



Diseño y desarrollo de una aplicación móvil para una solución de atención sociosanitaria remota

Memoria de Proyecto Final de Máster

Máster Universitario en Aplicaciones Multimedia

Área TFM Profesionalizadora

Autor: Daniel Guzmán López

Consultor: Mikel Zorrilla Berasategui

Profesora: Laura Porta Simó

05/06/2020

Créditos

Librerías Flutter:

Nombre	Licencia	Copyright	Condiciones
http	BSD 3-Clause "New" or "Revised" License	Copyright 2014, the Dart project authors. All rights reserved.	https://github.com/dart-lang/http/blob/master/LICENSE
google_maps_flutter	BSD	Copyright 2018 The Chromium Authors. All rights reserved.	https://github.com/flutter/plugins/blob/master/packages/google_maps_flutter/google_maps_flutter/LICENSE
flutter_bluetooth_serial	MIT License	Copyright (c) 2018 Eduardo Folly	https://github.com/edufolly/flutter_bluetooth_serial/blob/master/LICENSE
flutter_bloc	MIT License	Copyright (c) 2018 Felix Angelov	https://github.com/felangel/bloc/blob/master/packages/flutter_bloc/LICENSE
scoped_model	BSD	Copyright 2017 The Fuchsia Authors. All rights reserved.	https://github.com/brianegan/scoped_model/blob/master/LICENSE
equatable	MIT License	Copyright (c) 2018 Felix Angelov	https://github.com/felangel/equatable/blob/master/LICENSE
intl	BSD	Copyright 2013, the Dart project authors. All rights reserved.	https://github.com/dart-lang/intl/blob/master/LICENSE
local_auth	BSD	Copyright 2017 The Chromium Authors. All rights reserved.	https://github.com/flutter/plugins/blob/master/packages/local_auth/LICENSE
flutter_local_notifications	BSD	Copyright 2018 Michael Bui. All rights reserved.	https://github.com/MaikuB/flutter_local_notifications/blob/master/flutter_local_notifications/LICENSE
hive	Apache 2.0	Copyright 2019 Simon Leier	https://github.com/hivedb/hive/blob/master/hive/LICENSE
path_provider	BSD	Copyright 2017, the Flutter project authors. All rights reserved.	https://github.com/flutter/plugins/blob/master/packages/path_provider/path_provider/LICENSE
hive_generator	Apache 2.0	Copyright 2019 Simon Leier	https://github.com/hivedb/hive/blob/master/hive_generator/LICENSE
build_runner	BSD	Copyright 2016, the Dart project authors. All rights reserved.	https://github.com/dart-lang/build/blob/master/build_runner/LICENSE
flutter_lock_screen	MIT License	The MIT License (MIT) Copyright © 2018 Yasin Ilhan	https://github.com/kalismeras61/flutter_lock_screen/blob/master/LICENSE

Librerías Arduino:

Nombre	Licencia	Copyright	Condiciones
ArduinoJson	MIT License	Copyright © 2014-2020 Benoit BLANCHON	https://github.com/bblanchon/ArduinoJson/blob/6.x/LICENSE.md

Recursos gráficos:

- Icon made by Freepik (<https://www.flaticon.com/authors/freepik>) from Flaticon (www.flaticon.com). Flaticon License: Free for personal and commercial purpose with attribution.
- Icon made by Eucalyp (<https://www.flaticon.com/authors/eucalyp>) from Flaticon (www.flaticon.com). Flaticon License: Free for personal and commercial purpose with attribution.
- Font Roboto (Google Fonts), Designed by Christian Robertson. License: Apache License, Version 2.0.

Copyright



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada
[3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>Diseño y desarrollo de una aplicación móvil para una solución de atención sociosanitaria remota</i>
Nombre del autor:	<i>Daniel Guzmán López</i>
Nombre del consultor/a:	<i>Mikel Zorrilla Berasategui</i>
Nombre del PRA:	<i>Laura Porta Simó</i>
Fecha de entrega:	<i>06/2020</i>
Titulación:	<i>Máster Universitario en Aplicaciones Multimedia</i>
Área del Trabajo Final:	<i>Área TFM Profesionalizadora</i>
Idioma del trabajo:	<i>Español</i>
Palabras clave	<i>app eSalud, atención sociosanitaria, dependencia</i>
Resumen del Trabajo:	
<p>Las proyecciones de ciertos indicadores demográficos, como la esperanza de vida o la tasa de dependencia, revelan un incremento en las necesidades de atención sociosanitaria de la población y, por tanto, un aumento del gasto derivado de los servicios necesarios para cubrirlas. La aplicación de las TIC (Tecnologías de la Información y la Comunicación) al sector sociosanitario será clave en la sostenibilidad económica y calidad de sus servicios. Estamos ante un mercado en pleno desarrollo, con un notable impacto social y gran recorrido para la innovación.</p> <p>Este proyecto aborda el diseño e implementación de una aplicación móvil para un servicio de asistencia sociosanitaria remota. El trabajo realizado pretende contribuir al desarrollo de soluciones eficientes para el sector sociosanitario, en concreto, producir una aplicación versátil que se adapte a diferentes tecnologías y ámbitos de utilización, ya sea la prestación del servicio sociosanitario en un marco geográfico amplio o en contextos más reducidos con menor disponibilidad de recursos.</p> <p>La aplicación móvil está orientada a formar parte de una solución tecnológica más amplia que incluya equipamiento hardware con el cual sea interoperable. Se han empleado en su construcción las últimas tecnologías de mercado, produciendo una base de código única para su evolución multidispositivo y multiplataforma.</p> <p>Se ha puesto foco en este trabajo en el análisis, diseño e implementación de un conjunto de funcionalidades de valor añadido de la aplicación móvil, como la interacción con dispositivos hardware, planteando el resto como futuras líneas de trabajo para el posterior desarrollo de la aplicación y solución.</p>	

Abstract:

Projections of certain demographic indicators, such as life expectancy or the care dependency ratio, reveal an increase in the population's socio-health care needs and, therefore, an increase in the expense derived from the services necessary to cover them. The application of ICT (Information and Communication Technologies) to the social health sector will be key in the economic sustainability and quality of its services. We are facing a market in full development, with a notable social impact and a long path for innovation.

This project addresses the design and implementation of a mobile application for a remote socio-sanitary assistance service. The work carried out aims to contribute to the development of efficient solutions for the healthcare sector, in particular, to produce a versatile application that adapts to different technologies and areas of use, be it the provision of the healthcare service in a broad geographical framework or in smaller contexts. with less availability of resources.

The mobile application is aimed at being part of a broader technological solution that includes hardware equipment with which it is interoperable. The latest market technologies have been used in its construction, producing a unique code base for its multi-device and multi-platform evolution.

The focus of this work has been on the analysis, design and implementation of a set of value-added functionalities of the mobile application, such as interaction with devices hardware, posing the rest as future lines of work for the further development of the application and solution.

Agradecimientos

Gracias a Mikel, por el soporte, orientación y excelente tutorización durante todo el proyecto.

Gracias a Esther, por sus consejos, paciencia y apoyo en estos días extraños de confinamiento.

Abstract

Projections of certain demographic indicators, such as life expectancy or the care dependency ratio, reveal an increase in the population's socio-health care needs and, therefore, an increase in the expense derived from the services necessary to cover them. The application of ICT (Information and Communication Technologies) to the social health sector will be key in the economic sustainability and quality of its services. We are facing a market in full development, with a notable social impact and a long path for innovation.

This project addresses the design and implementation of a mobile application for a remote socio-sanitary assistance service. The work carried out aims to contribute to the development of efficient solutions for the healthcare sector, in particular, to produce a versatile application that adapts to different technologies and areas of use, be it the provision of the healthcare service in a broad geographical framework or in smaller contexts. with less availability of resources.

The mobile application is aimed at being part of a broader technological solution that includes hardware equipment with which it is interoperable. The latest market technologies have been used in its construction, producing a unique code base for its multi-device and multi-platform evolution.

The focus of this work has been on the analysis, design and implementation of a set of value-added functionalities of the mobile application, such as interaction with devices hardware, posing the rest as future lines of work for the further development of the application and solution.

Keywords

app, eHealth, socio-health care, remote patient monitoring, telecare, telehealth, care dependency.

Resumen

Las proyecciones de ciertos indicadores demográficos, como la esperanza de vida o la tasa de dependencia, revelan un incremento en las necesidades de atención sociosanitaria de la población y, por tanto, un aumento del gasto derivado de los servicios necesarios para cubrirlos. La aplicación de las TIC (Tecnologías de la Información y la Comunicación) al sector sociosanitario será clave en la sostenibilidad económica y calidad de sus servicios. Estamos ante un mercado en pleno desarrollo, con un notable impacto social y gran recorrido para la innovación.

Este proyecto aborda el diseño e implementación de una aplicación móvil para un servicio de asistencia sociosanitaria remota. El trabajo realizado pretende contribuir al desarrollo de soluciones eficientes para el sector sociosanitario, en concreto, producir una aplicación versátil que se adapte a diferentes tecnologías y ámbitos de utilización, ya sea la prestación del servicio sociosanitario en un marco geográfico amplio o en contextos más reducidos con menor disponibilidad de recursos.

La aplicación móvil está orientada a formar parte de una solución tecnológica más amplia que incluya equipamiento hardware con el cual sea interoperable. Se han empleado en su construcción las últimas tecnologías de mercado, produciendo una base de código única para su evolución multidispositivo y multiplataforma.

Se ha puesto foco en este trabajo en el análisis, diseño e implementación de un conjunto de funcionalidades de valor añadido de la aplicación móvil, como la interacción con dispositivos hardware, planteando el resto como futuras líneas de trabajo para el posterior desarrollo de la aplicación y solución.

Palabras clave

app, eSalud, atención sociosanitaria, gestión remota de pacientes, teleasistencia, telemedicina, dependencia.

Índice

Capítulo 1: Introducción	16
1. Introducción.....	16
2. Definición	17
3. Objetivos generales.....	19
3.1 Objetivos principales	19
3.2 Objetivos secundarios.....	20
3.3 Alcance.....	20
4. Metodología y proceso de trabajo.....	21
4.1 Enfoque de gestión agile	21
4.2 Definición y escritura de historias de usuario	22
4.3 Definición de pila inicial de producto	22
4.4 Ciclo de sprint 1	24
4.5 Revisión y actualización pila de producto después de sprint 1	26
4.6 Ciclo de sprint 2.....	27
4.7 Revisión y actualización pila de producto después de sprint 2.....	29
4.8 Ciclo de sprint 3.....	30
5. Planificación	31
6. Presupuesto.....	33
7. Estructura del resto del documento.....	36
Capítulo 2: Análisis.....	37
1. Estado del arte	37
1.1. Servicios y soluciones de atención sociosanitaria remota	37
1.2. Tipos de aplicaciones para dispositivos móviles.....	38
1.3. Marcos de trabajo (frameworks) para crear aplicaciones móviles.....	40
1.4. Discusión del estado del arte para los objetivos del proyecto	42
2. Definición de especificaciones del producto	43
2.1 Épicas e historias de usuario	43
2.2 Épica 1: Consultar alertas asistenciales.....	44
2.3 Épica 2: Consultar personas en atención	46
2.4 Épica 3: Ejecutar acciones asistenciales remotas.....	49
2.5 Épica 4: Acceder a la aplicación	50

2.6 Épica 5: Configurar la aplicación	51
Capítulo 3: Diseño.....	52
1. Arquitectura general de la solución de movilidad Hygia	52
1.1 Diagrama general ámbito de utilización WWAN.....	54
1.2 Diagrama general ámbito de utilización WPAN.....	55
2. Arquitectura de la aplicación móvil Hygia	56
3. Arquitectura entorno simulación de equipamiento hardware de la solución	58
3.1 Objetivo	58
3.2 Diseño	58
4. Arquitectura de la información	61
4.1 Baliza (beacon).....	61
4.2 Sensor	61
4.3 Actuador.....	61
4.4 Alerta asistencial	62
4.5 Persona en atención	62
4.6 Indicador asistencial.....	62
4.7 Acción asistencial.....	63
5. Diseño gráfico e interfaces	63
5.1 Proceso de trabajo	63
5.2 Arquetipos.....	64
5.3 Wireframes.....	67
5.4 Usabilidad/UX.....	73
5.5 Diseño de componentes de interfaz	75
5.6 Estilos.....	77
6. Lenguajes de programación y APIs utilizadas.....	79
6.1 Flutter como framework de desarrollo	79
6.2 APIs de terceros utilizadas en Flutter.....	79
6.3 Placa Arduino para entorno de simulación hardware	80
6.4 APIs de terceros utilizadas en Arduino.....	81
Capítulo 4: Implementación	82
1 Implementación de la aplicación con el <i>framework</i> Flutter.....	82
1.1 Estructura y descripción de clases	82
1.1 Comunicación bluetooth App móvil - HCU.....	86

2. Construcción del entorno para simulación de equipamiento hardware de la solución	88
2.1 Componentes utilizados en la construcción del entorno	88
2.2 Entorno hardware implementado.....	88
2.3 Programación Arduino	89
Capítulo 5: Evidencias de prueba y demo	90
1. Épica: Acceder a la aplicación.....	90
1.1 Historia de usuario: Acceso biométrico: huella digital.....	90
1.2 Historia de usuario: Acceso con pin de seguridad por teclado	92
2. Épica: Consultar alertas asistenciales	94
2.1 Historia de usuario: Consultar el listado de alertas asistenciales	94
2.2 Historia de usuario: Consultar el mapa de alertas asistenciales.....	95
2.3 Historia de usuario: Consultar el detalle de las alertas asistenciales	96
2.4 Historia de usuario: Consultar las acciones asistenciales asociadas a las alertas asistenciales.....	98
2.5 Historia de usuario: Recibir notificaciones de las alertas asistenciales	100
2.6 Historia de usuario: Recuperar alertas asistenciales en un ámbito de red inalámbrica de área extensa (WWAN)	101
2.7 Historia de usuario: Recuperar alertas asistenciales en un ámbito de red inalámbrica de área personal (WPAN).....	103
3. Épica: Consultar personas en atención	104
3.1 Historia de usuario: Consultar el listado de personas en atención	104
3.2 Historia de usuario: Consultar el mapa de personas en atención	106
3.3 Historia de usuario: Consultar la información de detalle de las personas en atención	107
3.4 Historia de usuario: Consultar los indicadores asistenciales disponibles para las personas en atención	109
3.5 Historia de usuario: Recuperar personas en atención en un ámbito de red inalámbrica de área extensa (WWAN)	111
3.6 Historia de usuario: Recuperar personas en atención en un ámbito de red inalámbrica de área personal (WPAN).....	113
4. Épica: Ejecutar acciones asistenciales remotas	114
4.1 Historia de usuario: Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área extensa (WWAN)	114

4.2 Historia de usuario: Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área personal (WPAN)	115
5. Épica: Configurar la aplicación.....	116
5.1 Historia de usuario: Configurar conexión Bluetooth con Hygia Control Unit.....	116
Capítulo 6: Conclusiones y líneas de futuro	119
1. Conclusiones.....	119
1.1 Objetivos, conclusiones y lecciones aprendidas	119
1.2 Revisión detallada de los objetivos.....	120
1.3 Revisión de planificación y metodología	121
2. Líneas de futuro	122
Bibliografía	123
Anexos.....	127
Anexo A: Herramientas utilizadas en el proyecto	127
Anexo B: Entregables del proyecto.....	130

Figuras y tablas

Índice de tablas

Tabla 1. Pila de producto inicial	24
Tabla 2. Pila del sprint 1	25
Tabla 3. Retrospectiva del sprint 1.....	26
Tabla 4. Pila de producto después de sprint 1.....	27
Tabla 5. Pila del sprint 2.....	28
Tabla 6. Retrospectiva del sprint 2.....	29
Tabla 7. Pila de producto después de sprint 2.....	29
Tabla 8. Pila de sprint 3	30
Tabla 9. Planificación fases generales proyecto.....	31
Tabla 10. Hitos principales proyecto	31
Tabla 11. Presupuesto detalle ideación y definición inicial.....	33
Tabla 12. Presupuesto resumen ideación y definición inicial.....	33
Tabla 13. Presupuesto detalle ciclo de sprint	34
Tabla 14. Presupuesto resumen ciclo de sprint	34
Tabla 15. Presupuesto recursos materiales, licencias	34
Tabla 16. Presupuesto total proyecto.....	35
Tabla 17. Arquetipo usuario 1.....	64
Tabla 18. Arquetipo usuario 2.....	65
Tabla 19. Arquetipo usuario 3.....	66
Tabla 20. APIs de terceros utilizadas en Flutter	80
Tabla 21. APIs de terceros utilizadas en Arduino	81

Índice de diagramas

Diagrama 1. Gantt planificación global de proyecto.....	32
Diagrama 2. Ambito de utilización WWAN	54
Diagrama 3. Ambito de utilización WPAN	55
Diagrama 4. Arquitectura software de la aplicación	57
Diagrama 5. Diseño Fritzing entorno simulación hardware	60
Diagrama 6. Secuencia inicio conexión App - HCU.....	87
Diagrama 7. Secuencia comunicación App - HCU.....	87

Índice de ilustraciones

Ilustración 1. Épicas e Historias de usuario.....	43
Ilustración 2. Wireframe Consultar el listado de alertas asistenciales.....	67
Ilustración 3. Wireframe Consultar el mapa de alertas asistenciales	67
Ilustración 4. Wireframe Consultar el detalle de las alertas asistenciales 1.....	68
Ilustración 5. Wireframe Consultar el detalle de las alertas asistenciales 2.....	68
Ilustración 6. Consultar el listado de personas en atención.....	69
Ilustración 7. Consultar el mapa de personas en atención	69
Ilustración 8. Consultar la información de detalle de las personas en atención 1	70
Ilustración 9. Consultar la información de detalle de las personas en atención 2	70
Ilustración 10. Acceso biométrico: huella digital	71

Ilustración 11. Acceso con pin de seguridad por teclado.....	71
Ilustración 12. Configurar conexión bluetooth con Hygia Control Unit 1.....	72
Ilustración 13. Configurar conexión bluetooth con Hygia Control Unit 2.....	72
Ilustración 14. Menú (app drawer)	73
Ilustración 15. Patrón Hamburger menu - Navigation drawer	74
Ilustración 16. Patrón Card.....	74
Ilustración 17. Espacios de trabajo	75
Ilustración 18. Vista de listado	76
Ilustración 19. Vista de mapa	76
Ilustración 20. Diseño tarjetas	77
Ilustración 21. Iconos notificaciones	77
Ilustración 22. Iconos lanzador	78

Índice de capturas

Captura 1. Android Studio: Proyecto Flutter	82
Captura 2. Android Studio: assets	82
Captura 3. Android Studio: main.....	83
Captura 4. Android Studio: resources_utils	83
Captura 5. Android Studio: UI.....	83
Captura 6. Android Studio: blocs	84
Captura 7. Android Studio: domain	84
Captura 8. Android Studio: data.....	85
Captura 9. Acceso biométrico: huella digital – inicio (dispositivo real Xiaomi A1. Android 9)	90
Captura 10. Acceso biométrico: huella digital – no disponible (emulación Nexus 5X API 29 x86).....	91
Captura 11. Acceso con pin de seguridad por teclado - inicio (emulación Nexus 5X API 29 x86).....	92
Captura 12. Acceso con pin de seguridad por teclado - error (emulación Nexus 5X API 29 x86).....	93
Captura 13. Consultar el listado de alertas asistenciales (emulación Nexus 5X API 29 x86)	94
Captura 14. Consultar el mapa de alertas asistenciales (emulación Nexus 5X API 29 x86)	95
Captura 15. Consultar el detalle de las alertas asistenciales - listado (emulación Nexus 5X API 29 x86).....	96
Captura 16. Consultar el detalle de las alertas asistenciales - mapa (emulación Nexus 5X API 29 x86)	97
Captura 17. Consultar las acciones asistenciales asociadas a las alertas asistenciales: icono listado (emulación Nexus 5X API 29 x86).....	98
Captura 18. Consultar las acciones asistenciales asociadas a las alertas asistenciales: sección detalle (emulación Nexus 5X API 29 x86).....	99
Captura 19. Recibir notificaciones de las alertas asistenciales (dispositivo real Xiaomi A1. Android 9)	100
Captura 20. Respuesta a petición API REST mock configurada con JSONPlaceholder	101
Captura 21. Ejecución Debug Android Studio estrategia adquisición WWMA (API remota REST HTTP)	102
Captura 22. Ejecución Debug Android Studio estrategia adquisición WPMA (bluetooth).....	103
Captura 23. Consultar el listado de personas en atención - menú (emulación Nexus 5X API 29 x86).....	104
Captura 24. Consultar el listado de personas en atención (emulación Nexus 5X API 29 x86)	105
Captura 25. Consultar el mapa de personas en atención (emulación Nexus 5X API 29 x86)	106
Captura 26. Consultar la información de detalle de las personas en atención - listado (emulación Nexus 5X API 29 x86).....	107
Captura 27. Consultar la información de detalle de las personas en atención - mapa (emulación Nexus 5X API 29 x86).....	108

Captura 28. Consultar los indicadores asistenciales disponibles para las personas en atención (emulación Nexus 5X API 29 x86)	109
Captura 29. Recuperar personas en atención en un ámbito de red inalámbrica de área extensa (WWAN).....	111
Captura 30. Ejecución Debug Android Studio estrategia adquisición WWMA (API remota REST HTTP)	112
Captura 31. Recuperar personas en atención en un ámbito de red inalámbrica de área personal (WPAN)	113
Captura 32. Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área extensa (WWAN).....	114
Captura 33. Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área personal (WPAN).....	115
Captura 34. Barra de tareas (emulación Nexus 5X API 29 x86)	116
Captura 35. Pantalla de configuración 1 (emulación Nexus 5X API 29 x86)	116
Captura 36. Pantalla de configuración 2 (emulación Nexus 5X API 29 x86)	117
Captura 37. Pantalla de configuración 3 (emulación Nexus 5X API 29 x86)	117
Captura 38. Pantalla de configuración 4 (emulación Nexus 5X API 29 x86)	118
Captura 39. Pantalla de configuración 5 (emulación Nexus 5X API 29 x86)	118
Captura 40. Repositorio creado para el proyecto: https://github.com/daniuoc/hygia	128
Captura 41. JSONPlaceholder	128
Captura 42. Configuración Google Cloud.....	129

Índice de fotos

Foto 1. Entorno de simulación hardware.....	88
---	----

Capítulo 1: Introducción

1. Introducción

La esperanza de vida a nivel mundial, y en España en particular, no ha dejado de crecer en los últimos años. Según un reciente estudio realizado de forma conjunta por la OMS (Organización Mundial de la Salud), la Universidad de Washington, la Universidad de Northumbria y el Imperial College London [18], en algunos de los países objeto del estudio (dentro de los considerados como países desarrollados), la esperanza de vida superará los 90 años en 2030. En el caso de España se refleja un incremento notable, desde los 78,66 años en hombres y 84,83 en mujeres alcanzados en 2010 hasta los 83,47 y 88,07 años, respectivamente, previstos para 2030.

Por otra parte, las últimas proyecciones de población a largo plazo publicadas por el INE (Instituto Nacional de Estadística) [16] señalan que la población española de más de 65 años alcanzará el 25,2% del total en el año 2033. Además, el más reciente informe de proyección de hogares, publicado por el mismo organismo [17], establece como previsión para el 2033 sobrepasar la cifra de los 5,8 millones de hogares españoles en los que vive una sola persona, lo que supondría el 28,8% del total y un incremento del 24,1% respecto a principios de 2018. El número de personas que viven solas pasará de representar el 10,1% de la población total en 2018 al 11,9% en 2033.

En relación a la tasa de personas en situación de dependencia, es decir, personas con falta o pérdida de autonomía por razones derivadas de la edad, enfermedad o discapacidad [5], la AIREF (Autoridad Independiente de Responsabilidad Fiscal) considera que podría duplicarse en España desde el 25% actual hasta un rango de entre el 45-60% en 2050 [1].

Del análisis de los indicadores demográficos expuestos anteriormente (esperanza de vida, índice de envejecimiento, hogares según su composición, tasa de dependencia) se desprende un panorama de envejecimiento de la población. Las personas tienden a vivir más años y, con cada vez más frecuencia, solas y en situación de dependencia.

En este contexto surge también el mercado denominado *Silver Economy* que conceptualiza las necesidades de consumo para el segmento de población a partir de los 50 años, con un enorme futuro para las TIC en general y, en particular, para aquellos servicios que permitan a los mayores llevar una vida independiente pero con sus necesidades cubiertas [3]. Se estima que este mercado genere en los próximos años 88 millones de empleos en Europa [2].

Después del análisis realizado, se concluye que el sector sociosanitario y el mercado estudiado se presentan como una buena oportunidad para abordar un proyecto tecnológico.

En el ámbito personal, además de la motivación que supone realizar un proyecto en el marco de un sector dedicado a dar soporte y mejorar de la calidad de vida de las personas, tengo interés por aprender y trabajar diversos aspectos tecnológicos específicos que serán necesarios para la construcción del producto, por ejemplo, el desarrollo aplicaciones móviles con tecnologías de última generación o la interacción de este tipo de aplicaciones con elementos hardware.

Además de formación profesional en el campo de la electrónica, apporto experiencia previa personal en diversas tecnologías del campo de la domótica/inmótica, pero siempre desde una orientación hardware. Es decir, con un enfoque centrado en el diseño y construcción del dispositivo electrónico. En este caso, se presenta la oportunidad inmejorable de abordar un sistema de estas características desde otro punto de vista, enfocado a una solución de movilidad y aplicando las últimas tecnologías de mercado. Subrayo en este aspecto el interés personal en el desarrollo de software que interaccione con algún tipo de sistema o dispositivo hardware y, por tanto, la motivación por la vertiente del proyecto en el que la aplicación móvil interactuará con elementos hardware.

Otra de las motivaciones personales que impulsan el proyecto es el interés particular en realizar la producción completa de una aplicación móvil (algo que no he llevado a cabo nunca), incluyendo la fase de ideación, análisis, diseño y desarrollo.

Considero por tanto que el proyecto propuesto será muy enriquecedor, me permitirá poner en práctica todos los conocimientos adquiridos durante el máster y me abrirá nuevas perspectivas profesionales y personales.

2. Definición

El presente trabajo aborda la ejecución de un proyecto tecnológico en el ámbito del sector sociosanitario. En concreto, el proyecto se enmarca específicamente en el área de los sistemas de gestión remota de pacientes (GRP) o personas con necesidades asistenciales, una de las vertientes más prometedoras y con más recorrido para la innovación. Estos sistemas se convertirán en una de las principales palancas para lograr reducir el gasto de atención sociosanitaria [12] y, por tanto, se espera un gran desarrollo en el mercado.

El proyecto define una solución de movilidad para un servicio de monitorización y atención remota sociosanitaria. En este contexto, se propone abordar el diseño y construcción de uno de los componentes de dicha solución, en concreto, la aplicación móvil que servirá de herramienta principal para la prestación del servicio.

Los sistemas de gestión remota de pacientes van orientados a reducir el tiempo de respuesta en la atención de las necesidades las personas, así como, permitir un seguimiento en tiempo real que permita la detección precoz de las situaciones de riesgo [12]. Para este propósito, una solución de movilidad como la propuesta en este trabajo se considera que puede ser especialmente útil y aportar

un claro valor añadido. El carácter remoto de la solución ofrece también evidentes ventajas en cuanto al ahorro de recursos al evitar desplazamientos a domicilios, habitaciones, etc.

El objetivo principal de la solución de movilidad propuesta es poder monitorizar y proporcionar asistencia de forma remota y en tiempo real a aquellas personas con algún tipo de necesidad sociosanitaria, a partir de ahora *personas en atención*. Para ello, se propone que la solución integre la adquisición de datos e interacción con dispositivos de distinto tipo: sensores (temperatura, humedad, presencia, etc.), balizas IoT (The Internet of Things), dispositivos wearables (relojes, pulseras de actividad, gafas, etc.), dispositivos médicos (tensiómetros, glucómetros, dispensador de medicamentos, etc.), etc.

La aplicación móvil, objeto de este proyecto, será la herramienta que los responsables de las personas en atención -a partir de ahora, *asistentes*- tendrán a su disposición en la solución para gestionar el servicio de atención. Son por tanto los asistentes los usuarios finales y para quienes va diseñada y dirigida la aplicación. En este sentido, el diseño de la aplicación se orienta hacia la sencillez, con un interfaz muy funcional que busca la fluidez en la interacción y un tiempo de respuesta óptimo.

Desde la aplicación móvil los Asistentes podrán prestar la atención remota a las personas en atención en base a dos tipos de interacciones principales:

- Monitorización: recibirá alertas asistenciales cuando se produzca alguna situación que requiera su atención. Podrá consultar además el conjunto de personas bajo su responsabilidad y el estado de los indicadores definidos para cada una de ellas.
- Actuación: el Asistente podrá interactuar sobre distintos actuadores asociados a la persona en atención, por ejemplo, accionar un dispensador de medicamentos.

Se propone, como característica importante y diferencial de la solución y aplicación móvil objeto de este trabajo, la adaptabilidad, es decir, que puedan adaptarse a diferentes ámbitos de utilización. El objetivo es que puedan cubrir servicios de distinta naturaleza, ya sean servicios complejos como la gestión de un conjunto de personas dentro de un ámbito geográfico extenso, por ejemplo una ciudad, pero que también puedan cubrir servicios más sencillos como la monitorización de una pequeña residencia geriátrica o una comunidad de pisos tutelados. Esta flexibilidad ofrece una ventaja de cara a una posible comercialización y penetración en el mercado.

A continuación se describen algunos casos de ejemplo de utilización de la solución de movilidad para el servicio especificado:

- Monitorización de una persona de edad avanzada: la solución permitiría a un asistente a través de la aplicación móvil monitorizar en cualquier momento y lugar a la persona en atención, accediendo a la información proporcionada por el conjunto de indicadores definidos para el

caso y recibiendo alertas en determinadas circunstancias de riesgo, por ejemplo, control de caídas, pulsador de llamada de emergencia, etc.

- Tutela de una persona con algún tipo de deterioro cognitivo: la solución, además de ofrecer una monitorización continua de los indicadores definidos para el caso, permitiría también la actuación sobre ciertos elementos que requieran control específico y puntual por parte del asistente, por ejemplo, accionar de forma remota un dispensador de medicamentos.
- Monitorización y control de los residentes de una residencia geriátrica: en este caso la solución de movilidad permite una atención ágil y descentralizada por parte del asistente (no es necesario la presencia física de una persona en el control de la planta gestionando la centralita, algo muy extendido en las soluciones actuales de mercado). La atención remota a través de la aplicación permitirá al asistente dar cobertura a un mayor número de personas, aumentando la eficiencia y reduciendo la necesidad de recursos necesarios para la atención.

3. Objetivos generales

3.1 Objetivos principales

Objetivos de la aplicación/solución:

- Diseñar e implementar una aplicación móvil que sirva de herramienta principal en la prestación de un servicio de atención sociosanitaria remota.
- Permitir que la aplicación móvil pueda integrarse en una solución tecnológica más amplia que utilice equipamiento hardware con el cual sea interoperable.
- Contemplar en la aplicación dos ámbitos de utilización para cubrir servicios de atención de distintas características:
 - ámbito de utilización WWAN (Wireless Wide Area Network o Red Inalámbrica de Área Extensa): es decir, con un dispositivo móvil con acceso 4G/5G.
 - ámbito de utilización WPAN (Wireless Personal Area Network o Red Inalámbrica de Área Personal): es decir, con un dispositivo móvil con acceso Bluetooth.
- Construir una aplicación con una interfaz gráfica sencilla pero de alto rendimiento y fluidez gráfica para dar cobertura a tiempos de respuesta exigentes. Probar la interacción de la aplicación con hardware real para verificar la consecución del objetivo planteado.
- Construir la aplicación sobre una base única de código que permita su evolución multiplataforma y multidispositivo, para reducir costes de desarrollo y mantenimiento y facilite la entrada del producto en el mercado.

Objetivos para el usuario:

- Permitir a los asistentes gestionar desde su dispositivo móvil la atención a un conjunto de personas ubicadas en distintas localizaciones dentro un ámbito geográfico determinado.
- Permitir a los asistentes monitorizar desde la aplicación móvil a las personas en atención mediante la captura y presentación de la información recogida por un conjunto de dispositivos hardware configurados en la solución que, de forma general, denominaremos *sensores*.

- Permitir a los asistentes actuar desde la aplicación móvil sobre un conjunto de dispositivos hardware específicos disponibles en la solución –que de forma general denominaremos *actuadores*– asociados a las personas en atención.

Objetivos personales del autor de este trabajo:

- Ejecutar de forma completa el proceso de producción una aplicación multimedia (ideación, análisis, diseño, implementación y pruebas) poniendo en práctica todos los conocimientos adquiridos en el máster.
- Aprender y poner en práctica las últimas tecnologías y herramientas de mercado de desarrollo de aplicaciones móviles.
- Aprender y experimentar con la interacción entre tecnologías móviles con comunicaciones y dispositivos hardware.

3.2 Objetivos secundarios

- Diseñar y abordar los primeros pasos de una idea o solución tecnológica que pueda tener continuidad más allá de este TFM.

3.3 Alcance

Sobre los objetivos anteriormente planteados, a continuación, se concreta el alcance específico que se abordará en este trabajo:

- La aplicación móvil formará parte de una solución tecnológica más amplia, no obstante, el alcance el proyecto se centrará en el diseño y desarrollo de dicha aplicación móvil, no al resto de componentes de la solución.
- Con relación a los componentes de la solución que quedan fuera del alcance, como por ejemplo el hardware de adquisición de datos, los drivers de comunicación con los sensores y actuadores, o posibles servicios *backend* adicionales, se realizará la simulación de estos mediante el componente que se considere más conveniente para poder probar la aplicación móvil, ya sea hardware (por ejemplo, placa open-hardware Arduino) o software (mockups).
- Dentro de las posibles tipologías de comunicación contempladas por la solución se implementará y testeará la correspondiente a la tecnología Bluetooth, dejando el resto para futuras líneas de desarrollo de la solución.
- La implementación se llevará a cabo mediante tecnologías con enfoque multiplataforma y multidispositivo, no obstante, la validación de la aplicación se llevará a cabo en un dispositivo smartphone Android.
- Se priorizará y se pondrá foco en el diseño e implementación de aquellos casos de uso asociados a la interacción con hardware (lectura de los sensores e interacción con los actuadores) para poder validar el rendimiento de la interfaz, uno de los objetivos principales del trabajo.

4. Metodología y proceso de trabajo

4.1 Enfoque de gestión agile

Este proyecto se ha abordado con enfoque de gestión ágil. En concreto, se ha adoptado un marco de trabajo Scrum, utilizando aquellos procesos más adecuados y que puedan ser de mayor utilidad a las características del proyecto, como por ejemplo, el ciclo de vida iterativo e incremental. Lógicamente, se ha prescindido de aquellas prácticas Scrum no aplicables un trabajo individual de estas características (por ejemplo, las reuniones *daily stand-up*). La justificación para la decisión de aplicar este enfoque ágil fue intentar:

- gestionar mejor el alcance y las prioridades en el desarrollo mediante una pila de producto dinámica en lugar de un conjunto de requisitos cerrado.
- poder incorporar de forma natural y en cualquier etapa del proyecto el *feedback* del tutor.
- mejorar la respuesta al cambio.
- identificar lo antes posibles problemas de viabilidad.
- tener capacidad de adaptación frente al empleo de nuevas tecnología de desarrollo.

Una vez concluida la fase inicial de ideación y elaborada la propuesta y mandato de proyecto, el trabajo se organizó en tres ciclos de sprint, con una distribución adaptada a los porcentajes de desarrollo especificados en las distintas PEC:

- PEC 3: Entrega 1
 - Definición y escritura de historias de usuario
 - Definición de pila inicial de producto (*product backlog*)
 - Ciclo de sprint 1
 - Planificación del sprint (*sprint backlog*)
 - Sprint
 - Revisión del sprint
 - Retrospectiva del sprint
 - Release 1 del producto
 - Revisión y actualización pila de producto después de sprint 1
- PEC 4: Entrega 2
 - Ciclo de sprint 2
 - Planificación del sprint (*sprint backlog*)
 - Sprint
 - Revisión del sprint
 - Retrospectiva del sprint
 - Release 2 del producto
 - Revisión y actualización pila de producto después de sprint 2

- PEC 5: Cierre
 - Ciclo de sprint 3
 - Planificación del sprint (*sprint backlog*)
 - Sprint
 - Revisión del sprint
 - Release 3 del producto

4.2 Definición y escritura de historias de usuario

En esta etapa se realizó, como técnica de análisis funcional, la definición y escritura de las historias de usuario. Las historias de usuario obtenidas representan las funcionalidades o *features* iniciales esperadas para el producto. Dada la naturaleza agile del trabajo, esta información es dinámica y evolucionará en las siguientes iteraciones, incorporándose alguna historia de usuario adicional si se considera necesario, o bien, descartando alguna de las definidas porque no aporte valor añadido al producto o presente algún inconveniente. Puede consultarse toda la información relativa a las historias de usuario obtenidas en el apartado “4. Definición de especificaciones del producto” de este documento.

En esta etapa se decide abordar también los primeros trabajos de análisis y definición de UX (User eXperience o Experiencia de Usuario). En el apartado “3. Diseño gráfico e interfaces” de este documento se incluye el proceso y resultado de estos trabajos.

4.3 Definición de pila inicial de producto

En esta etapa se generó el *product backlog* o pila de producto inicial del proyecto, es decir, el conjunto de elementos ordenados por prioridad identificados como necesarios para terminar el producto. Para ello, se procedió a realizar el levantamiento de todas aquellos trabajos que, adicionalmente a las historias de usuario ya identificadas en la tarea anterior, son necesarios llevar a cabo para la construcción del producto. El conjunto resultante incluye elementos de tres tipos:

- *Feature*: funcionalidades de la aplicación, representadas por las historias de usuario.
- *Technical work*: trabajos de carácter técnico (configuración entornos, arquitectura, etc.).
- *Knowledge acquisition*: trabajos de investigación y adquisición de los conocimientos necesarios para la construcción del producto.

