

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

FACULTAD DE INGENIERÍA

ESCUELA DE SISTEMAS

**DISERTACIÓN PREVIA A LA OBTENCIÓN DEL TÍTULO
DE INGENIERÍA EN SISTEMAS**

“Construcción de una aplicación web para la Gestión de Citas,
Tratamientos y Expediente Médico de los pacientes del Centro
Implantológico y de Especialidades Dental Access”

Nombre:

Nathaly Soledad Herrera Baldeón

Director: Ing. Fabián de la Cruz

Ciudad Quito, Año 2016

Dedicatoria

Este presente proyecto está dedicado principalmente a mis padres, por el apoyo que me han brindado en el transcurso de mi estudio universitario, también por los consejos brindados y el ejemplo que me han dado para nunca rendirme en mis objetivos establecidos y siempre buscar nuevas metas para superarme en la vida.

A mis mejores amigos que me acompañaron en todo momento especialmente en los momentos difíciles, que me aconsejaron y me ayudaron en temas personales y académicos.

A Dios por darme sabiduría, responsabilidad y paciencia para terminar mi carrera universitaria y lograr una meta más en mi vida.

Agradecimiento

Quiero agradecer al Centro odontológico Dental Access por darme la oportunidad de realizar mi disertación en su empresa y también por ayudarme en el proceso principalmente de elaboración de requerimientos y pruebas del sistema.

A mi director de tesis el ing. Fabián de la Cruz, y mis lectores el ing. Alfredo Calderón e ing. Beatriz Campos que me han ayudado en la realización de la disertación, dándome sus consejos para realizar y culminar una disertación de calidad.

A mi hermano por brindarme su apoyo y consejos principalmente en el trascurso de mis estudios y así ser una mejor persona y estudiante.

Tabla de contenidos

CAPITULO I: INTRODUCCIÓN.....	1
1.1. Datos de la Organización	2
1.2. Alcance	3
1.3. Objetivos	3
1.3.1. <i>Objetivo General</i>	3
1.3.2. <i>Objetivos Específicos</i>	3
1.4. Mapa de Procesos	4
1.5. Diagrama Jerárquico.....	5
CAPITULO II: FUNDAMENTO TEÓRICO	6
2.1. Metodología.....	6
2.1.1. <i>Metodología Estructurada</i>	6
2.1.2. <i>Metodología Orientada a Objetos</i>	8
2.1.3. <i>Metodologías ágiles</i>	9
2.1.4. <i>Selección de metodología</i>	10
2.2. Herramientas.....	13
2.2.1. <i>Modelado</i>	13
2.2.1.1. UML.....	14
2.2.2. <i>Lenguajes de Programación</i>	16
2.2.2.1. <i>Aplicación web</i>	18
2.2.2.2. <i>PHP</i>	19
2.2.2.3. <i>CSS</i>	21
2.2.2.4. <i>JavaScript</i>	22
2.2.3. <i>Base de Datos</i>	23
2.2.3.1. <i>PostgreSQL</i>	24
2.2.4. <i>Selección de las herramientas</i>	26
2.3. Framework.....	26
2.3.1. <i>Symfony</i>	26
2.4. Arquitectura de aplicación.....	29
2.4.1. <i>Tipos de arquitectura</i>	29
2.4.1.1. <i>Arquitectura Cliente servidor</i>	29
2.4.1.2. <i>Arquitectura de tres capas</i>	31
2.4.2. <i>Selección de arquitectura</i>	32
2.5. Patrón de diseño	33
2.5.1. <i>Modelo Vista Controlador (MVC)</i>	33
2.6. Resumen de las Herramientas Seleccionadas	34
CAPITULO III: CASO DE ESTUDIO	38
3.1. Primera Iteración	38
3.1.1. <i>Fase de Iniciación</i>	38
3.1.1.1. <i>Modelo de Negocio</i>	38

Actividades del Negocio	38
Descripción de la situación actual de la empresa	39
3.1.1.2. Requerimiento	39
Descripción del sistema	39
Requerimientos funcionales	40
Requerimientos no funcionales	40
Diseño Conceptual	41
Diagrama de casos de uso	41
Diagrama de clases	65
Requerimientos de hardware y software	67
Prototipo	67
3.1.2. Fase de Elaboración	70
3.1.2.1. Análisis y Diseño	70
Análisis del sistema	70
Diseño Arquitectónico	71
Diagrama de paquetes (MVC)	72
Diagrama de Secuencia	72
Diagrama Físico	84
3.1.3. Fase de Construcción	86
3.1.3.1. Implementación	86
Descripción de las clases	86
Interfaz	87
3.1.4. Fase de Transición	96
3.1.4.1. Pruebas	96
Resultados	103
3.1.4.2. Gestión de Cambios y Configuraciones	103
3.1.4.3. Entorno	104
3.2. Segunda Iteración	105
3.2.1. Fase de Iniciación	105
3.2.1.1. Requerimientos	105
Requerimientos funcionales	105
Requerimientos no funcionales	106
Diagrama de casos de uso	106
Diagrama de Clase	124
Prototipo	126
3.2.2. Fase de Elaboración	128
3.2.2.1. Análisis y Diseño	128
Diagramas de Secuencia	128
Diagrama Físico	135
Diagrama de Actividades	137
3.2.3. Fase de Construcción	138
3.2.3.1. Implementación	138
Descripción	139
Interface	140
3.2.4. Fase de Transición	150
3.2.4.1. Pruebas	150
Resultados	156
3.2.4.2. Despliegue	157
3.2.4.3. Entorno	158
CAPITULO IV: CONCLUSIONES Y RECOMENDACIONES	159

4.1. Conclusiones.....	159
4.2. Recomendaciones.....	160
<i>CAPITULO V</i>	163
Anexos	163
5.1. Manual de Usuario	163
5.2. Manual de Instalación	182
5.3. Glosario	189
Bibliografía.....	191

Tabla de Ilustraciones

ILUSTRACIÓN 1 - MAPA DE PROCESOS	5
ILUSTRACIÓN 2 - DIAGRAMA JERÁRQUICO	5
ILUSTRACIÓN 3 - CICLO DE VIDA EN CASCADA	7
ILUSTRACIÓN 4 - PRINCIPIOS DEL MANIFIESTO ÁGIL	9
ILUSTRACIÓN 5 - CICLO DE VIDA DE RUP	11
ILUSTRACIÓN 6 - ESQUEMA DE FUNCIONAMIENTO DE LAS PÁGINAS PHP	20
ILUSTRACIÓN 7 - FUNCIONAMIENTO DE LA BASE DE DATOS	23
ILUSTRACIÓN 8 - PATRÓN MVC	27
ILUSTRACIÓN 9 - ARQUITECTURA CLIENTE SERVIDOR	30
ILUSTRACIÓN 10 - ESTRUCTURA DE LA ARQUITECTURA DE TRES CAPAS	32
ILUSTRACIÓN 11 - MVC	34
ILUSTRACIÓN 12 - INTERFAZ VENTANA PRINCIPAL PACIENTE	68
ILUSTRACIÓN 13 - INTERFAZ DE INSERTAR PACIENTE	68
ILUSTRACIÓN 14 - INTERFAZ MODIFICAR PACIENTE	69
ILUSTRACIÓN 15 - INTERFAZ MOSTRAR Y ELIMINAR PACIENTE	69
ILUSTRACIÓN 16 – MVC	71
ILUSTRACIÓN 17 - VALIDACIÓN CAMPO VACÍO	87
ILUSTRACIÓN 18 - VALIDACIÓN EMAIL	87
ILUSTRACIÓN 19 - VALIDACIÓN CAMPO TELÉFONO	88
ILUSTRACIÓN 20 - VALIDACIÓN DEL NÚMERO DE DÍGITOS DE LA CÉDULA	88
ILUSTRACIÓN 21 - NOTIFICACIÓN DE QUE LA CÉDULA YA ESTÁ REGISTRADA	88
ILUSTRACIÓN 22 - NOTIFICACIÓN DEL QUE LA CÉDULA QUE SE ESTÁ BUSCADO NO EXISTE	89
ILUSTRACIÓN 23 - VALIDACIÓN DE NÚMERO DE CARACTERES	89
ILUSTRACIÓN 24 - NOTIFICACIÓN DE ELIMINACIÓN	89
ILUSTRACIÓN 25 - INTERFAZ PÁGINA PRINCIPAL PACIENTE	90
ILUSTRACIÓN 26 - INTERFAZ INGRESAR PACIENTE	90
ILUSTRACIÓN 27 - INTERFAZ MODIFICAR PACIENTE	91
ILUSTRACIÓN 28 - INTERFACE MOSTRAR Y ELIMINAR PACIENTE	91
ILUSTRACIÓN 29 - INTERFAZ PAGINA INICIAL DEL HORARIO DE ATENCIÓN	92
ILUSTRACIÓN 30 - INTERFACE INGRESAR HORARIO DE ATENCIÓN	92
ILUSTRACIÓN 31 - NOTIFICACIÓN DE COINCIDENCIA DE HORARIO	93
ILUSTRACIÓN 32 - INTERFACE MODIFICAR HORARIO DE ATENCIÓN	93
ILUSTRACIÓN 33 - INTERFACE MOSTRAR Y ELIMINAR HORARIO DE ATENCIÓN	94
ILUSTRACIÓN 34 - INTERFACE MOSTRAR Y ELIMINAR HORARIO DE ATENCIÓN POR DOCTOR	94
ILUSTRACIÓN 35 - INTERFACE PÁGINA PRINCIPAL FICHA MÉDICA	95
ILUSTRACIÓN 36 - INTERFACE MODIFICAR FICHA MÉDICA	95
ILUSTRACIÓN 37 - MOSTRAR Y ELIMINAR FICHA MÉDICA	96
ILUSTRACIÓN 38 - INTERFAZ VENTANA PRINCIPAL RESERVA DE CITAS	126
ILUSTRACIÓN 39 - INTERFAZ VENTANA RESERVAR CITA	126
ILUSTRACIÓN 40 - INTERFAZ VENTANA RESERVAR CITA (LADO DEL PACIENTE)	127
ILUSTRACIÓN 41 - INTERFAZ VENTANA MOSTRAR RESERVA DE CITAS POR PACIENTE	127
ILUSTRACIÓN 42 - INGRESO AL SISTEMA	140
ILUSTRACIÓN 43 - NOTIFICACIÓN DE ERROR DE USUARIO O CONTRASEÑA	141
ILUSTRACIÓN 44 - INTERFACE PÁGINA DE INICIO	141
ILUSTRACIÓN 45 - NOTIFICACIÓN USUARIO Y CONTRASEÑA EMAIL	142
ILUSTRACIÓN 46 - INTERFACE PÁGINA PRINCIPAL DE RESERVA CITA	142
ILUSTRACIÓN 47 - INTERFACE RESERVAR CITA (EL PACIENTE)	143
ILUSTRACIÓN 48 - INTERFACE RESERVAR CITA (USUARIOS DEL CENTRO ODONTOLÓGICO)	143
ILUSTRACIÓN 49 - NOTIFICACIÓN AL PACIENTE DE QUE YA TIENE CITA LA MISMA FECHA Y HORA	144
ILUSTRACIÓN 50 - MAIL DE CONFIRMACIÓN DE LA RESERVA DE CITA	144

ILUSTRACIÓN 51 - INTERFACE CONSULTAR RESERVA DE CITA	145
ILUSTRACIÓN 52 - INTERFACE CONSULTAR RESERVA DE CITAS DE UN DETERMINADO DOCTOR.....	145
ILUSTRACIÓN 53 - INTERFACE DE CANCELAR CITA.....	146
ILUSTRACIÓN 54 - CONFIRMACIÓN PARA CANCELAR LA CITA	146
ILUSTRACIÓN 55 - MAIL DE CANCELAR LA CITA.....	146
ILUSTRACIÓN 56 - INTERFACE PÁGINA PRINCIPAL DE LA FICHA DE LA CITA	147
ILUSTRACIÓN 57 - INTERFACE MODIFICAR FICHA DE LA CITA	147
ILUSTRACIÓN 58 - INTERFACE CONSULTAR Y ELIMINAR FICHA DE LA CITA	148
ILUSTRACIÓN 59 - PDF FICHA DE CITA ACUMULADA.....	149
ILUSTRACIÓN 60 - LINK DEL PDF FORMULARIOS.....	149
ILUSTRACIÓN 61 - PDF FORMULARIOS	150

Tabla de Tablas

TABLA 1- CUADRO DE METODOLOGÍAS ORIENTADA A OBJETOS	8
TABLA 2 - TIPOS DE METODOLOGÍA ÁGIL	10
TABLA 3 - HISTORIA DE LOS LENGUAJES DE PROGRAMACIÓN	17
TABLA 4 - ESQUEMA DE FUNCIONAMIENTO DE LAS PÁGINA PHP	22
TABLA 5 - LÍMITES GENERALES DE POSTGRESQL	25
TABLA 6 - RESUMEN DE HERRAMIENTAS.....	37
TABLA 7 - PRUEBAS DEL CASO DE USO PACIENTE	98
TABLA 8 - PRUEBAS DEL CASO DE USO HORARIO DE ATENCIÓN	101
TABLA 9 - PRUEBAS DEL CASO DE USO FICHA DE CITA	103
TABLA 10 - GESTIÓN DE CAMBIOS Y CONFIGURACIÓN	104
TABLA 11 - PRUEBAS CASO DE USO INGRESO AL SISTEMA	151
TABLA 12 - PRUEBAS CASO DE USO RESERVA CITA	153
TABLA 13 - PRUEBAS CANCELAR CITA	154
TABLA 14 - PRUEBAS DEL CASO DE USO FICHA DE LA CITA	156

Tabla de Diagramas

DIAGRAMA 1 - MODELO CONCEPTUAL	41
DIAGRAMA 2 - DIAGRAMA GENERAL.....	42
DIAGRAMA 3 - CASO DE USO GESTIONAR PACIENTE	43
DIAGRAMA 4 - CASO DE USO INGRESAR PACIENTE.....	44
DIAGRAMA 5 - CASO DE USO MODIFICAR PACIENTE	45
DIAGRAMA 6 - CASO DE USO ELIMINAR PACIENTE	46
DIAGRAMA 7 - CASO DE USO CONSULTA A DETALLE PACIENTE.....	47
DIAGRAMA 8 - CASO DE USO CONSULTA GENERAL PACIENTE	49
DIAGRAMA 9 - CASO DE USO GESTIONAR FICHA MÉDICA	50
DIAGRAMA 10 - CASO DE USO MODIFICAR FICHA MÉDICA	51
DIAGRAMA 11 - CASO DE USO ELIMINAR FICHA MÉDICA	52
DIAGRAMA 12 - CASO DE USO CONSULTA A DETALLE FICHA MÉDICA.....	53
DIAGRAMA 13 - CASO DE USO CONSULTA GENERAL FICHA MÉDICA	55
DIAGRAMA 14 - CASO DE USO HORARIO DE ATENCIÓN	56
DIAGRAMA 15 - CASO DE USO INSERTAR HORARIO DE ATENCIÓN	57
DIAGRAMA 16 - CASO DE USO MODIFICAR HORARIO DE ATENCIÓN.....	58
DIAGRAMA 17 - CASO DE USO ELIMINAR HORARIO DE ATENCIÓN	59
DIAGRAMA 18 - CASO DE USO ELIMINAR HORARIO DE ATENCIÓN	60
DIAGRAMA 19 - CASO DE USO CONSULTA A DETALLE POR DOCTOR EL HORARIO DE ATENCIÓN	62
DIAGRAMA 20 - CASO DE USO CONSULTA A DETALLE DE HORARIO DE ATENCIÓN.....	63
DIAGRAMA 21 - CASO DE USO CONSULTA GENERAL HORARIO DE ATENCIÓN	64
DIAGRAMA 22 - DIAGRAMA DE CLASE	66
DIAGRAMA 23 - DIAGRAMA DE PAQUETES.....	72
DIAGRAMA 24 - DIAGRAMA DE SECUENCIA INSERTAR PACIENTE.....	73
DIAGRAMA 25 - DIAGRAMA DE SECUENCIA MODIFICAR PACIENTE	74
DIAGRAMA 26 - DIAGRAMA DE SECUENCIA ELIMINAR PACIENTE.....	75
DIAGRAMA 27 - DIAGRAMA DE SECUENCIA CONSULTA A DETALLE PACIENTE	76
DIAGRAMA 28 - DIAGRAMA DE SECUENCIA CONSULTA GENERAL PACIENTE.....	76
DIAGRAMA 29 - DIAGRAMA DE SECUENCIA MODIFICAR FICHA MÉDICA.....	77
DIAGRAMA 30 - DIAGRAMA DE SECUENCIA ELIMINAR FICHA MÉDICA	78
DIAGRAMA 31 - DIAGRAMA DE SECUENCIA CONSULTA A DETALLE FICHA MÉDICA	79
DIAGRAMA 32 - DIAGRAMA DE SECUENCIA CONSULTA GENERAL FICHA MÉDICA.....	79
DIAGRAMA 33 - DIAGRAMA DE SECUENCIA INSERTAR HORARIO DE ATENCIÓN	80
DIAGRAMA 34 - DIAGRAMA DE SECUENCIA MODIFICAR HORARIO DE ATENCIÓN	81
DIAGRAMA 35 - DIAGRAMA DE SECUENCIA ELIMINAR HORARIO DE ATENCIÓN.....	82
DIAGRAMA 36 - DIAGRAMA DE SECUENCIA ELIMINAR HORARIO DE ATENCIÓN POR DOCTOR	82
DIAGRAMA 37 - DIAGRAMA DE SECUENCIA CONSULTA A DETALLE HORARIO DE ATENCIÓN POR DOCTOR	83
DIAGRAMA 38 - DIAGRAMA DE SECUENCIA CONSULTAR A DETALLE HORARIO DE ATENCIÓN	83
DIAGRAMA 39 - DIAGRAMA DE SECUENCIA CONSULTA GENERAL HORARIO DE ATENCIÓN	84
DIAGRAMA 40 - DIAGRAMA FÍSICO.....	85
DIAGRAMA 41 - CASO DE USO GENERAL.....	107
DIAGRAMA 42 - CASO DE USO INGRESO AL SISTEMA.....	108
DIAGRAMA 43 - CASO DE USO GESTIONAR FICHA DE CITAS	109
DIAGRAMA 44 - CASO DE USO MODIFICAR FICHA DE CITAS.....	110
DIAGRAMA 45 - CASO DE USO ELIMINAR FICHA DE CITAS	111
DIAGRAMA 46 - CASO DE USO CONSULTA A DETALLE POR PACIENTE LA FICHA DE CITAS	112
DIAGRAMA 47 - CASO DE USO CONSULTA A DETALLE FICHA DE CITAS	114
DIAGRAMA 48 - CASO DE USO CONSULTA GENERAL FICHA DE CITAS.....	115
DIAGRAMA 49 - CASO DE USO RESERVAR CITA.....	116
DIAGRAMA 50 - CASO DE USO INGRESAR RESERVA POR EL PACIENTE	117

DIAGRAMA 51 - CASO DE USO RESERVAR CITA.....	118
DIAGRAMA 52 - CASO DE USO CONSULTA A DETALLE RESERVA DE CITA POR PACIENTE.....	120
DIAGRAMA 53 - CASO DE USO CONSULTAR RESERVA DE CITA POR DOCTOR Y FECHA	121
DIAGRAMA 54 - CASO DE USO CONSULTA GENERAL RESERVAR CITA	122
DIAGRAMA 55 - CASO DE USO CANCELACIÓN DE CITA	123
DIAGRAMA 56 - DIAGRAMA DE CLASE COMPLETO.....	125
DIAGRAMA 57 - DIAGRAMA DE SECUENCIA INGRESO AL SISTEMA	128
DIAGRAMA 58 - DIAGRAMA DE SECUENCIA MODIFICAR FICHA DE CITA	129
DIAGRAMA 59 - DIAGRAMA DE SECUENCIA ELIMINAR FICHA DE CITA.....	130
DIAGRAMA 60 - DIAGRAMA DE SECUENCIA CONSULTA A DETALLE FICHA DE CITA POR PACIENTE.....	130
DIAGRAMA 61 - DIAGRAMA DE SECUENCIA CONSULTA A DETALLE FICHA DE CITA	131
DIAGRAMA 62 - DIAGRAMA DE SECUENCIA CONSULTA GENERAL FICHA DE CITA	131
DIAGRAMA 63 - DIAGRAMA DE SECUENCIA INGRESAR RESERVA DE CITA (PACIENTE)	132
DIAGRAMA 64 - DIAGRAMA DE SECUENCIA INGRESAR RESERVA DE CITA	133
DIAGRAMA 65 - DIAGRAMA DE SECUENCIA CONSULTA A DETALLE RESERVA DE CITA POR PACIENTE.....	134
DIAGRAMA 66 - DIAGRAMA DE SECUENCIA CONSULTA A DETALLE RESERVA DE CITA POR FECHA Y DOCTOR	134
DIAGRAMA 67 - DIAGRAMA DE SECUENCIA CONSULTA GENERAL RESERVA DE CITAS	135
DIAGRAMA 68 - DIAGRAMA DE SECUENCIA CANCELAR RESERVA DE CITA	135
DIAGRAMA 69 - DIAGRAMA FÍSICO COMPLETO	136
DIAGRAMA 70 - DIAGRAMA DE ACTIVIDADES RESERVAR CITA.....	137
DIAGRAMA 71 - DIAGRAMA DE ACTIVIDADES CANCELAR RESERVA DE CITA	138
DIAGRAMA 72 - DIAGRAMA DE DESPLIEGUE	157

CAPITULO I: INTRODUCCIÓN

La presente disertación se basa en el desarrollo de una aplicación web para el Centro Implantológico y de Especialidades Dental Access, que utiliza una alta tecnología para ofrecer tratamientos de máxima calidad buscando la satisfacción y bienestar de sus pacientes. Como cliente se observó que no llevaban un registro adecuado de fichas médicas, ni el seguimiento de los tratamientos del paciente y las reservas de citas no contaba con una buena coordinación.

Con el desarrollo de esta aplicación web se pretende automatizar la gestión de reservas de citas, tratamientos y expedientes médicos de los pacientes esperando alcanzar una mayor organización en el Centro Odontológico, con el fin de mejorar los servicios y con ellos alcanzar una mayor satisfacción de los clientes.

Para que cumpla con todos los requerimientos establecidos por el cliente, se realizaron pruebas y los cambios respectivos. Se elaboró manual de usuario y de instalación con el objetivo de dar a conocer el correcto funcionamiento y configuración del sistema.

Para la implementación del software se utilizó la metodología RUP, que está orientada a objetos y permite realizar cambios de manera sencilla, mediante la utilización de iteraciones.

Esta metodología está estructurada por flujos de trabajo los cuales son: modelado del negocio, requisitos, análisis y diseño, implementación, pruebas y despliegue que se encuentran distribuidas en las 4 fases: iniciación, elaboración, construcción y transición.

El framework Symfony facilitó el desarrollo de la aplicación web, mediante la automatización de tareas comunes. Los lenguajes de programación que se utilizó fue Php, JavaScript y la base de datos postgresql.

Finalmente, el patrón de diseño utilizado es el modelo vista controlador, que divide al aplicativo en las 3 capas, que son: el modelo, donde se encuentran las entidades y la conexión a la base de datos; en la vista donde se encuentra la interface que es visible por el usuario; y finalmente el controlador, que son todas las funciones que permite comunicar entre la capa del modelo y la vista.

Por el cual, se ha dividido la presente disertación con los siguientes capítulos: Capitulo I, es el presente capítulo; Capitulo II donde se presenta el marco teórico que sustenta la presente disertación, describiendo las herramientas que se utilizaron; Capítulo III presenta la

aplicación de la metodología es decir el caso de estudio donde detalla en cada una de las iteraciones la documentación del sistema; el Capítulo IV recoge las conclusiones y recomendaciones del presente trabajo; finalmente el Capítulo V se encuentra la bibliografía, anexos y glosario.

1.1. Datos de la Organización

Nombre: Centro Implantológico y de Especialidades Dental Access.

Ubicación: Voz Andes N39-34 y Juan Diguja, Centro Medico Livenza, Consultorio 401. Quito, Ecuador.

Actividad: Ofrecen una alta variedad de especialidades odontológicas como son: implantología oral, periodoncia, endodoncia, cirugía oral, ortodoncia – ortopedia y rehabilitación oral.

Características: El Centro Implantológico y de Especialidades Dental Access es un Centro Odontológico de alta tecnología que se enfocan en ofrecer tratamientos de máxima calidad buscando así la total satisfacción y bienestar de sus pacientes. Su principal fortaleza es el tratamiento personalizado bajo un excelente manejo de bioseguridad en donde trabajan bajo la premisa de calidad para la mayor satisfacción del cliente. Dental Access cuenta con un personal profesional en todas las especialidades graduados dentro y fuera del país.

El Centro ha adquirido los últimos adelantos tecnológicos en odontología para poder asegurar al paciente un diagnóstico y tratamiento exitoso con la más alta calidad.

Dental Access está bajo la dirección del Dr. Luis Hernández - Especialista en Implantología Oral y especialistas en cada una de las áreas. La Clínica Odontológica atiende de lunes a sábado de 8 de la mañana hasta las 6 de la tarde.

1.2. Alcance

El presente trabajo finalizará con el diseño, desarrollo e implantación de un Sistema de Gestión de Citas, Tratamientos y Expediente Médico de los pacientes del Centro Implantológico y de Especialidades Dental Access.

Después de que se hayan realizado las pruebas por parte de los usuarios involucrados, para comprobar su correcto funcionamiento, se realizará la implantación de la aplicación web en el servidor.

1.3. Objetivos

1.3.1. *Objetivo General*

Desarrollar un sistema computacional en el Centro Implantológico y de Especialidades Dental Access, mediante la utilización de tecnologías web que permita mejorar la gestión del proceso de citas, tratamientos y expediente médico de los pacientes.

1.3.2. *Objetivos Específicos*

- Aplicar de manera correcta la metodología RUP para el desarrollo de la aplicación web propuesta y en el mantenimiento del mismo.
- Analizar cada una de las fases de la metodología RUP, para aplicar en el desarrollo y diseño del sistema.
- Diseñar correctamente los procesos involucrados en la gestión de citas, tratamientos y expediente médico de los pacientes.
- Desarrollar la aplicación web en el tiempo establecido, que satisfaga los requerimientos del usuario.
- Desarrollar pruebas de la aplicación con los usuarios involucrados.
- Implantar la aplicación web en el servidor.
- Capacitar al personal de la clínica odontológica, para el uso del sistema.

1.4. Mapa de Procesos

El mapa de procesos, permite tener una perspectiva global de la empresa.

Procesos estratégicos: Son los procesos encargados de revisar las necesidades de los involucrados externamente, como son los accionistas y el mercado.

Para Dental Access los procesos estratégicos son la planificación operacional y presupuestal para administrar correctamente el presupuesto y no gastar más de lo necesario y también está la gestión de calidad que permite verificar si se ofrece un servicio de calidad en el Centro Odontológico.

Procesos de Cadena de Valor: Describe el proceso principal de la empresa o el funcionamiento, para cumplir el servicio que ofrece.

Procesos de Apoyo: Son los procesos que ayudan a cumplir el proceso principal de la organización.

Para el Centro Odontológico la gestión documental clínico, ayuda a llevar un historial médico a la organización.

Gestión administrativa apoya en la gestión de todos los procesos de la empresa.

Gestión de talento humano permite administrar al personal para que cumplan con sus actividades asignadas.

Gestión de Tecnología de la información ayuda a la empresa a supervisar las operaciones tecnológicas como es la página web.

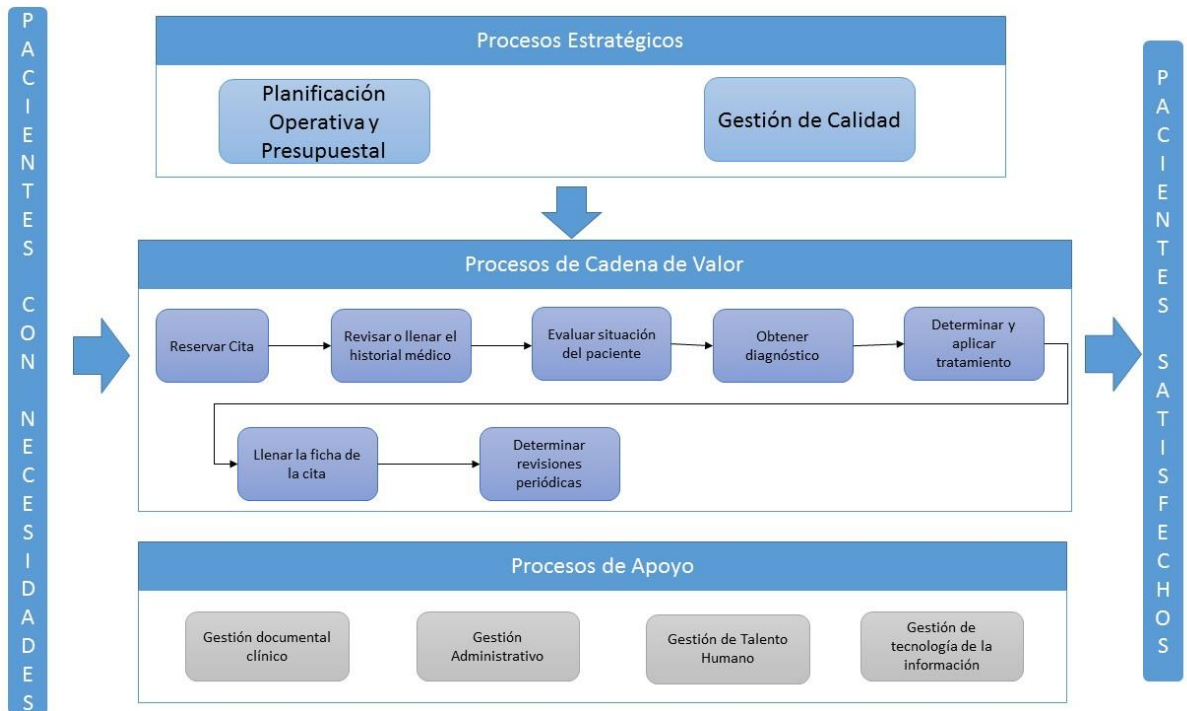


Ilustración 1 - Mapa de Procesos

1.5. Diagrama Jerárquico

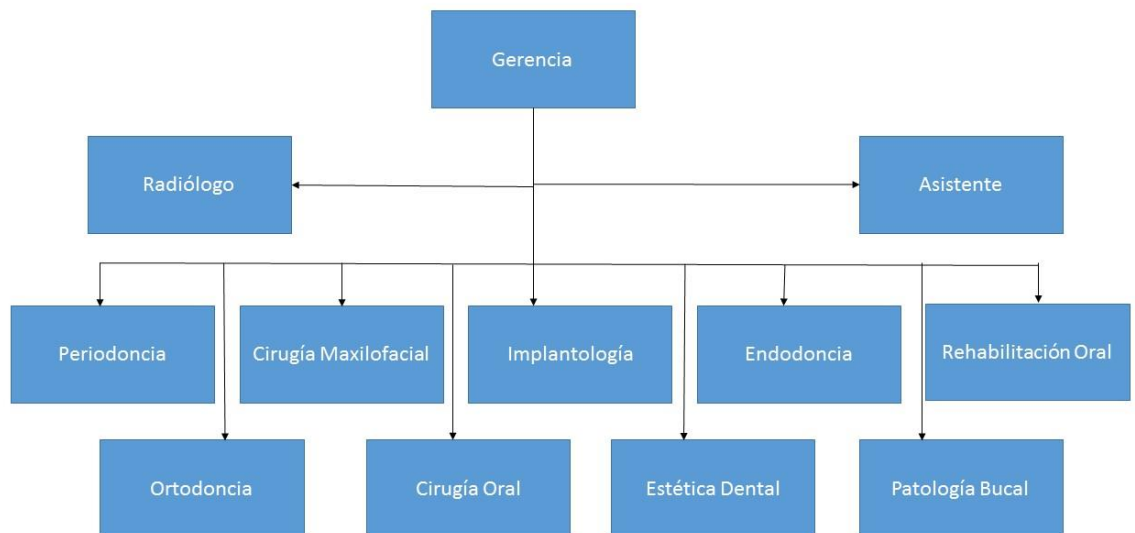


Ilustración 2 - Diagrama Jerárquico

CAPITULO II: FUNDAMENTO TEÓRICO

Este capítulo trata sobre la investigación técnica de varios aspectos que se van a emplear en el desarrollo de la aplicación web de la presente disertación, en la cual se describe los diferentes tipos de metodologías que existen y se explica a profundidad la metodología que se va utilizar. Posteriormente se detalla las herramientas que van a facilitar la realización del proyecto las cuales son de modelado, el lenguaje de programación y base de datos. También se expone la arquitectura de aplicaciones y sus tipos, finalmente se detalla los patrones de diseño a manejar.

2.1. Metodología

Metodología en el ámbito de la informática es un conjunto de métodos y reglas mediante el cual se realiza un análisis sistemático y teórico de cada uno de ellas para que así se pueda resolver los problemas que se presentan al momento de desarrollar un programa. Cada una de las metodologías consta de una serie de pasos que empieza por el planteamiento del problema y termina con un sistema que se encargue de resolver el problema (Universidad Técnica de Machala, 2010).

2.1.1. Metodología Estructurada

La metodología estructurada apareció en el año de 1970 para ayudar al diseño orientado a funciones creada por Constantino y Yourdon, y también en los años 1980 y 1990 ayudó al desarrollo de la metodología orientado a objetos.

La metodología estructurada, también llamada tradicional, construye los modelos de forma descendente por lo cual está basado en la “estructuración y descomposición funcional de problemas en unidades más pequeñas interrelacionadas entre sí” (Melder, 2014).

Esta metodología utiliza una herramienta CASE que permite aumentar la productividad en los programas debido a que se encuentra involucrado en el desarrollo e implantación del mismo. Pero, aunque aumenta la productividad también esta metodología elabora una documentación extensa debido a que tiene un alejamiento de los procesos con los

datos, y por esta razón es una metodología que lleva una gran cantidad de tiempo para su elaboración.

La clasificación de la metodología estructurada es:

- Orientada a procesos: Parte del modelo básico que es entrada, proceso y salida y partiendo de esto, aplica el método descendente para la descomposición funcional para indicar cada requisito.
- Orientada a datos: Igualmente que la anterior, parte del modelo básico, pero en función de eso define la estructura de datos.
- Mixta: Permite estandarizar los proyectos que se llevan a cabo en diferentes empresas (Universidad de Valladolid, s.f.).

Al ser una metodología estructurada su ciclo de vida es en cascada como podemos observar en la Ilustración 1, debido a que este ciclo de vida se basa en un conjunto de fases de forma secuencial por lo que no se puede pasar a la siguiente etapa si la anterior no ha terminado con todas sus especificaciones.

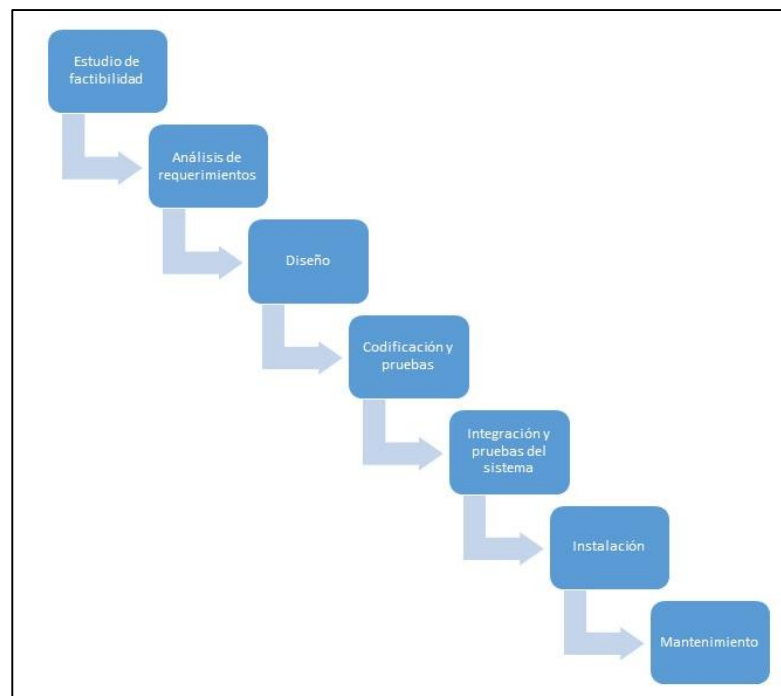


Ilustración 3 - Ciclo de vida en cascada

Elaborador por: Nathaly Herrera

2.1.2. Metodología Orientada a Objetos

La metodología orientada a objetos se creó para el manejo del desarrollo de software ya que mediante la utilización de objetos permite clasificar, describir, organizar y manipular de mejor manera dichos objetos en el software.

Al ser esta metodología orientada a objetos se constituye de 3 características principales las cuales son:

- Polimorfismo: Es la capacidad que tiene un objeto para “referenciar distintos elemento en distintas instancia de tiempo” (Cachero, Ponce de León, & Saquete).
- Herencia: Es la que permite que un objeto individual pueda extenderse a otro objeto ya existente en tiempo de ejecución (Cachero, Ponce de León, & Saquete).
- Encapsulamiento: Es la que permite colocar junto las variables o datos con los métodos y así ubicarles dentro del objeto (López, 2011).

UML es un lenguaje de modelado unificado que permite modelar software orientado a objetos. Este lenguaje consta de reglas de sintaxis, pero no de procedimiento por lo cual por sí solo no puede ser una metodología.

En la Tabla 1 se establecen los tipos de metodologías orientadas a objetos los principales son:

Metodología	Característica
Técnicas de modelado de objetos (OMT).	Es una de las primeras metodologías orientadas a objetos, en la cual se enfoca en la fase de análisis y diseño.
Metodología proceso de objetos (OPM)	Esta metodología apareció en 1995 y está enfocado en el desarrollo de sistemas generales y de información en particular, la cual para su modelado consta del diagrama de procesos.
Proceso racional unificado (RUP)	Esta metodología fue desarrollada en 1998, su ciclo de vida es iterativo y consta de varias fases para su desarrollo.

Tabla 1- Cuadro de metodologías orientada a objetos

(Kimble, Object Oriented Methodologies, 2008)

Elaborado por: Nathaly Herrera

Esta metodología utiliza objetos que son entidades, cosas o sujetos que consta de estado (abarca sus propiedades) y comportamiento por lo que ayuda a recibir, procesar y enviar mensaje entre los objetos (Kimble, Object Oriented Methodologies, 2008).

2.1.3. Metodologías ágiles

Las metodologías ágiles han ido evolucionando en el pasar de los años, para mejorar y competir con las metodologías tradicionales. Por lo que esta metodología permite el desarrollo de software centrándose en la flexibilidad debido a que se puede realizar cambios en cualquier momento sin afectar su proceso, en la rapidez de la elaboración de productos y en la mejora constante en todo el proceso del producto (Metodologías ágiles).

En el año del 2001 se reunieron varias entidades del campo de desarrollo ágil por lo cual llegaron a un acuerdo y crearon el manifiesto ágil con el objetivo de permitir a los grupos desarrollar software de manera rápida y efectuar cambios constantemente sin afectar el proceso que se está realizando.

En la ilustración 2 se establece los principios del manifiesto ágil los cuales son:

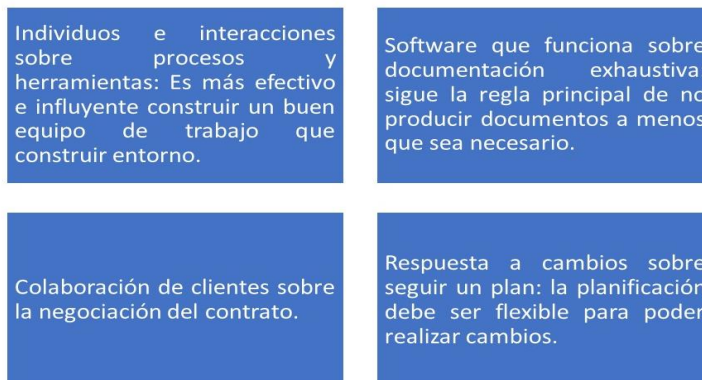


Ilustración 4 - Principios del manifiesto ágil

(Instituto Nacional de la Tecnología de la comunicación, 2009)

En la Tabla 2 se presenta los tipos principales de metodologías ágiles, que son:

Metodologías	Características
Scrum	Esta metodología desarrolla software mediante dos principales características que son la utilización de sprint (iteraciones), con una duración de 30 días. Y la realización de reuniones en todo el proceso de desarrollo.
Extreme Programming	Esta metodología está basada en la retroalimentación continua entre el cliente y el equipo de desarrollo, además se enfoca en el trabajo en equipo y en un ambiente adecuado.
Crystal Methodologies	Esta metodología se caracteriza por enfocarse en cada uno de los integrantes del equipo ya que de ellos depende el desarrollo del software, por lo que tienen reglas claves para el trabajo en equipo.
Dynamic Systems Development Method	Esta metodología está basada en la metodología RAD, debido a que se realiza de forma iterativa e incremental, pero lo que le diferencia es el tiempo de desarrollo de software (es más rápido) y que se puede efectuar cambios durante todo el proceso de desarrollo sin afectar al presupuesto o al producto final.

Tabla 2 - Tipos de metodología ágil

2.1.4. Selección de metodología

La metodología para el desarrollo de software de este proyecto es la metodología RUP (Proceso Racional Unificado).

La metodología RUP apareció en el año de 1998, creada por Grady Booch, Ivar Jacobson y James Jacobson. Pero esta metodología estuvo a disposición del público en general en el año de 1999 y su funcionamiento se enfoca en las personas, procesos y herramientas (Rational Unified Process equipo 1, 2012).

Esta metodología es un modelo orientado a objetos que permite el desarrollo de software, “mediante un proceso continuo de pruebas y retroalimentación, garantizando el cumplimiento de ciertos estándares de calidad” (Flores López & Santiago, s.f.).

Características de la metodología RUP:

- Casos de usos: Esta metodología está basada en caso de usos que son diagramas que describen como el usuario se comporta o interactuar con el software a desarrollar (Javier, 2014).
- Centrado en la arquitectura: La arquitectura del software es la organización de sus partes más importantes, por lo que le permite que todos los involucrados tengan una visión común y clara de todo el sistema (Serna).
- Iterativo o incremental: Esta metodología se basa en entregar al usuario varias versiones del producto en el transcurso del desarrollo del sistema, ya que cada fase consta de varias iteraciones que son los entregables, y así se va mejorando las versiones en las iteraciones de las diferentes fases.

En la Ilustración 3 se presenta el ciclo de vida de la metodología RUP:

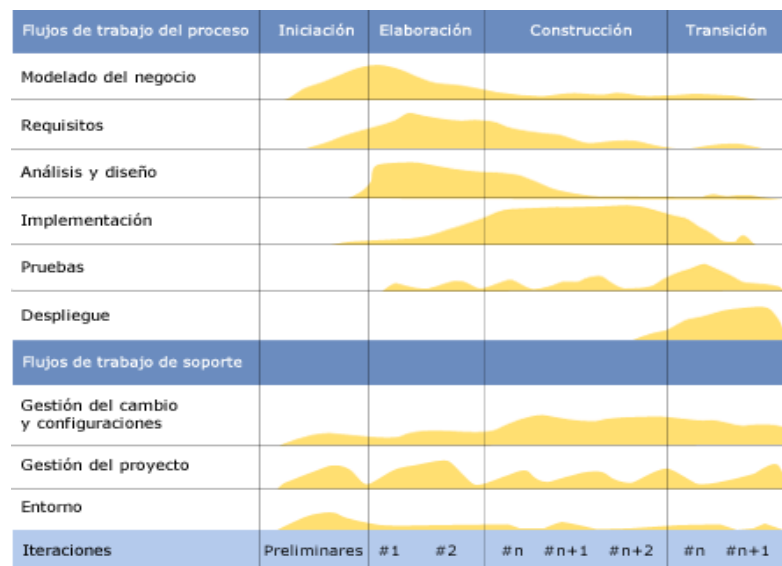


Ilustración 5 - Ciclo de vida de RUP

(Asamiya, 2011)

Esta metodología consta de cuatro fases las cuales son: inicio, elaboración, construcción y transición, para el desarrollo y diseño del sistema de gestión de citas y expediente médico se entregará en cada una de las fases avances, ya que es una metodología iterativa.

Lo que se va a realizar en cada fase es lo siguiente:

- **Inicio:** El objetivo de esta fase es identificar el ámbito del proyecto y establecer los límites del mismo también estima los costos y tiempo del proyecto.
Entregables: En esta fase se definirá el alcance del proyecto, se definirá los requerimientos del usuario y del sistema y se elaborarán los diagramas de caso de uso más importantes.
- **Elaboración:** El objetivo de esta fase es crear un plan íntegro para la fase de construcción y se identifica la arquitectura que soporte la visión del proyecto.
Entregables: Esta fase se analizará el dominio del problema, se realizará los casos de uso que permite definir la funcionalidad del sistema, se realizará el diseño de diagramas de clase, secuencia, actividades entre otros para cada módulo.
- **Construcción:** El objetivo de esta fase es que todos los componentes, características y requisitos deben estar implementados y funcionando correctamente (Rational Unified Process (RUP)).
Entregables: Esta fase se corregirán los diagramas y se completará la funcionalidad del sistema, es decir, se implementará todos los requisitos y componentes establecidos en el sistema, finalmente se integrará y se realizará pruebas en el sistema.
- **Transición:** El objetivo de esta fase es que el usuario final pruebe el software y también se corrijan los errores y defectos encontrados.
Entregables: En esta fase se corregirán los errores y defectos que se encontraron en las pruebas y se implantará el sistema.

Cada una de las fases consta de seis disciplinas de ingeniería que son:

- Modelaje de negocio
- Requisitos
- Análisis y diseño
- Implementación

- Pruebas o Despliegue

Tres disciplinas de soporte

- Gestión de la configuración y del cambio
- Gestión de proyectos
- Entorno

Ventajas de la metodología RUP:

- Se pueden realizar cambios de manera sencilla, debido a que es una metodología iterativa.
- Si no se cumple con los requerimientos del usuario al final, se puede crear una nueva fase de refinamiento, para agregar o modificar nuevas funcionalidades.
- El mantenimiento no es complejo
- Reduce el riesgo del proyecto, ya que se va evaluando en el transcurso del desarrollo.
- Se puede reutilizar el código.
- Garantiza el cumplimiento de ciertos estándares.

Desventajas de la metodología RUP:

- Se puede generar costos adicionales.
- Puede generar trabajo adicional, por lo que no es recomendado para trabajos pequeños.

2.2. Herramientas

2.2.1. Modelado

Dentro de la Ingeniería de Software, se considera el modelado como un elemento esencial debido a que es una técnica de ingeniería probada y aceptada que permite construir modelos¹ para entender de mejor manera la estructura y el comportamiento del

¹ Modelo: es una simplicidad de la realidad y proporciona los planos del sistema.

sistema y así producir un software de calidad y que cumplan con los requerimientos del usuario.

El modelado tiene cuatro objetivos principales para entender porque es importante modelar un sistema, los objetivos son:

- Nos permite visualizar como es y su funcionalidad del sistema.
- Nos ayuda a identificar la estructura y comportamiento del sistema.
- Nos facilita plantillas que nos ayuda de guía para la construcción del software.
- Finalmente los modelos son evidencia de las decisiones que se han tomado (Booch Grady, Rumbaugh, & Jacobson, 2000).

El tipo de modelado que se va usar para la realización de este proyecto es el de UML porque es un lenguaje sencillo de utilizar, es expresivo porque cubre todas las vistas necesarias para desarrollar el sistema (Booch Grady, Rumbaugh, & Jacobson, 2000) y finalmente UML nos ayuda a entender de mejor manera la estructura y funcionalidad del sistema con la utilización de una gran variedad de diagramas como son el conceptual, de estado, de componentes entre otros, por lo que es el más recomendado utilizar en el desarrollo de una aplicación web.

2.2.1.1. UML

En octubre de 1994 comenzó a formarse el lenguaje UML con la unión de Rumbaugh en la compañía Rational fundado por Booch, ya que su objetivo era fusionar al método Booch y al OMT (Object Modelling Tool) y así se obtuvo el primer borrador del lenguaje UML en 1995, pero en esa misma época también se unió a la compañía Jacobson, por lo que entre los tres empezaron a obtener ideas y así se obtuvo la primera versión de UML.

UML “es un lenguaje estándar para escribir planos de software por lo que se utiliza para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software” (Booch Grady, Rumbaugh, & Jacobson, 2000). UML es parte del OMG que es un estándar tecnológico sin fines de lucro.

Los objetivos de UML son:

- UML es un lenguaje para visualizar: ya que permite mostrar al usuario de una forma gráfica todo el sistema para que así se pueda entender de mejor manera.
- UML es un lenguaje para especificar: debido a que UML especifica las características del sistema antes de desarrollarlos y construirlos (Orallo, s.f.).
- UML es un lenguaje para construir: UML no es un lenguaje de programación, pero puede conectarse de forma directa a los diferentes lenguajes. Por ejemplo, un modelo UML conceptual al momento de conectarse con un lenguaje se puede obtener la base de datos. Para la realización de esto, se utiliza la ingeniería directa que es la generación de código a partir de un modelo o viceversa la ingeniería inversa que a partir del código se puede obtener el modelo (Booch Grady, Rumbaugh, & Jacobson, 2000).
- UML es un lenguaje para documentar: debido a que documenta todas las características del sistema, su funcionalidad y su arquitectura.

Los diagramas UML es la representación gráfica de los elementos y el comportamiento que compone un sistema. Y está conformado por 9 diagramas los cuales son:

1. Diagrama de clase: Está compuesto por un conjunto de clases, interfaces y sus relaciones.
2. Diagrama de objetos: Está compuesto por un conjunto de objetos y sus relaciones.
3. Diagrama de caso de uso: Está compuesto por un conjunto de casos de uso, actores y su relación entre ellos.
4. Diagrama de secuencia: Es un diagrama de iteraciones² en la cual se va incluyendo los mensajes de forma ordenada que pueden ser enviados entre ellos.
5. Diagrama de colaboración: Es un diagrama de iteraciones que muestra la estructura de los objetos que se encargan de enviar y recibir mensajes.
6. Diagrama de estado: Está compuesto por estados, transiciones y eventos, y se basa en la máquina de estados.
7. Diagrama de actividades: Es muy parecido al diagrama de estados con la diferencia que muestra un flujo de actividades que conforma el sistema.

² Diagrama de iteraciones: muestra una iteración, que consta de un conjunto de objetos y sus relaciones incluyendo los mensajes que pueden ser enviados.

