# 

# **Introducción**

# **Justificación**

Observando la necesidad que tiene el cirujano dentista de practica general y el especialista en áreas odontológicas de contar con un documento personalizado, ordenado y objetivo de cada paciente que cursa un tratamiento. Al tratarse de un expediente los registros se llevan a cabo en papel, en la actualidad los procesos realizados en cualquier campo están cambiando, la industria se está modernizando en todos los ámbitos, buscando siempre la automatización y tratando constantemente de hacer más eficientes sus procesos para mejorar sus resultados, el uso de documentos digitales juega un papel muy importante en estas mejoras ya que permite distribuir tareas, llevando un control de estas de forma eficaz.

Finalmente, se espera que al implementar el proyecto web sirva como apoyo para la gestión y control de citas, a su vez como una herramienta para que los clientes de los odontólogos puedan consultar de una manera más estructurada el seguimiento de su tratamiento.

# **OBJETIVOS**

GENERAL

Desarrollar una plataforma web utilizando ASP.NET Core MVC que facilite a los odontólogos la gestión y control de consultas realizadas a pacientes.

ESPECIFICOS

Diseñar una base de datos utilizando SQL server para proporcionar consultas oportunas y almacenar datos.

Crear una plataforma web utilizando el framework ASP.NET Core MVC para facilitar al usuario el registro de datos.

Programar un módulo para altas, bajas, modificaciones y eliminaciones por parte del administrador y odontólogos.

# **ALCANCES**

El sistema estará disponible para los profesionales de la salud bucodental.

El administrador podrá iniciar sesión para el registro de clientes (odontólogos).

El odontólogo podrá iniciar sesión para registrar pacientes, asignar y visualizar consultas.

# **LIMITACIONES**

La plataforma estará únicamente para navegadores web.

# **ESTADO DEL ARTE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Hallazgo | Año | Características | Referencia | Aporte |
| dindoc | 2014 | Se compone de un dispositivo instalado en un collar el cual se monta en el cuello de los animales. Tiene un diseño propio con un cajeado pequeño, robusto y ergonómico para que al momento de instalarlo en el animal sea cómodo, agradable a la vista y, sobre todo, duradero.  El equipo emite información durante las 24 horas del día. Esta información se recibe en servidores para posteriormente ser analizada y procesada para mostrar, de la mejor manera y de la forma más sencilla la información a ganaderos. La batería del dispositivo tiene una duración de 1 año. Los datos de las localizaciones se pueden visualizar desde cualquier dispositivo móvil mediante una aplicación. Cuenta con dispositivos para ganado bovino, equino, ovino, caprino y fauna silvestre. | Federico, C. (2017) Digitanimal [Internet] Digitanimal, Disponible desde: <https://digitanimal.com/> [Acceso 13 de abril de 2019] | 🠺Dispositivo instalado en forma de un collar.  🠺 El equipo emite información en todo momento.  🠺Las ubicaciones se pueden mostrar en dispositivos móviles mediante una aplicación. |
| Domodis | 2018 | Es un collar adaptado para el tipo de animal, incluye batería recargable y dura 3 meses. Es resistente a temperaturas extremas, incluye comunicación satelital, los datos se visualizan a través de una página web, se podrá descargar las posiciones de todo el año, se establece un perímetro en la aplicación web, avisa cuando un animal Rabasa el perímetro. | Javier, D. (2019) Domodis [Internet] Domodis, Disponible desde: <http://www.domodis.com/> [Acceso 9 de abril de 2019] | 🠺 Batería recargable  🠺 Se puede descargar las posiciones  🠺 Se establece un perímetro  🠺 avisa cuando un animal rabasa el perímetro |
| Geopos | 2014 | Es un sistema de localización de ganado basado en comunicaciones vía Satélite, con un 100% de cobertura, elimina las zonas de sombra existentes con tecnologías GSM, con el envío periódico de la posición del animal, los rebaños están supervisados de manera constante, aportando al ganadero una importante reducción de tiempo y gasto en trabajo diario. Para realizar dicho seguimiento se ha desarrollado una plataforma tecnológica ajustada a las necesidades del ganadero donde se puede ver en todo momento la ubicación actual del ganado, histórico de posiciones y gestiona las parcelas. | Vasco, G. (2014) Geopos [Internet] Hazi Fundazioa. Disponible desde: <http://geopos.hazi.es/#> [Acceso 27 de marzo 2019] | 🠺 Aporta al ganadero una importante reducción de tiempo y gasto en trabajo diario.  🠺 Se puede ver en todo momento la ubicación real del ganado |

