**UDB**

****

**PRYECTO DE CATEDRA AVANCE 1**

**PRESENTADO POR:**

DE LA CRUZ ULLOA JULIO GIOVANNI DU130073

MEJIA HERNANDEZ ZULMA IVETH MH040413

TEJADA CAÑAS JONATHAN ALEXANDER TC160861

CASTILLO GÓMEZ ARIEL OMAR CG161119

HENRIQUEZ PONCE DIEGO NOEL HP160046

PINEDA RAMOS JONATHAN EDGARDO PR180654

**MATERIA:**

DISEÑO Y PROGRAMACION DE SOFTWARE MULTIPLATAFORMA

**CATEDRATICO:**

ALEXANDER ALBERTO SINGUENZA CAMPOS

**CIUDADELA DON BOSCO 31/09/2020**

Tabla de contenido

[Introducción 3](#_Toc49802803)

[Diseño UX/UI 4](#_Toc49802804)

[Explicación del sistema 9](#_Toc49802805)

[Diagrama del Administrador 10](#_Toc49802806)

[Diagrama del Veterinario 11](#_Toc49802807)

[Diagrama de la secretaria 12](#_Toc49802808)

[Base de datos y lógica de negocio 13](#_Toc49802809)

[Diagramas entidad relación 17](#_Toc49802810)

[Tecnologías por utilizar 21](#_Toc49802811)

[Bibliografía 21](#_Toc49802812)

# Introducción

El presente documento forma parte de los estudios de la asignatura de Diseño y Programación de Software Multiplataforma, cuyo enfoque se interesa por el desarrollo y distribución de programas informáticos en las múltiples plataformas informáticas. La informática, como una de ciencias más importantes tanto para el procesamiento de información, almacenamiento y transmisión, tiene una presencia importante en el quehacer y necesidades de las personas, así como también su desarrollo acelerado debido a la aparición de tecnologías como los circuitos integrados, la internet y los dispositivos electrónicos.

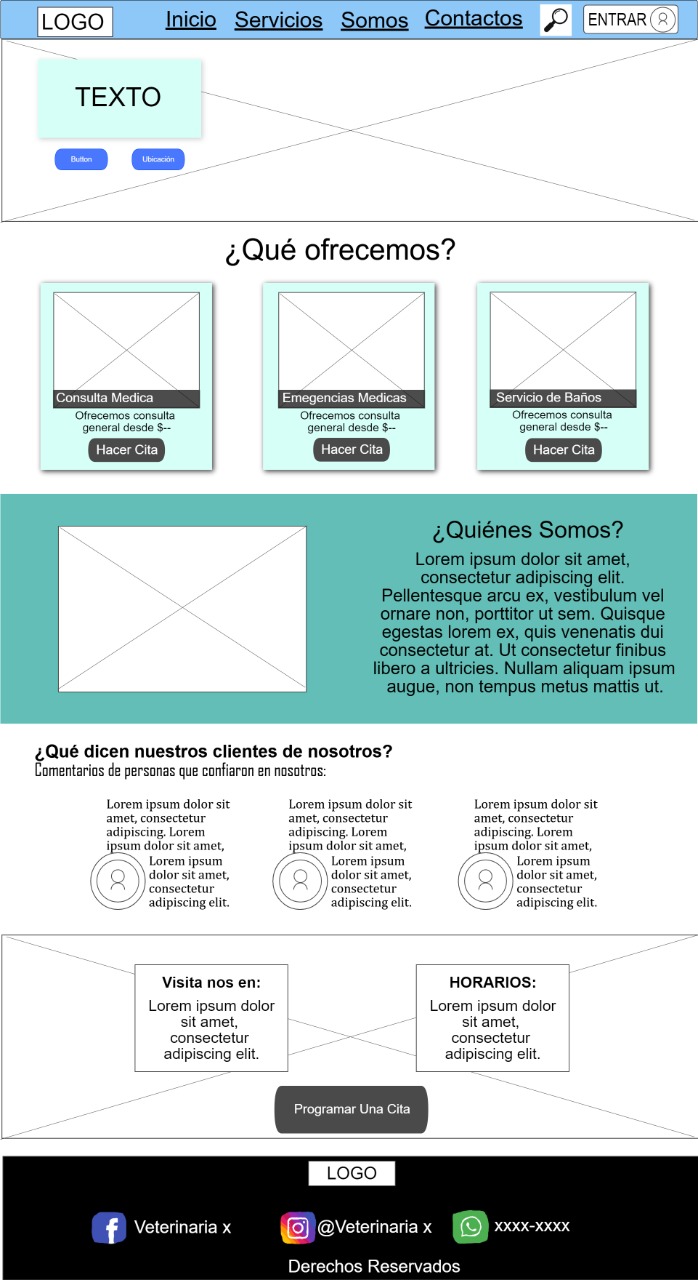
Tanto el diseño como la programación de software funcional en múltiples plataformas es una asignatura compleja, amplia, cuyos límites abarcan desde el uso básico de lenguajes interpretados por el cliente en sus navegadores, hasta la elaboración de sistemas interconectados de redes de datos, métodos típicos de las ciencias aplicadas. En ese panorama tan amplio, es sencillo extraviarse o que el software no esté adaptado adecuadamente a los sistemas que actualmente están en funcionamiento. La adquisición del razonamiento, interpretación y aplicación de los conocimientos adquiridos en las asignaturas relacionadas y por consiguiente está presente, se pretende brindar un acercamiento guiado, práctico y efectivo de la practicas y métodos para el funcionamiento correcto del software. Hablando también de objetivos, se pretende utilizar adecuadamente fundamentos y así también herramientas como los lenguajes de programación para la utilización de las necesidades de los usuarios, entidades u organizaciones.

