

# Memoria del proyecto

GESTIGAN: Aplicación web para la gestión de tareas en el sector agropecuario.

Trabajo de Fin de Grado  
INGENIERÍA INFORMÁTICA



**VNiVERSIDAD  
D SALAMANCA**

CAMPUS DE EXCELENCIA INTERNACIONAL

**Septiembre 2022**

**Autor:**

Sergio Gómez Sánchez

**Tutores:**

André Filipe Sales Mendes

Gabriel Villarubia González

Juan Franciso de Paz Santana

Memoria del proyecto

## CERTIFICADO DE LOS TUTORES

D.André Sales Mendes, D. Gabriel Villarubia González, D. Juan Francisco De Paz Santana,  
profesores/as del Departamento de Informática y Automática de la Universidad de Salamanca.

CERTIFICAN:

Que el trabajo titulado "*GESTIGAN: Aplicación web para la gestión de tareas en el sector agropecuario*" ha sido realizado por Sergio Gómez Sánchez, con DNI 70911097G, bajo su dirección.

Y para que conste a todos los efectos oportunos.

En Salamanca, a 7 de septiembre de 2022

D.André Sales Mendes

D. Gabriel Villarubia González

D. Juan Francisco De Paz Santana

## RESUMEN

Gestigan es un proyecto nacido por la necesidad de crear un sistema de gestión de explotaciones ganaderas usable e intuitivo que mejorase los sistemas ya existentes, que asegurase la seguridad de la información y que pudiese usarse desde cualquier dispositivo.

Esta necesidad surgió por parte de una gestora de una ganadería salmantina dedicada a la crianza y explotación con fines de producción de animales bovinos.

Este proyecto consiste en el diseño e implementación de una aplicación web para la gestión de tareas en el sector agropecuario, que se enfocará en la gestión de fincas, animales, alimentos y medicamentos. Se encargará de llevar a cabo procesos para facilitar a sus usuarios la correcta gestión de sus explotaciones ganaderas, ayudando en el mantenimiento de la administración de los datos, controlando las transacciones que se realicen, generando históricos de gestión y estadísticas que ayuden a su análisis, mejorando así el rendimiento en el trabajo de este sector.

Para lograr un proyecto escalable, de fácil mantenimiento, estable y eficiente se han puesto en práctica muchas de las técnicas aprendidas a lo largo de la carrera en distintas asignaturas, pero en especial las relacionadas con las asignaturas de Ingeniería del Software y de Gestión de Proyectos.

El proyecto ha pasado por distintas fases de acuerdo con el marco del Proceso Unificado, desde la planificación hasta la implementación, pasando por la especificación de requisitos, el análisis de estos y el diseño final del software, quedando documentado y reflejado en un total de seis anexos que acompañarán la memoria del trabajo.

Una memoria en la que se tratará de explicar el proceso de creación del proyecto explicando las distintas técnicas, herramientas usadas desde la idea inicial hasta el sistema final implementado.

Este proyecto será presentado como Trabajo de Fin de Grado por el desarrollador de la aplicación.

**Palabras clave:** Sector agropecuario, sistema de gestión, explotación, transacción.

## SUMMARY

Gestigan is a project born out of the need to create a easy to use and intuitive management system of livestock farms that improves existing systems, ensures security of information and can be used from any device.

This need came from a manager of a Salamanca livestock dedicated to raising and exploitation bovine animals for purposes related to production.

This work consists of the design and implementation of a web application for task management in the livestock sector, which will focus on the management of farms, animals, food and medicines. It will take over carrying out processes to ease for users the correct management of their livestock farms, helping in the maintenance of data administration and controlling the transactions that take place, generating management histories and statistics to assist in their analysis, improving work efficiency in this sector.

To achieve a scalable that is easy to maintain, stable and efficient project, many of the techniques learned throughout the degree in different subjects have been put into practice, but especially those related to Software Engineering and Project Management subjects.

The project has gone through different phases in accordance with the framework of the Unified Process, from planning to implementation, passing through the specification of requirements, their analysis and the final design of the software, being documented and reflected in a total of six annexes that will accompany the memory of the work.

A memory that will try to explain the project creation process explaining the different techniques, tools used from the initial idea to the final implemented system.

This project will be presented as Final Degree Project by the developer of the application.

**Keywords:** livestock sector, management system, exploitation, transactions

## TABLA DE CONTENIDO

1.- Introducción: .....	1
2.- Necesidad .....	3
2.1.- Sistema de gestión local .....	3
2.1.- Sistema de gestión profesional .....	5
3.- Estado del arte .....	10
3.1.- VacApp: .....	10
3.2.- Bovi Control: .....	11
4.- Objetivos .....	12
4.1.- Objetivos del sistema .....	12
4.1.- Objetivos personales .....	13
5.- Conceptos teóricos .....	14
5.1.- Framework Angular .....	14
5.2.- Firebase .....	14
5.3.- Diseño Responsive.....	14
5.4.- Librería o bibliotecas .....	15
6.- Técnicas y herramientas .....	15
6.1.- Framework: .....	15
6.1.1.- Angular: .....	15
6.2.- Lenguajes.....	15
6.2.1.- HTML.....	15
6.2.1.- CSS .....	15
6.2.1.- TypeScript .....	16
6.2.1.- JSON.....	16
6.2.1.- UML .....	16
6.3.- Librerías:.....	16
6.3.1.- Firebase: .....	16
6.3.1.- Router: .....	16
6.3.1.- Bootstrap: .....	17
6.3.1.- Chart.js:.....	17
6.3.1.- ng2-charts.js: .....	17
6.4.-Herramientas de compilación y hosting: .....	17
6.4.1.- Node.Js: .....	17
6.4.1.- Npm: .....	17
6.5.- Herramientas de control de versiones:.....	17

## Memoria del proyecto

6.5.1.- GitHub.....	17
6.5.1.- SourceTree.....	18
6.6.- Herramientas CASE.....	18
6.6.1.- Visual Paradigm .....	18
6.6.1.- EZEstimate .....	18
6.6.1.- Microsoft Project.....	18
6.7.-Herramienta de documentación.....	18
6.7.1.- Compodoc:.....	18
6.8.- Entorno de desarrollo: .....	18
6.8.1.- Visual Studio Code:.....	18
6.9.- Herramientas de diseño gráfico:.....	19
6.9.1.- Adobe XD: .....	19
6.9.1.- Adobe Photoshop CS6:.....	19
6.9.1.- Paletton: .....	19
7.- Aspectos relevantes del desarrollo.....	19
7.1.- Marco de trabajo.....	19
7.2.- Estimación del coste y esfuerzo .....	21
7.2.1.- UUCP - Punto de Casos de Uso Sin Ajustar.....	21
7.2.1.1.- UAW - Interacción de los Actores .....	21
7.2.1.2.- UUCW - Complejidad de los Casos de Uso .....	21
7.2.2.- TCF - Factores de Complejidad Técnica.....	22
7.2.3.- ECF - Factores de Complejidad del Entorno .....	22
7.3.- Planificación temporal.....	24
7.4.- Especificación de requisitos .....	26
7.4.1.- Participantes.....	26
7.4.2.- Catálogo de requisitos:.....	26
7.4.2.1.- Objetivos del sistema.....	26
7.4.2.1.- Actores .....	27
7.4.2.1.- Requisitos de información .....	28
7.4.2.1.- Requisitos de restricción de información .....	29
7.4.2.1.- Requisitos no funcionales.....	29
7.4.2.1.- Requisitos funcionales .....	29
7.4.2.1.- Matriz de rastreabilidad.....	31
7.5.- Análisis de requisitos.....	32
7.5.1.- Modelo de dominio .....	32
7.5.2.- Vista de Interacción de los Casos de Uso .....	33

## Memoria del proyecto

7.5.3.- Clases de Análisis .....	34
7.5.4.- Vista de la Arquitectura del Modelo de Análisis.....	34
7.6.- Diseño del sistema software .....	35
7.6.1.- Patrón Arquitectónico .....	35
7.6.2.- Subsistemas de diseño .....	36
7.6.2.- Vista Arquitectónica del Modelo de Diseño .....	36
7.6.2.- Diseño de la Base de Datos - Firebase.....	37
7.6.3.- Modelo de Despliegue.....	37
7.7.- Implementación .....	38
7.7.1.- Interfaz de usuario: diseño “responsive” y estructura general.....	38
7.7.2.- Gestión de Fincas.....	43
7.7.3.- Gestión de Animales .....	44
7.7.4.- Gestión de Usuarios y Organizaciones .....	45
7.7.5.- Configuración de la base de datos.....	46
7.7.6.- Históricos de transacciones .....	49
7.7.7.- Estadísticas .....	49
7.8.- Pruebas.....	49
7.9.- Funcionalidad del sistema .....	50
7.9.1.- Inicio .....	50
7.9.2.- Registro e Inicio de Sesión .....	51
7.9.3.- Inventario .....	52
7.9.4.- Finca.....	53
7.9.5.- Tablas de Animales, Alimentos y Medicamentos.....	53
7.9.6.- Estadísticas .....	54
7.9.7.- Gestión.....	56
7.9.8.- Animal.....	57
7.9.9.- Perfil.....	58
7.9.10.- Administración.....	59
9.- Conclusiones y líneas de trabajo futuras .....	60
9.1.- Conclusiones.....	60
9.2.- Líneas de trabajo futuras.....	60
10.- Referencias.....	62
11.- Bibliografía .....	65

## Índice de ilustraciones

Ilustración 1.- Excel de Gestión - Hoja de altas y bajas .....	3
Ilustración 2.- Excel de Gestión - Hoja de alimentos.....	4
Ilustración 3.- Excel de Gestión – Hoja de medicamentos.....	4
Ilustración 4.- Sistema Gestión Junta CyL .....	6
Ilustración 5.- Sistema Gestión Junta CyL. Iniciar Sesión .....	6
Ilustración 6.- Sistema Gestión Junta CyL. Fincas.....	6
Ilustración 7.-Sistema Gestión Junta CyL. Consulta General.....	7
Ilustración 8.- Sistema Gestión Junta CyL. Consulta de movimientos .....	7
Ilustración 9.- Sistema Gestión Junta CyL. Consulta de animales por Finca .....	8
Ilustración 10.- Sistema de Gestión CyL. Búsqueda de Animales .....	8
Ilustración 11.- Sistema de Gestión CyL. Consulta de Animal individual .....	8
Ilustración 12.- Aplicación VacApp.....	10
Ilustración 13.- Captura 1 BoviControl.....	11
Ilustración 14.- Captura 2 BoviControl.....	11
Ilustración 15.- Fases Proceso Unificado.....	20
Ilustración 16.- Cálculo EZEstimate .....	23
Ilustración 17.- Planificación Imagen 2 .....	24
<i>Ilustración 18.- Planificación Imagen 1 .....</i>	<i>25</i>
Ilustración 19.- Diagrama de Actores del Sistema .....	27
Ilustración 20.- Diagrama de Paquetes del Sistema.....	30
Ilustración 21.- Diagrama de Casos de Uso de Gestión de Usuarios.....	30
Ilustración 22.- Modelo de Dominio .....	32
Ilustración 23.- Diagrama de Secuencia Caso de Uso Añadir Finca .....	33
Ilustración 24.- Clases de Análisis. Gestión de Usuarios .....	34
Ilustración 25.- Modelo-Vista-Controlador MVC .....	35
Ilustración 26.- Subsistemas de Diseño.....	36
Ilustración 27.- Diagrama Secuencia: UC-0028 Modificar Alimento.....	36
Ilustración 28.- Modelo de Despliegue .....	37
Ilustración 29.- Captura de Página Inicio .....	50
Ilustración 30.- Diseño Mobile Inicio.....	50
Ilustración 31.- Captura de Página Registro.....	51
Ilustración 32.- Captura Diseño Mobile Iniciar Sesión .....	52
Ilustración 33.- Captura de Página Inventario.....	52
Ilustración 34.- Diseño Mobile Inventario.....	53
Ilustración 35.- Captura Modificar Tabla de Animales.....	53
Ilustración 36.- Captura de Tabla de Alimentos.....	54
Ilustración 37.- Captura Error en formulario Añadir Medicamento .....	54
Ilustración 38.- Captura de Página Estadísticas .....	55
Ilustración 39.- Captura de Página Gestión.....	56
Ilustración 40.- Captura de Página Animal.....	57
Ilustración 41.- Diseño Mobile Animal .....	57
Ilustración 42.- Diseño Mobile Perfil.....	58
Ilustración 43.- Captura de página Administración.....	59
Ilustración 44.- Captura de elección de Usuarios en Administración .....	59

## Índice de tablas

Tabla 1.- Tabla de complejidad de actores .....	21
Tabla 2.- Tabla complejidad Casos de Uso .....	21
Tabla 3.- TCF .....	22
Tabla 4.- ECF .....	23
Tabla 5.- Gestión de Usuarios .....	27
Tabla 6.- Actor Trabajador de una Organización .....	27
Tabla 7.- Animales Registrados .....	28
Tabla 8.- UC-0020 Buscar Animal .....	31

## 1.- Introducción:

En el mundo actual en el que vivimos existen todo tipo de aplicaciones o servicios web que mejoran y hacen más sencillo cualquier aspecto de nuestra vida, sin embargo, cuando se trata del sector agropecuario parece que esté olvidado.

Además de que no existen demasiadas ayudas a nivel de tecnologías, las que existen no parecen tener un gran éxito dentro del sector o están demasiado anticuadas, necesitando claramente un lavado de cara y una renovación.

Este proyecto tiene como objetivo facilitar la mejora de rendimiento en el trabajo dentro del sector agropecuario. En concreto se centrará en el apartado de gestión de fincas, animales, alimentos y medicamentos de una organización ganadera, cubriendo gran parte de las necesidades que pueda tener para su gestión.

En este documento se explicarán algunos de los aspectos más importantes del desarrollo del proyecto. La estructura de este documento será la siguiente:

- **Necesidad:** Se planteará la necesidad generada por parte de un cliente externo.
- **Estado del arte:** Se mostrarán algunas de las aplicaciones o servicios web que existen actualmente en el mercado que tengan objetivos similares.
- **Objetivos:** Se especificarán los objetivos que pretende alcanzar el proyecto y los objetivos que se desean conseguir a nivel personal.
- **Concetos teóricos:** Se realizará una breve explicación de algunos de los conceptos más importantes a nivel teórico para poder comprender de forma correcta el proyecto.
- **Técnicas y herramientas:** Documentación de las técnicas y herramientas usadas para llevar a cabo este proyecto.
- **Aspectos relevantes del desarrollo:** Recogerá los aspectos más importantes en el desarrollo y despliegue del sistema.
- **Limitaciones del sistema:** Se hablará de las limitaciones a nivel técnico encontradas al final del proyecto.
- **Conclusiones y líneas de trabajo futuras:** Se realizará una valoración del proyecto llegado a su fin comentando las conclusiones que se han obtenido y las posibilidades de trabajo futuro que presenta el sistema.
- **Bibliografía.**

Este documento irá complementado de los siguientes anexos:

- **Anexo I – Planificación temporal:** Lleva a cabo una estimación de la duración del proyecto con su correspondiente planificación temporal de tareas a realizar en el proyecto.
- **Anexo II – Especificación de requisitos software:** Recoge la especificación de requisitos del sistema.

## Memoria del proyecto

- **Anexo III – Análisis de requisitos:** Documentación sobre la fase de análisis de los requisitos recogidos en el anexo anterior.
- **Anexo IV – Diseño del sistema software:** Documentación que recoge la información sobre la fase de diseño del sistema.
- **Anexo V – Documentación técnica:** Documento que contendrá la información necesaria del sistema para facilitar la comprensión del código desarrollado.
- **Anexo VI – Manual de usuario:** Documento que contará de forma detallada la funcionalidad del sistema para que el usuario pueda utilizarlo sin problema.

## 2.- Necesidad

La idea de este proyecto no surgió por parte del creador, si no por la necesidad que tenía una las gestoras de una organización salmantina que se dedica a la producción agrícola, pero especialmente a la producción de ganado.

Esta gestora, Blanca Sánchez Boyero, perteneciente a la organización SánchezBoyeroSL planteó mejorar su sistema de gestión. Por tanto, el proceso daba comienzo con las conversaciones previas al planteamiento de objetivos. Se intentó establecer las necesidades que tenía la gestora para intentar cubrirlas con un nuevo sistema de gestión.

Antes de comentar los problemas, se definirán los sistemas de gestión que manejaba la gestora. En concreto eran 2, uno a nivel local y personal y otro a nivel profesional.

### 2.1.- Sistema de gestión local

Se trata de un Excel de varias páginas donde se guardaba la información de altas y bajas de animales por explotación, alimentos y medicamentos.

El sistema registraba todos estos datos en formato de tablas que quedaban alojados en el ordenador personal de Blanca.

A continuación, se mostrarán algunas capturas de pantalla:

MODELO IV-A						
HOJA DE ALTAS Y BAJAS DE ANIMALES						
Código de Explotación ES370081100891				Especie: BOVINO		Nº de hoja: _____
Balance de animales de Hoja de altas y bajas de animales anterior:						
FECHA	MOTIVO DEL ALTA/BAJA	2	NUMERO DEL DOCUMENTO	Nº DE ANIMALES QUE CAUSAN ALTA/BAJA	BALANCE FINAL	
6,12	31-jul.-20	SACRIFICIO	ES052472000011222	082223703202000003275	2	45
6,13	19-ago.-20	SACRIFICIO	ES09059200010122	0837203202000003606	1	44
6,14	02-oct.-20	SACRIFICIO	ES052472000011222	083272099402000061269	1	43
6,15	06-nov.-20	SACRIFICIO	ES052472000011222	083722209940000062351	1	42
6,16	13-nov.-20	SACRIFICIO	ES05019200004122	083222709940000062579	1	41
7,01	13-nov.-20	SACRIFICIO	ES052472000011222	08223709940000062576	1	40
7,02	27-nov.-20	SACRIFICIO	ES05019200004122	02283709940000062984	1	39
7,03	15-ene.-21	SACRIFICIO	ES05019200004122	08223709940000064398	2	37
7,04	26-feb.-21	SACRIFICIO	ES05019200004122	08223709940000065667	2	35
7,05	06-abr.-21	NACIMIENTO	ES37008110089122	20219000659539	5	40
7,06	13-abr.-21	NACIMIENTO	ES37008110089122	20219000714752	2	42
7,07	10-may.-21	VENTA	ES44076000042122	082237203202100002098	1	41
7,08	24-may.-21	VENTA	ES40208110062122	0822237032021000022389	1	40
7,09	02-jun.-21	VENTA	ES37008000017922	082322703202100002584	4	36
7,10	14-jun.-21	VENTA	ES31064000012422	0822370320210000279228	1	35
7,11	16-jun.-21	SACRIFICIO	ES10019000009022	220283703202100002833	7	28
7,12	30-jun.-21	MUERTE	ES37008110089122	2220219001300247	1	27
7,13						
7,14						
7,15						

Ilustración 1.- Excel de Gestión - Hoja de altas y bajas

Memoria del proyecto

MODELO VI				
HOJA DE ALIMENTOS SUMINISTRADOS (EXCEPTO PIENSOS MEDICAMENTOSOS)				
Código de Explotación ES370081100891				
Especie: BOVINO				
PAGINA ASIENTO	FECHA DE COMPRA	NATURALEZA DEL ALIMENTO	CANTIDAD KG	DEL DOCUMENTO COMERCIAL
2,11	26-may.-17	CAMPAL/MAIZ	25,81	
2,12	15-dic.-17	CAMPAL/MAIZ	27,68	
2,13	15-dic.-17	CAMPAL/CEBADA	26,68	
2,14	27-abr.-18	MAIZ/CAMPAL	26,72	
2,15	27-abr.-18	CEBADA/CAMPAL	26,7	
2,16	7-sep.-18	MAIZ/CAMPAL	27,04	
2,17	7-sep.-18	CEBADA/CAMPAL	31,3	
2,18	22-feb.-19	MAIZ/CAMPAL	27,7	
2,19	22-feb.-19	CEBADA/CAMPAL	26,64	
2,20	4-oct.-19	CEBADA/CAMPAL	26	
2,21	4-oct.-19	MAIZ/CAMPAL	31,8	
2,22	6-mar.-20	CEBADA/CAMPAL	26,62	
2,23	13-mar.-20	MAIZ/CAMPAL	26,6	
2,24				
3,01				

Ilustración 2.- Excel de Gestión - Hoja de alimentos

MODELO VII				
HOJA DE MEDICAMENTOS				
Código de Explotación ES370081100891				
Especie: BOVINO				
15-jun.-20				
PAGINA ASIENTO	ECHA DE COMPR	CÓDIGO DE RECETA/DOCUMENTO COMERCIAL (1)	MEDICAMENTO o PIENSO MEDICAMENTOSO	OBSERVACIONES
2,02	31-ene.-16	1/391	FINADYNE	
2,03	1-jul.-16	1/3048	DRAXXIN	
2,04	1-jul.-16	1/3048	CATTLEMASTER	
2,05	1-jul.-16	1/3048	RISPOVAL	
2,06	31-ago.-16	1/3756	NUFLOR	
2,07	30-sep.-16	1/4244	NUFLOR VACUNO 250 ML	
2,08	30-sep.-16	1/4244	FINADYNE 250 ML	
2,09	30-nov.-16	1/5248 AGROPAL GANADERA	DRAXXIN 100 ML	
2,10	30-nov.-16	1/5248 AGROPAL GANADERA	NUFLOR VACUNO 250 ML	
2,11	30-nov.-16	1/5248 AGROPAL GANADERA	FINADYNE 250 ML	
2,12	30-nov.-16	N/05671 Nutersa	HIDROFAT 25 KG	
2,13	28-feb.-17	1/651 AGROPAL GANADERA	POLIBASCOL 10 100 ML	
2,14	28-feb.-17	1/651 AGROPAL GANADERA	IVOMEK F VACUNO 1000 ML	
2,15	28-feb.-17	1/651 AGROPAL GANADERA	DRAXXIN 100 ML	
2,16	28-feb.-17	1/651 AGROPAL GANADERA	DUPHALYTE 500ML	
2,17	31-oct.-18	1/4794 AGROPAL GANADERA	NUFLOR VACUNO 250 ML	
2,18	31-dic.-18	1/5804 AGROPAL GANADERA	DUPHAFRAL MULTI 100 ML	
2,19	28-feb.-19	1/616 AGROPAL GANADERA	CASTLEMASTER 4 25 DS	
2,20	30-abr.-19	1/1629 AGROPAL GANADERA	IVOMEK F VACUNO 1000 ML	
2,21	31-oct.-19	1/4673 AGROPAL GANADERA	DUPHAFRAL MULTI 100 ML	
2,22	19-may.-20	M-37-LE1474-338-2020	DUPHALYTE 500ML	
2,23	15-jun.-20	M-37-LE1474-338-2020	LACTATO DE RINGER BRAUN 1000 ML	
2,24	15-jun.-20	M-37-LE1474-391-2020	DUPHALYTE 500ML	
3,01				

Ilustración 3.- Excel de Gestión – Hoja de medicamentos

## Memoria del proyecto

Este sistema no es obligatorio llevarlo al día, pero si necesario, ya que toda esta información debe quedar registrada en algún lugar para mostrar en posibles inspecciones que se puedan realizar por parte de un control de explotaciones ganaderas de la junta de Castilla y León.

