



Aplicación móvil para la búsqueda de médicos y agendamiento de consultas médicas

Autor: Elivar Largo

TFM-DESARROLLO DE APLICACIONES SOBRE DISPOSITIVOS MÓVILES

Máster Universitario en Ingeniería Informática

Consultores: - Jordi Almirall López
- Jordi Ceballos Villach

Enero 2021



Esta obra está bajo una Licencia Creative Commons Atribución-NoComercial-SinDerivadas 4.0 Internacional.



FICHA DEL TRABAJO FINAL

Título del trabajo:	Aplicación móvil para la búsqueda de médicos y agendamiento de consultas médicas
Nombre de la app:	helpdoctor
Nombre del autor:	Elivar Largo
Nombre del consultor/a:	- Jordi Almirall López - Jordi Ceballos Villach
Fecha de entrega:	Enero 2021
Titulación:	Máster Universitario en Ingeniería Informática
Área del Trabajo Final:	Desarrollo de aplicaciones sobre dispositivos móviles
Idioma del trabajo:	Español
Palabras clave:	Consultas Médicas, Flutter, Aplicación Móvil Médicos
Resumen del Trabajo (máximo 250 palabras):	
<p>Actualmente todos usamos un teléfono móvil más allá de únicamente hacer o recibir llamadas, gracias a internet y a la conectividad, este tipo de dispositivos se ha vuelto parte indispensable de nuestras vidas, con lo que el uso de aplicaciones móviles básicamente ha hecho que casi todo esté al alcance nuestro.</p> <p>En ese sentido el objetivo principal de este proyecto es obtener una aplicación que permita buscar médicos y poder agendar una consulta médica sin la necesidad de acudir físicamente o hacerlo a través de una llamada telefónica o simplemente alargar el proceso mediante búsquedas en internet o redes sociales como Facebook.</p> <p>Para el diseño del proyecto y con la finalidad de obtener un producto acorde a las necesidades de los usuarios finales se ha seguido la metodología de Diseño Centrado en el Usuario o DCU con sus respectivas fases: Usuarios y Contexto, Diseño conceptual, Prototipado y Evaluación.</p> <p>En cuanto a la implementación tratando de seguir una metodología como SCRUM en la que de la misma forma se trata de entregas parciales funcionales con valor para el usuario final, en cuanto a la aplicación como tal se la ha desarrollado usando Flutter y Spring Boot con una base de datos MySQL para la parte del backend.</p> <p>Finalmente se tiene una app llamada HelpDoctor funcional cumpliendo con los objetivos planteados, así mismo una experiencia en lo profesional muy gratificante ya que se ha puesto en prácticamente todo un proceso de desarrollo.</p>	
Abstract (in English, 250 words or less):	
Currently we all use a mobile phone beyond just making or receiving calls, thanks to the internet and connectivity, this type of device has become an indispensable part of our lives, with which the use of mobile applications has basically made almost everything within our reach.	

In this sense, the main objective of this project is to obtain an application that allows searching for doctors and being able to schedule a medical consultation without the need to physically go or do it through a phone call or simply lengthen the process through internet searches or social networks such as Facebook.

For the design of the project and in order to obtain a product according to the needs of the end users, the methodology of User Centered Design or DCU has been followed with its respective phases: Users and Context, Conceptual Design, Prototyping and Evaluation.

Regarding the implementation, trying to follow a methodology such as SCRUM in which in the same way it deals with functional partial deliveries with value for the end user, as for the application as such it has been developed using Flutter and Spring Boot with a MySQL database for the backend part.

Finally, there is an app called HelpDoctor that is functional, fulfilling the objectives set, as well as a very rewarding professional experience since it has been put into practically an entire development process.

Índice

Capítulo 1- Introducción	1
1.2 Objetivos.....	1
1.2.1 Objetivo General:	1
1.2.2 Objetivos Específicos:	1
1.3 Enfoque y método elegido.....	2
1.4 Planificación del Trabajo.....	3
1.4.1 Recursos:	3
1.4.2 Tareas.....	3
1.5 Costos	4
1.6 Productos Obtenidos	4
1.7 Capítulos	4
Capítulo 2- Diseño y Arquitectura	6
2.1 Justificación de métodos de indagación.....	6
2.1.1 Entrevista	6
2.1.2 Análisis competitivo.....	6
2.2 Formato de entrevista	7
2.2.1 Desarrollo de entrevistas	8
2.2.2 Resultados y conclusiones	8
2.3 Usuarios y contexto de uso.....	8
2.3.1 Perfiles	8
2.4 Diseño Conceptual	9
2.4.1 Escenarios de uso	9
2.4.2 Flujos de interacción.....	14
2.5 Prototipado.....	16
2.5.1 Sketches	16
2.5.2 Prototipos	18
2.5.2.1 Pantalla de Ingreso	18
2.5.2.2 Pantalla de Registro	19
2.5.2.3 Menú de usuario.....	19
2.5.2.4 Menú de médico.....	20
2.5.2.5 Pantalla principal de usuario.....	20
2.5.2.6 Pantalla de búsqueda de médicos	21
2.5.2.7 Pantalla para agendar cita médica.....	21

2.5.2.8 Pantalla de reserva	22
2.5.2.9 Pantalla de citas agendadas	22
2.5.2.10 Pantalla de registro/actualización datos de usuario.....	23
2.5.2.11 Pantalla de registro/actualización de datos médico.....	23
2.5.2.12 Pantalla de citas agendadas médico	24
2.6 Evaluación	24
2.6.1 Preguntas al usuario	24
2.6.1.1 Tareas a realizar.....	25
2.6.1.2 Preguntas sobre las tareas	25
2.7 Definición de Casos de Uso.....	25
2.7.1 Detalle de casos de uso	26
2.8 Diseño y arquitectura	33
2.8.1 Diagrama de Clases.....	33
2.8.2 Diagrama Entidad Relación	34
2.8.3 Arquitectura	35
Capítulo 3- Implementación	36
3.1 Flutter	36
3.1.1 Arquitectura de Flutter	36
3.1.1.1 Framework	37
3.1.1.2 Engine.....	37
3.1.1.3 Embebder	37
3.2 MySQL.....	37
3.3 Spring y Spring Boot	37
3.3.1 Arquitectura de Spring.....	38
3.3.1.1 Web.....	38
3.3.1.2 Data Access.....	38
3.3.1.2 Core Container.....	38
3.4 Android Studio	38
3.4 Git y GitHub.....	39
3.4.1 Git	39
3.4.2 GitHub.....	39
3.5. Postman.....	39
3.6. Maven.....	40
3.7 Estructura del proyecto	40

3.7.1. Estructura Backend.....	40
3.7.1.1. Paquete com.helpdoctor.controller	40
3.7.1.2. Paquete com.helpdoctor.dao.....	41
3.7.1.3. Paquete com.helpdoctor.entity.....	41
3.7.1.4. Paquete com.helpdoctor.exception	41
3.7.1.5. Paquete com.helpdoctor.security	41
3.7.1.6. Paquete com.helpdoctor.service.....	41
3.7.1.6. Recursos disponibles.....	42
3.7.2. Estructura Frontend.....	43
3.7.1.1. Paquete src.model	44
3.7.1.2. Paquete src.page	44
3.7.1.3. Paquete src.service	45
3.7.1.2. Paquete src.util.....	45
3.8. Estado del proyecto.....	45
3.9. Pruebas	46
3.9.1 Pruebas de Backend	46
3.9.2 Pruebas de Frontend	51
3.9.3. Imágenes de la aplicación en ejecución.....	52
3.9.3.1 Login de usuario.....	52
3.9.3.2 Pantalla principal del usuario	53
3.9.3.3 Menú del usuario y actualización de datos	54
3.9.3.4 Ver consultas agendadas.....	55
3.9.3.4 Calificar a un médico.....	56
3.9.3.5 Búsqueda por especialidad	56
3.9.3.6 Menú del médico y actualización de datos	57
3.9.3.7 Registro de usuarios.....	58
Capítulo 4- Conclusiones	59
4.1 Líneas de trabajo futuro	59
5. Código Fuente	60

Índice de Tablas

Tabla 1: Tareas del proyecto	3
Tabla 2: Costos estimados.....	4
Tabla 3: Formato de entrevista	8
Tabla 4: Resultados fase de indagación	9
Tabla 5: Escenario 1	10
Tabla 6: Escenario 2.....	12
Tabla 7: Escenario 3.....	13
Tabla 8: Caso de uso: Registro de usuario	26
Tabla 9: Caso de uso: Ingresar a la aplicación	27
Tabla 10: Caso de uso: Buscar un médico	27
Tabla 11: Caso de uso: Agendar una cita medica	28
Tabla 12: Caso de uso: Ver citas médicas	28
Tabla 13: Caso de uso: Actualizar datos	29
Tabla 14 Caso de uso: Cancelar una cita por el paciente.....	29
Tabla 15: Caso de uso: Chatear con un médico	30
Tabla 16: Caso de uso: Publicar perfil	30
Tabla 17: Caso de uso: Ver citas médicas	31
Tabla 18: Caso de uso: Cancelar cita por un médico	32
Tabla 19: Caso de uso: Salir de la aplicación	32

Índice de Ilustraciones

Ilustración 1. Flujo para reservar una cita médica	14
Ilustración 2. Flujo para registrar información por el usuario	15
Ilustración 3. Flujo para registrar información por el médico.....	16
Ilustración 4. Pantalla de ingreso.....	18
Ilustración 5. Pantalla de registro	19
Ilustración 6. Menú de usuario.....	19
Ilustración 7. Menú de médico	20
Ilustración 8. Pantalla lista de médicos	20
Ilustración 9. Pantalla de búsqueda de un médico	21
Ilustración 10. Pantalla agendar cita	21
Ilustración 11. Pantalla reserva.....	22
Ilustración 12. Pantalla citas agendadas.....	22
Ilustración 13. Pantalla datos de usuario	23
Ilustración 14. Pantalla datos de médico	23
Ilustración 15. Pantalla citas agendadas médico	24
Ilustración 16. Diagrama de Casos de Uso	25
Ilustración 17: Diagrama de Clases.....	33
Ilustración 18: Diagrama entidad relación.....	34
Ilustración 19: Arquitectura de la aplicación	35

Capítulo 1- Introducción

1.1 Contexto y justificación del Trabajo

Actualmente el uso del móvil se ha vuelto más indispensable que nunca, se dice que por ejemplo una persona dedica a usar el teléfono móvil ya sea en redes sociales, aplicaciones de música, aplicaciones de compras en línea entre otras es de aproximadamente 61 horas a la semana [1].

Por otra parte, en Ecuador operan unan gran cantidad de aplicaciones enfocadas a la compra y venta de artículos nuevos y usados, de búsqueda de trabajo, de movilidad y en estos últimos meses a raíz de la COVID-19 el repunte se ha dado en aplicaciones que tienen que ver con compras en línea tipo ecommerce y delivery las mismas que ha sido desarrolladas a pedido de las mismos dueños de emprendimientos y otras ofrecidas por peques startups que también surgieron por el tema de la COVID-19, sin embargo, en el tema médico actualmente no tenemos opciones, existen páginas web como: que ofrecen publicidad para médicos y agendamiento de citas médicas, sin embargo, no cuentan con una aplicación móvil, así mismo la mayoría de médicos cuenta con su página de Facebook, que si bien cierto permite a los usuarios encontrar médicos, es compleja la forma de hacerlo.

En ese contexto se propone el presente proyecto: Aplicación móvil para la búsqueda de médicos y agendamiento de consultas médicas.

La idea es desarrollar una aplicación que permita ejecutarse bajo un entorno multiplataforma y que permita tener una lista médicos, de tal manera que el usuario pueda buscar y acceder a una consulta médica sin tener que hacerlo en cada página del médico, además el usuario podrá tener a su alcance la información para consultar sus citas médicas.

1.2 Objetivos

1.2.1 Objetivo General:

Obtener una aplicación móvil multiplataforma que permita la búsqueda de médicos y agendamiento de consultas médicas.

1.2.2 Objetivos Específicos:

Requerimientos funcionales:

- ✓ **Registrar usuarios:** La aplicación permitirá el registro de usuarios tanto médicos como pacientes.
- ✓ **Perfiles y funcionalidades:** La aplicación permitirá crear perfiles y funcionalidades diferentes para cada uno.
- ✓ **Perfil de usuario paciente:** La aplicación permitirá crear y gestionar un perfil para un usuario paciente.

- ✓ **Perfil de usuario medico:** La aplicación permitirá crear y gestionar un perfil para un usuario médico en el que se incluirá información acerca de su especialidad, foto y los servicios que ofrece.
- ✓ **Configurar horarios de atención:** La aplicación permitirá a los usuarios médicos generar horarios de atención por semana.
- ✓ **Mostrar las consultas agendadas:** La aplicación permitirá a los usuarios médicos ver las consultas médicas agendadas por un paciente.
- ✓ **Buscar médicos:** La aplicación permitirá a los usuarios pacientes buscar médicos y ver su perfil.
- ✓ **Agendar una consulta médica:** La aplicación permitirá a los usuarios pacientes registrados agendar una consulta.
- ✓ **Mostrar las consultas agendadas:** La aplicación permitirá a los usuarios pacientes registrados mostrar las consultas agendadas.
- ✓ **Cancelar una consulta médica:** La aplicación permitirá a los usuarios pacientes registrados cancelar consultas agendadas con un tiempo máximo de 24 horas anteriores a la consulta.
- ✓ **Notificaciones:** La aplicación enviará a los usuarios pacientes aviso sobre su consulta médica 24 horas previo la consulta agendada.

Requerimientos no funcionales:

- ✓ **Usabilidad:** La aplicación deberá estar diseñada de tal forma que su uso no requiera usuarios expertos.
- ✓ **Desarrollo:** La aplicación se desarrollará usando el Framework Flutter [2] de tal manera que se pueda usar en un entorno multiplataforma.
- ✓ **Seguridad:** La aplicación deberá brindar seguridad en el acceso al usuario usando una validación de doble factor.

1.3 Enfoque y método elegido

En cuanto a la estrategia para llevar a cabo el proyecto, se empezará con un desarrollo desde cero, básicamente se ha seleccionado esta opción debido a que por lo general las aplicaciones que si bien es cierto ya existen y que bien se podrían modificar adaptándola al sector al que nos dirigimos, todas son de dominio de privado.

Por lo que, como se menciona la aplicación será desarrollada desde cero, basándonos en las funcionalidades generales de otras aplicaciones ya existentes, como, por ejemplo: aplicaciones de compra y venta, aplicaciones de delivery, de movilidad y ecommerce en general.

De la misma forma para llevar a cabo el proyecto y sus entregables nos basaremos en metodologías ágiles de desarrollo, de tal manera que podamos

adaptarnos a los cambios fácilmente haciendo entregables funcionales y en periodos cortos tiempo.

1.4 Planificación del Trabajo

1.4.1 Recursos:

- ✓ SDK Flutter
- ✓ Android Studio / Visual Code
- ✓ Software para diseño de interfaces
- ✓ Procesador de Texto
- ✓ Portátil con sistema operativo Windows 10 Home
- ✓ Teléfono móvil con sistema operativo Android
- ✓ Emulador para sistema operativo IOS

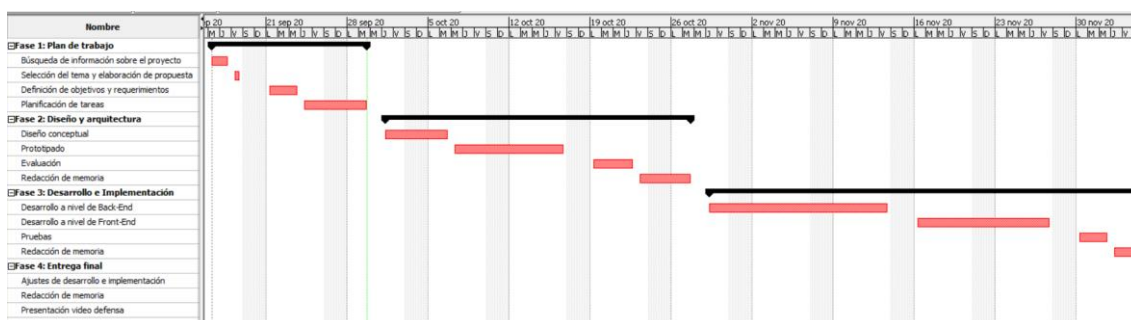
1.4.2 Tareas

TAREAS	DURACIÓN	INICIO	FIN
Fase 1: Plan de trabajo	14	16/09/2020	30/09/2020
Búsqueda de información sobre el proyecto	5	16/09/2020	30/09/2020
Selección del tema y elaboración de propuesta	2	16/09/2020	30/09/2020
Definición de objetivos y requerimientos	4	16/09/2020	30/09/2020
Planificación de tareas	3	16/09/2020	30/09/2020
Fase 2: Diseño y arquitectura	87,5	01/10/2020	28/10/2020
Diseño conceptual	20	01/10/2020	28/10/2020
Prototipado	30	01/10/2020	28/10/2020
Evaluación	20	01/10/2020	28/10/2020
Redacción de memoria	17,5	01/10/2020	28/10/2020
Fase 3: Desarrollo e Implementación	132,5	29/10/2020	09/12/2020
Desarrollo a nivel de Back-End	50	29/10/2020	09/12/2020
Desarrollo a nivel de Front-End	50	29/10/2020	09/12/2020
Pruebas	12,5	29/10/2020	09/12/2020
Redacción de memoria	20	29/10/2020	09/12/2020
Fase 4: Entrega final	57,5	10/12/2020	06/01/2020
Ajustes de desarrollo e implementación	37,5	10/12/2020	06/01/2020
Redacción de memoria	10	10/12/2020	06/01/2020
Presentación video defensa	10	10/12/2020	06/01/2020
Total horas:	291.5		

Tabla 1: Tareas del proyecto

Para la culminación del proyecto se plantea un esfuerzo total de 291.5 horas distribuidas en días laborables 2.5 horas por día, mientras que en fin de semana se pretende trabajar 5 horas por día, en esta planificación no se toma en cuenta los días festivos como por ejemplo navidad, fin de año y año nuevo, ya que en esos días se asume que no se va trabajar.

Tareas en un Diagrama de Gantt:



1.5 Costos

A continuación, se realiza un estimado en cuanto al presupuesto necesario para llevar a cabo el proyecto, para esto se ha tomado únicamente en cuenta las fases y tareas mostradas en la planificación, así mismo el cálculo para el costo hora/programador se lo ha realizado en promedio lo que se paga a un programador en Latinoamérica:

Fases	Horas	Costo Hora \$	Precio \$
Fase 1: Plan de trabajo	14	15	210
Fase 2: Diseño y arquitectura	87,5	15	1312,5
Fase 3: Desarrollo e Implementación	132,5	15	1987,5
Fase 4: Entrega final	57,5	15	862,5
Tota Horas/Costo en dólares	291,5	15	4372,5

Tabla 2: Costos estimados

1.6 Productos Obtenidos

- ✓ Aplicación instalable y código fuente
- ✓ Memoria
- ✓ Presentación en video

1.7 Capítulos

Introducción: En este capítulo definimos el contexto, justificación del tema y el plan de trabajo a llevar a cabo para el proyecto.

Diseño y Arquitectura: En este capítulo llevaremos a cabo el análisis de requerimientos tanto funcionales como no funcionales, el proceso para aplicar el método de Diseño Centrado en el Usuario y finalmente diseñar la arquitectura de la aplicación.