# **FUNDAMENTO TEORICO**

# **Aplicaciones web**

En la [Ingeniería de software](https://www.ecured.cu/Ingenier%C3%ADa_de_software) se denomina aplicación web a aquellas aplicaciones que los usuarios pueden utilizar accediendo a un [Servidor web](https://www.ecured.cu/Servidor_web) a través de [Internet](https://www.ecured.cu/Internet) o de una intranet mediante un navegador**.** (Mercadeo, 2020)

Las aplicaciones web están compuestas por tres partes:

Una base de datos: en la base de datos se almacenan precisamente eso, todos los datos, permisos, usuarios, contenidos, información de interés…

El código de la aplicación: la propia aplicación se aloja en un servidor en la nube de aplicaciones, en algunos casos puede almacenarse en un servidor local.

El acceso a través del navegador: puedes acceder utilizando un ordenador o un dispositivo móvil por medio de un navegador. Se incluye el administrador o el gestor que es el usuario final. Podrá acceder a distintas secciones dependiendo de los permisos con los que cuente. (einatec, 2019)

# **Asp net framework**

.NET es una plataforma de desarrollo compuesta por herramientas, lenguajes de programación y bibliotecas para crear muchos tipos diferentes de aplicaciones.

ASP.NET amplía la [plataforma de desarrollo .NET](https://translate.googleusercontent.com/translate_c?depth=1&pto=aue&rurl=translate.google.com&sl=auto&sp=nmt4&tl=es&u=https://dotnet.microsoft.com/learn/dotnet/what-is-dotnet&usg=ALkJrhh8f7AdPxIWEqrEzhCcEbDakeERfw) con herramientas y bibliotecas específicamente para crear aplicaciones web. ASP.NET funciona más rápido que cualquier marco web popular en los [puntos de referencia](https://translate.googleusercontent.com/translate_c?depth=1&pto=aue&rurl=translate.google.com&sl=auto&sp=nmt4&tl=es&u=https://www.techempower.com/benchmarks/&usg=ALkJrhiGb2sPs2BOehXW_bT29VCH_pypfg#hw=ph&test=plaintext) independientes de [TechEmpower](https://translate.googleusercontent.com/translate_c?depth=1&pto=aue&rurl=translate.google.com&sl=auto&sp=nmt4&tl=es&u=https://www.techempower.com/benchmarks/&usg=ALkJrhiGb2sPs2BOehXW_bT29VCH_pypfg#hw=ph&test=plaintext).

ASP.NET es compatible con los protocolos de autenticación estándar de la industria. Las funciones integradas ayudan a proteger sus aplicaciones contra la creación de scripts entre sitios (XSS) y la falsificación de solicitudes entre sitios (CSRF).

ASP.NET proporciona una base de datos de usuarios incorporada con soporte para autenticación multifactor y autenticación externa con Google, Twitter y más. (Microsoft, 2020)

# **Modelo Vista Controlador (MVC)**

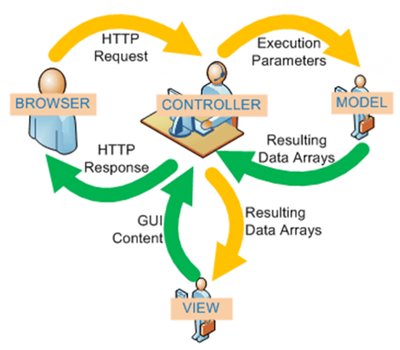
Modelo Vista Controlador (MVC) es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

Se trata de un modelo muy maduro y que ha demostrado su validez a lo largo de los años en todo tipo de aplicaciones, y sobre multitud de lenguajes y plataformas de desarrollo.