Los problemas que se planteaban por parte de la gestora eran:

- **Usabilidad:** El principal problema. Blanca, comentaba que era un sistema que podría mejorar en cuanto a la facilidad en la forma de usarse, a su manera de ver.
- **Portabilidad o Multiplataforma:** El sistema actual de gestión solo es accesible desde un computador, ya que se trata de un proyecto que se puede abrir únicamente con herramienta Microsoft Excel.
- **Seguridad de los datos:** Blanca mencionó que alguna vez perdió el Excel en el que estaba trabajando por borrarlo y no tener una copia de seguridad.

Los problemas que se encontraron por parte del desarrollador fueron:

- **Usabilidad:** Al igual que la gestora, este era el principal problema que se encontraba, las interfaces que ofrece la herramienta Microsoft Excel son muy básicas.
- **Portabilidad o Multiplataforma:** Lo mencionado en el anterior apartado de problemas.
- **Seguridad de los datos:** El sistema de guardado de información en local es el más simple.

### 2.1.- Sistema de gestión profesional

Este sistema es el oficial por la junta de Castilla y León. En este sistema se deberán tener registrados muchos de los datos de la organización: Fincas o explotaciones registradas en tu organización, animales pertenecientes a esas explotaciones, movimientos realizados, es decir, compras o ventas de animales que se han realizado con otras organizaciones, movimiento de animales de una explotación a otra de tu propia organización...

La información que recae en este sistema debe llevarse al día, actualizándola cada vez que surja cualquiera de los aspectos importantes mencionados anteriormente, ya que pueden suponer una amonestación por parte de la junta si se corrobora que los datos no son ciertos. Es la forma que tiene la junta de controlar las explotaciones ganadera de Castilla y León.

A continuación, se mostrarán algunas capturas de pantalla de este sistema:

# Memoria del proyecto



Ilustración 4.- Sistema Gestión Junta CyL



Ilustración 5.- Sistema Gestión Junta CyL. Iniciar Sesión

Selección de CEA-Especie					
Filtrar por CEA, ESPECIE, DNI, APELLIDOS o NOMBRE (o parte de sus caracteres):					
CEA	Especie	Municipio	CIF/NIF	Apellidos	Nombre
<a href="#">ES370080000129</a>	ÉQUIDOS	ALBA DE TORMES	B37346509	SANCHEZ BOYERO SL	
<a href="#">ES370080000179</a>	BÓVIDOS	ALBA DE TORMES	B37346509	SANCHEZ BOYERO SL	
<a href="#">ES370081100171</a>	BÓVIDOS	ALBA DE TORMES	007956541J	SANCHEZ HERNANDEZ	JOSE JAVIER
<a href="#">ES370081100481</a>	BÓVIDOS	ALBA DE TORMES	E37252012	BERNARDO SANCHEZ Y HERMANOS CB	
<a href="#">ES370081100482</a>	BÓVIDOS	ALBA DE TORMES	E37252012	BERNARDO SANCHEZ Y HERMANOS CB	
<a href="#">ES370081100891</a>	BÓVIDOS	ALBA DE TORMES	B37346509	SANCHEZ BOYERO SL	
<a href="#">ES370081100892</a>	BÓVIDOS	ALBA DE TORMES	B37346509	SANCHEZ BOYERO SL	
<a href="#">ES370084100891</a>	BÓVIDOS	ALBA DE TORMES	B37346509	SANCHEZ BOYERO SL	
<a href="#">ES370350220061</a>	CERDOS	ARMENTEROS	E37252012	BERNARDO SANCHEZ Y HERMANOS CB	
<a href="#">ES370351100591</a>	BÓVIDOS	ARMENTEROS	E37252012	BERNARDO SANCHEZ Y HERMANOS CB	
<a href="#">ES371821100021</a>	BÓVIDOS	MARTINAMOR	052411348Y	POLO GOMEZ	VICTOR MANUEL
<a href="#">ES372420000059</a>	CERDOS	PELAYOS	E37252012	BERNARDO SANCHEZ Y HERMANOS CB	
<a href="#">ES372420000065</a>	BÓVIDOS	PELAYOS	E37252012	BERNARDO SANCHEZ Y HERMANOS CB	
<a href="#">ES372421100181</a>	BÓVIDOS	PELAYOS	070856147R	MORETA AREVALO	FLORENTINO
<a href="#">ES373361100041</a>	BÓVIDOS	VALDEMIERQUE	070867095R	GARCIA ELICES	RICARDO

Ilustración 6.- Sistema Gestión Junta CyL. Fincas

# Memoria del proyecto

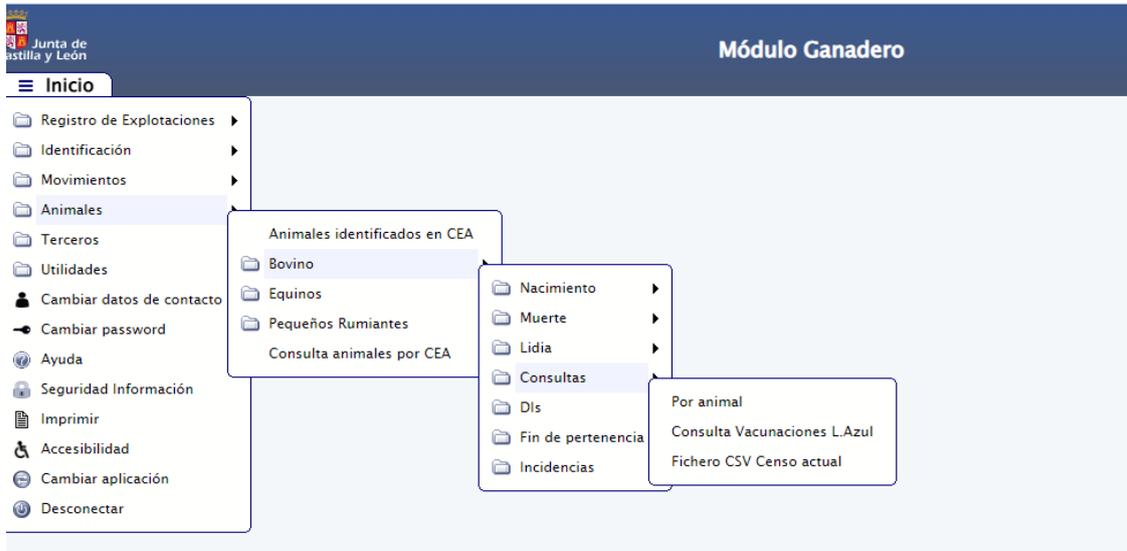


Ilustración 7.-Sistema Gestión Junta CyL. Consulta General

Consulta general de movimientos por CEA											
Criterios de selección											
CEA	ES370080000179										
Especie	BÓVIDOS										
Fecha Inicial	04-09-2019										
Fecha Final											
Tipo de Movimiento											
Primer Criterio Ordenación	Fecha de salida o muerte					Tipo Ordenación					
Segundo Criterio Ordenación	-----					Tipo Ordenación					
Página 1 de 137 (registros 1 a 10 de un total de 1362).											
Nº Registro	Nº Solicitud	F. Registro	Tipo	Tipo de Doc	Identificador Mto. o Nº Transacción o Animal	CEA Origen	Salida en origen o F. Muerte	CEA Destino	Llegada en destino o F. Nacimiento	Animales	Especie
20199001393849	370310077558	04-09-2019	Salida	Muerte	ES3700813070591	ES370080000179	04-09-2019		04-09-2019	1	BÓVIDOS
20199001407142	370310077729	06-09-2019	Salida	Documento sanitario de traslado IB (automatizada)	ES37008940000050716	ES370080000179	06-09-2019	ES052472000011	06-09-2019	12	BÓVIDOS
20199001407093	370310077726	06-09-2019	Salida	Documento sanitario de traslado IB (automatizada)	ES37008940000050716	ES370080000179	06-09-2019	ES050192000041	06-09-2019	9	BÓVIDOS
20199001436902	370310078162	12-09-2019	Entrada	G.O.S. II (automatizada)	ES3700201900003904	ES370081100892	12-09-2019	ES370080000179	12-09-2019	30	BÓVIDOS
20199001440433	370310078200	13-09-2019	Salida	Documento sanitario de traslado IB (automatizada)	ES37008940000050879	ES370080000179	13-09-2019	ES050192000041	13-09-2019	13	BÓVIDOS
20199001440433	370310078203	13-09-2019	Salida	Documento sanitario de traslado IB (automatizada)	ES37008940000050884	ES370080000179	13-09-2019	ES052472000011	13-09-2019	14	BÓVIDOS
20199001478767	370310078662	20-09-2019	Entrada	G.O.S. II (automatizada)	ES3700201900004024	ES370081100892	20-09-2019	ES370080000179	20-09-2019	46	BÓVIDOS
20199001513859	370310079082	27-09-2019	Salida	Documento sanitario de traslado IB (automatizada)	ES37008940000051211	ES370080000179	27-09-2019	ES052472000011	27-09-2019	13	BÓVIDOS
20199001513860	370310079083	27-09-2019	Salida	Documento sanitario de traslado IB (automatizada)	ES37008940000051212	ES370080000179	27-09-2019	ES050192000041	27-09-2019	12	BÓVIDOS
20199001547045	370310079500	03-10-2019	Entrada	G.O.S. II (automatizada)	ES3700201900004191	ES370081100892	02-10-2019	ES370080000179	02-10-2019	23	BÓVIDOS

El contenido de la columna "Animales" en rojo indica discordancia entre cantidades comunicadas en salida y llegada

Primera Anterior Siguiente Última Ir a página 1

Volver

Ilustración 8.- Sistema Gestión Junta CyL. Consulta de movimientos

Consulta animales del CEA ES370080000179 - BÓVIDOS por Titularidad										
Datos de la explotación										
CEA:	ES370080000179			Familia:	BÓVIDOS					
Titular:	SANCHEZ BOYERO SL									
Criterios de selección										
Especie (de los animales):	BÓVIDOS									
Estado Animal:		Raza:		Edad:		Incluir información partos:	No			
Sexo:		Filtro Hembras:	Con o sin hijos							
Incluir información Id. Electrónico:	No									
Tipo de búsqueda:	Por titularidad (04-09-2022 12:27:34)									
Identificador	NºMan.	Ident. Lidia	Sexo	Raza	F. Ident.	F. Nacim.	Ident. Madre	CEA Nacim.	F. Ini. Titu.	CEA Situ.
ES000605176974	6974	ES0806050991	HEMBRA	CONJUNTO MESTIZO	(vacío)	31-12-2021	ES080605099189	ES390960000173	27-07-2022	ES370080000179
ES000605279014	9014	ES0706041800	HEMBRA	CONJUNTO MESTIZO	(vacío)	10-06-2021	ES070604180006	ES390960000183	16-03-2022	ES370080000179
ES000605363028	3028	ES0306039322	HEMBRA	CHAROLESA	(vacío)	20-02-2022	ES030603932228	ES390960000183	27-07-2022	ES370080000179
ES000814162840	2840	(vacío)	HEMBRA	SALERS		09-10-2020	ES080811851666	ES370461100091	10-05-2022	ES370080000179
ES000814327656	7656	(vacío)	MACHO	CONJUNTO MESTIZO	15-11-2020	07-10-2020	ES080810545378	ES051080000021	16-03-2022	ES370080000179
ES000814506053	6053	(vacío)	MACHO	CONJUNTO MESTIZO	(vacío)	09-12-2020	ES030811580085	ES052467100171	16-03-2022	ES370080000179
ES000814516988	6988	(vacío)	HEMBRA	CONJUNTO MESTIZO	06-07-2021	04-07-2021	ES060810517325	ES052091100081	16-03-2022	ES370080000179
ES000814587901	7901	(vacío)	MACHO	CONJUNTO MESTIZO	(vacío)	11-01-2021	ES070812494019	ES370211100091	20-12-2021	ES370080000179
ES000814626943	6943	(vacío)	MACHO	CONJUNTO MESTIZO	09-02-2021	09-02-2021	ES090807446894	ES370351100071	17-05-2022	ES370080000179
ES000814707307	7307	(vacío)	HEMBRA	CONJUNTO MESTIZO	22-03-2021	22-03-2021	ES040809096700	ES050441100781	24-02-2022	ES370080000179
ES000814709847	9847	(vacío)	HEMBRA	CONJUNTO MESTIZO	15-02-2021	15-02-2021	ES070813042484	ES052171100201	07-12-2021	ES370080000179
ES000814710753	0753	(vacío)	HEMBRA	CONJUNTO MESTIZO	(vacío)	29-09-2021	ES020808089127	ES052281100411	22-04-2022	ES370080000179
ES000814714346	4346	(vacío)	HEMBRA	CONJUNTO MESTIZO	26-04-2021	25-04-2021	ES060810784642	ES050101100321	24-02-2022	ES370080000179
ES000814723905	3905	(vacío)	HEMBRA	CONJUNTO MESTIZO	06-10-2021	10-09-2021	FR5836353010	ES050190000352	10-05-2022	ES370080000179
ES000814732213	2213	(vacío)	HEMBRA	CONJUNTO MESTIZO	(vacío)	18-05-2021	ES050812210115	ES052171100201	22-04-2022	ES370080000179

Página 1 de 38 (registros 1 a 15 de un total de 560).

Primera Anterior Siguiente Última Ir a página 1

Volver

Ilustración 9.- Sistema Gestión Junta CyL. Consulta de animales por Finca

### Módulo Ganadero

ES370080000179 // BÓVIDOS

**Selección de Animal**

**Criterios de Selección**

Familia: BÓVIDOS

Especie: BÓVIDOS

Identificador (\*):

(\*) O bien sus últimos 4 dígitos

Siguiente Cancelar

Ilustración 10.- Sistema de Gestión CyL. Búsqueda de Animales

Consulta de Animales							
Criterios de selección							
Familia	BÓVIDOS						
Especie	BÓVIDOS						
Explotación de localización							
Explotación de nacimiento							
Identificador	ES080815474598						
Identificador	Id Madre	Raza	Sexo	F Nacim	Expl Nacim	Ubicacion	F Entrada
ES080815474598	ES070809552733	CONJUNTO MESTIZO	MACHO	25/03/2022	ES371821100021	ES370080000179	2022-07-27 08:00:01
Página 1 de 1 (animales 1 a 1 de un total de 1).							

Volver

Ilustración 11.- Sistema de Gestión CyL. Consulta de Animal individual

## Memoria del proyecto

Los problemas que se planteaban por parte de la gestora eran:

- **Usabilidad:** El principal problema de nuevo. Blanca, comentaba que era un sistema no demasiado intuitivo en la manera correcta de usarse.
- **Portabilidad o Multiplataforma:** Este sistema de gestión solo es accesible desde un computador y a través del navegador.

Los problemas que se encontraron por parte del desarrollador fueron:

- **Usabilidad:** Al igual que la gestora, este era el principal problema que se encontraba, las interfaces que se encontraron en este sistema de gestión de la Junta de Castilla y León eran realmente antiguas y no destacaban por ser intuitivas.
- **Portabilidad o Multiplataforma:** No se puede acceder al sistema desde otro dispositivo.
- **Diseño Responsive:** El sistema no tenía un diseño desarrollado para poder usarse en distintas resoluciones.

Por tanto, la necesidad ya estaba creada, concretamente se necesita desarrollar un sistema de gestión que cumpla ciertos requisitos de usabilidad, portabilidad, seguridad de los datos y diseño responsive.

### 3.- Estado del arte

Antes de comenzar con el desarrollo del sistema, se realizará una investigación documental denominada “estado del arte”, que trata del estudio del conocimiento acumulado dentro de un área específica. (Gómez, s.f.)

En este apartado desarrollaremos ese estudio, en concreto, se realizará una búsqueda y análisis de sistemas o aplicaciones existentes en el mercado que se encarguen de la gestión de una explotación agropecuaria. (Vela, 2022)

#### 3.1.- VacApp:

Aplicación móvil de pago diseñada para la gestión del ganado bovino. Permite analizar el crecimiento del rebaño, las ventas y las compras. (VacApp, Agosto)

Además de gestionar rebaños, permite llevar un históricos para ver tendencias, simplificar el papeleo del saneamiento...



Ilustración 12.- Aplicación VacApp

La principal ventaja que se encuentra es la de poder consultar los datos de forma offline, eso dará una mayor disponibilidad a los datos del sistema, además, teniendo en cuenta que en el campo no siempre hay cobertura.

Otra ventaja es la posibilidad de importar Excels donde se tenga ubicada toda la información registrada de la gestión de la empresa hasta el momento.

Una de las principales desventajas está en que no es un sistema multiplataforma. Es una app, lo que impide la gestión desde un ordenador. Aunque es importante poder acceder desde el móvil ya que te da una mayor disponibilidad, no hay que olvidar que la mayor parte de las gestiones se realizarán por ordenador. El uso de la aplicación por móvil se delega en consultar rápidas o gestiones al momento.

### 3.2.- Bovi Control:

Software de pago para la gestión de vacuno de carne desarrollado en colaboración con ganaderos y veterinarios del sector. (GanderosOnline, 2022)

Permite entrada de datos de manera fácil y rápida. Programa muy completo que permite dar respuesta a las múltiples necesidades del sector: control de reproductoras, engorde, control lechero, libros de registro, versión Multi-granjas, control económico...

Dispone de Informes donde comparar las características productivas de cada granja. Analiza el estado de la granja en cuestión de minutos.

