

**Instituto Técnico Ricaldone**



**Desarrollo del software**

**Documento de Manual Técnico para Sistema de Gestión para Clínicas Dentales.**

**Docente: Emerson González**

**Sección: 2-A**

**Alumnos:**

- **Jason Steven Miranda Navarrete.**
- **Tania Isabel Ramírez Castellón.**
- **Raúl Ernesto Mena Quito.**
- **Diego Alejandro Palacios Burgos.**

<b>1. Introducción:</b>	<b>3</b>
<b>2. Tecnologías y herramientas:</b>	<b>5</b>
<b>3. Estructura de la base de datos:</b>	<b>6</b>
Modelo Relacional	6
Script de la base de datos	6
<b>4. Diccionario de datos</b>	<b>7</b>
<b>5. Arquitectura de software</b>	<b>8</b>
<b>6. Estructura del proyecto</b>	<b>9</b>
Estructura de Modelos	9
Estructura de Vistas	11
<b>7. Diseño de la aplicación</b>	<b>12</b>
Paleta de colores	12
Tipos y tamaños de fuentes	17
Diagramas	19
<b>8. Buenas prácticas de desarrollo</b>	<b>24</b>
<b>9. Requerimientos de hardware y software</b>	<b>27</b>
Requerimientos del Hardware:	27
Requisitos de Software:	27
<b>10. Instalación y configuración:</b>	<b>28</b>

## **1. Introducción:**

El presente documento de manual técnico está dirigido para el administrador de este sistema de gestión. El objetivo de este documento es proporcionar una guía clara y detallada para la solución de problemas, operación, instalación y mantenimiento del Sistema de Gestión para Clínicas Dentales. Esto está dirigido para el administrador y personal que interactúan directamente con el sistema.

Este documento incluye procedimientos paso a paso, especificaciones técnicas y mejores prácticas para asegurar el uso eficiente y seguro del sistema.

El contenido ha sido organizado de manera estructurada y lógica para que se faciliten las consultas, permitiendo así, una rápida localización de la información necesaria. Se recomienda leer este manual detenida y completamente antes de comenzar cualquier tipo de intervención técnica.

En estas páginas encontrará explicaciones paso a paso, ilustraciones y recomendaciones útiles para que puedan aprovechar al máximo todas sus funciones sin complicaciones. También incluye una sección de resolución de problemas y errores comunes, que le permitirá solucionar rápidamente situaciones que puedan presentarse a la hora del uso normal

El manual está organizado de manera sencilla para que pueda consultar solamente la parte que necesite en cada momento. Entre ellos estarán:

- Detalles técnicos
- Requisitos técnicos
- Información de seguridad
- Diagramas y fotos
- Instrucciones de uso
- Guías de mantenimiento y reparación

Esto está pensado como un apoyo constante durante la vida útil del sistema, por lo que se recomienda tener este manual a la mano o accesible para consultarlo siempre cuando surjan dudas. El seguimiento de estas instrucciones es para garantizar un mejor

rendimiento y que también contribuye a la seguridad del usuario y maximizar la optimización de los recursos que ofrece el sistema.

Como conclusión para esta introducción, este documento constituye una herramienta fundamental para el uso responsable y eficiente del sistema. Se le invita al usuario a familiarizarse con cada sección antes de comenzar a operar con el equipo y a utilizar este manual como referencia cada vez que sea necesario.

## 2. Tecnologías y herramientas

- Visual Studio c#.

**Uso:** Utilizado para la creación de los formularios y conexión con la base de datos.

- SQL Server Management Studio 21.

**Uso:** Utilizado para la creación de la base de datos.

- .NET framework 4.8.

**Uso:** utilizado en la creación del código debido a la cantidad de herramientas que proporciona para construir y ejecutar aplicaciones.

