Documento de Requerimientos de Software

Sistema de Alquileres Logístico (S.A.L.)

**Fecha:** 22/11/2024

**Tabla de contenido**

### 

1. Propósito
2. Alcance del Producto / Software
3. Referencias
4. Funcionalidades del Producto
5. Clases y Características de Usuarios
6. Entorno Operativo
7. Requerimientos Funcionales
8. Reglas de Negocio
9. Requerimientos de Interfaces Externas
10. Requerimientos No Funcionales
11. Otros Requerimientos
12. Glosario

**Historial de Versiones**

| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
| --- | --- | --- | --- | --- |
| 28-10 | 1.0 | Javier Venegas | Soluciones Duoc | Creación del documento |
| 29-10 | 1.1 | Javier Venegas | Soluciones Duoc | Revisión de documento |

1. **Propósito**

El Sistema de Alquileres Logístico (S.A.L.) Tiene como propósito gestionar de manera eficiente el almacenamiento, alquiler y logística de productos en bodegas. El sistema incluirá funcionalidades de CRUD (Crear, Leer, Actualizar, Eliminar) para la gestión de inventarios, administración de clientes, generación de contratos de alquiler y seguimiento de pagos.

Se busca implementar un sistema web dinámico, basado en Django, que permita optimizar procesos logísticos y administrativos, además de facilitar el registro y monitoreo de operaciones en tiempo real.

1. **Alcance del producto / Software**

El sistema S.A.L. será una solución integral para empresas que deseen gestionar bodegas y su logística. Sus principales alcances son:

1. **Gestión de Almacenes:** Registro y control del stock disponible en las bodegas.
2. **Gestión de Alquileres:** Registro de contratos de alquiler con seguimiento de pagos, fechas y productos asociados.
3. **Control de Inventarios**: CRUD completo para registrar, actualizar, consultar y eliminar productos almacenados.
4. **Módulo de Pagos**: Integración con plataformas como PayPal o WebPay para realizar pagos de alquileres y servicios.
5. **Seguimiento Logístico:** Visualización en tiempo real de la disponibilidad de espacio en las bodegas.
6. **Reportes y Análisis:** Generación de reportes sobre inventario, contratos activos, ingresos por alquileres y más.

1. **Funcionalidades de requerimientos del producto**

**CRUD de Almacenes**:

* Registro de nuevas bodegas, con capacidad máxima, ubicación y estado actual.
* Edición de información sobre las bodegas existentes.
* Eliminación de bodegas no operativas.

**CRUD de Productos**:

* Registro de nuevos productos con datos como categoría, descripción, cantidad, precio de alquiler, y ubicación.
* Actualización del stock de productos en tiempo real.
* Eliminación de productos obsoletos o dañados.

**Gestión de Contratos**:

* Registro de nuevos contratos de alquiler, incluyendo cliente, producto(s), duración, y términos.
* Seguimiento de pagos asociados al contrato.
* Cierre o renovación de contratos.

**Módulo de Pagos**:

* Procesamiento de pagos a través de plataformas como PayPal o transferencia bancaria.
* Generación de comprobantes de pago.

**Reportes y Estadísticas**:

* Reporte de ocupación de bodegas.
* Resumen de ingresos por alquileres.
* Informe de productos más solicitados.

**Notificaciones**:

* Alertas de pagos pendientes o vencidos.
* Recordatorios de renovación de suscripción.

1. **Clases y características de usuarios**
2. **Usuario Administrador**:
   * Gestiona el sistema completo.
   * Puede crear, editar, eliminar usuarios, productos y contratos.
   * Supervisa el flujo de pagos.

1. **Usuario Logístico**:
   * Encargado de la organización y ubicación de productos en bodegas.
   * Verifica y actualiza el stock según las entradas o salidas.

1. **Usuario Cliente**:
   * Registra sus datos personales para alquilar productos o espacios.
   * Puede visualizar su historial de contratos y realizar pagos.

1. **Entorno operativo**

* **Frontend**: HTML5, CSS3, Bootstrap y JavaScript.
* **Backend**: Django (Python) con integración a bases de datos PostGress SQL
* **Servidores**: Windows Server 2019 o Linux (Ubuntu 22.04).
* **Integraciones**: API de PayPal/WebPay para pagos en línea.

