

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Sistema de Gestión de pacientes para estudiantes odontología-
OralData

Karen Melissa Calle Betancur

Trabajo de grado presentado como requisito parcial para optar al título de:
Especialista en ingeniería de software

Asesor(es)
Alicia Osorio Builes
Juan Pablo Vallejo

Instituto Tecnológico Metropolitano - ITM
Facultad de Ingenierías
Medellín, Colombia
2024

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

RESUMEN

El proyecto se centra en el desarrollo de un sistema de gestión de pacientes web para estudiantes de odontología, el cual busca automatizar tareas operativas y mejorar la experiencia en prácticas clínicas. Para cumplir con esto, se plantea comprender las necesidades técnicas para el diseño de la aplicación, seleccionar funcionalidades adecuadas, estructurar el proceso de prácticas clínicas de manera efectiva, y desarrollar un sistema que cumpla con estas necesidades, integrando las funcionalidades necesarias para una gestión eficiente de pacientes.

Para lograrlo, se empleará C# y Entity Framework para facilitar la interacción con bases de datos, y Blazor como marco para el frontend debido a su facilidad de uso y compatibilidad. Además, se implementará el patrón WPA para mejorar la seguridad y mantenibilidad de la aplicación, y se adoptará una arquitectura responsiva para garantizar accesibilidad en diferentes dispositivos. El desarrollo del proyecto se dividirá en dos etapas principales: una centrada en la definición de requisitos, diseño y planificación, y otra en la implementación, pruebas y despliegue del sistema. En resumen, el proyecto aspira a ofrecer una solución integral que mejore la gestión de pacientes en facultades de odontología, optimizando la organización y seguridad de la información, y elevando la eficiencia en el proceso de prácticas clínicas.

Palabras clave: Gestión de pacientes, automatización. Experiencia de usuario, C#, Entity Framework, Blazor, WPA, implementación.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

ACRÓNIMOS

- C#: C Sharp
- EF: Entity Framework
- WPA: Windows Presentation Foundation
- CRUD: Create, Read, Update, Delete (Crear, Leer, Actualizar, Eliminar)
- UI: User Interface (Interfaz de Usuario)
- SQL: Structured Query Language
- DB: Database (Base de Datos)
- UI/UX: User Interface/User Experience (Interfaz de Usuario/Experiencia de Usuario)
- QA: Quality Assurance (Aseguramiento de la Calidad)
- AWS: Amazon Web Services (si en algún momento consideras usarlo en lugar de Azure Blob Storage)
- DTO: Data Transfer Object (Objeto de transferencia de datos)

TABLA DE CONTENIDO

1.	INTRODUCCIÓN	7
	OBJETIVOS	8
	CRONOGRAMA DEL PROYECTO.....	10
2.	METODOLOGÍA.....	14
	2.1 Levantamiento de requisitos.....	14
	2.2 Priorización de requisitos.....	17
	2.3 Diseño y arquitectura.....	17
	2.4 Gestión del proyecto:.....	20
	2.3 Desarrollo de la aplicación	20
	2.4 Pruebas unitarias y de integración.....	21
	2.5 Pruebas de aceptación	21
	Encuestas de satisfacción.....	21
3.	RESULTADOS Y DISCUSIÓN.....	21
	3.1 Levantamiento de requisitos:.....	21
	3.2 Identificación de requisitos del sistema.....	24
	3.3 Desarrollo de la aplicación	30
	3.4 Pruebas unitarias y de integración.....	42
	3.5 Pruebas de aceptación	43
5.	CONCLUSIONES, RECOMENDACIONES Y TRABAJO FUTURO	53
	Conclusiones	53
	Recomendaciones	54
	Trabajo futuro	54
	REFERENCIAS	55
	ANEXOS.....	57

Índice de figuras

<i>Figura 1.</i> Evidencias proceso de elicitation.....	15
<i>Figura 2.</i> Esquema preconceptual del proyecto.	16
<i>Figura 3.</i> Backlog del producto	17
<i>Figura 4.</i> Arquitectura del proyecto.....	18
<i>Figura 5.</i> Diagrama causa-raíz	22
<i>Figura 6.</i> Diagrama de clases del proyecto	¡Error! Marcador no definido.
<i>Figura 7.</i> Historia de usuario funcional 1	24
<i>Figura 8.</i> Historia de usuario funcional 2	24
<i>Figura 9.</i> Historia de usuario funcional 3	25
<i>Figura 10.</i> Historia de usuario funcional 4	25
<i>Figura 11.</i> Historia de usuario funcional 5	26
<i>Figura 12.</i> Historia de usuario funcional 6	26
<i>Figura 13.</i> Historia de usuario funcional 7	27
<i>Figura 14.</i> Historia de usuario no funcional 1	27
<i>Figura 15.</i> Historia de usuario no funcional 2	28
<i>Figura 16.</i> Historia de usuario no funcional 3	28
<i>Figura 17.</i> Historia de usuario no funcional 4	29
<i>Figura 18.</i> Estructura OralData.Backend.....	30
<i>Figura 19.</i> OralData.Frontend	31
<i>Figura 20.</i> OralData.Shared.....	31
<i>Figura 21.</i> OralData.UnitTest	32
<i>Figura 22.</i> Endpoints de la API de Gestión de Cuentas en el Sistema OralData	32
<i>Figura 23.</i> Endpoints de la API de Gestión de Disponibilidades de estudiantes.....	33
<i>Figura 24.</i> Endpoints de la API de Gestión de Encuestas de Clasificación en el Sistema.....	33
<i>Figura 25.</i> Endpoints de la API de Gestión de Cursos	33
<i>Figura 26.</i> Endpoints de la API de Gestión de Citas	34
<i>Figura 27.</i> Endpoints de la API de Gestión de Dominios Permitidos para Estudiantes	34
<i>Figura 28.</i> Pagina de inicio- Menu administrador.....	35
<i>Figura 29.</i> Interfaz de usuario para registro de pacientes	35
<i>Figura 30.</i> Interfaz de usuario para registro de estudiantes.....	36
<i>Figura 31.</i> Interfaz de usuario para inicio de sesión	36
<i>Figura 32.</i> Interfaz de usuario para que pacientes diligencien encuesta de clasificación	37
<i>Figura 33.</i> Encuesta de clasificación diligenciada	37
<i>Figura 34.</i> Interfaz de usuario ver encuestas diligenciadas por pacientes.....	38
<i>Figura 35.</i> Profesor visualiza encuesta y clasifica	38

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Figura 36. Interfaz de usuario para estudiante ingresar horario de disponibilidad . **¡Error! Marcador no definido.**

Figura 37. Estudiante visualiza su disponibilidad..... **¡Error! Marcador no definido.**

Figura 38. Interfaz de usuario para estudiante ver pacientes disponibles 40

Figura 39. Estudiante agenda cita con paciente **¡Error! Marcador no definido.**

Figura 40. Interfaz para estudiante ver historial de citas agendadas 41

Figura 41. Interfaz para paciente ver historial de citas agendadas 41

Figura 42. Pruebas unitarias creadas 42

Figura 43. Cobertura de pruebas unitarias 43

Índice de tablas

Tabla 1. Cronograma del proyecto..... 10

Tabla 2. Uso patrón repositorio 18

Tabla 3. Uso patrón fachada 19

Tabla 4. Aplicación de principios SOLID 19

Tabla 5. Pasos pruebas de aceptación 49

Tabla 6. Preguntas encuesta de satisfacción..... 51

Tabla 7. Resultados encuesta de satisfacción 51

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

1. INTRODUCCIÓN

La formación de los odontólogos requiere de un equilibrio entre la teoría y la práctica. Sin embargo, en los últimos años, las clínicas prácticas en las facultades de odontología de Colombia han enfrentado un déficit importante. Esto se debe a que los estudiantes, en gran medida, tienen dificultades para encontrar pacientes que les permitan desarrollar sus competencias prácticas y habilidades manuales, adquiridas inicialmente en la teoría (Rojas & Galvis, 2023).

Este déficit puede afectar la calidad de la formación de los odontólogos y la atención que brindan a los pacientes. Los estudiantes que no tienen la oportunidad de practicar sus habilidades en pacientes reales pueden tener dificultades para adquirir la confianza y la experiencia necesarias para ejercer su profesión. Además, esto les genera un estancamiento académico ya que al no demostrar sus competencias estos reprueban y por último los pacientes que reciben atención odontológica de estudiantes con poca experiencia pueden estar expuestos a un mayor riesgo de complicaciones.

Todo esto, ha hecho que el proceso de gestión de pacientes en las facultades de odontología sea un proceso complejo y desafiante, generando dificultades para los estudiantes y profesores. Los profesionales de la salud deben dedicar mucho tiempo a tareas operativas, como la búsqueda de pacientes, el diligenciamiento de la historia clínica y la asignación de citas (F. Arevalo, 2015). La gestión de pacientes de manera tradicional, sin un sistema que permita automatizar estas tareas, es ineficiente y puede generar otras dificultades, como el aumento de los errores, la disminución de la productividad y la mala experiencia del paciente (Journal of Dental Education, 2021).

Actualmente no se cuenta con un sistema que sirva de puente directo entre los estudiantes y pacientes que permita la gestión, atención, fluidez y trazabilidad sin que exista un rol intermedio en las instituciones que en su mayoría de veces no cumple con estos requerimientos ya que usa una hoja de cálculo y no contempla una valoración acertada o información detallada que permita direccionar bien a los estudiantes para asignar sus clínicas, dejando un panorama de incertidumbre en los estudiantes para cumplir con los requisitos académicos necesarios para aprobar y continuar con su proceso de aprendizaje, estos actualmente tienen truncados estos procesos y les está generando retrasos por no aprobar los niveles o semestres a tiempo. Por otro lado, los pacientes no tienen la

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

posibilidad de identificar la existencia de jornadas o clínicas de atención en las cuales ellos pueden acceder a servicios de forma gratuita o a un bajo costo.

Estas dificultades evidencian la necesidad de un sistema el cual permita realizar gestión, trazabilidad y obtención de pacientes por parte de los estudiantes para que puedan ejercer sus clínicas de atención y que los pacientes con pocos recursos tengan la posibilidad de acceder a una atención respaldada por un docente con amplia experiencia de manera gratuita o más económica.

Por lo anterior que, para abordar este problema, es necesario implementar un sistema de gestión de pacientes que permita automatizar las tareas operativas ayudándole a los estudiantes a elegir, contactar y agendar pacientes a través de un sistema que permita mejorar la eficiencia, la seguridad y la experiencia entre paciente y el estudiante.

¿Es posible diseñar un sistema de gestión de pacientes para estudiantes de odontología que mejore la eficiencia y la experiencia en el proceso de las practicas clínicas?

OBJETIVOS

General

Proponer un sistema de gestión de pacientes orientado a la web para que los estudiantes de odontología que puedan automatizar las tareas operativas, mejorando la eficiencia y la experiencia en el proceso de prácticas clínica

Específicos

- Comprender las necesidades técnicas que se tendrían al momento de diseñar una aplicación para gestión de pacientes para las facultades de odontología
- Seleccionar de las necesidades técnicas anteriormente establecidas las funcionalidades que debe llevar el sistema de gestión de pacientes en su diseño
- Diseñar un sistema de gestión orientado a la web capaz de estructurar el proceso de prácticas clínicas
- Desarrollar un sistema de gestión de pacientes que cumpla con las necesidades técnicas anteriormente identificadas

Aplicaciones similares en el mercado

Después de una investigación de mercado se encuentran algunas aplicaciones con funcionalidades similares al software que se propone desarrollar. En primer lugar, se

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

encontró, la aplicación de origen escandinava llamada My Dental Clinic, el cual es una herramienta útil para estudiantes, en el cual pueden obtener y gestionar pacientes. Dicha aplicación, permite la gestión de la información de pacientes, teniendo acceso en todo momento a dichos datos, adicionalmente, permite rastrear los gastos de la clínica, crear, imprimir, enviar tratamientos a los pacientes, tener colaboración entre profesores, de manera que varios estudiantes y profesores puedan atender al paciente al mismo tiempo. Además, permite tener respaldo de los datos de pacientes. (Dental Clinic App, s.f.)

Dentro de dicha revisión, se encontró además, un software mexicano llamado Smile que está especializado en clínicas universitarias, el cual presenta similitudes a las del proyecto propuesto, entre éstas se encuentran el agendamiento y control de citas de los pacientes por cada estudiante, registro de expediente clínico con notas de evolución y plan de tratamiento, control de actividades de estudiantes por parte de profesores e incluye otras funcionalidades como lo son controles de órdenes de laboratorio, imágenes médicas al expediente clínico, control de inventarios e ingresos económicos de la clínica; mostrando altos beneficios tanto en el ámbito clínico-académico como a nivel administrativo. (Software Dentalink, s.f.) De esta misma manera, y compartiendo gran cantidad de funcionalidades que presentan las aplicaciones anteriores, se encontró un software chileno llamado Dentalink, que además tienen un módulo integrado de toda la parte de contabilidad, ingresos y trazabilidad de los pacientes, cuentan también con un dashboard que permite por medio de graficas tener informes precisos y en tiempo real de las citas asignadas, pacientes agendados, pacientes en consulta y pagos online. (Software Dentalink, s.f.)

El sistema propuesto se diferencia significativamente de las aplicaciones existentes por su enfoque innovador de autogestión del agendamiento por parte de los estudiantes, complementado con la clasificación previa de pacientes por parte de un profesor y la posibilidad de cancelación de citas tanto por estudiantes como por pacientes.