El conjunto de elementos del *product backlog* inicial se presenta agrupado en *releases* o entregas. Es importante subrayar que el *product backlog* es un artefacto dinámico del proyecto que evoluciona durante el transcurso del mismo, según se finalicen los elementos o se considere y justifique conveniente añadir nuevos o descartar elementos existentes. El objetivo fundamental del *product backlog* –y del resto de artefactos del marco de trabajo que estamos utilizando– no será otro que ayudarnos a conseguir los objetivos marcados en el proyecto.

Id	Product backlog ítem	Tipo
1	Análisis y selección de herramientas	Technical work
2	Instalación y configuración del entorno de desarrollo	Technical work
3	Formación en el framework específico seleccionado para el desarrollo	Knowledge acquisition
4	Diseño de la arquitectura de la aplicación	Technical work
5	Diseño del modelo de clases de la aplicación	Technical work
6	Construir el esqueleto base de la aplicación según el diseño de arquitectura	Technical work
7	Recuperar alertas asistenciales en un ámbito de red inalámbrica de área extensa (WWAN)	Features
8	Consultar el listado de alertas asistenciales	Features
9	Consultar el mapa de alertas asistenciales	Features
10	Estudio de utilización hardware/software de placa open hardware Arduino	Knowledge acquisition (<i>spike</i>)
Release 1		
11	Diseño y construcción de entorno hardware simulación WPAN con Arduino	Technical work
12	Recuperar alertas asistenciales en un ámbito de red inalámbrica de área personal (WPAN)	Features
13	Consultar el detalle de las alertas asistenciales	Features
14	Consultar las acciones asistenciales asociadas a las alertas asistenciales	Features
15	Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área personal (WPAN)	Features
16	Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área extensa (WWAN)	Features
17	Consultar el listado de personas en atención	Features
18	Consultar el mapa de personas en atención	Features
19	Consultar la información de detalle de las personas en atención	Features
20	Consultar los indicadores asistenciales disponibles para las personas en atención	Features
21	Recuperar personas en atención en un ámbito de red inalámbrica de área personal (WPAN)	Features
22	Recuperar personas en atención en un ámbito de red inalámbrica de área extensa (WWAN)	Features
Release 2		
23	Recibir notificaciones de las alertas asistenciales	Features

24	Revisión y mejora aspecto visual aplicación	Features
25	Pruebas integradas aplicación	Technical work
Release 3		

Tabla 1. Pila de producto inicial

4.4 Ciclo de *sprint* 1

Planificación del *sprint* 1

En esta etapa se generó el listado concreto de tareas a abordar en la etapa de desarrollo del *sprint* 1.

Como resultado, se obtuvo el primer *sprint backlog*. Para ello:

- se seleccionaron del *product backlog* aquellos ítems a abordar en el *sprint* actual, teniendo en cuenta la prioridad establecida en dicho *backlog*.
- se realizó la descomposición de cada ítem seleccionado en un conjunto de tareas concretas.

<i>Producto backlog</i> ítem	Id	Tarea de <i>sprint</i>
Análisis y selección de herramientas	1	Búsqueda de herramientas
	2	Registro online / Instalación de herramientas
Instalación y configuración del entorno de desarrollo	3	Instalación y configuración de entorno de desarrollo
	4	Instalación y configuración <i>Framework</i> específico de desarrollo
	5	Preparación dispositivo móvil Android para <i>debug</i>
	6	Alta y configuración repositorio GitHub
Formación en el <i>framework</i> específico seleccionado para el desarrollo	7	Estudio documentación, tutoriales
	8	Construir "Hola Mundo"
Diseño de la arquitectura de la aplicación	9	Búsqueda y estudio de arquitecturas de software para aplicaciones móviles
	10	Diseño de arquitectura y documentación
Diseño del modelo de clases de la aplicación	11	Diseño del modelo de clases
Construir el esqueleto base de la aplicación según el diseño de arquitectura	12	Crear proyecto de la aplicación y estructura de directorios
	13	Crear conjunto de clases base según arquitectura definida
Recuperar alertas asistenciales en un ámbito de red inalámbrica de área extensa (WWAN)	14	Habilitación servicio <i>mock</i> de monitorización remota de la solución sociosanitaria
	15	Desarrollo funcionalidad
	16	Pruebas integradas con servicio <i>mock</i> de monitorización remota de la solución sociosanitaria

Consultar el listado de alertas asistenciales	17	Definición y construcción clase mock para prueba unitaria
	18	Desarrollo
	19	Prueba unitaria con objetos <i>mock</i>
	20	Prueba integrada con servicio <i>mock</i> de monitorización remota
Consultar el mapa de alertas asistenciales	21	Alta en Google Cloud y activación API Google Maps
	22	Configuración entorno desarrollo y proyecto para utilización de Google Maps
	23	Definición y construcción clase mock para prueba unitaria
	24	Desarrollo
	25	Prueba unitaria con objetos mock
	26	Prueba integrada con servicio <i>mock</i> de monitorización remota
Estudio de utilización hardware/software de placa open hardware Arduinio	27	Montaje y configuración placa Arduino
	28	Construir "Hola Mundo"

Tabla 2. Pila del sprint 1

Sprint 1

En esta etapa se ejecutó la planificación de tareas establecida en la etapa anterior: *sprint backlog*. Surgen las primeras dificultades a la hora de enfrentarse una nueva tecnología: problemas en la configuración del entorno de desarrollo, dudas en el diseño, errores de ejecución, etc. En algunos casos se decide aplicar soluciones de compromiso para resolver algunas incidencias en la ejecución (por ejemplo, visualización incorrecta de los widgets en modo horizontal) y posponer su resolución para etapas posteriores. El objetivo fue siempre evitar bloqueos para cumplir la planificación establecida en el *sprint* y mantener un ritmo de trabajo estable.

Revisión del *sprint 1*

En esta tarea se llevó a cabo la revisión del resultado de *sprint* desarrollado en la etapa anterior mediante la realización de pruebas de las funcionalidades implementadas. Se procede a realizar pruebas en:

- el entorno de emulación de Android Studio: dispositivo Nexus 5X API 29 x86
- en un dispositivo móvil real Xiaomi A1. Android 9

Retrospectiva del *sprint* 1

En esta tarea se realizó la revisión del propio proceso de ejecución del *sprint*:

¿Qué ha salido bien?	Se han superado todos los problemas y dudas hasta alcanzar una arquitectura, una base de clases y un entorno funcionando correctamente que nos permitirá lograr un buen ritmo de desarrollo en la PEC 4.
¿Qué podría haber sido mejor?	Se podrían haber alcanzado mayor productividad en el desarrollo en presencia de menor número de incidencias del entorno/framework.
¿Qué cosas hay que probar?	Diferencia de rendimiento entre entorno de simulación y dispositivos reales
¿Qué problemas hay para escalar?	No detectado hasta la fecha

Tabla 3. Retrospectiva del *sprint* 1

4.5 Revisión y actualización pila de producto después de *sprint* 1

Tras la ejecución del primer ciclo de *sprint* del proyecto se procedió a actualizar el *product backlog* con la nueva situación. Se decide incluir un nuevo ítem “Optimización funcionalidades ya existentes en la aplicación” en el nuevo *backlog* para poder disponer en cada *sprint* de cierto tiempo para realizar refactorización y optimización de algunas funcionalidades ya entregadas. Como siempre, el objetivo es ir refinando hasta lograr la calidad esperada.

Por otra parte, durante el transcurso del *sprint* 1 se identificó también la conveniencia de incluir dos nuevas épicas para contemplar algunas funcionalidades adicionales necesarias para la aplicación, como el control de acceso o la sección de la configuración. Se incorporarán ambas funcionalidades al *product backlog* y se ajustó el contenido de las *releases*:

Id	<i>Product backlog</i> ítem	Tipo
1	Diseño y construcción de entorno hardware simulación WPAN con arduino	Technical work
2	Recuperar alertas asistenciales en un ámbito de red inalámbrica de área personal (WPAN)	Features
3	Consultar el detalle de las alertas asistenciales	Features
4	Consultar las acciones asistenciales asociadas a las alertas asistenciales	Features
5	Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área personal (WPAN)	Features
6	Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área extensa (WWAN)	Features
7	Consultar el listado de personas en atención	Features

8	Consultar el mapa de personas en atención	Features
9	Consultar la información de detalle de las personas en atención	Features
10	Recuperar personas en atención en un ámbito de red inalámbrica de área extensa (WWAN)	Features
11	Recibir notificaciones de las alertas asistenciales	Features
12	Configurar conexión bluetooth con Hygia Control Unit	Features
13	Acceso con pin de seguridad por teclado	Features
14	Acceso biométrico: huella digital	Features
15	Optimización funcionalidades ya existentes en la aplicación	Features
Release 2		
16	Recuperar personas en atención en un ámbito de red inalámbrica de área personal (WPAN)	Features
17	Consultar los indicadores asistenciales disponibles para las personas en atención	Features
18	Optimización funcionalidades ya existentes en la aplicación	Features
Release 3		

Tabla 4. Pila de producto después de sprint 1

4.6 Ciclo de sprint 2

Planificación del sprint

En esta etapa se generó el listado concreto de tareas a realizar en la fase de desarrollo del segundo *sprint*. Como resultado, se construyó el nuevo *sprint backlog*.

<i>Producto backlog</i> ítem	Id	Tarea de <i>sprint</i>
Diseño y construcción de entorno hardware simulación WPAN con Arduino	1	Diseño de esquema electrónico con herramienta open-source Fritzing.
	2	Montaje físico de componentes
	3	Programación placa Arduino: rutina de configuración de módulo bluetooth HC05
Recuperar alertas asistenciales en un ámbito de red inalámbrica de área personal (WPAN)	4	Programación placa Arduino: lectura de balizas y envío de información a app vía bluetooth
	5	Desarrollo nueva funcionalidad app: módulo carga información de configuración (Hygia Directory Information)
	6	Desarrollo nueva funcionalidad app: módulo de comunicación bluetooth - recepción de datos
	7	Desarrollo nueva funcionalidad app: tarea background de adquisición de datos
	8	Integrar la nueva funcionalidad de adquisición de datos en la pantalla existente de alertas asistenciales.

Consultar el detalle de las alertas asistenciales	9	Desarrollo nueva funcionalidad app
Consultar las acciones asistenciales asociadas a las alertas asistenciales	10	Desarrollo nueva funcionalidad app
Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área personal (WPAN)	11	Programación placa Arduino: recepción de eventos de app y activación de actuadores
	12	Ampliación funcionalidad app: módulo de comunicación bluetooth - envío de datos
	13	Ampliación funcionalidad app: ejecutar acción asistencial en contexto WPAN
Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área extensa (WWAN)	14	Ampliación funcionalidad app: envío de información vía API remota
	15	Ampliación funcionalidad app: ejecutar acción asistencial en ámbito WWAN
Consultar el listado de personas en atención	16	Desarrollo nueva funcionalidad app
Consultar el mapa de personas en atención	17	Desarrollo nueva funcionalidad app
Consultar la información de detalle de las personas en atención	18	Desarrollo nueva funcionalidad app
Recuperar personas en atención en un ámbito de red inalámbrica de área extensa (WWAN)	19	Ampliar servicio mock de monitorización remota de la solución sociosanitaria con información de personas
	20	Ampliación funcionalidad app estrategia de adquisición de datos remota para información de personas
Recibir notificaciones de las alertas asistenciales	21	Desarrollo nueva funcionalidad app
Configurar conexión bluetooth con Hygia Control Unit	22	Desarrollo nueva funcionalidad app
Acceso con pin de seguridad por teclado	23	Desarrollo nueva funcionalidad app
Acceso biométrico: huella digital	24	Desarrollo nueva funcionalidad app
Optimización funcionalidades ya existentes en la aplicación	25	Rediseño arquitectura
	26	Sustituir utilización componente material module tabs por contenedores independientes (solución problema eventos en mapas)
	27	Aplicación tema oscuro y mejoras ux
	28	Refactoring y limpieza de código

Tabla 5. Pila del sprint 2

Sprint

En esta etapa se ejecutaron las tareas planificadas en la etapa anterior. Como resultado, se ha obtenido la segunda *release* del producto, que supone alcanzar en torno al 90% de la funcionalidad prevista.

Revisión del sprint

En esta tarea se llevó a cabo la revisión del resultado de sprint desarrollado en la etapa anterior, mediante la realización de pruebas de las funcionalidades implementadas. Se procede a realizar pruebas integradas con:

- el entorno de emulación de Android Studio: dispositivo Nexus 5X API 29 x86
- un dispositivo móvil real Xiaomi A1. Android 9
- el entorno para la simulación del equipamiento hardware de la solución construido en este sprint.

Retrospectiva del sprint

En esta tarea se realizó la revisión del propio proceso de ejecución del segundo sprint del proyecto:

¿Qué ha salido bien?	Se ha cumplido la planificación prevista y se ha alcanzado la funcionalidad esperada. Se han solucionado los problemas de rendimiento detectados en el sprint anterior.
¿Qué podría haber sido mejor?	Con mayor conocimiento del framework Flutter se habrían evitado muchos retrabajos y problemas en la construcción.
¿Qué cosas hay que probar?	Excepciones y situaciones anómalas. Pruebas completas en otros dispositivos para ver el encaje visual.
¿Qué problemas hay para escalar?	No detectados hasta la fecha.

Tabla 6. Retrospectiva del sprint 2

4.7 Revisión y actualización pila de producto después de sprint 2

Tras la ejecución del segundo ciclo de sprint del proyecto se procedió a actualizar el *product backlog* con la nueva situación.

Id	<i>Product backlog</i> ítem	Tipo
1	Recuperar personas en atención en un ámbito de red inalámbrica de área personal (WPAN)	Features
2	Consultar los indicadores asistenciales disponibles para las personas en atención	Features
3	Optimización funcionalidades ya existentes en la aplicación	Features
Release 3		

Tabla 7. Pila de producto después de sprint 2

4.8 Ciclo de sprint 3

Planificación del sprint

En esta etapa se generó el listado concreto de tareas a realizar en la fase de desarrollo del tercer y último *sprint* del proyecto. Como resultado, se construyó *el sprint backlog*.

Producto backlog ítem	Id	Tarea de <i>sprint</i>
Recuperar personas en atención en un ámbito de red inalámbrica de área personal (WPAN)	1	Programación placa Arduino: recepción petición y envío de información de personas
	2	Ampliación funcionalidad app: módulo de comunicación bluetooth – envío petición y recepción de información de personas
Consultar los indicadores asistenciales disponibles para las personas en atención	3	Añadir funcionalidad de consulta de indicadores a la pantalla de detalle de personas en atención
Optimización funcionalidades ya existentes en la aplicación	4	Limpieza, corrección y documentación de código

Tabla 8. Pila de sprint 3

Sprint

En esta etapa se ejecutaron las tareas planificadas en la etapa anterior. Como resultado, se ha obtenido la tercera release del producto, que supone alcanzar el 100% de la funcionalidad prevista.

Revisión del sprint

En esta tarea se realizó la revisión del resultado de sprint desarrollado en la etapa anterior, mediante la realización de pruebas de las funcionalidades implementadas. Se procede a realizar pruebas integradas con:

- el entorno de emulación de Android Studio: dispositivo Nexus 5X API 29 x86
- un dispositivo móvil real Xiaomi A1. Android 9
- el entorno para la simulación del equipamiento hardware de la solución.

5. Planificación

A continuación se incluye una tabla con la visión a alto nivel de las principales fases del proyecto:

	Duración fase	Fecha de inicio	Fecha de fin
Ideación y definición inicial	33 días	20/02/2020	23/03/2020
Ciclo de sprint 1	17 días	24/03/2020	09/04/2020
Ciclo de sprint 2	24 días	14/04/2020	07/05/2020
Ciclo de sprint 3	9 días	12/05/2020	20/05/2020
Documentación	16 días	21/05/2020	05/06/2020

Tabla 9. Planificación fases generales proyecto

Hitos principales del proyecto:

	Fecha de consecución
Aprobación del proyecto	25/02/2020
Entrega PEC 1: Propuesta formal del proyecto	06/03/2020
Entrega PEC 2: Mandato del proyecto y planificación	16/03/2020
<i>Product backlog</i> inicial	23/03/2020
Release 1 producto	09/04/2020
Entrega PEC 3: Entrega 1	13/04/2020
Release 2 producto	07/05/2020
Entrega PEC 4: Entrega 2	11/05/2020
Release final producto	20/05/2020
Memoria	05/06/2020
Presentación académica	05/06/2020
Presentación pública	05/06/2020
Entrega PEC 5: Cierre	05/06/2020

Tabla 10. Hitos principales proyecto

A continuación, se proporciona un diagrama de Gantt con la planificación global detallada del proyecto. Se ha utilizado un código de colores para agrupar las tareas correspondientes a cada PEC. En la leyenda del diagrama se especifica la correspondencia.

Diseño y desarrollo de una aplicación móvil para una solución de atención sociosanitaria remota
Máster Universitario en Aplicaciones Multimedia, Daniel Guzmán López

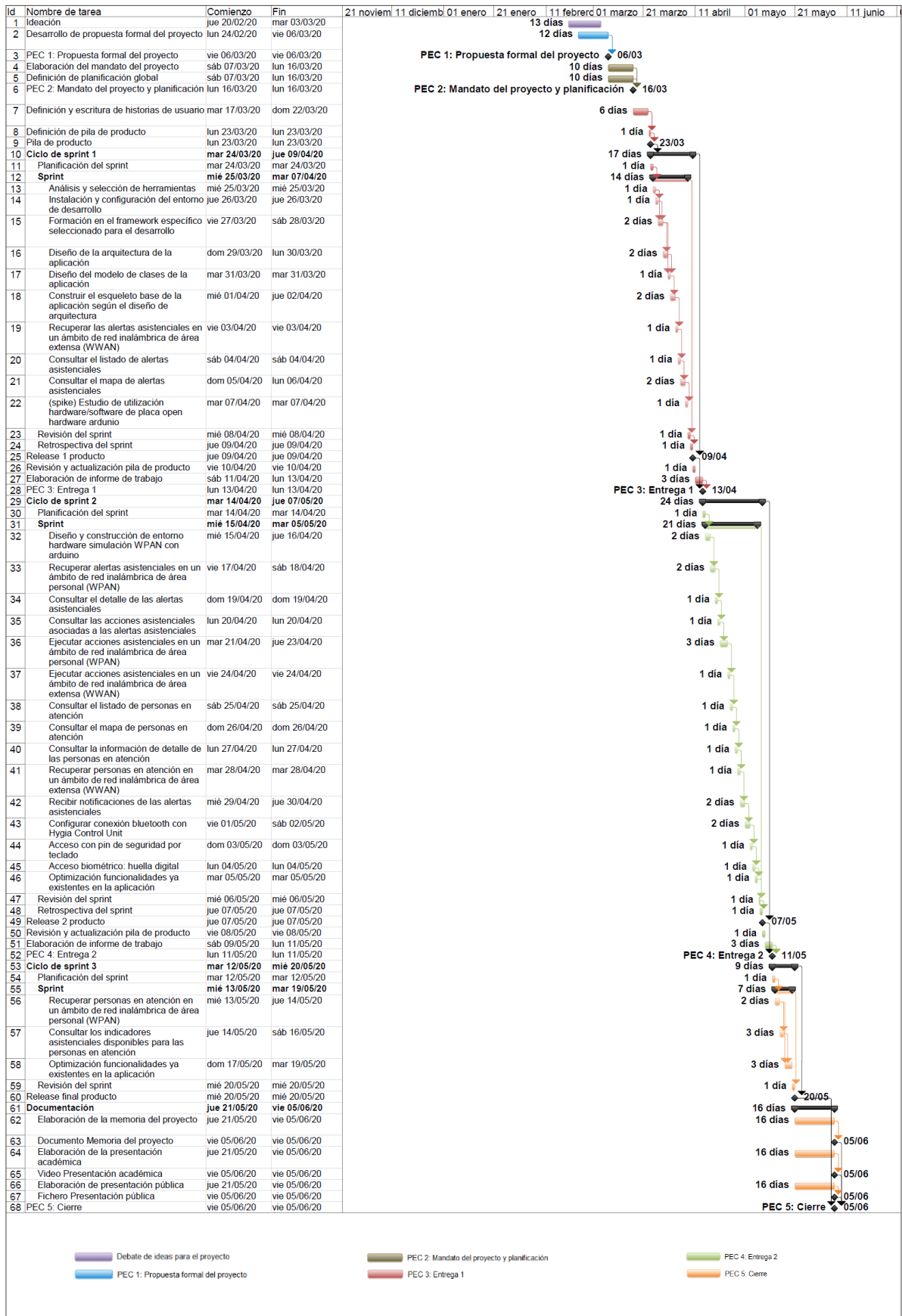


Diagrama 1. Gantt planificación global de proyecto

6. Presupuesto

Para abordar un proyecto de estas características, y en base al marco de trabajo *scrum* fijado, se estima un equipo (*Scrum Team*) con la siguiente composición:

- *Product Owner*: es la figura que representa la voz del cliente. Su función principal será la definición y gestión del *product backlog* durante todo el ciclo de vida del proyecto, así como, asegurar que el producto obtenido genera el valor esperado, es decir, maximizar el ROI (retorno de la inversión).
- *Scrum Master*: su función principal será velar por el correcto cumplimiento de los procesos que se hayan acordado en el marco de trabajo *scrum*.
- *Development Team*: equipo multidisciplinar y auto-organizado que se encargará de la ejecución de los ciclos de *sprint*. Todos los miembros del equipo se denominarán desarrolladores (según convención del marco de trabajo *scrum*), no obstante, se corresponden con todos los perfiles específicos necesarios para el desarrollo del proyecto.

Se estima, para la ejecución de este proyecto, una primera fase de definición inicial (o *sprint 0*) y, a continuación, la ejecución de tres ciclos de *sprint* de 2 semanas de duración cada uno:

		Recurso	Horas	Precio/Hora	Coste Total
Ideación y definición inicial (<i>sprint 0</i>)	Supervisión marco de trabajo <i>scrum</i>	Scrum Master	8	57 €/hora	456 €
	Concepción y diseño de solución y producto	<i>Product owner</i>	16	57 €/hora	912 €
	Definición y escritura de historias de usuario	<i>Product owner</i>	16	57 €/hora	912 €
	Construcción arquetipos y <i>wireframes</i> base iniciales	UX-Visual Digital Designer	16	42 €/hora	672 €
	Definición de <i>product backlog</i>	<i>Product owner</i>	8	57 €/hora	456 €

Tabla 11. Presupuesto detalle ideación y definición inicial

Resumen Ideación y definición inicial (<i>sprint 0</i>)	
Total horas	64
Total coste	3.408 €

Tabla 12. Presupuesto resumen ideación y definición inicial

		Recurso	Horas	Precio/Hora	Coste Total
Ciclo de sprint (2 semanas)	Supervisión marco de trabajo <i>scrum</i>	Scrum Master	16	57 €/hora	912 €
	Supervisión producto	<i>Product Owner</i>	16	57 €/hora	912 €
	Diseño de arquitectura	Desarrollador scrum: perfil Arquitecto Software	16	52 €/hora	832 €
	Experiencia de usuario	Desarrollador scrum: perfil UX-Visual Digital Designer	32	42 €/hora	1.344 €
	Desarrollo móvil (Flutter)	Desarrollador scrum: perfil desarrollo mobile	64	40 €/hora	2.560 €
	Programación y configuración hardware (Placa Arduino)	Desarrollador scrum: perfil ingeniero electrónico	32	40 €/hora	1.280 €

Tabla 13. Presupuesto detalle ciclo de sprint

Resumen Ciclo de sprint (2 semanas)	
Total horas	176
Total coste	7.840 €

Tabla 14. Presupuesto resumen ciclo de sprint

		Recurso	Coste
Recursos materiales, licencias	Entorno de simulación hardware	Placa ARDUINO UNO REV 3	19,90 €
		DSD Tech HC-05 módulo Bluetooth inalámbrico	12,33 €
		ELEGOO Kit de Componentes Electrónicos	10,99 €
		AZDelivery MB-102 Breadboard	5,99 €
		TOTAL	49,21 €

Tabla 15. Presupuesto recursos materiales, licencias

	Etapa	Horas totales	Coste Total
Total Proyecto	Ideación y definición inicial (<i>sprint</i> 0)	64	3.408 €
	Ciclo de <i>sprint</i> 1 (2 semanas)	176	7.840 €
	Ciclo de <i>sprint</i> 2 (2 semanas)	176	7.840 €
	Ciclo de <i>sprint</i> 3 (2 semanas)	176	7.840 €
	Recursos materiales, licencias	-	49,21 €
	TOTAL		592

Tabla 16. Presupuesto total proyecto

De cara a continuar con el desarrollo de la aplicación móvil, más allá de este proyecto, se tendrá en cuenta el coste unitario de ciclo de *sprint* (7.840 €). El número de *sprints* necesarios será variable y deberá establecerse en relación a la funcionalidad que se considere necesario añadir a la aplicación. Esta cuestión deberá ser discutida con el *Product Owner*.

7. Estructura del resto del documento

A continuación se describe la estructura y contenido del resto de capítulos de este documento:

- Capítulo 2: Análisis:
 - En este capítulo se aporta, en primer lugar, información acerca del estado del arte de los servicios y soluciones de mercado de atención sociosanitaria remota, así como, del ecosistema actual de *frameworks* para el desarrollo de aplicaciones móviles. Estos dos aspectos son fundamentales para el desarrollo del proyecto y, entender su situación actual, es necesario para poder orientar y construir el producto de forma adecuada. Por último, se ofrece una discusión acerca de los objetivos de contribución del proyecto al estado del arte descrito.
 - Se incluye la definición del producto y sus especificaciones a través del conjunto de épicas e historias de usuario generadas en el marco agile elegido para el proyecto.
- Capítulo 3: Diseño
 - Se describe la arquitectura definida para la solución y la aplicación móvil, aportando diagramas de contexto de los distintos ámbitos de utilización planteados.
 - Se define el entorno para la simulación del equipamiento hardware de la solución con el que interactuará de forma bidireccional la aplicación móvil.
 - Se describe el proceso de trabajo seguido en el aspecto gráfico y de experiencia de usuario de la aplicación. Se aportan los subproductos producidos, incluyendo los arquetipos de usuario definidos, *wireframes* y componentes diseñados/utilizados.
 - Se incluye información del framework y de los lenguajes y componentes de desarrollo utilizados tanto para la aplicación móvil (Flutter) como para el entorno de simulación hardware (Arduino), incluyendo las correspondientes justificaciones a las decisiones tomadas.
- Capítulo 4: Implementación
 - Se proporciona información de detalles de implementación de la aplicación móvil y del entorno de simulación hardware, incluyendo estructura de los proyectos, componentes utilizados y aspectos de interés en relación a las decisiones tomadas.
- Capítulo 5: Evidencias de prueba y demo
 - Sobre el esquema base de las especificaciones del producto construido (historias de usuario) se realiza un recorrido completo de todas las funcionalidades de la aplicación, explicando en detalle cada interfaz y comportamiento. Se aportan evidencias de las pruebas realizadas, incluyendo capturas reales de todas las pantallas de la aplicación.
- Capítulo 6: Conclusiones y líneas de futuro
 - Se realiza una revisión crítica y detallada de la consecución de los objetivos planteados en el proyecto, de la adecuación de la metodología empleada y las conclusiones obtenidas.
 - Se perfilan posibles líneas de futuro para la continuación del desarrollo del producto y solución.

Capítulo 2: Análisis

1. Estado del arte

1.1. Servicios y soluciones de atención sociosanitaria remota

Se observa, en el mercado de servicios de atención sociosanitaria remota, frecuentes marcos de colaboración entre operadoras de telecomunicaciones y empresas del sector de atención sociosanitaria. Se propone como referencia dos ejemplos:

- Orange con la compañía ILUNION Sociosanitario, del grupo social ONCE: el 20 de marzo de 2018, Orange España y la compañía ILUNION Sociosanitario lanzaron el servicio “Serena” [20] para la atención remota de personas con necesidades asistenciales. Se trata de un servicio de teleasistencia en movilidad proporcionado a través del teléfono móvil de las personas usuarias. La atención de los profesionales en este caso se centraliza en una sala con teleoperadores especializados, a través de una aplicación desktop. Es decir, el concepto de movilidad se aplica al lado del usuario del servicio y no a los profesionales que realizan la atención. Como complemento se dispone de un portal web de seguimiento para familiares y personas autorizadas.
- Telefónica con la compañía británica Tunstall Healthcare: El 23 de Julio de 2019, Telefónica Empresas y la compañía Tunstall Healthcare firmaron un acuerdo para el desarrollo de servicios y soluciones de telemonitorización y atención remota de pacientes en España [30]. Se trata de una solución alojada en la nube y que se comercializa en modalidad de pago por uso.

En este caso, Telefónica demuestra un interés continuado en el tiempo por el mercado de la gestión remota de pacientes. En 2013 adquirió la compañía brasileña Axismed [22], dedicada a la gestión de pacientes crónicos en Latinoamérica. En España, Telefónica cuenta en la actualidad con una línea de negocio para grandes empresas dedicada a telehealth (gestión remota de pacientes), telecare (teleasistencia móvil) y health IT (gestión de la demana e imagen médica digital).

Por su parte, Tunstall Healthcare es una multinacional dedicada a la salud digital y atención conectada, líder del sector (5 millones de usuarios en 51 países) [30], que ya aportaba una experiencia dilatada en atención remota, con varias plataformas tecnológicas en el mercado y un completo catálogo de equipamiento hardware multipropósito.

Se observa una fuerte componente tecnológica y de innovación en las diferentes compañías que integran el mercado de servicios y soluciones de atención sociosanitaria remota. Se identifican algunas de las compañías más relevantes: GrandCare Systems, RCare, NRS Healthcare, la ya citada Tunstall Healthcare o, entre otras, Home Care Lab, la unidad I+D+i del Grupo SSI homologada por la Red Vasca de Ciencia, Tecnología e Innovación de Euskadi.

Las soluciones de mercado suelen componerse a nivel general de:

- un equipamiento hardware: dispositivos de múltiples propósitos para recoger información y prestar distintas funcionalidades al usuario del servicio: pulsadores de alarma, intercomunicadores, sensores (de presencia, de ocupación de cama, de temperatura, de epilepsia, de humedad, etc.), dispositivos wearables para lectura de ubicación y constantes vitales, dispensadores de medicamentos, etc.
- una plataforma software: incluyendo aplicaciones de gestión y atención para el personal sociosanitario, aplicaciones para el usuario final del servicio y aplicaciones de seguimiento para familiares, mediante arquitecturas on-premise y SaaS (Software as a Service) en aplicaciones web y desktop.
- comunicaciones: el conjunto de líneas y protocolos de comunicación necesarios para lograr la interoperabilidad de la solución entre los diferentes dispositivos y aplicaciones.

Actualmente, debido al desarrollo creciente del campo de la inteligencia artificial y el 5G, las soluciones tienden a adoptar una arquitectura similar a la propuesta en MavHome [8] (capa de decisión, capa de información, capa de comunicación, capa física), potenciando la adquisición masiva de información y su análisis para la toma de decisiones, de cara a ofrecer servicios predictivos [4].

Se observa que Tunstall Healthcare presenta algunas de las soluciones más completas y avanzadas del mercado, tanto en soluciones completas como en equipamiento hardware:

- DMP - Device Management Platform: servicio en la nube para la gestión de sus terminales hardware Smart Hub.
- ICP - Integrated Care Platform: suite de aplicaciones software con distintos objetivos, incluyendo apps para el usuario final del servicio como myMobile®.
- PNC8.3: Plataforma para monitorización y gestión de llamadas.
- TSP - Tunstall Service Platform: suite SaaS (Software as a Service) con una arquitectura basada en microservicios.

1.2. Tipos de aplicaciones para dispositivos móviles

El mercado general actual de aplicaciones para dispositivos móviles puede organizarse, atendiendo a principios tecnológicos de desarrollo, según la siguiente clasificación:

- Aplicaciones web (Web applications o Web apps)
- Aplicaciones móviles nativas (Native applications o Native apps)
- Aplicaciones móviles híbridas (Cross-platform o Hybrid applications o Hybrid apps)

Aplicaciones web (Web applications o Web apps)

Son aplicaciones que se ejecutan a través del navegador web del sistema operativo correspondiente, por tanto, se trata de aplicaciones multiplataforma. Además, no son exclusivas para dispositivos móviles. En función del tipo y características del dispositivo, el interfaz de la aplicación podrá adaptarse

visualmente (comportamiento responsive), pero la aplicación es siempre la misma, es decir, requiere una única línea de desarrollo. Por tanto, se les considera también como aplicaciones multidispositivo.

Se estructuran, de forma general, en una arquitectura de tres capas que emplea tecnologías específicas para cada una de ellas:

- Cliente (navegador web): HTML5 + JavaScript + CSS3.
- Servidor: Ruby (Ruby on Rails), Python (Django, Flask, Pylons), PHP (Laravel), Java (Spring), Scala (Play), entre otras.
- Persistencia (Sistemas Gestores de Base de Datos): MySQL, PostgreSQL, MongoDB, entre otras.

Este tipo de aplicaciones presentan ciertas limitaciones en cuanto a rendimiento y aprovechamiento de muchas de las características específicas de cada dispositivo. Un ejemplo sería la imposibilidad de utilización de tecnología push. Para superar muchas de estas limitaciones, surgen las denominadas aplicaciones web progresivas (Progressive Web Apps o PWA).

Las PWA constituyen actualmente un mercado en auge [29], potenciado principalmente por la compañía Google. Este tipo de aplicaciones se ejecutan en el navegador web como el resto de las aplicaciones web (requiere compatibilidad) pero, en este caso, consiguen ofrecer una experiencia de usuario similar a las aplicaciones nativas. Para ello, utilizan una arquitectura basada en la utilización de componentes service workers para poder manejar eventos, que emplean un sistema de caché mejorado de forma notable el rendimiento en la ejecución.

Aplicaciones móviles nativas (Native applications o Native apps)

Son aplicaciones construidas de forma específica para una plataforma o sistema operativo concreto (siendo las plataformas Android e iOS las que se reparten actualmente la práctica totalidad del mercado [15]). Por tanto, se requieren distintas líneas de desarrollo por cada plataforma, algo que aumenta el coste y el time-to-market.

Este tipo de aplicaciones se construyen con los lenguajes nativos correspondientes a cada plataforma: Java para Android, y Objective-C en el caso de iOS. De esta forma, puede aprovecharse el porcentaje total de las capacidades de cada dispositivo y lograr así la mayor funcionalidad y el mejor rendimiento posible.

Aplicaciones móviles híbridas (Cross-platform o Hybrid applications o Hybrid apps)

Se trata de aplicaciones construidas con tecnologías web (HTML5 + JavaScript + CSS3) pero encapsuladas dentro un contenedor nativo de la plataforma correspondiente. El núcleo (HTML, JavaScript, CSS3) de la aplicación es un producto multiplataforma que requiere una única línea de desarrollo (el contenedor nativo sí es específico para cada plataforma). Presentan menor rendimiento

que las aplicaciones nativas, pues no están desarrolladas en el lenguaje nativo y requieren un intérprete para su ejecución, sin embargo, reducen los costes de desarrollo y el time-to-market.

La construcción de estas aplicaciones se realiza habitualmente mediante marcos de trabajo (*frameworks*), entre los que destacan actualmente: Ionic, PhoneGAP, Angular, jQuery mobile, React Native, Framework 7 o Native Script.

1.3. Marcos de trabajo (*frameworks*) para crear aplicaciones móviles

Uno de los elementos considerado como más relevante en el desarrollo de aplicaciones móviles es el denominado marco de trabajo o framework. El framework proporciona al desarrollador la base para construir la aplicación, proporcionando la abstracción necesaria sobre la plataforma, reduciendo tiempo, riesgos y asegurando una arquitectura consistente entre aplicaciones [11].

Los frameworks son clave para lograr una buena productividad en el ciclo de desarrollo y lograr un buen *time-to-market* [24], por ello, la elección de uno u otro se convierte en una de las decisiones más importantes a la hora de comenzar un proyecto. Deberá seleccionarse el adecuado para las necesidades del proyecto y el tipo de aplicación.

A continuación, se presenta el estudio de las distintas opciones de mercado:

Flutter

Flutter es un SDK (Software Development Kit o Kit de Desarrollo de Software) de código abierto creado por Google para el desarrollo de aplicaciones móviles [19]. Desde la versión 1.5 anunciada por Google en el Google I/O 2019, el mismo código base se puede utilizar en móvil, web, embebido y desktop, es decir, se convierte en un framework multiplataforma y multidispositivo [9].

Flutter utiliza el lenguaje Dart (creado por Google) y un motor escrito de C++ que utiliza Skia (Skia Graphics Engine, una librería de código abierto escrita en C++) para gráficos. Se trata de una de las grandes apuestas de Google que, de hecho, está utilizando en la construcción de su nuevo sistema operativo de tiempo real (Real-time operating systems o RTOS) Fuchsia [27]. Ejemplos de aplicaciones construidas (parcialmente) con Flutter: Google Ads, Alibaba o ebay.

React native

Se trata de un framework de código abierto de enfoque híbrido, creado y evolucionado por Facebook. Utiliza JavaScript (optimizado con el motor de ejecución específico Hermes presentado recientemente [23]) y la librería React, pero también puede combinar componentes en lenguajes nativos Objective-C, Java, o Swift. El resultado es una aplicación prácticamente indistinguible de un producto nativo, combinado con las ventajas de la aproximación multiplataforma.

