Documento de requerimientos de software

**HealthPet**

**Fecha: [29/08/2024]**

# Tabla de contenido

[**Tabla de contenido 2**](#_heading=h.wsu30woue3qk)

[**Historial de Versiones 3**](#_heading=h.o0msasssu7l3)

[**Información del Proyecto 3**](#_heading=h.30j0zll)

[**1. Propósito 3**](#_heading=h.3vm7f0d3d5fi)

[**2. Alcance del producto / Software 4**](#_heading=h.dnqvyqo2hhuw)

[**3. Funcionalidades del producto 5**](#_heading=h.ne3502pfca6d)

[**4. Clases y características de usuarios 5**](#_heading=h.6oy2xmp9cqv9)

[**5. Entorno operativo 7**](#_heading=h.r7n572bdd576)

[**6. Requerimientos funcionales 9**](#_heading=h.5r5ylolqsew7)

[**7. Reglas de negocio 17**](#_heading=h.avgscuhkqq3j)

[**8. Requerimientos no funcionales 23**](#_heading=h.u84ucccr11an)

[**Aprobaciones 26**](#_heading=h.1fob9te)

# Historial de Versiones

| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
| --- | --- | --- | --- | --- |
| 29/08/2024 | 1.0 | Carlos Cortez/Agustin Rodriguez | Duoc UC | Desarrollo del Documento |
| 05/09/2024 | 1.1 | Maria Bahamondes | Duoc UC | Revisión y Aprobación |

# Información del Proyecto

| Empresa / Organización | Duoc UC |
| --- | --- |
| Proyecto | HealthPet |
| Fecha de preparación | 17/08/2024 |
| Cliente | Hospital veterinario San Agustín |
| Patrocinador principal | - |
| Gerente / Líder de Proyecto | Maria Bahamondes Lara |
| Gerente / Líder de Análisis de negocio y requerimientos | Carlos Cortez Castañeda |

# Propósito

Desarrollar una plataforma web para **Healthpet S.A** en su versión 1.0 que permita gestionar el inventario de productos veterinarios, realizar compras de alimento en línea y hacer pedidos de alimentos que no estén en stock pagando un abono.

**Diseñar y construir un módulo de gestión de inventario** que permita a los administradores agregar, actualizar y eliminar productos, así como gestionar las existencias y categorías de productos.

**Implementar un sistema de carrito de compras y proceso de pago** en línea, permitiendo a los usuarios seleccionar productos, calcular el total de la compra y realizar pagos seguros.

**Integrar un sistema de Registro de ventas** el cual permita agregar eliminar y editar ventas y que se vaya actualizando .

**Realizar pruebas exhaustivas de funcionalidad y seguridad** para asegurar que la plataforma web cumpla con los estándares de calidad y seguridad requeridos para un entorno de producción.

# Alcance del producto / Software

Desarrollar una plataforma web (marketplace) que permita a los usuarios gestionar pedidos de alimentos y servicios veterinarios de manera eficiente, ofreciendo funcionalidades clave como un carrito de compras, integración con pasarelas de pago, notificaciones de estado de pedido, y opciones de contacto con la veterinaria.

Los beneficios de este sistema para los usuarios internos sería poder mejorar los tiempos de trabajo con un ingreso más ágil de ventas y evitar el trabajo de contar inventario manualmente.

Los beneficios para los usuarios del Hospital Veterinario serían mejorar la experiencia de los veterinarios al mantener un control de la contabilidad así evitando el error humano además de poder tener un control de los deudores y actualizar el stock de la veterinaria permitiendo un cuadre entre los productos que se usan por paciente y que son comprados en la veterinaria además también permitirá reservar sacos de alimentos para sus mascotas evitando gastar tiempo yendo al hospital veterinario y evitando lo pedidos que no son retirados ya que se debe hacer un abono previo para poder hacer la reserva del alimento obligatoria mente.

