



INSTITUTO TECNOLÓGICO JOSÉ MARIO MOLINA PASQUEL Y HENRÍQUEZ CAMPUS LA HUERTA

CLÍNICA “NUESTRA SEÑORA DE LA PAZ”

INFORME TÉCNICO DE RESIDENCIAS PROFESIONAL

“SITIO WEB DENTALMEDIC”

INGENIERÍA EN SISTEMAS COMPUTACIONALES

PLAN DE ESTUDIOS: ISIC-2010-224

CYNTHIA RAQUEL BRAMBILA PEÑA, 13101045, 9º, INGENIERÍA EN
SISTEMAS COMPUTACIONALES

LA HUERTA JALISCO A 12 DE FEBRERO DE 2018

INSTITUTO TECNOLÓGICO JOSÉ MARIO MOLINA PASQUEL
Y HENRÍQUEZ CAMPUS LA HUERTA

CLÍNICA “NUESTRA SEÑORA DE LA PAZ”

INFORME TÉCNICO DE RESIDENCIAS PROFESIONAL

“SITIO WEB DENTALMEDIC”

INGENIERÍA EN SISTEMAS COMPUTACIONALES

PLAN DE ESTUDIOS: ISIC-2010-224

CYNTHIA RAQUEL BRAMBILA PEÑA, 13101045, 9°, INGENIERÍA EN
SISTEMAS COMPUTACIONALES

ASESOR INTERNO

M.I. JOSÉ LUIS GONZÁLEZ ORTÍZ

ASESOR EXTERNO

CDEO. DIEGO SÁNCHEZ
COVARRUBIAS

Agradecimientos

Gracias Dios, tu amor y bondad no tienen fin, el permitirme alcanzar un logro más que es el resultado de tu ayuda. Agradezco a cada uno de los miembros de mi familia, amigos, maestros y compañeros de clases, sin su apoyo no hubiese podido lograr culminar esta etapa.

Índice General

CAPÍTULO I.- INTRODUCCIÓN	1
1.1 Justificación	2
1.2 Objetivos generales y específicos	2
1.2.1 Objetivo general	2
1.2.2 Objetivos específicos	2
1.3 Caracterización del área en que se participó	2
1.3 Problemas a resolver priorizándolos	3
1.4 Alcances y limitaciones	3
1.5.1 Alcances	3
1.5.2 Limitaciones	3
CAPÍTULO II.- FUNDAMENTOS TEÓRICOS	4
2.1 Sistemas de información	4
2.2 Software	5
2.2.1 Modelo de desarrollo de software	5
2.3 Hardware	6
2.4 Sistemas de Base de Datos	6
2.4.1 Diagrama E-R	7
2.4.2 Diccionario de Datos	7
2.4.3 Normalización	7
2.4.3.1 Primera Forma Normal (1FN)	8
2.4.3.2 Segunda Forma Normal (2FN)	8
2.4.3.3 Tercera Forma Normal (3FN)	8
2.4.3.4. Forma normal de Boyce-Codd (FNBC)	9
2.5 Diagramas	9

2.5.1 Diagrama caso de uso	9
2.5.2 Diagrama de secuencia	10
2.6 Sitios web, Estructura del documento Html5	10
2.7 HTML	15
2.7.1 HTML 5	16
2.7.2 Etiquetas y Atributos HTML	16
2.7.2.1 Elementos y Etiquetas	16
2.8 CSS	17
2.9 JavaScript	18
2.9.1 Lenguaje JavaScript	18
2.10 Lenguaje SQL	20
2.11 Lenguaje PHP	20
2.12 Wampserver	21
2.13 PHPMysqlAdmin	21
2.14 Framework Laravel	22
2.15 Plataforma de programación (SublimeText)	23
CAPÍTULO III.- METODOLOGÍA	24
3.1 Análisis y definición de requisitos	24
3.1.1 Requerimientos del cliente	24
3.1.2. Identificación de actores	25
3.1.3. Diagrama casos de uso	25
3.1.4. Modelos de casos de uso	26
3.1.5. Diagramas de secuencia	30
3.2 Diseño preliminar del sitio web	32
3.3 Diseño de la Base de Datos	37

3.3.1 Diagrama E-R.....	37
3.3.2 Diccionario de Datos	38
3.4 Codificación	41
CAPÍTULO IV.- ANÁLISIS DE RESULTADOS	43
4.1.- Prueba de usabilidad con diferentes resoluciones de pantalla.....	43
4.1.1. Desktop y laptop.....	43
4.1.2. Tablet.....	43
4.1.3. Celular o smartphone.....	43
4.2.- Prueba de satisfacción	44
4.3.- Manual de Usuario	45
CAPÍTULO V.- CONCLUSIONES Y RECOMENDACIONES	46
5.1 Conclusiones.....	46
5.2 Recomendaciones	46
Bibliografía.....	47
ANEXOS	49

Índice de Imágenes

Imagen 1.- Ejemplo Casos de Uso.....	9
Imagen 2.- Ejemplo Diagrama de Secuencia.....	10
Imagen 3.- Representación visual de un diseño web tradicional.....	11
Imagen 4.- Representación visual HTML5.	12
Imagen 5.- WampServer versión 3.0.6.	21
Imagen 6.- Interfaz usuario/contraseña en phpMyAdmin.	21
Imagen 7.- Interfaz phpMyAdmin.....	22
Imagen 8. Interfaz de SublimeText.	23
Imagen 9.- Diagrama de caso de uso Secretaria.	26

Imagen 10.- Diagrama de caso de uso Administrador CDEO.....	26
Imagen 11.- Acceder al sistema.....	30
Imagen 12.-Registrar usuario.	31
Imagen 13.- Registro cita.....	31
Imagen 14.- Vista publicitaria.	32
Imagen 15.- Vista publicitaria responsive.	32
Imagen 16.- Vista lista de expedientes.	33
Imagen 17.- Vista lista de expedientes responsive.	33
Imagen 18.- Vista nuevo expediente.	34
Imagen 19.- Vista nuevo expediente responsive.	34
Imagen 20.- Vista editar expediente.	35
Imagen 21.- Vista editar expediente responsive.....	35
Imagen 22.- Vista agregar, buscar, editar citas responsive.	36
Imagen 23.- Vista agregar, buscar, editar citas.....	36
Imagen 24.- Vista agregar, buscar, editar usuario.	37
Imagen 25.- Vista agregar, buscar, editar usuario.	37
Imagen 26.- Diagrama E-R.....	37
Imagen 27.- Resolución de pantallas 1280 x 1024.....	43
Imagen 28.- Resolución de pantallas 1024 x 1366.....	43
Imagen 29.- Resolución de pantallas 414 x 736.....	44
Imagen 30.- Código QR.	45

Índice de Tablas

Tabla 1.-Acceso al sistema.	27
Tabla 2.- Registro de Cita.....	27
Tabla 3.- Registro de paciente.	28
Tabla 4.- visualizar expedientes.	28
Tabla 5.- Registro de consultas.	29
Tabla 6.- Registro de tratamientos.....	29
Tabla 7.- Registro de usuarios.	30

Tabla 8.- Cita.	38
Tabla 9.- Consulta.	39
Tabla 10.- Tratamiento.	39
Tabla 11.- Paciente.	40
Tabla 13.-Evaluación Administrador CDEO.....	44
Tabla 14.- Evaluación Secretario.....	45

CAPÍTULO I.- INTRODUCCIÓN

La “UNIDAD MEDICA NUESTRA SEÑORA DE LA PAZ” se encuentra ubicado en calle Zaragoza n° 74 entre calle Santa Fe y Gómez Farías en La Huerta, Jalisco, este ofrece servicios profesionales de Ortodoncia y Ortopedia Dentofacial, no contaba con un sitio web publicitario en el cual pueda dar a conocer sus servicios de Ortodoncia a través de internet, tampoco con un control de expediente y de citas de manera digital.

Se desarrolló un sitio web publicitario donde se muestra información importante la clínica de ortodoncia, con un inicio de sección que permite ingresar al sistema web para llevar un control de expedientes y citas.

El desarrollo de este proyecto contiene la siguiente información, actividades realizadas para cumplir con los propósitos, se muestra de la siguiente manera:

- Capítulo 1.- Se visualizan los objetivos y la justificación del proyecto.
- Capítulo 2.- Se muestran los fundamentos teóricos donde se encuentra la teoría que sustenta la investigación y el desarrollo.
- Capítulo 3.- Metodología, análisis, requerimientos, diseño y codificación del proyecto sustentado con el capítulo anterior.
- Capítulo 4.- Análisis de resultados, manual de usuario y manual de programador estos se obtuvieron una vez que finalizó el proyecto.
- Capítulo 5.- Conclusión y recomendaciones, fin y recomendaciones sobre el proyecto.

1.1 Justificación

“El secreto del éxito de internet y de la tecnología web puede reducirse a un concepto: hacer fácil.” (Fontanals, 2004).

La popularización de la web ha influido en el ciclo de información de dos maneras: Primeramente propicia el cambio de funciones tradicionales de creadores, productores y distribuidores de bases de datos y segundamente ha modificado las formas de publicación, transmisión, gestión, recuperación y comercialización de documentos en cualquier tipo de formato digital.

Por esta razón es necesaria la creación de un sitio web el cual permitirá atraer posibles clientes o mantener informados a los clientes actuales, así también permitiendo la digitalización de citas y expedientes clínicos.

Facilitando la información de expedientes y citas al doctor desde su móvil fuera del consultorio dental.

1.2 Objetivos generales y específicos

1.2.1 Objetivo general

Desarrollo de un sistema web para control de citas y expedientes.

1.2.2 Objetivos específicos

- Elaborar una página principal donde el CDEO muestre publicidad referente a la clínica.
- Elaborar un inicio de sesión el cual permita ingresar al sistema web donde controle expedientes y citas.

1.3 Caracterización del área en que se participó

Para el desarrollo de este proyecto se realizó en un consultorio dental, utilizando los conocimientos obtenidos en las siguientes materias cursadas:

- Programación web
- Fundamentos de bases de datos.
- Taller de base de datos.
- Estructura de datos.

- Taller de investigación.
- Gestión de proyectos de software
- Ingeniería del software
- Fundamentos de ingeniería de software.

1.3 Problemas a resolver priorizándolos

- Inicio publicitario, medio de comunicación para mantener informados a los clientes.
- Registro, modificación y cancelación de citas, elaborar citas a pacientes el día y la hora que deben asistir al consultorio dental.
- Registro y cancelación de expediente clínico, cada paciente debe tener un expediente en el cual se registran datos importantes sobre él y su avance clínico.

1.4 Alcances y limitaciones

1.5.1 Alcances

Los puntos más importantes para la realización de este proyecto son:

- Sitio web publicitario
- Registro, modificación y cancelación de pacientes. En el sistema web se podrá registrar o modificarlos datos personales de los pacientes.
- Registro, modificación y cancelación de citas. El sistema web permite registrar, modifica y cancelar citas de los pacientes.
- Registro, modificación y cancelación de tratamientos. se permite recibir los datos más relevantes de los tratamientos que el ofrece el CDEO.
- Registro de consultas. En este apartado permite registrar los avances clínicos del paciente, cada que asiste a revisión.
- Visualiza expediente clínico dental donde se almacenan datos de los pacientes o avances clínicos.
- Seguridad. Para ingresar al sitio web es necesario poseer un usuario y contraseña.