Es por ello que en el presente trabajo se enfoca en satisfacer las necesidades que una veterinaria puede tener para el control de sus mascotas, es así como se han elaborado las diferentes pantallas para el usuario, los diagramas de uso para que cualquier experto en la materia pueda entender la lógica que se tendrá, la explicación del desarrollo lógico y las herramientas de lenguaje de programación que se han escogido para el desarrollo del mismo, además se presenta una propuesta de la base de datos finalizando por las referencias bibliográficas que se han tomado en cuenta para el desarrollo de la propuesta para el software de control de mascota.

# Diseño UX/UI

Las pantallas que se han diseñado a partir de UX/UI, ya que UX se refiere a la forma como el usuario interactúa con el servicio y UI se refiere a la parte atractiva de imágenes y colores, tomando en consideración ambos aspectos se presentan a continuación la propuesta de pantallas para la página web sobre control de mascotas.

*Pantalla principal*



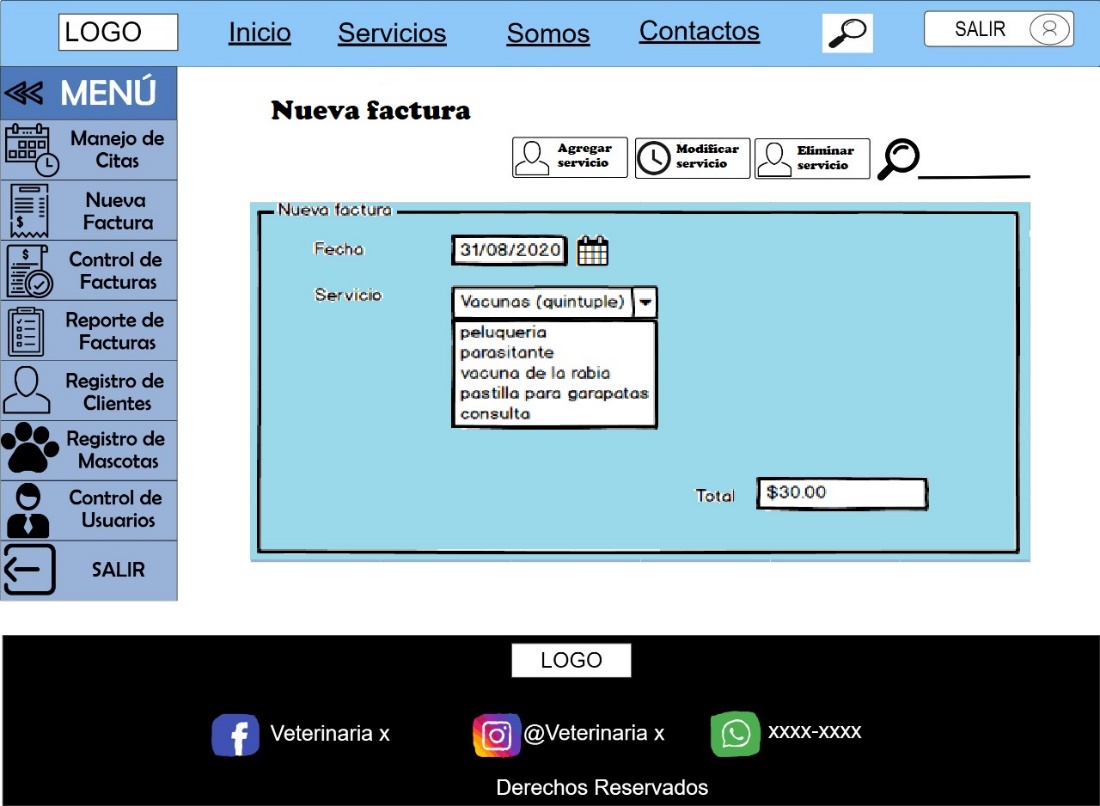
*Generar cita*



*Manejo de citas*



*Nueva factura*



*Reporte de factura*



*Registro de clientes*



*Registro de mascotas*



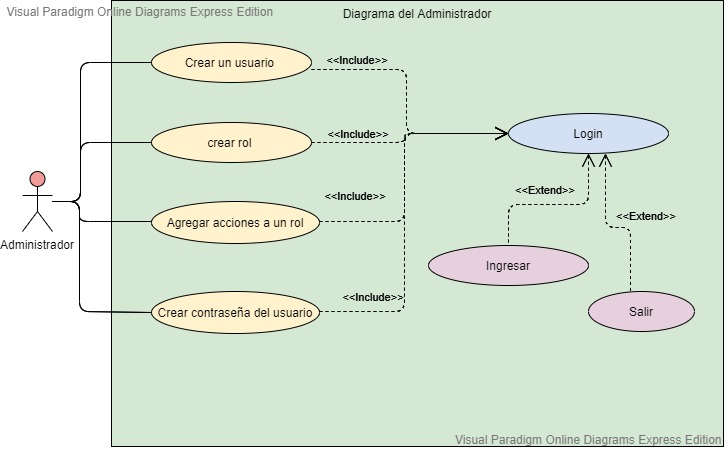
*Control de usuario*



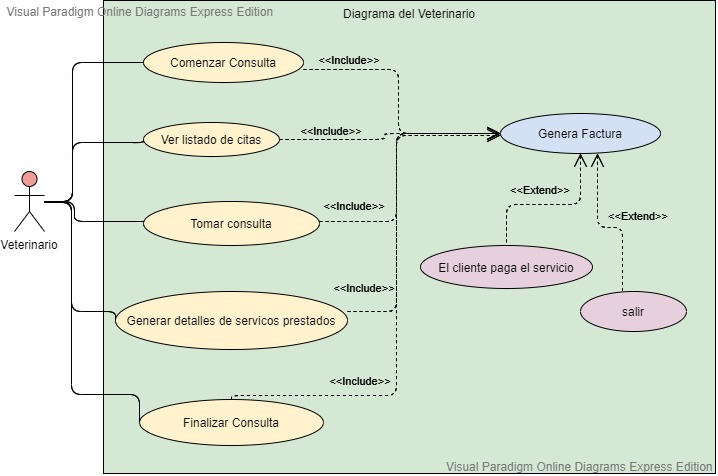
