

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

DESARROLLO DE UN SISTEMA WEB Y APLICACIÓN MÓVIL PARA LA GESTIÓN DE FUNDACIONES DE ANIMALES DE COMPAÑÍA EN LA CIUDAD DE QUITO

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN ANÁLISIS DE SISTEMA INFORMÁTICOS

MARÍA FERNANDA TIBANTA LEGÑA

maria.tibanta@epn.edu.ec

EDISON DAVID HERNÁNDEZ ALMAGRO

edison.hernandez@epn.edu.ec

DIRECTOR: Ing. Byron Loarte, MSc.

byron.loarteb@epn.edu.ec

CODIRECTOR: Ing. Luz Marina Vintimilla, MSc.

mariana.vintimilla@epn.edu.ec

Quito, abril 2020

DECLARACIÓN

Nosotros Hernández Almagro Edison David y Tibanta Legua María Fernanda, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

Sin perjuicio de los derechos reconocidos en el primer párrafo del artículo 114 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación - COESC-, somos titulares de la obra en mención y otorgamos una licencia gratuita, intransferible y no exclusiva de uso con fines académicos a la Escuela Politécnica Nacional. Entregaremos toda la información técnica pertinente. En el caso de que hubiese una explotación comercial de la obra por parte de la EPN, se negociará los porcentajes de los beneficios conforme lo establece la normativa nacional vigente.

Edison David Hernández Almagro

María Fernanda Tibanta Legña

CERTIFICACIÓN

Certificamos que el presente trabajo fue desarrollado por Hernández Almagro Edison David y Tibanta Legña María Fernanda, bajo nuestra supervisión.

Ing. Byron Loarte, MSc.

DIRECTOR DEL PROYECTO

Ing. Luz Marina Vintimilla, MSc

CODIRECTORA DEL PROYECTO

AGRADECIMIENTO

Quiero agradecer primeramente a Dios, por ser quien me guió en momentos más importantes y difíciles de mi vida para poder seguir adelante.

A mis padres y familiares, quienes fueron pilares fundamentales para conseguir este logro y por estar conmigo todo este tiempo, dándome ánimos y fuerzas para seguir en los momentos más difíciles de mi vida y creer en mí siempre, permitiéndome cumplir una meta más de mi vida.

A mis amigos, quienes formaron parte de mi día a día y siempre me apoyaron en todo momento.

A todo el personal que conforma la ESCUELA POLITÉCNICA NACIONAL, quienes transmitieron sus conocimientos, aportando un granito de arena para mi formación.

Al Ing. Byron Loarte, por su amistad y dedicación, quien con sus conocimientos y experiencia logró solventar cualquier duda, guiar nuestras ideas y por hacer posible la culminación de este proyecto.

A la Ing. Luz Marina Vintimilla, por su apoyo, cooperación y conocimientos brindados en cada revisión, mismos que ayudaron a la culminación del presente proyecto.

EDISON DAVID HERNÁNDEZ ALMAGRO

AGRADECIMIENTO

Quiero agradecer en primer lugar a Dios, quien gracias a su bendición siempre me acompaña en momentos de debilidad y difíciles, ayudándome a vencer todos los obstáculos que se han presentado, siendo mi apoyo, fortaleza y guía en cada paso que doy.

A mis padres y seres queridos, quienes han estado a mi lado, guiándome y apoyándome en cada etapa de mi vida, para seguir adelante y así cumplir con mis metas y objetivos planteados, por sus consejos que me han ayudado a ser mejor persona y por siempre darme palabras de aliento para no decaer en cada paso que he dado.

A mis compañeros y amigos, quienes me brindaron su amistad, apoyo y consejos en momentos de dificultad.

A todos los Ingenieros, quienes impartieron sus conocimientos en cada hora de clases y que ahora son útiles en mi vida laboral.

Al Ing. Byron Loarte, por su tiempo, apoyo y dedicación, quien siempre estuvo con nosotros guiando y solventando dudas que se iban presentando a lo largo del proyecto hasta su culminación.

A la Ing. Luz Marina Vintimilla, quien fue mi tutora académica y siempre estuvo al pendiente de mí; brindándome consejos en cada tutoría y por apoyarnos en el desarrollo de este proyecto.

MARÍA FERNANDA TIBANTA LEGÑA

DEDICATORIA

Lleno de regocijo amor y esperanza, dedico este proyecto, a cada uno de mis seres queridos.

A Dios, quien ha sido mi guía, fortaleza y me ha brindado su mano de fidelidad, amor y por estar conmigo hasta el día de hoy.

A mis padres, quienes con su amor, paciencia y esfuerzo me han permitido llegar a cumplir hoy un sueño más, gracias por inculcar en mí el ejemplo de esfuerzo y valentía.

EDISON DAVID HERNÁNDEZ ALMAGRO

DEDICATORIA

Dedico este trabajo a Dios, quien me ha dado la fuerza para no rendirme y ser guía a lo largo de mi vida.

A mi familia y seres queridos, quienes estuvieron siempre a mi lado con su amor, esfuerzo, paciencia, dedicación y apoyo en cada momento de mi vida para llegar a cumplir mis objetivos y metas.

MARÍA FERNANDA TIBANTA LEGÑA

ÍNDICE DE CONTENIDO

DECLARACIÓN.....	II
CERTIFICACIÓN.....	III
AGRADECIMIENTO	IV
AGRADECIMIENTO	V
DEDICATORIA	VI
DEDICATORIA.....	VII
ÍNDICE DE CONTENIDO	VIII
ÍNDICE DE FIGURAS.....	XI
ÍNDICE DE TABLAS.....	XIII
RESUMEN.....	XIV
ABSTRACT	XV
1. INTRODUCCIÓN.....	1
1.1. Planteamiento del problema	1
1.2. Objetivo General.....	2
1.3. Objetivos Específicos	2
1.4. Alcance.....	2
1.5. Estructura del documento	3
2. METODOLOGÍA.....	5
2.1. Metodología SCRUM.....	5
2.1.1. Roles centrales.....	6
2.1.2. Etapas de la Metodología	7
2.2. Diseño de interfaces del sistema web y aplicación móvil	10
2.2.1. Pencil	10
2.3. Diseño de la arquitectura del sistema web y aplicación móvil	11
2.3.1. Arquitectura Modelo-Vista-Controlador (MVC).....	11
2.4. Herramientas de desarrollo.....	13
2.4.1. Herramienta para la Base de Datos.....	13
2.4.2. Herramientas para el desarrollo del sistema web	14
2.4.3. Herramientas para el desarrollo de la aplicación móvil	16
3. RESULTADOS Y DISCUSIÓN	18
3.1. Sprint 0: Configuración del ambiente de desarrollo.....	18
3.1.1. Creación de la Base de Datos en MongoDB.....	18
3.1.2. Creación de modelos.....	19

3.1.3.	Estructura del proyecto.....	20
3.1.4.	Usuarios	22
3.1.5.	Requerimientos específicos para el sistema web y aplicación móvil	23
3.2.	Sprint 1: Inicio de sesión del usuario administrador - módulo fundación	24
3.2.1.	Página de inicio	25
3.2.2.	Inicio de sesión del usuario administrador	25
3.2.3.	Registro, visualización, modificación y eliminación de fundaciones	26
3.2.4.	Notificaciones de solicitudes de fundaciones.....	26
3.2.5.	Visualización, aprobación y/o negación de solicitudes de fundaciones.....	27
3.3.	Sprint 2: Inicio de sesión del usuario administrador de la fundación – módulo de emergencia, adopción, donación, voluntario y animales de compañía	27
3.3.1.	Página de inicio de la fundación	28
3.3.2.	Inicio de sesión del usuario administrador de la fundación en el sistema web	29
3.3.3.	Registro, visualización, modificación y eliminación de información de página de inicio	29
3.3.4.	Registro, visualización, modificación y eliminación de animales de compañía	30
3.3.5.	Registro, visualización, modificación y eliminación de voluntarios	30
3.3.6.	Notificaciones de solicitudes.....	31
3.3.7.	Visualización, aprobación y/o negación de solicitudes de emergencia	31
3.3.8.	Visualización, aprobación y/o negación de solicitudes de adopción.....	32
3.3.9.	Registro, visualización, aprobación y/o negación de solicitudes de donación	32
3.3.10.	Visualización y modificación de la información personal del usuario	34
3.4.	Sprint 3: Registro e inicio de sesión del usuario ciudadano - módulo fundación, formulario, animales de compañía y notificaciones.....	34
3.4.1.	Página de inicio, registro e inicio de sesión del usuario ciudadano	35
3.4.2.	Visualización de animales de compañía	36
3.4.4.	Envío de formularios de adopción, emergencia y donación	37
3.4.5.	Recibir notificaciones de aprobación o negación de solicitudes de adopción, emergencia y/o donación.....	38
3.4.6.	Visualización y modificación de la información personal del usuario.....	38
3.5.	Sprint 4: Inicio de sesión del usuario voluntario – módulo emergencia y solicitudes	39
3.5.1.	Página de inicio e inicio de sesión del usuario voluntario.....	39
3.5.2.	Enviar formulario de emergencia	40

3.5.3.	Visualización, aprobación y/o negación de solicitudes de emergencias y donaciones	40
3.5.4.	Visualización y modificación de la información personal del usuario.....	41
3.6.	Sprint 5: Pruebas en el sistema web y aplicación móvil	41
3.6.1.	Pruebas de carga	41
3.6.2.	Pruebas de compatibilidad.....	44
3.6.3.	Pruebas funcionales	45
3.7.	Sprint 6: Despliegue sistema web y aplicación móvil	47
3.7.1.	Despliegue del sistema web en Heroku	47
3.7.2.	Despliegue de la aplicación móvil en Play Store.....	48
4.	CONCLUSIONES Y RECOMENDACIONES	50
4.1.	Conclusiones	50
4.2.	Recomendaciones	51
5.	BIBLIOGRAFÍA.....	52
6.	ANEXOS.....	55

ÍNDICE DE FIGURAS

Fig. 1: Prototipo de inicio de sesión en el sistema web	10
Fig. 2: Prototipo de inicio de sesión en la aplicación móvil	11
Fig. 3: Arquitectura del sistema web	12
Fig. 4: Arquitectura de la aplicación móvil	13
Fig. 5: Estructura de la Base de Datos en MongoDB	19
Fig. 6: Creación de modelos para el sistema web y aplicación móvil	19
Fig. 7: Estructura del proyecto del sistema web (front-end).....	21
Fig. 8: Estructura del proyecto del sistema web (back-end)	21
Fig. 9: Estructura del proyecto de la aplicación móvil.....	21
Fig. 10: Usuarios del sistema web	22
Fig. 11: Usuarios de la aplicación móvil	22
Fig. 12: Página de inicio.....	25
Fig. 13: Inicio de sesión del usuario administrador.....	26
Fig. 14: Formulario para el registro de una fundación	26
Fig. 15: Visualización de notificaciones.....	27
Fig. 16: Pantalla para visualizar las solicitudes de las fundaciones.....	27
Fig. 17: Página de inicio de la fundación.....	28
Fig. 18: Inicio de sesión del usuario administrador de la fundación.....	29
Fig. 19: Formulario para subir fotos de portada a la página de inicio de la fundación	29
Fig. 20: Formulario para el registro de un animal de compañía.....	30
Fig. 21: Formulario para el registro de un voluntario	31
Fig. 22: Visualización de notificaciones de adopciones, emergencias y donaciones	31
Fig. 23: Pantalla para visualizar las solicitudes de emergencia.....	32
Fig. 24: Pantalla para visualizar las solicitudes de adopciones	32
Fig. 25: Formulario para el registro de una donación económica	33
Fig. 26: Pantalla para visualizar las solicitudes de donaciones	33
Fig. 27: Pantalla con opciones para el usuario administrador de la fundación	34
Fig. 28: Formulario para modificar los datos personales del usuario administrador y de la fundación	34
Fig. 29: Página de inicio.....	35
Fig. 30: Formulario de registro de un ciudadano	35
Fig. 31: Pantalla para visualizar los animales de compañía	36
Fig. 32: Pantalla para visualizar las fundaciones.....	37
Fig. 33: Formulario para solicitar una adopción.....	37
Fig. 34: Notificaciones para la visualización de solicitudes aprobadas o negadas	38
Fig. 35: Pantalla con opciones para el usuario ciudadano	38
Fig. 36: Página de Inicio	39
Fig. 37: Formulario para solicitar una emergencia	40
Fig. 38: Formulario para aprobar y/o negar la solicitud de una emergencia	40
Fig. 39: Formulario para aprobar y/o negar la solicitud de una donación	40
Fig. 40: Pantalla con opciones para el usuario ciudadano	41
Fig. 41: Prueba de carga #1 50 peticiones.....	43
Fig. 42: Prueba de carga #2 500 peticiones.....	43
Fig. 43: Prueba de carga #3 1000000 peticiones	44
Fig. 44: Creación de la aplicación en Heroku.....	48

Fig. 45: Despliegue del sistema web bajo un dominio.....	48
Fig. 46: Inicio de sesión en Google Play Console	49
Fig. 47: Notificación de la publicación de la aplicación.....	49
Fig. 48: Publicación de la aplicación móvil	49

ÍNDICE DE TABLAS

TABLA I: Equipo de desarrollo y asignación de roles.....	7
TABLA II: Historia de usuario 1-Inicio de sesión del usuario administrador en el sistema web	8
TABLA III: Librerías para el desarrollo del sistema web	17
TABLA IV: Librerías para el desarrollo de la aplicación móvil	17
TABLA V: Navegadores que soportan el sistema web	45
TABLA VI: Dispositivos móviles en los que se probó la aplicación móvil.....	45
TABLA VII: Prueba de funcionalidad - Inicio de sesión del administrador en el sistema web	46
TABLA VIII: Prueba de funcionalidad - Inicio de sesión del usuario ciudadano en la aplicación móvil	47

RESUMEN

En la ciudad de Quito y sus alrededores existe un gran número de animales de compañía (perros y gatos) abandonados en las calles por diversas circunstancias. Según el Instituto Nacional de Estadísticas y Censos (INEC) las estimaciones de los censos del año 2013 dieron a conocer que existían 41.676 perros callejeros, proyectándose que para el 2018 estas cifras aumentarían alrededor de 122.280; esta situación provocaría varios problemas, entre ellos la sobrepoblación, maltrato, salud, accidentes, entre otros. Por ello, grupos de personas han creado refugios o fundaciones, con la finalidad de ayudar a esta causa, encargándose de recogerlos, protegerlos y darlos en adopción.