1.5.2 Limitaciones

- No realiza notificaciones sobre las citas.
- No realiza reportes.

CAPÍTULO II.- FUNDAMENTOS TEÓRICOS

2.1 Sistemas de información

Según (Alarcón, 2006). Sistema de Información conjunto de elementos relacionados para lograr un objetivo común. Estos recolectan y procesan información para ayudar a la toma de decisiones y el control de cualquier organización.

“Entre más alto sea el nivel de la información, esta será más rica para la toma de decisiones. Entre menor sea el nivel, esta será más detallada y más operacional, llegando en el extremo a ser un dato.” (Cortés Morales, 1998)

Las características deseables para la información pueden enunciarse por medio de estos cuatro criterios:

- Validez: La información producida debe ser correcta.
- Exactitud: La información debe ser toda la que se requiera.
- Concisión: La información solo tendrá el nivel de detalle que se necesite.
- Oportunidad: La información debe estar disponible en el momento que se necesite.

Los cuatro criterios anteriores deben aplicarse a todos los niveles necesarios de información que el sistema produzca.

Podemos desglosar los sistemas de información en tres categorías:

- Sistemas Transaccionales: Reflejan un sistema en su comportamiento cotidiano. Se basan en las transacciones que se presentan en un sistema.
- Sistemas de control de la gestión (o gerenciales): Son sistemas que se basan en un sistema transaccional. Extraen de esta última información para producir una de mayor nivel que servirá para el control de la gestión.
- Sistemas de apoyo a la toma de decisiones: Estos sistemas son de un alto nivel de información, capaces de representar escenarios de posibles situaciones, de acuerdo con la información registrada o a ciertos parámetros que se le dan de entrada, permitiendo visualizar que decisiones tomar con base en investigaciones, pronósticos o simulaciones.

2.2 Software

“Programas de ordenador y la documentación asociada. Los productos de software se pueden desarrollar para algún cliente en partículas o para un mercado general.” (Sommerville, 2005)

Un proceso de software es un conjunto de actividades y resultados que realiza un producto de software, estas son algunas actividades de procesos de software:

- Especificación: donde los clientes e ingenieros definen el software a producir y las restricciones sobre su operación.
- Desarrollo: Donde el software se diseña y programa.
- Validación: Donde el software se valida para asegurar que es lo que el cliente requiere.
- Evolución: Donde el software se modifica para adaptarlo a los cambios requeridos por el cliente y el mercado.

Un modelo de proceso de software es una descripción simplificada de un proceso del software estas son algunos modelos de procesos de software:

- Modelo de flujo de trabajo: Secuencia de actividades en el proceso junto con sus entradas, salidas y dependencias.
- Modelo flujo de datos: Representa el proceso como un conjunto de actividades, cada una de las cuales realiza alguna transformación en los datos.
- Modelo de redacción: Representa los roles de las personas involucradas en el proceso del software y actividades de las que son responsables.

2.2.1 Modelo de desarrollo de software

La mayor parte de los modelos de procesos se basa en uno de los tres modelos de desarrollo de software en este caso explicaremos la Metodología en cascada:

“El modelo de la cascada, a veces llamado ciclo de vida clásico, sugiere un enfoque sistemático y secuencial⁶ para el desarrollo del software, que comienza con la especificación de los requerimientos por parte del cliente y avanza a través de planeación, modelado, construcción y despliegue, para concluir con el apoyo del software terminado.” (Pressman, 2010)

Este modelo consta de 6 etapas:

- **Ingeniería y Análisis del Sistema:** El software es parte de un sistema, comienza estableciendo los requisitos de todos los elementos del sistema y asigna algún subconjunto de estos requisitos al software.
- **Análisis de los requisitos del software:** Se localiza el proceso de los requisitos y se centra únicamente en el software.
- **Diseño:** Se Concentra en cuatro atributos distintos del programa: Estructura de los datos, Arquitectura del software, Detalle procedimental y Caracterización de la interfaz.
- **Codificación:** El diseño debe traducirse en una forma legible para la máquina.
- **Prueba:** Realizado el código comienzan las pruebas del programa.
- **Mantenimiento:** El software tendrá cambios después de que haya sido entregado al cliente. Esto sucede debido a los errores encontrados, con el fin de que el software realice las actividades que el cliente o mercado necesita de manera correcta

2.3 Hardware

“El Hardware. Es el componente físico, está constituido por la máquina en sí con los dispositivos auxiliares necesarios para realizar las funciones de procesamiento, almacenamiento y transferencia de datos”. (Niño, 2011).

2.4 Sistemas de Base de Datos

“Un sistema de bases de datos es básicamente un sistema computarizado para llevar registros. Es posible considerar a la propia base de datos como una especie de armario electrónico para archivar datos”. (Date, 2001)

Los usuarios del sistema pueden realizar una variedad de operaciones sobre los archivos:

- Agregar
- Insertar.
- Recuperar.

- Modificar.
- Eliminar.

En base de datos existen dos tipos de datos: Integrados y Compartidos. Los integrados se encuentran dentro de la base de datos como una combinación de varios archivos. Los compartidos son partes individuales de datos en la base, estas se pueden repartir entre distintos usuarios y que cada uno de ellos pueda tener acceso a la misma parte de los datos.

2.4.1 Diagrama E-R

“El modelo entidad-relación es una técnica para definir las necesidades de información de su organización, las propiedades de esos asuntos (atributos) y como se relacionan entre sí (relación)” (Baker, 1994)

Gráficamente un diagrama E-R, tiene los siguientes componentes:

- Rectángulos: representa un conjunto de entidades
- Elipses: representa atributos
- Rombos: representa relaciones entre conjuntos de entidades
- Líneas: unen los atributos con los conjuntos de entidades y los conjuntos de entidades con las relaciones.

2.4.2 Diccionario de Datos

“Un diccionario es una lista de nombres ordenada alfabéticamente incluida en los modelos del sistema. El diccionario deberá incluir, además del nombre una breve descripción asociada de dicha entidad con nombre y si el nombre representa un objeto compuesto, una descripción de la composición.” (Sommerville, 2005).

2.4.3 Normalización

“Podemos definir la Teoría de la Normalización como la descomposición sin pérdida de información ni de semántica de la relación universal” (Cabello, 2011)

Las bases de datos relacionales se normalizan para:

- Evitar la redundancia de los datos.

- Disminuir problemas de actualización de los datos en las tablas.
- Proteger la integridad de datos.
- En el modelo relacional es frecuente llamar tabla a una relación, aunque para que una tabla sea considerada como una relación tiene que cumplir con algunas restricciones:
- Cada tabla debe tener su nombre único.
- No puede haber dos filas iguales. No se permiten los duplicados.

Estas son las 3 reglas de Normalización:

2.4.3.1 Primera Forma Normal (1FN)

- Todos los atributos son atómicos.
- Un atributo es atómico si los elementos del dominio son simples e indivisibles.
- La tabla contiene una clave primaria única.
- La clave primaria no contiene atributos nulos.
- No debe existir variación en el número de columnas.
- Los Campos no clave deben identificarse por la clave (Dependencia Funcional)
- Debe Existir una independencia del orden tanto de las filas como de las columnas, es decir, si los datos cambian de orden no deben cambiar sus significados

Esta forma normal elimina los valores repetidos dentro de una Base de Datos.

2.4.3.2 Segunda Forma Normal (2FN)

Una relación está en 2FN si está en 1FN y si los atributos que no forman parte de ninguna clave dependen de forma completa de la clave principal. Es decir que no existen dependencias parciales. (Todos los atributos que no son clave principal deben depender únicamente de la clave principal).

En otras palabras podríamos decir que la segunda forma normal está basada en el concepto de dependencia completamente funcional.

2.4.3.3 Tercera Forma Normal (3FN)

Una relación se encuentra en 3FN si es 2FN y si no existe ninguna dependencia funcional transitiva en los atributos que no son clave.

Además el esquema debe cumplir necesariamente, con las condiciones de segunda forma normal.

2.4.3.4. Forma normal de Boyce-Codd (FNBC)

BCNF (Forma normal de Boyce-Codd, *Boyce-Codd Normal FOI'm*) propuesta como una forma más simple de la 3FN, es más estricta que ésta. Es decir, toda relación que esté en BCNF está en 3FN, pero una relación 3FN *no* está en BCNF.

2.5 Diagramas

2.5.1 Diagrama caso de uso

(Fowler, 1999) Describe un caso de uso como una relación entre un usuario y un sistema. Estos son los tipos de relaciones que se utilizan en los casos de uso:

- Comunica (<<communicates>>): Relación entre un actor y un caso de uso.
- Usa (<<uses>>) (o <<include>> en la nueva versión de UML): Relación de entre dos casos de uso, donde se nota el comportamiento de un escenario en otro.
- Extiende (<<extends>>): Relación entre dos casos de uso donde se nota que un caso de uso es una especialización de otro.

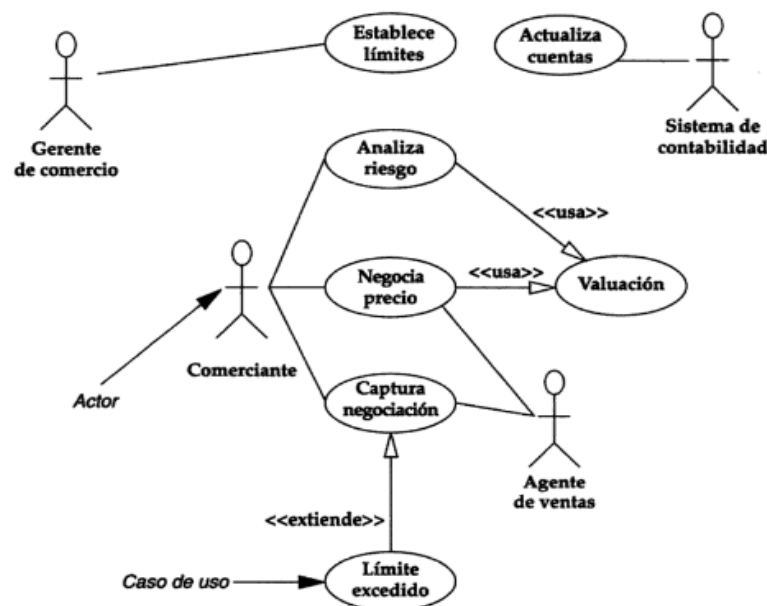


Imagen 1.- Ejemplo Casos de Uso.

2.5.2 Diagrama de secuencia

“Los diagramas de secuencia le ayudan a ver la secuencia de comportamiento” (Fowler, 1999)

De acuerdo con (Sommerville, 2005) Los diagramas de secuencia se usan para modelar las interacciones entre los actores y los objetos en un sistema, así como las interacciones entre los objetos.

