenará la información ingresada al sistema.

4.5. Menú Servicios

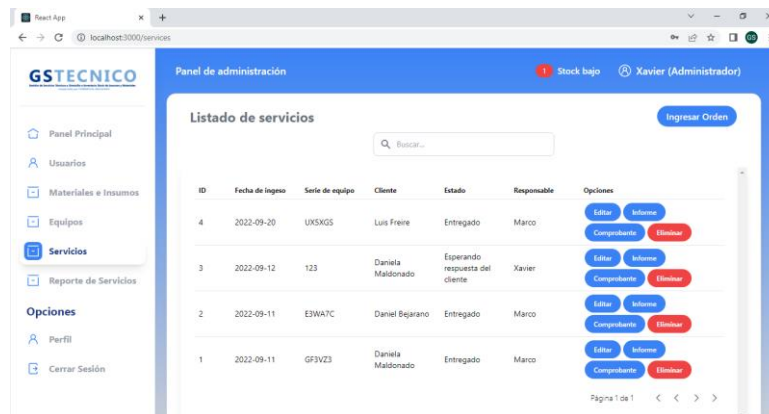


Gráfico No. 39 Guía menú servicios
Elaborado por: Xavier Bejarano

En el menú servicios usted podrá encontrar el listado de los servicios registrados en el sistema. Encontrará en este apartado datos del servicio como: id, fecha de ingreso, serie del equipo, cliente, estado, responsable; en la parte superior del listado encontrará una barra de búsqueda global que le permitirá acceder a cualquier dato del servicio de forma rápida.

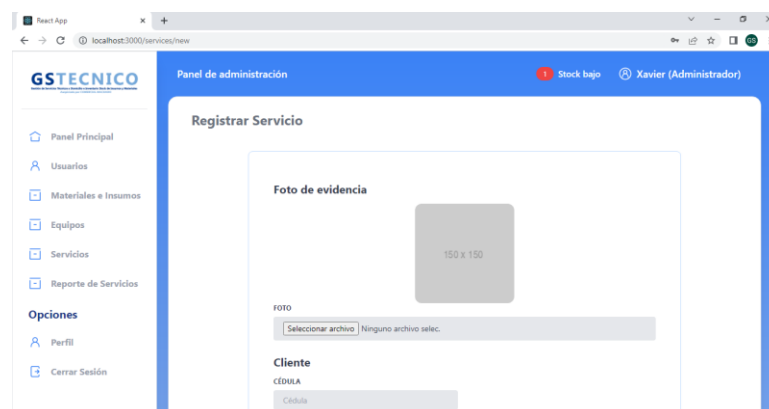


Gráfico No. 40 Guía añadir orden de reparación
Elaborado por: Xavier Bejarano

En la parte superior derecha del menú servicios encontrará el botón “Ingresar Orden”, cuando haga click en el mismo encontrará el apartado para ingresar una nueva orden de servicio; deberá llenar todos los datos requeridos. En la parte inferior encontrará el botón “Guardar”, haciendo click en el mismo almacenará la información ingresada al sistema, generando así un numero de Orden de Reparación y un comprobante el cual podrá ser impreso o descargado por el cliente.



Gráfico No. 41 Guía añadir informe técnico
Elaborado por: Xavier Bejarano

A lado de cada servicio técnico encontrará diferentes opciones: “Editar”; cuya función es darle acceso a la orden de reparación para poder modificar la información ingresada en el servicio técnico. “Informe”, cuya función es darle acceso a la orden de reparación para poder ingresar informes técnicos de los trabajos que se van realizando en el equipo. “Comprobante”, cuya función es darle acceso a un documento que se genera automáticamente tanto en el ingreso como en la salida del equipo a reparar. “Eliminar”, cuya función es suprimir o eliminar la orden de reparación.