En este contexto, el mayor problema que tiene la ciudadanía en general es la falta de información, pues no cuentan con un medio de comunicación estable de todas las fundaciones que existen en la ciudad de Quito, motivo por el cual han optado por el uso de las redes sociales. Provocando una serie de inconvenientes en varias actividades que realizan las fundaciones; un claro ejemplo es en el proceso de adopciones y donaciones; existen retrasos u otros inconvenientes que perjudican que este proceso sea rápido y efectivo.

De igual manera, la falta de un adecuado control sobre los voluntarios que ayudan a las fundaciones y las emergencias reportadas por los ciudadanos, provocan que las mismas no puedan ser atendidas de manera adecuada. Dificultando que cada día, estos animales de compañía logren encontrar un hogar y de esta manera reducir las estadísticas de abandono.

Este trabajo propone el diseño, estructura, desarrollo e implementación de un sistema web y aplicación móvil, aplicando la metodología SCRUM y herramientas de desarrollo libres, permitiendo de esta manera integrar todas las fundaciones que existen en la ciudad de Quito y permitiendo a la ciudadanía en general, obtener información relacionada a las fundaciones, información de voluntarios y reportando casos de emergencia, facilitando el acceso a la información y comunicación por medio de la tecnología.

Palabras clave: fundaciones, animales de compañía, Node, MongoDB, Angular, Ionic.

ABSTRACT

In the Quito city and its surroundings, there is a large number of abandoned pets (dogs and cats) in the streets, due to various circumstances. According to the National Statistics and Census Institute (INEC) the estimates from the censuses of year 2013 they announced that there are 41.676 stray dogs, with which it is projected that for 2018 there will be around 1222.280; which causes several problems, including overpopulation, health issues, accidents, among others. For this reason, groups of people have created shelters or foundations to help this cause, taking care of collecting, protecting and giving them up for adoption.

In this context, the biggest problem that citizens in general have is the lack of information, that they do not have stable communication with all the foundations that exist in the Quito city, reason for which, they have opted for the use of social networks. Causing a series of inconveniences with a lot of activities they carry out by foundations; a clear example is in the process of adoptions and donations; there are delays or other inconveniences that hinder this process to be fast and effective.

Likewise, the lack of adequate control over the volunteers that help the foundations and emergencies reported by the citizens, cause that they cannot be adequate care. Making it difficult every day, these pets to find a home and this way reduce the abandonment statistics.

This work proposes the design, structure, development and implementation of a web system and mobile application, applying the SCRUM methodology and free development tools, allowing in this way to integrate all the foundations that exist in the Quito city and allowing citizens in general, obtain information related to foundations, information from volunteers and reporting emergencies cases, facilitating access to information and communication through technology.

Keywords: foundations, pets, Node, MongoDB, Angular, Ionic

1. INTRODUCCIÓN

En esta sección, se da una introducción al planteamiento del problema, se definen los objetivos del proyecto, su alcance y la estructura que tendrá el presente informe.

1.1. Planteamiento del problema

Según el Instituto Nacional de Estadísticas y Censos (INEC), en la ciudad de Quito y sus alrededores, las estimaciones de los censos del año 2013 dieron a conocer que existían 41.676 perros callejeros, con lo cual se proyecta que para el 2018 existirían alrededor de 122.280 perros; lo que indica que el abandono de las mascotas va en aumento hasta casi triplicarse [1].

Es por ello que, actualmente se han unido varios grupos de personas con el objetivo de crear varias fundaciones sin fines de lucro, con la única finalidad de ayudar en esta causa. Sin embargo, muchas de estas fundaciones no se encuentran calificadas por el Municipio de Quito para ejercer dichas actividades y al no contar con un medio de comunicación como una página web y/o aplicación móvil, utilizan las redes sociales para darse a conocer y ser un medio de comunicación entre fundaciones, voluntarios y ciudadanía.

En la ciudad de Quito, existen varias fundaciones como: Lucky Bienestar Animal, Acción Animal Ecuador, Catman and Doging [2], Protección Animal Ecuador (PAE), Urbanimal [3] entre otras.

A pesar de ello, la ciudadanía no cuenta con un catálogo de todas las fundaciones existentes en la ciudad de Quito lo que conlleva a un sin número de inconvenientes, como: retraso en el proceso de adopción y donaciones, falta de información sobre voluntarios y el personal a cargo de las fundaciones, demora en la atención de emergencias, etc.

Para dar solución a esta problemática, se propone el desarrollo de un sistema web que permita integrar y gestionar todas las fundaciones que se encuentran dentro de la ciudad de Quito, posibilitando que cada fundación gestione el proceso de adopción, donación y atención de emergencias.

La aplicación móvil permitirá a la ciudadanía en general obtener información de las fundaciones, voluntarios, animales de compañía que se encuentren listos para ser adoptados, formularios de adopciones, notificaciones, entre otros.

El proyecto integrador “Desarrollo de un sistema web y una aplicación móvil para la gestión de fundaciones de animales de compañía en la ciudad de Quito” puede apoyar a las fundaciones y ciudadanía mejorando la organización y comunicación por medio de la tecnología.

1.2. Objetivo General

Desarrollar un sistema web y aplicación móvil para la gestión de fundaciones de animales de compañía en la ciudad de Quito.

1.3. Objetivos Específicos

- Determinar los requerimientos del sistema web y la aplicación móvil.
- Diseñar la arquitectura web, modelo de base de datos y los mockups del sistema web y la aplicación móvil.
- Codificar el sistema web y la aplicación móvil.
- Probar el funcionamiento del sistema web y la aplicación móvil.

1.4. Alcance

En este proyecto se ha implementado un sistema web y aplicación móvil para brindar una solución e información sobre los procesos de adopción, donaciones y emergencias de las diferentes fundaciones, beneficiando a los animales callejeros para que logren encontrar un hogar y de esta manera reducir las estadísticas de abandono.

En una era tecnológica en constante evolución, las aplicaciones móviles y sistemas web se han convertido en una parte fundamental del día a día de las personas.

Es por ello que, al desarrollar este proyecto se ayudará a que las personas puedan involucrarse en las actividades que realizan las fundaciones por el bienestar de los animales callejeros.

La propuesta garantiza la integridad, consistencia y seguridad de los datos, tanto de la aplicación móvil como del sistema web, utilizando un sistema de autenticación y varios perfiles descritos de la siguiente manera:

El perfil administrador permite:

- ✓ Registro, actualización y eliminación de fundaciones.
- ✓ Administración y control del sistema

El perfil fundaciones permite:

- ✓ Registrar información relevante a la fundación.
- ✓ Registro, actualización y eliminación de animales de compañía.
- ✓ Registro, actualización y eliminación de voluntarios.
- ✓ Registro de donaciones económicas y de productos.
- ✓ Aprobar o negar la petición de una adopción por parte de los ciudadanos.
- ✓ Aprobar o negar las emergencias por parte del ciudadano y del voluntario.

Mientras que la aplicación móvil está desarrollada para dispositivos que cuenten con Sistema Operativo Android e iOS y realiza lo siguiente:

Por parte de los ciudadanos:

- ✓ Registro por medio de Facebook y un formulario de registro.
- ✓ Visualización de todas las fundaciones registradas en el sistema web.
- ✓ Enviar el formulario para solicitar la adopción de un animal de compañía.
- ✓ Enviar el formulario para solicitar la emergencia de un animal de compañía.
- ✓ Enviar el formulario para realizar una donación.
- ✓ Recibir una notificación de aprobación de la adopción.
- ✓ Recibir una notificación de aprobación o negación de la emergencia.
- ✓ Recibir una notificación de aprobación o negación de la donación.

Por parte de los voluntarios:

- ✓ Enviar el formulario para solicitar la emergencia de un animal de compañía.
- ✓ Aprobar o negar la petición para atender la emergencia.
- ✓ Aprobar o negar la petición para retirar la donación.

1.5. Estructura del documento

El proceso de implementación del sistema web y aplicación móvil se encuentra dividido en cuatro secciones detallados de la siguiente manera:

En la sección I (Introducción) se describe el planteamiento del problema, objetivo general, objetivos específicos y el alcance del proyecto.

En la sección II (Metodología) se detalla la metodología ágil de desarrollo Scrum y su implementación en el proyecto. Adicional a ello, se especifica la Recopilación de

Requerimientos, Historias de Usuarios, Product Backlog y las herramientas de desarrollo utilizadas para la codificación y despliegue del sistema web y aplicación móvil.

En la sección III (Resultados y Discusión) se especifica la implementación de cada Sprint definido y los resultados obtenidos al finalizar cada etapa de desarrollo. No obstante, se incluye los resultados obtenidos al realizar las diferentes pruebas de cada Sprint.

Finalmente, la sección IV incluye las conclusiones y recomendaciones obtenidas durante la realización de este proyecto integrador.

2. METODOLOGÍA

El desarrollo de software conlleva una serie de etapas consecutivas como: planificación, diseño, implementación y pruebas.

En la primera etapa se llevó a cabo varias actividades como el análisis y levantamiento de requerimientos, siendo está la fase inicial y primordial para el desarrollo. Abasteciendo de ideas principales que ayuden a identificar las necesidades que se desea cubrir al desarrollar el software, garantizando de esta manera el éxito o el fracaso del proyecto.

Para asegurar el éxito en todo el desarrollo del sistema web y aplicación móvil se utilizó la metodología ágil SCRUM, siendo la más adecuada al momento de obtener y analizar los requerimientos, que se los realizó por medio de la elaboración de Historias de Usuario, Product Backlog, Sprint Backlog y la entrega de pequeños avances funcionales al finalizar cada uno de los Sprints planificados.

Por último, se llevaron a cabo una serie de reuniones con varias fundaciones, con la finalidad de obtener información sobre los procesos de la fundación, para definir adecuadamente el modelo de base de datos, herramientas de desarrollo, módulos y funcionalidades que se desarrollaron en el sistema web y aplicación móvil.

2.1. Metodología SCRUM

La metodología de desarrollo ágil SCRUM, fue implementada en el desarrollo de este proyecto; integrando un conjunto de buenas prácticas, técnicas y procesos que ayudaron a obtener mejores resultados [4].

Scrum, es una de las metodologías ágiles más populares caracterizada por ser iterativa, rápida, flexible, eficaz y diseñada para ofrecer un valor significativo en todo el desarrollo del proyecto, garantizando transparencia en la comunicación y creando un ambiente de responsabilidad colectivo y de proceso continuo [5].

Una de las fortalezas claves de Scrum, es el uso de equipos multifuncionales auto organizados, lo que permitió al pequeño equipo de este proyecto, la división del trabajo en ciclos cortos y concentrados llamados Sprints: permitiendo la entrega de pequeños avances al finalizar cada uno de ellos, satisfaciendo de esta manera las necesidades de la fundación que proporciono el conocimiento de sus procesos [5].

2.1.1. Roles centrales

Los roles centrales son requeridos en el desarrollo de un producto, definiendo las obligaciones que cada integrante debe cumplir para asegurar que la información entregada sea la correcta y que exista una comunicación constante con el cliente [5]. Aplicando SCRUM se determinaron los siguientes roles:

Propietario del Producto (Product Owner)

En el presente escrito, este rol está representado por la fundación Paws and Claws, la cual, por medio de reuniones, fue la encargada de proporcionar los requerimientos y lógica sobre los procesos de adopciones, voluntarios y donaciones.

Scrum Master

Está representado por el director del proyecto integrador quién guío, ayudó y enseñó las buenas prácticas de SCRUM, eliminando cualquier inconveniente que afecte al equipo en el cumplimiento de las metas establecidas.

Equipo de desarrollo (Development Team)

Fue conformado por dos desarrolladores con conocimientos técnicos necesarios para organizarse, resolver los requerimientos entregados por el Product Owner y tomar decisiones acertadas con la finalidad de conseguir el objetivo del proyecto [6].

Para el desarrollo de este proyecto integrador, se conformó un equipo sumamente pequeño de trabajo, descrito en la TABLA I.

TABLA I: Equipo de desarrollo y asignación de roles

NOMBRE	ROL
Paws and Claws y Acción Animal	Product Owner
Ing. Byron Loarte	Scrum Master
David Hernández María Fernanda Tibanta	Equipo de desarrollo y testeo

2.1.2. Etapas de la Metodología

Las etapas de SCRUM, permitieron que los roles indicados en la sección anterior puedan realizar adecuadamente las tareas definidas y proporcionando mayor transparencia de información [7].

El listado que se presenta a continuación corresponde a las etapas, necesarias que requiere la metodología.

Recopilación de requerimientos

Se define como el proceso de identificar las necesidades del negocio, solucionando posibles discrepancias y efectuar mejoras. Un papel fundamental en el proceso de desarrollo de software, definiendo especificaciones correctas de forma consistente y compacta [8].

Mediante una serie de reuniones mantenidas con fundaciones como: Paws and Claws y Acción Animal, se definió una lista de requerimientos de forma general que se solventarán con el desarrollo del sistema web y aplicación móvil. Las cuales se encuentran detalladas en el Manual Técnico – Sección Recopilación de Requerimientos.

Historias de Usuario

Son tarjetas con una breve descripción de los requerimientos de un cliente, ayudando a descubrir qué se espera como salida de la implementación y cómo se ve beneficiado el usuario final.

Las Historias de Usuario deben estar expresadas en lenguaje natural, evitando usar lenguaje técnico para un mejor entendimiento [9].

Las Historias de Usuario sirvieron para describir la funcionalidad que va a tener el sistema web y aplicación móvil, por lo cual deben tener una estructura según la metodología SCRUM. A continuación, en la TABLA II se presenta un ejemplo de las Historias de Usuario generadas en el presente proyecto.

Por último, las 27 Historias de Usuario se encuentran descritas en el Manual Técnico – Sección Historias de Usuario.

TABLA II: Historia de usuario 1-Inicio de sesión del usuario administrador en el sistema web

HISTORIA DE USUARIO	
Identificador (ID): HU001	Usuario: Administrador
Nombre Historia: Inicio de sesión del usuario administrador en el sistema web	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Iteración Asignada: 1	
Responsable (es): David Hernández - María Fernanda Tibanta	
Descripción: El usuario administrador, tiene la posibilidad de gestionar las fundaciones, por lo cual es necesario que se identifique en el sistema web con las credenciales correspondientes como son usuario y contraseña. El usuario administrador tiene la opción de cerrar sesión cuando sea necesario.	
Observación: El sistema web es gestionado por un único usuario administrador, por ende, las credenciales son proporcionadas por el equipo de desarrollo. El correo proporcionado será: admin@admin.com y la contraseña contendrá entre 8 a 20 caracteres, en la cual se debe incluir al menos 1 letra mayúscula, 1 letra minúscula, 1 número y un carácter no alfanumérico.	

Product Backlog

Es una lista ordenada en la cual se prioriza todas las funcionalidades o requisitos en el desarrollo de un producto o software [6]. No obstante, para ordenar esta lista con cada uno de los requerimientos es necesario basarse en el valor, riesgos, dependencias y necesidades de la organización [7].