Desarrollo en Implementación: En este capítulo se pretende exponer un estudio del arte del SDK de Flutter, la estructura de la aplicación y explicación de lo más importante y la documentación sobre las pruebas de validación.

Conclusiones: En esta parte se pretende exponer a forma de resumen los obstáculos, logros y todo lo aprendido en el desarrollo del proyecto.

Referencias Bibliográficas: En esta parte se documentará toda la bibliografía usada para la realización del proyecto.

Capítulo 2- Diseño y Arquitectura

Con la finalidad de obtener una aplicación con funcionalidades acorde a las necesidades de los usuarios, en esta fase del proyecto vamos a usar el proceso de Diseño Centrado en el Usuario o conocido simplemente como DCU, básicamente este proceso se basa en 4 fases:

1. Usuarios y contexto de uso
2. Diseño conceptual
3. Prototipado
4. Evaluación

2.1 Justificación de métodos de indagación

El proceso DCU en todas sus fases del diseño de un producto se enfoca en el usuario por lo que en esta primera fase vamos a recoger información acerca del usuario y el contexto basándonos en dos técnicas: Entrevista y Análisis competitivo.

Básicamente las preguntas que queremos responder en esta primera fase son las siguientes:

1. ¿Quiénes usarán la aplicación?
2. ¿Cuándo usarán la aplicación??
3. ¿En dónde usarán la aplicación?
4. ¿Obtener las funcionalidades de la aplicación?

Basados en los métodos de indagación [3], para esta fase vamos a usar dos métodos o técnicas que nos permitirán acercarnos un poco más a las necesidades del usuario.

2.1.1 Entrevista

Se usará esta técnica ya que nos permitirá abordar directamente al usuario y entender más a profundidad sus necesidades, su experiencia en el uso de dispositivos móviles y opiniones, con lo que nos facilitará elaborar el perfil de usuario y las funcionalidades de la aplicación.

2.1.2 Análisis competitivo.

Se usará esta técnica ya que en el mercado existen otras aplicaciones ya desarrolladas que no pertenecen al mismo sector, sin embargo, existen funcionalidades que ya son conocidas por los usuarios y las podemos valorar y adaptar o mejorar en nuestra aplicación, para este caso obtendremos las funcionalidades que se puedan adaptar de las aplicaciones de pedidos de comida Zaymi [4] y servicio de taxi Ktaxi[].

2.2 Formato de entrevista

1. Obtener la información personal del usuario

- ✓ Edad
- ✓ Sexo
- ✓ Ocupación
- ✓ Estado Civil
- ✓ Ciudad:
- ✓ Hijos :

2. Conocer la experiencia que tiene el usuario manejando aplicaciones móviles

- ✓ ¿Tienes aplicaciones móviles instaladas en tu celular?
- ✓ ¿Qué tipos de aplicaciones son las que más usas?
- ✓ ¿En qué lugar usas más tu teléfono celular para revisar aplicaciones?
- ✓ ¿Qué sistema operativo usa tu celular?

3. Conocer el problema del usuario en cuanto a buscar un doctor particular y al agendar una cita médica

- ✓ ¿Con qué frecuencia acudes a un médico particular y porque razones has tenido que acudir?
- ✓ ¿Tienes algún familiar que acude a un médico particular, sabes con qué frecuencia lo hace, sabes si ha tenido problemas al buscar un médico y agendar una consulta?
- ✓ ¿Cómo haces para buscar y agendar una cita con un médico particular?
- ✓ ¿Cuándo has buscado atención medica particular, buscas un médico especialista o un médico general?
- ✓ ¿Has tenido algún problema al buscar un médico particular y agendar una cita, cómo crees que podría solucionar esos problemas en caso de haberlos?
- ✓ ¿Has usado alguna aplicación móvil o web para obtener una cita médica, comenta cual ha sido tu experiencia con ella, que es lo que más te ha gustado o disgustado, cómo crees que podría solucionar esos problemas en caso de haberlos?
- ✓ ¿Solicitarías un doctor a través de una aplicación móvil, crees que sería una forma eficiente para poder encontrar un médico y agendar una cita?
- ✓ ¿Qué tareas o funcionalidades crees que debería tener la aplicación?

Tabla 3: Formato de entrevista

2.2.1 Desarrollo de entrevistas

Se ha realizado entrevistas a 8 personas amigos y conocidos de amigos, la (Ver entrevistas en anexo 2)

2.2.2 Resultados y conclusiones

De acuerdo a las entrevistas realizadas podemos identificar que las personas que más acuden al doctor son aquellas personas de entre 28 y 40 años con ingresos fijos mensuales que por lo general tienen un trabajo estable y que serían las que posiblemente usen la aplicación, este perfil de personas maneja perfectamente y está acostumbrado al uso y manejo de aplicaciones, por lo genera el uso lo hacen en casa, aunque que si generalizamos un poco lo hacen en cualquier lugar donde puedan tener conexión a internet, en cuanto a sus necesidades de buscar un médico se puede evidenciar que la mayoría acude al médico mínimo 2 veces por año y las personas que están siguiendo algún tratamiento lo hacen cada 3 meses y la mayoría busca médicos especialistas, por otra parte los principales problemas al agendar una cita médica es no tener el contacto del médico, ni la ubicación del lugar de trabajo, no saber el precio de la consulta ni los horarios disponibles, no tener una referencia de sus estudios o valoraciones por parte de los pacientes y que les permita tomar la mejor decisión, finalmente otro punto importante es que un 90% de los entrevistados maneja un teléfono móvil con sistema operativo Android.

2.3 Usuarios y contexto de uso

Una vez realizadas las entrevistas se ha podido identificar que existe un perfil general de personas de entre 28 y 40 años que de alguna u otra manera han agendado una cita médica una o dos veces por año, sin embargo, si vamos un poco más a detalle podríamos decir que hay un subperfil que engloba las personas que llevan algún tipo de tratamientos y que son las que más recurren al médico y que posiblemente usarían más la aplicación.

2.3.1 Perfiles

A continuación, se elabora un perfil de usuario en base a la información obtenida en las entrevistas realizadas.

Perfil	Usuario
	<ul style="list-style-type: none"> ✓ Personas con una edad de entre 28 y 40 años ✓ Padres con hijos menores a 5 años

Características	<ul style="list-style-type: none"> ✓ Personas que llevan tratamientos en su salud o que quieren hacer revisiones de rutina o chequeos generales. ✓ Motivación: Conocer el estado de su salud, tratar alguna dolencia o mejorar su salud a través de un tratamiento médico
Contexto de uso	<ul style="list-style-type: none"> ✓ Usan el celular a diario, la mayor parte del tiempo en sus casas o en lugares donde haya conexión.
Tareas	<ol style="list-style-type: none"> 1. Buscar un médico 2. Visualizar la ficha del medico 3. Registro y autenticación de usuario 4. Agendar cita médica 5. Consultar citas medicas 6. Calificar médico 7. Modificar cuenta de usuario
Lista de características	<ul style="list-style-type: none"> ✓ Que tenga facilidad de búsqueda y manejo ✓ Que muestre información relevante de un medico ✓ Que tenga una interfaz amigable ✓ Que permita busca un médico ✓ Que tenga varios médicos especialistas ✓ Que permita calificar a un médico ✓ Que permita agendar una cita médica ✓ Que permita el pago en línea ✓ Que muestre el costo de la cita médica

Tabla 4: Resultados fase de indagación

2.4 Diseño Conceptual

En esta fase vamos a diseñar la aplicación basados en el perfil de usuario creado anteriormente, para esto vamos a considerar 2 partes importantes: los escenarios de uso y los flujos de interacción.

2.4.1 Escenarios de uso

Vamos a describir diferentes escenarios en el que nuestro posible usuario usaría la aplicación, esto nos permitirá tener una mayor referencia y definir

correctamente o lo más apegado a las necesidades del usuario el diseño de la aplicación.

Escenario 1	
Perfil de usuario	Usuario Paciente
Personaje	Jhonny Padilla tiene 35 años es casado cuenta con un trabajo estable y una remuneración de aproximadamente \$500 su esposa también trabaja y tiene una hija de 8 años. Maneja algunas aplicaciones en su teléfono, redes sociales, compras en línea y otras.
Descripción	Su papá sufre de artritis y está buscando un médico que le ayude con el tratamiento, por lo que está pidiendo consejos de amigos sobre algún especialista, también está revisando algunos directorios médicos a ver si encuentra un especialista.
Contexto	En su casa en un día común y corriente
Objetivos	Buscar un médico especialista
Tareas	Búsqueda de un médico Registro y autenticación del usuario Visualizar la ficha del medico
Necesidades de información	Descargar e instalar la aplicación Registro / autenticación del usuario
Funcionalidades necesarias	Registrar Usuario Autenticación de usuario Buscar un médico por especialidad Mostrar el nombre del médico junto a su imagen Mostrar la ficha del medico
Desarrollo de tareas	Registro / autenticación de usuario, buscar el médico

Tabla 5: Escenario 1

Escenario 2	
Perfil de usuario	Usuario Paciente
Personaje	<p>Lis Angamarca tiene 32 años es casada y tiene 2 hijos de 10 años y 1 año 7 meses,</p> <p>Trabaja en una entidad bancaria y su esposo es profesor en un colegio particular, los ingresos mensuales entre ambos suma \$ 1800.</p> <p>Ella maneja algunas aplicaciones en su teléfono, redes sociales, compras en línea y otras, bancarias y educativas.</p>
Descripción	Ella lleva un tratamiento para controlar la alergia que padece, ella por lo general va al médico unas 7 veces al año, ella agenda por medio de llamada telefónica una cita medica
Contexto	En el trabajo ya que cuenta con conexión a internet.
Objetivos	Agendar una cita médica con un especialista
Tareas	<p>Registro y autenticación del usuario</p> <p>Buscar un médico</p> <p>Agendar cita médica</p> <p>Calificar médico</p>
Necesidades de información	<p>Descargar e instalar la aplicación</p> <p>Registro / autenticación del usuario</p>
Funcionalidades necesarias	<ul style="list-style-type: none"> ✓ Registrar Usuario ✓ Autenticación de usuario ✓ Buscar un médico por especialidad ✓ Mostrar el nombre del médico junto a su imagen ✓ Mostrar la ficha del medico ✓ Mostrar hora y día para agendar ✓ Agendar la cita médica
Desarrollo de tareas	Registro / autenticación de usuario, buscar el médico y agendar la cita

Tabla 6: Escenario 2

Escenario 3	
Perfil de usuario	Usuario Médico
Personaje	<p>Verónica Cevallos tiene 28 años es un médico especialista, con estudios dentro y fuera del Ecuador, es soltera y para publicidad maneja su página de Facebook, adicionalmente paga publicidad en periódicos de la localidad.</p> <p>Ella maneja algunas aplicaciones en su teléfono, redes sociales, compras en línea y no tiene problemas con la tecnología.</p>
Descripción	<p>Actualmente ella tiene un consultorio propio y atiende varios pacientes, sin embargo, cree que sería bueno publicar su perfil en un directorio medico de una aplicación móvil, ya que le ayudaría a llegar a más personas y evitar pagar publicidad adicional, además que le va permitir agendar directamente citas con sus pacientes.</p>
Contexto	En su consultorio
Objetivos	Publicar su perfil en la aplicación móvil
Tareas	<p>Registro y autenticación del usuario</p> <p>Publicar su perfil como médico</p>
Necesidades de información	<p>Descargar e instalar la aplicación</p> <p>Registro / autenticación del usuario</p>
Funcionalidades necesarias	<ul style="list-style-type: none"> ✓ Registrar Usuario ✓ Autenticación de usuario ✓ Registrar información básica ✓ Registrar información consultorio ✓ Registrar información estudios ✓ Registrar información especialidad ✓ Registrar información horario atención y precio cita médica

	✓ Registrar imagen
Desarrollo de tareas	Registro / autenticación de usuario, registro de información como médico

Tabla 7: Escenario 3

2.4.2 Flujos de interacción

Los flujos de interacción son una forma de plasmar las decisiones y el camino que un usuario toma para cumplir los objetivos, a continuación se muestra los flujos de interacción obtenidos de acuerdo a los escenarios planteados anteriormente.

