

Documentación de Software.

Manual de Usuario

Descripción del Software

El software de StockManager fue desarrollado para satisfacer la necesidad de llevar un control de las entradas y salidas que se tienen del inventario de la empresa donde se utilice.

Para acceder al software es necesario utilizar un navegador y entrar al sitio proporcionado por el administrador del sistema.

Para poder utilizar el software es necesario contar con un usuario registrado en la base de datos del sistema, en caso de no tener alguno, favor de contactar al encargado del sistema para que le asigne uno.

Dentro de este software es posible llevar catálogos de productos, clientes y proveedores, esto para poder realizar un registro tanto de ventas como de compras de productos, para así tener un control del inventario para posteriormente poder generar reportes de los movimientos realizados del stock.

Perfiles de Usuario

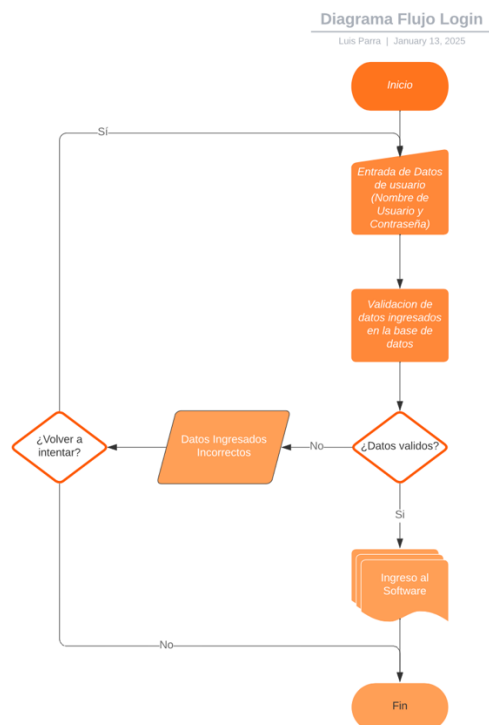
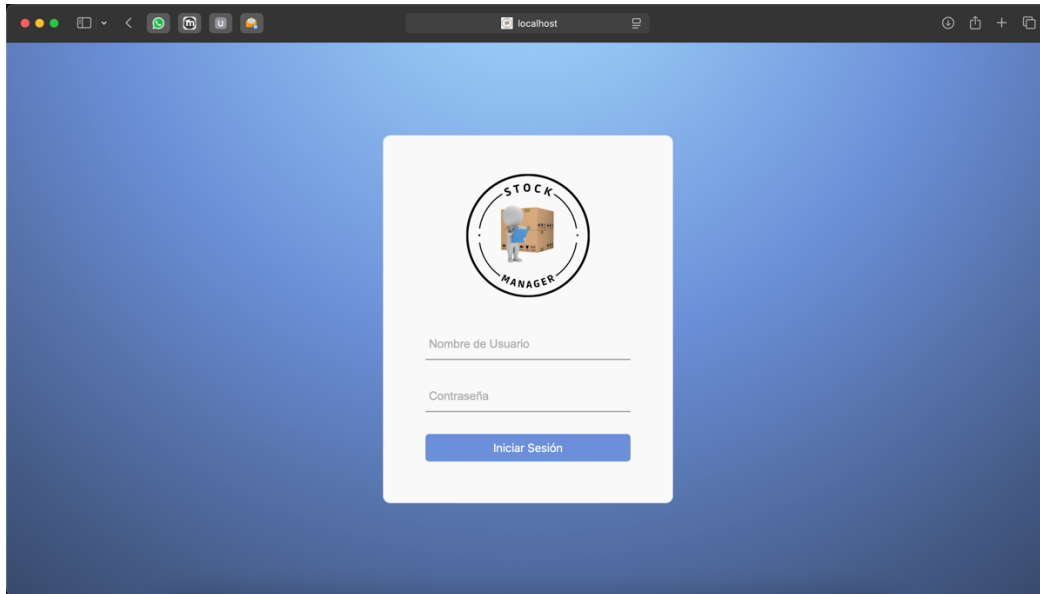
- Soporte:
 - ABC de usuarios.
 - ABC de catálogos (productos, clientes, proveedores).
 - Registro de movimientos (compras, ventas).
 - Generación de reportes

- Usuarios
 - ABC de catálogos (productos, clientes, proveedores).
 - Módulos de movimientos (compras, ventas).
 - Generación de reportes

