



Escuela
Politécnica
Superior

DCA Control. Prevención y seguimiento de pacientes con epilepsia.



Grado en Ingeniería Multimedia

Trabajo Fin de Grado

Autor:

Sergio Castro Sola

Tutor/es:

José García Rodríguez

Junio 2017



Universitat d'Alacant
Universidad de Alicante

Agradecimientos

Debo dedicar un agradecimiento a mi tutor, José García, por todo el tiempo y dedicación que ha empleado en el proyecto, por los consejos, y sugerencias.

Además, agradecer a toda la gente que ha contribuido en mi enseñanza durante los cuatro años que he cursado Ingeniería Multimedia.

Dedicatoria

Dedicado a todas las personas importantes en mi vida: mis padres, mi hermano, mi abuela y mi pareja sin ellos no podría haber conseguido este objetivo tan ansiado.

También está dedicado a mis amigos, los de toda la vida y los conocidos durante mi estancia en la Universidad de Alicante.

Por último, este trabajo está dedicado a todas aquellas personas que sufren DCA, o tienen algún familiar que lo sufre.

Resumen

El presente documento corresponde a la memoria del trabajo de fin de grado de nombre “DCA Control. Prevención y seguimiento de pacientes con epilepsia”. Ha sido realizado por Sergio Castro Sola, estudiante de ingeniería multimedia, bajo la supervisión de José García Rodríguez, su tutor.

Con este proyecto se trata de ayudar a resolver el problema de la prevención de ataques de epilepsia en personas con daño cerebral adquirido. Es un problema real, ya que en el mercado existen pocos sistemas capaces de lograrlo de forma fiable y que, además, proporcione datos de utilidad para un posterior análisis por parte del personal médico cualificado.

Este proyecto parte de la base del trabajo realizado por Gala M. García y su tutor, José García, que se basaba en la monitorización de pacientes con pulseras inteligentes. Este nuevo proyecto trata de darle un giro para añadir nuevas funcionalidades que supongan un gran valor añadido al trabajo realizado anteriormente.

El desarrollo del proyecto se divide en dos partes: el marco teórico y el desarrollo.

La primera parte contiene una revisión de las investigaciones previas a la presente propuesta, que sirven como base para un correcto desarrollo. Para ello es necesario un estudio de las tecnologías óptimas para el sistema, del hardware disponible en el mercado y de los sistemas que existen actualmente para afrontar el problema de la epilepsia. Además, es necesario comprender lo que significa el DCA y la epilepsia en la sociedad, conocer sus causas, síntomas y consecuencias, con el fin de realizar un sistema que sea realmente útil para estos pacientes.

El segundo apartado, el de desarrollo, está enfocado al producto final. Sin embargo, también es necesario estudiar la propuesta de la aplicación, concretar funcionalidades, interfaces y requisitos. Se definirán claramente las partes del sistema previo desarrollo, lo que ayudará a completar el sistema en las fechas acordadas.

Índice de contenido

ÍNDICE DE CONTENIDO	1
ÍNDICE DE ILUSTRACIONES.....	4
ÍNDICE DE TABLAS.....	6
1. INTRODUCCIÓN	9
2. JUSTIFICACIÓN Y OBJETIVOS.....	11
2.1. OBJETIVOS ESPECÍFICOS	12
2.2. RELACIÓN CON ASIGNATURAS	12
3. ESTADO DEL ARTE.....	14
3.1. DAÑO CEREBRAL ADQUIRIDO.	14
3.1.1. <i>Daño Cerebral infantil</i>	17
3.1.2. <i>Crisis epilépticas</i>	18
3.2. CAUSAS DEL DCA.....	21
3.3. CONSECUENCIAS DEL DCA	23
3.4. E-SALUD.....	24
3.5. ALTERNATIVAS EN LA COMPETENCIA.....	26
3.5.1. <i>Proyectos similares</i>	26
3.5.1.1. <i>EpSMon</i>	26
3.5.1.2. <i>Epilepsy Tool Kit</i>	27
3.5.1.3. <i>EpDetect</i>	29
3.5.1.4. <i>Daño Cerebral</i>	30
3.5.1.5. <i>SeizAlarm</i>	31
3.5.1.6. <i>DCA Epilepsia</i>	32
3.5.1.7. <i>Embrace</i>	33
3.5.1.8. <i>PulseGuard</i>	35
3.5.2. <i>Conclusiones</i>	36
3.6. ESTUDIO SMARTBANDS.....	39
3.6.1. <i>Microsoft Band2</i>	39
3.6.2. <i>Razer Nabu X</i>	40
3.6.3. <i>Fitbit Charge 2</i>	41
3.6.4. <i>Atlas Wristband 2</i>	41
3.6.5. <i>Samsung Simband 2</i>	42
3.6.6. <i>Xiaomi Mi Band 2</i>	43
3.6.7. <i>Garmin Fenix 5</i>	43
3.6.8. <i>Empatica</i>	44

3.6.9.	<i>Conclusiones</i>	45
3.7.	ESTUDIO PLATAFORMAS	46
3.7.1.	<i>Aplicaciones nativas</i>	46
3.7.1.1.	<i>React Native</i>	47
3.7.1.2.	<i>NativeScript</i>	48
3.7.1.3.	<i>Xamarin</i>	49
3.7.2.	<i>Aplicaciones híbridas</i>	49
3.7.2.1.	<i>Cordova/PhoneGap</i>	50
3.7.2.2.	<i>Ionic 2 Framework</i>	51
3.7.3.	<i>Conclusiones</i>	52
4.	METODOLOGÍA	53
4.1.	REQUISITOS MÍNIMOS	53
4.2.	ETAPAS	53
4.3.	METODOLOGÍAS ÁGILES.....	54
4.4.	HERRAMIENTAS HARDWARE	55
4.4.1.	<i>Xiaomi Mi Band 2</i>	55
4.4.2.	<i>BQ Aquaris M5</i>	56
4.4.3.	<i>Raspberry Pi 2 Model B</i>	57
4.5.	HERRAMIENTAS SOFTWARE	58
4.5.1.	<i>Documentación</i>	58
4.5.2.	<i>Planificación</i>	58
4.5.3.	<i>Diseño</i>	59
4.5.4.	<i>Desarrollo</i>	59
4.5.5.	<i>Pruebas</i>	60
4.6.	INGENIERÍA INVERSA	60
5.	ESTUDIO DE LA VIABILIDAD	63
5.1.	PLANIFICACIÓN TEMPORAL	63
5.2.	GESTIÓN DE RIESGOS.....	64
6.	ESPECIFICACIÓN DE REQUISITOS	73
6.1.	OBJETIVOS DEL SISTEMA	73
6.2.	CASOS DE USO	74
6.2.1.	<i>Usuario</i>	75
6.2.2.	<i>Personal médico</i>	80
6.2.3.	<i>Paciente</i>	81
6.3.	REQUISITOS FUNCIONALES.....	84
6.4.	REQUISITOS NO FUNCIONALES.....	88

7. DISEÑO Y ARQUITECTURA	90
7.1. PROTOTIPOS DE LA APLICACIÓN	90
7.2. DIAGRAMA ENTIDAD-RELACIÓN	97
8. DESARROLLO E IMPLEMENTACIÓN	99
8.1. API	99
8.2. APLICACIÓN WEB	100
8.3. APLICACIÓN MÓVIL	102
9. CONCLUSIONES.....	107
9.1. REVISIÓN DE LOS OBJETIVOS	107
9.2. TRABAJOS FUTUROS	108
9.3. CONCLUSIONES	108
10. REFERENCIAS	110
11. ANEXO.....	113
11.1. DOCUMENTACIÓN DE LA API	113
11.2. MANUAL DE USUARIO	116
11.2.1. REGISTRO WEB.....	117
11.2.2. INICIO.....	118
11.2.3. CREAR PACIENTE.....	119
11.2.4. LISTA DE PACIENTES.....	119
11.2.5. DETALLE DE PACIENTE	120
11.2.6. SEGUIMIENTO DEL RITMO CARDÍACO	121
11.2.7. LOGIN MÓVIL.....	122
11.2.8. HOME	122
11.2.9. TRATAMIENTO	123
11.2.10. HISTORIAL.....	123
11.2.11. MONITORIZAR	124
11.3. ENLACES DE INTERÉS	125

Índice de ilustraciones

ILUSTRACIÓN 1. PERSONAS CON DCA SEGÚN EL SEXO Y EL GRUPO DE EDAD.	16
ILUSTRACIÓN 2. ELECTROENCEFALOGRAMA DE UN NIÑO DURANTE UNA CRISIS EPILÉPTICA.....	19
ILUSTRACIÓN 3. NÚMERO DE ALTAS HOSPITALARIAS POR TCE EN ESPAÑA POR GRUPOS DE EDAD.	22
ILUSTRACIÓN 4. ACCIÓN SUDEP.....	26
ILUSTRACIÓN 5. CUESTIONARIO Y RESULTADO EPSMON.....	27
ILUSTRACIÓN 6. MENÚ Y REGISTRO DE ATAQUE.	28
ILUSTRACIÓN 7. EPDETECT: DETECCIÓN DE ATAQUE.....	29
ILUSTRACIÓN 8. DAÑO CEREBRAL: MENÚ Y CONSEJOS.....	30
ILUSTRACIÓN 9. SEIZALARM.	31
ILUSTRACIÓN 10. DCA EPILEPSIA: HOME Y ALARMAS.....	33
ILUSTRACIÓN 11. EMBRACE: ACTIVIDAD Y REGISTRO DE ATAQUE.....	34
ILUSTRACIÓN 12. PULSEGUARD: APLICACIÓN EN IPAD.....	35
ILUSTRACIÓN 13. ARTÍCULO SOBRE LA CANCELACIÓN DE ANGEL SENSOR.	39
ILUSTRACIÓN 14. MICROSOFT BAND 2.	40
ILUSTRACIÓN 15. RAZER NABU X.	40
ILUSTRACIÓN 16. FIBIT CHARGE 2.	41
ILUSTRACIÓN 17. ATLAS WRISTBAND 2.	42
ILUSTRACIÓN 18. SAMSUNG SIMBAND 2.	42
ILUSTRACIÓN 19. XIAOMI MI BAND 2.	43
ILUSTRACIÓN 20. GARMIN FENIX 5.....	44
ILUSTRACIÓN 21. EMBRACE EMPATICA.....	44
ILUSTRACIÓN 22. LOGOTIPO REACT NATIVE.	47
ILUSTRACIÓN 23. LOGOTIPO DE NATIVESCRIPT.....	48
ILUSTRACIÓN 24. LOGOTIPO DE XAMARIN.....	49
ILUSTRACIÓN 25. LOGOTIPO APACHE CORDOVA.....	50
ILUSTRACIÓN 26. LOGOTIPO IONIC 2 FRAMEWORK.	51
ILUSTRACIÓN 27. COMPARATIVA DEL INTERÉS DE LOS FRAMEWORKS.	52
ILUSTRACIÓN 28. XIAOMI MI BAND 2.	55
ILUSTRACIÓN 29. BQ AQUARIS M5.....	57
ILUSTRACIÓN 30. RASPBERRY PI 2.	57
ILUSTRACIÓN 31. COMPARACIÓN INGENIERÍA INVERSA Y DIRECTA.	61
ILUSTRACIÓN 32. DIAGRAMA DE GANTT HITO 0.....	63
ILUSTRACIÓN 33. DIAGRAMA DE GANTT HITO 1.....	64
ILUSTRACIÓN 34. DIAGRAMA DE GANTT HITO 2.....	64
ILUSTRACIÓN 35. DIAGRAMA DE GANTT HITO 3.....	64
ILUSTRACIÓN 36. CASO DE USO USUARIO.	75

ILUSTRACIÓN 37. CASO DE USO MÉDICO.....	80
ILUSTRACIÓN 38. CASO DE USO PACIENTE.....	82
ILUSTRACIÓN 39. PROTOTIPO: LOGIN WEB.	90
ILUSTRACIÓN 40. PROTOTIPO: REGISTRO.	91
ILUSTRACIÓN 41. PROTOTIPO: INICIO.	91
ILUSTRACIÓN 42. PROTOTIPO: MANUAL DE USUARIO.....	92
ILUSTRACIÓN 43. PROTOTIPO: CREAR PACIENTE.	92
ILUSTRACIÓN 44. PROTOTIPO: MIS PACIENTES.	93
ILUSTRACIÓN 45. PROTOTIPO: DETALLE DE PACIENTE.....	93
ILUSTRACIÓN 46. PROTOTIPO: GRÁFICA HR.....	94
ILUSTRACIÓN 47. PROTOTIPO: CREAR TRATAMIENTO.	94
ILUSTRACIÓN 48. PROTOTIPO: LOGIN APP.	95
ILUSTRACIÓN 49. PROTOTIPO: SELECCIÓN DE DISPOSITIVO.....	95
ILUSTRACIÓN 50. PROTOTIPO: MONITORIZACIÓN.....	96
ILUSTRACIÓN 51. PROTOTIPO: MEDICACIÓN ASOCIADA.	96
ILUSTRACIÓN 52. PROTOTIPO: CONFIGURAR MEDICACIÓN.	97
ILUSTRACIÓN 53. DIAGRAMA ENTIDAD-RELACIÓN.....	98
ILUSTRACIÓN 54. API: ERROR 405.	100
ILUSTRACIÓN 55. APLICACIÓN WEB: ROUTING.....	101
ILUSTRACIÓN 56. APLICACIÓN WEB: INICIO.....	102
ILUSTRACIÓN 57. APLICACIÓN MÓVIL. INICIO.....	103
ILUSTRACIÓN 58. APLICACIÓN MÓVIL: VINCULACIÓN.	105
ILUSTRACIÓN 59. APLICACIÓN MÓVIL: ACTIVAR SENSOR HR.	106
ILUSTRACIÓN 60. MANUAL: REGISTRO.	117
ILUSTRACIÓN 61. MANUAL: INICIO.	118
ILUSTRACIÓN 62. MANUAL: CREAR PACIENTE.....	119
ILUSTRACIÓN 63. MANUAL: LISTAR PACIENTES.....	119
ILUSTRACIÓN 64. MANUAL: DETALLE DE PACIENTE.....	120
ILUSTRACIÓN 65. MANUAL: SEGUIMIENTO.....	121
ILUSTRACIÓN 66. MANUAL: LOGIN.	122
ILUSTRACIÓN 67. MANUAL: HOME.	122
ILUSTRACIÓN 68. MANUAL: TRATAMIENTO.	123
ILUSTRACIÓN 69. MANUAL: HISTORIAL.	124
ILUSTRACIÓN 70. MANUAL: MONITORIZAR.....	124

Índice de tablas

TABLA 1. PERSONAS CON DCA EN ESPAÑA.	15
TABLA 2. TOTAL DE PERSONAS CON DCA ESPAÑA. AÑO 2008.....	16
TABLA 3. NÚMERO DE PERSONAS CON DCA EN MILES POR CA.....	17
TABLA 4. TASA DE PERSONAS CON DCA POR MIL HABITANTES.....	17
TABLA 5. COMPARATIVA DE SISTEMAS EN EL MERCADO.....	37
TABLA 6. COMPARATIVA GENERAL SMARTBANDS.....	45
TABLA 7. COMPARATIVA DE SENSORES DE LAS SMARTBANDS.....	46
TABLA 8. RIESGO 1.....	65
TABLA 9. RIESGO 2.....	65
TABLA 10. RIESGO 3.....	65
TABLA 11. RIESGO 4.....	66
TABLA 12. RIESGO 5.....	66
TABLA 13. RIESGO 6.....	66
TABLA 14. RIESGO 7.....	67
TABLA 15. RIESGO 8.....	67
TABLA 16. RIESGO 9.....	67
TABLA 17. RIESGO 10.....	68
TABLA 18. RIESGO 11.....	68
TABLA 19. RIESGO 12.....	68
TABLA 20. RIESGO 13.....	69
TABLA 21. RIESGO 14.....	69
TABLA 22. RIESGO 15.....	69
TABLA 23. RIESGO 16.....	70
TABLA 24. RIESGO 17.....	70
TABLA 25. RIESGO 18.....	70
TABLA 26. RIESGO 19.....	71
TABLA 27. RIESGO 20.....	71
TABLA 28. RIESGO 21.....	71
TABLA 29. RIESGO 22.....	72
TABLA 30. RIESGO 23.....	72
TABLA 31. RIESGO 24.....	72
TABLA 32. OBJETIVO DEL SISTEMA 1.....	73
TABLA 33. OBJETIVO DEL SISTEMA 2.....	74
TABLA 34. OBJETIVO DEL SISTEMA 3.....	74
TABLA 35. OBJETIVO DEL SISTEMA 4.....	74
TABLA 36. OBJETIVO DEL SISTEMA 5.....	74

TABLA 37. CU 1 USUARIO.....	75
TABLA 38. CU 2 USUARIO.....	76
TABLA 39. CU 3 USUARIO.....	76
TABLA 40. CU 4 USUARIO.....	76
TABLA 41. CU 5 USUARIO.....	77
TABLA 42. CU 6 USUARIO.....	77
TABLA 43. CU 7 USUARIO.....	77
TABLA 44. CU 8 USUARIO.....	77
TABLA 45. CU 9 USUARIO.....	78
TABLA 46. CU 10 USUARIO.....	78
TABLA 47. CU 11 USUARIO.....	78
TABLA 48. CU 12 USUARIO.....	78
TABLA 49. CU 13 USUARIO.....	79
TABLA 50. CU 14 USUARIO.....	79
TABLA 51. CU 15 USUARIO.....	79
TABLA 52. CU 16 USUARIO.....	79
TABLA 53. CU 17 MÉDICO.....	81
TABLA 54. CU 18 MÉDICO.....	81
TABLA 55. CU 19 PACIENTE.....	82
TABLA 56. CU 20 PACIENTE.....	83
TABLA 57. CU 21 PACIENTE.....	83
TABLA 58. CU 22 PACIENTE.....	83
TABLA 59. CU 23 PACIENTE.....	83
TABLA 60. REQUISITO FUNCIONAL 1.....	84
TABLA 61. REQUISITO FUNCIONAL 2.....	84
TABLA 62. REQUISITO FUNCIONAL 3.....	84
TABLA 63. REQUISITO FUNCIONAL 4.....	85
TABLA 64. REQUISITO FUNCIONAL 5.....	85
TABLA 65. REQUISITO FUNCIONAL 6.....	85
TABLA 66. REQUISITO FUNCIONAL 7.....	85
TABLA 67. REQUISITO FUNCIONAL 8.....	86
TABLA 68. REQUISITO FUNCIONAL 9.....	86
TABLA 69. REQUISITO FUNCIONAL 10.....	86
TABLA 70. REQUISITO FUNCIONAL 11.....	86
TABLA 71. REQUISITO FUNCIONAL 12.....	87
TABLA 72. REQUISITO FUNCIONAL 13.....	87
TABLA 73. REQUISITO FUNCIONAL 14.....	87
TABLA 74. REQUISITO FUNCIONAL 15.....	87

TABLA 75. REQUISITO FUNCIONAL 16.	88
TABLA 76. REQUISITO NO FUNCIONAL 1.	88
TABLA 77. REQUISITO NO FUNCIONAL 2.	88
TABLA 78. REQUISITO NO FUNCIONAL 3.	89
TABLA 79. REQUISITO NO FUNCIONAL 4.	89
TABLA 80. REQUISITO NO FUNCIONAL 5.	89
TABLA 81. REQUISITO NO FUNCIONAL 6.	89
TABLA 82. API: HISTORIAL.	113
TABLA 83. API: RITMO CARDÍACO.	114
TABLA 84. API: LOGIN.	114
TABLA 85. API: PACIENTE.	114
TABLA 86. API: PATOLOGÍA.	115
TABLA 87. API: TEMPERATURA.	115
TABLA 88. API: TRATAMIENTO.	116
TABLA 89. API: USUARIO.	116

1. Introducción

Este proyecto tiene como objetivo contribuir en la línea de trabajo en el uso de dispositivos vestibles inteligentes para la monitorización de pacientes. Siendo, así mismo; continuación del excelente trabajo realizado por Gala, compañera de Ingeniería Multimedia. Dicho trabajo trataba sobre la prevención de ataques de epilepsia en pacientes con Daño cerebral adquirido (DCA).

El Daño cerebral adquirido es una lesión que afecta a parte o a la totalidad del tejido cerebral. Este daño no tiene por qué producirse en nacimiento, siendo circunstancias externas las principales causas de la enfermedad. Se estima que el 78% de los casos tienen su origen en un ictus, mientras que el resto se producen por traumatismos craneoencefálicos.

En niños también es algo relativamente común. Se ha estimado que 250 niños de cada 100.000 se ven afectados. Sus causas también son traumatismos craneoencefálicos y accidentes cerebro-vasculares, pero también son comunes tumores o infecciones cerebrales. Es este uno de los casos más sensibles, debido a la preocupación de los padres, ya que sus hijos tienen necesidades sanitarias, educativas y sociales especiales. Además, un niño puede no expresarse con la misma claridad que un adulto, y esto puede provocar frustración a los padres.

Las secuelas del DCA pueden ser de muchos tipos. Desde físicas, sensitivas, cognitivas hasta emocionales, los afectados pueden o no presentar varios de estos síntomas, por lo que cada paciente ha de tratarse de forma diferente.

Uno de los principales síntomas, que es más probable que se exteriorice conforme van pasando los años, es la epilepsia. Los ataques de epilepsia son una serie de convulsiones o movimientos corporales incontrolados y de forma repetitiva. Aunque estos ataques se pueden producir en personas sin daño cerebral, ya que cualquier motivo que impida o distorsione el patrón de actividad neuronal normal puede conducir su aparición, la epilepsia está íntimamente relacionada con el DCA.

La prevención es una gran arma contra los ataques de epilepsia porque así se podrían evitar lesiones producidas por el movimiento incontrolado del cuerpo. El uso de cinturones de seguridad en vehículos, medicinas y terapias puede reducir los riesgos de lesiones y de aparición de ataques, pero no garantizan su desaparición.

Es por esto que pacientes y familiares necesitan un control más exhaustivo del paciente que, por otro lado, no condicione sus vidas y que sea lo menos intrusivo posible.

2. Justificación y objetivos

Hoy en día, gracias al avance tecnológico, se puede abordar este problema desde diferentes enfoques. Anteriormente sólo se podía consultar a los profesionales, que por otra parte siguen siendo imprescindibles, pero después de la consulta el enfermo dejaba de ser tratado hasta la siguiente visita. Esto producía y sigue produciendo en muchos casos, una sensación de impotencia tanto para el enfermo como para la familia, que pueden verse desbordados ante la situación.

Ante este problema hay diversos retos y tecnologías desde el punto de vista de un ingeniero. Por un lado, se encuentran sensores externos que se colocan en alguna parte del cuerpo o el uso de electrodos que miden la frecuencia cardíaca con precisión y en tiempo real. Sin embargo, estos sistemas son muy específicos y costosos para el paciente. Por otro lado, se podría plantear una aplicación que simplemente aconsejara cómo evitar ataques epilépticos y qué hacer cuando se producen, pero esto no elimina la posibilidad de prevención en un caso real.

Es por esto por lo que se ha optado por utilizar una smartband para monitorizar a los pacientes que lo necesiten. Las smartbands son uno de los muchos wearables que actualmente se encuentran en el mercado, siendo uno de los más destacados.

El proyecto a realizar será la continuación de la idea propuesta por Gala, que desarrolló una aplicación para Android capaz de detectar ataques de epilepsia con la pulsera Angel Sensor. Además, se establecerán mejoras y funcionalidades complementarias para que la monitorización y seguimiento del paciente sea más satisfactoria.

Desgraciadamente, los desarrolladores de Angel Sensor han abandonado el proyecto, por lo que, a día de hoy, es imposible obtener una en el mercado. Aunque cuento con una unidad y podría realizar el proyecto con parte del trabajo realizado por Gala, carece de sentido ya que los futuros usuarios del sistema no podrían utilizarlo al serles imposible adquirir esta smartband.

El sistema consistirá, por un lado, en una aplicación móvil que leerá datos de una pulsera y será capaz de calcular cuando se está produciendo un ataque de epilepsia y avisar vía sms a la persona o personas que se desee. Por otro lado, se dispondrá de una aplicación web, donde el usuario o profesional del sector médico podrá consultar el estado del paciente, su histórico de constantes vitales, ataques y otros datos de interés.

Por último y como enlace de las dos aplicaciones, se creará una API restfull que recibirá datos de la aplicación móvil, los guardará y los hará accesibles desde la aplicación web.

2.1. Objetivos específicos

Con este proyecto se pretende conseguir demostrar la utilidad del empleo de wearables mediante una serie de objetivos propuestos que permitan alcanzar la meta final de este trabajo.

- Investigar y analizar proyectos similares.
- Investigar en profundidad la enfermedad a tratar.
- Estudiar el mercado para la correcta selección de la *smartband* a emplear.
- Investigar y analizar la pulsera inteligente *Xiomi miBand 2* mediante ingeniería inversa.
- Establecer los requisitos mínimos que ha de cumplir la aplicación.
- Diseñar la aplicación móvil en consecuencia a los requisitos establecidos.
- Implementar prototipos en cada iteración del proyecto.
- Entregar la aplicación móvil en su versión acabada.

2.2. Relación con asignaturas

Según la universidad de Alicante la ingeniería Multimedia se encuentra entre las ingenierías tradicionales y la ingeniería informática y su objetivo es crear profesionales en el sector de las TIC capaces de gestionar proyectos en el ámbito de la Multimedia, ya sea en el sector del ocio y entretenimiento digital o en el de gestión de contenidos.

En mi caso, me sitúo en este segundo itinerario, denominado gestión de contenidos, siendo sus asignaturas las que claramente más se adaptan al proyecto. Además de estas hay otras de carácter obligatorio que también son necesarias para el correcto desarrollo del sistema.

Entre las asignaturas cursadas a lo largo del grado, de las de todas he aprendido algo, hay que destacar las siguientes:

- Usabilidad y accesibilidad: creación de interfaces de usuario con características de usabilidad, que sean fácilmente comprensibles y de rápido aprendizaje.
- Diseño de bases de datos multimedia: como su nombre indica, desarrollo y gestión de bases de datos.

- Análisis y especificación de sistemas multimedia: uso de metodologías ágiles de ingeniería de software.
- Programación hipermedia I y II: desarrollo en entorno web.
- Dispositivos e infraestructuras para sistemas multimedia: introducción del concepto multiplataforma y de lo que conlleva.
- Proyectos multimedia: Planificación, estimación y seguimiento de un proyecto.
- Servicios multimedia basados en internet: creación de API, seguridad y despliegue.
- Servicios multimedia avanzados: desarrollo de aplicación multiplataforma y resolución de problemas tecnológicos específicos de cada proyecto.

3. Estado del Arte.

En este apartado se profundiza sobre el DCA y todos los aspectos relacionados con esta lesión. Se analizan las causas que lo desencadenan, así como las consecuencias que conlleva. También se trata la epilepsia, estrechamente relacionada con el DCA y cuya prevención es el principal motivo de la realización de este proyecto.

Por último, se describen las opciones que la tecnología ofrece para tratar esta patología, así como un repaso por los sistemas que existen actualmente y su eficacia.

3.1. Daño Cerebral Adquirido.

Según la asociación para pacientes con daño cerebral adquirido de la provincia de Alicante el DCA es “aquella lesión o alteración permanente a nivel cerebral que se produce bruscamente en la vida de una persona, a consecuencia de: traumatismos craneoencefálicos, accidentes cerebro vasculares, tumores, hipoxias, infecciones, etc.” (ADACEA).

