

Aplicación buscadora de bancos de sangre y donación de sangre

Memoria Trabajo Fin de Grado Ingeniería Informática

Septiembre 2022



**VNiVERSIDAD
D SALAMANCA**

Autor:

Pedro Luis Alonso Díez

Tutores:

Luis Augusto Silva

Gabriel Villarrubia González

Juan Francisco de Paz Santana

Certificado de los tutores

D. Luis Augusto Silva, D. Gabriel Villarrubia González, y D. Juan Francisco de Paz Santana, profesores del Departamento de Informática y Automática de la Universidad de Salamanca

CERTIFICAN:

Que el trabajo titulado “Aplicación buscadora de bancos de sangre y donación de sangre” ha sido realizado por D. Pedro Luis Alonso Díez, con DNI 72190545-P, para la asignatura “Trabajo de Fin de Grado” de la titulación “Grado en Ingeniería Informática de la Universidad de Salamanca”.

Y para que así conste a todos los efectos oportunos.

En Salamanca, a 7 de Septiembre de 2022

SILVA LUIS
AUGUSTO -
Y8119893X

Assinado de forma digital por SILVA LUIS AUGUSTO - Y8119893X
Datos: 2022.09.07 08:31:41 +02'00'

D. Luis Augusto Silva

VILLARRUBIA
GONZALEZ
GABRIEL -
71932845T

Firmado digitalmente por VILLARRUBIA GONZALEZ GABRIEL - 71932845T
Nombre de reconocimiento (DN): c=ES, serialNumber=IDCES-71932845T, givenName=GABRIEL, sn=VILLARRUBIA GONZALEZ GABRIEL - 71932845T
Fecha: 2022.09.07 10:12:56 +02'00'

D. Gabriel Villarrubia González

Juan
Francisco De
Paz Santana

Firmado digitalmente por Juan Francisco De Paz Santana
Nombre de reconocimiento (DN): cn=Juan Francisco De Paz Santana, o=Universidad de Salamanca, ou, email=fcofds@usal.es, c=ES
Fecha: 2022.09.07 10:13:17 +02'00'

D. Juan Francisco de Paz Santana

Resumen

Vivimos en un mundo cada vez más globalizado, dónde viajar cada vez es más fácil y la tecnología nos permite estar siempre informados. Esto provoca que muchos ciudadanos estén periodos de tiempo fuera de su residencia habitual, por eso muchos donantes de sangre reciben las alertas del Servicio Nacional de Salud tarde o en hospitales muy lejanos a su ubicación actual.

En la actualidad, existen aplicaciones orientadas al usuario, dónde se puede registrar cómo donante de sangre y ver los puntos de donación cercanos, sin embargo, la mayoría de estas aplicaciones son gestionadas por Comunidades Autónomas, esto causa un uso ineficiente del sistema de donaciones debido a que, en caso de urgencia, los mensajes de necesidad de sangre no llegan a todas los donantes disponibles en la zona.

Este fue el punto de partida de este proyecto, ofrecer una plataforma que pone en contacto al Servicio Nacional de Salud y los donantes de sangre, para que, en caso de urgencia, se puedan emitir alertas para solicitar la donación en los hospitales en los que sea necesario, permitiendo al usuario de forma simple, cambiar su ubicación cuando viaje.

A su vez, permite al Servicio Nacional de Salud tener constancia del número de donantes de sangre en cada provincia en tiempo real, así como su tipo de sangre, permitiendo incluso comunicarse con el usuario a través de un chat interno. De esta forma se incrementa la eficiencia en la emisión de alertas, maximizando el número de donantes posible que puedan acudir a donar cuando sea necesario.

Palabras clave: Hospital, Donación, Sangre, Urgencia.

Abstract

We live in an increasingly globalized world, where travel is becoming easier and easier and technology allows us to be always informed. This causes many citizens to spend periods of time away from their usual residence, so many blood donors receive alerts from the National Health Service late or in hospitals far away from their current location.

Currently, there are user-oriented applications, where you can register as a blood donor and see nearby donation points, however, most of these applications are managed by Autonomous Communities, this causes an inefficient use of the donation system because, in case of emergency, the messages of blood need do not reach all available donors in the area.

This was the starting point of this project, to offer a platform that puts the National Health Service and blood donors in contact, so that, in case of emergency, alerts can be issued to request donation in the hospitals where it is necessary, allowing the user to simply change their location when traveling.

In turn, it allows the National Health Service to keep track of the number of blood donors in each province in real time, as well as their blood type, even allowing communication with the user through an internal chat. This increases the efficiency in the issuing of alerts, maximizing the number of donors who can come to donate when needed.

Keywords: Hospital, Donation, Blood, Emergency.

Tabla de contenido

1. Introducción.....	1
2. Objetivos del proyecto.....	3
2.1. Objetivos del sistema.....	3
2.2. Objetivos personales.....	3
3. Técnicas y herramientas.....	4
Javascript.....	4
NodeJS.....	4
NPM.....	4
Visual Studio Code.....	4
React.....	4
HTML y CSS.....	5
Bootstrap.....	5
Firebase.....	5
Github.....	5
Leaflet.....	5
GeoJSON.....	5
Colors y Paletton.....	5
Adobe XD.....	6
FontAwesome.....	6
Diagrams.net.....	6
EZEstimate.....	6
Trello.....	6
JSDoc.....	6
4. Aspectos relevantes del desarrollo.....	7
4.1. Metodología.....	7
4.2. Estimación de esfuerzo.....	8
4.3. Planificación temporal.....	9
4.4. Especificación de requisitos.....	11
4.5. Análisis de requisitos.....	16
4.5.1. Modelo de dominio.....	16
4.5.2. Realización de casos de uso de análisis.....	17
4.6. Diseño del software.....	18
4.6.1. Patrones arquitectónicos.....	18
4.6.2. Subsistemas de Diseño.....	19

4.6.3. Clases de diseño.....	20
4.6.4. Vista arquitectónica de diseño.....	22
4.6.5. Realización de casos de uso de diseño.....	23
4.6.6. Modelo de despliegue.....	24
4.6.7. Prototipo digital.....	24
4.7. Implementación.....	25
4.7.1. Implementación del código.....	25
4.7.2. Implementación de las interfaces.....	26
4.8. Pruebas.....	29
4.9. Funcionalidad del sistema.....	29
Registro.....	29
Inicio de sesión.....	30
Recuperar contraseña.....	30
Funcionalidades del chat.....	32
Perfil.....	33
Modificar información.....	33
Modificar contraseña y eliminar cuenta.....	34
Ver hospitales en el mapa.....	34
Ver página de control.....	34
Ver donantes por provincias.....	35
Enviar alerta de solicitud de sangre.....	35
5. Conclusiones y líneas de trabajo futuras.....	36
Conclusiones.....	36
Líneas futuras de trabajo.....	37
6. Bibliografía.....	38

Índice de figuras

Figura 1: Proceso unificado	8
Figura 2: Estimación de esfuerzo	9
Figura 3: Planificación temporal – Diagrama de Gantt	10
Figura 4: Diagrama de actores	12
Figura 5: Requisitos funcionales - Paquete Gestión de usuarios	15
Figura 6: Requisitos funcionales - Paquete Gestión Administrador.....	15
Figura 7: Modelo de dominio	17
Figura 8: Ejemplo de diagrama de secuencia – Análisis.....	18
Figura 9: Patrón arquitectónico MVVM	19
Figura 10: Patrón arquitectónico Presentation and Container components.....	19
Figura 11: Diagrama subsistemas de diseño	20
Figura 12: Clases de diseño Modelo.....	20
Figura 13: Clases de diseño Vista	21
Figura 14: Clases de diseño ViewModel.....	22
Figura 15: Vista arquitectónica	23
Figura 16: Ejemplo Diagrama de secuencia – Diseño	23
Figura 17: Modelo de despliegue.....	24
Figura 18: Ejemplo Prototipo digital - Perfil.....	25
Figura 19: Ejemplo de documentación – Componente.....	26
Figura 20: Interfaz del login.....	26
Figura 21: Interfaz responsive - 1.....	27
Figura 22: Interfaz responsive - 2.....	27
Figura 23: Interfaz responsive - 3.....	28
Figura 24: Registro	29
Figura 25: Inicio de sesión.....	30
Figura 26: Recuperar contraseña.....	31
Figura 27: Correo de recuperación de contraseña.....	32
Figura 28: Chat.....	32
Figura 29: Perfil	33
Figura 30: Modificar información.....	33
Figura 31: Modificar contraseña y eliminar cuenta	34
Figura 32: Ver hospitales en el mapa	34

Figura 33: Ver hospitales en el mapa - Administrador.....	35
Figura 34: Ver donantes por provincia - Administrador	35
Figura 35: Enviar alerta de solicitud de sangre	36
Figura 36: Mensaje de alerta enviado.....	36

Índice de tablas.

Tabla 1: Planificación temporal – Fechas estimadas.....	10
Tabla 2: Ejemplo de participante	11
Tabla 3: Ejemplo de Requisito de información	13
Tabla 4: Ejemplo de Requisito funcional.....	14
Tabla 5: Ejemplo de Requisito no funcional.....	16

1. Introducción.

Las donaciones de sangre constituyen una parte importante del sistema de salud, debido a que la sangre es indispensable para vida, y, por el momento, no existe un sustituto artificial, por eso se necesitan donaciones regulares y voluntarias.

Un donante sano puede donar sangre cada dos meses, debido a que es el tiempo en el que el donante puede reestablecer la cantidad de sangre donada, además tiene varios beneficios para la salud, como, por ejemplo:

- Equilibrar los niveles de hierro en la sangre.
- Reduce el riesgo de padecer infartos.
- Ofrece información de nuestra salud actual.
- Estimula la producción de células sanguíneas.

Los hospitales realizan transfusiones de sangre a diario, y la ayuda de los donantes es indispensable para asegurar la disponibilidad de sangre en el lugar dónde se precise, ya que la sangre solo puede almacenarse durante un tiempo limitado y luego deja de ser usable.

Algunas de los casos más frecuentes son:

- Personas con traumatismos graves.
- Enfermos de cáncer.
- Intervenciones quirúrgicas.
- Insuficiencia de médula ósea.

Las aplicaciones online para donar sangre suelen estar focalizadas en que el usuario solicite cita previa para acudir a donar a los bancos de sangre, sin embargo, en caso de urgencia pueden llegar a ser ineficientes, debido a que suelen estar administradas por los Servicios de Salud Autonómicos y las alertas de solicitud de donación pueden llegar a personas que no se encuentren en su lugar de residencia actual.

Por esta razón, se ha optado por desarrollar una aplicación que sirva como sistema central en casos de necesidad de cualquier tipo de sangre urgente en determinados hospitales. Permitiendo al usuario modificar su ubicación de una forma sencilla, en caso de que no se encuentre en su domicilio habitual. De esta forma se permitirá lanzar alertas personalizadas dependiendo de lo que se necesite en el momento, llegando al máximo número de donantes posibles.

También la plataforma permite al donante establecer un punto de comunicación online con el Servicio de salud a través de un chat interno en la aplicación, para posibles dudas concretas.

Con todas estas funcionalidades se permitirá al Servicio que lo gestione, disponer del número de donantes en cada provincia en tiempo real, permitiendo de esta forma que

el sistema de donaciones funcione de una manera eficiente y maximizando el objetivo de las donaciones.

A través del presente documento se mostrará el proceso de desarrollo de la aplicación, siguiendo la guía ubicada en la plataforma Diaweb [1]. Las fases que se detallarán son las siguientes.

- **Objetivos del proyecto:** Se desarrollarán y explicarán los objetivos que plantea alcanzar el proyecto.
- **Técnicas y herramientas:** Se detallarán las herramientas utilizadas para la realización del proyecto, así como las metodologías usadas.
- **Aspectos relevantes:** Se detallarán los aspectos más relevantes del proyecto que hayan surgido durante su desarrollo.
- **Conclusiones y líneas de trabajo futuro:** Se detallarán las conclusiones del proyecto y sus posibles actualizaciones futuras.

Los contenidos de esta memoria se complementarán con la realización de los siguientes anexos:

- Anexo I - Plan de proyecto software: En este anexo se detallan los costes del proyecto y su planificación temporal, con el objetivo de analizar su viabilidad.
- Anexo II - Especificación de requisitos software: En este anexo se detallan los requisitos software necesarios para el desarrollo de la aplicación.
- Anexo III - Análisis de requisitos: En este anexo se documentan los procesos de análisis y refinamiento de cada uno de los requisitos software.
- Anexo IV – Diseño del sistema software: En este anexo se detalla el modelo del sistema mediante la realización de casos de uso.
- Anexo V – Documentación técnica del programador: En este anexo se recoge toda la documentación relativa al código fuente de la aplicación y su estructura.
- Anexo VI – Manual de usuario: En este anexo se detalla de forma sencilla todas las funcionalidades que posee el sistema y su modo de uso, centrado en el usuario final.

2. Objetivos del proyecto.

La finalidad de este apartado es detallar todos los objetivos que el proyecto plantea. Se explicarán los objetivos del sistema y los objetivos personales.

2.1. Objetivos del sistema.

El objetivo principal del sistema es ofrecer una plataforma para gestionar las solicitudes de donación de sangre en casos de emergencia, dónde se detalle el tipo de sangre y lugar dónde se necesita y maximizando el número de receptores de la alerta compatibles.

Algunos de sus objetivos complementarios son:

- **Mostrar los hospitales cercanos:** El sistema debe mostrar a los usuarios sus hospitales más cercanos.
- **Comunicación bidireccional con el servicio de salud:** El sistema debe permitir al usuario y al servicio de salud gestor poder comunicarse a través de un chat interno en la aplicación.
- **Envío de alertas personalizadas:** El sistema debe permitir al Servicio de salud gestor emitir alertas solicitando la donación del tipo de sangre que necesiten en un determinado hospital y esta alerta debe llegar a todos los donantes registrados dentro del alcance especificado.
- **Disponer de la información actualizada de todos los hospitales de España:** El sistema debe ofrecer los datos de todos los hospitales españoles, y además mostrarlos en el mapa.
- **Gestión de usuarios:** El sistema debe permitir a los usuarios modificar su información y además debe mostrar determinada información, como, por ejemplo: nombre, tipo de sangre y ubicación, al servicio de salud gestor.

2.2. Objetivos personales.

El objetivo de este proyecto a nivel personal es el aprendizaje de nuevas tecnologías y metodologías para la realización de un proyecto desde cero, ser capaz de con solo un día construir algo funcional y escalable con el tiempo. Todos los conocimientos adquiridos en la realización del proyecto serán de gran utilidad en la vida laboral.

Además, también se busca ofrecer un servicio que ayude a la sociedad a mejorar en un ámbito tan importante y necesario como la salud. Esto provoca un incremento de la motivación a la hora del desarrollo del proyecto, buscando como objetivo final el uso de la aplicación en un escenario real.

Todo esto junto supone que el desarrollo de este proyecto suponga un gran reto personal en conocimientos técnicos, organización e investigación, que, con su superación aumentará la confianza personal de cara al mercado laboral.

3. Técnicas y herramientas.

Se detallan las herramientas de desarrollo y técnicas metodológicas usadas en el proyecto.

Javascript.

Javascript [2] es un lenguaje de programación orientado a objetos, débilmente tipado y dinámico. Su uso general es en el lado del cliente, sin embargo, se puede usar también en el lado del servidor. Se optó por este lenguaje de programación debido a que es multiplataforma y muy sencillo. Se trata del lenguaje más usado en el desarrollo de aplicaciones web.

NodeJS

NodeJS [3] es un entorno de tiempo de ejecución para la creación de aplicaciones de red en tiempo real. Se trata de un entorno JavaScript, pero del lado del servidor, basado en eventos. Se escogió este entorno debido a que permite utilizar el mismo lenguaje de programación tanto en el servidor como en el cliente.

NPM.

Node Package Module [4] es un gestor de paquetes desarrollado en JavaScript, que permite la descarga de multitud de librerías de una manera muy sencilla. Esto permite agregar y administrar módulos necesarios en el desarrollo del proyecto.

Visual Studio Code.

Visual Studio Code [5] es un IDE desarrollado por Microsoft, de software libre y multiplataforma. Posee multitud de extensiones que facilitan el desarrollo de la aplicación.

React.

React [6] es una librería escrita en JavaScript que permite desarrollar aplicaciones multiplataforma y ofrece una sencilla curva de aprendizaje. También permite al desarrollador realizar cambios en el código y ver su efecto producido en la aplicación en tiempo real.

HTML y CSS.

Se trata de lenguajes de marcado, usados para realizar las interfaces. Con HTML se crean los objetos disponibles en la página web, y con CSS se controla el estilo y el diseño que tendrá la página web.

Bootstrap.

Bootstrap es una biblioteca multiplataforma de código abierto usada para la creación de interfaces web adaptativas.

Firebase.

Para la base de datos se ha utilizado el servicio de Google Firebase, se trata de una base de datos NoSQL alojada en la nube. Se escogió este servicio debido a que tiene un plan gratuito, y permite a los datos sincronizarse en tiempo real [7].

Github.

Github es una plataforma que permite a los desarrolladores subir su código y establecer un control de versiones haciendo uso de Git.

Leaflet.

Para los mapas se ha utilizado la biblioteca Leaflet debido a que dispone de una interfaz amigable para dispositivos móviles, es open source, y ofrece un gran grado de personalización [8].

GeoJSON.

Para representar los hospitales en los mapas se ha usado archivos geojson, ya que implementan las coordenadas de los hospitales y se pueden implementar en los mapas de Leaflet. Los datos se han sacado de la página de datos del gobierno.

Colours y Paletton.

Se utilizó Colours [9] para la generación de paletas de colores y posteriormente Paletton [10] para generar distintas tonalidades de la paleta de colores escogida.

Adobe XD

Adobe XD [11] es una herramienta de diseño enfocada en crear interfaces de páginas web y aplicaciones. Se escogió esta aplicación para realizar el prototipado digital debido a que su uso es muy intuitivo.

FontAwesome

Se ha usado la página web FontAwesome [12] para la inserción de los iconos disponibles en la aplicación.

Diagrams.net

Diagrams.net [13] es una herramienta que te permite crear gran variedad de diagramas desde la web, además es gratuita. Se ha utilizado para la creación de los diagramas de secuencia, actores, modelo de dominio, etc.

EZEstimate.

Se trata de una herramienta dedicada a calcular la estimación de esfuerzo del proyecto a desarrollar.

Trello.

Trello es un software de administración de proyectos [14]. Se ha usado para la organización de las distintas tareas que se han ido realizando a lo largo del proyecto.

JSDoc.

JSDoc [15] es una sintaxis para agregar documentación de la API al código fuente de JavaScript. Genera la documentación del código fuente en archivos HTML.

4. Aspectos relevantes del desarrollo.

Se presentan las partes más importantes de las distintas fases del desarrollo del proyecto.

4.1. Metodología.

Se ha utilizado el modelo de desarrollo del Proceso Unificado [16] para la realización del proyecto. Se trata de la metodología más usada en el desarrollo de sistemas orientados a objetos. Esto se debe a que ofrece una alta capacidad de adaptación a distintos sistemas software, algunas de las características que ofrece son [17]:

- Está centrado en la arquitectura.
- Está basado en componentes.
- Utiliza Lenguaje de Modelado Unificado [18] para las representaciones visuales.
- Es iterativo e incremental.

El proceso unificado se repite a lo largo de ciclos de desarrollo denominados iteraciones. Cada iteración termina con un punto de control para observar el progreso del proyecto.

Las fases que componen una iteración son:

- **Inicio:** Se estima el alcance que tendrá el proyecto.
- **Elaboración:** Se planifica y detalla la arquitectura y el diseño del sistema.
- **Construcción:** Se construye la aplicación.
- **Transición:** Fase para incorporar mejoras y corregir errores.

Las iteraciones engloban tareas que se pueden clasificar en:

- Gestión de proyecto
- Entorno de desarrollo
- Modelo de negocio
- Requisitos
- Análisis y diseño
- Implementación
- Pruebas
- Despliegue
- Configuración y gestión de cambios

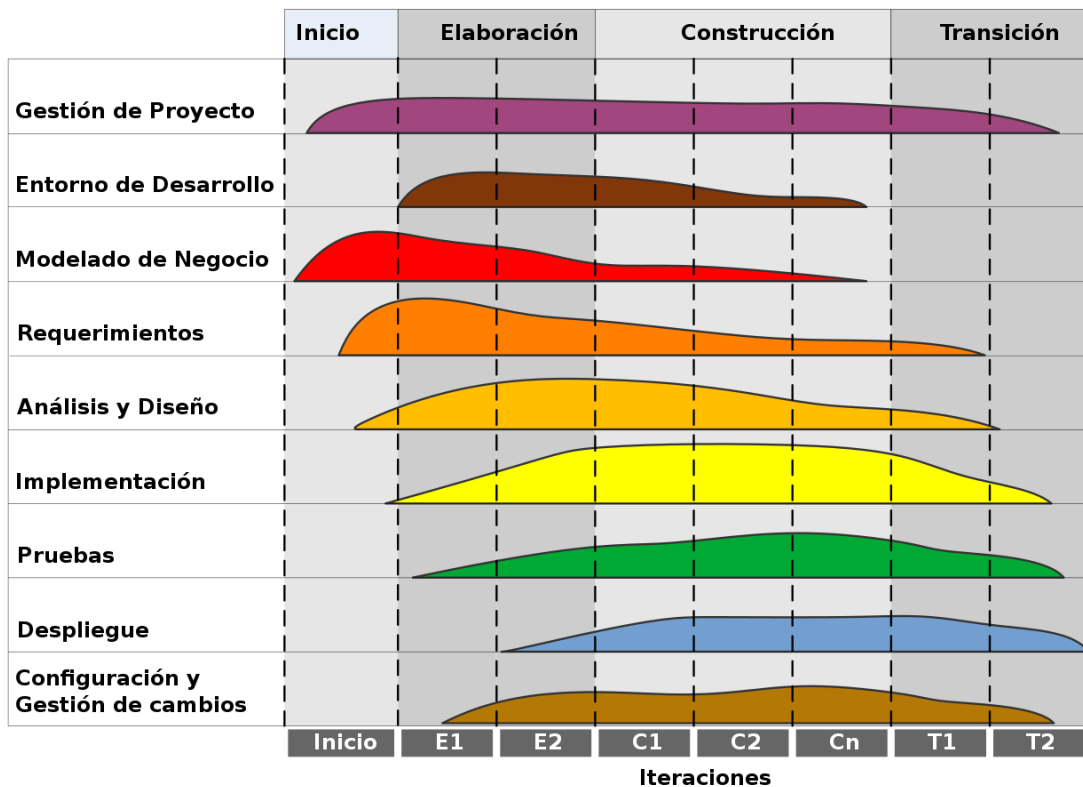


Figura 1: Proceso unificado

4.2. Estimación de esfuerzo.

Una vez especificados los requisitos que tendrá el sistema podemos calcular la estimación de esfuerzo. Para realizar este cálculo se usará la métrica UCP (Puntos de Caso de Uso) junto con la herramienta EZEstimate.

The screenshot shows the EZEstimate application window. It is divided into several sections:

- Module:** A dropdown menu is set to 'CU'. Below it are 'Add Module' and 'Delete' buttons.
- Summary:** Displays 'Total Modules' as 2. An 'Excel Report' button is present. Below are input fields for 'Use cases' (Simple: 6, Average: 9, Complex: 5) and 'Actors' (Simple: 0, Average: 0, Complex: 3).
- Add Actor / Use case:** A form with 'Actor / Use case Name', 'Select Type' (dropdown set to 'Usecase'), and 'Complexity' (dropdown set to 'Average') fields, followed by an 'Add' button.
- Tech / Env Factors:** Two buttons labeled 'Set Tech Factor' and 'Set Env Factors'.
- Estimation Summary:** A table of calculated values:

UAW	9
UUCW	195
UUPC = UAW + UUCW	204
TFactor	48
EFactor	21
TCF = 0.6 + (.01*TFactor)	1,08
EF = 1.4 + (-0.03*EFactor)	0,77
UCP = UUCP*TCT*EF	169,6464
Total Effort@ 8 Hrs/UCP	1357,1712
- Use case / Actor List:** A table listing 20 use cases:

Id	Module	Type	Name	complexity
7	CU	Usecase	CU-004	Complex
8	CU	Usecase	CU-005	Complex
9	CU	Usecase	CU-006	Simple
10	CU	Usecase	CU-007	Complex
11	CU	Usecase	CU-008	Average
12	CU	Usecase	CU-009	Simple
13	CU	Usecase	CU-010	Average
14	CU	Usecase	CU-011	Simple
15	CU	Usecase	CU-012	Average
16	CU	Usecase	CU-013	Complex
17	CU	Usecase	CU-015	Average
18	CU	Usecase	CU-016	Average
19	CU	Usecase	CU-014	Average
20	CU	Usecase	CU-017	Simple
21	CU	Usecase	CU-018	Simple
22	CU	Usecase	CU-019	Average
23	CU	Usecase	CU-020	Average

Figura 2: Estimación de esfuerzo

Mediante esta herramienta podemos estimar que la aplicación nos llevará unas 1357 horas, es decir, unas 24 semanas.

Toda esta información se encuentra más detallada en el “Anexo I: Plan de Proyecto Software”

4.3. Planificación temporal.

En este apartado definiremos y organizaremos las tareas a realizar en base a los ciclos de desarrollo del proceso unificado. Para detallar la planificación temporal que seguiremos se usará el diagrama de Gantt definido en la Figura 2.

DIAGRAMA DE GANTT

ESTRUCTURA DE	TÍTULO DE LA TAREA	SEMANAS
1	Fase 1 - Inicio	5
1,1	Análisis de lenguajes a usar y herramientas	1
1,2	Planificación temporal	1
1,3	Especificación de requisitos software	1
1,4	Diseño de BBDD	1
1,5	Prototipo digital	1
2	Fase 2 - Elaboración	4
2,1	Inicio de la documentación	1
2,2	Diagramas ed casos de uso	1
2,3	Refinamiento del prototipo digital	1
2,4	Configuración del framework	1
3	Fase 3 - Construcción	7
3,2	Implementación gestion de usuarios	1
3,3	Implementación gestion de hospitales	1
3,4	Implementación gestion de alertas	1
3,5	Implementación interfaz responsive	1
3,6	Implementación pequeñas funcionalidades adicionales	1
3,1	Continuación de la documentación	1
3,7	Pruebas de funcionalidad	1
4	Fase 4 - Transición	8
4,1	Revisión final requisitos	1
4,2	Revisión final diagramas	1
4,3	Revisión final funcionalidad de la aplicación	1
4,4	Arreglo de bugs	1
4,5	Elaboración final de la documentación	4

Figura 3: Planificación temporal – Diagrama de Gantt

Siguiendo esta planificación podemos estimar las fechas en las que elaboraremos las distintas fases del proceso unificado.

Fase	Comienzo	Fin
Inicio	14 de marzo de 2022	17 de abril de 2022
Elaboración	18 de abril de 2022	15 de mayo de 2022
Construcción	16 de mayo de 2022	3 de julio de 2022
Transición	4 de julio de 2022	28 de agosto de 2022

Tabla 1: Planificación temporal – Fechas estimadas

Toda esta información se encuentra más detallada en el “Anexo I: Plan de Proyecto Software”

4.4. Especificación de requisitos.

Para la especificación de requisitos se va a utilizar el Lenguaje Unificado de Modelado (UML) con la metodología Durán y Bernárdez [19] y la herramienta Diagrams.net.

Para empezar, debemos definir los participantes involucrados en el proyecto, incluyendo tanto al desarrollador como a los tutores académicos. Se incluye un ejemplo en la *Tabla 2*

Participante	Pedro Luis Alonso Díez
Organización	Universidad de Salamanca
Rol	Autor
Tipo de usuario	Desarrollador
Comentarios	No

Tabla 2: Ejemplo de participante

Posteriormente se debe describir el sistema actual.

La aplicación por desarrollar será una aplicación web, enfocada sobre todo a plataformas de escritorio, no obstante, podrá ser usada por dispositivos móviles y tabletas. La interfaz del sistema debe ser simple, debido a que va a ir destinada a todo tipo de usuarios, permitiendo así una buena eficiencia de uso sin necesidad de tener muchos conocimientos informáticos.

Para empezar la aplicación debe permitir a los usuarios registrarse, para ello necesitará introducir una serie de datos (Nombre, email, dirección, tipo de sangre, etc.). Una vez el usuario esté registrado, este será dado de alta en la base de datos y podrá modificar su ubicación.

El usuario podrá visualizar cuales son los hospitales más cercanos a su ubicación y ver sus datos (Nombre, dirección, número de teléfono, etc.)

El administrador de la aplicación podrá ver todos los donantes disponibles en cada provincia, así como la información de los hospitales de España y podrá emitir una alerta cuando se necesite un determinado tipo de sangre. Para difundir esta alerta será necesario escoger el hospital o centro de salud dónde se necesite sangre, el tipo de sangre que se solicita y el alcance de la difusión (municipio, provincia o CCAA), una vez emitida la alerta, se enviará un mensaje a todos los donantes que cumplan las condiciones a través de un chat interno en la aplicación.

A continuación, se *definen los actores* que formarán parte del sistema.

En la aplicación intervendrán distintos tipos de actores, y dependiendo de su tipo tendrán acceso a diferentes partes del sistema. En la *Figura 4* se detalla el diagrama de actores.

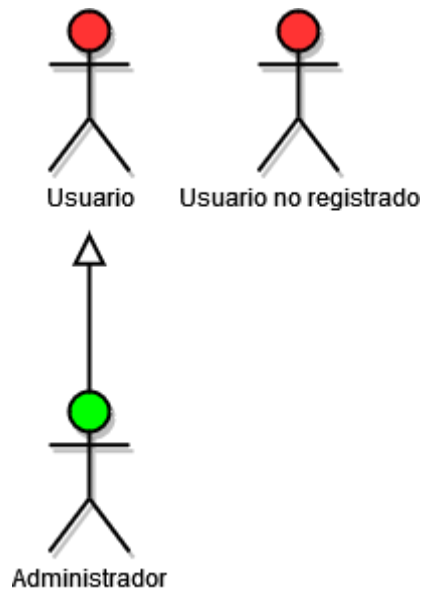


Figura 4: Diagrama de actores

Toda esta información se encuentra más detallada en el “Anexo II: Especificación de Requisitos Software”

A continuación, se detallarán los requisitos que tendrá el sistema, se dividirán en requisitos de información, funcionales y no funcionales. Para consultar la información completa de los requisitos se detallan en el “Anexo II: Especificación de Requisitos Software”

Los requisitos de información del sistema son:

- Información del usuario
- Información de los hospitales
- Mensajes

IRQ-001	Información del usuario
Versión	1.0 (12/05/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz santana
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-001] Gestión de usuarios
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-006] Protección e integridad de datos
Descripción	Se almacenará la información del usuario necesaria para el correcto funcionamiento del sistema.
Datos específicos	<ul style="list-style-type: none"> • Nombre y apellidos • Email • Municipio • Provincia • CCAA • Tipo de sangre
Importancia	Alta
Urgencia	Media
Estado	Activo
Estabilidad	Alta
Comentarios	El usuario podrá visualizar y modificar su información a través de una página de perfil.

Tabla 3: Ejemplo de Requisito de información

Los requisitos funcionales se detallarán en forma de tabla (*Tabla 3*), y, a su vez se dividirán en distintos paquetes agrupándolos por funcionalidades, de esta forma conseguimos una mejor distribución del proyecto.

- Gestión de usuarios: Detallado en la *Figura 5*
- Gestión de administrador: Detallado en la *Figura 6*

UC-001	Alta Usuario
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana

Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-001] Gestión de usuarios 												
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma • [NFR-006] Protección e integridad de datos 												
Descripción	El actor Usuario no registrado [ACT-001] desea darse de alta en el sistema.												
Precondición	El usuario no debe estar ya dado de alta.												
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El actor Usuario no registrado [ACT-001] entra en la aplicación web.</td> </tr> <tr> <td>2</td> <td>El actor Usuario no registrado [ACT-001] rellena los campos de registro.</td> </tr> <tr> <td>3</td> <td>El actor Usuario no registrado [ACT-001] presiona el botón de crear cuenta.</td> </tr> <tr> <td>4</td> <td>El sistema procede a crear la cuenta y almacenar su información.</td> </tr> <tr> <td>5</td> <td>El sistema redirige al usuario a la página de inicio de su cuenta.</td> </tr> </tbody> </table>	Paso	Acción	1	El actor Usuario no registrado [ACT-001] entra en la aplicación web.	2	El actor Usuario no registrado [ACT-001] rellena los campos de registro.	3	El actor Usuario no registrado [ACT-001] presiona el botón de crear cuenta.	4	El sistema procede a crear la cuenta y almacenar su información.	5	El sistema redirige al usuario a la página de inicio de su cuenta.
Paso	Acción												
1	El actor Usuario no registrado [ACT-001] entra en la aplicación web.												
2	El actor Usuario no registrado [ACT-001] rellena los campos de registro.												
3	El actor Usuario no registrado [ACT-001] presiona el botón de crear cuenta.												
4	El sistema procede a crear la cuenta y almacenar su información.												
5	El sistema redirige al usuario a la página de inicio de su cuenta.												
Postcondición	La cuenta del usuario estará registrada en el sistema.												
Excepciones	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>4</td> <td>El actor Usuario no registrado [ACT-001] introduce mal algún campo necesario en el registro.</td> </tr> <tr> <td>4</td> <td>El sistema no puede crear la cuenta.</td> </tr> <tr> <td>4</td> <td>El sistema informa al usuario de que introduzca el campo correctamente.</td> </tr> </tbody> </table>	Paso	Acción	4	El actor Usuario no registrado [ACT-001] introduce mal algún campo necesario en el registro.	4	El sistema no puede crear la cuenta.	4	El sistema informa al usuario de que introduzca el campo correctamente.				
Paso	Acción												
4	El actor Usuario no registrado [ACT-001] introduce mal algún campo necesario en el registro.												
4	El sistema no puede crear la cuenta.												
4	El sistema informa al usuario de que introduzca el campo correctamente.												
Importancia	Alta												
Urgencia	Alta												
Estado	Activo												
Estabilidad	Alta												
Comentarios	Ninguno												

Tabla 4: Ejemplo de Requisito funcional

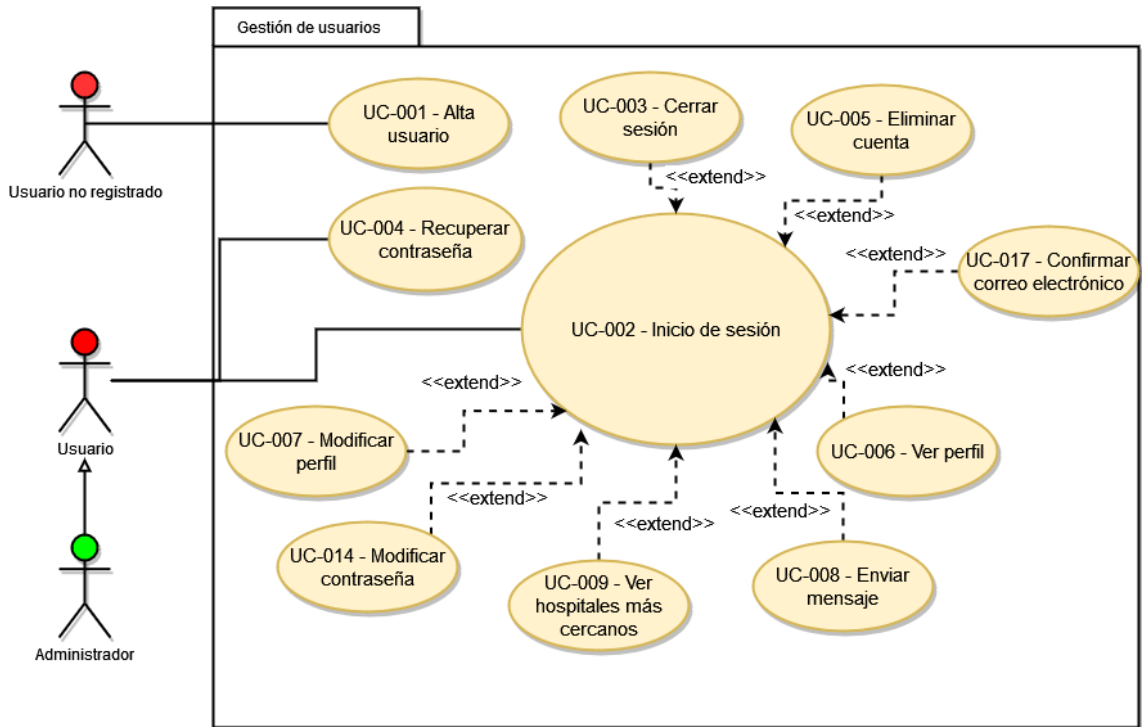


Figura 5: Requisitos funcionales - Paquete Gestión de usuarios

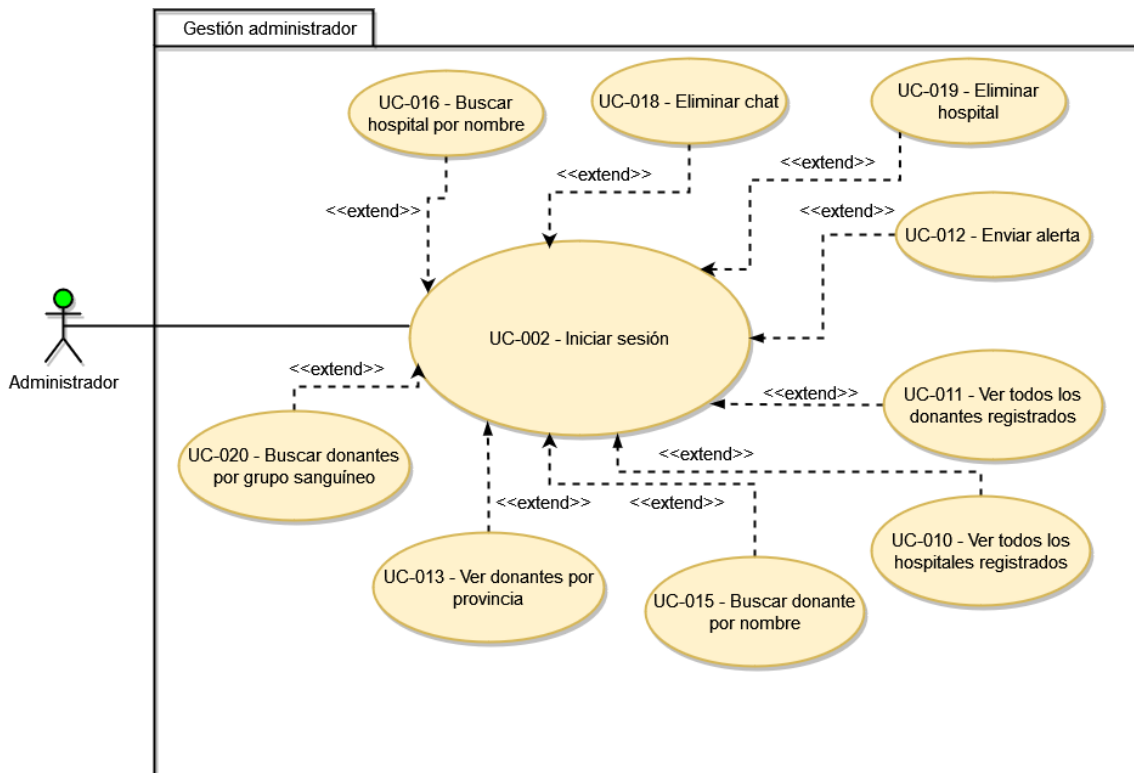


Figura 6: Requisitos funcionales - Paquete Gestión Administrador

Los requisitos no funcionales del sistema son:

- Usabilidad
- Eficiencia
- Fiabilidad
- Sistema responsivo y multiplataforma
- Disponibilidad
- Protección e integridad de datos

NFR-006	Protección e integridad de datos
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Dependencias	Todo el sistema
Descripción	El sistema debe proporcionar privacidad sobre los datos guardados en él, así mismo, debe ofrecer una integridad sobre ellos, cumpliendo con los requisitos sobre la ley orgánica de protección de datos en sistemas de información.
Importancia	Alta
Urgencia	Alta
Estado	Activo
Estabilidad	Alta
Comentarios	Ninguno

Tabla 5: Ejemplo de Requisito no funcional

4.5. Análisis de requisitos.

En este apartado se realizarán los procesos de análisis, refinamiento y estructuración para cada uno de los requisitos que posee el sistema para poder obtener una comprensión del sistema a bajo nivel.

Toda esta información se encuentra más detallada en el *“Anexo III: Análisis de requisitos”*

4.5.1. Modelo de dominio

El modelo de dominio puede ser tomado como un punto de partida para el diseño del sistema. Permite identificar las relaciones entre las entidades que forman el dominio del problema, y además identificar sus atributos.

Esto nos sirve para tener una visión estructural del dominio, es decir, obtener las necesidades de almacenamiento y la gestión de la información que va a necesitar el

sistema de una forma más esquemática y clara. El modelo de dominio se detalla en la *Figura 7*.

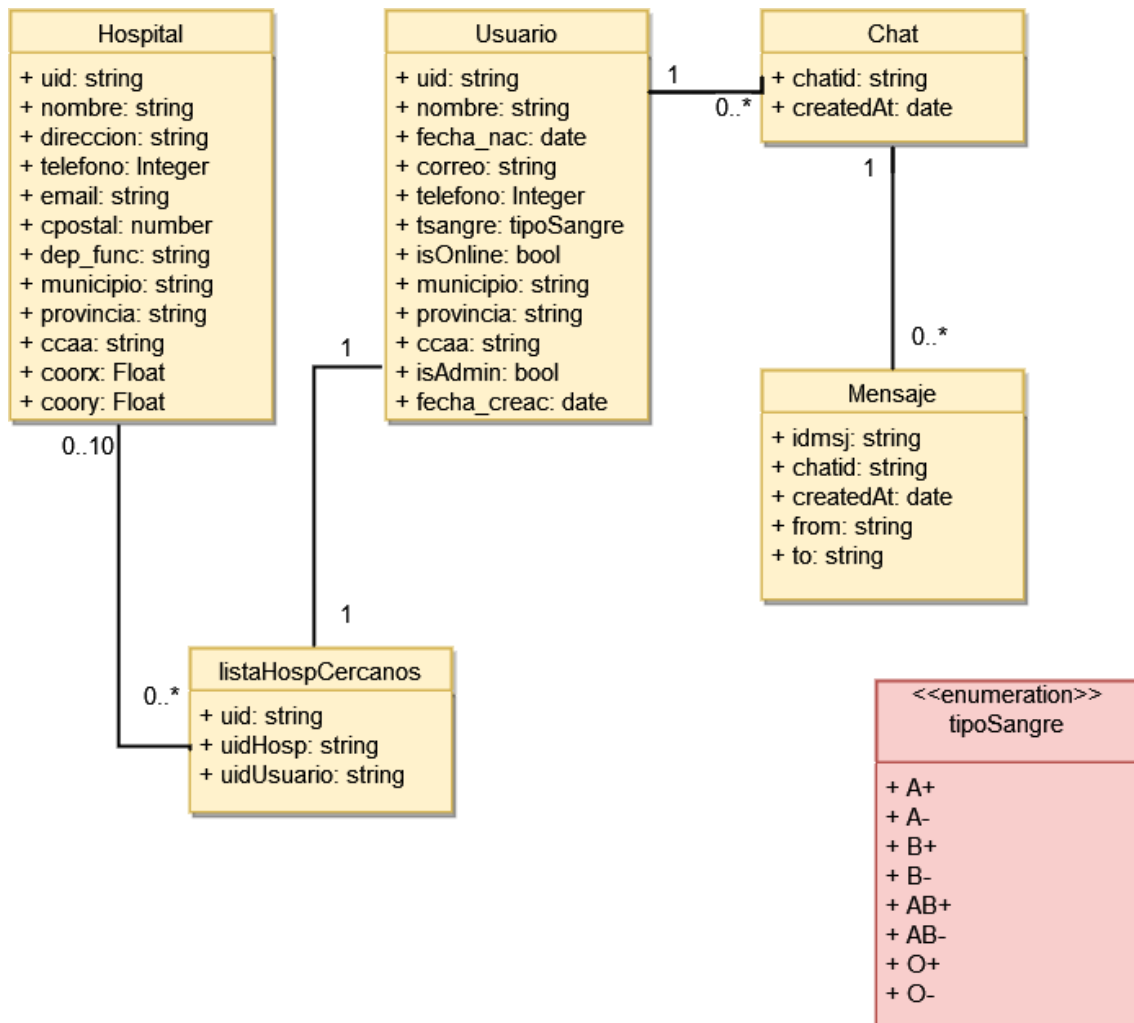


Figura 7: Modelo de dominio

4.5.2. Realización de casos de uso de análisis.

Mediante los diagramas de secuencia podemos visualizar cómo se ejecutarán los distintos casos de uso. A continuación, se muestra un ejemplo en la *Figura 8*.

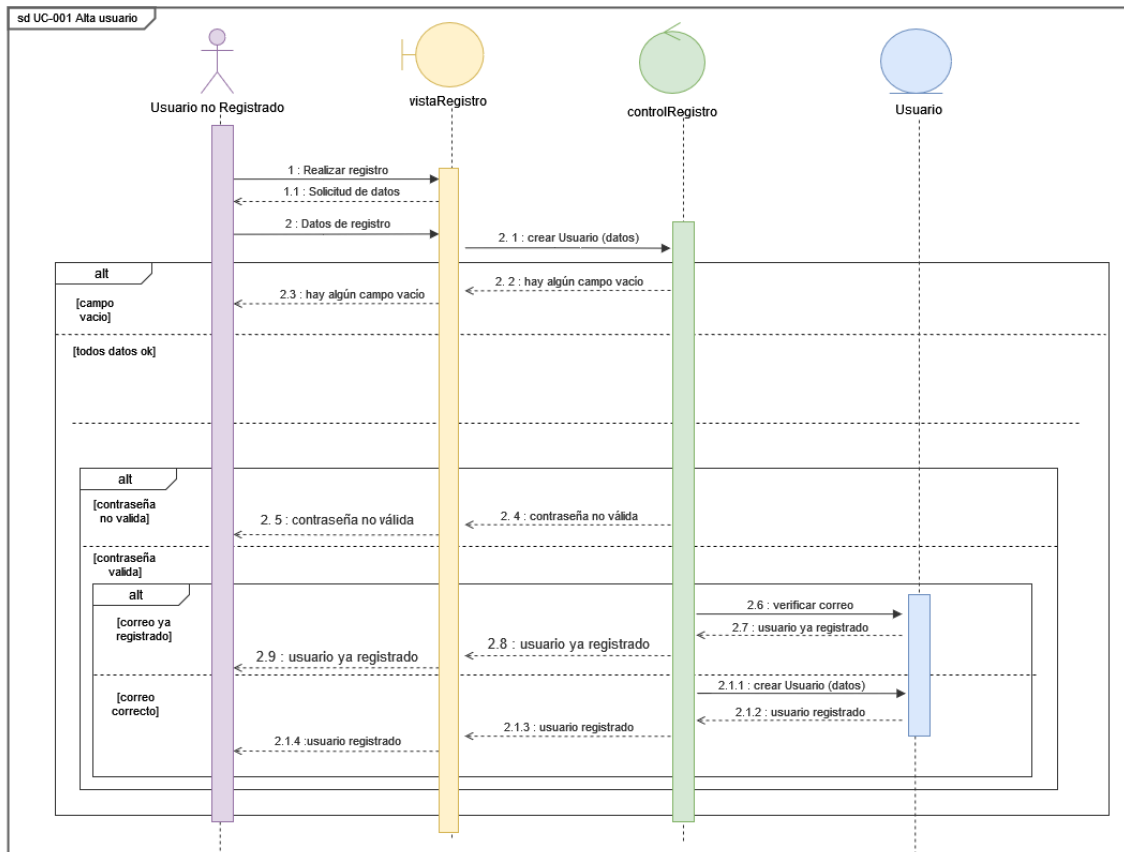


Figura 8: Ejemplo de diagrama de secuencia – Análisis

4.6. Diseño del software.

En este apartado se documenta el modelo de diseño del sistema. El modelo de diseño recoge la arquitectura global del sistema y como se relacionan sus componentes a través de una abstracción de la implementación.

Toda esta información se encuentra más detallada en el “Anexo IV: Diseño del sistema software”

4.6.1. Patrones arquitectónicos.

Para realizar el diseño del sistema se ha optado como guía el patrón arquitectónico **MVVM**. Debido a que este patrón está enfocado al desarrollo de interfaces de usuario.

Este patrón tiene como característica principal ofrecer un gran grado de desacoplo entre la interfaz de usuario y la lógica de la aplicación [20].

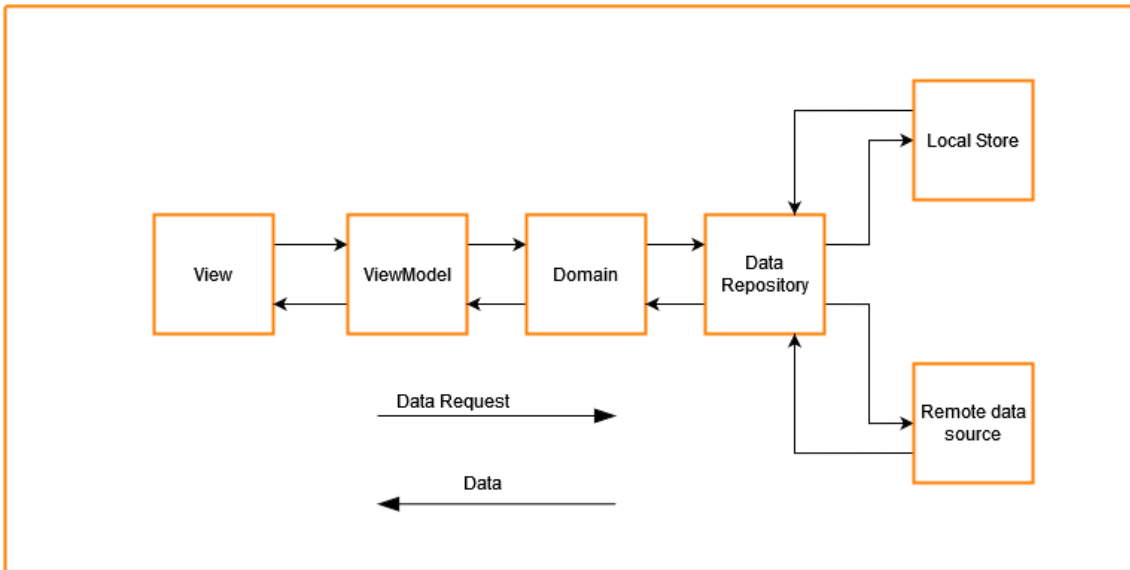


Figura 9: Patrón arquitectónico MVVM

Complementariamente, se utilizará el patrón **“Presentation and Container componentes”**

Este patrón divide el código en dos tipos de componentes, Presentation components y container components. Este tipo de patrón presenta varios beneficios, como, por ejemplo, mejor separación del código, debido a que por un lado tenemos la apariencia y por otro la lógica. También permite la reutilización del código ya que podemos crear componentes que sirvan para varias interfaces.

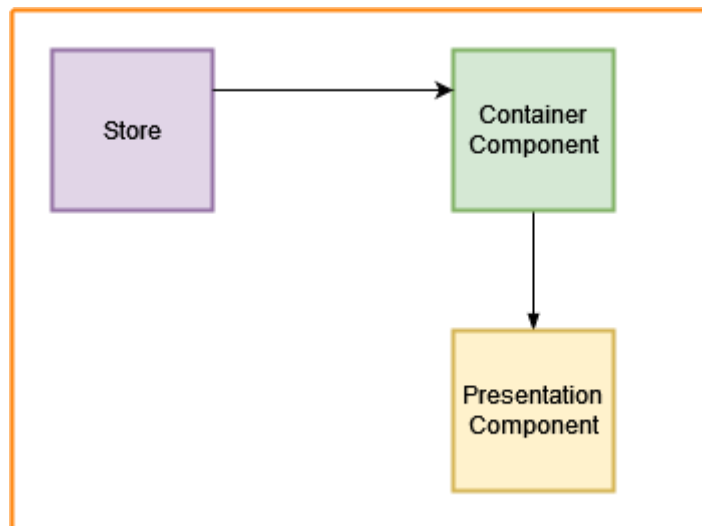


Figura 10: Patrón arquitectónico Presentation and Container components

4.6.2. Subsistemas de Diseño.

Tomando como punto de partida los patrones arquitectónicos definidos anteriormente, estableceremos las relaciones que tienen entre ellos los distintos componentes, en nuestra aplicación solo tendremos un sistema de diseño.