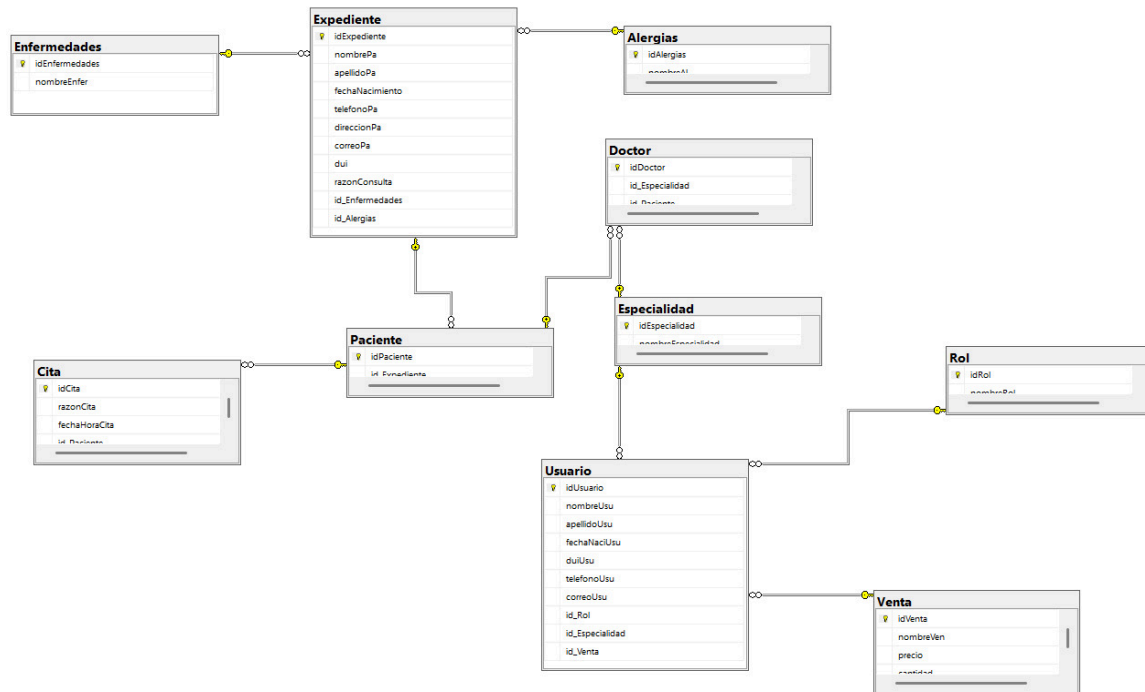
- Bcrypt

**Uso:** Utilizado para la encriptación de la contraseña de los usuarios

### 3. Estructura de la base de datos:

#### Modelo Relacional.

*Estructura de la base de datos:*



**Nota:** Se muestra el diagrama general

#### Script de la base de datos.

En el siguiente enlace se podrá encontrar el script de la base de datos con triggers, procedimientos almacenados y estructuras de las tablas.

[https://drive.google.com/file/d/1hljDA6EDHVBK7MKPI1XVO3j-LjxGUb-f/view?usp=sh  
aring](https://drive.google.com/file/d/1hljDA6EDHVBK7MKPI1XVO3j-LjxGUb-f/view?usp=sharing)

#### **4. Diccionario de datos**

En el siguiente enlace se encontrará el diccionario de datos de la base de datos.

<https://docs.google.com/spreadsheets/d/1BwgK-y5vDthOCJpr9no19WxfWAjjv2QjFtBOnhonYvY/edit?usp=sharing>

## **5. Arquitectura de software**

### **Modelos.**

Dentro de la carpeta modelos se encuentran los métodos que harán funcionar el programa, en esta carpeta se encapsularon los campos en privado de las tablas de la base de datos. A través de un método se realiza la conexión con la base de datos en SQL, y a través de este método se realizan otros métodos que realizan las queries según sea la petición recibida.

### **Vistas**

Este apartado es un apartado visual con el que interactúa el cliente directamente a través de los formularios, estos formularios tienen conexión con la base de datos en SQL utilizando y llamando los métodos creados en el apartado de Modelos.

En el siguiente enlace se presentará el repositorio del proyecto

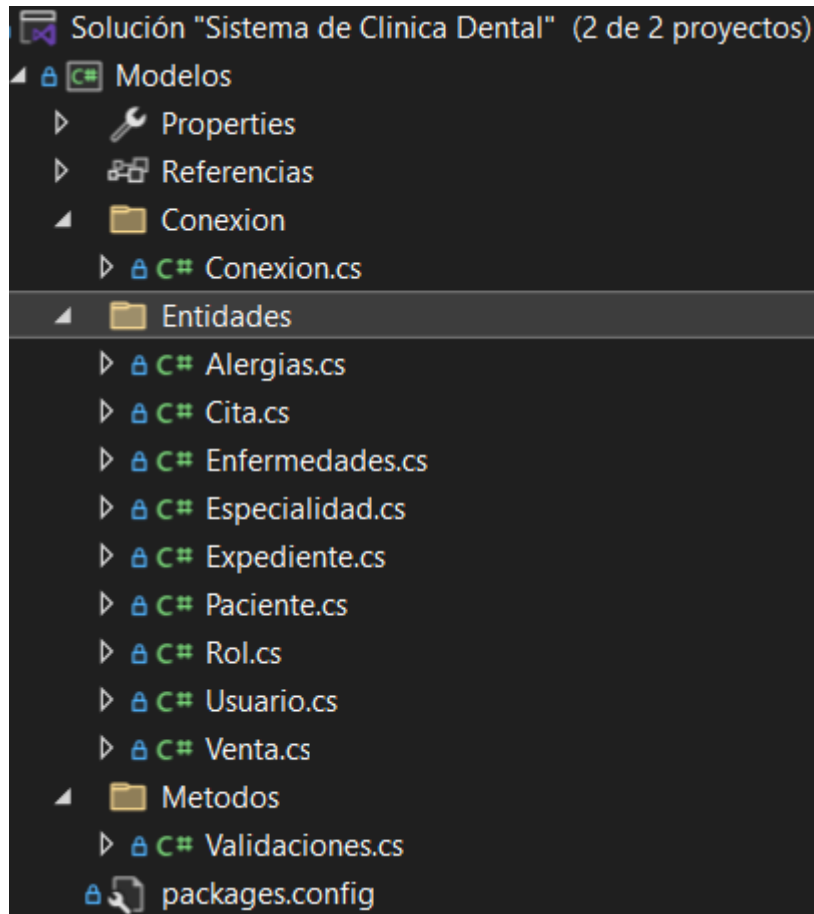
<https://github.com/Tania2509/ptcproyecto.git>