Ionic

Un framework de código abierto enfocado al desarrollo de aplicaciones híbridas muy bien posicionado para el nuevo mercado de aplicaciones progresivas [6]. Su base está construida en AngularJs y Cordova, Permite utilizar distintos frameworks para el desarrollo de la capa front-end, entre los que destaca el marco de código abierto Vue.js. La aplicación web se empaqueta para el acceso a las funciones nativas de cada plataforma con el contenedor Capacitor [21].

Native script

De la compañía búlgara Telerik, permite generar aplicaciones híbridas utilizando distintos frameworks de front-end como Angular o Vue.js pero utilizando las mismas APIs que se emplean en desarrollos nativos.

Framework7

Un framework de código abierto que permite crear aplicaciones móviles híbridas enfocado principalmente a la plataforma iOS.

PhoneGap

Framework de Adobe Systems enfocado al desarrollo de aplicaciones híbridas. El framework Apache Cordova es su versión de código abierto. Se basa en tecnologías web (HTML5 + JavaScript + CSS3) que se empaquetan para desplegarse después en los sistemas nativos.

Corona SDK

Es un framework multiplataforma que permite generar una única línea de código base para Android e iOS. Proporciona un lenguaje de scripting, Lua, enfocado a la velocidad y simplicidad.

JQuery Mobile

Se trata de un framework basado en JavaScript que permite crear aplicaciones web responsive. Recientemente está perdiendo mucha cuota de mercado y se está viendo remplazado por otros frameworks más completos [25].

Xamarin

Es un framework de código abierto multiplataforma de Microsoft basado en .NET y C#. Su arquitectura se basa en la construcción de interfaces nativas independientes para cada plataforma sobre un núcleo multiplataforma escrito en C# que concentra la lógica de negocio [14]. Xamarin presenta también interoperabilidad con lenguajes nativos Objective-C, Java, C y C++.

1.4. Discusión del estado del arte para los objetivos del proyecto

Las colaboraciones y proyectos conjuntos entre grandes corporaciones del mundo de las telecomunicaciones y otras empresas tecnológicas con compañías del sector sociosanitario ponen de manifiesto la importancia de las comunicaciones, la seguridad, la alta disponibilidad y el exigente tiempo de respuesta demandado en los servicios de atención sociosanitaria remota. Se observa en este mercado que las soluciones disponibles presentan un importante grado de desarrollo tecnológico en muchos aspectos, como el equipamiento hardware, los servicios interoperables con múltiples protocolos, la integración a sistemas de terceros a través de REST API, integración con Inteligencia Artificial y Big Data, seguridad o comunicaciones. En estos aspectos, este trabajo se moverá dentro del estado del arte actual.

La contribución que pretende realizar este proyecto a este tipo de soluciones se centrará en las aplicaciones móviles que utilizan los profesionales para prestar la atención a las personas con necesidades asistenciales, es decir, aquellas aplicaciones que reciben la información de monitorización en tiempo real y desde las cuales se actúa sobre el equipamiento hardware en remoto. Por tanto, aplicaciones en las que el rendimiento y tiempo de respuesta son requisitos de vital importancia. En este sentido, se pretende ir más allá del estado del arte y aportar valor:

- aplicando las últimas tecnologías de mercado para construir una aplicación que dé respuesta a las cada vez más exigentes necesidades de interacción con los dispositivos: cada vez más dispositivos conectados, enviando mayor cantidad de información, con más frecuencia y con mayor rapidez (debido al desarrollo emergente del 5G y Internet of Things, IoT). Se pretende construir una aplicación de alto rendimiento, orientada a los emergentes sistemas operativos de tiempo real (Real-time operating systems o RTOS) para dispositivos móviles, como, por ejemplo, el próximo Fuchsia OS de Google [26].
- construyendo una aplicación versátil que permita adaptarse a distintas tecnologías de adquisición de datos para cubrir servicios de distinta naturaleza. El objetivo es disponer de una aplicación flexible y genérica que permita distintos ámbitos de utilización según la solución tecnológica más adecuada, ya sea una solución con requisitos complejos de infraestructura y conectividad o soluciones más simples y de recursos limitados.
- produciendo una aplicación sobre una base de código única para su evolución multidispositivo y multiplataforma.

2. Definición de especificaciones del producto

2.1 Épicas e historias de usuario

Atendiendo a la definición de alcance establecida para nuestro producto en el proyecto, se decide priorizar aquellas funcionalidades de la aplicación relacionadas con la interacción del usuario (asistente) con las alertas asistenciales y personas en atención. Se pondrá foco especialmente en la interacción de la aplicación con los dispositivos hardware (sensores, actuadores). En este contexto, se decide bordar cinco épicas que nos permitirán trabajar las funcionalidades descritas:

- Consultar alertas asistenciales
- Consultar personas en atención
- Ejecutar acciones asistenciales remotas
- Acceder a la aplicación
- Configurar la aplicación

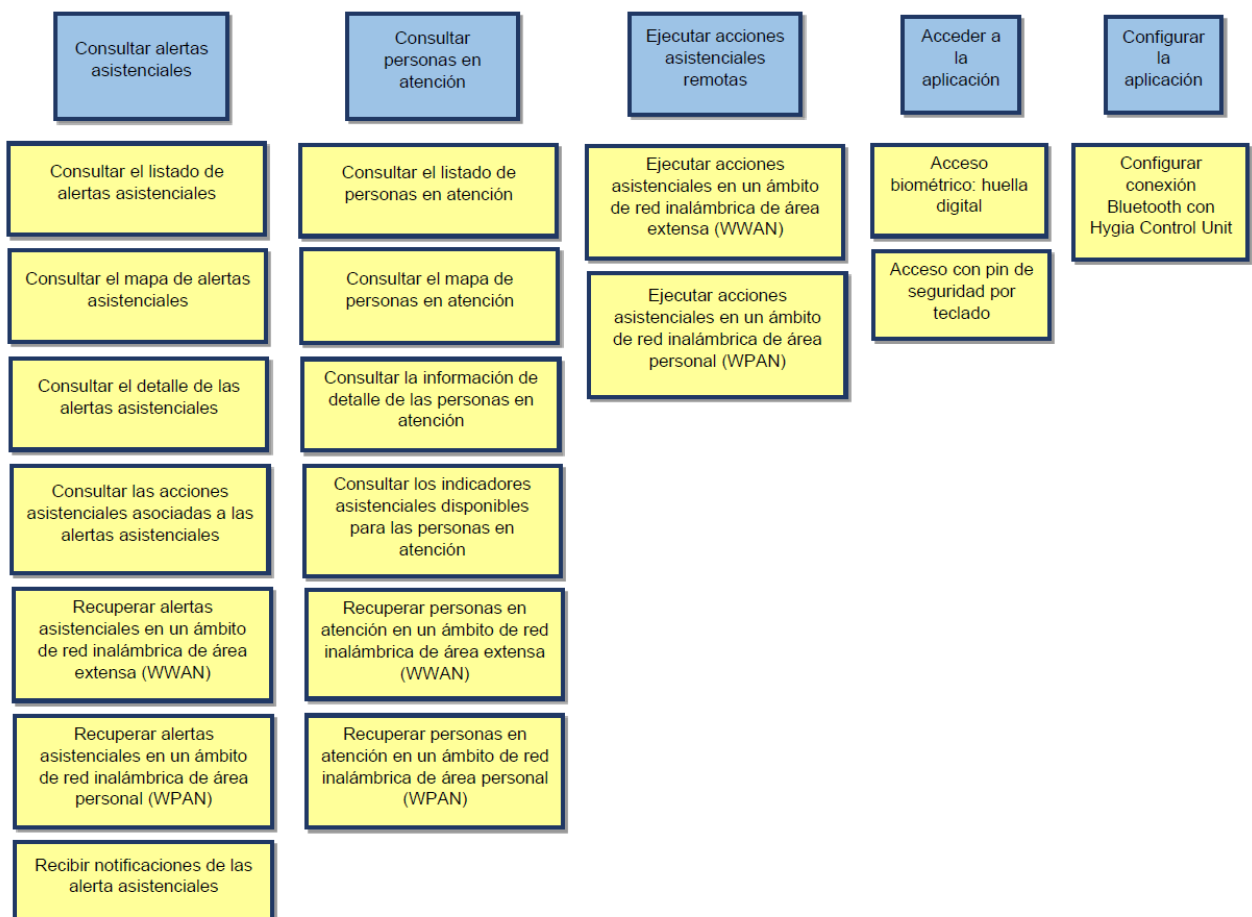


Ilustración 1. Épicas e Historias de usuario

2.2 Épica 1: Consultar alertas asistenciales

Nombre	Consultar alertas asistenciales	
Cómo	Criterios de aceptación	
Asistente	- <Dado> un Asistente <Cuando> abre la aplicación <Entonces> se muestran las alertas asistenciales de sus personas en atención. - <Dado> un Asistente <Cuando> no hay alertas asistenciales de sus personas en atención <Entonces> se muestra el mensaje “No hay alertas”.	
Quiero		
consultar las alertas asistenciales de mis personas en atención		
Para	conocer si hay alguna alerta que requiera atención por mi parte en el servicio de atención sociosanitaria remota	

Historias de usuario

Nombre	Consultar el listado de alertas asistenciales	Estimación	5
Épica	Consultar alertas asistenciales		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente <Cuando> abre la aplicación <Entonces> se muestra por defecto el listado de alertas asistenciales. - <Dado> un Asistente en el menú <Cuando> pulsa la opción “Alertas asistenciales” <Entonces> se muestra el listado de alertas asistenciales. - <Dado> un Asistente en el listado de alertas asistenciales <Cuando> no hay alertas asistenciales <Entonces> se muestra el mensaje “No hay alertas”. - <Dado> un Asistente en el listado de alertas asistenciales <Cuando> no pueden recuperarse las alertas <Entonces> se muestra el mensaje “No pueden recuperarse las alertas asistenciales”.		
Quiero			
consultar el listado de alertas asistenciales de mis personas en atención			
Para	conocer si hay alguna alerta que requiera atención por mi parte		

Nombre	Consultar el mapa de alertas asistenciales	Estimación	13
Épica	Consultar alertas asistenciales		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente en la pantalla de “Alertas asistenciales” <Cuando> pulsa la opción de “Mapa” <Entonces> se muestra el mapa de alertas asistenciales. - <Dado> un Asistente en el mapa de alertas asistenciales <Cuando> hay alertas asistenciales <Entonces> se muestran marcadas en el mapa en su ubicación correspondiente. - <Dado> un Asistente en el mapa de alertas asistenciales <Cuando> no hay alertas asistenciales <Entonces> no se muestra ninguna alerta marcada en el mapa. - <Dado> un Asistente en el mapa de alertas asistenciales <Cuando> no pueden recuperarse las alertas <Entonces> se muestra el mensaje “No pueden recuperarse las alertas asistenciales”.		
Quiero			
consultar el mapa de alertas que requieren mi atención			
Para	conocer la disposición geográfica de las alertas que requieren atención por mi parte		

Nombre	Recuperar alertas asistenciales en un ámbito de red inalámbrica de área extensa (WWAN)	Estimación	3
Épica	Consultar alertas asistenciales		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente <Cuando> accede a la pantalla de alertas asistenciales <Entonces> se muestra el conjunto de alertas asistenciales recuperadas en un ámbito de red inalámbrica de área extensa. - <Dado> un Asistente en la pantalla de alertas asistenciales <Cuando> no se recuperan alertas en el ámbito de red inalámbrica de área extensa <Entonces> no se muestra ninguna alerta. - <Dado> un Asistente en la pantalla de alertas asistenciales <Cuando> no pueden recuperarse las alertas en el ámbito de red inalámbrica de área extensa <Entonces> se muestra el mensaje “No pueden recuperarse las alertas asistenciales”.		
Quiero			
recuperar las alertas asistenciales en un ámbito de red inalámbrica de área extensa			
Para	poder prestar el servicio de atención sociosanitario en un ámbito geográfico amplio		

Nombre	Recuperar alertas asistenciales en un ámbito de red inalámbrica de área personal (WPAN)	Estimación	21
Épica	Consultar alertas asistenciales		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente <Cuando> accede a la pantalla de alertas asistenciales <Entonces> se muestra el conjunto de alertas asistenciales recuperadas en un ámbito de red inalámbrica de área personal. - <Dado> un Asistente en la pantalla de alertas asistenciales <Cuando> no se recuperan alertas en el ámbito de red inalámbrica de área personal <Entonces> no se muestra ninguna alerta. - <Dado> un Asistente en la pantalla de alertas asistenciales <Cuando> no pueden recuperarse las alertas en el ámbito de red inalámbrica de área personal <Entonces> se muestra el mensaje “No pueden recuperarse las alertas asistenciales”.		
Quiero			
recuperar las alertas asistenciales en un ámbito de red inalámbrica de área personal			
Para	poder prestar el servicio de atención sociosanitario en ámbitos sin acceso a redes inalámbricas de área extensa		

Nombre	Consultar el detalle de las alertas asistenciales	Estimación	5
Épica	Consultar alertas asistenciales		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente en el listado de alertas asistenciales <Cuando> pulsa sobre una de las alertas del listado <Entonces> se abre una pantalla con la información de detalle de la alerta. - <Dado> un Asistente en el mapa de alertas asistenciales <Cuando> pulsa sobre una de las alertas del mapa <Entonces> se abre una pantalla con la información de detalle de la alerta. - <Dado> un Asistente en la pantalla de información de detalle de la alerta <Cuando> no puede recuperarse la información de detalle de la alerta		
Quiero			
consultar la información de detalle de una alerta asistencial			
Para	conocer la información completa de la situación que ha provocado la alerta asistencial		

	<Entonces> se muestra el mensaje “No ha sido posible recuperar la información de la alerta”.
--	--

Nombre	Consultar las acciones asistenciales asociadas a las alertas asistenciales	Estimación	5
Épica	Consultar alertas asistenciales		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente en la pantalla de detalle de la alerta asistencial <Cuando> pulsa sobre la opción de acciones asistenciales <Entonces> se muestran las acciones disponibles para la alerta asistencial. - <Dado> un Asistente en la pantalla de detalle de la alerta asistencial <Cuando> no hay acciones asistenciales asociadas a la alerta <Entonces> se muestra el mensaje “No hay acciones asociadas”.		
Quiero			
consultar las acciones disponibles para una alerta asistencial			
Para			
decidir qué acción debo ejecutar para solucionar la alerta asistencial			

Nombre	Recibir notificaciones de las alertas asistenciales	Estimación	8
Épica	Consultar alertas asistenciales		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente con un dispositivo móvil, <Cuando> se produce una alerta asistencial <Entonces> recibe una notificación de dicha alerta. - <Dado> un Asistente con una notificación de alerta asistencial, <Cuando> pulsa en la notificación recibida <Entonces> se muestra el detalle de la alerta asistencial.		
Quiero			
recibir notificaciones de las alertas asistenciales			
Para			
poder asistir con rapidez a las personas en atención			

2.3 Épica 2: Consultar personas en atención

Título	Consultar personas en atención		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente en la aplicación <Cuando> accede a la opción “Personas en atención” <Entonces> se muestra el conjunto de las personas que tiene en atención. - <Dado> un Asistente <Cuando> no tiene personas en atención <Entonces> se muestra el mensaje “No tiene personas en atención”.		
Quiero			
consultar las personas en atención			
Para			
acceder a la información de las personas a las que estoy presentando el servicio de asistencia sociosanitaria remota.			

Historias de usuario

Nombre	Consultar el listado de personas en atención	Estimación	5
Épica	Consultar personas en atención		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente en el menú <Cuando> pulsa la opción “Personas en atención” <Entonces> se muestra el listado de personas en atención. - <Dado> un Asistente en la pantalla de “Personas en atención” <Cuando> no tiene personas asignadas <Entonces> se muestra el mensaje “No tiene personas en atención”. - <Dado> un Asistente en la pantalla de “Personas en atención” <Cuando> no es posible recuperar las personas <Entonces> se muestra el mensaje “No ha sido posible recuperar las personas en atención”.		
Quiero			
consultar el listado de personas en atención			
Para	conocer el conjunto de personas a las que estoy presentando el servicio de asistencia sociosanitaria remota.		

Nombre	Consultar el mapa de personas en atención	Estimación	13
Épica	Consultar personas en atención		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente en la pantalla de “Personas en atención” <Cuando> pulsa la opción de “Mapa” <Entonces> se muestra el mapa de personas en atención. - <Dado> un Asistente en el mapa de personas en atención <Cuando> tiene personas asignadas <Entonces> se muestran marcadas en el mapa en su ubicación correspondiente. - <Dado> un Asistente en el mapa de personas en atención <Cuando> no tiene personas asignadas <Entonces> no se muestra ninguna persona marcada en el mapa. - <Dado> un Asistente en el mapa de personas en atención <Cuando> no pueden recuperarse sus personas asignadas <Entonces> se muestra el mensaje “No pueden recuperarse las personas en atención”.		
Quiero			
consultar el mapa de alertas que requieren mi atención			
Para	conocer la disposición geográfica de las personas a las que estoy presentando el servicio de asistencia sociosanitaria remota.		

Nombre	Consultar la información de detalle de las personas en atención	Estimación	5
Épica	Consultar personas en atención		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente en el listado de personas en atención <Cuando> pulsa sobre una de las personas del listado <Entonces> se abre una pantalla con la información de detalle de la persona. - <Dado> un Asistente en el mapa de personas en atención <Cuando> pulsa sobre una de las personas del mapa <Entonces> se abre una pantalla con la información de detalle de la persona. - <Dado> un Asistente en la pantalla de información de detalle de la persona <Cuando> no puede		
Quiero			
consultar la información de detalle de una persona en atención			
Para	conocer la información completa de la persona a la que estoy prestando el servicio de asistencia sociosanitaria remota.		

	recuperarse la información de detalle de la persona <Entonces> se muestra el mensaje “No ha sido posible recuperar la información de la persona”.
--	--

Nombre	Consultar los indicadores asistenciales disponibles para las personas en atención	Estimación	5
Épica	Consultar personas en atención		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente en la pantalla de detalle de la persona en atención, <Cuando> pulsa sobre la opción de indicadores asistenciales <Entonces> se muestra la pantalla de indicadores disponibles para la persona en atención. - <Dado> un Asistente en la pantalla de detalle de la persona en atención <Cuando> no hay indicadores asistenciales definidos para la persona <Entonces> no se muestra ningún indicador. - <Dado> un Asistente en la pantalla de detalle de la persona en atención <Cuando> no se han podido recuperar los indicadores asistenciales definidos para la persona <Entonces> se muestra el mensaje “No ha sido posible recuperar los indicadores asistenciales”.		
Quiero			
consultar los indicadores asistenciales disponibles para una persona en atención			
Para			
conocer el estado de la persona en atención a través de los distintos indicadores definidos			

Nombre	Recuperar personas en atención en un ámbito de red inalámbrica de área extensa (WWAN)	Estimación	3
Épica	Consultar alertas asistenciales		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente <Cuando> accede a la pantalla de personas en atención <Entonces> se muestra el conjunto de personas recuperadas del ámbito de red inalámbrica de área extensa. - <Dado> un Asistente en la pantalla de personas en atención <Cuando> no se recuperan personas en el ámbito de red inalámbrica de área extensa <Entonces> no se muestra ninguna persona. - <Dado> un Asistente en la pantalla de personas en atención <Cuando> no pueden recuperarse las personas en el ámbito de red inalámbrica de área extensa <Entonces> se muestra el mensaje “No pueden recuperarse las personas en atención”.		
Quiero			
recuperar el conjunto de personas en atención en un ámbito de red inalámbrica de área extensa			
Para			
poder prestar el servicio de atención sociosanitario en un ámbito geográfico amplio			

Nombre	Recuperar el estado de los indicadores asistenciales asociados a la persona en atención en un ámbito de red de área personal (WPAN)	Estimación	21
Épica	Consultar personas en atención		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente <Cuando> accede a la pantalla de detalle de personas en atención <Entonces> se muestra el conjunto de indicadores asistenciales recuperados de la red de área personal. - <Dado> un Asistente en la pantalla de detalle de personas en atención <Cuando> la red de área personal no ha devuelto indicadores asociados a la persona <Entonces> no se muestra ningún indicador. - <Dado> un Asistente en la pantalla de indicadores asistenciales <Cuando> no pueden recuperarse los indicadores de la red de área personal <Entonces> se muestra el mensaje “No ha sido posible recuperar los indicadores asistenciales”.		
Quiero			
recuperar el conjunto de personas en atención en un ámbito de red inalámbrica de área personal			
Para			
para poder prestar el servicio de atención sociosanitario en ámbitos sin acceso a redes inalámbricas de área extensa			

2.4 Épica 3: Ejecutar acciones asistenciales remotas

Nombre	Ejecutar acciones asistenciales remotas		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente en la aplicación <Cuando> ejecuta una acción asistencial <Entonces> se actúa de forma remota sobre la persona en atención.		
Quiero			
ejecutar acciones asistenciales remotas			
Para			
para poder atender a distancia las necesidades de las personas en atención			

Historias de usuario

Nombre	Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área extensa (WWAN)	Estimación	3
Épica	Ejecutar acciones asistenciales remotas		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente en la pantalla de acciones asistenciales <Cuando> pulsa “ejecutar” sobre la acción asistencial <Entonces> se ejecuta la acción en remoto sobre la persona en atención a través de la red inalámbrica de área extensa. - <Dado> un Asistente pulsando la opción “ejecutar” de una acción <Cuando> la acción se ejecuta correctamente <Entonces> se muestra el mensaje “La acción se ha completado con éxito”. - <Dado> un Asistente pulsando la opción “ejecutar” de una acción <Cuando> la acción no se ha ejecutado correctamente <Entonces> se muestra el mensaje “La acción no ha podido completarse”.		
Quiero			
ejecutar una acción asistencial remota en un ámbito de red inalámbrica de área extensa			
Para			
para poder asistir de forma remota y en tiempo real sobre personas distribuidas en un ámbito geográfico extenso			

Nombre	Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área personal (WPAN)	Estimación	21
Épica	Ejecutar acciones asistenciales remotas		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente en la pantalla de acciones asistenciales <Cuando> pulsa “ejecutar” sobre la acción asistencial <Entonces> se ejecuta la acción en remoto sobre la persona en atención a través de la red inalámbrica de área extensa. - <Dado> un Asistente pulsando la opción “ejecutar” de una acción <Cuando> la acción se ejecuta correctamente <Entonces> se muestra el mensaje “La acción se ha completado con éxito”. - <Dado> un Asistente pulsando la opción “ejecutar” de una acción <Cuando> la acción no se ha ejecutado correctamente <Entonces> se muestra el mensaje “La acción no ha podido completarse”.		
Quiero			
ejecutar una acción asistencial remota en un ámbito de red inalámbrica de área personal			
Para			
para poder asistir de forma remota y en tiempo real sobre personas distribuidas en ámbitos sin acceso a redes inalámbricas de área extensa			

2.5 Épica 4: Acceder a la aplicación

Épica	Acceder a la aplicación		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente <Cuando> abre la aplicación <Entonces> se muestra la pantalla de acceso. - <Dado> un Asistente en la pantalla de acceso <Cuando> introduce datos de acceso incorrectos <Entonces> se muestra mensaje de error de acceso. - <Dado> un Asistente en la pantalla de acceso <Cuando> introduce datos de acceso correctos <Entonces> accede a la aplicación.		
Quiero			
acceder a la aplicación			
Para			
hacer uso de las funcionalidades de la aplicación			

Historias de usuario

Nombre	Acceso biométrico: huella digital	Estimación	2
Épica	Acceder a la aplicación		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente en la pantalla de acceso de la aplicación <Cuando> introduce una huella digital registrada en el dispositivo <Entonces> accede a la aplicación. - <Dado> un Asistente en la pantalla de acceso de la aplicación <Cuando> introduce una huella digital no registrada en el dispositivo <Entonces> se muestra icono de error.		
Quiero			
acceder a la aplicación con mi huella digital			
Para			
hacer uso de las funcionalidades de la aplicación			

Nombre	Acceso con pin de seguridad por teclado	Estimación	2
Épica	Acceder a la aplicación		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente en la pantalla de acceso de la aplicación <Cuando> introduce por teclado un pin de seguridad correcto <Entonces> accede a la aplicación. - <Dado> un Asistente en la pantalla de acceso de la aplicación <Cuando> introduce un pin de seguridad incorrecto por teclado <Entonces> se muestra el mensaje "Error de acceso".		
Quiero			
acceder a la aplicación con mi pin de seguridad			
Para			
hacer uso de las funcionalidades de la aplicación			

2.6 Épica 5: Configurar la aplicación

Épica	Configurar la aplicación		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente en la aplicación <Cuando> accede a la opción de configuración <Entonces> se muestra la pantalla de configuración de la aplicación. - <Dado> un Asistente en la pantalla de configuración <Cuando> modifica las opciones de configuración <Entonces> la aplicación se comporta según las opciones de configuración modificadas.		
Quiero			
configurar la aplicación			
Para			
que la aplicación se comporte según mis necesidades			

Historias de usuario

Nombre	Configurar conexión Bluetooth con Hygia Control Unit	Estimación	3
Épica	Configurar la aplicación		
Cómo	Criterios de aceptación		
Asistente	- <Dado> un Asistente en la pantalla de configuración <Cuando> pulsa la opción "Bluetooth" <Entonces> se habilita/inhabilita la conexión bluetooth del dispositivo. - <Dado> un Asistente en la pantalla de configuración <Cuando> pulsa la opción "Buscar Dispositivo" <Entonces> se muestra el listado de dispositivos bluetooth en alcance del dispositivo. - <Dado> un Asistente en la pantalla de configuración <Cuando> pulsa la opción "Conectar con dispositivo" <Entonces> la app se conecta con el dispositivo seleccionado.		
Quiero			
conectar la app con el componente hardware Hygia Control Unit.			
Para			
poder interactuar con el componente Hygia Control Unit.			

Capítulo 3: Diseño

1. Arquitectura general de la solución de movilidad Hygia

Este proyecto define una posible solución tecnológica para la prestación de servicios de atención remota sociosanitaria, dentro de la cual se integraría una aplicación móvil como herramienta principal de trabajo para el usuario. El alcance de este proyecto se centra en el diseño e implementación de la aplicación móvil, no obstante, a continuación se especifica el planteamiento de diseño de la solución.

A continuación se describen los componentes principales definidos en este proyecto para la solución de movilidad Hygia:

- **Hygia Control Unit o HCU:** dispositivo *hardware* disponible en la ubicación física de la persona en atención (piso, residencia, habitación, etc.):
 - proporcionar a la aplicación móvil una interfaz única de comunicación con el conjunto de sensores y equipamiento *hardware* de la solución.
 - comunicación con sensores y envío de información a la aplicación móvil
 - recepción de información de la aplicación móvil y actuación sobre dispositivos.
- **Hygia App:** aplicación móvil para los usuarios asistentes que prestan el servicio. Se comunica de forma bidireccional con el dispositivo HCU a través de servicios *backend* o directamente vía Bluetooth.
- **Hygia API (*Backend as a Service o BaaS*):** conjunto de APIs REST para ser consumidas por la aplicación móvil Hygia App. Se propone la utilización de la base de datos en tiempo real Firebase (Firebase Realtime Database) alojada en la nube como fuente de datos. Las funcionalidades que consumirá la aplicación móvil serán las siguientes:
 - Servicio de monitorización sociosanitaria: recuperar las alertas asistenciales, recuperar las personas en atención, etc., es decir, adquisición de datos del equipamiento *hardware*.
 - Servicio de asistencia sociosanitaria: ejecutar acciones asistenciales, es decir, envío de datos hacia el equipamiento *hardware*.
- **Hygia Directory Information o HDI:** base de datos de soporte/configuración de la solución. Ej: mensajes de las alertas, nombre de las personas en atención, indicadores asistenciales definidos en cada persona y sus sensores asociados, etc.

Se plantea como característica de la solución, la adaptabilidad a distintos ámbitos de utilización. En concreto, en este proyecto, se han planteado los siguientes:

- ámbito de utilización WWAN (Wireless Wide Area Network o Red Inalámbrica de Área Extensa):
 - permite una mayor cobertura geográfica del servicio sociosanitario pero requiere mayor infraestructura y recursos.
 - utilizará tecnologías 4G/5G para la comunicación inalámbrica entre los componentes Hygia Control Unit y la aplicación móvil Hygia App.
 - adecuado para prestar servicio en un ámbito geográfico amplio.

- ámbito de utilización WPAN (Wireless Personal Area Network o Red Inalámbrica de Área Personal):
 - soporta menor cobertura geográfica para la prestación del servicio pero es una solución autocontenida y muy flexible que necesita pocos recursos.
 - utilizará tecnología de comunicación Bluetooth para la comunicación inalámbrica entre los componentes Hygia Control Unit y la aplicación móvil Hygia App.
 - adecuado para espacios más reducidos, con menos recursos, temporales, etc. Por ejemplo, una pequeña residencia geriátrica, un bloque de pisos tutelados, un hospital de campaña, etc.

1.1 Diagrama general ámbito de utilización WWAN

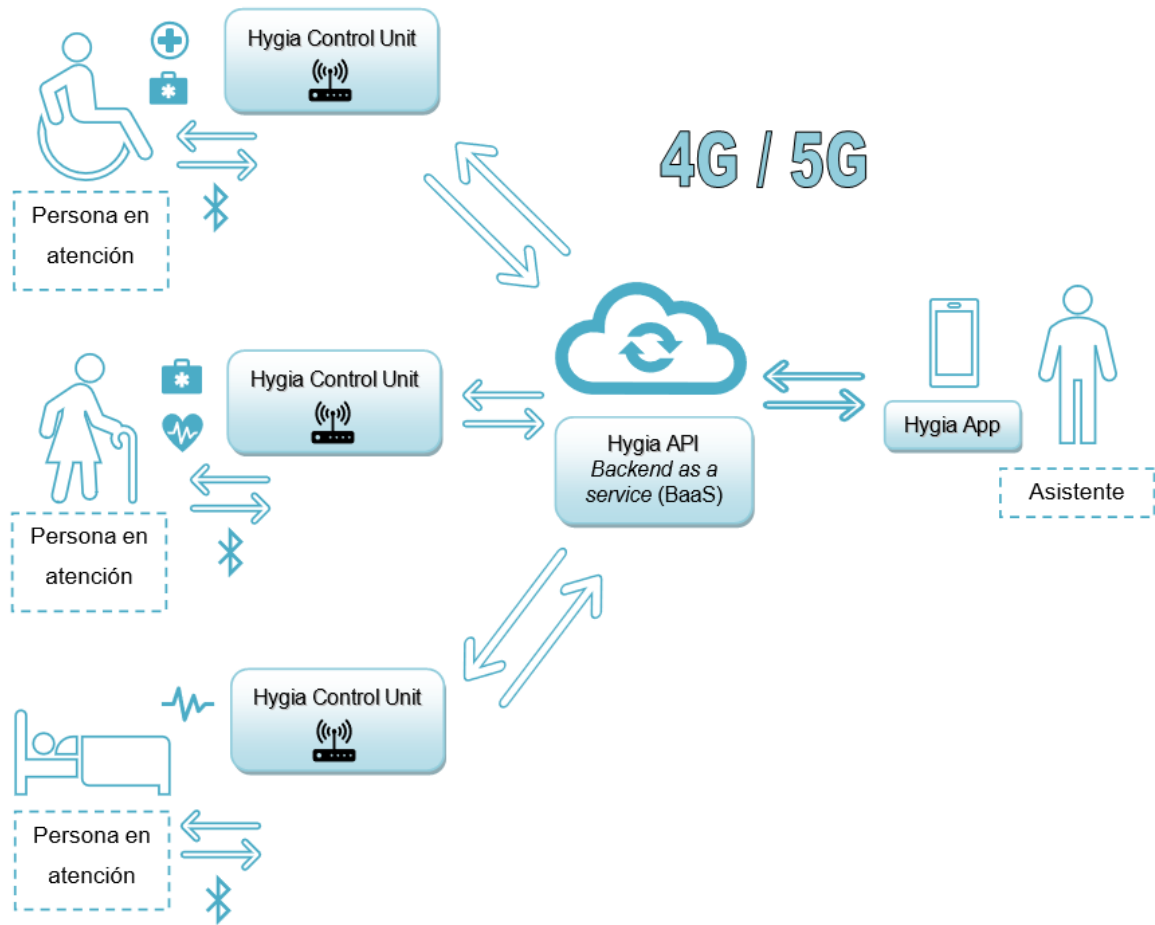


Diagrama 2. Ambito de utilización WWAN

1.2 Diagrama general ámbito de utilización WPAN

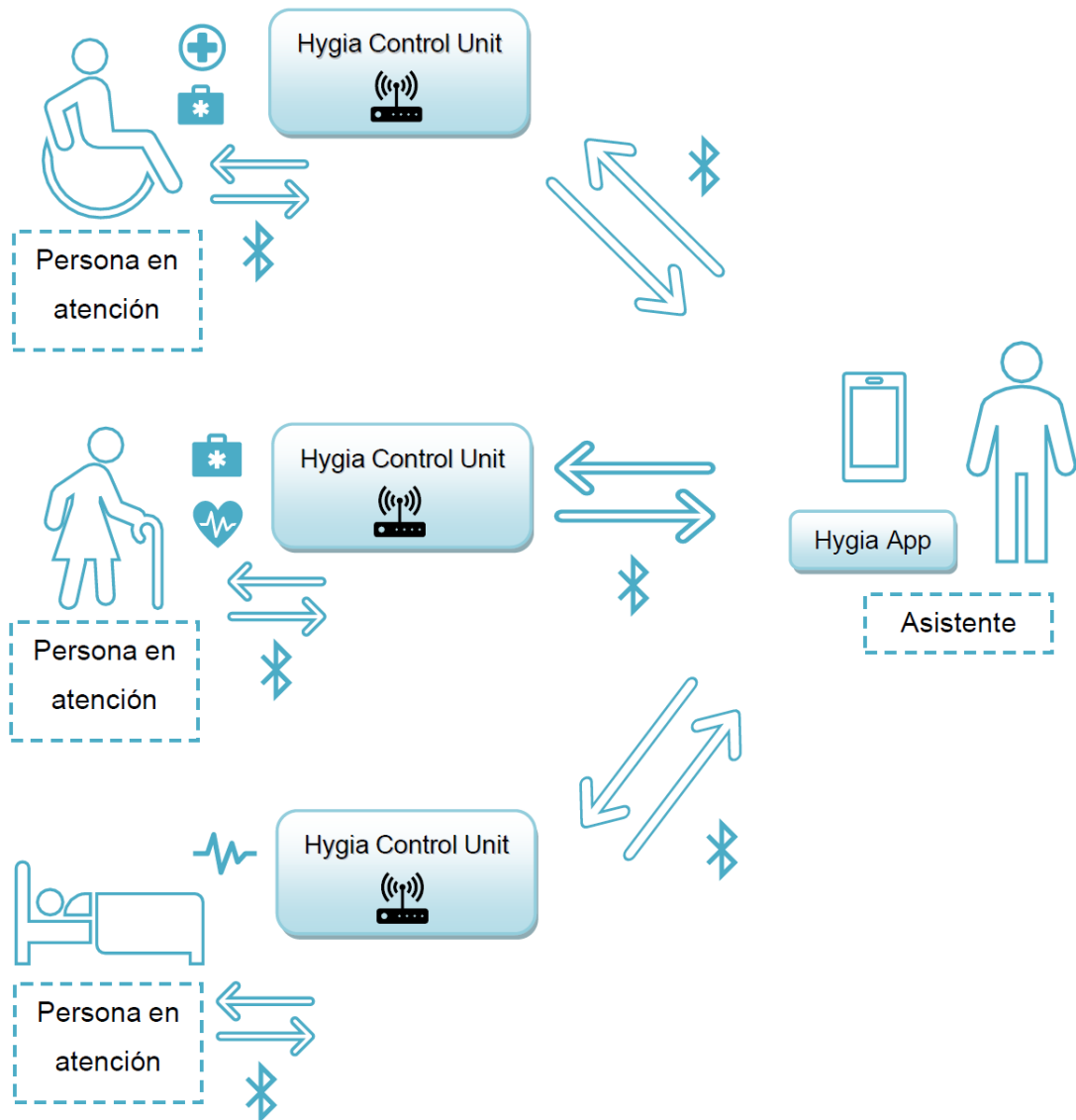


Diagrama 3. Ambito de utilización WPAN

2. Arquitectura de la aplicación móvil Hygia

El objetivo planteado fue aplicar un diseño que facilite el mantenimiento y la ampliación funcional progresiva de la aplicación. En este sentido, el diseño de arquitectura adoptado para la aplicación se ha realizado en base a la consulta y estudio de diversas fuentes en el ámbito de la ingeniería del software, como definiciones de patrones estándar de diseño o distintas aproximaciones de diversos autores.

En concreto, se han aplicado a la aplicación conceptos obtenidos de:

- Fundamentalmente los siguientes patrones de diseño estándar de arquitectura software [10]:
 - *BLoC (Business Logic Components) Pattern*: para independizar la lógica de negocio del interfaz de usuario y optimizar la gestión de los estados.
 - *Repository Pattern*: para independizar el dominio de datos de la tecnología de acceso a los mismos.
- La aproximación “The Clean Architecture” propuesta por Robert C. Martin [7], la cual integra a su vez conceptos de soluciones propuestas por otros autores (Hexagonal Architecture de Alistair Cockburn o Onion Architecture de Jeffrey Palermo).

La utilización del patrón BLoC (diseñado por Paolo Soares y Cong Hu de Google) se justifica por los siguientes motivos:

- BLoC es el patrón que recomienda Google para construir aplicaciones en Flutter. Fue presentado en su conferencia anual de desarrolladores Google I/O 2018. Lo considera como la mejor arquitectura para Flutter.
- La comunidad Flutter también recomienda su utilización.
- Permite la gestión eficiente de estados, algo sobre lo que precisamente estamos experimentado problemas.
- Permite ir añadiendo funcionalidad y aporta escalabilidad a la aplicación.
- Es flexible y fácil de mantener.