8. Diagrama de componentes: Está compuesta por un conjunto de componentes y su dependencia entre ellos.
9. Diagrama de despliegue: Indica los componentes que se encuentran dentro de los nodos de procesamiento en tiempo de ejecución (Booch Grady, Rumbaugh, & Jacobson, 2000).

2.2.2. Lenguajes de Programación

El lenguaje de programación es muy importante porque ayuda a comunicarse entre las computadoras y las personas, sobre las ideas de un algoritmo y así resolver problemas particulares. Por lo que es muy importante escoger el lenguaje de programación adecuado de acuerdo a la funcionalidad del software que se va a desarrollar.

Los lenguajes de programación aparecieron en los años cuarenta, pero fueron los lenguajes de código de máquina de las computadoras originales, posteriormente fueron apareciendo una gran variedad de lenguajes de alto nivel de los cuales algunos han ido evolucionando en cambio otros han desaparecido en el pasar de los años, como se puede observar en la Tabla 3.

Evolución de lenguajes de programación más importantes:

LENGUAJES	AÑO
Lenguaje ensamblador	1950-1955
Lenguajes experimentales	
Lenguajes de alto nivel	
FORTRAN	1956-1960
ALGOL 58 y 60	
COBOL LISP	
FORTRAN IV	1961-1965
COBOL 61	
ALGOL 60	
SNOBOL	
APL	
PL/I	1966-1970

FORTRAN 66	
COBOL 65	
ALGOL 68	
SNOBOL 4	
SIMULA 67	
BASIC	
APL/360	
PASCAL	
COBOL 74	1971-1975
PL/I	
ADA	
FORTRAN 77	1976-1980
PROLOG	
C	
ADA95	
EIFFEL	
JAVA	1981-2000
HASKELL	
VISUAL BASIC	

Tabla 3 - Historia de los lenguajes de programación

(Tucker & Noonan, 2003)

Elaborado por: Nathaly Herrera

Los lenguajes de programación están clasificados por varios paradigmas, esto permite ubicar a los lenguajes en diferentes grupos de acuerdo a características, funcionalidades y aplicaciones iguales, los paradigmas son:

- La programación imperativa: Este paradigma se basa en una serie de pasos que están agrupados en bloques como los bucles y cada uno de los cuales realizan cálculos en la cual tiene una entrada y se produce una salida. Ejemplo: Fortran, C, C++, Cobol y PHP.

- Programación Orientada a Objetos: Este paradigma utiliza un conjunto de objetos que llevan mensajes mediante la interacción entre ellos. Ejemplo: Java, C++, Eiffel y PHP.
- Programación funcional: Este paradigma se basa en funciones que interactúan una con las otras y utilizan principalmente la recursividad. Las funciones constan de una entrada y se obtiene un resultado. Ejemplo: Lisp, Schema, Haskell y PHP.
- Programación Lógica: Este paradigma consta de un conjunto de declaraciones lógicas³ que se utiliza en las funciones y así conseguir el resultado. Ejemplo Prolog (Tucker & Noonan, 2003).
- Programación Guiada por Eventos: Este paradigma está basado en un conjunto de eventos que se aplican en el sistema. Ejemplo: Visual Basic y Java.
- Programación Concurrente: Este paradigma utiliza el paralelismo entre tareas y permite resolver los problemas de comunicación que se da al momento de compartir información y sincronización entre procesos. Ejemplo: Linda 1980 y Fortran de alto rendimiento 1995.

2.2.2.1. Aplicación web

Se desarrolla una aplicación web porque permite que los usuarios puedan utilizar esta aplicación en distintos navegadores y dispositivos en cualquier momento.

La aplicación web es un sitio, una aplicación o una herramienta que se implementa en un determinado lenguaje soportado por navegadores web por lo que permite que los usuarios puedan manejar esta aplicación en cualquier momento, lugar y desde cualquier dispositivo, ingresando a un servidor web mediante la utilización de internet o de una intranet a través de un navegador.

Para la implementación de estas aplicaciones web están disponibles varios lenguajes de programación que son:

- PHP
- C# y Visual Basic con ASP/ASDP.NET
- Java con Java Servlets y JavaServer Pages (JSP)

³ Declaración Lógica: Es una declaración, que está formada por dos condiciones de las cuales una depende de otra (Bueno saber, 2014).

- JavaScript
- Perl
- Ruby
- Python

2.2.2.2. *PHP*

El lenguaje de programación aplicada a la realización de la aplicación web es PHP porque es fácil de usar, por su portabilidad, por su rendimiento, entre otras cualidades.

Rasmus Lerdorf fue un programador nacido en Groenlandia, en 1994 creó el lenguaje de PHP, con el objetivo de realizar un seguimiento de los visitantes de página personal. Esto fue desarrollado con Perl y posteriormente con el lenguaje C, con el objetivo de obtener mayores funcionalidades, por lo que PHP se conocía como Personal Home Page. En 1995 se creó la versión PHP 1, posteriormente en 1995 y 1997 apareció PHP 2, y así en el pasar de los años ha ido evolucionando PHP, con varias mejoras. En la actualidad se encuentra en la versión 7 que fue lanzada en el 2015, esta versión tiene mejoras de rendimiento y en funciones (Cobo, Gómez, Pérez, & Rocha, 2005).

PHP es un lenguaje interpretado del lado del servidor y está compuesto de una secuencia de comandos del servidor que particularmente está diseñado para la Web. En la Ilustración 4 se establece el funcionamiento de las páginas PHP, en el cual el código PHP está integrado en el código HTML y es ejecutado por un servidor Web que antes de ser mostrado al cliente el contenido es interpretado (Welling & Thomson, 2005).



Ilustración 6 - Esquema de funcionamiento de las páginas PHP

(Alvarez, 2001)

Característica de PHP:

- Rendimiento: El lenguaje de PHP sirve a millones de usuarios al día mediante la utilización de un servidor, por lo que es un lenguaje eficaz.
- Integración a la Base de datos: PHP para poderse conectar a todos los sistemas de base de datos consta de una conexión propia (Welling & Thomson, 2005).
- Multitud de extensiones: PHP para dar mayor versatilidad y flexibilidad a los usuarios por lo que consta de varias extensiones que ayuda a utilizar nuevas bases de datos, protocolos, enlaces a librerías entre otros (Cabezas & González, 2015).
- Costo: PHP no tiene ningún costo, por lo que se puede descargar de su página oficial que es: www.php.net.
- Fácil de usar: PHP es un lenguaje fácil de utilizar y aprender debido a que la mayoría de funciones están disponibles por defecto como la de la base de datos, para la realización de programas y también se basa en el lenguaje C y Perl.
- Compatibilidad con el enfoque orientado a objetos: PHP desde la versión 5 consta de funciones orientadas a objetos como la herencia entre otras.

- Portabilidad: PHP se ejecuta en una gran cantidad de plataformas de sistemas operativos y servidores existentes.
- Licencia Open Source: PHP tiene licencia de código abierto por lo que es libre de descargarse, ser modificado y agregado elementos por el usuario.
- Gran Comunidad de Apoyo: Consta de la empresa Zend Technologies, que es responsable del lenguaje PHP, por lo que ofrece asistencia. También al ser código abierto, tiene gran variedad de foros, blog que ayudan a los usuarios a resolver sus problemas.

2.2.2.3. CSS

CSS (cascading style sheets) es un lenguaje de hojas de estilo que describe la interface de un documento HTML que va ser presentado al usuario. CSS permite separar el contenido y la presentación de la aplicación y es útil utilizar en aplicaciones web para que sean amigables (Eguiluz, Introduccion CSS, s.f.).

Las hojas de estilo fueron creadas en el año de 1970, después de los lenguajes de etiquetas SGML. El organismo W3C (World Wide Web Consortium) el cual está encargado de crear todos los estándares que estén relacionados a la web, propuso la creación de un lenguaje que se encargue de las hojas de estilo para HTML. En 1995, W3C decidió utilizar CSS en HTML (Eguiluz, Introduccion CSS, s.f.).

Las características principales de CSS son las siguientes:

- Los estilos se pueden almacenar en un archivo tipo css.
- CSS ahorra tiempo debido a que puede controlar el diseño de varias páginas a la vez.
- Define los estilos de la página ya sea el diseño, las posiciones, tamaños de los diferentes componentes que utiliza html.

2.2.2.4. JavaScript

JavaScript es un lenguaje de programación interpretado y orientado a objetos que permite realizar páginas web de manera iterativa, al ser un lenguaje script permite de manera más rápida y fácil codificar en otros lenguajes como en HTML, C entre otros.

JavaScript fue creado por Brendan Eich en 1995 llamada LiveScript, principalmente por la necesidad de que una aplicación web se ejecute rápidamente por ejemplo al momento de indicar errores. Posteriormente la empresa Netscape donde trabajaba Eich firmó una alianza con Sun Microsystems para el desarrollo de un nuevo lenguaje que le pusieron el nombre de Javacript, al tener la primera versión éxito se continuó mejorando y sacando nuevas versiones hasta la actualidad (Eguiluz, Introducción a JavaScript, s.f.).

Javacript posee una librería llamada jquery que facilita la programación de la misma ya que maneja eventos, animaciones y validaciones de una manera más sencilla.

Javascript tiene las siguientes características:

- Puede ser ejecutado sin la necesidad de otro programa para su visualización.
- Es un lenguaje que trabaja con objetos por el cual posee eventos.
- JavaScript es seguro porque no envía datos al servidor ni al ordenador.
- JavaScript amplía las capacidades de los documentos HTML (Kolbeck, 1997).
- Javascript no es igual que java como se muestra en la Tabla 4:

JavaScript	Java
Se interpreta por el cliente	Se compila por el programador y se ejecuta por el cliente
El código se encuentra en HTML	Utiliza el código Applet
Tipo de datos muy sencillos	Reglas estrictas para el tipo de datos
Es basado en objetos por el cual utiliza objetos pero no posee clases ni herencia.	Orientado a objetos tienen clases y herencia.
La referencia de objetos se verifica en tiempo de ejecución.	En cambio en Java es en tiempo de compilación.

Tabla 4 - Esquema de funcionamiento de las página PHP

(Kolbeck, 1997)

2.2.3. Base de Datos

La Base de datos es importante porque permite que la aplicación tenga centralizado todos los datos de operación y así al momento que se use dicha información sea más sencillo de consultar y obtener todos los datos almacenados.

La Base de datos es un conjunto de información que está organizada de forma estructurada por lo que sencillamente se puede acceder, gestionar, eliminar y actualiza, como se puede observar en la Ilustración 5 el funcionamiento de la BDD (Leake & Rouse, 2005).

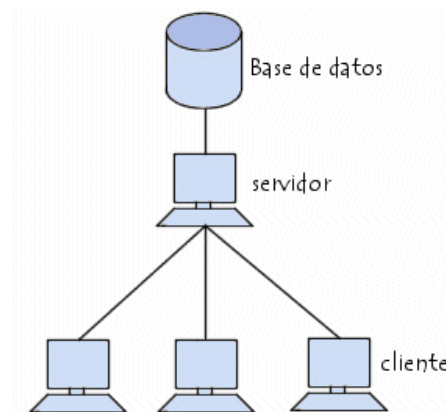


Ilustración 7 - Funcionamiento de la Base de datos

(Kioskea .net, 2014)

Las principales bases de datos son:

- PostgreSQL
- Mysql
- Oracle
- Access
- mSQL

Características de la Base de datos:

- Mediante la utilización de la base de datos, permite que los usuarios ingresen con mayor facilidad a los datos.
- Varios usuarios pueden acceder a los datos, al mismo tiempo.

- Si se utiliza una base de datos local, solo se puede manejar en un equipo, pero si se tiene distribuida se pueden acceder a los datos mediante una red.
- La información se encuentra organizada, por lo que es más sencillo y rápido realizar una búsqueda.

Un sistema de administración de base de datos (DBMS) es una colección de programas que controla la estructura y el acceso a la base de datos (Coronel, Morris, & Rob, 2011).

Ventajas del sistema de administración de base de datos:

- Los datos se comparten mejor: DBMS permite que los usuarios finales puedan acceder y manejar los datos de forma sencilla.
- Mejor seguridad de datos: El problema de seguridad es mayor cuando existe una gran cantidad de usuarios accediendo a la base de datos por lo que cada motor de base de datos invierte dinero y trabajo para solucionar este problema y ofrecer mayor seguridad (Coronel, Morris, & Rob, 2011).
- Mínima inconsistencia de datos: Se produce cuando varias versiones, aparecen en diferentes partes, por ejemplo, si se ingresa en la base de datos el nombre de una persona dos veces, pero de diferente estructura el nombre se produce inconsistencia, pero se puede evitar este problema si la base de datos se diseña de manera correcta.
- Mejor acceso a datos: DBMS permite que los usuarios puedan realizar consultas de cierta información de forma fácil y rápida.

2.2.3.1. *PostgreSQL*

La base de datos para este proyecto es PostgreSQL porque con este DBMS es más sencillo analizar los datos, es de multiplataforma, tiene una herramienta gráfica sencilla para la administración de la base datos y es gratuita.

PostgreSQL apareció en 1986 como un proyecto de investigación de la Universidad de Berkeley en California, encaminado por el investigador de base de datos de gran influencia Michael Stonebraker. PostgreSQL usaba un lenguaje de consultas llamado Postquel, con mayor profundidad de expresiones que el dominante SQL. En 1995 dos

estudiantes remplazaron a Postquel con subconjuntos extendidos de SQL. Y en el pasar de los años PostgreSQL ha ido evolucionando y obteniendo varias versiones como el 2005 la versión 8 que se creó puntos de restauración, y así hasta la actualidad que se encuentra en la versión 9.3 que posee soporte por la comunidad (2ndQuadrant, s.f.).

“PostgreSQL es un sistema de administración de base de datos relacionales orientada a objetos” (Borja, 2015).

Características:

- La licencia es libre por cual los usuarios pueden usar, distribuir y modificar PostgreSQL de manera libre y gratuita (Borja, 2015).
- Es desarrollado en C, por lo que puede ser compilado en la mayoría de los sistemas operativos.
- Maneja tipo de datos habituales, objetos binarios, imágenes, sonidos y videos.
- Es escalable tanto en los usuarios como en la cantidad de datos, por lo que puede trabajar hasta 4 Terabytes de datos (Borja, 2015).
- Tiene una arquitectura cliente-servidor.
- Soporta acceso y soporte de base de datos en Java, Python, Perl, PHP entre otros.
- Los lectores y escritores no se bloquean por lo que posee un diseño de alta concurrencia (2ndQuadrant, s.f.).

En la Tabla 5 se establece los límites de PostgreSQL con características esenciales:

Límites	Valor
Tamaño máximo de base de datos	Ilimitado
Tamaño máximo de tabla	32TB
Tamaño máximo de fila	1,6TB
Tamaño máximo de campo	1GB
Máximo de filas por tabla	Ilimitado
Máximo de columnas por tabla	250-1600
Máximo de índice por tabla	Ilimitado

Tabla 5 - Límites generales de PostgreSQL

(Borja, 2015)

Elaborado por: Nathaly Herrera

2.2.4. Selección de las herramientas

Después de analizar las herramientas que se requieren, en la construcción la aplicación para la presente disertación y de acuerdo a lo revisado en los numerales 2.2.1, 2.2.2 y 2.2.3 se utilizarán las siguientes herramientas:

Modelado: UML

Lenguaje de Programación: PHP, javascript y CSS

Base de Datos: PostgreSQL

2.3. Framework

Framework es un conjunto de componentes, herramientas, librerías que está formado de un diseño reutilizable debido a que se encuentra encapsulado las tareas en módulos genéricos para que finalmente pueda facilitar el desarrollo de la aplicación web (Eguiluz, s.f.).

Utilizar framework en el desarrollo de aplicaciones es importante:

- Porque proporciona código más fácil de entender y mantener.
- Porque permite automatizar tareas que son comunes en varias aplicaciones como es el CRUD (insertar, modificar, eliminar y consultar).
- Porque facilita en el parte de la programación debido a que “encapsula las operaciones complejas en instrucciones sencillas” (Potencier & Zaninotto, s.f.).

Para aplicaciones web existe una gran variedad de tipos de framework, por lo que para este proyecto de tesis se va utilizar Symfony porque tiene varios componentes que facilitan y se adecuan a la aplicación que se va realizar como es el CRUD, el login y enviar mail.

2.3.1. *Symfony*

Symfony es un Framework diseñado para facilitar el desarrollo de aplicaciones web, mediante la utilización de sus características permite automatizar las tareas que son comunes para que se pueda enfocar el desarrollador en la parte compleja de la aplicación.

Symfony fue elaborado por Fabien Potencier en octubre del 2005. Fabien es presidente de una empresa francesa de desarrollo de aplicaciones web. Para su elaboración Symfony fue perfilado en el trabajo del framework Mojavi que está fundamentado en el patrón de MVC. Dicho Framework fue utilizado por primera vez en el comercio electrónico de lencería. Después de manejar en varios proyectos y tener éxito Fabien publicó bajo la licencia de software libre (Potencier & Zaninotto, s.f.).

Symfony es desarrollado con php, y es compatible con una gran variedad de base de datos las cuales son: Mysql, Postgresql, Oracle y Sql Server y es multiplataforma debido a que se puede ejecutar tanto en Linux como Windows entre otras (Potencier & Zaninotto, s.f.).

“Symfony es un conjunto de componentes de PHP, un marco de aplicación web, una filosofía y una comunidad en la cual todos trabajan juntos” (Symfony, s.f.).

- Componentes de Symfony: Son reutilizables para construir las aplicaciones.
- Comunidad de Symfony.
- Filosofía de Symfony: Utiliza las mejores prácticas y estándares de programación

Symfony está basado en el patrón de MVC (modelo, vista y controlador), como se presenta en la Ilustración 6.

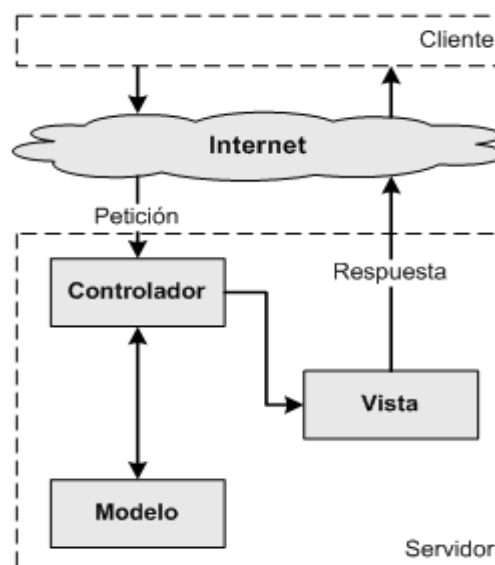


Ilustración 8 - Patrón MVC

(Potencier & Zaninotto, s.f.)

- El modelo es la lógica del negocio, que se encarga de buscar la información que requiere la aplicación, se comunica con la base de datos para obtener la información. En dicho Framework el almacenamiento de datos se realiza mediante la utilización de objetos por lo que no se accede de forma evidente a la base de datos por esta razón posee un alto nivel de abstracción (Potencier & Zaninotto, s.f.).

El componente que se encarga de manejar el modelo en el Framework de Symfony es la capa de tipo ORM (object/relation mapping) es un modelo que está enfocado en la transformación de las tablas de la base de datos en entidades que facilita al acceso de datos para el desarrollador (Gonzales, s.f.). Aplicar ORM tiene la ventaja principal de la reutilización permitiendo llamar a objetos de datos desde varias aplicaciones.

Esta capa consta de dos tipos que son los más utilizados los cuales son Propel y Doctrine.

Propel: Es de código abierto y para acceder a la base de datos utiliza un conjunto de objetos.

Doctrine: Consta de varias bibliotecas de PHP en la cual están enfocados en el almacenamiento de base de datos y mapeo de objetos (Doctrine, 2006-2016).

- Vista es la interface de la aplicación en donde el usuario puede manipular o interactuar con el software, debido a que el modelo es transformado en una página web que sea amigable.

Para la realización de la vista Symfony utiliza principalmente la plantilla twig que es un motor moderno de php. Esta plantilla tiene tres principales características las cuales son:

1. Rápido: Twig compila la plantilla de Php de forma optimizada y sin formato.
2. Seguro: Twig utiliza sandbox que permite evaluar el código por lo que el usuario puede modificar el diseño de la plantilla con confianza.
3. Flexible: Twig permite que el desarrollador defina sus propias etiquetas y filtros personalizados (sensiolabs, 2010).

- Controlador es el que se encarga de procesar la petición del usuario y realizar los cambios necesarios en el modelo o la vista, debido a que es el intermediador para la comunicación entre la vista y el modelo.

El controlador está dividido en varios componentes los cuales son:

- El controlador frontal es el que se encarga de aplicar la configuración y determinar las acciones que se pueden ejecutar.
- Verifica las peticiones del usuario y prepara los datos en la capa de presentación.
- Utiliza objetos como request que permite el acceso a los parámetros de la petición, response es encargada de las respuestas y sesión encargado de los usuarios.

Las características de Symfony son las siguientes:

- Este framework es fácil de instalar y configurar en varias plataformas.
- Es independiente del gestor de base de datos (Potencier & Zaninotto, s.f.).
- Utiliza las mejores prácticas y patrones de diseño de aplicaciones web.
- Es flexible porque se puede adaptar a cambios complejos en la aplicación.
- Su código es fácil de entender y admite un mantenimiento sencillo.

2.4. Arquitectura de aplicación

La arquitectura de aplicaciones es importante para la realización de este proyecto porque indica la estructura de la aplicación es decir la manera como está diseñada la aplicación tanto física como lógica.

2.4.1. Tipos de arquitectura

2.4.1.1. Arquitectura Cliente servidor

Arquitectura cliente – servidor es una arquitectura distribuida⁴ en la cual los usuarios finales tienen acceso a la información en forma transparente. Su

⁴ Un sistema distribuido: es un conjunto de ordenadores autónomos, acoplado en una red y soportados por un software que permite que actúe como un servicio integrado (Navarro & Márques, 2006).

funcionamiento es que el cliente envía un mensaje requiriendo un servicio y recibe mediante un mensaje la respuesta (Márquez & Zulaica, 2004).

En la Ilustración 7 se establece la estructura de la arquitectura Cliente servidor:

Cliente: Es el que se encarga de enviar una solicitud al servidor y posteriormente se queda en espera hasta recibir una respuesta. Su tiempo de vida es finito es decir hasta que todas sus peticiones se encuentren completas (EcuRed).

Servidor: Es un programa que se puede obtener en la red ya que ofrece un servicio. El funcionamiento es que acepta la solicitud del cliente, la realiza y envía la respuesta.

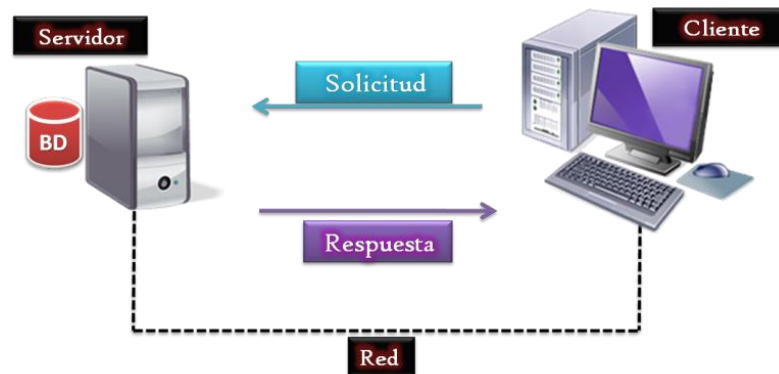


Ilustración 9 - Arquitectura Cliente Servidor

(Gama, 2010)

Ventajas de la arquitectura Cliente Servidor:

- Como el servidor se encuentra en el centro de red, administra los recursos comunes al usuario, por lo que la arquitectura Cliente-Servidor tiene recursos centralizados (CCM, 2016).
- En la arquitectura Cliente-servidor, cada vez posee plataformas de hardware más baratas.
- Posee una red escalable, que no afecta al funcionamiento de la red y no se necesita realizar muchas modificaciones al momento que se disminuye o aumenta el número de clientes.

- En esta arquitectura tiene una facilidad para la integración de nuevas tecnologías y el aumento de la infraestructura computacional (EcuRed).

Desventajas de la arquitectura Cliente Servidor:

- El mantenimiento para esta arquitectura es más complicada porque envuelve la interacción de las partes del hardware y software, distribuidas por distintos proveedores (EcuRed).
- No posee una variedad de herramientas para la administración y ajuste del desempeño de la arquitectura (EcuRed).
- Puede tener problemas de desempeño por la congestión de la red.

2.4.1.2. *Arquitectura de tres capas*

Como se puede observar en la Ilustración 8 la arquitectura consta de 3 capas las cuales son:

1. La capa de presentación es el browser o la interfaz gráfica que le presenta el sistema al usuario al momento de abrir la aplicación, por lo que debe poseer las características principales de ser amigable, entendible y fácil de usar, esta capa también permite comunicar y capturar la información del usuario. Solo se puede comunicar con la capa del negocio.
2. La capa lógica del negocio es la capa donde se encuentran los programas que están ejecutando, recibe las peticiones del usuario y envía la respuesta tras el proceso. En esta capa se establecen todas las normas o reglas que deben cumplirse en esta arquitectura.
3. Capa de datos es una base de datos donde se guarda toda la información, para después acceder a la misma. Por lo que su característica principal es almacenar y devolver datos a la capa de la lógica del negocio (Ecured, s.f.), ya que solo se puede comunicar con esta capa.



Ilustración 10 - Estructura de la arquitectura de tres capas

(NewComLab, s.f.)

Ventajas de la arquitectura de tres capas:

- Se puede separar en varios ordenadores, si se aumenta la complejidad y el tamaño de la base de datos o de la parte lógica.
- Si se cambia algún dato de la base de datos no afecta a la capa de la interface.
- Que exista tres capas, facilita al momento de reemplazar o modificar una capa sin afectar a los demás módulos (Universidad Simón Bolívar, 2011).

Desventajas de la arquitectura de tres capas:

- Puede bajar el desempeño cuando los clientes envía varias solicitudes a un solo servidor, por lo que generan tráfico en la red.
- Se necesita más balanceo de carga y tolerancia a fallos (Universidad Simón Bolívar, 2011).

2.4.2. Selección de arquitectura

Para la realización de aplicación de la presente disertación se aplicará la arquitectura de tres capas porque está organizado de mejor manera, por lo que es más fácil de utilizar, ya que se tiene la capa de presentación, la lógica y la de datos, y si se realiza alguna modificación en alguna capa no afecta a las demás.

2.5. Patrón de diseño

“Un patrón de diseño es una descripción de clase y objetos comunicándose entre sí adaptada para resolver un problema de diseño general en un contexto particular” (Goñi's, s.f.).

Elementos de los patrones:

- De creación: Es la que se encarga de la creación de objetos y separar de la implementación de los objetos con el cliente (Rubenfa, 2014).
- De estructura: Aquí se encuentra las características y como están compuestas las clases y objetos.
- De comportamiento: Se utiliza para conocer como las clases y objetos interactúan y reparten responsabilidades.

2.5.1. Modelo Vista Controlador (MVC)

El Modelo Vista Controlador es un patrón que se encarga de dividir la aplicación en tres elementos que son modelos, vista y controlador.

En la Ilustración 9 se estable los elementos del Modelo Vista Controlador, los cuales son:

- Modelos: En esta capa es donde se ubican los datos por lo que permite acceder y modificar dicha información. Estos datos están almacenados en una base de datos.
- Vista: Es la interface que es visible al usuario, con el objetivo de mostrar datos del objeto.
- Controlador: Es el que se encuentra en el medio por lo que actúa como mediador entre las capas del modelo y vista. Es la que responde las acciones que se solicitan en la aplicación por ejemplo la visualización de un elemento (Alvarez, Qué es MVC, 2014).

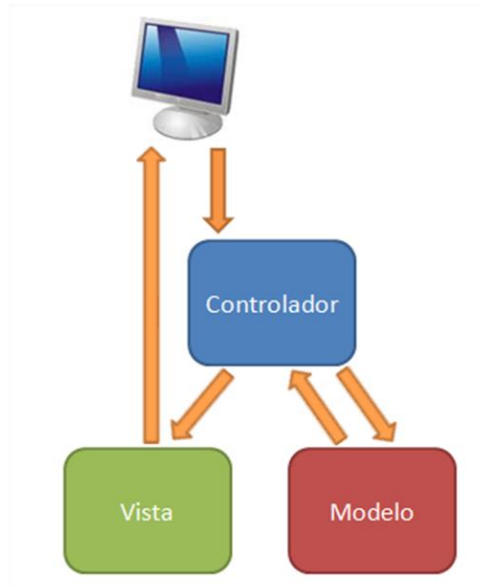


Ilustración 11 - MVC

(.net y punto, 2011)

2.6. Resumen de las Herramientas Seleccionadas

Herramienta	Tipo	Características	Razón de la elección
RUP	Metodología	<ul style="list-style-type: none"> • Es Orientada a objetos. • Está basada en casos de uso. • Centrado en la arquitectura (partes más importantes). • Es iterativo o incremental. 	Porque es una metodología orientada a objetos y se basa en un modelo iterativo que permite ir mejorando en cada una de las versiones y así obtener un producto de calidad.
UML	Herramienta de modelado	<ul style="list-style-type: none"> • Es un lenguaje para visualizar. • Es un lenguaje que especifica las características del sistema antes de desarrollar. 	Porque está herramienta de modelado es orientada a objetos y permite representar de manera clara, visual y específica el sistema,

Herramienta	Tipo	Características	Razón de la elección
		<ul style="list-style-type: none"> • Es un lenguaje para construir. • Es un lenguaje para documentar. 	para que la construcción de la aplicación sea más sencilla.
PHP	Herramienta tecnológica	<ul style="list-style-type: none"> • Es un lenguaje eficaz. • Integrado a las bases de datos. • Es fácil de usar. • No tiene costo. • Compatibilidad con un enfoque orientado a objetos. • Portabilidad. 	Porque PHP puede tener un enfoque orientado a objetos que se adapta a la metodología RUP. También porque PHP es el lenguaje de programación más utilizado para las aplicaciones web ya que tienen una gran cantidad de ventajas.
CSS	Herramienta tecnológica	<ul style="list-style-type: none"> • Lenguaje de hojas de estilo. • Describe la interface del documento HTML. • Controla el diseño de varias páginas al mismo tiempo. 	Porque CSS es un lenguaje de hojas de estilo que permite definir el diseño, posiciones y tamaños de los diferentes componentes de la aplicación.
JavaScript	Herramienta Tecnológica	<ul style="list-style-type: none"> • Se ejecuta sin necesidad de otro programa para su visualización. • Trabaja con objetos. • Permite codificar de manera más rápida y fácil. • Trabaja con eventos. 	Porque es un lenguaje orientado objetos, que permite de manera más fácil codificar las funciones de la aplicación mediante la

Herramienta	Tipo	Características	Razón de la elección
			utilización de funciones y eventos.
PostgresSQL	Herramienta de base de datos	<ul style="list-style-type: none"> • La licencia es libre. • Es multiplataforma. • Es escalable • Es una de la base de datos gratuitas más potentes. • Es más sencillo analizar los datos. 	Porque es una base de datos gratuita muy potente que soporta varias funcionalidades que ayudaran a la elaboración de la aplicación. También consta con una herramienta gráfica.
Symfony	Framework	<ul style="list-style-type: none"> • Es fácil de instalar y configurar. • Es independiente del gestor de BDD • Utiliza el patrón de diseño MVC. • Es fácil de entender. 	Porque este Framework trabaja con el patrón de diseño MVC que ayuda a codificar de mejor manera con las mejores prácticas de programación. También ayuda con funcionalidades simples como el CRUD, y así el desarrollador puede enfocarse principalmente en las complejas.
Tres Capas	Arquitectura de la aplicación	<ul style="list-style-type: none"> • Tiene la capa de presentación, negocio y datos. • Se puede separar cada capa en varios ordenadores. 	Porque esta arquitectura permite trabajar de mejor manera al separar en tres capas distintas.

Herramienta	Tipo	Características	Razón de la elección
		<ul style="list-style-type: none">• Es sencillo modificar cada capa ya que no afecta a la otra.	
MVC	Patrón de diseño	<ul style="list-style-type: none">• Divide la aplicación en tres elementos modelo, vista y controlador.	Porque este patrón de diseño se adapta de mejor manera al Framework a utilizar. Además permite dividir a la aplicación y así codificar de forma sencilla y entendible.

Tabla 6 - Resumen de herramientas

Elaborado por: Nathaly Herrera

CAPITULO III: CASO DE ESTUDIO

Para la realización del proyecto se aplica la metodología RUP, por lo cual, para su realización se va generar dos iteraciones donde todo lo elaborado en la primera se puede mejorar y aumentar en la segunda y así obtener un producto final de calidad. En cada iteración se cumple varias tareas que se describirán posteriormente en este documento.

3.1. Primera Iteración

En este primer ciclo se detalla las características principales de la empresa sobre lo que se está realizando el proyecto y por qué necesita este sistema y cómo le va a beneficiar al centro odontológico.

Además, en este ciclo para la parte de diseño y funcionalidad del programa se realiza 5 requerimientos funcionales de los 10 establecidos. La presentación o realización del sistema es con el lenguaje de programación PHP, el Framework Symfony y con la base de datos en Postgresql.

3.1.1. Fase de Iniciación

3.1.1.1. Modelo de Negocio

Actividades del Negocio

El Centro Implantológico y de Especialidades Dental Access Ofrecen una alta variedad de Especialidades Odontológicas como son:

- Implantología Oral: Reemplazo unitario, múltiple o total de piezas dentales.
- Periodoncia: Prevención y tratamientos de patologías que afectan a los tejidos que rodean al diente.
- Endodoncia: Tratar o prevenir procesos inflamatorios infecciosos.
- Cirugía oral: Diagnostico, prevención y tratamientos de patologías bucales.
- Ortodoncia-ortopedia: Prevención y tratamientos de anomalías en la posición de las piezas dentales.
- Rehabilitación Oral: Mediante diferentes tratamientos devuelven la función, estética y armonía dental, utilizando prótesis.

Descripción de la situación actual de la empresa

El Centro Dental Access ha estado manejando las citas de forma manual, es decir en un cuaderno guarda toda la información. El paciente se acerca a Dental Access para solicitar cita, el personal registra manualmente los datos del cliente y posteriormente agenda las citas al paciente de acuerdo a la especialidad solicitada, para anotar la cita revisan que días tienen disponibles el doctor que se solicita y le registran en un cuaderno con toda la información del paciente, al paciente le entregan anotado en una hoja de papel el horario de todas sus citas. Esta situación ha significado pérdida de tiempo tanto del personal del Centro como del paciente.

Igualmente, en el caso del expediente del usuario tienen registrado en hojas por lo que al momento de realizar una búsqueda de cierta información conlleva gran cantidad de tiempo y si necesitan ejecutar algún reporte no lo pueden efectuar con facilidad.

3.1.1.2. Requerimiento

Descripción del sistema

El sistema es una aplicación web en el cual permite a los administradores del centro odontológico llevar un historial médico de cada paciente, para facilitar la gestión del tratamiento asignado. De acuerdo al tratamiento o problema que se le diagnostica al paciente, se podrá determinar el especialista y así reservar las citas.

El sistema gestiona las citas para que el paciente pueda reservar desde cualquier lugar mediante el ingreso a la página de reservas de citas y se envía de forma automática la confirmación o cancelación de la cita al e-mail del cliente, además el personal del Centro Odontológico también se encarga de reservar cita y de cancelar la misma de acuerdo a la petición del paciente.

Como el sistema guarda toda la información del paciente y su tratamiento se puede realizar un reporte del historial médico. Por lo tanto, esta aplicación permite ofrecer a todos los clientes un servicio de calidad.

Para ingresar al sistema el doctor y el empleado se le asigna un usuario y contraseña, el cual es establecido por el administrador, mientras que el paciente ingresa exclusivamente a la página de reserva de citas.

Requerimientos funcionales

F1: El sistema permitirá el ingreso al sistema

F2: El sistema gestionará empleados

F3: El sistema gestionará pacientes

F4: El sistema gestionará doctores

F5: El sistema gestionará ficha de citas

F6: El sistema gestionará ficha médica

F7: El sistema gestionará horario de atención

F8: El sistema permitirá reserva de citas

F9: El sistema permitirá cancelación de citas

F10: El sistema gestionará usuarios para el ingreso al sistema

Los Requerimientos que se realiza en la primera iteración son 5 los cuales son: Paciente, Doctor, Empleado, Ficha Médica y Horario de Atención.

De los 5 requerimientos se especifica los 3 más representativos para el proyecto los cuales son: Paciente, ficha Médica y Horario de Atención, y los otros 2 requerimientos se encuentra en el CD.

Requerimientos no funcionales

NF1: El sistema necesita como mínimo 2GB de memoria RAM.

NF2: El sistema será eficiente es decir que tendrá un tiempo de respuesta de 5seg.

NF3: El sistema es multiplataforma (PHP).

NF4: El sistema funcionará con una base de datos postgresql.

NF5: El sistema será sencillo para el uso del usuario.

Diseño Conceptual

El diseño conceptual es un diagrama que permite traducir al desarrollador los requerimientos del cliente a forma gráfica para que sea un lenguaje común entre el usuario y el desarrollador.

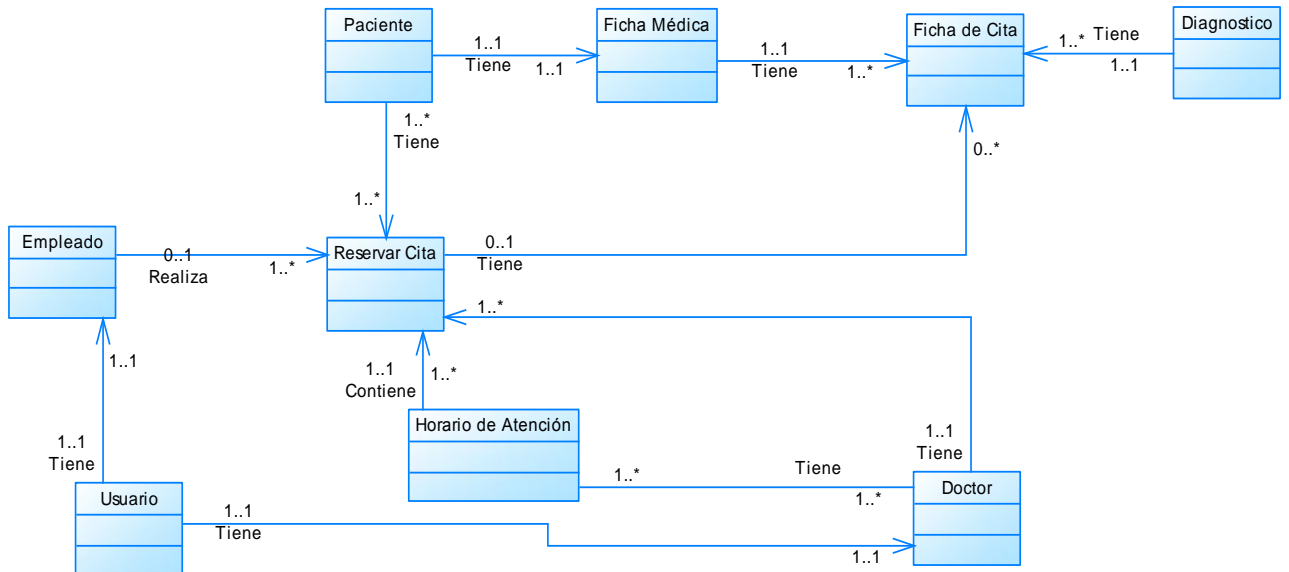


Diagrama 1 - Modelo Conceptual

Elaborado por: Nathaly Herrera

Diagrama de casos de uso

Los diagramas de caso de uso son elementos de modelado a través de los cuales representa la forma o los pasos del funcionamiento del sistema.

Diagrama General

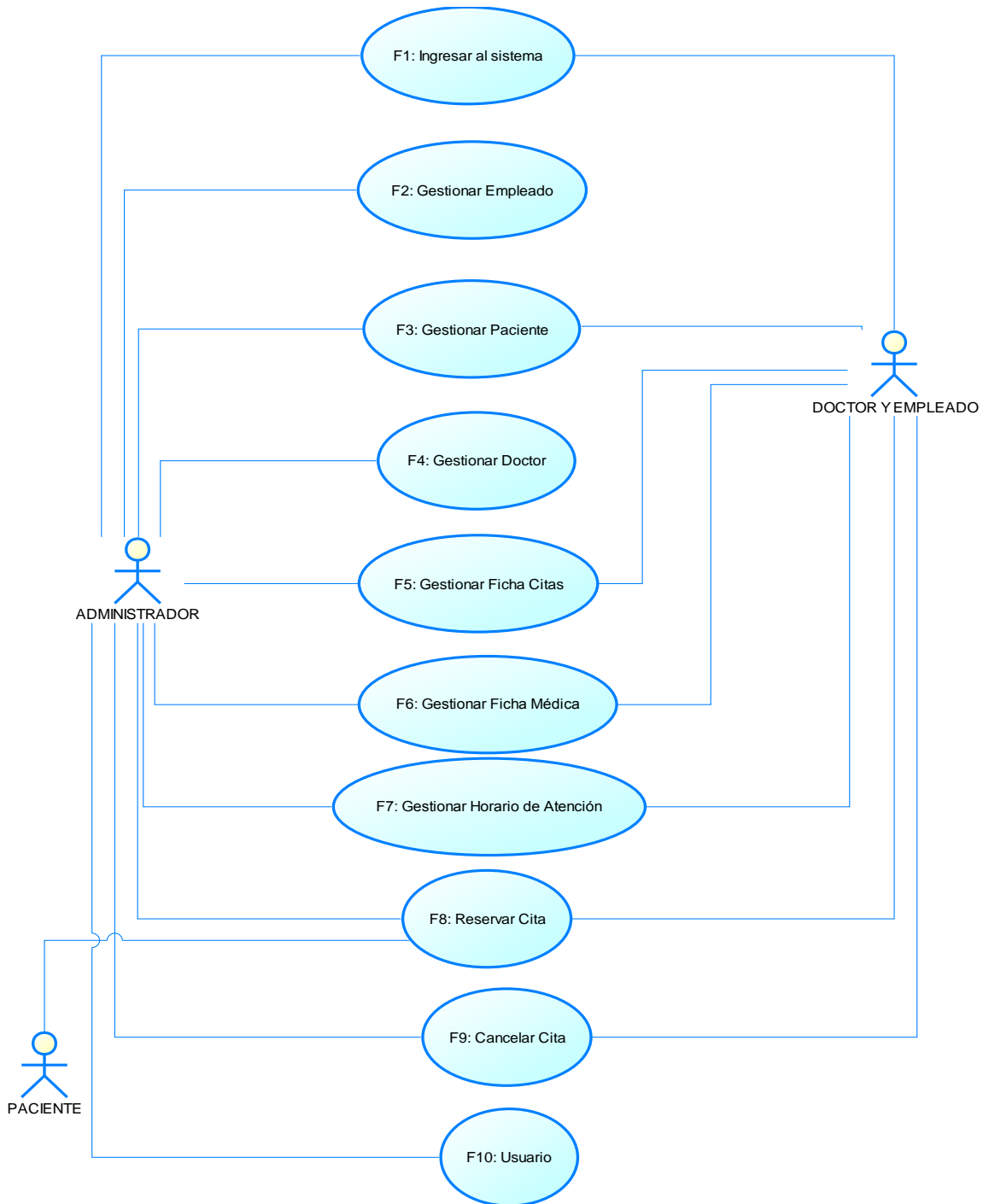


Diagrama 2 - Diagrama General

Elaborado por: Nathaly Herrera

Diagrama caso de uso Gestionar Paciente

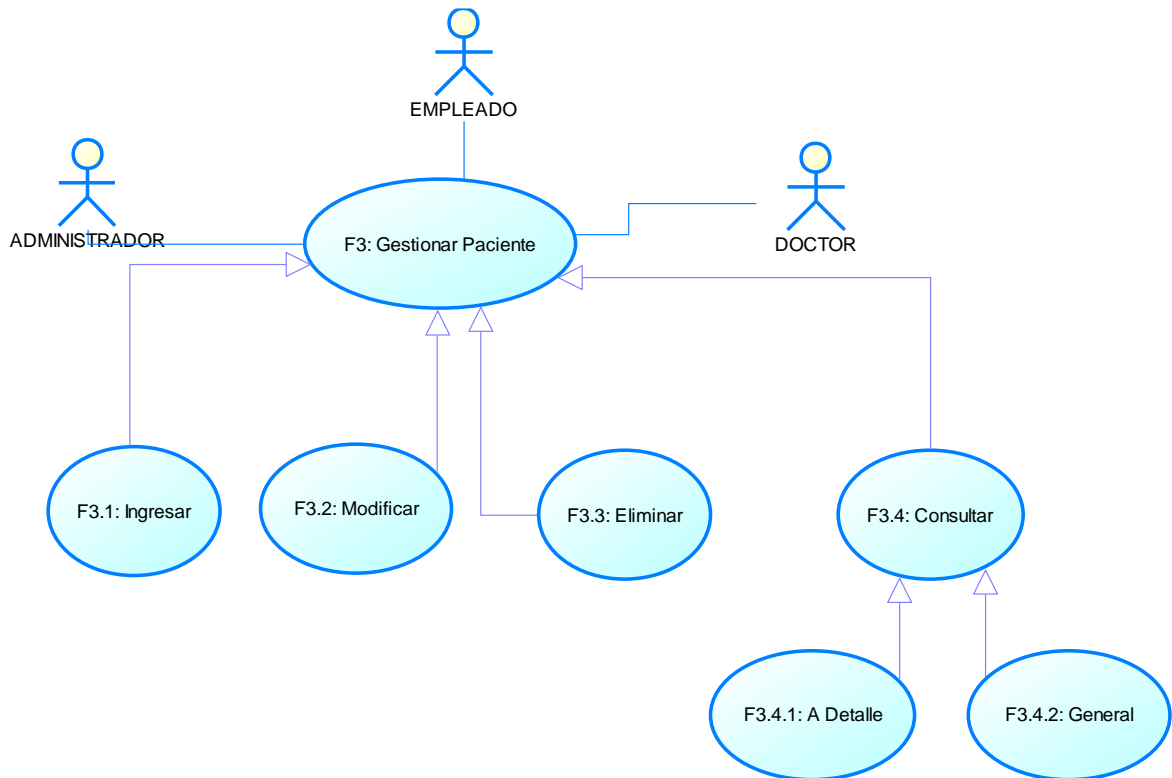


Diagrama 3 - Caso de Uso Gestionar Paciente

Elaborado Por: Nathaly Herrera

Detalle del caso de uso de Gestionar Paciente:

F3.1: Ingresar Paciente

Descripción: El sistema permitirá al actor ingresar los datos de cada paciente del Centro Odontológico y se generará automáticamente su respectivo código el cuál será único.

Actores: Administrador, Empleado y Doctor

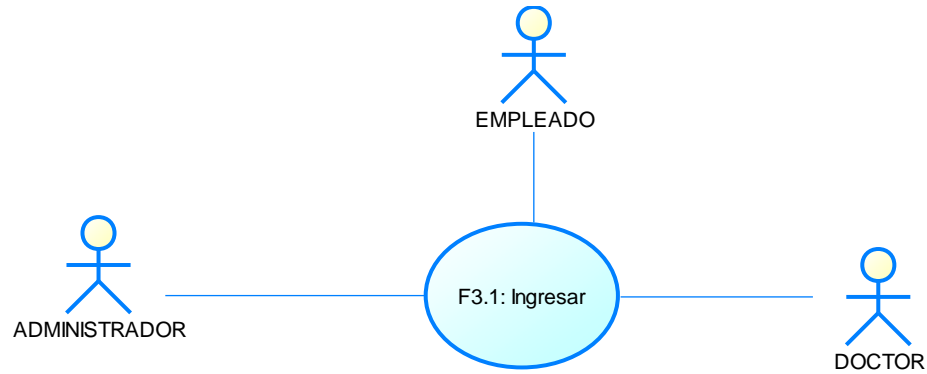


Diagrama 4 - Caso de Uso Ingresar Paciente

Elaborado Por: Nathaly Herrera

Flujo Principal:

1. El actor selecciona Administrar Paciente en el Menú.
2. El sistema muestra la ventana de administrar paciente.
3. El actor presiona el botón ingresar nuevo registro.
4. El sistema muestra la ventana de ingresar paciente.
5. El actor ingresa la cédula del paciente.
6. El sistema verifica la cédula. (E1)
7. El actor ingresa los demás datos en los campos.
8. El actor presiona el botón guardar.
9. El sistema verifica reglas de edición (E2).
10. El sistema ingresa los datos del paciente. (E1)

Flujo alternativo

- 6) Si la cédula del paciente existe, ir al Caso de uso F3.2 (Modificar) o F3.3 (Eliminar).

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

E2: No se pueden ingresar los datos, la solución volver a ingresar el dato.

F3.2: Modificar Paciente

Descripción: El sistema permitirá al actor modificar los datos de cada paciente del Centro Odontológico especificando su cédula del paciente.

Actor: Administrador, Empleado y Doctor

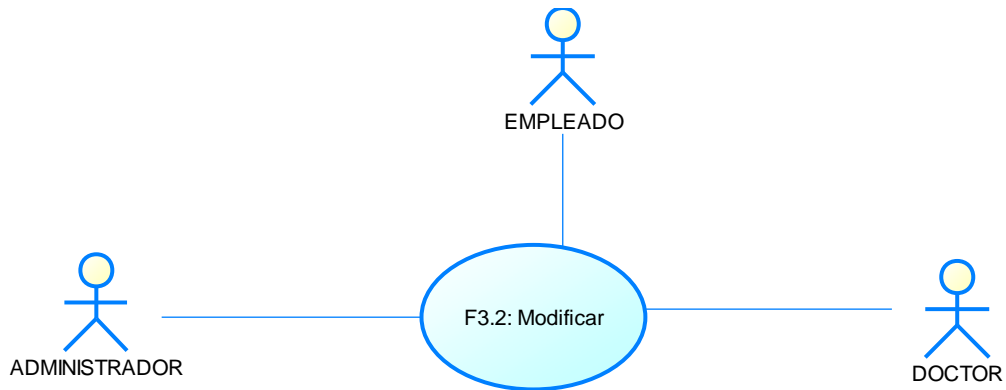


Diagrama 5 - Caso de Uso Modificar Paciente

Elaborado por: Nathaly Herrera

Flujo Principal:

1. El actor selecciona administrar paciente en el menú.
2. El sistema muestra la ventana de administrar paciente.
3. El actor ingresa la cédula del paciente.
4. El actor presiona el botón buscar.
5. El sistema verifica la cédula. (E1)
6. El sistema muestra los datos del paciente.
7. El actor presiona el botón editar
8. El sistema muestra la ventana de modificar
9. El actor modifica los datos del paciente.
10. El actor presiona el botón modificar.
11. El sistema verifica reglas de edición. (E2)
12. El sistema modifica los datos. (E1)

Flujo alterno

5) Si la cédula del paciente no existe, ir al Caso de uso F3.1 (ingresar).