# Funcionalidades del producto

1. **Funcionalidad de Gestión de Inventario**
2. **Funcionalidad de Gestión de ventas**
3. **funcionalidad de gestión de deudores**
4. **Funcionalidad de Reportes de Administración**
5. **Funcionalidad de Registro de Usuarios**
6. **Funcionalidad de pedidos de los usuarios**
7. **Funcionalidad de Seguimiento de Pedidos**
8. **Funcionalidad de Compra de Productos**
9. **Funcionalidad de Gestión de Carrito de Compras**

# Clases y características de usuarios

**1. Administrador del Sistema**

* **Funcionalidades**: Inventario,Gestión de ventas,gestión de deudores y reportes mensuales.

**2. Personal de Recepción**

* **Funcionalidades**: comunicación con clientes, Gestión de pedidos de alimentos para mascotas y registro de ventas .

3. Veterinarios

* **Funcionalidades**: Gestión de citas y procedimientos médicos.

4. Clientes de Veterinaria

* **Funcionalidades**:Compra y pedidos de alimentos para mascotas y realizar pagos.

# 

# Entorno operativo

**Entorno operativo:**

El software deberá ejecutarse en un entorno web hospedado en un servidor ubicado en Chile para garantizar un mejor acceso.

Servidor: Se requiere un servidor con las siguientes especificaciones mínimas:

* Procesador: CPU de al menos 4 núcleos.
* Memoria RAM: 16 GB.
* Almacenamiento: 100 GB de espacio libre en disco, con almacenamiento SSD preferido para mayor rendimiento.
* Conectividad: Ancho de banda mínimo de 100 Mbps para garantizar acceso rápido a los usuarios.
* Sistema operativo: Linux con soporte para entornos web.
* Base de datos: MySQL (versión 8.0 o superior).

**Entorno de ejecución:**

El sistema deberá ser accesible desde cualquier dispositivo de escritorio o portátil que permita el uso de un navegador web.

Dispositivos (mínimos):

* Procesador: CPU de 2 núcleos.
* Memoria RAM: 4 GB.
* Navegador: Google Chrome, Mozilla Firefox, Microsoft Edge (versiones actualizadas) con soporte para HTML5 y CSS3.
* Resolución de pantalla mínima: 1280 x 720 píxeles.
* Conexión a internet: Mínimo 10 Mbps para una experiencia óptima de navegación y uso.

**6.Requerimientos funcionales**

Módulo de pedidos de Alimentos Mascotas

**Funcionalidad 1: Gestión de Productos de Alimeantos**

* **Descripción**: Permitir a los usuarios visualizar y seleccionar alimentos para mascotas, organizados por categoría.
* **Acciones iniciadoras y comportamiento esperado**:
  + El usuario ingresa a la sección de compra de alimentos.
  + El sistema muestra las categorías (e.g., perros, gatos) y los productos disponibles.
  + El usuario selecciona un producto.
  + El sistema muestra la descripción del producto, precio, y opciones de cantidad y si hay stock disponible en veterinaria.
  + podra hacer pedido de algun alimento sin stock pagando un abono
  + El usuario agrega el producto al carrito.
* **Requerimientos funcionales**:
  + **REQ-1**: El sistema debe mostrar una lista de alimentos clasificados por categoría de especie.
  + **REQ-2**: El sistema debe permitir visualizar la descripción detallada del producto.
  + **REQ-3**: El sistema debe mostrar el precio del producto y permitir seleccionar la cantidad deseada.
  + **REQ-4**: El sistema debe permitir agregar productos seleccionados al carrito de compra.
  + **REQ-5**: El sistema debe alertar al usuario si el stock del producto es insuficiente y permitir hacer pedido pagando un abono.
  + **REQ-6**: El sistema debe responder con un mensaje de error si un producto no está disponible.

