

**Universidad Francisco José de Caldas**

FACULTAD INGENIERÍA DE SISTEMAS

# PROYECTO WEB

*Asistente Financiero*

Autor:

Henry Ricaurte Mora - 20221020084  
Santiago Buitrago

Diciembre 05 2023

# Índice general

1.	Objetivos Del proyecto . . . . .	1
2.	Prologo Pagina web . . . . .	1
2.1.	¿Que es HTML5, CSS3 y JavaScript? . . . . .	1
2.2.	¿Que es PHP y SQLite? . . . . .	2
2.3.	¿Qué es un asistente Financiero? . . . . .	2
3.	Diseño de la Pagina web . . . . .	2
3.1.	Mapa de navegacion Web . . . . .	2
3.2.	Casos de Uso . . . . .	3
3.3.	Paginacion Web . . . . .	3
4.	Conclusiones: . . . . .	8

## **Resumen**

En el presente documento, se abordará la descripción de Ingper, un asistente financiero web diseñado para gestionar aspectos clave como inicio de sesión, registro de gastos, ingresos y la gestión de cuentas asociadas a estas transacciones. Además, se incluirán diversas representaciones visuales, como gráficos y tablas, con el fin de mejorar la comprensión y seguimiento de nuestras finanzas.

El desarrollo de Ingper se llevará a cabo mediante el uso de tecnologías web avanzadas, tales como JavaScript, HTML5, CSS3, PHP y SQLite. La elección de estas tecnologías se fundamenta en su capacidad para proporcionar una solución legible y de fácil comprensión, facilitando así la resolución efectiva de los desafíos planteados en el presente proyecto.

# 1. Objetivos Del proyecto

El presente proyecto tiene como objetivos principales adquirir un conocimiento sólido de las tecnologías web en un nivel fundamental. Esto incluye la implementación de API's mediante llamadas Fetch, así como comprender la arquitectura por capas, que implica la separación de la lógica, la presentación y la persistencia.

Otro objetivo crucial es comprender los llamados por formularios de tipo GET y POST, además de almacenar cada uno de los procedimientos requeridos en una base de datos. Una vez alcanzados los objetivos técnicos, se busca proporcionar al usuario una experiencia cómoda y segura al visualizar sus gastos e ingresos. Esto permitirá mejorar la gestión financiera personal al identificar áreas de mayor gasto, analizar los meses con mayores gastos, y ofrecer sugerencias para mejorar en el ámbito financiero, todo ello a través de la programación.

## 2. Prologo Pagina web

### 2.1. ¿Que es HTML5, CSS3 y JavaScript?

HTML es un lenguaje de marcado para la elaboración de páginas web, este se usa mediante las etiquetas la cual son formas en las que es posible generar un contenido, la manera básica para un archivo html es:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
    initial-scale=1.0">
  <title>Document</title>
</head>
<body>

</body>
</html>
```

---

Por otro lado CSS es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado, la cual lleva una forma :

```
*{
  font-family: 'Fira Sans', sans-serif;
  margin: 0;
  padding: 0;
}
body{
  position: absolute;
  display: flex;
}
```

---

Y JavaScript es un lenguaje de secuencias de comandos que te permite crear contenido de actualización dinámica, controlar multimedia, animar imágenes y prácticamente todo lo demás, es un lenguaje de tipado débil, por lo que hace más amena su sintaxis que se evidenciara en los enunciados siguientes.

## 2.2. ¿Que es PHP y SQLite?

PHP es un lenguaje de programación de código abierto. En concreto, este lenguaje de scripting es uno de los más populares entre los desarrolladores backend (del lado del servidor).

