|  |
| --- |
| **ERS**  **CAPSTONE “PawFinder”** |

**Proyecto de Desarrollo de Aplicación para la Búsqueda de Mascotas Perdidas**

Curso: CAPSTONE\_301V

Profesor: Patricio Soto – Natalia Fuentes

Integrantes: Alonso Arellano

Javier Villalobos

Jorge Villalobos

Carrera: Ingeniería en Informática

Fecha: 05 de diciembre de 2024

**Índice**

[**1.** **Introducción** 3](#_Toc184523900)

[**2.** **Visión General del Sistema** 4](#_Toc184523901)

[**3.** **Análisis de Requerimientos De Software** 5](#_Toc184523902)

[**4.** **Restricciones** 9](#_Toc184523903)

[**5.** **Suposiciones y Dependencias** 11](#_Toc184523904)

[**6.** **Diagramas** 12](#_Toc184523905)

[**7.** **Vista Lógica** 13](#_Toc184523906)

[**8.** **Vista de Despliegue** 17](#_Toc184523907)

[**9.** **Vista Física** 17](#_Toc184523908)

[**10.** **Casos de Uso** 18](#_Toc184523909)

[**11.** **Diagrama de arquitectura** 43](#_Toc184523910)

[**12.** **Diagrama Base de Datos** 44](#_Toc184523911)

[**13.** **Diccionario de datos** 44](#_Toc184523912)

[**14.** **Criterios de Aceptación** 47](#_Toc184523913)

# **Introducción**

Este documento de Especificación de Requisitos del Software (ERS) tiene como propósito definir de manera detallada los requisitos funcionales y no funcionales del sistema PawFinder. La aplicación busca centralizar la información de mascotas perdidas, encontradas y en adopción, ofreciendo una plataforma accesible y eficiente para los usuarios, con el fin de facilitar la conexión entre quienes buscan o encuentran mascotas y promover la adopción responsable.

El ERS será utilizado como referencia principal para guiar el desarrollo, pruebas y despliegue del sistema, garantizando que el producto final cumpla con las expectativas de las partes interesadas.

* 1. **Alcance del Proyecto**
* **Incluido en el sistema:**
  + Registro y autenticación de usuarios.
  + Publicación de mascotas perdidas y encontradas, con detalles e imágenes.
  + Foro de adopción para facilitar la búsqueda de hogares para mascotas.
  + Módulo de donaciones integrado con una pasarela de pagos (WebPay).
  + Gestión de publicidad para mantener el sistema gratuito.
  + Compatibilidad con dispositivos móviles y navegadores web.
  + Sistema de almacenamiento de imágenes utilizando Firebase Storage.
  + Base de datos centralizada con MongoDB.
* **Excluido del sistema:**
  + Análisis avanzado de datos (considerado como mejora futura).
  + Integraciones con redes sociales para compartir publicaciones.
  + Notificaciones push personalizadas.
  1. **Audiencia**

Este documento está dirigido a los siguientes roles:

* **Desarrolladores:**  
  Personas encargadas de implementar las funcionalidades y la arquitectura técnica de PawFinder.
* **Testers:**  
  Equipos responsables de validar el correcto funcionamiento y la calidad del sistema mediante pruebas automatizadas y manuales.
* **Partes interesadas:**   
  Clientes, usuarios finales y colaboradores que proporcionan requisitos, retroalimentación y evaluación del sistema.
* **Líderes del proyecto:**

Responsables de supervisar y coordinar el desarrollo del sistema, asegurando que cumpla con los objetivos planteados**.**

* 1. **Definiciones y Acrónimos**
* **API (Application Programming Interface):** Interfaz que permite la comunicación entre el frontend y el backend.
* **CRUD:** Acrónimo para Create, Read, Update, Delete, que describe las operaciones básicas de gestión de datos.
* **E2E (End-to-End):** Pruebas que verifican el flujo completo de la aplicación desde el inicio hasta el fin.
* **Firebase Storage:** Herramienta de almacenamiento de Google utilizada para gestionar imágenes de forma externa.
* **MongoDB:** Base de datos NoSQL utilizada para almacenar datos del sistema de manera estructurada y escalable.
* **WebPay:** Pasarela de pagos utilizada para gestionar las donaciones dentro del sistema.

# **Visión General del Sistema**

* 1. **Descripción General**

PawFinder es un sistema diseñado para centralizar la información sobre mascotas perdidas, encontradas y en adopción, ofreciendo una plataforma accesible para conectar a los usuarios que buscan o encuentran animales. La aplicación web y móvil permite a los usuarios registrar mascotas, buscar publicaciones, realizar donaciones y acceder a un foro de adopción. Además, incluye un módulo de publicidad para garantizar que la plataforma sea gratuita para todos los usuarios.

* 1. **Contexto del Sistema**

Actualmente, la búsqueda de mascotas perdidas y en adopción depende de plataformas dispersas como Facebook, Instagram o grupos locales, lo que genera problemas de acceso a información centralizada. PawFinder busca solucionar este problema integrando toda la información en una única plataforma eficiente.

1. El sistema:

* Complementa el uso de redes sociales al ofrecer una solución específica para mascotas.
* Aprovecha tecnologías modernas como Firebase Storage, MongoDB y APIs escalables para garantizar una experiencia fluida.
* Se posiciona como una herramienta clave para organizaciones animalistas, rescatistas y dueños de mascotas al proporcionar un espacio dedicado exclusivamente a necesidades relacionadas con animales.

