



UNIVERSIDAD TÉCNICA DE MANABÍ



**FACULTAD DE CIENCIAS INFORMÁTICA
INGENIERÍA EN SISTEMAS DE LA INFORMACIÓN**

PROGRAMACIÓN AVANZADA

MEDIO CICLO

**TEMA:
PROYECTO**

INTEGRANTES:

Manrique Rezabala Johan.

Pico Mera Armando.

Tigua Avellan Abel.

Sánchez García Jhon.

DOCENTE:

ING. Hernán Vargas.

NIVEL:

6TO

PARA EL: " A "

PERIODO LECTIVO:

MAYO 2021 hasta SEPTIEMBRE del 2021

Contenido

TEMA	3
INTRODUCCION	4
MARCO TEORICO.....	5
APLICACIONES WEB	5
PYTHON.....	5
OPERACIONES CRUD	6
OPERACIONES MYSQL	6
OPERACIONES HTML.....	7
OPERACIONES CSS	7
OPERACIONES BOOSTRAP	7
AUTOMATIZACION DE PROCESOS	8
DESARROLLO DE LA APLICACIÓN.....	8
INDEX.html	12
CLIENTES.html.....	13
LISTADO DE CLIENTES.html	14
INTERFACES DEL SISTEMA.....	15
CONCUSIONES	24
BIBLIOGRAFIA.....	25



TEMA

Sistema web para la automatización de procesos en el control de deudas para organizaciones que manejen la venta de sus productos a crédito y para empresas que sus servicios sean de préstamos económicos.



INTRODUCCION

La era digital en la que pasa el mundo actual ha llevado a millones de usuarios que son propietarios de grandes y pequeñas empresas a lograr que su negocio llegue al estándar de la automatización de procesos, que para estos tienen muchos beneficios como la reducción de costos, y de tiempos.

La eficiencia de las organizaciones en reducción de tiempo es el enfoque principal de muchos gerentes ya que al realizar procesos manualmente en agendas o en programas que no están completamente dirigidos para que cumplan con los requerimientos que estos desean, llevan a consumir mucho tiempo y hacer gastos innecesarios que pueden ser evitados.

Para ello en la rama de la informática existe el desarrollo de aplicaciones ya sean de escritorio, o web que son más utilizadas actualmente, estas permiten manejar con mayor eficiencia este tipo de problemas como demorar realizando los procesos que maneja alguna empresa y gastos económicos fuera del rango.

Las aplicaciones web se desarrollan utilizando un lenguaje de programación y un lenguaje de estilo que beneficie al desarrollador e impacte al gerente dueño de la empresa y estas tienen el beneficio que pueden ser manejadas fácilmente desde cualquier ordenador, de forma que el desarrollador puede hacer cambios instantáneos sin tener que acudir a la empresa para la que está trabajando.

El objetivo de este proyecto es desarrollar una aplicación web que pueda reducir el tiempo que se demora un trabajador haciendo de forma manual los registros en libretas o cuadernos que pueden ser extraviados en cualquier tipo de empresa que realice ventas de cualquier clase de productos y manejen como modalidad de pago la forma a crédito y para organizaciones que sus servicios sean de dar préstamos económicos a distintos usuarios que lo requieran, mediante el lenguaje de programación de Python, con una base de datos desarrollada en Mysql y mejorando la estructura y diseño de interfaces por medio de Bootstrap, html y Css.

Este tipo de aplicación permitirá a la empresa aparte de ser innovadora poder competir de forma más eficiente a su alrededor.

MARCO TEORICO

APLICACIONES WEB

Una aplicación web se puede definir como una aplicación en la cual un usuario por medio de un navegador realiza peticiones a una aplicación remota accesible a través de Internet (o a través de una intranet) y que recibe una respuesta que se muestra en el propio navegador.

En las aplicaciones web suelen distinguirse tres niveles (como en las arquitecturas cliente/servidor de tres niveles): el nivel superior que interacciona con el usuario (el cliente web, normalmente un navegador), el nivel inferior que proporciona los datos (la base de datos) y el nivel intermedio que procesa los datos (el servidor web).

PYTHON

Es un lenguaje de alto nivel ya que contiene implícitas algunas estructuras de datos como listas, diccionarios, conjuntos y tuplas, que permiten realizar algunas tareas complejas en pocas líneas de código y de manera legible

Es un lenguaje de programación versátil multiplataforma y multiparadigma que se destaca por su código legible y limpio. Una de las razones de su éxito es que cuenta con una licencia de código abierto que permite su utilización en cualquier escenario. Esto hace que sea uno de los lenguajes de iniciación de muchos programadores siendo impartido en escuelas y universidades de todo el mundo. Sumado a esto cuenta con grandes compañías que hacen de este un uso intensivo. Tal es el caso de Google, Facebook o Youtube, ya que permite, entre otras de sus características la automatización de procesos y ejecución de tareas en tanto en entorno cliente como servidor.

OPERACIONES CRUD

En programación solemos usar el término CRUD para referirnos a las operaciones básicas que puedes realizar sobre un conjunto de datos y por sus siglas son:

Crearlos, nuevos registros, cuando hablamos de bases de datos esto quiere decir insertar información.

Leerlos, r por Read, esto quiere decir consultar esa información, ya sea un registro o una colección de estos registros.

Actualizarlos, u por Update, que significa tomar un registro que ya existe en la base de datos y modificar alguna de las columnas.

Delete, que significa tomar un registro y quitarlo del almacén.

Los CRUDs son programas que involucren todas estas operaciones sobre una entidad, como en nuestro caso en el que trabajaremos con clientes que harán uso de servicios o compra de algún producto de forma a crédito, de deudas que se genera después de recibir el servicio de préstamos o compra de algún producto y pagos de los clientes que realizaran.

OPERACIONES MYSQL

Es el lenguaje de programación que le ayuda a acceder y administrar datos en una base de datos relacional. Cuando se trata de almacenar datos en una base de datos, hay distintos enfoques que usted puede utilizar. MySQL opta por un enfoque llamado una base de datos relacional.

