



UNIVERSIDAD DE JAÉN
Escuela Politécnica Superior de Jaén

Trabajo Fin de Grado

APLICACIÓN PARA LA ASIGNACIÓN, REALIZACIÓN Y SEGUIMIENTO DE TERAPIAS EN CASA.

Alumno: Juan Antonio Béjar Martos

Tutor: Antonio Jesús Rueda Ruiz
Dpto: Departamento de Informática

Junio, 2019



Universidad de Jaén
Escuela Politécnica Superior de Jaén
Departamento de Informática

Don Antonio Jesús Rueda Ruiz, tutor del Proyecto Fin de Carrera titulado: “Aplicación para la asignación, realización y seguimiento de terapias en casa.”, que presenta Juan Antonio Béjar Martos, autoriza su presentación para defensa y evaluación en la Escuela Politécnica Superior de Jaén.

Jaén, Junio de 2019

El alumno:

Los tutores:

Juan Antonio Béjar Martos

Antonio Jesús Rueda Ruiz

Índice de contenido

1. Introducción.....	6
2. Objeto del proyecto.....	7
3. Antecedentes.....	8
4. Descripción de la situación actual.....	12
4.1. Descripción del entorno actual.....	12
4.2. Resumen de las principales deficiencias identificadas.....	12
5. Normas y referencias.....	13
5.1. Disposiciones legales y normas aplicadas.....	13
5.2. Bibliografía.....	13
5.3. Métodos, Herramientas, Modelos, Métricas y Prototipos.....	14
5.3.1. Métodos y herramientas.....	14
5.3.2. Modelos, métricas y prototipos.....	16
5.4. Mecanismos de control de calidad aplicados durante la redacción del proyecto.....	20
5.5. Otras referencias.....	20
6. Definiciones y abreviaturas.....	21
7. Requisitos iniciales.....	22
8. Alcance.....	24
9. Hipótesis y restricciones.....	26
10. Estudio de alternativas y viabilidad.....	27
11. Descripción de la solución propuesta.....	29
12. Análisis de riesgos.....	32
13. Organización y gestión del proyecto.....	34
13.1. Organización.....	34
13.1.1. Actores del proyecto y relaciones entre los mismos.....	34
13.1.2. Roles y responsabilidades.....	34
14- Planificación temporal.....	35
14.1 - Evolución del plan de proyecto.....	35
14.1.1 - Iteración 0.....	37
14.1.2 - Iteración 1.....	37
14.1.3 - Iteración 2.....	38
14.1.4 - Iteración 3.....	39
14.1.5 - Iteración 4.....	41
14.1.6 - Iteración 5.....	42
14.1.6 - Iteración 6.....	43
14.1.7 - Iteración 7.....	44
14.1.7 - Iteración 8.....	45
14.1.8 - Iteración 9.....	46
15- Resumen del presupuesto.....	47
16. Orden de prioridad de los documentos básicos del proyecto.....	48
17. Anexos.....	49
17.1. Documentación de entrada.....	49
17.2. Análisis y diseño del sistema.....	50

17.2.1 - Estructura del backend.....	50
Modelo del sistema.....	50
Servicios.....	52
Interfaces REST.....	52
DAOs.....	53
Diagrama de paquetes del backend.....	54
17.2.2- Diagrama entidad - relación de la base de datos.....	55
17.2.3 - Estructura del frontend.....	56
17.3. Plan de seguridad.....	57
18. Especificaciones del sistema.....	58
18.1 Historias de usuario.....	58
18.1.1 Médico crea ejercicio terapéutico.....	59
18.1.2. Médico modifica ejercicio terapéutico.....	60
18.1.3. Médico ve ejercicios terapéuticos.....	61
18.1.4. Usuario se registra en el sistema.....	62
18.1.5. Médico añade paciente.....	64
18.1.6. Médico asigna una terapia a un paciente.....	65
18.1.7. Médico consulta historial médico de un paciente.....	66
18.1.8. Médico modifica historial clínico de un paciente.....	67
18.1.9. Médico consulta terapias asignadas a un paciente.....	68
18.1.10. Médico ve progreso de una terapia.....	69
18.1.11. Usuario hace login en el sistema.....	70
18.1.12 Usuario envía mensaje de una terapia.....	71
18.1.13. Usuario lee mensaje de una terapia.....	72
18.1.14. Usuario modifica mensaje de una terapia.....	73
18.1.15. Paciente consulta terapias asignadas.....	74
18.1.16. Paciente realiza una terapia.....	75
18.1.17. Médico asigna un video a un ejercicio terapéutico.....	76
18.1.18. Usuario ve video de ejercicio terapéutico.....	77
18.1.19. Administrador registra un médico.....	78
18.1.20. Administrador ve un médico.....	79
18.1.21. Administrador modifica un médico.....	80
18.1.22. Usuario configura perfil.....	81
18.2. - End-points de la aplicación.....	82
19. Presupuesto.....	85
19.1. Coste de personal.....	85
19.2. Coste de recursos tecnológicos.....	86
19.3. Importe final.....	87
20. Estudios con entidad propia.....	88
20.1. Modelo de negocio.....	88
20.2. Despliegue de la aplicación.....	90
21. Conclusiones y trabajos futuros.....	91

Índice de ilustraciones

<i>Ilustración 1 . Interfaz MyTherapy(I)</i>	8
<i>Ilustración 2 . Interfaz MyTherapy (II)</i>	9
<i>Ilustración 3 . Interfaz MyTherapy(III)</i>	9
<i>Ilustración 4 . Interfaz Clinni (I)</i>	10
<i>Ilustración 5 . Interfaz Clinni (II)</i>	10
<i>Ilustración 6 . Cuadrantes de la gestión de proyectos</i>	14
<i>Ilustración 7 . Tipos de gestión del ciclo de vida de un proyecto</i>	14
<i>Ilustración 8 . Ciclo de vida de un proyecto iterativo</i>	15
<i>Ilustración 9 . Interfaz del prototipo desarrollado (I)</i>	17
<i>Ilustración 10 . Interfaz del prototipo desarrollado (II)</i>	17
<i>Ilustración 11 . Interfaz del prototipo desarrollado (III)</i>	18
<i>Ilustración 12 . Interfaz del prototipo desarrollado (IV)</i>	18
<i>Ilustración 13 . Interfaz del prototipo desarrollado (V)</i>	19
<i>Ilustración 14 . Triángulo del proyecto en gestión de proyectos tradicionales y ágiles</i>	24
<i>Ilustración 15 . Representación de una aplicación empresarial</i>	29
<i>Ilustración 16 . Arquitectura basada en capas</i>	29
<i>Ilustración 17 . Java</i>	30
<i>Ilustración 18 . Spring</i>	30
<i>Ilustración 19 . AngularJS</i>	30
<i>Ilustración 20 . Matriz de probabilidad e impacto</i>	32
<i>Ilustración 21 . Diagrama de evolución inicial</i>	36
<i>Ilustración 22 . Diagrama de evolución del proyecto al comienzo de la iteración 2</i>	38
<i>Ilustración 23 . Diagrama de evolución del proyecto al comienzo de la iteración 3</i>	39
<i>Ilustración 24 . Diagrama de evolución del proyecto replanificado</i>	40
<i>Ilustración 25 . Diagrama de evolución del proyecto al comienzo de la iteración 4</i>	41
<i>Ilustración 26 . Diagrama de evolución del proyecto al comienzo de la iteración 5</i>	42
<i>Ilustración 27 . Diagrama de evolución del proyecto al comienzo de la iteración 6</i>	43
<i>Ilustración 28 . Diagrama de evolución del proyecto al comienzo de la iteración 7</i>	44
<i>Ilustración 29 . Diagrama de evolución del proyecto al comienzo de la iteración 8</i>	45
<i>Ilustración 30 . Diagrama de evolución del proyecto al comienzo de la iteración 9</i>	46
<i>Ilustración 31 . Diagrama de clases del modelo del sistema</i>	50
<i>Ilustración 32 . Diagrama de clases del paquete servicios</i>	52
<i>Ilustración 33 . Diagrama de clases del paquete Interfaces REST</i>	52
<i>Ilustración 34 . Diagrama de clases del paquete DAO</i>	53
<i>Ilustración 35 . Diagrama de paquetes del backend</i>	54
<i>Ilustración 36 . Diagrama entidad-relación de la base de datos</i>	55
<i>Ilustración 37 . Diagrama de módulos y componentes del frontend</i>	56
<i>Ilustración 38 . Diagrama de secuencia de la historia de usuario “Médico crea ejercicio terapéutico”</i>	59
<i>Ilustración 39 . Diagrama de secuencia de la historia de usuario “Médico modifica ejercicio terapéutico”</i>	60
<i>Ilustración 40 . Diagrama de secuencia de la historia de usuario “Médico ve ejercicios terapéuticos”</i>	61
<i>Ilustración 41 . Diagrama de secuencia de la historia de usuario “Usuario se registra en el sistema”</i>	63

<i>Ilustración 42 . Diagrama de secuencia de la historia de usuario “Médico añade paciente”</i>	64
<i>Ilustración 43 . Diagrama de secuencia de la historia de usuario “Médico asigna una terapia a un paciente”</i>	65
<i>Ilustración 44 . Diagrama de secuencia de la historia de usuario “Médico consulta historial médico de un paciente”</i>	66
<i>Ilustración 45 . Diagrama de secuencia de la historia de usuario “Médico modifica historial clínico de un paciente”</i>	67
<i>Ilustración 46 . Diagrama de secuencia de la historia de usuario “Médico consulta terapias asignadas a un paciente”</i>	68
<i>Ilustración 47 . Diagrama de secuencia de la historia de usuario “Usuario hace login en el sistema”</i>	70
<i>Ilustración 48 . Diagrama de secuencia de la historia de usuario “Usuario envía mensaje de una terapia”</i>	71
<i>Ilustración 49 . Diagrama de secuencia de la historia de usuario “Usuario lee mensaje de una terapia”</i>	72
<i>Ilustración 50 . Diagrama de secuencia de la historia de usuario “Usuario modifica mensaje de una terapia”</i>	73
<i>Ilustración 51 . Diagrama de secuencia de la historia de usuario “Paciente consulta terapias asignadas”</i>	74
<i>Ilustración 52 . Diagrama de secuencia de la historia de usuario “Paciente realiza una terapia”</i>	75
<i>Ilustración 53 . Implementación de streaming de videos</i>	77
<i>Ilustración 54 . Diagrama de secuencia de la historia de usuario “Administrador registra un médico”</i>	78
<i>Ilustración 55 . Diagrama de secuencia de la historia de usuario “Administrador ve un médico”</i>	79
<i>Ilustración 56 . Diagrama de secuencia de la historia de usuario “Administrador modifica un médico”</i>	80
<i>Ilustración 57 . Diagrama de secuencia de la historia de usuario “Usuario configura perfil”</i>	81
<i>Ilustración 58 . Heroku</i>	90

Índice de tablas

<i>Tabla 1 . Análisis de riesgos.....</i>	<i>33</i>
<i>Tabla 2 . Iteraciones del proyecto.....</i>	<i>35</i>
<i>Tabla 3 . Historia de usuario “Médico crea ejercicio terapéutico”.....</i>	<i>59</i>
<i>Tabla 4 . Historia de usuario “Médico modifica ejercicio terapéutico”.....</i>	<i>60</i>
<i>Tabla 5 . Historia de usuario “Médico ve ejercicios terapéuticos”.....</i>	<i>61</i>
<i>Tabla 6 . Historia de usuario “Usuario se registra en el sistema”.....</i>	<i>62</i>
<i>Tabla 7 . Historia de usuario “Médico añade paciente”.....</i>	<i>64</i>
<i>Tabla 8 . Historia de usuario “Médico asigna una terapia a un paciente”.....</i>	<i>65</i>
<i>Tabla 9 . Historia de usuario “Médico consulta historial médico de un paciente”.....</i>	<i>66</i>
<i>Tabla 10 . Historia de usuario “Médico modifica historial clínico de un paciente”.....</i>	<i>67</i>
<i>Tabla 11 . Historia de usuario “Médico consulta terapias asignadas a un paciente”.....</i>	<i>68</i>
<i>Tabla 12 . Historia de usuario “Médico ve progreso de una terapia”.....</i>	<i>69</i>
<i>Tabla 13 . Historia de usuario “Usuario hace login en el sistema”.....</i>	<i>70</i>
<i>Tabla 14 . Historia de usuario “Usuario envía mensaje de una terapia”.....</i>	<i>71</i>
<i>Tabla 15 . Historia de usuario “Usuario lee mensaje de una terapia”.....</i>	<i>72</i>
<i>Tabla 16 . Historia de usuario “Usuario modifica mensaje de una terapia”.....</i>	<i>73</i>
<i>Tabla 17 . Historia de usuario “Paciente consulta terapias asignadas”.....</i>	<i>74</i>
<i>Tabla 18 . Historia de usuario “Paciente realiza una terapia”.....</i>	<i>75</i>
<i>Tabla 19 . Historia de usuario “Médico asigna un video a un ejercicio terapéutico”.....</i>	<i>76</i>
<i>Tabla 20 . Historia de usuario “Usuario ve video de ejercicio terapéutico”.....</i>	<i>77</i>
<i>Tabla 21 . Historia de usuario “Administrador registra un médico”.....</i>	<i>78</i>
<i>Tabla 22 . Historia de usuario “Administrador ve un médico”.....</i>	<i>79</i>
<i>Tabla 23 . Historia de usuario “Administrador modifica un médico”.....</i>	<i>80</i>
<i>Tabla 24 . Historia de usuario “Usuario configura perfil”.....</i>	<i>81</i>
<i>Tabla 25 . End-points accesibles por un usuario con rol médico.....</i>	<i>82</i>
<i>Tabla 26 . End-points accesibles por un usuario con rol paciente.....</i>	<i>83</i>
<i>Tabla 27 . End-points accesibles por un usuario con rol administrador.....</i>	<i>83</i>
<i>Tabla 28 . End-points accesibles por múltiples roles.....</i>	<i>84</i>
<i>Tabla 29 . Coste de personal.....</i>	<i>85</i>
<i>Tabla 30 . Coste recursos tecnológicos.....</i>	<i>86</i>
<i>Tabla 31 . Coste total.....</i>	<i>87</i>
<i>Tabla 32 . Importe final.....</i>	<i>87</i>

1. Introducción

EL presente documento constituye toda la documentación especificada en la norma UNE 157801 del Proyecto "Aplicación para la asignación, realización y seguimiento de terapias en casa"

La estructura de capítulos y apartado de este documento se ha realizado siguiendo los criterios y recomendaciones que establece la norma UNE 157801:2007 – "Criterios Generales para la elaboración de proyectos de Sistemas de Información".

Generalidades sobre este documento:

- Es el nexo de unión entre todos los documentos básicos del proyecto.
- Contiene toda la información relevante del proyecto para facilitar la lectura y comprensión.
- Su misión es justificar las soluciones adoptadas y describir totalmente y de forma unívoca el objeto del proyecto.
- Sirve como elemento de referencia común entre el ejecutor y el receptor del proyecto, limitando las obligaciones del ejecutor y garantizando los derechos del receptor.
- Se ha redactado de forma claramente comprensible, no sólo por profesionales especialistas sino por terceros, y en particular por el receptor de los entregables del proyecto. La información más detallada y técnica se incluye en otros documentos básicos y anexos de este proyecto.

Contenido de este documento

- Contiene la descripción de todos los elementos que el ejecutor deberá entregar al receptor a lo largo o al final de la ejecución del proyecto.
- Contiene las reglas que permiten identificar y gestionar los cambios que a lo largo del proyecto se puedan producir sobre el contenido del propio proyecto.
- Describe también los elementos que se deben utilizar en la ejecución para garantizar el éxito del proyecto: método, organización, validaciones, etc.

2. Objeto del proyecto

El objeto del proyecto consiste en la creación de un prototipo de aplicación empresarial destinada a la asignación, realización y seguimiento de terapias de forma remota.

Mediante la aplicación de la ingeniería del software, se busca crear un sistema que permita a los diferentes usuarios finales mejorar su experiencia y facilitar la gestión de terapias y rehabilitación, tanto por parte del especialista encargado de su asignación y seguimiento como por parte de los pacientes que la realizan.

Normalmente, la interacción entre un profesional médico y un paciente se suele limitar a citas médicas en una consulta. Estas citas en muchos casos pueden no llevarse a cabo con la periodicidad adecuada. Además, en el caso particular de profesionales como fisioterapeutas, optometristas, podólogos, etc. que trabajan con la asignación de planes de ejercicios para terapias o rehabilitación, la información correspondiente a estos puede ser difícil de recordar por parte del paciente.

Por ejemplo, se plantea el caso de un paciente que tras una lesión acude a un fisioterapeuta. En la primera cita médica, el fisioterapeuta le informa de la realización de diversos ejercicios, que deberá llevar a cabo durante un período de días determinado. El paciente puede tener problemas para recordar cómo realizar dichos ejercicios y las fechas concretas en las que debe hacerlos, mientras que el fisioterapeuta desconoce cómo evoluciona su paciente y los posibles problemas a los que se enfrenta hasta la próxima cita médica entre ambos.

El sistema final que se desarrolle tratará de buscar una solución efectiva a todos estos inconvenientes. Actuará como un nexo de unión entre especialista médico y paciente, permitiendo que en todo momento puedan acceder a la información que necesitan y llevar a cabo sus objetivos de forma efectiva.

3. Antecedentes

El proyecto planteado surge como una propuesta de TFG con el objetivo de llevar a cabo la realización de un sistema para la asignación, realización y seguimiento de terapias, que sirva como posible alternativa y especialización al diverso software ya existente diseñado con fines similares.

Para poder definir las metas finales que marcan el fin del proyecto, se ha realizado previamente un “estudio del arte”, en el que se han encontrado principalmente dos tipos predominantes de aplicaciones en materia sanitaria.

Por un lado, nos encontramos con aplicaciones, generalmente orientadas a dispositivos móviles y con una clara orientación al paciente. Estas aplicaciones actúan de forma similar a una agenda especializada en el ámbito médico, donde un paciente se encarga de anotar sus propios recordatorios, ya sea para la realización de terapias, toma de medicamentos, futuras consultas y citas médicas. La aplicación se encargará de notificar cuando alguno de los eventos que ha fijado el usuario tiene lugar. Un ejemplo concreto de este tipo de aplicaciones es “MyTherapy”, una aplicación móvil disponible para dispositivos iOS y Android. Con un rating de 4,7 en la Google Play Store, es una de las aplicaciones más completas en este ámbito ya que además ofrece versiones de la aplicación destinadas a médicos, permitiendo compartir informes y realizar asignación y seguimiento de planes de medicación por parte de estos e incluso una versión de la aplicación destinada a farmacéuticos.



Ilustración 1. Interfaz MyTherapy(I)



Ilustración 2. Interfaz MyTherapy (II)

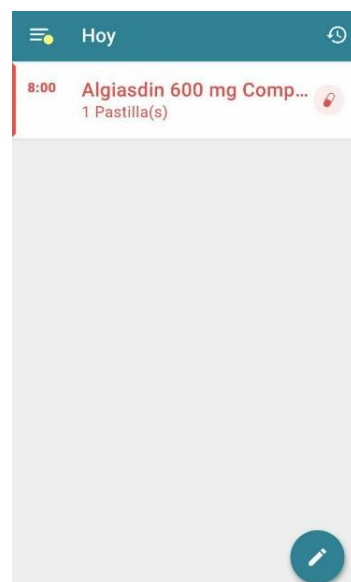


Ilustración 3. Interfaz MyTherapy(III)

Otras aplicaciones de un estilo similar que mantienen una filosofía similar a la de MyTherapy pero con un alcance menor a nivel de funcionalidad son MediSafe o MrPillster, ambas disponibles tanto en iOS como en Android.

Continuando con la diferenciación entre los tipos de aplicaciones en el foco sanitario, encontramos en el otro lado aplicaciones centrada principalmente en el ámbito de gestión de clínicas, orientadas al uso por parte de especialistas médicos.

Por lo general estos sistemas están diseñados para su uso desde clientes web donde se permite la posibilidad de registrar a los pacientes que atienden a la clínica, guardar información sobre sus historiales médicos particulares, futuras citas y llevar toda la información correspondiente a la facturación de la clínica.

En este ámbito encontramos diversas aplicaciones, algunas a destacar son Clinni, NetClinicas, Bonoom, Clinic-cloud o FISIOSALUS.

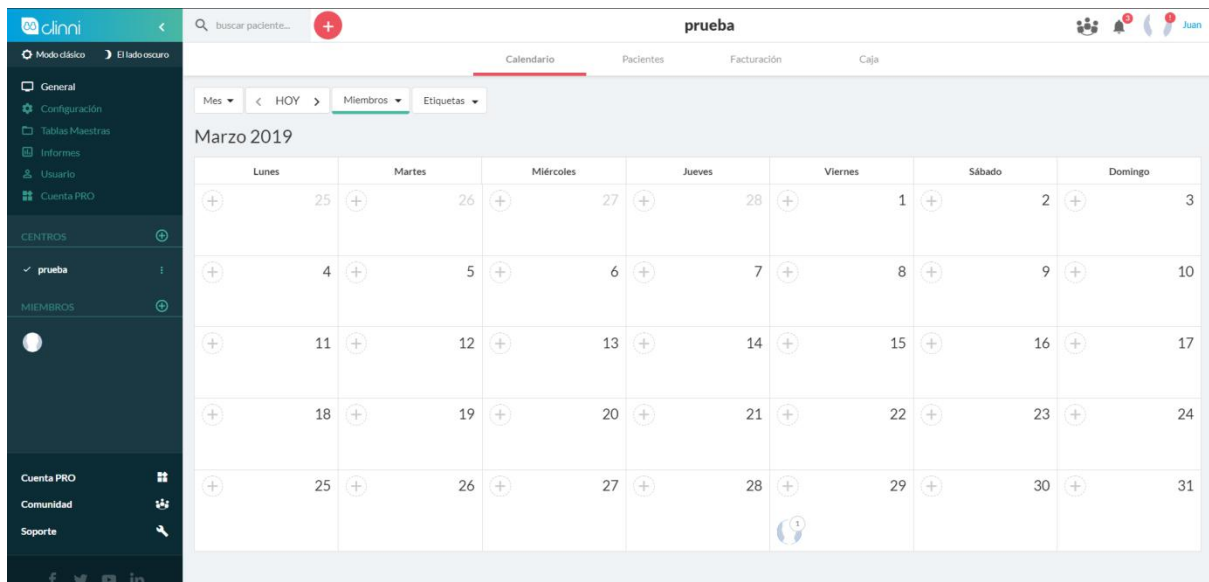


Ilustración 4. Interfaz Clinni (I)

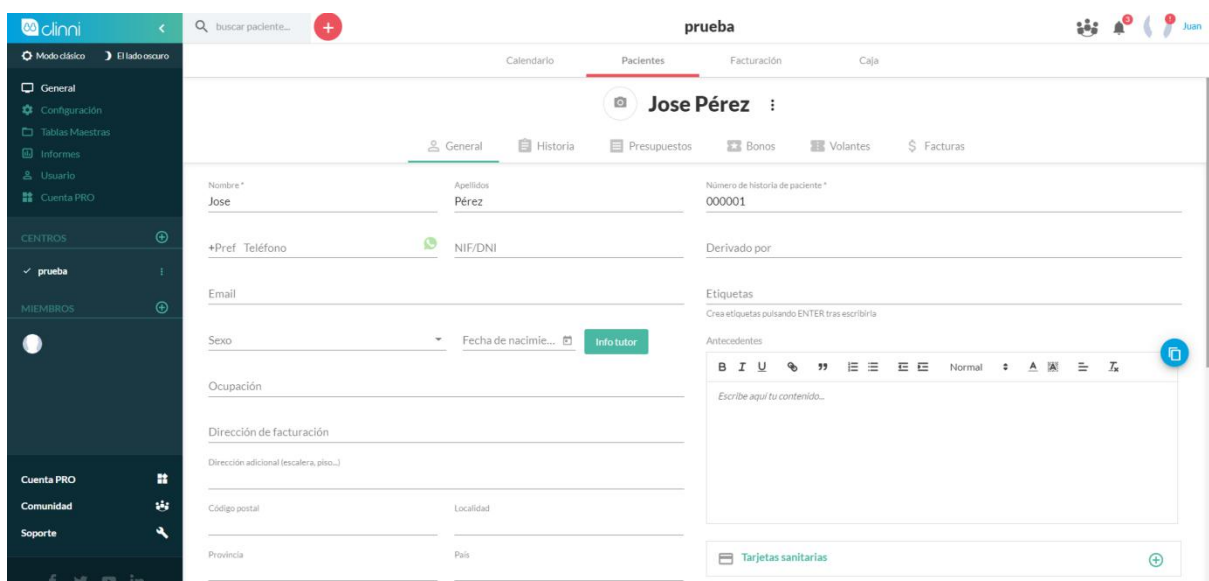


Ilustración 5. Interfaz Clinni (II)

Una vez tenemos una visión del panorama general en el tipo de sistemas que aparecen principalmente podemos analizar qué características de estos nos interesan para nuestro proyecto y ofrecer un nuevo enfoque adaptado a las necesidades particulares de este.

Debido a que la realización de ejercicios de este tipo se plantea como algo más complejo que la simple toma de medicamentos, es necesario plantear un nuevo tipo de sistema en el que tanto el rol de médico como de paciente tengan un nivel de relevancia importante e interacción mutua, de tal modo que cualquier posible problema a la hora de realizar los diferentes ejercicios terapéuticos se pueda solventar de forma rápida y eficiente, implicando a todos los participantes en la terapia.