1. Sistemas relacionados:

* Redes sociales (para compartir publicaciones directamente desde PawFinder).
* Pasarelas de pago como WebPay (para gestionar donaciones).
* Bases de datos (MongoDB) y almacenamiento en la nube (Firebase Storage) para garantizar accesibilidad y seguridad de la información.
  1. **Usuarios del Sistema**

PawFinder está diseñado para atender diferentes perfiles de usuarios, agrupados en los siguientes roles principales:

|  |  |  |
| --- | --- | --- |
| Tipo de Usuario del Sistema | Perfil | Rol |
| Usuarios Generales | Dueños de mascotas, rescatistas, y personas interesadas en adoptar. | Registrar mascotas perdidas o encontradas. |
| Buscar publicaciones de mascotas en adopción o reportadas. |
| Realizar donaciones para apoyar iniciativas animalistas. |
| Administradores | Miembros del equipo responsable de mantener y gestionar la plataforma. | Supervisar y moderar publicaciones. |
| Gestionar el contenido publicitario |
| Resolver problemas técnicos y atender consultas de usuarios |
| Anunciantes | Empresas o profesionales interesados en promover sus productos o servicios relacionados con mascotas | Publicar anuncios dentro de la plataforma para generar ingresos que mantengan gratuita la app. |
| Organizaciones Animalistas | Grupos dedicados al rescate y adopción de animales. | Utilizar el foro de adopciones para promover animales rescatados. |
| Gestionar campañas de donación directamente desde la plataforma. |

# **Análisis de Requerimientos De Software**

* 1. **ERS**

Este documento proporciona una introducción a la Especificación de Requisitos Software (ERS) del proyecto PawFinder, detallando el propósito, el ámbito del sistema, definiciones, referencias y una visión general del documento.

* 1. **Resumen Ejecutivo**

El presente documento ERS tiene como propósito definir las especificaciones funcionales, no funcionales y del sistema para la implementación de PawFinder, una aplicación web móvil. Esta aplicación permitirá a los usuarios registrar y buscar mascotas perdidas, reportar mascotas encontradas, realizar adopciones y gestionar donaciones, proporcionando una plataforma centralizada y accesible tanto para usuarios finales como para administradores del sistema.

* 1. **Ámbito**

El diseño, desarrollo e implementación del sistema PawFinder será una aplicación web móvil que permitirá la gestión centralizada de información sobre mascotas perdidas y encontradas. Su objetivo principal es facilitar la conexión entre personas que han perdido mascotas y quienes las han encontrado, promoviendo la adopción responsable y fomentando la colaboración comunitaria.

Procesos que soporta la aplicación:

1. Registro de mascotas perdidas.
2. Registro de mascotas encontradas.
3. Consulta y búsqueda de mascotas.
4. Administración de usuarios del sistema.
5. Foro de adopciones.
6. Donaciones y generación de estadísticas.
7. Asignación de responsables para validar publicaciones.
8. Generación de reportes de actividad y métricas.
   1. **Definiciones, Acrónimos y Abreviaturas**

(a) Del Negocio

* Mascotas perdidas: Registro de animales cuyo paradero es desconocido por sus dueños.
* Mascotas encontradas: Animales reportados por usuarios que no son sus dueños.
* Administrador: Usuario del sistema con el rol más alto, encargado de la administración global de la plataforma.
* Moderador: Usuario responsable de validar y revisar publicaciones y reportes.
* Usuario final: Persona que utiliza el sistema para registrar, buscar o adoptar mascotas.

(b) Del Sistema

* Administrar: Acción de agregar, modificar, eliminar y consultar información de un objeto o usuario.
* Visitante: Persona que navega o utiliza PawFinder sin autenticación.
* Usuario autenticado: Persona que accede a la plataforma mediante autenticación con nombre de usuario y contraseña.
* Rol: Conjunto de permisos asignados a un usuario.
* Permiso: Derecho asignado para realizar una acción específica dentro del sistema.
* Pruebas: Actividades destinadas a verificar la funcionalidad y calidad del sistema.

(c) De Tecnología

* PawFinder: Nombre del sistema desarrollado.
* URL: Localizador uniforme de recursos utilizado para acceder a páginas y recursos en línea.
* HTML, CSS, JavaScript: Lenguajes clave en la construcción del frontend de la aplicación.
* MongoDB: Sistema de gestión de base de datos utilizado para el almacenamiento de información.
* FastAPI: Framework utilizado para el desarrollo del backend del sistema.
* Firebase Storage: Herramienta para el almacenamiento de imágenes externas.
* Encriptar: Proceso de codificación para proteger la información.
  1. **Perspectiva del Producto**

El sistema PawFinder será un producto diseñado para entornos web y móvil, permitiendo su acceso desde cualquier dispositivo con conexión a internet, incluyendo computadoras de escritorio, portátiles, tabletas y teléfonos móviles.

Actores principales del sistema:

* Administrador: Encargado de la gestión general de la plataforma, revisión de usuarios y publicaciones
* Usuarios finales: Personas que registran y buscan mascotas, participan en foros de adopción o realizan donaciones.

La aplicación busca ser un puente entre la comunidad para fomentar la recuperación de mascotas y facilitar procesos de adopción responsables, aprovechando la accesibilidad desde dispositivos móviles para un uso más amplio y eficaz.