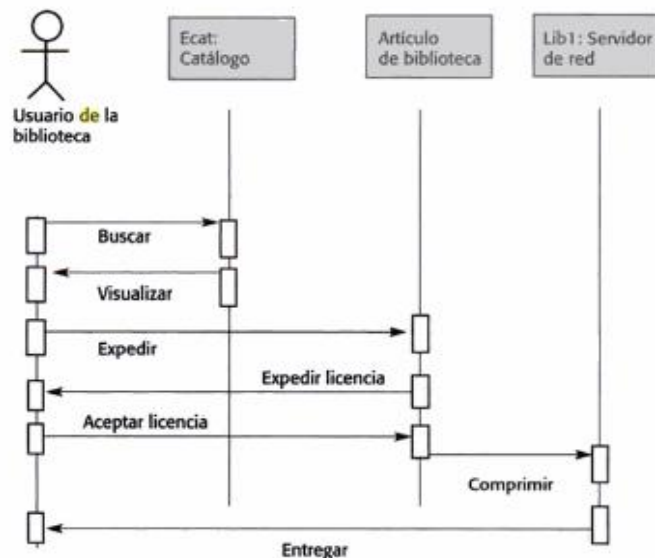


Imagen 2.- Ejemplo Diagrama de Secuencia.

2.6 Sitios web, Estructura del documento Html5

“Los sitios web son archivos que los usuarios descargan con sus navegadores desde ordenadores remotos.” (Gauchat, 2017)

Cuando un usuario accede a un sitio web, le informa al navegador la dirección del sitio y el programa descarga los archivos, procesa su contenido y lo muestra en la pantalla.

Estos archivos deben estar disponibles todo el tiempo ya que los sitios web son de acceso público y el internet es una red global. Es por esto que los sitios web se almacenan en ordenadores especiales, los cuales despachan los archivos a los usuarios que los solicitan. Este

ordenador encargado de almacenar los archivos y datos de un sitio web es llamado Servidor y el ordenador que accede a esta información es llamado Cliente.

Los servidores son similares a los ordenadores personales solo que estos están continuamente conectados a la red y ejecutando programas que les ayudan a responder las solicitudes de los usuarios, Algunos de los programas más usados para servidores son Apache (Linux) y IIS (Windows), estos son los encargados de establecer la conexión entre el cliente y el servidor.

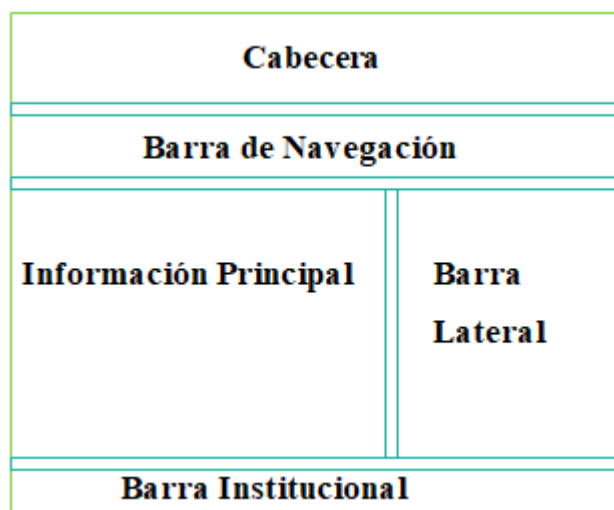


Imagen 3.- Representación visual de un diseño web tradicional.

La barra superior descrita como **Cabecera** en la figura 1, Lugar donde colocamos el logo, nombre del sitio, subtítulos y una breve descripción del sitio. La **Barra de Navegación** ubicada debajo de la cabecera la mayoría de desarrolladores coloca un menú o lista de enlaces para navegar dentro del sitio. En la columna de **Información Principal** generalmente se colocan artículos o noticias relacionados al sitio. **Barra Lateral** se adapta a las necesidades del cliente del sitio web. **Barra Institucional** en este apartado se muestra la información del autor, compañía, términos y condiciones etc.

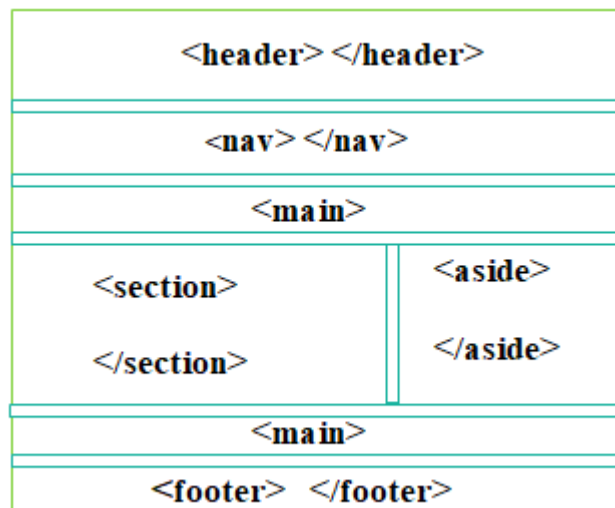


Imagen 4.- Representación visual HTML5.

Los elementos deben ser declarados en el orden q serán representados en la pantalla. En un diseño tradicional la **cabecera** `<header>`. Es muy casual que la mayoría con funda `<header>` con `<head>`.

`<header>` permite proveer información introductoria, títulos, subtítulos este representa el inicio del cuerpo por ejemplo:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Titulo del documento </title>
  <meta charset="utf-8">
  <meta name="descripcion" content="documento html5">
  <meta name="keywords" content="html, css, javascript">
  <link rel="stylesheet" href="css/grid.css">
</head>
<body>
  <header>
```

Este es el titulo

</header>

</body>

</html>

Barra de navegación <nav> representa la sección de navegación, así como se muestra en el siguiente ejemplo:

<!DOCTYPE html>

<html lang="es">

<head>

<title>Titulo del documento </title>

<meta charset="utf-8">

<meta name="descripcion" content="documento html5">

<meta name="keywords" content="html, css, javascript">

<link rel="stylesheet" href="css/grid.css">

</head>

<body>

<header>

Este es el titulo

</header>

<nav>

Inicio | Promociones | Conócenos | Contáctanos

</nav>

</body>

</html>

La sección **Información Principal y Barra Lateral** <main>, <section> y <aside> muestra el contenido del sitio por ejemplo:

<!DOCTYPE html>

<html lang="es">

<head>

<title>Titulo del documento </title>

```
<meta charset="utf-8">
<meta name="descripcion" content="documento html5">
<meta name="keywords" content="html, css, javascript">
<link rel="stylesheet" href="css/grid.css">
</head>
<body>
<header>
Este es el titulo
</header>
<nav>
Inicio | Promociones | Conócenos | Contáctanos
</nav>
<main>
<section>
Artículos
</section>
<aside>
Texto 1
</aside>
</main>
</body>
</html>
```

Por último la **Barra Institucional** **<footer>** la cual nos permite mostrar términos y condiciones entre otras cosas más. Por ejemplo:

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Titulo del documento </title>
<meta charset="utf-8">
<meta name="descripcion" content="documento html5">
```

```
<meta name="keywords" content="html, css, javascript">
  <link rel="stylesheet" href="css/grid.css">
</head>
<body>
<header>
Este es el titulo
</header>
<nav>
Inicio | Promociones | Conócenos | Contáctanos
</nav>
<main>
<section>
Articulos
</section>
<aside>
Texto 1
</aside>
</main>
<footer>
Derechos reservados 20016.
</footer>
</body>
</html>
```

2.7 HTML

El World Wide Web Consortium (W3C) es una comunidad internacional que desarrolla estándares abiertos para garantizar el crecimiento a largo plazo de la Web.

(W3C) público en su sitio web (Berners-Lee, 02) que HTML es el lenguaje para describir la estructura de las páginas web. HTML da a los autores los medios para:

- Publique documentos en línea con encabezados, texto, tablas, listas, fotos, etc.

- Recupere información en línea a través de enlaces de hipertexto, con solo presionar un botón.
- Diseñar formularios para realizar transacciones con servicios remotos, para usar en la búsqueda de información, realizar reservas, solicitar productos, etc.
- Incluya hojas de cálculo, videoclips, clips de sonido y otras aplicaciones directamente en sus documentos.

Con HTML, los autores describen la estructura de las páginas usando el marcado. Los elementos de la etiqueta de idioma incluyen fragmentos de contenido como "párrafo", "lista", "tabla", etc.

2.7.1 HTML 5

De acuerdo con (Prescott, 2015) es una versión más nueva de HTML la cual permite crear aplicaciones web que son capaces de interactuar con los datos locales y con los servidores de manera más sencilla que nunca y además proporciona mayor soporte multimedia,

HTML 5 ofrece además multitud de nuevas funciones que le permiten al usuario interactuar con la web, por ejemplo:

- Nuevos atributos.
- Nuevas reglas de análisis.
- Eliminación de elementos y atributos repetitivos
- Edición de páginas sin conexión.
- Capacidad para almacenar bases de datos MySQL con estándar común.

2.7.2 Etiquetas y Atributos HTML

HTML es escribir etiquetas, atributos y contenido. Es por eso que explicare como un elemento, etiquetas, atributos y contenido se deben utilizar.

2.7.2.1 Elementos y Etiquetas

Según el (Vertice, 2009) las etiquetas HTML son fragmentos de texto rodeados por corchetes angulares < >, que tienen funciones y usos específicos y se utilizan para escribir código HTML.

Normalmente un elemento se compone de dos etiquetas: la etiqueta de apertura y la etiqueta de cierre. Una etiqueta de apertura se compone del nombre del elemento encerrado por los símbolos menor-que "<" y mayor-que ">". La etiqueta de cierre se construye de igual manera que la de apertura pero, en este caso, el nombre del elemento es precedido por una barra diagonal ("/").

Algunas etiquetas con etiqueta de cierre:

Apertura	Cierre
<html>	</html>
<head>	</head>
<body>	</body>
<p>	</p>

También existen etiquetas que no necesitan una etiqueta de cierre:

Etiqueta

<hr>
<input>

2.8 CSS

(Durango, 2015), nos dice que CSS es un lenguaje de estilos que define la presentación de los formatos HTML en base a los estándares “Cascading Style Sheets”.

CSS consta de dos partes: selector y declaración. Declaración nos dice "qué hay que hacer" y el selector nos dice "a quién hay que hacérselo". Los selectores son indispensables para aplicar de forma correcta los estilos CSS en un sitio web. Por ejemplo:

h1 {color: green}

En este caso h1 sería el selector y color: green la declaración.

2.9 JavaScript

“JavaScript se presenta como un lenguaje de desarrollo de aplicaciones cliente/servidor a través de internet.” (Maza, 2001)

2.9.1 Lenguaje JavaScript

JavaScript es el lenguaje de programación de HTML, el cual fue creado para programar el comportamiento dinámico del sitio web, este lenguaje está construido por 3 bloques:

- Variables: datos soportados por JavaScript (números, lógicos, cadenas de caracteres y nulo).
- Objetos: Contiene la colección de propiedades (valores).
- Funciones: Actividades que una aplicación puede ejecutar.

Algunas de las características más importantes lenguaje son:

- Basado en objetos:

Lenguaje orientado a objetos reparte un programa en partes individuales (objetos) que realizan cosas diferentes.