Para la implementación del ámbito de utilización integrado con APIs remotas o la base de datos Hive, la arquitectura definida se organiza como una estructura multi-capa, integrando el patrón BloC junto al patrón de Repositorio para independizar el modelo y la lógica de negocio del UI (User Interface o Interfaz de Usuario). Este diseño de arquitectura nos facilitará la ampliación funcional de la aplicación a futuro.

En el ámbito de utilización que implica comunicación bluetooth, se ha empleado la arquitectura de Scoped Model de Flutter para la gestión del estado y regeneración del interfaz de usuario. Se ha observado que esta aproximación ofrece muy buen rendimiento en la aplicación.

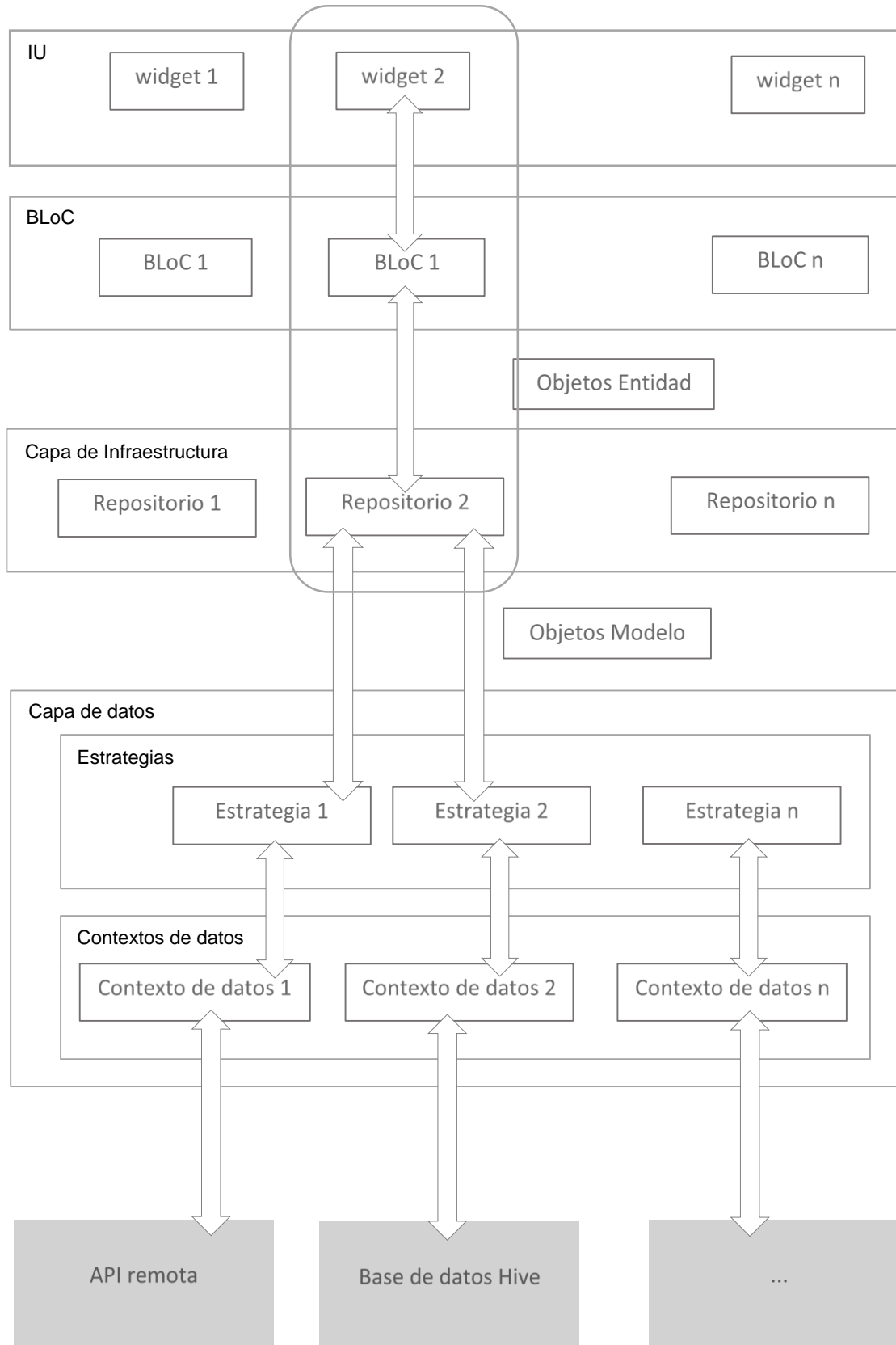


Diagrama 4. Arquitectura software de la aplicación

Otro de los objetivos del proyecto es producir una aplicación versátil, que pueda adaptarse a distintos ámbitos de utilización. El ámbito definido como red inalámbrica de área personal (WPAN) requiere disponer de información de soporte/configuración de los distintos componentes de la solución (sensores, actuadores, personas, etc.), de tal forma que la aplicación sea autocontenida y pueda utilizarse en ausencia de conectividad con servicios *backend*. En este sentido, y para almacenar dicha información de soporte/configuración se ha elegido la base de datos **Hive**. A continuación se especifican los motivos que justifican esta decisión:

- Se trata de una base de datos muy ligera y de alto rendimiento (está escrita 100% en Dart, el lenguaje que utiliza Google en Flutter y en su nuevo sistema operativo de tiempo real (Real-time operating systems o RTOS) Fuchsia).
- Es multiplataforma (mobile, desktop, browser), por tanto, nos ayuda a conseguir otro de los objetivos del proyecto (aplicación con enfoque multiplataforma). No tiene dependencias nativas.
- Fácil utilización y buena documentación.
- Soporta encriptación de la información.

3. Arquitectura entorno simulación de equipamiento hardware de la solución

3.1 Objetivo

Se ha considerado adecuado, y que añade valor al proyecto, diseñar y construir un entorno hardware real para poder realizar pruebas integradas con la aplicación, en lugar de usar solamente objetos o datos *mock*, puesto que una de las funcionalidades principales de la app móvil es precisamente la interoperabilidad con equipamiento hardware. Esta interacción real es fundamental, por ejemplo, para validar el tiempo de respuesta y rendimiento de la aplicación.

Este entorno se utilizará para desarrollar y probar el ámbito de utilización de red inalámbrica de área personal (WPAN) de la aplicación móvil.

3.2 Diseño

El diseño planteado para este entorno se basa en la utilización de un microcontrolador, como núcleo central de gestión, y un conjunto de componentes complementarios gobernados por dicho microcontrolador. Se ha acudido a una solución de mercado que integre estas características y facilite su implementación. En concreto, se ha seleccionado la placa *open-hardware* Arduino Uno REV3. En el apartado 4.3 *Placa Arduino para entorno de simulación hardware* de este documento se incluye el análisis realizado y la justificación a las decisiones tomadas al respecto.

Estos son los componentes utilizados en el diseño del entorno:

- Placa ARDUINO UNO REV3
- Módulo Bluetooth HC-05 DSD TECH
- 2 proto board
- 3 diodos led
- 3 pulsadores de botón
- 1 buzzer
- Cableado

El conjunto de la placa Arduino y el componente Bluetooth HC-05 simulan el elemento denominado en el proyecto **Hygia Control Unit o HCU**. Este elemento de la solución funcionaría como concentrador de los distintos sensores/actuadores configurados en un área determinada y se comunicaría con la app móvil Hygia, directamente (en un ámbito WPAN) o a través de servicios de *backend* (en un ámbito WWAN).

Por otra parte, se ha definido un conjunto de 2 habitaciones, cada una de las cuales con dos sensores y dos actuadores:

- Habitación 1
 - Sensor 1 (PIN2, pulsador)
 - Actuator 1 (PIN3, *buzzer*)
 - Sensor 2 (PIN4, pulsador)
 - Actuator 2 (PIN5, led)
- Habitación 2
 - Sensor 1 (PIN8, pulsador)
 - Actuator 1 (PIN9, led)
 - Sensor 2 (PIN10, pulsador)
 - Actuator 2 (PIN11, led)

Se ha definido que cada sensor del entorno esté caracterizado por la siguiente información:

- UUID (identificador único universal)
- valor
- instante de tiempo de lectura del valor
- información de geolocalización: longitud, latitud, nombre, descripción

A continuación se incluye el esquema realizado mediante la herramienta de código abierto Fritzing:

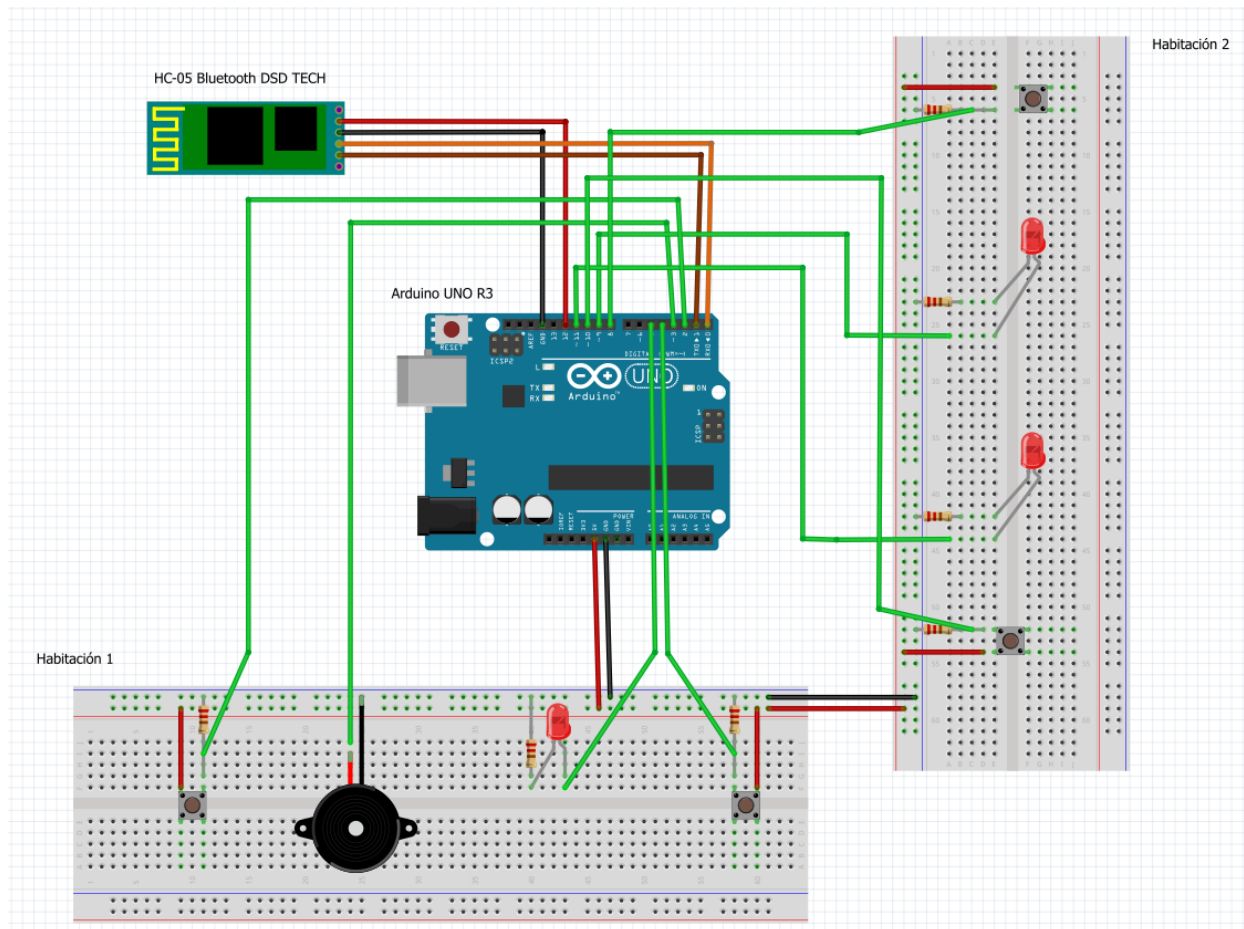


Diagrama 5. Diseño Fritzing entorno simulación hardware

4. Arquitectura de la información

A continuación se describen las principales entidades que componen el modelo informacional manejado por la aplicación y cómo se relacionan entre sí.

4.1 Baliza (*beacon*)

Se trata de una abstracción lógica que realiza el dispositivo *hardware* HCU (Hygia Control Unit) sobre los distintos elementos que tiene conectados y de los que recupera información. Estas balizas son enviadas por el HCU a la aplicación móvil cuando se produce algún cambio de estado, o bien, a petición de la app. La información de una baliza se compone de los siguientes atributos:

- UUID (Identificador único universal)
- valor
- fecha de lectura del valor
- información de geolocalización: longitud, latitud, nombre, descripción

El objetivo de esta entidad es disponer de un único interfaz genérico de datos entre el HCU y la aplicación, que represente a cualquier tipo de sensor o dispositivo de adquisición de datos conectado a la solución.

4.2 Sensor

Entidad que representa a un dispositivo de adquisición de datos asociado a una persona en atención. Por ejemplo, un tensiómetro digital. Son objetos lógicos que componen la aplicación móvil a partir de la información recibida en tiempo real del HCU (balizas) y que complementa con la información de configuración asociada recuperada de la base de datos Hygia Directory Information (HCU). Atributos principales:

- UUID (Identificador único universal)
- valor
- información de geolocalización: longitud, latitud, nombre, descripción
- nombre
- descripción
- tipo (digital, analógico)

La aplicación móvil actualizará los atributos de valor y geolocalización con la información recibida por las balizas correspondientes enviadas por el HCU. El resto de atributos (nombre, descripción, tipo) los recuperará de la base de datos HDI.

4.3 Actuador

Abstracción lógica que representa a cualquier dispositivo sobre el que se puede actuar en remoto para asistir a una persona. Por ejemplo, un dispensador de medicamentos. Atributos principales:

- UUID (Identificador único universal)
- valor
- información de geolocalización: longitud, latitud, nombre, descripción

La aplicación móvil enviará al HCU estas entidades para actuar sobre los dispositivos según las acciones asistenciales correspondientes.

4.4 Alerta asistencial

Entidad que representa un evento o situación asociada a una persona que requiere atención por parte del usuario Asistente. Son las unidades de información que gestionará el usuario Asistente desde el espacio de trabajo de alertas asistenciales. Principales atributos:

- Identificador único
- Tipo
- Nombre
- Mensaje
- Estado
- Persona en atención asociada
- Sensor asociado
- Acción asistencial asociada

4.5 Persona en atención

Entidad que representa una persona que requiere atención por parte del usuario Asistente. Son las unidades de información que gestionará el usuario Asistente desde el espacio de trabajo de personas en atención. Principales atributos:

- UUID (Identificador único universal)
- Nombre
- Edad
- Foto
- Información de soporte asistencial
- Información de contacto
- Información de geolocalización: longitud, latitud, nombre, descripción
- Indicadores asistenciales asociados

4.6 Indicador asistencial

Entidad que representa un aspecto asistencial determinado al que el usuario Asistente debe prestar atención de cara a asegurar el correcto soporte a la persona. Por ejemplo, nivel de dolor o estado cardiovascular. Los sensores de la solución se asociarán, por configuración, a los distintos indicadores definidos, de tal forma que un indicador podrá alimentarse de un conjunto de sensores. Los cambios de estado de los sensores determinarán el estado del indicador. Atributos principales:

- UUID (Identificador único universal)
- Tipo
- Nombre
- Mensaje
- Estado

4.7 Acción asistencial

Entidad que representa una acción posible de actuación del usuario Asistente sobre un dispositivo del de la persona en atención, de cara a proporcionarle algún tipo de asistencia en respuesta a una situación de alerta. Por ejemplo, activar un dispensador de medicamentos en respuesta a una solicitud del paciente. Atributos principales:

- UUID (Identificador único universal)
- Tipo
- Nombre
- Descripción
- Actuador asociado
- Estado
- Instante de tiempo de ejecución de la acción

5. Diseño gráfico e interfaces

5.1 Proceso de trabajo

Se ha definido en este proyecto un marco de trabajo que integra la visión UX en todas sus fases y no solo en momentos puntuales (como por ejemplo en una fase inicial de diseño). Siguiendo el enfoque *agile*, el objetivo perseguido es evitar realizar por anticipado tareas que no generen valor, por ejemplo, refinar diseños de pantallas que finalmente no se lleguen a implementar y, por tanto, no añadan valor al producto final.

En este sentido, se han realizado trabajos de UX:

- en la etapa del proyecto de definición y escritura de historias de usuario.
 - definiendo y caracterizando a los usuarios objetivo de la app mediante la generación de arquetipos.
 - diseñando el interfaz de usuario de la aplicación y generando un conjunto de *wireframes*, como complemento a las historias de usuario, que sirvieron como base a las etapas posteriores de desarrollo de los diferentes ciclos de sprint.
- en la etapa de desarrollo de los ciclos de sprint:
 - enriqueciendo y refinando el diseño del interfaz de usuario a partir de los *wireframes* base generados en la etapa de definición inicial, a medida que se fueron explorando las posibilidades del *framework* y la tecnología de desarrollo (widgets disponibles, librerías, etc.).
 - diseñando y produciendo los recursos gráficos necesarios (logo, iconos, fuentes, etc.).
 - revisando y probando la usabilidad de la aplicación una vez construida.

Todo ello para lograr la mejor experiencia de usuario posible y alcanzar los objetivos establecidos en el proyecto para este aspecto: simplicidad y rendimiento óptimo.

5.2 Arquetipos

El usuario principal de la aplicación será el denominado *asistente*. En esta persona se centrará el esfuerzo de UX de este trabajo. A continuación se especifican algunos arquetipos construidos en el proceso de análisis para su caracterización:

Nombre	María
Edad	38
Ocupación	Trabajadora social
Bio	María está casada y tiene un hijo de 3 años. Trabaja en una empresa de ámbito nacional atendiendo a personas con necesidades asistenciales ubicada en la ciudad autónoma de Ceuta. Utiliza un vehículo de empresa para desplazarse a los distintos domicilios de las personas a las que atiende. Tiene una planificación diaria para realizar las visitas aunque, en función de las necesidades de las personas atendidas, es habitual que dicha planificación se modifique y adapte varias veces durante el día.
Nivel tecnológico	Usuario medio. Utiliza habitualmente dispositivos móviles (<i>smartphone, tablet</i>) y distintas aplicaciones de productividad para su trabajo. Se muestra activa en redes sociales.
Escenario	<p>María acaba de salir del domicilio de Antonio, una de las personas que tiene bajo su atención. Antonio tiene problemas respiratorios y dispone en su domicilio de un dispositivo que le proporciona oxígeno. María se lleva apuntada de esta visita la necesidad de llamar a la empresa que gestiona las bombonas para que reemplace la existente, pues está a punto de terminarse.</p> <p>María tiene bajo su responsabilidad a varias personas que disponen de oxígeno en casa. Antes de llamar a la empresa proveedora de oxígeno, María accede a la app Hygia para verificar si tiene alguna alerta asistencial de nivel de oxígeno bajo de otras personas para realizar un pedido conjunto y evitar llamadas posteriores. Además de las alertas, María consulta los indicadores asistenciales de las personas en atención que disponen de oxígeno, pues puede que en algún caso no se haya generado alerta todavía pero se genere en breve, por tanto, quiere aprovechar el pedido que va a realizar (la distribución de las bombonas puede demorarse unos días). De esta forma, María se anticipa al posible problema y asegura el abastecimiento de oxígeno necesario. La app Hygia permite a María en este caso obtener esta información sin tener que desplazarse a los distintos domicilios, realizar una única gestión con la empresa distribuidora y asegurar que las necesidades de las personas están cubiertas.</p>

Tabla 17. Arquetipo usuario 1

Nombre	Pedro
Edad	55
Ocupación	Enfermero
Bio	<p>Pedro es soltero. Vive solo en un piso en propiedad en la provincia de Soria. Trabaja en una residencia geriátrica ubicada a 20 kilómetros de su domicilio. La residencia está ubicada en un pueblo pequeño, de apenas 100 habitantes en invierno. Se trata realmente de una casa grande de dos plantas habilitada para funcionar como residencia geriátrica y centro de día, pero con recursos limitados.</p> <p>Pedro tiene turnos rotativos. Prefiere trabajar en el turno de mañana.</p>
Nivel tecnológico	<p>Usuario básico. Dispone de <i>smartphone</i> de empresa. Utiliza habitualmente las aplicaciones de escritorio básicas de su empresa para registrar su trabajo o acceder la información de pacientes (ofimática).</p>
Escenario	<p>Pedro se encuentra en la habitación de Ángel, uno de las personas que atiende en la residencia geriátrica. Acaba de realizarle una cura y, como es habitual en Ángel, establece conversación cordial con Pedro, pues se conocen desde hace ya algunos años. En ese momento, Pedro recibe una notificación en el <i>smartphone</i> de empresa, que lleva siempre consigo durante su turno. Al consultar la notificación verifica que se trata de una alerta asistencial de la app Hygia en relación al aviso de llamada desde la habitación de Carmen, otra residente bajo su atención y, por tanto, Pedro decide posponer la conversación con Ángel y dirigirse inmediatamente a la habitación indicada. La app Hygia a permitido a Pedro mejorar el tiempo de respuesta de atención a Carmen, recibiendo la necesidad en tiempo real.</p>

Tabla 18. Arquetipo usuario 2

Nombre	Andrés
Edad	41
Ocupación	Médico
Bio	Andrés está casado y tiene dos hijos. Es autónomo. Vive en Barcelona y ha montado una pequeña empresa para proporcionar servicios de asistencia sociosanitaria domiciliaria. En este momento, el margen de ganancia por paciente es bajo y, junto a su socio Carlos, está intentando buscar marcos de colaboración con mutuas, aseguradoras o entidades públicas que les proporcionen un mayor número de clientes.
Nivel tecnológico	Usuario avanzado. Utiliza habitualmente dispositivos móviles (<i>smartphone, tablet</i>) y múltiples herramientas software para la gestión y administración de su negocio.
Escenario	Andrés acaba de llegar a un acuerdo con una conocida mutua de salud para proveer servicios de asistencia sociosanitaria a personas con domicilio en diferentes localidades de la provincia de Barcelona. La noticia es muy buena, no obstante, Andrés tiene ahora que planificar con su socio la mejor manera de prestar esta atención para que sea rentable, pues solo disponen de tres personas para proporcionar el servicio y el área geográfica a cubrir es relativamente amplia. Andrés encuentra en el mercado de herramientas tecnológicas para prestación de servicios sociosanitarios la solución Hygia que, por sus características de movilidad y asistencia remota, encaja perfectamente con las necesidades de ahorro de recursos.

Tabla 19. Arquetipo usuario 3

5.3 Wireframes

A continuación se incluye el conjunto de *wireframes* generados en el proyecto:

Consultar el listado de alertas asistenciales

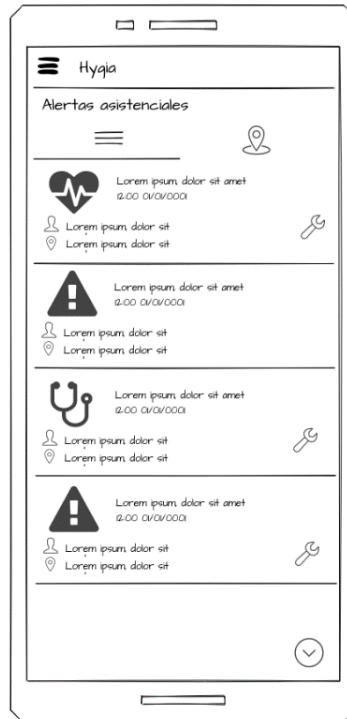


Ilustración 2. Wireframe Consultar el listado de alertas asistenciales

Consultar el mapa de alertas asistenciales

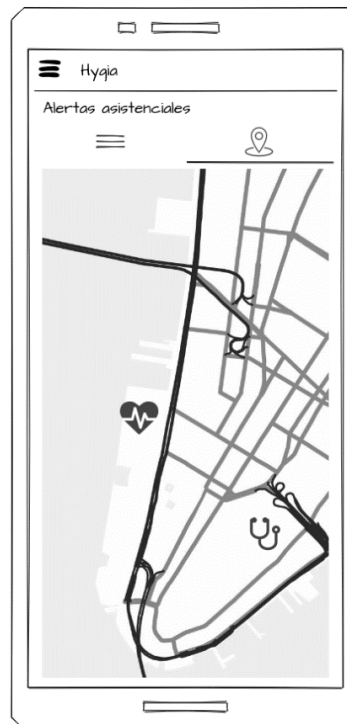


Ilustración 3. Wireframe Consultar el mapa de alertas asistenciales

Consultar el detalle de las alertas asistenciales



Ilustración 4. Wireframe Consultar el detalle de las alertas asistenciales 1

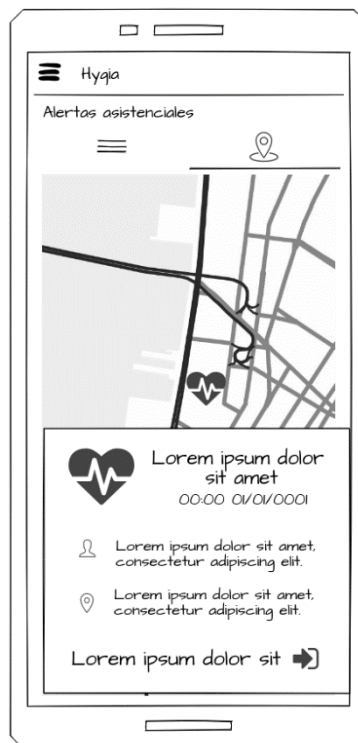


Ilustración 5. Wireframe Consultar el detalle de las alertas asistenciales 2

Consultar el listado de personas en atención

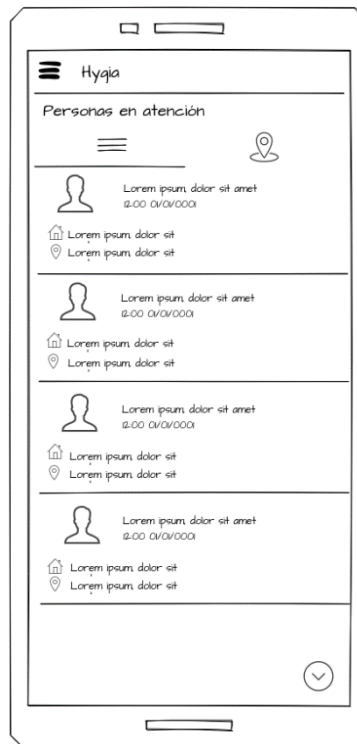


Ilustración 6. Consultar el listado de personas en atención

Consultar el mapa de personas en atención



Ilustración 7. Consultar el mapa de personas en atención

Consultar la información de detalle de las personas en atención



Ilustración 8. Consultar la información de detalle de las personas en atención 1

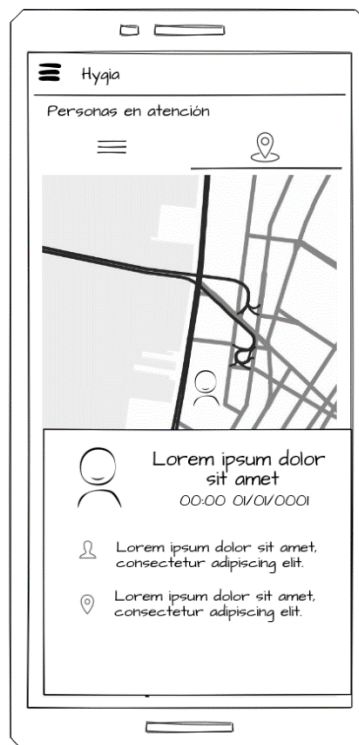


Ilustración 9. Consultar la información de detalle de las personas en atención 2

Acceso biométrico: huella digital



Ilustración 10. Acceso biométrico: huella digital

Acceso con pin de seguridad por teclado

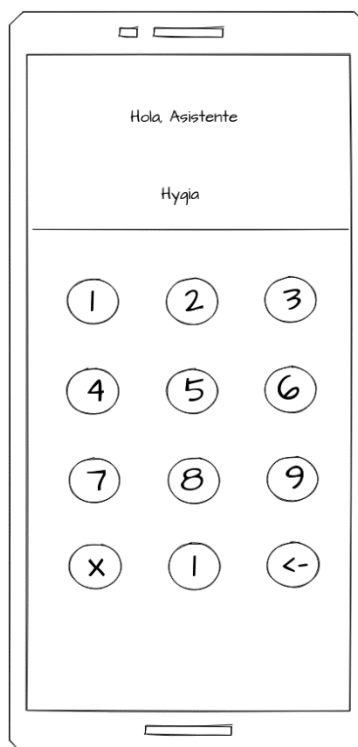


Ilustración 11. Acceso con pin de seguridad por teclado

Configurar conexión bluetooth con Hygia Control Unit

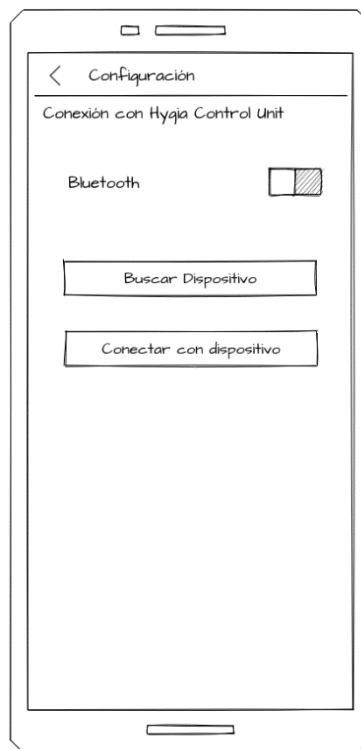


Ilustración 12. Configurar conexión bluetooth con Hygia Control Unit 1

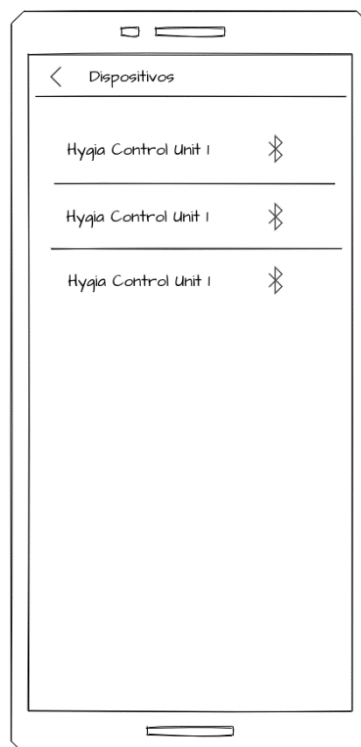


Ilustración 13. Configurar conexión bluetooth con Hygia Control Unit 2

Menú (*app drawer*)



Ilustración 14. Menú (*app drawer*)

5.4 Usabilidad/UX

El objetivo planteado en el proyecto en cuanto a usabilidad y experiencia de usuario de la aplicación es obtener un interfaz muy funcional, sencillo y con un tiempo de respuesta óptimo. La aplicación va destinada a personas con distintos niveles tecnológicos y se utilizará en el ámbito profesional, como herramienta de trabajo diario. Se busca un consumo bajo de recursos del dispositivo.

Se propone, como elemento base para el diseño de la aplicación, hacer utilización de patrones de diseño estándar. Se trata de soluciones ya probadas que nos permitirán crecer sobre una base sólida y efectiva.

En este sentido, se utilizará el estándar de diseño visual multiplataforma Material Design [13], orientado tanto a smartphones, tablets, smartwatches, televisores o incluso páginas web. Con Material Design se dispone de múltiples componentes que implementan patrones de diseño estándar. Flutter permite trabajar con naturalidad con este lenguaje de diseño visual y ofrece un conjunto amplio de componentes (Material widgets).

A continuación se enumeran algunos ejemplos de patrones de diseño que se ha identificado conveniente utilizar en la aplicación bajo el marco de Material Design:

- *Swipe To Refresh* o *Pull To Refresh*, para que el usuario pueda refrescar el panel de alertas asistenciales y personas en atención.
- *Hamburger menu - Navigation drawer*.



Ilustración 15. Patrón Hamburger menu - Navigation drawer

- *Componente Card*.

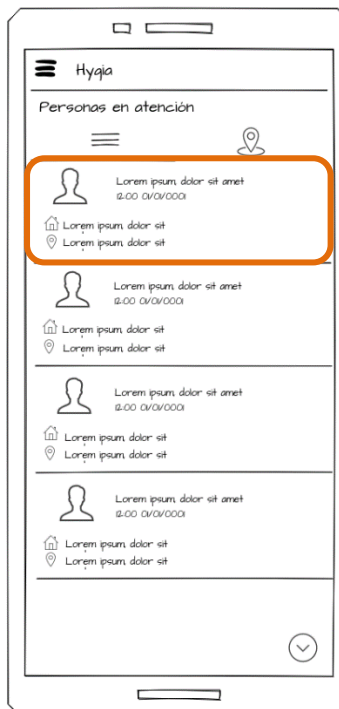


Ilustración 16. Patrón Card

5.5 Diseño de componentes de interfaz

Se han planteado dos espacios de trabajo principales en la aplicación: alertas asistenciales y personas en atención. A estos espacios se accede a través del menú de navegación de la aplicación (*app drawer*):

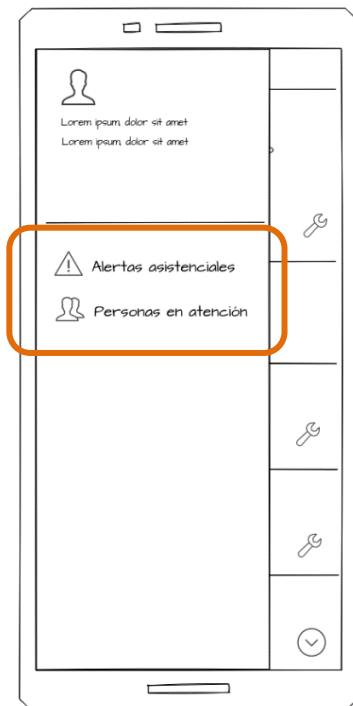


Ilustración 17. Espacios de trabajo

Por defecto, se abrirá el espacio de alertas asistenciales, pues contiene las situaciones que debe atender en primer lugar el usuario Asistente de la aplicación. Este espacio mostrará el listado de alertas generadas desde las personas en atención y recuperadas en el ámbito de utilización actual de la aplicación.

El espacio de trabajo de personas en atención mostrará el listado de personas bajo la responsabilidad del usuario Asistente.

En ambos espacios de trabajo se ofrecen dos tipos de visualizaciones:

- Vista de listado: conjunto de tarjetas.
- Vista de mapa: especialmente útil para la gestión de los elementos en movilidad.

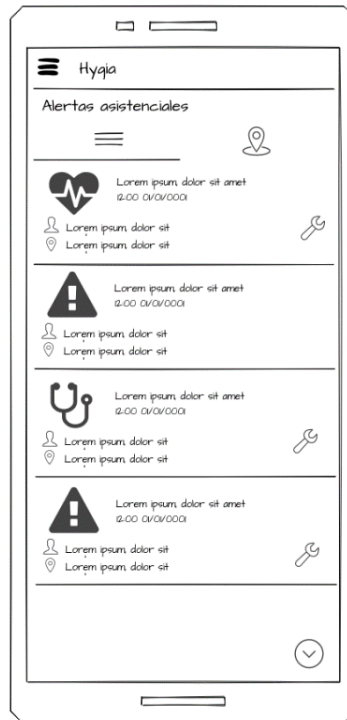


Ilustración 18. Vista de listado



Ilustración 19. Vista de mapa

Tanto en las tarjetas como en las pantallas de detalle de las entidades se ha buscado un diseño sencillo y coherente, que muestre exactamente la información que el usuario necesita para la gestión de su trabajo, evitando información superflua, respondiendo a :

- ¿Qué? evento se ha producido
- ¿Cuándo? se ha producido el evento
- ¿Quién? es la persona afectada
- ¿Dónde? se ha producido el evento
- ¿Cómo? puedo solucionarlo

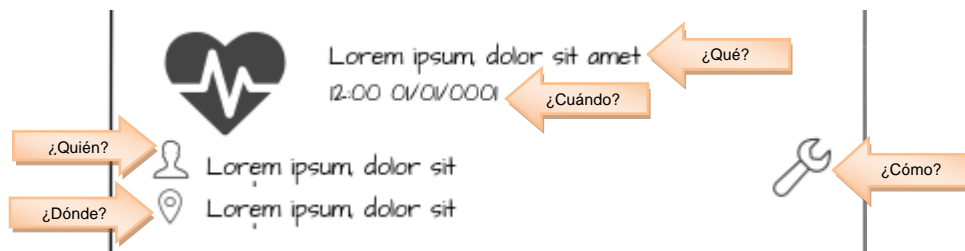


Ilustración 20. Diseño tarjetas

5.6 Estilos

La aplicación utiliza el tema oscuro de Material Design [13]. La decisión se basa en la identificación de las siguientes ventajas:

- Reducción de fatiga visual.
- Reducción de consumo de batería.
- Tendencia actual de diseño [28].

Fuentes tipográficas

Se ha utilizado en la aplicación la fuente Roboto de Google Fonts.

Designed by Christian Robertson. License: Apache License, Version 2.0.

