

DUOCUC

Especificación de requisitos de software

Proyecto: Waggy PetShop

Revisión: 1.0

15/07/2024

Contenido

Ficha del documento	3
1. Introducción	4
1.1. Propósito	4
1.2. Ámbito del Sistema	4
1.3. Definiciones, Acrónimos y Abreviaturas	4
1.4. Referencias	4
1.5. Visión General del Documento	4
2. Descripción General	5
2.1. Perspectiva del Producto	5
2.2. Funciones del Producto	5
2.3. Características de los Usuarios	5
2.4. Restricciones	5
2.5. Suposiciones y Dependencias	5
2.6. Requisitos Futuros	5
3. Requisitos Específicos	6
3.1. Interfaces Externas	6
3.2. Funciones	6
3.3. Requisitos de Rendimiento	6
3.4. Restricciones de Diseño	6
3.5. Atributos del Sistema	6
3.6. Otros Requisitos	6

Ficha del documento

Fecha	Revisión	Autor	Modificación
07-06-24	OK	Benjamín	Mockup
14-06-23	OK	Iván	Requisitos no funcionales

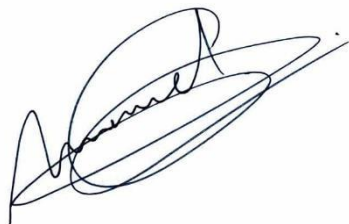
Integrantes:

Nombre Integrante del Equipo	Rol Definido
Juan Godoy	Administrador de la configuración del Proyecto.
Benjamín González	Diseñador (Genera solicitud del proyecto y sus características).
Iván Fernandez	Ingeniero Software/Analista de sistema (Genera información, requisitos y crea el diseño de la replicación de datos).

Documento validado por las partes en fecha: 15/07/2024

Por el cliente

Por la empresa suministradora




Sr./Sra.

Sr./Sra.

1. Introducción

Este documento es una especificación de requisitos de software para el sistema de Waggys. Todo el contenido del siguiente documento ha sido elaborado por el grupo subcontratado. La especificación se ha estructurado siguiendo la norma estándar IEEE830.

1.1. Propósito

El propósito de este documento es detallar los requisitos funcionales y no funcionales para el desarrollo de una aplicación web destinada a la venta de productos para mascotas. Esta aplicación tiene como objetivo mejorar el proceso de ventas y logística, así como facilitar la gestión de donaciones a una fundación sin fines de lucro

1.2. Ámbito del Sistema

La aplicación web permitirá a los usuarios registrados realizar compras en línea de productos para mascotas, generar listas de compras, monitorear sus compras, y realizar donaciones a una fundación sin fines de lucro.

1.3. Definiciones, Acrónimos y Abreviaturas

- **ERS:** Especificación de Requisitos del Software
- **Waggy:** Nombre de la tienda de mascotas
- **API:** Interfaz de Programación de Aplicaciones
- **REST:** Representational State Transfer

1.4. Referencias

Este documento sigue el estándar IEEE 830 para la especificación de requisitos de software.

1.5. Visión General del Documento

Este documento está organizado en secciones que describen la introducción, la descripción general del sistema, y los requisitos específicos tanto funcionales como no funcionales.