Animal	Reproductora	Fecha	Día	Cu	Salo 1	Salo 2	Salo 3	Dog	IDC	Día	Fecha	Fecha de	Días	Día	Viv	Mu	Mom	Nac	Destr	Fecha	M				
5094	03/07/06	1	m	1	6	5000	6	6	Posal	0	0	25/10/06	30/10/06	119	2	5	0	0	5	3	16/11/06	1			
3613	03/07/06	1	m	1	6	5000	6	6	Posal	17	0	25/10/06	28/10/06	117	7	5	0	0	5	3	16/11/06	7			
5092	03/07/06	1	m	1	6	5000	6	6	Posal	0	0	25/10/06	22/10/06	111	1	15	0	0	15	7	23/11/06	1			
5093	03/07/06	1	m	1	6	5000	6	6	Posal	0	0	25/10/06	25/10/06	115	5	7	1	0	8	11	23/11/06	1			
11492	02/07/06	1	m	3	5	5000	5	5	Posal	4	0	24/10/06	23/10/06	113	2	13	0	0	16	11	23/11/06	6			
28	01/07/06	6	f	1	6	5000	5	5000	6	5	Posal	5	0	23/10/06	21/10/06	112	7	14	3	0	17	1	16/11/06	2	
10150	01/07/06	6	f	1	6	5000	6	5000	6	2	5	Posal	5	0	23/10/06	22/10/06	113	1	13	1	0	14	3	23/11/06	8
12523	24/06/06	6	f	1	6	5000	5	5000	6	5	Ripal	5	19	16/10/06											
11122	24/06/06	6	f	1	6	5000	6	5000	6	5	Ripal	5	25	16/10/06											
25	24/06/06	6	f	1	6	5000	5	5000	5	5	Posal	5	0	16/10/06	12/10/06	110	5	15	0	0	15	10	03/11/06	2	
26	24/06/06	6	f	1	6	5000	5	5000	6	5000	5	Posal	5	0	16/10/06	11/10/06	109	4	0	6	0	6	0	24/10/06	2
24	24/06/06	6	f	2	6	5000	5	5000	6	5	Ripal	5	19	16/10/06											
12522	21/06/06	3	f	1	6	5000	6	5000	6	6	Posal	5	0	13/10/06	17/10/06	118	3	0	0	8	10	09/11/06	4		
36	21/06/06	3	f	1	6	5000	6	5000	6	6	Posal	5	0	13/10/06	13/10/06	114	6	9	0	4	13	8	05/11/06	7	
12185	21/06/06	3	f	1	6	5000	6	5000	6	6	Posal	5	0	13/10/06	14/10/06	115	7	0	1	0	9	7	03/11/06	5	
12615	21/06/06	3	f	2	6	5000	6	5000	6	6	Posal	6	0	13/10/06	13/10/06	114	6	11	0	3	14	0	26/10/06	4	
682	21/06/06	3	f	1	6	5000	6	5000	6	6	Ripal	5	22	13/10/06											
11460	21/06/06	3	f	1	6	5000	6	5000	6	6	Posal	5	0	13/10/06	14/10/06	115	7	10	6	0	16	3	03/11/06	6	
12623	20/06/06	2	f	3	6	5000	6	5000	6	6	Posal	6	0	12/10/06	14/10/06	116	7	10	1	0	11	3	03/11/06	3	
4	20/06/06	2	f	1	6	5000	6	5000	6	6	Ripal	4	42	12/10/06											
12	20/06/06	2	f	1	6	5000	5	5000	6	5	Posal	4	0	12/10/06	13/10/06	115	6	9	0	0	9	3	03/11/06	3	
5002	17/06/06	6	f	2	6	5000	6	5000	6	6	Posal	0	0	09/10/06	12/10/06	117	5	6	1	0	7	8	03/11/06	1	
19696	14/06/06	2	f	7	4	4000	4	4000	4	4	Posal	10	6	02/10/06	11/10/06	118	4	6	1	0	7	4	01/11/06	1	

Ilustración 13.- Captura 1 BoviControl



Ilustración 14.- Captura 2 BoviControl

Se encuentran algunas ventajas como la implementación de la generación automática de los libros de explotación que los ganaderos deben mostrar en las inspecciones o el control específico y exhaustivo de los costes.

A su vez, también se encuentran algunas desventajas, una de ellas es que el software se centra sobre todo en la gestión de costes y no tanto en la gestión de animales y explotaciones. La interfaz no tiene un grado alto de usabilidad.

## 4.- Objetivos

En este apartado se expondrán tanto los objetivos que debe cumplir el sistema para llevar a cabo el desarrollo de este proyecto, como los objetivos personales que se desean alcanzar con la realización de este.

### 4.1.- Objetivos del sistema

El objetivo principal del sistema es el desarrollo de una aplicación web que permita realizar las tareas necesarias para realizar correcta gestión de una explotación ganadera. Todas estas tareas se pueden englobar en diferentes paquetes de objetivos.

A continuación, se mostrarán esos paquetes de objetivos mencionados, además de otros que se corresponden con el almacenamiento de históricos de gestión y estadísticas:

- **Gestión de Usuarios:** El sistema deberá permitir registrar usuarios, así como la modificación de sus datos una vez registrados. También debe controlar la autenticación de los usuarios ya registrados cuando deseen acceder a través del inicio de sesión. Además, debe gestionar todos los permisos de acceso a la información de las distintas organizaciones que los Administradores de estas concedan a los usuarios Trabajadores.
- **Gestión de Organizaciones:** El sistema deberá permitir crear organizaciones con su propio administrador, con sus trabajadores y sus fincas, animales, alimentos y medicamentos propios.
- **Gestión de Fincas:** El sistema deberá permitir realizar acciones como crear, leer, modificar o eliminar las diferentes fincas de la organización a la que pertenezca el usuario.
- **Gestión de búsquedas de fincas:** El sistema deberá permitir buscar las fincas registradas en la organización y acceder a ellas para su gestión.
- **Gestión de Animales:** El sistema deberá permitir realizar acciones como crear, leer, modificar o eliminar, añadir a fincas, eliminar o mover de fincas animales de la organización a la que pertenezca el usuario.
- **Gestión de búsqueda de animales:** El sistema deberá permitir buscar los animales registrados en la organización y acceder a ellos para su gestión.
- **Gestión de Alimentos:** El sistema deberá permitir realizar acciones como crear, leer, modificar o eliminar alimentos a la organización a la que pertenezca al usuario.
- **Gestión de Medicamentos:** El sistema deberá permitir realizar acciones como crear, leer, modificar o eliminar medicamentos a la organización a la que pertenezca al usuario.

## Memoria del proyecto

- **Almacenamiento y visualización de históricos de gestión:** El sistema deberá almacenar un histórico de las transacciones realizadas por la organización en cada una de sus fincas tales como altas, bajas o modificaciones.
- **Almacenamiento y visualización de estadísticas:** El sistema deberá almacenar un histórico de las estadísticas relativas a las fincas y animales.

### 4.1.- Objetivos personales

En este subapartado expresaré cuales han sido los objetivos personales que me han llevado a realizar este proyecto.

El principal punto marcado al principio del proyecto para llevar a cabo el desarrollo de este proyecto fue el de tener la posibilidad de crear un servicio con utilidad real y planteada de una forma profesional. Al tener contacto con un cliente real, la sensación de poder cubrir una necesidad real con mi trabajo ha estado presente a lo largo de todo el proceso de desarrollo.

Otro de los puntos más importantes ha sido la posibilidad de crear desde cero un servicio web, llevaba un tiempo con la ilusión de poder realizar un proyecto así, en el que en mí recaía todo el peso. Probablemente uno de los objetivos con el que más miedo me enfrentaba a él, poner a prueba mi capacidad de enfrentarme a problemas que pueden surgir a lo largo del desarrollo de un proyecto así.

La tecnología utilizada, concretamente el framework de Angular al principio era desconocida, de hecho, se planteó utilizar otro tipo de framework, pero en la búsqueda de la elección correcta, Angular fue llamando cada vez más mi atención hasta un punto que logró crearme la ilusión de trabajar con él.

## 5.- Conceptos teóricos

### 5.1.- Framework Angular

Angular es una plataforma de desarrollo, construida sobre TypeScript. Es un framework basado en componentes para crear aplicaciones web escalables. Una colección de bibliotecas bien integradas que cubren una amplia variedad de características, que incluyen enrutamiento, administración de formularios, comunicación cliente-servidor y más. Un conjunto de herramientas para desarrolladores que permiten desarrollar, compilar, probar y actualizar el código fuente de la aplicación. (QualityDevs, 2022)

Angular es un Framework de JavaScript de código abierto escrito en TypeScript. Su objetivo principal es desarrollar aplicaciones de una sola página. Google se encarga del mantenimiento y constantes actualizaciones de mejoras para este framework. (HiberusBLog, 2022)

Un framework es una estructura de soporte conceptual y tecnológica definida, generalmente, con artefactos o módulos de software específicos, que pueden servir como base para la organización y desarrollo de software.

### 5.2.- Firebase

Firebase se trata de una plataforma móvil creada por Google, cuya principal función es desarrollar y facilitar la creación de aplicaciones de elevada calidad de una forma rápida, con el fin de que se pueda aumentar la base de usuarios.

La plataforma está subida en la nube y está disponible para diferentes plataformas como iOS, Android y web. Contiene diversas funciones para que cualquier desarrollador pueda combinar y adaptar la plataforma a medida de sus necesidades. (Cardona, 2022)

Firebase se inició cuando Google la compró en 2014, y seguidamente la fue mejorando.

Los servicios utilizados para el desarrollo del proyecto son:

- **Authentication:** Firebase Authentication proporciona servicios de backend, SDK fáciles de usar y bibliotecas de IU ya elaboradas para autenticar a los usuarios en tu app. Admite la autenticación mediante contraseñas, números de teléfono, o a través de un correo. (Firebase, 2022)
- **RealTime Database:** Firebase Realtime Database es una base de datos NoSQL alojada en la nube que te permite almacenar y sincronizar datos entre tus usuarios en tiempo real. (Firebase, 2022)

### 5.3.- Diseño Responsive

Un diseño web responsive es el que es capaz de adaptarse a pantallas de diferentes tamaños con un solo sitio web. El sistema detecta automáticamente el ancho de la pantalla y a partir de ahí adapta todos los elementos de la página, desde el tamaño de letra hasta las imágenes y los menús, para ofrecer al usuario la mejor experiencia posible. (Sol González, 2022)

## 5.4.- Librería o bibliotecas

Conjunto es un conjunto de archivos que se utiliza para desarrollar software que han realizadas con alguno de los distintos lenguajes de programación existentes. Suele estar compuesta de código y datos, y su fin es ser utilizada por otros programas de forma totalmente autónoma. Simple y llanamente, es un archivo importable. (DevCamp , 2022)

Ofrece poder extender la funcionalidad del servicio principal ofreciéndole un conjunto de implementaciones funcionales.

## 6.- Técnicas y herramientas

En este apartado se mostrarán y presentarán cada una de las técnicas metodológicas o herramientas utilizadas para concretar el desarrollo del proyecto.

### 6.1.- Framework:

#### 6.1.1.- Angular:

Angular es un framework opensource desarrollado por Google para facilitar la creación y programación de aplicaciones web de una sola página, las webs SPA (Single Page Application). Se basa principalmente en el uso de componentes que son elementos que encapsulan código HTML, CSS y TypeScript. (QualityDevs, 2022)

Una de sus características es que separa completamente el frontend y el backend en la aplicación, evita escribir código repetitivo y mantiene todo más ordenado gracias a su patrón MVC.

Este framework fue el elegido para desarrollar el sistema por el aumento de su popularidad en los últimos años y las buenas palabras que tienen los desarrolladores hacia él. Su popularidad facilita la posibilidad de encontrar contenido y tutoriales que facilitan el desarrollo, ya que no lo he usado previamente.

### 6.2.- Lenguajes

#### 6.2.1.- HTML

Se trata del componente más básico de la Web. Se trata de un lenguaje de etiquetas de hipertexto (HyperText Markup Language), define el significado y la estructura del contenido web. (Developer, 2022)

Se ha utilizado en los componentes angular para definir la interfaz de cada una de las vistas que se podrán ver en nuestro servicio web especificando cada uno de sus elementos.

#### 6.2.1.- CSS

CSS son las siglas en inglés para «hojas de estilo en cascada» (Cascading Style Sheets). Básicamente, es un lenguaje que maneja el diseño y presentación de las páginas web, es decir, cómo lucen cuando un usuario las visita. Funciona junto con el lenguaje HTML que se encarga del contenido básico de las páginas. (Diego Santos, 2022)

## Memoria del proyecto

Se ha utilizado para dar el diseño necesario a la estructura marcada por cada uno de los ficheros HTML. Como ya se ha mencionado, cada componente tiene un HTML y CSS encapsulado.

### 6.2.1.- TypeScript

Typescript es un superset, es decir, es una tecnología que puede ejecutar programas de la tecnología de la que es superset, en este caso de JavaScript. Esto significa que los programas de JavaScript son programas válidos de TypeScript, a pesar de que TypeScript sea otro lenguaje de programación. Está desarrollado por Microsoft. (Hernández, 2022)

Los navegadores solo entienden el código escrito en JavaScript, pero TS tiene la capacidad de realizar una “transpilación” del código a JavaScript. TypeScript tiene ventajas a nivel de tipados, donde se pueden crear tipos genéricos o interfaces. (Wikipedia, 2022)

Al igual que HTML y CSS, forma parte de cada uno de los componentes del sistema. Se utiliza durante en cada uno de ellos, además, de los servicios y modelos. Es el lenguaje principal.

### 6.2.1.- JSON

La notación de objeto de JavaScript (JavaScript Object Notation) es un formato de texto sencillo basado en un subconjunto del lenguaje de programación JavaScript, pero completamente independiente del mismo. Resulta un formato sencillo de leer y escribir para los humanos y simple de interpretar y generar por las máquinas. (Hostinger, 2022)

Es el formato de texto que utiliza la RealTime Database de Firebase para almacenar y recuperar los datos de ella.

### 6.2.1.- UML

El Lenguaje Unificado de Modelado (Unified Modeling Language) fue creado para forjar un lenguaje de modelado visual común y semántica y sintácticamente rico para la arquitectura, el diseño y la implementación de sistemas de software complejos, tanto en estructura como en comportamiento. (LucidChart, 2022)

Ha sido usado para la realización de diagramas de diferentes tipos en los apartados de ingeniería software del proyecto.

## 6.3.- Librerías:

### 6.3.1.- Firebase:

Firebase es una plataforma de Google que permite crear aplicaciones en forma rápida con una serie de servicios disponibles. Realtime Database nos permite contar con un sistema de almacenamiento de datos no-relacional en tiempo real. La librería de Firebase nos da pie a usar funciones Firebase dentro de nuestro proyecto. (Firebase, 2022)

Se ha utilizado para funciones de registro, inicio de sesión, cambio de contraseña, editar el perfil...

### 6.3.1.- Router:

Librería necesaria para aplicar el motor de Routing de Angular (RouterModule, Routes) en nuestro sistema de angular. El cual funciona asociando una URL concreta con su correspondiente controlador, permitiendo así trabajar con URL semánticas. Las URL semánticas se indexan mejor en los buscadores, siendo, además, más adecuadas para el posicionamiento web. (Ro, 2022)

## Memoria del proyecto

Se ha utilizado para redirigir de una url a otra y para generar los componentes que deseamos por cada una de las url de nuestro servicio.

### 6.3.1.- Bootstrap:

Bootstrap es una librería de estilos CSS, es decir, es un conjunto de archivos CSS que se añaden a los distintos proyectos de aplicaciones o servicios web, que permiten tener una serie de estilos ya preparado para utilizar. Este tipo de librerías CSS suelen incluir estilos para los elementos más comunes de una página web, como, por ejemplo, botones, tarjetas, navbars, etc. Además, tiene una serie de estilos para crear columnas fácilmente. Es muy conocida. (Bootstrap, 2022)

Se ha utilizado para la cereación de las vistar en los ficheros HTML y CSS.

### 6.3.1.- Chart.js:

Chart.js es una librería javascript open-source ideal para la visualización de datos en gráficos, como puede ser de barras, circular, líneas... donde destaca su sencillez. (Adictos al Trabajo, 2022)

Se ha utilizado para la generación de gráficos de estadísticas en uno de los apartados de la aplicación web del proyecto.

### 6.3.1.- ng2-charts.js:

Ng2-charts es un paquete de directivas de Angular para Chart.js que se utiliza para crear graficas en Angular a partir de la versión 8. (Rckstr, 2022)

Se ha utilizado para la generación de gráficos de estadísticas en uno de los apartados de la aplicación web del proyecto.

## 6.4.-Herramientas de compilación y hosting:

### 6.4.1.- Node.Js:

Node.Js es un entorno de tiempo de ejecución de JavaScript de código abierto y multiplataforma que se ejecuta del lado del servidor en una arquitectura orientada a eventos y basado en el motor V8 de Google (Wikipedia, 2022). Fue creado con el enfoque de ser útil en la creación de programas de red altamente escalables. Es necesario para poder realizar el proyecto en el framework de Angular. (Acibeiro, 2022)

La mayoría de las herramientas utilizadas en el desarrollo de proyectos con este lenguaje lo necesitan para funcionar.

### 6.4.1.- Npm:

Es el sistema de gestión de paquetes por defecto para Node.js, un entorno de ejecución para JavaScript. La utilidad de línea de comando del administrador de paquetes Node permite que node.js funcione correctamente.

Permite la instalación de paquetes, bibliotecas, dependencias y plugins disponibles como ng2-charts, charts.js... (Hostinger, 2022)

## 6.5.- Herramientas de control de versiones:

### 6.5.1.- GitHub

Github es un portal creado para alojar el código de las aplicaciones de cualquier desarrollador. La plataforma está creada para que los desarrolladores suban el código de sus aplicaciones y herramientas, y que como usuario no sólo puedas descargarte la aplicación, sino también entrar

## Memoria del proyecto

a su perfil para leer sobre ella o colaborar con su desarrollo. Su principal característica es que maneja un sistema de control de versiones para que los desarrolladores puedan administrar su proyecto. (Fernández, 2022)

Se ha utilizado para almacenar el proyecto en la nube y poder gestionar las distintas versiones.

### 6.5.1.- SourceTree

SourceTree simplifica la forma en que interactúa con sus repositorios de Git para que pueda concentrarse en la codificación. Permite visualizar y administrar sus repositorios a través de una sencilla interfaz de usuario que provee la aplicación. (rchavez, 2022)

Se ha utilizado para subir los archivos modificados o nuevos al repositorio de Github.

## 6.6.- Herramientas CASE

### 6.6.1.- Visual Paradigm

Es una herramienta que permite modelar diagramas UML de una manera intuitiva y sencilla a partir de los requisitos. Además, permite establecer relaciones entre los componentes y clases de los distintos diagramas, facilitando el mantenimiento de la trazabilidad. (Captterra, 2022)

Utilizado para realizar diagramas en la fase de ingeniería software.

### 6.6.1.- EZEstimate

Es una herramienta para la estimación del esfuerzo del desarrollo de un proyecto software. Esta herramienta realiza cálculos utilizando las fórmulas de los Puntos de Casos de Uso sin Ajustar (UUCP), teniendo en cuenta la interacción de los actores (UAW), la complejidad de los casos de uso (UUCW), los factores de complejidad técnica (TCF) y los factores de complejidad del entorno (ECF). Crea un informe de la estimación resultante.

Ha sido utilizado en el Anexo I para la estimación de coste y esfuerzo.

### 6.6.1.- Microsoft Project

Es un programa creado por Microsoft, que reúne un conjunto de herramientas para gestionar proyectos y portafolios. Permite la planificación y gestión de proyectos, además que permite generar un diagrama de Gantt que te permite visualizar las tareas que se han identificado y establecido en forma de barras en un periodo de tiempo determinado. (SergioOswaldo, 2022)

Se ha utilizado en el Anexo I para la planificación de tareas.

## 6.7.-Herramienta de documentación

### 6.7.1.- Compodoc:

Compodoc es una herramienta de código abierto que facilita la generación de documentación para un proyecto Angular. El resultado es un diseño de sitio web limpio y receptivo que incluye funcionalidad de búsqueda y hermosos temas. (Ediciones-Eni, 2022)

Se ha utilizado para generar la documentación del código de la aplicación web.

## 6.8.- Entorno de desarrollo:

### 6.8.1.- Visual Studio Code:

Se trata de un editor de código fuente desarrollado por Microsoft. Es software libre y multiplataforma. Tiene una buena integración con Git, cuenta con soporte para depuración

## Memoria del proyecto

de código, y dispone de un sinnúmero de extensiones, que básicamente te da la posibilidad de escribir y ejecutar código en cualquier lenguaje de programación. (OpenWebinars, 2022)

### 6.9.- Herramientas de diseño gráfico:

#### 6.9.1.- Adobe XD:

Adobe XD es un software usado para crear o diseñar interfaces, tanto para páginas webs como para aplicaciones. Te permite compartirlo con otros usuarios de forma muy sencilla. (Edix, 2022)

Se ha utilizado para crear el primer boceto del diseño de las vistas del servicio web.