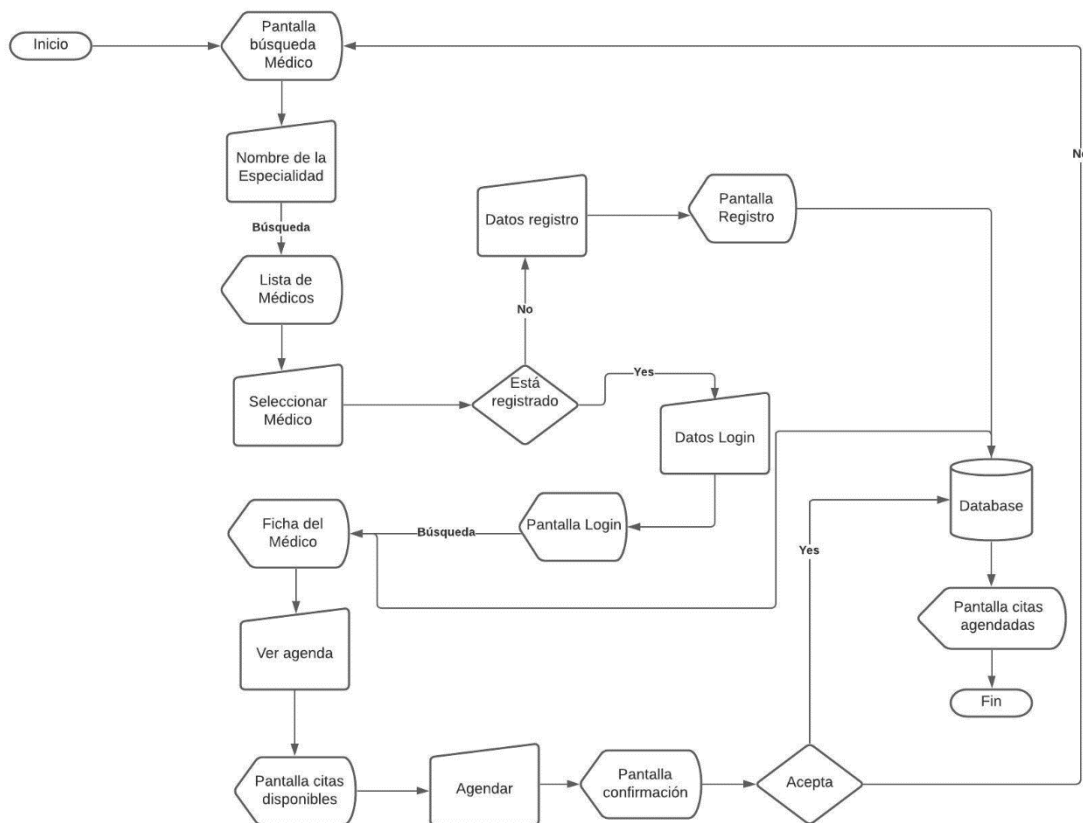


Ilustración 1. Flujo para reservar una cita médica

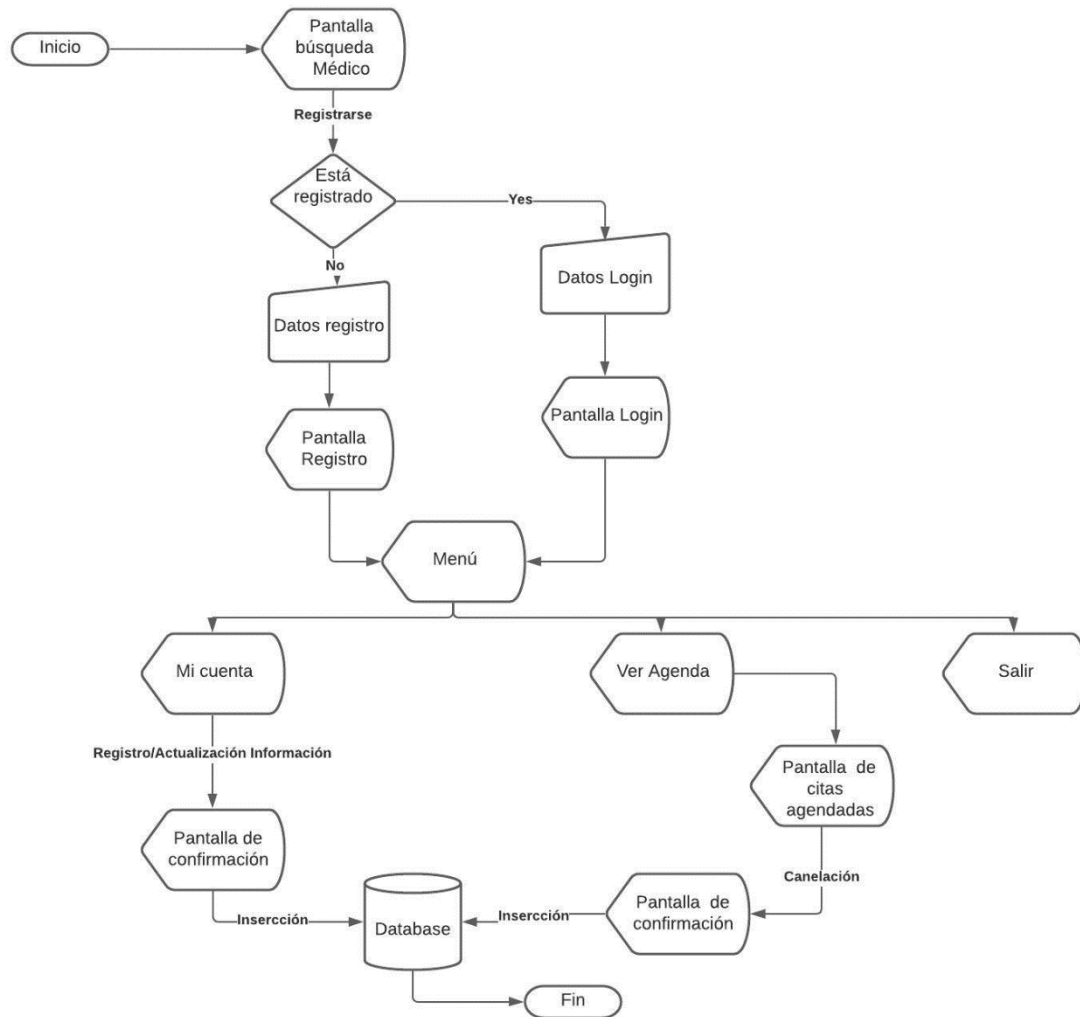


Ilustración 2. Flujo para registrar información por el usuario

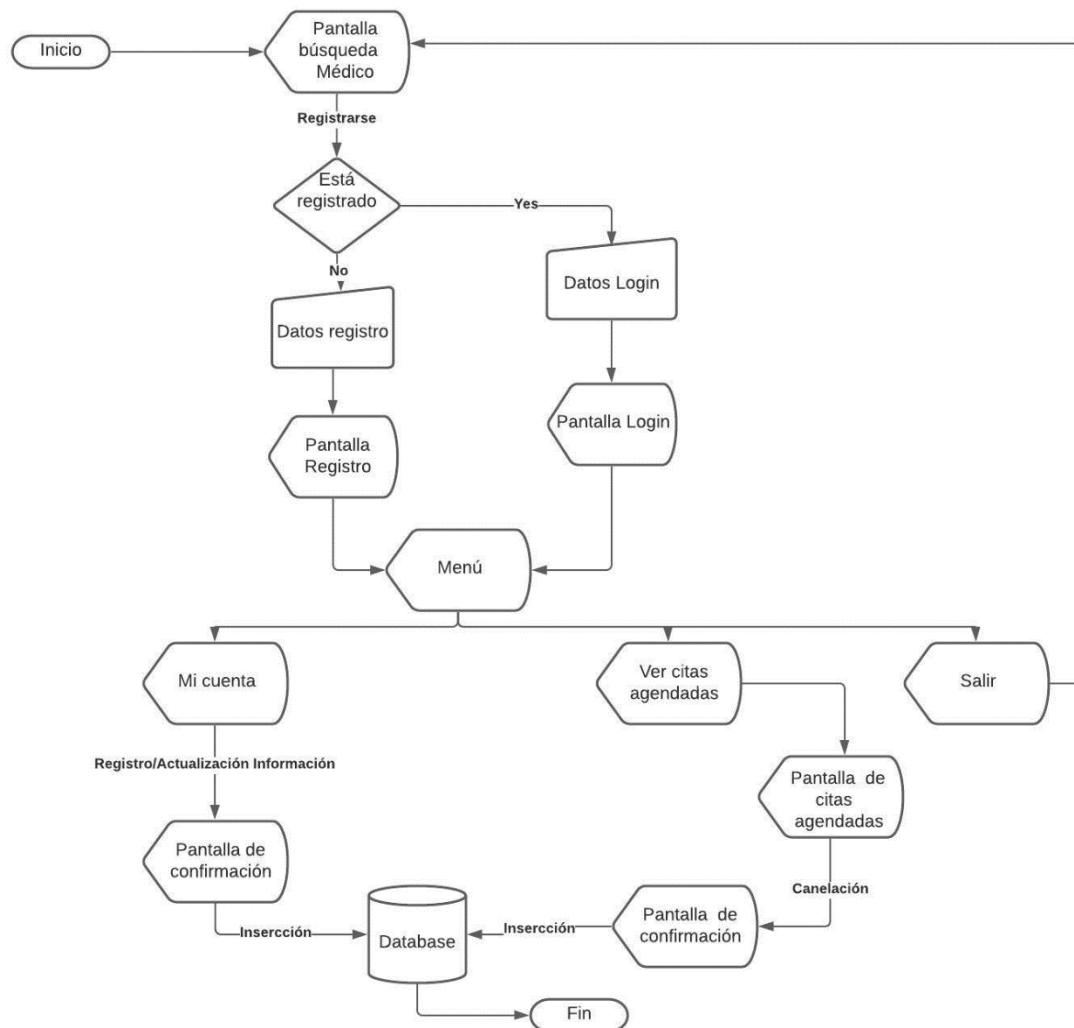


Ilustración 3. Flujo para registrar información por el médico

2.5 Prototipado

En esta parte realizaremos un prototipo horizontal de alta fidelidad de la aplicación, un prototipo no es más que una representación del diseño que tendrá nuestra aplicación antes de la implementación, en el caso del prototipo horizontal muestra las características de la aplicación sin las funcionalidades, y como se mencionó la idea es que es que pueda ser validada por el usuario antes del desarrollo.

2.5.1 Sketches

Para el proceso de prototipado primero empezamos con los sketches a mano antes de entrar a detalle con otras técnicas, todo esto ayudado del diseño conceptual.

Registrarse

LOGO

Email:

Contraseña:

Ingresar

REGISTRARSE

LOGO

Email:

Contraseña:

Confirmar Contraseña:

Es Médico

Registrarse

LISTA DE MÉDICOS

FOTO	NOMBRE: APELLIDO: DIRECCIÓN: PRECIO:
FOTO	NOMBRE: APELLIDO: DIRECCIÓN: PRECIO:

Perfil

Ver citas Agendadas

Salir

CITAS AGENDADAS

FECHA	HORA	ESTADO	ACCIÓN
15-12-2020	15:00	Agendada	Cancelar

Perfil

Salir

DATOS MÉDICO

FOTO

NOMBRES:

APELLIDOS:

DIRECCIÓN:

PRECIO CONSULTA:

ESPECIALIDAD:

HORARIO ATENCIÓN:

DETALLE ATENCIÓN:

Guardar

AGENDAR CITA

10-12-2020	<input type="radio"/>
11-12-2020	<input checked="" type="radio"/>
12-12-2020	<input type="radio"/>

HORA	Acción
09:00	Reservar
09:45	Reservar
09:30	Reservar

DATOS USUARIO

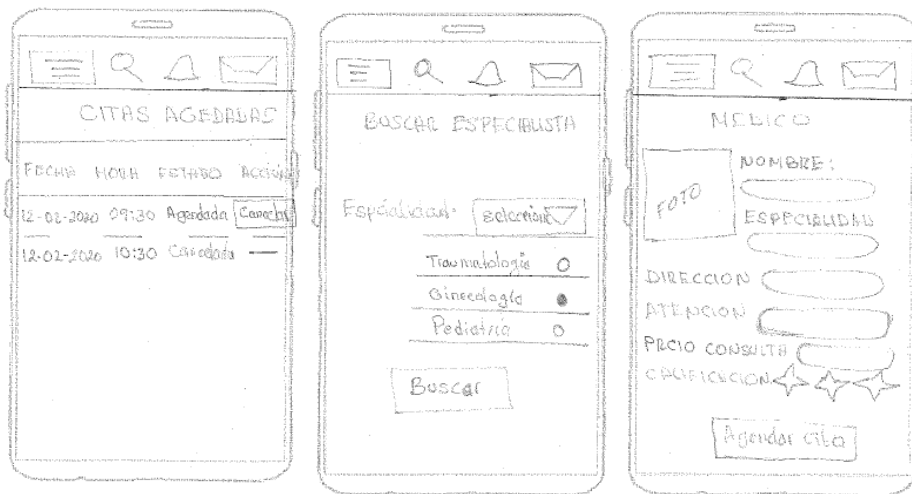
NOMBRES:

APELLIDOS:

CELULAR:

DIRECCIÓN:

Guardar



2.5.2 Prototipos

Una vez realizados los sketches, procedemos con un diseño de más alta fidelidad.

2.5.2.1 Pantalla de Ingreso



Ilustración 4. Pantalla de ingreso

Esta pantalla es la que verá el usuario antes de ingresar a la aplicación, aquí debe ingresar el email, su password y en el caso de ser un médico deberá seleccionar la casilla, posterior deberá presionar el botón ingresar, en el caso de tener una cuenta, procederá con la pantalla de registro.

2.5.2.2 Pantalla de Registro



Ilustración 5. Pantalla de registro

Esta pantalla permitirá registrar un usuario o médico, para esto deberá ingresar su email, un password, indicar si es médico y presionar el botón registrarse.

2.5.2.3 Menú de usuario

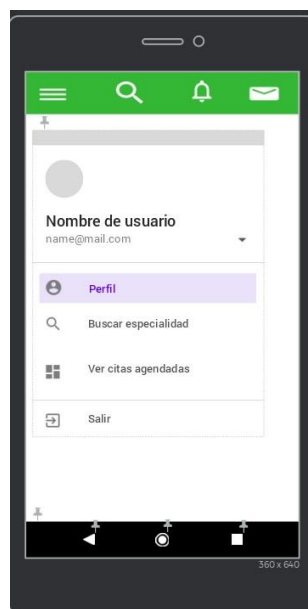


Ilustración 6. Menú de usuario

Esta pantalla se muestra el menú para el usuario, aquí puede acceder a modificar su perfil, buscar un médico por especialidad, ver las citas agendadas y salir de la aplicación.

2.5.2.4 Menú de médico

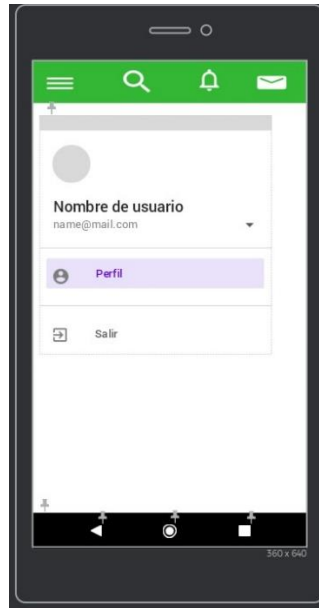


Ilustración 7. Menú de médico

Esta pantalla se muestra el menú para el médico, aquí puede acceder a modificar su perfil y salir de la aplicación.

Para ver las citas que tiene, debe ir al menú superior y elegir la opción de lupa, también puede ver las notificaciones cada vez que se haya agendado una cita por un usuario y una opción de chat.

2.5.2.5 Pantalla principal de usuario

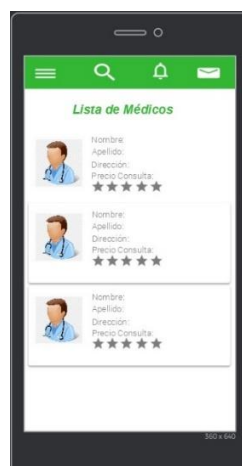


Ilustración 8. Pantalla lista de médicos

Esta pantalla es la que verá el usuario al acceder a la aplicación, en esta se muestran todos los médicos, si desea agendar una cita presiona el médico deseado y pasa a la pantalla de agendar cita.

En la parte superior existe el menú que le permite volver a mostrar todos los médicos, para esto debe ir a la opción de lupa, cuenta con una opción de notificaciones para recordatorios de antes de acudir a la cita médica y una opción de chat.

2.5.2.6 Pantalla de búsqueda de médicos



Ilustración 9. Pantalla de búsqueda de un médico

Esta pantalla permite al usuario buscar un médico por especialidad, elije la especialidad y presiona buscar.

2.5.2.7 Pantalla para agendar cita médica

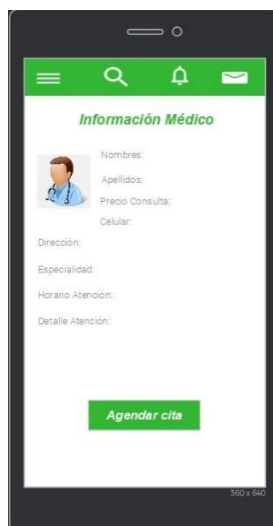


Ilustración 10. Pantalla agendar cita

Esta pantalla muestra la información más a detalle del médico y la opción de agendar una cita.

2.5.2.8 Pantalla de reserva



Ilustración 11. Pantalla reserva

Esta pantalla permite al usuario reservar la cita médica en el horario y fecha seleccionado, para esto deberá seleccionar la fecha y luego se le desplegará los horarios disponibles.

2.5.2.9 Pantalla de citas agendadas



Ilustración 12. Pantalla citas agendadas

Esta pantalla permite al usuario ver las citas médica agendadas, para esto deberá seleccionar la opción en el menú lateral, también le permite cancelar una cita médica agendada, para esto deberá presionar cancelar.

2.5.2.10 Pantalla de registro/actualización datos de usuario



Ilustración 13. Pantalla datos de usuario

Esta pantalla permite al usuario registrar o actualizar la información de su perfil, para esto deberá llenar la información y presionar el botón guardar.

2.5.2.11 Pantalla de registro/actualización de datos médico



Ilustración 14. Pantalla datos de médico

Esta pantalla permite al médico registrar o actualizar la información de su perfil y publicarlo, para esto deberá llenar la información y presionar el botón guardar.

2.5.2.12 Pantalla de citas agendadas médico



Ilustración 15. Pantalla citas agendadas médico

Esta pantalla permite al médico ver las citas agendadas, para esto deberá elegir la fecha, además puede cancelar una cita presionando el botón cancelar.

2.6 Evaluación

El proceso DCU es un proceso iterativo, con lo que en esta fase vamos a evaluar el prototipo anteriormente realizado, para esto en esta fase diseñamos un conjunto de tareas que nos permitirán obtener el feedback del usuario y mejorar nuestra aplicación.

A continuación, se muestran las preguntas propuestas para este fin.

2.6.1 Preguntas al usuario

Edad:

Sexo:

Ocupación:

Estado Civil:

Ciudad:

2.6.1.1 Tareas a realizar

Tarea 1: Registro e ingreso a la aplicación

Tarea 2: Buscar un médico por especialidad

Tarea 3: Agendar una cita médica

Tarea 4: Publicar un perfil como médico

2.6.1.2 Preguntas sobre las tareas

Aquí intentaremos evaluar la facilidad de uso de la aplicación, que funcionalidad le resulto más complicada.

1. ¿Con una puntuación entre 1 y 10 cómo valora el manejo de la aplicación?
2. ¿Qué tareas le resultaron complicadas y como cree que se mejorarían?
3. ¿Qué funcionalidades cree que le faltaría?
4. ¿Cómo le pareció el aspecto visual, en que debería mejorar?
5. ¿Usaría la aplicación para agendar una cita médica?

2.7 Definición de Casos de Uso

A continuación, se muestran el diagrama de casos de uso.

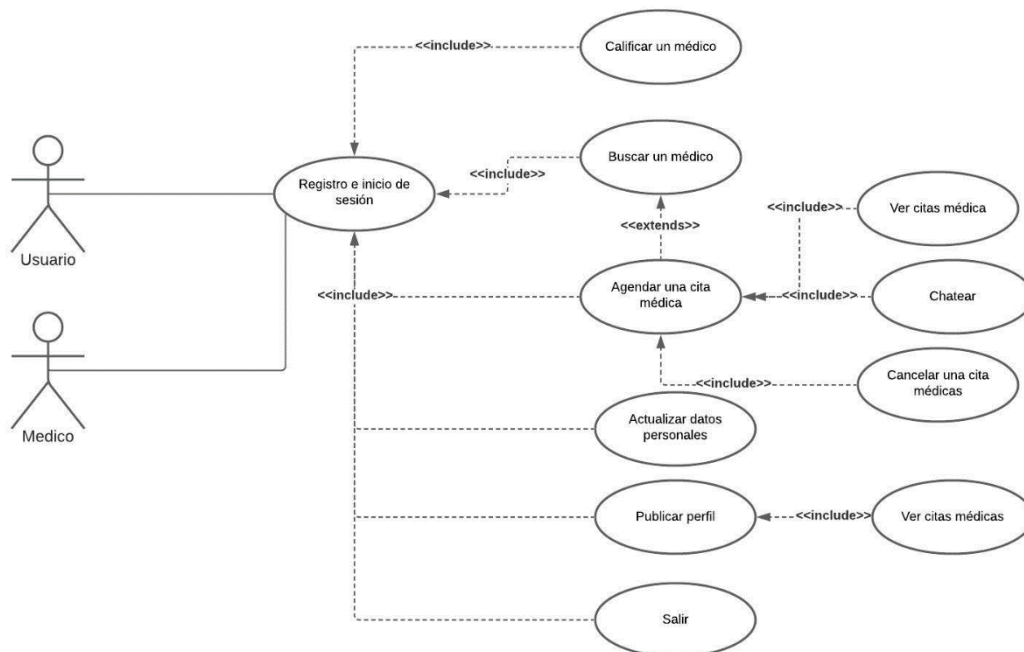


Ilustración 16. Diagrama de Casos de Uso

2.7.1 Detalle de casos de uso

A continuación, vamos a ver detalle el comportamiento de la aplicación frente acciones que realice el usuario.

Identificador	CU-001
Nombre	Registro de usuario
Prioridad	Alta
Descripción	Consiste en el registro del usuario para acceder a la aplicación
Actores	Usuario o médico de la aplicación
Precondiciones	Estar en la pantalla de registro y tener un correo electrónico
Iniciado por	Usuario o médico no registrado
Flujo	<ol style="list-style-type: none"> 1. Abrir la aplicación 2. En la pantalla de Login ir al boton Registarse 3. Ingresar los datos solicitados y presionar el botón Registrarse
Postcondiciones	Registro del usuario
Notas	Validará con un mensaje los campos obligatorios o si el usuario ya está registrado

Tabla 8: Caso de uso: Registro de usuario

Identificador	CU-002
Nombre	Ingresar a la aplicación
Prioridad	Alta
Descripción	Consiste en la autenticación del usuario para acceder a la aplicación y poder hacer uso de las funcionalidades
Actores	Usuario o médico de la aplicación
Precondiciones	Haberse registrado y estar en la pantalla de inicio de la aplicación
Iniciado por	Usuario o médico no registrado
Flujo	<ol style="list-style-type: none"> 1. Abrir la aplicación

	2. Ingresar los datos solicitados y presionar el botón Registrarse
Postcondiciones	Usuario autenticado
Notas	Validará con un mensaje los campos obligatorios o si la información no es correcta.

Tabla 9: Caso de uso: Ingresar a la aplicación

Identificador	CU-003
Nombre	Buscar un médico
Prioridad	Media
Descripción	Consiste en buscar un médico por especialidad
Actores	Usuario de la aplicación
Precondiciones	Haberse autenticado en la aplicación
Iniciado por	Usuario de la aplicación
Flujo	<ol style="list-style-type: none"> 1. Desplegar el menú lateral 2. Elegir la opción Buscar por especialidad 3. Elegir la especialidad 4. Presionar el botón Buscar
Postcondiciones	Lista de médicos filtrados por especialidad
Notas	Ninguna

Tabla 10: Caso de uso: Buscar un médico

Identificador	CU-004
Nombre	Agendar una cita médica
Prioridad	Media
Descripción	Consiste en agendar una cita médica
Actores	Usuario de la aplicación
Precondiciones	Haberse autenticado en la aplicación y realizado el CU-006
Iniciado por	Usuario de la aplicación

Flujo	<ol style="list-style-type: none"> 1. Presionar la lupa del menú superior o haber realizado el CU-003 2. De la lista mostrada presionar sobre el medico de su elección 3. En la pantalla Información Médico presionar le botón Agendar cita 4. En la pantalla Agendar Cita seleccionar la fecha y elegir el horario posteriormente presionar el botón Reservar
Postcondiciones	Cita médica agendada
Notas	Se mostrará un mensaje de confirmación antes de la reserva

Tabla 11: Caso de uso: Agendar una cita medica

Identificador	CU-005
Nombre	Ver citas médicas
Prioridad	Media
Descripción	Consiste en ver las cita médicas agendadas
Actores	Usuario de la aplicación
Precondiciones	Haberse autenticado en la aplicación y realizado el CU-004
Iniciado por	Usuario de la aplicación
Flujo	<ol style="list-style-type: none"> 1. En el menú lateral presionar la opción Ver citas agendadas
Postcondiciones	Citas agendadas por parte del usuario mostradas
Notas	Ninguna

Tabla 12: Caso de uso: Ver citas médicas

Identificador	CU-006
Nombre	Actualizar datos
Prioridad	Media
Descripción	Consiste en actualizar información del usuario posterior al registro

Actores	Usuario de la aplicación
Precondiciones	Haberse autenticado en la aplicación
Iniciado por	Usuario de la aplicación
Flujo	<ol style="list-style-type: none"> 1. En el menú lateral presionar la opción Perfil 2. En la pantalla Datos Usuario ingresar la información solicitada y presionar el botón Guardar
Postcondiciones	Datos del usuario actualizados
Notas	Se mostrará un mensaje de validación para campos obligatorios o campos no válidos.

