



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E
INDUSTRIAL**

**CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES E
INFORMÁTICOS**

Tema:

IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO APLICANDO EL
FRAMEWORK ANGULAR PARA LA GESTIÓN DE INFORMACIÓN Y CONTROL
DE CITAS EN EL CONSULTORIO DENTAL ODONTÓLOGO JUAN CARLOS
LEÓN.

Trabajo de Titulación Modalidad: Proyecto de Investigación, presentado previo a la
obtención del título de Ingeniero en Sistemas Computacionales e Informáticos

ÁREA: Software

LÍNEA DE INVESTIGACIÓN: Desarrollo de software

AUTOR: Diego Fernando Galindo Vásquez

TUTOR: Ing. Julio Enrique Balarezo López, PhD

Ambato -Ecuador

agosto 2024

APROBACIÓN DEL TUTOR

En calidad de tutor del Trabajo de Titulación con el tema: IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO APLICANDO EL FRAMEWORK ANGULAR PARA LA GESTIÓN DE INFORMACIÓN Y CONTROL DE CITAS EN EL CONSULTORIO DENTAL ODONTÓLOGO JUAN CARLOS LEÓN, desarrollado bajo la modalidad Proyecto de Investigación, por el señor Diego Fernando Galindo Vásquez, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 15 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y el numeral 7.4 del respectivo instructivo.

Ambato, agosto 2024.



Ing. Julio Enrique Balarezo López, PhD.

TUTOR

AUTORÍA

El presente proyecto de investigación titulado: IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO APLICANDO EL FRAMEWORK ANGULAR PARA LA GESTIÓN DE INFORMACIÓN Y CONTROL DE CITAS EN EL CONSULTORIO DENTAL ODONTÓLOGO JUAN CARLOS LEÓN, es absolutamente original, auténtico y personal. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo, son de exclusiva responsabilidad del autor.

Ambato, agosto 2024



Diego Fernando Galindo Vásquez

C.C: 1805064266

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos del presente Trabajo de Titulación en favor de la Universidad Técnica de Ambato, con fines de difusión pública. Además, autorizo su reproducción total o parcial dentro de las regulaciones de la institución.

Ambato, agosto 2024



Diego Fernando Galindo Vásquez

C.C: 1805064266

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del Informe Final del Trabajo de Titulación presentado por el señor Diego Fernando Galindo Vásquez, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO APLICANDO EL FRAMEWORK ANGULAR PARA LA GESTIÓN DE INFORMACIÓN Y CONTROL DE CITAS EN EL CONSULTORIO DENTAL ODONTÓLOGO JUAN CARLOS LEÓN, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 17 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y al numeral 7.6 del respectivo instructivo. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal

Ambato, agosto 2024



Firmado electrónicamente por:
KELSA PILAR URRUTIA

Ing. Pilar Urrutia, Mg.

PRESIDENTE DEL TRIBUNAL



Firmado electrónicamente por:
**LEONARDO DAVID
TORRES VALVERDE**

Ing. Leonardo Torres Valverde, Mg.

PROFESOR CALIFICADOR



Firmado electrónicamente por:
**PABLO ISRAEL
MORALES PAREDES**

Ing. Pablo Morales Paredes, Mg.

PROFESOR CALIFICADOR

DEDICATORIA

El presente trabajo lo dedico a mis padres Carlos Galindo y Sandra Vásconez por su tiempo, amor y sacrificio incondicional a lo largo de toda mi vida logrando que pueda cumplir mi primera meta.

A mis hermanos Leonardo y Alejandro por ser mi mayor motivación y pilares en los que me apoye en los momentos más difíciles.

A mi primo Juan por brindarme una segunda oportunidad, aconsejarme y ser un ejemplo para mí.

A mi abuelo Leonardo por su enseñanzas de perseverancia, resiliencia y fortaleza.

A mis tíos Carlos, Miriam, Mónica y Óscar por abrirme las puertas de su hogar y su corazón siempre.

AGRADECIMIENTO

A mi tutor Julio Balarezo por ser el faro que me ayudo a materializar este proyecto de titulación con sus conocimientos. Mostrando que además de ser un gran docente es aún mejor persona.

Al Ingeniero Clay Aldas por la ayuda y preocupación que me mostro a lo largo de la carrera. Siendo un amigo para mí.

A Gabriela Flores por ser una amiga, brindarme pronta ayuda y mostrar su preocupación sincera.

A los docentes Pablo Morales, Mayorga Franklin, David Guevara y Edison Álvarez. Que mostraron ser profesionales que admiro, respeto y aprecio. Por su manera de transmitir conocimiento y tratar a las personas.

ÍNDICE DE CONTENIDOS

CAPÍTULO I: MARCO TEÓRICO	1
1.1 Tema de investigación.....	1
1.2 Antecedentes investigativos	1
1.2.1 Contextualización del problema	1
1.2.2 Antecedentes investigativos.....	3
1.2.3 Fundamentación teórica.....	5
1.2.3.1 Desarrollo de Software.....	5
1.2.3.2 Sistema Web.....	5
1.2.3.3 Tipos de sistema web.....	5
1.2.3.4 Arquitectura de un sistema web.....	6
1.2.3.5 Motores de Bases de datos	7
1.2.3.6 Servicio web	9
1.2.3.7 Backend	10
1.2.3.8 Frontend.....	11
1.2.3.9 Metodologías de desarrollo	14
1.2.3.10 Odontología.....	15
1.2.3.11 Manejo de Historias Clínicas.....	15
1.2.3.12 Historia Clínica Única	16
1.2.3.13 Historia Clínica activa.....	16
1.2.3.14 Historia Clínica inactiva	16
1.3 Objetivos	16
1.3.1 Objetivo general.....	16
1.3.2 Objetivos específicos	17
CAPÍTULO II: METODOLOGÍA.....	18
2.1 Materiales.....	18
2.1.1 Institucionales.....	18
2.1.2 Humanos.....	18
2.1.3 Económicos	18
2.2 Métodos	21

2.2.1	Modalidad de investigación.....	21
2.2.2	Población y muestra.....	22
2.2.3	Recolección de la información	23
2.2.4	Procesamiento y análisis de datos.....	31
CAPÍTULO III.....		33
3.1	Desarrollo de la propuesta.....	33
3.1.1	Metodología de desarrollo de la tesis	33
3.1.1.1	Levantamiento de requerimientos con la IEEE 830.....	33
3.1.1.2	Definición de arquitectura	46
3.1.1.3	Metodología Desarrollo.....	47
3.1.1.4	Herramientas	50
3.1.1.5	Ejecución	55
CAPÍTULO IV		124
4.1	Conclusiones	124
4.2	Recomendaciones	125
BIBLIOGRAFÍA.....		126

ÍNDICE DE TABLAS

Tabla 1: Recursos económicos.....	18
Tabla 2: Estructura de la encuesta realizada al personal de salud.....	20
Tabla 3: Población.....	23
Tabla 4: Recolección de la información.....	23
Tabla 5: Pregunta 1 – Entrevista	24
Tabla 6: Pregunta 2 – Entrevista	25
Tabla 7: Pregunta 3 – Entrevista	25
Tabla 8: Pregunta 4 - Entrevista.....	26
Tabla 9: Pregunta 5 - Entrevista.....	26
Tabla 10: Pregunta 6 - Entrevista.....	27
Tabla 11: Pregunta 7 - Entrevista.....	27
Tabla 12: Pregunta 8 - Entrevista.....	28
Tabla 13: Pregunta 9 - Entrevista.....	28
Tabla 14: Pregunta 10 - Entrevista.....	28
Tabla 15: Pregunta 11 - Entrevista.....	29
Tabla 16: Pregunta 12 - Entrevista.....	29
Tabla 17: Pregunta 13 - Entrevista.....	30
Tabla 18: Pregunta 14 - Entrevista.....	30
Tabla 19: Pregunta 15 - Entrevista.....	30
Tabla 20: Pregunta 16 - Entrevista.....	31
Tabla 21: Características usuario administrador	37
Tabla 22: Características usuario secretaria	38
Tabla 23: Características usuario administrador	38
Tabla 24: Requisito gestionar Historias Clínicas	40
Tabla 25: Requisito gestionar pacientes.....	40
Tabla 26: Requisito gestionar citas	41
Tabla 27: Requisito Iniciar sesión.....	41
Tabla 28: Requisito gestionar usuarios	42
Tabla 29: Requisito gestionar roles	42

Tabla 30: Requisito gestionar especialidades	42
Tabla 31: Comparativa metodologías de desarrollo.....	48
Tabla 32: Características metodologías seleccionadas	49
Tabla 33: Comparativa tecnología frontend.....	50
Tabla 34: Comparativa tecnología backend.....	52
Tabla 35: Comparativa tecnología Base de Datos	54
Tabla 36: Plantilla utilizada para historia de usuario	58
Tabla 37: Historia de usuario 1	59
Tabla 38: Historia de usuario 2	59
Tabla 39: Historia de usuario 3	60
Tabla 40: Historia de usuario 4	60
Tabla 41: Historia de usuario 5	61
Tabla 42: Historia de usuario 6	62
Tabla 43: Historia de usuario 7	62
Tabla 44: Historia de usuario 8	63
Tabla 45: Historia de usuario 9	63
Tabla 46: Historia de usuario 10	64
Tabla 47: Historia de usuario 11	65
Tabla 48: Resumen requisitos funcionales junto a sus historias de usuario	65
Tabla 49: Resumen historias de usuario sin requisito funcional.....	66
Tabla 50: Plantilla Tarea	67
Tabla 51: Tarea 1	67
Tabla 52: Tarea 2	68
Tabla 53: Tarea 3	68
Tabla 54: Tarea 4	69
Tabla 55: Tarea 5	69
Tabla 56: Tarea 6	69
Tabla 57: Tarea 7	70
Tabla 58: Tarea 8	70
Tabla 59: Tarea 9	71
Tabla 60: Tarea 10	71

Tabla 61: Tarea 11	71
Tabla 62: Tarea 12	72
Tabla 63: Tarea 13	72
Tabla 64: Tarea 14	73
Tabla 65: Tarea 15	73
Tabla 66: Tarea 16	73
Tabla 67: Tarea 17	74
Tabla 68: Tarea 18	74
Tabla 69: Tarea 19	75
Tabla 70: Tarea 20	75
Tabla 71: Tarea 21	75
Tabla 72: Tarea 22	76
Tabla 73: Tarea 23	76
Tabla 74: Tarea 24	77
Tabla 75: Tarea 25	77
Tabla 76: Tarea 26	77
Tabla 77: Tarea 27	78
Tabla 78: Tarea 28	78
Tabla 79: Tarea 29	79
Tabla 80: Tarea 30	79
Tabla 81: Tarea 31	79
Tabla 82: Tarea 32	80
Tabla 83: Tarea 33	80
Tabla 84: Tarea 34	81
Tabla 85: Tarea 35	81
Tabla 86: Resumen de Historias de usuario	82
Tabla 87: Resumen tareas	83
Tabla 88: Resumen iteración 1	85
Tabla 89: Resumen iteración 2	91
Tabla 90: Resumen iteración 4	101
Tabla 91: Resumen iteración 5	104

Tabla 92: Formato prueba de aceptación	114
Tabla 93: Prueba de aceptación 1.....	115
Tabla 94: Prueba de aceptación 2.....	116
Tabla 95: Prueba de aceptación 3.....	117
Tabla 96: Prueba de aceptación 4.....	118
Tabla 97: Prueba de aceptación 5.....	119
Tabla 98: Prueba de aceptación 6.....	120
Tabla 99: Prueba de aceptación 7.....	122
Tabla 100: Resumen de requisitos funcionales con sus pruebas de aceptación.....	123

ÍNDICE DE FIGURAS

Figura 1. Arquitectura de un sistema web.....	7
Figura 2: Arquitectura de Angular	12
Figura 3: Arquitectura de Vue JS.....	13
Figura 4: Arquitectura de la aplicación web	47
Figura 5: Diagrama Metodología RAD.....	49
Figura 6: Fragmento de la base de datos	87
Figura 7: Estructura del proyecto en el cliente.....	88
Figura 8: Estructura del proyecto en el servidor	89
Figura 9: Logos más utilizados del Consultorio Dental.....	90
Figura 10: Paleta de colores del sistema	90
Figura 11: Pantalla Iniciar Sesión	92
Figura 12: HTML de pantalla Iniciar Sesión	92
Figura 13: Clase de JWT.....	93
Figura 14: Método login del servidor 1/2	94
Figura 15: Método login del servidor 2/2	94
Figura 16: Interfaz de gestión de usuarios	95
Figura 17: Creación del objeto con los datos del formulario	96
Figura 18: Consumo de servicios para manipulación de datos de usuario	97
Figura 19: Resumen iteración 3	97
Figura 20: Interfaz de roles	98
Figura 21: Consumo de servicios para manipulación de roles.....	99
Figura 22: Diseño interfaz odontólogos.....	100
Figura 23: Consumo de servicios para manipulación de odontólogos.....	101
Figura 24: Interfaz de pacientes	102
Figura 25: Creación del objeto con los datos del formulario	102
Figura 26: Consumo de servicios para manipulación de pacientes.....	103
Figura 27: Interfaz Historia Clínica 1/6	104

Figura 28: Interfaz Historia Clínica 2/6	105
Figura 29: Interfaz Historia Clínica 3/6	106
Figura 30: Interfaz Historia Clínica 4/6	106
Figura 31: Interfaz Historia Clínica 5/6	107
Figura 32: Interfaz Historia Clínica 6/6	107
Figura 33: Creación del objeto con los datos del formulario 1/2	108
Figura 34: Creación del objeto con los datos del formulario 2/2	109
Figura 35: Consumo de servicios para manipulación de Historias Clínicas	110
Figura 36: Interfaz Citas.....	111
Figura 37: Fragmento código HTML.....	112
Figura 38: Fragmento creación de eventos del calendario.....	113

RESUMEN EJECUTIVO

En los últimos años, la sociedad ha experimentado un rápido progreso tecnológico, lo cual ha permitido resolver tareas manuales repetitivas, automatizar procesos y almacenar información de manera eficiente. La salud, como una industria crucial a nivel global, debe integrar estas nuevas tecnologías para ofrecer servicios automatizados mejorados y asegurar el manejo seguro de la información. Por lo tanto, es fundamental adoptar historias clínicas digitalizadas para registrar y monitorear la información vital de los pacientes de manera efectiva.

El presente proyecto de investigación tiene como objetivo el desarrollo de una aplicación web orientada a resolver el inconveniente que genera el manejo en papel de historias clínicas y citas odontológicas, permitiendo el agendamiento de citas, gestión de pacientes, usuarios y odontólogos. De esta manera agilizar el proceso de búsqueda de los archivos únicos de pacientes junto a la información de cada paciente. Además de ordenar las citas odontológicas para evitar el cruce de horarios y evitar tiempos de espera largos en el establecimiento.

La metodología utilizada en el proyecto fue una combinación entre las bondades de la metodología ágil RAD, la que permitió entregas permanentes de software al cliente para una retroalimentación constante con el usuario final y las bondades de historias de usuario que nos presenta Kanban para ejecución de tareas que pasan por distintas fases: pendiente, en curso y finalizada. Se asignaron las iteraciones con sus tareas correspondientes diseñando la paleta de colores de la interfaz, la base de datos y la aplicación. Las tecnologías utilizadas en el presente proyecto son en el cliente Angular y para el servidor NodeJS.

Palabras clave: Aplicación web progresiva, Angular, metodologías de desarrollo

ABSTRACT

In recent years, society has experienced rapid technological progress, enabling the resolution of repetitive manual tasks, automation of processes, and efficient information storage. Healthcare, as a critical global industry, must integrate these new technologies to offer enhanced automated services and ensure secure information management. Therefore, adopting digital medical records to effectively register and monitor vital patient information is crucial.

The aim of this research project is to develop a web application focused on addressing the challenges posed by paper-based management of dental records and appointments. This includes scheduling appointments, managing patients, users, and dentists. This system aims to streamline the search process for patient records and their associated information. Additionally, it organizes dental appointments to prevent scheduling conflicts and reduce waiting times at the establishment.

The methodology employed in the project combined the strengths of Agile RAD methodology, facilitating continuous software deliveries to the client for ongoing user feedback, with Kanban's user story advantages for task execution across phases: backlog, in progress, and completed. Iterations were assigned with corresponding tasks, covering interface design, database development, and application implementation. The technologies utilized include Angular for client-side development and NodeJS for server-side operations.

Keywords: Progressive web application, Angular, development methodologies.

CAPÍTULO I: MARCO TEÓRICO

1.1 Tema de investigación

IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO APLICANDO EL FRAMEWORK ANGULAR PARA LA GESTIÓN DE INFORMACIÓN Y CONTROL DE CITAS EN EL CONSULTORIO DENTAL ODONTÓLOGO JUAN CARLOS LEÓN.

1.2 Antecedentes investigativos

1.2.1 Contextualización del problema

En los últimos años la sociedad ha sufrido un avance acelerado en el campo de la tecnología, logrando brindar soluciones a tareas manuales recurrentes, automatización de procesos y almacenamiento de la información. La salud es una de las industrias más importantes a nivel mundial, por ese motivo debe adaptarse a las nuevas tecnologías para brindar un mejor servicio automatizado y de esta manera mejorar el tratamiento de la información de forma segura. Por tanto, es necesario que la información vital y seguimiento de los pacientes se realice utilizando Historias Clínicas digitalizadas [1].

Las Historias Clínicas son generadas en papel lo que trae consigo problemas para los profesionales de la salud, pues el realizar la búsqueda de registros no solo es tediosa e incómoda, sino también retrasa el trabajo en los establecimientos de salud donde es común que se almacene la información en grandes cuartos, en este lugar a lo largo del tiempo se acumulan gran cantidad de Historias Clínicas, apiladas en cajas hasta ser desechadas, por

esto los documentos corren el riesgo de arrugarse, desgastarse o dañarse poniendo así en peligro la información vital de los pacientes [1].

A pesar de que la tecnología presenta muchas opciones que solventan las debilidades de las Historias Clínicas no digitalizadas, el campo de la salud es resiliente al cambio tecnológico como menciona Delgado [2] en su estudio donde indica que “El crecimiento de las Historias Clínicas automatizadas es un fenómeno mundial, aunque su avance es relativamente lento, pues constituye un cambio muy significativo en cuanto al trabajo de la asistencia médica”. Uno de los campos de la salud que presenta más resistencia es la odontología, donde el software disponible del mercado ha sido enfocado desde un punto de vista general pasando por alto las necesidades específicas que posee la odontología, donde el manejo de las Historias Clínicas no hace énfasis en la Ficha Clínica Única de Odontología, que es utilizada para representación de odontogramas, tratamientos o diagnósticos por sesión. Por tanto, el manejo de Historias Clínicas es solo una parte del problema, ya que con el paso del tiempo el número de profesionales de odontología ha crecido, lo que ha dado origen a muchos consultorios particulares en los cuales son muy comunes dos problemas: el primero es reservar una cita, y el segundo el largo tiempo de espera para recibir la atención [3].

El Consultorio Dental “Odontólogo Juan Carlos León” de Ambato no es la excepción, donde los pacientes solicitan permiso en sus trabajos o se dan un tiempo en el día para agendar una cita, y esperar su turno, esto genera una pérdida. Es por lo que se plantea una aplicación híbrida para gestión de la información y agendamiento de citas que brinde agilidad a los procesos a la vez que dé fácil acceso a la información.

1.2.2 Antecedentes investigativos

La sociedad en la actualidad ha sufrido cambios en muchos niveles debido a los avances en los distintos campos como la construcción, medicina y tecnología. Sin embargo, el ámbito de la tecnología es el último de los campos a tomar en cuenta por el sector de la salud, donde se observan edificios con la última tecnología de construcción, pero los procesos aún son llevados de una manera precaria. Según Corilla [4] “La práctica odontológica debe afrontar la mitigación de los problemas no solo desde el punto de vista estructural, sino también en relación con los planes de futuro donde la tecnología y la innovación están cobrando protagonismo cada día.” Las organizaciones recurren ampliamente a las aplicaciones web con objetivos comerciales, ya que estas facilitan la mejora de sus procesos mediante la automatización, posibilitan el acceso a información desde cualquier ubicación, ofrecen compatibilidad con diversos dispositivos y proporcionan otros beneficios que añaden valor al negocio. A nivel mundial, se observa un aumento gradual en la adopción de Historias Clínicas automatizadas, si bien el progreso es gradual debido a los cambios disruptivos que implica en la práctica médica. La migración de Historias Clínicas en formato papel hacia versiones informatizadas responde a diversas necesidades, como la optimización del espacio de almacenamiento, la agilización de la recuperación de información y el fortalecimiento de la seguridad y estabilidad de los datos [5].

Según Corilla [4] en su proyecto de titulación “Desarrollo de un Sistema Web Para Mejorar La Gestión De Historias Clínicas En El Consultorio Dental Odontostetic, Abancay - Perú” realizado en 2021 en la Universidad Tecnológica de Los Andes en Abancay indica que generar un sistema centralizado, que facilite el agendamiento de citas y generación de consultas médicas odontológicas logra una optimización de alrededor del cincuenta por ciento en el ingreso de datos, reduciendo el ingreso de una Historia Clínica de 9.75 min a 4.35 min mejorando el entorno de la Clínica Odontológica ODONTOSTETIC.

Dentro de las herramientas que existen para el desarrollo de aplicaciones web se encontró Angular un framework que se utiliza para proyectos software que escalan a nivel empresarial, por la facilidad en su manejo de componentes y características como precarga de datos de una vista, “Interceptables” para gestionar todas las peticiones de HTTP, y los Guards que proporcionan un nivel de seguridad protegiendo vistas a través del acceso de las URLs a usuarios no autorizados ha provocado que sea utilizado en diseño de aplicaciones web en el ámbito médico, según Alava [6] en su tesis “Diseño e Implementación de un Sistema Informático para el control de Historias Clínicas aplicando tecnología QR en el Patronato del cantón La Maná” realizado en Agosto de 2022 en Latacunga indica que para cumplir con el objetivo establecido, se empleó un conjunto de herramientas de desarrollo de software, entre las que se incluyen la metodología de modelo en cascada, así como una variedad de lenguajes de programación y frameworks como JavaScript, PHP, Codeigniter y Angular. Además, se utilizó el sistema de gestión de bases de datos MySQL. Esta combinación de herramientas no solo aportó beneficios al proyecto, sino que también simplificó su desarrollo de manera notable. Dado que angular ofrece seguridades y manejo seguro de formularios a través de sus ReactiveForms que logran que la información no sufra cambios si el DOM es alterado logrando que el flujo de la información sea unidireccional sin poner en riesgo la integridad del aplicativo.

De la misma manera una ventaja es la integración de angular con otras herramientas según Almache [7] en su tesis “Sistema Web utilizando tecnología Angular para aplicar protocolos de bioseguridad en el registro y control de ganado del Centro de Faenamiento E.T.” publicada en febrero de 2022 en la Universidad Técnica de Ambato concluye que Angular, al ser una herramienta de código abierto, facilita la detección y corrección rápida de errores, ya que proporciona acceso a una amplia comunidad donde los desarrolladores pueden encontrar ayuda para resolver diversas problemáticas que puedan surgir durante el desarrollo del proyecto. Esto promueve una mejor comunicación entre los miembros del equipo de programadores, además de permitir la depuración del código, pues está escrito en TypeScript un código de tipado fuerte basado en javascript.

1.2.3 Fundamentación teórica

1.2.3.1 Desarrollo de Software

El desarrollo de software implica más que solo describirlo; implica construirlo. Esta perspectiva justifica la visión de la actividad de desarrollo de software como una forma de ingeniería. Además, en un contexto más amplio, es evidente que hay una conexión directa entre el software y su entorno, ya que el software se implementa en el mundo para producir ciertos efectos en él [8].