4. Descripción de la situación actual

4.1. Descripción del entorno actual

Para la realización del proyecto partimos de un equipo compuesto por un estudiante del Grado en Ingeniería Informática y la colaboración del tutor responsable del mismo TFG. En este caso, el tutor representará la faceta de cliente adicionalmente, de tal modo que los requisitos y su implementación deberán ser aprobados en última instancia por él.

Por parte del alumno, la experiencia en la gestión de proyectos software es reducida y limitada a los conocimientos adquiridos durante la realización del grado. Además, debemos contar con unas restricciones de tiempo que vienen estimadas en las 300 horas de trabajo dedicadas que plantea la Universidad de Jaén a la realización de este mismo.

En materia de equipamiento hardware y software, se poseen todos los recursos necesarios para la realización del proyecto. En particular se trabajará con herramientas libres y gratuitas y en el caso de productos software de pago, la licencia que se utilizará vendrá suministrada por la propia Universidad de Jaén.

4.2. Resumen de las principales deficiencias identificadas

La principal deficiencia encontrada es la falta de experiencia en la gestión de un proyecto completo de software, lo cual puede suponer posibles fallos a nivel de planificación o dificultades para implementar algunas de los requisitos que se determinen para el sistema.

5. Normas y referencias

5.1. Disposiciones legales y normas aplicadas

- ISO/IEC 15504
- ISO/IEC 27000
- Ley Orgánica de Protección de Datos (LOPD)
- Reglamento General de Protección de Datos (RGPD)

5.2. Bibliografía

- Wysocki, R. (2013). *Effective project management*. Hoboken, N.J.: Wiley.
- Pohl, K. ; Rupp, C. (2015). *Requirements Engineering Fundamentals: A Study Guide for the Certified Professional for Requirements Engineering Exam - Foundation Level - IREB compliant*. Santa Barbara, CA: Rocky Nook
- Sommerville, I. (2016). *Software engineering*. Boston [etc.]: Pearson.
- Walls, C. (n.d.). *Spring in Action*.
- Keith, M. ; Schincariol, M. (2013). *Pro JPA 2*. Berkeley, CA: Apress.
- Fowler, M. (2019). *The New Methodology*. [online] martinowler.com. Disponible en: <https://www.martinfowler.com/articles/newMethodology.html>
- Project Management Institute. (2004). *A guide to the project management body of knowledge (PMBOK guide)*. Newtown Square, Pa: Project Management Institute.
- Baeldung. (2019). *REST with Spring Tutorial* | Baeldung. [online] Disponible en: <https://www.baeldung.com/rest-with-spring-series>

5.3. Métodos, Herramientas, Modelos, Métricas y Prototipos

5.3.1. Métodos y herramientas

La primera toma de decisión relevante que debemos hacer es definir el tipo de proyecto al que nos enfrentamos. El criterio a emplear se basará en la claridad del objetivo final del proyecto y la solución para cumplir dicho objetivo.

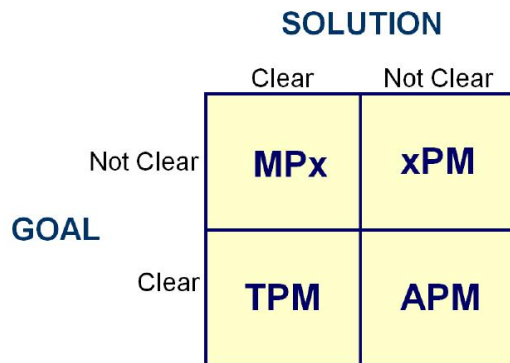


Ilustración 6. Cuadrantes de la gestión de proyectos

En nuestro caso, el objetivo se puede considerar como claramente definido. En última instancia, lo que queremos es una aplicación para la asignación, seguimiento y realización de terapias. Sin embargo, la solución para llegar a dicho objetivo mantiene cierta impredecibilidad. Los requisitos y el alcance pueden variar, además de que el equipo no posee experiencia previa en proyectos de carácter similar.

Teniendo en cuenta las características expuestas, nuestro proyecto quedaría dentro de la gestión de proyectos ágiles (APM). Dentro de esta quedan definidos dos tipos de ciclo de vida del proyecto: iterativo y adaptativo. En la siguiente imagen se pueden ver los diferentes ciclos de vida de proyectos considerados.

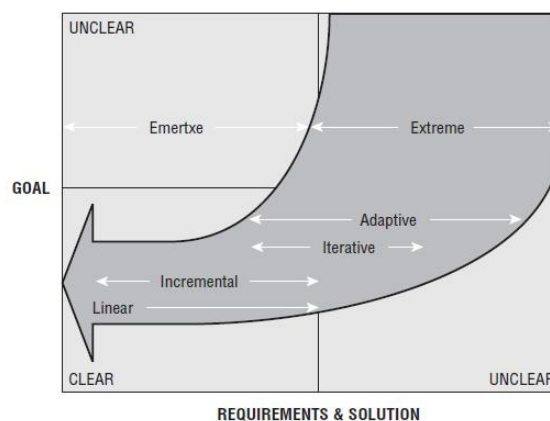


Ilustración 7. Tipos de gestión del ciclo de vida de un proyecto

El ciclo de vida adaptativo está pensado para casos en los que se desconoce en gran medida la solución, mientras que el iterativo queda planteado para casos menos severos. En este caso se utilizará un ciclo de vida iterativo, ya que no consideramos que el grado de incertidumbre de la solución sea lo suficientemente alto como para justificar el uso de un ciclo de vida adaptativo (Wysocki, 2013). En la siguiente imagen se pueden ver las diferentes etapas del ciclo de vida iterativo:

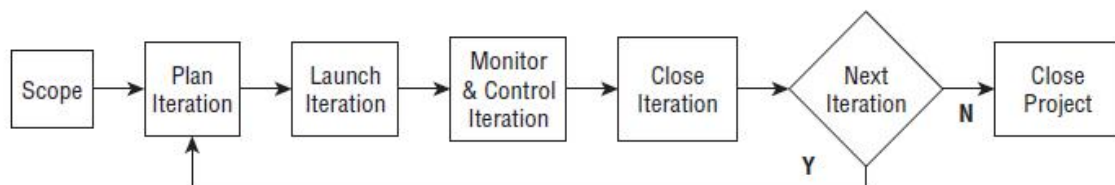


Ilustración 8. Ciclo de vida de un proyecto iterativo

Dentro de los ciclos de vida iterativos aparecen múltiples metodologías de trabajo. Algunas de ellas son las siguientes:

- Evolutionary Development Waterfall
- Scrum
- Rational Unified Process (RUP)
- Dynamic Systems Development Method (DSDM).

Nosotros hemos decidido utilizar Scrum. Se trata de la metodología más extendida dentro de la gestión ágil de proyectos y el principal motivo por el que es la elegida para este proyecto radica en que el equipo tiene más experiencia trabajando con Scrum que con el resto de metodologías ágiles planteadas.

Dentro de Scrum se trabajará con diferentes artefactos, predominando el Product Backlog y el Sprint Backlog. El primero se refiere al total de historias de usuario definidas, mientras que el segundo refleja las historias de usuario que quedan planificadas para un Sprint concreto.

La principal variación que realizaremos con respecto a Scrum serán las reuniones. En nuestro caso, para facilitar la realización de estas, sintetizaremos las diferentes reuniones que se proponen en la metodología Scrum (Sprint planning, Sprint review y Sprint retrospective) en una única reunión al principio de cada iteración que sirva tanto a nivel de planificación de dicha iteración como de análisis y validación del trabajo realizado en la iteración anterior.

5.3.2. Modelos, métricas y prototipos

Para la realización del proyecto se van a plantear diversas métricas. Al utilizar como metodología de trabajo Scrum, se utilizarán los puntos de historia como medida de estimación del esfuerzo requerido por las diferentes historias de usuario. Los puntos de historia permiten planificar el proyecto de manera menos rígida a una asignación de esfuerzo basada puramente en horas de trabajo. Concretamente se trabajará con valores dentro de la escala Fibonacci con límite en 8. La estimación de puntos de historia se hará por analogía partiendo de una historia de usuario que se utilizará como referencia para estimar el resto.

La priorización de las historias de usuario se hará utilizando la nomenclatura MoSCoW. De este modo, se establecen 4 niveles de prioridad: Must, Should, Could, Wouldn't, siendo Must el nivel de mayor prioridad y Wouldn't el de menor prioridad.

Para el cálculo de la duración del proyecto no se emplea ninguna métrica específica. De acuerdo a los fundamentos de los proyectos ágiles, la fecha final del proyecto quedará fijada desde el principio, viniendo definida por la fecha límite de entrega de Trabajos de Fin de Grado de la Universidad de Jaén para la convocatoria ordinaria II.

En materia presupuestaria, para el cálculo de coste de proyecto partiremos del salario mensual base definido en normativas legales para los diferentes roles que intervengan en el proyecto, y el número de meses que se empleen en la realización del proyecto. En la sección 19 de este documento se detallan todos los procedimientos seguidos.

Con respecto a los prototipos, al inicio del proyecto se ha realizado un prototipo del cliente de la aplicación. El objetivo de este ha sido principalmente la obtención de requisitos, ilustrar el funcionamiento inicial planteado para el sistema y el aspecto visual del cliente.

La realización de dicho prototipo ha sido realizada mediante el software para prototipado Axure RP 8. Incorpora funcionalidad básica del sistema para la navegación entre las diferentes vistas del sistema.

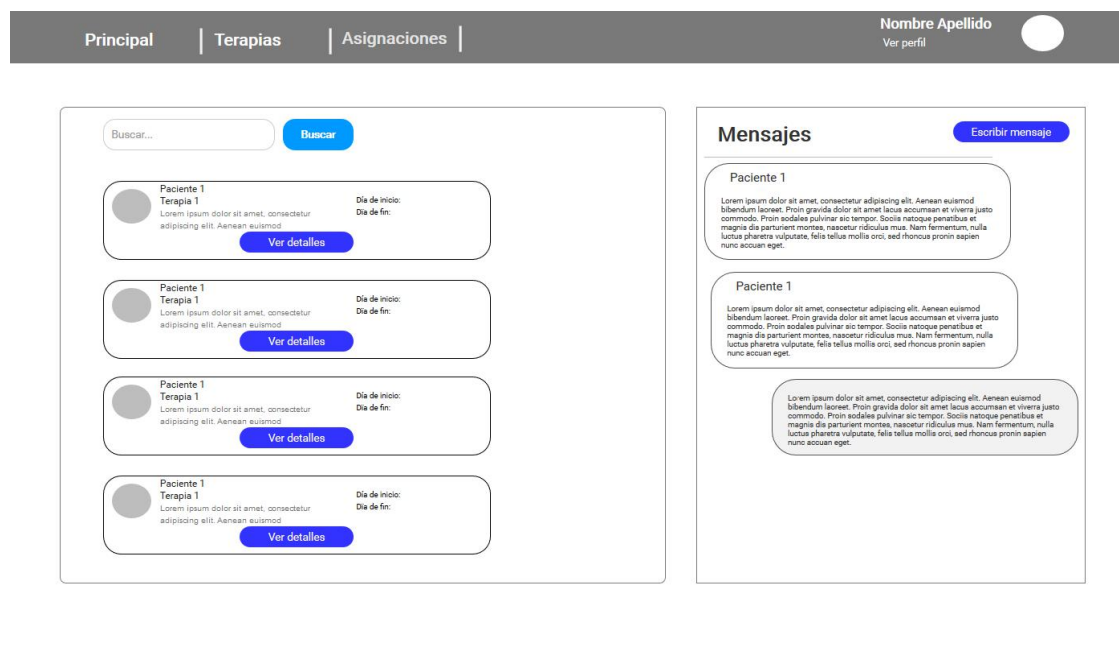


Ilustración 9. Interfaz del prototipo desarrollado (I)

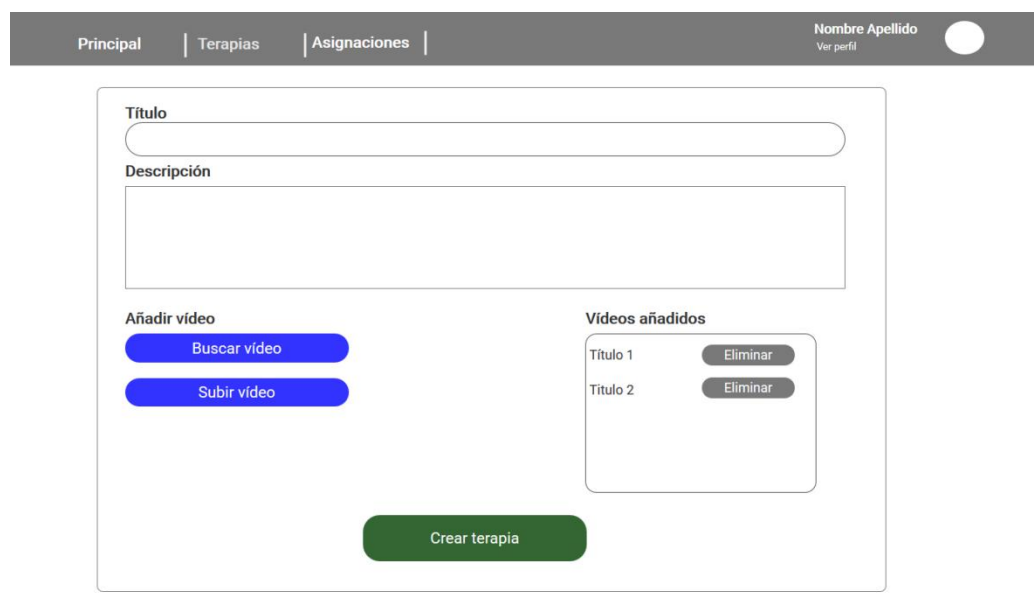


Ilustración 10. Interfaz del prototipo desarrollado (II)

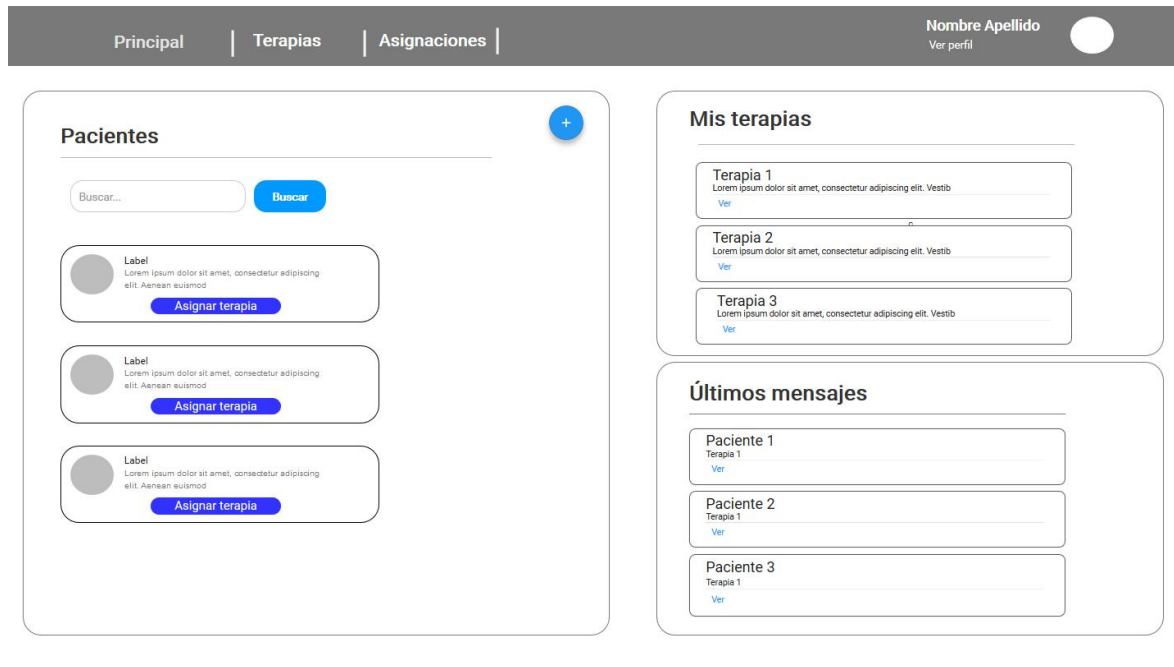


Ilustración 11. Interfaz del prototipo desarrollado (III)

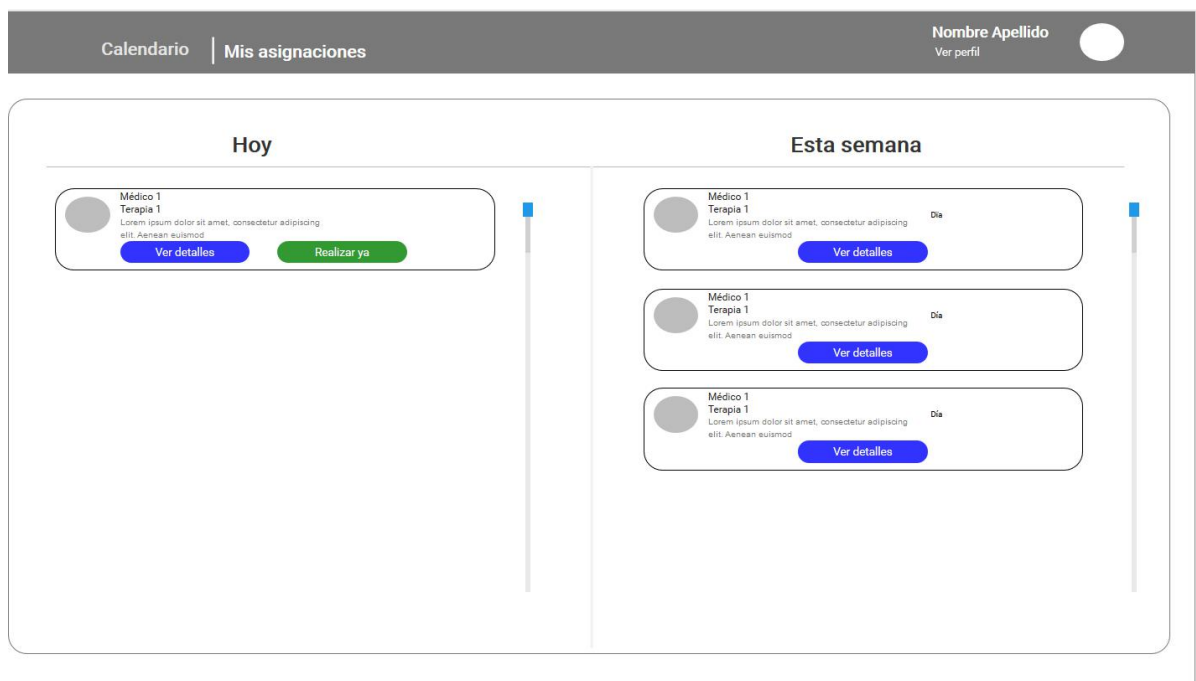


Ilustración 12. Interfaz del prototipo desarrollado (IV)

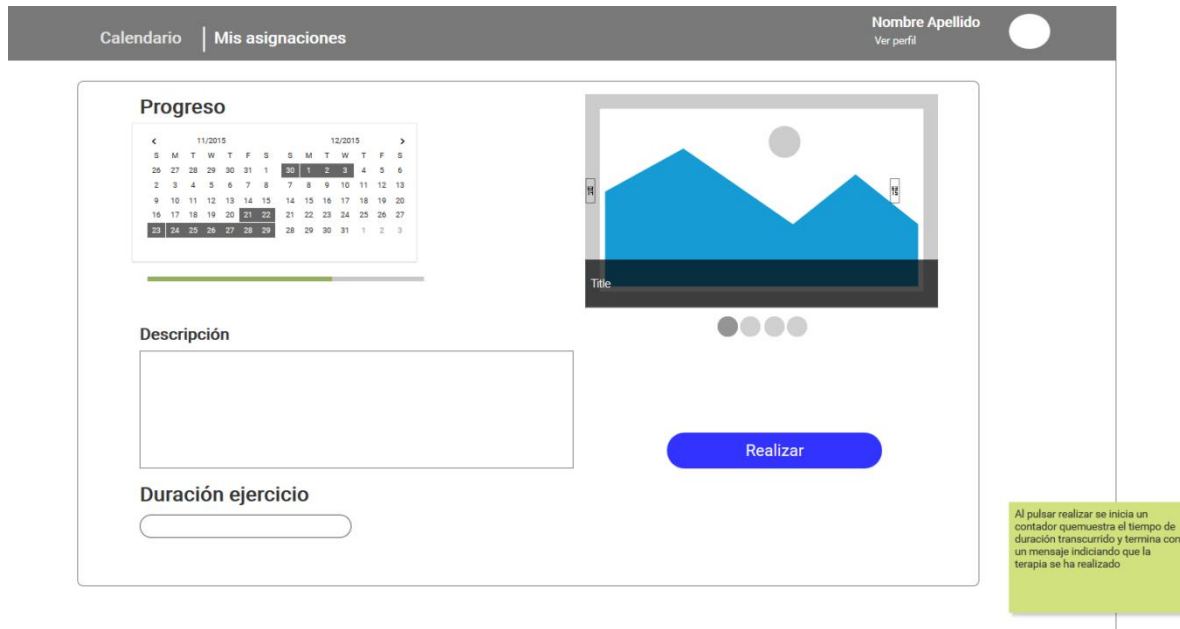


Ilustración 13. Interfaz del prototipo desarrollado (V)

5.4. Mecanismos de control de calidad aplicados durante la redacción del proyecto

La realización de la documentación del proyecto se ha realizado en base a la documentación de la norma “CCII-N2016-02 Norma Técnica para la realización de la Documentación de Proyectos en Ingeniería Informática”, tanto a un nivel estructural de los diferentes apartados que debe poseer la documentación como del contenido de cada una de dichas secciones.

Las referencias bibliográficas se han redactado en formato APA.

5.5. Otras referencias

- Spring.io. (2019). *Spring Projects*. [online] Disponible en: <https://spring.io/projects/spring-boot>
- Docs.angularjs.org. (2019). *AngularJS*. [online] Disponible en: <https://docs.angularjs.org/api>
- PostgreSQL.org. (2019). *PostgreSQL: Documentation*. [online] Disponible en: <https://www.postgresql.org/docs/>
- Docs.oracle.com. (2019). *Overview (Java SE 11 & JDK 11)*. [online] Disponible en: <https://docs.oracle.com/en/java/javase/11/docs/api/index.html>
- Devdocs.io. (2019). *DevDocs — JavaScript documentation*. [online] Disponible en: <https://devdocs.io/javascript/>
- Disposición 3156 del BOE núm. 57 de 2018. Disponible en: <https://www.boe.es/boe/dias/2018/03/06/pdfs/BOE-A-2018-3156.pdf>
- Devcenter.heroku.com. (2019). *Heroku Dev Center*. [online] Disponible en: <https://devcenter.heroku.com/>
- BootstrapMaterialDesign. (2019). *Bootstrap Material Design*. [online] Disponible en: <https://fezvrasta.github.io/bootstrap-material-design/>

6. Definiciones y abreviaturas

- **TFG:** *Trabajo de Fin de Grado.*
- **TPM:** *Traditional Project Management.* Gestión de proyectos tradicionales
- **APM:** *Agile Project Management.* Gestión de proyectos ágiles.
- **MPx:** *Emertxe Project Management.* Gestión de proyectos “emertxe” (proyectos con una solución clara y un objetivo no definido)
- **xPM:** *Extreme Project Management.* Gestión de proyectos extrema (proyectos que no tienen ni una solución ni un objetivo definido)
- **JPA:** *Java Persistence API.*
- **ORM:** mapeado objeto-relacional.
- **EHCaché:** caché de Java de código abierto.
- **Spring:** framework para el desarrollo de aplicaciones basado en Java.
- **Java:** lenguaje de programación de propósito general orientado a objetos.
- **AngularJS:** framework de desarrollo de aplicaciones web basado en Javascript
- **Javascript:** lenguaje de programación interpretado orientado a objetos.
- **Backend:** capa de acceso a datos del sistema
- **Frontend:** capa de presentación del sistema
- **SPA:** *Single Page Applications.* Aplicaciones que trabajan en solo una página.
- **INCIBE:** *Instituto Nacional de Ciberseguridad*

7. Requisitos iniciales

Los requisitos iniciales quedan definidos desde las siguientes perspectivas: requisitos funcionales, requisitos no funcionales o de calidad y restricciones.

Debido a las características del proyecto, la técnica de adquisición de requisitos por la que hemos apostado ha sido principalmente prototipado y brainstorming.

Idealmente, los requisitos funcionales del rol médico se podrían haber complementado con una entrevista a un profesional del sector, pero como esta situación es compleja de llevar a cabo hemos tenido que descartarla y tratar de inferir dichos requisitos a partir de las técnicas mencionadas previamente y documentación externa encontrada en la web y otras aplicaciones del sector.

De cara al paciente si que se han podido obtener los requisitos en base a nuestra propia experiencia. Concretamente, el tutor del proyecto ha seguido diariamente una terapia visual indicada por un especialista en optometría. Muchos de los requerimientos que se listan a continuación se han podido obtener a su propia experiencia personal durante el último año.