Con una base de datos relacional, sus datos son fragmentados en varias áreas de almacenamiento separadas llamadas tablas en lugar de poner todo junto en una gran unidad de almacenamiento.

OPERACIONES HTML

Html por sus siglas en inglés, HyperText Markup Language, es un lenguaje de marcado que nos permite hacer la estructura de una página web. Cabe aclarar que HTML no es un lenguaje de programación, pues carece de mecanismos para ejecutar cálculos, repeticiones o condiciones, lo que sí es, es un lenguaje de marcado. Es un lenguaje que le indicará al navegador web qué tipo de elemento es el que está en la estructura de una página web.

OPERACIONES CSS

CSS siglas en inglés de Cascading Style Sheets, en español "Hojas de estilo en cascada", un lenguaje de marcas enfocado a definir, crear y mejorar la presentación de un documento basado en HTML. Para muchos diseñadores gráficos CSS significó la puerta de entrada al mundo de la web y junto con otras tecnologías como JavaScript, CSS se ha ido imponiendo como uno de los pilares imprescindibles de la web de hoy en día.

OPERACIONES BOOTSTRAP

Bootstrap es un framework CSS desarrollado por Twitter en 2010, para estandarizar las herramientas de la compañía. El framework combina CSS y JavaScript para estilizar los elementos de una página HTML. Permite mucho más que, simplemente, cambiar el color de los botones y los enlaces.

Esta es una herramienta que proporciona interactividad en la página, por lo que ofrece una serie de componentes que facilitan la comunicación con el usuario, como menús de navegación, controles de página, barras de progreso y más. Además de todas las características que ofrece el framework, su principal objetivo es permitir la construcción de sitios web responsive para dispositivos móviles. Esto significa que las páginas están diseñadas para funcionar en desktop, tabletas y Smartphone, de una manera muy simple y organizada

AUTOMATIZACION DE PROCESOS

La automatización de procesos de TI se refiere a la capacidad de un sistema tecnológico para ejecutar una serie de tareas que originalmente son realizadas por seres humanos. Dicha automatización también controla; corrige y hace visible el estado de los flujos de trabajo y tareas; y genera reportes de todo el proceso.

Un aspecto muy importante de la automatización de procesos es la realimentación. A través de ella, el sistema evalúa; compara y hace correctivos en tiempo real; bajo ciertas restricciones pre configuradas y sin intervención humana.

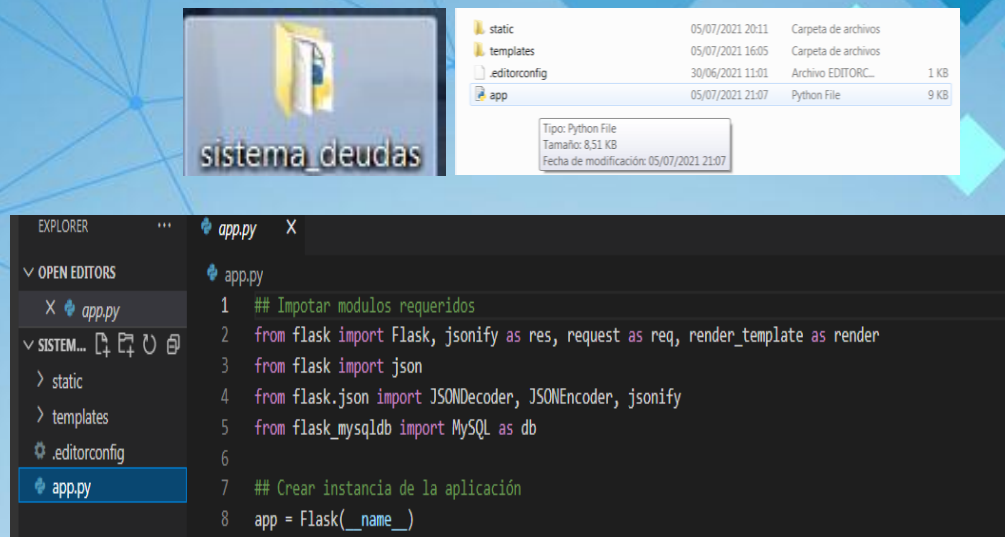
Adicionalmente, la automatización de procesos se puede programar para que se ejecute en determinado momento; dependiendo del resultado de procesos previos o de determinados factores. Esto permite la planificación de tareas a futuro.

DESARROLLO DE LA APLICACIÓN

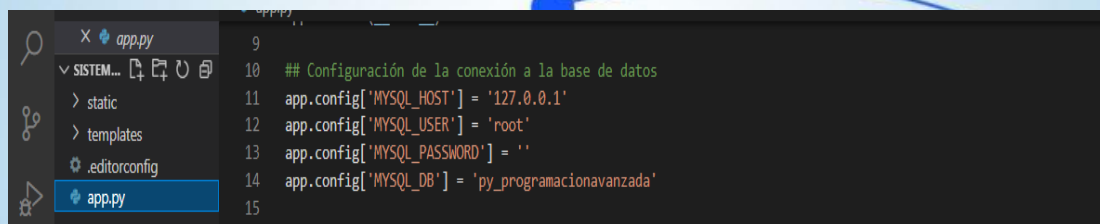
Para el desarrollo de este sistema, usaremos el lenguaje de Programación Python junto con el sistema de gestión de bases de datos MySQL, usamos como editor de código el Visual Studio Code donde empezamos a desarrollar.

Creamos una carpeta en el escritorio denominado sistema_deudas, arrastramos a nuestro editor de código y se nos generará el entorno de trabajo, dentro de nuestro Visual Studio Code, ya debemos tener nuestro Python correctamente instalado de igual forma el MySQL en nuestra PC ya que este será usado como nuestro servidor.