1.2.3.2 Sistema Web

Un sistema web o aplicación web, es un software con conexión a una base de datos que permite visualizar, editar y procesar información. Además, se encuentra alojado en un servidor web por lo que brinda disponibilidad para el cliente a través de la internet o una red local. Tiene beneficios en costo de implantación, rapidez y mantenimiento, uno de los factores variables es la rapidez de subida y descarga de información ligada directamente al ancho de banda del cliente, también es importante mencionar que no requiere de instalación por lo que puede ser ejecutado desde cualquier navegador web. A simple vista puede parecer una web estática, sin embargo, posee funcionalidades muy potentes para resolver casos particulares [9], [10].

1.2.3.3 Tipos de sistema web

- **Estático:** Tienden a ser menos susceptibles a cambios frecuentes debido a la complejidad y la dificultad asociada a su modificación. Esto se debe a que cualquier ajuste requerido implica descargar el HTML, realizar las modificaciones

necesarias y la posterior carga en el servidor. Estos sistemas son contruidos utilizando HTML y CSS.

- **Dinámico:** La complejidad de estos sistemas se incrementa en relación con las páginas estáticas. Permite realizar cambios en tiempo real a la base de datos a través de métodos HTTP, que consumen servicios REST o SOAP, alojados en un servidor web. Los lenguajes más comunes para construir este tipo de sitios son JavaScript, PHP y ASP.
- **Tienda virtual:** Diseñados para la comercialización de productos son sistemas muy complejos en los cuales existe un panel de control que permite inserción, edición y borrado de los productos de la empresa. De la misma manera brinda la manera de gestionar pagos a través de servicios como PayPal u otros métodos [10].

1.2.3.4 Arquitectura de un sistema web

La arquitectura de un sistema web, toma como referencia el esquema cliente-servidor en el cual un navegador web realiza una petición a través del protocolo HTTP o HTTPS solicitando un recurso en este caso una página web, el servidor busca el archivo y responde al usuario el recurso solicitado. En el caso de páginas dinámicas los navegadores solicitan a través del sistema cambios en la información en tiempo real mediante el uso servicios web [11], [12].

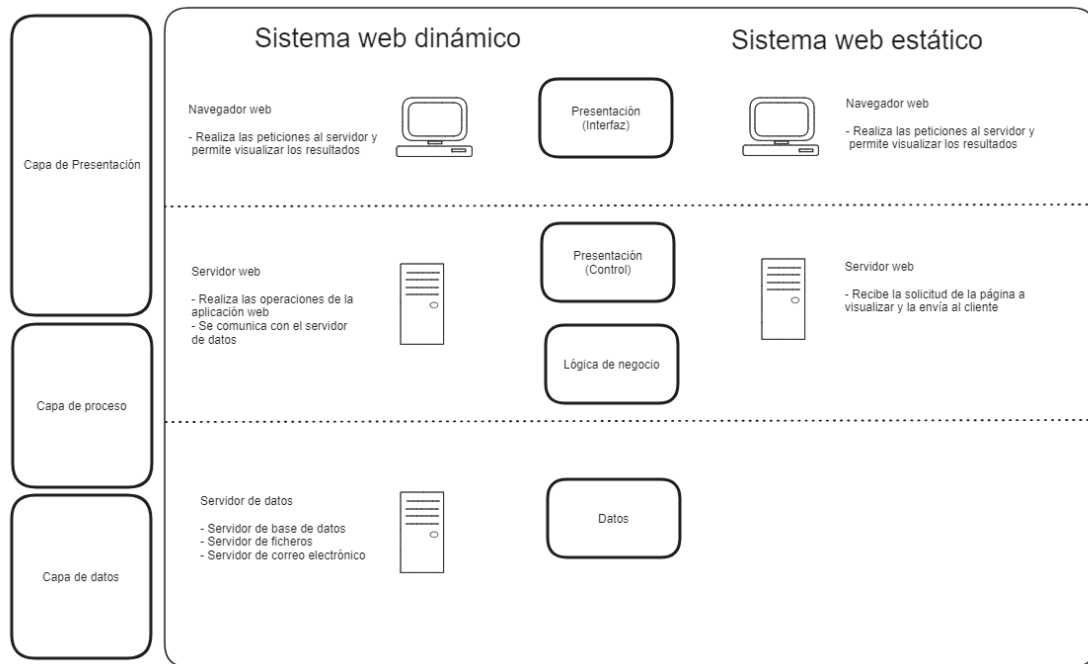


Figura 1. Arquitectura de un sistema web

Elaborado por: Investigador

1.2.3.5 Motores de Bases de datos

MongoDB

MongoDB es una tecnología de base de datos NoSQL, que tiene un enfoque totalmente distinto a SQL donde los datos son almacenados en tablas con relaciones dependientes, sin embargo, MongoDB se basa en archivos con una estructura BSON similar a JSON. Además de ser una aplicación de código abierto muy apoyada por la comunidad razón por la cual se encuentra en constante actualización. Algunas de las ventajas son:

Consultas ad hoc: Permite realizar todo tipo de consultas en la base de datos, desde consultas por rangos específicos, campos seleccionados y expresiones lambda.

Replicación: La replicación permite que los nodos secundarios funcionen como una copia del primario, logrando que en ellos solo se realice la acción de lectura. Mediante algoritmos de la base de datos la caída del nodo primario es corregida automáticamente con la asignación de un nuevo nodo primario entre los nodos secundarios.

Ejecución: A través del lenguaje de programación JavaScript permite realizar consultas directas a la base de datos[13].

Postgresql

PostgreSQL es una plataforma de gestión de bases de datos de código abierto y sin costo alguno que se distingue por su eficacia en el manejo de grandes cantidades de datos. Esta característica le brinda a Postgresql una ventaja sobre otros competidores como MySQL o SQLServer. También brinda total compatibilidad con el concepto ACID, que asegura la integridad y confiabilidad de la información guardada. PostgreSQL tiene la capacidad de realizar tanto consultas relacionales con SQL como consultas no relacionales utilizando JSON [14].

MySQL

MySQL es un gestor de base de datos con características como su facilidad de uso, fiabilidad, escalabilidad y velocidad. Originalmente fue diseñado para gestionar grandes bases de datos de manera eficiente, y ha sido utilizado en entornos exigentes durante décadas ofreciendo conectividad rápida y segura para bases de datos en línea. Sus beneficios clave incluyen una instalación sencilla, confiabilidad que está respaldada por más de 25 años de pruebas y escalabilidad para adaptarse a las demandas de aplicaciones exigentes con un alto rendimiento demostrado en múltiples benchmarks de la industria. Además, proporciona alta disponibilidad a través de tecnologías de replicación integradas,

haciéndolo adecuado para aplicaciones críticas y para cumplir con acuerdos de nivel de servicio [15].

1.2.3.6 Servicio web

Es un interfaz que permite interoperabilidad, para la transmisión de datos mediante el uso de la internet como medio de transmisión. Esta tecnología permite que distintos dispositivos accedan a ese servicio haciendo referencia a un URI (Uniform Resource Identifier) este es un código único a través del cual el cliente accede al recurso y consume procesos o solicita datos al servidor. Es importante señalar que el cliente y el servidor no están obligados a estar contruidos bajo la misma tecnología para lograr la comunicación. Las arquitecturas más utilizadas para servicios son REST que ofrece alto rendimiento con lectura masiva de datos con JSON y SOAP con un rendimiento bajo por que la comunicación se realiza con archivos XML [11], [16].

SOAP

Es una manera de compartir datos mediante el uso de la red a través de archivos XML, este se caracteriza por su modularidad y su manera para codificar los datos. Son muy estandarizados por lo cual un solo error en la programación provoca un fallo en todo el servicio. Son robustos, además al ser un medio de transferencia de datos a través de archivos XML genera gran cantidad de datos, por tal motivo consumen un gran ancho de banda [17].

REST

Es un estilo de arquitectura que se diseñó para la transferencia de información tomando como principio que encaje perfectamente en la Web, en cual el servidor recibe una llamada a través de una API RESTful, a continuación se procesa la solicitud y envía la respuesta al cliente utilizando el protocolo HTTP utilizando distintos formatos disponibles el más utilizado es JSON por la rapidez en la transmisión de datos masivos [18].

1.2.3.7 Backend

Es la parte de la programación que se realiza en el servidor se encarga de facilitar el acceso a los datos o servicios a los que no se puede acceder directamente desde el cliente. Aquí está encapsulada toda la lógica del negocio de la aplicación web, acceso a bases de datos, y servicios de correo electrónico. Aquí se debe tener presente el manejo a errores y excepciones de manera correcta para evitar caídas del sistema. Es importante tener en cuenta que las validaciones de los datos que ingresan al servidor deben ser validados antes de ejecutarse para evitar vulnerabilidades de seguridad [19].

Frameworks backend

Express

Es un framework para desarrollar servidores web escalables que tiene como características la velocidad y simplicidad, este se ejecuta sobre el entorno de nodeJS. Al estar escrito sobre JavaScript simple facilita la escritura entre servidor y cliente. Además, posee la librería npm y pnpm 2 de los mejores gestores de paquetes con la mejor resolución de dependencias que facilitan las operaciones y adición de características al servidor. La

comunidad de esta herramienta es una de las más activas y numerosas brindando la ayuda necesaria en foros o sitios de ayuda [20].

Laravel

Es el framework de backend más utilizado en la actualidad debido a que utiliza PHP como lenguaje de programación. Esta herramienta brinda una fuerte seguridad ante posibles ataques de inyección SQL o programa maligno. A su vez, utiliza el código preprogramado básico para que el usuario se pueda enfocar en la programación de aspectos específicos del software. Podría parecer que esta herramienta es una de las más difíciles, sin embargo, posee una curva de aprendizaje muy baja [21].

Django

Es una herramienta completa que brinda amplia documentación y facilidad de manejo que se encarga de lo más complicado del desarrollo web, logrando así que el usuario se enfoque en los aspectos específicos de su aplicación. Además, está basada en Python, lo cual lo convierte en un framework perfecto para aplicaciones web de todo tipo. Un aspecto importante de la herramienta es la seguridad al registrar que las contraseñas son convertidas mediante una función hash de longitud fija, de la misma manera brinda seguridad a la suplantación de cookies y la falsificación de autenticación [22], [23].

1.2.3.8 Frontend

El frontend es la parte de una aplicación web donde el cliente visualiza e interactúa con la información, por lo tanto, es esencial aplicar conocimientos de diseño de interfaces y experiencia de usuario para lograr páginas llamativas, consistentes y fáciles de usar. CSS, JavaScript y HTML son los lenguajes más utilizados, pero a través de frameworks, por mencionar algunos, se puede programar en Java, C#, TypeScript y Python [19].

Frameworks frontend

Angular

Angular es una plataforma de desarrollo basada en TypeScript que consiste en un conjunto de componentes diseñados para crear aplicaciones web escalables, adoptando una arquitectura MVC (Modelo-Vista-Controlador). Ofrece bibliotecas integradas que cubren diversas funcionalidades, como enrutamiento, gestión de formularios y comunicación cliente-servidor. Además, proporciona herramientas para facilitar tareas como creación, compilación, prueba y actualización del código fuente. Es reconocido como uno de los frameworks más completos para el desarrollo web. [24], [25].

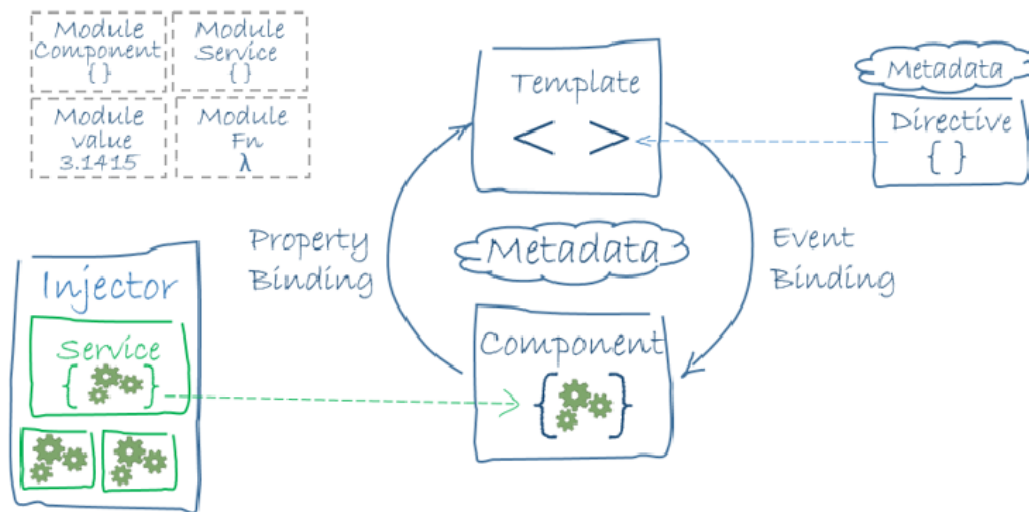


Figura 2: Arquitectura de Angular

Fuente: [19]

React

Es una librería de JavaScript de código abierto que se enfoca en la creación de sitios web que brinden más rapidez y potencia que el promedio, por tal motivo se encarga únicamente de renderizar los componentes de una aplicación web, por lo cual está enfocado en páginas web dinámicas con un alto rendimiento. Esta fue creada por Facebook y desde entonces se ha convertido en la tecnología de frontend más utilizada a nivel mundial [25], [26].

VueJS

VueJS, concebido por Evan You en 2014, surge como un framework de JavaScript destinado al diseño de interfaces de usuario y aplicaciones de una sola página. Su propósito principal radica en simplificar el proceso de desarrollo web, destacando por su reducido tamaño de apenas 18 kb, su facilidad de aprendizaje y uso, así como su óptimo rendimiento. Estas cualidades han impulsado su crecimiento y lo han posicionado como uno de los frameworks de JavaScript más demandados [27].

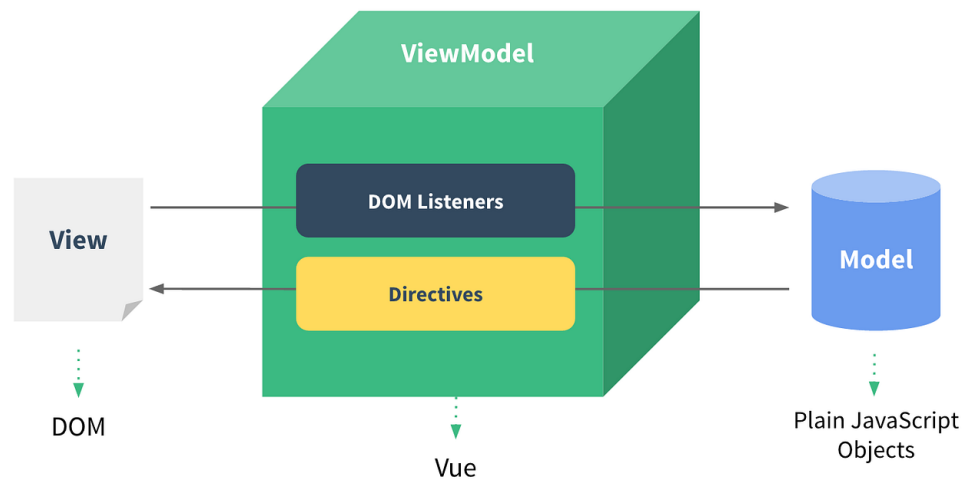


Figura 3: Arquitectura de Vue JS

Fuente: [27]

1.2.3.9 Metodologías de desarrollo

Metodología KANBAN

Se trata de una metodología predominantemente visual que facilita la organización de los procesos y las labores en curso. Los participantes tienen la capacidad de monitorear el avance de las tareas en tiempo real hasta su completa finalización. El tablero más básico consta de tres estados: "Solicitado", "En curso" y "Hecho". Uno de sus principios fundamentales reside en dirigir la gestión hacia el trabajo en sí, en lugar de centrarse en los trabajadores. Esto implica una orientación hacia la entrega oportuna al cliente y una comprensión precisa de sus necesidades, lo que a su vez se traduce en la calidad de los servicios ofrecidos [14], [28].

Metodología en Cascada

Es una metodología clásica de desarrollo que se caracteriza por su rigidez en los procesos puesto que el proceso no puede ser modificado hasta que se ha cumplido con todas las etapas del desarrollo. En casos puntuales donde se conocen a detalle los requerimientos y alcances del producto esta metodología es de gran utilidad ya que al definir claramente las necesidades del usuario los siguientes procesos son desarrollados con normalidad sin realizar cambios de ningún tipo. Para utilizar la metodología la fase análisis debe ser aplicada de manera correcta evitando dejar procesos y requerimientos ambiguos. Consta de 5 etapas muy marcadas que son las siguientes: análisis, diseño, implementación, verificación, mantenimiento [29].

Metodología RAD (Desarrollo Rápido de Aplicaciones)

Permite la entrega ágil de sistemas por lo que permite identificar y corregir errores de manera proactiva, minimizando los riesgos y los costos asociados con la integración tardía en proyectos de desarrollo de software. Esta capacidad de adaptabilidad y respuesta ágil a los cambios es una manera efectiva de desarrollo que agiliza los tiempos de entrega. Esta metodología requiere de constante comunicación entre el programador y el cliente para permitir la validación de prototipos y retroalimentación, sin embargo, el proyecto debe estar modularizado para poder desarrollarlo con RAD para permitir la integración y acoplamiento de los distintos módulos sin generar conflictos de dependencias o librerías [30].

1.2.3.10 Odontología

La odontología es una disciplina dentro de las ciencias de la salud que se dedica al diagnóstico, tratamiento y prevención de enfermedades relacionadas con el aparato estomatognático. Este sistema comprende no solo los dientes, sino también las encías, el tejido periodontal, el maxilar superior e inferior, y la articulación temporomandibular. Entre las principales afecciones que aborda la odontología se encuentran la caries dental, la maloclusión y la enfermedad periodontal [31].

1.2.3.11 Manejo de Historias Clínicas

El propósito de la historia clínica es asegurar la consolidación exhaustiva de la documentación médica de cada paciente, garantizando la máxima cohesión posible, al menos dentro del ámbito de cada institución sanitaria, y facilitando así una atención integral y eficaz [32].

1.2.3.12 Historia Clínica Única

“Es un documento médico legal que consigna la exposición detallada y ordenada de todos los datos relativos a un paciente o usuario, incluye la información del individuo y sus familiares, de los antecedentes, estado actual y evolución, además de los procedimientos y de los tratamientos recibidos” [33].

1.2.3.13 Historia Clínica activa

“Se considera a la historia clínica que tiene un periodo de vigencia hasta cinco años desde la última atención registrada” [33].

1.2.3.14 Historia Clínica inactiva

“Se considera a la historia clínica que no tiene ningún registro de atención por más de cinco años y por tanto debe ser trasladada al archivo pasivo” [32].

1.3 Objetivos

1.3.1 Objetivo general

Implementar un sistema web aplicando el framework Angular para la gestión de información y control de citas en el Consultorio Dental odontólogo Juan Carlos León.

1.3.2 Objetivos específicos

- Analizar la información y procesos que intervienen en el manejo y control de citas e historias clínicas.
- Investigar el framework Angular en el desarrollo de aplicaciones web.
- Desarrollar una aplicación web para la gestión de información y control de citas en el consultorio dental odontólogo Juan Carlos León.

CAPÍTULO II: METODOLOGÍA

2.4 Materiales

2.4.1 Institucionales

- Repositorios virtuales de la Universidad Técnica de Ambato.
- Instalaciones del Consultorio Dental Juan Carlos León.

2.4.2 Humanos

- Docente tutor de la Universidad Técnica de Ambato.
- Odontólogo Juan Carlos León.
- Investigador.

2.4.3 Económicos

Tabla 1: Recursos económicos

No	Detalle	Unidad	Cantidad	Valor Unitario	Valor Total
1	Resma de papel bond	c/u	1	5,50	5,50
2	Laptop	c/u	1	960,00	960,00
3	Pasajes de transporte	diario	80	1,20	96,00
4	Servicio de internet	mes	6	15,00	90,00

5	Impresiones	c/u	100	0,05	5,00
6	Dispositivo de almacenamiento	c/u	1	120,00	120,00
7	Suministros de oficina	c/u	1	20,00	20,00
8	Servicio de luz eléctrica	mes	6	3,00	18,00
9	Framework de desarrollo Angular	c/u	1	0,00	0,00
10	Formatos Clinica	c/u	5	0,00	0,00
				Subtotal	1314,00
				Imprevistos 10%	131,45
				Total	1445,45

Elaborado por: Investigador

En el proyecto de investigación se realizó una entrevista aplicada al personal médico del Consultorio Dental “Odontólogo Juan Carlos León”, para obtener la información sobre el proceso en agendamiento de citas y generación de Historias Clínicas, de la misma se recolectará información sobre los formatos que utiliza el Ministerio de Salud Pública del Ecuador para el campo de la odontología y como llenarlos de manera adecuada para que sean digitalizados correctamente. Además de conocer los problemas más comunes que genera el manejo de estos procesos en papel. La siguiente entrevista se divide en 2 partes una dirigida al manejo de las citas y otra dirigida al manejo de las Historias Clínicas.

Estructura de la entrevista aplicada al personal de salud del Consultorio Dental
“Odontólogo Juan Carlos León”

Tabla 2: Estructura de la encuesta realizada al personal de salud

N.º	Pregunta
Manejo de citas	
1	¿Cómo es el proceso al agendar una cita odontológica?
2	¿Qué inconvenientes son los más comunes que se presentan al agendar las citas odontológicas?
3	¿Cómo se subsanan los errores más comunes?
Manejo de Historias Clínicas	
4	El Ministerio de Salud Pública del Ecuador maneja formularios que forman la Historia Clínica, ¿Qué formularios se manejan en el consultorio dental?
5	¿El consultorio dental maneja solo odontología general o que especialidades maneja?
6	¿Cómo es el proceso al realizar una Historia Clínica?
7	¿Qué errores son los más comunes en el manejo de las Historias Clínicas?
8	¿Cuánto tiempo aproximadamente tarda en corregir el error más común que se presenta el proceso de creación de una Historia Clínica?
9	En el odontograma existen muchas simbologías, ¿Cómo son asignadas las mismas a cada una de las piezas dentales?
10	¿Qué significa el Índice CPO y CEO en Ficha Clínica Única de Odontología?
11	¿El paciente solo puede tener una Ficha Clínica Única de Odontología? O ¿A qué tiempo se genera una nueva?
12	¿Un diagnóstico puede ser presuntivo y definitivo a la vez?

13	¿En cada una de las sesiones relacionadas a la de odontología solo se realizan procedimientos de una o más especialidades a la vez?
14	Los códigos de cada pieza dental en el odontograma, ¿Cómo son asignados?
15	¿Existe una codificación para los diagnósticos?
16	¿Cómo se llena la información en el apartado de Higiene Oral Simplificada en la Ficha Clínica Única de Odontología?

Elaborado por: Investigador

Seguido a la entrevista se adjuntó el formato utilizado en Consultorio Dental para el manejo de la Ficha Clínica de Odontología.

- Formato 033 MSP (**Anexo A**)

2.5 Métodos

2.5.1 Modalidad de investigación

El presente trabajo tiene un enfoque cuantitativo-cualitativo puesto que se pretende analizar las causas y efectos que se producen con el mal manejo del agendamiento de citas y las Historias clínicas, además de hacer una comparación entre tiempos de respuesta con el sistema y sin el sistema.

Investigación bibliográfica

La investigación será bibliográfica debido a que se tomará como apoyo a la investigación el uso de libros, documentos técnicos, tesis del área informática, revistas, artículos y leyes existentes para la elaboración del marco teórico sobre los sistemas informáticos para clínicas dentales.