El Product Backlog, permitió listar los requerimientos funcionales que va a tener el sistema web y aplicación móvil, ordenándolos en base a prioridades para el desarrollo y un valor desde el punto de vista de la organización, como se presenta en el Manual Técnico – Sección Product Backlog, mismos que irán añadiendo valor a los productos finales mediante iteraciones sucesivas.

Sprints de desarrollo

Son un conjunto de actividades de desarrollo que tiene un periodo predefinido, con una duración máxima de 4 semanas, esto permitió que el proyecto se pueda descomponer en unidades más pequeñas y así determinar qué actividades son las que no están avanzando e intentar eliminar el problema para cumplir con el objetivo del Sprint [6].

En base a la Recopilación de Requerimientos, Historias de Usuario y Product Backlog generadas anteriormente, se procedió con la creación de seis Sprints de desarrollo: Configuración del ambiente de desarrollo, Inicio de sesión del usuario administrador, Inicio de sesión del usuario administrador de la fundación, Registro e inicio de sesión del usuario ciudadano, Inicio de sesión del usuario voluntario, Pruebas en el sistema web y aplicación móvil y Despliegue sistema web y aplicación móvil; detallados en el Manual Técnico – Sección Sprint Backlog.

Por último, para ejecutar cada uno de los Sprints fue necesario realizar una planificación con el objetivo de tomar decisiones acertadas en el desarrollo de los sistemas propuestos.

2.2. Diseño de interfaces del sistema web y aplicación móvil

Formalizados los requerimientos funcionales, se procede a describir la herramienta utilizada para la creación de las interfaces del sistema web y aplicación móvil, así como los beneficios que aportó en el desarrollo de este proyecto.

2.2.1. Pencil

Es una herramienta para la creación de interfaces de usuario que permite organizar contenidos y jerarquizar diferentes elementos en base a las funcionalidades del proyecto, con el objetivo de mostrar al cliente cual sería el resultado final de la implementación [10].

Se utilizó la herramienta Pencil, para el diseño de las interfaces que conforman el sistema web y aplicación móvil. Con el objetivo de crear un diseño minimalista, realista y sin tanto código.

Fig. 1 y 2, muestran las principales interfaces del sistema web y aplicación móvil, mientras que el diseño de las 30 interfaces se ilustra en el Manual Técnico – Sección Diseño de Interfaces.

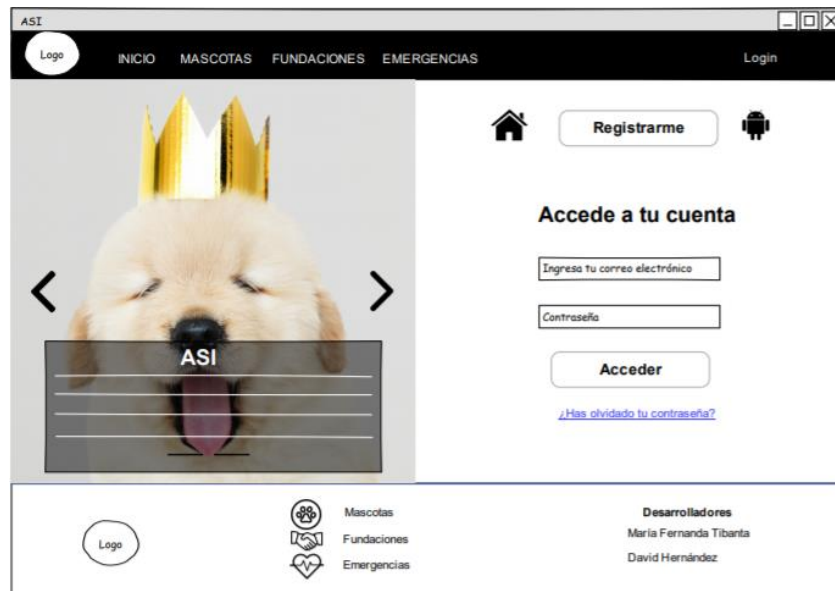


Fig. 1: Prototipo de inicio de sesión en el sistema web



Fig. 2: Prototipo de inicio de sesión en la aplicación móvil

2.3. Diseño de la arquitectura del sistema web y aplicación móvil

Formalizados los requerimientos funcionales, se procede a definir el patrón arquitectónico utilizado para el desarrollo del sistema web y aplicación móvil, así como los beneficios que aportó en el desarrollo del proyecto.

2.3.1. Arquitectura Modelo-Vista-Controlador (MVC)

Es un patrón de arquitectura de software que separa los datos de una aplicación, la lógica del negocio y la interfaz de usuario, el cual se representa a través de tres componentes: Modelo, Vista y Controlador [11].

Este patrón permitió modificar cada uno de los componentes sin la necesidad de afectar a los demás, garantizando de esta manera la actualización y el mantenimiento del software.

La arquitectura basada en estos tres componentes, permitió tener el código ordenado y separar funcionalidades entre las capas, ayudando a ahorrar tiempo en la detección y corrección de errores, estructurando un sistema web y aplicación móvil de forma robusta.

Por lo mencionado anteriormente y luego de haber analizado los beneficios del patrón de arquitectura. Las Fig. 3 y 4, ilustran el detalle de la arquitectura implementada en base a las herramientas a ser utilizadas, para el sistema web y la aplicación móvil respectivamente.

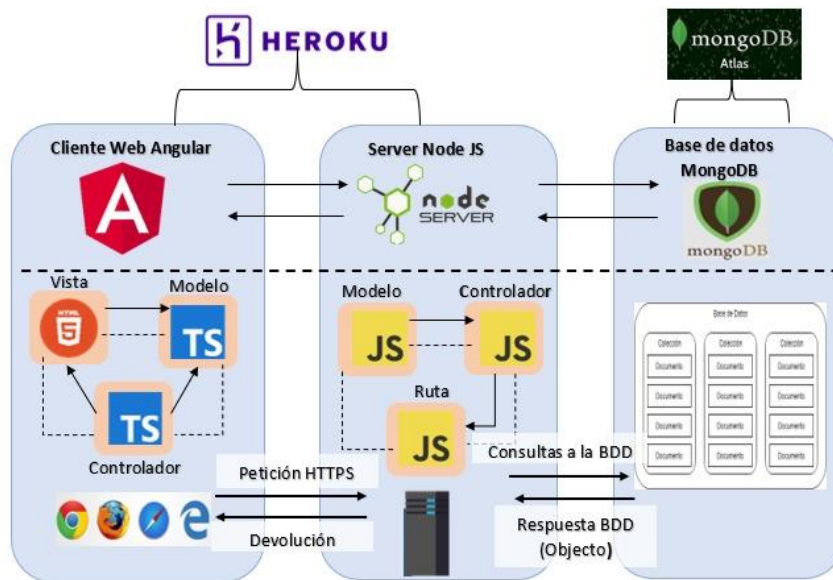


Fig. 3: Arquitectura del sistema web

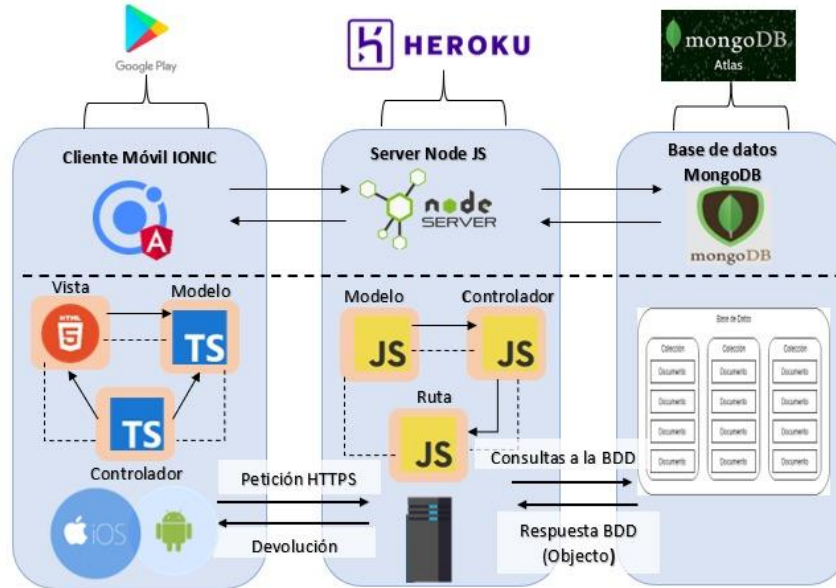


Fig. 4: Arquitectura de la aplicación móvil

2.4. Herramientas de desarrollo

Formalizados los requerimientos funcionales y definido el patrón arquitectónico, se procede a definir las herramientas utilizadas para el desarrollo del sistema web, aplicación móvil y base de datos, así como los beneficios que cada uno aportó en el desarrollo del proyecto.

2.4.1. Herramienta para la Base de Datos

MongoDB

Es un Sistema Gestor de Base de Datos NoSQL¹ multiplataforma de licenciamiento libre y orientado a documentos, es decir, los atributos no se repiten entre los diferentes registros [12]. NoSQL, permite la escalabilidad horizontal de datos, desarrollo sencillo y el almacenamiento datos de mayor magnitud.

¹ **NoSQL**: Es un sistema de gestión de datos caracterizado por no tener relaciones entre entidades o tablas existentes en los sistemas de gestión de Base de Datos relacionadas. Obtenido de: <https://blogs.oracle.com/spain/qu-es-una-base-de-datos-nosql>

El modelo de documento de datos que se utilizó fue JSON²/BSON³, el cual es fácil de programar y manejar, ofreciendo alto rendimiento por medio de la agrupación de datos relevantes entre sí [12].

La utilización de MongoDB permitió realizar una adecuada configuración, supervisión y administración de la información en la Base de Datos. Al ser una base de datos no relacional ofrece una mejor escalabilidad, rendimiento, disponibilidad y de fácil integración con las herramientas determinadas para el desarrollo, con el objetivo de que la comunicación con el cliente sea más estable y en menor tiempo.

Por último, se usó la librería Mongoose, para abstraer los documentos de la base de datos y colocarlos dentro de la lógica del negocio como clases. Además, posee un sistema de validación de datos acorde a las necesidades del proyecto [13].

2.4.2. Herramientas para el desarrollo del sistema web

Angular

Es un Framework⁴ moderno para el desarrollo de aplicaciones web basadas en el patrón de arquitectura MVC. Permitiendo una comunicación sencilla y optimizada con cualquier back-end por medio de un sin número de librerías. [14]

En el desarrollo del front-end del sistema web, Angular permitió que el desarrollo de las páginas web sea de forma dinámica, garantizando que la aplicación sea más ligera y rápida. Además, permitió trabajar con varios componentes obteniendo una serie de ventajas.

- **HTML:** permitió definir la estructura en cada una de las vistas y organizar la información y el contenido por medio de etiquetas.
- **CSS:** permitió definir el estilo visual en cada una de las vistas con el objetivo de crear diseños adaptables a cualquier dispositivo y tamaño de pantalla.
- **TypeScript:** permitió definir la lógica, efectos de animación y comunicación con servicios y otros componentes del sistema web y aplicación móvil.

² **JSON:** JavaScript Object Notation, es un documento para almacenar registros. Obtenido de: <https://www.mongodb.com/json-and-bson?lang=es-es>

³ **BSON:** Formato codificado en binario del modelo JSON para proporcionar datos adicionales y campos ordenados. Obtenido de: <https://www.mongodb.com/json-and-bson?lang=es-es>

⁴ **Framework:** Esquema o estructura que se establece y permite desarrollar y organizar un software. Obtenido de: <https://neoattack.com/neowiki/framework/>

Angular Material

En el desarrollo del sistema web, la librería de Angular Material permitió utilizar una gama de componentes tipo Bootstrap, es decir crear vistas llamativas y aligerando el proceso de desarrollo, ya que algunos componentes están desarrollados previamente.

NodeJS

Es un entorno de JavaScript del lado del servidor, el cual utiliza un modelo sincrónico/asincrónico controlado por eventos, lo que hace que sea ligero y eficiente, permitiendo la ejecución de cualquier operación, desde leer o escribir archivos de cualquier tipo, hasta hacer solicitudes HTTP [15].

El desarrollo del back-end del sistema web NodeJS, permitió realizar una adecuada configuración y el desarrollo del servidor web mediante el lenguaje de programación JavaScript. Además, permitió construir un back-end con un buen rendimiento y escalabilidad gracias a que utiliza el modelo de entrada y salida sin bloqueo.

Express.js

Es un Framework de desarrollo de aplicaciones web, permitiendo flexibilidad para NodeJS, caracterizado por ser robusto, rápido, flexible y simple [16].

El desarrollo del back-end del sistema web ExpressJS, permitió la creación de un servidor web y minimizar el tiempo de desarrollo en enrutamientos, middlewares, sesiones, cookies entre otros.

NodeMailer

NodeMailer es un módulo para NodeJS el cual permite enviar correos electrónicos de manera sencilla. Soportando HTML, texto plano, imágenes adjuntas o incrustadas en el contenido [17].

El desarrollo del back-end del sistema web NodeMailer, permitió el envío de correos electrónicos en las diferentes acciones como: aprobación de cuentas, recuperación de contraseñas, verificación de correos, accesos al sistema, entre otros.

OneSignal

OneSignal es una plataforma para envío de notificaciones Push a través de diferentes plataformas, siendo las más utilizada en el desarrollo actual [18].

El desarrollo del back-end del sistema web y aplicación móvil OneSignal, permitió el envío de notificaciones Push⁵ para informar acerca de: aprobación/negación de emergencias, adopciones, donaciones, entre otros.

JWT (JSON Web Token)

Es un estándar abierto basado en JSON, el cual permite crear un token que sirva para enviar datos entre aplicaciones o servicios y de esta manera garantizar que sean válidos y seguros. [19]

El desarrollo del back-end del sistema web JWT, permitió la creación de tokens para hacer posible la verificación de identidad, roles y manejar las sesiones por parte del sistema web y aplicación móvil.

2.4.3. Herramientas para el desarrollo de la aplicación móvil

Ionic

Es uno de varios Frameworks gratuitos que utilizan los programadores para desarrollar apps basadas en HTML, CSS y JavaScript [20]. Permitiendo el desarrollo de aplicaciones híbridas para móviles, tablets y en ocasiones aplicaciones web [21].

El equipo de desarrollo eligió desarrollar la aplicación móvil con este Framework, debido a la experiencia que tienen, permitiendo trabajar de forma óptima, reutilizando código y aprovechando de mejor manera las capacidades que este posee.

