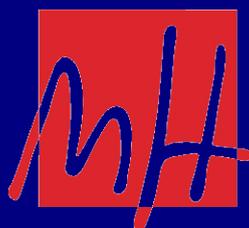


UNIVERSIDAD MIGUEL HERNÁNDEZ DE ELCHE

ESCUELA POLITÉCNICA SUPERIOR DE ELCHE

MÁSTER EN INGENIERÍA INDUSTRIAL



**UNIVERSITAS**  
*Miguel Hernández*

DISEÑO E IMPLEMENTACIÓN DE BASES DE  
DATOS REMOTAS PARA AUTOGESTIÓN DE  
SALUD: DIARIO DE SALUD DIGITAL PARA  
PERSONAS CON INMUNODEFICIENCIAS  
PRIMARIAS

TRABAJO FIN DE MÁSTER

Junio -2025

AUTOR: Daniel Alejandro Rodríguez López

DIRECTOR/ES: José María Sabater Navarro

Juan David Romero Ante





**UNIVERSIDAD MIGUEL HERNÁNDEZ DE ELCHE**

**ESCUELA POLITÉCNICA SUPERIOR DE ELCHE**

**GRUPO DE INVESTIGACIÓN EN NEUROINGENIERÍA  
BIOMÉDICA**



**DISEÑO E IMPLEMENTACIÓN DE BASES DE DATOS  
REMOTAS PARA AUTOGESTIÓN DE SALUD: DIARIO  
DE SALUD DIGITAL PARA PERSONAS CON  
INMUNODEFICIENCIAS PRIMARIAS**

**TRABAJO FIN DE MASTER**

Autor: Daniel Alejandro Rodríguez López

Director: José María Sabater Navarro

Codirector: Juan David Romero Ante

junio, 2025



## **Agradecimientos**

*A Dios, por concederme la serenidad para aceptar las cosas que no puedo cambiar, valor para cambiar las que puedo, y sabiduría para reconocer la diferencia.*

*A mi familia, especialmente a mis padres, que me apoyan en todo momento y confían en que puedo lograr lo que me propongo.*

*A Mariana, mi amor, por estar siempre a mi lado, animarme, aconsejarme y darme fuerza cuando más lo necesito.*

*A mis amigos, por su disposición a escucharme, por sus consejos y su compañía a lo largo de este camino.*

*A José María y a mi amigo Juan David, directores de este trabajo, por sus valiosas orientaciones, sugerencias y aportes durante todo el desarrollo del proyecto.*

*A los evaluadores, que con sus críticas constructivas incrementarán el alcance de este trabajo.*



*Daniel Alejandro Rodríguez López  
Elche, 2025*



**DISEÑO E IMPLEMENTACIÓN DE BASES DE DATOS REMOTAS PARA  
AUTOGESTIÓN DE SALUD: DIARIO DE SALUD DIGITAL PARA PERSONAS CON  
INMUNODEFICIENCIAS PRIMARIAS**

Daniel Alejandro Rodríguez López

Elche, junio 2025

**Resumen**

Las personas con inmunodeficiencias primarias (IDP) requieren de un seguimiento clínico constante para controlar sus tratamientos y compartir información con sus profesionales de salud de una manera organizada y accesible. Sin embargo, los sistemas de salud actuales no cuentan con herramientas adaptadas a las necesidades específicas de este tipo de pacientes. Este trabajo se enfoca en el diseño e implementación de una solución tecnológica para la autogestión de salud a través del uso de bases de datos remotas y el acceso desde dispositivos móviles.

El resultado del trabajo es una aplicación multiplataforma que funciona como un diario de salud electrónico, permitiendo a los usuarios registrar síntomas, resultados clínicos, tratamientos, vacunas, citas médicas y otros datos relevantes. Además, proporciona herramientas de consulta y comunicación con el equipo médico, fomentando un seguimiento más cercano y personalizado. Se desarrolló en Visual Studio, utilizando .NET MAUI y una arquitectura MVVM, con integración de Firebase Authentication y Firestore como base de datos. Además, incorpora procesamiento local de datos (*On the Edge*) para mejorar el rendimiento y reducir la dependencia con la nube.

Finalmente, se validó la experiencia de usuario mediante un cuestionario adaptado de la escala *uMARS*, dividido entre pacientes y profesionales. Los resultados reflejan una valoración positiva en funcionalidad, utilidad e impacto percibido, lo que sugiere que la herramienta tiene un gran potencial para mejorar la adherencia al tratamiento y la calidad de vida de las personas con IDP.

**Palabras clave:** telemedicina, aplicaciones móviles, inmunodeficiencias primarias, diario de salud, .NET MAUI, gestión de enfermedades crónicas, *uMARS*.



**DESIGN AND IMPLEMENTATION OF REMOTE DATABASES FOR HEALTH SELF-MANAGEMENT: A DIGITAL HEALTH JOURNAL FOR INDIVIDUALS WITH PRIMARY IMMUNODEFICIENCIES**

Daniel Alejandro Rodríguez López

Elche, jun. 2025

**Abstract**

*People with primary immunodeficiencies (PID) require continuous clinical monitoring to manage their treatments and share medical information with healthcare professionals in an organized and accessible manner. However, current healthcare systems often lack tools specifically tailored to the needs of this patient group. This project centers on designing and implementing a technological solution for health self-management by utilizing remote databases and mobile access.*

*The outcome is a cross-platform mobile application that serves as a digital health journal, allowing users to log symptoms, clinical results, treatments, vaccinations, medical appointments, and other relevant information. It also features options for data consultation and communication with healthcare providers, fostering more personalized and continuous follow-up. It was developed in Visual Studio using .NET MAUI and MVVM architecture, incorporating Firebase Authentication and Firestore integration. Additionally, it employs local data processing (On the Edge) to enhance performance and reduce dependence on clouds.*

*Finally, the user experience was validated through a questionnaire adapted from the uMARS scale, gathering responses from both patients and healthcare professionals. The results showed positive ratings in functionality, usefulness, and perceived impact, suggesting that this tool has significant potential to enhance treatment adherence and quality of life for individuals living with PID.*

**Keywords:** *mHealth, eHealth, telemedicine, mobile app, primary immunodeficiencies, health diary, .NET MAUI, chronic disease management, uMARS.*



## ***Índice general***

Agradecimientos .....	5
Resumen .....	7
Índice general .....	11
Índice de figuras.....	13
1. Introducción .....	15
1.1. Conceptos generales.....	15
1.1.1. Enfermedades no transmisibles .....	15
1.1.2. Inmunodeficiencias primarias (IDP) .....	17
1.2. Descripción de la problemática.....	19
1.3. Tecnologías para el seguimiento de inmunodeficiencias primarias .....	20
1.3.1. Dispositivos remotos .....	20
1.3.2. Aplicaciones móviles (mHealth) .....	24
1.3.3. Resumen y aprendizaje .....	28
1.4. Objetivos .....	29
2. Materiales y métodos .....	30
2.1. Base de datos .....	30
2.1.1. Tipo SQL .....	30
2.1.2. Tipo NoSQL.....	30
2.1.3. Firestore.....	31
2.2. Herramientas para gestión de datos “On the edge” .....	32
2.2.1. Firebase.....	33
2.3. Desarrollo de aplicaciones multiplataforma.....	33
2.3.1. .NET MAUI.....	34
2.3.2. Diseño de una aplicación multiplataforma con .NET MAUI .....	35
2.4. Integración de librerías .....	35
2.5. Autenticación de usuarios. ....	37
2.5.1. Firebase Authentication.....	37
3. Diseño de aplicación para personas afectas con inmunodeficiencias primarias.....	39
3.1. Proyecto IDPBook.....	39
3.2. Definición de la base de datos.....	40
3.3. Definición de la interfaz .....	44

3.3.1.	Descripción general de la interfaz de usuario .....	45
3.4.	Implementación en .NET MAUI .....	49
3.4.1.	Arquitectura MVVM.....	50
3.4.2.	Integración de librerías en .NET MAUI .....	52
3.4.3.	Configuración de la base de datos de Firestore .....	53
3.4.4.	Configuración de la gestión de usuarios.....	57
3.4.5.	Configuración de gráficos con DevExpressCharts.....	62
3.4.6.	Flujo de la aplicación.....	64
3.5.	Generación y publicación de la aplicación .....	65
3.5.1.	Android.....	65
3.5.2.	iOS .....	72
4.	Validación de la aplicación .....	77
4.1.	Manual de usuario.....	77
4.2.	Protocolo de pruebas para validación de la aplicación.....	77
4.2.1.	Objetivo .....	77
4.2.2.	Procedimiento .....	77
4.3.	Encuesta de experiencia de usuario.....	78
4.4.	Resultados .....	78
4.4.1.	Metodología .....	79
4.4.2.	Descripción .....	80
5.	Discusión .....	83
6.	Conclusiones y trabajos futuros .....	85
7.	Referencias .....	87
8.	Anexos .....	91
8.1.	Diseño de una aplicación multiplataforma con .NET MAUI .....	91
8.2.	Estructura de la base de datos remota de la aplicación IDPBook .....	101
8.3.	Manual IDPBook.....	111
8.4.	Escala de calificación uMARS .....	153
8.4.1.	Respuestas de los participantes.....	161

## Índice de figuras

Figura 1. Plataforma de detección SARS-CoV-2 RapidPlex [35].	21
Figura 2. Protocolo de comunicaciones del sistema Dyna-Vision. [36]	22
Figura 3. Esquema del sistema VeCare. [37].	23
Figura 4. Gráfico de funciones de la aplicación FlexIlg.	25
Figura 5. Gráfico de funciones de aplicación eMidCare.	26
Figura 6. Gráfico de funciones de aplicación hiCare Chronic.	27
Figura 7. Representación de datos NoSQL en Firestore.	31
Figura 8. Estructura de datos de Firestore.	32
Figura 9. Compatibilidad de .NET MAUI.	34
Figura 10. Arquitectura NuGet para el consumo de librerías.	36
Figura 11. Servicios de autenticación de Firebase.	37
Figura 12. Declaración de una clase con .NET MAUI.	42
Figura 13. Ejemplo de la estructura de datos definida para el proyecto.	42
Figura 14. Inicio de sesión y restablecimiento de contraseñas.	46
Figura 15. Controles y aspecto de ventana tipo principal.	47
Figura 16. Controles y aspecto de ventana tipo secundario.	47
Figura 17. Controles y aspecto en ventanas de tipo formulario.	48
Figura 18. Ventada de detalles del documento o formulario.	48
Figura 19. Ejemplo de gráficas evolutivas para pruebas de laboratorio.	49
Figura 20. Esquema del patrón de diseño MVVM.	50
Figura 21. Administrador de paquetes NuGet de Visual Studio.	52
Figura 22. Configuración de SDK para Android y Apple en Visual Studio.	54
Figura 23. Configuración del SDK de Firebase Admin.	54
Figura 24. Apartado de registro de usuarios.	60
Figura 25. Lista de usuarios de prueba registrados en Firebase.	61
Figura 26. Estructura de datos en Firestore.	61
Figura 27. Gráfico diseñado con la librería DevExpress Charts.	62
Figura 28. Flujo de navegación entre las páginas de la aplicación.	64
Figura 29. Configuración de paquete Android para distribución con Google Play.	66
Figura 30. Publicación del proyecto en Visual Studio.	67
Figura 31. Distribución de la aplicación en Visual Studio.	68
Figura 32. Firma del paquete de la aplicación.	68
Figura 33. Paquete AAB de la aplicación firmado.	69
Figura 34. Lanzamiento a producción.	70
Figura 35. Configuración finalizada con la aplicación en estado de revisión.	70
Figura 36. Aplicación IDPBook publicada en Google Play.	71
Figura 37. Instalación de certificado de distribución.	73
Figura 38. Configuración de la firma de lote de iOS.	74
Figura 39. Firma de paquete de la aplicación iOS.	75
Figura 40. Estado de revisión de IDPBook en App Store Connect.	76
Figura 41. Diagrama de cajas y bigotes con resultados de la validación.	82



## **1. Introducción**

En los años noventa el proyecto *Global Burden of Disease*, reveló que las enfermedades crónicas ya no eran más un problema exclusivo de países desarrollados [1]. El creciente impacto observado en regiones de bajos y medios recursos impulsó su reconocimiento como una epidemia global que amenazaba el desarrollo de cualquier país [2]. A partir del año 2000, gracias a la *Estrategia Global para la Prevención y el Control de las Enfermedades No Transmisibles* de la OMS, se han puesto en marcha un sinnúmero de proyectos para gestionar estas enfermedades, sin embargo, después de casi tres décadas de esfuerzos, los avances no son lo suficientes y los servicios de salud aún no cuentan con estrategias eficaces para la gestión integral de este tipo de patologías [2][3].

Gestionar este tipo de enfermedades representa un gran desafío. Requiere nuevos enfoques y soluciones prácticas como la tecnología digital, que puede ser un gran aliado en este proceso. Hoy en día existen aplicaciones móviles y dispositivos portátiles que permiten el seguimiento de síntomas, registrar tratamientos o recibir recordatorios [4]. Estas soluciones no solo buscan optimizar la gestión de datos, sino empoderar a los pacientes en la gestión de su propia salud mejorando aspectos como la adherencia a tratamientos, promover el conocimiento de su enfermedad, facilitar la comunicación con los profesionales a su cuidado, entre otros [5].

Este trabajo se enfoca en diseñar una guía práctica para el desarrollo de aplicaciones móviles fundamentadas en bases de datos remotas que incorporan mecanismos de anonimato y control de acceso acorde con el manejo responsable de información de salud. En concreto se presenta el desarrollo de una aplicación móvil, diseñada exclusivamente para personas con inmunodeficiencias primarias. Funciona como un diario de salud que permite registrar síntomas, analíticas, tratamientos, entre otras, permitiendo llevar un control de la mayoría de información clínica que el paciente recibe por parte de sus cuidadores. El objetivo de esta aplicación es ayudar al paciente a gestionar su salud con más autonomía y brindar a los profesionales médicos información útil para mejorar el seguimiento clínico.

### **1.1. Conceptos generales**

#### **1.1.1. Enfermedades no transmisibles**

Las enfermedades no transmisibles (ENT), también llamadas enfermedades crónicas, son un grupo de patologías no contagiosas que tienden a ser de larga duración y evolución lenta [7]. Entre las condiciones más comunes se encuentran las enfermedades cardiovasculares, cáncer, diabetes, inmunodeficiencias primarias. Las ENT son resultado de una combinación de factores genéticos,

fisiológicos y ambientales, siendo especialmente relevantes aquellos que ocurren durante las etapas críticas del desarrollo, como la vida prenatal y los primeros años de vida [7]. Entre las principales causas se encuentran los desequilibrios nutricionales, que abarcan tanto la desnutrición como la obesidad materna, la exposición a químicos ambientales y las mutaciones genéticas que afectan funciones metabólicas y celulares [8]. Estos factores, ya sea de forma aislada o en interacción directa, pueden incrementar la susceptibilidad a desarrollar ENT en etapas posteriores de la vida. Estas enfermedades requieren de un seguimiento constante principalmente para ajustar los tratamientos o evaluar el estado de la enfermedad, en etapas avanzadas cuando no es posible la recuperación completa del paciente es necesario brindar cuidados paliativos enfocados en el manejo del dolor, apoyo emocional y mejorar la calidad de vida del paciente.

Las ENT, aunque no tienen un carácter contagioso, son la principal causa de muerte en el mundo. Según datos de la Organización Mundial de la Salud (OMS), estas patologías son responsables de más de 41 millones de muertes al año, lo que equivale al 74 % de todas las muertes registradas mundialmente [9]. En la región europea, son responsables del 90% de todas las muertes. En España, para el año 2021 se reportó que el 84.4% de las muertes fueron a causa de las ENT [10].

Las ENT son más comunes de lo que se piensa, seguramente conoces a alguien cercano que padece alguna enfermedad cardiovascular, cáncer, enfermedad respiratoria crónica o diabetes, las cuales se clasifican como los cuatro principales tipos de ENT [7]. Otras patologías muy comunes como la obesidad, osteoporosis, enfermedades neurodegenerativas, inmunodeficiencias, también se clasifican como enfermedades crónicas, demostrando porque este grupo de ENT reporta tantas muertes en el mundo.

### ***Inmunodeficiencias***

Fuera de las cuatro principales enfermedades crónicas mencionadas previamente, las inmunodeficiencias constituyen un grupo relevante de patologías que afectan significativamente al sistema inmunológico. Estas condiciones predisponen a los individuos a infecciones recurrentes de difícil tratamiento y, en algunos casos, a desarrollar enfermedades autoinmunes [14]. Existen dos clases de inmunodeficiencias:

**1. Inmunodeficiencias primarias (IDP):** corresponden a desórdenes inmunológicos causados por síndromes genéticos. También se conocen como inmunodeficiencias congénitas ya que, en su mayoría, sino en su totalidad, son hereditarias [14].

**2. Inmunodeficiencias secundarias:** las inmunodeficiencias secundarias o adquiridas, se originan como consecuencia de factores externos, como la

administración de medicamentos inmunosupresores utilizados en el tratamiento de enfermedades como el VIH, la diabetes o el cáncer [14].

La distinción entre estos dos tipos de inmunodeficiencias es esencial para su diagnóstico y manejo clínico, especialmente en el contexto de tratamientos específicos y estrategias de prevención.

### **1.1.2. Inmunodeficiencias primarias (IDP)**

Las IDP constituyen un grupo de trastornos genéticos que afectan uno o varios componentes del sistema inmunológico desde el nacimiento, aunque algunos tipos no se manifiestan hasta la edad adulta [14][11]. Las IDP provocan una mayor predisposición a sufrir infecciones recurrentes o graves, siendo los pulmones, el sistema gastrointestinal, la piel, ojos y oídos las partes del cuerpo más afectadas. Actualmente, gracias a avances en la genética, se han descrito más de 400 trastornos relacionados con IDP [15], sin embargo, se desconoce la prevalencia exacta de las IDP en el mundo [14]. Se estima que hasta un 1-2 % de la población mundial podría verse afectada por algún tipo o variante de inmunodeficiencia primaria [13].

### **Clasificación de inmunodeficiencias primarias**

Las inmunodeficiencias primarias (IDP) pueden clasificarse de acuerdo con el componente específico del sistema inmune afectado o con las características inmunológicas y fenotípicas asociadas a cada trastorno. En 2022, el Comité de Clasificación de Inmunodeficiencias Primarias de la Unión Internacional de Sociedades de Inmunología (*IUIS* por sus siglas en inglés), presentó una clasificación actualizada que organiza estas afecciones en 10 categorías principales, basándose en características fenotípicas específicas [17]. Esta clasificación es altamente técnica, con múltiples subcategorías, y está diseñada para proporcionar a los profesionales médicos una herramienta detallada que facilite el diagnóstico preciso y la comprensión integral de estas patologías.

Sin embargo, dado que este trabajo está enfocado en un contexto tecnológico e ingenieril, no es necesario emplear una clasificación tan detallada y especializada. Por ello, la clasificación que se presenta a continuación adopta un enfoque más práctico e intuitivo, centrado en los componentes afectados del sistema inmunológico y en la identificación de los biomarcadores más relevantes, proporcionando una base útil para el diseño e implementación de herramientas tecnológicas orientadas al seguimiento y diagnóstico de estas enfermedades.

La siguiente clasificación está basada en el criterio del médico especialista en alergias e inmunodeficiencia, *James Fernández*, presentado en el manual sobre las generalidades de las inmunodeficiencias [18].

## 1. Deficiencias de la inmunidad Humoral:

Trastorno relacionado con defectos en los linfocitos B, un tipo de glóbulos blancos que producen anticuerpos (inmunoglobulinas) [18]. Cuando los niveles de estos anticuerpos son bajos el paciente está predispuesto a sufrir infecciones bacterianas. Representan el 50-60% de las IDP [24]. Entre los trastornos más comunes de esta categoría están:

- **Inmunodeficiencia Común Variable (IDCV):** Bajos niveles en IgG, IgA y/o IgM, respuesta alterada a vacunas.
- **Agammaglobulinemia ligada al X:** Falta de linfocitos B, gen *BTK* afectado.
- **Deficiencia Selectiva de IgA:** Falta de IgA, generalmente asintomática, pero con susceptibilidad a infecciones respiratorias y digestivas.

## 2. Deficiencias de la inmunidad Celular:

Esta condición está relacionada con defectos en los linfocitos T, un tipo de glóbulos blancos que ayudan a detectar y eliminar las células anómalas del cuerpo [25]. Niveles bajos de estos linfocitos predisponen al paciente a sufrir infecciones por virus, hongos u otros microorganismos oportunistas. Representan entre el 5-10% de las IDP [26]. Los trastornos más comunes de esta categoría son:

- **Síndrome de DiGeorge:** Deleción<sup>1</sup> en 22q11.2, hipoplasia tímica.
- **Candidiasis Mucocutánea Crónica:** Mutación en *STAT1* o *AIRE*, infecciones persistentes por *Cándida*.
- **Síndrome Linfoproliferativo Ligado al X:** Mutación en *SH2D1A* o *XIAP*, asociado a infecciones por *Epstein-Barr*.

## 3. Deficiencias combinadas:

Condición relacionada con deficiencias en la inmunidad celular y humoral simultáneamente, afectando los linfocitos B y T [27]. Estas suponen alrededor del 20% de las IDP [18]. Los trastornos más comunes son:

- **Inmunodeficiencia Combinada Grave (SCID):** Deficiencia de linfocitos T que produce alteraciones en la formación de anticuerpos.
- **Síndrome de Wiskott-Aldrich:** Mutación en el gen *WAS*, plaquetopenia y eczema.

## 4. Defectos en células fagocitarias:

Los fagocitos son células que ingieren (fagocitan) y eliminan los patógenos. Cuando estas células son afectadas, su capacidad para matar estos patógenos se deteriora [29]. Las infecciones cutáneas por estafilococos son características. Representan entre el 10-15 % de las IDP [18]. Algunos de los trastornos más comunes son:

---

<sup>1</sup> Una deleción, en genética, es una anomalía estructural cromosómica que consiste en la pérdida de un fragmento de ADN de un cromosoma. Deleción (genética) [en línea]. Wikipedia, Accedido diciembre 2024. Disponible en: [https://en.wikipedia.org/wiki/Deletion\\_\(genetics\)](https://en.wikipedia.org/wiki/Deletion_(genetics))

- **Enfermedad Granulomatosa Crónica:** Disfunción en la actividad fagocítica, con predisposición a infecciones bacterianas y fúngicas.
- **Neutropenia Cíclica:** Oscilaciones en la cantidad de neutrófilos, con predisposición a infecciones bacterianas recurrentes.

## **5. Deficiencias del Complemento:**

El sistema del complemento este compuesto por más de 30 proteínas que ayudan a los fagocitos a identificar y fagocitar microorganismos o células extrañas, además, tienen la capacidad de neutralizar bacterias o virus [28]. Las deficiencias del complemento alteran procesos inmunológicos, como la opsonización, fagocitosis y lisis de patógenos, dificultando la eliminación de microorganismos y complejos antígeno-anticuerpo, favoreciendo la aparición de infecciones recurrentes y trastornos autoinmunes en los pacientes [29]. Son patologías raras que representan menos del 2% de las IDP [18]. Los trastornos más comunes están relacionados con deficiencia de proteínas C1, C2, C3.

### **1.2. Descripción de la problemática**

Entre muchas de las limitantes que los sistemas de salud enfrentan hoy en día, la que más afecta a los pacientes con enfermedades crónicas es la falta de un programa de seguimiento continuo y personalizado [2]. Estos pacientes requieren de monitorización regular o adherencia estricta a los tratamientos y muchas veces se ven afectados por falta de personal médico, dificultad para desplazarse hasta los centros médicos o una comunicación limitada con los profesionales de la salud a su cuidado [5]. Esto no solo afecta la capacidad de ofrecer una atención de calidad, sino que también incrementa el riesgo de complicaciones prevenibles y hospitalizaciones innecesarias.

A pesar de los avances tecnológicos, muchos sistemas de salud aún carecen de herramientas efectivas para integrar la telemedicina y la tecnología móvil en el manejo de estas condiciones [5]. Muchas de las soluciones actuales no se ajustan a las necesidades reales de los pacientes. Una buena aplicación debe ser intuitiva y permitir el acceso desde cualquier lugar. Para quienes viven con enfermedades crónicas, estas características son esenciales. [5]. Además, las preocupaciones sobre la privacidad y seguridad de los datos continúan siendo barreras significativas para una implementación eficaz.

En el caso de patologías como las inmunodeficiencias primarias, debido a su naturaleza compleja y delicada, requieren un seguimiento constante de síntomas, pruebas de laboratorio y tratamientos especializados. Los pacientes deben gestionar grandes volúmenes de información, lo que refuerza la necesidad de una herramienta tecnológica diseñada para optimizar la organización y el intercambio de estos datos de manera eficiente y segura.

En este contexto, resulta indispensable explorar las tecnologías existentes para el seguimiento de enfermedades crónicas, como las inmunodeficiencias primarias, y evaluar su potencial para abordar estas limitaciones. En el próximo apartado se revisa algunas soluciones disponibles, identificando sus aportes, desafíos y casos de éxito en su implementación.

### **1.3. Tecnologías para el seguimiento de inmunodeficiencias primarias**

Se estima que la mitad de las personas que padecen enfermedades crónicas sufren de discapacidad en alguna etapa de su enfermedad, la cual puede prolongarse hasta por 6 meses, afectando sus actividades diarias y limitando su calidad de vida [34]. Esta discapacidad aumenta la dependencia de servicios sociales que muchas veces los sistemas de salud no pueden proveer. La integración de tecnologías remotas está transformando el acceso a la asistencia médica permitiendo que más pacientes puedan recibir una atención adecuada en el cuidado de sus enfermedades.

En esta sección se presentan las principales herramientas tecnológicas aplicadas en la medicina (*Telemedicine*), incluyendo aplicaciones móviles (*mHealth*), o dispositivos de monitoreo (*wearables*), que han impactado positivamente en el manejo de estas enfermedades al mejorar la comunicación, adherencia a tratamientos y la recopilación de datos clínicos.

#### **1.3.1. Dispositivos remotos**

En el contexto de las inmunodeficiencias primarias, los *wearables* actuales, aunque no desarrollados específicamente para estas patologías, destacan por su capacidad para medir múltiples señales biológicas. Estos sensores permiten registrar parámetros clave, como ritmo cardíaco, temperatura corporal u obtener un panel hematológico, que pueden asociarse a diversas enfermedades y aportar datos relevantes para el monitoreo indirecto del estado inmunológico o la identificación de complicaciones relacionadas.

La literatura revisada destaca algunos dispositivos remotos que poseen un potencial significativo para ser aplicados con inmunodeficiencias primarias. A continuación, se describen algunos de estos.

#### **1. RapidPlex**

Es una plataforma con tecnología electroquímica para la detección ultra rápida del SARS-CoV-2 (COVID-19). La característica que lo convierten en un potencial candidato para el uso en pacientes con IDPs, es la capacidad de detectar inmunoglobulinas específicas como IgG o IgM, biomarcadores que pueden brindar información clave sobre el estado inmunológico del paciente. Utiliza una



señal, se definen tres clases de riesgo, bajo, medio y alto. El sistema evalúa el estado del paciente casi en tiempo real y despliega alarmas o avisos según los datos analizados. Es importante resaltar que el *DVX Device* funciona como un modem entre la medición de los signos vitales y el servidor *DVX*. El registro de los signos vitales se realiza con dispositivos de terceros y el *DVX Device* los recibe vía Bluetooth, los almacena en una memoria SD y los transmite al servidor casi en tiempo real a través de una red móvil 3G/4G/LTE. En condiciones óptimas, el retraso en la transmisión de los datos es menor a un segundo. El sistema permite conectar múltiples *DVX Devices*, que representan múltiples pacientes, los dispositivos conectados se muestran simultáneamente en la interfaz web, permitiendo visualizar el estado de todos los pacientes y las alarmas generadas para cada nivel de riesgo [36].

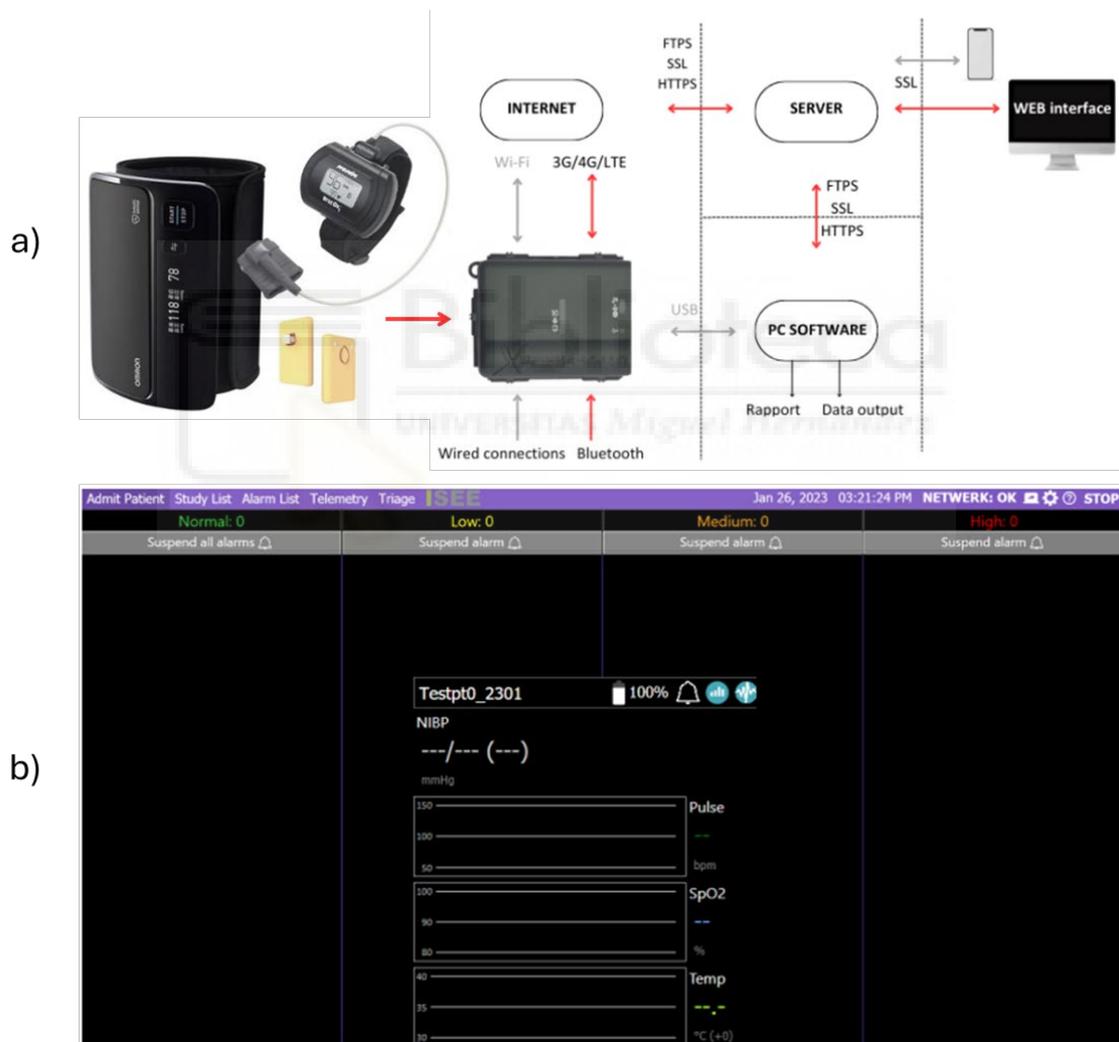


Figura 2. Protocolo de comunicaciones del sistema Dyna-Vision. [36]

### 3. VeCare

Se trata de un sistema para la detección de biomarcadores asociados con la inflamación y la carga microbiana en heridas crónicas como úlceras venosas. Como se puede observar en la Figura 3., el sensor cuenta con una tecnología flexible que se adhiere a las heridas y utiliza el exudado de estas para detectar estos biomarcadores. Tiene un sistema electrónico inalámbrico que transmite los datos a una aplicación que facilita el análisis y la gestión de la información en tiempo real. La interfaz gráfica y el análisis de datos esta desarrollado en Matlab. Esta tecnología permite gestionar perfiles de pacientes, recopilar datos de los sensores, visualizar la captura de datos en tiempo real y generar informes personalizados, además de registrarlos en los perfiles respectivos para un monitoreo a largo plazo. Este sistema pretende ayudar a la reducción de consultas presenciales y agilizar decisiones clínicas al proporcionar datos en tiempo real para mejorar la atención ambulatoria del paciente [37]. Aunque el dispositivo fue desarrollado principalmente para heridas crónicas, su capacidad para medir marcadores inflamatorios sugiere un potencial uso en el seguimiento de pacientes con inmunodeficiencias primarias, donde algunas de las manifestaciones clínicas más comunes son las infecciones en la piel, abscesos y candidiasis cutánea [6], que requieren un monitoreo igual que las úlceras venosas, objetivo principal de esta tecnología [37].



Figura 3. Esquema del sistema VeCare. [37]

### **1.3.2. Aplicaciones móviles (mHealth)**

Sin lugar a duda, los smartphones se han convertido en parte indispensable de nuestro día a día. A través de los años, estos dispositivos han reemplazado objetos como billeteras, libros impresos, cámaras, música en formato físico, entre otras, logrando compactar esto y mucho más en un solo dispositivo. La posibilidad de estar conectados en todo momento ha hecho que el uso de aplicaciones móviles crezca de forma acelerada. Esta tendencia también ha llegado al sector salud, donde cada vez más servicios integran estas herramientas para mejorar la atención médica. En 2014, cerca del 83 % de los servicios de salud en Estados Unidos, utilizaban smartphones y aplicaciones médicas para monitorear de manera remota las condiciones de sus pacientes [19], sin embargo, este tipo de aplicaciones estaban dirigidas para los proveedores de atención médica y requieren de un especialista para su uso adecuado [20]. Hoy en día, existen muchas aplicaciones enfocadas en el autocuidado por parte del paciente, buscando que el usuario sea capaz de gestionar su propia salud desde casa, con una supervisión remota de su médico de cabecera [21]. Este tipo de aplicaciones son claves para el cuidado de enfermedades no transmisibles como las inmunodeficiencias primarias, donde el seguimiento diario es fundamental para la gestión adecuada de este tipo de patologías. El número de aplicaciones diseñadas para la gestión de pacientes con IDP son limitadas. A continuación, se describen algunas de las más relevantes.

#### **1. Flexlg**

Es una aplicación diseñada para gestionar los tratamientos con inmunoglobulinas subcutáneas e intravenosas en pacientes con IDP o enfermedades crónicas autoinmunes e inflamatorias. La aplicación se lanzó en 2018 desarrollada por el laboratorio francés *Octapharma*, en colaboración con profesionales de la salud de organismos internacionales y franceses, así como asociaciones de pacientes que padecen IDP con el objetivo de ofrecer un producto diseñado específicamente a usuarios con este tipo de enfermedades. En 2024 se publicó un artículo que evalúa el impacto de la aplicación en la adherencia al tratamiento con inmunoglobulinas subcutáneas. Se trata de un estudio de dos años con 241 pacientes en Francia, donde la tasa de adherencia al tratamiento fue extremadamente alta, alcanzando el 99,7 %, y las dosis desperdiciadas representaron menos del 1 %. Además, se observó que altos niveles de satisfacción con la aplicación se correlacionaron positivamente con una mayor adherencia a los tratamientos [22].

El estudio concluye que las aplicaciones móviles como *Flexlg* son herramientas valiosas para mejorar la adherencia a tratamientos complejos, optimizando la educación y autogestión de los pacientes, además de reducir la necesidad de visitas hospitalarias [22].

Aunque los resultados son prometedores, los autores sugieren realizar estudios aleatorizados adicionales para validar estos hallazgos y explorar su implementación más amplia en el manejo de enfermedades crónicas [22]. La Figura 4, muestra algunos apartados de la aplicación *Flexlg*. Se pueden observar

los apartados de “Calendario”; donde se consignan las citas y recordatorios y “Mis dosis”; donde el paciente registra las tomas del tratamiento.



Figura 4. Gráfico de funciones de la aplicación Flexlg.

## 2. eMidCare

Citando textualmente la descripción encontrada en la página web de la aplicación, eMidCare es una “App para el seguimiento y la comunicación con pacientes diagnosticados de enfermedades inflamatorias mediadas por la inmunidad”. Es una aplicación desarrollada por la Sociedad Española de Farmacia Hospitalaria, diseñada para el Hospital General Universitario Gregorio Marañón en Madrid y su finalidad es apoyar a sus pacientes en el seguimiento domiciliario de su enfermedad y tratamientos. Está dirigida especialmente a pacientes diagnosticados de artritis reumatoide, psoriasis, artritis psoriásica, espondilitis anquilosante, enfermedad de Crohn o colitis ulcerosa. La aplicación le permite al paciente registrar las tomas de medicación, efectos secundarios, cuestionarios sobre calidad de vida, recordatorios y citas. Además, es posible contactar con los farmacéuticos del hospital para realizar consultas sobre su medicación. Un estudio publicado en 2022, donde participaron 85 pacientes del Hospital Gregorio Marañón, muestra que el 100% de los usuarios registraron su terapia biológica y el 26% usaron el módulo para registrar cada dosis administrada, que, relacionado con el nivel de satisfacción de la aplicación, con una puntuación media de 9,1/10, demostró gran potencial para mejorar la adherencia al tratamiento y la detección temprana de complicaciones [23]. El 32% de los pacientes registraron al menos un efecto adverso después de cada administración, entre los cuales la fatiga y las

reacciones en el sitio de inyección fueron los más comunes [23]. El módulo de mensajería fue el más utilizado, permitiendo a un 52 % de los pacientes comunicarse con sus farmacéuticos, principalmente para aclarar dudas sobre efectos adversos o preparación de medicamentos [23]. De acuerdo con el nivel de satisfacción y los porcentajes de uso de los distintos módulos de la aplicación, resulta importante reconocer el valioso aporte de esta herramienta para la gestión de enfermedades como las IDP ya que reflejan un gran potencial en la adherencia a tratamientos y el control de efectos secundarios respaldados por la comunicación rápida y asertiva de los profesionales a cargo. La Figura 5 muestra la página de inicio de sesión de la aplicación, además de la página de presentación, que en el contexto de programación de aplicaciones se conoce como “*Splash Screen*”. También se observa la sección de “Contacto”, donde el paciente realiza las llamadas con las diferentes dependencias clínicas.



Figura 5. Gráfico de funciones de aplicación eMidCare.

### 3. *hiCare Chronic*

Es una plataforma diseñada para el monitoreo remoto y la autogestión de pacientes con enfermedades crónicas. Esta incluye una aplicación móvil para pacientes y una plataforma web para profesionales de la salud, enfocándose en la recopilación de datos relacionados con síntomas, adherencia al tratamiento y calidad de vida. La aplicación permite sincronizar varios dispositivos que recolectan analíticas,

signos vitales o información sobre medicamentos, que posteriormente se utilizan para realizar análisis y reportes para el personal médico. Además, el usuario puede establecer umbrales para cada analítica o signos vitales que, cuando el paciente cruza estos valores, se generan alertas que informan tanto al paciente como al personal médico para resolver de la manera más rápida la emergencia detectada [38].

Un estudio publicado en 2024 donde participaron 150 pacientes reportó una mejora significativa en la adherencia al tratamiento, pasando del 72 % al 89 % tras tres meses usando la aplicación *hiCare Chronic* [38]. Además, los datos recopilados a través de la plataforma permitieron a los médicos ajustar el tratamiento en el 45 % de los casos, lo que resultó en una reducción del 30 % en hospitalizaciones relacionadas con la enfermedad [38]. Los pacientes calificaron la aplicación con una puntuación promedio de satisfacción de 8,8/10, destacando la facilidad de uso y la utilidad de las alertas personalizadas [38]. Gracias a todas sus funcionalidades demuestra ser una herramienta prometedoras para integrar el monitoreo remoto de enfermedades crónicas como las IDP, optimizando la comunicación, adherencia a tratamientos, resultados clínicos, además de reducir las hospitalizaciones gracias a la autogestión de los pacientes. En la Figura 6, se puede observar la interfaz gráfica de algunas secciones de la aplicación. Se puede observar *Splash Screen* de la app, el apartado de “Contactos” y la página principal donde se muestra la información del usuario y el resumen de algunas analíticas.



Figura 6. Gráfico de funciones de aplicación *hiCare Chronic*.

### **1.3.3. Resumen y aprendizaje**

En cuanto a los dispositivos remotos revisados, pese al impacto positivo de esta tecnología, su implementación presenta algunos desafíos. Entre algunas de estas limitaciones destacan el alto costo que pueden tener algunos dispositivos y la dependencia de una plataforma para gestionar los datos que recogen. La conectividad con servicios web es indispensable en este tipo de dispositivos, de lo contrario el personal médico a cargo del paciente no podría realizar un seguimiento preciso del mismo. Por otro lado, la implementación de aplicaciones móviles resulta mucho más versátil no solo por la no utilización de hardware, sino porque son más económicas de desarrollar y además están enfocadas en dispositivos que la mayoría o casi todo el mundo tiene al alcance de su mano. Si bien, algunas plataformas o aplicaciones necesitan de dispositivos adicionales para registrar datos, en el contexto médico, la gran mayoría de analíticas y signos vitales son provistos por los propios centros médicos o existen dispositivos especializados para ello y el paciente únicamente debe registrar dicha información para que la aplicación realice sus tareas y análisis correctamente. Además, considerando que el alcance de este trabajo está enfocado hacia pacientes con IDP, las analíticas que estas personas se realizan normalmente son de naturaleza compleja y únicamente son realizados en centros médicos especializados. Por esta razón, el esfuerzo de este trabajo se centra en desarrollar una aplicación móvil que utilice de manera óptima la información que el paciente ingresa en la app y no en como el paciente obtiene estos datos. Asimismo, de la revisión bibliográfica, se evidencia la escasez de aplicaciones diseñadas específicamente para pacientes con IDP, las pocas aplicaciones revisadas se han diseñado para un uso generalizado de enfermedades crónicas, sin embargo, las necesidades de un paciente con IDP pueden ser distintas y algunas veces más complejas que requieren de un servicio pensado especialmente para ellos. De acuerdo con lo anterior, resulta evidente la necesidad de desarrollar una aplicación que pueda apoyar al autocuidado de los pacientes con IDP, que permita una comunicación eficiente con sus médicos, que a su vez estos puedan acceder a la información de sus pacientes y así gestionar de manera óptima el estado de salud del usuario. Teniendo en cuenta la dirección del trabajo y basándose en las funciones y características identificadas de las tecnologías actuales, se establecen los siguientes objetivos.