Componentes propios utilizados

Logotipo App Hygia



Creado con la herramienta canva (canva.com)

Color de fondo: #03dac6

Color primer plano: #000000

Iconos notificaciones



Creado con la herramienta canva (canva.com)

Color de fondo: #000000

Color primer plano: #FFFFFF

Ilustración 21. Iconos notificaciones

Iconos lanzador

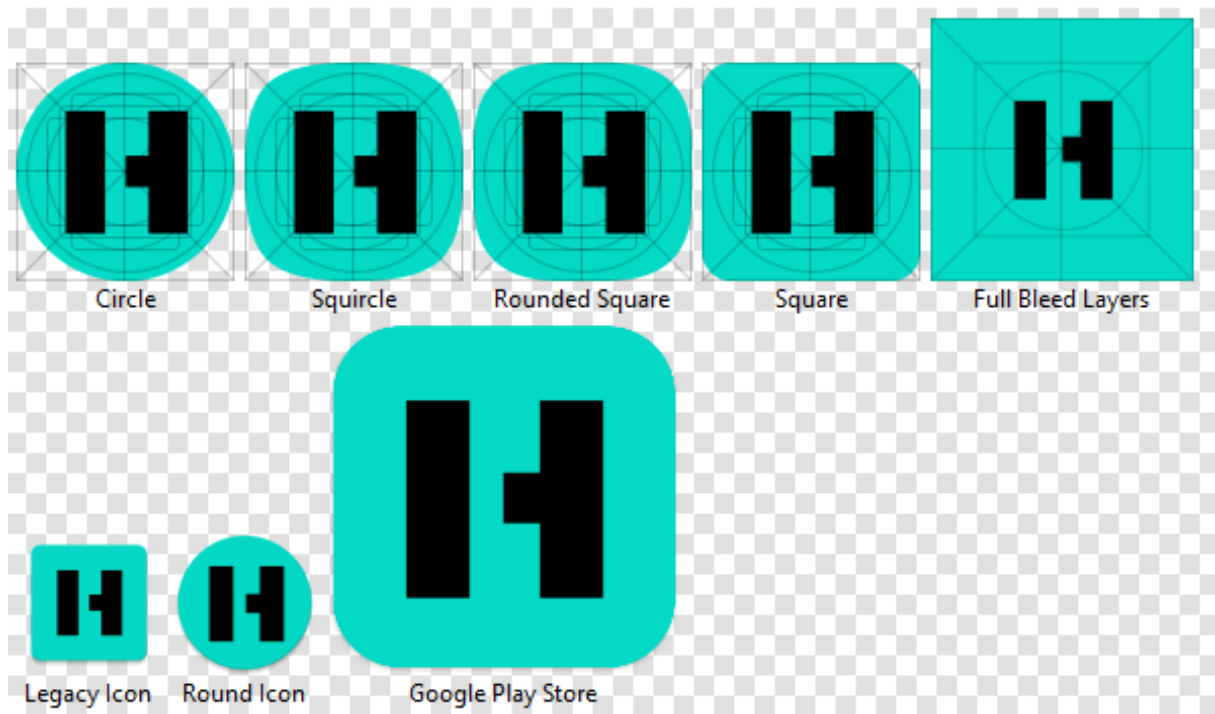


Ilustración 22. Iconos lanzador

Creado con la herramienta canva (canva.com)

Color de fondo: #03dac6

Color primer plano: #000000

Componentes ajenos utilizados

Iconos Material Design utilizados



Iconos alertas asistenciales

Icon made by Freepik (<https://www.flaticon.com/authors/freepik>) from Flaticon (www.flaticon.com).

Flaticon License: Free for personal and commercial purpose with attribution.



Iconos personas en atención

Estos iconos, en una release de producción real, se sustituirán por fotografías reales de las personas en atención.

Icon made by Eucalyp (<https://www.flaticon.com/authors/eucalyp>) from Flaticon (www.flaticon.com).

Flaticon License: Free for personal and commercial purpose with attribution.



6. Lenguajes de programación y APIs utilizadas

6.1 Flutter como framework de desarrollo

De las opciones de mercado analizadas en detalle en el estudio de estado del arte de este documento (Capítulo 2. Análisis), se ha seleccionado el *framework* Flutter de Google para el desarrollo de la aplicación. Este *framework* presenta mucho potencial para cubrir tres de los objetivos del proyecto:

- Construir la aplicación sobre una base única de código que permita su evolución multiplataforma y multidispositivo, para reducir costes de desarrollo y mantenimiento y facilite la entrada del producto en el mercado.
- Construir una aplicación con una interfaz gráfica sencilla pero de alto rendimiento y fluidez gráfica para dar cobertura a tiempos de respuesta exigentes.
- Aprender y poner en práctica las últimas tecnologías y herramientas de mercado de desarrollo de aplicaciones móviles.

Estos son los puntos más relevantes que justifican la decisión de elegir Flutter para el desarrollo del producto de este trabajo:

- Se trata de una de las grandes apuestas de Google que, de hecho, está utilizando en la construcción de su nuevo sistema operativo de tiempo real (Real-time operating systems o RTOS) Fuchsia.
- Se desarrolla un mismo código base que se puede utilizar en móvil, web, embebido y desktop, es decir, es un framework con orientación multiplataforma y multidispositivo.
- Se utiliza el lenguaje Dart (creado por Google) compilado en código nativo, por tanto, es muy rápido.
- El framework facilita en gran medida la construcción proporcionando múltiples componentes (widgets) basados en el estándar de diseño visual Material Design.
- Aunque es una tecnología relativamente joven, tiene una comunidad amplia, buen soporte y contribuciones (*Third-party libraries*).

6.2 APIs de terceros utilizadas en Flutter

Librería	Versión	Descripción
http	^0.12.0+4	API para gestionar peticiones HTTP: https://pub.dev/packages/http
google_maps_flutter	^0.5.25+3	Librería oficial de Google para la integración de Google Maps (tanto para iOS como Android): https://pub.dev/packages/google_maps_flutter
flutter_bluetooth_serial	^0.2.2	Librería para la comunicación con bluetooth: https://pub.dev/packages/flutter_bluetooth_serial
flutter_bloc	^4.0.0	Librería Google para la implementación del patrón BLoC: https://pub.dev/packages/flutter_bloc
scoped_model	^1.0.1	Modelo reactivo para la actualización de widgets: https://pub.dev/packages/scoped_model
equatable	^1.1.1	Clase abstracta para facilitar la gestión de

		comparación de objetos : https://pub.dev/packages/equatable
intl	^0.16.1	Utilidades para, por ejemplo, formateo de fechas e internacionalización: https://pub.dev/packages/intl
local_auth	^0.6.2+1	Plugin para facilitar la implementación de la autenticación vía lector de huellas: https://pub.dev/packages/local_auth
flutter_local_notifications	^1.4.2	Plugin para facilitar la implementación de notificaciones locales: https://pub.dev/packages/flutter_local_notifications
hive	^1.4.1+1	Base de datos HIVE: https://pub.dev/packages/hive
path_provider	^1.6.7	Utilidad para acceder al sistema de ficheros del dispositivo: https://pub.dev/packages/path_provider
hive_generator	^0.7.0+2	Utilidad para la generación de Type Adapters en HIVE: https://pub.dev/packages/hive_generator
build_runner	^1.9.0	Utilidad de generación de código y compilación modular: https://pub.dev/packages/build_runner
flutter_lock_screen	^1.0.6	Utilidad para la implementación de pantalla de acceso a la aplicación: https://pub.dev/packages/flutter_lock_screen

Tabla 20. APIs de terceros utilizadas en Flutter

6.3 Placa Arduino para entorno de simulación hardware

Para la construcción del entorno de simulación hardware de la solución se ha realizado un análisis y estudio de las diferentes alternativas de mercado. A continuación se indican las consideradas como más relevantes:

- Raspberry Pi: (System-on-a-chip Broadcom BCM2835). Se trata de un pequeño ordenador con una potencia de cálculo razonable, aunque requiere de sistema operativo completo. El software es de código abierto pero el hardware no (Raspberry Foundation fabrica en exclusiva las placas).
- Arduino: Esta plataforma es completamente abierta, tanto, el software como el hardware. No tiene tanta potencia de cálculo pero no requiere sistema operativo y tiene una muy buena curva de aprendizaje.
- Wasmote: creada por la startup española Libelium. Es una plataforma modular de código abierto orientada a redes inalámbricas de bajo consumo.
- BeagleBone: una placa reducida que conforma un ordenador completo, con microprocesador. Funciona con Linux.

De las alternativas estudiadas se ha elegido la plataforma Arduino por los siguientes motivos:

- Está basada en *hardware* y *software* libre.
- Como consecuencia del motivo anterior, la placa es barata. Las especificaciones y diagramas son de acceso público, por tanto, cualquier empresa puede replicar el diseño, de hecho, existen en el mercado distintas opciones oficiales y otras compatibles a menor precio como, por ejemplo, ELEGOO UNO R3.
- Existen diversas versiones de la placa, con distintas características, para adaptarse a las distintas necesidades de cada proyecto: Arduino Uno, Arduino Nano, Arduino Micro, etc.
- Es fácil de utilizar, con buena curva de aprendizaje. Incorpora un entorno de desarrollo integrado (Arduino IDE) que facilita mucho la programación.
- El lenguaje de programación, similar a C, es familiar para mí y me permitirá construir rápido para poder así poner foco en el desarrollo de la app.
- Es fácil de obtener.

6.4 APIs de terceros utilizadas en Arduino

Librería	Versión	Descripción
ArduinoJson	6.15.1	Librería para manejar de forma sencilla y eficiente objetos json. Por ejemplo, facilita la serialización/deserialización de objetos.

Tabla 21. APIs de terceros utilizadas en Arduino

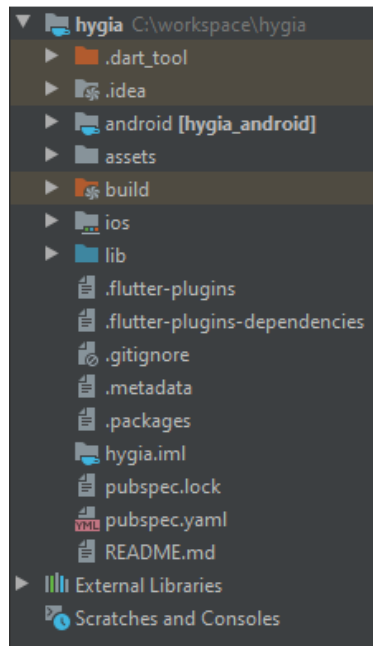
Capítulo 4: Implementación

1 Implementación de la aplicación con el *framework* Flutter

1.1 Estructura y descripción de clases

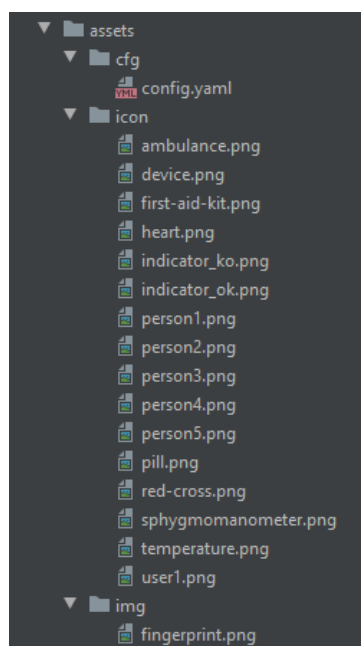
A continuación se describe la estructura aplicada al proyecto Flutter para la implementar el diseño de arquitectura propuesto.

- Proyecto Flutter:



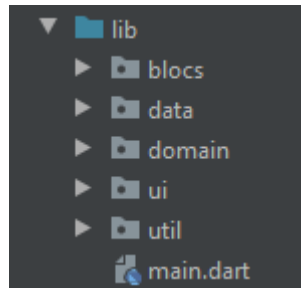
Captura 1. Android Studio: Proyecto Flutter

- /assets: recursos de la aplicación (ficheros de configuración, imágenes, etc.).



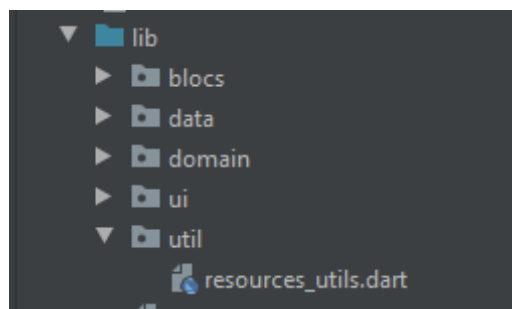
Captura 2. Android Studio: assets

- /lib/main.dart: Punto de entrada a la aplicación:



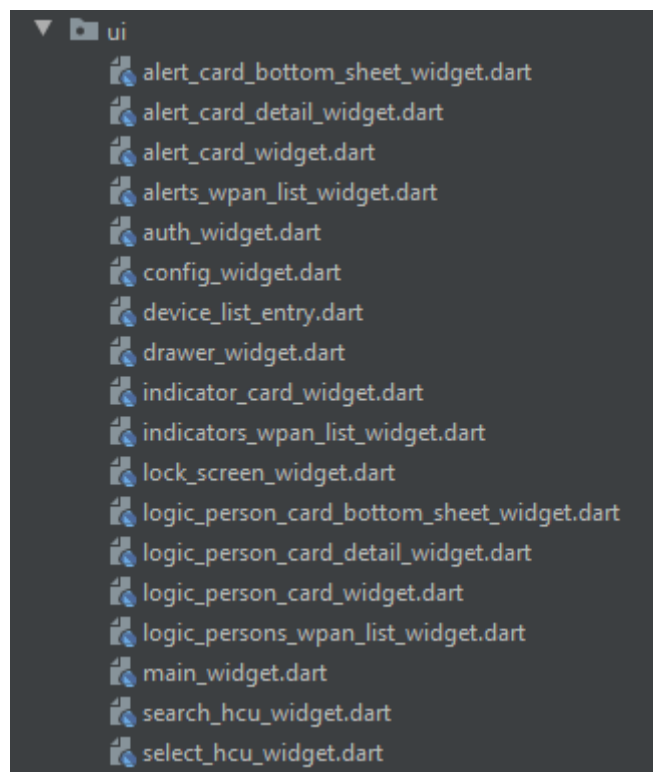
Captura 3. Android Studio: main

- /lib/util/resources_utils.dart: utilidades generales de sistema. Por ejemplo, mapeo de recursos gráficos:



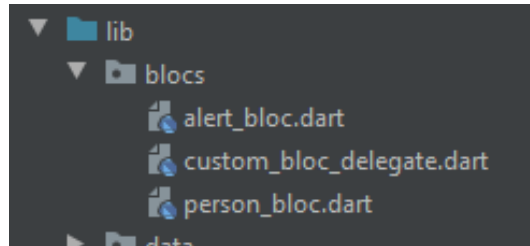
Captura 4. Android Studio: resources_utils

- /lib/ui: interfaz de usuario:



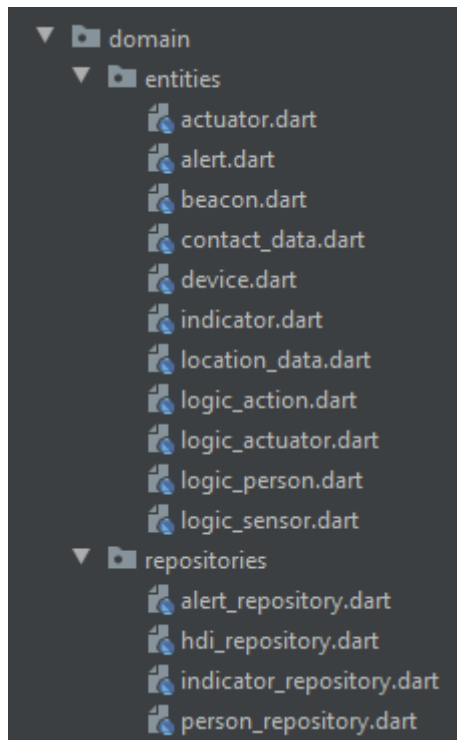
Captura 5. Android Studio: UI

- /lib/blocs: Implementación del patrón BLoC:



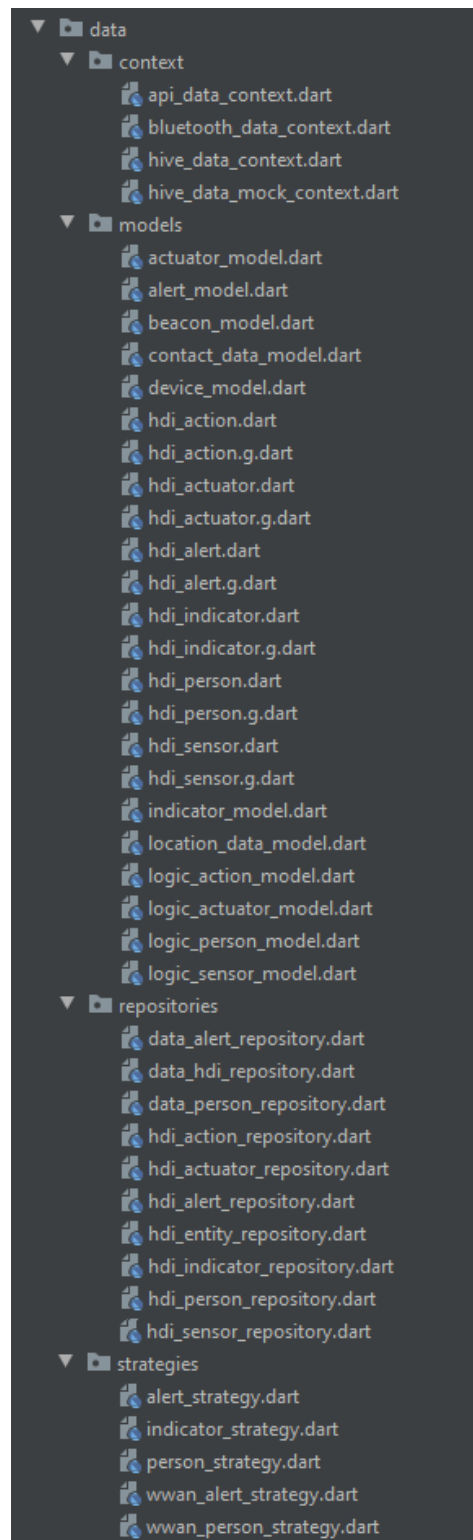
Captura 6. Android Studio: blocs

- /lib/domain: interfaces de dominio de la app:
 - /entities: modelo de negocio (entity objects o entidades de negocio)
 - /repositories: interfaces de los repositorios de las entidades de negocio



Captura 7. Android Studio: domain

- lib/data:
 - /models: *model objects* u objetos modelo.
 - /repositories: implementación de los repositorios de las entidades de negocio.
 - /strategies: estrategias de adquisición de datos.
 - /context: implementación de la comunicación con los distintas fuentes de datos.



Captura 8. Android Studio: data

1.1 Comunicación bluetooth App móvil - HCU

A continuación se describen las principales decisiones y aspectos a destacar de la implementación realizada para la comunicación bluetooth entre la aplicación móvil y el dispositivo HCU:

- Se utiliza el formato estándar JSON (JavaScript Object Notation o Notación de Objetos de JavaScript) para el intercambio de información entre el dispositivo HCU y la aplicación móvil. Las clases del modelo implementadas (/lib/data/model), como extensión a las clases de entidad (/lib/domain/entities), disponen por tanto de las correspondientes funciones fromJson()/ toJson().
- Se realiza un intercambio ligero de mensajes entre el HCU y la aplicación móvil, de cara a optimizar los tiempos de respuesta y simplificar el tratamiento y parseo de los datos intercambiados, tanto en la aplicación móvil como en el dispositivo HCU.
- La comunicación vía bluetooth entre la aplicación móvil y el dispositivo HCU está implementada en la clase BluetoothDataContext (lib/data/context/bluetooth_data_context.dart). Cuando el usuario selecciona el modo de trabajo WPAN en la aplicación móvil desde la sección de configuración (seleccionando un dispositivo HCU al que conectarse), se abre un hilo de ejecución en segundo plano para gestionar la comunicación con el dispositivo correspondiente a través de las funcionalidades implementadas en la citada clase BluetoothDataContext. A continuación se describen las principales funciones implementadas:
 - o *connect()*: conexión bluetooth con el dispositivo seleccionado por el usuario
 - o *fromConnection()*: escucha y serialización de la información recibida del dispositivo HCU (balizas). Se emplea un *buffer* para recibir los distintos caracteres hasta completar el mensaje JSON que compone la baliza de intercambio. Una vez disponible la baliza, se construyen los objetos del modelo correspondiente y se notifica a los listener (widget del interfaz de usuario) para que se regeneren y muestren su información actualizada (se ha empleado para ello una aproximación basada en *Scoped Model* pues e ha observado un rendimiento óptimo).
 - o *performActuator()*: permite enviar al HCU una orden de actuación sobre un dispositivo configurado en la solución (actuador).
 - o *requestPersons()*: permite enviar al HCU una petición de estado de las personas en atención.

- Cuando la aplicación se conecta a un dispositivo HCU le envía una primera petición para recuperar la información de las personas en atención (*requestPersons*):

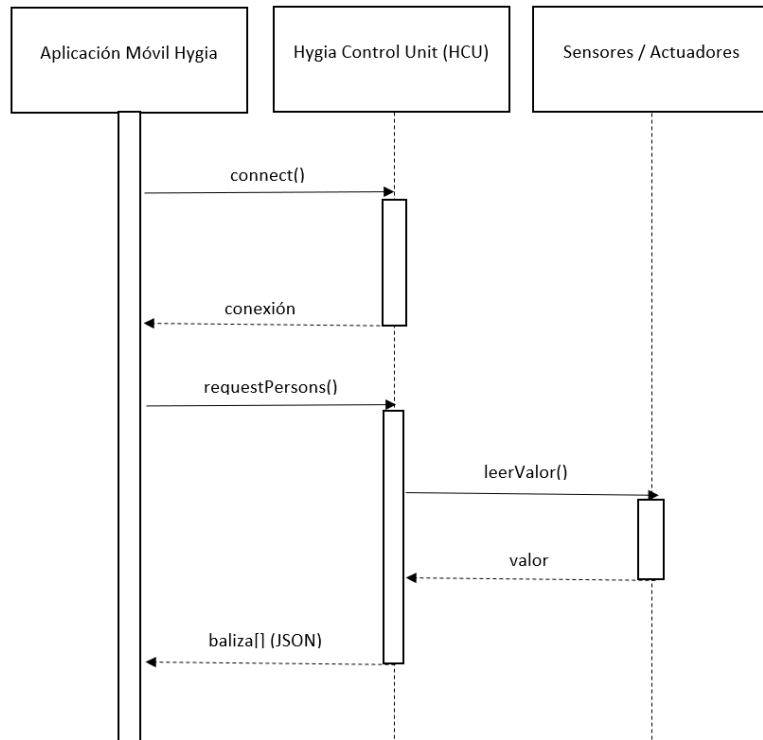


Diagrama 6. Secuencia inicio conexión App - HCU

- Seguidamente, la aplicación móvil se queda a la espera asíncrona de recepción de información del HCU. A petición del usuario, la app enviará al HCU órdenes de actuación:

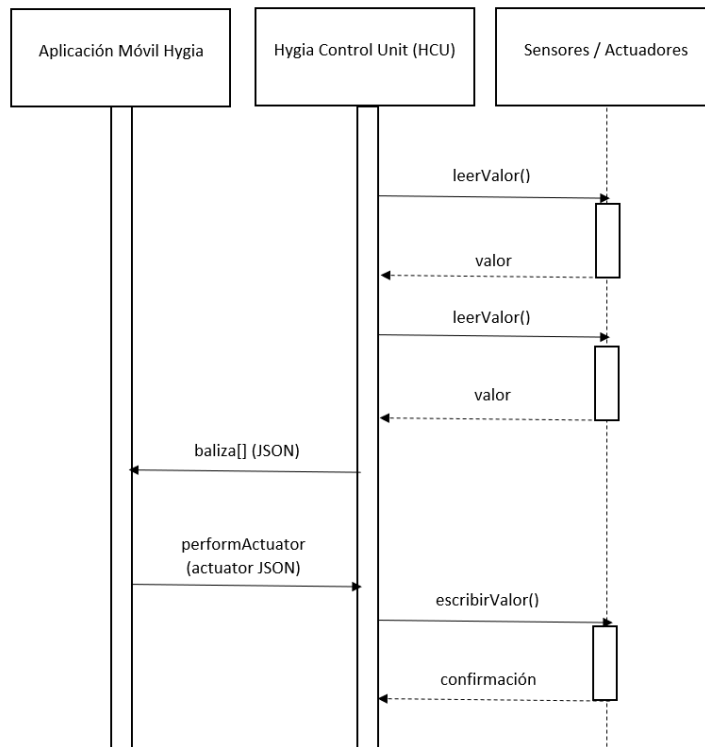


Diagrama 7. Secuencia comunicación App - HCU

2. Construcción del entorno para simulación de equipamiento hardware de la solución

2.1 Componentes utilizados en la construcción del entorno

Se han utilizado los siguientes componentes en la implementación del entorno de simulación hardware:

- Placa ARDUINO UNO REV3
- Módulo Bluetooth HC-05 DSD TECH
- 2 proto board
- 3 diodos led
- 3 pulsadores de botón
- 1 buzzer
- Cableado

2.2 Entorno hardware implementado

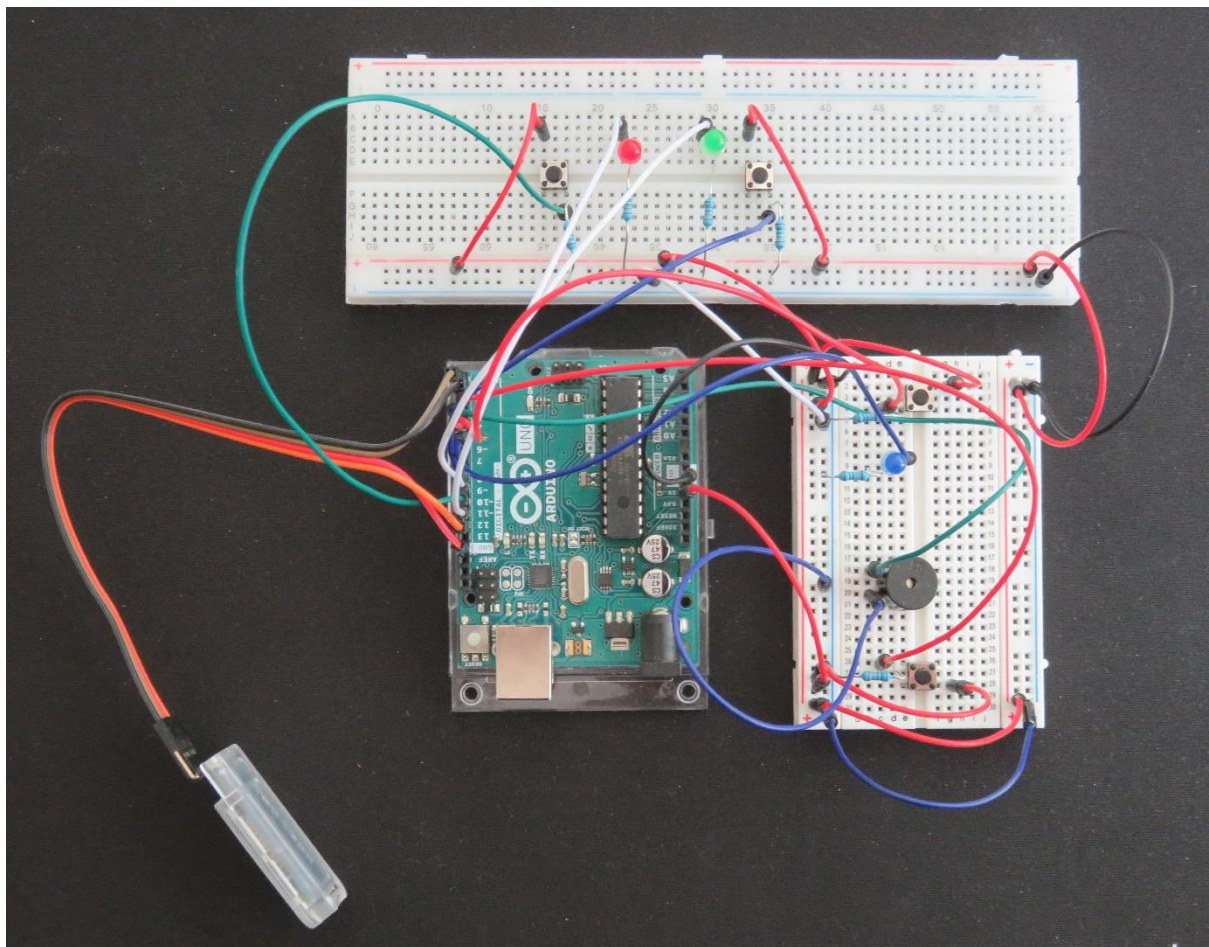


Foto 1. Entorno de simulación hardware

2.3 Programación Arduino

Se ha desarrollado y cargado en la placa Arduino un programa (en lenguaje C) con las siguientes responsabilidades:

- Gestión del dispositivo de comunicación Bluetooth HC-05
- Lectura de sensores
- Gestión de actuadores
- Comunicación con aplicación móvil

El programa desarrollado se estructura en dos grandes funciones:

- *setup()*: Función de inicialización (se ejecuta una única vez al iniciarse la placa). Acciones realizadas:
 - configuración pines entrada/salida según el diseño del entorno (habitación 1- sensor 1, habitación 2 – sensor 2, etc.).
 - configuración del dispositivo de comunicación bluetooth HC-05 mediante comandos Hayes (o AT) a través del puerto serie de la placa: velocidad de trabajo, nombre de dispositivo, pin de seguridad, etc.
- *loop()*: rutina principal que se ejecuta en bucle, mientras la placa reciba alimentación, después de la ejecución correcta de la función de inicialización *setup()*. Acciones:
 - activación/desactivación de actuadores según la información recibida de la app
 - escaneo de balizas de sensores y envío de información de estado a la app
 - escaneo de balizas de personas y envío de información de estado a la app

Para el intercambio de datos con la aplicación móvil, a través de Bluetooth, se empleará el formato JSON. Se utilizará un carácter finalizador de mensaje (!).

Ejemplo de datos enviados desde la placa Arduino hacia la aplicación móvil tras la activación de un sensor:

```
{"uuid": "2d39492f-d0ff-4fc1-9e30-2e2e8a33e5c8", "value": "1", "time": "11/04/2020 17:47", "lon": "-3.703445", "lat": "40.417264", "lName": "Planta 1", "lDesc": "Hab. 1"}!
```

Ejemplo de datos enviados desde la aplicación móvil a la placa Arduino para la activación de un actuador:

```
{"uuid": "90dff833-829a-4b58-9259-9a2440e7412a", "value": "1"}!
```

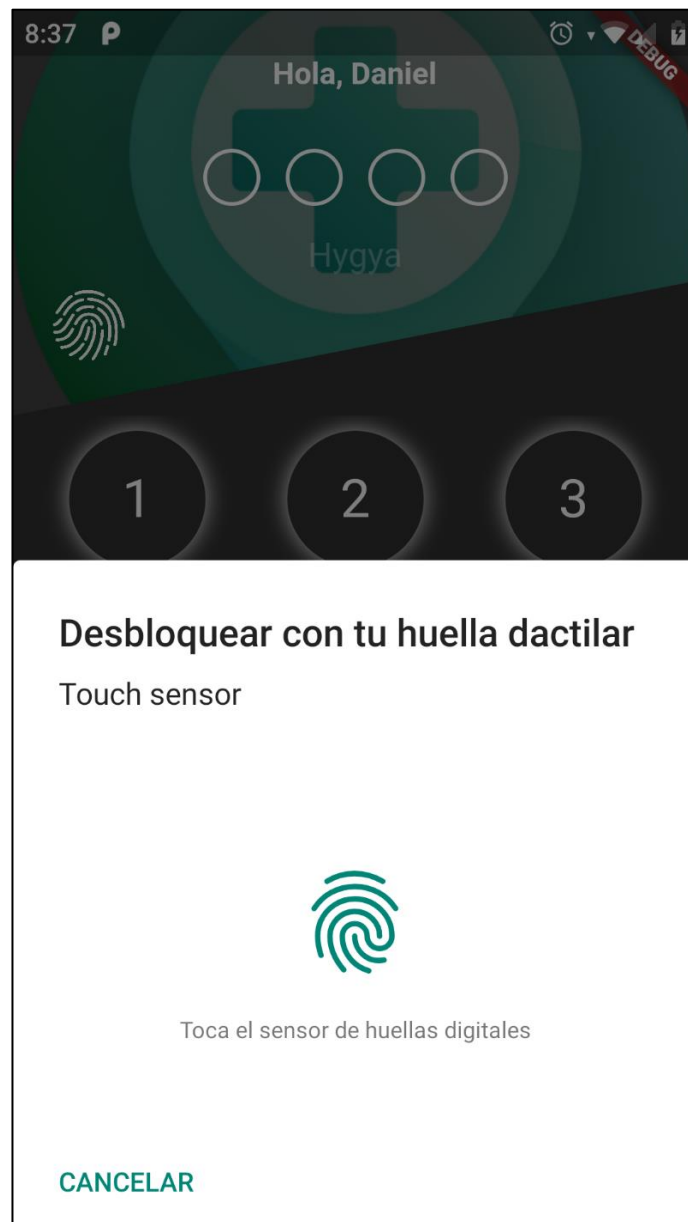
Capítulo 5: Evidencias de prueba y demo

1. Épica: Acceder a la aplicación

La aplicación permite el acceso biométrico mediante huella digital o la introducción de un pin personal de seguridad de cuatro dígitos por teclado.

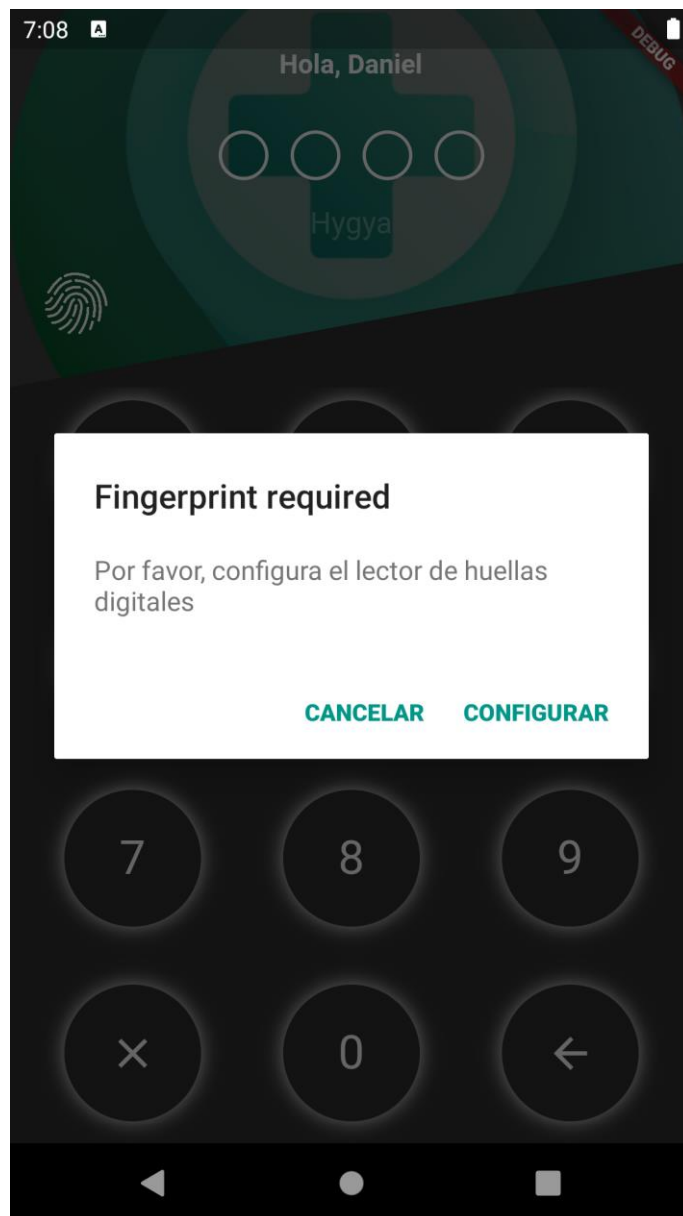
1.1 Historia de usuario: Acceso biométrico: huella digital

Al abrir la aplicación se muestra por defecto el acceso por huella digital:



Captura 9. Acceso biométrico: huella digital – inicio (dispositivo real Xiaomi A1. Android 9)

Si el sensor de huellas no está disponible en el dispositivo, se muestra un mensaje informativo:



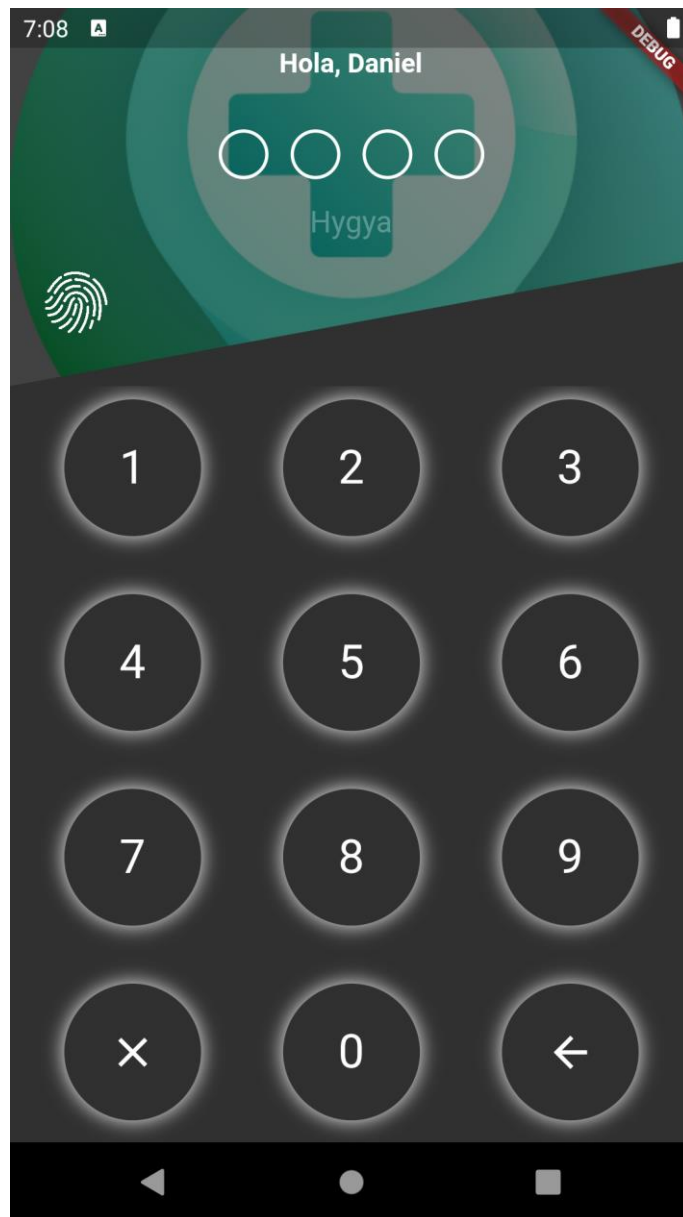
Captura 10. Acceso biométrico: huella digital – no disponible (emulación Nexus 5X API 29 x86)

En cualquier momento puede pulsarse la opción CANCELAR para acceder a la autenticación con pin de seguridad por teclado.