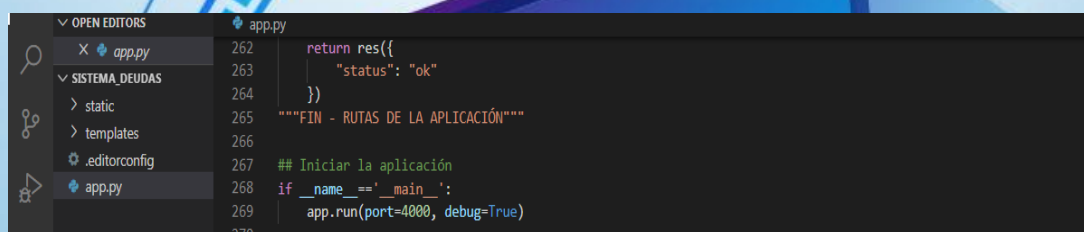
Creamos un archivo denominado App.py que será nuestro archivo principal que correrá nuestro proyecto, es aquí donde vamos a exportar los diferentes módulos que nos permitirán desarrollar nuestro sistema entre éstos está el framework Flask que es el que vamos a utilizar para la creación del sistema.



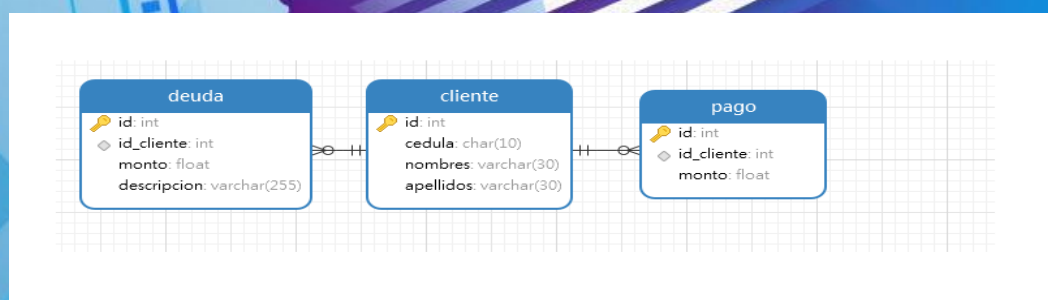
Para la siguiente sección dentro de nuestro archivo App.py configuramos que nuestra aplicación tenga una conexión al servidor, consta del usuario y contraseña del Sistema de gestión de BD y en este caso optamos por usar el host básico para la realización de proyectos (localhost), así mismo definimos el nombre con el cual creamos nuestra Base de Datos en el SGBD.



Con este parámetro vamos a poder ejecutar nuestro servidor



En el siguiente diagrama se muestra como estará estructura la base de datos de la aplicación mediante sus entidades en las tablas que utilizaremos, los atributos y tipo de datos que les pertenecen a cada una y como están relacionadas entre si

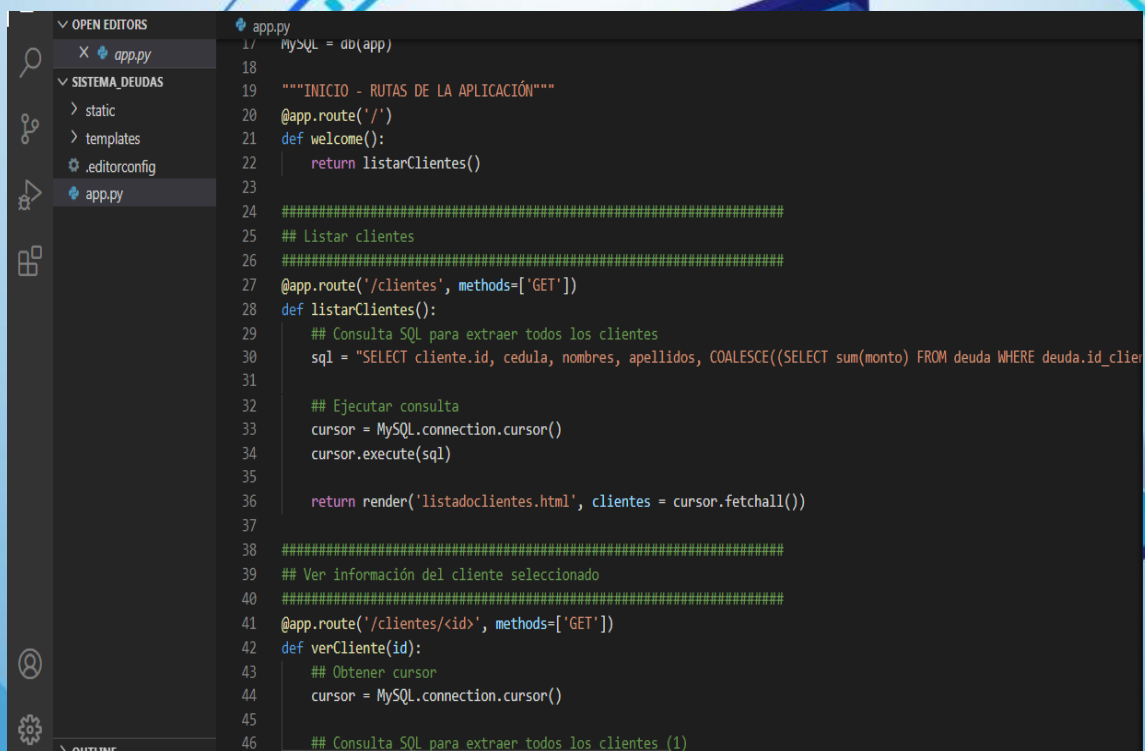


Proseguimos con la creación de las tablas y atributos para nuestra base de datos Mysql

```
1 CREATE DATABASE IF NOT EXISTS py_programacionavanzada;
2 use py_programacionavanzada;
3
4 DROP TABLE IF EXISTS cliente;
5 CREATE TABLE `cliente` (
6     id INT PRIMARY KEY AUTO_INCREMENT,
7     cedula CHAR(10) NOT NULL,
8     nombres VARCHAR(30) NOT NULL,
9     apellidos VARCHAR(30) NOT NULL
10 ) Engine=InnoDB AUTO_INCREMENT = 1;
```

```
12 DROP TABLE IF EXISTS deuda;
13 CREATE TABLE `deuda` (
14     id INT PRIMARY KEY AUTO_INCREMENT,
15     id_cliente INT NOT NULL,
16     monto FLOAT NOT NULL,
17     descripcion VARCHAR(255) NOT NULL,
18     FOREIGN KEY (id_cliente) REFERENCES cliente(id)
19 ) Engine=InnoDB AUTO_INCREMENT = 1;
20
21 DROP TABLE IF EXISTS pago;
22 CREATE TABLE `pago` (
23     id INT PRIMARY KEY AUTO_INCREMENT,
24     id_cliente INT NOT NULL,
25     monto FLOAT NOT NULL,
26     FOREIGN KEY (id_cliente) REFERENCES cliente(id)
27 ) Engine=InnoDB AUTO_INCREMENT = 1;
```