Una lesión cerebral es un trastorno en el tejido cerebral independientemente de la edad en que ocurra. Esta lesión produce una pérdida parcial o total de las funciones controladas por el cerebro que previamente estaban en perfectas condiciones.

Una lesión cerebral supone un impedimento de gran importancia en la persona que lo sufre, pudiendo causar deterioro cognitivo, problemas de atención, de memoria o motores. Estas carencias limitan la autonomía de los pacientes y, en muchos casos, les hace dependientes de una tercera persona.

La gravedad de la lesión y las posibles consecuencias requieren una rapidez de actuación por parte de los servicios médicos. Esta actuación se divide en cuatro fases ordenadas según la gravedad y desconocimiento de las consecuencias que acarreará el paciente.

- Fase crítica: Comienza justo después del ingreso hospitalario. Esta fase comienza una vez el paciente se encuentra estable dentro de la gravedad y se reestablece el riego sanguíneo al tejido cerebral. Su duración se prolonga mientras el paciente continúe ingresado en la unidad de cuidados intensivos.
- Fase aguda: Cuando el paciente abandona la UCI y es trasladado a planta. Este traslado supone una tranquilidad para el paciente y los familiares, ya que quiere decir que la actividad neuronal se ha estabilizado y el riesgo de recaídas es

menor. Como la fase crítica, la fase aguda tiene se prolonga mientras el paciente se encuentre ingresado en planta.

- Fase rehabilitadora: la tercera etapa supone un gran cambio ya que se los profesionales médicos determinan que el paciente es apto para recibir el alta hospitalaria. En estos momentos tanto el paciente como los familiares conocen los daños causados por la lesión y la rehabilitación que ha de seguir. Al conocer la lesión se establece una rehabilitación personalizada para el paciente.
- Fase de readaptación: en ella se pone al paciente en contacto con asociaciones de DCA. Gracias a ella los pacientes alivian la costosa reinserción en la sociedad. Estas asociaciones promueven actividades conjuntas y proporcionan ejercicios y consejos para intentar que los pacientes retomen su vida tal y como la conocían. La duración de esta etapa depende de la gravedad de la lesión, variando desde unos meses a toda la vida.

El DCA afecta a más de 400.000 personas sólo en España por lo que se convierte en un problema de primer nivel. Representa la principal causa de discapacidad en los países desarrollados. La gravedad y el grado de incidencia en su vida depende de cada caso por lo que las formas de tratar el DCA son particulares para cada paciente.

DCA según causa	Números absolutos (miles de personas)	Proporciones por cada 1.000 habitantes
Accidentes Cerebrovasculares	329.544	7,3
Sin especificar	90.520	2,0
Total	420.064	9,3

Tabla 1. Personas con DCA en España.

Fuente: INE.

Esta tabla se obtiene a partir de la encuesta de Discapacidad, autonomía Personal y Situaciones de Dependencia 2008 del Instituto Nacional de Estadística (INE). Siendo la encuesta más reciente y de la que se obtienen los datos posteriores.

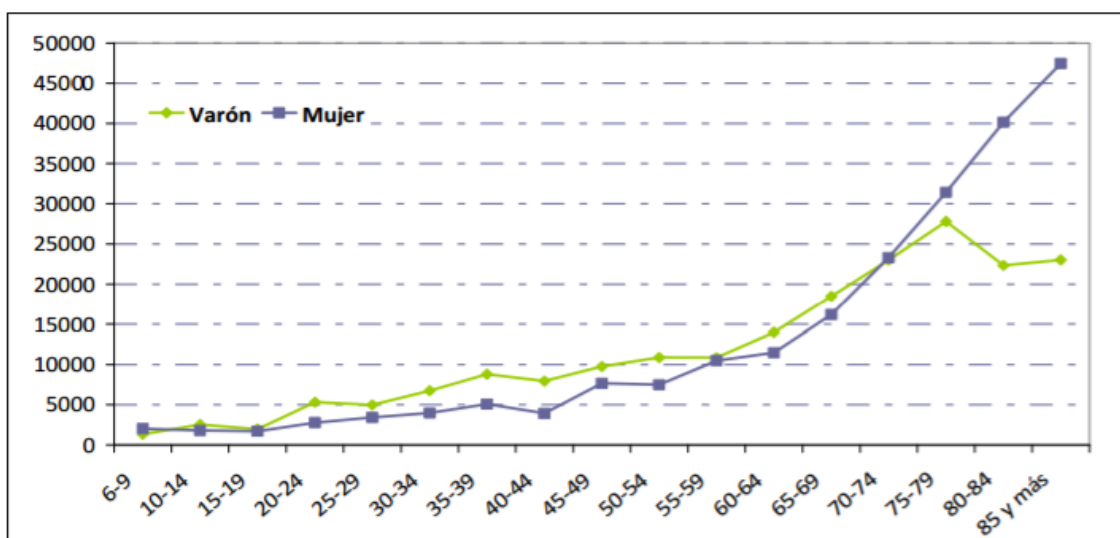


Ilustración 1. Personas con DCA según el sexo y el grupo de edad.

Fuente: INE.

Como se puede apreciar en la gráfica anterior, existe una relación entre personas con DCA y la edad. A partir de los 45 años los casos aumentan significativamente, representando estas personas el mayor porcentaje de pacientes.

Para una distribución por sexos más específica se muestra la tabla de abajo, de las que se pueden obtener varias conclusiones:

	Números absolutos (miles de personas)			Porcentajes		
	Varones	Mujeres	Ambos	Varones	Mujeres	Ambos
De 6 a 64 años	85.091	61.819	146.910	42,60%	28,06%	34,97%
65 en adelante	114.643	158.511	273.154	57,40%	71,94%	65,03%
Total	199.734	220.330	420.064	100,00%	100,00%	100,00%

Tabla 2. Total de personas con DCA España. Año 2008.

Fuente: INE.

Atendiendo a los resultados de la tabla anterior se observa que el 52,5% de afectados son mujeres, mientras que un 47,5% varones. Centrándonos en la edad la proporción cambia. Los varones de 6 a 64 años representan el 57,9% de casos, por el 42,1% de mujeres en el mismo periodo. Por último, es evidente que, aunque el número general de

pacientes es elevado, la mayoría son personas de 65 años o más, representando más del 65% de las personas que sufren DCA.

También es interesante la distribución por Comunidades Autónomas. Gracias a ella se puede saber cuáles son las comunidades con más casos o con más afectados por habitante.

CCAA	Varones	Mujeres	Total
Andalucía	37.848	43.407	81.255
Comunidad Valenciana	28.571	27.913	56.484
Cataluña	24.971	30.087	55.058

Tabla 3. Número de personas con DCA en miles por CA.

Fuente: INE.

En la Comunidad Valenciana existen 56.484 casos registrados: 28.571 varones y 27.913 mujeres. Resulta ser una de las comunidades autónomas con mayor número de personas junto con Andalucía y Cataluña. Esto también se debe a que también son las comunidades más pobladas de España junto a Madrid. Sin embargo no son en proporción las comunidades con más afectados.

CCAA	Varones	Mujeres	Total
Galicia	12,6	13,6	13,1
Asturias	12,3	12,8	12,5
Murcia	11,5	12,7	12,1

Tabla 4. Tasa de personas con DCA por mil habitantes.

Fuente: INE.

Gracias a esta tabla y sabiendo que el 78% de los casos son traumatismos craneoencefálicos, se puede establecer una relación directa con la cantidad de accidentes de tráfico, su gravedad y el número de afectados.

3.1.1. Daño Cerebral infantil

El DCA infantil no es más que un caso particular del DCA. Puede afectar a las personas en las primeras etapas de su vida. Estos primeros años de vida son en los que la persona se desarrolla física y mentalmente. Por esta razón se debe tratar de forma diferente a los pacientes en edad infantil.

En los pacientes con DCA infantil, por lo general, se producen mayores daños a nivel psíquico que en el resto de casos. Esto dificulta el correcto desarrollo de nuevas capacidades cognitivas o en casos graves el olvido de otras ya desarrolladas.

Al ser esta una etapa de aprendizaje en las personas, este se puede ver dificultado debido a los problemas que presente el paciente, como pueden ser déficits de memoria o de atención. Esto junto a la integración social en un ambiente adecuado es algo vital para que el paciente pueda desarrollar una vida como haría el resto de personas.

Por otro lado, los pacientes con DCA infantil tienen más facilidad para mejorar secuelas de carácter físico o del lenguaje, con lo que en un futuro será más complicado que se manifiesten problemas derivados del DCA (GALA M. García, 2016).

3.1.2. Crisis epilépticas

La epilepsia es un trastorno provocado por un desequilibrio en la actividad eléctrica de las neuronas en alguna zona del cerebro. La persona que sufra un episodio epiléptico manifiesta movimientos corporales incontrolados durante un corto período de tiempo. Dichos movimientos son el único síntoma de estas crisis epilépticas, por lo que una persona con epilepsia no se ve afectada por ella el resto del tiempo.

Una crisis epiléptica es un evento de corta duración, cuya principal característica es el movimiento involuntario en forma de convulsiones y, en ocasiones, pérdida de consciencia. Para diagnosticar la epilepsia es necesario conocer los antecedentes personales, ya que en principio cualquier persona puede sufrir un ataque epiléptico a lo largo de su vida sin poder decir que por ello sufra epilepsia. Además, es aconsejable conocer los antecedentes familiares y la realización de diversas pruebas, como puede ser un electroencefalograma.



Ilustración 2. Electroencefalograma de un niño durante una crisis epiléptica.

Fuente: Wikipedia.org.

Un electroencefalograma es una exploración que registra la actividad bioeléctrica cerebral en condiciones de reposo, vigilia o sueño. Como se puede observar en la fotografía anterior, un electroencefalograma sirve para detectar episodios epilépticos en una persona de forma muy clara. Sin embargo, las condiciones en las que se realiza esta prueba son excepcionales e imposibles de llevar a la vida cotidiana, por lo que se han de buscar otros síntomas o maneras de monitorizar al paciente de forma sencilla sin que afecte a su día a día.

El lugar donde el afectado sufra un episodio epiléptico puede aumentar el riesgo de gravedad, ya que la presencia de objetos móviles o el hecho de encontrarse a cierta altura puede desencadenar heridas físicas importantes.

En la mayoría de las ocasiones, los episodios se producen en circunstancias similares. Esto hace que si una persona presta atención pueda predecir, con cierta probabilidad, cuándo va a sufrir un nuevo ataque. Además, existen diversos signos que también ayudan a predecir un ataque. Estos van desde hormigueo, o sensación de olores inexistentes a cambios emocionales. También es posible que se perciban signos vegetativos como una taquicardia o bricardia. Estos dos últimos son clave para la monitorización que se pretende hacer de los pacientes, ya que la taquicardia es un aumento repentino del ritmo cardíaco y la bricardia por su parte es una disminución del mismo.

Según la Comisión Internacional de la Liga Internacional contra la Epilepsia (ILAE), las crisis epilépticas se clasifican en:

- Crisis generalizadas: se desarrollan de forma simétrica en ambos hemisferios cerebrales, sin aparente inicio local. En función de los síntomas expuestos, se distingue entre:
 - Crisis tónicas, clónicas o tónico-clónicas: Son las más graves, presentan crecientes y bruscas convulsiones que desencadenan expulsión de espuma por la boca o incontinencia.
 - Crisis de ausencia: Presentan una pérdida de la conciencia o, si son complejas, movimientos clónicos o automáticos breves. Sin embargo, son menos graves que las anteriores.
 - Síndrome de Lennox-Gastaut: Presente en niños de 1 a 8 años de edad, las crisis más frecuentes son las axiales, las atónicas y las ausencias. En el 60% de los casos, este síndrome sobreviene en niños con una encefalopatía preexistente, en el resto se desconoce la etiología.
 - Epilepsia mioclónica juvenil: Este tipo de epilepsia aparece en la pubertad. Se caracteriza por episodios de sacudidas arrítmicas e irregulares, que predominan en los miembros superiores. Hay riesgo de caída al suelo cuando se sufre un episodio de este tipo. Pueden verse favorecidas por la falta de sueño.
 - Síndrome de West: Su principal característica son los espasmos infantiles, detención del desarrollo psicomotor e hipsarritmia. Aparece antes del primer año de vida.
 - Crisis Atónicas: En ella la persona pierde el tono muscular y la consciencia durante unos segundos. Suelen ir acompañadas de otro tipo de crisis. La gravedad se debe a los golpes o caídas que puede sufrir cuando se tiene una crisis de este tipo.
- Crisis parciales: tienen lugar en una parte concreta del cerebro, donde originalmente se producen. Pueden extenderse a otras partes del cerebro. Debido a los distintos síntomas que producen, se dividen en:
 - Simples: El paciente con una crisis de este tipo padece síntomas motores que producen sacudidas rítmicas durante unos segundos o minutos. Es

posible que antes de producirse el afectado perciba sensaciones descritas anteriormente, como hormigueo u olor extraño.

Complejas: En ella la persona pierde el conocimiento, se queda inmóvil durante unos minutos o segundos y realiza movimientos automáticos de masticación o deglución con la boca. Otras veces se mueve como si estuviera consciente, pero con un comportamiento peculiar: no responde al hablarle o lo hace de forma incoherente. Es posible que una crisis de este tipo comience con una crisis parcial simple.

3.2. Causas del DCA

El DCA puede surgir debido a muchas causas, pero a día de hoy no se conocen todas ellas. Gracias a los avances médicos se han demostrado una serie de síntomas, enfermedades y sucesos que provocan DCA. El conjunto de estos casos son las causas mayoritarias por las que se adquiere DCA, por eso se va a profundizar en cada una de ellas.

Según el observatorio del ictus, el **ictus** “son un conjunto de enfermedades que afectan a los vasos sanguíneos que suministran la sangre al cerebro” (observatorio del ictus). A este número de patologías se las conoce como embolias o accidentes cerebrovasculares (ACV). Se puede decir que un ictus equivale a un infarto de corazón, pero en el cerebro.

Existen dos tipos de ictus: hemorragias cerebrales e infartos cerebrales. Los primeros se producen cuando un vaso sanguíneo se rompe, interrumpiendo la correcta circulación del flujo sanguíneo. En el caso de los infartos cerebrales, es una arteria la que se obstruye debido a un coágulo de sangre. Este trombo se origina en el corazón y se desplaza hasta el cerebro donde, como en las hemorragias, se interrumpe la circulación sanguínea.

Después de un ictus, sólo un tercio de los pacientes se recupera totalmente, otro tercio fallece y el resto queda con secuelas. Estas secuelas pueden ser más o menos graves, pero en cualquier caso condicionan la vida de las personas que las sufren.

En España ocurren 130.000 casos de ictus cada año, de los que cerca de 80.000 fallecen o padecen DCA. Se estima que el 25% de las personas que padecen un ictus mueren durante los 30 días siguientes (observatorio del ictus).

Afortunadamente hay factores de riesgo que se pueden prevenir. Los principales son el consumo de alcohol, drogas y tabaco. También afecta la hipertensión arterial, el alto nivel de colesterol, la diabetes, la obesidad y la vida sedentaria.

El **traumatismo craneoencefálico** (TCE) es otra de las causas principales de la adquisición de DCA. Se produce cuando el cerebro sufre un golpe a causa de una fuerza extrema que puede producir la inconsciencia temporal. Alrededor del 75% de los TCE se deben a accidentes de tráfico, por un 20% que se deben a caídas, y el resto a lesiones deportivas.

Se estima que en España ocurren 200 casos nuevos por cada 100.000 habitantes al año, de los cuales el 10% se consideran graves, otro 10% moderados y el resto, leves. Es tres veces más frecuente en varones que en mujeres.

Las consecuencias de sufrir un TCE son graves, ya que es la primera causa de muerte y discapacidad de las personas menores de 45 años. Además, el 21% de los supervivientes adquieren DCA de forma moderada o grave (Bárcena Orbe et al, 2006).

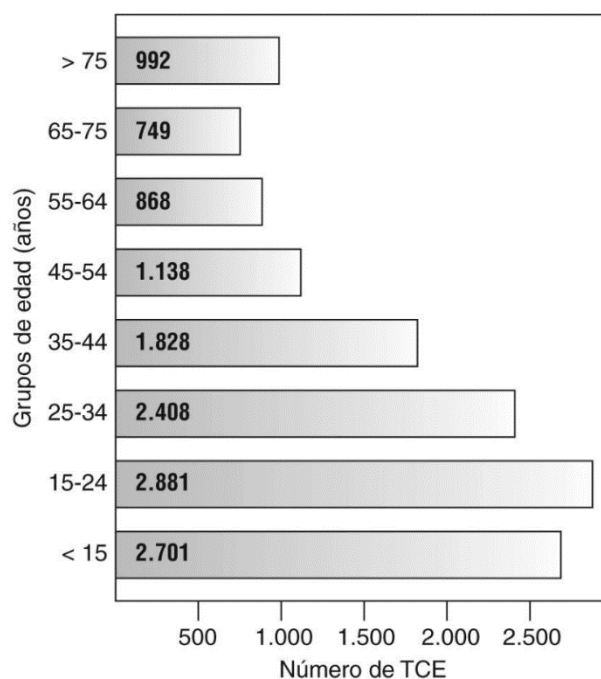


Ilustración 3. Número de altas hospitalarias por TCE en España por grupos de edad.

Fuente: INE.

Como se puede apreciar en la gráfica anterior, en las etapas tempranas de la vida de las personas es cuando se sufren más TCE. La mayoría de los TCE en personas de menos de 15 años se deben a caídas. Conforme va aumentando la edad es cuando la causa

principal pasa a ser los accidentes de tráfico. Cabe destacar que, en general, los traumatismos craneoencefálicos van disminuyendo conforme pasan los años, no siendo una de las causas más importantes de mortalidad.

Una **anoxia** es un tipo de encefalopatía que se produce por una detención del aporte de oxígeno al cerebro por un tiempo mayor al mínimo para que no se produzca muerte neuronal. El cerebro puede soportar como máximo, de cuatro a cinco minutos de anoxia, pero una vez sobrepasado este tiempo empezará a producirse un daño irreparable que se incrementa notablemente con el paso de los minutos. Al cabo de quince minutos el 95% del tejido cerebral estará seriamente dañado.

Las causas más comunes en edades adultas son el ahogamiento, el infarto y la asfixia. En caso de los recién nacidos es posible que se produzcan complicaciones en el parto que deriven en un axioma lo suficientemente prolongado como para presentar secuelas.

3.3. Consecuencias del DCA

Las consecuencias a largo plazo del DCA pueden ser muy variadas, dependiendo del área del cerebro afectada y de la gravedad de la lesión. También influyen factores propios de cada paciente, como puede ser la personalidad, la edad, o sus aptitudes previas.

En general, una persona con DCA es más irritable, impaciente, puede presentar síntomas de depresión o desánimo. También existe una falta de conciencia de los déficits que padecen, especialmente si son alteraciones cognitivas. Es por esto que el paciente puede rechazar la ayuda que se le presta. Existen principalmente cuatro formas de diferenciar las consecuencias de los pacientes:

Los **déficits físico-motores** son, como su nombre indica, alteraciones físicas que dificultan la vida del paciente. Estos pueden ser alteraciones del equilibrio, incapacidad para mantenerse de pie, para andar, limitaciones en las extremidades o incluso la cabeza. Otras alteraciones pueden ser temblores, falta de sensibilidad y pérdida total o parcial de alguno de los sentidos.

Un **déficit cognitivo** es aquel que afecta al aprendizaje, memoria, déficits de atención, alteración del lenguaje y del razonamiento lógico-deductivo. Este grupo de secuelas es especialmente grave en pacientes en etapas de aprendizaje, ya que les condicionará el resto de su vida el no haber podido aprender con normalidad en esta etapa.

Las **alteraciones de la comunicación** derivan de la alteración de habilidades físicas y cognitivas mencionadas anteriormente. En el caso de alteraciones físicas, el paciente sufrirá una alteración de la expresión oral, debido a la pérdida de control de los músculos encargados del habla. La alteración de la expresión puede producirse también por la incapacidad de encontrar las palabras adecuadas. A este fenómeno se le denomina afasia. También hay casos de la pérdida de la capacidad de escritura como consecuencia de la limitación motora en un brazo. Existen casos de pérdida de la comprensión al existir una pérdida de la audición o de la interpretación de sonidos.

Las **alteraciones de la conducta** o emocionales están asociadas con los cambios de carácter que se producen después del daño cerebral. El origen de estas alteraciones se podría considerar de dos formas: falta de inhibición, lo que se traduce en conductas física o verbalmente agresivas. Otros pacientes, en cambio, sufren un exceso de inhibición que tiene como consecuencia una falta de iniciativa, muestra apatía, no disfruta de las cosas que antes le gustaban, sensación de agotamiento y apenas muestra sus emociones.

3.4. E-Salud

La e-Salud o e-health, como se conoce en inglés, hace referencia a la práctica de cuidados sanitarios mediante el uso de las tecnologías de la información y las comunicaciones (TIC).

El término comenzó a aparecer hace más de quince años en diversas publicaciones científicas y como muchos conceptos nuevos, no estaba claramente definido. Fue entonces cuando G. Eysenbach (Eysenbach, 2001) estableció una definición muy ajustada y que hasta hoy no ha sufrido apenas modificaciones. En su artículo “What is e-health”, que hablaba también del auge de los conceptos del e-commerce y e-buisness. La definición propuesta por Eysenbach fue la siguiente: “La e-Salud es un campo emergente en la intersección de la informática médica, la salud pública y los negocios, referido a los servicios sanitarios y la información transmitida o mejorada a través de internet y las tecnologías relacionadas. En un sentido más amplio, el término representa no sólo un desarrollo técnico, sino también un estado mental, una forma de pensar, una actitud y un compromiso con un pensamiento conectado, global, para mejorar la sanidad local, regional y globalmente a través del uso de las TIC”

Además de esta definición, Eysebach afirmaba que la e-Salud no sólo estaba relacionada con el uso de la tecnología, sino que implicaba diez premisas fundamentales y que empezaban con la misma letra, en inglés. Los diez términos introducidos son los siguientes:

- Eficiencia (efficiency): se basa en abaratar los costes gracias a las ventajas que proponen las TIC.
- Mejora de la calidad (enhancing quality) del cuidado: gracias al uso de dispositivos se pueden obtener constantes vitales a tiempo real, se disponen de grandes cantidades de datos y se puede monitorizar de forma constante a un paciente.
- Basada en la evidencia (evidence based): es importante evaluar la validez de los datos proporcionados por los dispositivos en todo momento.
- Empoderamiento (empowerment) de consumidores y pacientes: esto surge gracias a la cantidad de información de la que disponen los pacientes. Principalmente gracias a internet, el paciente es mejor conocedor de lo que padece y esto se traduce en una mejor descripción al personal sanitario.
- Fomento (encouragement) de una nueva relación entre el paciente y el profesional, donde haya decisiones compartidas.
- Educación (education) a través de fuentes digitales por parte de los médicos a través de medios en línea.
- Permitir (enabling) el intercambio de información y la comunicación de manera estandarizada entre las instituciones de salud.
- Extender (extending) el alcance de la atención. La e-Salud permite a los consumidores obtener fácilmente servicios en línea en todo el mundo. Estos servicios pueden ir desde consejos a productos farmacéuticos.
- Ética (ethics): la e-Salud plantea nuevas cuestiones éticas como la práctica profesional en línea, la privacidad, o cuestiones de equidad.
- Equidad (equity): hacer que la e-Salud sea más equitativa es algo fundamental. La desigualdad puede ocurrir entre la gente que no tiene dinero para acceder a ella, o no poseen las habilidades para el uso de los dispositivos necesarios, como puede ser un ordenador.

Además, afirma que la e-Salud ha de ser fácil de usar, entretenida y emocionante, tres factores que suponen el éxito del término para usos médicos.

3.5. Alternativas en la competencia

En el mercado ya existen sistemas para ayudar a pacientes con DCA y para evitar ataques de epilepsia. Es por ello necesario realizar un estudio de las alternativas que existen para evitar realizar un trabajo que ya se encuentre a disposición de los pacientes. Con este estudio también se pueden obtener ideas que complementen el sistema con el fin de ofrecer nuevas características que proporcionen un valor añadido.

3.5.1. Proyectos similares

Hay sistemas de todo tipo: desde simples consejos, prevención mediante cuestionarios, hasta medidores del ritmo cardíaco en tiempo real. A continuación, se describirán los sistemas más interesantes o más extendidos.

3.5.1.1. EpSMon



Ilustración 4. Acción SUDEP.

Fuente: sudep.org.

Aplicación perteneciente a acción SUDEP, esta asociación británica trata de concienciar de los riesgos de la epilepsia incluyendo la muerte súbita. Entre otros muchos servicios disponen de una aplicación cuyo fin es mostrar los factores de riesgo que tiene cada paciente de sufrir SUDEP y ataques de epilepsia.

Para ello se basa en un cuestionario que ha de completarse cada tres meses. En él aparecen preguntas de todo tipo, relacionadas con la ansiedad, el estado de ánimo, etc. A partir de las respuestas proporcionadas por el paciente se elabora un informe que ha de entregarse al médico para una posterior evaluación.

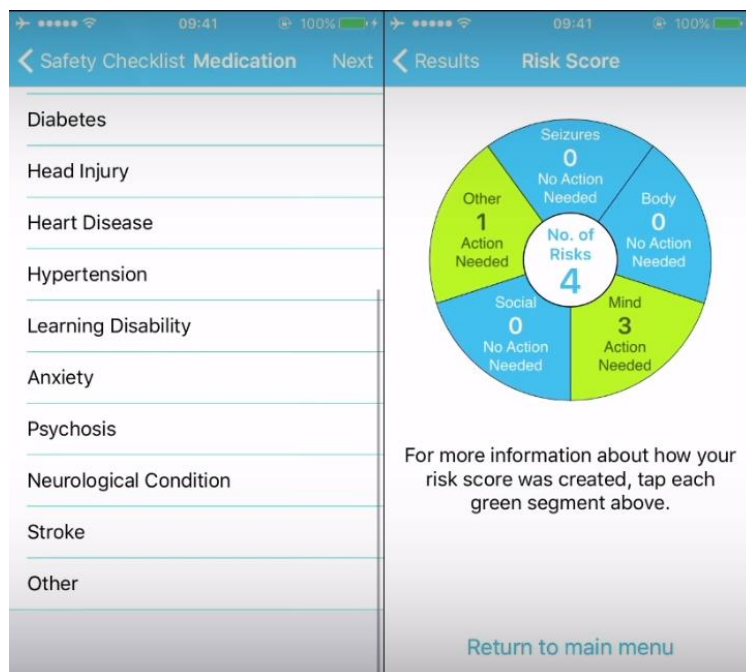


Ilustración 5. Cuestionario y resultado EpsMon.

Fuente: Apple Store.

Entre los aspectos positivos destacan que, al ser una asociación sin ánimo de lucro, la aplicación es totalmente gratuita. Otro aspecto positivo es que, al ser un cuestionario detallado, el resultado puede ser muy preciso. Esto junto a la evaluación de un doctor puede ser de gran utilidad.

Sin embargo y como principal inconveniente, esta aplicación no es válida para la prevención en tiempo real de ataques de epilepsia, ya que no dispone de ningún sensor capaz de monitorizar al paciente, y la aplicación carece de sistema de aviso en caso de ataque.

