



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería Informática

Aplicación para la gestión de reseñas y consultas de
intervenciones quirúrgicas

Trabajo Fin de Grado

Grado en Ingeniería Informática

AUTOR/A: Ruiz Carrillo, Joel

Tutor/a: Canós Cerdá, José Hilario

CURSO ACADÉMICO: 2022/2023

Resumen

El presente proyecto nace para ayudar a aquellas personas que necesiten encontrar o pretendan ayudar en la búsqueda de información sobre intervenciones quirúrgicas, ya que actualmente los usuarios deben depender de diversas comunidades segregadas para encontrar todos los conocimientos que requieran para tomar una decisión acertada.

Por consiguiente, se facilita una aplicación web donde usuarios puedan encontrar toda la información que busquen gracias a publicaciones aportadas por otros usuarios; los usuarios y centros tendrán la posibilidad de formar parte de la comunidad creando una cuenta para interactuar y comunicarse entre ellos.

El proyecto se ha desarrollado siguiendo metodologías ágiles, obteniendo así una versión funcional al acabar cada *sprint*. Con este propósito en mente y buscando una interfaz moderna e intuitiva, las tecnologías empleadas han sido React para la parte *front-end* de la aplicación, Django para el *back-end*, y *Visual Studio Code* como entorno de desarrollo.

Palabras clave: Intervención quirúrgica, comunidad, comunicación, aplicación web, metodologías.



Abstract

The present project was created to help those people who need to find or intend to help in the search for information about surgical interventions, since currently users must depend on various segregated communities to find all the knowledge they require to make the right decision.

Consequently, a web application is provided, where users can find all the information they are looking for thanks to publications contributed by other users; both users and centers will have the possibility of becoming part of the community by creating an account to interact and communicate with each other.

The project has been developed following agile methodologies, thus obtaining a fully functional version at the end of each sprint. With this purpose in mind and looking for a modern and intuitive interface, the chosen technologies have been React for the front-end of the application, Django for the back-end, and Visual Studio Code as the development environment.

Keywords: Surgical intervention, community, communication, web application, methodologies.

Índice de contenidos

1.	Introducción	9
1.1	Motivación	9
1.2	Objetivos.....	10
1.3	Estructura de la Memoria.....	10
2.	Estado del Arte	13
2.1	Aplicaciones similares.....	13
2.2	Páginas propias de los centros.....	14
2.3	Google Reviews.....	16
2.4	Conclusiones	16
3.	Metodología	19
4.	Análisis del problema.....	21
4.1	Lean Canvas	21
4.2	Brainstorming	22
4.3	Análisis DAFO.....	26
4.4	Diagrama de contexto.....	27
4.5	Características del sistema	28
4.6	Diagrama de casos de uso	29
4.7	Modelo de dominio	34
4.9	Prototipado	35
4.10	Análisis del marco legal y ético.....	37
5.	Diseño de la solución	39
5.1	Arquitectura del sistema	39
5.2	Diseño de la base de datos	40
5.3	Patrones aplicados.....	42
6.	Desarrollo de la solución	45
6.1	Tecnologías utilizadas	45
6.1.1	HTML	45
6.1.2	CSS.....	45
6.1.3	JavaScript	45
6.1.4	React	46
6.1.5	Python	46
6.1.6	Django.....	47



6.1.7 Postman	48
6.1.8 Visual Studio Code	48
6.1.9 Worki.....	48
6.1.10 Figma	48
6.2 Proceso de desarrollo.....	49
6.2.1 Sprint 0.....	49
6.2.2 Sprint 1	51
6.2.3 Sprint 2	53
6.2.4 Sprint 3	55
7. Pruebas	59
7.1 Pruebas API.....	59
7.2 Pruebas de aceptación	59
7.3 Cuestionario de usabilidad.....	59
8. Conclusiones	63
8.1 Relación del trabajo desarrollado con los estudios cursados	64
9. Referencias.....	65
Anexo A. Objetivos de Desarrollo Sostenible.....	67
Anexo B. Casos de uso identificados	69
Anexo C. Listado de pruebas de aceptación.....	79
Anexo D. Formulario escala de usabilidad	89
Anexo E. Manual de usuario	93

Índice de figuras

Figura 1: página web de Doctoralia.	13
Figura 2: página web TrustPilot.	14
Figura 3: ejemplo 1 de página web propia de una clínica.....	15
Figura 4: ejemplo 2 de página web propia de una clínica.....	15
Figura 5: modelo Lean Canvas aplicado al proyecto.	22
Figura 6: ejemplos de diapositivas usadas en la presentación.....	23
Figura 7: matriz DAFO aplicada al proyecto.	26
Figura 8: actores identificados.....	28
Figura 9: características del sistema.....	29
Figura 10: diagrama de casos de uso para la gestión de usuarios.	30
Figura 11: diagrama de casos de uso para la gestión de publicaciones.	31
Figura 12: diagrama de casos de uso para la gestión de clínicas.....	32
Figura 13: diagrama de casos de uso para la moderación.....	33
Figura 14: diagrama de casos de uso para los feeds.....	34
Figura 15: modelo de domino generado.	35
Figura 16: wireframe formulario creación de una publicación.	36
Figura 17: wireframe feed de publicaciones.	37
Figura 18: representación arquitectura dividida en tres capas.....	39
Figura 19: diseño logrado de la base de datos.....	41
Figura 20: estructura del patrón Singleton.....	42
Figura 21: código relevante del patrón Singleton.....	43
Figura 22: UTs designadas al sprint 0.	49
Figura 23: gráfica burndown sprint 0.	50
Figura 24: diagrama de Flujo Acumulado sprint 0.	50
Figura 25: gráfica estimaciones sprint 0.....	51
Figura 26: UTs designadas al sprint 1.	51
Figura 27: diagrama de Flujo Acumulado sprint 1.	52
Figura 28: gráfica burndown sprint 1.....	52
Figura 29: gráfica estimaciones sprint 1.....	53
Figura 30: UTs designadas al sprint 2.	53
Figura 31: diagrama de Flujo Acumulado sprint 2.	54
Figura 32: gráfica burndown sprint 2.....	54
Figura 33: gráfica estimaciones sprint 2.....	55
Figura 34: UTs designadas al sprint 3.....	55
Figura 35: diagrama de Flujo Acumulado sprint 3.....	56
Figura 36: gráfica burndown sprint 3.....	56
Figura 37: gráfica estimaciones sprint 3.....	57
Figura 38: página de inicio.....	57
Figura 39: funcionalidades dentro de una publicación.....	58
Figura 40: umbrales de la System Usability Scale.....	61
Figura 41: percentiles en la System Usability Scale.....	61
Figura 42: estructura de nodos,	79
Figura 43: página de inicio.....	93



Figura 44: formulario de registro.....	94
Figura 45: términos y condiciones del servicio.	94
Figura 46: mensaje de verificación de correo.	95
Figura 47: página de inicio al haberse identificado.	95
Figura 48: formulario de publicación de una reseña.	96
Figura 49: validación de creación de una clínica.....	96
Figura 50: formulario de publicación de una consulta.	97
Figura 51: formulario de publicación de una clínica.....	98
Figura 52: opción de añadir personal a la clínica.	98
Figura 53: formulario de inscripción de personal.	99
Figura 54: perfil de una clínica.....	100
Figura 55: formulario de publicación de una noticia.	100
Figura 56: feed de publicaciones de tipo reseña.	101
Figura 57: feed de publicaciones de tipo consulta.	101
Figura 58: notificaciones del usuario.	102
Figura 59: funcionalidades dentro de una publicación.	102
Figura 60: reportar contenido.	103
Figura 61: vista del perfil de un usuario.	103
Figura 62: chat privado con otro usuario.	104
Figura 63: vista del perfil personal.	104
Figura 64: ventana de datos personales.	105
Figura 65: página informativa sobre OperRating.	105
Figura 66: opciones del formulario de contacto.	106
Figura 67: datos del formulario de contacto.	106

Índice de tablas

Tabla 1: características encontradas en aplicaciones similares.	17
Tabla 2: ideas resultantes del ejercicio de brainstorming.	26
Tabla 3: CU01, registro de usuario	30
Tabla 4: CU10, visualizar publicación.	31
Tabla 5: CU18 visualizar doctros.	32
Tabla 6: CU07, reportar usuario.	33
Tabla 7: CU13: Seleccionar clínica desde el mapa.	34
Tabla 8: resultados obtenidos en la System Usability Scale.	61
Tabla 9: características de OperRating en comparación con la competencia.....	64
Tabla 10: CU02, identificación.....	69
Tabla 11: CU03, enviar mensaje.....	69
Tabla 12: CU04, cambiar datos personales.	69
Tabla 13: CU05, darse de baja.....	70
Tabla 14: CU06, acceder a mi perfil.	70
Tabla 15: CU09, salir de la sesión actual.....	70
Tabla 16: CU20, acceder al perfil de un usuario.	71
Tabla 17: CU34, registro de propietario.	71
Tabla 18: CU34, visualizar publicaciones guardadas.....	71
Tabla 19: CU11, compartir publicación.	72
Tabla 20: CU12, crear reseña.	72
Tabla 21: CU14, escribir comentario.	72
Tabla 22: CU15, guardar publicación.....	73
Tabla 23: CU16, notificaciones sobre publicación.	73
Tabla 24: CU26, borrar publicación.....	73
Tabla 25: CU35, crear consulta.	74
Tabla 26: CU37, eliminar comentario.	74
Tabla 27: CU39, acceder al perfil de una clínica.....	74
Tabla 28: CU19, crear clínica.....	75
Tabla 29: CU22, subir noticia.....	75
Tabla 30: CU23, eliminar noticia.....	75
Tabla 31: CU27, visualizar localización de la clínica	75
Tabla 32: CU29, usar filtros de búsqueda.....	76
Tabla 33: CU40, identificarse como propietario.....	76
Tabla 34: CU42, editar clínica.....	76
Tabla 35: CU08, bloquear usuario.	77
Tabla 36: CU38, moderar comentario.	77
Tabla 37: CU41, verificar propietario.	77
Tabla 38: CU21, consultar términos de servicio.....	77
Tabla 39: CU28, visualizar feed.	78
Tabla 40: CU31, visualizar mapa.....	78
Tabla 41: CU32, previsualizar publicación.....	78
Tabla 42: pruebas de aceptación nodo Clínica.	80
Tabla 43: pruebas de aceptación nodo Noticia.	81

Tabla 44: pruebas de aceptación nodo Doctor.....	81
Tabla 45: pruebas de aceptación nodo Página inicial.....	82
Tabla 46: pruebas de aceptación nodo Feed.....	82
Tabla 47: pruebas de aceptación nodo Filtros.	83
Tabla 48: pruebas de aceptación nodo Mapa.	83
Tabla 49: pruebas de aceptación nodo Tickets.	84
Tabla 50: pruebas de aceptación nodo Usuario.	85
Tabla 51: pruebas de aceptación nodo Chat.	86
Tabla 52: pruebas de aceptación nodo Notificación.	86
Tabla 53: pruebas de aceptación nodo Publicación.....	87
Tabla 54: pruebas de aceptación nodo Imagen.....	87
Tabla 55: pruebas de aceptación nodo Comentario.....	88
Tabla 56: pruebas de aceptación nodo Compartir.....	88

1. Introducción

En la actualidad, no es poco común tener que someterse a algún tipo de intervención quirúrgica. Los motivos pueden ser muchos, ya sea por el tratamiento de una enfermedad o lesión, la prevención de una de éstas o para mejorar la estética personal. Encontrar, en consecuencia, un centro médico que se ajuste a las necesidades particulares de cada individuo puede resultar una tarea ardua, ya que los centros suelen proclamar ser los mejores en su campo, y no siempre ofrecen toda la información que el paciente desea.

De este hecho surge la cuestión de dónde poder encontrar datos fiables al respecto, sin tener que ir cuestionando centro por centro, mediante citas concertadas o consultas por correo electrónico, sino mediante una comunidad que se comunique de forma eficiente y efectiva, compartiendo información útil y real. Para ello, existen ya varias comunidades segregadas, de las cuales el paciente depende para obtener la información que necesite.

En el presente Trabajo de Fin de Grado, se plantea crear una solución para este problema mediante el desarrollo de una aplicación web que cree una comunidad que facilite el acceso a la información para sus usuarios, brindándoles las herramientas necesarias para ello.

A lo largo de esta memoria, se presentan las fases de análisis, diseño y desarrollo de la aplicación, así como la metodología a seguir. A partir de este punto, la aplicación producto será denominada como **OperRating**, nombre nacido del inglés *Operations* y *Rating*, es decir, Operaciones y Valoración.

1.1 Motivación

La motivación del proyecto nace de la necesidad de encontrar un lugar donde poder compartir experiencias y que dicha información resulte interesante para el público objetivo.

En mi experiencia personal a la hora de buscar datos al respecto para intervenciones quirúrgicas, no he logrado encontrar una fuente de información fiable en España donde contactar con la persona que haya publicado la información fácilmente. Esto me llevó a realizar dicha búsqueda de formas más convencionales, como por ejemplo mediante contactos con gente de experiencia similar fuera de las redes sociales.

Yendo más allá, crear una aplicación desde cero, pasando por todas las etapas que eso conlleva, resulta ser un ejercicio de gran aprendizaje.

Por tanto, y dado que no existe una comunidad hispanohablante definida para este ámbito, considero conveniente el desarrollo de una herramienta que pueda agilizar la interacción entre la gente interesada.

1.2 Objetivos

El desarrollo de esta plataforma web tiene como principal propósito crear el mínimo producto viable (MVP¹) de una comunidad de apoyo entre usuarios, donde poder realizar todas las funciones necesarias para tener una comunicación efectiva.

Para lograr este objetivo se abordan tres puntos clave:

Acceso a información: Para lograr nuestro propósito es esencial contar con la información adecuada, la cual será brindada por los mismos usuarios. Estos datos deben ser fácilmente obtenibles, y se lograrán extraer mediante diversos métodos de búsqueda de información. También es necesario contar con cierta moderación por la parte administrativa para comprobar la veracidad de algunos datos proporcionados.

Incentivación hacia la mejora de los centros: Al contar con una plataforma donde los profesionales médicos tienen su reputación en juego, se crea cierto estímulo que promueve su competitividad, lo cual conlleva una mejor satisfacción del usuario.

Respeto de la privacidad: Para brindar facilidades en la búsqueda de información, será posible el uso de la geolocalización del usuario, siempre con el permiso del usuario, de una forma fiable sin uso de rastreadores maliciosos sin recoger más datos de los necesarios.

1.3 Estructura de la Memoria

El presente documento va a contar con la siguiente estructura:

Durante el capítulo dos se hará un repaso sobre el estado del arte, en el cual observaremos las principales alternativas actuales a nuestro producto, reflexionando sobre los puntos débiles y fuertes de cada una.

Seguidamente, en el capítulo tres comentaremos la metodología empleada para el desarrollo del proyecto, sus conceptos principales y cómo se distribuirá la carga de trabajo.

A lo largo del capítulo cuarto se mostrarán las actividades llevadas a cabo durante la especificación inicial de requisitos, llevando un seguimiento detallado de cada una de las secciones desarrolladas.

En el capítulo quinto, se mostrará el diseño de la base de datos, el aspecto y arquitectura de la aplicación, así como los patrones aplicados y el porqué de ellos.

Moviéndonos hacia el cuarto capítulo, encontraremos inicialmente las tecnologías escogidas para cada parte del desarrollo del proyecto. Prosiguiendo con el mismo, se comentará detalladamente el trabajo realizado durante cada *sprint*, mostrando las gráficas de progreso de cada uno de estos.

Enlazando con el sexto capítulo se hará una comprobación final de las Pruebas de Aceptación y una reflexión sobre los resultados de Cuestionarios de Usabilidad rellenos por usuarios reales.

¹ Un MVP se refiere a aquella versión de un producto que tenga las funcionalidades necesarias para ser usada por usuarios cuanto antes.

Para concluir, el séptimo capítulo contendrá las conclusiones del proyecto, donde se mencionarán también posibles mejoras a futuro.

2. Estado del Arte

Existen diferentes alternativas donde la gente interesada en recibir algún tipo de intervención quirúrgica puede informarse sobre estas. La mayoría de estos sitios cubren un sector muy amplio, por lo que ofrecen opciones genéricas, o bien son demasiado especializadas y específicas. A continuación, se detallan las alternativas más relevantes.

2.1 Aplicaciones similares

Doctoralia: Doctoralia² (Figura 1) es un sitio web destinado exclusivamente a facilitar la búsqueda de información sobre profesionales de la salud y servicios médicos. Entre todas las alternativas que mencionaremos en este capítulo, es la que más se acerca a los objetivos de nuestro proyecto.

Entre sus principales características contamos con la búsqueda de profesionales según una serie de filtros. Los perfiles de dichos profesionales cuentan con reseñas y preguntas de los usuarios interesados en ellos, de forma similar a lo que pretendemos realizar en este proyecto. También cuentan con una agenda de citas en línea, la cual agiliza el proceso de contacto con el profesional.

Doctoralia se centra en que los propios profesionales sean quienes respondan ante los usuarios. Por este motivo, no existe interacción entre usuarios más allá de leer las reseñas que uno haya publicado al encontrarlas en el perfil de algún profesional.



Figura 1: página web de Doctoralia.

² <https://www.doctoralia.es/>

TrustPilot: TrustPilot³ (figura 2) es una plataforma en línea que permite a los usuarios dejar reseñas y opiniones sobre empresas y servicios. Al contrario que Doctoralia, la cual se centra en intervenciones médicas, TrustPilot es usada para todo tipo de establecimientos y comercios. Sin embargo, siguen compartiendo muchas características, como, por ejemplo, el sistema de reseñas y búsqueda mediante filtros.

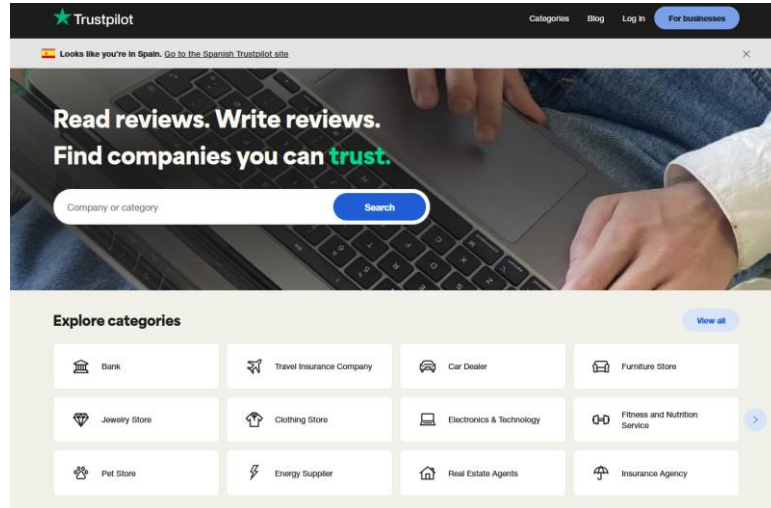


Figura 2: página web TrustPilot.

2.2 Páginas propias de los centros

Habitualmente, cada centro médico cuenta con su propia página donde consultar directamente cualquier duda. Sin embargo, el uso de una aplicación externa es preferible en muchas ocasiones, principalmente debido al sentimiento de comunidad que ofrecen estas alternativas, ya que brindan información más detallada ofrecida por parte de otros usuarios. Si bien existen centros que cuenten con este tipo de comunidad, no suele ser lo habitual.

Un ejemplo que ilustra lo previamente dicho es la página de la clínica del DR José Lara Montenegro⁴. En ella se ofrece todo tipo de información sobre las intervenciones que realiza, pero no se muestra una interacción directa con pacientes que hayan tenido alguna experiencia con él (figura 3).

³ <https://www.trustpilot.com/>

⁴ <https://drlara.es/>



Figura 3: ejemplo 1 de página web propia de una clínica.

Otro ejemplo puede ser la página de la clínica del Dr. Moscatiello⁵. (figura 4), el cual reside en Barcelona y Andorra. De forma similar al anterior ejemplo, tampoco contamos con las opiniones directas de sus clientes, sino con un formulario de contacto donde dejar nuestras dudas.

Figura 4: ejemplo 2 de página web propia de una clínica.

⁵ <https://drmoscatiello.com/>

2.3 Google Reviews

Google Reviews⁶ es una plataforma extensamente popular usada para obtener opiniones sobre empresas y servicios. No obstante, al tratarse de una plataforma más genérica no cuenta con un enfoque apropiado para encontrar profesionales según criterios dados. También cabe añadir que, a menudo, las valoraciones escritas por usuarios no se han verificado, lo que resulta en información poco fiable.

2.4 Conclusiones

Observando las alternativas existentes podemos concluir que, aunque encontremos aplicaciones con funcionalidades similares, no encontramos una que nos ofrezca todo lo que buscamos crear en este proyecto, hecho que deja vía libre para que este proyecto alcance el éxito y tenga cabida en el mercado.

Por ejemplo, en el caso de Doctoralia, a pesar de sus numerosas facultades positivas, no contamos con una comunicación directa entre usuarios.

Por el contrario, en Truspilot si podemos encontrar más fácilmente comunicación con otros usuarios, pero no disponemos de elementos más visuales como en Doctoralia o Google Reviews, los cuales nos ofrecen un mapa interactivo que muestra diversos marcadores con las opciones que más se ajustan a la búsqueda del usuario.

Debido a estas debilidades y fortalezas, se considera oportuno la creación de una aplicación web que englobe todas las características necesarias para crear una comunidad que cumpla todos los objetivos mencionados en el primer capítulo.

Toda esta información es ilustrada mediante la tabla 1, que simplifica las diferencias entre las posibilidades. Por otra parte, durante el capítulo octavo, el cual contiene las conclusiones del proyecto, volveremos a mencionar la tabla, incluyendo esta vez el resultado final obtenido con tal de contrastar las características conseguidas.

Productos -----	Google Reviews	Doctoralia	TrustPilot	Clinicas
Características				
Comentarios	✓	✓	✓	X
Mapa de las clínicas	X	✓	X	X
Chat privado	X	X	✓	X
Notificaciones	X	✓	✓	✓
Publicar consulta	X	✓	X	X
Publicar reseña	✓	✓	✓	X
Dedicado total- mente a cirujías	X	X	X	✓
Compartir	✓	✓	✓	X
Noticias de una clínica	X	X	X	✓

⁶ <https://business.google.com/reviews>

Tabla 1: características encontradas en aplicaciones similares.

Frente a los datos expuestos, nos encontramos ante la oportunidad de crear una prueba de concepto de un espacio que tenga en cuenta las necesidades reales de los usuarios, y cuyo producto final cumpla con las características mencionadas en la Tabla 1.

3. Metodología

En el contexto de desarrollo de aplicaciones software, una metodología se refiere al método usado para estructurar el trabajo, administrar el proceso y gestionar la planificación con tal de trabajar en equipo más eficientemente. Existe una amplia gama de metodologías, las cuales se encuentran divididas según sean **ágiles** o **tradicionales**. La metodología empleada para llevar a cabo este proyecto se trata de una ubicada en el ámbito de las metodologías ágiles, concretamente *Tune-Up process* [1].

A diferencia de otras metodologías, las ágiles se caracterizan por su entrega constante de valor, capacidad de resolución de problemas y flexibilidad. Esto último quiere decir que, en contraste a las metodologías tradicionales, como puede ser el modelo en cascada [2], el proceso en un proyecto que sigue las metodologías ágiles no viene tan definido inicialmente respecto a uno elaborado siguiendo una metodología tradicional, ya que está abierto a posibles cambios en los requisitos en cualquier momento.

Respecto a la resolución de problemas, en un enfoque tradicional se debería avisar al gerente del proyecto, mientras que en las metodologías ágiles nos encontramos con un equipo autogestionado capaz de tomar sus propias decisiones, siempre siendo dirigidos por el **Product Owner**.

Las metodologías ágiles se caracterizan principalmente por su aporte al final de cada *sprint*, el cual va añadiendo valor funcional al producto al final de cada iteración. Este concepto no existe en las metodologías tradicionales, ya que su enfoque se basa en un entregable final al terminar el proyecto. Realizar el trabajo en varios entregables resulta conveniente a la hora de determinar si el producto es similar a lo que el cliente espera recibir, y facilita la flexibilidad previamente descrita.

Una vez comentadas las características clave de las metodologías ágiles, explicaremos los conceptos principales de la metodología ágil *Tune-Up process*:

-Unidades de Trabajo (UT): Las Unidades de Trabajo o UT son piezas de trabajo solicitadas por el *Product Owner*. Son elementos clave para organizar el proyecto en las metodologías ágiles, ya que cada UT aporta valor al negocio.

-Pruebas de aceptación (PA): Las Pruebas de Aceptación se llevan a cabo durante la actividad de Pruebas de una Unidad de Trabajo y al finalizar el proyecto, pero son definidas al inicio del *sprint*. En ellas se especifica el comportamiento concreto que debería seguir la Unidad de Trabajo, ya que se basan en los requisitos definidos por el cliente.

-Scrum Master: El Scrum Master es uno de los roles más relevantes dentro de la metodología Scrum. Éste se encarga de facilitar y garantizar el correcto funcionamiento del proceso, y de la correcta aplicación de metodología. Sin embargo, ya que se trata de un proyecto personal, el mismo desarrollador asume el rol de Scrum Master.

-Product Owner: El *Product Owner* representa los intereses de los stakeholders y futuros usuarios, y es quien define y prioriza los elementos del backlog teniendo en cuenta dichos intereses. También trabaja junto al equipo de desarrollo para garantizar una comunicación efectiva con el cliente, ayudando a tomar decisiones sobre las funcionalidades a implementar.



Como se mencionó anteriormente, el desarrollador asume todos los roles al tratarse de una aplicación desarrollada por una persona.

-*Sprint*: Consiste en una iteración del producto, durante la cual se trabaja en las Unidades de Trabajo seleccionadas previamente. No suelen tener una duración mayor a cuatro semanas. El concepto de *sprint* ayuda al seguimiento en el corto plazo e introduce una disciplina y ritmo de trabajo que incluye planificación, seguimiento, revisión y mejora continua. Al final de cada *sprint*, se hace una entrega funcional del producto.

-*Backlog*: El backlog de un producto contiene Unidades de Trabajo, ordenadas por prioridad, que aún no han sido incluidas en un *sprint*. Habitualmente está en constante actualización, siendo añadidos cambios sobre el producto, los cuales sirven como guía para los desarrolladores. Como se mencionó anteriormente, el *Product Owner* es el responsable de gestionar el Backlog.

-*Workflow*: El *Workflow* se refiere al flujo de trabajo o proceso que se sigue durante una UT. Para este proyecto existen dos tipos de *Workflow*: desarrollo con *sprints* y Tarea. El *Workflow* Tarea fue empleado durante el *sprint* 0, durante el cual se especificaron los requisitos que mencionaremos más adelante, mientras que el *Workflow* de desarrollo con *sprints* es usado en todas las demás Unidades de Trabajo.

-*Tablero Kanban*: El tablero *kanban* es el centro de operaciones de los miembros del equipo. Es utilizado para gestionar y visualizar de forma sencilla el trabajo durante el proyecto. Cada vez que el usuario desarrollador finalice alguna UT o en general quede disponible deberá acceder al tablero *kanban* para elegir en qué UT trabajar a continuación.

-*Colaboradores*: Al tratarse de un trabajo individual, el equipo de trabajo está formado por una persona, por lo que los distintos roles mencionados recaen sobre ella.

En cuanto al desarrollo de la aplicación web, nos encontramos con una división en tres *sprints*, más uno inicial de elicitación (*sprint* 0). El transcurso de cada *sprint* será descrito en detalle en capítulos próximos; primeramente, comentaremos en el cuarto capítulo la primera iteración, dedicada a la elicitación, debido a los numerosos datos recopilados a comentar. En cuanto a los *sprints* aplicados a la programación, serán agrupados en el capítulo sexto, donde revisaremos los diagramas, Unidades de Trabajo y más datos resultantes al finalizar cada *sprint*. El rango de duración de cada *sprint* es variable, ya que en la mayoría de los casos contamos con cuatro sem

4. Análisis del problema

Durante el primer *sprint* del proyecto, el *sprint 0*, se ha llevado a cabo la fase de elicitación de requisitos, lo cual incluye un ejercicio de *brainstorming*, contextualizar más en profundidad el dominio de la aplicación, la definición de los casos de uso y más puntos relevantes que serán comentados a continuación.