Un claro ejemplo son los componentes emulados para cualquier Sistema Operativo móvil, permitiendo construir una aplicación híbrida multiplataforma desde una única base de código [22].

Apache Cordova

Es un Framework de desarrollo móvil de código abierto, el cual dispone API's de diversos dispositivos móviles para desarrollar aplicaciones multiplataforma, utilizando tecnologías web HTML, CSS y JavaScript [23].

⁵ **Notificaciones Push:** son mensajes que se envían desde un servidor remoto hasta los dispositivos que tienen instalada la aplicación. Obtenido de: <https://www.websa100.com/blog/notificaciones-push-que-son-por-que-usarlas/>

El desarrollo de la aplicación móvil Apache Cordova, permitió usar contenido HTML, CSS, TypeScript dentro de la aplicación móvil, de esta forma se hizo posible el desarrollo de una aplicación híbrida para diferentes plataformas móviles.

Librerías

A continuación, las TABLAS III y IV, detallan las librerías implementadas en el desarrollo del sistema web y aplicación móvil, respectivamente, mostrando una pequeña descripción de cada una de ellas.

TABLA III: Librerías para el desarrollo del sistema web

LIBRERÍA	DESCRIPCIÓN
"angular2-moment": "^1.9.0"	Librería para transformar fechas.
"bootstrap": "^4.2.1"	Librería de componentes estilo de Bootstrap.
"font-awesome": "^4.7.0"	Librería de iconos.
"hammerjs": "^2.0.8"	Librería de gestos de usuario.
"jquery": "^3.3.1"	Librería de componentes de jQuery, permitió agregar interactividad y efectos visuales.
"@ionic-native/onesignal": "^5.18.0"	Librería para usar notificaciones web y móvil.

TABLA IV: Librerías para el desarrollo de la aplicación móvil

LIBRERÍA	DESCRIPCIÓN
swiper: ^4.5.1	Librería para componente tipo slides, deslizador táctil móvil.
"@ionic-native/facebook": "^5.12.0"	Librería para login con Facebook en la aplicación móvil.

3. RESULTADOS Y DISCUSIÓN

En esta sección, se detalla la implementación y resultados obtenidos de los seis Sprints de desarrollo, pruebas de carga, compatibilidad, funcionalidad y el proceso de despliegue del sistema web y aplicación móvil.

3.1. Sprint 0: Configuración del ambiente de desarrollo

De acuerdo a la planificación realizada en el Sprint Backlog descrito en el Manual Técnico – Sección Sprint Backlog. El Sprint 0 contiene actividades relacionadas para dejar listo el entorno de desarrollo y trabajar en el Product Backlog desde un enfoque ágil. A continuación, se presentan las tareas definidas dentro de este Sprint.

- Creación de la Base de Datos en MongoDB
- Creación de modelos
- Estructura del proyecto
- Usuarios
- Requerimientos específicos para el sistema web y aplicación móvil

3.1.1. Creación de la Base de Datos en MongoDB

En base a los requerimientos funcionales para el sistema web y aplicación móvil, el almacenamiento de la información de fundaciones, voluntarios, ciudadanos y animales de compañía, se utilizó el Sistema Gestor de Base de Datos MongoDB, tomando en cuenta el tipo de dato que iba a ser almacenado.

Fig. 5, ilustra la estructura de la Base de Datos NoSQL, la cual es orientada a documentos con el esquema de un JSON como almacenamiento, realizando lo siguiente:

- Creación de la Base de datos llamado sistemaFundaciones.
- Creación de 9 colecciones.

Po último, el diseño de la Base de Datos se ilustra en el Manual Técnico – Sección Diseño de la Base de Datos.



Fig. 5: Estructura de la Base de Datos en MongoDB

3.1.2. Creación de modelos

Un modelo o Schema es la estructura que tendrá una colección en la Base de Datos NoSQL con el respectivo tipo de dato y validación [24].

Para el desarrollo de este proyecto se crearon 9 modelos, como ilustra Fig. 6, con el objetivo de asignar perfiles y que cada uno de ellos puedan realizar acciones como: inserción, actualización, eliminación y presentación de la información en el sistema web y aplicación móvil.

```
var msc= await Mascota.findOne({'_id':mascotaId});
var adp = await Adopcion.findOne({'_id':adopcionId});
var updateADP = await Adopcion.findByIdAndUpdate(adopcionId,{estado:1,observaciones:desc},{new:true})
var updateMSC = await Mascota.findByIdAndUpdate(mascotaId,{estado:2},{new:true})
```

Fig. 6: Creación de modelos para el sistema web y aplicación móvil

3.1.3. Estructura del proyecto

Para la creación del proyecto en Visual Studio Code⁶, se lo realizó tomando en cuenta el patrón arquitectónico (MVC), creando en cada uno de los componentes las clases necesarias para la interacción y comunicación en el desarrollo de los sistemas propuestos.

Cabe destacar que cada una de las capas se las fue creando en un orden en específico, con el objetivo de que el equipo de desarrollado se encargue de crear las clases que sean necesarias para un fácil mantenimiento a futuro.

Fig. 7, 8 y 9, ilustran la estructura del proyecto del sistema web (back-end y front-end) y aplicación móvil respectivamente, detallando el contenido de algunas de las carpetas más relevante de la siguiente manera:

- **Carpeta src:** contiene todos los archivos que iniciarán el sistema web y la aplicación móvil
- **Carpeta components:** contiene todas las páginas que se mostrarán por ejemplo el perfil de la fundación, emergencias, donaciones, notificaciones, etc.
- **Carpeta guards:** contiene los métodos que permiten validar el acceso a determinadas páginas.
- **Carpeta models:** contiene todos los modelos a ser utilizados según la estructura de las colecciones de la Base de Datos.
- **Carpeta services:** contiene todos los servicios que permiten conectarse al back-end y realizar los métodos CRUD, que es el acrónimo de Crear, Leer, Actualizar y Eliminar o en inglés Create, Read, Update and Delete.
- **Carpetas tabs:** contiene la navegación entre las páginas de animales de compañía, emergencias, notificaciones y fundaciones.

⁶ **Visual Studio Code:** Es un editor de código fuente sofisticado que admite muchas funcionalidades prácticas al momento de trabajar con el código, de varios lenguajes de programación. Obtenido de: <https://blogs.itpro.es/eduardocloud/2016/08/22/visual-studio-code-que-es-y-que-no-es/>

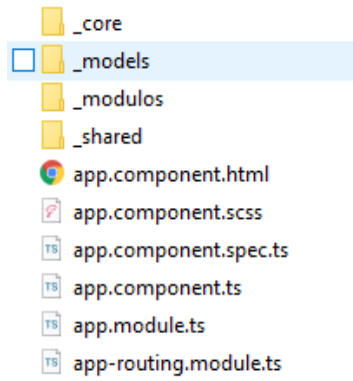


Fig. 7: Estructura del proyecto del sistema web (front-end)

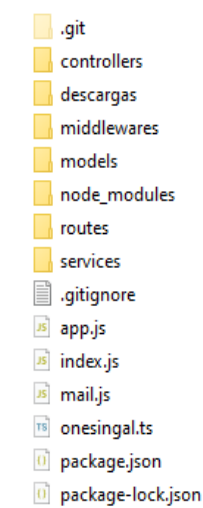


Fig. 8: Estructura del proyecto del sistema web (back-end)

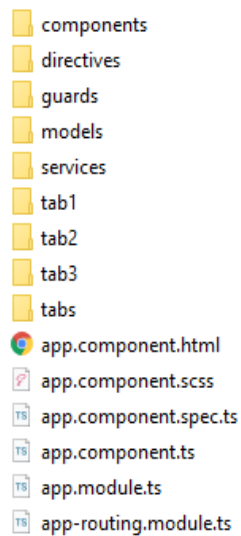


Fig. 9: Estructura del proyecto de la aplicación móvil

3.1.4. Usuarios

Fig. 10 y 11, ilustran los servicios que el sistema web y la aplicación móvil ofrecen a los usuarios respectivamente, y la relación que existe entre los mismos:

- En la sección de color naranja se visualiza a los usuarios: Administrador, Fundación, Ciudadano y Voluntario, los cuales interactúan con el sistema web y aplicación móvil de acuerdo a roles y permisos otorgados.
- En la sección de color amarillo se visualiza los servicios que proporciona el sistema web y la aplicación móvil de acuerdo a roles y permisos otorgados, por ejemplo, el usuario administrador puede gestionar fundaciones, es decir, registrar, visualizar, modificar y eliminar fundaciones.

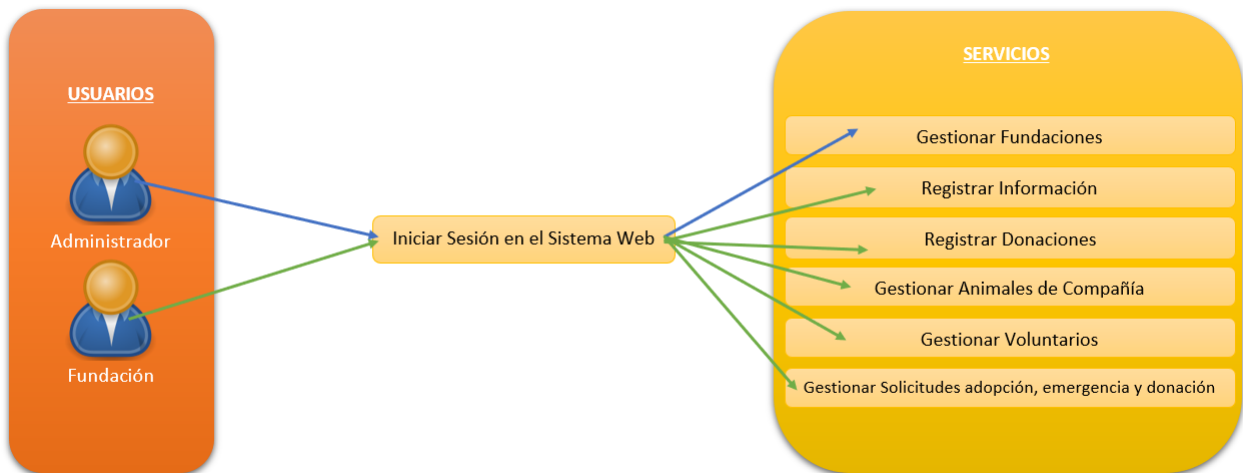


Fig. 10: Usuarios del sistema web

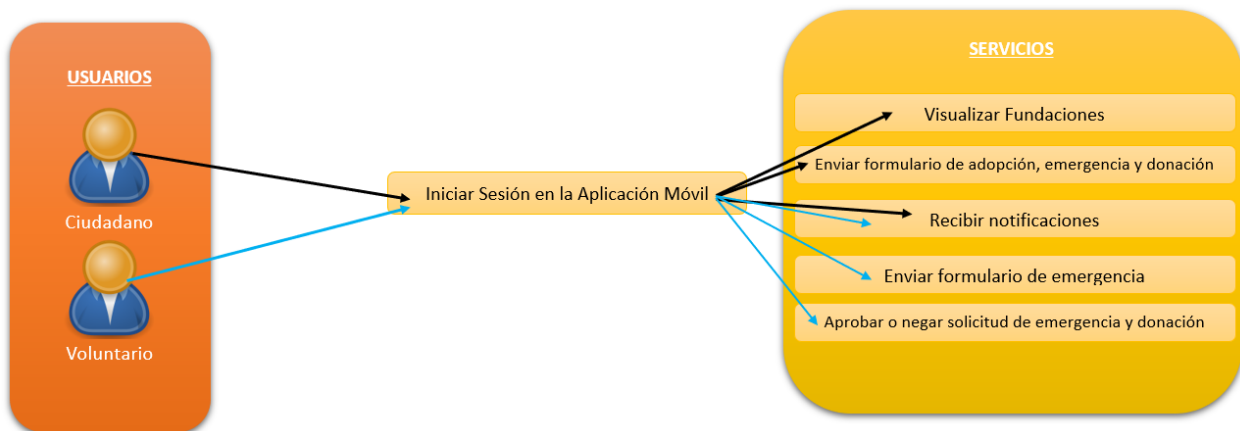


Fig. 11: Usuarios de la aplicación móvil

3.1.5. Requerimientos específicos para el sistema web y aplicación móvil

Autenticación de usuarios

Los usuarios pueden iniciar sesión en el sistema web y aplicación móvil con sus credenciales:

- **Email:** correo electrónico
- **Contraseña:** contiene entre 8-20 caracteres, al menos 1 letra mayúscula, 1 letra minúscula, 1 número y un carácter no alfanumérico.

El correo electrónico registrado para los perfiles de fundación, voluntario y ciudadano no podrá ser utilizado nuevamente en la creación de una nueva cuenta con otro perfil.

En el sistema web, únicamente pueden iniciar sesión el súper administrador y el administrador de cada fundación. El equipo de desarrollo proporcionará las credenciales al súper administrador para que pueda iniciar sesión en el sistema web.

Los usuarios administradores de las fundaciones deben usar las credenciales que hayan proporcionado a momento de registrarse, al igual que los ciudadanos y voluntarios al momento de iniciar sesión en aplicación móvil.

Los usuarios ciudadanos pueden registrarse e iniciar sesión en la aplicación móvil, por medio de la red social Facebook.

El usuario administrador de la fundación, tiene la posibilidad de acercarse directamente con el súper administrador, para realizar el proceso de registro y posterior a ello se envía un correo electrónico con un mensaje de bienvenida con las credenciales para el acceso al sistema web. No obstante, la contraseña será una temporal para que luego pueda cambiarla a una más segura.

El usuario administrador de la fundación, tiene la posibilidad de registrarse en el sistema web llenando un formulario, posterior a ello se envía un correo con un código de verificación, el cual debe ser ingresado en el campo respectivo.

Cabe destacar que la cuenta será habilitada cuando el súper administrador en un lapso de 24 a 48 horas apruebe dicha solicitud, en caso que la solicitud sea negada se enviarán las respectivas observaciones.

Cada fundación registrará a sus voluntarios, quienes tengan sus credenciales correspondientes para iniciar sesión en la aplicación móvil y realizar sus actividades correspondientes, como: enviar formularios de emergencia y aprobar o negar solicitudes de emergencia y/o donación.

Restablecimiento de contraseñas

El súper administrador no podrá restablecer sus credenciales, en caso de que la contraseña sea olvidada, deberá solicitar una nueva al equipo de desarrollo.

Los usuarios con el perfil: fundación, voluntario y ciudadano, podrán restablecer su contraseña a través de los siguientes pasos:

1. Ingresar el correo electrónico de la cuenta.
2. El sistema web envía un correo con un código de verificación de 4 dígitos, el cual deberá ser ingresado en el campo respectivo.
3. El sistema web verifica el código ingresado.
4. El usuario escribe una nueva contraseña.
5. Iniciar sesión con la nueva contraseña.