3.5.1.2. Epilepsy Tool Kit

Como EpSMon, Epilepsy Tool Kit también pertenece a una asociación. En este caso se trata de Epilepsy Society, que es el principal proveedor de servicios de epilepsia del Reino Unido. Abarcan todos los campos que conciernen a la epilepsia, desde la sensibilización de la sociedad hasta la inversión en investigación para buscar las causas subyacentes de la epilepsia y la optimización del control de las convulsiones.

Epilepsy Tool Kit es sólo una pieza más en Epilepsy Society, pero es capaz de proporcionar valor por sí sola. Cuenta con un registro de ataques, donde el paciente dice

si ha sufrido un ataque, cómo ha sido y cuándo ha ocurrido. Este historial puede ser enviado mediante correo electrónico al personal médico que se desee.

Como valor añadido, dispone de un gestor de medicación donde el usuario podrá consultar su actual tratamiento, así como recibir alertas cuando sea la hora de la dosis. También destaca una sección denominada primeros auxilios, donde se muestran consejos para ayudar al paciente que sufre una convulsión.



Ilustración 6. Menú y registro de ataque.

Fuente: Play Store.

Sus principales ventajas son la presencia de un registro de ataques bastante detallado. Además, este registro es posible enviarlo por correo electrónico a cualquier persona, pero la idea es que se envíe al médico del paciente para una posterior evaluación.

También es un aspecto positivo la gestión de la medicación, con un listado de los tratamientos que sigue el paciente y con la posibilidad de enviar notificaciones cuando sea la hora determinada.

Un aspecto negativo es la ausencia de prevención de ningún tipo, ni manual ni monitorizada, con el fin de evitar crisis epilépticas. Este apartado es esencial y se echa en falta ya que la prevención y alerta en caso de ataque en tiempo real es un factor diferencial en otros sistemas del mercado.

3.5.1.3. EpDetect

Esta aplicación fue desarrollada por Kenneth Austin y Robert Dlugosz. El hijo de Kenneth es epiléptico y su padre buscaba una solución en el mercado para los ataques epilépticos. Decidió entonces desarrollar él mismo el sistema y para ello contó con la ayuda de Robert, que anteriormente había desarrollado la aplicación móvil “Do not Touch Me”, que proporciona envíos de SMS con localización GPS. Ambos acordaron desarrollar EpDetect combinando la experiencia de Kenneth con el proyecto de Robert.

El resultado es una aplicación capaz de detectar ataques epilépticos y de avisar a alguien de lo que está ocurriendo vía SMS. Esta detección se hace mediante el acelerómetro del dispositivo móvil que contenga la aplicación. También dispone de un sistema de cancelación de avisos en caso de falso positivo. De este modo no se avisaría a la persona de que el paciente está recibiendo un ataque. El paciente dispone de treinta segundos para cancelar el aviso.



Ilustración 7. Epdetect: detección de ataque.

Fuente: epdetect.com.

Como aspectos positivos destacan la monitorización en tiempo real de los pacientes para alertar en caso de ataque epiléptico. Esto es un factor diferencial de la aplicación, ya que no existen muchas que realicen un seguimiento a tiempo real.

Su principal problema es que, en el resto de los aspectos, es una aplicación bastante pobre. No tiene ningún extra destacable más allá de la cancelación manual en caso de falso positivo. Además, una de las desventajas de utilizar el acelerómetro para detectar ataques epilépticos es que no se disponen datos de utilidad para que un médico los

pueda analizar. En este sistema, el registro del movimiento no sirve para hacer un seguimiento profesional del paciente. Para esto sería necesario que la aplicación fuera capaz de obtener datos de otros sensores y al componerse sólo de una aplicación móvil, es imposible que obtenga otros parámetros, como el ritmo cardíaco.

3.5.1.4. Daño Cerebral

Esta aplicación está desarrollada por el Servicio de neurorehabilitación de Hospitales Nisa, un grupo privado de servicios de salud líder en el sector en la Comunidad Valenciana. Esta aplicación está destinada a proporcionar consejos y pautas de recomendación para el cuidado del paciente con DCA. De este modo se consigue ayudar también a las personas de todo el entorno del paciente, facilitándoles la intervención.

Ofrece consejos en todos los aspectos en los que el DCA puede incidir, como pueden ser consejos de movilidad, cognición, conducta, comunicación o alimentación. A su vez también proporciona consejos para facilitar la vida diaria, como pueden ser actividades a realizar, readaptación al entorno, recursos sociales o consejos orientados a la familia.

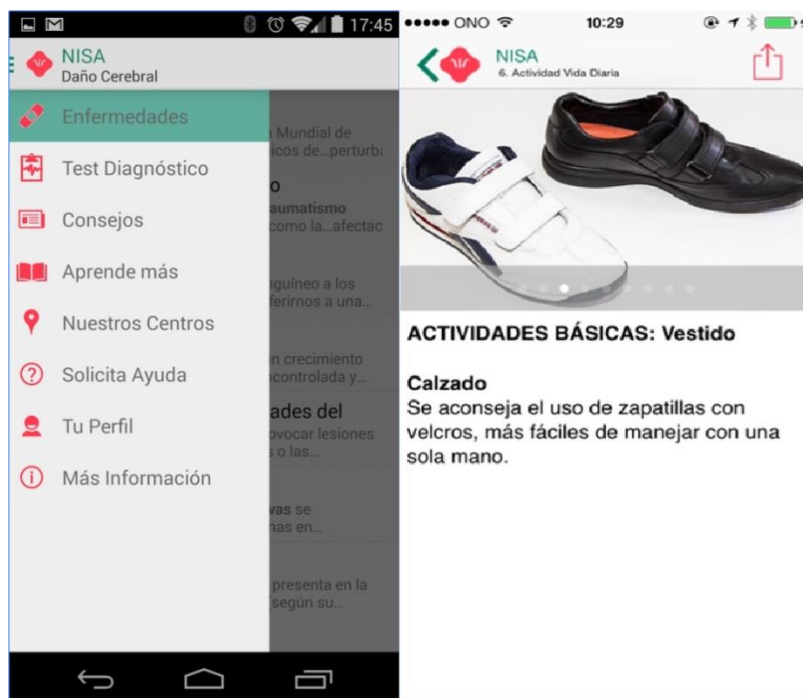


Ilustración 8. Daño Cerebral: Menú y consejos.

Fuente: Play Store.

Como principal ventaja destaca la gran cantidad de consejos que proporciona no sólo para los temas estrictamente médicos, sino también sociales o familiares. Estos consejos que van más allá del tema médico son realmente un valor añadido para la aplicación, ya

que ayudan a mejorar la calidad de vida y la integración social de los pacientes con DCA.

El principal aspecto negativo es la ausencia de monitorización de estos pacientes, en concreto, para detectar ataques de epilepsia y avisar a la persona a cargo de él. Al no tener monitorización de ninguna constante vital, es imposible un seguimiento médico en detalle, más allá de los formularios que dispone la aplicación.

3.5.1.5. SeizAlarm

La idea de esta aplicación surgió a raíz de que a Greg Pabst, su creador, le diagnosticaran epilepsia en 2006. Según afirma él, ha tenido auras cada semana desde entonces. Ante la incapacidad de avisar a alguien si comenzaban a ocurrir estos episodios planteo una herramienta de ayuda.

La aplicación se basa en el uso de Apple Watch, siendo exclusiva para este wearable. Permite avisar cuando se está sufriendo un ataque de manera rápida. Para ello puede enviar un email, sms o una llamada telefónica. Hay una versión gratuita que solo permite el registro de ataques, la funcionalidad de poder avisar está disponible en una versión de pago. Permite también la monitorización mediante el ritmo cardíaco.

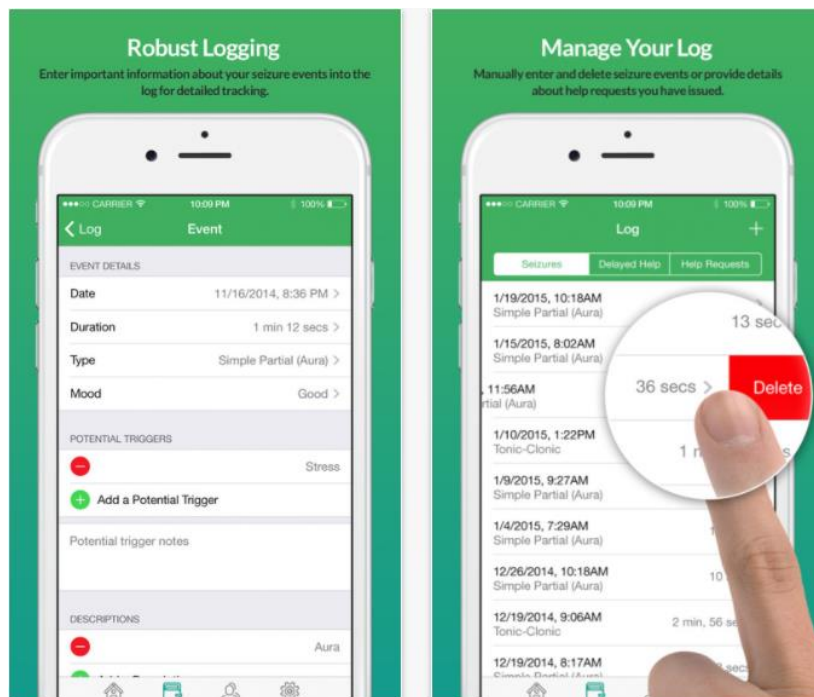


Ilustración 9. SeizAlarm.

Fuente: Apple Store.

Los aspectos positivos son la capacidad de monitorizar en tiempo real con el fin de detectar por sí solo un ataque cardíaco. Además, para esto utiliza el ritmo cardíaco, constante vital fiable a la hora de predecir un ataque.

En cuanto a los aspectos negativos, el uso exclusivo para Apple Watch hace que no todo el mundo pueda acceder a este servicio, ya que es un wearable que se podría considerar caro. Además, la aplicación no permite ningún seguimiento por parte del personal médico, por lo que su funcionalidad no va más allá de la detección de ataques epilépticos.

3.5.1.6. DCA Epilepsia

El proyecto es el resultado del trabajo de fin de grado de mi compañera Gala M. García. Gracias a su trabajo se está realizando este proyecto que busca completar las funcionalidades de un sistema capaz de detectar ataques de epilepsia en pacientes con DCA. Sin embargo, es necesario un estudio de su trabajo para determinar el alcance del mismo y para obtener conclusiones a cerca de lo que se puede añadir.

LA descripción de su proyecto se encuentra en este apartado debido a que es un proyecto totalmente funcional por sí solo. En él, la aplicación se conecta con la pulsera Angel Heath Sensor, que monitoriza en tiempo real las constantes vitales necesarias, en este caso, el ritmo cardíaco. En la aplicación se establece un rango de ritmo cardíaco que cuando se supere, alertará vía SMS a la persona que se desee.

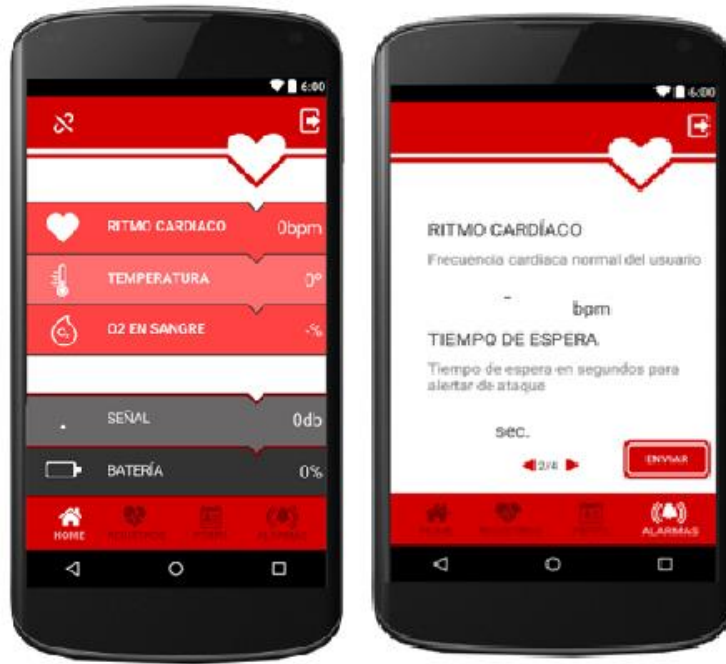


Ilustración 10. DCA epilepsia: home y alarmas.

Fuente: Monitorización de pacientes con pulseras inteligentes.

Como aspectos positivos destacar la monitorización del paciente en tiempo real, con la presencia del sistema de alertas en caso de que se produzca un ataque epiléptico. También destacar que se produce un registro de la actividad cardíaca para su posterior evaluación.

Como aspectos negativos, el sistema se encuentra almacena toda la información en local, es decir, los datos recibidos por la pulsera no se pueden enviar a ninguna base de datos para que puedan analizarse externamente. También destacar que la pulsera Angel Heath Sensor dejó de comercializarse y el proyecto se abandonó, por lo que es imposible utilizar este sistema hecho a medida para este wearable.

3.5.1.7. Embrace

Embrace Empatica es un proyecto realizado por un grupo de ingenieros del MIT (Instituto de Tecnología de Massachusetts) con el objeto de detectar ataques de epilepsia. Es una de los sistemas más prometedores, ya que tiene clientes como Intel o la NASA.

El sistema gira en torno a su pulsera creada especialmente para ello. Dicha pulsera, denominada embrace, es capaz de monitorizar en tiempo real las constantes vitales de los pacientes. Para ello dispone de un sensor electrodérmico, de temperatura, un

giroscopio y un acelerómetro. Con todos estos sensores en funcionamiento afirman que es posible detectar un ataque de epilepsia con gran precisión. Desde Embrace afirman que este sistema está especialmente destinado a niños, ya que puede mandar alertas a quien se desee en caso de que se produzca un ataque. La pulsera se financió en Indiegogo, con una cantidad necesaria de 100.000 dólares. Sin embargo, su lanzamiento tuvo tanto éxito que consiguió recaudar 288.000 dólares.

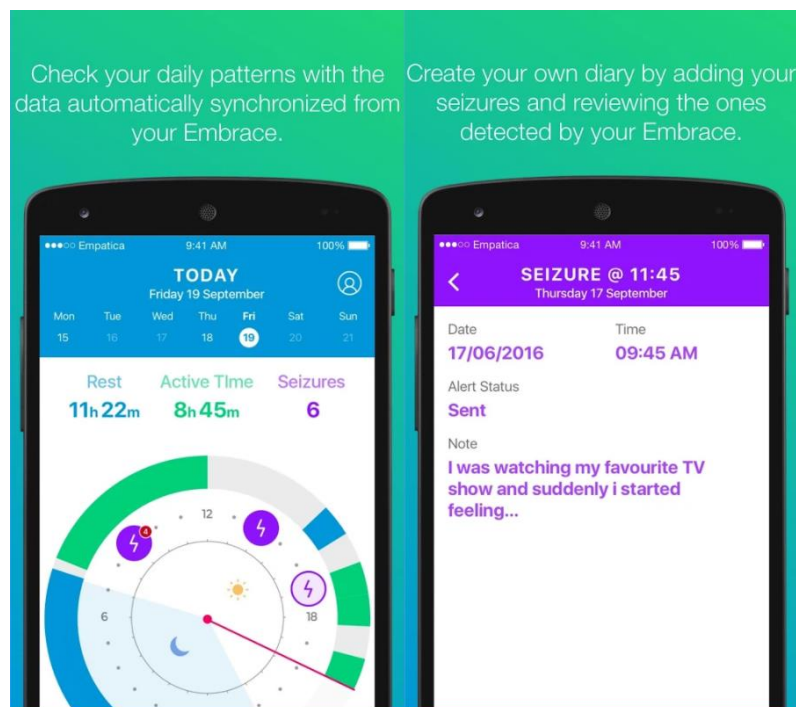


Ilustración 11. Embrace: actividad y registro de ataque.

Fuente: Play Store.

Como principales ventajas destacan la monitorización a tiempo real de los pacientes con epilepsia. También dispone de un sistema de alertas cuando se produce un ataque. Gracias a la gran cantidad de sensores es capaz de detectar un ataque de epilepsia con gran precisión. Además, dispone de un seguimiento de los ataques para que el personal médico pueda observarlos y evaluarlos.

Sin embargo, al principio hay que indicarle manualmente si ha ocurrido un ataque de epilepsia. Esto sucede debido a que los algoritmos de aprendizaje que utiliza necesitan de datos previos de entrenamiento con los cuales posteriormente decidirá si se produce un ataque o no. Además, el precio es algo elevado y existen constantes problemas con retrasos de meses en los envíos.

3.5.1.8. PulseGuard

Este sistema fue desarrollado inicialmente por Tom Perry, que padece síndrome de Dravet (una forma grave de epilepsia), ya que no tenía forma de prevenir los ataques de epilepsia que padecía.

El sistema es un monitor de ritmo cardíaco en tiempo real que detecta una distorsión en el ritmo cardíaco habitual, con lo que es posible detectar ataques de epilepsia. Además, permite un seguimiento del historial del ritmo cardíaco que se obtiene.

En la web se ofrecen varios paquetes, siendo el más sencillo uno compuesto por una pulsera especialmente desarrollada para el sistema y un iPad mini modificado para conectarse con la pulsera. Este iPad también está modificado para el uso exclusivo en este sistema.

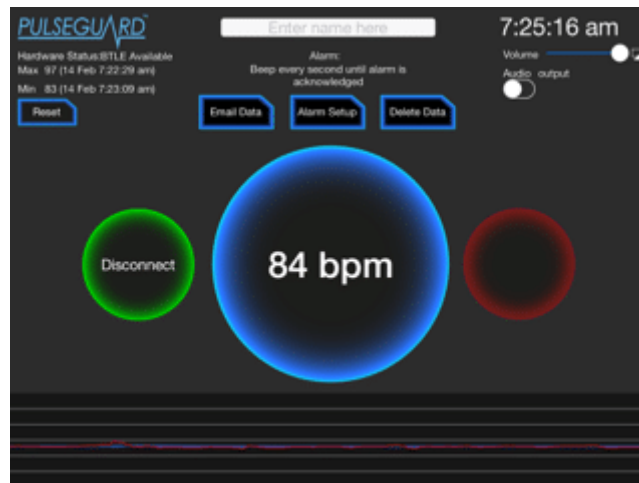


Ilustración 12. PulseGuard: aplicación en iPad.

Fuente: pulseguard.org.

Como principales ventajas se encuentran la monitorización del ritmo cardíaco y el aviso en tiempo real de un ataque de epilepsia, gracias al sensor de ritmo cardíaco que dispone la pulsera. Además, permite enviar los datos a quien se desee para su posterior evaluación.

Como principal desventaja se encuentra el hardware utilizado. Es el único sistema de la lista que no sólo utiliza un wearable personalizado, sino que el hardware receptor de datos (iPad) también está modificado. Esto hace que las personas que quieran adquirir PulseGuard tengan que realizar un gran desembolso (1.100 dólares el pack más

económico), por lo que es inaccesible a una gran parte de los afectados por el DCA y la epilepsia.

3.5.2. Conclusiones

Han sido muchos los sistemas analizados, desde simples aplicaciones informativas que ofrecen consejos por escrito, a hardware especializado capaz de monitorizar en tiempo real. Sin embargo, la gran mayoría de sistemas son puramente informativos y aunque la información que proporcionan puede resultar bastante útil, no van más allá de los consejos para mejorar la vida de los afectados.

Por un lado, hay que diferenciar las aplicaciones orientadas únicamente al tratamiento de la epilepsia: en este apartado sí existen sistemas capaces de monitorizar y gestionar alarmas. Estos sistemas, al requerir de hardware, son los más costosos. En algunos casos es posible realizar un seguimiento por parte de terceras personas, por ejemplo, enviando los datos por correo electrónico.

Por otro lado, están las aplicaciones que se centran en el DCA: estas son las aplicaciones informativas anteriormente mencionadas. Algunas ofrecen un calendario para introducir manualmente los ataques, o consultar la medicación.

Para resumir el punto anterior se ha realizado una tabla que intenta destacar los aspectos más importantes a la hora de cumplir los objetivos propuestos en este trabajo. Se busca un sistema capaz de monitorizar y alertar de un ataque de epilepsia, con seguimiento minucioso de las constantes vitales obtenidas, multiplataforma y económico para que sea accesible a todo el mundo.

	Sistema Operativo		Hardware adicional		Seguimiento mediante*		Monitorización	Diagnóstico con cuestionario	Registro de ataques manual	Alerta de ataque	Informativa	Precio
	Android	IOS	Específico	Comercial	email	Sistema web						
EpSMon												-
Epilepsy Tool Kit												-
EpDetect												-
Daño Cerebral												-
SeizAlarm												6,99 €/mes
DCA Epilepsia				**								-
Embrace												Desde 249 \$
PulseGuard												Desde 1.100 £

Tabla 5. Comparativa de sistemas en el mercado.

(*) El seguimiento se realiza por el personal médico de manera externa, es decir, sin tener que proporcionar el dispositivo donde se monitoriza.

(**) La pulsera utilizada es Angel Health Sensor, que ya no se comercializa.

Como se puede observar en la tabla, se puede diferenciar claramente los dos tipos de aplicaciones o sistemas: lo informativos o manuales y los que monitorizan a tiempo real. Mientras que el primer grupo ofrece sus servicios de forma gratuita, para el otro es necesario realizar una inversión en hardware, en algunos casos muy elevado.

Además, sólo Embrace dispone de sistema web capaz de realizar un seguimiento exhaustivo de las constantes vitales monitorizadas. El resto, como mucho, envían por email los datos obtenidos, lo que limita la capacidad de interacción con el personal médico. No se han tenido en cuenta en este apartado aquellas aplicaciones que para mostrar un seguimiento requieran mostrar los datos al médico usando el móvil.

En el apartado de hardware adicional cabe decir que la mayoría utilizan hardware desarrollado específicamente para esta finalidad. En el caso de SeizAlarm se trata de hardware comercial, Apple Watch, que por su elevado precio (desde 339 euros) resulta más caro que algún sistema con hardware específico.

Se puede apreciar que muchas aplicaciones son multiplataforma, algo totalmente normal actualmente. Sin embargo, hay que destacar que dos de los cinco sistemas analizados capaces de alertar de ataques de epilepsia no son multiplataforma, por lo que sus servicios quedan limitados a la plataforma disponible en cada caso. En el caso de Epdetect incluso no se ha testado en dispositivos modernos, con una versión de Android actual. Esto puede indicarnos que el desarrollo se ha abandonado y que no se esperan mejoras en el futuro.

Como conclusiones finales, cabe indicar que el sistema que más se acerca al deseado es Embrace. Posee un hardware fiable, una monitorización mediante varios parámetros y una web de seguimiento.

Las principales desventajas son, como he mencionado en el apartado anterior, que es necesario entrenar la aplicación con ejemplos de situaciones de ataque. Esto se traduce en que es posible que no detecte las primeras veces que ocurren los ataques, por lo que la sensación de inseguridad es constante. Es mejor dar falsos positivos iniciales e ir refinando el algoritmo que pecar de no alertar si se produjera un ataque.

El precio también es un aspecto que puede ser negativo. Obviamente habrá personas con recursos para invertir ese dinero, pero muchas otras no disponen de esta cantidad (desde 249 dólares).

Otro inconveniente, en este caso médico, es que el modelo de 249 dólares no mide el ritmo cardíaco. En su lugar utiliza una combinación de sensores que no son los más útiles para realizar un seguimiento posterior. El caso ideal sería que monitorizara el ritmo cardíaco, pero para esto habría que adquirir al modelo E4 Wristband, cuyo precio es de 1.690 dólares.

Por estos motivos es necesario un sistema multiplataforma, que monitorice el ritmo cardíaco, comunique los datos a una web externa de seguimiento, y que sea fiable y económico, para que pueda llegar al máximo de pacientes posibles.

3.6. Estudio smartbands

A continuación, se realizará un estudio de las smartbands más destacadas o que más se adaptan a las necesidades del proyecto. Este estudio ha sido necesario debido a la sorprendente cancelación del proyecto Angel Sensor, pulsera con la que inicialmente se pensaba realizar el proyecto.

Open source wearable Angel shuts down

By [Jonah Comstock](#) | December 02, 2016

SHARE  206

Angel, a company that has been working since 2013 on an open source wearable tracker that could be programmed for different use cases, has shut down the project and, likely, the company.

The company announced the news via a large banner [on its website](#) reading "This project is no longer active". Angel executives did not respond to MobiHealthNews's request for an interview. Bob Troia, known as "Quantified Bob" in quantified self circles, spotted the announcement and posted about it on Twitter and on the Quantified Self forum.

"Well, looks like the Angel Sensor folks have (finally) officially thrown in the towel," [he wrote](#). "Not really a surprise, as they had gone silent for nearly a year after delivering their crowdfunded product over two years late. They did release code for their open-source SDK, and there is a community of developers who have forked it on GitHub3 to continue development. Too bad they gave up, as the promise of a truly open source wearable with an array of useful sensors is lacking in the QS space."



Ilustración 13. Artículo sobre la cancelación de Angel Sensor.

Fuente: [mobihealthnews.com](#)

3.6.1. Microsoft Band2

La segunda pulsera de Microsoft, centrada en el deporte, ofrece una gran cantidad de sensores, pero no dispone de un SDK abierto. En su lugar tiene una API con la que poder obtener datos. Desgraciadamente, con esta API no se pueden realizar medidas en tiempo real, por lo que no es posible realizar un seguimiento continuo de manera fácil. Además, Microsoft ha dejado de venderlas de manera oficial, y ha cancelado la tercera versión de esta pulsera, lo que es una clara señal de que Microsoft ha dejado de lado a las smartbands, por el momento.



Ilustración 14. Microsoft Band 2.

Fuente: Microsoft.com.

3.6.2. Razer Nabu X

Versión mejorada de la primera smartband de Razer, Nabu, destaca por la desaparición de la pantalla y la inclusión de tres leds que permiten notificaciones. Razer proporciona un SDK para Android e IOS al desarrollador para que lo utilice. Destaca su precio ajustado, quizás esto explica la ausencia de muchos sensores, como ritmo cardíaco, temperatura corporal, etc.

Es por esto por lo que queda descartada, ya que sin estos sensores resulta imposible alcanzar los objetivos propuestos.



Ilustración 15. Razer Nabu X.

Fuente: razerzone.com.

3.6.3. Fitbit Charge 2

Charge 2 es otra de las smartbands centrada en el deporte. Una de los varios modelos de los que dispone Fitbit, es este el que ofrece funcionalidades que más se ajustan al proyecto. Destaca el sensor de ritmo cardíaco y la gran pantalla, en contraposición con los otros modelos más sencillos y con un precio más ajustado.

Sin embargo, para ninguno de los modelos se proporciona SDK, por lo que resulta difícil escogerla para el proyecto. En su lugar ofrece una API que resulta insuficiente para medir a tiempo real.



Ilustración 16. Fitbit Charge 2.

Fuente: fitbit.com.

3.6.4. Atlas Wristband 2

El segundo modelo de la gama Wristband está enfocado claramente al deporte, aunque esto no la descarta para el proyecto. Aunque dispone de sensor de ritmo cardíaco y acelerómetro, se echan en falta algún sensor extra que pueda detectar constantes vitales. Además, no dispone de SDK ni de API, ya que todo gira en torno a la App que los propios desarrolladores de Atlas han creado.



Ilustración 17. Atlas Wristband 2.

Fuente: atlaswearables.com.