Investigación de campo

La investigación será también de campo debido a que se buscará obtener información con las personas encargadas del manejo y archivo de las Historias Clínicas y documentación referente a los pacientes del consultorio dental.

Investigación exploratoria

La investigación será de nivel exploratorio porque se acudirá directamente con las personas que requieren un mejor manejo de la documentación y gestión de citas.

Investigación descriptiva

La investigación será descriptiva porque se realizará un análisis para llegar a determinar la incidencia que tiene la falta de manejo y orden en las Historias Clínicas que reduce los tiempos de respuesta en centro odontológico.

Investigación aplicada

La investigación será aplicada porque tiene como objetivo resolver un determinado problema o planteamiento específico en este caso el mal manejo de la información y agendamiento de citas.

2.5.2 Población y muestra

Se presenta la población del proyecto que son todas las personas que trabajan en el Consultorio Dental “Odontólogo Juan Carlos León”.

Tabla 3: Población

Cargo	Descripción	Frecuencia	Porcentaje
Odontólogo	Persona encargada de realizar los procedimientos médicos	5	83.333%
Secretaria	Encargada del agendamiento de citas médicas e interacción con los pacientes en la sala de espera	1	16.666%
Total		6	100%

Elaborado por: Investigador

En el presente proyecto se trabajará con la población total, debido a que la misma no supera los cien elementos.

2.5.3 Recolección de la información

Para el desarrollo del proyecto de investigación se utilizará los aspectos metodológicos de las fuentes primarias, es decir, la recolección de datos e información de manera escrita u oral transmitidos por los sujetos de la investigación, para este caso se realizará una entrevista para el levantamiento respectivo de la investigación.

Tabla 4: Recolección de la información

Preguntas básicas	Explicación
¿Para qué?	Cumplir con los objetivos planteados en el proyecto de investigación.
¿De qué personas u objetos	Odontólogos y secretaria del consultorio dental.

¿Sobre qué aspectos?	Proceso en el manejo de las historias clínicas y agendamiento de citas.
¿Quién, quiénes?	Investigador: Diego Galindo.
¿Cuándo?	Período Marzo - Agosto 2024.
¿Dónde?	Consultorio Dental Odontólogo Juan Carlos León.
¿Cuántas veces?	Una vez.
¿Qué técnicas de recolección?	Entrevista.
¿Con qué?	Cuestionario.

Elaborado por: Investigador

Para la recolección de la información se realizó la entrevista al personal del Consultorio Dental “Odontólogo Juan Carlos León”, puesto que la entrevista está dividida en dos secciones el personal de salud respondió todas las preguntas, mientras que el personal administrativo solo respondió las preguntas que hacen referencia al manejo de citas odontológicas.

Al realizar la entrevista, y procesarla se obtuvieron las siguientes respuestas.

Tabla 5: Pregunta 1 – Entrevista

Pregunta N.º	1	Sección:	Manejo de Citas
¿Cómo es el proceso al agendar una cita odontológica?			
Los pacientes se pueden acercar en persona al Consultorio Dental o puede ser realizada por teléfono. Se procede a solicitar nombre, apellido, teléfono, hora, día y especialidad en caso de ser necesario. Posteriormente se verifica la disponibilidad del odontólogo en la fecha indicada en caso de ser posible se agenda la cita, caso contrario se indica los horarios disponibles para la elección del paciente.			

Elaborado por: Investigador

Tabla 6: Pregunta 2 – Entrevista

Pregunta N.º	2	Sección:	Manejo de Citas
¿Qué inconvenientes son los más comunes que se presentan al agendar las citas odontológicas?			
Los inconvenientes que más se presentan son la pérdida de la agenda donde se programan las citas y la duplicación de contactos telefónicos en el dispositivo móvil que genera información duplicada. El menos común pero no menos importante es que citas en ocasiones son agendadas a la misma hora provocando un colisión.			

Elaborado por: Investigador

Tabla 7: Pregunta 3 – Entrevista

Pregunta N.º	3	Sección:	Manejo de Citas
¿Cómo se subsanan los errores más comunes?			
Para el error de la pérdida de la agenda, se procede a buscarla dentro del consultorio y en casos extremos de no encontrarla, cada uno de los médicos asigna la cita en su calendario personal por lo que el inconveniente genera un problema al encargado del registro y en colisiones de citas médicas se procede atender a los 2 pacientes uno tras de otro, sin embargo, genera un retraso de al menos una hora en el resto de las citas.			

Elaborado por: Investigador

Tabla 8: Pregunta 4 - Entrevista

Pregunta N.º	4	Sección:	Manejo de Historias Clínicas
El Ministerio de Salud Pública del Ecuador maneja formularios que forman la Historia Clínica, ¿Qué formularios se manejan en el consultorio dental?			
Dentro del consultorio dental se manejan dos formularios que son la Ficha Clínica Única de Odontología (Formulario 033) y el Consentimiento Informado (Formulario 024).			

Elaborado por: Investigador

Tabla 9: Pregunta 5 - Entrevista

Pregunta N.º	5	Sección:	Manejo de Historias Clínicas
¿El consultorio dental maneja solo odontología general o que especialidades maneja?			
Si se manejan especialidades en el consultorio y son las siguientes: - Endodoncia - Ortodoncia - Rehabilitación Oral - Periodoncia - Cirugía maxilofacial - implantología - Odontopediatría			

Elaborado por: Investigador

Tabla 10: Pregunta 6 - Entrevista

Pregunta N.º	6	Sección:	Manejo de Historias Clínicas
¿Cómo es el proceso al realizar una Historia Clínica?			
El proceso comienza solicitando los datos generales del paciente, motivo de la consulta, enfermedades o antecedentes familiares. Se realiza un examen del sistema estomatognático para posteriormente realizar el odontograma el cual se indica los procedimientos realizados y por realizar en cada pieza dental. Luego se completa el índice CEO-CPO, diagnósticos, examen de higiene oral. Finalmente se coloca los diagnósticos de la Ficha Clínica Única de Odontología y se abre las sesiones relacionadas al paciente. En ese momento se procede con el tratamiento de los diagnósticos. En el caso de una cita de un paciente que cuenta con Ficha Clínica Única de Odontología, se genera una nueva sesión y se procede a realizar el procedimiento.			

Elaborado por: Investigador

Tabla 11: Pregunta 7 - Entrevista

Pregunta N.º	7	Sección:	Manejo de Historias Clínicas
¿Qué errores son los más comunes en el manejo de las Historias Clínicas?			
Uno de los errores más comunes es la equivocación al momento de llenar los campos de un apartado que genera la repetición del documento o uso de correctores líquidos. Además, en el odontograma se generan dibujos, uno sobre otro perdiendo casi totalmente la visibilidad de todos los procedimientos que tiene la pieza dental generando confusión. En la búsqueda de estos archivos se producen problemas por la organización de los papeles.			

Elaborado por: Investigador

Tabla 12: Pregunta 8 - Entrevista

Pregunta N.º	8	Sección:	Manejo de Historias Clínicas
¿Cuánto tiempo aproximadamente tarda en corregir el error más común que se presenta el proceso de creación de una Historia Clínica?			
El error más común en el peor de los casos se soluciona con la repetición del documento que suele tardar alrededor de 8 a 10 minutos. Varía según los campos que se han sido llenados hasta que se generó el error.			

Elaborado por: Investigador

Tabla 13: Pregunta 9 - Entrevista

Pregunta N.º	9	Sección:	Manejo de Historias Clínicas
En el odontograma existen muchas simbologías, ¿Cómo son asignadas las mismas a cada una de las piezas dentales?			
En el odontograma existen figuras que representan el procedimiento. Si la figura es azul significa que el procedimiento ha sido realizado, si la figura es de color rojo el procedimiento esta por realizarse. En el caso de las caries es el único procedimiento que se puede asignar a cada una de las caras de la pieza dental, el resto de los procedimientos se asignan a toda la pieza dental.			

Elaborado por: Investigador

Tabla 14: Pregunta 10 - Entrevista

Pregunta N.º	10	Sección:	Manejo de Historias Clínicas
¿Qué significa el Índice CPO y CEO en Ficha Clínica Única de Odontología?			
El índice CEO es para niños su significado es:			
- C: Carie - E: Extraído			

- O: Obturado

El índice CPO es para adultos su significado es:

- C: Carie
- P: Perdido
- O: Obturado

En estos recuadros se coloca el total de piezas dentales que cumplan con las opciones.

Elaborado por: Investigador

Tabla 15: Pregunta 11 - Entrevista

Pregunta N.º	11	Sección:	Manejo de Historias Clínicas
¿El paciente solo puede tener una Ficha Clínica Única de Odontología? O ¿Bajo qué circunstancia se genera una nueva?			
No, al paciente se le genera una nueva Ficha Clínica Única de Odontología con el mismo número de Historia Clínica cuando ha pasado un año desde su última sesión.			

Elaborado por: Investigador

Tabla 16: Pregunta 12 - Entrevista

Pregunta N.º	12	Sección:	Manejo de Historias Clínicas
¿Un diagnóstico puede ser presuntivo y definitivo a la vez?			
No, solo puede seleccionarse uno de los dos.			

Elaborado por: Investigador

Tabla 17: Pregunta 13 - Entrevista

Pregunta N.º	13	Sección:	Manejo de Historias Clínicas
¿En cada una de las sesiones relacionadas a la de odontología solo se realizan procedimientos de una o más especialidades a la vez?			
En el caso de requerirse 2 o más procedimientos, se abre una nueva sesión con la misma fecha asignada a cada especialidad.			

Elaborado por: Investigador

Tabla 18: Pregunta 14 - Entrevista

Pregunta N.º	14	Sección:	Manejo de Historias Clínicas
Los códigos de cada pieza dental en el odontograma, ¿Cómo son asignados?			
Las piezas dentales que comienzan con 5, 6, 7 y 8 son utilizadas para piezas dentales de niños, mientras el resto son para el paciente adulto.			

Elaborado por: Investigador

Tabla 19: Pregunta 15 - Entrevista

Pregunta N.º	15	Sección:	Manejo de Historias Clínicas
¿Existe una codificación para los diagnósticos?			
Los códigos no son colocados arbitrariamente, está regido por el ministerio de salud pública a través de un documento publicado en 2014 [34].			

Elaborado por: Investigador

Tabla 20: Pregunta 16 - Entrevista

Pregunta N.º	16	Sección:	Manejo de Historias Clínicas
¿Cómo se llena la información en el apartado de Higiene Oral Simplificada en la Ficha Clínica Única de Odontología?			
En este apartado debe seleccionarse una pieza dental por cada fila, luego se debe colocar la gravedad de cada diagnostico números del 0 al 3 según la gravedad. En la totalidad se representa el promedio en decimales de cada columna de valores.			

Elaborado por: Investigador

2.5.4 Procesamiento y análisis de datos

En base a la encuesta realizada en el Consultorio Dental “Odontólogo Juan Carlos León” se puede determinar que el proceso de agendamiento de citas requiere mejoras, puesto que existen carencias en la manera que se almacena la información, lo cual provoca retraso en las atenciones médicas y problemas al buscar la información. Es por eso por lo que se requiere de un sistema que centralice los datos de pacientes y las citas asignadas. Además de facilitar la búsqueda de estos.

En general en el proceso de manejo de Historias Clínicas, presenta problemas cuando se comenten errores que comprometen la integridad física del documento, porque un eventual error requiere que todo el proceso se repita desde el inicio lo cual provoca que el tiempo de atención aumente generando incomodidad en los pacientes y molestia en el odontólogo. De la misma manera el odontograma no se puede visualizar correctamente cuando son varios los procedimientos aplicados a una pieza dental, generando confusión en la lectura de este.

Por último, se puede llegar a la conclusión de que el sistema a realizar necesita implementar una gestión de roles porque los procesos referentes a la historia Clínica no pueden ser realizados por todos los empleados del establecimiento. Debido a la sensible información que manejan estos archivos. Se puede brindar la opción de visualizar la información al personal administrativo, sin embargo, no podrán editar, eliminar o crear ninguna Historia Clínica.

CAPITULO III

3.1 Desarrollo de la propuesta

3.1.1 Metodología de desarrollo de la tesis

3.1.1.1 Levantamiento de requerimientos con la IEEE 830

Introducción

El propósito de este documento es documentar los procesos y flujo de la información. Además de los requisitos funcionales y no funcionales que debe cumplir el sistema web para que los requisitos sean verificados y validados objetivamente por parte del analista y del usuario final o cliente.

Propósito

Este documento es la especificación de requisitos para el grupo de investigación del Consultorio Dental “Odontólogo Juan Carlos León”. Se basa en la estructura y contenidos registrados en la Norma IEEE 830

Ámbito del sistema

La aplicación web se implementará en el Consultorio Dental “Odontólogo Juan Carlos León” en la ciudad de Ambato, para mejorar tanto el manejo como la generación de citas y las historias clínicas de los pacientes.

El objetivo principal es gestionar el agendamiento de las citas médicas asignadas a distintos especialistas del consultorio odontológico y el manejo de historias clínicas que involucra la creación, edición y acceso a las mismas de manera ordenada y controlada. Algunas de las funciones que el sistema debe realizar son las siguientes:

- Gestionar los usuarios según los roles que se establecidos en el sistema aplicando medidas de seguridad y protección.
- Administrar las historias clínicas y facilitar la búsqueda de información.
- Administrar el agendamiento de citas de manera rápida y ordenada.
- Administrar pacientes.
- Administrar especialistas.
- Asignar especialistas a cada sesión de un tratamiento.

Definiciones, Acrónimos y Abreviaturas

Administrador: Persona encargada de gestionar los cambios relevantes del sistema, que requieren conocimiento en sistemas.

Angular: Framework de desarrollo frontend desarrollado por angular para creación de clientes web.

ERS: Documento de Especificación de Requisitos del Software.

Ficha Clínica Única de Odontología: Documento utilizado para registro de los procesos médicos en el área de Odontología.

FCUO: Ficha Clínica Única de Odontología.

Historia Clínica: Documento que almacena tratamientos médicos, exámenes, signos vitales de un paciente a través del tiempo.

MYSQL: Es una herramienta de base de datos relacional propiedad de Oracle.

Odontograma: Es un esquema gráfico en el que el odontólogo puede anotar los diagnósticos de cada pieza dental a través de símbolos.

NodeJS: Es una tecnología web para desarrollo de aplicaciones web con arquitectura basada en eventos que utiliza el lenguaje JavaScript.

Permiso: Proceso que determina si un usuario tiene acción sobre el recurso mencionado.

Rol: Identifica al usuario dentro de una aplicación.

Visión General del Documento

El siguiente documento muestra la información de los requisitos de la aplicación web, de una manera general. Los requerimientos específicos del sistema se detallarán de una manera más profunda, que permitirá el diseño del sistema que cumpla con las necesidades planteadas por el usuario y luego realizar las pruebas oportunas para corroborar que el sistema cumpla con los objetivos establecidos por el usuario.

Descripción General

Este software tiene como fin agilizar y simplificar el trabajo que se realiza en el consultorio odontológico Juan Carlos León, además de lograr un nivel alto de seguridad

y orden de las historias clínicas logrando así dar fin al uso excesivo de papel, cambiándolo por información digital que a su vez no se deteriore en el tiempo logrando a su vez ahorrar espacio en el consultorio.

Este producto tiene como funcionalidad llevar un control adecuado de todos los procesos concernientes a la administración online de citas médicas e historias clínicas. Los usuarios del sistema constaran con los siguientes roles:

- a) Usuario administrador
- b) Usuario secretaria
- c) Usuario odontólogo

En la sección 2.3 se profundizará las características de cada usuario.

En relación con la seguridad se aplicará un sistema para inicio de sesión con roles, así se gestionará el acceso y manipulación de a los módulos de la aplicación a cada usuario.

Perspectiva del producto

Es una aplicación web diseñada para el consultorio odontológico Juan Carlos León, que se desarrollará como requerimiento de parte del gerente propietario. Está enfocado en la mejora del control y gestión de citas médicas e historias clínicas.

Funciones del producto

El sistema deberá tener los procesos necesarios para la gestión de los servicios odontológicos que brinda el consultorio, considerando el correcto flujo de la información

entre las distintas entidades. A través del acuerdo firmado con el Gerente Propietario. El sistema contará con las siguientes funciones.

- Gestión de Historias Clínicas.
- Gestión de pacientes.
- Gestión de usuarios.
- Gestión de citas.
- Gestión de especialidades
- Gestión de roles

Características de los usuarios

Tabla 21: Características usuario administrador

Administrador	
Formación	Universitaria
Habilidades	Alto conocimiento en las ciencias informáticas.
Actividades	Administrar el sistema manejando las configuraciones y aplicando mantenimientos para el correcto funcionamiento del sistema.

Elaborado por: Investigador

Tabla 22: Características usuario secretaria

Secretaria	
Formación	Profesional en Contabilidad y Auditoría, Bachiller en Contabilidad
Habilidades	Atención al cliente y un nivel básico en informática.
Actividades	Agendar las cita y, registrar nuevos pacientes.

Elaborado por: Investigador

Tabla 23: Características usuario administrador

Odontólogo	
Formación	Odontólogo
Habilidades	Nivel alto en la rama de las ciencias médicas referente a odontología y sus diferentes especialidades. Nivel básico en informática.
Actividades	Llenará las consultas realizadas, con su diagnóstico, enfermedades, receta médica, prescripción y certificados.

Elaborador por: Investigador

Restricciones

- La interfaz debe ser amigable y que permita navegar rápidamente por la aplicación al usuario
- El sistema debe adaptarse al mayor porcentaje de navegadores web.

- El sistema debe ser responsive.
- El sistema debe ser accesible en todo momento.
- El sistema autenticara cada petición realizada hacia el servidor web.

Los usuarios poseen restricciones para acceder a módulos del sistema o procesos que realiza el sistema:

- Para el usuario administrador, que se encarga del mantenimiento y actualizaciones del sistema no existirá ningún tipo de restricción, ellos poseerán los privilegios totales en la aplicación y tendrán al código fuente del sistema.
- El usuario secretaria, en este caso solo podrá acceder a la creación de pacientes, generación de citas médicas, y visualización de las historias clínicas como documentos relacionados a la misma, por último.
- El usuario odontólogo, tendrá accesos a toda la aplicación donde se realizan los procesos y las gestiones relacionadas a historias clínicas y agendamiento de citas. No podrán acceder al código fuente de la aplicación y a los parámetros de configuración de diagnósticos, enfermedades o especialidades.

Suposiciones y dependencias

La aplicación web será ejecutará sobre navegadores web de terceros, por lo que se tomará en cuenta que los estilos aplicados a la interfaz del sistema sean compatibles con gran parte de estos. También es importante tener presente que un cambio en los navegadores web influye en la ejecución del sistema.

Requisitos Futuros

Los siguientes requisitos pueden ser posibles mejoras, que luego podrán ser implementadas en el sistema para aumentar la automatización y digitalización de ciertos procesos en el consultorio:

- Módulo de edición de rayos X.
- Aplicación móvil para odontólogos.
- Horario de médicos en tiempo real.

Requisitos específicos

Tabla 24: Requisito gestionar Historias Clínicas

ID Requisito	RF1
Nombre de requisito	Gestionar Historias Clínicas
Descripción	El sistema presentará un formulario para generar el ingreso de datos y edición del odontograma de cada paciente. Sin permitir la edición de este pasado el día en que fue abierto para que tenga una validez legal como menciona el MSP.
Prioridad del requisito	Alta

Elaborado por: Investigador

Tabla 25: Requisito gestionar pacientes

ID Requisito	RF2
Nombre de requisito	Gestionar paciente

Descripción	El sistema permitirá visualizar y gestionar todos los datos de los pacientes que han sido atendidos en el establecimiento médico.
Prioridad del requisito	Alta

Elaborado por: Investigador

Tabla 26: Requisito gestionar citas

ID Requisito	RF3
Nombre de requisito	Gestionar cita odontológica
Descripción	El sistema permitirá la gestión de toda la información referente a citas odontológicas, asignando el paciente a un especialista en una fecha y hora específica.
Prioridad del requisito	Alta

Elaborado por: Investigador

Tabla 27: Requisito Iniciar sesión

ID Requisito	RF4
Nombre de requisito	Iniciar sesión
Descripción	El sistema deberá contar con un módulo que permita iniciar sesión a los miembros del Consultorio Dental “Odontólogo Juan Carlos León”. Obligatorio para visualizar el contenido.
Prioridad del requisito	Alta

Elaborador por: Investigador

Tabla 28: Requisito gestionar usuarios

ID Requisito	RF5
Nombre de requisito	Gestionar usuarios
Descripción	El sistema deberá contar con un módulo que permita la gestión y visualización de los datos de usuarios, de la misma manera permitirá la asignación de privilegios a cada uno de ellos. Existen 3 roles administrador, secretario, odontólogo.
Prioridad del requisito	Alta

Elaborado por: Investigador

Tabla 29: Requisito gestionar roles

ID Requisito	RF6
Nombre de requisito	Gestionar roles
Descripción	El sistema permitirá gestionar los privilegios que existen en el sistema asignando módulos a cada rol.
Prioridad del requisito	Alta

Elaborado por: Investigador

Tabla 30: Requisito gestionar especialidades

ID Requisito	RF7
Nombre de requisito	Gestionar especialidades
Descripción	El sistema permitirá gestionar las especialidades de cada médico. Existirán las siguientes especialidades:

	• Endodoncia • Ortodoncia • Rehabilitación Oral • Periodoncia • Cirugía Maxilofacial • Implantología • Odontopediatría
Prioridad del requisito	Alta

Elaborado por: Investigador

Interfaces externas

El usuario será capaz de utilizar la aplicación en Windows, Linux y OSX. Puesto que corre en navegadores web. No necesita de instalación de ningún aplicativo adicional.

Interfaces de comunicación

El sistema establecerá comunicación con el usuario mediante internet, a través de peticiones HTTP por lo que la comunicación estará regida por el protocolo TCP/IP.

Requisitos NO Funcionales

Rendimiento

Permitirá la conexión de usuarios de manera de simultanea permitiendo así un manejo eficiente de la información y la concurrencia.

Seguridad

El sistema proporcionara seguridad a la información en un hosting que permita evitar los ataques más comunes para los sistemas en línea, protección de puertos y esconder la IP.

Fiabilidad

El sistema deberá validar la información de manera que evite información errónea o no autorizada para garantizar la integridad de los datos.

Mantenibilidad

El sistema será desarrollado de manera modular para que posteriormente facilite el mantenimiento del sistema y el acoplamiento de módulos adicionales.

Portabilidad

En la portabilidad el sistema será desarrollado como aplicación web por lo tanto podrá ser ejecutado desde cualquier dispositivo con conexión a internet y que cuente con un navegador web.

Requisitos de Rendimiento

No se han determinado Requisitos de eficiencia, aunque es recomendable que se intente optimizar todo lo posible el servidor web que será configurado en Node JS con el framework Express que nos brinda mayor velocidad por ser orientado a eventos y la base de datos correrá sobre MySQL para rendimiento y escalabilidad, puesto que es una aplicación web donde tendrán acceso muchos usuarios y se llevaran a cabo diferentes procesos con mucha información.

Restricciones de Diseño

El desarrollo de la aplicación tiene ciertas restricciones bajo las cuales se debe llevar a cabo el proceso de diseño. A continuación, se enlistan las restricciones relacionadas con el diseño:

- El análisis y diseño de la aplicación se hace bajo los principios de la ingeniería de software.
- Programación MVC (modelo, vista, controlador) en el lado del servidor.
- El lenguaje de programación es TypeScript, con el framework angular para generar una SPA(Single Page Application).

Atributos del sistema

El sistema será fiable al momento de manejar toda clase de información necesaria para el agendamiento de citas y manejo de historias clínicas además de la seguridad que se emplea para cada uno de los roles que maneja. El mantenimiento se realizará siempre y cuando existan actualizaciones que den nuevas funcionalidades al software o han generado bugs. En cuanto escalabilidad el programa al ser modular puede crecer de manera exponencial con la adhesión de más módulos o submódulos.