3.2. Sprint 1: Inicio de sesión del usuario administrador - módulo fundación

De acuerdo a la planificación realizada en el Sprint Backlog descrito en el Manual Técnico – Sección Sprint Backlog. El Sprint 1 contiene la implementación del inicio de sesión del usuario administrador en el sistema web y la gestión de los módulos asignados a su cargo. A continuación, se presentan las tareas definidas:

- Página de inicio.
- Inicio de sesión del usuario administrador.
- Registro, visualización, modificación y eliminación de fundaciones.
- Notificaciones de solicitudes de fundaciones.
- Visualización, aprobación y/o negación de solicitudes de fundaciones

3.2.1. Página de inicio

Fig. 12, ilustra la página de inicio, la cual presenta información sobre el proyecto integrador, los autores y el enlace para descargar la aplicación móvil. Además, información sobre las fundaciones, animales de compañía y emergencias registradas en el sistema web con opciones de filtrado.

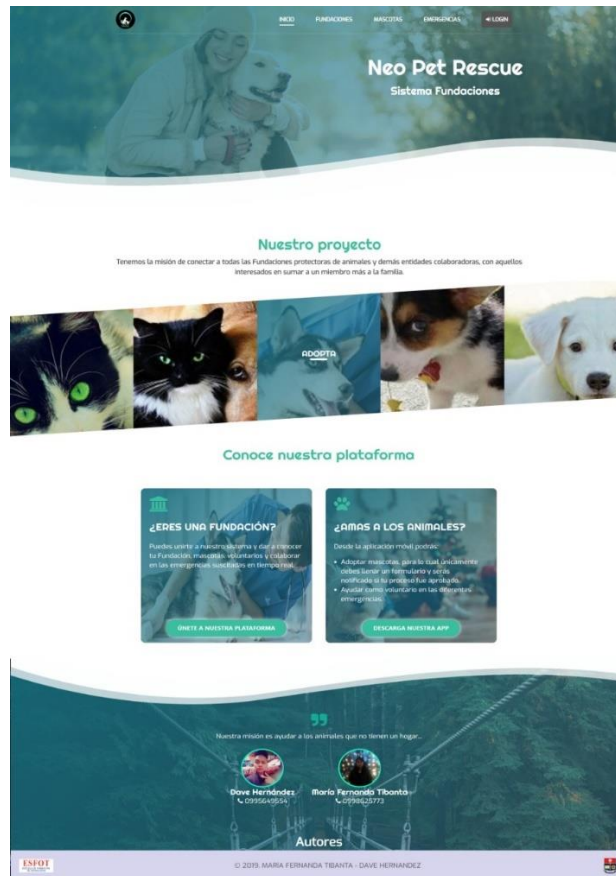


Fig. 12: Página de inicio

3.2.2. Inicio de sesión del usuario administrador

El sistema web es gestionado por un único usuario administrador para que inicie sesión con las credenciales asignadas anteriormente, como ilustra Fig. 13.

El sistema web verifica que el usuario este registrado en la base de datos y a que perfil o rol pertenece para su redireccionamiento a la página de panel de fundaciones, voluntarios y ciudadanos.

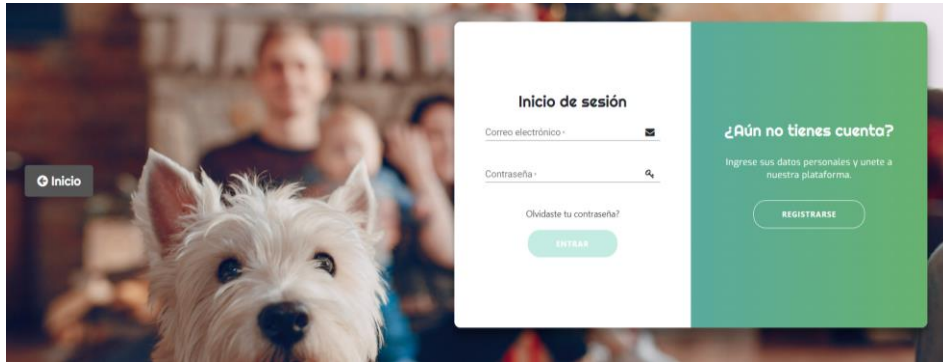


Fig. 13: Inicio de sesión del usuario administrador

3.2.3. Registro, visualización, modificación y eliminación de fundaciones

Fig. 14, ilustra el formulario para el registro de una nueva fundación conjuntamente con la validación de campos. Mientras que en el Manual de Usuario – Sección 1.3, se presenta de forma detalla la visualización, modificación y eliminación de una fundación. Cabe recalcar que los datos de la fundación no serán eliminados de la base de datos, sino que se procederá a cambiar el estado de activo a inactivo.

Fig. 14: Formulario para el registro de una fundación

3.2.4. Notificaciones de solicitudes de fundaciones

Fig. 15, ilustra una campana de notificaciones, es decir, una lista con todas solicitudes de fundaciones que desean unirse al sistema web, en donde el usuario administrador puede aprobar o negar dichas solicitudes.

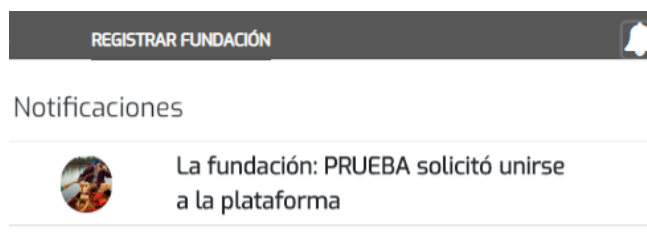


Fig. 15: Visualización de notificaciones

3.2.5. Visualización, aprobación y/o negación de solicitudes de fundaciones

Fig. 16, ilustra las solicitudes que envían las fundaciones que quieren formar parte del sistema web. Mientras que en el Manual de Usuario – Sección 1.5, se presenta de forma detalla el proceso de aprobación o negación de la solicitud de una fundación.

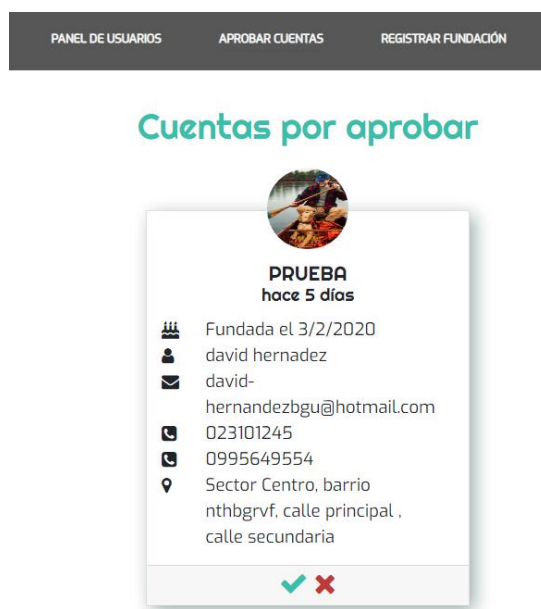


Fig. 16: Pantalla para visualizar las solicitudes de las fundaciones

3.3. Sprint 2: Inicio de sesión del usuario administrador de la fundación – módulo de emergencia, adopción, donación, voluntario y animales de compañía

De acuerdo a la planificación realizada en el Sprint Backlog descrito en el Manual Técnico – Sección Sprint Backlog. El Sprint 2 contiene la implementación del inicio de sesión del usuario administrador de la fundación en el sistema web y la gestión de los módulos asignados a su cargo. A continuación, se presentan las tareas definidas:

- Página de Inicio de la fundación.
- Inicio de sesión del usuario administrador de la fundación en el sistema web.
- Registro, visualización, modificación y eliminación de información de página de inicio
- Registro, visualización, modificación y eliminación de animales de compañía.
- Registro, visualización, modificación y eliminación de voluntarios.
- Notificaciones de solicitudes.
- Visualización, aprobación y/o negación de solicitudes de emergencia.
- Visualización, aprobación y/o negación de solicitudes de adopción.
- Registro, visualización, aprobación y/o negación de solicitudes de donación.
- Visualización y modificación de la información de la fundación.

3.3.1. Página de inicio de la fundación

Figura 17, lustra la página de inicio de la fundación la cual presenta información como: foto de perfil, banner, nosotros, misión, visión, historias inspiradoras, etc. Mientras que en el Manual de Usuario – Sección 2.1, se presenta de forma detallada la visualización de la información sobre animales de compañía, donaciones, ubicación, etc.

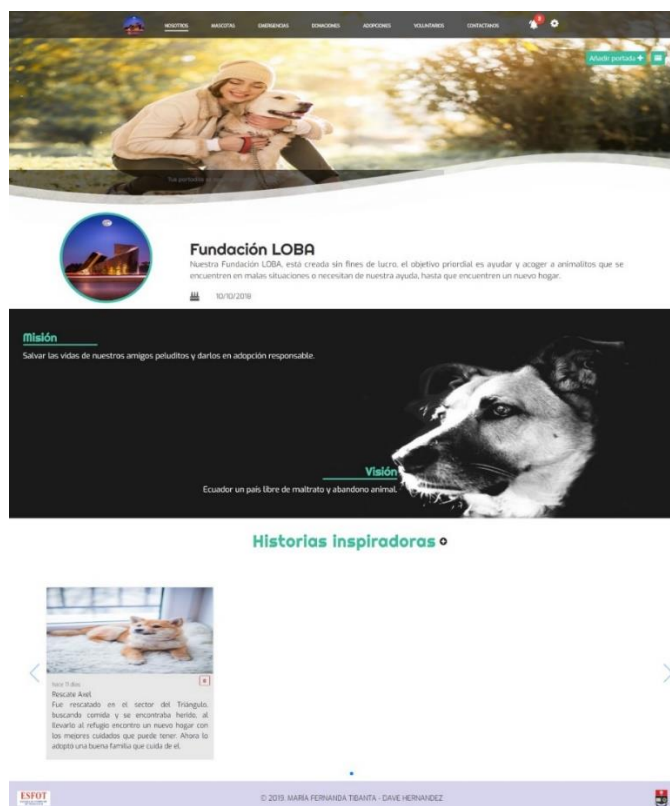


Fig. 17: Página de inicio de la fundación

3.3.2. Inicio de sesión del usuario administrador de la fundación en el sistema web

La fundación tiene como objetivo, que un único usuario inicie sesión en el sistema web con las respectivas credenciales, como ilustra Fig. 18.

El sistema web verifica que el usuario este registrado en el sistema web y a que perfil o rol pertenece para su redireccionamiento a la página de inicio de la fundación.

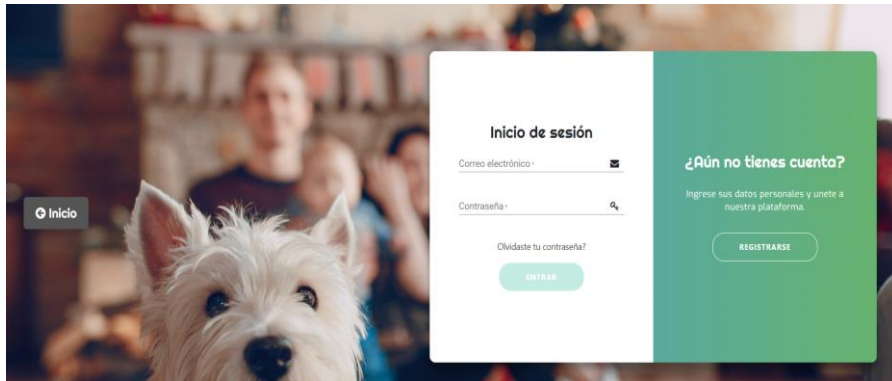


Fig. 18: Inicio de sesión del usuario administrador de la fundación

3.3.3. Registro, visualización, modificación y eliminación de información de página de inicio

Fig. 19, ilustra las secciones para editar la información de la página de inicio como misión, visión, imágenes de portadas, historias inspiradoras, entre otras. Mientras que en el Manual de Usuario – Sección 2.3, se presenta de forma detallada el procedimiento para el registro, visualización, modificación y eliminación de información de página de inicio.

Fig. 19: Formulario para subir fotos de portada a la página de inicio de la fundación

3.3.4. Registro, visualización, modificación y eliminación de animales de compañía

Fig. 20, ilustra el formulario para el registro de un animal de compañía conjuntamente con la validación de campos. Mientras que en el Manual de Usuario – Sección 2.4, se presenta de forma detallada el procedimiento para la visualización, modificación y eliminación de un animal de compañía. Cabe recalcar que los datos del animal de compañía no serán eliminados de la base de datos, sino que se procederá a cambiar el estado de activo a inactivo, siempre y cuando el animal de compañía sea adoptado o fallezca por alguna circunstancia.

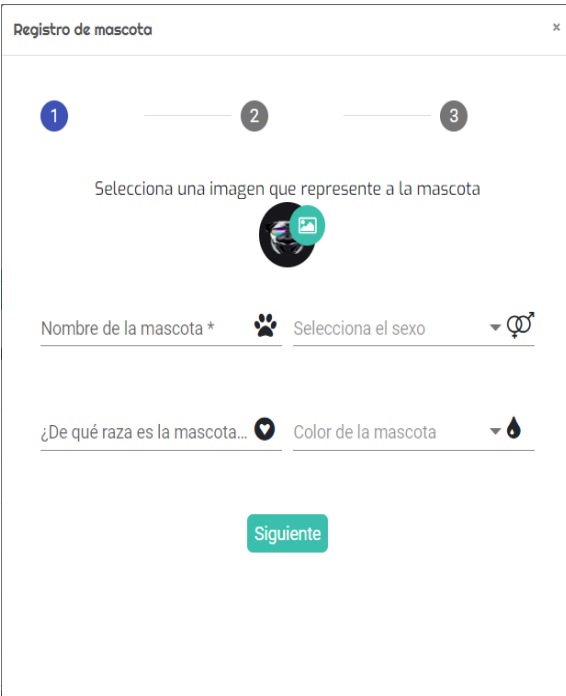


Fig. 20: Formulario para el registro de un animal de compañía

3.3.5. Registro, visualización, modificación y eliminación de voluntarios

Fig. 21, ilustra el formulario para el registro de un voluntario conjuntamente con la validación de campos. Mientras que en el Manual de Usuario – Sección 2.5, se presenta de forma detallada el procedimiento para la visualización, modificación y eliminación de un voluntario. Cabe recalcar que los datos del voluntario no serán eliminados de la base de datos, sino que se procederá a cambiar el estado de activo a inactivo.

