

# ANIMALES SIN ABRIGO

## SPRINT 1

### 1. Documento Scrum

#### 1.1. Estructura y funcionamiento del equipo Scrum

La organización del equipo se ha tratado de forma fundamental para el inicio del proyecto. Desde el primer momento, hemos intentado organizar los roles de tal forma que al final del proyecto, todos los integrantes del equipo hayamos colaborado en cada una de las áreas del mismo. Es por ello que estos roles quizá se encuentran modificados con respecto al final del proyecto. Los primeros roles designados han sido:

- *Scrum Master*: Antonio Álvarez
- *Software design and architecture*: Javier Peña y Eneko Retolaza
- *GUI design and development*: Fabrizio Zeballos, Guillermo López y Eneko Retolaza
- *Backend development*: Antonio Álvarez y Fabrizio Zeballos
- *Product Owner and Management*: Alonso Salgueiro y Alberto Martínez

A lo largo del desarrollo se buscará que los integrantes de las secciones de desarrollo roten entre sí, para asegurar el completo conocimiento del proyecto.

Las herramientas utilizadas para manejar la información han sido:

- GitHub: Hemos creado un repositorio en remoto para controlar las versiones y el código programado y manejado por Git de forma local en cada uno de los integrantes.
- Google Drive: Hemos creado una unidad compartida como plataforma de redacción de documentos formales simultáneamente.
- Discord: Hemos creado un servidor como plataforma de comunicación y chat de voz. En todo momento en el que uno de los integrantes se encuentre trabajando en el proyecto, debe entrar en la llamada del servidor, para conocer en todo momento quién está trabajando y facilitar la comunicación.
- Miró: Tenemos creado un grupo con pizarras como punto de brainstorming y elaboración de los documentos correspondientes a la metodología scrum, como las historias de usuario o el product backlog.
- Figma: Hay un proyecto en el que tenemos hechos los bocetos de las pantallas que queremos implementar, con las imágenes, botones y funcionamiento.

Como se puede observar, todas estas herramientas tienen en común las facilidades para la edición y el trabajo de forma remota y simultánea. Las reuniones presenciales obligatorias son todas aquellas organizadas durante horario lectivo, especialmente los martes de 13:00 a 15:00, y también aquellas extraordinarias que organicemos de forma prioritaria. El resto de reuniones y trabajo se realiza de forma remota.

## 1.2. Historias de Usuario

Las actuales historias de usuario tienen proyecto para ser implementadas en el proyecto a lo largo de su desarrollo. A medida que vayamos avanzando con el desarrollo, encontraremos los impedimentos y/o decisiones que nos harán descartarlas.

Iniciar Sesión <input type="checkbox"/>	Cerrar sesión <input type="checkbox"/>	Eliminar Cuenta <input type="checkbox"/>	Registrarse <input type="checkbox"/>	Blog <input type="checkbox"/>
Adoptar animal <input type="checkbox"/>	Solicitar Curso de adopción <input type="checkbox"/>	Apadrinar Animal <input type="checkbox"/>	Acogida Animal <input type="checkbox"/>	Curso adopción <input type="checkbox"/>
Añadir Animal <input type="checkbox"/>	Eliminar Animal <input type="checkbox"/>	Actualizar Animal <input type="checkbox"/>	Ver Lista Animales (adopter) <input type="checkbox"/>	Ver Lista Animales (refuge) <input type="checkbox"/>
Solicitar Voluntarios <input type="checkbox"/>	Solicitar Trabajadores <input type="checkbox"/>	Enviar CV <input type="checkbox"/>	Donaciones <input type="checkbox"/>	
Contactar refugio <input type="checkbox"/>	FAQ <input type="checkbox"/>	Sobre Nosotros <input type="checkbox"/>	Reviews de los Refugios <input type="checkbox"/>	
Solicitar examen <input type="checkbox"/>	Ofrecer plazas Guardería Animales <input type="checkbox"/>	Buscar plazas Guardería Animales (adopter) <input type="checkbox"/>	Eventos promocionales <input type="checkbox"/>	

### Iniciar sesión:

ID: 2

Como usuario quiero poder iniciar sesión.