#### **1.4. Objetivos**

Desarrollar y publicar una aplicación móvil multiplataforma cuya función sea la de diario de salud electrónico-diseñada para personas afectas con inmunodeficiencias primarias (IDP).

Mientras que los objetivos específicos son:

- Diseñar e implementar un sistema de gestión de datos basado en servicios en la nube.
- Integrar soluciones de bases de datos con tecnologías “*On the edge*” para optimizar el procesamiento local de datos.
- Desarrollar una aplicación móvil multiplataforma intuitiva y accesible.
- Realizar la validación funcional y de usabilidad de la aplicación a través de una evaluación basada en la experiencia del usuario.



## **2. Materiales y métodos**

En esta sección se detallan las necesidades y requisitos tecnológicos que forman parte de la aplicación, considerando las funcionalidades y características clave que debe ofrecer, orientadas a satisfacer la experiencia del usuario final.

### **2.1. Base de datos**

Una de las tareas más importantes en el desarrollo de una aplicación es la elección de una buena base de datos (BBDD) acorde con el volumen de información y seguridad que el sistema requiera. Existen varias clasificaciones de BBDD basadas en modelo de datos, tipo de almacenamiento, propósito o tipo de uso, nivel de acceso, distribución, entre otras. La clasificación por modelo de datos es la más referenciada, la cual describe a las BBDD SQL y NoSQL.

#### **2.1.1. Tipo SQL**

Son bases de datos basadas en un modelo relacional donde la información es almacenada en tablas que utiliza el lenguaje SQL para la manipulación de los datos [30]. Es un sistema muy robusto y seguro, muchos servicios bancarios y comercios online utilizan este tipo de BBDD por su fiabilidad. Su organización estructurada facilita la consulta y escritura de datos siempre que la cantidad de información no sea masiva. En aplicaciones donde el uso simultaneo no supere los 1.000 usuarios, este tipo de estructura es muy eficiente y no habría problemas con el desempeño de la aplicación.

#### **2.1.2. Tipo NoSQL**

Las bases de datos NoSQL están diseñadas para trabajar con datos no estructurados o semiestructurados, es decir, que la información no viene organizada en tablas como tradicionalmente se hace en SQL. Para la manipulación de los datos no existe un único lenguaje como en SQL, en este tipo de BBDD la programación y sintaxis dependen de cada modelo que por una parte lo hace muy flexible y adaptable a muchas situaciones, mientras que, por otro lado, puede representar cierta dificultad cuando se trata de operar con otras bases de datos ya que pueden existir conflictos en dicha interacción. Este tipo de BBDD son ideales para procesar grandes volúmenes de datos sin perder velocidad en dichas operaciones, incluso en tiempo real. Además, es compatible con formatos como JSON, BSON, XML, grafos, documental, entre otros, que lo convierte en una herramienta muy versátil e idónea para utilizar en aplicaciones móviles gracias a la compatibilidad de formatos que brinda.

Para el desarrollo de la aplicación se utilizó una base de datos NoSQL llamada Firestore que hace parte de la suite de Firebase. A continuación, se realiza una breve descripción de esta plataforma.

### 2.1.3. Firestore.

Firestore es una base de datos NoSQL documental, es decir, tiene una estructura de documentos y colecciones que facilita la organización y consulta de datos de manera jerárquica. Una analogía que facilita el entendimiento de esta estructura son los ficheros y documentos en un ordenador. Como se puede observar en la Figura 7, un fichero sería lo equivalente a una colección y los archivos que contiene ese fichero serían los documentos. Por su parte, la información contenida en estos archivos, representan los datos que utiliza la aplicación.

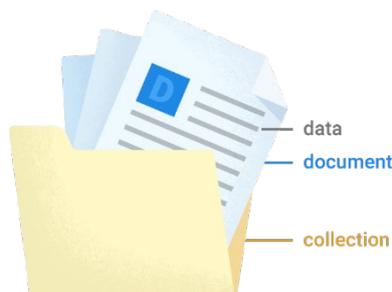


Figura 7. Representación de datos NoSQL en Firestore<sup>2</sup>.

La Figura 8 muestra un ejemplo de la jerarquía con la que funciona la base de datos de la aplicación. En esta se puede ver que cada usuario tiene su propia ficha de datos, almacenada como un documento único (*User1*, *User2*...). Estos documentos se agrupan en una colección (*Users*). La ventaja de esta estructura es que no todos los documentos necesitan tener los mismos campos. Es posible añadir un nuevo dato en un solo registro, sin tener que modificar los demás.

Con esta estructura la gestión de los datos es muy flexible e intuitiva, a diferencia de las bases de datos tradicionales. Además, su capacidad de lectura/escritura en tiempo real, posibilita la actualización automática de la información mostrada en la aplicación.

---

<sup>2</sup> Fuente de la imagen: Google Cloud, “Data model: Cloud Firestore,” [En línea]. Disponible: <https://cloud.google.com/firestore/docs/data-model> [Accedido: ene. 2025].

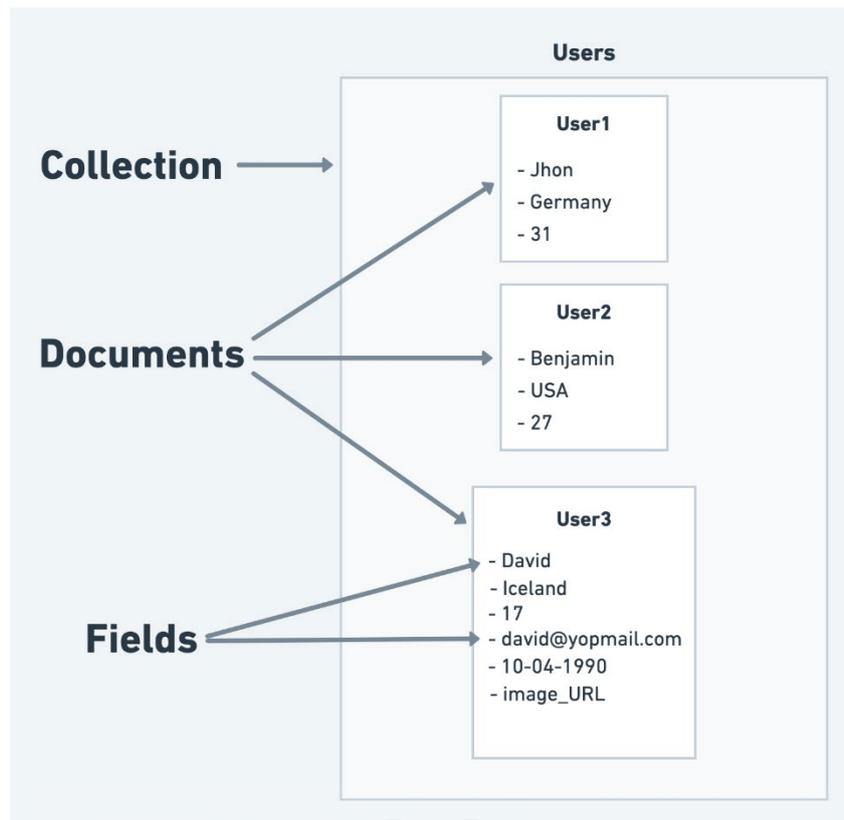


Figura 8. Estructura de datos de Firestore.<sup>3</sup>

## 2.2. Herramientas para gestión de datos “On the edge”

El término “On the edge” hace referencia a la ejecución de procesos, como almacenamiento o análisis de datos directamente en el dispositivo local, en lugar de depender exclusivamente de servidores en la nube. Esto se asocia con el concepto de *Edge Computing* que permite a las aplicaciones procesar datos más cerca de donde se generan, como en smartphones, tablets, dispositivos IoT, entre otros. El objetivo de estas herramientas es optimizar el uso de la base de datos, reducir las solicitudes innecesarias que ralentizan el sistema y afectan el desempeño de la aplicación. Mucho del procesamiento de datos es posible realizarlo localmente y existen muchas herramientas para implementar este tipo de procesamiento en el desarrollo de una aplicación y la elección depende del tipo de dispositivos al cual se dirige el producto, volumen de información que se busca gestionar, compatibilidad con otros servicios o módulos de la aplicación [31].

Plataformas como Microsoft Azure o Google Cloud, cuentan con servicios que extienden la funcionalidad de la nube al entorno local. Estos servicios permiten el

<sup>3</sup> Fuente de la imagen: FlutterFlow Docs, “Getting started with Cloud Firestore,” [En línea]. Disponible: <https://docs.flutterflow.io/integrations/database/cloud-firestore/getting-started/> [Accedido: ene. 2025].

procesamiento de datos en tiempo real antes de enviarlos a la base de datos correspondiente. Considerando que el ecosistema de Google ofrece mayor flexibilidad con otro tipo de servicios como el de autenticación y bases de datos, además de la compatibilidad con varios lenguajes de programación. Para el desarrollo de la aplicación se ha utilizado la plataforma de Firebase como herramienta de gestión de datos.

### **2.2.1. Firebase**

Firebase es la plataforma de desarrollo de aplicaciones de Google, y cuenta con varios servicios de compilación robustos y flexibles para adaptar con otras plataformas de desarrollo. En este caso se han integrado dos servicios fundamentales para la administración de la aplicación como son, la base de datos con *Firestore* y un servicio de gestión de usuarios con *Firebase Authentication*, de la cual se hablará más adelante en este capítulo.

Con el SDK de Firebase se implementaron tareas para la gestión de datos como:

- **Estructura de datos:** toda la información que es suministrada por el usuario se almacena con una estructura documental similar al formato JSON. Esta configuración se realiza localmente previo a la sincronización con la nube, de esta manera la BBDD no realiza ninguna operación de clasificación o estructuración que ralentice la sincronización de los datos.
- **Sincronización de datos:** Firestore ofrece sincronización en tiempo real y almacenamiento local en caché. Esto significa que los datos ingresados por los usuarios se almacenan localmente primero, permitiendo el acceso y uso de la aplicación incluso cuando no hay conexión a internet.
- **Gestión de usuarios:** Aunque Firebase Authentication depende de la nube para validar credenciales y autenticar usuarios, las sesiones de usuario y algunos datos de autenticación pueden gestionarse directamente en el dispositivo. Esto permite a los usuarios continuar utilizando ciertas funciones básicas de la aplicación sin una conexión constante a internet.

### **2.3. Desarrollo de aplicaciones multiplataforma**

El desarrollo multiplataforma permite crear aplicaciones que funcionan en varios sistemas operativos utilizando un solo código fuente, es decir, a partir de un solo proyecto se obtienen aplicaciones para dispositivos Android, iOS, Mac, Windows, entre otros. Esto trae consigo varias ventajas, siendo las dos más importantes la reducción de tiempo y recursos durante el desarrollo de la aplicación, y la flexibilidad para mantener y actualizar la aplicación a lo largo del tiempo. Lo anterior permite enfocar el desarrollo de la lógica e interfaz de la aplicación sin preocuparse por adaptar el código a diferentes sistemas operativos.

Plataformas de desarrollo multiplataforma como *Flutter*, *React Native*, *Xamarin*, *.NET MAUI* (la evolución de *Xamarin*), se han consolidado como alternativas modernas a los entornos de desarrollo tradicionales como XCode y Android Studio por las ventajas mencionadas anteriormente. Para este trabajo se utilizó la plataforma *.NET MAUI* de Visual Studio. A continuación, se realiza una breve descripción del sistema.

### 2.3.1. *.NET MAUI*

Es una plataforma de Microsoft que permite desarrollar aplicaciones multiplataforma utilizando *C#* y *.NET MAUI*. Es la evolución de *Xamarin.Forms*, que como se puede observar en la Figura 9, además de compatibilidad con Android y iOS, ofrece la posibilidad de crear aplicaciones para Windows y macOS, todo desde un mismo código. *.NET MAUI* ofrece acceso completo a las API nativas de cada plataforma, lo que permite crear aplicaciones con una apariencia y comportamiento nativos. Su integración con Visual Studio facilita el desarrollo de aplicaciones multiplataforma gracias a la integración de una amplia categoría de bibliotecas desarrolladas tanto por Microsoft como por terceros.

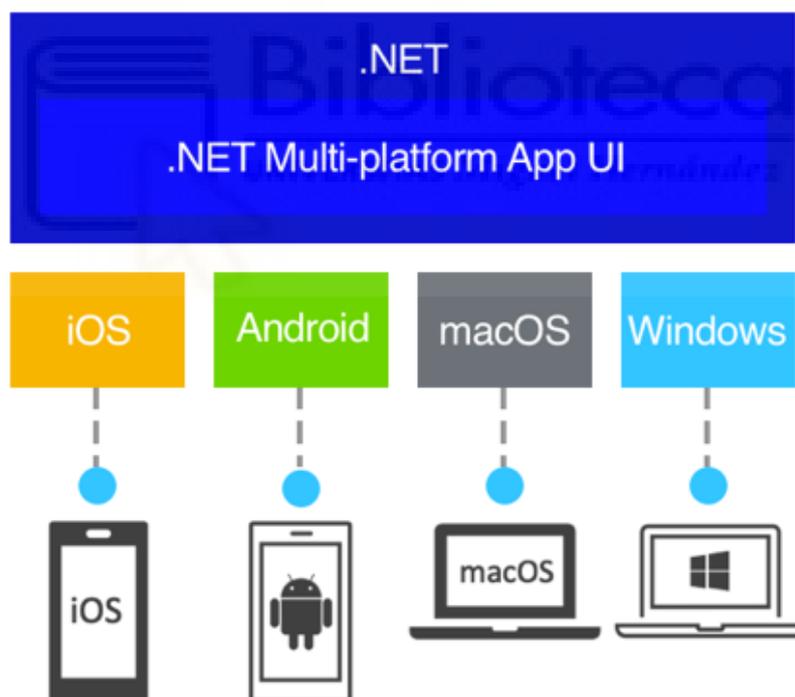


Figura 9. Compatibilidad de *.NET MAUI*<sup>4</sup>.

Durante el desarrollo de una app, MAUI permite depurar la aplicación tanto en emuladores como en dispositivos locales, únicamente es necesario instalar los

<sup>4</sup> Fuente de la imagen: Microsoft, ".NET MAUI documentation," [En línea]. Disponible: <https://learn.microsoft.com/en-us/dotnet/maui/?view=net-maui-8.0> [Accedido: ene. 2025].

SDK correspondientes a cada sistema operativo. Los requerimientos pueden variar según el tipo de ordenador que se utilice. A continuación, se describen los elementos más importantes por cada sistema operativo utilizando un ordenador Windows.

- **Android:** Para desplegar aplicaciones de Android en emuladores o dispositivos locales únicamente es necesario instalar los SDK compatibles con Windows y configurar el modo desarrollador del dispositivo.
- **iOS:** Para compilar aplicaciones en dispositivos locales iOS, es necesario instalar iTunes en el ordenador, contar con una cuenta *Apple Developer* y una *API Key* de desarrollador que permita utilizar dispositivos de *iOS* para desarrollar aplicaciones con plataformas distintas a *XCode*. Mientras que, para utilizar emuladores de *Apple*, es necesario conectarse a través de la red ethernet a un ordenador *Mac*, el proceso de emparejamiento con el ordenador es sencillo si se cuenta con todos los requerimientos mencionados anteriormente.

En el caso de este trabajo, el desarrollo se centró en dispositivos Android y iOS, pues son los sistemas operativos que más utilizan las personas hoy en día. Android tiene cerca del 72 % del mercado mundial. iOS, por su parte, tiene alrededor del 28 % de los clientes [32].

### **2.3.2. Diseño de una aplicación multiplataforma con .NET MAUI**

El proceso de diseño puede ser extenso y dependerá de la cantidad de funcionalidades que se quieran añadir a la aplicación. Sin embargo, el procedimiento general para crear una primera aplicación es sencillo. Visual Studio cuenta con guías paso a paso que ayudan a implementar una aplicación sencilla que sirva como punto de partida de una app mucho más completa. Además de estas guías, cuenta con una herramienta llamada *IntelliSense*, una clase de autocorrección de código, muy útil para principiantes en C# o XAML.

Para comprender de mejor manera el proceso de diseño de una aplicación multiplataforma con .NET MAUI, se ha dispuesto un anexo a modo de tutorial donde se explica detalladamente como crear una aplicación para dispositivos Android y iOS utilizando Visual Studio en Windows. Este material se encuentra en el anexo 8.1.

## **2.4. Integración de librerías**

Las librerías son códigos reutilizables indispensables en el desarrollo de aplicaciones, son herramientas que agilizan la integración de funcionalidades, muchas veces desarrolladas por terceros, donde, normalmente los autores construyen paquetes con bibliotecas, documentación y dependencias que

permiten trabajar de manera más eficiente. En el entorno de .NET, el gestor de paquetes/librerías es *NuGet*. En Visual Studio, *NuGet* permite instalar, actualizar y administrar las librerías de nuestros proyectos a través de línea de comandos o mediante el administrador de paquetes *NuGet* que viene integrado en el programa.

*NuGet* actúa como un host público que mantiene un repositorio con más de 100,000 paquetes disponibles. Además, permite hospedar paquetes de forma privada en la nube, o sistemas locales, ofreciendo control sobre quién puede acceder a ellos. Cuando se descarga un paquete para integrarse a un proyecto, se instala una copia de toda la documentación del paquete y se incluyen las API correspondientes para desarrollar el código de la función deseada. En la Figura 10 se puede observar el esquema de funcionamiento de *NuGet* descrito anteriormente.

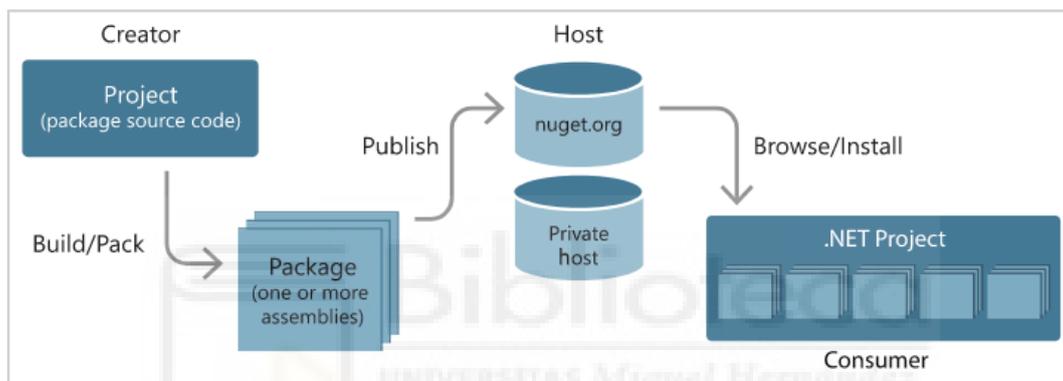


Figura 10. Arquitectura NuGet para el consumo de librerías.<sup>5</sup>

NuGet también garantiza que las plataformas de destino admitidas por el paquete son compatibles con el proyecto. Si el paquete no contiene ensamblados compatibles, NuGet muestra un mensaje de error y no se completa la instalación. La causa más común de incompatibilidad son las versiones de .NET, muchas veces el paquete contiene ensamblados compilados para una versión menor que la instalada en el proyecto de destino y no es posible hacer uso de la librería. La solución más práctica es buscar un paquete de otro creador más actualizada, sin embargo, es posible regresar a una versión de .NET compatible con el paquete, pero no es recomendable sobre todo por el mantenimiento y actualización de la aplicación a largo plazo, pues versiones de .NET anteriores a 7.0 ya no reciben soporte.

Durante el desarrollo de la app se utilizaron paquetes para la implementación de la base de datos con Firestore, las herramientas para gestión de datos “*On the edge*” con Google Firebase, estrategias de programación con patrones MVVM, manejo de

<sup>5</sup> Fuente de la imagen: Microsoft, “¿Qué es NuGet?,” [En línea]. Disponible: <https://learn.microsoft.com/es-es/nuget/what-is-nuget> [Accedido: ene. 2025].

archivos JSON, gestión de usuarios con Firebase Authentication y, por último, el paquete para el uso de gráficos en .NET MAUI. En el capítulo 3, se describirán los procesos de configuración y uso de estos paquetes.

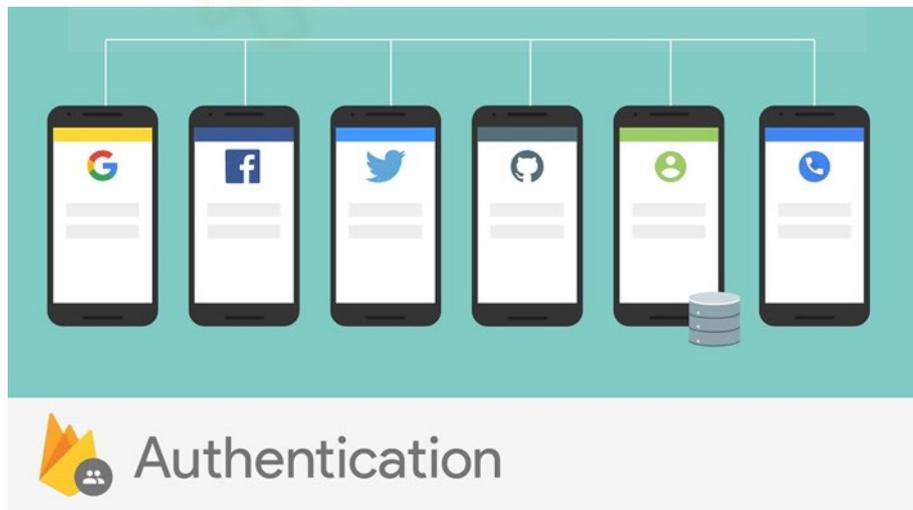
## **2.5. Autenticación de usuarios.**

La gestión de usuarios resulta fundamental en aplicaciones que buscan brindar algún tipo de personalización. Permite controlar el tipo de usuarios que acceden a la aplicación, configurar el nivel de acceso según permisos otorgados, crear o eliminar usuarios, entre otras funciones.

Para este cometido existen varios tipos de autenticación de usuarios y estos dependen de la herramienta utilizada para dicha implementación. En este caso se ha utilizado el servicio de *Firebase Authentication*, aprovechando la compatibilidad con la base de datos utilizada y la gestión “*On the edge*” implementada con la suite de *Google Firebase*.

### **2.5.1. Firebase Authentication**

*Authentication*, es el kit de desarrollo para la gestión de usuarios de Firebase, este permite desarrollar varios métodos de autenticación, como, correo electrónico y contraseña, número de teléfono, proveedores de identidad como Google, Facebook, Twitter (ahora X), Apple, GitHub o Microsoft, tal como se observa en la Figura 11.



*Figura 11. Servicios de autenticación de Firebase<sup>6</sup>.*

Uno de los métodos de autenticación más versátiles es a través de correo electrónico y contraseña, ya que permite tener más control sobre la experiencia de

---

<sup>6</sup> Fuente de la imagen: Google Firebase, “Autenticación,” [En línea]. Disponible: <https://firebase.google.com/docs/auth?hl=es-419> [Accedido: ene. 2025].

acceso, garantizando la seguridad de los datos del usuario. El SDK o kit de desarrollo permite una fácil integración de este servicio en la aplicación, además ofrece varias funciones de seguridad como la verificación de usuarios por correo electrónico, detección de intentos de inicio de sesión no autorizados, recuperación de contraseña mediante correo electrónico, entre otras. En apartados posteriores se describe el proceso de configuración y uso de esta herramienta.



### **3. Diseño de aplicación para personas afectas con inmunodeficiencias primarias.**

Actualmente, la gestión activa y el seguimiento continuo de enfermedades crónicas, como las inmunodeficiencias primarias (IDP), no se encuentran entre las principales prioridades de los sistemas de salud [2]. Esto se debe, en gran parte, a la complejidad inherente de estas patologías, la baja prevalencia en comparación con otras enfermedades crónicas y la falta de recursos dedicados exclusivamente a su manejo integral [5]. Adicionalmente, la revisión bibliográfica realizada en este trabajo ha puesto en evidencia la falta de soluciones tecnológicas especializadas y adaptadas a las necesidades de las personas afectadas por IDP. Aunque existen herramientas generales para la gestión de enfermedades crónicas, estas no suelen considerar las particularidades clínicas de los pacientes con IDP, como el monitoreo de infecciones recurrentes, la adherencia a tratamientos inmunológicos o el registro de episodios adversos. Esta falta de enfoque en las IDP resalta la necesidad de desarrollar tecnologías innovadoras que permitan no solo un seguimiento personalizado, sino también promover la autogestión de los pacientes, los empodere en el cuidado de su propia salud y les permitan mejorar su calidad de vida desde la comodidad de sus hogares.

#### **3.1. Proyecto IDPBook**

Este trabajo se enmarca en el Programa UMH-La Fe 2021, una iniciativa de ayudas para promover acciones preparatorias conjuntas entre el personal investigador de la Universidad Miguel Hernández (UMH) y los investigadores y profesionales del Hospital Universitari i Politècnic/Instituto de Investigación Sanitaria La Fe (IIS La Fe). Este programa forma parte de los esfuerzos de desarrollo e investigación respaldados por la Fundación para el Fomento de la Investigación Sanitaria y Biomédica de la Comunitat Valenciana (FISABIO), con el objetivo de fomentar la colaboración interdisciplinar y desarrollar soluciones innovadoras en el ámbito sanitario.

La idea de este proyecto nace en el IIS La Fe como respuesta a la necesidad de herramientas tecnológicas diseñadas específicamente para la autogestión de pacientes con inmunodeficiencias primarias. Estas patologías, debido a su naturaleza crónica, generan una considerable carga económica para los sistemas de salud, principalmente por el alto uso de servicios médicos, hospitalizaciones frecuentes y la necesidad constante de atención médica especializada. En este contexto, la aplicación tiene como objetivo optimizar la gestión de las asistencias no urgentes, reduciendo los costes derivados de visitas hospitalarias repetitivas y contribuyendo a disminuir el riesgo de infecciones intrahospitalarias en estos pacientes inmunocomprometidos.

La solución se fundamenta en la creciente adopción de las *eHealth*, un sector clave para transformar los servicios de salud tradicionales mediante la tecnología. Inspirados en el éxito de aplicaciones destinadas a la autogestión de pacientes con patologías crónicas como diabetes, insuficiencia cardíaca y otras enfermedades, los investigadores buscaban una plataforma que ofreciera funcionalidades específicas para los pacientes con IDP. Entre estas se incluyen: recordatorios de citas médicas, alertas sobre dosis de medicamentos, acceso a información actualizada sobre su enfermedad y una vía directa de comunicación con los servicios sanitarios y asociaciones de pacientes.

Para materializar este proyecto, el IIS La Fe lanzó una convocatoria con el propósito de identificar un grupo de investigación capacitado para desarrollar una solución adaptada a las necesidades específicas de los pacientes con IDP. Tras un proceso de selección, el proyecto fue concedido al grupo Medical Robotics Unit, grupo de investigación al que pertenezco desde 2022 y que forma parte del Instituto de Bioingeniería de la Universidad Miguel Hernández. Esta adjudicación fue posible gracias a la experiencia del grupo en el desarrollo de aplicaciones móviles y soluciones tecnológicas aplicadas al ámbito biomédico. La colaboración con el IIS La Fe permitió establecer un marco interdisciplinario que combina experiencia clínica y tecnológica, asegurando el desarrollo de una aplicación eficaz y centrada en las necesidades reales de los pacientes y los profesionales sanitarios.

Este proyecto pretende marcar un hito siendo la primera aplicación móvil diseñada exclusivamente para pacientes con inmunodeficiencias primarias a nivel mundial, posicionándose como una herramienta integral para empoderar al paciente, mejorar su calidad de vida y optimizar el impacto socioeconómico derivado del alto uso de los servicios sanitarios por parte de este tipo de pacientes.

Como resultado de una primera fase de trabajo, se identificaron y definieron las variables de interés para el seguimiento de los pacientes, así como los requisitos técnicos, funcionales y visuales para la implementación de la aplicación. Este proceso permitió concretar tanto la arquitectura de datos como el diseño de la interfaz adaptada a las necesidades específicas de los pacientes y del personal sanitario. En los siguientes apartados se detallan aspectos claves sobre la construcción de la base de datos e interfaz de usuario.

### **3.2. Definición de la base de datos**

La base de datos se compone de varios conjuntos de variables, en este contexto cada conjunto se denomina modelo o clase, y cada modelo se compone de muchas variables también llamadas propiedades. En primer lugar, se estableció el número de apartados o áreas con las que la aplicación debería contar y en función de esta organización asignar las variables para cada una. Con base en esta

distribución de apartados se definieron nueve clases que almacenan todas las variables del sistema.

- **Admin.cs:** clase asociada a los posibles efectos secundarios debido a administraciones de medicamentos como las inmunoglobulinas.
- **Analitica.cs:** gestiona las pruebas de laboratorio o analíticas que el paciente registra en la aplicación.
- **Cuestionario.cs:** correspondiente a la información solicitada en el cuestionario de calidad de vida.
- **Episodio.cs:** gestiona todas las variables asociadas a los episodios que el paciente registra en la aplicación.
- **Historia.cs:** clase asociada a las historias clínicas que el personal médico registra para cada paciente.
- **OtroTrat.cs:** define aquellos tratamientos que no están relacionados con inmunoglobulinas, es decir, medicamentos que no se vinculan directamente con el abordaje de inmunodeficiencias.
- **Paciente.cs:** con esta clase se gestionan los datos personales del paciente, así como diagnósticos preexistentes.
- **Tratamiento.cs:** clase que define los tratamientos con inmunoglobulinas, pueden ser subcutáneas o intravenosas
- **Vacuna.cs:** gestiona todos los datos asociados a las vacunas que el paciente registra en la aplicación.

La Figura 12 muestra un ejemplo de la declaración de una clase en el entorno de programación de .NET MAUI y su representación en la base de datos de Firestore. La imagen describe la definición de la clase “*Vacuna*” (12a) con sus propiedades y el tipo de cada variable, en este caso todos son “*string*” o texto plano. Se puede observar que se utiliza una definición en formato JSON correspondiente con una base de datos NoSQL. Por otro lado, (12b) se muestra un registro de la clase “*Vacuna*” en la base de datos Firestore. Aquí se puede ver los valores de tipo string de cada propiedad.

Todas las variables definidas y la estructura final de la base de datos se encuentran en el anexo 8.2.

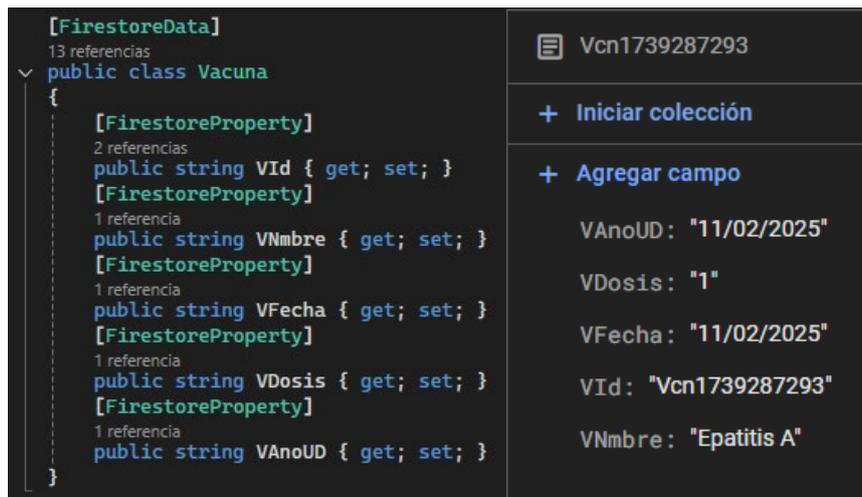


Figura 12. Declaración de una clase con .NET MAUI.

### Estructura de la base de datos en Firestore

En cuanto a la organización de los datos, como se mencionaba en el apartado 2.1.3, corresponde a una base de datos de tipo NoSQL. Esta estructura permite organizarla información de manera flexible y eficiente, facilitando la gestión y escalabilidad de la base de datos. La Figura 13, muestra un ejemplo de la distribución de los datos en Firestore. El esquema pretende mostrar la diferencia entre colecciones (color gris) y documentos (color azul) y como se relacionan a efectos de la construcción de rutas para la gestión de la información en la base de datos.

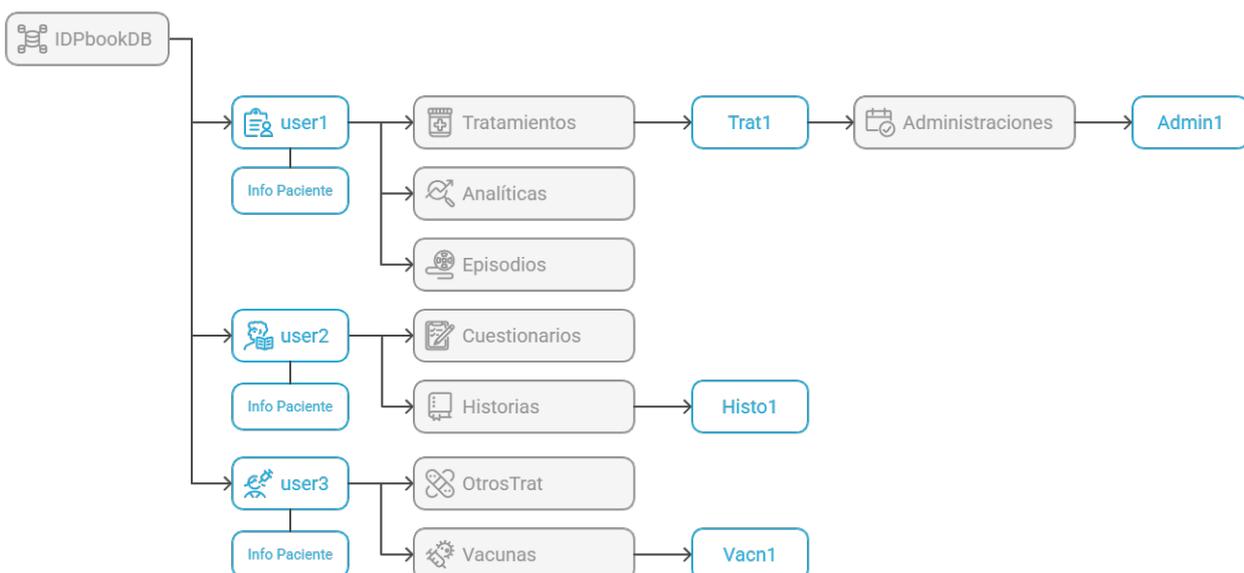


Figura 13. Ejemplo de la estructura de datos definida para el proyecto.

## Identificadores

Para realizar cualquier acción con la base de datos se utiliza los “*path*” o rutas de acceso. Estas rutas son construidas con identificadores (en adelante *id*) que son únicos para cada usuario o registro dentro de la base de datos. Los *id* de los usuarios se asignan de acuerdo con el UID generado por Firebase Authentication al momento de registrar la persona en el sistema. Este UID corresponde con un *id* único dentro del proyecto de Firebase y el cual permite controlar el acceso a la aplicación y gestionar las credenciales del usuario. De esta manera, cada vez que un paciente inicia sesión en la aplicación, se obtiene este UID de sus credenciales y se actualiza la ruta de acceso a la base de datos de la siguiente forma:

*/IDPBookDB/UID\_usuario/...*

En cuanto a los *id* de los documentos registrados, estos se construyen con una clave y un *timestamp*. La clave corresponde con una abreviación del tipo de variable y la marca de tiempo es una representación basada en el conteo de segundos transcurridos desde el 1 de enero de 1970 a las 00:00:00 UTC o también llamado *Unix Epoch Time*. Es un formato simple construido únicamente con números, sin ningún carácter especial o letras. Es único porque el tiempo transcurrido entre un registro y otro nunca es igual. Además, funciona sin importar la ubicación geográfica o el uso horario que utiliza el móvil. A continuación, se muestra un ejemplo de este tipo de identificador.

*/IDPBookDB/UID\_usuario/Episodios/Ep1739288614*

Donde, *Ep1739288614*, representa un episodio registrado el martes 11 de febrero del 2025 a las 16:43:34 GMT +01. Los *id* para cada tipo de variables son:

- Administraciones: “*Admin*” + timestamp.
- Analíticas: “*Anltc*” + timestamp
- Cuestionarios calidad de vida: “*Cuest*” + timestamp
- Episodios: “*Ep*” + timestamp
- Historias: “*Histo*” + timestamp
- Otros tratamientos: “*OTrat*” + timestamp
- Tratamientos: “*ITrat*” + timestamp
- Vacunas: “*Vcn*” + timestamp

Las rutas siempre tendrán una estructura similar. Por ejemplo, basándose en la Figura 13, para consultar el historial número 1 del paciente “*user2*”, la ruta deberá ser la siguiente:

*/IDPBookDB/user2/Historias/Histo1*

De esta manera, para consultar cualquier tipo de información en la base de datos, se debe modificar los identificadores (azul) y el *tipo de variables* que se quiere consultar. Con esta estructura la gestión se realiza de una manera muy intuitiva y rápida. Cada apartado de la aplicación modifica el *tipo de variables* y utiliza las credenciales del paciente para modificar el identificador del usuario en cada consulta o registro.

Todas las rutas de acceso contienen los mismos elementos a excepción de los documentos relacionados con las administraciones de un tratamiento con inmunoglobulinas. La ruta de estos documentos es más extensa ya que las administraciones se registran por cada tratamiento. Por ejemplo, basándose en la Figura 13, la ruta para acceder a los efectos secundarios registrados para la administración 1 del tratamiento “*Trat1*”, corresponde a la siguiente estructura.

*/IDPBookDB/user1/Tratamientos/Trat1/Administraciones/Admin1*

Extensión para gestionar administraciones

### 3.3. Definición de la interfaz

El número de apartados principales definido en la primera fase de trabajo fueron 6 de acuerdo con la siguiente organización:

1. **Área de registro e información personal:** En este apartado se asocian los datos personales del paciente e información de diagnóstico. La información es registrada por el personal médico, aunque puede ser editada por el paciente.
2. **Estado de la enfermedad:** Es el apartado con mayor cantidad de variables ya que se compone de varios subapartados como, Episodios, Pruebas de laboratorio, Tratamientos, Vacunación, entre otros.
3. **Próximas visitas:** Apartado donde se registran las citas que el paciente programa con los distintos servicios del hospital. Esta información es sincronizada con Google Calendar para generar recordatorios que mantengan al paciente informado sobre dichos eventos.
4. **Bibliografía y documentación:** Brinda acceso a un servidor en Google Drive donde se puede encontrar información (documentos, artículos, videos) relacionada con su enfermedad.
5. **Contacto sanitario:** Área donde se gestionan las comunicaciones con la unidad de Inmunodeficiencias Primarias del IIS La Fe, se permiten solicitar videollamadas y enviar correos electrónicos.

**6. Redes de pacientes:** En esta sección se encuentran enlaces de organizaciones nacionales e internacionales de interés para los pacientes. También se encuentra un enlace para acceder a un grupo de Telegram creado exclusivamente para pacientes de la Unidad de Inmunodeficiencias Primarias del IIS la Fe.

Además de estos apartados principales se incluyeron secciones básicas de una app como una página de inicio de sesión, cambio de contraseña y gestión de usuarios, entre otras.

### 3.3.1. Descripción general de la interfaz de usuario

Para el diseño de la interfaz se utilizó la siguiente paleta de colores:

#### 1. Color Primario

- Azul (#2E86C1)

#### 2. Colores Secundarios

- Verde (#28B463)
- Gris claro (#F2F4F4)

#### 3. Colores de Acento

- Naranja (#E67E22)
- Amarillo (#F4D03F)

#### 4. Colores de Fondo y Texto

- Blanco (FFFFFF)
- Gris oscuro (#5D6D7E)



La aplicación ha sido desarrollada para que el usuario tenga una experiencia agradable e intuitiva, sin complejas instrucciones o gran cantidad de controles. En las siguientes capturas se realiza la descripción de los elementos más importantes de la interfaz de usuario.

Para empezar, en la Figura 14, se muestra los apartados de inicio de sesión y restablecimiento de contraseña. El proceso de restablecimiento varía de acuerdo con el tipo de usuario, si es de tipo paciente, la contraseña se permite actualizar directamente en la aplicación, mientras que, si se trata del personal médico o administrador de la aplicación, se enviará un correo con las instrucciones para cambiar la contraseña a través de un enlace que gestiona la propia plataforma de Firebase.



Figura 14. Inicio de sesión y restablecimiento de contraseñas.

Dejando de lado los apartados de la Figura 14, la aplicación incluye cuatro tipos de ventanas con las siguientes funciones:

- **Ventanas principales** (Figura 15): agrupan las secciones principales y ofrecen acceso directo a las funciones secundarias.
- **Ventanas secundarias** (Figura 16): permiten explorar funciones o contenidos específicos dentro de cada apartado, manteniendo la jerarquía visual y funcional.
- **Formularios** (Figura 17): están diseñadas para el registro de la información por parte del usuario o del personal sanitario. Los campos cambian según el apartado al que se haya accedido.
- **Ventanas de detalle** (Figura 18): muestran información específica como resultados individuales o evolución de parámetros registrados.



Figura 15. Controles y aspecto de ventana tipo principal.

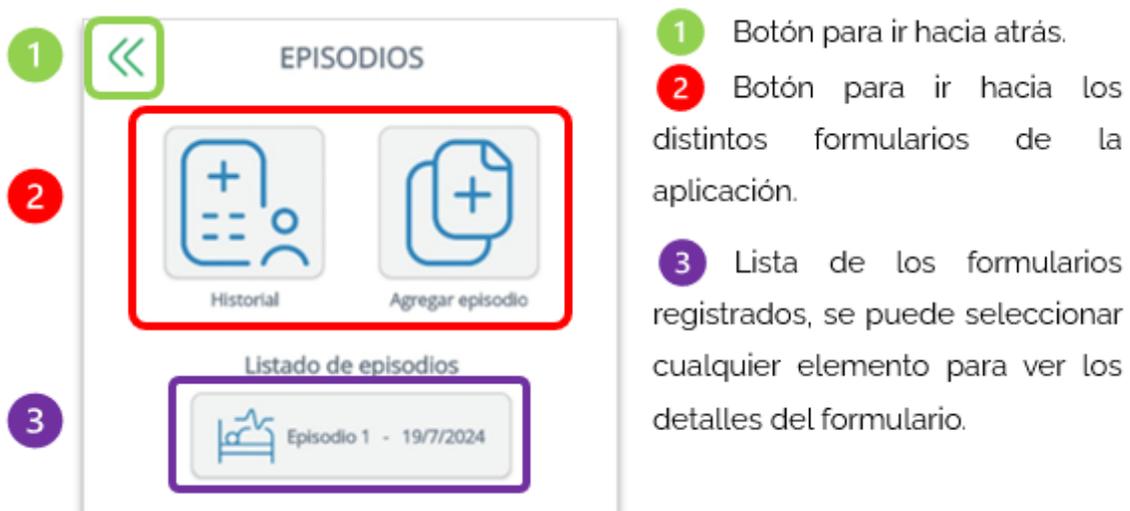


Figura 16. Controles y aspecto de ventana tipo secundario.

