



## **Especificación de requisitos de software**

Proyecto: LiquorLogix

Fecha: 2024



Ficha del documento

Fecha	Autor	Revisión	Revisor	Verificado dep. calidad.
15-10/2024	Cristopher Benavides		Inst. Albeiro Ramos	



## Contenido

FICHA DE DOCUMENTO	2
CONTENIDO	3
1. Introducción	4
1.1 Objetivo	4
1.2 Alcance	4
1.3 Definiciones y Acrónimos	4
1.4 Referencias	4
1.5 Vista general del SRS	4
2. Descripción general	5
2.1 Perspectiva del producto	5
2.2 Funcionalidad del producto	5
2.3 Características de usuario	5
2.4 Restricciones	6
2.5 Suposiciones y dependencias	6
2.6 Evolución previsible del sistema	6
3. Requisitos específicos	7
3.1 Requisitos comunes de las interfaces	7
3.2 Interfaces de comunicación	8
3.3 Requisitos funcionales	8
3.4 Requisitos no funcionales	9
3.5 Otros requisitos	10
4. Apéndices	10



## 1. Introducción

### 1.1 Objetivo

El objetivo de este Documento de Especificación de Requisitos de Software (SRS) es definir de manera clara y precisa los requisitos funcionales y no funcionales del sistema de gestión de inventario para una licorera. Este sistema permitirá a los usuarios registrar, gestionar y visualizar productos, controlar el stock, manejar distribuidores y realizar un seguimiento de la contabilidad.

### 1.2 Alcance

El sistema de gestión de inventario se enfocará en las siguientes funcionalidades clave:

- Ingreso y salida de productos.
- Visualización del stock disponible.

Este sistema estará destinado a ser utilizado por administradores y empleados de la licorera, facilitando la gestión eficiente de los productos y transacciones.

### 1.3 Definiciones y Acrónimos

- **SRS:** Documento de Especificación de Requisitos de Software.
- **Inventario:** Conjunto de productos almacenados en la licorera.
- **Stock:** Cantidad disponible de cada producto.
- **Producto:** Artículo a la venta en la licorera, que incluye información como nombre, precio, etc.

### 1.4 Referencias

- IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications.
- Documentación sobre gestión de inventarios y contabilidad para comercios.

### 1.5 Vista General del SRS

Este SRS está organizado en varias secciones que cubren la descripción general del sistema, requisitos funcionales, requisitos no funcionales, y consideraciones sobre el diseño y la implementación. Cada sección tiene como objetivo proporcionar



una comprensión completa del sistema propuesto y servir como base para el desarrollo del software.

## 2. Descripción General

### 2.1 Perspectiva del Producto

El sistema de gestión de inventario para la licorera es un producto independiente que no forma parte de un sistema mayor en este momento. Sin embargo, está diseñado para poder integrarse con sistemas de contabilidad o gestión de ventas en el futuro, lo que permitirá una gestión más holística de las operaciones de la licorera.

### 2.2 Funcionalidad del Producto

El producto debe realizar las siguientes funcionalidades principales:

- **Ingreso de Productos:** Permitir a los usuarios registrar nuevos productos en el inventario con detalles como nombre, descripción, precio de compra, precio de venta y stock inicial.
- **Salida de Productos:** Facilitar el registro de ventas de productos, actualizando automáticamente el stock disponible.
- **Visualización de Stock:** Proporcionar una interfaz para que los usuarios vean el stock actual de todos los productos, con opciones de filtrado.

### 2.3 Características de los Usuarios

#### Tipo de Usuario

- **Administradores:** Encargados de gestionar el sistema y mantener la información actualizada.
- **Empleados:** Usan el sistema para registrar ventas y manejar el inventario diario.

#### Formación

- Se espera que los usuarios tengan formación básica en computación y un conocimiento práctico de la gestión de inventarios.