El Modelo que contiene una representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia.

La Vista, o interfaz de usuario, que compone la información que se envía al cliente y los mecanismos interacción con éste.

El Controlador, que actúa como intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno. (Alicante, 1996-2020)



# **C#**

La sintaxis viene derivada de C y C+ + y utiliza el modelo de objetos de la plataforma .NET, muy parecido al de Java, aunque incluye mejoras propias de otros lenguajes. Como curiosidad, el nombre de este lenguaje fue inspirado por la escala musical. En ella, la letra C equivale a la nota musical do y el símbolo # significa sostenido, lo que indica que es un semitono más alto. Así, C# sugiere que es superior a C y C++. (S.L, 2018-2020)

# **SQL SERVER**

El lenguaje de desarrollo utilizado (por línea de comandos o mediante la interfaz gráfica de Management Studio) es [Transact-SQL](https://es.wikipedia.org/wiki/Transact-SQL) (TSQL), una implementación del estándar ANSI del lenguaje SQL, utilizado para manipular y recuperar datos ([DML](https://es.wikipedia.org/wiki/Lenguaje_de_manipulaci%C3%B3n_de_datos)), crear tablas y definir relaciones entre ellas ([DDL](https://es.wikipedia.org/wiki/Lenguaje_de_definici%C3%B3n_de_datos)). (Microsoft, 2020)

**JavaScript**

JavaScript (JS) es un lenguaje de programación ligero, interpretado, o compilado [justo-a-tiempo](https://en.wikipedia.org/wiki/Just-in-time_compilation) (just-in-time) con [funciones de primera clase](https://developer.mozilla.org/es/docs/Glossary/Funcion_de_primera_clase). Si bien es más conocido como un lenguaje de scripting (secuencias de comandos) para páginas web, y es usado en [muchos entornos fuera del navegador](https://en.wikipedia.org/wiki/JavaScript#Uses_outside_Web_pages), tal como [Node.js](https://developer.mozilla.org/es/docs/Glossary/Node.js), [Apache CouchDB](https://couchdb.apache.org/) and [Adobe Acrobat](http://www.adobe.com/devnet/acrobat/javascript.html). JavaScript es un lenguaje de [programación basada en prototipos](https://developer.mozilla.org/es/docs/Glossary/Prototype-based_programming), multiparadigma, de un solo hilo, dinámico, con soporte para programación orientada a objetos, imperativa y declarativa (por ejemplo programación funcional). Lee más en [acerca de JavaScript](https://developer.mozilla.org/es/docs/Web/JavaScript/About_JavaScript). (docs, 2005-2020)

**CSS**

Hojas de Estilo en Cascada (del inglés Cascading Style Sheets) o CSS es el lenguaje de [estilos](https://wiki.developer.mozilla.org/es/docs/Web/API/StyleSheet) utilizado para describir la presentación de documentos [HTML](https://developer.mozilla.org/es/docs/HTML) o [XML](https://developer.mozilla.org/es/docs/XML) (incluyendo varios languages basados en XML como [SVG](https://wiki.developer.mozilla.org/es/docs/Web/SVG), [MathML](https://wiki.developer.mozilla.org/es/docs/Web/MathML) o [XHTML](https://developer.mozilla.org/es/docs/XHTML)). CSS describe como debe ser renderizado el elemento estructurado en la pantalla, en papel, en el habla o en otros medios.