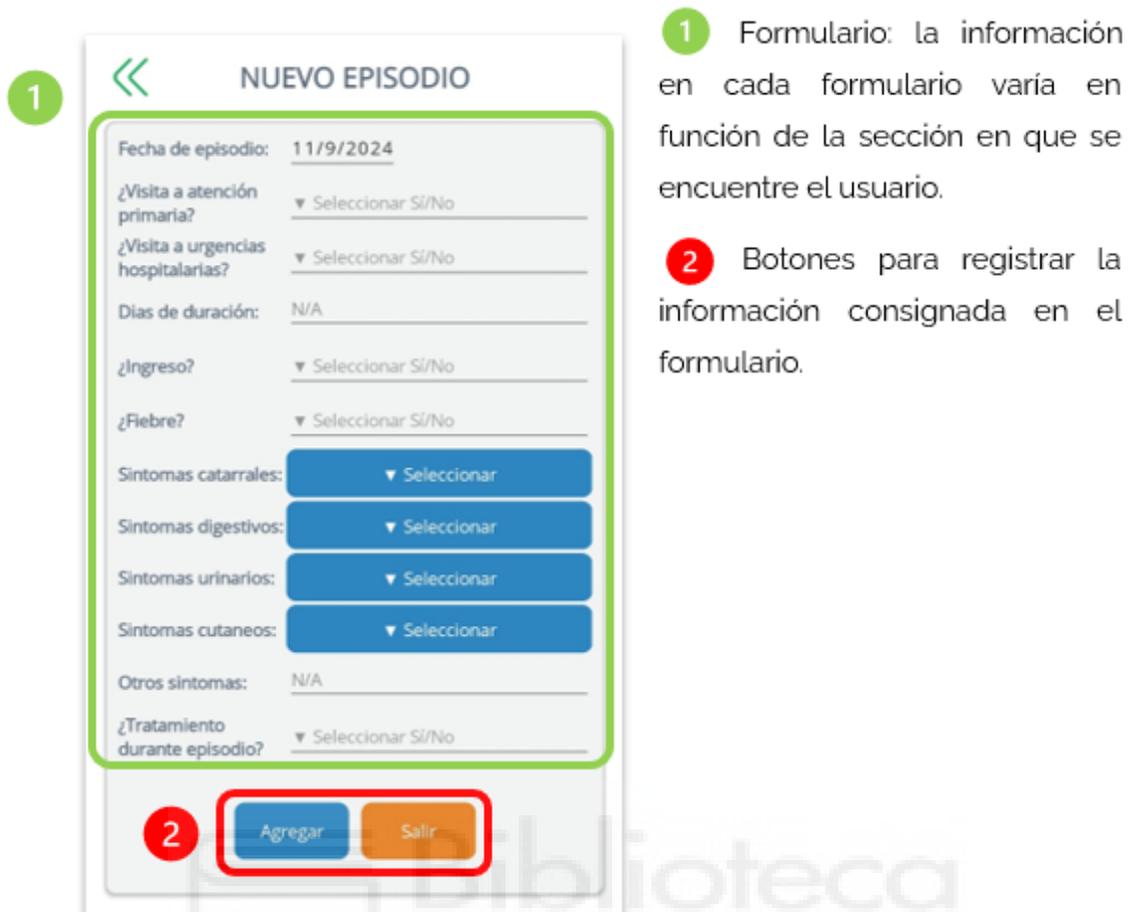


Figura 17. Controles y aspecto en ventanas de tipo formulario.

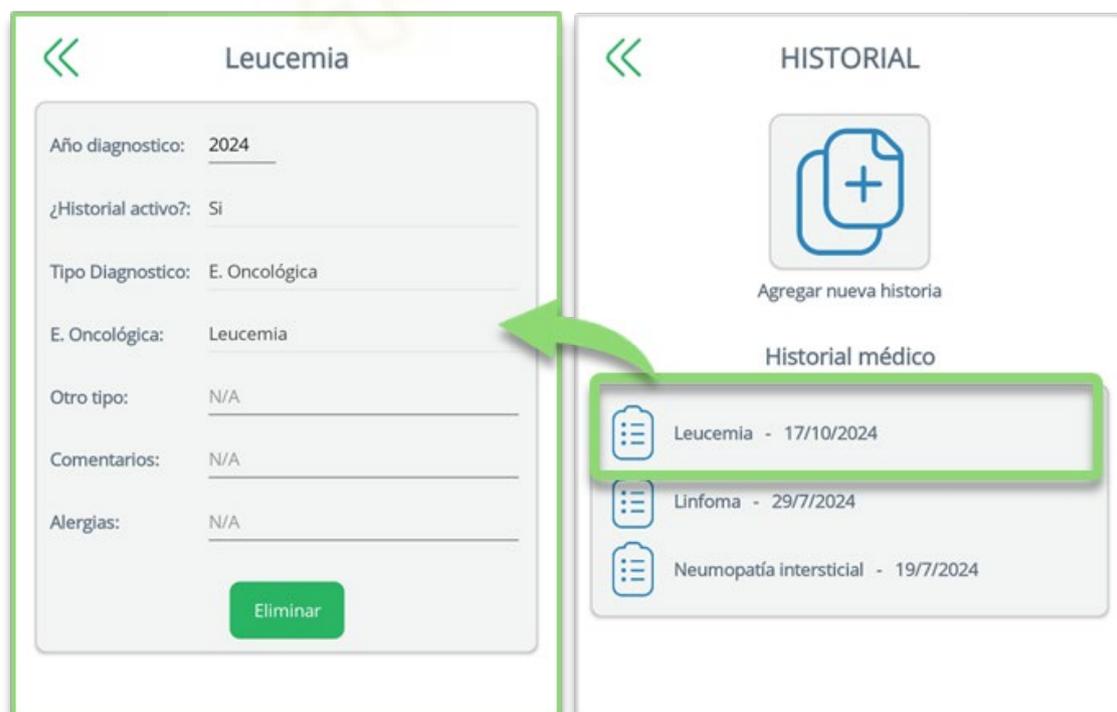


Figura 18. Ventada de detalles del documento o formulario.

Por otro lado, en los apartados de *Cuestionarios y Pruebas de laboratorio*, la aplicación permite generar gráficas a partir de los datos registrados en cada sección. Las gráficas buscan cumplir las siguientes necesidades:

- Ayudar a visualizar la evolución de parámetros clave.
- Facilitar la detección de tendencias o cambios relevantes.
- Mejorar el seguimiento clínico y fomentan la autogestión del paciente.

La Figura 19 muestra un ejemplo de las gráficas generadas para la sección de *Pruebas de laboratorio*, las cuales se construyen a partir de analíticas registradas por el paciente. En este caso se relacionan parámetros como Gammaglobulinas y Perfil metabólico.

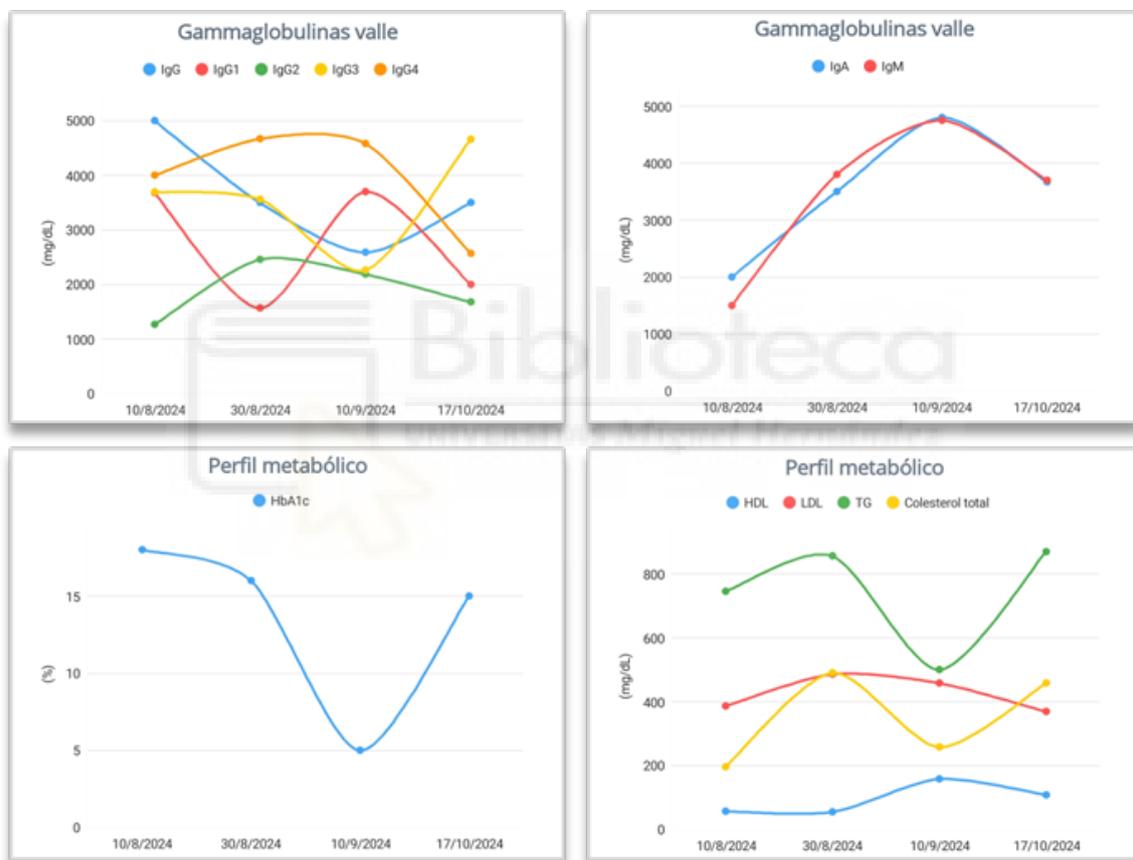


Figura 19. Ejemplo de gráficas evolutivas para pruebas de laboratorio.

### 3.4. Implementación en .NET MAUI

En esta sección se detallan los aspectos fundamentales que componen la estructura principal de la aplicación: la configuración y uso de la base de datos, la incorporación de herramientas para la gestión de datos “*On the edge*”, la integración de librerías externas para ampliar las capacidades de la app, y la

implementación del sistema de autenticación de usuarios mediante Firebase Authentication.

### 3.4.1. Arquitectura MVVM

Antes de continuar con la descripción de estas funcionalidades, es importante mencionar que la aplicación se desarrolló bajo una arquitectura MVVM. La arquitectura *MVVM (Model-View-ViewModel)* es uno de los patrones de diseño más utilizado en aplicaciones de .NET MAUI, ya que promueve una separación clara de responsabilidades dentro del proyecto. Este patrón divide la aplicación en tres componentes principales: el Modelo (*Model*), que representa la lógica de negocio o clases de datos; la Vista (*View*), que contiene la interfaz de usuario; y el *ViewModel*, que actúa como intermediario entre ambos, gestionando la lógica de presentación y sirviendo como fuente de datos para la vista mediante mecanismos de enlace (*binding*). Esta organización no solo facilita el mantenimiento del código y la reutilización de objetos o clases, sino que también permite una implementación más eficaz de la *inyección de dependencias*.

La dependencia es básicamente cuando una clase depende de otra, es decir, consume los métodos y propiedades definidas dentro de otras clases. En .NET MAUI, la inyección de dependencias se integra de forma nativa en la arquitectura MVVM, lo que permite registrar servicios, modelos de datos o APIs externas en el contenedor de servicios (*Dependency Injection Container*) y luego consumirlos fácilmente desde cualquier *ViewModel*.

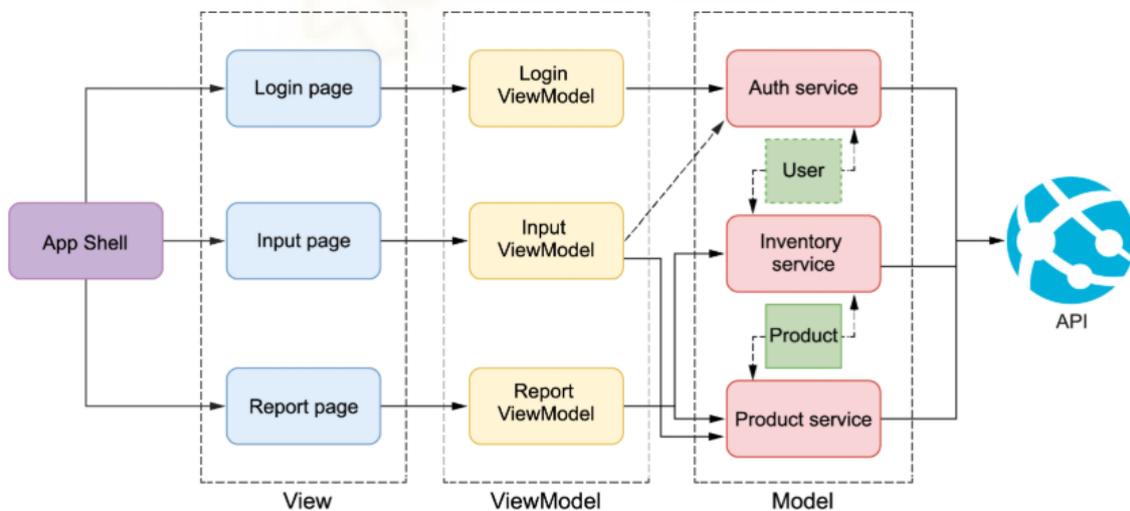


Figura 20. Esquema del patrón de diseño MVVM.

En la Figura 20 se puede observar las dependencias de los distintos objetos del proyecto. Para empezar, *App Shell*, es el componente principal de la aplicación, contiene todas las páginas definidas y gestiona la navegación entre las mismas, no hace falta implementar instrucciones de direccionamiento porque *Shell* lo realiza automáticamente. Ahora bien, cada página depende de un *ViewModel*, que, a su

vez, dependen de varios servicios o clases de datos. Por ejemplo, la página *LoginPage* depende de *LoginViewModel* y esta consume los métodos de autenticación declarados en *AuthService*, que es una herramienta “*On the edge*” de la API para gestión de usuarios online. *AuthService* a su vez, depende del modelo de datos *User* para gestionar las credenciales del usuario y así utilizarlas dentro de la aplicación.

La configuración de estas dependencias es bastante simple. Todo inicia declarando los servicios en el archivo *MauiProgram.cs*, este es el archivo principal de configuración y construcción de la aplicación. En éste se define cómo se inicializa la app, qué servicios estarán disponibles mediante inyección de dependencias, y cómo se configura el entorno antes de lanzar la interfaz gráfica. Una vez registrados, estos estarán disponibles en el contenedor de servicios (*Dependency Injection Container*), y podrán declararse en el constructor de cualquier *ViewModel* o *View* que lo necesite.

```
1 namespace IDPBookApp;
2 public static class MauiProgram
3 {
4     public static MauiApp CreateMauiApp()
5     {
6         var builder = MauiApp.CreateBuilder();
7         builder
8             .UseMauiApp<App>()
9             .ConfigureFonts(fonts =>
10             {
11                 fonts.AddFont(...);
12             });
13
14         builder.Services.AddSingleton<FirebaseConnecty>();
15         builder.Services.AddTransient<MainViewModel>();
16
17 -----
16 using IDPBookApp.DataBase;
17 namespace IDPBookApp.ViewModel;
18 public partial class MainViewModel : BaseViewModel
19 {
20     FirebaseConnecty firebaseConnecty;
21     public MainViewModel(FirebaseConnecty firebaseConnecty)
22     {
23         this.firebaseConnecty = firebaseConnecty;
24     }
25 }
```

En la primera sección (1-15) del código anterior, se muestra la declaración del servicio *FirebaseConnecty* dentro de la clase *MauiProgram.cs*. Los objetos/clases se pueden registrar como *Singleton* (una vez solicitado se mantiene en memoria) o *Transient* (cada vez que es solicitado, se crea una nueva instancia del objeto/clase). En la segunda sección (16-24) del código se indica la inyección de dependencias a través del constructor de la clase *MainViewModel*. Así, cada vez que la aplicación solicite la página asociada a *MainPage*, tendrá disponibles los servicios de Firebase

implementados en *FirestoreConnectivity*. La misma estructura se utiliza para la declaración de dependencias entre objetos *View* (interfaz gráfica) y *ViewModel*.

### 3.4.2. Integración de librerías en .NET MAUI

La instalación de librerías, API's, SDK, entre otros, se puede realizar por consola, bien sea la del ordenador o la herramienta nativa de Visual Studio, o también, a través del administrador de paquetes NuGet de Visual Studio. La ventaja de utilizar el administrador de paquetes NuGet, es su interfaz gráfica. La Figura 21 muestra dicha interfaz, en esta se puede observar los distintos apartados disponibles para la búsqueda e instalación de librerías, así como la información detallada del paquete *FirestoreDatabase.net*.

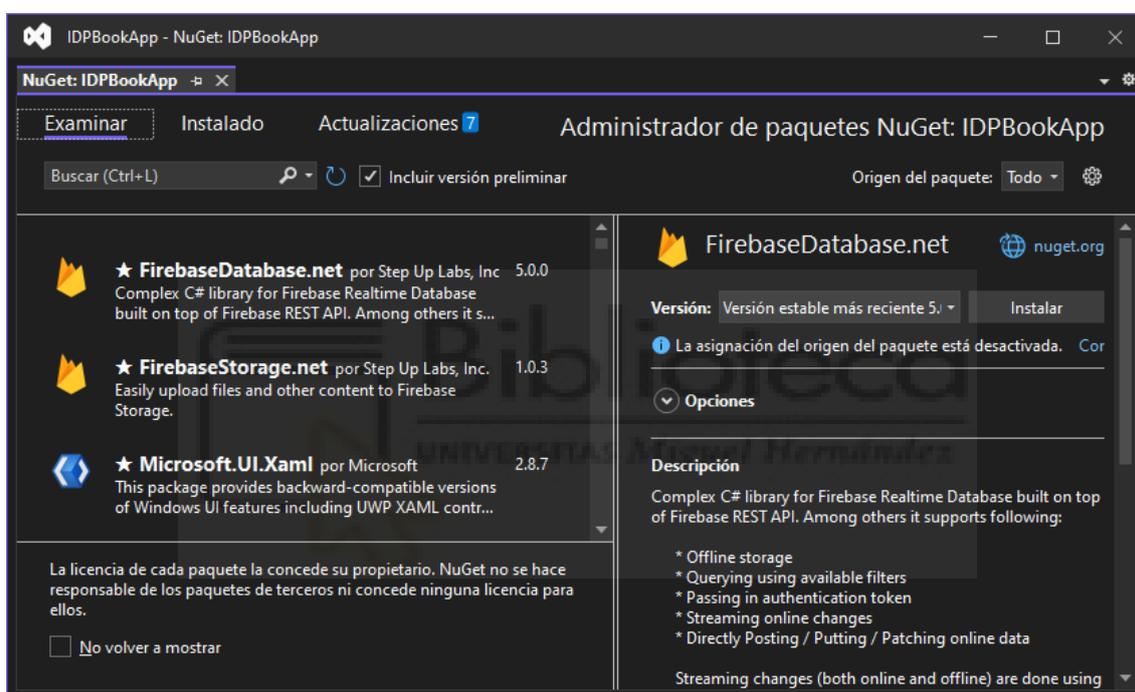


Figura 21. Administrador de paquetes NuGet de Visual Studio.

A diferencia del método en consola, aquí es posible encontrar mucha más información de los paquetes como versiones disponibles, notificaciones sobre actualizaciones, documentación de configuración, valoraciones del proyecto, y mucho más. Una de las ventajas más significativas frente al método tradicional es que cuenta con un buscador integrado que resulta muy práctico pues en algunas ocasiones hay paquetes que deseamos utilizar y por algún motivo no son compatibles con el proyecto y gracias a esta herramienta, el proceso de encontrar una alternativa viable y actualizada es mucho más eficiente.

La configuración de los paquetes es automática, durante la instalación simplemente se solicitan algunos permisos de uso, sin embargo, algunas librerías necesitan algún *set-up* adicional. Por ejemplo, retomando el tema de la arquitectura *MVVM*, para su correcta implementación, es necesario invocar el

paquete de NuGet en el archivo de configuración principal *MauiProgram*. El siguiente código muestra la invocación mencionada.

```
1 using CommunityToolkit.Maui;
2 namespace IDPBookApp;
3 public static class MauiProgram
4 {
5     public static MauiApp CreateMauiApp()
6     {
7         var builder = MauiApp.CreateBuilder();
8         builder
9             .UseMauiApp<App>()
10            .UseMauiCommunityToolkit();
11     }
12 }
```

Para el uso de la librería, funciona como en la mayoría de los lenguajes de programación. Se declara en la cabecera de la clase, en este caso con C# se utiliza la palabra *using*, tal como se indica en la línea 1 del código anterior.

La mayor parte de librerías utilizadas para el desarrollo de la aplicación son nativas de Microsoft/Visual Studio, sin embargo, para implementar funciones especiales es necesario el uso de librerías de terceros. En este caso se han utilizado principalmente tres.

1. **Google.Cloud.Firestore:** Librería de desarrollo de Google para acceder a las API de Firestore.
2. **FirebaseAuthentication.net:** Librería no oficial de Firebase, enfocada especialmente al desarrollo en .NET.
3. **DevExpress.Maui.Charts:** Kit de desarrollo de gráficos 2D para .NET MAUI.

En los siguientes apartados se describirá el uso y configuración de cada paquete.

### 3.4.3. Configuración de la base de datos de Firestore

El primer paso consiste en crear un nuevo proyecto desde la plataforma web de Firebase. Después, se debe añadir una base de datos. Firebase ofrece dos opciones: Realtime Database (tipo SQL) y Firestore (tipo NoSQL). Para esta aplicación se eligió Firestore, ya que su modelo flexible se adapta mejor a la estructura de datos definida y ofrece ventajas que ya fueron expuestas en secciones anteriores.

### Registro de aplicaciones

En la configuración del proyecto de Firebase, se deben registrar las aplicaciones que tendrán acceso a la base de datos, en este caso, al tratarse de una app multiplataforma, se registraron dos apps, una para Android y otra de Apple. El

proceso de configuración del SDK es el mismo para cualquier sistema operativo, simplemente se relaciona el ID del proyecto, que en este caso es “com.mru.idpbookapp”, luego, Firebase genera los archivos de configuración para cada plataforma los cuales contienen las credenciales de acceso a la base de datos. Estos archivos de extensión **.json**, se deben agregar al directorio de cada plataforma en el explorador de soluciones del proyecto, tal como se muestra en la captura de la Figura 22.

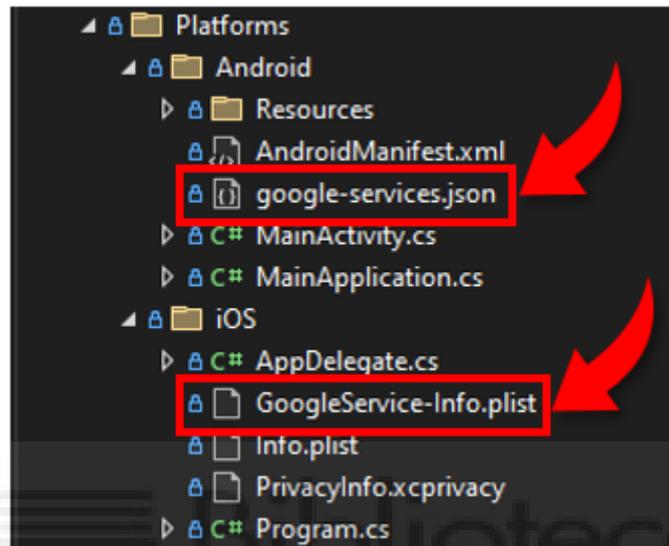


Figura 22. Configuración de SDK para Android y Apple en Visual Studio.

También es necesario generar una clave para el uso de una cuenta de servicio de Firebase, este proceso también se realiza desde la configuración del proyecto, dirigiéndose a *Cuentas de servicio*, luego, en generar nueva clave privada. El archivo generado se debe agregar al directorio raíz del proyecto, sin embargo, para mantener un orden de los recursos utilizados, se agregó al fichero *Resources*, en la carpeta *Raw*. La Figura 23 muestra la configuración descrita.

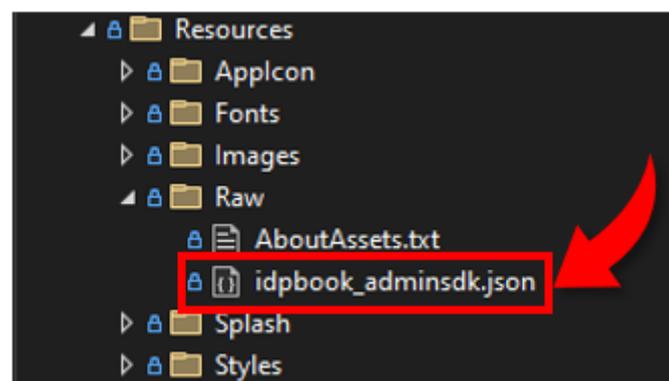


Figura 23. Configuración del SDK de Firebase Admin.

Una vez realizada la configuración anterior, en el archivo *FirebaseConnecty.cs* se debe inicializar la base de datos, para ello se utiliza la siguiente configuración.

```
1 private FirestoreDb db;
2 public async Task SetupFirestore()
3 {
4     if (db == null)
5     {
6         var st = await FileSystem.OpenApp("idpbook_adminsdk.json");
7         var reader = new StreamReader(st);
8         var contents = reader.ReadToEnd();
9         db = new FirestoreDbBuilder
10        {
11            ProjectId = "idpbook-lafe-umh",
12            JsonCredentials = contents
13        }.Build();
14    }
15 }
```

El código anterior, permite el acceso a la base de datos utilizando las credenciales de la configuración SDK extraídas desde el archivo *idpbook\_adminsdk.json*. Este método *SetupFirestore*, se invoca antes de arrancar la aplicación, específicamente desde el *ViewModel* de la página *LoginPage*, puesto que es la página por defecto que solicita la aplicación cuando se inicializa.

### Implementación de consultas

Para la gestión de la base de datos desde la aplicación se diseñaron funciones asíncronas con el objetivo de garantizar una comunicación eficiente y evitar errores con Firestore, permitiendo al usuario interactuar con la aplicación sin interrupciones mientras se realizan operaciones de lectura, escritura o actualización en tiempo real. Cada función utiliza distintos argumentos para acceder a diferentes rutas en la base de datos, sin embargo, la estructura es igual para todas, la función se encapsula dentro de una instrucción *try/catch* para gestionar cualquier error que pueda presentarse durante la operación con Firestore. Esto garantiza que la aplicación nunca se cierre inesperadamente.

Como un ejemplo de implementación de funciones con *C#* en Visual Studio se plantea el siguiente código.

```
1 // Declaración
2 public async Task<List<T>> GetModelList<T>(string uid, string tipo)
3 {
4     try
5     {
6         var snapshot = await db.Collection("IDPbookDB/"+uid+"/"+tipo).GetSnapshotAsync();
7         var docs = new List<T>();
8         foreach (var doc in snapshot.Documents)
9         {
10            docs.Add(doc.ConvertTo<T>());
11        }
12        return docs;
13    }
```

```

14 catch (Exception ex)
15 {
16     Debug.WriteLine(ex);
17     await Shell.Current.DisplayAlert("GetModelList", $"Error: {ex.Message}", "Ok");
18     return null;
19 }
20 }
-----
21 // Invocación
22 public ObservableCollection<Model> Lista { get; set; } = [];
23 async void GetFunc()
24 {
25     var list = await GetModelList<Model>(pacInfo.Uid, "tipo");
26     if (list != null && list.Count > 0)
27     {
28         Lista.Clear();
29         foreach (var doc in list)
30         {
31             Lista.Add(doc);
32         }
33     }
34 }

```

Estas líneas corresponden a la definición y uso del método *GetModelList<T>*. Esta función se utiliza para recuperar todos los documentos de la ruta "IDPbookDB/uid/tipo" y convertir cada documento en un objeto del tipo <T>, devolviendo una lista con todos ellos. Se puede observar el control de excepciones con las instrucciones *try/catch*, que en dado caso la operación no se pueda realizar, se mostrará una alerta al usuario con la descripción del error (*ex.Message*).

Las funciones se invocan desde cualquier *ViewModel* que tenga *dependencia* del servicio *FirestoreConnecty.cs*. En el código anterior (*//Invocación*), muestra un ejemplo del uso de la función *GetModelList<T>*.

En la Tabla 1 se consignan todas las funciones implementadas para la gestión de la base de datos desde la aplicación IDPBook.

Nombre	Descripción	Argumentos	Devolución
<i>GetModelList&lt;T&gt;</i>	Recupera los documentos dentro de una colección correspondiente a la ruta que se construye con los argumentos de la función.	UID paciente ( <i>uid</i> ) Colección ( <i>tipo</i> )	Lista de objetos correspondiente a cada documento recuperado.
<i>GetAdmins</i>	Recupera todas las administraciones registradas para un tratamiento con inmunoglobulinas, el tratamiento puede ser el actual o antiguo.	UID paciente ( <i>uid</i> ) Colección ( <i>tipo</i> ) Tratamiento ( <i>trat</i> )	Lista de administraciones del tratamiento referenciado.

<i>GetPacientesModel</i>	Recupera los pacientes registrados en la aplicación.	Ninguno	Lista con los pacientes registrados en la base de datos.
<i>GetPaciente</i>	Obtiene los datos del paciente que ha iniciado sesión.	UID paciente ( <i>uid</i> )	Objeto de la clase <i>Paciente</i> .
<i>SaveData&lt;T&gt;</i>	Permite guardar documentos de cualquier clase de variables.	UID paciente ( <i>uid</i> ) Colección ( <i>tipo</i> ) Id documento ( <i>id</i> ) Objeto ( <i>T modelo</i> )	Ninguno.
<i>SavePac&lt;T&gt;</i>	Registra pacientes en la base de datos.	UID paciente ( <i>uid</i> ) Objeto ( <i>Paciente</i> )	Ninguno.
<i>ElimDocs</i>	Permite eliminar documentos de cualquier colección de la base de datos.	UID paciente ( <i>uid</i> ) Colección ( <i>tipo</i> ) Id documento ( <i>id</i> )	Ninguno.
<i>EliminarTrat</i>	Elimina tratamientos con sus respectivas administraciones.	UID paciente ( <i>uid</i> ) Colección ( <i>tipo</i> ) Tratamiento ( <i>trat</i> )	Ninguno.
<i>GetMedUsers</i>	Comprueba la existencia de usuarios de tipo personal médico en la base de datos.	Correo ( <i>email</i> )	<i>True</i> , si existe. <i>False</i> , si no existe.

Tabla 1. Funciones para la gestión de datos en Firestore

#### 3.4.4. Configuración de la gestión de usuarios

Al igual que la base de datos, el servicio de Firebase Authentication utiliza las credenciales de la configuración SDK extraídas desde el archivo *idpbook\_adminsdk.json* para vincular la cuenta de servicio con el proyecto de .NET MAUI. Adicionalmente, es necesario declarar explícitamente la *APIKey* y el dominio autorizado por Firebase para completar la configuración del servicio de autenticación. Esta configuración corresponde con el siguiente código.

```

1 public static FirebaseAuthConfig config = new()
2 {
3     ApiKey = "<API key>",
4     AuthDomain = "<Domain>.firebaseapp.com",
5     Providers =
6     [
7         new EmailProvider()
8     ],
9 };

```

El *APIKey* y el dominio se obtienen desde la configuración del proyecto de Firebase, accediendo al módulo *Authentication*, luego en Configuración, aparecerán todos los dominios que Firebase ha dispuesto para el proyecto.

También se deben configurar los proveedores de autenticación que se desean utilizar en la aplicación, en este proyecto únicamente se ha dispuesto del acceso por correo y contraseña. Existen otros métodos nativos como el acceso con número telefónico o el acceso como invitado, además de proveedores de identidad federadas como Google, Apple y GitHub. Sin embargo, de acuerdo con los intereses del proyecto, el acceso con correo y contraseña resulta lo más práctico y sencillo para la gestión de usuarios en la aplicación.

### Funciones implementadas para la gestión de usuarios

Las funciones de autenticación también se declaran en el archivo de servicios *FirebaseConnecty.cs*, gracias a la interoperabilidad de las librerías del ecosistema de Google, es posible unificar la gestión de la base de datos y la gestión de usuarios en la misma clase.

Por otro lado, dentro de esta implementación, también se utilizan las propiedades “*On the edge*” del SDK de Firebase, como por ejemplo los repositorios locales, que permiten gestionar el estado de sesión del usuario. Esto evita que el usuario tenga que identificarse siempre que se abra la aplicación, una vez el usuario inicia sesión, los repositorios guardan sus credenciales y acceden automáticamente.

En el siguiente código se muestra un ejemplo del uso de los repositorios implementado en la función *Login*, diseñada para el acceso a la aplicación.

```
1 // Declaración
2 private readonly FirebaseAuthClient client = new(config);
3 public readonly FileUserRepository PacRepo = new("ListPac");
4 public UserInfo pacInfo;
5 private FirebaseCredential fCrndl;
6 public UserCredential UserCrndl;
7
8 public async Task Login(string email, string password)
9 {
10     fCrndl = await client.SignInWithEmailAndPasswordAsync(email,password);
11     PacRepo.SaveUser(UserCrndl.User);
12     (pacInfo, fCrndl) = PacRepo.ReadUser();
13 }
-----
14 // Invocación
15 await firebaseConnecty.Login(Email,Password);
```

El método tiene como parámetros de entrada el correo y contraseña que se toman desde los campos de texto que el usuario relleno, con esta información se invoca la operación *SignInWithEmailAndPasswordAsync*, que asigna las credenciales del ingreso del usuario a la variable *fCrndl*, para posteriormente guardarlas en el

repositorio *PacRepo*. De esta manera, la sesión del usuario quedará registrada en la aplicación y no será necesario hacerlo nuevamente hasta que la sesión se cierre explícitamente.

A continuación, en la Tabla 2 se consignan todas las funciones implementadas para la gestión de usuarios con la aplicación IDPBook.

Nombre	Descripción	Argumentos	Devolución
<i>Login</i>	Inicia sesión en la aplicación.	Correo ( <i>email</i> ) Contraseña ( <i>pass</i> )	Ninguno.
<i>CheckUser</i>	Comprueba el estado de la sesión en los repositorios.	Ninguno.	<i>True</i> , si la sesión sigue activa, <i>False</i> , si el usuario ha cerrado sesión.
<i>RegistMed</i>	Registra usuarios de tipo personal médico.	Correo ( <i>email</i> ) Contraseña ( <i>pass</i> ) Nombre ( <i>name</i> )	Ninguno.
<i>RegistPac</i>	Registra usuarios de tipo paciente.	Correo ( <i>email</i> ) Contraseña ( <i>pass</i> ) Nombre ( <i>name</i> )	Ninguno.
<i>LogOut</i>	Finaliza sesión del usuario en la aplicación.	Ninguno.	Ninguno.
<i>ChangePassword</i>	Cambia la contraseña de usuarios de tipo paciente.	Correo ( <i>email</i> ) Contraseña ( <i>pass</i> )	Ninguno.
<i>SendEmail</i>	Envía un enlace a los usuarios de tipo personal médico para cambio de contraseña.	Correo ( <i>email</i> )	Ninguno.

*Tabla 2. Funciones para la gestión de usuarios con Firebase Authentication.*

## Autenticación con *Firestore*

El proceso de autenticación es muy simple y solo es tarea del personal médico comprenderlo. En la aplicación se diseñó un apartado para el registro de los usuarios o pacientes, llamado “Registro usuarios” al cual solo tiene acceso el personal médico o administrador. Se accede desde la ventana principal, presionando el icono verde de la esquina superior izquierda, el cual solo está disponible para usuarios de tipo personal médico. Tal como se muestra en la Figura 24.



Figura 24. Apartado de registro de usuarios.

En esta sección se registra el nuevo paciente con el correo correspondiente, la aplicación asignará la contraseña predeterminada para un usuario de tipo paciente y este se registra en la base de datos del servicio de autenticación de *Firestore*. En la Figura 25, se muestra una captura de algunos usuarios de prueba registrados en la aplicación durante el desarrollo del trabajo. El parámetro llamado “UID de usuario” es el identificador que se vincula con la base de datos de *Firestore*, de esta manera, cuando un usuario inicia sesión, la aplicación identifica ese UID y sincroniza los datos correspondientes al identificador del paciente.

Identificador	Proveedores	Fecha de creación ↓	Fecha de acceso	UID de usuario
al@gmail.com	✉	8 nov 2024	23 nov 2024	HMkZS31q1zXwp0rPqVJLREz...
medicalroboticsumh@g...	✉	6 nov 2024	6 nov 2024	UbOVprWUfFSP4sKg6gRLyCR...
jromero@gmail.com	✉	17 jul 2024	5 dic 2024	DFnPHgaJkhVYtfYcfn1NleXN...
adrodriguez@unicauca...	✉	9 jul 2024	9 jul 2024	UzXZwUwjcPdm83KewcL4Wy...
idpbook1@gmail.com	✉	2 may 2024	5 dic 2024	40KG0u4RZ6aqTLy6bZrTb6...

Figura 25. Lista de usuarios de prueba registrados en Firebase.

En la Figura 26, se puede observar un ejemplo de la identificación de usuarios entre el servicio de autenticación y la base de datos, en este caso, el UID del usuario de prueba “al@gmail.com”, se asigna al documento que contiene toda la información registrada para dicho usuario.

```

{
  "Apellido": "López",
  "Correo": "al@gmail.com",
  "Diagnsc": 2,
  "FechDiag": "9/10/2018",
  "FechDiag1": null,
  "FechDiag2": null,
  "FechNac": "8/11/2003",
  "Gener": "femenino",
  "IdMed": "40KG0u4RZ6aqTLy6bZrTb6sssD2",
  "Nombre": "Alejandro",
  "OtroDiag1": null,
  "OtroDiag2": null
}

```

Figura 26. Estructura de datos en Firestore.

Otra funcionalidad implementada en la aplicación es la capacidad de verificar los usuarios para otorgar permisos de uso. En este caso, únicamente los usuarios registrados como personal médico, reciben un correo para validar el perfil creado en el servicio de autenticación, de esta manera cuando ingresan a la aplicación puede acceder a las secciones que los pacientes no tienen autorización. Uno de esos apartados es el de registro de usuarios mencionado anteriormente.

### 3.4.5. Configuración de gráficos con DevExpressCharts

*DevExpress* es una biblioteca de desarrollo software que proporciona soluciones adaptadas a varias plataformas como .NET, Blazor, WinForms, WPF, entre otras. Ofrece una solución completa para interfaz de usuario, desde herramientas para visualización de datos, generador de reportes y analíticas, hasta integración de bases de datos. Muchas herramientas son de pago, sin embargo, los controles de interfaz de usuario para .NET MAUI como el paquete *Charts*, es gratuito, aunque requiere de un registro en su página web para obtener el enlace de descarga en *NuGet*.

La librería soporta varios tipos de gráficos, desde diagramas circulares, de barras, áreas, líneas, hasta gráficos financieros. Para este proyecto se buscaba presentar gráficos que describieran la evolución de las analíticas del paciente, así como del estado de salud derivado de los cuestionarios de calidad de vida que el paciente registra en la aplicación. En este caso se ha utilizado la herramienta *SplineSeries*, esta clase de gráfico despliega un plano cartesiano donde se ubican los puntos correspondientes a cada dato suministrado y son conectados por líneas curvas suavemente en cada punto. En la Figura 27 se muestra un ejemplo de este tipo de gráficas, representa la evolución de un tipo de analítica llamado “Gammaglobulinas valle”.

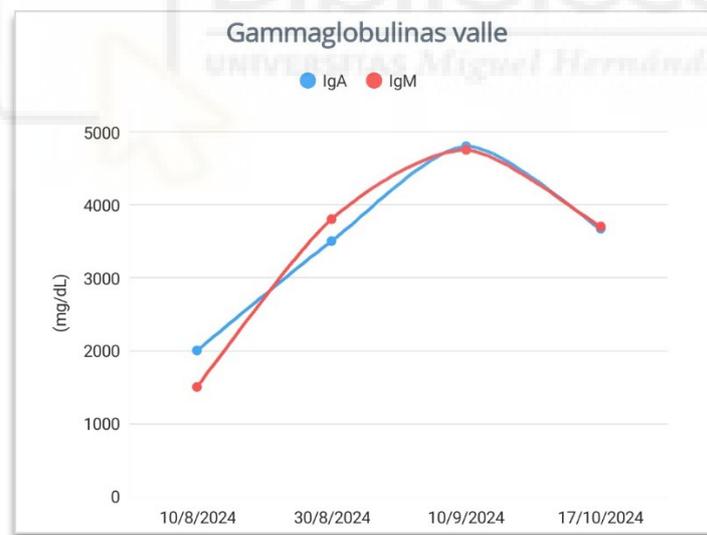


Figura 27. Gráfico diseñado con la librería *DevExpress Charts*.

Cuando el usuario interactúa con los puntos de la gráfica, se despliega un cuadro informativo llamado “*Tooltip*”, que muestra detalles relevantes como valores exactos, etiquetas, categorías o cualquier información que se haya configurado. Esta y muchas más opciones de personalización permiten presentar datos de manera muy estética y eficiente, que el usuario puede comprender fácilmente, ofreciendo una experiencia de usuario agradable e intuitiva.