La elicitación de requisitos resulta un proceso esencial para cualquier tipo de proyecto o negocio. Nos ayuda a descubrir quienes son los usuarios interesados (stakeholders), y sus necesidades y, por tanto, es fundamental la participación colaborativa entre el equipo de desarrollo y los stakeholders para garantizar la comprensión entre ambas partes.

4.1 Lean Canvas

El primer paso tomado para la elicitación de requisitos fue la elaboración de un *Lean Canvas* [3].

El modelo *Lean Canvas* resulta una herramienta visual y sencilla para que el cliente analice y valide la idea de negocio, ya que muestra de forma simplificada las principales cuestiones en todo proyecto: cuál es el problema y su solución, qué aporta nuestra solución que no tengan otros competidores, cómo se accederá a los resultados, quién podría interesarse y cómo mediremos el éxito obtenido, así como la estructura de costes y posibles ingresos.

Seguidamente, explicaremos las ‘parcelas’ o puntos clave del *Lean Canvas*, mostrados en la figura 5:

- **Problema:** Nuestro problema a resolver consiste en la dificultad de encontrar una clínica o centro médico que se ajuste al ideal buscado por el usuario. Esto se debe a que la principal fuente de información usada por la gente interesada en este ámbito es la encontrada en Google Reviews o Doctoralia.
- **Solución:** La solución pasa por la creación de una aplicación web donde clientes de clínicas puedan subir sus experiencias, o aquellos que busquen saber más sobre ellas pueden contactar con estos primeros
- **Métricas:** El éxito del proyecto puede ser medido según la cantidad de visitas que reciba la aplicación producto del proyecto, al igual que mediante la cantidad de publicaciones publicados y la cantidad de usuarios satisfechos.
- **Propuestas de valor:** Creación de un espacio donde los usuarios puedan comunicarse entre ellos con el fin de compartir sus experiencias.
- **Ventaja competitiva:** Contamos con la ventaja de que ningún competidor cuenta con el monopolio del sector. Este punto fue analizado en profundidad en las conclusiones del capítulo tercero.
- **Canales:** La forma de usar nuestro producto será a través de un navegador. Los canales de comunicación para hacer conocer el producto será publicidad en redes sociales.
- **Clientes:** Los principales clientes del producto final serán los usuarios que busquen compartir sus experiencias, así como aquellos que busquen información específica

sobre alguna intervención. Como *early adopters*⁷ contamos con las clínicas interesadas en publicitar su negocio mediante nuestra aplicación.

- **Estructura de costos:** Los costes implican tanto los sueldos de todos los miembros del equipo, como el equipamiento e infraestructura *cloud* necesaria. Es posible que si el volumen de usuario aumente más de lo esperado requiera más inversión.
- **Fuente de ingresos:** La web será de uso gratuito, por lo que obtendremos ingresos mediante anuncios. Sin embargo, al tratarse de una idea inicial de producto, no se ha tenido en cuenta este hecho.

LEAN CANVAS				
Problema Dificultad a la hora de encontrar la clínica que más se ajuste al ideal que busca el usuario. Esto se debe a que la principal fuente de información es Google Reviews o la página web que pueda tener la propia clínica, donde no se puede contactar con otros clientes.	Solución Creación de una página web donde clientes de clínicas puedan subir sus experiencias, o aquellos que busquen saber más sobre ellas pueden contactar con estos primeros.	Propuestas de valor Comunicación entre usuarios con el fin de compartir su experiencia.	Ventaja competitiva Encuentra la clínica que mas se acerque a tus ideales de manera gratuita gracias al contacto directo con otros usuarios. -Sector no monopolizado	Clientes Gente que busque compartir su buena o mala experiencia en alguna clínica. Gente que busque información más detallada o específica sobre alguna intervención en una clínica. Early adopters: Clínicas interesadas en la publicidad de su negocio
	Métricas Visitas a la página Publicaciones de posts Usuarios satisfechos		Canales La forma de usar el producto sería a través de un navegador web de cualquier tipo de dispositivo. Los canales de comunicación para conocer el producto será publicidad en redes sociales.	
Estructura de costos Los costes son tanto los sueldos de todos los miembros del equipo, como el equipamiento e infraestructura cloud necesaria. Es posible que si el volumen de usuario aumente más de lo esperado requiera más inversión.			Fuente de ingresos La web será gratuita, por lo que obtendremos ingresos mediante anuncios.	

Figura 5: modelo *Lean Canvas* aplicado al proyecto.

4.2 Brainstorming

La técnica de *Brainstorming* consiste en recoger ideas e información de todos los stakeholders involucrados en el proyecto, para poder descubrir objetivos y requisitos previamente no detectados, obteniendo al finalizar una lista de requisitos funcionales. El ejercicio consta de tres fases: preparación, generación y consolidación.

La actividad de *Brainstorming* se desarrolló mediante dos sesiones a lo largo de un día, en la cual participaron cuatro personas, contando con el desarrollador. Ambas sesiones fueron separadas por una franja de cinco horas: en la primera de estas se ejecutaron las dos primeras fases del ejercicio, y en la segunda la última.

A lo largo de la primera fase, se explicó a los participantes el motivo del ejercicio, ya que se trataba de personas externas al proyecto. Esto se consiguió mediante una presentación en la cual se explicaba en qué se basa la técnica de *Brainstorming* y los objetivos a lograr con relación al

⁷ Con el término *early adopters* nos referimos a los primeros consumidores del producto, es decir, aquellos que estarán interesados en usar el producto nada más lanzarlo.

proyecto mediante un resumen de éste. Algunas capturas de la presentación pueden observarse en la figura 6.



Figura 6: ejemplos de diapositivas usadas en la presentación.

A lo largo del ejercicio, un participante asume el rol de ‘moderador’. Este tiene como tarea crear un ambiente que estimule a los demás integrantes a aportar ideas, así como evitar las críticas de cualquier idea aportada por parte de otros participantes. Otro de los participantes actúa como secretario, el cual es el encargado de apuntar todas las ideas que surjan durante la siguiente fase en notas adhesivas. La persona encargada de este rol habitualmente no participa en el proceso de creación de ideas, pero dado que se trabajó con un grupo reducido, el secretario también asumió el papel de un stakeholder. El resto de los participantes tienen como misión principal producir todo tipo de ideas en la siguiente fase: estas abarcan desde las más básicas hasta las más poco convencionales, siempre y cuando puedan ser realizables.

Durante la segunda parte del ejercicio se transmiten, en primer lugar, los objetivos y procedimiento a seguir. Las normas básicas de la fase de generación son la espontaneidad y la no repudiación de ninguna idea, es decir, evitar las ideas preconcebidas y expresiones que puedan congestionar el proceso creativo. Lo importante durante este periodo es la cantidad de ideas surgidas, no la calidad, ya que más adelante serán ordenadas.

En el último fragmento del *Brainstorming*, se revisan todas las ideas anotadas. Estas son clarificadas, organizadas según su prioridad o incluso descartadas si estas no pueden ser realizables.

En total, al finalizar el ejercicio surgieron veinticinco ideas de requisitos, que más adelante fueron formalizadas (ver tabla 2), añadiendo su descripción, stakeholders involucrados, y si se trata de un requisito funcional, no funcional o centrado en el usuario.

Descripción de idea de requisito <i>Se debe seguir la estructura de historia de usuario: Como (rol) quiero/necesito (algo) [para (beneficio/objetivo)]</i>	Tipo de idea (Funcional, No Funcional)	Prioridad (alta, media, baja)	Stakeholders involucrados
---	---	----------------------------------	---------------------------

Como administrador necesito tener buenos términos de servicio para cumplir la ley	No Funcional	Alta	Administrador
Como usuario quiero una sección donde enviar tickets y propuestas para enviar mis quejas y sugerencias	Funcional	Alta	Administrador, Usuario reseña, Usuario preguntas
Como administrador quiero aprobar comentarios para decidir cuáles son verídicos o no contienen nada ofensivo	Funcional	Media	Administrador, Usuario reseña, Usuario preguntas
Como administrador quiero poder borrar usuarios para que su información ya no exista	Funcional	Media	Administrador, Usuario reseña, Usuario preguntas
Como propietario de clínica quiero verificar ser el propietario para que quede constancia de la veracidad de la información	Funcional	Media	Propietario clínica
Como propietario de clínica quiero tener una sección en mi página donde incluir noticias para que los usuarios sepan sobre mí	Funcional	Media	Propietario clínica
Como propietario de la clínica quiero mostrar fichas sobre mi personal para mostrar sus cualidades y valoraciones	Funcional	Media	Propietario clínica, Usuario reseña, Usuario preguntas
Como propietario de clínica quiero incluir los métodos de contacto en mi página para que los usuarios sepan cómo contactarme	Funcional	Media	Propietario clínica, Usuario reseña, Usuario preguntas
Como usuario con dudas quiero recibir recomendaciones de clínicas para saber cuáles se asemejan a mis gustos	Funcional	Media	Usuario preguntas
Como usuario reseñador quiero poder valorar las características concretas de la clínica para valorar de forma más detallada	Funcional	Media	Usuario reseña
Como usuario quiero un chat privado para hacer preguntas más personales a otros usuarios	Funcional	Alta	Usuario reseña, Usuario preguntas
Como usuario quiero un perfil propio para ver mis publicaciones y visitar otros perfiles	Funcional	Alta	Usuario reseña, Usuario preguntas

Como usuario necesito identificarme y registrarme para hacer uso de la web	Funcional	Alta	Usuario no identificado, Usuario reseña, Usuario preguntas
Como usuario quiero escribir comentarios en una publicación para dar mi opinión	Funcional	Alta	Usuario reseña, Usuario preguntas
Como usuario quiero poder filtrar las publicaciones para poder buscar más fácilmente lo que necesito	Funcional	Alta	Usuario no identificado, Usuario reseña, Usuario preguntas
Como usuario necesito subir una publicación para realizar una consulta o dar una reseña	Funcional	Alta	Usuario reseña, Usuario preguntas, Propietario clínica
Como usuario quiero un mapa de la localización de las clínicas para visualizarlas más fácilmente	Funcional	Alta	Usuario no identificado, Usuario reseña, Usuario preguntas
Como usuario quiero varias secciones para poder ver cada clase de contenido (preguntas, reseñas, etc.)	Funcional	Alta	Usuario no identificado, Usuario reseña, Usuario preguntas
Como usuario quiero solicitar la inclusión de una clínica que aún no esté registrada para poder investigar sobre ella	Funcional	Media	Usuario reseña, Usuario preguntas
Como usuario quiero reportar a otros usuarios para dar constancia de su comportamiento ofensivo	Funcional	Media	Usuario reseña, Usuario preguntas
Como usuario quiero poder responder a otros comentarios para seguir el hilo de la conversación	Funcional	Media	Usuario reseña, Usuario preguntas
Como usuario quiero poder borrar mis publicaciones o archivarlos para que no aparezcan en el <i>feed</i>	Funcional	Media	Usuario reseña, Usuario preguntas
Como usuario quiero bloquear a otros usuarios para no recibir mensajes suyos	Funcional	Media	Usuario reseña, Usuario preguntas
Como usuario quiero poder subir fotos para explicar mejor mis publicaciones	Funcional	Media	Usuario reseña, Usuario preguntas
Como usuario quiero un historial para ver las publicaciones que he creado	Funcional	Baja	Usuario reseña, Usuario preguntas

Como usuario quiero compartir una publicación para que tenga más visibilidad	Funcional	Baja	Usuario no identificado, Usuario reseña, Usuario preguntas
---	-----------	------	--

Tabla 2: ideas resultantes del ejercicio de *brainstorming*.

4.3 Análisis DAFO

La matriz DAFO [4] es un método de análisis que nos ayuda a conocer la situación actual del proyecto. Para preparar un análisis DAFO es fundamental haber definido el objetivo, y analizar el resultados y cambios que puedan ser derivados. El resultado de la matriz DAFO analiza aspectos tanto internos como externos, mostrándolos en una matriz de 2x2, en la que cada celda corresponde con una inicial del nombre de la matriz: Debilidades, Amenazas, Fortalezas y Oportunidades.

Las oportunidades y amenazas son los aspectos ajenos que se encuentran fuera de nuestro alcance. En cuanto a las amenazas, representan aquellos factores que pueden llegar a poner en peligro la viabilidad del proyecto. Por otro lado, las oportunidades recogen aspectos que resultan favorables para el desarrollo

Las celdas de la matriz que reflejan los aspectos internos, por descarte, son las debilidades y fortalezas. Para comprender las debilidades, debemos analizar aquellos factores que limitan nuestra capacidad de desarrollo, mientras que las fortalezas se centran en comprender las ventajas competitivas que poseemos de las cuales nos podamos beneficiar.

Debilidades	Amenazas
Capacidad de desarrollo para una sola plataforma.	Aumento de los costes en plataformas publicitarias.
Tiempo limitado de desarrollo.	KeyWords utilizadas por grandes empresas.
Desconocimiento inicial de la normativa legal.	Aplicaciones alternativas.
Mal uso por parte de los usuarios.	
Fortalezas	Oportunidades
Cantidad de usuarios potenciales.	No hay ninguna aplicación web/app que se haya "adueñado" del mercado
Tecnologías modernas.	
Funcionalidades novedosas para el dominio.	Facilidad para extenderse a nuevos mercados geográficos

Figura 7: matriz DAFO aplicada al proyecto.

Observando la matriz (figura 7) podemos analizar tanto los aspectos débiles como los puntos fuertes de nuestro proyecto.

Por una parte, podemos observar varias **debilidades** detectadas. La mayoría de estas derivan del hecho de que se trata de un trabajo personal y con fecha de entrega, por lo que no se disponen de recursos y tiempo ilimitado. También cabe destacar que nuestra web requiere de cierta seguridad al ser usada, ya que algunos temas pueden resultar personales, por lo que debe respaldarse en la normativa legal de protección de datos, entre otras.

Ubicada en la celda superior izquierda, tenemos las **amenazas**. Entre ellas se encuentra la posibilidad del aumento de los costes en plataformas publicitarias. Esto viene a advertir la posible limitación del proyecto para darse a conocer mediante publicidad, al ser lanzado. Dicha amenaza viene de la mano de la siguiente: el monopolio en la búsqueda por palabras clave por parte de las alternativas a nuestro proyecto. Dado que las palabras usadas para buscar nuestra aplicación web son muy concretas, puede derivar en la imposibilidad de localizar la aplicación al buscar términos relacionados en el navegador.

En cuanto a las **fortalezas**, contamos con una gran cantidad de usuarios potencialmente interesados, ya que se trata de un proyecto que podría llegar a ser usado a escala nacional si se contase con los recursos adecuados. En segundo lugar, mencionar el uso de tecnologías modernas y cada vez más usadas, lo cual nos permite mejorar la experiencia de usuario y rendimiento respecto a otras alternativas más antiguas. La última fortaleza por destacar es, como se mencionó en las conclusiones del segundo capítulo, las funcionalidades novedosas que aporta esta nueva solución.

En último lugar, analizando las **oportunidades** nos encontramos con el hecho de que no existe una aplicación web haya conseguido el monopolio del mercado. A su vez, otra de las oportunidades vigentes es la facilidad para extenderse a nuevos mercados geográficos. Dado que nuestra aplicación web estará centrada en operaciones, resultaría sencillo ampliar algunas funcionalidades para abarcar otro tipo de atención médica profesional.

4.4 Diagrama de contexto

A partir de este punto, todos los esquemas e información abstraída de ellos han sido elaboradas mediante el software *Enterprise Architect*⁸, una herramienta utilizada para la documentación y análisis de sistemas complejos creada en el año 2000.

Un diagrama de contexto es una representación gráfica de alto nivel del producto. Resulta una herramienta útil para identificar factores y eventos externos que puedan ser de impacto en los procesos desarrollados durante el proyecto. En él se muestran los actores y sus interfaces junto con el sistema.

⁸ <https://sparxsystems.com/>

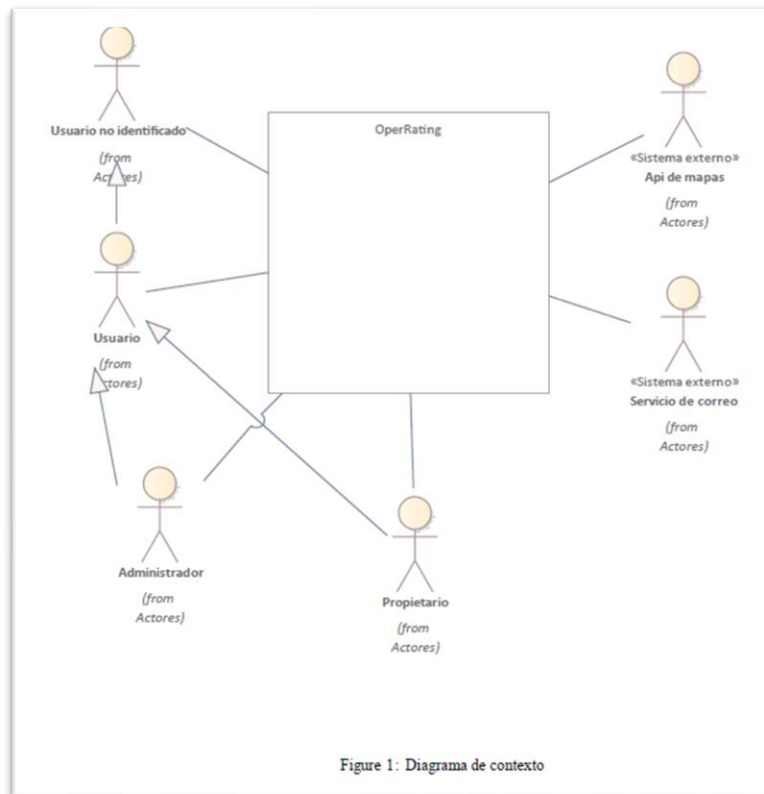


Figura 8: actores identificados.

Como podemos observar en la figura 8, se han identificado un total de cuatro actores y dos sistemas externos. El bloque central representa el sistema, el cual se encuentra conectado con las demás entidades.

Respecto a los **actores**, podemos diferenciar varios tipos de usuarios relacionados mediante herencia: aquellos que no se han identificado o los que sí, los que administran algún tipo de centro (propietarios) y los propios administradores de la web. Al margen derecho de la figura se muestran los sistemas externos identificados, el servicio de correos que será usado para enviar y recibir mensajes en relación con la aplicación, y el Servicio de mapas, el cual es el encargado de proporcionarnos información geográfica a medida que el usuario interactúe con los mapas mostrados en nuestra aplicación web.

4.5 Características del sistema

Las características mostradas en la figura 9 nos definen las cualidades principales que debe poseer nuestro sistema software a desarrollar.

En total, se han identificado cinco características principales, que serán descompuestas en casos de uso en la siguiente sección:

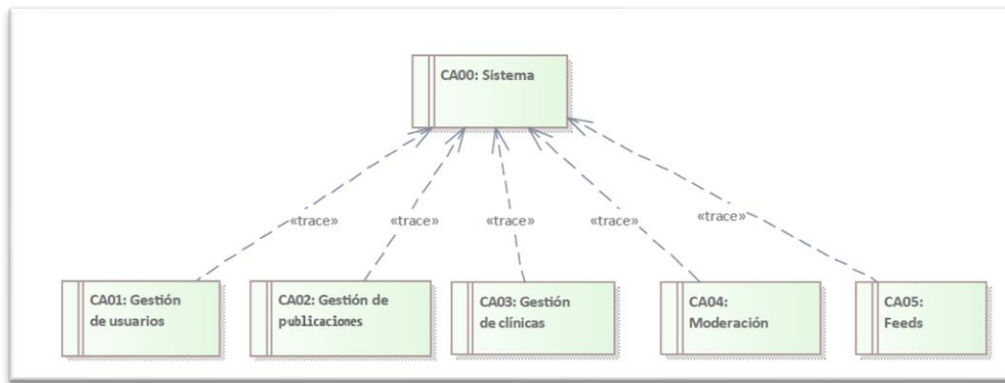


Figura 9: características del sistema.

Sistema: Característica transversal que engloba las demás.

Gestión de usuarios: Actividades relacionadas con los usuarios: registro, identificación, borrar cuenta, historial y mensajes directos.

Gestión de publicaciones: Agrupa funcionalidades relacionadas con los publicación y comentarios en ellos.

Gestión de clínicas: Creación y perfiles de las clínicas.

Moderación: Funcionalidades relacionadas con la moderación integradas en el sistema.

Feeds: Funcionalidades de mapas, filtrados y agrupaciones de publicaciones.

4.6 Diagrama de casos de uso

Un diagrama de casos de uso sirve para modelizar cuáles son los casos de uso [5] del sistema, los actores relacionados con cada caso de uso, y los pasos a seguir para reproducir cada funcionalidad a implementar en el proyecto. Son un medio de comunicación con los stakeholders para determinar lo que se piensa hacer en el sistema.

Cada caso de uso representa una unidad de trabajo significativo. En ellos se definen las acciones que se deben llevar a cabo para ejecutar correctamente el caso de uso, y se especifican los resultados esperados. En el diagrama, se provee una vista de alto nivel del comportamiento observable mediante una forma de elipse. Estos se especifican más en profundidad separadamente, explicando su descripción, relaciones entre sí y demás con tal de detallar el proceso lo mejor posible.

A continuación, se muestra un ejemplo por cada uno de los casos de uso especificados según su característica. Los casos de uso restantes pueden ser encontrados en el [Anexo B](#).

4.6.1 Gestión de usuarios

Para la gestión de usuarios se han identificado un total de doce casos de uso, mostrados en la figura 10.

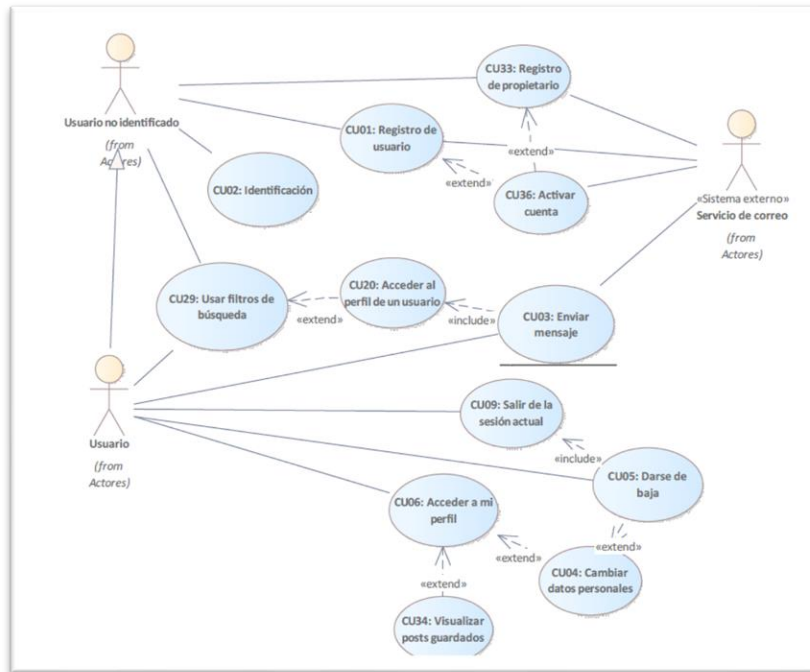


Figura 10: diagrama de casos de uso para la gestión de usuarios.

Caso de uso	CU01: Registro de usuario
Actor	Usuario no identificado
Descripción	Registra en el sistema un nuevo usuario
Relaciones	Extend CU21 se han de consultar términos
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario no identificado accede a crear usuario 2. El sistema muestra todos los campos necesarios para registrar al usuario 3. El usuario no identificado rellena todos los campos 4. El sistema muestra un mensaje de éxito a la hora de crear el usuario. 5. El servicio de correo envía un email al usuario no identificado 6. El sistema redirige al usuario no identificado a la página de inicio de sesión 7. El sistema envía un correo al nuevo usuario registrado confirmando su registro
Alternativas	<ol style="list-style-type: none"> 3. El sistema encuentra algún usuario o propietario creado con la misma información (mismo nombre o correo) 3.1 El sistema le muestra un mensaje de error al usuario no identificado

Tabla 3: CU01, registro de usuario

4.6.2 Gestión de publicaciones

En la característica Gestión de publicaciones se han reconocido once casos de uso esenciales para el proyecto de entre los requisitos funcionales surgidos del ejercicio de *brainstorming*, mostrados en la figura 11.

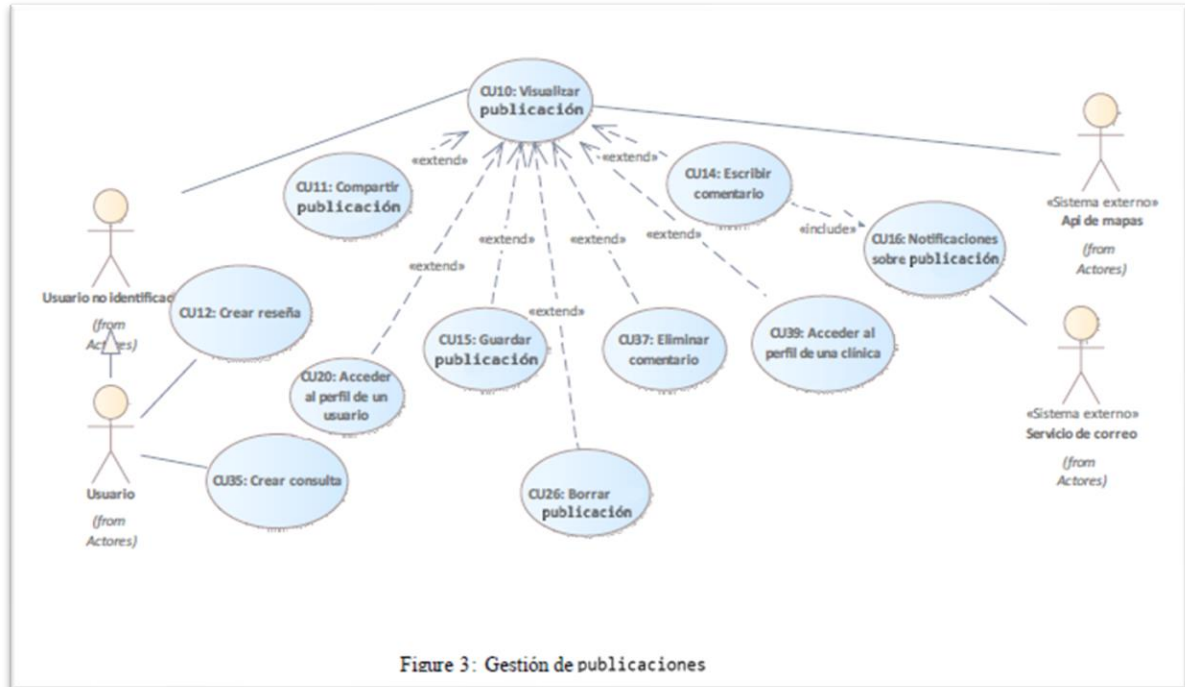


Figura 11: diagrama de casos de uso para la gestión de publicaciones.

Caso de uso	CU10: Visualizar publicación
Actor	Usuario no identificado Servicio de mapas
Descripción	Permite al usuario ver los detalles de una publicación
Relaciones	
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede a una publicación 2. El sistema muestra los datos relacionados con la publicación, como puede ser el contenido, ubicación del suceso y comentarios
Alternativas	

Tabla 4: CU10, visualizar publicación.

4.6.3 Gestión de clínicas

La característica Gestión de clínicas, como se explicó anteriormente, está relacionada con las funciones que se podrán llevar a cabo respecto los centros médicos en nuestra aplicación web. De entre las ideas generadas en el ejercicio de *brainstorming*, se han seleccionado un total de nueve casos de uso para esta característica (figura 12).