Si se introduce una huella digital correcta se realizará la navegación a la página de trabajo por defecto de la aplicación: listado de alertas asistenciales.

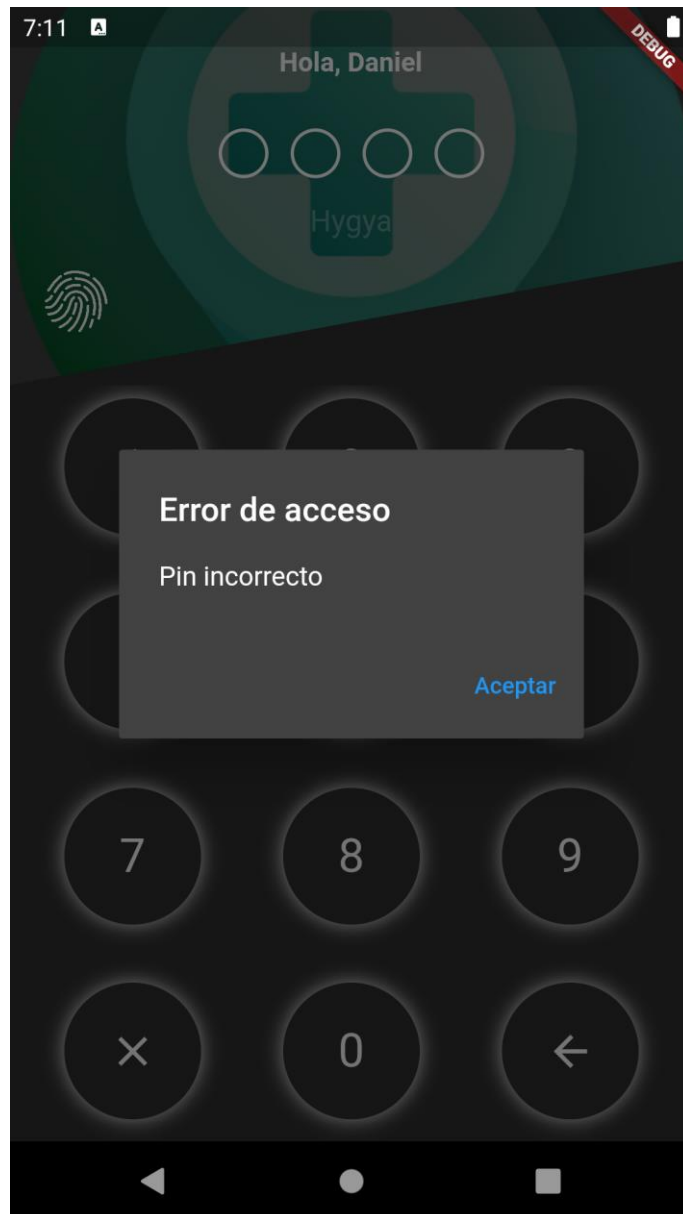
1.2 Historia de usuario: Acceso con pin de seguridad por teclado

El pin establecido para la realización de pruebas es: 2020.



Captura 11. Acceso con pin de seguridad por teclado - inicio (emulación Nexus 5X API 29 x86)

Si se introduce un pin de acceso correcto se accede a la aplicación. En caso contrario se muestra un mensaje de error:



Captura 12. Acceso con pin de seguridad por teclado - error (emulación Nexus 5X API 29 x86)

En cualquier momento puede accederse a la validación biométrica pulsando el icono ubicado en la parte superior izquierda de la pantalla:

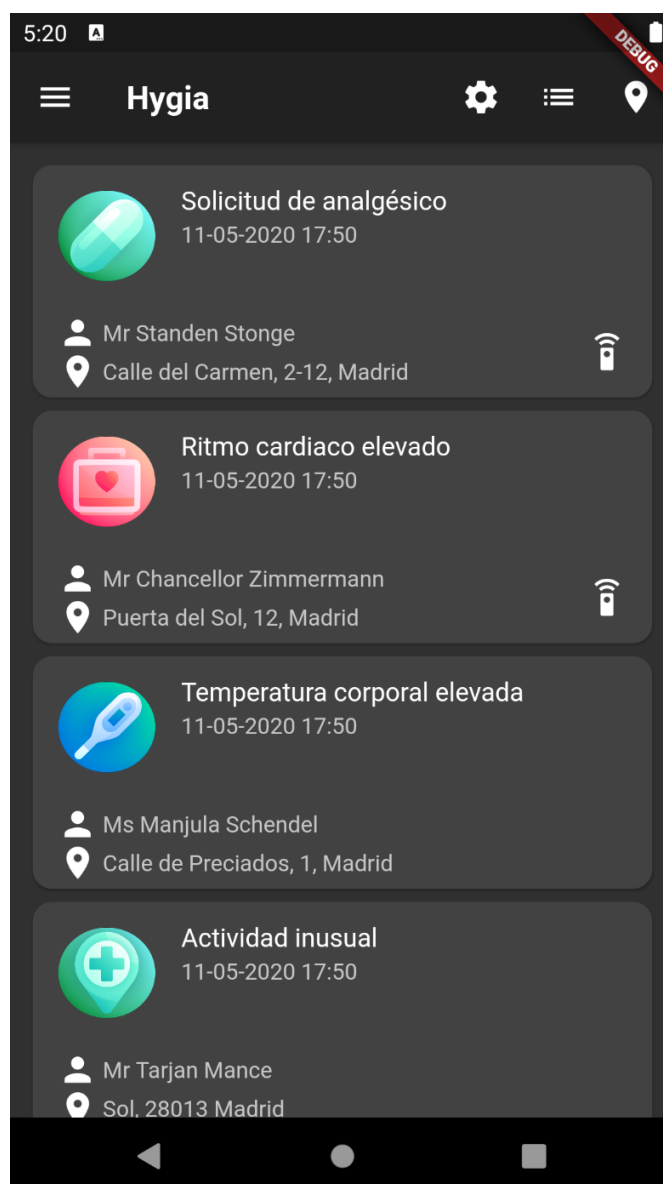


2. Épica: Consultar alertas asistenciales

2.1 Historia de usuario: Consultar el listado de alertas asistenciales

Por defecto, la aplicación muestra el espacio de trabajo de alertas asistenciales, en el cual se presenta el conjunto de alertas pendientes de atender por parte del usuario Asistente. La aplicación está configurada para recuperar por defecto las alertas asistenciales en un ámbito de red inalámbrica de área extensa. En este caso recupera las alertas devueltas por el API REST remoto que en este proyecto simulamos con la herramienta JSONPlaceholder:

<http://my-json-server.typicode.com/daniuoc/hygiaapi/alerts>).



Captura 13. Consultar el listado de alertas asistenciales (emulación Nexus 5X API 29 x86)

Las alertas recuperadas se muestran representadas mediante tarjetas con su información correspondiente.

2.2 Historia de usuario: Consultar el mapa de alertas asistenciales

A través del botón correspondiente de la barra superior de herramientas puede accederse a la vista de mapa:

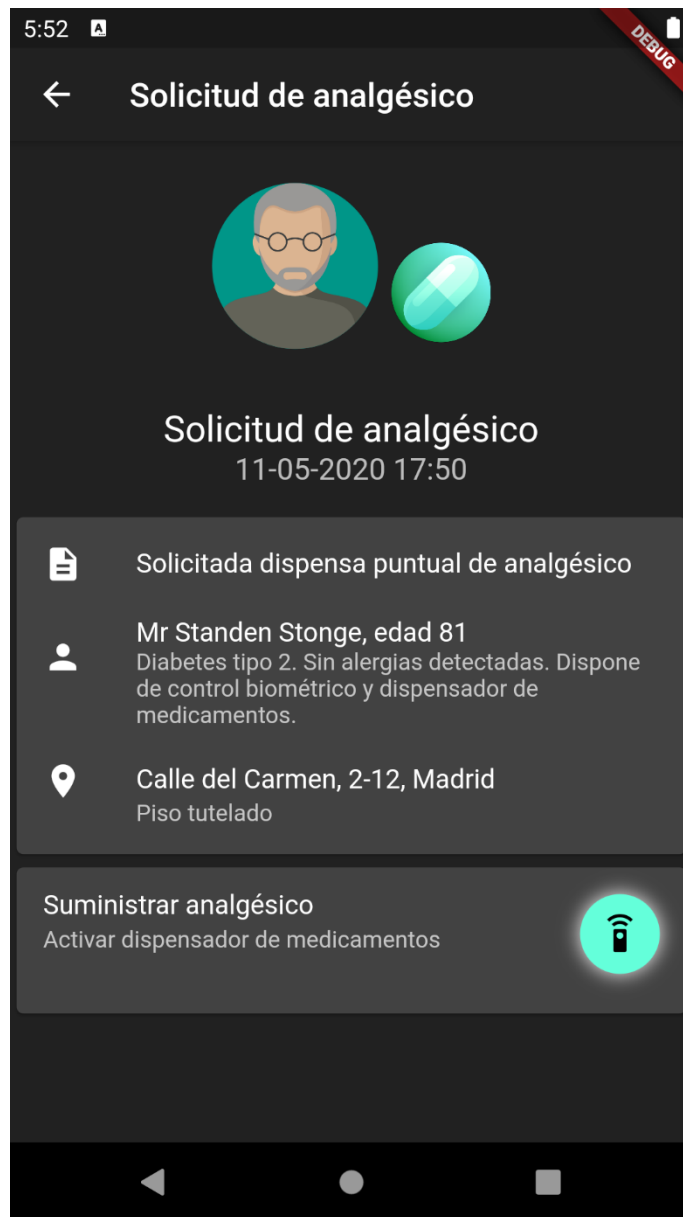


Captura 14. Consultar el mapa de alertas asistenciales (emulación Nexus 5X API 29 x86)

Las alertas en este caso aparecen geoposicionadas y contextualizadas con una descripción que incluye el mensaje de la alerta y la persona asociada.

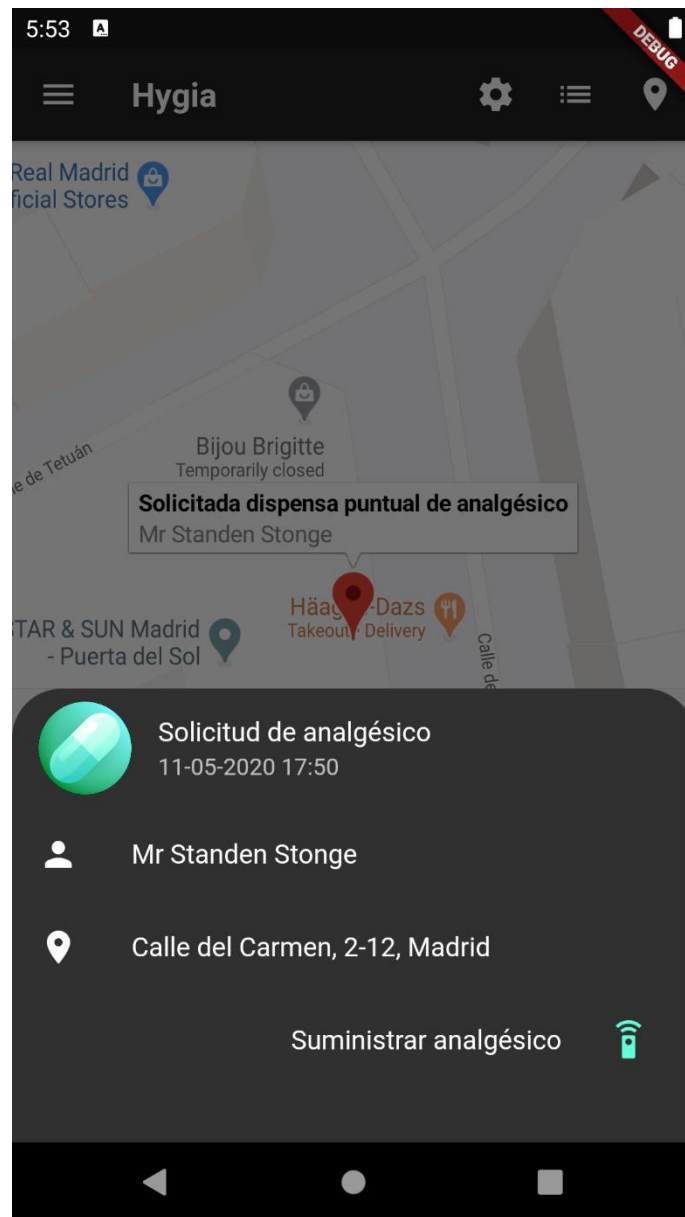
2.3 Historia de usuario: Consultar el detalle de las alertas asistenciales

Al pulsar en una de las tarjetas del listado de alertas se muestra su correspondiente pantalla de detalle:



Captura 15. Consultar el detalle de las alertas asistenciales - listado (emulación Nexus 5X API 29 x86)

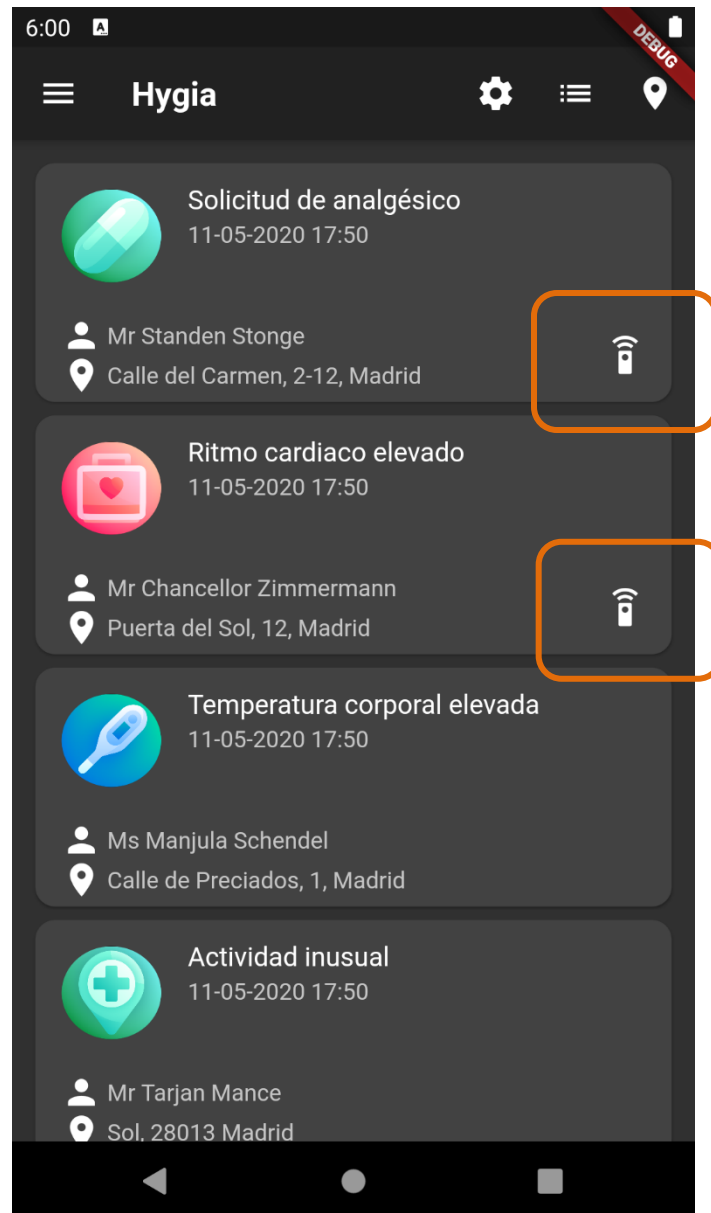
También puede accederse al detalle de la alerta desde la vista de mapa, pulsando sobre el marcador correspondiente:



Captura 16. Consultar el detalle de las alertas asistenciales - mapa (emulación Nexus 5X API 29 x86)

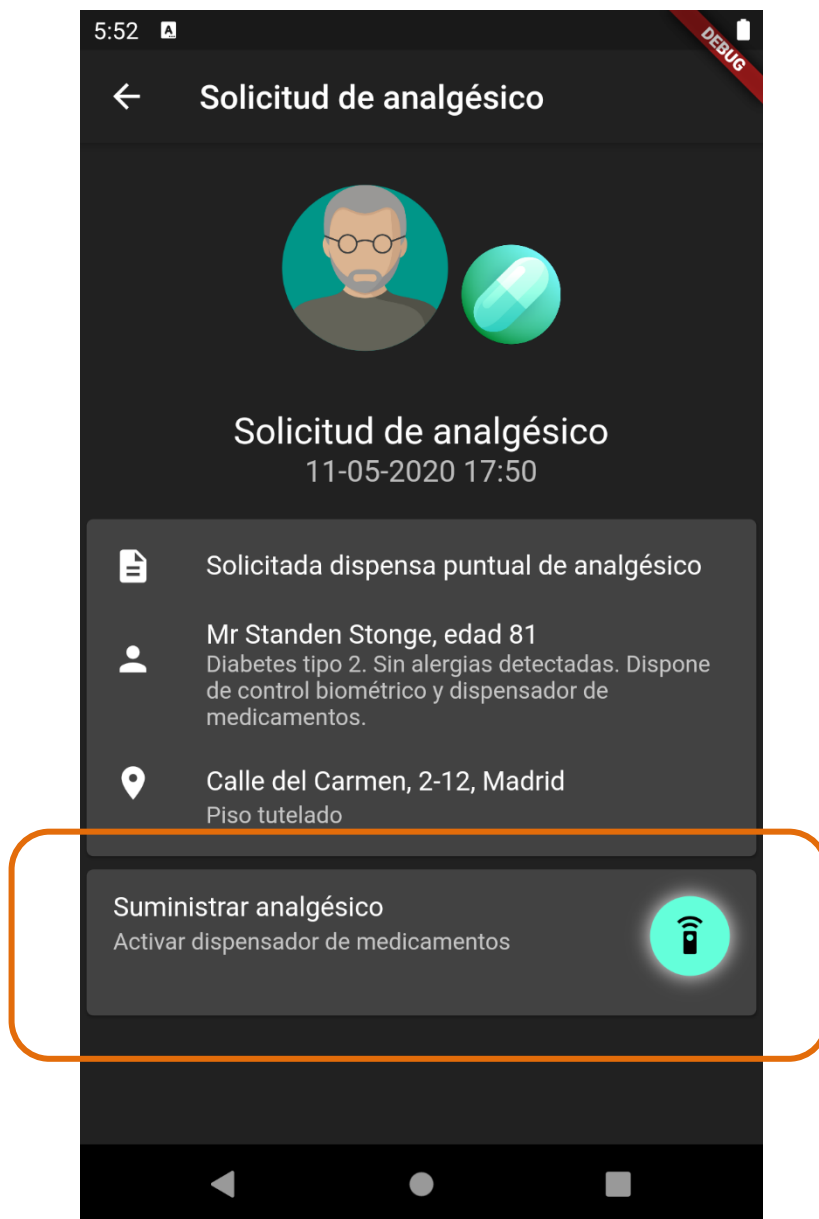
2.4 Historia de usuario: Consultar las acciones asistenciales asociadas a las alertas asistenciales

En las tarjetas de las alertas que tienen acciones asistenciales disponibles se muestra un icono informativo:



Captura 17. Consultar las acciones asistenciales asociadas a las alertas asistenciales: icono listado (emulación Nexus 5X API 29 x86)

En la pantalla de detalle de las alertas que tienen acciones asociadas se muestra la sección correspondiente con la información de la acción y el botón para ejecutar la acción remota:

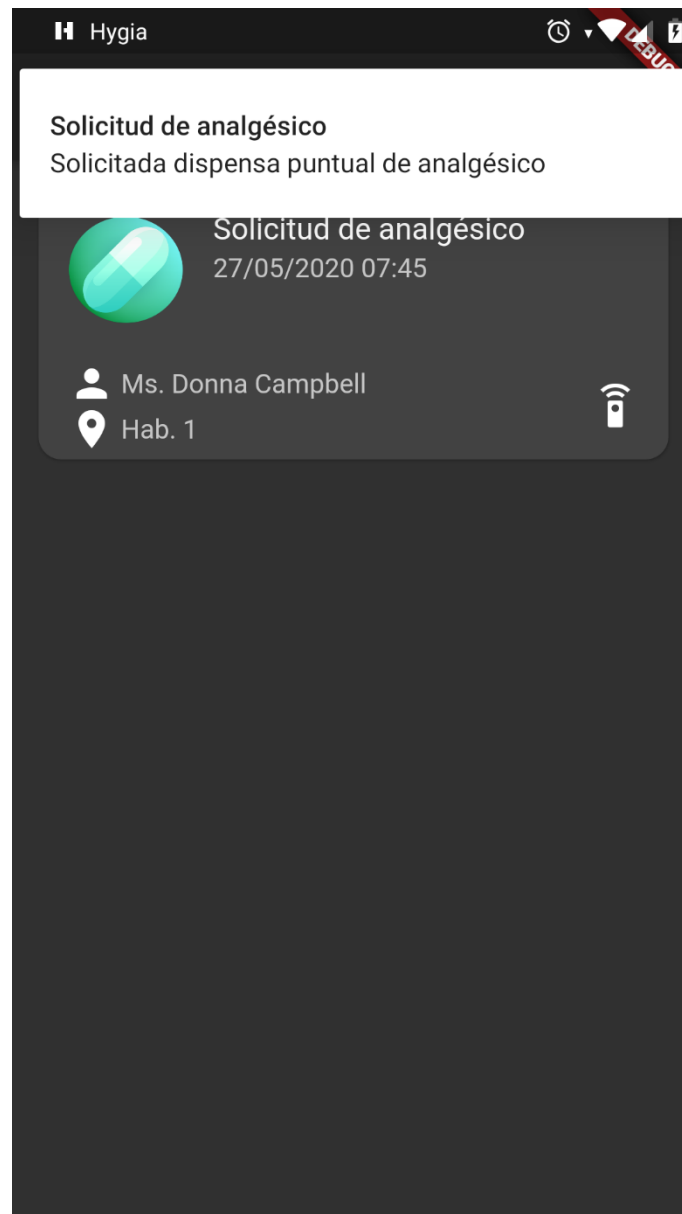


Captura 18. Consultar las acciones asistenciales asociadas a las alertas asistenciales: sección detalle (emulación Nexus 5X API 29 x86)

Al pulsar el botón la aplicación envía la orden de ejecución de acción al dispositivo hardware, utilizando la tecnología del ámbito de utilización correspondiente (API REST / Bluetooth).

2.5 Historia de usuario: Recibir notificaciones de las alertas asistenciales

Cada vez que se genera una alerta asistencial se muestra una notificación en el dispositivo



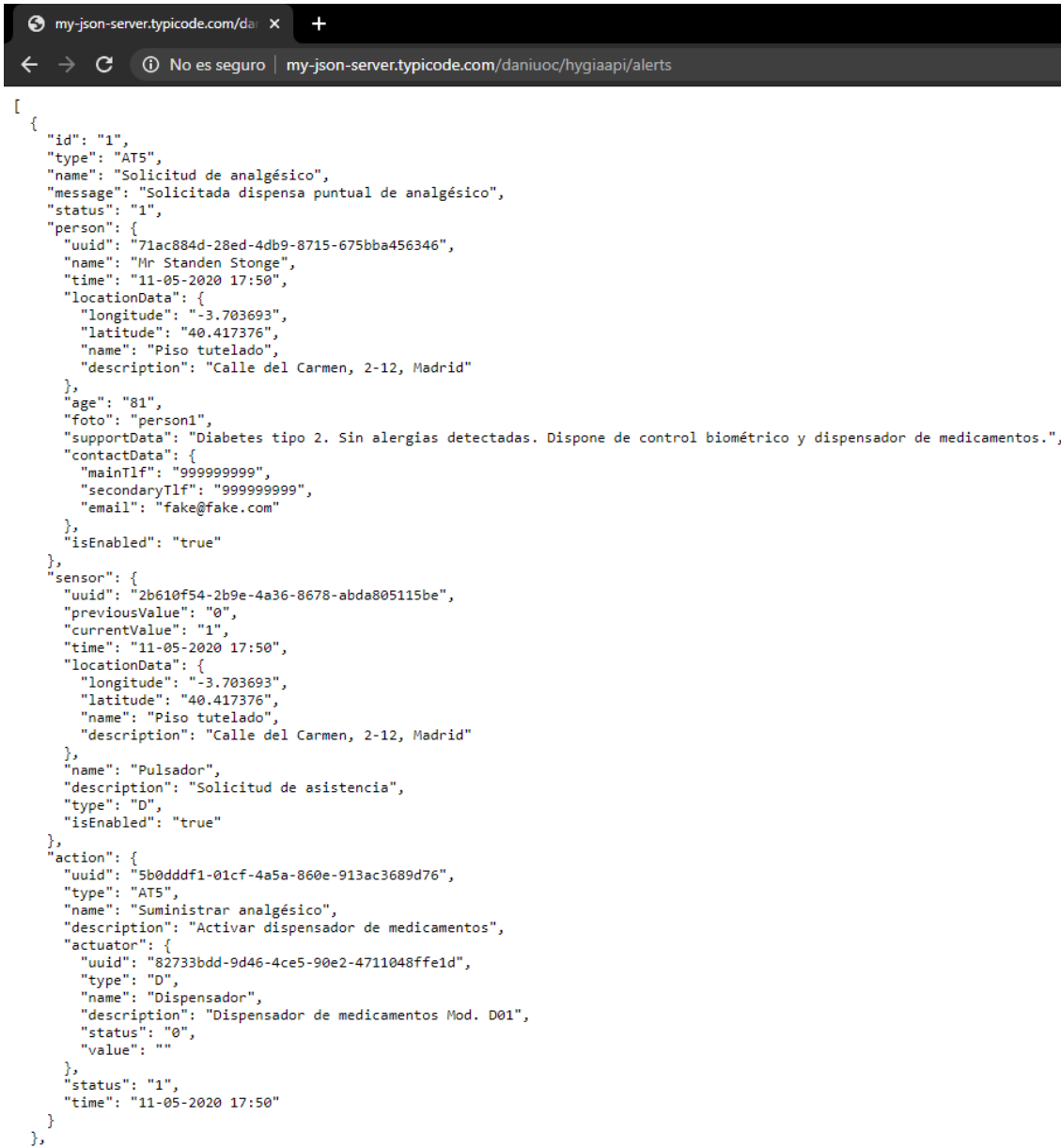
Captura 19. Recibir notificaciones de las alertas asistenciales (dispositivo real Xiaomi A1. Android 9)

2.6 Historia de usuario: Recuperar alertas asistenciales en un ámbito de red inalámbrica de área extensa (WWAN)

Las peticiones realizadas por la aplicación en el ámbito de red inalámbrica de área extensa, realizadas a través de HTTP, devuelven un listado de alertas asistenciales en formato JSON:

<http://my-json-server.typicode.com/daniuoc/hygiaapi/alerts>

Ejemplo:



```
[
  {
    "id": "1",
    "type": "AT5",
    "name": "Solicitud de analgésico",
    "message": "Solicitada dispensa puntual de analgésico",
    "status": "1",
    "person": {
      "uuid": "71ac884d-28ed-4db9-8715-675bba456346",
      "name": "Mr Standen Stonge",
      "time": "11-05-2020 17:50",
      "locationData": {
        "longitude": "-3.703693",
        "latitude": "40.417376",
        "name": "Piso tutelado",
        "description": "Calle del Carmen, 2-12, Madrid"
      },
      "age": "81",
      "foto": "person1",
      "supportData": "Diabetes tipo 2. Sin alergias detectadas. Dispone de control biométrico y dispensador de medicamentos.",
      "contactData": {
        "mainTlf": "999999999",
        "secondaryTlf": "999999999",
        "email": "fake@fake.com"
      },
      "isEnabled": "true"
    },
    "sensor": {
      "uuid": "2b610f54-2b9e-4a36-8678-abda805115be",
      "previousValue": "0",
      "currentValue": "1",
      "time": "11-05-2020 17:50",
      "locationData": {
        "longitude": "-3.703693",
        "latitude": "40.417376",
        "name": "Piso tutelado",
        "description": "Calle del Carmen, 2-12, Madrid"
      },
      "name": "Pulsador",
      "description": "Solicitud de asistencia",
      "type": "D",
      "isEnabled": "true"
    },
    "action": {
      "uuid": "5b0dddf1-01cf-4a5a-860e-913ac3689d76",
      "type": "AT5",
      "name": "Suministrar analgésico",
      "description": "Activar dispensador de medicamentos",
      "actuator": {
        "uuid": "82733bdd-9d46-4ce5-90e2-4711048ffe1d",
        "type": "D",
        "name": "Dispensador",
        "description": "Dispensador de medicamentos Mod. D01",
        "status": "0",
        "value": ""
      },
      "status": "1",
      "time": "11-05-2020 17:50"
    }
  }
]
```

Captura 20. Respuesta a petición API REST mock configurada con JSONPlaceholder

```
4  
5 /// [hygia APP] api_data_context.dart  
6 /// Contexto de datos para integración con API REST  
7  
8 class ApiDataContext {  
9   Future<dynamic> getData(endPoint) async {  
10    var response = await http.get(endPoint);  
11    if (response.statusCode == 200) {  
12      var jsonResponse = response.body;  
13      print('[ApiDataContext][getData] Response: ' + jsonResponse);  
14      return jsonResponse;  
15    } else {  
16      print('[ApiDataContext][getData] Request failed with status: ${response.statusCode}');  
17      return null;  
18    }  
19  }  
20 }
```

```
I/flutter (18227): [CustomBlocDelegate][onEvent] FetchAlerts  
I/flutter (18227): [CustomBlocDelegate][onEvent] FetchPersons  
I/flutter (18227): [CustomBlocDelegate][onTransition] Transition { currentState: AlertsEmpty, event: FetchAlerts, nextState: AlertsLoading }  
I/flutter (18227): [CustomBlocDelegate][onTransition] Transition { currentState: PersonsEmpty, event: FetchPersons, nextState: PersonsLoading }  
I/flutter (18227): [ApiDataContext][getData] Response: [  
I/flutter (18227): {  
I/flutter (18227):   "uuid": "71ac884d-28ed-4db9-8715-675bba456346",  
I/flutter (18227):   "name": "Mr Standen Stonge",  
I/flutter (18227):   "time": "11-05-2020 17:50",  
I/flutter (18227):   "locationData": {  
I/flutter (18227):     "longitude": "-3.703693",  
I/flutter (18227):     "latitude": "40.417376",  
I/flutter (18227):     "name": "Piso tutelado",  
I/flutter (18227):     "description": "Calle del Carmen, 2-12, Madrid"  
I/flutter (18227):   },  
I/flutter (18227):   "age": "81",  
I/flutter (18227):   "foto": "person1",  
I/flutter (18227):   "supportData": "Diabetes tipo 2. Sin alergias detectadas. Dispone de control biométrico y dispensador de medicamentos.",  
I/flutter (18227):   "contactData": {  
I/flutter (18227):     "mainTlf": "999999999",  
I/flutter (18227):     "secondaryTlf": "999999999",  
I/flutter (18227):     "email": "fake@fake.com"  
I/flutter (18227):   },  
I/flutter (18227):   "isEnabled": "true",  
I/flutter (18227):   "indicators": [  
I/flutter (18227):     ]  
I/flutter (18227): }
```

Captura 21. Ejecución Debug Android Studio estrategia adquisición WWMA (API remota REST HTTP)

2.7 Historia de usuario: Recuperar alertas asistenciales en un ámbito de red inalámbrica de área personal (WPAN)

La aplicación recupera correctamente por *bluetooth* la información enviada por el dispositivo *hardware* Hygia Control Unit de la solución, simulado en este proyecto con el entorno hardware construido sobre la placa Arduino:

```
43 List<Alert> alerts = List<Alert>();
44 List<LogicPerson> logicPersons = List<LogicPerson>();
45
46 BluetoothDataContext._fromConnection(this._connection) {
47   print("[BluetoothDataContext][_fromConnection] inicio");
48
49   _connection.input.listen((data) {
50     print("[BluetoothDataContext][listen] inicio escucha eventos");
51
52     String dataDecoded = ascii.decode(data);
53     print("[BluetoothDataContext][listen] Evento recibido: " + dataDecoded);
54
55     _buffer += dataDecoded;
56     print("[BluetoothDataContext][listen] Contenido actual del buffer: " +
57       _buffer);
58
59     if (dataDecoded.contains("!")) {
60       // Encontramos terminador !, por tanto, hemos leído un mensaje completo
61       if (_buffer[0] == 'S') {
62         // Hemos recibido información de una baliza de sensor. Serializamos
63         _buffer = _buffer.substring(1, _buffer.length - 1);
64         print("[BluetoothDataContext][listen] Baliza de sensor recibida: " +
65           _buffer);
66         beaconModel = BeaconModel.fromJson(jsonDecode(_buffer));
67         beaconModel.time = DateTimeFormat.format(DateTime.now());
68
69         // Generamos la alerta asistencial correspondiente al sensor recibido
70         Alert alert = _dataHDIRepository.getAlert(beaconModel);
71         alerts.add(alert);
```

Run: main

Console

```
I/flutter (1326): [BluetoothDataContext][listen] Inicio escucha eventos
I/flutter (1326): [BluetoothDataContext][listen] Evento recibido: 8a33e5c8,"value": "1","ti
I/flutter (1326): [BluetoothDataContext][listen] Contenido actual del buffer: S{"uuid": "2d39492f-d0ff-4fc1-9e30-2e2e8a33e5c8","value": "1","ti
I/flutter (1326): [BluetoothDataContext][listen] Inicio escucha eventos
I/flutter (1326): [BluetoothDataContext][listen] Evento recibido: me": "11/04/2020 17:47","lon": "-3.7034
I/flutter (1326): [BluetoothDataContext][listen] Contenido actual del buffer: S{"uuid": "2d39492f-d0ff-4fc1-9e30-2e2e8a33e5c8","value": "1","time": "11/04/2020 17:47","lon"
I/flutter (1326): [BluetoothDataContext][listen] Inicio escucha eventos
I/flutter (1326): [BluetoothDataContext][listen] Evento recibido: 45","lat": "40.417264","l
I/flutter (1326): [BluetoothDataContext][listen] Contenido actual del buffer: S{"uuid": "2d39492f-d0ff-4fc1-9e30-2e2e8a33e5c8","value": "1","time": "11/04/2020 17:47","lon"
I/flutter (1326): [BluetoothDataContext][listen] Inicio escucha eventos
I/flutter (1326): [BluetoothDataContext][listen] Evento recibido: Name": "Planta 1","IDesc": "Hab. 1"}!
I/flutter (1326): [BluetoothDataContext][listen] Contenido actual del buffer: S{"uuid": "2d39492f-d0ff-4fc1-9e30-2e2e8a33e5c8","value": "1","time": "11/04/2020 17:47","lon"
I/flutter (1326): [BluetoothDataContext][listen] Baliza de sensor recibida: {"uuid": "2d39492f-d0ff-4fc1-9e30-2e2e8a33e5c8","value": "1","time": "11/04/2020 17:47","lon": "
```

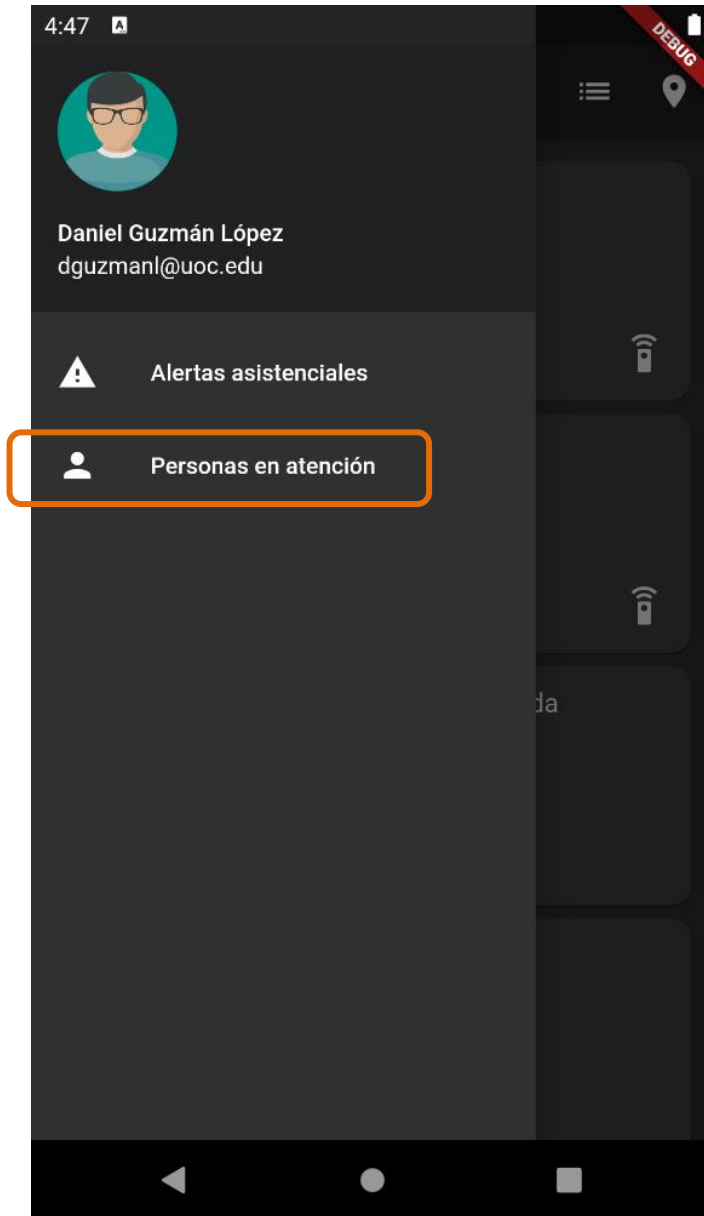
daemon started successfully (33 minutes ago)