- Eventos:

Los eventos son herramientas proporcionadas por el estándar HTML, diseñados para permitir a los autores la ejecución de programas en sus páginas web en respuesta a la interacción del usuario o el sistema. La finalidad de un evento es asociar una acción, realizada por el usuario o por el sistema.

La manera de declarar eventos es muy similar a la de los atributos solo se necesita: el nombre del evento seguido por el signo igual ("=") y la función o el script entre comillas.

Los eventos más importantes de JavaScript:

Onblur: Deseleccionar el elemento y está definido para estos elementos <button>, <input>, <label>, <select>, <textarea>, <body>.

Onchange: Deseleccionar un elemento que se ha modificado y está definido para estos elementos <input>, <select>, <textarea>

onclick: Pinchar y soltar el ratón y está definido para todos los elementos

ondblclick: Pinchar dos veces seguidas con el ratón y está definido para todos los elementos

onfocus: Seleccionar un elemento y está definido para estos elementos <button>, <input>, <label>, <select>, <textarea>, <body>

onkeydown: Pulsar una tecla (sin soltar) y está definido para elementos de formulario y <body>

onkeypress: Pulsar una tecla y está definido para elementos de formulario y <body>

onkeyup: Soltar una tecla pulsada y está definido para elementos de formulario y <body>

onload: La página se ha cargado completamente y está definido para elementos <body>

onmousedown: Pulsar (sin soltar) un botón del ratón y está definido para todos los elementos

onmousemove: Mover el ratón y está definido para todos los elementos

onmouseout: El ratón "sale" del elemento (pasa por encima de otro elemento) y está definido para todos los elementos

onmouseover: El ratón "entra" en el elemento (pasa por encima del elemento) y está definido para todos los elementos

onmouseup: Soltar el botón que estaba pulsado en el ratón y está definido para todos los elementos

onreset: Inicializar el formulario (borrar todos sus datos) <form>

onresize: Se ha modificado el tamaño de la ventana del navegador <body>

onselect: Seleccionar un texto <input>, <textarea>

onsubmit: Enviar el formulario <form>

onunload: Se abandona la página (por ejemplo al cerrar el navegador) <body>

2.10 Lenguaje SQL

Según (Godoc, 2014) El lenguaje SQL significa lenguaje de consulta estructurada. Lo creo IBM a principios de los años 70.

El lenguaje SQL se divide en varios subconjuntos:

- DDL (Data Definition Language): Agrupa todos los comandos utilizados para crear, modificar o eliminar las estructuras de la base de datos. Se trata principalmente de los comandos CREATE, ALTER Y DROP.
- DML (Data Manipulation Language): Agrupa comandos utilizados para manipular los datos contenidos en la base de datos. Se trata principalmente de los comandos SELECT, INSERT, DELETE Y UPDATE.
- DCL (Data Control Language): Agrupa comandos utilizados para administrar la seguridad de acceso a los datos. Se trata principalmente de los comandos GRANT y REVOKE.
- TCL (Transaction Control Language): Agrupa comandos utilizados para administrar la confirmación o no de las actualizaciones realizadas sobre la base de datos. Se trata principalmente de los comandos COMMIT y ROLLBACK.

2.11 Lenguaje PHP

“El uso conjunto del lenguaje PHP y el sistema gestor de bases de datos MYSQL permite la construcción, de una manera sencilla y eficiente, de verdaderos sitios web dinámicos.” (Cobo, 2005)

(Beati, 2015) nos explica que PHP es un acrónimo de PHP: Hypertext Preprocessor es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML pero se debe especificar que el lenguaje con una extensión “PHP” para que sea reconocido por el servidor antes de ser enviado al cliente.

.Ejemplo para especificar la existencia de PHP en un documento HTML:

```
<?php foreach ($users as $user): ?>  
  
    <li><?php echo $user->name ?></li>  
  
<?php endforeach; ?>
```

2.12 Wampserver

“WampServer es un entorno de desarrollo web de Windows. Le permite crear aplicaciones web con Apache2, PHP y una base de datos MySQL. Además, PhpMyAdmin le permite administrar fácilmente sus bases de datos.” (Bourdon, 2017)

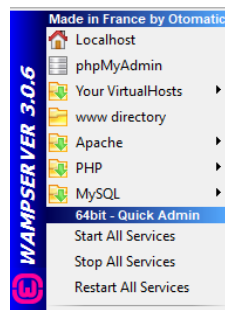


Imagen 5.- WampServer versión 3.0.6.

2.13 PHPMyAdmin

(Alvarez, 2002) nos explica que PhpMyAdmin es una herramienta que se ofrece desde los paneles de control de los alojamientos web con la que podremos manejar y administrar nuestras bases de datos MySQL. Se pueden crear, eliminar, modificar bases de datos así como gestionar las tablas de las mismas.

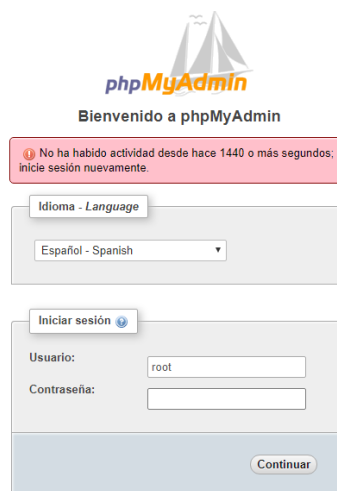


Imagen 6.- Interfaz usuario/contraseña en phpMyAdmin.

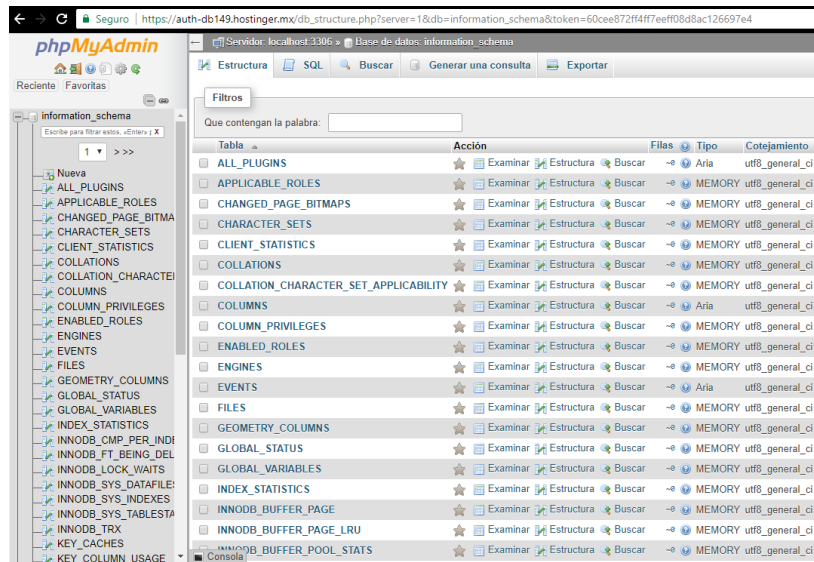


Imagen 7.- Interfaz phpMyAdmin..

La aplicación es un conjunto de archivos escritos en PHP que podemos copiar en un directorio de nuestro servidor web, cuando accedemos a esos archivos, nos muestran unas páginas donde podemos encontrar las bases de datos a las que podemos ingresar en nuestro servidor de bases de datos. La herramienta nos permite crear tablas, insertar datos en las tablas existentes, navegar por los registros de las tablas, editarlos y borrarlos, borrar tablas y un largo etcétera, incluso ejecutar sentencias SQL y hacer un backup de la base de datos.

2.14 Framework Laravel

Un framework, es una estructura creada que sirve de soporte para la construcción de algo que expande la estructura en algo útil.

Laravel es un framework web gratuito de código abierto, creado por Taylor Otwell para el desarrollo de aplicaciones web. A partir de marzo de 2015 , Laravel es considerado como uno de los frameworks PHP más populares , junto con Symfony , Zend, CodeIgniter , Yii2 y otros. El código fuente de Laravel está alojado en GitHub y tiene licencia bajo los términos de la licencia de MIT.

2.15 Plataforma de programación (SublimeText)

Sublime Text es un editor de texto y editor de código fuente está escrito en C++ y Python para los plugins. Se puede descargar y evaluar de forma gratuita. Sin embargo no es software libre o de código abierto y se debe obtener una licencia para su uso continuado, aunque la versión de evaluación es plenamente funcional y no tiene fecha de caducidad.

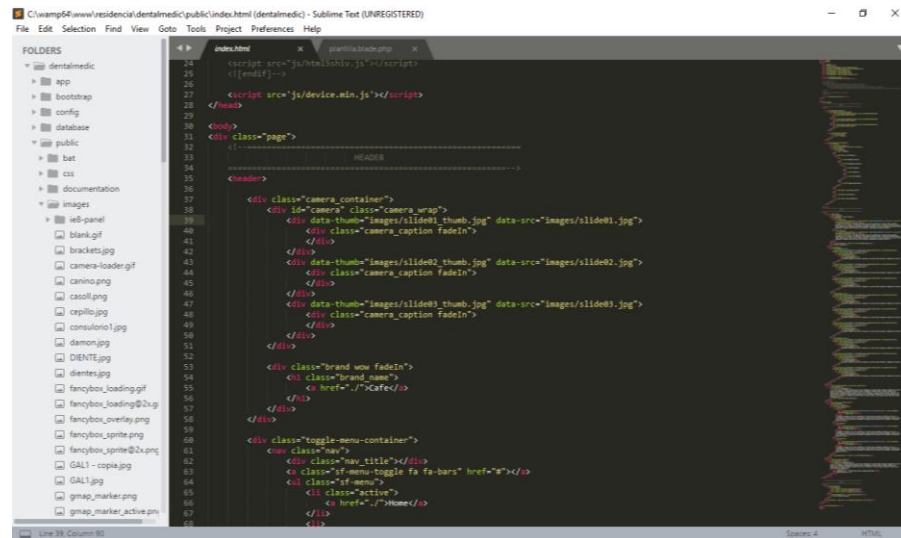


Imagen 8. Interfaz de SublimeText.

CAPÍTULO III.- METODOLOGÍA

El desarrollo de este proyecto se realiza utilizando el modelo en cascada, explicado anteriormente.

3.1 Análisis y definición de requisitos

La “UNIDAD MEDICA NUESTRA SEÑORA DE LA PAZ” donde el CDEO Diego Sánchez Covarrubias es encargado del área de Ortodoncia al cual se le realizó una entrevista, que dio como resultado la manera actual de como realizan sus procesos dentro de la unidad médica.

Secretario

- Registra, modifica y cancela citas.

CDEO

- Registra, modifica y cancela pacientes.
- Registra consultas.
- Registra, cancela expediente clínico.
- Registra, modifica y cancela citas.

3.1.1 Requerimientos del cliente

Los resultados arrojados en la entrevista nos permiten determinar los requerimientos

- El CDEO tendrá acceso al sistema.
- El CDEO requiere un usuario y contraseña para acceder al sistema.
- El CDEO podrá registrar, un paciente (generando automáticamente su expediente) en el sistema.
- El CDEO podrá modificar los datos de un paciente en el sistema.
- El CDEO podrá cancelar un paciente en el sistema.
- El CDEO podrá visualizar los datos de un paciente en el sistema.
- El CDEO podrá buscar un paciente por su nombre en el sistema.