#### 6.9.1.- Adobe Photoshop CS6:

Photoshop es un programa informático de edición de imágenes, desarrollado por Adobe. (Significados.com, 2022)

Se ha utilizado para la edición de algunas de las imágenes que se observan el servicio web, así como el diseño del logo.

#### 6.9.1.- Paletton:

Aplicación online que te permite seleccionar el color más adecuado para tus diseños con posibilidad de combinar degradados o tonalidades complementarias, generando automáticamente una paleta de colores a partir de un color. (TysMagazine, 2022)

Se ha utilizado para genera la paleta de colores que será usada en las interfaces del servicio web.

## 7.- Aspectos relevantes del desarrollo

En ese apartado se detallarán las partes más importantes y representativas de las distintas fases de desarrollo del proyecto.

### 7.1.- Marco de trabajo

Las pautas seguida en el proceso de desarrollo y trabajo en el proyecto han sido las marcadas por el marco de trabajo del Proceso Unificado. Este marco es de carácter genérico y puede especializarse para una gran variedad de sistemas software por estar basado en componente y gracias a su flexibilidad (Peñalvo, 2022) . Sus características principales son:

- **Iterativo e incremental:** El proceso unificado es un marco de trabajo iterativo e incremental que se divide en diferentes fases las tareas, estas tareas son: Inicio, Elaboración, Construcción y Transición. Cada una de estas fases se divide a su vez en distintas iteraciones, el número de iteraciones varía en función de la necesidad. Cada iteración tiene como resultado un incremento del producto que mejora el resultado conseguido en la iteración anterior.

- **Dirigido por casos de uso:** Los casos de uso proporcionan una estructura para expresar requisitos funcionales y definir los contenidos en cada iteración. De esta forma los casos de uso se utilizan para capturar la funcionalidad, constituyendo así la base del Proceso Unificado.
- **Centrado en la arquitectura:** Como ya se ha mencionado en este mismo punto, se asume que no existe un único modelo o proceso universal. Existen diferentes formas que cubren la arquitectura del sistema software. Por lo que se partirá de un primer modelo de arquitectura que se irá mejorando y refinando con el paso de las iteraciones, obteniendo finalmente el producto final.

Como ya se ha mencionado, el proceso unificado permite por tanto dividir en diferentes fases las tareas, estas fases son:

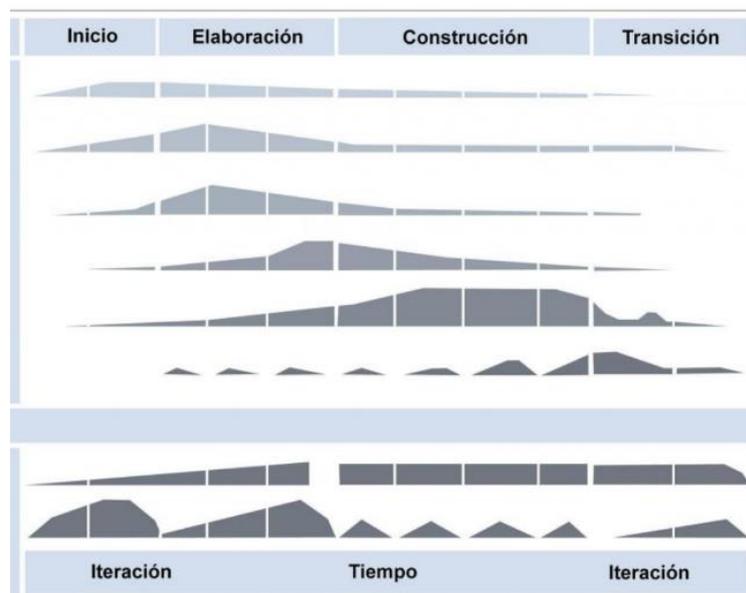


Ilustración 15.- Fases Proceso Unificado

- **Inicio:** Se define el alcance del proyecto y se desarrollan los casos de negocio. Se realiza la búsqueda y detección de necesidades, así como la de los objetivos del sistema, se presenta una estimación de costes y viabilidad.
- **Elaboración:** Se planifica el proyecto, especificándose más en detalle los casos de uso y se realiza el diseño de la arquitectura del sistema.
- **Construcción:** Se lleva a cabo el proceso de desarrollo y construcción del sistema.
- **Transición:** Se corresponde con la etapa final, por tanto, en el final de la última iteración de la transición, el sistema debe estar listo para ser entregado, pasando por tanto por esta fase por pruebas para comprobar que puede ser utilizado sin problemas.

## Memoria del proyecto

Las tareas específicas que se han realizado en las distintas iteraciones y fases las veremos en el punto 7.3 - *Planificación temporal*.

### 7.2.- Estimación del coste y esfuerzo

La estimación de coste y esfuerzo fue una de las primeras tareas en ser realizadas. Es necesario tener los datos de la estimación sobre la duración del proyecto antes de comenzar con la planificación, porque así en función de los resultados de la estimación podremos asignar más o menos tiempo a las tareas.

Para calcular la estimación se utilizó el método de los *Use Case Points (UCP)* o Puntos de Casos Uso. Los cuales se basan en la funcionalidad representada en forma de casos de uso, considerando actores, escenarios y factores técnicos y del entorno.

La función para calcular la estimación a través de estos puntos es la siguiente:

$$UCP = UUCP * TCF * ECF$$

#### 7.2.1.- UUCP - Punto de Casos de Uso Sin Ajustar

Los UUCP se corresponden con los puntos de casos de uso sin ajustar, que a su vez se calculan con el sumatorio del peso de los casos de uso sin ajustar UUCW y la de peso de los actores sin ajustar UAW.

##### 7.2.1.1.- UAW - Interacción de los Actores

El peso de los actores sin ajustar se calcula como la suma de los pesos asignado a cada actor en función de la complejidad que tenga cada uno.

Se definen 3 niveles de complejidad:

Complejidad	Descripción	Factor
Simple	Si el actor es un sistema y la aplicación se comunica con él mediante una API	1
Medio	Si el actor es un sistema y la aplicación se comunica con él mediante un protocolo	2
Complejo	Persona con una interfaz gráfica	3

Tabla 1.- Tabla de complejidad de actores

##### 7.2.1.2.- UUCW - Complejidad de los Casos de Uso

Calcula la complejidad de los casos de uso considerando el número de los pasos de los escenarios, la complejidad de la interfaz de usuario, número de entidades de la base de datos a las que accede y el número de clases de análisis.

Existen 3 niveles de complejidad para los casos de uso.

Tipo de caso de Uso	Descripción	Factor
Simple	3 transacciones o menos	5
Medio	De 4 a 7 transacciones	10
Complejo	Más de 7 transacciones	15

Tabla 2.- Tabla complejidad Casos de Uso

En el Anexo I – Planificación temporal se podrá encontrar cada uno de los valores asignados a los casos de uso y a los actores de los distintos paquetes del sistema.

## Memoria del proyecto

### 7.2.2.- TCF - Factores de Complejidad Técnica

Existen trece factores de complejidad técnica. Todos y cada uno de estos factores se suman para conseguir el valor final de los TCF:

$$TCF = C_1 + C_2 \sum_{i=1}^{13} W_i F_i$$

A cada uno de los treces factores de complejidad técnica se les asignó un valor del peso acorde a su impacto y otro valor de la complejidad percibida.

Factor	Peso	Complejidad	Motivo
T1: Sistemas distribuidos		1	Se trata de un sistema cliente-servidor, donde el encargado de la gestión del servidor no será el desarrollador.
T2: Rendimiento		3	El servicio debe garantizar un buen rendimiento donde interesa tener tiempos de respuesta cortos, pero sin que resulte fatal un retraso muy alto para una buena experiencia de usuario.
T3: Eficiencia del usuario final		5	Este factor es uno de los más importantes ya que al ser un servicio destinado al trabajo, se entiende que será usado de forma diaria y prácticamente constante, por lo que hay que lograr la satisfacción del usuario.
T4: Procesamiento interno complejo		2	El sistema no realizará cálculos muy complejos. Únicamente búsqueda de animales, fincas en base de datos.
T5: Reusabilidad		2	Aunque Angular esté destinado a ser reutilizable, no será uno de los objetivos principales del proyecto, aunque si es importante la reutilización de ciertos componentes.
T6: Facilidad de instalación		0	Este punto no es importante. El servicio podrá ser accesible desde cualquier navegador.
T7: Facilidad de uso		5	Este factor coincidirá con uno de los objetivos principales del sistema, ya que busca mejorar la usabilidad de servicios existentes.
T8: Portabilidad		4	Debe ser importante ya que es necesario que cualquier usuario pueda acceder al sistema desde cualquier parte o dispositivo.
T9: Facilidad de cambio		3	Sería moderadamente importante valorar este punto y por eso se construirá en módulos, pero no es un factor esencial.
T10: Concurrencia		4	Factor significativo ya que se entiende que los usuarios podrían utilizar el servicio de forma concurrente de forma constante.
T11: Características especiales de seguridad		4	Es significativo tener en cuenta este punto, ya que ciertas informaciones no deben ser accesibles por usuarios, aunque no se tratará con información personal de los usuarios demasiado sensible.
T12: Acceso directo a terceras partes		3	Un factor moderadamente importante ya que el sistema deberá poder acceder a su APIs de Firebase.
T13: Se requiere entrenamiento especial del usuario		0	Ninguna influencia tendrá este factor ya que debería ser un sistema fácil de usar de primeras.

Tabla 3.- TCF

### 7.2.3.- ECF - Factores de Complejidad del Entorno

Existen ocho factores de la complejidad del entorno. Todos y cada uno de estos factores se suman para conseguir el valor final de los ECF:

$$ECF = C_1 + C_2 \sum_{i=1}^8 W_i F_i$$

## Memoria del proyecto

A estos factores de complejidad el entorno se les asignó un valor del peso acorde a su impacto y otro valor de la relevancia que varía en 0 y 5, siendo 0 ninguna relevancia y 5 esencial.

Factor	Peso	Relevancia	Motivo
E1: Familiaridad con UML		2	La experiencia en UML se resume al aprendizaje y uso académico por lo que moderadamente reducida.
E2: Trabajadores a tiempo parcial		5	Durante parte de este proyecto estaré cursando el grado de Ingeniería Informática en su último año y un trabajo a tiempo completo.
E3: Capacidad de los análisis		2	Muy reducida, se limita al igual que en el E1, al ámbito académico.
E4: Experiencia en la aplicación		2	Mínima, muy mínimas nociones de HTML y CSS y ninguna en Angular, TS, Firebase o JS.
E5: Experiencia en orientación a objetos		3	Experiencia media, se limita al ámbito académico, pero con mucha más experiencia que E1 o E3, ya que se han realizado más proyectos a lo largo de la carrera.
E6: Motivación		5	Gran motivación y el punto más relevante para empezar el desarrollo de este sistema.
E7: Dificultad del lenguaje de programación		2	Angular no tiene una dificultad alta, por lo que resultará moderadamente sencillo.
E8: Estabilidad de los requisitos		4	Muy alta, los requisitos deberán ser muy estables y no necesitarán demasiados cambios a lo largo del desarrollo

Tabla 4.- ECF

Para calcular la estimación final se ha utilizado la herramienta EZEstimate que en función de los valores que hemos asignado tanto a los UAW - Interacción de los Actores, UUCW - Complejidad de los Casos de Uso, TCF - Factores de Complejidad Técnica y ECF - Factores de Complejidad del Entorno realizará la fórmula ya mencionada de los Puntos de Casos de Uso.

Estimation Summary	
UAW	14
UUCW	190
UUPC = UAW + UUCW	204
TFactor	40
EFactor	15
TCF = 0.6 + (.01*TFactor)	1
EF = 1.4 + (-0.03*EFactor)	0,95
UCP = UUPC*TCF*EF	193,8
Total Effort@ [g] Hrs/UCP	1550,4

Ilustración 16.- Cálculo EZEstimate

### 7.3.- Planificación temporal

La planificación de tarea la primera tarea que se realizó después de la estimación de costes y esfuerzo, con ella se identifican cada una de las tareas que habrá que realizar para conseguir el objetivo del desarrollo del proyecto.

Para obtener más información sobre la planificación temporal como por ejemplo el calendario que se estableció al principio del proyecto, se puede acceder al documento *Anexo I - Planificación temporal*.

En la siguiente captura se puede observar una captura de pantalla de la planificación realizada sobre un proyecto en Microsoft Project.

Nombre de tarea	Duración	Comienzo	Fin	Predi	Nombres de los recursos
<b>▲ Proyecto</b>	<b>131 días</b>	<b>jue 01/07/21</b>	<b>jue 01/09/22</b>		
<b>▲ Inicio</b>	<b>21 días</b>	<b>jue 01/07/21</b>	<b>lun 09/08/21</b>		
<b>▲ Iteración 1</b>	<b>10 días</b>	<b>jue 01/07/21</b>	<b>lun 19/07/21</b>		
<b>▲ Modelado del negocio</b>	<b>3 días</b>	<b>jue 01/07/21</b>	<b>mar 06/07/21</b>		
Planteamiento del problema	3 días	jue 01/07/21	mar 06/07/21		Sergio Gómez Sánchez
Búsqueda de necesidades	3 días	jue 01/07/21	mar 06/07/21		Sergio Gómez Sánchez
<b>▲ Requisitos</b>	<b>3 días</b>	<b>mar 06/07/21</b>	<b>lun 12/07/21</b>		
Fijar Objetivos	3 días	mar 06/07/21	lun 12/07/21	6	Sergio Gómez Sánchez
<b>▲ Análisis</b>	<b>2 días</b>	<b>lun 12/07/21</b>	<b>jue 15/07/21</b>		
Análisis de tecnologías disponibles	2 días	lun 12/07/21	jue 15/07/21	8	Sergio Gómez Sánchez
<b>▲ Diseño</b>	<b>3 días</b>	<b>lun 12/07/21</b>	<b>sáb 17/07/21</b>		
Planificación diseño	3 días	lun 12/07/21	sáb 17/07/21	8	Sergio Gómez Sánchez
<b>▲ Implementación</b>	<b>2 días</b>	<b>jue 15/07/21</b>	<b>lun 19/07/21</b>		
Definición del lenguaje de programación	2 días	jue 15/07/21	lun 19/07/21	10	Sergio Gómez Sánchez
Fin Iteración 1	0 días	lun 19/07/21	lun 19/07/21	4;7;9;11	
<b>▲ Iteración 2</b>	<b>11 días</b>	<b>mar 20/07/21</b>	<b>lun 09/08/21</b>	<b>15</b>	
<b>▲ Modelado del negocio</b>	<b>3 días</b>	<b>mar 20/07/21</b>	<b>sáb 24/07/21</b>		
Investigación de servicios similares	1 día	mar 20/07/21	mié 21/07/21		Sergio Gómez Sánchez
Definición de posibles actores	2 días	mié 21/07/21	sáb 24/07/21	18	Sergio Gómez Sánchez
<b>▲ Requisitos</b>	<b>7 días</b>	<b>sáb 24/07/21</b>	<b>vie 06/08/21</b>		
Identificación de escenarios	2 días	sáb 24/07/21	mié 28/07/21	19	Sergio Gómez Sánchez
Identificación de casos de uso	5 días	jue 29/07/21	vie 06/08/21	21	Sergio Gómez Sánchez
<b>▲ Análisis</b>	<b>1 día</b>	<b>mar 20/07/21</b>	<b>mié 21/07/21</b>		
Identificación de paquetes	1 día	mar 20/07/21	mié 21/07/21		Sergio Gómez Sánchez
<b>▲ Diseño</b>	<b>1 día</b>	<b>sáb 07/08/21</b>	<b>lun 09/08/21</b>		
Búsqueda de patrones arquitectónicos	1 día	sáb 07/08/21	lun 09/08/21	24;22	Sergio Gómez Sánchez
Fin Iteración 1	0 días	lun 09/08/21	lun 09/08/21	17;23;2	
Fin Inicio	0 días	lun 09/08/21	lun 09/08/21	27	

Ilustración 17.- Planificación Imagen 2

## Memoria del proyecto

La herramienta Microsoft Project permite realizar automáticamente un diagrama de Gantt resultante de la planificación que se genera en la herramienta Microsoft Project una vez finalizada la planificación de tareas.

En la siguiente captura se puede ver el diagrama de Gantt generado para las tareas de las iteraciones y fases mostrada en la captura anterior *Ilustración 16.- Planificación Imagen 2*.

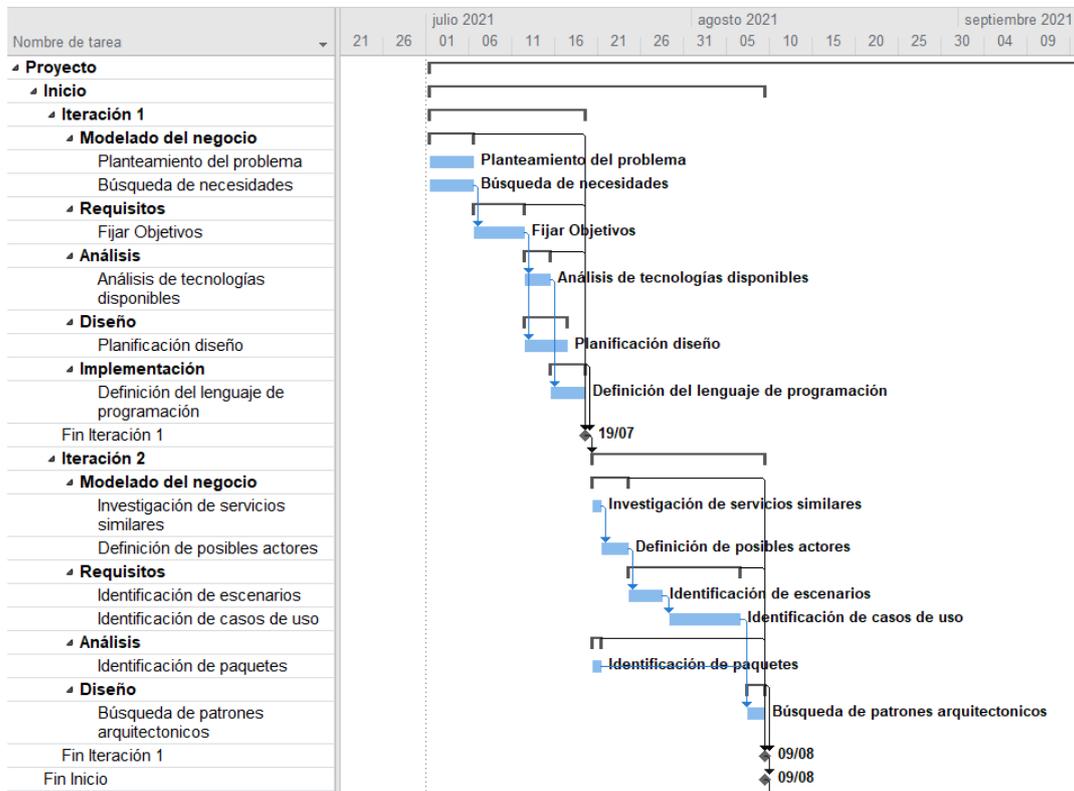


Ilustración 18.- Planificación Imagen 1

## 7.4.- Especificación de requisitos

En esta fase se han identificado y especificado todos los requisitos del sistema que se pretenden que alcance el sistema en su implementación, realizando un catálogo de requisitos reuniendo los objetivos del sistema, actores, requisitos de información, requisitos de restricción de información, requisitos no funcionales, casos de uso como requisitos funcionales y la matriz de rastreabilidad.

Además, se establecen todos los participantes del proyecto.

Toda esta información vendrá documentada en el *Anexo II – Especificación de requisitos del sistema* que acompaña a esta memoria.

A continuación, se expondrá algunos ejemplos de cada una de las etapas de la fase de especificación de requisitos.