**Funcionalidad 2: Gestión del Carrito de Compras**

* **Descripción**: Permitir a los usuarios visualizar y gestionar los productos seleccionados antes de realizar la compra.
* **Acciones iniciadas y comportamiento esperado**:
  + El usuario selecciona el carrito de compra.
  + El sistema muestra los productos seleccionados, con sus cantidades y precios.
  + El usuario puede modificar la cantidad o eliminar productos.
  + El usuario procede al pago o continúa comprando.
* **Requerimientos funcionales**:
  + **REQ-7**: El sistema debe permitir visualizar los productos añadidos al carrito, incluyendo cantidad y precio total.
  + **REQ-8**: El sistema debe permitir modificar la cantidad de cada producto dentro del carrito.
  + **REQ-9**: El sistema debe permitir eliminar productos del carrito.
  + **REQ-10**: El sistema debe actualizar automáticamente el precio total cuando se modifica la cantidad o se eliminan productos.
  + **REQ-11**: El sistema debe permitir continuar la compra o proceder al pago desde el carrito.
  + **REQ-12**: El sistema debe colocar la opción de hacer pedido de inmediato al quedar sin stock disponible.

**Funcionalidad 3: Procesamiento de Pago**s

* **Descripción**: Permitir a los usuarios realizar el pago de los productos seleccionados.
* **Acciones iniciadas y comportamiento esperado**:
  + El usuario selecciona la opción de pago.
  + El sistema muestra las opciones de pago disponibles.
  + El usuario selecciona una opción de pago y confirma los datos.
  + El sistema procesa el pago y confirma la compra.
* **Requerimientos funcionales**:
  + **REQ-13**: El sistema debe mostrar las opciones de pago disponibles (tarjeta, PayPal, etc.).
  + **REQ-14**: El sistema debe validar la información de pago ingresada por el usuario.
  + **REQ-15**: El sistema debe notificar al usuario si la transacción fue exitosa o fallida.
  + **REQ-16**: El sistema debe generar una confirmación de pedido una vez procesado el pago.
  + **REQ-17**: El sistema debe enviar un correo electrónico al usuario con los detalles de la compra.

**Funcionalidad 4: Seguimiento de Pedidos**

* **Descripción**: Permitir a los usuarios revisar el estado de sus pedidos realizados.
* **Acciones iniciadoras y comportamiento esperado**:
  + El usuario ingresa a la sección de historial de pedidos.
  + El sistema muestra el listado de pedidos previos y su estado actual.
* **Requerimientos funcionales**:
  + **REQ-18**: El sistema debe mostrar un historial de pedidos para usuarios registrados.
  + **REQ-19**: El sistema debe permitir al usuario ver el estado actual de su pedido (en preparación, listo para retiro).

Funcionalidad 5: Registro y Login de Usuarios

* **Descripción**: Permitir a los usuarios crear una cuenta, iniciar sesión y acceder a funcionalidades personalizadas.
* **Acciones iniciadoras y comportamiento esperado**:
  + El usuario selecciona la opción de registro o login.
  + El sistema permite ingresar datos de cuenta o registrarse con un nuevo email.
  + El usuario registrado puede acceder a su perfil y pedidos previos.
* **Requerimientos funcionales**:
  + **REQ-22**: El sistema debe permitir a los usuarios registrarse con un correo electrónico válido.
  + **REQ-23**: El sistema debe permitir a los usuarios iniciar sesión con sus credenciales.
  + **REQ-24**: El sistema debe proporcionar una opción de recuperación de contraseña en caso de olvido.
  + **REQ-25**: El sistema debe permitir a los usuarios registrados acceder a sus perfiles y gestionar sus datos.