- El CDEO podrá registrar consultas a un paciente en específico (que automáticamente se añadirá al expediente), en el sistema.
- El CDEO podrá buscar una consulta por su fecha en el sistema.
- El CDEO podrá visualizar los datos de una consulta en el sistema.
- El CDEO podrá visualizar el expediente de un paciente en el sistema.
- El CDEO podrá registrar, un tratamiento en el sistema.
- El CDEO podrá buscar un tratamiento por su nombre en el sistema.
- El CDEO podrá modificar los datos de un tratamiento en el sistema.
- El CDEO podrá cancelar un tratamiento en el sistema.
- El CDEO podrá visualizar los datos de un tratamiento en el sistema.
- El CDEO podrá registrar, una cita en el sistema.
- El CDEO podrá buscar una cita por su fecha en el sistema.
- El CDEO podrá modificar los datos de una cita en el sistema.
- El CDEO podrá cancelar una cita en el sistema.
- El CDEO podrá visualizar los datos de una cita en el sistema.
- El secretario tendrá acceso al sistema.
- El secretario requiere un usuario y contraseña para acceder al sistema.
- El secretario podrá registrar, una cita en el sistema.
- El secretario podrá buscar una cita por su fecha en el sistema.
- El secretario podrá modificar los datos de una cita en el sistema.
- El secretario podrá cancelar una cita en el sistema.
- El secretario podrá visualizar los datos de una cita en el sistema.

3.1.2. Identificación de actores

- Secretario.
- Administrador CDEO.

3.1.3. Diagrama casos de uso

La siguiente imagen muestra el caso de uso de la secretaria la cual solo tiene acceso a las actividades relacionadas con las citas en el sistema.

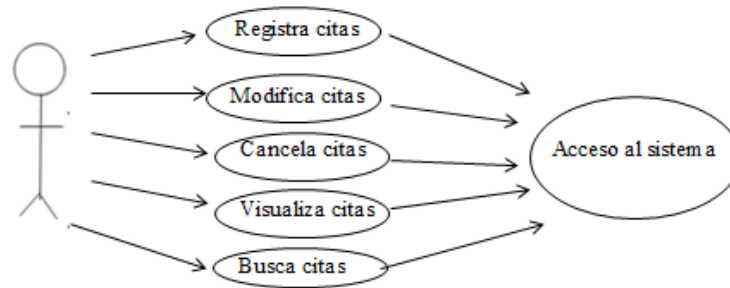


Imagen 9.- Diagrama de caso de uso Secretaria.

La siguiente imagen muestra el caso de uso del Administrador (CDEO) el cual tiene acceso a todas las actividades del sistema.

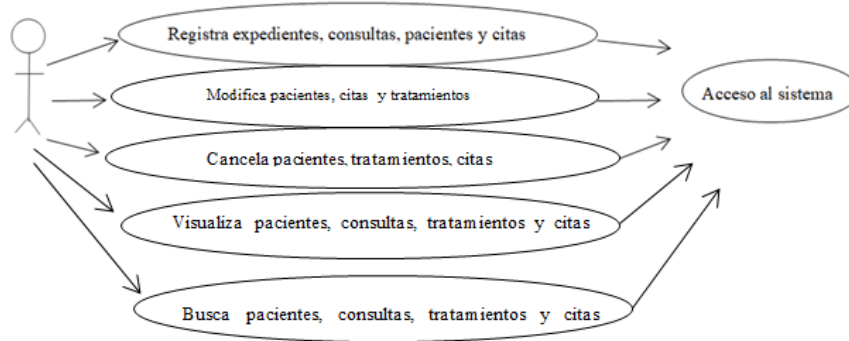


Imagen 10.- Diagrama de caso de uso Administrador CDEO.

3.1.4. Modelos de casos de uso

Una vez terminado el sistema web estas son algunas de las actividades en cada uno de los escenarios:

La secretaria y el administrador CDEO deben ingresar al sistema web a través del caso de uso “Acceso al sistema”, ver tabla.1- Acceso al sitio.

Nombre del caso de uso	Acceso al sistema
Actor participante	Secretaria, Administrador CDEO
Condición Inicial	Debe tener un dispositivo electrónico que le permita acceder a internet (celular, computadora). Conexión a internet. Tener la clave de acceso al sistema.
Flujo de eventos	Abrir el sitio web (http://ortodonciadiegosanchez.esy.es). Para ingresar al sitio da clic en el menú después clic en la opción Inicio de sesión. Debe de iniciar sesión colocando los siguientes datos en el formulario: usuario y contraseña.
Condición de salida	Debe de haber ingresado como secretaria o administrador CDEO. Finalizar la sesión.

Tabla 1.-Acceso al sistema.

La secretaria o Administrador CDEO realizaran el registro de cita a través del caso de uso “Registro de cita”, ver tabla 2.- Registro de cita.

Nombre del caso de uso	Registrar la cita
Actor participante	Secretaria o Administrador CDEO
Condición inicial	Debe haber iniciado sesión en el sistema con el usuario y contraseña correspondiente.
Flujo de eventos	Secretaria o Administrador CDEO correctamente todos los datos del formulario. Registrar la cita
Condición de salida	La cita debe de quedar registrada

Tabla 2.- Registro de Cita.

El Administrador CDEO realizara el registro de pacientes (al realizar el registro del paciente automáticamente se genera su expediente) a través del caso de uso “Registrar paciente”, ver tabla 3.- Registro de paciente

Nombre del caso de uso	Registrar paciente
Actor participante	Administrador CDEO
Condición inicial	Debe de haber iniciado sesión en el sistema con el usuario y contraseña correspondiente.
Flujo de eventos	Debe ingresar correctamente todos los datos del formulario. Registra el paciente.
Condición de salida	El paciente debe de quedar registrado

Tabla 3.- Registro de paciente.

El Administrador CDEO puede visualizar expedientes clínicos, caso de uso “Visualizar expediente”, ver tabla 4.- Visualizar de expedientes.

Nombre del caso de uso	Visualizar expediente
Actor participante	Administrador CDEO
Condición inicial	Debe haber iniciado sesión en el sistema con el usuario y contraseña correspondiente. El paciente debe de estar dado de alta en sitio. Las consultas del paciente deben estar registradas.
Flujo de eventos	Solo visualiza datos del paciente y las observaciones de consultas a las que ha asistido el paciente.
Condición de salida	Cerrar la vista del expediente.

Tabla 4.- visualizar expedientes.

El Administrador CDEO puede registrar las observaciones dentro del apartado consultas y estas solas se anexaran al expediente de paciente correspondiente. Para esto se llevara a cabo el caso de uso “Registro de consultas” que podemos ver en la siguiente tabla 5.- Registro de consultas.

Nombre del caso de uso	Registro de Observaciones
Actor participante	Administrador CDEO
Condición inicial	Debe haber iniciado sesión en el sistema con el usuario y contraseña correspondiente. El paciente debe de estar dado de alta. El tratamiento debe estar dado de alta.
Flujo de eventos	Si durante la consulta el doctor nota algo importante con relación al avance clínico del paciente. Registra consulta.
Condición de salida	La consulta queda registrada.

Tabla 5.- Registro de consultas.

El Administrador CDEO puede registrar tratamientos. Para esto se llevara a cabo el caso de uso “Registro de tratamiento” que podemos ver en la siguiente tabla 6.- Registro de tratamiento.

Nombre del caso de uso	Registro de tratamientos
Actor participante	Administrador CDEO
Condición inicial	Debe haber iniciado sesión en el sistema con el usuario y contraseña correspondiente.
Flujo de eventos	Debe ingresar correctamente todos los datos del formulario. Registra el tratamiento.
Condición de salida	El tratamiento queda registrado.

Tabla 6.- Registro de tratamientos.

El Administrador CDEO puede registrar usuarios. Para esto se llevara a cabo el caso de uso “Registro de usuarios” que podemos ver en la siguiente tabla 7.- Registro de usuarios.

Nombre del caso de uso	Registro de Observaciones
Actor participante	Administrador CDEO
Condición inicial	Debe haber iniciado sesión en el sitio en el sistema con el usuario y contraseña correspondiente.
Flujo de eventos	Registrar datos del usuario.
Condición de salida	El usuario queda registrado.

Tabla 7.- Registro de usuarios.

3.1.5. Diagramas de secuencia

A continuación muestran algunos diagramas de secuencia los cuales reflejan la interacción de los usuarios en el sistema (ver anexos para apreciar mejor las imágenes).

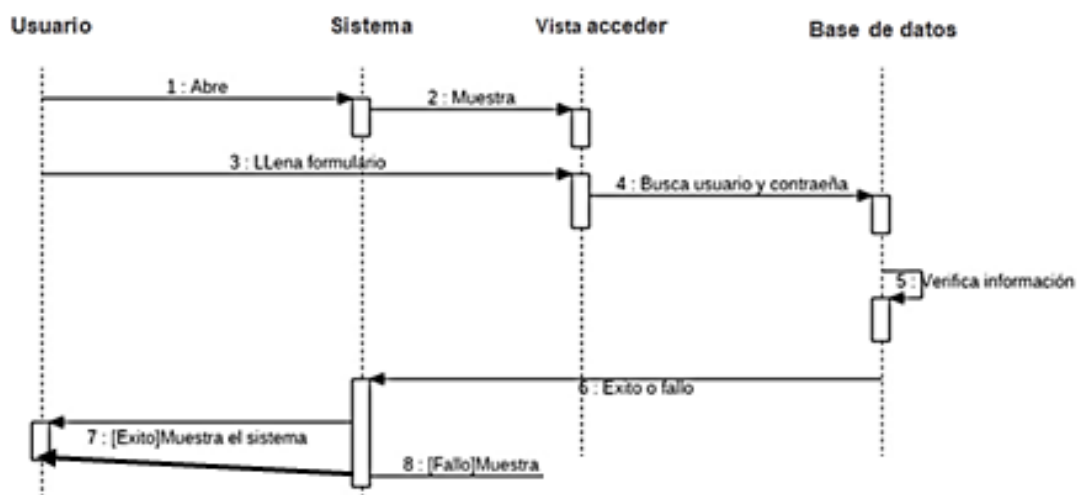


Imagen 11.- Acceder al sistema

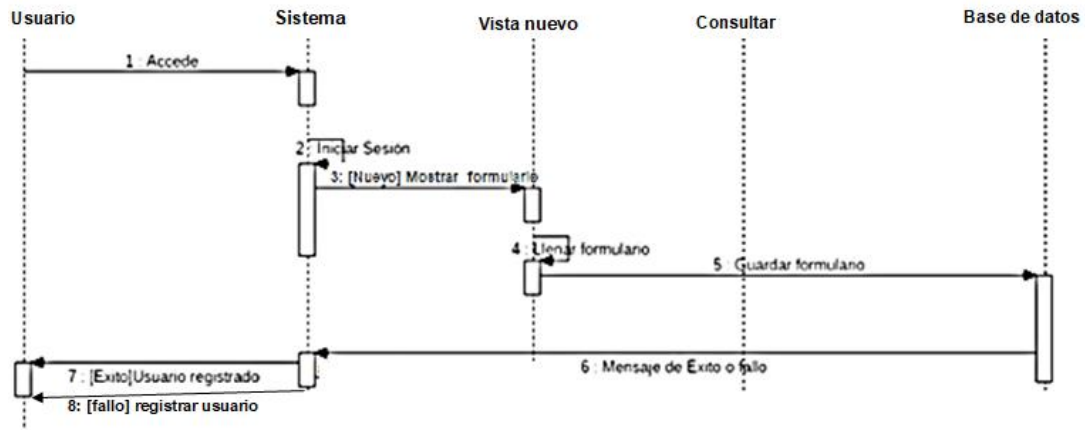


Imagen 12.-Registrar usuario.