Al igual que el paquete de arquitectura MVVM, es necesario invocarlo la librería en el archivo de configuración principal *MauiProgram*. Se deben registrar dos ensambladores *.UseDevExpress* y *UseDevExpressCharts*. El siguiente código muestra la invocación mencionada.

```
1 using CommunityToolkit.Maui;
2 namespace IDPBookApp;
3 public static class MauiProgram
4 {
5     public static MauiApp CreateMauiApp()
6     {
7         var builder = MauiApp.CreateBuilder();
8         builder
9             .UseMauiApp<App>()
10            .UseMauiCommunityToolkit()
11            .UseDevExpress()
12            .UseDevExpressCharts()
```

A diferencia de las librerías descritas anteriormente, donde su uso está destinado a la lógica de funcionamiento, esta librería gráfica se utiliza en los componentes dedicados a la interfaz, es decir en los archivos XAML del proyecto. Por ejemplo, el código correspondiente a la gráfica de la Figura 27, se implementó en el archivo *AnalyticsPage.xaml* y corresponde a las siguientes líneas de código.

```
1 <dx:SplineSeries MarkersVisible="True" DisplayName="IgA">
2     <dx:SplineSeries.Data>
3         <dx:SeriesDataAdapter Data="{Binding Analiticas}" Argument="AFecha">
4             <dx:ValueDataMember Type="Value" Member="AIgA"/>
5         </dx:SeriesDataAdapter>
6     </dx:SplineSeries.Data>
7 </dx:SplineSeries>
8 <dx:SplineSeries MarkersVisible="True" DisplayName="IgM">
9     <dx:SplineSeries.Data>
10        <dx:SeriesDataAdapter Data="{Binding Analiticas}" Argument="AFecha">
11            <dx:ValueDataMember Type="Value" Member="AIgM"/>
12        </dx:SeriesDataAdapter>
13    </dx:SplineSeries.Data>
14 </dx:SplineSeries>
```

Como se mencionaba al inicio de este apartado, se utiliza la herramienta *SplineSeries* para generar un gráfico con líneas curvadas. En las líneas 1 y 8 se define el nombre que se quiere mostrar para cada conjunto de datos (*DisplayName*). Luego en las líneas 3 y 10, se realiza la asignación de los datos con la propiedad *Data*, en este caso, corresponde a una colección enlazada desde el *ViewModel* de la página, llamada *Analíticas*. Esta colección contiene todas las analíticas que se han obtenido de la base de datos del paciente. El parámetro *Argument* hace referencia a los datos que la gráfica utiliza como eje de abscisas, en este caso la fecha de cada analítica. Mientras que en la línea 4 y 11, se define el valor del eje de ordenadas con la propiedad *Member*. Para este caso en particular, se vincula los datos de “AIgA” y “AIgM”, que son propiedades definidas para la

clase *Analítica*. De esta manera, se obtiene una gráfica con dos grupos de datos, como se muestra en la Figura 27.

### 3.4.6. Flujo de la aplicación

La secuencia de navegación entre las ventanas de la aplicación es fija, es decir, el flujo siempre es el mismo y el usuario no puede navegar aleatoriamente entre las páginas. El sistema de navegación indicado en el esquema de la Figura 28, describe el flujo entre los apartados de la aplicación, se puede observar que es una estructura jerárquica en la que no existen atajos entre los distintos elementos.

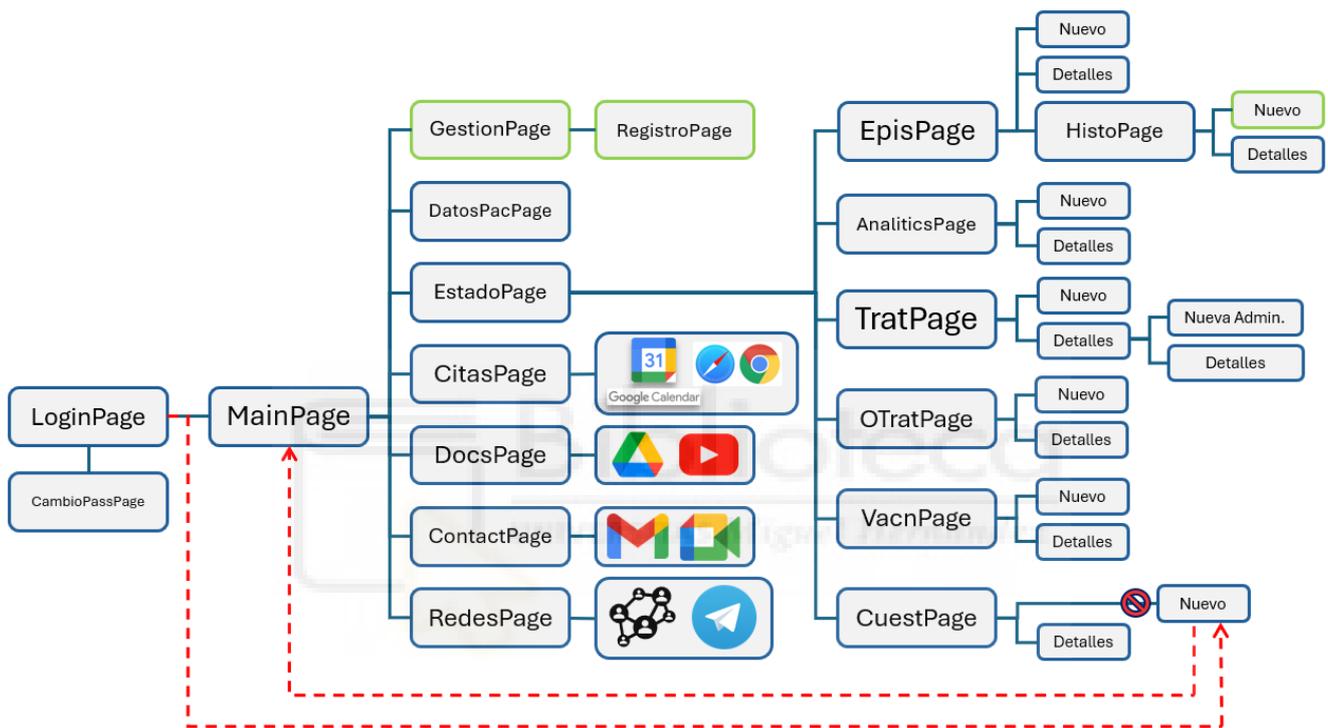


Figura 28. Flujo de navegación entre las páginas de la aplicación.

Existe una excepción a la navegación preestablecida cuando se accede por primera vez a la aplicación. Cuando un usuario inicia sesión por primera vez, es trasladado al apartado de Cuestionarios de calidad de vida para realizar un primer registro de manera obligatoria. Esto significa que el usuario no puede acceder al *MainPage* hasta que realice un primer cuestionario. La aplicación está diseñada para bloquear cualquier acción de navegación hacia atrás en el flujo de navegación y así garantizar que este registro se complete exitosamente. Como se puede observar en la Figura 28, este flujo está representado por la línea punteada de color rojo.

Por otro lado, las páginas representadas con color verde solo son accesibles por usuarios de tipo personal médico. Por lo tanto, los elementos gráficos de navegación hacia estas páginas solo estarán disponibles cuando el usuario que haya iniciado sesión sea de este tipo. Esta comprobación se hace gracias a las

credenciales de autenticación de Firebase, que permite verificar usuarios a través del envío de correos de verificación que únicamente reciben los usuarios registrados como personal médico.

La única página configurada como un servicio de tipo *Singleton* es *MainPage*, lo que implica que su instancia se mantiene activa en memoria durante todo el ciclo de vida de la aplicación. Por el contrario, el resto de las páginas han sido registradas como servicios de tipo *Transient*, lo cual hace que se genere una nueva instancia cada vez que se accede a ellas. Este enfoque permite optimizar la gestión de recursos del dispositivo, ya que dichas instancias son eliminadas automáticamente al abandonar la vista correspondiente, evitando así un uso innecesario de memoria y mejorando el rendimiento general de la aplicación.

### **3.5. Generación y publicación de la aplicación**

Durante la fase de desarrollo de la aplicación, se llevaron a cabo pruebas principalmente en modo *Debug*, una configuración que permite ejecutar la app de forma rápida tanto en emuladores como en dispositivos físicos. Este modo es ideal para identificar errores, realizar ajustes en tiempo real y evaluar el desempeño funcional del sistema. Una vez se alcanzó una versión estable, se procedió con pruebas en modo *Release*, un proceso fundamental para validar el rendimiento, la eficiencia y la estabilidad de la aplicación bajo condiciones más cercanas al entorno de producción. La compilación en modo *Release* constituye el primer paso para el proceso de distribución, ya que permite generar paquetes optimizados para ser publicados en las tiendas de aplicaciones. Dado que se trata de un desarrollo multiplataforma, los procesos de publicación varían según el sistema operativo destino. A continuación, se detallan los procedimientos específicos para la distribución en plataformas Android y iOS.

#### **3.5.1. Android**

La tienda de aplicaciones Android más utilizada es *Google Play* y para ser capaces de publicar con esta plataforma es necesario contar con una cuenta de desarrollador. El proceso inicia con la creación de un nuevo proyecto en *Google Play Console*, aquí solo es necesario ingresar el nombre, confirmar el tipo de aplicación (app o juego) y si es de pago o gratis, la configuración completa se describe en el siguiente paso.

##### **1. Configuración de aplicación en Google Play Console**

El proceso de configuración no es difícil, pero requiere de mucho tiempo. El paso a paso completo no es necesario describirlo para efectos de este documento, pero en pocas palabras, durante la configuración se deben ir completando tareas de muchas categorías y algunas son de obligatorio cumplimiento. Algunas categorías son Anuncios, Clasificación de Contenido, Privacidad y seguridad de datos, Público objetivo, entre otros. La interfaz cuenta con una barra de progreso que se actualiza

con cada tarea completada, además de una lista con las tareas completas, pendientes o que requieren de una configuración adicional. La mayor parte de este proceso termina con la solicitud del archivo AAB (*Android App Bundle*) de la aplicación. Este paso se describe con mayor detalle en el paso 3. Algo que si es importante mencionar es que la aplicación se ha configurado para estar solo disponible en España y su distribución es gratuita.

## 2. Compilación y distribución en Visual Studio

El primer paso es seleccionar el formato del paquete de la aplicación, en este caso para realizar la distribución con Google Play el formato correcto es *conjunto de productos* o en ingles *bundle*. Esto genera el archivo AAB solicitado en la configuración del proyecto en *Console*. Esta configuración se realiza desde las propiedades del proyecto en Visual Studio, tal como se muestra en la Figura 29.

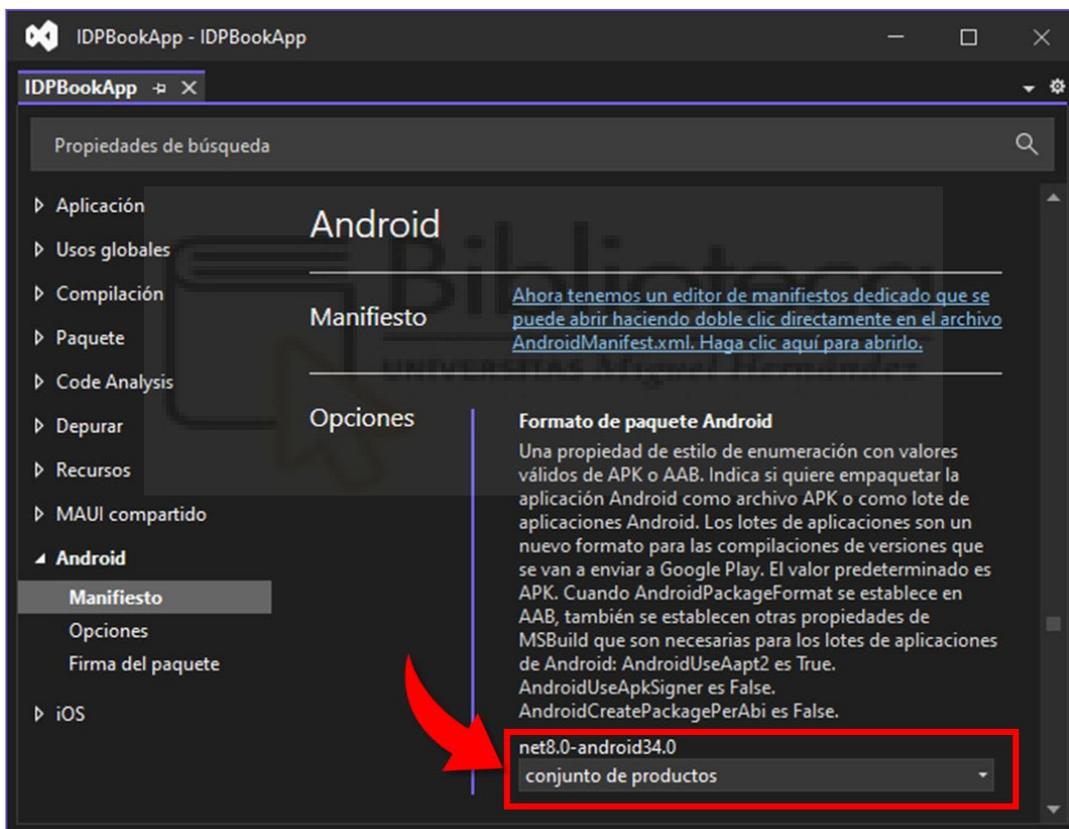


Figura 29. Configuración de paquete Android para distribución con Google Play.

Luego, asegurándose que el proyecto está en modo *Release*, desde el explorador de soluciones del proyecto, se procede con la publicación de la aplicación como se muestra en la Figura 30.

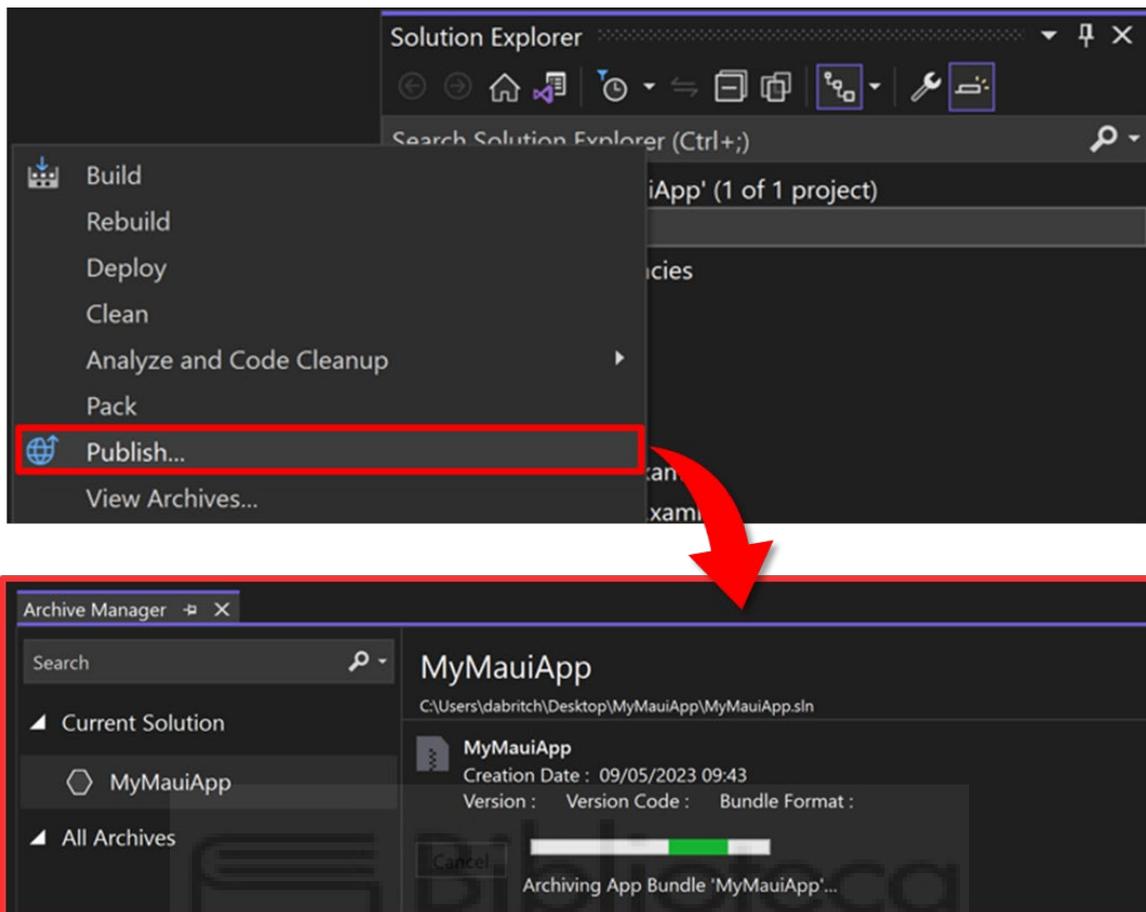


Figura 30. Publicación del proyecto en Visual Studio.

Una vez el proceso ha terminado y la aplicación se haya archivado correctamente, se procede con la firma del paquete. Desde el Administrador de archivos, se elige la opción Distribuir, que abre un cuadro de diálogo para continuar con el proceso de firma. En la Figura 31, se puede observar el proceso descrito.

La firma se realiza con una clave llamada *Android Keystore*. Se puede utilizar una clave que se haya creado anteriormente o crear una nueva. *Es muy importante guardar muy bien este archivo porque es indispensable para realizar futuras actualizaciones de la app en Google Play*. Si la identidad de la firma de futuras versiones de la app no coincide con la original, no se podrá realizar ningún cambio en la aplicación.

Continuando con el proceso, una vez se elige la identidad para firmar, se solicita la contraseña de la clave y se procede con la firma de la aplicación (Figura 32). Finalmente, desde el Administrador de archivos, se elige la opción Abrir distribución, que abrirá la carpeta que contiene el paquete AAB firmado y listo para subir en *Google Play Console*. En la Figura 33, se puede observar este procedimiento.

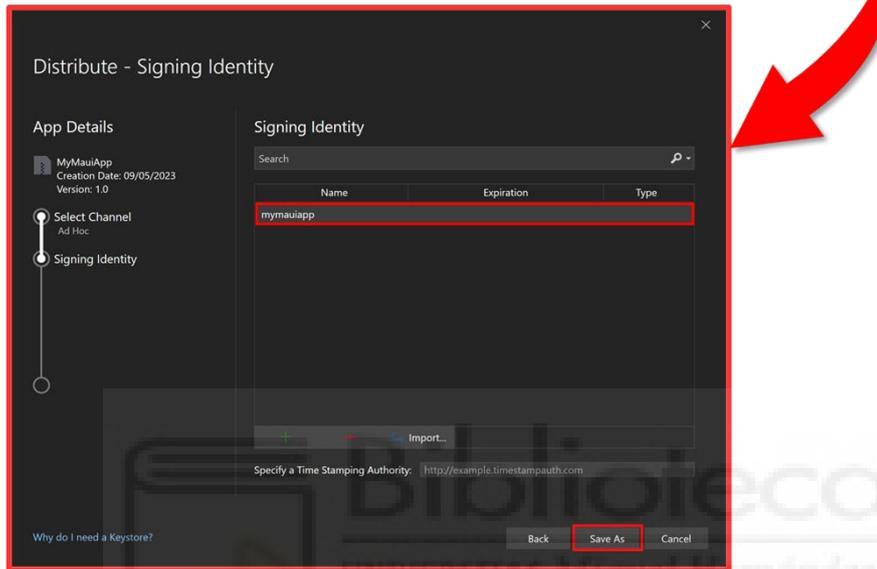
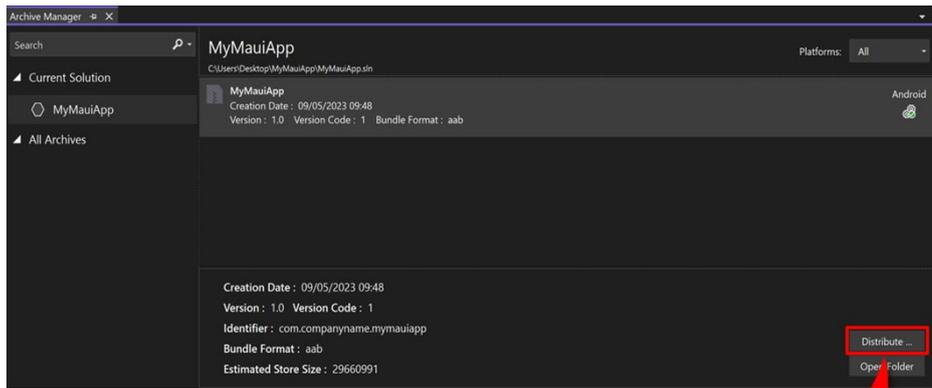


Figura 31. Distribución de la aplicación en Visual Studio.

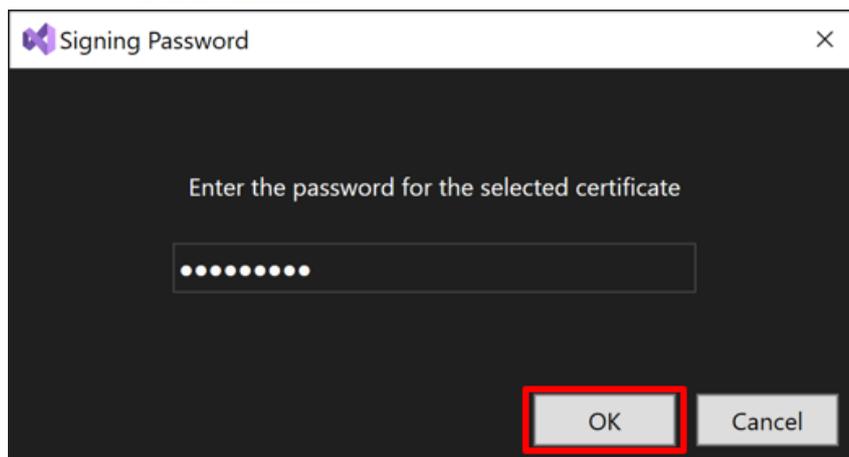


Figura 32. Firma del paquete de la aplicación.

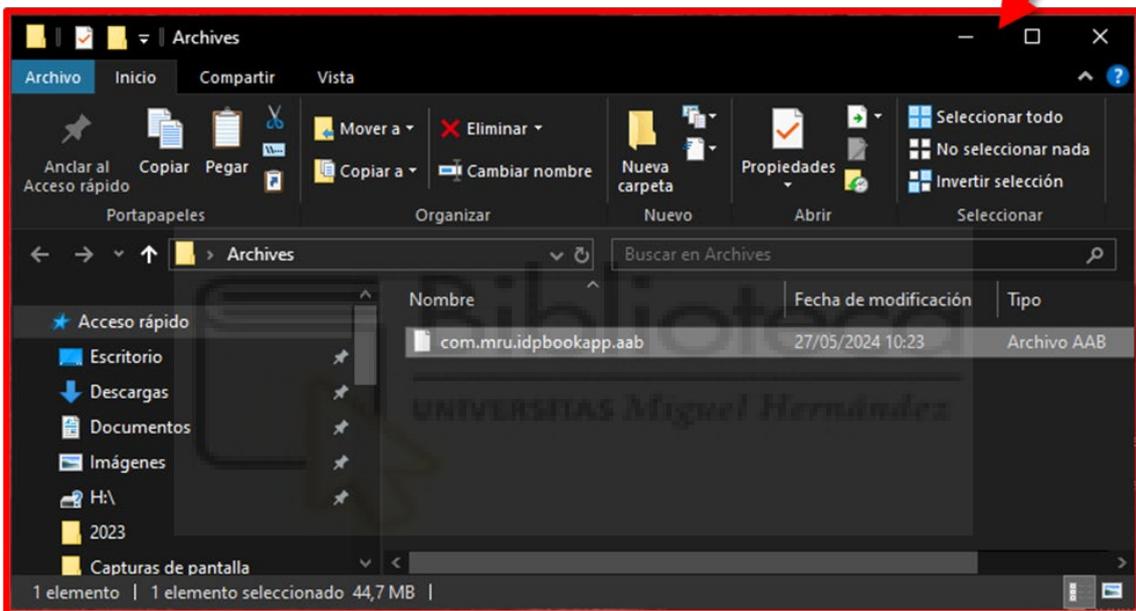
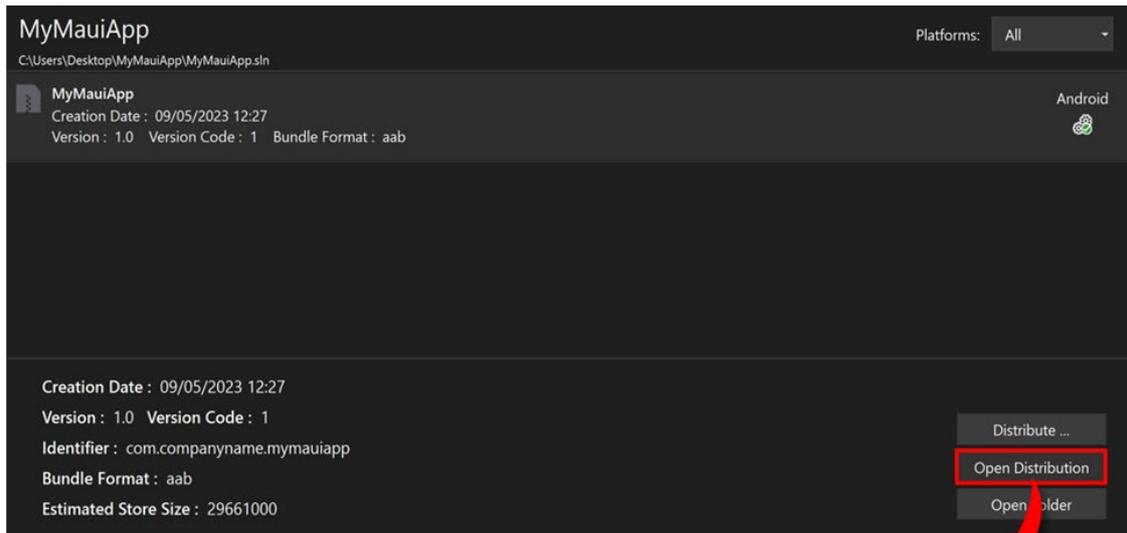


Figura 33. Paquete AAB de la aplicación firmado.

### 3. Publicación en Google Play

Por último, desde la plataforma de desarrollo de *Console*, se adjunta el archivo AAB y se configura las notas de versión. En este caso como es la primera vez que se va a lanzar la aplicación, la versión por defecto es 1(1.0). Para futuras actualizaciones de la app, esta versión *siempre* debe ser *diferente* a la última publicada y el paquete *siempre* tiene que estar *firmado con la misma clave de identidad*.

Una vez se ha realizado la revisión del archivo y se asegura que la versión y la firma es correcta, se procede con el lanzamiento a producción de la aplicación y finalmente esta quedara en revisión a la espera de ser publicada en Google Play. En las figuras 34 y 35 se puede observar el proceso descrito.

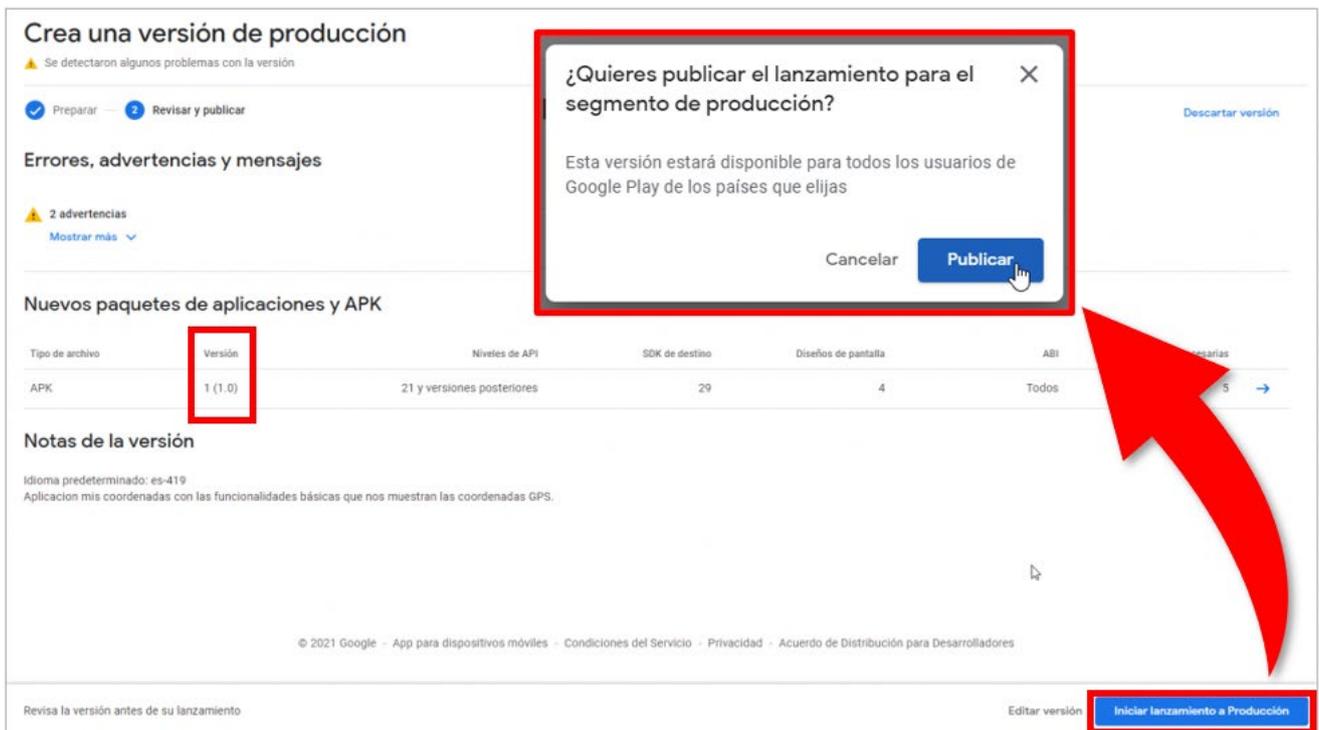


Figura 34. Lanzamiento a producción.

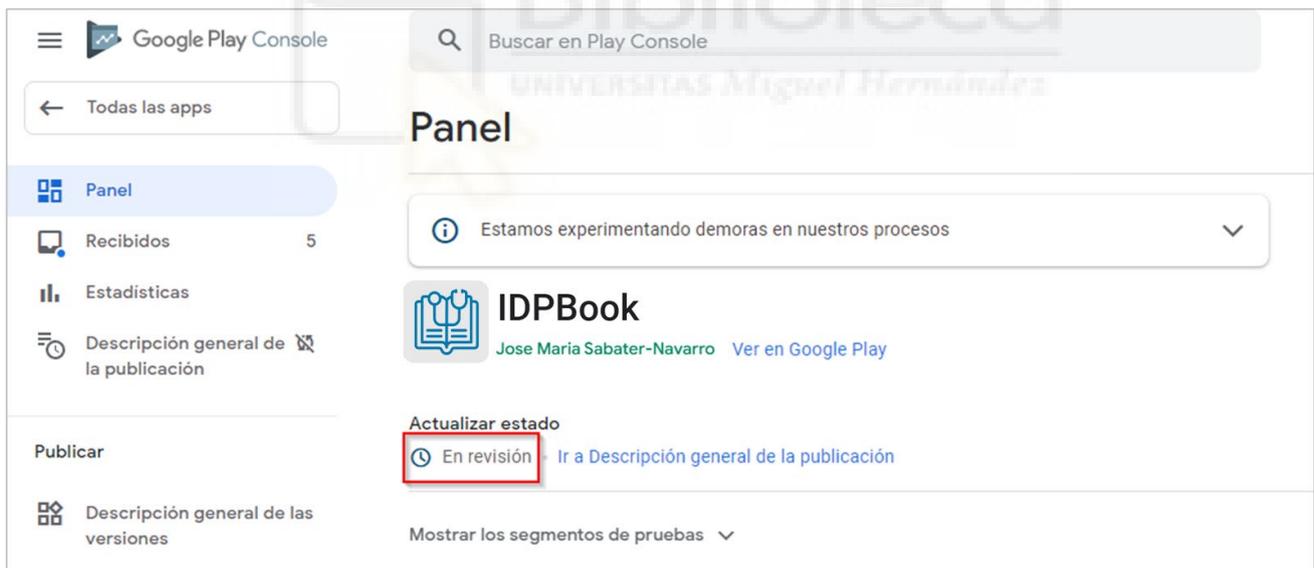


Figura 35. Configuración finalizada con la aplicación en estado de revisión.

En este punto, la configuración no se puede modificar y la publicación está en manos de un revisor de Google que realizará las pruebas necesarias y comprobará que toda la información suministrada es válida y determinará si la aplicación es apta para su publicación. Para cuentas de desarrollador que han sido creadas a partir del 9 de noviembre del 2023, es necesario realizar pruebas con al menos 20 “testers” durante 14 días antes de poder realizar el lanzamiento a producción. En

este caso no fue necesario realizar este procedimiento porque el perfil de desarrollador del grupo *Medical Robotics* fue creado antes de esta fecha y cuenta con varios proyectos publicados en la plataforma.

En la Figura 36 se observa una captura de la aplicación que fue publicada el día 15 de octubre del 2024.

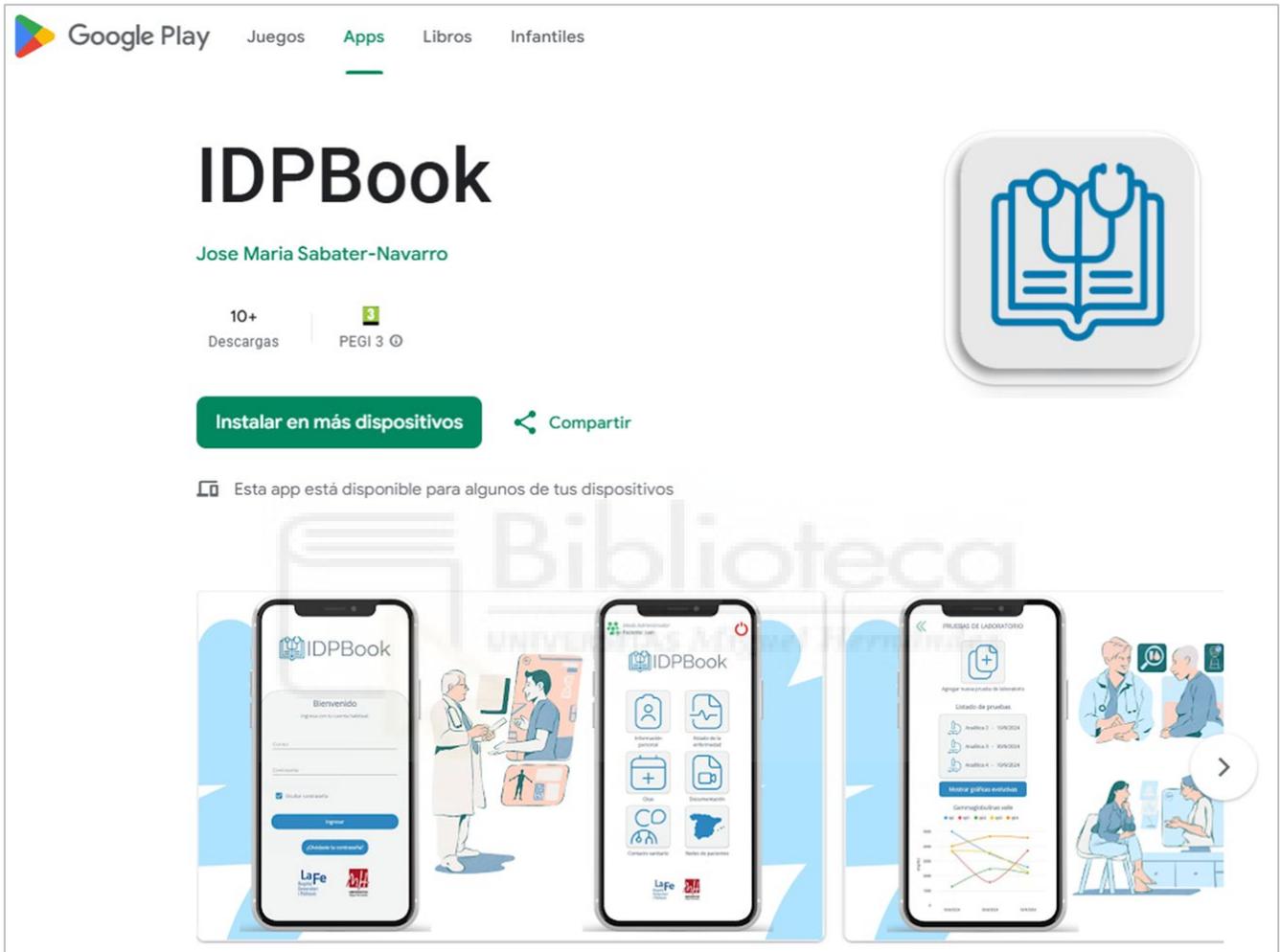


Figura 36. Aplicación IDPBook publicada en Google Play.

### 3.5.2. iOS

La distribución de aplicaciones iOS se realiza principalmente a través de *App Store*, utilizando la plataforma *App Store Connect*, disponible únicamente para desarrolladores registrados en el *Apple Developer Program*. Para empezar, es necesario generar un perfil de aprovisionamiento de distribución, el cual permite firmar digitalmente la aplicación y autorizar su instalación en dispositivos iOS, similar al *Keystore* en Android. Este certificado es distinto al *API Key* descrito en el apartado [2.3.2](#), aunque se puede utilizar el mismo certificado de autoridad para configurar el perfil de aprovisionamiento porque se trata de la misma aplicación.

Para crear este perfil es necesario definir algunos permisos desde la cuenta de desarrollador de *Apple*.

1. **Certificado de distribución:** este es necesario para compilar aplicaciones iOS en dispositivos *Apple*.
2. **Identificador de aplicación:** se trata de un *id* que identifica la aplicación dentro del ecosistema *Apple*. Para definirlo es necesario el certificado de distribución creado en el paso 1.
3. **Perfil de aprovisionamiento:** este perfil permite la firma digital de la aplicación para ser distribuida correctamente por *App Store Connect*.

Estos pasos se describen detalladamente en la guía<sup>7</sup> online sobre publicación de aplicaciones iOS de Visual Studio.

Una vez se hayan creado estos permisos, se procede con el registro de la aplicación en *App Store Connect*. Es el mismo proceso realizado en *Google Play Console*, pero el correspondiente a aplicaciones de iOS. También es un proceso que requiere de mucho tiempo y es necesario configurar muchos apartados para obtener la aprobación de distribución de la app. Una vez configurado el proyecto en *App Store Connect*, se procede con la compilación de la app a través de Visual Studio.

### Compilación y distribución en Visual Studio

Antes de iniciar el proceso de publicación se instala el certificado de distribución desde la configuración del proyecto, como se indica en la Figura 37.

---

<sup>7</sup> Fuente: Microsoft, "Publish a .NET MAUI iOS app to the App Store," [En línea]. Disponible: <https://learn.microsoft.com/en-us/dotnet/maui/ios/deployment/publish-app-store?view=net-maui-9.0> [Accedido: ene. 2025].

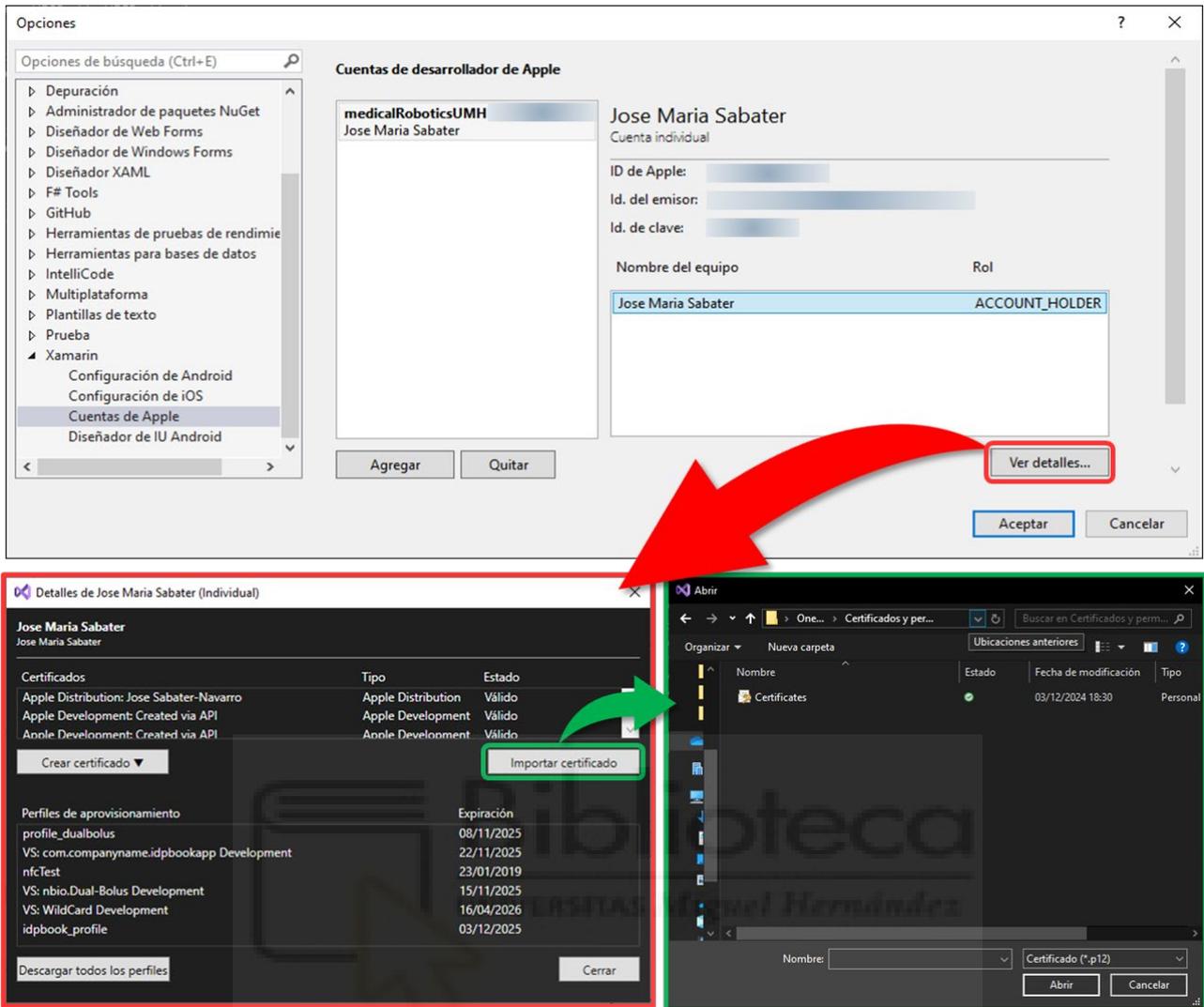


Figura 37. Instalación de certificado de distribución.

Una vez instalada la identidad de firma y el perfil de aprovisionamiento se puede realizar la firmar del paquete de la aplicación.

Ahora bien, para el proceso de publicación es necesario un ordenador Mac que se haya configurado como host de compilación remoto. Esto es necesario porque Visual Studio no puede empaquetar la aplicación directamente, necesita de un compilador como Xcode para hacerlo. El proceso de emparejamiento con Mac es sencillo siempre que se cuenten con los permisos y claves correspondientes.