Esto garantiza efectividad y calidad en el proceso de atención debido a que:

- Un profesor calificado revisa los perfiles de los pacientes registrados y los clasifica según sus necesidades clínicas, asignándolos al curso correspondiente dentro de las prácticas clínicas.
- Los estudiantes tienen la autonomía de seleccionar a los pacientes que desean atender, basándose en sus competencias e intereses.
- Tanto estudiantes como pacientes tienen la flexibilidad de cancelar citas con anticipación

CRONOGRAMA DEL PROYECTO

Tabla 1. Cronograma del proyecto



Actividad	Semana	Agosto		Sept		Octubre		Noviembre		Febrero		Marzo		Abril		Mayo	
		S1-S2	S3-S4	S1-S2	S3-S4	S1-S2	S3-S4	S1-S2	S3-S4	S2	S3-S4	S1-S2	S3-S4	S1-S2	S3-S4	S1-S2	S3-S4
Elaboración de la propuesta inicial																	
Técnicas de elicitación --> Entrevistas a profesores de facultades de odontología para comprender sus necesidades y expectativas																	
Diseño de modelo de datos para el sistema de gestión de pacientes (entidades, las relaciones y atributos)																	
Análisis de los requisitos técnicos necesarios para el sistema de gestión de pacientes (requisitos funcionales y requisitos no funcionales)																	
Priorización de las necesidades técnicas identificadas (teniendo en cuenta importancia, complejidad)																	
Definición de requisitos funcionales (Funciones y los procesos que el sistema debe realizar)																	
Definición de requisitos no funcionales específicos del sistema																	
Desarrollo de diagramas del proceso, para comprender: pasos, interacciones y actores																	

Definición de los componentes del sistema de gestión de paciente (lenguaje, Framework, otras tecnologías a usar)																		
Desarrollo de un prototipo del sistema OralData																		
Desarrollo de todas las funcionalidades de la aplicación																		
Testing de requisitos funcionales y no funcionales del sistema																		
Implementación del sistema en una facultad de odontología																		

MARCO TEÓRICO

Aprendizaje práctico: Es un enfoque de aprendizaje centrado en la experiencia de aprendizaje activo y en la aplicación de conocimientos para situaciones reales. Dicho enfoque, se basa en la teoría de aprendizaje experiencial de John Dewey, quien afirma que el conocimiento se construye a través de la experiencia y son los estudiantes quienes son los protagonistas de su propio aprendizaje. (Ausubel, Novak y Hanesian, 1978)

Este aprendizaje, puede incluir actividades como resolución de problemas, exploración y experimentación, por tanto, los estudiantes construyen su propio conocimiento a medida que avanzan (Ausubel, Novak y Hanesian, 1978), incluyendo estrategias como el aprendizaje cooperativo y basado en el servicio, permitiendo abordar problemas en situaciones reales con el objetivo de beneficiar a la comunidad.

Este tipo de aprendizaje es importante para la formación de los odontólogos, ya que les permite desarrollar las competencias necesarias para ejercer su profesión. En las áreas de la salud, dicho aprendizaje, se da a través de prácticas clínicas.

Prácticas clínicas odontológicas: Las prácticas clínicas en el área odontológica son esenciales para la formación de estudiantes, en ellas los estudiantes pueden aplicar los

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

conocimientos adquiridos durante las clases teóricas en un entorno real (Universidad de Santiago de Compostela, 2023). Durante dichas prácticas, se realiza evaluación y tratamiento a pacientes, las cuales son realizadas bajo la supervisión de profesores con experiencia tanto en docencia como en atención clínicas.

Las prácticas clínicas, incluyen adquisición de competencias de comunicación asistencial, gestión clínica, razonamiento y juicio crítico, por tanto, no solo aprenden a realizar procedimientos dentales, sino que también a comunicarse efectivamente con pacientes, gestionar citas, recursos y a tomar decisiones basados en evidencia (Universidad de Santiago de Compostela, 2023).

Para desarrollar dichas prácticas, es necesario, contar con un sistema que permita realizar tareas administrativas, las cuales permitan realizar la gestión de pacientes.

Sistemas de gestión de pacientes: Es una parte fundamental de la atención médica, en la cual se realizan procesos administrativos, en los cuales se coordina de punta a punta la atención al paciente. Durante la gestión, se recopila información del paciente, se realizan actividades administrativas, se gestionan las horas de atención, los profesionales que prestarán la atención con los profesionales de la red asistencial y se coordina los cuidados (F. Aguayo & R. Mella, 2015)

Dicho proceso, implica, planificación, organización, motivación en la atención y control de provisión de cuidados a prestarse de manera oportuna, segura e integral, sustentado en políticas de cada centro de atención. (F. Aguayo & R. Mella, 2015)

Los sistemas de gestión de pacientes han evolucionado significativamente en los últimos años, por tanto, su evolución ha estado marcada por los siguientes hitos:

- En 1960 comienzan a usarse en la década de 1960, estos sistemas eran simples, dicha información se diligenciaba de manera manual en papel y era almacenada en carpetas físicas
- En 1970 se comienza a usar tecnología informática, con sistemas que permitían a médicos y profesionales en salud, acceder a información de los pacientes de manera rápida y fácil.
- En 1980: se introducen los primeros sistemas de gestión de pacientes con sistemas integrados, los cuales combinaban información de los pacientes como historial médico, resultado de exámenes médicos y prescripciones, consultando dicha información desde diferentes fuentes.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

- En 1990: Se comienza a desarrollar sistemas para gestión de pacientes en la web, permitiendo a profesionales de la salud consultar información de los pacientes desde cualquier lugar
- En la década del 2000, se comienzan a realizar sistemas para gestión de pacientes para aplicaciones móviles
- En la década del 2010, se introducen los sistemas para gestión en la nube, los cuales almacenan información de pacientes desde cualquier lugar con acceso a internet
- En la década del 2020, se introduce el aprendizaje automático y la inteligencia artificial a los sistemas de gestión de pacientes, permitiendo tomar decisiones sobre el tratamiento de los pacientes

Al ser la odontología una rama de la salud, la evolución de los sistemas de gestión de pacientes se ha visto influenciada por los sistemas de gestión de pacientes, en los cuales se hacen ajustes de acuerdo con las necesidades específicas del sector, por tanto, ha sido un reflejo de la evolución de la tecnología en los sistemas de gestión en salud.

La gestión de pacientes de manera efectiva puede ayudar a recopilar información la cual permita comprender mejor las necesidades de los usuarios de manera que se pueda prestar una atención de calidad.

Gestión de la calidad: Es un proceso llevado a cabo con el fin de mejorar la eficiencia y efectividad de las operaciones y servicios de una organización. Se enfoca en la mejora continua de los procesos, productos y servicios para satisfacer o superar las expectativas y necesidades de los clientes.

Este proceso implica la recopilación y análisis de la información de los clientes para entender sus necesidades y expectativas, y luego utilizar esta información para desarrollar productos y servicios que satisfagan o superen estas expectativas (Definición ABC, gestión de la calidad)

Esta, además, implica la implementación de sistemas y procedimientos para controlar y mejorar la calidad de los productos y servicios. Esto puede incluir la inspección de los productos o servicios, la realización de pruebas y auditorías, y la implementación de acciones correctivas cuando se detectan problemas de calidad.

El desarrollo de un sistema que permita desarrollar un sistema de gestión en salud requiere un lenguaje de programación potente y flexible y un marco de desarrollo que permita amplia gama de bibliotecas y herramientas, como las presentadas a continuación:

Entity Framework (EF): es un marco que permite mapeo objeto-relación (ORM) de código abierto para ADO.NET e introducido por Microsoft en 2008 con .NET Framework 3.5. Este

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

marco, ha tenido varias versiones, donde las más recientes son EF 6 y EF Core (Entity Framework Tutorial, 2023).

EF permite a los desarrolladores trabajar con datos relacionales utilizando objetos específicos del dominio en lugar de centrarse en las tablas y columnas subyacentes de la base de datos donde se almacenan los datos. Esto significa que EF elimina la necesidad de escribir el código de acceso a datos que los desarrolladores suelen necesitar escribir (Entity Framework Tutorial, 2023).

Dicho framework, realiza operaciones CRUD utilizando EDM en la construcción de consultas SQL a partir de consultas LINQ, la construcción de comandos INSERT, UPDATE y DELETE, y transforma el resultado de la base de datos en objetos de entidad (Entity Framework Tutorial, 2023).

Blazor es un marco de desarrollo web de código abierto que permite a los desarrolladores crear aplicaciones web utilizando C# y HTML. Fue desarrollado por Microsoft y fue lanzado en 2018 para unificar el desarrollo de aplicaciones web, permitiendo a los desarrolladores utilizar C# tanto en el backend como en el frontend. (<https://www.onestopdevshop.io/net-core-blazor/>)

Blazor es conocido por su capacidad para ejecutar vistas Razor en el cliente en lugar de en el servidor, lo que permite una mayor interactividad y rendimiento. Además, Blazor es compatible con los estándares web abiertos, lo que significa que las aplicaciones Blazor pueden ejecutarse en cualquier navegador que soporte estos estándares (Blazor University, s.f.)

Azure Blob Storage: Es un sistema de almacenamiento para objetos en la nube para Microsoft, siendo una parte integral del modelamiento en Azure, diseñado para almacenar grandes cantidades de datos no estructurados, que contentan texto o datos binarios.

Teniendo en cuenta el contexto, blob es un objeto que contiene grandes cantidades de datos, los cuales pueden ser de tres tipos: bloque, anexar y pagina. (Microsoft Learn, s.f.)

2. METODOLOGÍA

La metodología adoptada para el desarrollo del proyecto se dividió en varias fases, cada una con su propio enfoque y técnicas específicas para garantizar el éxito del proyecto. A continuación, se describe cada una de ellas:

2.1 Levantamiento de requisitos

Para el proceso de levantamiento de requisitos, la técnica de elicitación seleccionada fue la entrevista, por tanto, se realizaron 3 entrevistas iniciales a 2 profesores (uno de la

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Universidad de Antioquia y otras 2 con una profesora de la Universidad de Antioquia), con el objetivo de conocer el estado del proceso, donde se utilizaron cuestionarios para recopilar información detallada sobre las necesidades y expectativas de los usuarios.




	Entrevista elicitación UCC.mp4	September 7, 2023	KAREN MELISSA CALLE BE
	Entrevista elicitation UdeA.mp4	September 7, 2023	KAREN MELISSA CALLE BE
	entrevista elicitación.docx	September 4, 2023	KAREN MELISSA CALLE BE

Figura 1. Evidencias proceso de elicitation

Una vez concluidas las primeras reuniones y obtenida toda la información necesaria, se construyó el diagrama causa raíz el cual permitió identificar y comprender los problemas, donde se destacaron las principales causas de ineficiencias y problemas en la gestión de pacientes para los estudiantes de odontología.

Posteriormente, se realizaron varias versiones de esquema preconceptual hasta dejar una definitiva que mejor se adaptara a sus necesidades, donde se visualizaran y organizaran las ideas principales y los conceptos relevantes y a partir de este esquema, se establecieron los requisitos tanto funcionales como los requisitos no funcionales del sistema.

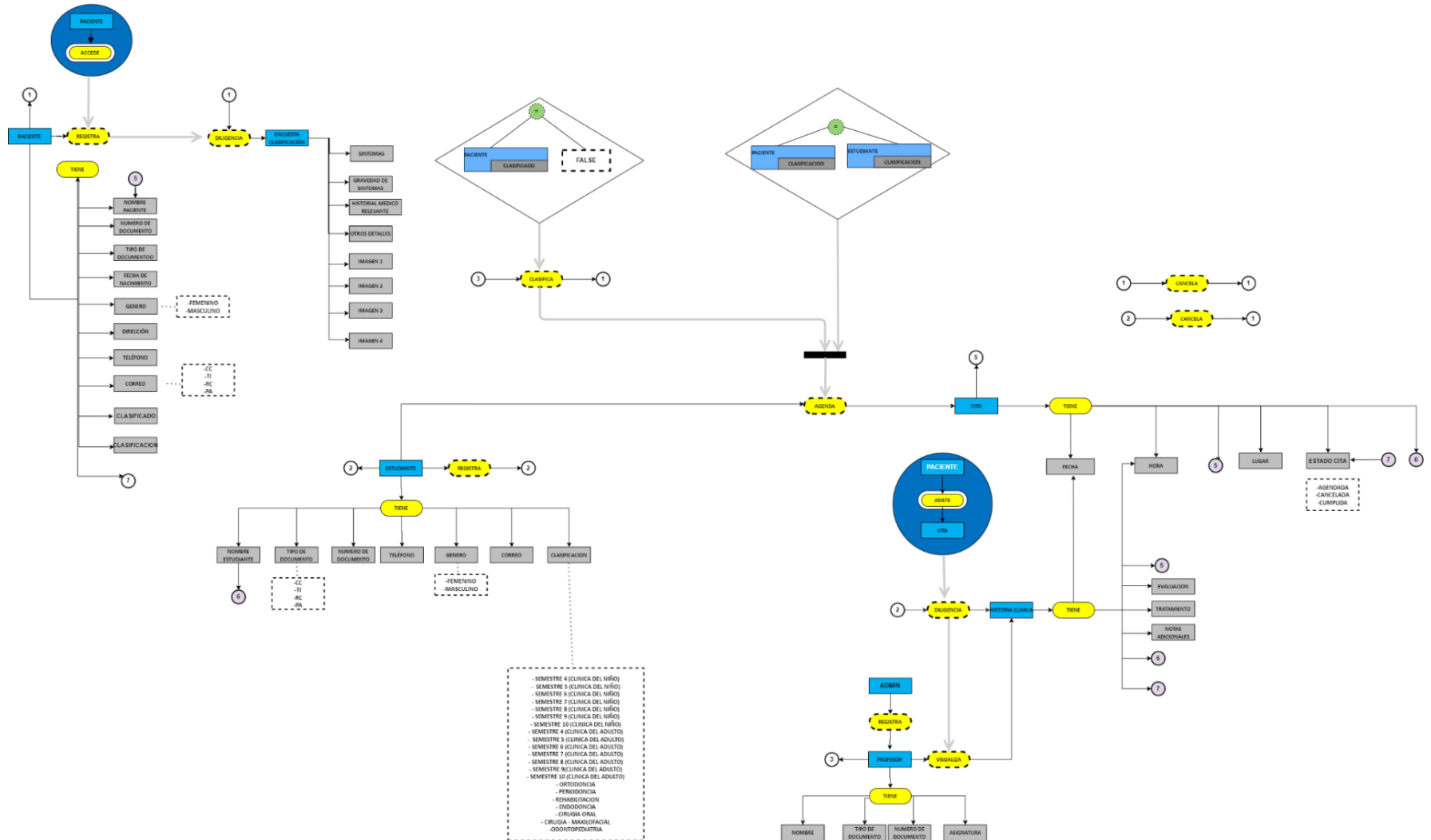


Figura 2. Esquema preconceptual del proyecto.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

2.2 Priorización de requisitos

Una vez capturada la información necesaria y realizados los esquemas principales, los requisitos fueron priorizados haciendo uso de la técnica MosCoW (Must have, Should have, Could have, won't have), en compañía de la stakeholder, donde se buscó asegurar que los requisitos más críticos del sistema fueran los primeros en abordarse (buscando alinearse con las principales necesidades).

Al mismo tiempo, se elaboraron las historias de usuario detalladas para los requisitos funcionales, donde fueron plasmadas en la herramienta Azure Devops y se tenía en cuenta la necesidad y los criterios de aceptación para cada historia. Adicional a las historias de usuario funcionales, se crearon las historias de usuario no funcionales, donde se tienen en cuenta los aspectos estructurales y arquitectónicos de la solución.