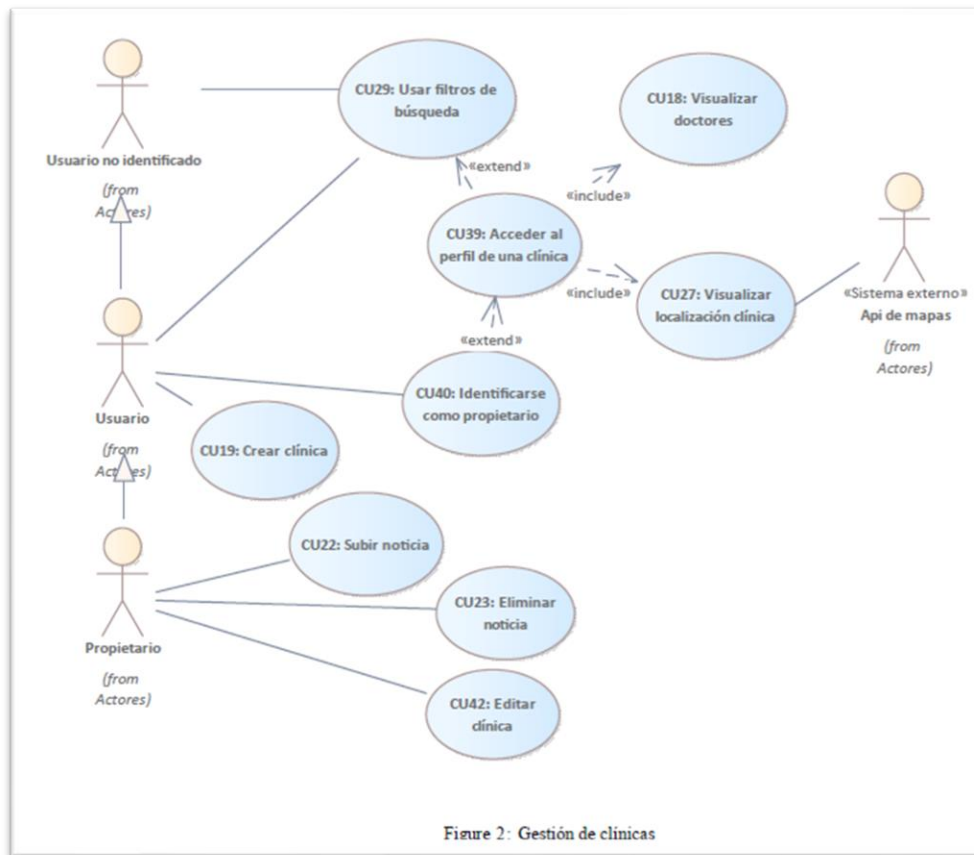


Figura 12: diagrama de casos de uso para la gestión de clínicas.

Caso de uso	CU18: Visualizar doctores
Actor	Usuario no identificado
Descripción	Permite al usuario ver la lista de doctores que trabajan en una clínica.
Relaciones	
Precondición	La clínica tiene asociada doctores
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede a visualizar la clínica 2. El sistema muestra una lista de los doctores que trabajan en ella y la valoración de los usuarios sobre ellos
Alternativas	

Tabla 5: CU18 visualizar doctores.

4.6.4 Moderación

La característica Moderación resulta relevante para cualquier aplicación que se asemeje a una red social. Por tanto, los casos de usos seleccionados son los mínimos necesarios para crear un entorno seguro en la aplicación, así como prevenir el acoso a los usuarios. Para conseguir este objetivo, se han seleccionado cuatro casos de uso, mostrados en la figura 13.

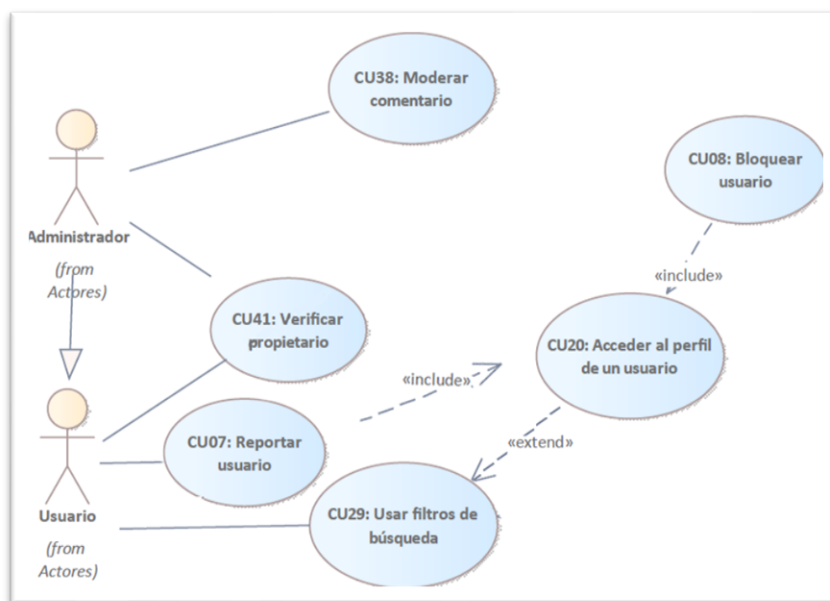


Figura 13: diagrama de casos de uso para la moderación.

Caso de uso	CU07: Reportar usuario
Actor	Usuario
Descripción	Permite al usuario comunicar al sistema que otro usuario se comporta de forma inadecuada
Relaciones	Extend CU20, acceder al perfil de un usuario
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede al perfil del usuario a reportar 2. El usuario pulsa sobre reportar usuario 3. El sistema muestra varios motivos de reporte y un campo para la descripción 4. El usuario rellena los datos 5. El sistema registra el reporte y se notifica a un administrador
Alternativas	

Tabla 6: CU07, reportar usuario.

4.6.5 Feeds

En la figura 14 podemos distinguir los casos de uso determinados. Al igual que en los anteriores puntos, disponemos de una tabla explicativa por cada caso encontradas en el Anexo B:

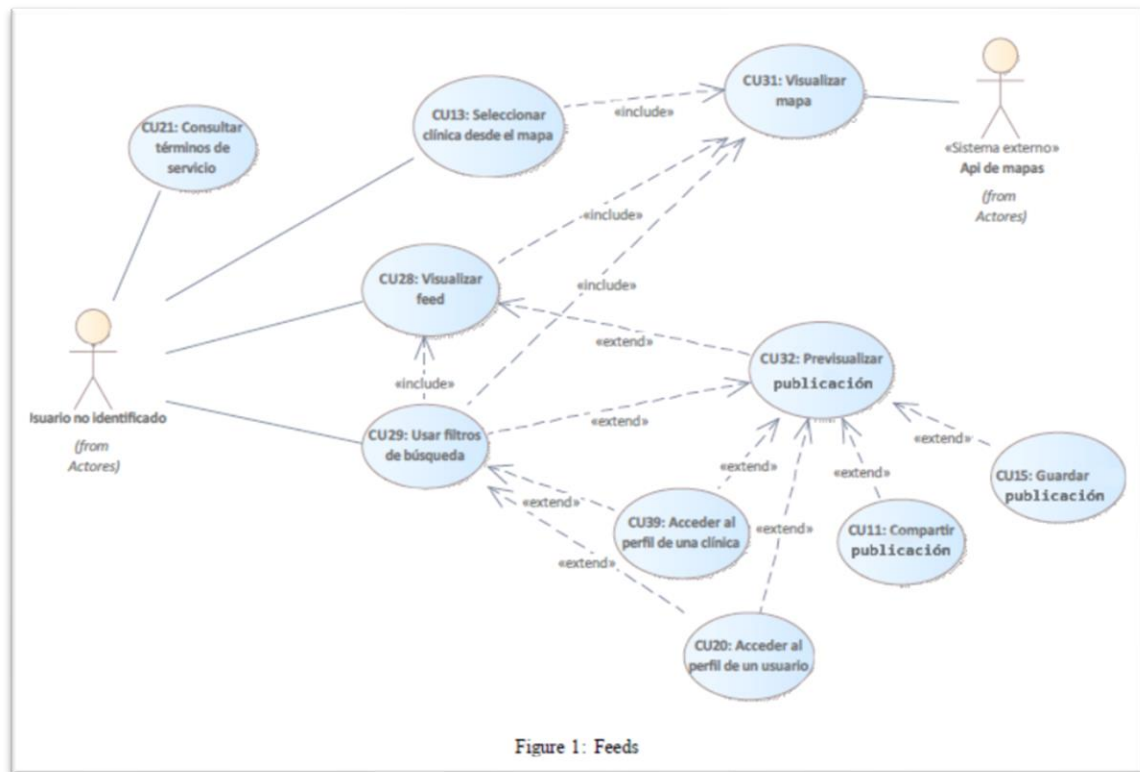


Figure 1: Feeds

Figura 14: diagrama de casos de uso para los feeds.

Caso de uso	CU13: Seleccionar clínica desde el mapa
Actor	Usuario no identificado Servicio de mapas
Descripción	Permite al usuario entrar en el perfil de una clínica desde el mapa del <i>feed</i>
Relaciones	<i>Include</i> CU31
Precondición	Existen clínicas en el sistema
Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona desde el mapa una clínica 2. El sistema muestra un resumen sobre ésta. 3. El usuario vuelve a seleccionarla 4. El sistema redirige al usuario al perfil de la clínica
Alternativas	

Tabla 7: CU13: Seleccionar clínica desde el mapa.

4.7 Modelo de dominio

Producir un modelo de dominio en nuestro proyecto nos brinda un mapa conceptual que representa los conceptos centrales y relaciones de interés del dominio del proyecto (figura 15).

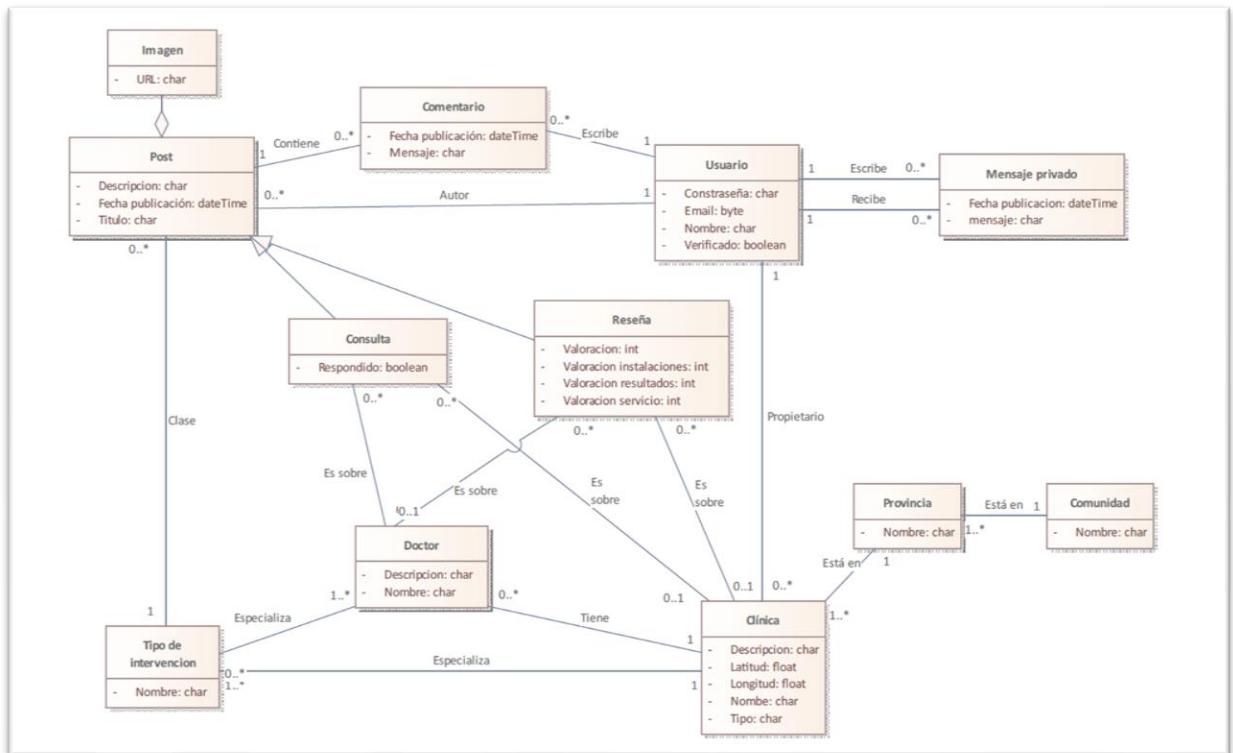


Figura 15: modelo de dominio generado.

El pilar central de la aplicación es, como podemos suponer, la habilidad de crear una publicación ya sea con motivo de hacer una valoración a una intervención ya realizada o una consulta. Por tanto, contamos con el concepto de ‘Post’, es decir, de publicación, y dos clases que heredan atributos de esta, Consulta y Reseña. A su vez, para poder organizar dichas publicaciones, es necesaria la relación directa entre cada entidad de tipo Reseña o Consulta con los necesarios. Dependiendo ya del autor de la publicación y sus necesidades, una publicación puede estar relacionada con un tipo de intervención, un doctor que esté asociado a una clínica específica, o a la propia clínica. Por descontado, la publicación se encuentra vinculado con los comentarios que llegue a recibir, y con el propio autor de la publicación, es decir, el usuario que la ha creado. Sobre dicho usuario también tenemos los mensajes privados, en los cuales intervienen dos usuarios.

Respecto a las clínicas, para que el sistema de búsqueda sea más sencillo, se proporciona también la provincia a la cual

4.9 Prototipado

La técnica de prototipado [6] es una de las más frecuentemente utilizadas durante el proceso de elicitación de requisitos software. Los prototipos nos proveen de una representación preliminar y parcial de la interfaz que se asemeje al resultado que conseguimos esperar, lo cual nos permite hacer los conceptos y requisitos más tangibles.



Al ya contar con los suficientes datos como para plantear una interfaz adecuada, se ha optado por crear un prototipo de baja fidelidad o *Wireframe* [7]. Los *Wireframes* son diseños centrados en mostrar la disposición del contenido, sin preocuparse de los detalles que serán implementados, como bien pudieran ser el tamaño de la fuente o los colores elegidos. Esto lo logramos mediante representaciones básicas mediante campos de texto, imágenes y secciones de contenido.

Para lograr diseñar dichos *Wireframes*, se ha elegido el software *Figma*, una herramienta de prototipado web creada en 2016. Dicha herramienta resulta muy intuitiva y sencilla de usar, incluso para usuarios sin experiencia previa.

Dado que se trata de un proyecto desarrollado mediante metodologías ágiles dividido por *sprints*, unos días antes del inicio de cada iteración se deben preparar los prototipos para cada Unidad de Trabajo, en lugar de especificar todos al inicio.

En las figuras 16 y 17 se pueden observar ejemplos de *Wireframes* desarrollados, los cuales muestran diseños preliminares para varios de los casos de uso especificados en la sección anterior. El Wireframe mostrado en la figura 16 representa un caso de uso de la característica gestión de publicaciones, concretamente el caso de uso CU12: Crear reseña. Mediante el boceto podemos comprender más fácilmente cómo se llevaría a cabo el proceso de redactar la información necesaria para crear una publicación de tipo reseña. Por otro lado, la figura 17 agrupa varios casos de uso de la característica *Feeds*, como bien podrían ser el CU29: Usar filtros de búsqueda, CU28: Visualizar *feed*, CU31: Visualizar mapa o CU13: Seleccionar clínica desde el mapa.

Estos bocetos son un ejemplo del trabajo realizado ya que, mediante 27 bocetos similares, se ha cubierto la totalidad de funcionalidades a implementar.

El wireframe muestra una interfaz de usuario para crear una publicación. En la parte superior, hay una barra de navegación con el logo 'OperRating' a la izquierda, los enlaces 'Feed' y 'Clinicas' en el centro, y 'Logout' a la derecha. El contenido principal está dividido en una columna central y una columna lateral a la derecha. La columna central contiene un formulario con los siguientes campos: 'Titulo', 'Profesional', 'Clínica', 'Tipo de interveccion' (con un menú desplegable que muestra 'v'), 'Texto' (un área de texto grande) y 'Valoración' (con cinco estrellas). A la izquierda del formulario, hay un enlace 'Crear un post de reseña'. A la derecha, hay un botón 'Subir post'. Debajo del formulario, hay un botón 'Añadir imagen' y una barra de desplazamiento horizontal.

Figura 16: wireframe formulario creación de una publicación.

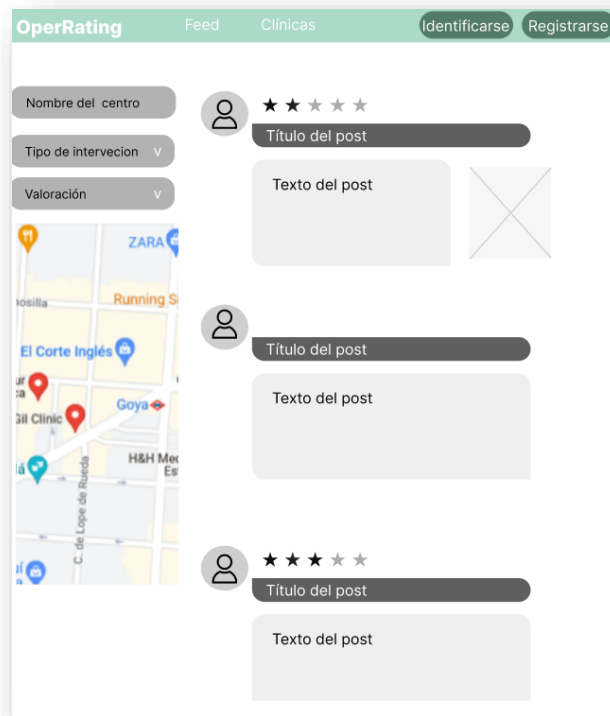


Figura 17: wireframe feed de publicaciones.

4.10 Análisis del marco legal y ético

En un proyecto cuyo objetivo es recopilar personal de los usuarios y compartirla, uno de los aspectos más fundamentales es la cumplimentación de la normativa vigente del Reglamento General de Protección de Datos (RGPD) y la Ley Orgánica de Protección de Datos y Garantía de los Derechos Digitales (LOPDGDD).

Es un hecho que resulta primordial aclarar diversas cuestiones respecto a cómo se tratan los datos que recogemos, como puede ser la dirección de correo electrónico de los usuarios, o la propia información que puedan publicar. Teniendo en cuenta las normativas actuales, la aplicación deberá contener toda la información necesaria para la tranquilidad de los usuarios, dividida en varias páginas. La primera de estas comentará la política de privacidad, mencionando sobre qué leyes se respaldan diversos aspectos de la aplicación, qué datos se guardan y la gestión de *cookies*, entre otros. La segunda de ellas incluirá los términos y condiciones del servicio, donde se redactarán las normas a las que está sujeta la aplicación y condiciones de uso.

A la hora de realizar el registro de usuario o clínica médica en nuestra aplicación, los usuarios deberán aceptar varias casillas informativas que confirmarán que están de acuerdo con los términos y condiciones del servicio y la política de privacidad, dando su consentimiento explícito para almacenar los datos.

En relación con la ética del proyecto, es importante destacar la ética representa también un papel esencial, siendo necesaria la intervención de moderadores o administradores en la aplicación

con tal de ayudar a construir un ambiente más seguro para los usuarios. Los administradores serán los encargados de leer las solicitudes enviadas por los usuarios mediante el formulario de contacto o reportes a publicaciones, usuarios o perfiles de clínicas. Así mismo, también serán capaces de eliminar comentarios o publicaciones que no cumplan las condiciones de uso del servicio, al igual que modificar los datos de cualquier perfil, tanto de un usuario como de un centro médico. Esta presencia activa de administradores no solo ayuda a mantener un ambiente seguro y ético en la aplicación a desarrollar, sino que también contribuye a mejorar la confianza de los usuarios en el servicio que se ofrece.

5. Diseño de la solución

Habiendo explicado ya los requisitos clave de nuestra aplicación, pasamos a comenzar su diseño.

El presente capítulo tiene como objetivo repasar la arquitectura resultante y la estructura detallada que se desea conseguir en la base de datos.

5.1 Arquitectura del sistema

Para el desarrollo del proyecto se ha usado un patrón arquitectónico dividido en niveles o capas, concretamente en tres [8].

La arquitectura de tres niveles es una de las técnicas más utilizadas para separar sistemas de software cliente-servidor habituales, quedando fraccionado el sistema en presentación, negocio o dominio y persistencia, las cuales tienen un bajo nivel de acoplamiento entre ellas. Actualmente, existen varias alternativas más modernas para este tipo de aplicaciones software, como pueden ser los microservicios [9] o los contenedores [10].

Respetar el orden de las capas es muy importante, dado que saltar una capa para situarnos sobre una inferior suele ser un grave error; aunque es habitual hacer esto último, se crea un desorden sobre el flujo de comunicación, lo que nos puede llevar al *antipatrón de diseño*⁹ conocido como código espagueti, el cual consiste en empezar a realizar llamadas desde cualquier capa a otra capa, complicando en gran medida el mantenimiento del sistema.

La comunicación entre capas adyacentes requiere una comprensión mutua de los servicios y la información que la capa inferior necesita proporcionar a la capa superior, como puede ser visualizado en la figura 18.

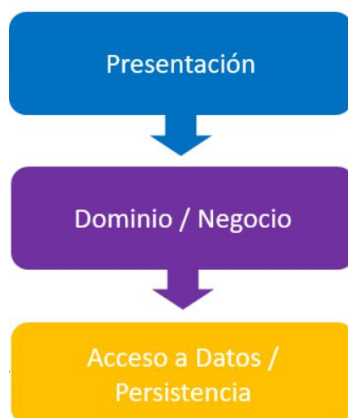


Figura 18: representación arquitectura dividida en tres capas.

⁹ Patrón de diseño que resulta ser una mala solución para el problema a afrontar [11].

A continuación, explicaremos qué responsabilidades asume cada capa en nuestro proyecto:

- La capa de presentación, también llamada *front-end*, es la encargada del manejo de la interfaz gráfica de usuario. Su objetivo principal, por tanto, es mostrar información al usuario y recopilar datos. Mediante la capa de presentación, nos comunicaremos con la *API*¹⁰ desarrollada en el *back-end* para recibir o enviar dichos datos.
- La capa de negocio o *back-end* es el componente encargado de codificar las reglas que determinan cómo crear, almacenar y cambiar los datos de los objetos, así como las restricciones y operaciones que se apliquen. Por tanto, es la encargada de procesar las solicitudes enviadas por la capa de presentación, poner en uso las reglas creadas e interactuar con la capa de persistencia.
- La capa de persistencia o base de datos es la responsable de almacenar y obtener datos desde un almacén de datos. En nuestro caso, la tecnología escogida para la capa de negocio nos proporciona facilidades a la hora de definir los modelos de la base de datos y guardarlos en la base de datos seleccionada, la cual ha sido *SQLITE*¹¹.

5.2 Diseño de la base de datos

En cuanto al diseño de la base de datos, contamos con varias clases y atributos añadidos respecto al modelo de dominio presentado en el capítulo cuarto (figura 19).

¹⁰ El concepto API, o *application programming interface*, sirve para intercambiar datos entre diferentes piezas de software, automatizando así procedimientos complejos.

¹¹ <https://www.sqlite.org/index.html>

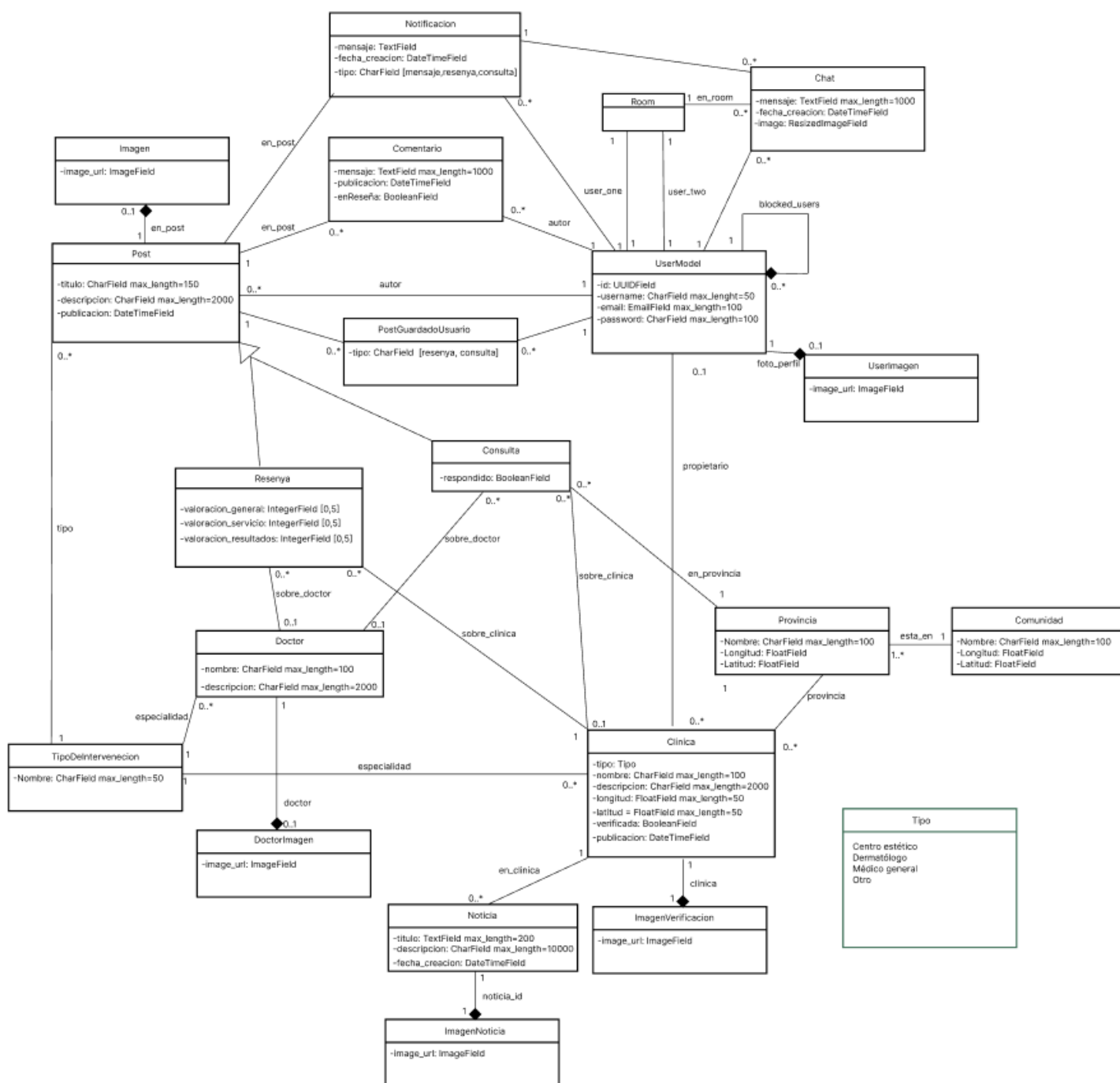


Figura 19: diseño logrado de la base de datos.

Los aspectos por resaltar que divergen en cuanto a las predichas en el modelo de dominio son, en su mayoría, la incorporación de clases para las imágenes, necesarias para los usuarios, doctores, noticias y verificación de las clínicas. También contamos con la clase Noticia, Notificación y Room, la cual viene a describir el espacio privado en el que se comunican dos usuarios. Se han incorporado a demás, una nueva relación entre UserModel y sí mismo, la cual se refiere a los usuarios bloqueados que tiene el usuario origen, y una relación entre Consulta y Provincia, ya que una consulta puede no estar adherida a una clínica concreta, pero se debe saber su localización.