CSS es uno de los lenguajes base de la Open Web y posee una [especificación estandarizada](http://www.w3.org/Style/CSS/#specs) por parte del W3C. Anteriormente , el desarrollo de varias partes de las especificaciones de CSS era realizado de manera sincrónica, lo que permiía el versionado de las recomendaciones. Probablemente habrás escuchado acerca de CSS1, CSS2.1, CSS3. Sin embargo, CSS4 nunca se ha lanzado como una versión oficial. (docs, 2005-2020)

# **AXIOS**

[Axios](https://github.com/axios/axios) es una librería JavaScript que puede ejecutarse en el navegador y que nos permite hacer sencillas las operaciones como cliente HTTP, por lo que podremos configurar y realizar solicitudes a un servidor y recibiremos respuestas fáciles de procesar. Aunque podría resultarnos muy útil en infinidad de situaciones, deberíamos antes analizarla las casuísticas con cuidado para determinar si es la mejor solución. En este artículo, analizamos las principales características de Axios y qué debemos tener en cuenta antes de empezar a trabajar con esta librería. (ARSYS, 2020)

# **Bootstrap**

Bootstrap es un [framework](https://rockcontent.com/es/blog/framework/) CSS desarrollado por Twitter en 2010, para estandarizar las herramientas de la compañía.

Inicialmente, se llamó Twitter Blueprint y, un poco más tarde, en 2011, se transformó en código abierto y su nombre cambió para Bootstrap. Desde entonces fue actualizado varias veces y ya se encuentra en la versión 4.4.

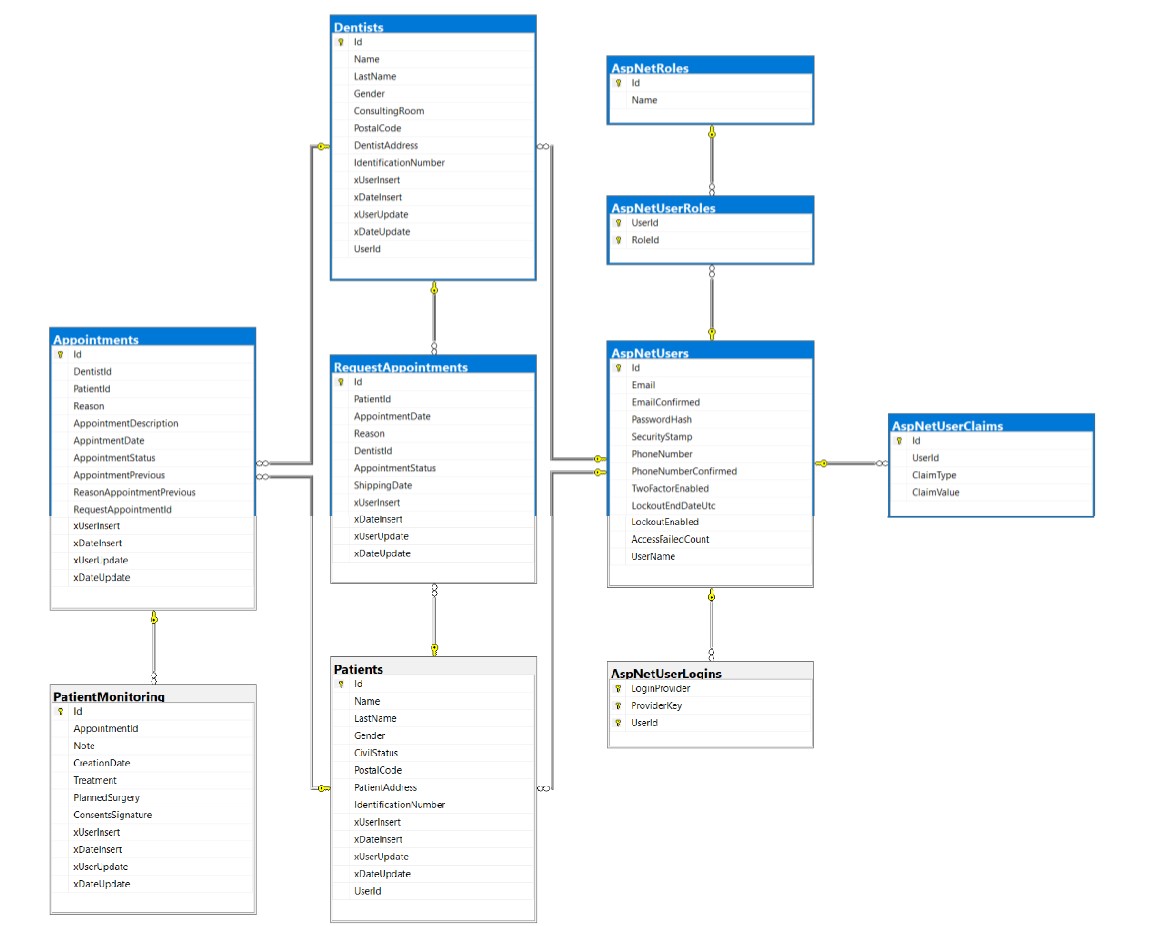
El framework combina [CSS](https://rockcontent.com/es/blog/que-es-css/) y JavaScript para estilizar los elementos de una página HTML. Permite mucho más que, simplemente, cambiar el color de los botones y los enlaces.