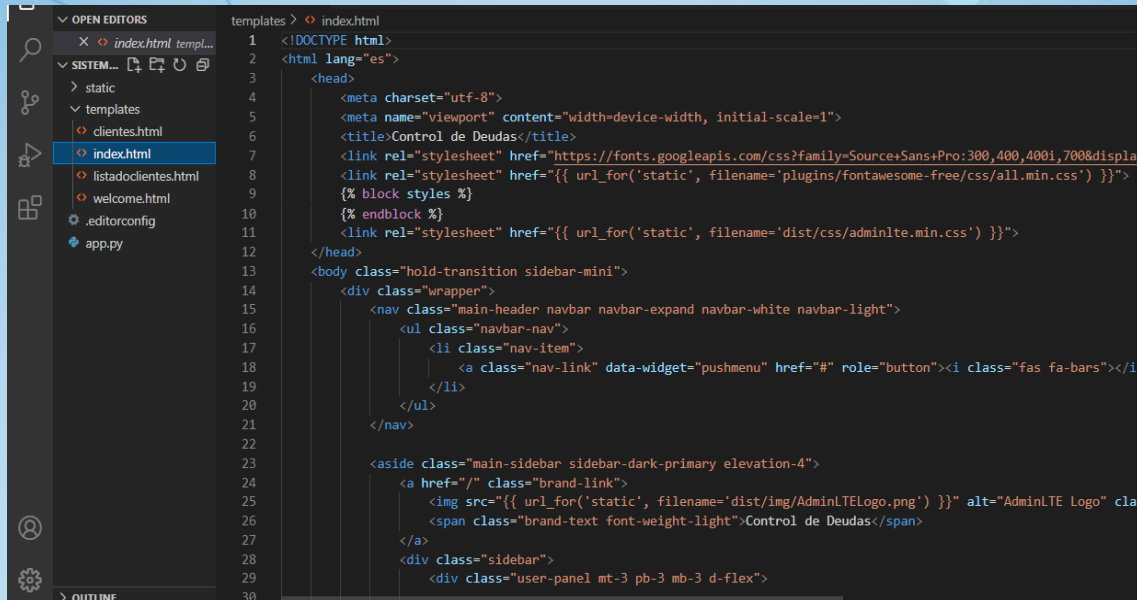
Para poder establecer nuestras rutas de conexión, las creamos de la siguiente manera estableciendo funciones que aceptarán nuestra petición, en este caso ya tenemos creado un archivo HTML que alberga el en maquetado principal de nuestro sistema.



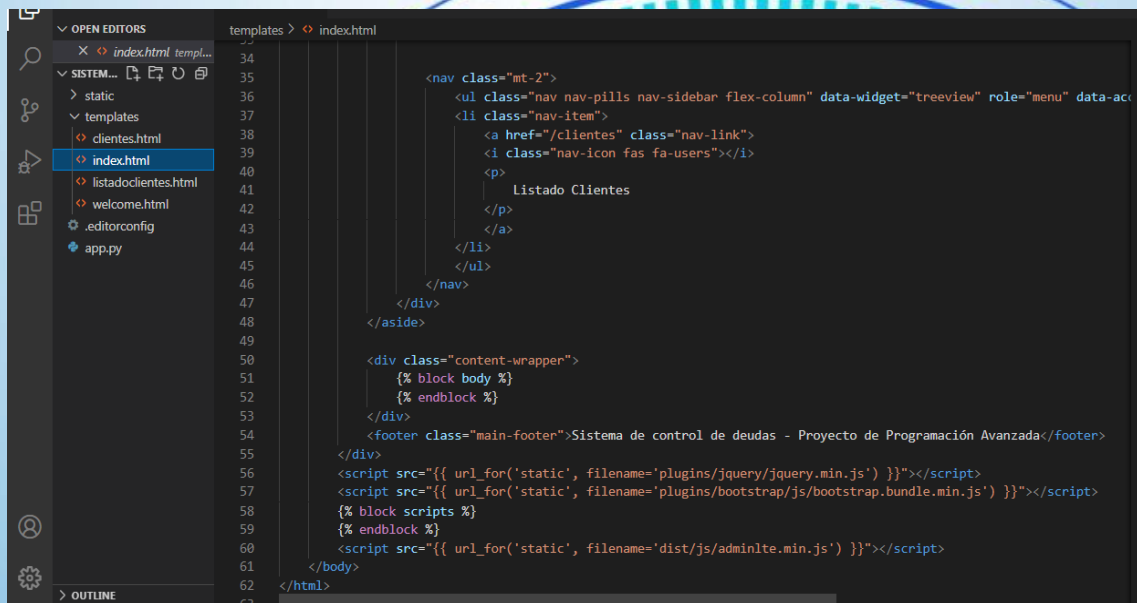
```
app.py
1 MySQL = db(app)
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19 """INICIO - RUTAS DE LA APLICACIÓN"""
20 @app.route('/')
21 def welcome():
22     return listarClientes()
23
24 #####
25 ## Listar clientes
26 #####
27 @app.route('/clientes', methods=['GET'])
28 def listarClientes():
29     ## Consulta SQL para extraer todos los clientes
30     sql = "SELECT cliente.id, cedula, nombres, apellidos, COALESCE((SELECT sum(monto) FROM deuda WHERE deuda.id_cliente = cliente.id), 0) as monto"
31
32     ## Ejecutar consulta
33     cursor = MySQL.connection.cursor()
34     cursor.execute(sql)
35
36     return render('listadoclientes.html', clientes = cursor.fetchall())
37
38 #####
39 ## Ver información del cliente seleccionado
40 #####
41 @app.route('/clientes/<id>', methods=['GET'])
42 def verCliente(id):
43     ## Obtener cursor
44     cursor = MySQL.connection.cursor()
45
46     ## Consulta SQL para extraer todos los clientes (1)
```


Dentro del Index, será nuestra portada de presentación del sistema. Este mismo proceso lo realizamos para los otros archivos html que albergan el código del en maquetado de nuestro sistema dentro de la carpeta templates.