5.3 Patrones aplicados

La aplicación de patrones de diseño resulta una solución eficiente para varios problemas frecuentes en el desarrollo de software. Existe una amplia gama de patrones, divididos en grupos según qué comportamiento sigan, pueden ser creacionales, estructurales o de comportamiento. En cuanto a los patrones creacionales, se refieren a aquellos que proporcionan formas de creación de objetos y aumentan la reutilización del código existente. Los patrones estructurales, por otra parte, nos muestran cómo unir objetos en estructuras más eficientes que las originales. En último lugar, los patrones de comportamiento son usados para crear comunicaciones eficientes entre distintos objetos.

Cabe recalcar que un patrón es una pauta que seguir basada en las buenas prácticas de diseño, lo que significa que cada patrón debe ser ajustado al problema presentado, y no es un algoritmo listo para ser usado en cualquier ocasión.

Con relación al proyecto, se ha empleado el patrón creacional *Singleton*, cuyo propósito es asegurarnos que solo se cree una instancia de una clase, con tal de ahorrar recursos.

Se ha hecho uso del patrón *Singleton* a la hora de solicitar los datos del usuario que se ha identificado. Esto se hace en varias pantallas de la aplicación, pero es usado constantemente en la barra de navegación superior, donde aparece el nombre de usuario y su foto de perfil.

La estructura del patrón es la mostrada en la figura 20, contamos con la clase *UsuarioService*, la cual tiene un constructor, y el método que recoge los datos, *getUserInfo*.

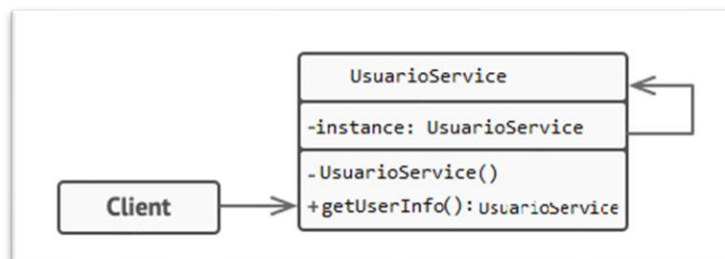


Figura 20: estructura del patrón *Singleton*.

```

export class UsuarioService {
  constructor() {
    // Verificar si ya existe una instancia de la clase
    if (typeof UsuarioService.instance === "object") {
      return UsuarioService.instance;
    }

    // Si no existe, crear una nueva instancia
    this.token = null;
    this.info = null;
    this.blocked_users = null;
    this.image = null;
    UsuarioService.instance = this;

    // Configurar la cookie CSRF para las solicitudes POST
    axios.defaults.xsrfCookieName = "csrftoken";
    axios.defaults.xsrfHeaderName = "X-CSRFToken";

    return this;
  }

  async getUserInfo(token, csrftoken) {
    if (this.info !== null) {
      return this.info;
    }
    // Si la información del usuario no ha sido cargada aún, llamar a la API
    this.token = token;
    try {
      const response = await axios.post(
        "http://localhost:8000/api/auth/user/" + token + "/",
        {}
      );
    }
  }
}

```

Figura 21: código relevante del patrón *Singleton*.

En la figura 21 podemos observar el esquema previamente comentado llevado a la práctica, mostrando las partes más relevantes. El método *getUserInfo* es el encargado de conseguir y preservar los datos del usuario, concretamente su nombre de usuario, imagen de perfil, lista de usuarios bloqueados y su estado, es decir, si es administrador o no.

La instancia resultante es empleada en otros componentes además de en la barra de navegación principal, como por ejemplo en el perfil tanto del usuario como de la clínica.

6. Desarrollo de la solución

La solución propuesta pasa por la ejecución de cuatro *sprints* de desarrollo web más uno inicial de elicitación de requisitos. Durante estos *sprints* se ha registrado todo el trabajo y proceso en la herramienta de gestión Works, la cual será revisada en profundidad durante la sección de tecnologías utilizadas. En las siguientes subsecciones profundizaremos también en las tecnologías escogidas, tareas elegidas para cada *sprint*, su estimación y su desarrollo.

6.1 Tecnologías utilizadas

En la presente sección comentaremos las tecnologías elegidas para cada aspecto de la aplicación.

6.1.1 HTML

HyperText Markup Language o HTML¹² es el lenguaje más básico de una web. A pesar de soler estar apoyado en otras tecnologías para describir su apariencia, HTML describe la estructura del contenido de la página web a desarrollar mediante diversos elementos como, por ejemplo, imágenes, encabezados o listas.

Como se mencionó anteriormente, HTML necesita de otras tecnologías ya que su función es proporcionar una estructura, por lo que es necesario usar herramientas como CSS, JavaScript y un navegador.

6.1.2 CSS

CSS o *Cascading Style Sheets*¹³ es una herramienta complementaria al lenguaje HTML. Gracias a CSS podemos indicar al navegador el estilo que deben seguir los elementos de nuestra aplicación web; cada elemento puede tener su propio estilo de forma individual, o varios de ellos compartir el mismo.

Dominar CSS puede suponer una tarea compleja, ya que, como su nombre indica, los estilos se aplican en cascada. Esto hace muy sencillo introducir código repetido, lo cual conlleva que estilos ya definidos se apliquen a otros nuevos, siendo sobrescritos y siendo a veces complicado encontrar el origen del error.

6.1.3 JavaScript

¹² <https://developer.mozilla.org/en-US/docs/Web/HTML>

¹³ <https://developer.mozilla.org/en-US/docs/Web/CSS>

JavaScript¹⁴ es un lenguaje de programación dinámico, orientado a objetos y débilmente tipado capaz de ser usado en una amplia gama de dispositivos.

Se dice que es dinámico ya que no necesita de una compilación previa para funcionar en tiempo de ejecución, lo cual permite probar el código de forma inmediata, pero nos impide encontrar errores hasta que el programa sea ejecutado.

Este lenguaje es comúnmente utilizado en el desarrollo web *front-end*, pero cada vez es más frecuente emplearla en el *back-end* mediante el apoyo de *frameworks* como Next.js o Node.js.

6.1.4 React

React¹⁵ (o React.js) es una librería de código abierto de JavaScript. Actualmente, junto con Angular, es una de las herramientas más utilizadas para crear páginas web. Ofrece varias características innovadoras respecto a otras tecnologías:

Lenguaje JSX: Extensión de la sintaxis de JavaScript que permite juntar JavaScript y HTML en el mismo archivo.

Virtual DOM: Permite que el rendimiento de las páginas creadas mejore considerablemente, ya que reduce la cantidad de actualizaciones necesitadas, pero aun reflejando fielmente los datos a mostrar. El Virtual DOM se encarga de actuar como memoria intermediaria entre el DOM (*Document Object Model*) real y las actualizaciones de la página.

Ciclos de vida: En React, existe un método que es activado cuando cambia de estado algún elemento específico, y se permite llamar a acciones cuando éste es provocado.

Componentes: Las aplicaciones creadas mediante React están compuestas por varios componentes. Cada uno encapsula una funcionalidad, vista y estado, por lo que pueden llegar a ser reutilizables en varios escenarios. Estos aceptan también atributos, los cuales son definidos al ser llamado el componente por otro componente.

6.1.5 Python

Python¹⁶ es uno de los lenguajes de programación más populares y utilizados en la actualidad, debido principalmente a su versatilidad y su facilidad para los principiantes. Frecuentemente es usado en el *back-end* ya que existe una gran cantidad de librerías que facilitan en gran medida el trabajo de los desarrolladores de software gracias a su popularidad. Comparado con otros

¹⁴ <https://www.javascript.com/>

¹⁵ <https://react.dev/>

¹⁶ <https://www.python.org/>

lenguajes, Python proporciona un código relativamente seguro, pero siendo posible contar con algunas vulnerabilidades, las cuales pueden ser cubiertas mediante el uso de librerías o *Frameworks* como Django o Flask.

6.1.6 Django

Django¹⁷ es un *framework* de desarrollo para Python creado en 2005. El término *framework* quiere decir que gracias a Django tenemos una estructura base o plantilla que nos ayuda durante las diferentes fases de desarrollo del software.

En la parte *back-end* del proyecto, juntamente con Django, se ha usado Django *Rest Framework* (DRF), una extensión de Django especializada en la creación de *APIs RESTful* que proporciona elementos específicos para la creación de las *API*, como serializadores de objetos o técnicas de autenticación y permisos para el uso de la *API*. Cuando hablamos de una *API RESTful* nos referimos a aquellas *API* que siguen la guía de los servicios REST:

Tecnología sin estado: Cada solicitud a la *API* debe contener toda la información necesaria para procesarla, ya que el servidor no puede almacenar los datos relacionados con la solicitud del cliente.

Sistema por capas: Se proporciona transparencia entre la comunicación servidor-cliente con tal de que ninguna de las dos partes sepa si se está comunicando con un intermediario o con la aplicación final.

Operaciones CRUD: Las *API REST* utilizan operaciones de creación (*create*), lectura (*read*), actualización (*update*) y borrado (*delete*), cuyas iniciales en inglés forman las siglas CRUD.

Interfaz uniforme: Las solicitudes recibidas deben tener el mismo formato, por lo que se debe asegurar que un identificador solo corresponde a un recurso, el cual no debe ser muy grande.

Django, además, proporciona también diversos mecanismos por defecto que añaden seguridad a las aplicaciones desarrolladas mediante este *framework*. Entre los aspectos más destacables, tenemos la prevención contra inyecciones SQL [12]; esto quiere decir que Django evita consultas maliciosas dirigidas hacia nuestra base de datos. También contamos con protección contra ataques *Cross-Site Request Forgery*, o CSRF. Estos ataques consisten en conseguir que un usuario envíe, sin ser consciente en la mayoría de los casos, solicitudes maliciosas a la aplicación web.

¹⁷ <https://www.djangoproject.com/>



6.1.7 Postman

La herramienta Postman¹⁸ es utilizada por desarrolladores de software para probar y ayudar a construir *APIs*. Mediante ella, podemos introducir el enlace de la petición a probar, junto con los valores deseados, el tipo de operación (*CRUD*) y comprobar si funciona como se espera.

Es posible, como será demostrado en siguientes capítulos, generar una documentación completa de forma automática a partir de las peticiones y descripciones introducidas.

6.1.8 Visual Studio Code

Visual Studio Code¹⁹ es un editor de código fuente multiplataforma desarrollado por Microsoft. Cuenta con un sinnúmero de extensiones de gran utilidad para la depuración y desarrollo de código, como pueden ser *Docker* o *Pylance*. Estas extensiones nos permiten agregar funcionalidad de forma modular, consiguiendo así una mejor experiencia general.

En el caso de nuestro proyecto, se ha trabajado con la extensión ESLint, la cual es una herramienta de análisis de código estático de JavaScript enfocada en mantener un código ‘limpio’, es decir, nos muestra errores de sintaxis, sugerencias de mejora y demás.

6.1.9 Worki

Para gestionar el trabajo de manera ágil, es necesario contar con alguna herramienta organizativa de apoyo. Como se mencionó en el primer capítulo, el proyecto ha sido desarrollado basándose en la metodología *Tune-Up Process*, por lo que Worki representa una elección firme como herramienta de gestión de proyectos.

Worki proporciona facilidades a la hora de crear Unidades de Trabajo, definir los *sprints* y seguir el estado del proyecto mediante diversas gráficas que se actualizan según los tiempos registrados en cada actividad, y cuenta con facilidades para el trabajo colaborativo en caso de tratarse de proyectos grupales.

6.1.10 Figma

Al desarrollar nuestro proyecto mediante metodologías ágiles, es necesario crear prototipos que validen las funcionalidades proporcionadas por cada Unidad de Trabajo. Estos prototipos son diseñados antes de iniciar el *sprint* que contenga dichas Unidades de Trabajo, ya que es necesario tener los requisitos preparados antes de comenzar a programarlos.

Para este proyecto, se ha escogido Figma²⁰ para crear los prototipos, como ya se mencionó en el capítulo anterior.

¹⁸ <https://www.postman.com/>

¹⁹ <https://code.visualstudio.com/>

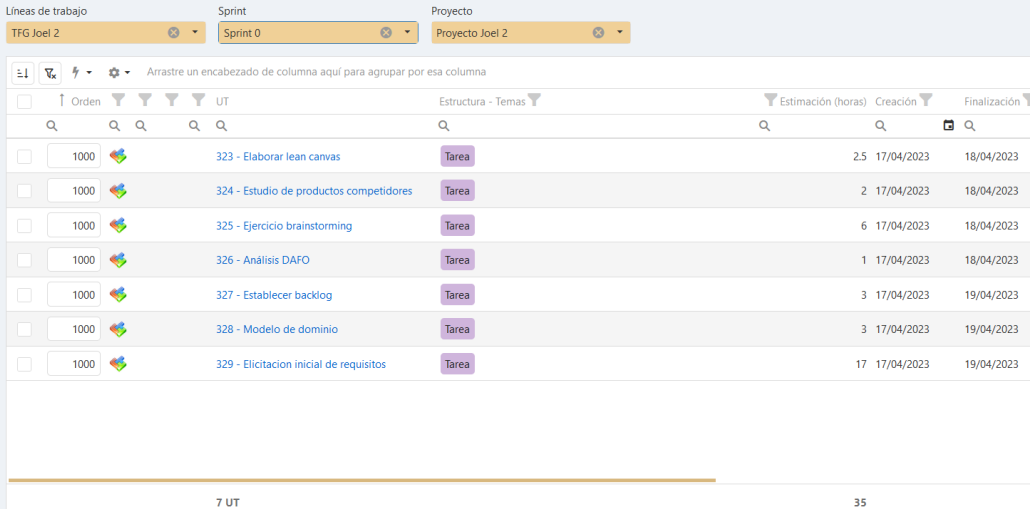
²⁰ <https://www.figma.com/>

6.2 Proceso de desarrollo

6.2.1 Sprint 0

El *sprint 0* consiste en la puesta en marcha del proyecto. A lo largo del *sprint*, se deben realizar las tareas de organización del Backlog, preparar el entorno de desarrollo e infraestructura del trabajo, y, en nuestro caso, la elicitación inicial de requisitos. El comienzo del *sprint* se marcó el 26 de marzo y finalizó el 23 de abril.

Las Unidades de Trabajo incluidas para el *sprint 0* fueron las comentadas en el capítulo cuarto: Elaborar *lean canvas*, estudio de productos competidores, ejercicio de brainstorming, análisis DAFO, establecimiento del backlog, elaboración del modelo de dominio y elicitación inicial de requisitos, la cual engloba, por ejemplo, la creación de los diagramas de casos de uso. En la figura 22 se pueden observar las UTs ordenadas por prioridad, junto con la estimación temporal y fecha de inicio y finalización de cada una de éstas.



Orden	UT	Estructura - Temas	Estimación (horas)	Creación	Finalización
1000	323 - Elaborar lean canvas	Tarea	2.5	17/04/2023	18/04/2023
1000	324 - Estudio de productos competidores	Tarea	2	17/04/2023	18/04/2023
1000	325 - Ejercicio brainstorming	Tarea	6	17/04/2023	18/04/2023
1000	326 - Análisis DAFO	Tarea	1	17/04/2023	18/04/2023
1000	327 - Establecer backlog	Tarea	3	17/04/2023	19/04/2023
1000	328 - Modelo de dominio	Tarea	3	17/04/2023	19/04/2023
1000	329 - Elicitacion inicial de requisitos	Tarea	17	17/04/2023	19/04/2023

Figura 22: UTs designadas al *sprint 0*.

El avance del *sprint* se encuentra resumido en varias gráficas obtenibles desde el panel *Dashboard* que nos ofrece Worki. En la figura 23 podemos ver la gráfica *Burndown*. Esta gráfica nos permite supervisar el ritmo de avance del trabajo teniendo en cuenta el tiempo restante hasta la finalización del *sprint*, respecto a la suma de las estimaciones de cada Unidad de Trabajo: la línea roja nos muestra el esfuerzo restante y la azul el esfuerzo registrado, mientras que la verde representa la estimación total

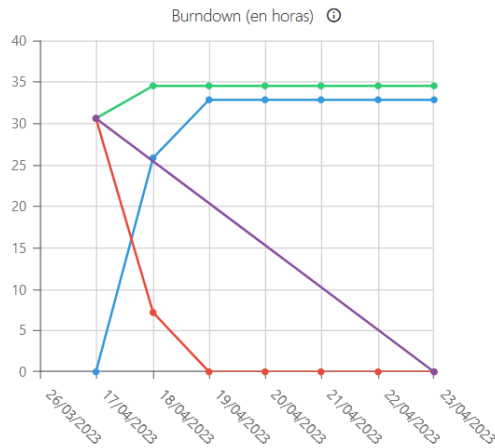


Figura 23: gráfica burndown sprint 0.

La gráfica de Flujo Acumulado muestra las actividades que están activas día a día, teniendo cada actividad un color asociado. De esta forma, se puede ver el ritmo de trabajo a medida que pasa el tiempo, detectando si hay problemas en alguna Unidad de Trabajo. Cabe anotar que, al ser posible realizar varias actividades en paralelo sobre la misma Unidad de Trabajo, esta puede verse reflejada en varias actividades al mismo tiempo. Dado que este *sprint* está dedicado a tareas de especificación y elicitación de requisitos, las Unidades de Trabajo se encuentran en el *Workflow* tarea, cuyos colores son cian, para la actividad Realizar Tarea, y azul para Terminar (figura 24).

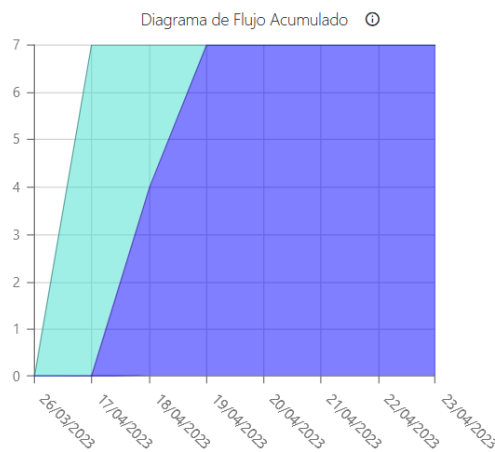


Figura 24: diagrama de Flujo Acumulado sprint 0.

En la gráfica inferior (figura 25) podemos comparar la estimación inicial de cada Unidad de Trabajo respecto a la duración real registrada. Las franjas marrones indican el tiempo estimado mientras que las rojas muestran la diferencia. Por ejemplo, en el caso de la Unidad de Trabajo 326 (Análisis DAFO), contaba con una estimación inicial de una hora aproximadamente, mientras que el total registrado fue menor al esperado.

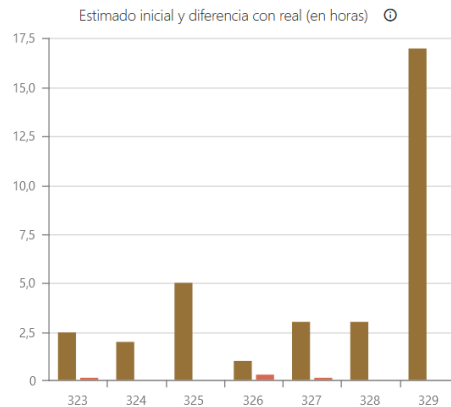


Figura 25: gráfica estimaciones *sprint 0*.

6.2.2 Sprint 1

El *sprint 1* tuvo comienzo el 23 de abril y tuvo finalización el 21 de mayo. Durante la última semana del *sprint* anterior, se eligieron las Unidades de Trabajo a realizar, y se crearon sus respectivas pruebas de aceptación y Wireframes. En este *sprint* se incluyeron las Unidades de Trabajo con las estimaciones indicadas en la figura 26.

Orden	UT	Estructura - Temas	Estimación (horas)	Creación	Finalización
1002	333 - Crear post reseña	Usuario → Post	13	17/04/2023	15/05/2023
1002	334 - Crear post consulta	Usuario → Post	5	17/04/2023	15/05/2023
1004	337 - Crear clinica	Clinica Clinica → Doctor	12	17/04/2023	20/05/2023
1005	330 - Inicio de sesión	Usuario	5	17/04/2023	07/05/2023
1005	331 - Registro usuario	Usuario	18	17/04/2023	21/05/2023
1006	335 - Feed	Pagina inicial → Feed	9	17/04/2023	17/05/2023
1007	336 - Pagina inicial	Pagina inicial	10	17/04/2023	16/05/2023
2006	332 - Añadir imagen a post	Usuario → Post	3	17/04/2023	21/05/2023
8 UT			75		

Figura 26: UTs designadas al *sprint 1*.

El diagrama de Flujo Acumulado (figura 27) nos señala que durante la primera semana del *sprint* no se terminó ninguna Unidad de Trabajo, debido a que al ser el primer *sprint* dedicado a la programación de la aplicación se deben hacer muchas tareas iniciales para la puesta en marcha del proyecto.

Por otra parte, en los últimos días se eliminó una Unidad de Trabajo del *sprint*, y se modificó la Unidad de Trabajo de Registro de Usuario para añadir una continuación de ésta durante el segundo *sprint* debido a problemas relacionados con la autenticación por correo electrónico. El código de colores para el *Workflow* de desarrollo con *sprints* nos muestra para la etapa de programación un amarillo, mientras que para Terminar tenemos un color azul.

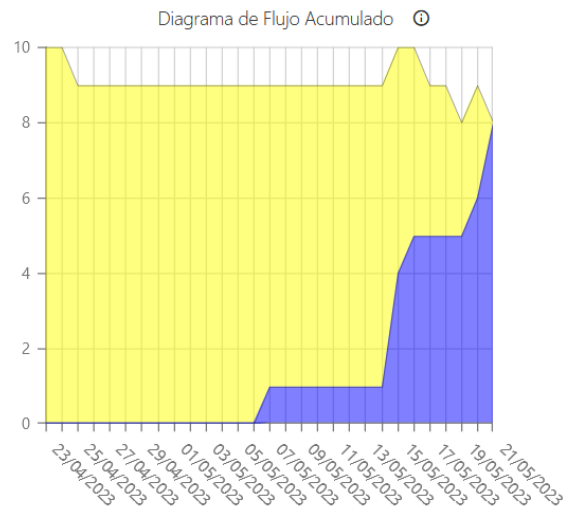


Figura 27: diagrama de Flujo Acumulado *sprint 1*.

En la gráfica *Burndown* (figura 28) del *sprint 1*, podemos ver un ritmo de programación generalmente adecuado, ya que el esfuerzo restante (línea roja) no se aleja de la línea central de forma desmedida.

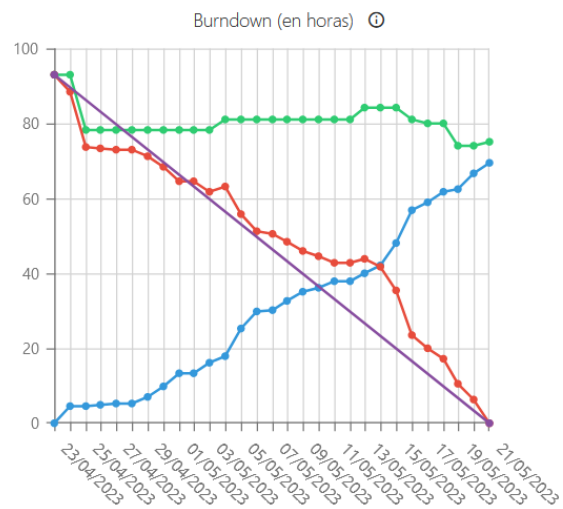


Figura 28: gráfica *burndown sprint 1*.

Seguidamente, en la gráfica de la figura 29, podemos observar que la mayoría de las estimaciones fueron acertadas. Sin embargo, para las Unidades de Trabajo 332, 333 y 337, añadir imagen a publicación, crear publicación reseña y crear clínica respectivamente, para dos de estas se tardó menos de lo esperado, mientras que para otra el desarrollo se demoró tres horas más de lo inicialmente esperado.

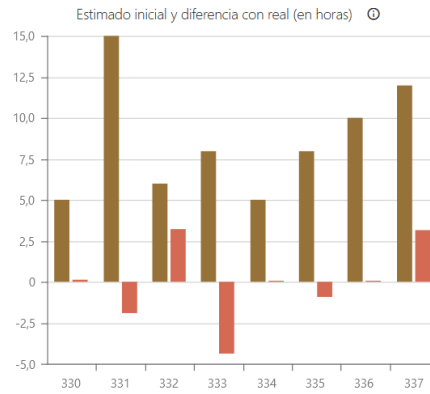


Figura 29: gráfica estimaciones *sprint 1*.

6.2.3 Sprint 2

El segundo *sprint* de nuestro proyecto cuenta con un total de nueve Unidades de Trabajo. Entre ellas contamos con la continuación de Registro de Usuario, mapa interactivo, búsqueda con filtros, comentarios, perfiles de usuarios y clínicas, notificaciones, edición de perfiles y guardar publicaciones. Se pueden ver dichas UT en la figura 30, junto con las estimaciones, en horas, de cada una. El *sprint* comenzó el día 21 de mayo y finalizó el 15 de junio.

Orden	UT	Estructura - Temas	Estimación (horas)	Creación	Finalización
1001	373 - (Continuación de) Registro usuario	Usuario	3	21/05/2023	27/05/2023
1003	340 - Mapa interactivo	Pagina inicial → Mapa	20	17/04/2023	31/05/2023
1005	339 - Búsqueda con filtros	Pagina inicial → Filtros	15	17/04/2023	13/06/2023
1007	344 - Comentarios	Usuario → Notificación Usuario → Post → Comentario	7.2	17/04/2023	29/05/2023
1009	355 - Perfil clínica	Clinica	10	17/04/2023	03/06/2023
1010	343 - Perfil usuario	Usuario	11	17/04/2023	09/06/2023
1011	345 - Notificaciones	Usuario → Notificación	6	17/04/2023	15/06/2023
1012	346 - Guardar post	Usuario Usuario → Post	7	17/04/2023	11/06/2023
1018	338 - Editar clínica	Clinica	6	17/04/2023	08/06/2023
9 UT			85		

Figura 30: UTs designadas al *sprint 2*.

Al contrario que en el primer *sprint*, se puede ver un ritmo de finalización de actividades más constante desde el inicio del *sprint* en el diagrama de Flujo Acumulado, terminando la primera Unidad de Trabajo a los cuatro días tras comenzar el *sprint* (figura 31).

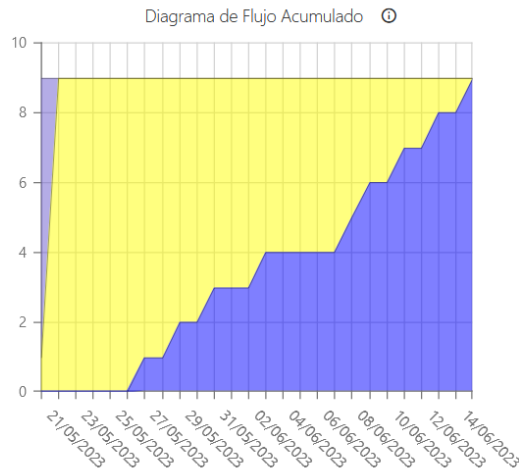


Figura 31: diagrama de Flujo Acumulado *sprint 2*.

Siguiendo el trabajo marcado por la gráfica *Burndown* en la figura 32, comprobamos que no solo el diagrama de flujo acumulado muestra progreso constante, sino que ésta también. Se puede verificar mediante el esfuerzo estimado (línea verde) que no se han realizado reestimaciones elevadas sobre las horas de trabajo en el *sprint*, lo cual indica una correcta preparación anterior del *sprint*.

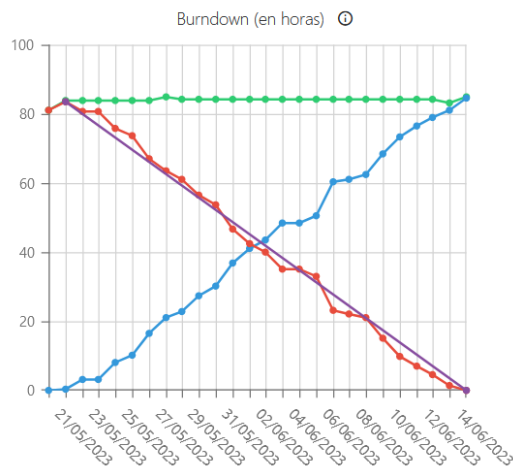


Figura 32: gráfica *burndown* *sprint 2*.