### 7.4.1.- Participantes

Los participantes que se han identificado son de 3 tipos:

- **Autor:** Solo hay uno, y es el creador del proyecto y el principal desarrollador del proyecto. Tendrá la siguiente tabla.
- **Tutor:** Hay 3 tutores en total dentro de los participantes. Estos aportarán su experiencia y su punto de vista, aconsejando a lo largo del proyecto al autor de este proyecto. Tendrán la siguiente tabla:
- **Cliente:** Hay un solo cliente y es el ya mencionado en otros apartados de esta memoria. Forma parte de la organización que se encarga de la explotación de ganado. Aportará conocimiento desde el punto de vista del cliente, es decir, desde la necesidad, ya que fue la persona que deseaba mejorar su sistema de gestión y que provocó el inicio a este proyecto. Tendrá la siguiente tabla:

### 7.4.2.- Catálogo de requisitos:

#### 7.4.2.1.- Objetivos del sistema

Se describen los objetivos del sistema que debe cumplir. Los principales objetivos que se han definido son:

- Gestión de Usuarios
- Gestión de Organizaciones
- Gestión de Fincas
- Gestión de búsqueda de fincas
- Gestión de Animales
- Gestión de búsqueda de animales
- Gestión de Usuarios
- Gestión de Alimentos
- Gestión de Medicamentos
- Almacenamiento y visualización de históricos de gestión
- Almacenamiento y visualización de estadísticas

## Memoria del proyecto

En la siguiente tabla se muestra una descripción de un paquete de objetivos del sistema, concretamente el de Gestión de Usuarios:

OBJ-0001	Gestión de Usuarios
<b>Versión</b>	1.0 - (24/08/1998)
<b>Autores</b>	Sergio Gómez Sánchez
<b>Descripción</b>	El sistema deberá permitir registrar usuarios, así como la modificación de sus datos una vez registrados. También debe controlar la autenticación de los usuarios ya registrados cuando deseen acceder a través del inicio de sesión.. Además, debe gestionar todos los permisos de acceso a la información de las distintas organizaciones que los Administradores de estas concedan a los usuarios Trabajadores.
<b>Subobjetivos</b>	Ninguno
<b>Importancia</b>	Vital
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 5.- Gestión de Usuarios

Para conocer en profundidad el resto de los objetivos consultar el *Anexo II – Especificación de requisitos del sistema* que acompaña a esta memoria.

### 7.4.2.1.- Actores

Después de definir los objetivos del sistema se establecieron los distintos actores que se podrán encontrar en el sistema.

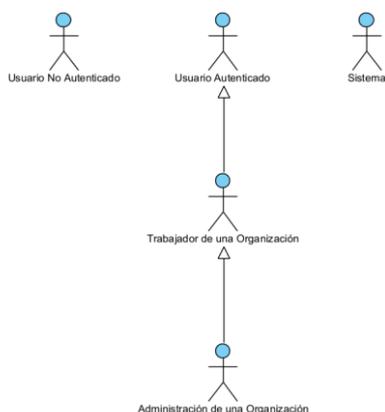


Ilustración 19.- Diagrama de Actores del Sistema

Aquí podemos observar una tabla de ejemplo de uno de los actores.

ACT-0003	Trabajador de una Organización
<b>Versión</b>	1.0 - (24/08/1998)
<b>Autores</b>	Sergio Gómez Sánchez
<b>Descripción</b>	Este actor representa a cada uno de los usuarios registrado e iniciado sesión en el sistema y que pertenecen a una de las organizaciones registradas en el sistema.
<b>Comentarios</b>	Ninguno

Tabla 6.- Actor Trabajador de una Organización

## Memoria del proyecto

Para conocer en profundidad el resto de los actores consultar el *Anexo II – Especificación de requisitos del sistema* que acompaña a esta memoria.

### 7.4.2.1.- Requisitos de información

Los primeros requisitos en definirse fueron los de información, estos requisitos reflejarán los datos almacenados en el sistema.

Los requisitos de información son:

- Datos de usuarios registrados
- Fincas registradas
- Animales registrados
- Organizaciones registradas
- Alimentos y medicamentos
- Transacciones de animales realizadas

Aquí podemos observar una tabla de ejemplo de uno de los requisitos de información.

IRQ-0003	Animales registrados
<b>Versión</b>	1.0 - (24/08/1998)
<b>Autores</b>	Sergio Gómez Sánchez
<b>Dependencias</b>	
<b>Descripción</b>	El sistema deberá almacenar la información correspondiente a los datos de los animales registrados en el sistema. En concreto:
<b>Datos específicos</b>	ID idAnimal Imagen de animal Fecha de Nacimiento Sexo Raza Peso Finca de Nacimiento Finca Actual País de Nacimiento País Engorde idMadre idPadre
<b>Importancia</b>	Vital
<b>Estado</b>	Validado
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 7.- Animales Registrados

Para conocer en profundidad el resto de los requisitos de información consultar el *Anexo II – Especificación de requisitos del sistema* que acompaña a esta memoria.

## Memoria del proyecto

### 7.4.2.1.- Requisitos de restricción de información

Otros de los primeros requisitos en definirse fueron los de restricción de información, estos requisitos describen las restricciones de información que deberán tener los datos almacenados por el sistema.

Los requisitos de información son:

- Usuarios registrados
- Administradores de una organización
- Usuarios pertenecientes a una organización
- Permisos del administrador a trabajadores

Para conocer en profundidad el resto de los requisitos de restricción de información y poder consultar las tablas de descripción de cada uno de ellos, consultar el *Anexo II – Especificación de requisitos del sistema* que acompaña a esta memoria.

### 7.4.2.1.- Requisitos no funcionales

Los requisitos no funcionales se definen para imponer alguna restricción de diseño o estándares de calidad. Son requisitos que describen características de funcionamiento, es decir, no describen funciones o información a guardar como en los requisitos de información.

Los requisitos no funcionales son:

- Usabilidad
- Seguridad de los datos
- Eficiencia
- Portabilidad o Multiplataforma
- Diseño “responsive”

### 7.4.2.1.- Requisitos funcionales

Los requisitos funcionales describen las funciones o actividades que debe realizar el sistema cuando se cumplen ciertas condiciones, son los requisitos que se corresponden con la funcionalidad del sistema.

Lo primero que se ha realizado ha sido un diagrama de paquetes para tener una división inicial del sistema.

## Memoria del proyecto

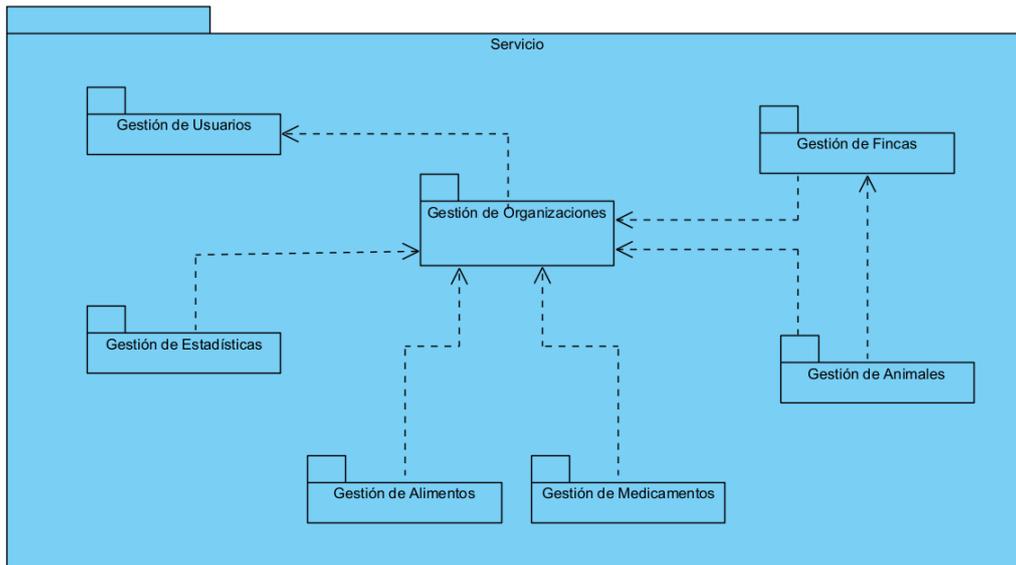


Ilustración 20.- Diagrama de Paquetes del Sistema

Posteriormente, se identificaron y explicaron cada uno de los casos de uso recogidos en el sistema dividiéndolos en cada uno de estos paquetes.

La siguiente ilustración se corresponde con el diagrama de casos de uso para el paquete Gestión de Usuarios y servirá como ejemplo:

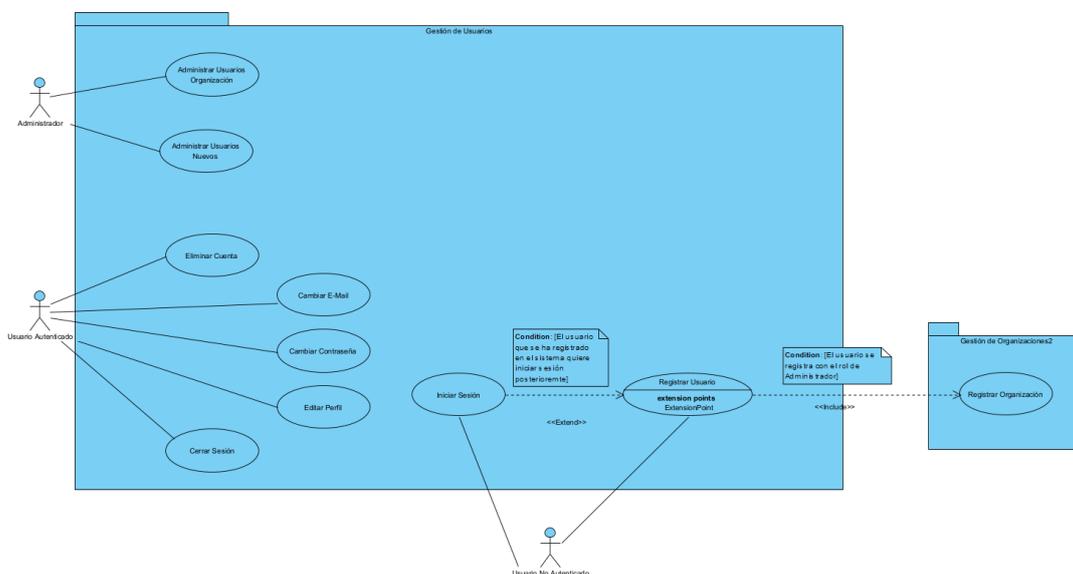


Ilustración 21.- Diagrama de Casos de Uso de Gestión de Usuarios

En el anterior diagrama podemos ver como interactúan los diferentes actores con los casos de uso identificados

La siguiente ilustración se corresponde con la definición y especificación del caso de uso *Añadir Animal* para el paquete Gestión de Usuarios y servirá como ejemplo:

UC-0020		Buscar Animal	
<b>Versión</b>	1.0 - (24/08/1998)		
<b>Autores</b>	Sergio Gómez Sánchez		
<b>Dependencias</b>			
<b>Descripción</b>	El sistema deberá comportarse tal y como se describe en el siguiente en este caso de uso cuando: Un usuario autenticado desee buscar un Animal de la organización a la que pertenece.		
<b>Precondición</b>	El usuario autenticado debe ser trabajador y por tanto tener los permisos por parte del administrador para ver la información de la organización.		
<b>Secuencia normal</b>	<b>Paso</b>	<b>Acción</b>	
	1	Un <b>Usuario Trabajador ACT-0003</b> introduce el idAnimal del Animal que desea buscar.	
	2	El sistema desea conocer la información de los animales de la organización registrados en el sistema.	
	3	El <b>Sistema ACT-0005</b> busca el Animal deseado.	
	6	El <b>Sistema ACT-0005</b> finaliza el caso de uso.	
<b>Postcondición</b>	El Animal será buscado y si existe comenzará el caso de uso <b>Consultar Animal</b> .		
<b>Excepciones</b>	<b>Paso</b>	<b>Acción</b>	
	1	Si el Animal buscado no existe, no se mostrará y el <b>Sistema ACT-0005</b> mostrará un mensaje notificándolo.	
<b>Importancia</b>	Vital		
<b>Urgencia</b>	Inmediatamente		
<b>Estado</b>	Validado		
<b>Estabilidad</b>	Alta		
<b>Comentarios</b>	Ninguno		
<b>Estabilidad</b>	Alta		
<b>Comentarios</b>	Ninguno		

Tabla 8.- UC-0020 Buscar Animal

#### 7.4.2.1.- Matriz de rastreabilidad

Las matrices de rastreabilidad nos permiten conocer relaciones entre diferentes aspectos definidos de nuestro proyecto como pueden ser, requisitos, objetivos, casos de uso...

Concretamente se han realizado dos tipos de matrices:

- **Matriz de requisitos:** Nos permite conocer la relación entre los casos de uso y los requisitos no funcionales y los requisitos de información.
- **Matriz de objetivos:** Nos permite conocer la relación entre los casos de uso y los objetivos del sistema definidos.

## 7.5.- Análisis de requisitos

En esta fase del desarrollo del proyecto se ha llevado a cabo un análisis exhaustivo de los requisitos definidos en la fase anterior. Para poder realizar el análisis se han expuesto varios puntos a desarrollar.

Para un mayor detalle de la fase de análisis de requisitos consultar el *Anexo III – Análisis de requisitos* que acompaña a esta memoria.

### 7.5.1.- Modelo de dominio

Un modelo de dominio es una representación de las clases del mundo real, no de componentes software. El modelo de dominio recoge las necesidades de almacenamiento y gestión de la información del sistema. (Holgado, 2022)

Por tanto, para realizar este modelo de dominio se han determinados las entidades del modelo de negocios que sea necesario gestionar, sus atributos, y las relaciones entre ellas.

En el siguiente diagrama podremos ver la representación del modelo de dominio.

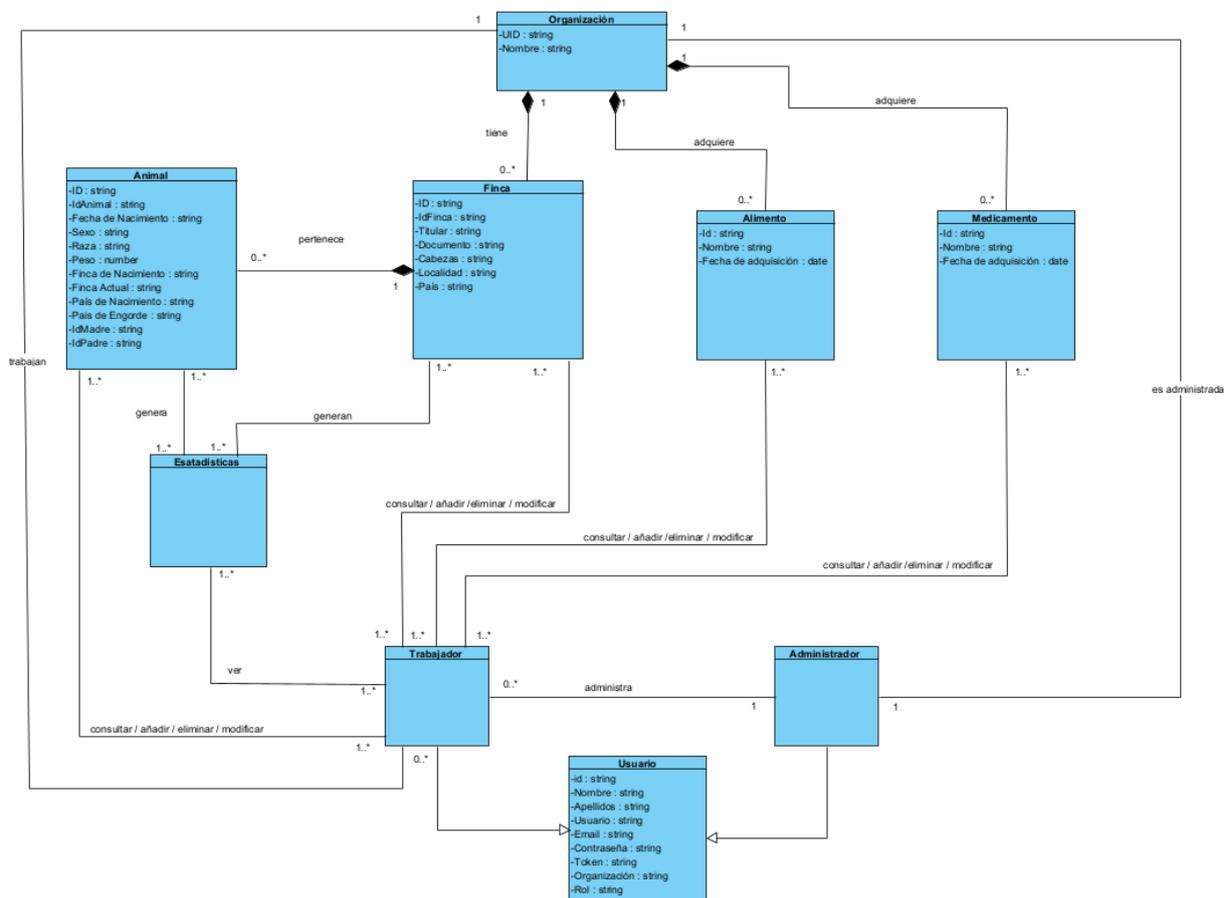


Ilustración 22.- Modelo de Dominio

## Memoria del proyecto

### 7.5.2.- Vista de Interacción de los Casos de Uso

En esta fase también se ha realizado el análisis de los casos de uso, explicando uno por uno los pasos que deberían seguir en forma de secuencia, para ello se han utilizado diagramas de secuencia realizados a través de la herramienta Visual Paradigm.

En la siguiente ilustración podremos observar el diagrama de secuencia del caso de uso “Añadir Finca” como ejemplo:

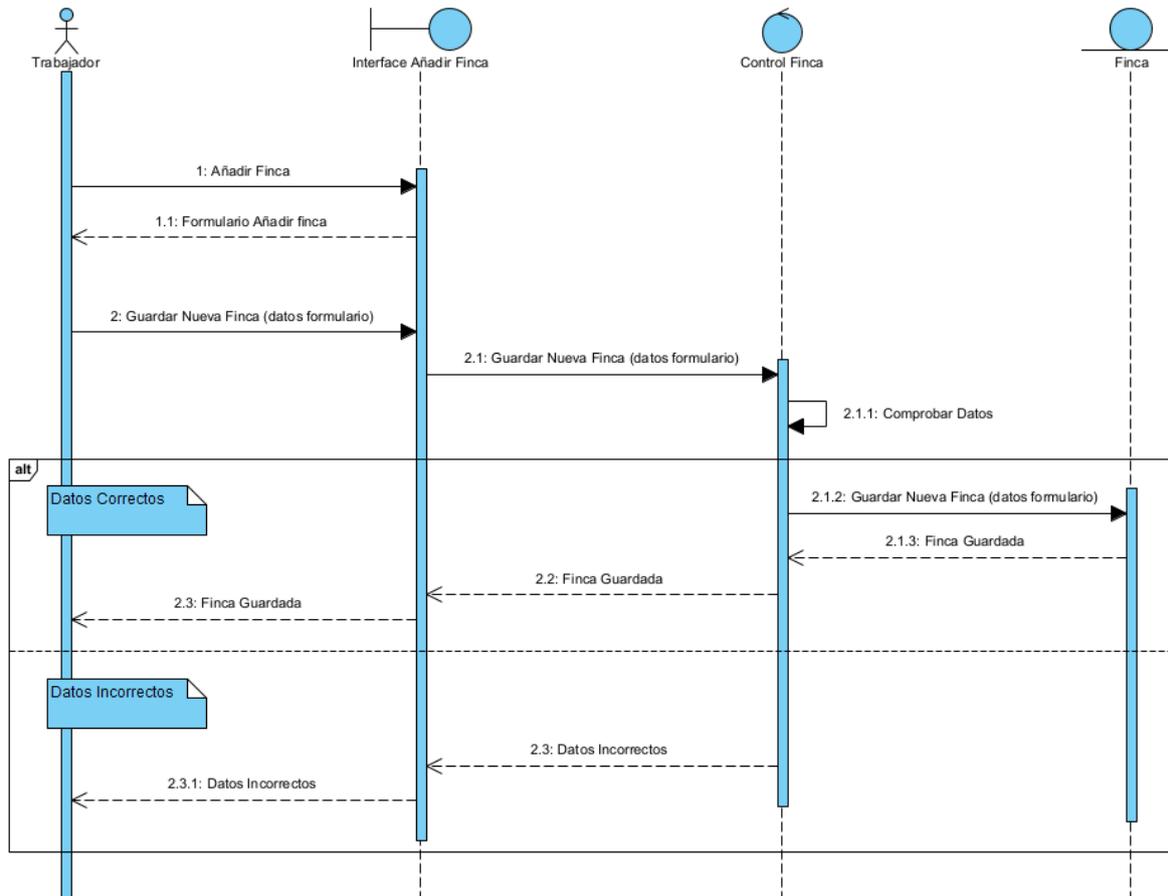


Ilustración 23.- Diagrama de Secuencia Caso de Uso Añadir Finca

Para conocer el resto de los diagramas de secuencia de los casos de uso consultar el *Anexo III – Análisis de requisitos* que acompaña a esta memoria.