Una vez el emparejamiento se ha realizado exitosamente se inicia el proceso de publicación. El primer paso es configurar el tipo de firma de la aplicación iOS, esto se realiza desde las propiedades del proyecto en el apartado Firma de lote o *Bundle Signing*. Aquí se debe asegurar que el tipo de aprovisionamiento sea manual, ya que el certificado de distribución se suministrará al momento de firma de la aplicación. El procedimiento anterior se representa en la Figura 38.

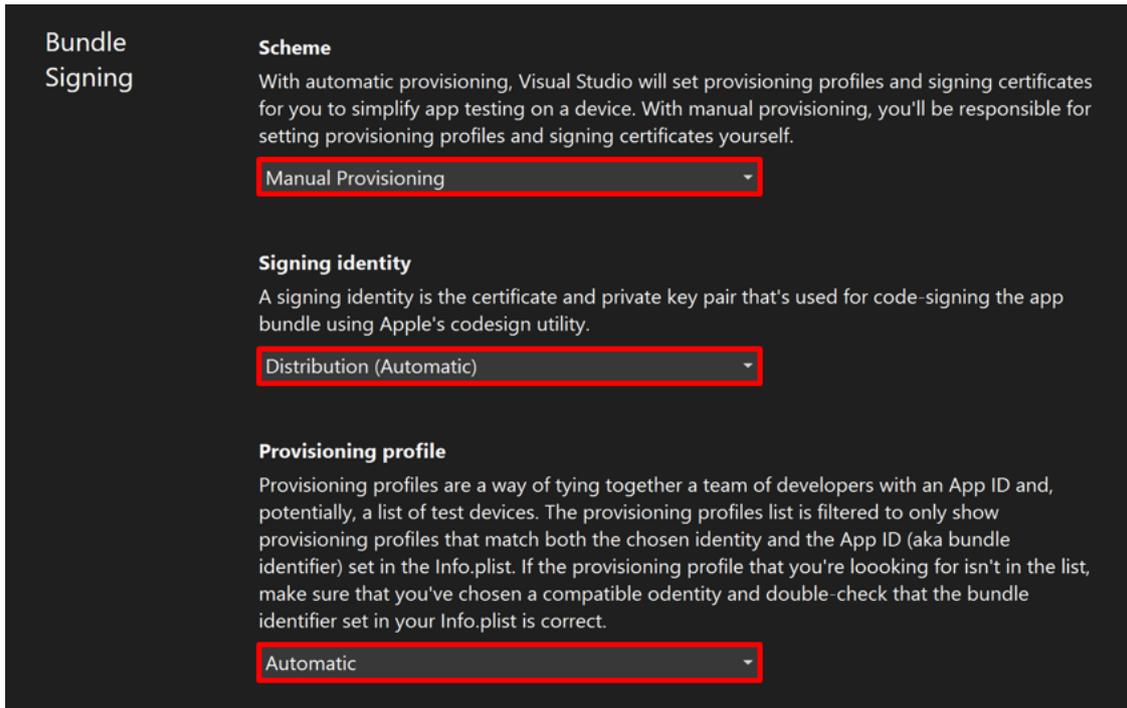


Figura 38. Configuración de la firma de lote de iOS.

Una vez configurado este apartado, el proceso es igual al realizado con aplicaciones Android, se inicia comprobando que el proyecto está configurado en modo *Release*, luego, desde el explorador de soluciones del proyecto, se procede con la publicación de la aplicación, tal como se muestra en el paso [2. Compilación y distribución en Visual Studio](#) para aplicaciones Android.

Luego, desde el explorador de archivos, seleccionando el paquete iOS publicado, se presiona en Distribuir. Se abrirá una ventana emergente donde se elige el perfil de aprovisionamiento para posteriormente ingresar la contraseña del certificado de distribución, tal como se muestra en la Figura 39.

Como resultado del paso anterior, se genera un archivo de extensión *.ipa* que contiene el paquete de distribución iOS de la aplicación, este se descarga y se procede con la publicación de la aplicación a través de *App Store Connect*.

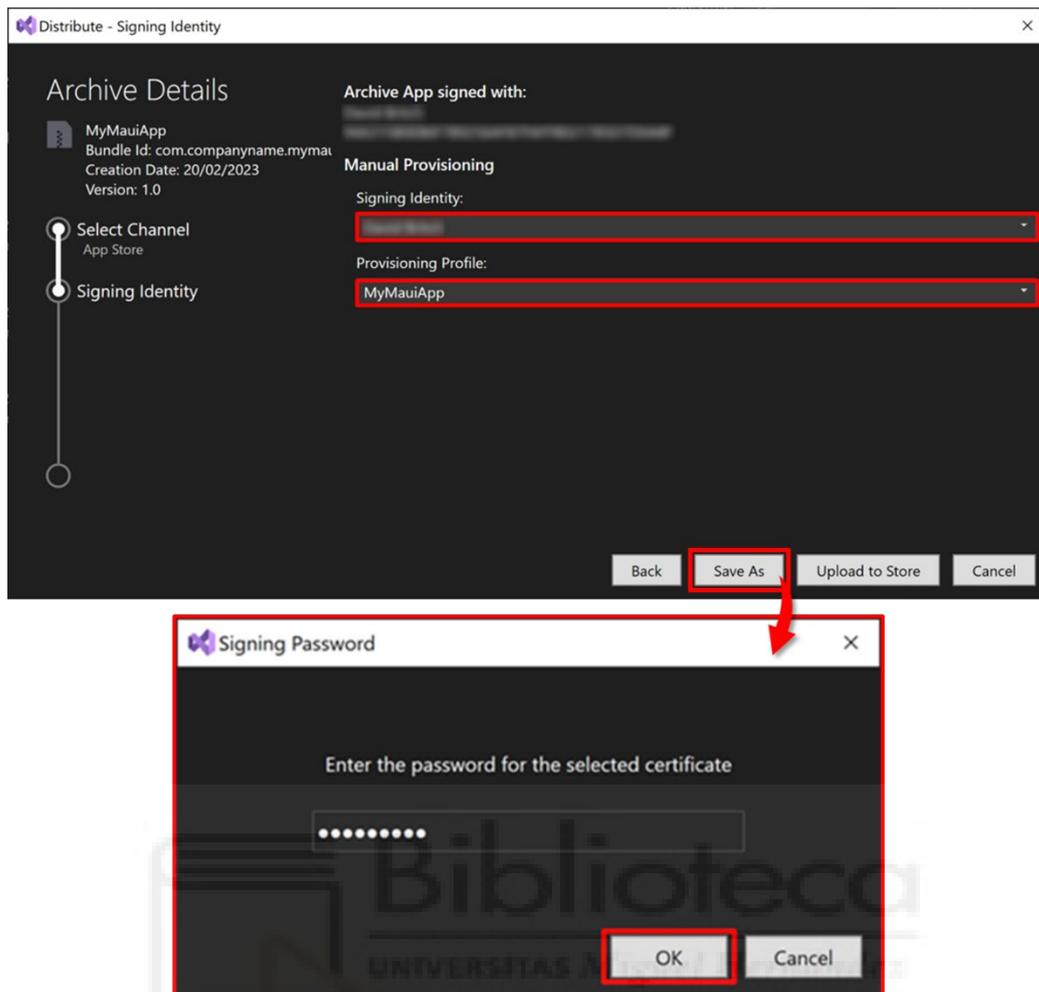


Figura 39. Firma de paquete de la aplicación iOS.

### **Publicación en App Store Connect**

Finalmente, el archivo de distribución (.ipa) se sube a App Store mediante la aplicación Transporter, una herramienta oficial de Apple que facilita la entrega de contenidos a plataformas como App Store o iTunes Store. Este proceso debe realizarse desde un ordenador Mac configurado con el perfil de desarrollador correspondiente, necesario para validar la firma digital incluida en el paquete. Transporter verifica que todos los certificados y perfiles de aprovisionamiento estén correctamente configurados, y una vez completado, deja el archivo listo para su gestión y publicación en App Store Connect.

La figura 40 muestra el estado actual (junio 2025) de la aplicación en App Store Connect. En este momento, la versión 2.0 se encuentra en proceso de revisión por parte de Apple, por lo que todavía no está disponible para su descarga en la tienda.

App Store Connect    Apps    Análisis    Tendencias    Informes    Negocio    Usuarios y acceso    Jose Maria Sabater

**Idpbook**    Distribución    TestFlight    Xcode Cloud

App para iOS    2.0 Pendiente de revisión    Guardar

**App para iOS Versión 2.0**

Puedes editar cierta información mientras la versión está pendiente de revisión. Para enviar una nueva compilación, debes eliminar esta versión del proceso de revisión.

Los componentes y metadatos siguientes aparecerán en la página de producto de tu app cuando los usuarios la instalen, y se utilizarán para los resultados de motores de búsqueda cuando tu app sea publicada.    Español (España)

**Vistas previas y capturas de pantalla**

Añadir capturas de pantalla precisas de tu app en los dispositivos más recientes puede ayudarte a representar la experiencia de usuario de la app. Ten en cuenta que utilizaremos estas capturas de pantalla para todos los tamaños de pantalla e idiomas. Las capturas de pantalla solo son necesarias para las apps de iOS y solo se utilizarán las 3 primeras en las hojas de instalación de la app.

iPhone    iPad    Ver todos los tamaños en "Gestor de recursos multimedia"

iPhone Pantalla de 6,9"    iPhone Pantalla de 6,5"

Texto promocional

Diario de salud electrónico para la autogestión y empoderamiento de los pacientes afectos de Inmunodeficiencias Primarias.

Figura 40. Estado de revisión de IDPBook en App Store Connect.

## **4. Validación de la aplicación**

En este capítulo se describe el procedimiento realizado para evaluar el funcionamiento y aceptación de la aplicación desarrollada. La validación se centró en la elaboración de un protocolo para verificar el comportamiento de las principales funcionalidades de la aplicación, así como en la creación de un manual de usuario que facilitó la comprensión y navegación por la interfaz.

### **4.1. Manual de usuario**

El manual de usuario es un documento de apoyo elaborado con el objetivo de facilitar la comprensión y el uso adecuado de la aplicación IDPBook por parte de los usuarios finales. Este manual fue diseñado con un lenguaje claro, accesible y neutral, considerando que puede estar dirigido a personas de distintos niveles de conocimiento digital, edades o contextos culturales. El contenido del manual se enfoca en explicar de forma detallada las principales funcionalidades de la aplicación, incluyendo el proceso de registro e inicio de sesión, la navegación entre las distintas secciones (como citas médicas, historial clínico, episodios, tratamientos y vacunas), así como las acciones básicas para introducir, consultar y actualizar la información médica del paciente. Este material se encuentra disponible en el anexo 8.3.

### **4.2. Protocolo de pruebas para validación de la aplicación**

#### **4.2.1. Objetivo**

Evaluar la usabilidad, funcionalidad y experiencia general de uso de la aplicación *IDPBook*, diseñada para el seguimiento clínico de pacientes con inmunodeficiencias primarias. El propósito es identificar fortalezas, posibles errores, oportunidades de mejora y validar si la solución cumple con los objetivos establecidos.

#### **4.2.2. Procedimiento**

- **Introducción:** Se explica brevemente el propósito del test, se indica el tiempo que dura el estudio y se entrega el manual de usuario.
- **Tareas:** Se solicita a cada participante realizar una serie de tareas básicas dentro de la aplicación, como: iniciar sesión, consultar una cita médica, registrar un episodio, acceder al historial médico, navegar entre las secciones de vacunas, tratamientos y analíticas.
- **Cuestionario:** Al finalizar el periodo de tiempo propuesto, se entrega el cuestionario para evaluar la experiencia de uso.
- **Análisis:** Se recopilan los datos y se analizan de forma cualitativa y cuantitativa.

### 4.3. Encuesta de experiencia de usuario

Como parte del proceso de validación y de experiencia de usuario de la aplicación, se aplicó una encuesta basada en la escala *uMARS (User Version of the Mobile Application Rating Scale)*, una herramienta estandarizada y validada que permite evaluar la calidad de aplicaciones móviles en el ámbito de la salud [33]. Esta escala proporciona una medida de 26 ítems que incluye cuatro categorías de calidad objetiva (participación, funcionalidad, estética e información) y una categoría de calidad subjetiva. También se incluye un apartado extra que busca medir el impacto percibido por los usuarios tras el uso de la aplicación durante un tiempo determinado [33].

Las categorías del cuestionario se dividen de la siguiente manera:

1. **SECCIÓN A. Participación:** divertido, interesante, personalizable, interactivo (envía alertas, mensajes, recordatorios, permite compartir etc.).
2. **SECCIÓN B. Funcionalidad:** funcionamiento de la app, facilidad de aprendizaje, navegación, transcurso lógico y diseño gestual.
3. **SECCIÓN C. Estética:** diseño gráfico, atractivo visual, combinación de colores y consistencia en estilo.
4. **SECCIÓN D. Información:** Contiene información de calidad (por ejemplo, textos, retroalimentación, medidas, referencias) de una fuente fiable.
5. **SECCIÓN E. Calidad subjetiva de la aplicación:** preguntas que incluyen conceptos como: recomendación de la app a otras personas, si pagarías por la aplicación, entre otras.
6. **SECCIÓN F. Impacto percibido:** apartado extra que intenta evaluar un posible impacto sobre los hábitos saludables del paciente, tras un periodo usando la aplicación.

La versión *uMARS* validada al español se encuentra en el anexo 8.4.

### 4.4. Resultados

En esta actividad de validación, participaron diez personas mayores de edad, cinco de ellos pacientes diagnosticados con alguna IDP, pertenecientes a la Unidad de Inmunodeficiencias del servicio de Medicina Interna del Hospital La Fe. Tres trabajadores sanitarios que habitualmente manejan pacientes afectados de IDP. Por último, dos programadores con conocimientos en desarrollo de tecnologías

para la salud que pertenecen a distintos grupos de investigación de la Universidad Miguel Hernández. Todos los participantes realizaron el cuestionario tras un mes de uso de la aplicación IDPBook.

#### 4.4.1. Metodología

Los datos del cuestionario se presentan como un promedio aritmético simple. El resultado de cada pregunta se obtiene de acuerdo con la ecuación (1).

$$\text{Promedio por ítem} = \frac{1 * n_1 + 2 * n_2 + 3 * n_3 + 4 * n_4 + 5 * n_5}{\text{Total de respuestas válidas}} \quad (1)$$

Donde,  $n_1, n_2, \dots, n_5$ , representan la cantidad de participantes que eligieron cada valor. El total de respuestas válidas no incluye la opción N/A. Las respuestas de los participantes para realizar el cálculo del promedio mencionado se encuentran al final del anexo 8.4.

Los datos se han segmentado en dos grupos: pacientes y otros usuarios. Esta división permite observar con claridad la experiencia de las personas que realmente usarán la aplicación día a día. Por su parte, la opinión de los sanitarios o programadores, brindan un concepto más técnico y relativo al contenido e implementación de la aplicación.

Los resultados de la valoración se consignan en la Tabla 3. Los datos se han organizado de acuerdo con las secciones de la escala *uMARS* y para cada una se indica la valoración media por sección y por pregunta del cuestionario.

CATEGORÍA / Pregunta	RESPUESTA PROMEDIO	
	Pacientes	Otros usuarios
<b>SECCIÓN A: PARTICIPACIÓN</b>	<b>3,76</b>	<b>4</b>
Entretención	3	3,4
Interés	4,2	4,2
Personalización	3,4	3,6
Interacción	3,6	3,8
Público objetivo	4,6	5
<b>SECCIÓN B: FUNCIONALIDAD</b>	<b>4,4</b>	<b>4,75</b>
Rendimiento	4,8	4,8
Facilidad de uso	4	5
Navegación	4,6	5
Diseño gestual	4,2	4,2
<b>SECCIÓN C: ESTÉTICA</b>	<b>4,13</b>	<b>4,13</b>
Disposición	4,4	4
Gráficos	4,2	4,2
Atractivo visual	3,8	4,2

<b>SECCIÓN D: INFORMACIÓN</b>	<b>4,06</b>	<b>4,3</b>
Calidad de la información	4,33	4
Cantidad de información	3,5	4
Información visual	4	4,2
Credibilidad	4,4	5
<b>SECCIÓN E: CALIDAD SUBJETIVA</b>	<b>3,9</b>	<b>4,05</b>
Recomendación	4,4	5
Uso esperado en 12 meses	4,2	4
Pago por la app	2,8	3,6
Calificación general	4,2	3,6
<b>SECCIÓN F: IMPACTO</b>	<b>3,69</b>	<b>4,27</b>
Conciencia	3,8	4,2
Conocimiento	3,6	4,2
Actitudes	3,6	4,2
Intención de cambio	3,8	4,2
Búsqueda de ayuda	4	4,4
Cambio de comportamiento	3,33	4,4

*Tabla 3. Valoración de la aplicación según escala uMARS.*

#### **4.4.2. Descripción**

La siguiente descripción pretende relacionar los resultados de cada pregunta con las opciones de respuesta de la escala uMARS.

- 1. Sección A:** con respecto al ítem de participación, ambos grupos tienen una opinión similar. Los participantes consideran que (Entretenimiento) el grado de diversión es suficiente para entretener al usuario durante un breve tiempo, (Interés) es moderadamente interesante, capta la atención del usuario un rato, (Personalización) permite una personalización básica con adecuada funcionalidad, (Interacción) ofrece características de interacción básicas con adecuada funcionalidad y (Público) el contenido de la aplicación está perfectamente dirigido, sin problemas aparentes.
- 2. Sección B:** según los resultados sobre la funcionalidad de la aplicación, los participantes opinan que (Rendimiento) la aplicación funciona bien con algún problema menor o insignificante, (Facilidad) es fácil de usar con instrucciones claras, intuitiva y simple, (Navegación) tiene un flujo muy lógico, fácil, claro e intuitivo y (Diseño) cuenta con un diseño principalmente consistente con problemas menores.
- 3. Sección C:** con relación a la estética de la aplicación, los participantes tienen una opinión muy parecida, consideran que la aplicación (Disposición) tiene una disposición clara con elementos fáciles de seleccionar, localizar o leer, (Gráficos) contiene gráficos de calidad y resolución alta con un diseño visual apropiado y

consistente en estilo y, por último, (Atractivo visual) que la aplicación es atractiva visualmente con un diseño profesional.

4. **Sección D:** en cuanto a la información presentada en la aplicación, los grupos presentan opiniones similares, frente a la calidad de la información consideran que es relevante, apropiada, coherente y correcta. Sobre la cantidad de información la opinión de los pacientes considera que es aceptable, pero ni exhaustiva ni concisa, mientras que los demás participantes consideran que ofrece un amplio abanico de información con algunos detalles innecesarios. En relación con la información visual los participantes consideran que es principalmente clara, lógica y correcta con algunos fallos insignificantes. Por último, con respecto a la credibilidad de la información, el grupo de pacientes considera que la información proviene de una fuente que puede ser fiable, mientras que, para los demás participantes, la información es totalmente segura y proviene de una fuente fiable y especializada.

Por otro lado, de acuerdo con la escala *uMARS*, la valoración media se obtiene del promedio aritmético de las secciones A, B, C y D [33]. En este caso los valores obtenidos por cada grupo de participantes son de 4,1 para pacientes y 4,3 para otros usuarios. En ese orden de ideas, la valoración global de la aplicación es de 4,2.

5. **Sección E:** En cuanto a la calidad subjetiva de la aplicación, ambos grupos consideran que recomendarían el uso de la aplicación a muchas personas. Con relación al tiempo de uso en los próximos 12 meses, los participantes consideran que la usarían entre 10 y 50 veces. Frente a la opción de pagar por obtener la aplicación, los pacientes mantienen una posición más cercana a definitivamente no pagar, mientras que, los demás usuarios tienen en postura más flexible. Sobre la valoración general de la app, los pacientes tienden a verla como una de las mejores que han usado. Mientras que, el resto de los participantes la califica como correcta.
6. **Sección F:** con respecto a los primeros cuatro enunciados sobre, si es probable que la aplicación promueva la conciencia, el conocimiento, las actitudes y la intención de cambio hacia una conducta más saludable; los pacientes consideran estar de acuerdo con estas afirmaciones, mientras que los demás participantes mantienen una postura más cercana estar en total acuerdo con estas ideas. Por otro lado, frente a si la aplicación promueve la búsqueda de ayuda para abordar la conducta saludable, la opinión es muy similar en los grupos y es más cercana a un total acuerdo con la afirmación. Finalmente, al valorar si el uso de la aplicación implicó un cambio de comportamiento hacia la conducta saludable, los pacientes mantienen una postura neutra, ni de acuerdo ni en desacuerdo, mientras que el resto de los participantes se encuentran más cerca de un total acuerdo con la afirmación.

La Figura 41 presenta un gráfico de cajas y bigotes que resume la validación de la aplicación *IDPBook* mediante la escala *uMARS*, comparando las respuestas de dos grupos: pacientes y usuarios clasificados como "otros" (profesionales sanitarios y desarrolladores). El objetivo del gráfico es visualizar de forma comparativa la distribución, tendencia y variabilidad de las valoraciones en cada una de las seis categorías evaluadas: participación, funcionalidad, estética, información, calidad subjetiva e impacto.

Para la construcción de la gráfica, los datos fueron agrupados según el tipo de participante y sección. Se calculó el promedio de cada ítem a partir de las respuestas individuales en una escala de 1 a 5, siguiendo la ecuación (1). Este enfoque permite representar no solo la media de cada categoría, sino también la dispersión y los valores atípicos, facilitando una interpretación más precisa de la percepción de cada grupo sobre la calidad de la aplicación.

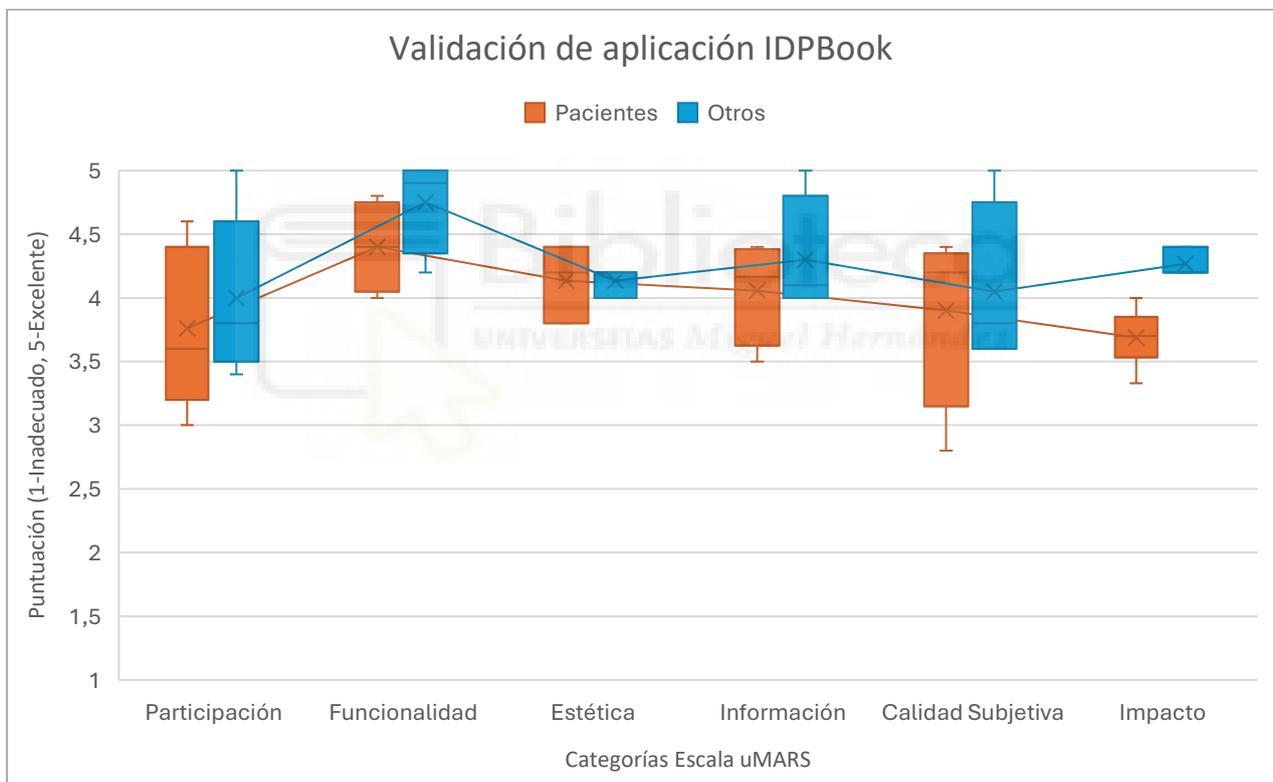


Figura 41. Diagrama de cajas y bigotes con resultados de la validación.

Se puede observar que la media de los grupos en cada sección es muy parecida, a excepción de la categoría de impacto, donde los sanitarios y programadores perciben un mayor impacto en la conducta saludable que los pacientes. También se observa una mayor variabilidad en las categorías de Participación y Calidad subjetiva. Mientras que, en las secciones de Estética e Impacto la opinión es más consistente, no hay mucha dispersión.

## 5. *Discusión*

Los resultados obtenidos con la escala *uMARS* indican que IDPBook es valorada positivamente tanto por pacientes como por profesionales sanitarios y desarrolladores. La media global de 4,2 refleja una aceptación general favorable en términos de funcionalidad, diseño, utilidad y contenido.

Si bien la valoración media de ambos grupos es similar, en la Figura 41 se puede apreciar la ligera diferencia entre las tendencias de cada categoría, donde la puntuación media de los pacientes está entre 3,5 y 4,5, mientras que los demás participantes se ubican entre 4,0 y 5,0, demostrando que, en términos generales, hay una mayor aprobación de la app por parte de estos últimos frente a la opinión de los pacientes. Este nivel de aceptación por parte de los sanitarios y desarrolladores refleja que la aplicación ha sido diseñada con una base técnica robusta y eficiente. Por otro lado, gracias al punto de vista de los pacientes, se reconoce el hecho que la aplicación debe mejorarse y ofrecer un mejor servicio a los usuarios finales.

Una de las categorías mejores valoradas fue la Funcionalidad, con una puntuación promedio de 4,4 por parte de los pacientes y 4,75 entre los demás participantes. Esto respalda las herramientas utilizadas durante el desarrollo de la aplicación y sugiere que la app cumple con estándares de funcionamiento óptimos, es intuitiva, fácil de navegar y presenta un diseño funcional con pocos errores. Estos aspectos son clave para garantizar una buena experiencia de uso, especialmente en entornos de salud donde la facilidad de acceso y comprensión son esenciales.

La sección sobre Información también obtuvo una puntuación alta en ambos grupos, aunque se observan diferencias en la percepción de la cantidad y credibilidad del contenido. Mientras que los pacientes consideran la información como adecuada pero no extensa, los demás participantes valoran su amplitud y la reconocen como fiable y de calidad. Esto sugiere que la app logra transmitir confianza, aunque podría ajustarse mejor a los distintos niveles de comprensión del usuario.

En cuanto a la Estética, la puntuación media fue igual (4,13) para ambos grupos. Esto indica que la interfaz visual es clara, atractiva y profesional. La disposición de los elementos, el uso de gráficos y los colores elegidos fueron percibidos como consistentes, lo cual facilita la navegación y mejora la experiencia del usuario.

Respecto a la calidad subjetiva, los usuarios manifestaron su intención de seguir utilizando la aplicación en los próximos meses y recomendarla a otros. Sin embargo, la disposición a pagar por ella fue baja, especialmente entre pacientes. Esto puede explicarse por la expectativa de gratuidad en herramientas digitales.

para la salud, sobre todo cuando forman parte de un sistema público o de apoyo clínico.

La categoría con mayor diferencia entre grupos fue el impacto percibido. Mientras que los sanitarios y desarrolladores le otorgaron una media de 4,27, los pacientes se situaron en 3,69. Aunque ambos grupos reconocen que la aplicación puede generar conciencia, fomentar actitudes positivas y facilitar la búsqueda de ayuda, los pacientes no percibieron un cambio claro en su comportamiento. Esta diferencia puede deberse al corto periodo de prueba (un mes), que quizá no haya sido tiempo suficiente para generar cambios sostenibles en hábitos o conductas de salud.

La Figura 41 también muestra una mayor variabilidad en categorías como la participación y la calidad subjetiva, las cuales incluyen preguntas que dependen directamente de la percepción y expectativa de cada usuario. Por ejemplo, la primera pregunta de la categoría de Participación trata sobre el entretenimiento ofrecido. Este apartado que no es tan relevante para este trabajo, puesto que la aplicación no se diseñó como una herramienta de entretenimiento sugiere que para algunas personas puede representar una característica importante en el proceso de familiarización de una aplicación y a largo plazo puede ser determinante para lograr un hábito de uso en el paciente.

Para explorar si la aplicación puede influir en la adherencia a tratamientos, se tomó como referencia un estudio realizado con la app Flexlg, de la cual se habló en la sección 1.3 sobre tecnologías actuales para el seguimiento de inmunodeficiencias primarias. En este artículo [22] se analizó la relación entre el nivel de satisfacción del usuario y su adherencia a la terapia con inmunoglobulinas. Aunque no se utilizó la escala uMARS, las herramientas aplicadas (SUS<sup>8</sup>/UEQ<sup>9</sup>) mostraron una experiencia de usuario positiva y un alto nivel de satisfacción. Estos resultados se vincularon directamente con una tasa de adherencia a tratamientos del 99,7 %. Esto refuerza la idea de que una app bien diseñada y fácil de usar puede facilitar el seguimiento médico y mejorar los resultados del tratamiento. En el caso de IDPBook, aunque no se midió la adherencia de forma directa, los resultados obtenidos en la escala uMARS (media de 4,2) sugieren que la aplicación tiene potencial para convertirse en un apoyo real para los pacientes, gracias a su buena aceptación y a la valoración positiva por parte de los usuarios participantes.

---

<sup>8</sup> *System Usability Scale (SUS)*: cuestionario estandarizado para medir la usabilidad percibida de un sistema o producto.

<sup>9</sup> *User Experience Questionnaire (UEQ)*: es una encuesta diseñada para medir la experiencia de usuario con un producto o servicio digital y evaluar la calidad de su interacción.

## **6. Conclusiones y trabajos futuros**

Este trabajo tuvo como objetivo desarrollar IDPBook, una aplicación multiplataforma para personas con inmunodeficiencias primarias. Funciona como un diario de salud electrónico que permite registrar datos clínicos, gestionar citas, acceder a información médica y mantener contacto con profesionales de salud. La aplicación disponible para dispositivos Android y iOS, busca mejorar el seguimiento médico y fomentar la participación del paciente en su tratamiento.

Antes de empezar con el desarrollo de la aplicación, se realizó una investigación sobre las tecnologías actuales en el ámbito de la salud, especialmente orientadas a la gestión de enfermedades crónicas. Se identificó que la mayoría de las soluciones existentes no están adaptadas a las necesidades específicas de personas con IDP, lo que ayudó a definir las funcionalidades clave que debía integrar la aplicación.

IDPBook incorpora Firebase para la autenticación, Firestore como base de datos remota, y funciones “On the edge” para procesamiento local de datos. Se desarrolló en Visual Studio usando .NET MAUI, con arquitectura MVVM, lo que permitió una estructura modular, eficiente y fácil de mantener. El resultado es una app fluida, funcional y centrada en el usuario.

Para respaldar el trabajo realizado con IDPBook, se realizó un proceso de validación en la que participaron 10 personas, las cuales utilizaron la aplicación durante 1 mes y respondieron a un cuestionario diseñado para medir la experiencia de usuario. Este cuestionario está basado en la escala uMARS, con la cual se obtuvo una puntuación media global de 4.2/5. Este resultado refleja una buena aceptación por parte de los usuarios y confirma que la aplicación ofrece una experiencia positiva en términos de funcionalidad, estética, contenido y utilidad. La evaluación incluyó a pacientes, profesionales sanitarios y desarrolladores, lo que permitió comparar perspectivas complementarias. Los resultados mostraron una percepción favorable de la app en todos los grupos, aunque se notó que los pacientes perciben menos impacto inmediato, algo comprensible debido al corto tiempo del estudio.

Al comparar estos resultados con otras apps similares, se encontró algo interesante: una buena experiencia de uso puede mejorar la adherencia a los tratamientos. En el caso de la aplicación Flexlg, la satisfacción del usuario se relacionó con una adherencia casi total. Aunque con IDPBook no evaluó este indicador de forma directa, las altas puntuaciones en funcionalidad y calidad subjetiva sugieren que la app podría tener un efecto positivo en esta necesidad del paciente.

Como parte del proceso de validación, también se diseñó un protocolo de prueba y un manual de usuario. Gracias a esto, los participantes reportaron que la app era fácil de entender y que las instrucciones proporcionadas eran claras. Esto refuerza la importancia de acompañar el desarrollo tecnológico con materiales de apoyo, como guías o manuales, especialmente en herramientas orientadas a pacientes con necesidades particulares, como las personas afectadas con IDP.

A lo largo de este proyecto y gracias a la opinión de los participantes, se identificaron aspectos que pueden mejorarse para ofrecer una mejor experiencia y satisfacción al usuario. También se reconoce que un estudio con una muestra más amplia y extensa en el tiempo podría ofrecer una visión más completa sobre el impacto real de la aplicación en la calidad de vida de los usuarios.

En síntesis, IDPBook demuestra que es posible utilizar aplicaciones móviles a para mejorar el seguimiento clínico, empoderar al paciente con su enfermedad y optimizar los recursos sanitarios. Este proyecto representa un primer paso hacia soluciones digitales más inclusivas, centradas en las necesidades del paciente y orientadas a mejorar el manejo de enfermedades crónicas poco visibilizadas como las inmunodeficiencias primarias.

Algunos trabajos futuros que se pueden desarrollar a partir de los resultados obtenidos en este proyecto son:

- Mejorar la funcionalidad de la aplicación incluyendo más opciones de registro de información médica, herramientas para registrar la toma de medicamentos y generar reportes sobre la adherencia del tratamiento. Igualmente, incluir secciones más entretenidas o llamativas en pro de mejorar la experiencia de usuario y aumentar el tiempo de uso de la aplicación.
- Realizar una validación más extensa con más participantes, durante un periodo más largo, para obtener resultados más robustos.
- Con una visión a largo plazo, utilizar los datos recogidos por la aplicación para identificar marcadores que ayuden a predecir complicaciones o episodios de los pacientes. Por otro lado, utilizar esta información para realizar diagnóstico preventivo de enfermedades crónicas como las inmunodeficiencias primarias.

## 7. Referencias

- [1] M. M. Coates et al., "Burden of disease among the world's poorest billion people: An expert-informed secondary analysis of Global Burden of Disease estimates," *PLoS One*, vol. 16, no. 1, ene. 2021.
- [2] J. Herrick, "Clean problems: Simplicity, complexity and the contemporary history of global noncommunicable disease prioritisation," *Soc. Sci. Med.*, vol. 295, feb. 2022.
- [3] M. O. Omotayo et al., "The rise of non-communicable diseases: A global health review of challenges and prevention strategies," *J. Public Health Res.*, vol. 13, no. 2, abr. 2024.
- [4] J. van Olmen et al., "Implementation barriers for mHealth for non-communicable diseases management in low- and middle-income countries: a scoping review and field-based views from implementers," *Wellcome Open Res.*, vol. 5, abr. 2020.
- [5] E. Osei et al., "Mobile health applications for disease screening and treatment support in low-and middle-income countries: A narrative review," *Digit. Health*, vol. 7, pp. 1-15, 2021.
- [6] C. Hernández-Martínez et al., "Conceptos básicos de las inmunodeficiencias primarias," *Rev. Alerg. Mex.*, vol. 63, no. 2, pp. 180-189, abr.-jun. 2016.
- [7] A. Budreviciute et al., "Management and prevention strategies for non-communicable diseases (NCDs) and their risk factors," *Front. Public Health*, vol. 8, oct. 2020.
- [8] R. Barouki et al., "Developmental origins of non-communicable disease: Implications for research and public health," *Environ. Health*, vol. 11, jun. 2012.
- [9] World Health Organization, "NCD mortality data," WHO Global Health Observatory, nov. 30, 2024. [En línea]. Disponible: <https://www.who.int/data/gho/data/themes/topics/topic-details/GHO/ncd-mortality> [Accedido dic. 2024]
- [10] Institute for Health Metrics and Evaluation, "Global Burden of Disease Study 2021," IHME, 2021. [En línea]. Disponible: <https://www.healthdata.org/research-analysis/gbd> [Accedido dic. 2024]
- [11] European Food Information Council, "¿Qué son las enfermedades no transmisibles y cómo podemos reducir nuestro riesgo?" [En línea]. Disponible:

<https://www.eufic.org/es/vida-sana/articulo/que-son-las-enfermedades-no-transmisibles-y-como-podemos-reducir-nuestro-riesgo> [Accedido ene. 2025]

[12] InfoLAR, "Enfermedades no transmisibles y cómo reducir los riesgos," [En línea]. Disponible: <https://www.infoilar.org/es/articulos/enfermedades-no-transmisibles-y-como-reducir-los-riesgos> [Accedido ene. 2025]

[13] P. Soler-Palacín, "Inmunodeficiencias primarias," en Congreso de Actualización Pediatría 2020, Madrid: Lúa Ediciones 3.0., pp. 311-320, 2020.

[14] N. Rezaei et al., "Introduction on primary immunodeficiency diseases," en Primary Immunodeficiency Diseases, N. Rezaei, A. Aghamohammadi, and L. Notarangelo, Eds. Berlin, Germany: Springer, 2017.

[15] N. Klangkalya, T. A. Fleisher, and S. D. Rosenzweig, "Diagnostic tests for primary immunodeficiency disorders: Classic and genetic testing," Allergy Asthma Proc., vol. 45, no. 5, pp. 355-363, sep. 2024.

[16] L. Yel, "Selective IgA deficiency," J. Clin. Immunol., vol. 30, no. 1, pp. 10-16, ene. 2010.

[17] S. G. Tangye et al., "Human inborn errors of immunity: 2022 update on the classification from the International Union of Immunological Societies Expert Committee," J. Clin. Immunol., vol. 42, no. 7, pp. 1473-1507, oct. 2022.

[18] MSD Manuals, "Overview of immunodeficiency disorders," dic. 3, 2024. [En línea]. Disponible: <https://www.msdmanuals.com/professional/immunology-allergic-disorders/immunodeficiency-disorders/overview-of-immunodeficiency-disorders> [Accedido ene. 2025]

[19] D. Inupakutika et al., "Facilitating the development of cross-platform mHealth applications for chronic supportive care and a case study," Int. J. Med. Inform., vol. 134, feb. 2020.

[20] M. Hood et al., "What do we know about mobile applications for diabetes self-management? A review of reviews," J. Behav. Med., vol. 39, no. 6, pp. 981-994, dic. 2016.

[21] European Commission, "Register of commission documents," 2014. [En línea]. Disponible: <http://ec.europa.eu/transparency/regdoc/rep/1/2014/ES/1-2014-219-ES-F1-1.Pdf> [Accedido ene. 2025]

- [22] T. Pindi Sala et al., "Contribution of Flexig mobile application to assess adherence of patients treated with immunoglobulins in chronic diseases," *J. Allergy Clin. Immunol. Glob.*, vol. 3, no. 1, art. 100173, 26-sep-2023.
- [23] R. Romero-Jiménez et al., "Design and implementation of a mobile app for the pharmacotherapeutic follow-up of patients diagnosed with immune-mediated inflammatory diseases: eMidCare," *Front. Immunol.*, vol. 13, art. 915578, 28-jul-2022.
- [24] V. Modell et al., "Global report on primary immunodeficiencies: 2018 update from the Jeffrey Modell Centers Network on disease classification, regional trends, treatment modalities, and physician reported outcomes," *Immunol. Res.*, vol. 66, no. 3, pp. 367-380, jun. 2018.
- [25] W. Y. Moon and S. J. Powis, "Does Natural Killer Cell Deficiency (NKD) Increase the Risk of Cancer? NKD May Increase the Risk of Some Virus Induced Cancer," *Front. Immunol.*, vol. 10, jul. 2019.
- [26] N. Schleinitz et al., "Natural killer cells in human autoimmune diseases," *Immunology*, vol. 131, no. 4, pp. 451-458, dic. 2010.
- [27] A. R. Gennery et al., "Autoimmunity in combined immunodeficiency," en *Translational Immunology*, N. Rezaei, Ed., vol. 3, pp. 97-127, Academic Press, 2022.
- [28] H. Abolhassani et al., "Global systematic review of primary immunodeficiency registries," *Expert Rev. Clin. Immunol.*, vol. 16, no. 7, pp. 717-732, 2020.
- [29] I. K. Chinn et al., "Diagnostic interpretation of genetic studies in patients with primary immunodeficiency diseases: A working group report of the Primary Immunodeficiency Diseases Committee of the American Academy of Allergy, Asthma and Immunology," *J. Allergy Clin. Immunol.*, vol. 145, no. 1, pp. 46-69, ene. 2020.
- [30] A. Gómez, "Bases de datos en aplicaciones móviles para sistema operativo Android," *Rev. Cient. Univ. Tecnol. Intercont.*, vol. 3, no. 2, pp. 45-60, 2019.
- [31] T. Taleb et al., "Mobile edge computing: A survey on enabling technologies, applications, and research challenges," *IEEE Commun. Surv. Tutor.*, vol. 19, no. 3, pp. 1657-1681, Q3 2017.
- [32] StatCounter, "Mobile operating system market share worldwide," may. 13, 2025. [En línea]. Disponible: <https://gs.statcounter.com/os-market-share/mobile/worldwide> [Accedido may. 2025]

- [33] R. Martín-Payo et al., "Spanish adaptation and validation of the User Version of the Mobile Application Rating Scale (uMARS)," *J. Am. Med. Inform. Assoc.*, vol. 28, no. 12, pp. 2681-2686, nov. 2021.
- [34] G. Mattison et al., "The influence of wearables on health care outcomes in chronic disease: Systematic review," *J. Med. Internet Res.*, vol. 24, no. 7, jul. 2022.
- [35] R. M. Torrente-Rodríguez et al., "SARS-CoV-2 RapidPlex: A graphene-based multiplexed telemedicine platform for rapid and low-cost COVID-19 diagnosis and monitoring," *Matter*, vol. 3, no. 6, pp. 1981-1998, 2020.
- [36] M. van Well, *Near real-time continuous remote monitoring of vital signs of patients during intravenous administration of medication at home*, M.Sc. thesis, Mechanical Engineering Dept., Delft University of Technology, Países Bajos, 2022. [En línea]. Disponible: <https://repository.tudelft.nl/record/uuid:31448b75-b111-4052-b565-e3d03c324266>
- [37] Y. Gao et al., "A flexible multiplexed immunosensor for point-of-care in situ wound monitoring," *Sci. Adv.*, vol. 7, no. 21, may. 2021.
- [38] C. Charlie et al., "Introducing the Chronic Disease mHealth App Quality Evaluation Framework (CHRONIQ)," en *2024 Conference on Information Communications Technology and Society (ICTAS)*, Durban, Sudáfrica, pp. 32-37, 2024.

