

<Dame la Pata>

Software Requirements Specification

<Versión 1>

<13/4/2023>

Equipo N°1: Patitas

Integrantes: Babiachuk, Yaco

Blázquez, Máximo

Díaz, Alejo

Fernández, Mateo

Fini Minué, Marco

Genaro, Juan Pablo

Morais, Joaquín

Salonia, Agustín

Revision History

Date	Description	Author	Comments
12/4/2023	Versión 1	Mateo Fernández	First Revision -No se completó los datos que no teníamos conocimientos. Tampoco información todavía no analizada ej. Requerimientos Funcionales. Se quitó del índice a esos datos para una mayor comprensión. En posteriores versiones serán agregadas.

Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date

Table of Contents

REVISION HISTORY	II
DOCUMENT APPROVAL	II
1. INTRODUCTION	IV
1.1 PURPOSE	IV
1.2 SCOPE	IV
1.3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS	IV
1.5 OVERVIEW	IV
2. GENERAL DESCRIPTION	V
2.1 PRODUCT PERSPECTIVE	V
2.2 PRODUCT FUNCTIONS	V
2.3 USER CHARACTERISTICS	V
2.4 GENERAL CONSTRAINTS	V
2.5 ASSUMPTIONS AND DEPENDENCIES	V
3. SPECIFIC REQUIREMENTS	V
3.5 NON-FUNCTIONAL REQUIREMENTS	VII
3.5.1 Performance	vii
3.5.2 Reliability	vii
3.5.3 Availability	vii
3.5.4 Security	vii
3.5.5 Maintainability	vii
3.5.6 Portability	vii
3.6 DESIGN CONSTRAINTS	VII
3.7 LOGICAL DATABASE REQUIREMENTS	
A. APPENDICES (JUST IF IT IS NECESSARY)	VII
A.1 APPENDIX 1	VIII
A.2 APPENDIX 2	VIII

INTRODUCTION IV

Este documento presenta los requerimientos técnicos y funcionales que se deben cumplir para la creación de la página web de adopción de perros para la asociación Garra. Se describirán las funcionalidades específicas de la página web, los requerimientos técnicos para su desarrollo, así como los requerimientos de seguridad y privacidad necesarios para garantizar la protección de los usuarios y sus datos personales.

1.1 PURPOSE IV

La asociación Garra es una organización sin fines de lucro que se dedica a la protección de los animales y a la promoción de su bienestar. Garra ha estado trabajando en la adopción de perros durante varios años, y ha notado que el proceso de adopción puede ser complicado y desalentador para algunas personas. Para abordar este problema, Garra ha decidido crear una página web de adopción de perros que facilite el proceso de adopción y promueva la adopción responsable.

En resumen, este proyecto busca facilitar el proceso de adopción de perros y promover la adopción responsable mediante la creación de una página web eficiente y efectiva. Con la colaboración de la asociación Garra, se espera poder proporcionar una experiencia positiva para los usuarios y los perros disponibles para adopción.

1.2 SCOPE IV

El alcance del proyecto sería el desarrollo de una página web para la asociación Garra en Córdoba, Argentina, que permita la facilitación de la adopción de perros de manera eficiente y efectiva. La página web permitirá a los usuarios llamados “protector” publicar perros que se encuentran en condiciones de calle para que los usuarios “adoptantes” puedan buscar perros disponibles para adopción, proporcionando información detallada sobre cada animal, incluyendo su edad, tamaño, personalidad y otros detalles relevantes. Además, proporcionará información del adoptante en cuestión para posibilitar el acceso a datos relevantes para conocer si está en correctas condiciones para poder adoptarlo.

Los administradores podrán recibir críticas constructivas respecto a la manera de operar de la misma, como así contar con un sistema el cual provea los métodos necesarios para restringir a usuarios maliciosos.

La filtración tanto de mascotas para uso del adoptante como de las personas interesadas para el uso de la administración del refugio se diseñará de tal manera que haga la acción con el menor tiempo requerido y de diversas maneras de hacerlo(filtros).

Se podrá apoyar a la causa mediante donaciones, de manera segura y concreta para proveer confianza a la persona voluntaria que realiza esta acción.

1.3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS IV

Definiciones:

MySQL: lenguaje utilizado en bases de datos para consultar, actualizar y manejar bases de datos relacionales.

Python: lenguaje de programación usado para el desarrollo.

React: Framework de desarrollo Front-end

Flask: Framework de desarrollo Back-end

Railway: plataforma en línea que permite a los desarrolladores alojar y gestionar aplicaciones web y API en la nube de una manera simple y eficiente.