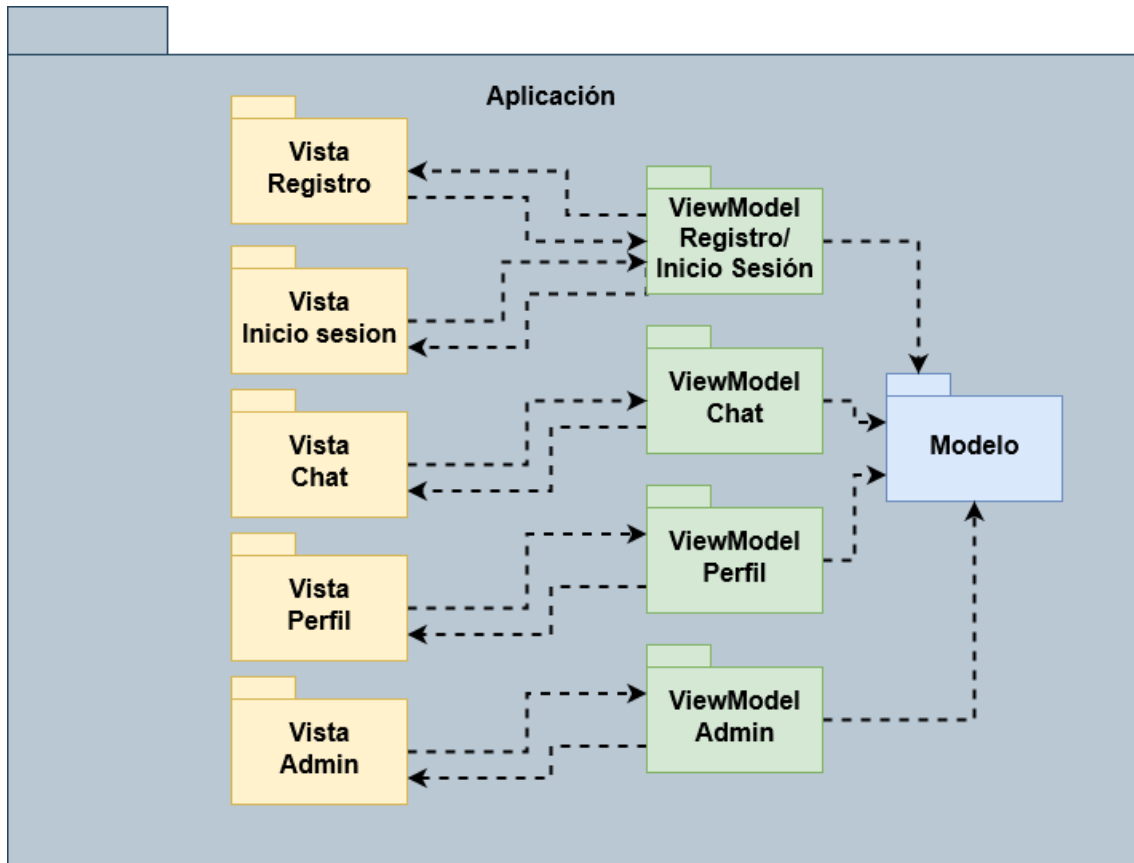


Figura 11: Diagrama subsistemas de diseño

4.6.3. Clases de diseño.

En este apartado se detallarán los paquetes que componen el sistema.

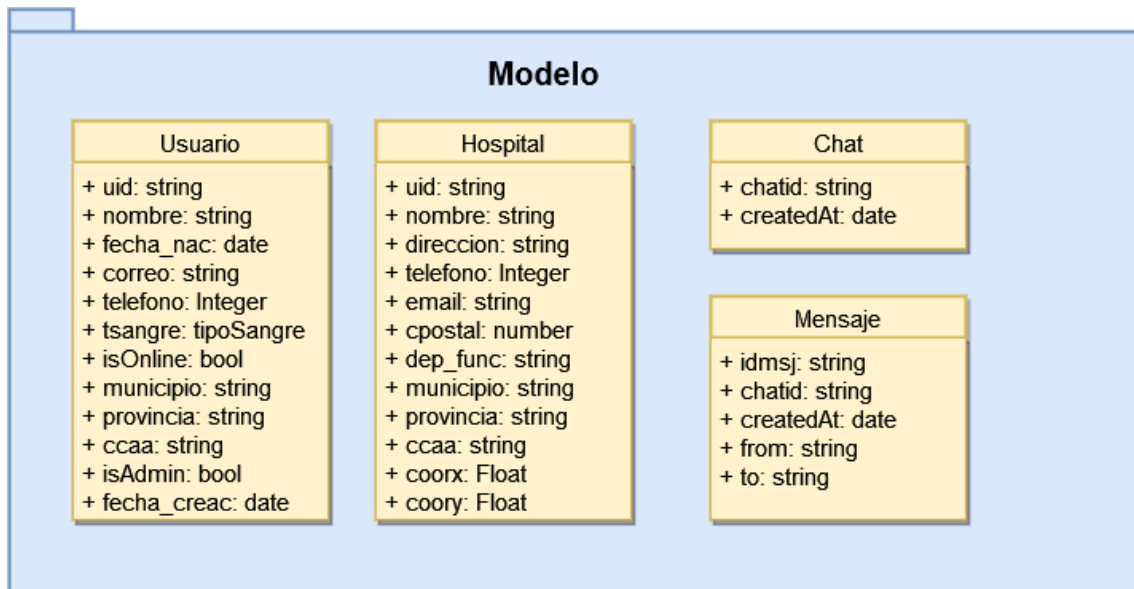


Figura 12: Clases de diseño Modelo

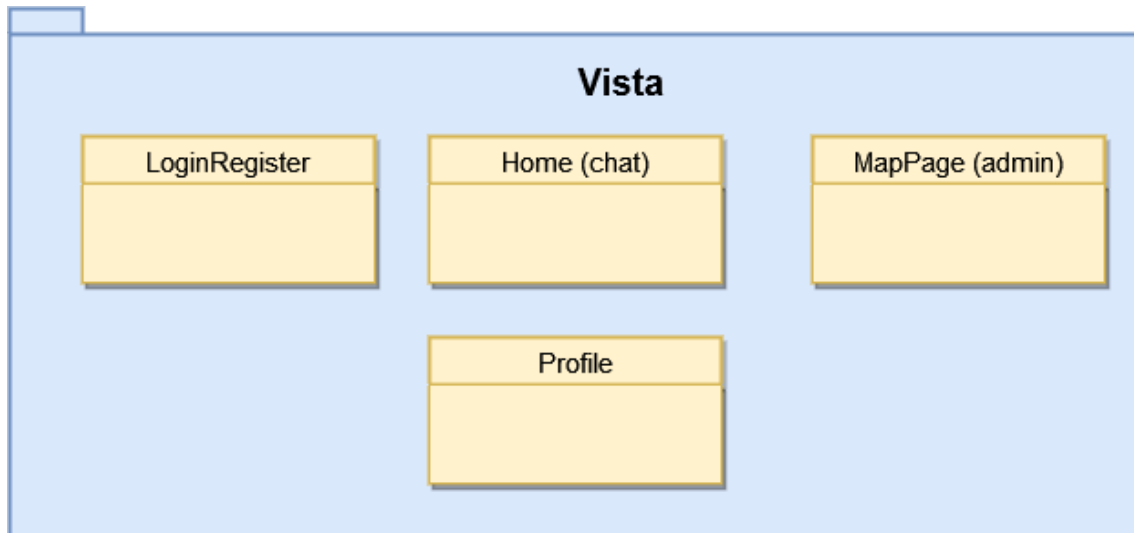


Figura 13: Clases de diseño Vista

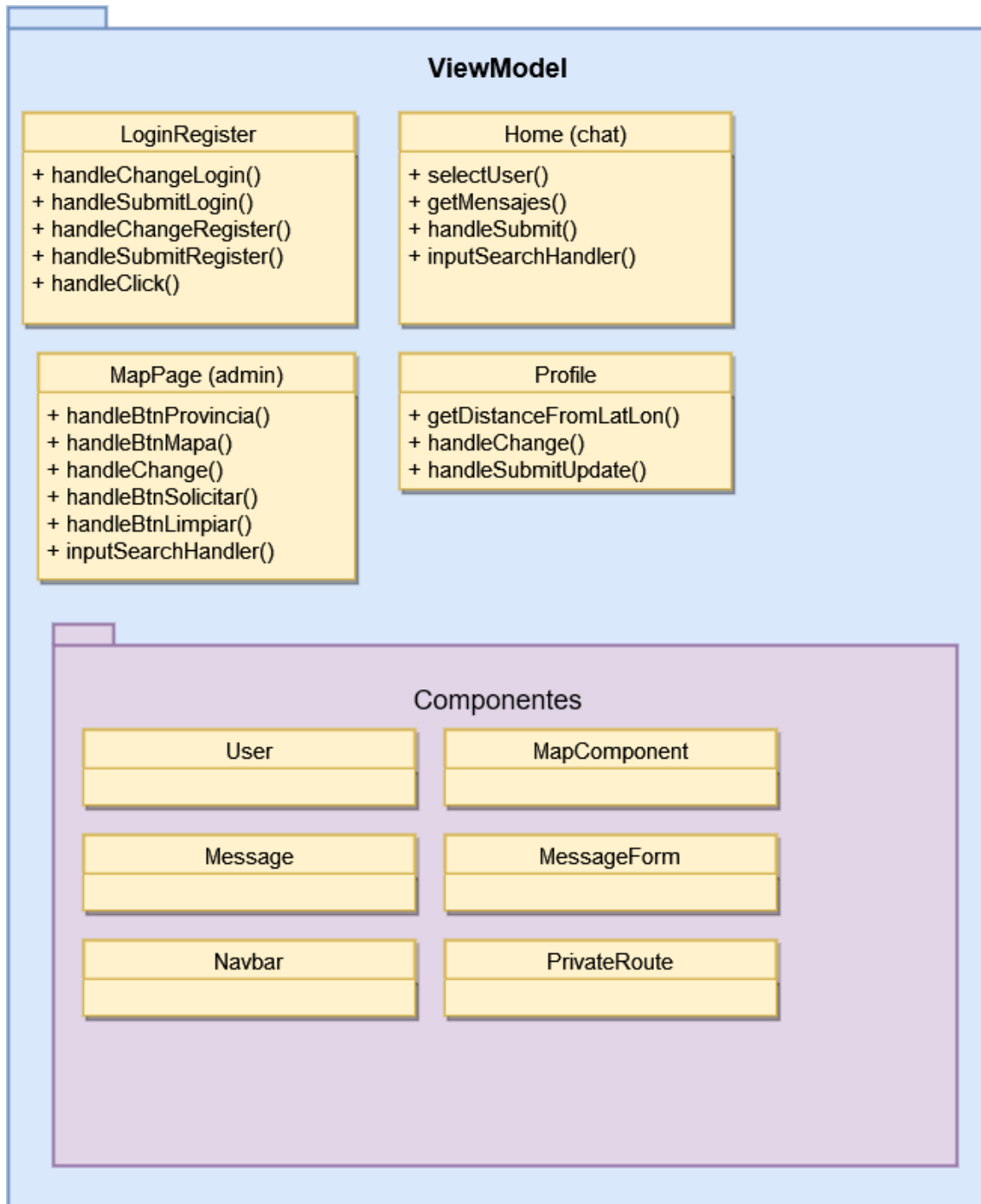


Figura 14: Clases de diseño ViewModel

4.6.4. Vista arquitectónica de diseño.

En este apartado se detalla como quedaría todo el sistema con todas las relaciones, siguiendo los patrones descritos previamente.

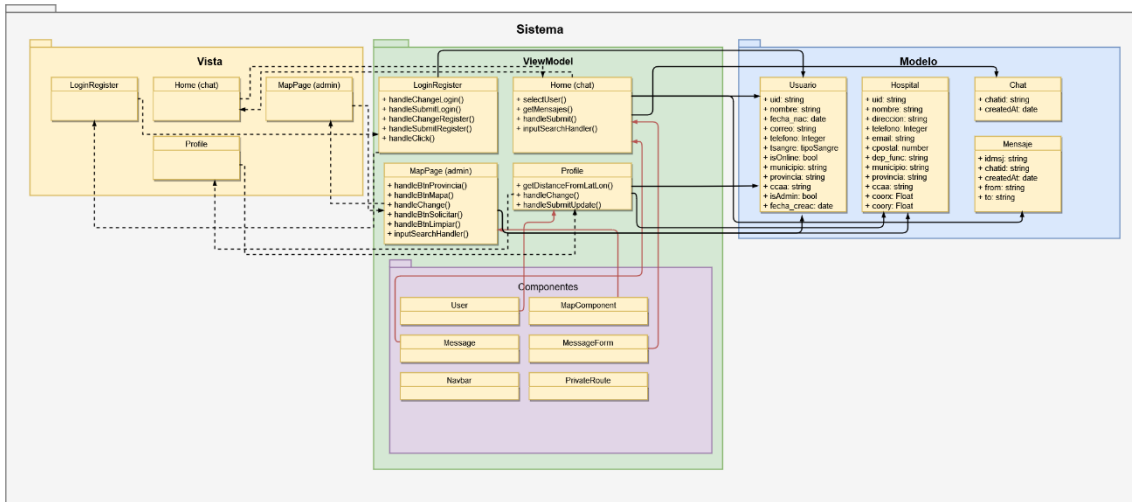


Figura 15: Vista arquitectónica

4.6.5. Realización de casos de uso de diseño.

En este apartado se especifica el flujo de funciones que se llevan a cabo entre los distintos componentes del sistema para llevar a cabo una funcionalidad. Esto se realiza a través de diagramas. Se muestra un diagrama de ejemplo en la Figura 16.

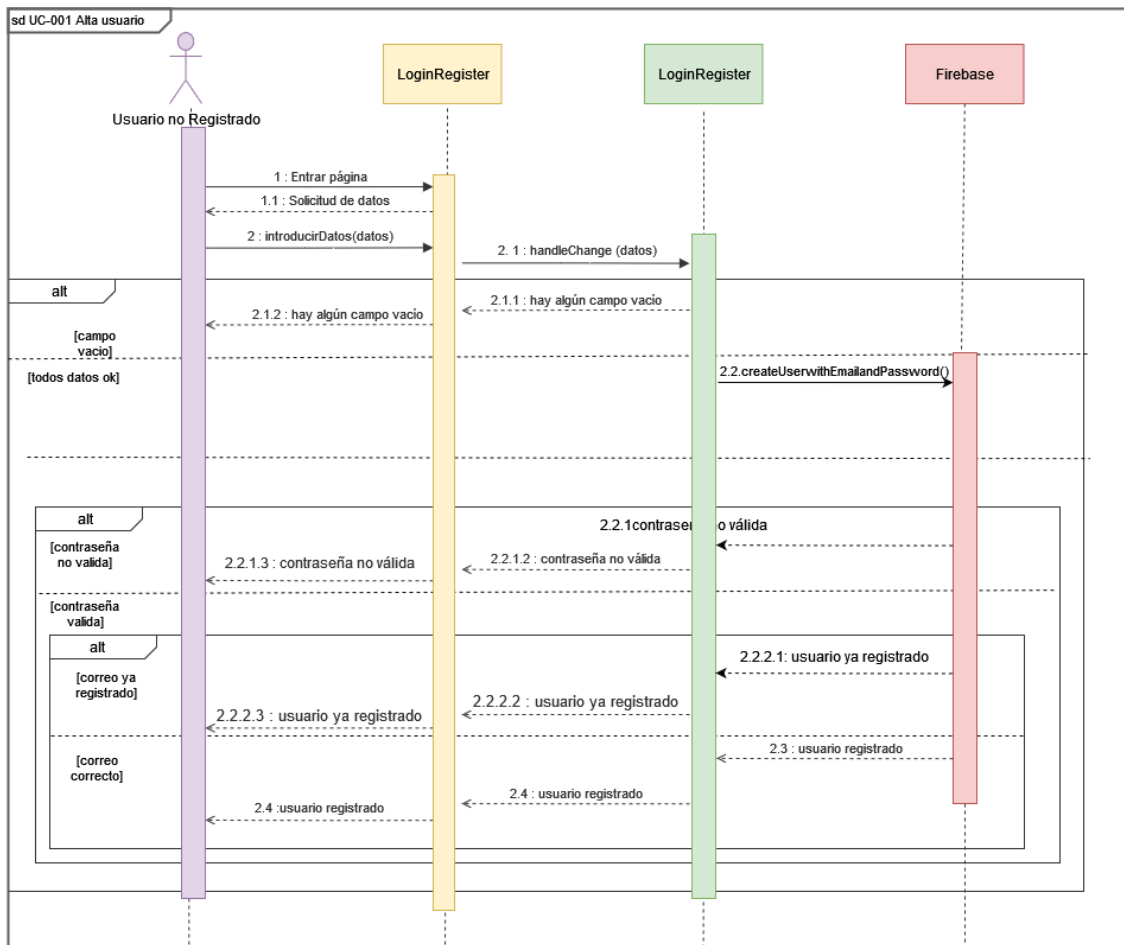


Figura 16: Ejemplo Diagrama de secuencia – Diseño

Toda esta información se encuentra más detallada en el “Anexo IV: Diseño del sistema software”

4.6.6. Modelo de despliegue.

En esta fase se detallan las actividades necesarias para desplegar la aplicación en un entorno real.

La aplicación web se desplegará sobre un servidor de hosting fiable, para así garantizar al usuario la disponibilidad del sistema, a la aplicación se accederá a través de un navegador de internet. La base de datos que utiliza el sistema es Firebase y a su vez utiliza la api de OpenRouteService.

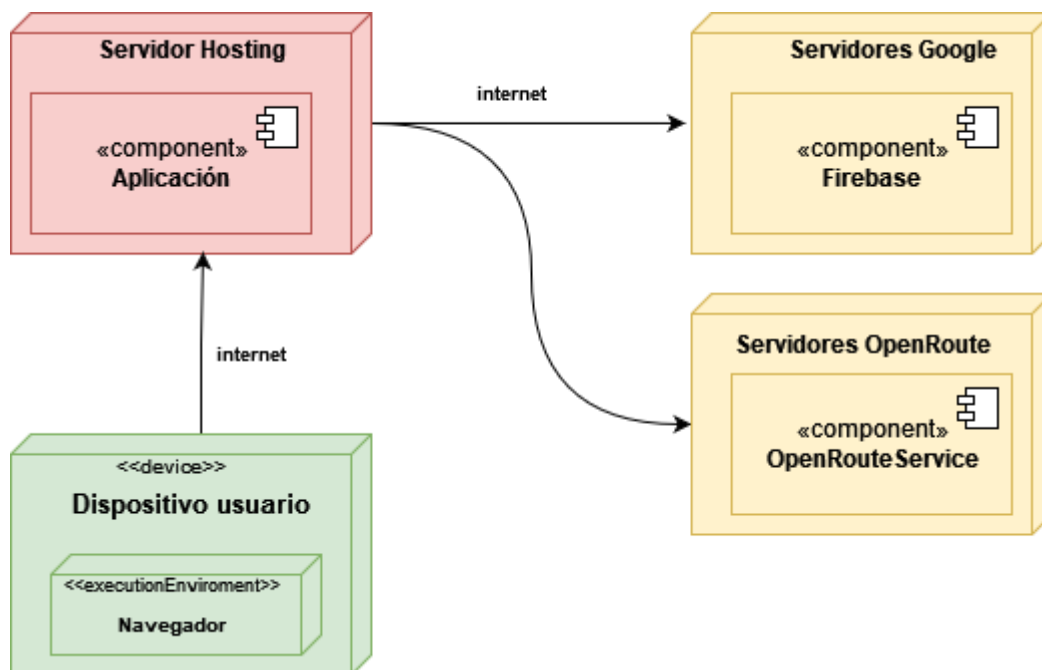


Figura 17: Modelo de despliegue

4.6.7. Prototipo digital.

Para finalizar con esta fase, se diseñaron las distintas interfaces de las que dispondrá el sistema. Se detalla una interfaz a modo de ejemplo en la Figura 18.

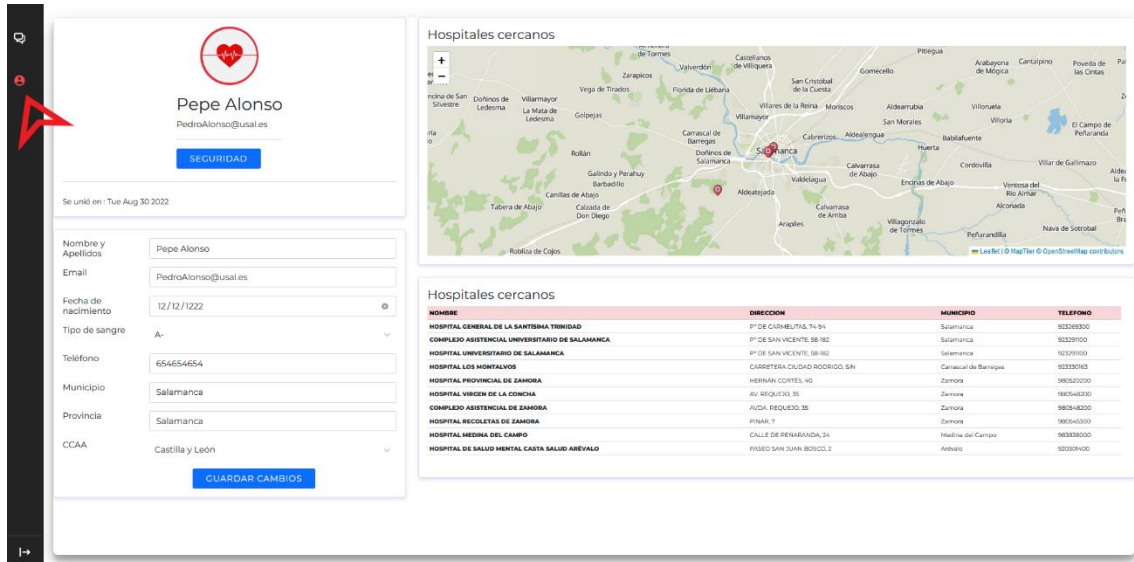


Figura 18: Ejemplo Prototipo digital - Perfil

Para visualizar todas las interfaces prototipadas ver “Anexo IV: Diseño del sistema software”

4.7. Implementación.

Esta fase conforma la construcción de la aplicación. Para ello se han seguido los esquemas planteados a lo largo de la especificación, análisis y diseño de los requisitos.

Para su realización se han tenido que poner en práctica las técnicas y herramientas detalladas en el apartado “Técnicas y herramientas”.

4.7.1. Implementación del código.

Para la implementación del código, ya se partía con una base de conocimientos en JavaScript, sin embargo, se tuvo que realizar un proceso de aprendizaje de la librería React, no se tardó mucho debido a que la curva de aprendizaje de esta librería es bastante sencilla, por lo que pronto pudo empezar el desarrollo de la aplicación.

Una vez realizada la implementación del código, se realizó la documentación técnica, mediante herramientas especializadas que ya hemos detallado anteriormente [15]. La documentación técnica se ha generado en formato HTML.

Para obtener más información por favor, consulte “Anexo V: Documentación técnica de programación”.

Module: components/MessageForm

Componente MensajeForm, componente donde el usuario escribe el mensaje

Parameters:

Name	Type	Description
handleSubmit	function	Funcion para manejar el submit - La manda desde Home.js
text	String	El texto que se escribe es un bind para poder actualizar el value del input
setText	function	Setter del text

Source: [components/MessageForm.js, line 3](#)

Home

Modules

Chat
 components/Loading
 components/MapComponent
 components/MapComponentUser
 components/Message
 components/MessageForm
 components/Navbar
 components/PrivateRoute
 components/PrivateRouteAdmin
 components/Tabla
 components/User
 components/UserInfo
 Login/Register
 pages/MapPage
 pages/Profile
 Global

Figura 19: Ejemplo de documentación – Componente

4.7.2. Implementación de las interfaces.

Para la implementación de las interfaces se han seguido los principios C.R.A.P [21].

- **Contraste:** Diferenciar elementos según su prioridad, puede ser con colores y con distintos tamaños.
- **Repetición:** El estilo de las interfaces debe ser consistente en toda la aplicación. De esta forma el usuario se familiarizará pronto con la aplicación.
- **Alineación:** Los elementos con la misma prioridad deberán ser tratados y dispuestos por igual.
- **Proximidad:** Los elementos relacionados entre si deben estar cerca, separados por algún tipo de delimitador.

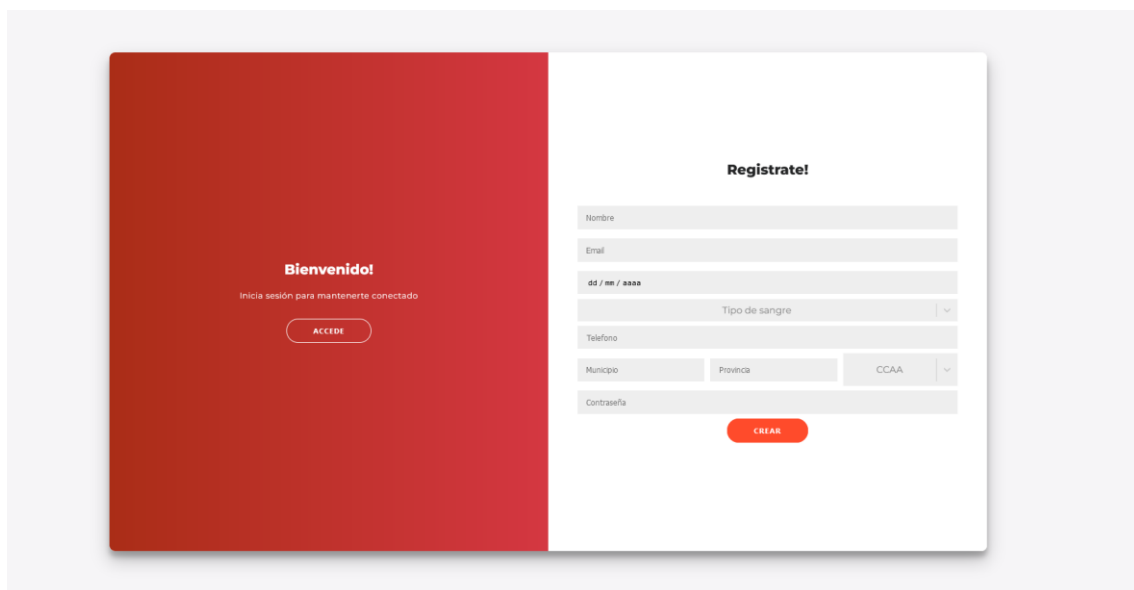


Figura 20: Interfaz del login.

Para la paleta de colores se escogieron tonalidades rojizas, debido a que la aplicación iba destinada al campo de la salud y a las donaciones de sangre, las interfaces se han realizado siguiendo los principios C.R.A.P, y, además, implementando interfaces adaptativas como se muestran en la Figura 21, en la Figura 22 y en la Figura 23.

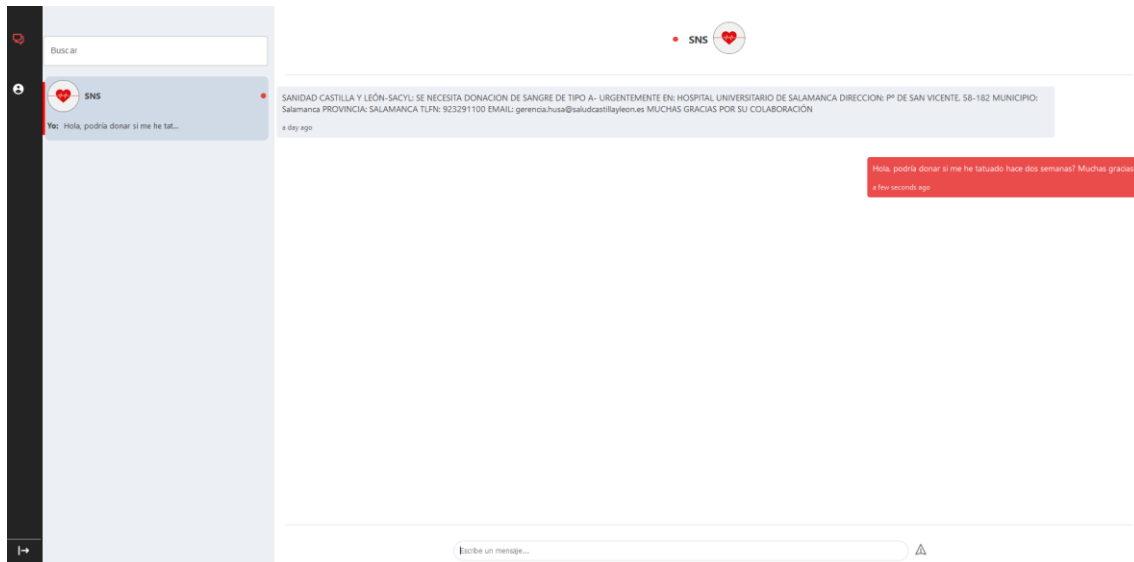


Figura 21: Interfaz responsive - 1.

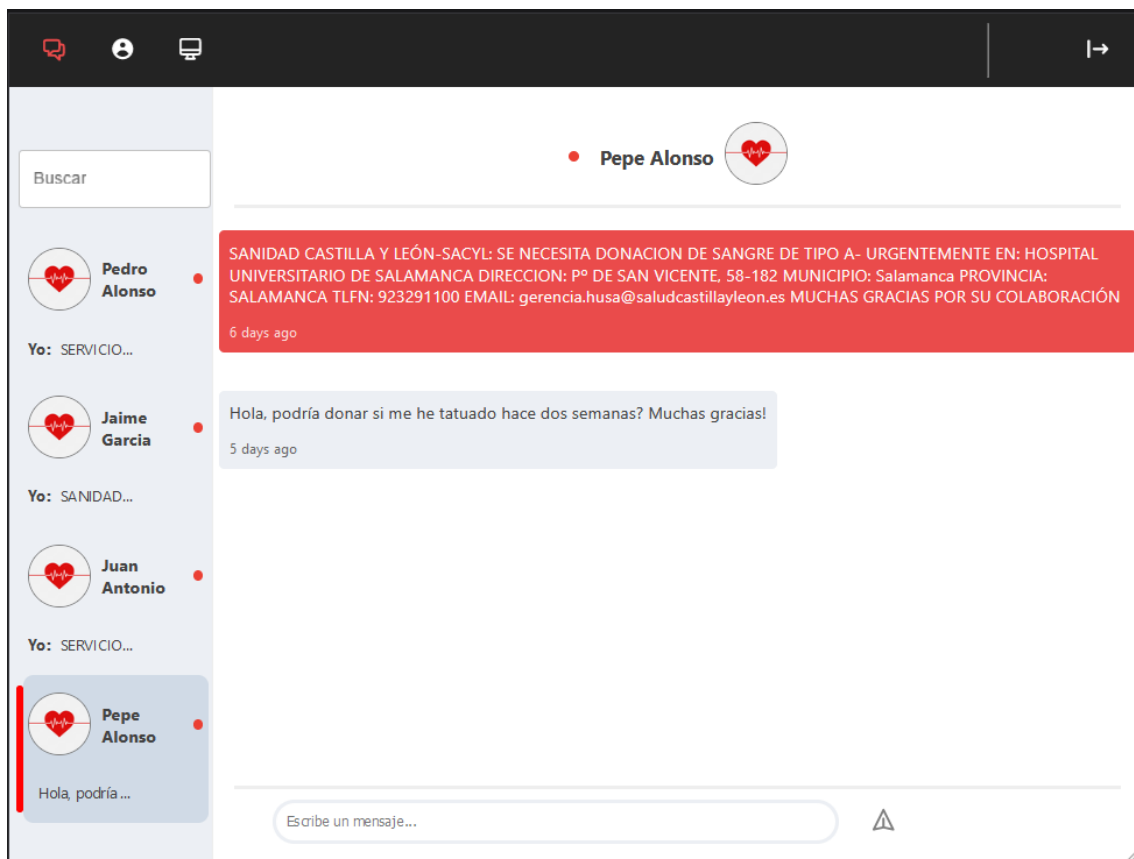


Figura 22: Interfaz responsive - 2.

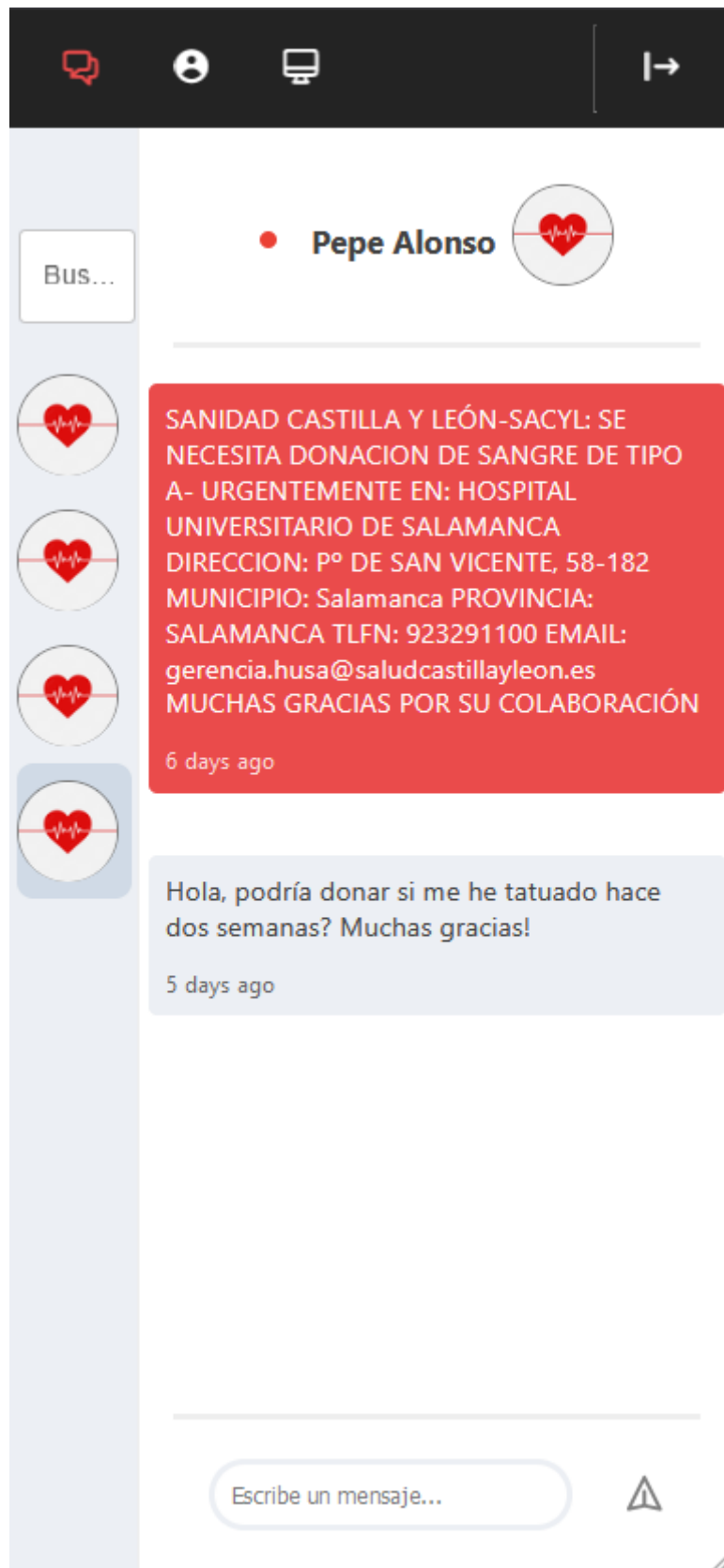


Figura 23: Interfaz responsive - 3.

4.8. Pruebas.

Las pruebas forman una parte fundamental del ciclo de desarrollo software, porque permite identificar fallos y corregirlos, lo que provoca un producto final de mayor calidad.

Como desarrollador, se ha llevado a cabo un proceso de pruebas unitarias de cada funcionalidad, solucionando aquellos errores encontrados.

También se ha testeado la aplicación con personas ajenas al desarrollo, y con distinto grado de familiarización con la tecnología, con el fin de encontrar fallos en casuísticas que, como desarrollador familiarizado con la aplicación, no contemplamos.

4.9. Funcionalidad del sistema.

En este apartado se detallarán las funcionalidades del sistema de una manera resumida. Para obtener más información, se puede consultar en *“Anexo VI: Manual de usuario”*.

Registro.

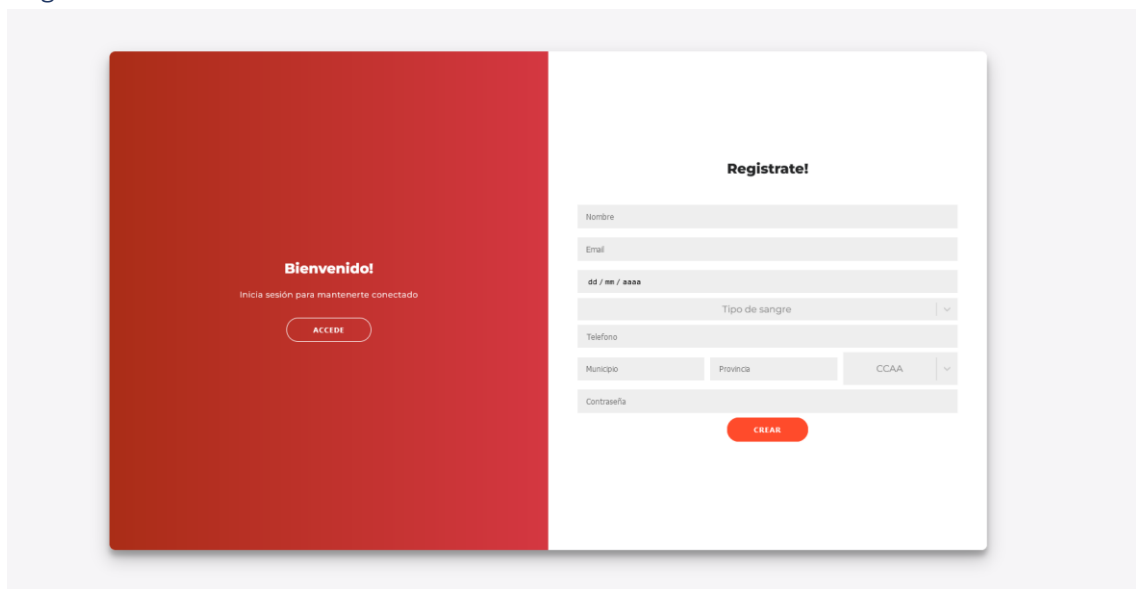


Figura 24: Registro

Inicio de sesión.

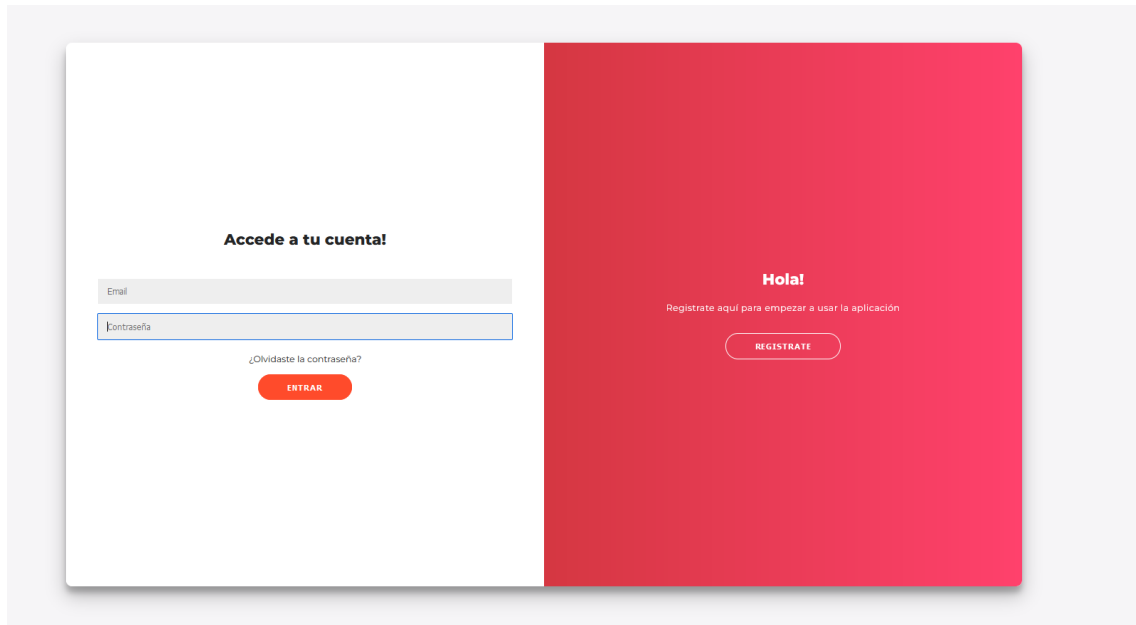


Figura 25: Inicio de sesión.

Recuperar contraseña.

Se permite al usuario recuperar la contraseña introduciendo el email con el que se registró, tal y como se muestra en la *Figura 26* y *Figura 27*.

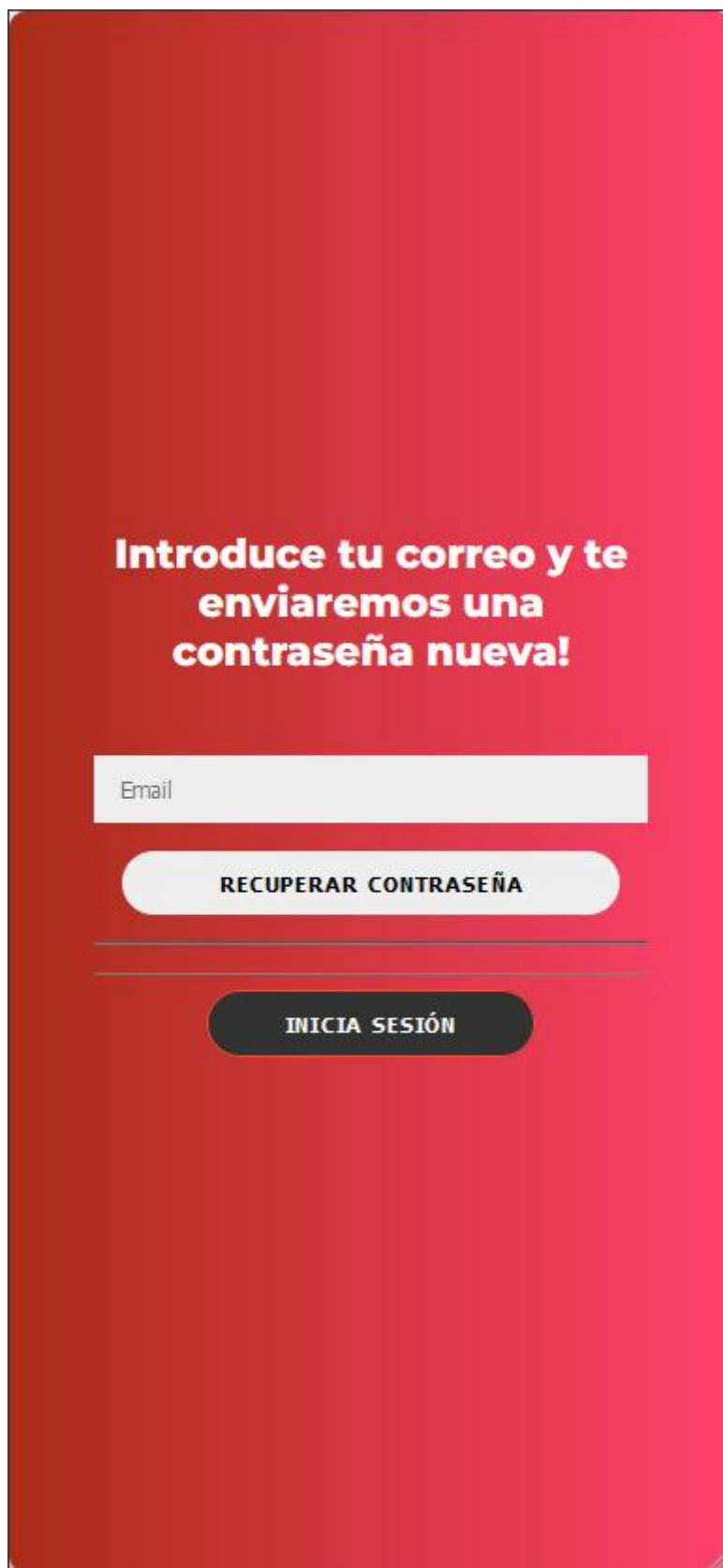


Figura 26: Recuperar contraseña.

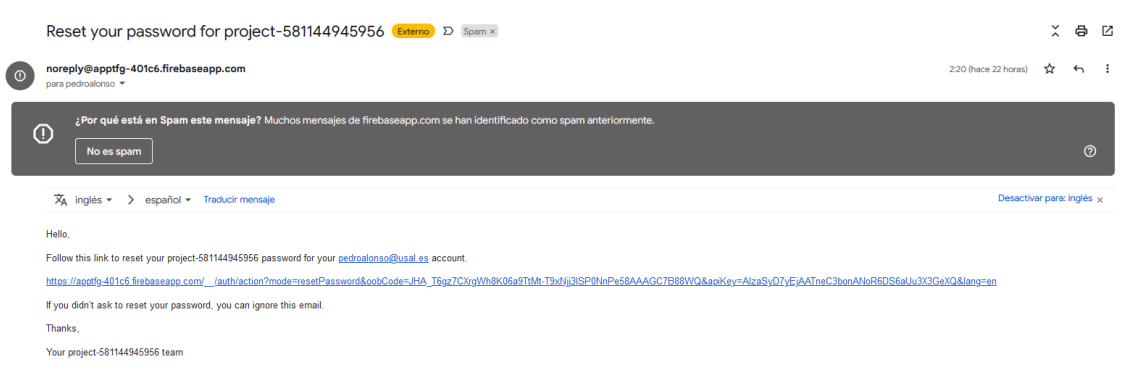


Figura 27: Correo de recuperación de contraseña.

Funcionalidades del chat.

Se permite al usuario comunicarse a través de un chat interno.

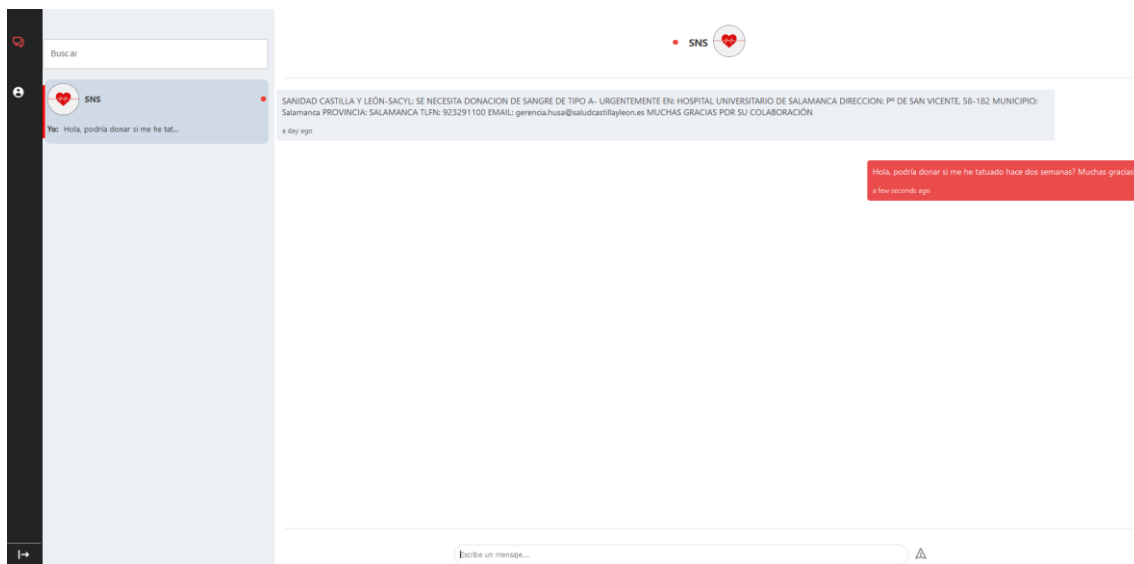


Figura 28: Chat.

Perfil.

Hospitales cercanos

NOMBRE	DIRECCION	MUNICIPIO	TELEFONO
HOSPITAL GENERAL DE LA SANTISIMA TRINIDAD	PT DE CARMELITAS, 74-94	Salamanca	923203000
COMPLEJO ASISTENCIAL UNIVERSITARIO DE SALAMANCA	PT DE SAN VICENTE, 58-182	Salamanca	923207000
HOSPITAL UNIVERSITARIO DE SALAMANCA	PT DE SAN VICENTE, 58-182	Salamanca	923207000
HOSPITAL LOS MONTAÑOS	CARRITERA CIUDAD RODRIGO, SAN	Carreras de Bañegas	923330043
HOSPITAL PROVINCIAL DE ZAMORA	HERNAN CORTES, 40	Zamora	980520000
HOSPITAL VIRGEN DE LA CONCHA	AV REQUERO, 35	Zamora	980540000
COMPLEJO ASISTENCIAL DE ZAMORA	AVDA. REQUERO, 35	Zamora	980540000
HOSPITAL RECOLETAS DE ZAMORA	PIÑAL, 7	Zamora	980540000
HOSPITAL MEDINA DEL CAMPO	CALLE DE PENARANDA, 24	Medina del Campo	983830000
HOSPITAL DE SALUD MENTAL CASTA SALUD AREVALO	PASEO SAN JUAN BOSCO, 2	Arevalo	920204000

Figura 29: Perfil

Modificar información.

Se permite al usuario modificar su información personal.

Hospitales cercanos

NOMBRE	DIRECCION	MUNICIPIO	TELEFONO
HOSPITAL GENERAL DE LA SANTISIMA TRINIDAD	PT DE CARMELITAS, 74-94	Salamanca	923203000
COMPLEJO ASISTENCIAL UNIVERSITARIO DE SALAMANCA	PT DE SAN VICENTE, 58-182	Salamanca	923207000
HOSPITAL UNIVERSITARIO DE SALAMANCA	PT DE SAN VICENTE, 58-182	Salamanca	923207000
HOSPITAL LOS MONTAÑOS	CARRITERA CIUDAD RODRIGO, SAN	Carreras de Bañegas	923330043
HOSPITAL PROVINCIAL DE ZAMORA	HERNAN CORTES, 40	Zamora	980520000
HOSPITAL VIRGEN DE LA CONCHA	AV REQUERO, 35	Zamora	980540000
COMPLEJO ASISTENCIAL DE ZAMORA	AVDA. REQUERO, 35	Zamora	980540000
HOSPITAL RECOLETAS DE ZAMORA	PIÑAL, 7	Zamora	980540000
HOSPITAL MEDINA DEL CAMPO	CALLE DE PENARANDA, 24	Medina del Campo	983830000
HOSPITAL DE SALUD MENTAL CASTA SALUD AREVALO	PASEO SAN JUAN BOSCO, 2	Arevalo	920204000

Figura 30: Modificar información

Modificar contraseña y eliminar cuenta.

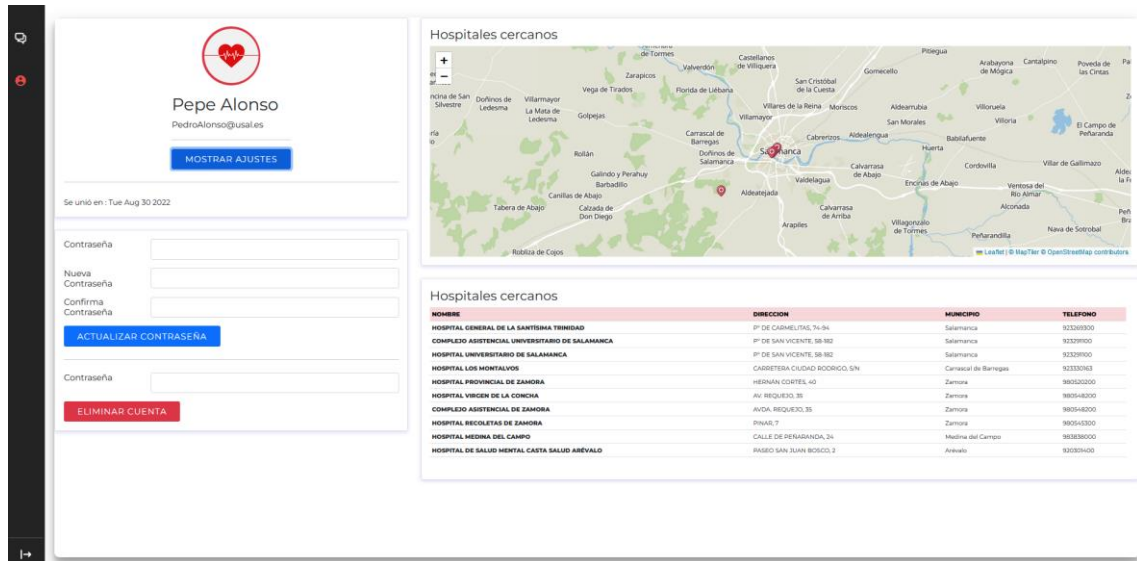


Figura 31: Modificar contraseña y eliminar cuenta

Ver hospitales en el mapa.

Se permite al usuario interactuar con el mapa, para ver la información de los hospitales y también su ubicación. Simplemente debe pulsar sobre cualquier hospital.

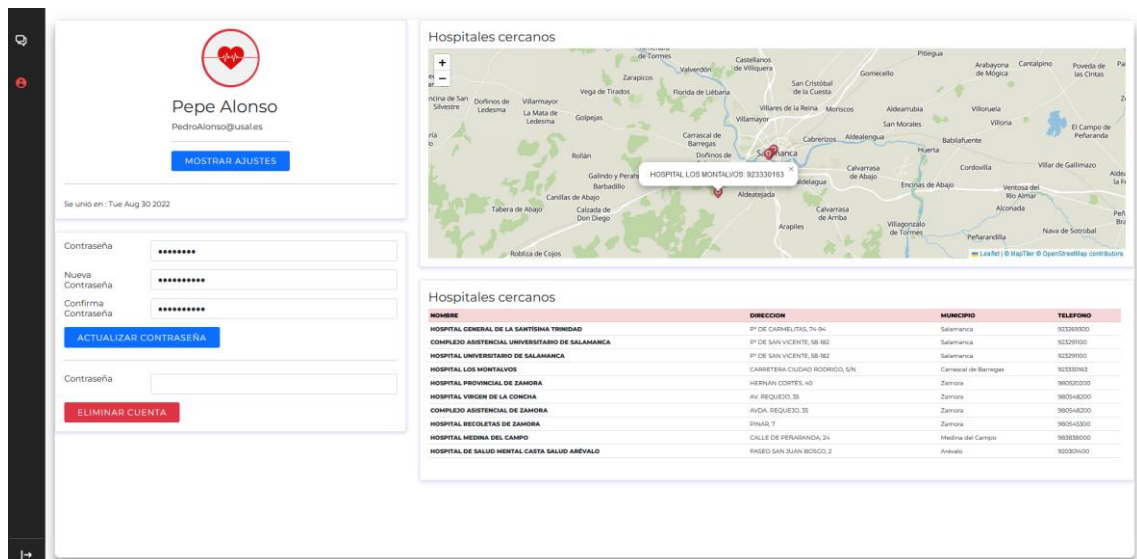


Figura 32: Ver hospitales en el mapa

Ver página de control.