4.6. Menú Reportes

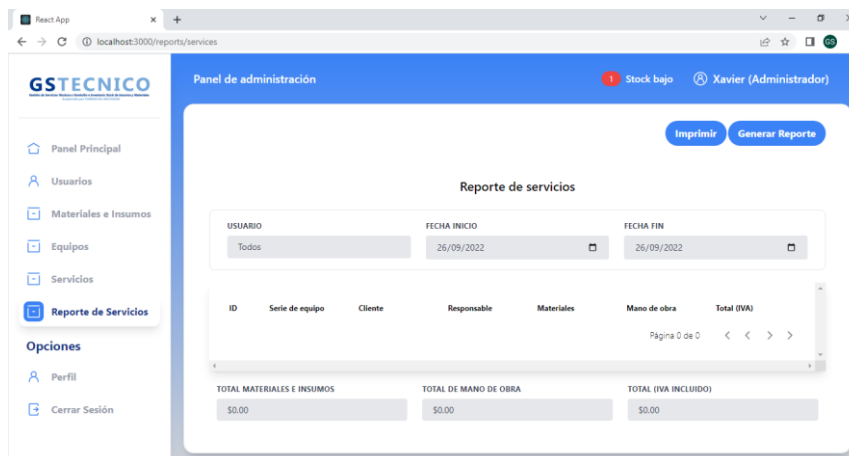


Gráfico No. 42 Guía reportes
Elaborado por: Xavier Bejarano

El menú reportes lo llevará a una ventana en la cual podrá realizar una búsqueda de servicios filtrados por fecha y responsable encargado, dicha búsqueda podrá ser impresa por el usuario.

4.7. Perfil

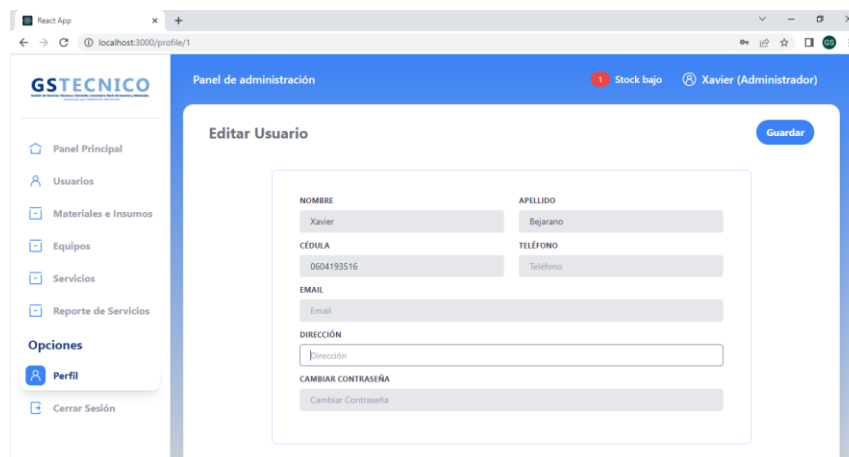


Gráfico No. 43 Guía modificar perfil
Elaborado por: Xavier Bejarano

En el apartado perfil encontrará una ventana donde se presentarán los datos ingresados por el usuario, dichos datos podrán ser modificados si así lo desea el usuario.

ANEXO 2.

MANUAL DEL TÉCNICO

MANUAL DEL TÉCNICO

GSTECNICO

Gestión de Servicios Técnicos a Domicilio e Inventario Stock de Insumos y Materiales
Auspiciado por COMERCIAL MACHADO

APLICATIVO WEB

CENTRO COMERCIAL MACHADO

GSTECNICO

VERSIÓN 1.0

REALIZADO POR:

XAVIER PATRICIO BEJARANO GAVILANES

Derechos Reservados 2022

CONTENIDO

1	REQUERIMIENTOS.....	69
2	BACKEND	69
2.1	INTERACCIONES CON LA BASE DE DATOS.....	81
2.1.1.	USUARIOS	81
2.1.2.	ROLES	82
2.1.3.	INSUMOS Y MATERIALES.....	82
2.1.4.	EQUIPOS	83
2.1.5.	SERVICIOS	84
3	FRONTEND.....	86
3.1	COMPONENTES	86
3.2	SERVICIOS WEB	88
3.3	MODIFICAR COMPROBANTES.....	94

2. REQUERIMIENTOS

Software Recomendado

- Windows Server 2016 (x64) o superior
- Ubuntu 20 (x64) o superior

Hardware Recomendado

- Servidor Virtual o Físico con 2 vCPUs o más
- 2GB de RAM o más
- 30GB de SSD o más
- 1TB de transferencia o más