Responsive: capacidad de una página web de adaptarse a diferentes tamaños de pantalla y dispositivos, para que los usuarios puedan tener una experiencia de usuario óptima en cualquier dispositivo que utilicen para acceder a la página.

Navbar: barra horizontal que se ubica en la parte superior de la página o aplicación y contiene enlaces a diferentes secciones o páginas del sitio o aplicación.

API RESTful: estilo de arquitectura de software que se utiliza para crear servicios web. REST es un acrónimo de Representational State Transfer, que significa transferencia de estado representacional. RESTful API utiliza el protocolo HTTP para transmitir datos, lo que significa que puede ser utilizado por cualquier aplicación que tenga acceso a internet. Es un método de comunicación entre diferentes sistemas que permite a los usuarios acceder a datos y servicios desde cualquier dispositivo y en cualquier momento.

Acrónimos:

UML: Unified Modeling Language (Lenguaje de Modelado Unificado) -lenguaje de modelado visual que se utiliza para diseñar, representar y documentar sistemas de software complejos. Su objetivo es proporcionar una serie de diagramas gráficos que permiten a los desarrolladores de software describir de manera detallada los distintos aspectos del sistema, como su estructura, comportamiento y relaciones.-

SRS: Software Requirements Specifications (Especificación de Requerimientos de Software).

URL: Uniform Resource Locator (Localizador de Recursos Uniforme)
-cadena de caracteres que identifica de manera única una dirección en la web de un recurso.-

ER: Entidad Relación -tipo de diagrama se utiliza para representar las relaciones entre las entidades en una base de datos. Su objetivo es proporcionar una vista visual y clara de la estructura de la base de datos, y cómo las entidades se relacionan entre sí.-

Abreviaturas:

DB: Bases de datos

Ing: Ingeniero

UI: Interfaz de Usuario

RF: Requerimiento Funcional

RNF: Requerimiento No Funcional

1.5 OVERVIEW IV

El SRS está compuesto por las siguientes secciones:

- 1) Introducción: En ésta sección se detalla los objetivos que tiene el SRS y de nuestro sistema en forma general.
- 2) Descripción General: Describe una perspectiva general del producto a desarrollarse, como también las características del usuario y las limitaciones que podría tener. La información está orientada al cliente/usuario potencial.
- 3) Requerimientos Específicos: Muestra paso a paso todos los requerimientos que el usuario desea en el producto final. Para el cual se ha utilizado el “Prototipo 2 del Estándar IEEE 380”. Se emplean términos técnicos orientados principalmente a los desarrolladores y programadores.

GENERAL DESCRIPTION IV

2.1 PRODUCT PERSPECTIVE IV

El sistema funcionará en un ambiente web, por lo que deberá ser capaz de funcionar en cualquier plataforma con un navegador web instalado, siempre y cuando disponga de conexión al servidor en que va a estar corriendo la aplicación, ya sea por Internet o alguna red privada.

2.2 PRODUCT FUNCTIONS IV

El sistema permitirá:

Como Usuarios en la aplicación:

- Los usuarios podrán registrarse en el sistema proporcionando su nombre completo, dirección de correo electrónico, número de teléfono y contraseña.
- Una vez creada la cuenta, se podrá recurrir al inicio de sesión.
- Podrá cerrar la sesión en el momento que el usuario desee.
- Anular la solicitud de adopción.
- Recibir notificaciones.
- Realizar críticas para en el futuro mejorar la experiencia del cliente.
- Acceder a las redes sociales de la asociación.
- Acceder al papeleo necesario para llevar un fluido proceso de adopción.
- Los usuarios podrán buscar perros disponibles por características como edad, tamaño, etc.
- Podrá escoger el método de ordenamiento que más se adecue a sus necesidades (fecha de publicación, por ejemplo)
- Los usuarios podrán ver los perros disponibles con fotos y una descripción detallada.
- Los usuarios podrán guardar perros en una lista de "favoritos" para ver más tarde.
- Los usuarios "protectores" podrán publicar perros en condiciones de calle para la disponibilidad de los adoptantes.
- Formar una lista con usuarios que no desea que le aparezcan.
- Rechazar/confirmar la solicitud de adopción de los usuarios "adoptantes".
- Reportar a los usuarios que consideren que actúen de manera inadecuada.

Como Administrador en la aplicación:

- El refugio podrá acceder al historial de sus clientes.

- Podrá calcular la posibilidad de la adopción del usuario adoptante (hogar, comida entre otras cosas para el perro).
- Realizar una lista de aquellas personas inadecuadas que se hayan registrado en el sistema.