### 7.5.3.- Clases de Análisis

En este punto de la fase de análisis se han creado diagrama de comunicación por cada uno de los paquetes que tiene el sistema para representar de una forma gráfica como se comunican y distribuyen las clases de análisis de los diagramas del punto anterior *Vista de Interacción de los Casos de Uso*.

La siguiente ilustración representa el diagrama de comunicación del paquete Gestión de Usuarios que servirá como ejemplo:

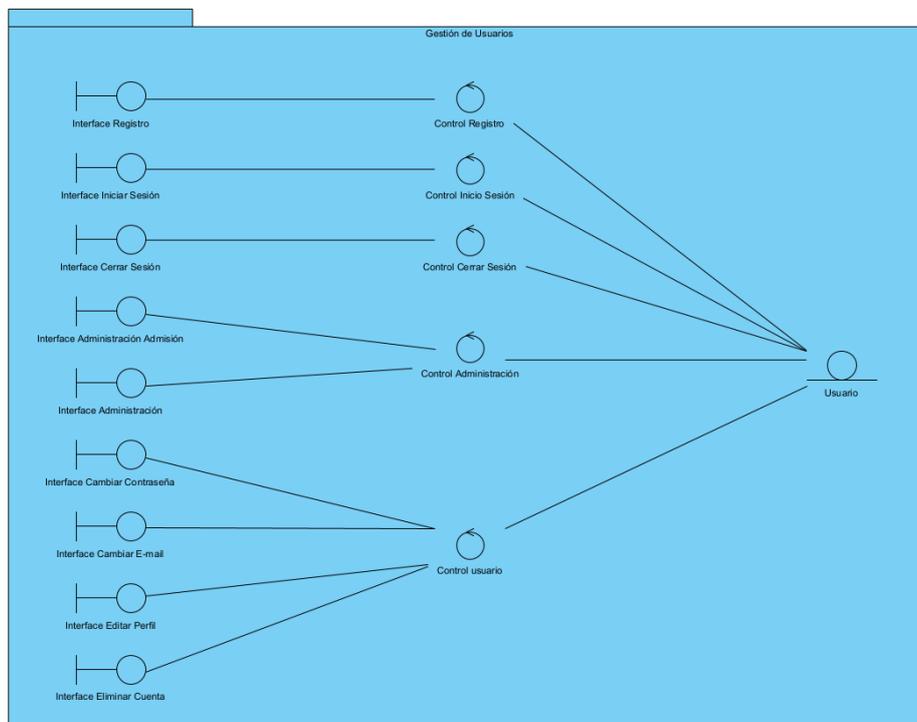


Ilustración 24.- Clases de Análisis. Gestión de Usuarios

Para conocer el resto de los diagramas de comunicación de los paquetes consultar el *Anexo III – Análisis de requisitos* que acompaña a esta memoria.

### 7.5.4.- Vista de la Arquitectura del Modelo de Análisis

En ese punto del análisis se realizó la primera versión de la vista arquitectónica del proyecto. Esta servirá de aproximación para la arquitectura que se obtendrá en la fase de diseño.

Para conocer más detalles sobre la vista arquitectónica consultar el *Anexo III – Análisis de requisitos* que acompaña a esta memoria.

## 7.6.- Diseño del sistema software

En esta fase se lleva a cabo la la fase del diseño del sistema software. Esta fase es la fase más cercana a la implementación, es más cercana que todas las anteriores.

En esta fase se realizar una traducción de los requisitos en una representación del software que construirá la base de la implementación que se realizará en el siguiente punto.

Se puede encontrar la información más detallada de la fase de diseño del sistema, como el modelo de diseño en el *Anexo IV - Diseño del Sistema Software*.

### 7.6.1.- Patrón Arquitectónico

El patrón arquitectónico usado para el desarrollo de la aplicación web es el Modelo-Vista-Controlador (MVC), ya que es el patrón que sigue el framework de Angular que es el framework que se decidió usar para desarrollar este servicio.

Este patrón se divide en tres partes, que son las que le dan su nombre: modelo, vista y controlador.

- **Modelo:** El modelo define qué datos debe contener la aplicación, su lógica de negocio y sus mecanismos de persistencia.
- **Vista:** Define como se deben mostrar los datos de la aplicación y la información que se envía al cliente y los mecanismos de interacción.
- **Controlador:** El controlador contiene la lógica que actualizar el modelo y la vista gestionando el flujo de información entre ellos junto a los datos para adaptarlos a las necesidades de cada uno.

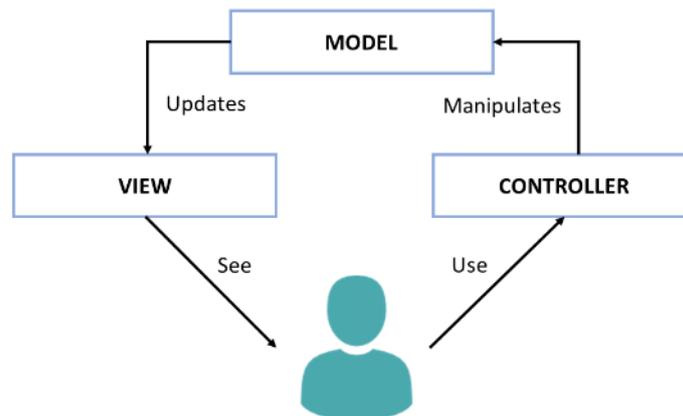


Ilustración 25.- Modelo-Vista-Controlador MVC

Para conocer más acerca de este patrón arquitectónico, sus ventajas o como Angular hace uso de este patrón consultar el *Anexo IV - Diseño del Sistema Software*.

## Memoria del proyecto

### 7.6.2.- Subsistemas de diseño

En este punto se ha realizado un reparto del sistema en subsistemas específicos y la relación entre ellos. La siguiente ilustración muestra el diagrama de subsistemas de diseño que representa de forma gráfica estos subsistemas.

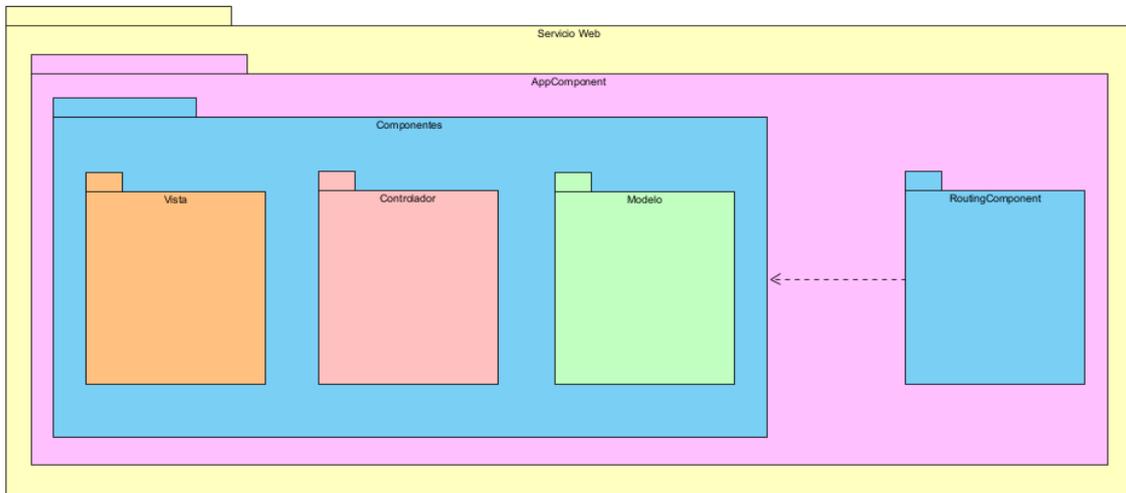


Ilustración 26.- Subsistemas de Diseño

Para conocer más acerca de los subsistemas de diseño, sus características y demás, consultar el *Anexo IV - Diseño del Sistema Software*.

### 7.6.2.- Vista Arquitectónica del Modelo de Diseño

En este punto se ha realizado una distinción del patrón MVC adaptado a las distintas clases de diseño.

Se ha realizado un diagrama de secuencia por cada uno de los casos de uso reflejando las relaciones y comunicación de los distintos componentes del sistema.

En estos diagramas podremos encontrar los tres puntos del MVC.

Un ejemplo de diagrama de secuencia para el caso de uso *UC-0027 - Modificar Alimento*.

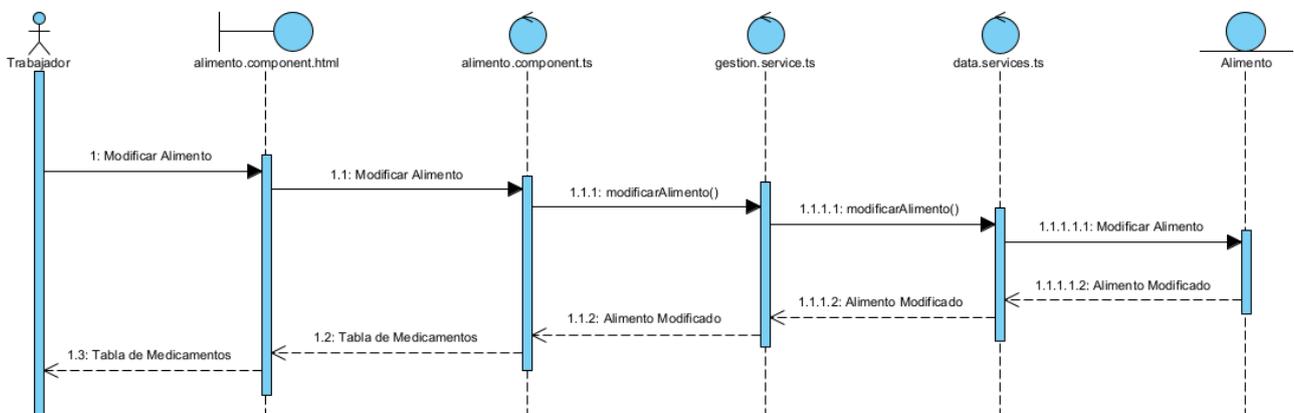


Ilustración 27.- Diagrama Secuencia: UC-0028 Modificar Alimento

## Memoria del proyecto

### 7.6.2.- Diseño de la Base de Datos - Firebase

En este apartado se ha realizado una descripción del diseño de la base de datos utilizada, en este caso es la RealTime Database de Firebase.

Los datos en esta base de datos se almacenan en formato JSON y es capaz de sincronizar en tiempo real con cada cliente conectado.

### 7.6.3.- Modelo de Despliegue

Durante esta fase se realizó este modelo que trata de describir la implementación física de la información generada por el programa software, muestra todos los nodos y componentes que forman el sistema y los canales a través de los que interactúan.

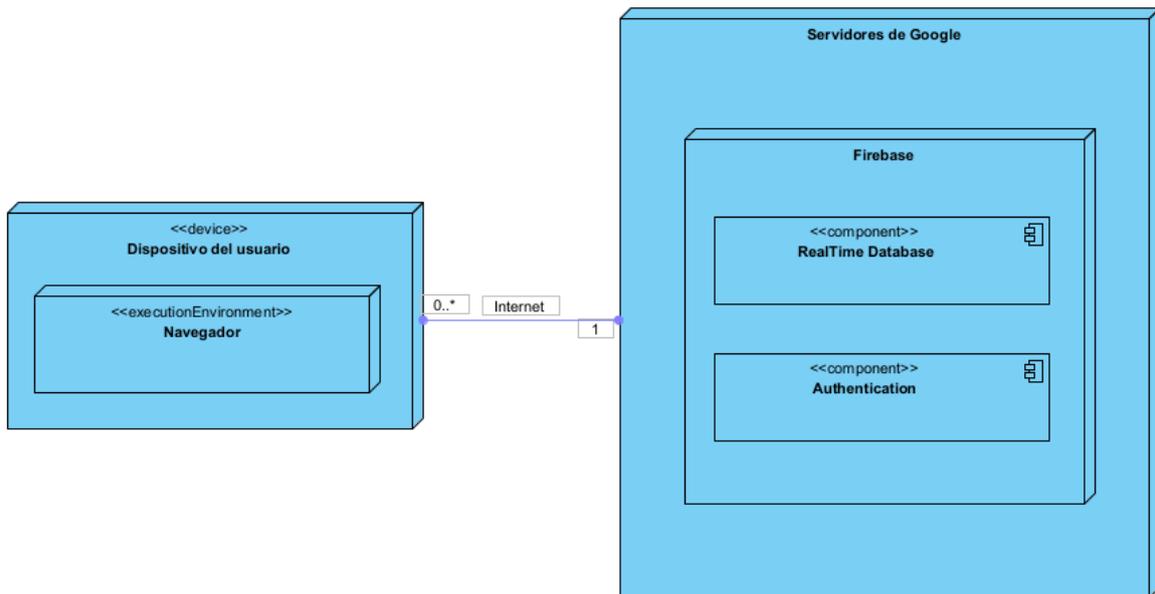


Ilustración 28.- Modelo de Despliegue

El sistema accederá a la API de Google Firebase a través de internet para poder tener una base de datos en línea.

## 7.7.- Implementación

La fase de implementación es una de las últimas fases del desarrollo, es la fase donde se ha llevado a cabo la codificación del sistema, esta fase ha sido la más larga en cuanto a duración. Es la fase en la que se han puesto en prácticas las técnicas y herramientas explicadas en el apartado *Técnicas y Herramientas* de esta memoria

Para la realización de la implementación se deben tener en cuenta los diseños y resultados de la fase de diseño del sistema software. Esto ha facilitado el proceso, consiguiendo como resultado un proyecto ordenado.

En este apartado se comentarán los aspectos a destacar de la fase de implementación.

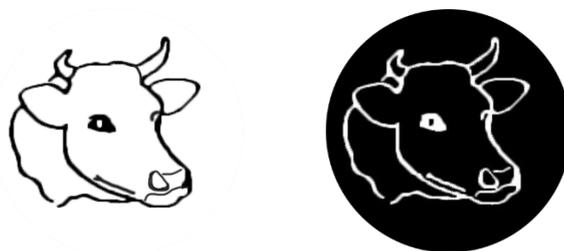
### 7.7.1.- Interfaz de usuario: diseño “responsive” y estructura general

El primer apartado de la implementación que se realizó fue la implementación del frontend de la aplicación web, es decir, la parte visual, las vistas, las interfaces.

Para ellos se fueron creado los distintos componentes Angular, en los que cada uno tenía diferentes vistas y se fue escribiendo el código de la parte de los ficheros html y css.

El logo de la aplicación fue uno de los primeros diseños, fue realizado a mano con Adobe Photoshop y se trata de la imagen en blanco y negro de una vaca.

Se realizaron dos versiones para poder usar en diferentes contextos de diseño de la aplicación, uno para usarlo en fondo negros y otro para usarlo en fondo blancos.



Aunque no se llegue a apreciar la foto de la izquierda está sobre un fondo circular blanco.

Los colores de la aplicación web no son aleatorios, fueron escogidos con intención, quería utilizarse colores verdes que recordasen al campo donde se encuentran las ganaderías, pero a la vez quería darse un toque moderno, por lo que se eligió una paleta de colores específica con la herramienta web Paletton, la paleta de colores elegida fue la siguiente:



Para la realización de las interfaces se tuvieron en cuenta estos colores, no se salen de la línea excepto para ciertos botones que comentaremos más adelante.

El diseño de las interfaces fue realizado en Adobe XD, aquí se muestran algunas capturas del boceto inicial realizado:

The landing page features a dark green header with navigation links: Inicio, Gestión, Registro, and Opción 4. A search bar and buttons for 'Iniciar Sesión' and 'Registrarse' are also present. The main content area has a light green background with a large title: 'Conectando el mundo de la ganadería con la tecnología'. Below the title are buttons for 'Inicia Sesión' and 'Registrarse'. A central image shows a person with a smartphone, connected by a green line to three circular icons: a cow, a computer monitor with a bar chart, and a smartphone with a signal icon. Below these icons are three columns of text describing the platform's features: 'Gestión de tu ganadería', 'Control de compras y ventas', and 'Siempre en contacto con tu ganado'. A link 'Descubre el funcionamiento de NOMBRE' is located at the bottom of the main content area.

Inicio Gestión Registro Opción 4

Iniciar Sesión Registrarse

# Conectando el mundo de la ganadería con la tecnología

Inicia Sesión  
o  
Registrarse

**Gestión de tu ganadería**  
Controla y gestiona de forma sencilla las altas como las bajas de tu ganadería

**Control de compras y ventas**  
Ten el control de tus compras y tus ventas en todo momento.

**Siempre en contacto con tu ganado**  
Obtén notificaciones a tiempo real de todo lo que ocurren en tus fincas.

[Descubre el funcionamiento de NOMBRE](#)

The login form is titled 'Iniciar sesión en GESTIGAN'. It contains two input fields: 'Nombre de usuario' and 'Contraseña'. Below the password field is a link for '¿Olvidaste la contraseña?'. A green 'Iniciar Sesión' button is positioned below the fields. At the bottom of the form, there is a link '¿Nuevo en NOMBRE?' and a 'Registrarse' button. The footer contains links for 'Términos', 'Privacidad', 'Seguridad', and 'Contacto'.

Iniciar sesión en **GESTIGAN**

Nombre de usuario

Contraseña

[¿Olvidaste la contraseña?](#)

Iniciar Sesión

[¿Nuevo en NOMBRE?](#) [Registrarse](#)

[Términos](#) [Privacidad](#) [Seguridad](#) [Contacto](#)

Inicio Gestión Registro Opción 4

Fecha	Motivo	Procedencia/Destino	Documento	Cantidad	Balance
24/04/2021	NACIMIENTO	IDENTIFICADOR FINCA	Número de Documento	15	386
28/04/2021	VENTA	IDENTIFICADOR FINCA	Número de Documento	10	376

Eliminar Modificar Añadir

Inicio Gestión Registro Opción 4

# REGISTRO POR FINCAS

Identificador Identificador Identificador Identificador Identificador

Inicio   Gestión   Registro   Opción 4

# REGISTRO POR FINCAS



Identificador

Lugar donde se encuentra  
Amatos de Alba

Cantidad de animales  
376

Lugar donde se encuentra  
Amatos de Alba



376 animales

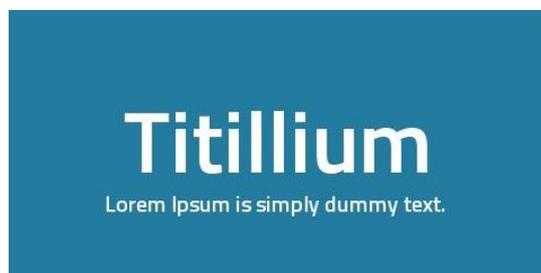
Fecha de acceso ▼

Fecha	Identificador	Fecha de Nacimiento	Sexo	Raza
24/04/2021	Identificador del animal	03/08/2013	Macho	Limusine



Como se puede comprobar el diseño estaba aún sin acabar, pero era el primer boceto desarrollado.

Contaba con los colores que se mencionaron anteriormente en la paleta de colores. Además, se decidió usar una tipografía general para todo el proyecto, la elegida fue Titillium.



Esta tipografía pertenece al estilo sans serif, por lo que es bastante agradable a la vista. Su versatilidad le permite tener múltiples usos; desde un fuerte encabezado hasta un largo texto sin cansar la vista.

Es recomendable para diseño web ya que tiene un toque moderno y es muy legible. Además, es ideal para sitios web como Gestigan que requieren más uso de texto que de imágenes.

A partir de estos bocetos y estas elecciones de tipografía se realizó la implementación, obviamente, estas ideas fueron variando entre otras cosas, por las limitaciones que ofrece el diseño web o incluso, por lo contrario, las oportunidades que ofrecen más allá del diseño.

Para poder crear diseño responsive se consideró utilizar la librería de Bootstrap debió a su aportación y facilidad a la hora de construir diseños web sencillos reutilizables y que cumplan con la responsividad requerida.

Además, se ha tenido en cuenta las distintas resoluciones de Bootstrap a la hora de diseñar el contenido, modificándose en función de cada resolución.