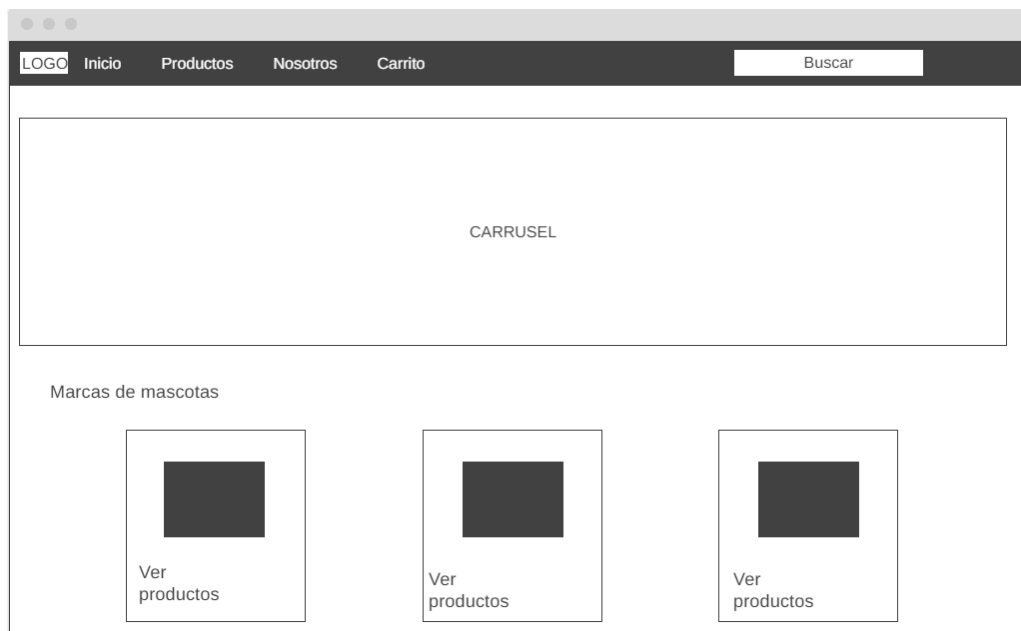
2. Descripción General

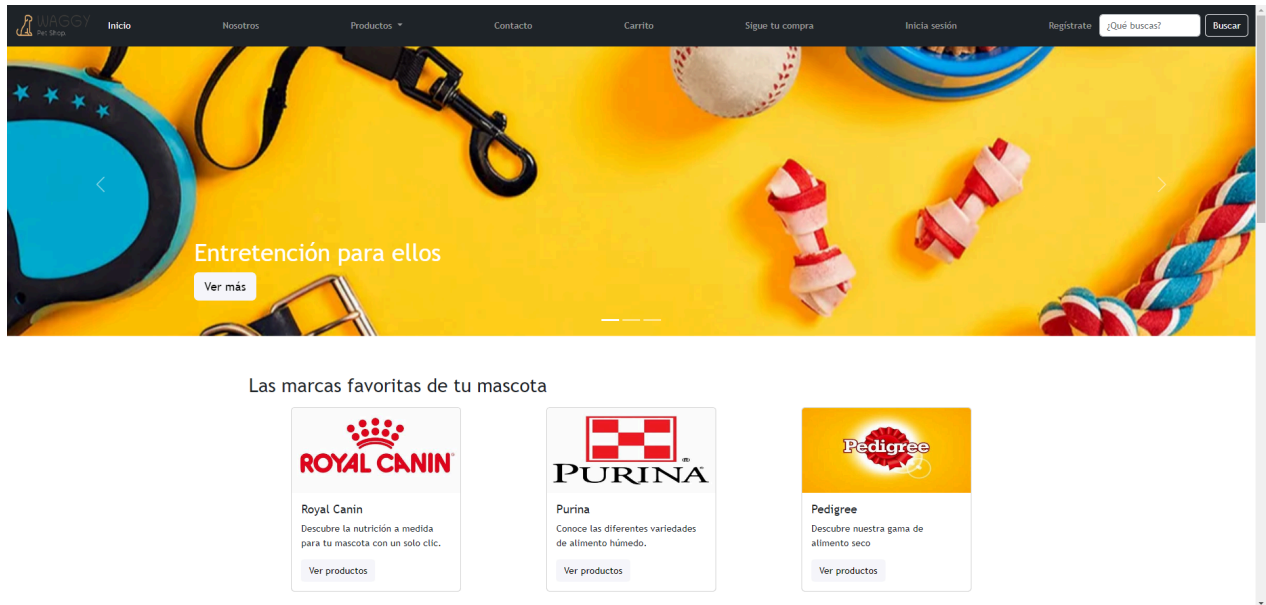
La aplicación web "Waggy Pet Shop" será una plataforma en línea que permitirá a los usuarios registrados comprar productos para mascotas, ver el historial de sus compras, y hacer donaciones a una fundación sin fines de lucro.