A continuación, se presenta el backlog de Oral Data, el cual está dividido en 3 épicas que contienen las 3 grandes funcionalidades principales de la aplicación, las cuales son registro en plataforma, agendamiento de citas y registro de la historia clínica:

Title	State	Tags	Assigned To
<ul style="list-style-type: none"> ▼ 👑 Registro en plataforma 🏠 HU_NF 001 - Garantizar contraseña segura 🏠 HU_RF 001- Registrar pacientes 🏠 HU_RF 002- Registrar estudiantes 🏠 Pruebas - Etapa 1 	<ul style="list-style-type: none"> ● Done ● Done ● Done ● Done ● Done 		<ul style="list-style-type: none"> KAREN MELISSA CALLE BETANCUR KAREN MELISSA CALLE BETANCUR KAREN MELISSA CALLE BETANCUR
<ul style="list-style-type: none"> ▼ 👑 Clasificación de pacientes 🏠 HU_RF 003 - Diligenciar encuesta de clasificación 🏠 HU_RF 004 - Clasificar pacientes 	<ul style="list-style-type: none"> ● Done ● Done ● Done 		<ul style="list-style-type: none"> KAREN MELISSA CALLE BETANCUR KAREN MELISSA CALLE BETANCUR
<ul style="list-style-type: none"> ▼ 👑 Agendamiento de citas 🏠 HU_NF 002 - Factibilizar Usabilidad 🏠 HU_NF 003 - Garantizar escalabilidad 🏠 HU_RNF 004 - Ratificar mantenibilidad 🏠 HU_RF 005 - Gestionar horarios de disponibilidad 🏠 HU_RF 006 - Visualización de pacientes disponibles 🏠 HU_RF 007 - Agendar citas 🏠 Pruebas - Etapa 2 	<ul style="list-style-type: none"> ● Doing ● Done ● Done ● Done ● Done ● Done ● Done ● Doing 		<ul style="list-style-type: none"> KAREN MELISSA CALLE BETANCUR KAREN MELISSA CALLE BETANCUR KAREN MELISSA CALLE BETANCUR KAREN MELISSA CALLE BETANCUR KAREN MELISSA CALLE BETANCUR KAREN MELISSA CALLE BETANCUR KAREN MELISSA CALLE BETANCUR

Figura 3. Backlog del producto

2.3 Diseño y arquitectura

Para el desarrollo de la aplicación, se planteó usar C#, el cual es un lenguaje de programación flexible, con el uso del Entity Framework debido a que este facilita la interacción con bases de datos relacionales las cuales se usaron en el proceso para almacenamiento de

información de usuarios e historias clínicas y proporciona características de seguridad muy importante para proteger los datos de los pacientes. Teniendo en cuenta esto, se hace uso de Blazor como marco para crear el frontend de la aplicación debido a que facilita la creación de interfaces de usuario, la creación de código modular y su alta compatibilidad con diferentes navegadores. Las imágenes de perfil de los usuarios y las solicitadas para la clasificación de pacientes se almacenarán en Azure Blob Storage. El desarrollo tendrá en cuenta el patrón WPA, el cual mejora la seguridad de la aplicación y permite la mantenibilidad y separa la aplicación a través de capas.

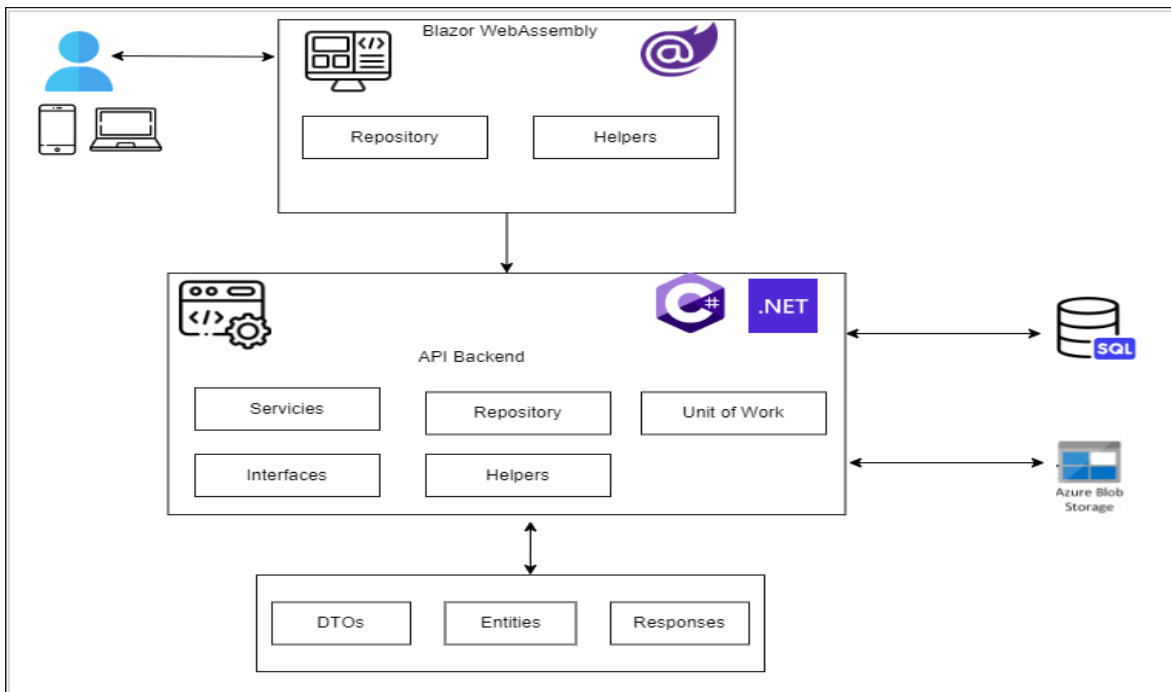


Figura 4. Arquitectura del proyecto

Esto, junto con la arquitectura responsiva, permitirá que la aplicación se vea bien en diferentes dispositivos y tamaños de pantalla, sin que se tenga que generar código adicional para esto, adicionalmente cargará gradualmente la aplicación, mejorando la experiencia. Patrones de diseño usados en el proyecto:

Tabla 2. Uso patrón repositorio

Nombre del patrón	<i>Patrón repositorio</i>
--------------------------	---------------------------

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Descripción	<i>Este patrón se aplica con el fin de crear una capa de abstracción entre la capa de acceso a datos y la capa empresarial</i>
Problema	<p><i>Testing del código se puede hacer difícil a el patrón repositorio hace que sea fácil de probar, usando mocks del repositorio en test unitarios los cuales simulan el comportamiento de las bases de datos</i></p> <p><i>Capa de acceso a datos requiere ser modificada podría suponer modificación en gran parte del código a puede ayudar a facilitar el mantenimiento de la capa de acceso a datos.</i></p>

Tabla 3. Uso patrón fachada

Nombre del patrón	<i>Patrón fachada (Facade)</i>
Descripción	<i>Se utiliza para proporcionar una interfaz sencilla para un subsistema complejo, ayudando a reducir la complejidad del subsistema</i>
Problema	<p><i>Se puede cambiar el subsistema sin afectar a los clientes.</i></p> <p><i>El cliente deberá acceder a parte de las funcionalidades del sistema, con este patrón exponemos las funcionalidades esenciales</i></p>

A continuación, se presenta la aplicación de principios SOLID en el proyecto

Tabla 4. Aplicación de principios SOLID

Principios SOLID	
Single Responsibility	Cada controlador en el proyecto maneja una única responsabilidad. Ejemplo: El AccountsController es responsable solo de las operaciones relacionadas con la cuenta del usuario.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Open/Closed	<p>Se hizo uso de interfaces para definir contratos para servicios, permitiendo que las implementaciones puedan ser extendidas sin modificar el código existente.</p> <p>Ejemplo: La interfaz <code>IRepository<T></code>, cualquier implementación de esta interfaz puede sustituir a otra sin afectar al sistema.</p>
Liskov Substitution	<p>En lugar de tener una única interfaz con muchos métodos, el sistema tiene interfaces más pequeñas y específicas.</p>
Interface Segregation	<p>Se realizó inyección de dependencias para invertir las dependencias de alto nivel en abstracciones.</p>
Dependency Inversion	<p>Los controladores no crean instancias de los servicios directamente. En cambio, reciben interfaces como dependencias, lo que facilita la prueba y el mantenimiento.</p>

2.4 Gestión del proyecto:

El proyecto, tomó como stakeholder a la profesora de odontología de la Universidad de Antioquia Laura Tamayo.

Para la gestión del proyecto, la herramienta usada fue Azure Devops, debido a la facilidad de adoptar la metodología ágil del proyecto.

De esta herramienta se utilizaron las siguientes funcionalidades:

- Organización del proyecto
- Creación del backlog del producto
- Planificación de los sprints
- Seguimiento de cada una de las historias de usuario
- Integración con repositorio de código.

2.3 Desarrollo de la aplicación

El desarrollo de la aplicación se dividió en sprints, utilizando la metodología ágil, donde cada sprint, buscaba implementar un conjunto de funcionalidades específicas. Durante el desarrollo, se crearon interfaces de usuario intuitivas, utilizando Blazor, integrando de manera eficiente el frontend con el backend elaborado en C#. Las funcionalidades implementadas incluyeron: registro en plataforma para estudiantes y pacientes, diligenciamiento de encuestas de clasificación para pacientes, clasificación de pacientes y agendamiento de citas.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Para el desarrollo, además, se creó un repositorio en GitHub, donde se realizaban commits regulares, asegurando que cada cambio fuera documentado y gestionado de manera eficiente.

2.4 Pruebas unitarias y de integración

Se utilizaron Microsoft.EntityFrameworkCore.InMemory y Moq para las pruebas unitarias y pruebas de integración, simulando el comportamiento de la base de datos y las dependencias externas y además, validando la integración entre la capa de datos y la capa de servicios.

La selección de Microsoft.EntityFrameworkCore.InMemory, fue debido a que su rapidez, facilidad del proceso de las pruebas y aislamiento (cada prueba se ejecuta en su propia instancia de base de datos

Moq: debido a que permite simular comportamientos y respuestas de objetos y servicios. Y se instalaron las extensiones Fine Code Coverage y Run Coverlet Report VS2022. Para poder medir la cobertura de las pruebas unitarias.

2.5 Pruebas de aceptación

Se realizaron pruebas con usuarios reales, donde se tenían planteado varios escenarios, validando que el sistema cumpliera con todos los requisitos, además, dichas pruebas permitieron recopilar comentarios y opiniones reales, importantes para realizar ajustes finales.

Encuestas de satisfacción

Para evaluar la satisfacción de los usuarios con el sistema, se diseñó una encuesta que incluía preguntas cerradas (con un valor a calificar) y preguntas abiertas (donde se esperaba que los usuarios dieran detalles), las cuales buscaban medir aspectos como la experiencia de usuario, facilidad de uso, eficiencia y la interfaz de usuario. La encuesta se realizó a través de GoogleForms y los datos se recolectan via online.

3. RESULTADOS Y DISCUSIÓN

3.1 Levantamiento de requisitos:

Durante el proceso de levantamiento de requisitos, se usaron diferentes técnicas que permitieron que los aspectos más importantes fueran cubiertos, 2 de las herramientas más importantes en esta etapa fueron el diagrama causa raíz y el diagrama de clases.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Diagrama causa-raíz: Esta herramienta, permitió identificar y comprender los problemas en los procesos de gestión de pacientes en las facultades de odontología. A continuación, se presenta su versión final

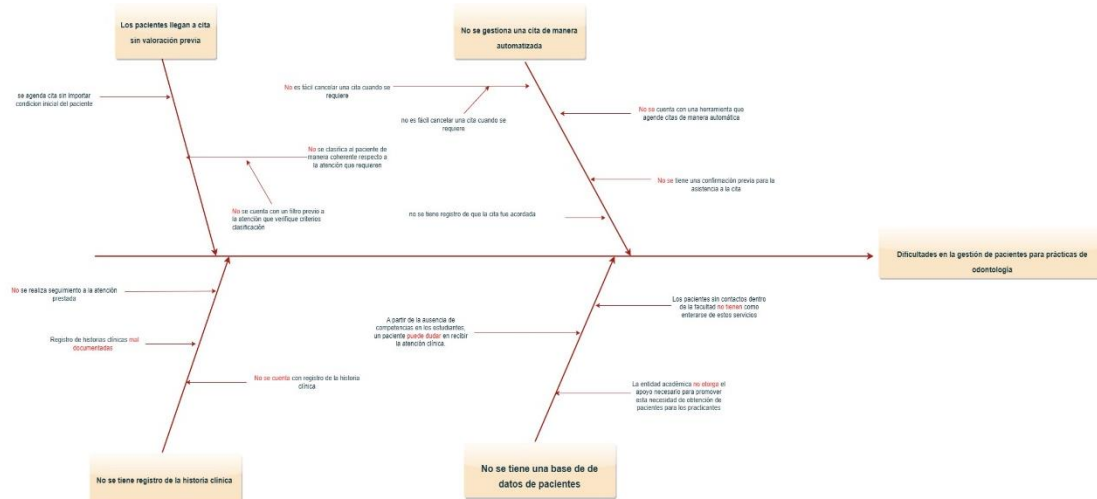


Figura 5. Diagrama causa-raíz

El diagrama, mostró varios problemas, como la falta de valoración previa de los pacientes, ausencia de automatización del proceso e inexistencia de bases de datos de pacientes, los cuales permitieron identificar las áreas a tratar dentro del contexto general.

Diagrama de clases: El cual permitió modelar la estructura del sistema, sus entidades y relaciones y de esta forma, ver cómo interactúan los diferentes componentes del sistema.