Captura 22. Ejecución Debug Android Studio estrategia adquisición WPMA (bluetooth)

3. Épica: Consultar personas en atención

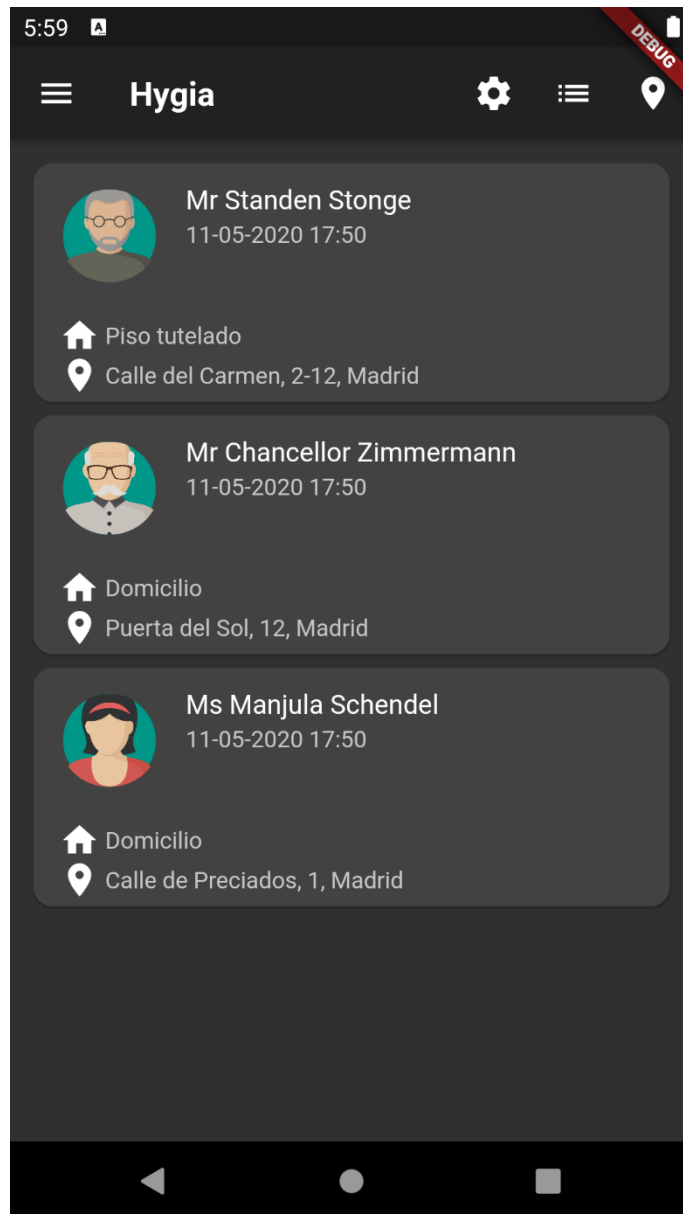
3.1 Historia de usuario: Consultar el listado de personas en atención

Puede accederse al espacio de trabajo de personas en atención a través del menú lateral:



Captura 23. Consultar el listado de personas en atención - menú (emulación Nexus 5X API 29 x86)

La aplicación recupera las personas en atención según el ámbito de utilización seleccionado:

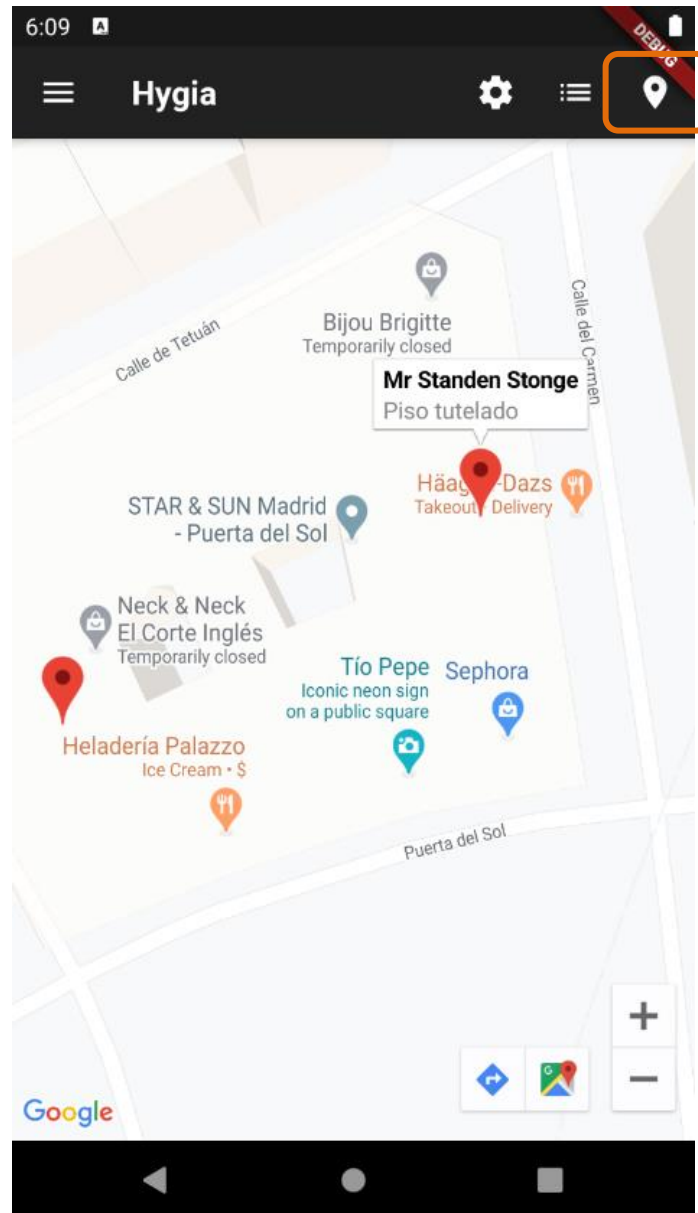


Captura 24. Consultar el listado de personas en atención (emulación Nexus 5X API 29 x86)

Las personas en atención se muestran representadas mediante tarjetas con su información correspondiente.

3.2 Historia de usuario: Consultar el mapa de personas en atención

A través del botón correspondiente de la barra superior de herramientas puede accederse a la vista de mapa:

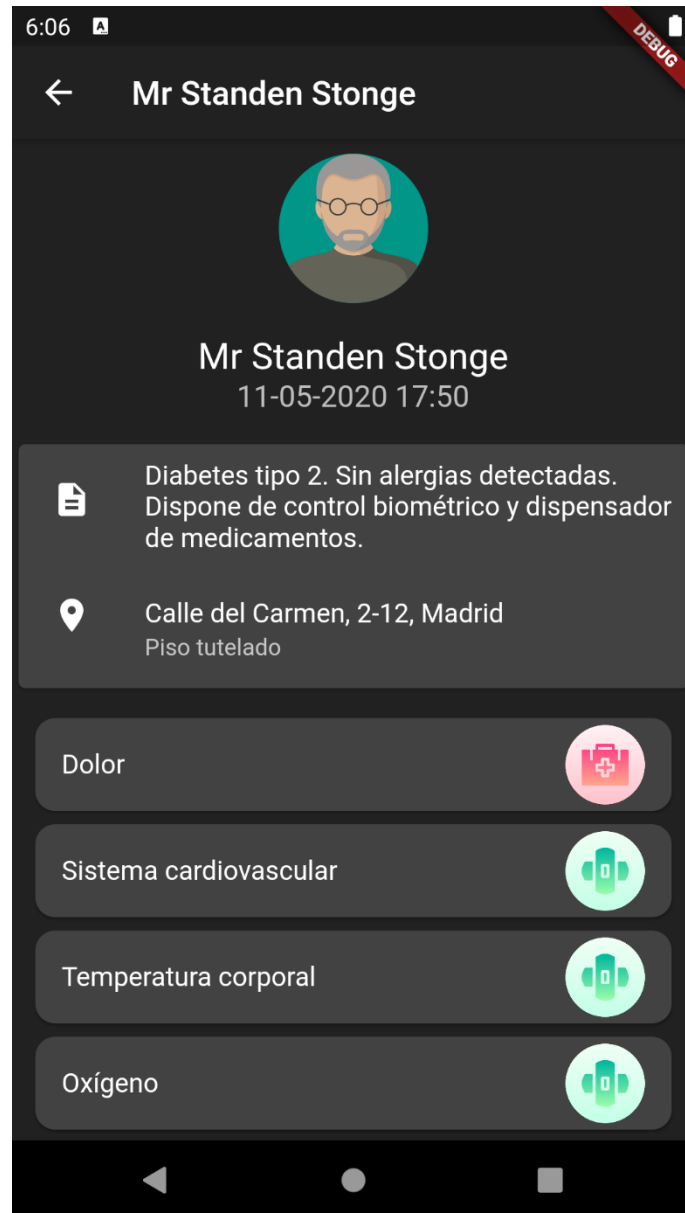


Captura 25. Consultar el mapa de personas en atención (emulación Nexus 5X API 29 x86)

Las personas en atención en este caso aparecen geopositionadas y contextualizadas con una descripción que incluye el nombre de la persona y la descripción de su residencia.

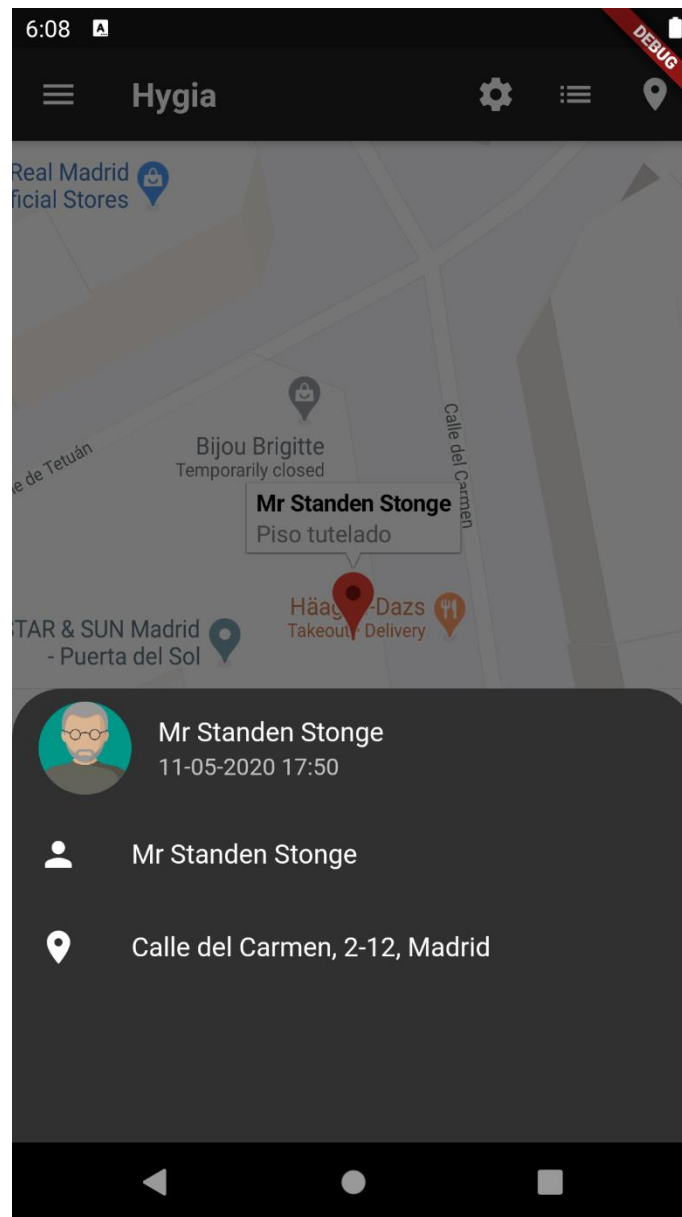
3.3 Historia de usuario: Consultar la información de detalle de las personas en atención

Al pulsar en una de las tarjetas del listado de personas en atención se muestra su correspondiente pantalla de detalle:



Captura 26. Consultar la información de detalle de las personas en atención - listado (emulación Nexus 5X API 29 x86)

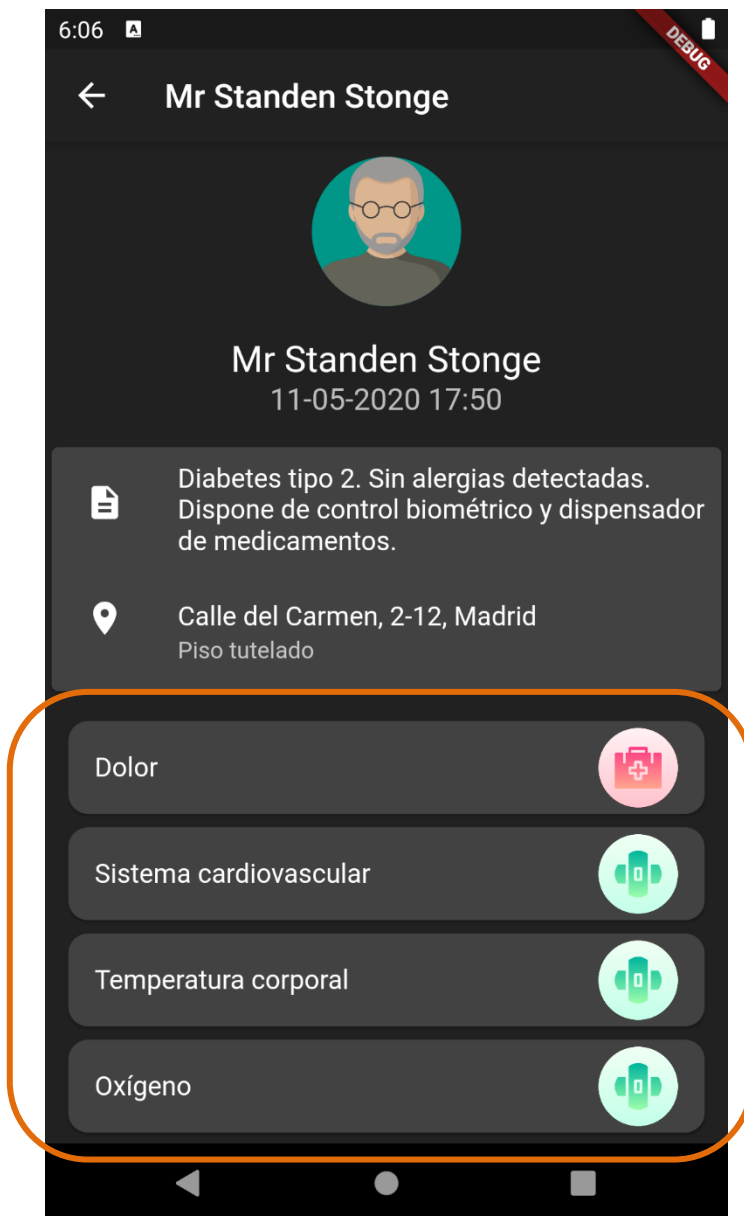
También puede accederse al detalle de la persona desde la vista de mapa, pulsando sobre el marcador correspondiente:



Captura 27. Consultar la información de detalle de las personas en atención - mapa (emulación Nexus 5X API 29 x86)

3.4 Historia de usuario: Consultar los indicadores asistenciales disponibles para las personas en atención

En la pantalla de detalle de las personas en atención se muestran los indicadores asistenciales asociados y su estado:



Captura 28. Consultar los indicadores asistenciales disponibles para las personas en atención (emulación Nexus 5X API 29 x86)

El significado de los iconos es el siguiente:

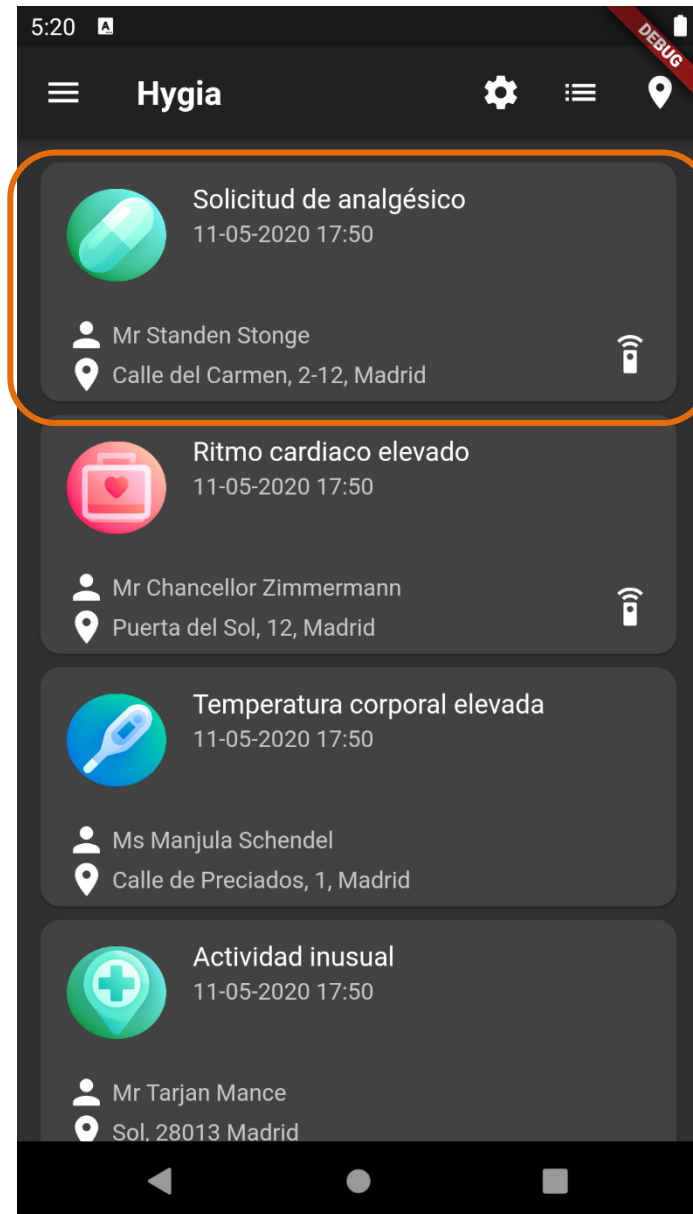


Indicador asistencial estable. No se han producido situaciones que requieran atención por parte del asistente.



Indicador asistencial activado. Se han producido alertas asistenciales relacionadas que han activado el indicador y se requiere por tanto asistencia por parte del asistente. Las alertas relacionadas pueden consultarse desde el espacio de trabajo de alertas asistenciales.

En este caso concreto, el estado del indicador de 'Dolor' la persona 'Mr. Standen' indica al Asistente que la situación de la persona no es óptima y, por tanto, debería ser asistido. Este indicador ha sido activado porque la persona ha solicitado un analgésico y, por tanto, se ha generado la alerta asistencial correspondiente. Dicha alerta efectivamente está presente en el panel de alertas:

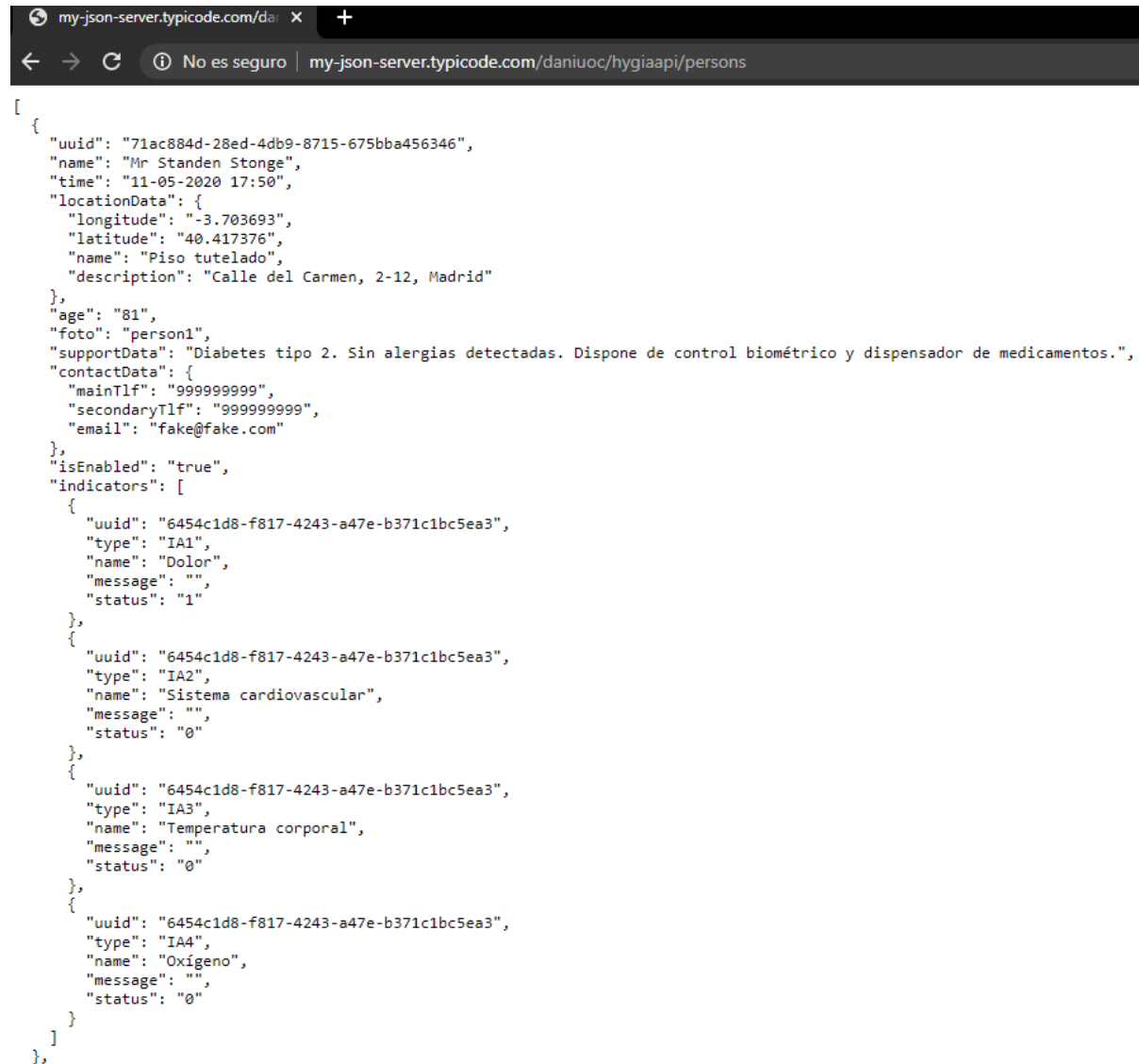


3.5 Historia de usuario: Recuperar personas en atención en un ámbito de red inalámbrica de área extensa (WWAN)

Las peticiones realizadas por la aplicación en el ámbito de red inalámbrica de área extensa, realizadas a través de HTTP, devuelven un listado de personas en atención en formato JSON:

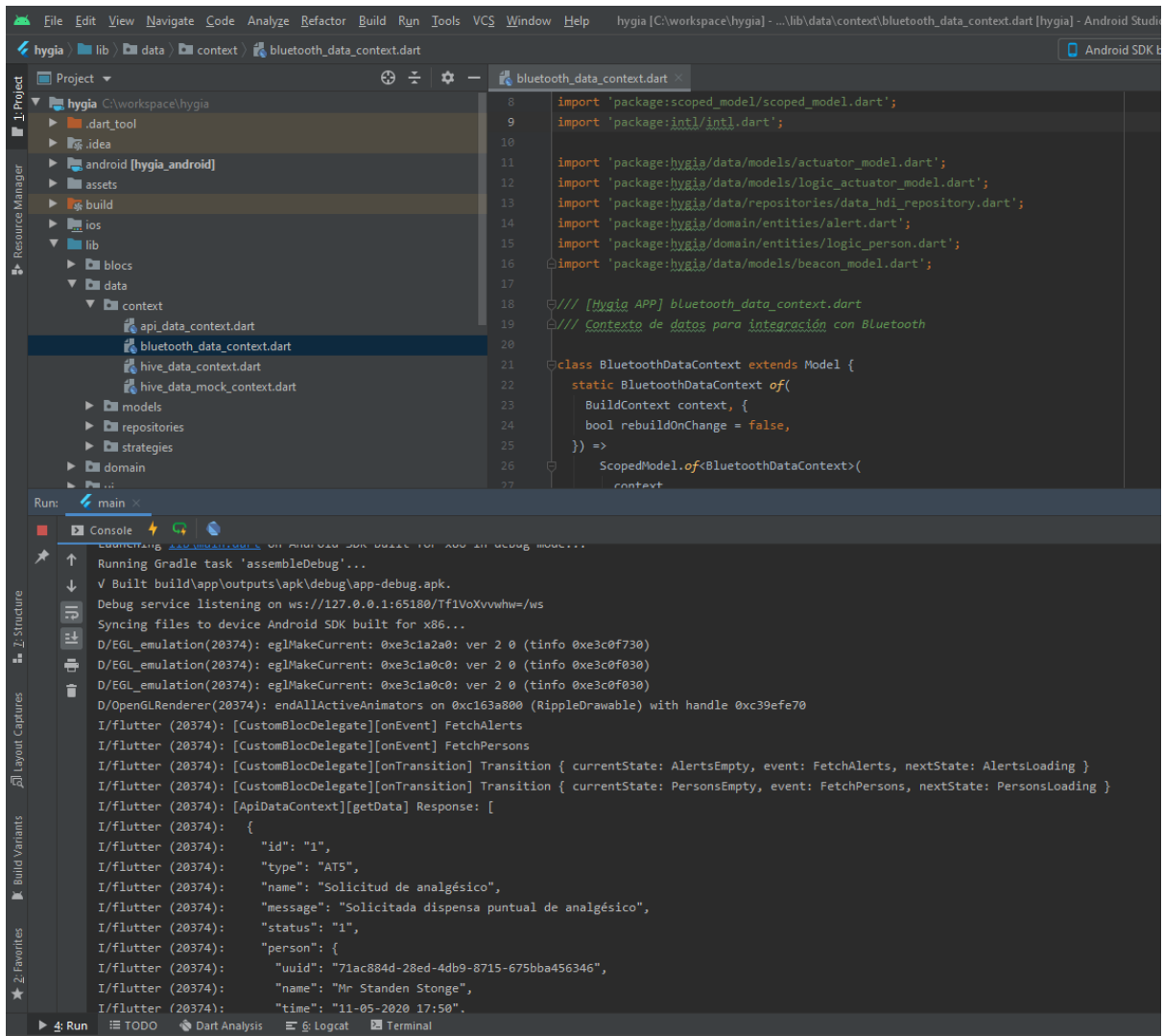
<http://my-json-server.typicode.com/daniuoc/hygiaapi/persons>

Ejemplo:



```
[
  {
    "uuid": "71ac884d-28ed-4db9-8715-675bba456346",
    "name": "Mr Standen Stonge",
    "time": "11-05-2020 17:50",
    "locationData": {
      "longitude": "-3.703693",
      "latitude": "40.417376",
      "name": "Piso tutelado",
      "description": "Calle del Carmen, 2-12, Madrid"
    },
    "age": "81",
    "foto": "person1",
    "supportData": "Diabetes tipo 2. Sin alergias detectadas. Dispone de control biométrico y dispensador de medicamentos.",
    "contactData": {
      "mainTlf": "999999999",
      "secondaryTlf": "999999999",
      "email": "fake@fake.com"
    },
    "isEnabled": "true",
    "indicators": [
      {
        "uuid": "6454c1d8-f817-4243-a47e-b371c1bc5ea3",
        "type": "IA1",
        "name": "Dolor",
        "message": "",
        "status": "1"
      },
      {
        "uuid": "6454c1d8-f817-4243-a47e-b371c1bc5ea3",
        "type": "IA2",
        "name": "Sistema cardiovascular",
        "message": "",
        "status": "0"
      },
      {
        "uuid": "6454c1d8-f817-4243-a47e-b371c1bc5ea3",
        "type": "IA3",
        "name": "Temperatura corporal",
        "message": "",
        "status": "0"
      },
      {
        "uuid": "6454c1d8-f817-4243-a47e-b371c1bc5ea3",
        "type": "IA4",
        "name": "Oxígeno",
        "message": "",
        "status": "0"
      }
    ]
  }
],
```

Captura 29. Recuperar personas en atención en un ámbito de red inalámbrica de área extensa (WWAN)



Captura 30. Ejecución Debug Android Studio estrategia adquisición WWMA (API remota REST HTTP)

3.6 Historia de usuario: Recuperar personas en atención en un ámbito de red inalámbrica de área personal (WPAN)

La aplicación móvil, al conectarse por bluetooth al dispositivo Hygia Control Unit (simulado en este proyecto con el entorno hardware construido sobre la placa Arduino), le envía un evento inicial solicitando la información de las personas en atención. El dispositivo responde con la información solicitada y la aplicación recupera y carga correctamente el listado de personas en atención recibido:

```
43 List<Alert> alerts = List<Alert>();
44 List<LogicPerson> logicPersons = List<LogicPerson>();
45
46 BluetoothDataContext._fromConnection(this._connection) {
47   print("[BluetoothDataContext][listen] Inicio");
48
49   connection.input.listen((data) {
50     print("[BluetoothDataContext][listen] Inicio escucha eventos");
51
52     String dataDecoded = escii.decode(data);
53     print("[BluetoothDataContext][listen] Evento recibido: " + dataDecoded);
54
55     _buffer += dataDecoded;
56     print("[BluetoothDataContext][listen] Contenido actual del buffer: " +
57       _buffer);
58
59     if (dataDecoded.contains("!")) {
60       // Encontramos terminador !, por tanto, hemos leído un mensaje completo
61       if (_buffer[0] == 'S') {
62         // Hemos recibido información de una baliza de sensor. Serializamos
63         _buffer = _buffer.substring(1, _buffer.length - 1);
64         print("[BluetoothDataContext][listen] Baliza de sensor recibida: " +
65           _buffer);
66         beaconModel = BeaconModel.fromJson(jsonDecode(_buffer));
67         beaconModel.time = DateTimeFormat.format(DateTime.now());
68
69         // Generamos la alerta asistencial correspondiente al sensor recibido
70         Alert alert = _dataAPIRepository.getAlert(beaconModel);
71         state.add(alert);
72       }
73     }
74   });
75 }
```

```
I/flutter ( 1326): [BluetoothDataContext][listen] Contenido actual del buffer: P[{"uuid": "ba221462-53a0-4111-b85f-aaa87514f197", "name": "Mr Noah Scott", "time": "11/04/2020 17:47", "lon": "1"}, {"uuid": "38b92632-fa08-46cd-8303-843723e8c93e", "name": "Mrs Donna Watson", "time": "11/04/2020 17:47", "lon": "-3.703445"}]
I/flutter ( 1326): [BluetoothDataContext][listen] Inicio escucha eventos
I/flutter ( 1326): [BluetoothDataContext][listen] Evento recibido: "lat": "40.417264", "IN
I/flutter ( 1326): [BluetoothDataContext][listen] Contenido actual del buffer: P[{"uuid": "ba221462-53a0-4111-b85f-aaa87514f197", "name": "Mr Noah Scott", "time": "11/04/2020 17:47", "lon": "1"}, {"uuid": "38b92632-fa08-46cd-8303-843723e8c93e", "name": "Mrs Donna Watson", "time": "11/04/2020 17:47", "lon": "-3.703445", "lat": "40.417264", "IN
I/flutter ( 1326): [BluetoothDataContext][listen] Inicio escucha eventos
I/flutter ( 1326): [BluetoothDataContext][listen] Evento recibido: ame": "Planta 1", "IDesc": "Hab. 2"]!
I/flutter ( 1326): [BluetoothDataContext][listen] Contenido actual del buffer: P[{"uuid": "ba221462-53a0-4111-b85f-aaa87514f197", "name": "Mr Noah Scott", "time": "11/04/2020 17:47", "lon": "1"}, {"uuid": "38b92632-fa08-46cd-8303-843723e8c93e", "name": "Mrs Donna Watson", "time": "11/04/2020 17:47", "lon": "-3.703445", "lat": "40.417264", "IName": "Planta 1", "IDesc": "Hab. 2"}]!
I/flutter ( 1326): [BluetoothDataContext][listen] Balizas de persona recibidas: [{"uuid": "ba221462-53a0-4111-b85f-aaa87514f197", "name": "Mr Noah Scott", "time": "11/04/2020 17:47", "lon": "1"}, {"uuid": "38b92632-fa08-46cd-8303-843723e8c93e", "name": "Mrs Donna Watson", "time": "11/04/2020 17:47", "lon": "-3.703445", "lat": "40.417264", "IName": "Planta 1", "IDesc": "Hab. 2"}]
I/flutter ( 1326): [_MainWidgetState][_updateEntityView]: PERSON
```

Captura 31. Recuperar personas en atención en un ámbito de red inalámbrica de área personal (WPAN)

4. Épica: Ejecutar acciones asistenciales remotas

4.1 Historia de usuario: Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área extensa (WWAN)

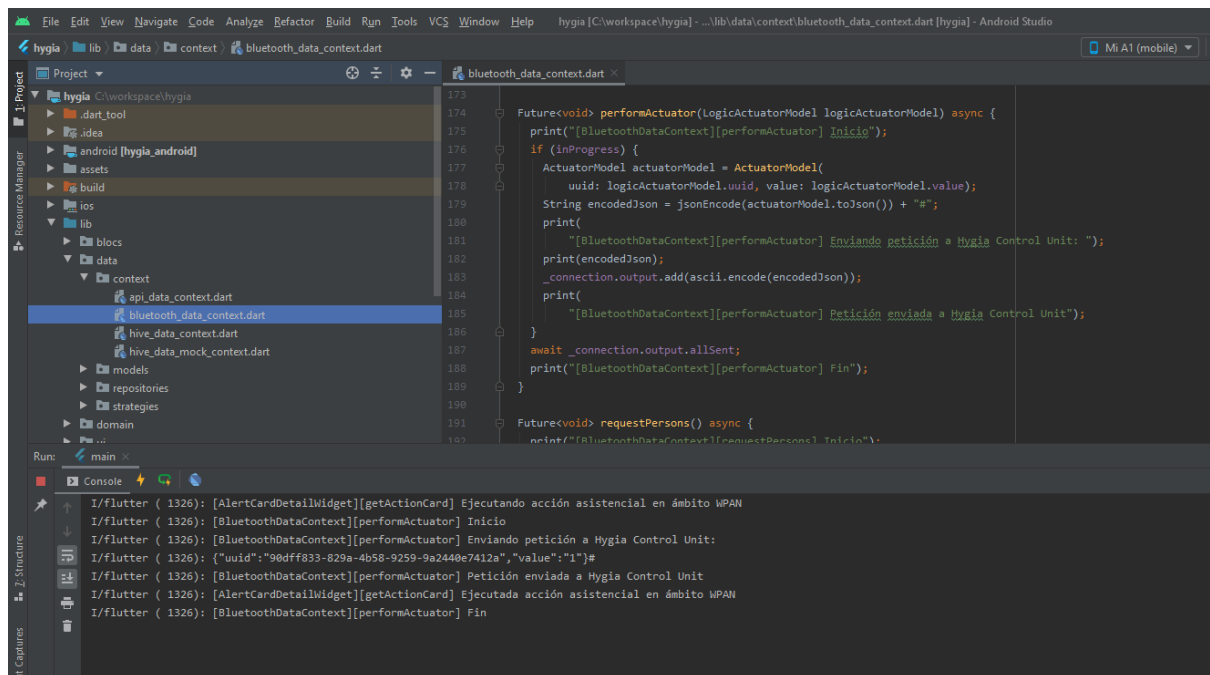
Al pulsar el botón de la acción asistencial desde la pantalla de detalle de la alerta, se realiza la petición correspondiente para la ejecución de la acción asistencial remota. En este caso, una petición HTTP al API REST (en este caso no observaremos el efecto de la acción, pues se trata de una petición simulada a un teórico dispositivo remoto):

```
1 import 'package:hygia/data/models/logic_action_model.dart';
2
3 import 'package:hygia/data/strategies/alert_strategy.dart';
4
5
6
7
8 /// [Hygia APP] wwan_alert_strategy.dart
9 /// Implementación de estrategia WWAN de adquisición de datos para el repositorio de alertas asistenciales
10
11 class WWANAlertStrategy implements AlertStrategy {
12   final ApiDataContext apiDataContext = ApiDataContext();
13
14   WWANAlertStrategy();
15
16   @override
17   FutureList<AlertModel>> getAlerts() async {
18     final url = 'http://my-json-server.typicode.com/daniuoc/hygiaapi/alerts';
19
20     dynamic data = await apiDataContext.getData(url);
21
22     final parsedJson = jsonDecode(data).cast<Map<String, dynamic>>();
23     List<AlertModel> alerts = parsedJson
24       .map<AlertModel>((json) => AlertModel.fromJson(json))
25       .toList();
26     return alerts;
27   }
28
29   @override
30   FutureString> sendAction(LogicActionModel logicActionModel) async {
31     final url = 'http://my-json-server.typicode.com/daniuoc/hygiaapi/action';
32
33     dynamic response =
34       await apiDataContext.sendData(url, logicActionModel.toJson());
35
36     return response;
37   }
38 }
39
```