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

E2: No se puedo modificar los datos, la solución volver a ingresar el dato.

F3.3: Eliminar Paciente

Descripción: El sistema permitirá al actor eliminar los datos de cada paciente del Centro Odontológico dependiendo de su cédula.

Actor: Administrador, Empleado y Doctor

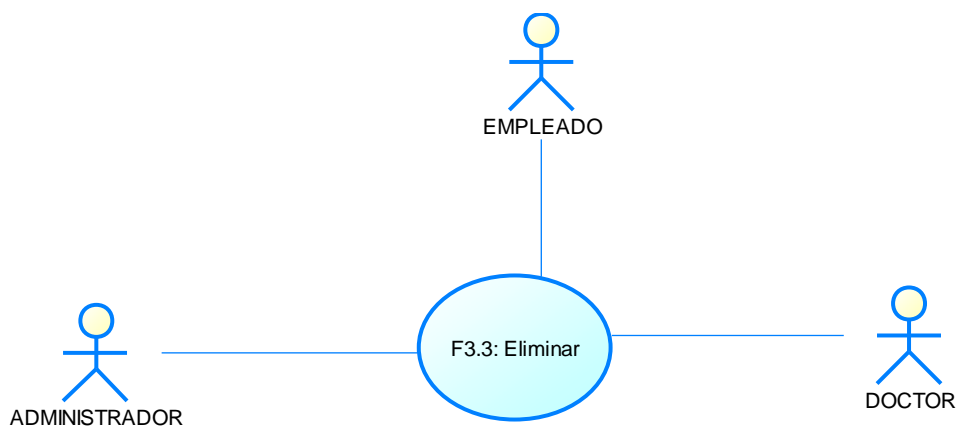


Diagrama 6 - Caso de Uso Eliminar Paciente

Elaborado por: Nathaly Herrera

Flujo Principal

1. El actor selecciona administrar paciente en el menú.
2. El sistema muestra la ventana de administrar paciente.
3. El actor ingresa la cédula.

4. El actor presiona el botón buscar.
5. El sistema verifica la cédula. (E1)
6. El sistema muestra los datos del paciente.
7. El actor presiona el botón eliminar.
8. El sistema elimina los datos. (E1)

Flujo alterno

5) Si la cédula del paciente no existe, ir al Caso de uso F3.1 (ingresar)

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

F3.4.1: Consulta a Detalle del Paciente

Descripción: El sistema permitirá al actor consultar los datos de cada paciente del Centro Odontológico especificando su cédula.

Actor: Administrador, Empleado y Doctor

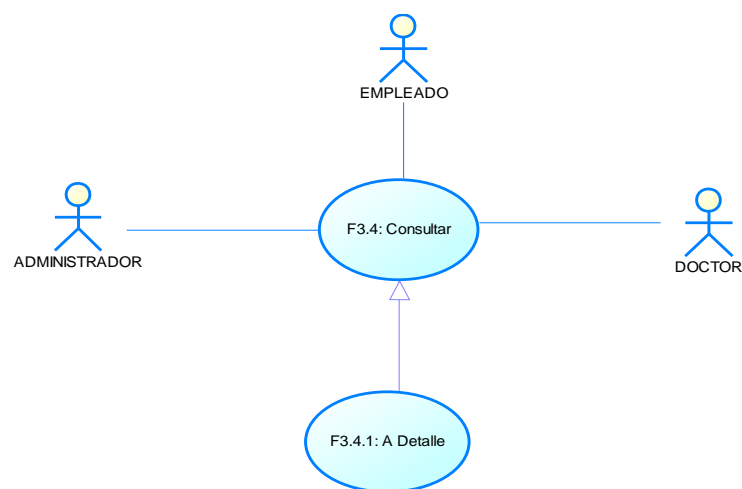


Diagrama 7 - Caso de Uso Consulta a Detalle Paciente

Elaborado por: Nathaly Herrera

Flujo Principal

1. El actor selecciona administrar paciente en el menú.
2. El sistema muestra la ventana de administrar paciente.
3. El actor ingresa la cédula.
4. El actor presiona el botón buscar.
5. El sistema verifica la cédula. (E1)
6. El sistema muestra los datos del paciente especificado.

Flujo alterno

- 5) Si la cédula del paciente no existe, ir al Caso de uso F3.1 (ingresar)

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

F3.4.2: Consulta General del Paciente

Descripción: El sistema permitirá al actor consultar los datos de todos los pacientes del Centro Odontológico.

Actor: Administrador, Empleado y Doctor

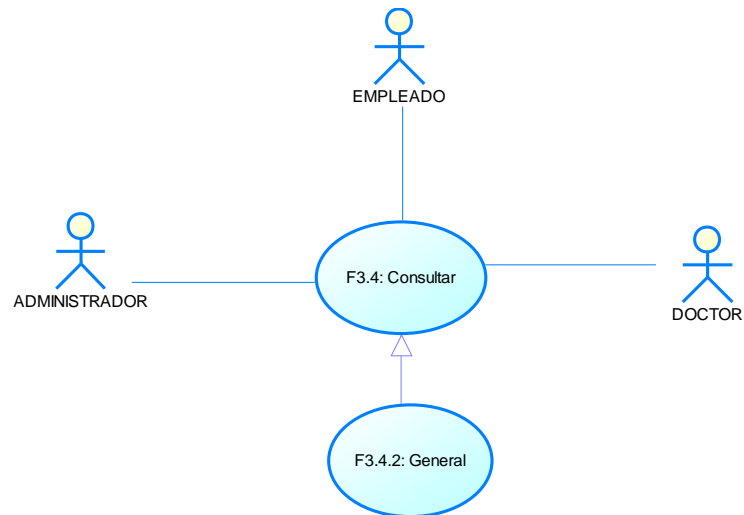


Diagrama 8 - Caso de Uso Consulta General Paciente

Elaborado por: Nathaly Herrera

Flujo Principal

1. El actor selecciona administrar paciente en el menú.
2. El sistema muestra la ventana de administrar paciente.
3. El sistema muestra los datos de todos los pacientes. (E1)

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

Diagrama caso de uso Gestionar Ficha Médica

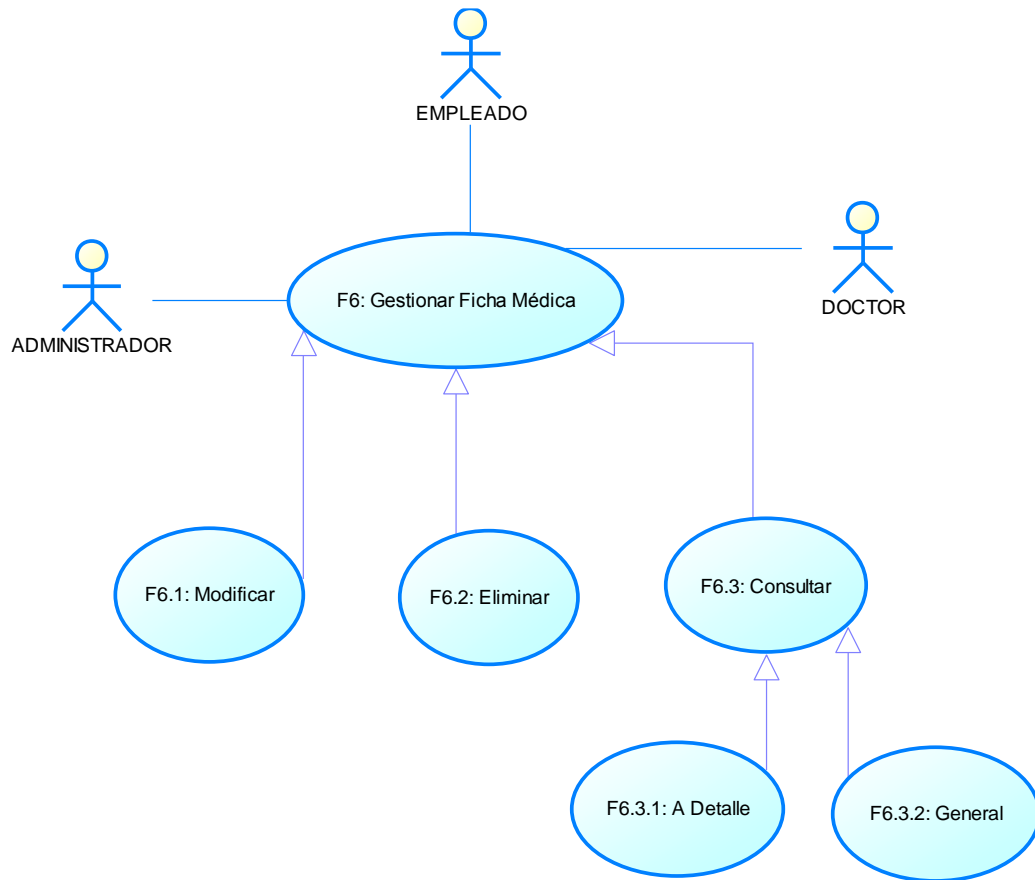


Diagrama 9 - Caso de Uso Gestionar Ficha Médica

Elaborado por: Nathaly Herrera

Detalle del caso de uso de Gestionar Ficha de Médica:

F6.1: Modificar Ficha Médica

Descripción: El sistema permitirá al actor modificar los datos de la ficha médica de un determinado paciente del Centro Odontológico especificando la cédula.

Actor: Administrador, Empleado y Doctor

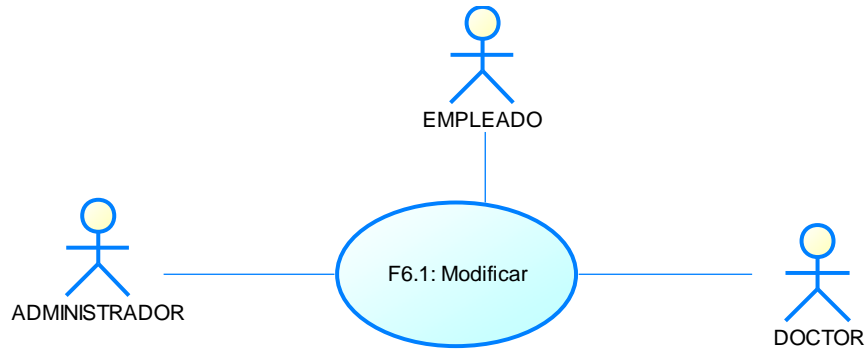


Diagrama 10 - Caso de Uso Modificar Ficha Médica

Elaborado por: Nathaly Herrera

Flujo Principal:

1. El actor selecciona administrar ficha médica en el menú.
2. El sistema muestra la ventana administrar ficha médica.
3. El actor ingresa la cédula del paciente.
4. El actor presiona el botón buscar.
5. El sistema verifica la cédula. (E1)
6. El sistema muestra los datos del paciente y su ficha médica.
7. El actor presiona el botón editar.
8. El sistema muestra la ventana de modificar.
9. El actor modifica los datos.
10. El actor presiona el botón modificar.
11. El sistema verifica las reglas de edición. (E2)
12. El sistema modifica los datos. (E1)

Flujo alterno

- 5) Si la cédula del paciente no existe, ir al Caso de uso F3.1 (ingresar) y si el paciente no tiene ficha médica, ir al caso de uso F6.1 (Insertar).

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

E2: No se puede modificar, la solución es volver a ingresar los datos.

F6.3: Eliminar Ficha Médica

Descripción: El sistema permitirá al actor eliminar la ficha médica de un determinado paciente del Centro Odontológico especificando su cédula.

Actor: Administrador, Empleado y Doctor

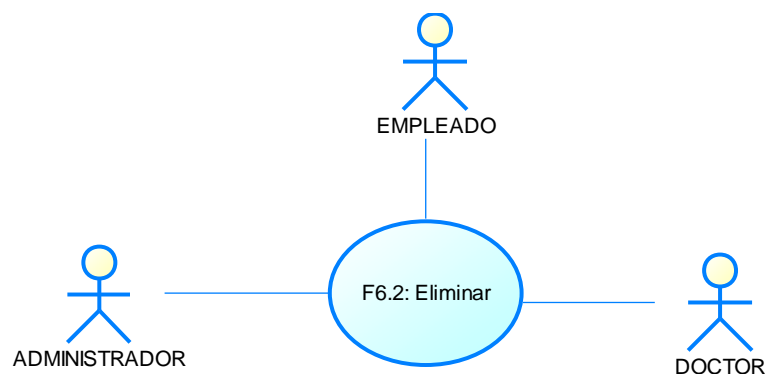


Diagrama 11 - Caso de uso Eliminar Ficha Médica

Elaborado por: Nathaly Herrera

Flujo Principal

1. El actor selecciona administrar ficha médica en el menú.
2. El sistema muestra la ventana de administrar ficha médica.
3. El actor ingresa la cédula del paciente.
4. El actor presiona el botón buscar.
5. El sistema verifica la cédula. (E1)
6. El sistema muestra los datos del paciente con su ficha médica.

7. El actor presiona el botón eliminar.
8. El sistema elimina los datos. (E1)

Flujo alterno

5) Si la cédula del paciente no existe, ir al Caso de uso F3.1 (ingresar) y si la ficha médica del determinado paciente no existe, ir al caso de uso F6.1 (ingresar).

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

F6.4.1: Consulta a Detalle de la Ficha Médica

Descripción: El sistema permitirá al actor consultar los datos de la ficha médica especificando la cedula del paciente del Centro Odontológico.

Actor: Administrador, Empleado y Doctor

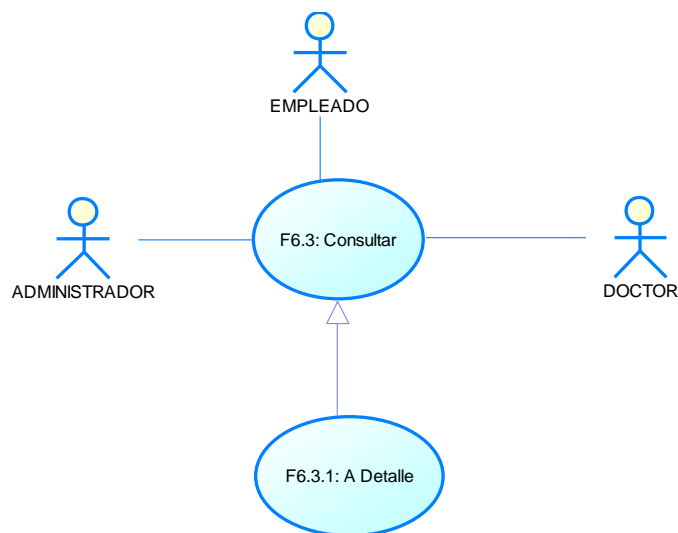


Diagrama 12 - Caso de Uso Consulta a Detalle Ficha Médica

Elaborado por: Nathaly Herrera

Flujo Principal

1. El actor selecciona Administrar Ficha Médica en el menú.
2. El sistema muestra la ventana de administrar ficha médica.
3. El actor ingresa la cédula del paciente.
4. El actor presiona el botón buscar.
5. El sistema verifica la cédula. (E1)
6. El sistema muestra la ficha médica.

Flujo alterno

5) Si la cédula paciente no existe, ir al Caso de uso F3.1 (ingresar) y si el paciente no tiene una ficha médica, ir al caso de uso F6.1 (ingresar).

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

F6.4.2: Consulta General de la Ficha Médica

Descripción: El sistema permitirá al actor consultar todos los pacientes con su respectiva ficha médica en el Centro Odontológico.

Actor: Administrador, Empleado y Doctor

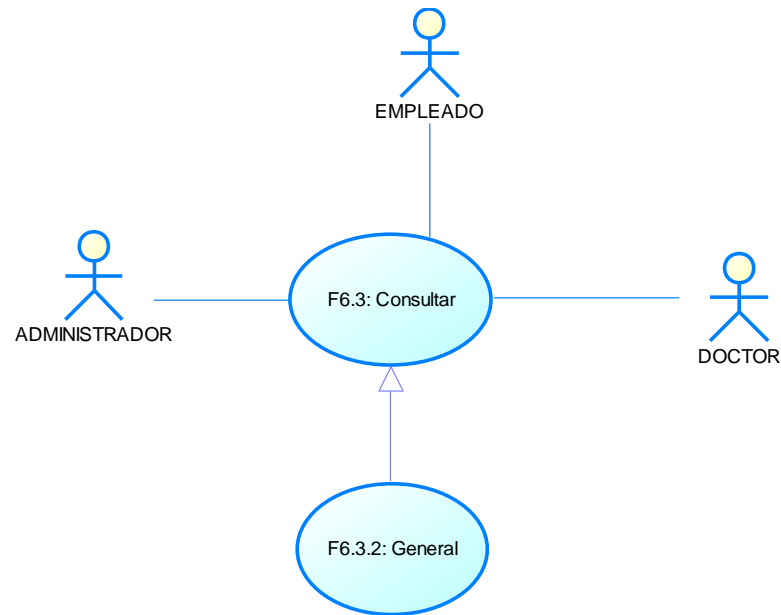


Diagrama 13 - Caso de Uso Consulta General Ficha Médica

Elaborado por: Nathaly Herrera

Flujo Principal

1. El actor selecciona administrar ficha médica en el menú.
2. El sistema muestra la ventana de administrar ficha médica.
3. El sistema muestra todos los pacientes con su ficha médica. (E1)

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

Diagrama caso de uso Gestionar Horario de Atención

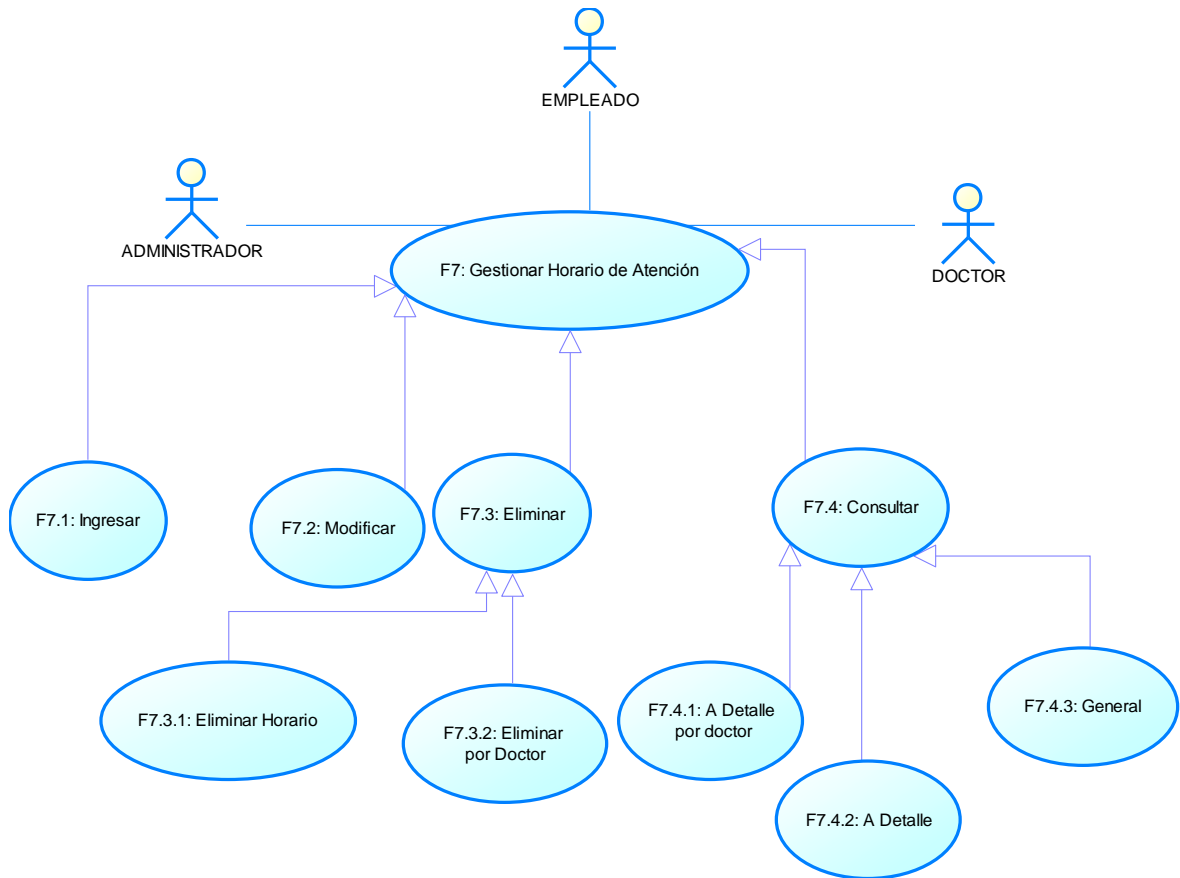


Diagrama 14 - Caso de Uso Horario de Atención

Elaborado Por: Nathaly Herrera

Detalle del caso de uso de Gestionar Horario de Atención:

F7.1: Ingresar Horario de Atención

Descripción: El sistema permitirá al actor ingresar los datos del horario de atención de cada doctor del Centro Odontológico.

Actor: Administrador, Empleado y Doctor

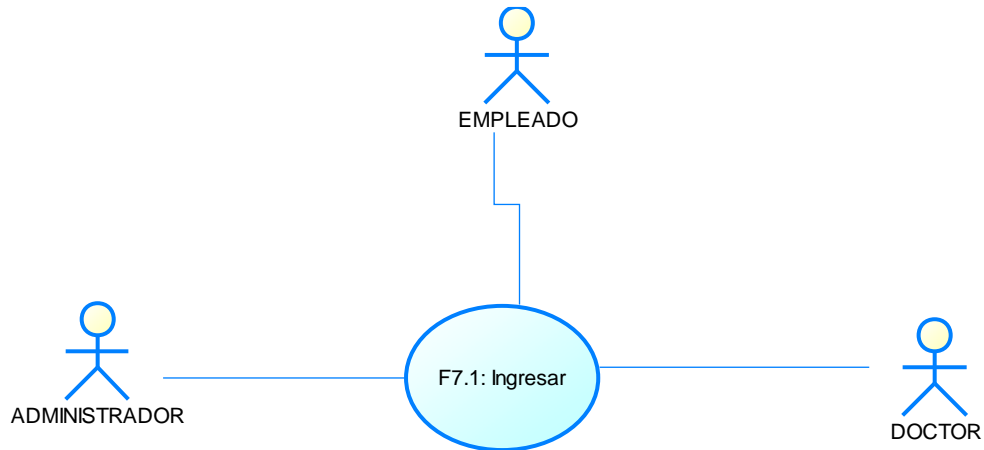


Diagrama 15 - Caso de Uso Insertar Horario de Atención

Elaborado Por: Nathaly Herrera

Flujo Principal:

1. El actor selecciona Administrar Horario de atención en el Menú.
2. El sistema muestra la ventana de administrar horario de atención.
3. El actor presiona el botón nuevo horario
4. El sistema muestra la ventana de ingreso
5. El actor ingresa o escoge los campos de horario de atención.
6. El actor presiona el botón guardar.
7. El sistema verifica las reglas de edición. (E2)
8. El sistema ingresa los datos. (E1)

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

E2: No se puede ingresar los datos, la solución es volver a ingresar los datos.

F7.2: Modificar Horario de Atención

Descripción: El sistema permitirá al actor modificar los datos del horario de atención especificando su código, fecha o hora del Centro Odontológico.

Actor: Administrador, Empleado y Doctor

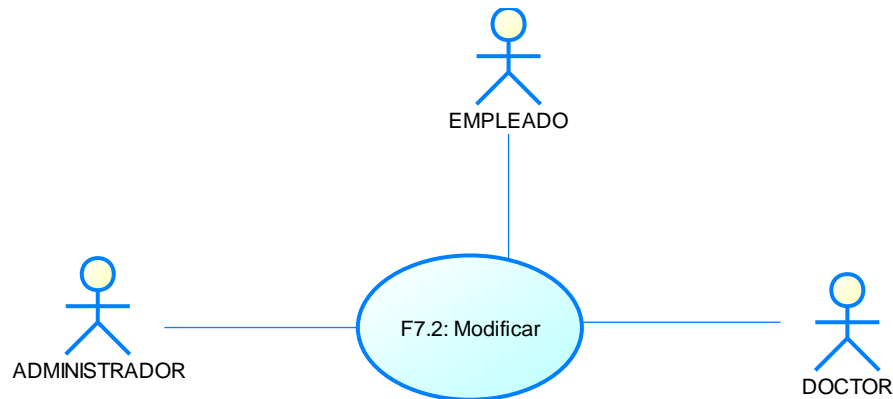


Diagrama 16 - Caso de Uso Modificar Horario de Atención

Elaborado por: Nathaly Herrera

Flujo Principal:

1. El actor selecciona administrar horario de atención en el menú.
2. El sistema muestra la ventana administrar horario de atención.
3. El actor ingresa el código, hora o la fecha del horario.
4. El sistema filtra en la tabla el horario de atención. (E1)
5. El actor presiona el botón editar.
6. El sistema muestra la ventana de modificar.
7. El actor modifica los datos.
8. El actor presiona el botón modificar.
9. El sistema verifica las reglas de edición. (E2)
10. El sistema modifica los datos. (E1)

Flujo alterno

- 4) Si el horario especificado no existe, ir al Caso de uso F7.1 (ingresar).

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

E2: No se puede modificar, la solución es volver a ingresar los datos.

F7.3: Eliminar

F7.3.1: Eliminar Horario de Atención

Descripción: El sistema permitirá al actor eliminar el horario de atención especificando su código, hora o fecha del Centro Odontológico.

Actor: Administrador, empleado y Doctor

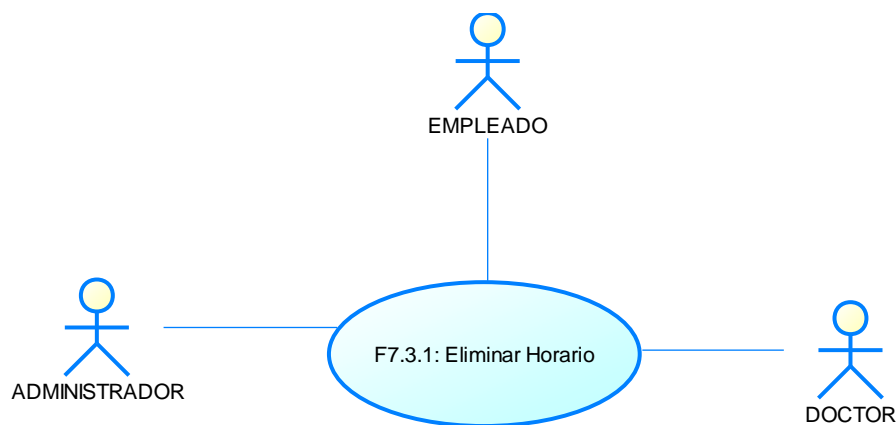


Diagrama 17 - Caso de Uso Eliminar Horario de Atención

Elaborado por: Nathaly Herrera

Flujo Principal

1. El actor selecciona administrar horario de atención en el menú.
2. El sistema muestra la ventana de administrar horario de atención.
3. El actor ingresa el código, hora o la fecha del horario.

4. El sistema filtra la tabla con los datos especificados. (E1)
5. El actor escoge el horario
6. El sistema muestra datos del horario.
7. El actor presiona el botón eliminar.
8. El sistema elimina los datos. (E1)

Flujo alterno

- 3) Si el horario de atención especificado no existe, ir al caso de uso F7.1 (ingresar).

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

F7.3.2: Eliminar Horario de Atención por Doctor

Descripción: El sistema permitirá al actor eliminar el horario de atención de un determinado doctor del Centro Odontológico especificando su cédula.

Actor: Administrador, empleado y Doctor

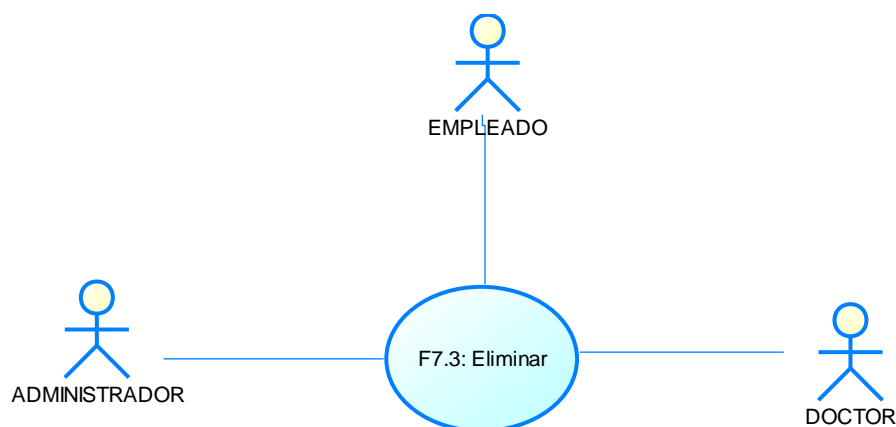


Diagrama 18 - Caso de Uso Eliminar Horario de Atención

Elaborado por: Nathaly Herrera

Flujo Principal

1. El actor selecciona administrar horario de atención en el menú.
2. El sistema muestra la ventana de administrar horario de atención.
3. El actor ingresa la cédula del doctor.
4. El actor presiona el botón buscar.
5. El sistema verifica la cédula. (E1)
6. El sistema muestra los datos del doctor y sus horarios.
7. El actor seleccionar el horario a eliminar.
8. El actor presiona el botón eliminar.
9. El sistema elimina los datos. (E1)

Flujo alterno

5) Si la cédula del doctor no existe, ir al Caso de uso F4.1 (ingresar) y si el horario de atención del doctor especificado no existe, ir al caso de uso F7.1 (ingresar).

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

F7.4: Consulta

F7.4.1: Consulta a Detalle del Horario de Atención por Doctor

Descripción: El sistema permitirá al actor consultar los datos del horario de atención especificando la cedula del doctor del Centro Odontológico.

Actor: Administrador, empleado y Doctor

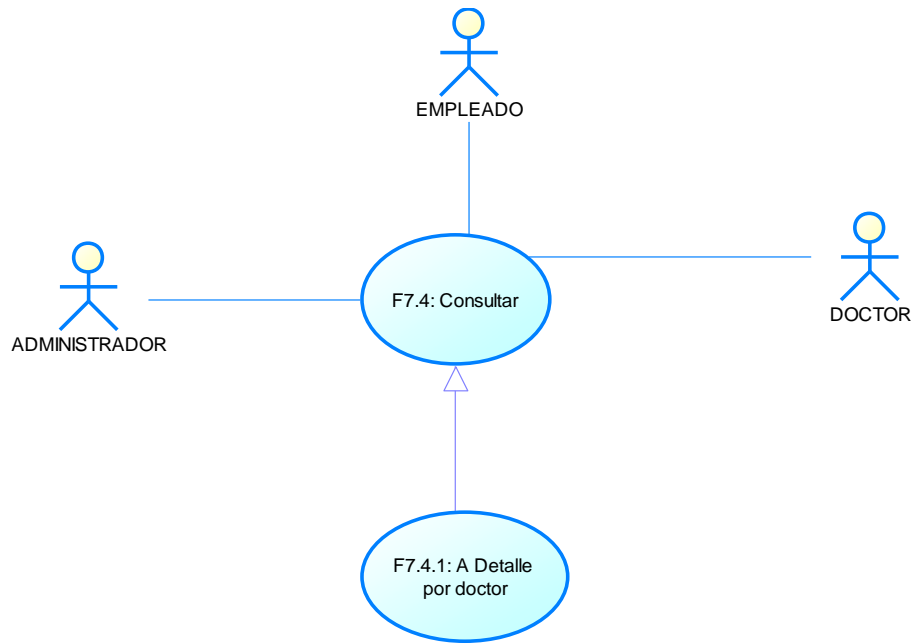


Diagrama 19 - Caso de Uso Consulta a Detalle por doctor el Horario de Atención

Elaborado por: Nathaly Herrera

Flujo Principal

1. El actor selecciona administrar horario de atención en el menú.
2. El sistema muestra la ventana de administrar horario de atención.
3. El actor ingresa la cédula del doctor.
4. El actor presiona el botón buscar.
5. El sistema verifica la cédula. (E1)
6. El sistema muestra todos los horarios de atención del determinado doctor.

Flujo alterno

- 5) Si la cédula del doctor no existe, ir al Caso de uso F4.1 (ingresar) y el horario de atención del doctor especificado no existe, ir al caso de uso F7.1 (ingresar).

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

F7.4.2: Consulta a Detalle el Horario de Atención

Descripción: El sistema permitirá al actor consultar los datos del horario de atención especificando su código, fecha o hora del Centro Odontológico.

Actor: Administrador, empleado y Doctor

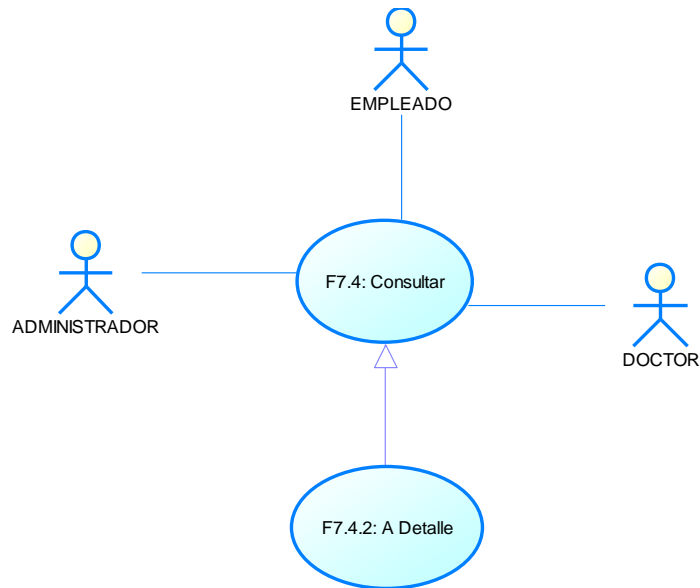


Diagrama 20 - Caso de Uso Consulta a Detalle de Horario de Atención

Elaborado por: Nathaly Herrera

Flujo Principal

1. El actor selecciona administrar horario de atención en el menú.
2. El sistema muestra la ventana de administrar horario de atención.
3. El actor ingresa la fecha, el código o la hora del horario de atención.
4. El sistema filtra en la tabla los horarios de atención. (E1)
5. El actor escoge el horario.
6. El sistema muestra todos los datos del horario de atención especificado.

Flujo alternativo

- 4) Si el horario de atención especificado no existe, ir al Caso de uso F7.1 (ingresar).

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

F7.4.3: Consulta General Horario de Atención

Descripción: El sistema permitirá al actor consultar todos los horarios de atención en el Centro Odontológico.

Actor: Administrador, empleado y Doctor

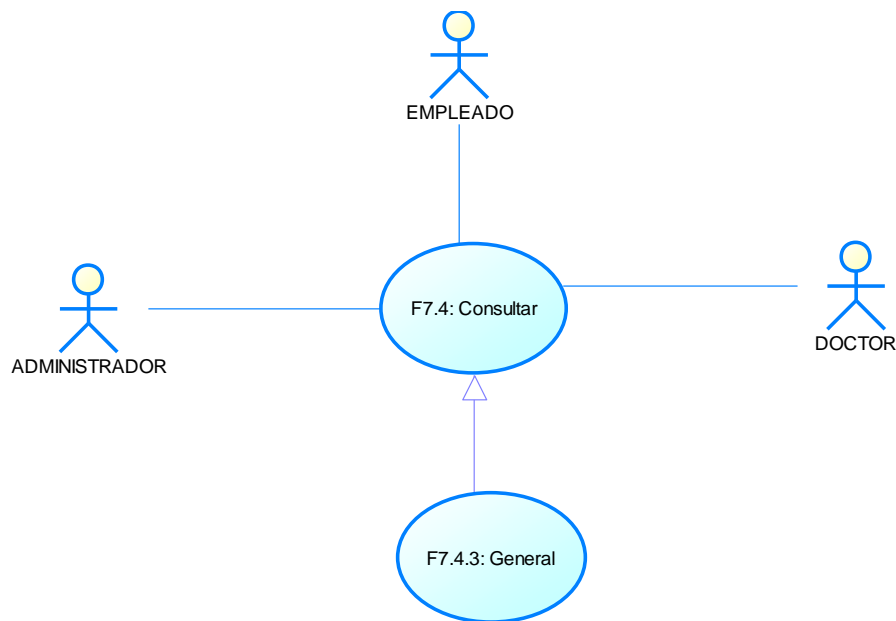


Diagrama 21 - Caso de Uso Consulta General Horario de Atención

Elaborado por: Nathaly Herrera

Flujo Principal

1. El actor selecciona administrar horario de atención en el menú.
2. El sistema muestra la ventana de administrar horario de atención.
3. El sistema muestra todos los horarios de atención. (E1)

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

Diagrama de clases

Son diagramas que permite detallar las clases con sus respectivos atributos y métodos que le describen, finalmente muestra la relación que tienen entre las diferentes clases.

Requerimientos de hardware y software

Dentro de los requerimientos es necesario examinar las necesidades a nivel de hardware y software, tal como se presenta a continuación:

Hardware

Servidor

- La memoria Ram 2GB.
- Disco Duro con 25GB libre
- 1 tarjeta de interfaz de red (NIC) con un ancho de banda de al menos 1 GB/s

Cliente

- Cualquier tipo de procesador
- La memoria Ram 500mb
- Disco duro 50mb espacio libre.

Software

- 1) El sistema funcionará en multiplataforma ya sea Linux, Mac o Windows

Prototipo

Ventana principal del paciente

Logo Home Paciente Horario de Atención Ficha Médica Ficha de la Cita Reservar Cita Cerrar Sesión

Paciente

CÉDULA:

Mostrar 10 Registros

Cédula	Primer Nombre	Segundo Nombre	Apellido Paterno	Apellido Materno	Fecha de Nacimiento	Género	Editar
--------	---------------	----------------	------------------	------------------	---------------------	--------	--------

Ilustración 12 - Interfaz ventana Principal Paciente

Elaborado por: Nathaly Herrera

Ventana Insertar Paciente

Logo Home Paciente Horario de Atención Ficha Médica Ficha de la Cita Reservar Cita Cerrar Sesión

Nuevo Paciente

Cédula:

Primer Nombre: Segundo Nombre:

Apellido Paterno: Apellido Materno:

Fecha Nacimiento: Ciudad De Origen:

Género: Masculino Femenino

Estado Civil:

Ilustración 13 - Interfaz de insertar paciente

Elaborado por: Nathaly Herrera

Ventana Modificar Paciente

Logo Home Paciente Horario de Atención Ficha Médica Ficha de la Cita Reservar Cita Cerrar Sesión

Editar Paciente

Cédula:

Primer Nombre: Segundo Nombre:

Apellido Paterno: Apellido Materno:

Fecha Nacimiento: Ciudad De Origen:

Género: Masculino Femenino

Estado Civil:

Ocupación:

Ilustración 14 - Interfaz Modificar Paciente

Elaborado por: Nathaly Herrera

Ventana Mostrar y Eliminar Paciente

Logo Home Paciente Horario de Atención Ficha Médica Ficha de la Cita Reservar Cita Cerrar Sesión

Paciente

Id	1
Primer Nombre	Zoe
Segundo Nombre	Micaela
Apellido Paterno	Martinez
Apellido Materno	Barriga
Fecha Nacimiento	2000-09-12
Género	F
Cédula	434334234
Ciudad Nacimiento	Quito
Estado Civil	Soltera
Ocupación	Ninguna
Dirección Domicilio	Chillos
Teléfono	4435543543
Celular	4343545435
Email	zoe@hotmail.com

[Regresar a la lista de pacientes](#)

[Editar](#) [Eliminar](#)

Ilustración 15 - Interfaz Mostrar y Eliminar Paciente

Elaborado por: Nathaly Herrera

3.1.2. Fase de Elaboración

3.1.2.1. Análisis y Diseño

Análisis del sistema

El Centro Odontológico Dental Access actualmente reserva las citas manualmente es decir en un cuaderno, el paciente llama o se acerca al Centro para registrar la cita, por lo que es muy complicado buscar y cancelar citas debido a que en el Consultorio no cuenta con una organización correcta de la misma y causa inconveniente a los pacientes.

El historial médico se lleva en papel y también cada cita se registra en diferentes escritos por lo que al momento de consultar una cita pasada resulta más complicado y se pierde tiempo en realizar esta tarea, a causa de la desorganización de los datos de cada paciente.

Debido a todos estos inconvenientes es necesario elaborar la aplicación web especificada ya que el sistema ayudará automatizar el proceso de reservar citas, el paciente podrá reservar la cita desde su casa o también llamar al Centro odontológico para que le registre en el sistema y se realizará un mejor registro de las citas, igualmente facilita el proceso de cancelación de una determinada cita.

Además, permitirá el sistema llevar un registro del historial médico de cada paciente y la ficha de cada cita, para que se pueda realizar búsquedas determinadas de los datos de cada usuario y sus consultas de una forma rápida, sin inconvenientes y para que Dental Access ofrezca un servicio de calidad.

La aplicación contará con un ingreso al sistema mediante un usuario y clave que se les asignará al empleado y doctor del Centro Odontológico para que así puedan administrar las diferentes funcionalidades desde cualquier parte y en el caso del usuario ingresará a la página de reserva de citas exclusivamente para el paciente sin necesidad de un usuario y contraseña solo con el requerimiento de estar registrado en el sistema, para realizar el proceso de reservar.

Diseño Arquitectónico

Para la realización de este proyecto se va utilizar el patrón de diseño arquitectónico MVC (Modelo, Vista y controlador), este patrón está asociado con la arquitectura de 3 capas. Debido a que la capa lógica del negocio y acceso a datos se encuentra en el modelo y la capa de presentación en la vista. Como se puede mostrar en la Ilustración 10.

- El Modelo al unir las dos capas: lógica del negocio y acceso a datos por lo que este elemento se encarga de buscar la información que requiere la aplicación, mediante la comunica con la base de datos para obtener dicha información.
- Vista es la interface de la aplicación en donde el usuario puede manipular o interactuar con el software, debido a que el modelo es transformado en una aplicación web que sea amigable.
- Controlador es el que se encarga de procesar la petición del usuario y realiza los cambios necesarios en el modelo o la vista, debido a que es el intermediador para la comunicación entre la vista y el modelo

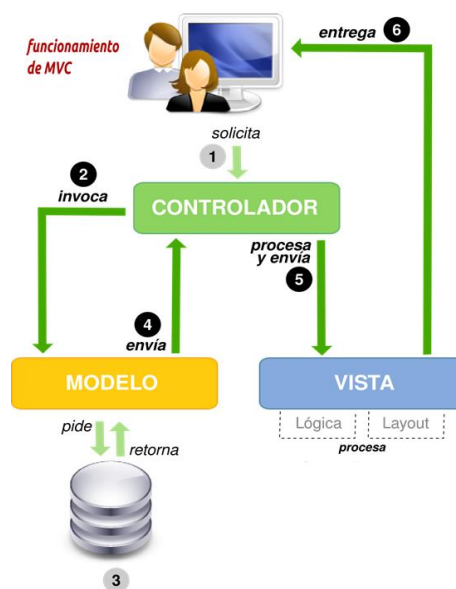


Ilustración 16 – MVC

(Expermicid, 2013)

Diagrama de paquetes (MVC)

El diagrama de paquetes indica de forma detallada como el sistema está estructurado y que archivos cuenta cada una de sus partes, para que se comprenda de manera organizada la estructura de la aplicación.

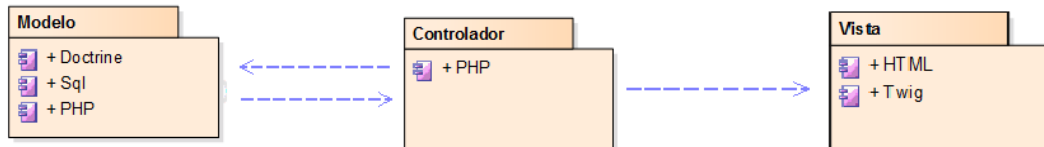


Diagrama 23 - Diagrama de Paquetes

Elaborado por: Nathaly Herrera

Diagrama de Secuencia

En la realización de este proyecto se utilizará el framework Symfony por lo que consta de varias funciones nativas para su funcionamiento correctos, que se ven reflejadas en los diagramas de secuencia como son: `find()`, `findOneByPcedula()` entre otras.

Los diagramas de secuencia que se presentan a continuación fueron elaborados por: Nathaly Herrera, la autora de la presente disertación.