INDEX.html



```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4 <meta charset="utf-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1">
6 <title>Control de Deudas</title>
7 <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&display=block">
8 <link rel="stylesheet" href="{{ url_for('static', filename='plugins/fontawesome-free/css/all.min.css') }}">
9 {% block styles %}
10 {% endblock %}
11 <link rel="stylesheet" href="{{ url_for('static', filename='dist/css/adminlte.min.css') }}">
12 </head>
13 <body class="hold-transition sidebar-mini">
14 <div class="wrapper">
15 <nav class="main-header navbar navbar-expand navbar-white navbar-light">
16 <ul class="navbar-nav">
17 <li class="nav-item">
18 <a class="nav-link" data-widget="pushmenu" href="#" role="button"><i class="fas fa-bars"></i>
19 </li>
20 </ul>
21 </nav>
22
23 <aside class="main-sidebar sidebar-dark-primary elevation-4">
24 <a href="/" class="brand-link">
25 Control de Deudas</span>
27 </a>
28 <div class="sidebar">
29 <div class="user-panel mt-3 pb-3 mb-3 d-flex">
```



```
34
35 <nav class="mt-2">
36 <ul class="nav nav-pills nav-sidebar flex-column" data-widget="treeview" role="menu" data-acc
37 <li class="nav-item">
38 <a href="/clientes" class="nav-link">
39 <i class="nav-icon fas fa-users"></i>
40 <p>
41 Listado Clientes
42 </p>
43 </a>
44 </li>
45 </ul>
46 </nav>
47 </div>
48 </aside>
49
50 <div class="content-wrapper">
51 {% block body %}
52 {% endblock %}
53 </div>
54 <footer class="main-footer">Sistema de control de deudas - Proyecto de Programación Avanzada</footer>
55 </div>
56 <script src="{{ url_for('static', filename='plugins/jquery/jquery.min.js') }}"></script>
57 <script src="{{ url_for('static', filename='plugins/bootstrap/js/bootstrap.bundle.min.js') }}"></script>
58 {% block scripts %}
59 {% endblock %}
60 <script src="{{ url_for('static', filename='dist/js/adminlte.min.js') }}"></script>
61 </body>
62 </html>
```

CLIENTES.html

```
templates > clientes.html
1  {% extends "index.html" %}
2
3  {% block styles %}
4  <link rel="stylesheet" href="{{ url_for('static', filename='plugins/datatables-bs4/css/dataTables.bootstrap4.min.css'
5  <link rel="stylesheet" href="{{ url_for('static', filename='plugins/datatables-responsive/css/responsive.bootstrap4.r
6  <link rel="stylesheet" href="{{ url_for('static', filename='plugins/datatables-buttons/css/buttons.bootstrap4.min.cs
7  {% endblock %}
8
9  {% block body %}
10 <section class="content-header">
11 <section class="content-header">
12 <div class="container-fluid">
13 <div class="row mb-2">
14 <div class="col-sm-6">
15 <h1>{{data.cliente.2}} {{data.cliente.3}}</h1>
16 </div>
17 </div>
18 </div>
19 </section>
20 <section class="content">
21 <div class="col-12 col-sm-12">
22 <div class="card card-primary card-outline card-outline-tabs">
23 <div class="card-header p-0 border-bottom-0">
24 <ul class="nav nav-tabs" id="custom-tabs-four-tab" role="tablist">
25 <li class="nav-item">
26 <a class="nav-link active" id="custom-tabs-four-home-tab" data-toggle="pill" href="#custom-tabs-
27 </li>
28 <li class="nav-item">
29 <a class="nav-link" id="custom-tabs-four-profile-tab" data-toggle="pill" href="#custom-tabs-four-
30 </li>
```

```
templates > clientes.html
373 deudaSeleccionada = 0
374 pagoSeleccionado = 0
375
376 function aplicarDataTable(name){
377     $(name).DataTable({
378         "paging": true,
379         "lengthChange": true,
380         "searching": true,
381         "ordering": true,
382         "info": true,
383         "autoWidth": false,
384         "responsive": true,
385     });
386 }
387
388 aplicarDataTable('#tblHistorialDeudas')
389 aplicarDataTable('#tblHistorialPagos')
390
391 function aplicarCambiosCliente(id){
392     $.ajax({
393         url: "/clientes2/{{data.cliente.0}}",
394         method: 'POST',
395         contentType: 'application/json',
396         data: JSON.stringify({
397             "cedula": document.getElementById("inputNumeroCedula").value,
398             "nombres": document.getElementById("inputNombres").value,
399             "apellidos": document.getElementById("inputApellidos").value,
400         }),
401         success: function(data){
402             window.location.reload()
```