Captura 32. Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área extensa (WWAN)

4.2 Historia de usuario: Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área personal (WPAN)

La aplicación envía correctamente por bluetooth al dispositivo hardware Hygia Control Unit de la solución y, como resultado, el dispositivo trata la información y ejecuta la acción sobre el actuador correspondiente. En este caso sí podemos ver el efecto de la acción sobre el dispositivo del entorno de simulación hardware (led encendido/apagado o sonido en *buzzer*):



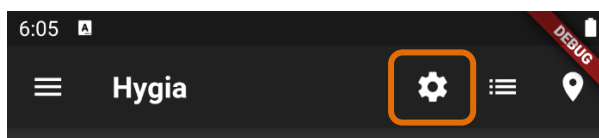
```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help hygia [C:\workspace\hygia] - ...\lib\data/context/bluetooth_data_context.dart [hygia] - Android Studio
hygia lib data context bluetooth_data_context.dart Mi A1 (mobile)
Project C:\workspace\hygia
.dart_tool
.idea
android [hygia_android]
assets
build
ios
lib
  blocs
  data
    api_data_context.dart
    bluetooth_data_context.dart
    hive_data_context.dart
    hive_data_mock_context.dart
  models
  repositories
  strategies
  domain
  ...
Resources Manager
Run: main x
Console
I/flutter ( 1326): [AlertCardDetailWidget][getActionCard] Ejecutando acción asistencial en ámbito WPAN
I/flutter ( 1326): [BluetoothDataContext][performActuator] Inicio
I/flutter ( 1326): [BluetoothDataContext][performActuator] Enviando petición a Hygia Control Unit:
I/flutter ( 1326): {"uuid":"90dff833-829a-4b58-9259-9a2440e7412a","value":"1"}#
I/flutter ( 1326): [BluetoothDataContext][performActuator] Petición enviada a Hygia Control Unit
I/flutter ( 1326): [AlertCardDetailWidget][getActionCard] Ejecutada acción asistencial en ámbito WPAN
I/flutter ( 1326): [BluetoothDataContext][performActuator] Fin
Future<void> performActuator(LogicActuatorModel logicActuatorModel) async {
  print("[BluetoothDataContext][performActuator] Inicio");
  if (InProgress) {
    ActuatorModel actuatorModel = ActuatorModel(
      uuid: LogicActuatorModel.uuid, value: LogicActuatorModel.value);
    String encodedJson = jsonEncode(actuatorModel.toJson()) + "#";
    print(
      "[BluetoothDataContext][performActuator] Enviando petición a Hygia Control Unit: ");
    print(encodedJson);
    _connection.output.add(ascii.encode(encodedJson));
    print(
      "[BluetoothDataContext][performActuator] Petición enviada a Hygia Control Unit");
  }
  await _connection.output.allSent;
  print("[BluetoothDataContext][performActuator] Fin");
}
Future<void> requestPersons() async {
  print("[BluetoothDataContext][requestPersons] Inicio");
}
```

Captura 33. Ejecutar acciones asistenciales en un ámbito de red inalámbrica de área personal (WPAN)

5. Épica: Configurar la aplicación

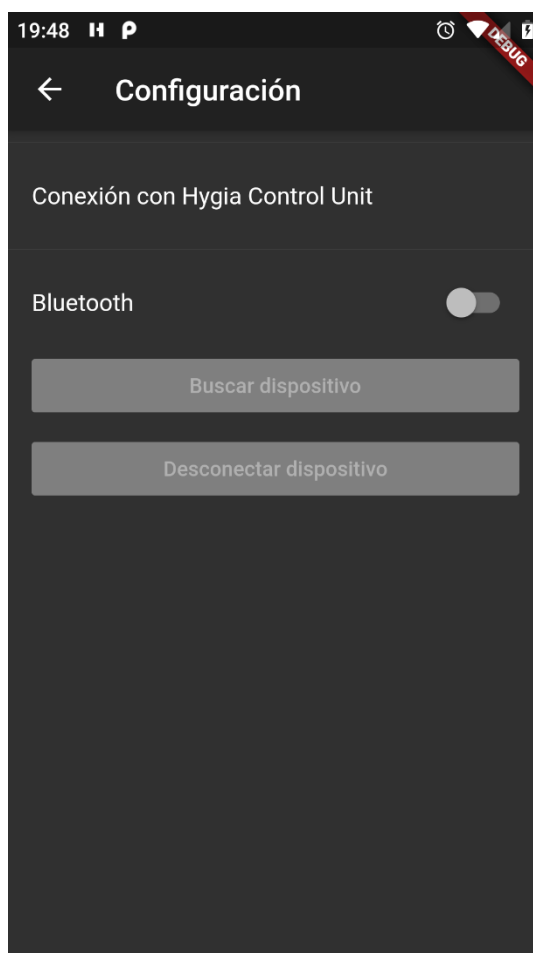
5.1 Historia de usuario: Configurar conexión Bluetooth con Hygia Control Unit

Para que la aplicación comience a trabajar en un ámbito de utilización de red inalámbrica personal, deberá conectarse a un dispositivo Hygia Control Unit (en este proyecto, reproducido en el entorno de simulación hardware). La conexión con este dispositivo se gestionará desde la sección de configuración de la aplicación, a través del icono correspondiente de la barra de herramientas superior:

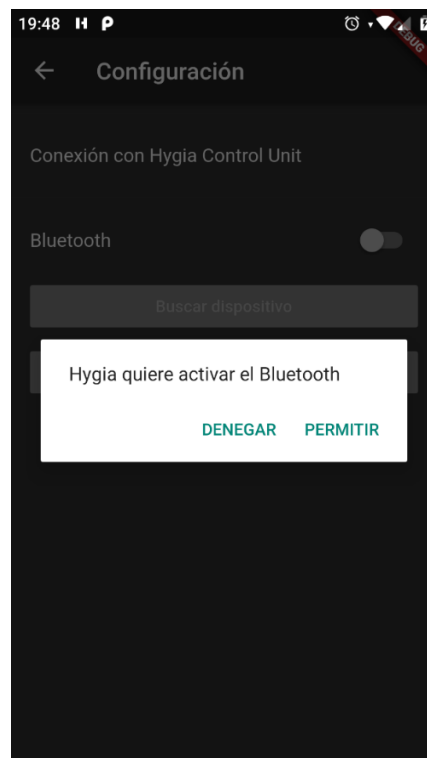


Captura 34. Barra de tareas (emulación Nexus 5X API 29 x86)

En esta sección de propiedades, en primer lugar, activamos la funcionalidad Bluetooth del dispositivo:

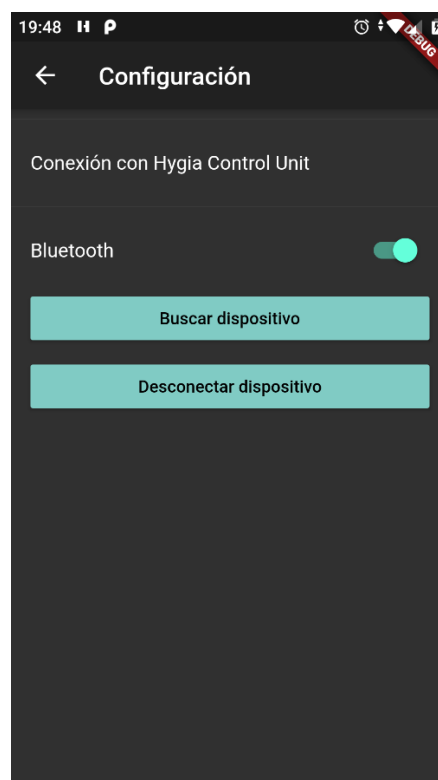


Captura 35. Pantalla de configuración 1 (emulación Nexus 5X API 29 x86)

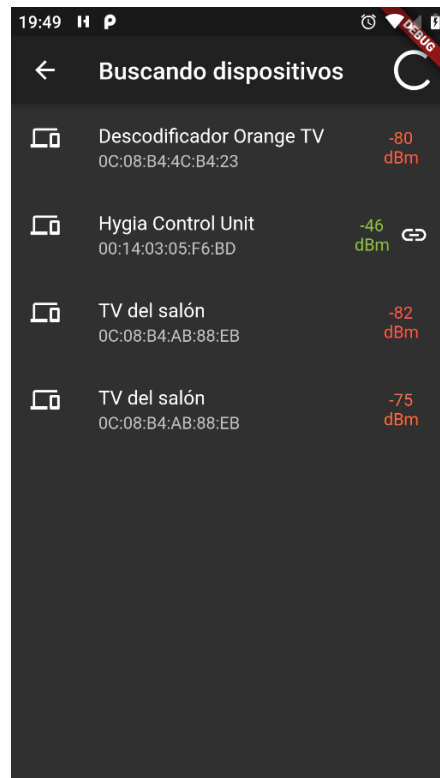


Captura 36. Pantalla de configuración 2 (emulación Nexus 5X API 29 x86)

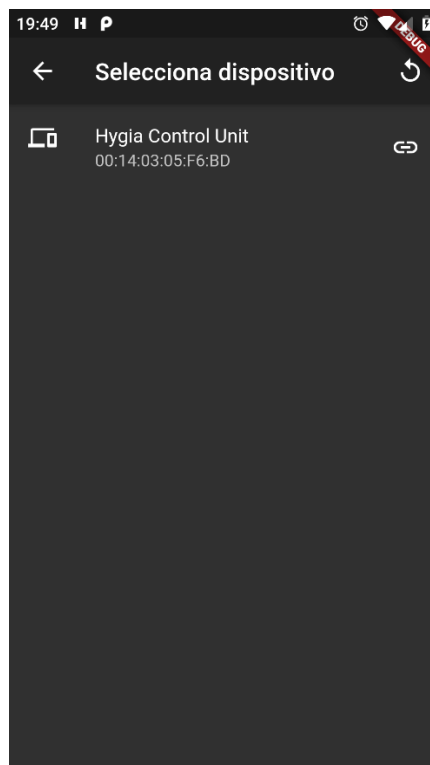
Una vez activada la funcionalidad bluetooth, se activan el resto de opciones. A continuación pueden explorarse los dispositivos en rango desde la opción “Buscar dispositivo” o conectar con un dispositivo Hygia Control Unit previamente vinculado seleccionándolo del listado:



Captura 37. Pantalla de configuración 3 (emulación Nexus 5X API 29 x86)



Captura 38. Pantalla de configuración 4 (emulación Nexus 5X API 29 x86)



Captura 39. Pantalla de configuración 5 (emulación Nexus 5X API 29 x86)

Capítulo 6: Conclusiones y líneas de futuro

1. Conclusiones

1.1 *Objetivos, conclusiones y lecciones aprendidas*

En relación a los objetivos planteados al inicio del proyecto, estoy muy satisfecho del resultado obtenido. Si bien la aplicación construida no es perfecta y tiene recorrido en varios aspectos (mayor funcionalidad, diseño, etc.), se han trabajado e implementando con éxito un conjunto de funcionalidades clave en este tipo de productos, como la interoperabilidad con dispositivos *hardware*, y se han puesto las bases para la continuidad del desarrollo del resto. El producto obtenido se ajusta a las especificaciones funcionales y de alcance planteadas al inicio del proyecto y constituye un buen producto de cara a su evolución en futuras líneas de desarrollo.

El proyecto planteaba varios retos interesantes, como contextualizar la aplicación móvil en una solución tecnológica más amplia de la que formaría parte o la interacción de ésta con elementos *hardware*. En este aspecto, considero que se ha realizado un buen trabajo de análisis y definición. En el apartado *hardware*, considero además un valor añadido para el proyecto el diseño, construcción y programación del entorno real realizado con la placa Arduino. Disponer de este entorno para realizar pruebas integradas con la aplicación han sido clave para determinar si la aplicación móvil responde adecuadamente a las necesidades requeridas de trabajo y comunicación en tiempo real con los dispositivos.

Uno de los retos personales importantes del proyecto ha sido enfrentarme por primera vez al desarrollo de una aplicación móvil. Aunque tengo mucha experiencia como desarrollador en otros ámbitos tecnológicos, el desarrollo *mobile* era completamente nuevo para mí, de hecho, un nuevo paradigma al que no me había enfrentado hasta el momento. El proceso ha sido muy positivo y enriquecedor, aunque ha requerido un gran esfuerzo y, a veces, se ha convertido en un hándicap por la falta de experiencia. En este aspecto, creo que la utilización del *framework* Flutter de Google ha sido una buena elección, pues facilita mucho algunos aspectos de la construcción de los elementos visuales, tiene una comunidad amplia y mucha gente desarrollando librerías de utilidad de código abierto.

A nivel personal, el aprendizaje durante el proyecto ha sido constante y, sin duda, he podido, y necesitado, aplicar todos los conocimientos adquiridos durante la realización del máster. He cubierto completamente también uno de los objetivos personales principales para mí, ejecutar de forma completa un proyecto de producción de una aplicación multimedia, desde la ideación hasta el producto final, con todas sus fases y productos intermedios de análisis, diseño, implementación, etc. Además, he podido aplicar un marco de trabajo ágil, algo que ha resultado muy enriquecedor.

1.2 Revisión detallada de los objetivos

A continuación detallo y justifico la respuesta sobre la base planteada en el apartado 3. Objetivos generales de este documento.

Objetivos de la aplicación/solución

- Se ha diseñado e implementado una aplicación móvil de acuerdo a las especificaciones funcionales y de alcance establecidas en el proyecto.
- Se ha definido, de forma general, una solución tecnológica en la cuál se integraría la aplicación para cubrir el servicio planteado.
- La aplicación contempla distintos ámbitos de utilización y diferentes aproximaciones para la adquisición/envío de datos.
- La aplicación construida es interoperable con dispositivos hardware, es decir, se ha conseguido la comunicación bidireccional entre la aplicación móvil y el entorno *hardware* construido (basado en la placa Arduino). Se ha diseñado e implementado un entorno *hardware* real para poder probar dicha integración, y las pruebas han resultado satisfactorias, obteniéndose el rendimiento esperado.
- Se ha utilizado un *framework* de desarrollo (Flutter) que ha permitido generar una única línea base de código multidispositivo y multiplataforma. Esto permitirá seguir evolucionando la aplicación como producto único ahorrando costes de desarrollo y mantenimiento.

Objetivos para el usuario

- Se ha implementado una aplicación muy funcional, con un interfaz sencillo y con buen rendimiento que ofrezca una experiencia adecuada al perfil de usuario objetivo (Asistente).
- La aplicación permite al usuario monitorizar y asistir, es decir, recibir en tiempo real alertas asistenciales y personas en atención según dos tipos de vistas (listado de trabajo y mapa), acceder a su información de detalle y actuar de forma remota sobre dispositivos.

Objetivos personales del autor de este trabajo

- He ejecutado de forma completa, aplicando un enfoque agile, el proceso de producción una aplicación multimedia, poniendo en práctica todos los conocimientos adquiridos en el máster.
- He utilizado uno de los frameworks que se perfila con más futuro para el desarrollo de aplicaciones móviles: Flutter.
- La aplicación construida interacciona con elementos hardware, por tanto, he podido experimentar el desarrollo de elementos de comunicaciones (Bluetooth), adquisición de datos etc.

Objetivos secundarios

- Se ha definido en el proyecto una posible solución tecnológica para cubrir servicios de atención remota sociosanitaria, aportando diagramas e identificación de componentes.

1.3 Revisión de planificación y metodología

La decisión de utilizar un marco de trabajo *agile* para este proyecto ha resultado acertada. Teniendo en cuenta las incertidumbres iniciales del proyecto, en relación a la falta de experiencia y conocimientos en la tecnología a emplear, las dudas en cuanto al posible alcance o las posibilidades de integración con los elementos hardware, este enfoque ágil nos ha permitido aproximarnos poco a poco a la solución del problema y evitar condicionar el producto por decisiones tomadas en etapas tempranas sin el conocimiento necesario. Un ejemplo sería la construcción del interfaz de la aplicación. Se decidió trabajar sobre una base inicial de *wireframes* de baja fidelidad y refinar y decidir el diseño final de las pantallas durante la ejecución de los distintos ciclos de *sprint*, a medida que se fueron explorando y conociendo las posibilidades ofrecidas por el framework de desarrollo seleccionado (Flutter).

Sobre una base de definición inicial, se fueron perfilando las características del producto a través de un proceso iterativo coherente, bien documentado y enfocado siempre a alcanzar los objetivos iniciales marcados. Sin perder la perspectiva general del proyecto, en cada ciclo de *sprint* nos enfocamos en producir un entregable completo, con el conjunto de funcionalidades planificadas y bien documentadas. Los resultados parciales de cada *sprint* nos ha permitido validar con el tutor si el camino elegido era correcto y la evolución era la esperada.

En definitiva, un proceso de trabajo que nos ha facilitado tener controlado el proyecto en todo momento, algo que nos ha permitido cumplir con la planificación prevista en cada etapa, aplicando los ajustes necesarios y minimizando el impacto de los errores o imprevistos.

2. Líneas de futuro

Se ha considerado, como valor añadido al proyecto, enfocar el diseño del producto (la aplicación móvil Hygia App) más allá del alcance de este TFM, de cara a una posible continuación en una línea futura de desarrollo. De hecho, el producto se ha enmarcado en una solución más amplia cuyas bases han sido descritas en este trabajo, así como, se han identificado y descrito cómo deberían ser los componentes que la integrarían.

En este sentido, se considera adecuado una posible línea futura de desarrollo en la que, continuando con el enfoque agile, se acordara con el *Product Owner* una actualización del *product backlog*, incluyendo aquellas funcionalidades necesarias para completar la solución de cara a una posible comercialización. A continuación se incluyen algunas épicas que podría servir como base para la construcción de la nueva pila de producto y completar la solución:

- Módulo configuración Hygia Directory Information (implementar las pantallas para al alta, modificación y consulta de datos sobre la infraestructura de base de datos HIVE que utiliza la aplicación)
- Servicio *backend* de monitorización sociosanitaria
- Servicio *backend* de asistencia sociosanitaria
- Dispositivo Hygia Control Unit
- Cifrado de comunicaciones
- Autenticación centralizada

Bibliografía

- [1] **AIREF (Autoridad Independiente de Responsabilidad Fiscal)** (2018, 4 de octubre). "Previsiones demográficas: una visión integrada". *AIREF (Autoridad Independiente de Responsabilidad Fiscal)* [publicación en línea]. [Fecha de consulta: 03 de Marzo del 2020].
<https://www.airef.es/wp-content/uploads/2018/10/NOTICIAS/PREVISIONES_DEMOGRAFICAS/2018-10-04-DOC-ESPECIAL-DEMOGRAFIA-FINAL.pdf>
- [2] **Alcaide J.C.** (2019, 18 de Julio). "Hablemos de silver economy". *ESIC* [artículo en línea]. [Fecha de consulta: 3 de Marzo del 2020].
<<https://www.esic.edu/rethink/2019/07/18/hablemos-de-silver-economy/>>
- [3] **Alcaide J.C.** (2017, 10 de Octubre). "La Silver Economy y el Senior Marketing". *Expansión* [artículo en línea]. [Fecha de consulta: 3 de Marzo del 2020].
<<https://www.expansion.com/blogs/consumidor-que-viene/2017/10/10/la-silver-economy-y-el-senior-marketing.html>>
- [4] **Augusto J.** (2019, 28 de octubre). "Conoce nuestro proyecto SMART ASSIST: Teleasistencia inteligente y accesible para el hogar" Televes Corporation [artículo en línea]. [Fecha de consulta: 15 de marzo del 2020].
<<https://blogcorporation.televes.com/sociosanitario/conoce-nuestro-proyecto-smart-assist-teleasistencia-inteligente-y-accesible-para-el-hogar/>>
- [5] **BOE (Boletín Oficial del Estado)** (2006, 15 de diciembre). "Ley 39/2006, de 14 de diciembre, de Promoción de la Autonomía Personal y Atención a las personas en situación de dependencia". *Agencia Estatal Boletín Oficial del Estado* [publicación en línea]. I. Disposiciones generales (núm. 299, páginas 44142 a 44156) [Fecha de consulta: 2 de Marzo del 2020].
<<https://www.boe.es/eli/es/l/2006/12/14/39>>
- [6] **Carney B.** (2020, 11 de febrero). "Announcing Ionic 5!" Ionic Blog [artículo en línea]. [Fecha de consulta: 15 de marzo del 2020].
<<https://ionicframework.com/blog/announcing-ionic-5/>>
- [7] **C. Martin, R.** (2017). *Clean Architecture: A Craftsman's Guide to Software Structure and Design (Robert C. Martin Series)*. US: Pearson Education.
- [8] **Cook D., Youngblood G., III E.O., Gopalratnam K., Rao S., Litvin A., Khawaja F.** (2003). "MavHome: An agent-based smart home". *Proceedings of the 1st IEEE International Conference on Pervasive Computing and Communications, PerCom 2003*. 521 - 524. 10.1109/PERCOM.2003.1192783.

[9] **Flutter Team** (2019, 7 de mayo). "Flutter: a Portable UI Framework for Mobile, Web, Embedded, and Desktop". Google Developers [artículo en línea]. [Fecha de consulta: 10 de marzo del 2020].
<<https://developers.googleblog.com/2019/05/Flutter-io19.html>>

[10] **Fowler, M.; Rice D.; Foemmel, M.; Hieatt, E.; Mee, R.; Stafford, R.** (2002). *Patterns of Enterprise Application Architecture*. Addison-Wesley Professional.

[11] **Galindo J.M.** (2010, 16 de febrero). "Diseño e implementación de un marco de trabajo (framework) de presentación para aplicaciones JEE". Trabajo Fin de Carrera. Universitat Oberta de Catalunya [Fecha de consulta: 11 de marzo del 2020].
<<http://hdl.handle.net/10609/876>>

[12] **Giménez Artes, A.; Arenas Díaz, C.; Lobato Beleiro, B.; Del Arco Herrera, I.; Peña Laguna, C.; Mañez, M.; Mur de Víu, C.; López Llorente, D.; Álvarez Rodríguez, F.; Aranaz Andrés, J.; Sánchez Cassinello, P.; Rambla Lop, J.; Arnott González-Tova, I.; Casas Torres, M.; Figuerola Martín, M.; Guerrero Fonseca, J.; Burgueño Jerez, A.** (2017). *100 Medidas que mejoran el sector salud en España*. Barcelona: Economía y Salud Fundación.

[13] **Google**. "Dark theme". *Material design* [artículo en línea]. [Fecha de consulta: 25 de abril del 2020].
<<https://material.io/design/color/dark-theme.html>>

[14] **iamquinmu, olprod** (2019, 16 de septiembre). "¿Qué es Xamarin?" Microsoft [artículo en línea]. [Fecha de consulta: 14 de marzo del 2020].
<<https://docs.microsoft.com/es-es/xamarin/get-started/what-is-xamarin>>

[15] **IDC Corporate** (2020, 20 de enero). "Smartphone Market Share" IDC Corporate [artículo en línea]. [Fecha de consulta: 15 de marzo del 2020].
<<https://www.idc.com/promo/smartphone-market-share/os>>

[16] **INE (Instituto Nacional de Estadística)** (2018, 10 de octubre). "Proyecciones de Población 2018". *INE (Instituto Nacional de Estadística)* [nota de prensa en línea]. [Fecha de consulta: 28 de Febrero del 2020].
<https://ine.es/prensa/pp_2018_2068.pdf>

[17] **INE (Instituto Nacional de Estadística)** (2018, 10 de octubre). "Proyección de Hogares. 2018-2033". *(INE) Instituto Nacional de Estadística* [nota de prensa en línea]. [Fecha de consulta: 29 de Febrero del 2020].
<https://ine.es/prensa/ph_2018_2033.pdf>

- [18] **Kontis V., Bennett J.E., Mathers C. D., Li G., Foreman K., Ezzati M.** (2017, 21 de Febrero). "Future life expectancy in 35 industrialised countries: projections with a Bayesian model ensemble". *The Lancet* [artículo en línea]. [Fecha de consulta: 28 de Febrero del 2020].
<[https://www.thelancet.com/journals/lancet/article/PIIS0140-6736\(16\)32381-9/fulltext](https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(16)32381-9/fulltext)>
- [19] **Leler Wm.** (2017, 20 de agosto). "What's Revolutionary about Flutter" Hackernoon [artículo en línea]. [Fecha de consulta: 9 de marzo del 2020].
< <https://hackernoon.com/whats-revolutionary-about-flutter-946915b09514>>
- [20] **López D.** (2018, 20 de marzo). "Orange e Ilunion lanzan 'Serena', la teleasistencia en movilidad" By Orange: Blog oficial de Orange [artículo en línea]. [Fecha de consulta: 05 de marzo del 2020].
<<http://blog.orange.es/noticias/orange-e-ilunion-lanzan-serena-la-teleasistencia-en-movilidad/>>
- [21] **Lynch M.** (2019, 22 de mayo). "Announcing Capacitor 1.0" Ionic Blog [artículo en línea]. [Fecha de consulta: 15 de marzo del 2020].
<<https://blog.ionicframework.com/announcing-capacitor-1-0/>>
- [22] **Millán S.** (2013, 4 de febrero). "Telefónica adquiere la brasileña Axismed, especializada en la gestión de pacientes" CincoDías [artículo en línea]. [Fecha de consulta: 05 de marzo del 2020].
<https://cincodias.elpais.com/cincodias/2013/02/04/empresas/1360148342_850215.html/>
- [23] **Nabors R.** (2019, 17 de julio). "Meet Hermes, a new JavaScript Engine optimized for React Native" React Native Blog [artículo en línea]. [Fecha de consulta: 15 de marzo del 2020].
<<https://reactnative.dev/blog/2019/07/17/hermes>>
- [24] **Riehle D.** (2000). "Framework Design: A Role Modeling Approach". Doctoral Dissertation, no. 13509. Swiss Federal Institute of Technology at Zurich (ETH Zurich), Zurich, Switzerland. [Fecha de consulta: 13 de marzo del 2020].
<<https://riehle.org/computer-science/research/dissertation/index.html>>
- [25] **Schiemann D., Couriol B., Humble C.** (2020, 12 de marzo). "JavaScript and Web Development InfoQ Trends Report 2020" InfoQ [artículo en línea]. [Fecha de consulta: 17 de marzo del 2020].
<<https://www.infoq.com/articles/javascript-web-development-trends-2020/>>
- [26] **Smith A.** (2019, 28 de noviembre). "Why Businesses Should Start Focusing on Google's Flutter and Fuchsia" The Startup [artículo en línea]. [Fecha de consulta: 10 de marzo del 2020].
<<https://medium.com/swlh/why-businesses-should-start-focusing-on-googles-flutter-and-fuchsia-48e16820f2a9>>

[27] **Tarnowski D.** (2019, 12 de mayo). "Flutter and Fuchsia. The next big thing?" The Startup [artículo en línea]. [Fecha de consulta: 10 de marzo del 2020].

<<https://medium.com/swlh/flutter-and-fuchsia-the-death-of-react-android-a34f6d12bb82>>

[28] **Undermatic** (2020, 6 de enero). "Tendencias de Diseño para el 2020 (Parte I)". *UNDERMATIC* [artículo en línea]. [Fecha de consulta: 25 de abril del 2020].

<<http://www.undermatic.com/tendencias-de-diseno-para-el-2020/>>

[29] **Vrevsky G.** (2019, 9 de septiembre). "Progressive Web Apps: What Do Marketers Need to Know?" Business 2 Community [artículo en línea]. [Fecha de consulta: 10 de marzo del 2020].

<<https://www.business2community.com/digital-marketing/progressive-web-apps-what-do-marketers-need-to-know-02237775>>

[30] **Zabala X.** (2019, 23 de julio). "Telefónica y la compañía británica Tunstall Healthcare desarrollarán proyectos de gestión remota de pacientes en España" Telemedicina [artículo en línea]. [Fecha de consulta: 07 de marzo del 2020].

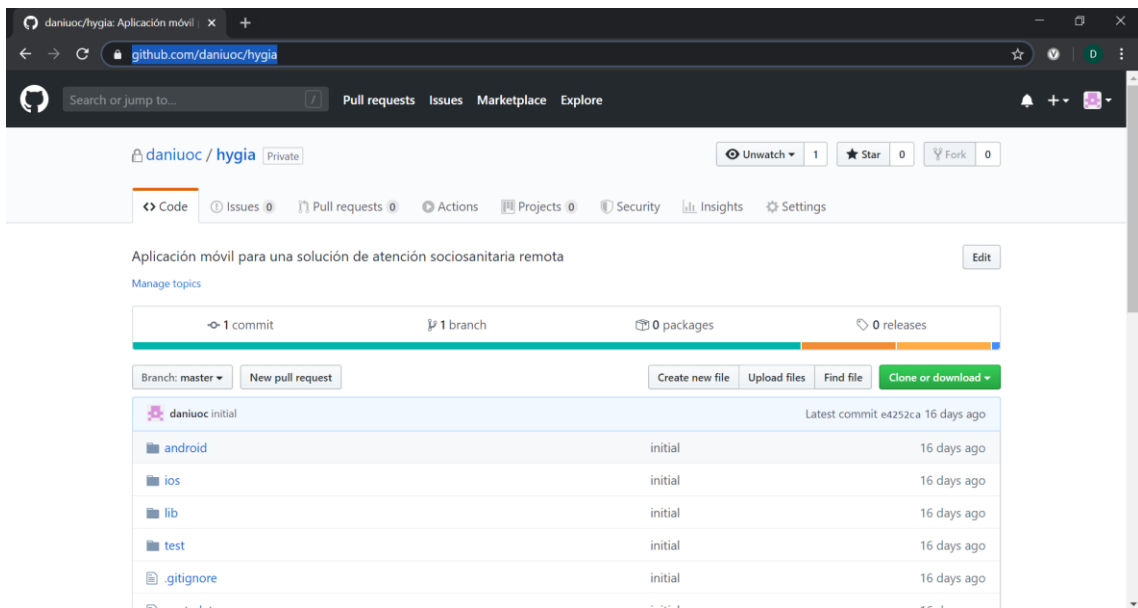
<<https://latelemedicina.es/index.php/2019/07/23/telefonica-y-tunstall-gestion-remota-de-pacientes>>

Anexos

Anexo A: Herramientas utilizadas en el proyecto

A continuación se enumeran las herramientas más relevantes seleccionadas para cubrir las distintas necesidades del proyecto. Se han utilizado herramientas sin coste, o bien, herramientas de pago pero en su versiones gratuitas limitadas por periodo de uso o funcionalidad.

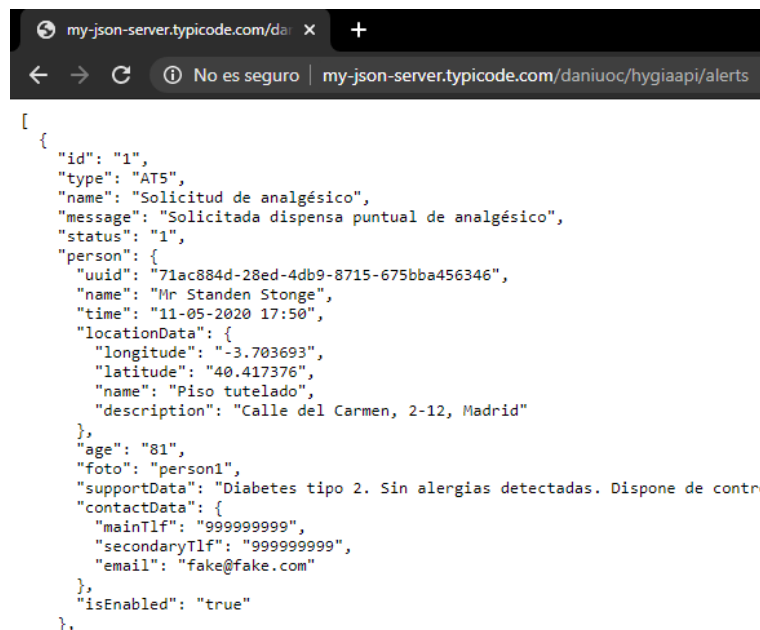
- MockFlow (mockflow.com): herramienta para la producción de wireframes. La utilizaremos para trabajar el UX (User eXperience o Experiencia de usuario) del proyecto. Usaremos la licencia básica que es gratuita de forma permanente para un único proyecto.
- Canva (canva.com): herramienta de diseño gráfico. La utilizaremos para la producción de los recursos gráficos de la aplicación (iconos, imágenes, etc.).
- Fritzing (fritzing.org): herramienta open-source de diseño electrónico. La emplearemos para diseñar el entorno de simulación del equipamiento hardware de la solución.
- Android Studio (developer.android.com/studio): 3.6.2 for Windows 64-bit. Es el entorno de desarrollo integrado oficial para la plataforma Android que usaremos para desarrollar la aplicación.
- Flutter (flutter-es.io): v1.12.13+hotfix.9. Es un SDK (Software Development Kit o Kit de Desarrollo de Software) de código abierto creado por Google para el desarrollo de aplicaciones móviles. Se integra perfectamente como plugin en Android Studio.
- Controlador USB de Google: necesario para poder realizar depuración de dispositivos Android con Android Debug Bridge.
- GitHub: herramienta de gestión de repositorios online, sistema de control de versiones distribuido Git. Se utiliza en el proyecto, integrado en Android Studio, para la gestión de versiones de los componentes software desarrollados.



Captura 40. Repositorio creado para el proyecto: <https://github.com/daniuoc/hygia>

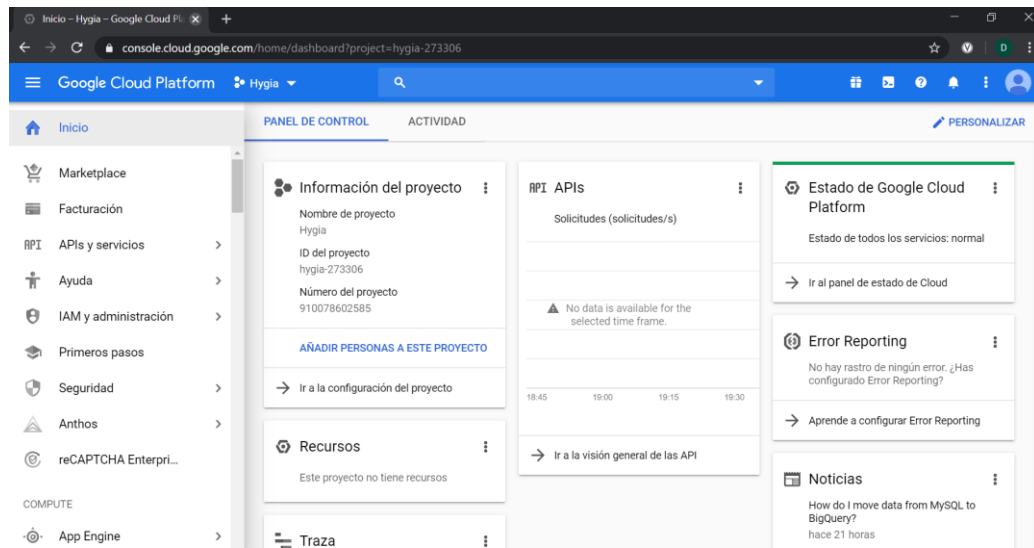
- Arduino Software (IDE): 1.8.12. Entorno de desarrollo integrado open source para el desarrollo de software para las placas open-hardware Arduino. Lo utilizaremos para construir el software de nuestro entorno de simulación hardware de WPAN (Wireless Personal Area Networks o Red Inalámbrica de Área Personal).
- JSONPlaceholder (jsonplaceholder.typicode.com): herramienta online gratuita integrada con GitHub para simular APIs REST, simplemente dejando un fichero con contenido JSON en un repositorio de git. Ejemplo de utilización para el proyecto:

<http://my-json-server.typicode.com/daniuoc/hygiaapi/alerts>



Captura 41. JSONPlaceholder

- Google Cloud (console.cloud.google.com): La plataforma Google en Cloud en la que ha sido necesario dar de alta nuestro proyecto para poder utilizar, por ejemplo, el API de Google Maps.



Captura 42. Configuración Google Cloud

Anexo B: Entregables del proyecto

- Proyecto: Entrega del producto completo y acabado.
 - o Producto:
 - Aplicación móvil Hygia: *app-release.apk*
 - Instrucciones de instalación y manual de usuario: *App Hygia - Instrucciones de instalación y uso.pdf*
 - Código fuente proyecto Flutter: carpetas *lib*, *android*, *ios*, *assets* y archivo *pubspec.yaml*.
 - Código fuente Arduino: *HygiaControlUnit.ino*
 - o Archivos de trabajo:
 - Wireframes: *wireframes.zip*
 - Esquema fritzing diseño entorno simulación *hardware: HygiaLab.fzz*
 - Recursos gráficos propios y ajenos.

- Memoria:
 - o El presente documento: *dguzmanl_TFM_memoria.pdf*

- Presentación académica: video en formato mp4 publicado en el aula mediante la herramienta Present@:
 - o *dguzmanl_TFM_presentación_académica.mp4*

- Presentación pública: formato Prezi ejecutable (.exe)
 - o *dguzmanl_TFM_presentación_pública.exe*

- Informe de trabajo: informe de trabajo de la PEC 5
 - o *dguzmanl_PEC5_Informe_de_Trabajo.pdf*