El actor administrador dispone de una página de control, para administrar las alertas de solicitud de donación.

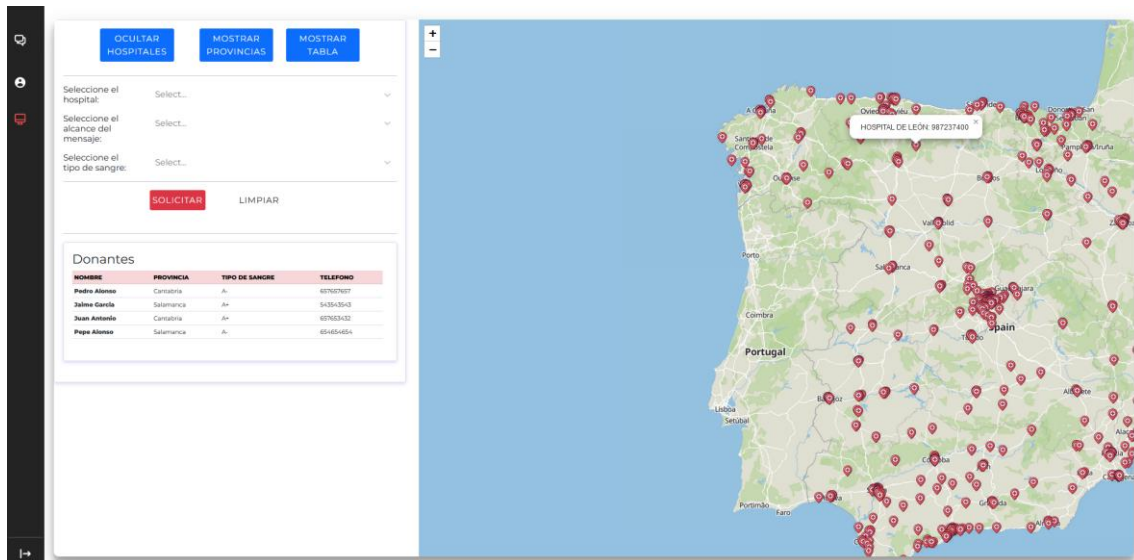


Figura 33: Ver hospitales en el mapa - Administrador

Ver donantes por provincias.

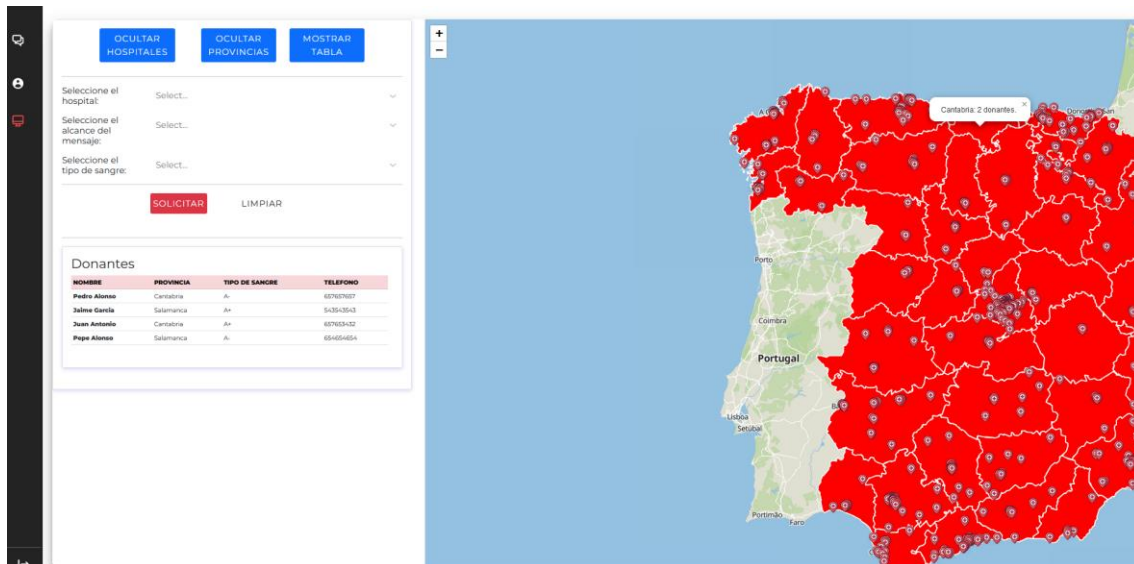


Figura 34: Ver donantes por provincia - Administrador

Enviar alerta de solicitud de sangre.

Los campos que rellenar son:

- **Hospital:** Se necesita especificar en que hospital se necesita sangre.
- **Alcance:** Se refiere a la difusión del mensaje, puede ser Municipio, Provincia o CCAA.
- **Tipo de sangre:** Tipo de sangre que se solicita.

Una vez rellenados se pulsará en solicitar y se difundirá un mensaje a todos los donantes que cumplan las condiciones. También se podrá visualizar los usuarios a los que se les ha enviado la alerta, tal y como se detalla en la figura 25.

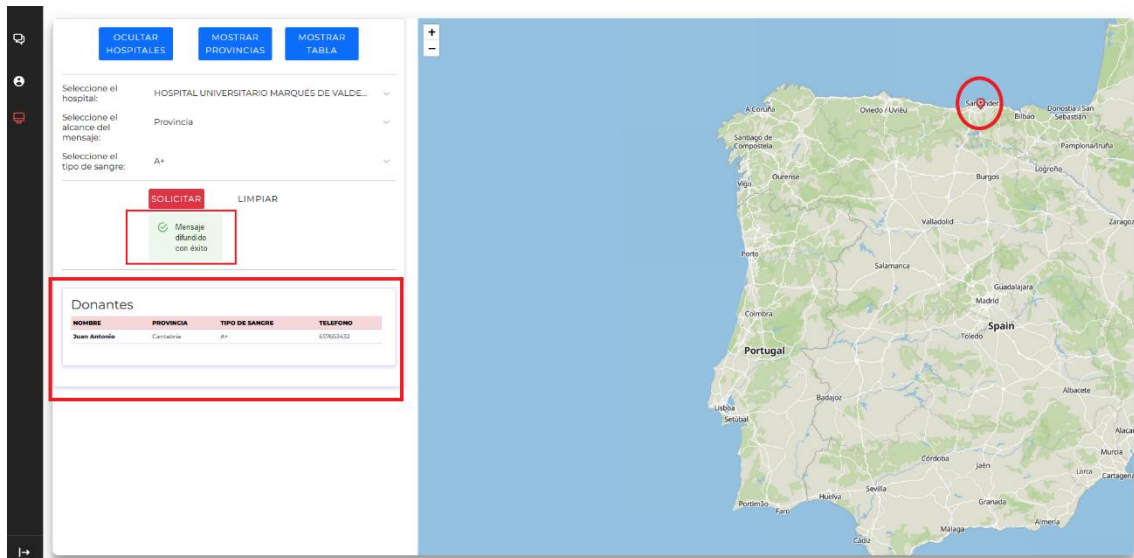


Figura 35: Enviar alerta de solicitud de sangre

En el mensaje de alerta se indicará el servicio de salud responsable, el tipo de sangre que necesitan y la ubicación del hospital.

SERVICIO CÁNTABRO DE SALUD-SCS: SE NECESITA DONACION DE SANGRE DE TIPO A- URGENTEMENTE EN: HOSPITAL UNIVERSITARIO MARQUÉS DE VALDECILLA DIRECCION: AV. DE VALDECILLA, S/N MUNICIPIO: Santander PROVINCIA: CANTABRIA TLFN: 942202520 EMAIL: dirger.humv@scsalud.es MUCHAS GRACIAS POR SU COLABORACIÓN

Figura 36: Mensaje de alerta enviado

5. Conclusiones y líneas de trabajo futuras.

Para concluir, se presentan las conclusiones tras la realización del proyecto “Aplicación buscadora de bancos de sangre y donación de sangre”

Conclusiones.

La realización de este proyecto ha supuesto un gran reto en lo personal, pero desde luego finaliza con una sensación gratificante. Desde el principio ha sido un constante aprendizaje de nuevas tecnologías, herramientas y métodos para realizar el desarrollo de la aplicación.

Además, el desarrollo de un proyecto que puede contribuir a un mejor funcionamiento de una parte importante del sistema sanitario como son las donaciones aumenta esa sensación de haber contribuido en algo a la sociedad.

Durante este proyecto se han aplicado los conocimientos adquiridos durante 4 años de multitud de asignaturas. En este proyecto las asignaturas que más he encontrado realmente útiles han sido las dedicadas a Ingeniería del Software, Interacción Persona Ordenador y Programación, que, aunque no se haya estudiado programación web, los fundamentos de la programación aprendidos en otro lenguaje facilitan el aprendizaje de nuevos lenguajes de programación y tecnologías.

De este proyecto me llevo el aprendizaje y la mejora de mis conocimientos de *front-end* y *back-end*, además de un refuerzo en el diseño de interfaces agradables a la vista y fáciles de usar. Con todo esto, me sumo a un mercado laboral en el que las ofertas de desarrollador web son abundantes, por lo que aprender a desarrollar este tipo de Software es de gran valor.

Por último y, para resumir, el llevar a cabo este proyecto, ha supuesto un reto sobre todo en la organización, el ver que soy capaz de desarrollar una aplicación funcional desde 0 compaginándolo con un trabajo, hace que aumente mi confianza y mis ganas de emprender y llevar a cabo más proyectos personales.

Líneas futuras de trabajo.

Creo que el potencial de esta aplicación es bastante alto, si se consigue centralizar todas las funcionalidades necesarias. Es por ello por lo que durante el desarrollo han ido surgiendo ideas que no han sido posibles de implementar pero que podrían ser unas ampliaciones interesantes.

- **Poder comunicarse con otros bancos de sangre a través de la aplicación:** El sistema está diseñado para ser gestionado por un solo servicio de salud, sin embargo, sería interesante que pudiese haber varios servicios encargados, y poder comunicarse entre sí, de esta forma se aumentaría la disponibilidad de sangre.
- **Permitir a los usuarios reservar cita para donar sangre en aquellos casos dónde no sea urgente:** Permitir a los hospitales poner días de donación, (el icono del hospital podría salir de otro color o enviar un mensaje a los usuarios de la provincia) y que permita a los usuarios reservar una cita para donar.
- **Obtener la ubicación del usuario en tiempo real:** Siempre que el usuario lo prefiera, poder obtener la ubicación en tiempo real, de este modo el sistema será aún más eficiente en la difusión de mensajes.
- **Implementar un chatbot:** Implementar un chatbot con comandos que resuelva las dudas más comunes, (cada cuanto puedo donar, requisitos para donar, etc.)

6. Bibliografía.

- [1] Guía de realización y documentación Proyecto Fin de Carrera en la Ingeniería Técnica Informática,” Diaweb, [2000]
- [2] «Javascript» Disponible: [[JavaScript.com](https://www.javascript.com/)], [1995]
- [3] «NodeJS» Disponible: [<https://nodejs.org/es/>], [2009]
- [4] «NPM» Disponible: [<https://www.npmjs.com/>], [2010]
- [5] «Visual Studio Code» Disponible: [https://code.visualstudio.com], [2015]
- [6] «React» Disponible: [<https://es.reactjs.org/>], [2013]
- [7] «Firebase» Disponible: [https://firebase.google.com], [2014]
- [8] «Leaflet» Disponible: [<https://leafletjs.com/>], [2008]
- [9] «Colours» Disponible: [https://colors.co], [2022]
- [10] «Paletton» Disponible: [<https://paletton.com/>], [2022]
- [11] «Adobe XD» Disponible: [<https://www.adobe.com/es/products/xd.html>], [2016]
- [12] «FontAwesome» Disponible: [https://fontawesome.com], [2022]
- [13] «Diagrams.net» Disponible: [https://www.diagrams.net], [2005]
- [14] «Trello» Disponible: [https://trello.com], [2011]
- [15] «JSDoc» Disponible: [https://jsdoc.app], [2010]
- [16] M. N. Moreno García “Transparencias de Gestión de Proyectos: Planificación Temporal” [2021]
- [17] F. J. García Peñalvo y M. N. Moreno García, “Transparencias de Ingeniería del Software -Tema 4: Introducción al proceso unificado”, [2020]
- [18] F. J. García Peñalvo, A. García Holgado, and A. Vázquez Ingelmo, “Transparencias de Ingeniería del Software I - Tema UML”, [2020]
- [19] F. J. García Peñalvo y M. N. Moreno García, “Transparencias de Ingeniería del Software I, Tema 6 – UML Fundamentos de la vista de casos de uso”, [2020]
- [20] María N. Moreno García, Francisco J. García Peñalvo, “Transparencias ingeniería del software II, Tema 4 – Patrones”, [2021]
- [21] Raúl Ramírez, «Diseñando con CRAP (Contraste, Repetición, Alineación y Proximidad)» Disponible: [<https://isopixel.net/2007/08/07/disenando-con-crap/>], [7 agosto 2007]

ANEXO I:

Plan del proyecto

software

Trabajo Fin de Grado Ingeniería Informática

Septiembre 2022



VNiVERSIDAD
DE SALAMANCA

Autor:

Pedro Luis Alonso Díez

Tutores:

Luis Augusto Silva

Gabriel Villarrubia González

Juan Francisco de Paz Santana

Tabla de contenido

1. Introducción.....	1
2. Estimación del esfuerzo.....	2
2. Planificación temporal.....	6
2.1 Diagrama de Gantt.....	6
3. Bibliografía.....	9

Índice de figuras

Figura 1: Proceso unificado	1
Figura 2: Complejidad de los actores	2
Figura 3: Complejidad de los casos de uso.....	3
Figura 4: Factores de complejidad técnica.....	4
Figura 5: Factores de complejidad del entorno	5
Figura 6: Resumen de estimación	6
Figura 7: Diagrama de Gantt 1	7
Figura 8: Diagrama de Gantt 2	9

1. Introducción.

En este anexo se realiza la planificación temporal del proyecto con el fin de recoger, organizar y distribuir todas las tareas a realizar para llegar a obtener una aplicación funcional. En este punto también se analiza la viabilidad del proyecto, obteniendo el tiempo, el esfuerzo y los costes que tendrá el proyecto.

Para el desarrollo nos basaremos en las fases de desarrollo del proceso unificado.

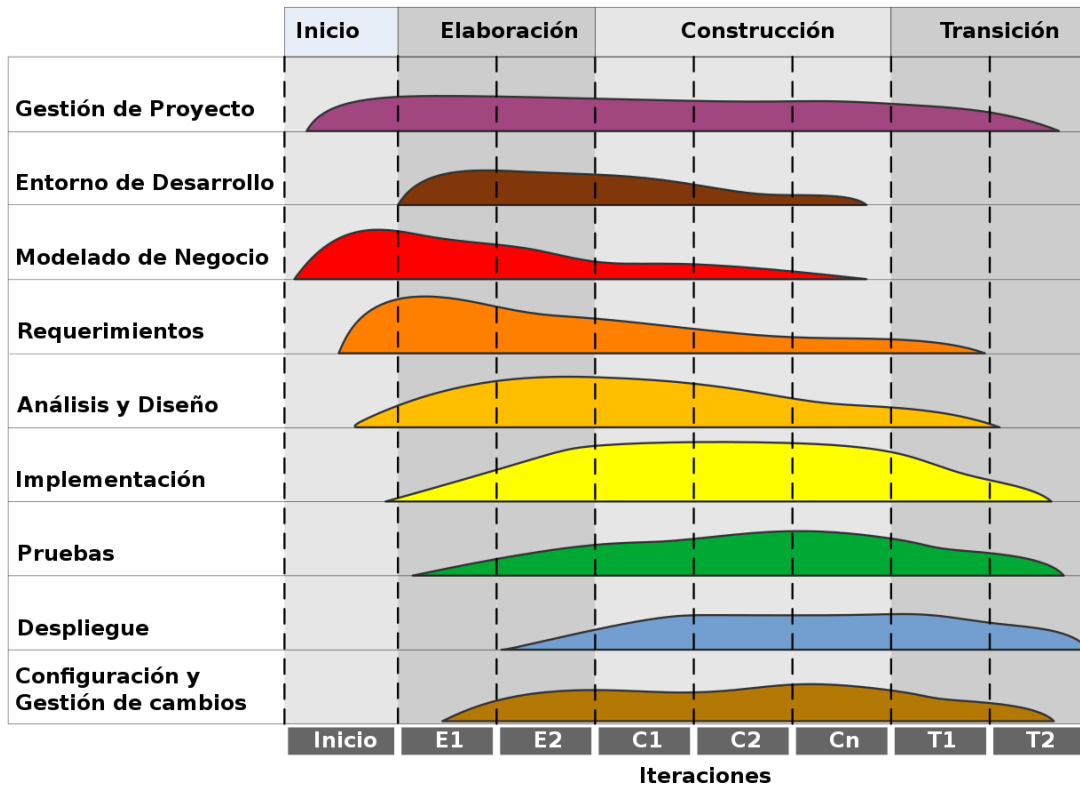


Figura 1: Proceso unificado

Y finalmente se establecerá un diagrama de Gantt para definir la estimación temporal de una forma más visual.

2. Estimación del esfuerzo.

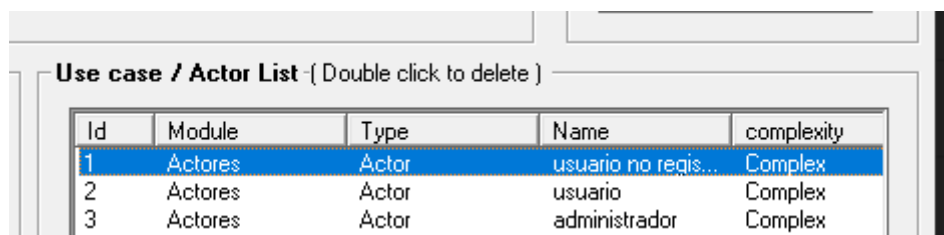
Para medir el esfuerzo necesario para el desarrollo del sistema vamos a usar la metodología *métrica de puntos de caso de uso* (UCP). Esta metodología divide el esfuerzo en secciones:

- **UUCP:** Puntos de caso de uso desajustados. Se obtiene sumando las siguientes variables:
 - **UAW:** Peso de los actores desajustado.
 - **UUCW:** Peso de los casos de uso desajustado.
- **TCF:** Factor de complejidad técnica.
- **ECF:** Factor de complejidad del entorno.

Para realizar los cálculos vamos a usar la herramienta EZEstimate [1].

Se debe establecer la complejidad de los actores, que dependen de:

- **Simple:** A través de API.
- **Media:** a través de protocolo.
- **Compleja:** A través de interfaz gráfica.

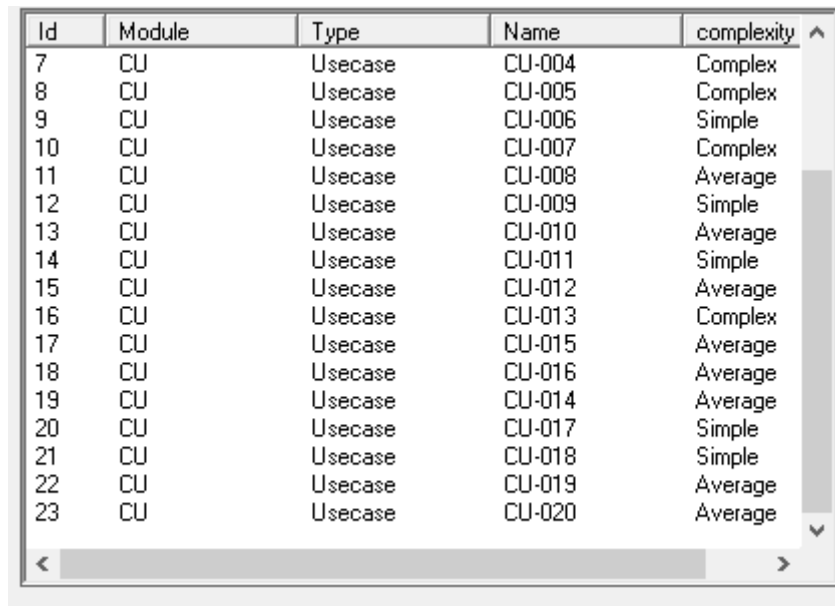


Id	Module	Type	Name	complexity
1	Actores	Actor	usuario no regis...	Complex
2	Actores	Actor	usuario	Complex
3	Actores	Actor	administrador	Complex

Figura 2: Complejidad de los actores

Se debe establecer la complejidad de los casos de uso según el número de transacciones, las transacciones son las veces que el usuario interactúa con el sistema hasta que este le responde.

- **Simple:** 3 o menos transacciones
- **Media:** entre 4 y 7 transacciones
- **Compleja:** más de 7 transacciones

A screenshot of a table with five columns: 'Id', 'Module', 'Type', 'Name', and 'complexity'. The table lists 17 use cases (IDs 7-23) from Module CU. The 'Type' for all is 'Usecase'. The 'Name' column contains IDs from CU-004 to CU-020. The 'complexity' column lists levels: Complex, Simple, and Average. The table is presented in a window with scrollbars.

Id	Module	Type	Name	complexity
7	CU	Usecase	CU-004	Complex
8	CU	Usecase	CU-005	Complex
9	CU	Usecase	CU-006	Simple
10	CU	Usecase	CU-007	Complex
11	CU	Usecase	CU-008	Average
12	CU	Usecase	CU-009	Simple
13	CU	Usecase	CU-010	Average
14	CU	Usecase	CU-011	Simple
15	CU	Usecase	CU-012	Average
16	CU	Usecase	CU-013	Complex
17	CU	Usecase	CU-015	Average
18	CU	Usecase	CU-016	Average
19	CU	Usecase	CU-014	Average
20	CU	Usecase	CU-017	Simple
21	CU	Usecase	CU-018	Simple
22	CU	Usecase	CU-019	Average
23	CU	Usecase	CU-020	Average

Figura 3: Complejidad de los casos de uso

También se debe definir los factores de complejidad técnica (TCF).

Set Technical Complexity ×

Technical complexity factors

Factor	Relevance
Distributed system	<input type="text" value="1"/>
Response / Throughput performance objectives	<input type="text" value="4"/>
End-user efficiency	<input type="text" value="3"/>
Complex internal processing	<input type="text" value="3"/>
Reusable code	<input type="text" value="4"/>
Easy to install	<input type="text" value="4"/>
Easy to use	<input type="text" value="4"/>
Portable	<input type="text" value="5"/>
Easy to change	<input type="text" value="3"/>
Concurrent	<input type="text" value="4"/>
Includes security features	<input type="text" value="3"/>
Third party access	<input type="text" value="3"/>
Special user training facilities required	<input type="text" value="1"/>

Figura 4: Factores de complejidad técnica

Por último, se debe definir los factores de complejidad del entorno (ECF).

Factor	Relevance
Familiar with Rational unified process	3
Application experience	4
Object oriented experience	4
Lead analyst capability	3
Motivation	5
Stable requirements	3
Part-time workers	0
Difficult programming language	4

Figura 5: Factores de complejidad del entorno

Con todo esto, la herramienta nos estima el número de horas que va a necesitar nuestro proyecto. En este caso han salido 1357 horas. Esto dividido entre 8 horas diarias, nos sale una estimación de 170 días que son 24 semanas.

UAW	9
UUCW	195
UUPC = UAW + UUCW	204
TFactor	48
EFactor	21
TCF = 0.6 + (.01*TFactor)	1.08
EF = 1.4 + (-0.03*EFactor)	0.77
UCP = UUCP*TCT*EF	169,6464
Total Effort@ <input type="text" value="8"/> Hrs/UCP	1357,1712

Figura 6: Resumen de estimación

2. Planificación temporal.

En este apartado definiremos y organizaremos las tareas a realizar en base a los ciclos de desarrollo del proceso unificado [1].

Este proceso cuenta con cuatro fases y con cada ciclo se obtiene una versión entregable del producto.

- **Inicio:** Se estima el alcance que tendrá el proyecto.
- **Elaboración:** Se planifica y detalla la arquitectura y el diseño del sistema.
- **Construcción:** Se construye la aplicación.
- **Transición:** Fase para incorporar mejoras y corregir errores.

Cada una de estas fases se puede descomponer en iteraciones y cuando cada una de las fases finaliza se realiza un punto de control para observar el progreso del proyecto.

Las iteraciones engloban tareas que se pueden clasificar en:

- Gestión de proyecto
- Entorno de desarrollo
- Modelo de negocio
- Requisitos
- Análisis y diseño
- Implementación
- Pruebas
- Despliegue
- Configuración y gestión de cambios

Se usará el diagrama de la Figura 1 para la definición de fases.

2.1 Diagrama de Gantt.

A través de un diagrama de Gantt podremos representar de forma gráfica todas las tareas necesarias para la correcta realización del proyecto y sus respectivos tiempos. Se planificará de la forma más eficiente posible. Planificaremos el proyecto sobre las 24 semanas que hemos calculado en las estimaciones.

DIAGRAMA DE GANTT

ESTRUCTURA DE	TÍTULO DE LA TAREA	SEMANAS
1	Fase 1 - Inicio	5
1,1	Análisis de lenguajes a usar y herramientas	1
1,2	Planificación temporal	1
1,3	Especificación de requisitos software	1
1,4	Diseño de BBDD	1
1,5	Prototipo digital	1
2	Fase 2 - Elaboración	4
2,1	Inicio de la documentación	1
2,2	Diagramas ed casos de uso	1
2,3	Refinamiento del prototipo digital	1
2,4	Configuración del framework	1
3	Fase 3 - Construcción	7
3,2	Implementación gestion de usuarios	1
3,3	Implementación gestion de hospitales	1
3,4	Implementación gestion de alertas	1
3,5	Implementación interfaz responsive	1
3,6	Implementación pequeñas funcionalidades adicionales	1
3,1	Continuación de la documentación	1
3,7	Pruebas de funcionalidad	1
4	Fase 4 - Transición	8
4,1	Revisión final requisitos	1
4,2	Revisión final diagramas	1
4,3	Revisión final funcionalidad de la aplicación	1
4,4	Arreglo de bugs	1
4,5	Elaboración final de la documentación	4

Figura 7: Diagrama de Gantt 1

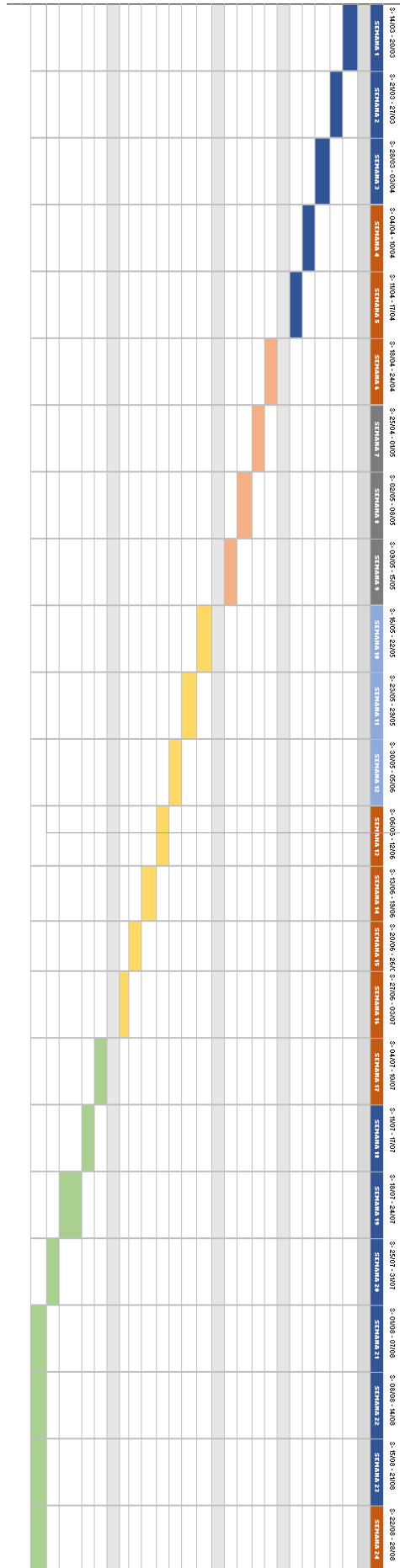


Figura 8: Diagrama de Gantt 2

3. Bibliografía

- [1] M. N. Moreno García “Transparencias de Gestión de Proyectos: Planificación Temporal” [2021]

ANEXO II:

Especificación de requisitos software

Trabajo Fin de Grado Ingeniería Informática

Septiembre 2022



**VNiVERSIDAD
D SALAMANCA**

Autor:

Pedro Luis Alonso Díez

Tutores:

Luis Augusto Silva

Gabriel Villarrubia González

Juan Francisco de Paz Santana

Tabla de contenido

1. Introducción.....	1
2. Participantes.....	2
3. Descripción del sistema actual.....	3
4. Objetivos del proyecto	4
5. Catálogo de requisitos del sistema	5
5.1. Requisitos de información	5
5.2. Requisitos no funcionales	6
6. Especificación de los requisitos.....	6
6.1. Actores	6
6.2. Especificaciones de información.....	8
6.3. Especificaciones funcionales.....	9
6.4. Especificaciones no funcionales.....	28
7. Bibliografía.	30

Índice de figuras

Figura 1: Actores	7
Figura 2: Diagrama de casos de usos paquete Gestión de usuarios	27
Figura 3: Diagrama de casos de usos paquete Gestión Administrador	27

Índice de tablas

Tabla 1: Participante Pedro Luis Alonso Díez	2
Tabla 2: Participante Luis Augusto Silva	2
Tabla 3: Participante Gabriel Villarrubia González	2
Tabla 4: Participante Juan Francisco de Paz Santana	2
Tabla 5: Objetivo Gestión de usuarios	4
Tabla 6: Objetivo Gestión de mensajes	4
Tabla 7: Objetivo Gestión de alertas	5
Tabla 8: Objetivo Gestión de hospitales	5
Tabla 9: Actor Usuario no registrado	6
Tabla 10: Actor Usuario	6
Tabla 11: Actor Administrador	7
Tabla 12: Requisito de información de usuario	8
Tabla 13: Requisito de información de hospitales	9
Tabla 14: Requisito de información de mensajes	9
Tabla 15: Caso de uso Alta Usuario	10
Tabla 16: Caso de uso Inicio de sesión	11
Tabla 17: Caso de uso Cerrar sesión	12
Tabla 18: Caso de uso Recuperar contraseña	13
Tabla 19: Caso de uso Eliminar cuenta	13
Tabla 20: Caso de uso Ver perfil	14
Tabla 21: Caso de uso Modificar perfil	15
Tabla 22: Caso de uso enviar mensaje	16
Tabla 23: Caso de uso Ver hospitales más cercanos	17
Tabla 24: Caso de uso Ver todos los hospitales registrados	18
Tabla 25: Caso de uso Ver todos los donantes registrados	18
Tabla 26: Caso de uso Enviar alerta	19
Tabla 27: Caso de uso Ver donantes por provincia	20
Tabla 28: Caso de uso modificar contraseña	21
Tabla 29: Caso de uso buscar usuario por nombre	22
Tabla 30: Caso de uso buscar hospital por nombre	23
Tabla 31: Caso de uso Confirmar correo electrónico	24
Tabla 32: Caso de uso Eliminar chat	24
Tabla 33: Caso de uso Eliminar hospital	25
Tabla 34: Caso de uso Buscar donante por grupo sanguíneo	26

Tabla 35: Requisito no funcional Usabilidad	28
Tabla 36: Requisito no funcional Eficiencia.....	28
Tabla 37: Requisito no funcional Fiabilidad	29
Tabla 38: Requisito no funcional Sistema responsivo y multiplataforma.....	29
Tabla 39: Requisito no funcional Disponibilidad.....	29
Tabla 40: Requisito no funcional Protección e integridad de datos	30

1. Introducción.

En este anexo se va a detallar la especificación de requisitos de software de la aplicación desarrollada en este proyecto.

La aplicación permitirá a usuarios registrarse como donantes de sangre, para ello tendrán que rellenar un registro con una serie de datos (Nombre, dirección, tipo de sangre, etc.). La aplicación permitirá, con una cuenta administradora, tener la visión de todos los hospitales de España y su localización, y a su vez, todos los donantes registrados. Desde la cuenta administradora, se podrá solicitar cualquier tipo de sangre en un determinado hospital y con un determinado alcance (Municipio, provincia y CCAA) y esta alerta será enviada a todos los usuarios que cumplan la condición mediante un chat interno de la aplicación.

El objetivo que se desea conseguir con este proyecto es una herramienta para el ámbito médico, fácil de usar para los usuarios y que permita mantener una información actualizada de la localización del donante, en caso de urgencias por falta de sangre.

Para los siguientes diagramas se va a utilizar el Lenguaje Unificado de Modelado (UML) con la metodología Durán y Bernárdez [1] y la herramienta Diagrams.net [2].

2. Participantes

A continuación, se van a detallar los participantes del proyecto, indicando el rol que ha desempeñado cada uno en el desarrollo de la aplicación.

Participante	Pedro Luis Alonso Díez
Organización	Universidad de Salamanca
Rol	Autor
Tipo de usuario	Desarrollador
Comentarios	No

Tabla 1: Participante Pedro Luis Alonso Díez

Participante	Luis Augusto Silva
Organización	Universidad de Salamanca
Rol	Tutor
Tipo de usuario	Revisor
Comentarios	No

Tabla 2: Participante Luis Augusto Silva

Participante	Gabriel Villarrubia González
Organización	Universidad de Salamanca
Rol	Tutor
Tipo de usuario	Revisor
Comentarios	No

Tabla 3: Participante Gabriel Villarrubia González

Participante	Juan Francisco de Paz Santana
Organización	Universidad de Salamanca
Rol	Tutor
Tipo de usuario	Revisor
Comentarios	No

Tabla 4: Participante Juan Francisco de Paz Santana

3. Descripción del sistema actual

La aplicación por desarrollar será una aplicación web, enfocada sobre todo a plataformas de escritorio, no obstante, podrá ser usada por dispositivos móviles y tabletas. La interfaz del sistema debe ser simple, debido a que va a ir destinada a todo tipo de usuarios, permitiendo así una buena eficiencia de uso sin necesidad de tener muchos conocimientos informáticos.

Para empezar la aplicación debe permitir a los usuarios registrarse, para ello necesitará introducir una serie de datos (Nombre, email, dirección, tipo de sangre, etc.). Una vez el usuario esté registrado, este será dado de alta en la base de datos y podrá modificar su ubicación.

El usuario podrá visualizar cuales son los hospitales más cercanos a su ubicación y ver sus datos (Nombre, dirección, número de teléfono, etc.)

El administrador de la aplicación podrá ver todos los donantes disponibles en cada provincia, así como la información de los hospitales de España y podrá emitir una alerta cuando se necesite un determinado tipo de sangre. Para difundir esta alerta será necesario escoger el hospital o centro de salud donde se necesite sangre, el tipo de sangre que se solicita y el alcance de la difusión (municipio, provincia o CCAA), una vez emitida la alerta, se enviará un mensaje a todos los donantes que cumplan las condiciones a través de un chat interno en la aplicación.

Por último, se permitirá al usuario cerrar la sesión, así como eliminar su cuenta en caso de que el usuario así lo quiera.

4. Objetivos del proyecto

En este apartado detallaremos los objetivos principales del sistema a desarrollar siguiendo la metodología Durán y Bernárdez.

A través de estos objetivos dividiremos el sistema en subsistemas individuales siguiendo una serie de características que los hagan lo más independientes posibles del resto.

OBJ-001	Gestión de usuarios
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Descripción	El sistema deberá implementar métodos para dar de alta o baja a usuarios, así como permitir a los usuarios visualizar y modificar su información personal. También deberá proporcionar métodos para la recuperación de la cuenta en caso de pérdida de contraseña.
Importancia	Alta
Urgencia	Alta
Estado	Activo
Estabilidad	Alta
Comentarios	No

Tabla 5: Objetivo Gestión de usuarios

OBJ-002	Gestión de mensajes
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Descripción	El sistema deberá permitir comunicarse a usuario y administrador a través de un chat interno en el sistema. Los usuarios no podrán comunicarse entre sí.
Importancia	Alta
Urgencia	Alta
Estado	Activo
Estabilidad	Alta
Comentarios	No

Tabla 6: Objetivo Gestión de mensajes

OBJ-003	Gestión de alertas
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Descripción	El sistema deberá permitir al usuario administrador mandar alertas de necesidad de sangre a los usuarios dependiendo del tipo de sangre y de la ubicación de los mismos, a través del chat interno.
Importancia	Alta
Urgencia	Alta
Estado	Activo
Estabilidad	Alta
Comentarios	No

Tabla 7: Objetivo Gestión de alertas

OBJ-004	Gestión de hospitales
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Descripción	El sistema deberá permitir ver la información de los distintos hospitales registrados en España.
Importancia	Alta
Urgencia	Alta
Estado	Activo
Estabilidad	Alta
Comentarios	No

Tabla 8: Objetivo Gestión de hospitales

5. Catálogo de requisitos del sistema

En este apartado se detallarán la lista de requisitos funcionales, no funcionales y de información de sistema [3].

5.1. Requisitos de información

Los requisitos de información describen que información se debe almacenar y gestionar en el sistema.

1. Usuario
2. Hospitales
3. Mensajes del chat

5.2. Requisitos no funcionales

1. Usabilidad
2. Eficiencia
3. Fiabilidad
4. Sistema responsivo y multiplataforma
5. Disponibilidad
6. Protección e integridad de datos

6. Especificación de los requisitos.

En este apartado utilizaremos el método de Durán y Bernárdez para detallar los requisitos descritos en el apartado anterior.

6.1. Actores

En la aplicación intervendrán distintos tipos de actores, y dependiendo de su tipo tendrán acceso a diferentes partes del sistema. Los actores se detallarán a continuación.

ACT-001	Usuario no registrado
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Descripción	Este actor representa al usuario no registrado en el sistema. Este usuario no tendrá acceso al sistema hasta que no complete el formulario de registro.
Comentarios	No

Tabla 9: Actor Usuario no registrado

ACT-002	Usuario
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Descripción	Este actor representa el usuario genérico (donante) de la aplicación que accederá a su cuenta previamente registrada.
Comentarios	No

Tabla 10: Actor Usuario

ACT-003	Administrador
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Descripción	Este actor representa el administrador de la aplicación. Podrá enviar alertas y enviar mensajes a los usuarios de la aplicación. También podrá eliminar y administrar las cuentas de los usuarios en caso necesario.
Comentarios	No

Tabla 11: Actor Administrador

El diagrama de actores quedaría:

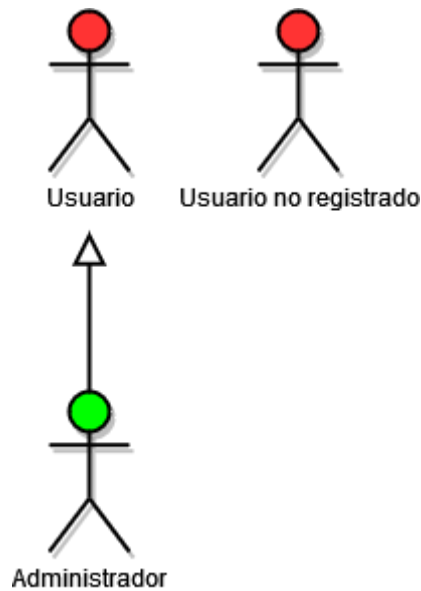


Figura 1: Actores

El actor administrador podrá hacer todo lo que haga el usuario y aparte, más funciones.

6.2. Especificaciones de información

Se describe a información nueva que se va a almacenar en el sistema.

IRQ-001	Información del usuario
Versión	1.0 (12/05/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz santana
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-001] Gestión de usuarios
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-006] Protección e integridad de datos
Descripción	Se almacenará la información del usuario necesaria para el correcto funcionamiento del sistema.
Datos específicos	<ul style="list-style-type: none"> • Nombre y apellidos • Email • Municipio • Provincia • CCAA • Tipo de sangre
Importancia	Alta
Urgencia	Media
Estado	Activo
Estabilidad	Alta
Comentarios	El usuario podrá visualizar y modificar su información a través de una página de perfil.

Tabla 12: Requisito de información de usuario

IRQ-002	Información de hospitales
Versión	1.0 (12/05/2022)
Autores	Pedro Luis Alonso Díez.
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz santana
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-004] Gestión de hospitales
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-006] Protección e integridad de datos
Descripción	Se almacenará la información de los hospitales de España, para poder ser consultado por los usuarios.
Datos específicos	<ul style="list-style-type: none"> • Nombre • Dirección • Email • Número de teléfono
Importancia	Alta

Urgencia	Media
Estado	Activo
Estabilidad	Alta
Comentarios	El usuario podrá visualizar la información de los hospitales más cercanos a su ubicación.

Tabla 13: Requisito de información de hospitales

IRQ-003	Mensajes
Versión	1.0 (12/05/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz santana
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-002] Gestión de mensajes • [OBJ-003] Gestión de alertas
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-006] Protección e integridad de datos
Descripción	Se almacenarán los mensajes enviados entre usuario y administrador.
Datos específicos	
Importancia	Alta
Urgencia	Media
Estado	Activo
Estabilidad	Alta
Comentarios	Los usuarios podrán ver el histórico de mensajes.

Tabla 14: Requisito de información de mensajes

6.3. Especificaciones funcionales

En este apartado se describirán los requisitos funcionales del sistema.

UC-001	Alta Usuario
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-001] Gestión de usuarios
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma • [NFR-006] Protección e integridad de datos
Descripción	El actor Usuario no registrado [ACT-001] desea darse de alta en el sistema.
Precondición	El usuario no debe estar ya dado de alta.

Secuencia normal	Paso	Acción
	1	El actor Usuario no registrado [ACT-001] entra en la aplicación web.
	2	El actor Usuario no registrado [ACT-001] rellena los campos de registro.
	3	El actor Usuario no registrado [ACT-001] presiona el botón de crear cuenta.
	4	El sistema procede a crear la cuenta y almacenar su información.
	5	El sistema redirige al usuario a la página de inicio de su cuenta.
Postcondición	La cuenta del usuario estará registrada en el sistema.	
Excepciones	Paso	Acción
	4	El actor Usuario no registrado [ACT-001] introduce mal algún campo necesario en el registro.
	4	El sistema no puede crear la cuenta.
	4	El sistema informa al usuario de que introduzca el campo correctamente.
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 15: Caso de uso Alta Usuario

UC-002	Inicio de sesión	
Versión	1.0 (12/08/2022)	
Autores	Pedro Luis Alonso Díez	
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana 	
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-001] Gestión de usuarios 	
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma • [NFR-006] Protección e integridad de datos 	
Descripción	El actor Usuario [ACT-002] desea iniciar sesión en su cuenta.	
Precondición	El usuario debe estar ya dado de alta.	
Secuencia normal	Paso	Acción
	1	El actor Usuario [ACT-002] entra en la aplicación web.

	2	El actor Usuario [ACT-002] rellena los campos del login y presiona el botón de iniciar sesión.
	3	El sistema redirige al usuario a su página personal.
Postcondición	Se guardará una cookie con la cuenta del usuario para que en caso de que cierre la pestaña siga la sesión durante un tiempo limitado.	
Excepciones	Paso	Acción
	3	El actor Usuario [ACT-002] introduce mal algún campo necesario en el inicio de sesión.
	4	El sistema no puede iniciar sesión con la cuenta del usuario.
	5	El sistema informa al usuario de que introduzca el campo correctamente.
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 16: Caso de uso Inicio de sesión

UC-003	Cerrar sesión	
Versión	1.0 (12/08/2022)	
Autores	Pedro Luis Alonso Díez	
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana 	
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-001] Gestión de usuarios 	
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma • [NFR-006] Protección e integridad de datos 	
Descripción	El actor Usuario [ACT-002] desea cerrar su sesión.	
Precondición	El usuario debe estar ya logeado en la aplicación.	
Secuencia normal	Paso	Acción
	1	El actor Usuario [ACT-002] desea cerrar su sesión.
	2	El actor Usuario [ACT-002] pulsa el botón para cerrar su sesión.
	3	El sistema redirige al usuario a la página de inicio de sesión.
Postcondición	La cookie que mantiene su cuenta iniciada se elimina.	
Excepciones	Ninguna	

Importancia	Alta
Urgencia	Alta
Estado	Activo
Estabilidad	Alta
Comentarios	Ninguno

Tabla 17: Caso de uso Cerrar sesión

UC-004	Recuperar contraseña	
Versión	1.0 (12/08/2022)	
Autores	Pedro Luis Alonso Díez	
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana 	
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-001] Gestión de usuarios 	
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma • [NFR-006] Protección e integridad de datos 	
Descripción	El actor Usuario [ACT-002] ha perdido su contraseña y desea recuperarla.	
Precondición		
Secuencia normal	Paso	Acción
	1	El actor Usuario [ACT-002] desea recuperar su contraseña.
	2	El actor Usuario [ACT-002] pulsa el botón para recuperar contraseña.
	3	El sistema redirige al usuario a la página de recuperación de contraseña.
	4	El actor Usuario [ACT-002] introduce el correo electrónico con el que creo la cuenta.
	5	El sistema enviará una nueva contraseña al correo electrónico del usuario.
Postcondición	La contraseña se habrá modificado y enviado al usuario.	
Excepciones	Paso	Acción
	1	El correo electrónico introducido no existe.
	2	El sistema no hará nada e informará al usuario de que el correo electrónico introducido no es válido.
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 18: Caso de uso Recuperar contraseña

UC-005	Eliminar cuenta	
Versión	1.0 (12/08/2022)	
Autores	Pedro Luis Alonso Díez	
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana 	
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-001] Gestión de usuarios 	
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma • [NFR-006] Protección e integridad de datos 	
Descripción	El actor Usuario [ACT-002] desea eliminar su cuenta	
Precondición	El usuario debe estar ya logeado en la aplicación	
Secuencia normal	Paso	Acción
	1	El actor Usuario [ACT-002] desea eliminar su cuenta.
	2	El actor Usuario [ACT-002] entra en la página de su perfil.
	3	El actor Usuario [ACT-002] presiona el botón de eliminar cuenta.
	4	El actor Usuario [ACT-002] confirma su acción.
	5	El sistema borrará la cuenta del usuario y su información asociada.
	6	El sistema redirige al usuario a la página de inicio de sesión.
Postcondición	La cuenta del usuario y su información asociada será eliminada del sistema.	
Excepciones	Paso	Acción
	4	El actor Usuario [ACT-002] no confirma su acción.
	5	El sistema no borrará la cuenta.
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 19: Caso de uso Eliminar cuenta

UC-006	Ver Perfil
Versión	1.0 (12/08/2022)

Autores	Pedro Luis Alonso Díez	
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana 	
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-001] Gestión de usuarios 	
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma • [NFR-006] Protección e integridad de datos 	
Descripción	El actor Usuario [ACT-002] desea ver su información.	
Precondición	El usuario debe estar ya logeado en la aplicación.	
Secuencia normal	Paso	Acción
	1	El actor Usuario [ACT-002] desea ver su información.
	2	El actor Usuario [ACT-002] accede a la página de perfil.
	3	El sistema muestra la información vinculada a ese usuario.
Postcondición	Ninguna	
Excepciones	Ninguna	
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 20: Caso de uso Ver perfil

UC-007	Modificar perfil
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-001] Gestión de usuarios
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma • [NFR-006] Protección e integridad de datos
Descripción	El actor Usuario [ACT-002] desea modificar información personal de su cuenta.
Precondición	El usuario debe estar ya logeado en la aplicación.

Secuencia normal	Paso	Acción
	1	El actor Usuario [ACT-002] desea modificar su información personal.
	2	El actor Usuario [ACT-002] entra en la página de su perfil.
	3	El actor Usuario [ACT-002] edita los campos que desea modificar.
	4	El actor Usuario [ACT-002] confirma su acción.
	5	El sistema modificará la información del usuario y se guardará en el sistema.
	6	El sistema mostrará la nueva información del usuario.
Postcondición	La información modificada será guardada en el sistema.	
Excepciones	Paso	Acción
	4	El actor Usuario [ACT-002] no confirma su acción.
	5	El sistema no modificará la información.
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 21: Caso de uso Modificar perfil

UC-008	Enviar mensaje	
Versión	1.0 (12/08/2022)	
Autores	Pedro Luis Alonso Díez	
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana 	
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-002] Gestión de mensajes • [OBJ-003] Gestión de alertas 	
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma • [NFR-006] Protección e integridad de datos 	
Descripción	El actor Usuario [ACT-002] desea enviar un mensaje al administrador de la página.	
Precondición	El usuario debe estar ya logeado en la aplicación.	
Secuencia normal	Paso	Acción

	1	El actor Usuario [ACT-002] o el actor Administrador [ACT-003] desea enviar un mensaje.
	2	El actor Usuario [ACT-002] o el actor Administrador [ACT-003] entra en el chat de la aplicación y selecciona al administrador o al usuario.
	3	El actor Usuario [ACT-002] o el actor Administrador [ACT-003] escribe el mensaje.
	4	El actor Usuario [ACT-002] o el actor Administrador [ACT-003] envía el mensaje.
	5	El sistema enviará el mensaje y alertará al destinatario.
Postcondición	El mensaje será guardado en el sistema como histórico del chat	
Excepciones	Ninguna	
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 22: Caso de uso enviar mensaje

UC-009	Ver hospitales más cercanos	
Versión	1.0 (12/08/2022)	
Autores	Pedro Luis Alonso Díez	
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana 	
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-004] Gestión de hospitales 	
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma • [NFR-006] Protección e integridad de datos 	
Descripción	El actor Usuario [ACT-002] desea ver los hospitales más cercanos a su ubicación.	
Precondición	El usuario debe estar ya logeado en la aplicación.	
Secuencia normal	Paso	Acción
	1	El actor Usuario [ACT-002] desea ver la información de los hospitales más cercanos.
	2	El actor Usuario [ACT-002] entra en la página de su perfil.