## 6. Estructura del proyecto

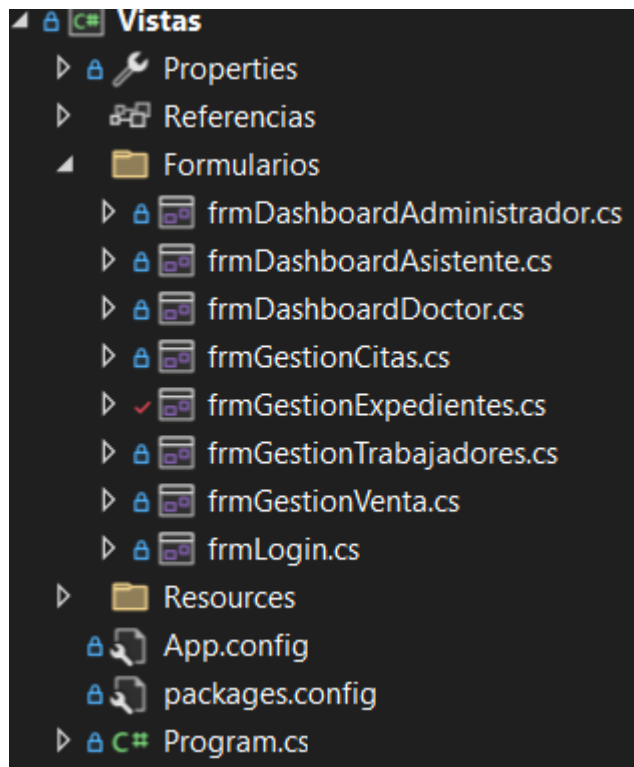
El proyecto se divide en Vistas y Modelos.

### Estructura de Modelos.



- En la carpeta Conexión, se encuentra el método para poder conectarse con la base de datos.
- En la carpeta de entidades, se encuentran todas las tablas de la base de datos importantes, que son necesarias para el funcionamiento del programa.
- En la carpeta de Métodos, se encuentran métodos secundarios, que son necesarios en la manera en la que el cliente podría usar el programa.

### Estructura de Vistas.

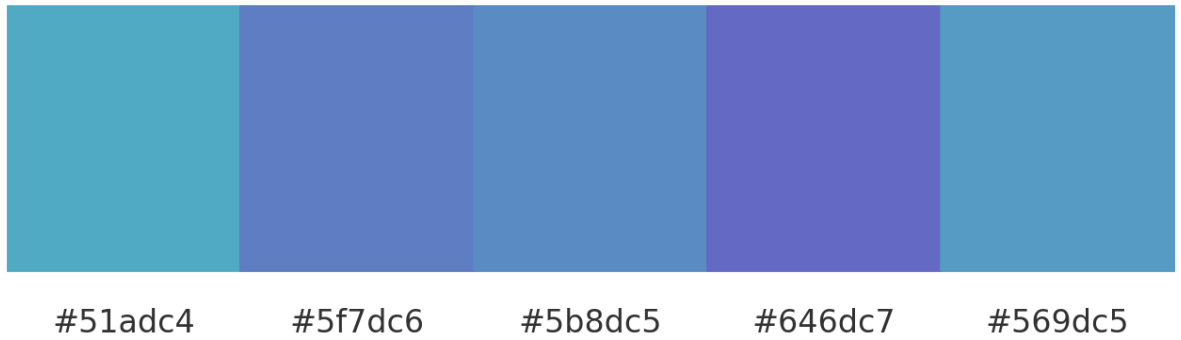


- En la carpeta de formularios, es donde se han creado todos los formularios con los que el cliente podrá interactuar.
- En el apartado de Referencias se encuentra referenciada el apartado de Modelos y la herramienta Bcrypt para la encriptación de contraseña de Usuarios.