2.3 USER CHARACTERISTICS IV

El sistema contendrá 3 tipos de usuarios:

Como protector: Usuario en el cual su interacción con el programa se basará en facilitar y adecuar de manera dinámica el proceso de publicación del animal encontrado para su posterior publicación. El usuario puede no contar con conocimiento en software.

Como adoptante: Usuario con la intención de adoptar una mascota. Se entiende que es un cliente que cuenta con las capacidades para tener una mascota en su hogar. De la misma manera deberá contar con conocimiento básico en navegación web.

Como administrador: Personas encargadas de la administración de la página, el mismo deberá actuar a manera de garantizar una experiencia de usuario amigable tanto para el adoptante como el protector. Deberá tener un perfecto entendimiento del funcionamiento del sistema (atajos, funcionalidades, capacidad de respuesta). Se encargará de bloquear el acceso a usuarios maliciosos, la correcta publicación de las mascotas. verificar que el perro pueda gozar de una buena vida.

2.4 GENERAL CONSTRAINT IV

Restricciones presentes en el sistema:

Plazo de entrega: A confirmar en posteriores reuniones.

Presupuesto: Proyecto benéfico para apoyar a la causa de la adopción de perros en estado de calle. El equipo designado para el sistema se presenta como voluntario.

Recursos: el equipo de desarrollo estará limitado a un máximo de 8 miembros y se dispondrá de un servidor dedicado para la página web.

Donaciones: El medio para apoyar económicamente a la causa de la asociación va a ser Mercado pago.

2.5 ASSUMPTIONS AND DEPENDENCIES IV

Suposiciones:

Los usuarios de la página web estarán interesados en adoptar mascotas de refugios y organizaciones de rescate de animales.

Los usuarios tendrán acceso a un dispositivo con conexión a Internet y un navegador web compatible.

Los usuarios estarán dispuestos a proporcionar comentarios y retroalimentación sobre la página web y su funcionalidad.

Se supondrá que la asociación brinde información requerida en lo referido al papeleo necesario para cumplir el proceso de adopción.

Dependencias:

La página web dependerá de un servidor web para alojar y servir contenido a los usuarios.

La página web dependerá de una base de datos para almacenar y recuperar información de usuarios y animales disponibles para adopción.

SPECIFIC REQUIREMENTS V

3.1 EXTERNAL INTERFACE REQUIREMENTS VI

3.1.1 User Interfaces vi

El sistema debe proporcionar una pantalla de inicio de sesión para que los usuarios autenticuen su identidad antes de poder acceder a las funcionalidades del sistema. Al completar el acceso se encontrará en la página principal del mismo.

La navegabilidad de la página web consta de un navbar en la cual se encontrarán las diferentes secciones del sistema.

El sistema debe ser fácil de navegar para los usuarios novatos y avanzados. Por ejemplo, debe ser fácil para los usuarios novatos encontrar funciones comunes y para los usuarios avanzados personalizar la interfaz para sus necesidades específicas.

La interfaz debe ser accesible para todos los usuarios, incluyendo aquellos con discapacidades visuales o motoras. Por ejemplo, el sistema debe ser compatible con lectores de pantalla y dispositivos de entrada alternativos como teclados virtuales.

La interfaz debe ser intuitiva y fácil de entender. Por ejemplo, los botones, iconos y etiquetas deben ser claros y significativos para los usuarios.

El sistema debe ser consistente en toda la interfaz de usuario. Por ejemplo, los botones, iconos y etiquetas deben ser consistentes en todo el sistema para que los usuarios sepan qué esperar y cómo interactuar con el sistema en todo momento.

La interfaz debe ser atractiva y fácil de usar. Por ejemplo, debe utilizar una paleta de colores y fuentes coherentes y atractivas para que los usuarios se sientan cómodos al usar el sistema.

3.1.2 Hardware Interfaces vi

La página web no requiere interfaces de hardware específicas. Sin embargo, debe tener en cuenta las limitaciones de hardware y software de los dispositivos y navegadores web utilizados por los usuarios.

3.1.3 Software Interfaces vi

SO : macOS/ Windows: mayor al XP/Linux

Explorador : Garantizamos que mientras se encuentre con acceso a internet, podrá acceder a la página desde cualquier explorador.

3.1.4 Communications Interfaces vi

La página web debe ser capaz de comunicarse con diferentes sistemas, como servidores web, bases de datos y APIs. Debe ser compatible con los protocolos de comunicación y formatos de datos estándar utilizados por estos sistemas. Se trabajará mediante las URLs para redireccionar a otras páginas como por ejemplo las redes sociales.