* 1. **Requerimientos de Funcionales y No Funcionales**

| Tipo de Requerimiento | Nombre del Requerimiento | Descripción |
| --- | --- | --- |
| Funcional | Registro y Autenticación de Usuarios | Permite a los usuarios registrarse e iniciar sesión mediante correo y contraseña o cuentas de Google y Apple. |
| Funcional | Gestión de Mascotas Perdidas y Encontradas | Permite a los usuarios publicar anuncios de mascotas perdidas o encontradas con detalles relevantes. |
| Funcional | Búsqueda y Filtro de Mascotas | Facilita la búsqueda de mascotas con filtros por ubicación, raza o fecha de pérdida. |
| Funcional | Módulo de Adopciones | Ofrece un foro donde se pueden publicar y consultar mascotas disponibles para adopción. |
| Funcional | Módulo de Donaciones | Permite recibir donaciones mediante una pasarela de pago, con montos entre $5 y $200. |
| Funcional | Reportes y Análisis de Datos | Proporciona reportes y patrones de pérdida o adopción basados en datos de usuarios. |
| Funcional | Notificaciones | Envia notificaciones push sobre actualizaciones relevantes a los usuarios. |
| Funcional | Administración de Usuarios y Moderación de Contenidos | Permite moderar anuncios reportados y gestionar usuarios. |
| No Funcional | Escalabilidad | El sistema debe ser capaz de escalar a medida que aumenta el número de usuarios y publicaciones. |
| No Funcional | Disponibilidad | La aplicación debe estar disponible el 99.9% del tiempo. |
| No Funcional | Seguridad | Autenticación segura con Firebase y encriptación de datos sensibles. |
| No Funcional | Rendimiento | El tiempo de respuesta debe ser inferior a 2 segundos bajo carga normal. |
| No Funcional | Compatibilidad | La aplicación debe ser compatible con navegadores modernos y dispositivos móviles. |
| No Funcional | Usabilidad | Interfaz intuitiva y diseño responsivo para mejorar la experiencia del usuario. |
| No Funcional | Mantenibilidad | El sistema debe ser fácil de mantener y actualizar, con código modular y documentado. |
| No Funcional | Capacidad de Almacenamiento | Debe soportar al menos 1 TB de almacenamiento para datos e imágenes. |
| No Funcional | Localización | Inicialmente en español, con posibilidad de añadir otros idiomas. |
| No Funcional | Cumplimiento Legal | Cumple con normativas de protección de datos como GDPR y leyes locales. |
| Funcional | Recuperación de Contraseña | Permite a los usuarios recuperar su contraseña a través de un correo de recuperación. |
| Funcional | Geolocalización en Tiempo Real | Muestra la ubicación aproximada de mascotas perdidas utilizando geolocalización. |
| Funcional | Comentarios y Valoraciones en Adopciones | Permite a los usuarios dejar comentarios y valoraciones sobre las experiencias de adopción. |
| Funcional | Historial de Mascotas Perdidas | Proporciona acceso a un historial de mascotas reportadas como perdidas por cada usuario. |
| No Funcional | Backup Automático | El sistema debe realizar copias de seguridad automáticas de la base de datos diariamente. |
| No Funcional | Integración Continua | Automatización de pruebas y despliegues mediante herramientas de integración continua (CI/CD). |
| No Funcional | Tiempo de Inactividad Programado | Realización de actualizaciones importantes fuera de horas pico, notificando previamente a los usuarios. |
| No Funcional | Soporte Multiplataforma | La aplicación debe funcionar en dispositivos móviles y navegadores de escritorio sin pérdida de funcionalidad. |

# **Restricciones**

* 1. **Tecnológicas:**
  2. **Compatibilidad del Sistema:**

El sistema debe ser compatible únicamente con navegadores modernos (Google Chrome, Mozilla Firefox, Microsoft Edge) y dispositivos móviles con sistemas operativos iOS y Android. No se garantiza funcionalidad en navegadores desactualizados.

* 1. **Capacidad de Almacenamiento:**

El manejo de imágenes estará limitado a los estándares establecidos por Firebase Storage, lo que podría restringir el tamaño y cantidad de imágenes subidas por los usuarios.

* 1. **Escalabilidad Inicial:**

El sistema está diseñado para soportar hasta 1,000 usuarios concurrentes durante el lanzamiento inicial. Crecimientos adicionales requerirán optimización de la arquitectura.

* 1. **Presupuesto:**
  2. **Recursos Financieros Limitados:**

Con un presupuesto total de **$5.750.000 CLP**, la implementación debe priorizar funcionalidades clave. Esto excluye mejoras futuras, como integración de notificaciones push o análisis avanzado de datos.

* 1. **Regulaciones Legales:**
  2. **Protección de Datos Personales:**

La plataforma debe cumplir con la Ley N° 19.628 sobre Protección de la Vida Privada en Chile, asegurando la privacidad de los datos personales de los usuarios.

* 1. **Normas de Comercio Electrónico:**

Las transacciones a través de la pasarela de pagos WebPay deben cumplir con las regulaciones locales de seguridad financiera.

* 1. **Uso de Imágenes:**

Los usuarios deben aceptar términos y condiciones que limiten el uso indebido de imágenes o contenido compartido en la plataforma.

* 1. **Restricciones Operativas:**
  2. **Dependencia de Servicios Externos:**

El sistema depende de Firebase Storage para almacenamiento de imágenes y WebPay para gestionar donaciones. Interrupciones en estos servicios afectarían la operatividad del sistema.