	3	El sistema mostrará en un mapa los hospitales más cercanos, y en una tabla su información.
Postcondición	Ninguna	
Excepciones	Ninguna	
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 23: Caso de uso Ver hospitales más cercanos

UC-010	Ver todos los hospitales registrados	
Versión	1.0 (12/08/2022)	
Autores	Pedro Luis Alonso Díez	
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana 	
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-004] Gestión de hospitales 	
Requisitos asociados	<ul style="list-style-type: none"> • [[NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma • [NFR-006] Protección e integridad de datos 	
Descripción	El actor Administrador [ACT-003] desea ver la información de los hospitales registrados.	
Precondición	El usuario logeado debe tener cuenta de administrador.	
Secuencia normal	Paso	Acción
	1	El actor Administrador [ACT-003] desea ver la información de los hospitales registrados.
	2	El actor Administrador [ACT-003] entra en la página de administración solo disponible para su usuario.
	3	El sistema mostrará en un mapa la ubicación de todos los hospitales.
	4	El actor Administrador [ACT-003] desea ver información adicional.
	5	El actor Administrador [ACT-003] hace click en el botón ver tabla.
	6	El sistema mostrará en una tabla la información de todos los hospitales, permitiendo buscar por nombre.
Postcondición	Ninguna	
Excepciones	Ninguna	
Importancia	Alta	

Urgencia	Alta
Estado	Activo
Estabilidad	Alta
Comentarios	Ninguno

Tabla 24: Caso de uso Ver todos los hospitales registrados

UC-011	Ver todos los donantes registrados	
Versión	1.0 (12/08/2022)	
Autores	Pedro Luis Alonso Díez	
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana 	
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-001] Gestión de usuarios 	
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma • [NFR-006] Protección e integridad de datos 	
Descripción	El actor Administrador [ACT-003] desea ver la información de los donantes registrados.	
Precondición	El usuario logeado debe tener cuenta de administrador.	
Secuencia normal	Paso	Acción
	1	El actor Administrador [ACT-003] desea ver la información de los donantes registrados.
	2	El actor Administrador [ACT-003] entra en la página de administración solo disponible para su usuario.
	3	El sistema mostrará en una tabla todos los donantes registrados y la información útil.
Postcondición	Ninguna	
Excepciones	No hay ningún donante dado de alta	
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 25: Caso de uso Ver todos los donantes registrados

UC-012	Enviar alerta	
Versión	1.0 (12/08/2022)	
Autores	Pedro Luis Alonso Díez	
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana 	
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-002] Gestión de mensajes 	

	<ul style="list-style-type: none"> • [OBJ-003] Gestión de alertas 														
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma • [NFR-006] Protección e integridad de datos 														
Descripción	El actor Administrador [ACT-003] desea enviar una alerta solicitando un determinado tipo de sangre.														
Precondición	El usuario logeado debe tener cuenta de administrador.														
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El actor Administrador [ACT-003] desea enviar una alerta a los actores Usuarios [ACT-002].</td> </tr> <tr> <td>2</td> <td>El actor Administrador [ACT-003] entra en la página de administración solo disponible para su usuario.</td> </tr> <tr> <td>3</td> <td>El actor Administrador [ACT-003] seleccionará el hospital donde se necesita sangre, el tipo de sangre que necesita y el alcance de difusión (Municipio, provincia o CCAA).</td> </tr> <tr> <td>4</td> <td>El sistema enviará una alerta indicando los datos del hospital y el tipo de sangre que se necesita a todos los actores Usuario [ACT-002] que cumplan con las condiciones seleccionadas.</td> </tr> <tr> <td>5</td> <td>El sistema avisará a los actores Usuario [ACT-002] de que tienen un nuevo mensaje.</td> </tr> <tr> <td>6</td> <td>El sistema mostrará al actor Administrador [ACT-003] los datos de los actores Usuario [ACT-002] a los que se les ha enviado el mensaje.</td> </tr> </tbody> </table>	Paso	Acción	1	El actor Administrador [ACT-003] desea enviar una alerta a los actores Usuarios [ACT-002].	2	El actor Administrador [ACT-003] entra en la página de administración solo disponible para su usuario.	3	El actor Administrador [ACT-003] seleccionará el hospital donde se necesita sangre, el tipo de sangre que necesita y el alcance de difusión (Municipio, provincia o CCAA).	4	El sistema enviará una alerta indicando los datos del hospital y el tipo de sangre que se necesita a todos los actores Usuario [ACT-002] que cumplan con las condiciones seleccionadas.	5	El sistema avisará a los actores Usuario [ACT-002] de que tienen un nuevo mensaje.	6	El sistema mostrará al actor Administrador [ACT-003] los datos de los actores Usuario [ACT-002] a los que se les ha enviado el mensaje.
Paso	Acción														
1	El actor Administrador [ACT-003] desea enviar una alerta a los actores Usuarios [ACT-002].														
2	El actor Administrador [ACT-003] entra en la página de administración solo disponible para su usuario.														
3	El actor Administrador [ACT-003] seleccionará el hospital donde se necesita sangre, el tipo de sangre que necesita y el alcance de difusión (Municipio, provincia o CCAA).														
4	El sistema enviará una alerta indicando los datos del hospital y el tipo de sangre que se necesita a todos los actores Usuario [ACT-002] que cumplan con las condiciones seleccionadas.														
5	El sistema avisará a los actores Usuario [ACT-002] de que tienen un nuevo mensaje.														
6	El sistema mostrará al actor Administrador [ACT-003] los datos de los actores Usuario [ACT-002] a los que se les ha enviado el mensaje.														
Postcondición	Ninguna														
Excepciones	No hay ningún donante que cumpla los requisitos														
Importancia	Alta														
Urgencia	Alta														
Estado	Activo														
Estabilidad	Alta														
Comentarios	Ninguno														

Tabla 26: Caso de uso Enviar alerta

UC-013	Ver donantes por provincia
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva

	<ul style="list-style-type: none"> • Gabriel Villarrubia González • Juan Francisco de Paz Santana 										
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-001] Gestión de usuarios 										
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma • [NFR-006] Protección e integridad de datos 										
Descripción	El actor Administrador [ACT-003] desea ver cuántos donantes hay en cada provincia.										
Precondición	El usuario logeado debe tener cuenta de administrador.										
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El actor Administrador [ACT-003] desea ver cuántos donantes [ACT-002] hay en cada provincia.</td> </tr> <tr> <td>2</td> <td>El actor Administrador [ACT-003] entra en la página de administración solo disponible para su usuario.</td> </tr> <tr> <td>3</td> <td>El actor Administrador [ACT-003] seleccionará el botón mostrar provincias.</td> </tr> <tr> <td>4</td> <td>El sistema mostrará un pop up cuando se seleccione una provincia en el mapa mostrando el número de donantes en esa provincia.</td> </tr> </tbody> </table>	Paso	Acción	1	El actor Administrador [ACT-003] desea ver cuántos donantes [ACT-002] hay en cada provincia.	2	El actor Administrador [ACT-003] entra en la página de administración solo disponible para su usuario.	3	El actor Administrador [ACT-003] seleccionará el botón mostrar provincias.	4	El sistema mostrará un pop up cuando se seleccione una provincia en el mapa mostrando el número de donantes en esa provincia.
Paso	Acción										
1	El actor Administrador [ACT-003] desea ver cuántos donantes [ACT-002] hay en cada provincia.										
2	El actor Administrador [ACT-003] entra en la página de administración solo disponible para su usuario.										
3	El actor Administrador [ACT-003] seleccionará el botón mostrar provincias.										
4	El sistema mostrará un pop up cuando se seleccione una provincia en el mapa mostrando el número de donantes en esa provincia.										
Postcondición	Ninguna										
Excepciones	Ninguna										
Importancia	Alta										
Urgencia	Alta										
Estado	Activo										
Estabilidad	Alta										
Comentarios	Ninguno										

Tabla 27: Caso de uso Ver donantes por provincia

UC-014	Modificar contraseña
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-001] Gestión de usuarios
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma

	<ul style="list-style-type: none"> [NFR-006] Protección e integridad de datos 	
Descripción	El actor Usuario [ACT-002] desea ver modificar su contraseña	
Precondición	El usuario debe estar logeado.	
Secuencia normal	Paso	Acción
	1	El actor Usuario [ACT-002] desea modificar su contraseña
	2	El actor Usuario [ACT-002] accede a la página de perfil
	3	El actor Usuario [ACT-002] seleccionará el botón Seguridad ubicado en su perfil
	4	El actor Usuario [ACT-002] deberá introducir la antigua contraseña, su nueva contraseña y confirmar su nueva contraseña
	5	El sistema valida las contraseñas y si son correctas cambia la contraseña
Postcondición	Ninguna	
Excepciones	Paso	Acción
	1	El actor Usuario [ACT-002] introduce la contraseña incorrecta o las nuevas contraseñas no coinciden
	2	El actor sistema no cambia la contraseña y muestra al actor Usuario [ACT-002] el error
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 28: Caso de uso modificar contraseña

UC-015	Buscar donante por nombre
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> Luis Augusto Silva Gabriel Villarrubia González Juan Francisco de Paz Santana
Objetivos asociados	<ul style="list-style-type: none"> [OBJ-001] Gestión de usuarios
Requisitos asociados	<ul style="list-style-type: none"> [NFR-001] Usabilidad [NFR-002] Eficiencia [NFR-003] Fiabilidad [NFR-004] Sistema responsivo y multiplataforma [NFR-006] Protección e integridad de datos

Descripción	El actor Administrador [ACT-003] desea buscar a un actor Usuario [ACT-002] por nombre	
Precondición	El usuario logeado debe tener cuenta de administrador.	
Secuencia normal	Paso	Acción
	1	El actor Administrador [ACT-003] desea buscar un usuario por nombre
	2	El actor Administrador [ACT-003] se encuentra en la página del chat
	3	El actor Administrador [ACT-003] introduce el nombre o apellido que desea buscar en la barra de búsqueda
	4	El sistema muestra las coincidencias
Postcondición	Ninguna	
Excepciones	Ninguna	
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 29: Caso de uso buscar usuario por nombre

UC-016	Buscar hospital por nombre	
Versión	1.0 (12/08/2022)	
Autores	Pedro Luis Alonso Díez	
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana 	
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-004] Gestión de hospitales 	
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-004] Sistema responsivo y multiplataforma • [NFR-006] Protección e integridad de datos 	
Descripción	El actor Administrador [ACT-003] desea buscar un hospital por nombre	
Precondición	El usuario logeado debe tener cuenta de administrador.	
Secuencia normal	Paso	Acción
	1	El actor Administrador [ACT-003] desea buscar un hospital por nombre
	2	El actor Administrador [ACT-003] se encuentra en la página de control

	3	El actor Administrador [ACT-003] introduce el nombre del hospital que desea buscar en la barra de búsqueda
	4	El sistema muestra las coincidencias
Postcondición	Ninguna	
Excepciones	Ninguna	
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 30: Caso de uso buscar hospital por nombre

UC-017	Confirmar correo electrónico	
Versión	1.0 (12/08/2022)	
Autores	Pedro Luis Alonso Díez	
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana 	
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-001] Gestión de usuarios 	
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-006] Protección e integridad de datos 	
Descripción	El actor Usuario [ACT-002] deberá confirmar su correo electrónico	
Precondición	El usuario debe estar registrado.	
Secuencia normal	Paso	Acción
	1	El actor Usuario [ACT-002] solicita registrarse
	2	El actor Usuario [ACT-002] accede al enlace enviado al correo con el que se ha registrado
	3	El sistema verifica el correo electrónico y confirma el correo electrónico mostrando un mensaje de que todo ha ido bien
Postcondición	El usuario queda verificado.	
Excepciones	Ninguna	
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	

Comentarios	Ninguno
--------------------	---------

Tabla 31: Caso de uso Confirmar correo electrónico

UC-018	Eliminar chat	
Versión	1.0 (12/08/2022)	
Autores	Pedro Luis Alonso Díez	
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana 	
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-002] Gestión de mensajes 	
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad • [NFR-006] Protección e integridad de datos 	
Descripción	El actor Usuario [ACT-002] o el actor Administrador [ACT-003] desea eliminar un chat	
Precondición	El usuario debe estar registrado.	
Secuencia normal	Paso	Acción
	1	El actor Usuario [ACT-002] o administrador [ACT-003] selecciona el chat que quiere borrar
	2	El actor Usuario [ACT-002] o administrador [ACT-003] presiona el botón eliminar chat
	3	El sistema borrará el chat y todos sus mensajes asociados del sistema
Postcondición	Ninguna	
Excepciones	Ninguna	
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 32: Caso de uso Eliminar chat

UC-019	Eliminar hospital	
Versión	1.0 (12/08/2022)	
Autores	Pedro Luis Alonso Díez	
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana 	
Objetivos asociados	<ul style="list-style-type: none"> • [OBJ-004] Gestión de hospitales 	
Requisitos asociados	<ul style="list-style-type: none"> • [NFR-001] Usabilidad • [NFR-002] Eficiencia • [NFR-003] Fiabilidad 	

	<ul style="list-style-type: none"> [NFR-006] Protección e integridad de datos 	
Descripción	El actor Administrador [ACT-003] desea eliminar un hospital del sistema	
Precondición	El usuario debe estar registrado debe tener cuenta de administrador	
Secuencia normal	Paso	Acción
	1	El administrador [ACT-003] debe abrir la vista de control
	2	El actor administrador [ACT-003] realizará el UC-010 – “Ver todos los hospitales registrados”
	3	El actor administrador [ACT-003] seleccionará el hospital que desea eliminar
	4	El actor administrador [ACT-003] seleccionará el hospital que desea eliminar
	5	El actor administrador [ACT-003] pulsará el botón eliminar
	6	El sistema eliminará el hospital de su base de datos.
Postcondición	Ninguna	
Excepciones	Ninguna	
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 33: Caso de uso Eliminar hospital

UC-020	Buscar donantes por grupo sanguíneo
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> Luis Augusto Silva Gabriel Villarrubia González Juan Francisco de Paz Santana
Objetivos asociados	<ul style="list-style-type: none"> [OBJ-001] Gestión de usuarios
Requisitos asociados	<ul style="list-style-type: none"> [NFR-001] Usabilidad [NFR-002] Eficiencia [NFR-003] Fiabilidad [NFR-004] Sistema responsivo y multiplataforma [NFR-006] Protección e integridad de datos
Descripción	El actor Administrador [ACT-003] desea buscar los donantes registrados con un determinado tipo de sangre
Precondición	El usuario debe estar registrado debe tener cuenta de administrador

Secuencia normal	Paso	Acción
	1	El administrador [ACT-003] debe abrir la vista de control
	2	El actor administrador [ACT-003] realizará el UC-011 – “Ver todos los donantes registrados”
	3	El actor administrador [ACT-003] pulsará sobre la columna tipo de sangre de la tabla
	4	El sistema proporcionará un dialogo con los tipos de sangre registrados
	5	El actor administrador [ACT-003] introducirá los tipos de sangre que desea buscar
	6	El sistema mostrará los donantes con ese tipo de sangre
Postcondición	Ninguna	
Excepciones	Ninguna	
Importancia	Alta	
Urgencia	Alta	
Estado	Activo	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 34: Caso de uso Buscar donante por grupo sanguíneo

A continuación, dividiremos los casos de uso anteriormente detallados en paquetes, con el fin de organizar más la aplicación. Los paquetes serán los siguientes.

Los casos de uso UC-001, UC-002, UC-003, UC-004, UC-005, UC-006, UC-007, UC-008, UC-009, UC-014 y UC-017 formarán parte del paquete “*Gestión de usuarios*”

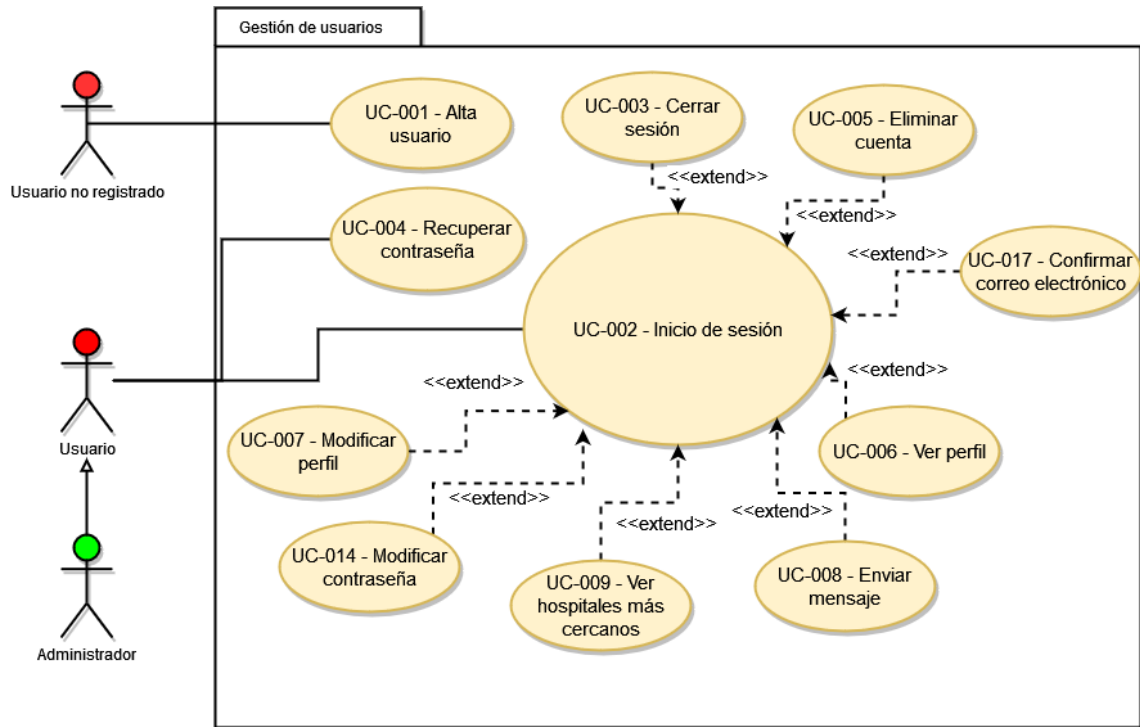


Figura 2: Diagrama de casos de usos paquete Gestión de usuarios

Los casos de uso UC-010, UC-011, UC-012, UC-013, UC-015, UC-016, UC-018, UC-019, UC-20 formarán parte del paquete “Gestión Administrador”

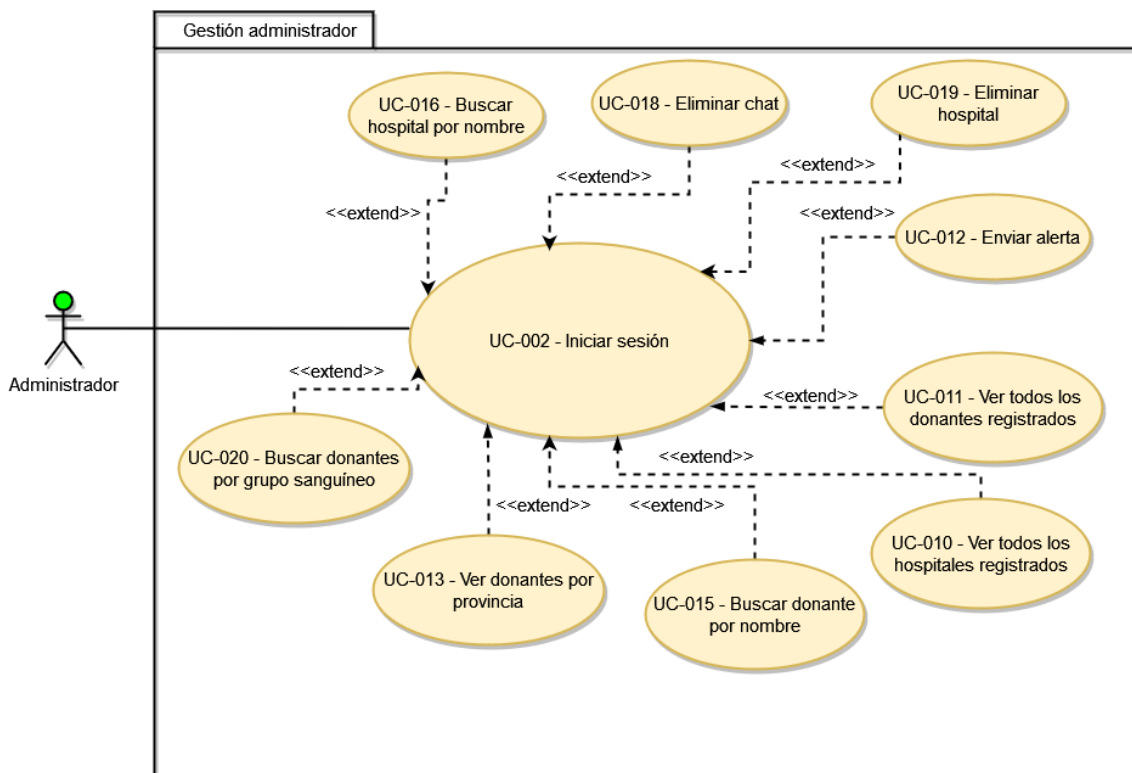


Figura 3: Diagrama de casos de usos paquete Gestión Administrador

6.4. Especificaciones no funcionales

A continuación, se especificarán en tablas los requisitos no funcionales que se describieron anteriormente.

NFR-001	Usabilidad
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Dependencias	Todo el sistema
Descripción	El sistema debe de ser intuitivo y sencillo de utilizar para todos los usuarios siguiendo el modelo de reglas heurísticas.
Importancia	Alta
Urgencia	Alta
Estado	Activo
Estabilidad	Alta
Comentarios	Ninguno

Tabla 35: Requisito no funcional Usabilidad

NFR-002	Eficiencia
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Dependencias	Todo el sistema
Descripción	El sistema debe responder de manera rápida a las acciones del usuario.
Importancia	Alta
Urgencia	Alta
Estado	Activo
Estabilidad	Alta
Comentarios	Ninguno

Tabla 36: Requisito no funcional Eficiencia

NFR-003	Fiabilidad
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Dependencias	Todo el sistema
Descripción	El sistema debe asegurar su correcto funcionamiento. Avisando de los posibles fallos que surjan.

Importancia	Alta
Urgencia	Alta
Estado	Activo
Estabilidad	Alta
Comentarios	Ninguno

Tabla 37: Requisito no funcional Fiabilidad

NFR-004	Sistema responsivo y multiplataforma
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Dependencias	Todo el sistema
Descripción	El sistema debe ejecutarse en cualquier plataforma y debe adaptar su interfaz al tamaño del dispositivo por el que se acceda.
Importancia	Alta
Urgencia	Alta
Estado	Activo
Estabilidad	Alta
Comentarios	Ninguno

Tabla 38: Requisito no funcional Sistema responsivo y multiplataforma

NFR-005	Disponibilidad
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Dependencias	Todo el sistema
Descripción	El sistema debe estar operativo la mayor parte del tiempo posible.
Importancia	Alta
Urgencia	Alta
Estado	Activo
Estabilidad	Alta
Comentarios	Ninguno

Tabla 39: Requisito no funcional Disponibilidad

NFR-006	Protección e integridad de datos
Versión	1.0 (12/08/2022)
Autores	Pedro Luis Alonso Díez
Fuentes	<ul style="list-style-type: none"> • Luis Augusto Silva

	<ul style="list-style-type: none"> • Gabriel Villarrubia González • Juan Francisco de Paz Santana
Dependencias	Todo el sistema
Descripción	El sistema debe proporcionar privacidad sobre los datos guardados en él, así mismo, debe ofrecer una integridad sobre ellos, cumpliendo con los requisitos sobre la ley orgánica de protección de datos en sistemas de información.
Importancia	Alta
Urgencia	Alta
Estado	Activo
Estabilidad	Alta
Comentarios	Ninguno

Tabla 40: Requisito no funcional Protección e integridad de datos

7. Bibliografía.

- [1] F. J. García Peñalvo y M. N. Moreno García, “Transparencias de Ingeniería del Software I, Tema 6 – UML Fundamentos de la vista de casos de uso” [2020]
- [2] «Diagrams.net» Enlace: <https://www.diagrams.net> [2005]
- [3] F. J. García Peñalvo, A. García Holgado, and A. Vázquez Ingelmo, “Transparencias de Ingeniería del Software I, Tema 5 - Ingeniería de Requisitos” [2020]

ANEXO III:

Análisis de requisitos

Trabajo Fin de Grado Ingeniería Informática

Septiembre 2022



**VNiVERSIDAD
D SALAMANCA**

Autor:

Pedro Luis Alonso Díez

Tutores:

Luis Augusto Silva

Gabriel Villarrubia González

Juan Francisco de Paz Santana

Tabla de contenido

1. Introducción.....	1
2. Modelo de dominio.....	2
3. Diagramas de secuencia.....	3
4. Descripción de la arquitectura.....	15
5. Bibliografía.....	17

Índice de figuras

Figura 1: Modelo de dominio	2
Figura 2: Diagrama de secuencia Alta usuario	4
Figura 3: Diagrama de secuencia Iniciar sesión.....	5
Figura 4: Diagrama de secuencia Cerrar sesión	6
Figura 5: Diagrama de secuencia Recuperar contraseña	6
Figura 6: Diagrama de secuencia Eliminar cuenta	7
Figura 7: Diagrama de secuencia Ver perfil.....	7
Figura 8: Diagrama de secuencia Modificar perfil.....	8
Figura 9: Diagrama de secuencia Enviar mensaje.....	9
Figura 10: Diagrama de secuencia Ver hospitales cercanos	9
Figura 11: Diagrama de secuencia Ver todos los hospitales registrados.....	10
Figura 12: Diagrama de secuencia Ver todos los donantes registrados	10
Figura 13: Diagrama de secuencia Enviar alerta	11
Figura 14: Diagrama de secuencia Ver donantes por provincia.....	11
Figura 15: Diagrama de secuencia Modificar contraseña	12
Figura 16: Diagrama de secuencia Buscar donante por nombre	13
Figura 17: Diagrama de secuencia Buscar hospital por nombre.....	13
Figura 18: Diagrama de secuencia Confirmar correo electrónico.....	14
Figura 19: Diagrama de secuencia Eliminar chat	15
Figura 20: Diagrama de secuencia Eliminar hospital	15
Figura 21: Diagrama de secuencia Buscar donantes por grupo sanguíneo	15
Figura 22: Descripción de la Arquitectura del Modelo de Análisis Gestión de usuarios	16
Figura 23: Descripción de la Arquitectura del Modelo de Análisis Gestión Administrador.....	16

1. Introducción.

En este anexo trataremos la documentación de los procesos de análisis, refinamiento y estructuración para cada uno de los requisitos que posee el sistema para poder obtener una comprensión del sistema a bajo nivel. Para este proceso se usará el lenguaje unificado de modelado UML [1]. Para los diagramas diseñados se usará la herramienta Diagrams.net [2].

2. Modelo de dominio.

El modelo de dominio puede ser tomado como un punto de partida para el diseño del sistema. Permite identificar las relaciones entre las entidades que forman el dominio del problema, y además identificar sus atributos.

Esto nos sirve para tener una visión estructural del dominio, es decir, obtener las necesidades de almacenamiento y la gestión de la información que va a necesitar el sistema de una forma más esquemática y clara.

A continuación, se detallarán las clases y objetos que utilizaremos en el sistema.

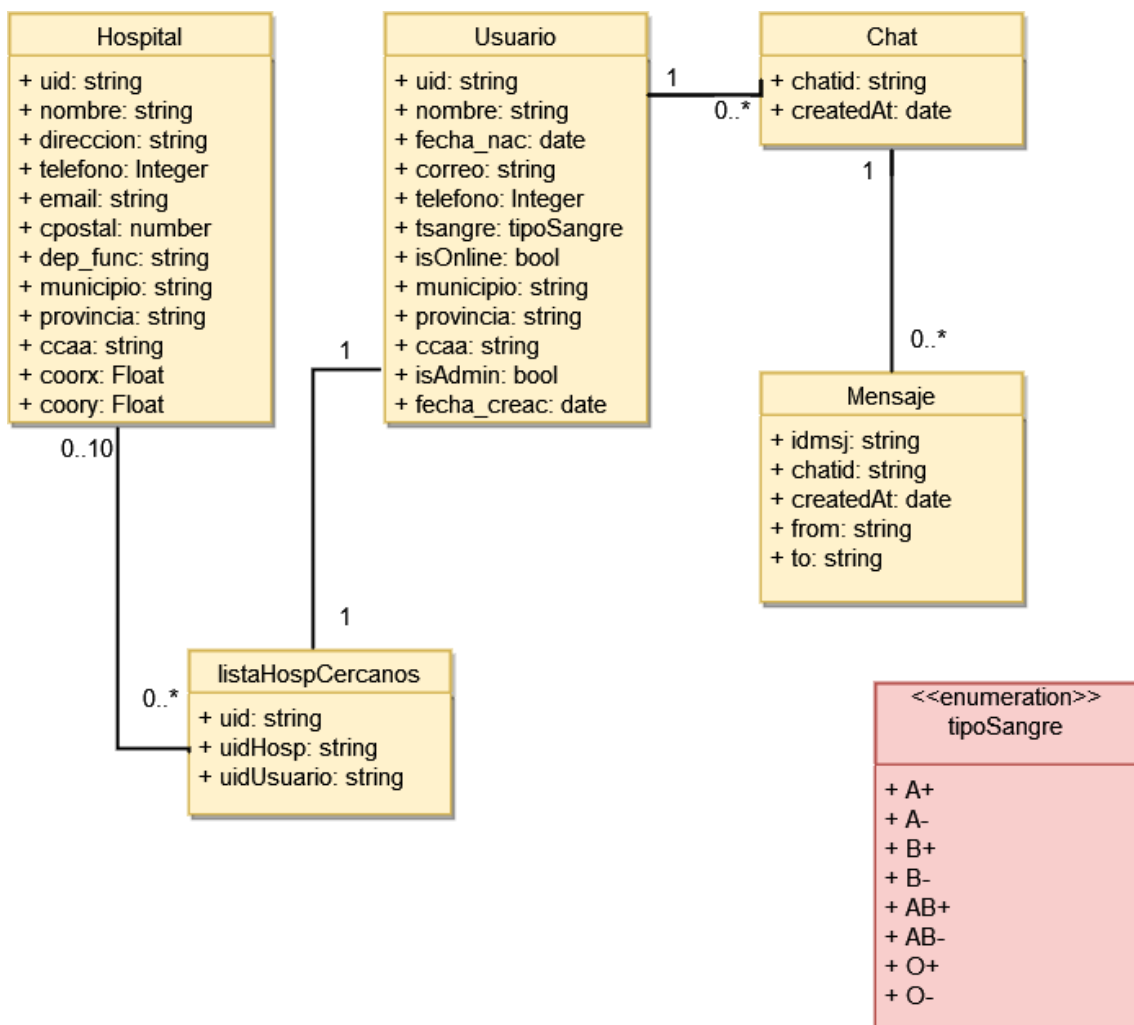


Figura 1: Modelo de dominio

A continuación, se explican cada una de las clases detalladas en el modelo de dominio, y posteriormente serán representadas mediante diagramas de clases.

- **Hospital:** Almacena información sobre un hospital.
- **Usuario:** Almacena información sobre un usuario y representa una cuenta.
- **Chat:** Representa el chat creado entre dos usuarios.
- **Mensaje:** Representa el contenido a enviar en un chat entre dos usuarios.
- **listaHospCercanos:** Representa la lista de los hospitales más cercanos a cada usuario.

Enumeraciones:

- **tipoSangre:** Recoge todos los posibles tipos de sangre existentes.

3. Diagramas de secuencia.

Se va a realizar un diagrama de secuencia para cada caso de uso descrito en el Anexo II. Estos casos de uso representan el modelo de análisis.

UC-001 – Alta usuario.

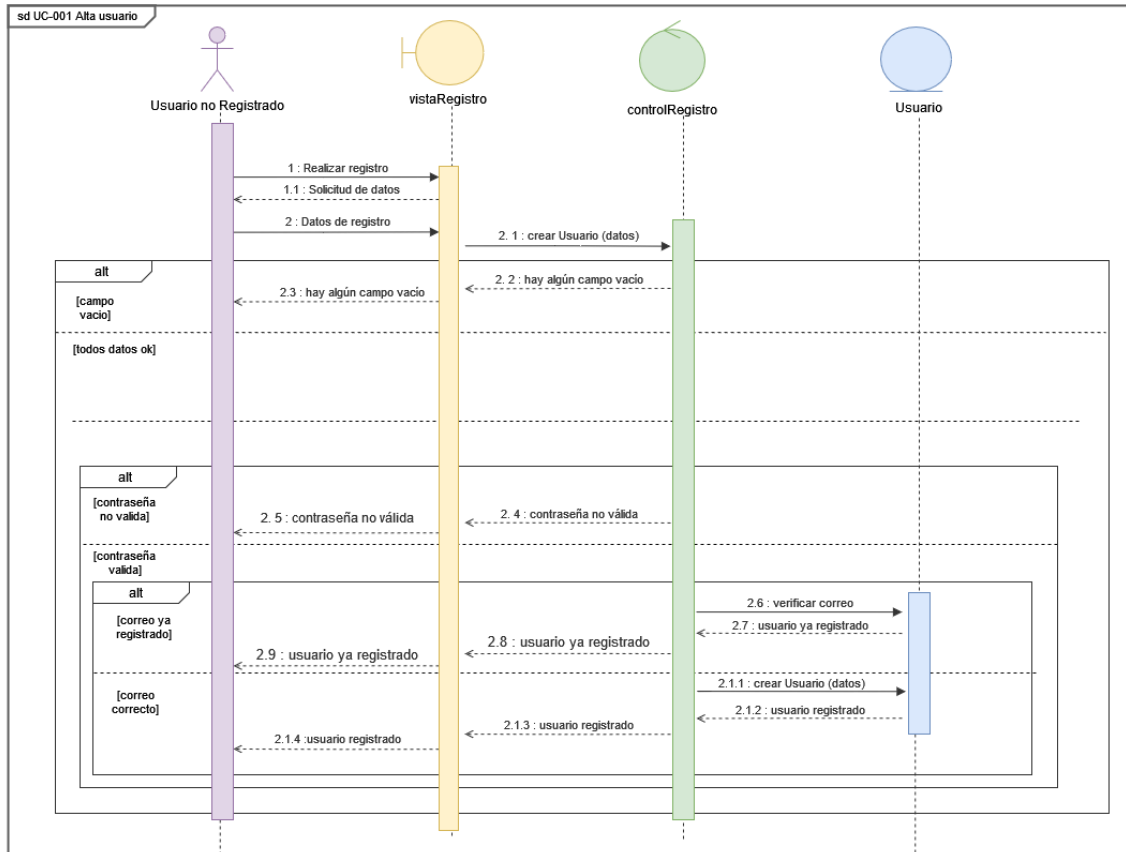


Figura 2: Diagrama de secuencia Alta usuario

UC-002 – Inicio de sesión.

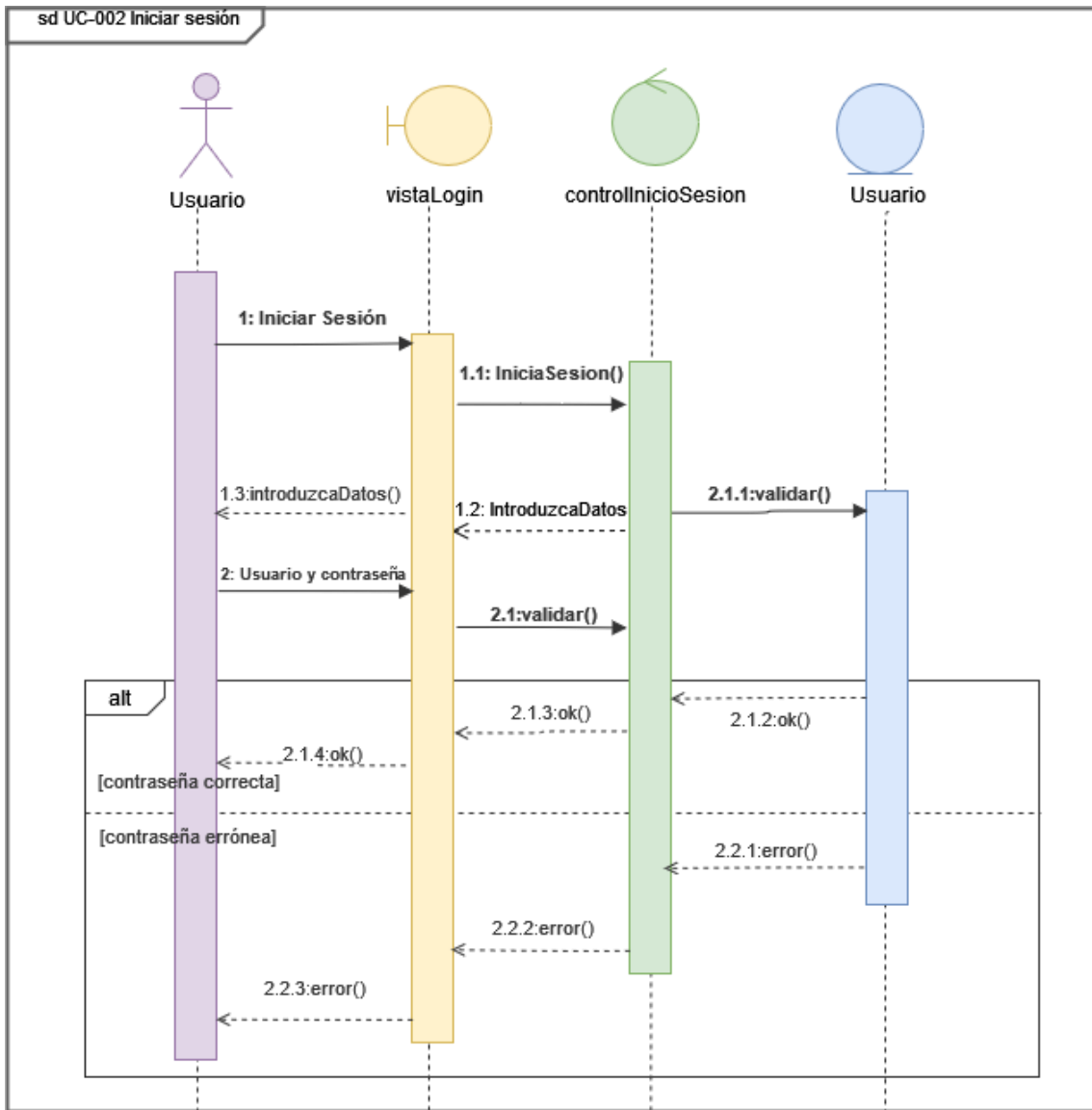


Figura 3: Diagrama de secuencia Iniciar sesión

UC-003 – Cerrar sesión.

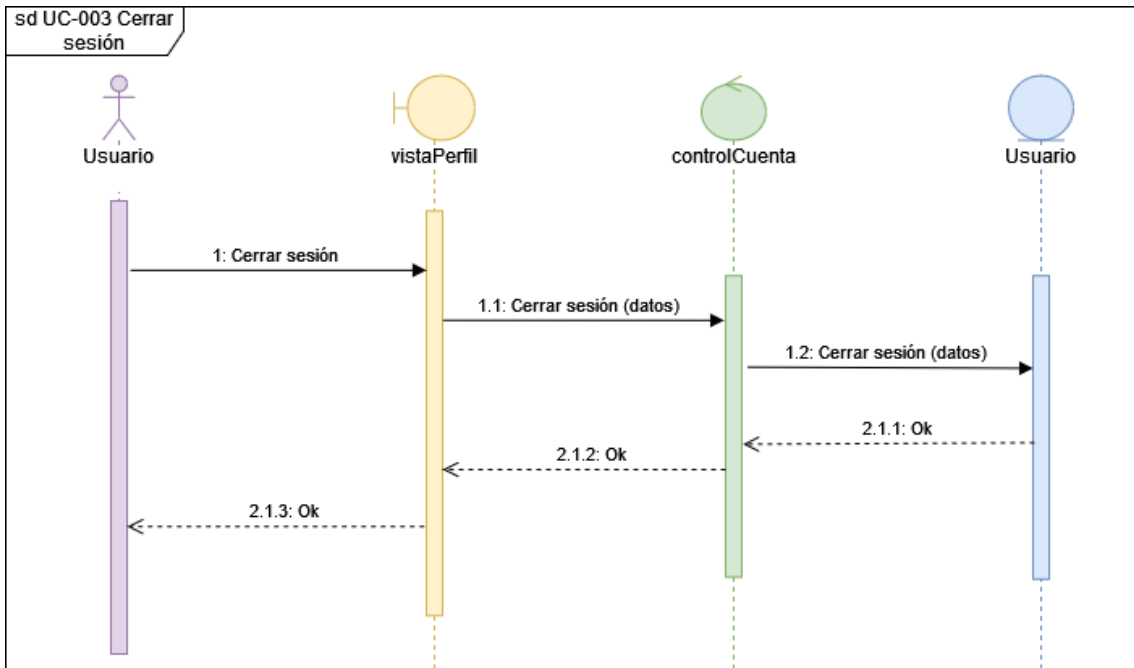


Figura 4: Diagrama de secuencia Cerrar sesión

UC-004 – Recuperar contraseña.

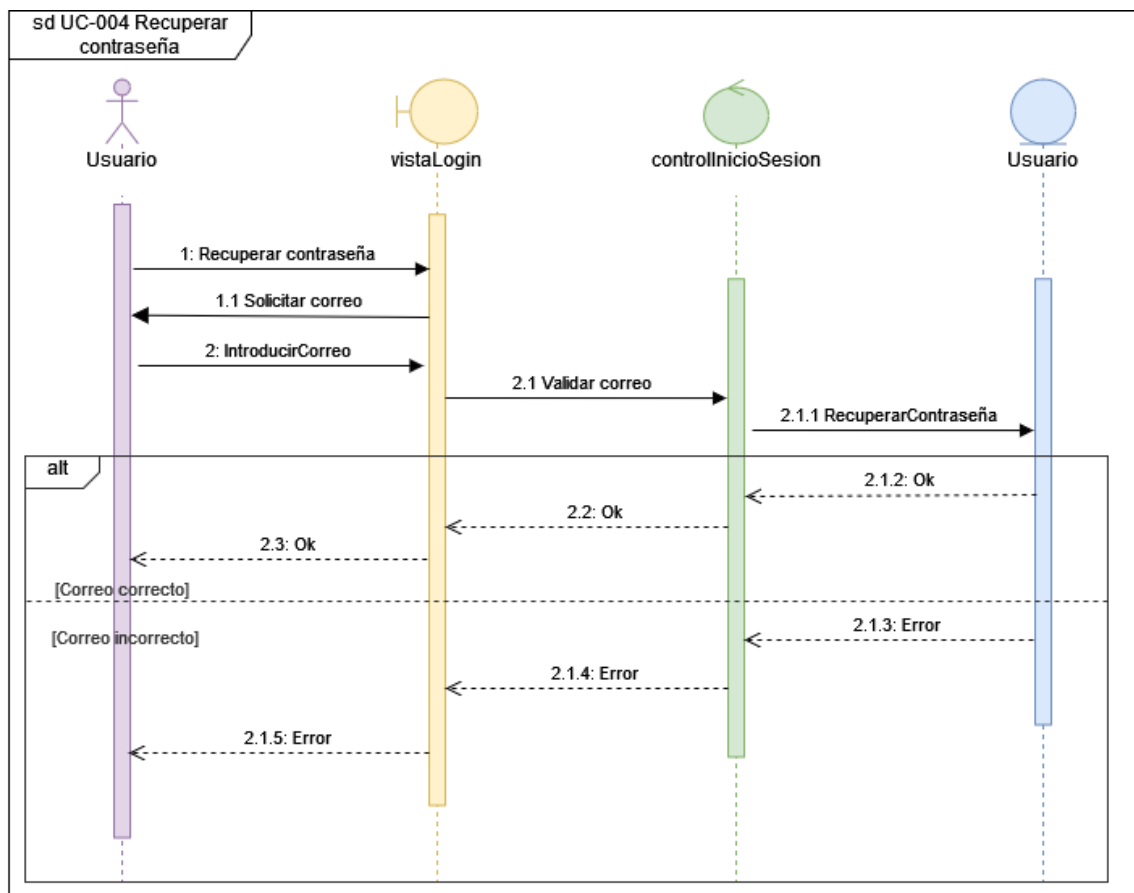


Figura 5: Diagrama de secuencia Recuperar contraseña

UC-005 – Eliminar cuenta.

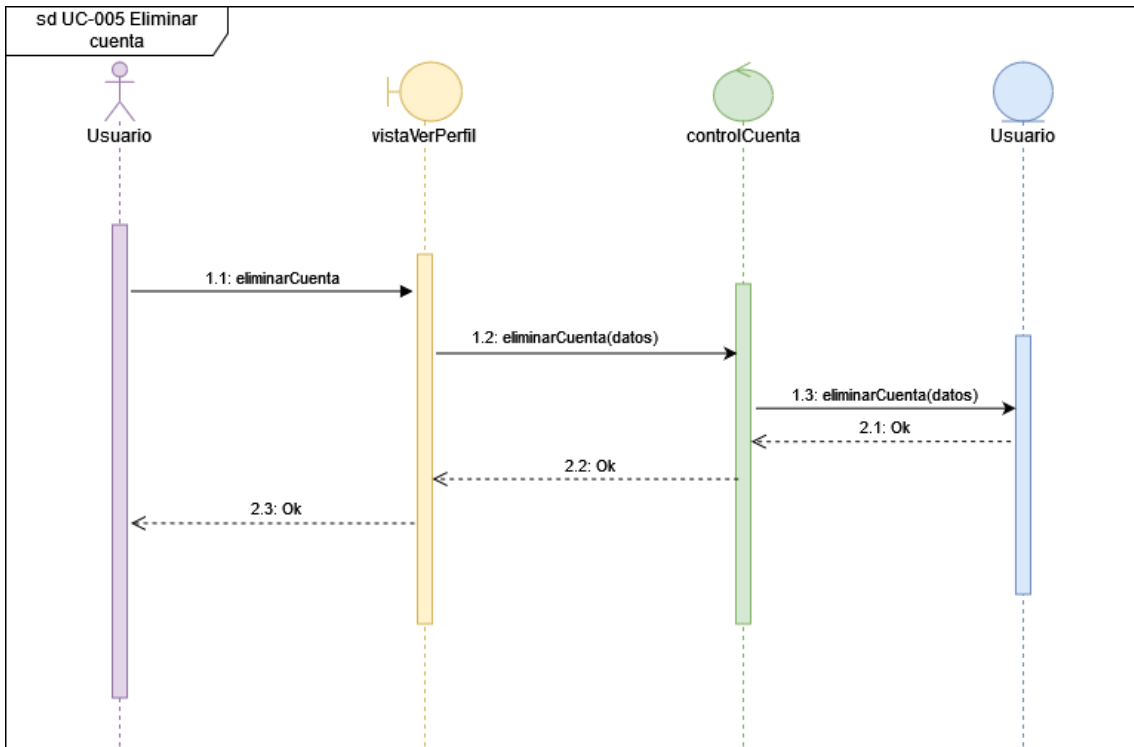


Figura 6: Diagrama de secuencia Eliminar cuenta

UC-006 – Ver perfil.

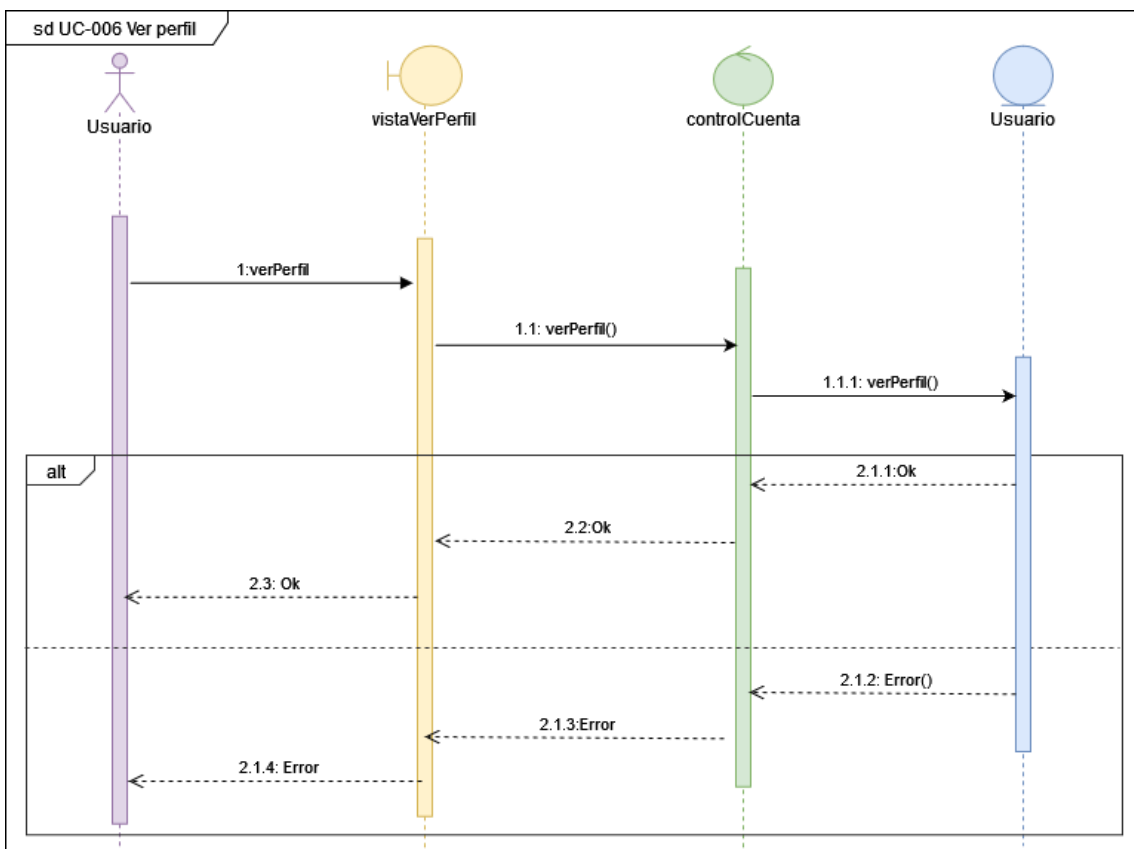


Figura 7: Diagrama de secuencia Ver perfil

UC-007 – Modificar perfil.

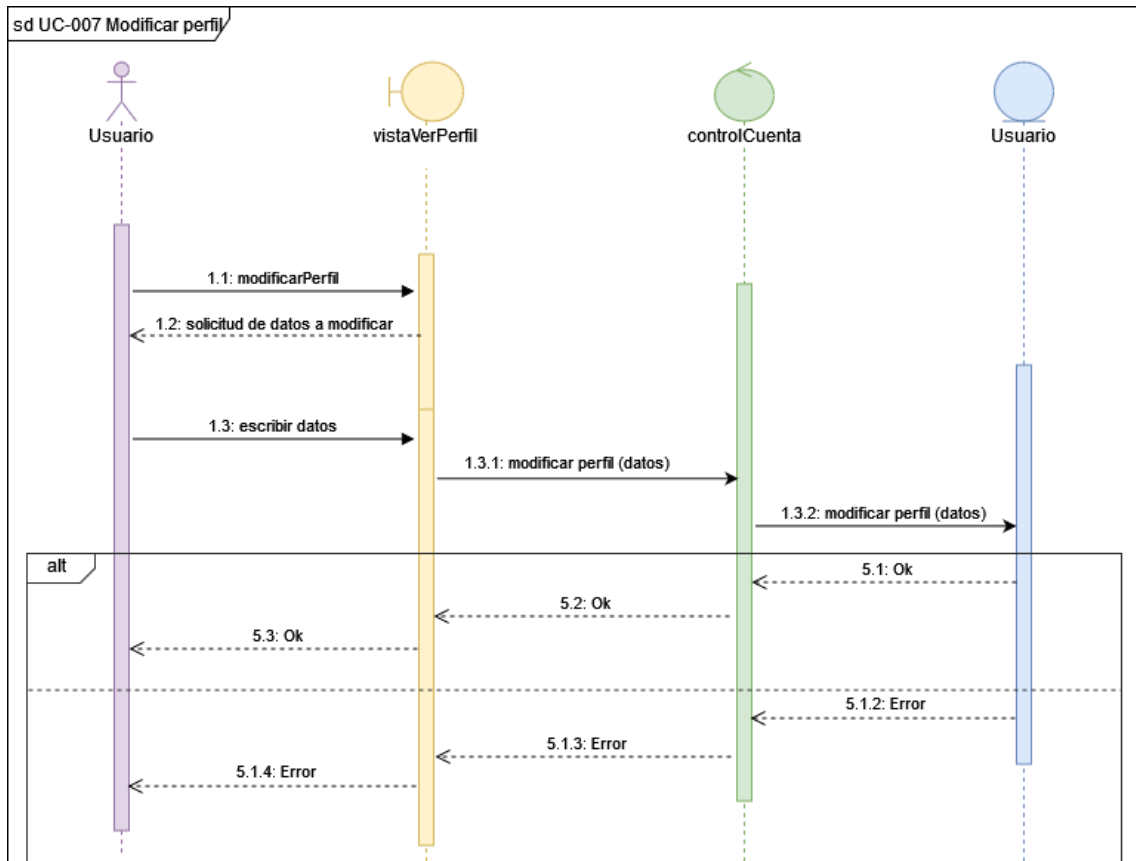


Figura 8: Diagrama de secuencia Modificar perfil

UC-008 – Enviar mensaje.

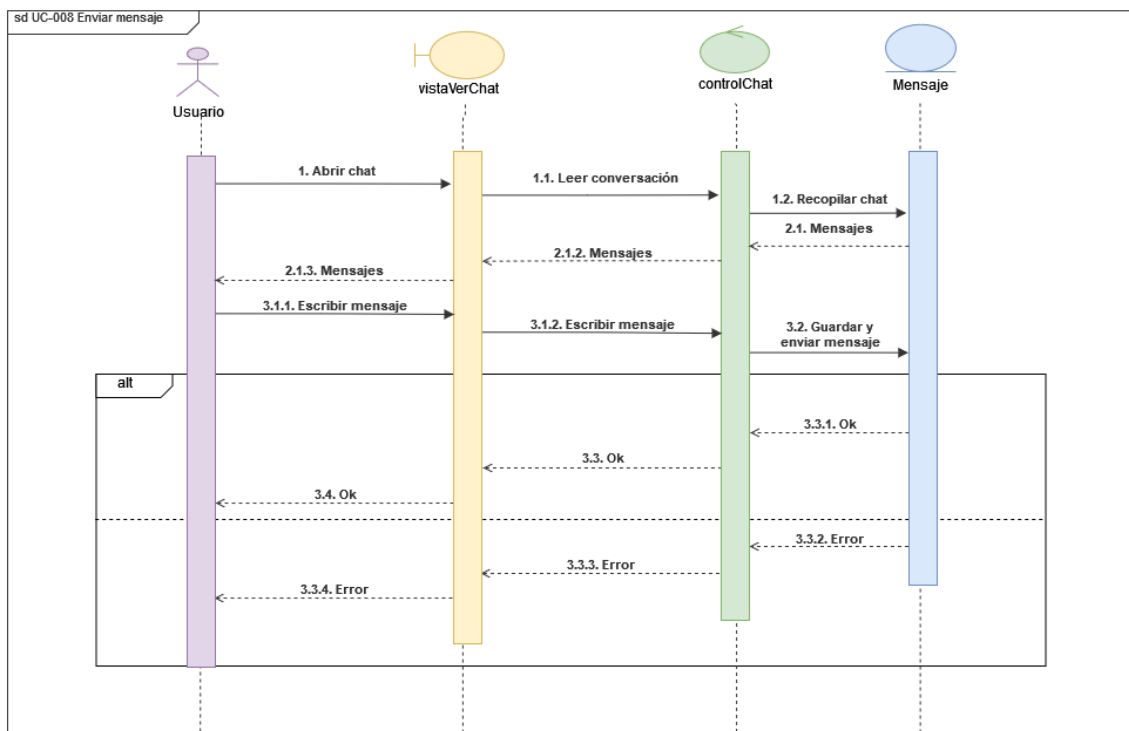


Figura 9: Diagrama de secuencia Enviar mensaje

UC-009 – Ver hospitales más cercanos.