## 8. Anexos

### 8.1. Diseño de una aplicación multiplataforma con .NET MAUI

En este anexo se explica brevemente cómo crear una aplicación con .NET MAUI, en especial la ejecución en dispositivos Android y iOS, utilizando Visual Studio en Windows.

1. El primer paso es descargar e instalar Visual Studio, para el año 2025 la versión más actualizada es 2022. Una vez instalada y abierta, el primer paso es crear un nuevo proyecto, tal como se muestra en la Figura 1.

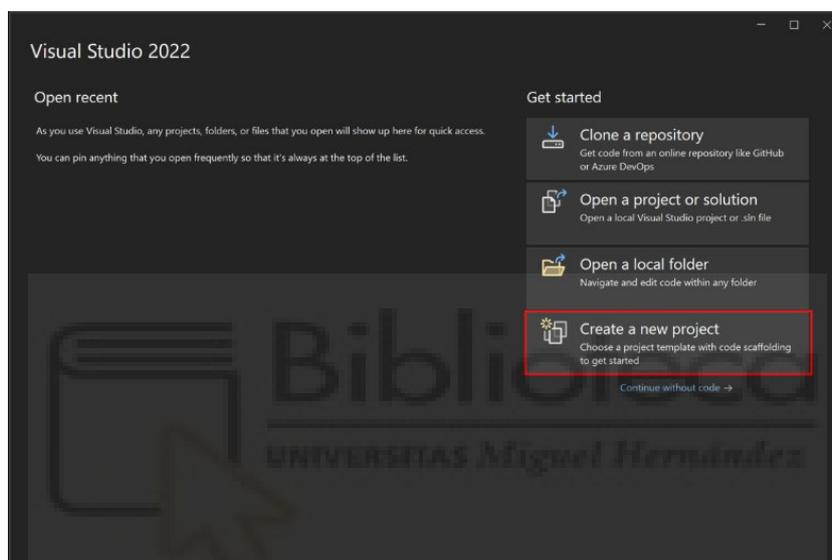


Figura 1. Crear nuevo proyecto con Visual Studio y .NET MAUI.

2. Se abrirá una segunda ventana donde se podrá seleccionar la plantilla correspondiente al tipo de proyecto que se desea iniciar. Se selecciona la plantilla .NET MAUI App y luego se da a “Siguiente”. Ver Figura 2.

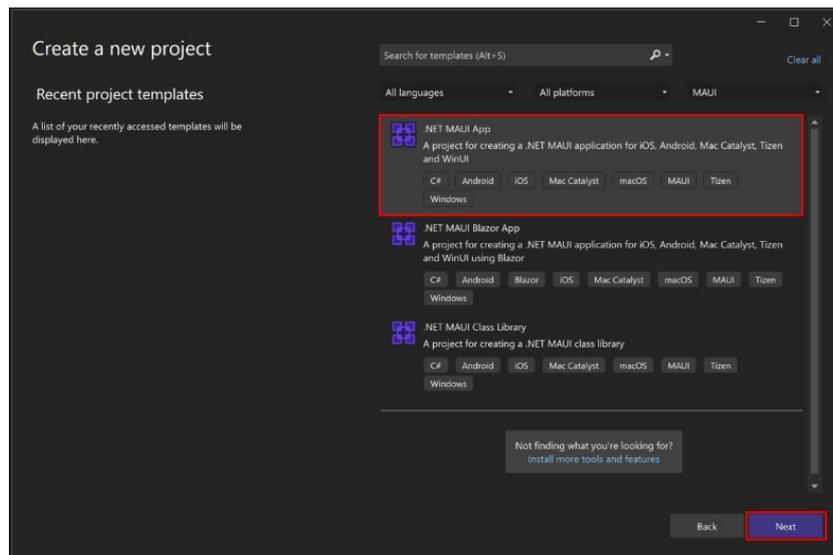


Figura 2. Selección de tipo de proyecto y plantilla.

3. En la ventana de configuración del proyecto, se asigna un nombre y la ubicación del proyecto, luego se oprime “Siguiente”.

4. En la ventana de Información adicional, se elige la versión de .NET a la que estará dirigida la aplicación, tal como se muestra en la captura de la Figura 3. Este paso es muy importante, ya que las librerías que se desean utilizar deben ser compatibles con la versión que se escoja. Para el año 2025, .net 8 es la versión con mayor soporte y compatibilidad con librerías de terceros, mientras que la versión 9 es la que tendrá soporte a largo plazo, aunque, actualmente la compatibilidad con algunas librerías no está totalmente garantizada.

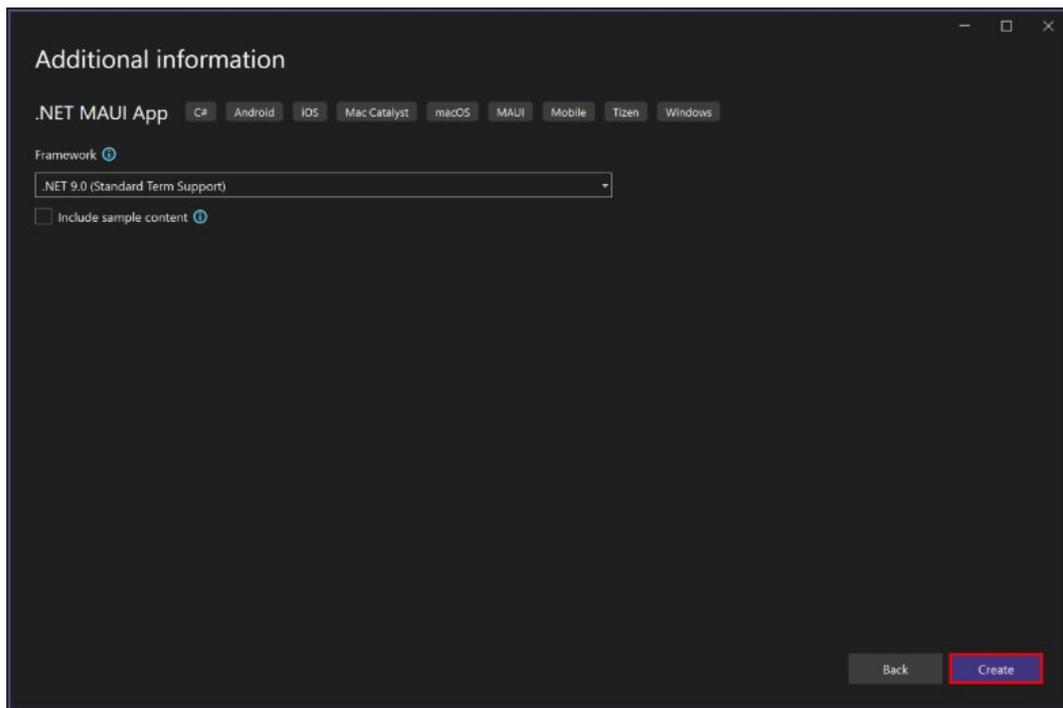


Figura 3. Selección de la versión de .NET del proyecto.

Una vez seleccionada, se oprime a crear y Visual Studio empezará a restaurar todas las dependencias e instalar los paquetes predeterminados de una aplicación básica.

5. Una vez finalizada la restauración, el proyecto ya está listo para ejecutarse. De manera predeterminada, el destino de depuración es el mismo equipo Windows, sin embargo, en la lista de destinos (Figura 4) de depuración se encuentra los emuladores y dispositivos locales disponibles para compilar y ejecutar la aplicación.

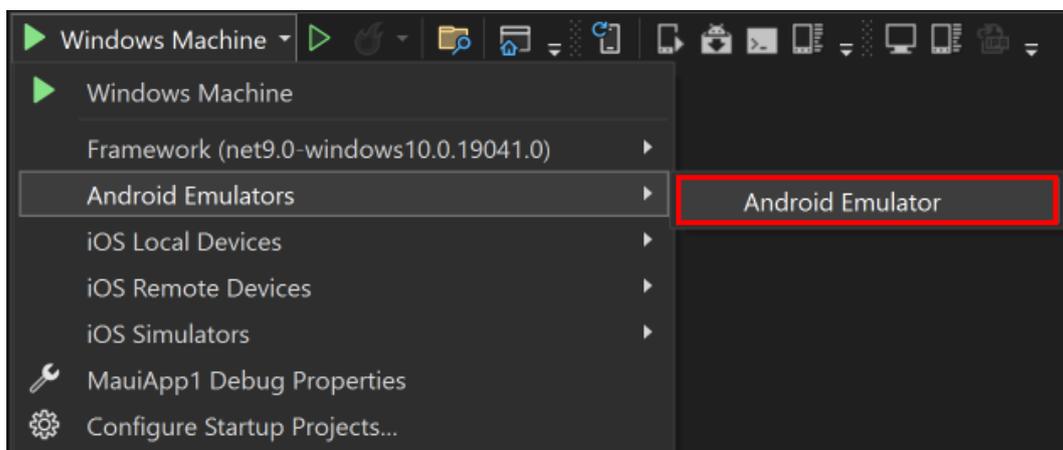
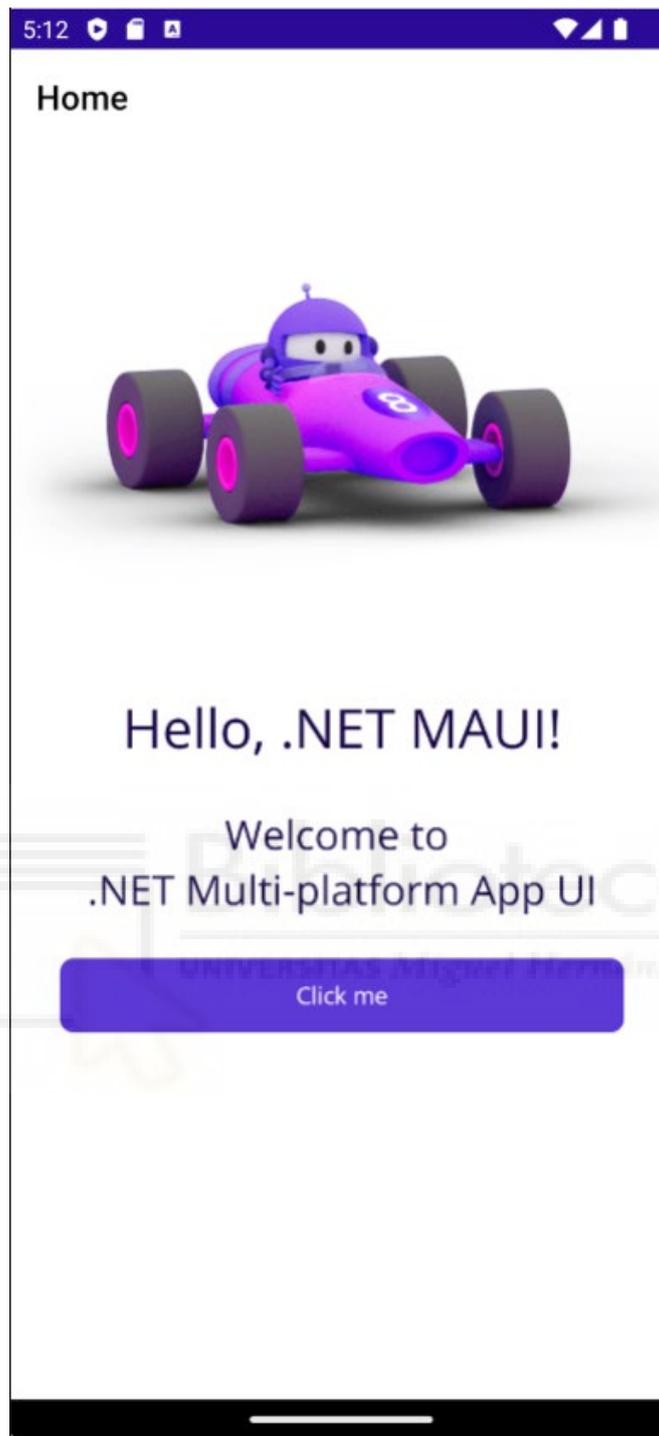


Figura 4. Lista de dispositivos y emuladores.





*Figura 6. Aplicación desplegada en dispositivos Android.*

8. Para la edición de código, en .NET MAUI existe la recarga activa durante la depuración, es decir, se puede cambiar la interfaz de usuario, mientras la aplicación se está ejecutando y la interfaz se actualizará automáticamente. Esta propiedad automática está disponible solo para la interfaz de usuario que se desarrolla en los archivos XAML o archivos de contenido gráfico. En la Figura 7, el archivo *MainPage.xaml*, representa la interfaz de la ventana principal correspondiente con la captura de las figuras 6 y 8.

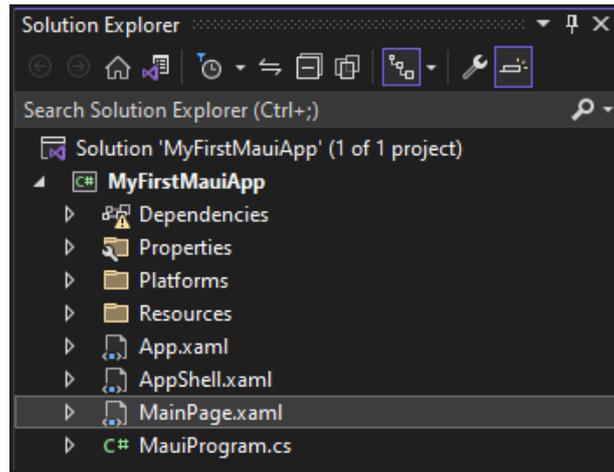


Figura 7. Explorador de soluciones del proyecto.

9. Los cambios realizados en el archivo *MainPage.xaml* durante la depuración se pueden observar en tiempo real. Por ejemplo, si se cambia el texto del *label* que dice: “Hello, .NET MAUI!” por “Hello, World”, se observará un cambio como el de la Figura 8.

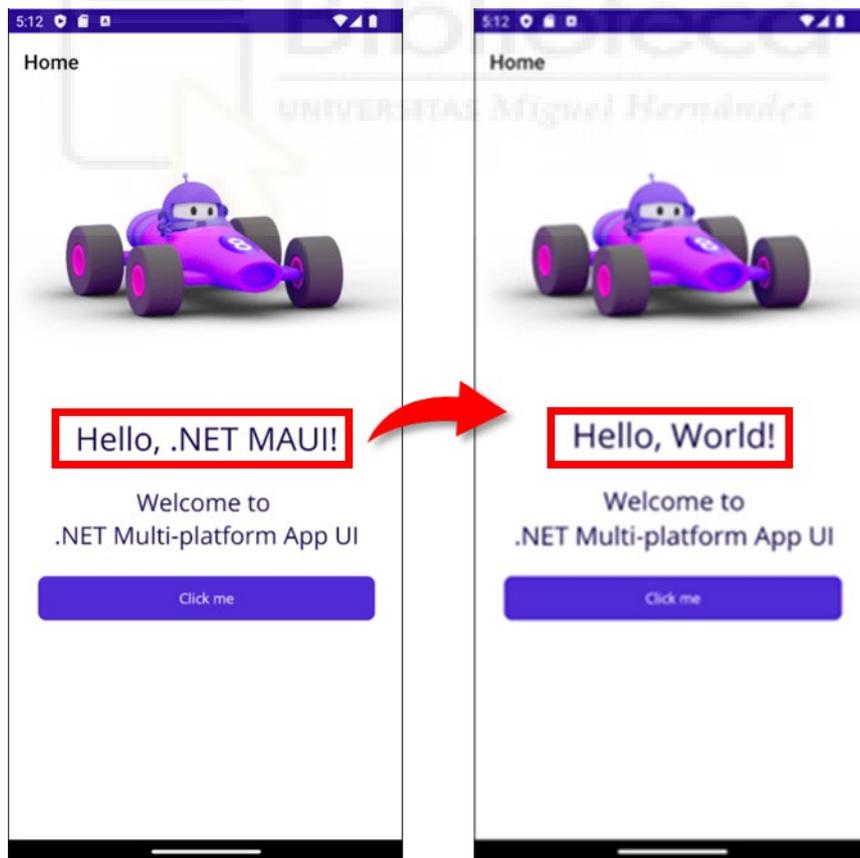


Figura 8. Cambios en tiempo real en modo depuración.

10. Por otra parte, para la edición de la lógica de la aplicación se debe acceder a los archivos .cs de cada interfaz, conocidos como “code behind”, hace referencia al código encargado de ejecutar las acciones necesarias para el funcionamiento de la aplicación. Los archivos (.cs) están anidados a cada interfaz (.xaml), para acceder a estos desde el Explorador de soluciones, se da clic en la flecha del archivo xaml que deseamos editar y este desplegara el “code behind” correspondiente o también se puede hacer clic derecho y en el menú desplegado seleccionar “Ver código”. El lenguaje que se utiliza en estos archivos es C#. En la Figura 9, se muestra la lógica de funcionamiento del botón “Click me”, simplemente es un contador que incrementa 1 unidad cada vez que se presiona el botón.

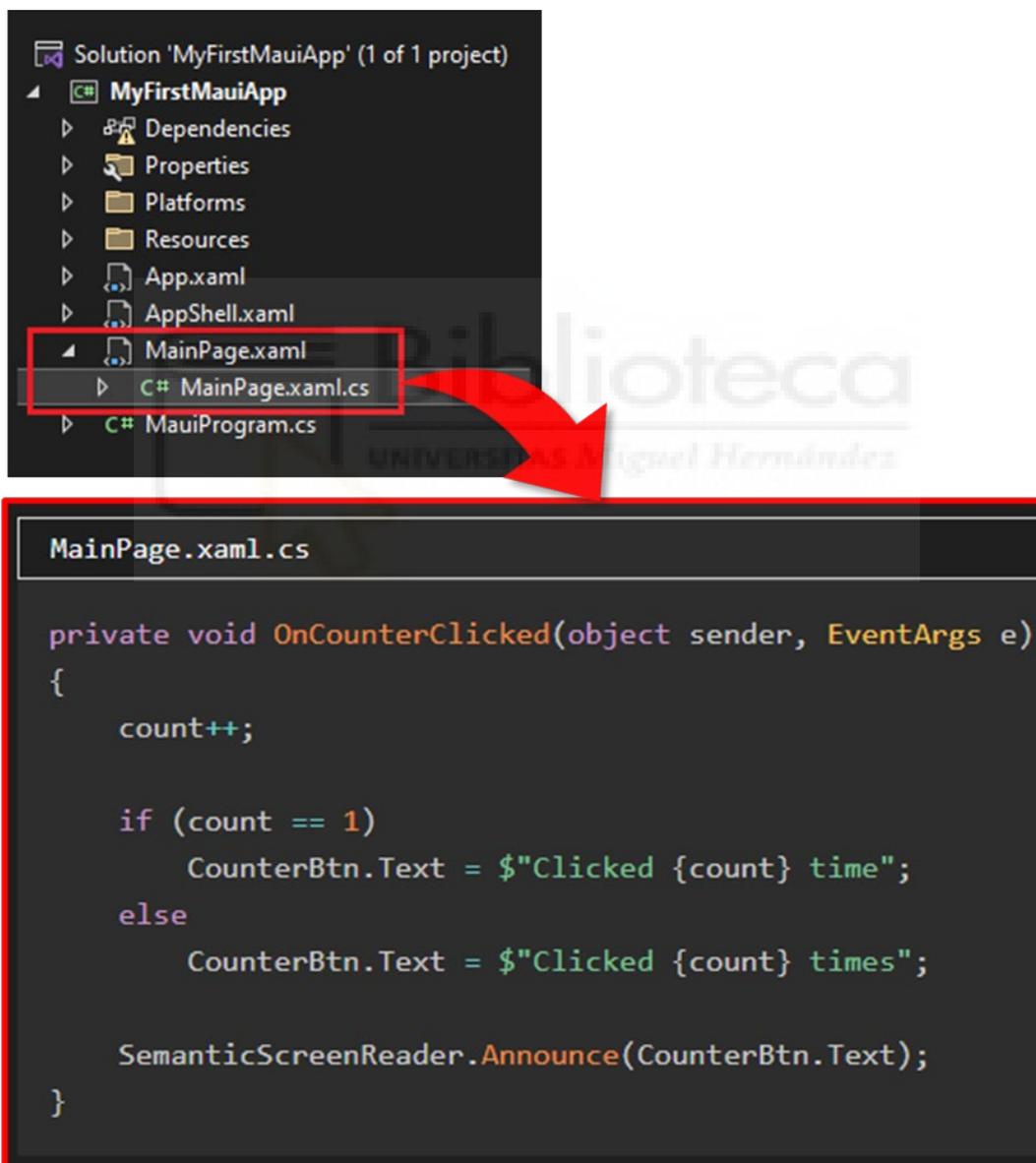


Figura 9. Acceso al “Code behind” desde el Explorador de soluciones.

11. Los cambios en el “code behind” también se pueden actualizar mientras la aplicación se está ejecutando. A diferencia de la interfaz que se actualiza automáticamente, para actualizar archivos C# se debe presionar el botón de ‘Recarga activa’ ubicado en la barra de herramientas, corresponde a un icono de una llama, como se puede ver en la siguiente captura.

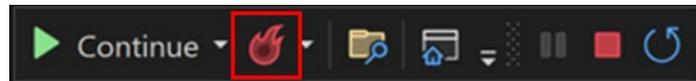


Figura 10. Activación de recarga activa para actualización del "Code behind" en modo depuración.

## Dispositivos iOS

12. Como se mencionaba anteriormente, para ejecutar aplicaciones en dispositivos iOS es necesario contar con una cuenta *Apple Developer* y tener activa la membresía de desarrollador (*Apple Developer Program*). El siguiente paso es asignar una *API Key* para que Visual Studio pueda utilizar a los certificados necesarios para el uso de dispositivos iOS. En la Figura 11 se muestra un ejemplo de cómo se ve una *API Key* creada desde la web de *App Store Connect*.

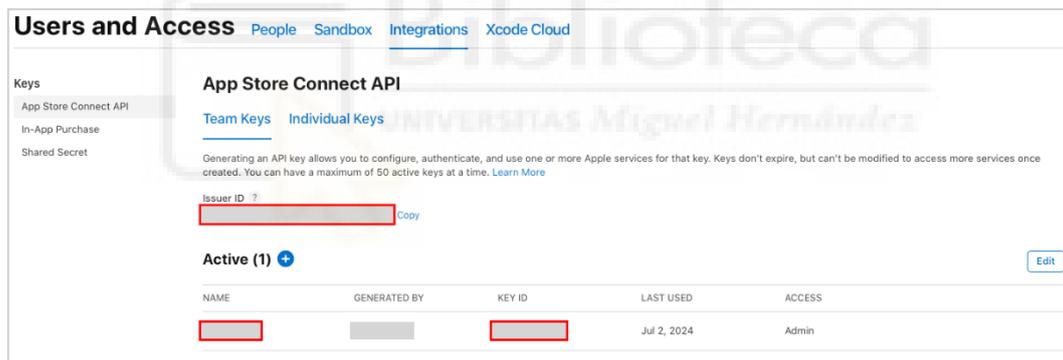


Figura 11. Asignación de API Key en la web de App Store Connect.

13. Una vez asignada la llave y el archivo .p8 descargado en el ordenador, se conecta el dispositivo iOS, no sin antes habilitar el modo desarrollador en el equipo, luego dirigirse a la lista de destinos de depuración y seleccionar el dispositivo conectado, tal como se muestra en la Figura 12.

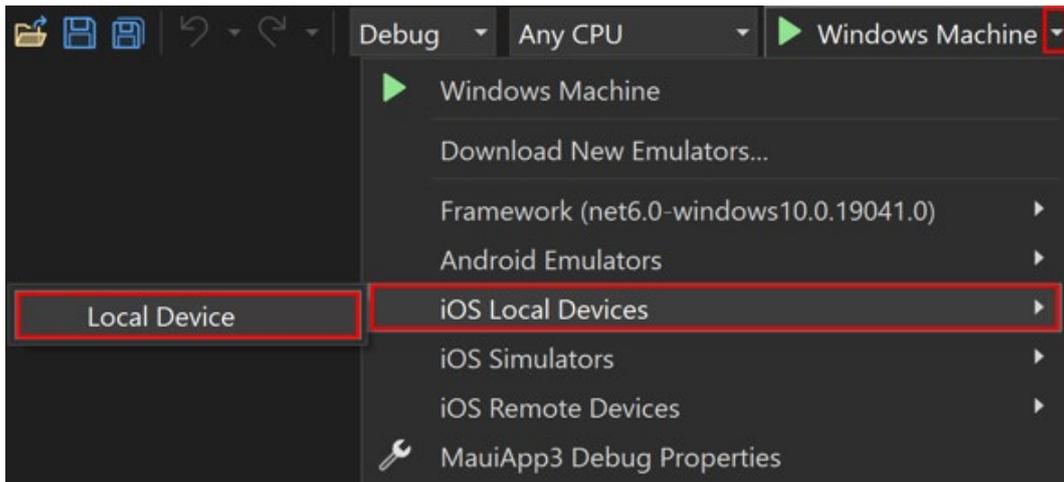


Figura 12. Lista de dispositivos iOS locales.

14. Acto seguido, se debe configurar el “Hot restart” que hace posible la depuración de aplicaciones en este tipo de dispositivos Apple. El proceso inicia como se muestra en la Figura 13, al presionar siguiente se inicia con la configuración. Se comprueba la instalación de iTunes, si el dispositivo conectado tiene habilitado el modo desarrollador, entre otros apartados.

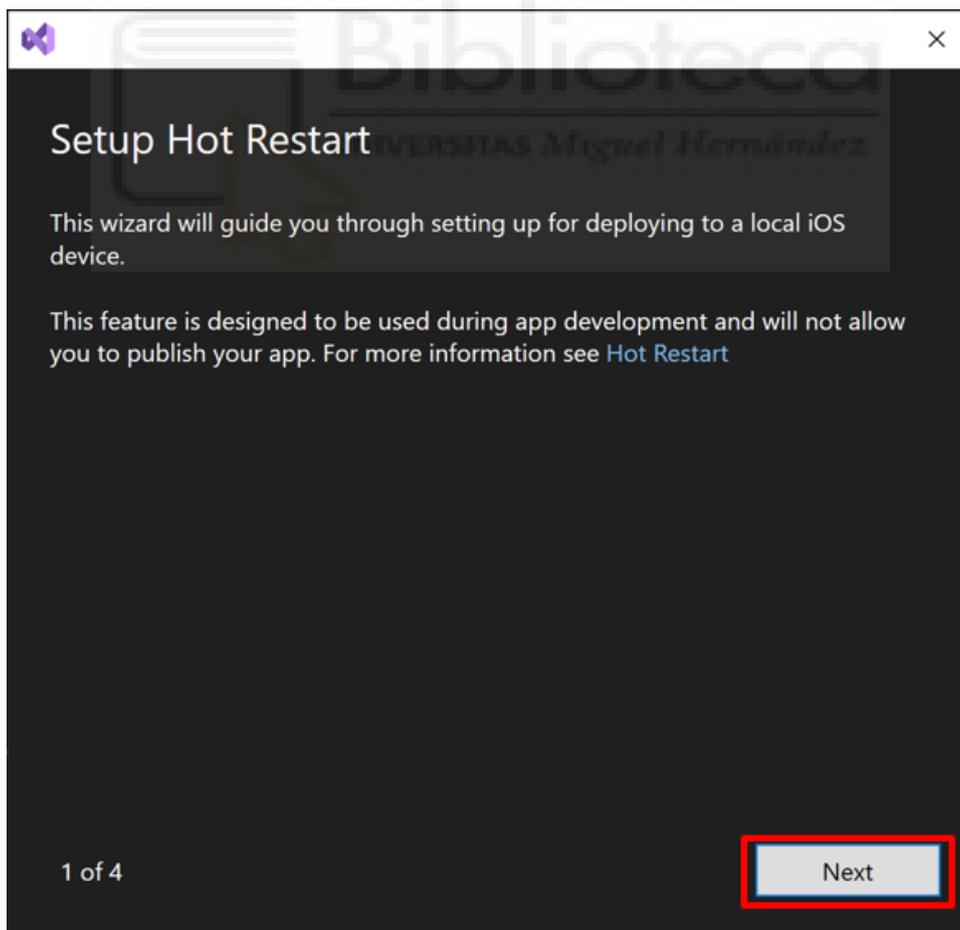


Figura 13. Configuración de "Hot Restar" para dispositivos iOS.

15. Al final de esta configuración es donde se debe utilizar la *API Key* descargada (archivo .p8) e ingresar los datos correspondientes a la llave. Los campos que se muestran en la Figura 14, se establecen durante la creación de la *API Key* en *App Store Connect*.

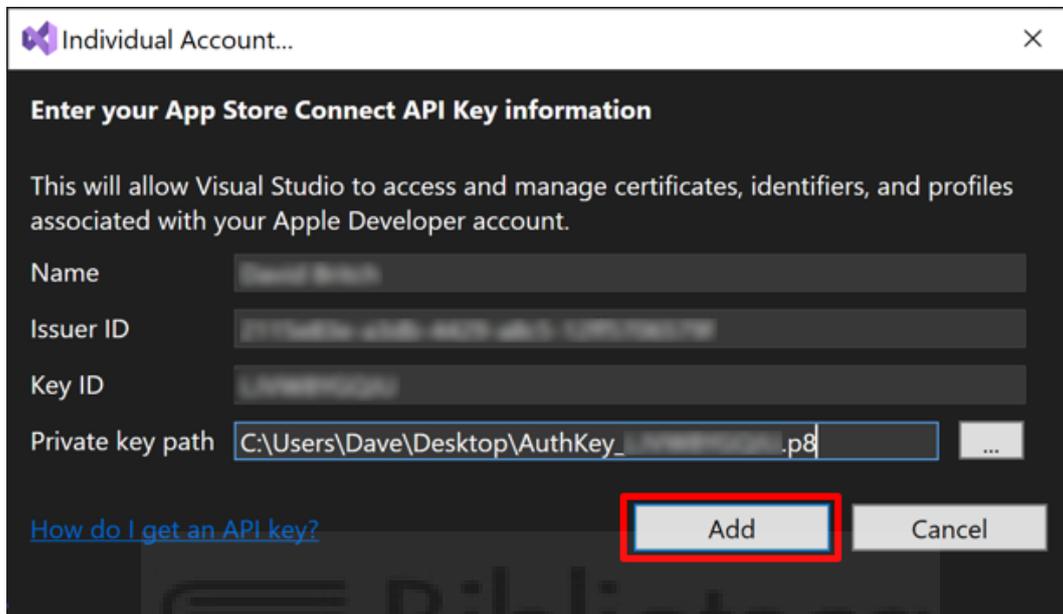


Figura 14. Confirmación de credenciales de la *API Key*.

16. Una vez finalizada la configuración, el nombre del dispositivo iOS aparecerá en la lista de depuración y podrá ejecutarse la aplicación, tal como se muestra en la Figura 15.



Figura 15. Barra de herramientas para compilación en dispositivos iOS.

17. Es importante mencionar que la recarga activa durante la depuración también está habilitada en dispositivos iOS, sin embargo, para la actualización del “code behind”, no se realiza con el botón de recarga activa (el icono de la llama), en este caso se debe reiniciar la aplicación con el botón de “Restart”, tal como se indica en la Figura 16.



Figura 16. Reinicio de aplicación en dispositivos iOS.

## 8.2. Estructura de la base de datos remota de la aplicación IDPBook

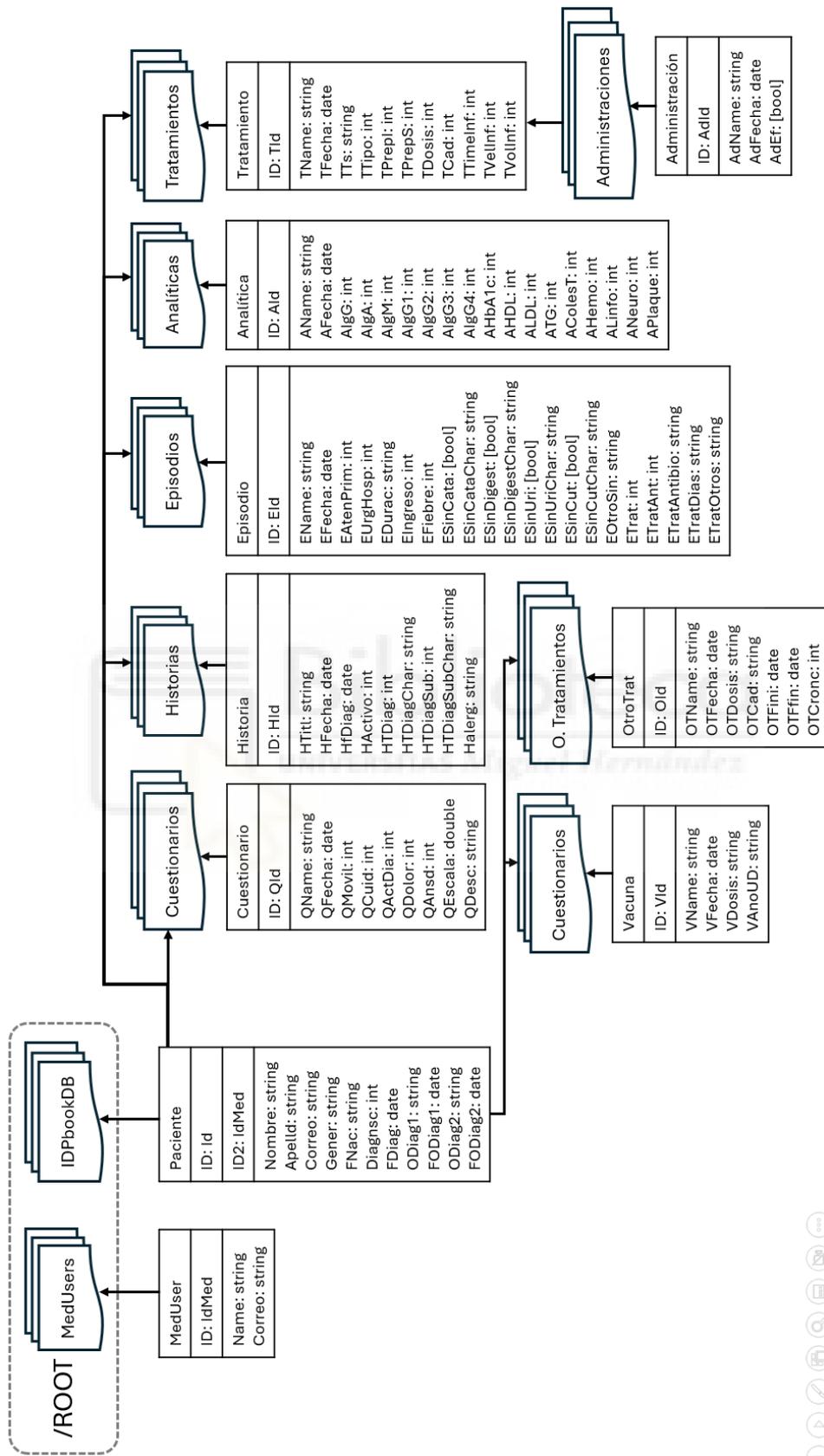


Figura 1. Estructura de la base de datos No SQL en Firestore.

A continuación, se detallan las variables que conforman el sistema descrito en la imagen anterior (Figura 1). Estas variables han sido organizadas según los seis bloques funcionales que estructuran la interfaz de la aplicación móvil, y representan los datos gestionados dentro de cada entidad del modelo de base de datos. La descripción incluye el nombre de cada variable, su tipo de dato y la función que cumple dentro del sistema. Esta organización permite comprender con mayor claridad cómo se almacena y gestiona la información clínica de los pacientes, y facilita su implementación técnica durante el desarrollo de la aplicación.

### **1. Área de registro e información personal** (rellenado por el médico en el primer contacto)

- Nombre
- Apellidos
- Sexo (desplegable hombre/mujer/no definido)
- Fecha de nacimiento
- Edad (automatizada al rellenar la anterior)
- Fecha de diagnóstico
- Diagnóstico (desplegable):
  - Inmunodeficiencia Común Variable
  - Agammaglobulinemia
  - Déficit de IgA
  - Déficit de Subclases de Inmunoglobulinas
  - Déficit de respuesta a antígenos específicos.
  - Inmunodeficiencia Combinada.
  - Enfermedad Granulomatosa Crónica
  - Microdelección 22q.
  - Síndrome de Wiscott-Aldrich.
  - Síndrome de Hiper-IgE
  - Síndrome de Hiper-IgM
  - ALPS
  - Candidiasis mucocutánea crónica
  - Otros (campo abierto donde se pueda escribir)

### **2. Estado de la enfermedad**

#### **2.1. Episodios**

##### **A) Historial médico** (rellenado por el médico)

- Año de diagnóstico:
- ¿Sigue activo? Sí/No

- Tipo de diagnóstico -> desplegable con las siguientes opciones, que a su vez se despliegan en más opciones:

- Enfermedad infecciosa: (desplegable):
  - Neumonía
  - Meningitis
  - Osteomielitis
  - Infección gastrointestinal
  - Celulitis o piel y partes blandas
  - Infección vía urinaria con fiebre
  - Infecciones oportunistas
  - Otros (campo libre)
- Enfermedad hematológica (desplegable)
  - Anemia Inmuno hemolítica.
  - PTI
  - Síndrome de Evans
  - Neutropenia autoinmune
  - Linfadenopatías benignas
  - Infiltración esplénica
  - Otros (campo libre)
- Enfermedad del aparato digestivo (desplegable)
  - Enfermedad Inflamatoria intestinal.
  - Celiaquía
  - Celiaquía-like
  - Hiperplasia Nodular Linfoide.
  - Colitis microscópica.
  - Otros (campo libre)
- Enfermedad pulmonar (desplegable)
  - Bronquiectasias
  - Neumopatía intersticial
  - Neumatoceles
  - Afectación pleural
  - Otros (campo libre)
- Enfermedad hepática (desplegable)
  - Hiperplasia nodular regenerativa
  - Hipertensión Portal
  - Cirrosis
  - Otro (campo libre)
- Enfermedad oncológica (desplegable)
  - Linfoma
  - Leucemia
  - Adenocarcinoma gástrico

- Otros (campo libre)
- Enfermedad endocrinológica (desplegable)
  - DM tipo 1
  - Hipotiroidismo autoinmune
  - Otros (campo libre)
- Enfermedad cardiovascular
  - HTA
  - Dislipemia
  - DM 2
  - Insuficiencia cardíaca
  - Enfermedad vascular cerebral
  - Arteriopatía periférica
  - Insuficiencia renal crónica
  - Enfermedad tromboembólica
  - Otros (campo libre)
- Enfermedad autoinmune/autoinflamatoria
  - Lupus
  - Artritis reumatoide
  - Síndrome Sjogren
  - Esclerosis sistémica
  - Espondiloartropatía seronegativa
  - Vasculitis
  - Miositis
  - Otros (campo libre)
- Enfermedad neurológica
  - Lesiones desmielinizantes-like SNC
  - Meningitis linfocitaria
  - Paquimeningitis
  - Neuritis óptica
  - Mielitis transversa
  - Neuropatía periférica
- Enfermedad cutánea
  - Vitíligo
  - Psoriasis
  - Eccema
  - Verrugas
  - Alopecia
  - Otros (campo libre)
- Alergias (campo libre)

**B) Añadir nuevo episodio** (rellenado por el paciente)

- Visita a Medico Atención primaria (Sí/No)
- Visita a Urgencias Hospitalarias (Sí/No)
- Ingreso (Sí/No)
- Días de duración
- Fiebre: (Sí/No)
- Síntomas catarrales: Sí -> desplegable
  - Tos
  - Moco
  - Dolor de garganta
  - Dolor torácico
  - Sensación de ahogo
  - Otros (campo libre)
- Síntomas digestivos: Sí -> desplegable
  - Diarrea
  - Nauseas o vómitos
  - Estreñimiento
  - Dolor abdominal
  - Otros (campo libre)
- Síntomas urinarios: Sí -> desplegable
  - Escozor al orinar
  - Orina oscura
  - Orina maloliente
  - Otros (campo libre)
- Sintomatología cutánea (desplegable)
  - Picor
  - Dolor
  - Coloración rojiza
  - Otros (campo libre)
- Otros (campo libre)
- Tratamiento durante el episodio: Sí -> se abren más opciones anidadas:
  - Tratamiento antibiótico: Sí/No
  - Antibiótico (campo abierto)
  - N° días de antibiótico (campo abierto)
  - Otros (campo abierto)

**2.2. Pruebas de laboratorio** (entre corchetes los límites de error). La primera analítica se rellena por el equipo médico para que posteriormente, los pacientes rellenen los valores más importantes

- Gammaglobulinas valle (valor numérico) (mg/L) [límites de error 0 - 10000]
  - IgG
    - IgG1
    - IgG2
    - IgG3
    - IgG4
  - IgA
  - IgM
- Perfil metabólico (valor numérico)
  - HbA1c (%) [1-20%]
  - HDL (mg/dL) [0-200]
  - LDL (mg/dL) [0-500]
  - TG (mg/dL) [0-1000]
  - Colesterol total (mg/dL) [0-500]
- Hemograma (valor numérico)
  - Hemoglobina (g/dL) [2-30]
  - Linfocitos (células/uL) [0-100000]
  - Neutrófilos (células/uL) [0-100000]
  - Plaquetas (células/uL) [0-2000000]

### 2.3. Tratamiento con Inmunoglobulinas

A) **Inmunoglobulinas intravenosas.** Notificaciones de siguiente dosis en función de la última administración y la cadencia. Rellenado por el médico.

- Tratamiento con inmunoglobulinas intravenosas: sí/no
- Preparado (campo libre)
- Dosis (gramos)
- Cadencia (desplegable)
  - Cada 2 semanas
  - Cada 3 semanas
  - Cada 4 semanas
- Efectos secundarios (desplegable)
  - Reacción cutánea
  - Fiebre
  - Dolor articular
  - Cefalea
  - Mal estar general
  - Tensión arterial baja
  - Anafilaxia

B) **Inmunoglobulinas subcutáneas.** Notificaciones de siguiente dosis en función de la última administración y la cadencia. Rellenado por el paciente.