Tabla 13: Caso de uso: Actualizar datos

Identificador	CU-007
Nombre	Cancelar una cita médica
Prioridad	Media
Descripción	Consiste en cancelar una cita médica previamente agendada
Actores	Usuario de la aplicación
Precondiciones	Haberse autenticado en la aplicación y por lo menos una cita médica agendada
Iniciado por	Usuario de la aplicación
Flujo	<ol style="list-style-type: none"> 1. En el menú lateral presionar la opción Ver citas agendadas 2. En la pantalla Citas Agendadas presionar el botón Cancelar
Postcondiciones	Cita cancelada por el usuario
Notas	Antes de cancelar la cita se mostrará un botón de confirmación, posterior se mostrará la cita como cancelada

Tabla 14 Caso de uso: Cancelar una cita por el paciente

Identificador	CU-008
Nombre	Chatear con un médico

Prioridad	Media
Descripción	Consiste en cancelar iniciar un chat con un médico
Actores	Usuario de la aplicación
Precondiciones	Haberse autenticado en la aplicación y haber tenido una cita médica agendada con ese médico
Iniciado por	Usuario de la aplicación
Flujo	<ol style="list-style-type: none"> 1. En el menú superior presiona el icono de mensajes 2. Elegir el médico para iniciar el chat
Postcondiciones	Chat con un médico iniciado
Notas	Ninguna

Tabla 15: Caso de uso: Chatear con un médico

Identificador	CU-009
Nombre	Publicar perfil
Prioridad	Alta
Descripción	Consiste en publicar el perfil por un médico para que un usuario pueda agendar una cita
Actores	Médico
Precondiciones	Haberse autenticado en la aplicación
Iniciado por	Médico
Flujo	<ol style="list-style-type: none"> 1. En el menú lateral elegir la opción Perfil 2. En la pantalla Datos Médico ingresar la información solicitada y presionar el botón Guardar
Postcondiciones	Médico guardado y/o perfil publicado
Notas	Se mostrará un mensaje de validación para campos obligatorios o campos no válidos.

Tabla 16: Caso de uso: Publicar perfil

Identificador	CU-010
----------------------	---------------

Nombre	Ver citas médicas
Prioridad	Alta
Descripción	Consiste en que un médico pueda ver las citas médicas agendadas
Actores	Médico
Precondiciones	Haberse autenticado en la aplicación y tener publicado el perfil como médico
Iniciado por	Médico
Flujo	<ol style="list-style-type: none"> 1. En el menú superior elegir la opción de lupa 2. En la pantalla Citas Agendadas seleccionar la fecha y posterior se mostrará las citas agendadas.
Postcondiciones	Mostrar citas agendadas
Notas	Ninguna

Tabla 17: Caso de uso: Ver citas médicas

Identificador	CU-011
Nombre	Cancelar una cita médica por un médico
Prioridad	Media
Descripción	Consiste en cancelar una cita médica
Actores	Usuario de la aplicación
Precondiciones	Haberse autenticado en la aplicación y tener publicado el perfil como médico y tener al menos una cita agendada
Iniciado por	Usuario de la aplicación
Flujo	<ol style="list-style-type: none"> 1. En el menú superior elegir la opción de lupa 2. En la pantalla Citas Agendadas seleccionar la fecha y posterior se mostrará las citas agendadas. 3. En la pantalla Citas Agendadas presionar el botón Cancelar
Postcondiciones	Cita cancelada por el medico

Notas	Antes de cancelar la cita se mostrará un botón de confirmación, posterior se mostrará la cita como cancelada
--------------	--

Tabla 18: Caso de uso: Cancelar cita por un médico

Identificador	CU-011
Nombre	Salir de la aplicación
Prioridad	Media
Descripción	Consiste en cerrar sesión en la aplicación
Actores	Usuario de la aplicación
Precondiciones	Haberse autenticado en la aplicación
Iniciado por	Usuario de la aplicación
Flujo	1. En el menú lateral elegir la opción Salir
Postcondiciones	Sesión de usuario cerrada
Notas	Ninguna

Tabla 19: Caso de uso: Salir de la aplicación

2.8 Diseño y arquitectura

En este apartado vamos a definir la arquitectura que tendrá la aplicación y todos los artefactos y diagramas en los cuales nos apoyaremos para la implementación.

2.8.1 Diagrama de Clases

A continuación, se muestra el diagrama de clases con los objetos de dominio, a través del diagrama de clases con UML podemos tener una visión general de la aplicación a desarrollar [4].

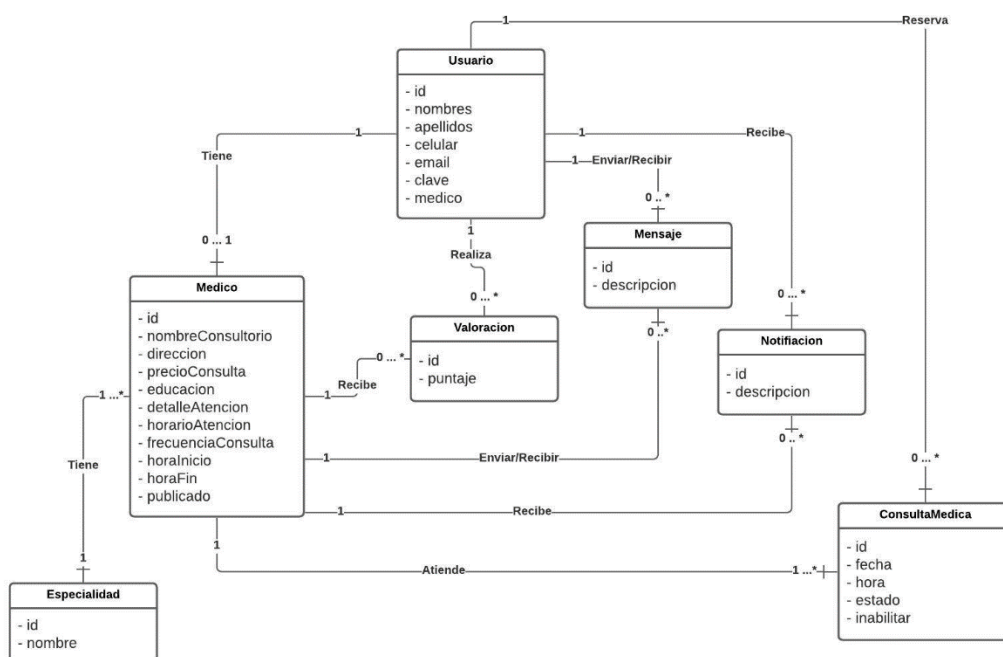


Ilustración 17: Diagrama de Clases

2.8.2 Diagrama Entidad Relación

A continuación, se muestra el diagrama Entidad-Relación, a través del diagrama UML se muestran las entidades o tablas de base de datos que vamos a usar para almacenar la información y posterior mostrarla al usuario.

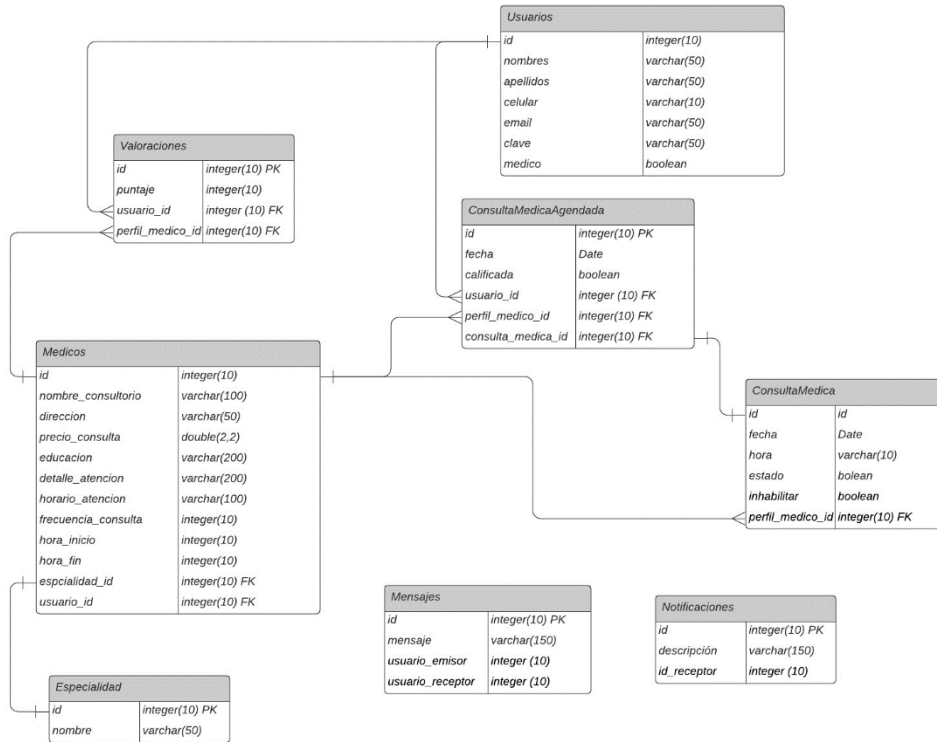


Ilustración 18: Diagrama entidad relación

2.8.3 Arquitectura

Finalmente mostramos la arquitectura como tal, en este caso cualquier dispositivo móvil que instale la aplicación se va a conectar a una API REST que forma parte del backend y a la que realizará las diferentes peticiones.

La aplicación móvil se desarrollará usando el Framework Flutter[5] ya que permite construir una misma aplicación y que se pueda ejecutar en sistemas operativos Android e IOS.

El backend como tal estará en un droplet de Digital Ocean[6], se ha considerado esta opción ya que se tiene experiencia en el manejo de droplets y son muy fácil de configurar, así mismo el lenguaje que se usará para el desarrollo del API REST es Java junto con el Framework Spring Boot y para el almacenamiento se usará una base de datos MySQL.

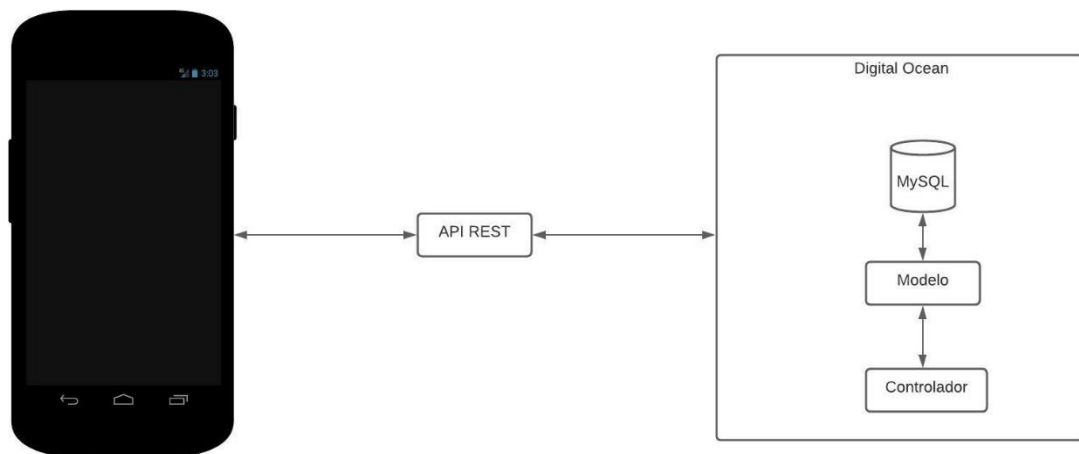


Ilustración 19: Arquitectura de la aplicación

Capítulo 3- Implementación

En este capítulo revisaremos de forma general las herramientas desarrollo usadas para llevar a cabo el proyecto, además revisaremos la estructura principal del proyecto como tal, así mismo la descripción de la implementación de las funcionalidades principales de la aplicación.

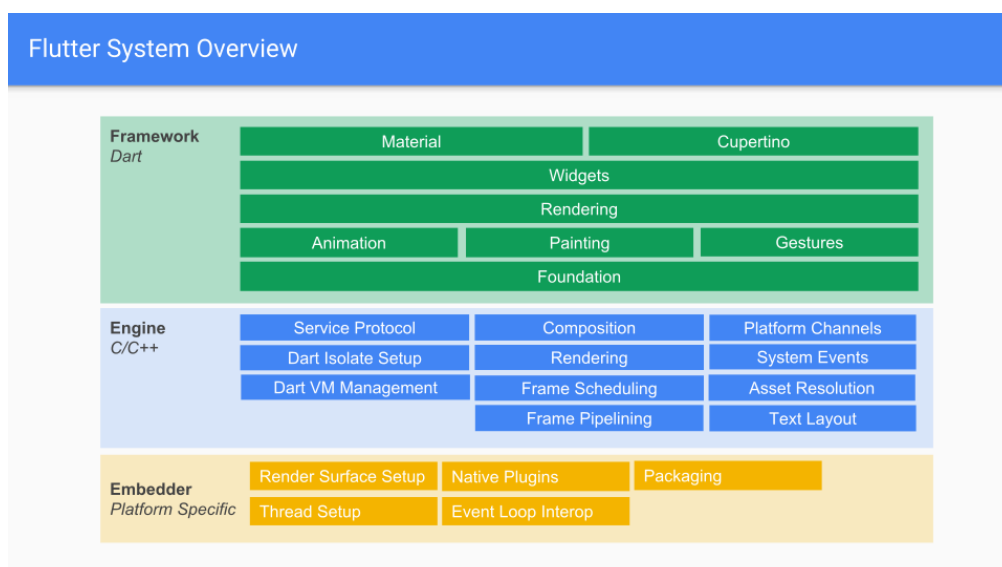
3.1 Flutter

Flutter es un SDK (Software Development Kit) o framework de desarrollo de aplicaciones móviles, aplicaciones web, de escritorio desarrollado por Google y de características open source, por otro lado en la parte de aplicaciones móviles la principal característica que tiene desarrollar con flutter es que podemos compilar el mismo proyecto de forma nativa para Android e IOS, Flutter adaptará a cada sistema operativo y de forma nativa, realizando únicamente cambios ligeros por el programador, otra característica importante y que aporta a la velocidad de desarrollo con Flutter es la implementación de la funcionalidad Hot Reload con lo que los cambios que hacemos se muestran ya sea en el dispositivo o emulador con solo guardarlos en el editor.

Flutter está desarrollado en C++ y Dart que es un lenguaje del mismo Google, hoy en día es un framework muy usado ya que evita el esfuerzo de crear una aplicación por separado para cada una de las plataformas como son Android e IOS y adicional a esto por su velocidad de desarrollo y la implementación de interfaces de usuario adaptados a cada plataforma.

3.1.1 Arquitectura de Flutter

A continuación, se muestra la arquitectura de Flutter:



3.1.1.1 Framework

Esta capa está escrita en Dart y tiene todo lo relacionado con bibliotecas de Animaciones, Widgets y todo lo necesario para crear interfaces de usuario y herramientas de compilación de interfaces de usuario a cada una de las plataformas tanto Android como IOS.

3.1.1.2 Engine

Escrito completamente en C y C++ proporciona soporte a bajo nivel para comunicarse con los SDK propios de cada sistema operativo.

3.1.1.3 Embedder

En esta capa Flutter se rederiza a cada pltaforma en específica y proporciona los servicios para las adaptaciones a cada plataforma (Android IOS) en específico.

3.2 MySQL

MySQL es una base de datos relacional propiedad de Oracle con versiones open source y comercial, es muy usada para el almacenamiento de información de aplicaciones web, aplicaciones de escritorio y móviles en este último mediante el uso de una API.

MySQL se puede administrar mediante una interface de comandos que se instala con el servidor y también a través de interfaces gráficas como, por ejemplo, las más usadas Adminer (phpMyAdmin) o MySQL WorkBeanch.

Entre las características principales que podemos destacar tenemos:

La velocidad, tiene arquitectura multi-hilos lo que permite acceso de clientes de forma concurrentes, así mismo tiene un sistema de caché de consultas que aceleran el procesamiento de datos

Es portable, es un motor de base de datos que puede usarse en diferentes sistemas operativos Linux y Windows.

Fiabilidad y escalable, cuenta con un sistema de seguridad y autenticación de usuarios basado en privilegios, así mismo soporta grandes cantidades de registros por tabla (hasta 50 millones de registros).

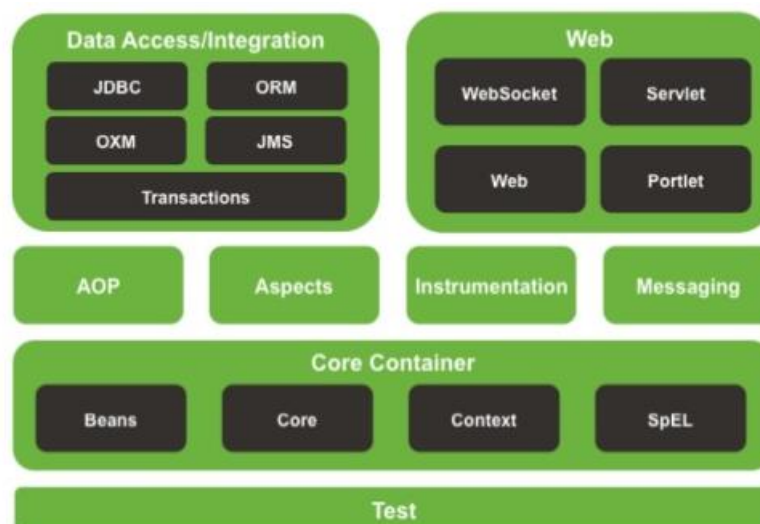
3.3 Spring y Spring Boot

Spring es un potente Framework con características open source y que se usa para desarrollar aplicaciones web con Java, actualmente es muy usado por grandes empresas para desarrollar Microservicios y APIs RESTful que son consumidas por lo general en clientes usando Angular y dispositivos móviles.

Por otra parte, Spring Boot es un complemento de Spring Framework, ya que permite crear aplicaciones con Spring si tener que realizar configuraciones, gestión de versiones sobre las dependencias y todas las tareas adicionales que en un inicio se hacía, con Spring Boot nos dedicamos puramente al tema de desarrollo.

3.3.1 Arquitectura de Spring

A continuación, mostramos la arquitectura de Spring y una explicación breve de las capas más importantes.



3.3.1.1 Web

Esta capa está relacionada con la capa web de una aplicación y es la encargada de manejar y gestionar las peticiones en ambos sentidos tanto desde el cliente a la aplicación como desde la aplicación al cliente.

3.3.1.2 Data Access

Esta capa es la que se encarga de gestionar el acceso a base de datos y todo el tema de transacciones contra una base de datos que realiza una aplicación.

3.3.1.2 Core Container

Esta capa conocida también como el contenedor de Spring, maneja el patrón DI (Inyección de dependencias) y es la encargada de administrar el ciclo de vida de los objetos usados en una aplicación.

3.4 Android Studio

Android Studio es un IDE para desarrollar aplicaciones móviles que en primera instancia soportaba únicamente desarrollo para Android, sin embargo, con la evolución de lenguajes y SDKs para el desarrollo de aplicaciones móviles ha ido

a la par también, actualmente se puede desarrollar aplicaciones móviles con Flutter únicamente añadiendo el plugin con el lenguaje Dart y un plugin propio para el SDK de Flutter.

Android Studio provee un conjunto de APIs y librerías que aportan al desarrollo de aplicaciones móviles, así mismo posee un emulador que permite testear aplicaciones sin tener que precisamente instalarlas en un dispositivo físico.

Algo que se debe tener en cuenta que el IDE como tal es bastante pesado por lo que se necesita un computador con buenas características más que todo en el tema de memoria RAM.

3.4 Git y GitHub

Git y Github son herramientas que nos permiten llevar un completo versionamiento de nuestro proyecto.