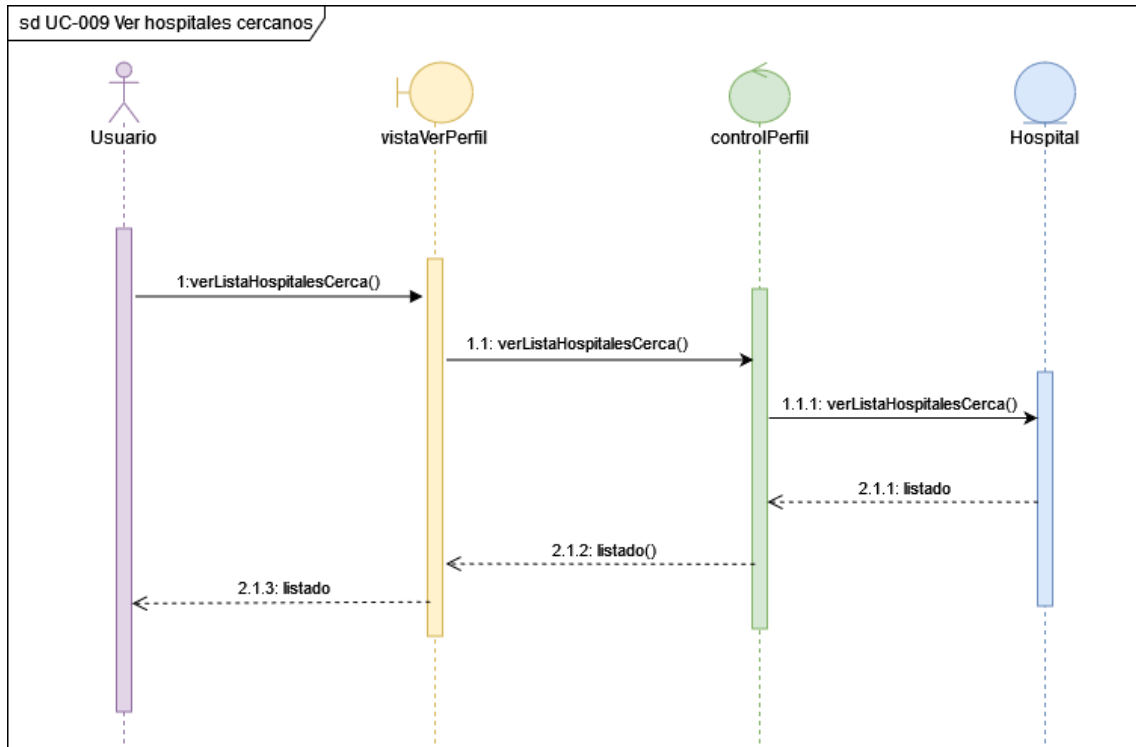


Figura 10: Diagrama de secuencia Ver hospitales cercanos

UC-010 – Ver todos los hospitales registrados.

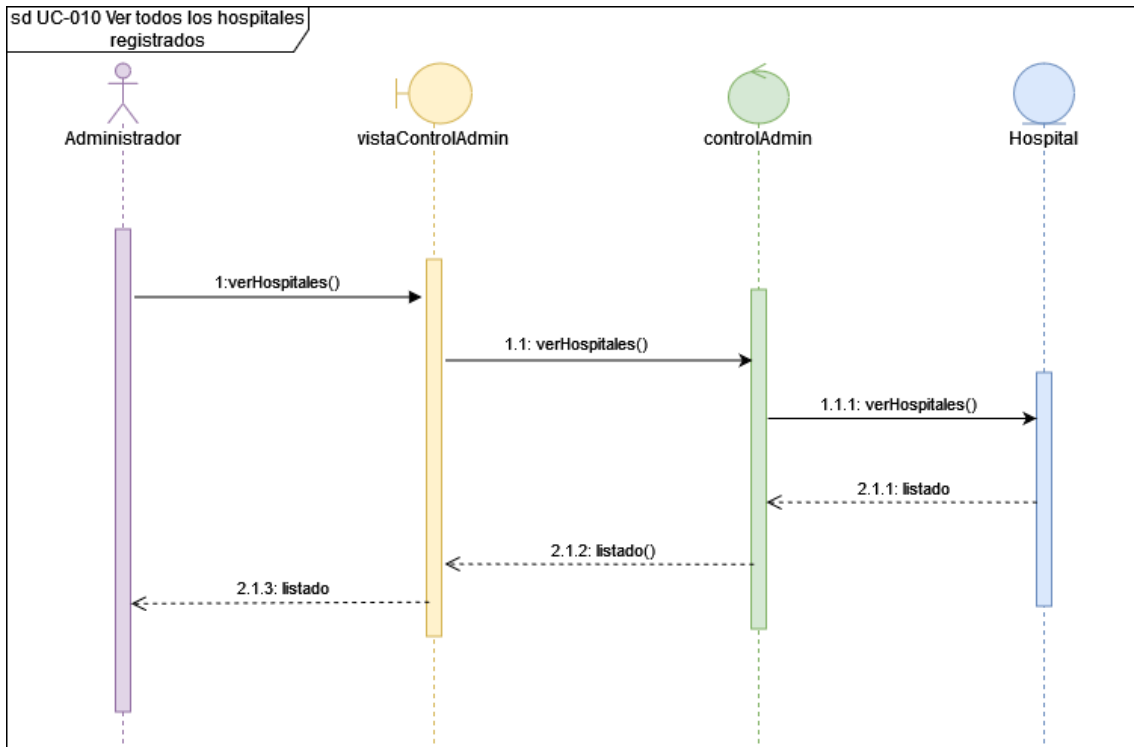


Figura 11: Diagrama de secuencia Ver todos los hospitales registrados

UC-011 – Ver todos los donantes registrados.

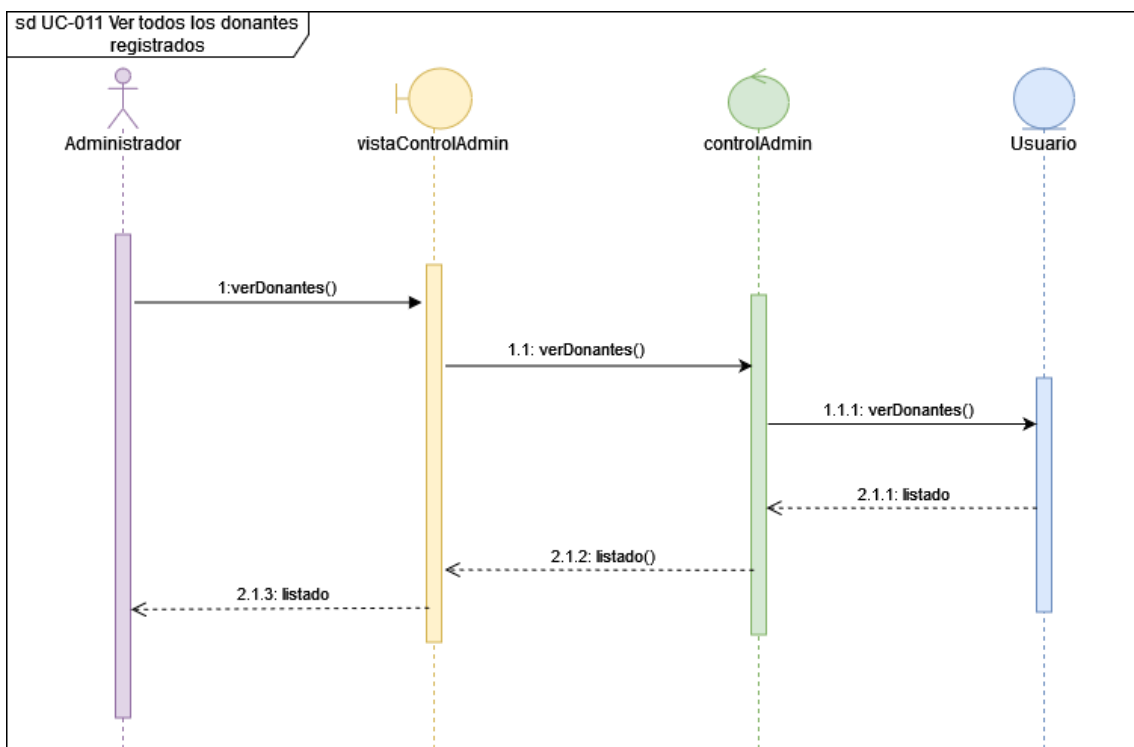


Figura 12: Diagrama de secuencia Ver todos los donantes registrados

UC-012 – Enviar alerta.

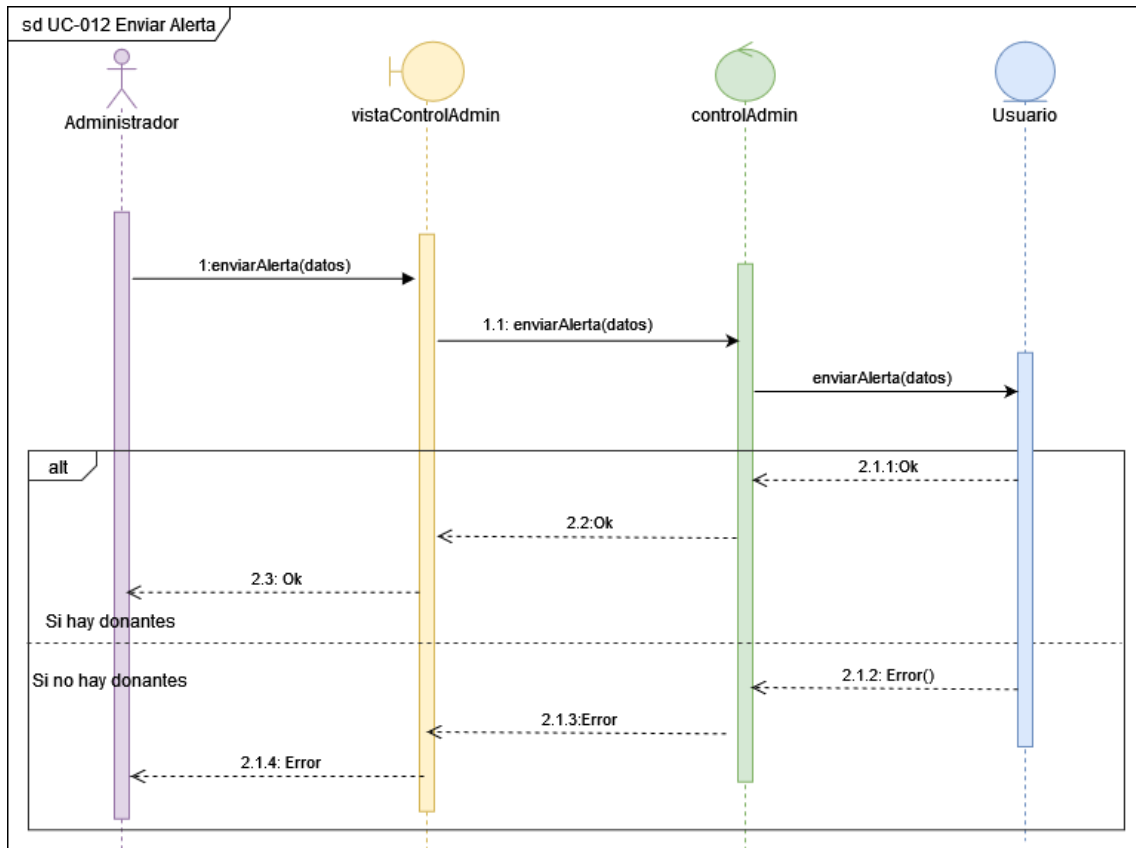


Figura 13: Diagrama de secuencia Enviar alerta

UC-013 – Ver donantes por provincia.

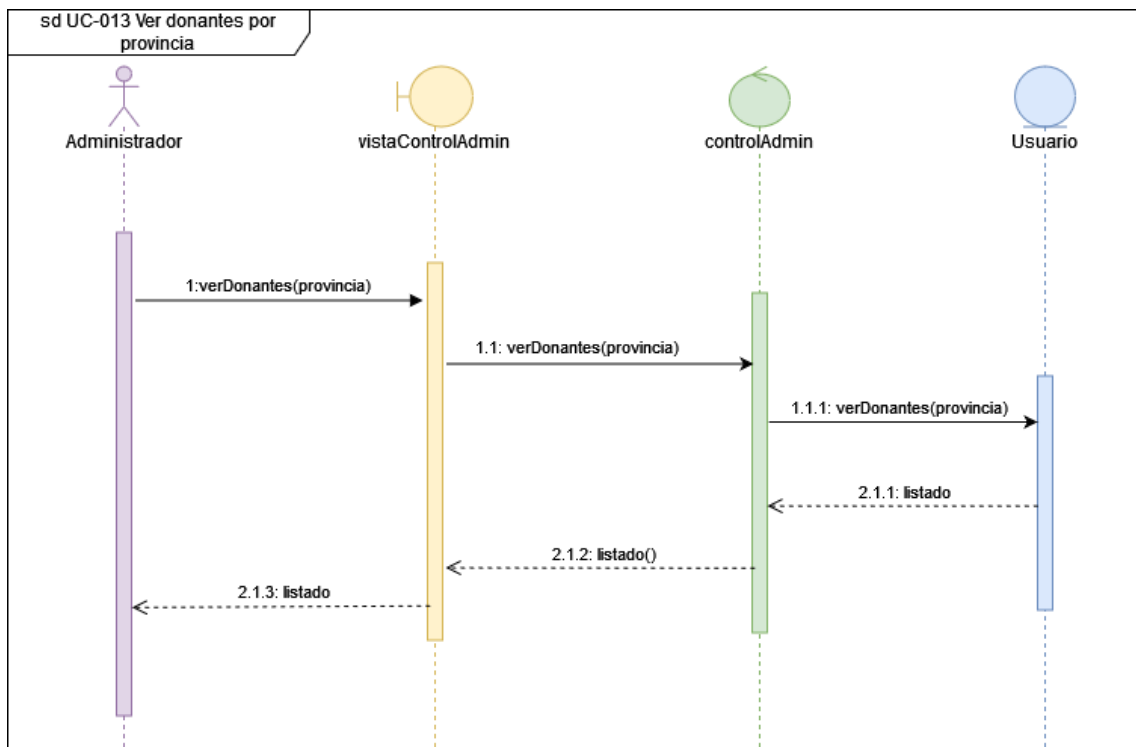


Figura 14: Diagrama de secuencia Ver donantes por provincia

UC-014 – Modificar contraseña.

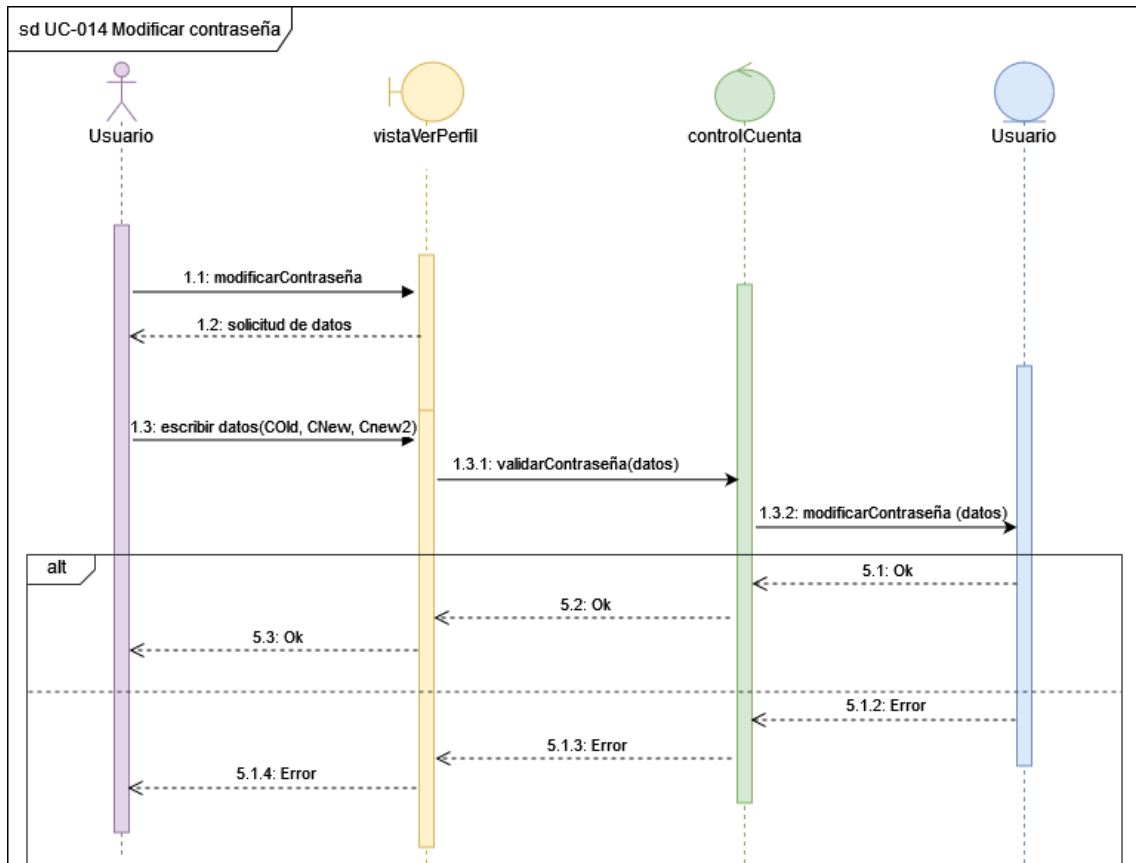


Figura 15: Diagrama de secuencia Modificar contraseña

UC-015 – Buscar donante por nombre.

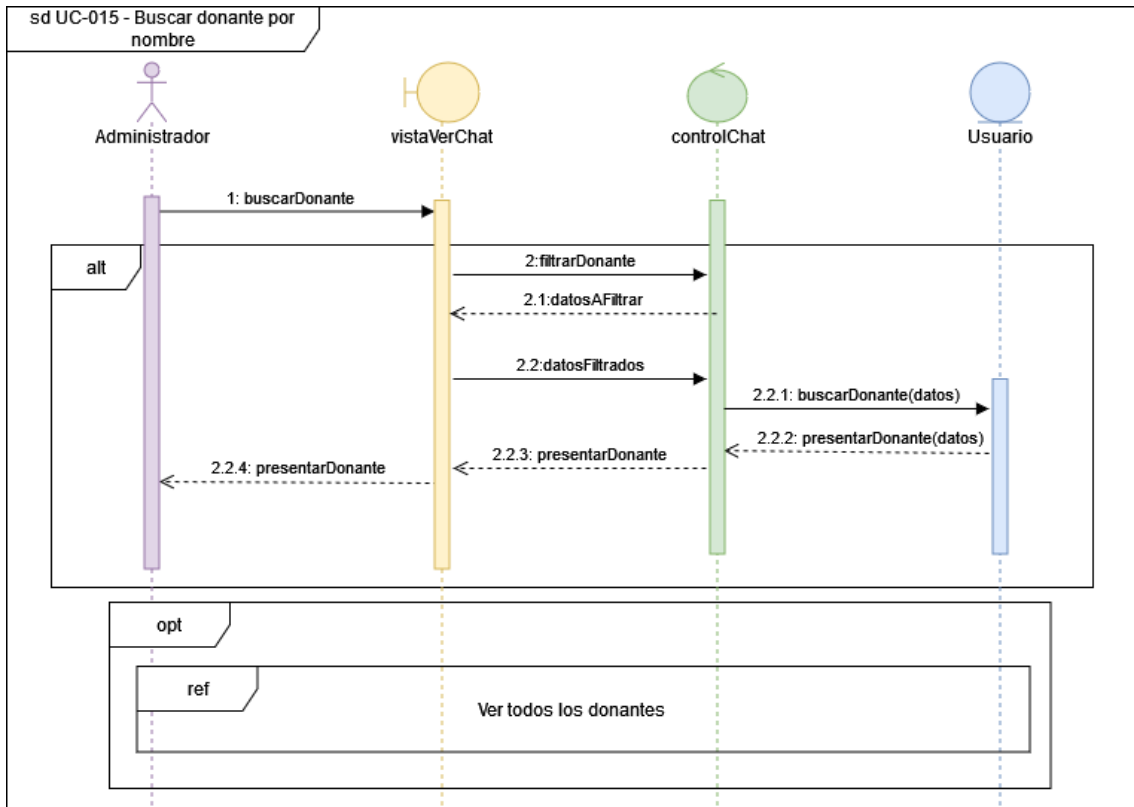


Figura 16: Diagrama de secuencia Buscar donante por nombre

UC-016 – Buscar hospital por nombre.

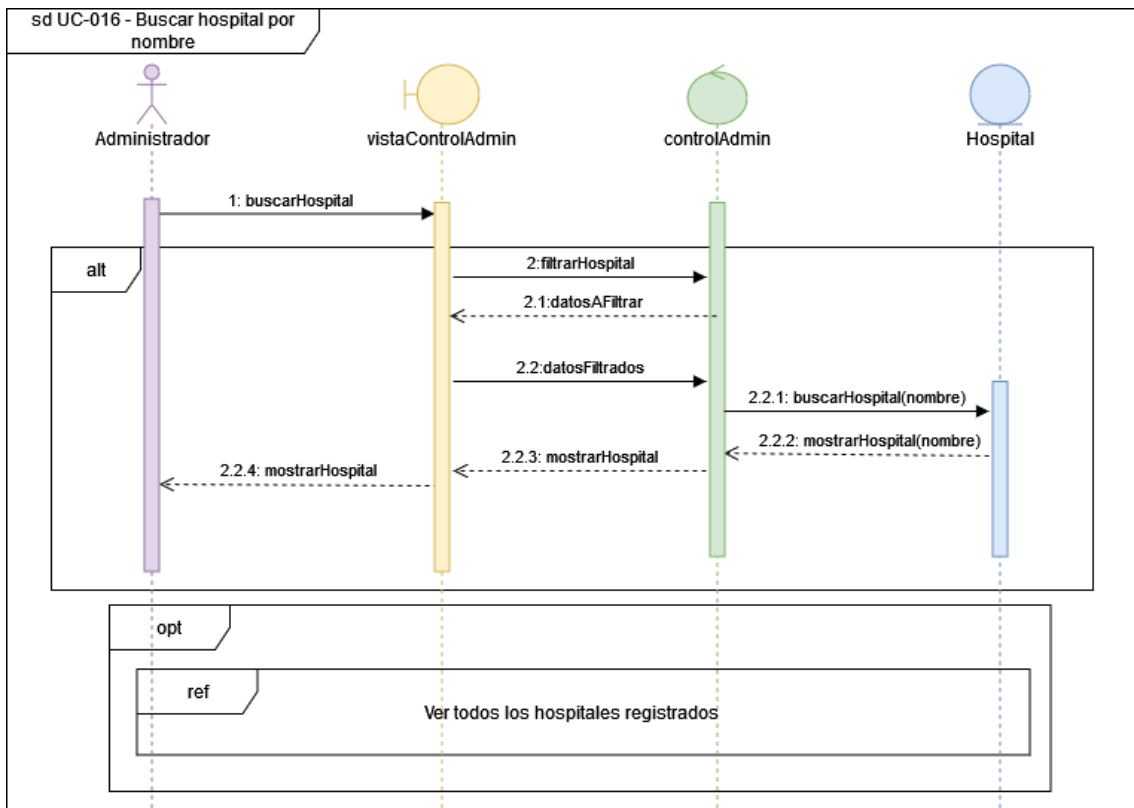


Figura 17: Diagrama de secuencia Buscar hospital por nombre

UC-017 – Confirmar correo electrónico.

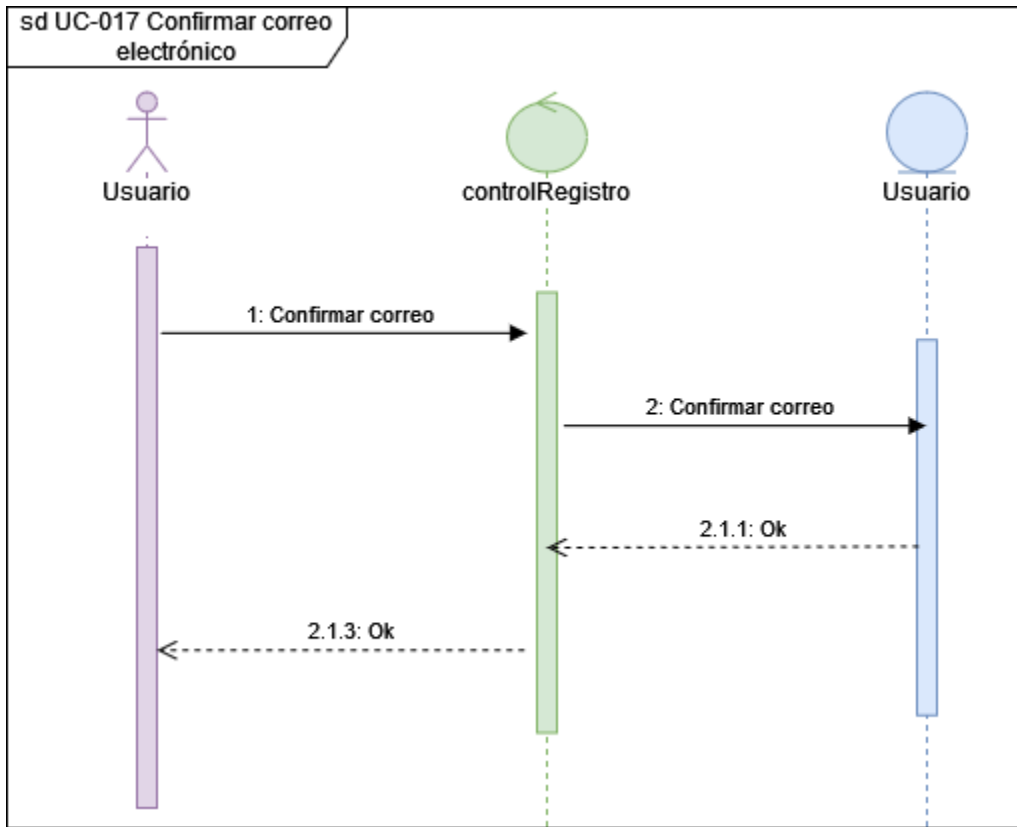


Figura 18: Diagrama de secuencia Confirmar correo electrónico

UC-018 – Eliminar chat.

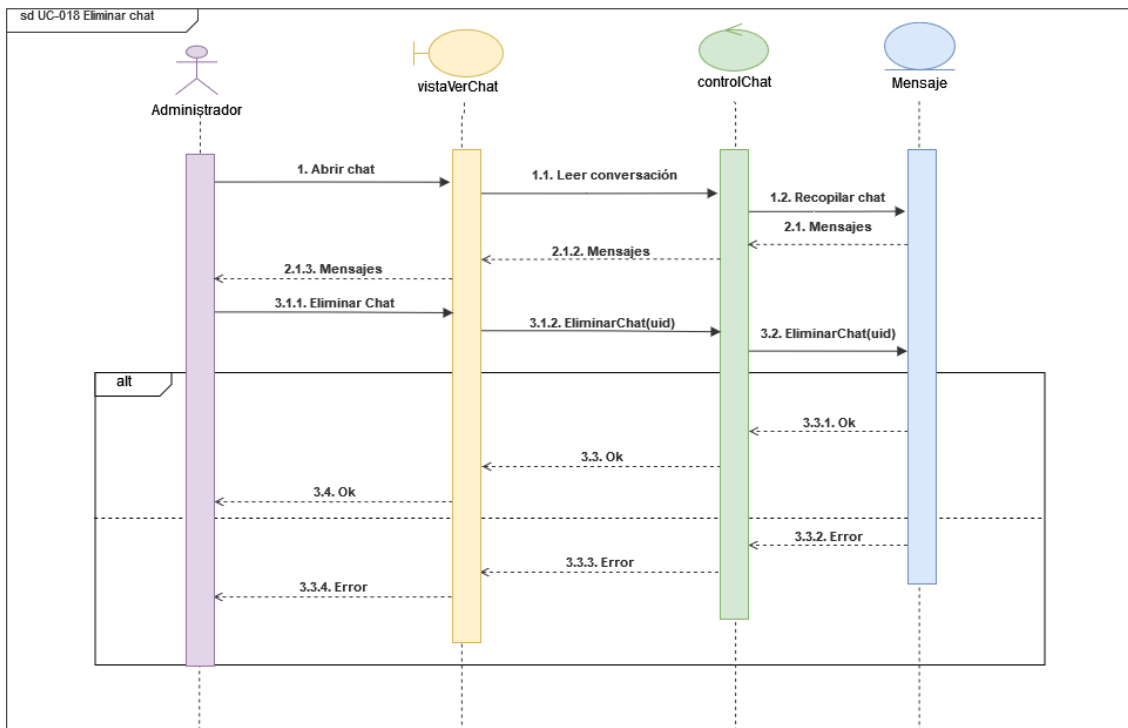


Figura 19: Diagrama de secuencia Eliminar chat

UC-019 – Eliminar hospital.

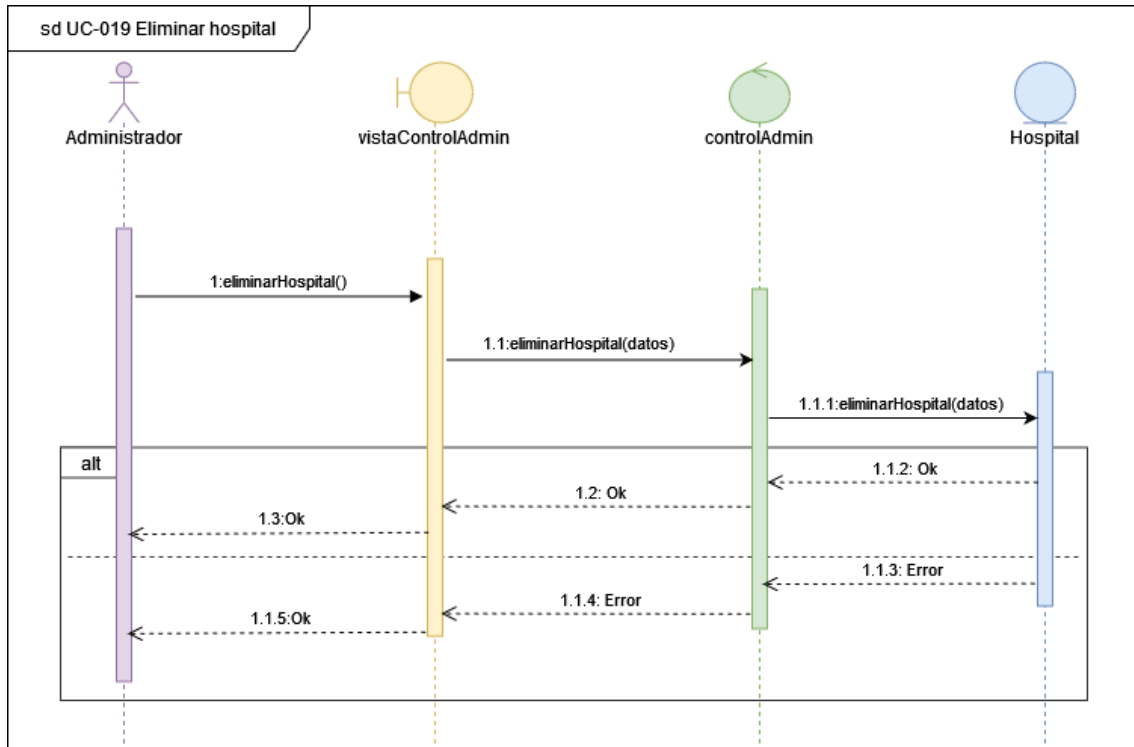


Figura 20: Diagrama de secuencia Eliminar hospital

UC-020 – Buscar donantes por grupo sanguíneo.

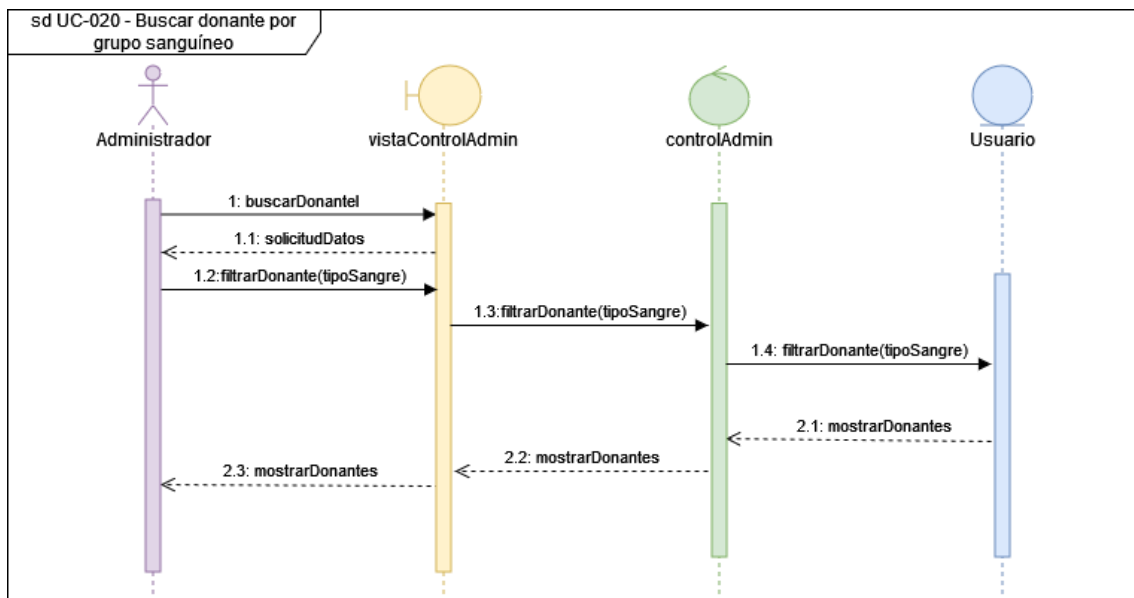


Figura 21: Diagrama de secuencia Buscar donantes por grupo sanguíneo

4. Descripción de la arquitectura.

Para concluir, se definen las relaciones entre las distintas clases previamente mostradas, así como la división en paquetes.

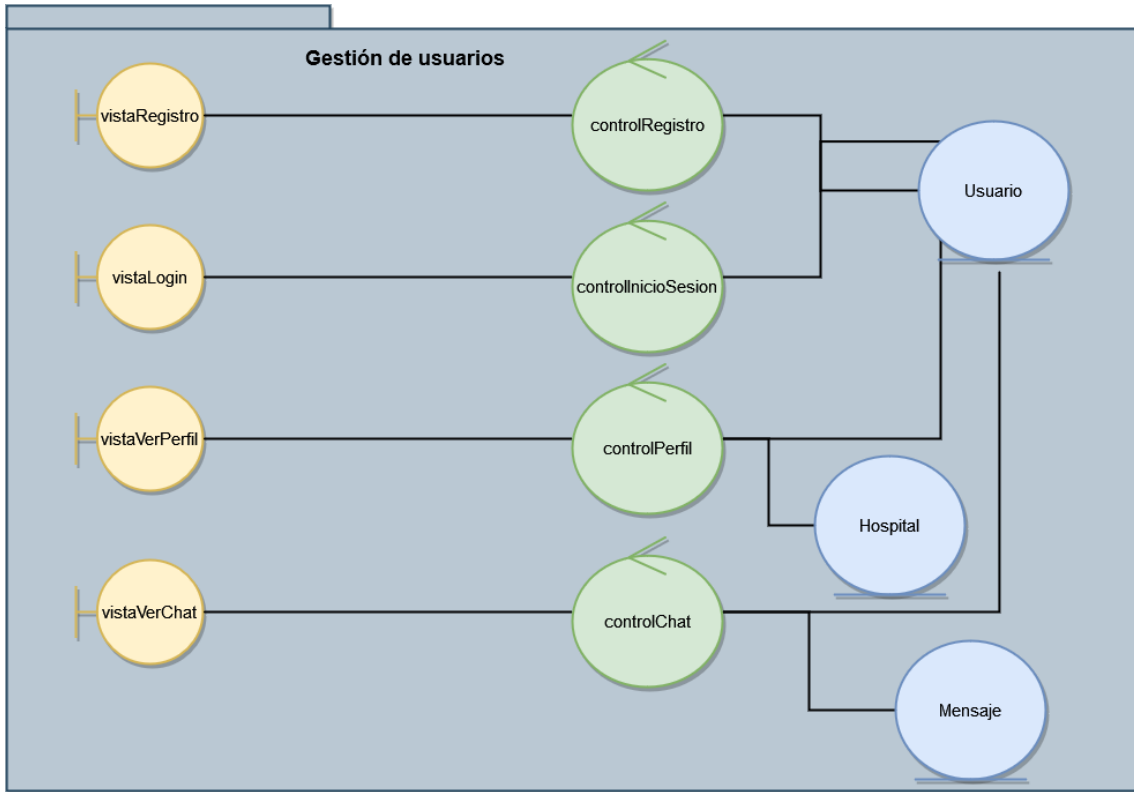


Figura 22: Descripción de la Arquitectura del Modelo de Análisis Gestión de usuarios

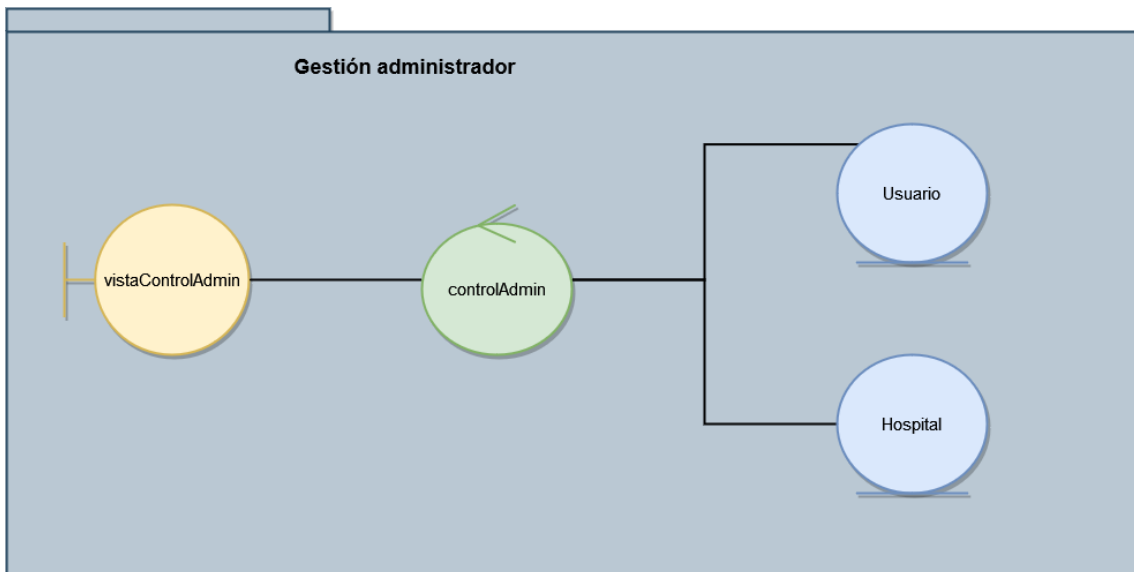


Figura 23: Descripción de la Arquitectura del Modelo de Análisis Gestión Administrador

5. Bibliografía.

[1] F. J. García Peñalvo y M. N. Moreno García, “Transparencias de Ingeniería del Software I, Tema 6 – UML Fundamentos de la vista de casos de uso” [2020]

[2] «Diagrams.net» Enlace: <https://www.diagrams.net> [2005]

ANEXO IV:

Diseño del sistema software

Trabajo Fin de Grado Ingeniería Informática

Septiembre 2022



**VNiVERSIDAD
D SALAMANCA**

Autor:

Pedro Luis Alonso Díez

Tutores:

Luis Augusto Silva

Gabriel Villarrubia González

Juan Francisco de Paz Santana

Tabla de contenido

1. Introducción.....	1
2. Modelo de diseño.....	2
2.1. Patrón arquitectónico.....	2
2.2. Presentation and Container components.....	3
3. Subsistemas de diseño.....	4
4. Clases de diseño.....	5
5. Vista arquitectónica de diseño.....	7
6. Realización de casos de uso de diseño.....	7
7. Prototipo digital.....	17
7.1. Registro/Inicio de sesión.....	18
7.2. Chat.....	19
7.3. Perfil.....	19
7.4. Control administrador.....	20
8. Bibliografía.....	22

Índice de figuras

Figura 1: Patrón arquitectónico MVVM	2
Figura 2: Patrón arquitectónico Presentation and Container components.....	3
Figura 3: Vista subsistemas de diseño.....	4
Figura 4: Clases de diseño Modelo.....	5
Figura 5: Clases de diseño Vista	5
Figura 6: Clases de diseño ViewModel.....	6
Figura 7: Vista arquitectónica	7
Figura 8: Diagrama de secuencia Alta usuario	8
Figura 9: Diagrama de secuencia Inicio sesión.....	8
Figura 10: Diagrama de secuencia Cerrar sesión	9
Figura 11: Diagrama de secuencia Recuperar contraseña	9
Figura 12: Diagrama de secuencia Eliminar cuenta	10
Figura 13: Diagrama de secuencia Ver perfil.....	10
Figura 14: Diagrama de secuencia Modificar perfil.....	11
Figura 15: Diagrama de secuencia Enviar mensaje	11
Figura 16: Diagrama de secuencia Ver hospitales cercanos	12
Figura 17: Diagrama de secuencia Ver todos los hospitales registrados	12
Figura 18: Diagrama de secuencia Ver todos los donantes registrados	13
Figura 19: Diagrama de secuencia Enviar alerta	13
Figura 20: Diagrama de secuencia Ver donantes por provincia.....	14
Figura 21: Diagrama de secuencia Modificar contraseña	14
Figura 22: Diagrama de secuencia Buscar donante por nombre	15
Figura 23: Diagrama de secuencia Buscar hospital por nombre.....	15
Figura 24: Diagrama de secuencia Confirmar correo electrónico.....	16
Figura 25: Diagrama de secuencia Eliminar chat	16
Figura 26: Diagrama de secuencia Eliminar hospital	17
Figura 27: Diagrama de secuencia Buscar donante por grupo sanguíneo.....	17
Figura 28: Prototipo digital Registro	18
Figura 29: Prototipo digital Inicio sesión.....	18
Figura 30: Prototipo digital Chat	19
Figura 31: Prototipo digital Perfil	19
Figura 32: Prototipo digital Perfil modificar contraseña.....	20
Figura 33: Prototipo digital Administrador base.....	20
Figura 34: Prototipo digital Administrador ver donantes por provincia.....	21

Figura 35: Prototipo digital Administrador ver hospitales..... 21

Figura 36: Prototipo digital Administrador ver hospitales tabla..... 22

Figura 37: Modelo de despliegue..... 22

1. Introducción.

En este anexo se documenta el modelo de diseño del sistema. Durante esta fase se describirá la realización de los casos de uso, teniendo en cuenta los requisitos funcionales y no funcionales en el sistema. El objetivo es describir la estructura del software a implementar junto con las interfaces, algoritmos y datos que forman parte de él [1].

Siguiendo el modelo de los anteriores anexos se utilizará Diagrams.net para la realización de los diagramas [2].

2. Modelo de diseño.

El modelo de diseño recoge la arquitectura global del sistema y como se relacionan sus componentes a través de una abstracción de la implementación.

2.1. Patrón arquitectónico.

Para realizar el diseño del sistema se ha optado como guía el patrón arquitectónico MVVM. Debido a que este patrón está enfocado al desarrollo de interfaces de usuario.

Este patrón tiene como característica principal ofrecer un gran grado de desacoplo entre la interfaz de usuario y la lógica de la aplicación, sin embargo, a diferencia de otros patrones como MVC, el patrón MVVM permite actualizar cambios en la vista o en los datos de forma automática [3].

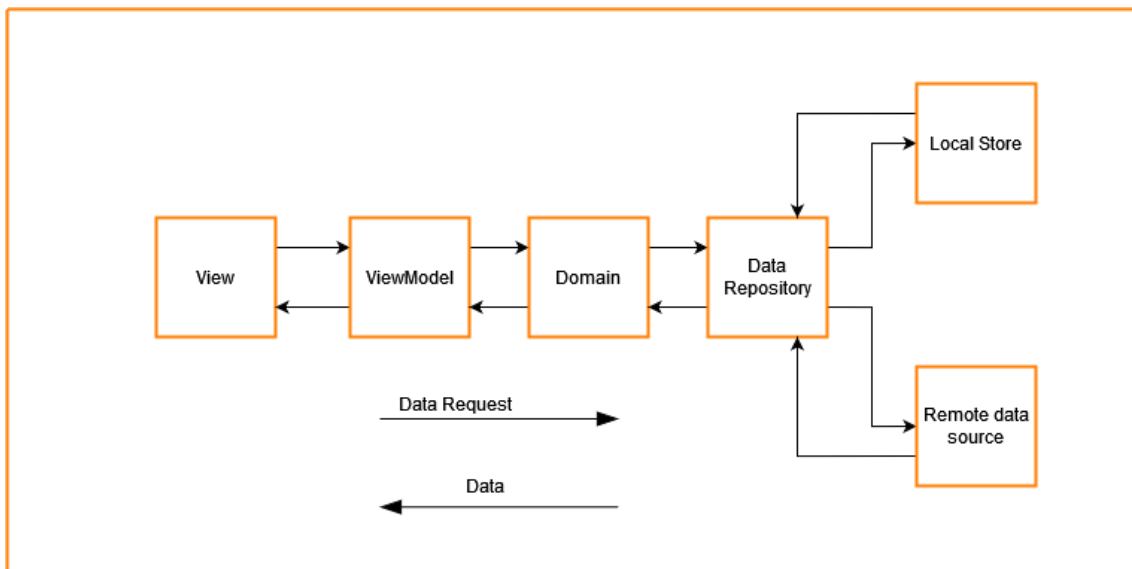


Figura 1: Patrón arquitectónico MVVM

A continuación, se detallan las funciones de cada parte del patrón arquitectónico MVVM.

- **View**: La vista son las interfaces con las que los usuarios interactúan. Reciben inputs de los usuarios y muestran información recibida por el ViewModel o el modelo
- **ViewModel**: Es el responsable de coordinar las interacciones de la vista con las clases de modelo necesarias.
- **Domain**: Contiene la funcionalidad principal del sistema, es decir, sus casos de uso. Esta capa encapsula lógica que pueden consumir varios ViewModels. De esta forma evitamos duplicidad de código y código más simple ya que permite dividir responsabilidades.

- **Data Repository:** Aquí se encuentran los datos de los que dispone el sistema. Estos pueden ser obtenidos mediante datos locales (ficheros json, xml, etc.) o mediante datos remotos, es decir, bases de datos.

2.2. Presentation and Container components.

Se trata de un patrón muy simple que divide el código en dos tipos de componentes, Presentation components y container components. Este tipo de patrón presenta varios beneficios, como, por ejemplo, mejor separación del código, debido a que por un lado tenemos la apariencia y por otro la lógica. También permite la reutilización del código ya que podemos crear componentes que sirvan para varias interfaces.

- **Presentation components:** Se trata de la apariencia de la interfaz, solo recibe datos y no manejan estado.
- **Container components:** Este tipo de componentes contienen la lógica del sistema y deben proporcionar los datos a los componentes de presentación.

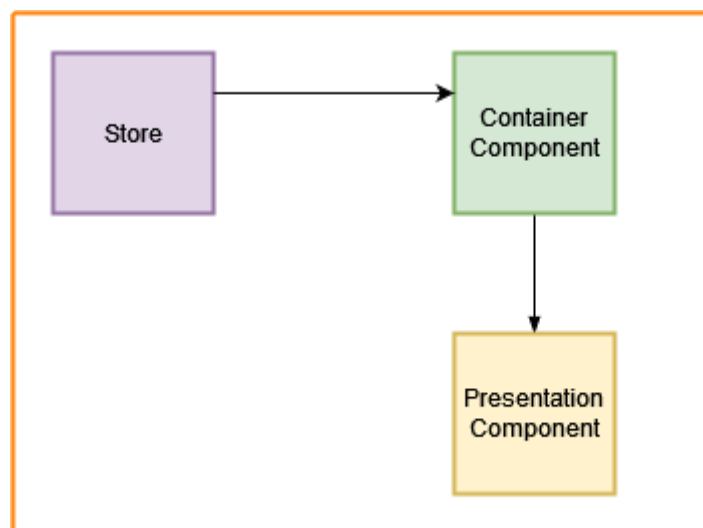


Figura 2: Patrón arquitectónico Presentation and Container components

3. Subsistemas de diseño.

Tomando como punto de partida el patrón arquitectónico previamente descrito, describiremos las relaciones entre los distintos componentes, en nuestra aplicación solo dispondremos de un sistema de diseño.

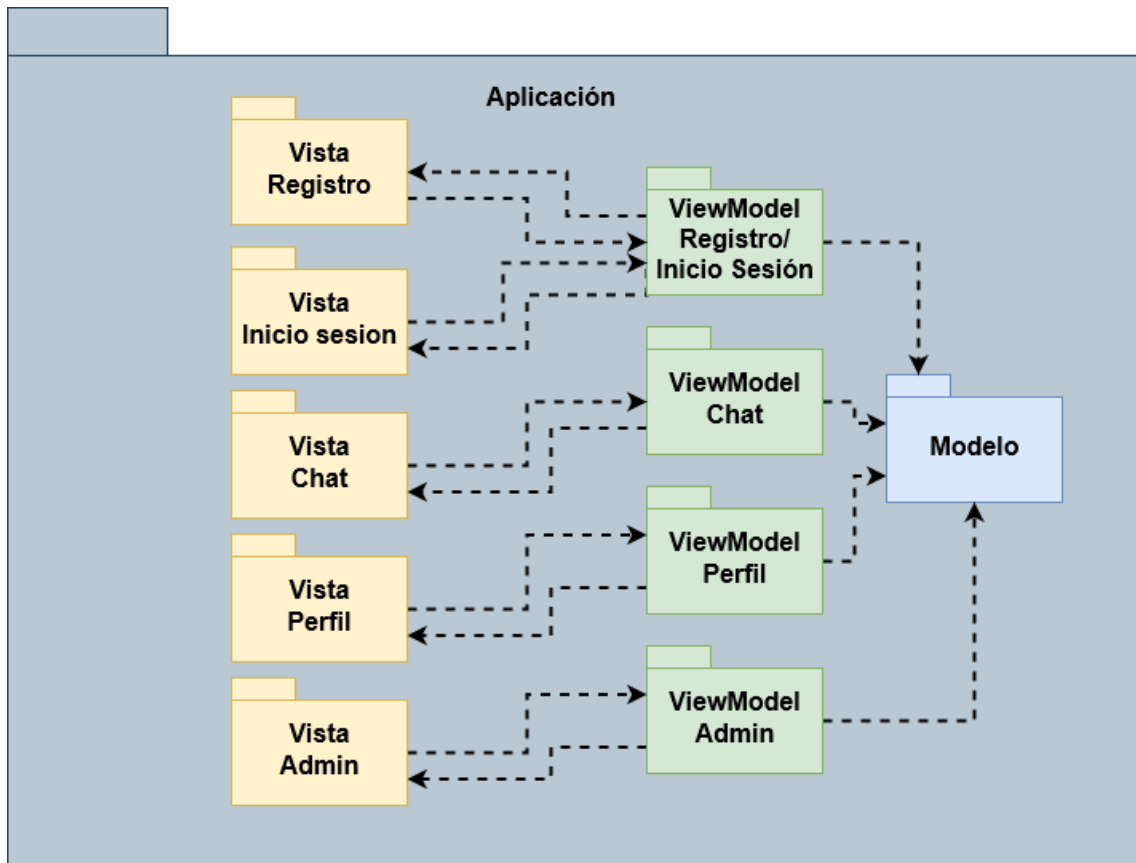


Figura 3: Vista subsistemas de diseño

4. Clases de diseño.

En esta fase describiremos a detalle los componentes del modelo, vista y viewmodel a bajo nivel para conseguir un sistema con bajo acoplamiento y alta cohesión.