```
<?php
echo "Hola, mundo!";
?>
```

## 2.3. ¿Qué es un asistente Financiero?

Un asistente financiero es aquel que se encarga de poder realizar un seguimiento a cada una de las acciones monetarias o económicas de un cliente determinado, este asistente recopila la información y genera tablas, datos, informes y gráficas correspondientes para una mejor lectura y de esta manera poder dar a entender la situación actual del solicitante.

# 3. Diseño de la Página web

## 3.1. Mapa de navegación Web

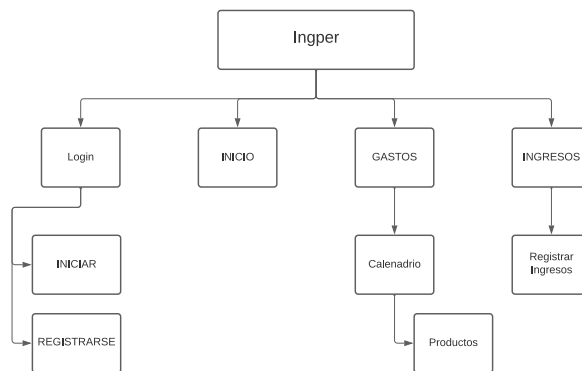


Figura 1: Diagrama de navegación web.

### 3.2. Casos de Uso

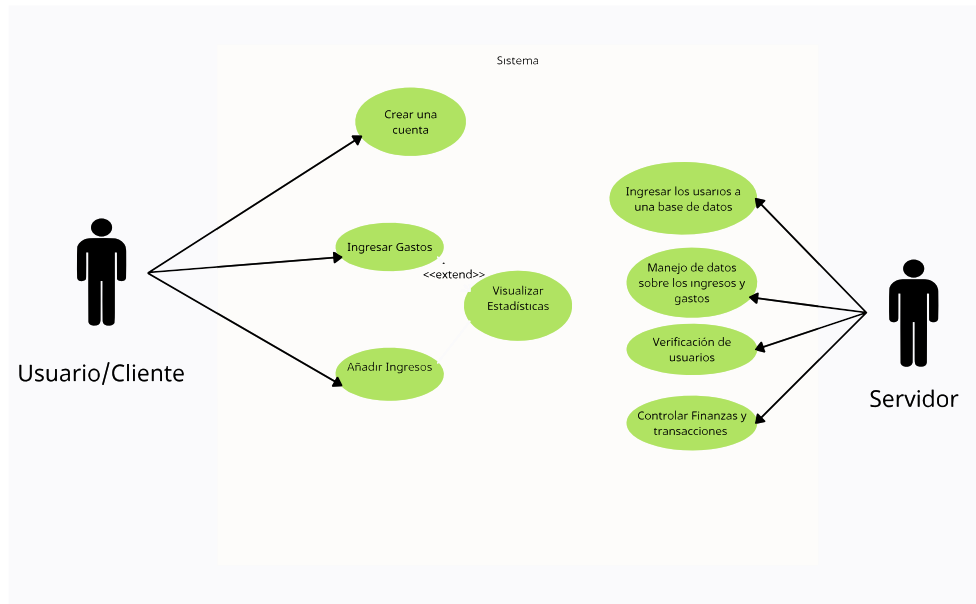


Figura 2: Diagrama de casos de uso.

### 3.3. Paginacion Web

Login

EL codigo de html para el login consta , dentro del main, de tres subdivisiones en el cual, el primero muestra un form donde va el form para envio POST para el login, en el otro POST para la hora de ingresar, y el ultimo apartado es para generar los botones submit:

```
<!DOCTYPE html>
<html lang="es">
<head>
    <!--Link css-->
</head>
<body>
    <header>
        <!--El logotipo en el header-->
    </header>
    <main>
        <div class="contenedorPrincipal">
            <div class="formContenedor ingresar">
                <form id="log-in">
                    <!-- Form login -->
                </form>
            </div>
        </div>
    </main>
</body>
</html>
```

```

        <div class="formContenedor registrarse">
            <form id="sing-in">
                <!-- Form Ingreso -->
            </form>
        </div>
        <div class="contenedorCambio">
            <!-- Los cambios -->
        </div>

    </div>
</main>
<!--Scripts-->
</body>
</html>

```

Los dos codigos de php el primero para el login y el segundo para el sing in:

```

<?php
//iniciamos la sesion:
session_start();
//miramos la base de datos:
$login = new SQLite3('baseDato.db');
$user = $_POST['user'];
$pass = $_POST['password'];
// ahora buscamos al usurio:
$busca = $login->prepare('SELECT id,Usuario,Contrasena
,Nombre,Apellido,Edad FROM login WHERE Usuario = :usuario');
$busca->bindParam(':usuario', $user,SQLITE3_TEXT);
$respuesta = $busca->execute();
//miramos con un fecht Array , si ingresa
if ($row = $respuesta->fetchArray()) {
    //miramos la contrasena
    if($pass === $row['Contrasena']){
        //coloco los session para mandar
        $_SESSION['nombre'] = $row['Nombre'];
        $_SESSION['apellido'] = $row['Apellido'];
        $_SESSION['edad'] = $row['Edad'];
        $_SESSION['id'] = $row['id']; //pasamos el id
        // redirigimos
        $jj = array("url" => "../Vistas/Paginas/inicio.html");
        echo json_encode($jj); // enviamos como json
        die(); // se acaba
    }
    else{
        $jj = array("url" => '1' , "txt" =>"El usuario o contrasena
han sido invalidos");
        echo json_encode($jj);
    }
}
else{
    $jj = array("url" => '1' , "txt" =>"El usuario o contrasena
han sido invalidos");
    echo json_encode($jj);
}
?>

```