Fig. 21: Formulario para el registro de un voluntario

3.3.6. Notificaciones de solicitudes

Fig. 22, ilustra una campana de notificaciones, es decir, una lista de todas las notificaciones sobre solicitudes de adopciones, emergencias y donaciones.

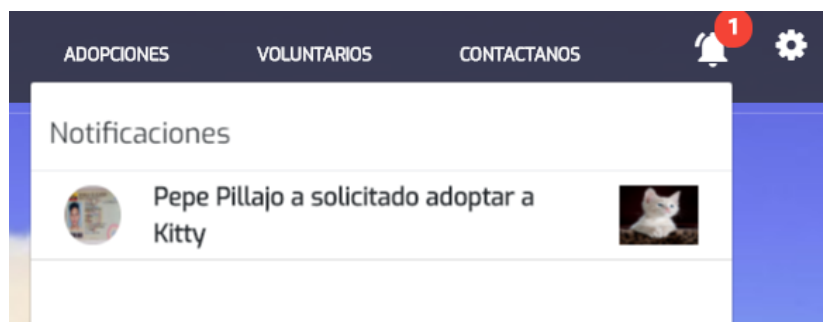


Fig. 22: Visualización de notificaciones de adopciones, emergencias y donaciones

3.3.7. Visualización, aprobación y/o negación de solicitudes de emergencia

Fig. 23, ilustra todas las solicitudes de emergencias enviadas por voluntarios o ciudadanos a través de la aplicación móvil. Mientras que en el Manual de Usuario – Sección 2.7, se presenta de forma detallada la información de la emergencia y el proceso por el cual pasa la emergencia para ser atendida, en donde se asigna la tarea a uno o varios voluntarios.

Posterior a ello se envía una notificación a la aplicación móvil, donde los voluntarios aceptan o niegan dicha solicitud.



Fig. 23: Pantalla para visualizar las solicitudes de emergencia

3.3.8. Visualización, aprobación y/o negación de solicitudes de adopción

Fig. 24, ilustra todas las solicitudes de adopciones enviadas por ciudadanos a través de la aplicación móvil. Mientras que en el Manual de Usuario – Sección 2.8, se presenta de forma detallada la información de la adopción y el proceso de aprobación o negación de dicha solicitud, para que la notificación sea enviada al ciudadano quien solicitó la adopción.

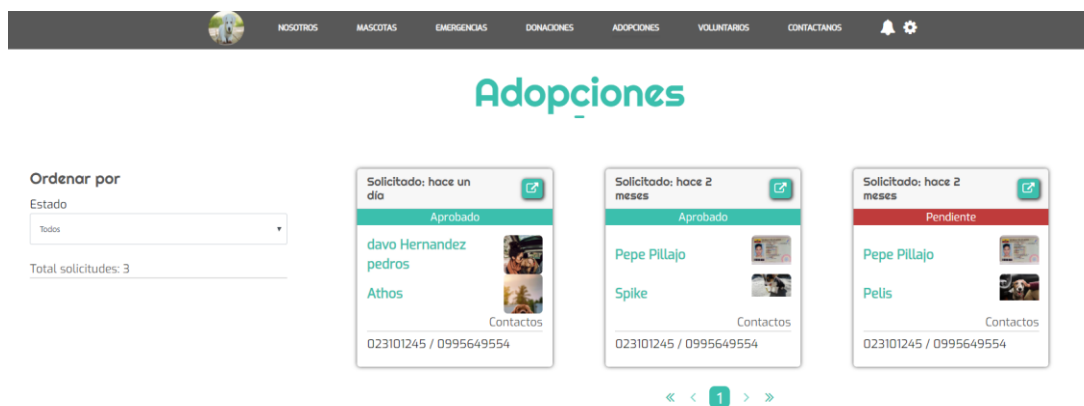


Fig. 24: Pantalla para visualizar las solicitudes de adopciones

3.3.9. Registro, visualización, aprobación y/o negación de solicitudes de donación

Fig. 25, ilustra el formulario para registrar una donación ya sea económica o de producto. Fig. 26, ilustra todas las solicitudes de donaciones económicas y de productos enviadas por ciudadanos a través de la aplicación móvil. Mientras que en el Manual de Usuario – Sección 2.9, se presenta de forma detallada la información y el proceso por el

cual pasa la donación para ser atendida, en donde se asigna la tarea a uno o varios voluntarios. Posterior a ello se envía una notificación a la aplicación móvil, donde los voluntarios aceptan o niegan dicha solicitud.

Registro de donación

2

Selecciona la foto del comprobante que asegure que se realizo la donación

Cantidad de dinero *

Descripción de la donación *

Regresar Guardar

Fig. 25: Formulario para el registro de una donación económica

Registrar donación

Ordenar por

Tipo de donación

Todos

Total donaciones: 1

Producto

Asignada

Pepas mi mascota

Para ocasiones importantes

Donador

Pepe Pillajo

« < 1 > »

Fig. 26: Pantalla para visualizar las solicitudes de donaciones

3.3.10. Visualización y modificación de la información personal del usuario

Fig. 27, ilustra las opciones que tiene el usuario administrador de la fundación en el sistema web a parte de los módulos asignados a su cargo.

Cuando el usuario seleccione Actualizar perfil, se le presenta un menú de opciones permitiéndole la actualización de sus datos personales y de la fundación, como ilustra Fig. 28.

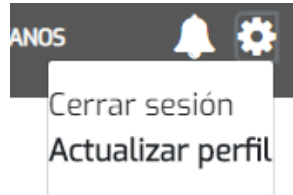


Fig. 27: Pantalla con opciones para el usuario administrador de la fundación

The screenshot shows a web interface for updating a profile. At the top is a dark navigation bar with a globe icon and menu items: NOSOTROS, MASCOTAS, EMERGENCIAS, DONACIONES, ADOPCIONES, VOLUNTARIOS, CONTACTANOS, and a notification bell/gear icon. Below the navigation bar, the page title is 'Actualizar perfil'. On the left is a sidebar with three items: 'Perfil' (highlighted in green), 'Datos de la cuenta', and 'Contraseña'. The main content area is titled 'Perfil, misión y visión' with a subtitle 'Describe la misión y visión de tu fundación'. It contains three text input fields, each with a green checkmark on the right and a character count on the bottom right. The first field is 'Perfil *' with the value 'Somos una fundación que lucha a favor de los derechos de las mascotas.' and a count of 70/300. The second field is 'Misión *' with the value 'Nuestra fundación ayuda a todos los animalitos sin distinción de especie a tener una mejor calidad de vida.' and a count of 107/300. The third field is 'Visión *' with the value 'Ser una organización autosustentable, reconocida a nivel nacional el bienestar animal, responsable del cambio en la relación humano animal en el Ecuador.' and a count of 153/300. At the bottom right of the form is a green 'Actualizar' button. Below the form are two expandable sections: 'Depósitos' (Información para depósitos y transferencias bancarias) and 'Contactos' (Información para contactos con la fundación).

Fig. 28: Formulario para modificar los datos personales del usuario administrador y de la fundación

3.4. Sprint 3: Registro e inicio de sesión del usuario ciudadano - módulo fundación, formulario, animales de compañía y notificaciones

De acuerdo a la planificación realizada en el Sprint Backlog descrito en el Manual Técnico – Sección Sprint Backlog. El Sprint 3 contiene la implementación del registro e inicio de sesión del ciudadano en la aplicación móvil y la gestión de los módulos asignados a su cargo. A continuación, se presentan las tareas definidas:

- Página de inicio, registro e inicio de sesión del usuario ciudadano.
- Visualización de animales de compañía.
- Visualización de fundaciones.
- Envío de formularios de adopción, emergencia y donación
- Recibir notificaciones de aprobación o negación de solicitudes de adopción, emergencia y/o donación
- Visualización y modificación de la información personal del usuario

3.4.1. Página de inicio, registro e inicio de sesión del usuario ciudadano

Al iniciar la aplicación móvil por primera vez, se presenta una página de inicio como ilustra Fig. 29, presentando opciones para el inicio de sesión o el registro.

Cuando el usuario ciudadano inicie sesión, la aplicación móvil verifica que el usuario ciudadano este previamente registrado y a que perfil o rol pertenece para su redireccionamiento a la visualización de mascotas.

Si el usuario ciudadano no se encuentra registrado en la aplicación móvil, debe llenar un formulario, como ilustra Fig. 30. Mientras que en el Manual de Usuario – Sección 3.1, se presenta de forma detallada el inicio de sesión, cambio de contraseña y el menú de navegación inferior del usuario voluntario.



Fig. 29: Página de inicio



Fig. 30: Formulario de registro de un ciudadano

3.4.2. Visualización de animales de compañía

Fig. 31, ilustra todos los animales de compañía registrados por cada fundación, los cuales pueden ser filtrados para una mejor visualización. Mientras que en el Manual de Usuario – Sección 3.2, se presenta de forma detallada la información que presenta cada animal de compañía.

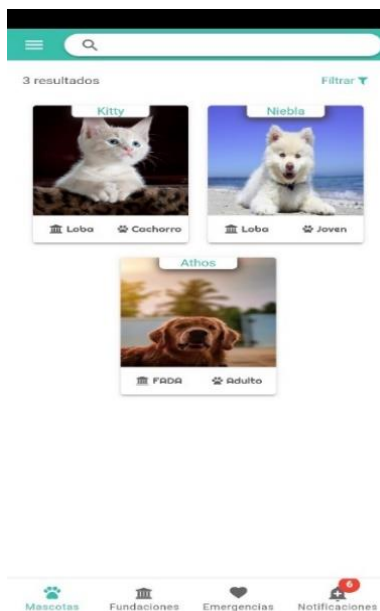


Fig. 31: Pantalla para visualizar los animales de compañía

3.4.3. Visualización de Fundaciones

Fig. 32, ilustra todas las fundaciones registradas en el sistema web, las cuales pueden ser filtradas para una mejor visualización. Mientras que en el Manual de Usuario – Sección 3.3, se presenta de forma detallada la información de cada fundación como: perfil, mascotas, donaciones y contacto.

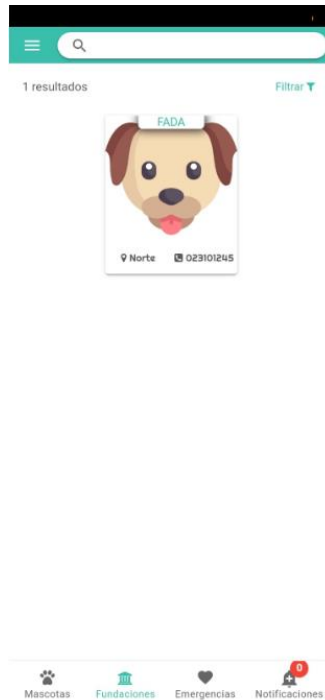


Fig. 32: Pantalla para visualizar las fundaciones

3.4.4. Envío de formularios de adopción, emergencia y donación

Fig. 33, ilustra el formulario para que el usuario ciudadano realice una adopción. Mientras que en el Manual de Usuario – Sección 3.4, se presenta de forma detallada la información sobre los formularios de emergencia y donación económica o de producto.

Fig. 33: Formulario para solicitar una adopción

3.4.5. Recibir notificaciones de aprobación o negación de solicitudes de adopción, emergencia y/o donación

Fig. 34, ilustra las notificaciones que llegan a la aplicación móvil informando si la solicitud de adopción, emergencia y/o donación fue aprobada o negada respectivamente. Además, todas las notificaciones pueden ser filtradas para una mejor visualización.

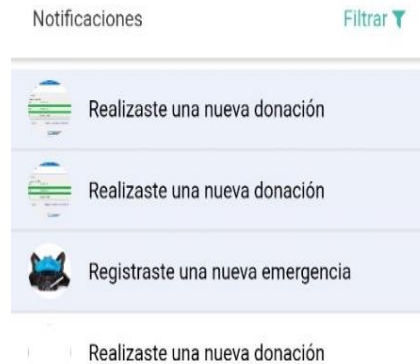


Fig. 34: Notificaciones para la visualización de solicitudes aprobadas o negadas

3.4.6. Visualización y modificación de la información personal del usuario

Fig. 35, ilustra todas las opciones que tiene el usuario ciudadano a parte de los módulos asignados a su cargo. Además, cuando el usuario seleccione actualizar perfil, se le presenta un menú de opciones para actualizar sus datos personales. Mientras que en el Manual de Usuario – Sección 3.6, se presenta de forma detallada cada una de las opciones que tiene el usuario para actualizar su información personal.

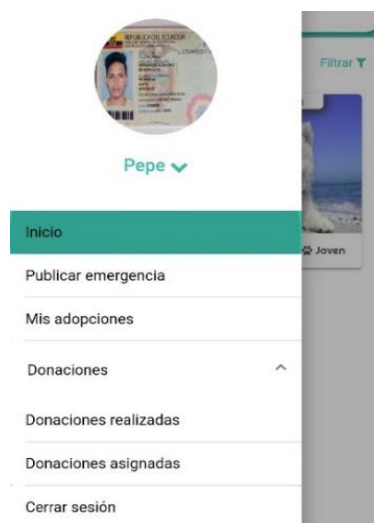


Fig. 35: Pantalla con opciones para el usuario ciudadano

3.5. Sprint 4: Inicio de sesión del usuario voluntario – módulo emergencia y solicitudes

De acuerdo a la planificación realizada en el Sprint Backlog descrito en el Manual Técnico – Sección Sprint Backlog. El Sprint 4 contiene la implementación del inicio de sesión del voluntario en la aplicación móvil y la gestión de los módulos asignados a su cargo. A continuación, se presentan las tareas definidas:

- Página de inicio e inicio de sesión del usuario voluntario.
- Envío del formulario de emergencia.
- Visualización, aprobación y/o negación de solicitudes de emergencia y donaciones
- Visualización y modificación de la información personal del usuario

3.5.1. Página de inicio e inicio de sesión del usuario voluntario

Al iniciar la aplicación móvil por primera vez, se presenta una página de inicio como ilustra Fig. 36, mostrando las opciones para el inicio de sesión o el registro. Cuando el usuario voluntario inicie sesión, la aplicación móvil verifica que este previamente registrado y a que perfil o rol pertenece para su redireccionamiento.

En el Manual de Usuario – Sección 4.1, se presenta de forma detallada el cambio de contraseña y el menú de navegación inferior del usuario voluntario.



Fig. 36: Página de Inicio

3.5.2. Enviar formulario de emergencia

Fig. 37, ilustra el formulario para solicitar una emergencia. Mientras que en el Manual de Usuario – Sección 4.2, se presenta de forma detallada cada uno de los campos y validaciones para reportar una emergencia.