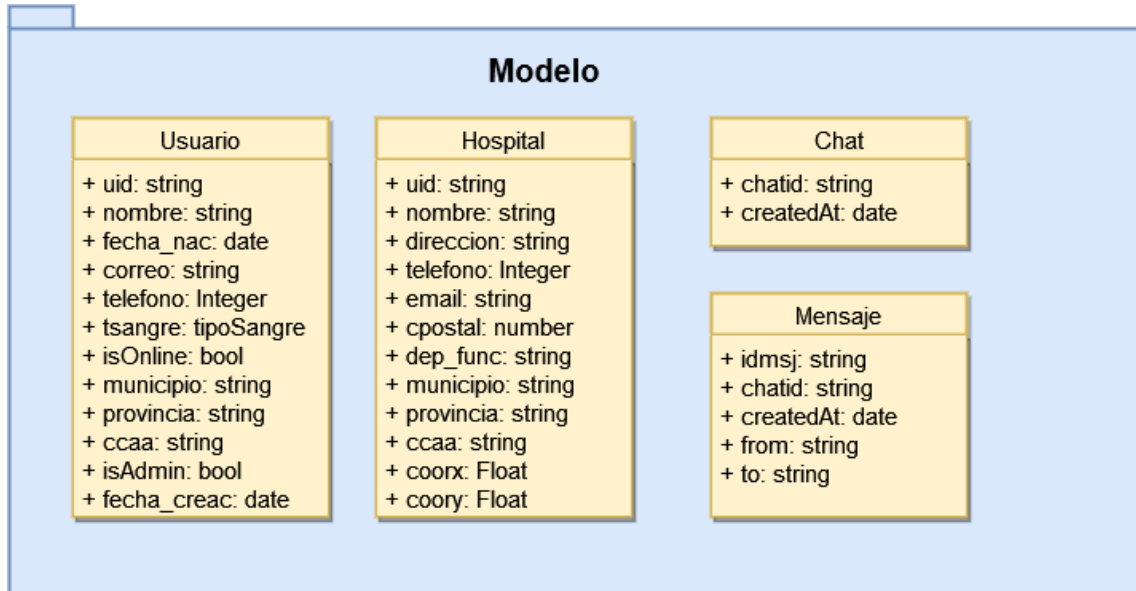


Figura 4: Clases de diseño Modelo

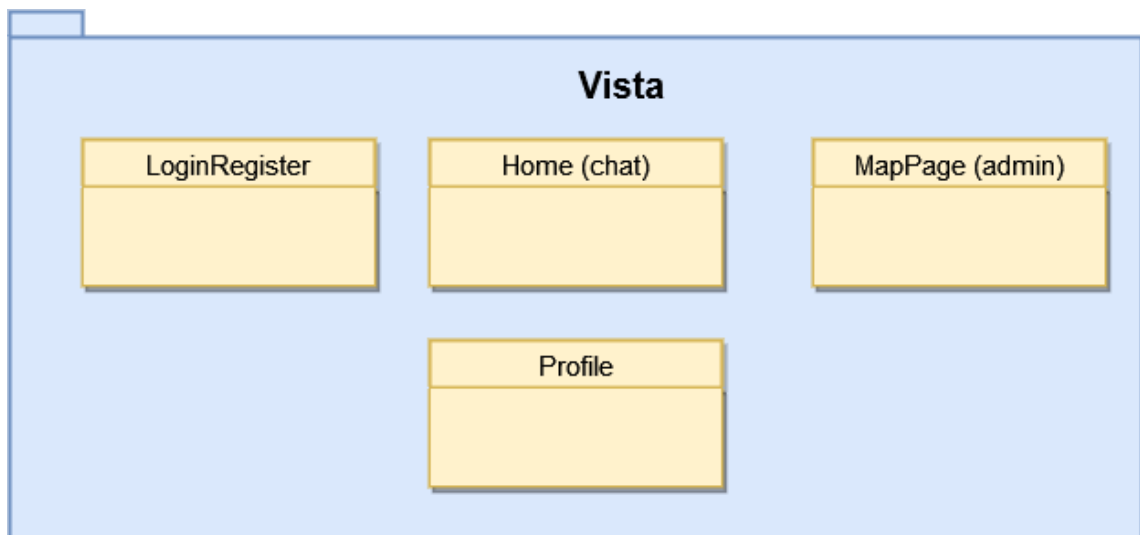


Figura 5: Clases de diseño Vista

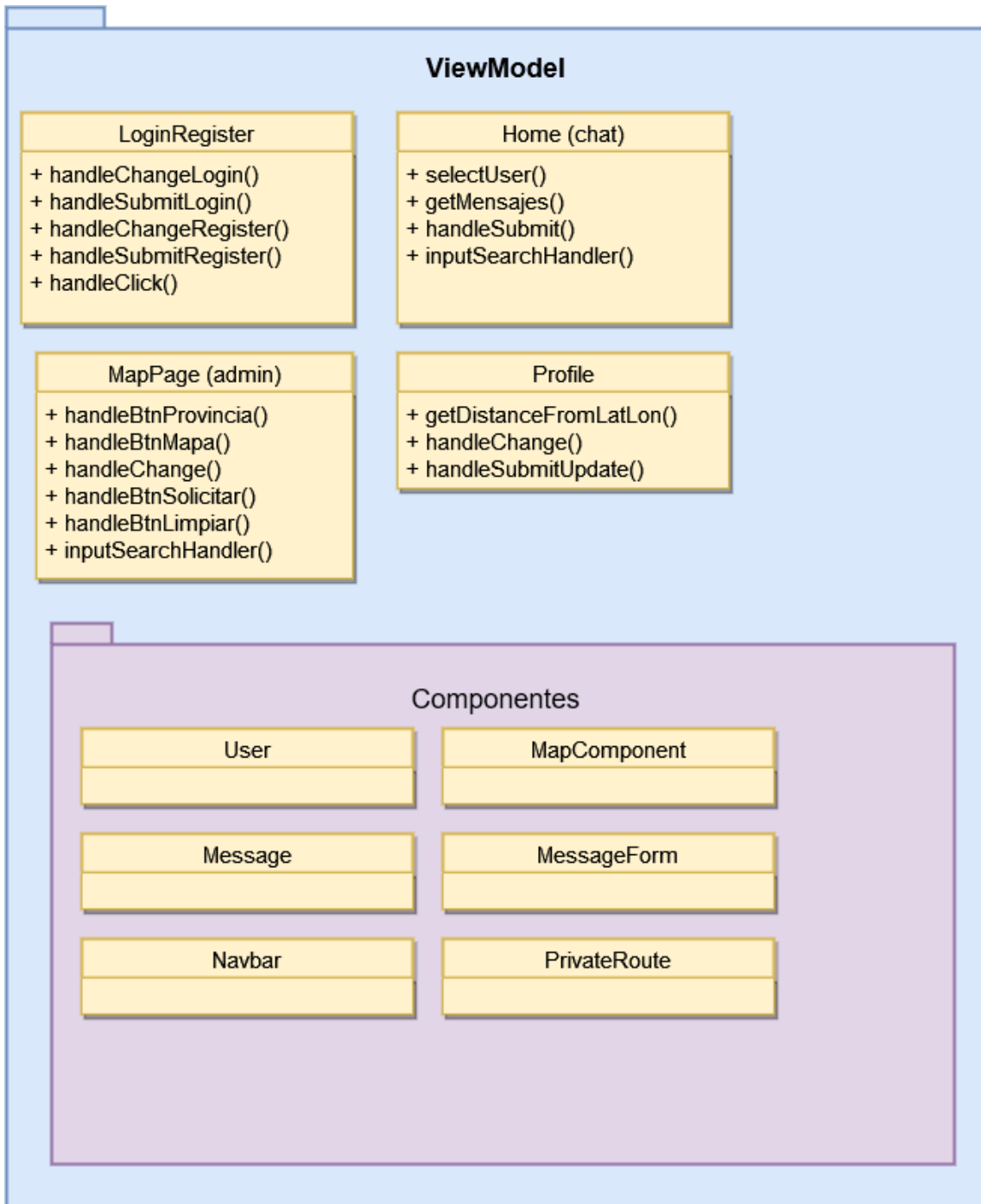


Figura 6: Clases de diseño ViewModel

5. Vista arquitectónica de diseño.

En se detalla como quedaría todo el sistema con todas las relaciones, siguiendo los patrones descritos previamente.

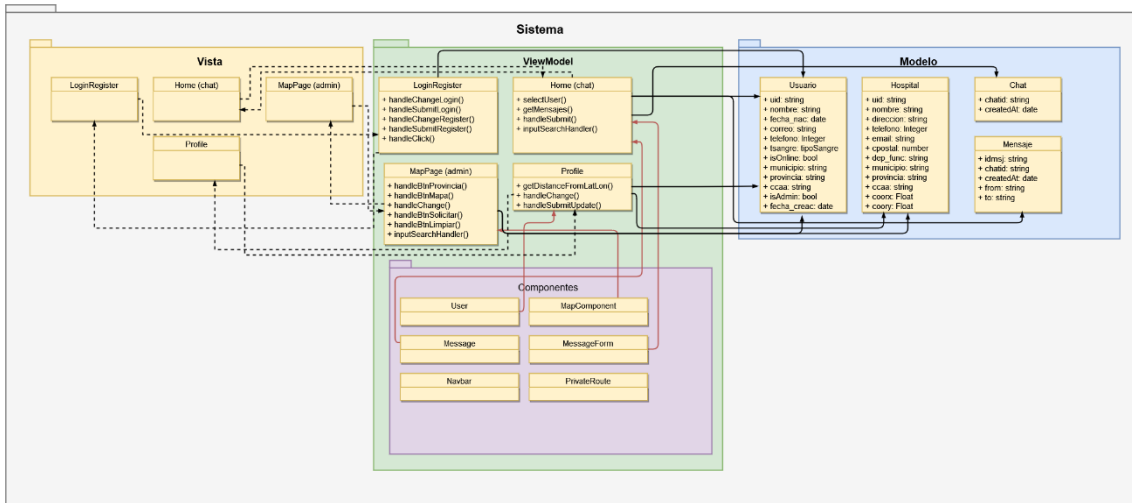


Figura 7: Vista arquitectónica

6. Realización de casos de uso de diseño.

Se realizarán diagramas de secuencia para todos los casos de usos presentados en el Anexo II