Tecnologías

- PHP 7.1
- LARAVEL 8
- MYSQL 8
- XAMPP 8.1
- VISUAL STUDIO CODE 1.71

2. BACKEND

Se puede acceder al archivo de configuración de la base de datos desde el archivo. env ubicado dentro de la carpeta \wsmantenimiento. En este apartado se puede modificar los siguientes parámetros:

```
APP_NAME=Laravel
APP_ENV=local
APP_KEY=base64:EiNcXQB7V5/MALqAJMqB9nOmrFHkMx+9dPWMT5YI3QU=
APP_DEBUG=true
APP_URL=http://localhost

LOG_CHANNEL=stack
LOG_DEPRECATED_CHANNELS=null
LOG_LEVEL=debug

DB_CONNECTION=mysql
```



```
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=mantenimiento
DB_USERNAME=root
DB_PASSWORD=

BROADCAST_DRIVER=log
CACHE_DRIVER=file
FILESYSTEM_DRIVER=local
QUEUE_CONNECTION=sync
SESSION_DRIVER=file
SESSION_LIFETIME=120

MEMCACHED_HOST=127.0.0.1

REDIS_HOST=127.0.0.1
REDIS_PASSWORD=null
REDIS_PORT=6379

MAIL_MAILER=smtp
MAIL_HOST=smtp.gmail.com
MAIL_PORT=587
MAIL_USERNAME=gstecnicoec@gmail.com
MAIL_PASSWORD=udpzqwbswqftkpa
MAIL_ENCRYPTION=tls
MAIL_FROM_ADDRESS=gstecnicoec@gmail.com
MAIL_FROM_NAME="GSTECNICO"

AWS_ACCESS_KEY_ID=
AWS_SECRET_ACCESS_KEY=
AWS_DEFAULT_REGION=us-east-1
AWS_BUCKET=
AWS_USE_PATH_STYLE_ENDPOINT=false

PUSHER_APP_ID=
PUSHER_APP_KEY=
PUSHER_APP_SECRET=
PUSHER_APP_CLUSTER=mt1

MIX_PUSHER_APP_KEY="{PUSHER_APP_KEY}"
MIX_PUSHER_APP_CLUSTER="{PUSHER_APP_CLUSTER}"
```

2.1. INTERACCIONES CON LA BASE DE DATOS

2.1.1. Usuarios

Se puede acceder al archivo de configuración de usuarios desde el archivo user.php ubicado dentro de la carpeta \wsmantenimiento. En este apartado se puede modificar los siguientes parámetros:

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateUsersTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('users', function (Blueprint $table) {
            $table->id();
            $table->string('ci')->unique();
            $table->string('name')->nullable();
            $table->string('last_name')->nullable();
            $table->string('phone')->nullable();
            $table->string('address')->nullable();
            $table->string('email')->nullable();
            $table->timestamp('email_verified_at')->nullable();
            $table->string('password');
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('users');
    }
}
```

2.1.2. Roles

Se puede acceder al archivo de configuración de roles desde el archivo rol.php ubicado dentro de la carpeta \wsmantenimiento. En este apartado se puede modificar los siguientes parámetros:

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateRolesTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('roles', function (Blueprint $table) {
            $table->id();
            $table->string('name');
            $table->string('description')->nullable();
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('roles');
    }
}
```

2.1.3. Insumos y Materiales

Se puede acceder al archivo de configuración de insumos y materiales desde el archivo insumo_material.php ubicado dentro de la carpeta \wsmantenimiento. En este apartado se puede modificar los siguientes parámetros:

```

<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateMaterialsTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('materials', function (Blueprint $table) {
            $table->id();
            $table->string('serie');
            $table->integer('brand');
            $table->string('model');
            $table->text('description')->nullable();
            $table->integer('stock');
            $table->decimal('price', 9, 2);
            $table->integer('type');
            $table->longText('image')->nullable();
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('materials');
    }
}

```

2.1.4. Equipos

Se puede acceder al archivo de configuración de equipos desde el archivo equipment.php ubicado dentro de la carpeta \wsmantenimiento. En este apartado se puede modificar los siguientes parámetros:

```

<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateEquipmentsTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('equipments', function (Blueprint $table) {
            $table->id();

            $table->string('serie');
            $table->integer('type');
            $table->integer('brand');
            $table->string('model');

            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('equipments');
    }
}

```

2.1.5. Servicios

Se puede acceder al archivo de configuración de servicios desde el archivo `services.php` ubicado dentro de la carpeta `\wsmantenimiento`. En este apartado se puede modificar los siguientes parámetros:

```

<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;