1. **Requerimientos funcionales**

* **Gestión de Almacenes**
  + Crear, leer, actualizar y eliminar almacenes.
  + Asignar ubicación física (ciudad, región, país) a cada almacén.
  + Permitir definir la capacidad máxima de almacenamiento para cada almacén (en metros cúbicos o número de ítems).
* **Gestión de Inventario**
  + Crear, leer, actualizar y eliminar ítems en inventario.
  + Asociar cada ítem a un almacén específico.
  + Registrar atributos de cada ítem: código único, nombre, descripción, categoría, cantidad disponible, proveedor, fecha de ingreso y fecha de vencimiento (si aplica).
  + Validar automáticamente si el stock supera la capacidad del almacén.
* **Registro de Usuarios y Roles**
  + Registro y autenticación de usuarios con roles específicos: **administrador**, **gestor logístico**, y **cliente**.
  + Permitir asignar permisos según el rol:
    - Administrador: gestión total del sistema.
    - Gestor Logístico: supervisión y modificación del inventario.
    - Cliente: consulta de disponibilidad y solicitudes de alquiler.
* **Gestión de Solicitudes de Alquiler**
  + Crear solicitudes de alquiler por parte de los clientes.
  + Asociar cada solicitud a un almacén y especificar los ítems requeridos y las cantidades.
  + Generar un identificador único para cada solicitud.
  + Registrar el estado de las solicitudes: **pendiente**, **aprobada**, **en preparación**, **completada** o **rechazada**.
  + Permitir que los administradores gestionen el estado de las solicitudes.
* **Método de Pago**
  + Integrar un sistema de pago para clientes para confirmar las solicitudes de alquiler.
  + Permitir pagos mediante tarjetas de crédito, débito o plataformas como PayPal.
  + Generar comprobantes de pago en formato PDF.
* **Historial de Transacciones**
  + Guardar un registro de todas las solicitudes de alquiler procesadas, incluidas las rechazadas.
  + Mostrar detalles como cliente, almacén, ítems alquilados, fecha de inicio y fin del alquiler, monto pagado y estado actual.

1. **Reglas de negocio**

**Reglas de Alquiler**

* Un cliente solo puede alquilar ítems que estén disponibles en cantidad suficiente.
* Las solicitudes de alquiler tienen una duración máxima configurable (ej., 30 días).
* Si un cliente no realiza el pago en un tiempo definido (ej., 24 horas), la solicitud se marca como cancelada.

**Reglas de Inventario**

* Los ítems en estado "agotado" no deben estar disponibles para alquiler.
* Al aprobar una solicitud de alquiler, se debe descontar automáticamente el stock del almacén correspondiente.

**Reglas de Pago**

* Todos los pagos deben ser confirmados antes de procesar la solicitud de alquiler.
* Las transacciones deben realizarse de manera segura usando HTTPS y cifrado SSL.

**Reglas de Seguridad**

* Contraseñas almacenadas de forma cifrada.
* Los administradores no pueden visualizar contraseñas de usuarios.

**Reglas de Usuario**

* Los clientes solo pueden ver sus propias solicitudes y transacciones.
* Los administradores no pueden eliminar solicitudes completadas.

1. **Requerimientos de interfaces externas**

**Interfaces de Usuario**

* Diseño responsivo compatible con dispositivos móviles y de escritorio.
* Panel de administrador: Gestión de usuarios, almacenes, inventario y solicitudes.
* Panel de cliente: Visualización de ítems disponibles, generación de solicitudes y pagos.

**Interfaces de Hardware**

* Compatible con servidores que soporten tecnologías como Django y bases de datos MySQL o PostgreSQL.
* Optimización para dispositivos con navegadores modernos.

**Interfaces de Software**

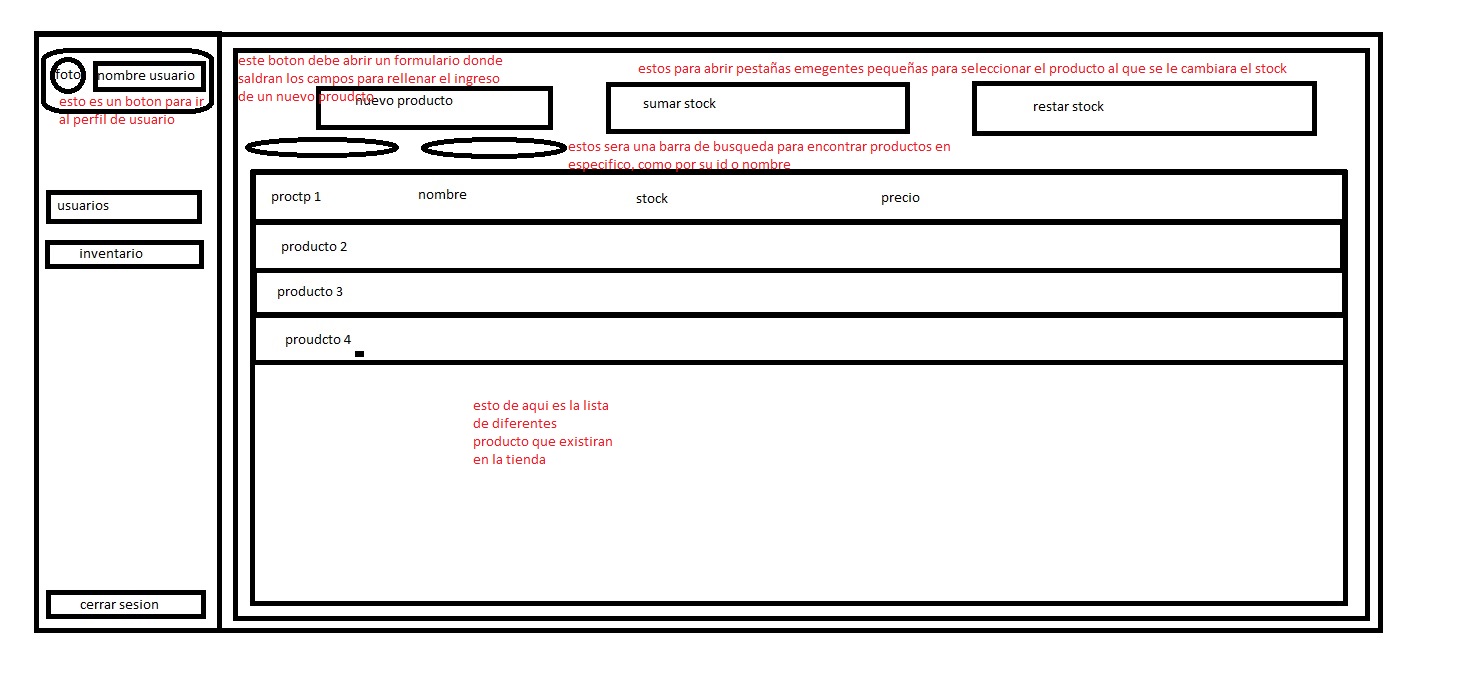
* Uso de APIs REST para comunicación entre frontend y backend.
* Integración con servicios de pago (PayPal, Stripe).
* Notificaciones por correo electrónico para actualizaciones de estado de las solicitudes.