Criterios de validación:

- Verificar que el usuario y la contraseña sean correctos y dar acceso al sistema.

Valor: 8

Prioridad: 5

### Cerrar sesión:

ID:

Como usuario quiero poder cerrar mi sesión.

Criterios de validación:

- Verificar que se ha cerrado la sesión del usuario y que ya no tiene acceso al sistema.

Valor: 2

Prioridad: 5

### Eliminar cuenta:

ID: 8

Como usuario quiero poder eliminar mi cuenta.

Criterios de validación:

- Verificar que se ha eliminado el usuario de la base de datos y con ello todos los datos del usuario.

Valor: 3

Prioridad: 4

### Registrarse:

ID: 1

Como usuario quiero poder registrarme para poder interactuar en la aplicación.

Criterios de validación:

- Comprobar que el nombre y contraseña tienen los requerimientos necesarios.
- Comprobar que se ha creado el usuario en la base de datos.

Valor: 5

Prioridad: 3

Blog:

ID:

Como refugio quiero poder publicar entradas en el blog para dar información y noticias.

Criterios de validación:

- Comprobar que la entrada tiene un autor del refugio.
- Comprobar que la entrada no está vacía.

Valor: 8

Prioridad: 2

Adoptar animal:

ID: 4

Como usuario quiero poder adoptar un animal.

Criterios de validación:

- Poder dar al usuario una posibilidad de adoptar animales en los centros de acogida.

Valor: 7

Prioridad: 5

Solicitar curso de adopción:

ID:

Como usuario quiero poder hacer un curso de adopción de animales.

Criterios de validación:

- Enseñar al usuario las habilidades necesarias para poder adoptar a un animal.
- Comprobar el buen estado psicológico del adoptante para poder adoptar un animal.

Valor: 5

Prioridad: 1

Apadrinar animal:

ID: 10

Como usuario quiero poder apadrinar a un animal.

Criterios de validación:

- Verificar que el usuario haya realizado el pago.
- Verificar que el animal haya sido apadrinado.

Valor: 7

Prioridad: 3

Acogida animal:

ID: 9

Como usuario quiero poder acoger un animal.

Criterios de validación:

- Verificar que el usuario está en plenas capacidades para acoger un animal.
- Hacer la transacción de acoger un animal.

Valor: 7

Prioridad: 3

Curso adopción:

ID: 11

Se pone disponible el curso con la información necesaria sobre el curso de adopción para el usuario.

Criterios de validación:

- Poner disponible la información que necesita el usuario sobre el curso.

Valor: 6

Prioridad: 2

Añadir animal:

ID: 5

Se añade la opción al sistema gestor de poder añadir un animal a un refugio.

Criterios de validación:

- Comprobar que los datos del animal son coherentes y añadirlo a la base de datos.

Valor: 2

Prioridad: 4

Eliminar animal:

ID: 16

Se añade la opción al sistema gestor de poder eliminar un animal del refugio.

Criterios de validación:

- Eliminar el animal de la base de datos.

Valor: 3

Prioridad: 4

Actualizar animal:

ID: 17

Se añade la opción de actualizar los datos de un animal ya existente en el sistema.

Criterios de validación:

- Comprobar que los nuevos datos del animal son coherentes.

Valor: 2

Prioridad: 3

Ver lista animales (adopter):

ID: 3

Como adopter quiero poder visualizar la lista de animales disponibles.

Criterios de validación:

- Comprobar los animales que hay disponibles en el centro de acogida y mostrarlos.

Valor: 6

Prioridad: 2

Ver lista animales (refuge):

ID: 6

Como administradores del refugio quiero poder ver la lista de animales que hay disponibles.

Criterios de validación:

- Comprobar la lista de animales disponibles y mostrarlos.

Valor: 2

Prioridad: 2

Solicitar voluntarios:

ID:2

Como refugio quiero poder solicitar voluntarios para mejorar la plantilla y el funcionamiento del refugio.

Criterios de validación:

- Poder dar al refugio una posibilidad de mejorar la plantilla.
- Comprobar si hay espacio para nuevos voluntarios y solicitarlos.