* 1. **Soporte Técnico:**

Actualmente, el equipo no proporcionará soporte técnico en tiempo real a los usuarios finales.

* 1. **Restricciones Temporales:**
  2. **Cronograma Ajustado:**

Todas las fases deben completarse en un plazo de 18 semanas, lo que podría limitar la posibilidad de incluir funcionalidades adicionales o realizar pruebas exhaustivas más allá de lo planeado.

# **Suposiciones y Dependencias**

* 1. **Suposiciones**

Estas son las premisas asumidas durante el diseño e implementación del sistema, que se consideran verdaderas para el correcto desarrollo del proyecto:

* + 1. **Suposiciones sobre los Usuarios:**
  + Los usuarios finales estarán familiarizados con el uso de aplicaciones móviles y navegadores web modernos.
  + Los usuarios proporcionarán información válida y precisa al registrarse o publicar en la plataforma.
  + Los usuarios cumplirán con los términos y condiciones del sistema, incluyendo el respeto a la privacidad y propiedad intelectual de las imágenes subidas.
    1. **Suposiciones Técnicas:**
  + Los servicios externos como Firebase Storage y WebPay estarán disponibles y operativos durante todo el desarrollo y despliegue del proyecto.
  + La conectividad a internet será estable para garantizar una experiencia fluida tanto en el uso como en el acceso a los datos.
    1. **Suposiciones de Desarrollo:**
  + Los requisitos funcionales y no funcionales definidos son suficientes para satisfacer las necesidades de los usuarios y las partes interesadas.
  + Las pruebas realizadas serán representativas de los casos de uso reales de los usuarios.
  + El equipo de desarrollo tendrá acceso continuo a las herramientas y recursos necesarios durante todo el ciclo del proyecto.
  1. **Dependencias**

Estas son las dependencias externas e internas que el sistema necesita para su correcto funcionamiento:

* + 1. **Dependencias Tecnológicas:**
  1. **Firebase Storage:**
* Servicio externo para el almacenamiento y recuperación de imágenes de mascotas.
  1. **MongoDB Atlas:**
* Base de datos en la nube para gestionar usuarios, publicaciones y transacciones de manera escalable.
  1. **WebPay (Transbank):**
* Pasarela de pagos para gestionar donaciones y transacciones financieras.
  1. **Frameworks y Herramientas de Desarrollo:**
* FastAPI para la creación del backend.
* Ionic y Angular para el desarrollo del frontend.
* Swagger UI para pruebas e integración de la API.
* Cypress para pruebas end-to-end.
  + 1. **Dependencias Operativas:**
  1. **Servidores Cloud:**
* Infraestructura para alojar el backend y la base de datos.
  1. **Proveedores de Servicios de Internet (ISP):**
* La funcionalidad del sistema depende de conexiones estables a internet para acceder a Firebase, MongoDB y otros servicios externos.
  + 1. **Dependencias de Equipo:**
* Disponibilidad y coordinación de los desarrolladores y testers asignados al proyecto.
* Retroalimentación de los usuarios finales para validar funcionalidades y experiencia de usuario.
  + 1. **Dependencias Legales y Normativas:**
* Cumplimiento de la Ley N° 19.628 sobre Protección de la Vida Privada para proteger la información de los usuarios.
* Aceptación de términos y condiciones por parte de los usuarios para el uso de la plataforma.

# **Diagramas**

* 1. **Diagramas de casos de uso:**

|  |  |
| --- | --- |
| Buscar mascota perdida | |
| Actor Principal | Usuario Registrado |
| Pre-Condición | El usuario debe haber iniciado sesión y debe haber publicaciones en el sistema |
| Flujo Principal | 1. Seleccionar “Mascotas perdidas” 2. Utilizar filtros para la búsqueda (raza, ubicación, fecha) 3. El sistema muestra una lista de mascotas según filtros 4. El usuario accede a la información detallada del reporte |

|  |  |
| --- | --- |
| Publicar Mascota adopción | |
| Actor Principal | Usuario Registrado |
| Pre-Condición | El usuario debe haber iniciado sesión y debe haber publicaciones en el sistema |
| Flujo Principal | 1. Seleccionar “Poner en Adopción” 2. Rellenar Información de la mascota a adoptar 3. El sistema valida que el formulario fue correctamente llenado 4. El sistema confirma la creación del registro al usuario |

|  |  |
| --- | --- |
| Buscar Mascota adopción | |
| Actor Principal | Usuario Registrado |
| Pre-Condición | El usuario debe haber iniciado sesión y debe haber publicaciones en el sistema |
| Flujo Principal | 1. Seleccionar “Adopción” 2. El sistema retorna una lista de las mascotas en adopción generados por distintos usuarios. 3. El usuario accede a la publicación de la mascota para mayores detalles |

|  |  |
| --- | --- |
| Eliminar publicación propia | |
| Actor Principal | Usuario Registrado |
| Pre-Condición | El usuario debe haber iniciado sesión y debe tener publicaciones propias en el sistema |
| Flujo Principal | 1. Seleccionar Perfil 2. Seleccionar el tipo de registro a borrar (mascota perdida o mascota en adopción) 3. El usuario elige la publicación a eliminar 4. El sistema elimina la publicación usando la API y actualizando la base de datos |

# **Vista Lógica**

* 1. **Diagrama de Clases**

**Texto

Descripción generada automáticamente**

* 1. **Diagrama de Secuencia (Proceso Registrar mascota perdida)**

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente con confianza mediaTabla