**Módulo de Sistema Integral de Gestión interna para la Clínica Veterinaria**

**Funcionalidad 1:** Gestión de Inventario de Productos y Medicamentos

* **Descripción**: Gestionar el inventario de productos y medicamentos, asegurando su disponibilidad para las atenciones médicas.
* **Acciones iniciadas y comportamiento esperado**:
  + El administrador accede al sistema y selecciona la opción de inventario.
  + El sistema muestra la lista de productos y medicamentos, junto con su cantidad disponible.
  + El administrador actualiza o añade productos al inventario, y el sistema refleja los cambios.
* **Requerimientos funcionales**:
  + **REQ-1**: El sistema debe permitir la visualización del inventario, mostrando una lista de productos y medicamentos junto con su cantidad disponible.
  + **REQ-2**: El sistema debe generar alertas cuando el stock de un producto o medicamento esté por debajo del nivel mínimo.
  + **REQ-3**: El sistema debe permitir la actualización de productos y cantidades en el inventario.
  + **REQ-4**: El sistema debe reflejar inmediatamente cualquier cambio en el inventario después de confirmarlo.
  + **REQ-5**: El sistema debe registrar automáticamente los productos ingresados, actualizando el stock disponible.

**Funcionalidad 2: Registro y Actualización de Ventas.**

* **Descripción**: Registrar y actualizar las ventas de productos, servicios y hospitalizaciones, para llevar un control preciso de los ingresos.
* **Acciones iniciadoras y comportamiento esperado**:
  + El administrador realiza la venta de un producto, servicio o hospitalización.
  + El sistema actualiza el inventario en caso de productos y refleja la venta en los registros.
* **Requerimientos funcionales**:
  + **REQ-6**: El sistema debe registrar automáticamente las ventas de productos cuando se confirme una venta, actualizando el inventario correspondiente.
  + **REQ-7**: El sistema debe registrar los servicios veterinarios completados, reflejando los ingresos generados.
  + **REQ-10**: El sistema debe reflejar los ingresos de cada venta o servicio de forma inmediata en el registro general.

Funcionalidad 4: Registro de Deudas

**Descripción**: Registrar deudas, facilitando el seguimiento financiero de la clínica.

* **Acciones iniciadoras y comportamiento esperado**:
  + El administrador registra el servicio prestado, y el sistema genera una deuda si el pago está pendiente.
  + El cliente realiza un pago, y el sistema actualiza la deuda en función del pago.
* **Requerimientos funcionales**:
  + **REQ-15**: El sistema debe registrar automáticamente una deuda en la cuenta del cliente si se ha prestado un servicio médico con pago pendiente.
  + **REQ-16**: El sistema debe permitir registrar pagos parciales o totales de una deuda, actualizando el monto pendiente.
  + **REQ-17**: El sistema debe permitir visualizar un historial de pagos, mostrando detalles de los pagos realizados, fechas y montos.
  + **REQ-18**: El sistema debe reflejar cualquier cambio en la deuda del cliente en tiempo real, tras registrar un pago.

Funcionalidad 5: Visualización del Historial Financiero

* **Descripción**: Permitir la visualización del historial de pagos y deudas de los clientes.
* **Acciones iniciadoras y comportamiento esperado**:
  + El administrador accede a la sección de pagos.
  + El sistema muestra un historial detallado de todos los pagos y deudas de un cliente.
* **Requerimientos funcionales**:
  + **REQ-19**: El sistema debe mostrar un historial detallado de los pagos realizados por cada cliente, incluyendo fechas, montos, y los servicios asociados.
  + **REQ-20**: El sistema debe permitir a los administradores filtrar el historial de pagos por rut y fecha.

# Reglas de negocio

**Plataforma web de compras de alimentos para mascotas**

Categorías de Mascotas:

* Mostrar categorías (Perro, Gato) y filtrar productos por categoría seleccionada.
* Los productos se muestran en orden de precio (rango).

Disponibilidad de Productos:

* Solo productos en stock pueden agregarse al carrito.
* Mostrar "sin stock" para productos sin stock.

Pago:

* Usuarios registrados confirman dirección y proceden al pago.
* Invitados pueden pagar ingresando sus dato para contactar con el.