Valor: 1

Prioridad: 1

Solicitar trabajadores:

ID:2

Como centro de acogida, quiero poder solicitar nuevos trabajadores

Criterios de validación:

- Comprobar si hay espacio para nuevos trabajadores y solicitarlos

Valor: 1

Prioridad:1

Enviar CV:

ID:2

Como candidato, quiero poder enviar el CV al centro de acogida

Criterios de validación:

- Comprobar que el CV sea válido y meterlo al centro de solicitudes

Valor: 2

Prioridad:1

Donaciones:

ID:2

Como centro de acogida, quiero poder recibir donaciones para los centros de acogida

Criterios de validación:

- Comprobar que la transacción se ha realizado y codificado correctamente

Valor: 4

Prioridad:1

#### Contactar refugio:

ID:2

Como usuario quiero poder contactar con el refugio

Criterios de validación:

- Comprobar que el refugio está disponible para poder ser contactado

Valor: 5

Prioridad:1

#### Sobre nosotros:

ID:2

Como refugio quiero poder explicar los motivos por los que tenemos un refugio y darme a conocer.

Criterios de validación:

- Comprobar que el usuario coincide con el refugio
- Comprobar que se ha creado bien la descripción

Valor: 5

Prioridad:1

#### Reviews de los refugios:

ID:2

Como adoptante quiero poder dar y consultar reviews para valorar a los mejores refugios.

Criterios de validación:

- Comprobar la lista de las críticas de antiguos adoptantes

Valor: 6

Prioridad:2

#### Solicitar examen:

ID:2

Como usuario quiero poder solicitar un examen

Criterios de validación:

- Comprobar que el usuario es compatible con el examen

Valor: 6

Prioridad:1

#### Ofrecer plazas guardería animales:

ID:2

Como usuario quiero poder dejar a mi animal al cuidado del refugio de forma temporal.

Criterios de validación:

- Comprobar el usuario que va a dejar a su animal
- Comprobar que los datos del animal son correctos

- Comprobar los datos del servicio: fechas de inicio y final, datos del usuario y del animal, precio, etc.

Valor: 5

Prioridad:5

#### Buscar plazas guardería animales:

ID:2

Como usuario quiero poder buscar plazas para guardería de animales

Criterios de validación:

- Comprobar que hay plazas de guardería en el centro solicitado para poder ofrecerle asistencia al usuario solicitante

Valor: 6

Prioridad:1

#### Eventos promocionales:

ID:2

Como centro de acogida, quiero poder promocionar eventos sobre el centro

Criterios de validación:

- Comprobar que la información haya llegado bien a los potenciales clientes

Valor: 3

Prioridad:1

## **1.3. Sprint Reviews**

### Review Sprint 1 - 15/03/2022

Organizamos una reunión informal para analizar especialmente el grado de éxito del sprint y los fallos que hemos tenido durante la organización.

Los principales fallos que hemos encontrado han sido:

- La falta de orden a la hora de empezar a trabajar. Esto es, querer diseñar partes que necesitaban previo diseño de otras que no estaban diseñadas, entre otros casos.
- La falta de asistencia en algunas ocasiones para reuniones, que nos ha dificultado un poco la comunicación.
- Hemos necesitado mucho tiempo para integrar GitHub con nuestro código y aprender a usarlo.
- Hemos encontrado problemas con JavaFX, que es la librería que hemos decidido usar para nuestras interfaces de usuario.

Sin embargo, el sprint ha sido exitoso porque hemos encontrado muchísimas ideas de implementación y mejora de los servicios de la aplicación, y hemos conseguido transmitir de forma satisfactoria los objetivos e ideas del proyecto, por lo que todos tenemos una idea general de cómo hacer las cosas.

### Review Sprint 2 - 29/03/2022

Organizamos una reunión on line para ver el grado de éxito del sprint, los fallos y las mejoras posibles.

Principales fallos:

- Falta de organización y coordinación del equipo

- Incompatibilidad de horarios entre los diferentes integrantes
- Fallos notables en los documentos del primer sprint: clases y diagrama uml
- Grandes cambios en la estructura del programa, tipo de interfaz gráfica (javaFX a swing), lo que ha provocado que desechamos el proyecto inicial y empezáramos de nuevo

Aun con todos estos problemas, hemos conseguido que quede claro la estructura del programa, y hemos hecho los diagramas necesarios para aclarar la estructura. Además, hemos elegido los estándares de diseño de programa necesarios para llevarlo a cabo.