#### Habilidades

- **Administradores:** Habilidades intermedias en gestión de datos y conocimientos sobre la industria de bebidas.



- **Empleados:** Habilidades básicas en la operación de computadoras y manejo de software de gestión.

#### Actividades

- Los administradores realizarán tareas de configuración y mantenimiento del sistema, mientras que los empleados se enfocarán en registrar ingresos y salidas de productos.

#### 2.4 Restricciones

- El sistema debe desarrollarse utilizando tecnologías web (HTML, CSS, JavaScript) y puede estar basado en frameworks como React o Angular para el frontend.
- El backend debe ser implementado en PHP debe ser compatible con bases de datos relacionales como MySQL o PostgreSQL.
- Debe ser accesible desde navegadores web modernos y debe ser responsivo para su uso en dispositivos móviles.

#### 2.5 Suposiciones y Dependencias

- Se asume que los usuarios tendrán acceso a dispositivos conectados a Internet para utilizar el sistema.
- Se asume que la infraestructura de red es suficiente para soportar la carga del sistema.
- La disponibilidad de un sistema operativo compatible (Windows, macOS, Linux) para la ejecución del servidor es fundamental.

#### 2.6 Evolución Previsible del Sistema

- En el futuro, se podrían considerar mejoras como:
  - Integración con sistemas de punto de venta (POS) para un manejo más eficiente de las ventas.
  - Funcionalidades de reportes avanzados para análisis de ventas y gestión de inventarios.
  - Módulos adicionales para la gestión de promociones y descuentos.
  - Integración con plataformas de e-commerce para expandir las ventas online.

### 3. Requisitos Específicos



### 3.1 Requisitos Comunes de las Interfaces

#### Interfaces de Usuario

El sistema tendrá una interfaz web intuitiva que incluirá:

- **Modulo Usuarios:** Ingreso del administrador y empleado
- **Pantalla de Inicio:** Resumen del inventario y acceso a funciones principales.
- **Ingreso de Productos:** Formulario con campos para nombre, descripción, precio de compra, precio de venta y stock.
- **Salida de Productos:** Formulario para registrar ventas, que mostrará los productos disponibles y sus precios.
- **Visualización de Stock:** Tabla con búsqueda y filtros por nombre y categoría.

#### Estilo y Colores:

- Colores predominantes: Azul y blanco.
- Tipografía legible y moderna.
- Botones de acción claramente visibles y diferenciados.

#### Interfaces de Hardware

El sistema se ejecutará en servidores estándar que cumplan con los siguientes requisitos:

- Procesador: Mínimo 2 GHz.
- Memoria RAM: Mínimo 4 GB.
- Almacenamiento: SSD de al menos 100 GB.
- Conexión a Internet de al menos 10 Mbps.

#### Interfaces de Software

El sistema se integrará con las siguientes aplicaciones:

- **Sistema de stock:** Para la sincronización de datos financieros.
  - **Descripción:** Software utilizado para llevar la gestión de inventario de la licorera.
  - **Propósito:** Integrar datos de stock.
  - **Definición de interfaz:** JSON para intercambiar información entre aplicaciones.

### 3.2 Interfaces de Comunicación



El sistema utilizará el protocolo HTTP/HTTPS para la comunicación entre el cliente y el servidor. Además, se pueden implementar APIs RESTful para la comunicación con otros sistemas.

### 3.3 Requisitos Funcionales

No. de Requisito	Nombre de Requisito	Tipo	Prioridad	Responsable
RF1	Ingreso de Productos	Funcional	Alta	Desarrollador
RF2	Salida de Productos	Funcional	Alta	Desarrollador
RF3	Visualización de Stock	Funcional	Alta	Desarrollador

#### Comprobación de Validez de las Entradas

- Validar que todos los campos requeridos estén completos.
- Comprobar que los precios y cantidades sean números positivos.