# Explicación del sistema

El principal objetivo del sistema es ayudar a “X” veterinaria a poder llevar un mejor control de todas las actividades que realizan día con día, así como mejorar la experiencia del usuario/cliente ya que sabrá que dicha veterinaria llevará mejor control de su mascota sin miedo a perder los datos de consultas/servicios anteriores y así mismo evitar estar registrando a las mascotas en caso de que perdieran el registro que normalmente lo tiene en un cuaderno o en páginas de papel, con la ayuda de nuestro sistema evitaremos estos inconvenientes ya que todos los datos de cada clientes estarán seguros dentro de una base de datos a los cuales solamente podrá ingresar ya sea un administrador o un empleado pero con la diferencia que dicho empleado solamente podrá visualizar los datos ya ingresados, así como ingresar a algún cliente nuevo, ya que por su rol de usuario tendrá ciertas restricciones dentro del sistema y solo tendrá acceso a ciertos módulos del mismo, ya que otros módulos solamente tendrá acceso el administrador, el cual tendrá acceso a todos los módulos que el sistema tendrá y será el único que podrá realizar modificaciones importantes de los clientes o eliminar algún cliente, si el usuario normal quisiera hacer alguna de estas acciones tendrá que pedir autorización al administrador o pedir directamente al mismo que realice dichas acciones que desee efectuar.