```

<?php
$singin = new SQLite3('baseDato.db');

$nombreIn = $_POST['nombreIn'];
$usuarioIn = $_POST['usuarioIn'];
$passwordIn = $_POST['passwordIn'];
$apellidoIn = "-";
$edadIn = 0;
//ahora el comando para ingresar
$ingresar = "INSERT INTO login(Usuario,Contrasena,Nombre,Apellido,
    Edad)
VALUES('$usuarioIn','$passwordIn','$nombreIn','$apellidoIn','$
    $edadIn')";
// aca ingresamos cada uno
//ejecutamos la consulta:
$res = $singin->exec($ingresar);

if ($res) {
    echo json_encode("Fila insertada con exito");
} else {
    echo json_encode("Error al insertar fila: " . $singin->
        lastErrorMsg());
}
?>

```

Foto:

## Inicio

En el inicio se utilizo diferentes archivos, el primero de ellos fue un scrip la cual, junto con el css global, daba lugar a el header, en donde se pone en cada una de las rutas a la cual se dirige la pagina. Tal script esta en todas las paginas junto al global, esto debido a que en todas debe de estar la misma referencia.

En este , el html recoge diferentes actividades, tanto un modalScreen para modificar el perfil como la aplicacion de diferentes apis para la creacion de graficos y de tablas respectivamente, generando una vista atractiva para el usuario:

### ■ Pantalla:

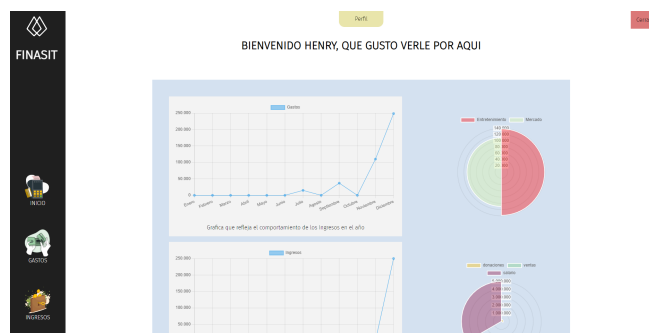


Figura 3: Pagina Inicio.



- Modal:

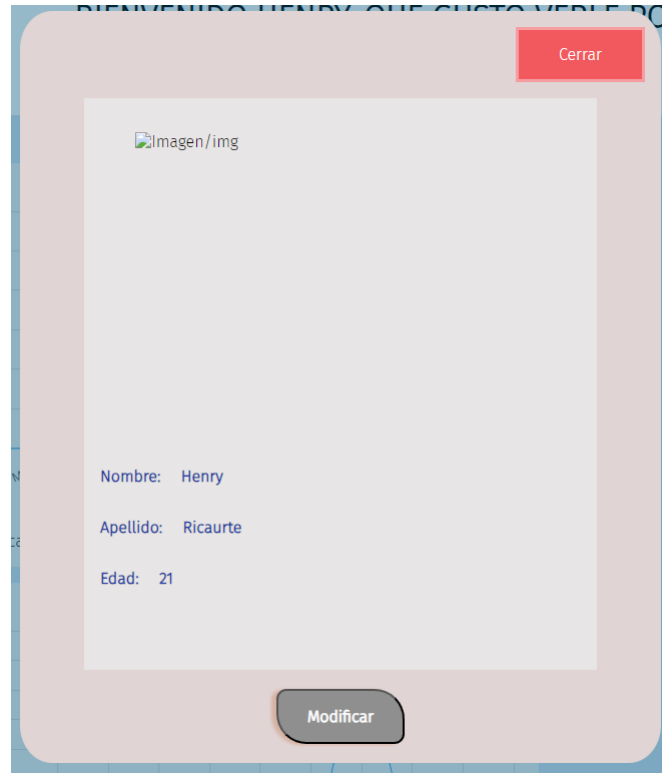


Figura 4: Modal Inicio.

- PHP y JavaScript: En javascript se escribieron codigos para obtener los datos con funciones asincronas como obtenerDatos(), y ahi mismo crear las graficas mediante el script de chartjs y gridjs para las tablas requeridas. Así realizando llamados:

```
async function obtenerDatos(){
  try{
    const respuesta = await fetch('../src/PHP/inicio.php'
    ,{method:'POST'});
    const data = await respuesta.json()

    if(data.dataPersona.url){
      window.location.replace(data.url);
    }
    nombrePerfil.innerHTML=data.dataPersona.nombre;
    nombrePerfil.innerHTML=data.dataPersona.nombre;
    apellidoPerfil.innerHTML = data.dataPersona.apellido;
    edadPerfil.innerHTML = data.dataPersona.edad;
  }
}
```

```

//ahora pondermos lo que son lo de los meses en lkos
gastos, se puede automatizar en vez de colocar "
ENERO,FEBRERO" con un numer y un for de 6, pero
despues:

dataFechas.push(data.dataMeses.ENERO);
//resto de meses
//ya con los datos, creamos las graficas

crearGraficas();

}
catch{
console.error('Error al obtener datos:', error);
}
}

```

## Gastos

Para la seccion de los gastos se crea una especie de Google Calendar para poder dar forma a el ingreso de gastos diario por persona, ademas de tambien traer consigo una tabla donde se evidencia cuales son los gastos por mes; dentro de los gastos por mes se puede buscar por gasto, ademas hay una grafica que nos dice en que hemos gastado mas, si en alimentos, transporte, entretenimiento, etc.

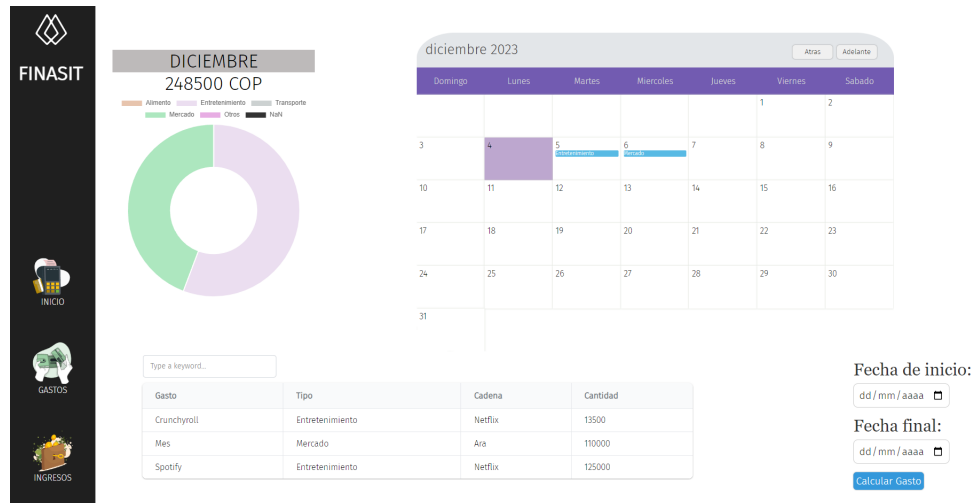


Figura 5: Pagina Inicio.

## Ingresos

En el apartado de ingresos mantenemos una idea mas tradicional y simple debido a que no todos los dias mantenies ingresos constantes, asi pues, contamos solo con una tabla y el correspondiente input de los datos:

Tipo	Fecha	Valor
donaciones	2023-12-21	250000
ventas	2023-12-23	25000
salario	2023-12-09	5000000

Figura 6: Pagina Inicio.

## 4. Conclusiones:

El desarrollo de Finasit, nuestro asistente financiero basado en tecnologías web, ha sido un viaje educativo y enriquecedor que nos ha permitido explorar y aplicar diversos conceptos fundamentales en el ámbito de la programación y desarrollo web.

Sumergirnos en las tecnologías HTML5, CSS3, JavaScript y PHP nos proporcionó las herramientas necesarias para crear una experiencia de usuario dinámica y atractiva. La integración de SQLite3 como sistema de gestión de bases de datos fue fundamental para almacenar y recuperar eficientemente la información relacionada con gastos, ingresos y cuentas.

La implementación de llamadas Fetch y la manipulación asincrónica de datos en JavaScript mejoraron significativamente la eficiencia y la velocidad de carga de la aplicación. Esta integración nos permitió conectarnos a servicios externos y recuperar información dinámica de manera efectiva.

Exploramos y aplicamos el concepto de arquitectura por capas para separar la lógica, la presentación y la persistencia de datos. Esta metodología proporcionó una estructura organizada y mantenible para el desarrollo de Ingper.

Más allá de la programación, nuestro objetivo fue crear una herramienta educativa que empoderara a los usuarios en la gestión de sus finanzas. Finasit

no solo registra transacciones, sino que también brinda a los usuarios una visión clara de sus patrones de gastos e ingresos, fomentando así una mayor conciencia financiera.