Diagrama de Secuencia Paciente

Ingresar Paciente

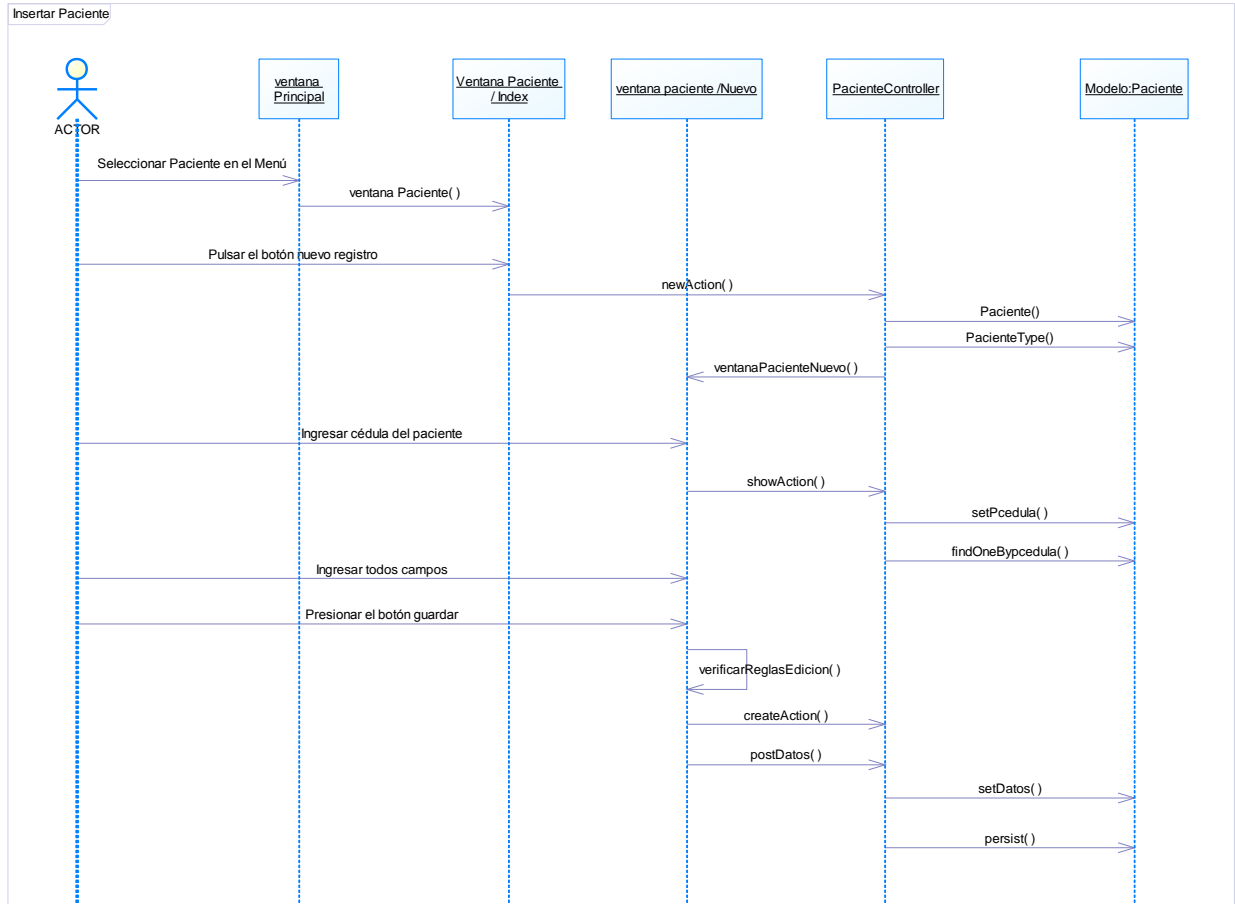


Diagrama 24 - Diagrama de Secuencia Insertar Paciente

Modificar Paciente

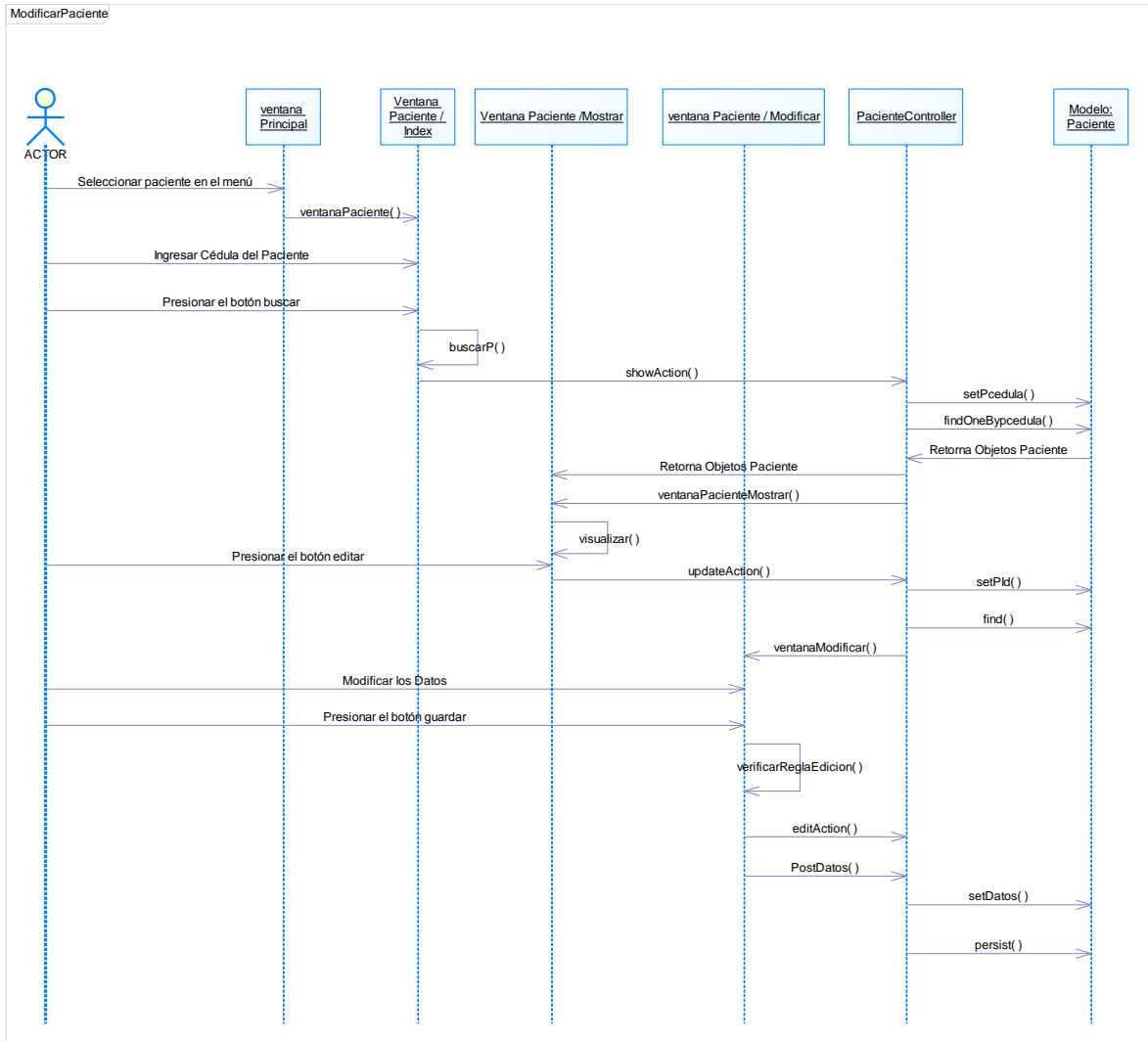


Diagrama 25 - Diagrama de Secuencia Modificar Paciente

Eliminar Paciente

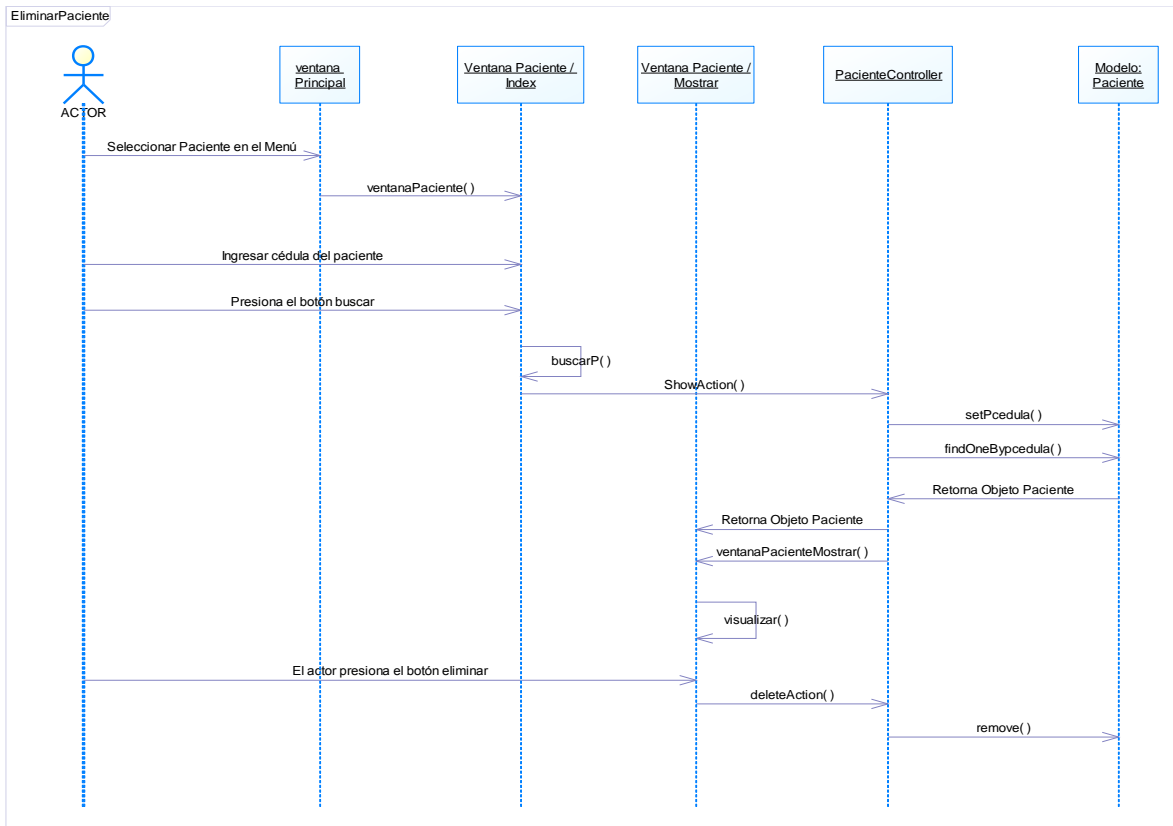


Diagrama 26 - Diagrama de Secuencia Eliminar Paciente

Consultar a Detalle Paciente

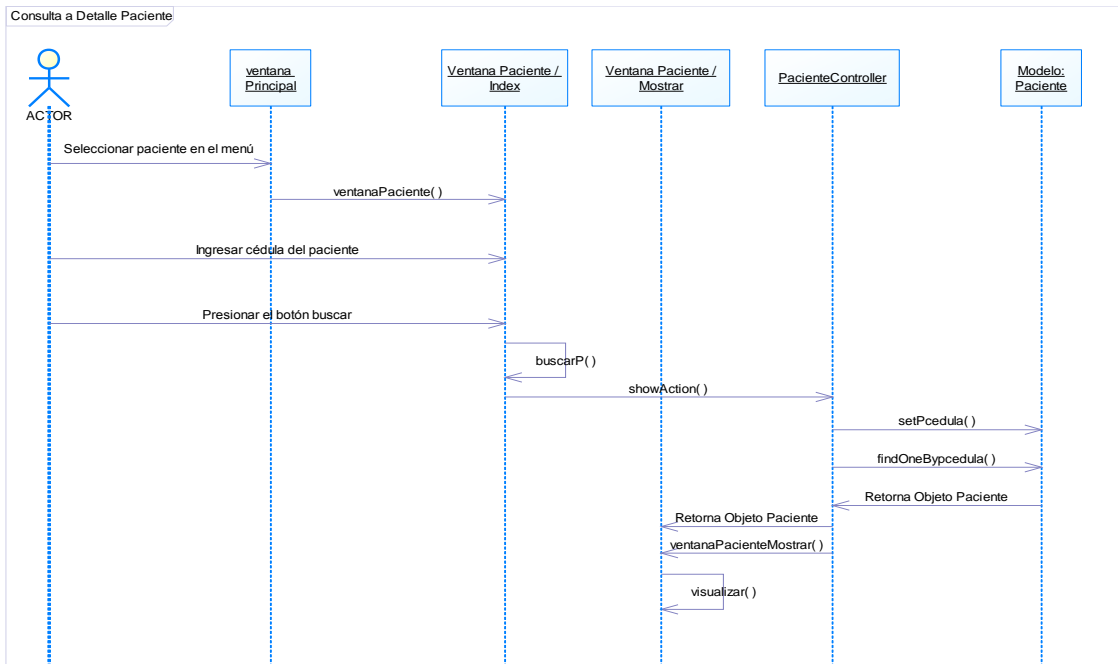


Diagrama 27 - Diagrama de Secuencia Consulta a Detalle Paciente

Consulta General Paciente

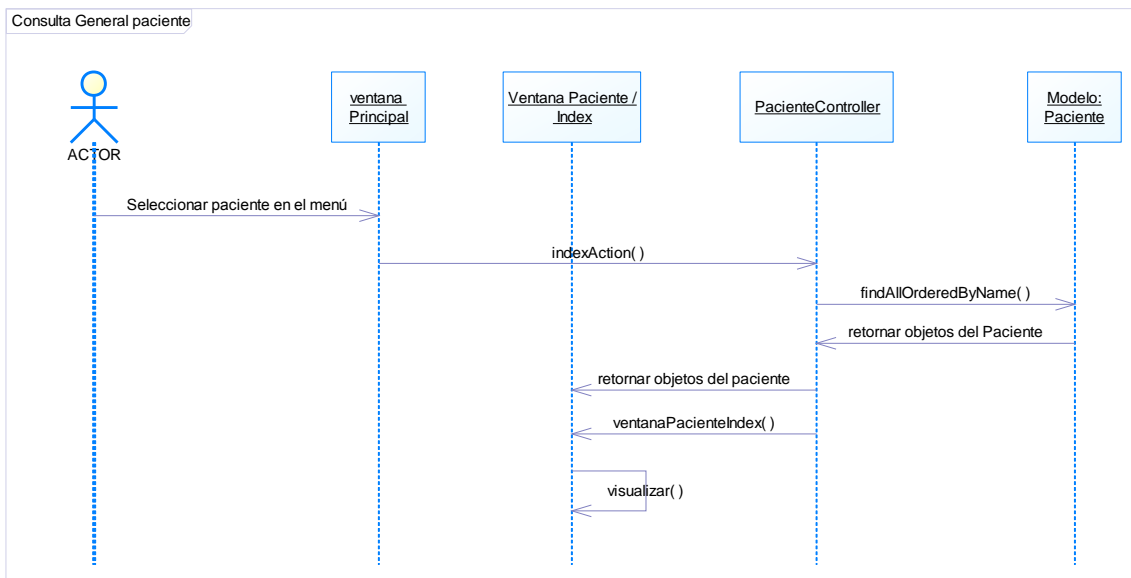


Diagrama 28 - Diagrama de Secuencia Consulta General Paciente

Diagrama de Secuencia Ficha Médica

Modificar Ficha Médica

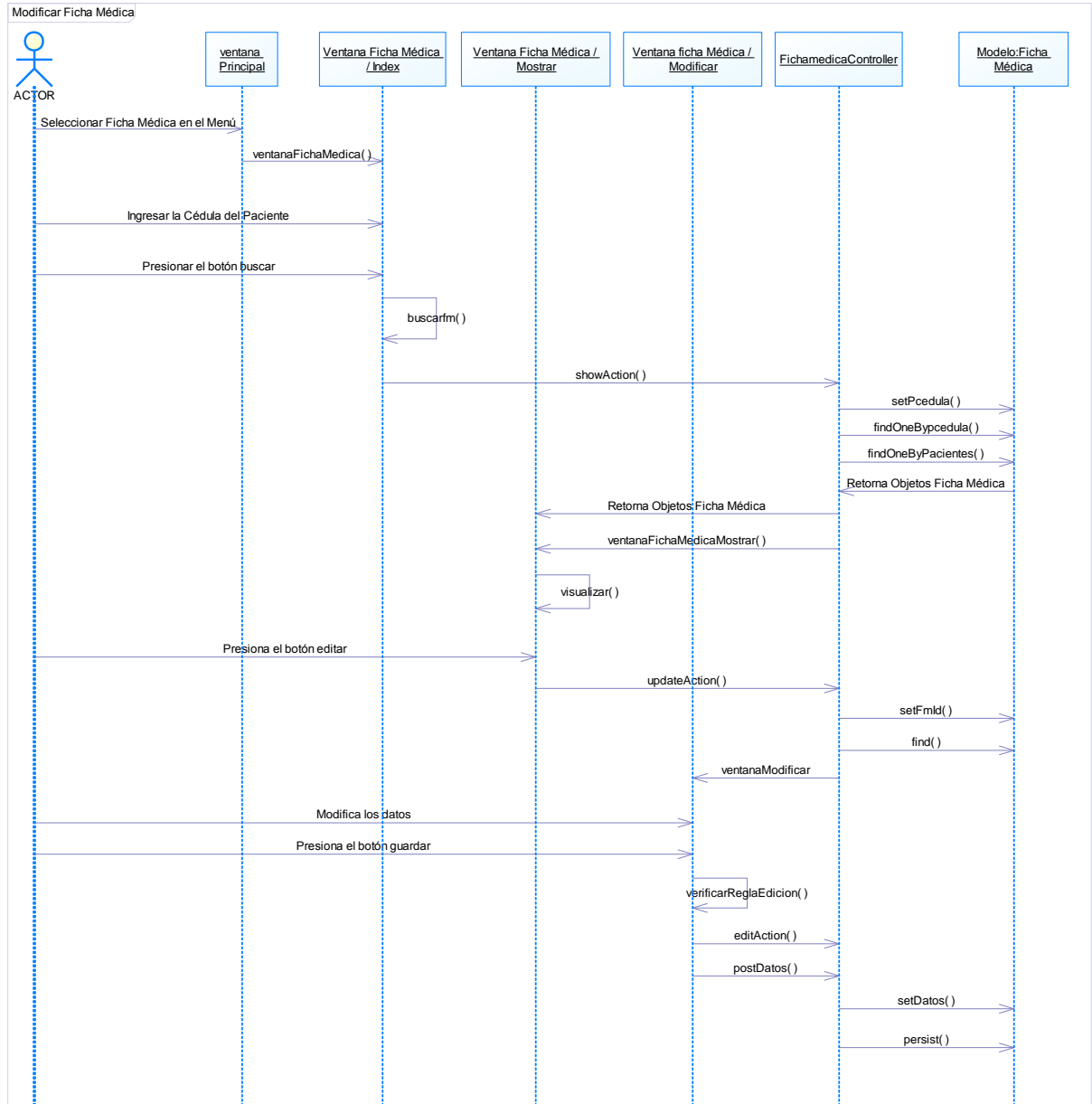


Diagrama 29 - Diagrama de Secuencia Modificar Ficha Médica

Eliminar Ficha Médica

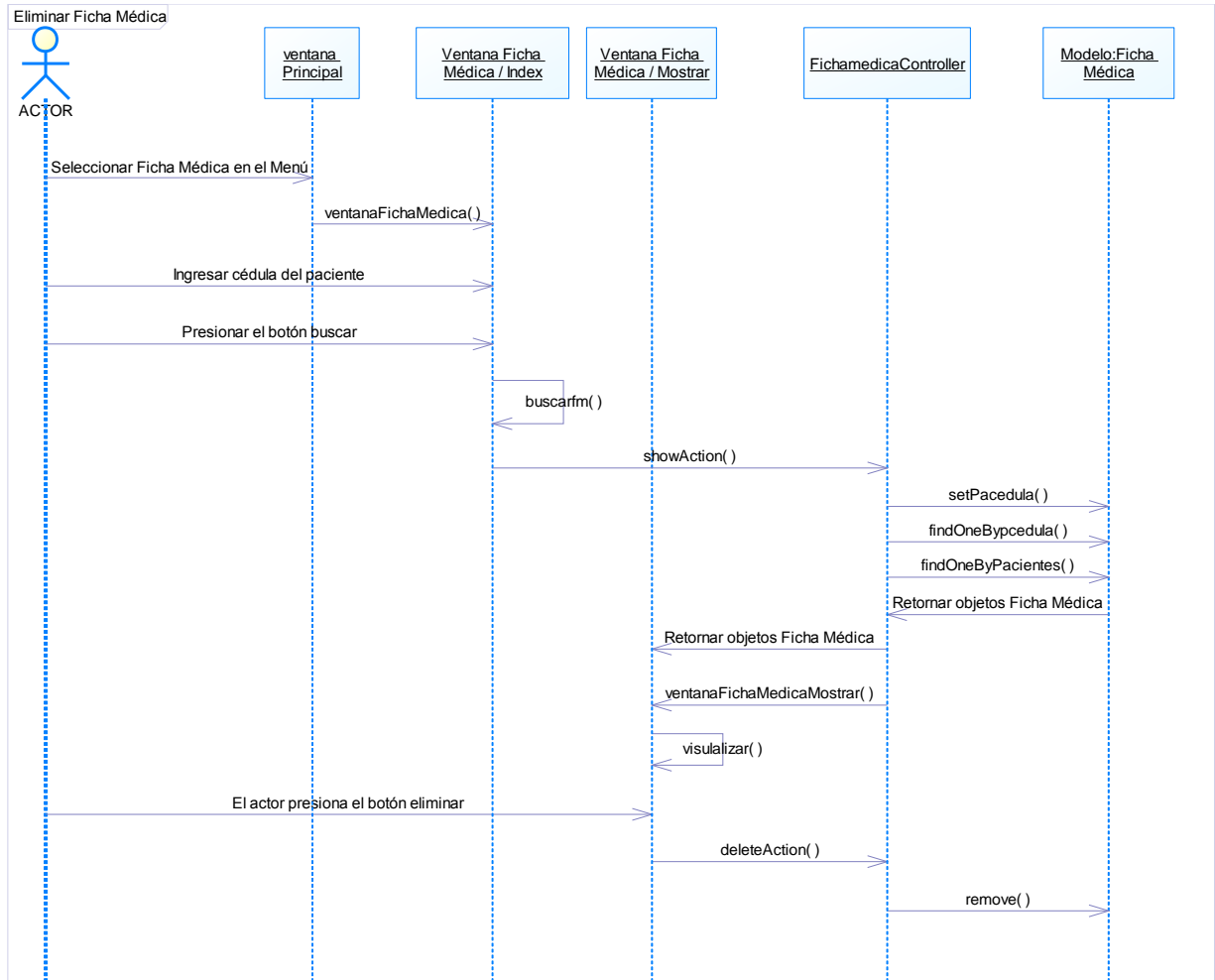


Diagrama 30 - Diagrama de Secuencia Eliminar Ficha Médica

Consulta a Detalle Ficha Médica

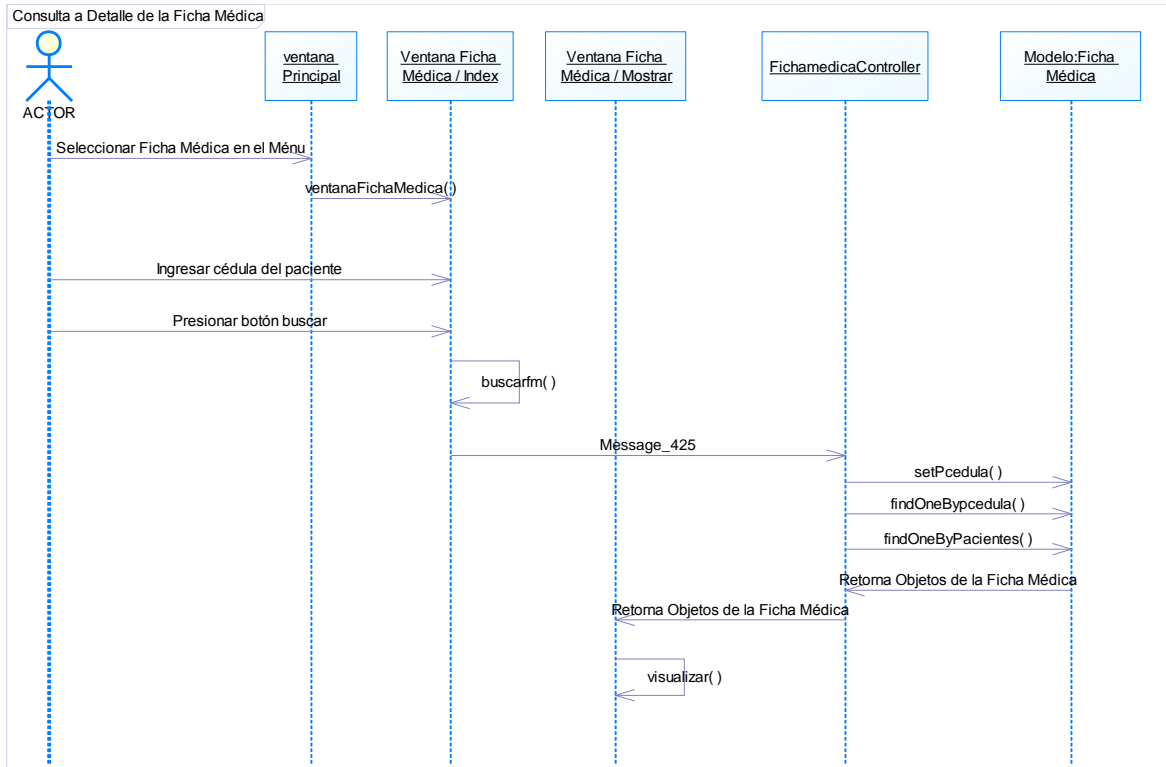


Diagrama 31 - Diagrama de Secuencia Consulta a Detalle Ficha Médica

Consulta General Ficha Médica

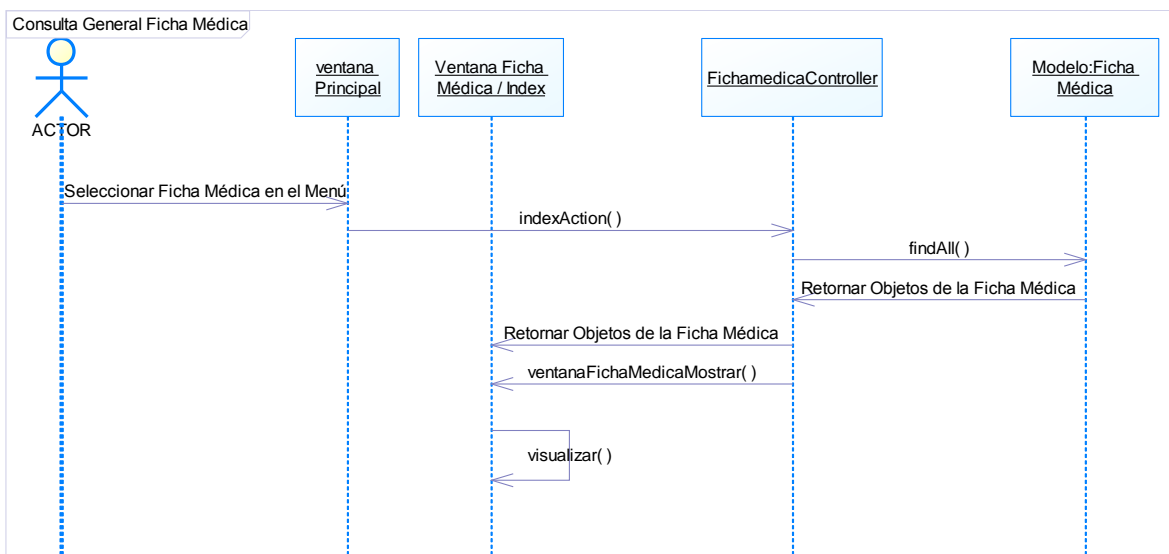


Diagrama 32 - Diagrama de Secuencia Consulta General Ficha Médica

Diagrama de Secuencia Horario de Atención

Insertar Horario de Atención

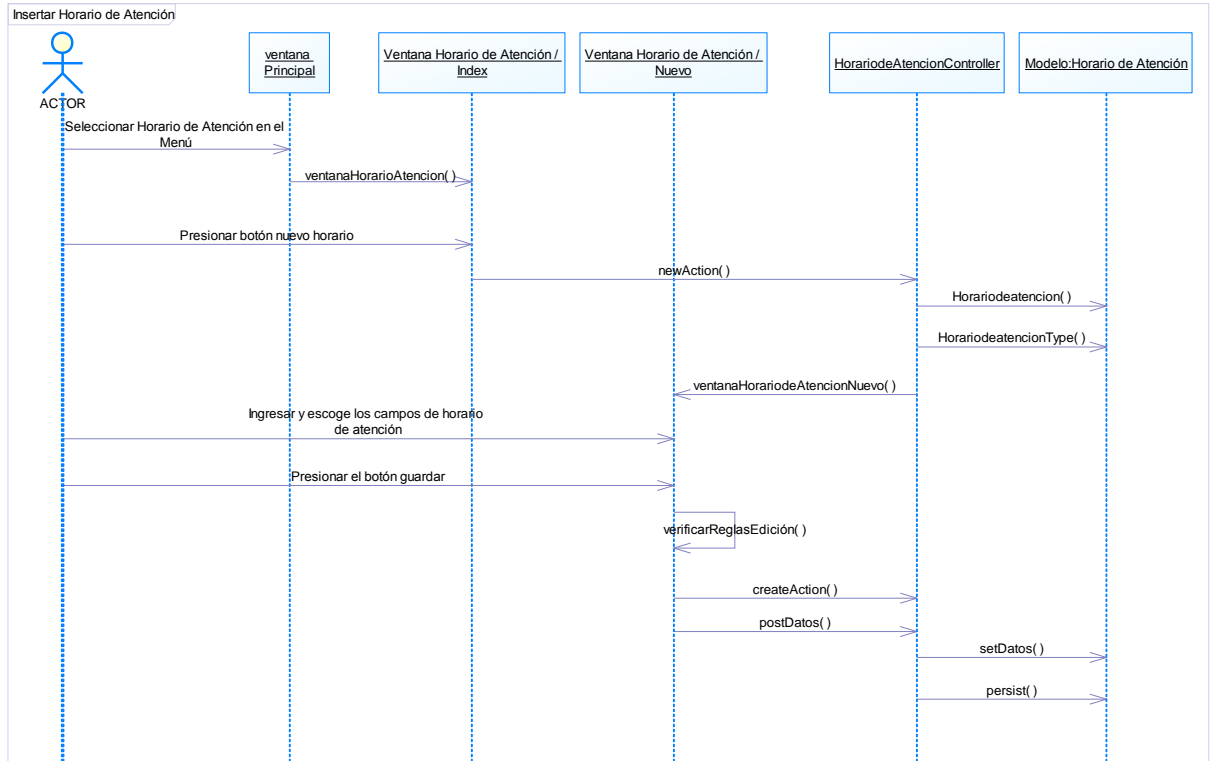


Diagrama 33 - Diagrama de Secuencia Insertar Horario de Atención

Modificar Horario de Atención

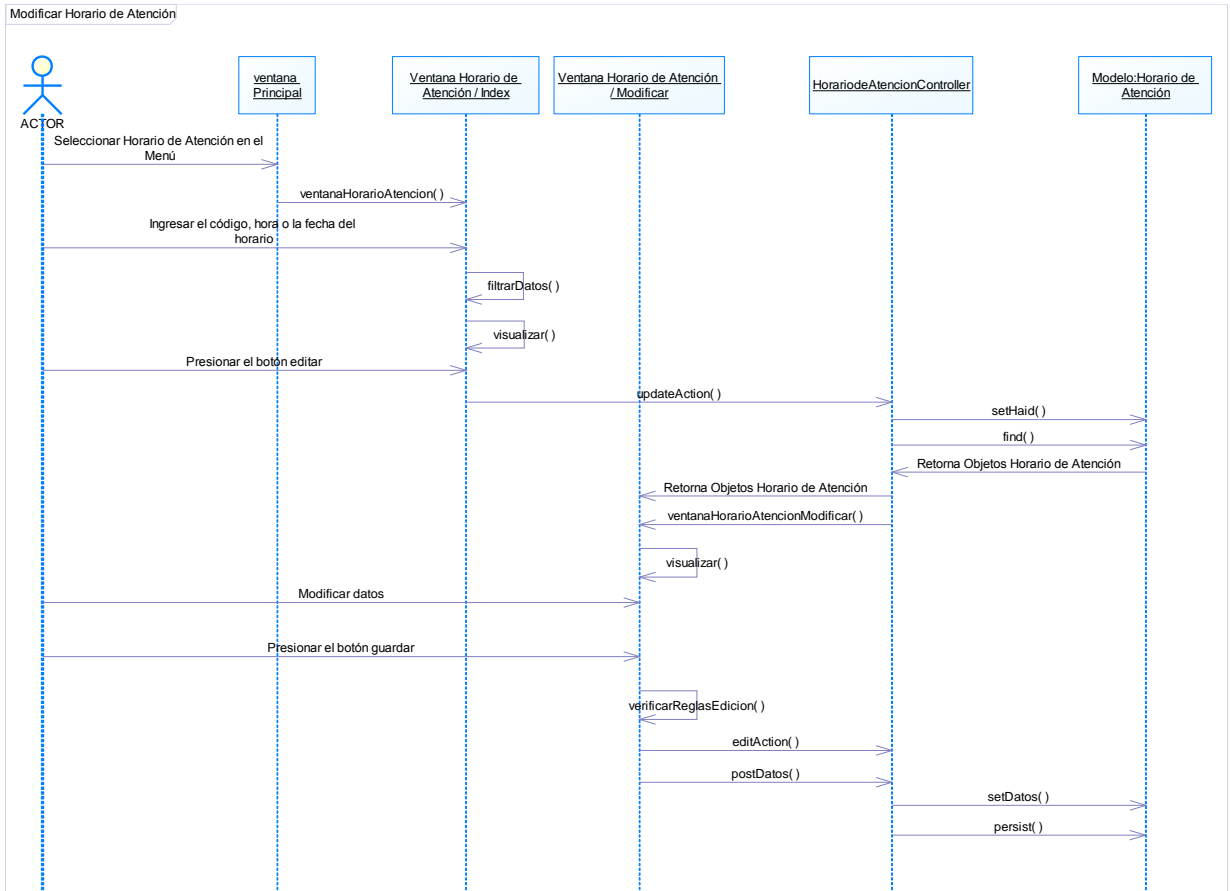


Diagrama 34 - Diagrama de Secuencia Modificar Horario de Atención

Eliminar Horario de Atención

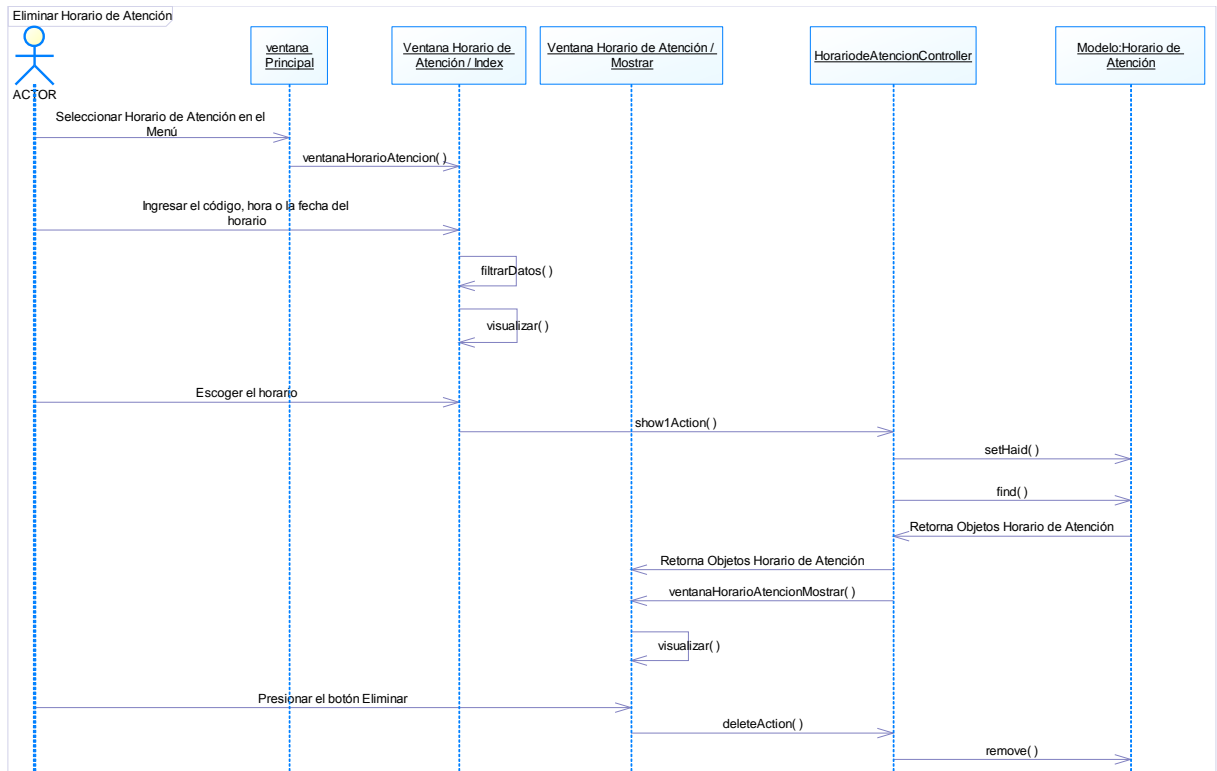


Diagrama 35 - Diagrama de Secuencia Eliminar Horario de Atención

Eliminar Horario de Atención por Doctor

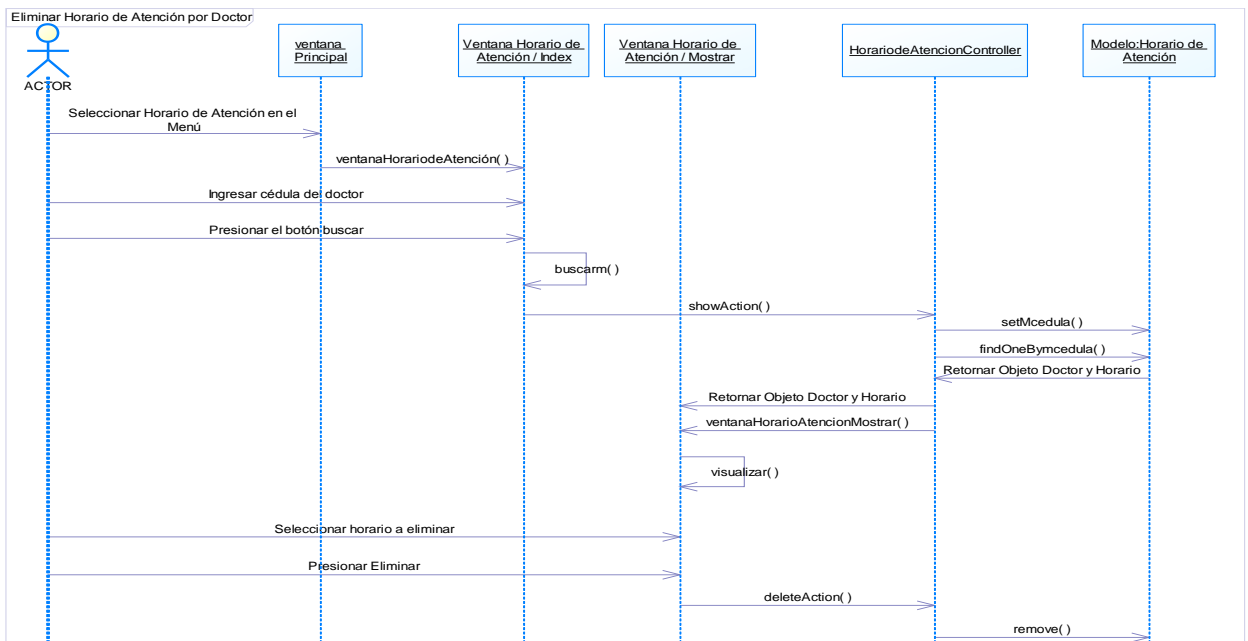


Diagrama 36 - Diagrama de Secuencia Eliminar Horario de Atención por Doctor

Consultar a Detalle Horario de Atención por Doctor

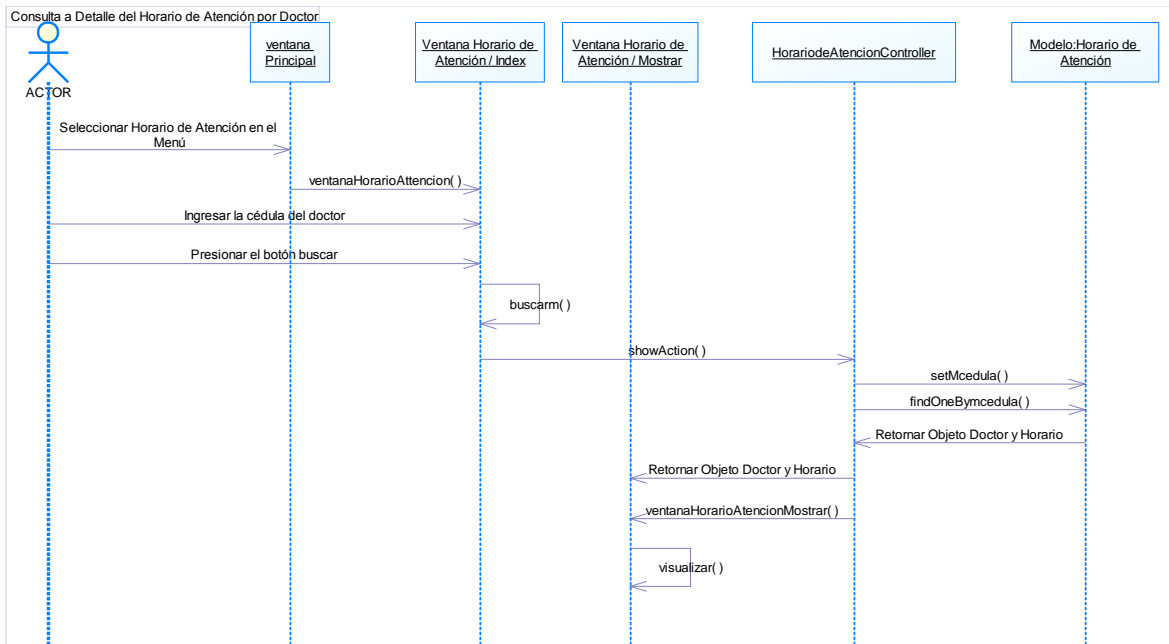


Diagrama 37 - Diagrama de Secuencia Consulta a Detalle Horario de Atención por Doctor

Consultar a Detalle Horario de Atención

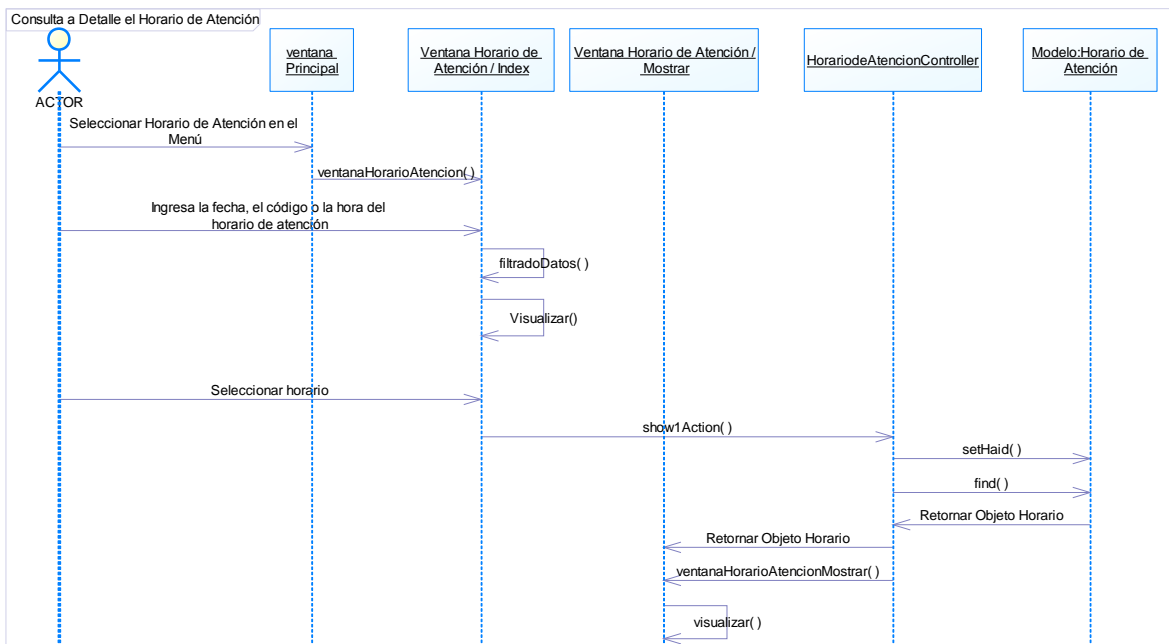


Diagrama 38 - Diagrama de Secuencia Consultar a Detalle Horario de Atención

Consulta General Horario de Atención

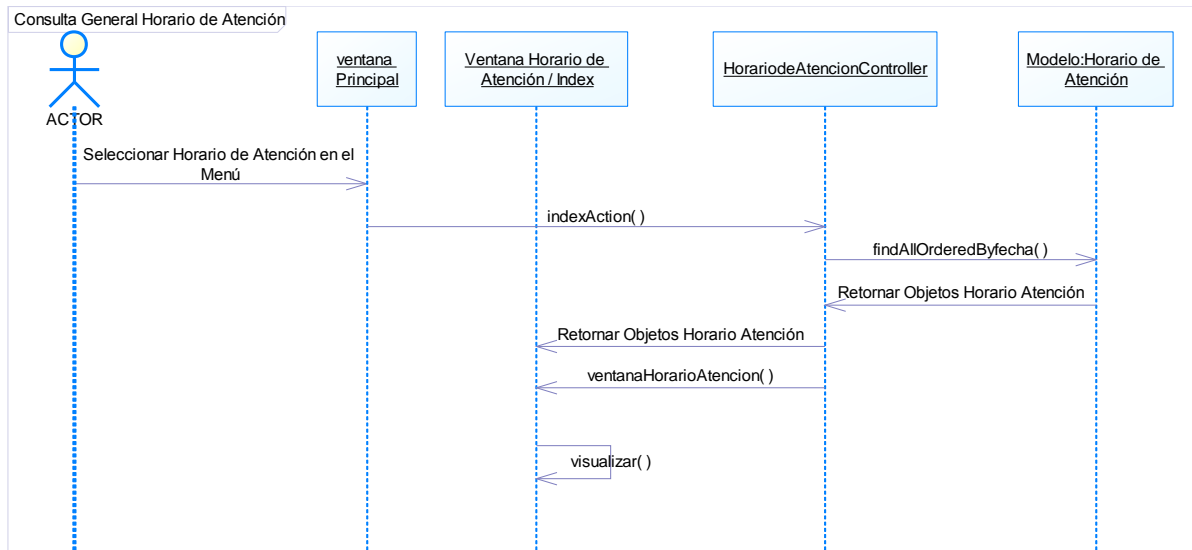


Diagrama 39 - Diagrama de Secuencia Consulta General Horario de Atención

Diagrama Físico

Es un diagrama que muestra las entidades con sus relaciones entre ellas, atributos con su respectivo tipo de dato, que se usa en la base de datos, para la creación se debe especificar la base de datos a utilizar.

CONSTRUCCIÓN DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE CITAS, TRATAMIENTOS
Y EXPEDIENTE MÉDICO DE LOS PACIENTES DEL CENTRO IMPLANTOLÓGICO Y DE
ESPECIALIDADES DENTAL ACCESS

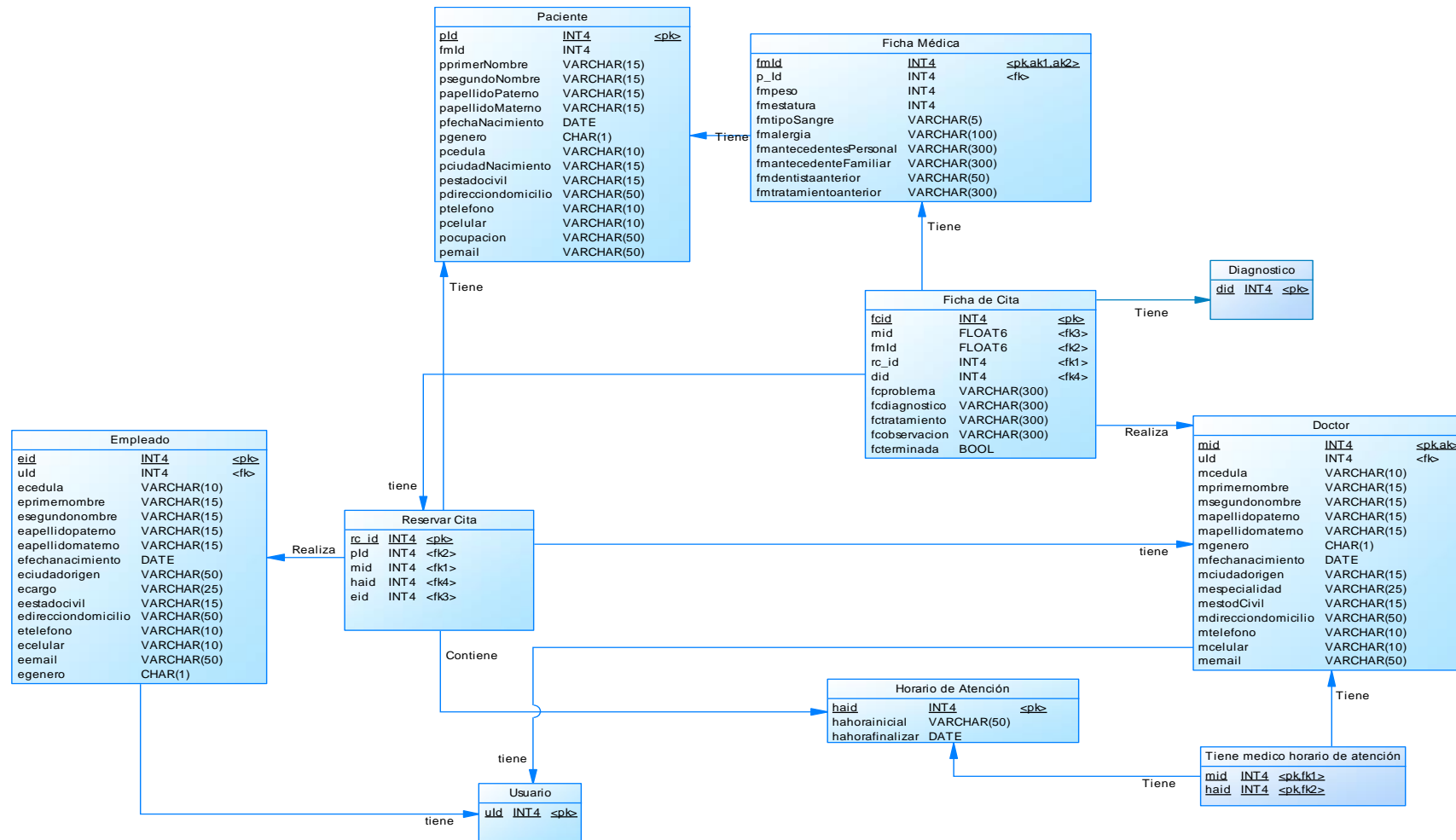


Diagrama 40 - Diagrama Físico

Elaborado por: Nathaly Herrera

3.1.3. Fase de Construcción

3.1.3.1. Implementación

La implementación de la aplicación web se realiza con el framework Symfony que involucra php, css y twig.

El código de la aplicación se encuentra en el CD, en una carpeta llamada implementación.

Descripción de las clases

Clase Empleado: Esta clase contiene toda la información de los empleados de los cuales son: secretaria y ayudante del Centro Odontológico. Para administrar esta clase empleado tiene permisos el administrador. Por el otro lado, el empleado tiene permisos para administrar paciente, horarios de atención, ficha médica, ficha de cita, reserva y cancelación de cita.

Clase Paciente: Esta clase contiene toda la información necesaria del paciente. Para administrar esta clase paciente tiene permisos el administrados, empleado y doctor. Por el otro lado, el paciente puede reservar citas.

Clase Doctor: Esta clase contiene toda la información necesaria de los doctores que atienden en el Centro Odontológico especificando su especialidad. Para administrar esta clase doctor tiene permisos el administrados. Por el otro lado, el doctor tiene permisos al igual que el empleado para administrar paciente, horarios de atención, ficha médica, ficha de cita, reserva y cancelación de cita.


Clase Ficha Médica: Esta clase contiene toda la información médica necesaria para realizar un seguimiento exhaustivo del historial médico del paciente. Para administrar esta clase ficha médica tiene permiso el administrador, empleado y doctor.

Clase Horario de atención: Esta clase contiene la fecha y hora que comienza y finaliza la cita de un determinado doctor del Centro Odontológico. Para administrar esta clase horario de atención tiene permisos el administrador, empleado y doctor.

Interfaz

Las imágenes de la interface que se presentan a continuación fueron elaboradas por: Nathaly Herrera, la autora de la presente disertación.

Validación Campo Vacío



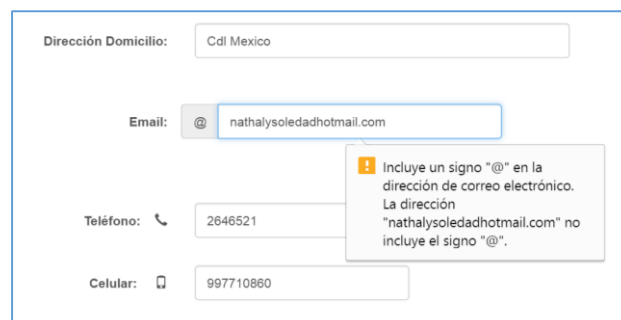
The screenshot shows a registration form with the following fields and values:

- Apellido Paterno: (empty field)
- Apellido Materno: Villarreal
- Fecha Nacimiento: 29/01/1955
- Ciudad De Origen: Quito
- Género: Masculino Femenino
- Estado Civil: Casado
- Ocupación: Consultor

A validation message is displayed over the empty 'Apellido Paterno' field: "Completa este campo".

Ilustración 17 - Validación Campo vacío

Validación Email: Debe tener el formato correcto de mail



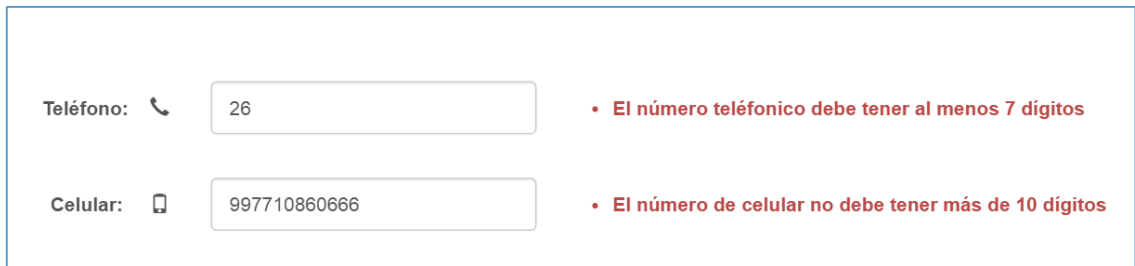
The screenshot shows a registration form with the following fields and values:

- Dirección Domicilio: Cdi Mexico
- Email: @ nathalysoledadhotmail.com
- Teléfono: 2646521
- Celular: 997710860

A validation message is displayed over the email field: "Incluye un signo '@' en la dirección de correo electrónico. La dirección 'nathalysoledadhotmail.com' no incluye el signo '@'".