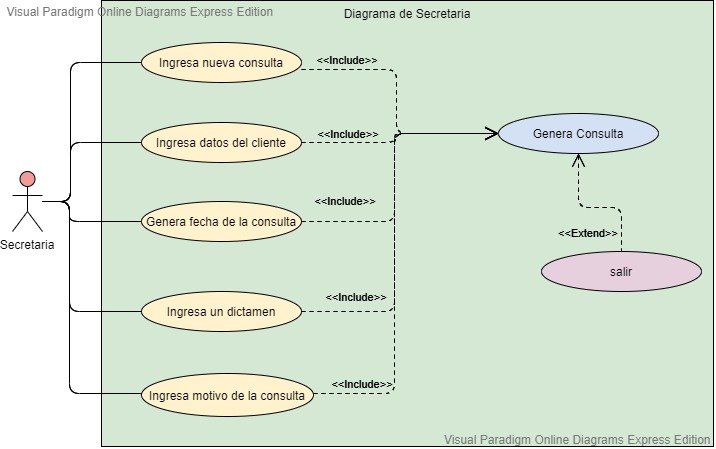
# Diagrama del Administrador



## Diagrama del Veterinario



# Diagrama de la secretaria



## Base de datos y lógica de negocio

|  |  |  |
| --- | --- | --- |
| **Esquema** | **Tabla** | **Descripción** |
| dbo | Accion | contiene acciones o endpoints del sistema |
| dbo | Cita | contiene registros de citas para obtención de una consulta |
| dbo | Cliente | contiene clientes que utilizan los servicios de la clínica |
| dbo | Consulta | entidad que representa una consulta otorgada por un médico veterinario |
| dbo | Empleado | entidad encapsula los usuarios que son empleados de la clínica |
| dbo | Factura | representa una factura que encapsula los costes de los servicios prestados |
| dbo | Mascota | representa una mascota relacionada a un cliente |
| dbo | Rol | entidad que encapsula los distintos tipos de usuarios dentro del sistema |
| dbo | RolAccion | acciones que encapsula un rol |
| dbo | RolUsuario | determina el tipo de usuario dentro del sistema |
| dbo | Servicio | entidad que representa los distintos servicios que ofrece la clínica veterinaria |
| dbo | ServicioConsulta | entidad que encapsula los servicios prestados en una consulta |
| dbo | Usuario | Entidad que representa un usuario dentro del sistema |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Esquema** | **Tabla** | **Columna** | **Tipo dato** | **Tamaño** | **Acepta nulo** | **Llave primaria** | **Llave foranea** | **Descripción** |
| dbo | Accion | Accion | VarChar (500) | 500 | N | Y | N | código de acción en el api controller |
| dbo | Accion | Nombre | VarChar (100) | 100 | N | N | N | nombre de la acción |
| dbo | Cita | Codigo | UniqueIdentifier | 16 | N | Y | N | código único |
| dbo | Cita | FechaReservada | DateTime | 8 | N | N | N | fecha en que se llevara a cabo la cita |
| dbo | Cita | FechaIngreso | DateTime | 8 | N | N | N | fecha en que se ingresó el registro a la base |
| dbo | Cita | IdCliente | UniqueIdentifier | 16 | N | N | N | identificador del cliente que solicito la cita |
| dbo | Cita | MotivoConsulta | VarChar (500) | 500 | Y | N | N | motivo de la cita para la consulta |
| dbo | Cita | IdMascota | UniqueIdentifier | 16 | N | N | N | identificador de la mascota del cliente |
| dbo | Cliente | Codigo | UniqueIdentifier | 16 | N | Y | N | código único del cliente |
| dbo | Cliente | Nombre | VarChar (50) | 50 | N | N | N | nombres del cliente |
| dbo | Cliente | Apellido | VarChar (50) | 50 | N | N | N | apellidos del cliente |