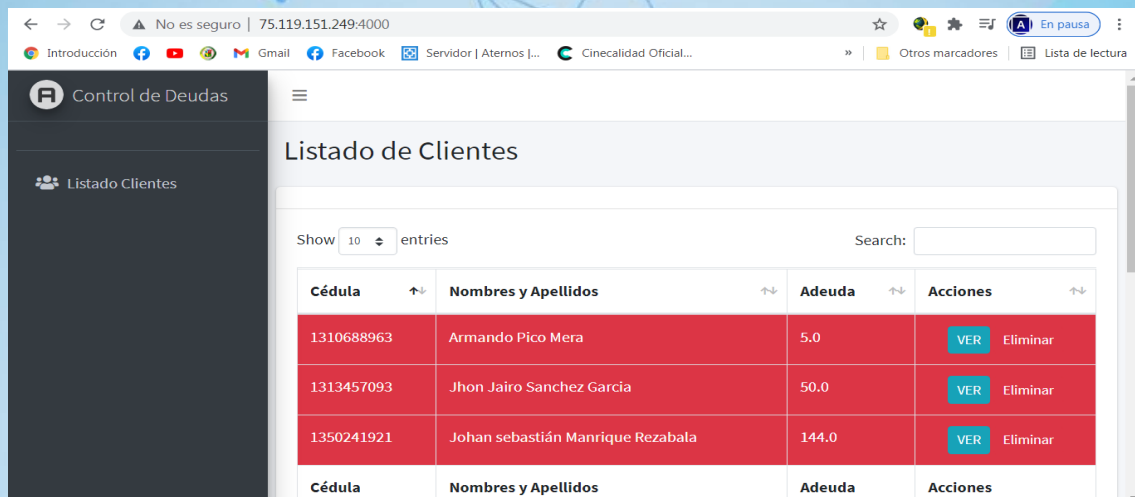

LISTADO DE CLIENTES.html

```
templates > listadoclientes.html
1 {% extends "index.html" %}
2
3 {% block styles %}
4 <link rel="stylesheet" href="{{ url_for('static', filename='plugins/datatables-bs4/css/dataTables.bootstrap4.min.css'
5 <link rel="stylesheet" href="{{ url_for('static', filename='plugins/datatables-responsive/css/responsive.bootstrap4.r
6 <link rel="stylesheet" href="{{ url_for('static', filename='plugins/datatables-buttons/css/buttons.bootstrap4.min.cs
7 {% endblock %}
8
9 {% block body %}
10 <section class="content-header">
11   <div class="container-fluid">
12     <div class="row mb-2">
13       <div class="col-sm-6">
14         <h1>Listado de Clientes</h1>
15       </div>
16     </div>
17   </div>
18 </section>
19
20 <section class="content">
21   <div class="card">
22     <div class="card-header">
23     </div>
24     <div class="card-body">
25       <table id="example2" class="table table-bordered table-hover">
26         <thead>
27           <tr>
28             <th>Cédula</th>
29             <th>Nombres y Apellidos</th>
30             <th>Adeuda</th>
```

```
templates > listadoclientes.html
147 clienteSeleccionado = id
148 $("#deuda-modal-eliminar").modal("show")
149 }
150
151 function eliminarCliente(){
152   $.ajax({
153     url: "/clientes/"+clienteSeleccionado,
154     method: 'DELETE',
155     success: function(data){
156       window.location.reload()
157     },
158     error: function(er) {
159       console.log(er)
160     }
161   })
162 }
163
164 function crearCliente(){
165   $.ajax({
166     url: "/clientes",
167     method: 'POST',
168     contentType: 'application/json',
169     data: JSON.stringify({
170       "cedula": document.getElementById("inputDNI").value,
171       "nombres": document.getElementById("inputNAME").value,
172       "apellidos": document.getElementById("inputLASTNAME").value,
173     }),
174     success: function(data){
175       window.location.reload()
176     },
177   })
178 }
```

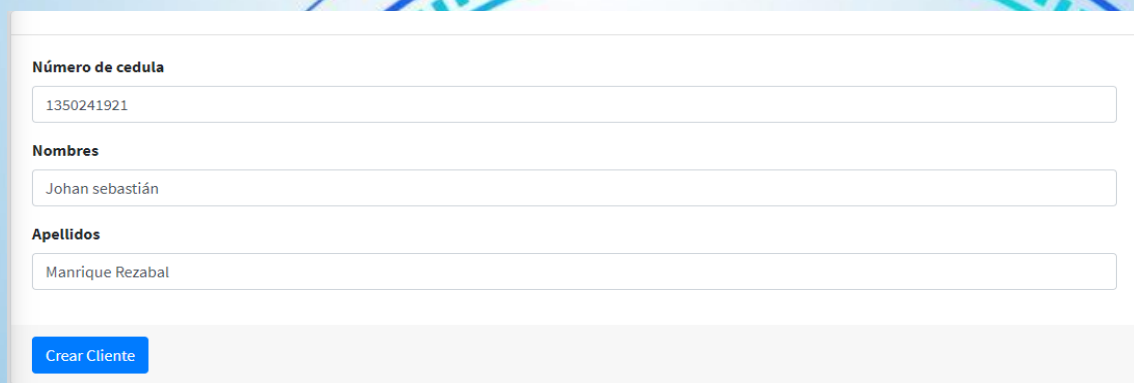
INTERFACES DEL SISTEMA

Esta es la vista principal, una presentación referente que describe al sistema como tal y un menú con acceso a las diferentes entidades que se desarrollaron. Por último, nos vamos a enfocar a explicar una parte de cómo funciona nuestra interfaz que desarrollamos para mejorar la visualización de nuestra aplicación.



Cédula	Nombres y Apellidos	Adeuda	Acciones
1310688963	Armando Pico Mera	5.0	VER Eliminar
1313457093	Jhon Jairo Sanchez García	50.0	VER Eliminar
1350241921	Johan Sebastián Manrique Rezabala	144.0	VER Eliminar

La interfaz muestra la creación de un cliente nuevo



Número de cedula
1350241921

Nombres
Johan Sebastián

Apellidos
Manrique Rezabal

[Crear Cliente](#)

Listado de clientes, el cliente que acabamos de crear le falta una “a” al final en el apellido.