2.1. Perspectiva del Producto



Wireframe:



Versión final:**2.2. Funciones del Producto**

- Registro y administración de clientes
- Gestión de productos y stock
- Realización de compras en línea
- Monitoreo y seguimiento de pedidos
- Gestión de promociones y descuentos
- Donaciones a una fundación sin fines de lucro

2.3. Características de los Usuarios

- Usuarios finales: Clientes que compran productos y realizan donaciones.
- Administradores: Personal que gestiona productos, promociones y usuarios.

2.4. Restricciones

- La aplicación debe utilizar la base de datos ORACLE.
- La aplicación debe estar desarrollada en un ambiente web y seguir una arquitectura de capas.
- La aplicación debe acceder a servicios REST para gestionar suscripciones y donaciones.

2.5. Suposiciones y Dependencias

- **Acceso a Internet:** Se asume que los usuarios tendrán acceso a una conexión a Internet para utilizar la aplicación.

- **API de la Fundación:** La fundación asociada proporcionará y mantendrá un API REST para gestionar las suscripciones y validar donaciones.
- **Disponibilidad de Personal:** Se contará con un equipo de soporte técnico disponible para el mantenimiento y actualización del sistema.

2.6. Requisitos Futuros

- **Integración con Nuevas Plataformas de Pago:** A medida que crezca la demanda, se podrían integrar nuevas opciones de pago.
- **Soporte Multilingüe:** Implementación de soporte para múltiples idiomas para atender a una base de usuarios más amplia.
- **Aplicaciones Móviles:** Desarrollo de aplicaciones móviles nativas para iOS y Android para complementar la plataforma web

3. Requisitos Específicos

3.1 Requisitos comunes de los interfaces

3.1.1 Interfaces de usuario

- La aplicación debe tener una interfaz web amigable y accesible desde navegadores modernos (Chrome, Firefox, Safari).
- La interfaz debe ser responsive, adaptándose a diferentes tamaños de pantalla (escritorio, tablet, móvil).
- Debe existir un formulario de registro y autenticación para los usuarios.
- El sistema debe proporcionar una página de carrito de compras donde los usuarios puedan ver y modificar sus selecciones.
- La aplicación debe mostrar notificaciones y alertas a los usuarios sobre el estado de sus pedidos y promociones.

3.1.2 Interfaces de hardware

No se requiere hardware específico aparte de un servidor web y la infraestructura necesaria para mantener la base de datos ORACLE y los servicios REST.

3.1.3 Interfaces de software

- Integración con la base de datos ORACLE para la gestión de usuarios, productos, pedidos y promociones.
- Consumo de servicios REST proporcionados por la fundación para la gestión de suscripciones y validación de donaciones.

3.1.4 Interfaces de comunicación

La aplicación debe comunicarse de manera segura a través de HTTPS.

3.2 Requisitos funcionales

3.2.1 Requisito funcional 1

Registro y Gestión de Clientes

- El sistema debe permitir a los clientes registrarse con sus datos personales y de contacto.
- Los clientes deben poder iniciar sesión y gestionar su perfil.

3.2.2 Requisito funcional 2

Gestión de Productos y Stock

- El sistema debe permitir a los administradores agregar, modificar y eliminar productos del catálogo.
- Debe haber un control de stock que actualice automáticamente la disponibilidad de productos tras cada venta.

3.2.3 Requisito funcional 3

Carrito de Compras y Procesamiento de Pedidos

- Los clientes deben poder agregar productos a su carrito de compras y proceder al pago.

- El sistema debe procesar pagos en línea de manera segura.
- Una vez completada la compra, el sistema debe descontar los productos vendidos del stock.

3.2.4 Requisito funcional 4

Gestión de Promociones y Descuentos

- Los administradores deben poder crear y gestionar promociones y descuentos.
- El sistema debe aplicar automáticamente los descuentos a los clientes suscritos y durante promociones.