**Interfaces de Comunicación**

* Mensajes automáticos por correo electrónico o SMS para actualizaciones importantes (aprobación/rechazo de solicitudes, pagos confirmados, etc.).
* Reportes exportables en formatos PDF y Excel para administradores.

Ejemplos de pantallas:





1. **Requerimientos no funcionales**
2. **Seguridad**
   * Cifrado de datos sensibles, como información de usuario y transacciones.
   * Implementar autenticación multifactor (2FA) para administradores.
3. **Disponibilidad**
   * Garantizar una disponibilidad mínima del sistema del 99.5% anual.
4. **Mantenibilidad**
   * Código limpio y documentado siguiendo estándares PEP-8 (Python).
   * Pruebas automatizadas para garantizar la estabilidad del sistema.
5. **Escalabilidad**
   * Soporte para hasta 50.000 usuarios concurrentes.
   * Capacidad de crecimiento para añadir nuevos almacenes y categorías de ítems sin afectar el rendimiento.
6. **Rendimiento**
   * Tiempo de carga promedio de la página inferior a 2 segundos.
   * Consultas a la base de datos ejecutadas en menos de 200 ms.

***El sistema debe hacerse conforme a los***

***requerimientos de calidad de la ISO 25000.***

1. **Otros requerimientos**

**Base de Datos**

1. Soporte para múltiples relaciones: almacenes, ítems, usuarios, solicitudes y transacciones.
2. Copias de seguridad automáticas cada 24 horas.
3. Garantizar la integridad referencial de las tablas.

**Integración con Servicios Externos**

1. Sistema de pago externo (PayPal, Stripe).
2. Servicios de geolocalización para optimizar las ubicaciones de los almacenes.

**Compatibilidad del Sistema**

1. Navegadores: Chrome, Firefox, Safari, Edge.
2. Sistemas operativos: Windows, macOS, Linux, Android, iOS.

**Cumplimiento Legal**

1. Cumplir con normativas locales e internacionales de protección de datos (ej., GDPR).
2. Almacenar información confidencial según estándares de seguridad reconocidos.
3. **Glosario**

* Software: Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora.
* Requisito funcional: Un requisito funcional esuna declaración de cómo debe comportarse un sistema**.** Define lo que el sistema debe hacer para satisfacer las necesidades o expectativas del usuario. Los requisitos funcionales se pueden considerar como características que el usuario detecta.
* Requisito no funcional: Los requisitos no funcionales (NFR) son las restricciones impuestas a un sistema que definen sus atributos de calidad. Por lo general, se denotan con adjetivos como seguridad, rendimiento y escalabilidad.
* Hardware: Hardware es la parte física de un ordenador o sistema informático. Está formado por componentes eléctricos, electrónicos, electromecánicos y mecánicos.
* CRUD: En informática, CRUD es el acrónimo de "Crear, Leer, Actualizar y Borrar" (del original en inglés: Create, Read, Update and Delete), que se usa para referirse a las funciones básicas en bases de datos o la capa de persistencia en un software.
* HTTPS: Es la versión segura de **HTTP,** que es el principal protocolo utilizado para **enviar datos entre un navegador web** y **un sitio web.** El HTTPS está encriptado para aumentar la seguridad de las transferencias de datos.
* SSL:Un certificado SSL es un certificado digital que autentica la identidad de un sitio web y habilita una conexión cifrada. La sigla SSL significa Secure Sockets Layer (Capa de sockets seguros), un protocolo de seguridad que crea un enlace cifrado entre un servidor web y un navegador web.