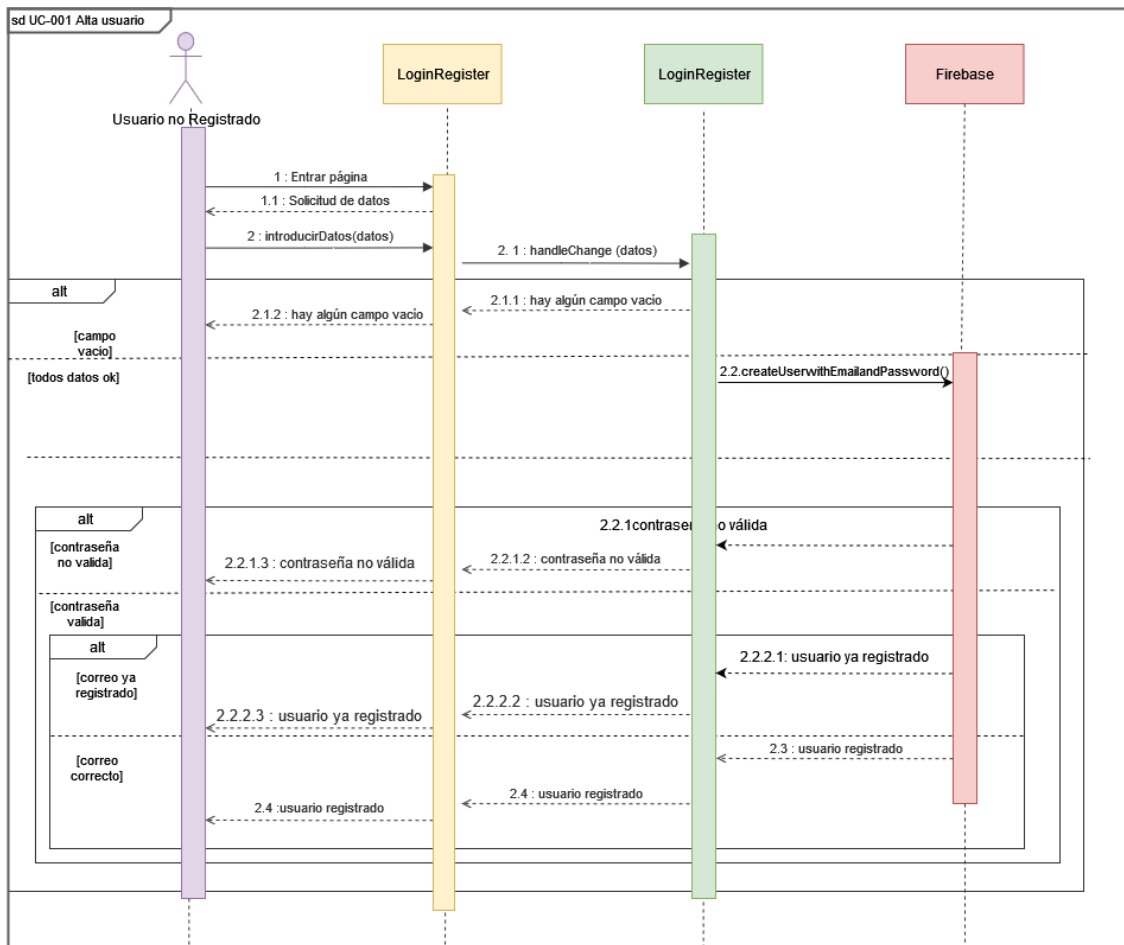


Figura 8: Diagrama de secuencia Alta usuario

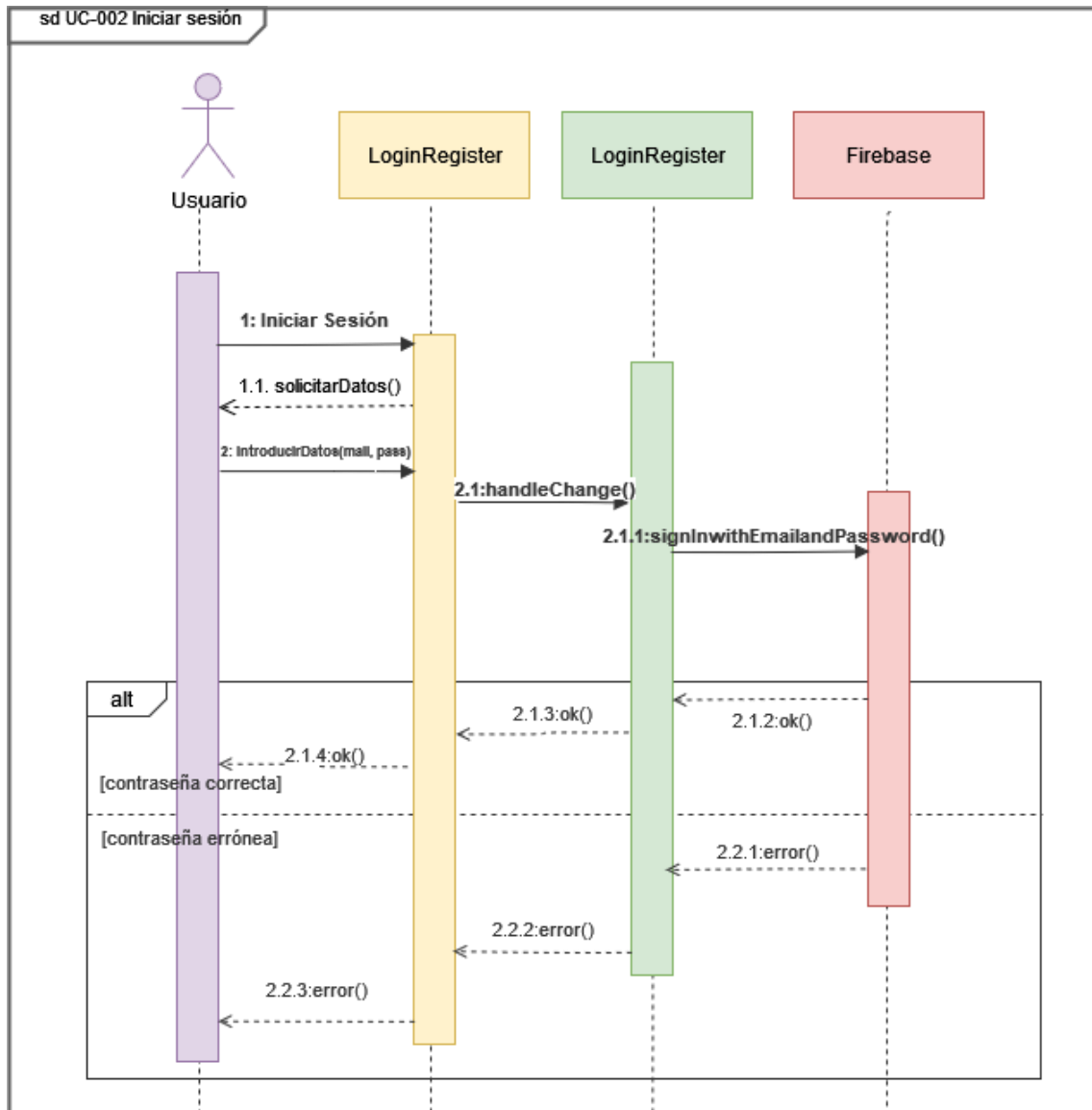


Figura 9: Diagrama de secuencia Inicio sesión

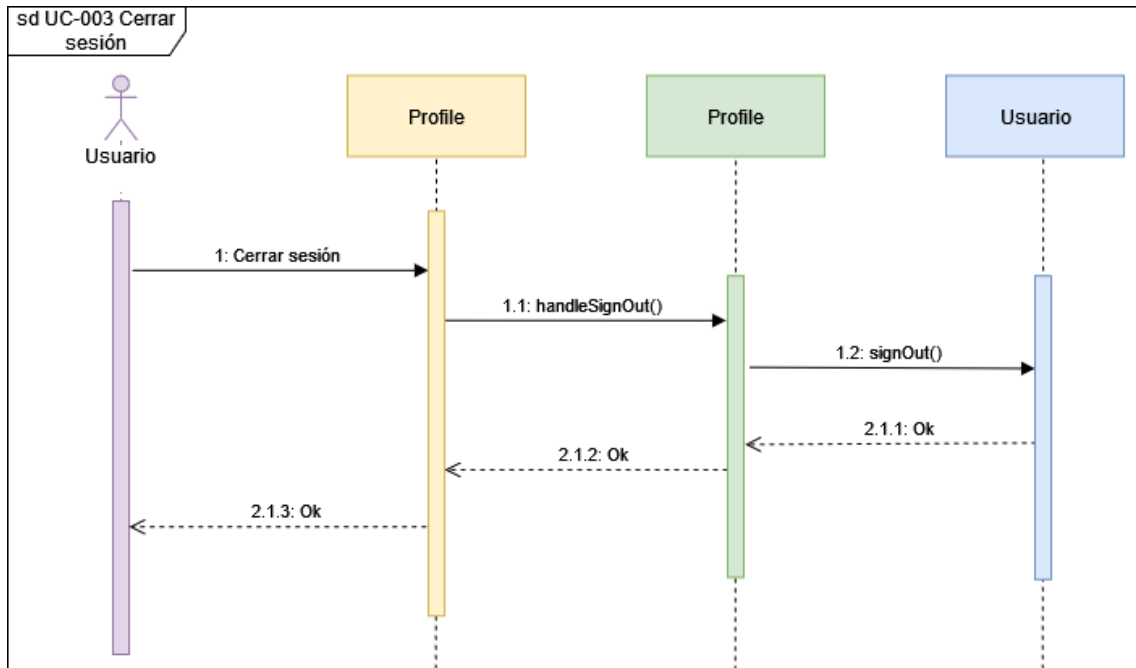


Figura 10: Diagrama de secuencia Cerrar sesión

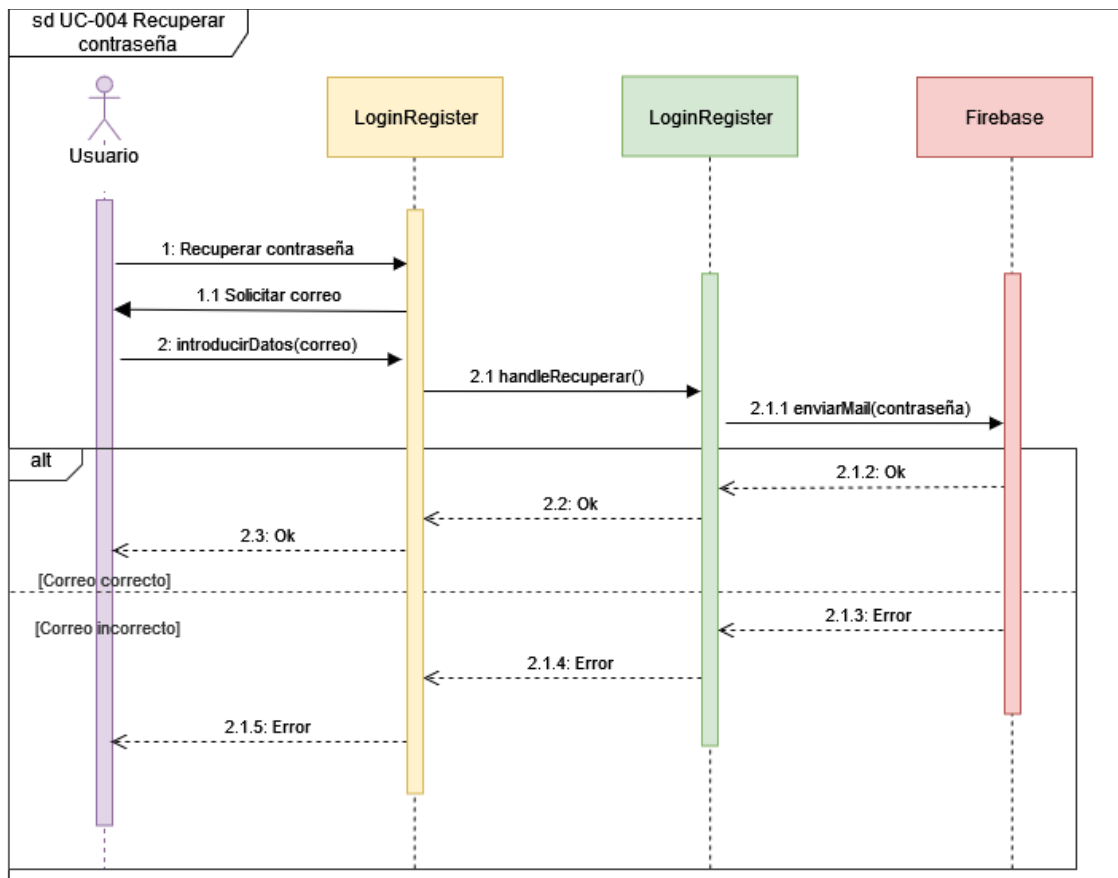


Figura 11: Diagrama de secuencia Recuperar contraseña

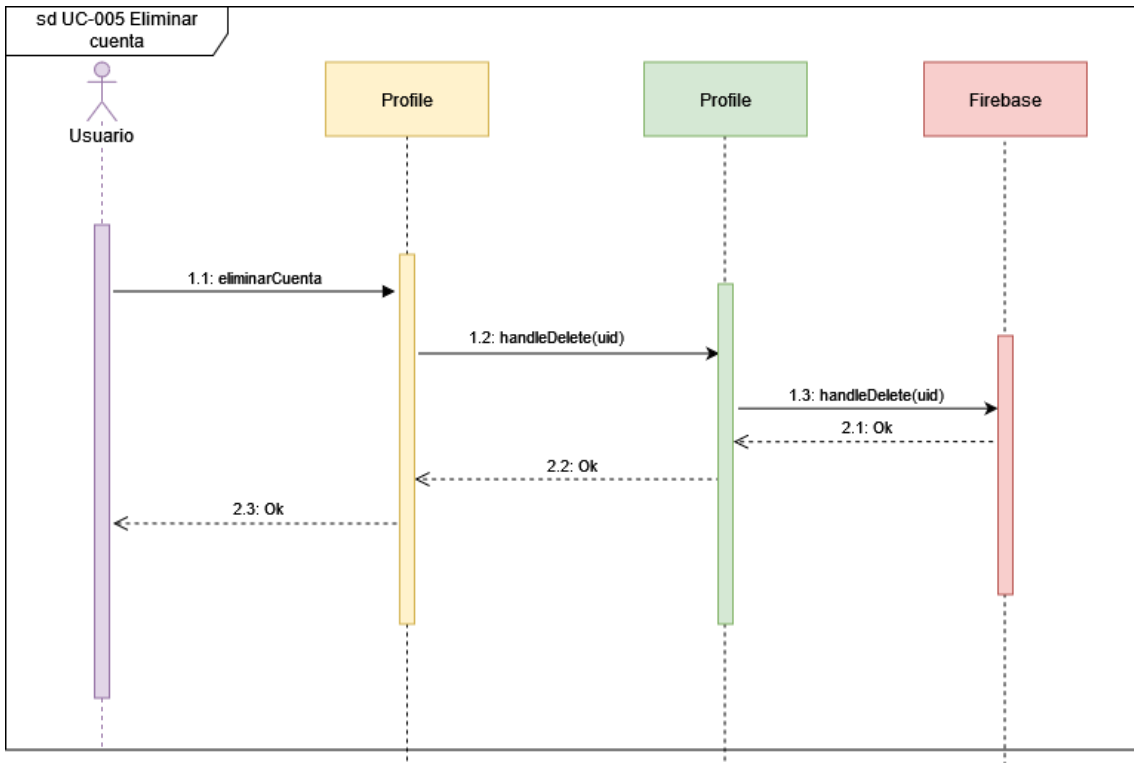


Figura 12: Diagrama de secuencia Eliminar cuenta

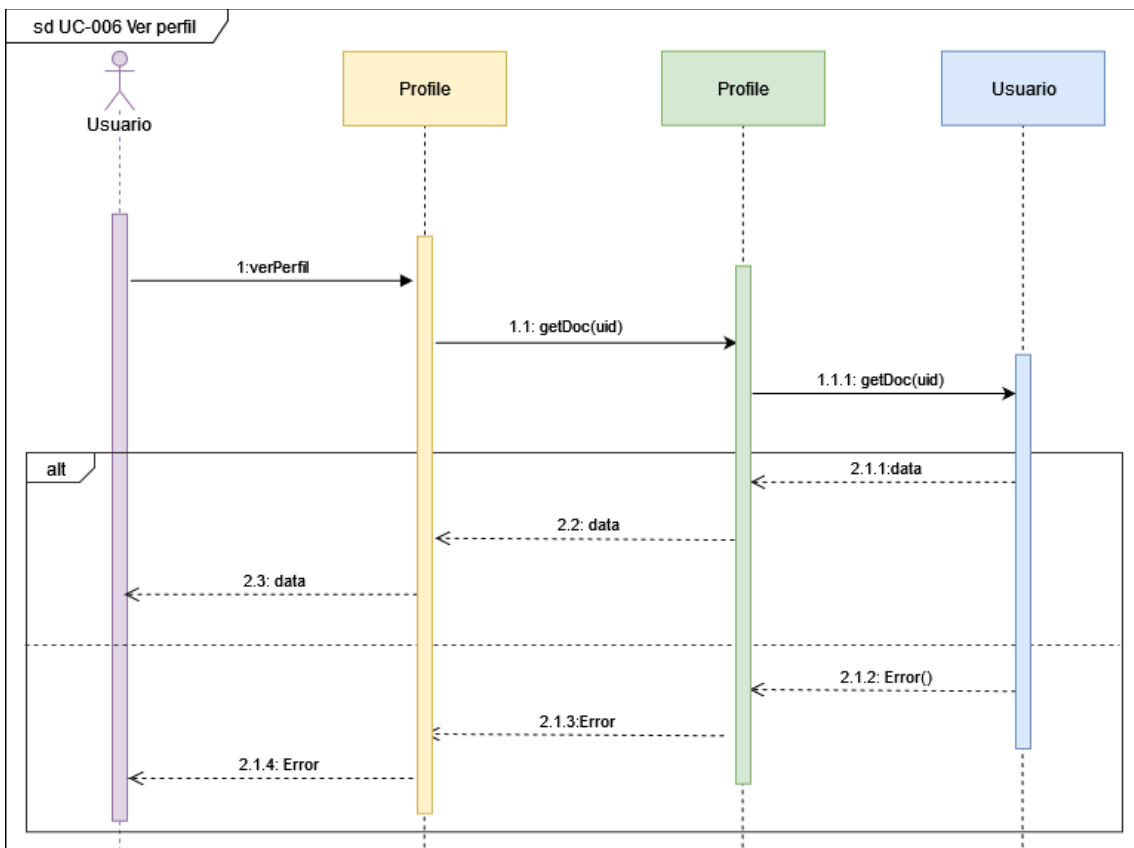


Figura 13: Diagrama de secuencia Ver perfil

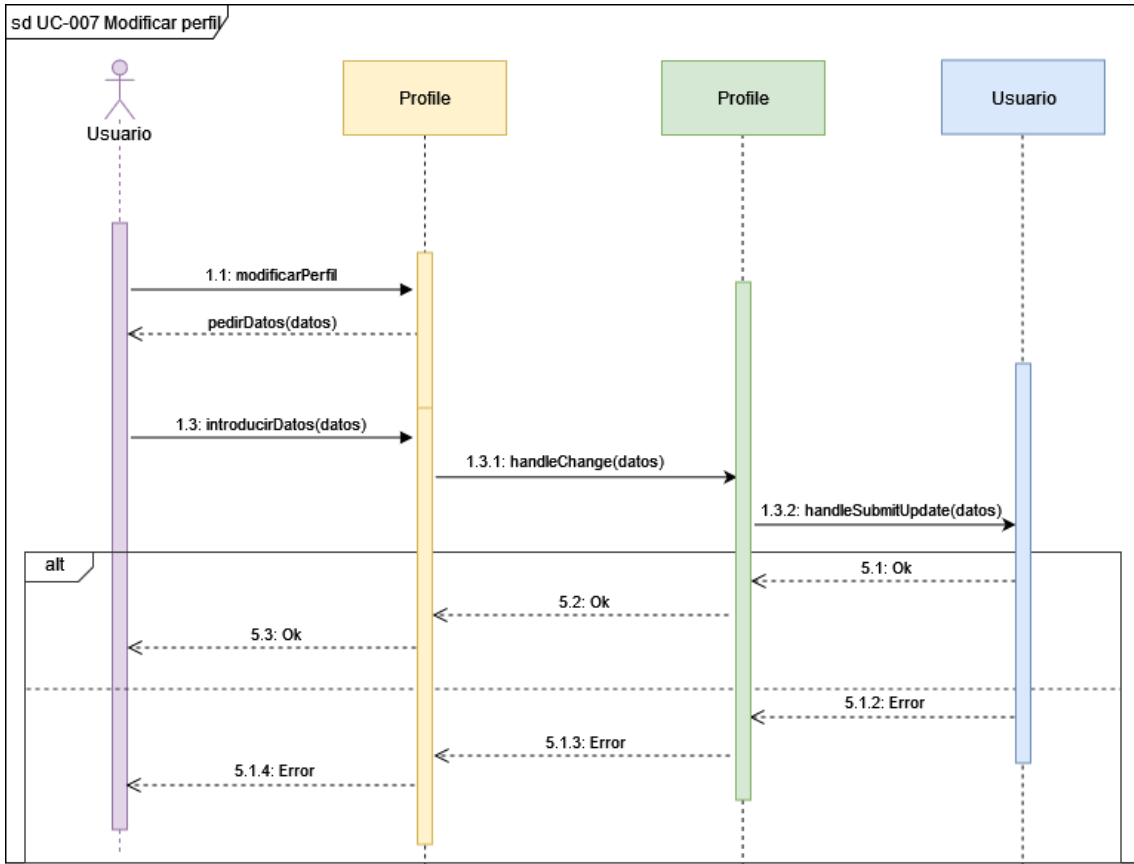


Figura 14: Diagrama de secuencia Modificar perfil

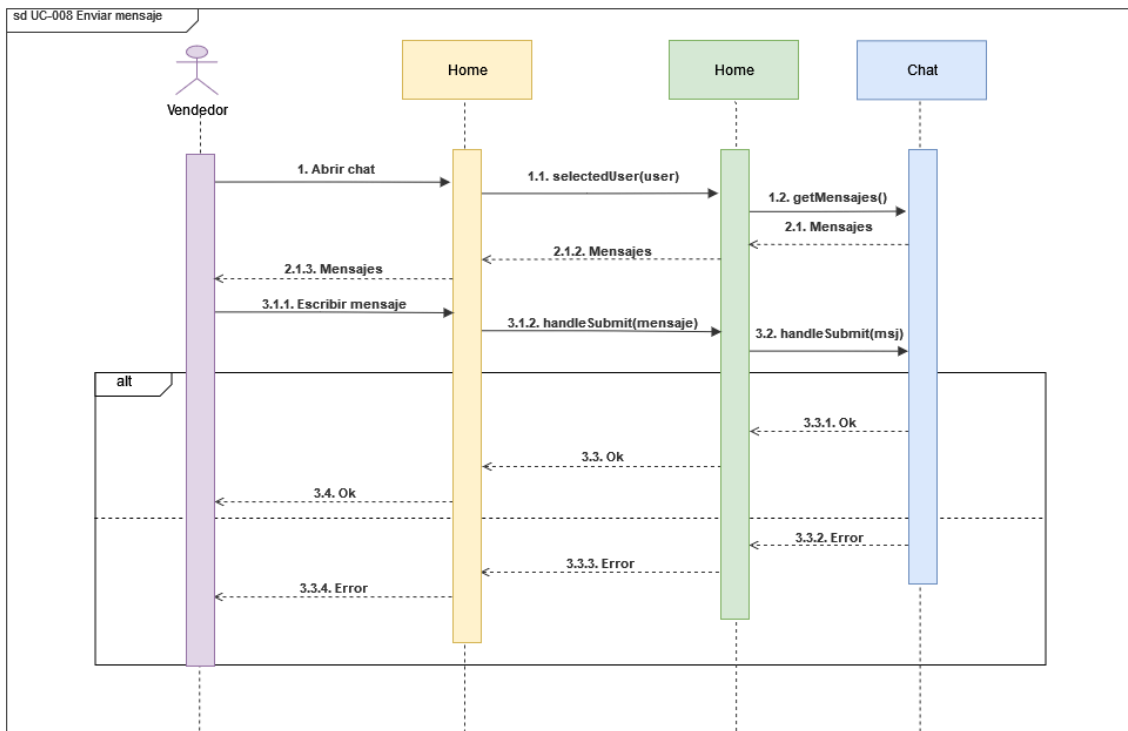


Figura 15: Diagrama de secuencia Enviar mensaje

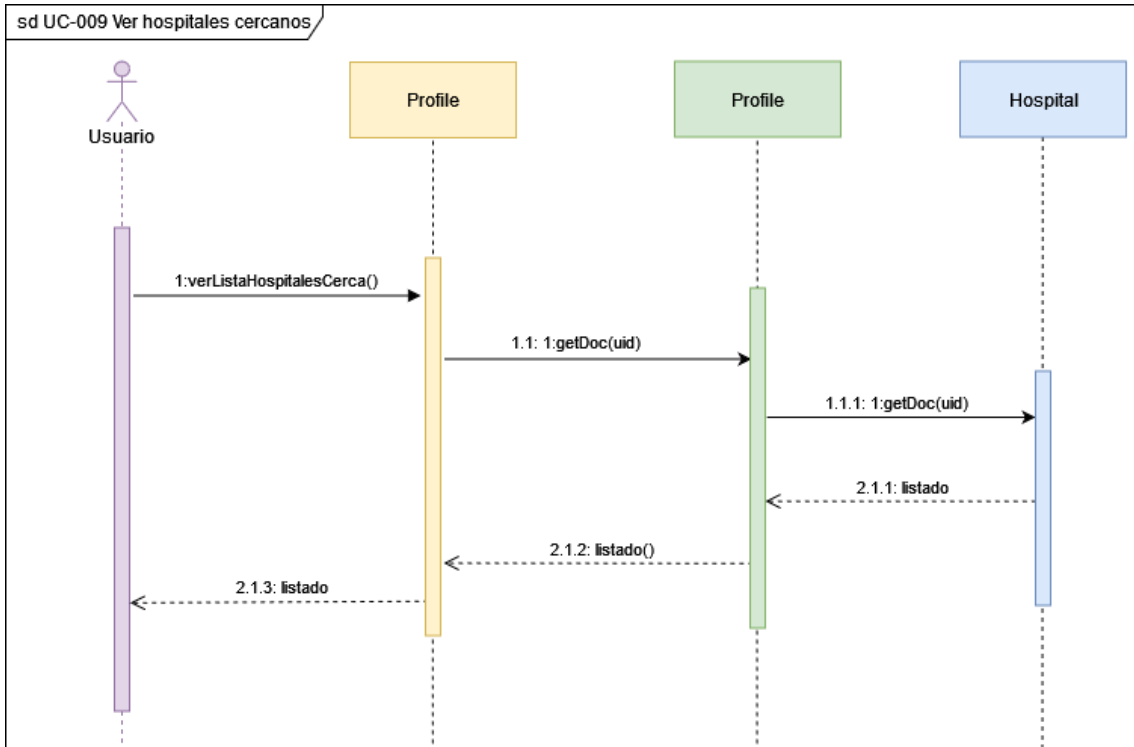


Figura 16: Diagrama de secuencia Ver hospitales cercanos

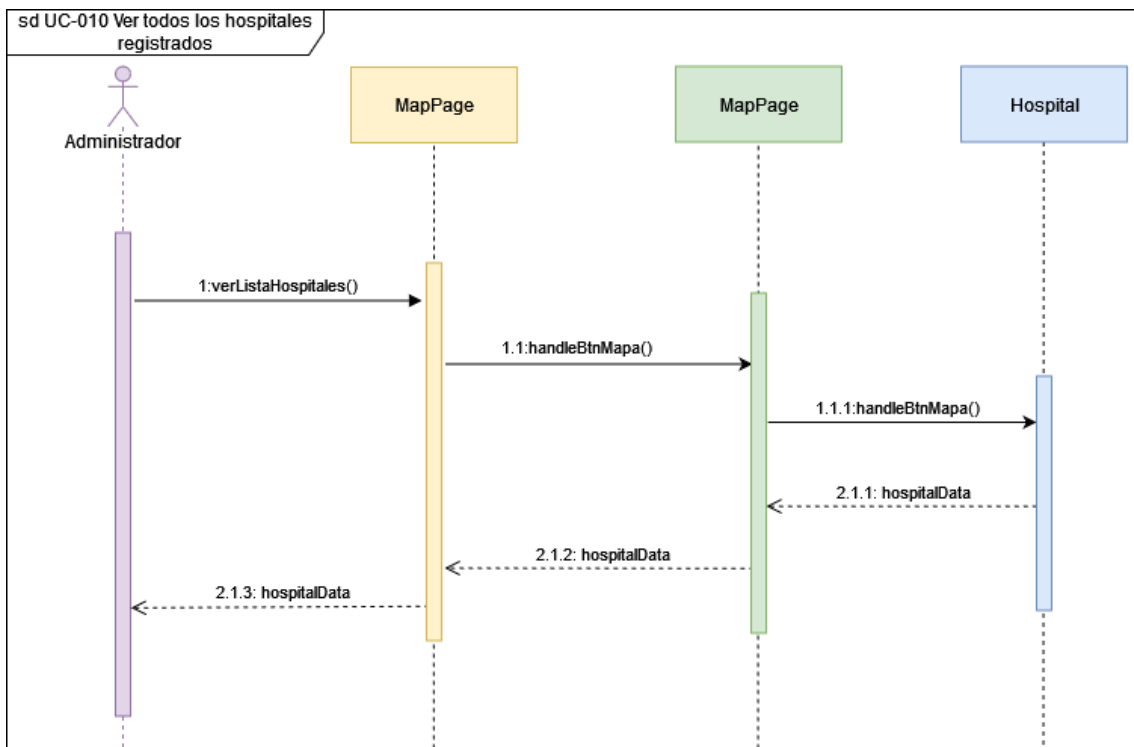


Figura 17: Diagrama de secuencia Ver todos los hospitales registrados

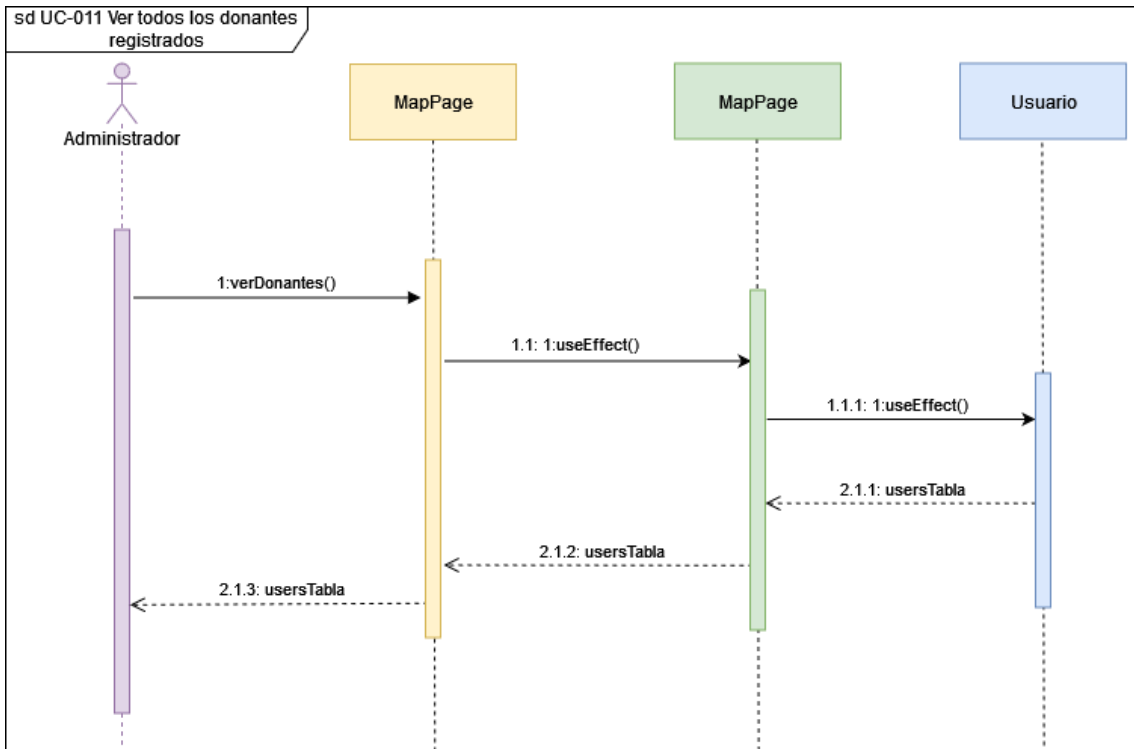


Figura 18: Diagrama de secuencia Ver todos los donantes registrados

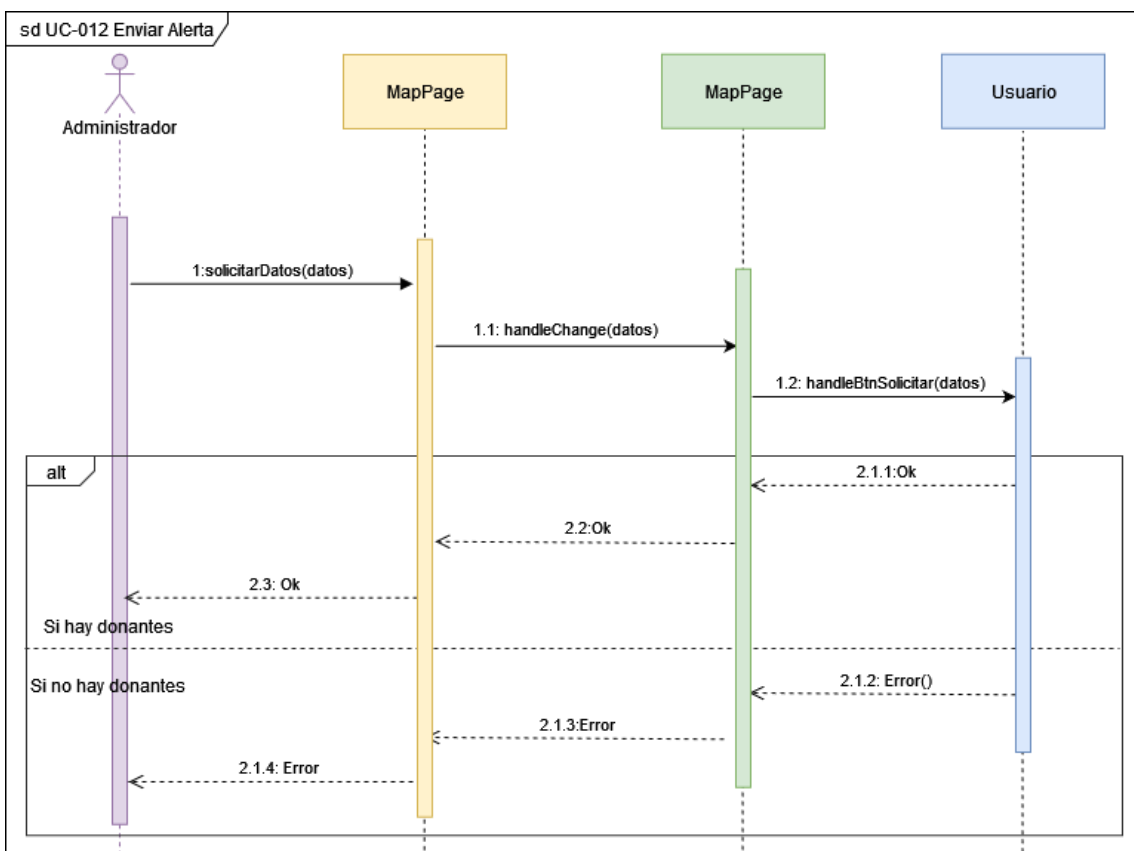


Figura 19: Diagrama de secuencia Enviar alerta

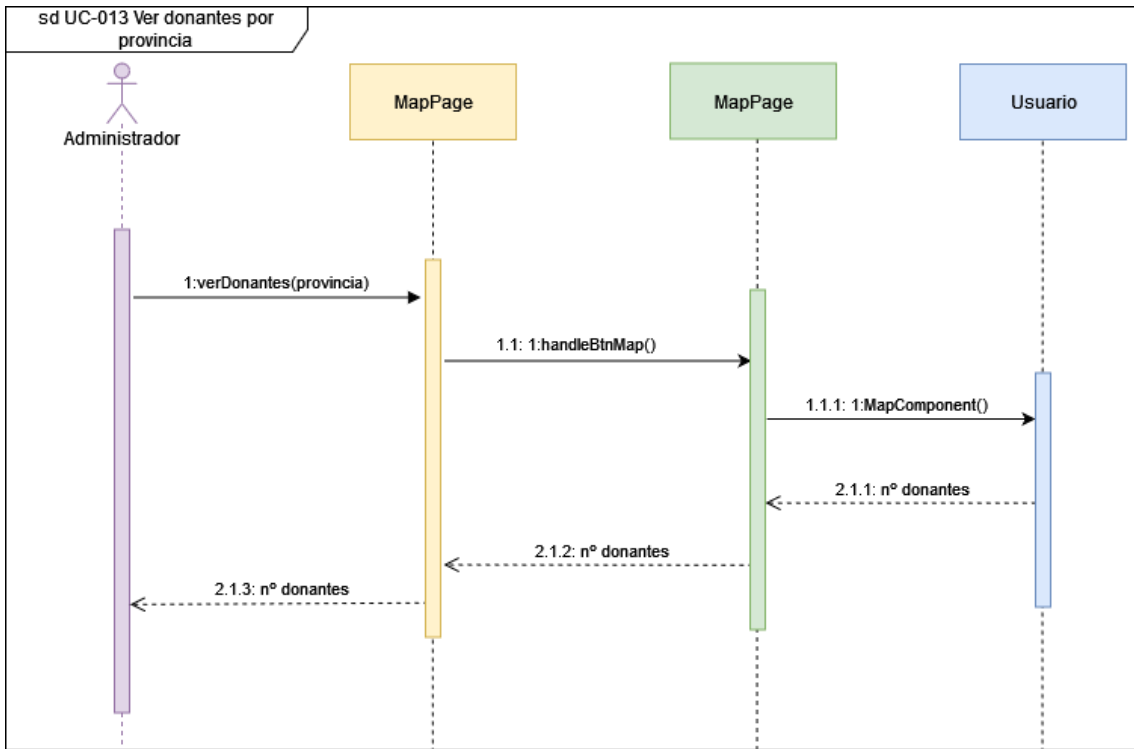


Figura 20: Diagrama de secuencia Ver donantes por provincia

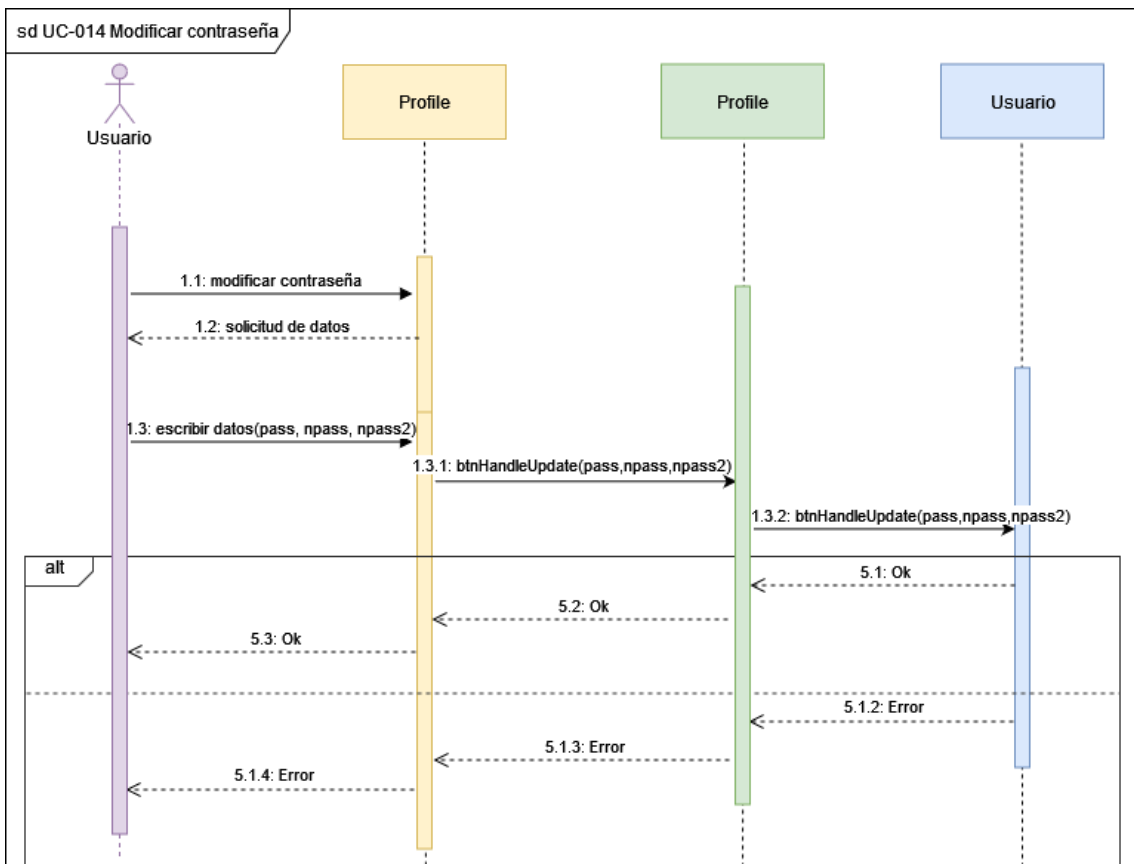


Figura 21: Diagrama de secuencia Modificar contraseña

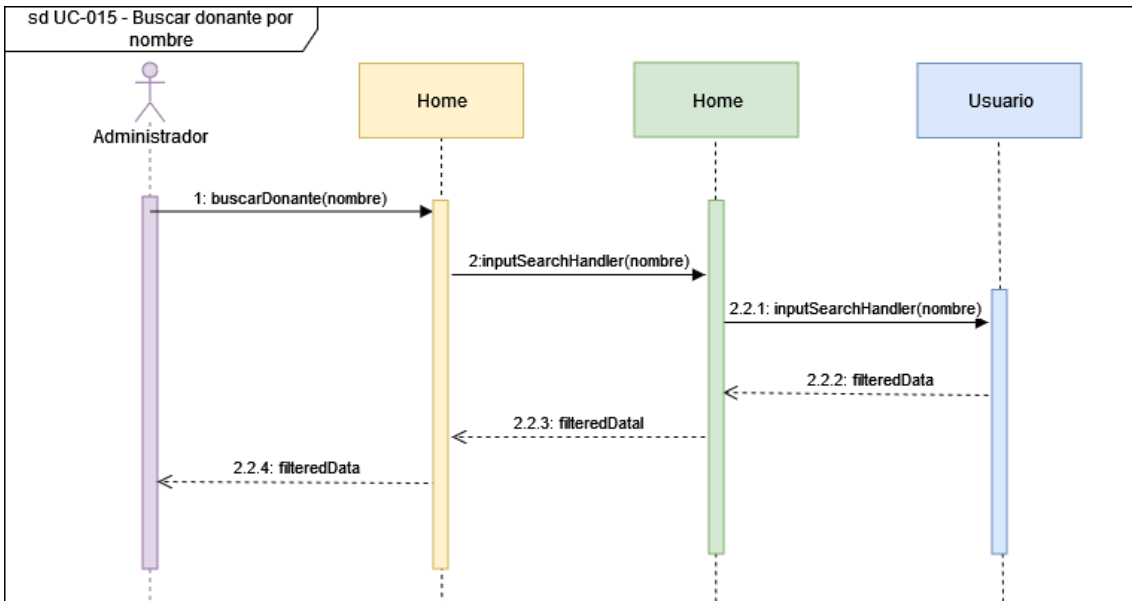


Figura 22: Diagrama de secuencia Buscar donante por nombre

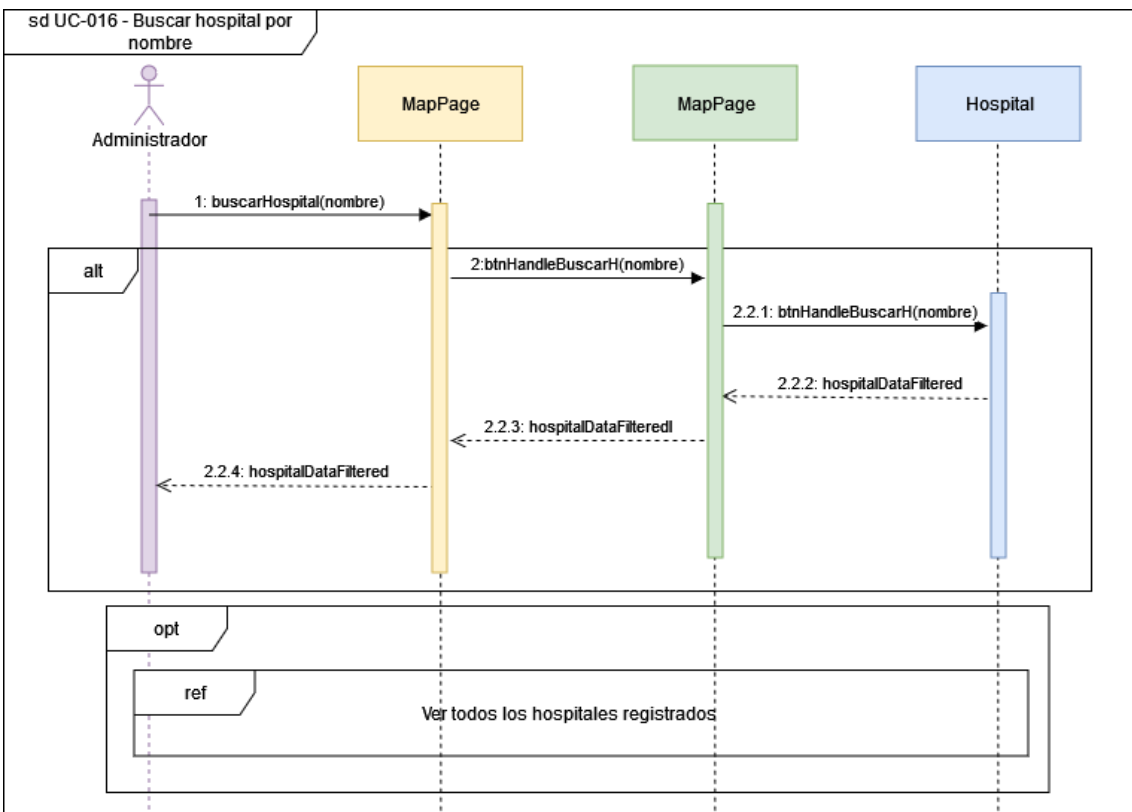


Figura 23: Diagrama de secuencia Buscar hospital por nombre

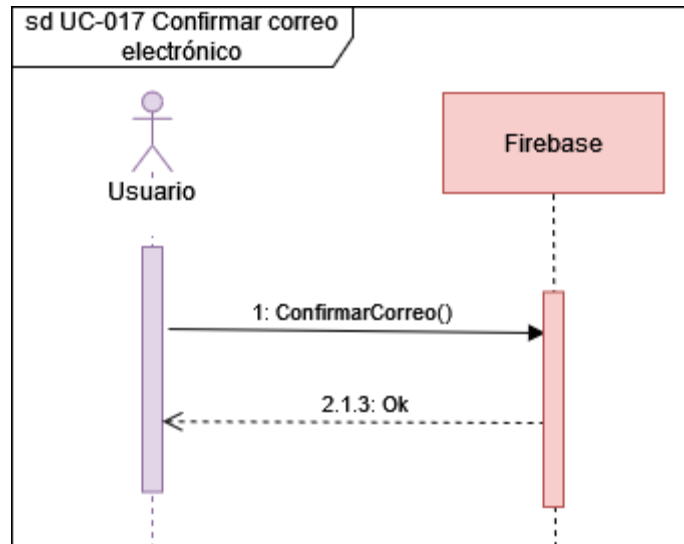


Figura 24: Diagrama de secuencia Confirmar correo electrónico

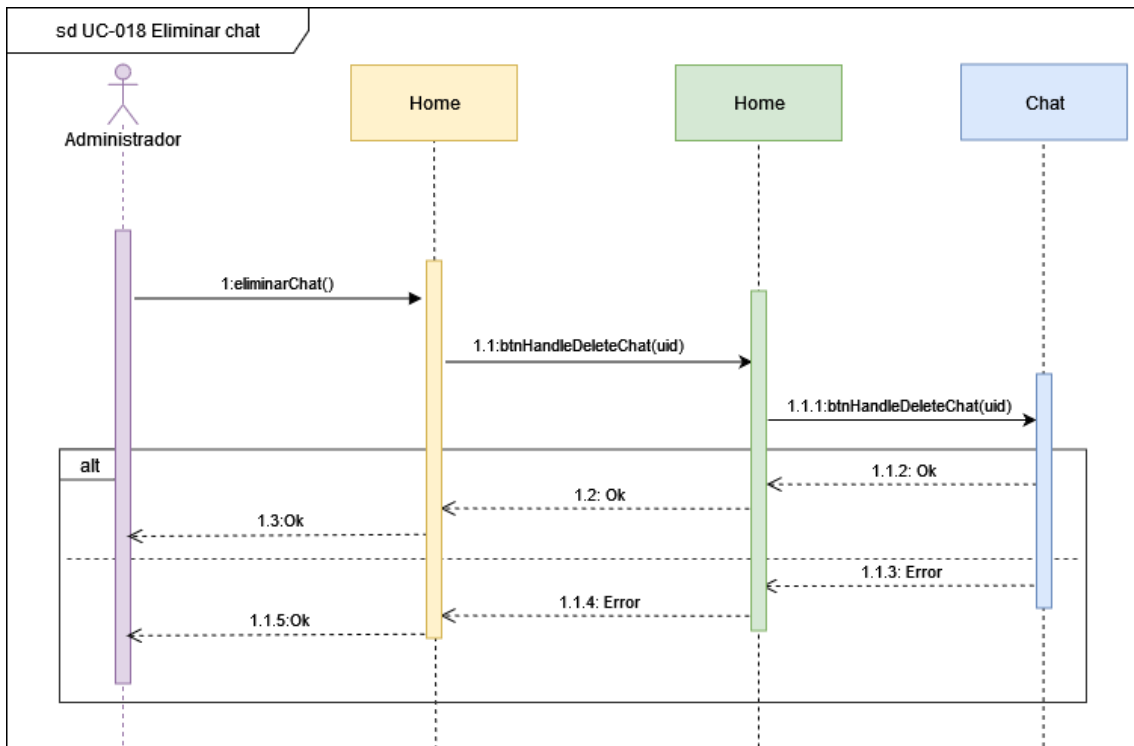


Figura 25: Diagrama de secuencia Eliminar chat

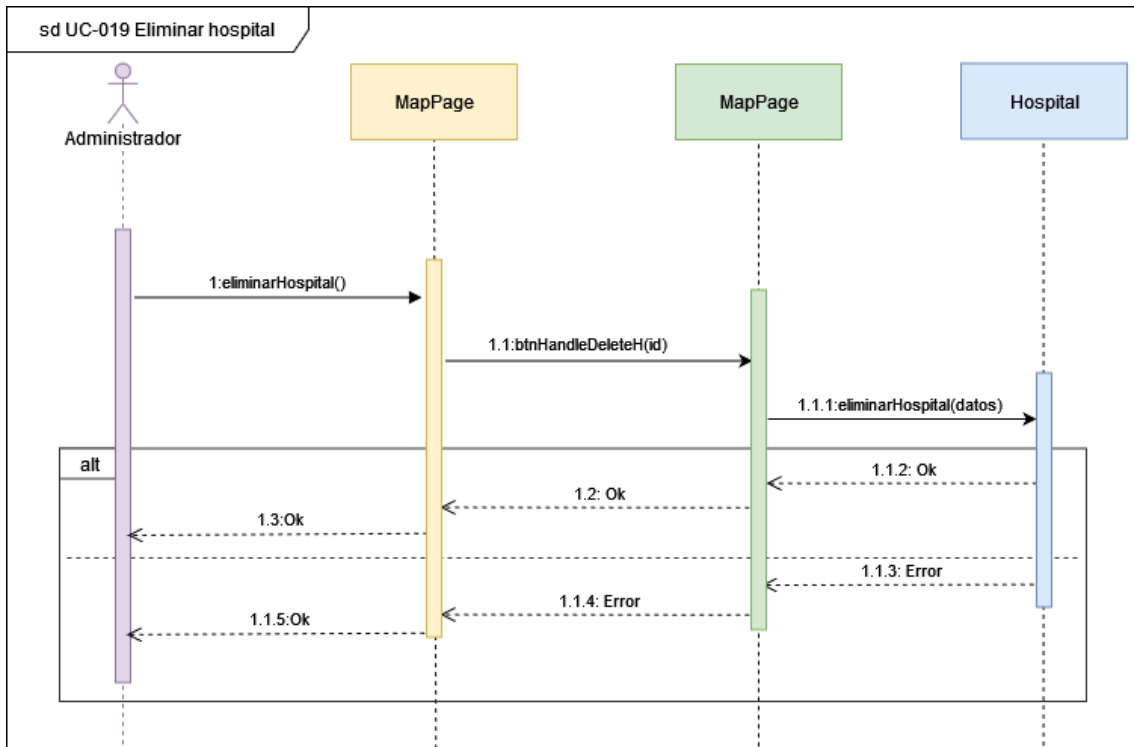


Figura 26: Diagrama de secuencia Eliminar hospital

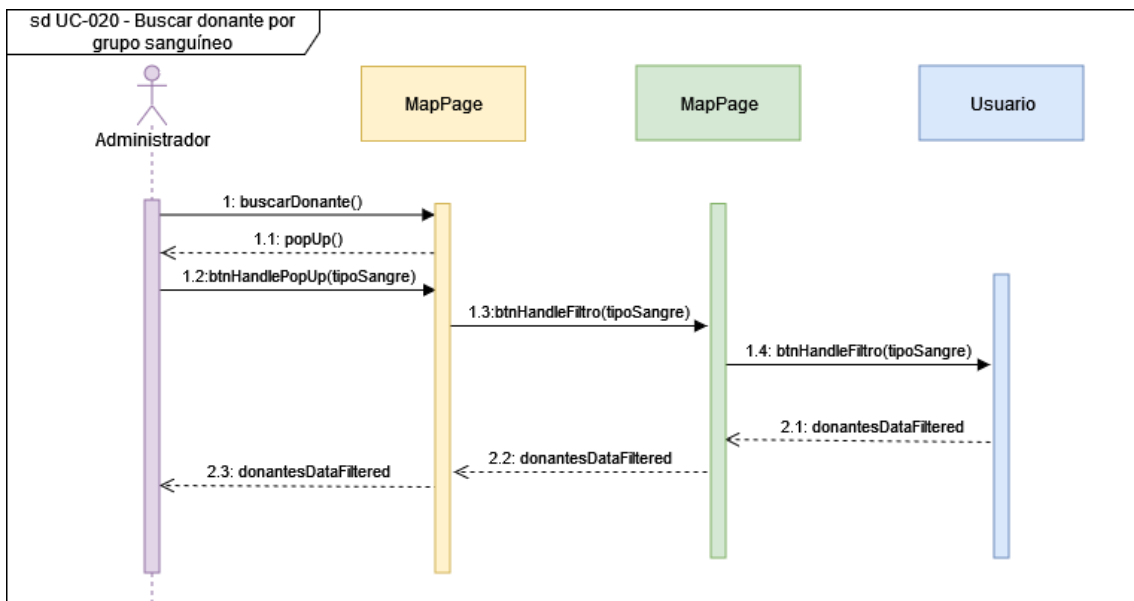


Figura 27: Diagrama de secuencia Buscar donante por grupo sanguíneo

7. Prototipo digital.

En este apartado mostraremos las interfaces de usuario de las que dispondrá el sistema. Al ser una aplicación apta para todo tipo de usuario debemos prestar mucha atención al diseño de las interfaces, diseñándolas de forma que sean intuitivas. También se debe hacer hincapié en que sean usables desde todo tipo de dispositivos [4].

7.1. Registro/Inicio de sesión.

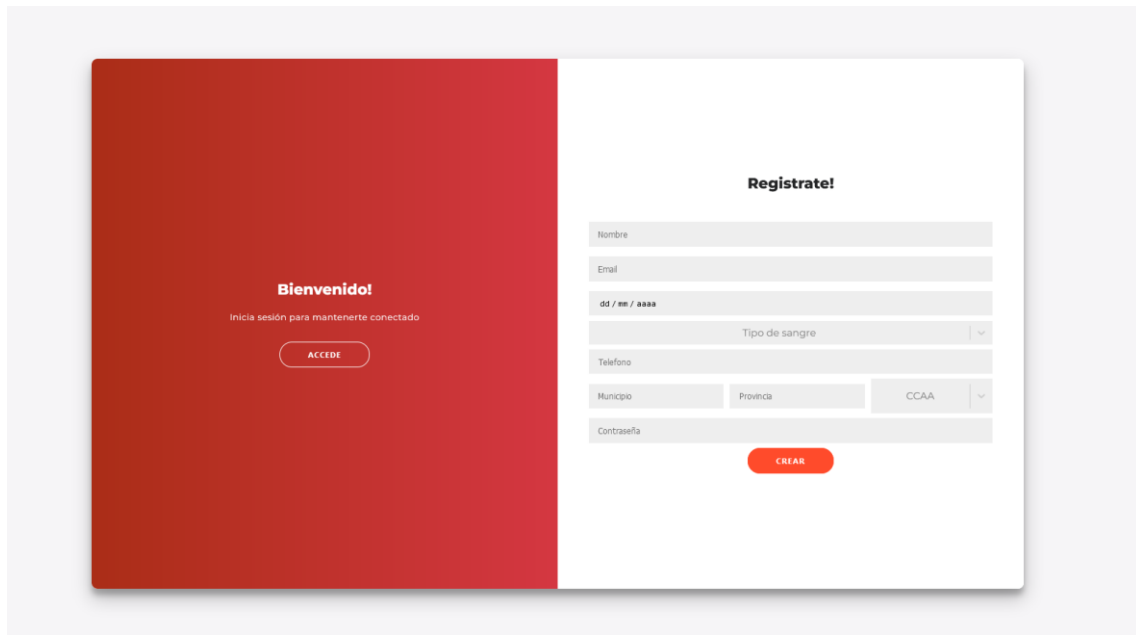


Figura 28: Prototipo digital Registro

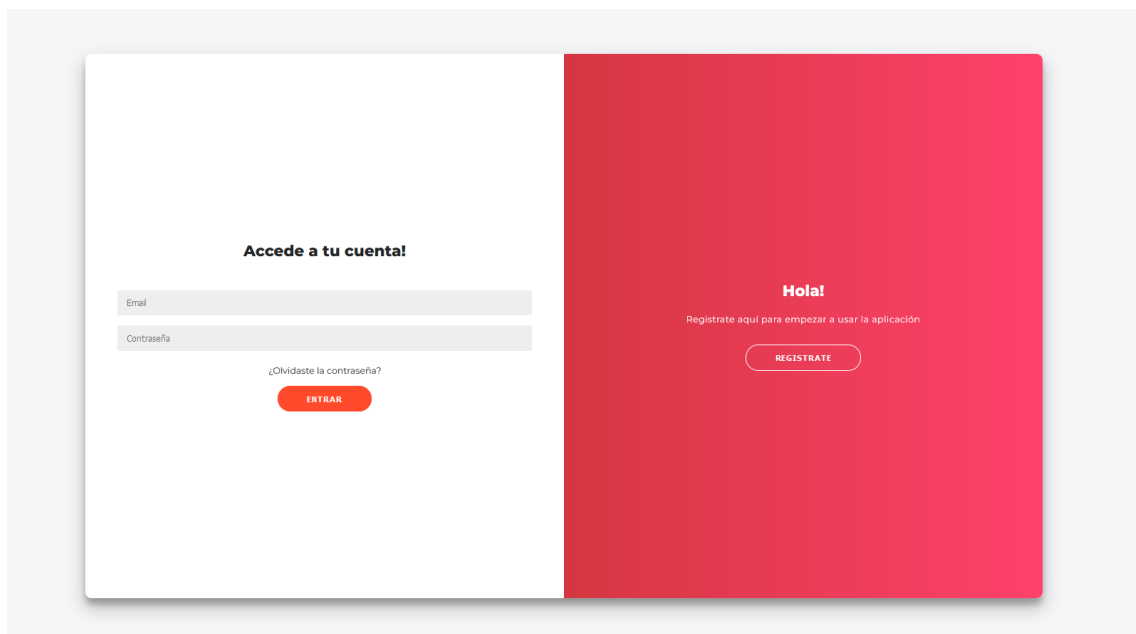


Figura 29: Prototipo digital Inicio sesión

7.2. Chat

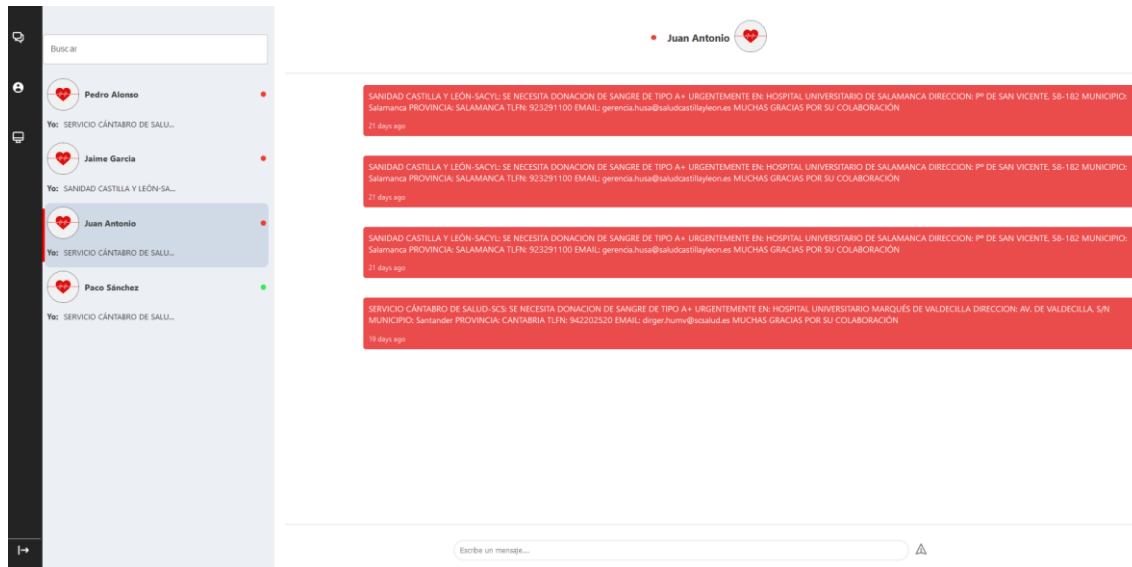


Figura 30: Prototipo digital Chat

7.3. Perfil

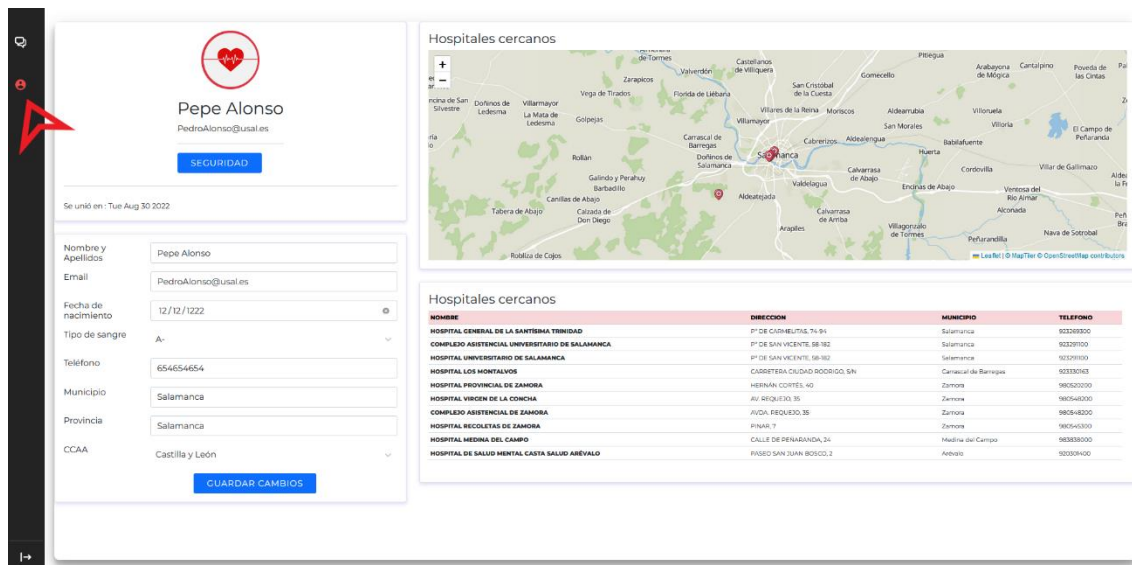


Figura 31: Prototipo digital Perfil

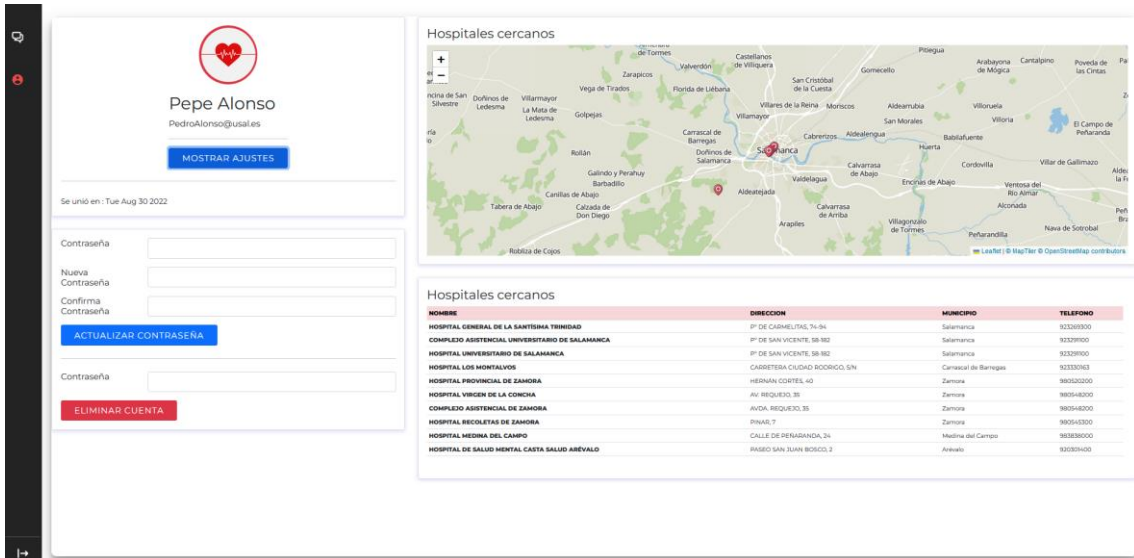


Figura 32: Prototipo digital Perfil modificar contraseña

7.4. Control administrador

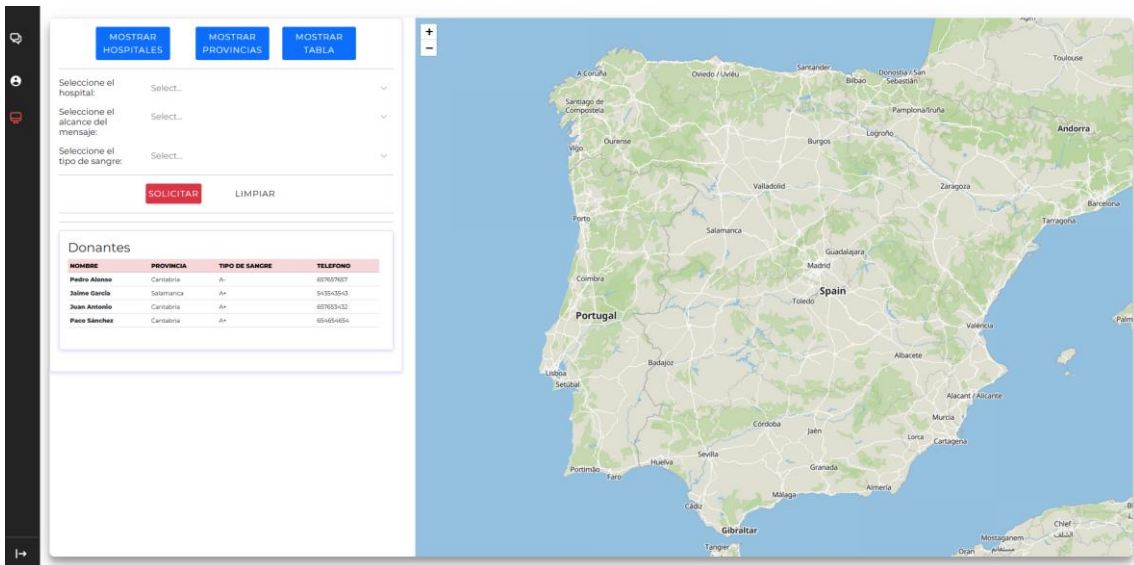


Figura 33: Prototipo digital Administrador base

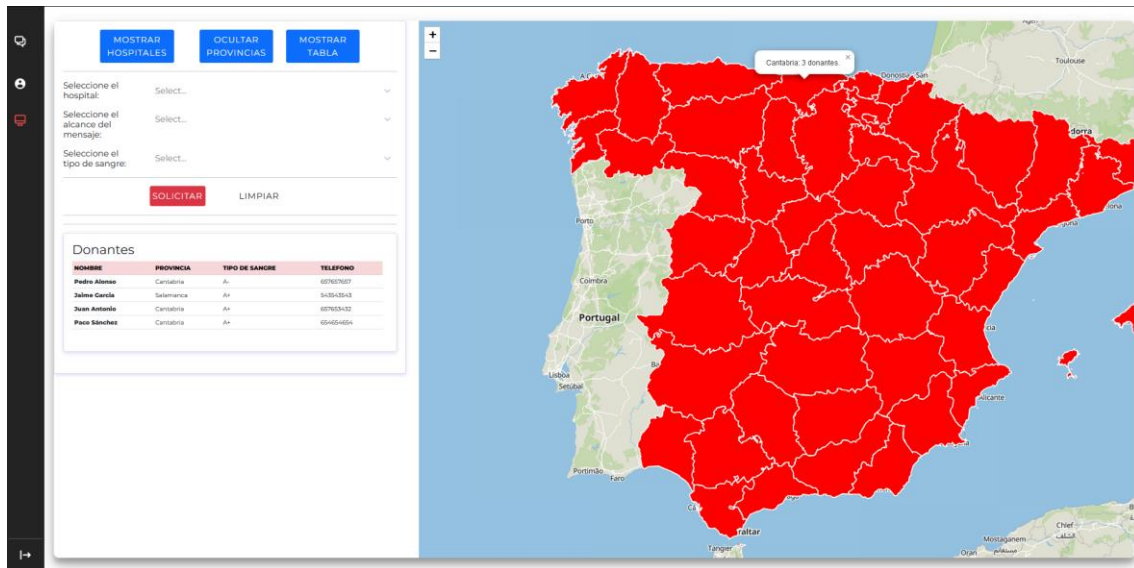


Figura 34: Prototipo digital Administrador ver donantes por provincia

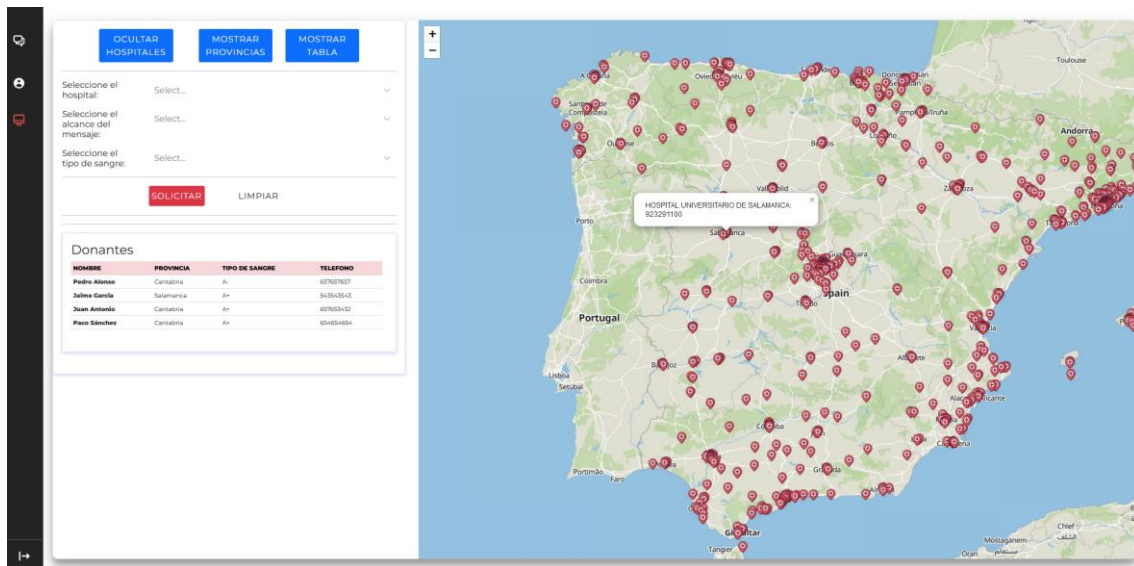


Figura 35: Prototipo digital Administrador ver hospitales

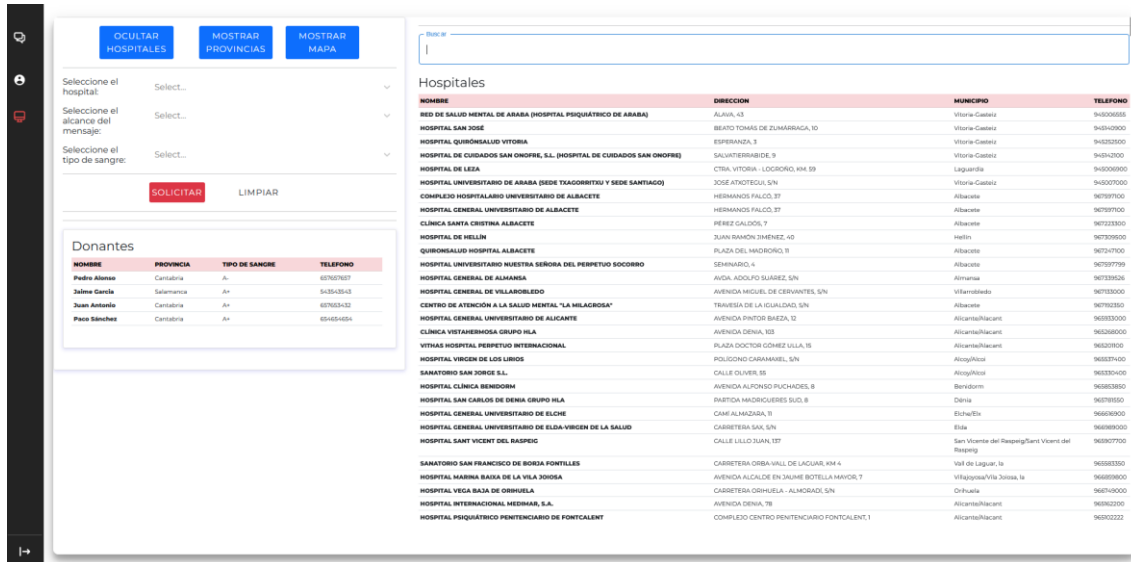


Figura 36: Prototipo digital Administrador ver hospitales tabla

8. Modelo de despliegue.

A continuación, se desarrollará el modelo de despliegue que se usará mostrando los componentes que forman el sistema y sus relaciones.

La aplicación web se desplegará sobre un servidor de hosting fiable, para así garantizar al usuario la disponibilidad del sistema, a la aplicación se accederá a través de un navegador de internet. La base de datos que utiliza el sistema es Firebase y a su vez utiliza la api de OpenRouteService.

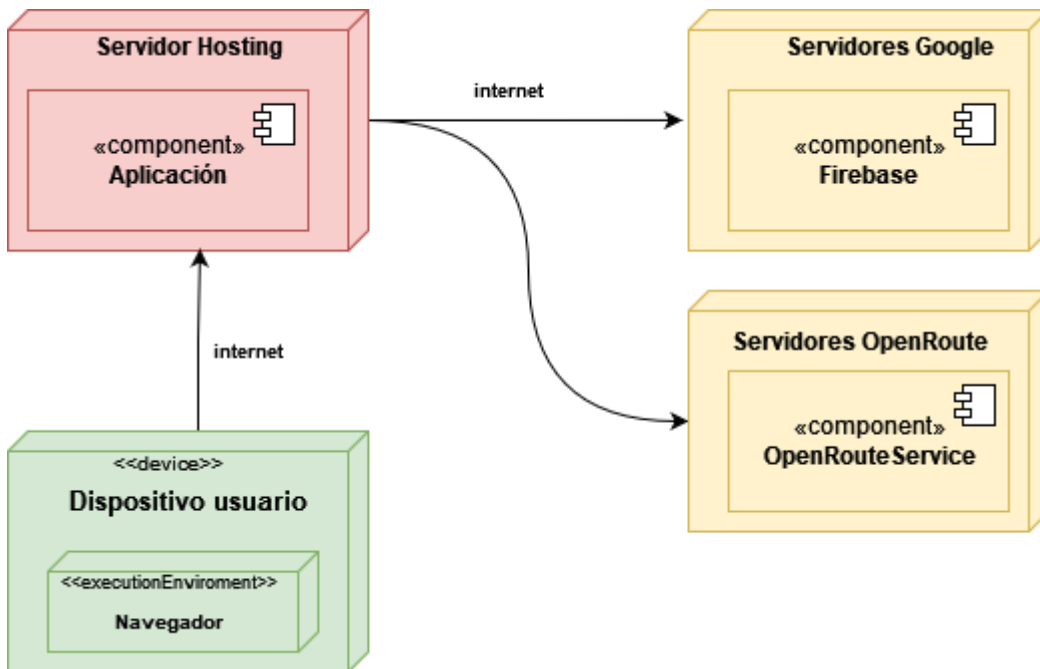


Figura 37: Modelo de despliegue

9. Bibliografía.

[1] María N. Moreno García, “Transparencias Ingeniería del software II, Tema 1 – Diseño del software” [2021]

[2] «Diagrams.net» Enlace: <https://www.diagrams.net> [2005]

[3] María N. Moreno García, Francisco J. García Peñalvo, “Transparencias ingeniería del software II, Tema 4 – Patrones” [2021]

[4] R. Therón, "Transparencias de Interacción Persona Ordenador, Tema 4 - Diseño centrado en el usuario" [2020]

ANEXO V:

Documentación Técnica de Programación

Trabajo Fin de Grado Ingeniería Informática

Septiembre 2022



**VNiVERSIDAD
D SALAMANCA**

Autor:

Pedro Luis Alonso Díez

Tutores:

Luis Augusto Silva

Gabriel Villarrubia González

Juan Francisco de Paz Santana

Tabla de contenido

1. Introducción.	1
2. Lenguajes de programación.	2
NODEJS.	2
REACT.	2
HTML, CSS, JAVASCRIPT.	2
BOOTSTRAP.	2
LEAFLET.	2
GEOJSON.	2
FIREBASE.	2
3. Estructura del sistema.	2
Componentes.	3
Componentes Data y GeoJSON.	4
Componentes Images y SVG.	4
Pages.	5
4. Especificación de la documentación.	5
5. Bibliografía.	6

Índice de figuras

Figura 1: Estructura del sistema.....	3
Figura 2: Estructura del sistema - Componentes	4
Figura 3: Estructura del sistema – Componentes Data.....	4
Figura 4: Estructura del sistema – Componentes Images/SVG	4
Figura 5: Estructura del sistema – Pages.....	5
Figura 6: Esquema de la documentación	5
Figura 7: Documentación componente ejemplo	6

1. Introducción.

En este anexo se comentará la estructura que tiene el proyecto a nivel de código, su lenguaje de programación con el objetivo de facilitar la escalabilidad de la aplicación y sus respectivas labores de mantenimiento.

2. Lenguajes de programación.

NODEJS.

Se ha utilizado el entorno Nodejs para la ejecución de la aplicación en entornos de desarrollo. Debido a que permite que el código JavaScript se ejecute en el servidor, a parte, su gestor de paquetes NPM es muy útil para implementar funcionalidades.

REACT.

Para la aplicación se ha utilizado React Native como framework, este framework está construido sobre la librería React. Se ha elegido esta opción debido a que es multiplataforma y ofrece un buen grado de escalabilidad a la aplicación [1].

HTML, CSS, JAVASCRIPT.

La lógica de la aplicación está escrita en JavaScript, y su maquetado, es decir, la apariencia de la aplicación se ha realizado con HTML y CSS.

BOOTSTRAP.

También nos ayudamos de Bootstrap para diseñar ciertas interfaces de un modo más sencillo, sin embargo, para aquellas que queremos algo más específico se realiza con HTML y CSS [2].

LEAFLET.

Para los mapas se ha utilizado la biblioteca Leaflet debido a que dispone de una interfaz amigable para dispositivos móviles, es open source, y ofrece un gran grado de personalización [3].

GEOJSON.

Para representar los hospitales en los mapas se ha usado archivos geojson, ya que implementan las coordenadas de los hospitales y se pueden implementar en los mapas de Leaflet. Los datos se han sacado de la página de datos del gobierno [4].

FIREBASE.

Para la base de datos se ha utilizado el servicio de Google Firebase, debido a que tiene un plan gratuito, y permite a los datos sincronizarse en tiempo real [5].

3. Estructura del sistema.

La estructura general de la aplicación se representa en la Figura 1. Posteriormente iremos desglosando las carpetas.

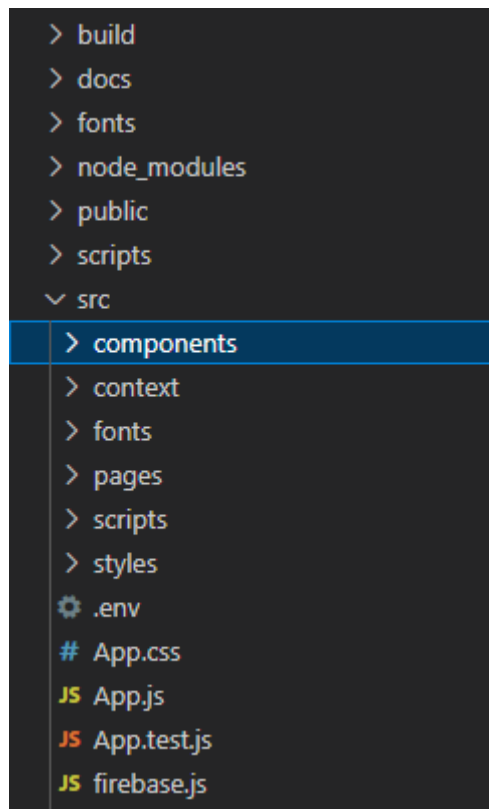


Figura 1: Estructura del sistema

Componentes.

En esta carpeta se guardan todos los componentes que usa el sistema, posteriormente en la documentación se detalla la utilidad de cada componente.

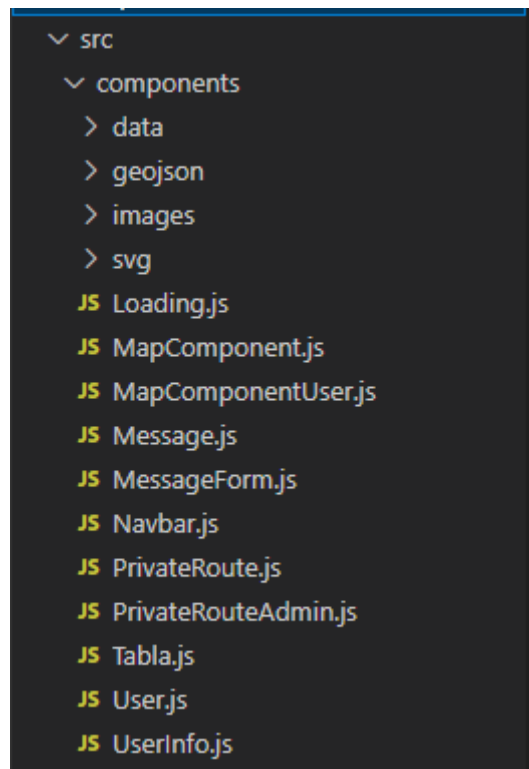
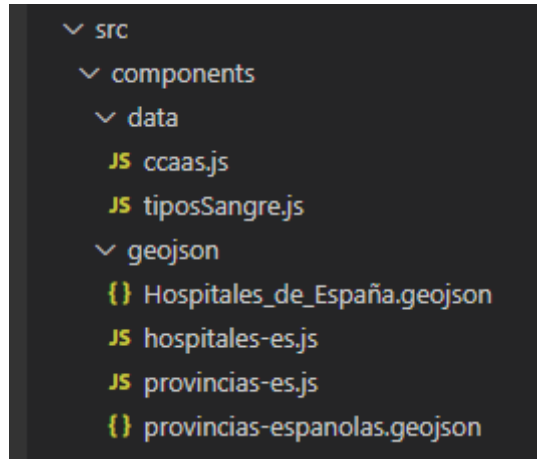


Figura 2: Estructura del sistema - Componentes

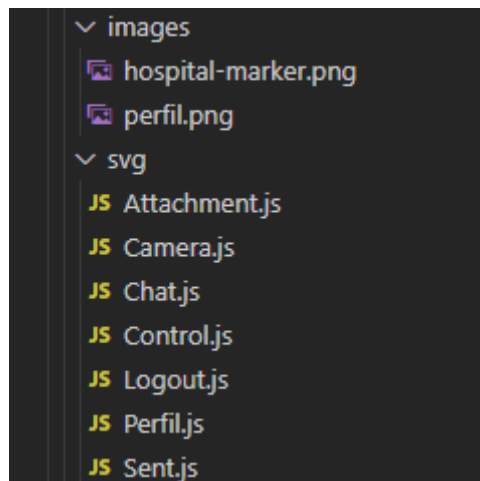
Componentes Data y GeoJSON.

En esta carpeta se guardan los archivos “.geojson” utilizados en el sistema, también se guardan estructuras de datos para los tipos de sangre y las CCAA de España.

*Figura 3: Estructura del sistema – Componentes Data*

Componentes Images y SVG.

En estas carpetas se guardan las imágenes usadas en el sistema y los SVG utilizados para iconos.

*Figura 4: Estructura del sistema – Componentes Images/SVG*

Pages.

En esta carpeta se guardan las distintas páginas funcionales que tiene la web, son las interfaces de usuario. Y según se introduzca la URL la página de inicio será Login/Register.

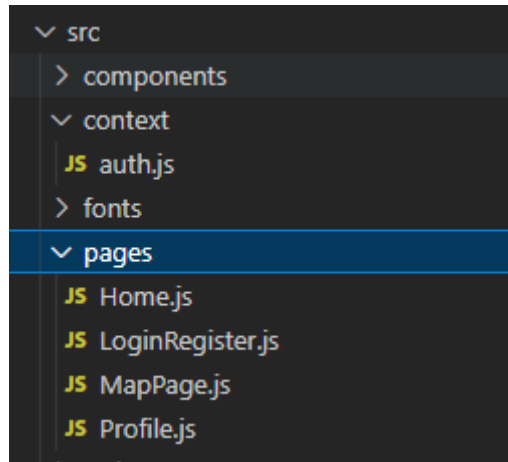


Figura 5: Estructura del sistema – Pages

4. Especificación de la documentación.

Para generar la documentación se ha utilizado la herramienta JSDOC. Se trata de una sintaxis para agregar documentación de la API al código fuente de JavaScript.

Esta documentación se podrá encontrar en /docs y su esquema será el siguiente.

Module: Chat

Página del chat
Source: [pages/Home.js, line 21](#)

Methods

(inner) `getMensajes(id)`
Método para extraer los mensajes de un determinado chat

Parameters:

Name	Type	Description
id	String	Id del chat del que queremos extraer los mensajes

Source: [pages/Home.js, line 115](#)

(inner) `handleSubmit(e)`
Handler cuando pulsamos el botón de enviar mensaje

Parameters:

Name	Type	Description
e	event	Evento

Home

Modules

- Chat
- components/Loading
- components/MapComponent
- components/MapComponentUser
- components/Message
- components/MessageForm
- components/Navbar
- components/PrivateRoute
- components/PrivateRouteAdmin
- components/Tabla
- components/User
- components/UserInfo
- Login/Register
- pages/MapPage
- pages/Profile

Global

- ccaas
- hospitalesData
- provinciasData
- tiposSangre

Figura 6: Esquema de la documentación

Para navegar por la documentación simplemente tendremos que pulsar sobre el elemento que queremos leer en la parte derecha de la pantalla.

Module: components/MessageForm

Componente MensajeForm, componente donde el usuario escribe el mensaje

Parameters:

Name	Type	Description
handleSubmit	function	Funcion para manejar el submit - La manda desde Home.js
text	String	El texto que se escribe es un bind para poder actualizar el value del input
setText	function	Setter del text

Source: [components/MessageForm.js, line 3](#)

Home

Modules

Chat
 components/Loading
 components/MapComponent
 components/MapComponentUser
 components/MessageForm
 components/MessageForm
 components/Navbar
 components/PrivateRoute
 components/PrivateRouteAdmin
 components/Tabla
 components/User
 components/UserInfo
 Login/Register
 pages/MapPage
 pages/Profile
 Global

Figura 7: Documentación componente ejemplo

5. Bibliografía.

- [1] «React» Disponible: [<https://es.reactjs.org/>], [2013]
- [2] «Bootstrap» Disponible: [<https://getbootstrap.com/>], [2011]
- [3] «Leaflet» Disponible: [<https://leafletjs.com/>], [2008]
- [4] Ministerio de Asuntos Económicos y Transformación Digital, [<https://datos.gob.es/>] [2022]
- [4] «Firebase» Disponible: [<https://firebase.google.com>] [2014]

ANEXO VI:

Manual de Usuario

Trabajo Fin de Grado Ingeniería Informática

Septiembre 2022



**VNiVERSIDAD
D SALAMANCA**

Autor:

Pedro Luis Alonso Díez

Tutores:

Luis Augusto Silva

Gabriel Villarrubia González

Juan Francisco de Paz Santana

Tabla de contenido

1. Introducción.	1
2. Registro.	2
3. Inicio de sesión.	2
4. Recuperar contraseña.	3
5. Cerrar sesión.	4
6. Chat.	4
7. Perfil.	7
7.1. Modificar información.	7
7.2. Modificar contraseña.	8
7.3. Eliminar cuenta.	9
7.4. Ver hospitales en el mapa.	9
8. Administrador.	10
8.1. Vista administrador.	10
8.2. Buscar donante por nombre en el chat.	10
8.3. Acceder a la Página de control.	11
8.4 Ver todos los hospitales.	11
8.5 Ver donantes por provincias.	12
8.6 Ver información de los hospitales registrados.	12
8.7 Ver donantes registrados.	13
8.8 Enviar alerta.	14
8.9 Limpiar datos.	16
9. Bibliografía.	16

Índice de figuras

Figura 1: Registro	2
Figura 2: Inicio de sesión	2
Figura 3: Recuperar contraseña	3
Figura 4: Correo electrónico de recuperación	3
Figura 5: Cerrar sesión	4
Figura 6: Chat	4
Figura 7: Chat con mensaje enviado	5
Figura 8: Mensaje sin leer	7
Figura 9: Perfil	7
Figura 10: Perfil Modificar información	8
Figura 11: Cambiar contraseña parte 1.....	8
Figura 12: Cambiar contraseña parte 2.....	8
Figura 13: Contraseña cambiada.....	9
Figura 15: Eliminar cuenta.....	9
Figura 16: Ver hospitales en el mapa	10
Figura 17: Chat Administrador	10
Figura 18: Búsqueda en el chat por nombre.....	11
Figura 19: Página de control.....	11
Figura 20: Página de control – botón mostrar hospitales.....	12
Figura 21: Ver donantes por provincia.....	12
Figura 22: Ver información todos los hospitales tabla.....	13
Figura 23: Filtrar hospitales.....	13
Figura 24: Ver donantes registrados	14
Figura 25: Campos a rellenar para enviar alerta.....	14
Figura 26: Alerta enviada con éxito.....	15
Figura 27: Ver donantes a los que se les ha enviado la alerta	15
Figura 28: Mensaje de alerta enviado.....	15

1. Introducción.

En este anexo se detallará una guía del funcionamiento del sistema orientado al usuario. En esta guía se detallarán todas las acciones posibles del sistema, explicando sus funcionalidades y modo de ejecutarlas para cubrir todas las opciones que el sistema ofrece al usuario en caso de duda.

2. Registro.

Para acceder a la aplicación se debe rellenar un formulario de registro, rellenando los datos personales que se piden y darle al botón de crear.

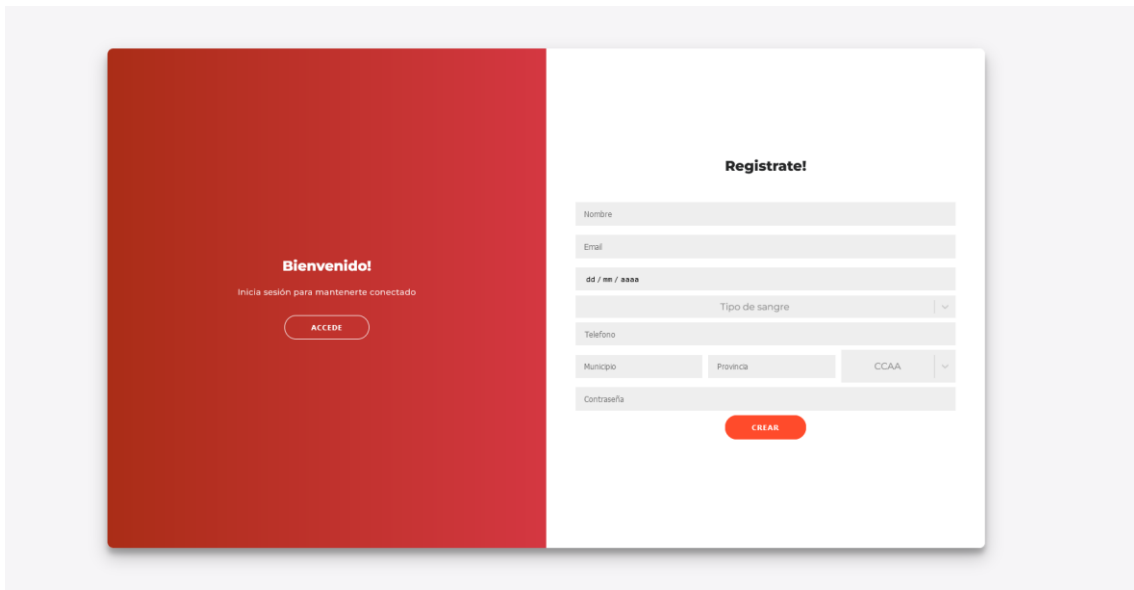


Figura 1: Registro

Una vez creada la cuenta ya se podrá entrar en el sistema. Para acceder al inicio de sesión deberemos pulsar el botón de “Accede”

3. Inicio de sesión.

Si ya se dispone de una cuenta se puede iniciar sesión. Para ello introduciremos el email y la contraseña.

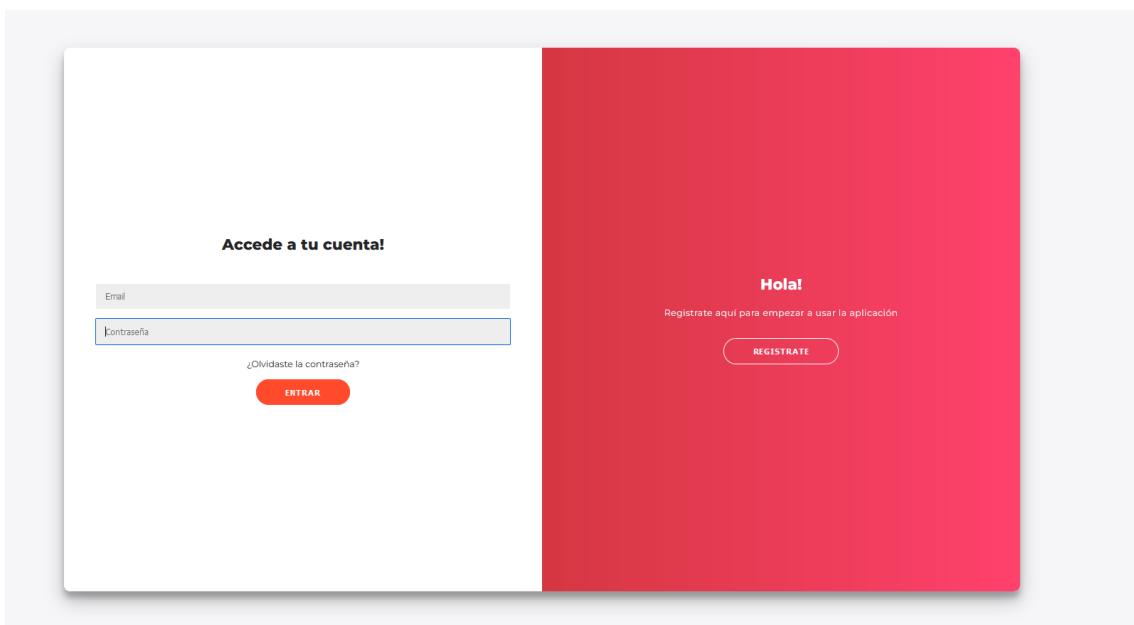


Figura 2: Inicio de sesión

4. Recuperar contraseña.

En caso de que nos hayamos olvidado de nuestra contraseña tenemos la opción de recuperarla. Para ello vamos a pulsar en “¿Olvidaste la contraseña?” e introducir el correo electrónico con el que nos hemos registrado.