3.4.1 Git

Git es un software para el control de versiones, es decir que podemos ir llevando como un historial de los cambios que hacemos en nuestros proyectos y no tenerlo que hacer manualmente como por ejemplo proyecto1, proyecto2 etc., la ventaja de un software de control de versiones es que nos permite en cada momento poder volver a un estado anterior del proyecto con las funcionalidades que en ese momento se encontraba la aplicación, así mismo nos permite trabajar sobre un mismo proyecto diferentes programadores sin tener que luego unir manualmente las partes que cada uno realizó.

3.4.2 GitHub

GitHub es una plataforma que nos permite almacenar proyectos por lo general de software y se conecta con cualquier software de versionamiento que tengamos en local, es una plataforma de tipo colaborativo donde programadores de diferentes lugares pueden trabajar sobre un proyecto, en GitHub podemos tener proyectos públicos como privados, adicionalmente podemos pagar una especie de membresía para proyectos grandes.

3.5. Postman

Postman es un cliente que permite testear APIs tanto servicios SOAP como RESTful, es de uso libre, sin embargo, existen funcionalidades un poco más avanzadas que si tienen un costo, en nuestro caso únicamente usaremos las de uso libre ya que para testear los servicios es suficiente.

3.6. Maven

Maven es un software para la gestión y construcción de proyectos Java, por una parte, permite gestionar las dependencias externas usadas en el proyecto y por otra gestiona el ciclo de vida del proyecto hasta obtener el archivo .jar que será el que despleguemos el servidor virtual.

3.7 Estructura del proyecto

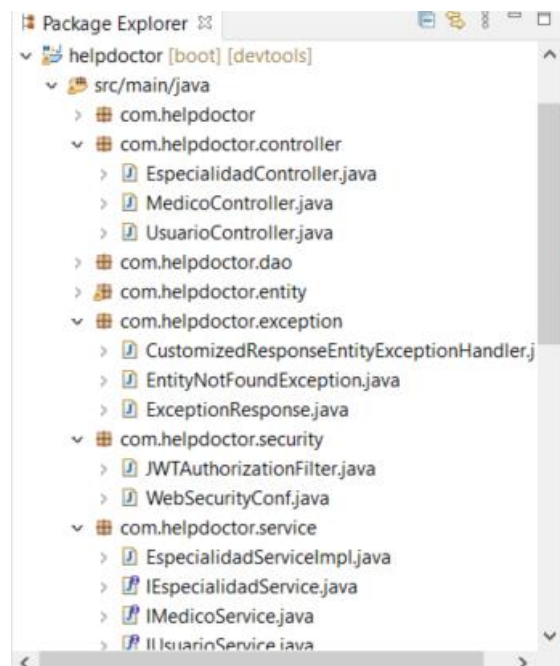
En esta parte vamos a describir brevemente la estructura del proyecto y de los archivos y carpetas más importantes tanto del Backend como del Frontend.

3.7.1. Estructura Backend

Como se ha mencionado se ha desarrollado una API RESTful usando Spring Framework junto con Spring Boot la cual proporciona un conjunto de endpoints que van ser consumidos desde la aplicación móvil en Flutter.

Para el despliegue del API hemos configurado un droplet en Digital Ocean que no es más que un servidor privado en el cual vamos a desplegar el proyecto.

A continuación, se detalla algunos paquetes principales del proyecto.



3.7.1.1. Paquete com.helpdoctor.controller

En este paquete se encuentran los controladores que exponen los endpoints para los recursos expuestos.

Tenemos una clase java llamada `UsuarioController.java` que expone recursos relacionados con el usuario, registro, actualización de datos entre otros.

Tenemos una clase java llamada `MedicoController.java` que expone recursos relacionados con el médico, registro de información que verá el usuario al buscar un médico, actualización de datos entre otros.

Tenemos una clase java llamada `EspecialidadController.java` que permite configurar las especialidades para ser elegidas cuando un médico se registra.

3.7.1.2. Paquete `com.helpdoctor.dao`

Contiene interfaces java extendiendo de `JpaRepository` que permiten el acceso a datos para cada uno de los recursos que expone el API.

Por lo general existe una interface para cada una de las tablas en base de datos.

3.7.1.3. Paquete `com.helpdoctor.entity`

Contiene las clases que se mapean a tablas de base de datos, estas clases sirven para crear objetos, los mismos que son usados para obtener información desde la base de datos y ser usados por los DAOS, posterior por las clases de servicio y los controladores que envían la información al cliente, así mismo se usan para enviar información a la base de datos enviada desde el cliente, haciendo básicamente el proceso inverso.

3.7.1.4. Paquete `com.helpdoctor.exception`

Contiene las clases en donde se configura las posibles excepciones y mensajes de retorno en el caso que existan errores al hacer peticiones desde el cliente o error por parte del servidor.

3.7.1.5. Paquete `com.helpdoctor.security`

Contiene las clases que gestionan la parte de seguridades de la aplicación, entre ellas podemos mencionar la comprobación y validación del token en cada petición que realiza el cliente, encriptación y decodificación de claves, así mismo también gestionan la parte de seguridades a nivel de roles, usuarios y general de todos los recursos expuestos al cliente.

3.7.1.6. Paquete `com.helpdoctor.service`

Contiene interfaces propias para los métodos CRUD del proyecto, también contiene clases de servicio que implementan las interfaces propias del proyecto y que hacen uso de los DAOS del paquete anterior para el acceso a datos.

3.7.1.6. Recursos disponibles

Información: Recurso para crear especialidades

Método HTTP: POST

URL: <http://157.245.142.187:8080/helpdoctor/api/v1/especialidades>

Información: Recurso para crea un usuario/médico

Método HTTP: POST

URL: <http://157.245.142.187:8080/helpdoctor/api/v1/usuarios>

Información: Recurso para acceder a la aplicación

Método HTTP: POST

URL: <http://157.245.142.187:8080/helpdoctor/api/v1/usuarios/login>

Información: Actualizar información de usuario

Método HTTP: PUT

URL: <http://157.245.142.187:8080/helpdoctor/api/v1/usuarios/{idusuario}>

Información: Publicar Perfil de Médico

Método HTTP: PUT

URL: <http://157.245.142.187:8080/helpdoctor/api/v1/medicos/{idmedico}>

Información: Buscar la lista de especialidades

Método HTTP: GET

URL: <http://157.245.142.187:8080/helpdoctor/api/v1/especialidades>

Método HTTP: GET

Información: Buscar doctores por especialidades

URL:

<http://157.245.142.187:8080/helpdoctor/api/v1/medicos/{idespecialidad}/especialidades>

Información: Buscar usuarios que tienen perfil de medico

Método HTTP: GET

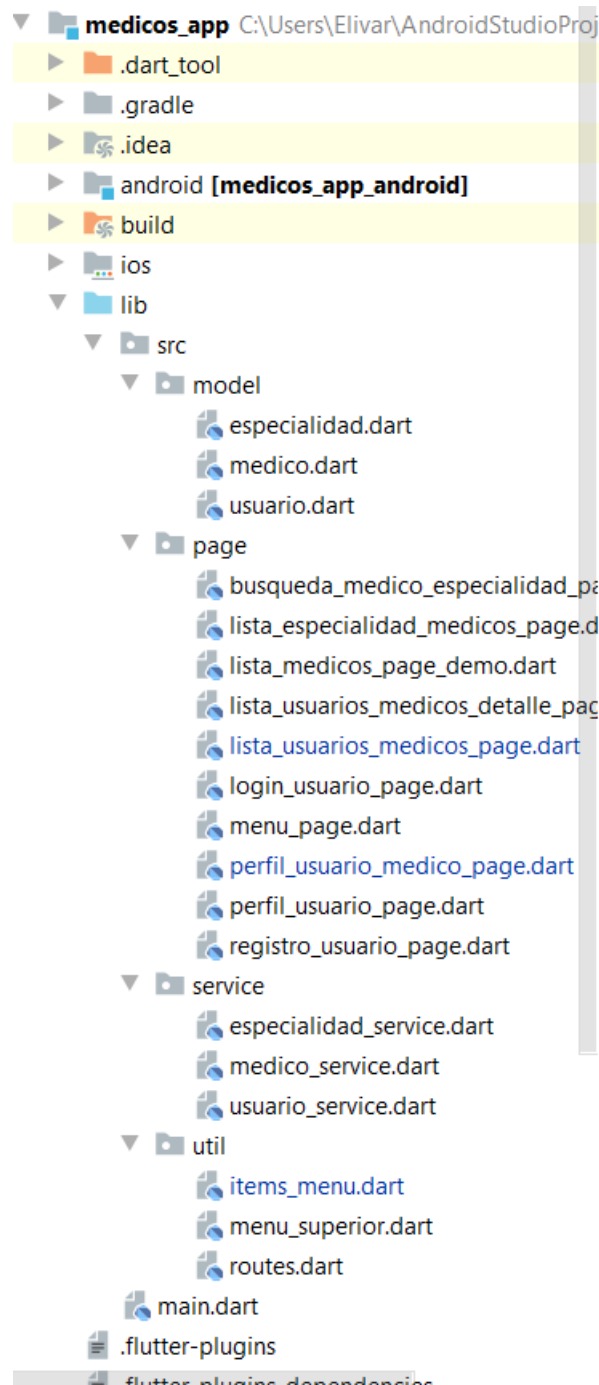
URL: <http://localhost:8080/helpdoctor/api/v1/medicos/{idusuario}/usuarios>

3.7.2. Estructura Frontend

El frontend consiste como tal la aplicación móvil usando Flutter, la cual conectaremos a través del API expuesto en el apartado anterior.

Para esto usamos librerías propias de Flutter que facilitan el desarrollo en cuanto al consumo de servicios API RESTful.

A continuación, se presenta la estructura de la aplicación móvil y una breve descripción de cada uno de los paquetes. Acotar que Flutter no obliga a llevar una arquitectura específica, en ese sentido es muy flexible, con lo que está flexibilidad puede llevar a tener un proyecto totalmente desorganizado, en ese sentido para este proyecto se ha intentado usar el patrón BLoC que como se mencionó Flutter no impone, pero si lo recomienda.



3.7.1.1. Paquete src.model

Este paquete contiene las clases que son las que almacenan la información recibida desde el backend.

3.7.1.2. Paquete src.page

Este paquete contiene las clases que extienden de StatelessWidget y de StatefulWidget y que contienen los diferentes Widgets (campos de texto, botones, formularios etc.) para las interfaces de usuario de la aplicación.

3.7.1.3. Paquete src.service

Este paquete contiene las clases que hacen las llamadas y que se comunican directamente con el API RESTful y que son usadas en las interfaces de usuario.

3.7.1.2. Paquete src.util

Este paquete contiene clases que son de apoyo para el proyecto, como, por ejemplo: el archivo de rutas de la aplicación, menús de usuario entre otros.

3.8. Estado del proyecto

Como se puede ver en la tabla a continuación se tiene avanzado aproximadamente en un 70% en funcionalidades contempladas en un inicio, comentar también que se usó las tecnologías y herramientas descritas en los apartados anteriores debido a que se tenía conocimientos de las mismas y con esto evitar una mayor curva de aprendizaje y tiempo de implementación, por ejemplo al usar Firebase que era una opción de tecnología a usar, adicionalmente al realizarlo con estas herramientas tenemos el control total del backend sin temor a tener actualizaciones tanto en términos de contratos o de funcionalidades que pueda afectar al proyecto.

En esta fase la mayor parte de tiempo que ha contemplado el proyecto, es el tema de configuraciones y despliegue del backend hasta dejarla totalmente estable, por otra parte, se ha tenido que investigar bastante el tema del desarrollo basado en Flutter, su diseño basado en Widgets, implementación de la maquetación diseñada en la fase anterior, el consumo de APIs, pruebas de los servicios REST entre otros temas netos de desarrollo.

Para poder concluir el proyecto satisfactoriamente se ha decidido desarrollar las funcionalidades: reserva, ver citas agendadas y obviar la funcionalidad de chat debido al tiempo ya que además cualquier conversación adicional se la podría hacer vía whatsapp hasta poder implementar la funcionalidad mencionada.

Funcionalidades	Estado (%)	Observaciones
Login	100	Terminado
Menú usuario	100	Terminado
Menú médico	100	Terminado
Pantalla principal(Listar médicos)	80	Completar temas de diseño
Búsqueda de médicos por especialidad	100	Terminado
Información médico(Agendar cita)	100	Terminado
Reserva	10	Iniciado
Ver citas agendadas usuario	0	No iniciado
Chat	0	No iniciado
Ver citas agendadas medico	0	No iniciado
Registro Usuario(médico y paciente)	100	Terminado

Actualizar datos usuario	100	Terminado
Actualizar datos medico	100	Terminado

3.9. Pruebas

Para el proyecto se tiene previsto realizar pruebas unitarias tanto para el backend y para el frontend (aplicación móvil), con lo que validaremos que la funcionalidad sea la correcta.

Primeramente, haremos pruebas unitarias sobre cada recurso de la API usando Postman, con lo que garantizaremos que los recursos implementados del lado del backend están correctamente implementados, que se obtiene los resultados esperados y que podremos acceder a futuro desde la aplicación móvil.

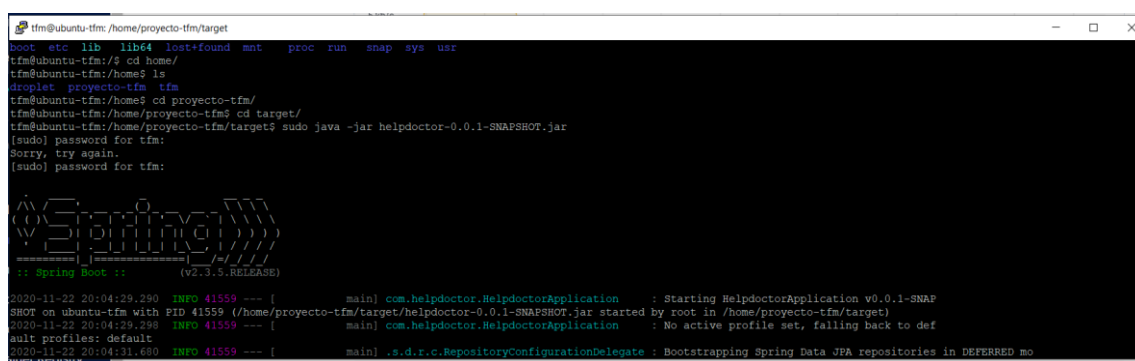
En el caso de las pruebas desde la aplicación, Flutter provee un módulo de test que permite hacer pruebas unitarias, para esto hacemos uso de las dependencias test y mockito desde el mismo proyecto en Flutter.

Posterior probaremos la aplicación sobre el emulador y consumiendo los servicios directamente desde la API que se encuentra en el servidor privado en Digital Ocean.

3.9.1 Pruebas de Backend

En este punto vamos a realizar pruebas unitarias para cada de los recursos expuestos en nuestra API RESTful para esto usaremos Postman y tomaremos como referencia los casos de usos realizados en el capítulo anterior y que apliquen para este tipo de pruebas.

Previamente para hacer las pruebas unitarias con Postman debemos arrancar la aplicación dentro del droplet:



```

tfm@ubuntu-tfm: /home/proyecto-tfm/target
root etc lib lib64 lost+found mnt proc run snap sys usr
tfm@ubuntu-tfm:/$ cd home/
tfm@ubuntu-tfm:~/home$ ls
droplet proyecto-tfm tfm
tfm@ubuntu-tfm:~/home$ cd proyecto-tfm/
tfm@ubuntu-tfm:~/home/proyecto-tfm$ cd target/
tfm@ubuntu-tfm:~/home/proyecto-tfm/target$ sudo java -jar helpdoctor-0.0.1-SNAPSHOT.jar
[sudo] password for tfm:
Sorry, try again.
[sudo] password for tfm:

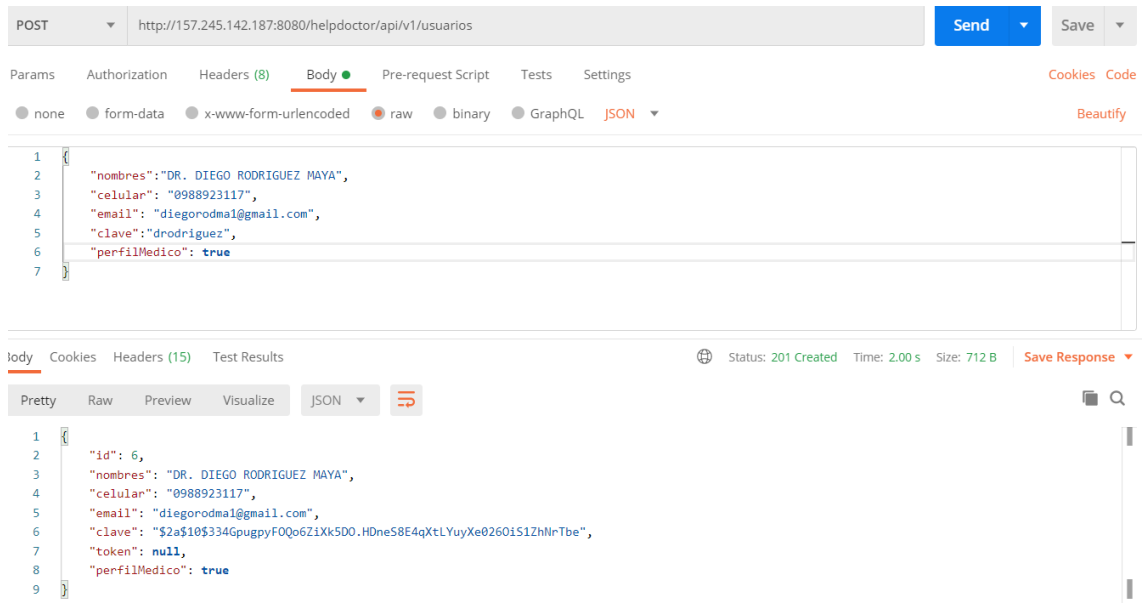
  ____ _
 / ___ \| | | |
/ /___ \| |_| |
 \___ \|____|_|_|_|
  | | | |
  | |_| |
  |_____|

: Spring Boot : (v2.3.5.RELEASE)

2020-11-22 20:04:29.290 INFO 41559 --- [          ] main com.helpdoctor.HelpdoctorApplication : Starting HelpdoctorApplication v0.0.1-SNAPSHOT on ubuntu-tfm with PID 41559 (/home/proyecto-tfm/target/helpdoctor-0.0.1-SNAPSHOT.jar started by root in /home/proyecto-tfm/target)
2020-11-22 20:04:29.298 INFO 41559 --- [          ] main com.helpdoctor.HelpdoctorApplication : No active profile set, falling back to default profiles: default
2020-11-22 20:04:31.680 INFO 41559 --- [          ] main org.springframework.data.jpa.repository.config.JpaRepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in DEFERRED mode
  
```

CU-001: Registro de usuario médico

Procedimiento: Para esto vamos a consumir a través de Postman el recurso: <http://157.245.142.187:8080/helpdoctor/api/v1/usuarios>



POST http://157.245.142.187:8080/helpdoctor/api/v1/usuarios

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings Cookies Code

none form-data x-www-form-urlencoded **raw** binary GraphQL JSON Beautify

```

1 {
2   "nombres": "DR. DIEGO RODRIGUEZ MAYA",
3   "celular": "0988923117",
4   "email": "diegorodma1@gmail.com",
5   "clave": "drodriguez",
6   "perfilMedico": true
7 }

```

body Cookies Headers (15) Test Results Status: 201 Created Time: 2.00 s Size: 712 B Save Response