3.1.1.2 Definición de arquitectura

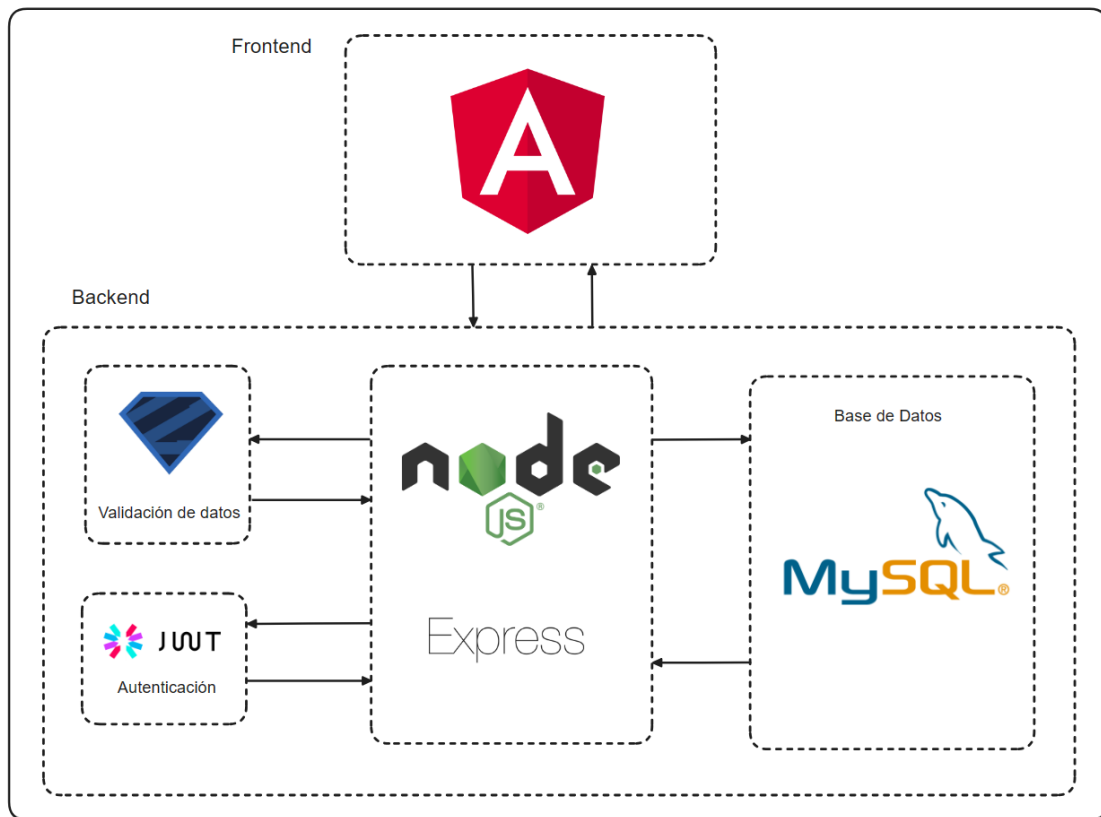
Arquitectura del servidor Web

Se escogió la arquitectura MVC (Modelo-Vista-Controlador) para la codificación del servidor que manejará un servicio de API REST, ahí se dividirá la aplicación en tres capas bien diferenciadas.

- Modelo: Se encarga de establecer la conexión hacia la base de datos del sistema, para brindar el acceso a la información que sea requerida al hacer una petición a un recurso del servidor controlando los errores y excepciones que se presenten.
- Vista: Sirve la información solicitada una vez se han aplicado las reglas de negocio en el controlador del servidor y se han aplicado los filtros correspondientes.
- Controlador: Permite la comunicación entre el Modelo y la Vista, aplicando las reglas de negocio y gestiona todas las excepciones del proceso.

Para la validación de datos una parte importante en la seguridad de la aplicación se decidió utilizar la herramienta ZOD que permite gestionar los atributos de un objeto. Facilitando el control de la extensión, tipo de dato, valores nulos permitidos y muchas características más que permite la validación parcial o total del objeto. Mejora el control de errores para validar datos porque se autogeneran con una estructura determinada, sin embargo, permite personalizar los mensajes de error específicos para cada atributo y el mensaje de error general.

Figura 4: Arquitectura de la aplicación web



Elaborado por: Investigador

3.1.1.3 Metodología Desarrollo

Metodología de desarrollo

Después de analizar las metodologías de desarrollo clásicas y ágiles se eligieron 3 metodologías para ser comparadas y determinar la metodología que mejor se adapta al proyecto. Las metodologías seleccionadas tomando como punto de referencia el tamaño de los equipos de trabajo son: cascada, RAD y Kanban. A continuación, la tabla de comparación.

Tabla 31: Comparativa metodologías de desarrollo

	Cascada	RAD	Kanban
Tamaño del equipo	Grande o pequeño	Pequeño	Necesita ser pequeño
Tiempo	Sigue un cronograma específico. Si se realizan cambios relevantes empieza nuevamente.	El proyecto concluye cuando es aprobado por el cliente.	El proyecto concluye al cuando no existen historias de usuario pendientes
Proyecto ideal	Aplicaciones de escritorio	Aplicaciones web	Todo tipo de aplicación
Cliente ideal	Sabe lo que desea del software desde el comienzo	Comunicador y dispuesto a participar	Comunicador y dispuesto a participar
Prototipos	Entregados al final del cronograma	Entregados frecuentemente para validación	Entregados frecuentemente para validación

Elaborado por: Investigador

En la ingeniería de software es fundamental escoger la metodología de desarrollo correcta está nos brinda un conjunto de pautas y estrategias para lograr culminar el software deseado de manera correcta controlando la estructura, planificación y seguimiento sobre los procesos de este. Siguiendo las fases y técnicas que engloba la metodología, sin embargo, se debe tomar en cuenta un aspecto muy importante y es que el proyecto no debe adaptarse a la metodología de desarrollo, sino en su lugar la metodología de desarrollo debe adaptarse al proyecto. Después de un análisis extenso de las metodologías clásicas y ágiles se ha tomado la decisión de tomar como base del proyecto la metodología ágil RAD añadiendo la técnica del flujo de trabajo de la metodología Kanban.

Tabla 32: Características metodologías seleccionadas

Técnicas	RAD	Kanban
1.	Entregas paulatinas de prototipos	Flujo de trabajo por medio de tableros para visualización del progreso del desarrollo
2.	Validación de prototipos por parte del cliente	
3.	Tiempos de entrega cortos	
4.	Interacción constante con el cliente	

Elaborado por: Investigador

Según las fases de la metodología seleccionada en este caso RAD se debe cumplir con las siguientes fases: planificación de requisitos, diseño con el cliente, construcción y transición. En cada una de estas fases se debe cumplir con algunos requisitos para que el proceso se ejecute de manera correcta.

Figura 5: Diagrama Metodología RAD



Elaborado por: Investigador

El Consultorio Dental “Odontólogo Juan Carlos León” brindo todas las facilidades para cumplir con la metodología seleccionada, interactuando de manera constante en el diseño del producto para que cumpla con las necesidades del usuario final

3.1.1.4 Herramientas

Selección de la tecnología de frontend

Se han investigado gran cantidad de tecnologías para decidir en cual se desarrollará el cliente. Se ha descartado una de las tecnologías emergentes más populares Astro ya que está enfocado en la creación de sitios web estáticos y esta aplicación será dinámica, es así que se selecciona los 3 frameworks con mejor oferta laboral y más populares los cuales son: Angular, React y VueJS.

Tabla 33: Comparativa tecnología frontend

	Angular	React	VueJS
Creador	Google	Facebook	Evan You (exemplado de Google)
Empresas que lo utilizan	Nike, HBO, American Express	Netflix, Airbnb, Instagram	Grammarly, Adobe
Curva de aprendizaje	Alta	Baja	Media
Arquitectura	Basada en componentes	Basada en componentes	Basada en componentes
Alcance	Es un framework completo, permite	Es una librería para creación de interfaces enfocada en el	Es un framework de nivel medio que

	generar aplicaciones a gran escala	rendimiento en la capa de aplicación	genera aplicaciones de mediana escala
Migración de versiones	CLI brinda la facilidad al usuario para las migraciones a través de un par de comandos	Se debe verificar compatibilidad entre versiones en muchas librerías una por una.	Migración sencilla a través de la actualización de dependencias
Flujo de datos	Two-way Data binding	One-Way Data binding	One-Way Data binding
Lenguaje	TypeScript	JavaScript	JavaScript
Ciclo de vida del componente	Manejado con métodos	Manejado con Hooks	Manejado con Hooks
Integración	Compleja	Fácil	Medio
Tipo de DOM	DOM real	DOM virtual	DOM virtual
Rendimiento	Medio	Bueno	Bueno
Taza de mercado	Baja	Muy alta	Baja
Código	Abierto	Abierto	Abierto

Elaborado por investigador

Una vez realizada la comparativa entre las tecnologías y evaluando sus características se decidió seleccionar la herramienta Angular, que será destinada el desarrollo de la aplicación web, el software puede presentar un menor rendimiento en comparación a sus adversarios por el manejo de DOM real. Sin embargo, para el presente proyecto se requiere de una aplicación que sea escalable, segura, completa y estructurada. Además, uno de los aspectos más importantes para ser seleccionado es que el proyecto secreto llamado WIZ que utiliza Google para programar todas sus web apps se integrará con

Angular en un solo framework. A continuación, se resaltan algunas de las características de las características que brinda el framework:

- Se maneja el árbol de estados a través de Signals, evitando así recorrer todo el árbol de componentes y actualizarlo completamente.
- Los Resolvers permiten la precarga de datos en un componente antes de que esté se dibuje en el DOM.
- Las peticiones HTTP cada vez que se ejecutan invocan a la clase Interceptor que permite manejar los errores de usuario de manera uniforme, o agregar encabezados de autorización.
- Guards ayuda a la protección de rutas a través de directrices establecidas dentro de su programación.
- Lazy Loading brinda la posibilidad de no cargar las rutas de la aplicación de manera abrupta, sino que carga las rutas según la demanda.
- Directivas que forman parte del núcleo de Angular existen tres tipos: directiva estructural, directiva de componente, directiva de atributos.

Selección de la tecnología de backend

Se han investigado gran cantidad de tecnologías para decidir en cual se desarrollará el servidor de la aplicación. Se ha descartado muchas tecnologías es así como se selecciona los 3 frameworks con mejor oferta laboral y más populares los cuales son: NodeJS, Django y Laravel.

Tabla 34: Comparativa tecnología backend

	NodeJs	Django	Laravel
Lenguaje	JavaScript	Python	PHP

Curva de aprendizaje	Baja	Baja	Media
Rendimiento	Muy Alto	Alto	Medio
Seguridad	Alta	Muy alta	Media
Código	Abierto	Abierto	Abierto
Comunidad y Documentación	Alta	Media	Muy Alta
Arquitectura	Orientada eventos	MVC	MVC

Elaborado por: Investigador

Tras la comparativa entre todas las tecnologías de backend se decidió utilizar como tecnología de backend NodeJS, esta tecnología se basa en el motor V8 utilizado por Chrome por lo que brinda un tratamiento de flujo de datos rápido y una alta velocidad. Además, permite conexión de gran cantidad de usuarios de manera simultánea y seguridad para el servidor a través de librerías disponibles con npm. La herramienta se utilizará junto a Express. Express es una librería de NodeJS que nos brinda las siguientes características:

- Actúa como un middleware en las solicitudes HTTP, así podemos verificar un token de autorización dentro de una solicitud antes de que sea ejecutada.
- Brinda una estructura para el desarrollo de código logrando que este sea más limpio.
- Definir las rutas del servidor de manera ordenada y modular, logrando que cada ruta responda con una función para los diferentes métodos existentes.

Motor de Base de Datos

Una de la parte fundamental del sistema radica en donde se almacenará la información es así como se han tomado como referencia 3 de las bases de datos más utilizadas y conocidas para seleccionar la más adecuada al proyecto.

Tabla 35: Comparativa tecnología Base de Datos

	MySQL	PostgreSQL	Mongo DB
Utiliza	Netflix, Uber	Start ups, GitLab	LinkedIn, Cisco
Seguridad	Control de acceso y autenticación	Control de acceso y autenticación	Control de acceso y autenticación
Código	Abierto en su versión Estándar	Abierto	Abierto
Concurrencia	Buen manejo de concurrencia, escritura lenta	Maneja la concurrencia con su lectura en paralelo	Muy alta al ser datos no estructurados su escritura es más veloz
Escalabilidad	Horizontal y vertical	Horizontal y vertical	Horizontal y vertical
Base de datos	Relacional	Relacional	No relacional
Lenguaje de consulta	SQL	SQL	MQL
Curva de aprendizaje	Media	Alta	Media
Rendimiento	Bueno	Muy bueno	Muy bueno
Cumplimiento ACID	Alta	Alta	Media

Elaborado por: Investigador

Una vez realizada la comparación entre las tecnologías de Base de datos se tomó la decisión de utilizar MySQL para gestionar la información del proyecto. Debido a la escalabilidad horizontal y vertical que brinda, además, la concurrencia que maneja de manera simultánea es muy buena y en caso de necesitar más características posee una versión de pago que brinda soluciones para grandes empresas. Tiene un manejo sólido de la filosofía ACID que da seguridad y confiabilidad en el manejo de datos, así como la transaccionalidad múltiple que permite enlazar muchas transacciones. Por la cantidad de información sensible que se manejará se escogió este software que brinda seguridad en el acceso y tratamiento de datos.

3.1.1.5 Ejecución

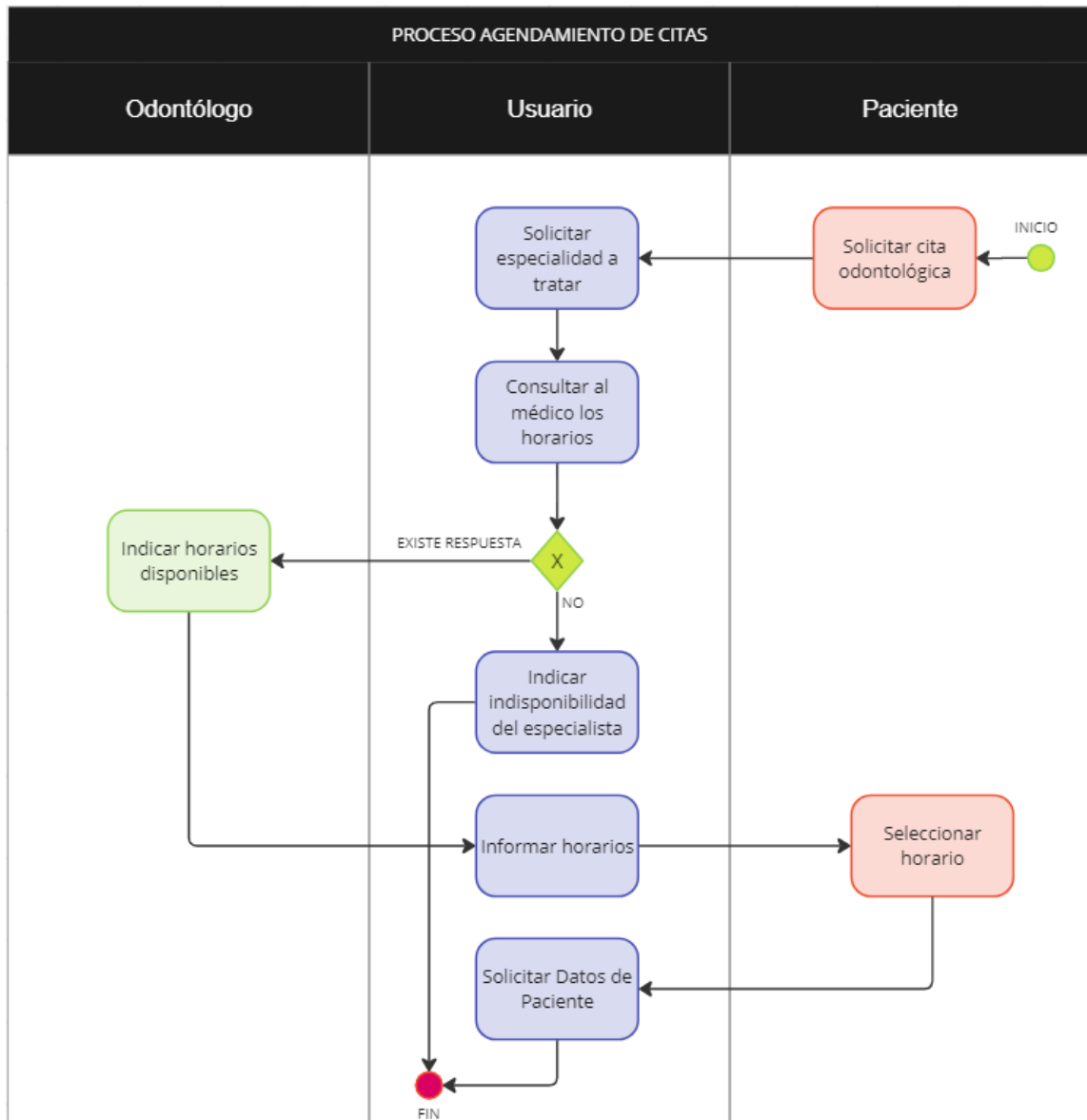
Planificación de Requisitos

Modelado de procesos

Tras el análisis de los procesos con los que cuenta el Consultorio Dental “Odontólogo Juan Carlos León” se modelan los procesos de agendamiento de citas y gestión de Historias Clínicas donde se especifica los distintos usuarios que participan dentro de este. Teniendo como resultado los siguientes gráficos que explican el manejo manual de los procesos que se llevan a cabo dentro del establecimiento.

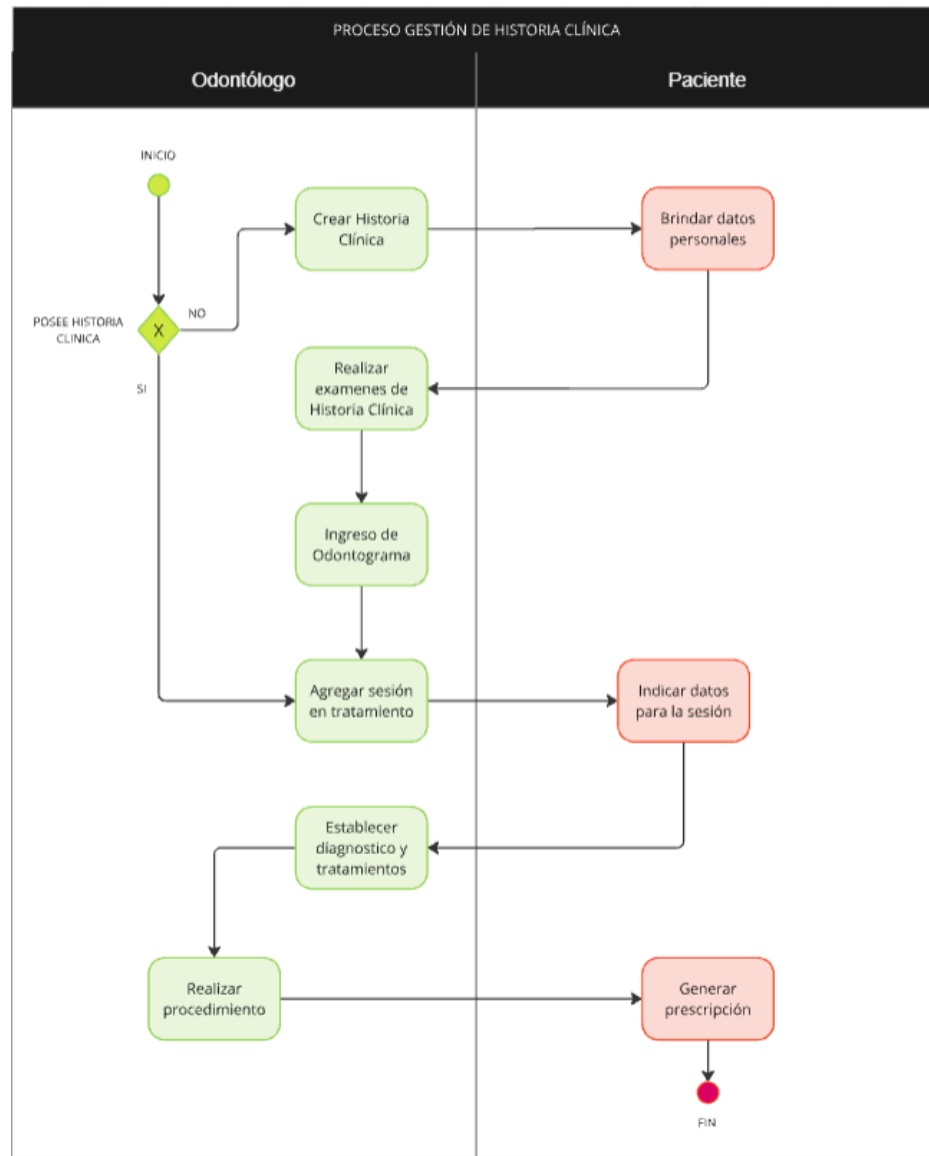
Proceso agendamiento de citas

En el proceso de agendamiento de citas se define claramente la intervención de 3 actores en este caso: paciente, usuario y odontólogo cada uno de ellos cumple tareas específicas.



Proceso gestión de la Historia Clínica

En el proceso de agendamiento de citas se define claramente la intervención de 2 actores en este caso: paciente y odontólogo cada uno de ellos cumple tareas específicas. La mayor parte del proceso corre por cuenta del odontólogo donde el usuario se presta a brindar la información necesaria.



Historias de Usuario

Las historias contienen la información que se necesita para cumplir con los requisitos del usuario o cliente. Estas contienen información general con breves rasgos sobre lo que se desea de usuarios son explicaciones cortas con rasgos breves como la prioridad, usuario, descripción, etc. Para este caso se utilizó el siguiente esquema de historia de usuario.

Tabla 36: Plantilla utilizada para historia de usuario

Historia de usuario			
ID Requisito			
ID		Usuario	
Nombre de historia			
Prioridad		Tiempo estimado	
Responsable			
Descripción			
Observación			

Elaborado por: Investigador

Donde:

- **ID:** Identificador único para la historia de usuario.
- **Usuario:** El usuario que ejerce de actor en la historia de usuario.
- **ID Requisito:** Requisito al que hace referencia la historia de usuario (Si es el caso)
- **Nombre:** Título de la Historia.
- **Prioridad:** Escala de importancia comprendida entre alta, media, baja para cualificar la prioridad de la historia.
- **Tiempo estimado:** Tiempo en días que se estima para finalizar la historia de usuario.

- **Responsable:** Personal asignado a la historia de usuario.
- **Descripción:** Explica que se desea conseguir con la historia de usuario.
- **Observación:** Detalles a tomar en cuenta.