3.6.5. Samsung Simband 2

Uno de los wearables más prometedores. La marca coreana creó la primera pulsera modular con licencia opensource, para que los desarrolladores tanto de software como de hardware crearan mejoras para esta pulsera. La modularidad y la gran cantidad de sensores ofrecen muchas posibilidades para el usuario. Además, dispone de SDK abierto, por lo que sería una gran candidata. Desgraciadamente no se encuentra a la venta para el público, estando disponible solo para desarrolladores. Esto hace que la expansión del proyecto para ayudar realmente en casos prácticos sea imposible.

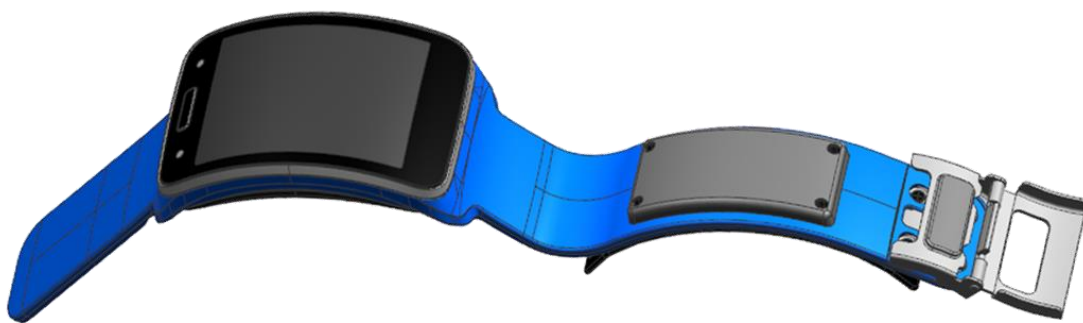


Ilustración 18. Samsung Simband 2.

Fuente: simband.io.

3.6.6. Xiaomi Mi Band 2

Quizás la smartband con precio más ajustado, se puede encontrar a partir de 25 euros aproximadamente. La pulsera de origen chino viene con sensor de ritmo cardíaco como principal característica. Se echan en falta algunos sensores como el de temperatura, que ayudaría al cumplimiento de objetivos. Tampoco dispone de SDK abierto y aunque la comunidad ha hecho avances con ingeniería inversa, es insuficiente y resulta complicado desarrollar el proyecto sin apoyo del fabricante.



Ilustración 19. Xiaomi Mi Band 2.

Fuente: mi.com.

3.6.7. Garmin Fenix 5

Garmin es una empresa que dispone de cantidad de smartbands con una gran horquilla de precios. Sin embargo, es el modelo Fenix 5 la que más posibilidades ofrece.

Fenix 5 está claramente orientada al deporte, contando con cantidad de perfiles como ciclismo, natación, golf, etc. Aun así, ofrece unos cuantos sensores verdaderamente útiles para el desarrollo del proyecto, como el medidor del ritmo cardíaco o acelerómetro.

Además, Fenix ofrece el SDK de todas sus smartbands de forma gratuita para poder desarrollar apps, pero esto no justifica su elevado precio, siendo el más alto de los comparados.



Ilustración 20. Garmin Fenix 5.

Fuente: buy.garmin.com.

3.6.8. Empatica

La pulsera de la empresa empática es una de las más sofisticadas. Con sensor de respuesta galvánica de la piel hace posible detectar un ataque de epilepsia. Además dispone de acelerómetro y sensor de temperatura corporal.

Ganadora de diversos premios es la smartband más enfocada a la salud, con una gran app creada por Embrace que la hace realmente fiable. Dispone de una API abierta que obtiene datos en tiempo real.



Ilustración 21. Embrace Empatica.

Fuente: empatica.com.

3.6.9. Conclusiones

Después de analizar las pulseras de manera general, se va a realizar una tabla a modo de resumen donde se mencionan especificaciones técnicas de cada una, cosa necesaria para decidir la smartband con la que se realizará el proyecto.

	Pantalla	Android	IPhone	SDK	API	Precio
Microsoft Band 2						210 €
Razer Nabu X						60 €
Fibit Charge 2						160 €
Angel Sensor						-
Atlas Wristband 2						200 €
Samsung Simband 2						-
Xiaomi Mi Band 2						25 €
Garmin Fenix 5						599 €
Empatica						249 €

Tabla 6. Comparativa general smartbands.

También es necesario comparar los sensores que proporciona cada una, para así poder delimitar la funcionalidad que tendrá el sistema en un futuro y el máximo grado de ampliación de funcionalidad con cada smartband.

	Ritmo cardíaco	Acelerómetro	Temperatura corporal	GPS	Girómetro	Respuesta galvánica de la piel	Vibración	Sensor de actividad	Oxígeno en sangre	Detección de gestos manuales
Microsoft Band 2										
Razer Nabu X										
Fibit Charge 2										
Angel Sensor										
Atlas Wristband 2										
Samsung Simband 2										
Xiaomi Mi Band 2										
Garmin Fenix										
Empatica										

Tabla 7. Comparativa de sensores de las smartbands.

3.7. Estudio plataformas

Elegir la mejor plataforma de desarrollo es muy importante si se quiere realizar una aplicación multiplataforma. Existen diferentes opciones en cuanto a resultado y en cuanto a metodología, por lo que es necesario realizar un estudio previo valorando aspectos positivos y negativos.

Como punto de partida se busca la realización de una aplicación multiplataforma, que soporte comunicación BLE (Bluetooth de bajo consumo) y que permita un desarrollo ágil. Se valorará también el lenguaje o lenguajes utilizados, la curva de aprendizaje y la facilidad del diseño de interfaces.

3.7.1. Aplicaciones nativas

En este caso, la aplicación está optimizada para cada dispositivo, consiguiendo así el mejor resultado en cuanto a rendimiento y fluidez. Es más eficiente al cargar gráficos, gracias al uso de API nativa.

Con esta opción se entiende el uso de alguna plataforma que aproveche el código fuente para realizar la compilación para las dos plataformas (Android e IOS). Queda totalmente descartado trabajar en dos versiones de la aplicación totalmente separadas.

3.7.1.1. React Native

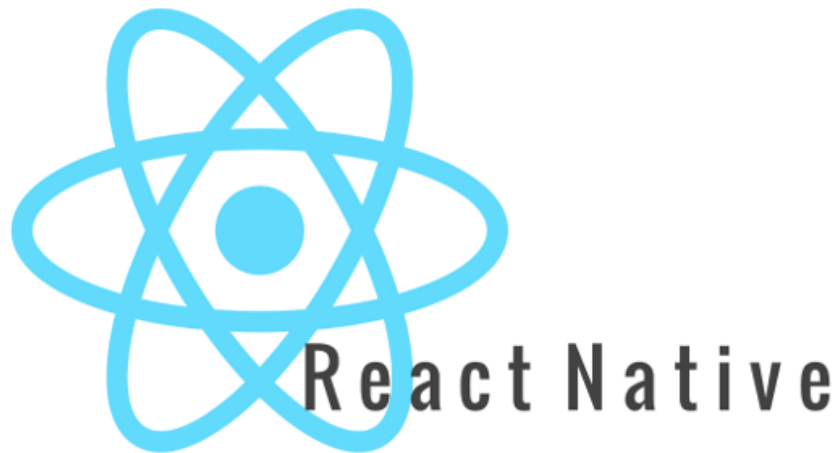


Ilustración 22. Logotipo React Native.

Fuente: Facebook.github.io.

Plataforma perteneciente a Facebook es opensource para el desarrollo de apps móviles. Este framework utiliza JavaScript como principal lenguaje, pero la app se renderiza utilizando componentes nativos. Esto es posible mediante una capa intermedia que traduce desde javascript al lenguaje nativo del dispositivo. Gracias a esto el rendimiento mejora considerablemente respecto a las híbridas.

Otra ventaja viene relacionada con la anterior: al renderizar en componentes nativos, la apariencia final es más parecida a los estilos de cada sistema. Gracias a ello se mejora la experiencia de usuario al verse familiarizado con el estilo y la colocación de los elementos.

Como principal desventaja destaca el propio JavaScript, un lenguaje que una vez introducido en TypeScript complica la vuelta hacia atrás. Se creó inicialmente para IOS, por lo que para algunos componentes de Android se necesitan conocimientos avanzados sobre la plataforma. Además, la curva de aprendizaje sería grande, ya que no se dispone de experiencia en el uso de esta plataforma y al ser algo relativamente nuevo la comunidad es más pequeña.

3.7.1.2. NativeScript



Ilustración 23. Logotipo de NativeScript.

Fuente: nativescript.org.

La empresa Telerik define a NativeScript como “un framework open source para crear aplicaciones nativas con Angular, TypeScript o JavaScript”. Al igual que React Native, hay una capa intermedia que traduce a interfaces nativas. Al ser el resultado final una app nativa, el rendimiento es bastante satisfactorio para proyectos pesados. Sin embargo, el peso de las aplicaciones es grande en comparación con otras plataformas, factor diferencial a la hora de la descarga con datos móviles. Además, presenta pocos plugins oficiales por lo que el desarrollo normal del proyecto se puede ver interrumpido. Por último, destacar la curva de aprendizaje que presentaría, al desconocer el framework, sobre todo a la hora de desarrollar interfaces.

3.7.1.3. Xamarin



Ilustración 24. Logotipo de Xamarin.

Fuente: xamarin.com.

Otra de las plataformas que generan aplicaciones nativas. En este caso el propietario es Microsoft, con todo lo que ello conlleva (comunidad, soporte, guías). Utiliza C# como lenguaje de programación y .net. Una de las principales ventajas es poder visualizar diferentes analíticas sobre la aplicación, consiguiendo así un seguimiento de manera sencilla. Como principales contras están las limitaciones en cuanto al uso de librerías de terceros y la necesidad de crear código para Android e IOS en tareas complejas como el uso de BLE, requisito indispensable para el desarrollo del proyecto.

3.7.2. Aplicaciones híbridas

A diferencia de las aplicaciones nativas, las apps híbridas utilizan el WebView del dispositivo para mostrar la aplicación. Este WebView depende de cada dispositivo, pero generalmente cumplen con lo esperado. La integración con el SO depende también del dispositivo, permitiendo acceder al hardware e incluso a librerías. Este acceso sí que se realiza de forma nativa mediante plugins propios de la plataforma o de terceros. En general se puede decir que la reutilización del código es prácticamente del 100%, y que los lenguajes utilizados son JavaScript, Css y HTML.

3.7.2.1. Cordova/PhoneGap



Ilustración 25. Logotipo Apache Cordova.

Fuente: cordova.apache.org.

Es necesario hablar de ambos simultáneamente porque están estrechamente relacionadas. La empresa Nitobi creó PhoneGap en 2008 en el iPhoneDevCamp, con el fin de crear aplicaciones híbridas de manera sencilla. En 2011 donó el código fuente a Apache, ya que creían que podrían obtener mejores resultados trabajando juntos. Ese mismo año Adobe adquirió Nitobi, siendo propietaria de la marca PhoneGap. Es entonces cuando Adobe cambia oficialmente el nombre del proyecto para llamarse Cordova. A día de hoy PhoneGap es una extensión de Cordova con algunos servicios añadidos, tales como el build service o la PhoneGap Developer app, cuya principal utilidad destaca en el testeo o debugging.

Cordova es una plataforma gratuita para el desarrollo sobre Android, IOS, Windows Phone, BlackBerry, etc. El núcleo de las aplicaciones se basa en HTML y css3 para las interfaces y JavaScript para la lógica. Proporciona acceso a APIs nativas y librerías, permitiendo así el acceso al hardware como cámaras o acelerómetros. Además, dispone de una de las mayores comunidades en cuanto a desarrollo de aplicaciones se refiere, disponiendo de cursos, plugins, etc.

El principal handicap es el rendimiento, debido al uso del WebView. Afortunadamente los móviles actuales tienen gran capacidad de procesamiento por lo que la diferencia de rendimiento es perfectamente asumible.

3.7.2.2. Ionic 2 Framework



Ilustración 26. Logotipo Ionic 2 framework.

Fuente: ionic.io.

Ionic 2 es la segunda versión del popular framework para Cordova. Es una extensión de Cordova que utiliza css, HTML5 y Sass. Es un conjunto de librerías en JavaScript que añaden directivas y módulos en una capa por encima de Cordova. Esta versión utiliza Angular 2, una versión mejorada de Angular cuyas principales ventajas son:

- Mayor rendimiento.
- Código modulado por componentes.
- Definición más simple de componentes y servicios.
- Detección de errores en fase de compilación gracias a TypeScript.

En resumen, las ventajas de usar Ionic 2 son:

- Uso de AngularJS 2, uno de los frameworks de front-end con mayor seguimiento y comunidad.
- Variedad de plugins adaptados a Ionic 2, entre ellos el que permite el uso de BLE.
- Creación de forma predefinida de interfaces propias de cada sistema.
- Mejor estructuración del código, lo que conlleva una mejor optimización.
- Uso de Ionic CLI, una interfaz de consola que permite crear fácilmente nuevas páginas o módulos.
- Ionicons: proporciona gran variedad de iconos para Android e IOS, cambiando de forma dinámica según la plataforma.

3.7.3. Conclusiones

Llegado a este punto, se eligió Ionic 2 como herramienta para realizar el proyecto. Esto es debido, principalmente, al conocimiento de la plataforma (tanto Ionic 2 como Cordova), lo que facilitará el desarrollo y garantizará resultados en mucho menos tiempo. Además del uso de interfaces para cada sistema operativo y el uso de AngularJS 2, framework de frontend realmente bueno.

A continuación, se muestra una gráfica que muestra el interés que tiene cada una de las plataformas en los últimos 12 meses. Esta gráfica no muestra noticias, imágenes o vídeos. Simplemente se refiere a las búsquedas de los usuarios en Google. Con esto se puede suponer la cantidad de documentación, guías, preguntas y respuestas que se hacen sobre cada plataforma.

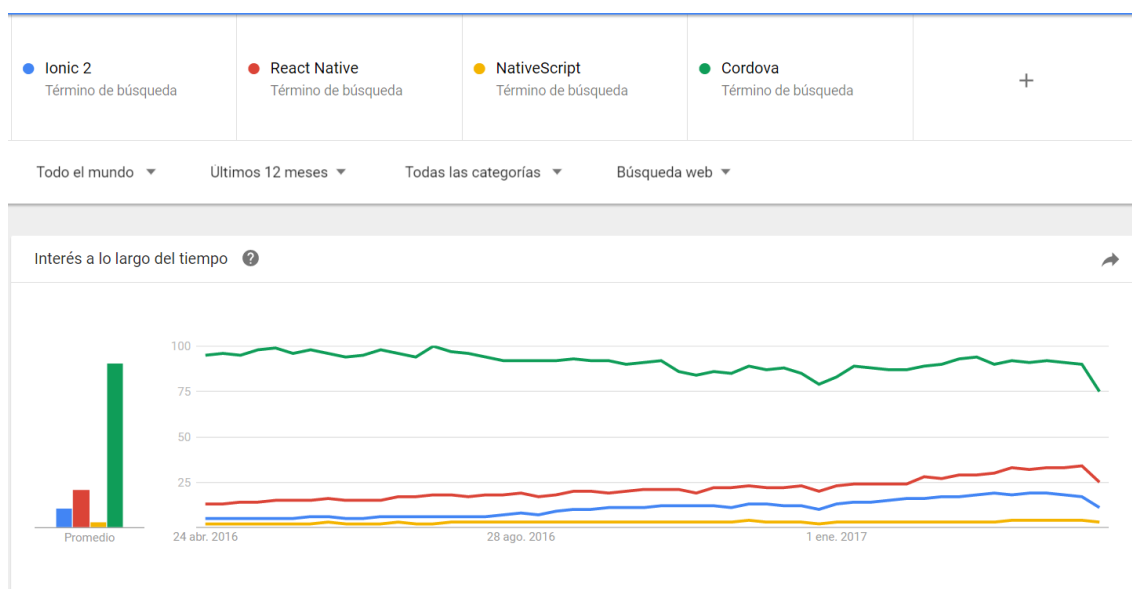


Ilustración 27. Comparativa del interés de los frameworks.

Fuente [Google.es/trends](https://www.google.es/trends),

Se puede apreciar con claridad cómo Cordova supera a los demás, en segundo lugar, se encuentra React Native, que puede ser que tenga tanta repercusión no solo por la tecnología, sino por pertenecer a Facebook. Destacar por último a NativeScript en último lugar. Esto podría ser debido a su bajo nivel de madurez con respecto a las demás tecnologías.

Aunque aparecen como términos separados, Ionic 2 y Cordova están estrechamente relacionados por las razones explicadas en los puntos anteriores, por lo que se toman como una.

4. Metodología

En este punto se describen las herramientas necesarias para el correcto desarrollo del proyecto en el ámbito de la ingeniería del software. La ingeniería del software se define como la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software. El desarrollo de software es un proceso intrínsecamente creativo y la ingeniería del software trata de establecer unas pautas y modelos para conseguir llegar al objetivo deseado en menos tiempo y con menos contratiempos.

El desarrollo de software académico y profesional se beneficia de los métodos de la ingeniería del software, ya que su eficacia está más que demostrada. Por ello esta serie de técnicas se van a utilizar para el desarrollo del proyecto.

La ingeniería del software se compone de diferentes etapas y técnicas. A continuación, se detallarán cada una de las utilizadas en el proyecto.

4.1. Requisitos mínimos

Para desarrollar el proyecto, probarlo y posteriormente utilizar el sistema, son necesarios unos requisitos mínimos que han de cumplirse para que se puedan acceder a todas las funcionalidades.

- Versión de **Android 4.4** o superior. Es a partir de esta versión en la que se integra la tecnología Bluetooth Low Energy (BLE). Sin ella, es imposible conectar la smartband con el smartphone.
- Versión de **iOS 5** o superior. Como ocurre con Android, es en esta versión donde se implementa la tecnología BLE necesaria para la monitorización.
- Smartband **Xiaomi Mi Band 2**. Con esta smartband se ha decidido realizar el proyecto y por ello es la única pulsera con la que funciona.

4.2. Etapas

En este apartado se muestran cada una de las etapas de las que se compone el proyecto. Sin embargo, se analizará en profundidad cada una de ellas en apartados posteriores de la memoria.

- Análisis de requisitos: primera y fundamental etapa en el desarrollo software, Consiste en identificar en qué se está trabajando, plantear necesidades, acotar

funcionalidades y finalmente definir estos requisitos de la manera más clara posible. Un mal análisis puede llevar a pérdidas de tiempo, funcionalidades innecesarias y callejones sin salida en un momento en el que puede ser delicado volver atrás.

- **Diseño y arquitectura:** en esta etapa se definen de forma conceptual los requisitos del proyecto, a través de los casos de uso y los mockups esenciales para tener una idea clara del aspecto visual del proyecto.
- **Desarrollo:** esta fase consiste en plasmar lo establecido en la etapa anterior en forma de código. Tanto lógica como diseño, se realizan de forma conjunta o separada, y el objetivo al final de esta etapa es tener un producto completamente funcional.
- **Pruebas:** consiste en probar el producto realizado en la etapa anterior. Estas pruebas han de ser lo más exhaustivas posibles y si puede ser, contar con la ayuda de una persona ajena al proyecto para que valore la experiencia de usuario.

4.3. Metodologías ágiles

Para el desarrollo del proyecto se ha decidido emplear de metodologías ágiles. A diferencia de otras, como las secuenciales o evolutivas, las ágiles se adaptan perfectamente a este proyecto. Los principales inconvenientes del uso de metodologías pesadas en proyectos de tamaño pequeño o medio son:

- Sobrecarga de trabajo en el proceso de desarrollo.
- Se invierte más tiempo en pensar cómo se va a desarrollar el proyecto que en el proceso de desarrollo.
- Si se cambia un requisito hay que hacer mucho trabajo de rediseño y nueva documentación.

Las principales características de las metodologías ágiles son:

- Mejoran la satisfacción del cliente ya que en cada etapa del desarrollo se le entrega software con valor.
- Se pueden cambiar los requisitos incluso en etapas tardías del desarrollo, sin demasiado coste adicional.

- Aumento de la productividad: al trabajar con iteraciones, se pueden realizar pruebas de carga y rendimiento casi desde un primer momento.

Mejor gestión de riesgo: al cerrar etapas y entregar software funcional constantemente, se puede observar qué requisitos requieren más trabajo o tienen más urgencia y ajustar el ciclo de trabajo cuando sea necesario.

4.4. Herramientas hardware

En este apartado se describen los principales dispositivos hardware necesarios para el desarrollo del proyecto. Se asume, y por lo tanto no se describe, el uso de un pc para desarrollar todas las partes del proyecto.

4.4.1. Xiaomi Mi Band 2

La popular pulsera de la marca china es una de las más económicas del mercado. Se caracteriza por un diseño simple y, desde el punto de vista del desarrollador, documentación escueta, tanto que el fabricante no proporciona SDK ni API alguna con la que desarrollar para ella. Sin embargo, se han logrado avances con el uso de ingeniería inversa, ya que, aunque no se proporciona documentación, la comunicación se basa en el estándar BLE.



Ilustración 28. Xiaomi Mi Band 2.

Fuente: diegocampy.it.

Su principal finalidad es medir distancia recorrida y pasos dados por defecto; y ritmo cardíaco cuando se le pida. Además, dispone de vibración y es capaz de avisar de notificaciones.

Hay que decir que esta smartband no fue la elección inicial. En un primer momento se optó por darle continuidad al proyecto de la compañera Gala utilizando Angel Sensor. De este modo se podía aprovechar parte de sus conocimientos, especialmente los relacionados con el SDK que proporcionaban los creadores. Desgraciadamente el proyecto se abandonó y la pulsera dejó de estar a la venta, con lo que había que cambiar de dispositivo.

Esto requirió un reajuste en los requisitos del proyecto al tener que empezar de cero y sin documentación oficial con la nueva Xiaomi. Del proyecto original sólo se conserva la idea y los consejos que pueden ofrecer tanto Gala como el tutor, José García.

4.4.2. BQ Aquaris M5

Se ha escrito un apartado para este dispositivo ya que al principio del proyecto no se disponía de él. En un principio se contaba con un Samsung Galaxy S3, con la versión de Android 4.1.2. Desde un primer momento se supo que la conexión mediante BLE que usan las pulseras sólo está disponible a partir de la versión 4.4 de Android. Ante esto y la ausencia de actualizaciones al tratarse de un dispositivo antiguo, se optó por adquirir uno más moderno.

Antes de decantarse por uno se investigó a cerca de las versiones mínimas requeridas para los entornos de desarrollo de aplicaciones multiplataforma. Estas rondaban entre la 4.2 y la 4.4.

Finalmente se optó por un BQ Aquaris M5, dispositivo cuya versión de Android (6.0) permite cumplir todos los requisitos del proyecto que dependen de él.



Ilustración 29. BQ Aquaris M5.

Fuente: bq.com.

4.4.3. Raspberry Pi 2 Model B

La segunda generación del ordenador de placa reducida es un producto tanto económico como versátil. Desarrollado por la fundación Raspberry Pi, esta placa es una de las más utilizadas por la comunidad. Puede servir como mini ordenador, servidor web, servidor de correo o centro multimedia, entre otros muchos usos más creativos.

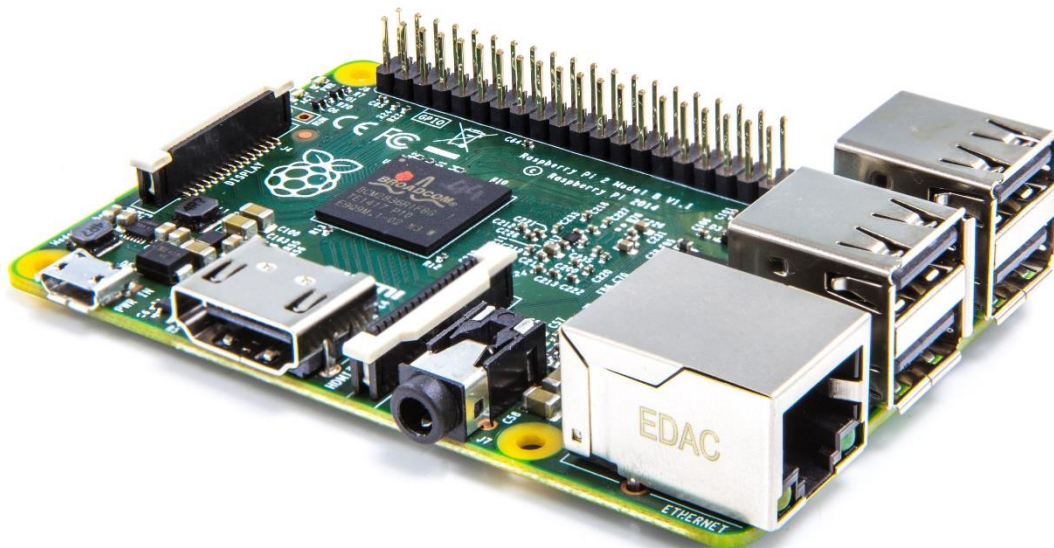


Ilustración 30. Raspberry Pi 2.

Fuente: raspberrypi.org.

A la hora de describir sus especificaciones, dispone de un procesador ARM11, 1 GB de memoria RAM y soporte para almacenamiento con tarjeta microSD, ya que no dispone de memoria física por sí sola. En cuanto a conectividad tiene puerto HDMI 1.4 y un puerto rj45.

Su sistema operativo cuenta con una versión adaptada de Debian, denominada Raspbian. Obviamente soporta otros sistemas operativos, pero este es el que recomienda el fabricante y cuenta con todo lo necesario para el desarrollo del proyecto.

En este proyecto se ha utilizado como servidor web donde se aloja la API rest. Para la puesta a punto ha sido necesario instalar y configurar Apache 2, PostgreSQL y pgAdmin 3, el gestor de base de datos para Debian.

4.5. Herramientas software

A continuación, se muestran desglosadas cada una de las herramientas software necesarias para la consecución del proyecto. Aunque separadas por propósitos, todas son igualmente necesarias a lo largo del proceso de desarrollo.

4.5.1. Documentación

En este apartado se define el software necesario tanto para crear la documentación como para guardarla y distribuirla.

- Microsoft Word 2016: Dentro del paquete Microsoft Office 365 que se ofrece gratuitamente a estudiantes, se utiliza para redactar la memoria y adaptarla a las guías de estilo marcadas por la universidad.
- Dropbox: Servicio de alojamiento compartido para almacenar la documentación. Sirve de seguimiento para con el tutor.

4.5.2. Planificación

Para lograr una correcta planificación del proyecto se necesitan herramientas que puedan crear tareas, asignarles tiempo, prioridades, etc. A la vez que se puedan visualizar fácilmente para poder conocer el estado del proyecto de un golpe de vista. Estas son las que se utilizan a lo largo del proyecto.

- Microsoft Project 2016: Herramienta de planificación, gestión de recursos y medición de tiempo. Tiene muchas utilidades para manejar las horas

establecidas, las horas reales, el porcentaje de trabajo realizado, establecer tareas dependientes de otras, etc.

- Trello: creador de pizarras Kanban para gestión de tareas. Permite crear diferentes secciones en la pizarra y así dividir las taras en, por ejemplo, por hacer, haciendo y hechas.