Descripción generada automáticamente con confianza mediaTexto

Descripción generada automáticamente con confianza media

* 1. **Diagrama de actividad (publicación mascota perdida)** Diagrama

     Descripción generada automáticamente

# **Vista de Despliegue**

**8.1 Diagrama de Componentes**

Diagrama

Descripción generada automáticamente

# **Vista Física**

Diagrama

Descripción generada automáticamente

# **Casos de Uso**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso ID: | CU01 | | | |
| Nombre: | Registrar información de mascota perdida | | | |
| Creado por: | Javier Villalobos | Nombre del último que lo actualizó: | | Jorge Villalobos |
| Fecha de Creación: | 2024-01-01 | Fecha de la última actualización: | | 2024-01-01 |
| Actor: | Usuario | | | |
| Descripción: | Registrar información de mascota perdida | | | |
| Precondiciones: | Usuario registrado | | | |
| Postcondiciones: | Mascota perdida registrada | | | |
| Prioridad: | Alta | | | |
| Frecuencia de uso: | N veces | | | |
| Curso normal de eventos: | Acción del actor | | Respuesta del sistema | |
| El usuario ingresa los datos de la mascota | | El sistema registra la mascota | |
|  | |  | |
|  | |  | |
|  | |  | |
|  | |  | |
| Cursos Alternativos: | Datos incompletos -> Error de validación | | | |
| Excepciones: |  | | | |
| Includes: |  | | | |
| Extensión: |  | | | |
| Requerimientos especiales: |  | | | |
| Supuestos: | - Manejo básico de parte del usuario en el ámbito computacional e Internet. | | | |
| Notas y asuntos: |  | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso ID: | CU02 | | | |
| Nombre: | Reportar información de mascota encontrada | | | |
| Creado por: | Javier Villalobos | Nombre del último que lo actualizó: | | Jorge Villalobos |
| Fecha de Creación: | 2024-01-01 | Fecha de la última actualización: | | 2024-01-01 |
| Actor: | Usuario | | | |
| Descripción: | Reportar información de mascota encontrada | | | |
| Precondiciones: | Usuario registrado | | | |
| Postcondiciones: | Mascota encontrada reportada | | | |
| Prioridad: | Alta | | | |
| Frecuencia de uso: | N veces | | | |
| Curso normal de eventos: | Acción del actor | | Respuesta del sistema | |
| El usuario sube los datos de la mascota encontrada | | El sistema registra el reporte | |
|  | |  | |
|  | |  | |
|  | |  | |
|  | |  | |
| Cursos Alternativos: | Sin conexión -> Error al cargar | | | |
| Excepciones: |  | | | |
| Includes: |  | | | |
| Extensión: |  | | | |
| Requerimientos especiales: |  | | | |
| Supuestos: | - Manejo básico de parte del usuario en el ámbito computacional e Internet. | | | |
| Notas y asuntos: |  | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso ID: | CU03 | | | |
| Nombre: | Buscar mascotas perdidas por filtros | | | |
| Creado por: | Javier Villalobos | Nombre del último que lo actualizó: | | Jorge Villalobos |
| Fecha de Creación: | 2024-01-03 | Fecha de la última actualización: | | 2024-01-01 |
| Actor: | Usuario | | | |
| Descripción: | Buscar mascotas perdidas por filtros | | | |
| Precondiciones: | Usuario registrado | | | |
| Postcondiciones: | Mascotas perdidas listadas | | | |
| Prioridad: | Alta | | | |
| Frecuencia de uso: | N veces | | | |
| Curso normal de eventos: | Acción del actor | | Respuesta del sistema | |
| El usuario selecciona filtros | | El sistema muestra resultados | |
|  | |  | |
|  | |  | |
|  | |  | |
|  | |  | |
| Cursos Alternativos: | Sin resultados -> Lista Vacía | | | |
| Excepciones: |  | | | |
| Includes: |  | | | |
| Extensión: |  | | | |
| Requerimientos especiales: |  | | | |
| Supuestos: | - Manejo básico de parte del usuario en el ámbito computacional e Internet. | | | |
| Notas y asuntos: |  | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso ID: | CU04 | | | |
| Nombre: | Publicar y buscar adopciones | | | |
| Creado por: | Javier Villalobos | Nombre del último que lo actualizó: | | Jorge Villalobos |
| Fecha de Creación: | 2024-01-03 | Fecha de la última actualización: | | 2024-01-01 |
| Actor: | Usuario | | | |
| Descripción: | Publicar y buscar adopciones | | | |
| Precondiciones: | Usuario registrado | | | |
| Postcondiciones: | Mascotas para adopción publicadas | | | |
| Prioridad: | Media | | | |
| Frecuencia de uso: | Ocasional | | | |
| Curso normal de eventos: | Acción del actor | | Respuesta del sistema | |
| El usuario publica una mascota | | El sistema confirma la publicación | |
|  | |  | |
|  | |  | |
|  | |  | |
|  | |  | |
| Cursos Alternativos: | Error en la publicación -> Mensaje al usuario | | | |
| Excepciones: |  | | | |
| Includes: |  | | | |
| Extensión: |  | | | |
| Requerimientos especiales: |  | | | |
| Supuestos: | - Manejo básico de parte del usuario en el ámbito computacional e Internet. | | | |
| Notas y asuntos: |  | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso ID: | CU05 | | | |
| Nombre: | Gestionar cuentas de usuario | | | |
| Creado por: | Javier Villalobos | Nombre del último que lo actualizó: | | Jorge Villalobos |
| Fecha de Creación: | 2024-01-03 | Fecha de la última actualización: | | 2024-01-01 |
| Actor: | Administrador | | | |
| Descripción: | Gestionar cuentas de usuario | | | |
| Precondiciones: | Administrador autenticado | | | |
| Postcondiciones: | Cuenta gestionada | | | |
| Prioridad: | Media | | | |
| Frecuencia de uso: | Ocasional | | | |
| Curso normal de eventos: | Acción del actor | | Respuesta del sistema | |
| El administrador selecciona una cuenta | | El sistema realiza la acción | |
|  | |  | |
|  | |  | |
|  | |  | |
|  | |  | |
| Cursos Alternativos: | Usuario no encontrado -> Error | | | |
| Excepciones: |  | | | |
| Includes: |  | | | |
| Extensión: |  | | | |
| Requerimientos especiales: |  | | | |
| Supuestos: | - Manejo básico de parte del usuario en el ámbito computacional e Internet. | | | |
| Notas y asuntos: |  | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso ID: | CU06 | | | |
| Nombre: | Realizar donaciones a la plataforma | | | |
| Creado por: | Javier Villalobos | Nombre del último que lo actualizó: | | Jorge Villalobos |