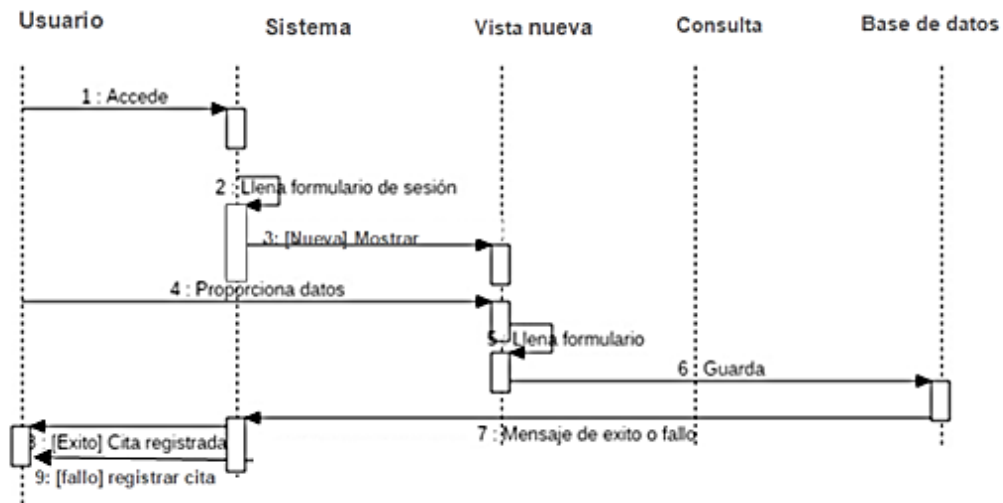


Imagen 13.- Registro cita.

3.2 Diseño preliminar del sitio web

A continuación se muestra un diseño preliminar de cada una de las vistas del sitio web



Imagen 14.- Vista publicitaria.



Imagen 15.- Vista publicitaria responsive.

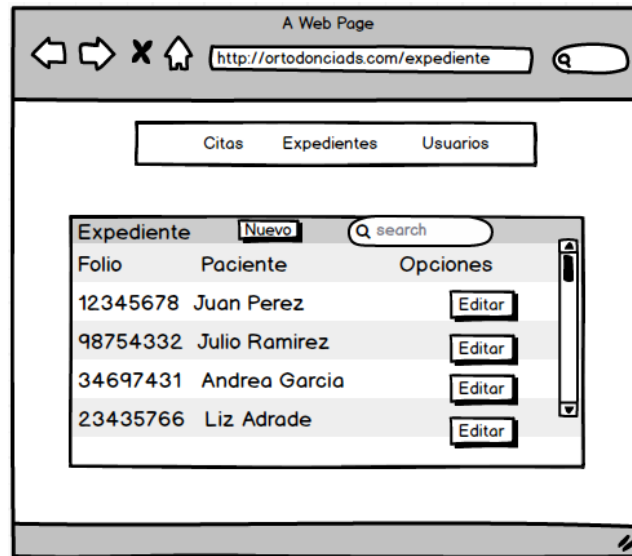


Imagen 16.- Vista lista de expedientes.

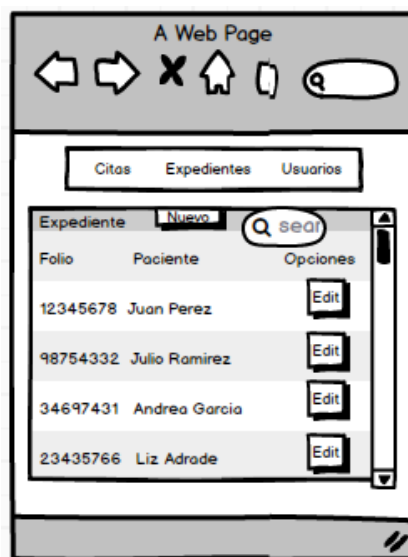


Imagen 17.- Vista lista de expedientes responsive.

A Web Page

http://ortodonciads.com/expediente/nuevo

Citas Expedientes Usuarios

Nuevo Expediente

Nombres

Apellidos

Fecha de nacimiento / /

Domicilio

Telefono

Folio

Agregar

Imagen 18.- Vista nuevo expediente.

A Web Page

http:

Citas Expedientes Usuarios

Nuevo Expediente

Nombres

Apellidos

Fecha de nacimiento / /

Domicilio

Telefono

Folio

Agregar

Imagen 19.- Vista nuevo expediente responsive.

A Web Page

http://ortodonciads.com/expediente/nuevo

Citas Expedientes Usuarios

Editor Expediente

Nombres

Apellidos

Fecha de nacimiento / /

Domicilio

Telefono

Folio

Agregar

Imagen 20.- Vista editar expediente.

A Web Page

http:

Citas Expedientes Usuarios

Editor Expediente

Nombres

Apellidos

Fecha de nacimiento / /

Domicilio

Telefono

Folio

Agregar

Imagen 21.- Vista editar expediente responsive.

A Web Page

Navigation: Citas, Expedientes, Usuarios

Form Fields:

- Citas: Search input (placeholder: search), Editar button
- Paciente: Dropdown menu (selected: Juan Lopez)
- Hora: Time picker (3:30 a.m.)
- Día: Date picker (/ /)
- Agregar button

Imagen 22.- Vista agregar, buscar, editar citas responsive.

A Web Page

Navigation: Citas, Expedientes, Usuarios

Form Fields:

- Citas: Search input (placeholder: search), Editar button
- Paciente: Dropdown menu (selected: Juan Lopez)
- Hora: Time picker (3:30 a.m.)
- Día: Date picker (23 /12 / 2016)
- Agregar button

Imagen 23.- Vista agregar, buscar, editar citas.

A Web Page

Navigation: Citas, Expedientes, Usuarios

Form Fields:

- Usuarios: Search input (placeholder: search), Editar button
- Nombre: Text input
- Contraseña: Text input
- Agregar button

Imagen 24.- Vista agregar, buscar, editar usuario.

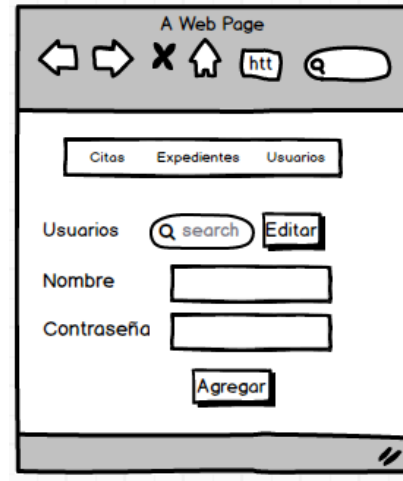


Imagen 25.- Vista agregar, buscar, editar usuario.

3.3 Diseño de la Base de Datos

3.3.1 Diagrama E-R

El siguiente diagrama E-R representa la relación de las tablas realizadas en la base de datos llamada “dentalmedic”. La imagen 23 muestra el diagrama, para su visualización en tamaño aumentado ir a los anexos.

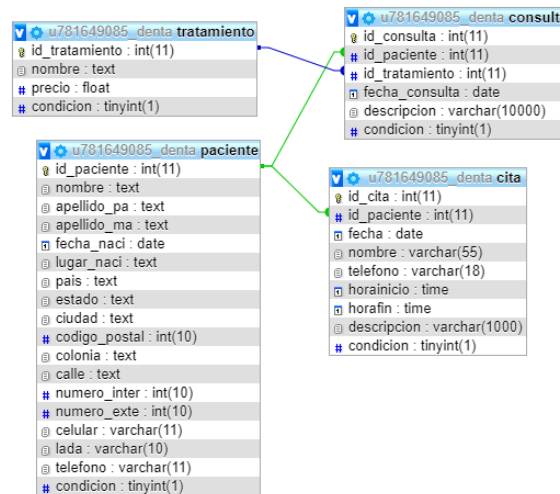


Imagen 26.- Diagrama E-R.

3.3.2 Diccionario de Datos

A continuación se muestran una descripción sobre los campos de las tablas de la bases de datos “dentalmedic”.

Nombre de la tabla: citas

Descripción: Almacena los datos relacionados con las citas.

Número de campos: 9.

Clave primaria: id_cita.

cita			
Columna	Tipo	Nulo	Comentarios
id_cita (<i>Pk</i>)	int(11)	No	Llave primaria para identificar una cita.
id_paciente	int(11)	No	Llave foránea para identificar un paciente.
nombre	varchar(55)	Sí	Nombre del paciente no registrado.
telefono	varchar(18)	Sí	Teléfono del paciente no registrado.
fecha	date	No	Fecha en que debe asistir a la cita.
horainicio	time		Hora en que debe iniciar aproximadamente la cita.
horafin	time	No	Hora en que debe finalizar aproximadamente la cita.
descripcion	varchar(1000)	No	Motivo por el cual desea asistir a la cita.
condicion	tinyint(1)	No	Si está en 1 se muestra. En 0 lo oculta esto es para evitar eliminar una cita.

Tabla 8.- Cita.

Nombre de la tabla: Consulta

Descripción: Almacena los datos relacionados con la consulta.

Número de campos: 6.

Clave primaria: id_consulta.

Consulta			
Columna	Tipo	Nulo	Comentarios
id_consulta (PK)	int(11)	No	Llave primaria para identificar una consulta.
id_paciente (FK)	int(11)	No	Llave foránea para obtener los datos de un paciente.
id_tratamiento	int(11)	No	Llave foránea para obtener los datos de un tratamiento.
fecha_consulta	date	No	Fecha de la cita en que asistió a la consulta.
descripcion	varchar(1000)	No	Notaciones importantes sobre la evolución del paciente con el tratamiento que se le está realizando.
condicion	tinyint(1)	No	Si está en 1 se muestra. En 0 lo oculta esto es para evitar eliminar una consulta.

Tabla 9.- Consulta.

Nombre de la tabla: Tratamiento

Descripción: Almacena los datos relacionados con el tratamiento

Número de campos: 4.

Clave primaria: id_tratamiento.

tratamiento			
Columna	Tipo	Nulo	Comentarios
id_tratamiento (Pk)	int(11)	No	Llave primaria para identificar el tratamiento
nombre	text	No	Nombre del tratamiento
precio	float	No	Precio del tratamiento
condicion	tinyint(1)	No	Si está en 1 se muestra. En 0 lo oculta esto es para evitar eliminar un tratamiento.