## 1.5. Sprint Plannings

### Sprint 1 Planning

En cuanto a documentación, hemos planeado preparar la base de los documentos UML y la organización del proyecto. Pensamos que establecer unos objetivos, historias de usuario y diseño fijos y claros es bastante importante para que todos los integrantes entiendan el objetivo del proyecto.

En relación a lo implementado para el primer sprint, el objetivo es conseguir el manejo de los usuarios de forma correcta. Para su almacenamiento, queremos usar “*JSON Objects*”, que nos facilitará cargar los documentos sin tener que aplicar bases de datos en nuestro proyecto.

### Sprint 2 Planning

Después de los problemas del sprint 1, hemos replanteado el proyecto entero y hemos ido cambiando el enfoque de cada parte del proyecto, y ver qué patrones podríamos usar para hacer que todo funcione de forma correcta.

Para la interfaz gráfica, hemos decidido dejar de lado la idea inicial de usar javaFX y empezar a implementar la interfaz con swing, ya que tenemos conocimientos de cómo utilizarlo y tenemos más conocimientos. En cuanto a la estructura de objetos, hemos reorganizando los objetos, haciendo un diagrama de cómo se relacionan entre sí.

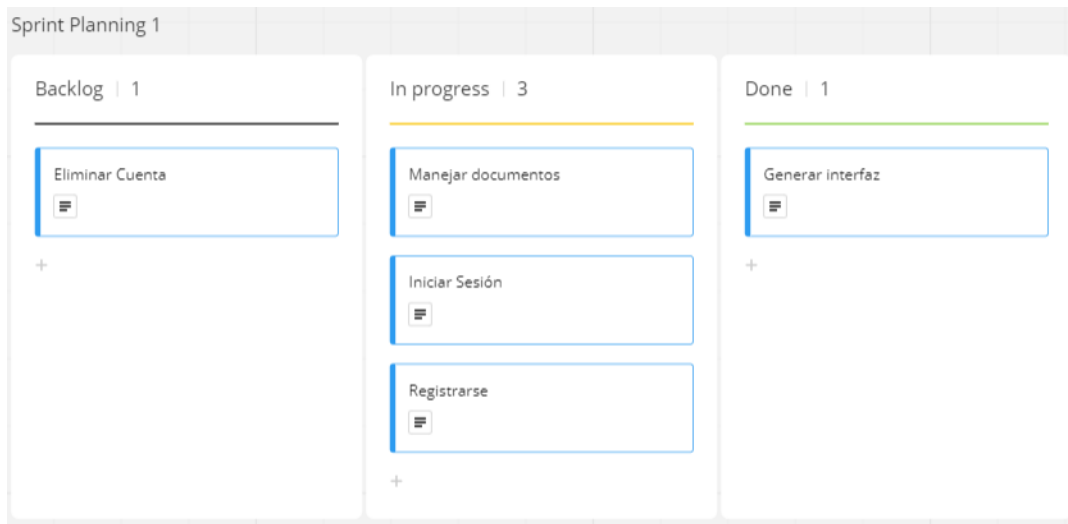


## 1.6. Product Backlog

ID	HISTORIA DE USUARIO	PRIORIDAD	VALOR
1	Como usuario quiero poder registrarme para mantener mi cuenta y mi información	3	5
2	Como usuario quiero poder iniciar sesión para acceder a mi cuenta y a las funcionalidades de la aplicación	5	8
3	Como adoptante quiero poder comprobar los animales disponibles para decidir correctamente	2	6
4	Como adoptante quiero poder adoptar animales para contactar con el refugio y recibir el animal	5	7
5	Como refugio quiero poder añadir animales para aumentar mi refugio	4	2
6	Como refugio quiero poder comprobar los datos de los animales adoptados para asegurar su estado		
7	Como refugio, quiero poder solicitar voluntarios para mejorar la plantilla y el funcionamiento del refugio		
8	Como usuario, quiero poder eliminar mi cuenta para preservar mis datos y finalizar mis adopciones.	4	3
9	Como adoptante, quiero poder acoger a un animal para cuidarlo y liberar el refugio hasta que alguien lo adopte		
10	Como adoptante, quiero poder apadrinar a un animal para dar dinero al refugio y apoyarlo		
11	Como refugio quiero poder ofertar cursos de adopción para poder	2	6
12	Como refugio quiero poder publicar entradas en el blog para dar información y noticias	2	8
13	Como adoptante quiero poder solicitar realizar el examen para poder adoptar animales		
14	Como adoptante quiero poder realizar donaciones libres para		
15	Como adoptante quiero poder acceder a la FAQ para resolver las dudas		
16	Como refugio quiero poder eliminar animales del refugio para no dar información falsa	4	3
17	Como refugio quiero poder editar la información de los animales para que estén actualizados	3	2
18	Como adoptante quiero poder dar y consultar reviews para valorar a los mejores refugios	2	6
19	Como adoptante quiero poder recibir notificaciones y promociones para estar informado		