Pretty Raw Preview Visualize JSON

```

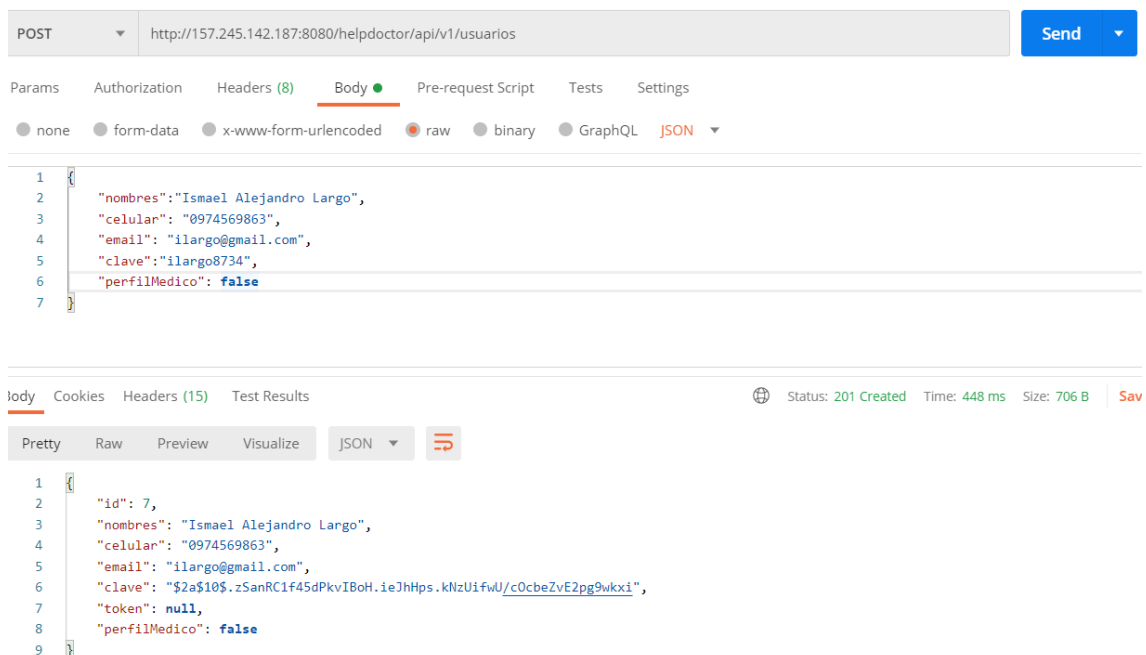
1 {
2   "id": 6,
3   "nombres": "DR. DIEGO RODRIGUEZ MAYA",
4   "celular": "0988923117",
5   "email": "diegorodma1@gmail.com",
6   "clave": "$2a$10$3346pugpyF0Qo6ZiXk5D0.HDneS8E4qXtLYuyXe0260iS1ZhrTbe",
7   "token": null,
8   "perfilMedico": true
9 }

```

Resultado: Obtenemos un código http 201, lo que nos indica que el usuario médico fue creado correctamente, además podemos ver que nos retorna los datos del médico, incluyendo la clave encriptada.

CU-001: Registro de usuario

Procedimiento: Para esto vamos a consumir a través de Postman el recurso: <http://157.245.142.187:8080/helpdoctor/api/v1/usuarios> en este caso el parámetros perfilMedico lo enviaremos con un valor de false.



POST http://157.245.142.187:8080/helpdoctor/api/v1/usuarios

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings Cookies Code

none form-data x-www-form-urlencoded **raw** binary GraphQL JSON Beautify

```

1 {
2   "nombres": "Ismael Alejandro Largo",
3   "celular": "0974569863",
4   "email": "ilargo@gmail.com",
5   "clave": "ilargo8734",
6   "perfilMedico": false
7 }

```

body Cookies Headers (15) Test Results Status: 201 Created Time: 448 ms Size: 706 B Save

Pretty Raw Preview Visualize JSON

```

1 {
2   "id": 7,
3   "nombres": "Ismael Alejandro Largo",
4   "celular": "0974569863",
5   "email": "ilargo@gmail.com",
6   "clave": "$2a$10$.zSanRC1f45dPkvIBoH.ieJhHps.kNzUifwJ/c0cbeZvE2pg9wKxi",
7   "token": null,
8   "perfilMedico": false
9 }

```

Resultado: Obtenemos un código http 201, lo que nos indica que el usuario médico fue creado correctamente.



```
PUT http://157.245.142.187:8080/helpdoctor/api/v1/medicos/6
Body
[id]: 6,
[direccion]: "3ra planta, B1oque B",
[precioConsulta]: 50,
[educacion]: "Cirujano General",
[detalleAtencion]: "Cirujano General",
[horarioAtencion]: "Lunes a viernes de 09:00 a 18:00",
[duracionConsulta]: "30",
[horaInicio]: 9,
[horaFin]: 18
Status: 204 No Content Time: 265 ms Size: 383 B
```

Resultado: Obtenemos un código http 204, este código se retorna cuando el registro se actualizó correctamente.

CU-003: Buscar un médico por especialidad

Procedimiento: Para esto vamos a consumir a través de Postman el recurso: <http://157.245.142.187:8080/helpdoctor/api/v1/especialidades> para buscar todas las especialidades.

```
GET http://157.245.142.187:8080/helpdoctor/api/v1/especialidades
Status: 200 OK Time: 279 ms Size: 1.06 KB
[
  {
    "id": 1,
    "nombre": "NEUROCIURUGÍA"
  },
  {
    "id": 2,
    "nombre": "OTORRINO"
  },
  {
    "id": 3,
    "nombre": "PEDIATRÍA"
  },
  {
    "id": 4,
    "nombre": "ANATOMÍA PATOLÓGICA"
  }
]
```

Resultado: Obtenemos un código http 200 y una lista de especialidades.

Procedimiento: Para esto vamos a consumir a través de Postman el recurso: <http://157.245.142.187:8080/helpdoctor/api/v1/medicos/4/especialidades> en este caso pasamos como parámetro el id de la especialidad y obtenemos una lista con los médicos para esa especialidad.



GET http://157.245.142.187:8080/helpdoctor/api/v1/medicos/4/especialidades Send Save

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies Code

<input checked="" type="checkbox"/> Cookie	JSESSIONID=F1D3AC82DD7476C91AD286669F24AE1D
<input checked="" type="checkbox"/> Postman-Token	<calculated when request is sent>
<input checked="" type="checkbox"/> Host	<calculated when request is sent>
<input checked="" type="checkbox"/> User-Agent	PostmanRuntime/7.26.8
<input checked="" type="checkbox"/> Accept	*/*

Body Cookies (1) Headers (14) Test Results Status: 200 OK Time: 158 ms Size: 985 B Save Response

Pretty Raw Preview Visualize JSON ≡

```
2 {
3   "id": 2,
4   "direccion": "Bloque B, Piso: Planta baja, Consultorio: 103",
5   "precioConsulta": 50.0,
6   "educacion": "Dermatóloga Clínica y Estética",
7   "detalleAtencion": "Dermatóloga Clínica y Estética",
8   "horarioAtencion": "08:00 a 15:00",
9   "duracionConsulta": null,
10  "horaInicio": 0,
11  "horaFin": 0,
12  "usuario": {
13    "id": 2,
14    "nombres": "DRA. HILDA CELI CELI",
15    "celular": "",
16    "email": "hceli10@hotmail.com",
17    "password": "67c6106708071c7a4c416144e5448b2c78bf40e2d70914b40d9e57c6d"
```

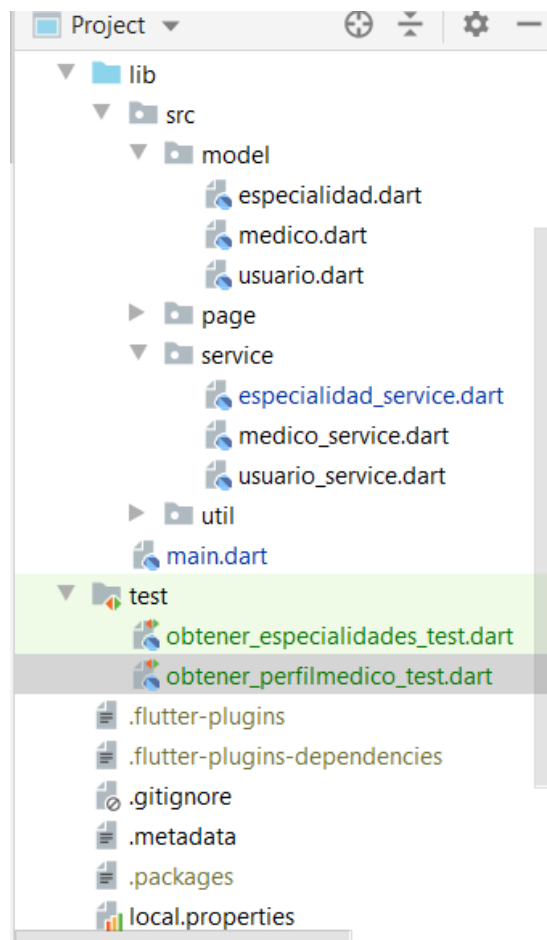
Resultado: Obtenemos un código http 200 y una lista con los médicos para la especialidad buscada.

3.9.2 Pruebas de Frontend

En este apartado vamos a realizar pruebas unitarias directamente desde la aplicación, vamos a usar un paquete llamado Mockito que tiene disponible Flutter para este tipo de pruebas.

Antes de proceder con las pruebas debemos añadir en el archivo pubspec.yaml la dependencia de mockito y luego en el proyecto en la carpeta test crear los archivos para los test que queramos usar, posterior se debe ejecutar el comando **flutter test** desde la terminal

A continuación, se muestra la estructura de los archivos y su ejecución desde la terminal.



```
Terminal: Local x +
C:\Users\Elivar\AndroidStudioProjects\medicos_app>flutter test
00:02 +2: C:\Users\Elivar\AndroidStudioProjects\medicos_app\test\obtener_perfilmedico_test.dart: obtenerMedicoDetalle Retorna el perfil de un médico por usuario
{"titulo": "Test Obtener Perfil Médico por usuario"}
00:02 +5: All tests passed!

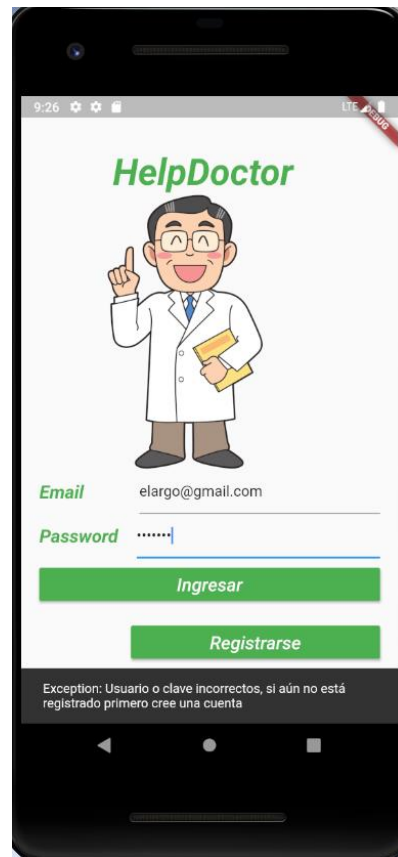
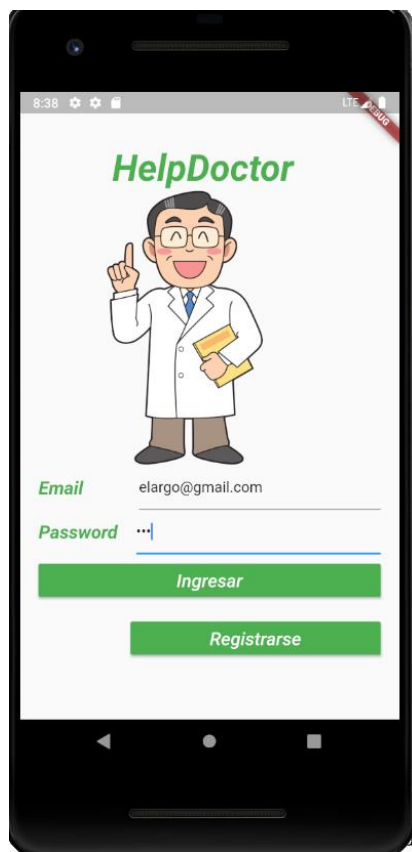
C:\Users\Elivar\AndroidStudioProjects\medicos_app>
```

3.9.3. Imágenes de la aplicación en ejecución

En este apartado se muestra algunas imágenes de la aplicación en ejecución desde el emulador de Android Studio.

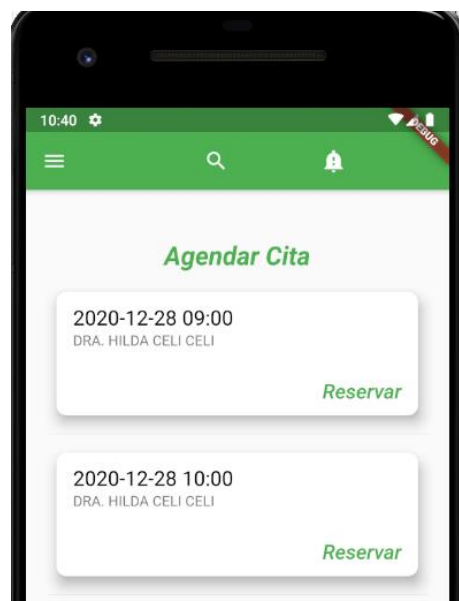
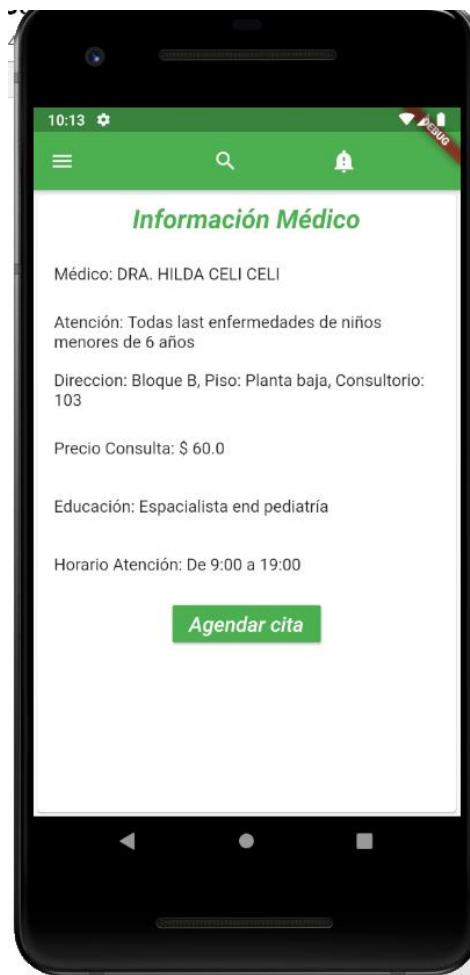
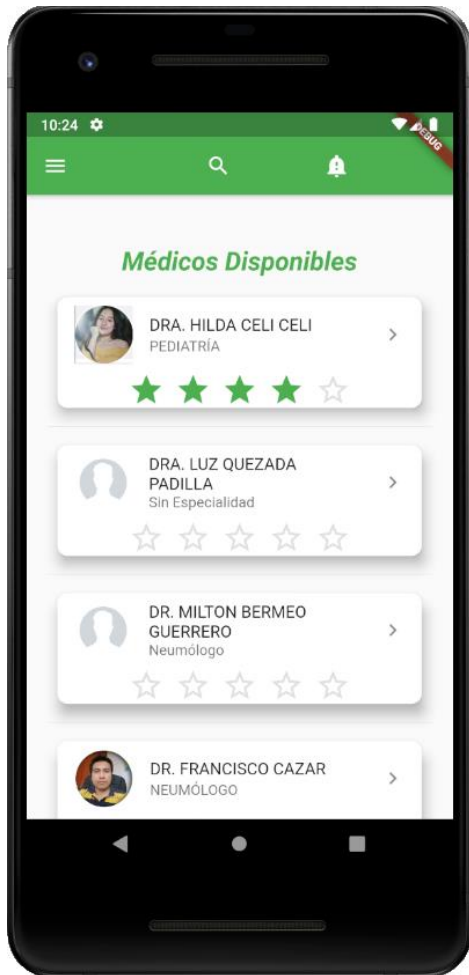
3.9.3.1 Login de usuario

Aquí se muestra la pantalla de inicio en donde el usuario debe ingresar su email y el password, también se muestra cuando las credenciales no son correctas.



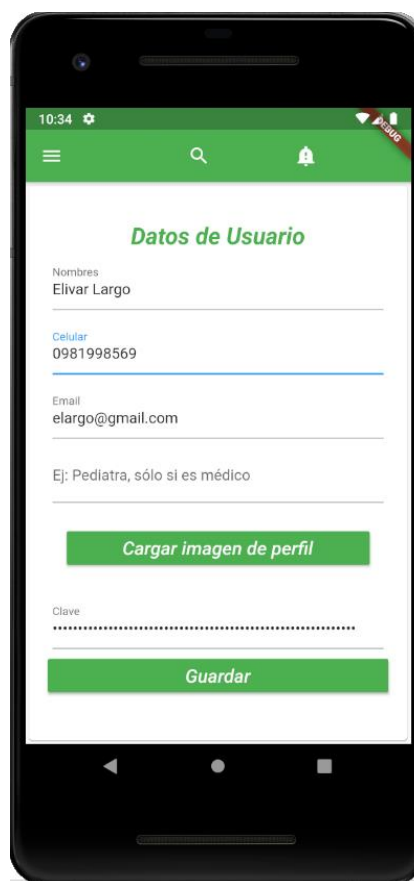
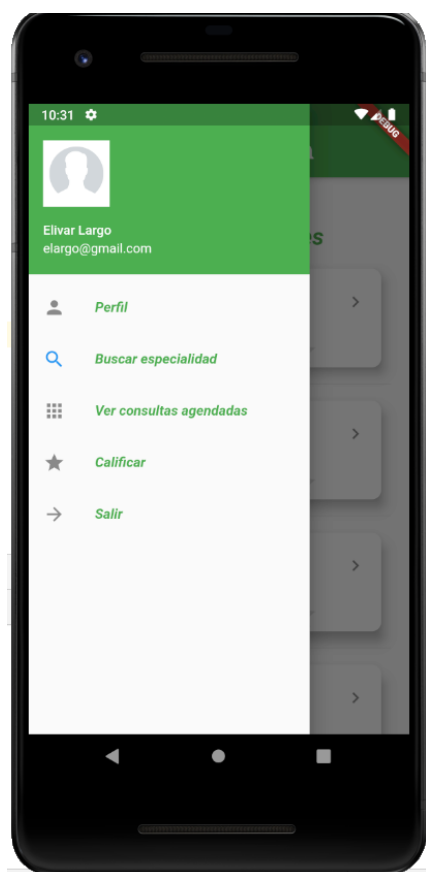
3.9.3.2 Pantalla principal del usuario

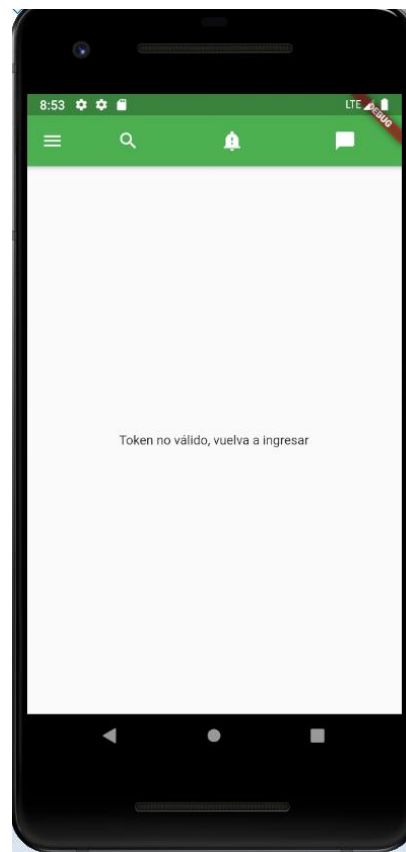
Se muestra la pantalla principal con los médicos disponibles y la pantalla cuando se ha elegido un médico para agendar una cita.