Seguimiento de Pedidos:

* Usuarios registrados ven sus pedidos en "Mis Pedidos".
* usuario podra ver estado del pedido.

**Gestión administrativa de la plataforma de ventas para mascotas**

Añadir Nuevos Productos:

* Solo los administradores pueden añadir nuevos productos al catálogo.
* El sistema debe permitir ingresar nombre, categoría, precio, cantidad y descripción del producto.
* Tras ingresar los datos y guardar, el producto se añade al catálogo y está disponible para la compra.

Archivar Productos:

* Solo los administradores pueden archivar productos del catálogo.
* El sistema debe permitir archivar productos sin stock o descontinuados.
* Se requiere confirmación antes de archivar el producto, que después deja de mostrarse en el catálogo.

Modificar Productos:

* Solo los administradores pueden modificar la información de los productos.
* El sistema debe mostrar un formulario con la información actual del producto para su edición.
* Tras realizar cambios y guardar, la información del producto se actualiza en el catálogo.

Registrar Ventas:

* El sistema debe registrar automáticamente las ventas cuando se completa una compra.
* Los administradores pueden visualizar un resumen de todas las ventas registradas, incluyendo detalles del producto, cantidad y fecha.

Monitorear Estado de Pedidos:

* Solo los administradores pueden monitorear el estado de los pedidos en curso.
* El sistema debe mostrar una lista de pedidos con su estado actual (en proceso, entregado) y actualizar el estado en tiempo real.

Registrar Deudas y Pagos:

* El sistema debe registrar deudas cuando se presta un servicio con pago pendiente.
* Debe registrar pagos parciales o totales y actualizar la cuenta del cliente en consecuencia.
* Los administradores pueden visualizar un historial de pagos detallado, con fechas, montos y servicios asociados.

**Gestión interna de veterinaria:**

Visualizar Inventario:

* Los administradores deben poder acceder a una lista de todos los productos y medicamentos, mostrando su cantidad disponible.
* El sistema debe mostrar esta lista cuando el administrador acceda a la sección de inventario.

Alerta de Bajo Stock:

* Los administradores deben recibir alertas cuando el stock de un producto o medicamento esté por debajo del nivel mínimo.
* El sistema debe generar automáticamente una alerta para que el administrador gestione el reabastecimiento.

Actualización de Inventario:

* Los administradores deben poder ingresar nuevos productos o actualizar el stock existente.
* El sistema debe actualizar el inventario y reflejar la nueva cantidad disponible una vez que los cambios sean confirmados.

Registrar y Actualizar Ventas

* Registro de Ventas de Productos:
  + Los administradores deben registrar automáticamente las ventas de productos y actualizar el inventario.
  + El sistema debe registrar la venta y ajustar el inventario de manera inmediata al confirmar una venta.
* Registro de Servicios Veterinarios:
  + Los administradores deben registrar la venta de servicios veterinarios y reflejarla en los ingresos.
  + El sistema debe registrar el servicio y actualizar los ingresos correspondientes al confirmar el servicio.
* Registro de Hospitalizaciones:
  + Los administradores deben registrar los costos de hospitalizaciones y reflejarlos en el registro de ventas e ingresos.
  + El sistema debe actualizar el registro de ventas e ingresos con los costos de la hospitalización al confirmar el alta del paciente.

Registrar Deudas¡

* Registro de Deuda:
  + Los administradores deben registrar deudas cuando se preste un servicio médico con un pago pendiente.
* Registro de Pago Parcial o Total:
  + Los administradores deben registrar pagos parciales o totales y actualizar la deuda del cliente.
  + El sistema debe ajustar la cuenta del cliente y reducir el monto de la deuda según el pago realizado.
* Visualización del Historial de Pagos:
  + Los administradores deben ver un historial detallado de pagos de un cliente, incluyendo fechas, montos y servicios asociados.
  + El sistema debe mostrar un historial completo cuando se acceda a la sección de pagos.