4.5.3. Diseño

Una vez planificado el proyecto se necesitan una serie de herramientas para realizar los diseños de bajo nivel que posteriormente facilitará la implementación de forma más rápida. Además, es necesario realizar los diagramas de casos de uso y para ello se necesita una herramienta diferente. Por lo tanto, se ha optado por:

- GIMP: programa de edición de imágenes libre y gratuito. Forma parte del proyecto GNU. Es el programa de edición fotográfica disponible en más sistemas operativos. Utilizado para los logos e iconos tanto de la web como de la app.
- Gliffy: sitio web para el desarrollo de todo tipo de diagramas. Este servicio es de pago, pero cuenta con una versión de prueba de 15 días, suficientes para realizar los diagramas y revisarlos.
- Prototyper: aplicación de escritorio destinada a la creación de prototipos de baja fidelidad. Sin embargo, se pueden realizar diseños interactivos que sirven para visualizar cómo es la interacción con el sistema, lo que ayuda a definir mejor las funcionalidades.

4.5.4. Desarrollo

En esta etapa se encuentra el software más complejo, ya que en ella se sitúan los diferentes IDEs utilizados tanto para el frontend como el backend.

- Eclipse: es una plataforma de desarrollo de software compuesto por un conjunto de herramientas de código abierto. Dispone de un editor de texto con analizador sintáctico de compilación en tiempo real. También posee pruebas unitarias con JUnit y la posibilidad de control de versiones con GitHub. Es en este entorno donde se crea la API restfull en PHP.
- DBeaver: gestor de bases de datos relacionales que proporciona un editor de texto y visual. Es una aplicación de escritorio basado en Eclipse. Es software de

código abierto. Se utiliza para la creación y gestión de la base de datos en PostgreSQL.

- Visual Studio Code: editor de código fuente creado por Microsoft. Es gratuito, de código abierto, pero bajo licencia propietaria. Es realmente útil con el uso de Typescript, ya que este IDE es capaz de detectar errores en tiempo real, compilar, debuggear y gestionar todo un proyecto en este lenguaje. Es el editor más completo, quizás sea porque el propio Microsoft es el creador del lenguaje. Visual Studio Code se utiliza en el proyecto tanto para la aplicación web como para la de dispositivo móvil.
- GitHub: plataforma de control de versiones Git. Es gratuita para crear repositorios públicos. En este caso no importa ya que el proyecto es sin ánimo de lucro.

4.5.5. Pruebas

En esta última etapa se necesitan herramientas fiables que garanticen que todo funciona correctamente. Estas pruebas se pueden realizar tanto al terminar etapas de desarrollo como al finalizar el proyecto.

- SoapUI: herramienta desarrollada por SmartBear en java para la realización de pruebas en servicios SOA y REST. Capaz de soportar protocolos SOAP, REST o HTTP, es una forma fácil de comprobar el correcto funcionamiento de la API del proyecto. Se pueden realizar peticiones GET, POST, PUT y DELETE entre otras, y enviar con ellas datos en formato Json o parámetros por query string.

4.6. Ingeniería inversa

La ingeniería inversa es un proceso cuyo objetivo es obtener información o diseño a partir de un producto, software o hardware, con el fin de determinar sus componentes y la manera de interactuar entre sí.

Se llama inversa porque hace el recorrido opuesto a las tareas habituales de la ingeniería. Se trata de obtener la mayor información técnica de un producto, del cual no se dispone de documentación de ningún tipo y mediante el razonamiento abductivo de su estructura, función y operación, lograr entender su funcionamiento de manera profunda.

La ingeniería inversa surgió en la Segunda Guerra Mundial, cuando uno de los dos bandos capturaba dispositivos del otro. Era entonces cuando los ingenieros investigaban a cerca de él llegando a conocer todos los detalles de su funcionamiento, lo que les llevaba a conocer sus puntos débiles y, en consecuencia, obtener ventaja en la batalla.

La siguiente imagen muestra una comparación entre ingeniería directa e inversa.

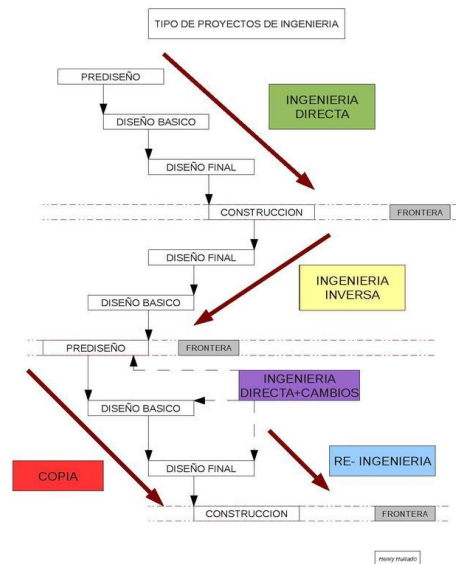


Ilustración 31. Comparación ingeniería inversa y directa.

Fuente: Wikipedia.org.

Como se puede ver se trata de hacer el proceso inverso hasta llegar a la primera etapa, la del prediseño. A partir de ahí surgen dos opciones: realizar una copia o aplicar un proceso de reingeniería. Esto puede suponer problemas legales si el producto al que se le aplica ingeniería inversa es de propiedad privada.

Los principales usos de la ingeniería inversa son:

- Llegar a comprender la tecnología utilizada por otros.
- Analizar productos de la competencia para ver si infringen alguna ley.
- Desarrollar productos compatibles con otro producto, sin tener acceso a detalles técnicos de esto último.
- Conocer las brechas de seguridad que puede tener un producto.

Este es el tercer punto en el que se basa este apartado del proyecto. Ya que la smartband elegida no dispone de documentación oficial para que desarrolladores externos creen

productos compatibles, es necesario un proceso de ingeniería inversa para conocer cómo se recogen datos de esta y cómo se le pide que realice mediciones en tiempo real.

Al ser una de las smartbands más exitosas, numerosos desarrolladores han empezado con este proceso. Existen demostraciones de aplicaciones para Android capaces de pedir el ritmo cardíaco de forma constante. Sin embargo, este código no puede ser aprovechado ya que en este caso se pretende realizar una aplicación multiplataforma cuyo código fuente está en TypeScript.

5. Estudio de la viabilidad

Este es un paso importante en la realización del proyecto ya que en él se establecen una fecha de entrega y unas horas estimadas. Por ello es necesario realizar una planificación exhaustiva del proyecto, estableciendo fechas y requisitos de la app.

5.1. Planificación temporal

En primer lugar, se establece una planificación temporal. Consiste en dividir el tiempo en hitos, fechas donde se pueda mostrar algo tangible del proyecto. Los hitos, explicados a grandes rasgos, son los siguientes:

Hito 0: toma de contacto con el proyecto. En este hito se tratará de documentar el problema, así como realizar un estudio de las smartbands del mercado y de la competencia. En este caso es necesario porque no he trabajado con smartbands ni he realizado proyectos relacionados con la salud.

Hito 1: Se empezará a desarrollar una pequeña demo con la tecnología elegida, con el fin de familiarizarse con la misma. Además, se realizarán los diseños previos tanto de la App como de la base de datos.

Hito 2: Se desarrollará la BBDD, API Rest, requisitos funcionales y front-end. Este hito es la pieza central del proyecto y el que determinará si se cumplen los objetivos o no.

Hito 3: Desarrollo de la web, detalles finales de la app, finalización y revisión de la memoria.

Para visualizar la planificación se ha creado un diagrama de Gantt con la herramienta Project.

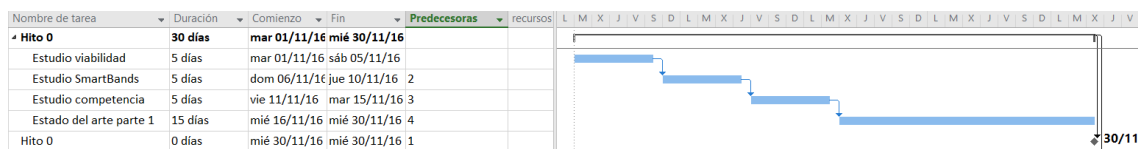


Ilustración 32. Diagrama de Gantt Hito 0.

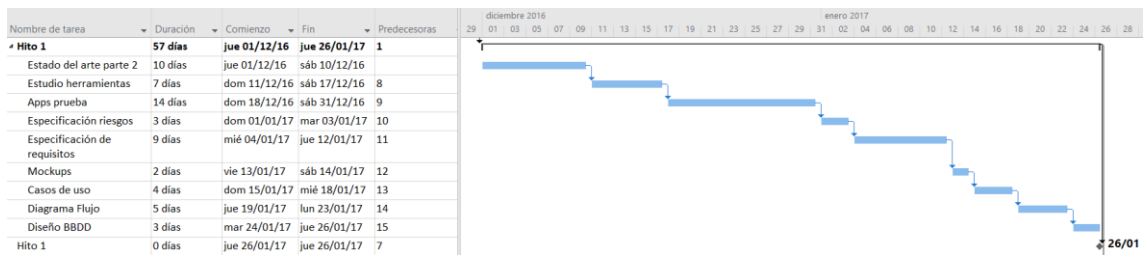


Ilustración 33. Diagrama de Gantt Hito 1.

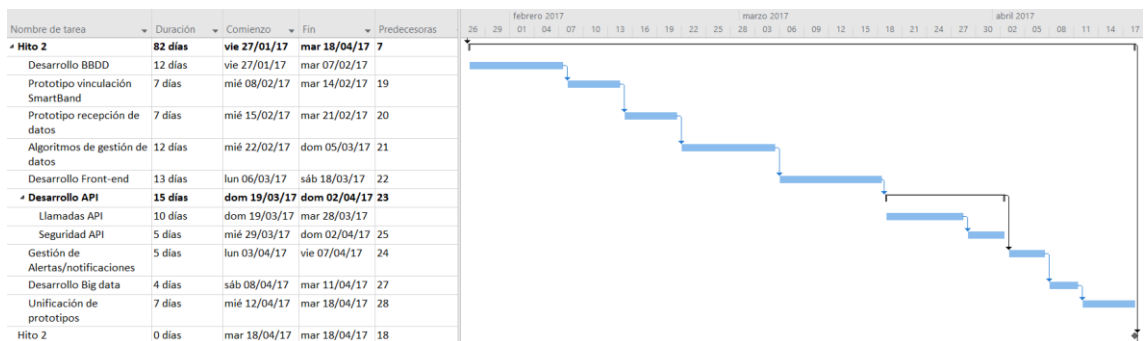


Ilustración 34. Diagrama de Gantt Hito 2.

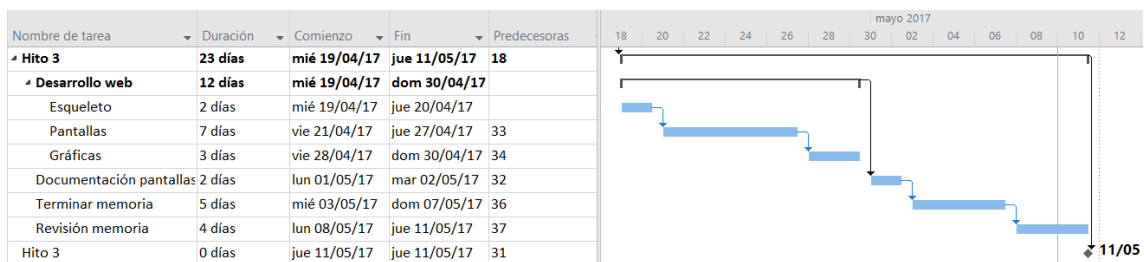


Ilustración 35. Diagrama de Gantt Hito 3.

5.2. Gestión de riesgos

La gestión de riesgos es un paso necesario para identificar y prever posibles contratiempos que surjan a lo largo del proyecto. Además, se establecen pautas para solucionarlos.

Para cada riesgo se detallan los siguientes campos:

- Identificador: código de identificación.
- Nombre: palabra o conjunto de palabras breves y descriptivas.
- Tipo: para diferenciar los riesgos según su origen.
- Probabilidad: posibilidad de que ocurra dicho riesgo.
- Efectos: nivel de importancia de que ocurra el riesgo.
- Estrategia: pautas a seguir para resolver el riesgo.

- Identificadores potenciales: síntomas que evidencien que está sucediendo el riesgo.

Identificador	RI-01
Nombre	Hardware obsoleto
Tipo	Tecnología
Probabilidad	Alta
Efectos	Serio
Estrategia	Adquirir nuevo hardware, ya sea ordenador o teléfono móvil.
Identificadores potenciales	Incompatibilidad entre sistemas.

Tabla 8. Riesgo 1.

Identificador	RI-02
Nombre	Rotura del equipo
Tipo	Tecnología
Probabilidad	Baja
Efectos	Serio
Estrategia	Adquirir un nuevo equipo.
Identificadores potenciales	Ralentización del equipo, sobrecalentamiento.

Tabla 9. Riesgo 2.

Identificador	RI-03
Nombre	Código no eficiente
Tipo	Tecnología
Probabilidad	Alta
Efectos	Medio
Estrategia	Claridad al desarrollar software.
Identificadores potenciales	Bajo rendimiento.

Tabla 10. Riesgo 3.

Identificador	RI-04
Nombre	No disponer de la smartband elegida
Tipo	Tecnología
Probabilidad	Media
Efectos	Serio
Estrategia	Adquirir presupuesto.
Identificadores potenciales	Falta de presupuesto para adquirirla.

Tabla 11. Riesgo 4.

Identificador	RI-05
Nombre	Baja por enfermedad
Tipo	Persona
Probabilidad	Alta
Efectos	Tolerable
Estrategia	Adelantar el máximo de trabajo posible
Identificadores potenciales	Llegado el invierno hay más probabilidad de caer enfermo.

Tabla 12. Riesgo 5.

Identificador	RI-06
Nombre	Conocimientos necesarios
Tipo	Persona
Probabilidad	Media
Efectos	Seria
Estrategia	Documentarse lo máximo posible sobre las tecnologías usadas.
Identificadores potenciales	Incapacidad de empezar a desarrollar el sistema.

Tabla 13. Riesgo 6.

Identificador	RI-07
Nombre	Desmotivación
Tipo	Persona
Probabilidad	Baja
Efectos	Serio
Estrategia	Establecer reuniones con el tutor.
Identificadores potenciales	Incumplimiento de los plazos establecidos en la planificación previa

Tabla 14. Riesgo 7.

Identificador	RI-08
Nombre	Falta de tiempo por prácticas de empresa.
Tipo	Persona
Probabilidad	Alta
Efectos	Serio
Estrategia	Eliminar algo de tiempo libre.
Identificadores potenciales	Incumplimiento de los plazos establecidos en la planificación previa.

Tabla 15. Riesgo 8.

Identificador	RI-09
Nombre	Imposibilidad de realizar reuniones con el tutor
Tipo	Organizacional
Probabilidad	Media
Efectos	Serio
Estrategia	Establecer un horario fijo de reuniones.
Identificadores potenciales	-

Tabla 16. Riesgo 9.

Identificador	RI-10
Nombre	Imposibilidad de solventar los problemas
Tipo	Organizacional
Probabilidad	Media
Efectos	Catastrófico
Estrategia	Documentarse lo suficiente para las tareas
Identificadores potenciales	Estancamiento del proyecto en algún apartado de desarrollo.

Tabla 17. Riesgo 10.

Identificador	RI-11
Nombre	Acumulación de incidencias
Tipo	Organizacional
Probabilidad	Media
Efectos	Serio
Estrategia	Organizar las tareas con un margen de tiempo para imprevistos.
Identificadores potenciales	Estancamiento del proyecto con alguna tarea.

Tabla 18. Riesgo 11.

Identificador	RI-12
Nombre	Mala organización en el reparto de tareas
Tipo	Organizacional
Probabilidad	Media
Efectos	Serio
Estrategia	Organizar las tareas con un margen de tiempo para imprevistos.
Identificadores potenciales	Incumplimiento de plazos en los hitos.

Tabla 19. Riesgo 12.

Identificador	RI-13
Nombre	API Rest incompleta.
Tipo	Herramientas
Probabilidad	Media
Efectos	Serio
Estrategia	Dejar la aplicación preparada para futuras actualizaciones.
Identificadores potenciales	Incumplimiento de plazos asignados a las tareas.

Tabla 20. Riesgo 13.

Identificador	RI-14
Nombre	Perdida de Backups
Tipo	Herramientas
Probabilidad	Baja
Efectos	Catastrófico
Estrategia	Utilizar un repositorio de control de código fuente.
Identificadores potenciales	Pérdida parcial/total del software desarrollado.

Tabla 21. Riesgo 14.

Identificador	RI-15
Nombre	Bugs imprevistos
Tipo	Herramientas
Probabilidad	Alta
Efectos	Tolerable
Estrategia	Realizar pruebas de la aplicación cada vez que se termina un requisito.
Identificadores potenciales	Errores al ejecutar el código.

Tabla 22. Riesgo 15.

Identificador	RI-16
Nombre	Licencias de herramientas
Tipo	Herramientas
Probabilidad	Media
Efectos	Tolerable
Estrategia	Buscar soporte por parte de la universidad o utilizar herramientas gratuitas.
Identificadores potenciales	Imposibilidad de utilizar algún tipo de software.

Tabla 23. Riesgo 16.

Identificador	RI-17
Nombre	Adición de nuevo requerimiento
Tipo	Requerimientos
Probabilidad	Media
Efectos	Serio
Estrategia	Organizar las tareas con un margen de tiempo para imprevistos.
Identificadores potenciales	Cambios de smartband.

Tabla 24. Riesgo 17.

Identificador	RI-18
Nombre	Adición de nuevos campos en la base de datos
Tipo	Requerimientos
Probabilidad	Baja
Efectos	Tolerable
Estrategia	Organizar las tareas con un margen de tiempo para imprevistos.
Identificadores potenciales	Necesidad de algún nuevo campo.

Tabla 25. Riesgo 18.

Identificador	RI-19
Nombre	Adición de alguna funcionalidad secundaria
Tipo	Requerimientos
Probabilidad	Medio
Efectos	Serio
Estrategia	Organizar las tareas con un margen de tiempo para imprevistos.
Identificadores potenciales	-

Tabla 26. Riesgo 19.

Identificador	RI-20
Nombre	Cambios en el Front-end
Tipo	Requerimientos
Probabilidad	Baja
Efectos	Serio
Estrategia	Organizar las tareas con un margen de tiempo para imprevistos.
Identificadores potenciales	Escaso tiempo empleado para el desarrollo de mockups.

Tabla 27. Riesgo 20.

Identificador	RI-21
Nombre	Infraestimar el tiempo de desarrollo
Tipo	Estimación
Probabilidad	Media
Efectos	Serio
Estrategia	Organizar las tareas con un margen de tiempo para imprevistos.
Identificadores potenciales	Incumplimiento de plazos asignados a las tareas.

Tabla 28. Riesgo 21.

Identificador	RI-22
Nombre	Infraestimar el tiempo de documentación
Tipo	Estimación
Probabilidad	Media
Efectos	Serio
Estrategia	Organizar las tareas con un margen de tiempo para imprevistos
Identificadores potenciales	Incumplimiento de plazos asignados a las tareas.

Tabla 29. Riesgo 22.

Identificador	RI-23
Nombre	Infraestimar el tiempo de aprendizaje
Tipo	Estimación
Probabilidad	Media
Efectos	Serio
Estrategia	Organizar las tareas con un margen de tiempo para imprevistos.
Identificadores potenciales	Incumplimiento de plazos asignados a las tareas.

Tabla 30. Riesgo 23.

Identificador	RI-24
Nombre	No finalizar algún elemento del proyecto
Tipo	Estimación
Probabilidad	Media
Efectos	Catastrófico
Estrategia	Seguir rigurosamente las fechas y observar los riesgos.
Identificadores potenciales	Incumplimiento de plazos asignados a las tareas.

Tabla 31. Riesgo 24.

6. Especificación de requisitos

En esta temprana etapa del proyecto se pretende definir claramente los objetivos que debe cumplir el sistema. A la hora de definir requerimientos y funcionalidades hay que tener muchos factores en cuenta, como pueden ser:

- Estimación de horas: el TFG de ingeniería multimedia tiene un valor de 12 créditos, lo que se traduce en 300 horas de trabajo.
- Grado de dificultad: plantear objetivos difícilmente alcanzables puede suponer pérdida de tiempo y un estancamiento en el desarrollo.
- Conocimiento de las tecnologías: antes de plantear los objetivos se ha de reflexionar acerca del nivel de conocimiento de las tecnologías a usar, ya que las curvas de aprendizaje de estas pueden requerir tiempo que inicialmente se suponía para otros requerimientos.

Por lo tanto, en este punto se acotan en primer lugar los objetivos del sistema. A partir de ahí se realizarán los casos de uso, que ayudarán a descubrir nuevos requisitos. Por último, se especificará más en los requisitos funcionales y no funcionales del sistema que surgen a raíz de los casos de uso.

6.1. Objetivos del sistema

Esta es una lista con los objetivos esenciales que debe cumplir el proyecto. Se trata de una sucesión de propósitos genéricos que resumirían la funcionalidad completa del sistema.

Para cada objetivo se muestra la siguiente información:

- Identificador: código de identificación.
- Nombre: palabra o conjunto de palabras breves y descriptivas.
- Descripción: definición del objetivo.
- Prioridad: grado de importancia del objetivo.

Identificador	OBJ-01
Nombre	Gestión de usuarios
Descripción	El sistema será capaz de realizar funcionalidades de registro, login monitorización y consulta de datos.
Prioridad	Alta

Tabla 32. Objetivo del sistema 1.

Identificador	OBJ-02
Nombre	Registro en base de datos
Descripción	Los datos obtenidos serán guardados en una base de datos externa.
Prioridad	Alta

Tabla 33. Objetivo del sistema 2.

Identificador	OBJ-03
Nombre	Aplicación multiplataforma
Descripción	Se creará una aplicación móvil para los sistemas Android e IOS.
Prioridad	Alta

Tabla 34. Objetivo del sistema 3.

Identificador	OBJ-04
Nombre	Web de seguimiento
Descripción	Se creará una web donde tanto usuarios como personal sanitario accedan para consultar datos.
Prioridad	Alta

Tabla 35. Objetivo del sistema 4.

Identificador	OBJ-05
Nombre	Alertas
Descripción	La aplicación será capaz de alertar ante emergencias.
Prioridad	Alta.

Tabla 36. Objetivo del sistema 5.

6.2. Casos de uso

Los diagramas de casos de uso describen la interacción entre un agente externo y el sistema. Permiten definir los límites del sistema y las relaciones entre este y el entorno. Sirven para expresar el comportamiento deseado del sistema, a la vez que son un medio de comprensión del mismo que establece las necesidades según la categoría de los usuarios.

Para realizar los casos de uso primero hay que identificar los agentes externos. En este sistema serán Usuario, personal médico y paciente. Para cada uno es necesario realizar

los casos de uso donde se plasme la interacción que se desea que tengan, pero sin explicar cómo se realiza.

6.2.1. Usuario

Este actor es la persona que necesita una solución para sí mismo o para un familiar. Su entorno es la aplicación web donde podrá registrarse y consultar datos. Este usuario tendrá un paciente usuario, que dentro del sistema será otro actor diferente.

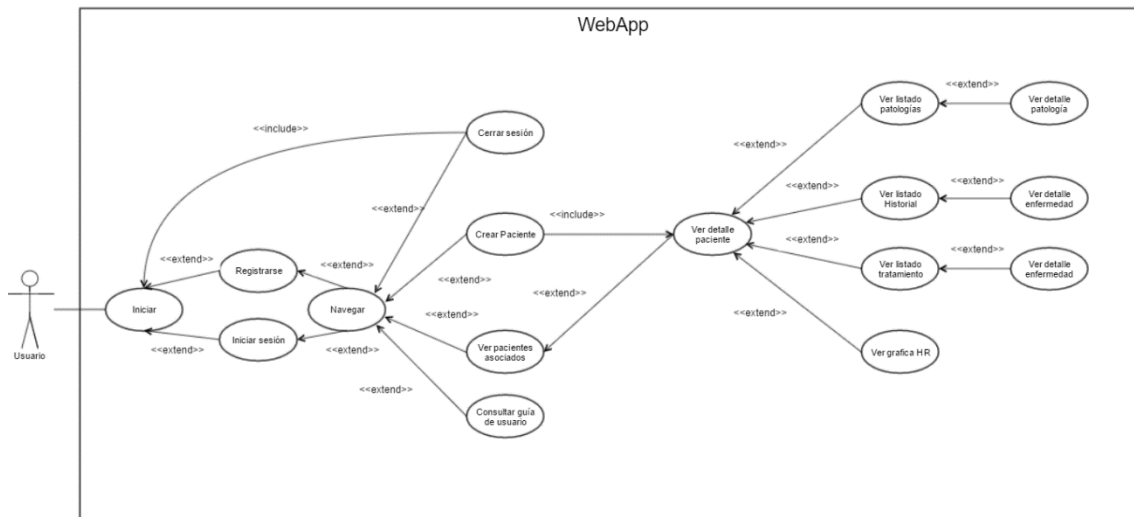


Ilustración 36. Caso de uso usuario.

Para detallar más estos diagramas se realizan una serie de tablas explicativas para iguales para cada actor, donde los campos son:

- Identificador: código de identificación.
- Nombre: palabra o conjunto de palabras breves y descriptivas.
- Pre-condición: premisas que deben darse para que suceda.
- Flujo básico: pasos que habría que seguir para un uso ideal.
- Flujo alternativo: pasos opcionales o que se desencadenan de un error.

Post-condición: la acción que sucede cuando se completa el flujo básico.

Identificador	CU-01
Nombre	Iniciar.
Pre-condición	El usuario accede al dominio.
Flujo básico	La web se carga.
Flujo alternativo	-
Post-condición	Al abrirse, se muestra la pantalla de inicio o login.

Tabla 37. CU 1 usuario.

Identificador	CU-02
Nombre	Registro.
Pre-condición	El usuario no dispone de perfil en el sistema
Flujo básico	<ol style="list-style-type: none"> 1. Introducción de datos. 2. Registro en la base de datos del sistema. 3. Inicio de sesión.
Flujo alternativo	<ol style="list-style-type: none"> 1. Introducción de datos. 2. El email es incorrecto o ya existe en el sistema. 3. Se vuelve a facilitar el formulario.
Post-condición	Se accede a la página de inicio.

Tabla 38. CU 2 usuario.

Identificador	CU-03
Nombre	Inicio sesión.
Pre-condición	El usuario accede al dominio.
Flujo básico	<ol style="list-style-type: none"> 1. Introducción de datos. 2. Inicio de sesión.
Flujo alternativo	<ol style="list-style-type: none"> 1. Introducción de datos. 2. Email o contraseña incorrectos. 3. Se vuelve a facilitar el formulario.
Post-condición	Se accede a la página de inicio.

Tabla 39. CU 3 usuario.

Identificador	CU-04
Nombre	Navegar.
Pre-condición	Hay una sesión iniciada.
Flujo básico	Se muestra el menú y las distintas opciones.
Flujo alternativo	-
Post-condición	-

Tabla 40. CU 4 usuario.

Identificador	CU-05
Nombre	Cerrar sesión.
Pre-condición	Hay una sesión iniciada.
Flujo básico	Se selecciona dicha opción en el menú.
Flujo alternativo	-
Post-condición	Redirección a la página inicial de registro o login.

Tabla 41. CU 5 usuario.

Identificador	CU-06
Nombre	Crear paciente
Pre-condición	Hay una sesión iniciada.
Flujo básico	<ol style="list-style-type: none"> 1. Se selecciona dicha opción en el menú. 2. Introducción de datos del paciente. 3. Registro en base de datos.
Flujo alternativo	<ol style="list-style-type: none"> 1. Se selecciona dicha opción en el menú. 2. Introducción de datos errónea o ya existente. 3. Se vuelve a facilitar el formulario.
Post-condición	Redirección a la página principal.

Tabla 42. CU 6 usuario.