En cuanto a la estimación inicial y su diferencia final (figura 33), como se comentó para la figura anterior, contamos con una precisión bastante acertada a la hora de ejecutar la actividad de especificación de requisitos para las Unidades de Trabajo del *sprint 2*. La UT más alejada de lo esperado es la 344, comentarios.

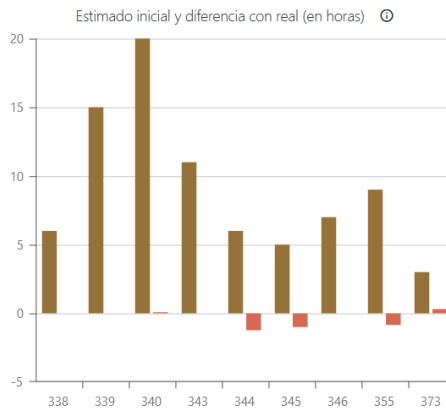


Figura 33: gráfica estimaciones *sprint 2*.

6.2.4 Sprint 3

El inicio del tercer *sprint* de desarrollo del proyecto tuvo lugar el 15 de junio y su finalización se ubica en el 18 de julio. En él, se llevaron a cabo las unidades de trabajo mostradas en la figura 34, con un total de cincuenta y cuatro horas dedicadas a la programación.

Líneas de trabajo		Sprint	Proyecto		UT	Estructura - Temas	Estimación (horas)	Creación	Finalización
TFG Joel 2		Sprint 3	Proyecto Joel 2						
1002	347 - Chat privado		Usuario	Chat		14	17/04/2023	18/07/2023	
1002	352 - Noticias de clinicas		Clinica	Noticia		7	17/04/2023	17/06/2023	
1003	354 - Verificar propietarios		Usuario			8	17/04/2023	28/06/2023	
1004	348 - Compartir post		Usuario	Post	Compartir	3	17/04/2023	29/06/2023	
1005	342 - Formulario de contacto		Pagina inicial		Tickets	4	17/04/2023	21/06/2023	
1005	349 - Darse de baja		Usuario			1	17/04/2023	24/06/2023	
1006	350 - Reportar usuario		Usuario			4	17/04/2023	01/07/2023	
1006	351 - Bloquear usuario		Usuario			4	17/04/2023	09/07/2023	
1007	353 - Moderar comentarios		Usuario	Post	Comentario	4	17/04/2023	02/07/2023	
1009	341 - Página sobre temas legales		Pagina inicial			5	17/04/2023	17/07/2023	
10 UT							54		

Figura 34: UTs designadas al *sprint 3*.

Al igual que en el segundo *sprint*, en el diagrama de flujo acumulado (figura 35) percibimos que se finalizaron algunas unidades de trabajo desde las fases iniciales del *sprint*. También contamos, durante un periodo de tres días, con una unidad de trabajo que se encuentra en fase de Probar, es decir, aplicar pruebas de aceptación.

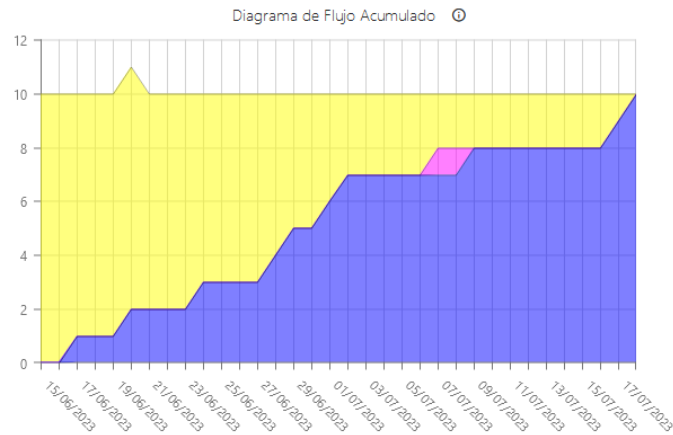


Figura 35: diagrama de Flujo Acumulado *sprint 3*.

En la gráfica *Burndown* (figura 36) se aprecia que, durante las primeras semanas del *sprint*, se llevó a cabo gran parte del trabajo de programación, habiendo una pausa de varios días en la mitad final del *sprint*.

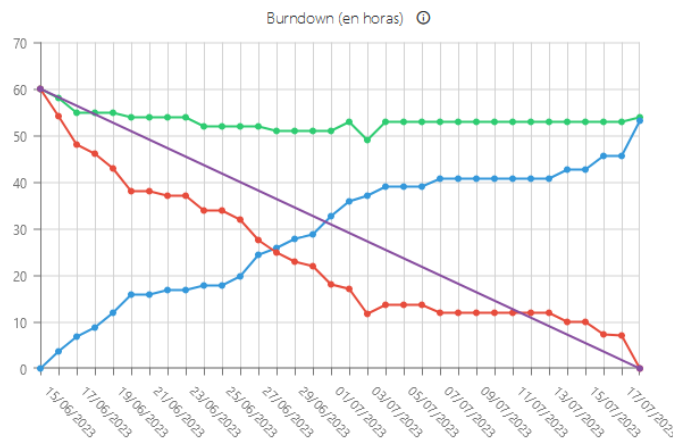


Figura 36: gráfica *burndown* *sprint 3*.

En cuanto a las reestimaciones de las unidades de trabajo (figura 37), en su mayoría las valoraciones iniciales fueron correctas, exceptuando las unidades de trabajo 351, bloquear usuario, y 352, noticias de clínicas, las cuales fueron considerablemente menores a lo esperado.

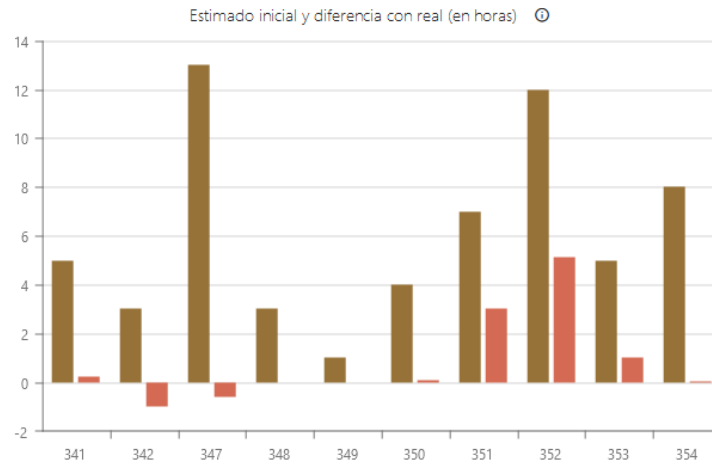


Figura 37: gráfica estimaciones *sprint* 3.

El resultado de la ejecución total de estos *sprints* puede encontrarse en el manual de usuario dado en el **Anexo E**, pudiendo encontrar imágenes como las mostradas en las figuras 38 y 39.

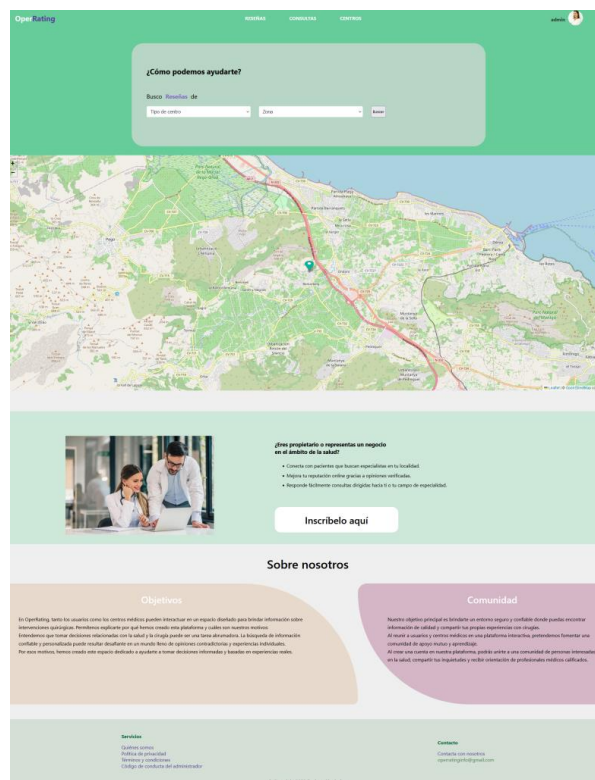


Figura 38: página de inicio.

Aplicación para la gestión de reseñas y consultas de intervenciones quirúrgicas

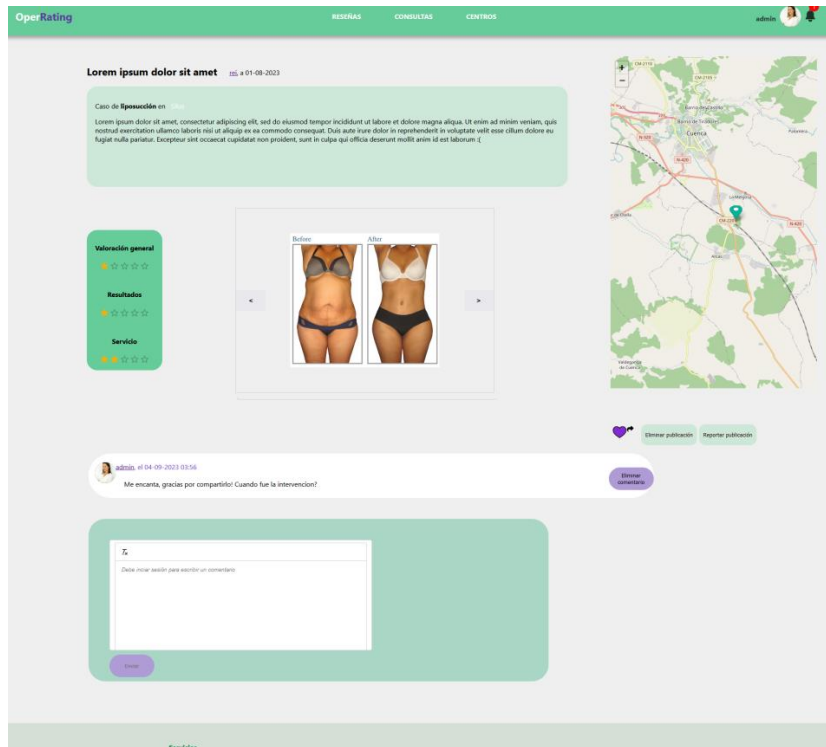


Figura 39: funcionalidades dentro de una publicación.

7. Pruebas

A lo largo del séptimo capítulo de la memoria, comentaremos el producto resultante de la ejecución de los *sprints* realizados, las pruebas de aceptación realizadas, cuestionarios de usabilidad llevados a cabo y el posible despliegue de la aplicación.

7.1 Pruebas API

El desarrollo de una API conlleva la creación de diversos enlaces, o *endpoints*, mediante los cuales se comunican el *front* y *back-end*, enviando solicitudes y respuestas, respectivamente.

A lo largo del proceso de desarrollo, se han creado múltiples *endpoints* para la API, necesitando entonces de un método de verificación para comprobar su correcto funcionamiento. Para ello, se ha recurrido a la herramienta de pruebas Postman, descrita durante el capítulo anterior en la sección Tecnologías utilizadas.

La documentación lograda Django²¹ es el producto del testeado paralelo al desarrollo de la aplicación; según se han ido creando enlaces en la API desarrollada, se ha creado su respectiva entrada en la colección de Postman

7.2 Pruebas de aceptación

En nuestro contexto dirigido por metodologías ágiles, el producto se estructura en una jerarquía de nodos, ubicando en cada nodo la unidad de trabajo que se vea conveniente. Los nodos y pruebas de aceptación (PA) elaboradas durante los *sprints* están disponibles en el **Anexo C**, agrupadas según el nodo al que pertenezcan, su Unidad de Trabajo y *sprint* en el que fueron descritas y ejecutadas por primera vez.

Las pruebas de aceptación creadas para cada unidad de trabajo definidas y ejecutadas durante cada *sprint* son vueltas a ejecutar al finalizar el proyecto, con tal de validar el aún correcto funcionamiento los requisitos previamente especificados.

En total, se han descrito 40 pruebas de aceptación para las 35 unidades de trabajo desarrolladas en el proyecto. Al ejecutar de nuevo las PAs se comprueba que, efectivamente, el sistema sigue cumpliendo las expectativas propuestas inicialmente.

7.3 Cuestionario de usabilidad

Dado ya por finalizado el proceso de desarrollo, es de suma importancia comprobar la usabilidad de la aplicación con tal de detectar problemas potenciales antes de hacer público el producto. Se define como usabilidad la medida en la que un producto se puede usar por determinados usuarios para conseguir objetivos específicos en un contexto específico. Realizar

²¹ <https://documenter.getpostman.com/view/25879401/2s9XtxFaV>



este proceso de determinación de la usabilidad ayuda también a medir la eficiencia y eficacia media de los usuarios y su satisfacción general con el producto resultante.

Para este proyecto se ha escogido la escala *System Usability Scale (SUS)* [13], una de las herramientas más rápidas y fiables que se sigue usando en proyectos actuales para medir la usabilidad. De los resultados de la escala se puede extrapolar la existencia de problemas en el sistema, si la puntuación resultante es baja. Sin embargo, con tal de obtener resultados fiables, es recomendable la ejecución de la SUS por un total de entre 8 a 12 usuarios.

El cuestionario relacionado con la escala SUS consta de 10 preguntas, contando con la siguiente traducción oficial al español:

1. Me gustaría usar esta herramienta frecuentemente.
2. Considero que esta herramienta es innecesariamente compleja.
3. Considero que la herramienta es fácil de usar.
4. Considero necesario el apoyo de personal experto para poder utilizar esta herramienta.
5. Considero que las funciones de la herramienta están bien integradas.
6. Considero que la herramienta presenta muchas contradicciones.
7. Imagino que la mayoría de las personas aprenderían a usar esta herramienta rápidamente.
8. Considero que el uso de esta herramienta es tedioso.
9. Me sentí muy confiado al usar la herramienta.
10. Necesité saber bastantes cosas antes de poder empezar a usar esta herramienta.

El usuario debe responder a las anteriores afirmaciones según su nivel de conformidad con los hechos expuestos, variando las respuestas desde ‘Totalmente de acuerdo’ (5) hasta ‘Totalmente en desacuerdo’ (1). Como se puede apreciar en las afirmaciones dadas, están ordenadas de forma que cinco de ellas están formuladas de forma negativa y otras cinco positivas. Con esto conseguimos evitar que el usuario sujeto seleccione la primera respuesta en todas las preguntas formuladas.

Para calcular la puntuación SUS, debemos repetir una serie de pasos para cada usuario:

1. Convertir las respuestas del usuario en las diez preguntas a puntuaciones: Si la pregunta es impar, se subtrae un punto de la valoración dada, es decir, si el usuario ha valorado la pregunta con un 5, la puntuación obtenida sería 4. Por el contrario, si la pregunta es par, la puntuación obtenida es 5 menos la valoración dada.
2. Sumar las puntuaciones obtenidas.
3. Multiplicar la puntuación total por 2.5 para obtener la puntuación individual.
4. Repetir los pasos 1 al 3 para todos los usuarios, y después calcular el promedio de todos los puntajes de los usuarios para obtener un puntaje SUS.

Aplicado a nuestro proyecto, hemos realizado las encuestas sobre un total de ocho usuarios, con un rango de edades y experiencia en las tecnologías variable. Para que los usuarios se familiarizaran con el sistema, se les pidió que realicen una serie de tareas antes de completar el formulario SUS. Entre dichas tareas contamos con el registro e identificación del usuario, publicación de una reseña o consulta, cambios de ajustes en el perfil personal, escribir un comentario y comenzar un chat privado.

En el **Anexo D** encontramos las gráficas representativas de cada pregunta, según las respuestas dadas por los usuarios. Las puntuaciones obtenidas son las mostradas en la tabla 8.

Usuario 1	92.5
Usuario 2	87.5
Usuario 3	85
Usuario 4	82.5
Usuario 5	72.5
Usuario 6	80
Usuario 7	77.5

Tabla 8: resultados obtenidos en la *System Usability Scale*.

De estos usuarios se obtiene un promedio del 79.06. Los resultados pueden interpretarse bajo diversos puntos de vista (figura 40) [14].

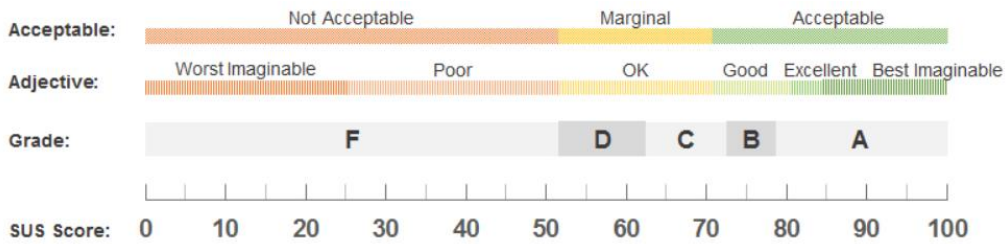


Figura 40: umbrales de la *System Usability Scale*.

A continuación, comentaremos algunas interpretaciones posibles:

Percentiles: Las puntuaciones obtenidas son convertidas en rangos de percentiles. Para medir su éxito, se compara el resultado obtenido con datos recogidos de otros sistemas que han aplicado el SUS (figura 41). El percentil promedio, por tanto, resulta ser del 68, mientras que nuestra aplicación se encuentra en el 79.

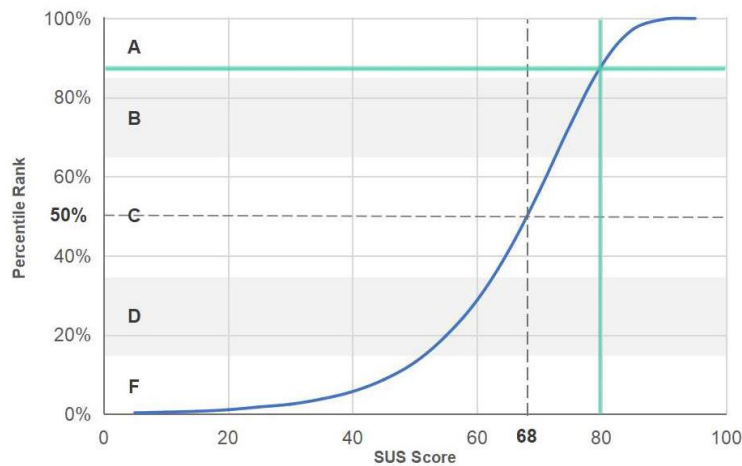


Figura 41: percentiles en la *System Usability Scale*.

Notas: (*Grade*) Similar a las puntuaciones dadas en un sistema educativo, nuestra puntuación SUS puede ser adaptada a una de las notas posibles en un rango, desde excelente (A) hasta suspenso (F). Ya que nuestra puntuación es de 79, nos encontramos con un A, es decir, la mayor puntuación.

Adjetivos: (*Adjective*) En lugar de usar puntuaciones para describir la experiencia del usuario, algunos expertos decidieron asociar las calificaciones con palabras clave (figura 61). En total, son 7 palabras, que pasan desde “Excelente” y “Pasable” hasta “Pobre”. Por ejemplo, las puntuaciones por encima de 85 son asociadas con “Excelente”, mientras que las menores de 85 y hasta 71 son “Bien”. Nuestro sistema, por tanto, está en este rango, calificado como “Bien”.

Aceptabilidad: (*Acceptability*) Se trata de otra variación para la descripción el SUS mediante palabras, ingeniado por los mismos expertos que diseñaron la interpretación mediante adjetivos comentada el punto superior. Un sistema aceptable es aquel que ha conseguido más de 70 puntos, por lo que nuestra aplicación se llega a considerar aceptable. Los sistemas inaceptables, por el contrario, han tenido una calificación menor a 51.6 puntos. La figura 61 muestra estos niveles de una forma más visual.

8. Conclusiones

A lo largo de esta memoria se han comentado todos los aspectos necesarios para llevar a cabo el desarrollo de una aplicación web desde un punto inicial, incluyendo los primeros pasos de enumeración de objetivos, estudio de mercado actual, elicitación de requisitos, desarrollo del software y mucho más.

Habiendo llegado a este punto, podemos afirmar que los objetivos iniciales han sido cumplidos, ya que se ha creado una aplicación web funcional capaz de cumplir su finalidad como punto de encuentro para las personas que necesiten compartir sus experiencias o buscar ayuda en temas relacionados con intervenciones quirúrgicas. El éxito a la hora de alcanzar nuestros objetivos viene dado por haber logrado los siguientes puntos fundamentales:

- Se han conseguido ofrecer las funcionalidades que presentan otros productos competidores, yendo más allá y añadiendo otras nuevas.
- Se ha permitido el contacto directo entre usuarios, mediante comentarios y mensajes privados, pero siempre siendo posible reportar o denunciar a otros usuarios por comportamiento inadecuado.
- Los usuarios están protegidos gracias a los términos y condiciones del servicio, política de privacidad y otros servicios descritos en la aplicación web. También se les protege mediante cifrados de contraseña en la base de datos, y otras protecciones que ofrece Django.
- Tiempo de respuesta rápido, con una interfaz atractiva y simple de usar.

Sin embargo, la aplicación está lejos de darse por finalizada, dado que siempre se pueden añadir nuevas funcionalidades que mejoren la experiencia del usuario, como por ejemplo la publicación de la aplicación, el lanzamiento de una aplicación móvil, o la adición de una agenda para concertar citas, al igual que hacen otros productos similares.

Podemos realizar la comparación mostrada en la tabla 9 para dar por concluida la investigación iniciada en el segundo capítulo sobre el estado del arte.

Productos	Google Reviews	Doctoralia	TrustPilot	Clínicas	OperRating

Características					
Comentarios	✓	✓	✓	X	✓
Mapa de las clínicas	X	✓	X	X	✓
Chat privado	X	X	✓	X	✓
Notificaciones	X	✓	✓	✓	✓
Publicar consulta	X	✓	X	X	✓
Publicar reseña	✓	✓	✓	X	✓
Dedicado total- mente a cirujías	X	X	X	✓	✓
Compartir	✓	✓	✓	X	✓
Noticias de una clínica	X	X	X	✓	✓

Tabla 9: características de OperRating en comparación con la competencia.

Llevar a cabo este trabajo de fin de grado ha supuesto un desafío, tanto académica como personalmente, teniendo que dedicar mucho esfuerzo y dedicación.

Se trata del primer proyecto a ‘gran escala’ que he desarrollado en solitario, y el primero en el que me he inmerso completamente en el *back-end*, lo cual ha derivado en un retraso inicial al no comprender el comportamiento de la mayoría de las funciones desarrolladas. Estos retrasos me han aportado una experiencia valiosa y gratificante al haber sido solventados sin tirar la toalla, de lo cual me enorgullezco.

He logrado aprender mucho sobre todas las tecnologías escogidas, incluso habiendo hecho uso de Django y React para algunas asignaturas a lo largo de la carrera, gracias a la sensación de trabajo constante proporcionada por las metodologías ágiles.

8.1 Relación del trabajo desarrollado con los estudios cursados

La realización del proyecto ha culminado con la creación de una aplicación web capaz de ofrecer una interfaz de usuario óptima ajustada a las necesidades de los usuarios. Por estos motivos, las asignaturas relacionadas con la creación y diseño de interfaces web han jugado un papel primordial en la creación de OperRating. Entre ellas podemos destacar **Diseño de sitios web** e **Interfaz persona computador**.

Durante el cuarto capítulo, dedicado al análisis del problema, se ha hecho un uso primario de conocimientos impartidos en la materia **Análisis y especificación de requisitos**. Esta asignatura, junto con **Ingeniería del Software**, nos aportan herramientas muy útiles para todos los puntos expuestos durante la memoria, desde los diagramas UML y casos de uso hasta el ejercicio de brainstorming.

Por otro lado, durante la realización del proyecto se ha hecho un uso mayoritario de tecnologías y herramientas no impartidas en la carrera, como React y Django. Sin embargo, asignaturas como **Bases de datos y sistemas de información** nos aportan el conocimiento necesario para comprender cómo crear el modelo empleado en la aplicación, encontrado en la capa de persistencia. Los patrones de diseño estudiados en **Diseño de software** nos sirven para aprender buenas prácticas de programación y patrones de diseño a aplicar, como puede ser el patrón *Singleton* implementado en el proyecto.

De entre todas las materias cursadas estos años, las que más impacto han tenido sobre este proyecto de final de carrera han sido sin duda **Proceso de software** y su continuación, **Proyecto de ingeniería software**, ya que en ellas se han recogido los conocimientos necesarios para poner en práctica las metodologías ágiles empleadas en este trabajo.

9. Referencias

- [1] LETELIER, Patricio “*TUNE-UP Software Process Metodología y herramienta de apoyo para la gestión ágil de proyectos de desarrollo y mantenimiento de software*”, 2011. [En línea]. Disponible en: <https://acortar.link/FTBfQM>
- [2] VAN CASTEREN, Wilfred. The Waterfall Model and the Agile Methodologies: A comparison by project characteristics. *Research Gate*, 2017, vol. 2, p. 1-6.
- [3] NIDAGUNDI, Padmaraj; NOVICKIS, Leonids. Introducing lean canvas model adaptation in the scrum software testing. *Procedia Computer Science*, 2017, vol. 104, p. 97-103.
- [4] GURL, Emet. SWOT analysis: A theoretical review. 2017.
- [5] ALEXANDER, Ian F.; MAIDEN, Neil (ed.). *Scenarios, stories, use cases: through the systems development life-cycle*. John Wiley & Sons, 2005.
- [6] CARR, Mahil; VERNER, June. Prototyping and software development approaches. *Department of Information Systems, City University of Hong Kong, Hong Kong*, 1997, p. 319-338.
- [7] HAMM, Matthew J. *Wireframing essentials*. Packt Publishing Ltd, 2014.
- [8] GAT, Erann, et al. On three-layer architectures. *Artificial intelligence and mobile robots*, 1998, vol. 195, p. 210.
- [9] WOLFF, Eberhard. *Microservices: flexible software architecture*. Addison-Wesley Professional, 2016.
- [10] SMITH, Connie U.; WILLIAMS, Lloyd G. Software performance antipatterns. En *Proceedings of the 2nd international workshop on Software and performance*. 2000. p. 127-136.
- [11] SMITH, Connie U.; WILLIAMS, Lloyd G. Software performance antipatterns. En *Proceedings of the 2nd international workshop on Software and performance*. 2000. p. 127-136.
- [12] CLARKE-SALT, Justin. *SQL injection attacks and defense*. Elsevier, 2009.
- [13] BANGOR, Aaron; KORTUM, Philip T.; MILLER, James T. An empirical evaluation of the system usability scale. *Intl. Journal of Human-Computer Interaction*, 2008, vol. 24, no 6, p. 574-594.
- [14] LEWIS, James R.; SAURO, Jeff. Item benchmarks for the system usability scale. *Journal of Usability Studies*, 2018, vol. 13, no 3.



Anexo A. Objetivos de Desarrollo Sostenible

Los Objetivos de Desarrollo Sostenible (ODS) representan una meta universal a alcanzar con el propósito de mejorar la calidad de vida y perspectivas en el mundo. Para ello, se pactaron un total de 17 objetivos a alcanzar en 15 años entre los estados miembros de las Naciones Unidas.