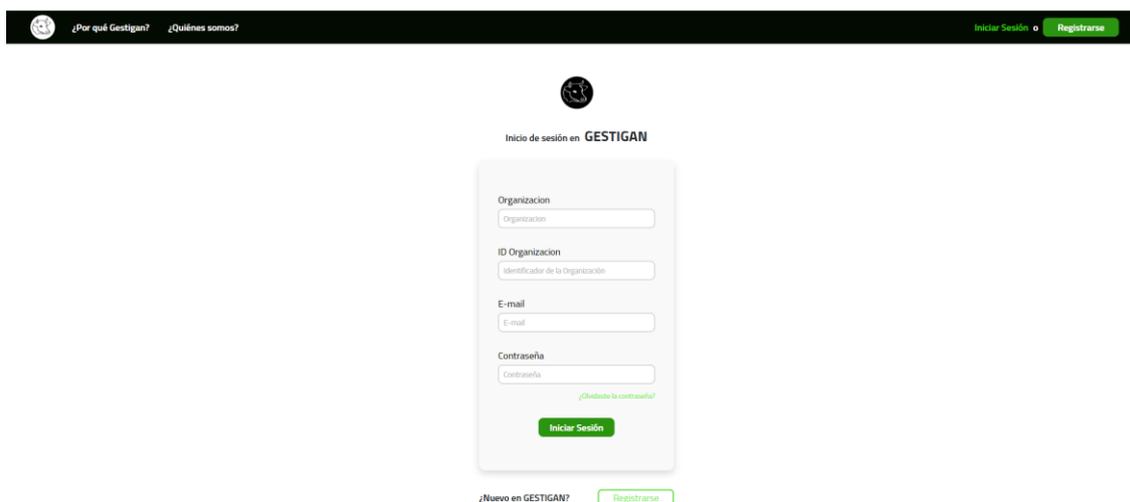
## Memoria del proyecto

A la hora de crear la maquetación de las interfaces se ha tenido en cuenta la patrón de diseño MobileFirst. Mobile First es un concepto de diseño web optimizado para móviles, donde inicialmente se crea una página web para tablets y smartphones, para adaptarla luego a las necesidades de un navegador de escritorio.

Por lo que se realiza un primer diseño para mobile, (mobile-first) y se va modificando según aumenta la resolución según se van pasando por puntos de ruptura. Los puntos de ruptura que se han utilizado son los siguientes:

- **360 px**
- **576 px**
- **768 px**
- **992 px**
- **1200 px**
- **1920 px**

Algunas imágenes de como quedo finalmente la interfaz del sistema se mostrarán a continuación:



### 7.7.2.- Gestión de Fincas

La implementación de la funcionalidad para la gestión de fincas fue una de las primeras tareas que se realizó. Se creó un modelo de Finca que tiene los siguientes atributos:

- **UID** de tipo string
- **Imagen** de tipo string
- **idFinca** de tipo string
- **Titular** de tipo string
- **Documento** de tipo string
- **Cabezas** de tipo string
- **Localidad** de tipo string

## Memoria del proyecto

- **País** de tipo string

Para poder manejar todos los datos de las fincas y poder mostrarlo en cada una de las interfaces que fuese necesario, especialmente en la interfaz de finca a la que se podrá acceder a través de la página inventario.

También se podrá acceder a la URL componiendo con el parámetro finca y el parámetro que corresponde con el identificador de finca.

```
path: "finca/:idFinca"
```

Además, se creó la funcionalidad necesaria para poder crear fincas, creando un componente específico para ello.

También se desarrollo la funcionalidad para eliminar o modificar fincas dentro de la propia página de la finca.

Para poder guardar los datos en las bases de datos o poder realizar una petición y recoger toda la información de las fincas se creó un servicio que se comunicará con el servicio que tiene comunicación directa con la base de datos que será explicado más adelante.

### 7.7.3.- Gestión de Animales

La implementación de la funcionalidad para la gestión de animales fue otra de las primeras tareas que se realizó. Se creó un modelo de Animal que tiene los siguientes atributos:

- **UID** de tipo string
- **Imagen** de tipo string
- **idAnimal** de tipo string
- **fechaNacimiento** de tipo date
- **Sexo** de tipo string
- **Raza** de tipo string
- **Peso** de tipo number
- **País de Engorde** de tipo string
- **País de Nacimiento** de tipo string
- **Id Finca Nacimiento** de tipo string
- **Id Finca Actual** de tipo string
- **Id madre** de tipo string
- **Id padre** de tipo string

Al igual que se implemento en la gestión de fincas se creó para poder manejar todos los datos de los animales y poder mostrarlo en cada una de las interfaces que fuese necesario, especialmente en la interfaz de animal a la que se podrá acceder a través de la página finca o buscando el crotal del animal en el buscador.

También se podrá acceder a la URL componiendo con el parámetro animal y el parámetro que corresponde con el crotal del animal.

```
path: 'animal/:idAnimal'
```

Se implementó toda la funcionalidad para poder registrar animales, eliminarlos o modificar sus datos. Los animales que formen parte de una de las fincas aparecerán en el módulo de finca, y se mostrarán en forma de tabla.

Estarán suscritos a un servicio que les permite recoger y modificar la información de la base de datos en el apartado de animales a través de distintos tipos de peticiones

#### 7.7.4.- Gestión de Usuarios y Organizaciones

Uno de los puntos más importantes para la seguridad de los datos, tanto porque cada uno de los usuarios deberá tener la seguridad de que sus datos son privados como porque las organizaciones podrán ser accedidas únicamente por los trabajadores que pertenezcan a ella, por lo que se creó una funcionalidad que permite a los usuarios no autenticados registrarse en el sistema como Administrador o como Trabajado

Si se registran como administrador crearán una organización en el momento del registro.

En cambio, si se registran como trabajadores, indicarán en el momento del registro el nombre de la organización a la que desean pertenecer y se realizará petición automática a los administradores de la organización indicándoles el deseo del trabajador de pertenecer a la organización y así poder tener los permisos de consultar y modificar la información de ella.

Los usuarios se almacenarán en la base de datos siguiendo el modelo de Usuario que se ha creado, que tendrá los siguientes atributos:

- **id** de tipo string
- **Nombre** de tipo string
- **Apellidos** de tipo string
- **Usuario** de tipo string
- **E-mail** de tipo string
- **Contraseña** de tipo string
- **Token** de tipo string
- **Organización** de tipo string
- **Rol** de tipo string

La funcionalidad del registro, el inicio de sesión o el perfil fueron implementados en distintos componentes, creando así 3 tipos de servicios que se encargarán diferentes funciones que realizarán consultar o modificaciones en la base de datos en el apartado de usuarios.

Las organizaciones como hemos dicho se crearán automáticamente en el registro del usuario con rol de administrador, a estas se les generará un identificador en el momento del registro para poder diferenciarlas cuando existan organizaciones con el mismo nombre.

Las organizaciones también tendrán su propio modelo:

- **UID** de tipo string
- **Nombre** de tipo string

Las organizaciones solo podrán ser eliminadas por los administradores de cada una en el apartado perfil del usuario.

## Memoria del proyecto

### 7.7.5.- Configuración de la base de datos

La base de datos es uno de los puntos más importantes, ya que la forma en la que se guardan los datos en la base de datos RealTime Database de Firebase es en formato JSON, es decir, se trata de una base de datos que no es SQL ni relacional.

Firebase aconseja evitar la anidación de datos a la hora de generar la estructura de datos. Firebase dice:

*“Firebase Realtime Database permite anidar datos en un máximo de 32 [...]. Sin embargo, cuando obtienes datos de una ubicación de la base de datos, también se recuperan todos los nodos secundarios. Además, cuando le otorgas a alguien acceso de lectura o escritura en un nodo de la base de datos, también le otorgas acceso a todos los datos que se encuentran en ese nodo. Por lo tanto, en la práctica, lo ideal es que la estructura de los datos sea lo más simple posible”*

Firebase recomienda compactar las estructuras de datos para evitar este tipo de problemas:

*“[...] si los datos se dividen en rutas de acceso independientes, [...] se pueden descargar de manera eficaz en llamadas separadas, según sea necesario.”*

Por lo tanto, la información se dividirá en diferentes apartados.

Uno de ellos serán las organizaciones donde se registran cada una de las organizaciones con sus datos siguiendo su modelo, permitiendo así que cuando se quiere comprobar datos de organizaciones en el momento de iniciar sesión solo compruebe la información de la organización, pero no sus datos internos como fincas, animales, trabajadores, transacciones...

Otro serán todos los usuarios registrados en el sistema con el rol de trabajador que desean acceder a una organización pero que el administrador aún no les ha dado el permiso.

Y el último apartado será el de los datos donde se guardará la información de cada una de las organizaciones: fincas, animales, usuarios, alimentos y medicamentos. Cada uno de estos será un apartado dentro de la anidación a la organización, por lo que los animales no están anidados a la fincas a las que pertenecen, si no que están por separado, dotando así a la estructura de datos de animales mayor independencia, cuando un animal cambie de finca solo habrá que cambiar el parámetro del animal idFinca Actual.

## Memoria del proyecto

Las tablas muestran como se guarda la información en la base de datos.

usuariosOUT	0	id
		usuario
		email
		...
	1	id
		usuario
		email
		...
	2	id
		usuario
		email
		...
	n	id
		usuario
		email
		...

organizaciones	0	uid
		nombre
	1	uid
		nombre
	2	uid
		nombre
	n	uid
		nombre

Memoria del proyecto

datos	organización-uid	fincas	0	idFinca
				titular
				país
				...
		1		
		2		
		n		
		animales	0	idAnimal
				sexo
				raza
				...
		1		
		2		
	n			
	alimentos	0	id	
			nombre ..	
	1			
	2			
	n			
	medicamentos	0	id	
			nombre ..	
	1			
2				
n				
usuariosIN	0	id		
		usuario		
		email		
1				
2				
n				
transacciones	0	id		
		fecha		
n				
organización-uid	fincas	0		
		1		
		n		
	animales	0		
		1		
		n		
	alimentos	0		
		1		
		2		
	n			
	medicamentos	0		
		1		
		n		
	usuariosIN	0		
		1		
n				
transacciones	0			
	1			
	n			

## Memoria del proyecto

### 7.7.6.- Históricos de transacciones

Cada vez que se realice altas o bajas de cada uno de los animales se registrarán como una transacción y se registrarán la base de datos formando así un nuevo campo dentro de las organizaciones.

Las transacciones tendrán un modelo también:

- **idTransaccion** de tipo string
- **Fecha** de tipo string
- **Motivo** de tipo string
- **Procedencia** de tipo string
- **Destino** de tipo string
- **Cantidad** de tipo number
- **Balance** de tipo number

### 7.7.7.- Estadísticas

Se ha implementado un apartado dentro del sistema que te genera gráficos con las herramientas ya comentada ng2-charts y Charts.JS en función de datos correspondiente a las fincas y animales que están registrado en el sistema, como por ejemplo comparación de machos y hembras, comparación por razas...

## 7.8.- Pruebas

Durante el proceso de implementación se iban realizando también pruebas sobre la misma implementación.

Cada vez que se implementaba una funcionalidad en la aplicación se realizaban pruebas para comprobar que funcionaba correctamente y no surgía ningún error o problema, además de comprobar con las pruebas unitarias que el sistema reaccionase como se esperaba. Sobre todo, a la hora de realizar la funcionalidad de la gestión de usuarios, gestión de fincas y gestión de animales, ya que se trata de algunos de los puntos más importantes del sistema. Otro de los puntos vitales es la configuración de la base de datos, ésta paso por distintas fases, donde se ha ido transformando hasta acabar de la manera que ha sido comentada.

Además, al finalizar la implementación se han realizado pruebas para comprobar el flujo de funcionalidad de un usuario normal desde que se registra hasta que eliminar la cuenta pasando por cada una de las posibles funciones que pueden realizarse en él, algunas de estas pruebas han reportado fallos, por lo cual han provocado tener que realizarse mejoras para terminar de perfeccionar la funcionalidad.

También se han realizado prueba globales del sistema, esto permite saber si el sistema se comporta correctamente una vez integrados y trabajando juntos todos los componentes.

Con las pruebas al fin y al cabo se consigue tener un sistema estable y eficiente.

## 7.9.- Funcionalidad del sistema

En este apartado se va muestra la funcionalidad final del sistema, de forma que sirva de manual o explicación para el usuario, para que conozca la forma de usar la aplicación web y que sea comprensible en su uso.

### 7.9.1.- Inicio

Es la página principal para todos los usuarios no autenticados que acceden a la aplicación web. En esta primera página se exponen las tareas que se pueden realizar una vez registrados en el sistema.

En esta primera vista, existen diferentes botones, dos en el centro de la vista y dos ubicados en la parte derecha de la Navbar de navegación (barra de navegación) que nos permiten acceder al registro de usuario o al formulario de iniciar sesión, para que el usuario con una cuenta ya registrada y creada pueda acceder al sistema.



Ilustración 29.- Captura de Página Inicio



Ilustración 30.- Diseño Mobile Inicio

## Memoria del proyecto

### 7.9.2.- Registro e Inicio de Sesión

En estas dos vistas se mostrará un formulario para poder el registro o la autenticación en el sistema. En la zona central de la pantalla aparecerá este formulario con un conjunto de pares párrafo e input, donde los párrafos nos indican que información deberemos rellenar en cada uno de los input que se muestran.

The screenshot shows the registration page for the GESTIGAN system. At the top, there is a dark navigation bar with a logo on the left, links for '¿Por qué Gestigan?' and '¿Quiénes somos?', and buttons for 'Iniciar Sesión' and 'Registrarse'. The main content area features a central white card with the title 'Registro en GESTIGAN'. The card contains several input fields: 'Organización a la que perteneces' (with a sub-label 'Organización'), 'Rol en la Organización' (a dropdown menu), 'Nombre', 'Apellidos', 'Usuario', 'E-mail', 'Contraseña', and 'Confirmar contraseña'. A green 'Registrarse' button is at the bottom of the card. Below the card, there are links for '¿Ya estás registrado?' and 'Inicia sesión'.

Ilustración 31.- Captura de Página Registro

The screenshot shows the login page for the GESTIGAN system. At the top, there is a dark navigation bar with a menu icon, a logo, and buttons for 'Iniciar Sesión' and 'Registrarse'. The main content area features a central white card with the title 'Inicio de sesión en GESTIGAN'. The card contains input fields for 'Organizacion', 'ID Organizacion' (with a sub-label 'Identificador de la Organización'), 'E-mail', and 'Contraseña'. A green 'Iniciar Sesión' button is at the bottom of the card. Below the card, there are links for '¿Nuevo en GESTIGAN?' and 'Registrarse'.

Ilustración 32.- Captura Diseño Mobile Iniciar Sesión

### 7.9.3.- Inventario

Uno de los principales apartados del sistema, desde aquí se puede acceder a algunos de los módulos más importante del sistema como son el de las fincas o animales.

Está compuesto por tres partes, el encabezado o header de la página donde se muestra una imagen y se explica que tipo de tareas se puede realizar en este apartado. Otro apartado que se corresponde con la búsqueda de animales por crotal, el cual se explicará más adelante.

Para conocer más en profundidad estos dos apartados consultar el *Anexo VI – Manual de Usuario*.

El último apartado corresponde con las fincas registradas en el sistema, donde se pueden observar cada una de ellas de una forma sencilla, una tarjeta compuesta por una imagen y el identificado de la finca. Si pulsamos en una de esas tarjetas nos redirigirá a la página de finca deseada.

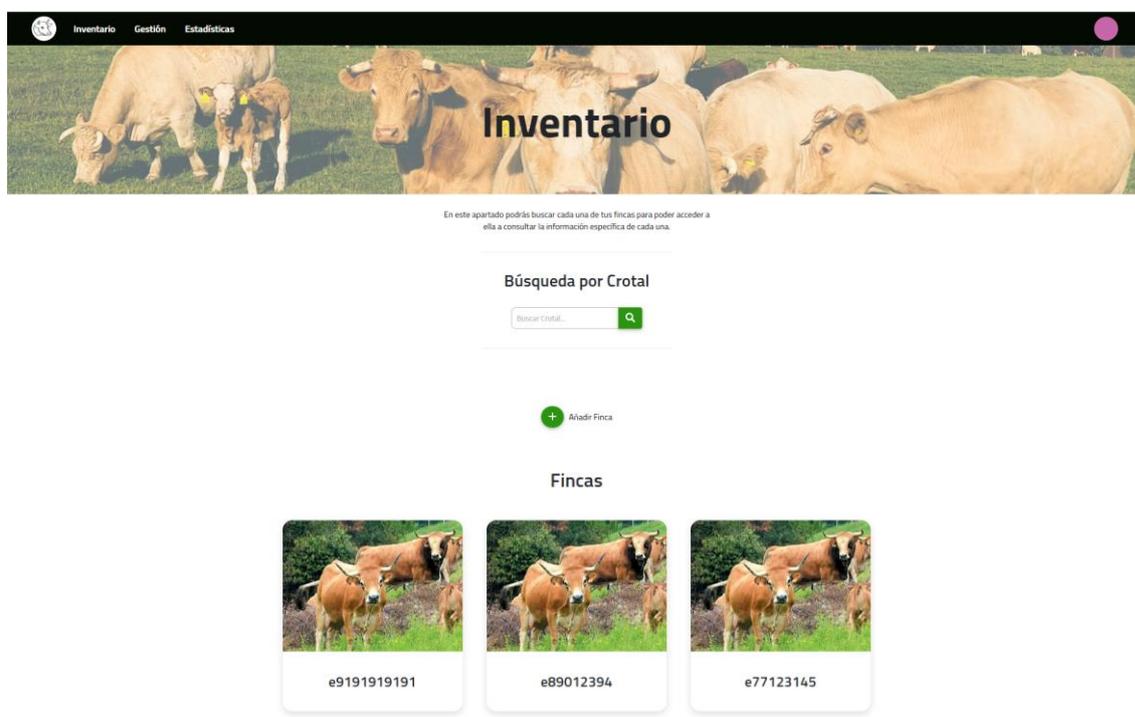


Ilustración 33.- Captura de Página Inventario



Ilustración 34.- Diseño Mobile Inventario

#### 7.9.4.- Finca

Esta página nos permitirá consultar información de la finca que deseemos, a ella podremos acceder a través de la página inventario como ya hemos mencionado antes o a través de esta misma página por el buscador que se mostrará a continuación.

Además de la información de la propia finca podremos consultar a través de una tabla, todos los animales asociados a esta finca, con esa tabla podremos realizar las típicas funciones CRUD: create (añadir animal), read (consultar animal), update (modificar animal) y delete (eliminar animal) sobre cada uno de los animales.

#### 7.9.5.- Tablas de Animales, Alimentos y Medicamentos

Aunque se encuentren en distintos apartados de la página web la funcionalidad y diseño de estas tablas es prácticamente el mismo.

Para conocer más en profundidad esta funcionalidad consultar el *Anexo VI – Manual de Usuario*.

Identificador	Fecha de nacimiento	Sexo	Raza	Peso	
e9191919191	24/08/1998	Macho	Lidia	500 Kg	
e22222222	02/08/2002	Hembra	Charolesa	300 Kg	
e987654321	02/08/2002	Macho	Charolesa	450 Kg	
e000011111	02/08/2002	Hembra	Charolesa	476 Kg	
<input type="text" value="identificador"/>	<input type="text" value="Fecha de nacimiento"/>	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value="554"/>	
e222211111	02/08/2002	Macho	Lidia	600 Kg	
e194126123	02/08/2002	Hembra	Rubia Gallega	300 Kg	

Ilustración 35.- Captura Modificar Tabla de Animales

### Alimentos

En este apartado podrás consultar cada uno de los Alimentos que la organización ha adquirido.

Código Receta	Nombre	Fecha de Adquisición
receta989	Pienso1	22/06/2022
receta0909	Pienso2	25/06/2022
receta3434	Pienso3	03/07/2022
receta5656	Pienso4	24/07/2022
receta1212	Pienso5	15/08/2022

Eliminar Modificar **Añadir**

Ilustración 36.- Captura de Tabla de Alimentos

Código Receta	Nombre	Fecha de Adquisición
receta111	Medicamento1	23/06/2022
receta222	Medicamento2	26/06/2022
receta333	Medicamento3	04/07/2022
receta444	Medicamento4	16/08/2022

Debe rellenar todos los campos

Cancelar **Guardar**

Ilustración 37.- Captura Error en formulario Añadir Medicamento

#### 7.9.6.- Estadísticas

Esta página nada más acceder cargará una primera gráfica que nos muestra de una forma más representativa la comparación de cabezas que existen en cada una de las fincas de la organización.