Tabla 100.- Tratamiento.

Nombre de la tabla: paciente

Descripción: Almacena los datos relacionados con el paciente

Número de campos: 18.

Clave primaria: id_paciente.

Columna	Tipo	Nulo	Comentarios
id_paciente (<i>pk</i>)	int(11)	No	Llave primaria para identificar al paciente
nombre	varchar(20)	No	Nombre del paciente
apellido_pa	varchar(20)	No	Apellido paterno del paciente
apellido_ma	varchar(20)	No	Apellido Materno del paciente
fecha_naci	date	No	Fecha de nacimiento del paciente
lugar_naci	varchar(45)	No	Lugar de nacimiento del paciente
pais	varchar(20)	No	País donde vive el paciente
estado	int(20)	No	Estado donde vive el paciente
ciudad	varchar(20)	No	Ciudad donde vive el paciente
codigo_postal	int(10)	No	Código postal del lugar donde vive el paciente
colonia	varchar(20)	No	Colonia donde vive el paciente
calle	varchar(20)	No	Calle donde vive el paciente
numero_inter	int(10)	No	Numero interior del lugar donde vive el paciente
numero_exte	int(10)	No	Numero exterior del lugar donde vive el paciente
celular	int(10)	si	Numero celular del paciente
lada	int(5)	Sí	Lada telefónica
telefono	int(10)	Sí	Teléfono de casa o empresa
condicion	tinyint(1)	No	Si está en 1 se muestra. En 0 lo oculta esto es para evitar eliminar un paciente.

Tabla 111.- Paciente.

3.4 Codificación

El siguiente código solo es una parte para realizar el inicio de sesión

```
@extends('layouts.app')
@section('content')
<div class="container">
  <div class="row">
    <div class="col-md-8 col-md-offset-2">
      <div class="panel panel-primary">
        <div class="panel-heading">Acceder</div>
        <div class="panel-body">
          <form class="form-horizontal" role="form" method="POST" action="{{
url('/login') }}">
            {{ csrf_field() }}

            <div class="form-group{{ $errors->has('email') ? ' has-error' : '' }}">
              <label for="email" class="col-md-4 control-label">Dirección E-Mail
</label>

              <div class="col-md-6">
                <input id="email" type="email" class="form-control" name="email"
value="{{ old('email') }}">
                @if ($errors->has('email'))
                  <span class="help-block">
                    <strong>{{ $errors->first('email') }}</strong>
                  </span>
                @endif
              </div>
            </div>

            <div class="form-group{{ $errors->has('password') ? ' has-error' : '' }}">
              <label for="password" class="col-md-4 control-label">Contraseña</label>
              <div class="col-md-6">
                <input id="password" type="password" class="form-control"
name="password">
                @if ($errors->has('password'))
                  <span class="help-block">
                    <strong>{{ $errors->first('password') }}</strong>
                  </span>
                @endif
              </div>
            </div>
          </form>
        </div>
      </div>
    </div>
  </div>
</div>
```

El siguiente código solo es una parte para realizar la búsqueda de una consulta.

```
{!!
Form::open(array('url'=>'co/consulta','method'=>'GET','autocomplete'=>'off','role'=>'search'))
!!}
```

```
<div class="form-group">
  <div class="input-group">
    <input      type="text"      class="form-control"      name="searchText"
placeholder="Buscar..." value="{{ $searchText }}">
    <span class="input-group-btn">
      <button type="submit" class="btn btn-primary">Buscar</button>
    </span>
  </div>
</div>

{{ Form::close() }}
```

CAPÍTULO IV.- ANÁLISIS DE RESULTADOS

En el presente capítulo se muestran los resultados obtenidos de las prueba de resolución de pantalla y satisfacción de usuarios.

4.1.- Prueba de usabilidad con diferentes resoluciones de pantalla

4.1.1. Desktop y laptop

Para la realización las siguientes pruebas resolución de pantallas 1280 x 1024, se utilizó dispositivos electrónicos del resultado obtenido es el siguiente.

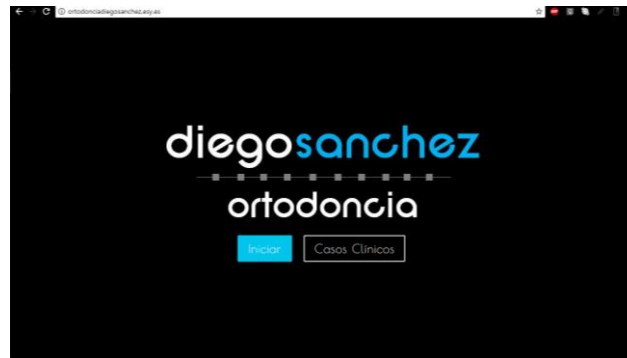


Imagen 27.- Resolución de pantallas 1280 x 1024.

4.1.2. Tablet

Para la realización las siguientes pruebas resolución de pantallas 1024 x 1366, se utilizó dispositivos electrónicos del resultado obtenido es el siguiente.

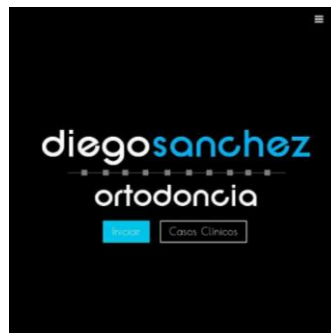


Imagen 28.- Resolución de pantallas 1024 x 1366.

4.1.3. Celular o smartphone

Para la realización las siguientes pruebas resolución de pantallas 414 x 736, se utilizó dispositivos electrónicos del resultado obtenido es el siguiente.



Imagen 29.- Resolución de pantallas 414 x 736.

4.2.- Prueba de satisfacción

Se realizaron las siguientes encuestas a los usuarios los resultados obtenidos son los siguientes.

Encuesta de satisfacción sistema dentalmedic.

Evaluación Administrador CDEO.

Criterios a evaluar	valor	evaluación
Página publicitaria	30	30
Rendimiento del sistema	30	30
Interfaces grafica de usuario	20	20
El proyecto supero sus expectativas	20	20
Calificación final	100	100

Tabla 12.-Evaluación Administrador CDEO

Encuesta de satisfacción sistema dentalmedic.

Evaluación Secretario.

Criterios a evaluar	valor	evaluación
Página publicitaria	30	15
Rendimiento del sistema	30	30
Interfaces grafica de usuario	20	15
El proyecto supero sus expectativas	20	10
Calificación final	100	70

Tabla 13.- Evaluación Secretario.

4.3.- Manual de Usuario

El manual de usuario se encuentra dentro del sistema, si desea visualizarlo puede utilizar una aplicación llamada código QR la cual le permitirá escanear el siguiente código.



Imagen 30.- Código QR.

CAPÍTULO V.- CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

La presente investigación se ha dedicado al estudio basado en construir un sitio web de publicidad para la clínica dental de ortodoncia por medio de programación en los lenguajes HTML, PHP, Javascript, JQuery, SQL y los frameworks Bootstrap y Laravel que permita al CDEO darse a conocer a posibles clientes a través de internet. A lo largo del documento se analizó con detalle los requerimientos del cliente y del sistema, programando un sistema web que cubrió satisfactoriamente las necesidades del cliente y el objetivo general.

En conclusión este proyecto le permite al CDEO darse a conocer a través de internet y a su vez contar con un sistema web que le permite controlar expedientes y citas por medio de formularios para la obtención de los datos necesarios.

5.2 Recomendaciones

Una vez concluido el presente trabajo, se considera interesante investigar el desarrollo de un sitio web con las mismas características pero utilizando otro framework como lo es Ruby on Rails.

Bibliografía

- Alarcón, V. (2006). *Desarrollo de Sistemas de Información*. Barcelona: Edicions UPC.
- Alvarez, M. A. (19 de julio de 2002). *desarrolloweb.com*. Recuperado el 22 de noviembre de 2017, de <https://desarrolloweb.com/articulos/844.php>
- Baker, R. (1994). *El modelo entidad-relación CASE METHOD*. U.S.A: Adison-Wesky Iberoamericana, S.A.
- Beati, H. (2015). *PHP creacion de paginas web dinamicas*. anonimo: alfaomega.
- Berners-Lee, T. (2017 de noviembre de 02). *World Wide Web Consortium W3C*. Recuperado el 22 de noviembre de 2017, de <http://www.w3.org/TR/2017/PR-html52-20171102/introduction.html#introduction>
- Bourdon, R. (anonimo de anonimo de 2017). *WampServer apache, PHP, MySQL en Windows*. Recuperado el 22 de noviembre de 2017, de <http://www.wampserver.com>
- Cabello, V. N. (2011). *Introducción a las bases de Datos Relacionales*. Madrid: vision libros.
- Cobo, G. P. (2005). *PHP y MYSQL Tecnologias para el desarrollo de aplicaciones web*. España: Diaz De Santos.
- Cortes. (1998). *Gerencia Efectiva*. Caracas: HCZ Consulying.
- Cortés Morales, R. (1998). *Introducción Al Análisis de Sistemas Y la Ingeniería de Software*. Universidad Estatal a Distancia.
- Date. (2001). *Introducción a los Sistemas de Bases de Datos*. Anonimo: Pearson.
- Durango, A. (2015). *Diseño Web con CSS*. Anonimo: Anonimo.
- Fontanals, C. R. (2004). *Informacion y Documentacion Digital*. Anonimo: Anonimo.
- Fowler, M. (1999). *Gota a Gota*. Mexico: anonimo.
- Galipienso, M. L. (2005). *Ingenieria del Software*. Madrid: Pearson.

- Gauchat, J. (2017). *HTML5 para Mentas Maestras*. anonimo: anonimo.
- Godoc, E. (2014). *SQL Los Fundamentos del lenguaje*. Anonimo: eni ediciones.
- Martin Fowler, K. S. (1999). *Gota a Gota*. Ciudad de Mexico: ADDISQN WESLEY LQNGMAN DE MEXICQ. SA DE C.V.
- Maza, M. A. (2001). *JavaScript*. anonimo: Inovación y cualificación.
- Menéndez, M. A. (20014). *eBooks*. Madrid: Libros.
- Niño, J. (2011). *Sistemas Operativos Monopuestos*. Madrid: Editorial Editex, S.A.
- Prescott, P. (2015). *HTML5*. Anonimo: Anonimo.
- Pressman, R. (2010). *Ingeniería del Software*. New York: MC Graw Hil.
- Ramez Elmasri, S. B. (2007). *Fundamentos de Bases de Datos*. Madrid: PEARSON EDUCACIÓN S.A.
- Sommerville, I. (2005). *Igenieria del Software*. Anonimo: Anonimo.
- Vertice, E. (2009). *Diseño Basico de Paginas Web en HTML*. Anonimo: Vertice.