Tabla 37: Historia de usuario 1

Historia de usuario			
ID Requisito			
ID	HU001	Usuario	Desarrollador
Nombre de historia	Análisis y diseño de la base de datos		
Prioridad	Alta	Tiempo estimado	6
Responsable	Diego Galindo		
Descripción	Construir el esquema físico y lógico para la base de datos creando de manera oportuna las relaciones con las tablas.		
Observación			

Elaborado por: Investigador

Tabla 38: Historia de usuario 2

Historia de usuario			
ID Requisito			
ID	HU002	Usuario	Desarrollador
Nombre de historia	Estructura de proyectos servidor y cliente		
Prioridad	Alta	Tiempo estimado	1
Responsable	Diego Galindo		

Descripción	Construir el esquema físico y lógico para la base de datos creando de manera oportuna las relaciones con las tablas.
Observación	Ninguna

Elaborador por: Investigador

Tabla 39: Historia de usuario 3

Historia de usuario			
ID Requisito			
ID	HU003	Usuario	Desarrollador
Nombre de historia	Selección paleta de colores que manejará el aplicativo		
Prioridad	Alta	Tiempo estimado	1
Responsable	Diego Galindo		
Descripción	Tomar como referencia de colores los incorporados en la marca en su Brand Book.		
Observación	Se solicita el logo del Consultorio Dental “Odontólogo Juan Carlos León”		

Elaborador por: Investigador

Tabla 40: Historia de usuario 4

Historia de usuario			
ID Requisito		RF4	
ID	HU004	Usuario	Administrador Odontólogo Secretaria

Nombre de historia	Pantalla Iniciar Sesión		
Prioridad	Alta	Tiempo estimado	3
Responsable	Diego Galindo		
Descripción	La aplicación requiere de un módulo para que cada usuario inicie sesión.		
Observación	La autenticación en la aplicación se realiza mediante la cédula y contraseña las cuales son validadas.		

Elaborado por: Investigador

Tabla 41: Historia de usuario 5

Historia de usuario			
ID Requisito		RF5	
ID	HU005	Usuario	Administrador
Nombre de historia	Gestión de usuarios		
Prioridad	Alta	Tiempo estimado	5
Responsable	Diego Galindo		
Descripción	Se pueden visualizar todos los usuarios, allí podrán buscar el usuario de manera rápida. Asignando roles a cada usuario para gestionar los módulos permitidos a cada rol.		
Observación	Los usuarios no se pueden eliminar si poseen registros asociados. Se puede realizar la actualización y eliminación de estos registros.		

Elaborado por: Investigador

Tabla 42: Historia de usuario 6

Historia de usuario			
ID Requisito		RF6	
ID	HU006	Usuario	Administrador
Nombre de historia	Gestión de roles		
Prioridad	Alta	Tiempo estimado	4
Responsable	Diego Galindo		
Descripción	Los usuarios pueden asignar roles para los usuarios y a su vez gestionar los privilegios que posee cada rol para brindar acceso a los módulos de la aplicación.		
Observación	Los roles no podrán ser eliminados si existen usuarios con referencia a ese rol.		

Elaborado por: Investigador

Tabla 43: Historia de usuario 7

Historia de usuario			
ID Requisito		RF3	
ID	HU007	Usuario	Administrador
Nombre de historia	Gestión Odontólogos		
Prioridad	Alta	Tiempo estimado	2
Responsable	Diego Galindo		
Descripción	Los usuarios gestionarán los datos de los odontólogos, donde se asignará también la especialidad.		

Observación	Los odontólogos no pueden ser eliminados. Un odontólogo puede tener una o más especialidades
-------------	---

Elaborado por: Investigador

Tabla 44: Historia de usuario 8

Historia de usuario			
ID Requisito		RF7	
ID	HU008	Usuario	Desarrollador
Nombre de historia	Gestión especialidades		
Prioridad	Alta	Tiempo estimado	1
Responsable	Diego Galindo		
Descripción	Los usuarios pueden asignar especialidades a cada uno de los odontólogos, a su vez pueden ir generar nuevas especialidades o editar las existentes.		
Observación	Una especialidad no puede ser eliminada si existen registros asociados.		

Elaborado por: Investigador

Tabla 45: Historia de usuario 9

Historia de usuario			
ID Requisito		RF2	
ID	HU009	Usuario	Administrador Odontólogo Secretaria
Nombre de historia	Gestión pacientes		

Prioridad	Alta	Tiempo estimado	5
Responsable	Diego Galindo		
Descripción	Los usuarios pueden visualizar todos los pacientes y buscar los registros rápidamente. Además de realizar la actualización y eliminación de estos.		
Observación	Los pacientes no serán eliminados solo podrán cambiar de estado activo a inactivo.		

Elaborado por: Investigador

Tabla 46: Historia de usuario 10

Historia de usuario			
ID Requisito		RF1	
ID	HU010	Usuario	Administrador Odontólogo
Nombre de historia	Gestión de las Historias Clínicas		
Prioridad	Alta	Tiempo estimado	8
Responsable	Diego Galindo		
Descripción	Los usuarios gestionarán las historias clínicas de cada paciente. Brindará la opción de filtrar las Historias Clínicas por paciente.		
Observación	Las Historias Clínicas no se puede eliminar pueden tendrán estados pendiente, completada, cancelada. El número de historia será un auto incremental generado en caso de o existir cédula del paciente.		

Elaborado por: Investigador

Tabla 47: Historia de usuario 11

Historia de usuario			
ID Requisito		RF3	
ID	HU011	Usuario	Administrador Secretaria
Nombre de historia	Gestión citas		
Prioridad	Alta	Tiempo estimado	5
Responsable	Diego Galindo		
Descripción	Los usuarios pueden gestionar las citas asignadas a cada doctor y paciente. Brindará la opción de filtrar las citas médicas por especialidad.		
Observación	Las citas no se pueden eliminar tendrán los estados: pendiente, completada, cancelada.		

Elaborado por: Investigador

Se procede a realizar un resumen de los requisitos funcionales definiendo cuales son las historias de usuario que se han utilizado para cumplirlos.

Tabla 48: Resumen requisitos funcionales junto a sus historias de usuario

ID Requisito Funcional	ID	Historia de Usuario
RF1	HU010	Gestión de las Historias Clínicas
RF2	HU009	Gestión pacientes
RF3	HU007	Gestión Odontólogos

	HU011	Gestión citas
RF4	HU004	Pantalla Iniciar Sesión
RF5	HU005	Gestión de usuarios
RF6	HU006	Gestión de roles
RF7	HU008	Gestión especialidades

Elaborado por: Investigador

Se muestran las historias de usuario que no cumplen con un requisito funcional específico, pero son necesarias para el desarrollo del sistema.

Tabla 49: Resumen historias de usuario sin requisito funcional

ID Requisito Funcional	ID	Historia de Usuario
-	HU001	Análisis y diseño de la base de datos
	HU002	Estructura de proyectos servidor y cliente
	HU003	Selección paleta de colores que manejará el aplicativo

Elaborado por: Investigador

Tareas

Se procede a generar tareas y subtareas a partir de las historias de usuario que se han generado en cada una de estas se asignará un responsable, historia de usuario a la que pertenece y tiempo estimado.

Tabla 50: Plantilla Tarea

Tarea					
ID		ID HU		Tiempo	
Responsable					
Nombre					
Descripción					

Elaborado por: Investigador

Donde:

- **ID:** Identificador de la tarea.
- **ID HU:** Identificador de la Historia de Usuario.
- **Tiempo:** Tiempo estimado en días en que culminara la tarea.
- **Responsable:** Persona encargada de ejecutar la tarea.
- **Nombre:** Título de la tarea.
- **Descripción:** Síntesis de la tarea.

Una vez se han especificado las definiciones para cada campo manejado en las tareas, a continuación, se muestran cada una de estas.

Tabla 51: Tarea 1

Tarea					
ID	001	ID HU	HU001	Tiempo	6
Responsable		Desarrollador			

Nombre	Definir la base de datos
Descripción	Analizar el modelo de negocio y definir las entidades para generar el MER, modelo lógico y modelo físico.

Elaborado por: Investigador

Tabla 52: Tarea 2

Tarea					
ID	002	ID HU	HU002	Tiempo	1
Responsable		Desarrollador			
Nombre	Estructurar el proyecto del cliente				
Descripción	Establecer los plugins necesarios en el Visual Studio Code, las librerías que se utilizarán y las carpetas para destinar los distintos recursos.				

Elaborado por: Investigador

Tabla 53: Tarea 3

Tarea					
ID	003	ID HU	HU002	Tiempo	1
Responsable		Desarrollador			
Nombre	Estructurar el proyecto del servidor				
Descripción	Establecer los plugins necesarios en el Visual Studio Code, las librerías que se utilizarán y las carpetas para destinar los distintos recursos. Y generar las carpetas para la arquitectura del servidor MVC.				

Elaborado por: Investigador

Tabla 54: Tarea 4

Tarea					
ID	004	ID HU	HU003	Tiempo	1
Responsable		Desarrollador			
Nombre	Analizar el Brand Book y contraste de colores				
Descripción	Analizar los colores que permitan mejorar de mejor manera la información y contrasten con los de la marca.				

Elaborado por: Investigador

Tabla 55: Tarea 5

Tarea					
ID	005	ID HU	HU003	Tiempo	1
Responsable		Desarrollador			
Nombre	Definir los colores del aplicativo				
Descripción	Establecer los colores a utilizar en el aplicativo				

Elaborado por: Investigador

Tabla 56: Tarea 6

Tarea					
ID	006	ID HU	HU004	Tiempo	1
Responsable		Desarrollador			
Nombre	Diseñar la pantalla de iniciar sesión				

Descripción	Esta interfaz debe ser amigable con el usuario.
-------------	---

Elaborado por: Investigador

Tabla 57: Tarea 7

Tarea					
ID	007	ID HU	HU004	Tiempo	1
Responsable		Desarrollador			
Nombre	Validación de campos y funcionalidad				
Descripción	Establecer las validaciones de los campos y funcionalidades de la interfaz.				

Elaborado por: Investigador

Tabla 58: Tarea 8

Tarea					
ID	008	ID HU	HU004	Tiempo	2
Responsable		Desarrollador			
Nombre	Desarrollar la API de autenticación				
Descripción	Se realizará la ruta de autenticación en el servidor utilizando el consumo de JWT para implementar una autenticación basada en tokens.				

Elaborado por: Investigador

Tabla 59: Tarea 9

Tarea					
ID	009	ID HU	HU004	Tiempo	1
Responsable		Desarrollador			
Nombre	Consumir la información para la autenticación desde la base de datos				
Descripción	Se desarrollará las consultas necesarias en la base de datos aplicando las tecnologías disponibles como: funciones, procedimientos almacenados, triggers, etc.				

Elaborado por: Investigador

Tabla 60: Tarea 10

Tarea					
ID	010	ID HU	HU005	Tiempo	2
Responsable		Desarrollador			
Nombre	Diseñar la pantalla de usuarios				
Descripción	Aquí se realizará todas las acciones sobre los usuarios incluyendo dar de alta o baja un usuario.				

Elaborado por: Investigador

Tabla 61: Tarea 11

Tarea					
ID	011	ID HU	HU005	Tiempo	1
Responsable		Desarrollador			

Nombre	Validación de los campos y funcionalidad
Descripción	Establecer normas de datos que serán permitidos dentro de los usuarios, además de permitir la asignación de roles.

Elaborado por: Investigador

Tabla 62: Tarea 12

Tarea					
ID	012	ID HU	HU005	Tiempo	1
Responsable		Desarrollador			
Nombre	Desarrollar la API de usuarios				
Descripción	Se desarrollará la API para gestionar los usuarios con gestión de credenciales.				

Elaborado por: Investigador

Tabla 63: Tarea 13

Tarea					
ID	013	ID HU	HU005	Tiempo	1
Responsable		Desarrollador			
Nombre	Consumir la información de usuarios para la API desde la base de datos				
Descripción	Se desarrollará las consultas necesarias en la base de datos aplicando las tecnologías disponibles como: funciones, procedimientos almacenados, triggers, etc. Utilizando las tablas necesarias.				

Elaborado por: Investigador

Tabla 64: Tarea 14

Tarea					
ID	014	ID HU	HU006	Tiempo	1
Responsable		Desarrollador			
Nombre	Diseñar la pantalla de roles				
Descripción	Aquí se gestionará los roles que se encuentran activos en inactivos y asignar acceso a los módulos a cada rol				

Elaborado por: Investigador

Tabla 65: Tarea 15

Tarea					
ID	015	ID HU	HU006	Tiempo	1
Responsable		Desarrollador			
Nombre	Validación de los campos y funcionalidad				
Descripción	Establecer normas de datos que serán permitidos dentro de los roles y gestionar los privilegios que pueden tener.				

Elaborado por: Investigador

Tabla 66: Tarea 16

Tarea					
ID	016	ID HU	HU006	Tiempo	2
Responsable		Desarrollador			
Nombre	Desarrollar la API de roles				

Descripción	Se desarrollará la API para gestionar los roles disponibles en el sistema.
-------------	--

Elaborado por: Investigador

Tabla 67: Tarea 17

Tarea					
ID	017	ID HU	HU006	Tiempo	1
Responsable		Desarrollador			
Nombre	Consumir la información de roles para la API desde la base de datos				
Descripción	Se desarrollará las consultas necesarias en la base de datos aplicando las tecnologías disponibles como: funciones, procedimientos almacenados, triggers, etc.				

Elaborado por: Investigador

Tabla 68: Tarea 18

Tarea					
ID	018	ID HU	HU007	Tiempo	1
Responsable		Desarrollador			
Nombre	Diseñar la pantalla de odontólogos				
Descripción	Realizar todas las acciones sobre los usuarios incluyendo dar de alta o baja un usuario.				

Elaborado por: Investigador

Tabla 69: Tarea 19

Tarea					
ID	019	ID HU	HU007	Tiempo	1
Responsable		Desarrollador			
Nombre	Validación de los campos y funcionalidad				
Descripción	Establecer normas de datos que serán permitidos dentro de los odontólogos y gestionar las especialidades asignadas a cada uno de ellos.				

Elaborado por: Investigador

Tabla 70: Tarea 20

Tarea					
ID	020	ID HU	HU007	Tiempo	1
Responsable		Desarrollador			
Nombre	Desarrollar la API de odontólogos				
Descripción	Se desarrollará las consultas necesarias en la base de datos aplicando las tecnologías disponibles como: funciones, procedimientos almacenados, triggers, etc. Se aplicará transaccionalidad en caso de ser necesario.				

Elaborado por: Investigador

Tabla 71: Tarea 21

Tarea					
ID	021	ID HU	HU007	Tiempo	1
Responsable		Desarrollador			

Nombre	Consumir la información de odontólogos para la API desde la base de datos
Descripción	Se desarrollará las consultas necesarias en la base de datos aplicando las tecnologías disponibles como: funciones, procedimientos almacenados, triggers, etc. Utilizando la transaccionalidad para proteger la integridad de la información.

Elaborado por: Investigador

Tabla 72: Tarea 22

Tarea					
ID	022	ID HU	HU008	Tiempo	1
Responsable		Desarrollador			
Nombre	Desarrollar la API de especialidades				
Descripción	Se desarrollará la API para gestionar las especialidades odontológicas.				

Elaborado por: Investigador

Tabla 73: Tarea 23

Tarea					
ID	023	ID HU	HU008	Tiempo	1
Responsable		Desarrollador			
Nombre	Consumir la información de especialidades para la API.				
Descripción	Se desarrollará las consultas necesarias en la base de datos aplicando las tecnologías disponibles como: funciones, procedimientos almacenados, triggers, etc.				

Elaborado por: Investigador

Tabla 74: Tarea 24

Tarea					
ID	024	ID HU	HU009	Tiempo	3
Responsable		Desarrollador			
Nombre	Diseñar la pantalla de pacientes				
Descripción	Esta interfaz debe ser amigable con el usuario. Aquí se gestionarán los pacientes.				

Elaborado por: Investigador

Tabla 75: Tarea 25

Tarea					
ID	025	ID HU	HU009	Tiempo	1
Responsable		Desarrollador			
Nombre	Validación de los campos y funcionalidad				
Descripción	Establecer normas de datos que serán permitidos para los nombres, teléfono, etc.				

Elaborado por: Investigador

Tabla 76: Tarea 26

Tarea					
ID	026	ID HU	HU009	Tiempo	2
Responsable		Desarrollador			
Nombre	Desarrollar la API de pacientes				

Descripción	Se desarrollará la API para gestionar los usuarios con su gestión de roles.
-------------	---

Elaborado por: Investigador

Tabla 77: Tarea 27

Tarea					
ID	027	ID HU	HU009	Tiempo	1
Responsable		Desarrollador			
Nombre	Consumir la información de pacientes para la API desde la base de datos				
Descripción	Se desarrollará las consultas necesarias en la base de datos aplicando las tecnologías disponibles como: funciones, procedimientos almacenados, triggers, etc. Utilizando las tablas necesarias.				

Elaborado por: Investigador

Tabla 78: Tarea 28

Tarea					
ID	028	ID HU	HU010	Tiempo	3
Responsable		Desarrollador			
Nombre	Diseñar la pantalla de las Historias Clínicas				
Descripción	Realizar la interfaz donde se observarán las historias clínicas que posee el establecimiento, creación de odontogramas índices CEO-CPO, etc. Observas las sesiones asociadas a las Historias Clínicas.				

Elaborado por: Investigador

Tabla 79: Tarea 29

Tarea					
ID	029	ID HU	HU010	Tiempo	2
Responsable		Desarrollador			
Nombre	Validación de los campos, manejo de errores y funcionalidad de botones interactuando con las interfaz				
Descripción	Establecer la longitud de los campos a ingresar y mensajes de error.				

Elaborado por: Investigador

Tabla 80: Tarea 30

Tarea					
ID	030	ID HU	HU010	Tiempo	4
Responsable		Desarrollador			
Nombre	Desarrollar la API de Historias Clínicas				
Descripción	Se desarrollará las consultas necesarias en la base de datos aplicando las tecnologías disponibles como: funciones, procedimientos almacenados, triggers, etc.				

Elaborado por: Investigador

Tabla 81: Tarea 31

Tarea					
ID	031	ID HU	HU010	Tiempo	1
Responsable		Desarrollador			

Nombre	Consumir la información de las Historias Clínicas para la API desde la base de datos
Descripción	Se desarrollará las consultas necesarias en la base de datos aplicando las tecnologías disponibles como: funciones, procedimientos almacenados, triggers, etc. Utilizando la transaccionalidad para proteger la integridad de la información.

Elaborado por: Investigador

Tabla 82: Tarea 32

Tarea					
ID	032	ID HU	HU011	Tiempo	2
Responsable		Desarrollador			
Nombre	Diseñar la pantalla de citas				
Descripción	Realizar la interfaz donde se puede ver las citas médicas asignadas, generar nuevas y cambiar el estado de cada una de estas.				

Elaborado por: Investigador

Tabla 83: Tarea 33

Tarea					
ID	033	ID HU	HU011	Tiempo	1
Responsable		Desarrollador			
Nombre	Validación de los campos y funcionalidad				
Descripción	Establecer la longitud de los campos a ingresar y mensajes de error.				

Elaborado por: Investigador

Tabla 84: Tarea 34

Tarea					
ID	034	ID HU	HU011	Tiempo	1
Responsable		Desarrollador			
Nombre	Desarrollar la API de citas				
Descripción	Se desarrollará las consultas necesarias en la base de datos aplicando las tecnologías disponibles como: funciones, procedimientos almacenados, triggers, etc. Se aplicará transaccionalidad en caso de ser necesario.				

Elaborado por: Investigador

Tabla 85: Tarea 35

Tarea					
ID	035	ID HU	HU011	Tiempo	1
Responsable		Desarrollador			
Nombre	Consumir la información de citas para la API desde la base de datos				
Descripción	Se desarrollará las consultas necesarias en la base de datos aplicando las tecnologías disponibles como: funciones, procedimientos almacenados, triggers, etc. Utilizando la transaccionalidad para proteger la integridad de la información.				

Elaborado por: Investigador

Con la metodología seleccionada la cual consiste en la presentación de prototipos funcionales en el menor tiempo posible cada iteración posee tiempos distintos según el tiempo estimado de las tareas asignadas. Las iteraciones fueron planificadas de la siguiente manera con una duración estimada de 7 días. Por lo tanto, se planean

presentación de prototipos semanalmente ya que se desarrollará todos los días de la semana. A continuación, se detallan cada una de las Historias de usuario con la iteración a la que se designó,

Tabla 86: Resumen de Historias de usuario

Iteración	ID	Historia de Usuario	Prioridad	Tiempo estimado
1	HU001	Análisis y diseño de la base de datos	Alta	6
	HU002	Estructura de proyectos servidor y cliente	Alta	1
	HU003	Selección paleta de colores que manejará el aplicativo.	Alta	1
2	HU004	Pantalla Iniciar Sesión	Alta	3
	HU005	Gestión de usuarios	Alta	5
3	HU006	Gestión de roles	Alta	4
	HU007	Gestión Odontólogos	Alta	2
	HU008	Gestión especialidades	Alta	1
4	HU009	Gestión pacientes	Alta	5
5	HU010	Gestión de las Historias Clínicas	Alta	8
6	HU011	Gestión citas	Alta	5
Total, tiempo de estimación (días)				41

Elaborado por: Investigador

Se muestra las historias de usuario desglosadas con las tareas respectivas que se cumplirán para que cada historia sea finalizada correctamente junto con los plazos de tiempo de estas.