3.2.5 Requisito funcional 5

Historial de Compras y Seguimiento de Pedidos

- Los clientes deben poder ver el historial de sus compras y el estado actual de sus pedidos.
- El sistema debe proporcionar un seguimiento del despacho, mostrando la fecha y hora en que se toma el pedido, se despacha y es recibido por el cliente.

3.2.6 Requisito funcional 6

Gestión de Donaciones

- Los clientes deben poder suscribirse para realizar donaciones a la fundación sin fines de lucro.
- El sistema debe aplicar un 5% de descuento en las compras de los clientes suscritos.
- La información de suscripción debe ser enviada y validada a través del servicio REST de la fundación.

3.3 Requisitos no funcionales

3.3.1 Requisitos de rendimiento

- La aplicación debe soportar hasta 10,000 usuarios concurrentes.
- Las respuestas de la base de datos deben ser menores a 2 segundos.

3.3.2 Seguridad

Los datos personales y de pago deben ser manejados de forma segura y confidencial.

La seguridad de los servidores es indispensable para su buen funcionamiento, los datos que crean los servidores están encriptados en caso de alguna falla en el sistema de seguridad los datos estarán a salvo. También se da hincapié a un sistema de antivirus que prevé diferentes virus y amenazas que puedan perturbar la seguridad de los servidores.

3.3.3 Fiabilidad

- La aplicación debe tener una disponibilidad del 99.9%, con un tiempo de inactividad máximo de 8.76 horas al año.
- El sistema debe contar con mecanismos de recuperación ante fallos para minimizar el tiempo de inactividad.

3.3.4 Disponibilidad

La disponibilidad de la aplicación web "Waggy Pet Shop" es un factor crítico para asegurar la satisfacción del cliente y la continuidad del negocio. La siguiente especificación de disponibilidad debe ser cumplida:

- **Uptime:** La aplicación debe estar disponible al menos el 99.9% del tiempo, lo que equivale a un tiempo de inactividad máximo de aproximadamente 8.76 horas al año.
- **Mantenimiento Programado:** El tiempo de inactividad programado para mantenimiento no debe exceder 4 horas al mes. Las tareas de mantenimiento deben realizarse en horarios de menor tráfico (preferiblemente durante la noche o fines de semana) y deben ser notificadas a los usuarios con al menos 48 horas de antelación.
- **Recuperación de Desastres:** La aplicación debe contar con un plan de recuperación ante desastres que permita la restauración de los servicios en un máximo de 2 horas después de un incidente grave. Se deben realizar pruebas de recuperación al menos una vez al año.
- **Redundancia:** Se debe implementar una arquitectura de alta disponibilidad que incluya balanceo de carga y servidores redundantes para minimizar el impacto de fallos individuales.

- **Monitoreo y Alertas:** El sistema debe contar con herramientas de monitoreo que permitan la detección y notificación inmediata de cualquier problema que afecte la disponibilidad. Se deben configurar alertas para el equipo de soporte técnico para que puedan tomar acciones correctivas de manera oportuna.
- **Escalabilidad:** La infraestructura del sistema debe ser escalable para manejar picos de tráfico sin degradar el rendimiento o la disponibilidad del servicio.

3.3.5 Mantenibilidad

- El código debe ser modular y bien documentado para facilitar el mantenimiento y las futuras actualizaciones.
- Las actualizaciones y mejoras deben poder implementarse sin interrumpir el servicio.

3.3.6 Portabilidad

La aplicación debe ser compatible con los principales navegadores web (Chrome, Firefox, Safari).

El Código va a ser fácilmente elaborado por lo que su traspaso a otro lenguaje no será complicado.

3.4 Otros Requisitos

- La aplicación debe ser compatible con los principales navegadores web.
- El sistema debe ser accesible para personas con discapacidades, cumpliendo con las normas de accesibilidad web (WCAG 2.1).