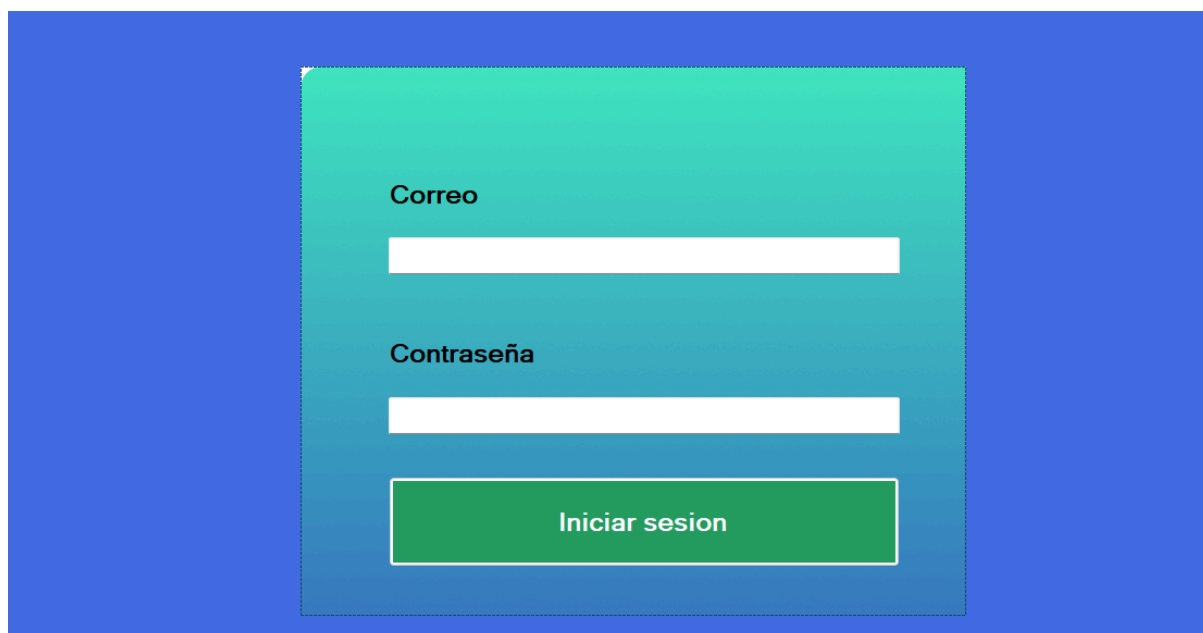
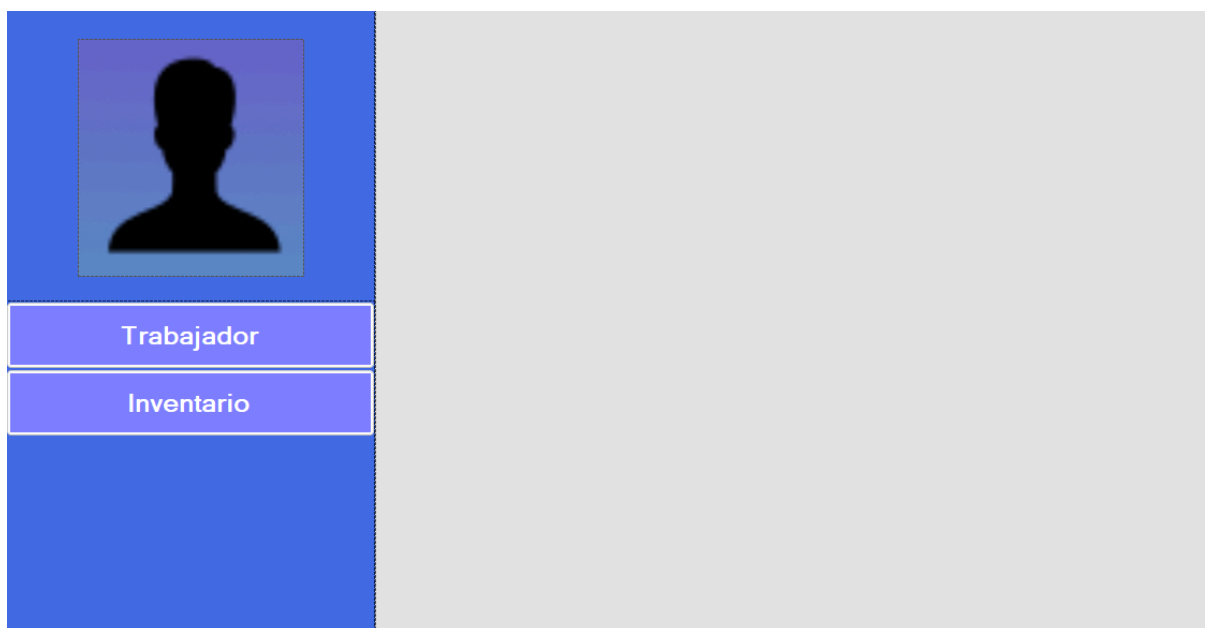
## 7. Diseño de la aplicación.

### **Paleta de colores.**

*Paleta de colores del diseño.*



Nota: En esta imagen se representan los colores que lleva cada diseño del proyecto.



A login form overlay on a blue background. The form has a teal-to-blue gradient background. It contains the following elements from top to bottom: the label 'Correo' in black text above a white text input field; the label 'Contraseña' in black text above a white text input field; and a green rectangular button with the text 'Iniciar sesion' in white.