3.5 NON-FUNCTIONAL REQUIREMENTS VII

3.5.1 Performance vii

La página web debe ser capaz de manejar un alto tráfico de usuarios y una gran cantidad de información de mascotas. Además, la página debe tener una velocidad de carga rápida para que los usuarios puedan acceder a la información de los perros en un tiempo mínimo.

El sistema debe ser capaz de responder a las solicitudes con el mínimo tiempo necesario.

La carga de la página debe ser inferior a 3 segundos.

3.5.2 Reliability vii

La página web debe funcionar correctamente y sin errores técnicos durante al menos un año.

Es importante asegurarse de que los datos de los perros estén actualizados y de que la información proporcionada por los usuarios sea veraz.

Ante posibles errores del servidor se contemplará una rápida resolución del problema de manera que no profundice el error.

3.5.3 Availability vii

La inactividad de la misma será para posibles mantenimientos.

La página web debe estar disponible el 99,9% del tiempo para que los usuarios puedan acceder a ella y encontrar perros disponibles para su adopción.

Los administradores del sistema tendrán acceso el 100% del tiempo.

3.5.4 Security vii

La página web debe cumplir con los estándares de seguridad de la industria y garantizar la protección de la información de los usuarios.

El inicio de sesión cuenta con un sistema de encriptación de contraseña.

3.5.5 Maintainability vii

La página web debe ser fácil de mantener y actualizar. Los administradores deben ser capaces de actualizar los datos de los perros y de los usuarios de manera sencilla y eficiente.

Se debe contar con una documentación clara y fácil de entender para que los administradores puedan realizar actualizaciones sin interrumpir el funcionamiento de la plataforma.

3.5.6 Portability:

La página web debe ser accesible al español. Aunque contará con la capacidad de traducir al inglés para lograr una mayor integridad con los usuarios que no manejan el idioma nativo de Argentina.

La página web debe ser compatible con diferentes sistemas operativos y navegadores web para que los usuarios puedan acceder a ella desde cualquier dispositivo y sistema.

3.6 DESIGN CONSTRAINTS VII

3.6.1 Tecnologías Usadas:

Flask es un marco de aplicación web de Python que se utiliza para desarrollar aplicaciones web y API RESTful. Si se desea utilizar Flask en el proyecto, es importante tener en cuenta que es compatible con una amplia gama de bases de datos, incluyendo MySQL. Esto significa que se podría utilizar Flask para desarrollar la parte de backend del proyecto, que estaría conectada a la base de datos MySQL.

Por otro lado, React es una biblioteca de JavaScript que se utiliza para desarrollar interfaces de usuario. React es compatible con Flask y se puede utilizar para desarrollar la parte de frontend del proyecto. Una de las ventajas de utilizar React es que es altamente escalable, lo que significa que se puede utilizar para desarrollar interfaces de usuario complejas y dinámicas para el proyecto.

Flask se puede utilizar para desarrollar la parte de backend del proyecto y conectarse a la base de datos MySQL, mientras que React se puede utilizar para desarrollar la parte de frontend del proyecto. Al elegir estas tecnologías, se pueden satisfacer las restricciones de diseño y garantizar que el proyecto se desarrolle de manera eficiente y efectiva.

3.6.2 Arquitectura del software:

Se desea que la aplicación tenga una arquitectura cliente-servidor, con la mayor parte de la lógica del negocio en el servidor y una interfaz de usuario en el cliente.

3.6.3 Compatibilidad con otras aplicaciones:

La aplicación debe ser compatible con los sistemas de pago en línea populares, como PayPal o mercado pago, para permitir a los usuarios hacer las donaciones correspondientes para apoyar a la asociación.

APPENDICES

Appendix 1

Reuniones: Estas reuniones suelen ser la forma en que el equipo de desarrollo se comunica con el cliente para entender sus necesidades y expectativas, y para mantenerlos informados sobre el progreso del proyecto.

En estas reuniones, el equipo de desarrollo y el cliente pueden discutir aspectos como los requisitos del software, el alcance del proyecto, los plazos, el presupuesto y cualquier otro tema relacionado con el proyecto de software en cuestión. También se pueden presentar prototipos o demostraciones del software para recibir retroalimentación y comentarios del cliente.

Appendix 2

Metodología de trabajo: El equipo de desarrollo trabaja mediante la metodología ágil scrum, la cual se enfoca en la entrega continua de productos de alta calidad, a través de un proceso iterativo e incremental. Se confecciona el product backlog (lista priorizada de todas las funcionalidades o características que se desean en el producto a desarrollar). Para luego trabajar mediante sprint (ciclo de trabajo que dura de 2 a 4 semanas en el que se desarrolla un conjunto de funcionalidades priorizadas del Product Backlog).