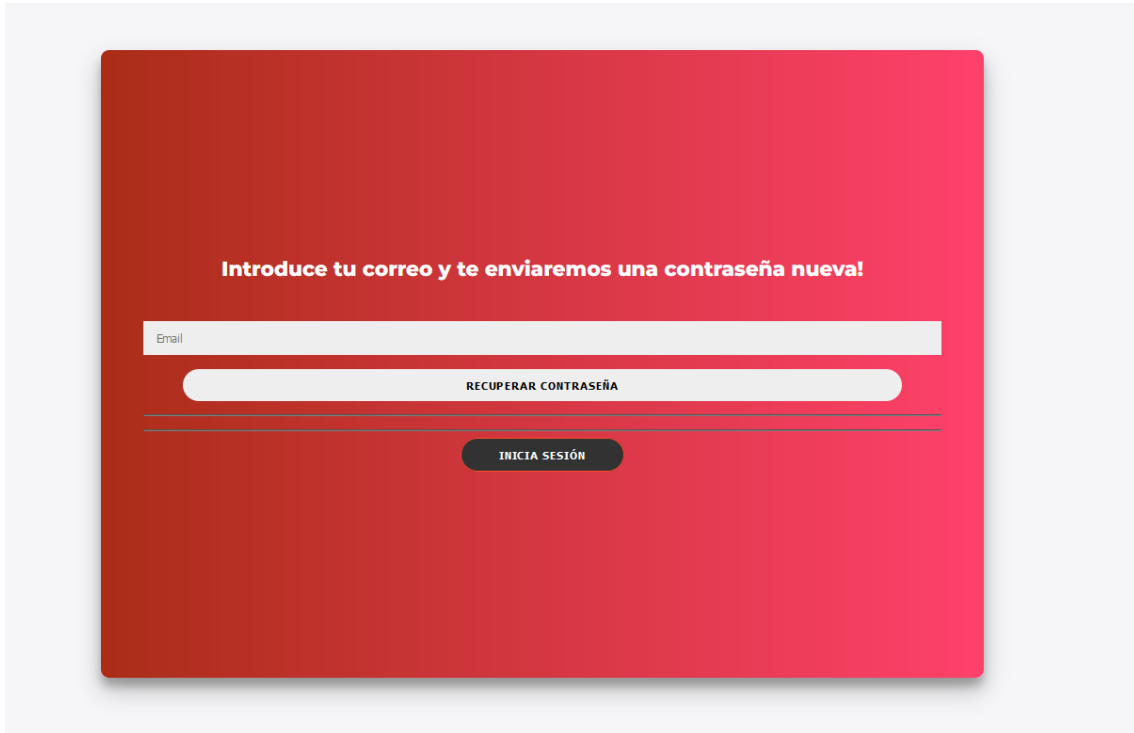


Figura 3: Recuperar contraseña

Una vez introducido el correo electrónico nos llegará un correo como el siguiente:

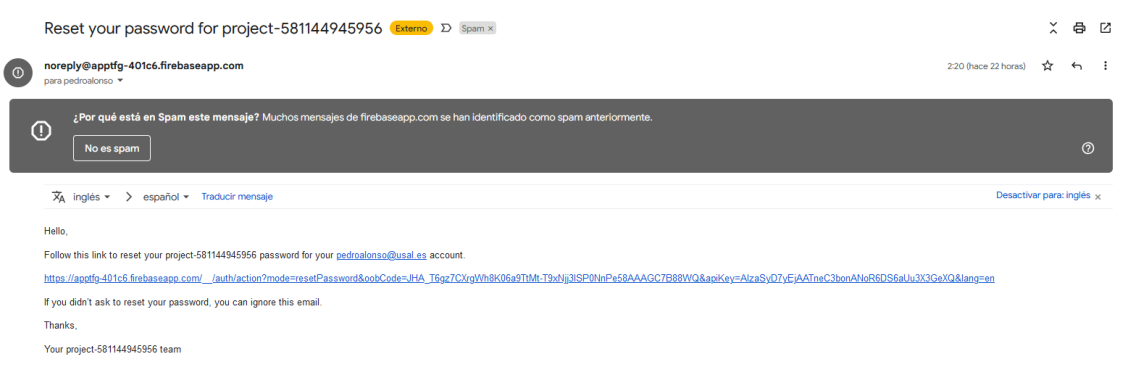


Figura 4: Correo electrónico de recuperación

Para volver al inicio de sesión solo deberemos pulsar el botón “Inicia sesión”

5. Cerrar sesión.

Para cerrar sesión habrá un botón en todas las páginas situados en la parte de abajo a la izquierda. Una vez pulsado se redirigirá a la página de inicio de sesión.

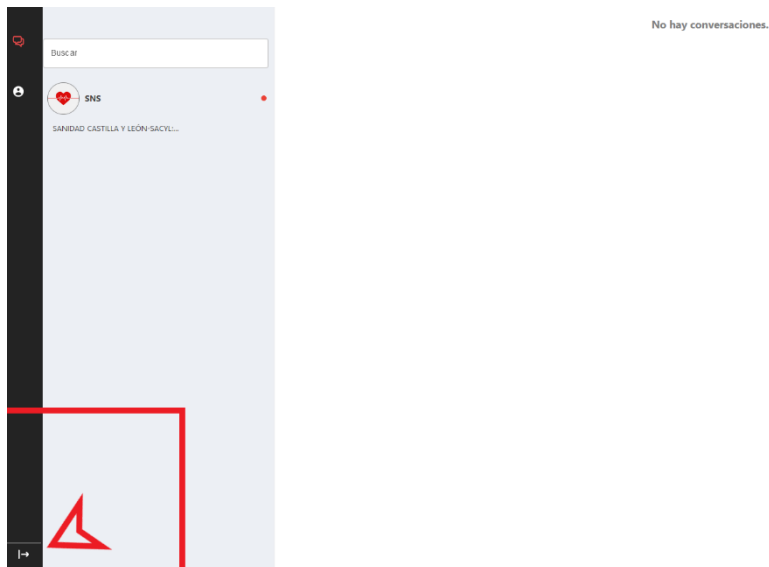


Figura 5: Cerrar sesión

6. Chat.

Si el usuario desea ponerse en contacto con el administrador (Servicio de Salud) para cualquier duda puede hacerlo mediante el chat interno. El usuario solo podrá chatear con el Servicio nacional de salud. Para hacerlo solo debe pulsar sobre su nombre. El punto rojo indica si el usuario está online u offline.

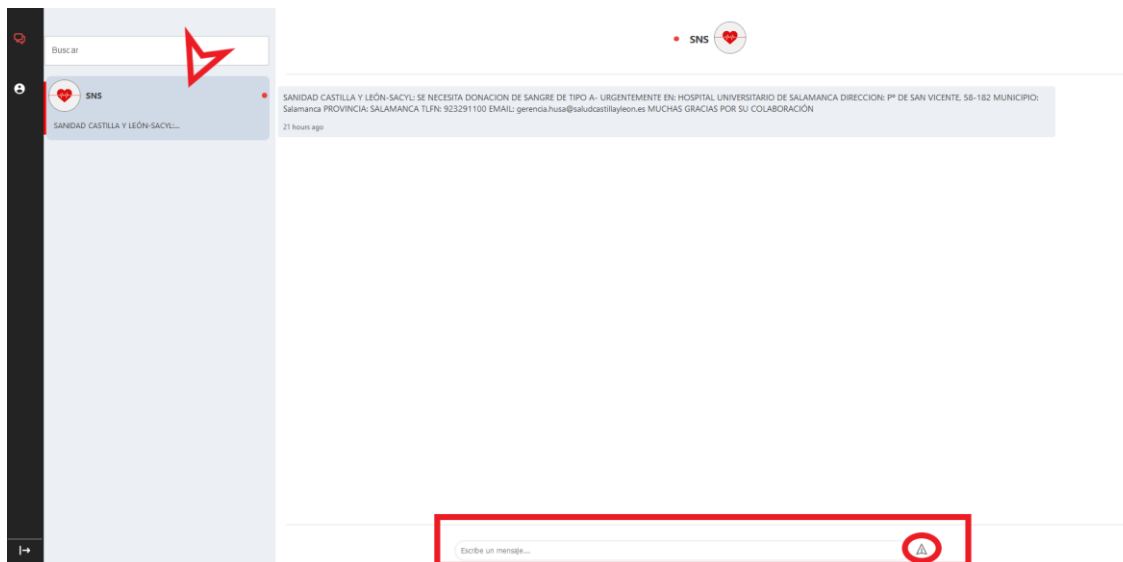


Figura 6: Chat

Para escribir un mensaje se debe introducir el mensaje a escribir en el cuadrado de texto y se podrá enviar pulsando "Enter" o bien, pulsando el botón señalado en la figura.

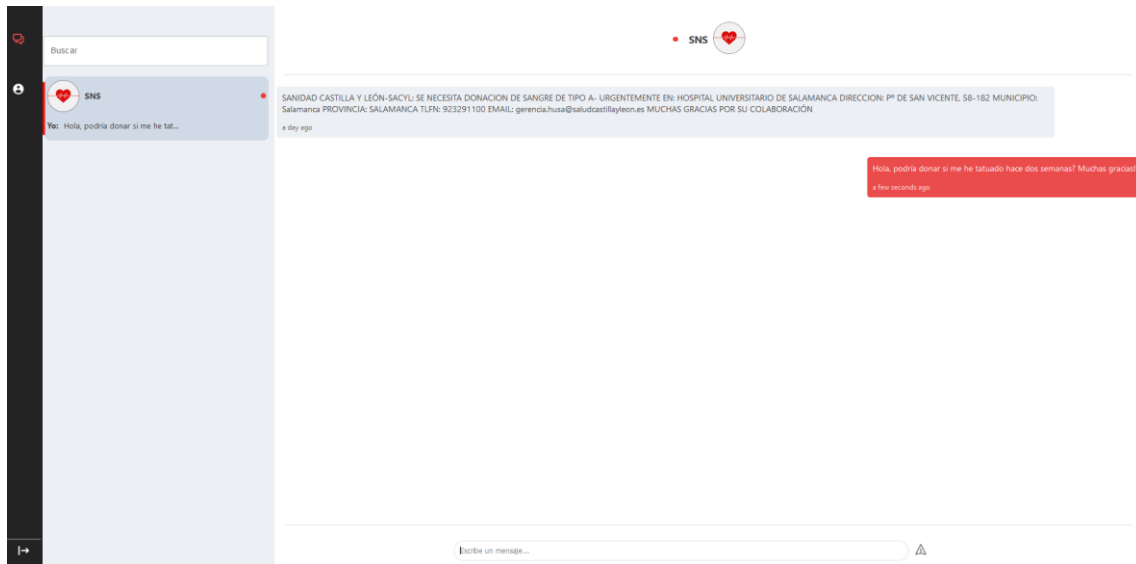


Figura 7: Chat con mensaje enviado

Los mensajes de color rojizo son los enviados con nuestra cuenta.

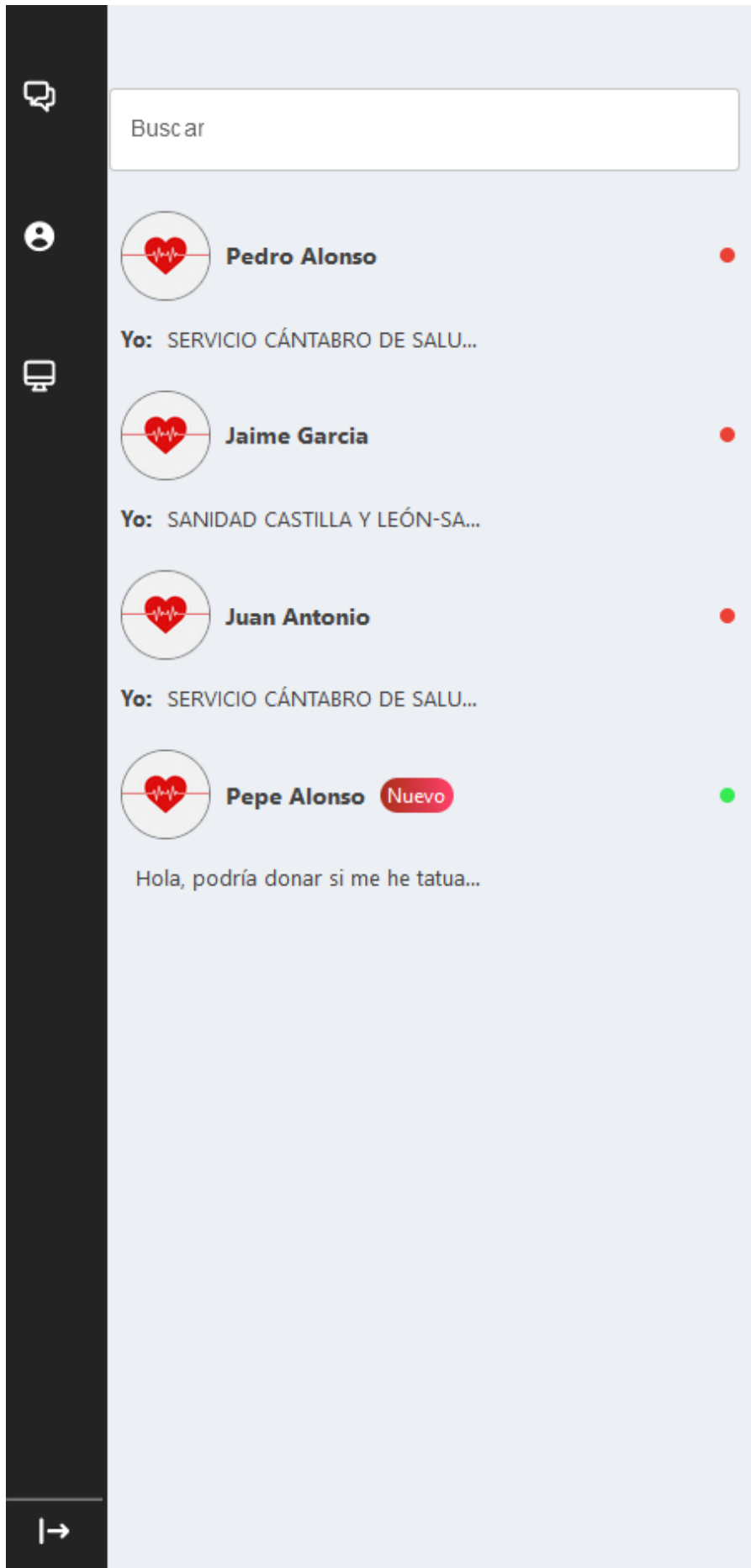


Figura 8: Mensaje sin leer

Una vez enviado el mensaje el receptor verá que tiene un mensaje nuevo sin leer.

7. Perfil.

Para acceder al perfil deberemos pulsar en el botón señalado en la Figura 9.

Hospitales cercanos

NOMBRE	DIRECCION	MUNICIPIO	TELEFONO
HOSPITAL GENERAL DE LA SANTISIMA TRINIDAD	Pº DE CARMELITAS, 74-94	Salamanca	923209300
COMPLEJO ASISTENCIAL UNIVERSITARIO DE SALAMANCA	Pº DE SAN VICENTE, 58-182	Salamanca	923209300
HOSPITAL UNIVERSITARIO DE SALAMANCA	Pº DE SAN VICENTE, 58-182	Salamanca	923209300
HOSPITAL LOS MONTAVALOS	CALLE DE SAN VICENTE, 58-182	Salamanca	923209300
HOSPITAL PROVINCIAL DE ZAMORA	HERNAN CORTES, 40	Zamora	980548200
HOSPITAL VIRGEN DE LA CONCHA	AV. REQUERO, 35	Zamora	980548200
COMPLEJO ASISTENCIAL DE ZAMORA	AVDA. REQUERO, 35	Zamora	980548200
HOSPITAL RECOLETAS DE ZAMORA	RINAR, 7	Zamora	980548300
HOSPITAL MEDINA DEL CAMPO	CALLE DE PENARANDA, 24	Medina del Campo	983638000
HOSPITAL DE SALUD MENTAL CASTA SALUD AREVALO	PASEO SAN JUAN BOSCO, 2	Arevalo	920209400

Figura 9: Perfil

Como se observa, se dispone de un mapa que muestra los hospitales más cercanos y sus datos en la tabla inferior. Los hospitales están ordenados por cercanía, siendo el primero el más cercano a nuestra ubicación.

7.1. Modificar información.

Para modificar la información de nuestro perfil solo deberemos editarla en el recuadro señalado en la figura 10 y presionar guardar cambios.

Hospitales cercanos

NOMBRE	DIRECCION	MUNICIPIO	TELEFONO
HOSPITAL GENERAL DE LA SANTISIMA TRINIDAD	Pº DE CARMELITAS, 74-94	Salamanca	923209300
COMPLEJO ASISTENCIAL UNIVERSITARIO DE SALAMANCA	Pº DE SAN VICENTE, 58-182	Salamanca	923209300
HOSPITAL UNIVERSITARIO DE SALAMANCA	Pº DE SAN VICENTE, 58-182	Salamanca	923209300
HOSPITAL LOS MONTAVALOS	CALLE DE SAN VICENTE, 58-182	Salamanca	923209300
HOSPITAL PROVINCIAL DE ZAMORA	HERNAN CORTES, 40	Zamora	980548200
HOSPITAL VIRGEN DE LA CONCHA	AV. REQUERO, 35	Zamora	980548200
COMPLEJO ASISTENCIAL DE ZAMORA	AVDA. REQUERO, 35	Zamora	980548200
HOSPITAL RECOLETAS DE ZAMORA	RINAR, 7	Zamora	980548300
HOSPITAL MEDINA DEL CAMPO	CALLE DE PENARANDA, 24	Medina del Campo	983638000
HOSPITAL DE SALUD MENTAL CASTA SALUD AREVALO	PASEO SAN JUAN BOSCO, 2	Arevalo	920209400

Figura 10: Perfil Modificar información

7.2. Modificar contraseña.

Para modificar la contraseña deberemos pulsar sobre el botón seguridad y nos mostrará otro componente en la parte inferior, tal y como se muestra en la figura 11.

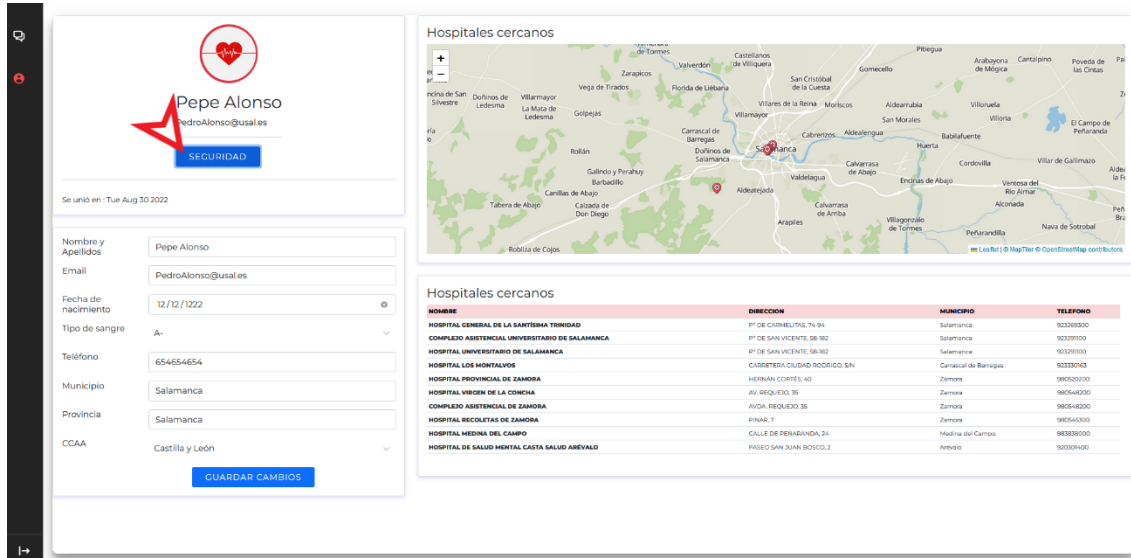


Figura 11: Cambiar contraseña parte 1

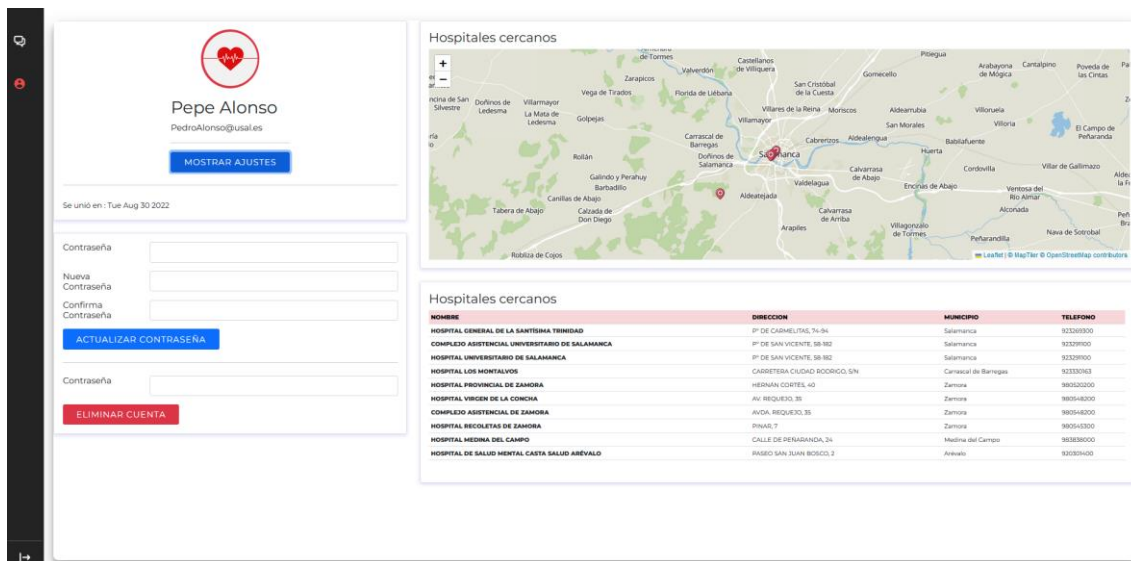


Figura 12: Cambiar contraseña parte 2

Una vez en este menú, solo deberemos introducir nuestra antigua contraseña, la nueva contraseña e introducir de nuevo la nueva contraseña para confirmar, y posteriormente pulsar sobre actualizar contraseña.

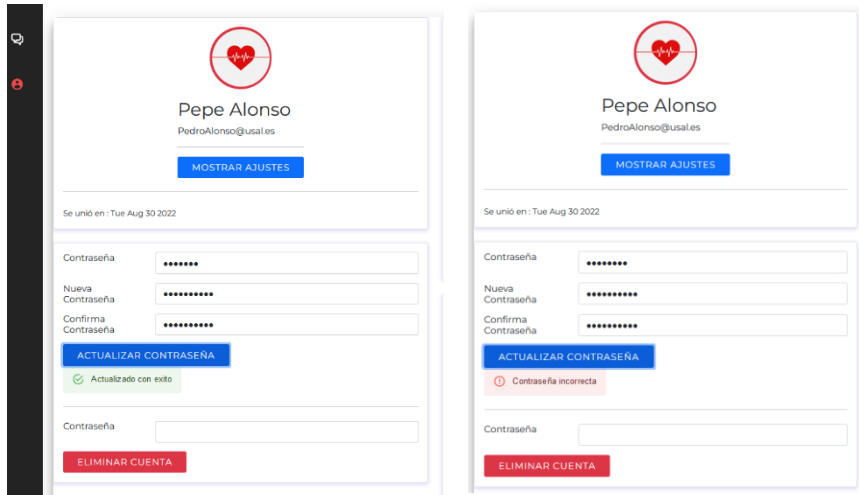


Figura 13: Contraseña cambiada. Figura 14: Error cambiar contraseña

Como podemos observar el sistema nos avisará de si la contraseña se ha actualizado correctamente o si ha habido algún error.

7.3. Eliminar cuenta.

En el mismo menú que en el apartado “6.2 Modificar contraseña” podremos eliminar nuestra cuenta. Para ello solo necesitaremos introducir nuestra contraseña y pulsar sobre “Eliminar cuenta” una vez pulsado, toda la información de su cuenta será completamente borrada del servidor y será redirigido a la página de inicio de sesión.

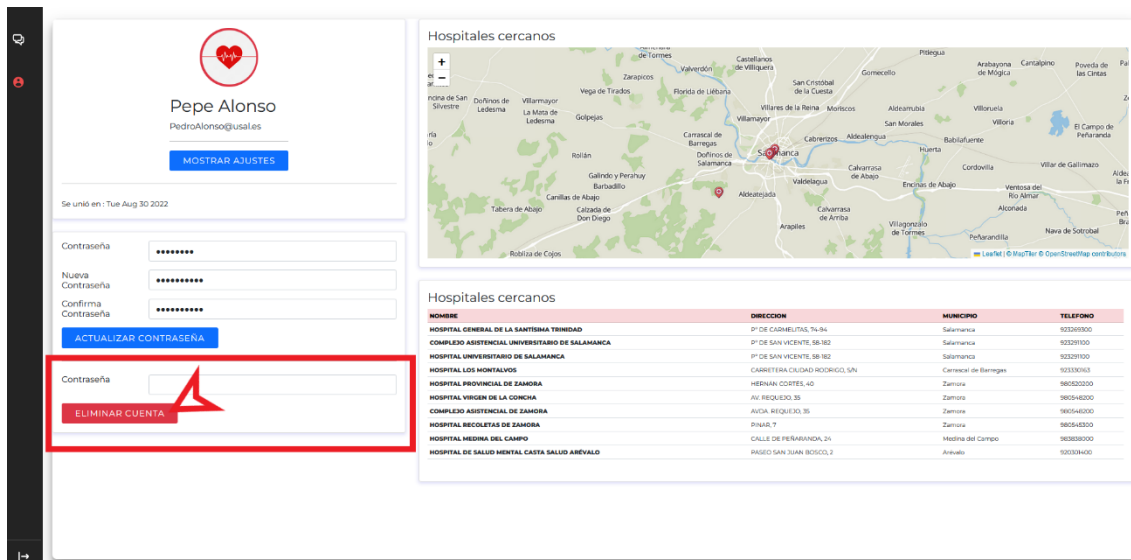


Figura 15: Eliminar cuenta

7.4. Ver hospitales en el mapa.

Se puede interactuar con el mapa para que nos de el nombre del hospital. Para ello solo debemos pulsar sobre cualquier marker situado en el mapa, tal y como se muestra en la Figura 15.

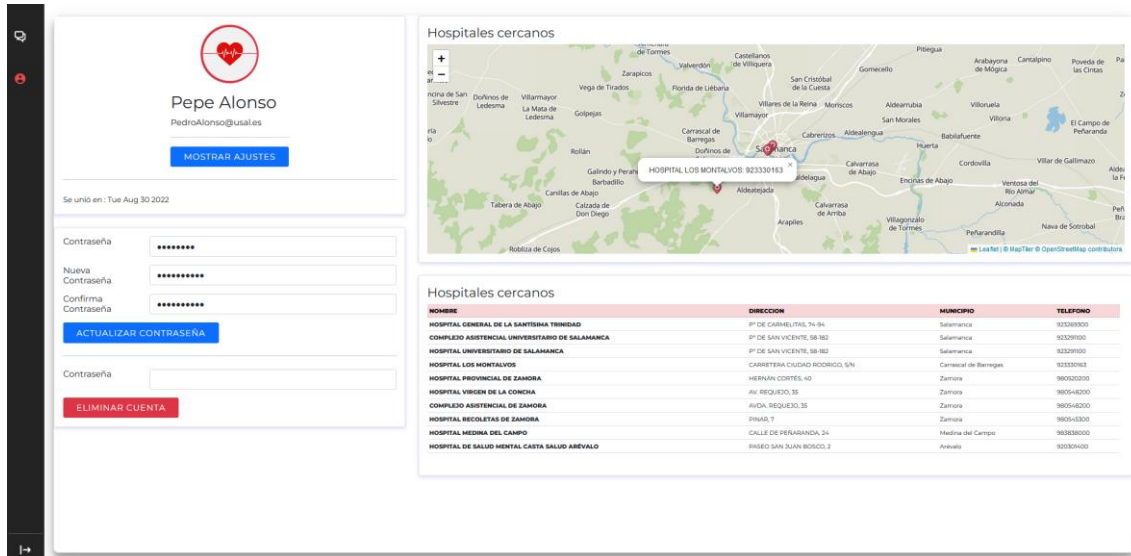


Figura 16: Ver hospitales en el mapa

8. Administrador.

Este apartado se describirá las tareas que puede realizar el administrador. El administrador puede realizar todas las tareas previamente descritas para los usuarios y además puede realizar una serie de funciones especiales.

8.1. Vista administrador.

El usuario administrador podrá chatear con cualquier usuario, además dispondrá de una barra de búsqueda para buscar al donante por nombre.

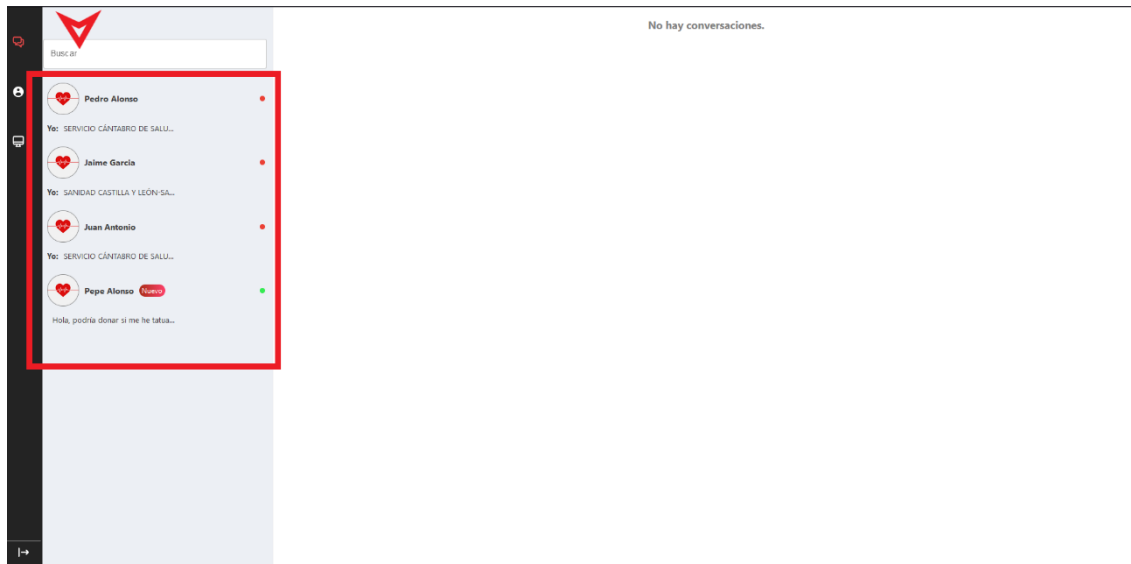


Figura 17: Chat Administrador

8.2. Buscar donante por nombre en el chat.

Para filtrar por nombre solo deberemos pulsar en "Buscar" y escribir que queremos buscar. Se puede buscar tanto por nombre como por apellido.

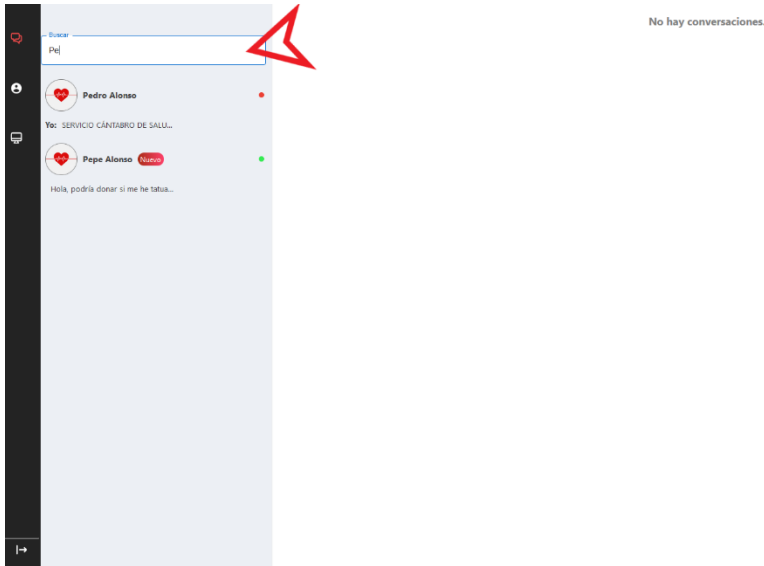


Figura 18: Búsqueda en el chat por nombre

8.3. Acceder a la Página de control.

El usuario administrador dispone de una página de control. Para acceder a ella deberemos pulsar en el botón señalado en la figura 18.

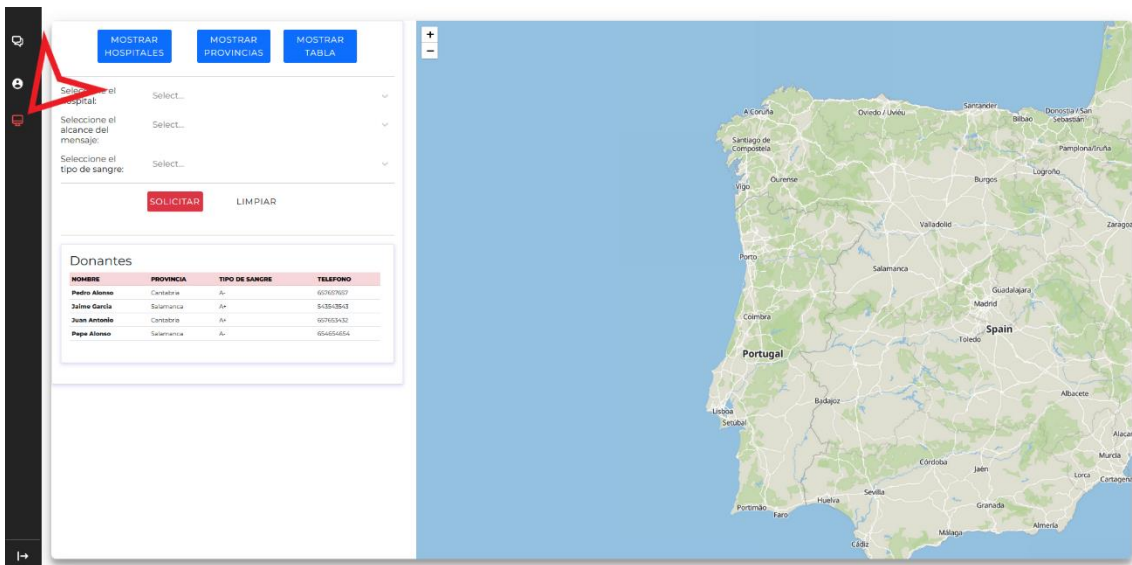


Figura 19: Página de control

Una vez dentro de la página de control observamos distintos botones en la parte superior.

8.4 Ver todos los hospitales.

El botón **“Mostrar Hospitales”** mostrará todos los hospitales registrados en España y su ubicación en el mapa tal y como se muestra en la figura 19. A su vez se podrá seleccionar cualquier “marker” para ver su nombre.

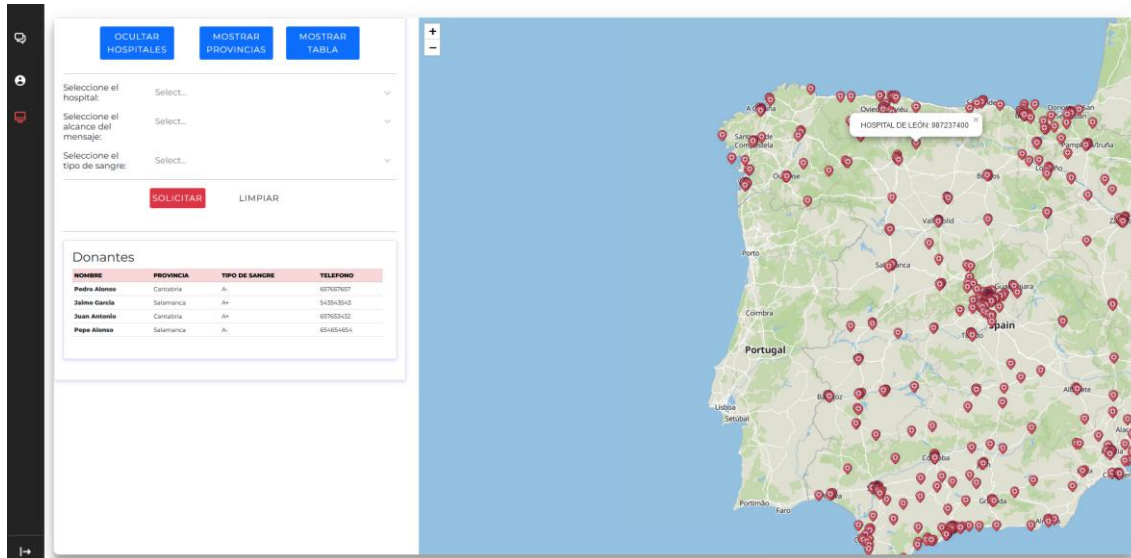


Figura 20: Página de control – botón mostrar hospitales

8.5 Ver donantes por provincias.

El botón **“Mostrar provincias”** dibujará las provincias en el mapa. Para ver los donantes que hay por cada provincia solo deberemos pulsar la provincia en el mapa, tal y como se enseña en la Figura 20.

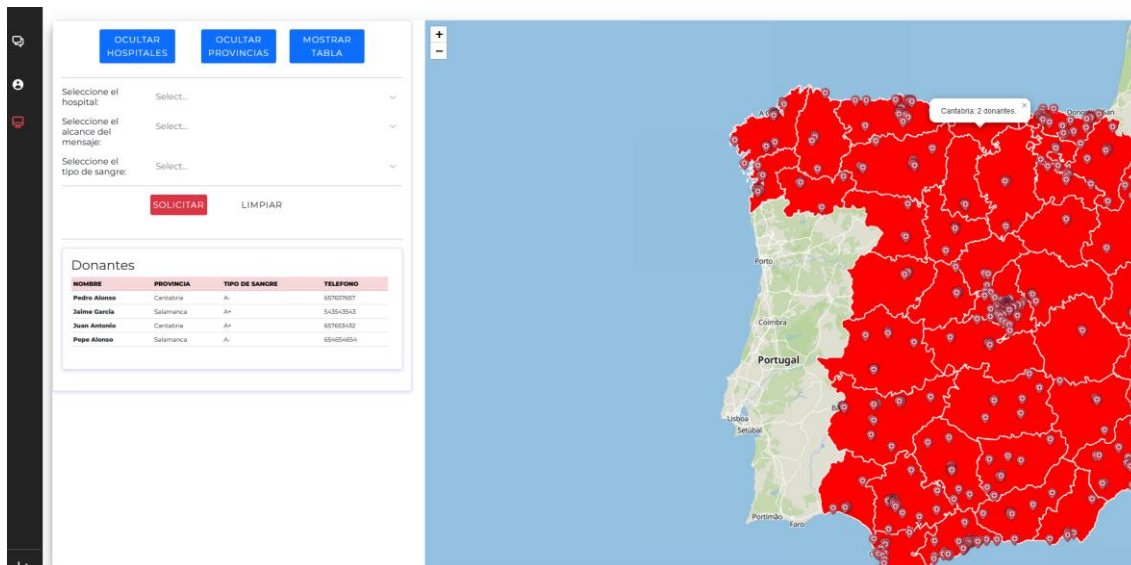


Figura 21: Ver donantes por provincia

8.6 Ver información de los hospitales registrados.

También se puede disponer de la información de los hospitales registrados en forma de tabla, para ello solo deberemos pulsar sobre el botón **“Mostrar tabla”**

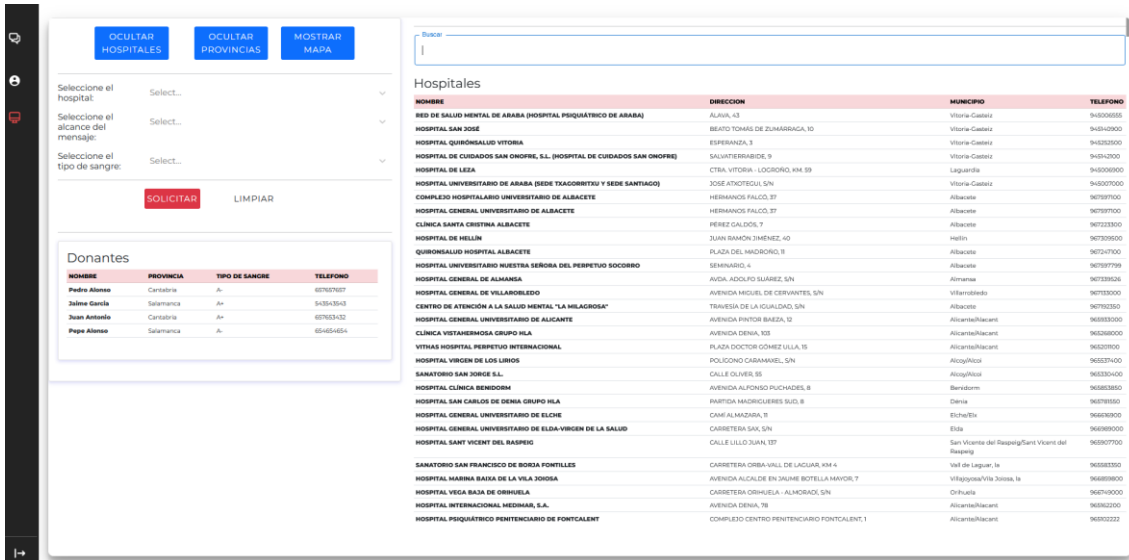


Figura 22: Ver información todos los hospitales tabla

Si se desea se puede filtrar por el nombre del hospital tal y como se detalla en la Figura 22.

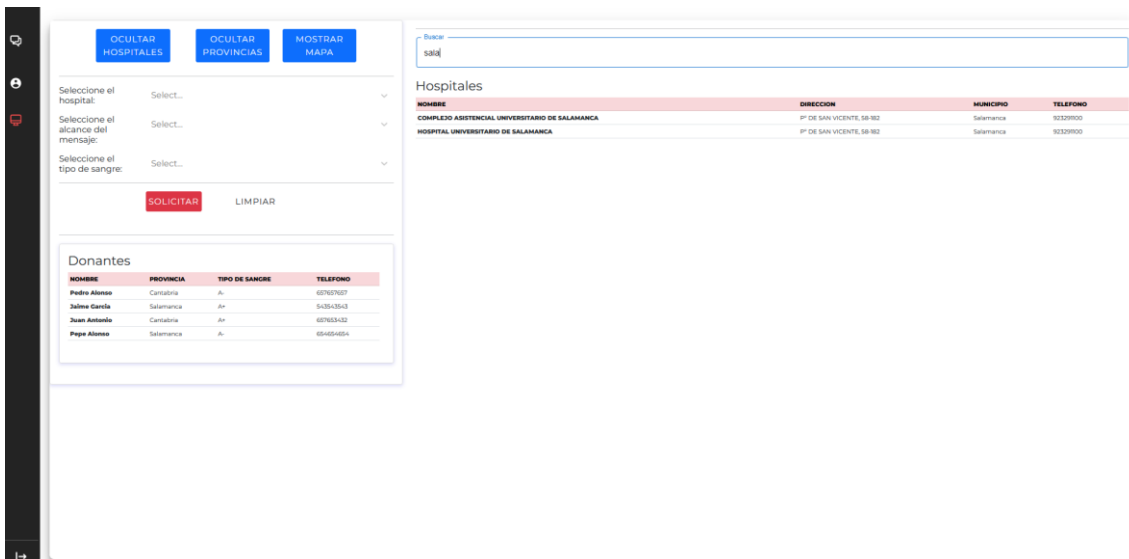


Figura 23: Filtrar hospitales

8.7 Ver donantes registrados.

Simplemente entrando a la página de control ya se pueden observar los donantes registrados, tal y como se muestra en la Figura 23.

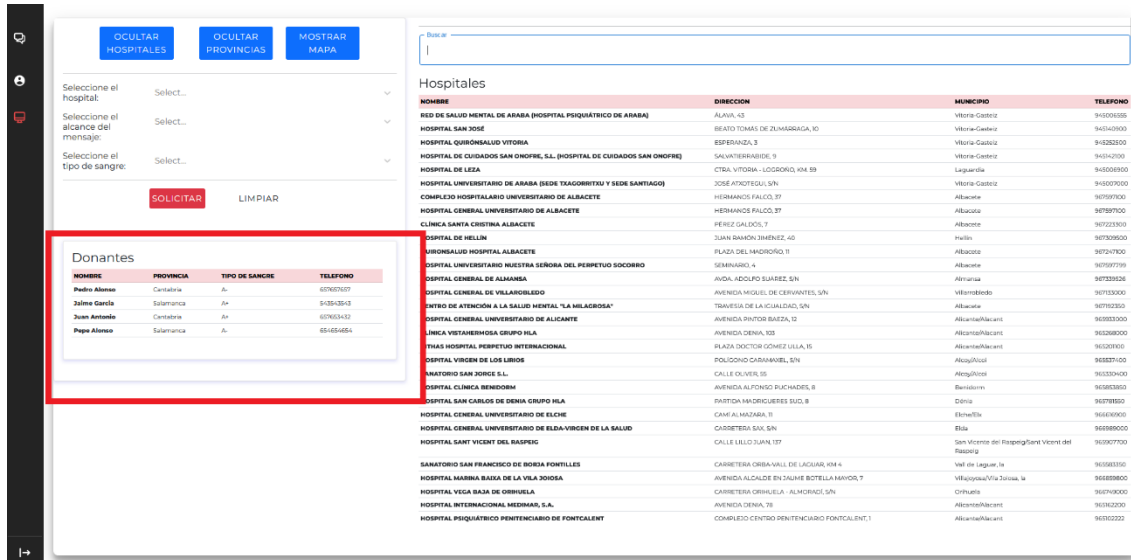


Figura 24: Ver donantes registrados

8.8 Enviar alerta.

Para enviar la alerta debemos rellenar los campos mostrados en la figura 24.

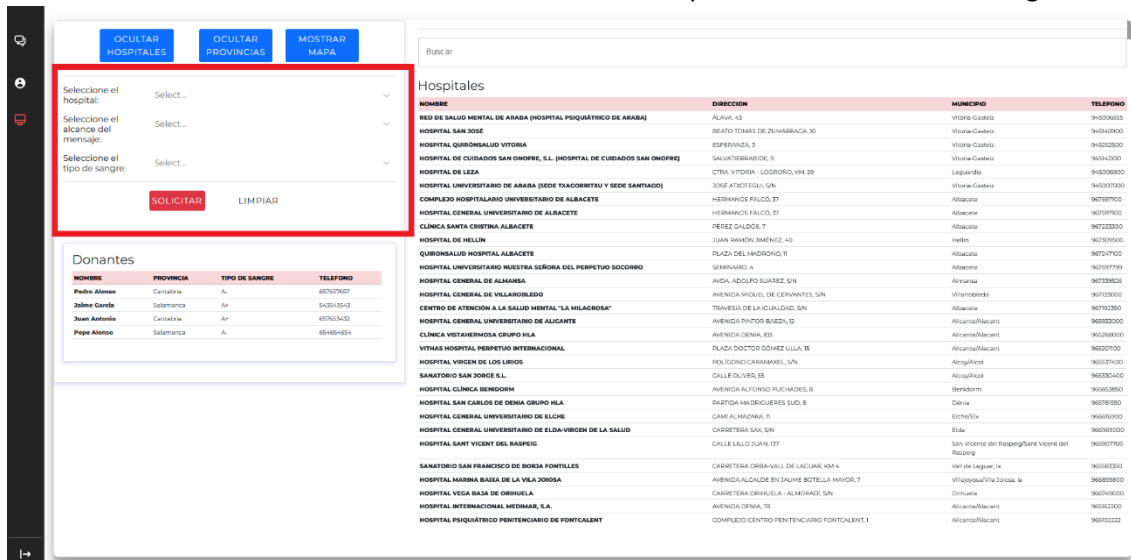


Figura 25: Campos a rellenar para enviar alerta.

Los campos que rellenar son:

- **Hospital:** Se necesita especificar en que hospital se necesita sangre.
- **Alcance:** Se refiere a la difusión del mensaje, puede ser Municipio, Provincia o CCAA.
- **Tipo de sangre:** Tipo de sangre que se solicita.

Una vez rellenados se pulsará en solicitar y se difundirá un mensaje a todos los donantes que cumplan las condiciones. También se podrá visualizar los usuarios a los que se les ha enviado la alerta, tal y como se detalla en la figura 25.

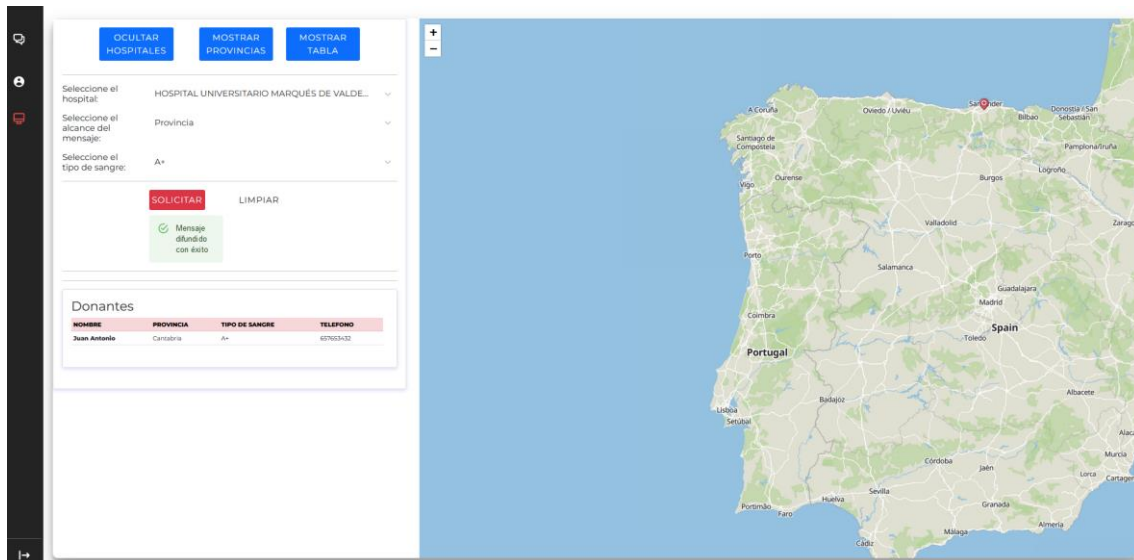


Figura 26: Alerta enviada con éxito

Si la alerta se difunde bien el sistema avisará al administrador, y se mostrará en la tabla señalada en la Figura 26, los usuarios a los que se les ha difundido la alerta. Por otra parte, mostrará en el mapa el hospital dónde se ha solicitado la donación de sangre.

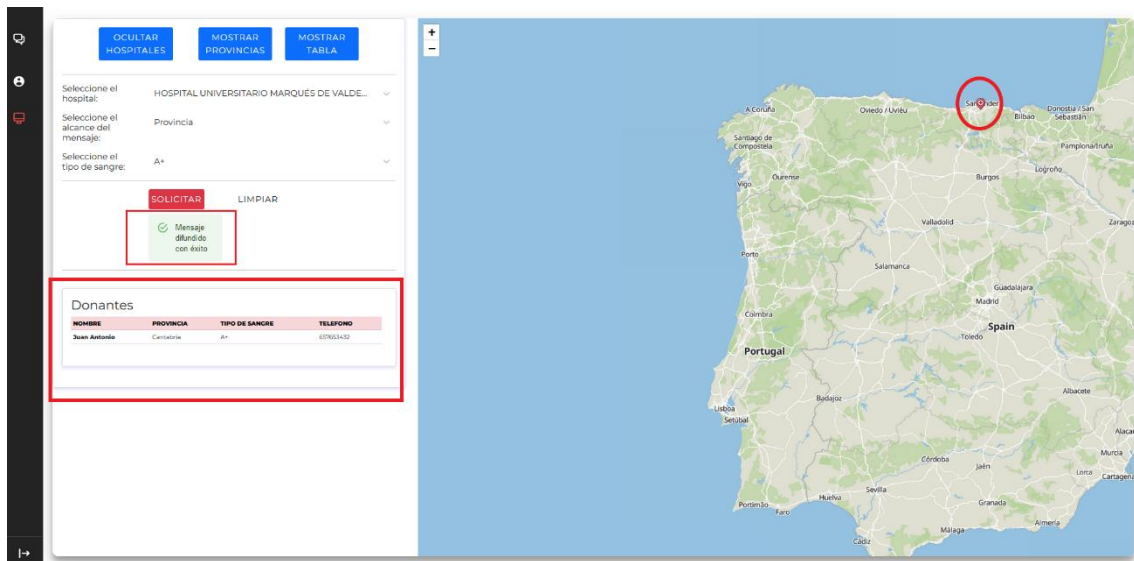


Figura 27: Ver donantes a los que se les ha enviado la alerta

En el mensaje de alerta se indicará el servicio de salud responsable, el tipo de sangre que necesitan y la ubicación del hospital. Adicionalmente, se envía también el correo electrónico y el número de teléfono por si el usuario precisa contactar con el servicio de salud.

SERVICIO CÁNTABRO DE SALUD-SCS: SE NECESITA DONACION DE SANGRE DE TIPO A- URGENTEMENTE EN: HOSPITAL UNIVERSITARIO MARQUÉS DE VALDECILLA DIRECCION: AV. DE VALDECILLA, S/N MUNICIPIO: Santander PROVINCIA: CANTABRIA TLFN: 942202520 EMAIL: dirger.humv@scsalud.es MUCHAS GRACIAS POR SU COLABORACIÓN

Figura 28: Mensaje de alerta enviado

8.9 Limpiar datos.

Para volver a la vista de la página de control original solo deberemos pulsar el botón de limpiar y eliminará todos los filtros generados por la alerta.

9. Bibliografía.