- Tratamiento con inmunoglobulinas subcutáneas: sí/no
- Fecha
- Hora
- Tiempo de infusión
- Velocidad de infusión
- Efectos secundarios (desplegable):
  - Reacción cutánea
  - Fiebre
  - Dolor articular
  - Dolor de cabeza
  - Tensión arterial baja

**2.4. Otros tratamientos** (con posibilidad de eliminar cada uno o editarlo). Rellenado por el paciente

- Nombre del tratamiento (campo abierto)
- Dosis (campo abierto)
- Cadencia (campo abierto)
- Fecha inicio
- Fecha fin

**2.5. Vacunación.** Rellenado por el paciente y médico

- SARS-CoV-2
  - Sí/No
  - Dosis (desplegable: 1, 2, 3)
  - Fecha (dosis 1, dosis 2, dosis 3)
  - QR de vacunación
- Neumococo 13 valente
  - Sí/No
  - Dosis n – año
- Neumococo 23 valente
  - Sí/No
  - Dosis n – año
- Gripe
  - Sí/No
  - Dosis n – año
- Triple vírica
  - Sí/No

- Dosis n – año
- Difteria-tétanos-tosferina
  - Sí
  - Dosis n – año
- Varicela
  - Sí/No
  - Dosis n – año
- *Haemophilus influenzae*
  - Sí/No
  - Dosis n – año
- Meningococo ACWY
  - Sí/No
  - Dosis n – año
- Meningococo B
  - Sí/No
  - Dosis n – año
- Virus hepatitis A
  - Sí/No
  - Dosis n – año
- Virus hepatitis B
  - Sí/No
  - Dosis n – año
- Virus papiloma humano
  - Sí/No
  - Dosis n – año
- *Salmonella typhi*
  - Sí/No
  - Dosis n - año

**2.6. Calidad de vida:** cuestionario calidad de vida (EQ-5D-5L).  
Configurado para realizar de manera obligatoria con bloqueo de la aplicación, cada 6 meses y en el primer acceso.

### Movilidad

- No tengo problemas para caminar
- Tengo algunos problemas para caminar
- Tengo que estar en la cama

### Cuidado-Personal

- No tengo problemas con el cuidado personal
- Tengo algunos problemas para lavarme o vestirme solo
- Soy incapaz de lavarme o vestirme solo

### Actividades de Todos los Días (ej, trabajar, estudiar, hacer tareas domésticas, actividades familiares o realizadas durante el tiempo libre)

- No tengo problemas para realizar mis actividades de todos los días
- Tengo algunos problemas para realizar mis actividades de todos los días
- Soy incapaz de realizar mis actividades de todos los días

### Dolor/Malestar

- No tengo dolor ni malestar
- Tengo moderado dolor o malestar
- Tengo mucho dolor o malestar

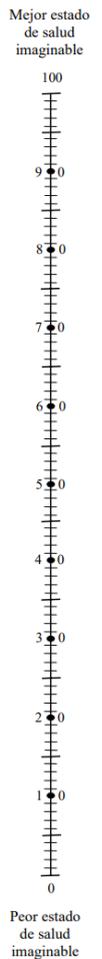
### Ansiedad/Depresión

- No estoy ansioso/a ni deprimido/a
- Estoy moderadamente ansioso/a o deprimido/a
- Estoy muy ansioso/a o deprimido/a

Para ayudar a la gente a describir lo bueno o malo que es su estado de salud, hemos dibujado una escala parecida a un termómetro en el cual se marca con un 100 el mejor estado de salud que pueda imaginarse, y con un 0 el peor estado de salud que pueda imaginarse.

Por favor, dibuje una línea desde el cuadro que dice "su estado de salud hoy," hasta el punto en la escala que, en su opinión, indique lo bueno o malo que es su estado de salud en el día de hoy.

Su estado de salud hoy



### 3. Próximas visitas. Actualización en Google Calendar y recordatorios 5-7 días antes. Rellenado por el paciente

- Próximas citas en consultas externas
- Próximas citas en extracciones
- Próximas citas con enfermera gestora
- Otras citas

### 4. Bibliografía y documentación. Rellenado por el médico

- Documentación con información sobre su enfermedad
- Vídeos explicativos sobre su enfermedad y la administración de tratamientos
- Documentación sobre medicamentos
- Documentación sobre estilos de vida sana
- Novedades sobre su enfermedad y tratamientos
- Buscador

## **5. Contacto sanitario**

- Correos con enfermera gestora (con respuesta automática predeterminada inicial)
- Llamadas programadas con enfermera gestora
- Videollamadas grupales/individuales programadas con enfermera gestora (destinadas a formación de los pacientes)

## **6. Redes de pacientes**

- Links a organizaciones internacionales
- Links a organizaciones nacionales
- Foro de pacientes
- Chat entre pacientes



### 8.3. Manual IDPBook



## MANUAL DE USUARIO IDPBOOK APP



App desarrollada en colaboración por:



## Contenido

Tabla de figuras.....	4
Introducción .....	5
Instalación de IDPBook.....	6
Requisitos del sistema.....	6
Descarga e instalación de IDPBook.....	6
Aplicaciones adicionales .....	7
Uso de la aplicación .....	8
Inicio de sesión .....	8
Descripción general de la interfaz de usuario .....	9
Cambio de contraseña .....	11
Personal médico.....	11
Pacientes .....	12
Registro de usuarios.....	13
Registro de pacientes .....	14
Registro de personal médico.....	15
Descargar información de paciente / Eliminar paciente .....	16
Gestión de registros de salud.....	17
Historias .....	18
Episodios .....	19
Pruebas de laboratorio .....	20
Gráficas evolutivas: Pruebas de laboratorio .....	21
Tratamiento con inmunoglobulinas .....	22
Crear nuevo tratamiento con inmunoglobulinas .....	22
Registrar administraciones del tratamiento .....	25
Finalizar tratamiento con inmunoglobulinas.....	26
Otros tratamientos .....	28
Vacunas .....	29
Cuestionarios calidad de vida .....	30
Gestión de citas y recordatorios.....	31
Abrir calendario .....	31
Agendar citas.....	32

Bibliografía y documentación.....	35
Documentación .....	35
Canal de YouTube.....	36
Contacto sanitario.....	37
E-mail Unidad de Inmunodeficiencias Primarias.....	37
Llamada Unidad de Inmunodeficiencias Primarias.....	39
Redes de pacientes .....	41
Soporte y contacto .....	42



## Tabla de figuras

Figura 1. Interfaz de usuario para inicio de sesión.....	8
Figura 2. Controles en ventana principal de la aplicación. ....	9
Figura 3. Controles en ventanas secundarias de la aplicación .....	10
Figura 4. Controles en ventanas de tipo formulario de la aplicación.....	10
Figura 5. Solicitud de cambio de contraseña para personal médico. ....	11
Figura 6. Aviso de usuario no encontrado en la base de datos. ....	11
Figura 7. Formulario para cambio de contraseña.....	12
Figura 8. Interfaz principal en modo personal médico .....	13
Figura 9. Interfaz de pacientes registrados .....	13
Figura 10. Apartado de registro de pacientes .....	14
Figura 11. Mensaje de confirmación registro nuevo personal médico.....	15
Figura 12. Gestión de pacientes.....	16
Figura 13. Ventana correspondiente a "Estado de la enfermedad" .....	17
Figura 14. Registro de historias .....	18
Figura 15. Registro de episodios.....	19
Figura 16. Registro de pruebas de laboratorio.....	20
Figura 17. Gráficas evolutivas analíticas.....	21
Figura 18. Registro de tratamientos con inmunoglobulinas.....	22
Figura 19. Recordatorios confirmados del tratamiento .....	23
Figura 20. Detalles del tratamiento en calendario .....	24
Figura 21. Registro de administraciones.....	25
Figura 22. Finalizar tratamiento con inmunoglobulinas .....	27
Figura 23. Registro de otros tratamientos.....	28
Figura 24. Registro de vacunas .....	29
Figura 25. Registro de cuestionarios de calidad de vida .....	30
Figura 26. Abrir calendario de Google .....	31
Figura 27. Abrir agenda de citas IDPBook.....	32
Figura 28. Procedimiento de reserva.....	33
Figura 29. Visualización de citas agendadas .....	34
Figura 30. Cancelación de citas.....	34
Figura 31. Acceso a documentación en Google Drive .....	35
Figura 32. Contenido audiovisual Hospital La Fe.....	36
Figura 33. Selección de aplicación para envío de correos .....	37
Figura 34. Plantilla para envío de correo electrónico .....	38
Figura 35. Acceso a llamadas con el personal médico .....	39
Figura 36. Procedimiento para darse de baja de llamada en Meet.....	40
Figura 37. Acceso al chat de pacientes IDPBook.....	41

## Introducción

¡Bienvenidos a IDPBook! Este manual de usuario ha sido creado para guiarte a través de la experiencia de usar nuestra aplicación de diario de salud electrónico, diseñada específicamente para pacientes con inmunodeficiencias primarias. Con IDPBook, te ofrecemos las herramientas necesarias para que puedas autogestionar y tomar el control de tu salud de manera más efectiva.

Sabemos que vivir con una inmunodeficiencia primaria puede presentar muchos desafíos, y nuestra aplicación tiene como objetivo ayudarte a enfrentar estos desafíos de la manera más sencilla y organizada posible. Con IDPBook, tendrás la capacidad de registrar y seguir de cerca tus síntomas, tratamientos, citas médicas, pruebas de laboratorio y mucho más. Además, nuestra aplicación te permite registrar episodios detallados para compartir con tus profesionales de la salud, lo que facilita una atención más personalizada y efectiva.

Nos enorgullece ofrecerte una plataforma segura y fácil de usar que te empodera en tu camino hacia una vida más saludable. Te invitamos a explorar todas las funciones que IDPBook tiene para ofrecerte y a descubrir cómo puede ayudarte a mejorar tu calidad de vida.

Si en algún momento necesitas asistencia adicional, no dudes en consultar este manual o ponerte en contacto con nuestro equipo de soporte.

¡Gracias por elegir IDPBook como tu compañero de salud!

## Instalación de IDPBook

En este apartado se informa sobre los requisitos para la instalación de la aplicación, además de la configuración inicial para empezar a utilizar la aplicación de manera correcta.

### Requisitos del sistema

La aplicación se encuentra disponible para sistemas operativos Android y iOS, tanto para móviles como para tabletas. En la siguiente tabla se indica el sistema operativo mínimo requerido para instalar IDPBook.

Sistema operativo	Versión mínima	Objetivo / Ideal
Android	5.0 (API 21)	14.0 (API 34)
iOS	11.0	-

En el caso de Android, el nivel API mínimo requerido por Play Store es 34, por lo tanto, la aplicación se ha desarrollado enfocándose en este nivel, así entonces, se recomienda que los dispositivos Android cuenten con una versión 14.0 o superior.

Por otro lado, para iOS, no existe una versión ideal, únicamente se limita para dispositivos con iOS 11.0 o superior.

### Descarga e instalación de IDPBook

En función del sistema operativo del dispositivo, la aplicación se encuentra en la tienda de aplicaciones correspondiente. A continuación, se encuentran los enlaces de descarga en Google Play (Play Store) y App Store.



### Aplicaciones adicionales

IDPBook utiliza aplicaciones externas necesarias para que la experiencia de usuario sea más agradable. A continuación, se listan las aplicaciones necesarias.

- Aplicación de correo electrónico: bien sea Gmail, Outlook, iCloud, Yahoo, entre otros, es necesario instalar una aplicación de este tipo, porque IDPBook utiliza este servicio para comunicaciones con el personal médico y agendar las citas que se solicitan en la aplicación. Se recomienda el uso de Gmail, por temas de sincronización con las citas gestionadas desde Google Calendar.
- Google Calendar: muy importante para la gestión de las citas solicitadas desde IDPBook. Al igual que con la aplicación de correo electrónico.
- Google Meet: necesaria para comunicarse con la Unidad de Inmunodeficiencias Primarias del hospital, al igual que con otros usuarios de la aplicación.
- Google Drive (opcional): necesaria para visualizar la documentación que el personal médico haya preparado para la consulta de los pacientes. Se marca como opcional porque también se puede acceder por algún navegador web.

**Importante:** en estas aplicaciones iniciar sesión con el mismo correo que está registrado en IDPBook.

## Uso de la aplicación

### Inicio de sesión

1

2

1 Inicio de sesión: se debe ingresar con el correo y contraseña que le ha proporcionado el personal médico. La contraseña por defecto es: 12345678, es decir, se utiliza la primera vez que se ingresa a la aplicación.

Se recomienda que después de ingresar la primera vez se realice la actualización de la contraseña. Para cambiar o actualizar la contraseña se debe pulsar en:

2 ¿Olvidaste tu contraseña?

Figura 1. Interfaz de usuario para inicio de sesión.

## Descripción general de la interfaz de usuario

La aplicación ha sido desarrollada para que el usuario tenga una experiencia agradable e intuitiva, sin complejas instrucciones o gran cantidad de controles. En las siguientes capturas se realiza la descripción de los elementos más importantes de la interfaz de usuario.



The screenshot shows the main interface of the IDPBook application. At the top left, a green box labeled '1' contains the text 'Paciente: Juan'. At the top right, a red power button icon labeled '2' is visible. The main content area features the 'IDPBook' logo and a grid of six navigation icons labeled '3': 'Información personal', 'Estado de la enfermedad', 'Citas', 'Documentación', 'Contacto sanitario', and 'Redes de pacientes'. At the bottom, there are logos for 'LaFe Hospital Universitari i Politècnic' and 'UNIVERSITAT Miguel Hernández'. On the right side, a green box labeled '1' shows the user profile 'Modo Administrador' and 'Paciente: Juan'. Below it, a red power button icon labeled '2' is shown, and a purple box labeled '3' highlights the navigation icons.

1 Paciente: Juan

2 

1 Nombre de usuario que ha iniciado sesión. Cuando el usuario es de tipo personal médico, el aspecto cambia tal como se muestra a continuación.

3  Modo Administrador  
Paciente: Juan

2 Botón para cerrar sesión.

3 Botones de navegación hacia los distintos apartados de la aplicación.

Figura 2. Controles en ventana principal de la aplicación.



Figura 3. Controles en ventanas secundarias de la aplicación

- 1 Botón para ir hacia atrás.
- 2 Botón para ir hacia los distintos formularios de la aplicación.
- 3 Lista de los formularios registrados, se puede seleccionar cualquier elemento para ver los detalles del formulario.



Figura 4. Controles en ventanas de tipo formulario de la aplicación.

- 1 Formulario: la información en cada formulario varía en función de la sección en que se encuentre el usuario.
- 2 Botones para registrar la información consignada en el formulario.

## Cambio de contraseña

El proceso para cambio de contraseña es distinto dependiendo del tipo de usuario, es decir paciente o personal médico.

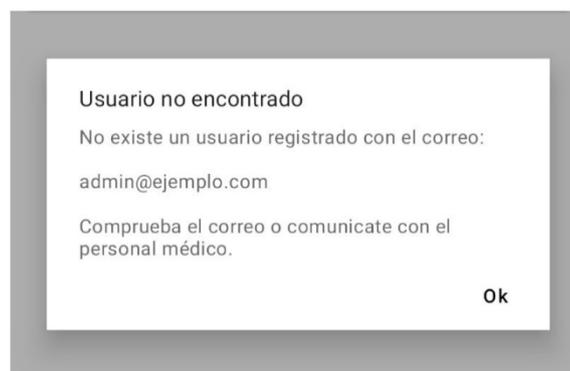
### Personal médico



The screenshot shows a web form titled "RESTABLECER CONTRASEÑA" with a back arrow icon. It contains a checkbox labeled "Per. médico:" which is checked. Below it is a text input field for "Correo:" containing the email "admin@ejemplo.com". At the bottom, there are two buttons: "Confirmar" (blue) and "Salir" (orange).

Figura 5. Solicitud de cambio de contraseña para personal médico.

Cuando se trata de personal médico se debe marcar la casilla "Per. médico" y suministrar un correo válido para este tipo de usuario. La aplicación valida con la base de datos si el correo existe y corresponde con un usuario autorizado como personal médico. Si los datos son correctos enviara un correo con las instrucciones para cambiar la contraseña, de lo contrario, mostrara un mensaje como el de la figura 6.



The screenshot shows a white dialog box with a grey border. The text inside reads: "Usuario no encontrado", "No existe un usuario registrado con el correo:", "admin@ejemplo.com", and "Comprueba el correo o comunícate con el personal médico." There is an "Ok" button in the bottom right corner.

Figura 6. Aviso de usuario no encontrado en la base de datos.

## Pacientes

Los pacientes deben actualizar la contraseña directamente en la aplicación como se muestra en la siguiente captura.



Figura 7. Formulario para cambio de contraseña.

Durante el proceso de cambio de contraseña, la aplicación puede mostrar algunos avisos en función del error que detecte. Los avisos más comunes son los siguientes:

Aviso	Causa
Campos vacíos	Alguno campo no ha sido completado.
Usuario incorrecto	El correo ingresado esta registrado como personal médico.
Contraseña no valida	La contraseña no contiene mínimo 6 caracteres
Verificar contraseña	La contraseña y la confirmación de ésta no coinciden.

Cualquier otro aviso que muestre la aplicación vendrá acompañado de un texto que explica al usuario la causa de este y como debería solucionarlo.

### Registro de usuarios

El registro de usuarios es una función exclusiva de los usuarios de tipo personal médico, por lo tanto, para los usuarios registrados como pacientes no será visible este apartado.

Para acceder a la sección de registro, el usuario médico debe pulsar el icono verde de la esquina superior izquierda en la ventana principal de la aplicación.

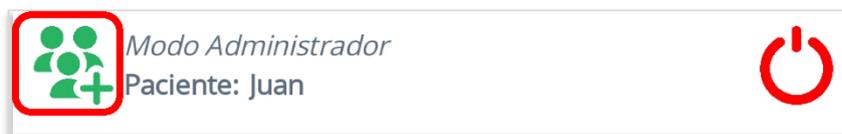


Figura 8. Interfaz principal en modo personal médico

En esta sección se muestran dos elementos principales, el icono para agregar usuarios y la lista de pacientes.



Figura 9. Interfaz de pacientes registrados

### Registro de pacientes

Para el registro de pacientes se debe diligenciar toda la información solicitada en el formulario. El único campo que no se diligencia es la edad, que se actualiza automáticamente con la fecha de nacimiento. En la siguiente captura se muestra un ejemplo del aspecto del formulario de registro.

REGISTRO USUARIOS

Tipo: Paciente  Personal médico

Nombre: Paciente

Apellidos: Apellido

Correo: ejemplo@gmail.com

Sexo: Femenino

Fecha Nac. 18/10/1992

Edad: 31

Diagnóstico: ALPS

Fecha Diag. 15/10/2024

Otro Diag. 1: Ingresa otro diagnostico 1

Fecha Diag. 1: 15/10/2024

Otro Diag. 2: Ingresa otro diagnostico 2

Fecha Diag. 2: 15/10/2024

Agregar Salir

Figura 10. Apartado de registro de pacientes

Como se menciona anteriormente en el apartado de [Inicio de sesión](#), la contraseña por defecto para los pacientes es 12345678, que el usuario puede cambiar una vez haya iniciado sesión por primera vez.

### Registro de personal médico

Para registrar un usuario de tipo personal médico solo es necesario el nombre y el correo electrónico del nuevo usuario. Si el correo ya existe en la base de datos, la aplicación mostrara un aviso para que se cambie este campo.

La contraseña por defecto para personal médico es 0987654 y al igual que los pacientes, se puede cambiar la contraseña una vez haya iniciado sesión por primera vez.



Figura 11. Mensaje de confirmación registro nuevo personal médico

Es muy importante autenticar el correo electrónico pues la aplicación utiliza esta propiedad para otorgar los permisos necesarios al nuevo usuario médico y que este pueda modificar información que los pacientes no pueden, así como la creación de nuevos pacientes.

### Descargar información de paciente / Eliminar paciente

En el apartado de "Pacientes" se puede gestionar la información de los pacientes o eliminarlos, en la siguiente imagen se puede ver un ejemplo de cada caso.

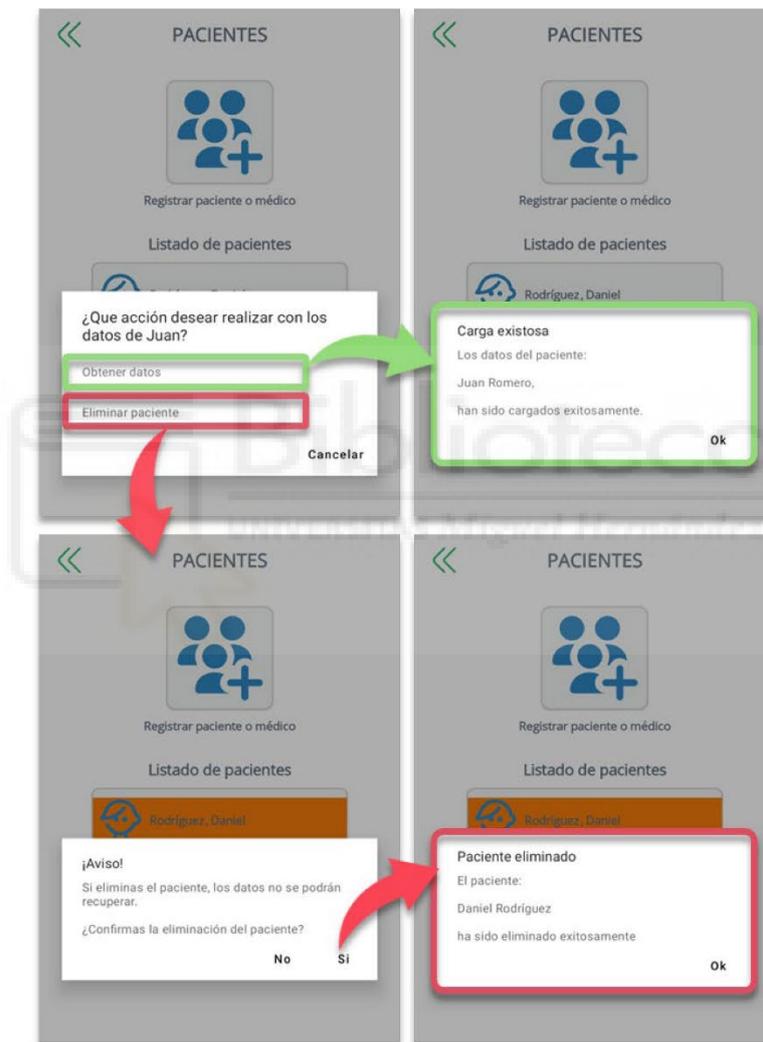


Figura 12. Gestión de pacientes

Al cerrar la ventana pulsando ok o fuera de la ventana auxiliar, la aplicación se dirige automáticamente a la ventana principal.

## Gestión de registros de salud



Figura 13. Ventana correspondiente a "Estado de la enfermedad"

Cada uno de los elementos de este apartado, dirige al usuario hacia una nueva ventana correspondiente a la descripción del icono.

Es importante comentar que la mayoría de los formularios no solicitan información obligatoria, es decir es posible enviar formularios con campos vacíos. En la visualización del formulario registrado, estos campos que no se diligenciaron se mostrara como N/A.

## Historias

El registro de historias clínicas es de acceso exclusivo para el modo administrador. El paciente podrá ver sus historias, pero no podrá modificar o eliminar las historias.

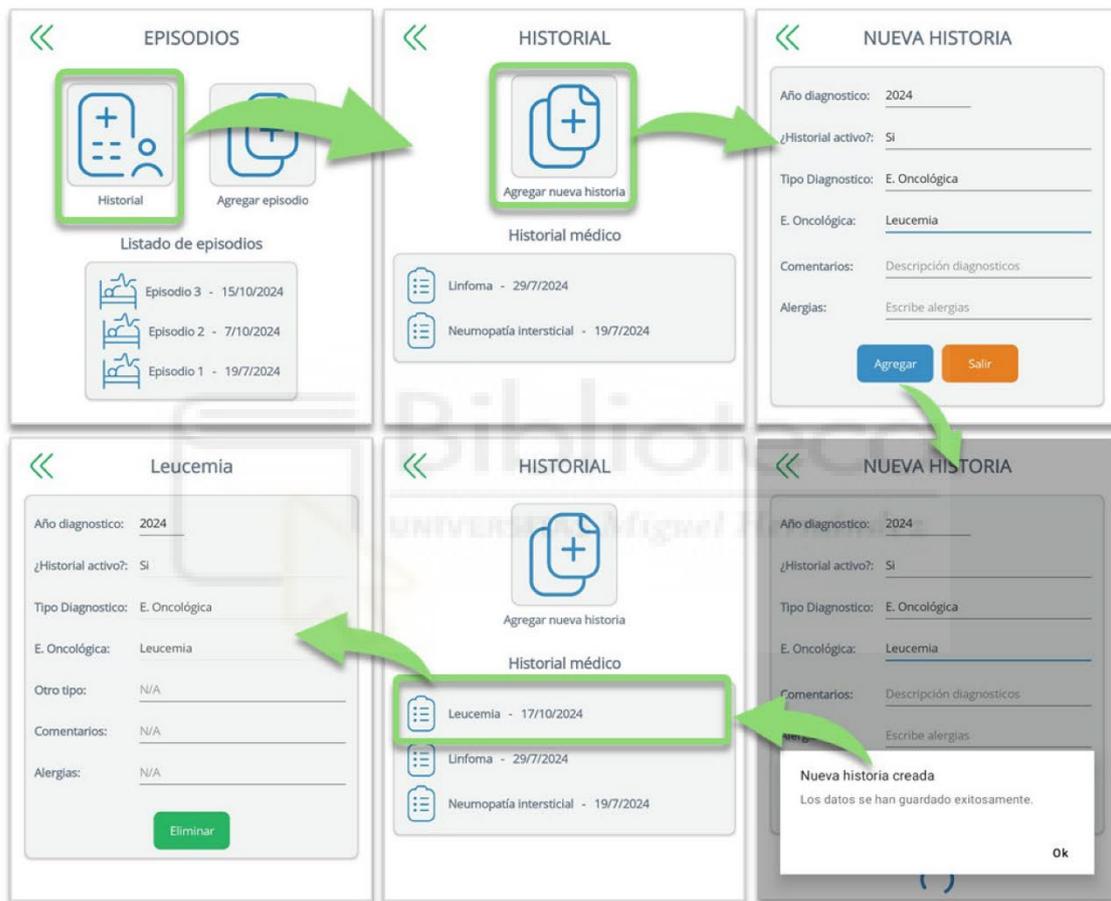


Figura 14. Registro de historias

La historia y cualquier otro registro se puede eliminar ingresando a la misma desde la lista de registros, como se puede ver en la captura de la esquina inferior izquierda.

## Episodios

The figure displays four screenshots of a mobile application interface for recording episodes, arranged in a 2x2 grid. The top-left screenshot shows the 'EPISODIOS' list with the 'Agregar episodio' button highlighted. The top-right screenshot shows the 'EPISODIOS' list with 'Episodio 4 - 17/10/2024' highlighted. The bottom-left screenshot shows the 'NUEVO EPISODIO' form with the 'Agregar' button highlighted. The bottom-right screenshot shows a confirmation message: 'Nuevo Episodio registrado. Los datos se han guardado exitosamente.' with an 'Ok' button.

**EPISODIOS**

Historial    Agregar episodio

Listado de episodios

- Episodio 3 - 15/10/2024
- Episodio 2 - 7/10/2024
- Episodio 1 - 19/7/2024

**NUEVO EPISODIO**

Fecha de episodio: 17/10/2024

¿Visita a atención primaria? Si

¿Visita a urgencias hospitalarias? No

Días de duración: 1

¿Ingreso? No

¿Fiebre? No

Sintomas catarrales: ▼ Seleccionar

- Tos
- Moco
- Dolor de garganta
- Dolor torácico
- Sensación de ahogo

Otro sintoma: N/A

Sintomas digestivos: ▼ Seleccionar

Sintomas urinarios: ▼ Seleccionar

Sintomas cutaneos: ▼ Seleccionar

Otros síntomas: N/A

¿Tratamiento durante episodio? ▼ Seleccionar Si/No

Agregar    Salir

**NUEVO EPISODIO**

Fecha de episodio: 17/10/2024

¿Visita a atención primaria? Si

¿Visita a urgencias hospitalarias? No

Días de duración: 1

¿Ingreso? No

¿Fiebre? No

Sintomas catarrales: ▼ Seleccionar

**Nuevo Episodio registrado**

Los datos se han guardado exitosamente.

Ok

Sintomas digestivos: ▼ Seleccionar

Sintomas urinarios: ▼ Seleccionar

Sintomas cutaneos: ▼ Seleccionar

Otros síntomas: N/A

¿Tratamiento durante episodio? ▼ Seleccionar Si/No

Agregar    Salir

Figura 15. Registro de episodios

Pruebas de laboratorio

**PRUEBAS DE LABORATORIO**

+ Añadir nueva prueba de laboratorio

Listado de pruebas

- Analítica 1 - 17/10/2024
- Analítica 2 - 20/10/2024

Mostrar gráficas evolutivas

**NUEVA PRUEBA**

Fecha de analítica: 17/10/2024

Gammaglobulinas valle (mg/dL)

IgG	3500
IgG1	2000
IgG2	1680
IgG3	4658
IgG4	2568
IgA	3669
IgM	3699

Perfil metabólico

HbA1c (%)	15
HDL (mg/dL)	108
LDL (mg/dL)	369
TG (mg/dL)	869
Colesterol total (mg/dL)	458

Hemograma

Hemoglobina (g/dL)	18
Linfocitos (células/uL)	86900
Neutrófilos (células/uL)	76580
Plaquetas (células/uL)	1500000

Registrar Cancelar

Figura 16. Registro de pruebas de laboratorio

En esta sección es posible observar “Gráficas evolutivas” de las analíticas registradas. Como se puede observar en la figura 16, las gráficas están disponibles cuando hay dos o más registros en la lista. A continuación, se muestran los tipos de gráficos disponibles.

Gráficas evolutivas: Pruebas de laboratorio



Figura 17. Gráficas evolutivas analíticas

## Tratamiento con inmunoglobulinas

### Crear nuevo tratamiento con inmunoglobulinas

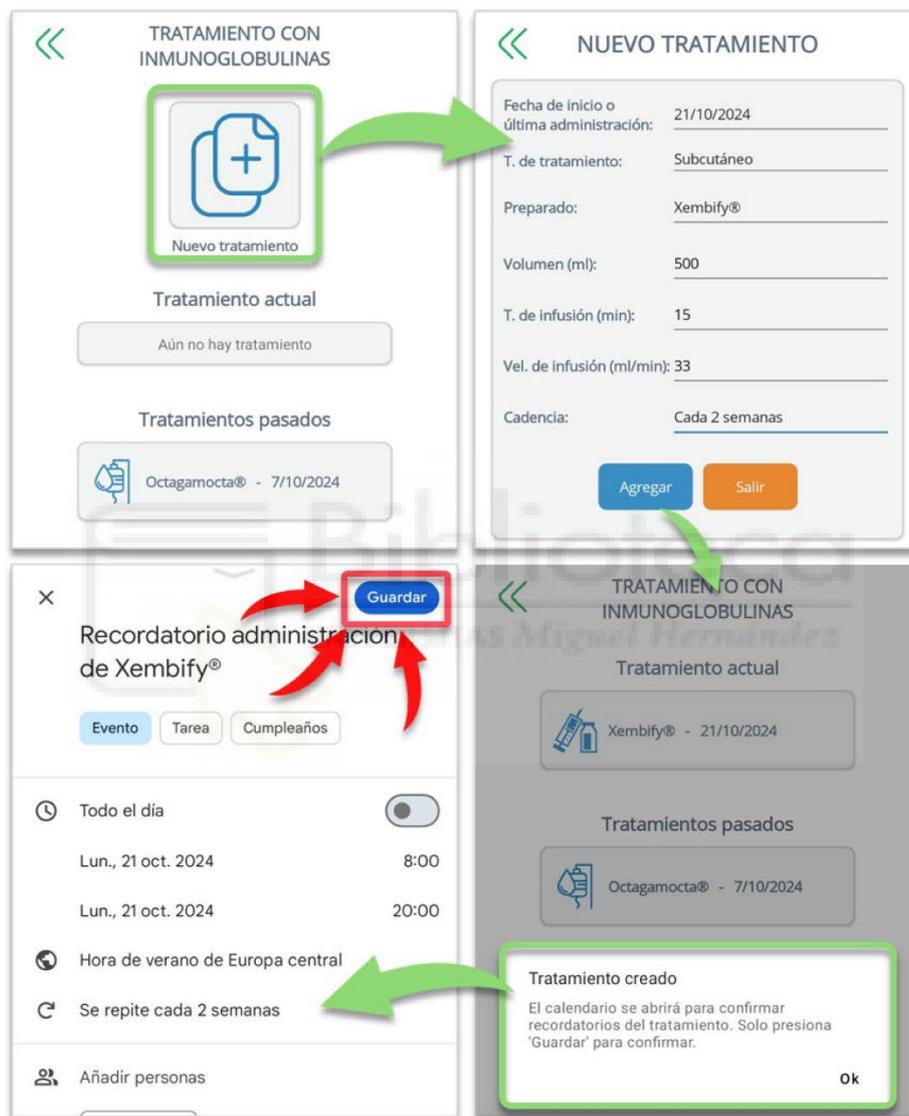


Figura 18. Registro de tratamientos con inmunoglobulinas

**Importante:** presionar el botón "Guardar" para que los recordatorios del tratamiento se guarden en el calendario según la cadencia de este.

Una vez se guarda el evento, los recordatorios quedan configurados en función de la cadencia del tratamiento y el paciente recibirá una notificación del calendario recordándole la administración de la siguiente dosis del tratamiento.

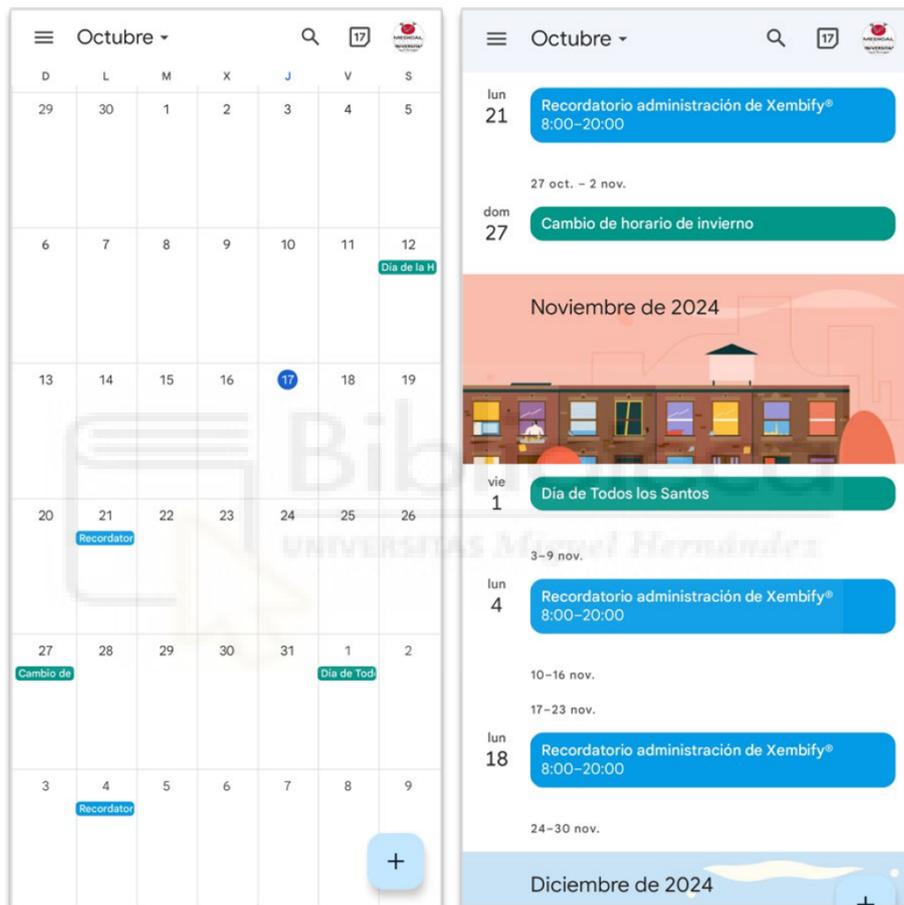


Figura 19. Recordatorios confirmados del tratamiento

En la figura anterior se presentan los recordatorios del nuevo tratamiento registrado, los cuales pueden visualizarse en diferentes formatos, como el formato mensual (izquierda) o en vista de agenda (derecha). Al seleccionar un evento, se despliega la información detallada del tratamiento, tal como se muestra en la siguiente figura.



Figura 20. Detalles del tratamiento en calendario

Importante: los recordatorios del tratamiento se configuran conforme al calendario personal de cada paciente. Es importante señalar que IDPBook no tiene la capacidad de crear recordatorios específicos de manera automática, ya que estos dependen de las preferencias y ajustes establecidos por cada usuario en su propio calendario.

### Registrar administraciones del tratamiento

The diagram illustrates the process of registering a treatment administration through a mobile application. It is divided into four main panels:

- Panel 1: TRATAMIENTO CON INMUNOGLOBULINAS**
  - Tratamiento actual:** Xembify® - 21/10/2024 (highlighted with a green box).
  - Tratamientos pasados:** Octagamocta® - 7/10/2024.
- Panel 2: Xembify®**
  - T. de tratamiento: Subcutáneo.
  - Fecha de inicio: 21/10/2024.
  - Preparado: Xembify®.
  - Volumen (ml): 500.
  - T. de infusión (min): 15.
  - Vel. de infusión (ml/min): 33.
  - Cadencia: Cada 2 semanas.
  - Buttons: Finalizar tratamiento (green), Registrar administración (orange).
- Panel 3: NUEVA ADMINISTRACIÓN**
  - Fecha de administración: 21/10/2024.
  - Ef. secundarios:
    - Reacción cutánea.
    - Fiebre.
    - Dolor articular.
    - Cefalea.
    - Malestar general.
    - Tensión arterial baja.
    - Anafilaxia.
  - Buttons: Agregar (blue, highlighted with a green box), Salir (orange).
- Panel 4: Xembify®**
  - Administraciones:
    - Administración 1 - 21/10/2024 (highlighted with a green box).

Arrows indicate the flow: from the overview to the treatment details, then to the 'NUEVA ADMINISTRACIÓN' form, and finally to the updated list of administrations.

Figura 21. Registro de administraciones

### Finalizar tratamiento con inmunoglobulinas

The process is shown in four steps:

- Finalizar tratamiento:** In the 'Xembify®' treatment details screen, the 'Finalizar tratamiento' button is highlighted with a green box.
- Confirmación:** A dialog box asks '¿Aviso! ¿Confirmas la finalización del tratamiento?'. The 'Si' button is circled in green.
- Recordatorio:** A calendar view for October and November 2024 shows a red box around a 'Recordatorio administración de Xembify®' event on November 18th.
- Finalización:** A 'Tratamiento finalizado' dialog box appears, with a green box around the text: 'El calendario se abrirá para eliminar los recordatorios. Selecciona el evento y elige "Todos los eventos" para eliminar todos los recordatorios.' An 'OK' button is also visible.

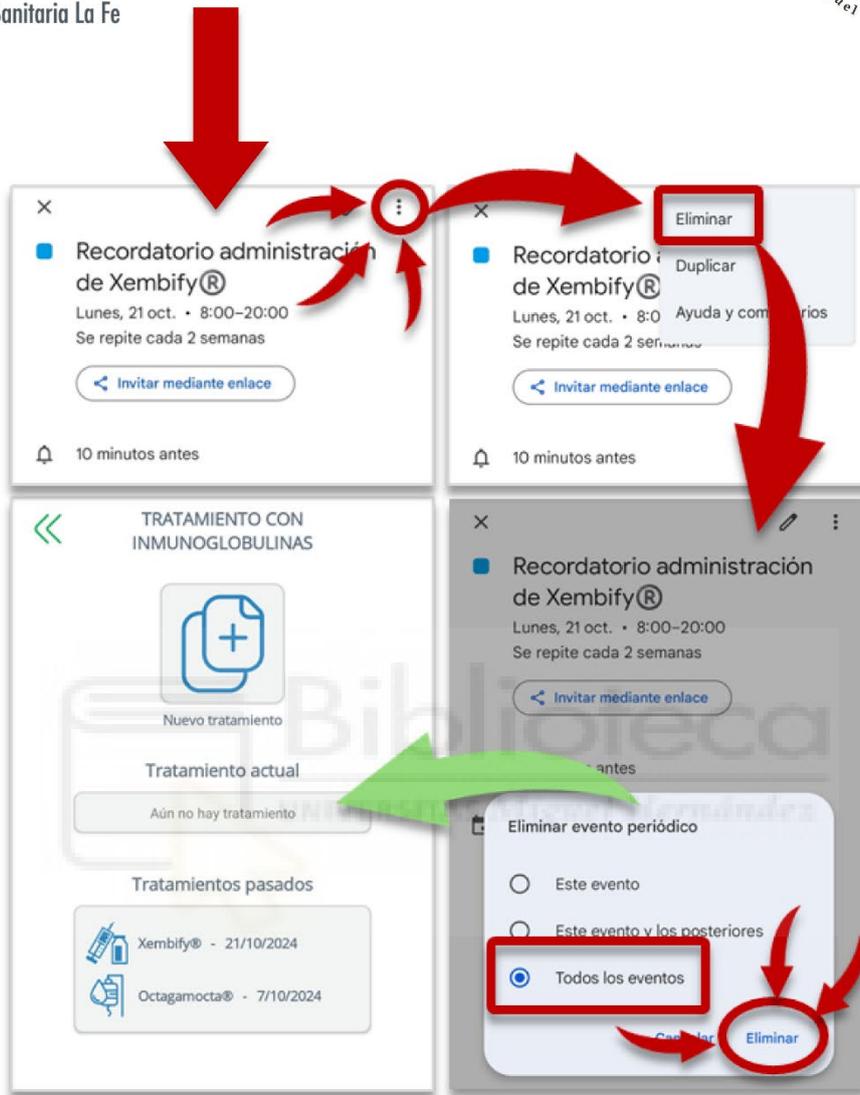


Figura 22. Finalizar tratamiento con inmunoglobulinas

Importante: en el calendario, seleccionar "Todos los eventos" al momento de eliminar los recordatorios del tratamiento.