Nuestro proyecto cumple con el siguiente grado de relación con los ODS:

Objetivos de Desarrollo Sostenible	Alto	Medio	Bajo	No Procede
ODS 1. Fin de la pobreza.				X
ODS 2. Hambre cero.				X
ODS 3. Salud y bienestar.	X			
ODS 4. Educación de calidad.				X
ODS 5. Igualdad de género.				X
ODS 6. Agua limpia y saneamiento.				X
ODS 7. Energía asequible y no contaminante.				X
ODS 8. Trabajo decente y crecimiento económico.				X
ODS 9. Industria, innovación e infraestructuras.			X	
ODS 10. Reducción de las desigualdades.		X		
ODS 11. Ciudades y comunidades sostenibles.				X
ODS 12. Producción y consumo responsables.				X
ODS 13. Acción por el clima.				X
ODS 14. Vida submarina.				X
ODS 15. Vida de ecosistemas terrestres.				X
ODS 16. Paz, justicia e instituciones sólidas.				X
ODS 17. Alianzas para lograr objetivos.	X			

Dados los objetivos de desarrollo sostenibles, se relaciona el proyecto con los mismos debido a los siguientes motivos:

Salud y bienestar: El propósito principal del proyecto expuesto es brindar información relativa a intervenciones médicas a los usuarios que usen la aplicación. Este hecho está fuertemente atado a la salud del paciente, lo que hace de la tercera de las ODS uno de los principales objetivos que persigue nuestro proyecto.

Industria, innovación e infraestructuras: La idea propuesta fomenta la mejora continua en la calidad del servicio de las clínicas y centros adscritos a la aplicación mediante las calificaciones de los usuarios.

Reducción de las desigualdades: En OperRating no existen diferencias a la hora de ofrecer nuestro servicio, por lo que no se crea ningún tipo de distinción según el género, lugar de procedencia u otros factores. Cualquier persona es bienvenida a usar los

Aplicación para la gestión de reseñas y consultas de intervenciones quirúrgicas

servicios proporcionados fruto del proyecto, ya que se trata de un servicio gratuito y neutro de políticas.

Alianzas para lograr objetivos: El éxito del proyecto se centra en el sentimiento de comunidad que se pueda llegar a obtener, ya que tanto las reseñas como las consultas publicadas dependen de los usuarios que participen en la aplicación. Por tanto, es de gran importancia esta interacción entre usuarios con tal de conseguir la mejor experiencia posible.

Anexo B. Casos de uso identificados

11.1 Gestión de usuarios

Caso de uso	CU02: Identificación
Actor	Usuario no identificado
Descripción	Registra en el sistema un nuevo usuario
Relaciones	
Precondición	
Escenario principal	<ol style="list-style-type: none">1. El usuario no identificado accede a identificarse2. El usuario no identificado escribe su nombre y contraseña3. El sistema le identifica y le redirige a la página inicial, con el usuario ya identificado.
Alternativas	<ol style="list-style-type: none">3. El sistema no encuentra algún usuario o propietario creado con esa información3.1 El sistema le muestra un mensaje de error al usuario no identificado

Tabla 10: CU02, identificación.

Caso de uso	CU03: Enviar mensaje
Actor	Usuario
Descripción	El usuario envía un mensaje a otro usuario
Relaciones	<i>Extend CU20: se ha accedido al perfil</i>
Precondición	
Escenario principal	<ol style="list-style-type: none">1. El usuario pulsa sobre el botón de iniciar chat2. El sistema muestra el interfaz de mensajería3. El usuario escribe y manda su mensaje4. El sistema <i>Outlook</i> envía una notificación por correo al receptor avisándole de que alguien quiere enviarle un mensaje
Alternativas	<ol style="list-style-type: none">1. El usuario ya había interactuado con el receptor del mensaje1.1 Los mensajes anteriores se muestran en el chat1.2 El usuario escribe y manda su mensaje1.3 El sistema <i>Outlook</i> no envía más notificaciones, al haberse aprobado previamente el chat

Tabla 11: CU03, enviar mensaje.

Caso de uso	CU04: Cambiar datos personales
Actor	Usuario
Descripción	Permite al usuario acceder a los datos de su perfil personal
Relaciones	<i>Extend CU06 el usuario ha accedido a su perfil</i>
Precondición	
Escenario principal	<ol style="list-style-type: none">1. El usuario pulsa en "Mis datos"2. El sistema muestra sus datos personales
Alternativas	

Tabla 12: CU04, cambiar datos personales.

Caso de uso	CU05: Darse de baja
Actor	Usuario no identificado
Descripción	Permite al usuario borrar su cuenta personal
Relaciones	<i>Extend:</i> CU04, se ha accedido al perfil del usuario
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario pulsa sobre el botón de borrar perfil 2. El sistema pide confirmación de la decisión 3. El usuario confirma la decisión 4. El sistema borra al usuario y se cierra la sesión actual (CU09)
Alternativas	

Tabla 13: CU05, darse de baja.

Caso de uso	CU06: Acceder a mi perfil
Actor	Usuario
Descripción	Acceso al perfil del usuario, donde ver publicaciones previamente creados
Relaciones	
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede a su perfil desde la barra de navegación o una publicación suya 2. El sistema muestra las publicaciones que el usuario ha creado ordenados por fecha de adición, y varias secciones (publicaciones 2. guardados y datos personales)
Alternativas	

Tabla 14: CU06, acceder a mi perfil.

Caso de uso	CU09: Salir de la sesión actual
Actor	Usuario
Descripción	Salir de la sesión actual de usuario
Relaciones	
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario pulsa sobre cerrar sesión 2. El sistema le redirige a la página principal, pero sin credenciales guardadas
Alternativas	

Tabla 15: CU09, salir de la sesión actual.

Caso de uso	CU20: Acceder al perfil de un usuario
Actor	Usuario no identificado
Descripción	Permite al usuario acceder al perfil de otro usuario
Relaciones	<i>Extend</i> CU29 se ha hecho uso de los filtros de búsqueda, CU32, se ha buscado la publicación y CU10, la publicación está siendo visualizada
Precondición	Existen usuarios.
Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona el nombre del usuario del cual quiere ver su perfil 2. El sistema muestra otras publicaciones que ha creado dicho usuario
Alternativas	

Tabla 16: CU20, acceder al perfil de un usuario.

Caso de uso	CU33: Registro de propietario
Actor	Usuario no identificado
Descripción	Registra en el sistema un nuevo propietario
Relaciones	<i>Extend</i> CU21 el usuario ha consultado los términos de servicio
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario no identificado accede a crear propietario 2. El usuario no identificado rellena todos los campos personales y sobre su clínica 3. El sistema muestra un mensaje de éxito a la hora de crear al propietario y su clínica. 4. El servicio de correo envía un email al usuario no identificado 5. El sistema redirige al usuario no identificado a la página de inicio de sesión como propietario
Alternativas	<ol style="list-style-type: none"> 3. El sistema encuentra algún usuario, propietario o clínica creado con la misma información (mismo nombre o correo) <ol style="list-style-type: none"> 3.1 El sistema le muestra un mensaje de error al usuario no identificado

Tabla 17: CU34, registro de propietario.

Caso de uso	CU34: Visualizar publicaciones guardadas
Actor	Usuario
Descripción	Acceso a las publicaciones previamente guardadas del usuario
Relaciones	<i>Extend</i> CU06 el usuario ha accedido a su perfil
Precondición	Ninguna
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede a visualizar publicaciones guardados desde su perfil 2. El sistema muestra las publicaciones ordenadas por fecha de adición
Alternativas	No existen <i>publicaciones</i> guardados, por lo que el sistema no muestra nada

Tabla 18: CU34, visualizar publicaciones guardadas.

11.2 Gestión de publicaciones

Caso de uso	CU11: Compartir publicación
Actor	Usuario no identificado
Descripción	Permite al usuario compartir de forma cómoda la publicación
Relaciones	<i>Extend</i> CU10, la publicación es visualizada
Precondición	
Escenario principal	El usuario pulsa sobre el icono de compartir El sistema muestra varias opciones para compartir la publicación El usuario selecciona una de las opciones para compartir la publicación
Alternativas	

Tabla 19: CU11, compartir publicación.

Caso de uso	CU12: Crear reseña
Actor	Usuario
Descripción	Permite al usuario crear una publicación de tipo reseña
Relaciones	<i>Include</i> CU19
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede a crear una publicación y selecciona el tipo reseña 2. El sistema muestra los campos necesarios para crear una reseña 3. El usuario rellena los campos y escribe la descripción 4. El sistema guarda la publicación y procede a ser mostrado en el <i>feed</i>. 5. El usuario es redirigido al <i>feed</i> después de ver un mensaje de confirmación
Alternativas	<ol style="list-style-type: none"> 3. Si la clínica en la que se realizó la intervención el usuario no se encuentra disponible, éste puede solicitar su inclusión 4. Si algún campo no es correcto, se muestra un mensaje de error

Tabla 20: CU12, crear reseña.

Caso de uso	CU14: Escribir comentario
Actor	Usuario
Descripción	Permite al usuario crear un comentario en una publicación.
Relaciones	<i>Extend</i> CU10 la publicación está siendo visualizada, <i>Include</i> CU16
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede a una <i>publicación</i>. 2. El usuario escribe un comentario 3. El sistema registra el comentario escrito y lo muestra en la publicación, así como la fecha y el usuario autor del comentario
Alternativas	

Tabla 21: CU14, escribir comentario.

Caso de uso	CU15: Guardar <i>publicación</i>
Actor	Usuario
Descripción	Permite al usuario guardar <i>publicaciones</i> en su perfil
Relaciones	
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario pulsa sobre el icono de "Guardar publicación" 2. El sistema guarda la publicación en la sección de <i>publicaciones</i> guardados del perfil personal del usuario
Alternativas	

Tabla 22: CU15, guardar publicación.

Caso de uso	CU16: Notificaciones sobre <i>publicación</i>
Actor	Usuario Servicio de correo
Descripción	El usuario recibe notificaciones debido a una interacción con una publicación que ha creado
Relaciones	
Precondición	El usuario que recibe la notificación es el autor de la publicación.
Escenario principal	<ol style="list-style-type: none"> 1. El sistema envía un correo avisando al usuario de que alguien ha comentado en su publicación
Alternativas	

Tabla 23: CU16, notificaciones sobre publicación.

Caso de uso	CU26: Borrar <i>publicación</i>
Actor	Usuario
Descripción	Permite al usuario borrar una publicación que haya creado
Relaciones	<i>Extend</i> : CU10, la <i>publicación</i> es visible
Precondición	Debe haber creado alguna publicación.
Escenario principal	<ol style="list-style-type: none"> 1. El usuario pulsa sobre el botón de borrar publicación 2. El sistema pide confirmación de la decisión 3. El usuario confirma la decisión 4. El sistema borra la publicación
Alternativas	

Tabla 24: CU26, borrar publicación

Caso de uso	CU35: Crear consulta
Actor	Usuario
Descripción	Permite al usuario crear una publicación
Relaciones	
Precondición	<i>Include</i> CU19
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede a crear una publicación y selecciona el tipo consulta 2. El sistema muestra los campos necesarios para crear una consulta 3. El usuario rellena los campos y escribe la descripción 4. El sistema guarda la publicación y procede a ser mostrado en el <i>feed</i>. 5. El usuario es redirigido al <i>feed</i> después de ver un mensaje de confirmación
Alternativas	<ol style="list-style-type: none"> 3. Si la clínica sobre la que el usuario quiere interrogar no se encuentra disponible, éste puede solicitar su inclusión 4. Si algún campo no es correcto, se muestra un mensaje de error

Tabla 25: CU35, crear consulta.

Caso de uso	CU37: Eliminar comentario
Actor	Usuario
Descripción	Permite al usuario eliminar un comentario que haya escrito en una publicación.
Relaciones	Extend CU10, la publicación está siendo visualizada
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede a una publicación 2. El usuario elige un comentario que haya escrito y lo marca como eliminado 3. El sistema registra el comentario eliminado y desaparece de la publicación
Alternativas	

Tabla 26: CU37, eliminar comentario.

Caso de uso	CU39: Acceder al perfil de una clínica
Actor	Usuario no identificado
Descripción	Permite al usuario acceder al perfil de una clínica
Relaciones	Extend CU32, el usuario ha visto la publicación de la clínica y CU29, el usuario ha buscado la publicación
Precondición	Existen clínicas.
Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona el nombre de la clínica de la cual quiere ver su perfil 2. El sistema muestra datos sobre dicha clínica
Alternativas	

Tabla 27: CU39, acceder al perfil de una clínica.

11.3 Gestión de clínicas

Caso de uso	CU19: Crear clínica
Actor	Usuario
Descripción	Permite a un usuario crear una clínica.
Relaciones	
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede a crear clínica 2. El sistema muestra los campos a rellenar 3. El usuario rellena los campos 4. El sistema muestra un mensaje de éxito a la hora de crear la clínica 5. El administrador verifica los datos 5. El sistema crea la clínica y su perfil
Alternativas	

Tabla 28: CU19, crear clínica.

Caso de uso	CU22: Subir noticia
Actor	Propietario
Descripción	Permite al propietario publicar una noticia sobre su clínica
Relaciones	
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El propietario accede a subir una noticia 2. El sistema muestra los campos a rellenar 3. El propietario rellena los campos 4. El sistema crea la noticia
Alternativas	

Tabla 29: CU22, subir noticia.

Caso de uso	CU23: Eliminar noticia
Actor	Propietario
Descripción	Permite a un propietario borrar una noticia
Relaciones	
Precondición	Existe alguna noticia sobre una clínica que el propietario posee
Escenario principal	<ol style="list-style-type: none"> 1. El propietario accede al perfil de su clínica y selecciona una noticia para eliminar 2. El sistema elimina la noticia
Alternativas	

Tabla 30: CU23, eliminar noticia.

Caso de uso	CU27: Visualizar localización de la clínica
Actor	Usuario no identificado Servicio de mapas
Descripción	Mapa que muestra la localización de la clínica
Relaciones	
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede a visualizar la clínica 2. El sistema muestra un mapa con la localización de la clínica
Alternativas	

Tabla 31: CU27, visualizar localización de la clínica

Caso de uso	CU29: Usar filtros de búsqueda
Actor	Usuario no identificado
Descripción	Permite al usuario hacer búsquedas en el sistema basándose en sus preferencias (tipo de intervención, clínica, etc.)
Relaciones	<i>Include</i> CU28, CU18, CU27
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede al <i>feed</i> 2. El sistema muestra los filtros de búsqueda 3. El usuario introduce los filtros convenientes 4. El sistema muestra publicaciones, clínicas o usuarios relacionados con los filtros introducidos
Alternativas	4. El sistema no encuentra nada relacionado con la búsqueda y muestra el <i>feed</i> vacío

Tabla 32: CU29, usar filtros de búsqueda

Caso de uso	CU40: Identificarse como propietario
Actor	Usuario
Descripción	Permite a un usuario identificarse como el propietario de una clínica ya creada Precondición: existen clínicas
Relaciones	Extend CU39, se está visualizando la clínica
Precondición	Existen clínicas
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede al perfil de la clínica cuyo propietario no está registrado en el sistema 2. El usuario pulsa sobre Identificación de propietario 3. El sistema muestra un formulario identificativo 4. El usuario rellena el formulario y lo envía 5. El sistema registra los datos
Alternativas	

Tabla 33: CU40, identificarse como propietario.

Caso de uso	CU42: Editar clínica
Actor	Propietario
Descripción	Permite a un propietario editar información sobre su clínica (lista de doctores, descripción, etc.)
Relaciones	
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El propietario accede al perfil de su clínica y selecciona la opción de editar datos 2. El sistema muestra los campos editables 3. El propietario edita los datos necesarios 4. El sistema guarda los cambios
Alternativas	

Tabla 34: CU42, editar clínica.

11.4 Moderación

Caso de uso	CU08: Bloquear usuario
Actor	Usuario
Descripción	El usuario bloquea las interacciones que otro usuario quiera tener con él
Relaciones	<i>Extend:</i> CU20, se ha accedido al perfil del usuario que se desea bloquear
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario pulsa sobre el botón de bloquear usuario 2. El sistema bloquea los mensajes directos del usuario bloqueado
Alternativas	

Tabla 35: CU08, bloquear usuario.

Caso de uso	CU38: Moderar comentario
Actor	Administrador
Descripción	Permite a un administrador eliminar comentarios publicados en una publicación
Relaciones	<i>Include</i> CU10, la publicación está siendo visualizada
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El administrador pulsa sobre el botón de eliminar comentario asociado a un comentario 2. El sistema elimina el mensaje seleccionado
Alternativas	

Tabla 36: CU38, moderar comentario.

Caso de uso	CU41: Verificar propietario
Actor	Administrador
Descripción	Permite a un administrador verificar que un usuario es propietario de una clínica
Relaciones	
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El administrador pulsa sobre el botón de verificar propietario asociado a una clínica 2. El sistema registra la clínica como verificada
Alternativas	

Tabla 37: CU41, verificar propietario.

11.5 Feeds

Caso de uso	CU21: Consultar términos de servicio
Actor	Usuario no identificado
Descripción	Permite al usuario conocer los términos y condiciones de servicio
Relaciones	
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede al enlace de Términos y condiciones del servicio 2. El sistema muestra la información legal de OperRating
Alternativas	

Tabla 38: CU21, consultar términos de servicio.

Caso de uso	CU28: Visualizar <i>feed</i>
Actor	Usuario no identificado
Descripción	Permite al usuario visualizar un <i>feed</i> y las publicaciones y de éste
Relaciones	<i>Include</i> CU32, CU31
Precondición	Existe alguna publicación creada
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede a un <i>feed</i> 2. El sistema muestra su contenido
Alternativas	

Tabla 39: CU28, visualizar *feed*.

Caso de uso	CU31: Visualizar mapa
Actor	Usuario no identificado Servicio de mapas
Descripción	Permite al usuario tener una vista general de la ubicación de las clínicas registradas en el sistema
Relaciones	<i>Extend</i> CU29 las clínicas mostradas dependen de los filtros
Precondición	
Escenario principal	El sistema muestra el interfaz de un mapa dependiendo de dónde esté el usuario y de los filtros aplicados
Alternativas	Si el usuario no ha activado la localización se muestra un punto predeterminado

Tabla 40: CU31, visualizar mapa.

Caso de uso	CU32: Previsualizar publicación
Actor	Usuario no identificado
Descripción	Resume una publicación mostrando su título, valoraciones (en caso de ser reseña), y parte de la descripción
Relaciones	<i>Extend</i> CU29 las publicaciones dependen de los filtros, <i>Include</i> CU11
Precondición	
Escenario principal	<ol style="list-style-type: none"> 1. El usuario accede a un <i>feed</i> 2. El sistema muestra la previsualización de una publicación dependiendo de los filtros aplicados
Alternativas	

Tabla 41: CU32, previsualizar publicación.

Anexo C. Listado de pruebas de aceptación

El árbol de nodos dado en la figura 65 y las pruebas de aceptación (PA) de las unidades de trabajo (UT) llevadas a cabo son el resultado obtenido de la fase de especificación de requisitos. Esta estructura es definida al inicio del desarrollo de la parte software, conectando todas las unidades de trabajo entre sí mediante los nodos creados, y escribiendo las pruebas de aceptación antes de iniciar cada unidad de trabajo.

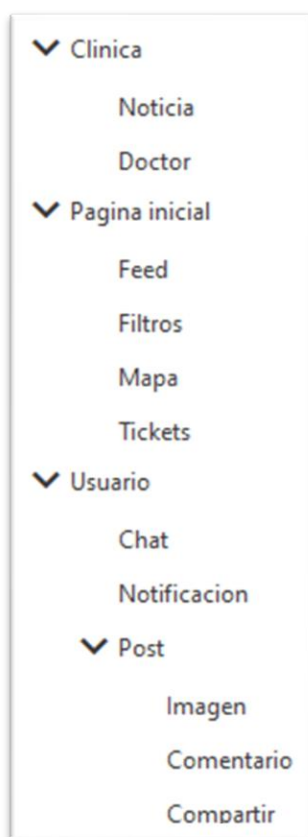


Figura 42: estructura de nodos,

Nodo Clínica		
Sprint 1	UT 337 - Crear clínica	<p>PA 145 - Crear clínica</p> <p>CONDICIONES</p> <p>El usuario se ha registrado e identificado</p> <p>PASOS</p> <p>El usuario accede a crear una clínica</p> <p>El usuario rellena los datos</p> <p>RESULTADO</p> <p>El sistema crea una clínica.</p>
Sprint 2	UT 338 - Editar clínica	<p>PA 150 - Editar clínica</p> <p>CONDICIONES</p> <p>El usuario identificado es el dueño</p> <p>PASOS</p> <p>El usuario accede los ajustes del perfil de su clínica</p> <p>El usuario modifica la información del centro: descripción, dirección, personal, etc.</p> <p>RESULTADO</p> <p>El sistema registra los cambios</p> <p>Si se elimina la clínica las reseñas y preguntas sobre ésta también, pero si se elimina un Doctor no</p>
		<p>PA 148 - Visualizar perfil clínica</p> <p>CONDICIONES</p> <p>PASOS</p> <p>El usuario accede a una clínica</p> <p>RESULTADO</p> <p>Se muestran los datos sobre esta: descripción, personal registrado y valoración del público</p>
Sprint 3	UT 354 – Verificar propietarios	<p>PA 159 - Reclamar como propietario</p> <p>CONDICIONES</p> <p>La clínica fue registrada por un usuario que no era su propietario</p> <p>PASOS</p> <p>El propietario se dirige al perfil de la clínica y selecciona 'reclamar como propietario'</p> <p>El sistema pide datos para verificarlo</p> <p>El propietario cumplimenta las verificaciones</p> <p>RESULTADO</p> <p>El centro queda registrado a nombre del propietario</p>
		<p>PA 172 - Verificar datos</p> <p>CONDICIONES</p> <p>Existe una clínica con datos sin verificar</p> <p>PASOS</p> <p>Un usuario administrador revisa los datos de la clínica y cambia su estado a 'Verificada'</p> <p>RESULTADO</p> <p>Los datos de la clínica son verificados</p>

Tabla 42: pruebas de aceptación nodo Clínica.

Nodo Noticia		
<i>Sprint 3</i>	UT 352 - Noticias de clínicas	<p>PA 158 - Crear noticia</p> <p>CONDICIONES El usuario identificado es el propietario de la clínica</p> <p>PASOS El usuario accede a su clínica y selecciona la opción de añadir noticia. El usuario rellena los campos</p> <p>RESULTADO La noticia se crea y se muestra en el <i>feed</i> de clínicas, así como en el perfil de esta.</p>
		<p>PA 173 - Eliminar noticia</p> <p>CONDICIONES Existe alguna noticia</p> <p>PASOS El propietario del centro va al perfil del centro y selecciona la noticia a eliminar</p> <p>RESULTADO La noticia se elimina</p>
		<p>PA 183 - Visualizar noticia</p> <p>CONDICIONES Existen noticias en la clínica</p> <p>PASOS El usuario accede al perfil de la clínica</p> <p>RESULTADO Se muestran las noticias</p>

Tabla 43: pruebas de aceptación nodo Noticia.

Nodo Doctor		
<i>Sprint 1</i>	UT 337 - Crear clínica	<p>PA 146 - Asociar doctores a clínica</p> <p>CONDICIONES El usuario autenticado es propietario de una clínica</p> <p>PASOS El usuario accede a añadir personal a la clínica y rellena los formularios</p> <p>RESULTADO Los doctores quedan registrados en el sistema y en la clínica</p>

Tabla 44: pruebas de aceptación nodo Doctor.

Nodo Página inicial		
<i>Sprint 3</i>	UT 341 - Página sobre temas legales	PA 182 - Registro enlaza a página sobre políticas CONDICIONES PASOS El usuario accede a registrarse y pulsa sobre la opción de 'He leído y acepto las condiciones del servicio' RESULTADO Dicha opción redirige al usuario a la página sobre temas legales

Tabla 45: pruebas de aceptación nodo Página inicial.

Nodo Feed		
<i>Sprint 1</i>	UT 335 - Feed	PA 109 - visualización de publicaciones CONDICIONES Existe alguna publicación PASOS El usuario accede al <i>feed</i> RESULTADO Se muestran las publicaciones existentes, ordenados por fecha de creación

Tabla 46: pruebas de aceptación nodo *Feed*.

Nodo Filtros		
<i>Sprint 2</i>	UT 339 - Búsqueda con filtros	PA 133 - Búsqueda con filtros en página inicial CONDICIONES El usuario no tiene por qué haber iniciado sesión PASOS El usuario se dirige a la página principal El usuario selecciona o escribe en algún campo de los filtros RESULTADO El usuario es redirigido al <i>feed</i> correspondiente con los filtros aplicados
		PA 134 - Búsqueda con filtros en <i>feed</i> CONDICIONES El usuario no tiene por qué haber iniciado sesión PASOS El usuario se dirige a uno de los <i>feeds</i> . El sistema muestra todas las publicaciones El usuario elige o escribe en alguno de los filtros de búsqueda RESULTADO El sistema muestra las publicaciones relacionadas con los filtros

		<p>PA 149 - Búsqueda de clínica</p> <p>CONDICIONES</p> <p>PASOS</p> <p>Se accede al <i>feed</i> de clínicas y se hace uso de los filtros de búsqueda.</p> <p>RESULTADO</p> <p>Se muestran las clínicas que se adaptan a la búsqueda</p>
--	--	--

Tabla 47: pruebas de aceptación nodo Filtros.

Nodo Mapa		
<i>Sprint 2</i>	UT 340 - Mapa interactivo	<p>PA 135 - Ubicación inicial del mapa</p> <p>CONDICIONES</p> <p>PASOS</p> <p>El usuario accede a la página principal</p> <p>RESULTADO</p> <p>La ubicación del mapa es la posición del usuario</p> <p>Si el usuario no ha activado la ubicación, se mostrará una predeterminada</p>
		<p>PA 136 - Ubicación clínica</p> <p>CONDICIONES</p> <p>Existe alguna clínica</p> <p>PASOS</p> <p>El usuario accede al perfil de una clínica</p> <p>RESULTADO</p> <p>El sistema muestra la ubicación de la clínica</p>
		<p>PA 137 - Ubicación de reseña</p> <p>CONDICIONES</p> <p>Existe alguna reseña</p> <p>PASOS</p> <p>El usuario accede a una reseña</p> <p>RESULTADO</p> <p>El sistema muestra la ubicación de la clínica sobre la cual se ha escrito la reseña</p>
		<p>PA 138 - Ubicación en consulta</p> <p>CONDICIONES</p> <p>El usuario está identificado</p> <p>PASOS</p> <p>El usuario accede a crear una consulta y elige una ubicación</p> <p>RESULTADO</p> <p>El sistema registra la provincia y comunidad de la consulta, y la muestra en el mapa del <i>feed</i> de consultas</p>

Tabla 48: pruebas de aceptación nodo Mapa.

Nodo Tickets		
<i>Sprint 3</i>	UT 342 - Formulario de contacto	<p>PA 179 - Envío de formulario</p> <p>CONDICIONES</p> <p>El usuario puede haberse identificado</p> <p>PASOS</p> <p>El usuario accede a la página de envío de tickets</p> <p>El usuario escribe un mensaje y lo envía</p> <p>RESULTADO</p> <p>El equipo de OperRating recibe un e-mail con el mensaje</p>

Tabla 49: pruebas de aceptación nodo *Tickets*.