Ilustración 18 - Validación Email

Validación campo teléfono: No debe tener menos de 7 dígitos o más de 10 dígitos



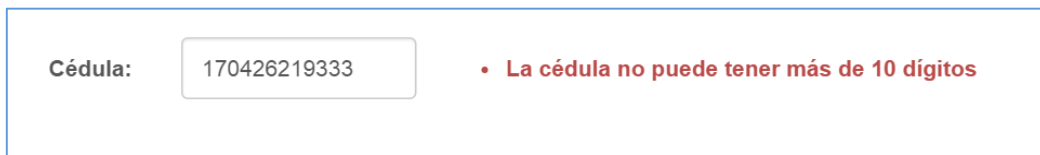
Teléfono: • El número telefónico debe tener al menos 7 dígitos

Celular: • El número de celular no debe tener más de 10 dígitos

Ilustración 19 - Validación Campo Teléfono

Validación Cédula

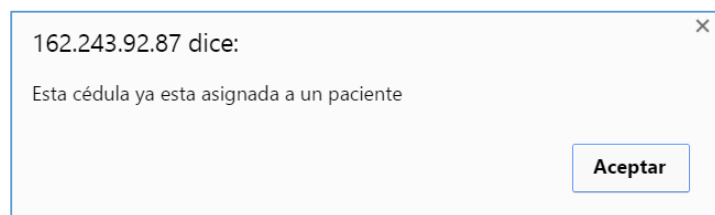
No puede tener más de 10 dígitos la cédula



Cédula: • La cédula no puede tener más de 10 dígitos

Ilustración 20 - Validación del Número de Dígitos de la Cédula

Notificar si la cédula la existe en la base de datos, al momento de ingresar un nuevo registro ya sea doctor, paciente o empleado



162.243.92.87 dice:
Esta cédula ya esta asignada a un paciente

Aceptar

Ilustración 21 - Notificación de que la cédula ya está registrada

Notificar si la cédula no existe en la base de datos al momento de buscar el doctor, paciente o empleado. Y se da dos opciones si desea cancelar o ingresar un nuevo registro con la cédula ingresada.

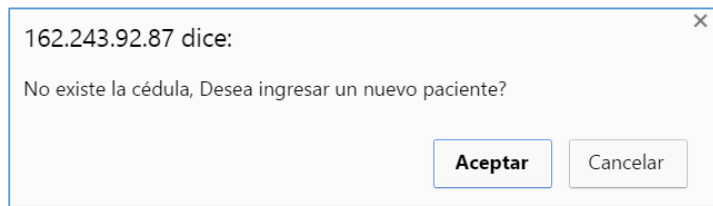


Ilustración 22 - Notificación del que la cédula que se está buscado no existe

Validación del número de caracteres

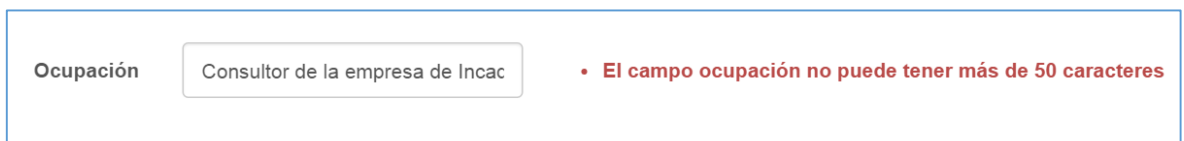


Ilustración 23 - Validación de número de caracteres

Eliminar

Confirmación si desea eliminar un registro.

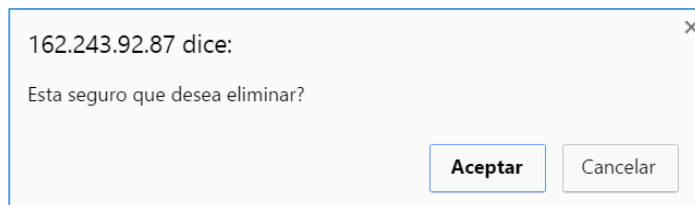


Ilustración 24 - Notificación de Eliminación

Interface del sistema (CRUD Paciente)

Página principal paciente

Cédula	Primer Nombre	Segundo Nombre	Apellido Paterno	Apellido Materno	Fecha de Nacimiento	Género	Editar
1704262193	Gonzalo	Oriando	Herrera	Villarreal	1955-01-29	M	✎
1727453476	Christian	Gonzalo	Parra	Haro	1992-07-07	M	✎
1745239876	Daniela	Stephanie	Tobar	Cifuentes	1994-05-06	F	✎

Ilustración 25 - Interfaz Página Principal Paciente

Ingresar Paciente

Nuevo Paciente

Cédula: 1745239876

Primer Nombre: Segundo Nombre:

Apellido Paterno: Apellido Materno:

Fecha Nacimiento: dd/mm/aaaa Ciudad De Origen:

Género: Masculino Femenino

Estado Civil:

Ocupación:

Dirección Domicilio:

Email: @

Ilustración 26 - Interfaz Ingresar paciente

Modificar Paciente

Editar Paciente

Cédula:

Primer Nombre: Segundo Nombre:

Apellido Paterno: Apellido Materno:

Fecha Nacimiento: Ciudad De Origen:

Género: Masculino Femenino

Estado Civil:

Ilustración 27 - Interfaz Modificar Paciente

Consultar y Eliminar Paciente

Paciente	
Id	3
Primer Nombre	Daniela
Segundo Nombre	Stephanie
Apellido Paterno	Tobar
Apellido Materno	Cifuentes
Fecha Nacimiento	1994-05-06
Género	F
Cédula	1745239876
Ciudad Nacimiento	Quito
Estado Civil	casada
Ocupación	Parvularia
Dirección Domicilio	5 esquinas
Teléfono	2768754
Celular	997645328
Email	nathalyherrera14@yahoo.com
<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>	
Regresar a la lista de pacientes	

Ilustración 28 - Interface Mostrar y Eliminar paciente

Interface del sistema (CRUD Horario de Atención)

Página principal Horario de Atención

Inicio Usuario Empleado Doctor Paciente Horario de Atención Ficha Médica Ficha de la Cita Reservar Cita Cerrar Sesión

Horario de Atención

Ingrese la cédula del doctor para buscar sus Horarios de Atención

Cédula:

Mostrar Registros Buscar:

Id	Fecha	Hora Inicial	Hora Finalizar	Tipo de Atención	Editar
Ningún dato disponible en esta tabla					

Mostrando registros del 0 al 0 de un total de 0 registros

Ilustración 29 - Interfaz Pagina Inicial del Horario de Atención

Ingresar Horario de Atención

Crear Horario de Atención

Cédula Doctor:

Fecha:

Tipo de Atención:

Hora:

[Regresar a la lista de horarios de atención](#)

Ilustración 30 - Interface Ingresar Horario de Atención

Notificar Cuando un doctor ya tiene horario de atención el mismo día y en el mismo rango de hora.

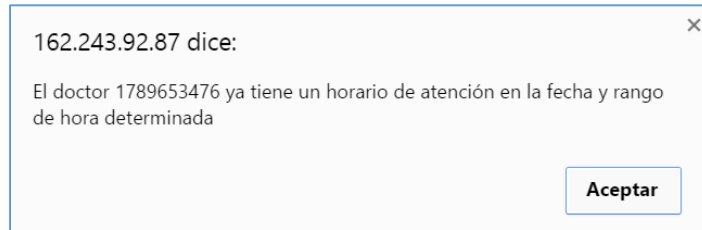


Ilustración 31 - Notificación de coincidencia de horario

Modificar Horario de Atención

Editar Horario de Atención

Cédula Doctor: ▲

 ▼

Fecha:

Tipo de Atención: ▼

Hora:

Guardar 

Ilustración 32 - Interface Modificar Horario de Atención

Mostrar y Eliminar Horario de Atención

Muestra los datos del Horario de Atención, con sus respectivos doctores.

Horario De Atención

Id	1
Fecha	2016-06-01
Hora Inicial	09:00
Hora Final	09:45
Tipo de Atención	Tratamiento

Lista de los doctores en un horario de atención determinado

Cédula	Primer Nombre	Segundo Nombre	Apellido Paterno	Apellido Materno	Género	Fecha Nacimiento
1789653476	Luis		Hernandez	Herrera	M	1985-10-10
1787236736	Karla	Daniela	Lazo	Arroyo	F	1970-05-10

[Editar](#) [Eliminar](#)

Ilustración 33 - Interface Mostrar y Eliminar Horario de Atención

Muestra los horarios de atención del doctor especificado

Horario de Atención

Datos del Doctor

Cédula	1789653476
Nombre	Luis Hernandez Herrera
Género	M
Fecha Nacimiento	1985-10-10
Especialidad	Implantología

Lista de los horarios de atención del doctor determinado

Mostrar Registros

Id	Fecha	Hora Inicial	Hora Finalizar	Tipo de Atención	Eliminar
1	2016-06-01	09:00	09:45	Tratamiento	
3	2016-06-01	10:00	10:30	Consulta	

Mostrando registros del 1 al 2 de un total de 2 registros

[Anterior](#) [1](#) [Siguiente](#)

Ilustración 34 - Interface Mostrar y Eliminar Horario de Atención por Doctor

Interface del sistema (CRUD Ficha Médica)

Página principal Ficha Médica

Ingrese la cédula del paciente para buscar su Ficha Médica

Cédula:

Mostrar 10 Registros

Buscar:

Cédula del paciente	Nombre del paciente	Apellido del paciente	Tipo de Sangre	Alergia	Editar
1704262193	Gonzalo	Herrera			
1727453476	Christian	Olivo			
1745239876	Daniela	Tobar			

Mostrando registros del 1 al 3 de un total de 3 registros

Anterior 1 Siguiente

Ilustración 35 - Interface Página Principal Ficha Médica

Modificar Ficha Médica

Editar Ficha Médica

Cédula del paciente: 1704262193 Nombre del paciente: Gonzalo Herrera

Tipo de Sangre:

Alergia:

Antecedentes Personales:

Antecedentes Familiares:

Ilustración 36 - Interface Modificar Ficha Médica

Mostrar y Eliminar Ficha Médica

Fichamedica

Cédula del Paciente	1704262193
Nombre del paciente	Gonzalo Herrera
Id Ficha Médica	1
Tipo de Sangre	
Alergia	
Antecedentes personales	
Antecedentes familiares	
Doctor Anterior	
Tratamiento Anterior	

[Regresar a la lista de ficha médica](#)

[Editar](#) [Eliminar](#)

Ilustración 37 - Mostrar y Eliminar Ficha Médica

3.1.4. Fase de Transición

3.1.4.1. Pruebas

Las pruebas se realizan a los usuarios finales a los que van utilizar el sistema, que en este caso son los del Centro Odontológico Dental Access. Las pruebas permiten probar el sistema y así identificar si se está cumpliendo con lo descrito en los casos de uso caso contrario realizar los cambios detectados en esta fase, para así obtener un sistema que cumpla con todos los requerimientos del usuario y así tener la aprobación del Centro Odontológico.

Caso de Uso: F3: Paciente

Pre-Condición:

El usuario debe tener permisos para gestionar los pacientes.

Para modificar, eliminar y consultar debe existir al menos un registro.

Funcionalidad	Entrada	Resultados Esperados	Cumple?
F3: Administrar Paciente	Seleccionar en el menú la opción de paciente.	Se despliega la pantalla principal del paciente	OK
F3.1: Ingresar	<ul style="list-style-type: none"> • Presiona el botón de nuevo registro. • Ingresar la cédula del paciente. • Ingresar los demás datos del paciente. • Se ingresa algunos datos incorrectos como el mail y el número de teléfono. 	<ul style="list-style-type: none"> • Se despliega la pantalla para el ingreso de un nuevo registro. • El sistema verifica que la cédula no existe en la base de datos. Caso contrario ir al Caso de uso F3.3 (eliminar) o F.4.1 (consultar). • Se ingresa correctamente y se despliega la pantalla de mostrar datos del paciente nuevo. • No guarda los datos, e indica el error que ocurrió al ingresar los datos. 	<ul style="list-style-type: none"> • OK • OK • OK • OK
F3.2: Modificar	<ul style="list-style-type: none"> • Presiona el botón Editar paciente. • Modifica los datos. • Ingresar un dato incorrecto. 	<ul style="list-style-type: none"> • Despliega la pantalla con todos los datos del paciente determinado. • El sistema modifica correctamente los datos y muestra los datos del paciente determinado. • No le permite modificar los datos e indica el error que ocurrió. 	<ul style="list-style-type: none"> • OK • OK • OK

Funcionalidad	Entrada	Resultados Esperados	Cumple?
F3.3: Eliminar	Presiona el botón eliminar	Muestra un mensaje para confirmar si se desea eliminar, al presionar ok se elimina correctamente y al presionar cancelar no se elimina el registro	OK
F3.4.1: Consultar a Detalle	<ul style="list-style-type: none"> • Buscar por cédula al paciente. • Buscar una cédula que no existe. 	<ul style="list-style-type: none"> • Despliega toda la información del paciente. • Muestra una notificación de que la cédula no existe y da la opción de crear un nuevo paciente con la determinada cédula o no. 	<ul style="list-style-type: none"> • OK • OK
F3.4.2: Consulta General	Presionar en el menú paciente.	Despliega la pantalla principal con todos los pacientes	OK

Tabla 7 - Pruebas del Caso de Uso Paciente

Elaborado por: Nathaly Herrera

Post-Condición:

Si se ingresa, modifica o elimina algún registro se realizará los cambios en la base de datos respectivamente.

Caso de Uso: F7: Horario de atención

Pre-Condición:

El usuario debe tener permisos para gestionar horario de atención.

Para modificar, eliminar y consultar debe existir al menos un registro.

Debe existir al menos un doctor ingresado en el sistema.

Funcionalidad	Entrada	Resultados Esperados	Cumple?
F7: Administrar Horario de Atención	Seleccionar en el menú la opción de horario de atención.	Se despliega la pantalla principal del horario de atención	OK
F7.1: Ingresar	<ul style="list-style-type: none"> • Presiona el botón de nuevo horario. • Escoger los datos del horario de atención. • Se deja vacío algunos datos. • Se escoge un horario para un determinado doctor que coincida con un anterior 	<ul style="list-style-type: none"> • Se despliega la pantalla para el ingreso de un nuevo horario. • Se ingresa correctamente y se despliega la pantalla de mostrar datos del horario de atención nuevo. • No guarda los datos, e indica campo se encuentra vacío al ser obligatorio. • No guarda los datos y notificar mediante un alerta que el determinado doctor ya tiene un horario la misma fecha y en el mismo rango de hora. 	<ul style="list-style-type: none"> • OK • OK • OK • OK
F7.2: Modificar	<ul style="list-style-type: none"> • Presiona el botón Editar horario de atención. • Modifica los datos. • Se deja algunos datos vacíos. 	<ul style="list-style-type: none"> • Despliega la pantalla con todos los datos del horario de atención determinado. • El sistema modifica correctamente los datos y 	<ul style="list-style-type: none"> • OK • OK • OK

Funcionalidad	Entrada	Resultados Esperados	Cumple?
		<p>muestra los datos del horario determinado.</p> <ul style="list-style-type: none"> No le permite modificar los datos e indica que campo se encuentra vacío, debido a que es obligatorio. 	
F7.3.1: Eliminar	Presiona el botón eliminar	Muestra un mensaje para confirmar si se desea eliminar, al presionar ok se elimina correctamente y al presionar cancelar no se elimina el registro.	OK
F7.3.2: Eliminar por doctor	<ul style="list-style-type: none"> Busca al doctor. Ingresa una cédula que no existe. Presiona el botón eliminar. 	<ul style="list-style-type: none"> Ingresa la cédula del doctor y despliega todos los horarios de atención del determinado doctor. Sale una notificación de que la cédula no existe y se queda en la página principal. Muestra un mensaje para confirmar si se desea eliminar, al presionar ok se elimina correctamente y al presionar cancelar no se elimina el registro. 	<ul style="list-style-type: none"> OK OK OK

Funcionalidad	Entrada	Resultados Esperados	Cumple?
F7.4.1: Consulta a Detalle por Doctor	<ul style="list-style-type: none"> • Buscar por cédula del doctor. • Buscar una cédula que no existe. 	<ul style="list-style-type: none"> • Despliega todos los horarios de atención del determinado doctor. • Muestra una notificación de que la cédula no existe y se queda en la página principal. 	<ul style="list-style-type: none"> • OK • OK
F7.4.2: Consulta a Detalle	Buscar por código del horario de atención.	Despliega toda la información del horario de atención determinado.	OK
F3.4.2: Consulta General	Presionar en el menú horario de atención.	Despliega la pantalla principal con todos los horarios de atención.	OK

Tabla 8 - Pruebas del Caso de Uso Horario de Atención

Elaborado por: Nathaly Herrera

Post-Condición:

Si se ingresa, modifica o elimina algún registro se realizará los cambios en la base de datos respectivamente.

Caso de Uso: F6: Ficha Médica

Pre-Condición:

El usuario debe tener permisos para gestionar ficha médica.

Para modificar, eliminar y consultar debe existir al menos un registro.

La ficha médica se crea automáticamente con campos vacíos al momento de ingresar un paciente, por lo que debe estar ingresado al menos un paciente para tener ficha médica.

Funcionalidad	Entrada	Resultados Esperados	Cumple?
F6: Administrar Ficha Médica	Seleccionar en el menú la opción de ficha médica.	Se despliega la pantalla principal ficha médica.	OK
F6.1: Modificar	<ul style="list-style-type: none"> • Presiona el botón Editar ficha médica. • Modifica los datos. • Ingresar datos incorrectos. 	<ul style="list-style-type: none"> • Despliega la pantalla con todos los datos de la ficha médica determinado. • El sistema modifica correctamente los datos y muestra los datos de la ficha médica. • No le permite modificar los datos e indica que campo tiene el error. 	<ul style="list-style-type: none"> • OK • OK • OK
F6.2: Eliminar	Presiona el botón eliminar	Muestra un mensaje para confirmar si se desea eliminar, al presionar ok se elimina correctamente y al presionar cancelar no se elimina el registro.	OK
F6.3.1: Consulta a Detalle	<ul style="list-style-type: none"> • Buscar por cédula del paciente. • Buscar una cédula que no existe. 	<ul style="list-style-type: none"> • Despliega toda la información de la ficha médica del paciente determinado. • Muestra una notificación de que la cédula no existe y se queda en la página principal. 	<ul style="list-style-type: none"> • OK • OK

Funcionalidad	Entrada	Resultados Esperados	Cumple?
F6.3.2: Consulta General	Presionar en el menú ficha médica.	Despliega la pantalla principal con todas las fichas médicas.	OK

Tabla 9 - Pruebas del Caso de Uso Ficha de Cita

Elaborado por: Nathaly Herrera

Post-Condición:

Si se modifica o elimina algún registro se realizará los cambios en la base de datos respectivamente.

Resultados

Después de realizar las pruebas de la primera iteración se puede determinar, que se está cumpliendo con todos los requerimientos establecidos por el usuario y que no se ha encontrado ningún error ni observación de los 5 requerimientos establecidos en esta primera parte. El usuario probó con datos reales del Centro Odontológico, y se ingresó correctamente toda la información. Pero el usuario solicitó que se agregue un link para imprimir los formularios de consentimientos, que se firman al momento de realizar un tratamiento al paciente.

3.1.4.2. Gestión de Cambios y Configuraciones

El cambio que se va a realizar no afecta a los requerimientos, solo es un cambio en la interface, que consiste en aumentar en la ficha médica un link para imprimir los formularios de consentimientos, El cambio se realizará en la segunda iteración. A continuación, se presenta la ficha en la tabla para realizar los cambios.

Gestión de Cambios			
Nombre:	Nathaly Herrera	Fecha:	may-16
Información del Producto			
Nombre:	Aplicación web para el Centro Odontológico Dental Access	Dueño:	Nathaly Herrera
Información del cambio			
Razón del Cambio			
El usuario solicito que se aumente el link para los formularios de consentimiento			
Beneficios del cambio			
El usuario podrá imprimir estos formularios en cualquier momento que lo necesite. Especialmente cuando inicia un tratamiento el paciente.			
Descripción del Cambio			
Aumentar href en el index de ficha médica y redireccionar al pdf			
Aumentar el pdf en la carpeta de la aplicación			
Estado:			
Aprobado:	SI	Información	
Descripción:			
Ninguna			

Tabla 10 - Gestión de Cambios y Configuración

Elaborado por: Nathaly Herrera

3.1.4.3. Entorno

La aplicación web se encuentra en un servidor local llamado XAMPP.

Se utiliza la base de datos postgresql

Para su elaboración el framework Symfony que está basado en php y twig.

3.2. Segunda Iteración

En este segundo ciclo para la parte de diseño y funcionalidad del programa se realiza los 5 requerimientos funcionales de los 10 establecidos. La presentación o realización del sistema es con el lenguaje de programación PHP, el Framework Symfony y con la base de datos en Postgresql.

3.2.1. Fase de Iniciación

3.2.1.1. Requerimientos

Requerimientos funcionales

F1: El sistema permitirá el ingreso al sistema

F2: El sistema gestionará empleados

F3: El sistema gestionará pacientes

F4: El sistema gestionará doctores

F5: El sistema gestionará ficha de citas

F6: El sistema gestionará ficha médica

F7: El sistema gestionará horario de atención

F8: El sistema permitirá reserva de citas

F9: El sistema permitirá cancelación de citas

F10: El sistema gestionará usuarios para el ingreso al sistema

Los Requerimientos que se realiza en la segunda iteración son 5 restantes los cuales son:
Ingreso al sistema, Ficha de Cita, Usuario, Reservar de Cita y Cancelar de Cita.

De los 5 requerimientos se especifica los 4 más representativos para el proyecto los cuales son: Ingreso al sistema, Ficha de Cita, Reservar de Cita y Cancelar de Cita, y el otro requerimiento se encuentra en el CD.

Requerimientos no funcionales

NF1: El sistema necesita como mínimo 2GB de memoria RAM.

NF2: El sistema será eficiente es decir que tendrá un tiempo de respuesta de 5seg.

NF3: El sistema es multiplataforma (PHP).

NF4: El sistema funcionará con una base de datos postgresql.

NF5: El sistema será sencillo para el uso del usuario.

Diagrama de casos de uso

La descripción de la utilidad de los diagramas de caso de uso se explicó en la iteración anterior.

Diagrama General

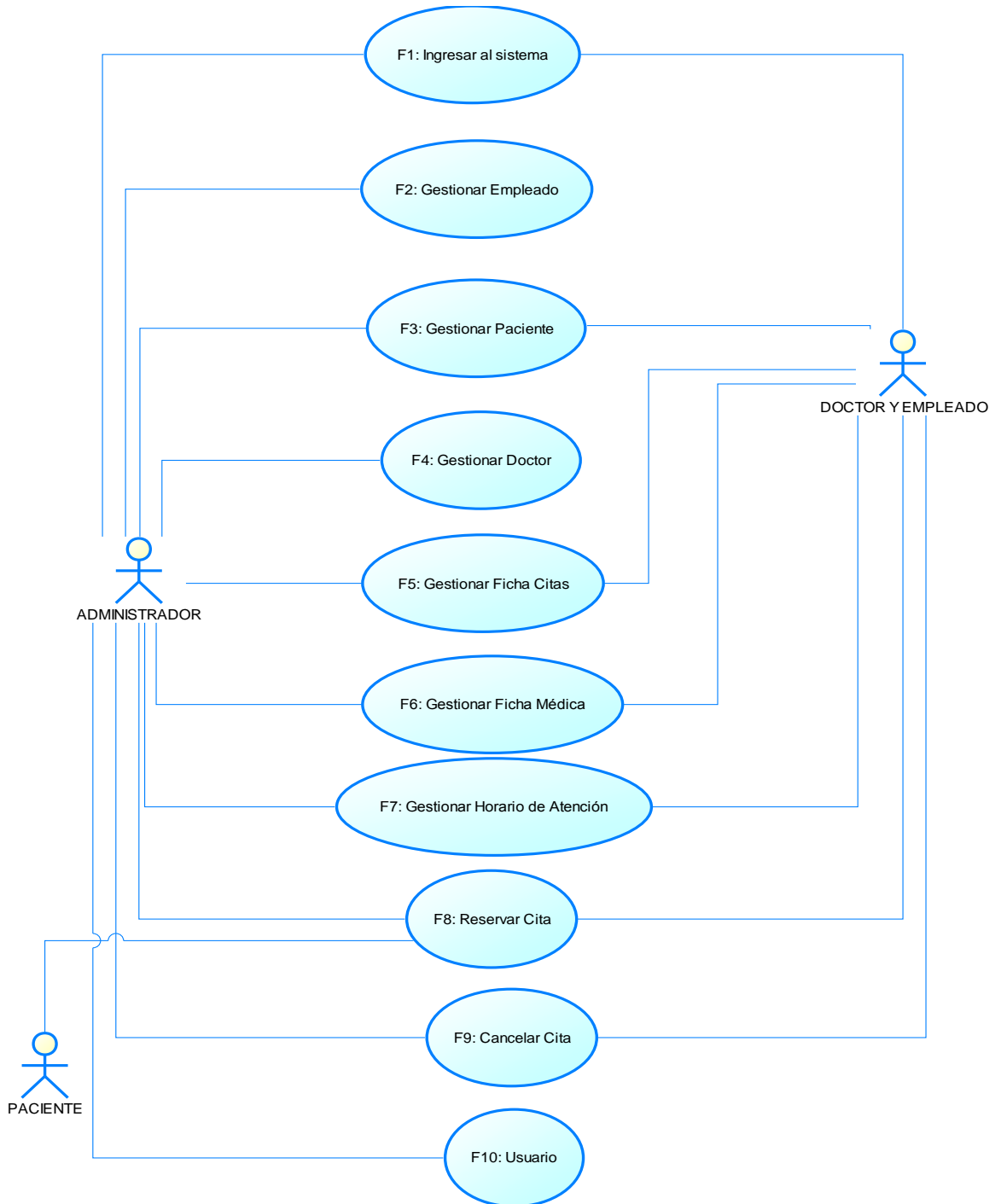


Diagrama 41 - Caso de uso General

Elaborado por: Nathaly Herrera

Diagrama Caso de Uso Ingreso al Sistema

F1: Ingreso al Sistema

Descripción: El sistema permitirá al actor escribir el usuario y contraseña para el ingreso al sistema.

Actores: Administrador, Empleado y Doctor

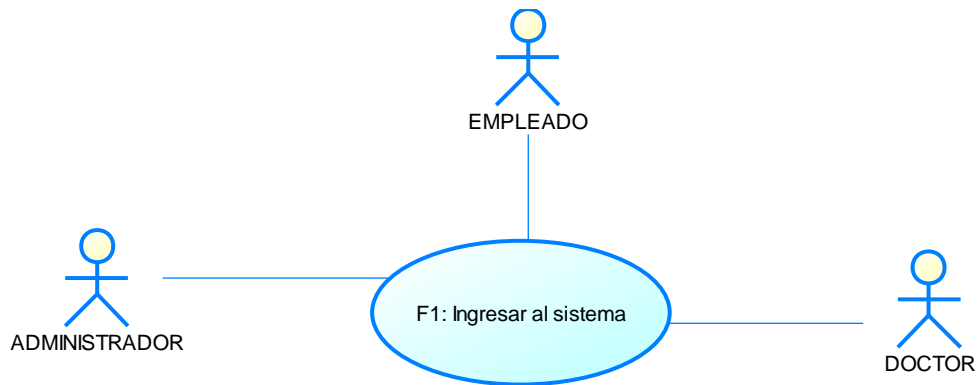


Diagrama 42 - Caso de Uso Ingreso al Sistema

Elaborado por: Nathaly Herrera

Flujo Principal:

1. El actor ingresa el usuario y contraseña.
2. El actor presiona el botón iniciar sesión.
3. El sistema verifica el usuario y contraseña. (E1) (E2)
4. Se ingresa al sistema.

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

E2: El usuario y contraseña son incorrectos, ingresar nuevamente.

Diagrama caso de uso Gestionar Ficha de Citas

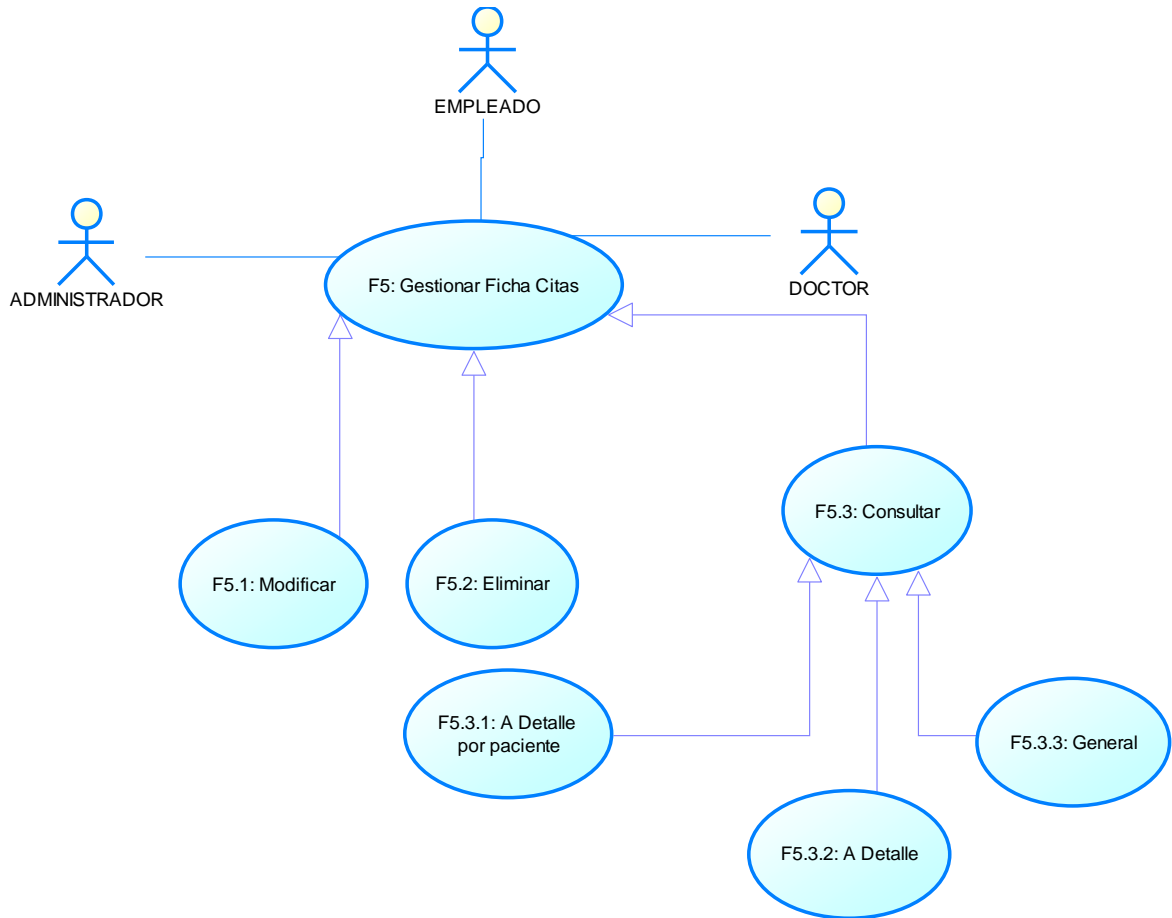


Diagrama 43 - Caso de uso Gestionar Ficha de Citas

Elaborado por: Nathaly Herrera

Detalle del caso de uso de Gestionar Ficha de Citas:

F5.1: Modificar Ficha de Citas

Descripción: El sistema permitirá al actor modificar los datos de cada cita del paciente del Centro Odontológico especificando la cédula del paciente.

Actor: Administrador, Empleado y Doctor

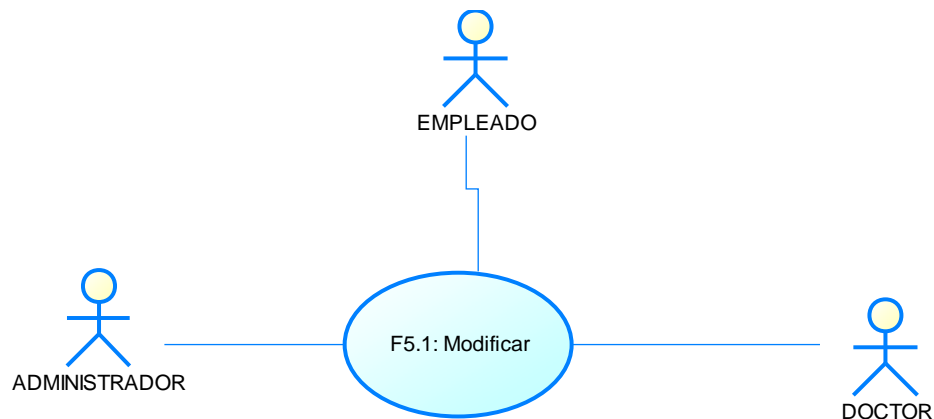


Diagrama 44 - Caso de Uso Modificar Ficha de Citas

Elaborado por: Nathaly Herrera

Flujo Principal:

1. El actor selecciona administrar ficha de la cita en el menú.
2. El sistema muestra la ventana administrar ficha de citas.
3. El actor ingresa la cédula del paciente.
4. El actor presiona el botón buscar.
5. El sistema verifica la cédula. (E1)
6. El sistema muestra todas las citas del paciente.
7. El actor selecciona la cita a modificar.
8. El sistema le muestra a más detalle los datos de la ficha de la cita especificada.
9. El actor modifica los campos.
10. El actor presiona el botón modificar.
11. El sistema verifica reglas de edición. (E2)
12. El sistema modifica los datos. (E1)

Flujo alternativo

- 5) Si la cédula del paciente no existe, ir al Caso de uso F3.1 (ingresar)

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

E2: No se puede modificar, la solución es volver a ingresar los datos.

F5.2: Eliminar Ficha de la Citas

Descripción: El sistema permitirá al actor eliminar la ficha de la cita de un determinado paciente del Centro Odontológico especificando su cédula del paciente.

Actor: Administrador, Empleado y Doctor

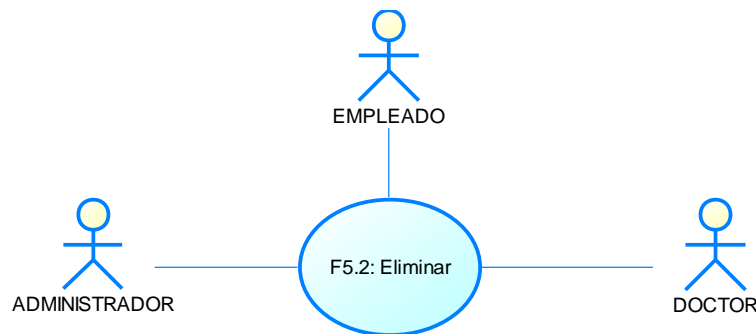


Diagrama 45 - Caso de Uso Eliminar Ficha de Citas

Elaborado por: Nathaly Herrera

Flujo Principal

1. El actor selecciona administrar ficha de la cita en el menú.
2. El sistema muestra la ventana de administrar ficha de la cita.
3. El actor ingresa la cédula del paciente.
4. El actor presiona el botón buscar.
5. El sistema verifica la cédula. (E1)
6. El sistema muestra los datos del paciente con todas sus citas.

7. El actor selecciona la cita.
8. El actor presiona el botón eliminar.
9. El sistema elimina los datos. (E1)

Flujo alterno

5) Si la cédula del paciente no existe, ir al Caso de uso F3.1 (ingresar)

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

F5.4.1: Consulta a Detalle por Paciente de la Ficha de Citas

Descripción: El sistema permitirá al actor consultar los datos de la cita especificando la cédula del paciente del Centro Odontológico.

Actor: Administrador, Doctor y Empleado

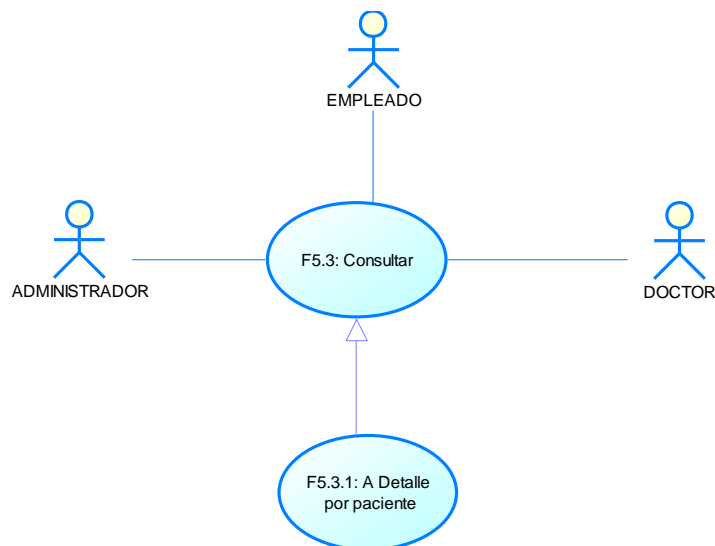


Diagrama 46 - Caso de Uso Consulta a Detalle por Paciente la Ficha de Citas

Elaborado por: Nathaly Herrera

Flujo Principal

1. El actor selecciona Administrar Ficha de la cita en el menú.
2. El sistema muestra la ventana de administrar ficha de la cita.
3. El actor ingresa la cédula del paciente.
4. El actor selecciona el botón buscar.
5. El sistema verifica la cédula. (E1)
6. El sistema muestra todas las citas del paciente. (E1)

Flujo alterno

- 5) Si la cédula paciente no existe, ir al Caso de uso F3.1 (ingresar)

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

F5.4.2: Consulta a Detalle de la Ficha de Citas

Descripción: El sistema permitirá al actor consultar los datos de la cita especificando la cédula del paciente, la cita o el nombre del doctor del Centro Odontológico.

Actor: Administrador, Doctor y Empleado

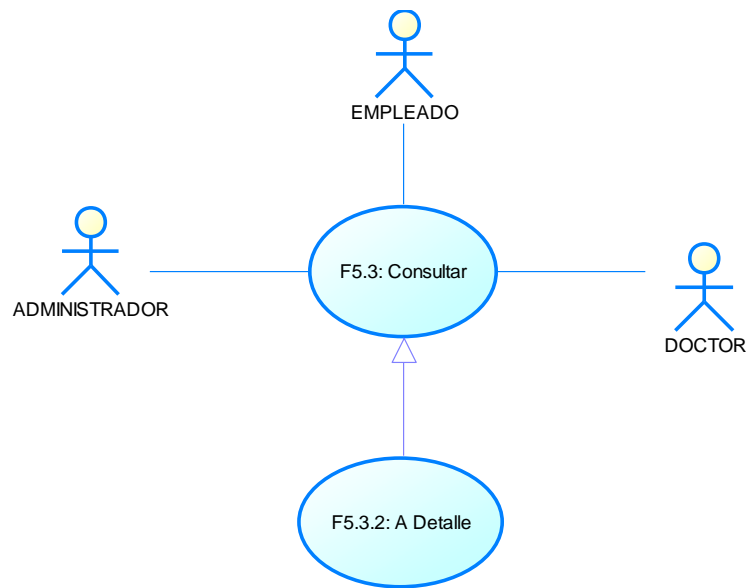


Diagrama 47 - Caso de Uso Consulta a Detalle Ficha de Citas

Elaborado por: Nathaly Herrera

Flujo Principal

1. El actor selecciona Administrar Ficha de la cita en el menú.
2. El sistema muestra la ventana de administrar ficha de la cita.
3. El actor ingresa la cédula del paciente o el código de la cita o el nombre del doctor.
4. El sistema verifica. (E1)
5. El sistema filtra en la tabla con las citas de paciente que cumplen el filtro de búsqueda.
6. El actor selecciona la cita.
7. El sistema muestra los datos de la cita. (E1)

Flujo alternativo

- 4) Si la cédula paciente no existe, ir al Caso de uso F3.1 (ingresar)

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

F5.4.3: Consulta General de la Ficha de Citas

Descripción: El sistema permitirá al actor consultar los datos de todas las citas de un determinado paciente Centro Odontológico.

Actor: Administrador, Empleado y Doctor

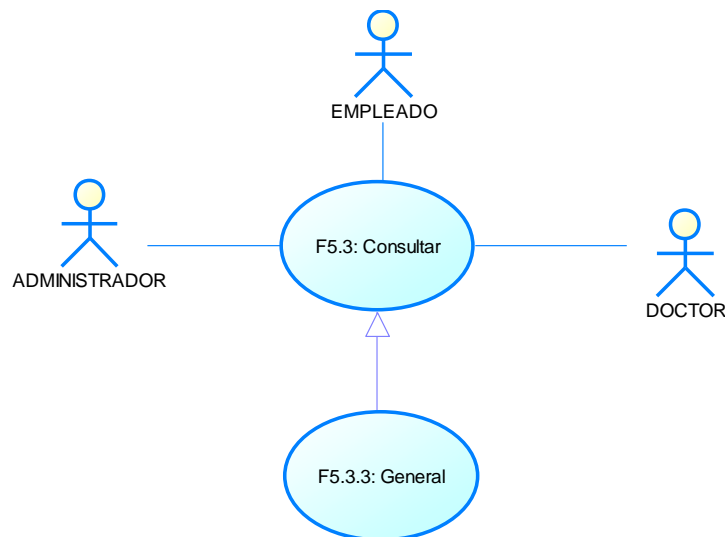


Diagrama 48 - Caso de Uso Consulta General Ficha de Citas

Elaborado por: Nathaly Herrera

Flujo Principal

1. El actor selecciona administrar ficha de citas en el menú.
2. El sistema muestra la ventana de administrar ficha de citas.
3. El sistema muestra todas las citas de cada paciente.

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

Diagrama Caso de Uso Reservar Cita

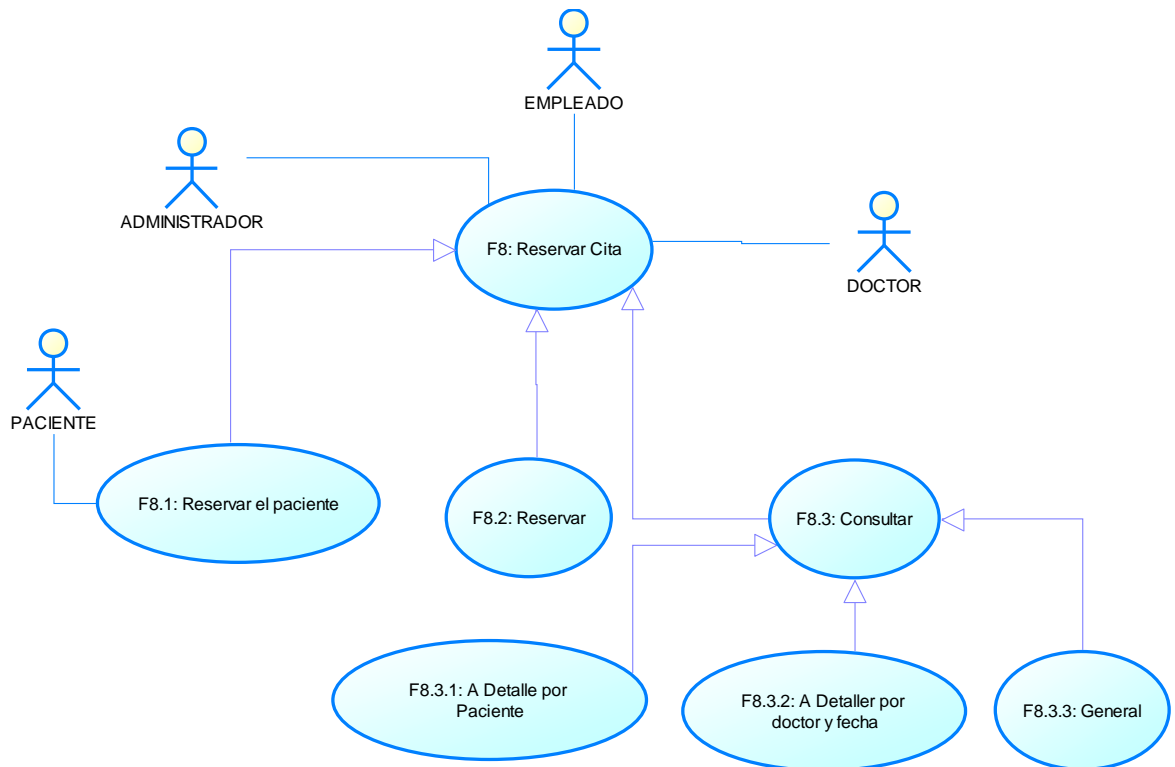


Diagrama 49 - Caso de Uso Reservar Cita

Elaborado por: Nathaly Herrera

Detalle del caso de uso Reservar Cita:

F8.1: Ingresar Reserva de Cita el paciente

Descripción: El sistema permitirá al actor reservar la cita, ingresando la cédula del paciente.

Actor: Administrador, Empleado, Doctor y Paciente

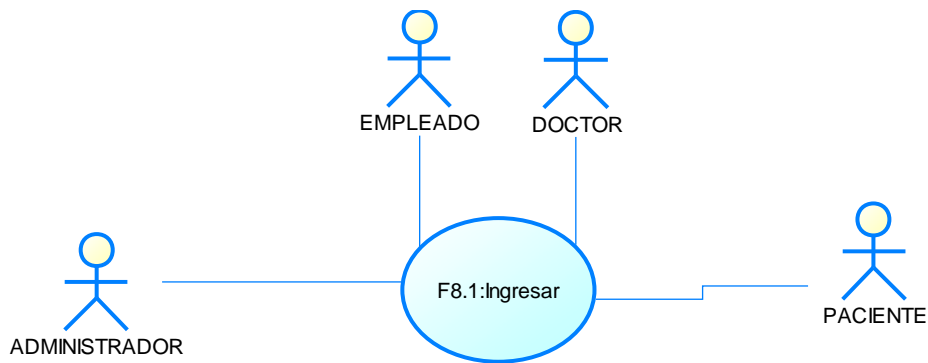


Diagrama 50 - Caso de Uso Ingresar Reserva por el Paciente

Elaborado por: Nathaly Herrera

Flujo Principal:

1. El actor Ingresar a la página de reservas.
2. El actor ingresa la cédula del paciente.
3. El actor presiona el botón buscar.
4. El sistema verifica la cédula. (E1)
5. El actor escoge la especialidad y fecha de la cita.
6. El actor presiona el botón buscar cita.
7. El sistema verifica. (E1)
8. El sistema muestra las citas disponibles.
9. El actor escoge la cita.
10. El actor presiona el botón reservar.
11. El sistema verifica cita. (E2)
12. El sistema reserva la cita

Flujo alternativo

- 6) Si la cédula del paciente no existe, ir al Caso de uso F3.2 (Ingresar).

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

E2: No se puede ingresar la reserva si el paciente tiene otra cita el mismo día a la misma hora, la solución es escoger otra hora o fecha.

F8.2: Ingresar Reserva

Descripción: El sistema permitirá al actor reservar la cita del Centro Odontológico Dental Access

Actor: Administrador, Empleado y Doctor.

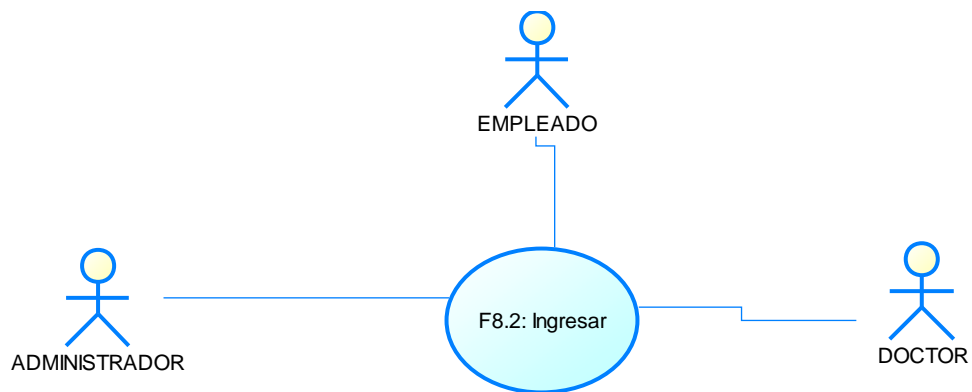


Diagrama 51 - Caso de Uso Reservar Cita

Elaborado por: Nathaly Herrera

Flujo Principal:

1. El actor selecciona en el menú Reservar Cita
2. El sistema muestra la ventana de reserva de citas.
3. El actor presiona el botón reservar cita
4. El sistema muestra la ventana de ingreso.

5. El actor escoge la cédula del paciente, el empleado, la especialidad y fecha de la cita.
6. El actor presiona el botón buscar cita.
7. El sistema verifica. (E1)
8. El sistema muestra las citas disponibles.
9. El actor escoge la cita.
10. El actor presiona el botón reservar.
11. El sistema verifica cita con paciente. (E2)
12. El sistema reserva la cita

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

E2: No se puede ingresar la reserva si el paciente tiene otra cita el mismo día a la misma hora, la solución es escoger otra hora o fecha.

F8.3.1: Consultar A Detalle Reserva de Cita por Paciente

Descripción: El sistema permitirá al actor consultar la reservar de la cita, especificando el número de cédula del paciente.

Actor: Administrador, Empleado y Doctor.

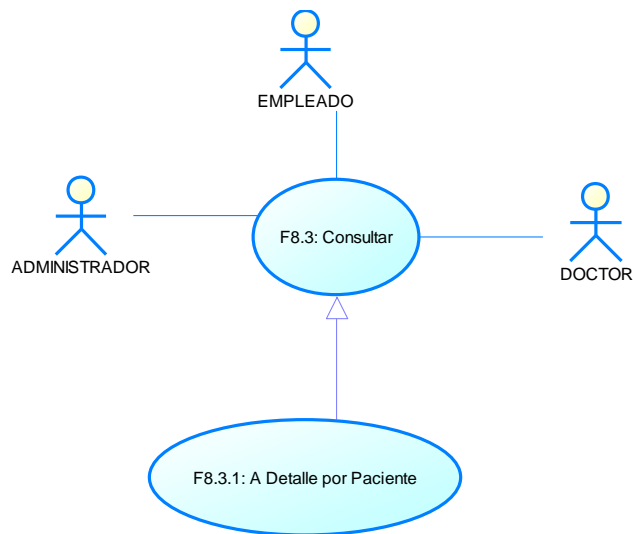


Diagrama 52 - Caso de Uso Consulta A Detalle Reserva de Cita por Paciente

Elaborado por: Nathaly Herrera

Flujo Principal:

1. El actor selecciona en el menú Reservar Cita
2. El sistema muestra la ventana reservar cita
3. El actor ingresa la cédula del paciente.
4. El actor presiona el botón buscar.
5. El sistema verifica la cédula. (E1)
6. El sistema muestra los datos de las reservas del paciente.

Flujo alterno

- 6) Si la cédula del paciente no existe, ir al Caso de uso F3.2 (Ingresar).

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

F8.3.2: Consultar A Detalle Reserva de Cita por Doctor y Fecha

Descripción: El sistema permitirá al actor consultar la reservar de la cita, especificando el número de cédula del doctor y especificando el rango de fecha de la consulta

Actor: Administrador, Empleado y Doctor.