Funcionamiento del Software

- Inicio de Sesión:

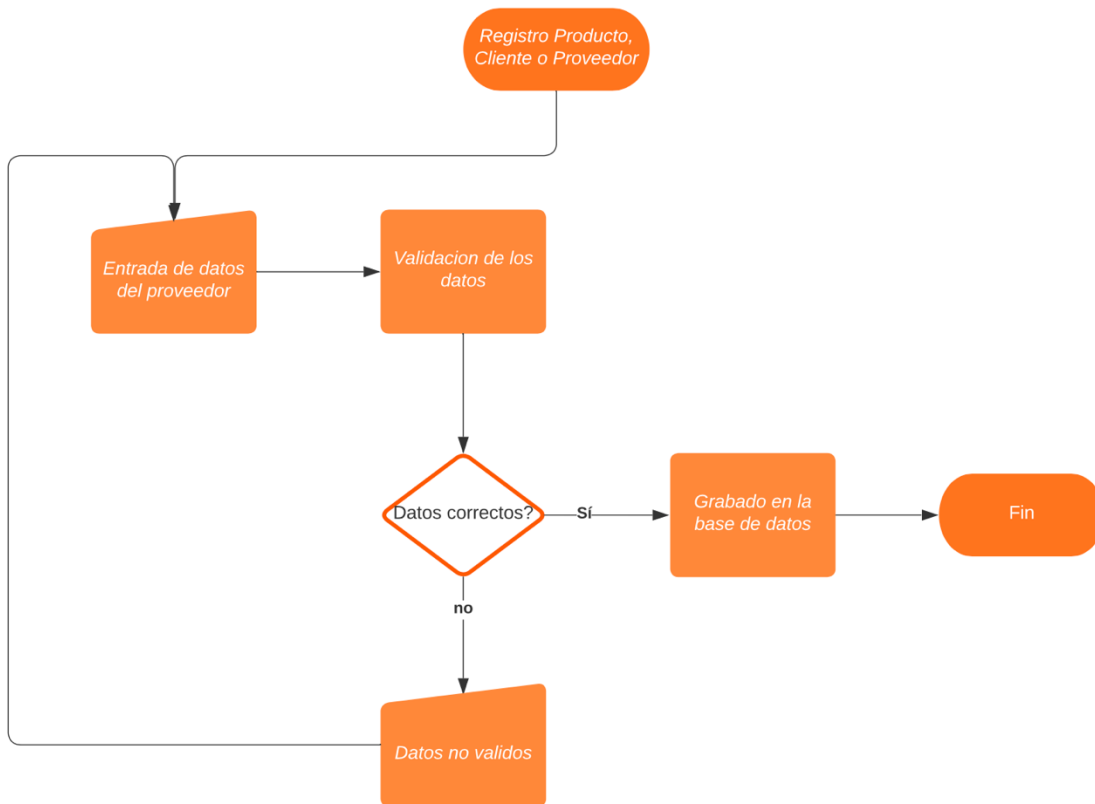
El usuario accede al sitio proporcionado, introduce su nombre de usuario y contraseña registrados en la base de datos.



- **Gestión de Inventario:**
Los usuarios pueden registrar entradas de producto en tiempo real.
- **Catálogos:**
Los catálogos permiten gestionar información (Altas, Bajas y Consultas) de:
 - Productos
 - Clientes
 - Proveedores

Registro de Información en Catálogos

Luis Parra | January 13, 2025



- **Reportes:**
Generación de reportes personalizados sobre movimientos del inventario (por fechas, productos o usuarios).

Glosario de Términos

- **Inventario:**
Conjunto de productos disponibles para la venta o consumo dentro de la empresa.
- **Catálogo:**
Base de datos que contiene información organizada de productos, clientes o proveedores.
- **Reporte:**
Documento generado por el sistema que resume los movimientos realizados en el inventario.
- **Proveedor:**
Entidad o persona que suministra productos o servicios a la empresa.
- **Entrada:**
Registro de un nuevo producto o una reposición en el inventario.
- **Salida:**
Registro de la disminución de productos en el inventario, ya sea por venta o consumo interno.

Soporte Técnico

- Correo electrónico: soporte@stockmanager.com.
- Teléfono: 33 1754 6787 (lunes a viernes, 9:00 AM a 6:00 PM).
- **Información Necesaria:**
 - Nombre del usuario.
 - Descripción detallada del problema.
 - Captura de pantalla o código del error (si es posible).