Tabla 87: Resumen tareas

Número Iteración	ID		Nombre de Historias de usuario y tareas	Tiempo
	Historia	Tarea		
1	HU001		Análisis y diseño de la base de datos	
		001	Definir la base de datos	6
	HU002		Estructura de proyectos servidor y cliente	
		002	Estructurar el proyecto del cliente	1
		003	Estructurar el proyecto del servidor	1
	HU003		Selección paleta de colores que manejará el aplicativo.	
		004	Analizar el Brand Book y contraste de colores	1
		005	Definir los colores del aplicativo	1
2	HU004		Pantalla Iniciar Sesión	
		006	Diseñar la pantalla de iniciar sesión	1
		007	Validación de campos y funcionalidad	1
		008	Desarrollar la API de autenticación	2
		009	Consumir la información para la autenticación desde la base de datos	1
	HU005		Gestión de usuarios	
		010	Diseñar la pantalla de usuarios	2
		011	Validación de los campos y funcionalidad	1
		012	Desarrollar la API de usuarios	1

		013	Consumir la información de usuarios para la API desde la base de datos	1
3	HU006		Gestión de roles	
		014	Diseñar la pantalla de roles	1
		015	Validación de los campos y funcionalidad	1
		016	Desarrollar la API de roles	2
		017	Consumir la información de roles para la API desde la base de datos	1
	HU007		Gestión Odontólogos	
		018	Diseñar la pantalla de odontólogos	1
		019	Validación de los campos y funcionalidad	1
		020	Desarrollar la API de odontólogos	1
		021	Consumir la información de odontólogos para la API desde la base de datos	1
	HU008		Gestión especialidades	
		022	Desarrollar la API de especialidades	1
		023	Consumir la información de especialidades para la API desde la base de datos	1
4	HU009		Gestión pacientes	
		024	Diseñar la pantalla de pacientes	3
		025	Validación de los campos y funcionalidad	1
		026	Desarrollar la API de pacientes	2
		027	Consumir la información de usuarios para la API desde la base de datos	1

5	HU010		Gestión de las Historias Clínicas	
		028	Diseñar la pantalla de las Historias Clínicas	3
		029	Validación de los campos, manejo de errores y funcionalidad de botones interactuando con las interfaz	2
		030	Desarrollar la API de Historias Clínicas	4
		031	Consumir la información de las Historias Clínicas para la API desde la base de datos	1
6	HU011		Gestión citas	
		032	Diseñar la pantalla de citas	2
		033	Validación de los campos y funcionalidad	1
		034	Desarrollar la API de citas	1
		035	Consumir la información de citas para la API desde la base de datos	1

Elaborado por: Investigador

Diseño con el usuario y construcción

Iteración 1

Tabla 88: Resumen iteración 1

Iteración	ID	Nombre	Prioridad	Tiempo
1	HU001	Análisis y diseño de la base de datos	Alta	6
1	HU002	Estructura de proyectos servidor y cliente	Alta	1

1	HU003	Selección paleta de colores que manejará el aplicativo.	Alta	1
Tiempo estimado en días				8

Elaborado por: Investigador

Modelado de la base de Datos

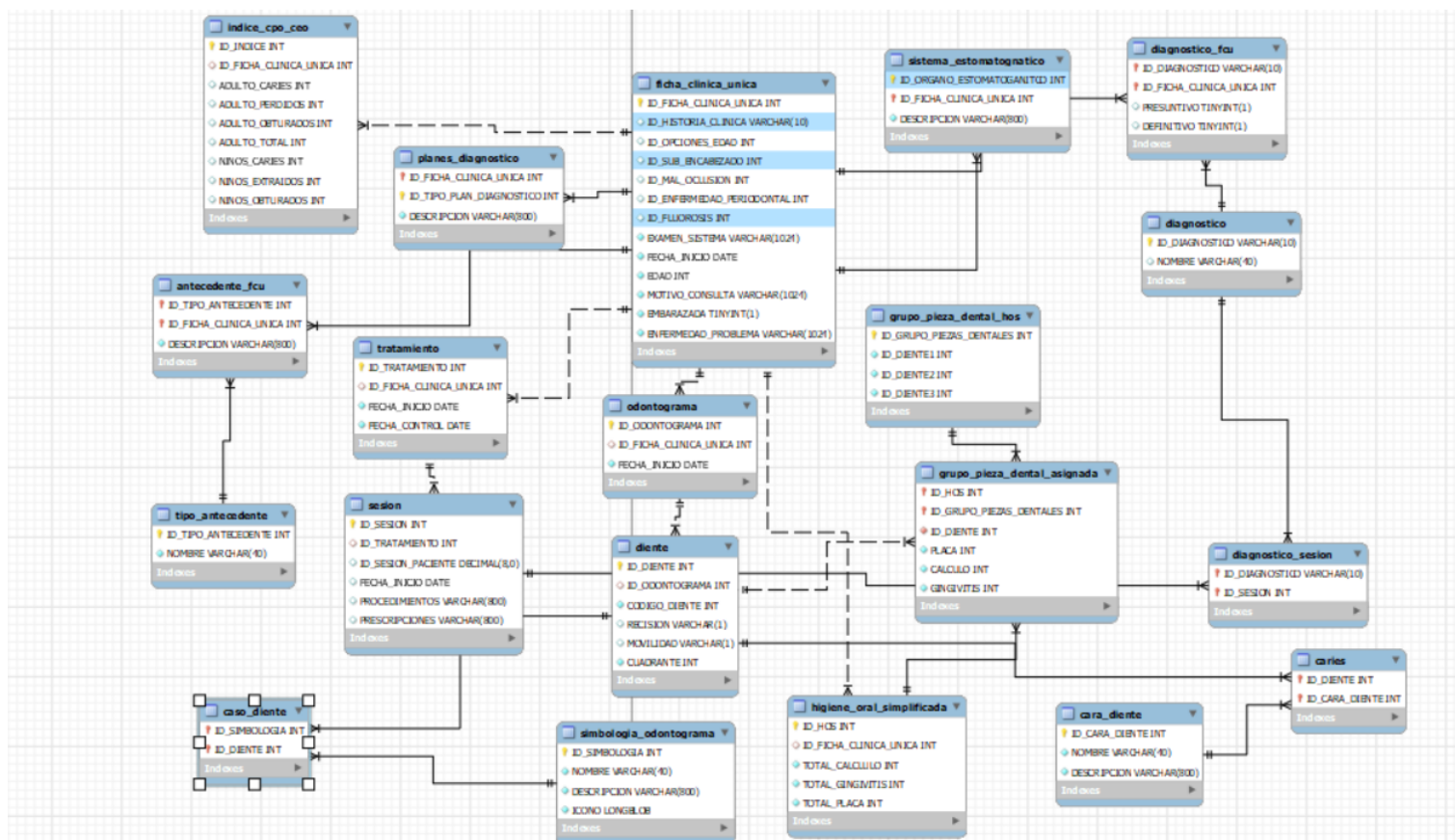
Se realiza la fase de análisis de la base de datos en el cual se toma en cuenta como eje principal o punto de partida que la creación de una historia clínica enfocada en el campo odontológico que se basa en el Formato 033 (**Anexo A**) que dicta el Ministerio de Salud Pública del Ecuador de acuerdo con su normativa vigente hasta el año de la elaboración de este proyecto de titulación.

Para la elaboración de la base de datos se realizó una fase previa de Diseño Conceptual donde se tomó la decisión de separar las partes principales:

- Signos Vitales
- Higiene Oral Simplificada
- Sistema Estomatognático
- Planes de diagnóstico
- Índice CPO CEO
- Odontograma
- Antecedentes
- Tratamiento

Normalizando la base de datos de la mejor manera. Se han generado 42 tablas dentro de la base de datos, se debe tomar en cuenta que para que la base de datos sea tan grande existe un gran número de tablas que son de configuración de parámetros o a su vez se derivan de las partes principales mencionadas en el párrafo anterior. Para el modelo completo de la base de datos observar el **Anexo B**.

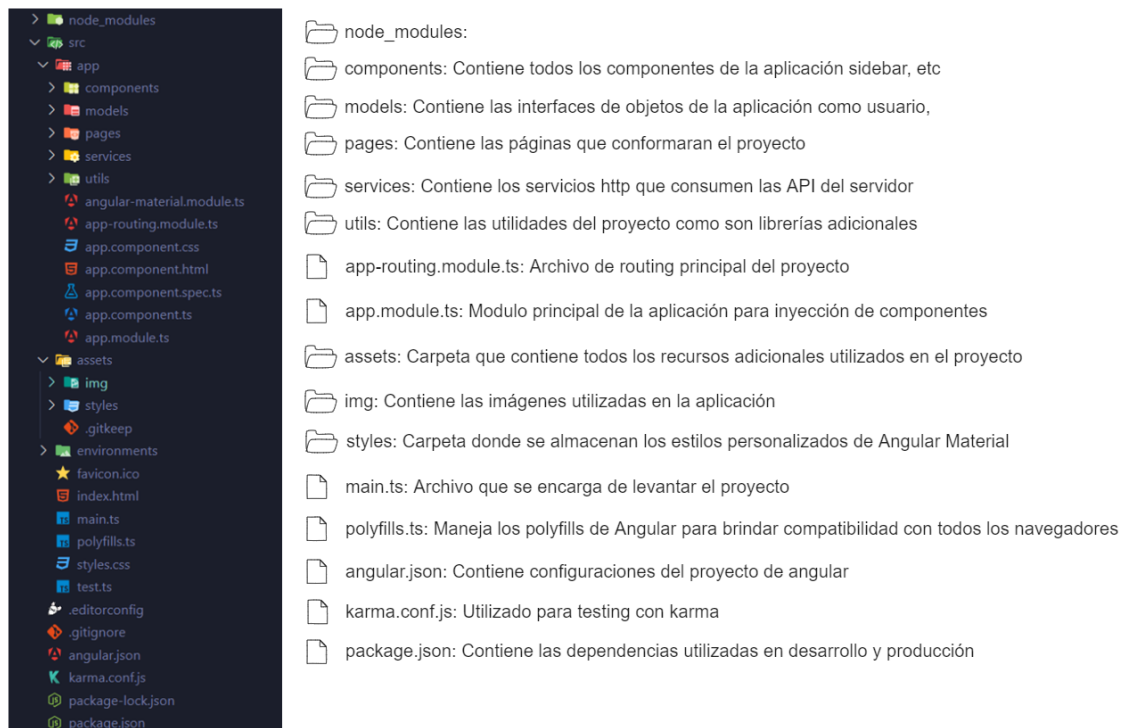
Figura 6: Fragmento de la base de datos



Elaborado por: Investigador

Una vez se instaló la herramienta Angular en este caso se trabajará con la versión 16.2.0 para realizar este proyecto. Instalando la herramienta a través de la consola. Se generan las estructura, carpetas y documentos que se van a utilizar dentro del proyecto web en el lado del cliente.

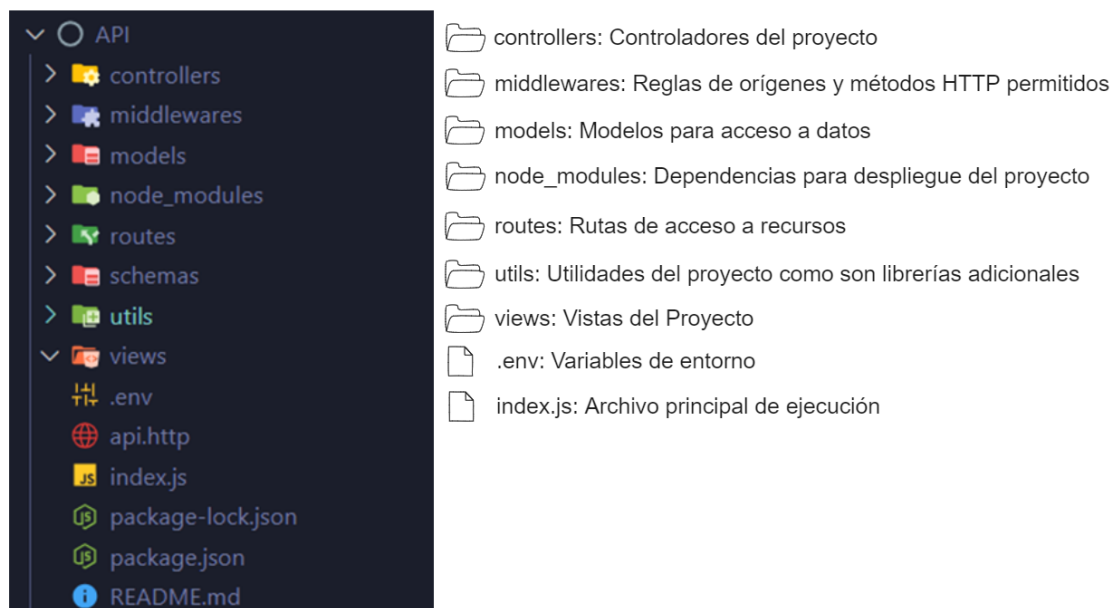
Figura 7: Estructura del proyecto en el cliente



Elaborado por: Investigador

Una vez definida la estructura del proyecto del cliente nos dirigimos al lado del servidor donde se trabajará con NodeJS se utilizará el MVC por lo que la estructura del proyecto queda definida de la siguiente manera en lado del servidor.

Figura 8: Estructura del proyecto en el servidor



Elaborado por: Investigador

Definir la paleta de colores

Se toma como referencia los 2 logos más utilizados por el establecimiento de salud. Tomando como referencia los colores que no tengan problemas de contrastes y puedan facilitar la lectura de las Historias Clínicas sin cansancio excesivo, pero tampoco se puede realizar el diseño de la interfaz con colores oscuros porque al ser un sistema de salud requiere colores claros que logren una completa atención en la digitación de la información.

Figura 9: Logos más utilizados del Consultorio Dental



Elaborado por: Investigador

Se utilizó la herramienta Adobe Color para generar una paleta con colores de contraste opuestos. Tomando como eje central de los tonos el celeste. En la paleta se seleccionó colores opuestos que hagan contraste entre sí.

Figura 10: Paleta de colores del sistema



Elaborado por: Investigador

Iteración 2

Tabla 89: Resumen iteración 2

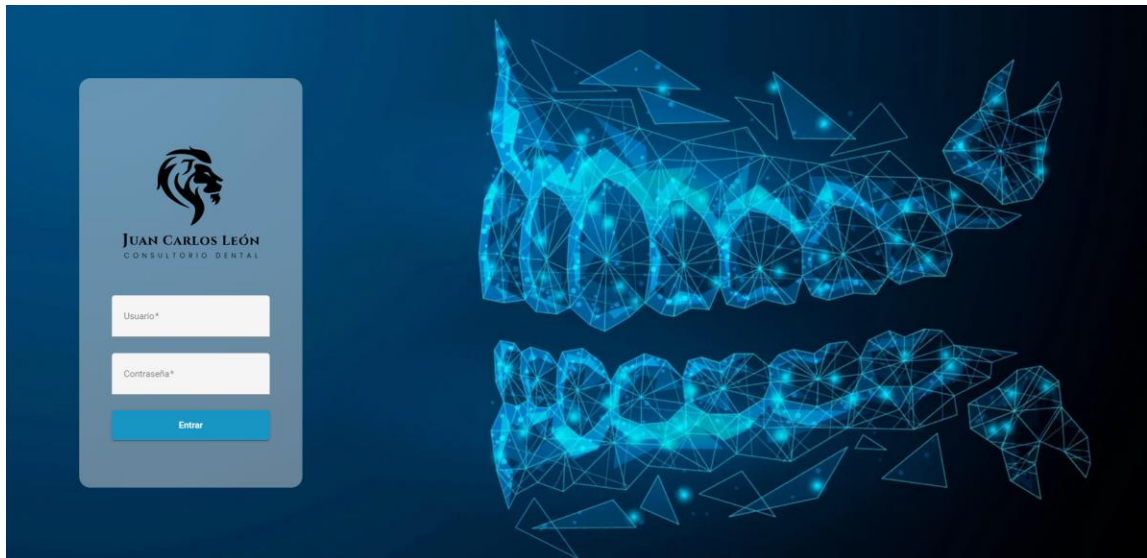
Iteración	ID	Nombre	Prioridad	Tiempo
2	HU004	Pantalla Iniciar Sesión	Alta	3
2	HU005	Gestión de usuarios	Alta	5
Tiempo estimado en días				8

Elaborado por: Investigador

Diseño de Iniciar Sesión

Desde la aplicación se permite el inicio de sesión a odontólogos, administradores y secretaria través del usuario y la contraseña. Se colocó como fondo de pantalla una foto alusiva a la disciplina colocando los campos de ingreso para el usuario y la contraseña junto al logo del Consultorio Dental “Juan Carlos León”, al ocurrir un error en la validación de los datos se muestra un panel de error que ha sido implementado con la librería OpenSource SweetAlert. El resultado fue el siguiente.

Figura 11: Pantalla Iniciar Sesión



Elaborado por: Investigador

Para la creación del formulario HTML se utilizó la librería de Angular Material en su versión 15.4. Se definen los campos donde se asigna un FormControl que servirá para el acceso unidireccional a la información junto a su botón asignando en el evento clic el método login.

Figura 12: HTML de pantalla Iniciar Sesión

```
1 <div class="contenedor">
2   <form [formGroup]="form">
3     <div class="formulario">
4       <div class="formulario_logo">
5         
8       </div>
9       <div class="formulario_campos">
10        <mat-form-field class="formulario_campoTexto" appearance="fill">
11          <mat-label>Usuario</mat-label>
12          <input matInput formControlName="dni" autocomplete="off">
13        </mat-form-field>
14        <mat-form-field class="formulario_campoTexto" appearance="fill">
15          <mat-label>Contraseña</mat-label>
16          <input type="password" matInput formControlName="clave">
17        </mat-form-field>
18        <button class="formulario_boton" mat-raised-button color="primary" (click)="login()">Entrar</button>
19      </div>
20    </div>
21  </form>
22 </div>
```

Elaborado por: Investigador

Generar token

Para la autenticación de usuario entre muchas de las tecnologías disponibles se optó por utilizar un token para cada sesión con una duración de 24 horas debido a larga jornada laboral del profesional de la salud. Se utilizó la biblioteca JWT. Para lo cual se generó la clase JWT dentro del servidor para la generación del token con el tiempo establecido y los datos que contiene el token en este caso el usuario y el tiempo en que expira.

Figura 13: Clase de JWT

```
export class JWT{
  static crearToken(data) {
    const { dni } = data;
    const token = jwt.sign({
      usuario: dni
    }, process.env.SECRET_TOKEN , {expiresIn: '24h'});

    return token;
  }

  static verificarToken(token){
    const verify = jwt.verify(token, process.env.SECRET_TOKEN);
    return verify;
  }
}
```

Elaborado por: Investigador

Se realiza el método de login que fue asignado al botón de ingresar dentro de la interfaz de iniciar sesión, para lo cual se procedió a realizar el proceso en el servidor en las distintas capas en las cuales se dividió la estructura de carpetas del servidor Node JS.

Figura 14: Método login del servidor 1/2

```

static async login({ data }) {
  try {
    const { dni, clave } = data;
    const [respuesta, fields] = await connection.query(
      `CALL sp_login_getUser(?,?);`,
      [dni, clave]
    );

    const filas = respuesta[0]
    return filas[0];
  } catch (error) {
    console.log(error);
  }
}

//FIXME: Cambiar el campo last_login de date a date time
/**
 * Funcion que actualiza el campo token en el
 * nombre de usuario correspondiente
 * @param { data }
 * @returns update
 */
static async actualizarToken({ data }) {
  const { id_usuario, token_sesion } = data;
  console.log(data);
  const update = await connection.query(
    `CALL sp_login_updateToken(?,?)`,
    [id_usuario, token_sesion]
  );
}

```

Elaborado por: Investigador

Figura 15: Método login del servidor 2/2

```

const resultado = validateLogin(req.body);
console.log(resultado);
if (resultado.error) {
  return res.status(200).json({
    success: false,
    error: JSON.parse(resultado.error),
    message: 'Error campos no validos',
  });
}

const respuesta = (await LoginModel.login({ data: resultado.data }));
if (!respuesta || respuesta < 1) { ...
}

const token = JWT.crearToken(resultado.data);
const user = respuesta;
user.TOKEN_SESION = token;
console.log(user);
await LoginModel.actualizarToken({ data: user });

return res.json({
  success: true,
  data: { user: user }
});

} catch (error) {
  console.log(error);
  return res.status(200).json({
    success: false,
    error: "Error al loguear"
  });
}

```

Elaborado por: Investigador

Módulo Gestión apartado usuarios

La interfaz de usuario permite muestra en una tabla todos los usuarios con su estado además de contar con la opción de una búsqueda ágil a través de campo de texto “Buscar” y la paginación de los registros devueltos. Se generó un barra amarilla donde se colocarán los submenús. Los usuarios pueden realizar las acciones de un CRUD sobre la interfaz. Se maneja un flujo de datos unidireccional para mejorar la seguridad. El resultado es el siguiente:

Figura 16: Interfaz de gestión de usuarios

The screenshot displays a web application interface for user management. At the top, a blue header bar contains a menu icon, the user name "Dr. Juan Carlos", and a logout icon. Below this is an orange navigation bar with links for "Usuarios", "Roles", "Odontólogos", and "Especialidades". The main content area is divided into two sections. The top section, titled "Usuario", contains a form with fields for "Nombre*", "Apellido*", "Usuario*", and "Contraseña*", along with a dropdown for "Rol del usuario*" (Selecione...). Below the form are "Guardar" and "Cancelar" buttons. The bottom section features a search bar labeled "Buscar" and a table of users. The table has columns for "Usuario", "Estado", "Rol", and "Acción". It lists two users: "Diego Fernando Galindo Vasconez" and "Juan Fernandez Ramirez Aquiles", both with an "Estado" of "Activo" (indicated by a green checkmark) and a "Rol" of "administrador". The "Acción" column contains edit icons. At the bottom right, there is a pagination control showing "Registros por página" (5) and "1 - 2 de 2".

Usuario	Estado	Rol	Acción
Diego Fernando Galindo Vasconez	Activo	administrador	[Edit]
Juan Fernandez Ramirez Aquiles	Activo	administrador	[Edit]

Elaborado por: Investigador

Una parte importante en la seguridad es la validación de datos que ingresan a la aplicación donde el primer filtro de validación ocurre en el lado de la aplicación, por lo que se realizó una validación de los distintos campos utilizando expresiones lambda para restringir el uso de minúsculas, mayúsculas y números.

Figura 17: Creación del objeto con los datos del formulario

```
getUser(id:string){
  this.userService.getById(id).subscribe((data) => {
    this.user$ = data[0];
    this.loading = false;
    this.createForm();
  });
}

createForm(){
  this.form = this.fb.group({
    us_first_name : [this.user$ === undefined ? '' : this.user$.primer_nombre, [Validators.required, Validators.pattern('^[A-Z]*)']],
    us_last_name : [this.user$ === undefined ? '' : this.user$.apellido_paterno, [Validators.required, Validators.pattern('^[A-Z]*)']],
    us_user_handle : [this.user$ === undefined ? '' : this.user$.dni, [Validators.required, Validators.pattern('^[a-zA-Z0-9]{5,16}')]],
    us_password : [this.user$ === undefined ? '' : this.user$.clave, [Validators.required, Validators.pattern('^[a-zA-Z0-9]{5,16}')]],
    us_rol_id : [this.user$ === undefined ? '' : this.user$.id_rol, [Validators.required]]
  })
}
```

Elaborado por: Investigador

Se realizó el consumo de los servicios para la manipulación del CRUD que se desarrollaron en la API alojada en el servidor donde se reciben los datos en un formato JSON, y para enviar información se utiliza el body de la petición Http a través de la librería HttpClient que posee Angular dentro del core de utilidades disponibles para el desarrollo web.

Figura 18: Consumo de servicios para manipulación de datos de usuario

```

createUser(){
  SweetAlert.loadingAlert();
  this.userService.create(this.loadData()).subscribe(
    (res:any) => {
      SweetAlert.successAlert("Mensaje" + res.message);
    },
    (err) => {
      SweetAlert.errorAlert("Error:" + err);
    }
  );
}

updateUser(){
  SweetAlert.loadingAlert();
  this.userService.update(this.loadData()).subscribe([
    (res:any) => {
      SweetAlert.successAlert("Mensaje" + res.message);
    },
    (err) => {
      SweetAlert.errorAlert("Error:" + err);
    }
  ]);
}

deleteUser(){
  SweetAlert.loadingAlert();
  this.userService.update(this.loadData()).subscribe(
    (resp:any )=>{
      SweetAlert.successAlert(resp.message);
    },
    (err) => {
      SweetAlert.errorAlert("Error"+err);
    }
  ));
}

```

Elaborado por: Investigador

Iteración 3

Figura 19: Resumen iteración 3

Iteración	ID	Nombre	Prioridad	Tiempo
3	HU006	Gestión de roles	Alta	4
3	HU007	Gestión Odontólogos	Alta	2
3	HU008	Gestión especialidades	Alta	1

Tiempo estimado en días	7
-------------------------	---

Elaborado por: Investigador

Módulo Gestión apartado roles

Se desarrollo la interfaz gráfica con las funcionalidades que permitirán a los usuarios del sistema gestionar los roles de manera adecuada y el acceso a los módulos del sistema a través de un checkbox podrá seleccionar cuales son los módulos a los que tiene permiso el usuario correspondiente. Se muestra una tabla de todos los usuarios generados con los datos esenciales del usuario y permiten ejecutar acciones sobre los mismos.

Figura 20: Interfaz de roles

Rol	Estado	Accion
administrador	✓	

Elaborado por: Investigador

Se realizo el consumo de los servicios para la manipulación del CRUD que se desarrollaron en la API alojada en el servidor donde se reciben los datos en un formato JSON. Se creo el formulario para el manejo unidireccional de la información con los métodos para inserción, creación y actualización.