Identificador	CU-07
Nombre	Ver pacientes
Pre-condición	Hay una sesión iniciada.
Flujo básico	<ol style="list-style-type: none"> 1. Se selecciona dicha opción en el menú. 2. Recuperación de información. 3. Creación de una tabla con dicha información.
Flujo alternativo	-
Post-condición	-

Tabla 43. CU 7 usuario.

Identificador	CU-08
Nombre	Guía de usuario
Pre-condición	Hay una sesión iniciada.
Flujo básico	<ol style="list-style-type: none"> 1. Se selecciona dicha opción en el menú. 2. Se muestra una guía con los pasos a seguir.
Flujo alternativo	-
Post-condición	Redirección a las páginas que proporciona la guía.

Tabla 44. CU 8 usuario.

Identificador	CU-09
Nombre	Ver detalle de paciente
Pre-condición	El usuario tiene al menos un paciente asociado.
Flujo básico	<ol style="list-style-type: none"> 1. Visualización de la lista de pacientes. 2. Selección del paciente. 3. Redirección a la página de detalle de paciente.
Flujo alternativo	<ol style="list-style-type: none"> 1. Creación de paciente. 2. Redirección a la página de detalle de paciente.
Post-condición	-

Tabla 45. CU 9 usuario.

Identificador	CU-10
Nombre	Ver listado patologías.
Pre-condición	La página actual es detalle de paciente.
Flujo básico	<ol style="list-style-type: none"> 1. Selección del botón apropiado. 2. Recuperación de información 3. Creación de una tabla con dicha información.
Flujo alternativo	-
Post-condición	-

Tabla 46. CU 10 usuario.

Identificador	CU-11
Nombre	Ver detalle patología.
Pre-condición	Selección de listado de patologías.
Flujo básico	<ol style="list-style-type: none"> 1. Selección de una patología. 2. Se muestra en detalle dicha patología.
Flujo alternativo	-
Post-condición	-

Tabla 47. CU 11 usuario.

Identificador	CU-12
Nombre	Ver listado historial
Pre-condición	La página actual es detalle de paciente.
Flujo básico	<ol style="list-style-type: none"> 1. Selección del botón apropiado. 2. Recuperación de información. 3. Creación de una tabla con dicha información.
Flujo alternativo	-
Post-condición	-

Tabla 48. CU 12 usuario.

Identificador	CU-13
Nombre	Ver detalle historial.
Pre-condición	<ol style="list-style-type: none"> 1. Selección de un episodio en el listado de historial. 2. Se muestra en detalle dicho episodio.
Flujo básico	Selección de un episodio.
Flujo alterno	-
Post-condición	-

Tabla 49. CU 13 usuario.

Identificador	CU-14
Nombre	Ver listado tratamiento.
Pre-condición	La página actual es detalle de paciente.
Flujo básico	<ol style="list-style-type: none"> 1. Selección del botón apropiado. 2. Recuperación de información. 3. Creación de una tabla con dicha información
Flujo alterno	-
Post-condición	-

Tabla 50. CU 14 usuario.

Identificador	CU-15
Nombre	Ver detalle tratamiento.
Pre-condición	Selección de listado de tratamiento.
Flujo básico	<ol style="list-style-type: none"> 1. Selección de un tratamiento. 2. Se muestra en detalle dicho tratamiento.
Flujo alterno	-
Post-condición	-

Tabla 51. CU 15 usuario.

Identificador	CU-16
Nombre	Ver gráfica HR.
Pre-condición	La página actual es detalle de paciente.
Flujo básico	<ol style="list-style-type: none"> 1. Selección del botón apropiado. 2. Recuperación de información. 3. Creación de una gráfica con dicha información.
Flujo alterno	-
Post-condición	Posibilidad de filtrar la gráfica con un formulario.

Tabla 52. CU 16 usuario.

6.2.2. Personal médico

El segundo actor del sistema es el encargado de diagnosticar y valorar los resultados obtenidos en la monitorización. Como el actor anterior, su entorno es la aplicación web. En ella además de realizar tareas de visualización como el primer actor, tendrá la capacidad de editar ciertos campos de sus pacientes.

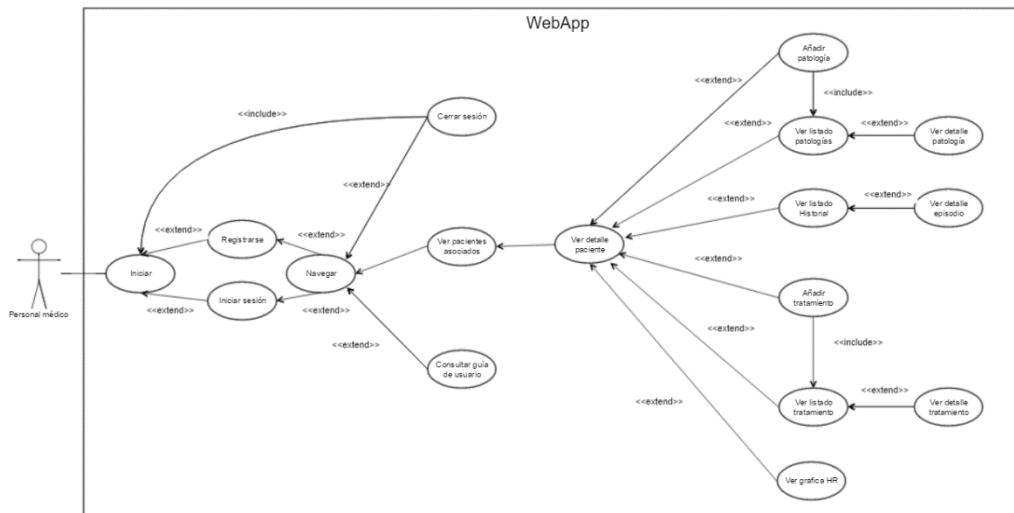


Ilustración 37. caso de uso médico.

Como se puede apreciar en la imagen, muchos de los casos de uso que posee el actor anterior también están disponibles para el personal médico. Esto es lógico ya que gran parte de estos casos de uso son de visualización de diversos aspectos de los pacientes. A continuación, se muestra la lista con código y nombre de los casos de uso que comparten los actores usuario y personal médico:

- CU-01: Iniciar.
- CU-02: Registro.
- CU-03: Inicio sesión.
- CU-04: Navegar.
- CU-05: Cerrar sesión.
- CU-07: Ver pacientes.
- CU-08: Guía de usuario.
- CU-09: Ver detalle de paciente.
- CU-10: Ver listado patologías.
- CU-11: Ver detalle patología.
- CU-12: Ver listado historial.
- CU-13: Ver detalle historial.
- CU-14: Ver listado tratamiento.
- CU-15: Ver detalle tratamiento.
- CU-16: Ver gráfica HR.

El resto de casos de uso se explican en tablas del mismo modo que el actor anterior.

Identificador	CU-17
Nombre	Añadir patología.
Pre-condición	Página detalle de paciente.
Flujo básico	<ol style="list-style-type: none"> 1. Selección del botón apropiado. 2. Introducción de datos. 3. Registro en la base de datos.
Flujo alterno	-
Post-condición	Redirección a página de detalle de paciente.

Tabla 53. CU 17 médico.

Identificador	CU-18
Nombre	Añadir tratamiento.
Pre-condición	Página detalle de paciente.
Flujo básico	<ol style="list-style-type: none"> 1. Selección del botón apropiado. 2. Introducción de datos. 3. Registro en la base de datos.
Flujo alterno	-
Post-condición	Redirección a página de detalle de paciente.

Tabla 54. CU 18 médico.

6.2.3. Paciente

El último actor del sistema. Su entorno es la aplicación móvil, donde podrá realizar unas acciones limitadas. Esto se debe a que el uso de la aplicación móvil es monitorizar al paciente y enviar los datos a través de una API, por lo que hay más funcionalidades en segundo plano y ajenas al actor que interacción de este.

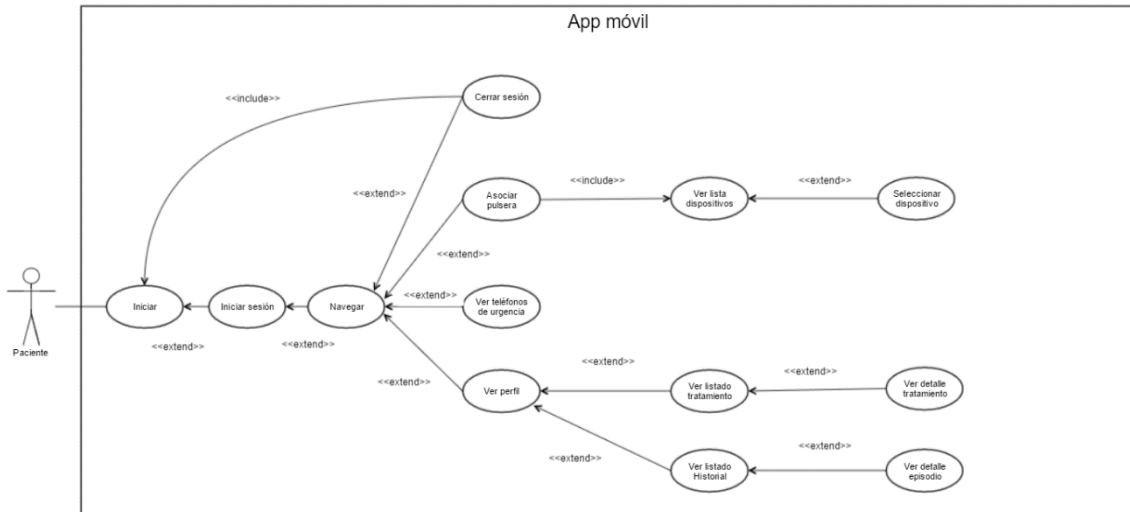


Ilustración 38. caso de uso paciente.

Como ocurre con el resto de actores, hay casos de uso que se comparten. En este caso son menos ya que se trata de un actor completamente diferente. A continuación, se mencionan los casos de uso compartidos, seguidos de los exclusivos de este tercer actor.

- CU-01: Iniciar.
- CU-03: Inicio sesión.
- CU-04: Navegar.
- CU-05: Cerrar sesión.
- CU-12: Ver listado historial.
- CU-13: Ver detalle historial.
- CU-14: Ver listado tratamiento.
- CU-15: Ver detalle tratamiento.

Identificador	CU-19
Nombre	Asociar pulsera.
Pre-condición	Página detalle de paciente.
Flujo básico	1. Selección del botón apropiado. 2. Escaneo de dispositivos.
Flujo alternativo	-
Post-condición	Se muestran los dispositivos encontrados.

Tabla 55. CU 19 paciente.

Identificador	CU-20
Nombre	Ver lista de dispositivos.
Pre-condición	Asociar pulsera.
Flujo básico	Se muestran los dispositivos encontrados.
Flujo alternativo	-
Post-condición	Se podrá seleccionar o volver atrás.

Tabla 56. CU 20 paciente.

Identificador	CU-21
Nombre	Seleccionar dispositivo.
Pre-condición	Ver lista de dispositivos.
Flujo básico	<ol style="list-style-type: none"> 1. Selección de un dispositivo. 2. Vinculación mediante BLE.
Flujo alternativo	-
Post-condición	Mensaje informativo si ha ido bien.

Tabla 57. CU 21 paciente.

Identificador	CU-22
Nombre	Ver perfil.
Pre-condición	Hay una sesión iniciada.
Flujo básico	<ol style="list-style-type: none"> 1. Selección del botón apropiado. 2. Se muestra el perfil del paciente.
Flujo alternativo	-
Post-condición	-

Tabla 58. CU 22 paciente.

Identificador	CU-23
Nombre	Ver teléfonos de urgencia
Pre-condición	Hay una sesión iniciada.
Flujo básico	<ol style="list-style-type: none"> 1. Selección del botón apropiado. 2. Recuperación de datos. 3. Creación de una tabla con dicha información.
Flujo alternativo	-
Post-condición	-

Tabla 59. CU 23 paciente.

6.3. Requisitos funcionales

Los requisitos funcionales definen funciones del sistema que tienen un conjunto de entradas, comportamientos y salidas. Unos requisitos pobres o mal formulados dan como resultado un producto deficiente o de poca calidad.

El éxito o fallo de un sistema depende de cómo de bien se hayan capturado, entendido y usado los requisitos, ya que un error en los requisitos supone un coste mayor que, por ejemplo, un error de código.

Para cada requisito se muestra la siguiente información:

- Identificador: código de identificación.
- Nombre: palabra o conjunto de palabras breves y descriptivas.
- Descripción: definición del objetivo.
- Dependencias: código de los requisitos funcionales que se necesitan completos para la realización del descrito.
- Prioridad: grado de importancia del objetivo.

Identificador	RF-01
Nombre	Registrar Usuario
Descripción	El sistema será capaz de registrar a los distintos tipos de usuario
Dependencia	-
Prioridad	Alta

Tabla 60. Requisito funcional 1.

Identificador	RF-02
Nombre	Iniciar sesión
Descripción	El sistema será capaz de iniciar sesión en cada plataforma para el usuario que está enfocado en cada caso.
Dependencia	RF-01
Prioridad	Alta

Tabla 61 Requisito funcional 2.

Identificador	RF-03
Nombre	Modificar perfil
Descripción	El sistema será capaz de modificar usuarios.
Dependencia	RF-02
Prioridad	Media

Tabla 62. Requisito funcional 3.

Identificador	RF-04
Nombre	Cerrar sesión
Descripción	El sistema podrá cerrar sesión, siendo de este modo inaccesibles los datos.
Dependencia	RF-02
Prioridad	Media

Tabla 63. Requisito funcional 4.

Identificador	RF-05
Nombre	Relacionar usuarios
Descripción	El sistema establecerá una relación de usuario/médico y paciente.
Dependencia	RF-01
Prioridad	Alta

Tabla 64. Requisito funcional 5.

Identificador	RF-06
Nombre	Obtener ritmo cardíaco cada x tiempo
Descripción	La aplicación móvil será capaz de pedir y recibir datos del ritmo cardíaco de la pulsera.
Dependencia	-
Prioridad	Alta

Tabla 65. Requisito funcional 6.

Identificador	RF-07
Nombre	Mostrar gráficas del ritmo cardíaco
Descripción	La aplicación web será capaz de mostrar gráficas sobre el ritmo cardíaco de un paciente, y establecer diversos filtros.
Dependencia	RF-06
Prioridad	Alta

Tabla 66. Requisito funcional 7.

Identificador	RF-08
Nombre	Calcular baremo de ataque
Descripción	La aplicación móvil será capaz de establecer un rango de seguridad sobre el ritmo cardíaco.
Dependencia	-
Prioridad	Alta

Tabla 67. Requisito funcional 8.

Identificador	RF-09
Nombre	Detectar ataque epiléptico
Descripción	La aplicación móvil será capaz de determinar cuándo hay un posible ataque epiléptico.
Dependencia	RF-08
Prioridad	Alta

Tabla 68. Requisito funcional 9.

Identificador	RF-10
Nombre	Alertar de ataque epiléptico
Descripción	La aplicación móvil será capaz avisar vía SMS al teléfono que previamente se le indique.
Dependencia	RF-09
Prioridad	Alta

Tabla 69. Requisito funcional 10.

Identificador	RF-11
Nombre	Confirmar falso positivo
Descripción	La aplicación móvil será capaz de, si el usuario lo confirma, avisar de falso positivo.
Dependencia	RF-09
Prioridad	Media

Tabla 70. Requisito funcional 11.

Identificador	RF-12
Nombre	Crear datos de paciente
Descripción	La aplicación web será capaz de crear historial, patologías y medicación para los pacientes.
Dependencia	-
Prioridad	Media

Tabla 71. Requisito funcional 12.

Identificador	RF-13
Nombre	Consultar datos de paciente
Descripción	La aplicación web será capaz de mostrar el historial, patologías y medicación de los pacientes.
Dependencia	-
Prioridad	Media

Tabla 72. Requisito funcional 13.

Identificador	RF-14
Nombre	Avisar de toma de medicación
Descripción	La aplicación móvil será capaz de avisar al paciente cuando tenga que tomar la medicación.
Dependencia	-
Prioridad	Media

Tabla 73. Requisito funcional 14.

Identificador	RF-15
Nombre	Alertar de caídas.
Descripción	La aplicación será capaz de alertar sobre caídas al número previamente indicado.
Dependencia	-
Prioridad	Media

Tabla 74. Requisito funcional 15.

Identificador	RF-16
Nombre	Vincular smartband.
Descripción	La aplicación móvil será capaz de vincularse con una smartband de manera permanente.
Dependencia	-
Prioridad	Media

Tabla 75. Requisito funcional 16.

6.4. Requisitos no funcionales

Los requisitos no funcionales son requisitos no relacionados directamente con la funcionalidad del sistema. Pueden establecer criterios de calidad en función de la fiabilidad, escalabilidad, tiempo de respuesta, etc.

También pueden ser utilizados para describir restricciones que el sistema no puede sobrepasar, o restricciones externas al sistema como el uso de APIs de terceros.

Estos requisitos normalmente se aplican al sistema en su totalidad, ya que describen aspectos técnicos globales que debe incluir el sistema. Son por lo tanto requisitos más críticos que los funcionales y suelen ser más difíciles de verificar. Para cada requisito se muestra la siguiente información:

- Identificador: código de identificación.
- Nombre: palabra o conjunto de palabras breves y descriptivas.
- Descripción: definición del objetivo.
- Prioridad: grado de importancia del objetivo.

Identificador	RNF-01
Nombre	Aplicación móvil multiplataforma
Descripción	La aplicación móvil será desarrollada para Android e IOS
Prioridad	Alta

Tabla 76. Requisito no funcional 1.

Identificador	RNF-02
Nombre	Seguridad
Descripción	El sistema garantizará que cada usuario acceda a la información que le corresponde.
Prioridad	Alta

Tabla 77. Requisito no funcional 2.

Identificador	RNF-03
Nombre	Diseño amigable
Descripción	La aplicación móvil se adaptará a los estándares de cada plataforma, para mejorar así la experiencia de usuario.
Prioridad	Alta

Tabla 78. Requisito no funcional 3.

Identificador	RNF-04
Nombre	Fachada
Descripción	En la aplicación móvil se creará una fachada para que sea utilizable por diferentes smartbands.
Prioridad	Alta

Tabla 79. Requisito no funcional 4.

Identificador	RNF-05
Nombre	Optimización
Descripción	La aplicación móvil llamará constantemente a la smartband. Es aconsejable optimizar estas llamadas para mejorar el consumo.
Prioridad	Media

Tabla 80. Requisito no funcional 5.

Identificador	RNF-06
Nombre	Reducir el número de falsos positivos.
Descripción	La aplicación móvil usará un algoritmo para detectar ataques. Ajustar este algoritmo para reducir los falsos positivos.
Prioridad	Media

Tabla 81. Requisito no funcional 6.

7. Diseño y arquitectura

Esta es la última etapa previa al desarrollo. En ella, se realizan los prototipos de la aplicación para poder ver de manera fácil cómo va a ser el sistema. Además, se realiza el esquema entidad-relación de la base de datos, necesario para que el sistema tenga la persistencia de datos que necesita.

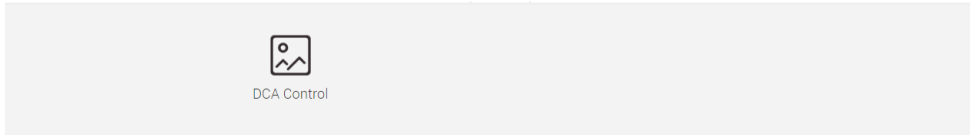
7.1. Prototipos de la aplicación

Realizar prototipos es una parte importante del proyecto, ya que gracias a ella se puede tener una idea de lo que será el producto final. Además, al realizar estos prototipos se pueden descubrir nuevas funcionalidades o replantear requisitos. También sirve para averiguar si el sistema es algo pobre y es necesario añadir nuevos aspectos que complementen el sistema y lo hagan más llamativo.

A continuación, se muestran los prototipos realizados tanto para la página web como para la aplicación móvil. Destacar que los elementos no tienen por qué estar exactamente en el mismo lugar, ya que lo importante es determinar qué aparece en cada pantalla y cómo interactúa con el usuario.

The image shows a web login prototype. At the top, there is a header bar with a logo icon and the text "DCA Control". Below the header is a central login form. The form contains two input fields labeled "Email" and "Password". Below these fields is a checkbox labeled "Personal médico". At the bottom of the form, there is a blue "Sign in" button followed by the text "o regístrate". At the bottom of the page, there is a footer bar with the text "2016 copyright Sergio Castro".

Ilustración 39. Prototipo: login web.



Email

Nombre

Password

repetir password

Teléfono

Personal médico

[Registro](#)

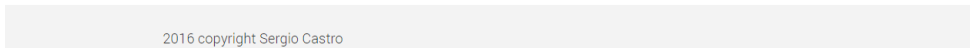


Ilustración 40. Prototipo: registro.



Ilustración 41. Prototipo: inicio.

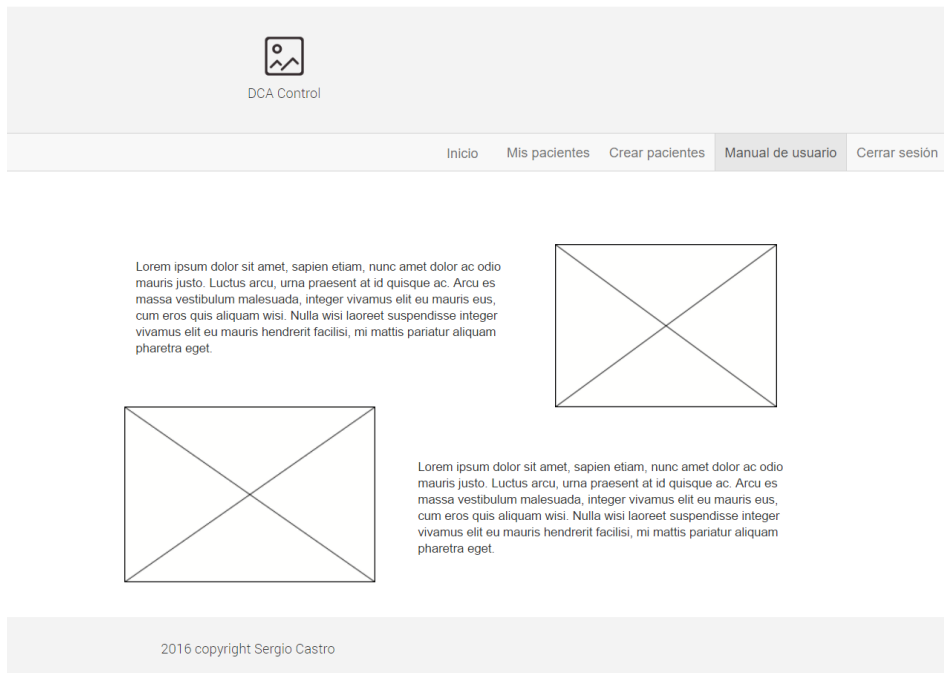


Ilustración 42. Prototipo: manual de usuario.

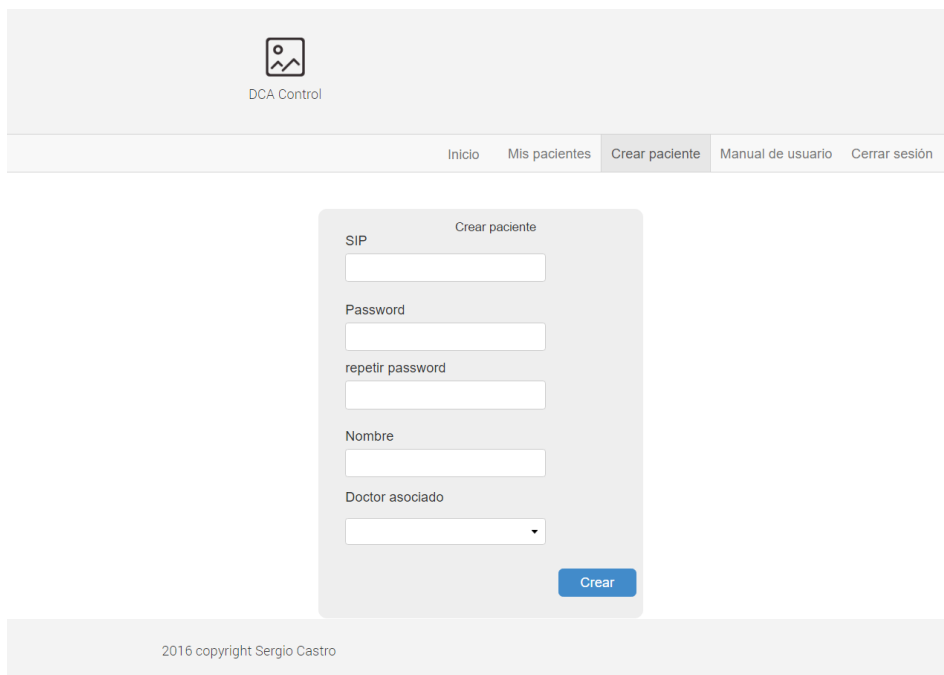
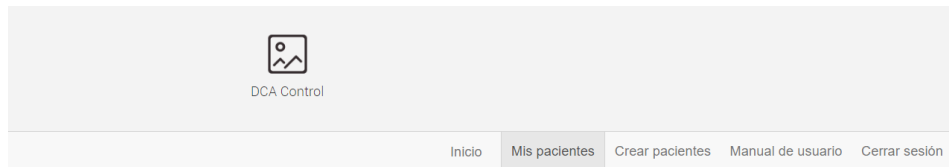


Ilustración 43. Prototipo: crear paciente.



SIP	Nombre	Acciones
4862486345	Paciente prueba	Detalles, editar
5426856587	Paciente prueba dos	Detalles, editar

2016 copyright Sergio Castro

Ilustración 44. Prototipo: mis pacientes.

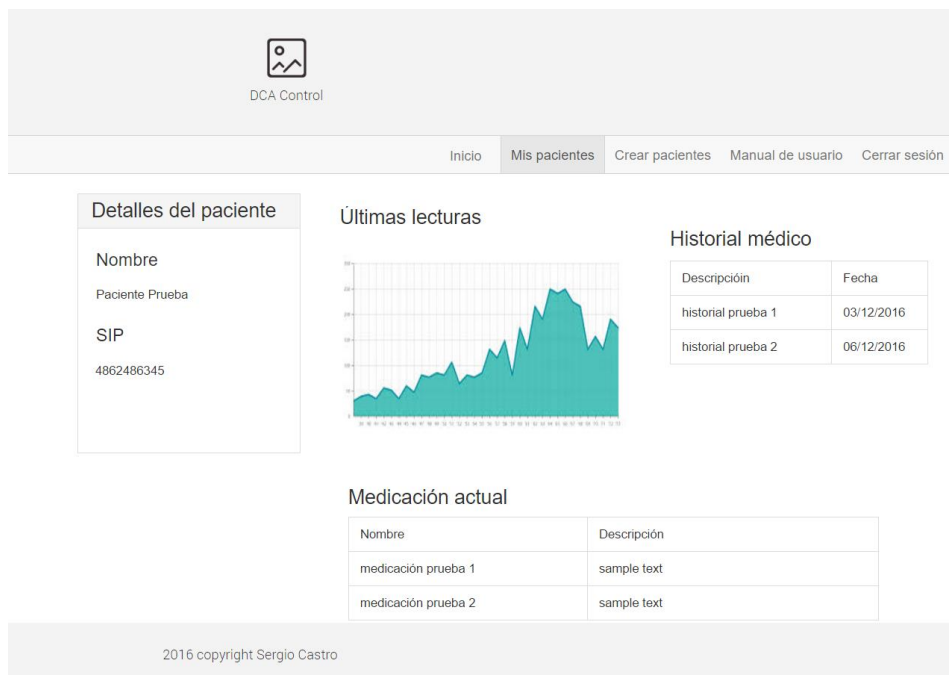


Ilustración 45. Prototipo: detalle de paciente.