Tiempo de Respuesta:

- Respuesta inicial: 24 horas hábiles.
- Resolución de problemas críticos: 48 horas hábiles.

Actualizaciones:

- Actualizaciones del sistema disponibles a través del sitio oficial.

Manual Técnico

Requerimientos mínimos de Hardware

- Servidor:
 - 480gb almacenamiento (SSD de preferencia)
 - 16GB memoria RAM
 - Intel Core i5-12400s o Ryzen 5-8500g (ambos incluyen gráficos integrados)
 - Puerto ethernet
- Cliente:
 - 240gb almacenamiento (SSD de preferencia)
 - 8gb memoria RAM
 - Intel Core i3-12100s o Ryzen 3-3300u (ambos incluyen gráficos integrados)
 - Puerto ethernet

Requerimientos de Software

- Servidor:
 - Sistema Operativo Linux (Preferencia) o Windows (versión server o pro)
 - Docker con Mysql8 y node.js
- Cliente:
 - Sistema Operativo Linux, Windows o MacOS (cual sea de su preferencia)
 - Navegador de su preferencia

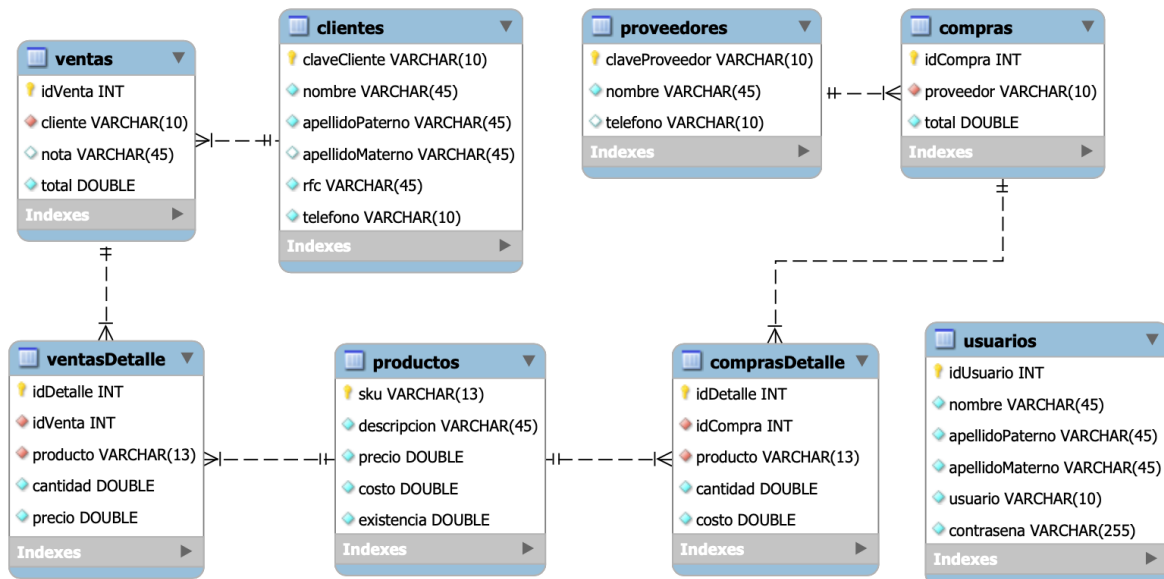
Tecnologías del Sistema

- BackEnd:
 - Node.js
 - MySQL
- FrontEnd:
 - HTML, CSS y JavaScript
 - jQuery y AJAX

- Servidor:
 - o Node.js
- Navegador Compatible:
 - o Safari, Brave, Opera, Google Chrome, Mozilla Firefox, Microsoft Edge.
- Conexión a Internet

Bases de datos

- Diagrama entidad relación:



- Script:
 - MySQL Script generated by MySQL Workbench
 - Tue Dec 3 20:52:27 2024
 - Model: New Model Version: 1.0
 - MySQL Workbench Forward Engineering

```

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
    
```

```
SET                                @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_D
ATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTIT
UTION';
```

```
-----
-- Schema stockManager
-----
```

```
-----
-- Schema stockManager
-----
```

```
CREATE SCHEMA IF NOT EXISTS `stockManager` DEFAULT CHARACTER SET
utf8 ;
USE `stockManager` ;
```

```
-----
-- Table `stockManager`.`productos`
-----
```

```
CREATE TABLE IF NOT EXISTS `stockManager`.`productos` (
  `sku` VARCHAR(13) NOT NULL,
  `descripcion` VARCHAR(45) NOT NULL,
  `precio` DOUBLE NOT NULL DEFAULT 0.0,
  `costo` DOUBLE NOT NULL DEFAULT 0.0,
  `existencia` DOUBLE NOT NULL DEFAULT 0.0,
  PRIMARY KEY (`sku`))
ENGINE = InnoDB;
```

```
-----
-- Table `stockManager`.`clientes`
-----
```

```
CREATE TABLE IF NOT EXISTS `stockManager`.`clientes` (
  `claveCliente` VARCHAR(10) NOT NULL,
  `nombre` VARCHAR(45) NOT NULL,
  `apellidoPaterno` VARCHAR(45) NOT NULL,
  `apellidoMaterno` VARCHAR(45) NULL,
  `rfc` VARCHAR(45) NOT NULL,
```

```
`telefono` VARCHAR(10) NOT NULL,  
PRIMARY KEY (`claveCliente`))  
ENGINE = InnoDB;
```

```
-----  
-- Table `stockManager`.`proveedores`  
-----
```

```
CREATE TABLE IF NOT EXISTS `stockManager`.`proveedores` (  
  `claveProveedor` VARCHAR(10) NOT NULL,  
  `nombre` VARCHAR(45) NOT NULL,  
  `telefono` VARCHAR(10) NULL,  
  PRIMARY KEY (`claveProveedor`))  
ENGINE = InnoDB;
```

```
-----  
-- Table `stockManager`.`ventas`  
-----
```

```
CREATE TABLE IF NOT EXISTS `stockManager`.`ventas` (  
  `idVenta` INT NOT NULL AUTO_INCREMENT,  
  `cliente` VARCHAR(10) NOT NULL,  
  `nota` VARCHAR(45) NULL,  
  `total` DOUBLE NOT NULL,  
  PRIMARY KEY (`idVenta`),  
  INDEX `fk_cliente_idx` (`cliente` ASC),  
  CONSTRAINT `fk_cliente`  
    FOREIGN KEY (`cliente`)  
      REFERENCES `stockManager`.`clientes` (`claveCliente`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
-----  
-- Table `stockManager`.`ventasDetalle`  
-----
```

```
CREATE TABLE IF NOT EXISTS `stockManager`.`ventasDetalle` (  

```



```
`idDetalle` INT NOT NULL AUTO_INCREMENT,
`idVenta` INT NOT NULL,
`producto` VARCHAR(13) NOT NULL,
`cantidad` DOUBLE NOT NULL,
`precio` DOUBLE NOT NULL,
PRIMARY KEY (`idDetalle`),
INDEX `fk_idVenta_idx` (`idVenta` ASC),
INDEX `fk_producto_idx` (`producto` ASC),
CONSTRAINT `fk_idVenta`
  FOREIGN KEY (`idVenta`)
    REFERENCES `stockManager`.`ventas` (`idVenta`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
CONSTRAINT `fk_producto`
  FOREIGN KEY (`producto`)
    REFERENCES `stockManager`.`productos` (`sku`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

-----
-- Table `stockManager`.`compras`
-----

CREATE TABLE IF NOT EXISTS `stockManager`.`compras` (
  `idCompra` INT NOT NULL AUTO_INCREMENT,
  `proveedor` VARCHAR(10) NOT NULL,
  `total` DOUBLE NOT NULL,
  PRIMARY KEY (`idCompra`),
  INDEX `fk_proveedor_idx` (`proveedor` ASC),
  CONSTRAINT `fk_proveedor`
    FOREIGN KEY (`proveedor`)
      REFERENCES `stockManager`.`proveedores` (`claveProveedor`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