Figura 21: Consumo de servicios para manipulación de roles

```
createForm(){
  this.form = this.fb.group({
    nombre : [this.rol$ === undefined ? '':this.rol$.nombre, [Validators.required, Validators.pattern('^[A-Z]*)']],
    estado_id : [this.rol$ === undefined ? '':this.rol$.id_estado, [Validators.required]]
  })
}

createRol(){
  SweetAlert.loadingAlert();
  this.rolService.create(this.loadData()).subscribe(
    (res:any) => {
      SweetAlert.successAlert("Mensaje" + res.message);
    },
    (err) => {
      SweetAlert.errorAlert("Error:" + err);
    }
  );
}

updateRol(){
  SweetAlert.loadingAlert();
  this.rolService.update(this.loadData()).subscribe(
    (res:any) => {
      SweetAlert.successAlert("Mensaje" + res.message);
    },
    (err) => {
      SweetAlert.errorAlert("Error:" + err);
    }
  );
}

deleteRol(){
  SweetAlert.loadingAlert();
  this.rolService.delete(this.loadData()).subscribe(
    (resp:any )=>{
      SweetAlert.successAlert(resp.message);
    },
  );
}
```

Elaborado por: Investigador

Módulo Gestión apartado Odontólogos

Se desarrollo la interfaz gráfica con las funcionalidades que permitirán a los usuarios la creación de nuevos odontólogos o gestionar los que ya han sido creados. Así mismo asignándole una especialidad dentro del área odontológica. Uno de los datos importantes a tomar en cuenta y que es esencial es que cada uno de los miembros del personal médico posee un solo código médico que es asignado por el Ministerio de Salud Pública del Ecuador. Se mantiene la congruencia en la distribución de los componentes para mejorar la experiencia de usuario.

Figura 22: Diseño interfaz odontólogos

Dr. Juan Carlos

Usuarios Roles Odontólogos Especialidades

Odontólogo

Nombre* Apellido* Usuario*

Código Médico* Especialidad* Seleccione...

Guardar Cancelar

Doctor

Usuario	Estado	Especialidad	Acción
Diego Fernando Galindo Vasconez	✓	Implantología	✎
Juan Fernandez Ramirez Aquiles	✓	Endodoncia	✎

Registros por página 5 1 - 2 de 2 < >

Elaborado por: Investigador

Se realizó el consumo de los servicios para la manipulación del CRUD que se desarrollaron en la API alojada en el servidor donde se reciben los datos en un formato JSON. Se creó el formulario para el manejo unidireccional de la información con los métodos para inserción, creación y actualización.

Figura 23: Consumo de servicios para manipulación de odontólogos

```

createOdontologo(){
  SweetAlert.loadingAlert();
  this.userService.create(this.loadData()).subscribe(
    (res:any) => {
      SweetAlert.successAlert("Mensaje" + res.message);
    },
    (err) => {
      SweetAlert.errorAlert("Error:" + err);
    }
  );
}

updateOdontologo(){
  SweetAlert.loadingAlert();
  this.userService.update(this.loadData()).subscribe(
    (res:any) => {
      SweetAlert.successAlert("Mensaje" + res.message);
    },
    (err) => {
      SweetAlert.errorAlert("Error:" + err);
    }
  );
}

deleteOdontologo(){
  SweetAlert.loadingAlert();
  this.userService.delete(this.loadData()).subscribe(
    (resp:any) => {
      SweetAlert.successAlert(resp.message);
    },
    (err) => {
      SweetAlert.errorAlert("Error"+err);
    }
  );
}

```

Elaborado por: Investigador

Iteración 4

Tabla 90: Resumen iteración 4

Iteración	ID	Nombre	Prioridad	Tiempo
4	HU009	Gestión pacientes	Alta	5
Tiempo estimado en días				5

Elaborado por: Investigador

Módulo de pacientes

Se desarrollo la interfaz gráfica con las funcionalidades que permitirán a los usuarios del sistema gestionar de manera adecuada todos los pacientes del establecimiento de salud. Este se encuentra en el apartado de pacientes en la barra de menú lateral. Se omitió el submenú amarillo conservando la elegancia y congruencia del sistema. Dando como resultado lo siguiente.

Figura 24: Interfaz de pacientes

Paciente

Primer Nombre* Segundo Nombre* Primer Apellido* Segundo Apellido*

DNI* Teléfono* Dirección* Sexo Est...

Guardar Cancelar

Buscar

Apellidos	Nombres	Teléfono	Sexo	Estado	Accion
Cevallos Andrade	Carlos Alejandro	032852837	M	✓	✎
Palacios Yancha	Jesus Alejandro	032852837	M	✖	✎
Cevallos Fiallos	Daniel Marcelo	032452747	M	✓	✎

Elaborado por: Investigador

Para la validación de los campos del formulario se utilizó expresiones lambda en los validators, en el proyecto se eliminó la opción de utilizar Template Forms y se procedió con el desarrollo de Reactive Forms por cuestiones de manejo unidireccional de datos.

Figura 25: Creación del objeto con los datos del formulario

```
getPaciente(id:string){
  this.pacienteService.getById(id).subscribe((data:any) => {
    this.paciente$ = data[0]
    this.loading = false;
    this.createForm();
  });
}

createForm(){
  this.form = this.fb.group({
    primer_nombre : [this.paciente$ === undefined ? '':this.paciente$.primer_nombre, [Validators.required, Validators.pattern('^[A-Z]*')]],
    segundo_nombre : [this.paciente$ === undefined ? '':this.paciente$.segundo_nombre, [Validators.required, Validators.pattern('^[A-Z]*')]],
    apellido_paterno : [this.paciente$ === undefined ? '':this.paciente$.apellido_paterno, [Validators.required, Validators.pattern('^[A-Z]*')]],
    apellido_materno : [this.paciente$ === undefined ? '':this.paciente$.apellido_materno, [Validators.required, Validators.pattern('^[A-Z]*')]],
    direccion : [this.paciente$ === undefined ? '':this.paciente$.direccion, [Validators.required, Validators.pattern('^[A-Z]*')]],
    dni : [this.paciente$ === undefined ? '':this.paciente$.dni, [Validators.required, Validators.pattern('^[a-zA-Z0-9]{5,16}$')]],
    telefono : [this.paciente$ === undefined ? '':this.paciente$.clave, [Validators.required, Validators.pattern('^[a-zA-Z0-9]{5,16}$')]],
    estado_id : [this.paciente$ === undefined ? '':this.paciente$.estado_id, [Validators.required]],
    sexo_id : [this.paciente$ === undefined ? '':this.paciente$.sexo_id, [Validators.required]]
  })
}
```

Elaborado por: Investigador

Se utilizaron los servicios para interactuar con las operaciones CRUD implementadas en la API, la cual está alojada en un servidor y recibe datos en formato JSON. Además, se diseñó un formulario que permite manejar la información de manera unidireccional, incorporando métodos para la inserción, creación y actualización de datos.

Figura 26: Consumo de servicios para manipulación de pacientes

```
updatePaciente(){
  SweetAlert.loadingAlert();
  this.pacienteService.update(this.loadData()).subscribe(
    (res:any) => {
      SweetAlert.successAlert("Mensaje" + res.message);
    },
    (err) => {
      SweetAlert.errorAlert("Error:" + err);
    }
  );
}

deletePaciente(){
  SweetAlert.loadingAlert();
  this.pacienteService.update(this.loadData()).subscribe(
    (resp:any )=>{
      SweetAlert.successAlert(resp.message);
    },
    (err) => {
      SweetAlert.errorAlert("Error"+err);
    }
  );
}

loadData(){
  let savePaciente = {
    ...this.paciente$,
    primer_nombre : this.form.get('primer_nombre')?.value,
    segundo_nombre : this.form.get('segundo_nombre')?.value,
    apellido_paterno : this.form.get('apellido_paterno')?.value,
    apellido_materno : this.form.get('apellido_materno')?.value,
    direccion : this.form.get('direccion')?.value,
    dni: this.form.get('dni')?.value,
    telefono : this.form.get('telefono')?.value,
    estado_id : parseInt(this.form.get('estado_id')?.value),
  }
```

Elaborado por: Investigador

Iteración 5

Tabla 91: Resumen iteración 5

Iteración	ID	Nombre	Prioridad	Tiempo
5	HU015	Gestión de las Historias Clínicas	Alta	8
Tiempo estimado en días				8

Elaborado por: Investigador

Módulo Historias Clínicas

En la parte más compleja del sistema que son las Historias Clínicas, para la interfaz se tomó como punto de partida el formato de FCU vigente del Ministerio de Salud Pública bajo el código de Formulario 033, para hacerla lo más familiar posible a los especialistas para brindar un sistema que no sea invasivo y provoque el rechazo del usuario. Para lo cual esta interfaz se dividió en acordeones desplegable para evitar saturar la pantalla del usuario final.

Figura 27: Interfaz Historia Clínica 1/6

The screenshot displays a web application interface for a medical history. On the left is a sidebar with a blue header containing a menu icon and the text 'Dr. Juan Carlos', and a user icon. The sidebar lists five options: 'Citas', 'Pacientes', 'Gestion', 'Historias Clínicas' (which is highlighted), and 'Odontograma'. The main content area has a light blue header with the title 'MOTIVO DE CONSULTA' and an upward arrow. Below this is a text input field labeled 'Motivo'. The next section is titled 'ENFERMEDAD O PROBLEMA ACTUAL' with a subtitle 'Registrar síntomas, cronología, localización, características, intensidad, causa aparente' and an upward arrow. It contains a text input field labeled 'Enfermedad'. The third section is titled 'ANTECEDENTES FAMILIARES Y PERSONALES' with an upward arrow. At the bottom, there is a row of checkboxes for various conditions: 'ALERGIA ANTIBIOTICO', 'ALERGIA ANESTESIA', 'HEMORRAGIAS', 'VIHSIDA', 'TUBERCULOSIS', 'ASMA', and 'DIABETES'.

Elaborado por: Investigador

Al seleccionar uno de los checkbox se genera un nuevo input con el place holder del nombre del antecedente. Al desmarcar el checkbox se elimina automáticamente el input relacionado con ese antecedente.

Figura 28: Interfaz Historia Clínica 2/6

The screenshot displays a medical history interface for 'Dr. Juan Carlos'. It features a sidebar with navigation options: Citas, Pacientes, Gestion, Historias Clínicas, and Odontograma. The main content area is divided into three sections:

- ENFERMEDAD O PROBLEMA ACTUAL**: A section for recording current symptoms, with a placeholder 'Enfermedad' and a text area.
- ANTECEDENTES FAMILIARES Y PERSONALES**: A section for recording family and personal history. It includes a row of checkboxes for various conditions: ALERGIA ANTIBIOTICO (checked), ALERGIA ANESTESIA (checked), HEMORRAGIAS, VIH/SIDA, TUBERCULOSIS, ASMA, DIABETES, HIPERTENSION, ENF. CARDIACA, and OTROS. Below this, there are two text input fields labeled '1. Alergia Antibiotico' and '2. Alergia Anestesia'.
- SIGNOS VITALES**: A section for recording vital signs, with four input fields: Presión Arterial*, Frecuencia Cardiaca*, Temperatura*, and F. Respiratoria*.

Elaborado por: Investigador

Al seleccionar uno de los checkbox se genera un nuevo input con el place holder del nombre del órgano que forma parte del sistema estomatognático. Al desmarcar el checkbox se elimina automáticamente el input relacionado con ese organo. Logrando un control de los campos para evitar la saturación que puedan provocar al usuario final.

Figura 29: Interfaz Historia Clínica 3/6

Elaborado por: Investigador

Figura 30: Interfaz Historia Clínica 4/6

Elaborado por: Investigador

Figura 31: Interfaz Historia Clínica 5/6

Dr. Juan Carlos

- Citas
- Pacientes
- Gestion
- Historias Clinicas
- Odontograma

☒ BIOMETRIA
 ☒ QUÍMICA SANGÜÍNEA
 ☒ RAYOS-X
 ☒ OTROS

Biometria

Quimica Sanguinea

Rayos-X

Otros

DIAGNOSTICO

Agregar

CIE

☐ PRESUNTIVO
 ☐ DEFINITIVO

Elaborado por: Investigador

La pantalla principal que se muestra en la opción de Historias Clínicas es la siguiente donde se puede ver un panel para observar una Historia Clínica, o generar una nueva historia clínica desde cero que son las imágenes mostradas en la parte superior del sistema.

Figura 32: Interfaz Historia Clínica 6/6

Dr. Juan Carlos

Nuevo

Buscar

N. FCU	Usuario	Paciente	Sexo	Edad	Fecha Apertura	Estado
1	1706428971	Cevallos Andrade Carlos Alejandro	M	25	27/05/2024	✓
1	1802422971	Palacios Yancha Jesus Alejandro	M	45	27/05/2024	✗
1	1406428125	Cevallos Fiallos Daniel Marcelo	M	21	27/05/2024	✓
1	1802982564	Naranjo Inga Marcela Paola	F	8	27/05/2024	✓

Registros por página 5 1 - 4 de 4

Elaborado por: Investigador

Para la validación de los campos del formulario se utilizó expresiones lambda en los validators, en el proyecto se eliminó la opción de utilizar Template Forms y se procedió con el desarrollo de Reactive Forms por cuestiones de manejo unidireccional de datos. Por la cantidad de campos existentes se generó un código extenso donde se controló que los datos sean requeridos si así lo ameritan. Y en el caso de campos que autogeneren campos se creó un array dentro del control de ese campo.

Figura 33: Creación del objeto con los datos del formulario 1/2

```
constructor(private fb: FormBuilder) {
  this.createForm();
}

createForm() {
  this.form = this.fb.group({
    presion_arterial : [this.hc$ === undefined ? '' : this.hc$.presion_arterial, [Validators.required, Validators.pattern('^[A-Z]*)']],
    frecuencia_cardiaca : [this.hc$ === undefined ? '' : this.hc$.frecuencia_cardiaca, [Validators.required, Validators.pattern('^[A-Z]*)']],
    temperatura : [this.hc$ === undefined ? '' : this.hc$.temperatura, [Validators.required, Validators.pattern('^[a-zA-Z0-9]{5,16}')]],
    frecuencia_respiratoria : [this.hc$ === undefined ? '' : this.hc$.frecuencia_respiratoria, [Validators.required, Validators.pattern('^[a-zA-Z0-9]{5,16}')]],
    motivo_consulta : [this.hc$ === undefined ? '' : this.hc$.motivo_consulta, [Validators.required, Validators.pattern('^[A-Z]*)']],
    antecedentes : [this.hc$ === undefined ? '' : this.hc$.antecedentes, [Validators.required, Validators.pattern('^[A-Z]*)']],
    diagnostico : [this.hc$ === undefined ? '' : this.hc$.diagnostico, [Validators.required]],
    planes_diagnostico : this.fb.array([
      [this.hc$ === undefined ? '' : this.hc$.biometria, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.otros, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.rayosx, [Validators.required]]
    ]),
    sistema_estomatognatico : this.fb.array([
      [this.hc$ === undefined ? '' : this.hc$.labios, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.lengua, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.rayosx, [Validators.required]],
    ]),
    enfermedad : [this.hc$ === undefined ? '' : this.hc$.enfermedad, [Validators.required, Validators.pattern('^[A-Z]*)']],
    odontograma : this.fb.array([
      [this.hc$ === undefined ? '' : this.hc$.diente11, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.diente12, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.rayosX13, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.diente14, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.diente15, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.rayosX16, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.diente21, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.diente22, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.rayosX23, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.diente24, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.diente25, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.rayosX26, [Validators.required]],
      [this.hc$ === undefined ? '' : this.hc$.diente31, [Validators.required]],
    ])
  });
}
```

Elaborado por: Investigador

Figura 34: Creación del objeto con los datos del formulario 2/2

```
[this.hc$ === undefined ? '' : this.hc$.diente14, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.diente15, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.rayosX16, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.diente21, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.diente22, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.rayosX23, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.diente24, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.diente25, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.rayosX26, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.diente31, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.diente32, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.rayosX33, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.diente34, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.diente35, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.rayosX36, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.diente41, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.diente42, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.rayosX43, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.diente44, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.diente45, [Validators.required]],  
[this.hc$ === undefined ? '' : this.hc$.rayosX46, [Validators.required]]],  
tratamiento : this.fb.array([  
  [this.hc$ === undefined ? '' : this.hc$.sesion, [Validators.required]],  
  [this.hc$ === undefined ? '' : this.hc$.procedimiento, [Validators.required]],  
  [this.hc$ === undefined ? '' : this.hc$.diagnostico, [Validators.required]],  
  [this.hc$ === undefined ? '' : this.hc$.complicaciones, [Validators.required]],  
  [this.hc$ === undefined ? '' : this.hc$.prescripciones, [Validators.required]]],  
)]
```

Elaborado por: Investigador

Se utilizaron los servicios para interactuar con las operaciones CRUD implementadas en la API, la cual está alojada en un servidor y recibe datos en formato JSON. Además, se diseñó un formulario que permite manejar la información de manera unidireccional, incorporando métodos para la inserción, creación y actualización de datos.

Figura 35: Consumo de servicios para manipulación de Historias Clínicas

```

createHistoriaClinica(){
  SweetAlert.loadingAlert();
  this.historiaClinicaService.create(this.loadData()).subscribe(
    (res:any) => {
      SweetAlert.successAlert("Mensaje" + res.message);
    },
    (err) => {
      SweetAlert.errorAlert("Error:" + err);
    }
  );
}

updateHistoriaClinica(){
  SweetAlert.loadingAlert();
  this.historiaClinicaService.update(this.loadData()).subscribe(
    (res:any) => {
      SweetAlert.successAlert("Mensaje" + res.message);
    },
    (err) => {
      SweetAlert.errorAlert("Error:" + err);
    }
  );
}

deleteHistoriaClinica(){
  SweetAlert.loadingAlert();
  this.historiaClinicaService.update(this.loadData()).subscribe(
    (resp:any) => {
      SweetAlert.successAlert(resp.message);
    },
    (err) => {
      SweetAlert.errorAlert("Error"+err);
    }
  );
}

loadData(){

```

Elaborado por: Investigador

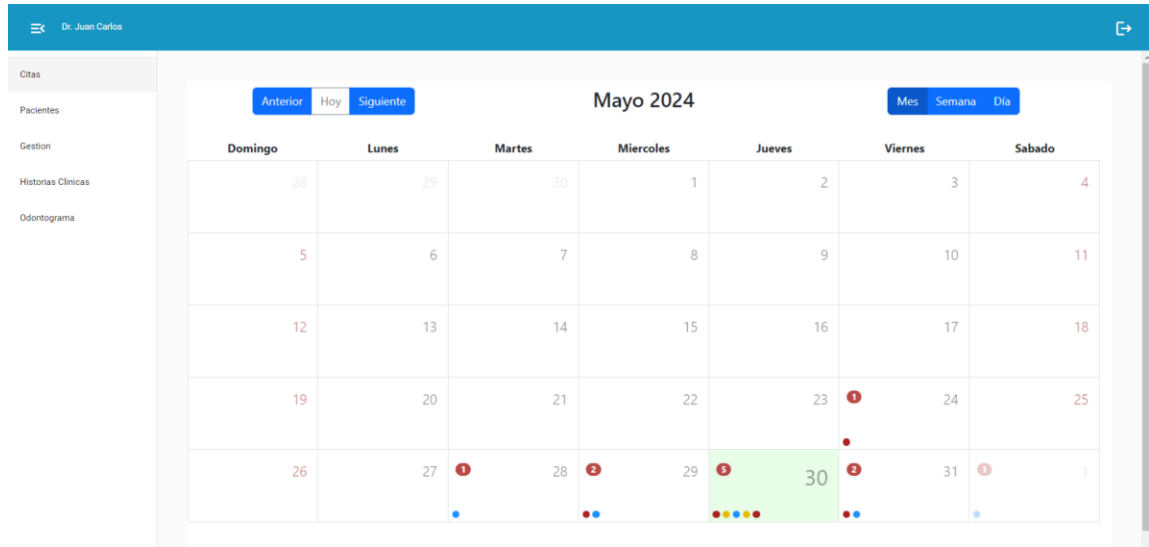
Iteración 6

Iteración	ID	Nombre	Prioridad	Tiempo
6	HU011	Gestión citas	Alta	5
Tiempo estimado en días				5

Elaborado por: Investigador

En la última iteración se centró en la creación de un calendario para asignar los eventos o citas que se van a generar a través del sistema por lo que se utilizó la librería angular-calendar en su versión 15. Donde seleccionara la hora, fecha y especialista solicitado por el paciente. El resultado se muestra a continuación.

Figura 36: Interfaz Citas



Elaborado por: Investigador

Al formar parte de una componente de código abierto se utilizó la documentación adecuada para la generación del código HTML para que el componente se vea de la manera mostrada se muestra en el siguiente fragmento de código.

Figura 37: Fragmento código HTML

```
src > app > pages > citas > citas.component.html > div.row.text-center > div.col-md-4 > div.btn-group > div.btn.btn-primary
1 <div class="row text-center">
2   <div class="col-md-4">
3     <div class="btn-group">
4       <div
5         class="btn btn-primary"
6         mwlCalendarPreviousView
7         [view]="view"
8         [(viewDate)]="viewDate"
9         (viewDateChange)="closeOpenMonthViewDay()"
10      >
11        Previous
12      </div>
13      <div
14        class="btn btn-outline-secondary"
15        mwlCalendarToday
16        [(viewDate)]="viewDate"
17      >
18        Today
19      </div>
20      <div
21        class="btn btn-primary"
22        mwlCalendarNextView
23        [view]="view"
24        [(viewDate)]="viewDate"
25        (viewDateChange)="closeOpenMonthViewDay()"
26      >
27        Next
28      </div>
29    </div>
30  </div>
31  <div class="col-md-4">
32    <h3>{{ viewDate | calendarDate:(view + 'ViewTitle'):'en' }}</h3>
33  </div>
```

Elaborado por: Investigador

Se desarrollo los métodos código para permitir la creación de eventos para cada uno de los días seleccionando la hora indicada. Además de los eventos para la funcionalidad de transición entre las distintas fechas ya sea por mes o año.

Figura 38: Fragmento creación de eventos del calendario

```
activeDayIsOpen: boolean = true;

constructor(private modal: NghModal) {}

dayClicked({ date, events }: { date: Date; events: CalendarEvent[] }): void {
  if (isSameMonth(date, this.viewDate)) {
    if (
      (isSameDay(this.viewDate, date) && this.activeDayIsOpen === true) ||
      events.length === 0
    ) {
      this.activeDayIsOpen = false;
    } else {
      this.activeDayIsOpen = true;
    }
    this.viewDate = date;
  }
}

eventTimesChanged({
  event,
  newStart,
  newEnd,
}: CalendarEventTimesChangedEvent): void {
  this.events = this.events.map((iEvent) => {
    if (iEvent === event) {
      return {
        ...event,
        start: newStart,
        end: newEnd,
      };
    }
    return iEvent;
  });
  this.handleEvent('Dropped or resized', event);
}
```

Elaborado por: Investigador

Finalización

Pruebas de aceptación

Durante la etapa de pruebas, se lleva a cabo una evaluación exhaustiva para garantizar el correcto funcionamiento de la aplicación final y la obtención de los resultados deseados. Esta fase se desarrolla en colaboración con los usuarios finales, con el fin de obtener su aprobación, y se fundamenta en las historias de usuario planteadas. Cada prueba se realiza siguiendo un formato específico. Para las siguientes pruebas se las realizo con el gerente de la empresa y el usuario administrador.

Tabla 92: Formato prueba de aceptación

Prueba de Aceptación			
Numero:		Identificador de Historia	
Nombre:			
Descripción:			
Condición de ejecución:			
Pasos:			
Resultado esperado:			
Resultado:			

Elaborado por: Investigador

Donde:

- **Número:** Numeración de cada prueba realizada
- **Identificador de Historia:** Números de Historia o Historias que se cumplen con la prueba
- **Nombre:** Título de la prueba.
- **Descripción:** Explica la actividad de manera breve.
- **Condiciones de ejecución:** Especifica las condiciones para que la prueba sea considerada como satisfactoria.
- **Pasos:** Conjunto de métodos para realizar la actividad.
- **Resultado Esperado:** Resultados obtenidos en la prueba.
- **Resultado:** Resultado final puede ser calificado como satisfactorio y no satisfactorio

Con esta prueba de aceptación se logró cumplir con el requisito funcional planteado dentro del análisis con el ID RF4. Para la primera prueba de aceptación se realizó la validación correcta del ingreso de un usuario a la plataforma. Logrando realizar una autenticación y generación del token de manera adecuada con el límite de tiempo que se pre estableció en 24 horas debido a las largas jornadas de trabajo en el establecimiento de salud. Mostrando mensajes de error en caso de un resultado incorrecto o un usuario inexistente.