Nodo Usuario		
<i>Sprint 1</i>	UT 330 - Inicio de sesión	<p>PA 106 - Inicio de sesión exitoso</p> <p>CONDICIONES</p> <p>El usuario se ha registrado</p> <p>PASOS</p> <p>El usuario introduce sus datos en la ventana de inicio de sesión</p> <p>RESULTADO</p> <p>El usuario es identificado y se le permite acceder a más funcionalidades</p>
	UT 331 - Registro usuario	<p>PA 104 - Registro exitoso</p> <p>CONDICIONES</p> <p>PASOS</p> <p>El usuario accede a registrarse y rellena los datos</p> <p>RESULTADO</p> <p>Sus datos quedan registrados</p> <p>No pueden existir varios usuarios con el mismo email o nombre</p>
		<p>PA 105 - Verificación email</p> <p>CONDICIONES</p> <p>El usuario se ha registrado</p> <p>PASOS</p> <p>El usuario recibe un correo a la dirección que escribió a la hora de registrarse</p> <p>El usuario sigue los pasos detallados para su verificación</p> <p>RESULTADO</p> <p>El usuario es verificado</p>
<i>Sprint 2</i>	UT 343 - Perfil usuario	<p>PA 156 - Visualización perfil personal</p> <p>CONDICIONES</p> <p>El usuario se ha identificado</p> <p>PASOS</p> <p>El usuario accede a su perfil</p> <p>RESULTADO</p> <p>El sistema muestra sus reseñas y consultas publicadas, comentarios escritos, publicaciones guardados y ocultos. También</p>

		se ofrece la opción de cambiar datos personales
		PA 157 - Visualizar perfil de otro usuario CONDICIONES PASOS El usuario accede al perfil de otro usuario RESULTADO El sistema muestra sus reseñas y consultas publicadas, y comentarios escritos.
<i>Sprint 3</i>	UT 349 - Darse de baja	PA 177 - Eliminar cuenta CONDICIONES El usuario se ha identificado PASOS El usuario accede a los ajustes de su perfil y selecciona eliminar cuenta RESULTADO Las publicaciones, comentarios y clínicas relacionadas con el usuario son eliminadas OBSERVACIONES
	UT 350 - Reportar usuario	PA 181 - Reportar CONDICIONES El usuario debe estar identificado PASOS El usuario accede al perfil del usuario el cual quiere reportar RESULTADO El usuario no recibirá más mensajes del usuario al cual ha bloqueado y se enviará la información del reporte al equipo de OperRating OBSERVACIONES
	UT 351 - Bloquear usuario	PA 180 - Bloquear usuario CONDICIONES El usuario debe estar identificado PASOS El usuario accede al perfil del usuario el cual quiere bloquear RESULTADO El usuario no recibirá más mensajes del usuario al cual ha bloqueado OBSERVACIONES

Tabla 50: pruebas de aceptación nodo Usuario.

Nodo Chat		
<i>Sprint 3</i>	UT 347 - Chat privado	PA 175 - Enviar mensaje CONDICIONES El usuario está identificado PASOS El usuario se dirige al perfil de otro usuario y selecciona 'abrir chat' El usuario escribe un mensaje y lo envía RESULTADO El mensaje se envía al usuario receptor

Tabla 51: pruebas de aceptación nodo *Chat*.

Nodo Notificación		
<i>Sprint 2</i>	UT 344 - Comentarios	PA 142 - Notificación en correo CONDICIONES PASOS Alguien escribe un comentario en una publicación del usuario RESULTADO El usuario recibe una notificación en su correo
		PA 144 - Notificación propietario de clínica CONDICIONES PASOS Un usuario escribe una reseña sobre una clínica RESULTADO El propietario recibe una notificación
<i>Sprint 3</i>	UT 347 - Chat privado	PA 174 - Notificación mensaje CONDICIONES PASOS El usuario ha recibido un mensaje RESULTADO El usuario recibe una notificación por correo y en su bandeja de entrada

Tabla 52: pruebas de aceptación nodo Notificación.

Nodo Publicación		
<i>Sprint 1</i>	UT 333 - Crear publicación reseña	PA 107 - Creación reseña CONDICIONES PASOS El usuario accede a crear una publicación y elige el tipo reseña El usuario rellena los campos RESULTADO La publicación se guarda en el sistema

	UT 334 - Crear publicación consulta	PA 108 - Creación consulta CONDICIONES PASOS El usuario accede a crear una publicación y elige el tipo consulta El usuario rellena los campos RESULTADO La publicación se guarda en el sistema
<i>Sprint 2</i>	UT 346 - Guardar publicación	PA 152 - Guardar publicación CONDICIONES El usuario está identificado PASOS El usuario accede a una publicación y lo marca como guardado RESULTADO La publicación se muestra en su perfil en la pestaña de 'publicaciones guardadas'
		PA 153 - Desmarcar publicación guardado CONDICIONES El usuario está identificado PASOS El usuario accede a una publicación que tenía marcado como guardado y lo vuelve a marcar RESULTADO La publicación se elimina de su lista de publicaciones guardados

Tabla 53: pruebas de aceptación nodo Publicación.

Nodo Imagen		
<i>Sprint 1</i>	UT 332 - Añadir imagen a publicación	PA 147 - Imagen añadida CONDICIONES PASOS El usuario accede a crear una publicación y añade una imagen RESULTADO La imagen se guarda y se muestra en el <i>feed</i>

Tabla 54: pruebas de aceptación nodo Imagen.

Nodo Comentario		
<i>Sprint 2</i>	UT 344 - Comentarios	PA 139 - Escribir comentario CONDICIONES El usuario debe haber iniciado sesión PASOS El usuario accede a una publicación y escribe un comentario RESULTADO

		<p>El comentario se queda registrado en la publicación</p> <p>El orden de los comentarios es por fecha de creación</p>
		<p>PA 140 - Eliminar comentario</p> <p>CONDICIONES</p> <p>El usuario que desea borrar el comentario es su autor</p> <p>PASOS</p> <p>El usuario se dirige a la publicación del cual quiere borrar un comentario que escribió y selecciona sobre su comentario la opción de eliminar</p> <p>RESULTADO</p> <p>Su comentario se elimina de la publicación</p>
<i>Sprint 3</i>	UT 353 - Moderar comentarios	<p>PA 176 - Eliminar comentario</p> <p>CONDICIONES</p> <p>El usuario identificado es administrador</p> <p>PASOS</p> <p>El administrador se dirige a una publicación y selecciona un comentario a borrar</p> <p>RESULTADO</p> <p>El comentario queda marcado como eliminado por un administrador</p>

Tabla 55: pruebas de aceptación nodo Comentario.

Nodo Compartir		
<i>Sprint 3</i>	UT 348 - Compartir publicación	<p>PA 178 - Compartir por plataforma externa</p> <p>CONDICIONES</p> <p>PASOS</p> <p>El usuario accede a una publicación y pulsa sobre compartir</p> <p>RESULTADO</p> <p>Se crea un enlace de la publicación a través de la plataforma seleccionada</p>

Tabla 56: pruebas de aceptación nodo Compartir.

Anexo D. Formulario escala de usabilidad

El presente formulario tiene como objetivo medir la usabilidad de la aplicación web OperRating mediante una serie de preguntas (10 en total) relacionadas con la aplicación.

Todas las respuestas son numeradas del 1 al 5, siendo **1** fuertemente en **desacuerdo**, y **5** fuertemente **de acuerdo**.

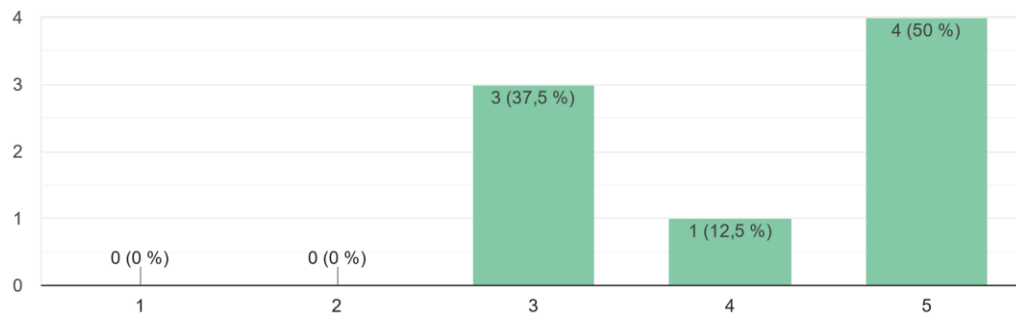
Se ha contado con un total de ocho usuarios a lo largo de tres días. A dichos usuarios se les pidió que realizaran las siguientes tareas con tal de familiarizarse con el sistema:

- Registro e inicio de sesión.
- Búsqueda de una publicación y escritura de un comentario en ella.
- Publicar una consulta o reseña.
- Enviar un mensaje directo a otro usuario.
- Modificar datos personales (nombre de usuario o imagen).
-

Todos los usuarios fueron capaces de cumplir las tareas propuestas sin dificultades, lo cual se ve reflejado en las respuestas dadas en el formulario. La encuesta realizada refleja diversidad en términos de la edad y género de los participantes, siendo que el colaborador más joven cuenta con una edad de veintitrés años, mientras que el más adulto con sesenta y tres. En cuanto al género de los usuarios encuestados, tres de ellos fueron hombres, mientras que los cinco restantes fueron mujeres.

Me gustaría usar esta herramienta frecuentemente.

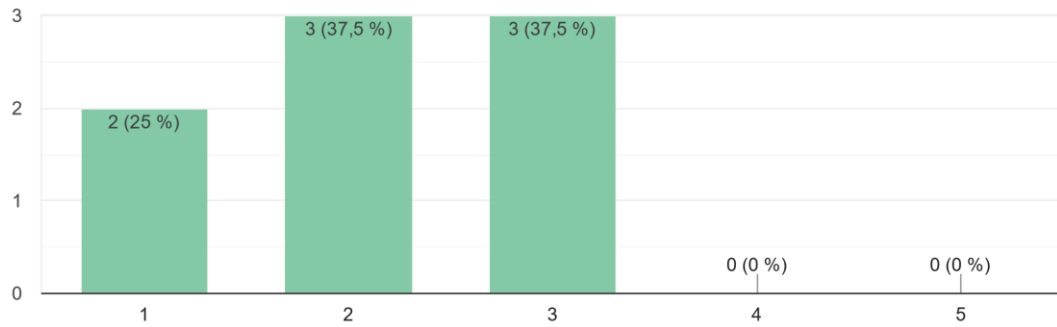
8 respuestas



Aplicación para la gestión de reseñas y consultas de intervenciones quirúrgicas

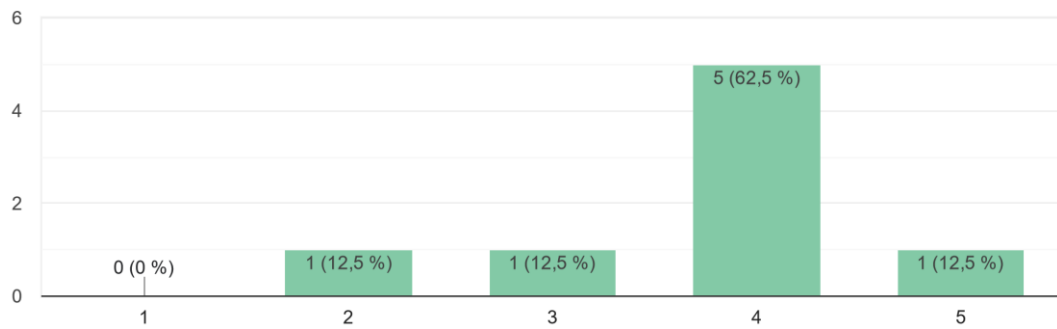
Considero que esta herramienta es innecesariamente compleja

8 respuestas



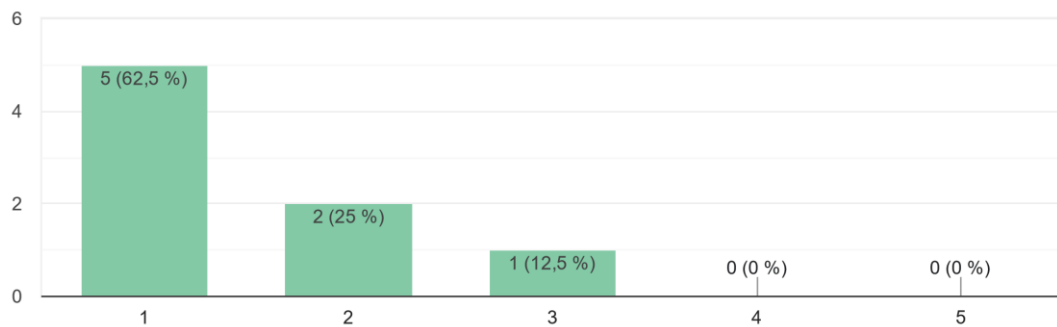
Considero que la herramienta es fácil de usar.

8 respuestas



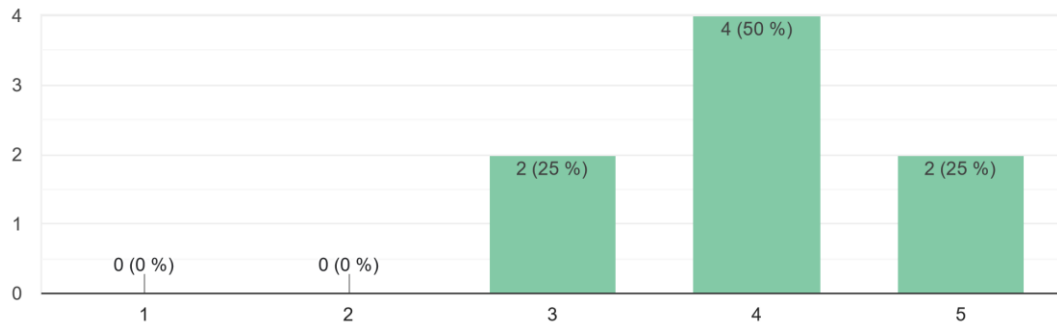
Considero necesario el apoyo de personal experto para poder utilizar esta herramienta

8 respuestas



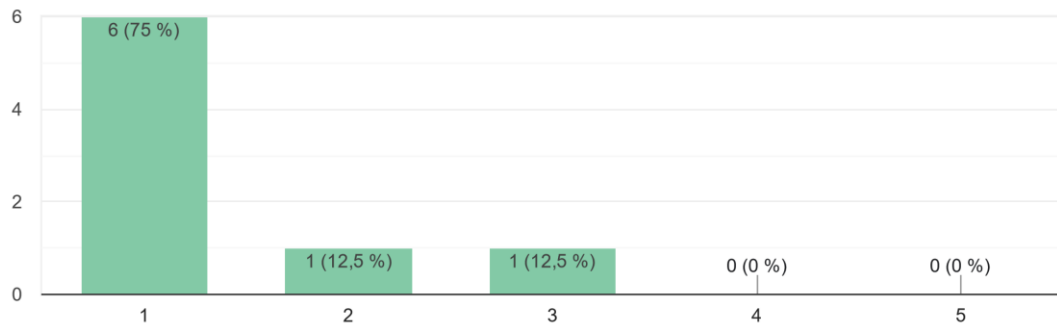
Considero que las funciones de la herramienta están bien integradas

8 respuestas



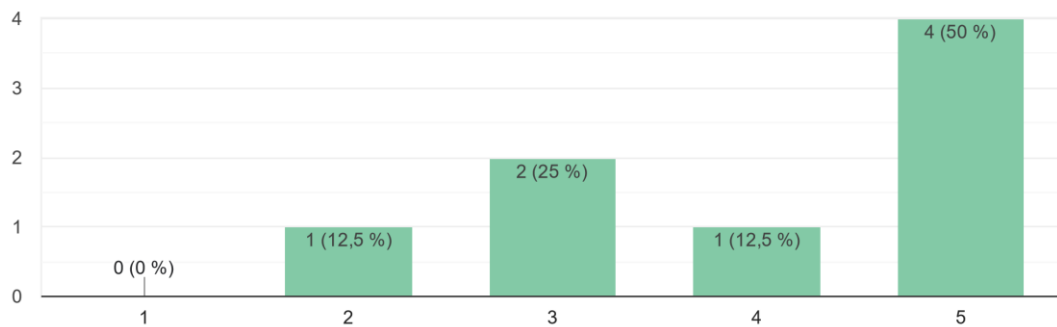
Considero que la herramienta presenta muchas contradicciones

8 respuestas



Imagino que la mayoría de las personas aprenderían a usar esta herramienta rápidamente

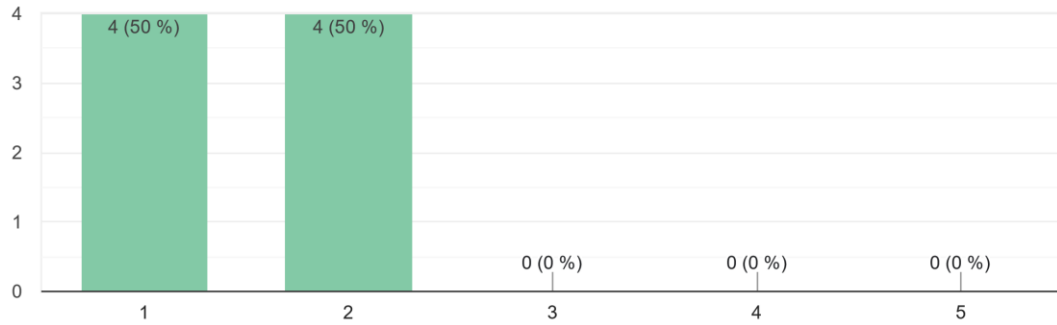
8 respuestas



Aplicación para la gestión de reseñas y consultas de intervenciones quirúrgicas

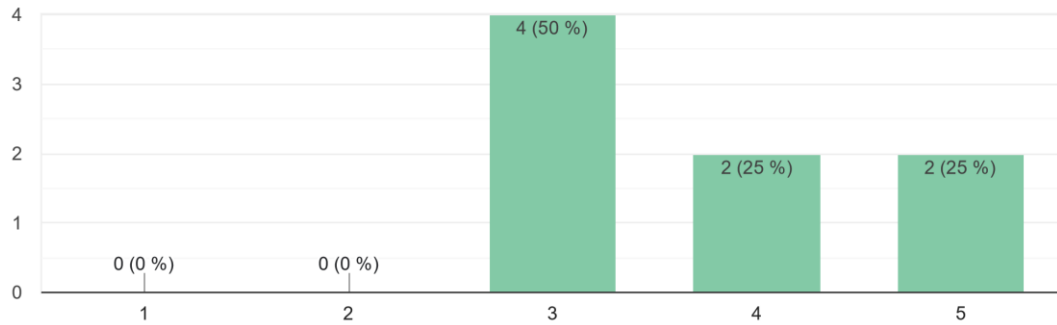
Considero que el uso de esta herramienta es tedioso

8 respuestas



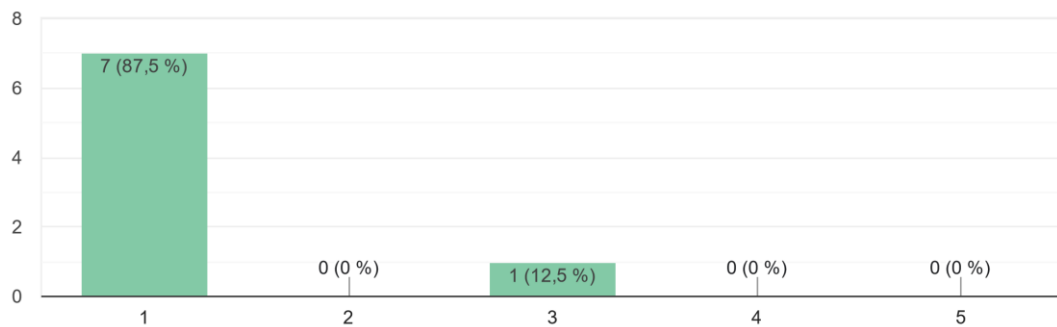
Me sentí muy confiado al usar la herramienta

8 respuestas



Necesité saber bastantes cosas antes de poder empezar a usar esta herramienta

8 respuestas



Anexo E. Manual de usuario

Para repasar la conclusión del producto, nos haremos pasar por un usuario final que desea realizar diversas tareas en nuestra aplicación, pasado así por los diversos escenarios posibles que hemos creado a lo largo del proyecto.

Primeramente, al acceder a la aplicación por primera instancia nos encontramos con la página de inicio, la cual nos muestra en la parte superior los enlaces para los *feeds* principales, los cuales nos muestran las reseñas y consultas publicadas respectivamente, y los centros existentes. También, en la misma barra de navegación, contamos con los enlaces para la identificación o registro del usuario. En la parte inferior a la barra de navegación localizamos unos filtros de búsqueda que nos facilitan la búsqueda inicial de información entre los *feeds*. Seguidamente, el mapa nos muestra los centros médicos según tu localización actual, en caso de que el usuario acepte compartir ésta. En caso contrario, se muestra una ubicación predeterminada. Bajo el mapa está ubicado el enlace hacia el registro de clínicas. En último lugar, en el pie de página están los enlaces de interés, como la política de privacidad o el formulario de contacto de OperRating. Por descontado, la barra de navegación principal y el pie de página se encuentran disponibles en todas las pantallas de la aplicación. (Figura 43)

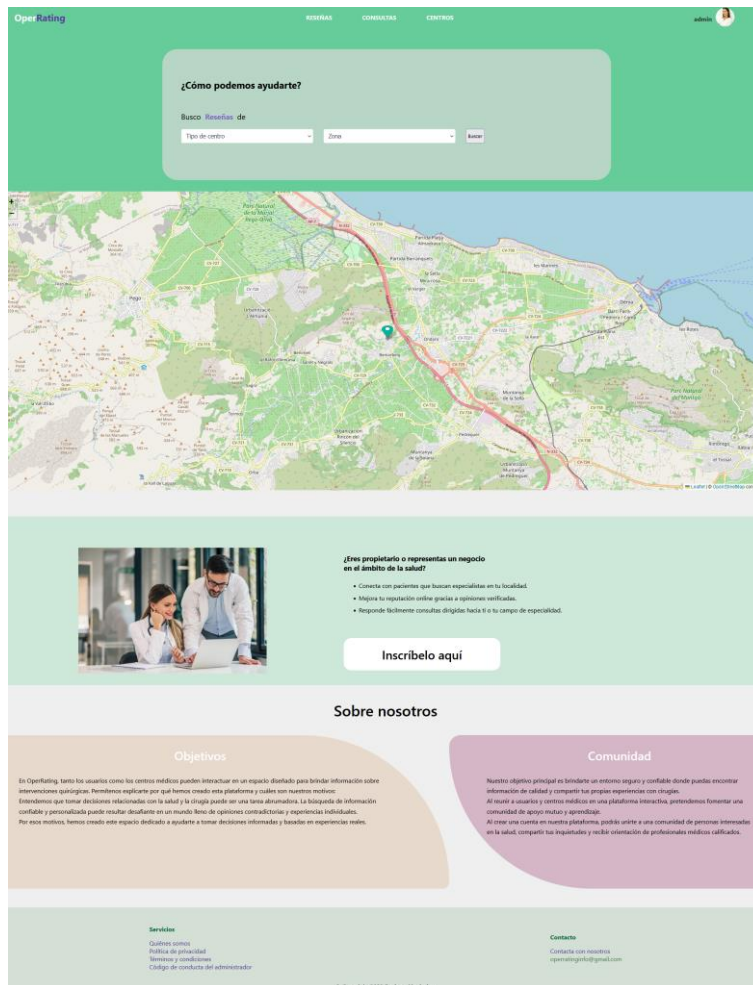


Figura 43: página de inicio.

Para poder crear cualquier clase de contenido, tanto publicaciones como comentarios o proclamarse dueño de un centro médico, es necesario el registro del usuario mediante el formulario mostrado en la figura 44, el cual nos pide un nombre de usuario, correo electrónico y una contraseña.

Figura 44: formulario de registro.

El enlace ubicado al pie del formulario nos envía a uno de los servicios ofrecidos en el pie de página, los términos y condiciones del servicio (figura 45)

Figura 45: términos y condiciones del servicio.

Volviendo al registro del usuario, al acabar de rellenar los datos, es enviado un correo de verificación al email escrito durante el registro, por lo que no es posible acceder a la cuenta hasta que se abra dicho email. Al pulsar sobre el enlace proporcionado por el email, se abre una nueva pantalla en nuestro navegador con el mensaje mostrado en la figura 46.



Figura 46: mensaje de verificaci3n de correo.

Una vez registrado e identificado el usuario, la barra de navegaci3n principal de la aplicaci3n muestra un contenido diferente, como se puede observar en la figura 47. Ahora, el usuario es capaz de crear publicaciones, acceder a su perfil personal y cambiar sus datos, o cerrar sesi3n desde el submen3 visible.

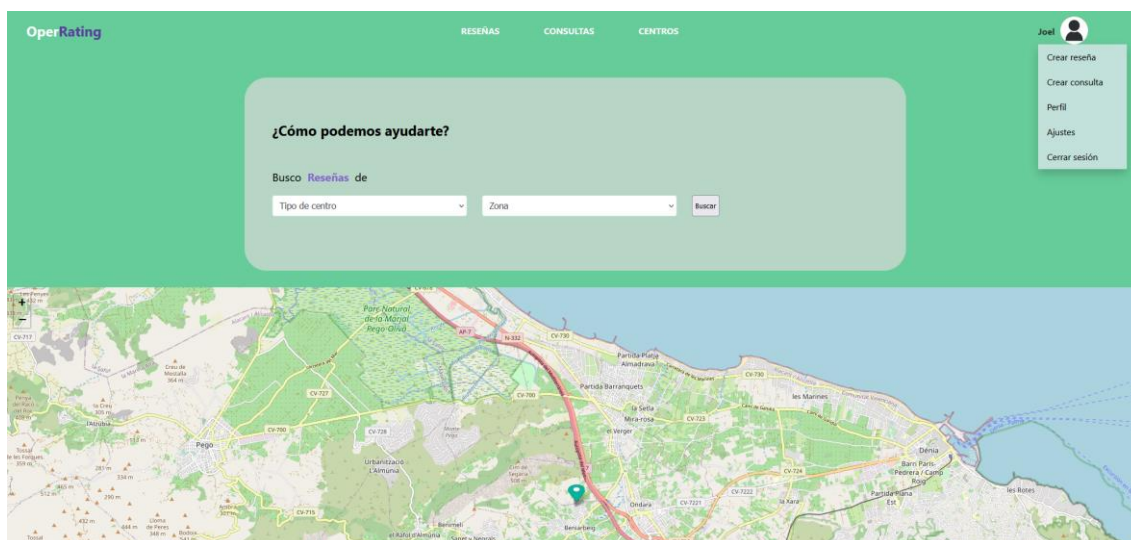


Figura 47: p3gina de inicio al haberse identificado.

Supongamos que el usuario desea publicar una reseña sobre una intervenci3n que se ha realizado recientemente. En ese caso, el usuario se dirige a la ventana de Crear reseña ubicada en el submen3 de la barra de navegaci3n y se encuentra con el formulario de la figura 48. Mediante el formulario, podemos indicar el t3tulo de nuestra reseña a publicar, el tipo de intervenci3n a la que nos hemos sometido, diversas valoraciones para aspectos para tener en cuenta, la cl3nica a la que hemos asistido, una descripci3n y varias im3genes, las cuales son opcionales.

OperRating

RESERVA CONSULTA CENTROS Joel

Cuentanos tu experiencia

Los campos con * son opcionales

Título

Mi reseña es sobre

Seleccione un tipo de intervención

Valoración general

☆☆☆☆

Valoración servicio

☆☆☆☆

Valoración resultados

☆☆☆☆

Clínica

Clinicas existentes [¿No encuentras la clínica que buscas?](#)

Descripción

Normal B I U

*Añade una imagen

Eliminar... No se ha seleccionado ningún archivo.

Crear post

Figura 48: formulario de publicación de una reseña.

En cuanto a la elección de la clínica, es posible que el centro al que haya ido el usuario no esté registrado en nuestra aplicación, por lo que éste puede añadirla él mismo al catálogo siguiendo el formulario mostrado en la figura 48 a través del enlace “¿No encuentras la clínica que busca?”, pero su información quedará pendiente de verificación (figura 49).

☆☆☆☆

Valoración servicio

☆☆☆☆

Valoración resulta

☆☆☆☆

Clínica

Clinicas existentes [¿No encuentras la clínica que buscas?](#)

Descripción

Normal B I U

Añádela al catálogo

Su información quedará en estado de supervisión, pero podrá realizar consultas o reseñas sobre ella

Cancelar Aceptar

Figura 49: validación de creación de una clínica.