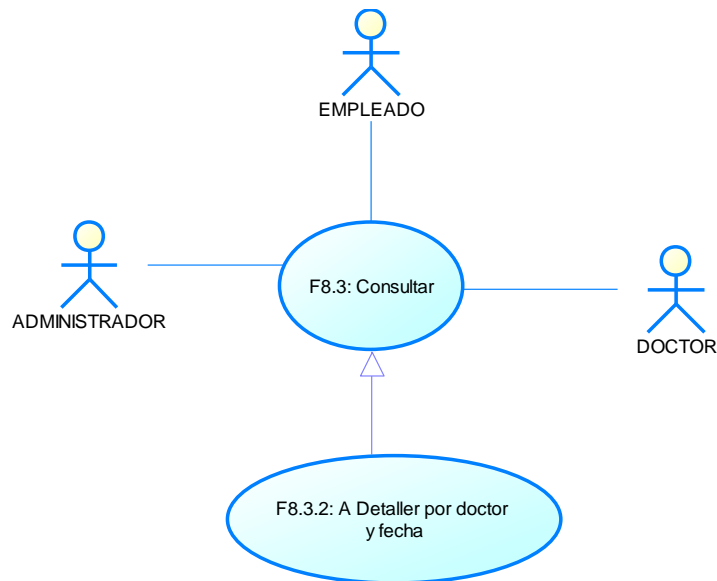


Diagrama 53 - Caso de Uso Consultar Reserva de Cita por Doctor y Fecha

Elaborado por: Nathaly Herrera

Flujo Principal:

1. El actor selecciona en el menú Reservar Cita
2. El sistema muestra la ventana reservar cita
3. El actor escoge la cédula del doctor y el rango de fecha.
4. El actor presiona el botón buscar.
5. El sistema verifica. (E1)
6. El sistema muestra los datos de las reservas.

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

F8.3.3: Consultar Genera de Reserva de Cita

Descripción: El sistema permitirá al actor consultar todas las reservas de citas.

Actor: Administrador, Empleado y Doctor.

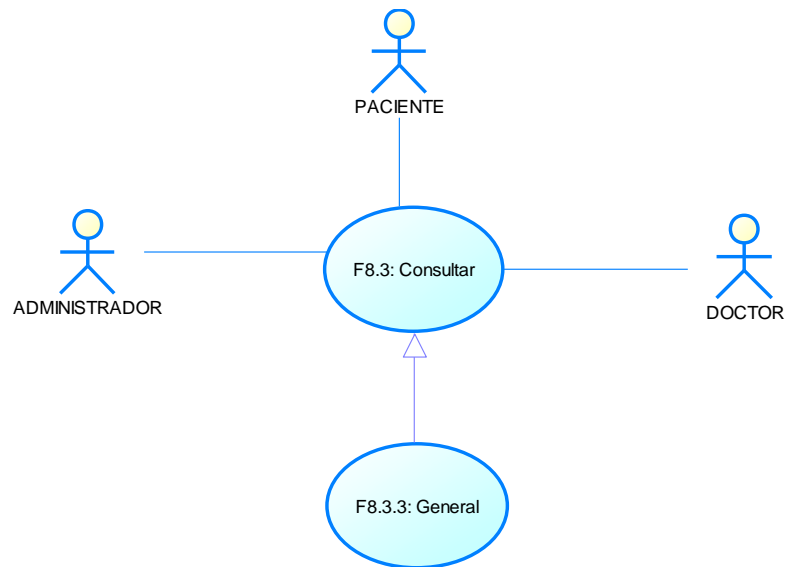


Diagrama 54 - Caso de Uso Consulta General Reservar Cita

Elaborar por: Nathaly Herrera

Flujo Principal:

1. El actor selecciona en el menú Reservar Cita
2. El sistema muestra la ventana reservar cita
3. El sistema muestra todas las reservas.

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

F9: Cancelación de Cita

Descripción: El sistema permitirá al actor cancelar las reservas de las citas.

Actor: Administrador, Empleado y Doctor.

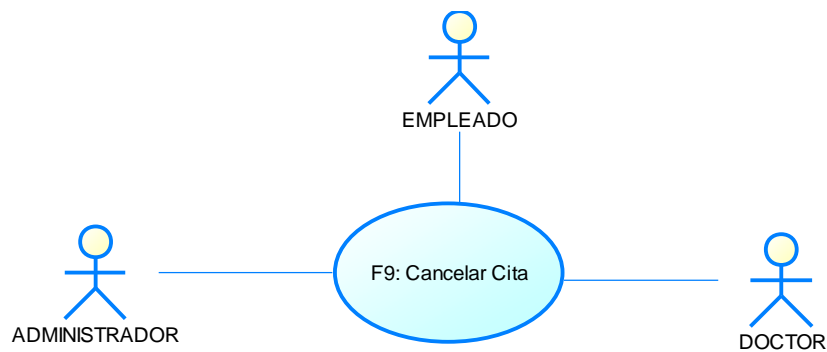


Diagrama 55 - Caso de Uso Cancelación de Cita

Elaborado por: Nathaly Herrera

Flujo Principal:

1. El actor selecciona en el menú Reservar Cita
2. El sistema muestra la ventana reservar cita
3. El actor ingresa la cédula del paciente.
4. El actor presiona el botón buscar.
5. El sistema verifica la cédula. (E1)
6. El sistema muestra los datos de las reservas del paciente.

7. El actor Presiona el botón cancelar
8. El sistema cancela la cita.

Flujo alterno

- 6) Si la cédula del paciente no existe, ir al Caso de uso F3.2 (Ingresar).

Excepciones

E1: Problema con la base de datos, la solución es llamar al administrador de la base de datos (DBA).

Diagrama de Clase

La descripción del diagrama de clase se explicó en la iteración anterior.

Prototipo

Ventana Principal Reserva de Citas

Logo Inicio Usuario Empleado Doctor Paciente Horario de Atención Ficha Médica Ficha de la Cita Reservar Cita Correr Sesión

Reserva de Citas

Ingrese la cédula del paciente para buscar sus reservas

Cédula: [Buscar por Paciente](#)

Escoja la cédula del doctores y un rango de fecha

Cédula:

Fecha: desde hasta

[Buscar](#)

[Reservar Cita](#)

Mostrar 10 Registros

Id	Cédula del Paciente	Nombre del Paciente	Nombre del Doctor	Especialidad	Fecha de la Cita	Hora empieza la cita	Hora termina la cita	Tipo	Estado
----	---------------------	---------------------	-------------------	--------------	------------------	----------------------	----------------------	------	--------

Ilustración 38 - Interfaz Ventana Principal Reserva de Citas

Elaborado por: Nathaly Herrera

Ventana Reservar Cita

Reservar Cita

Cédula del Empleado:

Cédula del Paciente:

Especialidad:

Fecha de la Cita:

[Buscar Cita](#)

[Regresar a la lista de reservas](#)

Ilustración 39 - Interfaz Ventana Reservar Cita

Elaborado por: Nathaly Herrera

Ventana Reservar Cita (Lado del paciente)

Ilustración 40 - Interfaz Ventana Reservar Cita (Lado del Paciente)

Elaborado por: Nathaly Herrera

Ventana Mostrar Reserva de Citas por Paciente

Id	Nombre del Doctor	Especialidad	Fecha de la Cita	Hora empieza la cita	Hora termina la cita	Tipo	Estado	Cancelar
3	Zoe Martinez	Rehabilitación Oral	2016-05-18	09:00	09:30	Consulta	EN CURSO	Cancelar
1	Luis Hernandez	Endodoncia	2016-05-18	11:00	12:40	Cirugia	EN CURSO	Cancelar

Ilustración 41 - Interfaz Ventana Mostrar Reserva de citas por Paciente

Elaborado por: Nathaly Herrera

3.2.2. Fase de Elaboración

3.2.2.1. Análisis y Diseño

Diagramas de Secuencia

En la realización de este proyecto se utilizará el framework Symfony por lo que consta de varias funciones navitas para su funcionamiento correctos, que se ven reflejadas en los diagramas de secuencia como son: find(), findOneByPcedula() entre otras.

Los diagramas de secuencia que se presentan a continuación fueron elaborados por Nathaly Herrera, la autora de la presente disertación.

Diagrama de Secuencia Ingreso al sistema

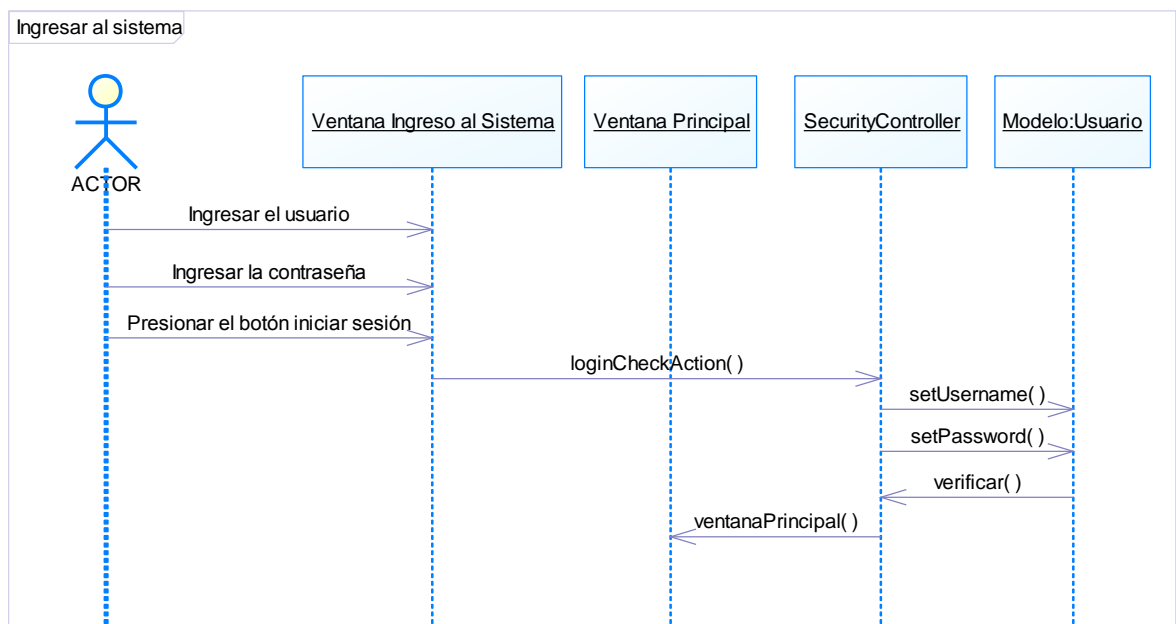


Diagrama 57 - Diagrama de Secuencia Ingreso al Sistema

Diagrama de Secuencia Gestionar Ficha de Cita

Modificar Ficha de Cita

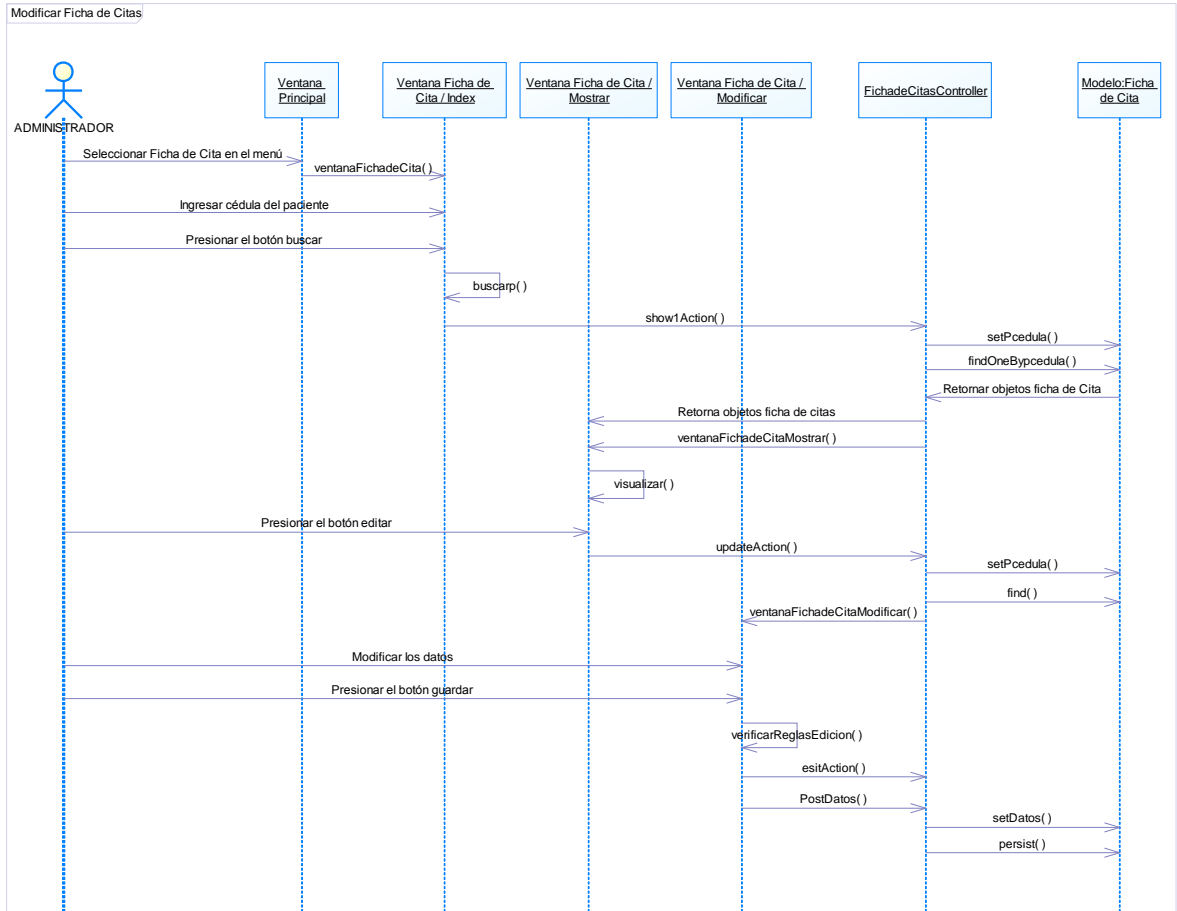


Diagrama 58 - Diagrama de Secuencia Modificar Ficha de Cita

Eliminar Ficha de Cita

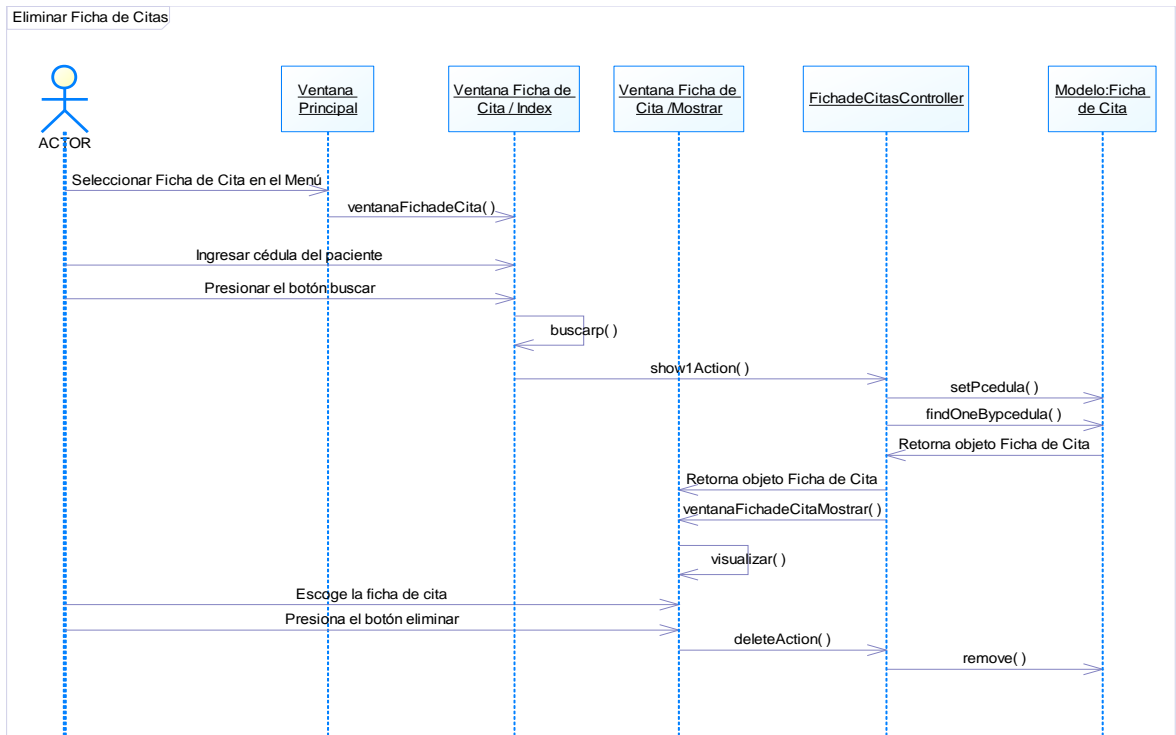


Diagrama 59 - Diagrama de Secuencia Eliminar Ficha de Cita

Consultar a Detalle la Ficha de Cita por Paciente

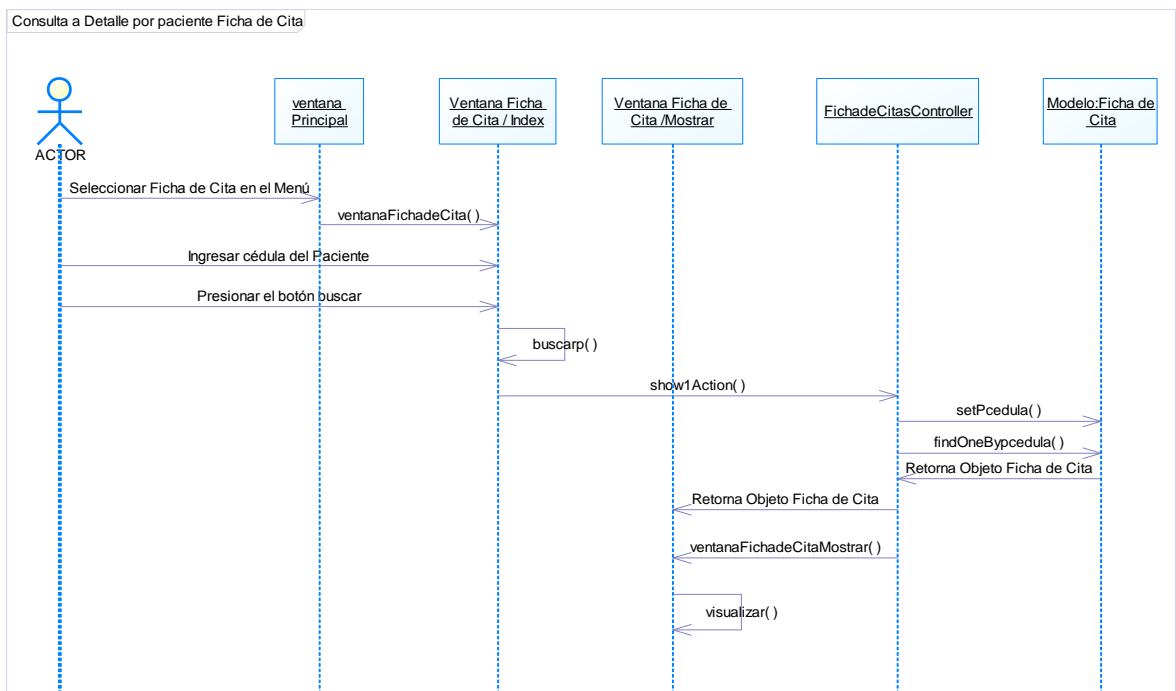


Diagrama 60 - Diagrama de Secuencia Consulta a Detalle Ficha de Cita por Paciente

Consulta a Detalle Ficha de Cita

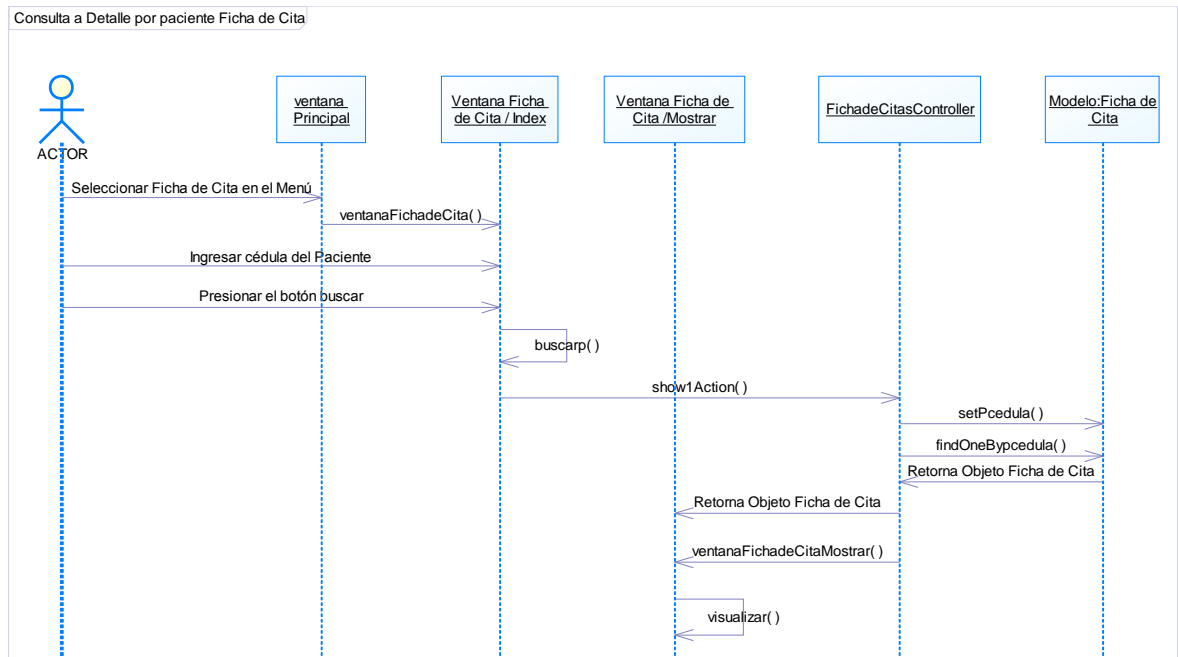


Diagrama 61 - Diagrama de Secuencia Consulta a Detalle Ficha de Cita

Consulta General Ficha de Cita

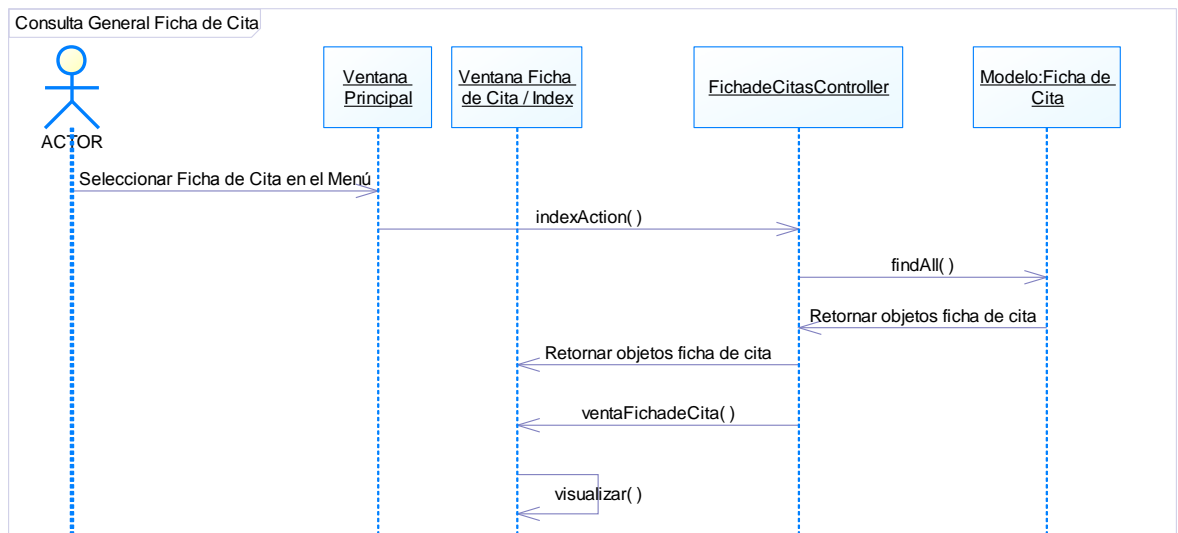


Diagrama 62 - Diagrama de Secuencia Consulta General Ficha de Cita

Diagrama de Secuencia Reservar Cita

Ingresar Reserva de Cita por parte del Paciente

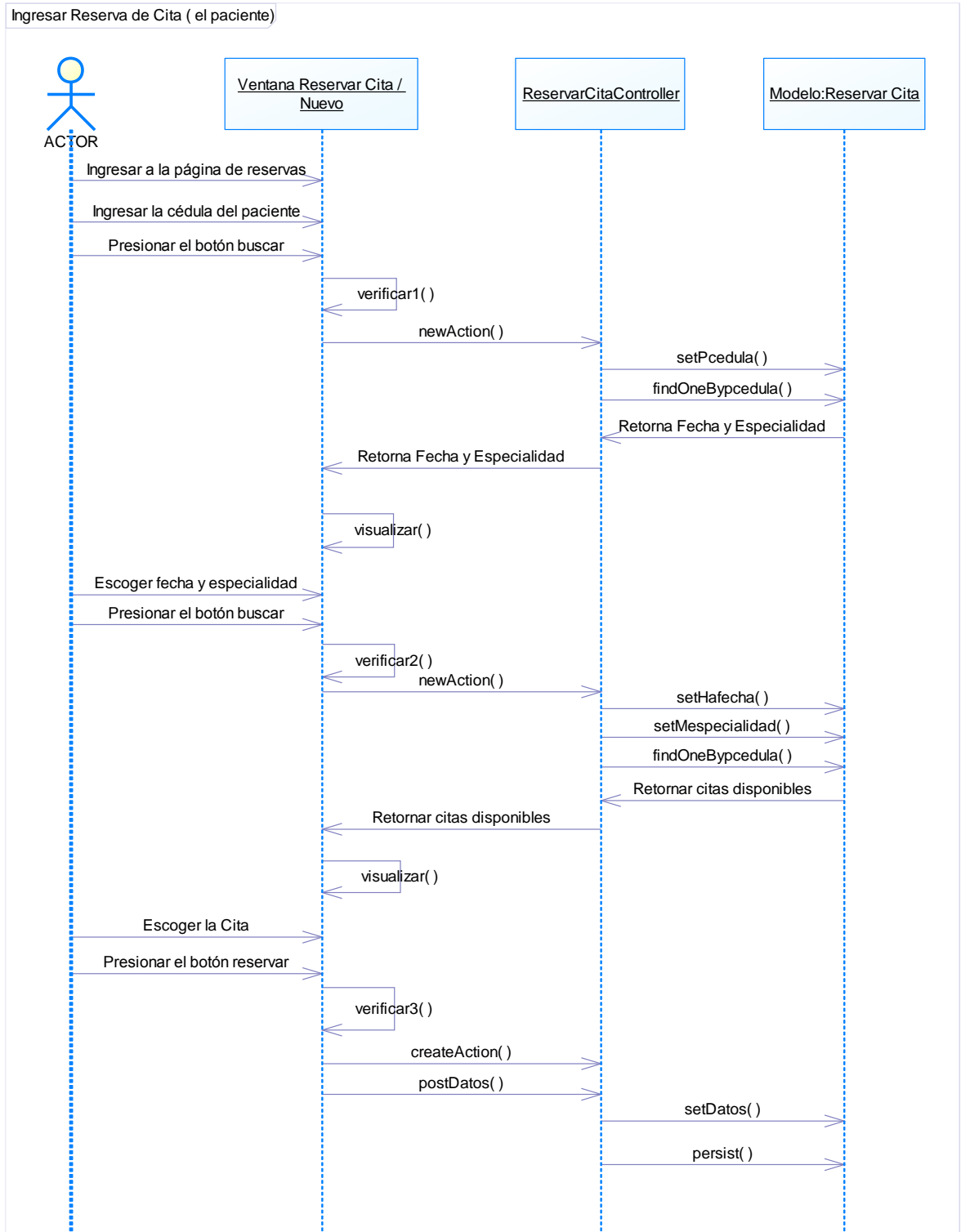


Diagrama 63 - Diagrama de Secuencia Ingresar Reserva de Cita (Paciente)

Ingresar Reserva de Cita

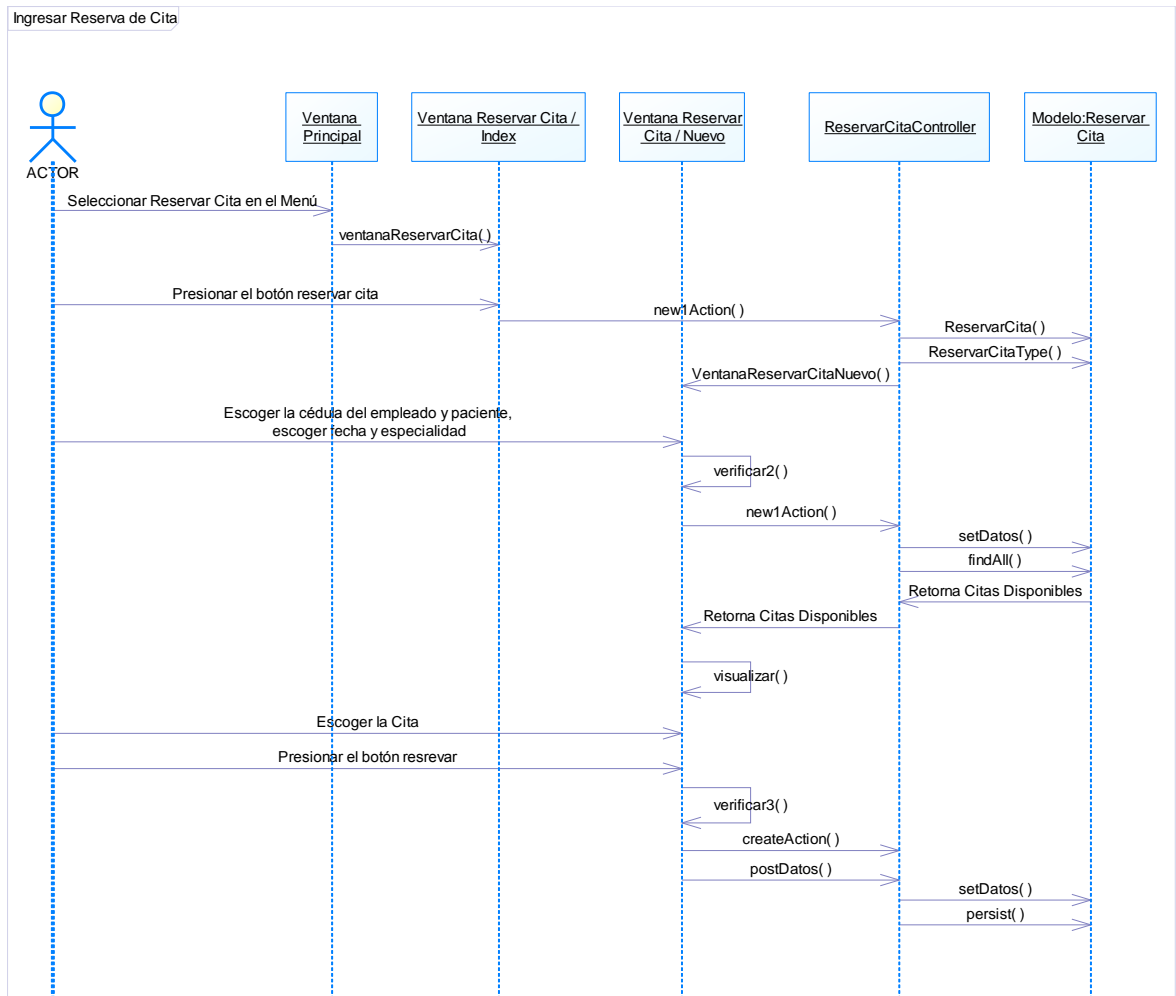


Diagrama 64 - Diagrama de Secuencia Ingresar Reserva de Cita

Consultar a Detalle Reserva de Cita por Paciente

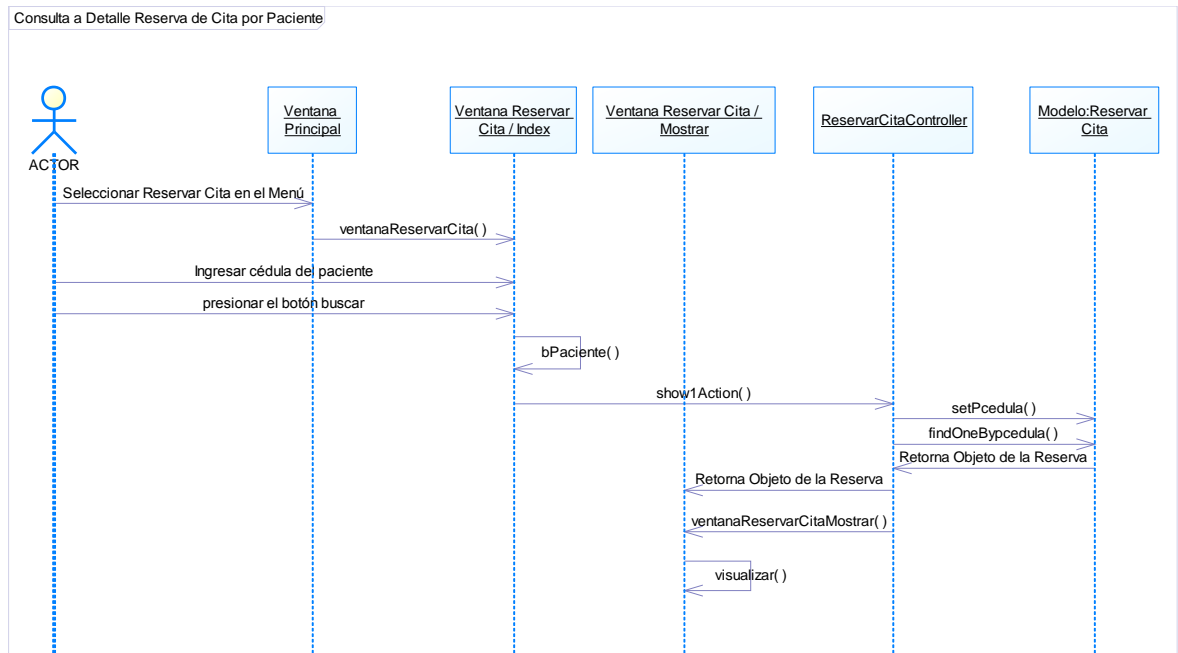


Diagrama 65 - Diagrama de Secuencia Consulta a Detalle Reserva de Cita por Paciente

Consulta a Detalle Reserva de Cita por Fecha y Doctor

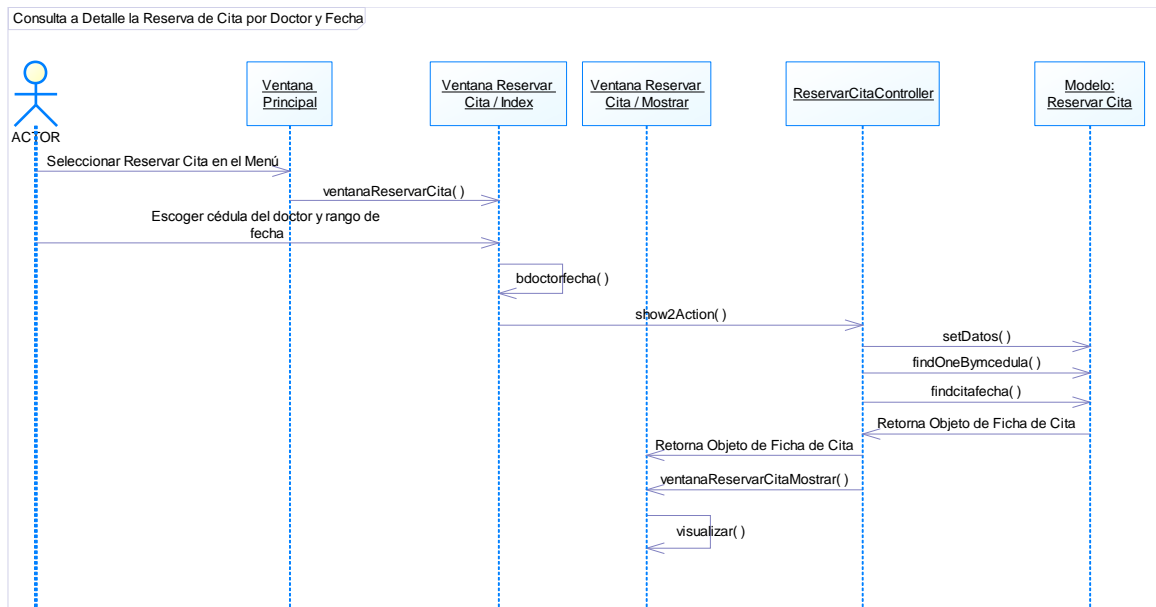


Diagrama 66 - Diagrama de Secuencia Consulta a Detalle Reserva de Cita por Fecha y Doctor

Consulta General de la Reserva de Citas

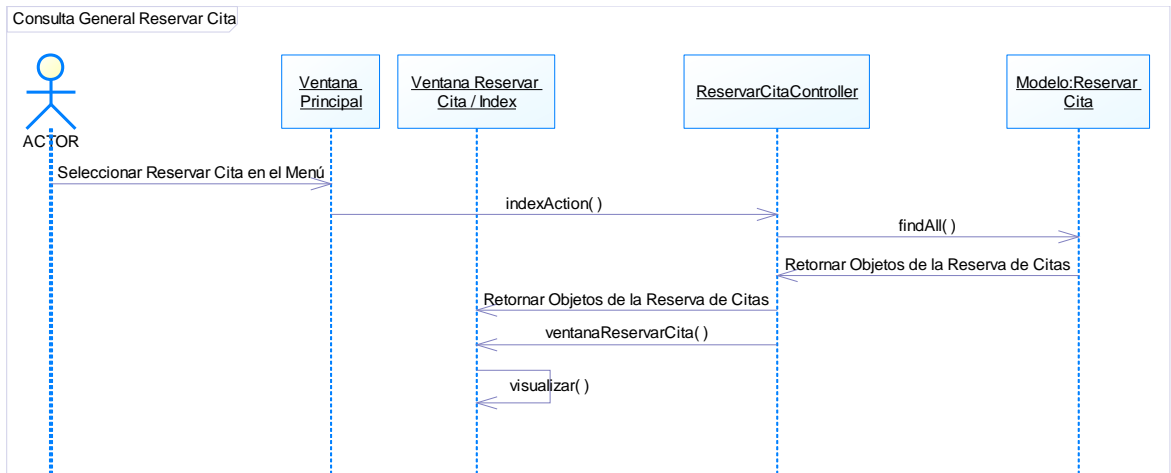


Diagrama 67 - Diagrama de Secuencia Consulta General Reserva de Citas

Diagrama de Secuencia Cancelar Reserva de Cita

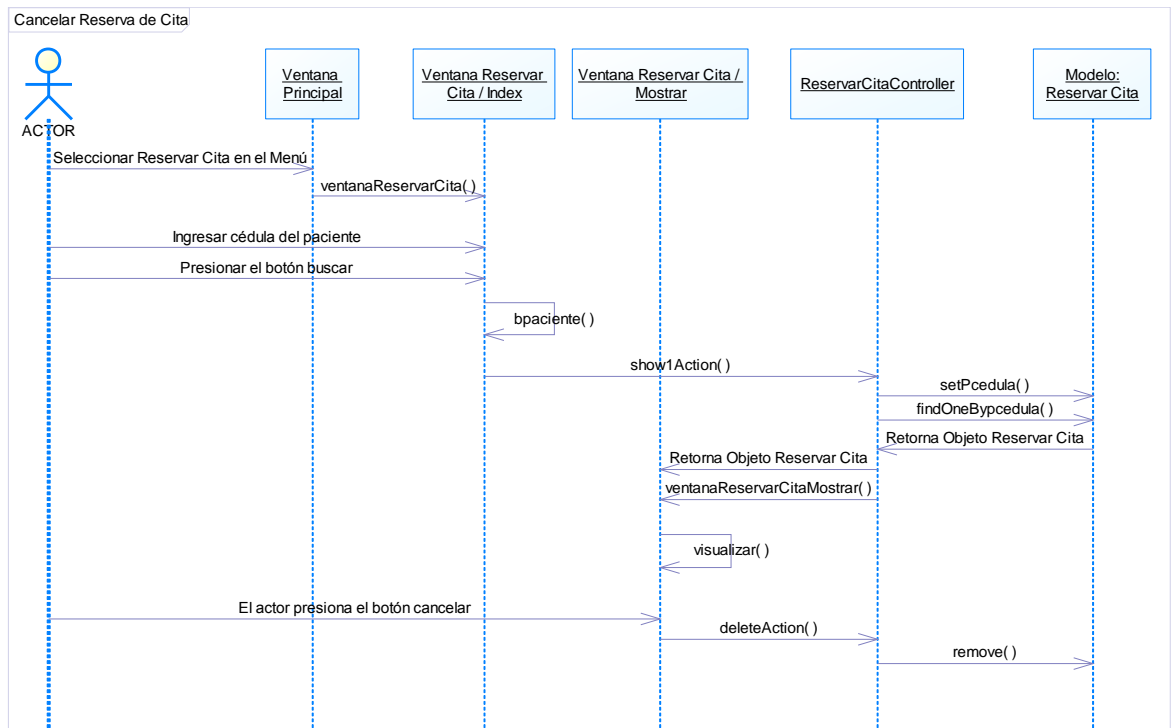


Diagrama 68 - Diagrama de Secuencia Cancelar Reserva de Cita

Elaborado por: Nathaly Herrera

Diagrama Físico

CONSTRUCCIÓN DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE CITAS, TRATAMIENTOS Y EXPEDIENTE MÉDICO DE LOS PACIENTES DEL CENTRO IMPLANTOLÓGICO Y DE ESPECIALIDADES DENTAL ACCESS

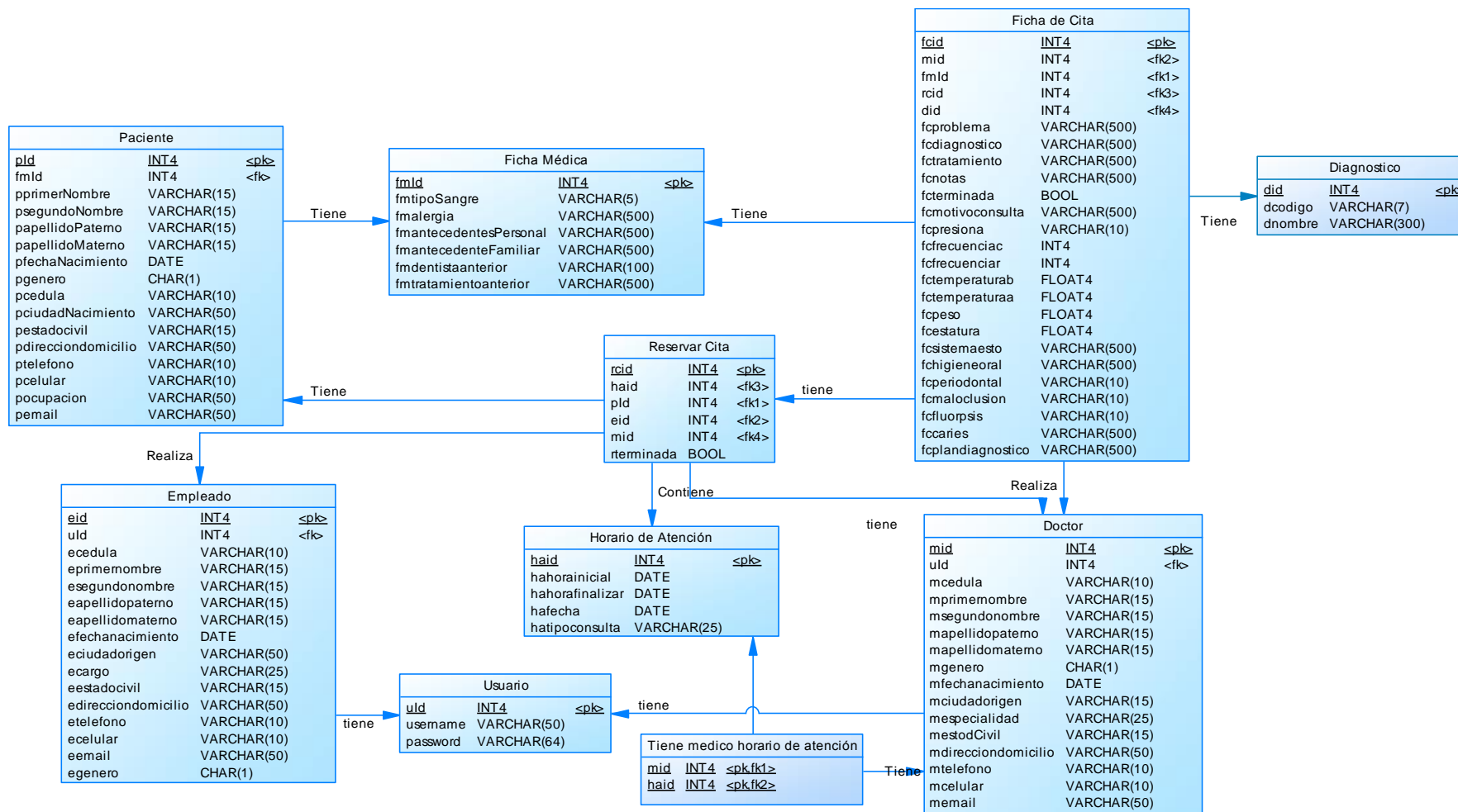


Diagrama 69 - Diagrama Físico Completo

Elaborado por: Nathaly Herrera

Diagrama de Actividades

El diagrama de actividades es el que indica el proceso o flujo de una actividad importante del sistema.

Reservar Cita

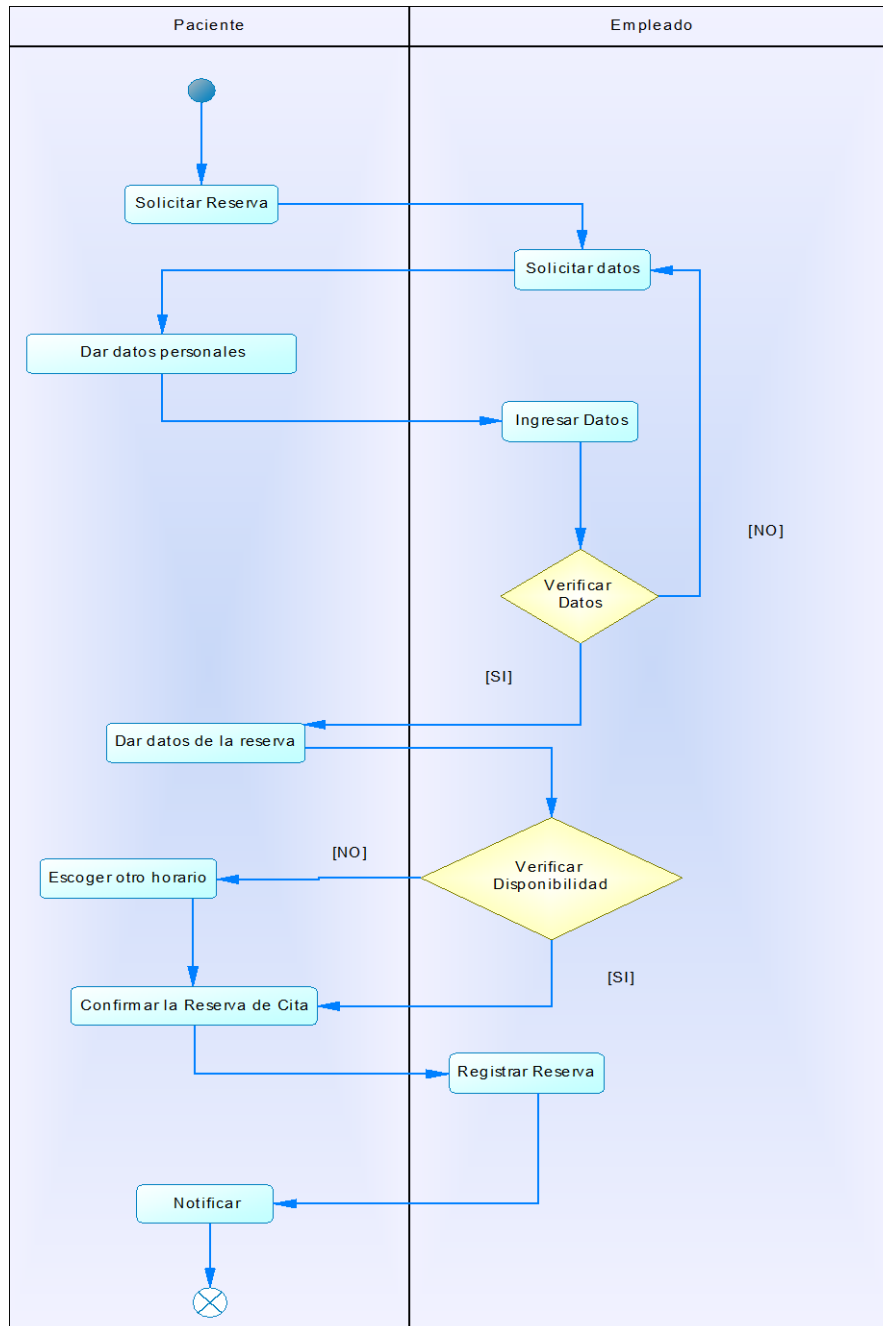


Diagrama 70 - Diagrama de Actividades Reservar Cita

Elaborado por: Nathaly Herrera

Cancelar reserva de Cita

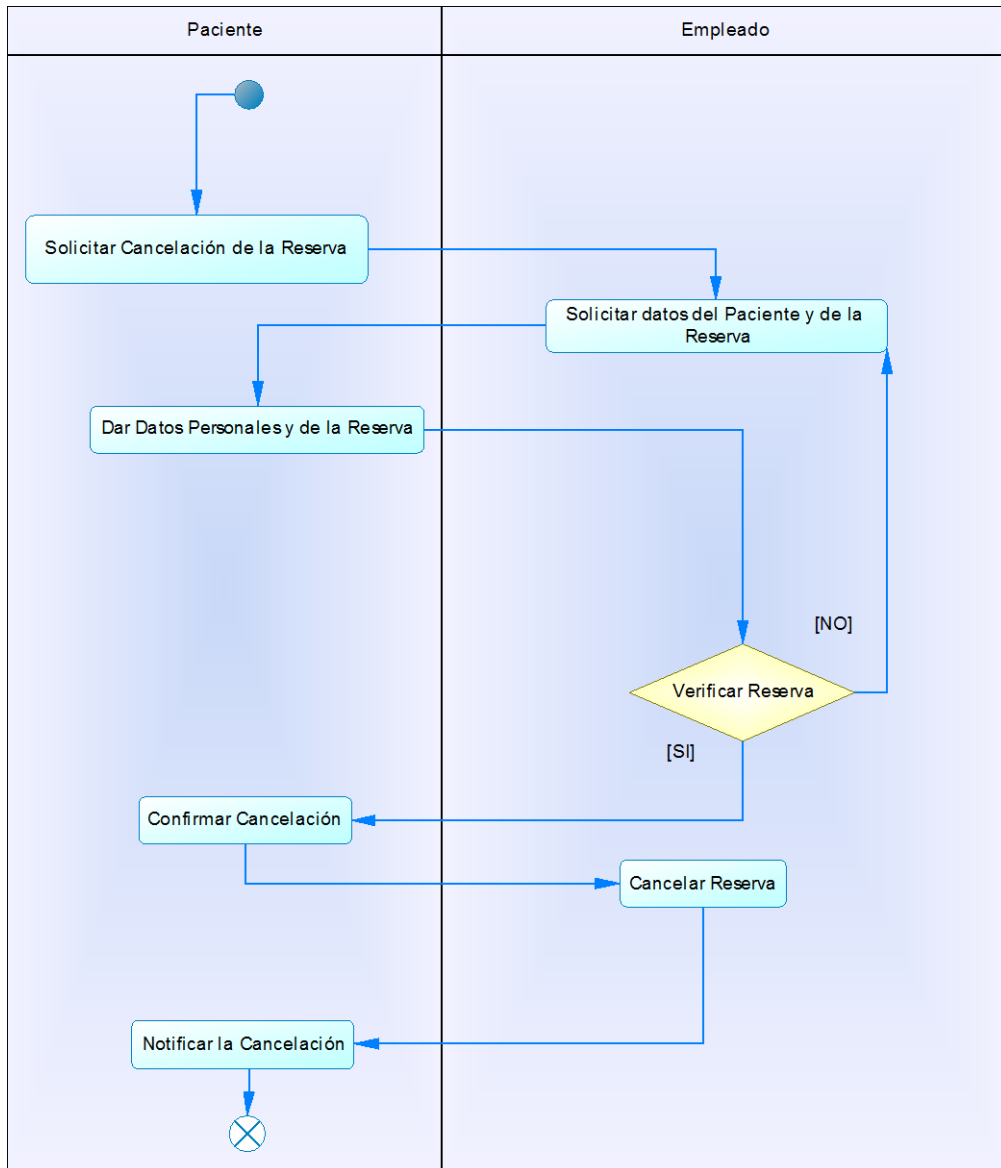


Diagrama 71 - Diagrama de Actividades Cancelar Reserva de Cita

Elaborado por: Nathaly Herrera

3.2.3. Fase de Construcción

3.2.3.1. Implementación

La implementación de la aplicación web se realiza con el framework Symfony que involucra php, css y twig.