Por lo tanto los requisitos definidos al inicio del proyecto son los siguientes:

- Requisitos funcionales para el rol paciente:
 - o El sistema permitirá al paciente ver las terapias que tiene asignadas.
 - o El sistema permitirá al paciente realizar una terapia que tiene asignada.
 - o El sistema mostrará toda la información fundamental necesaria para la correcta realización de una terapia al paciente.
 - o El sistema permitirá al paciente comunicarse con un médico que le haya asignado una terapia mediante un sistema de mensajería interno.
 - o El sistema permitirá al usuario ver un histórico de todas las terapias que se le han asignado desde su registro en la aplicación.
 - o El paciente podrá ver su perfil de usuario y modificarlo.
- Requisitos funcionales para el rol médico:
 - o El sistema ofrecerá al médico un listado con todos sus pacientes.
 - o El sistema permitirá al médico añadir nuevos pacientes a su listado.
 - o El sistema permitirá al médico crear terapias
 - o El sistema permitirá asignar videos a las terapias, los cuales ya podrán existir ya en la aplicación o subir sus propios videos.
 - o El sistema permitirá al médico ver el progreso de las asignaciones realizadas a pacientes que estén en desarrollo.

- o El sistema permitirá al médico el envío de mensajes a un paciente.
 - o El médico podrá ver su perfil y modificarlo
- Requisitos de calidad:
 - o El sistema ofrecerá a los clientes diferentes formas de acceder a los servicios ofrecidos desde diferentes dispositivos
 - o El sistema ofrecerá una interfaz de usuario intuitiva y fácil de utilizar
 - o El sistema protegerá los datos de los usuarios que así lo requieran según la normativa de protección de datos
 - o El sistema debe de estar preparado para facilitar la escalabilidad de este y la posibilidad de incluir nuevas funcionalidades en el futuro
- Restricciones:
 - o El esfuerzo para la realización del sistema no deberá exceder las 300 horas de trabajo.

8. Alcance

Para la estimación del alcance es fundamental tener en cuenta la metodología de trabajo que se va a emplear para el proyecto. En nuestro caso, estamos trabajando con Scrum, una metodología ágil.

Una de las principales características de dichos proyectos es lo que se conoce como inversión del triángulo del proyecto con respecto a las metodologías tradicionales.

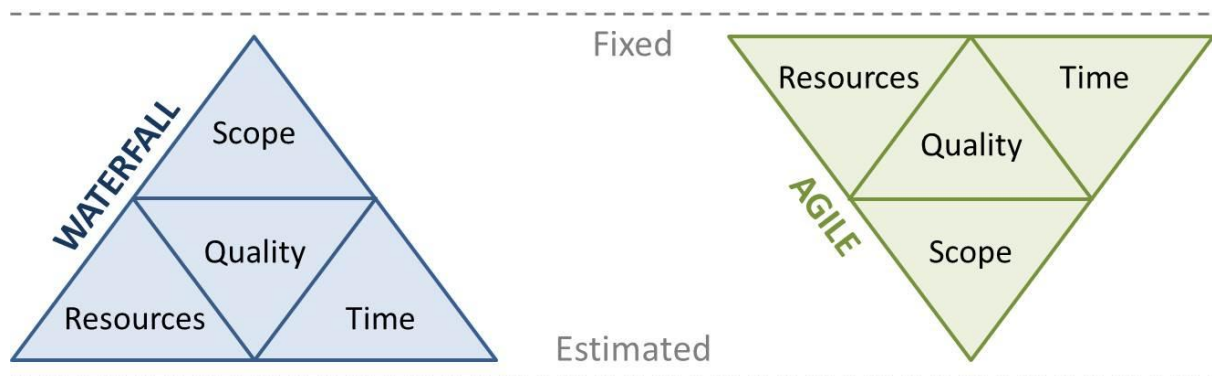


Ilustración 14. Triángulo del proyecto en gestión de proyectos tradicionales y ágiles

En las metodologías ágiles, los recursos para la realización del proyecto y la fecha final de este quedan fijadas, siendo el alcance lo que varía.

En primera instancia, el alcance quedará determinado por los requisitos iniciales del proyecto y las historias de usuario negociadas a partir de ellos. Sin embargo, es probable que dichos requisitos varíen, aparezcan nuevos y desaparezcan otros, por lo que el alcance también variará.

En la conclusión del proyecto, se deberá de entregar un entregable final con las funcionalidades que se hayan podido implementar dentro de las restricciones de tiempo y recursos especificadas, además de la presente documentación

A lo largo del proyecto, en cada iteración se presentará un entregable parcial e incremental del sistema desarrollado hasta el momento. Adicionalmente, se definirán tres hitos principales dentro del transcurso del proyecto: el primero, marcará la conclusión del desarrollo de las funcionalidades del rol médico. El segundo, el de las funcionalidades del rol paciente. Y el tercero, coincidirá con el final del proyecto. Será importante tener en cuenta que estos hitos tendrán una función netamente orientativa, y que en cualquier momento se podrá trabajar en la

implementación de diferentes historias de usuario para diferentes roles si así lo requiere el cliente.

Dentro de los entregables del proyecto también se incluirá un prototipo del cliente con funcionalidad limitada que se desarrollará durante la primera iteración del proyecto.

9. Hipótesis y restricciones

En esta sección se han recogido las hipótesis y restricciones referidas al proyecto agrupadas por categoría:

- Tecnológicas:
 - El sistema se adaptará de forma adecuada a los principales navegadores web.
 - El sistema se deberá de ajustar a los requisitos indicados en la documentación de entrada del trabajo de fin de grado. Es decir, deberá ser en última instancia una aplicación empresarial con el backend siendo desarrollado utilizando Java junto al framework Spring y se creará un cliente para acceder a la aplicación.
- Entorno de trabajo:
 - Los integrantes del equipo permanecerán plenamente disponibles durante el transcurso del proyecto
 - La duración total del trabajo queda estimada en 300 horas, siendo esta cifra la dada por los organismos institucionales de la Universidad para la completitud de un trabajo de fin de grado.
 - El proyecto concluirá antes de la fecha final establecida por la Universidad para la entrega de Trabajos de Fin de Grado.
 - Las herramientas y licencias utilizadas para trabajar permanecerán plenamente disponibles y funcionarán de forma adecuada durante el transcurso del proyecto

10. Estudio de alternativas y viabilidad

A continuación se proceden a enumerar todas las alternativas estudiadas que han surgido a lo largo del proyecto.

- Elección del cliente a desarrollar: a la hora de decidir el desarrollo del cliente que accederá a los servicios proporcionados por el frontend, la decisión ha venido dada principalmente por las restricciones temporales, de recursos humanos y la adecuación del cliente al mayor número posible de usuarios de la aplicación. Principalmente se han considerado dos opciones, la creación de un cliente web o la de una aplicación para dispositivos móviles. Desde el punto de vista temporal, la realización de ambos se plantea como algo complicado de llevar a cabo en los márgenes definidos para la realización del proyecto, por lo que la elección de uno de estos era algo que debía quedar definido desde el comienzo del proyecto. Finalmente se ha apostado por un cliente web que se ajuste a los principios “responsive” y “mobile-first”, permitiendo un funcionamiento aceptable en dispositivos móviles también. El cliente web es más polivalente, cubriendo todo tipo de dispositivos que tengan acceso a un navegador web y asegurando que el rol del usuario médico, el cual deberá de escribir el suficiente texto como para justificar el uso de la aplicación desde un ordenador con teclado. Otra alternativa planteada fue el uso de tecnologías como Cordova, que permiten exportar el código desarrollado en el cliente web a una aplicación móvil, aunque por falta de experiencia con dicha tecnología se descartó de forma temporal.
- Elección del lenguaje de programación y framework para la realización del cliente web: en este sentido la decisión ha venido dada principalmente por el factor experiencia del equipo de desarrollo. De toda la gran variedad de tecnologías en este campo, se ha apostado por el uso de la última versión de AngularJS, sobre el cual ya se tienen conocimientos básicos y por lo tanto, supone una reducción en el periodo de aprendizaje. Además, al ser un framework bien documentado y de uso extendido también se reduce la probabilidad de encontrarnos ante posibles problemas de implementación.
- Elección del sistema de gestión de base de datos: como tipo de base de datos se utilizará un modelo relacional. Si bien es cierto que un sistema No SQL posee múltiples ventajas en aspectos como la flexibilidad o escalabilidad, las características de nuestro sistema no presentan una dependencia vital en dichos factores como para justificar la transición hacia un sistema no relacional. Además, en un principio no se considera que el volumen de datos sea excesivamente grande. En cuanto a la herramienta concreta que utilizaremos como sistema de gestión de base de datos, en el mercado hay

una gran cantidad de alternativas. En un principio se planteó utilizar Derby, un sistema más ligero y simple, como base de datos en la fase de desarrollo y una vez entrada la fase de producción, hacer una transición a PostgreSQL, aunque finalmente se ha empleado PostgreSQL en todas las facetas del proyecto.

11. Descripción de la solución propuesta

Desde un punto de vista tecnológico, el sistema se caracterizará por contar con un con una arquitectura orientada a servicios.

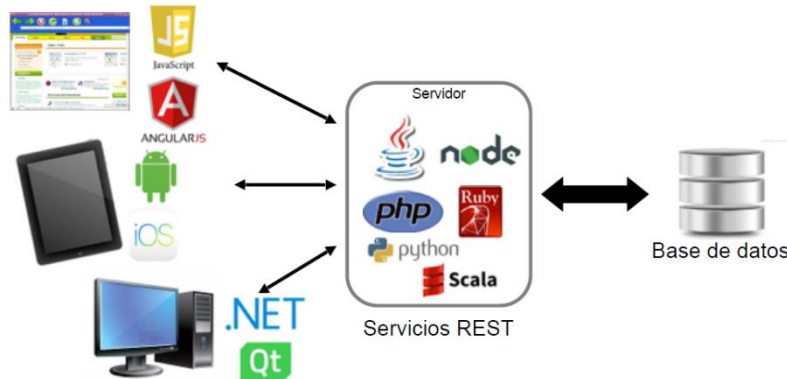


Ilustración 15. Representación de una aplicación empresarial

El backend tendrá una arquitectura basada en capas:

- Capa de servicios: hace accesible las funcionalidades del sistema desde los clientes. Se utilizará la filosofía Resftul para su implementación.
- Capa de lógica de negocio: implementa las reglas de funcionamiento del sistema mediante objetos de negocio.
- Capa de persistencia: gestiona el acceso a la base de datos. En nuestro caso utilizaremos JPA y ORM. Además, contará con un sistema de cacheado para mejorar el rendimiento, utilizando EHCache.

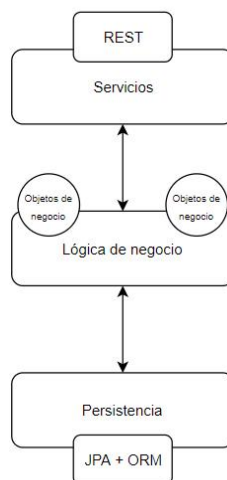


Ilustración 16. Arquitectura basada en capas

Para la implementación del backend utilizaremos Java junto al framework Spring.



Ilustración 17. Java



Ilustración 18. Spring

Desde el frontend se podrá acceder a los distintos servicios proporcionados, desde una aplicación de tipo SPA. La comunicación entre cliente y servidor se realizará mediante el protocolo HTTP y utilizando principalmente la representación JSON para la transferencia de datos entre ambos.



Ilustración 19. AngularJS

A nivel funcional, usuario y médico representarán dos roles diferentes, cada uno accediendo a sus servicios correspondientes, que representarán las diferentes historias de usuario definidas y adaptadas a lo largo del proyecto en las iteraciones de este.

En la aplicación se trabajará principalmente con dos elementos, ejercicios terapéuticos y terapias. Los ejercicios terapéuticos representarán las diferentes actividades que deberá hacer un paciente. Una terapia contará con una serie de ejercicios terapéuticos, un calendario con las fechas en las que se deben realizar dichos ejercicios terapéuticos, el tiempo que deberá realizar cada ejercicio el paciente y una sección de información adicional donde explicar de forma concreta algunas de las particularidades de la terapia.

El usuario del rol médico tiene la posibilidad de gestionar estas entidades y asignar terapias a los pacientes. Los pacientes podrán realizar terapias, de tal forma que cada terapia refleje de forma actualizada el estado de esta y el progreso que está

haciendo el paciente. Adicionalmente cada terapia tendrá una sección de chat, donde paciente y médico pueden comentar información sobre la terapia y preguntar y responder dudas.

12. Análisis de riesgos

Para el análisis de riesgos y su priorización se ha seguido un enfoque cualitativo, utilizando la matriz de probabilidad e impacto del INCIBE como referencia.

		IMPACTO		
		Bajo	Medio	Alto
PROBABILIDAD	Baja	Muy bajo	Bajo	Medio
	Media	Bajo	Medio	Alto
	Alta	Medio	Alto	Muy alto

Ilustración 20. Matriz de probabilidad e impacto

Para documentar los riesgos se ha seguido un enfoque basado en plantillas. Los puntos que se definirán para cada riesgo son:

- Riesgo: identificación y breve descripción significativa de este.
- Categoría a la que pertenece: se han tomado las siguientes categorías (Sommerville, 2016)
 - Riesgo tecnológico
 - Riesgo personal
 - Riesgo organizacional
 - Riesgo de herramientas
 - Riesgo de requisitos
 - Riesgo de estimación
- Probabilidad de que ocurra el riesgo
- Impacto del riesgo en el proyecto
- Puntuación, obtenida a partir de los valores de probabilidad e impacto asignados al riesgo y el resultado obtenido en función de estos a partir de la matriz de probabilidad e impacto de INCIBE
- Ranking, mide la relevancia del riesgo en comparación con el resto de riesgos potenciales considerados.
- Respuesta al riesgo, representa el modo a actuar en caso de que tenga lugar el riesgo especificado.

Identificación del riesgo		Clasificación cualitativa				Respuesta al riesgo
Riesgo	Categoría	Probabilidad	Impacto	Puntuación	Ranking	
Baja por enfermedad	Riesgo de personal	Baja	Medio	Baja	4º	Realizar una replanificación del proyecto a corto y medio plazo, dependiendo el número de cambios en función de la gravedad de la enfermedad, priorizando siempre las historias de usuario de mayor relevancia para el cliente.
Cambio en los requisitos iniciales	Riesgo de requisitos	Media	Medio	Media	1º	Renegociación con el cliente de las historias de usuario, replanificación de las historias de usuario tanto antiguas como nuevas para poder ajustarlas a las diferentes iteraciones y manteniendo siempre en cuenta la fecha límite del final del proyecto.
Incapacidad de implementar de forma exitosa alguna de las historias de usuario especificadas	Riesgo de requisitos	Baja	Alto	Media	1º	Búsqueda de una posible alternativa con el cliente que resulte más factible a nivel de implementación.
Incapacidad de integrar en el cliente componentes auxiliares	Riesgo de herramientas	Baja	Medio	Media	1º	Búsqueda de otros componentes con características similares, en caso de que esto no sea posible, se planteará con el cliente la búsqueda de una alternativa o la implementación manual de dichos componentes.
El rendimiento de la base de datos no cumple con los niveles de calidad esperados	Riesgo de herramientas	Baja	Bajo	Muy baja	5º	Búsqueda de otro sistema de gestión de base de datos que se adecue de mejor manera al sistema

Tabla 1. Análisis de riesgos

13. Organización y gestión del proyecto

13.1. Organización

13.1.1. Actores del proyecto y relaciones entre los mismos

El proyecto involucra a dos actores, el alumno que realiza el proyecto de TFG y el tutor del mismo.

13.1.2. Roles y responsabilidades

Los roles que se consideran de cara al proyecto se van a definir en base a los fundamentos de Scrum. De acuerdo a dicha metodología, en el conjunto de encargados de la creación del producto final intervienen los siguientes roles:

- Product owner (responsable del producto): este rol vendrá asumido por el tutor. Se encargará de la priorización de historias de usuario y la validación o rechazo de los resultados obtenidos y de forma general, representar las necesidades y deseos del cliente en relación con el producto.
- Scrum Master: este rol vendrá asumido por el alumno. Se encargará de la gestión del proyecto, la supervisión del seguimiento de los principios de Scrum y facilitar el trabajo de los miembros del equipo de desarrollo.
- Equipo de desarrollo: debido a la naturaleza del proyecto, contará con un solo miembro, el alumno, que se encargará de cumplir las responsabilidades de analista, diseñador, desarrollador y tester.
- Cliente: rol asumido por el tutor. Representa a los diferentes stakeholders del proyecto.

14- Planificación temporal

14.1 - Evolución del plan de proyecto

En el proyecto quedan inicialmente definidas como fecha de inicio el 4/2/2019 y como fecha de fin el 17/6/2019. Al seguir una metodología Scrum, se han definido 10 iteraciones, que comienzan en la iteración 0 y terminan en la iteración 9, con una duración de 2 semanas, excepto para el caso de la iteración 9 cuya duración será de 7 días y su objetivo sería gestionar los últimos detalles de la conclusión del proyecto.

En la tabla siguiente se muestran tanto la fecha de inicio como de fin prevista para cada una de las iteraciones.

Iteración	Fecha Inicio	Fecha Fin
Iteración 0	4/2/2019	18/2/2019
Iteración 1	19/2/2019	4/3/2019
Iteración 2	5/3/2019	18/3/2019
Iteración 3	19/3/2019	1/4/2019
Iteración 4	2/4/2019	15/4/2019
Iteración 5	16/4/2019	29/4/2019
Iteración 6	30/4/2019	13/5/2019
Iteración 7	14/5/2019	27/5/2019
Iteración 8	28/5/2019	10/6/2019
Iteración 9	11/6/2019	17/6/2019

Tabla 2. Iteraciones del proyecto

Al inicio del proyecto se han estimado un total de 19 historias de usuario, que suman un total de 53 puntos de historia.

La implementación de las historias de usuario deberá de llevarse a cabo en el periodo existente entre la iteración 1 y 9, incluidas estas. La iteración 0 no será considerada debido a que tiene un papel de adquisición de requisitos y planificación del proyecto. Por lo tanto tendremos 9 iteraciones.

La velocidad inicial prevista vendrá determinada por la división del total de puntos de historia entre el número de iteraciones. El valor de puntos de historia se ha obtenido mediante la suma de los puntos de historia de cada una de las historias de usuario diseñadas al inicio del proyecto.

$$53 / 9 = 5,89$$

Este valor lo hemos redondeado, estimando por lo tanto la velocidad del proyecto en 6.

El diagrama de evolución inicial esperado sería el siguiente.

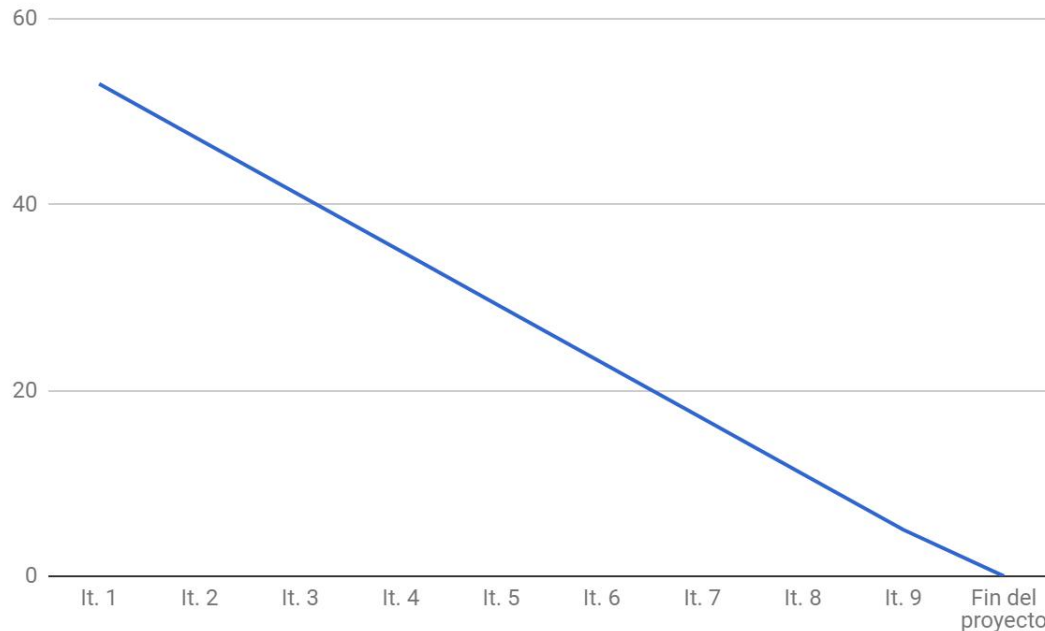


Ilustración 21. Diagrama de evolución inicial

A continuación se procede a describir el trabajo realizado en cada una de las iteraciones. Las historias de usuario quedarán detalladas en la sección 18 “Especificaciones del sistema” de este documento. En este apartado se hará referencia a las distintas historias de usuario de acuerdo a su identificador y título.

Para cada iteración se detallarán las historias de usuario en las que se trabaja durante dicha iteración, lo que se conoce como sprint backlog. Al inicio de cada iteración también se mostrará un diagrama de evolución con los puntos de historia realizados hasta el momento de comenzar dicha iteración.

Adicionalmente, se recogerán los principales detalles de la reunión mantenida al inicio de cada iteración.

14.1.1 - Iteración 0

La iteración comienza con una reunión con el product owner para concretar los objetivos del proyecto. Se plantean los requisitos del sistema y se trabaja en labores de documentación. Para el final de la iteración se estipula la creación del prototipo visto en la sección [5.3.2. Modelos, métricas y prototipos](#). Además se trabajará en el diseño de casos de uso a partir de los requisitos iniciales negociados y en el diseño de las historias de usuario con las que se trabajará a lo largo del proyecto.

14.1.2 - Iteración 1

En esta iteración tiene lugar una reunión con el product owner para mostrar los resultados del prototipo realizado, sobre el que se realizan pequeños ajustes. También se presentan una serie de casos de uso obtenidos a partir de los requisitos iniciales y una primera versión de las historias de usuario que compondrán el proyecto.

Sobre las historias de usuario se realiza un proceso de negociación que concluye con el descarte de alguna de las historias de usuario y la introducción de otras nuevas.

El trabajo a realizar durante esta iteración será la implementación de las historias de usuario:

- Médico crea ejercicio terapéutico, con ID = 1 ([18.1.1 Médico crea ejercicio terapéutico](#))
- Médico modifica ejercicio terapéutico, con ID = 2 ([18.1.2. Médico modifica ejercicio terapéutico](#))
- Médico ve ejercicios terapéuticos, con ID = 3. ([18.1.3. Médico ve ejercicios terapéuticos](#))

14.1.3 - Iteración 2

La reunión con el product owner da como resultado la validación de las historias de usuario implementadas en la iteración 1.

Una vez validado el trabajo realizado, se procede a definir las historias de usuario que compondrán el sprint backlog de esta iteración:

- Usuario se registra en el sistema, con ID = 4 ([18.1.4. Usuario se registra en el sistema](#))
- Médico añade paciente, con ID = 5 ([18.1.5. Médico añade paciente](#))
- Médico asigna una terapia a un paciente, con ID = 6 ([18.1.6. Médico asigna una terapia a un paciente](#))

El gráfico de evolución del proyecto al inicio de esta iteración es el siguiente:

Planificado y Real

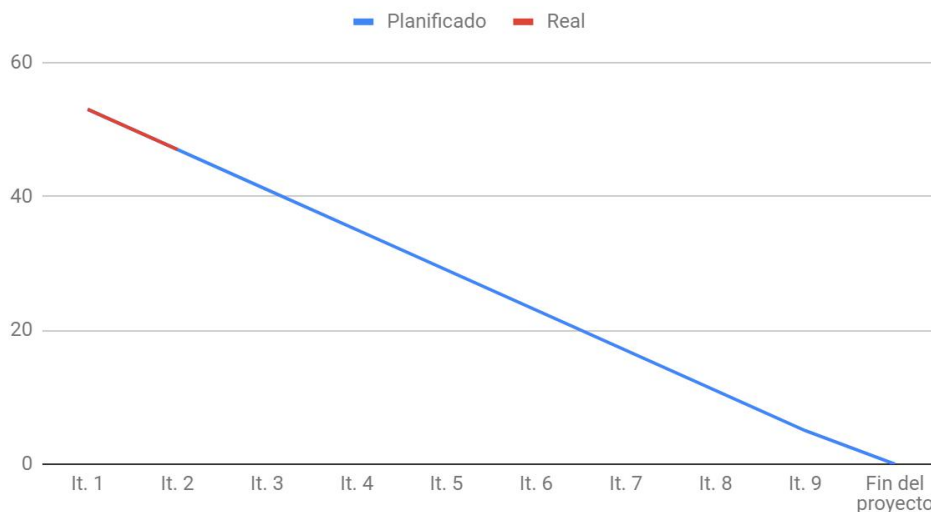


Ilustración 22. Diagrama de evolución del proyecto al comienzo de la iteración 2

14.1.4 - Iteración 3

En la reunión con el cliente al inicio de la iteración 3 se procede a revisar los resultados obtenidos hasta el momento. Con respecto a la historia “Usuario se registra en el sistema” surgen nuevos planteamientos, ya que en el momento actual de esta iteración los usuarios entran de forma libre al sistema, creando la cuenta por sí mismos. Sin embargo, se llega a la conclusión de que lo mejor para el sistema es que sea de acceso restringido. Será necesario un nuevo rol en el sistema, el de administrador, que se encargará de dar de alta a los especialistas médicos en el sistema. Por otro lado, la creación de pacientes pasará a ser responsabilidad del médico.