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. (team, s.f.)

# **RESULTADOS, PLANOS, GRÁFICAS, PROTOTIPOS, MAQUETAS, PROGRAMAS**

## **Problemática**

## **Diseño de la base de datos**



## **Diccionario de datos**

Un diccionario de datos es un conjunto de definiciones  que contiene las características lógicas y puntuales de los datos que se van a utilizar en el sistema que se programa, incluyendo nombre, descripción, alias, contenido y organización.  
Identifica los procesos donde se emplean los datos y los sitios donde se necesita el acceso inmediato a la información, se desarrolla durante el análisis de flujo de datos y auxilia a los analistas que participan en la determinación de los requerimientos del sistema, su contenido también se emplea durante el diseño.  
En un diccionario de datos se encuentra la lista de todos los elementos que forman parte del flujo de datos de todo el sistema. Los elementos más importantes son flujos de datos, almacenes de datos y procesos. El diccionario de datos guarda los detalles y descripción de todos estos elementos.  
El diccionario se desarrolla durante el análisis de flujo de datos y auxilia a los analistas que participan en la determinación de los requerimientos de sistemas. (weebly, 2020)

## **Tabla AspNetRoles**

Almacenara los roles que estarán dentro de la plataforma.

|  |  |  |
| --- | --- | --- |
| Tabla: AspNetRoles | | |
| CAMPO | **DESCRIPCIÓN** | **TIPO DE DATO** |
| Id | Identificador principal de la tabla AspNetRoles. | nvarchar(128) |
| Name | Nombre de los diferentes roles que tendrán los usuarios | nvarchar(256) |

## **Tabla Appointments**

Almacenara las citas que realizara el dentista.

|  |  |  |
| --- | --- | --- |
| Tabla: Appointments | | |
| CAMPO | **DESCRIPCIÓN** | **TIPO DE DATO** |
| Id | Identificador principal de la tabla Appointments. | int |
| DentistId | Indicador foráneo del dentista que está realizando la cita. | int |
| PatientId | Indicador foráneo del paciente al cual se le asigna la cita. | int |
| Reason | Motivo por la cual se asignó la cita | nvarchar(100) |
| Description | Descripción detallada del motivo de la cita | nvarchar(200) |
| Date | Fecha y hora de la cita | Datetime |
| Status | Estado de la cita | nvarchar(15) |
| AppointmentPrevious | Indica si la cita es hija de otra cita anterior | nvarchar(50) |
| ReasonAppointmentPrevious | Explica el porque la cita es hija de otra anterior. | nvarchar(300) |
| RequestAppointmentId | Indicador foráneo para indicar si la cita es una solicitada por un paciente. | int |
| UserInsert | Indicador foráneo para tener un control de los usuarios que realizan cambios o inserciones. | nvarchar(128) |
| DateInsert | Fecha y hora en la cual se realizó la inserción a la tabla | datetime |
| UserUpdate | Indicador foráneo para tener un control de los usuarios que realizan cambios o inserciones. | nvarchar(128) |
| DateUpdate | Fecha y hora en la cual se realizó la actualización a la tabla | datetime |