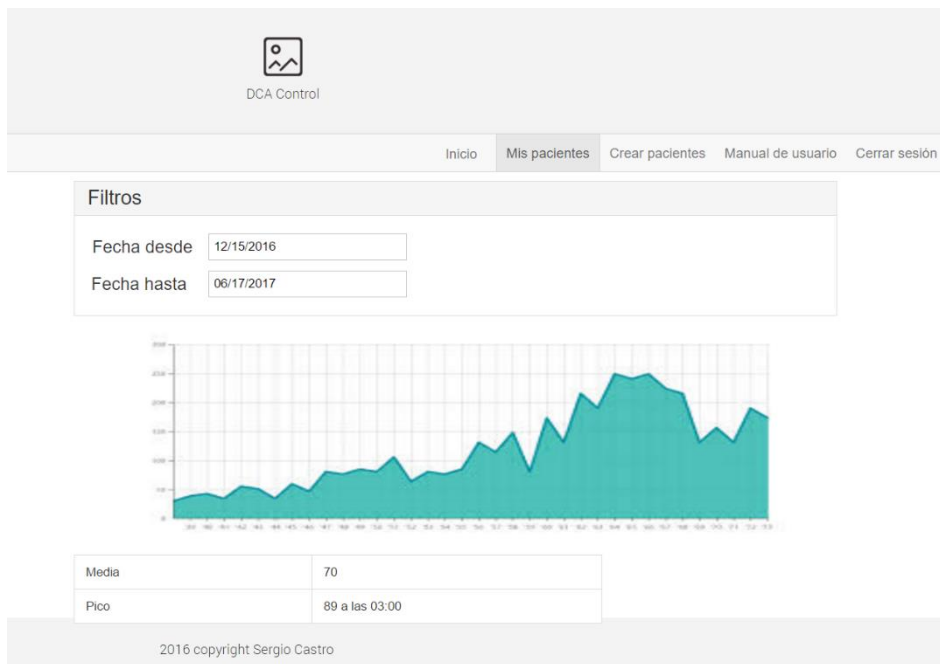


Ilustración 46. Prototipo: gráfica hr.

The screenshot shows the 'DCA Control' web application with a 'Crear tratamiento' modal dialog open. The dialog has a title 'Crear tratamiento' and a close button (X). The main text inside the dialog says 'Crea un tratamiento para Paciente Prueba'. There are two input fields: 'Nombre' and 'Descripción'. At the bottom of the dialog, there are two buttons: 'Crear' (blue) and 'Cancelar' (white). The background shows a patient details page with the following information: 'Nombre: Paciente Prueba', 'SIP: 4862486345'. There is also a table with two rows: 'medicación prueba 1' and 'medicación prueba 2', each with a 'sample text' column. The footer of the application reads '2016 copyright Sergio Castro'.

Ilustración 47. Prototipo: crear tratamiento.

Como se puede observar, la aplicación web cuenta con todo lo necesario para crear y gestionar pacientes. Por otro lado, y como principal funcionalidad, se encuentra la posibilidad de mostrar una gráfica detallada para la evaluación del personal adecuado.

A continuación, se muestran los prototipos de la aplicación móvil.

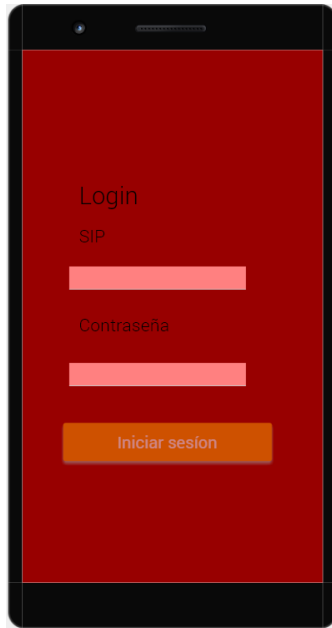


Ilustración 48. Prototipo: login app.

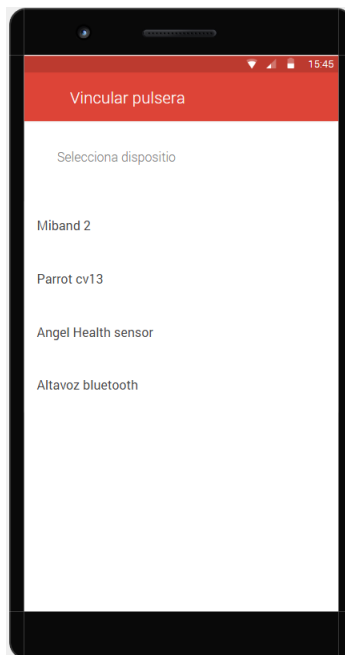


Ilustración 49. Prototipo: selección de dispositivo.



Ilustración 50. Prototipo: monitorización.

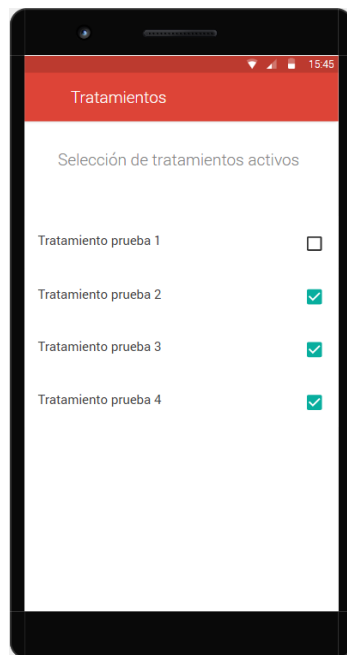


Ilustración 51. Prototipo: medicación asociada.

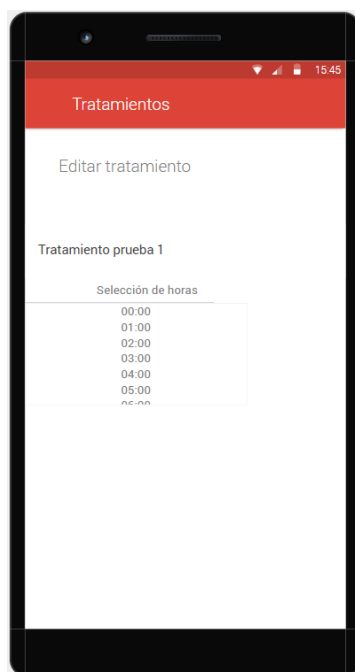


Ilustración 52. Prototipo: configurar medicación.

Como se puede observar, la aplicación móvil es muy sencilla, ya que su finalidad es, principalmente, monitorizar al paciente. Es ella la encargada de detectar un ataque y de avisar vía SMS en caso de que ocurran, pero estas funcionalidades se ejecutan a nivel interno y no requieren de interacción con el usuario ni de interfaz.

7.2. Diagrama entidad-relación

Este diseño conceptual de la base de datos externa tiene como objetivo representar todas las entidades y los campos que la conforman.

En él se encuentran todos los elementos necesarios para el correcto funcionamiento del sistema. Además, se plantea de tal manera para que puede tener una fácil expansión en el futuro si se adquiere una pulsera nueva capaz de medir alguna constante vital más. Como ejemplo de esto se ha añadido la tabla temperatura, que no se utiliza en el proyecto al no disponer la pulsera de sensor capaz de medir este parámetro.

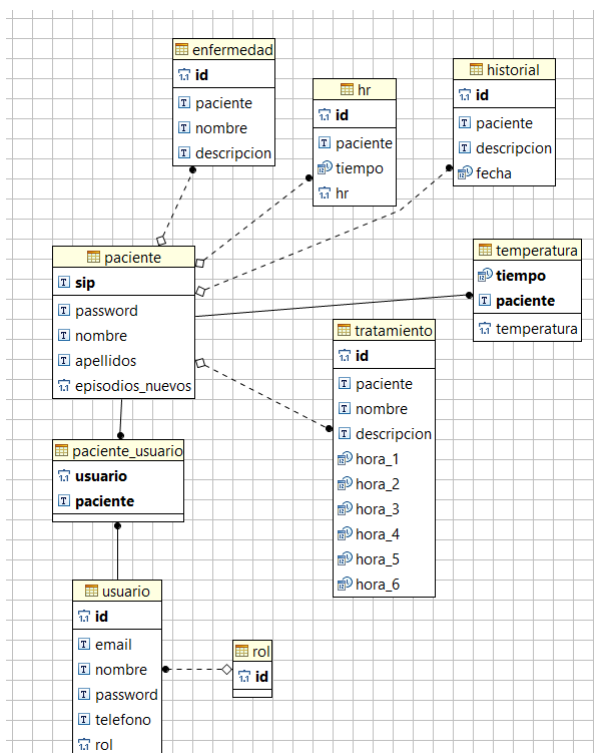


Ilustración 53. Diagrama entidad-relación.

La tabla central es el paciente, que está relacionada con prácticamente todas las demás. Destaca también la tabla usuario que, gracias a su relación con rol, permite la creación de dos tipos diferentes de usuarios (usuario “normal” y personal médico) con una sola tabla.

8. Desarrollo e implementación

El desarrollo del sistema consta de tres partes fundamentales. En primer lugar, la creación de una API restfull conectada a la base de datos descrita anteriormente. Todo esto corresponde al back-end del sistema. Posteriormente se realiza la aplicación web y móvil, cada una con diferentes características: mientras que la móvil se ocupa de monitorizar y alertar, en la web se pueden gestionar usuarios, pacientes y datos de estos últimos. Es por eso por lo que se ha separado su desarrollo, ya que además de alguna llamada común a la API, no comparten código.

8.1. API

Para el desarrollo de la API se ha optado por PHP como lenguaje de programación. El uso de este lenguaje en creación de API's está más que demostrado, siendo uno de los lenguajes más utilizados.

En cuanto a la estructura, se ha buscado desacoplar lo máximo posible la lógica de negocio con la conexión a base de datos. Para ello, la mayoría de ficheros están clasificados en resources, services y DAO. Por ejemplo, para todo lo relacionado con usuario existe UsuarioResource, UsuarioService y UsuarioDAO. A continuación, se explican cada uno de ellos:

- **Resources:** los encargados de detectar el tipo de petición (GET, POST, PUT, DELETE, etc.), así como parsear el resto de la url para identificar la llamada. Además, es el encargado de validar en el caso de que se introduzcan parámetros vía Json.
- **Services:** en ellos, se llama al método del DAO (Data Access Object) que corresponda en cada caso y, una vez recibida la respuesta de la consulta, prepara una respuesta que pueda ser entendida. No siempre es necesario trabajar con la respuesta que proporciona el DAO, pero en caso de que fuera, se realiza a este nivel. Por último, devuelve la respuesta al cliente que la solicitó.
- **DAO:** proporcionan una capa de conexión entre la API y la base de datos. En ellos, se abre la conexión, se crea la consulta y se envía. Está preparada para capturar errores provenientes de la base de datos. Una vez reciba la respuesta, la envía al correspondiente service para su posterior tratamiento.

De toda la API destacar el fichero API.php, que es el encargado de extraer los parámetros de la URL y llamar al método correspondiente. Además, existe un tratamiento de errores en caso de que sea una petición no válida, que el tipo de llamada no se corresponda con los argumentos, o que los campos introducidos no pasen el filtro.


Name	Status
 1 /TFG/Desarrollo/Server/rest/usuario/enfermedad	405 Method Not Allowed

Ilustración 54. API: error 405.

Como se puede ver en la imagen, este error aparece cuando la API no contiene una combinación de tipo y URL válida. En este caso se ha realizado una petición tipo GET con la URL “/usaurio/enfermedad/1”. Al no existir esta llamada devuelve este código de error.

Esto mismo ocurre, por ejemplo, con el error 404 Not Found, que aparece cuando los argumentos no son correctos. La llamada “/hospital/alicante” no contiene argumentos válidos y, por lo tanto, devolvería error 404.

También cabe la posibilidad de que devuelva error 501 Not Implemeted. Esto ocurre cuando el tipo de método no es soportado por la API. Por ejemplo, si se hace una llamada PATH, devolvería este error.

Por último, se han implementado códigos de respuesta para validaciones. Si se necesitan cinco parámetros para hacer un PUT de usuario y en el Json recibido no están todos, se enviará la respuesta con el código 200 y el mensaje “Falta algún parámetro requerido”. Esto sirve de gran utilidad a la hora de desarrollar la aplicación web y móvil, ya que ahorra quebraderos de cabeza y agiliza el trabajo.

8.2. Aplicación web

El propósito de la aplicación web es el de realizar un seguimiento del paciente. Esta web está destinada tanto al personal médico como al usuario que quiere ver las novedades del paciente. En ella, tanto médico como usuario pueden ver los principales datos médicos de los pacientes asociados.

Está realizada con Angular v4.0, siendo su principal característica el uso de Typescript como lenguaje de desarrollo. Este cambio se produjo desde la versión 2.0, y supuso un cambio en la estructura y las posibilidades del framework de Google.

Su principal característica es dividir las páginas en componentes que siguen el patrón modelo vista controlador (MVC). En ellas, el componente realiza la lógica, conecta con los servicios y después, actualiza la vista.

Los servicios son funciones que son incluidas en los componentes. En ellos, se conecta el componente con la API, lo que permite un desacoplamiento entre el componente y la base de datos.

A todo esto, se suma el uso de angular-cli, que permite crear componentes, servicios, e incluso plantillas de aplicaciones con comandos de consola. Al crear un componente, por ejemplo, se modifican los archivos necesarios para incluir todas las dependencias que se necesitan para agregar este nuevo componente al proyecto.

Destacar el uso de un routing basado en hijos, creado de tal manera que se produzca una navegación intuitiva y una estructura definida que facilita el desarrollo y la adición de nuevos componentes.

```
const routes: Routes = [
  { path: '', redirectTo: '/login', pathMatch: 'full' },
  { path: 'login', component: LoginComponent },
  { path: 'panel', component: PanelComponent,
    children: [
      { path: '', redirectTo: '/panel/inicio', pathMatch: 'full' },
      { path: 'inicio', component: InicioComponent },
      { path: 'usuarios', component: UsuariosComponent },
      { path: 'newpaciente', component: NewpacienteComponent },
      { path: 'pacientes', component: PacientesComponent },
      { path: 'pacientes/:id', component: DetailpacienteComponent },
      { path: 'pacientes/:id/hr', component: HrComponent },
      { path: 'guia', component: GuiaComponent },
      { path: 'perfil', component: PerfilComponent },
      { path: "**", redirectTo: "/inicio" }
    ]
  },
  { path: "**", redirectTo: "/login" }
];
```

Ilustración 55. Aplicación web: routing.

En la foto se puede observar la estructura de la web. Para el usuario, esto se traduce en una serie de URL's intuitiva, que le ayudara a situarse en todo momento en la web.

Para el diseño, se ha utilizado una Bootstrap como framework, que permite una clara estructuración visual del contenido. A continuación, se muestran unas capturas de la aplicación. Para entrar más en detalle, acudir al manual de usuario.

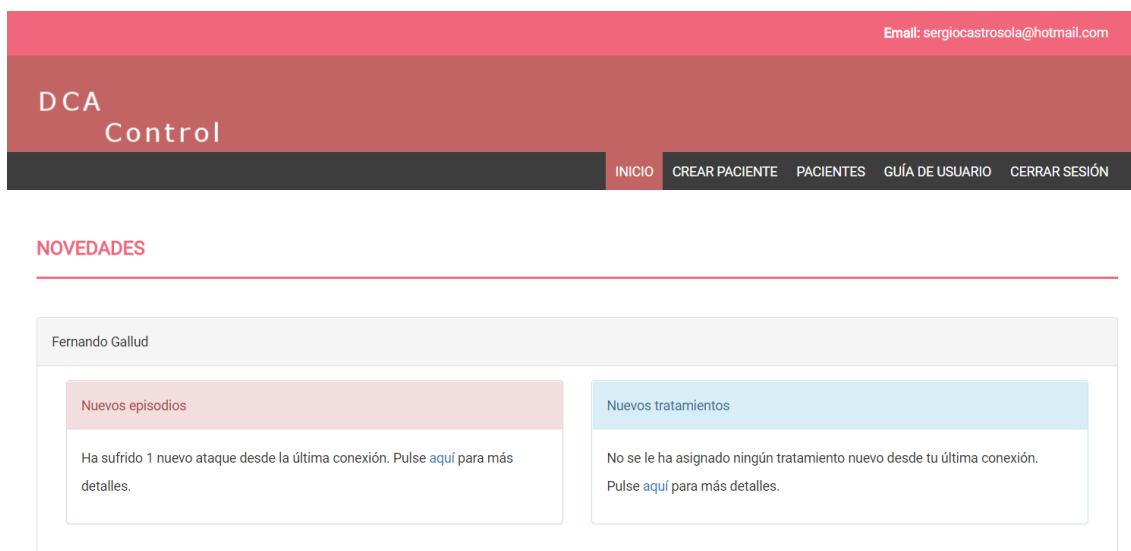


Ilustración 56. Aplicación web: inicio

Como se puede observar en la imagen, la web dispone de un menú superior para acceder a todas las funcionalidades. Cada página se traduce en un componente, mientras que el menú es un componente común a todas las páginas, menos a login.

8.3. Aplicación móvil

Para desarrollar la aplicación móvil se ha utilizado Ionic 2 como framework de desarrollo. Este framework permite el desarrollo de aplicaciones multiplataforma a través de la generación de un Webview.

Este framework funciona por encima de Apache Cordova, lo que es una gran ventaja gracias al gran abanico de plugins que existen. Ionic 2 utiliza Angular como lenguaje de programación, y su estructura es similar a la de la aplicación web. Gracias a esto la curva de aprendizaje ha sido llevadera, salvo la parte de la conexión con la pulsera, que se hablará más adelante.

Los plugins permiten añadir funcionalidades propias de aplicaciones nativas. Gracias a ellos, se puede acceder a la cámara, a las notificaciones, etc. Esta es la lista de los principales plugins utilizados:

- Local notifications: permite crear notificaciones, pudiendo editar el texto y la hora en la que se visualizan. Utilizado para las alertas de la medicación.

- BLE: proporciona las principales funciones para interactuar con los dispositivos Bluetooth. Se ha utilizado para la interacción con la pulsera Xiaomi Mi Band 2.
- SMS: permite enviar mensajes de texto cuyo remitente es el número donde se encuentra la aplicación. Utilizado para enviar alertas cuando se produzca un ataque de epilepsia.

Del mismo modo que la aplicación web, se ha seguido una estructura de componentes y servicios, con lo que las ventanas quedan claramente separadas en el código. También existen cosas comunes, como el menú inferior o el logo de cerrar sesión.

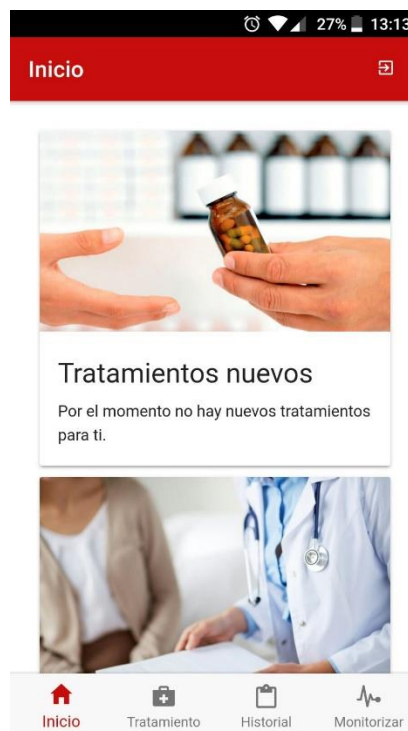


Ilustración 57. Aplicación móvil. Inicio.

Esta imagen se corresponde a la pantalla de inicio. En ella el paciente puede ver si tiene nuevos tratamientos asignados o si hay algún episodio epiléptico que todavía no ha sido visto por el médico.

Se ha utilizado el rojo como color principal, ya que es uno de los asociados a la salud y, en mi caso, a la lectura del ritmo cardíaco. El blanco es el secundario y proporciona sensación de claridad en la aplicación.

La principal complicación de la aplicación ha sido la interacción de la pulsera, ya que el propietario de esta no proporciona ninguna documentación para que los desarrolladores

puedan crear aplicaciones. Para ello se ha utilizado la ingeniería inversa y, a través de prueba y error, se ha conseguido vincular la pulsera y monitorizar al paciente.

El estándar Bluetooth Low Energy (BLE) que utiliza la pulsera se basa en el intercambio de información a través de servicios y características. De un dispositivo se puede conectar, leer y escribir, principalmente.

Para leer, por ejemplo, es necesario decirle en qué servicio se lee y qué característica de este servicio se desea leer. Por ejemplo, para el ritmo cardíaco el estándar dice que el servicio es el 180D, mientras que la característica para leer es la 2a37.

Sin embargo, un dispositivo puede tener servicios y características propias, y son en estas en las que se ha tenido que utilizar ingeniería inversa.

Los pasos para leer el ritmo cardíaco en la Xiaomi Mi Band 2 son:

1. Escanear: Sirve para buscar los dispositivos que se encuentren en el rango del móvil. Para versiones de Android superiores a la 6.0 es necesario activar la geolocalización.
2. Vincular: Permite la interacción con la pulsera de manera constante sin interrupciones ni fallos de conexión. Utiliza un servicio propio cuyo código es “feel”, y la característica con la que se produce la comunicación es “00000009-0000-3512-2118-0009af100700”. Consta de tres pasos principales de escritura, mientras se escucha en esa característica para recibir las respuestas. Al final de este punto se muestra una captura resumen de este paso.
 - 2.1. Escribir en dicha característica un array de bytes cuyo contenido es la clave secreta de la pulsera, precedido de unas cabeceras de autenticación. Estas cabeceras son los bytes 0x01 y 0x08. La respuesta al escuchar es un array cuyo contenido es 0x10, 0x01, 0x01.
 - 2.2. Escribir un array de bytes formado por 0x02 y 0x08. La respuesta es otro array de bytes cuyo contenido destacable es a partir del cuarto elemento, que es un código aleatorio que habrá que cifrar y enviar en el siguiente paso. Los tres primeros elementos son cabeceras.
 - 2.3. Realizar cifrado AES ECB de la respuesta anterior (sin cabeceras) con la clave secreta de la pulsera como key. Para esto se ha implementado un plugin que accede a librerías nativas y se encarga de cifrar. El resultado de este cifrado se

envía en un array de bytes cuyos dos primeros elementos son 0x03 y 0x08. Si el cifrado se ha realizado correctamente se habrá vinculado la pulsera.

```
console.log("Conectando...");
this.ble.connect(this.idselect).subscribe(data => {
  console.log("Conectado");
  this.conectado = true;
  console.log("Vinculando...");
  console.log("Realizando la primera escritura..");
  this.ble.writeWithoutResponse(this.idselect, 'fee1', '00000009-0000-3512-2118-0009af100700', datosEnv.buffer); //se envía la clave secreta
  this.ble.startNotification(this.idselect, 'fee1', '00000009-0000-3512-2118-0009af100700').subscribe(buffer => {
    console.log("Ha habido cambios! se ha recibido lo siguiente:");
    let data = new Uint8Array(buffer);
    console.log(data);
    if(data[1]==0x01){ //he recibido la primera respuesta, procedo a la segunda escritura
      this.segundowrite();
    }else if(data[1]==0x02){ //he recibido la segunda respuesta, procedo al cifrado y a la tercera escritura
      this.cifrar(data);
    } else{ //he recibido la tercera respuesta, proceso finalizado
      console.log("Ultimo cambio en el servicio fee1:");
      let data = new Uint8Array(buffer);
      console.log(data);
      this.navCtrl.pop();
      console.log("Cancelando escucha en servicio fee1...")
      this.ble.stopNotification(this.idselect, 'fee1', '00000009-0000-3512-2118-0009af100700');
    }
  });
});
```

Ilustración 58. Aplicación móvil: vinculación.

3. Pedir el ritmo cardíaco en un intervalo de tiempo: Para ello son necesarios un paso de escritura y uno de escucha. Para activar el sensor en la pulsera y medir es necesario realizar una escritura en el servicio “180d”, en la característica “2a39”. Se a de enviar un array de bytes formado por los elementos 0x15, 0x02 y 0x01. Una vez hecho esto se puede observar cómo se activa el sensor en la pulsera. La escritura es necesaria hacerla en un intervalo de tiempo establecido en 20 segundos para monitorizar constantemente al paciente. Por último, es necesario realizar una escucha en este mismo servicio, pero en la característica “2a37”. La respuesta de la escritura es un array de bytes cuyo segundo elemento es el ritmo cardíaco. A continuación, se muestra una captura de la parte del código que se encarga de activar el sensor y medir el ritmo cardíaco.


```
console.log("Pedir HR");
this.datosEnv = new Uint8Array(3);
this.datosEnv[0] = 0x15;
this.datosEnv[1] = 0x02;
this.datosEnv[2] = 0x01;

this.ble.write(this.idband, '180d', '2a39', this.datosEnv.buffer).then(buffer => {
  console.log("Pidiendo HR...");

}, error=>{
  console.log("Error al escribir");
});
```

Ilustración 59. Aplicación móvil: Activar sensor HR.

Con todo esto solo hace falta crear la lógica de guardado de datos y de detección de ataque epiléptico. En el primer caso, se ha creado un intervalo para que cada 10 minutos se inserten en la base de datos los registros obtenidos.

En caso de la alerta de ataque epiléptico, se ha establecido un rango en función de la media de las pulsaciones que, si se sobrepasa, se alertará de ataque epiléptico vía SMS.

9. Conclusiones

Una vez finalizado el proyecto, es posible realizar una valoración global del mismo para determinar si se han cumplido los objetivos iniciales, a la vez que valorar las mejoras futuras.

9.1. Revisión de los objetivos

Los objetivos establecidos al inicio del proyecto son los siguientes:

- **Investigar y analizar proyectos similares.** Se ha realizado un estudio de diversas aplicaciones o sistemas enfocados a la s-Salud, Siendo la mayoría puramente informativos.
- **Investigar en profundidad la enfermedad a tratar.** Investigación realizada sobre el DCA y la epilepsia para niños, con el fin de comprender las causas, consecuencias y los síntomas que preceden un ataque epiléptico.
- **Estudiar el mercado para la correcta selección de la *smartband* a emplear.** Se han revisado las principales smartband del mercado. La pulsera seleccionada tenía que tener los sensores necesarios para la detección de ataques epilépticos y ser económica.
- **Investigar y analizar la pulsera inteligente *Xiomi miBand 2* mediante ingeniería inversa.** Se ha tenido que averiguar cómo funcionaba la pulsera y cómo se vinculaba a un smartphone.
- **Establecer los requisitos mínimos que ha de cumplir la aplicación.** Los objetivos mínimos del sistema se han cumplido. Gracias a ello, se obtiene un sistema funcional que permite monitorizar, alertar y realizar un seguimiento detallado.
- **Diseñar la aplicación móvil en consecuencia a los requisitos establecidos.** Se han cumplido también objetivos no principales como el envío de notificaciones de avisos de medicación a modo de recordatorio para el paciente.
- **Implementar prototipos en cada iteración del proyecto.** Se han realizado prototipos funcionales en cada iteración, que se han ido uniendo hasta conseguir el producto final.
- **Entregar la aplicación móvil en su versión acabada.** Se finalizó con éxito la aplicación móvil que es completamente funcional. Además, también es funcional la aplicación web encargada del seguimiento.