Las historias de usuario en las que se trabajará serán por tanto:

- Administrador registra un médico, con ID = 19 ([18.1.19. Administrador registra un médico](#))
- Administrador ve un médico, con ID = 20 ([18.1.20. Administrador ve un médico](#))
- Administrador modifica médico, con ID = 21 ([18.1.21. Administrador modifica un médico](#))
- Usuario se registra en el sistema, con ID = 4, ([18.1.4. Usuario se registra en el sistema](#))
- Médico añade paciente, con ID = 5 ([18.1.5. Médico añade paciente](#))

El gráfico de evolución que muestra el estado actual del proyecto es el siguiente:

Planificado y Real

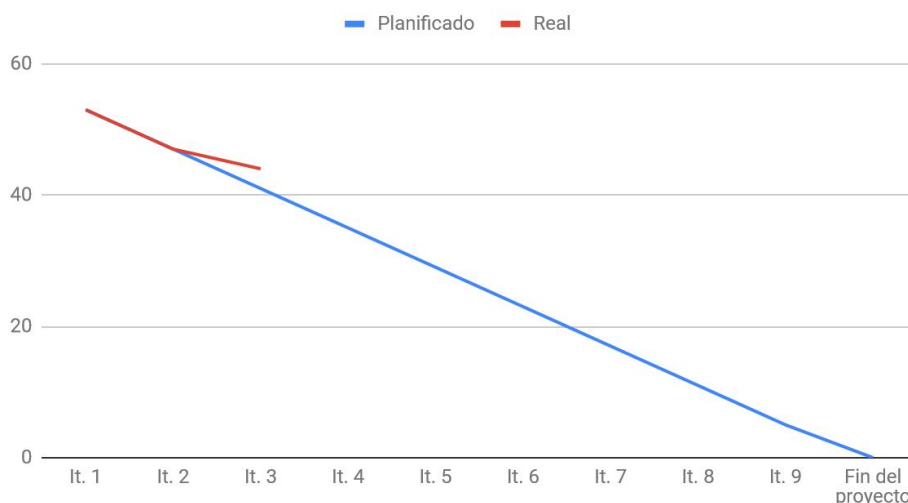


Ilustración 23. Diagrama de evolución del proyecto al comienzo de la iteración 3

Como se puede apreciar en el gráfico, con respecto a la planificación inicial nos encontramos con un retraso al haber sido rechazadas las historias de usuario con ID = 4 e ID = 5. Además, debemos de tener en cuenta de que se han añadido 6 puntos

de historia de adicionales al total. Esto supondría que a los 44 puntos de historia que nos restan para completar todas las historias de usuario pasarían a ser 50.

En este momento, debemos de replantearnos la velocidad del proyecto. Si dividimos los 50 puntos de historia entre las 7 iteraciones que faltan tendríamos:

$$50 / 7 = 7,14$$

Este valor de 7,14 lo estimamos a la alza, por lo que la velocidad nueva del proyecto será de 8. Esto supone un aumento de la velocidad de 2 puntos, por lo que deberemos de replantear durante las próximas iteraciones si se puede mantener esta velocidad o tendremos que recortar el alcance del proyecto eliminando alguna historia de usuario de menor prioridad.

Teniendo en cuenta esta nueva velocidad, el gráfico de evolución del proyecto estimado planeado al principio pasará a ser este:

Planificado

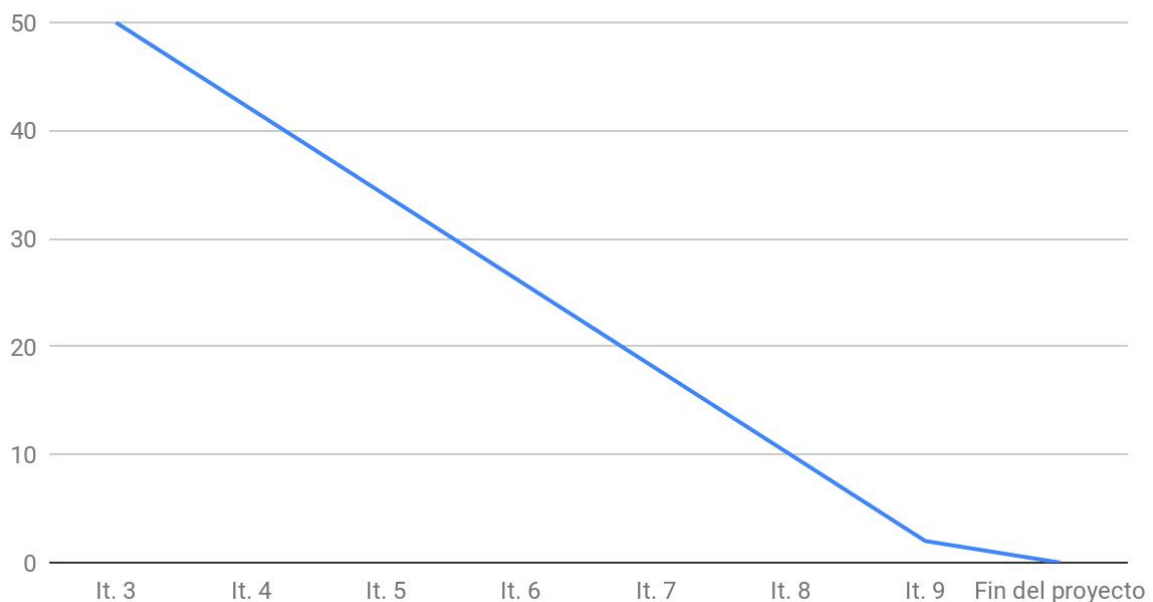


Ilustración 24. Diagrama de evolución del proyecto replanificado

14.1.5 - Iteración 4

Los cambios de la iteración 3 son validados en la reunión con el cliente y aceptados. El estado del proyecto sería el siguiente, continuando con la nueva planificación estimado en la iteración anterior:

Planificado y Real

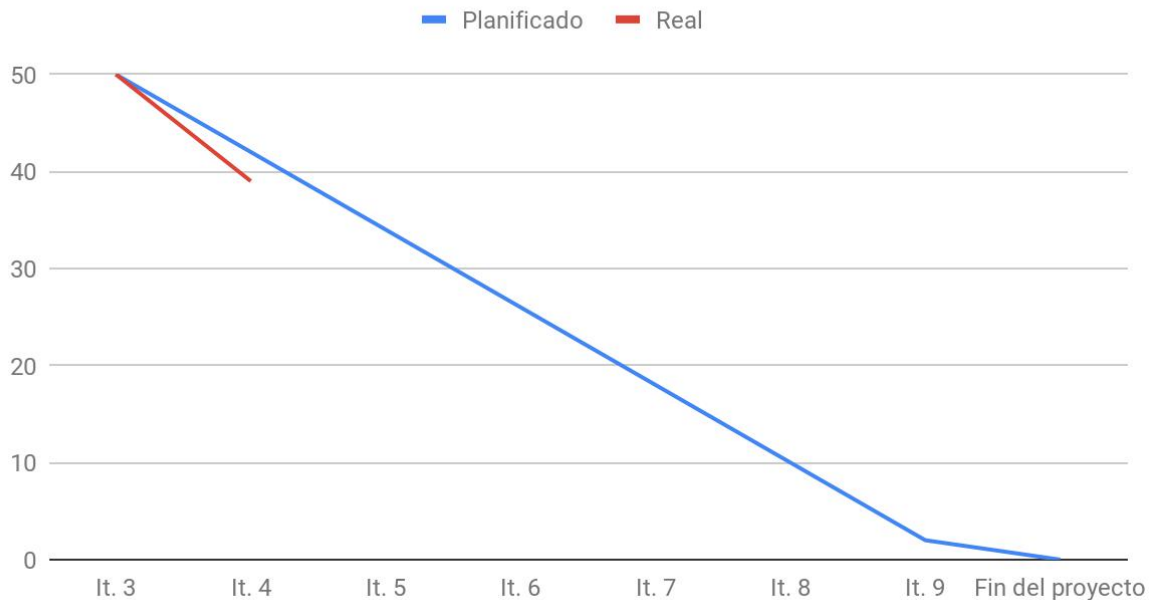


Ilustración 25. Diagrama de evolución del proyecto al comienzo de la iteración 4

La velocidad en la iteración anterior fue bastante alta. Esto se puede deber en parte a que para dos de las historias de usuario que se planificaron (ID = 4, ID = 5), al ser historias previamente rechazadas pero con una base de código existente, se ha podido utilizar parte de dicho código.

Por otro lado, las historias de usuario en las que se trabajará esta iteración 4 serán:

- Médico consulta historial médico, con ID = 7 ([18.1.7. Médico consulta historial médico de un paciente](#))
- Médico modifica historial médico, con ID = 8 ([18.1.8. Médico modifica historial clínico de un paciente](#))
- Médico consulta terapias asignadas a un paciente, con ID = 9 ([18.1.9. Médico consulta terapias asignadas a un paciente](#))
- Usuario hace login en el sistema, con ID = 11 ([18.1.11. Usuario hace login en el sistema](#))

14.1.6 - Iteración 5

Las historias de la iteración 4 son aceptadas. El diagrama de evolución hasta el momento es el siguiente:

Planificado y Real

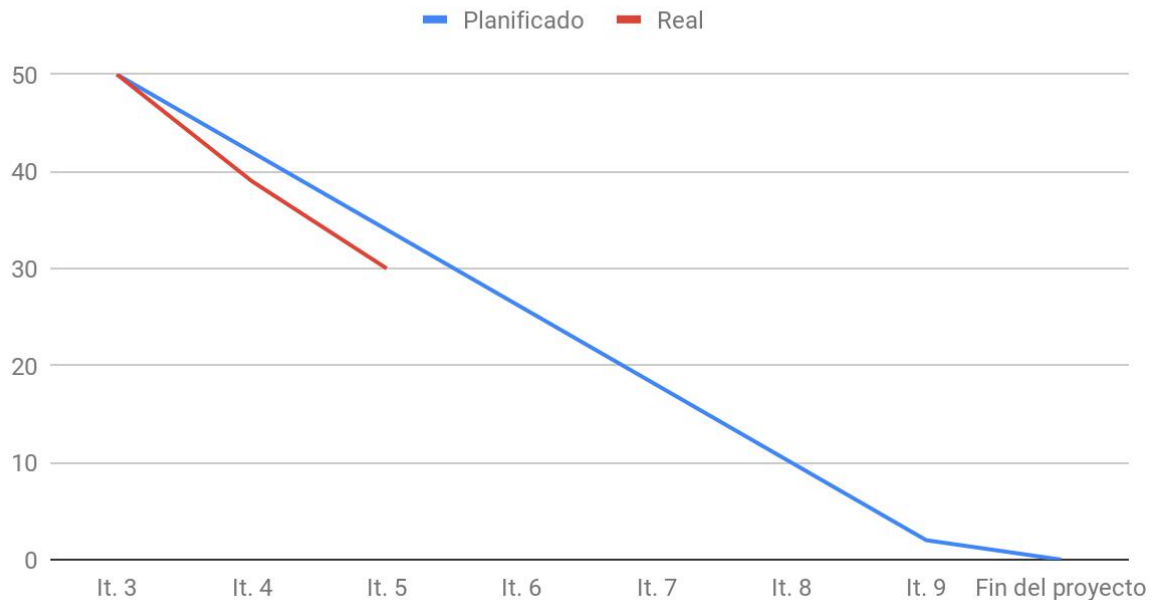


Ilustración 26. Diagrama de evolución del proyecto al comienzo de la iteración 5

El proyecto sigue siendo estando adelantado. Se podría plantear un nuevo cálculo de la planificación estimada y un ajuste de velocidad, pero se decide mantenerlo tal y como está ya que los resultados son buenos.

Hasta este momento se ha trabajado principalmente en la funcionalidad correspondiente con el rol médico. Con el fin de poder mostrar un prototipo funcional correspondiente al rol paciente, se acuerda con el cliente la priorización de algunas de las historias de usuario más relevantes para este rol. En este caso las historias “Paciente consulta terapias asignadas” y “Paciente realiza una terapia”. Por lo tanto, las funcionalidades correspondientes a la gestión de mensajería de la terapia quedan replanificadas para iteraciones posteriores.

- Paciente consulta terapias asignadas, con ID = 15 ([18.1.15. Paciente consulta terapias asignadas](#))
- Paciente realiza una terapia, con ID = 16 ([18.1.16. Paciente realiza una terapia](#))
- Médico ve progreso de una terapia, con ID = 10 ([18.1.10. Médico ve progreso de una terapia](#))

14.1.6 - Iteración 6

En la reunión con el product owner se aceptan las historias de usuario implementadas. El diagrama de evolución actual es el siguiente:

Planificado y Real

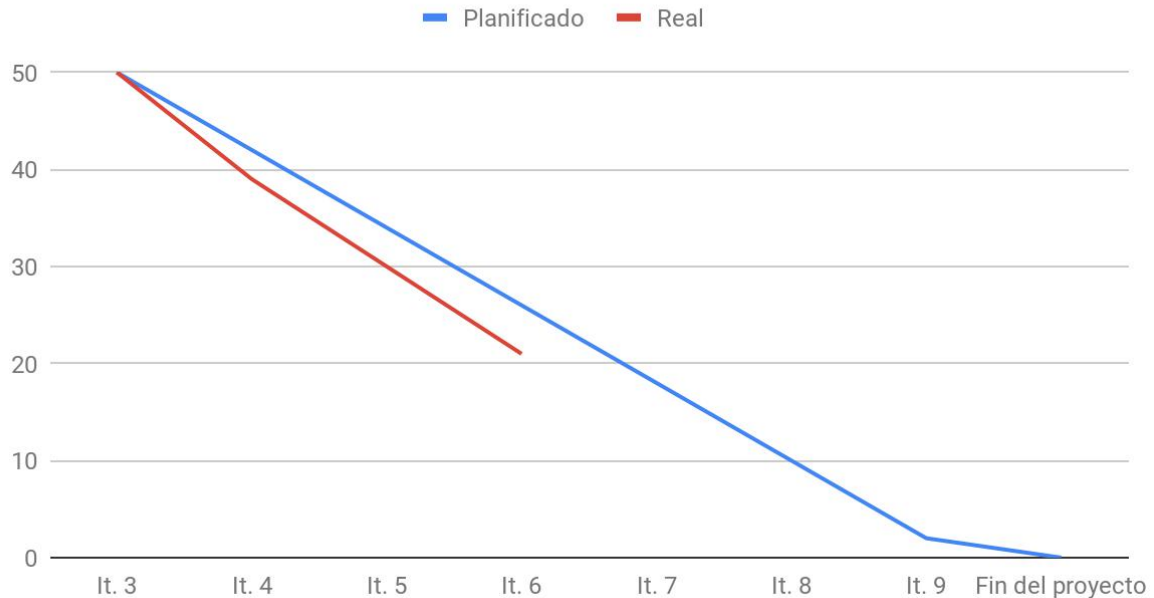


Ilustración 27. Diagrama de evolución del proyecto al comienzo de la iteración 6

Se acuerdan las siguientes historias de usuario para trabajar en la iteración 6:

- Usuario envía mensaje de una terapia, con ID = 12 ([18.1.12 Usuario envía mensaje de una terapia](#))
- Usuario lee mensaje de una terapia, con ID = 13 ([18.1.13. Usuario lee mensaje de una terapia](#))
- Usuario modifica mensaje de una terapia, con ID = 14 ([18.1.14. Usuario modifica mensaje de una terapia](#))

14.1.7 - Iteración 7

En la reunión al principio de la iteración con el product owner se evalúan primero los resultados obtenidos en la iteración anterior, siendo las historias de usuario implementadas como aceptadas.

Planificado y Real

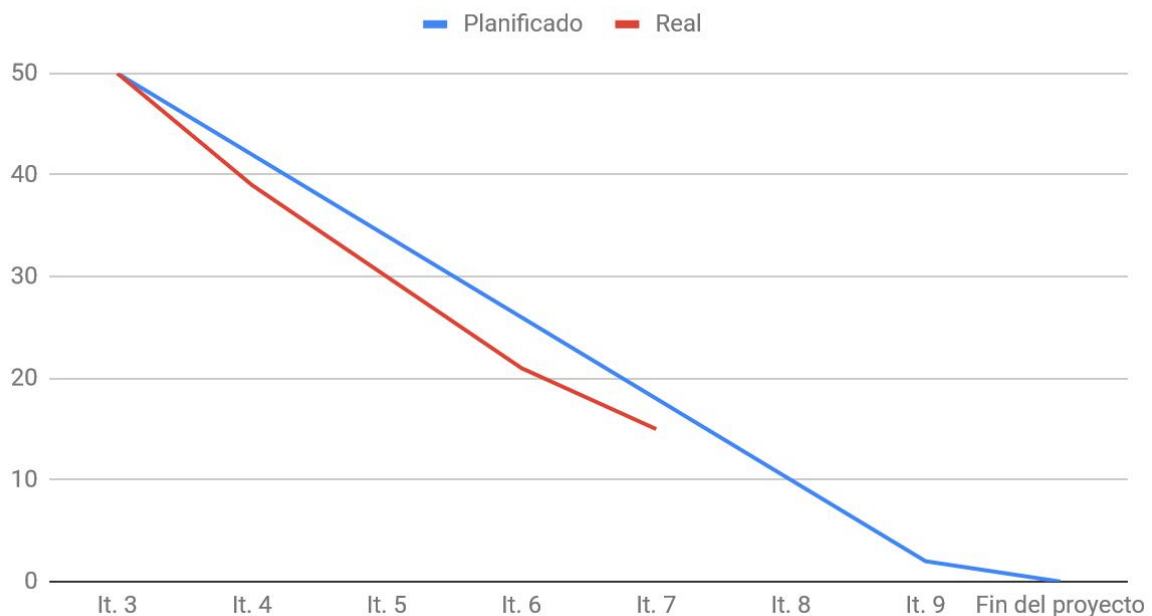


Ilustración 28. Diagrama de evolución del proyecto al comienzo de la iteración 7

Existe una ligera bajada de velocidad en la iteración 6, ya que las historias de usuario implementadas suman un total de 6 puntos de historia. Sin embargo, seguimos adelantados con respecto a la planificación inicial.

En este punto del proyecto se consideran como realizadas las funcionalidades del sistema consideradas como de mayor prioridad por el cliente. El objetivo ahora es la gestión de contenido multimedia que sirva como refuerzo a los ejercicios terapéuticos. En esta iteración se plantea para su realización la siguiente historia de usuario:

- Médico asigna video a ejercicio terapéutico, con ID = 17 ([18.1.17. Médico asigna un video a un ejercicio terapéutico](#))

14.1.7 - Iteración 8

Al igual que hemos hecho anteriormente, al inicio de la iteración se evalúan los resultados obtenidos en la iteración previa. En este caso en particular, cabe mencionar que la historia con ID = 17 que se realizó en la iteración anterior pudo ser desarrollada en un tiempo menor del esperado, por lo que se procedió de forma adicional a la implementación por parte del equipo de la siguiente historia de usuario:

- Usuario ve video de ejercicio terapéutico, con ID = 18 ([18.1.18. Usuario ve video de ejercicio terapéutico](#))

Ambas historias, tanto la de ID = 17 como la de ID = 18 quedan aceptadas por parte del product owner. El diagrama de evolución al inicio de la iteración 8 sería el siguiente:

Planificado y Real

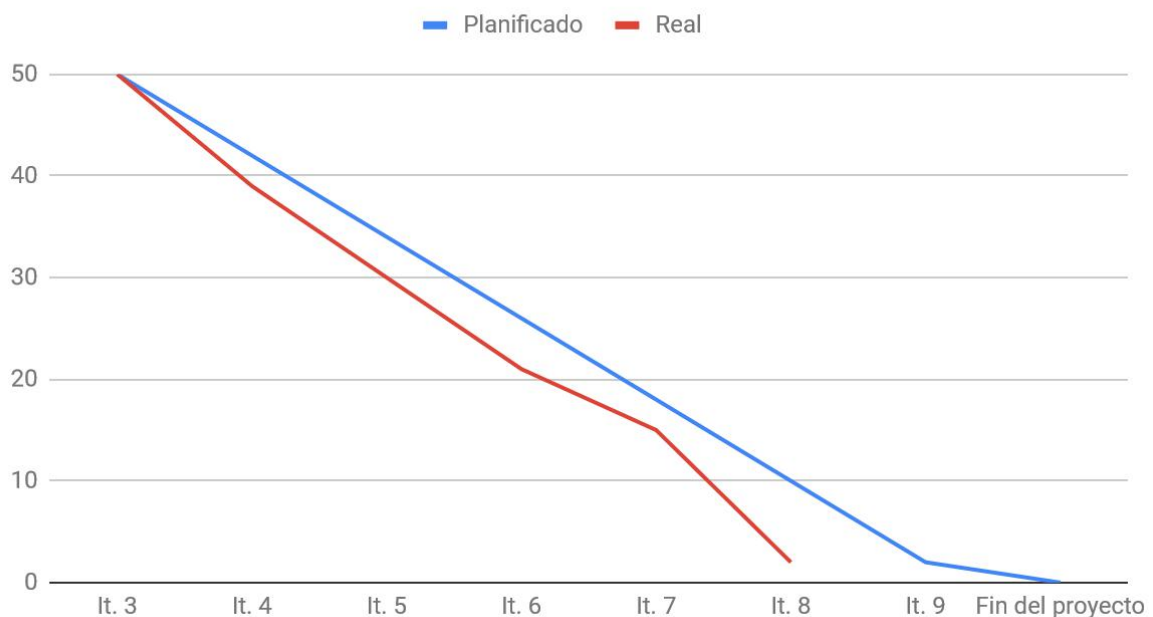


Ilustración 29. Diagrama de evolución del proyecto al comienzo de la iteración 8

Como se puede apreciar, ya se han concluido casi del todo todas las historias de usuario planificadas en el proyecto, solo faltaría “Modificar perfil de usuario”.

Se procede a realizar un análisis del sistema para tratar de mejorar algunos aspectos. Se acuerdan algunos pequeños cambios sobre las siguientes historias de usuario:

- En “Médico asigna una terapia a un paciente”, permitir cambiar las unidades de duración de los ejercicios.
- En “Paciente consulta terapias asignadas”, sólo se mostrarán las terapias actuales.

- En “Paciente realiza una terapia”, se permitirá al paciente realizar una sesión de la terapia en un día distinto al estipulado.
- En “Médico ve evolución de una terapia”, el médico podrá ver cuando una terapia no ha sido realizada en su día estipulado, en consecuencia con el cambio de la funcionalidad visto en “Paciente realiza una terapia”.

14.1.8 - Iteración 9

Se muestran los resultados obtenidos al cliente y se trabaja en la elaboración de documentación. En este punto ya se han implementado todas las historias de usuario previstas y se ha llegado al fin del proyecto.

Planificado y Real

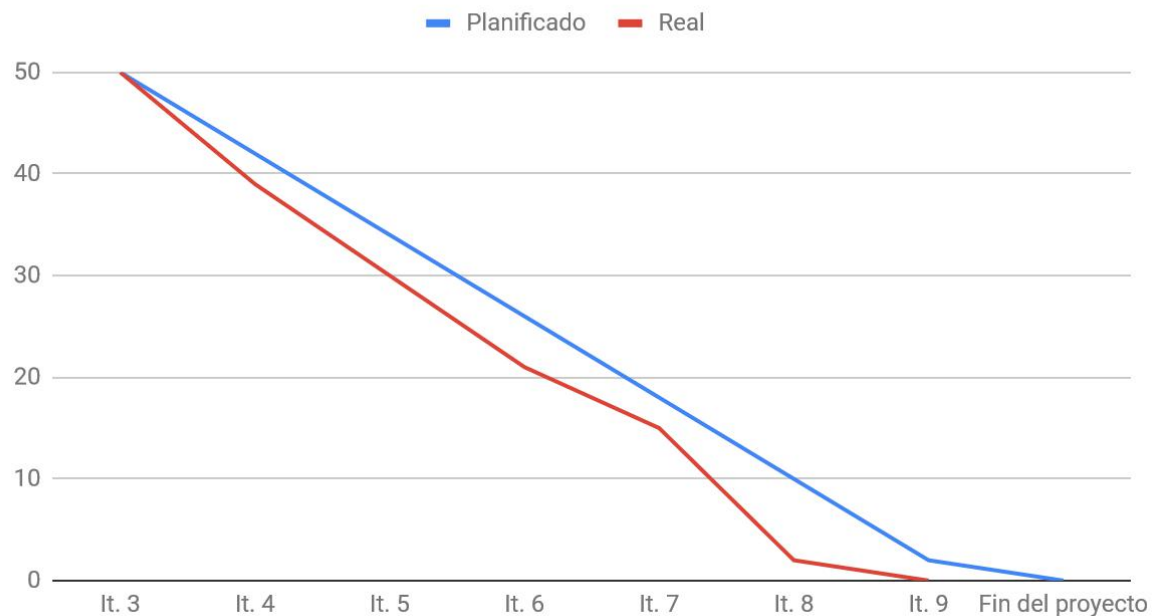


Ilustración 30. Diagrama de evolución del proyecto al comienzo de la iteración 9

15- Resumen del presupuesto

El importe final del proyecto será de 11.552,09€ . Este valor ha sido calculado en base al coste de ejecución del proyecto, valorado en 9626,74 € , al que se le ha aplicado un 20% de beneficio.

En la sección [19. Presupuesto](#) se analiza de forma detallada todas las consideraciones y cálculos realizados para la obtención de las cifras dadas.

16. Orden de prioridad de los documentos básicos del proyecto

En el caso en que se presenten discrepancias o incompatibilidades entre diversos documentos básicos del proyecto, deben prevalecer en este orden las Especificaciones del Sistema, el Presupuesto y la Memoria.

