Documentación

Gestión Empresarial

Programa: ERP

Autor: Cristina Fuster García

# **1. Documentación de usuario final:**

## **1.1. Guía de usuario**

Es un sistema integral de planificación de recursos empresariales, diseñado para poder meterse en cualquier módulo que tenga el escritorio, y acceder a la base de datos dinámicamente con una interfaz más agradable e intuitiva.

El proyecto permite a los usuarios poder ingresar, buscar, insertar y eliminar registros.

Funcionalidades Principales

1.Inicio de sesión: donde nos vamos a encontrar un login, el usuario tendra que ingresar en los inputs la contraseña y el nombre se usuario. Y una vez que la autentificación es valida le redirigira al escritorio.

2.Escritorio: una vez dentro apareceran una lista de aplicaciones con un estilo de tarjeta y efecto, pulsara encima de una tarjeta y lo reconducir al supercontrolador.

3. Supercontrolador: aqui es donde estan todas las funciones aplicadas:

* Visualiza los datos
* Podrá hacer búsquedas con filtrado
* Inserta datos
* Actualización de datos: esto le permitirá al usuario que al hacer doble click encima de registro y salga de la casilla del registro se guardará automáticamente en la base de datos.

## **1.2. FAQ (Preguntas frecuentes)**

¿Que es la aplicación?

Es un sistema diseñado para la gestión de una empresa , donde se puede ingresar y hacer operaciones.

¿Cómo se gestionan los permisos de usuario?

Esta gestión se realiza desde la base de datos .

# **2. Documentación técnica para desarrolladores**

## **2.1. Introducción al software**

Es una aplicación desarrollada con php (backend), javascript (frontend) perimitiendo interactuar con diversas tablas de datos y realizar las operaciones CRUD.

Objetivos Principales

Facilitar la administración de información empresarial a través de una interfaz intuitiva.

Proveer a los usuario funcionalidades dinámicas en tiempo real.

Implementar mecanismos de autenticación para proteger los datos.

Tecnologías Utilizadas

Frontend:

* Html, CSS, JavaScript
* Fetch Api para las comunicaciones

Backend

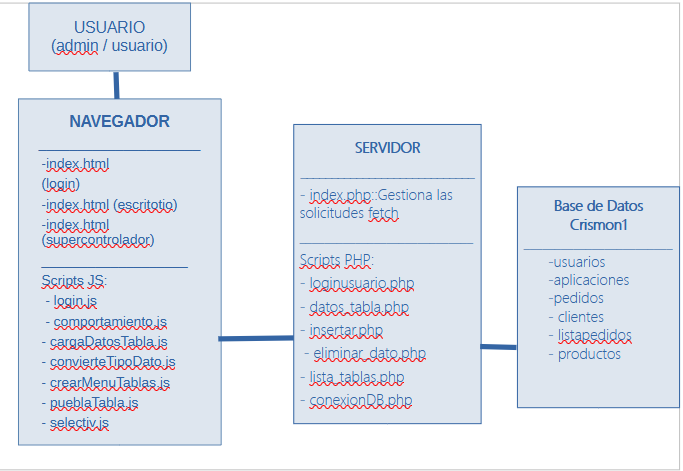
* PHP
* MYSQL

Otras Herramientas: Json para el intercambio de datos entre el cliente y servidor.

## **2.2. Arquitectura del sistema**

Estructura General

Está diseñado siguiendo una estructura cliente-servidor, donde el frontend interactúa con el backend, la comunicación se realiza mediante HTTP utilizando métodos POST y GET.



Componentes Principales

1. Cliente (Frontend)

html/CSS: estructura y los estilos de la pagina

JS: maneja los eventos, dinámicas de interfaz principal y la comunicación con el servidor.

1. Servidor (Backend)

PHP Scripts: Manejo de lógica de negocio y operaciones de base de datos.

Clases PHP: conexionDb.php es la clase principal para la interacción de base de datos.

Archivos clave:

index.php: es el controlador que va a dirigir las solicitudes a las funciones correspondientes.

la carpeta de antiguo es la que tiene los scripts para manejar los datos

1. Base de datos:

MySQL: almacena todas las tablas y datos necesarios para el funcionamiento del ERP.

## **2.3. Estructura del código**

#### **Directorios y Archivos**

Descripción de los componentes claves

Cliente

* login.js: maneja el proceso de autentificacionse usuarios.
* estilo.css aplica los estilos del login
* index.html: página de inicio de sesion
* Carpeta Aplicaciones
* Carpeta escritorio:
  + comportamiento.js: gestiona la carga dinámica de aplicaciones y navegación.
  + estilo.css: aplicara los estilos al escritorio.
  + index.html: es el esqueleto donde se carga la taba de aplicaciones
* Carpeta Supercontrolador
  + comportamiento.js: maneja la interacción con los datos recibidos del servidor
  + estilo.css: aplicamos los efectos visuales del supercontrolador
  + index.html: la página principal del supercontrolador
  + Carpeta JS
    - cargaDatosTabla.js: es el archivo que va cargar el contenido de los datos que el usuario seleccione.
    - convierteTipoData.js: convierte los tipos e datos SQL a tipo de entrad HTML.
    - crearMenuTabla.js : crea el menu dinamico de las tablas que contiene la base de datos.
    - pueblaTbla.js: rellena a tabla co los datos recibidos del servidor
    - Carpeta lib/selectjv
      * selectjv y el css: implementamos un servidor personalizados con funcionalidades avanzadas

- Carpeta Servidor:

- Carpeta Antiguo:

# **3. Guías de instalación y despliegue**

## **3.1. Requisitos previos**

Software:

* Sistema Operativo
* servidor WEB: Apache2.4
* php con las extensiones de mysql y json
* base de datos (Mysql)

## **3.2. Instrucciones de instalación**

1. Instalar Apache
2. Habilitar php
3. Crear base de datos y asignarle los privilegios de usuario.
4. Editar conexionDB para que coincida las credenciales de la base de datos con la original o con otro nombre.

# **4. Mantenimiento y soporte**

## **4.1. Manejo de errores**

Error de autentificación, porque no hay registros en la tabla.

Problema al insertar un nuevo registro : que los valores que has introducido no correspondan.

Error al procesar la solicitud o consulta SQL mal formada, asegurarnos de que los parámetros estén proporcionados.

## **4.2. Registro de cambios (changelog)**

version 21: Añadimos el emoji de la papelar creando la columna respectiva desde el comportamiento del supercontrolador llamando d¡con una peticion fecth.

version 38. implementemos la barra de buscador y realicemos la integracion con acceso a datos, llevando y modificando los archivos, solo tuvimos que cambiar la conexion a la base de datos y creemos un index.php que es el que maneja las peticiones y nos lleva a las funciones de conexionDB.

Version 47. Integración del select , aqui tuvimos que modificar dos archivos para que empezara a funcionar .Uno fue el archivo de cargaDatosTabla.js: al que le creamos la pantalla modal que va a ser la encargada de enseñarnos un formulario donde tambien aparece el select de la asignatura de interfaces.

## **4.3. Soporte y contacto**