9.2. Trabajos futuros

Este proyecto tiene margen de mejora en varios aspectos. Aunque es un sistema completo y estable, siempre existen formas de mejorarlo y añadir más funcionalidades y fiabilidad. Para complementarlo, se han planteado una serie de ideas entre las que destacan:

- **Detección de otras enfermedades o sucesos:** estudiar qué patologías se pueden detectar con los sensores disponibles, o alertar de nuevos sucesos como caídas accidentales.
- **Utilización de otra smartband:** permitir que se pueda monitorizar usando diferentes smartband, no solo Xiaomi Mi Band 2. Esto permitiría flexibilidad a la hora de que los usuarios se decidan por DCA Control.
- **Estudio del ritmo cardíaco:** emplear algoritmos de aprendizaje que detecten de manera personalizada los ataques epilépticos. Con esto se reducirían los falsos positivos y se mejoraría la fiabilidad general del sistema.

9.3. Conclusiones

El desarrollo de este proyecto ha sido una experiencia muy satisfactoria. En los primeros compases se profundizó en el problema y en la obtención de una solución. Los niños que padecen DCA acompañado de ataques epilépticos y sus familiares necesitan sistemas capaces de facilitarles la convivencia con estas patologías y desde el punto de vista de un ingeniero multimedia es lo que se ha tratado de hacer. Actualmente, con el indudable auge de los wearables surgen nuevas posibilidades a la hora de desarrollar sistemas enfocados a la e-Salud.

Sin embargo, todavía nos encontramos en una etapa muy prematura, por lo que la fiabilidad y confianza en estos sistemas ha de fortalecerse empleando el mejor hardware y software posible.

DCA Control ha conseguido ser el sistema que en un principio se quería: un sistema formado por aplicación web, aplicación móvil y wearable que fuera capaz de monitorizar, alertar y realizar un seguimiento del paciente con un precio económico.

Es evidente que con la elección de otra smartband más costosa se podrían haber conseguido extender las funcionalidades, pero la elección de Xiaomi Mi Band 2 como encargada de monitorizar ha sido un éxito, ya que se ha conseguido darle un uso

sanitario a una smartband inicialmente creada para el ejercicio físico. Además, es una de las más económicas del mercado sino la que más, y se puede encontrar en muchos establecimientos. Las aplicaciones, por su parte, son sencillas de utilizar, por lo que el público objetivo de este sistema no necesita tener grandes conocimientos ni de informática ni de salud.

10. Referencias

ADACEA, 2017. Autor. [En línea] <http://www.adaceaalicante.org/que-es-el-dano-cerebral/>.

Gala M. García, 2016. Monitorización de pacientes con pulseras inteligentes, página 34.

Observatorio del ictus, 2017. Autor [En línea] <http://www.observatoriodelictus.com/index.php/que-es-un-ictus>

Bárcena Orbe et al, 2006.
http://scielo.isciii.es/scielo.php?script=sci_arttext&pid=S1130-14732006000600001&lng=es&nrm=iso&tlng=es

FEDACE, [En línea]. Qué es (el daño cerebral) y causas principales. <https://fedace.org/>

Reintegra, [En línea]. ¿Qué es el Daño Cerebral Adquirido? <http://reintegra-dca.es/que-es-el-dano-cerebral-adquirido/>

Servicio de neurorrehabilitación, hospitales Nisa. [En línea]. Daño Cerebral Adquirido. <https://www.neurorhb.com/dano-cerebral-adquirido/>

ATENEU Castelló, [En línea]. ¿Qué es el daño cerebral adquirido?
http://www.ateneocastellon.org/que_es_el_da%C3%B1o_cerebral_adquirido

Cuídate Plus,2015. [En línea]. Epilepsia.
http://www.ateneocastellon.org/que_es_el_da%C3%B1o_cerebral_adquirido

ADACCA, 2014. Adacca. El DCA en los colegios. <http://adacca.org/el-dca-en-los-colegios/>

Maite Palacios, 2014. Qué es el daño cerebral y cuáles son sus secuelas.
<http://www.adacen.org/blog/index.php/que-es-el-dano-cerebral-y-cuales-son-sus-secuelas/>

Martha Quezada García, Agustín Huete García, 2013. Magnitud del Daño Cerebral Adquirido en España y en Castilla La-Mancha.
<http://www.adaceclm.org/DocumentosInteres/Documentosda%C3%B1ocerebral/Documents/Magnitud%20de%20da%C3%B1o%20Cerebral%20Sobvenido%20en%20Espa%C3%B1a%20y%20Castilla-La%20Mancha%202013.pdf>

ADACEN, [En línea]. El DCA en datos. <http://www.adacen.org/el-dca-en-datos/>

CogniFit, [En línea]. Lesiones cerebrales, las alteraciones en los tejidos del cerebro. <https://www.cognifit.com/es/lesiones-cerebrales>

Der lange, 2005. Epilepsia. <https://commons.wikimedia.org/wiki/File:Spike-waves.png>

Joseph Roger, Marco Tulio Medina, Pierre Genton, Charlotte Dravet, Michelle Bureau, [En línea]. Avances en la clasificación de las epilepsias y los síndromes epilépticos. <http://www.uninet.edu/neurocon/congreso-1/conferencias/epilepsia-4.html>

G. Eysenbach, 2001. What is e-health? <https://www.jmir.org/2001/2/e20/>

Chema Cepeda, 2015. eSalud: definición y evolución del término. <https://saludconectada.com/e-salud/>

Appio, 2013. Tipos de apps: nativas, híbridas y web apps. <http://appio.es/tipos-de-apps/>

Danyal Zia, 2017. Ionic 2 vs ReactNative vs NativeScript. <http://www.discoversdk.com/blog/ionic-2-vs-reactnative-vs-nativescript>

Danny Markov, 2015. Comparing The Top Frameworks For Building Hybrid Mobile Apps. <http://tutorialzine.com/2015/10/comparing-the-top-frameworks-for-building-hybrid-mobile-apps>

Pulseguard, [En línea]. <https://www.pulseguard.org/>

Epilepsy society, 2015. Free epilepsy smartphone app. <https://www.epilepsysociety.org.uk/free-epilepsy-smartphone-app>

EpDetect, 2012. EpDetect Application. http://www.epdetect.com/epilepsy_mobile_phone_application_features.html

EpSMon, [En línea]. EpSMon – Epilepsy Self-Monitor. <https://sudep.org/epilepsy-self-monitor>

Daño Cerebral app, 2015. <https://www.neurorhb.com/apps/>

Gala M. García, DCA Epilepsia, 2016. https://i-gala.github.io/epilepsia_dca/

Embrace empática, [En línea]. <https://www.empatica.com/product-embrace>

Olmo Axayacati, 2013. ¿Qué es la ingeniería inversa?

<https://blogingenieria.com/general/ingenieria-inversa/>

MutanteHenry, 2013. Procesos de la ingeniería directa, inversa, reingeniería y copia.

<https://commons.wikimedia.org/wiki/File:ProyectosEng.pdf>

11. Anexo

11.1. Documentación de la API

A continuación, se muestran todos los tipos de llamadas implementados para la API. Todas ellas están compuestas por la combinación de uno o los dos tipos de elementos: recursos o id. Un recurso es la palabra identificativa que determina qué hace esa llamada. Un ejemplo de recurso sería “usuario” o “tratamiento”.

Por otra parte, un id determina con qué elemento de un recurso dado se trabaja. En el caso de usuarios, la llamada sería “/usuario/1”, donde 1 es el id, que sirve para identificar a un usuario concreto de una lista de usuarios.

En algunas es posible enviar datos mediante Json. Esto se utiliza para insertar o modificar en la base de datos. Otras, en cambio, permiten realizar filtros de búsqueda, en este caso mediante Query string.

Dicho esto, las llamadas a la API son las siguientes:

Historial				
Tipo	URL	Descripción	Formato de envío	Filtros
GET	/historial/paciente/id_paciente	Listar el historial de un paciente	-	-
GET	/historial/id_historial	Detalle de un episodio	-	-
PUT	/historial	Crear episodio	paciente descripcion fecha	-
POST	/historial/id	Modificar episodio	descripcion fecha	-
DELETE	/historial/id	Eliminar episodio	-	-

Tabla 82. API: historial.

Ritmo cardíaco				
Tipo	URL	Descripción	Formato de envío	Filtros
GET	/hr/paciente/id_paciente	Listar el HR de un paciente	-	fecha
PUT	/hr	Crear registro HR	paciente tiempo: [t1, t2, ...] hr: [hr1, hr2, ...]	-

Tabla 83. API: ritmo cardíaco.

Login				
Tipo	URL	Descripción	Formato de envío	Filtros
POST	/login/paciente	Login paciente. App móvil.	sip password	-
POST	/login/usuario	Login usuario. Para la web.	email password	-
POST	/rest/login/password	Recuperar contraseña	id	-

Tabla 84. API: login.

Paciente				
Tipo	URL	Descripción	Formato de envío	Filtros
GET	/paciente/id	Detalle de paciente	-	-
GET	/paciente/usuario/id_paciente	Usuario responsable del paciente	-	-
GET	/paciente/novedades/id	Novedades paciente.	-	-
POST	/paciente/novedades/id	Reestablece las novedades del paciente.	tipo	-
PUT	/paciente	Crear paciente	sip password nombre apellidos doctor	-
POST	/paciente/id	Modificar paciente	password nombre apellidos	-
DELETE	/paciente/id	Eliminar paciente	-	-

Tabla 85. API: paciente.

Patología				
Tipo	URL	Descripción	Formato de envío	Filtros
GET	/patología/paciente/id_paciente	Lista de patologías de un paciente	-	-
GET	/patología/id	Detalle de una patología	-	-
PUT	/patología	Crear patología	paciente nombre descripcion	-
POST	/patologia/id	Modificar patología	nombre desripcion	-
DELETE	/patologia/id	Eliminar patologia		-

Tabla 86. API: patología.

Temperatura				
Tipo	URL	Descripción	Formato de envío	Filtros
GET	/temperatura/paciente/id_paciente	Listar la temperatura de un paciente	-	fecha
PUT	/temperatura	Crear registro temperatura	paciente tiempo: [t1, t2, ...] temperatura: [hr1, hr2, ...]	-

Tabla 87. API: temperatura

Tratamiento

Tipo	URL	Descripción	Formato de envío	Filtros
GET	/tratamiento/paciente/id_paciente	Listar los tratamientos de un paciente	-	-
GET	/tratamiento/id	Detalle de un tratamiento	-	-
PUT	/tratamiento	Crear tratamiento	paciente nombre descripción hora_1, ... , hora_6	-
POST	/tratamiento/id	Modificar tratamiento	nombre descripción hora_1, ... , hora_6	-
DELETE	/tratamiento/id	Eliminar tratamiento	-	-

Tabla 88. API: tratamiento.

Usuario				
Tipo	URL	Descripción	Formato de envío	Filtros
GET	/usuario/id	Detalle de un usuario	-	-
GET	/usuario/paciente/id_usuario	Pacientes de un usuario	-	-
GET	/usuario/novedades/id	Novedades desde la última conexión	-	-
GET	/usuario/medico	Listado médicos	-	-
PUT	/usuario	Crear usuario	email nombre password telefono rol	-
POST	/usuario/id	Modificar usuario	email nombre password telefono	-
DELETE	/usuario/id	Eliminar usuario	-	-

Tabla 89. API: usuario.

11.2. Manual de usuario

Este manual pretende servir de guía para los distintos tipos de usuarios.

11.2.1. Registro web

DCA
Control

POR FAVOR INICIA SESIÓN PARA ACCEDER

Inicia sesión con tu cuenta de **Health control** :

Email :

Contraseña :

 Iniciar sesión

O regístrate :

Email :

Nombre :

Contraseña :

Repetir contraseña :

Teléfono :

Personal médico

 Registro

Ilustración 60. Manual: Registro.

Este es el primer paso para acceder a todas las funcionalidades de DCA Control. Mediante el registro se produce la creación de un usuario o de personal médico en la base de datos.

11.2.2. Inicio

The screenshot shows the 'Inicio' page of the 'DCA Control' system. At the top, there is a dark red header with the text 'DCA Control' and a navigation bar with links: 'INICIO', 'CREAR PACIENTE', 'PACIENTES', 'GUÍA DE USUARIO', and 'CERRAR SESIÓN'. Below the header, the 'NOVEDADES' section is highlighted in red. It contains two patient cards. The first card is for 'Fernando Gallud' and has two sub-sections: 'Nuevos episodios' (highlighted in light red) stating 'Ha sufrido 1 nuevo ataque desde la última conexión. Pulse aquí para más detalles.' and 'Nuevos tratamientos' (highlighted in light blue) stating 'No se le ha asignado ningún tratamiento nuevo desde tu última conexión. Pulse aquí para más detalles.' The second card is for 'pepe perez' and has similar sub-sections: 'Nuevos episodios' stating 'No ha sufrido ningún ataque desde la última conexión. Pulse aquí para más detalles.' and 'Nuevos tratamientos' stating 'No se le ha asignado ningún tratamiento nuevo desde tu última conexión. Pulse aquí para más detalles.'

Ilustración 61. Manual: inicio.

En esta página se visualizan rápidamente las novedades para cada uno de los pacientes. Se muestran los ataques y los tratamientos nuevos desde la última conexión. De esta forma, de un golpe de vista se pueden visualizar las novedades de cada paciente.

11.2.3. Crear paciente

DCA Control

INICIO CREAR PACIENTE PACIENTES GUÍA DE USUARIO CERRAR SESIÓN

CREAR PACIENTE

Crea el paciente que estará a tu cargo

SIP :

Contraseña :

Repite la contraseña :

Nombre :

Apellidos :

Médico:

[Crear paciente](#)

Ilustración 62. Manual: crear paciente.

Al pulsar sobre esta pestaña aparece un formulario para registrar un paciente. Este paciente es el que accederá a la aplicación móvil en el futuro. Destaca el campo de elección del médico, con el que se asigna el paciente creado a un médico para que pueda realizar un seguimiento profesional.

11.2.4. Lista de pacientes

DCA Control

INICIO CREAR PACIENTE PACIENTES GUÍA DE USUARIO CERRAR SESIÓN

LISTA DE PACIENTES

SIP	Nombre	Apellidos	Acciones
123456789123	Fernando	Gallud	
123123123123	pepe	perez	

Ilustración 63. Manual: listar pacientes.

Esta página sirve para visualizar los pacientes asociados. Para cada paciente se puede acceder a detalle o directamente a la página de seguimiento del ritmo cardíaco.

11.2.5. Detalle de paciente

En esta página se muestra con detalla toda la información que se dispone de los pacientes. Aparecen los datos principales, datos del usuario responsable, patologías, tratamientos y ataques sufridos. Además, dispone de una gráfica con los últimos datos del ritmo cardíaco.

DETALLE DE PACIENTE

Información personal

Nombre:
Fernando

Apellidos:
Gallud

SIP:
123456789123

Contacto:

Nombre:
Sergio

Teléfono:
637126771

Correo electrónico:
sergiocastrosola@hotmail.com

[Editar Perfil](#)

Patologías:

Nombre	Descripción
DCA	DCA desde el nacimiento, se necesita revisión periódicamente.
Epilepsia	Epilepsia controlada, se espera nuevo diagnóstico en breve.

Últimos datos médicos

00:37:55 00:38:20 00:38:38 00:38:55 00:39:14 00:39:34 00:39:54 00:40:14 00:40:34 00:40:54

Historial médico

Nombre	Descripción
20/06/2017	Ataque nocturno leve.
21/05/2017	Ataque sufrido por la noche. Se precisó ingreso hospitalario.
13/01/2017	Ataque repentino sufrido mientras estaba en casa.

Tratamiento actual

Nombre	Descripción
Carbamacepina	Tomar con comidas. una dosis al día.
Lacosamida	Tomar una cada 8 horas.

Ilustración 64. Manual: detalle de paciente.

Destacar que hay opciones que aparecerán o no en función de si eres el responsable o el médico asociado. En caso de que seas la persona responsable del paciente, podrás modificar tu perfil pulsando el botón “Editar Perfil” situado en la parte izquierda. Por otro lado, si eres el médico asociado al paciente, podrás añadirle un nuevo tratamiento, que se le notificará tanto al responsable como al paciente.

11.2.6. Seguimiento del ritmo cardíaco

A esta página se accede pulsando sobre la gráfica de detalle de paciente. Se puede visualizar un formulario de filtro para realizar un seguimiento detallado por fecha y hora. El resultado es una gráfica con la que se puede interactuar para añadir o quitar zoom, o descargar los resultados en varios formatos (SVG, PDF) imprimir la gráfica e incluso pintar por encima, todo gracias al botón situado en la esquina superior derecha de la gráfica.

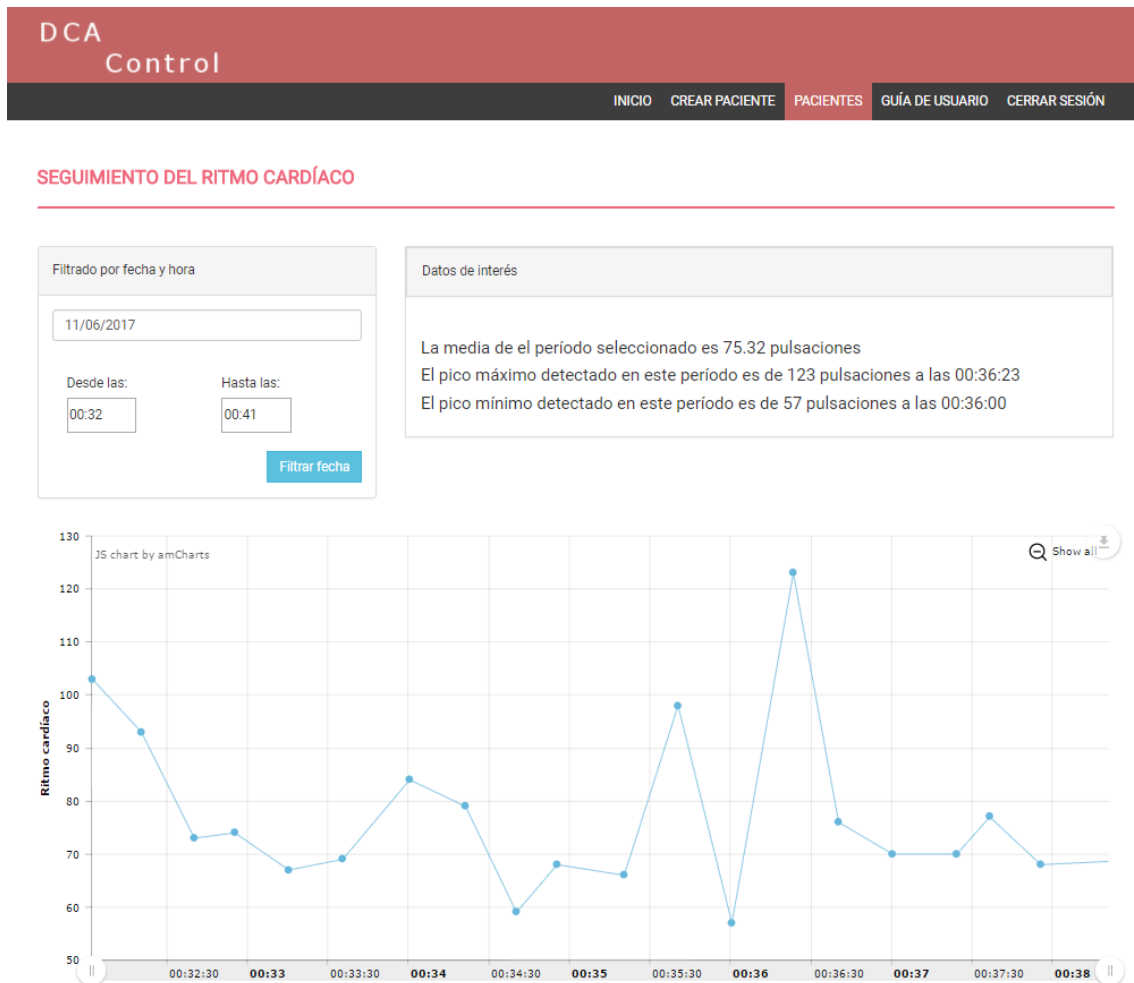
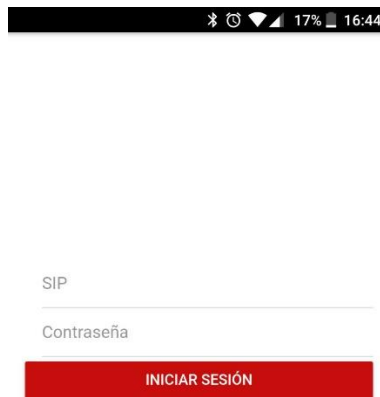


Ilustración 65. Manual: seguimiento.

Además, para cada período consultado, obtiene la media y los picos mínimos y máximos, con el fin de facilitar al personal cualificado datos que le sirvan de ayuda.

11.2.7. Login móvil



SIP

Contraseña

INICIAR SESIÓN

Ilustración 66. Manual: login.

Esta es la pantalla inicial de la aplicación móvil. Para el acceso se necesitan el SIP y la contraseña creadas por el usuario en la creación de paciente. Una vez iniciado sesión, no es necesario volverlo a hacer salvo que se cierre sesión.

11.2.8. Home



Ilustración 67. Manual: home.

La pantalla principal, donde un paciente puede ver las novedades relacionadas con él. Aparecen del mismo modo que en la aplicación web, pero en este caso no se actualizan al pulsar sobre ellas. Esto es debido a que se prioriza la visualización por parte del médico o del responsable antes que la del paciente, ya que en muchos casos será un niño.

11.2.9. Tratamiento

Esta pestaña inicialmente muestra la lista de los tratamientos activos del paciente. Si se pulsa sobre cualquiera de ellas, se puede acceder al detalle. En esta página se muestra tanto el nombre como la descripción proporcionada por el médico.

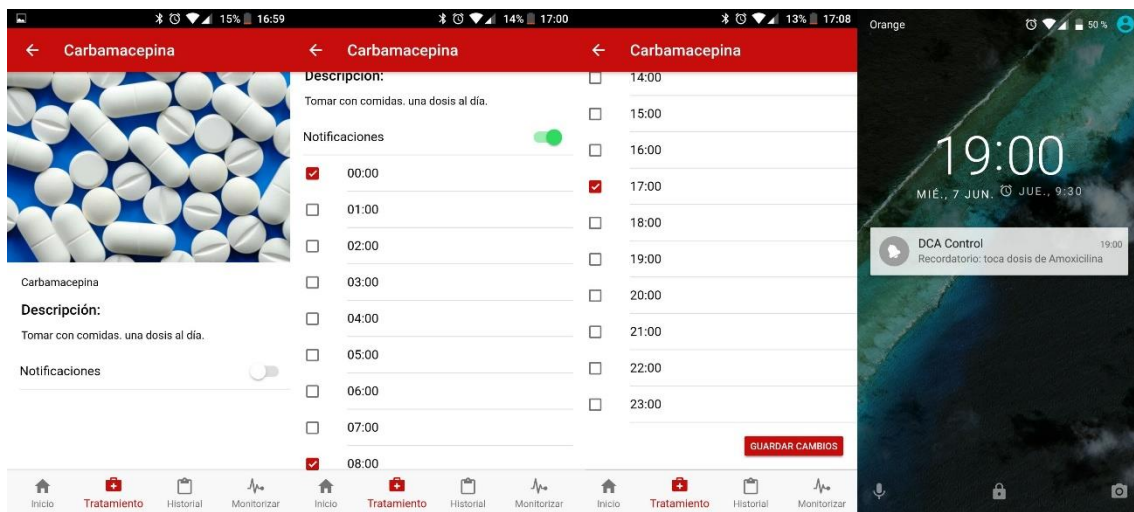


Ilustración 68. Manual: tratamiento.

Opcionalmente, se pueden activar las notificaciones a modo de recordatorio. Para activarlas, se pulsa sobre el botón correspondiente y se seleccionan las horas a las que se desean recibir notificaciones. Por último, se guardan los cambios pulsando el botón inferior de la página.

11.2.10. Historial

En esta sección, como en tratamiento, inicialmente se muestra una lista del historial sufrido por el paciente. Al pulsar en uno se accede al detalle.



Ilustración 69. Manual: historial.

La principal característica de esta ventana es que se puede editar la descripción del ataque, para informar así de lo ocurrido al médico.

11.2.11. Monitorizar

En esta página se puede realizar el escaneo y la monitorización del paciente. Para ello es necesario disponer de Android 4.4 o superior, ya que la tecnología Bluetooth Low Energy (BLE) solo está disponible para estos dispositivos.

El primer caso es la búsqueda de dispositivos. Al pulsar el botón se abrirá la ventana donde se escanea y muestran los dispositivos encontrados. Para vincular con Xiaomi Mi Band 2, pulsa sobre ella en los resultados. A continuación, la pulsera vibrará y tendrás que pulsar el botón que posee. Una vez vinculado se vuelve a la página anterior.

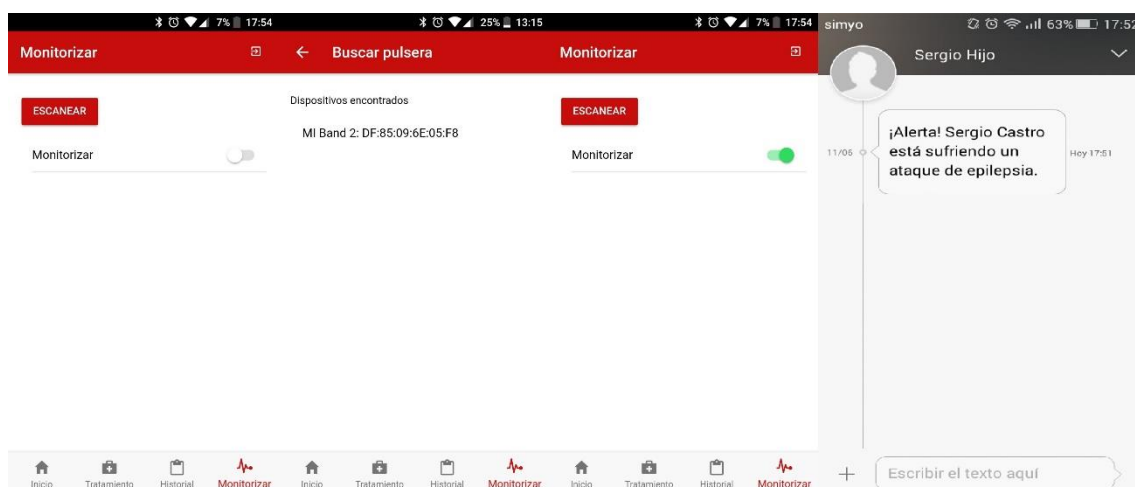


Ilustración 70. Manual: monitorizar.

Cuando se haya vinculado. Se podrá activar la monitorización. Si se produce un ataque epiléptico, se enviará un SMS al teléfono móvil de la persona responsable.

11.3. Enlaces de interés

En este enlace a GitHub se encuentra todo el código desarrollado en el proyecto: tanto API como aplicación web y móvil.

<https://github.com/sergiocastro95/TFG>