#### Secuencia Exacta de Operaciones

1. Ingreso de producto: Ingreso de datos, validación, confirmación de almacenamiento.
2. Registro de salida: Selección de producto, validación de stock, actualización de base de datos.

#### Respuesta a Situaciones Anormales

- Mostrar mensajes de error claros cuando los datos de entrada son inválidos.
- Registrar errores en un log para revisión posterior.

#### Parámetros

- Precio de compra y venta se almacenará como tipo decimal.





- El Stock se almacenará como tipo entero.

**Generación de Salidas**

- Listados de productos en formato HTML y posibilidad de exportar a CSV.
- Reportes de ventas en formato PDF.

**Relaciones entre Entradas y Salidas**

- La salida de stock debe reflejar la cantidad ingresada menos las salidas registradas.

**3.4 Requisitos No Funcionales**

No. de Requisito	Nombre de Requisito	Tipo	Prioridad	Responsable
RNF 1	Rendimiento	No funcional	Alta	Desarrollador
RNF2	Seguridad	No funcional	Alta	Desarrollador

**Requisitos de Rendimiento**

- El sistema debe ser capaz de manejar al menos 1000 transacciones por día.
- El 95% de las transacciones deben realizarse en menos de 2 segundos.

**Seguridad**

- Uso de HTTPS para asegurar la transmisión de datos.
- Implementación de autenticación por roles para acceder a funciones críticas.

**Fiabilidad**

- El sistema debe tener un tiempo de inactividad permitido de menos del 1% al mes.

**Disponibilidad**

- El sistema debe estar disponible el 99% del tiempo, con mantenimientos programados notificados con anticipación.



## Mantenibilidad

- El código debe estar documentado y seguir estándares de codificación para facilitar futuras modificaciones.

## Portabilidad

- El sistema debe ser independiente del hardware y sistema operativo, utilizando tecnologías web estándar.

## 3.5 Otros Requisitos

- **Requisitos Legales:** Cumplimiento con normativas locales de manejo de datos y contabilidad.
- **Requisitos Culturales:** Considerar la adaptación a diferentes idiomas en caso de expansión.

## 4. Apéndices

### A. Glosario de Términos

- **API (Interfaz de Programación de Aplicaciones):** Conjunto de reglas y protocolos para la interacción entre diferentes software.
- **Frontend:** Parte de la aplicación que interactúa directamente con el usuario.
- **Backend:** Parte del sistema que maneja la lógica de negocio, base de datos y procesamiento de datos.
- **CRUD:** Acrónimo de Crear, Leer, Actualizar y Eliminar, operaciones básicas de manipulación de datos.

### B. Diagramas Adicionales

- **Diagrama de Clases:** Puede incluir un diagrama que muestre las relaciones entre las clases del sistema, como Producto, Distribuidor y Transacción.
- **Diagrama de Secuencia:** Un diagrama que ilustre cómo se manejan las interacciones entre el usuario y el sistema durante el ingreso y salida de productos.

### C. Referencias

- Documentación de Laravel: [Manual PHP](#)

### D. Consideraciones sobre Escalabilidad



- **Escalabilidad Vertical:** Consideraciones para aumentar la capacidad del servidor (RAM, CPU) a medida que aumenta la carga.
- **Escalabilidad Horizontal:** Planificación para implementar múltiples instancias del sistema para balancear la carga.

#### E. Plan de Pruebas

- **Pruebas Unitarias:** Implementación de pruebas para funciones individuales del sistema.
- **Pruebas de Integración:** Verificación de que diferentes partes del sistema funcionan juntas correctamente.
- **Pruebas de Usuario:** Evaluaciones realizadas por usuarios finales para asegurar que el sistema cumpla con las expectativas.

#### F. Estrategia de Mantenimiento

- **Mantenimiento Preventivo:** Actualizaciones regulares del sistema y revisiones de seguridad.
- **Mantenimiento Correctivo:** Procedimientos para abordar errores o problemas reportados por los usuarios.