El código de la aplicación se encuentra en el CD, en una carpeta llamada implementación.

Descripción

Ingresar al Sistema: Permite a los usuarios ingresar al sistema dependiendo de los 2 roles que existe, si es administrador o usuario.

Ficha de Cita: Esta clase contiene toda la información necesaria para cada cita médica del paciente y así tener un seguimiento de cada consulta o tratamiento realizado de cada paciente. Para su mejor visualización se puede observar la ficha acumulada en formato PDF de un determinado paciente. Para administrar esta clase de ficha de cita tiene permisos el administrador, empleado y doctor.

Usuario: Esta clase contiene el usuario y contraseña de cada doctor y médico, para el ingreso al sistema. Para administrar usuarios tiene permisos el administrador. Al momento de crear el usuario se le notifica al empleado o doctor al email, su usuario y clave.

Reservar Cita: Es un proceso que permite al paciente reservar su cita dependiendo de la especialidad del doctor, fecha y hora. Para administrar este proceso tiene permisos el administrador, empleado, doctor y el paciente (solo para ingresar la reserva de cita).

Cancelar Reserva de Cita: Es un proceso que permite cancelar la reserva de la cita del paciente. Para administrar este proceso tiene permisos el administrador, empleado y doctor.

Interface

Interface del sistema (Ingreso al sistema)

Para ingresar al sistema se debe ingresar el usuario y contraseña ya sea administrador o usuario.

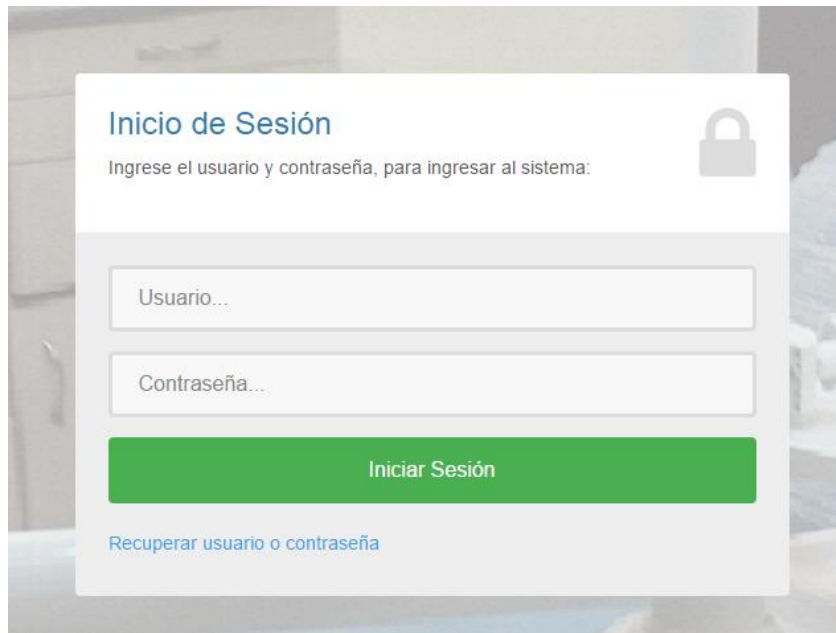


Ilustración 42 - Ingreso al sistema

Elaborado por: Nathaly Herrera

Si no se ingresa correctamente el usuario o contraseña le notifica al usuario.

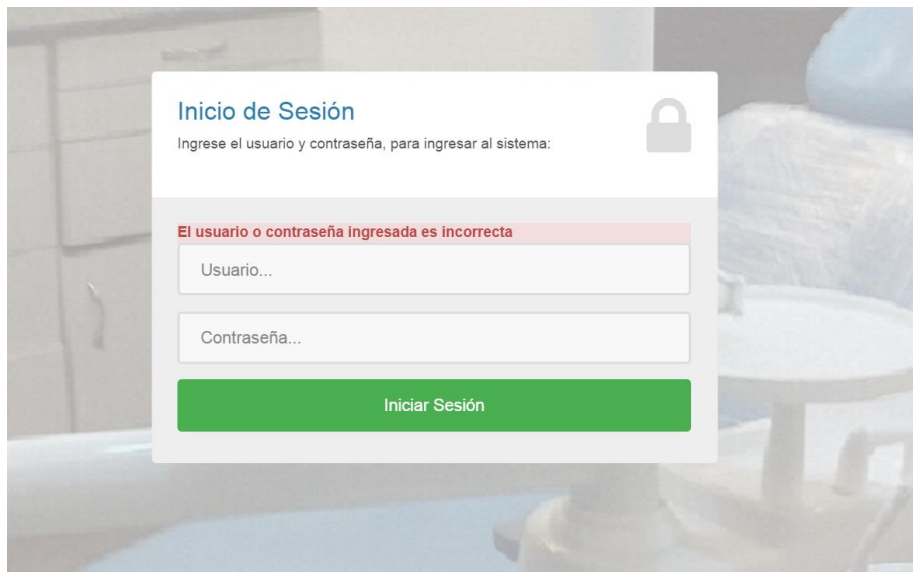


Ilustración 43 - Notificación de Error de Usuario o Contraseña

Elaborado por: Nathaly Herrera

Cuando ingresa al sistema, se re direcciona directamente a la página de inicio.



Ilustración 44 - Interface página de Inicio

Elaborado por: Nathaly Herrera

Notificación en el email el usuario y contraseña, al momento de asignar al empleado o doctor.

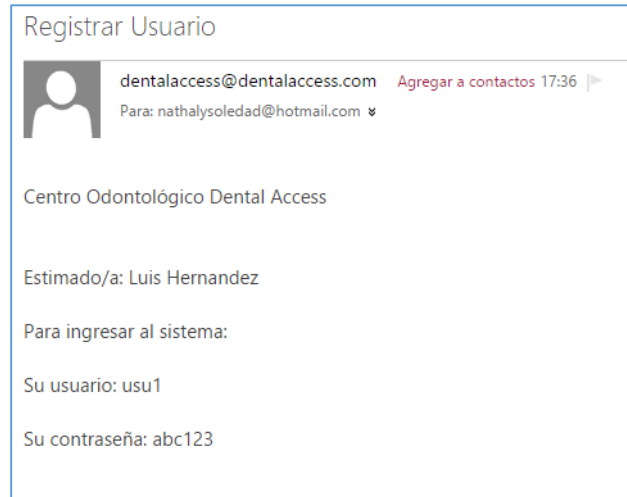


Ilustración 45 - Notificación Usuario y Contraseña Email

Elaborado por: Nathaly Herrera

Interface del sistema (Reservar Cita)

Página principal de reservar cita

Reserva de Citas

Ingrese la cédula del paciente para buscar sus reservas

Cédula: [Buscar por Paciente](#) 🔍

Escoja la cédula del doctores y un rango de fecha

Cédula:

Fecha: desde hasta

[Buscar](#) 🔍

[Reservar Cita](#) 📅

Mostrar Registros

Buscar:

Id	Cédula del Paciente	Nombre del Paciente	Nombre del Doctor	Especialidad	Fecha de la Cita	Hora empieza la cita	Hora termina la cita	Tipo	Estado
Ningún dato disponible en esta tabla									

Mostrando registros del 0 al 0 de un total de 0 registros

[Anterior](#) [Siguiente](#)

Ilustración 46 - Interface Página Principal de Reserva Cita

Elaborado por: Nathaly Herrera

Ingresar reserva de cita

Se ingresa la reserva de cita el paciente. El paciente busca por su cédula, para reservar cita, primero debe estar registrado en el sistema, los datos del paciente.

Doctor	Especialidad	Fecha	Hora empieza la cita	Hora termina la cita	tipo	Escoger
Luis Hernandez	Implantología	2016-06-01	10:00	10:30	Consulta	<input type="radio"/>
Luis Hernandez	Implantología	2016-06-01	09:00	09:45	Tratamiento	<input type="radio"/>
Karla Lazo	Implantología	2016-06-01	09:00	09:45	Tratamiento	<input type="radio"/>

Ilustración 47 - Interface Reservar Cita (el paciente)

Elaborado por: Nathaly Herrera

Se ingresa la reserva de cita los usuarios del Centro Odontológico

Doctor	Especialidad	Fecha	Hora empieza la cita	Hora termina la cita	tipo	Escoger
Sebastián Lema	Ortodoncia	2016-06-01	10:00	10:30	Consulta	<input type="radio"/>

Ilustración 48 - Interface Reservar Cita (Usuarios del Centro Odontológico)

Elaborado por: Nathaly Herrera

Si el paciente ya tiene una cita a la misma hora y el mismo día le notifica que no puede reservar otra cita.

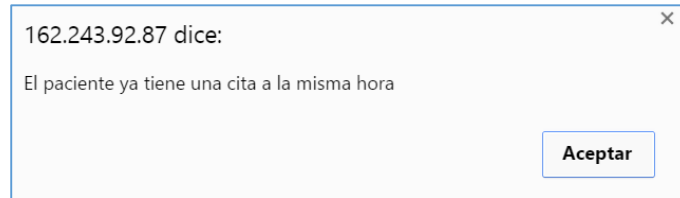


Ilustración 49 - Notificación al paciente de que ya tiene cita la misma fecha y hora

Elaborado por: Nathaly Herrera

Al momento de ingresar la reservar la cita, al paciente le llega un mail con los datos de la cita que se reservó.

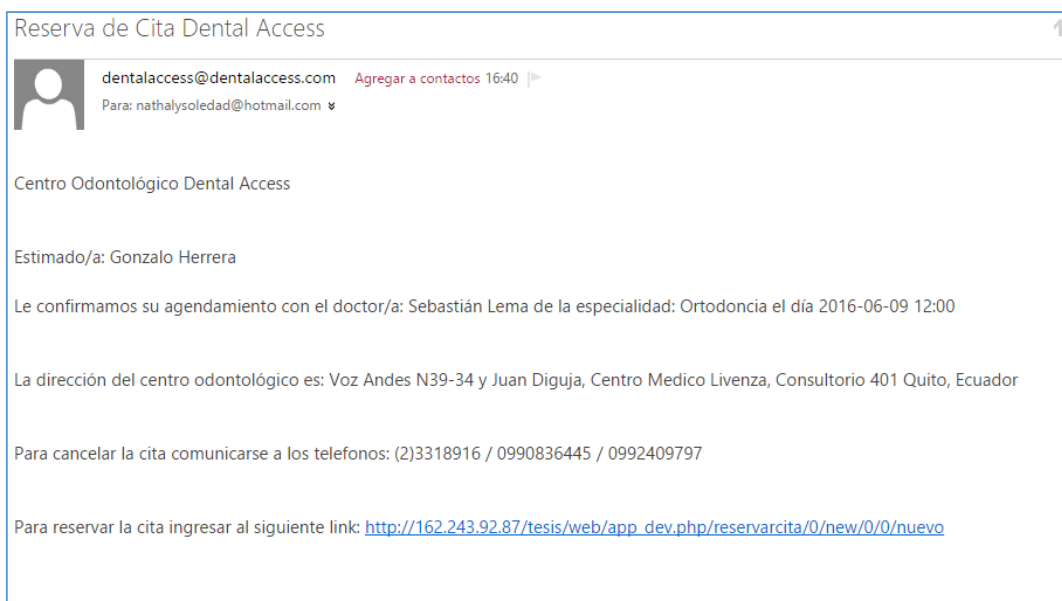


Ilustración 50 - Mail de Confirmación de la Reserva de Cita

Elaborado por: Nathaly Herrera

Consultar Reserva de Cita

Consultar todas las citas de un determinado paciente, se muestra en una tabla las citas que se encuentran en curso en cambio en la otra tabla se muestra las citas terminadas.

CONSTRUCCIÓN DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE CITAS, TRATAMIENTOS Y EXPEDIENTE MÉDICO DE LOS PACIENTES DEL CENTRO IMPLANTOLÓGICO Y DE ESPECIALIDADES DENTAL ACCESS

Reserva de cita

Datos del Paciente

Cédula	1704262193
Nombre	Gonzalo Herrera
Género	M
Fecha Nacimiento	1955-09-29

Lista de las reservas de citas del paciente en curso

Id	Nombre del Doctor	Especialidad	Fecha de la Cita	Hora empieza la cita	Hora termina la cita	Tipo	Estado
5	Luis Hernandez	Implantología	2016-06-01	09:00	09:45	Tratamiento	EN CURSO
1	Sebastián Lema	Ortodoncia	2016-06-01	10:00	10:30	Consulta	EN CURSO

Lista de las reservas de citas del paciente terminadas

Mostrar Registros Buscar:

Id	Nombre del Doctor	Especialidad	Fecha de la Cita	Hora empieza la cita	Hora termina la cita	Tipo	Estado
Ningún dato disponible en esta tabla							

Mostrando registros del 0 al 0 de un total de 0 registros

Ilustración 51 - Interface Consultar Reserva de Cita

Elaborado por: Nathaly Herrera

Consultar las reservas de citas especificando al doctor y un rango de fecha.

Datos del Doctor

Cédula	1789653476
Nombre	Luis Hernandez Herrera
Género	M
Fecha Nacimiento	1985-10-10
Especialidad	Implantología

Lista de las reservas de citas en curso

Código de la reserva	Cédula del Paciente	Nombre del Paciente	Fecha de la Cita	Hora empieza la cita	Hora termina la cita	Tipo	Estado
5	1704262193	Gonzalo Herrera	2016-06-01	09:00	09:45	Tratamiento	EN CURSO

Lista de las reservas de citas terminadas

Mostrar Registros Buscar:

Código de la reserva	Cédula del Paciente	Nombre del Paciente	Fecha de la Cita	Hora empieza la cita	Hora termina la cita	Tipo	Estado
Ningún dato disponible en esta tabla							

Mostrando registros del 0 al 0 de un total de 0 registros

Ilustración 52 - Interface Consultar Reserva de Citas de un Determinado Doctor

Elaborado por: Nathaly Herrera

Interface del sistema (Cancelar Cita)

Id	Nombre del Doctor	Especialidad	Fecha de la Cita	Hora empieza la cita	Hora termina la cita	Tipo	Estado	Cancelar
5	Luis Hernandez	Implantología	2016-06-01	09:00	09:45	Tratamiento	EN CURSO	<input type="button" value="Cancelar"/>
1	Sebastián Lema	Ortodoncia	2016-06-01	10:00	10:30	Consulta	EN CURSO	<input type="button" value="Cancelar"/>

Ilustración 53 - Interface de Cancelar Cita

Elaborado por: Nathaly Herrera

Notificación para confirmar la cancelación de la cita.

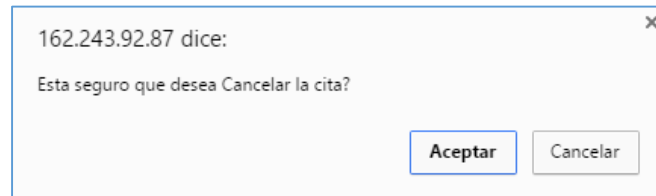


Ilustración 54 - Confirmación para Cancelar la Cita

Elaborado por: Nathaly Herrera

Cuando se cancela la cita al paciente le llega un mail informando la cancelación.

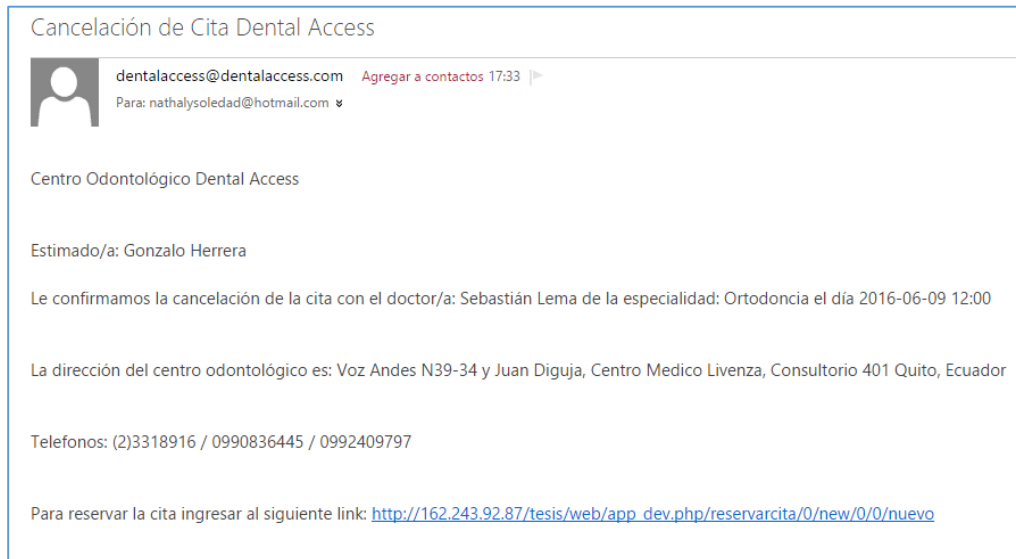


Ilustración 55 - Mail de Cancelar la Cita

Elaborado por: Nathaly Herrera

Interface del sistema (Ficha de la Cita)

Página principal de la Ficha de la Cita

Ingrese la cédula del paciente para buscar sus Citas

Cédula:

Mostrar 10 Registros

Buscar:

Id	Cédula del paciente	Nombre del paciente	Nombre del Doctor	Fecha de la cita	Hora de la cita	problema	Estado de la cita	Editar
1	1704262193	Gonzalo Herrera	Sebastián Lema	2016-06-01	10:00		EN CURSO	
5	1704262193	Gonzalo Herrera	Luis Hernandez	2016-06-01	09:00		EN CURSO	

Mostrando registros del 1 al 2 de un total de 2 registros

Anterior 1 Siguiente

Ilustración 56 - Interface Página principal de la Ficha de la Cita

Elaborado por: Nathaly Herrera

Modificar Ficha de la Cita

Ficha de Cita

Código de Reserva: 1

Nombre del Dentista: Sebastián Lema

Cédula del Paciente: 1704262193 Nombre del Paciente: Gonzalo Herrera

Edad del paciente: 61 Años

[VER FICHA MÉDICA DEL PACIENTE](#) [VER FICHAS DE CITAS ANTERIORES](#)

Fecha de la Cita: 2016-06-01 Hora de la Cita: 10:00

Motivo de Consulta:
(versión del informante)

Problema:

Signos Vitales y Mediciones:

Presión Arterial	Frecuencia Cardíaca (min.)	Frecuencia Respirat (min.)	Temperatura Bucal (°C)	Temperatura Axilar (°C)	Peso (kg)	talla (m)
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Ilustración 57 - Interface Modificar Ficha de la Cita

Elaborado por: Nathaly Herrera

Consultar y Eliminar Ficha de la Cita

Ficha de cita	
Código de la Reserva	1
Cédula del Paciente	1704262193
Nombre del Paciente	Gonzalo Herrera
Nombre del Doctor	Sebastián Lema
Fecha de la Cita	2016-06-01
Hora de la Cita	10:00
Código de la Ficha	1
Motivo de Consulta	
Problema	
Presión Arterial	
Frecuencia Cardíaca	49 min.
Frecuencia Respiratoria	49 min.
Temperatura Bucal	69 °C
Temperatura Axilar	70.9 °C
Peso	70.9 kg.
Estatura	78.9 m.
Examen del Sistema Estomatognático	
Higiene Oral Simplificada	
Periodontal	
Maloclusión	
Fluorosis	
Caries	
Plan Diagnóstico	
Diagnóstico	
Tratamiento	
Notas de Evolución	
Estado de la Cita	EN CURSO

[Editar](#) [Eliminar](#)

Ilustración 58 - Interface Consultar y Eliminar Ficha de la Cita

Elaborado por: Nathaly Herrera

PDF de la Ficha de Citas Acumulada

CENTRO ODONTOLOGICO DENTAL ACCESS
FICHA DE LA CITA ACUMULADA

Establecimiento: DENTAL ACCESS		
Nombres: Cristina Monserrat	Apellidos: Ochoa Albuja	Número De Cédula: 1784873787
Sexo: F	Número de ficha: 2	Número de Historial: 2

Motivo de la Consulta
Le duele la última muela, del lado inferior izquierdo

Problemas Actuales
Tiene Caries

Antecedentes Personales
Ninguna

Ilustración 59 - PDF Ficha de Cita Acumulada

Elaborado por: Nathaly Herrera

El link para imprimir los formularios de consentimientos, solicitados al final de la primera iteración, se realiza el cambio en esta iteración.

Ficha Médica

Ingrese la cédula del paciente para buscar su Ficha Médica

Cédula:

[PDF de Formularios de Consentimientos](#)

Ilustración 60 - Link del PDF Formularios

Elaborado por: Nathaly Herrera

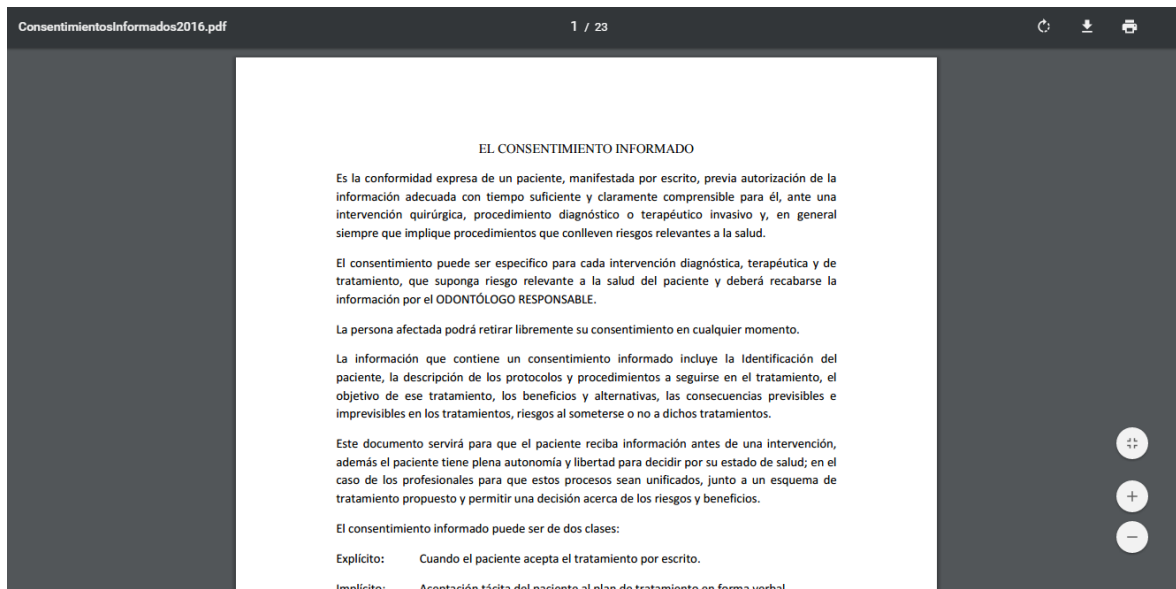


Ilustración 61 - PDF Formularios

Elaborado por: Nathaly Herrera

3.2.4. Fase de Transición

3.2.4.1. Pruebas

Caso de Uso: F1: Ingreso al sistema

Pre-Condición:

El usuario debe tener permisos ya sea de administrados o usuario.

Funcionalidad	Entrada	Resultados Esperados	Cumple?
F1: Ingreso al sistema	<ul style="list-style-type: none"> Ingresar el usuario y contraseña correctamente. Ingresar el usuario o contraseña incorrecta. 	<ul style="list-style-type: none"> Ingresar al sistema, le despliega la página principal, y en el menú le aparece lo que puede gestionar dependiendo del permiso de usuario que tiene. 	<ul style="list-style-type: none"> OK OK

		<ul style="list-style-type: none"> No ingresa al sistema y le notifica que el usuario o contraseña es incorrecta. 	
--	--	--	--

Tabla 11 - Pruebas Caso de Uso Ingreso al sistema

Elaborado por: Nathaly Herrera

Post-Condición:

Si se ingresa al sistema, dependiendo de que permisos tenga, le aparece en el menú lo que puede gestionar.

Caso de Uso: F8: Reservar Cita

Pre-Condición:

El usuario debe tener permisos para reservar cita.

Para consultar reserva de cita debe existir al menos un registro.

Debe existir al menos un horario de atención y un paciente ingresado en el sistema.

Funcionalidad	Entrada	Resultados Esperados	Cumple?
F8: Reservar Cita	Seleccionar en el menú la opción de reservar cita.	Se despliega la pantalla principal de reservar cita.	OK
F8.1: Ingresar	<ul style="list-style-type: none"> Presiona el botón de reservar cita. Escoger los datos para la reserva de cita y verificar disponibilidad. 	<ul style="list-style-type: none"> Se despliega la pantalla para el ingreso de la reserva de cita. Se escoge los datos para la reserva, y si no hay disponibilidad notifica 	<ul style="list-style-type: none"> OK OK OK OK OK

Funcionalidad	Entrada	Resultados Esperados	Cumple?
	<ul style="list-style-type: none"> • Presiona el botón reservar cita. • Se deja un campo vacío. • Escoge una reserva de cita a la misma fecha y rango de hora. 	<p>caso contrario muestra los horarios.</p> <ul style="list-style-type: none"> • Guarda la reserva de la cita, desplegando la página de mostrar nueva reserva. • No guarda los datos y muestra que campo obligatorio se encuentra vacío. • No guarda los datos y notifica que el paciente determinado ya tienen una reserva de cita la misma fecha y rango de hora. 	
F8.2: Ingresar	<ul style="list-style-type: none"> • Página de reservar cita del paciente, ingresar a la página. • Ingresar la cédula del paciente. • Escoge datos de la reserva y verifica la disponibilidad de cita. • Presiona el botón reservar cita. • Se deja campos vacíos. 	<ul style="list-style-type: none"> • Se despliega la pantalla para el ingreso de la reserva de cita. • El sistema verifica la cédula, si existe la cédula muestra las opciones de la reserva, caso contrario notifica que no existe la cédula determinada. • Verifica la disponibilidad, si existe muestra las citas disponibles, caso contrario notifica que no existe horario. 	<ul style="list-style-type: none"> • OK • OK • OK • OK • OK • OK

Funcionalidad	Entrada	Resultados Esperados	Cumple?
	<ul style="list-style-type: none"> El paciente escoge una reserva de cita a la misma fecha y rango de hora. 	<ul style="list-style-type: none"> Guarda la reserva de cita y despliega la página de mostrar datos de la nueva reserva. No guarda los datos, y notifica que hay campos obligatorios que se encuentran vacíos. No guarda datos y notifica que el paciente ya tienen una cita en la misma fecha y rango de hora. 	
F8.3.1: Consulta a Detalle por Paciente	<ul style="list-style-type: none"> Buscar por cédula del paciente. Buscar una cédula que no existe. 	<ul style="list-style-type: none"> Despliega todas las reservas de citas del determinado paciente Muestra una notificación de que la cédula no existe y se queda en la página principal. 	<ul style="list-style-type: none"> OK OK
F8.3.2: Consulta a Detalle por doctor y fecha	Buscar por cédula del doctor y rango de fecha.	Despliega todas las reservas de citas del determinado doctor, y en el rango de fecha.	OK
F8.3.3: Consulta General	Presionar en el menú reserva de cita.	Despliega la pantalla principal con todas las reservas de citas.	OK

Tabla 12 - Pruebas Caso de uso Reserva Cita

Elaborado por: Nathaly Herrera

Post-Condición:

Si se ingresa la reserva de cita, envía un mail al correo con todos los datos de la cita y crea automáticamente una ficha de cita.

Caso de Uso: F9: Cancelar Cita

Pre-Condición:

El usuario debe tener permisos para cancelar la cita.

Para cancelar la cita debe existir al menos una reserva de cita registrado.

Funcionalidad	Entrada	Resultados Esperados	Cumple?
F9: Cancelar Cita	Presionar el botón cancelar cita.	Muestra un mensaje para confirmar si se desea cancelar cita, al presionar ok se cancela correctamente y al presionar cancelar no se efectúa la cancelación de la reserva de cita.	<ul style="list-style-type: none">• OK• OK

Tabla 13 - Pruebas Cancelar Cita

Elaborado por: Nathaly Herrera

Post-Condición:

Al momento de cancelar la cita, al paciente le llega una notificación al mail de la cancelación de la cita.

Caso de Uso: F5: Ficha de Citas

Pre-Condición:

El usuario debe tener permisos para gestionar ficha de citas.

Para modificar, eliminar y consultar debe existir al menos un registro.

La ficha de citas se crea automáticamente con campos vacíos al momento de ingresar una reserva de cita, por lo que debe estar ingresado al menos una reserva para tener ficha de cita.

Funcionalidad	Entrada	Resultados Esperados	Cumple?
F5: Administrar Ficha de Cita	Seleccionar en el menú la opción de ficha de cita.	Se despliega la pantalla principal ficha de cita.	OK
F5.1: Modificar	<ul style="list-style-type: none"> • Presiona el botón Editar ficha de cita. • Modifica los datos. • Ingresar datos incorrectos. 	<ul style="list-style-type: none"> • Despliega la pantalla con todos los datos de la ficha de cita determinado. • El sistema modifica correctamente los datos y muestra los datos de la ficha de cita nuevo. • No le permite modificar los datos e indica que campo tiene el error. 	<ul style="list-style-type: none"> • OK • OK • OK
F5.2: Eliminar	Presiona el botón eliminar.	Muestra un mensaje para confirmar si se desea elimina, al presionar ok se elimina correctamente y al presionar cancelar no se elimina el registro.	OK

Funcionalidad	Entrada	Resultados Esperados	Cumple?
F5.3.1: Consulta a Detalle por paciente	<ul style="list-style-type: none"> • Buscar por cédula del paciente. • Buscar una cédula que no existe. 	<ul style="list-style-type: none"> • Despliega todas las fichas de citas del paciente determinado • Muestra una notificación de que la cédula no existe y se queda en la página principal. 	<ul style="list-style-type: none"> • OK • OK
F5.3.2: Consulta a Detalle	Busca por el código de la ficha de cita	Despliega toda la información de la ficha de cita.	OK
F5.3.3: Consulta General	Presionar en el menú ficha de cita.	Despliega la pantalla principal con todas las fichas de las citas.	OK

Tabla 14 - Pruebas del Caso de Uso Ficha de la Cita

Elaborado por: Nathaly Herrera

Post-Condición:

Si se modifica o elimina algún registro se realizará los cambios en la base de datos respectivamente.

Resultados

Después de realizar las pruebas de la segunda iteración que consta de los últimos 5 requerimientos, se puede determinar que cumple con todo lo descrito en los casos de uso, no se encontró ningún error al momento que se realizó las pruebas en el Centro Odontológico Dental Access, cumple con todos los requerimientos principales del usuario, el único inconveniente que se presento fue en la modificación de ficha de citas, porque dos campos que no debían ser obligatorios, estaba puesto como obligatorio pero eso fue corregido inmediatamente.

Todo el sistema cumple con los requerimientos establecidas, y paso todas las pruebas realizadas al usuario.

3.2.4.2. Despliegue

Este flujo de trabajo describe cómo utilizar la aplicación web y así los usuarios finales puedan utilizar de mejor manera el software sin ningún inconveniente. Por lo tanto, se realiza un manual de usuario que se encuentra en el anexo (5.1) del mismo documento.

También describe como instalar la aplicación web, ya que el software se encuentra instalada en un servidor llamado Digita Ocean que será detallado en el manual de instalación que se encuentra en el anexo (5.2), del mismo documento.

El sistema ya fue aprobado por el Centro odontológico y para su utilización el personal fue capacitado para la utilización del software. Igualmente, se le entregara un manual de usuario y de instalación.

Topología de la aplicación:

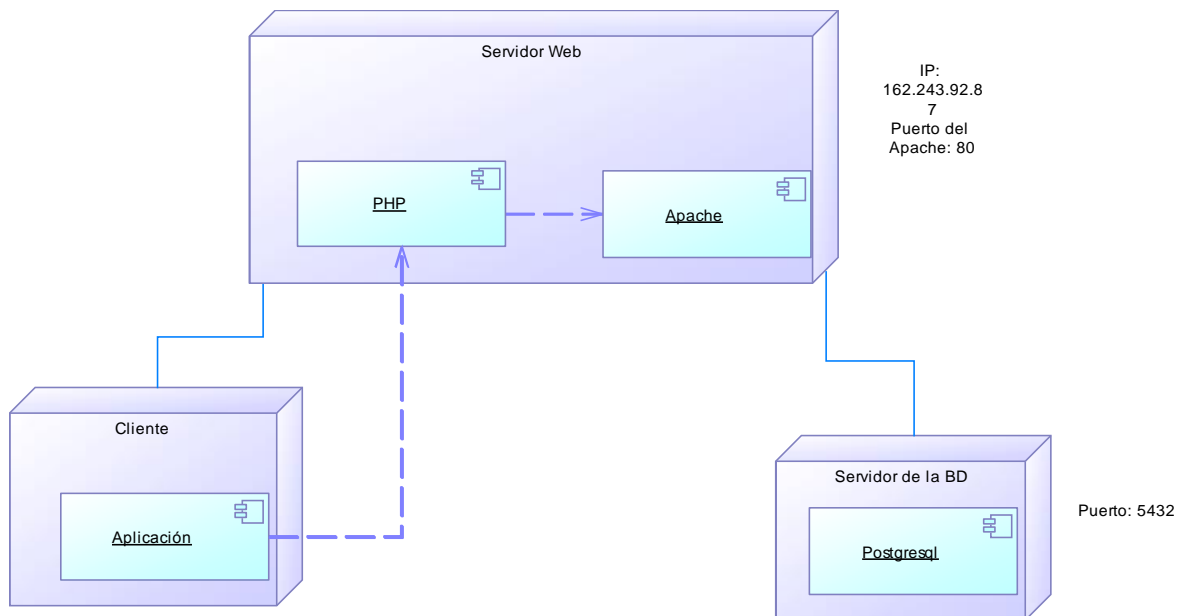


Diagrama 72 - Diagrama de Despliegue

Elaborado por: Nathaly Herrera

3.2.4.3. *Entorno*

La aplicación web se encuentra instalada en un servidor llamado Digital Ocean, que está basado en Ubuntu LAMP 14.04.4 x64, con memoria de disco duro de 20GB, memoria RAM de 512MB y 1000GB de transferencia.

Se utiliza la base de datos postgresql

Para su elaboración el framework Symfony que está basado en php y twig.

CAPITULO IV: CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

- El Centro Odontológico Dental Access actualmente no cuenta con un sistema que le permita gestionar un expediente médico de los pacientes y las reservas de citas para las diferentes especialidades, lo que ha significado la pérdida de clientes por no realizar un seguimiento adecuado y no contar con una apropiada coordinación al momento de reservar citas.
- La metodología RUP permitió elaborar la aplicación web mediante la realización de pruebas continuas y retroalimentación por parte del usuario en cada una de las iteraciones de tal manera que se satisfaga las necesidades del cliente en el sistema desarrollado.
- Durante el ciclo de desarrollo de la aplicación web, se ha cumplido de manera organizada y distribuidas con todas las fases exigidas por la metodología RUP.
- Durante la elaboración de la aplicación web se utilizó el framework Symfony logrando automatizar las funciones que son comunes para enfocarse en el desarrollo de la parte compleja del sistema reduciendo el tiempo de implementación.
- El framework Symfony está basado en el patrón de diseño MVC, dividiendo a la aplicación en tres elementos identificables para que el código este correctamente distribuido e integrado.
- El software ayudará al Centro Odontológico Dental Access a realizar un seguimiento adecuado del paciente mediante la gestión de fichas médicas y de citas.

- El sistema cuenta con la gestión de pacientes, doctores, empleados, que le permitirá llevar un registro detallado de la ficha médica y de cada cita, también admite la automatización del proceso de reservar de citas, permitiendo al Centro odontológico llevar un registro organizado de las citas y los pacientes pueden reservar desde cualquier computador que tenga acceso a internet, generando un valor agregado a Dental Access.
- Al estar la aplicación en un servidor, el manejo del sistema es accesible y fácil de usar, en razón que la aplicación no necesita una instalación extra y se puede ingresar desde cualquier computador que tenga acceso a internet.
- Utilizando un servidor basado en Linux Ubuntu se configuró fácilmente debido a que se adapta a cualquier necesidad de la aplicación, y trabaja con los diferentes tipos de motores de base de datos, que en este caso es Postgresql.
- El servidor Linux Ubuntu a comparación de los demás servidores como Windows server tiene menor costo y una amplia documentación que soluciona los problemas que transcurran en la instalación, configuración o producción del servidor.
- Mediante el manejo de estándares, en la codificación permite obtener una aplicación más funcional y fácil de utilizar por parte del usuario. El estándar principal utilizado es el CIE10 para los diagnósticos.

4.2. Recomendaciones

- Al utilizar la metodología RUP, se recomienda que el usuario debe estar involucrado en todas las iteraciones del proyecto principalmente en las fases de requisitos y pruebas, para que el proyecto cumpla con todas las necesidades del cliente y no requiera un cambio profundo al final del desarrollo de la aplicación.

- La fase de requerimientos es la base para desarrollar la aplicación de acuerdo a las necesidades del usuario se recomienda que se entienda desde el principio los requerimientos para que no se realice cambios profundos en el trascurso del proyecto.
- En la aplicación hay que tomar en cuenta, para el ingreso de ficha médica y la ficha de cita primero se ingresa el registro del paciente y la reserva de cita respectivamente, para que se cree automáticamente las fichas del paciente.
- Para el correcto manejo del sistema, se recomienda revisar el manual de usuario en donde detalla el funcionamiento correcto de la aplicación.
- Si se requiere cambiar de servidor ya sea en el mismo proveedor o no, se recomienda revisar el manual de instalación para configurar de mejor manera el sistema en el nuevo servidor.
- Se recomienda capacitar al administrador de la aplicación porque es el encargado de manejar todo el sistema, las funciones principales son gestionar usuarios para el ingreso al sistema, empleados, pacientes de la organización.
- Se recomienda mantener la estructura del Framework y utilizar las mismas versiones de las herramientas de trabajo al momento de agregar nuevas funciones al aplicativo, para garantizar el mantenimiento y compatibilidad del sistema.
- Se recomienda realizar respaldos diarios o utilizar un repositorio para guardar las diferentes versiones para evitar cualquier tipo de inconvenientes como es la pérdida de información o error en la aplicación.
- Para una versión 2.0 de la presente aplicación web, se recomienda la realización de un odontograma para tener un mejor seguimiento de los tratamientos realizados al paciente. También se puede dar la opción de guardar las radiografías.

- Para una aplicación futura que complemente al actual sistema se recomienda realizar la facturación electrónica con el objetivo de tener un software completo, que beneficie al Centro Odontológico.

CAPITULO V

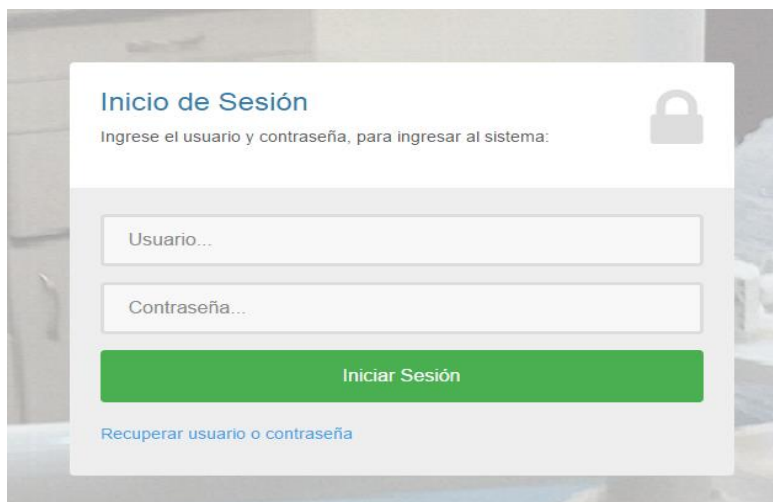
Anexos

5.1. Manual de Usuario

Bienvenido al Sistema del Centro Odontológico Dental Access que permitirá la gestión de usuarios, empleados, doctores, pacientes, horarios de atención, fichas médicas, ficha de citas, el ingreso al sistema y los procesos de reserva cita y cancelar cita.

1. Lo primero que debe realizar es ingresar a la aplicación web mediante la dirección URL: http://XXX.XXX.XX.XX/tesis/web/app_dev.php/login
2. Ingresar al sistema mediante el usuario y clave asignado.

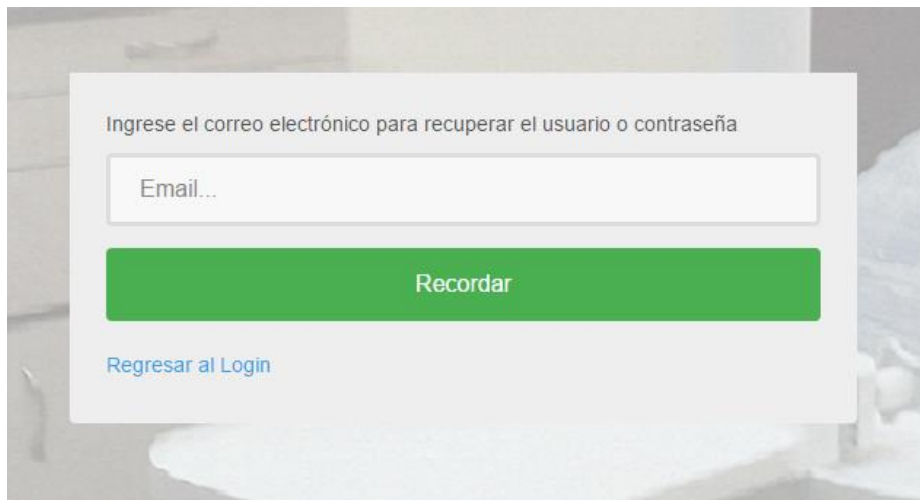
Dependiendo de que permisos tenga al ingresar al sistema le aparece las diferentes opciones para administrar. En el sistema existe 2 tipos de roles los cuales son el administrador que puede gestionar usuarios, empleado, doctores, pacientes, horarios de atención, fichas médicas, ficha de citas, el ingreso al sistema y los procesos de reserva cita y cancelar cita. El otro rol es del empleado y doctor que puede gestionar paciente, horario de atención, ficha médica, ficha de cita y los procesos reservar cita y cancelar cita.



Si ingresan el usuario o contraseña incorrecto les despliega una notificación



Si no recuerda el usuario o contraseña, ir al link y poner el email, para que le llegue el usuario y contraseña del doctor o empleado.



Al ingresar correctamente al sistema se despliega la pantalla principal

Si tiene el rol de administrador le despliega en el menú todas las opciones mencionadas anteriormente:

CONSTRUCCIÓN DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE CITAS, TRATAMIENTOS Y EXPEDIENTE MÉDICO DE LOS PACIENTES DEL CENTRO IMPLANTOLÓGICO Y DE ESPECIALIDADES DENTAL ACCESS

The screenshot shows a web application interface with a dark blue navigation bar at the top containing the following menu items: Inicio, Usuario, Empleado, Doctor, Paciente, Horario de Atención, Ficha Médica, Ficha de la Cita, Reservar Cita, and Cerrar Sesión. The main content area has a white background with the title 'Administrar Centro Odontológico Dental Access' in blue. Below the title is the 'DENTAL Access' logo, which features a stylized tooth and the tagline 'Tu sonrisa perfecta'. The text below the logo states: 'El sistema permitirá administrar el centro odontológico de la siguiente forma: Si el usuario tiene permisos de administrador podrá acceder a todos los módulos del sistema como son:'. A bulleted list follows, detailing the permissions for an administrator:

- Ingresar, modificar, eliminar y mostrar usuarios
- Ingresar, modificar, eliminar y mostrar doctor
- Ingresar, modificar, eliminar y mostrar empleado
- Ingresar, modificar, eliminar y mostrar paciente
- Ingresar, modificar, eliminar y mostrar horario de atención
- Modificar, eliminar y mostrar ficha médica
- Modificar, eliminar y mostrar ficha de cita
- Ingresar, mostrar y cancelar reserva de cita


Si tiene el rol de empleado o doctor se le despliega las opciones mencionadas anteriormente.

This screenshot is identical to the one above, showing the same navigation bar and main content area. However, the 'Inicio' menu item in the navigation bar is highlighted, indicating the user is currently on the home page. The rest of the interface, including the title, logo, and text, remains the same.

3. Administrar Usuario, empleado, doctor y paciente

Al asignar un usuario y contraseña al empleado o doctor, se le notifica por email

Registrar Usuario

**dentalaccess@dentalaccess.com** [Agregar a contactos](#) 17:36 ▶
Para: nathalysoledad@hotmail.com ▼

Centro Odontológico Dental Access

Estimado/a: Luis Hernandez

Para ingresar al sistema:

Su usuario: usu1

Su contraseña: abc123

Para Cambiar el usuario y contraseña, se ingresa el número de cédula y contraseña anterior para el cambio.

Editar Usuario

Ingrese el número de cédula y contraseña para editar usuario o contraseña

Cédula del Paciente:

Contraseña Anterior:

Buscar

Usuario:

Contraseña:

Guardar

Al ser estas 4 clases similares se describirá la clase paciente su funcionamiento.

Primero se escoge en el menú la opción paciente y se despliega la pantalla principal, en donde muestra una tabla con todos los pacientes ingresados.

Paciente

CÉDULA:

Mostrar Registros Buscar:

Cédula	Primer Nombre	Segundo Nombre	Apellido Paterno	Apellido Materno	Fecha de Nacimiento	Género	Editar
1704262193	Gonzalo	Orlando	Herrera	Villarreal	1955-09-29	M	✎
1727453476	Christian	Gonzalo	Olivo	Parra	1992-07-07	M	✎
1745239876	Daniela	Stephanie	Tobar	Cifuentes	1994-06-05	F	✎

Mostrando registros del 1 al 3 de un total de 3 registros

Para ingresar un nuevo paciente se presiona el botón crear nuevo registro y se despliega la pantalla de nuevo paciente.

Nuevo Paciente

Cédula:

Primer Nombre: Segundo Nombre:

Apellido Paterno: Apellido Materno:

Fecha Nacimiento: Ciudad De Origen:

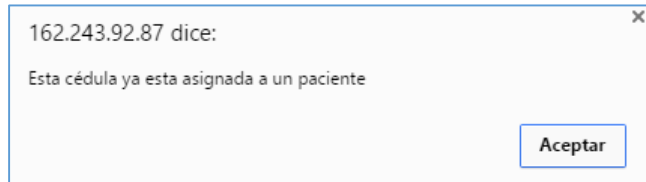
Género: Masculino Femenino

Estado Civil:

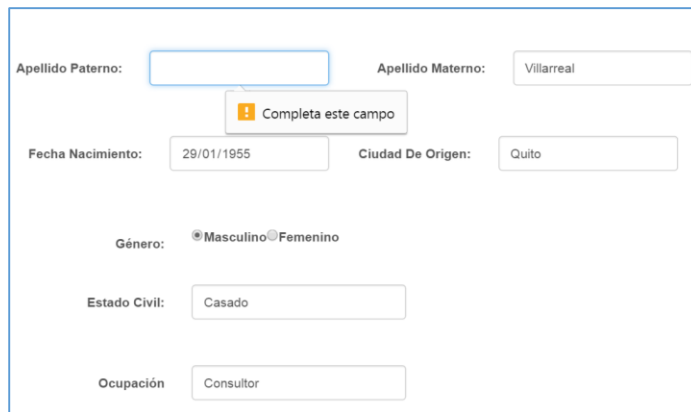
Se ingresa los datos y se presiona en el botón guardar, y se registra en la base de datos.