Nombre del Producto:

Precio del Producto:

Unidades vendidas:

Nombre del Producto:

Precio del Producto:

Unidades vendidas:

Nombre:

Apellido:

Telefono:

DUI:

Fecha de nacimiento:

jueves , 4 de septiemb ▾

Contraseña:

Rol:

▾

Especialidad:

▾

Q

Agregar

Actualizar


Eliminar

Nombre:

Apellido:

Correo Electrónico

Fecha de Nacimiento

jueves , 4 de septiembre de 2025  ▾

Dui:

Q

Agregar

Actualizar

Eliminar

Nombre:

Apellido:

Motivo de la cita

Correo Electrónico

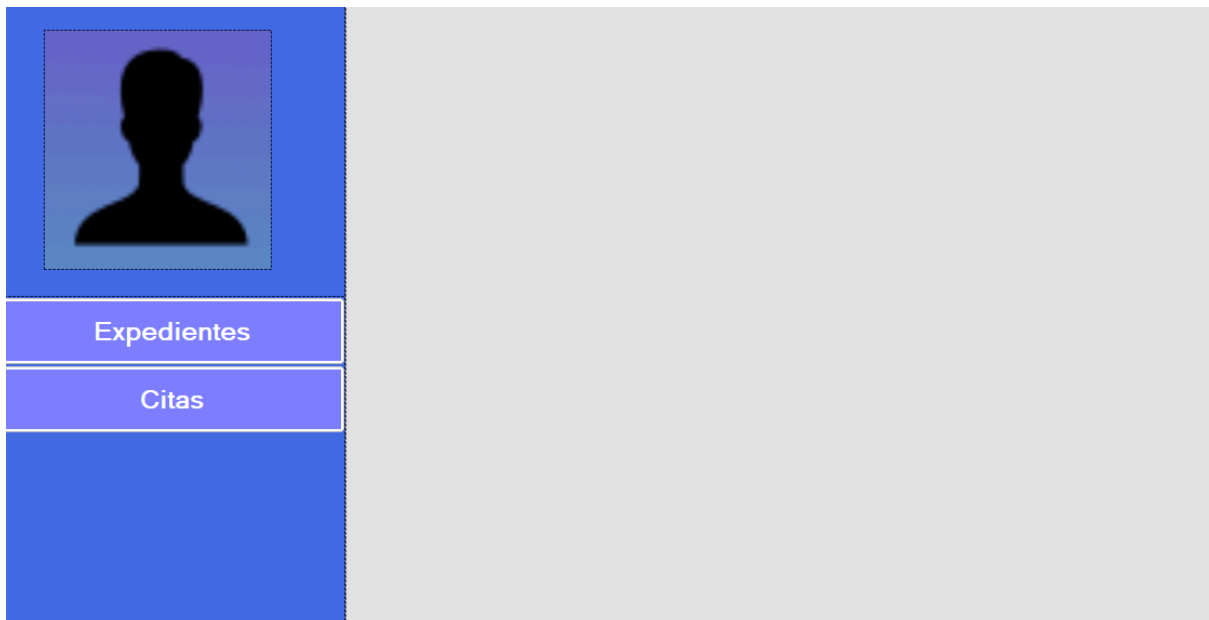
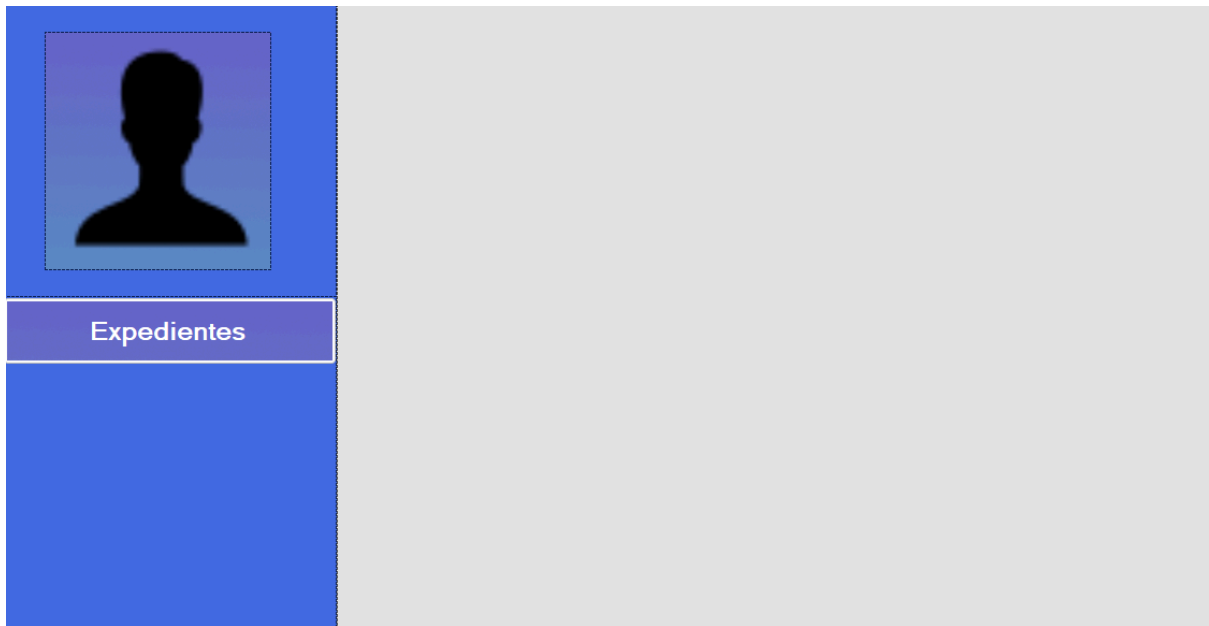
Fecha de la cita:

2025-09-04 10:40

Agregar Cita

Actualizar

Eliminar



### **Tipos y tamaños de fuentes.**

El tipo de fuente empleada en el proyecto es: Microsoft Sans Serif. El cual se emplea con un tamaño estándar de 12.