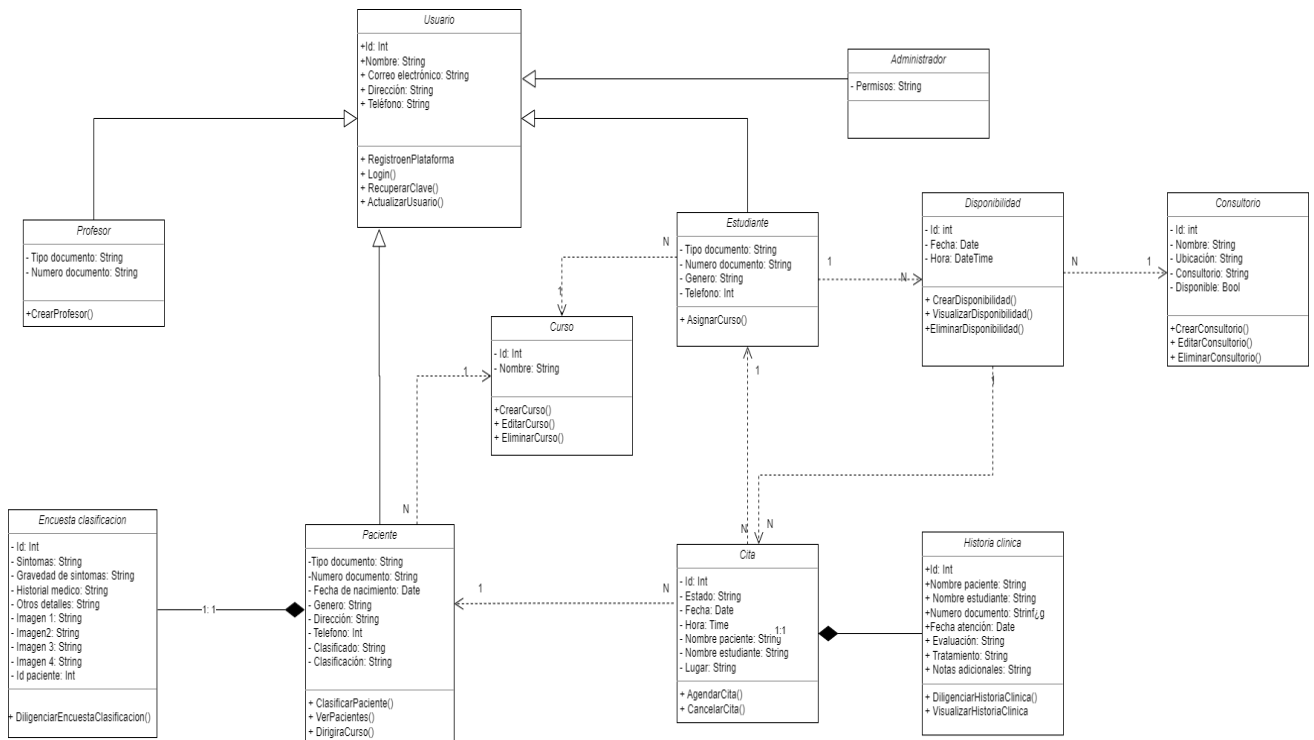


Figura 6. Diagrama de clases del proyecto

Dicho diagrama, permitió definir las principales clases a tener en el sistema, paso importante para la creación de las entidades en el Entity Framework y guiar el desarrollo del Backend del sistema.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

3.2 Identificación de requisitos del sistema

se identificaron y detallaron los siguientes requisitos funcionales relacionadas con el registro en plataforma:

44 HU_RF 001- Registrar pacientes

 KAREN MELISSA CALLE BETANCUR
  0 comments
Add tag

State	● Done	Area	OralData-Project
Reason	 Completed	Iteration	OralData-Project

Description

Yo como dueño del producto requiero que mi sistema permita registrar pacientes para poder agendarles citas.

Criterios de aceptación:

- Cuando las personas ingresen a la plataforma para registrarse, se espera que el sistema proporcione un formulario de registro en línea donde los pacientes puedan ingresar su información personal.
- Los campos requeridos en el formulario de registro deben incluir nombre completo, dirección, número de teléfono, dirección de correo electrónico, fecha de nacimiento, género y número de documento.
- Cuando la persona ingrese su correo electrónico, el sistema debe verificar la unicidad de dichos datos en la base de datos para evitar duplicados en el registro.
- Cuando la persona finalice el registro en plataforma, se debe cumplir que el sistema envíe un correo electrónico de confirmación al paciente para verificar su dirección de correo electrónico y poder continuar usando la aplicación.

Figura 7. Historia de usuario funcional 1

45 HU_RF 002- Registrar estudiantes

 KAREN MELISSA CALLE BETANCUR
  0 comments
Add tag

State	● Done	Area	OralData-Project
Reason	 Completed	Iteration	OralData-Project

Description

Yo como dueño del producto requiero que mi sistema permita registrar estudiantes para agendarles citas como profesionales.

Criterios de aceptación:

- Cuando el estudiante ingrese por primera vez en plataforma, el sistema debe proporcionar un formulario de registro en línea donde puedan ingresar su información personal.
- Los campos requeridos en el formulario de registro deben incluir nombre completo, número de teléfono, dirección de correo electrónico, número de documento, materia en curso.
- Cuando el estudiante ingrese su correo electrónico, el sistema debe verificar la unicidad de dichos datos en la base de datos para evitar duplicados en el registro.
- La plataforma para el registro de estudiantes solo permite acceso con el dominio de la universidad.
- Cuando el estudiante finalice el registro en plataforma, se debe cumplir que el sistema envíe un correo electrónico de confirmación al estudiante para verificar su dirección de correo electrónico.

Figura 8. Historia de usuario funcional 2

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

A continuación, se presentan las historias de usuario funcionales asociadas a la épica de clasificación de pacientes

46 HU_RF 003 - Diligenciar encuesta de clasificación

 KAREN MELISSA CALLE BETANCUR  0 comments [Add tag](#)

State	● Done	Area	OralData-Project
Reason	 Completed	Iteration	OralData-Project

Description

Yo como dueño de la plataforma requiero información del estado actual del paciente para conocer su necesidad.

Criterios de aceptación:

- Cuando el paciente finalice su registro, el sistema debe proporcionar un formulario que permita realizar una clasificación
- Dicha encuesta contará con los campos: Síntomas, gravedad de los síntomas, historial médico relevante, otros detalles
- Para finalizar con el formulario de la clasificación, el paciente debe adjuntar 4 fotos (una de frente, una de perfil, una sonriendo y abriendo la boca). El sistema deberá validar que se hayan cargado 4 archivos en formato (.jpg, .png), en caso de no ser proporcionados, debe arrojar un error "Recuerda subir las 4 imágenes en los formatos permitidos .jpg, .png)

Figura 9. Historia de usuario funcional 3

47 HU_RF 004 - Clasificar pacientes

 KAREN MELISSA CALLE BETANCUR  0 comments [Add tag](#)

State	● Done	Area	OralData-Project
Reason	 Completed	Iteration	OralData-Project

Description

Yo como dueño de la plataforma requiero que mi sistema permita clasificar a los pacientes para que sean atendidos por estudiantes de acuerdo a su necesidad.

Criterios de aceptación:

- El profesor podrá visualizar un listado con todos los pacientes que no se encuentran clasificados
- El profesor podrá ingresar a visualizar el detalle de la encuesta diligenciada seleccionando la línea en el listado del paciente específico. Donde encontrará Síntomas, gravedad de los síntomas, historial médico relevante, otros detalles y 4 imágenes adjuntas)
- Una vez analizada la información seleccionar en un drop-down-list la clasificación que dará al paciente (dicha clasificación corresponderá a curso en el cual se podrá atender su necesidad)

Figura 10. Historia de usuario funcional 4

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

A continuación, se presentan las historias de usuario funcionales asociadas al agendamiento de citas:

50 HU_RF 005 - Gestionar horarios de disponibilidad

 KAREN MELISSA CALLE BETANCUR  0 comments Add tag

State	● Done	Area	OralData-Project
Reason	 Completed	Iteration	OralData-Project

Description  

Yo como estudiante debo de tener la posibilidad de parametrizar la disponibilidad semanal de las clínicas, para que el paciente pueda visualizar las franjas de atención.

Criterios de aceptación:

- El sistema permitirá al estudiante configurar sus horarios de disponibilidad en un horario entre las 8am y las 6pm. Horas por fuera de este límite no saldrán para seleccionarse
- Las citas estarán programadas para una hora, por tanto, si el estudiante selecciona un día y poner 8-12m, el sistema configurará 4 horarios de disponibilidad para ese día: 8am-9am, 9am-10 am, 10am-11am, 11am-12m
- Cuando el estudiante finalice la parametrización de su disponibilidad, el sistema debe de notificar que las configuraciones fueron exitosas a través de un mensaje de confirmación.
- Un estudiante podrá visualizar las disponibilidades agregadas en el sistema y eliminarlas si y solo si no tiene una cita programada para ese horario.

Figura 11. Historia de usuario funcional 5

51 HU_RF 006 - Visualización de pacientes disponibles

 KAREN MELISSA CALLE BETANCUR  0 comments Add tag

State	● Done	Area	OralData-Project
Reason	 Completed	Iteration	OralData-Project

Description

Yo como dueño del producto requiero que mi sistema permita visualizar los pacientes disponibles para agendar citas.

Criterios de aceptación:

- Cuando el paciente haya sido clasificado, el sistema permitirá agendar citas con dicho paciente. Si un paciente no ha sido clasificado, esta opción no estará habilitada
- Una vez el estudiante ingrese a la opción de agendar citas, el sistema debe de mostrar el listado de pacientes disponibles y clasificados de acuerdo a la competencia del estudiante (día, hora y nombre del estudiante)

Figura 12. Historia de usuario funcional 6

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

52 HU_RF 007 - Agendar citas

 KAREN MELISSA CALLE BETANCUR 0 comments Add tag

State ● Done Area OralData-Project
Reason 🔒 Completed Iteration OralData-Project

Description

Yo como usuario de la plataforma requiero poder agendar citas en el sistema, proporcionando detalles tales como fecha y hora.

Criterios de aceptación:

- Una vez seleccionado el paciente, el estudiante podrá visualizar información diligenciada en su clasificación
- El sistema permitirá seleccionar día y hora para agendar cita de acuerdo con las disponibilidades del estudiante
- Una vez asignada la cita, los usuarios (pacientes y estudiantes) deben de recibir una notificación por parte de la plataforma Oral Data

Figura 13. Historia de usuario funcional 7

A continuación, se presentan las historias no funcionales del sistema

43 HU_NF 001 - Garantizar contraseña segura

 KAREN MELISSA CALLE BETANCUR 0 comments Add tag

State ● Done Area OralData-Project
Reason 🔒 Completed Iteration OralData-Project

Description

Como líder de arquitectura y soluciones TI requiero que mi sistema almacene de manera segura las credenciales de los usuarios.



Criterios de aceptación:

- El sistema exigirá unas políticas de contraseña (Caracteres especiales, mayúsculas, números y un mínimo de 8 caracteres)
- El sistema generará un JWT (JSON Web Token) seguro al autenticarse, que contendrá información esencial del usuario y expirará en un tiempo determinado para garantizar la seguridad.

Figura 14. Historia de usuario no funcional 1

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

48 HU_NF 002 - Factibilizar Usabilidad

 KAREN MELISSA CALLE BETANCUR
  0 comments
[Add tag](#)

State	● Done	Area	OralData-Project
Reason	 Completed	Iteration	OralData-Project

Description

Como líder de arquitectura y soluciones TI requiero que mi plataforma de gestión odontológica Oral DATA presente un manejo sencillo y de fácil usar


Criterios de aceptación:

- Interfaz fácil de entender, procesos multinivel con enfoques específicos a lo que el usuario requiere.
- Diseño acorde al servicio que manifiesta la plataforma
- El sistema debe contar con una interfaz de usuario intuitiva con navegación clara y consistente.

Figura 15. Historia de usuario no funcional 2

49 HU_NF 003 - Garantizar escalabilidad

 KAREN MELISSA CALLE BETANCUR
  0 comments
[Add tag](#)

State	● Done	Area	OralData-Project
Reason	 Completed	Iteration	OralData-Project

Description

Como líder de arquitectura y soluciones .TI requiero que mi plataforma de gestión odontológica Oral DATA, tenga la función de aumentar su performance en caso de que mi la infraestructura en la que esta implementada Oral DATA este procesando más información de forma simultánea.



Criterios de aceptación:


- El sistema debe poder escalarse agregando más recursos o servidores según sea necesario para manejar un mayor volumen de usuarios o datos.
- Implementar pruebas unitarias y de integración para garantizar la estabilidad del código y facilitar futuras actualizaciones.
- Utilizar una arquitectura modular que permita agregar nuevas características y funcionalidades sin afectar negativamente a las partes existentes de la aplicación.

Figura 16. Historia de usuario no funcional 3

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

53 HU_RNF 004 - Ratificar mantenibilidad

 KAREN MELISSA CALLE BETANCUR  0 comments [Add tag](#)

State	 Done	Area	OralData-Project
Reason	 Completed	Iteration	OralData-Project

Description

Como líder de arquitectura y soluciones .TI requiero que mi plataforma de gestión odontológica Oral DATA esté desarrollado de tal forma que pueda ser modificado efectiva y eficientemente

Criterios de aceptación:

- Deben proporcionarse documentación clara y completa para el mantenimiento y la administración del sistema.
- El sistema debe ser desarrollado con las mejores prácticas de desarrollo de software y seguir un estilo de codificación consistente
- El sistema debe ser diseñado de manera modular para permitir actualizaciones y mantenimiento sin interrupciones importantes.

Figura 17. Historia de usuario no funcional 4

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

3.3 Desarrollo de la aplicación

El proyecto OralData, está formado por 4 proyectos, cada uno con una función distinta, a continuación, se describe la función de cada uno dentro del sistema y su distribución interna.