20	Como adoptante quiero poder utilizar el servicio de guardería para ocasiones en las que no me pueda hacer cargo del animal		

## 1.7. Sprint Backlog

El sprint backlog lo hemos elaborado a partir de un Tablero Kanban gracias a la herramienta de Miró. La tarea más fundamental terminada ha sido la interfaz de login, pues hemos tenido algunas dificultades para terminar de forma completa el resto de tareas.



## 1.8. Descripción del trabajo realizado por cada miembro del grupo

### Sprint 1

El trabajo realizado por cada miembro del grupo está muy relacionado en este primer sprint con los roles establecidos en el punto 1.1. anterior.

**Fabrizio Zeballos** se ha encargado del diseño de la GUI junto a **Guillermo López**, y se han encargado de estudiar en profundidad el funcionamiento de JavaFX y han estudiado cómo implementarlo en nuestro proyecto. Además, junto a la ayuda de **Eneko Retolaza** han programado el código correspondiente a la pantalla de inicio. Además también han preparado los objetos JSON y su estructura.

**Javier Peña** se ha encargado del diseño de los diagramas UML de clases y el diseño de los objetos básicos, como los animales o los usuarios.

**Alonso Salgueiro** y **Alberto Martínez** se han encargado de elaborar las historias de usuario y preparar los documentos de backlog e historias de usuarios, después del brainstorming y las ideas principales del proyecto. **Antonio Álvarez** les ha ayudado en la revisión de estos documentos, así como la planificación y asignación de tareas del proyecto. También ha trabajado en el ejecutable de GitHub junto a **Fabrizio Zeballos**.

### Sprint 2

Hemos creado consenso en la arquitectura y diseño de la aplicación. **Eneko Retolaza**, **Antonio Álvarez** y **Guillermo López** se encargaron de desarrollar el modelo de dominio con la intención de satisfacer las necesidades del cliente (ficticio). **Javier Peña** y **Fabrizio Zeballos** se han encargado del diseño de los patrones creacionales de los apartados importantes del proyecto, enfocándose en alto

nivel (diseño UML). **Alonso Salgueiro** y **Alberto Martínez** han seguido la misma metodología y han estado trabajando en la documentación sprint backlog, entre otros.

**Eneko Retolaza** y **Fabrizio Zeballos** han creado los JSON que se utilizarán como recursos para los datos de los usuarios y las entidades correspondientes.

## 2. Documento de Diseño UML

### 2.1. Arquitectura

Accesible desde el repositorio de GitHub.

### 2.2. Diseño

Accesible desde el repositorio de GitHub.

## 3. Código

El código se encuentra en el mismo repositorio de GitHub que este documento, y está accesible para todos los colaboradores y visitantes, pues se trata de un repositorio público.

## 4. Referencias

A continuación se encuentran enlaces e información extra acerca de las herramientas utilizadas:

- GitHub: <https://github.com>
- Google Drive: [https://www.google.com/intl/es\\_es/drive/](https://www.google.com/intl/es_es/drive/)
- Discord: <https://discord.com>
- Miró: <https://miro.com>
- Figma: <https://www.figma.com>