17. Anexos

17.1. Documentación de entrada

La documentación de entrada del proyecto consiste en la proporcionada por la propuesta del TFG (Cód.: 18/19-2462) Aplicación para la asignación, realización y seguimiento de terapias en casa y las normativas con respecto a Trabajos de Fin de Grado de la Universidad de Jaén y la Escuela Politécnica Superior.

17.2. Análisis y diseño del sistema

17.2.1 - Estructura del backend

Modelo del sistema

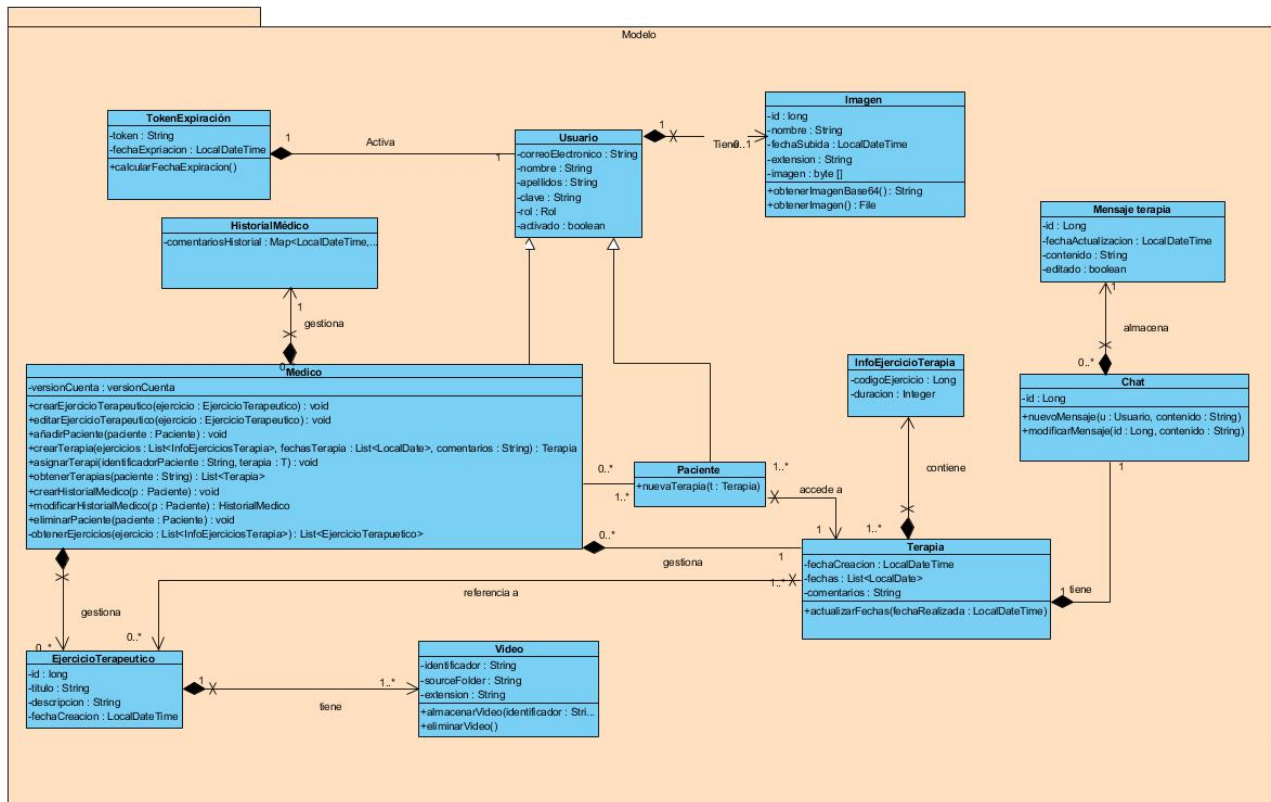


Ilustración 31. Diagrama de clases del modelo del sistema

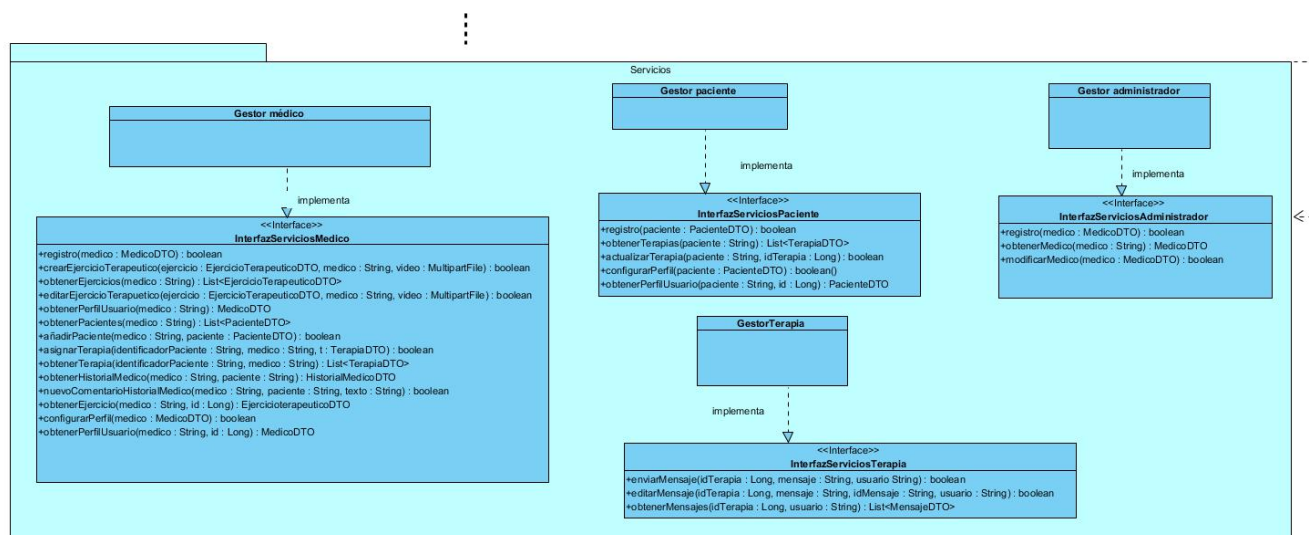
A continuación se realiza una breve descripción de las clases más relevantes:

- **USUARIO**: clase padre de la que heredan los diferentes tipo de usuarios considerados en el sistema.
- **MÉDICO**: clase que representa al usuario con rol médico. Se encarga de gestionar los objetos de las clases *EJERCICIOTERAPÉUTICO*, *TERAPIA* e *HISTORIALMÉDICO* además de interactuar con la clase *PACIENTE*.
- **PACIENTE**: clase que representa al usuario con rol paciente. Se relaciona con la clase *MÉDICO* y la clase *TERAPIA*. Tiene acceso a la información terapias que le han sido asignadas.

- *EJERCICIOTERAPÉUTICO*: representa cada uno de los ejercicios diseñados por un usuario con rol médico. Cada ejercicio terapéutico cuenta con información para identificarlo claramente y opcionalmente, un objeto de la clase *VIDEO*, que contendrá contenido audiovisual sobre su realización.
- *TERAPIA*: representa un plan de ejercicios terapéuticos. Cada terapia tendrá asociada diversos objetos de la clase *EJERCICIOTERAPÉUTICO*, información general sobre la terapia, las fechas de realización de la terapia y varios objetos de la clase *INFOEJERCICIOTERAPIA*.
- *INFOEJERCICIOTERAPIA*: almacena la duración de cada ejercicio de una terapia. Es una clase con un carácter principalmente auxiliar, ya que la información que contiene se podría haber incluido en la propia clase *TERAPIA* utilizando una estructura de datos, por ejemplo un mapa. Sin embargo, la transmisión de información entre cliente y servidor se simplifica bastante al incluir esta clase.
- *CHAT*: clase utilizada para gestionar el sistema de mensajería que ofrece el sistema. Gestiona objetos de la clase *MENSAJETEREAPIA*
- *MENSAJETEREAPIA*: representa cada uno de los mensajes enviados por el médico o el paciente.
- *VIDEO*: representa y gestiona la información de los videos con los que se trabaja en el sistema. Se encarga tanto de almacenar la información correspondiente a un video en el sistema de archivos como de recuperarla y prepararla para que los usuarios puedan acceder a ella.
- *IMAGEN*: *representa y gestiona la información de las imágenes con las que se trabaja en el sistema.*

Servicios

Ilustración 32. Diagrama de clases del paquete servicios.



El paquete de servicios contiene diferentes interfaces que especifican los servicios disponibles en el sistema. Cada una de estas interfaces es implementada por un objeto de negocio.

Interfaces REST

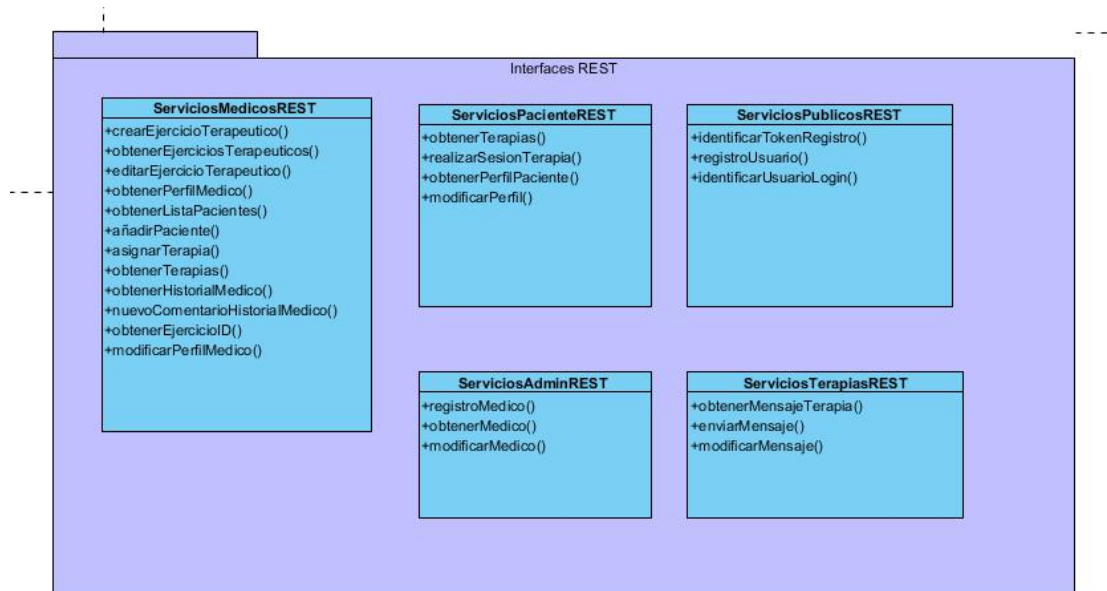


Ilustración 33. Diagrama de clases del paquete Interfaces REST

El API REST implementado por las clases de este diagrama se detalla más adelante en la sección [18.2. - End-points de la aplicación](#)

DAOs

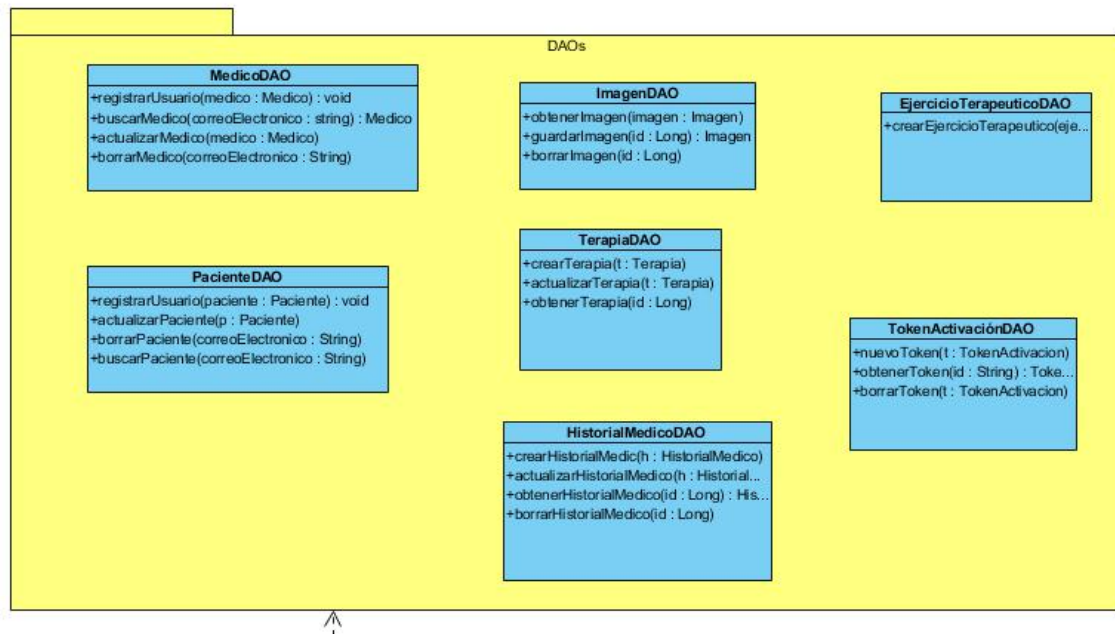


Ilustración 34. Diagrama de clases del paquete DAO.

El paquete DAOs contiene cada uno de los objetos utilizados para gestionar la persistencia. Concretamente, hay un objeto DAO especializado para cada una de las clases del modelo que requieren persistencia en la base de datos.

Diagrama de paquetes del backend

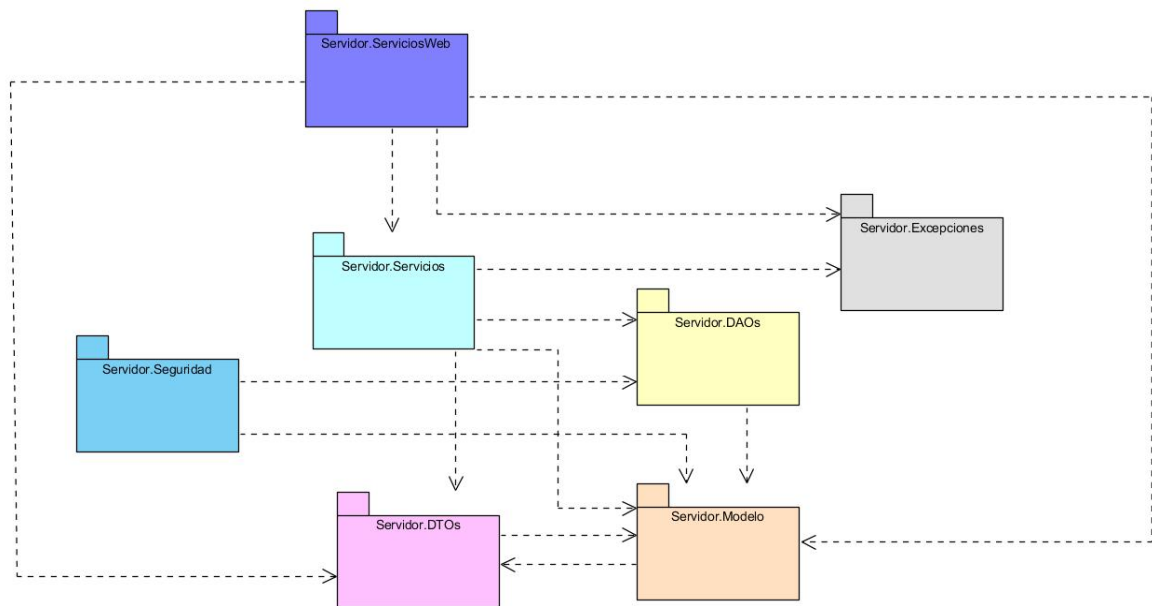


Ilustración 35. Diagrama de paquetes del backend

17.2.2- Diagrama entidad - relación de la base de datos

Las creación de tablas y sus relaciones se ha gestionado automáticamente mediante Hibernate y JPA. El diagrama entidad-relación que refleja el resultado generado por las tecnologías mencionadas se muestra en la siguiente imagen:

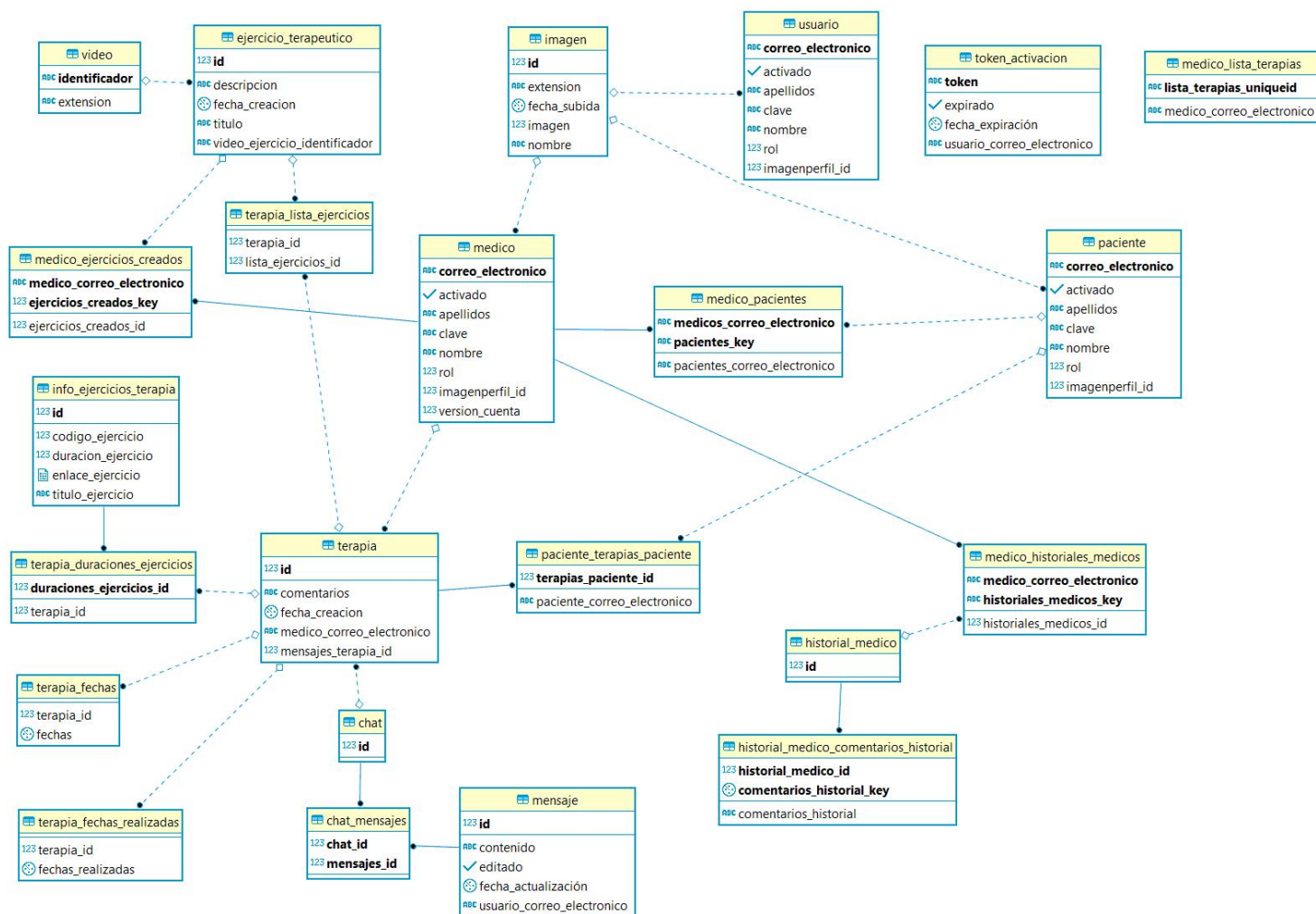


Ilustración 36. Diagrama entidad-relación de la base de datos

17.2.3 - Estructura del frontend

La estructura del frontend se ha enfocado de manera que permita desacoplar al máximo las relaciones entre los diferentes roles que soporta el sistema.

Tendremos una página de login, que sirve como nexo de acceso a los ficheros .html principales asociados a cada uno de los roles. Cada rol tendrá un directorio especializado en él, con diferentes módulos y componentes para poder acceder a las funcionalidades que tiene asignadas en el sistema. Se seguirá en todos ellos un enfoque puramente basado en SPA, de tal modo que cada rol contará con un único fichero .html, y a partir de las tecnologías que nos proporciona AngularJS, se irá actualizando la vista en función de las opciones seleccionadas durante el flujo de trabajo del usuario final.

Para la documentación de AngularJS he decidido utilizar un diagrama en el que se muestran los diferentes módulos y componentes del sistema. Cada módulo contiene una serie de componentes, trabajando como si fuera una aplicación de Angular independiente pero pudiendo relacionarse con otros módulos si así fuera necesario. Los módulos se han creado de manera que las funcionalidades que engloben posean similitud semántica y funcional. Cada componente por su parte tendrá un controlador con diferentes métodos y atributos y un fichero de tipo .template.html, cuyo código html se mostrará en el fichero .html principal cuando sea necesario.

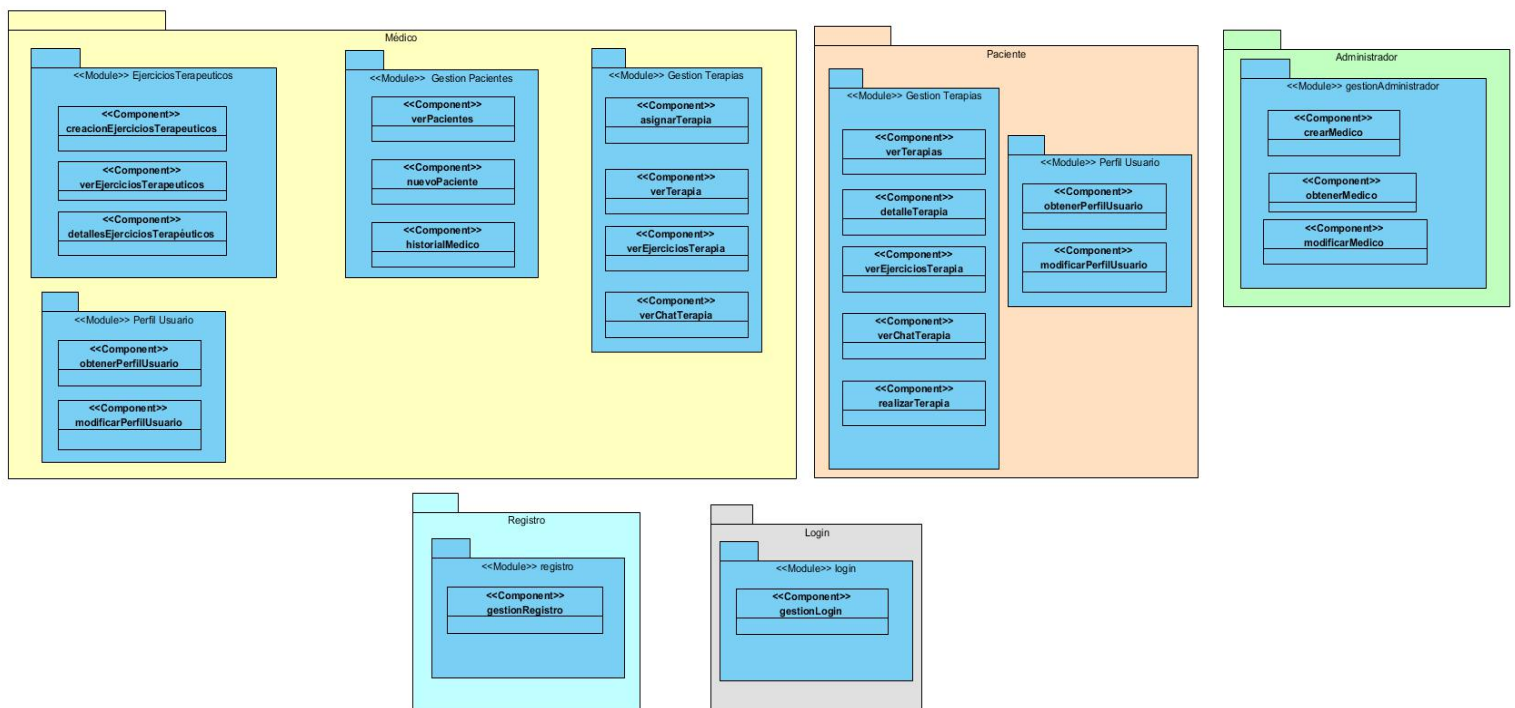


Ilustración 37. Diagrama de módulos y componentes del frontend

17.3. Plan de seguridad

Con el objetivo de cumplir con la normativa en materia de seguridad, es necesario tener en cuenta los puntos claves de la LOPD y RGPD.

El sistema contará con encriptado de contraseñas y se atenderá a la normativa de derechos ARCO de cara a los usuarios finales, permitiendo satisfacer los derechos de rectificación y cancelación de forma rápida y eficaz. Además, el sistema contará con conexión segura vía HTTPS mediante la obtención de un certificado.

De cara a la empresa que encarga el desarrollo de la aplicación, deberá nombrar y formar a una figura encargada de la responsabilidad del fichero de datos, a una figura encargada del fichero de datos y a una figura de delegado de protección de datos, que estarán obligadas al deber de secreto. También se recomienda la realización de auditorías anuales al menos con un carácter periódico de un año, las cuales podrán ser realizadas por la propia empresa que ha desarrollado el sistema.