Listado de Clientes

Show entries

Search:

Cédula	Nombres y Apellidos	Adeuda	Acciones
1310688963	Armando Pico Mera	20.0	VER Eliminar
1313457093	Jhon Jairo Sanchez García	50.0	VER Eliminar
1350241921	Johan sebastián Manrique Rezabal	0.0	VER Eliminar
Cédula	Nombres y Apellidos	Adeuda	Acciones

Showing 1 to 3 of 3 entries

[Previous](#) [1](#) [Next](#)

Podemos corregir el apellido del cliente.

Johan sebastián Manrique Rezabal

[Información Personal](#) [Historial de Deudas](#) [Historial de Pagos](#) [Registrar Pago](#) [Registrar Deuda](#)

Número de cedula

Nombres

Apellidos

[Guardar Cambios](#)

Luego ya muestra el cliente con el apellido ya corregido

Listado de Clientes

Show entries

Search:

Cédula	Nombres y Apellidos	Adeuda	Acciones
1310688963	Armando Pico Mera	20.0	VER Eliminar
1313457093	Jhon Jairo Sanchez García	50.0	VER Eliminar
1350241921	Johan sebastián Manrique Rezabala	0.0	VER Eliminar
Cédula	Nombres y Apellidos	Adeuda	Acciones

Showing 1 to 3 of 3 entries

[Previous](#) [1](#) [Next](#)

Historial de deudas del cliente creado.

Johan Sebastián Manrique Rezabala

Información Personal

Historial de Deudas

Historial de Pagos

Registrar Pago

Registrar Deuda

Show 10 entries

Search:

Monto	Descripción	Acciones
No data available in table		
Monto	Descripción	Acciones

Showing 0 to 0 of 0 entries

Previous

Next

Historial de pagos del cliente creado.

Johan Sebastián Manrique Rezabala

Información Personal

Historial de Deudas

Historial de Pagos

Registrar Pago

Registrar Deuda

Show 10 entries

Search:

Monto	Acciones
No data available in table	
Monto	Acciones

Showing 0 to 0 of 0 entries

Previous

Next

Registrando unas deudas al cliente.

Johan Sebastián Manrique Rezabala

Información Personal

Historial de Deudas

Historial de Pagos

Registrar Pago

Registrar Deuda

Monto

600

Descripción

Compra de: Procesador Intel Core i7-11700 11va Gen

Guardar Cambios

Johan Sebastián Manrique Rezabala

[Información Personal](#)[Historial de Deudas](#)[Historial de Pagos](#)[Registrar Pago](#)[Registrar Deuda](#)**Monto**

500

Descripción

Compra de: Tarjeta de video Gigabyte GeForce GT 710 2GB 64-Bit DDR5

[Guardar Cambios](#)[Información Personal](#)[Historial de Deudas](#)[Historial de Pagos](#)[Registrar Pago](#)[Registrar Deuda](#)**Monto**

75

Descripción

Compra de: RAM Adata 8GB DDR4 2666MHZ DIMM AD4U266688G19-S

[Guardar Cambios](#)

Historial de deudas del cliente actualizado con las nuevas deudas ingresadas.

Johan Sebastián Manrique Rezabala

[Información Personal](#)[Historial de Deudas](#)[Historial de Pagos](#)[Registrar Pago](#)[Registrar Deuda](#)Show entriesSearch:

Monto	↑↓	Descripción	↕	Acciones	↕
75.0		Compra de: RAM Adata 8GB DDR4 2666MHZ DIMM AD4U266688G19-S		Editar	Eliminar
500.0		Compra de: Tarjeta de video Gigabyte GeForce GT 710 2GB 64-Bit DDR5		Editar	Eliminar
600.0		Compra de: Procesador Intel Core I7-11700 11va Gen		Editar	Eliminar
Monto		Descripción		Acciones	

Showing 1 to 3 of 3 entries

[Previous](#) [1](#) [Next](#)

Proceso a crear una deuda que voy a eliminar.

Johan Sebastián Manrique Rezabala

Información Personal Historial de Deudas Historial de Pagos Registrar Pago Registrar Deuda

Monto

2000

Descripción

Ejemplo de deuda a eliminar

Guardar Cambios

Johan Sebastián Manrique Rezabala

Información Personal

Show 10 entries

Monto		Acciones
75.0	Compra de: RAM Adata 8GB DDR4 2666MHZ DIMM AD4U266688G19-S	<button>Editar</button> <button>Eliminar</button>
500.0	Compra de: Tarjeta de video Gigabyte GeForce GT 710 2GB 64-Bit DDR5	<button>Editar</button> <button>Eliminar</button>
600.0	Compra de: Procesador Intel Core I7-11700 11va Gen	<button>Editar</button> <button>Eliminar</button>
2000.0	Ejemplo de deuda a eliminar	<button>Editar</button> <button>Eliminar</button>
Monto	Descripción	Acciones

Showing 1 to 4 of 4 entries

Previous 1 Next

Sistema de control de deudas - Proyecto de Programación Avanzada

Johan Sebastián Manrique Rezabala

Información Personal Historial de Deudas Historial de Pagos Registrar Pago Registrar Deuda

Show 10 entries

Search:

Monto	Descripción	Acciones
75.0	Compra de: RAM Adata 8GB DDR4 2666MHZ DIMM AD4U266688G19-S	<button>Editar</button> <button>Eliminar</button>
500.0	Compra de: Tarjeta de video Gigabyte GeForce GT 710 2GB 64-Bit DDR5	<button>Editar</button> <button>Eliminar</button>
600.0	Compra de: Procesador Intel Core I7-11700 11va Gen	<button>Editar</button> <button>Eliminar</button>
Monto	Descripción	Acciones

Showing 1 to 3 of 3 entries

Previous 1 Next

Actualizando el precio de la compra de RAM.