The figure shows two sequential steps of a mobile form for reporting an emergency. Step 1 (left) is titled 'Emergencia' and includes a progress indicator with '1' and '2'. It features a 'Selecciona una foto' section with a camera icon, a 'Tipo de emergencia' dropdown menu, and a 'Descripción de la emergencia *' text field. Step 2 (right) also titled 'Emergencia' includes a 'Sector' dropdown menu, a 'Seleccionar dirección' button with a location pin icon, a 'Referencia *' text field, and a 'Número de contacto extra *' text field. Both steps have 'Regresar' and 'Siguiete' (or 'Enviar') buttons at the bottom.

Fig. 37: Formulario para solicitar una emergencia

3.5.3. Visualización, aprobación y/o negación de solicitudes de emergencias y donaciones

Fig. 38 y 39, ilustran el detalle de la solicitud de emergencia y donación que llegan a la aplicación móvil por medio de notificaciones. Posterior a ello el usuario voluntario puede aprobar la solicitud y seguir las tareas asignadas o negar dicha solicitud dependiendo del caso ver Manual de Usuario – Sección 4.3.

The screenshot shows the details of an emergency request. At the top, there is a back arrow and the title 'Emergencia'. Below it is a photo of a dog with the title 'Mascota'. Underneath the photo are three tabs: 'Emergencia', 'Dirección', and 'Observaciones'. The main content area shows 'Publicado: hace un mes' and 'Tipo: Mal estado de salud'. Below that is a 'Descripción' field with a dropdown arrow. At the bottom, there are two buttons: 'Aprobar' (green) and 'Negar' (red).

Fig. 38: Formulario para aprobar y/o negar la solicitud de una emergencia

The screenshot shows the details of a donation request. At the top, there is a back arrow and the title 'Donación'. Below it is a product image with the title 'Donación' and a 'pendiente' status. Underneath the image are three tabs: 'Donación', 'Dirección', and 'Observaciones'. The main content area shows 'Tipo: Producto', 'Nombre Producto: yyyy', and a 'Descripción' field with a dropdown arrow. At the bottom, there are two buttons: 'Aprobar' (green) and 'Negar' (red).

Fig. 39: Formulario para aprobar y/o negar la solicitud de una donación

3.5.4. Visualización y modificación de la información personal del usuario

Fig. 40, ilustra las opciones que tiene el usuario voluntario a parte de los módulos asignados a su cargo. Cuando el usuario seleccione actualizar perfil, se le presenta un menú de opciones para actualizar sus datos personales. Mientras que en el Manual de Usuario – Sección 4.4, se presenta de forma detallada cada una de las opciones que tiene el usuario para actualizar su información personal.

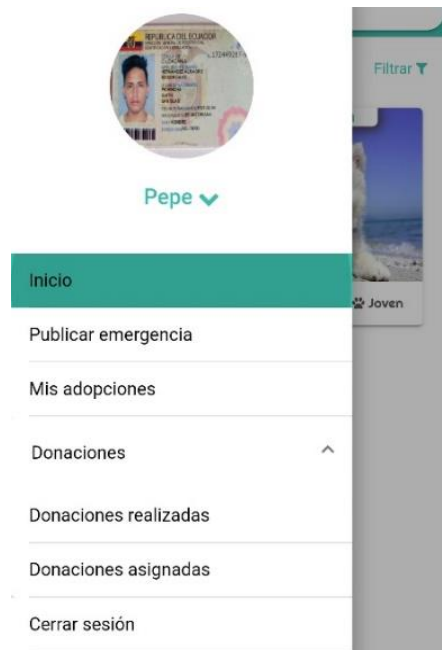


Fig. 40: Pantalla con opciones para el usuario ciudadano

3.6. Sprint 5: Pruebas en el sistema web y aplicación móvil

De acuerdo a la planificación realizada en el Sprint Backlog descrito en el Manual Técnico – Sección Sprint Backlog. El Sprint 5 contiene las pruebas que se realizaron al sistema web y aplicación móvil, antes de su implementación en Heroku y App Store. A continuación, se presentan las tareas definidas:

- Pruebas de carga
- Pruebas de compatibilidad
- Pruebas funcionales

3.6.1. Pruebas de carga

Las pruebas de carga, permiten medir el comportamiento de un sistema, es decir, la capacidad máxima que tiene un servidor web, para atender a un conjunto de peticiones por parte de los usuarios de manera simultánea [25].

Las pruebas de carga ayudaron a conocer el número de peticiones que los usuarios pueden realizar al sistema sin que este colapse o tenga alguna dificultad en el funcionamiento del mismo.

Por tal motivo, para este tipo de pruebas se utilizó la herramienta ApacheBench que viene incluida en el paquete de XAMPP⁷, permitiendo medir el rendimiento de un servidor web HTTP por medio de línea de comandos [26].

Desde la línea de comandos del Sistema Operativo Windows se ejecutó el siguiente comando **ab -c 10 -n 100 http://localhost:4200/**, en el cual -c indica las conexiones concurrentes y -n indica el número de peticiones en total.

Realizando 3 pruebas de carga con 50, 500, 10000 peticiones respectivamente y 10 conexiones concurrentes al servidor del sistema web.

Prueba 1 (50 peticiones)

Fig. 41, ilustra las 50 peticiones que fueron exitosas y que cada petición se realizó en un tiempo de 409 milisegundos, menos de medio segundo aproximadamente. Además, de indicar que la petición 8 fue la más rápida en atenderse y la petición 10 fue la que se demoró más en ser completada.

⁷ **XAMPP:** Es un servidor independiente de plataforma de código libre, permite utilizar varios servidores y probar el desarrollo de un proyecto en un ordenador propio sin necesidad de acceder a internet. Obtenido de: <http://janda1617smr2curro.blogspot.com/2017/03/que-es-xampp-y-para-que-sirve.html>

```

Server Software:
Server Hostname:      localhost
Server Port:         4200

Document Path:      /
Document Length:    1277 bytes

Concurrency Level:   10
Time taken for tests: 2.047 seconds
Complete requests:  50
Failed requests:    0
Total transferred:  76700 bytes
HTML transferred:  63850 bytes
Requests per second: 24.43 [#/sec] (mean)
Time per request:   409.381 [ms] (mean)
Time per request:   40.938 [ms] (mean, across all concurrent requests)
Transfer rate:      36.59 [kbytes/sec] received

Connection Times (ms)
  min  mean[+/-sd] median  max
Connect:    0    0  0.2    0    1
Processing:  2    8  1.7    8   10
Waiting:    2    6  1.7    6    9
Total:      2    8  1.7    8   10

Percentage of the requests served within a certain time (ms)
 50%    8
 66%    8
 75%    9
 80%    9
 90%   10
 95%   10
 98%   10
 99%   10
100%   10 (longest request)

```

Fig. 41: Prueba de carga #1 50 peticiones

Prueba 2 (500 peticiones)

Fig. 42, ilustra las 500 peticiones que fueron exitosas y que cada petición se realizó en un tiempo de 54.325 milisegundos, siendo la petición 28 la que tardo más en ser procesada. Llegando a la conclusión que el sistema web no tendrá fallos al procesar dicha cantidad de peticiones sin ningún problema.

```

Server Software:
Server Hostname:      localhost
Server Port:         4200

Document Path:      /
Document Length:    1277 bytes

Concurrency Level:   10
Time taken for tests: 2.716 seconds
Complete requests:  500
Failed requests:    0
Total transferred:  767000 bytes
HTML transferred:  638500 bytes
Requests per second: 184.08 [#/sec] (mean)
Time per request:   54.325 [ms] (mean)
Time per request:   5.433 [ms] (mean, across all concurrent requests)
Transfer rate:      275.76 [kbytes/sec] received

Connection Times (ms)
  min  mean[+/-sd] median  max
Connect:    0    0  0.3    0    1
Processing:  5   13  3.8   13   28
Waiting:    3    9  3.7    9   25
Total:      5   14  3.8   13   28

Percentage of the requests served within a certain time (ms)
 50%   13
 66%   15
 75%   16
 80%   17
 90%   18
 95%   20
 98%   23
 99%   23
100%   28 (longest request)

```

Fig. 42: Prueba de carga #2 500 peticiones

Prueba 3 (100.0000 peticiones)

Fig. 43, ilustra las 100.0000 peticiones que fueron procesadas exitosamente en un tiempo menor a medio segundo por cada petición. Llegando a la conclusión que el sistema web no tendrá fallos en procesar dicha cantidad de peticiones aún más cuando el sistema web este en producción.

```
Completed 800000 requests
Completed 900000 requests
Completed 1000000 requests
Finished 1000000 requests

Server Software:
Server Hostname: localhost
Server Port: 4200

Document Path: /
Document Length: 1277 bytes

Concurrency Level: 100
Time taken for tests: 583.647 seconds
Complete requests: 1000000
Failed requests: 0
Total transferred: 1534000000 bytes
HTML transferred: 1277000000 bytes
Requests per second: 1713.36 [#/sec] (mean)
Time per request: 58.365 [ms] (mean)
Time per request: 0.584 [ms] (mean, across all concurrent requests)
Transfer rate: 2566.70 [Kbytes/sec] received

Connection Times (ms)
min mean[+/-sd] median max
Connect: 0 0 0.3 0 7
Processing: 19 58 301.2 52 30315
Waiting: 1 47 299.6 41 30298
Total: 19 58 301.2 52 30315

Percentage of the requests served within a certain time (ms)
50% 52
66% 56
75% 60
80% 63
90% 71
95% 80
98% 94
99% 107
100% 30315 (longest request)
```

Fig. 43: Prueba de carga #3 1000000 peticiones

3.6.2. Pruebas de compatibilidad

Las pruebas de compatibilidad, son pruebas que se le realizan a un software o aplicación determinada en diferentes navegadores de internet, con el objetivo de determinar cómo se visualizará el diseño y estructura del contenido [27].

Sistema web

Para realizar este tipo de pruebas se utilizó tres navegadores, el detalle de los resultados de las pruebas se ilustran en el Manual Técnico – Sección Pruebas de Compatibilidad.

Con el resultado de las pruebas realizadas, se determina que el contenido, módulos y funcionalidades del sistema web funcionarán correctamente en todas las versiones de los navegadores web, como se presenta en la TABLA V.

TABLA V: Navegadores que soportan el sistema web

NOMBRE	VERSIÓN
Mozilla Firefox	68.0.2
Google Chrome	80.0.3987.132
Microsoft Edge	44.18362.449.0

Aplicación móvil

Para realizar este tipo de pruebas se utilizó tres dispositivos móviles con diferentes Sistemas Operativos, los resultados de las pruebas se ilustran en el Manual Técnico – Sección Pruebas de Compatibilidad.

Con las pruebas realizadas, se determina que el contenido, módulos, funcionalidades y notificaciones de aplicación móvil se ejecutan correctamente en los diferentes dispositivos móviles, como se presenta en la TABLA VI.

TABLA VI: Dispositivos móviles en los que se probó la aplicación móvil

NOMBRE	VERSIÓN
Xiaomi Note 8 pro	Android 9
Iphone Ionic View	----
Samsung S6 edge	Android 7

3.6.3. Pruebas funcionales

Las pruebas de funcionalidad se centran en comprobar que los sistemas desarrollados funcionen acorde a las especificaciones funcionales y requisitos del cliente [28].

Las pruebas realizadas en el sistema web y aplicación móvil, se llevaron a cabo con un grupo de usuarios y equipos informáticos para que las funcionalidades sean probadas en su totalidad, obteniendo de esta manera una retroalimentación.

A continuación, las TABLAS VII – VIII presentan los resultados de las pruebas en algunas de las funcionalidades más relevantes del sistema web y aplicación móvil. Mientras que el resultado de las demás pruebas realizadas se encuentra descritas en el Manual Técnico – Sección Pruebas de Funcionalidad.

TABLA VII: Prueba de funcionalidad - Inicio de sesión del administrador en el sistema web

PRUEBA DE FUNCIONALIDAD	
Identificador (ID): PF001	Identificador historia de usuario: HU001
Nombre prueba de funcionalidad: Inicio de sesión del administrador en el sistema web	
Descripción: El usuario administrador puede iniciar sesión en el sistema web, con las credenciales respectivas.	
Pasos de ejecución: Ir a la página de inicio de la fundación Hacer clic en el botón “Login” Llenar el formulario con el correo electrónico y contraseña Hacer clic en el botón “Entrar”.	
Resultado deseado: El sistema web presento el panel de las todas las fundaciones que están registradas en el sistema web con su respectiva información. Además, se pudo ver un listado de todos los voluntarios y ciudadanos registrados y su información respectiva.	
Evaluación de la prueba: Se comprobó el resultado esperado. Aprobación del cliente 100%	

TABLA VIII: Prueba de funcionalidad - Inicio de sesión del usuario ciudadano en la aplicación móvil

PRUEBA DE FUNCIONALIDAD	
Identificador (ID): PF019	Identificador historia de usuario: HU019
Nombre prueba de funcionalidad: Inicio de sesión del ciudadano en la aplicación móvil	
Descripción: El usuario ciudadano puede iniciar sesión en la aplicación móvil, con las credenciales respectivas o con su cuenta de Facebook.	
Pasos de ejecución: Abrir la aplicación móvil Hacer tap en el botón “Iniciar sesión” Llenar el formulario con el correo electrónico y contraseña Hacer tap en el botón “Iniciar sesión”.	
Resultado deseado: La aplicación móvil presente una página con los animales de compañía registrados en cada fundación.	
Evaluación de la prueba: Se comprobó el resultado esperado. Aprobación del cliente 100%	

3.7. Sprint 6: Despliegue sistema web y aplicación móvil

De acuerdo a la planificación realizada en el Sprint Backlog descrito en el Manual Técnico – Sección Despliegue. El Sprint 6 contiene los pasos a seguir para el despliegue del sistema web y la aplicación móvil. A continuación, se presentan las tareas definidas:

- Despliegue del sistema web en Heroku
- Despliegue de la aplicación móvil en Play Store

3.7.1. Despliegue del sistema web en Heroku

El despliegue del sistema web en Heroku, fue realizado en una serie de pasos que se encuentran descritos en el Manual de Instalación – Sección 1.1 y 1.2.

A continuación, Fig. 44 y 45, ilustran la creación de una nueva aplicación en Heroku y el despliegue del sistema web a producción.