Tabla 93: Prueba de aceptación 1

Prueba de Aceptación			
Numero:	01	Identificador de Historia	HU004
Nombre:	Pantalla iniciar sesión		
Descripción:	Validación de las credenciales de usuario		
Condición de ejecución:	El usuario debe ingresar credenciales las cuales son usuario y contraseña		
Pasos:	Ingresar en los campos de texto un usuario y una contraseña, y dar clic en el botón de ingresar		
Resultado esperado:	Se valida si las credenciales existen en la base de datos y son correctas se da paso a la pantalla de dashboard. Caso contrario muestra un mensaje de error de credenciales incorrectas.		
Resultado:	Satisfactorio		

Elaborado por: Investigador

Con esta prueba de aceptación se logró cumplir con el requisito funcional planteado dentro del análisis con el ID RF5. Donde se realizó un procedimiento correcto de la creación, visualización y edición de usuarios. Generando mensajes de error que no revelen información del sistema y sean de fácil comprensión para el usuario final al recibir

respuestas del servidor de errores inesperados que se generan como por ejemplo una falla de conexión provocada por la caída del servidor o la base de datos.

Tabla 94: Prueba de aceptación 2

Prueba de Aceptación			
Numero:	02	Identificador de Historia	HU005
Nombre:	Pantalla usuarios		
Descripción:	Gestión de la tabla de usuarios		
Condición de ejecución:	Se presenta una interfaz que muestra una lista de usuarios existentes junto con sus datos. Si se desea añadir uno nuevo, los campos requeridos se encuentran en la parte superior de la pantalla; en caso de querer editar, se cargan todos los datos a los campos para ser modificados. Para eliminar un usuario, el sistema solicita confirmación antes de ejecutar la acción el borrado será lógico y registro solo pasará a inactivo.		
Pasos:	Presionar sobre el botón de cada opción para verificar la consulta, creación, actualización y cambio de estado del registro.		
Resultado esperado:	La interfaz muestra correctamente el listado, y los campos para agregar, editar. En todas las acciones mostrará un mensaje para verificar si se realizó o no la actividad		
Resultado:	Satisfactorio		

Elaborado por: Investigador

Con esta prueba de aceptación se logró cumplir con el requisito funcional planteado dentro del análisis con el ID RF6. Donde se realizó un procedimiento correcto de la creación, visualización y edición de roles. Generando mensajes de error que no revelen información del sistema y sean de fácil comprensión para el usuario final al recibir respuestas del servidor de errores inesperados que se generan como por ejemplo una falla de conexión provocada por la caída del servidor o la base de datos. Además de lograr asignar los privilegios correspondientes a cada usuario o de la misma manera revocarlos. Esto es controlado en el sistema a través de Guards o guardianes que es una característica propia de angular para protección de rutas.

Tabla 95: Prueba de aceptación 3

Prueba de Aceptación			
Numero:	03	Identificador de Historia	HU006
Nombre:	Pantalla roles		
Descripción:	Gestión de la tabla de roles		
Condición de ejecución:	Se presenta una interfaz que muestra una lista de roles existentes junto con sus datos. Si se desea añadir un nuevo, los campos requeridos se encuentran en la parte superior de la pantalla donde se podrá asignar los módulos de acceso del rol; en caso de querer editar, se cargar todos los datos a los campos para ser modificados. Para eliminar un rol, el sistema solicita confirmación antes de ejecutar la acción el borrado será lógico y registro solo pasará a inactivo.		
Pasos:	Presionar sobre el botón de cada opción para verificar la consulta, creación, actualización y cambio de estado del registro.		
Resultado esperado:	La interfaz muestra correctamente el listado, y los campos para agregar, editar. En todas las acciones mostrará un mensaje para verificar si se realizó o no la actividad		

Resultado:	Satisfactorio
------------	---------------

Elaborado por: Investigador

Con esta prueba de aceptación se logró cumplir con parte del requisito funcional planteado ID RF3 y completamente con el requisito funcional RF7. Donde se realizó un procedimiento correcto de la creación, visualización y edición de odontólogos en donde las especialidades están predefinidas en la base de datos y evitan la manipulación externa de los médicos o usuarios ajenos al rol administrador. Generando mensajes de error que no revelen información del sistema y sean de fácil comprensión para el usuario final al recibir respuestas del servidor de errores inesperados que se generan como por ejemplo una falla de conexión provocada por la caída del servidor o la base de datos.

Tabla 96: Prueba de aceptación 4

Prueba de Aceptación			
Numero:	04	Identificador de Historia	HU007
Nombre:	Pantalla odontólogos		
Descripción:	Gestión de la tabla de odontólogos		
Condición de ejecución:	Se presenta una interfaz que muestra una lista de odontólogos existentes junto con sus datos. Si se desea añadir uno nuevo, los campos requeridos se encuentran en la parte superior de la pantalla; en caso de querer editar, se cargan todos los datos a los campos para ser modificados. Para eliminar un odontólogo, el sistema solicita confirmación antes de ejecutar la acción el borrado será lógico y registro solo pasará a inactivo.		
Pasos:	Presionar sobre el botón de cada opción para verificar la consulta, creación, actualización y cambio de estado del registro.		

Resultado esperado:	La interfaz muestra correctamente el listado, y los campos para agregar, editar. En todas las acciones mostrará un mensaje para verificar si se realizó o no la actividad
Resultado:	Satisfactorio

Elaborado por: Investigador

Con esta prueba de aceptación se logró cumplir con el requisito funcional con el ID RF2. Donde se realizó un procedimiento correcto de la creación, visualización y edición de pacientes. Generando mensajes de error que no revelen información del sistema y sean de fácil comprensión para el usuario final al recibir respuestas del servidor de errores inesperados que se generan como por ejemplo una falla de conexión provocada por la caída del servidor o la base de datos. Además de visualizar los pacientes existentes que puedan estar en estado activo o inactivo para un mejor control.

Tabla 97: Prueba de aceptación 5

Prueba de Aceptación			
Numero:	05	Identificador de Historia	HU009
Nombre:	Pantalla pacientes		
Descripción:	Gestión de la tabla de pacientes		
Condición de ejecución:	Se presenta una interfaz que muestra una lista de pacientes existentes junto con sus datos. Si se desea añadir uno nuevo, los campos requeridos se encuentran en la parte superior de la pantalla; en caso de querer editar, se cargar todos los datos a los campos para ser modificados. Para eliminar un paciente, el sistema solicita confirmación antes de ejecutar la acción el borrado será lógico y registro solo pasará a inactivo.		

Pasos:	Presionar sobre el botón de cada opción para verificar la consulta, creación, actualización y cambio de estado del registro.
Resultado esperado:	La interfaz muestra correctamente el listado, y los campos para agregar, editar. En todas las acciones mostrará un mensaje para verificar si se realizó o no la actividad
Resultado:	Satisfactorio

Elaborado por: Investigador

Esta prueba de aceptación aseguró el cumplimiento del requisito funcional identificado como RF1. Durante esta prueba, se llevó a cabo correctamente el proceso de creación y visualización de Historias Clínicas. Se implementaron mensajes de error que protegen la información del sistema y son comprensibles para los usuarios finales en caso de recibir respuestas de errores inesperados del servidor, como problemas de conexión debido a la caída del servidor o la base de datos. Además, se facilitó la visualización de todas las Historias Clínicas brindando un campo de filtrado para realizar una búsqueda precisa de registros de pacientes requeridos.

Tabla 98: Prueba de aceptación 6

Prueba de Aceptación			
Numero:	06	Identificador de Historia	HU010
Nombre:	Pantalla Historias Clínicas		
Descripción:	Gestión de la tabla de Historias Clínicas		
Condición de ejecución:	Se presenta una interfaz que muestra una lista de Historias Clínicas junto con sus datos. Si se desea añadir una nueva HC se redirige hacia otra ruta donde se encuentra todos los campos necesarios para el ingreso de una nueva Historia Clínica. Para eliminar una Historia Clínica, el sistema solicita confirmación antes de ejecutar la acción el		

	borrado será lógico y registro solo pasará a inactivo. Para la edición de la Historia Clínica los datos son cargados en los campos al dar clic en editar, sin embargo, la edición está permitida durante el mismo día de creación caso contrario deberá darse por inválida la historia clínica y generar otra nuevamente, ya que la actualización de Historias Clínicas hace que pierda legitimidad.
Pasos:	Presionar sobre el botón de cada opción para verificar la consulta, creación, actualización y cambio de estado del registro.
Resultado esperado:	La interfaz muestra correctamente el listado, y los campos para agregar, editar. En todas las acciones mostrará un mensaje para verificar si se realizó o no la actividad
Resultado:	Satisfactorio

Elaborado por: Investigador

Esta prueba de aceptación aseguró el cumplimiento del requisito funcional identificado como RF3. Durante esta prueba, se llevó a cabo correctamente el proceso de creación y visualización de Citas con la respectiva asignación de hora y especialista asignado a dicha cita. Se implementaron mensajes de error que protegen la información del sistema y son comprensibles para los usuarios finales en caso de recibir respuestas de errores inesperados del servidor, como problemas de conexión debido a la caída del servidor o la base de datos. Además, se facilitó la visualización de la Citas en el calendario marcadas por puntos de colores que al dar clic se despliegan mostrando en detalle la información.

Tabla 99: Prueba de aceptación 7

Prueba de Aceptación			
Numero:	07	Identificador de Historia	HU011
Nombre:	Pantalla Citas		
Descripción:	Gestión de la tabla de Citas		
Condición de ejecución:	Se presenta una interfaz que muestra un calendario de Citas junto con los eventos asignados a cada una de las fechas. Si se da clic en una fecha y se abren los eventos asignados a esa fecha. Se redirige hacia otra ruta donde se encuentra todos los campos necesarios para el ingreso de una nueva cita. Para eliminar una cita, el sistema solicita confirmación antes de ejecutar la acción el borrado será lógico y registro solo pasará a inactivo.		
Pasos:	Presionar sobre el botón de cada opción para verificar la consulta, creación, actualización y cambio de estado del registro.		
Resultado esperado:	La interfaz muestra correctamente el listado, y los campos para agregar, editar. En todas las acciones mostrará un mensaje para verificar si se realizó o no la actividad		
Resultado:	Satisfactorio		

Elaborado por: Investigador

A continuación, se muestra un resumen con los Requisitos funcionales y las pruebas de aceptación que ayudan a dar el requisito como cubierto de manera satisfactoria.

Tabla 100: Resumen de requisitos funcionales con sus pruebas de aceptación

ID Requisito	Descripción	ID Prueba Aceptación	Descripción
RF1	Gestionar Historias Clínicas	06	Pantalla Historias Clínicas
RF2	Gestionar paciente	05	Pantalla pacientes
RF3	Gestionar cita odontológica	04	Pantalla Odontólogos
		07	Pantalla Citas
RF4	Iniciar sesión	01	Pantalla iniciar sesión
RF5	Gestionar usuarios	02	Pantalla usuarios
RF6	Gestionar roles	03	Pantalla roles
RF7	Gestionar especialidades	04	Pantalla Odontólogos

Elaborado por: Investigador

CAPITULO IV

4.1 Conclusiones

Al concluir el presente proyecto se implementó un SPA que permite el control del manejo de historias clínicas y citas odontológicas dentro del Consultorio Dental “Odontólogo Juan Carlos León” con las siguientes conclusiones.

- Las encuestas realizadas a todos los empleados y la observación de campo que se realizó evidenciaron el proceso para el manejo de la información dentro de la clínica mostrando la manera en que se ejecutaba el trabajo de manera manual, documentando los actores que intervenían, así como cada una de las acciones que realizaban a través del Software Bonita Soft que nos ayudo a representar de manera gráfica y ordenada.
- Durante la investigación, se exploraron diversas tecnologías para la creación de aplicaciones de una sola página (SPA), tomando Angular como el marco de desarrollo preferido. Su estructura MVC facilitó un proceso de desarrollo ágil, destacándose por su capacidad para dividir claramente las responsabilidades de cada componente teniendo bondades como los Guards, Interceptors, HttpClient que ayudaron a tomar esta tecnología como la indica, para el proyecto
- Con ayuda de la aplicación desarrollada se simplifico el proceso de búsqueda de la información y agendamiento de citas evitando los errores que conlleva el manejo manual de la información. El tiempo de agendamiento de citas disminuyo de un promedio de 5 minutos a 4.30 minutos. Lo que evidencia una optimización de 30 segundos en el proceso. En la búsqueda de un registro médico el tiempo bajo de un promedio de 15 minutos entre las historias más antiguas y recientes a tomar 10 segundos con la implementación de la búsqueda por campo implementada en la tabla de Historias Clinicas.

4.2 Recomendaciones

- Se sugiere documentar los procesos futuros de manera correcta según estándares previstos por el Consultorio Dental Odontólogo Juan Carlos León aplicando normas ISO para asegurar la calidad del producto y documentación correcta de la información con estándares internacionales logrando así que estos procesos sean más fáciles de implementar en futuras actualizaciones del sistema.
- La persona que asumirá el rol de administrador de la aplicación será el encargado de sistemas que contratará el establecimiento de salud por lo que se sugiere una capacitación adecuada sobre el aplicativo web y manejo tanto del servidor como de la base de datos por lo que se recomienda una persona con conocimientos en las tecnologías Angular, NodeJS y MySQL.
- Es importante tener actualizadas las aplicaciones que ejecutarán la página web para un buen funcionamiento de esta. De igual manera se recomienda actualizar periódicamente el servidor para evitar fallos de seguridad o parches que comprometan la integridad de la aplicación web.
- Se recomienda al Consultorio Dental Odontólogo Juan Carlos León la creación de un cuarto de cómputo que cumpla con todas las normas para almacenamiento de un servidor tomando en cuenta que este puede llegar a tener escalamiento horizontal para redundancia de la información.
- Se recomienda tomar este software como un punto de partida para la creación de un sistema general que pueda ser implementado en distintas casas de salud de la ciudad. Brindando una solución oportuna a consultorios pequeños y medianos de la localidad.

BIBLIOGRAFIA

- [1] J. Guevara, «DESARROLLO E IMPLEMENTACIÓN DE SOFTWARE PARA LA GESTIÓN DE CONTROL DE HISTORIAS CLÍNICAS ODONTOLÓGICAS DEL CONSULTORIO DIVINO NIÑO DE LA CIUDAD DE PORTOVIEJO», Universidad Estatal del Sur de Manabí, Portoviejo, Ecuador, 2018.
- [2] L. Delgado, «DESARROLLO DE UN SOFTWARE PARA LA GESTIÓN DE CONTROL DE HISTORIAS CLINICAS ODONTOLÓGICAS EN UN CONSULTORIO DENTAL», Universidad Estatal del Sur de Manabi, Portoviejo, Ecuador, 2019.
- [3] S. Miranda, «ANÁLISIS Y DISEÑO DE APLICACIÓN MÓVIL PARA CITAS EN CONSULTORIOS ODONTOLÓGICOS PARTICULARES EN LA CIUDAD DE PIURA», Universidad de Piura, Piura, Perú, 2015.
- [4] J. Corilla, «DESARROLLO DE UN SISTEMA WEB PARA MEJORAR LA GESTIÓN DE HISTORIAS CLÍNICAS EN EL CONSULTORIO DENTAL ODONTOSTETIC», Universidad Tecnológica de Los Andes, Abancay, Perú, 2021.
- [5] E. Toaza y H. Armijos, «APLICACIÓN WEB CON TECNOLOGÍAS OPEN SOURCE PARA MEJORAR LA GESTIÓN DEL CONSULTORIO DENTAL “ODONTOLOGICAORALFAMILY”», Universidad de Guayaquil, Guayaquil, Ecuador, 2021.
- [6] M. Alava y Y. Suarez, «DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO PARA EL CONTROL DE HISTORIAS CLÍNICAS APLICANDO TECNOLOGÍA QR EN EL PATRONATO DEL CANTÓN LA MANÁ», Universidad Técnica de Cotopaxi, La Maná, Ecuador, 2022.

- [7] D. Almache, «SISTEMA WEB UTILIZANDO TECNOLOGÍA ANGULAR PARA APLICAR PROTOCOLOS DE BIOSEGURIDAD EN EL REGISTRO Y CONTROL DE GANADO DEL CENTRO DE FAENAMIENTO E.T.», Universidad Técnica de Ambato, Ambato, Ecuador, 2022.
- [8] «DESARROLLO WEB». Accedido: 25 de abril de 2024. [En línea]. Disponible en: https://www.ecured.cu/Desarrollo_Web
- [9] «¿Qué es un sistema web?» [En línea]. Disponible en: <https://www.addappto.com/que-es-un-sistema-web/>
- [10] «Tipos de sistemas web». Accedido: 6 de mayo de 2024. [En línea]. Disponible en: <https://www.atura.mx/blog/tipos-de-sistemas-web>
- [11] R. Lerma, J. Murcia, y E. Mifsud, *Aplicaciones Web*, Segunda. España: Mc Graw Hill, 2013.
- [12] K. Harsh, «¿Qué es la Arquitectura de las Aplicaciones Web? Desglosando una Aplicación Web». Accedido: 3 de mayo de 2024. [En línea]. Disponible en: <https://kinsta.com/es/blog/arquitectura-aplicaciones-web/#:~:text=La%20arquitectura%20de%20las%20aplicaciones%20web%20define%20c%C3%B3mo%20se%20estructuran,causar%20estragos%20en%20tu%20negocio.>
- [13] A. Robledano, «Qué es MongoDB». [En línea]. Disponible en: <https://openwebinars.net/blog/que-es-mongodb/>
- [14] C. Martinez, «APLICACIÓN WEB PROGRESIVA (PWA) PARA EL CONTROL DE INVENTARIOS EN LA EMPRESA FERRIMAR.», Universidad Técnica de Ambato, Ambato, Ecuador, 2023.

- [15] «What is MySQL?» [En línea]. Disponible en: <https://www.oracle.com/mysql/what-is-mysql/>
- [16] «Web services: servicios de máquina a máquina». Accedido: 3 de mayo de 2024. [En línea]. Disponible en: <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/web-services/>
- [17] A. Oliva, «APLICACIÓN DE SEGURIDAD EN SERVICIOS WEB XML PARADISPOSITIVOS MÓVILES MEDIANTE LA IMPLEMENTACIÓN DE UN PERFIL SAML», Universidad de Sevilla, Sevilla, España, 2006.
- [18] A. Cubo, «REPRESENTATIONAL STATE TRANSFER (REST). UN ESTILO DE ARQUITECTURA PARA SERVICIOS WEB. PANORÁMICA Y ESTADO DEL ARTE.», Universidad de Sevilla, Sevilla, España.
- [19] M. Montávez, «ARQUITECTURA DE PROGRAMACIÓN WEB: BACKEND», Universidad de Jaén, Jaén, España.
- [20] «Introducción a Express/Node». [En línea]. Disponible en: https://developer.mozilla.org/es/docs/Learn/Server-side/Express_Nodejs/Introduction
- [21] «Laravel: Qué es y para qué sirve». [En línea]. Disponible en: https://axarnet.es/blog/que-es-laravel?gad_source=1&gclid=EAIaIQobChMI7uWF4Nz0hQMVPKxaBR0QEAJB EAAYASAAEgJAN_D_BwE#
- [22] «Top 10 Backend Frameworks en 2024». [En línea]. Disponible en: <https://www.turing.com/resources/backend-frameworks>

- [23] «Introducción a Django». [En línea]. Disponible en: <https://developer.mozilla.org/es/docs/Learn/Server-side/Django/Introduction>
- [24] J. Goncalves, «QUE ES ANGULAR Y PARA QUE SIRVE». Accedido: 25 de abril de 2024. [En línea]. Disponible en: <https://www.hiberus.com/crecemos-contigo/que-es-angular-y-para-que-sirve/>
- [25] M. Rolfo, «React vs Angular vs Vue: ¿Qué framework elegir?» [En línea]. Disponible en: <https://codigoencasa.com/react-vs-angular-vs-vue-que-framework-elegir/>
- [26] M. Coppola, «¿Qué es React y para qué sirve?» [En línea]. Disponible en: <https://blog.hubspot.es/website/que-es-react>
- [27] «¿Qué es Vue.JS y para qué sirve?» [En línea]. Disponible en: <https://www.dongee.com/tutoriales/que-es-vue-js-para-que-sirve/>
- [28] J. Martins, «¿Qué es la metodología Kanban y cómo funciona?» [En línea]. Disponible en: <https://asana.com/es/resources/what-is-kanban>
- [29] «El modelo en cascada: desarrollo secuencial de software». [En línea]. Disponible en: <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/el-modelo-en-cascada/>
- [30] M. Castro, «Metodología RAD o DRA. El Desarrollo Rápido de Aplicaciones». [En línea]. Disponible en: <https://www.incentro.com/es-ES/blog/metodologia-rad-desarrollo-rapido-aplicaciones>

- [31] «ODONTOLOGIA». Accedido: 25 de abril de 2024. [En línea]. Disponible en: <https://centroclinicoespecialidadesdentales.com/odontologia-general/>
- [32] «HISTORIAS CLINICAS». Accedido: 25 de abril de 2024. [En línea]. Disponible en: <https://www.igaleno.com/blog/que-es-historia-clinica/#:~:text=La%20historia%20cl%C3%ADnica%20comprende%20el,que%20han%20intervenido%20en%20ellos.>
- [33] «EL REGLAMENTO PARA EL MANEJO DE INFORMACIÓN CONFIDENCIAL EN EL SISTEMA NACIONAL DE SALUD». 2016. [En línea]. Disponible en: <https://www.salud.gob.ec/wp-content/uploads/2016/09/AM-5216-A-Confidencialidad.pdf>
- [34] «Protocolos odontológicos». 2014. [En línea]. Disponible en: <https://www.salud.gob.ec/wp-content/uploads/2016/09/Protocolos-Odontol%C3%B3gicos.pdf>

ANEXO A: FORMULARIO 033 DEL MSP HISTORIA CLÍNICA DE ODONTOLOGÍA

ODONTOLOGIA (1)

10 PLANES DE DIAGNÓSTICO, TERAPÉUTICO Y EDUCACIONAL				
BIOMETRIA	QUIMICA SANGUINEA	RAYOS - X	OTROS	

11 DIAGNÓSTICO		PRE= PRESUNTIVO DEF= DEFINITIVO	CIE	PRE	DEF	CIE	PRE	DEF
1					3			
2					4			

FECHA DE APERTURA	FECHA DE CONTROL	PROFESIONAL	FIRMA	CÓDIGO	NÚMERO DE HOJA
-------------------	------------------	-------------	-------	--------	----------------

12 TRATAMIENTO

SESIÓN Y FECHA	DIAGNOSTICOS Y COMPLICACIONES	PROCEDIMIENTOS	PRESCRIPCIONES	CÓDIGO Y FIRMA
SESIÓN 1				CÓDIGO
FECHA				FIRMA
SESIÓN 2				CÓDIGO
FECHA				FIRMA
SESIÓN 3				CÓDIGO
FECHA				FIRMA
SESIÓN 4				CÓDIGO
FECHA				FIRMA
SESIÓN 5				CÓDIGO
FECHA				FIRMA
SESIÓN 6				CÓDIGO
FECHA				FIRMA
SESIÓN 7				CÓDIGO
FECHA				FIRMA
SESIÓN 8				CÓDIGO
FECHA				FIRMA
SESIÓN 9				CÓDIGO
FECHA				FIRMA

SNS-MSP / HCU-form.033 / 2008

ODONTOLOGÍA (2)

ANEXO B: BASE DE DATOS