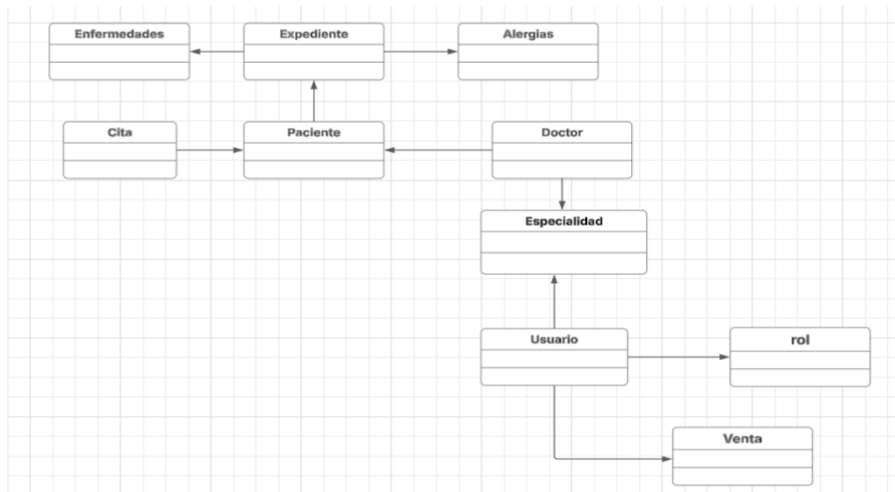


### Tipos y dimensión de imágenes utilizadas

Los tipos de imágenes utilizados en el sistema son PNG, ICON, JPG. Los cuales las dimensiones varían entre 32 x 32 hasta 120 x 120.

A login form interface. It features a teal gradient background with a blue border. The form contains three main elements: a label 'Correo' above a white input field, a label 'Contraseña' above another white input field, and a green button with the text 'Iniciar sesion' at the bottom. The entire form is enclosed in a dashed blue border.

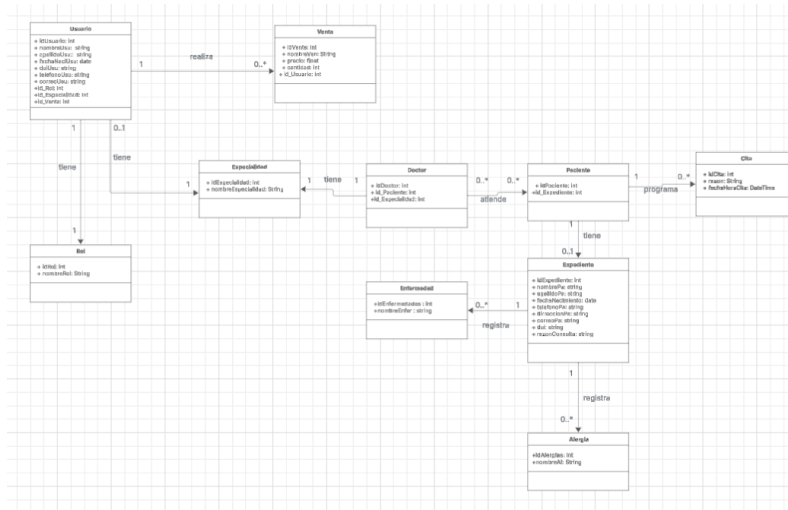




En el siguiente enlace se encontrará el diagrama de modelo de dominio y diagrama de actividades en mejor calidad.

[https://lucid.app/lucidchart/e5def669-f042-4fac-94c0-0bd2d2ce99e5/edit?viewport\\_lo c=1099%2C-2223%2C2527%2C1051%2C0\\_0&invitationId=inv\\_7a2555eb-aad4-4d24-ac73-e055e686e91d](https://lucid.app/lucidchart/e5def669-f042-4fac-94c0-0bd2d2ce99e5/edit?viewport_lo c=1099%2C-2223%2C2527%2C1051%2C0_0&invitationId=inv_7a2555eb-aad4-4d24-ac73-e055e686e91d)