1. OralData.Backend: Contiene la lógica del servidor y la gestión de datos y lógica del negocio.

En la siguiente figura se muestra su distribución

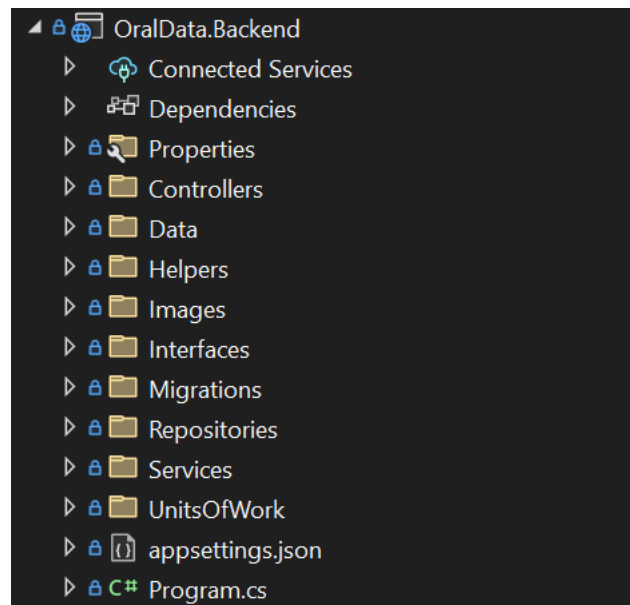


Figura 18. Estructura OralData.Backend

Su estructura consta de

- **Controllers:** Contiene los controladores que manejan las solicitudes http y retornan respuestas a los clientes
- **Data:** Contiene el contexto de las bases de datos y configuraciones asociadas
- **Helpers:** Clases auxiliares del proyecto para el backend
- **Images:** Imágenes usadas en el backend
- **Interfaces:** Proporciona una abstracción en el sistema, donde se definen los contratos para los servicios y repositorios
- **Migrations:** Archivos de migración que genera el Entity Framework que reflejan los cambios en los esquemas de la base de datos
- **Repositories:** En esta se implementa el patrón repositorio
- **Units of Work:** Implementación de del patrón unidad de trabajo en el proyecto

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

2. OralData.Frontend: Maneja la lógica del lado del cliente, utilizando Blazor para construir la interfaz de usuario

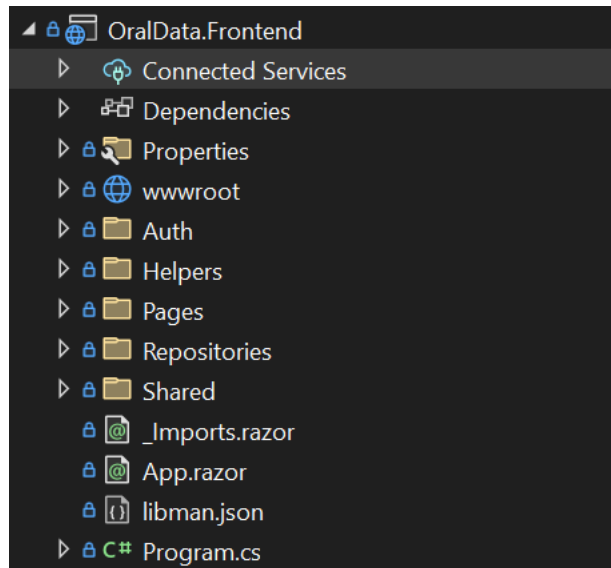


Figura 19. OralData.Frontend

Su estructura consta de:

- Auth: Maneja la autenticación, registro y recuperación de claves del lado del cliente
 - Helpers: Clases auxiliares del proyecto para el frontend
 - Pages: Contiene las diferentes páginas de Blazor de interfaz de usuario
 - Repositories: Implementación de patrones de repositorio para el frontend
3. OralData.Shared: Contiene las clases compartidas entre el frontend y el backend, entre las que se encuentran las entidades, DTOs, enumeraciones; de manera que se tenga consistencia entre los datos y modelos

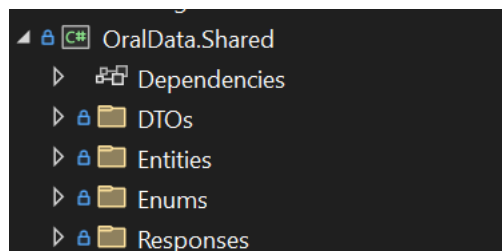


Figura 20. OralData.Shared

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Su estructura conta de:

- DTOs: clases que manejan la tranferencia de datos desde el lado del cliente hacia el servidor
 - Entities: Clases que representan entidades de dominio de la solución
 - Enum: Grupos especiales de datos donde se asignan sus respectivos valores
 - Responses: Representación de la respuesta de las API.
4. OralData.UnitTest: Dedicado a las pruebas unitarias del sistema, asegurando que cada componente cumpla con su respectiva función

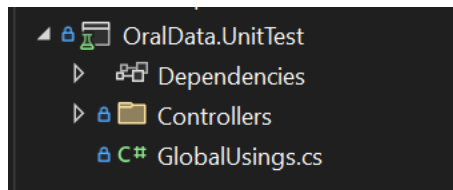


Figura 21. OralData.UnitTest

A continuación, se presentan imágenes del backend desarrollado con sus diferentes métodos desde la vista de swagger.

La figura 22, muestra el conjunto de endpoints de la API relacionados con la gestión de cuentas en el sistema OralData, permitiendo operaciones relacionadas como autenticación y gestión de usuarios.

Accounts		^
POST	/api/Accounts/RecoverPassword	v 🔒
POST	/api/Accounts/ResetPassword	v 🔒
POST	/api/Accounts/changePassword	v 🔒
PUT	/api/Accounts/assignCourse/{userId}	v 🔒
PUT	/api/Accounts/updateclassif-flag	v 🔒
PUT	/api/Accounts	v 🔒
GET	/api/Accounts	v 🔒
POST	/api/Accounts/CreateUser	v 🔒
POST	/api/Accounts/Login	v 🔒
POST	/api/Accounts/ResedToken	v 🔒
GET	/api/Accounts/ConfirmEmail	v 🔒

Figura 22. Endpoints de la API de Gestión de Cuentas en el Sistema OralData

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

La figura 23, muestra el conjunto de endpoints de la API relacionados con la gestión de disponibilidades de estudiantes en el sistema.

Availabilities	
GET	/api/Availabilities/{id}
DELETE	/api/Availabilities/{id}
GET	/api/Availabilities/fullAvailability/{date}
GET	/api/Availabilities
GET	/api/Availabilities/totalPages
POST	/api/Availabilities/availability
PUT	/api/Availabilities/update-available{id}

Figura 23. Endpoints de la API de Gestión de Disponibilidades de estudiantes

Estos endpoints de la figura 24, proporcionan las funcionalidades necesarias para gestionar las encuestas de clasificación de los pacientes.

ClassificationSurvey	
GET	/api/ClassificationSurvey/{id}
GET	/api/ClassificationSurvey
GET	/api/ClassificationSurvey/totalPages
POST	/api/ClassificationSurvey/full
PUT	/api/ClassificationSurvey/full
PUT	/api/ClassificationSurvey/sheduled{id}

Figura 24. Endpoints de la API de Gestión de Encuestas de Clasificación en el Sistema

La siguiente figura, muestra el conjunto de endpoints de la API relacionados con la gestión de cursos en el sistema OralData.

Courses	
GET	/api/Courses
POST	/api/Courses
PUT	/api/Courses
GET	/api/Courses/totalPages
GET	/api/Courses/combo
GET	/api/Courses/{id}
DELETE	/api/Courses/{id}

Figura 25. Endpoints de la API de Gestión de Cursos

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

La imagen 26, muestra los endpoints que permiten realizar diversas operaciones para manejar las citas de los pacientes.

Appointments	
GET	/api/Appointments/{id}
GET	/api/Appointments
GET	/api/Appointments/totalPages
POST	/api/Appointments/shedule-appointment/{userId}/{surveyDetailsId}
PUT	/api/Appointments/updatestate

Figura 26. Endpoints de la API de Gestión de Citas

La siguiente figura, muestra los endpoints que permiten realizar diversas operaciones para manejar los dominios de correo electrónico que los estudiantes pueden usar para registrarse en el sistema.

AllowedStudentDomains	
GET	/api/AllowedStudentDomains
POST	/api/AllowedStudentDomains
PUT	/api/AllowedStudentDomains
GET	/api/AllowedStudentDomains/totalPages
GET	/api/AllowedStudentDomains/combo
GET	/api/AllowedStudentDomains/{id}
DELETE	/api/AllowedStudentDomains/{id}

Figura 27. Endpoints de la API de Gestión de Dominios Permitidos para Estudiantes

La combinación de una arquitectura de endpoints bien definida y el uso de interfaces ha permitido desarrollar un sistema robusto y escalable. Esto asegura que cada componente del sistema pueda ser modificado y probado de manera independiente, mejorando la calidad del software, facilitando futuras expansiones y además facilitó la implementación de diversas funcionalidades y la realización de pruebas unitarias e integración.

OralData proporciona una interfaz de usuario intuitiva y accesible, diseñada con Blazor para ofrecer una experiencia fluida y moderna. En las siguientes capturas de pantalla, se detallan algunas de las principales funcionalidades de la aplicación.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

En la siguiente figura se visualiza el menú de inicio de la aplicación y adicionalmente se observan las opciones presentes para el usuario administrador quien podrá agregar dominios de correo permitidos para estudiantes, agregar profesores, cursos y ciudades.



Figura 28. Pagina de inicio- Menu administrador

La pantalla de registro de pacientes donde los pacientes diligencian su información básica

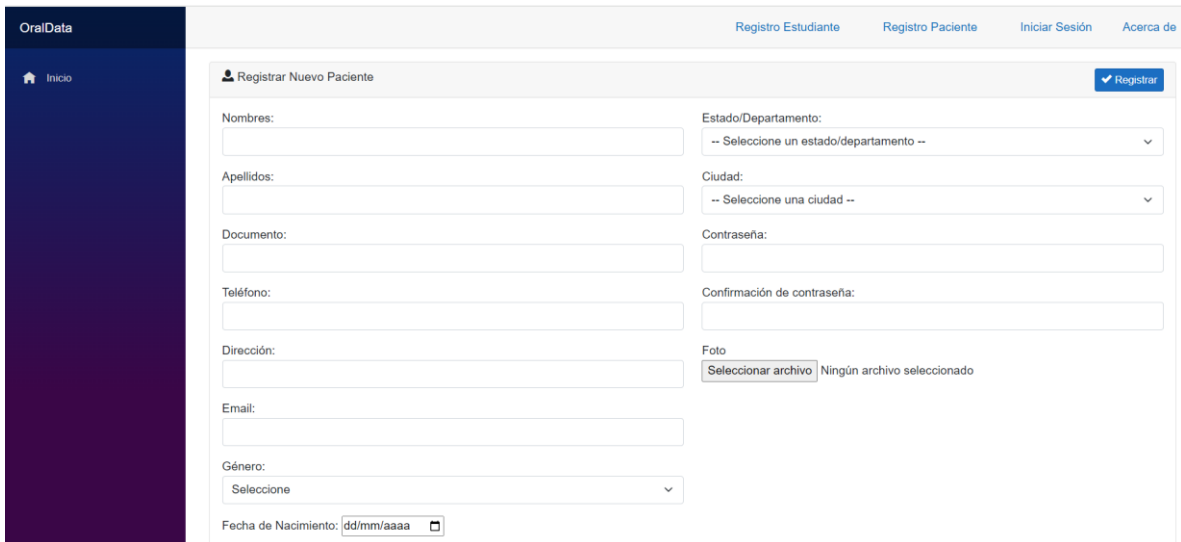


Figura 29. Interfaz de usuario para registro de pacientes

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

La pantalla de registro de estudiantes, donde diligencian su información básica y el curso que se encuentran cursando.

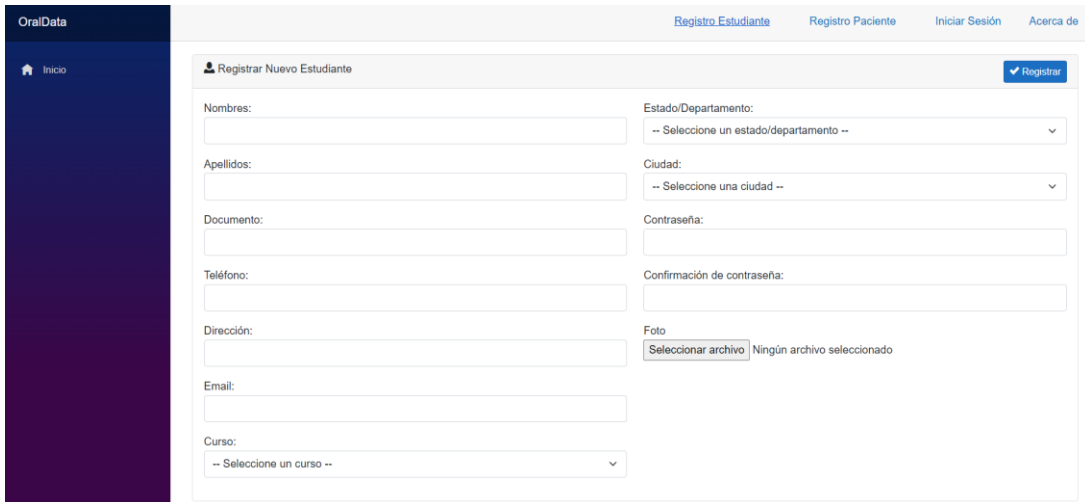


Figura 30. Interfaz de usuario para registro de estudiantes

Pantalla de inicio de sesión a la aplicación

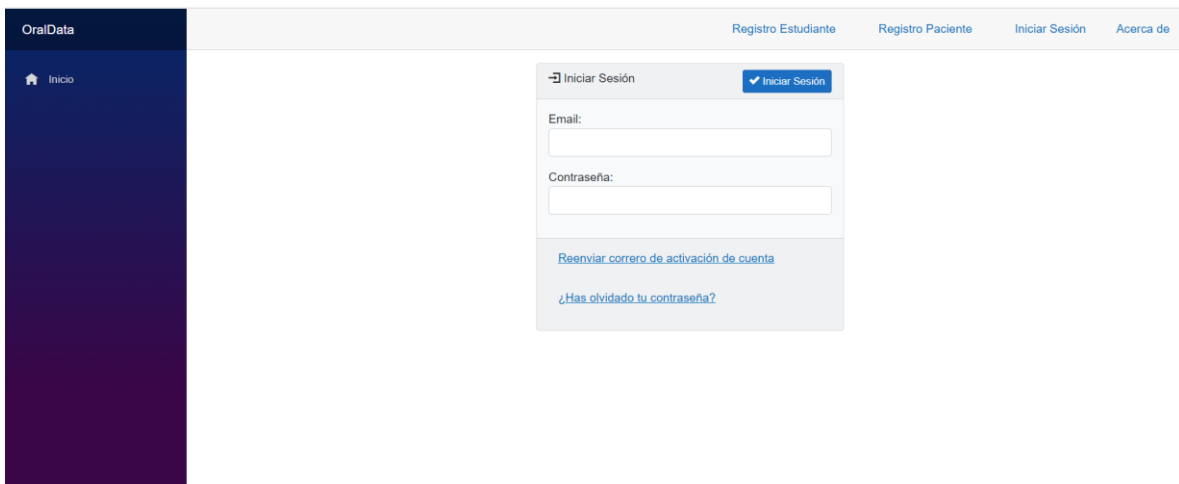


Figura 31. Interfaz de usuario para inicio de sesión

Una vez el paciente confirme su correo electrónico, disponible en el menú tendrá la opción de Clasificación, donde podrá diligenciar la información que le permita ser remitido al curso correcto y ser atendido por un estudiante en capacidad de hacerlo.