| dbo | Cliente | Direccion | VarChar (150) | 150 | N | N | N | dirección del domicilio del cliente |
| dbo | Cliente | Telefono | VarChar (20) | 20 | N | N | N | teléfono de contacto |
| dbo | Cliente | Email | VarChar (50) | 50 | Y | N | N | email del cliente |
| dbo | Cliente | Dui | VarChar (15) | 15 | N | N | N | número de identificación del cliente |
| dbo | Consulta | Codigo | UniqueIdentifier | 16 | N | Y | N | código único de la consulta |
| dbo | Consulta | Fecha | DateTime | 8 | N | N | N | fecha en que se atendió |
| dbo | Consulta | IdMascota | UniqueIdentifier | 16 | N | N | Y | identificador de la mascota que se atendió |
| dbo | Consulta | IdMedico | VarChar (50) | 50 | N | N | Y | identificador del empleado médico que atendió la consulta |
| dbo | Consulta | Dictamen | VarCharMax | -1 | Y | N | N | resultado emitido por el medico |
| dbo | Consulta | IdCita | UniqueIdentifier | 16 | Y | N | Y | identificador de la cita |
| dbo | Empleado | Usuario | VarChar (50) | 50 | N | Y | N | usuario del empleado |
| dbo | Empleado | Direccion | VarChar (150) | 150 | N | N | N | dirección del domicilio del empleado |
| dbo | Empleado | Telefono | VarChar (20) | 20 | N | N | N | teléfono de contacto del empleado |
| dbo | Empleado | Email | VarChar (50) | 50 | N | N | N | email del empleado |
| dbo | Factura | Codigo | UniqueIdentifier | 16 | N | Y | N | código único de la factura |
| dbo | Factura | IdConsulta | UniqueIdentifier | 16 | N | N | Y | identificador de la consulta relacionada |
| dbo | Factura | Fecha | DateTime | 8 | N | N | N | fecha de emisión |
| dbo | Factura | Total | Decimal (18,0) | 9 | N | N | N | monto total para pagar |
| dbo | Mascota | Codigo | UniqueIdentifier | 16 | N | Y | N | identificador único de la mascota en el sistema |
| dbo | Mascota | Nombre | VarChar (50) | 50 | N | N | N | nombre de la mascota |
| dbo | Mascota | Especie | VarChar (50) | 50 | N | N | N | especie de la mascota |
| dbo | Mascota | Raza | VarChar (50) | 50 | Y | N | N | raza |
| dbo | Mascota | Color | VarChar (50) | 50 | Y | N | N | color pelaje o piel |
| dbo | Mascota | Tamaño | VarChar (50) | 50 | Y | N | N | dimensiones físicas |
| dbo | Mascota | IdCliente | UniqueIdentifier | 16 | N | N | Y | identificador del cliente dueño de la mascota |
| dbo | Rol | Rol | VarChar (20) | 20 | N | Y | N | código del rol |
| dbo | Rol | Descripcion | VarChar (150) | 150 | N | N | N | descripción |
| dbo | RolAccion | Rol | VarChar (20) | 20 | N | Y | Y | código del rol |