Si, por el contrario, el usuario desea publicar una consulta, el proceso será similar al descrito anteriormente. En este formulario, sin embargo, el tipo de intervención y la clínica son opcionales. Si no es proporcionada una clínica, el usuario debe introducir una ubicación con tal de que su publicación sea clasificada más adelante según la provincia en la que se encuentra. (Figura 50)

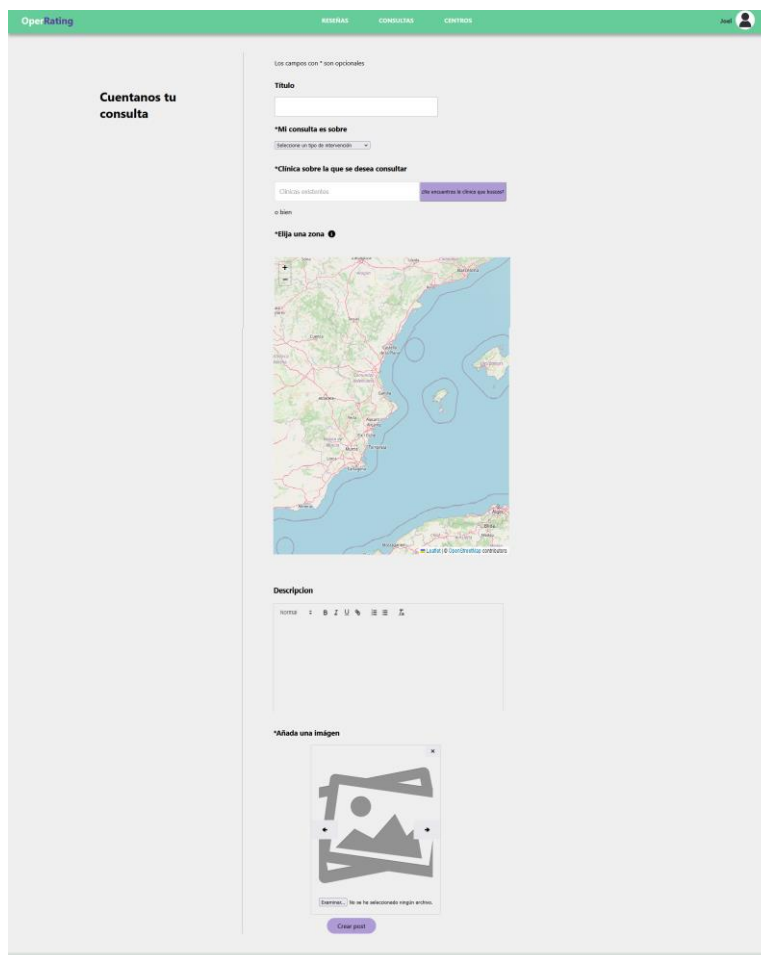
The image shows a web form titled "OperRating" for publishing a consultation. The form is structured as follows: 1. A header bar with "OperRating" on the left and navigation links "RESERVA", "CONSULTAS", and "CENTROS" on the right. 2. A sidebar on the left with the text "Cuentanos tu consulta". 3. A main content area with the heading "Los campos con * son opcionales". 4. A "Titulo" field. 5. A dropdown menu for "Mi consulta es sobre" with the option "Selecciona un tipo de intervención". 6. A field for "Clínica sobre la que se desea consultar" with a "No encuentro la clínica que busco" button. 7. A "o bien" separator. 8. A "Elige una zona" section with a map of the Canary Islands. 9. A "Descripción" section with a rich text editor. 10. An "Añade una imagen" section with a file upload icon and a "Cargar post" button.

Figura 50: formulario de publicación de una consulta.

Si el usuario registrado es propietario de un centro médico y desea inscribirlo en nuestra aplicación, deberá hacerlo mediante el formulario mostrado en la página inicial, comentado anteriormente. En este, nos piden información relevante respecto al centro a registrar, como su nombre, tipo de centro, como por ejemplo si se trata de un centro estético o un médico general, si tiene alguna especialidad, una descripción del centro y su ubicación (figura 51). En cuanto a la verificación de los datos de la clínica, se le pide al usuario una imagen acreditativa que será validada por un administrador, y la posición que ocupa el usuario en el centro. Para registrar la clínica en nuestra aplicación, el usuario también debe aceptar los términos y condiciones respecto al registro de centros, así como la política de privacidad de OperRating.

Figura 51: formulario de publicación de una clínica.

El propietario de una clínica puede añadir los datos sobre los profesionales que trabajen en su centro en cualquier momento. Esta funcionalidad es opcional y se ofrece en primer lugar a la hora de registrar el centro, tal y como se muestra en la figura 52.

Figura 52: opción de añadir personal a la clínica.

Si se accede a la opción *Añadir personal*, el usuario será redirigido a una pantalla en la cual podrá rellenar el formulario cuantas veces sea necesario para compartir la información de sus

profesionales (figura 53), ofreciendo su nombre, su especialidad y, opcionalmente, descripción e imagen.

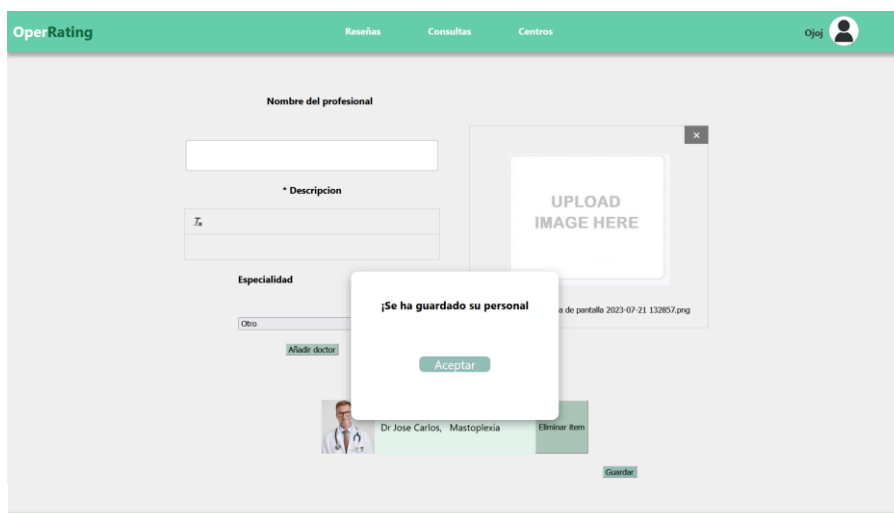


Figura 53: formulario de inscripción de personal.

Al finalizar el proceso de creación de la clínica y, opcionalmente, la adición de su personal, el usuario propietario será redirigido al perfil de la nueva clínica (figura 54). En él, se puede visualizar de forma sencilla la información del centro, la valoración que ha recibido en nuestra aplicación mediante las reseñas publicadas por los usuarios, con sus respectivas reseñas, las noticias publicadas por el centro mediante el formulario mostrado en la figura 55 y los profesionales registrados. En caso de ser el usuario identificado el propietario del centro, este tiene la posibilidad de modificar los datos o eliminarlos, tanto del centro como del personal, mediante formularios similares a los mostrados en las figuras 51 y 52. Si por el contrario el usuario identificado es un administrador, este puede verificar o invalidar los datos con tal de advertir al público sobre la veracidad de la información proporcionada.

Cabe mencionar también, que, si el centro de un propietario no reclamado existe, éste tiene la posibilidad de reclamarlo como suyo enviando los datos pertinentes al equipo de OperRating para su validación.

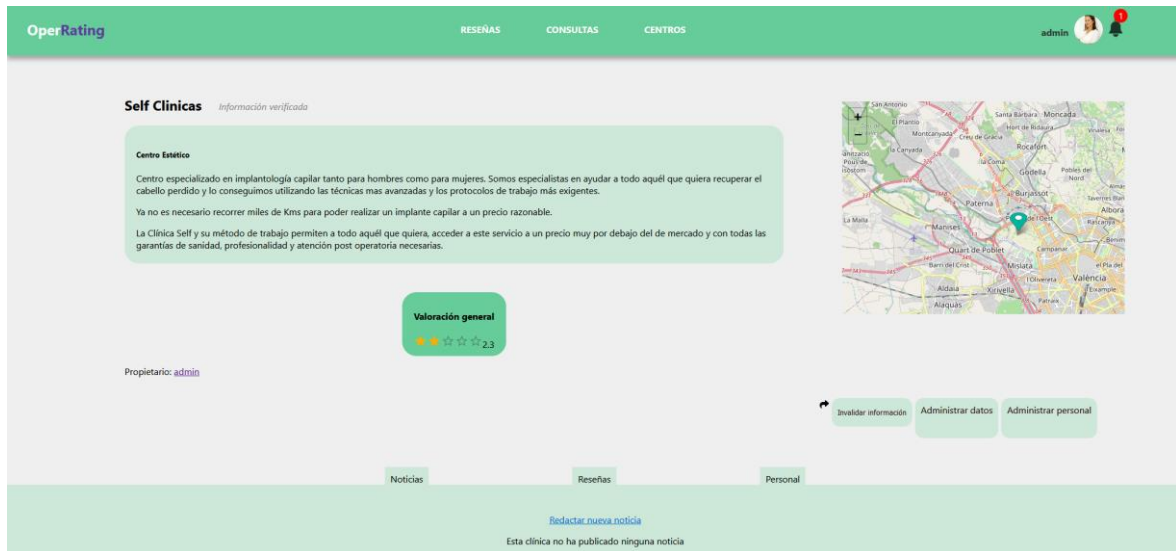


Figura 54: perfil de una clínica.

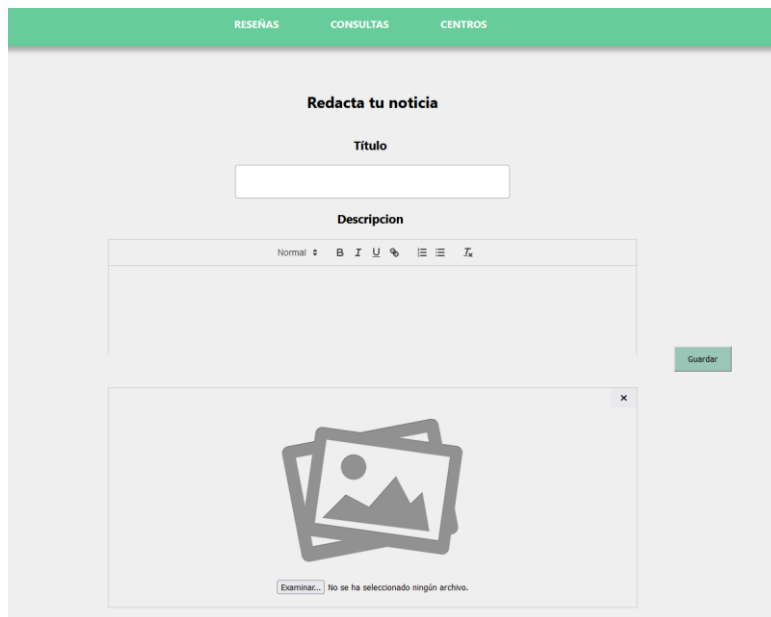


Figura 55: formulario de publicación de una noticia.

Retomando el hilo anterior de creación de publicaciones, al dirigirnos al *feed* del tipo de publicación que hemos publicado, lo encontraremos ubicado en la primera posición al acceder, si no existe alguna publicación más reciente. En la parte inferior a la barra de navegación principal encontramos más filtros para encontrar las publicaciones que nos resulten relevantes según el tipo de intervención, valoración media del centro o palabras clave y, al igual que en la página inicial, es posible buscar según el tipo de centro y la zona. En el mapa ubicado en el lado derecho se muestran las reseñas de forma más visual, agrupadas según la clínica en la que se hayan realizado. Desde el *feed* es posible compartir las publicaciones, según se puede observar en la figura 56. La información mostrada en un *Infinite Scroll*, lo cual hace que las publicaciones se vayan cargando según el usuario se desplace verticalmente por la aplicación.

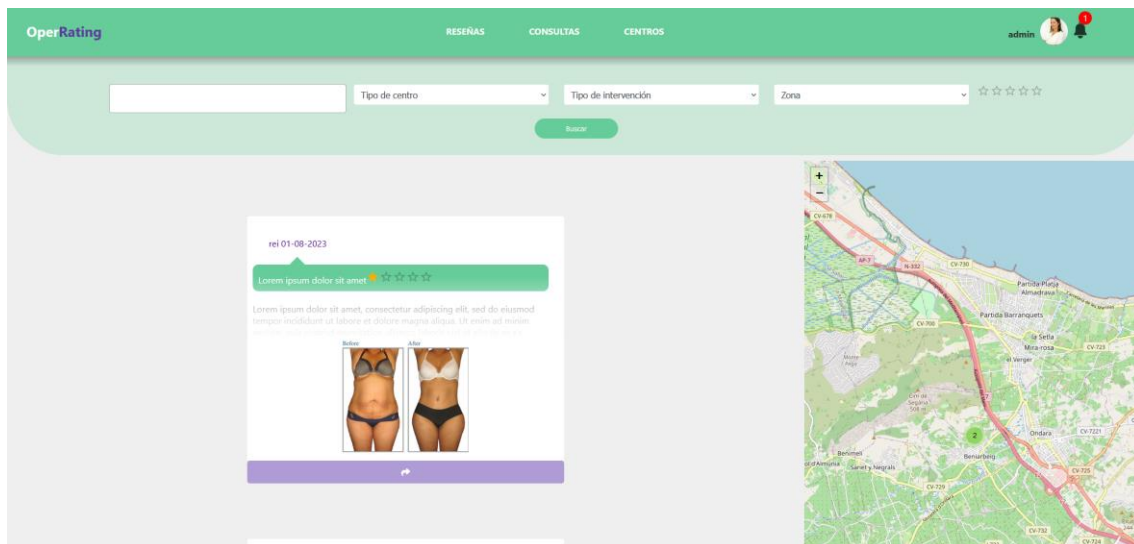


Figura 56: feed de publicaciones de tipo reseña.

Los feeds de centros y de consultas (figura 57) resultan idénticos exceptuando los filtros del último de estos, ya que las consultas no tienen valoración. También varía el color de los pines mostrados en el mapa, según el tipo de información que muestre.

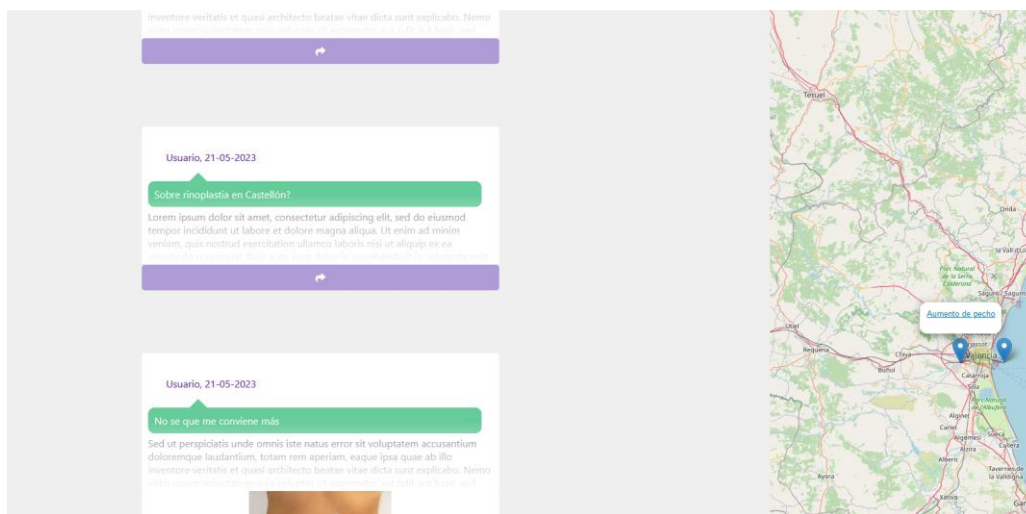


Figura 57: feed de publicaciones de tipo consulta.

Al pulsar sobre alguna de las publicaciones visibles, ya sea desde el feed o desde el mapa interactivo podremos acceder a la información detallada de cada publicación. Desde la pantalla de visualización de la información podemos ver la fecha de la publicación, ubicación, comentarios y los datos sobre el centro en cuestión. Al escribir un comentario el autor de la publicación recibirá una notificación, tanto por la aplicación (figura 58) como mediante su correo electrónico, las cuales contienen un enlace a la publicación en la cual se ha escrito el comentario.



Figura 58: notificaciones del usuario.

Es posible también añadir la publicación a favoritos mediante el icono mostrado al lado del icono de compartir publicación, o eliminarlo si el usuario identificado es el autor. En caso de que la publicación fuera una consulta, el autor tiene la posibilidad de cambiar el estado de la consulta de *Sin respuesta* a *Respondido* (figura 59).

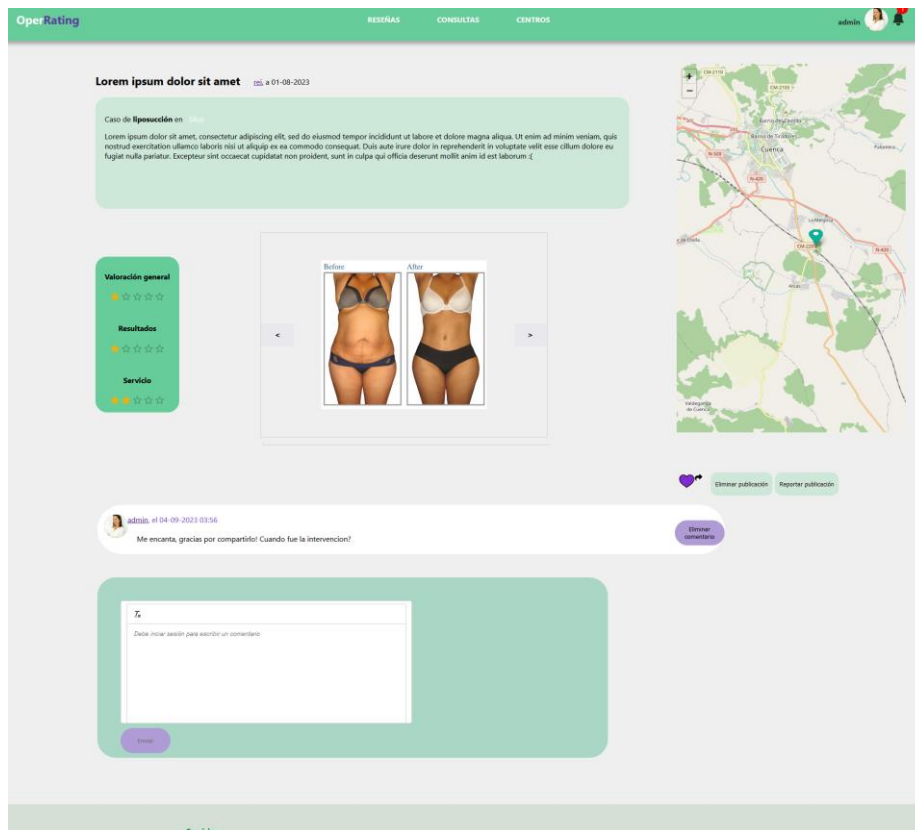


Figura 59: funcionalidades dentro de una publicación.

Si el contenido mostrado en la publicación viola alguna de las políticas de OperRating o supone un contenido no apto, disponemos de la posibilidad de reportar la publicación (figura 60). Esta acción envía un correo al equipo administrativo de OperRating con la información proporcionada. Por otra parte, un administrador es capaz de eliminar directamente la publicación si la considera ofensiva o no apropiada.

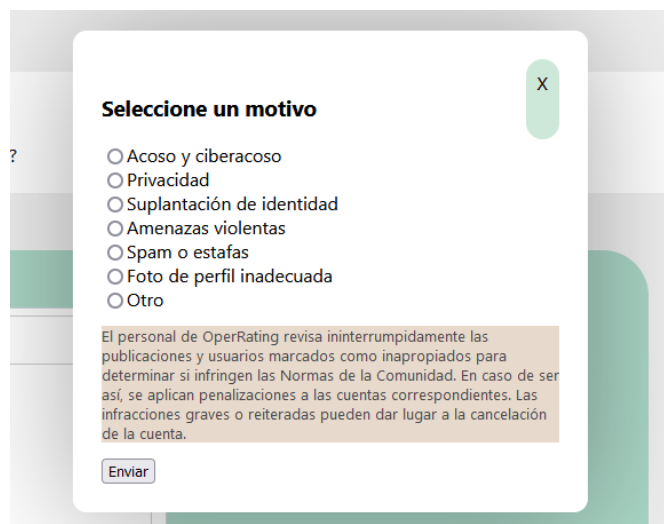


Figura 60: reportar contenido.

Al dirigirnos al perfil del usuario que ha creado la publicación previamente comentada, podemos encontrar las publicaciones y comentarios del usuario seleccionado, así como varias opciones: iniciar chat, reportar usuario o bloquearlo. La segunda opción nos muestra el mismo formulario que el comentado en la figura 60, teniendo su misma funcionalidad. La opción de bloquear usuario, por otro lado, impide que este pueda comentar en las publicaciones del usuario identificado, o enviarle mensajes privados. (figura 61)

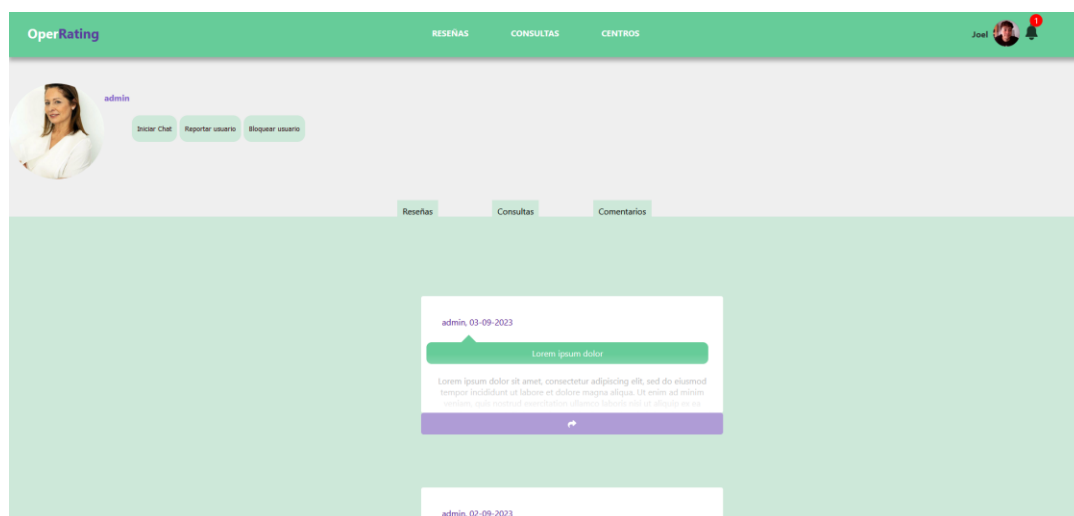


Figura 61: vista del perfil de un usuario.

Al seleccionar la primera opción, se creará una sala de chat entre el usuario identificado y el seleccionado, según se advierte en la figura 62. Cada mensaje genera una notificación similar a la descrita en la funcionalidad los de comentarios en publicaciones. El enlace que contienen las notificaciones por mensajes privados lleva, sin embargo, al perfil del usuario que ha enviado el mensaje al usuario identificado.

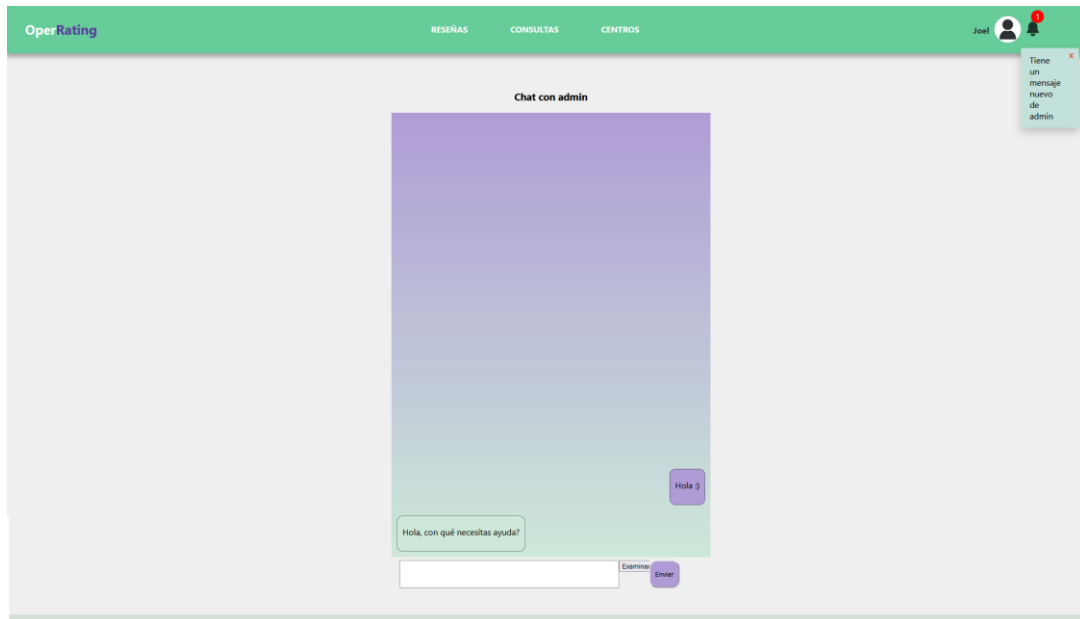


Figura 62: chat privado con otro usuario.

Si nos dirigimos al perfil del usuario identificado, contamos con diversas opciones respecto a la personalización de la cuenta personal, siendo posible modificar los datos del usuario o la imagen de perfil. Al igual que al visitar un perfil externo al personal, podemos ver las publicaciones y comentarios del usuario. Sin embargo, al ser el usuario identificado el que ha accedido a ver la página, es posible ver también las publicaciones que ha marcado como guardadas, cambiar su imagen de perfil y los datos de este (figura 63).



Figura 63: vista del perfil personal.

Una vez accedido a los ajustes del perfil del usuario, la aplicación nos ofrece varias opciones, según la figura 64, como modificar nuestro nombre de usuario, email o eliminar la cuenta. Eliminar la cuenta conlleva borrar todo el contenido relacionado con ella, como podrían ser sus publicaciones, comentarios y clínicas registradas.

Figura 64: ventana de datos personales.

En último lugar, el contenido ubicado en el lado izquierdo del pie de página incluye los datos relacionados con la información sobre la página, como, por ejemplo, la política de privacidad, el código de conducta del administrador y una explicación de la motivación de la aplicación (figura 65).

Figura 65: página informativa sobre OperRating.

En el lado opuesto a los Servicios ofrecidos en el pie de página, encontramos los datos de contacto, junto con un enlace que nos lleva al formulario de contacto (figura 66). Para poder hacer uso del formulario no es necesario estar registrado en la aplicación, por lo que se debe introducir el correo electrónico. Al seleccionar un motivo de solicitud (figura 67), se muestran los demás campos que deben ser rellenados por el usuario, los cuales varían según el motivo seleccionado. Una vez aceptado el *checkbox* y enviada la solicitud, es enviado un correo al equipo de OperRating con los datos proporcionados por el usuario.

OperRating Reseñas

Enviar una solicitud

¿Con qué necesitas ayuda?

-

- Necesito ayuda con mi cuenta
- Reportar bug
- Reportar infracción de la política de contenidos
- Consultas relacionadas con derechos de privacidad
- Reportar mala conducta por parte de un administrador
- Otro tipo de ayuda

Servicios

- Quiénes somos
- Política de privacidad
- Términos y condiciones
- Código de conducta del administrador

Figura 66: opciones del formulario de contacto.

OperRating Reseñas Consultas Centros

Enviar una solicitud

¿Con qué necesitas ayuda?

Reportar mala conducta por parte de un administrador

Dirección de email

chococoolu@gmail.com

¿Quiénes reportar?

-

Asunto de la consulta

Detalles de la consulta

Escribo aquí un comentario

He leído el Código de Conducta del Administrador y he compartido toda la información necesaria para este reporte. Tus reportes nos ayudan a identificar patrones constantes e impugnables de infracción de las pautas.

Enviar

Figura 67: datos del formulario de contacto.