ANEXOS

Acceder al sistema

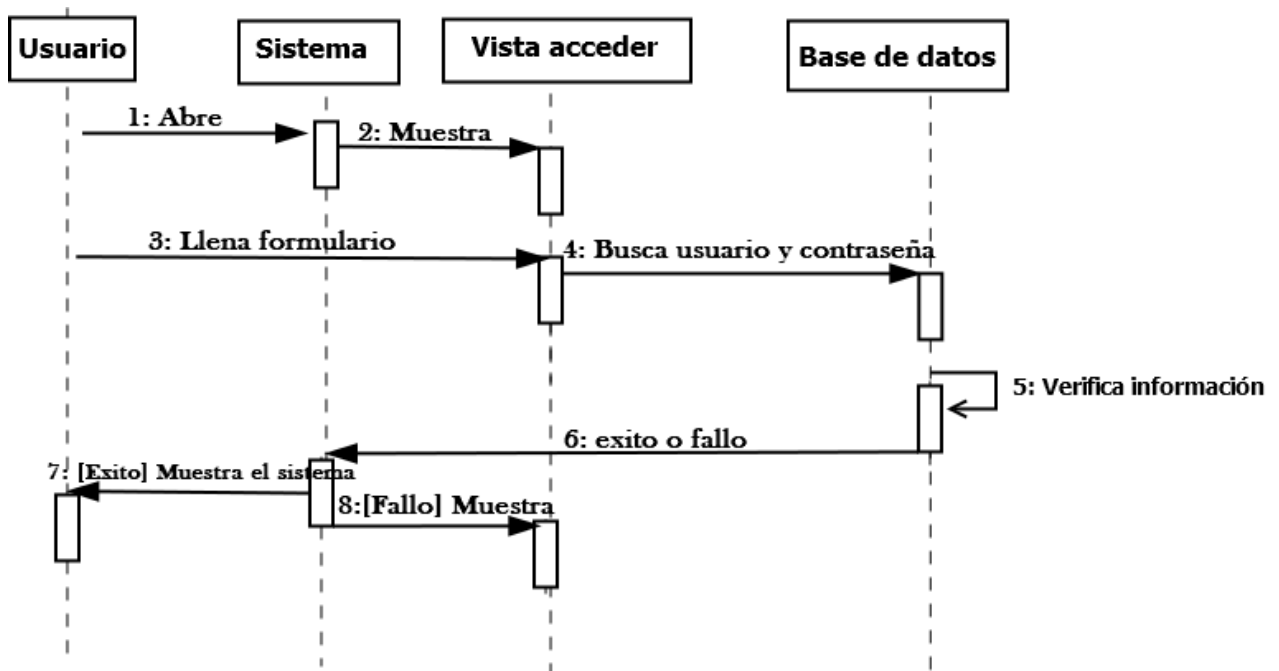


Imagen 11.- Acceder al sistema.

Registrar usuario.

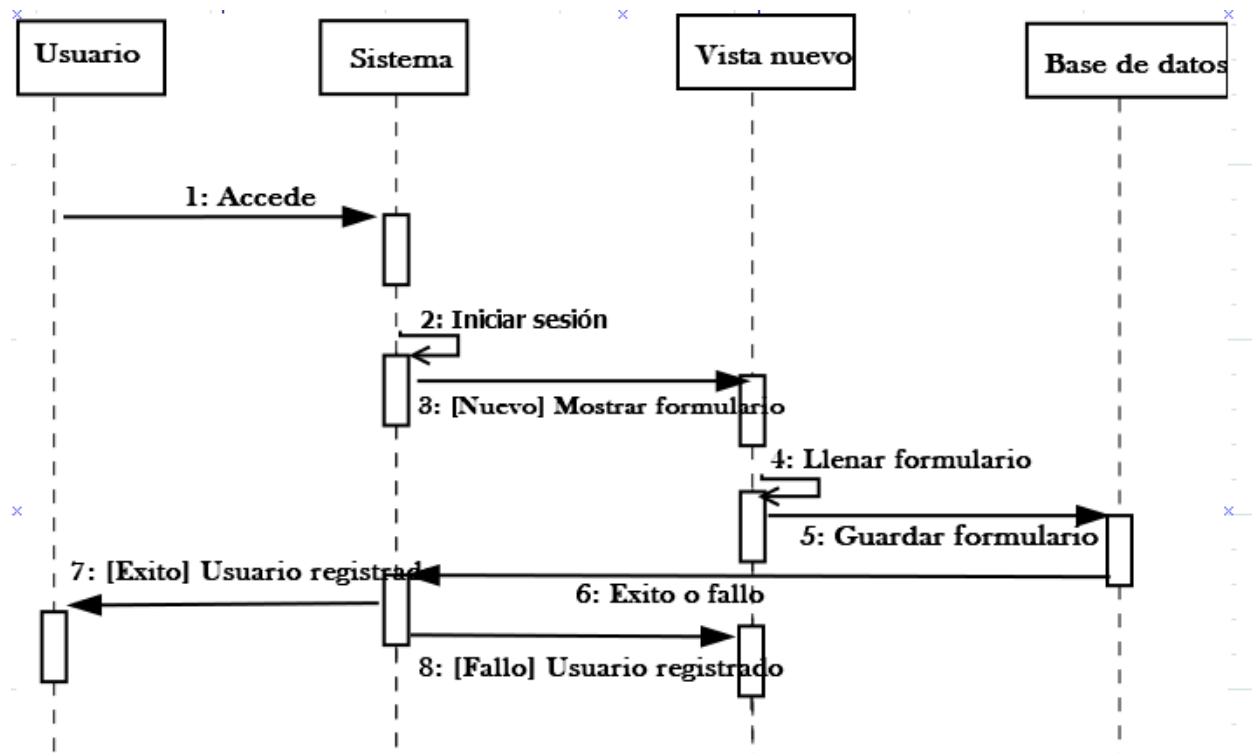


Imagen 12.-Registrar usuario.

Registro cita.

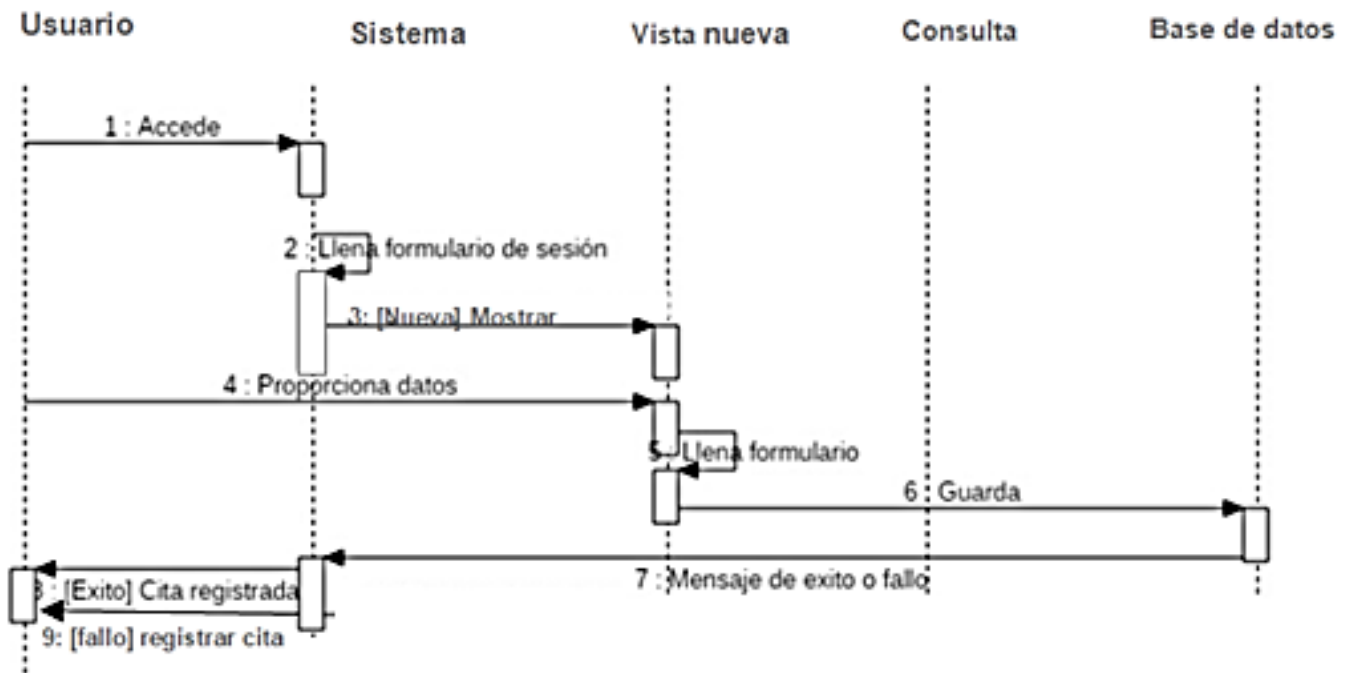


Imagen 13.- Registro cita.

Diagrama E-R

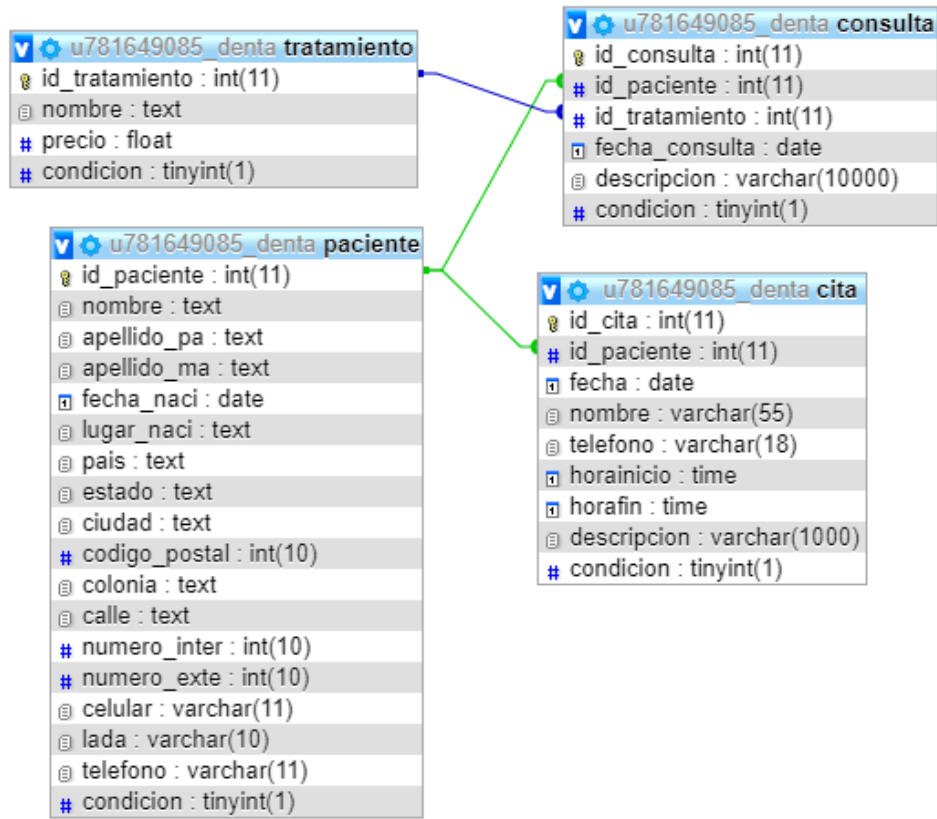


Imagen 26.- Diagrama E-R.