Figura 32. Interfaz de usuario para que pacientes diligencien encuesta de clasificación

A continuación, se muestra un ejemplo de una encuesta de clasificación diligenciada por un paciente

Figura 33. Encuesta de clasificación diligenciada

En la siguiente figura, logueado desde el perfil de un profesor, se observa las encuestas de clasificación diligenciadas por pacientes, donde un paciente nuevo podrá ser clasificado por el profesor.

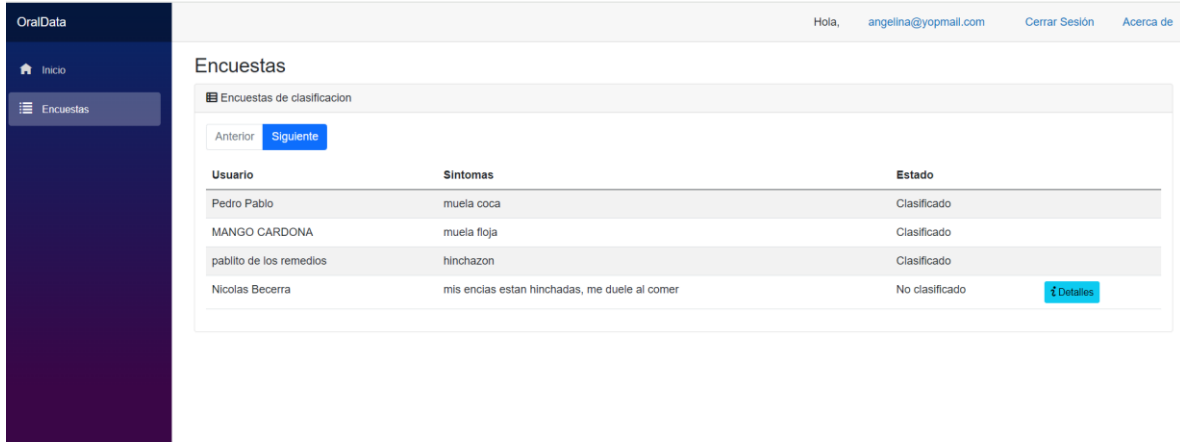


Figura 34. Interfaz de usuario ver encuestas diligenciadas por pacientes

En la siguiente figura se puede observar la información que el profesor visualiza al entrar al detalle de la Encuesta de un paciente específico y de acuerdo con lo que vea selecciona el curso al que lo dirigirá.

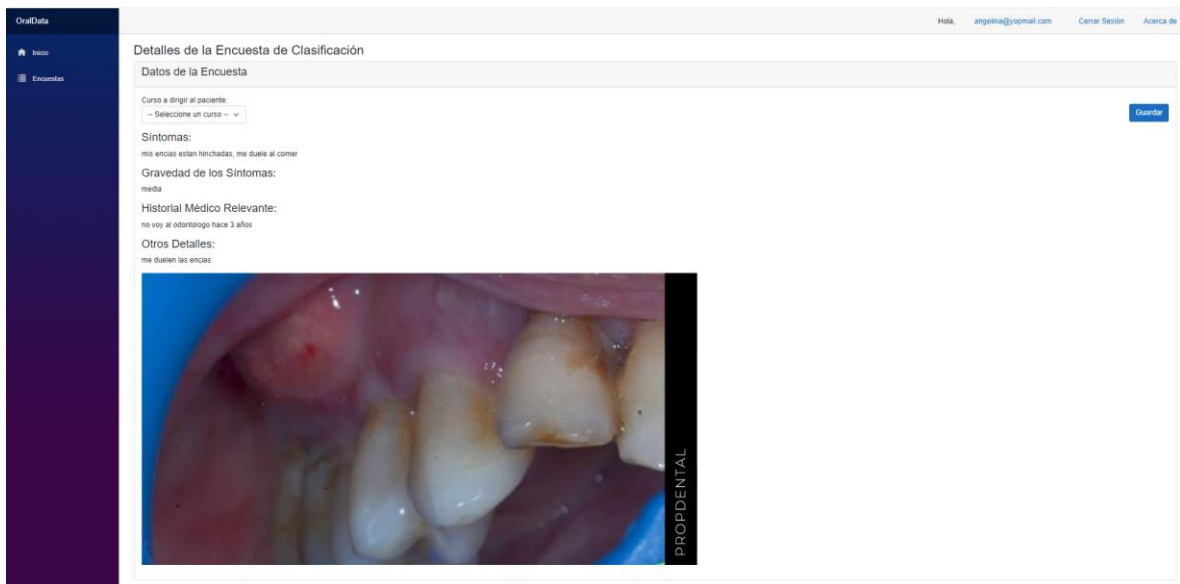
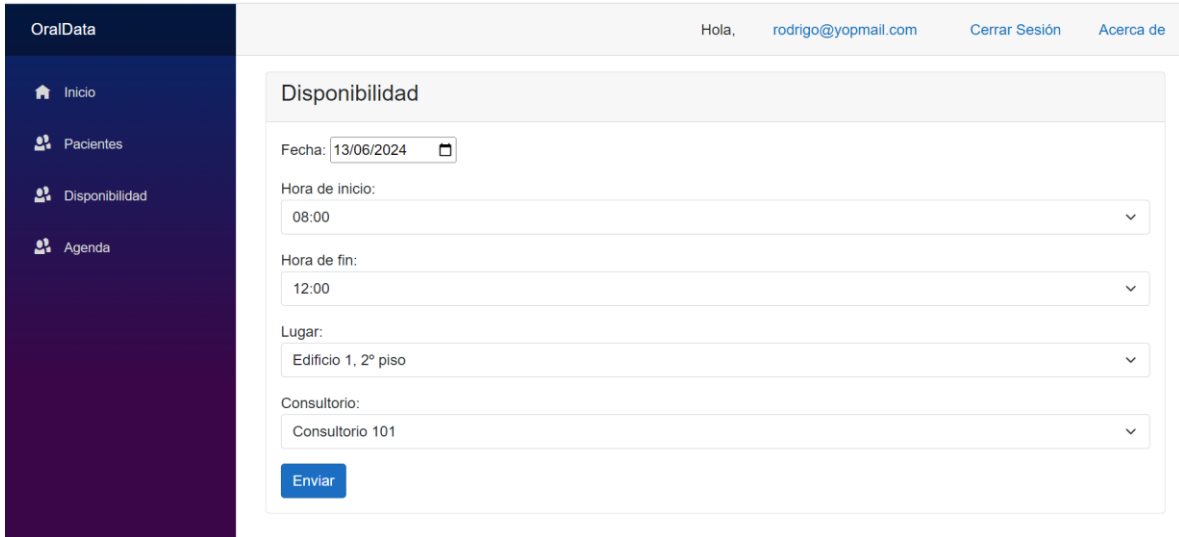


Figura 35. Profesor visualiza encuesta y clasifica

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

La figura 36 permite ver la manera en que un estudiante puede crear sus horarios de disponibilidad.



OralData Hola, rodrigo@yopmail.com [Cerrar Sesión](#) [Acerca de](#)

Disponibilidad

Fecha:

Hora de inicio:

Hora de fin:

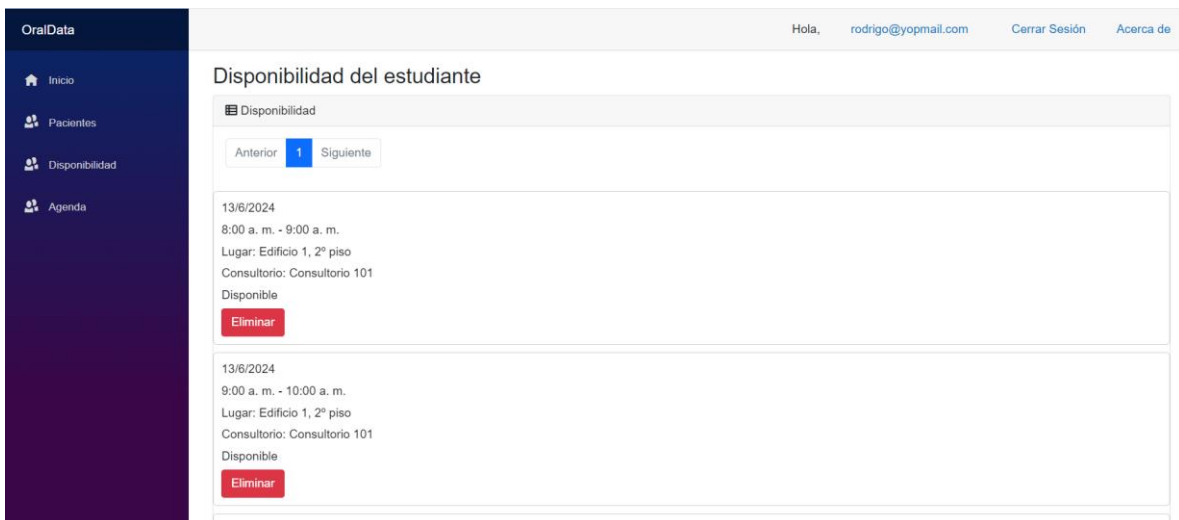
Lugar:

Consultorio:

[Enviar](#)

Figura 36. Interfaz de usuario para estudiante ingresar horario de disponibilidad

Una vez creados los horarios de disponibilidad, el estudiante podrá visualizarlos y en caso de no tener citas agendadas, podrá eliminar la franja horaria deseada.



OralData Hola, rodrigo@yopmail.com [Cerrar Sesión](#) [Acerca de](#)

Disponibilidad del estudiante

Disponibilidad

Anterior **1** Siguiente

13/6/2024

8:00 a. m. - 9:00 a. m.

Lugar: Edificio 1, 2º piso

Consultorio: Consultorio 101

Disponible

[Eliminar](#)

13/6/2024

9:00 a. m. - 10:00 a. m.

Lugar: Edificio 1, 2º piso

Consultorio: Consultorio 101

Disponible

[Eliminar](#)

Figura 37. Estudiante visualiza su disponibilidad

En la figura 38, se puede ver la forma en la que el estudiante puede ver los pacientes que han sido dirigidos al curso que está cursando

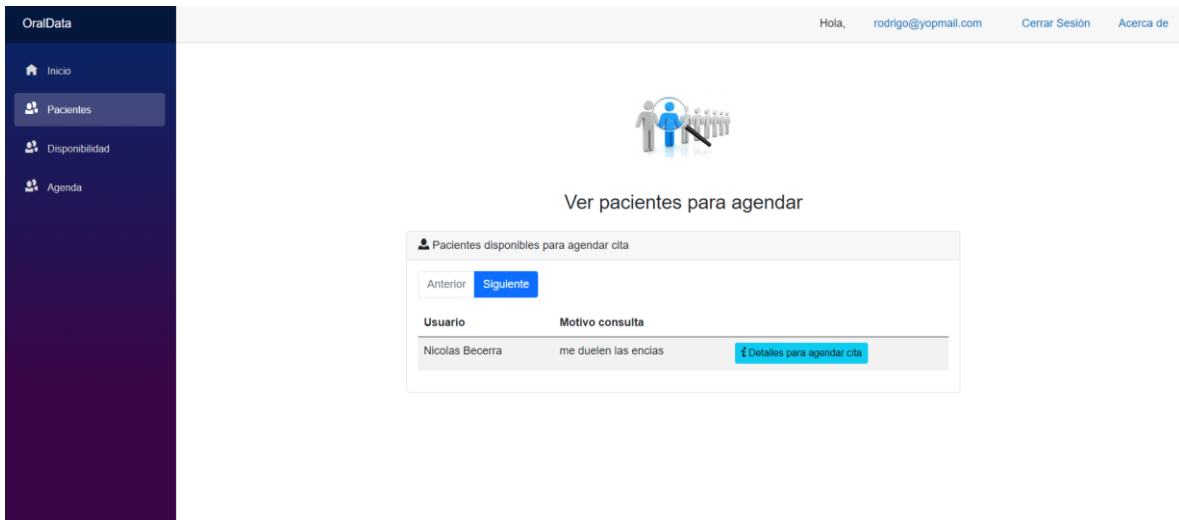


Figura 38. Interfaz de usuario para estudiante ver pacientes disponibles

La figura 39, permite visualizar el agendamiento que un estudiante ha realizado con un paciente y su respectivo mensaje de confirmación.

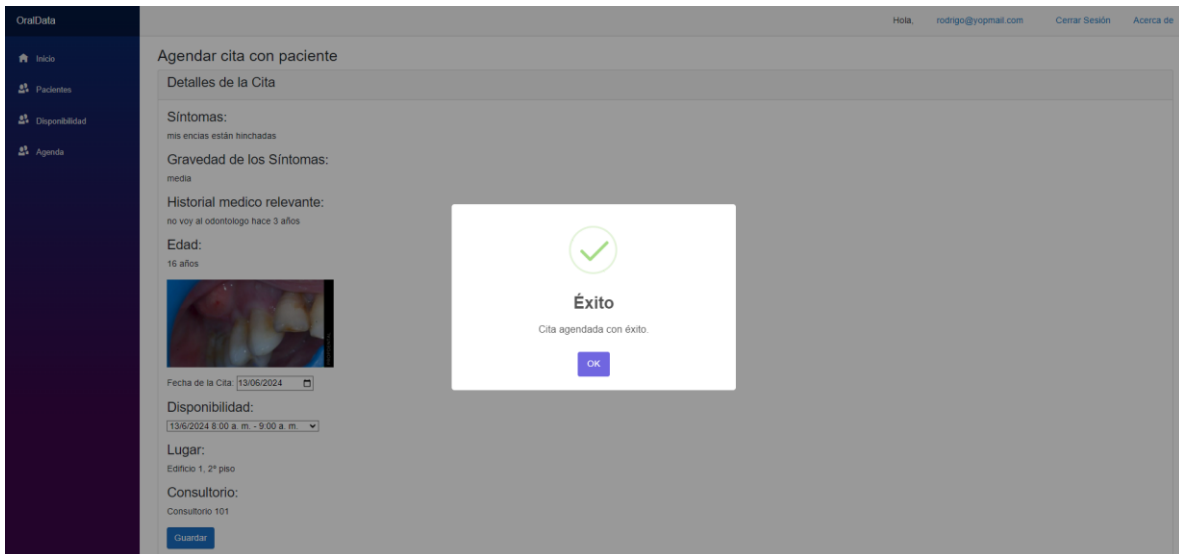


Figura 39. Estudiante agenda cita con paciente

La siguiente figura, permite visualizar desde el perfil de un estudiante, el historial de citas médicas programadas.