```

```

use Illuminate\Support\Facades\Schema;

class CreateServicesTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('services', function (Blueprint $table) {
            $table->id();
            $table->longText('image')->nullable();

            $table->foreignId('income_id')->constrained("users")->onDelete('cascade')-
>onUpdate('cascade');
            $table->foreignId('client_id')->constrained("users")->onDelete('cascade')-
>onUpdate('cascade');
            $table->foreignId('responsible_id')->constrained("users")->onDelete('cascade')-
>onUpdate('cascade');
            $table->foreignId('equipment_id')->constrained("equipments")-
>onDelete('cascade')->onUpdate('cascade');
            $table->foreignId('delivery_id')->nullable()->constrained("users")-
>onDelete('cascade')->onUpdate('cascade');

            $table->text('accessories')->nullable();
            $table->text('failure')->nullable();

            $table->text('technical_report')->nullable();
            $table->text('client_report')->nullable();

            $table->integer('status');

            $table->timestamp('delivery_date');
            $table->timestamp('finish_date')->nullable();
            $table->timestamp('delivery_date_real')->nullable();

            $table->decimal('price', 8, 2)->nullable();

            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()

```

```

    {
      Schema::dropIfExists('services');
    }
  }
}

```

3. FRONTEND

3.1. Componentes

Se puede acceder al archivo de configuración de componentes desde el archivo `components.php` ubicado dentro de la carpeta `\uimantenimiento`. En este apartado se puede modificar los siguientes parámetros:

```

import React from 'react';
import { Link } from 'react-router-dom';
import { Home, User, LogOut, Box } from 'iconoir-react';
import { ButtonNav, ButtonNavLogout } from './ButtonNav';

import logo from '../assets/images/logo.png';
/* import logo2 from "../assets/images/logo2.png"; */
import { AppContext } from '../context/AppContext';

export default function Navbar({ closeSession }) {
  const [appState] = React.useContext(AppContext);

  const idRole = appState.user.roles[0]?.id;

  return (
    <div className="max-h-screen w-[260px] border bg-white py-8 px-4 shadow-xl">
      <div className="mb-6 border-b px-4 pb-6">
        <Link to="/" className="">
          <img src={logo} alt="logo" />
        </Link>
      </div>

      <div className="">
        <ButtonNav text="Panel Principal" to="/">
          <Home />
        </ButtonNav>

        {idRole === 1 && (
          <ButtonNav text="Usuarios" to="/users">
            <User />
          </ButtonNav>
        )}

        {idRole === 1 && (

```

```

    <ButtonNav text="Materiales e Insumos" to="/materials">
      <Box />
    </ButtonNav>
  )}

  {idRole === 1 && (
    <ButtonNav text="Equipos" to="/equipments">
      <Box />
    </ButtonNav>
  )}

  {(idRole === 1 || idRole === 2) && (
    <ButtonNav text="Servicios" to="/services">
      <Box />
    </ButtonNav>
  )}

  {idRole === 3 && (
    <ButtonNav text="Mis Servicios" to="/services/client">
      <Box />
    </ButtonNav>
  )}

  {(idRole === 1 || idRole === 2) && (
    <ButtonNav text="Reporte de Servicios" to="/reports/services">
      <Box />
    </ButtonNav>
  )}
</div>

<div className="mx-3 mt-4 mb-3">
  <span className="text-xl font-bold text-blue-900">Opciones</span>
</div>

<ButtonNav text="Perfil" to={`\profile/${appState.user.id}`}>
  <User />
</ButtonNav>
<ButtonNavLogout text="Cerrar Sesión" onClick={closeSession}>
  <LogOut />
</ButtonNavLogout>

{ /* <div className="mt-12 flex w-full items-center justify-center">
  <div className="w-28 border-b">
    <Link to="/" className="">
      <img src={logo2} alt="logo" />
    </Link>
  </div>
</div> */ }
</div>
); }