| Fecha de Creación: | 2024-01-03 | Fecha de la última actualización: | | 2024-01-01 |
| Actor: | Usuario | | | |
| Descripción: | Realizar donaciones a la plataforma | | | |
| Precondiciones: | Usuario registrado | | | |
| Postcondiciones: | Donación realizada | | | |
| Prioridad: | Alta | | | |
| Frecuencia de uso: | Ocasional | | | |
| Curso normal de eventos: | Acción del actor | | Respuesta del sistema | |
| El usuario elige el monto y realiza el pago | | El sistema confirma la donación | |
|  | |  | |
|  | |  | |
|  | |  | |
|  | |  | |
| Cursos Alternativos: | Error en el pago -> Transacción fallida | | | |
| Excepciones: |  | | | |
| Includes: |  | | | |
| Extensión: |  | | | |
| Requerimientos especiales: |  | | | |
| Supuestos: | - Manejo básico de parte del usuario en el ámbito computacional e Internet. | | | |
| Notas y asuntos: |  | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso ID: | CU7 | | | |
| Nombre: | Actualizar información de perfil | | | |
| Creado por: | Javier Villalobos | Nombre del último que lo actualizó: | | Jorge Villalobos |
| Fecha de Creación: | 2024-01-03 | Fecha de la última actualización: | | 2024-01-01 |
| Actor: | Usuario | | | |
| Descripción: | Actualizar información de perfil | | | |
| Precondiciones: | Usuario registrado | | | |
| Postcondiciones: | Perfil actualizado | | | |
| Prioridad: | Media | | | |
| Frecuencia de uso: | Frecuente | | | |
| Curso normal de eventos: | Acción del actor | | Respuesta del sistema | |
| El usuario edita su perfil | | El sistema actualiza datos | |
|  | |  | |
|  | |  | |
|  | |  | |
|  | |  | |
| Cursos Alternativos: | Error al guardar -> Reintentar | | | |
| Excepciones: |  | | | |
| Includes: |  | | | |
| Extensión: |  | | | |
| Requerimientos especiales: |  | | | |
| Supuestos: | - Manejo básico de parte del usuario en el ámbito computacional e Internet. | | | |
| Notas y asuntos: |  | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso ID: | CU8 | | | |
| Nombre: | Subir y gestionar fotos de mascotas | | | |
| Creado por: | Javier Villalobos | Nombre del último que lo actualizó: | | Jorge Villalobos |
| Fecha de Creación: | 2024-01-03 | Fecha de la última actualización: | | 2024-01-01 |
| Actor: | Usuario | | | |
| Descripción: | Subir y gestionar fotos de mascotas | | | |
| Precondiciones: | Usuario registrado | | | |
| Postcondiciones: | Foto gestionada | | | |
| Prioridad: | Media | | | |
| Frecuencia de uso: | Frecuente | | | |
| Curso normal de eventos: | Acción del actor | | Respuesta del sistema | |
| El usuario sube una foto | | El sistema almacena la foto | |
|  | |  | |
|  | |  | |
|  | |  | |
|  | |  | |
| Cursos Alternativos: | Error al subir imagen -> Reintentar | | | |
| Excepciones: |  | | | |
| Includes: |  | | | |
| Extensión: |  | | | |
| Requerimientos especiales: |  | | | |
| Supuestos: | - Manejo básico de parte del usuario en el ámbito computacional e Internet. | | | |
| Notas y asuntos: |  | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso ID: | CU9 | | | |
| Nombre: | Consultar historial de mascotas reportadas | | | |
| Creado por: | Javier Villalobos | Nombre del último que lo actualizó: | | Jorge Villalobos |
| Fecha de Creación: | 2024-01-03 | Fecha de la última actualización: | | 2024-01-01 |
| Actor: | Usuario | | | |
| Descripción: | Consultar historial de mascotas reportadas | | | |
| Precondiciones: | Usuario registrado | | | |
| Postcondiciones: | Historial consultado | | | |
| Prioridad: | Baja | | | |
| Frecuencia de uso: | Ocasional | | | |
| Curso normal de eventos: | Acción del actor | | Respuesta del sistema | |
| El usuario selecciona una mascota | | El sistema muestra el historial | |
|  | |  | |
|  | |  | |
|  | |  | |
|  | |  | |
| Cursos Alternativos: | Historial vacío -> Mensaje informativo | | | |
| Excepciones: |  | | | |
| Includes: |  | | | |
| Extensión: |  | | | |
| Requerimientos especiales: |  | | | |
| Supuestos: | - Manejo básico de parte del usuario en el ámbito computacional e Internet. | | | |
| Notas y asuntos: |  | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso ID: | CU10 | | | |
| Nombre: | Ofrecer recompensas por mascotas | | | |
| Creado por: | Javier Villalobos | Nombre del último que lo actualizó: | | Jorge Villalobos |
| Fecha de Creación: | 2024-01-03 | Fecha de la última actualización: | | 2024-01-01 |
| Actor: | Usuario | | | |
| Descripción: | Ofrecer recompensas por mascotas | | | |
| Precondiciones: | Publicación existente | | | |
| Postcondiciones: | Recompensa ofrecida | | | |
| Prioridad: | Media | | | |
| Frecuencia de uso: | Ocasional | | | |
| Curso normal de eventos: | Acción del actor | | Respuesta del sistema | |
| El usuario añade recompensa | | El sistema actualiza la publicación | |
|  | |  | |
|  | |  | |
|  | |  | |
|  | |  | |
| Cursos Alternativos: |  | | | |
| Excepciones: |  | | | |
| Includes: |  | | | |
| Extensión: |  | | | |
| Requerimientos especiales: |  | | | |
| Supuestos: | - Manejo básico de parte del usuario en el ámbito computacional e Internet. | | | |
| Notas y asuntos: |  | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso ID: | CU11 | | | |
| Nombre: | Enviar sugerencias de mejora | | | |
| Creado por: | Javier Villalobos | Nombre del último que lo actualizó: | | Jorge Villalobos |
| Fecha de Creación: | 2024-01-03 | Fecha de la última actualización: | | 2024-01-01 |
| Actor: | Usuario | | | |
| Descripción: | Enviar sugerencias de mejora | | | |
| Precondiciones: | Usuario registrado | | | |
| Postcondiciones: | Feedback recibido | | | |
| Prioridad: | Baja | | | |
| Frecuencia de uso: | Ocasional | | | |
| Curso normal de eventos: | Acción del actor | | Respuesta del sistema | |
| El usuario envía feedback | | El sistema registra la sugerencia | |
|  | |  | |
|  | |  | |
|  | |  | |
|  | |  | |
| Cursos Alternativos: | Error en feedback -> Reintentar | | | |
| Excepciones: |  | | | |
| Includes: |  | | | |
| Extensión: |  | | | |
| Requerimientos especiales: |  | | | |
| Supuestos: | - Manejo básico de parte del usuario en el ámbito computacional e Internet. | | | |
| Notas y asuntos: |  | | | |