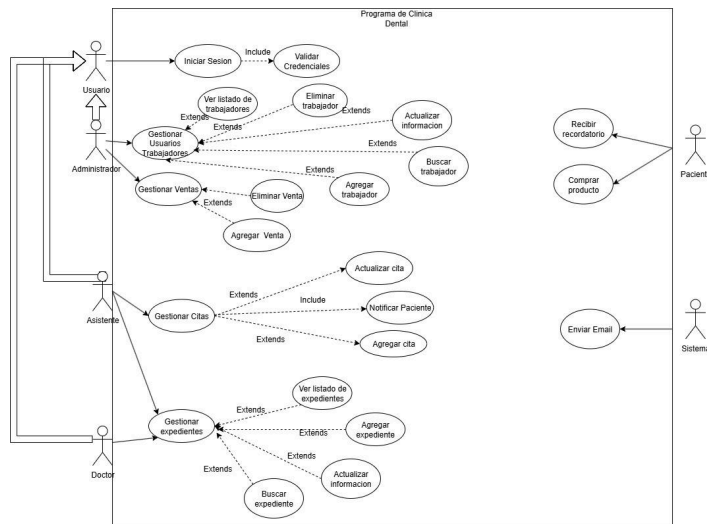
### Diagrama de Clases.



En el siguiente enlace se encontrará el diagrama de clases en mejor calidad.

[https://lucid.app/lucidchart/e7361683-2405-4c3a-aae4-138e124c7082/edit?viewport\\_loc=-339%2C-1328%2C2558%2C1064%2C0\\_0&invitationId=inv\\_5b0af87c-e648-40e6-881a-faa4f77020d5](https://lucid.app/lucidchart/e7361683-2405-4c3a-aae4-138e124c7082/edit?viewport_loc=-339%2C-1328%2C2558%2C1064%2C0_0&invitationId=inv_5b0af87c-e648-40e6-881a-faa4f77020d5)

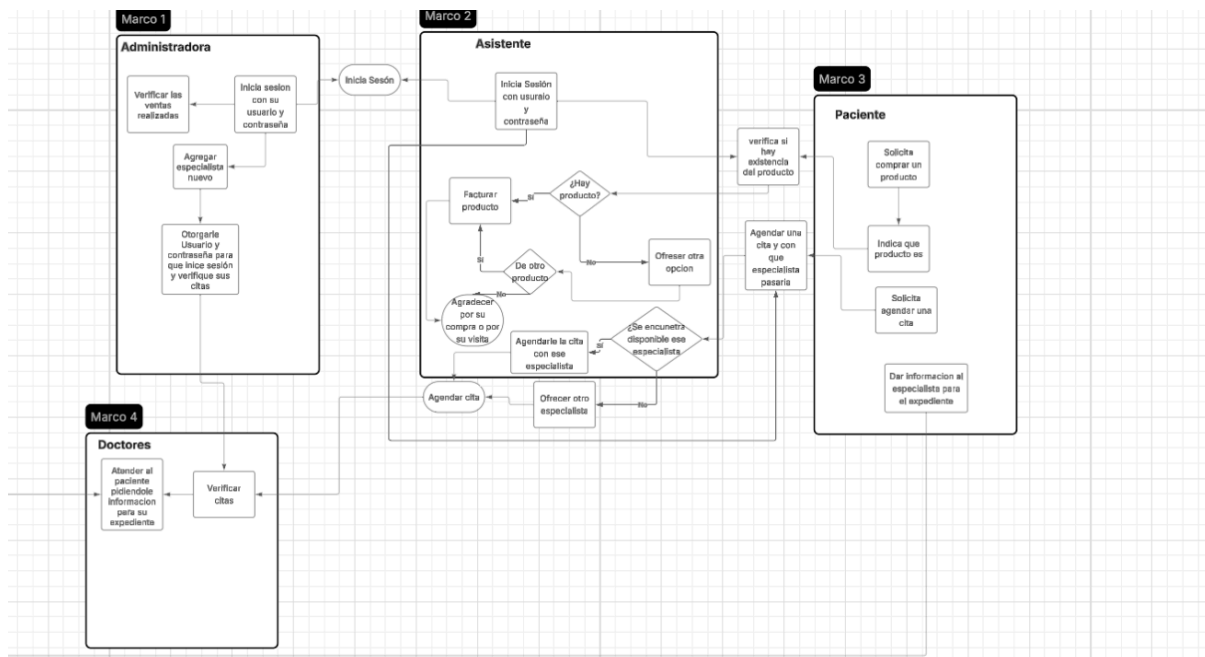
### Diagrama de Casos de Uso.



En el siguiente enlace se encontrará el diagrama de casos de uso en mejor calidad.

[https://drive.google.com/file/d/1jtLoiAWFe\\_KVA2GQWoMMr3qll2r\\_m3EX/view?usp=sharing](https://drive.google.com/file/d/1jtLoiAWFe_KVA2GQWoMMr3qll2r_m3EX/view?usp=sharing)

### Diagrama de actividades.



En el siguiente enlace se encontrará el diagrama de modelo de dominio y diagrama de actividades en mejor calidad.