```


3.2. Servicios Web

Se puede acceder al archivo de configuración de servicios web desde el archivo `web_services.php` ubicado dentro de la carpeta `\uimantenimiento`. En este apartado se puede modificar los siguientes parámetros:

```
const url = 'http://localhost:8000/api';

export const login = async (user) => {
  const response = await fetch(url + 'users/login', {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
    },
    body: JSON.stringify(user),
  });
  return await response.json();
};

export const restoreLogin = async (id, token) => {
  try {
    const response = await fetch(url + 'users/restoreLogin/' + id, {
      method: 'POST',
      headers: {
        'Content-Type': 'application/json',
      },
      body: JSON.stringify({ token: token }),
    });
    return await response.json();
  } catch (error) {
    return { success: false, message: 'Error' };
  }
};

export const postUser = async (user) => {
  const response = await fetch(url + 'users', {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
    },
    body: JSON.stringify(user),
  });
  return await response.json();
};

export const getUsers = async () => {
  const response = await fetch(url + 'users');
  return await response.json();
};
```

```

};

export const getUser = async (id) => {
  const response = await fetch(url + 'users/' + id);
  return await response.json();
};

export const getUserByCI = async (ci) => {
  const response = await fetch(url + 'users/ci/' + ci);
  return await response.json();
};

export const putUser = async (user) => {
  const response = await fetch(url + 'users/' + user.id, {
    method: 'PUT',
    headers: {
      'Content-Type': 'application/json',
    },
    body: JSON.stringify(user),
  });
  return await response.json();
};

export const deleteUser = async (id) => {
  const response = await fetch(url + 'users/' + id, {
    method: 'DELETE',
  });
  return await response.json();
};

export const getRoles = async () => {
  const response = await fetch(url + 'roles');
  return await response.json();
};

export const postRecoverPassword = async (user) => {
  const response = await fetch(url + 'users/recoverPassword', {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
    },
    body: JSON.stringify(user),
  });
  return await response.json();
};

export const postMaterial = async (material) => {
  const response = await fetch(url + 'materials', {
    method: 'POST',
    headers: {

```

```

    'Content-Type': 'application/json',
  },
  body: JSON.stringify(material),
});
return await response.json();
};

export const getMaterials = async () => {
  const response = await fetch(url + 'materials');
  return await response.json();
};

export const getMaterial = async (id) => {
  const response = await fetch(url + 'materials/' + id);
  return await response.json();
};

export const putMaterial = async (material) => {
  const response = await fetch(url + 'materials/' + material.id, {
    method: 'PUT',
    headers: {
      'Content-Type': 'application/json',
    },
    body: JSON.stringify(material),
  });
  return await response.json();
};

export const deleteMaterial = async (id) => {
  const response = await fetch(url + 'materials/' + id, {
    method: 'DELETE',
  });
  return await response.json();
};

export const getLowStock = async (lowStock = 5) => {
  const response = await fetch(url + 'materials/lowStock/' + lowStock);
  return await response.json();
};

export const postEquipment = async (equipment) => {
  const response = await fetch(url + 'equipments', {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
    },
    body: JSON.stringify(equipment),
  });
  return await response.json();
};

```

```

export const getEquipments = async () => {
  const response = await fetch(url + 'equipments');
  return await response.json();
};

export const getEquipment = async (id) => {
  const response = await fetch(url + 'equipments/' + id);
  return await response.json();
};

export const putEquipment = async (equipment) => {
  const response = await fetch(url + 'equipments/' + equipment.id, {
    method: 'PUT',
    headers: {
      'Content-Type': 'application/json',
    },
    body: JSON.stringify(equipment),
  });
  return await response.json();
};

export const deleteEquipment = async (id) => {
  const response = await fetch(url + 'equipments/' + id, {
    method: 'DELETE',
  });
  return await response.json();
};

export const getEquipmentBySerie = async (equipment) => {
  const response = await fetch(url + 'equipments/serie', {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
    },
    body: JSON.stringify(equipment),
  });
  return await response.json();
};

export const postService = async (service) => {
  const response = await fetch(url + 'services', {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
    },
    body: JSON.stringify(service),
  });
  return await response.json();
};

```

```

export const getServices = async () => {
  const response = await fetch(url + 'services');
  return await response.json();
};

export const getService = async (id) => {
  const response = await fetch(url + 'services/' + id);
  return await response.json();
};

export const putService = async (service) => {
  const response = await fetch(url + 'services/' + service.id, {
    method: 'PUT',
    headers: {
      'Content-Type': 'application/json',
    },
    body: JSON.stringify(service),
  });
  return await response.json();
};

export const deleteService = async (id) => {
  const response = await fetch(url + 'services/' + id, {
    method: 'DELETE',
  });
  return await response.json();
};

export const putServiceReported = async (service) => {
  const response = await fetch(url + 'services/reported/' + service.id, {
    method: 'PUT',
    headers: {
      'Content-Type': 'application/json',
    },
    body: JSON.stringify(service),
  });
  return await response.json();
};

export const putServiceReportedFinished = async (service) => {
  const response = await fetch(
    url + 'services/reported/finished/' + service.id,
    {
      method: 'PUT',
      headers: {
        'Content-Type': 'application/json',
      },
      body: JSON.stringify(service),
    }
  )
};