## **Tabla Dentists**

Almacena la información de los dentistas.

|  |  |  |
| --- | --- | --- |
| Tabla: Dentists | | |
| CAMPO | **DESCRIPCIÓN** | **TIPO DE DATO** |
| Id | Identificador principal de la tabla Dentists. | int |
| Name | Guarda el nombre de los/las dentistas. | nvarchar(20) |
| LastName | Guarda los apellidos de los/las dentistas. | nvarchar(30) |
| Gender | Almacena el género de los/las dentistas. | nvarchar(10) |
| ConsultingRoom | Almacena el consultorio. | nvarchar(100) |
| PostalCode | Almacena el código postal de los/las dentistas. | int |
| Address | Almacena la dirección de los/las dentistas. | nvarchar(100) |
| IdentificationNumber | Almacena el número de identificación de los/las dentistas. | nvarchar(13) |
| UserInsert | Indicador foráneo para tener un control de los usuarios que realizan cambios o inserciones. | nvarchar(128) |
| DateInsert | Fecha y hora en la cual se realizó la inserción a la tabla | datetime |
| UserUpdate | Indicador foráneo para tener un control de los usuarios que realizan cambios o inserciones. | nvarchar(128) |
| DateUpdate | Fecha y hora en la cual se realizó la actualización a la tabla | datetime |

## **Tabla Patients**

Almacena la información de los pacientes de los dentistas.

|  |  |  |
| --- | --- | --- |
| Tabla: Dentists | | |
| CAMPO | **DESCRIPCIÓN** | **TIPO DE DATO** |
| Id | Identificador principal de la tabla Patients. | int |
| Name | Guarda el nombre de los/las dentistas. | nvarchar(20) |
| LastName | Guarda los apellidos de los/las dentistas. | nvarchar(30) |
| Gender | Almacena el género de los/las dentistas. | nvarchar(10) |
| CivilStatus | Almacena el estado civil de los/las pacientes. | nvarchar(15) |
| PostalCode | Almacena el código postal de los/las dentistas. | int |
| Address | Almacena la dirección de los/las dentistas. | nvarchar(100) |
| IdentificationNumber | Almacena el número de identificación de los/las dentistas. | nvarchar(13) |
| UserInsert | Indicador foráneo para tener un control de los usuarios que realizan cambios o inserciones. | nvarchar(128) |
| DateInsert | Fecha y hora en la cual se realizó la inserción a la tabla | datetime |
| UserUpdate | Indicador foráneo para tener un control de los usuarios que realizan cambios o inserciones. | nvarchar(128) |
| DateUpdate | Fecha y hora en la cual se realizó la actualización a la tabla | datetime |

## **Tabla RequestAppointments**

Almacena las citas solicitadas por los pacientes.

|  |  |  |
| --- | --- | --- |
| Tabla: RequestAppointments | | |
| CAMPO | **DESCRIPCIÓN** | **TIPO DE DATO** |
| Id | Identificador principal de la tabla RequestAppointments. | int |
| PatientId | Indicador foráneo del paciente que está solicitando la cita. | int |
| Date | Fecha en la cual solicita la cita. | datetime |
| Reason | Motivo por el cual se solicita la cita. | nvarchar(100) |
| DentistId | Indicador foráneo del dentista al cual se le está solicitando la cita. | nvarchar(15) |
| Status | Almacena el estado de la cita si fue aprobada o rechazada. | nvarchar(15) |
| ShippingDate | Fecha en la que se envió la solicitud. | nvarchar(100) |
| UserInsert | Indicador foráneo para tener un control de los usuarios que realizan cambios o inserciones. | nvarchar(128) |
| DateInsert | Fecha y hora en la cual se realizó la inserción a la tabla | datetime |
| UserUpdate | Indicador foráneo para tener un control de los usuarios que realizan cambios o inserciones. | nvarchar(128) |
| DateUpdate | Fecha y hora en la cual se realizó la actualización a la tabla | datetime |