18. Especificaciones del sistema

18.1 Historias de usuario

Debido a la metodología utilizada, de los requisitos iniciales que aparecen en el apartado 7 de este mismo documento se han desarrollado una serie de historias de usuario, que reflejan las diferentes funcionalidades del sistema. A lo largo del proyecto han surgido nuevas historias de usuario y se han modificado otras. Las historias de usuario que aparecen en la siguiente sección muestran la última versión de cada historia de usuario.

Además, para la mayoría de historias de usuario se muestra un diagrama de secuencia simplificado del funcionamiento a nivel de backend para facilitar la posterior aplicación, y en algunos casos información adicional sobre cómo llevarla a cabo desde un punto de vista tecnológico,

En algunos casos también se ha añadido información adicional para reflejar alguna característica relevante dicha historia de usuario de cara a su posterior implementación.

18.1.1 Médico crea ejercicio terapéutico

ID	Título	Puntos de historia	Prioridad
1	Médico crea ejercicio terapéutico	2	Must
Descripción			
Como médico quiero crear ejercicios terapéuticos para poder asignarlos posteriormente a mis pacientes			
Criterios de aceptación			
- Cuando se crea un ejercicio terapéutico este aparecerá en la lista de ejercicios creados del médico			
- Un ejercicio terapéutico consiste de un título y una descripción sobre cómo realizarlo. Si no se incluye información sobre estos campos, se mostrará un error.			

Tabla 3. Historia de usuario "Médico crea ejercicio terapéutico"

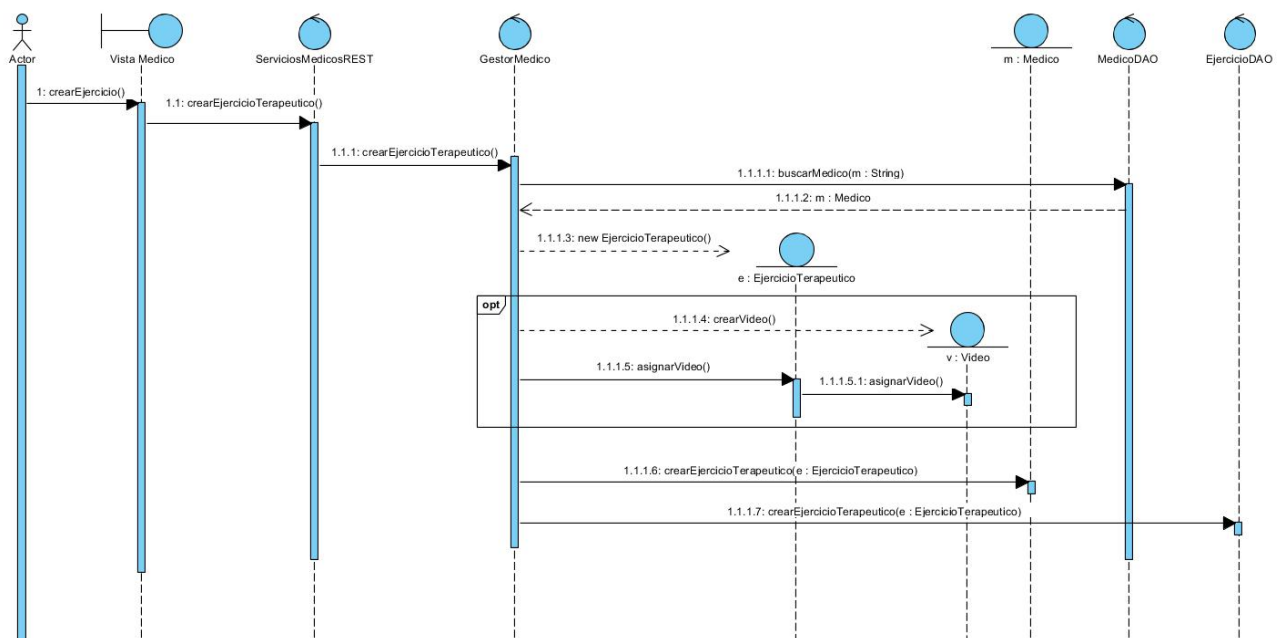


Ilustración 38. Diagrama de secuencia de la historia de usuario "Médico crea ejercicio terapéutico"

18.1.2. Médico modifica ejercicio terapéutico

ID	Título	Puntos de historia	Prioridad
2	Médico modifica ejercicio terapéutico	2	Must
Descripción			
Como médico quiero poder modificar un ejercicio terapéutico que he creado para modificar o corregir información de este			
Criterios de aceptación			
- Si el título o la descripción quedan vacíos, se anulará la modificación			
- Los resultados se reflejarán inmediatamente en la lista de ejercicios del médico			

Tabla 4. Historia de usuario “Médico modifica ejercicio terapéutico”

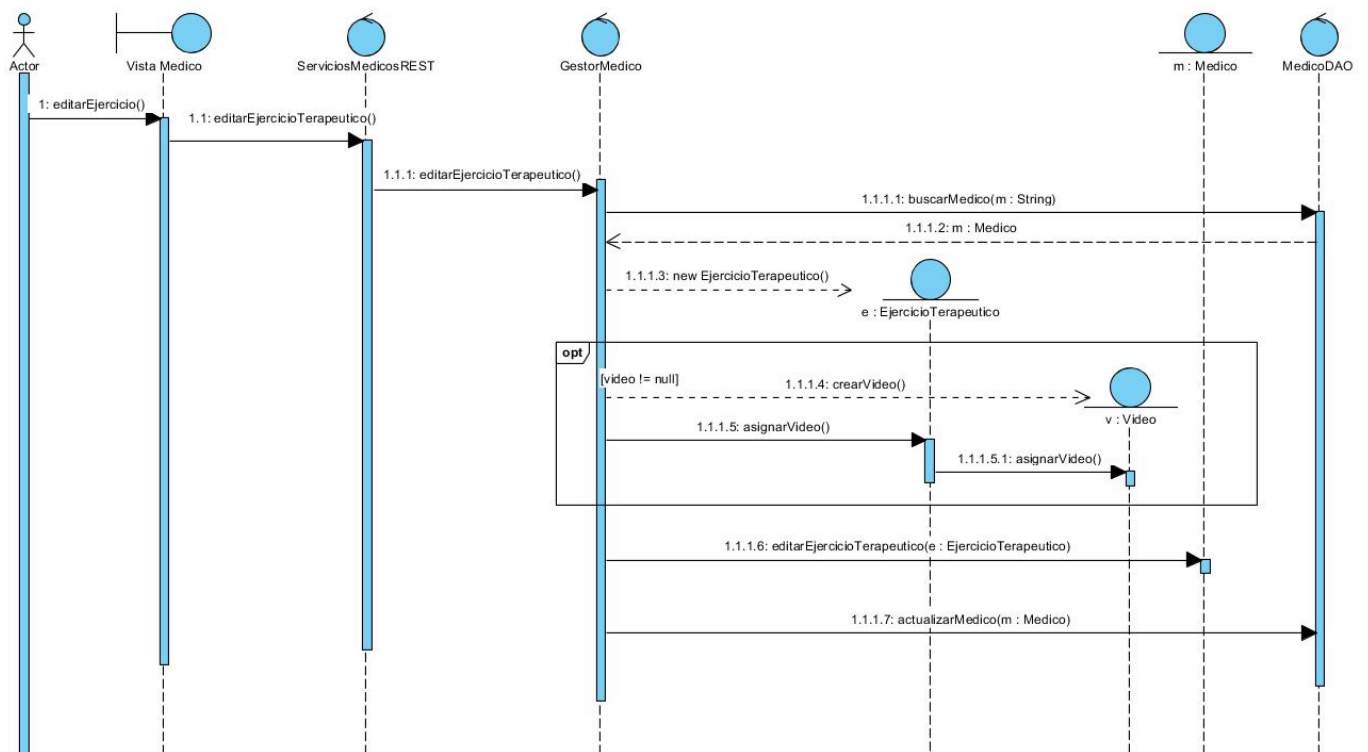


Ilustración 39. Diagrama de secuencia de la historia de usuario “Médico modifica ejercicio terapéutico”

18.1.3. Médico ve ejercicios terapéuticos

ID	Título	Puntos de historia	Prioridad
3	Medico ve ejercicios terapéuticos	2	Must
Descripción			
Como médico quiero ver la información de los ejercicios terapéuticos que he creado para poder obtener información de ellos			
Criterios de aceptación			
- Los ejercicios terapéuticos pueden ser ordenados de forma alfabética por su título.			
- Los ejercicios terapéuticos pueden ser ordenados por fecha de creación			
- Se pueden buscar ejercicios terapéuticos por palabras clave.			

Tabla 5. Historia de usuario “Médico ve ejercicios terapéuticos”

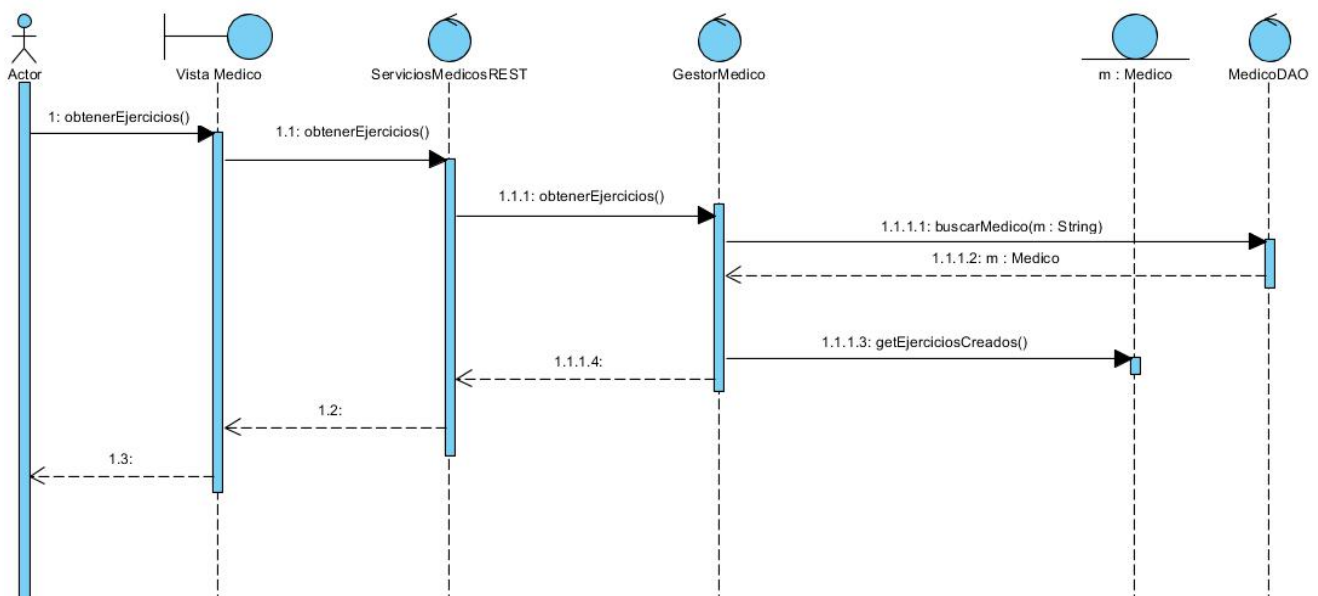


Ilustración 40. Diagrama de secuencia de la historia de usuario “Médico ve ejercicios terapéuticos”

18.1.4. Usuario se registra en el sistema

En un principio la aplicación se planteó como de uso abierto, por lo que el registro era accesible para cualquiera. Sin embargo, tras una de las reuniones con el product owner, se decidió cambiar la fórmula de acceso al sistema. La siguiente historia de usuario y diagrama de secuencia muestran la fórmula de acceso acordada tras dicha reunión.

La idea es que un usuario esté pre-registrado en el sistema, con un correo electrónico y su nombre y apellidos. Los administradores pre-registran a los médicos, y los médicos a los pacientes. Cuando se pre-registra a un usuario, se envía un correo con un acceso a la página de registro. Este acceso contiene un token identificador en la URL, que está asociado al usuario pre-registrado. Los usuarios desde esta página completan su registro, incluyendo una contraseña y una imagen de perfil de forma opcional.

ID	Título	Puntos de historia	Prioridad
4	Usuario se registra en el sistema	3	Must
Descripción			
Como usuario quiero poder registrarme en el sistema para acceder a las funcionalidades de este			
Criterios de aceptación			
- El usuario accede a la página de registro a través de un enlace a través de correo			
- La información como su nombre y correo electrónico ya estará preconfigurada. El usuario deberá introducir su contraseña y una imagen de perfil de forma optativa.			

Tabla 6. Historia de usuario "Usuario se registra en el sistema"

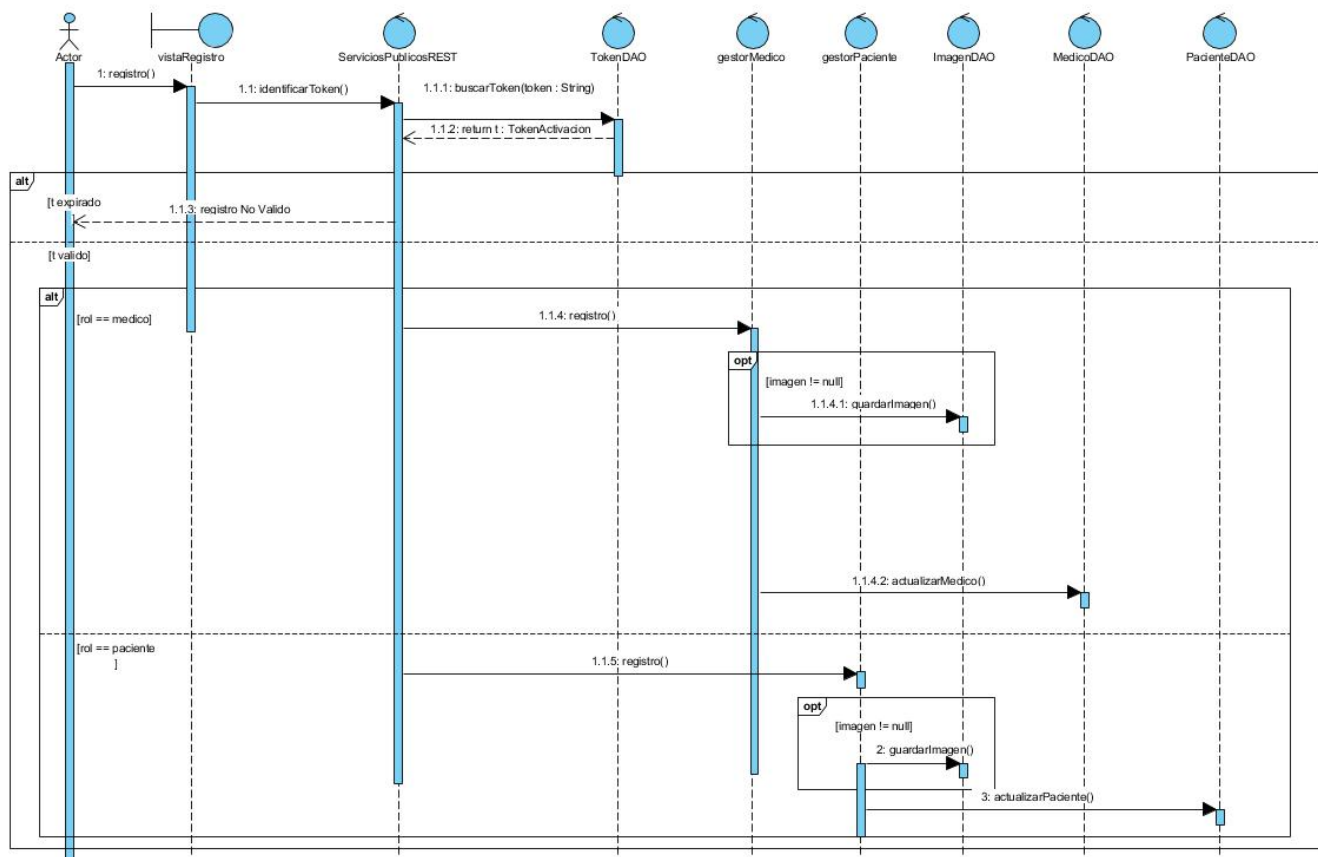


Ilustración 41. Diagrama de secuencia de la historia de usuario “Usuario se registra en el sistema”

18.1.5. Médico añade paciente

ID	Título	Puntos de historia	Prioridad
5	Médico añade paciente	2	Must
Descripción			
Como médico quiero añadir pacientes para poder asignarles terapias y consultar su progreso			
Criterios de aceptación			
- Un médico agrega a un paciente a partir del correo electrónico asociado a este			
- Si se intenta agregar a un paciente que ya está en la lista de pacientes, se mostrará error			

Tabla 7. Historia de usuario “Médico añade paciente”

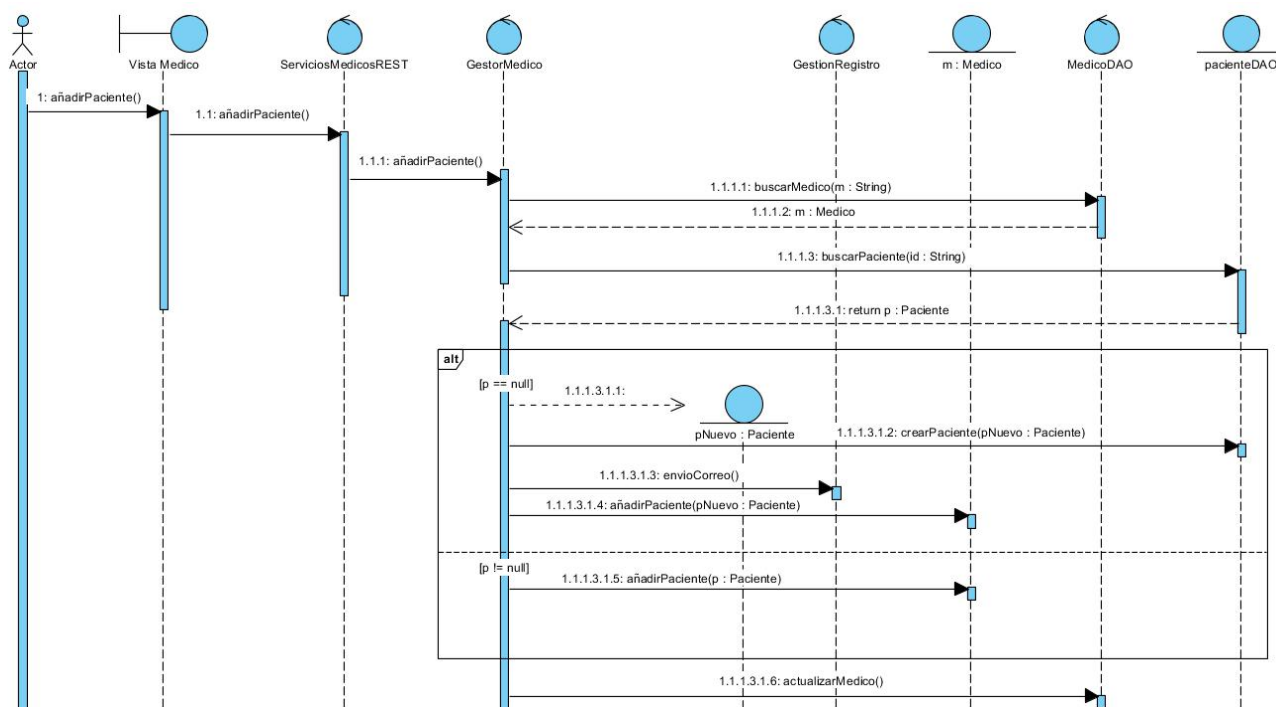


Ilustración 42. Diagrama de secuencia de la historia de usuario “Médico añade paciente”

18.1.6. Médico asigna una terapia a un paciente

ID	Título	Puntos de historia	Prioridad
6	Médico asigna una terapia a un paciente	3	Must
Descripción			
Como médico quiero poder asignar terapias para que mis pacientes puedan realizarlas			
Criterios de aceptación			
- El médico crea una terapia indicando el paciente a quien va dirigida, una serie de fechas, ejercicios terapéuticos con sus respectivas duraciones y una explicación de cómo llevarla a cabo - En caso de que no se incluyan todos los campos mencionados, no se podrá crear la terapia. - Cuando se asigna una terapia esta deberá ser visible por el paciente a quien se ha asignado y para el propio médico.			

Tabla 8. Historia de usuario "Médico asigna una terapia a un paciente"

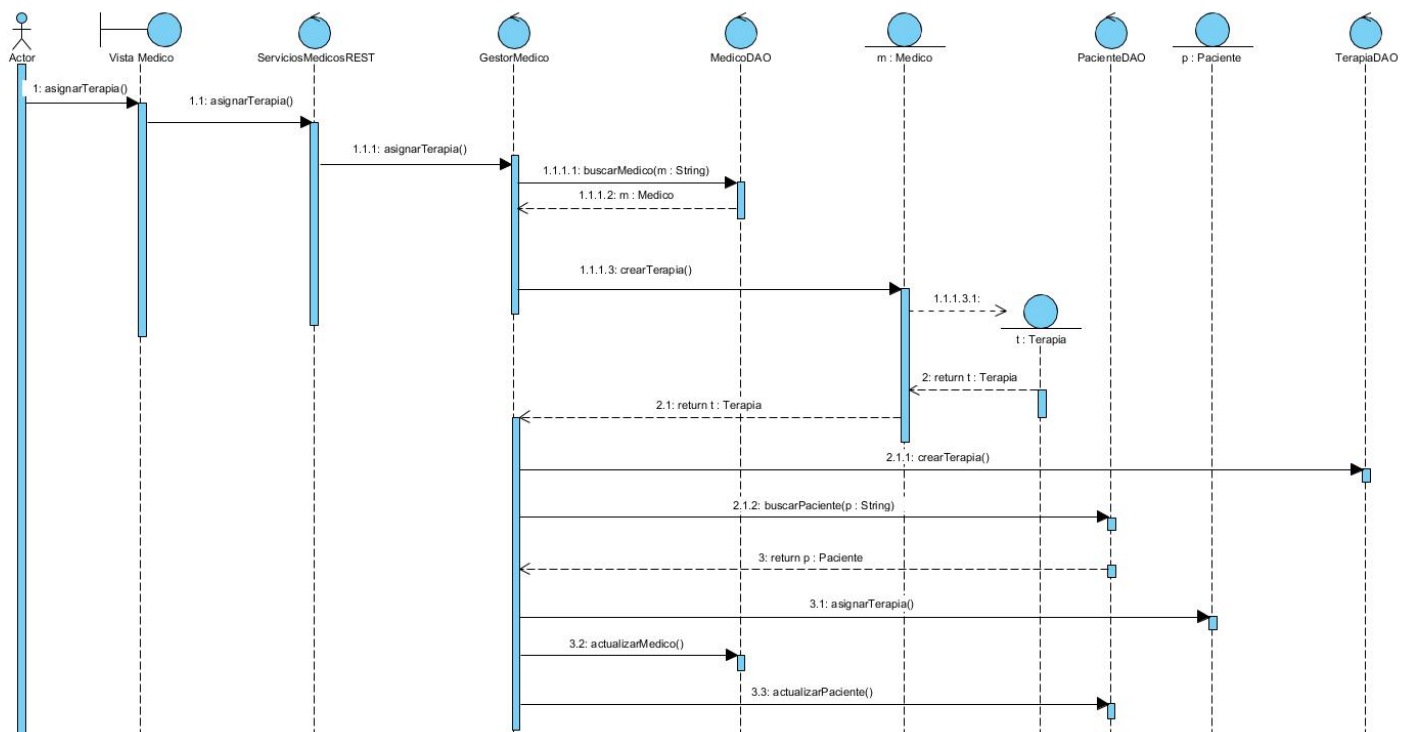


Ilustración 43. Diagrama de secuencia de la historia de usuario "Médico asigna una terapia a un paciente"

18.1.7. Médico consulta historial médico de un paciente

ID	Título	Puntos de historia	Prioridad
7	Médico consulta historial clínico de un paciente	2	Must
Descripción			
Como médico quiero consultar el historial médico de un paciente para obtener información sobre este			
Criterios de aceptación			
- El médico podrá ver las diferentes entradas que ha incluido en el historial médico de un paciente			
- Cada entrada del historial médico mostrará la fecha en la que fue redactada			
- Las entradas del historial médico se mostrarán en orden cronológico, comenzando por la más reciente			

Tabla 9. Historia de usuario “Médico consulta historial médico de un paciente”