```

```

);
return await response.json();
};

export const putServiceReportedReopen = async (service) => {
  const response = await fetch(url + 'services/reported/reopen/' + service.id, {
    method: 'PUT',
    headers: {
      'Content-Type': 'application/json',
    },
    body: JSON.stringify(service),
  });
  return await response.json();
};

export const validateEquipmentService = async (json) => {
  const response = await fetch(url + 'services/validateEquipmentService', {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
    },
    body: JSON.stringify(json),
  });
  return await response.json();
};

export const getServicesByClient = async (idUser) => {
  const response = await fetch(url + 'services/client/' + idUser);
  return await response.json();
};

export const reportServiceResponsibleByDate = async (json) => {
  const response = await fetch(url + 'services/report', {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
    },
    body: JSON.stringify(json),
  });
  return await response.json();
};

```

3.3. Modificar Comprobantes

Se puede acceder al archivo de configuración de comprobantes desde el archivo `utils.php` ubicado dentro de la carpeta `\uimantenimiento`. En este apartado se puede modificar los siguientes parámetros:

```

import React from "react";

export const firm = "COMERCIAL MACHADO";

export const direction =
  "Guayaquil y Colón, CENTRO COMERCIAL MACHADO - Riobamba - Teléfono
  0987751349";

export function Description() {
  return (
    <div>
      <p className="text-justify text-sm">
        Al ingresar un equipo a nuestro Centro Especializado de Servicio Técnico
        el personal podría encontrar fallos adicionales a los reportados por
        usted en cuyo caso se procederá a informarle sobre el hecho y bajo
        ningún concepto se podrá responsabilizar al técnico por la ocurrencia de
        fallas no mencionadas al momento del ingreso.
      </p>
      <p className="mt-2 text-justify text-sm">
        Toda orden de reparación tendrá un costo mínimo de diez dólares por
        concepto de revisión y diagnóstico. Una vez encontrada la solución dicho
        valor será abonado al costo total de la reparación previo acuerdo con el
        cliente, en el caso de que el cliente no desee la reparación cancelará
        el costo de la revisión y diagnóstico.
      </p>
      <p className="mt-2 text-justify text-sm">
        <b>AUTORIZACIÓN:</b> Yo, el CLIENTE, en caso de no retirar mi equipo en
        reparación en un lapso de 60 días a partir de la fecha de ingreso,
        AUTORIZO a GSTECNICO a reciclar el mismo. Al firmar este documento
        ACEPTO todas las condiciones mencionadas.
      </p>
      <p className="mt-2 text-justify text-sm">
        <b>
          PUEDE ACCEDER A GSTECNICO EN LINEA PARA CONSULTAR ESTADO
          Y NOVEDADES
          DE SU EQUIPO EN: www.gstecnico.com / usuario y contraseña: cédula de
          identidad registrada con el equipo
        </b>
      </p>
      <p className="mt-2 justify-center text-center text-sm font-semibold">
        ESTE DOCUMENTO ES EL ÚNICO COMPROBANTE VALIDO QUE LE
        ACREDITA COMO DUEÑO
        DEL EQUIPO POR ENDE EL ORIGINAL DE ESTE DOCUMENTO ES DE
        PRESENTACIÓN
        OBLIGATORIA PARA SU RETIRO
      </p>
    </div>
  );
}

```

```
export function Description2() {
  return (
    <div>
      <p className="text-justify text-sm">
        El servicio técnico realizado tiene una garantía de treinta días sobre
        la falla reportada, insumos y materiales tienen garantía especificada
        por el fabricante.
      </p>

      <p className="mt-2 justify-center text-center text-sm font-semibold">
        ESTE DOCUMENTO ES EL ÚNICO COMPROBANTE VALIDO PARA
        VALIDAR GARANTÍA POR
        ENDE EL ORIGINAL DE ESTE DOCUMENTO ES DE PRESENTACIÓN
        OBLIGATORIA PARA
        LA GESTIÓN DE LA MISMA.
      </p>
    </div>
  );
}
```