# Requerimientos no funcionales

1. Rendimiento

* **RNF-1**: El sistema debe ser capaz de procesar al menos 100 transacciones simultáneas (ventas, citas, actualizaciones de inventario, etc.) sin afectar el rendimiento.
* **RNF-2**: El tiempo de respuesta del sistema para consultas de inventario, ventas, y citas no debe superar los 2 segundos bajo condiciones de carga normal.

2. Escalabilidad

* **RNF-3**: El sistema debe ser capaz de escalar para soportar un crecimiento del 200% en el número de usuarios y transacciones en un plazo de 2 años.
* **RNF-4**: La infraestructura del sistema debe permitir la adición de nuevas funcionalidades o módulos (e.g., gestión de reportes financieros) sin requerir una reestructuración completa del sistema.

3. Seguridad

* **RNF-5**: El sistema debe cumplir con los estándares de seguridad para el manejo de datos sensibles (información de clientes y animales), asegurando que todos los datos estén cifrados tanto en tránsito como en reposo.
* **RNF-6**: El acceso al sistema debe estar restringido mediante autenticación multifactor (MFA) para los administradores, con roles y permisos definidos según sus funciones.
* **RNF-7**: Todas las actividades críticas, como la modificación de inventario o la gestión de pagos, deben ser registradas en un log de auditoría que no pueda ser alterado.

4. Disponibilidad

* **RNF-8**: El sistema debe tener una disponibilidad del 99.9% para asegurar que esté operativo prácticamente en todo momento, minimizando el tiempo de inactividad.
* **RNF-9**: En caso de una falla del sistema, el tiempo de recuperación debe ser inferior a 30 minutos.

5. Usabilidad

* **RNF-10**: La interfaz de usuario debe ser intuitiva y fácil de usar, de manera que los administradores puedan realizar tareas comunes (gestionar inventario, ventas, citas) con un máximo de 3 clics.
* **RNF-11**: El sistema debe contar con manual de usuario integrado.

6. Mantenibilidad

* **RNF-12**: El sistema debe permitir actualizaciones y parches de seguridad sin requerir tiempo de inactividad significativo (menos de 20 minutos).
* **RNF-13**: El código fuente del sistema debe seguir estándares de desarrollo que faciliten su mantenimiento y evolución, como la modularidad y el uso de control de versiones.

7. Interoperabilidad

* **RNF-14**: El sistema debe ser capaz de integrarse con otros sistemas externos como pasarelas de pago, sistemas de facturación, y proveedores de medicamentos, a través de APIs estandarizadas.
* **RNF-15**: El sistema debe poder exportar e importar datos en formatos estándar como CSV, XML y JSON para permitir el intercambio de información con otros sistemas.

8. Portabilidad

* **RNF-16**: El sistema debe ser compatible con múltiples navegadores web, incluyendo Chrome, Firefox y Safari, sin perder funcionalidad.
* **RNF-17**: Debe ser posible migrar el sistema a otras plataformas cloud (e.g., AWS, Azure) sin pérdida de datos o funcionalidad.

9. Tolerancia a Fallos

* **RNF-18**: El sistema debe contar con un mecanismo de backup automático que permita la recuperación de datos en caso de fallos críticos, con un backup realizado cada hora.
* **RNF-19**: Ante un fallo de red, el sistema debe mantener las transacciones en cola y procesarlas una vez restablecida la conexión, sin pérdida de datos.

10. Accesibilidad

* **RNF-20**: Debe ofrecer soporte para navegación mediante teclado.

# 

# 

# Aprobaciones

| **Nombre y Apellido** | **Cargo** | **Departamento u Organización** | **Fecha** | **Firma** |
| --- | --- | --- | --- | --- |
| Maria Bahamondes | Scrum master | DuocUC | 05-09-2024 | MBB |