# **Diagrama de arquitectura**

**Diagrama

Descripción generada automáticamente**

# **Diagrama Base de Datos**

**Texto

Descripción generada automáticamente**

# **Diccionario de datos**

**USER**



**AdoptRegister**



**ReportPet**

**PET**



# **Criterios de Aceptación**

* 1. **Criterios Funcionales**

1. **Registro y Autenticación de Usuarios:**

* Los usuarios deben poder registrarse con información válida y autenticarse de forma segura mediante un sistema de inicio de sesión con validación de credenciales.
* La funcionalidad de recuperación de contraseñas debe estar operativa y ser efectiva.

1. **Publicaciones de Mascotas Perdidas y Encontradas:**

* Los usuarios deben poder crear, editar y eliminar publicaciones de mascotas con todos los campos necesarios (nombre, raza, ubicación, foto, etc.).
* Las publicaciones deben ser visibles para otros usuarios con opciones de búsqueda y filtrado por ubicación y características.

1. **Foro de Adopción:**

* Los usuarios deben poder acceder al foro para publicar mascotas en adopción o buscar mascotas disponibles.
* El sistema debe permitir el contacto directo entre usuarios interesados en adopciones.

1. **Módulo de Donaciones:**

* La pasarela de pagos (WebPay) debe permitir transacciones seguras y registrar correctamente las donaciones en la base de datos.
* Los usuarios deben recibir una confirmación de éxito o error tras cada intento de donación.

1. **Publicidad y Sostenibilidad:**

* El sistema debe integrar un módulo de publicidad para mostrar anuncios relacionados con mascotas o servicios afines.
  1. **Criterios de Usabilidad**

1. **Interfaz de Usuario (UI):**

* La interfaz debe ser intuitiva y accesible, adaptándose correctamente a dispositivos móviles y navegadores web.
* Los tiempos de carga no deben superar los 2 segundos en conexiones estándar.

1. **Experiencia de Usuario (UX):**

* Los flujos principales (registro, publicación, búsqueda, donaciones) deben ser claros y fáciles de completar.
  1. **Criterios Técnicos**

1. **Estabilidad del Sistema:**

* El sistema debe soportar al menos 1,000 usuarios concurrentes sin degradación significativa en el rendimiento.
* Todas las funcionalidades deben ser probadas y alcanzar una tasa de éxito del 100% en pruebas de integración y end-to-end.