En cambio, la segunda gráfica según se accede está vacía, ya que el usuario tendrá que generarla, esta gráfica podrá ser una comparación entre razas o sexos dentro de una finca.

Para conocer más en profundidad esta funcionalidad consultar el *Anexo VI – Manual de Usuario*.

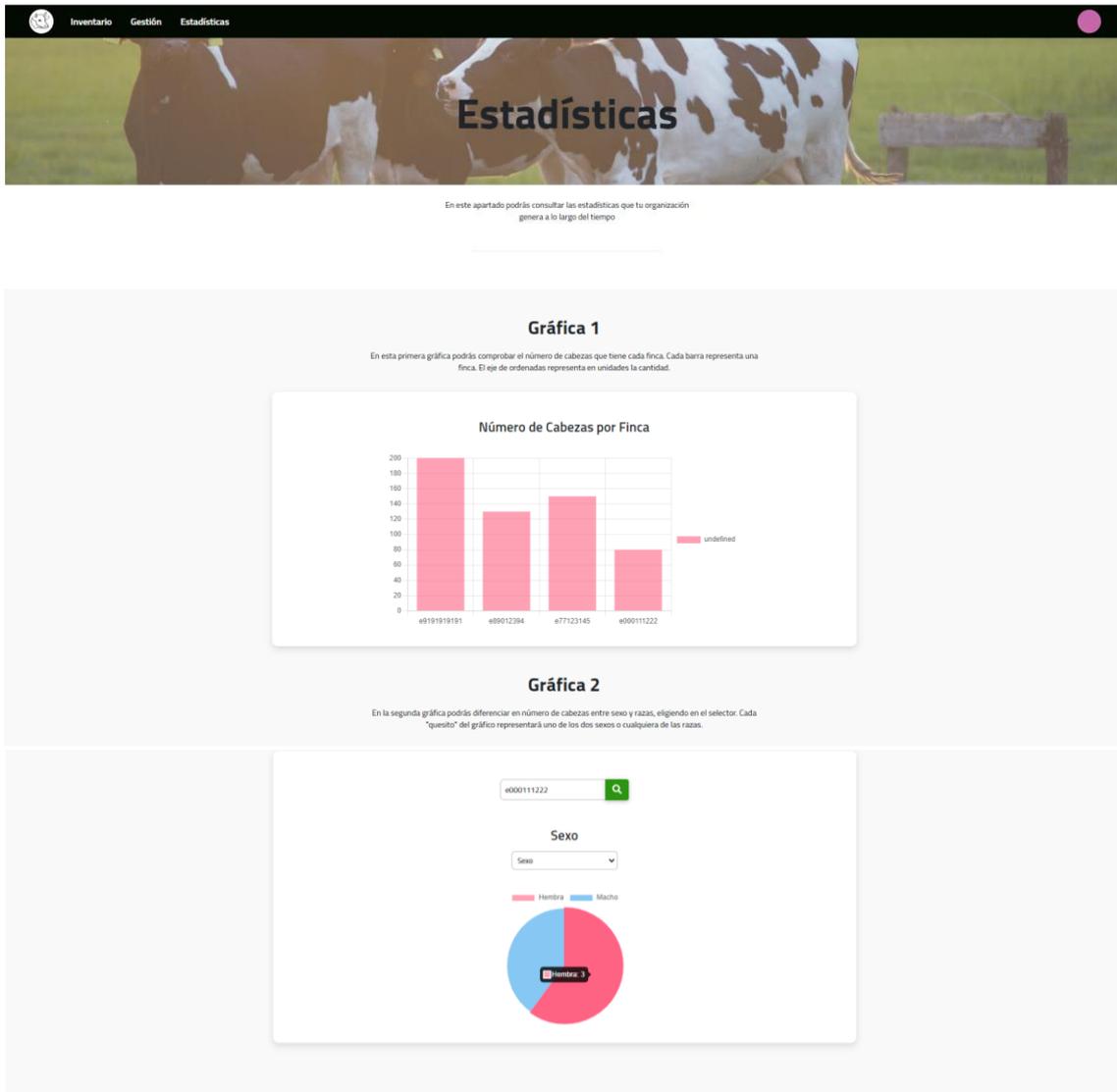


Ilustración 38.- Captura de Página Estadísticas

## Memoria del proyecto

### 7.9.7.- Gestión

Este apartado se mostrarán diferentes contenidos, entre ellos se podrá observar un bonito encabezado con un texto explicando las posibilidades que ofrece esta página.

Esta página, por tanto, reunirá todas las transacciones de animales realizadas en el sistema, bajas, altas...

Además de las tablas correspondientes a los alimentos y medicamentos adquiridos por la empresa.



Ilustración 39.- Captura de Página Gestión

### 7.9.8.- Animal

Esta página mostrará la información correspondiente a un animal, para poder acceder a ella podremos hacerlo a través del buscador de animales del inventario o a través de la tabla de animales de la finca a la que pertenezca este animal.

En esta página la única funcionalidad que existe es la del buscador de animal por crotal, el cual se ve reutilizado en este apartado para facilitar la búsqueda de los animales.

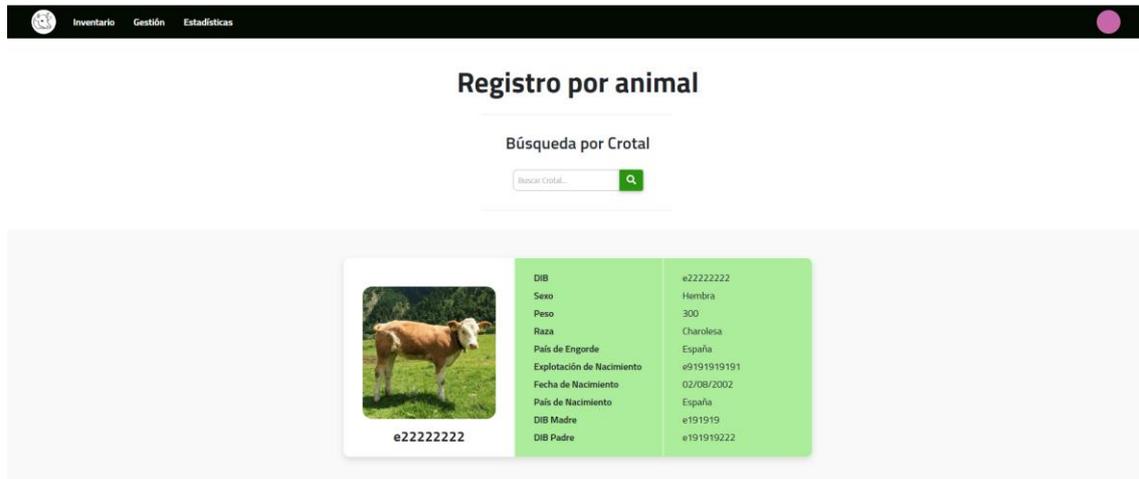


Ilustración 40.- Captura de Página Animal

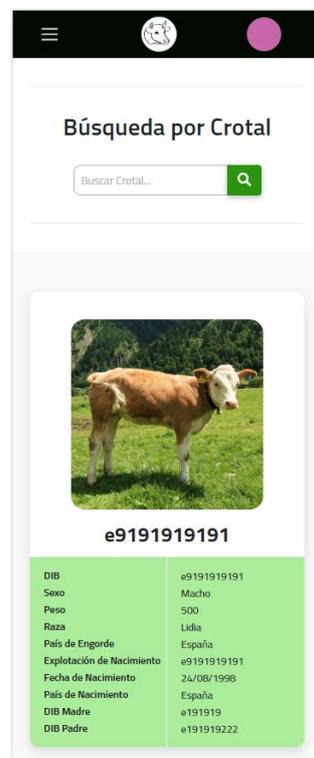


Ilustración 41.- Diseño Mobile Animal

### 7.9.9.- Perfil

Este apartado contendrá la información del usuario con la sesión iniciada. Podrá ser accedido a través del icono de la derecha de la navbar o barra de navegación.

En el se podrá editar el perfil, cambiar la contraseña, cambiar el e-mail o incluso eliminar la cuenta o la organización siempre y cuando el usuario autenticado tenga el rol de Administrador de una Organización.

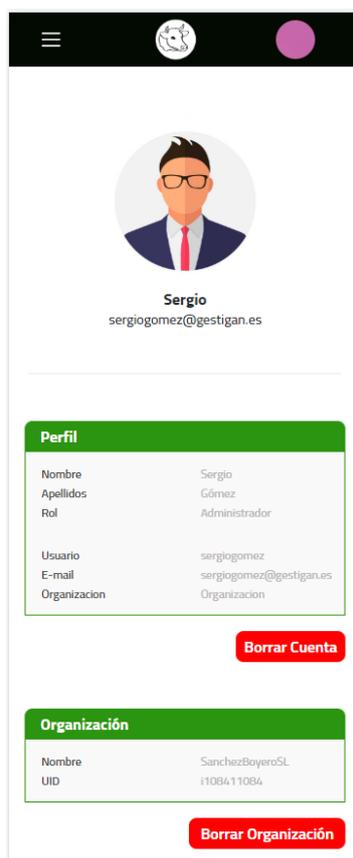


Ilustración 42.- Diseño Mobile Perfil

## Memoria del proyecto

### 7.9.10.- Administración

Este apartado puede ser accedido únicamente por parte del usuario con rol de Administrador, en él se podrá gestionar los permisos de los usuarios para que puedan consultar y modificar la información de la Organización de la cual es responsable, y por tanto convertirlos o no en Trabajadores de la Organización.

Para conocer más en profundidad esta funcionalidad consultar el *Anexo VI – Manual de Usuario*.

Inventario Gestión Estadísticas Administración

## Administración

En este apartado podrás gestionar como Administrador todos los permisos de los usuarios.

### Usuarios que desean entrar

Esta tabla contiene todos los usuarios que desean formar parte de la organización. Podrás seleccionar aquellos que desees agregar.

E-mail	Nombre	Apellidos	Usuario	
teresa@gestigan.es	teresa	gonz	teresagonz	<input type="checkbox"/>

Guardar

### Trabajadores de la organización

Esta tabla contiene todos los trabajadores que forman parte de la organización. Podrás seleccionar aquellos que desees quitar.

E-mail	Nombre	Apellidos	Usuario	
saraser@gestigan.es	saraser	sae	saraser	<input type="checkbox"/>

Guardar

→  
Avanzar a Inventario

Ilustración 43.- Captura de página Administración

E-mail	Nombre	Apellidos	Usuario	
saragomez@gestigan.es	sara	gomez	saragomez	<input checked="" type="checkbox"/>
inmagomez@gestigan.es	inma	gomez	inmagomez	<input checked="" type="checkbox"/>
marinagomez@gestigan.es	marina	gomez	marinagomez	<input type="checkbox"/>

Guardar

Ilustración 44.- Captura de elección de Usuarios en Administración

## 9.- Conclusiones y líneas de trabajo futuras

Una vez finalizado todo el proceso de desarrollo, es decir, que hayan terminado de realizarse todas las fases del desarrollo del proyecto se recogen en este apartado todas y cada una de las conclusiones obtenidas.

Además, se comentarán alguna mejorar o líneas de trabajo a futuro que podrán realizarse en el proyecto.

### 9.1.- Conclusiones

Tras haber finalizado el proyecto se puede afirmar que se han cumplido los objetivos marcados y los requisitos definidos.

Por tanto, podemos concluir que,

- La elección de las herramientas con las que se trabajan marcan la diferencia y su elección es importante, ya que trabajar con una herramienta o con otros puede cambiar mucho la forma en la que se desarrolla el proyecto. Por ejemplo, la elección de trabajar con el framework Angular provoca que el desarrollo de la aplicación web se haga de una forma particular, que ha permitido un desarrollo rápido y organizado.
- Se ha conseguido crear un servicio web capaz de realizar con seguridad y correctamente funciones de gestión para una organización ganadera.
- Se ha conseguido desarrollar un proyecto que cumpliera con una necesidad surgida por parte de un cliente, pudiéndola cubrir en todos los aspectos.
- Se ha podido comprobar cuál es el esfuerzo necesario para realizar un proyecto de estas características de forma individual, obteniendo tras su finalización una experiencia que será imprescindible para proyectos posteriores.
- Las herramientas y conocimientos que se han adquirido a lo largo de la carrera han sido necesarias para el desarrollo del proyecto. Especialmente con aquellas aprendidas en las asignaturas de Ingeniería del Software o la de Gestión de Proyectos.

### 9.2.- Líneas de trabajo futuras

Algunos de los aspectos a mejorar y que aumentarían la eficiencia serían el desarrollo de algunas aplicaciones o funciones adicionales que aumentarían la satisfacción del usuario.

Por ejemplo:

- Implementar una función para poder leer documentos Excel u otro tipo de documentos donde se guardase la información de la gestión de una ganadería antes de pasar a este nuevo sistema, ya que ahorraría mucho tiempo a la hora de trasladar toda la información de la organización.

## Memoria del proyecto

- Sería interesante tratar de ver la viabilidad de desarrollar una aplicación que fuese capaz de leer el crotal del animal a través de la cámara del móvil para poder buscar la información del animal de una forma más sencilla.
- Otra de las funciones que mejorarían este servicio es que pudiese guardarse la información de la organización para poder verla sin conexión, pensando sobre todo en esas zonas del campo donde la cobertura es mínima.

Además, sería interesante, si el sistema escala a un gran número de usuarios, poder realizar algunos estudios, para conocer el grado de capacidad de la aplicación para interactuar de manera sencilla con los usuarios final (entendibilidad) midiendo el número de veces que el usuario accede a páginas web y rápidamente las abandona, comprobando el número de mensaje de error visualizado por parte del usuario...

También estudios que midan la efectividad de la aplicación controlando el número de sesiones de trabajo o transacciones diarias abandonadas, o incluso estudios para controlar la capacidad que tiene la aplicación web para ayudar a los usuarios finales a alcanzar sus objetivos en menos tiempo que el requerido para alcanzar los mismos objetivos utilizando otros instrumentos y en función de los resultados poder mejorar el sistema.

## 10.- Referencias

- Acibeiro, M. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://www.lucushost.com/blog/que-es-node-js/>
- Adictos al Trabajo. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://www.adictosaltrabajo.com/2022/07/01/como-anadir-graficos-en-tu-web-con-chart-js/>
- Bootstrap. (Acceido Jueves 25 de Agosto de 2022). Obtenido de <https://getbootstrap.com/docs/5.2/getting-started/introduction/>
- Capterra. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://www.capterra.es/software/145716/visual-paradigm>
- Cardona, M. P. (Accedido Martes 23 de Agosto de 2022). Obtenido de <https://www.iebschool.com/blog/firebase-que-es-para-que-sirve-la-plataforma-desarrolladores-google-seo-sem/>
- DevCamp . (Accedido Miércoles 24 de Agosto de 2022). Obtenido de <https://devcamp.es/que-es-libreria-programacion/>
- Developer. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://developer.mozilla.org/es/docs/Web/HTML>
- Diego Santos. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://blog.hubspot.es/website/que-es-css>
- Ediciones-Eni. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://www.ediciones-eni.com/open/mediabook.aspx?idR=d11b8603ec6ef90a5234506bfc48846c>
- Edix. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://www.edix.com/es/instituto/adobe-xd/>
- Fernández, Y. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://www.xataka.com/basics/que-github-que-que-le-ofrece-a-desarrolladores>
- Firebase. (Accedido Martes 23 de Agosto de 2022). Obtenido de <https://firebase.google.com/products/realtime-database>
- Firebase. (Accedido Martes 23 de Agosto de 2022). Obtenido de <https://firebase.google.com/docs/auth>
- Firebase. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://firebase.google.com/docs/libraries?hl=es-419>
- GanderosOnline. (Accedido Lunes 22 de Agosto de 2022). Obtenido de <https://www.ganderosonline.com/vacuno-de-carne>
- Gómez, M. J. (s.f.). Obtenido de <https://leo.uniandes.edu.co/images/Guias/Estadodelarte.pdf>
- Hernández, U. (Acceido Jueves 25 de Agosto de 2022). Obtenido de <https://codigofacilito.com/articulos/typescript>
- HiberusBLog. (Accedido Martes 23 de Agosto de 2022). Obtenido de <https://www.hiberus.com/crecemos-contigo/que-es-angular-y-para-que-sirve/>

## Memoria del proyecto

- Holgado, A. G. (Accedido Viernes 26 de Agosto de 2022). Obtenido de <https://repositorio.grial.eu/bitstream/grial/2552/1/8.%20Modelo%20de%20dominio-2022.pdf>
- Hostinger. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://www.hostinger.es/tutoriales/que-es-json>
- Hostinger. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://www.hostinger.es/tutoriales/que-es-npm>
- LucidChart. (Jueves 25 de Agosto de 2022). Obtenido de <https://www.lucidchart.com/pages/es/que-es-el-lenguaje-unificado-de-modelado-uml>
- OpenWebinars. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://openwebinars.net/blog/que-es-visual-studio-code-y-que-ventajas-ofrece/>
- Peñalvo, J. G. (Accedido Viernes 26 de Agosto de 2022). Obtenido de [https://repositorio.grial.eu/bitstream/grial/1144/1/IS\\_I%20Tema%205%20-%20Proceso%20Unificado.pdf](https://repositorio.grial.eu/bitstream/grial/1144/1/IS_I%20Tema%205%20-%20Proceso%20Unificado.pdf)
- QualityDevs. (Accedido Lunes 22 de Agosto de 2022). Obtenido de <https://www.qualitydevs.com/2019/09/16/que-es-angular-y-para-que-sirve/>
- QualityDevs. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://www.qualitydevs.com/2019/09/16/que-es-angular-y-para-que-sirve/>
- rchavez. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://soporte.codefu.net.ni/sourcetree-tutorial/#:~:text=%C2%BFQu%C3%A9%20es%20SourceTree%3F,usuario%20que%20prvee%20la%20aplicaci%C3%B3n>.
- Rckstr. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://rckstr1.github.io/2019/09/06/GLTrack4.html#:~:text=Ng2%2Dcharts%20es%20un%20paquete,Angular%20%20utilizaremos%20ng2%2Dcharts>.
- Ro. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://www.acontracorrientech.com/routing-angular-guia-completa-parte-1/>
- SergioOswaldo. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://www.alphaconsultoria.com/que-es-microsoft-project-y-para-que-sirve/>
- Significados.com. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://www.significados.com/photoshop/>
- Sol Gonzáles. (Accedido Martes 23 de Agosto de 2022). Obtenido de <https://www.cyberclick.es/que-es/diseño-web-responsive>
- TysMagazine. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://tysmagazine.com/paletton-una-herramienta-util-los-cartografos/#:~:text=En%20esta%20ocasi%C3%B3n%20os%20presentamos,APP%20online%20creada%20en%202002>.
- VacApp. (Accedido Lunes de 25 de Agosto). Obtenido de <https://vacapp.net/es>

## Memoria del proyecto

Vela, L. (Accedido 25 de Agosto de 2022). Obtenido de <https://prezi.com/hgfpam-ynxpo/estado-del-arte-de-la-ingenieria-de-software/>

Wikipedia. (Acceido Jueves 25 de Agosto de 2022). Obtenido de <https://es.wikipedia.org/wiki/TypeScript>

Wikipedia. (Accedido Jueves 25 de Agosto de 2022). Obtenido de <https://es.wikipedia.org/wiki/Node.js>

## 11.- Bibliografía

- La lista de reproducción completa de como hacer un proyecto en Angular:  
[ [https://youtu.be/fXpMiweCC\\_o](https://youtu.be/fXpMiweCC_o) ]
- Documentos de Gestión de Proyectos  
[ <https://studium21.usal.es/course/view.php?id=2101004> ]
- Documentos de las asignaturas de Ingeniería Informática de la USAL: Ingeniería del Software I , Ingeniería del Software II como:  
[ <https://repositorio.grial.eu/bitstream/grial/2552/1/8.%20Modelo%20de%20dominio-2022.pdf> ]  
[ [https://repositorio.grial.eu/bitstream/grial/1144/1/IS\\_I%20Tema%205%20-%20Proceso%20Unificado.pdf](https://repositorio.grial.eu/bitstream/grial/1144/1/IS_I%20Tema%205%20-%20Proceso%20Unificado.pdf) ]
- Serie de tutoriales: como comenzar un proyecto en Angular desde 0  
[ <https://youtu.be/AR1tLGQ7COs> ]