Por último, la URL para acceder al sistema web, es la siguiente:

- <https://neo-front.herokuapp.com/home/inicio>

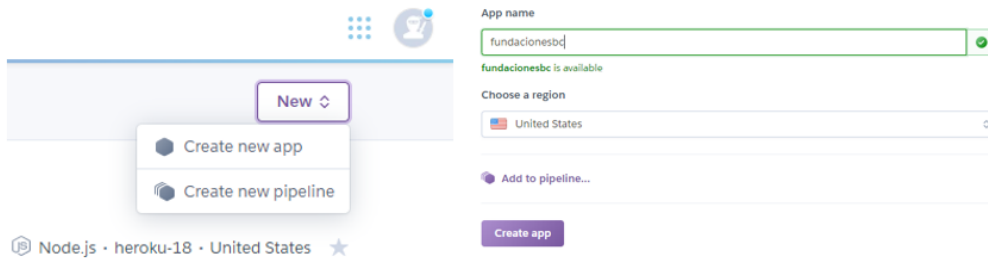


Fig. 44: Creación de la aplicación en Heroku

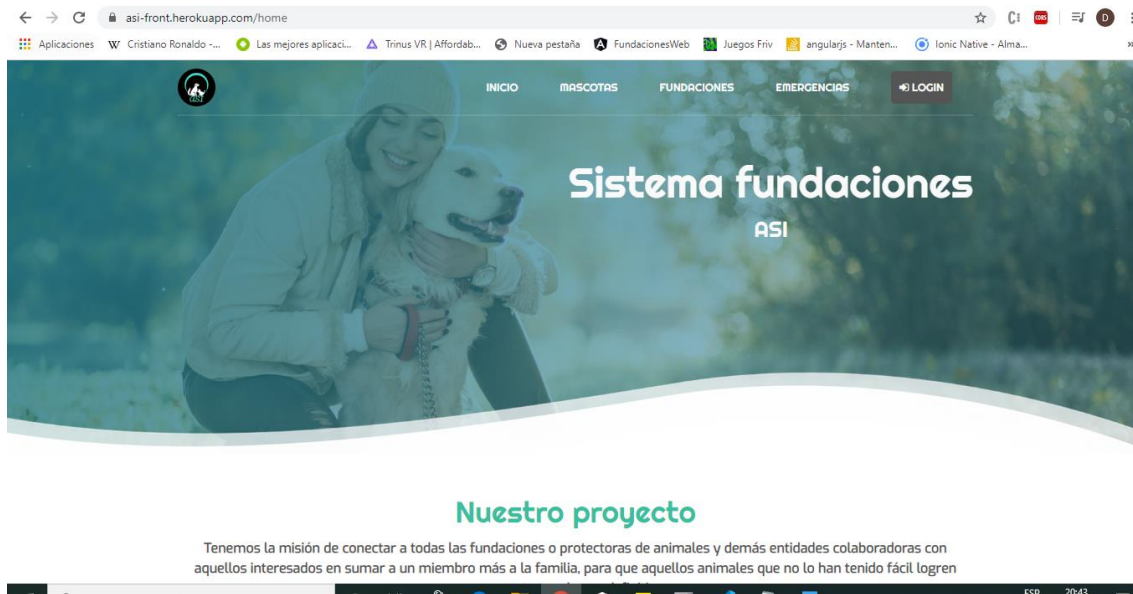


Fig. 45: Despliegue del sistema web bajo un dominio

3.7.2. Despliegue de la aplicación móvil en Play Store

El despliegue de la aplicación móvil en Play Store, se lo realizó en varios pasos que se encuentran descritos en el Manual de Instalación – desde la Sección 2.1 hasta la 2.6.

A continuación, Fig. 46 y 47 ilustran el inicio de sesión en Google Play Console y el resumen de que la aplicación fue publicada. Cabe recalcar que este proceso se demoró de 2 a 3 días mientras Google Play realizaba las verificaciones pertinentes.

Finalmente, Fig. 48 ilustra la publicación de la aplicación móvil en Play Store.

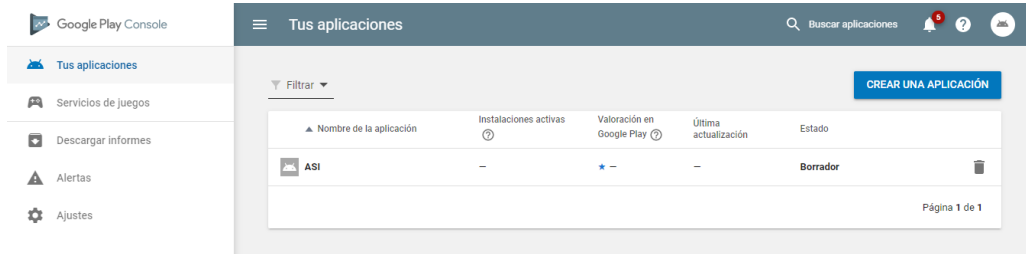


Fig. 46: Inicio de sesión en Google Play Console

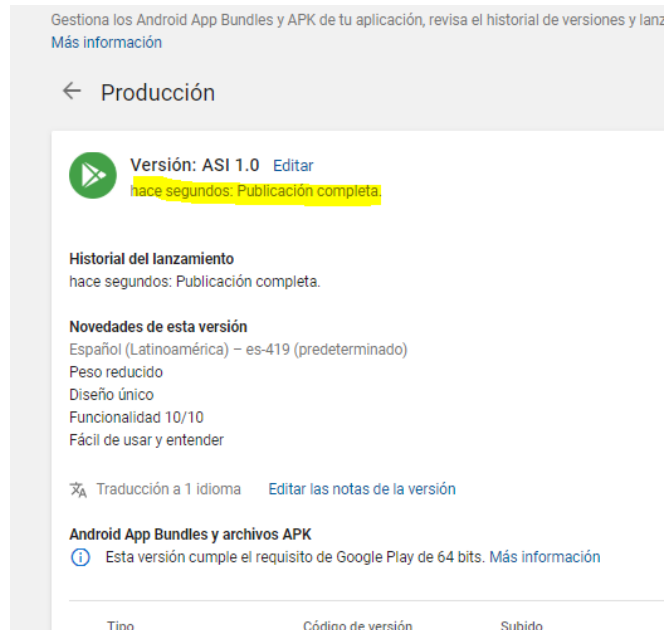


Fig. 47: Notificación de la publicación de la aplicación

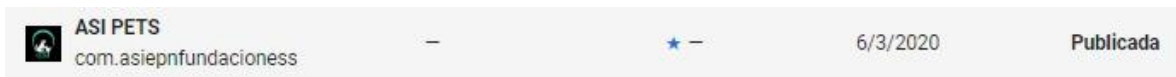


Fig. 48: Publicación de la aplicación móvil

4. CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

El trabajo en conjunto con el cliente y la disposición del equipo de desarrollo guiado por el Scrum Master, permitió cumplir satisfactoriamente los objetivos y el alcance del proyecto integrador.

Existe alrededor de 122.280 animales de compañía abandonados en las calles, motivo por el cual grupos de personas deciden crear fundaciones para ayudar a esta causa y así poder darles una nueva y mejor vida. El desarrollo del sistema web y aplicación móvil, permitirán la centralización de todas las fundaciones, llevando un mejor control, al igual que cada fundación de forma estandarizada gestionará sus actividades y procesos en un lapso de tiempo adecuado.

La utilización de la metodología Scrum, en el desarrollo de las funcionalidades del sistema web y aplicación móvil evitó la sobrecarga de trabajo dividiendo en periodos cortos de tiempo y así poder evaluar los resultados obtenidos por medio de los Sprints de desarrollo.

La Recopilación de Requerimientos permitió determinar la arquitectura y herramientas adecuadas para el desarrollo y despliegue del sistema web y aplicación móvil.

La utilización de Angular y Ionic ayudaron al desarrollo del sistema web y aplicación móvil de forma ágil, ya que, al utilizar Angular en el front-end del sistema web ayudó que el desarrollo de la aplicación móvil sea fácil, debido a que Ionic utiliza Angular como entorno de trabajo.

Las pruebas de carga, compatibilidad y funcionalidad realizadas en el sistema web y aplicación móvil, permitieron verificar que cada una de estas provee los resultados deseados. En el caso del sistema web se verificó que no es compatible con el navegador Internet Explorer porque Angular no es compatible con dicho navegador.

4.2. Recomendaciones

Para el desarrollo del sistema web y aplicación móvil se recomienda utilizar la arquitectura Modelo-Vista-Controlador, ya que permite la construcción e integración de nuevos módulos y la corrección de errores de manera fácil y rápida.

Los usuarios finales del sistema web y aplicación móvil deben tener conocimiento sobre el uso de navegadores web y celulares inteligentes; en el caso del sistema web no es recomendable utilizar el navegador Internet Explorer, ya que este no es compatible y se perderán ciertas funcionalidades.

En el caso de realizar mejoras en el sistema web y aplicación móvil, será necesario ver la versión de Angular y Ionic en la que se va a trabajar, ya que cada versión posee actualizaciones importantes.

Es recomendable que cada fundación establezca políticas de privacidad de información para los voluntarios que deseen apoyar a cada una de las fundaciones.

5. BIBLIOGRAFÍA

- [1] G. Castellanos, «El Comercio,» 09 Octubre 2018. [En línea]. Available: <https://www.elcomercio.com/narices-frias/poblacion-perros-callejeros-quito-censo.html>.
- [2] G. Castellanos, «El Comercio,» 15 Agosto 2017. [En línea]. Available: <https://www.elcomercio.com/narices-frias/voluntariado-refugios-animales-quito-abandono.html>.
- [3] V. Díaz y D. Landeta, «El Comercio,» 4 Octubre 2018. [En línea]. Available: <https://www.elcomercio.com/afull/perros-abandonados-quito-censo-parroquias.html>.
- [4] A. Menzinsky, G. López y J. Palacio., «Scrum Manager,» Julio 2016. [En línea]. Available: https://www.scrummanager.net/files/sm_proyecto.pdf.
- [5] SCRUMstudy, «Una guía para el conocimiento de Scrum,» 2013. [En línea]. Available: https://www.tenstep.ec/portal/images/pdfs/Suscripciones_TenStep/Silver/SCRUMstudy_GUIA_SBOK_espanol.pdf.
- [6] M. Trigas, «Metodología Scrum,» Agosto 2017. [En línea]. Available: <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/17885/1/mtrigasTFC0612memoria.pdf>.
- [7] K. Schwaber y J. Sutherland, «La Guía de Scrum,» Julio 2013. [En línea]. Available: <https://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-es.pdf>.
- [8] U. d. Cauca, «Programa de Ingeniería,» 26 06 2017. [En línea]. Available: http://artemisa.unicauca.edu.co/~cardila/IS_03__Ing_Requerimientos.pdf.
- [9] SCRUMstudy, «Tenstep,» 13 Enero 2018. [En línea]. Available: <https://www.tenstep.ec/portal/articulos-boletin-tenstep/41-scrum/253-scrum-como-escribir-historias-de-usuarios-sin-morir-en-el-intento>.
- [10] «Atareado,» 25 Septiembre 2017. [En línea]. Available: <https://www.atareao.es/software/disenio/pencil/>.
- [11] M. García, «Codingornot,» 05 Octubre 2017. [En línea]. Available: <https://codingornot.com/mvc-modelo-vista-controlador-que-es-y-para-que-sirve>.
- [12] S. Bellido, «Bibling,» 12 Agosto 2015. [En línea]. Available: http://bibling.us.es/proyectos/abreproy/12037/fichero/PFC_Sergio_Bellido_Sanchez%252F Tema5_mongodb.pdf.

- [13] J. Munro, «Tutsplus,» 09 Octubre 2017. [En línea]. Available: <https://code.tutsplus.com/es/articles/an-introduction-to-mongoose-for-mongodb-and-nodejs--cms-29527>.
- [14] F. Vega, «Platzi,» 23 Agosto 2017. [En línea]. Available: <https://platzi.com/desarrollo-angular/>.
- [15] J. Lucas, «Open Weinars,» 04 Septiembre 2019. [En línea]. Available: <https://openwebinars.net/blog/que-es-nodejs/>. [Último acceso: 26 Marzo 2020].
- [16] P. Morcillo, «Solucionex,» 27 Diciembre 2015. [En línea]. Available: <https://www.solucionex.com/blog/expressjs-un-framework-para-nodejs>. [Último acceso: 28 Marzo 2020].
- [17] Uesteibar, «Medium,» 03 Marzo 2015. [En línea]. Available: <https://medium.com/@uesteibar/env%C3%ADa-emails-desde-node-js-con-nodemailer-178cac5cf6b>. [Último acceso: 27 Marzo 2020].
- [18] D. J, «Capterra,» 17 Julio 2018. [En línea]. Available: <https://www.capterra.ec/software/163452/onesignal>. [Último acceso: 27 Marzo 2020].
- [19] Sergiodxa, «Platzi,» 2017. [En línea]. Available: <https://platzi.com/blog/introduccion-json-web-tokens/>. [Último acceso: 27 Marzo 2020].
- [20] A. Ardións, «RandyVarela,» 09 Septiembre 2016. [En línea]. Available: <http://www.randyvarela.es/ionic-definicion-ventajas/>.
- [21] M. Alvarez, «Desarrollo Web,» 02 Marzo 2017. [En línea]. Available: <https://desarrolloweb.com/articulos/que-es-ionic2.html>.
- [22] S/N, «Angular,» 10 Agosto 2017. [En línea]. Available: <https://angular.io>.
- [23] S/N, «CordovaApache,» 25 Mayo 2017. [En línea]. Available: <https://cordova.apache.org/docs/es/latest/guide/overview/>.
- [24] A. Roa, «Medium,» 1 Noviembre 2019. [En línea]. Available: <https://medium.com/@alejandrora/aprender-node-js-api-con-express-js-y-mongodb-12023cf62a0c>.
- [25] S/A, «Guia Digital,» 2018. [En línea]. Available: <http://www.guiadigital.gob.cl/articulo/pruebas-de-carga.html>. [Último acceso: 26 Marzo 2020].
- [26] D. Amoedo, «Ubunlog,» 2017. [En línea]. Available: <https://ubunlog.com/apachebench-carga-pagina-web/>.

[27] Globe, «Globe,» [En línea]. Available: <https://www.globetesting.com/pruebas-funcionales/>.

[28] Globe, «Globe,» S/F. [En línea]. Available: <https://www.globetesting.com/pruebas-funcionales/>. [Último acceso: 05 Marzo 2020].

6. ANEXOS

A continuación, se listan todos los documentos que se generaron a partir del desarrollo e implementación del sistema web y aplicación móvil.

- **MANUAL TÉCNICO**

1. Recopilación de Requerimientos
2. Historias de Usuario
3. Product Backlog
4. Sprint Backlog
5. Prototipo de Interfaces
6. Diseño de la Base de Datos
7. Pruebas de compatibilidad
8. Pruebas de funcionalidad

- **MANUAL DE USUARIO**

1. Manual de usuario del sistema web
2. Manual de usuario de la aplicación móvil

- **MANUAL DE INSTALACIÓN**

1. Despliegue del sistema web en Heroku
2. Despliegue de la aplicación móvil en Play Store