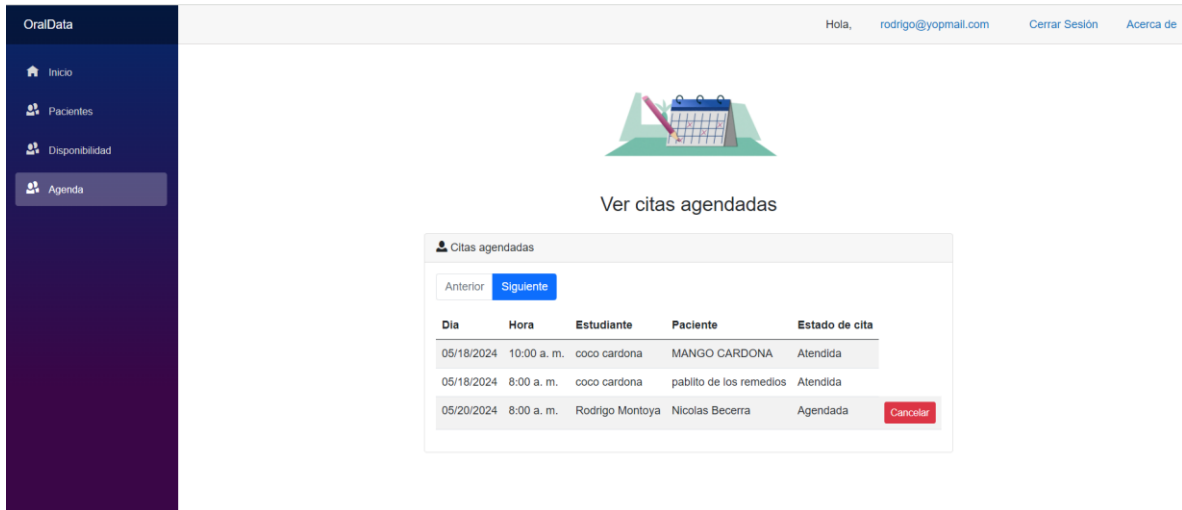


Figura 40. Interfaz para estudiante ver historial de citas agendadas

La siguiente figura, permite visualizar desde el perfil de un paciente, el historial de citas médicas programadas.

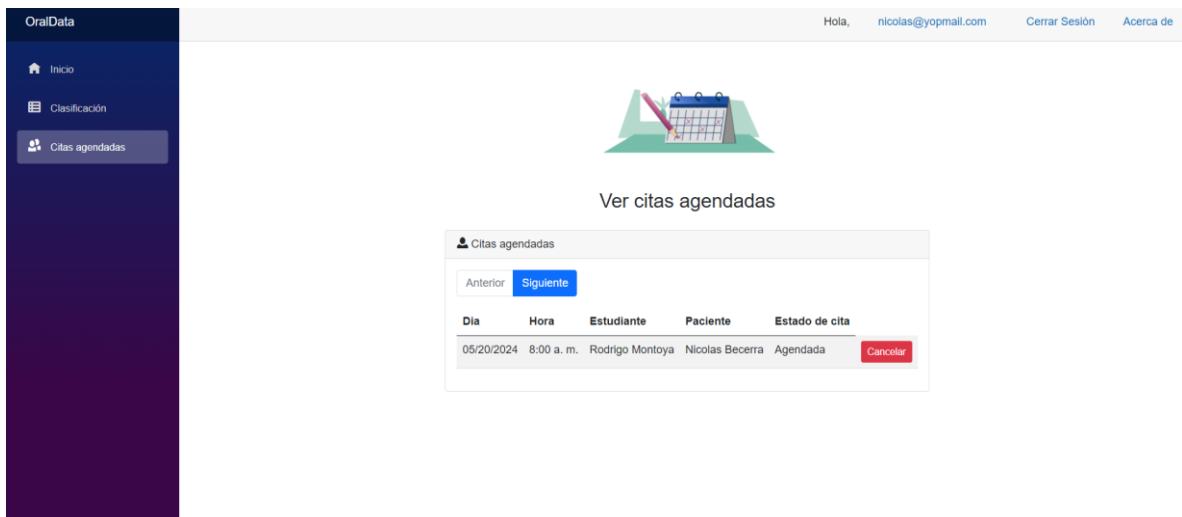


Figura 41. Interfaz para paciente ver historial de citas agendadas

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

3.4 Pruebas unitarias y de integración

Las pruebas de integración se enfocaron en validar la interacción entre los diferentes componentes del sistema. Se probaron todos los controladores del backend y cada uno de sus métodos para asegurar que funcionaran correctamente en conjunto.

En la siguiente figura, se presenta el resultado de la ejecución de las pruebas para los controladores en el backend, donde en total se crearon 134 pruebas a los diferentes métodos de los controladores.

Test	Duration	T.	Error Message
✓ OralData.UnitTest (134)	8,6 min		
✓ OralData.UnitTest.Controllers (134)	8,6 min		
✓ AccountsControllerTest (29)	46,6 sec		
✓ AgendasControllerTest (14)	16 sec		
✓ AllowedStudentDomainsControllerTest (8)	12,2 sec		
✓ AppointmentsControllerTest (26)	2 min		
✓ AvailabilitiesControllerTest (12)	56,3 sec		
✓ CitiesControllerTest (2)	1,4 sec		
✓ ClassificationSurveyControllerTest (19)	1,1 min		
✓ CountriesControllerTest (5)	29,7 sec		
✓ CoursesControllerTest (2)	1,2 min		
✓ GenericControllerTest (10)	43 sec		
✓ StatesControllerTest (4)	16,7 sec		
✓ TeacherProfilesControllerTest (3)	34,9 sec		

Figura 42. Pruebas unitarias creadas

Para cada método, se validaron los diferentes caminos de ejecución posibles, incluyendo escenarios de éxito, error y alternativos de la siguiente manera

Pruebas a métodos:

- Se verificó que los métodos post crearan correctamente el recurso solicitado (por ejemplo, un nuevo usuario tipo paciente) y retornaran las respuestas adecuadas.
- Se aseguró que los métodos get devolvieran correctamente la información de la base de datos y se retornara en el correcto formato
- Se validó que los métodos put actualizaran exitosamente los recursos solicitados
- Se probaron los métodos delete para garantizar que el recurso solicitado si fuera eliminado correctamente

Pruebas a los diferentes caminos:

- Se verificó que el método completara la operación de manera exitosa
- Se validó que el método manejara situaciones como entradas inválidas, recursos no encontrados y errores de aplicación en la ejecución de operaciones
- Se validaron caminos alternativos, asegurando que el sistema respondiera de manera correcta.
- Se realizaron pruebas para asegurarse de que las citas se registraran correctamente en la base de datos y que las notificaciones se enviaran oportunamente.

La utilización de interfaces permitió la creación de mocks para pruebas unitarias, lo que hizo posible probar componentes individuales de manera aislada, mejorando la cobertura de pruebas y la detección temprana de errores, contribuyendo a un sistema más robusto y confiable.

Dichas pruebas, representan una cobertura del 73.1% de líneas de código como se muestra en la siguiente figura

Assemblies:	3
Classes:	61
Files:	61
Covered lines:	3143
Uncovered lines:	1151
Coverable lines:	4294
Total lines:	7637
Line coverage:	73.1% (3143 of 4294)

Figura 43. Cobertura de pruebas unitarias

Dicho resultado, refleja un enfoque exhaustivo, buscando asegurar que gran parte del código haya sido probado, sin embargo, hay controladores donde se podría ampliar la cobertura (como el caso del controlador de citas, para el tema de las diferentes excepciones que se pueden presentar).

Aunque el 73.1% es una buena cobertura, el objetivo es incrementarla para mejorar aún más la robustez del sistema. Las áreas de mejora incluyen:

- Ampliación de Pruebas a Escenarios Más Complejos: Incluir más casos de prueba para escenarios de borde y excepciones.
- Pruebas de Integración Más Detalladas: Aumentar la cobertura de integración para validar la interacción entre módulos.

3.5 Pruebas de aceptación

La etapa de pruebas de aceptación estuvo dividida en las siguientes fases:

La estrategia de pruebas se dividirá en las siguientes etapas

1. Planificación y diseño: En esta etapa, se identifica y documentan los escenarios de prueba específicos de cada funcionalidad, teniendo en cuenta los criterios de aceptación. Esta etapa incluye, además, los datos de prueba, pasos de reproducción y resultados esperados
2. Ejecución de pruebas: Se realizan pruebas manuales para validar las funcionalidades, usabilidad y seguridad.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

3. Monitoreo y seguimiento: Se documenta y analiza los resultados de las pruebas ejecutadas, de manera que se identifiquen los riesgos, y errores en código, para poder decidir los ajustes necesarios en el código y mejorar la calidad de la aplicación.
4. Métricas: Se evalúa la cobertura de las pruebas, satisfacción del usuario, número de defectos encontrados.

Para las pruebas de aceptación, se definieron los siguientes escenarios de pruebas, basados en las historias de usuario y sus criterios de aceptación

Prueba # 1 Registro a la aplicación Oral Data

Objetivo de la prueba

Verificar la función de registro del software de Oral Data.

Descripción de la prueba

- Verificar los servicios de registro al software con la validación de creación del perfil a través de un correo electrónico.
- Verificar que los campos requeridos en el formulario de registro incluyen nombre completo, dirección, número de teléfono, dirección de correo electrónico, fecha de nacimiento, género y número de documento.
- Verificar que el sistema valida la unicidad del correo electrónico en la base de datos para evitar duplicados en el registro.
- Verificar que, tras el registro, el sistema envía un correo electrónico de confirmación para validar la dirección de correo electrónico y permitir que los usuarios continúen usando la aplicación.
- Verificar que el sistema asigna de forma estándar el rol correspondiente (paciente o estudiante) durante el registro.

Resultados Esperados

- Durante el proceso de registro, el software debe exigir un correo electrónico para validar la creación del perfil en calidad de paciente. El correo generará la orquestación para validar la autenticidad del registro y permitir el acceso a la plataforma.
- El sistema debe asignar de manera estándar en cualquier registro de paciente sus respectivos permisos y accesos, asegurando que el rol (paciente o estudiante) se asigna correctamente.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Prueba # 2 Verificación de acceso a la aplicación de Oral Data

Objetivo de la prueba

Verificar que el servicio de acceso a la plataforma de Oral Data funcione de forma adecuada, asegurando que los roles y permisos asignados durante el registro se aplican correctamente.

Descripción de la prueba

- Acceder con el usuario y contraseña generados a partir del registro.
- Verificar la funcionalidad (permisos a nivel de plataforma) del rol de paciente asignado a través del registro.
- Verificar la funcionalidad (permisos a nivel de plataforma) del rol de estudiante asignado a través del registro.
- La opción de restablecimiento de contraseña debe de generar también una orquestación vía correo electrónico para validar la solicitud a partir de un perfil de usuario ya existente.

Resultados esperados

- El software debe presentar la primera interfaz de menú principal de navegación al momento de acceder, esta interfaz debe exponer los módulos de servicios que se van a gestionar a partir del proceso académico odontológico.
- El rol estándar de paciente solo tendrá acceso a módulo para diligenciar encuesta de clasificación y visualización de citas agendadas.
- El rol de estudiante debe tener acceso a funcionalidades adicionales específicas de su rol, como la gestión de citas

Prueba # 3 Proceso de encuesta de clasificación de pacientes

Objetivo de la prueba

Validar la clasificación de pacientes a partir de la encuesta inicial presentada para inicializar un servicio odontológico si almacene y procese la información que se describe en la encuesta.

Descripción de la prueba

Completar en su totalidad la encuesta, principalmente en el apartado de valoración inicial odontológica.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Verificar la sesión de carga de imágenes en relación con el defecto odontológico, asegurando que se cumplan los criterios de aceptación definidos.

Resultados esperados

- El sistema debe permitir a los pacientes completar la encuesta de clasificación, incluyendo todos los campos requeridos (síntomas, gravedad de los síntomas, historial médico relevante, otros detalles).
- El sistema debe validar y aceptar las imágenes adjuntas en los formatos permitidos (.jpg, .png).
- La información de la encuesta debe almacenarse sistematizadamente en el software, permitiendo a los especialistas de odontología realizar la clasificación.

Prueba # 4 Clasificación docente-paciente

Objetivo de la prueba

Verificar que la clasificación de pacientes ejecutada por los profesores se registre correctamente y que la definición de la clasificación se almacene adecuadamente en el software.

Descripción de la prueba

- Se ingresa a la plataforma con un perfil de profesor
- Se valida visualización de nuevas encuestas diligenciadas
- Validar la segmentación de asignación de clasificaciones en las encuestas completadas por los pacientes.
- Realizar diferentes pruebas con el servicio de clasificación, incluyendo la visualización, edición y asignación de clasificaciones.

Resultados esperados

- El profesor podrá visualizar un listado con todos los pacientes que no se encuentran clasificados.
- El profesor podrá ingresar a visualizar el detalle de la encuesta diligenciada seleccionando la línea en el listado del paciente específico. Donde encontrará síntomas, gravedad de los síntomas, historial médico relevante, otros detalles y 4 imágenes adjuntas.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

- Una vez analizada la información, el profesor podrá seleccionar en un drop-down-list la clasificación que dará al paciente (dicha clasificación corresponderá a un curso en el cual se podrá atender su necesidad).

Prueba # 6: Gestión de Horarios de Disponibilidad

Objetivo de la Prueba

Verificar que los estudiantes puedan parametrizar su disponibilidad semanal de las clínicas, permitiendo a los pacientes visualizar las franjas de atención disponibles.

Descripción de la Prueba

- Validar la funcionalidad del módulo de gestión de horarios de disponibilidad.
- Configurar horarios de disponibilidad en un horario entre las 8 am y las 6 pm.
- Confirmar que se generen las disponibilidades creadas para una hora específica.
- Validar que los estudiantes puedan visualizar y eliminar las disponibilidades agregadas, siempre que no haya citas programadas para ese horario.

Resultados Esperados

- Los estudiantes deben poder configurar y visualizar sus horarios de disponibilidad dentro del rango permitido.
- El sistema debe programar las citas para las horas específicas configuradas por el estudiante.
- Los estudiantes deben recibir una confirmación después de configurar su disponibilidad.
- Los estudiantes visualizarán los horarios configurados.
- Los estudiantes deben poder eliminar las disponibilidades agregadas siempre que no haya citas programadas para esos horarios.