3.9.3.3 Menú del usuario y actualización de datos

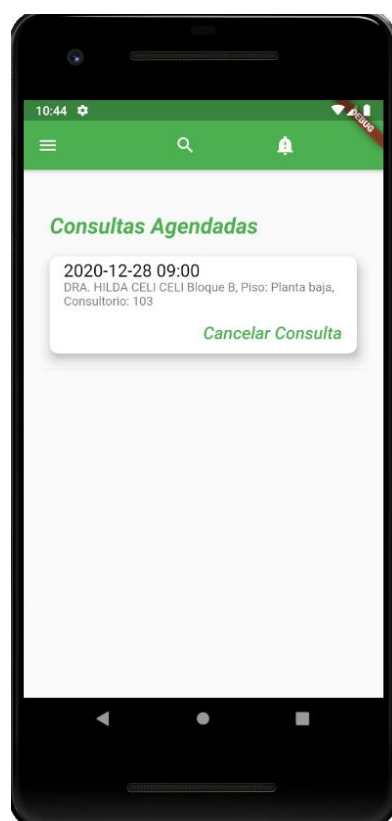
En la secuencia de pantallas anteriores podemos observar el menú lateral del usuario, cuando elige la opción **Perfil** vemos la pantalla de actualización de datos de usuario, así mismo una pantalla donde el usuario procede con la actualización, posterior tenemos una pantalla que se muestra cuando ha expirado el token para esa sesión.





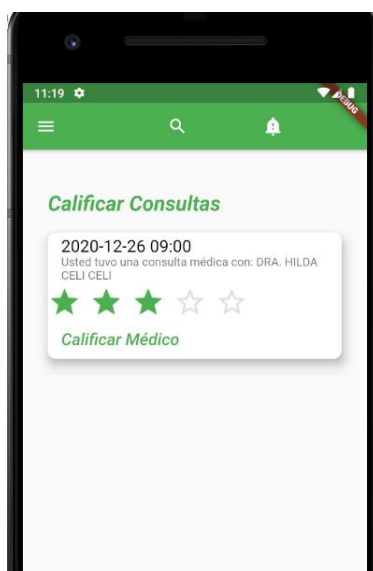
3.9.3.4 Ver consultas agendadas

Cuando vamos al menú Ver consultas agendadas podemos observar la lista de consultas que tenemos, indicando la fecha, hora el médico y la dirección.



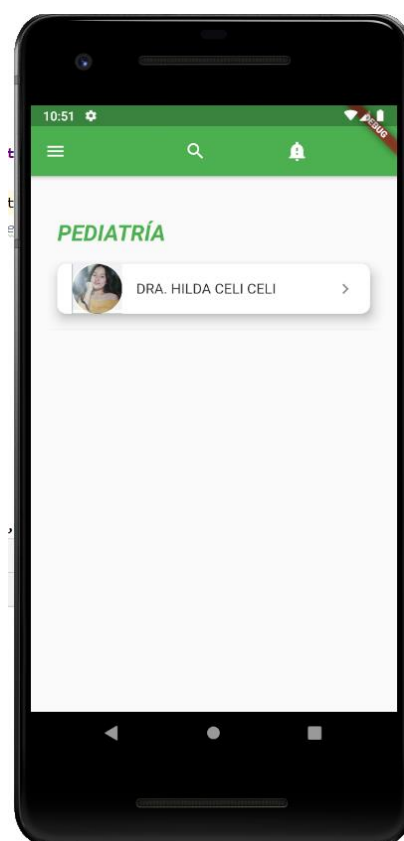
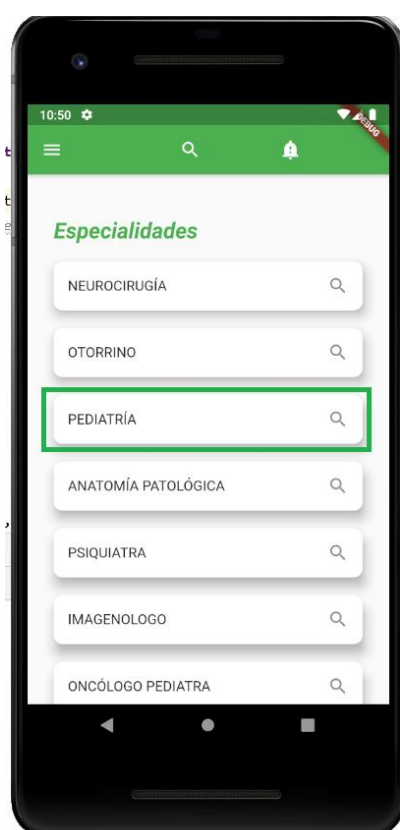
3.9.3.4 Calificar a un médico

Para esto se debe ir al menú lateral y presionar en la opción **Calificar**, se le presentará la pantalla, en la que debe poner la valoración y dar en **Calificar Médico**.



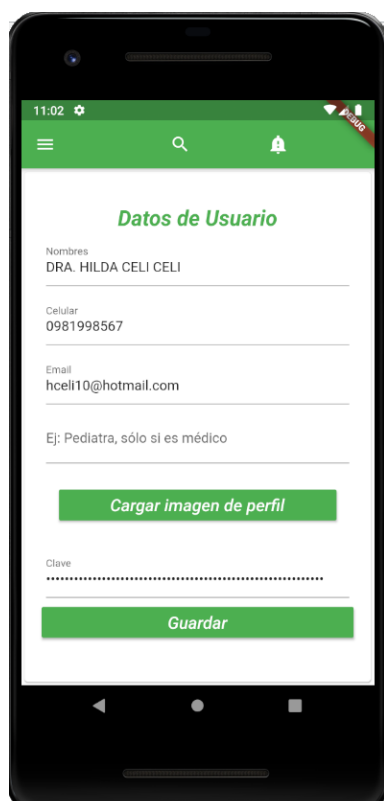
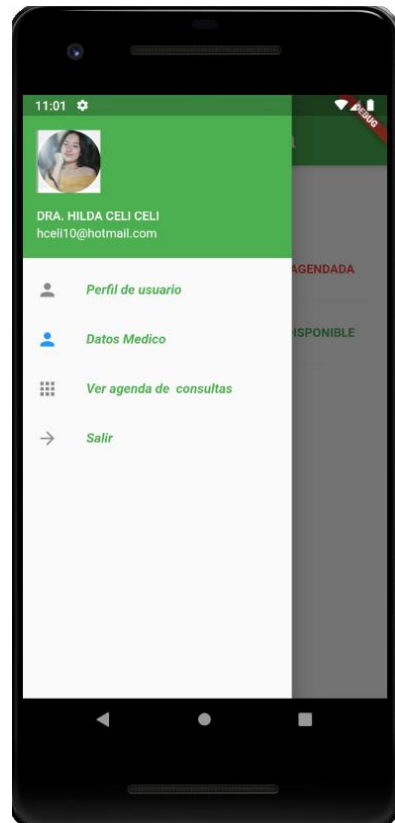
3.9.3.5 Búsqueda por especialidad

Aquí se muestran las pantallas para la búsqueda por especialidad del menú lateral, posterior se muestra la pantalla para el agendamiento de una consulta.



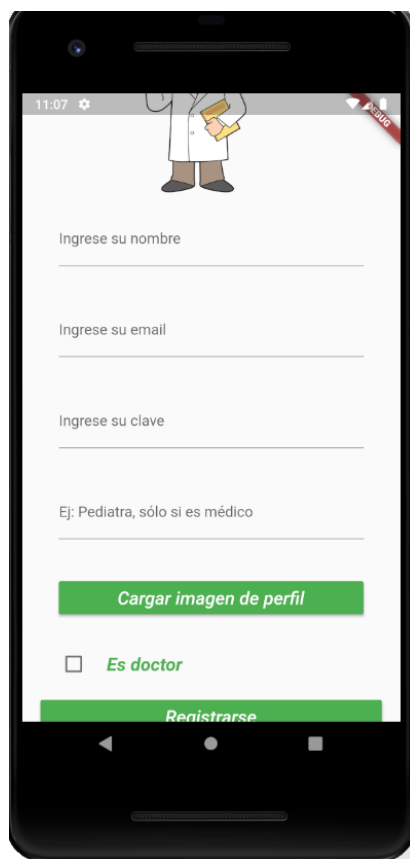
3.9.3.6 Menú del médico y actualización de datos

La primera pantalla que ve un usuario médico es la Agenda de Consultas, posteriormente puede ir al menú lateral y actualizar tanto sus datos de usuario, como sus datos de médico como se muestra en la secuencia de imágenes a continuación.



3.9.3.7 Registro de usuarios

Aquí se muestran la pantalla de registro tanto para un usuario paciente como para un usuario médico.



Capítulo 4- Conclusiones

Al finalizar el presente proyecto podría mencionar las siguientes conclusiones:

- ✓ El uso de metodologías enfocadas al diseño centrado en el usuario como el DCU permiten trabajar cara a cara con el usuario y poder obtener en funcionalidades que verdaderamente necesitan.
- ✓ En este proyecto se apostó por el control total del backend evitando soluciones como Firebase, lo que influyó en que casi la mayoría de funcionalidades se tenga que desarrollar y por ende consumiendo más tiempo de lo planificado.
- ✓ A nivel de desarrollo de la aplicación móvil, puedo decir que Flutter es un framework que tiene una curva de aprendizaje bastante sencilla y que con poco tiempo podemos conseguir grandes resultados.
- ✓ Se ha tenido problemas en la planificación inicial en cuanto a la implementación de las notificaciones y chat, ya que en este caso se tiene que programar desde cero y el tiempo es limitado.
- ✓ En lo personal ha sido una experiencia súper gratificante y enriquecedora ya que a pesar de ser un poco estresante por el hecho de enfrentarme a todas las etapas del proceso de desarrollo: Análisis, Diseño, Desarrollo, Implementación, con tiempos límite y más cuando se tiene que armar dos entornos backend como frontend por separado, sin embargo, esto me ha permitido conocer tanto de forma teórica como práctica y entender todos los desafíos que lleva crear un producto como es una aplicación móvil.

4.1 Líneas de trabajo futuro

Si bien es cierto que el proyecto se ha llevado a cabo en cuanto a requisitos y las funcionalidades obtenidas de acuerdo a las entrevistas con los usuarios, existe muchas otras funcionalidades que sin embargo por la limitación de tiempo no se han podido implementar, pero que sin embargo darían mejor aceptación a la aplicación y que se muestran a continuación.

- ✓ Implementación de la recuperación de contraseña.
- ✓ Validaciones para tener una mejor experiencia de usuario.
- ✓ Geolocalización de médicos de acuerdo a la ubicación.
- ✓ Funcionalidades de chat y notificaciones.
- ✓ Mejorar la parte de diseño gráfico.
- ✓ Comercializar mediante marketing y publicidad.
- ✓ Compilarla para IOS y distribuirla en las tiendas



5. Código Fuente

Aplicación Móvil (app-release.apk)

Drive: <https://www.dropbox.com/s/20i90ex00aotxmj/app-release.apk?dl=0>

Código Fuente Aplicación Móvil (medicos_app.zip)

https://drive.google.com/file/d/196499fodiY-bTC0D_trLPkXHo2OH0vBR/view?usp=sharing

Código Fuente Backend (helpdoctor-backend.zip)

https://drive.google.com/file/d/14YTpkjCGNMvBRR07Nvs6kOXZj8KJ_KXg/view?usp=sharing

ANEXOS

ANEXO 1: Primera Entrega

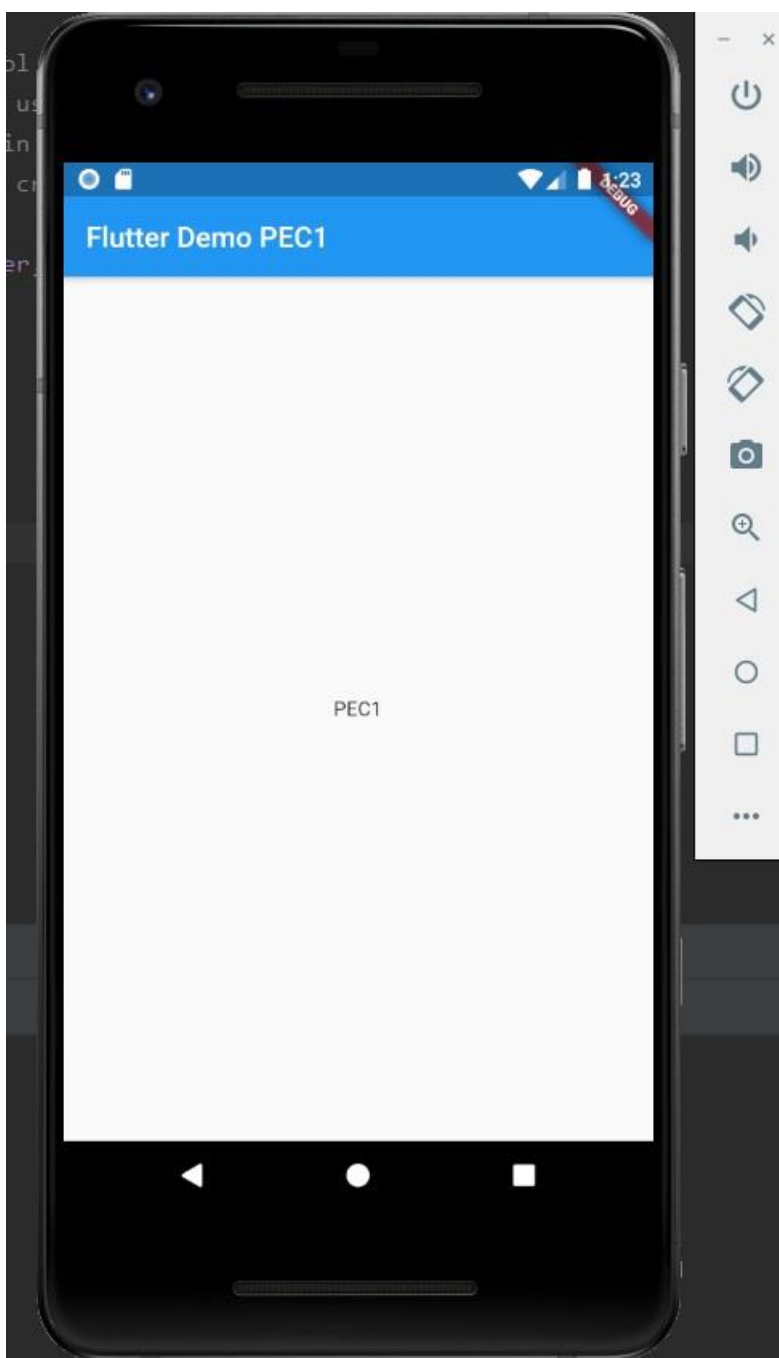
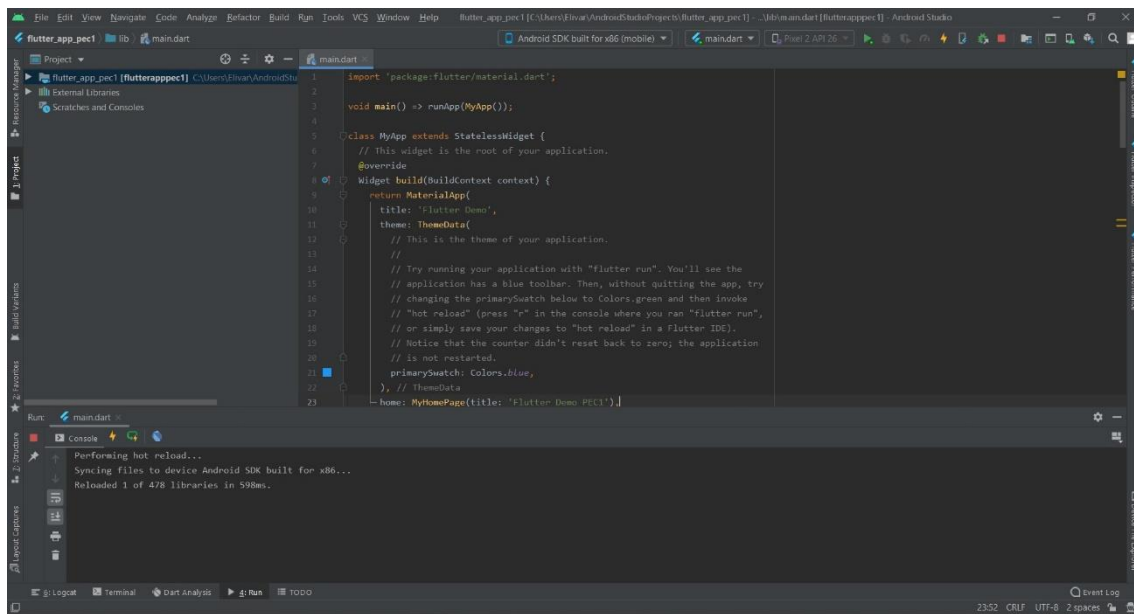


Imagen 1: Presentación texto PEC1 desde la aplicación



```

1 import 'package:flutter/material.dart';
2
3 void main() => runApp(MyApp());
4
5 class MyApp extends StatelessWidget {
6   // This widget is the root of your application.
7   @override
8   Widget build(BuildContext context) {
9     return MaterialApp(
10      title: 'Flutter Demo',
11      theme: ThemeData(
12        // This is the theme of your application.
13        //
14        // Try running your application with "flutter run". You'll see the
15        // application has a blue toolbar. Then, without quitting the app, try
16        // changing the primarySwatch below to Colors.green and then invoke
17        // "hot reload" (press "r" in the console where you ran "flutter run",
18        // or simply save your changes to "hot reload" in a Flutter IDE).
19        // Notice that the counter didn't reset back to zero; the application
20        // is not restarted.
21        primarySwatch: Colors.blue,
22      ), // ThemeData
23      home: MyHomePage(title: 'Flutter Demo Home Page')
24    );
25  }
26 }

```

Console:

```

Performing hot reload...
Syncing files to device Android SDK built for x86...
Reloaded 1 of 478 libraries in 598ms.

```

Imagen 2: Código de la aplicación

Código Fuente: Mi código fuente al estar hecho en Flutter tiene un peso de 180Mb ya comprimido y mes imposible cargarlo en el apartado de entrega, por esta razón lo he subido en mi drive para su descarga.