## **Tabla PatientMonioring**

Almacena el seguimiento que se le da los pacientes de acuerdo a las citas que haya tenido.

|  |  |  |
| --- | --- | --- |
| Tabla: PatientMonitoring | | |
| CAMPO | **DESCRIPCIÓN** | **TIPO DE DATO** |
| Id | Identificador principal de la tabla PatientMonitoring. | int |
| AppointmentId | Indicador foráneo de la cita atendida. | int |
| Note | Almacena la nota que el dentista asigna en relación con la cita. | nvarchar(300) |
| CreationDate | Almacena la fecha en la cual se le realizo su seguimiento. | nvarchar(100) |
| Treatment | Almacena el tratamiento asignado por el dentista en relación con la cita. | nvarchar(300) |
| PlannedSurgery | Almacena si se tiene una cirugía planeada para poder realizar la cita. | nvarchar(300) |
| ConsentsSignature | Almacena la firma del paciente en la cual está de acuerdo con las notas de su tratamiento. | nvarchar(100) |
| UserInsert | Indicador foráneo para tener un control de los usuarios que realizan cambios o inserciones. | nvarchar(128) |
| DateInsert | Fecha y hora en la cual se realizó la inserción a la tabla | datetime |
| UserUpdate | Indicador foráneo para tener un control de los usuarios que realizan cambios o inserciones. | nvarchar(128) |
| DateUpdate | Fecha y hora en la cual se realizó la actualización a la tabla | datetime |

## **Tabla AsNetUserRoles**

Almacena a los usuarios con sus roles.

|  |  |  |
| --- | --- | --- |
| Tabla: AspNetUserRoles | | |
| CAMPO | **DESCRIPCIÓN** | **TIPO DE DATO** |
| UserId | Identificador foráneo del usuario que se le asignara algún rol. | nvarchar(128) |
| RoleId | Identificador foráneo del rol que se le asignara a el usuario. | nvarchar(128) |

## **Tabla AsNetUsers**

Almacena a los usuarios con sus roles.

|  |  |  |
| --- | --- | --- |
| Tabla: AspNetUsers | | |
| CAMPO | **DESCRIPCIÓN** | **TIPO DE DATO** |
| Id | Identificador principal de la tabla AspNetUsers. | nvarchar(128) |
| Email | Almacena el correo de los usuarios. | nvarchar(256) |
| PasswordHash | Almacena la contraseña cifrada del usuario. | nvarchar(max) |
| PhoneNumber | Almacena el número de teléfono del usuario. | Nvarchar(max) |

# **REFERENCIAS BIBLIOGRAFICAS**

# **REFERENCIAS WEB**

Alicante, U. d. (1996-2020). *si.ua.es*. Obtenido de https://si.ua.es/es

ARSYS. (2020). *arsys.es*. Obtenido de https://www.arsys.es

docs, M. w. (2005-2020). *developer.mozilla.org*. Obtenido de https://developer.mozilla.org

einatec. (2019). *www.einatec.com.mx*. Obtenido de https://einatec.com/

Mercadeo. (2020). *Mercadeo.com*. Obtenido de http://www.mercadeo.com

Microsoft. (2020). *dotnet.microsoft.com*. Obtenido de https://dotnet.microsoft.com

S.L, O. (2018-2020). *openwebinars.net*. Obtenido de https://openwebinars.net

team, B. (s.f.). *getbootstrap.com*. Obtenido de https://getbootstrap.com/