Prueba # 7: Visualización de Pacientes Disponibles

Objetivo de la Prueba

Verificar que el sistema permita visualizar los pacientes disponibles para agendar citas, y que la función de agendamiento funcione correctamente una vez que el paciente haya sido clasificado.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Descripción de la Prueba

- Ingresar con un usuario tipo paciente, navegar al menú de pacientes.
- Verificar que solo los pacientes clasificados para el mismo curso del estudiante logueado estén disponibles para agendar citas.
- Confirmar que el sistema muestre el listado de pacientes disponibles y clasificados de acuerdo con la competencia del estudiante.

Resultados esperados

- El sistema debe permitir agendar citas a estudiantes solo con pacientes clasificados para su curso y que no hayan sido agendados.
- El sistema debe mostrar el listado de pacientes disponibles y clasificados según la competencia del estudiante.

Prueba # 7 Pruebas de agendamiento de citas odontológicas

Objetivo de la prueba

Validar si la función principal de asignación de citas odontológicas emite de forma coherente el registro de relacionamiento entre paciente y estudiante de acuerdo con la disponibilidad presentada en el software.

Descripción de la prueba

- Ingresar con un usuario tipo paciente, navegar al menú de pacientes
- Seleccionar un paciente de la lista
- Observar los horarios disponibles en la sesión de agendamiento de citas, de acuerdo con la selección del día.
- Validar que las notificaciones de citas agendadas se hagan efectivas.

Resultados esperados

- Una vez seleccionado el paciente, el estudiante podrá visualizar la información diligenciada en su clasificación.
- Los horarios que se presentan en el software deben de ser horarios, pero de disponibilidad por parte de los estudiantes.
- Las notificaciones de agendamiento ya sea por parte del estudiante o por el paciente deben de hacerse efectivas en los perfiles de cada involucrado en la cita.

La tabla 1, presenta los resultados de las pruebas de aceptación realizadas

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Tabla 5. Pasos pruebas de aceptación

Prueba #	Nombre de la Prueba	correo del usuario para prueba	Pasos Realizados	Resultados Esperados	Resultados Obtenidos	Comentarios
1	Registro a la Aplicación Oral Data	pepito@yopmail.com	<ol style="list-style-type: none"> Ingresar al sistema Completar el formulario de registro Confirmar el correo electrónico 	El sistema debe permitir el registro y enviar un correo de confirmación	El sistema permitió el registro y envió el correo de confirmación	Ninguno
2	Verificación de Acceso a la Aplicación	karencalle07@gmail.com	<ol style="list-style-type: none"> Ingresar con usuario y contraseña Verificar permisos del rol de paciente Verificar permisos del rol de estudiante Restablecer la contraseña 	Acceso correcto y verificación de permisos según el rol	Acceso correcto y permisos verificados	Ninguno
3	Proceso de Encuesta de Clasificación de Pacientes	pepito@yopmail.com	<ol style="list-style-type: none"> Completar la encuesta de clasificación Adjuntar 4 imágenes Guardar y verificar la información 	La encuesta debe ser completada y guardada correctamente, con validación de imágenes	La encuesta se completó y guardó correctamente, con validación de imágenes	Ninguno
4	Clasificación Docente de Pacientes	maria@yopmail.com	<ol style="list-style-type: none"> Visualizar listado de pacientes no clasificados Ingresar al detalle de la encuesta Seleccionar la clasificación en el drop-down-list 	El sistema debe permitir la clasificación y almacenar la información correctamente	La clasificación se realizó y almacenó correctamente	Ninguno

6	Gestión de Horarios de Disponibilidad	karencalle07@gmail.com	1. Configurar horarios de disponibilidad de miércoles 8am a 12pm 2. Guardar la configuración 3. Esperar mensaje la confirmación 4. Visualizar franjas establecidas	Los horarios deben configurarse correctamente y una notificación de confirmación debe ser enviada	Los horarios se configuraron correctamente y se envió la notificación de confirmación	Ninguno
7	Visualización de Pacientes Disponibles	karencalle07@gmail.com	1. Ingresar a la opción de pacientes 2. Verificar la lista de pacientes clasificados 3. Seleccionar un paciente y agendar una cita	Solo los pacientes clasificados deben aparecer y la cita debe ser agendada correctamente	Solo los pacientes clasificados aparecieron y la cita se agendó correctamente	Ninguno
5	Agendamiento de Citas Odontológicas	karencalle07@gmail.com	1. Seleccionar paciente clasificado 2. Seleccionar día y hora para la cita 3. Confirmar la cita 4. Verificar la notificación de la cita	La cita debe ser agendada y las notificaciones enviadas	La cita se agendó y las notificaciones fueron enviadas correctamente	Ninguno

Análisis de resultados

- En general, todas las funcionalidades probadas cumplieron con los criterios de aceptación definidos en las historias de usuario. El sistema demostró ser capaz de manejar las tareas esenciales requeridas por los usuarios.
- No se identificaron fallos críticos, lo que sugiere que el sistema está bien diseñado y ejecutado.
- Los usuarios encontraron las interfaces intuitivas y fáciles de usar.
- Aunque los resultados fueron positivos, se identificaron algunas áreas de mejoras de la usabilidad en ciertas funciones específicas.

Encuestas de satisfacción

Durante la ejecución de las pruebas, se realizaron a los usuarios las encuestas de satisfacción del software donde se tenían las siguientes preguntas cerradas

Tabla 6. Preguntas encuesta de satisfacción

Pregunta	Calificación de 1 a 5
¿Qué tan fácil le parece que es navegar por la aplicación?	
¿Qué tan bueno considera el diseño y la apariencia de la aplicación?	
¿Qué tan fácil fue registrar un nuevo paciente en la aplicación?	
¿Qué tan fácil fue agendar citas usando la aplicación?	
¿Qué tan fácil fue configurar sus horarios de disponibilidad?	
¿Qué tan útil le resulta la funcionalidad de visualizar pacientes disponibles?	
¿Qué tan satisfecho está con la precisión y la oportunidad de las notificaciones de la aplicación?	
¿Cómo calificaría el rendimiento general de la aplicación?	
¿Qué tan seguro se siente al usar la aplicación y proporcionar información sensible?	

Y las siguientes preguntas abiertas

- ¿Qué le gusta más de la aplicación Oral Data?
- ¿Qué le gusta menos de la aplicación Oral Data?
- ¿Hay alguna funcionalidad que le gustaría en la aplicación?
- ¿Tiene alguna sugerencia adicional para mejorar la aplicación?

Las respuestas se distribuyeron así:

Tabla 7. Resultados encuesta de satisfacción

Pregunta	Promedio respuestas
¿Qué tan fácil le parece que es navegar por la aplicación?	4.6
¿Qué tan bueno considera el diseño y la apariencia de la aplicación?	4.5/5.0

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

¿Qué tan fácil fue registrar un nuevo paciente en la aplicación?	5.0/5.0
¿Qué tan fácil fue agendar citas usando la aplicación?	5.0/5.0
¿Qué tan fácil fue configurar sus horarios de disponibilidad?	4.5/5.0
¿Qué tan útil le resulta la funcionalidad de visualizar pacientes disponibles?	4.8/5.0
¿Qué tan satisfecho está con la precisión y la oportunidad de las notificaciones de la aplicación?	5.0/5.0
¿Cómo calificaría el rendimiento general de la aplicación?	5.0/5.0
¿Qué tan seguro se siente al usar la aplicación y proporcionar información sensible?	5.0/5.0

Los resultados de la encuesta de satisfacción indican un alto nivel de satisfacción entre los usuarios con respecto a la mayoría de los aspectos evaluados. La facilidad de navegación, la estética de la aplicación, la eficiencia en la gestión de pacientes y citas, la configuración de horarios de disponibilidad, la utilidad de visualizar pacientes disponibles, la precisión de las notificaciones, el rendimiento general del sistema y la percepción de seguridad son todos puntos fuertes destacados por los usuarios. Estos resultados no solo validan el enfoque y las decisiones de diseño adoptadas durante el desarrollo del sistema, sino que también proporcionan una base sólida para futuras mejoras y expansiones del sistema.

Además, las preguntas abiertas, mostraron la alta satisfacción de los usuarios donde les gustó mucho que la aplicación permitiera cancelar citas con anticipación de 24 horas mínimo y que el sistema enviara notificaciones al instante de las citas agendadas.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

5. CONCLUSIONES, RECOMENDACIONES Y TRABAJO FUTURO

Conclusiones

- La aplicación de la metodología ágil para el proyecto permitió un desarrollo iterativo y colaborativo en la implementación del sistema, haciendo posible la adaptabilidad a medida que surgen los cambios o ajustes en los requerimientos. A pesar de que el equipo de desarrollo consistía en solo una persona, la metodología permitió un marco estructurado que tenga en cuenta la planificación, ejecución y revisión de tareas de manera eficiente. Teniendo en cuenta que la comunicación con los stakeholders fue crucial para que el proyecto se alineara con sus expectativas.
- El desarrollo del sistema se caracterizó por adoptar un enfoque integral, abarcando todos los aspectos críticos para la construcción de software, desde el levantamiento meticuloso de requisitos hasta las pruebas finales. Este proceso comenzó con un levantamiento exhaustivo de requisitos, seguido de una fase de validación con los usuarios para asegurar un entendimiento claro de sus necesidades, útil para construir las historias de usuario funcionales y no funcionales y priorizar los requisitos. Este enfoque usado para el levantamiento y priorización de requisitos, junto con la elaboración de historias de usuario, sentó las bases sólidas para el diseño de una arquitectura robusta que guiaría todo el desarrollo posterior.
- La implementación de pruebas de aceptación jugó un papel muy importante en el proyecto, debido a que permitieron validar que el sistema no solo cumplía con las historias de usuario y sus criterios de aceptación específicos, sino que también satisfacía plenamente los requisitos técnicos necesarios para atender las necesidades reales y expectativas de los usuarios. Este proceso de validación fue fundamental para asegurar la calidad y fiabilidad del producto, demostrando que el sistema podía operar de manera fluida y eficiente bajo condiciones reales. Al confirmar que el sistema cumplía con todos los criterios de aceptación, se garantizó que el producto final no solo cumplía con los requisitos funcionales, sino que también ofrecía una experiencia de usuario superior, lo que fue esencial para el éxito del proyecto.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Recomendaciones

- Es importante prestar atención al rendimiento del sistema, especialmente en momentos en que se espera un alto flujo de personas usando la aplicación, esto implica implementar medidas que garanticen la estabilidad del sistema durante largos periodos de altos picos de manera que el sistema soporte eficientemente las demandas y reduzca el riesgo de colapsos.
- Es importante explorar la posibilidad de incorporar nuevas funcionalidades que no se contemplaron inicialmente durante el levantamiento de requisitos. Funcionalidades como un plan de tratamiento personalizado para los pacientes y un sistema de recordatorios de citas pueden ofrecer un valor significativo agregado, mejorando la experiencia del paciente. La inclusión de estas características no solo mejora la funcionalidad general del sistema, sino que también permite a los usuarios aprovechar al máximo las capacidades del software, contribuyendo a una mayor satisfacción y retención de usuarios.
- Se deben incluir sesiones de capacitación sobre el sistema a los usuarios como estudiantes, profesores y administradores del sistema. Estas sesiones deben cubrir en profundidad todas las funcionalidades del sistema, proporcionando a los usuarios las herramientas necesarias para navegar y utilizar el sistema de manera efectiva. La capacitación adecuada es fundamental para asegurar que los usuarios comprendan completamente las capacidades del sistema y puedan aprovecharlas al máximo.

Trabajo futuro

- La implementación de las historias clínicas y plan de tratamiento de los pacientes son funcionalidades que proporcionarían una solución más completa y que representarían un paso significativo hacia la creación de una solución más integral y completa. Estas funcionalidades no solo mejoran la capacidad del sistema para gestionar la información de los pacientes de manera más precisa y organizada, sino que también ofrecen a los estudiantes herramientas valiosas para tomar decisiones informadas y personalizadas.
- Explorar la viabilidad de aplicar conceptos avanzados de inteligencia artificial y aprendizaje automático al sistema para la automatización de la clasificación de pacientes basada en la información recopilada, de manera que se pueda optimizar el proceso de direccionamiento para una correcta atención de los pacientes

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

REFERENCIAS

- Software Dentalink. (2023). Recuperado el 23 de octubre de 2023, de <https://www.softwaredentalink.com>
- My Dental Clinic. (2023). Recuperado el 23 de octubre de 2023, de <https://web.dentalclinicapp.com/login>
- Arevalo, F. (2015). Gestión de pacientes en facultades de odontología: Una revisión de literatura. Revista de Odontología de la Universidad de Costa Rica, 27(2), 115-124.
- OPS/OMS Chile. (2009). Tecnologías de la información en salud para la atención primaria: Guía para la toma de decisiones. Santiago de Chile: OPS/OMS.
- Rojas, R., & Galvis, G. (2023). El impacto de la tecnología en la educación dental. Journal of Dental Education, 45(3), 150-160
- DefiniciónABC. (2023). Definición de Gestión de Calidad. Recuperado el 23 de octubre de 2023, de <https://www.definicionabc.com/economia/gestion-de-calidad.php>
- Universidad de Santiago de Compostela. (2023). Prácticas de la Facultad de Medicina y Odontología. Recuperado el 23 de octubre de 2023, de <https://www.usc.gal/es/centro/facultad-medicina-odontologia/practicas>
- Entity Framework Tutorial. (2023). What is Entity Framework? Recuperado el 23 de octubre de 2023, de <https://www.entityframeworktutorial.net/what-is-entityframework.aspx>
- Blazor University. (2023). What is Blazor? Recuperado el 23 de octubre de 2023, de <https://blazor-university.com/overview/what-is-blazor/>

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

- Microsoft. (2023). Introduction to Azure Blob Storage. Recuperado el 23 de octubre de 2023, de <https://learn.microsoft.com/en-us/azure/storage/blobs/storage-blobs-introduction>
- Ausubel, D. P., Novak, J. D., & Hanesian, H. (1978). Aprendizaje y desarrollo humano. Holt, Rinehart and Winston.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

ANEXOS

Los anexos deben ser nombrados con letras para diferenciarse unos de otros (p. ej: Anexo A, Anexo B, etc.). Estos hacen extensiva la información del contenido del trabajo realizado tales como cálculos matemáticos extensos, códigos de programación, etc. El contenido de los apéndices debe permitir a alguien externo al desarrollo del trabajo, llegar a los mismos resultados siguiendo la misma metodología complementada con la información que en este aparte reposa.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

FIRMA ESTUDIANTES

Karen m Calle

FIRMA ASESORES

FECHA ENTREGA: _____