[https://lucid.app/lucidchart/e5def669-f042-4fac-94c0-0bd2d2ce99e5/edit?viewport\\_lo c=1099%2C-2223%2C2527%2C1051%2C0\\_0&invitationId=inv\\_7a2555eb-aad4-4d24-ac73-e055e686e91d](https://lucid.app/lucidchart/e5def669-f042-4fac-94c0-0bd2d2ce99e5/edit?viewport_lo c=1099%2C-2223%2C2527%2C1051%2C0_0&invitationId=inv_7a2555eb-aad4-4d24-ac73-e055e686e91d)

## 8. Buenas prácticas de desarrollo

Los estándares de programación en C# con un conjunto de pautas y convenciones que los desarrolladores siguen con el fin de escribir código de forma consistente, legible y mantenible. A continuación, se describen los estándares aplicados en el desarrollo del proyecto Sistema de Gestión para Clínicas Dentales.

### 1. Nomenclatura:

- **Nombre de clases:** Las clases representan cosas y no “acciones”.
- **Convenciones de nombre:** Uso de PascalCase, se usa para nombres de espacios de nombres, clases, métodos públicos, propiedades y eventos.

Ejemplo.

```
public class Expediente
public void MostrarExpedientes()
```

### 2. Organización del código.

- **Variables:** Los nombres de las variables son claras e indican su propósito.

Ejemplo.

```
public void MostrarExpedientes()
{
    dgvVerExpedientes.DataSource = null;
    dgvVerExpedientes.DataSource = Expediente.CargarExpedientes("select *from VerExpediente");
}
```

- **Regiones:** Se usan regiones para agrupar métodos relacionados.

## Ejemplo

```
#region Combobox
1 referencia
private void MostrarAlergias()
{
    cbAlergias.DataSource = null;
    cbAlergias.DataSource = Alergias.CargarAlergias();
    cbAlergias.DisplayMember = "nombreAl";
    cbAlergias.ValueMember = "idAlergias";
    cbAlergias.SelectedIndex = -1;
}
```

### 3. Formato.

- **Encapsulación:** Declare las propiedades como privadas para lograr la encapsulación y garantizar la ocultación de los datos.
- **Espacio:** Separe los métodos y las diferentes secciones del programa con un espacio.

## Ejemplo

```
public static DataTable CargarProductos()
{
    SqlConnection con = Conexion.Conexion.conectar();
    string comando = "select idVenta as ID, nombreVen as [Nombre del P
    SqlDataAdapter ad = new SqlDataAdapter(comando, con);
    DataTable dt = new DataTable();
    ad.Fill(dt);

    return dt;
}

1 referencia
public bool InsetarProductos()
{
    SqlConnection con = Conexion.Conexion.conectar();

    string comando = "Insert into Venta(nombreVen, precio, cantidad)"
```

### 4. Controles.

- **Prefijos:** Se agregó un prefijo seguido del nombre del control.

## **9. Requerimientos de hardware y software.**

### **Requerimientos del Hardware:**

- Procesador: mínimo Intel i5 / Ryzen 5 o superior.
- Memoria RAM: mínimo 8 GB (recomendado 16 GB si usas muchas funciones).
- Disco: mínimo 250 GB SSD (más rápido que un HDD).
- Pantalla: resolución Full HD (1920x1080) para ver bien la interfaz.
- Conexión a internet: si usas base de datos en la nube o notificaciones en línea.

### **Requisitos de Software:**

- Sistema Operativo: Windows 10/11 (o Linux si lo desarrollas multiplataforma).
- Base de datos: SQL Server.
- Lenguaje / Framework de desarrollo: C#.
- Drivers y librerías: tener instalados los paquetes de soporte que necesite el programa.
- Opcional: Servidor local (XAMPP, WAMP, etc.) si la base de datos no está en la nube.



## 10. Instalación y configuración.

Sistema de Clínica Dental – C# + SQL Server.

### 1. Requisitos primarios.

Antes de comenzar, asegúrese de contar con lo siguiente:

- Computadora con Windows 10/11.
- .NET Framework 4.8 o .NET 6/7 Runtime (dependiendo de la versión del sistema).
- SQL Server Express o versión superior.
- SQL Server Management Studio (SSMS).
- Copia del sistema (instalador o carpeta con .exe).
- Copia de la base de datos (archivo .bak o script .sql).

### 2. Instalación de SQL Server

1. Descargue e instale SQL Server Express.
2. Instale SQL Server Management Studio (SSMS).
3. Abra el SQL Server Configuration Manager y habilite el protocolo TCP/IP.
4. Reinicie el servicio de SQL Server.

### 3. Restauración de la Base de Datos

Según el tipo de archivo entregado:

Si cuenta con un archivo .bak:

1. Abra **SSMS** y conéctese a la instancia de SQL Server.
  2. Clic derecho en **Databases** → **Restore Database**.
  3. Seleccione **Device** → **Add** → **el archivo .bak**.
  4. Asigne el nombre Clínica Dental y ejecute la restauración.
5. Instalación del Programa
1. Ejecute el instalador del sistema (o copie la carpeta con el ejecutable).
  2. Si es necesario, instale el .NET Runtime correspondiente.
  3. Abra el archivo de configuración y verifique la cadena de conexión.
6. Verificación
1. Abra el sistema de clínica dental.
  2. Inicie sesión con las credenciales iniciales.
  3. Registre un paciente o cita de prueba.
  4. Confirme que los datos se almacenan en la base de datos.