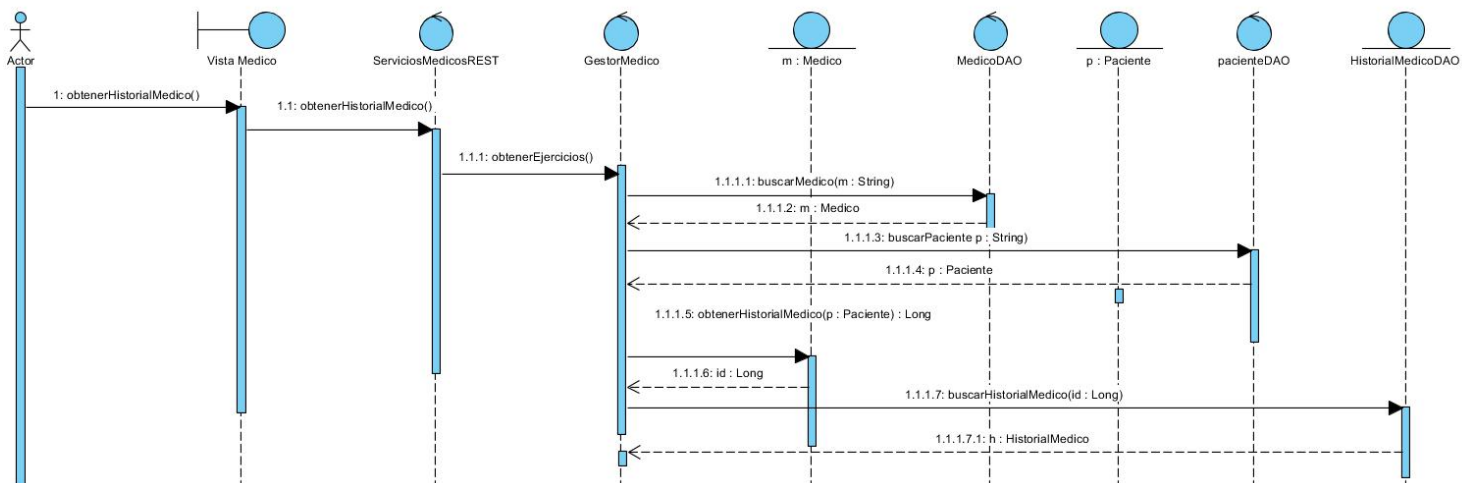


Ilustración 44. Diagrama de secuencia de la historia de usuario “Médico consulta historial médico de un paciente”

18.1.8. Médico modifica historial clínico de un paciente

ID	Título	Puntos de historia	Prioridad
8	Médico modifica historial clínico de un paciente	2	Must
Descripción			
Como médico quiero modificar el historial médico de un paciente para poder actualizar la información relativa a su estado de salud			
Criterios de aceptación			
-Cada nueva entrada en el historial médico quedará documentada con la fecha en la que se escribió			

Tabla 10. Historia de usuario "Médico modifica historial clínico de un paciente"

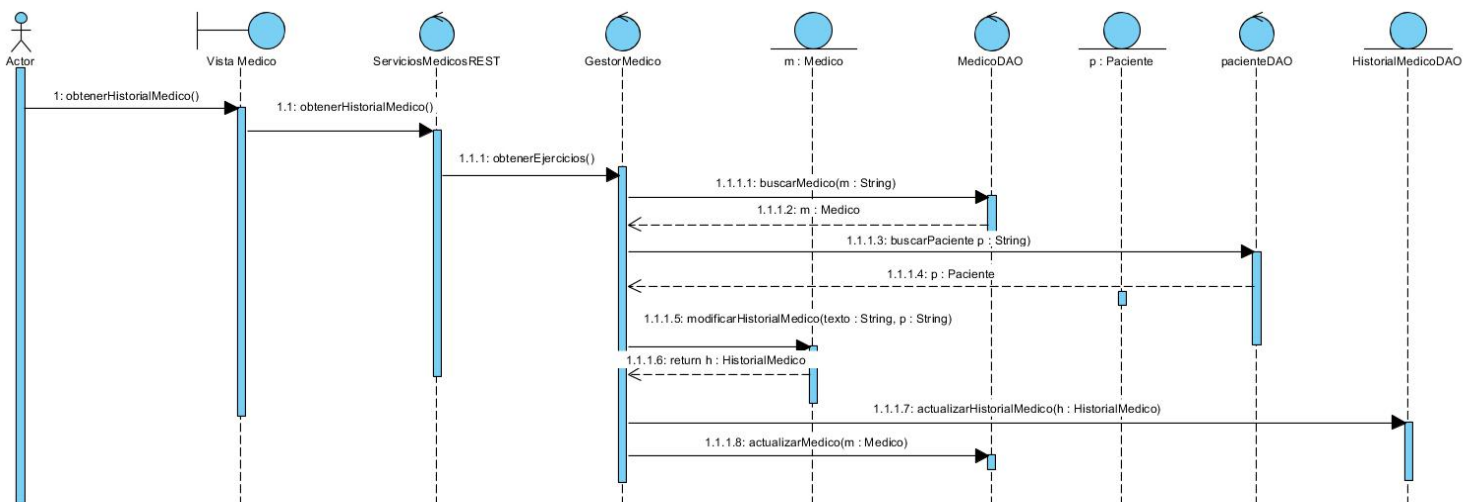


Ilustración 45. Diagrama de secuencia de la historia de usuario "Médico modifica historial clínico de un paciente"

18.1.9. Médico consulta terapias asignadas a un paciente

ID	Título	Puntos de historia	Prioridad
9	Médico consulta terapias asignadas a un paciente	3	Must
Descripción			
Como médico, quiero poder ver las terapias que he asignado a un paciente para obtener información sobre su evolución			
Criterios de aceptación			
- Se podrán ver las terapias ya realizadas como actuales de un paciente asignadas por el médico que hace la consulta			
- Las terapias en desarrollo tendrán una marca diferenciativa para ser fácilmente distinguibles por el médico			
- Cada terapia será accesible y se mostrarán las fechas y ejercicios asociados.			

Tabla 11. Historia de usuario "Médico consulta terapias asignadas a un paciente"

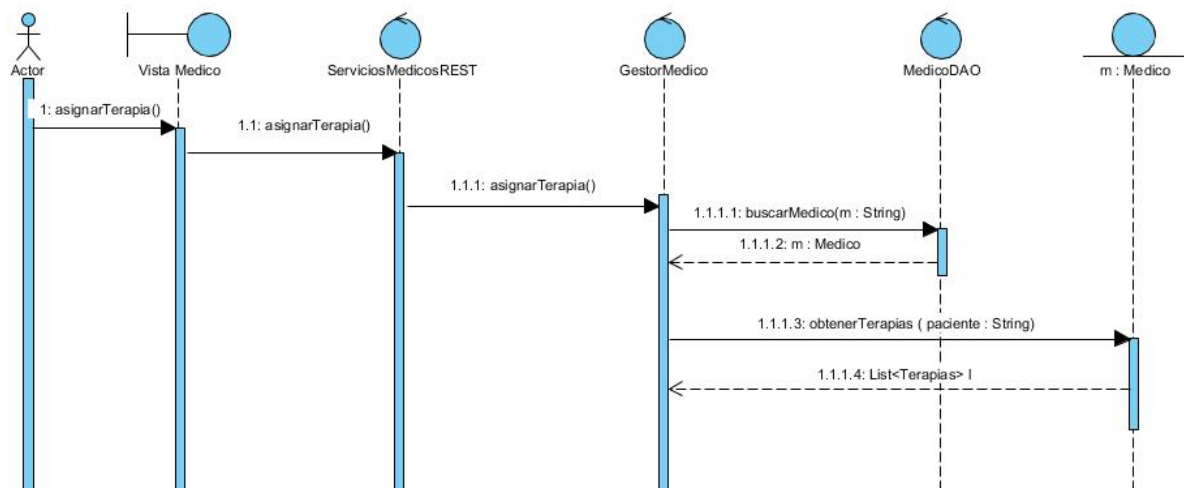


Ilustración 46. Diagrama de secuencia de la historia de usuario "Médico consulta terapias asignadas a un paciente"

18.1.10. Médico ve progreso de una terapia

ID	Título	Puntos de historia	Prioridad
10	Médico ve progreso de una terapia	2	Must
Descripción			
Como médico quiero ver el progreso realizado por un paciente en una terapia			
Criterios de aceptación			
- Se mostrarán las fechas con los días en las que una sesión de la terapia ha sido realizada y los días en los que una sesión de una terapia tendría que haber sido realizada pero no lo fue - Las sesiones de la terapia se mostrarán en un objeto con forma de calendario en el cliente y en una lista.			

Tabla 12. Historia de usuario “Médico ve progreso de una terapia”

La nueva información con la que se trabaja en esta historia de usuario quedará estructurada dentro de la propia clase *TERAPIA*, por lo que para acceder a la información de esta se sigue la misma secuencia de pasos que en el diagrama de secuencia de la historia de usuario “Médico consulta terapias asignadas a un paciente”. Ahora cada terapia cuenta con atributos adicionales para distinguir las diferentes fechas de las sesiones de las terapias en base a los criterios de aceptación descritos.

18.1.11. Usuario hace login en el sistema

ID	Título	Puntos de historia	Prioridad
11	Usuario hace login en el sistema	2	Must
Descripción			
- Como usuario quiero poder identificarme en el sistema para poder utilizar las funcionalidades de este			
Criterios de aceptación			
- Un usuario se identificará a través de su cuenta de correo y contraseña			

Tabla 13. Historia de usuario "Usuario hace login en el sistema"

La identificación del usuario se deberá de hacer en cada petición que se haga del cliente al servidor. Todas las peticiones a servicios marcados con restricciones de seguridad deberán de poseer unos headers adecuados que identifiquen al usuario. Para evitar que el usuario del sistema deba introducir sus credenciales de identificación constantemente, el login consistirá en una llamada al servidor, que en caso de indicarnos que la combinación correo electrónico - contraseña es correcta, nos devolverá un mensaje de confirmación. Una vez recibida dicha respuesta, desde el cliente se procederá a guardar dichos credenciales y utilizarlos para todas las peticiones que se hagan desde el cliente hasta que el usuario termine su sesión.

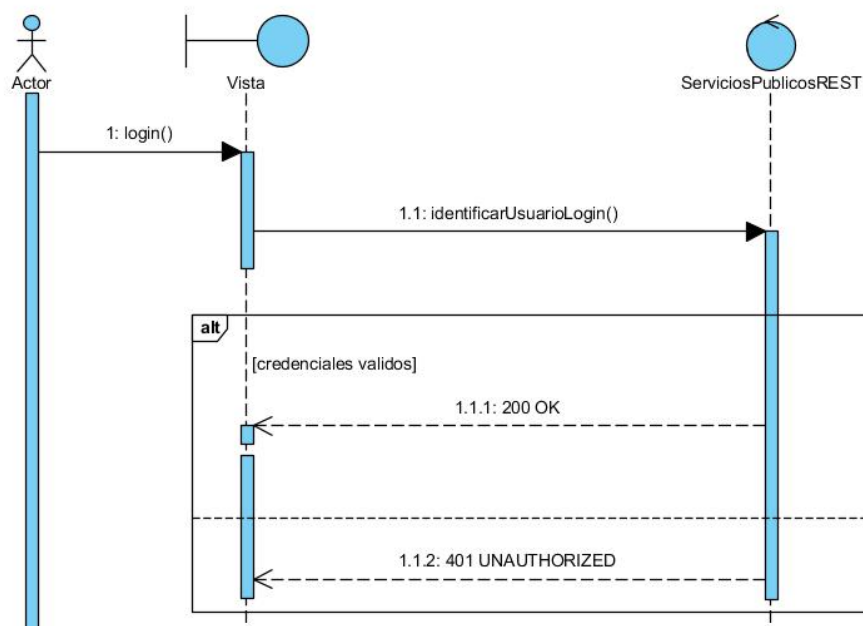


Ilustración 47. Diagrama de secuencia de la historia de usuario "Usuario hace login en el sistema"

18.1.12 Usuario envía mensaje de una terapia

ID	Título	Puntos de historia	Prioridad
12	Usuario envía mensaje en una terapia	2	Must
Descripción			
Como usuario quiero enviar mensajes sobre una terapia para poder preguntar, responder o aclarar información de esta.			
Criterios de aceptación			
- Cada mensaje incluirá la hora en la que fue enviado			

Tabla 14. Historia de usuario "Usuario envía mensaje de una terapia"

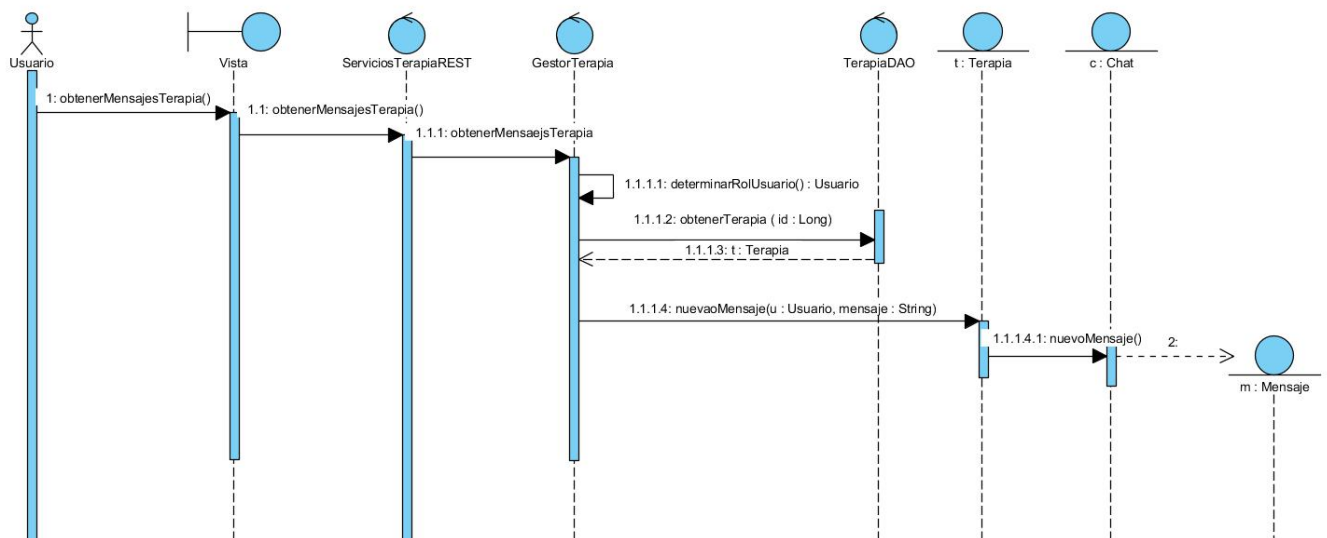


Ilustración 48. Diagrama de secuencia de la historia de usuario "Usuario envía mensaje de una terapia"

18.1.13. Usuario lee mensaje de una terapia

ID	Título	Puntos de historia	Prioridad
13	Usuario lee mensajes de una terapia	2	Must
Descripción			
Como usuario quiero poder ver los mensajes asociados a una de mis terapias			
Criterios de aceptación			
- Cada mensaje mostrará la imagen de perfil del usuario que lo ha enviado, el nombre y los apellidos de este, el cuerpo del mensaje y la fecha en la que fue enviado - Los mensajes se mostrarán ordenados por orden cronológico de más reciente a más antiguo			

Tabla 15. Historia de usuario “Usuario lee mensaje de una terapia”

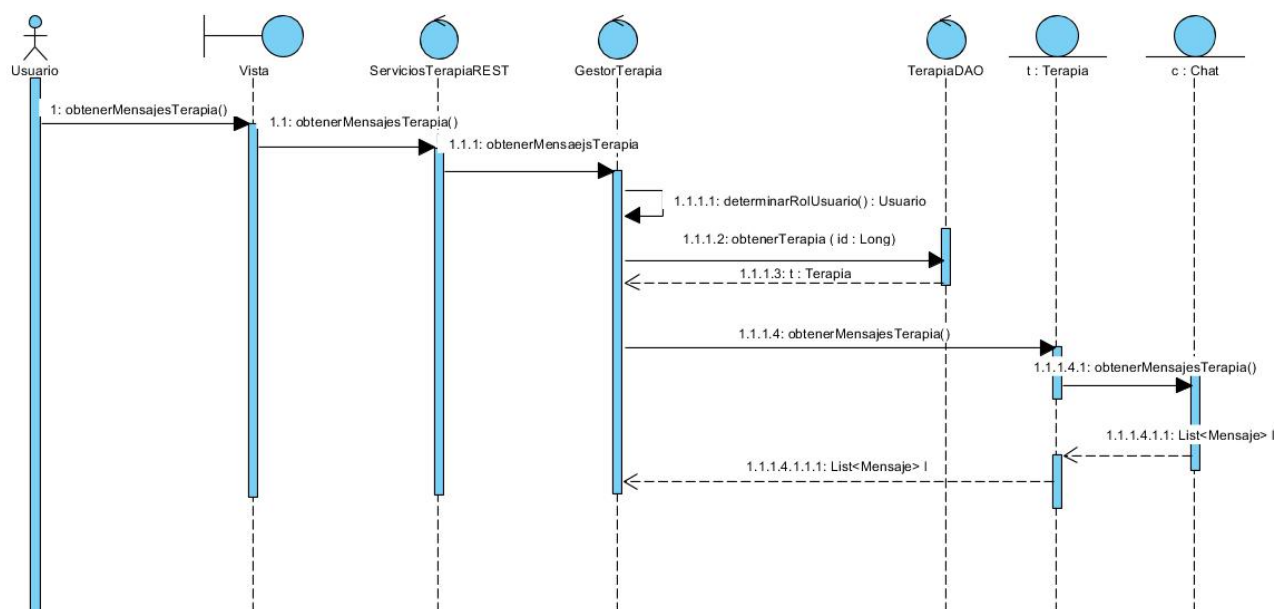


Ilustración 49. Diagrama de secuencia de la historia de usuario “Usuario lee mensaje de una terapia”

18.1.14. Usuario modifica mensaje de una terapia

ID	Título	Puntos de historia	Prioridad
14	Usuario modifica mensaje en una terapia	2	Could
Descripción			
Como usuario quiero poder modificar uno de los mensajes que he enviado para poder solucionar posibles errores cometidos durante la escritura de este.			
Criterios de aceptación			
- Los mensajes que se modifiquen actualizarán su fecha de envío por la nueva fecha en la que se modificó el mensaje - Los mensajes modificados tendrán algún tipo de indicador para que los usuarios sepan que el mensaje ha sido modificado			

Tabla 16. Historia de usuario “Usuario modifica mensaje de una terapia”

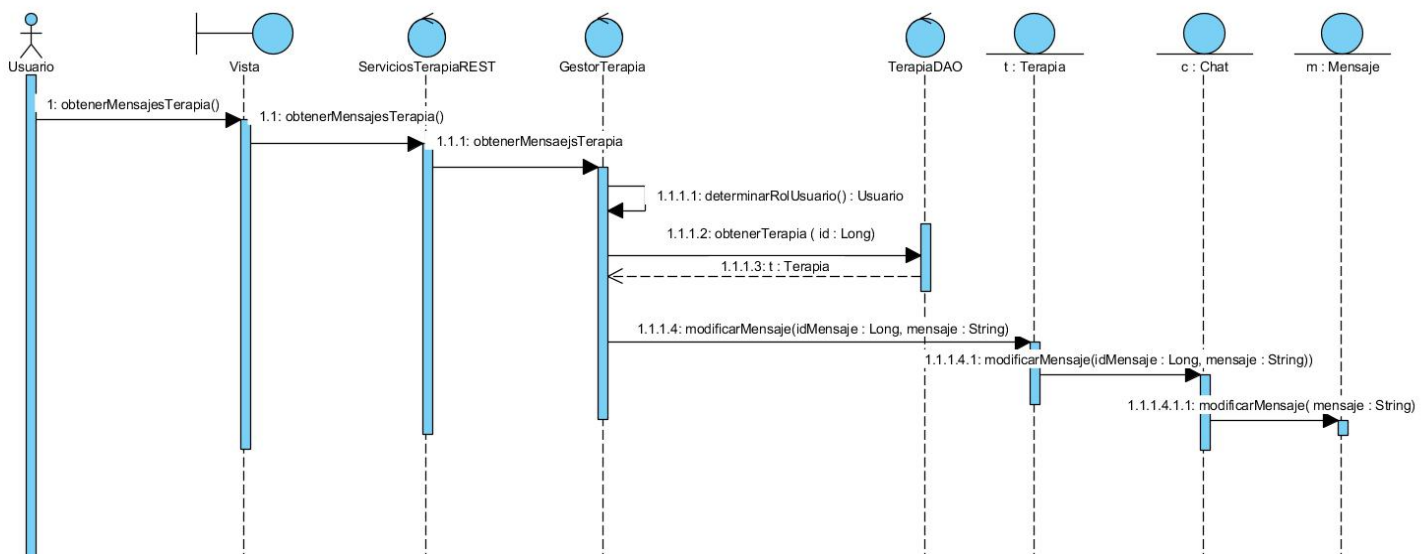


Ilustración 50. Diagrama de secuencia de la historia de usuario “Usuario modifica mensaje de una terapia”

18.1.15. Paciente consulta terapias asignadas

ID	Título	Puntos de historia	Prioridad
15	Paciente consulta terapias asignadas	2	Must
Descripción			
Como paciente quiero ver las terapias que tengo asignadas para poder realizarlas.			
Criterios de aceptación			
- El paciente podrá ver las terapias que tiene asignadas, añadiendo algun tipo de indicador para las terapias que tengan una fecha de sesión que coincida con el día actual - Para cada una de las terapias podrá ver las fechas de las sesiones que conforman la terapia y los ejercicios a realizar.			

Tabla 17. Historia de usuario "Paciente consulta terapias asignadas"

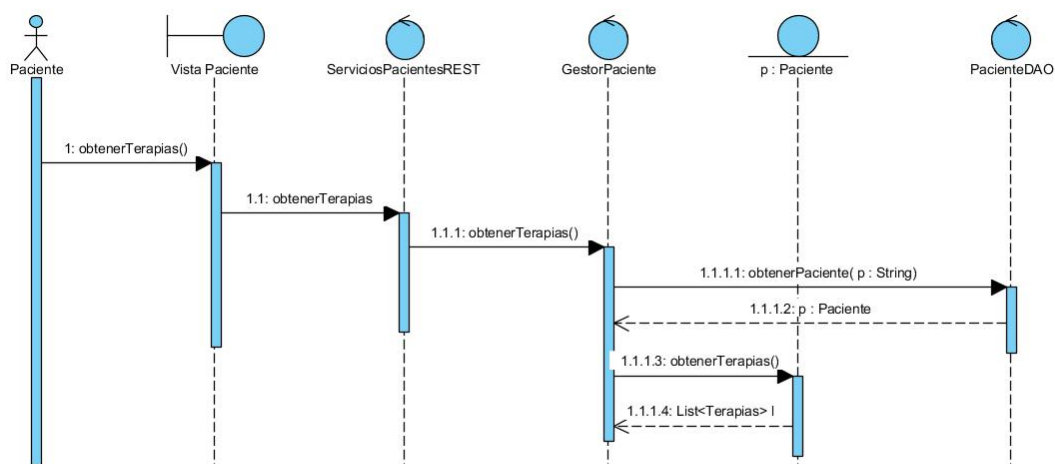


Ilustración 51. Diagrama de secuencia de la historia de usuario "Paciente consulta terapias asignadas"

18.1.16. Paciente realiza una terapia

ID	Título	Puntos de historia	Prioridad
16	Paciente realiza una terapia	5	Must
Descripción			
Como paciente quiero ver las terapias que tengo asignadas para poder realizarlas.			
Criterios de aceptación			
- La terapia aparecerá como "realizable" si una de sus sesiones coincide con la fecha actual			
- Se mostrarán los ejercicios de la terapia			
- Cada uno de los ejercicios contará con un contador que se ajustará a la duración que especificó el médico para estos			

Tabla 18. Historia de usuario "Paciente realiza una terapia"

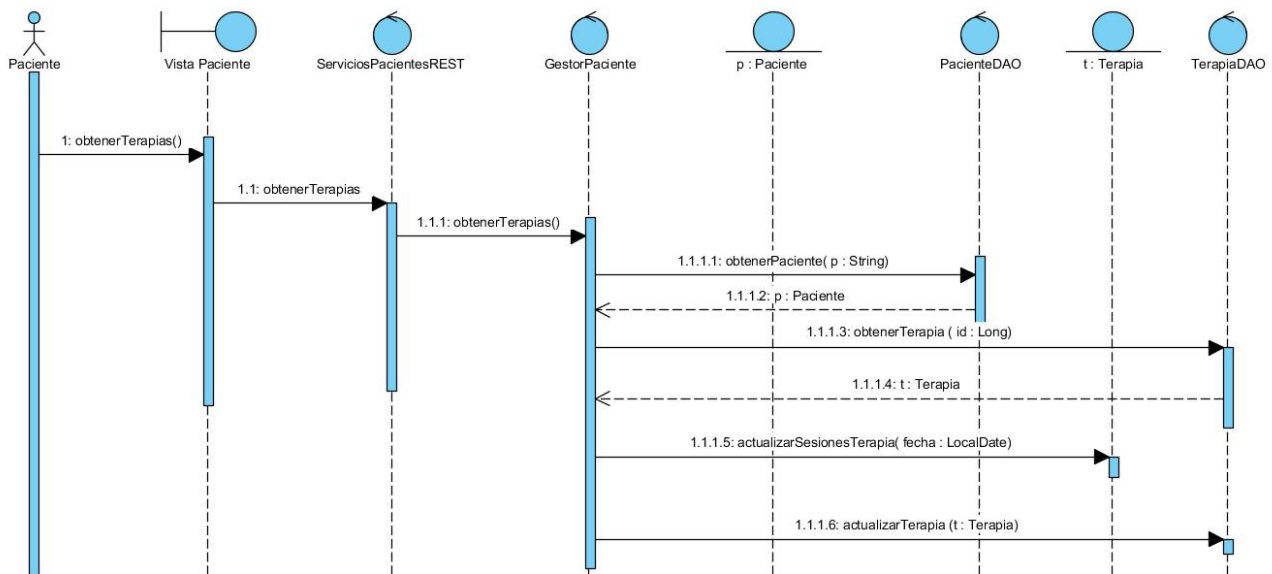


Ilustración 52. Diagrama de secuencia de la historia de usuario "Paciente realiza una terapia"