1. **Seguridad:**

* Los datos sensibles (como contraseñas y transacciones) deben transmitirse cifrados.
* El acceso a funcionalidades protegidas debe estar restringido a usuarios autenticados.
  1. **Criterios de Documentación**

1. **Documentación Técnica:**

* Debe incluir manuales de usuario, administrador y desarrollador, además de una especificación técnica de la API.
* Los resultados de pruebas deben estar documentados y disponibles para revisión.

1. **Términos y Condiciones:**

* El sistema debe incluir términos y condiciones claros que sean aceptados por los usuarios al registrarse.
  1. **Referencias**

A continuación, se enumeran los documentos, estándares y recursos que se han utilizado como base para la creación de este documento de Especificación de Requisitos del Software (ERS) para PawFinder:

1. **Normativas y Estándares Legales:**

* **Ley N° 19.628 sobre Protección de la Vida Privada (Chile):**  
  Base legal para garantizar la privacidad y protección de los datos personales de los usuarios.

1. **Documentación Técnica:**

* **Guías de Firebase Storage:** Documentación oficial de Firebase sobre manejo de almacenamiento de imágenes y datos.
* **Documentación oficial de MongoDB:** Guía para diseño y gestión de bases de datos NoSQL escalables.
* **Swagger UI:** Referencias técnicas para la implementación y pruebas de APIs.

1. **Estándares de Ingeniería de Software:**

* **IEEE 830-1998:** Recomendaciones para la especificación de requisitos de software.
* **ISO/IEC 25010:** Modelo de calidad de software utilizado para evaluar los requisitos no funcionales.

1. **Metodologías de Desarrollo:**

* **Agile Development Handbook:** Estrategias para definir requisitos y priorizar entregables.

1. **Recursos en Línea:**

* **Cypress Documentation:** Referencias para la implementación de pruebas end-to-end.
* **WebPay API Documentación:** Guía para integración de la pasarela de pagos en sistemas web.
* **Angular e Ionic Frameworks:** Documentación oficial para el desarrollo de interfaces de usuario adaptativas.

1. **Análisis de Proyectos Similares:**

* Plataformas como Sosafe y otras soluciones de búsqueda de mascotas fueron investigadas para identificar puntos fuertes y áreas de mejora.

1. **Reuniones y Talleres con las Partes Interesadas:**

* Información recopilada a partir de reuniones con clientes y usuarios finales para entender necesidades y expectativas.
  1. **Criterios de Aceptación**

**14.6.1 Condiciones Mínimas**

Para que el sistema **PawFinder** sea aceptado por los clientes y partes interesadas, debe cumplir con las siguientes condiciones:

1. **Funcionalidad Completa:**

* Permitir a los usuarios registrar mascotas perdidas y encontradas con detalles completos, incluyendo ubicación y fotos.
* Incluir un módulo funcional para adopciones, donde los usuarios puedan buscar y publicar mascotas disponibles.
* Ofrecer un sistema de donaciones totalmente integrado con WebPay, permitiendo transacciones seguras.
* Contar con una interfaz intuitiva y responsive, adaptada para dispositivos móviles y navegadores web.

1. **Rendimiento y Usabilidad:**

* El tiempo de carga de cualquier página no debe superar los 2 segundos en condiciones normales de conexión.
* El sistema debe soportar hasta 1,000 usuarios concurrentes en el lanzamiento sin pérdida de rendimiento.

1. **Pruebas Exitosas:**

* Obtener una tasa de éxito del 100% en todas las pruebas realizadas (end-to-end, integración y manuales).

1. **Cumplimiento de Seguridad:**

* Garantizar la protección de datos sensibles mediante cifrado y autenticación segura.
* Limitar el acceso a áreas protegidas únicamente a usuarios autenticados.

1. **Documentación y Soporte:**

* Proveer manuales técnicos para administradores y desarrolladores, además de guías de usuario.
* Incluir términos y condiciones claros aceptados por los usuarios al registrarse.

1. **Despliegue en Producción:**

* Asegurar que la aplicación esté completamente funcional y accesible desde el entorno de producción configurado correctamente.
  1. **Referencias**

**14.7.1 Documentos y Recursos Utilizados**

1. **Normativas Legales y Estándares:**

* **Ley N° 19.628 sobre Protección de la Vida Privada (Chile):**  
  Guía para garantizar la privacidad y protección de datos personales.
* **ISO/IEC 25010:** Modelo de calidad de software enfocado en usabilidad, seguridad y rendimiento.
* **IEEE 830-1998:** Estructura recomendada para la especificación de requisitos de software.

1. **Recursos Técnicos:**

* **Firebase Storage:** Documentación oficial para la configuración y gestión de almacenamiento en la nube.
* **MongoDB Atlas:** Manual para el diseño y uso de bases de datos NoSQL.
* **Swagger UI:** Referencias para la documentación y prueba de APIs.
* **WebPay API:** Documentación técnica para la integración de pasarelas de pago en aplicaciones.

1. **Metodologías y Herramientas de Desarrollo:**

* **Cypress Documentation:** Guías para la implementación de pruebas end-to-end.
* **Angular e Ionic Frameworks:** Recursos para el diseño de interfaces de usuario adaptativas.

1. **Investigación de Proyectos Similares:**

* Análisis de plataformas como Sosafe y grupos de redes sociales relacionados con mascotas.

1. **Reuniones y Talleres:**

* Datos obtenidos de reuniones con partes interesadas y usuarios finales para comprender necesidades específicas.

1. **Otras Referencias:**

* **Manual Agile Development:** Guía para priorizar entregables y manejar iteraciones en desarrollo de software.