Ver archivo adjunto flutter_app_pec1.zip

Drive: https://www.dropbox.com/s/bj930lkrxbwrtp/flutter_app_pec1.zip?dl=0



ANEXO 2: Entrevistas

Entrevista 1:

1. Datos básicos del usuario
Edad: 33 años Sexo: femenino Ocupación: abogado Estado Civil: casada Ciudad: Loja
2. Conocer la experiencia que tiene el usuario manejando aplicaciones móviles
1. ¿Tienes aplicaciones móviles instaladas en tu celular, que tipo (juegos, redes, sociales, compras en línea, bancarias)? Si
2. ¿Qué tipos de aplicaciones son las que más usas? Redes sociales.
3. ¿En qué lugar usas más tu teléfono celular para revisar aplicaciones? Casa
4. ¿Qué sistema operativo usa tu celular? Android
3. Conocer el problema del usuario en cuanto a buscar un doctor particular y al agendar una cita médica
5. ¿Con que frecuencia acudes a un médico particular y porque razones has tenido que acudir? Casi nunca
6. ¿Tienes algún familiar que acude a un médico particular, sabes porque razón acude al médico, sabes con qué frecuencia lo hace, sabes si ha tenido problemas al buscar un médico y agendar una consulta? Si, regular, sin problemas para encontrar médico
7. ¿Cómo haces para buscar y agendar una cita con un médico particular? Médico siempre recomendado, llamando al teléfono
8. ¿Cuándo has buscado atención medica particular, buscas un médico especialista o un médico general? Especialista
9. ¿Has tenido algún problema al buscar un médico particular y agendar una cita, como crees que podría solucionar esos problemas en caso de haberlos? Si, guía de médicos, cuando me recomienda un médico, pero no tengo el contacto
10. ¿Has usado alguna aplicación móvil o web para obtener una cita médica, comenta cual ha sido tu experiencia con ella, que es lo que más te ha gustado o disgustado, como crees que podría solucionar esos problemas en caso de haberlos? No
11. ¿Solicitarías un doctor a través de una aplicación móvil, crees que sería una forma eficiente para poder encontrar un médico y agendar una cita? No, por lo general busco con recomendaciones buenas
12. ¿Qué tareas o funcionalidades crees que debería tener la aplicación? Facilidad de búsqueda y manejo, que tenga una forma de calificarlos para que se pueda ver la experiencia de otras personas y en base a ello tomar la decisión de agendar cita



Entrevista 2:

1. Datos básicos del usuario
Edad: 31 Sexo: Hombre Ocupación: Estado Civil: Casado Ciudad: Loja
2. Conocer la experiencia que tiene el usuario manejando aplicaciones móviles
1. ¿Tienes aplicaciones móviles instaladas en tu celular, que tipo (juegos, redes, sociales, compras en línea, bancarias)? Redes sociales, juegos
2. ¿Qué tipos de aplicaciones son las que más usas? Redes sociales
3. ¿En qué lugar usas más tu teléfono celular para revisar aplicaciones? en casa, cuando viajo
4. ¿Qué sistema operativo usa tu celular? Android
3. Conocer el problema del usuario en cuanto a buscar un doctor particular y al agendar una cita médica
5. ¿Con que frecuencia acudes a un médico particular y porque razones has tenido que acudir? Regularmente. Razones: Atención preventiva, por una emergencia
6. ¿Tienes algún familiar que acude a un médico particular, sabes porque razón acude al médico, sabes con qué frecuencia lo hace, sabes si ha tenido problemas al buscar un médico y agendar una consulta? Si. Razón: Atención preventiva, surge algún problema de salud Problemas: No se conoce la ubicación del consultorio del médico, no se conoce la información del médico (Si es recomendable o no), No se sabe la disponibilidad del médico.
7. ¿Cómo haces para buscar y agendar una cita con un médico particular? Acudir directamente al consultorio o visitar una clínica privada y ver la disponibilidad del médico con la especialidad que estoy buscando.
8. ¿Cuándo has buscado atención medica particular, buscas un médico especialista o un médico general? Un médico especialista.
9. ¿Has tenido algún problema al buscar un médico particular y agendar una cita, como crees que podría solucionar esos problemas en caso de haberlos? Si. Problemas: No se conoce la ubicación del consultorio del médico, no se conoce la información del médico (Si es recomendable o no), No se conoce la agenda del médico
10. ¿Has usado alguna aplicación móvil o web para obtener una cita médica, comenta cual ha sido tu experiencia con ella, que es lo que más te ha gustado o disgustado, como crees que podría solucionar esos problemas en caso de haberlos?



No he usado ningún aplicativo.

En caso de haberlo sería de un gran aporte puesto que se podría tener a la mano toda la información del médico, así como su disponibilidad con lo que no se tendría que salir a aventurarse buscando la disponibilidad de un médico.

11. ¿Solicitarías un doctor a través de una aplicación móvil, crees que sería una forma eficiente para poder encontrar un médico y agendar una cita?

Claro que sí, sería eficiente puesto que lo primero que se necesita para hacerse atender es la cita médica previa por lo que se evitaría salir solamente para realizar ese trámite.

12. ¿Qué tareas o funcionalidades crees que debería tener la aplicación?

Mostrar Información relevante del médico tratante.

Agendamiento de citas

Pago en línea

Entrevista 3:

1. Datos básicos del usuario

Edad:32

Sexo: Masculino

Ocupación:

Estado Civil: Casado

Ciudad: Loja

No. y edad de hijos: 1, 7 años

2. Conocer la experiencia que tiene el usuario manejando aplicaciones móviles

1. ¿Tienes aplicaciones móviles instaladas en tu celular, que tipo (juegos, redes, sociales, compras en línea, bancarias)?

Sí, tengo redes sociales, office, compras en línea, pedidos en línea.

2. ¿Qué tipos de aplicaciones son las que más usas?

Ofimática y la nube.

3. ¿En qué lugar usas más tu teléfono celular para revisar aplicaciones?

Tiempos libres.

4. ¿Qué sistema operativo usa tu celular?

Android.

3. Conocer el problema del usuario en cuanto a buscar un doctor particular y al agendar una cita médica

5. ¿Con que frecuencia acudes a un médico particular y porque razones has tenido que acudir?

1 vez al año para médico general para valoraciones, al dentista dos veces al año para profilaxis.

6. ¿Tienes algún familiar que acude a un médico particular, sabes porque razón acude al médico, sabes con qué frecuencia lo hace, sabes si ha tenido problemas al buscar un médico y agendar una consulta?

Sí, mi padre tiene la enfermedad de artritis acude al reumatólogo para poder valorar el avance de la enfermedad y controles de esta, las visitas es cada tres meses y en caso de empeorar acude de inmediato, también visita al oculista con igual frecuencia del reumatólogo.

Es difícil encontrar un médico especialista que tenga recomendaciones o valoraciones en el trabajo que realizan, al agendarlas al ser privados no existe mucho problema.

7. ¿Cómo haces para buscar y agendar una cita con un médico particular?

Busco en la web médicos de la ciudad con la especialidad requerida, acudo al edificio del consultorio para agendar.

8. ¿Cuándo has buscado atención medica particular, buscas un médico especialista o un médico general?

Trato de ir a un especialista.

9. ¿Has tenido algún problema al buscar un médico particular y agendar una cita, como crees que podría solucionar esos problemas en caso de haberlos?

Si, dentro de la ciudad no existen especialistas en algunas ramas en especialidades de niños menores a 6 años y en otros casos solo hay un solo especialista, teniendo una agenda llena por lo cual a agendar la cita es para una fecha a futuro; para solucionarlo se podría contar con una lista de doctores de la ciudad con su especialidad y lugares de trabajo en la web para poder consultarlo.

10. ¿Has usado alguna aplicación móvil o web para obtener una cita médica, comenta cual ha sido tu experiencia con ella, que es lo que más te ha gustado o disgustado, como crees que podría solucionar esos problemas en caso de haberlos?

Por el momento no he usado aplicativo alguno.

11. ¿Solicitarías un doctor a través de una aplicación móvil, crees que sería una forma eficiente para poder encontrar un médico y agendar una cita?

Sería de mucha ayuda, el poder contar con aplicativo que posea la información del doctor, sus estudios, referencias de pacientes; ayudando a agendar la cita de forma rápida y con varias opciones de médicos.

12. ¿Qué tareas o funcionalidades crees que debería tener la aplicación?

Búsqueda por lugar, por especialidad, por agenda libre dentro de la semana de búsqueda, referencias y puntuación al médico, costo de la consulta.

Entrevista 4:

1. Datos básicos del usuario

Edad: 29

Sexo: Masculino

Ocupación: Comerciante

Estado Civil: Soltero

Ciudad: Loja

No. y edad de hijos: 2 hijos, 8 y 3 años.

2. Conocer la experiencia que tiene el usuario manejando aplicaciones móviles

1. ¿Tienes aplicaciones móviles instaladas en tu celular, que tipo (juegos, redes, sociales, compras en línea, bancarias)?

Si cuento con aplicaciones para diferentes servicios.

2. ¿Qué tipos de aplicaciones son las que más usas?

Aplicaciones de servicios bancarios y redes sociales.

3. ¿En qué lugar usas más tu teléfono celular para revisar aplicaciones?

Al momento de despertar, el dormitorio, antes de empezar la jornada de trabajo.

4. ¿Qué sistema operativo usa tu celular?

Android.

3. Conocer el problema del usuario en cuanto a buscar un doctor particular y al agendar una cita médica

5. ¿Con que frecuencia acudes a un médico particular y porque razones has tenido que acudir?

Rara vez, por revisión rutinaria.

6. ¿Tienes algún familiar que acude a un médico particular, sabes porque razón acude al médico, sabes con qué frecuencia lo hace, sabes si ha tenido problemas al buscar un médico y agendar una consulta?



No tengo ningún familiar que acuda constantemente, pero en las raras veces que acuden, generalmente es por revisiones rutinarias.

7. ¿Cómo haces para buscar y agendar una cita con un médico particular?

Simplemente se acude al consultorio y se espera hasta que llegue el turno que lo atiendan.

8. ¿Cuándo has buscado atención medica particular, buscas un médico especialista o un médico general?

En las pocas veces que se ha acudido al médico, es al médico general.

9. ¿Has tenido algún problema al buscar un médico particular y agendar una cita, como crees que podría solucionar esos problemas en caso de haberlos?

Usualmente, no.

10. ¿Has usado alguna aplicación móvil o web para obtener una cita médica, comenta cual ha sido tu experiencia con ella, que es lo que más te ha gustado o disgustado, como crees que podría solucionar esos problemas en caso de haberlos?

No, jamás lo he intentado.

11. ¿Solicitarías un doctor a través de una aplicación móvil, crees que sería una forma eficiente para poder encontrar un médico y agendar una cita?

Talvez, pero en lo personal me gusta, o bien llamar a reservar, o ir directamente al consultorio, porque la mayoría de profesionales de la salud no revisan plataformas web para la agenda de turnos.

12. ¿Qué tareas o funcionalidades crees que debería tener la aplicación?

Reservar el turno, visualizando las citas reales con el especialista, así como el tiempo estimado en cada consulta.

Entrevista 5:

1. Datos básicos del usuario

Edad: 32 años

Sexo: Femenino

Ocupación: Empleada privada

Estado Civil: Casada

Ciudad: Loja

No. y edad de hijos: 2 hijas – 10 años y 1 año 7 meses

2. Conocer la experiencia que tiene el usuario manejando aplicaciones móviles

1. ¿Tienes aplicaciones móviles instaladas en tu celular, que tipo (juegos, redes, sociales, compras en línea, bancarias)?

SI, juegos, redes, sociales, compras en línea, bancarias, educativas.

2. ¿Qué tipos de aplicaciones son las que más usas?

Redes sociales, educativas y bancarias

3. ¿En qué lugar usas más tu teléfono celular para revisar aplicaciones?

En casa

4. ¿Qué sistema operativo usa tu celular?

Android

3. Conocer el problema del usuario en cuanto a buscar un doctor particular y al agendar una cita médica

5. ¿Con que frecuencia acudes a un médico particular y porque razones has tenido que acudir?

6 a 7 veces al año por control médico o enfermedad

6. ¿Tienes algún familiar que acude a un médico particular, sabes porque razón acude al médico, sabes con qué frecuencia lo hace, sabes si ha tenido problemas al buscar un médico y agendar una consulta?

6 a 7 veces al año y los inconvenientes es mas cuando se trata de un control a veces no es inmediato, sino después de unos 4 a 5 días de lo que se llama

7. ¿Cómo haces para buscar y agendar una cita con un médico particular?

Llamo al consultorio y lo agendo con su secretaria

8. ¿Cuándo has buscado atención medica particular, buscas un médico especialista o un médico general?

especialista

9. ¿Has tenido algún problema al buscar un médico particular y agendar una cita, como crees que podría solucionar esos problemas en caso de haberlos?

Problemas en si no los he tenido

10. ¿Has usado alguna aplicación móvil o web para obtener una cita médica, comenta cual ha sido tu experiencia con ella, que es lo que más te ha gustado o disgustado, como crees que podría solucionar esos problemas en caso de haberlos?

No he usado una aplicación movil



11. ¿Solicitarías un doctor a través de una aplicación móvil, crees que sería una forma eficiente para poder encontrar un médico y agendar una cita?

Si lo solicitaría

12. ¿Qué tareas o funcionalidades crees que debería tener la aplicación?

Facilidad, que no sea nada complejo, que muestre los días y horarios disponibles

Entrevista 6:

1. Datos básicos del usuario

Edad: 32

Sexo: Femenino

Ocupación: Sistemas

Estado Civil: Soltera

Ciudad: Loja

No. y edad de hijos:

2. Conocer la experiencia que tiene el usuario manejando aplicaciones móviles

1. ¿Tienes aplicaciones móviles instaladas en tu celular, que tipo (juegos, redes, sociales, compras en línea, bancarias)?

Juegos, redes sociales, compras, bancarias.

2. ¿Qué tipos de aplicaciones son las que más usas?

Bancarias, compras

3. ¿En qué lugar usas más tu teléfono celular para revisar aplicaciones?

Casa

4. ¿Qué sistema operativo usa tu celular?

IOS

3. Conocer el problema del usuario en cuanto a buscar un doctor particular y al agendar una cita médica

5. ¿Con que frecuencia acudes a un médico particular y porque razones has tenido que acudir?

Cada 3 meses, enfermedades



6. ¿Tienes algún familiar que acude a un médico particular, sabes porque razón acude al médico, sabes con qué frecuencia lo hace, sabes si ha tenido problemas al buscar un médico y agendar una consulta?

Cada tres meses, enfermedades

7. ¿Cómo haces para buscar y agendar una cita con un médico particular?

Llamando

8. ¿Cuándo has buscado atención medica particular, buscas un médico especialista o un médico general?

Especialista

9. ¿Has tenido algún problema al buscar un médico particular y agendar una cita, como crees que podría solucionar esos problemas en caso de haberlos?

Ser más organizados

10. ¿Has usado alguna aplicación móvil o web para obtener una cita médica, comenta cual ha sido tu experiencia con ella, que es lo que más te ha gustado o disgustado, como crees que podría solucionar esos problemas en caso de haberlos?

No ninguna.

11. ¿Solicitarías un doctor a través de una aplicación móvil, crees que sería una forma eficiente para poder encontrar un médico y agendar una cita?

Si solicitaría.

12. ¿Qué tareas o funcionalidades crees que debería tener la aplicación?

Citas médicas, historial clínico.

Entrevista 7:

1. Datos básicos del usuario

Edad: 30

Sexo: M

Ocupación: Desarrollador se Sistemas

Estado Civil: Soltero

Ciudad: Loja



No. y edad de hijos: Ninguno

2. Conocer la experiencia que tiene el usuario manejando aplicaciones móviles

1. ¿Tienes aplicaciones móviles instaladas en tu celular, que tipo (juegos, redes, sociales, compras en línea, bancarias)?

Juegos, sociales bancarias

2. ¿Qué tipos de aplicaciones son las que más usas?

Redes sociales

3. ¿En qué lugar usas más tu teléfono celular para revisar aplicaciones?

Play Store

4. ¿Qué sistema operativo usa tu celular?

Android

3. Conocer el problema del usuario en cuanto a buscar un doctor particular y al agendar una cita médica

5. ¿Con que frecuencia acudes a un médico particular y porque razones has tenido que acudir?

Por chequeo general, no he ido ya hace más de un año

6. ¿Tienes algún familiar que acude a un médico particular, sabes porque razón acude al médico, sabes con qué frecuencia lo hace, sabes si ha tenido problemas al buscar un médico y agendar una consulta?

Mi abuelita por problemas cardiacos, Mi abuelo por cáncer, el problema es que le dan una fecha muy extensa a la fecha actual

7. ¿Cómo haces para buscar y agendar una cita con un médico particular?

Ir al hospital y sacar historial médico para sacar una ficha

8. ¿Cuándo has buscado atención medica particular, buscas un médico especialista o un médico general?

Pues ambos dependen de lo que necesite.

9. ¿Has tenido algún problema al buscar un médico particular y agendar una cita, como crees que podría solucionar esos problemas en caso de haberlos?

El problema es que en veces no contestan o ya existe agendado muchas personas.

Se podría solucionar por medio de un sistema desde casa



10. ¿Has usado alguna aplicación móvil o web para obtener una cita médica, comenta cual ha sido tu experiencia con ella, que es lo que más te ha gustado o disgustado, como crees que podría solucionar esos problemas en caso de haberlos?

No, no he tenido la oportunidad de agendar de manera web

11. ¿Solicitarías un doctor a través de una aplicación móvil, crees que sería una forma eficiente para poder encontrar un médico y agendar una cita?

Si, si agendaría por una aplicación

12. ¿Qué tareas o funcionalidades crees que debería tener la aplicación?

Agendar, buscar alternativas de clínicas y de médicos, tener un historial de los médicos, debe tener un chat online para cualquier accidente nos sepa ayudar el doctor

Referencias Bibliográficas:

- [1] El teléfono móvil acapara 61 horas de tu vida cada semana, 21-01-2019, Tomado de: https://www.abc.es/tecnologia/redes/abci-telefono-movil-acapara-61-horas-vida-cada-semana-201901200150_noticia.html.
- [2] Apps más visuales y rápidas con Flutter. Tomado de: <https://www2.deloitte.com/es/es/pages/technology/articles/apps-mas-visuales-y-rapidas-con-flutter.html>
- [3] Indagación [análisis]. Tomado de: <http://cv.uoc.edu/webapps/xwiki/wiki/matm1202es/view/Main/6.1+Indagaci%C3%B3n+%5Ban%C3%A0lisi%5D>
- [4] Aplicación de pedido de comida: <https://zaymi-web.com/>
- [5] Aplicación de servicio de taxi: <https://ktaxi.com.ec/>
- [6] Tutorial de diagrama de clases UML. Tomado de: <https://www.lucidchart.com/pages/es/tutorial-de-diagrama-de-clases-uml>
- [7] Página oficial de Flutter; <https://flutter.dev/>
- [8] Página oficial de Digital Ocean: <https://www.digitalocean.com/>
- [9] Parseando JSON complejo en Flutter, 17-01-2019. Tomado de: <https://medium.com/@carlosAmillan/parseando-json-complejo-en-flutter-18d46c0eb045>
- [10] Spring Boot File Upload / Download Rest API Example, 18-04-2018. Tomado de: <https://www.callicoder.com/spring-boot-file-upload-download-rest-api-example/>
- [11] Flutter [Android] — Receiving Push Notifications + Sending from a Webapp, 14-04-2020. Tomado de: <https://medium.com/@fabiojansen86/flutter-android-receiving-push-notifications-sending-from-an-webapp-d2e43e19ba36>