18.1.17. Médico asigna un video a un ejercicio terapéutico

ID	Título	Puntos de historia	Prioridad
17	Médico asigna video a un ejercicio terapéutico	8	Should
Descripción			
Como médico quiero asignar un video a un ejercicio terapéutico			
Criterios de aceptación			
- Al subir un video desde el cliente se mostrará una preview de este para que el médico pueda corroborar su validez.			
- El video no podrá pesar más de 100 Mb			
- Se podrá añadir un video tanto en la fase de creación de un ejercicio como en una posible posterior modificación de este.			
- En caso de que un ejercicio ya tenga asignado un video, se podrá borrar este o cambiar el video			

Tabla 19. Historia de usuario "Médico asigna un video a un ejercicio terapéutico"

La asignación de vídeos a nivel de diseño e implementación se ha unificado con la creación y modificación de ejercicios terapéuticos. En los diagramas de secuencia de la historia de usuario Médico crea ejercicio terapéutico y Médico modifica ejercicios terapéuticos aparecen las consideraciones pertinentes a esta historia de usuario

18.1.18. Usuario ve video de ejercicio terapéutico

ID	Título	Puntos de historia	Prioridad
18	Usuario ve video de ejercicio terapéutico	5	Should
Descripción			
Como usuario quiero poder ver el video asignado a un ejercicio terapéutico			
Criterios de aceptación			
- El video se retransmitirá mediante streaming desde el servidor			
- Los videos serán accesibles desde la misma vista donde se muestre el resto de información de un ejercicio			

Tabla 20. Historia de usuario "Usuario ve video de ejercicio terapéutico"

Para el streaming de contenido multimedia se ha utilizado un módulo adicional de Spring, *Spring Contents*, que nos permite simplificar enormemente el proceso. De cara a la implementación solo se ha de definir una interfaz e indicar la carpeta del sistema de archivos donde se almacenan los videos.

```
@StoreRestResource(path = "videos")
public interface VideoStore extends Store<String> {
}
```

Ilustración 53. Implementación de streaming de videos

18.1.19. Administrador registra un médico

ID	Título	Puntos de historia	Prioridad
19	Administrador registra médico	2	Must
Descripción			
Como administrador quiero poder añadir nuevos médicos al sistema			
Criterios de aceptación			
- El administrador incluirá la información del médico correspondiente a su correo electrónico, nombre, apellidos y modelo de suscripción			

Tabla 21. Historia de usuario “Administrador registra un médico”

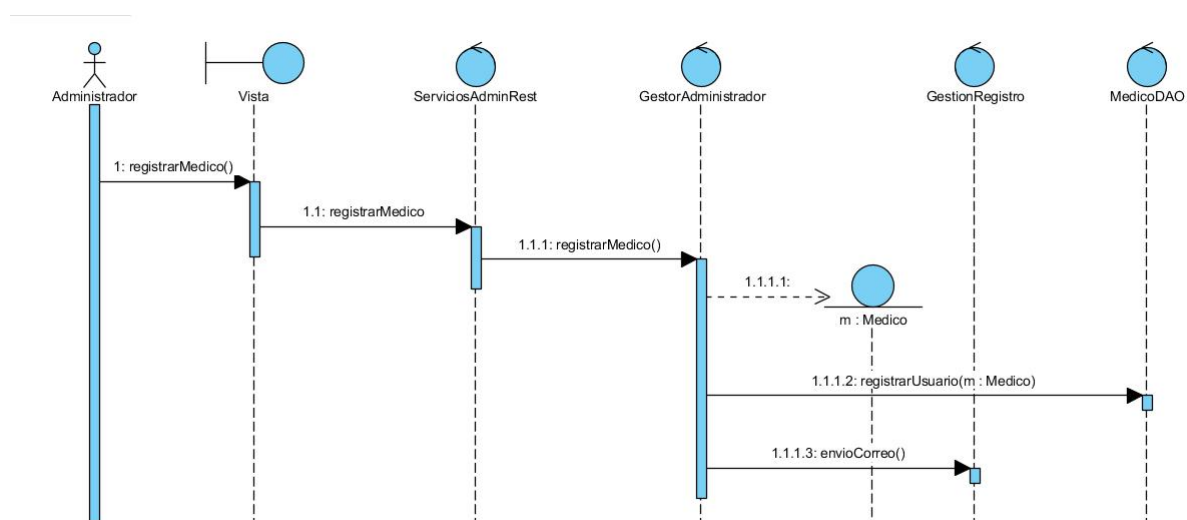


Ilustración 54. Diagrama de secuencia de la historia de usuario “Administrador registra un médico”

18.1.20. Administrador ve un médico

ID	Título	Puntos de historia	Prioridad
20	Administrador ve médico	2	Must
Descripción			
Como usuario quiero poder ver la información de un médico			
Criterios de aceptación			
- El administrador podrá acceder a la información de un médico a partir del correo electrónico de este.			

Tabla 22. Historia de usuario "Administrador ve un médico"

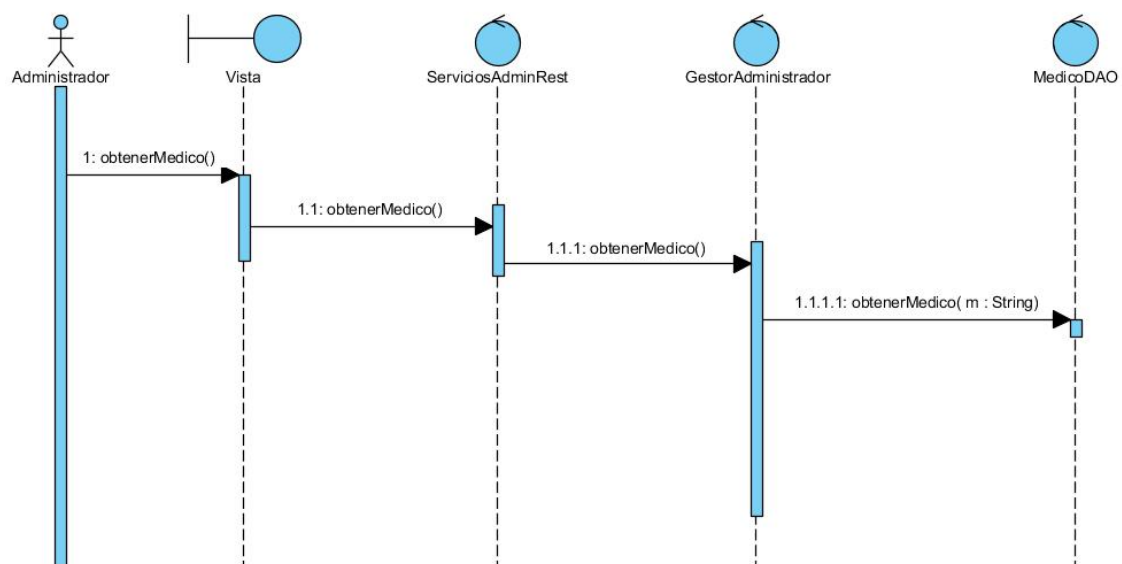


Ilustración 55. Diagrama de secuencia de la historia de usuario "Administrador ve un médico"

18.1.21. Administrador modifica un médico

ID	Título	Puntos de historia	Prioridad
21	Administrador modifica médico	2	Must
Descripción			
Como usuario quiero poder ver el video asignado a un ejercicio terapéutico			
Criterios de aceptación			
- El administrador podrá cambiar los campos de nombre, apellidos y modelo de suscripción.			

Tabla 23. Historia de usuario "Administrador modifica un médico"

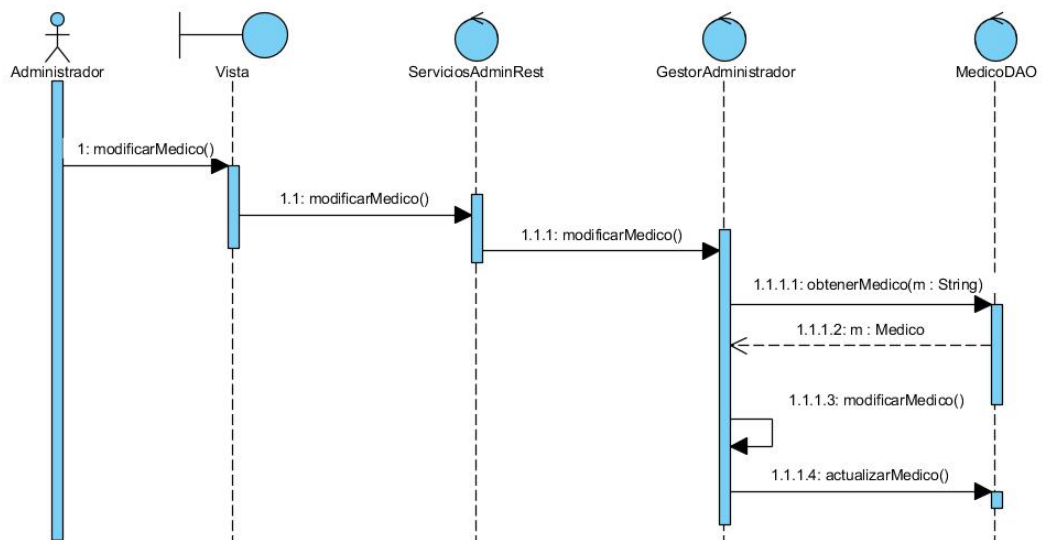


Ilustración 56. Diagrama de secuencia de la historia de usuario "Administrador modifica un médico"

18.1.22. Usuario configura perfil

ID	Título	Puntos de historia	Prioridad
22	Usuario modifica su perfil	2	Should
Descripción			
Como usuario quiero poder modificar mi perfil para poder actualizar datos de este			
Criterios de aceptación			
- Tanto médico como paciente podrán modificar su perfil			
- Se podrá cambiar la imagen de perfil del usuario			
- Se podrá cambiar la contraseña de la cuenta del usuario			

Tabla 24. Historia de usuario "Usuario configura perfil"

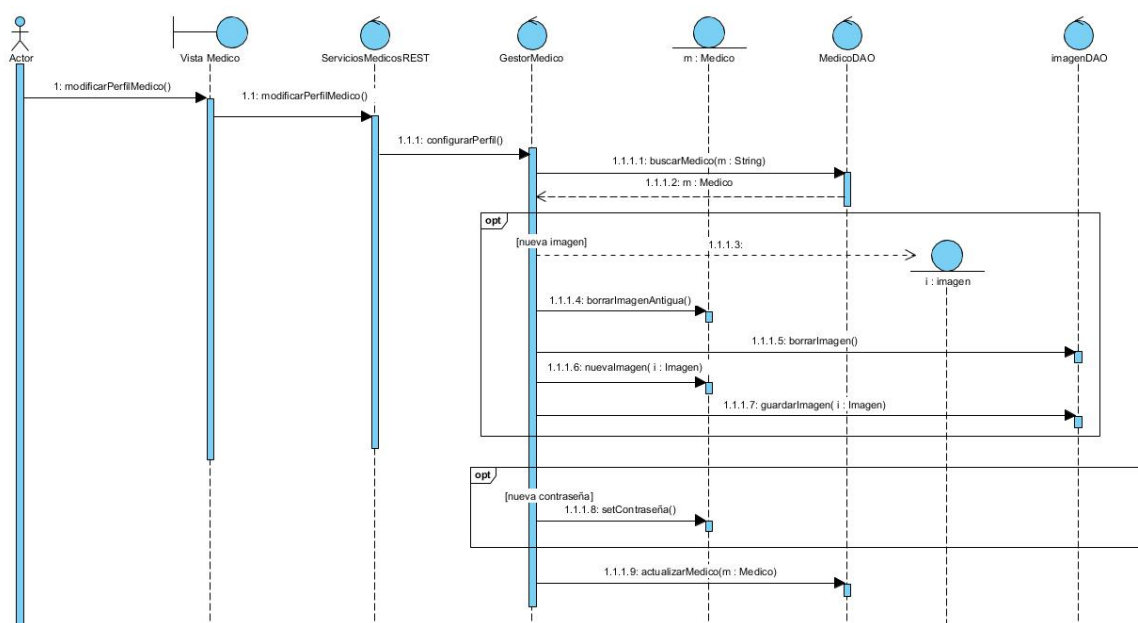


Ilustración 57. Diagrama de secuencia de la historia de usuario "Usuario configura perfil"

18.2. - End-points de la aplicación

- End-points accesibles por un usuario con rol médico

Recurso	Métodos HTTP	Función	URI	Código éxito	Código error
Ejercicio terapéutico	POST	crearEjercicioTerapéutico	/medicos/{medico}/ejercicios	201 - CREATED	409 - CONFLICT
	PUT	editarEjercicioTerapéutico	/medicos/{medico}/ejercicios	200 - OK	409 - CONFLICT
	GET	obtenerEjerciciosTerapéuticos	/medicos/{medico}/ejercicios	200 - OK	404 - NOT FOUND
	GET	obtenerEjercicioId	/medicos/{medico}/ejercicios/{id}	200 - OK	404 - NOT FOUND
Médico	GET	obtenerPerfilMedico	/medicos/{medico}	200 - OK	404 - NOT FOUND
	PUT	modificarPerfilMedico	/medicos/{medico}	200 - OK	409 - CONFLICT
Paciente	GET	obtenerListaPacientes	/medicos/{medico}/pacientes	200 - OK	404 - NOT FOUND
	POST	añadirPaciente	/medico/{medico}/pacientes	201 - CREATED	409 - CONFLICT
Terapia	POST	asignarTerapia	/medico/{medico}/terapias/{paciente}	201 - CREATED	409 - CONFLICT
	GET	obtenerTerapias	/medico/{medico}/terapias/{paciente}	200 - OK	404 - NOT FOUND
Historial médico	GET	obtenerHistorialMedico	/medico/{medico}/historial/{paciente}	200 - OK	404 - NOT FOUND
	POST	nuevoComentarioHistorialMedico	/medico/{medico}/historial/{paciente}	201 - CREATED	409 - CONFLICT

Tabla 25. End-points accesibles por un usuario con rol médico

- End-points accesibles por un usuario con rol paciente

Recurso	Métodos HTTP	Función	URI	Código éxito	Código error
Terapia	GET	obtenerTerapias	/pacientes/{paciente}/terapias	200 - OK	404 - NOT FOUND
	POST	realizarSesionTerapia	/pacientes/{paciente}/terapias/{terapia}	200 - OK	409 - CONFLICT
Paciente	GET	obtenerPerfilPaciente	/pacientes/{paciente}	200 - OK	404 - NOT FOUND
	PUT	modificarPerfilPaciente	/pacientes/{paciente}	200 - OK	409 - CONFLICT

Tabla 26. End-points accesibles por un usuario con rol paciente

- End-points accesibles por un usuario con rol administrador

Recurso	Métodos HTTP	Función	URI	Código éxito	Código error
Médico	POST	registroMedico	/administrador/medicos	201 - CREATED	409 - CONFLICT
	GET	obtenerMedico	/administrador/medicos/{medico}	200 - OK	404 - NOT FOUND
	PUT	modificarMedico	/administrador/medicos	200 - OK	409 - CONFLICT

Tabla 27. End-points accesibles por un usuario con rol administrador

- End-points accesibles por usuarios de múltiples roles

Recurso	Métodos HTTP	Función	URI	Código éxito	Código error
Usuario	POST	registroUsuario	/usuarios	201 - CREATED	409 - CONFLICT
	GET	identificarTokenRegistro	/usuarios/{token}	200 - OK	404 - NOT FOUND / 410 - GONE
	POST	identificarUsuarioLogin	/usuarios/login	200 - OK	401 - UNAUTHO RIZED
Mensaje terapia	GET	obtenerMensajesTerapia	/terapias/{usuario}/{terapia}/mensajes	200 - OK	404 - NOT FOUND
	POST	enviarMensaje	/usuarios/{usuario}/{terapia}/mensajes	201 - CREATED	409 - CONFLICT
	PUT	modificarMensaje	/usuarios/{usuario}/{terapia}/mensajes	200 - OK	409 - CONFLICT

Tabla 28. End-points accesibles por múltiples roles

19. Presupuesto

19.1. Coste de personal

Para el cálculo del coste debido a recursos humanos se va a utilizar los sueldos bases definidos en la disposición 3156 del BOE núm. 57 de 2018.

El equipo está integrado por un único miembro, con roles de analista-programador y diseñador web. De acuerdo a los sueldos bases obtenidos en la disposición mencionada previamente, ambos roles tienen acordado el mismo sueldo base anual (22.993,74 €) , por lo que no tendremos que hacer ninguna diferenciación en este sentido.

El cálculo del sueldo se va a realizar de manera mensual, en base al número de meses total que abarca el proyecto. De acuerdo a la planificación realizada en la sección 14 de este mismo documento, el proyecto comenzará el 4/2/2019 y concluirá el 17/6/2019, por lo que el número de meses de trabajo será 5.

El sueldo mensual será:

$$22.993,74 \text{ €} / 12 = 1916,145 \text{ €} \approx 1916,15 \text{ €}$$

Partiendo de esto podemos obtener el coste asociado a recursos humanos del proyecto:

Rol	Cantidad	Salario mensual	Meses de trabajo	Coste total
Analista programador / Diseñador web	1	1.916,15€	5	9.580,75€

Tabla 29. Coste de personal

Por lo tanto el coste de recursos humanos es de 9580, 75 €.

19.2. Coste de recursos tecnológicos

Para el coste de recursos tecnológicos se van a amortizar tanto los dispositivos hardware como las licencias software utilizadas.

Para ello se van a seguir las bases descritas por la Ley 27/2014 sobre el Impuesto de Sociedades, publicada en el BOE del 28 de noviembre de 2014.

Esta ley diferencia en el periodo de amortización correspondiente al software y al hardware, de tal modo que la amortización del software se hará en base a 6 años (2190 días) y la del hardware en base a 8 años (2920 días).

Algunas consideraciones importantes de acuerdo a los recursos son las siguientes:

- Se ha estipulado que el número de días por mes es 30.
- Solo se nombrarán las licencias software que son de pago. Durante el transcurso del proyecto se han empleado múltiples tecnologías con carácter gratuito con el objetivo de reducir el coste al mínimo.
- Las licencias que aparecen en la tabla con un coste diario de 0 € pertenecen a software que aunque es de pago, se ha accedido a él de forma gratuita mediante una licencia proporcionada por la Universidad de Jaén.

Hardware	Cantidad	Coste	Periodo de amortización	Amortización/día	Amortización para 5 meses de trabajo
Portátil de desarrollo	1	550,00 €	2920	0,19 €	28,25 €
Visual Paradigm Standard	1	0,00 €	2190	0,00 €	0,00 €
Axure RP 8	1	0,00 €	2190	0,00 €	0,00 €
Microsoft Windows 10	1	259,00 €	2190	0,12 €	17,74 €
Coste amortización	45,99 €				

Tabla 30. Coste recursos tecnológicos

19.3. Importe final

El importe final del proyecto vendrá dado por la suma de los costes de personal y de recursos tecnológicos calculada previamente:

Recurso	Coste
Recursos humanos	9.580,75€
Recursos tecnológicos	45,99 €
Coste total	9.626,74€

Tabla 31. Coste total

Además, sobre este coste total se estipula una ganancia de beneficio igual al 20% de este. Por lo tanto el importe final será:

Coste total	Porcentaje de beneficio	Beneficio
9.626,74€	20 %	1.925,35€
Importe final	11.552,09€	

Tabla 32. Importe final

20. Estudios con entidad propia

20.1. Modelo de negocio

Con el objetivo de decidir el modelo de negocio que seguirá nuestra aplicación se ha realizado un estudio de mercado para analizar a la competencia y tratar de tomar un plan de acción inspirado en las políticas generales del sector y al mismo tiempo ofreciendo un plan de negocio de cara a los futuros usuarios de la aplicación que pueda resultar satisfactorio.

- Clinni: ofrece dos versiones, una gratuita con capacidad de pacientes limitada y otra con un coste de 19,99 € al mes (para médicos pertenecientes a algunos colegios profesionales este precio se reduce a 9,99 €), con capacidad ilimitada de pacientes y otras mejoras adicionales a la aplicación.
- NetClinicas: ofrece 3 planes, el primero es una versión pensada para un uso individual, con un coste de compra 240 €. La segunda versión tiene un coste de 350 €, enfocada a un entorno de clínica con varios médicos y también sigue un modelo de compra única. Por último, hay una versión de suscripción mensual con un coste de 36 €.
- Bonoom: ofrece un único plan de suscripción con un coste de 24,95 € al mes.
- Clinic-cloud: ofrece tres planes, uno gratuito con funcionalidad limitada y una capacidad de gestión de 300 pacientes, y dos de pago basados en un sistema de suscripción mensual, con un coste de 29,65 € y 57,48 € al mes.

En base a los datos encontrados, se ha decidido establecer un modelo de negocio que ofrezca dos tipos de versiones para los especialistas médicos. El primero se trata de un modelo gratuito, que permite a los médicos trabajar con un máximo de pacientes. El segundo se ajustaría a un modelo de suscripción mensual, ofreciendo capacidad ilimitada.

Puesto que estamos en el supuesto de ser una empresa nueva en el sector, una táctica que se plantea es la de ofrecer un precio menor al del resto. Es por ello que la idea de una suscripción mensual de 9,99 € al mes, similar a la de Clinni, la cual es la más baja de todas las estudiadas y sólo en condiciones especiales (colegiado), parece interesante. Por otro lado, debemos de estipular cuál será el número de pacientes máximo para la versión gratuita. En el caso de clinic-cloud, se marca en 300, aunque es una aplicación con un alcance mayor de la nuestra y con un coste mensual bastante alto. Es por ello que se plantea reducir esa marca a 100, que al ser un número inferior junto al coste reducido de la suscripción podría contribuir a

que una gran parte de los médicos del sistema apuesten por la versión premium que ofrecemos.

De cara a los pacientes, el acceso a la aplicación sería siempre gratuito, aunque para poder acceder a la cuenta deberán de tener un correspondiente invitación de un médico registrado en el sistema.

20.2. Despliegue de la aplicación

Para el despliegue de la aplicación se ha decidido utilizar Heroku, una plataforma como servicio que se ha utilizado tanto para alojar el frontend como el backend del sistema.



Ilustración 58. Heroku

El motivo de elección de esta plataforma se ha debido principalmente a que es sencilla de utilizar y ofrece un modelo gratuito básico que nos permite trabajar dentro de un entorno de producción. Además nos proporciona conexiones HTTPS a nuestro servidor y persistencia en una base de datos Postgres de forma gratuita.

De momento se ha escogido la configuración más básica, con una sola instancia del servidor, ya que para los fines de testeo del prototipo desarrollado durante el proyecto se considera que sería suficiente.

La URL base de la API REST del servidor es tfg-jabm.herokuapp.com. La página de login del cliente web estaría en la URL: gestionterapias.herokuapp.com/login/login.html.

Para la realización de pruebas y debido a que el acceso al sistema es restringido, se han definido tres usuarios básicos para testear las diferentes funcionalidades del sistema.

- | | |
|-------------------------------|----------------------------------|
| - Usuario: medico@gmail.com | Clave: <i>claveMedico</i> |
| - Usuario: paciente@gmail.com | Clave: <i>clavePaciente</i> |
| - Usuario: admin@gmail.com | Clave: <i>claveAdministrador</i> |

21. Conclusiones y trabajos futuros

La realización de este trabajo de fin de grado ha supuesto un desafío interesante, en el cual he puesto a prueba los conocimientos adquiridos durante el grado aplicándolos al contexto de un proceso de ingeniería del software completo.

En cuanto a los resultados obtenidos en el ámbito tecnológico, el sistema desarrollado considero que ha conseguido satisfacer los requisitos acordados. Es cierto que a partir de aquí podemos añadir todas las nuevas funcionalidades que queramos. Algunas podrían ser las siguientes:

- Extensión del campo de la medicina en el que trabajamos, incluyendo nuevas funcionalidades no solo para el ámbito de los ejercicios terapéuticos y de rehabilitación sino también para fármacos o medicamentos.
- Ofrecer funcionalidades centradas en la gestión de clínicas tradicional, ya sea mediante la gestión de citas médicas, aspectos de presupuesto como pago de nóminas o cobro a los pacientes.
- Sistema de pago dentro del sistema para los médicos que utilicen el modelo premium de negocio definido.
- Autenticación OAuth2 que permita a los usuarios utilizar sus credenciales de otra plataforma.
- Sistema de notificaciones, por ejemplo mediante correo electrónico, que permita a los usuarios ser notificados ante diferentes eventos: un nuevo mensaje en el chat de una terapia particular, cambios en el ejercicio que forma parte de una terapia, etc.

Además, se podría considerar la implementación de un cliente nuevo, nativo para dispositivos móviles. Debido a la arquitectura de nuestro sistema, la inclusión de este cliente en el ecosistema de la aplicación se haría de forma muy limpia, ya que solo necesitaría acceder a los servicios que ofrece nuestro backend.

De cara al despliegue del sistema, actualmente contamos con una versión bastante limitada con fines principalmente demostrativos. En caso de querer realizar un despliegue con unos objetivos más serios, sería necesario estudiar cómo deberá de escalar el sistema ante la llegada de nuevos usuarios. Además, se podría plantear utilizar servidores adicionales especializados exclusivamente en el streaming del contenido multimedia de nuestro sistema.

A nivel personal, considero que la realización de este proyecto me ha aportado nuevos conocimientos y experiencia de cara a afrontar proyectos futuros, tanto a nivel de gestión de proyectos como a la hora de utilizar las diferentes tecnologías empleadas.