Johan sebas

Información Personal

Show 10 entries

Monto	
75.0	
500.0	
600.0	
Monto	

Showing 1 to 3 of 3 entries

Actualizar de Deuda

Código

7

Monto a Pagar

85

Descripción

Compra de: RAM Adata 8GB DDR4 2666MHZ DIMM AD4U2666

No

Guardar Cambios

Registrar Deuda

Search:

Acciones

Editar

Eliminar

Editar

Eliminar

Editar

Eliminar

Acciones

Previous

1

Next

Sistema de control de deudas - Proyecto de Programación Avanzada

Johan Sebastián Manrique Rezabala

Información Personal

Historial de Deudas

Historial de Pagos

Registrar Pago

Registrar Deuda

Show 10 entries

Search:

Monto	Descripción	Acciones
85.0	Compra de: RAM Adata 8GB DDR4 2666MHZ DIMM AD4U26668G19-S	<div>Editar</div> <div>Eliminar</div>
500.0	Compra de: Tarjeta de video Gigabyte GeForce GT 710 2GB 64-Bit DDR5	<div>Editar</div> <div>Eliminar</div>
600.0	Compra de: Procesador Intel Core I7-11700 11va Gen	<div>Editar</div> <div>Eliminar</div>
Monto	Descripción	Acciones

Showing 1 to 3 of 3 entries

Previous

1

Next

Historial de pagos del cliente.

Johan Sebastián Manrique Rezabala

Información Personal

Historial de Deudas

Historial de Pagos

Registrar Pago

Registrar Deuda

Show 10 entries

Search:

Monto	Acciones
No data available in table	
Monto	Acciones

Showing 0 to 0 of 0 entries

Previous

Next

Registrando un pago a nuestro cliente, en el campo “Adeuda” se muestra el valor que el cliente debe cancelar, no es obligatorio pagar toda la cantidad, puede realizar pequeños aportes.

Johan Sebastián Manrique Rezabala

Información Personal Historial de Deudas Historial de Pagos Registrar Pago Registrar Deuda

Adeuda

1185.0

Monto a Pagar

125

Verificar Pago

Historial de pagos actualizado.

Johan Sebastián Manrique Rezabala

Información Personal Historial de Deudas Historial de Pagos Registrar Pago Registrar Deuda

Show 10 entries Search:

Monto	Acciones
125.0	Editar Eliminar
576.0	Editar Eliminar
Monto	Acciones

Showing 1 to 2 of 2 entries Previous 1 Next

Modificando el pago de “25” a “50”.

Johan Sebastián Manrique Rezabala

Información Personal Historial de Deudas Historial de Pagos Registrar Pago Registrar Deuda

Show 10 entries Search:

Monto	Acciones
25.0	Editar Eliminar
125.0	Editar Eliminar
576.0	Editar Eliminar
Monto	Acciones

Showing 1 to 3 of 3 entries Previous 1 Next

Actualización del Pago

Código

7

Monto a Pagar

25.0

No Guardar Cambios

Historial actualizado.

Johan Sebastián Manrique Rezabala

Información Personal

Historial de Deudas

Historial de Pagos

Registrar Pago

Registrar Deuda

Show 10 entries

Search:

Monto	Acciones
50.0	<div>EditarEliminar</div>
125.0	<div>EditarEliminar</div>
576.0	<div>EditarEliminar</div>
Monto	Acciones

Showing 1 to 3 of 3 entries

Previous

1

Next

Eliminando el pago de 125.

Johan Sebastián Manrique Rezabala

Información Personal

Historial de Deudas

Historial de Pagos

Registrar Pago

Registrar Deuda

Show 10 entries

Search:

Monto	Acciones
50.0	<div>EditarEliminar</div>
125.0	<div>EditarEliminar</div>
576.0	<div>EditarEliminar</div>
Monto	Acciones

Showing 1 to 3 of 3 entries

Previous

1

Next

Confirmación de eliminación de pago

¿Esta seguro de eliminar el pago?

No

Si

Historial actualizado.

Johan Sebastián Manrique Rezabala

Información Personal

Historial de Deudas

Historial de Pagos

Registrar Pago

Registrar Deuda

Show 10 entries

Search:

Monto	Acciones
50.0	<div>EditarEliminar</div>
576.0	<div>EditarEliminar</div>
Monto	Acciones

Showing 1 to 2 of 2 entries

Previous

1

Next

Listado de clientes actualizado.

Show 10 entries

Search:

Cédula	Nombres y Apellidos	Adeuda	Acciones
1310688963	Armando Pico Mera	20.0	VER Eliminar
1313457093	Jhon Jairo Sanchez Garcia	50.0	VER Eliminar
1350241921	Johan sebastián Manrique Rezabala	559.0	VER Eliminar
Cédula	Nombres y Apellidos	Adeuda	Acciones

Showing 1 to 3 of 3 entries

[Previous](#) [1](#) [Next](#)



CONCLUSIONES

El desarrollo de la aplicación web influye satisfactoriamente en la automatización y agilización de los procesos y control de una empresa prestamista; ya sea cobros, abonos y préstamos; cumpliendo y satisfaciendo los indicadores de calidad planteados en los objetivos del presente trabajo de investigación:

La funcionalidad de la aplicación web desarrollada, además de la eficiencia y usabilidad, influyen satisfactoriamente en la automatización y agilización de los procesos y control de una empresa prestamista; ya sea cobros, abonos y préstamos, características que definen y describen a este indicador de calidad.

La confiabilidad de la aplicación web desarrollada, influye satisfactoriamente en la automatización y agilización de los procesos y control de una empresa prestamista; ya sea cobros, abonos y préstamos; debido a la satisfacción de los usuarios al tener un sistema sin errores, características que definen y describen a este indicador de calidad.



BIBLIOGRAFIA

LUJÁN MORA, Sergio. Programación de aplicaciones web: Recuperado de <https://sergiolujanmora.es/programacion-aplicaciones-web-historia-principios-basicos-clientes-web>

Henríquez Taboada, H., & Arias Muñoz, M. A. (Agosto de 2018). Recuperado de <http://repositorio.uigv.edu.pe/bitstream/handle/20.500.11818/2930/TESIS-MARCO%20ANTONIO%20ARIAS%20MU%C3%91OZ.pdf?sequence=2&isAllowed=y>

Python Flask y Mysql Aplicación Web | Aplicación de Ejemplo
<https://www.youtube.com/watch?v=IgCfZkR8wME&t=20s>