### Otros tratamientos

OTROS TRATAMIENTO

Agregar nuevo tratamiento

Listado de otros tratamientos

- Tratamiento Y - 11/9/2024
- Tratamiento X - 11/9/2024

NUEVO TRATAMIENTO

Nombre: Nuevo

Número de dosis: 5

Cadencia: 3 días

Fecha de inicio: 21/10/2024

¿Tratamiento crónico? No

Fecha de fin: 12/11/2024

Agregar Salir

OTROS TRATAMIENTO

Agregar nuevo tratamiento

Listado de otros tratamientos

- Nuevo - 21/10/2024
- Tratamiento Y - 11/9/2024
- Tratamiento X - 11/9/2024

NUEVO TRATAMIENTO

Nombre: Nuevo

Número de dosis: 5

Cadencia: 3 días

Fecha de inicio: 21/10/2024

¿Tratamiento crónico? No

Fecha de fin: 12/11/2024

Nuevo Tratamiento registrado  
Los datos se han guardado exitosamente.

Ok

Figura 23. Registro de otros tratamientos

## Vacunas

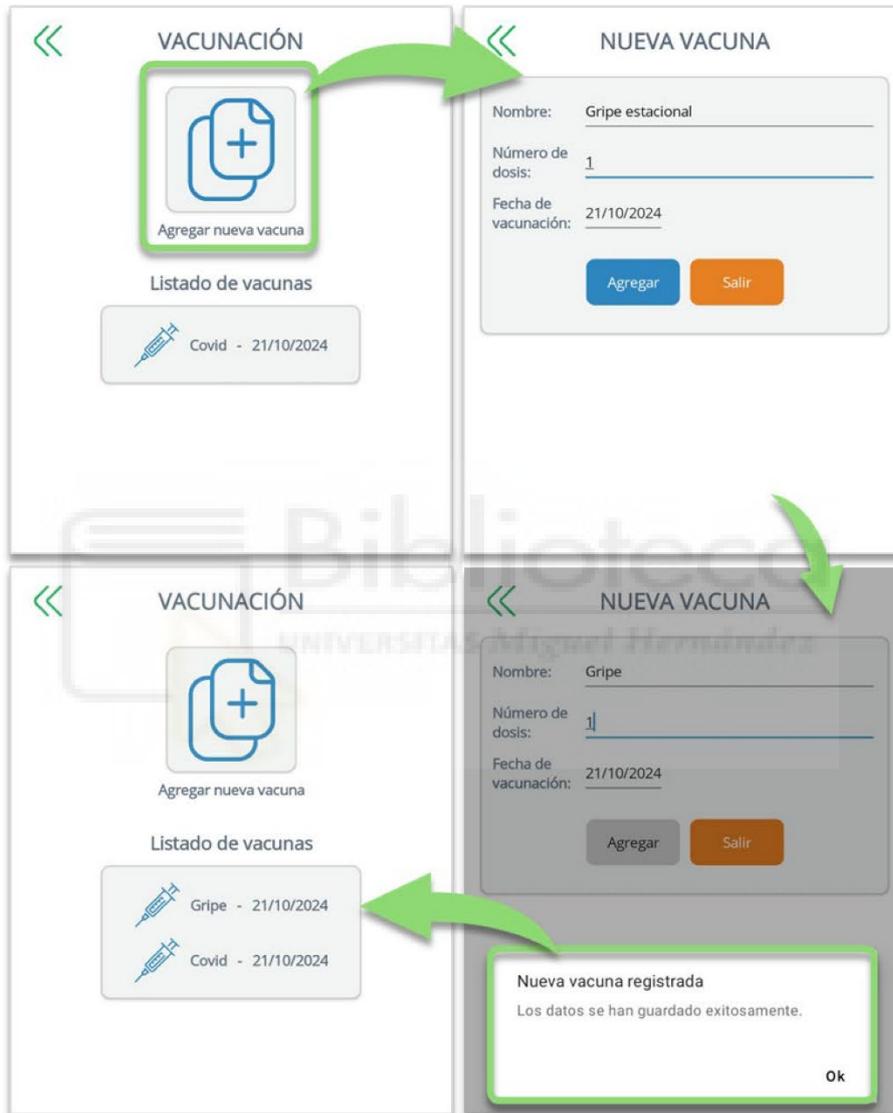


Figura 24. Registro de vacunas

### Cuestionarios calidad de vida

The image displays a mobile application interface for quality of life questionnaires. It is divided into several sections:

- CUESTIONARIO CALIDAD DE VIDA (Top Left):** Shows a list of questionnaires with scores and dates. A green box highlights the 'Agregar nuevo cuestionario' button.
- NUEVO CUESTIONARIO (Top Right):** A form to register a new questionnaire. It includes sections for 'Movilidad', 'Cuidado personal', 'Actividades diarias', 'Dolor/Malestar', and 'Ansiedad/Depresión'. A scale for 'Su estado de salud hoy' is shown with a value of 93.
- CUESTIONARIO CALIDAD DE VIDA (Middle Left):** Shows a list of questionnaires with scores and dates. A green box highlights the 'Mostrar gráficas evolutivas' button.
- Cuestionario registrado (Middle Right):** A confirmation message: 'Cuestionario registrado. Los datos se han guardado correctamente.' with an 'OK' button.
- Gráfico (Bottom):** A line graph titled 'Evolución estado de salud' showing the progression of health status over time. The x-axis shows dates: 10/8/2024, 30/8/2024, and 21/10/2024. The y-axis shows a score from 0 to 100. The graph shows a steady increase from approximately 65 to 93.

Figura 25. Registro de cuestionarios de calidad de vida

Al igual que en el apartado de "Pruebas de laboratorio", la gráfica de la evolución del estado de salud está disponible cuando hay dos o más registros en la lista.

## Gestión de citas y recordatorios

Desde la sección de citas es posible dirigirse hacia el calendario (Google Calendar) y revisar las citas programadas, además, agendar una cita o crear un evento en el calendario, correspondiente a una cita o cualquier diligencia que el usuario vea conveniente.

### Abrir calendario

En el apartado de citas, pulsar sobre el icono "Calendario", a continuación, la aplicación se dirige hacia el calendario de Google. La siguiente captura representa la descripción anterior.

En el calendario se podrán observar las citas que se hayan agendado con anterioridad, pudiendo gestionar los recordatorios o cancelar la cita si lo desea.

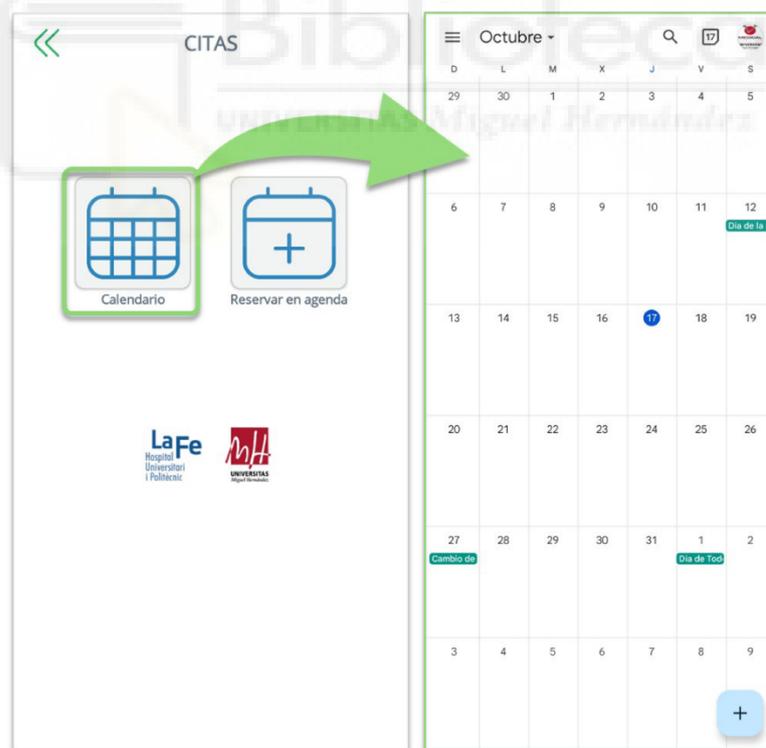


Figura 26. Abrir calendario de Google

## Agendar citas

Para agendar citas se debe pulsar en “Reservar en agenda”, a continuación, la aplicación se dirigirá al navegador web que el móvil tenga por defecto y abrirá una nueva página con la agenda de citas de IDPBook.

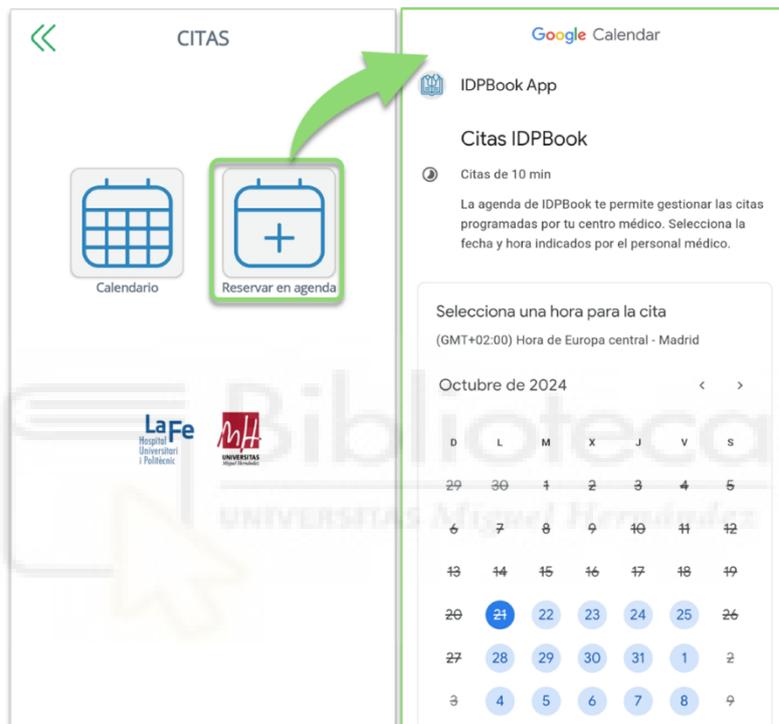


Figura 27. Abrir agenda de citas IDPBook

Una vez ahí, se procede a seleccionar el día y la hora acordada con el personal médico, en el caso que el paciente sea quien este solicitando la cita. En la figura 28, se muestra un ejemplo del procedimiento para realizar una reserva para el lunes 20 de mayo de 2024, a las 11:00.

Se recomienda el uso de Google Chrome e iniciar sesión con el mismo correo que se ha registrado en IDPBook, de esta manera, los campos del formulario de reserva se diligencian automáticamente con los datos del usuario.

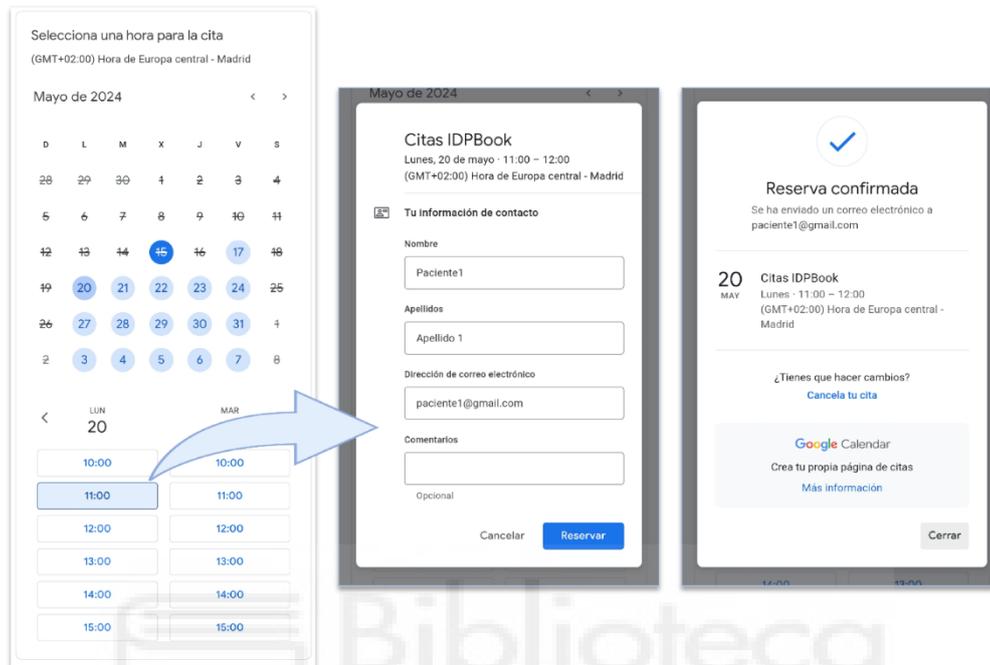


Figura 28. Procedimiento de reserva

Cuando la reserva este confirmada, revisar el calendario para comprobar que la cita se haya sincronizado correctamente. Si la cita no está en el calendario, ir al correo y sincronizar el evento con el calendario. Esta configuración solo será necesaria una vez, en adelante las citas se sincronizarán automáticamente.

En la figura 29, se muestra un ejemplo de como se muestra la cita en el día agendado y el periodo de tiempo que se ha reservado durante ese día.

Pulsando en la cita, se accede a la información de esta, pudiendo también modificar los recordatorios como el usuario vea convenientes y también cancelar la cita si así lo desea. Cuando una cita es cancelada, se enviará un correo al médico informando que el paciente ha cancelado la cita, y por supuesto, el evento desaparecer del calendario. En la figura 30, muestra que la cita agendada se ha eliminado después de enviar la cancelación.

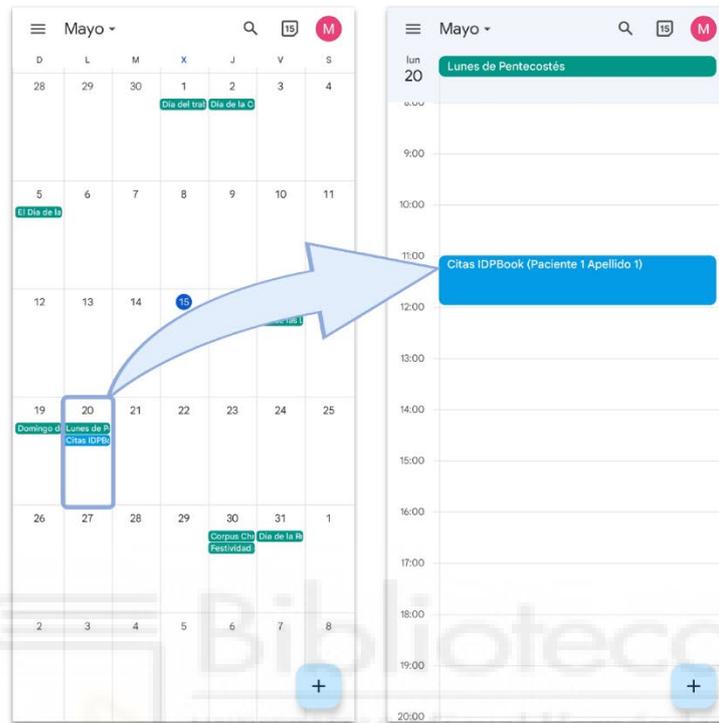


Figura 29. Visualización de citas agendadas

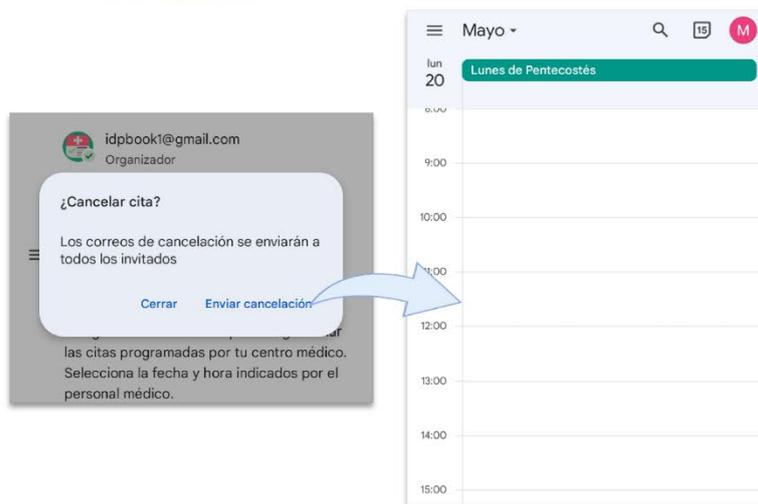


Figura 30. Cancelación de citas

## Bibliografía y documentación

Toda la documentación y recursos disponibles para el paciente se pueden acceder a través de Google Drive. En esta plataforma el paciente encontrará información actualizada sobre su enfermedad y sus tratamientos.

### Documentación

Para acceder a dicha información, se debe pulsar en el icono de "Documentación", y la aplicación procederá a abrir Google Drive. Se podrán encontrar varias carpetas o ficheros clasificados según el tipo de información consignada. Desde información sobre su enfermedad, información sobre medicamentos, material audiovisual sobre tratamientos, documentación sobre estilos de vida saludables, entre otros.

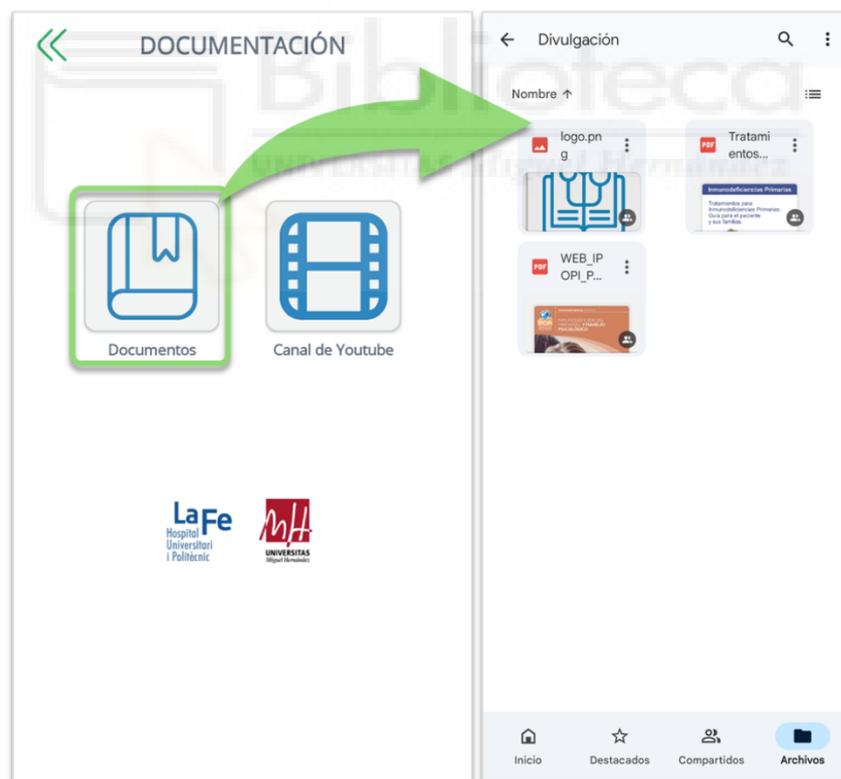


Figura 31. Acceso a documentación en Google Drive

### Canal de YouTube

Canal del hospital donde se puede encontrar gran cantidad de material audiovisual que el paciente necesite.



Figura 32. Contenido audiovisual Hospital La Fe

## Contacto sanitario

### E-mail Unidad de Inmunodeficiencias Primarias

Para enviar un mensaje al personal médico se debe pulsar el icono de "E-mail Unidad de Inmunodeficiencias Primarias". Si es la primera vez que se realiza esta actividad, se mostrará un mensaje preguntándole al usuario que aplicación de correo electrónico desea utilizar para enviar el correo. Nunca seleccionar IDPBook la aplicación no soporta esta acción y se cierra inmediatamente.



Figura 33. Selección de aplicación para envío de correos

Se recomienda seleccionar la opción "Siempre", para que la aplicación no pregunte lo mismo cada vez que se quiera enviar un correo.

Una vez en la aplicación de correo seleccionada, se abrirá una plantilla con destinatario y asunto predeterminado, como se muestra en la figura 34, el correo predeterminado es [idpbook1@gmail.com](mailto:idpbook1@gmail.com), y el asunto es "Contacto sanitario IDPBook App".

Se recomienda que el asunto no se modifique para que el personal médico pueda identificar de manera más sencilla los correos enviados por los usuarios de la aplicación.



Figura 34. Plantilla para envío de correo electrónico

## Llamada Unidad de Inmunodeficiencias Primarias

Para contactar por llamada a la Unidad se deber tener en cuenta que el horario es de 8:30 a 15:00, fuera de este horario saldrá un aviso que no permite realizar la llamada.



Figura 35. Acceso a llamadas con el personal médico

Una vez finalizada la llamada el usuario debe darse de baja del grupo para que los pacientes puedan realizar a una llamada privada, de lo contrario la próxima vez que otro paciente realice una llamada el paciente que no se haya dado de baja también recibirá dicha llamada, lo que puede resultar incomodo cuando estas llamadas se realizan con bastante frecuencia. El proceso para darse de baja se describe en la siguiente figura.

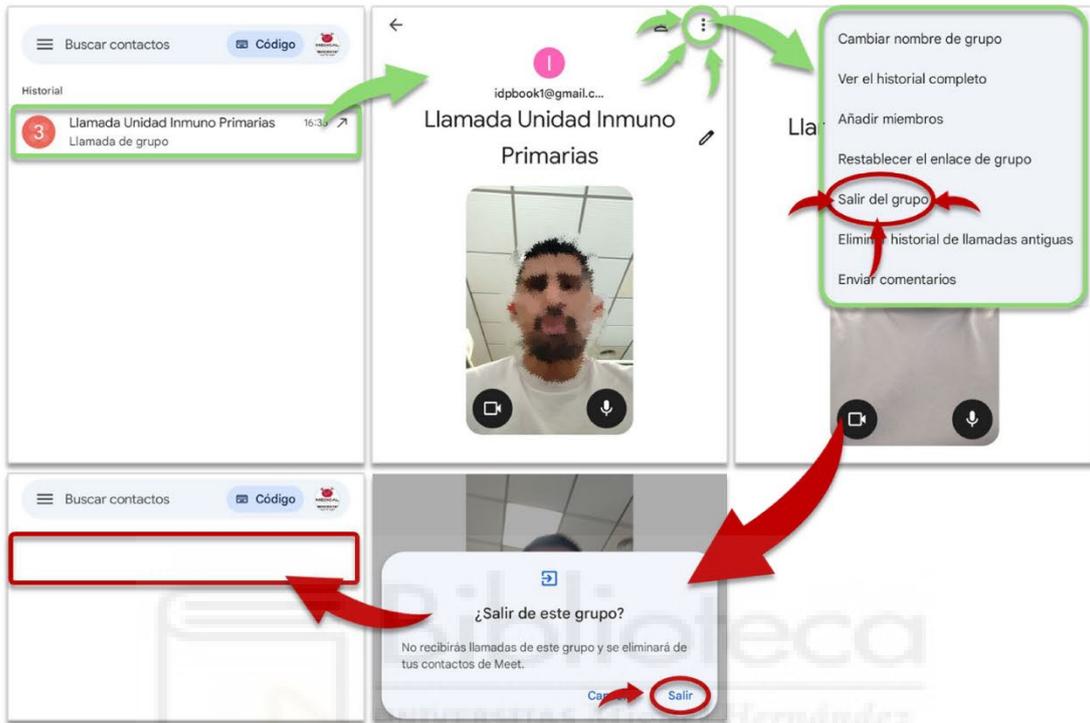


Figura 36. Procedimiento para darse de baja de llamada en Meet

## Redes de pacientes

En este apartado se encuentran varios enlaces para distintas páginas web de interés para los pacientes al igual que un enlace a un grupo de "Telegram" en el cual la Unidad de Inmunodeficiencias Primarias y todos los usuarios de la aplicación pueden compartir información y hablar sobre su experiencia con IDPBook.

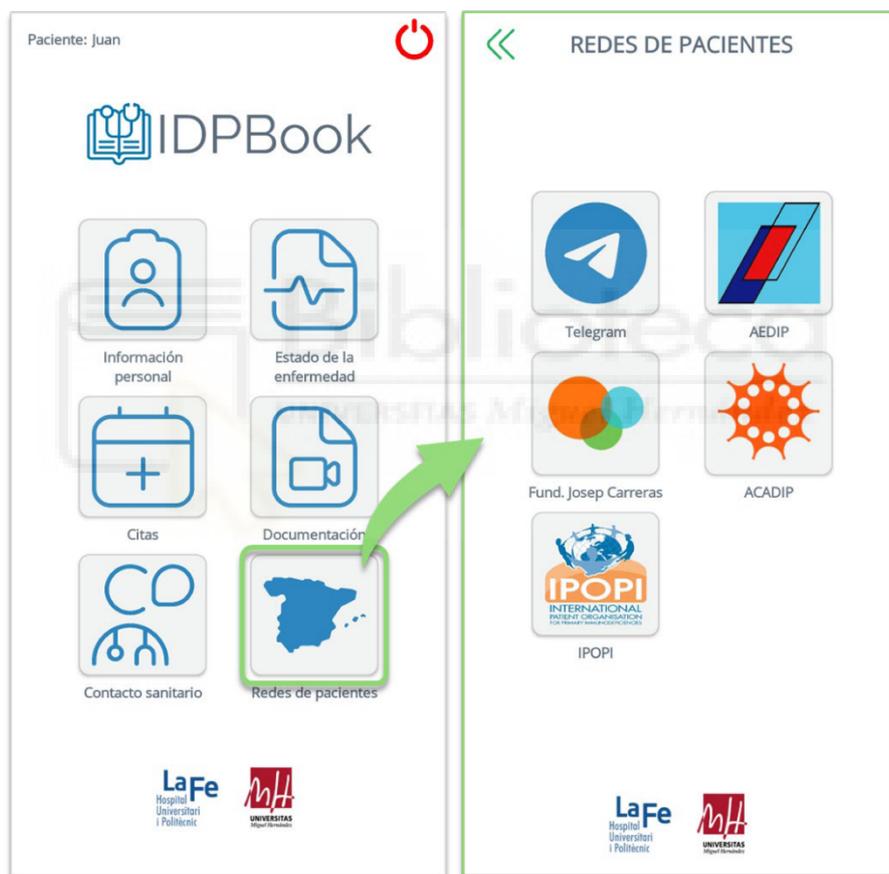


Figura 37. Acceso al chat de pacientes IDPBook

## Soporte y contacto

Si desea comunicarse con el equipo desarrollador, o informar sobre alguna incidencia con la aplicación, enviar un correo a las siguientes cuentas, [medicalroboticsumh@gmail.com](mailto:medicalroboticsumh@gmail.com)

[idpbook1@gmail.com](mailto:idpbook1@gmail.com)

O puede visitar la página web de nuestro grupo de investigación, Medical Robotics Unit, perteneciente al Instituto de Bioingeniería de la Universidad Miguel Hernández. Puedes dejar un mensaje desde el apartado de contacto.

Pulsa la siguiente imagen para visitar la web.



#### 8.4. Escala de calificación uMARS

Nombre de la aplicación: \_\_\_\_\_

Plataforma: Apple  Android  Otras: \_\_\_\_\_

Rodee con un círculo el número de respuesta que mejor defina la cualidad de la app. Todas las respuestas están clasificadas con una puntuación que va de “1- inadecuado a 5- excelente”. Si considera que alguna cualidad no puede ser evaluada seleccione la opción N/A (no aplicable).

##### **SECCIÓN A Participación- divertido, interesante, personalizable, interactivo (envía alertas, mensajes, recordatorios, permite compartir etc.)**

1. Entretenimiento: ¿Es la aplicación divertida/entretenida?, ¿Usa estrategias de entretenimiento para aumentar la participación (por ejemplo, a través de juegos)?

1. Aburrida, no es divertida o entretenida en absoluto.
2. Mayormente aburrida.
3. OK, diversión suficiente para entretener al usuario durante un breve tiempo (<5 minutos).
4. Moderadamente divertida y entretenida, entretendría al usuario durante algún tiempo (5-10 minutos en total).
5. Muy entretenida y divertida, estimularía el uso repetido.
6. N/A.

2. Interés: ¿Es interesante la aplicación?, ¿Presenta su contenido de manera interesante en comparación con otras apps?

1. En absoluto interesante.
2. Poco interesante.
3. Normal, suficiente para entretener por un tiempo breve ( <5 minutos).
4. Moderadamente interesante. Capta la atención del usuario un rato (5-10 minutos).
5. Muy interesante. Estimula su uso repetido.
6. N/A.

3. Personalización: ¿Permite realizar ajustes o seleccionar preferencias de las características de configuración (sonido, contenido, notificaciones, etc....)?

1. No permite ninguna personalización o requiere que la configuración se introduzca cada vez.
2. La personalización es insuficiente con funciones limitadas.
3. Permite una personalización básica con adecuada funcionalidad.
4. Dispone de opciones de personalización numerosas.
5. La personalización es avanzada, completamente adaptada a preferencias y características individuales.
6. N/A.

4. Interacción: ¿Permite entradas de usuario, provee retroalimentación, envía avisos?

1. No dispone de características interactivas y/o no responde a la interacción del usuario.
2. Interacción insuficiente, o retroalimentación, u opciones de entrada de usuario, con funciones limitadas.
3. Características de interacción básicas con adecuada funcionalidad.
4. Ofrece una variedad de características interactivas/retroalimentación/opciones de entrada de usuario.
5. Nivel muy alto de respuesta a características interactivas/retroalimentación/opciones de entrada de usuario.
6. N/A.

5. Público al que va dirigido: ¿El contenido de la aplicación (información visual, lenguaje, diseño) es apropiado para el tipo de usuario al que se dirige?

1. Completamente inapropiado/ confuso.
2. Mayormente inapropiado/ confuso.
3. Aceptable pero no Objetivo. Puede ser inapropiado/confuso.
4. Bien dirigido, con problemas de poca importancia.
5. Perfectamente dirigido, sin problemas aparentes.
6. N/A.



## **SECCIÓN B Funcionalidad- funcionamiento de la app, facilidad de aprendizaje, navegación, transcurso lógico y diseño gestual.**

6. Rendimiento: ¿Con qué velocidad/ precisión funcionan las características de la aplicación (funciones) y sus componentes (botones/menús)?

1. La aplicación está dañada o funciona mal. Respuesta insuficiente/imprecisa (por ejemplo, bloqueos del sistema/ fallos / características erróneas, etc.)
2. Permite algunas funciones, pero sufre problemas técnicos graves.
3. En general funciona, aunque a veces sufre problemas técnicos reparables o va lenta.
4. Principalmente funciona. Problemas menores/insignificantes.
5. Responde perfectamente a su tiempo. Sin problemas técnicos aparentes/ contiene un indicador de “tiempo restante de carga”.
6. N/A.

7. Facilidad de uso: ¿Es fácil aprender a usar la aplicación; las etiquetas/iconos del menú y las instrucciones son claras?

1. Sin instrucciones o con instrucciones limitadas; las etiquetas/ iconos del menú son confusas/ complicadas.
2. Requiere mucho tiempo/ esfuerzo aprender a utilizarla.
3. Requiere un poco de tiempo/ esfuerzo aprender a utilizarla.
4. Fácil de usar (o con instrucciones claras).
5. Permite uso inmediato; intuitiva; simple.
6. N/A.

8. Navegación: ¿Moverse entre pantallas es lógico/preciso/ apropiado/ininterrumpido; están presentes todos los enlaces de pantalla necesarios?

1. Varias secciones de la aplicación están desconectadas lógicamente y la navegación es confusa/ aleatoria/ difícil.
2. Requiere mucho tiempo/ esfuerzo.
3. Requiere un poco de tiempo/ esfuerzo.
4. Fácil de usar o carece de enlaces sin importancia.
5. Flujo muy lógico, fácil, claro e intuitivo. Ofrece accesos directos.
6. N/A.

9. Diseño gestual: ¿Las interacciones (pulsaciones, tecleos, deslizamientos) son consistentes e intuitivos en todos los componentes o pantallas?

1. Totalmente inconsistentes/ confusos.
2. A menudo inconsistentes/ confusos.
3. Normal con algunos elementos inconsistentes/ confusos.
4. Principalmente consistente/ intuitivo con problemas menores.
5. Muy consistente e intuitivo.
6. N/A.

### **SECCIÓN C Estética- diseño gráfico, atractivo visual, combinación de colores y consistencia en estilo.**

10. Disposición: ¿Es adecuada la disposición y tamaño de los botones/ iconos/ menús/ contenidos de la pantalla?

1. Diseño muy malo, desordenado. Algunas opciones no pueden seleccionarse/ localizarse/ verse/ leerse. No está optimizada a la pantalla.
2. Mal diseño, aleatorio, poco claro. Algunas opciones son difíciles de seleccionar/localizar/ ver/ leer.
3. Satisfactorio. Algunos problemas con la selección/ localización/ vista/ lectura o problemas menores de tamaño de pantalla.
4. Principalmente claro. Asuntos fáciles de seleccionar/ localizar/ ver/ leer.
5. Profesional, simple, claro, ordenado, lógico, exposición optimizada. Cada componente del diseño tiene un propósito.
6. N/A.

11. Gráficos: ¿Cómo es la calidad/ resolución de los gráficos usados como botones/ iconos/ menús/ contenidos?

1. Los gráficos tienen un diseño visual muy pobre- desproporcionado, un estilo completamente inconsistente.
2. Los gráficos tienen una calidad/ resolución baja- desproporcionado, inconsistente estilísticamente.
3. Los gráficos y el diseño visual tienen una calidad moderada (generalmente consistentes en estilo).
4. Calidad/ resolución alta- diseño visual apropiado, consistente en estilo.
5. Calidad/ resolución muy alta- diseño visual muy apropiado, consistente en estilo.
6. N/A

12. Atractivo visual: ¿Tiene la app buena apariencia?

1. No es visualmente atractiva. Tiene una apariencia desagradable. El diseño es pobre, los colores son incompatibles.
2. Poco atractivo visual. El diseño es pobre, mal uso del color. Es visualmente aburrida.
3. Atractivo visual mediocre, ni agradable ni desagradable.
4. Atractiva visualmente. Diseño profesional.
5. Muy atractiva visualmente, diseño notablemente destacado; usa colores para intensificar las características/ menús.
6. N/A.

**SECCIÓN D Información- Contiene información de calidad (por ejemplo, textos, retroalimentación, medidas, referencias) de una fuente fiable. Selecciona N/A si el componente es irrelevante.**

13. Calidad de la información: ¿El contenido es correcto, está bien escrito y es coherente con los objetivos/temas de la aplicación?

1. Es irrelevante/ inapropiada/ incoherente/ incorrecta.
2. Poco relevante/ apropiada/ coherente/ correcta.
3. Moderadamente relevante/ apropiada/ coherente/ correcta.
4. Relevante/ apropiada/ coherente/ correcta.
5. Muy relevante, apropiada, coherente y correcta.
6. N/A - No hay información en la aplicación.

14. Cantidad de información: ¿la información incluida en la app es exhaustiva y concisa?

1. Mínima o abrumadora.
2. Insuficiente o posiblemente abrumadora.
3. Aceptable, pero ni exhaustiva ni concisa.
4. Ofrece un amplio abanico de información, tiene alguna laguna o detalles innecesarios; o no tiene enlaces a más información y recursos.
5. Exhaustiva y concisa; contiene enlaces a más información y recursos.
6. N/A - No hay información en la aplicación.

15. Información visual: ¿Es lógica, clara y correcta la explicación visual de los conceptos- a través de gráficos/ imágenes/ videos etc.?

1. Confusa, poco clara, errónea o ausente de lo necesario.
2. Principalmente confusa, poco clara, errónea.
3. Aceptable, pero confusa, poco clara, errónea con frecuencia.
4. Principalmente clara, lógica y correcta con fallos insignificantes.
5. Perfectamente clara, lógica y correcta.
6. N/A – No hay información visual en la aplicación.

16. Credibilidad: ¿La información que contiene la app parece provenir de una fuente fiable)?

1. El origen de la información no es fiable en absoluto.
2. El origen de la información carece de suficiente fiabilidad.
3. La información no es inadecuada pero no se conoce con claridad la fuente de la que proviene.
4. La información proviene de una fuente que puede ser fiable.
5. La información, con total seguridad, proviene de una fuente fiable o especializada.
6. N/A – No hay información de la aplicación.

## **CALIDAD SUBJETIVA DE LA APLICACIÓN**

### **SECCIÓN E**

17. ¿Recomendarías esta aplicación a personas que pudieran obtener beneficio de ella?

1. En absoluto..... No recomendaría esta aplicación a nadie.
2. .... La recomendaría a pocas personas.
3. Quizá..... La recomendaría a algunas personas.
4. .... La recomendaría a muchas personas.
5. Definitivamente..... La recomendaría a todo el mundo.
6. N/A

18. ¿En los próximos 12 meses, ¿cuántas veces crees que usarías esta aplicación si fuera relevante para ti?

1. Ninguna.
2. 1-2
3. 3-10
4. 10-50
5. >50
6. N/A

19. ¿Pagarías por obtener la aplicación?

1. Definitivamente no.
2. ....
3. ....
4. ....
5. Definitivamente sí.
6. N/A

20. ¿Cuál es la calificación general de la aplicación?

1. Una de las peores aplicaciones que he usado.
2. ....
3. Correcta.
4. ....
5. Una de las mejores aplicaciones que he usado.
6. N/A



## IMPACTO PERCIBIDO

### SECCIÓN F

1. Conciencia: Es probable que la aplicación me haga tomar conciencia de la importancia de abordar la conducta saludable.

1. Totalmente en desacuerdo
- 2.
- 3.
- 4.
5. Totalmente de acuerdo
6. N/A

2. Conocimiento: Es probable que la aplicación aumente mi conocimiento/comprensión acerca de la conducta saludable.

1. Totalmente en desacuerdo
- 2.
- 3.
- 4.
5. Totalmente de acuerdo
6. N/A

3. Actitudes: Es probable que la aplicación cambie mi actitud para mejorar la conducta saludable.

1. Totalmente en desacuerdo
- 2.
- 3.
- 4.
5. Totalmente de acuerdo
6. N/A

4. Intención de cambio: Es probable que la aplicación aumente mi intención/motivación para abordar la conducta saludable.

1. Totalmente en desacuerdo
- 2.
- 3.
- 4.
5. Totalmente de acuerdo
6. N/A

5. Búsqueda de ayuda: Es probable que la aplicación me motive a buscar ayuda para realizar la conducta saludable.

1. Totalmente en desacuerdo
- 2.
- 3.
- 4.
5. Totalmente de acuerdo
6. N/A

6. Cambio de comportamiento: Utilizando esta app realizaré la conducta saludable.

1. Totalmente en desacuerdo
- 2.
- 3.
- 4.
5. Totalmente de acuerdo
6. N/A

¿Quieres incluir algún comentario acerca de la app?



### 8.4.1. Respuestas de los participantes

En las tablas 1 y 2, se registran las respuestas de los diez participantes del proceso de evaluación de la aplicación. Ambos grupos están conformados por 5 participantes. En las casillas de cada pregunta se registra el número total de participantes que eligieron cada puntuación.

CATEGORÍA / Pregunta	1	2	3	4	5	N/A
<b>SECCIÓN A: PARTICIPACIÓN</b>						
Entretenimiento	0	1	3	1	0	0
Interés	0	0	0	4	1	0
Personalización	0	1	2	1	1	0
Interacción	0	0	3	1	1	0
Público objetivo	0	0	0	2	3	0
<b>SECCIÓN B: FUNCIONALIDAD</b>						
Rendimiento	0	0	0	1	4	0
Facilidad de uso	0	0	1	3	1	0
Navegación	0	0	0	2	3	0
Diseño gestual	0	0	1	2	2	0
<b>SECCIÓN C: ESTÉTICA</b>						
Disposición	0	0	1	1	3	0
Gráficos	0	0	1	2	2	0
Atractivo visual	0	0	2	2	1	0
<b>SECCIÓN D: INFORMACIÓN</b>						
Calidad de la información	0	0	0	2	1	2
Cantidad de información	0	0	3	0	1	1
Información visual	0	0	1	2	1	1
Credibilidad	0	0	1	1	3	0
<b>SECCIÓN E: CALIDAD SUBJETIVA</b>						
Recomendación	0	0	1	1	3	0
Uso esperado en 12 meses	0	0	1	2	2	0
Pago por la app	0	2	2	1	0	0
Calificación general	0	0	1	2	2	0
<b>SECCIÓN F: IMPACTO</b>						
Conciencia	0	0	2	2	1	0
Conocimiento	0	0	3	1	1	0
Actitudes	0	0	3	1	1	0
Intención de cambio	0	0	2	2	1	0
Búsqueda de ayuda	0	0	2	1	2	0
Cambio de comportamiento	0	0	2	1	0	2

Tabla 1. Valoración de IDPBook de pacientes según la escala uMARS.

CATEGORÍA / Pregunta	1	2	3	4	5	N/A
<b>SECCIÓN A: PARTICIPACIÓN</b>						
Entretenimiento	0	0	3	2	0	0
Interés	0	0	0	4	1	0
Personalización	0	0	2	3	0	0
Interacción	0	0	1	4	0	0
Público objetivo	0	0	0	0	5	0
<b>SECCIÓN B: FUNCIONALIDAD</b>						
Rendimiento	0	0	0	1	4	0
Facilidad de uso	0	0	0	0	5	0
Navegación	0	0	0	0	5	0
Diseño gestual	0	0	0	4	1	0
<b>SECCIÓN C: ESTÉTICA</b>						
Disposición	0	0	0	5	0	0
Gráficos	0	0	0	4	1	0
Atractivo visual	0	0	0	4	1	0
<b>SECCIÓN D: INFORMACIÓN</b>						
Calidad de la información	0	0	0	5	0	0
Cantidad de información	0	0	0	5	0	0
Información visual	0	0	0	4	1	0
Credibilidad	0	0	0	0	5	0
<b>SECCIÓN E: CALIDAD SUBJETIVA</b>						
Recomendación	0	0	0	0	5	0
Uso esperado en 12 meses	0	0	0	5	0	0
Pago por la app	0	0	2	3	0	0
Calificación general	0	0	2	3	0	0
<b>SECCIÓN F: IMPACTO</b>						
Conciencia	0	0	0	4	1	0
Conocimiento	0	0	0	4	1	0
Actitudes	0	0	0	4	1	0
Intención de cambio	0	0	0	4	1	0
Búsqueda de ayuda	0	0	0	3	2	0
Cambio de comportamiento	0	0	0	3	2	0

Tabla 2. Valoración de IDPBook de sanitarios y programadores según la escala uMARS.

Por otra parte, ningún participante registro comentarios en el espacio destinado a observaciones sobre la aplicación.