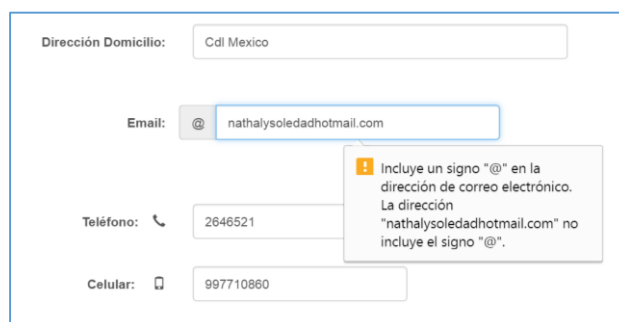
Si se ingresa una cédula que ya existe, se despliega una alerta de que ya existe la cédula.





Si un campo obligatorio se deja vacío se despliega la siguiente notificación:



Si en el campo email se ingresa en formato incorrecto, se presenta la siguiente notificación:



Si en el campo teléfono y celular se ingresa los números menores de 7 dígitos o mayores a 10, se presenta el siguiente error:

Teléfono: 	<input type="text" value="26"/>	• El número telefónico debe tener al menos 7 dígitos
Celular: 	<input type="text" value="997710860666"/>	• El número de celular no debe tener más de 10 dígitos

Si se ingresa la cédula con más de 10 dígitos, se presenta el siguiente error:

Cédula:	<input type="text" value="170426219333"/>	• La cédula no puede tener más de 10 dígitos
---------	---	--

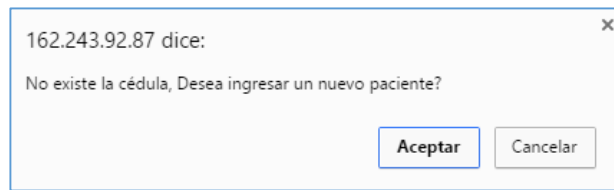
Si se excede el número de caracteres en un campo, se presenta el siguiente error:

Ocupación	<input type="text" value="Consultor de la empresa de Incac"/>	• El campo ocupación no puede tener más de 50 caracteres
-----------	---	--

Para Mostrar los datos de un paciente en la pantalla principal se ingresa el número de cédula.

CÉDULA:	
<input type="text"/>	<input type="button" value="Buscar"/> 

Si se ingresa un número de cédula que no se encuentra registrado se despliega la siguiente notificación. Si se da aceptar se despliega la pantalla de nuevo paciente caso contrario se queda en la página principal.

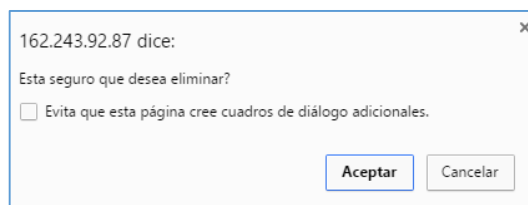


Si se ingresa una cédula existente se despliega la pantalla que muestra todos los datos del paciente.

Paciente	
Id	3
Primer Nombre	Daniela
Segundo Nombre	Stephanie
Apellido Paterno	Tobar
Apellido Materno	Cifuentes
Fecha Nacimiento	1994-06-05
Género	F
Cédula	1745239876
Ciudad Nacimiento	Loja
Estado Civil	Casada
Ocupación	Parvularia
Dirección Domicilio	Upano S7-87 y Paute
Teléfono	2876545
Celular	
Email	nathalysoledad29@gmail.com

Editar Eliminar

Para eliminar el registro se presiona el botón eliminar que se encuentra en la pantalla de mostrar paciente. Y presenta una confirmación para la eliminación.



Para editar se presiona el botón editar en la pantalla de mostrar paciente y se despliega la pantalla modificar paciente, que al igual que el ingresar tiene las mismas validaciones.

Editar Paciente

Cédula:

Primer Nombre: Segundo Nombre:

Apellido Paterno: Apellido Materno:

Fecha Nacimiento: Ciudad De Origen:

Género: Masculino Femenino

Estado Civil:

También se puede editar o consultar en la pantalla principal, en la tabla donde muestra todos los pacientes.

Mostrar Registros Buscar:

Cédula	Primer Nombre	Segundo Nombre	Apellido Paterno	Apellido Materno	Fecha de Nacimiento	Género	Editar
1704262193	Gonzalo	Orlando	Herrera	Villarreal	1955-09-29	M	✎
1727453476	Christian	Gonzalo	Olivo	Parra	1992-07-07	M	✎
1745239876	Daniela	Stephanie	Tobar	Cifuentes	1994-06-05	F	✎

Mostrando registros del 1 al 3 de un total de 3 registros

4. Administrar Ficha médica y Ficha de la cita

Para el modificar, consultar y eliminar es igual que el administrar paciente descrito anteriormente. Con la diferencia que no tiene ingresar porque se crea automáticamente las fichas al momento de ingresar un paciente se crea la ficha médica y al reservar una cita se crea la ficha de la cita.

En la Ficha Médica, en la pantalla principal, se encuentra un link para abrir un pdf con todos los formularios de consentimiento que se necesita para iniciar un tratamiento.

Ficha Médica

Ingrese la cédula del paciente para buscar su Ficha Médica

Cédula:

Buscar por Paciente 🔍

[PDF de Formularios de Consentimientos](#)

PDF de formularios:

EL CONSENTIMIENTO INFORMADO

Es la conformidad expresa de un paciente, manifestada por escrito, previa autorización de la información adecuada con tiempo suficiente y claramente comprensible para él, ante una intervención quirúrgica, procedimiento diagnóstico o terapéutico invasivo y, en general siempre que implique procedimientos que conlleven riesgos relevantes a la salud.

El consentimiento puede ser específico para cada intervención diagnóstica, terapéutica y de tratamiento, que suponga riesgo relevante a la salud del paciente y deberá recabarse la información por el ODONTÓLOGO RESPONSABLE.

La persona afectada podrá retirar libremente su consentimiento en cualquier momento.



La información que contiene un consentimiento informado incluye la Identificación del paciente, la descripción de los protocolos y procedimientos a seguirse en el tratamiento, el objetivo de ese tratamiento, los beneficios y alternativas, las consecuencias previsibles e imprevisibles en los tratamientos, riesgos al someterse o no a dichos tratamientos.

Este documento servirá para que el paciente reciba información antes de una intervención, además el paciente tiene plena autonomía y libertad para decidir por su estado de salud; en el caso de los profesionales para que estos procesos sean unificados, junto a un esquema de tratamiento propuesto y permitir una decisión acerca de los riesgos y beneficios.

El consentimiento informado puede ser de dos clases:

Explícito: Cuando el paciente acepta el tratamiento por escrito.

Para observar la ficha de la cita acumulada en formato pdf se busca la ficha por un determinado paciente, y se presiona el link del PDF:

Notas de Evolución	
Estado de la Cita	TERMINADO
Editar  Eliminar 	
Regresar a la lista de ficha de citas	
Ver Ficha Médica del Paciente	
Ver PDF de la Ficha	

Y se abre el PDF con la ficha del determinado paciente, en donde se puede imprimir dicha ficha.

 CENTRO ODONTOLÓGICO DENTAL ACCESS	
FICHA DE LA CITA ACUMULADA	
Establecimiento: DENTAL ACCESS	
Nombres: Cristina Monserrat Apellidos: Ochoa Albuja Número De Cédula: 1784873787	
Sexo: F Número de ficha: 2 Número de Historial: 2	
Motivo de la Consulta	
Le duele la última muela, del lado inferior izquierdo	
Problemas Actuales	
Tiene Caries	
Antecedentes Personales	
Ninguna	

5. Administrar Horario de Atención

Para ingresar un horario de atención debe estar registrado por lo menos un doctor en el sistema.

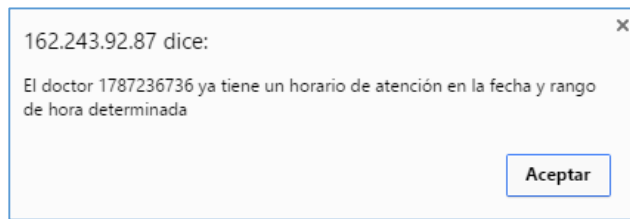
The screenshot shows the 'Horario de Atención' interface. At the top, there is a search bar for doctor IDs with a 'Buscar por Doctor' button. Below it is a 'Crear Nuevo Horario de Atención' button. A table displays existing appointments with columns for Id, Fecha, Hora Inicial, Hora Finalizar, Tipo de Atención, and Editar. The table contains two records: one for 'Tratamiento' at 09:00-09:45 and another for 'Consulta' at 10:00-10:30. A pagination bar at the bottom shows 'Mostrando registros del 1 al 2 de un total de 2 registros' and navigation buttons for 'Anterior', '1', and 'Siguiete'.

Id	Fecha	Hora Inicial	Hora Finalizar	Tipo de Atención	Editar
1	2016-06-01	09:00	09:45	Tratamiento	✎
3	2016-06-01	10:00	10:30	Consulta	✎

Para ingresar un nuevo horario se presiona el botón crear nuevo horario de atención y se despliega la pantalla de ingreso.

The screenshot shows the 'Crear Horario de Atención' form. It includes a dropdown menu for 'Cédula Doctor' with three options: 1789653476, 1787236736, and 1787652634. Below this are input fields for 'Fecha' (03/06/2016), 'Tipo de Atención' (Consulta), and 'Hora' (09:00). A 'Guardar' button is located at the bottom right, and a link 'Regresar a la lista de horarios de atención' is at the bottom left.

Si se escoge un horario en la misma fecha y rango de hora para el mismo doctor le despliega una notificación:



Las validaciones, el modificar, consultar y eliminar es el mismo funcionamiento que el administrar paciente.

6. Reservar Cita

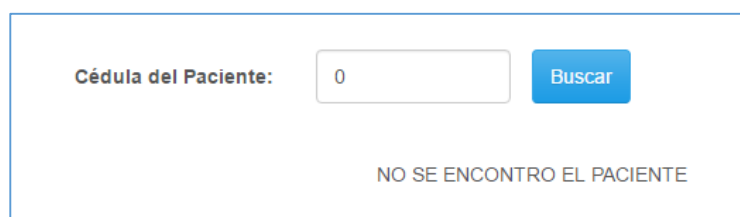
Si el paciente reserva la cita, ingresa a la aplicación con el siguiente url:

http://XXX.XXX.XX.XX /tesis/web/app_dev.php/reservarcita/0/new/0/0/nuevo

Muestra la siguiente pantalla



Ingresa el número de cédula, si la cédula no está registrada le notifica lo siguiente, para reservar cita el paciente debe estar registrado en el sistema:



Si existe el número de cédula muestra las especialidades y la fecha a escoger.

Ingrese el número de cédula del paciente, para reservar la cita

Cédula del Paciente:

Nombre del Paciente: Gonzalo Herrera

Especialidad: Fecha de la Cita:

Se escoge la especialidad y la fecha, y se presiona el botón buscar cita, si no hay disponibilidad les muestra una notificación.

Ingrese el número de cédula del paciente, para reservar la cita

Cédula del Paciente:

Nombre del Paciente: Gonzalo Herrera

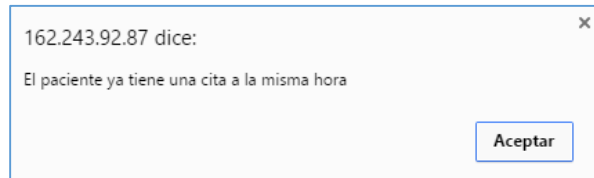
Especialidad: Fecha de la Cita:

No se encontro horarios en esa fecha para la determinada especialidad

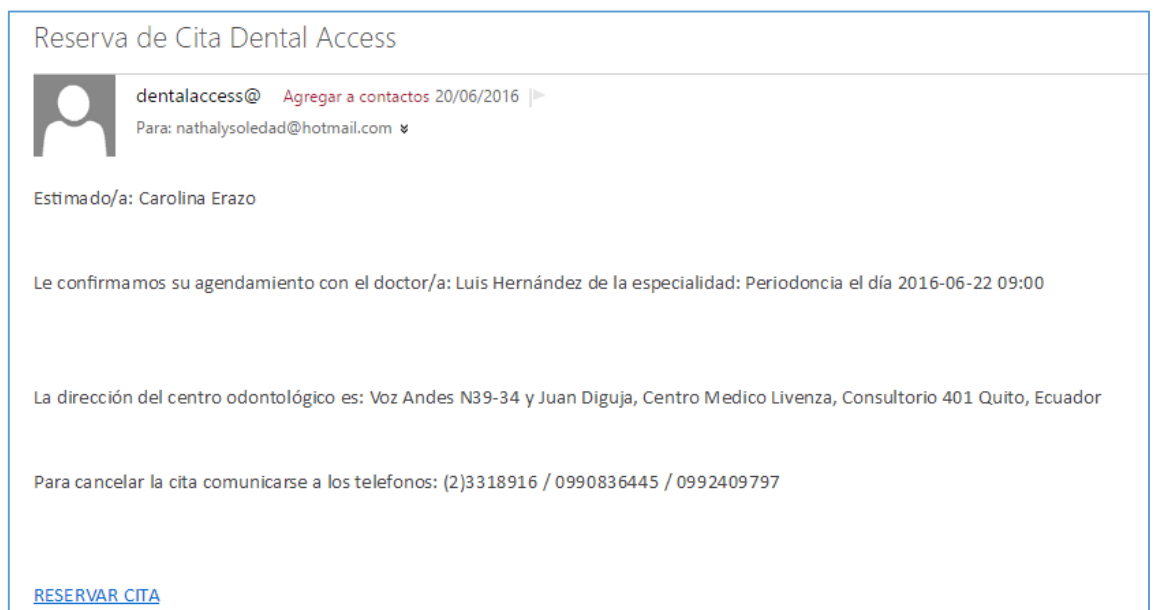
Si Hay disponibilidad le muestra las citas disponibles y presiona en el botón reservar.

Doctor	Especialidad	Fecha	Hora empieza la cita	Hora termina la cita	tipo	Escoger
Karla Lazo	Implantología	2016-06-03	09:00	09:30	Consulta	<input type="radio"/>

Si el paciente ya tiene una cita en la misma fecha y en el mismo rango de hora, se despliega una notificación.



Al momento de reserva cita, al paciente le llega un mail notificando la reserva de la cita:



Reserva realizada por los usuarios del sistema

Si la reserva es realizada por los usuarios del Centro Odontológico, deben ingresar al sistema.



Presionan el botón reservar cita y se despliega la pantalla de reservar cita

Reservar Cita

Cédula del Empleado:

Cédula del Paciente:

Especialidad: Fecha de la Cita:

Se escoge las opciones y es el mismo procedimiento descrito anteriormente.

Para Consultar la reserva de citas hay dos opciones:

- Consultar por paciente

Se ingresa el número de cédula.

Ingrese la cédula del paciente para buscar sus reservas

Cédula:

Se despliega las reservas de citas del paciente

Datos del Paciente								
Cédula	1704262193							
Nombre	Gonzalo Herrera							
Género	M							
Fecha Nacimiento	1955-09-29							
Lista de las reservas de citas del paciente en curso								
Id	Nombre del Doctor	Especialidad	Fecha de la Cita	Hora empieza la cita	Hora termina la cita	Tipo	Estado	Cancelar
6	Karla Lazo	Implantología	2016-06-03	09:00	09:30	Consulta	EN CURSO	<input type="button" value="Cancelar"/>
5	Luis Hernandez	Implantología	2016-06-01	09:00	09:45	Tratamiento	EN CURSO	<input type="button" value="Cancelar"/>
1	Sebastián Lema	Ortodoncia	2016-06-01	10:00	10:30	Consulta	EN CURSO	<input type="button" value="Cancelar"/>
Lista de las reservas de citas del paciente terminadas								
Mostrar <input type="text" value="5"/> Registros Buscar: <input type="text"/>								

- Por doctor y rango de fecha

Escoja la cédula del doctores y un rango de fecha

Cédula:

Fecha: desde hasta

Se escoge las opciones y se despliega la información

Datos del Doctor

Cédula	1789653476
Nombre	Luis Hernandez Herrera
Género	M
Fecha Nacimiento	1985-10-10
Especialidad	Implantología

Lista de las reservas de citas en curso

Código de la reserva	Cédula del Paciente	Nombre del Paciente	Fecha de la Cita	Hora empieza la cita	Hora termina la cita	Tipo	Estado	Cancelar
5	1704262193	Gonzalo Herrera	2016-06-01	09:00	09:45	Tratamiento	EN CURSO	<input type="button" value="Cancelar"/>

7. Cancelar Cita

Para Cancelar la cita se busca por paciente o por doctor con rango de fecha.

Cuando se despliega la información se presiona el botón cancelar

CONSTRUCCIÓN DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE CITAS, TRATAMIENTOS Y EXPEDIENTE MÉDICO DE LOS PACIENTES DEL CENTRO IMPLANTOLÓGICO Y DE ESPECIALIDADES DENTAL ACCESS

Datos del Paciente

Cédula	1704262193
Nombre	Gonzalo Herrera
Género	M
Fecha Nacimiento	1955-09-29

Lista de las reservas de citas del paciente en curso

Id	Nombre del Doctor	Especialidad	Fecha de la Cita	Hora empieza la cita	Hora termina la cita	Tipo	Estado	Cancelar
6	Karla Lazo	Implantología	2016-06-03	09:00	09:30	Consulta	EN CURSO	<input type="button" value="Cancelar"/>
5	Luis Hernandez	Implantología	2016-06-01	09:00	09:45	Tratamiento	EN CURSO	<input type="button" value="Cancelar"/>
1	Sebastián Lema	Ortodoncia	2016-06-01	10:00	10:30	Consulta	EN CURSO	<input type="button" value="Cancelar"/>

Lista de las reservas de citas del paciente terminadas

Mostrar Registros Buscar:

Se despliega la confirmación para la cancelación de la reserva de cita.

162.243.92.87 dice: x

Esta seguro que desea cancelar la cita?

Al momento de Cancelar la cita, al paciente le llega una notificación al mail, que se ha cancelado la cita:

Cancelación de Cita Dental Access

dentalaccess@ Agregar a contactos 20/06/2016 ▶
Para: nathalysoledad@hotmail.com v

Estimado/a: Carolina Erazo

Le confirmamos la cancelación de la cita con el doctor/a: Luis Hernández de la especialidad: Periodoncia el día 2016-06-22 09:00

La dirección del centro odontológico es: Voz Andes N39-34 y Juan Diguja, Centro Medico Livenza, Consultorio 401 Quito, Ecuador

Telefonos: (2)3318916 / 0990836445 / 0992409797

RESERVAR CITA

8. Cerrar sesión

Esta opción se encuentra en todas las páginas en el menú en la parte superior derecha:

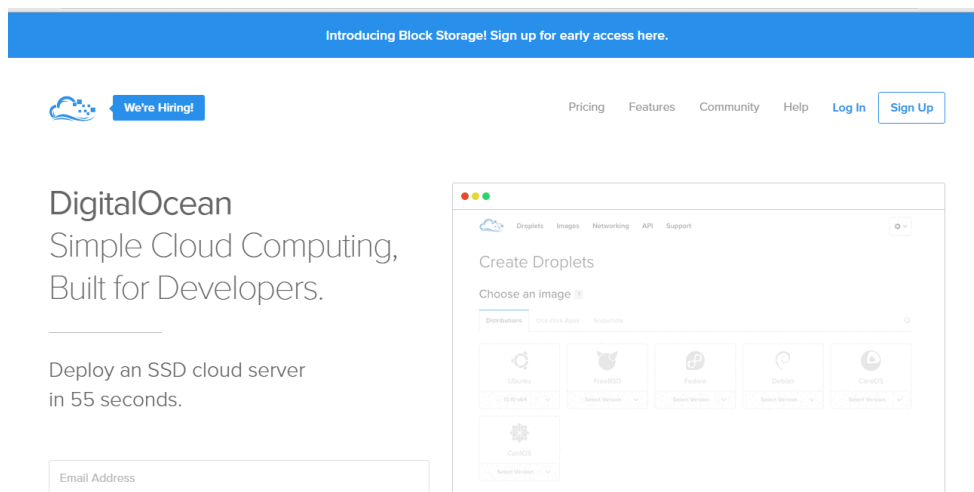


5.2. Manual de Instalación

Bienvenido al manual de instalación, este manual explicará cómo instalar la aplicación web del Centro Odontológico Dental Access en el Servidor Digita Ocean.

1. Primero Ingresar a la página de Digital Ocean, si no tiene cuenta registrarse, caso contrario iniciar sesión.

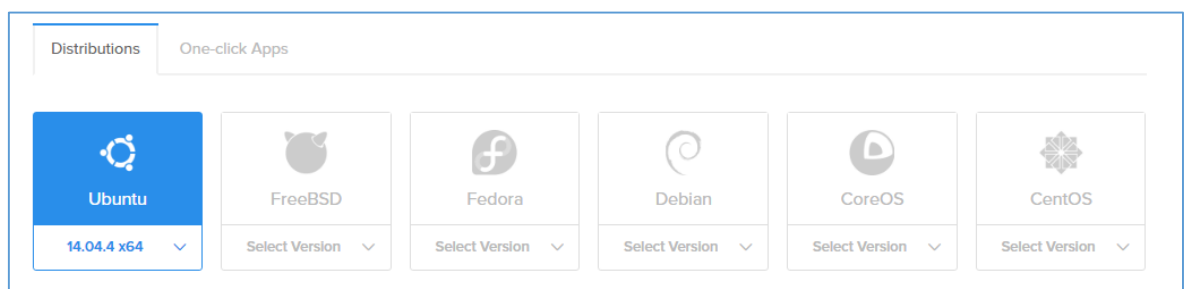
<https://www.digitalocean.com/>



2. Presionar el botón crear droplet



3. Escoger la pestaña Distribution y escoger la opción Ubuntu.



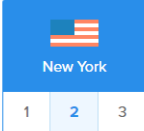
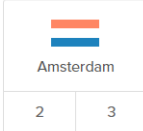
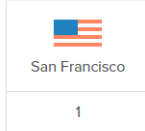
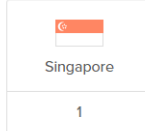

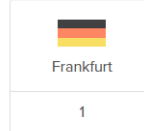
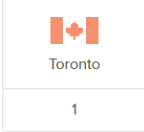
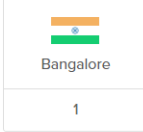
4. Escoger el tamaño que en este caso es la primera opción

Choose a size

\$5/mo \$0.007/hour 512 MB / 1 CPU 20 GB SSD Disk 1000 GB Transfer	\$10/mo \$0.015/hour 1 GB / 1 CPU 30 GB SSD Disk 2 TB Transfer	\$20/mo \$0.030/hour 2 GB / 2 CPUs 40 GB SSD Disk 3 TB Transfer	\$40/mo \$0.060/hour 4 GB / 2 CPUs 60 GB SSD Disk 4 TB Transfer	\$80/mo \$0.119/hour 8 GB / 4 CPUs 80 GB SSD Disk 5 TB Transfer	\$160/mo \$0.238/hour 16 GB / 8 CPUs 160 GB SSD Disk 6 TB Transfer
\$320/mo \$0.476/hour 32 GB / 12 CPUs 320 GB SSD Disk 7 TB Transfer	\$480/mo \$0.714/hour 48 GB / 16 CPUs 480 GB SSD Disk 8 TB Transfer	\$640/mo \$0.952/hour 64 GB / 20 CPUs 640 GB SSD Disk 9 TB Transfer			

5. Escoger la región, en donde se encuentre el servidor

Choose a datacenter region

 New York 1 2 3	 Amsterdam 2 3	 San Francisco 1	 Singapore 1	 London 1	 Frankfurt 1
 Toronto 1	 Bangalore 1				

6. Poner el nombre del droplet

Finalize and create

<p>How many Droplets?</p> <p>Deploy multiple Droplets with the same configuration .</p> <p>– 1 Droplet +</p>	<p>Choose a hostname</p> <p>Give your Droplets an identifying name you will remember them by. Your Droplet name can only contain alphanumeric characters, dashes, and periods.</p> <p>ubuntu-512mb-nyc2-01</p>
---	---

7. Abrir la consola, poner el usuario y contraseña del servidor



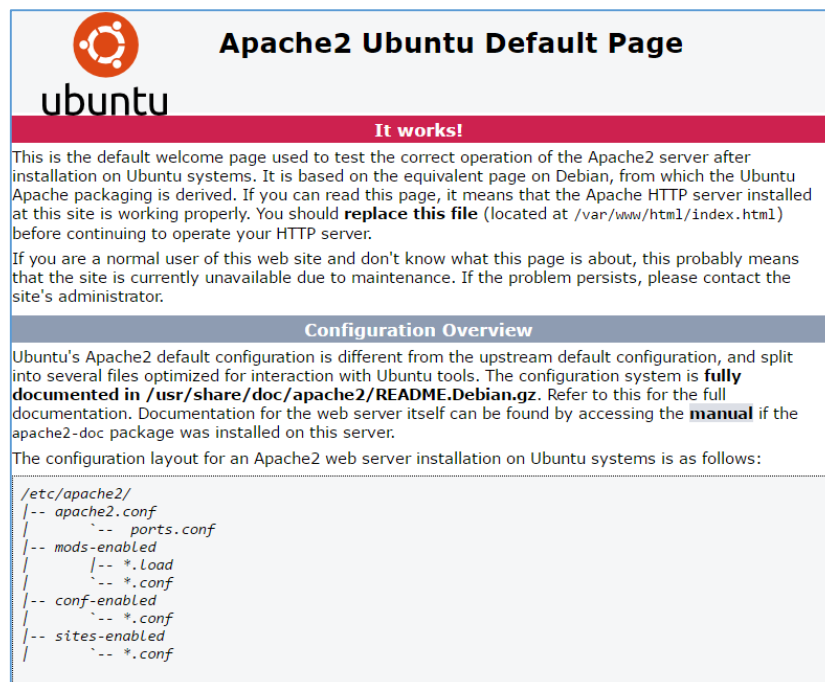
```
162.243.92.87 - PuTTY
login as: root
root@162.243.92.87's password: █
```

8. Para instalar el Apache se pone estos dos comandos:

```
To delete this message of the day: rm -rf /etc/motd.tail
Last login: Wed Jun  1 10:24:08 2016 from 190.63.119.9
root@DentalAccess:~# sudo apt-get update █
```

```
To delete this message of the day: rm -rf /etc/motd.tail
Last login: Wed Jun  1 10:24:08 2016 from 190.63.119.9
root@DentalAccess:~# sudo apt-get install apache2 █
```

Para verificar si se ha instalado correctamente se pone el IP en la URL:



9. Para instalar PHP se pone los siguientes comandos (Versión de PHP 5.5.9-1)

```
To delete this message of the day: rm -rf /etc/motd.tail
Last login: Wed Jun 1 10:24:08 2016 from 190.63.119.9
root@DentalAccess:~# sudo apt-get install php5 libapache2-mod-php5 php5-mcrypt
```

Se ingresa al siguiente archivo

```
To delete this message of the day: rm -rf /etc/motd.tail
Last login: Wed Jun 1 10:24:08 2016 from 190.63.119.9
root@DentalAccess:~# sudo nano /etc/apache2/mods-enabled/dir.conf
```

Se modifica el directorio del index lo siguiente:

```
GNU nano 2.2.6 file: /etc/apache2/mods-enabled/dir.conf
<IfModule mod_dir.c>
    DirectoryIndex index.php index.html index.cgi index.pl index.xhtml index.htm
</IfModule>
```

Para verificar si se instala correctamente se pone en el URL el IP seguido de info ()

162.243.92.87/info.php

PHP Version 5.5.9-1ubuntu4.17 	
System	Linux DentalAccess 3.13.0-83-generic #127-Ubuntu SMP Fri Mar 11 00:25:37 UTC 2016 x86_64
Build Date	May 19 2016 19:05:33
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d
Additional .ini files parsed	/etc/php5/apache2/conf.d/05-opcache.ini, /etc/php5/apache2/conf.d/10-pdo.ini, /etc/php5/apache2/conf.d/20-curl.ini, /etc/php5/apache2/conf.d/20-gd.ini, /etc/php5/apache2/conf.d/20-intl.ini, /etc/php5/apache2/conf.d/20-json.ini, /etc/php5/apache2/conf.d/20-mysql.ini, /etc/php5/apache2/conf.d/20-mysqli.ini, /etc/php5/apache2/conf.d/20-pdo_mysql.ini, /etc/php5/apache2/conf.d/20-pdo_pgsql.ini, /etc/php5/apache2/conf.d/20-pgsql.ini, /etc/php5/apache2/conf.d/20-readline.ini
PHP API	20121113
PHP Extension	20121212
Zend Extension	220121212
Zend Extension Build	API20121212.NTS
PHP Extension	API20121212.NTS

10. Finalmente se instala postgresql

Mediante los siguientes comandos:

```
root@DentalAccess:~# sudo apt-get update
```

```
root@DentalAccess:~# sudo apt-get install postgresql postgresql-contrib
```

Para poner clave al postgresql se pone los siguientes comandos:

```
root@DentalAccess:~# sudo -u postgres psql postgres
could not change directory to "/root": Permission denied
Password: 
```

Para crear una nueva base de datos se realiza lo siguiente:

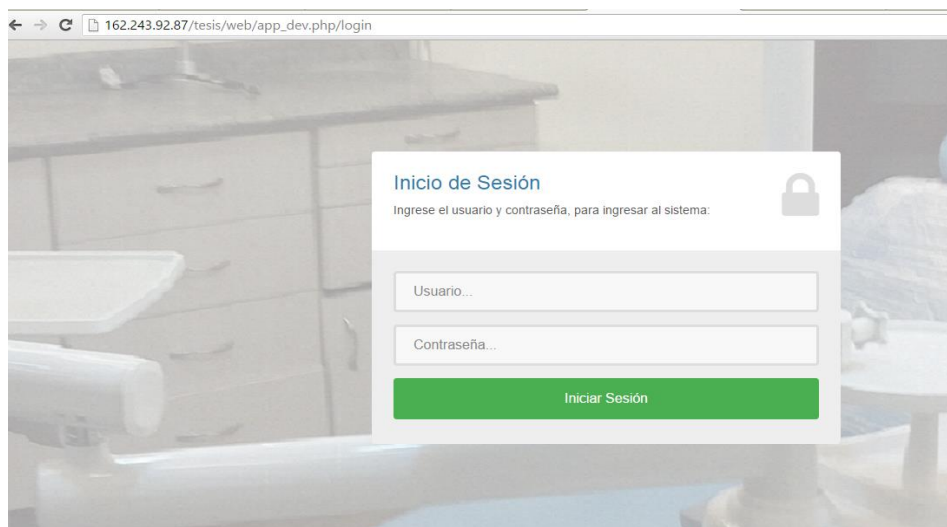
```
postgres=# sudo -u postgres createdb centrodontologico
```

Y se ejecuta el script de la base de datos que se encuentra en el CD.

11. Para instalar la aplicación se debe copiar el archivo tesis.zip, que se encuentra en el CD, en el directorio /var/www/html/

Pero lo primero se debe descomprimir es la carpeta en el directorio mencionado anteriormente mediante el comando: `sudo unzip tesis.zip`.

Después se puede probar que funciona el sistema en el URL:



Consideración adicional: la aplicación utiliza el Framework Symfony, que se encuentra en la carpeta de la aplicación, por lo cual no necesita descargarse ninguna librería adicional para el framework.

12. Configurar el servidor para enviar mail.

Primero se debe instalar el paquete mailutils

```
root@DentalAccess:~#  
root@DentalAccess:~#  
root@DentalAccess:~#  
root@DentalAccess:~# sudo apt-get install mailutils
```

Después se configura el postfix, ingresando al siguiente archivo:

```
root@DentalAccess:~#  
root@DentalAccess:~#  
root@DentalAccess:~# sudo nano /etc/postfix/main.cf
```

Y se modifica la parte final del archivo, para poder enviar mail:

```
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128  
mailbox_size_limit = 0  
recipient_delimiter = +  
inet_interfaces = loopback-only  
inet_protocols = all
```

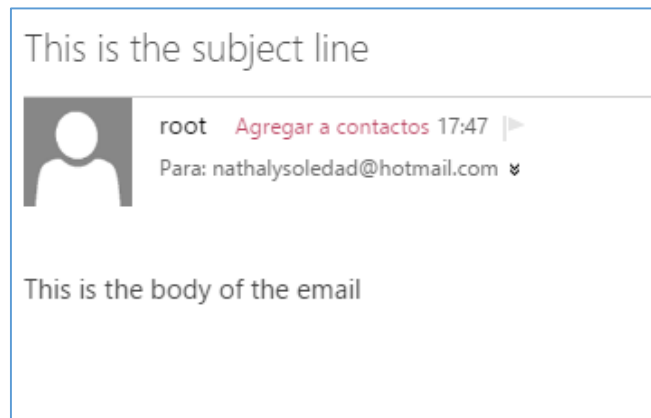
Para que los cambios sean efectuados se reinicia el postfix

```
root@DentalAccess:~#  
root@DentalAccess:~#  
root@DentalAccess:~#  
root@DentalAccess:~# sudo service postfix restart
```

Se prueba la configuración del mail con lo siguiente:

```
root@DentalAccess:~# echo "This is the body of the email" | mail -s "This is the  
subject line" nathalysoledad@hotmail.com  
root@DentalAccess:~#
```

Confirmar que llego al mail



Finalmente se puede usar el sistema correctamente, con la aplicación instalada, y la base de datos creada, Para el funcionamiento del sistema, se puede conocer en el manual de usuarios.

5.3. Glosario

A

Access
Pertenece a Microsoft office, el cual es un gestor de base de datos. 23

algoritmo
Es un conjunto de operaciones de forma secuencial, que permite resolver un problema. 16

Applet
Esta escrito en java, y se ejecuta como aplicación web..... 22

arquitectura distribuida
Es un conjunto de ordenadores autónomos, acoplado en una red y soportados por un software que permite que actué como un servicio integrado (Navarro & Márques, 2006) 29

C

C
Es un lenguaje de programación estructurado. 19

CASE
Ingeniería de software asistida por computador 6

ciclo de vida
es el periodo de ti empo que trascurre un producto desde su creación hasta la desaparición 7

CSS
es un lenguaje baso en hojas de estilo, para aplicaciones web. 21

D

DBMS
Un sistema de administración de base de datos... 24

declaraciones lógicas
Es una declaración, que está formada por dos condiciones de las cuales una depende de otra (Bueno saber, 2014) 18

diagrama de iteraciones
Muestra una iteración, que consta de un conjunto de objetos y sus relaciones incluyendo los mensajes que pueden ser enviados 15

diagramas
Es una representación gráfica que simplifica la comunicación y la información (El concepto de objeto, 2016) 11

Digita Ocean
Es un proveedor se servidores virtuales privados, ubicado principalmente en New York. 182

distribuida

es un conjunto de ordenadores autónomos, acoplado en una red y soportados por un software que permite que actué como un servicio integrado..... 29

Doctrine
Esta enfocada en el almacenamiento de la base de datos y mapeo de objetos. 28

droplet
Es un servidor virtual..... 182

F

Framework
Es un conjunto de componentes, herramientas, librerías, que facilita el desarrollo de la aplicaciones. 26

H

HTML
Es un lenguaje de programación que permite realizar aplicaciones web. 19

I

intranet
Es un conjunto de servicios de internet dentro de una red local (CCM, 2016). 18

IP
Es el protocolo de internet..... 185

iterativo
Se repite 8

J

JavaScript
Es un lenguaje de programación interpretado y orientado a objetos. 22

L

lenguajes de alto nivel
Permite al programador escribir instrucciones en palabras parecidas al lenguaje inglés (CARLOSPES.COM, s.f.)..... 16

M

mailutils

Es un paquete para instalar mail en el servidor,
independiente de protocolos. 187

método Booch
Es un diseño orientado a objetos de Grady Booch.
..... 14

métodos
son todas las actividades que el objeto puede
realizar (El concepto de objeto, 2016)..... 8

modelado
es la representación simplificada de la realidad
tomando en cuenta sus detalles principales
(EcuRed, s.f.)..... 8

modelos
Es una simplicidad de la realidad y proporciona los
planos del sistema. 13

mSQL
Es un sistema de gestión de base de datos,
opensource..... 23

MVC
Modelo, vista y controlador. 27

Mysql
Es un motor de base de datos, basado en un
lenguaje de consulta estructurado. 23

O

objetos
Es la representación de una cosa que consta de un
estado y comportamiento. 8

OMT
Es un modelado de objetos. 14

Oracle
Es una herramienta potente de base de datos,
basado en cliente/servidor..... 23

ORM
Object/ relation mapping. 28

P

Perl
Es un lenguaje de programación interpretado..... 19

PHP
Es un lenguaje interpretado del lado servidor,
diseñado para la web. 19

postfix
Es un servidor de correo electrónico basado en
software libre. 187

PostgreSQL

Es un sistema de gestión de base de datos,
orientada a objetos. 25

Propel
Utiliza un conjunto de objetos para acceder a la BD.
..... 28

S

sandbox
Permite evaluar el código. 28

servidor web
Es el que se encarga de realizar todas las funciones
en el lado del servidor y utiliza el protocolo HTTP
(Servidor Web, s.f.)..... 18

SGML
Lenguaje de marcado generalizado estándar. 21

software
es un programa o aplicación que se encarga de
realizar diferentes tareas o funciones y así
automatizar procesos..... 8

Symfony
Es un framework diseñado para facilitar el
desarrollo de aplicaciones web. 26

T

twig
Es una plantilla de Symfony. 28

U

Ubuntu
Es un sistema operativo basado en GNU/Linux. . 182

UML
Es un lenguaje que permite escribir planos del
software. 14

Un patrón de diseño
Es una descripción de clases y objetos..... 33

usuario
Es el encargado de realizar una función específica.
..... 11

W

W3C
World wide web consortium..... 21

Bibliografía

Libros:

- Booch Grady, G., Rumbaugh, J., & Jacobson, I. (2000). El lenguaje Unificado de Modelado. En G. Booch Grady, J. Rumbaugh, & I. Jacobson, *El lenguaje Unificado de Modelado* (págs. 3-11). España: Addison Wesley.
- Borja, A. (2015). *Curso práctico avanzado de PostgreSQL la base de datos más potente*. España: Publicaciones Altaria, S.L.
- Cabezas, L. M., & González, F. J. (2015). *Desarrollo Web con PHP y MYSQL*. Madrid: ANAYA MULTIMEDIA.
- Cachero, C., Ponce de León, P., & Saquete, E. (s.f.). Introducción a la programación orientada a objetos. En C. Cachero, P. Ponce de León, & E. Saquete, *Introducción a la programación orientada a objetos* (págs. 90-147). Universidad de Alicante.
- Coronel, C., Morris, S., & Rob, P. (2011). *Bases de datos Diseño, implementación y administración*. Mexico DF.: CENGAGE Learning.
- Instituto Nacional de la Tecnología de la comunicación. (Marzo de 2009). INGENIERÍA DEL SOFTWARE: METODOLOGÍAS Y CICLOS DE VIDA. España.
- Kolbeck, R. (1997). *El gran Libro de JavaScript*. (M. Sánchez Pérez, Ed., J. Arroyo, J. Cortez, N. González, & P. Villena, Trads.) Barcelona: MARCOMBO.
- Serna, J. A. (s.f.). *REPORTE 1 INVESTIGACION Y FUNDAMENTO DE LA METODOLOGIA "RUP"*. UAM I.
- Tucker, A., & Noonan, R. (2003). *Lenguajes de Programación Pincipios y paradigmas*. Madrid: McGraw-Hill.
- Welling, L., & Thomson, L. (2005). *Desarrollo Web con PHP*. Madrid: Anaya Multimedia.
- López, L. (2011). Progración estructurada y orientada a objetos. En L. López, *Progración estructurada y orientada a objetos* (pág. 384). México: Alfaomega.

Artículos:

- Goñi's, A. (s.f.). *Reutilización del Software*. Obtenido de Patrones de Diseño: <http://siul02.si.ehu.es/~alfredo/iso/06Patrones.pdf>
- Kimble, C. (Febrero de 2008). *Object Oriented Methodologies*. Obtenido de <http://www.chris-kimble.com/Courses/sdm/Presentations/SDM6.pdf>

- Orallo, E. H. (s.f.). *El Lenguaje Unificado de Modelado (UML)*. Obtenido de <http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>
- Márquez, B. M., & Zulaica, J. M. (Enero de 12 de 2004). *mplementación de un reconocedor de voz gratuito a el sistema de ayuda a invidentes Dos-Vox en español*. Obtenido de Capítulo 5. Cliente-Servidor.: http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/marquez_a_bm/capitulo5.pdf
- *Rational Unified Process (RUP)*. (s.f.). Obtenido de <http://ima.udg.edu/~sellares/EINF-ES2/Present1011/MetodoPesadesRUP.pdf>
- (s.f.). *Tema4: Concepto de Metodologías. Metodología Estructurada* (<http://www.infor.uva.es/~jvalvarez/docencia/ptema4is1.pdf>). Segovia: Universidad de Valladolid.
- Universidad de Valladolid. (s.f.). *Conceptos de Metodología. Metodología Estructurada*. Obtenido de <http://www.infor.uva.es/~jvalvarez/docencia/ptema4is1.pdf>
- Universidad Politecnica de Valencia. (12 de Noviembre de 2003). *Metodologías Ágiles en el Desarrollo del Software*. Obtenido de <http://issi.dsic.upv.es/archives/f-1069167248521/actas.pdf>

Internet

- .net y punto. (6 de Junio de 2011). *ASP.NET MVC: Introducción*. Obtenido de <https://pontonetypunto.wordpress.com/2011/06/16/asp-net-mvc-introduccion/>
- 2ndQuadrant. (s.f.). *PostgreSQL*. Obtenido de <http://2ndquadrant.com/es/postgresql/>
- 2ndQuadrant. (s.f.). *PostgreSQL – La historia hasta ahora*. Obtenido de <http://2ndquadrant.com/es/postgresql/postgresql-la-historia-hasta-ahora/>
- Alvarez, M. A. (9 de Mayo de 2001). *Que es PHP*. Obtenido de <http://www.desarrolloweb.com/articulos/392.php>
- Alvarez, M. A. (02 de enero de 2014). *Qué es MVC*. Obtenido de <http://www.desarrolloweb.com/articulos/que-es-mvc.html>
- Asamiya, K. (28 de Noviembre de 2011). *Metodología RUP*. Obtenido de Ciclo de vida: fases y disciplina: <http://rupisw.blogspot.com/2011/11/ciclo-de-vida-las-fases-del-ciclo-de.html>
- *Bueno saber*. (2014). Obtenido de ¿Qué es una declaración de lógica?: <http://bueno-saber.com/aficiones-juegos-y-juguetes/ciencia-y-naturaleza/que-es-una-declaracion-de-logica.php>
- Burbach, R. L. (1998). *Modelado de objetos Técnica (OMT)*. Obtenido de <http://infolab.stanford.edu/~burbach/watersluice/node56.html>

- Cárdenas Rovira, J. (05 de Mayo de 2012). *Ciclo de vida del producto*. Obtenido de Gestipolis: <http://www.gestipolis.com/ciclo-de-vida-del-producto/>
- CARLOSPES.COM. (s.f.). *Definición de Lenguaje de Alto Nivel*. Obtenido de Carlospes.com: http://www.carlospes.com/minidiccionario/lenguaje_de_alto_nivel.php
- CCM. (Enero de 2016). *Entorno cliente/servidor*. Obtenido de <http://es.ccm.net/contents/148-entorno-cliente-servidor>
- CCM. (Mayo de 2016). *Intranet y extranet*. Obtenido de CCM: <http://es.ccm.net/contents/213-intranet-y-extranet>
- Cobo, Á., Gómez, P., Pérez, D., & Rocha, R. (2005). *PHP y MYSQL: Tecnología para el desarrollo de aplicaciones web*. España: Ediciones Díaz de Santos. Obtenido de https://books.google.com.ec/books?id=zMK3GOMOpQ4C&printsec=frontcover&hl=es&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false
- Concepto.de. (s.f.). *Qué es un diagrama*. Obtenido de Concepto.de: <http://concepto.de/diagrama/>
- Díaz Yicela, S. D. (21 de Noviembre de 2010). *Estrategias Metodológicas*. Obtenido de METODOLOGÍA INFORMÁTICA: <http://estrategiasmetodologicasinformaticas.blogspot.com/2011/01/metodologia-informatica.html>
- Doctrine. (2006-2016). *Doctrine*. Obtenido de Welcome to the Doctrine Project: <http://www.doctrine-project.org/>
- EcuRed. (s.f.). *Arquitectura Cliente servidor*. Obtenido de Arquitectura Cliente servidor: http://www.ecured.cu/Arquitectura_Cliente_Servidor
- Ecured. (s.f.). *Arquitectura de tres niveles*. Obtenido de http://www.ecured.cu/Arquitectura_de_tres_niveles
- EcuRed. (s.f.). *Modelado*. Obtenido de EcuRed: <http://www.ecured.cu/Modelado>
- Eguiluz, J. (s.f.). *Introducción a JavaScript*. Obtenido de Libros Web: http://librosweb.es/libro/javascript/capitulo_1/breve_historia.html
- Eguiluz, J. (s.f.). *Introduccion CSS*. Obtenido de Libros web: https://librosweb.es/libro/css/capitulo_1.html
- Eguiluz, J. (s.f.). *Libros web*. Obtenido de Capítulo 5. Frameworks: http://librosweb.es/libro/css_avanzado/capitulo_5.html
- *El concepto de objeto*. (Mayo de 2016). Obtenido de CCM: <http://es.ccm.net/contents/412-el-concepto-de-objeto>
- Expermicid. (30 de Noviembre de 2013). *Modelo Vista Controlador*. Obtenido de UNDERCODE: <https://undercode.org/foro/java/modelo-vista-controlador/>

- Flores López, L., & Santiago, M. (s.f.). *Proceso de desarrollo unificado (RUP)*. Obtenido de <http://www.utvm.edu.mx/OrganoInformativo/orgJul07/RUP.htm>
- Gama, A. (31 de Enero de 2010). *Arquitectura*. Obtenido de <https://ulyla003.files.wordpress.com/2010/02/imagen3.png>
- Gonzales, A. (s.f.). *¿Qué es un ORM?* Obtenido de Glosario: <http://www.tuprogramacion.com/glosario/que-es-un-orm/>
- (2008). <https://ifs.host.cs.st-andrews.ac.uk/Books/SE9/Web/Structured-methods/>. Sommerville: Ia.
- Javier. (20 de Enero de 2014). *La Metodología RUP, Ventajas Y Desventajas*. Obtenido de <https://www.clubensayos.com/Tecnolog%C3%ADa/La-Metodologia-RUP-Ventajas-Y-Desventajas/1391437.html>
- Kioskea .net. (Julio de 2014). *¿Qué es una base de datos?* Obtenido de <http://es.ccm.net/contents/66-introduccion-bases-de-datos>
- Leake, A., & Rouse, M. (2005). *Database Definition*. Obtenido de <http://searchsqlserver.techtarget.com/definition/database>
- Luján Mora, S. (2002). *Programación de Aplicaciones web: Historia, Pirncipios básicos y Clientes Web*. gamma. Obtenido de https://books.google.com.ec/books?id=r9CqDYh2-loC&pg=PR16&dq=arquitectura+de+aplicaciones&hl=es&sa=X&redir_esc=y#v=onepage&q=arquitectura%20de%20aplicaciones&f=false
- Machala, U. T. (2010 de Noviembre de 21). *Estratégias Metodológicas*. Obtenido de Metodología Informática: <http://estrategiasmetodologicasinformaticas.blogspot.com/2011/01/metodologia-informatica.html>
- Masip, D. (19 de Julio de 2002). *Qué es Oracle*. Obtenido de Desarrolloweb: <http://www.desarrolloweb.com/articulos/840.php>
- Melder, M. (9 de Noviembre de 2014). *Metodología Estructurada*. Obtenido de <http://es.slideshare.net/MelderMelendez/metodologia-estructurada-41333655>
- *Metodologías ágiles*. (s.f.). Obtenido de http://www.umsl.edu/~sauterv/analysis/6840_f09_papers/Nat/Agile.html
- *mvc – model view controller (patrón de diseño)*. (2 de Octubre de 2013). Obtenido de <https://ingeniods.wordpress.com/2013/10/02/mvc-model-view-controller-patron-de-diseno/>
- Navarro, L., & Márques, J. M. (Junio de 2006). *Arquitectura de Sistemas Distribuidos*. Obtenido de <http://bernal.pro/informatica/apuntesdeinformatica/finish/6-apuntes-de-informatica/63-arquitectura-d>

- NewComLab. (s.f.). *Modelo de tres capas*. Obtenido de http://www.newcomlab.com/default.aspx?id_seccion=936
- Potencier, F., & Zaninotto, F. (s.f.). *Symfony 1.4, la guía definitiva*. Obtenido de Libro Web: http://librosweb.es/libro/symfony_1_4/
- *Programación Concurrente*. (s.f.). Obtenido de <https://www.fing.edu.uy/tecnoinf/mvd/cursos/so/material/teo/so07-concurrencia.pdf>
- *Programacion Orientada a Objetos*. (12 de febero de 2014). Obtenido de Prezi: <https://prezi.com/iamk-ws25aid/programacion-orientada-a-objetos/>
- Propel org. (s.f.). *Propel* . Obtenido de A highly customizable and blazing fast ORM library for PHP 5.5+.: <http://propelorm.org/>
- Rational Unified Process equipo 1. (2012 de Diciembre de 2012). *Metodología RUP*. Obtenido de <http://rupequipo1.blogspot.com/2012/12/algo-de-historia.html>
- Rouse, M. (s.f.). *MYSQL*. Obtenido de techtarget: <http://searchdatacenter.techtarget.com/es/definicion/MySQL>
- Rubenfa. (14 de Julio de 2014). *Patrones de diseño: qué son y por qué debes usarlos*. Obtenido de <http://www.genbetadev.com/metodologias-de-programacion/patrones-de-diseno-que-son-y-por-que-debes-usarlos>
- sensiolabs. (2010). *Twig*. Obtenido de The flexible, fast, and secure: <http://twig.sensiolabs.org/>
- *Servidor Web*. (s.f.). Obtenido de EcuRed: http://www.ecured.cu/Servidor_Web
- Symfony. (s.f.). *What is Symfony*. Obtenido de <https://symfony.com/what-is-symfony>
- Universidad Simón Bolívar. (Abril-Junio de 2011). *Arquitectura de 3 capas ventajas y deventajas*. Obtenido de https://www.assembla.com/spaces/tecsolutions/documents/bjw_omGNSr4iHAeJe5cbLr/download/bjw_omGNSr4iHAeJe5cbLr
- Universidad Técnica de Machala. (21 de Noviembre de 2010). *Estrategias Metodológicas*. Obtenido de Metodología Informática: <http://estrategiasmetodologicasinformaticas.blogspot.com/2011/01/metodologia-informatica.html>
- w3schools. (s.f.). *CSS Tutorial*. Obtenido de CSS: http://www.w3schools.com/css/css_intro.asp