| dbo | RolAccion | Accion | VarChar (500) | 500 | N | Y | Y | código de la acción o endpoint |
| dbo | RolUsuario | Usuario | VarChar (50) | 50 | N | Y | Y | id del usuario |
| dbo | RolUsuario | Rol | VarChar (20) | 20 | N | Y | Y | código del rol |
| dbo | Servicio | Codigo | UniqueIdentifier | 16 | N | Y | N | código del servicio |
| dbo | Servicio | Nombre | VarChar (50) | 50 | N | N | N | nombre |
| dbo | Servicio | Descripcion | VarChar (150) | 150 | N | N | N | descripción del servicio |
| dbo | Servicio | TipoServicio | VarChar (50) | 50 | N | N | N | tipo de servicio (enum) |
| dbo | Servicio | Periocidad | VarChar (50) | 50 | Y | N | N | indica si el servicio es repetitivo |
| dbo | Servicio | Activo | Bit | 1 | N | N | N | para denotar si el servicio se presta actualmente |
| dbo | Servicio | Precio | Decimal (18,0) | 9 | N | N | N | precio del servicio |
| dbo | ServicioConsulta | IdConsulta | UniqueIdentifier | 16 | N | Y | Y | código único de la consulta |
| dbo | ServicioConsulta | IdServicio | UniqueIdentifier | 16 | N | Y | Y | código único del servicio |
| dbo | Usuario | Usuario | VarChar (50) | 50 | N | Y | Y | identificador único |
| dbo | Usuario | Nombre | VarChar (100) | 100 | N | N | N | nombre del usuario |
| dbo | Usuario | Apellido | VarChar (100) | 100 | N | N | N | apellido |
| dbo | Usuario | FechaIngreso | DateTime | 8 | N | N | N | fecha que se ingresó al sistema |
| dbo | Usuario | UltimoLogin | DateTime | 8 | N | N | N | última fecha de logan |
| dbo | Usuario | Activo | Bit | 1 | N | N | N | activo o inactivo |
| dbo | Usuario | Contrasena | Binary | 16 | N | N | N | contraseña |

## Diagramas entidad relación

|  |
| --- |
| Proceso creación, edición y accesos de usuarios |

|  |
| --- |
| Proceso creación de cita |

|  |
| --- |
| Proceso consulta |

|  |
| --- |
| Procesos administrativos, catálogos |

|  |
| --- |
| Sistema de clínica veterinaria |

# Tecnologías por utilizar

Para el presente proyecto “sistema de control de consultas para clínica veterinaria” se utilizarán las siguientes tecnologías:

* Base de datos en SQL server
* El backend será desarrollado en .NET utilizando:
  + C#
  + Entity Framework
* Para el frontend se desarrollará un proyecto en angular 9 utilizando:
  + Typescript
  + React Js

# Bibliografía

-Desarrollo de Software, Claudia Pons, Roxana Giandini, Gabriela Pérez, Editorial de la Universidad Nacional de La Plata, 1° edición -2010.

-Angular 4 desde cero, Pedro Jiménez Castela, Madri julio 2017