```
-----  
-- Table `stockManager`.`comprasDetalle`  
-----
```

```
CREATE TABLE IF NOT EXISTS `stockManager`.`comprasDetalle` (  
  `idDetalle` INT NOT NULL AUTO_INCREMENT,  
  `idCompra` INT NOT NULL,  
  `producto` VARCHAR(13) NOT NULL,  
  `cantidad` DOUBLE NOT NULL,  
  `costo` DOUBLE NOT NULL,  
  PRIMARY KEY (`idDetalle`),  
  INDEX `fk_producto2_idx` (`producto` ASC),  
  INDEX `fk_compra_idx` (`idCompra` ASC),  
  CONSTRAINT `fk_producto2`  
    FOREIGN KEY (`producto`)  
      REFERENCES `stockManager`.`productos` (`sku`)  
      ON DELETE NO ACTION  
      ON UPDATE NO ACTION,  
  CONSTRAINT `fk_compra`  
    FOREIGN KEY (`idCompra`)  
      REFERENCES `stockManager`.`compras` (`idCompra`)  
      ON DELETE NO ACTION  
      ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
-----  
-- Table `stockManager`.`usuarios`  
-----
```

```
CREATE TABLE IF NOT EXISTS `stockManager`.`usuarios` (  
  `idUsuario` INT NOT NULL AUTO_INCREMENT,  
  `nombre` VARCHAR(45) NOT NULL,  
  `apellidoPaterno` VARCHAR(45) NOT NULL,  
  `apellidoMaterno` VARCHAR(45) NOT NULL,  
  `usuario` VARCHAR(10) NOT NULL,  
  `contrasena` VARCHAR(255) NOT NULL,  
  PRIMARY KEY (`idUsuario`))  
ENGINE = InnoDB;
```

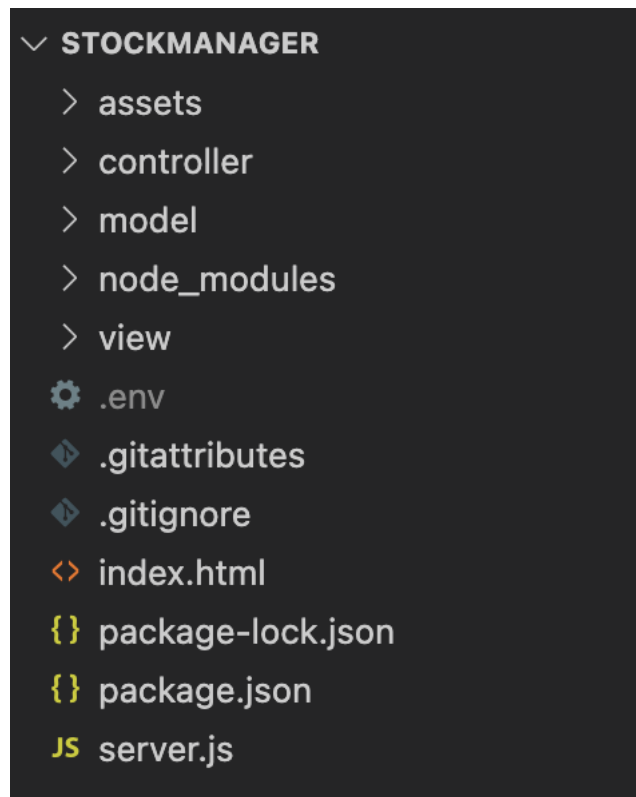
```
SET SQL_MODE=@OLD_SQL_MODE;  
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;  
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

Servidores

Para el servidor se necesita tener corriendo el servicio de NodeJs y MySQL, es posible levantar los servicios utilizando contenedores de Docker.

Arquitectura del Sistema

El proyecto usa la estructura de Modelo Vista Controlador (MVC) por lo cual, si es necesario realizar alguna implementación, será fácil hacerlo por la organización que se tiene de todas las clases.




Matrices de Pruebas

	Fecha: 06/01/2025	Ambiente: Desarrollo
	Caso de Uso: Inicio de sesión	Iniciativa: StockManager


HU:	El usuario podrá iniciar sesión al sistema con usuario y contraseña validos	Sprint: N/A
------------	---	--------------------

ID	Nombre	Datos de entrada	Resultado esperado	Aprobado		Comentarios
				SI	NO	
CP1	Inicio de sesión exitoso	Usuario: soporte Password: hola1234	El usuario ingresa correctamente al sitio, se despliega menu de opciones.			
CP2	Inicio de sesión usuario vacio	Usuario: Password: hola1234	El sistema indica una alerta de que los datos ingresados son incorrectos			
CP3	Inicio de sesión password vacio	Usuario: soporte Password:	El sistema indica una alerta de que los datos ingresados son incorrectos			

	Fecha: 06/01/2025	Ambiente: Desarrollo
	Caso de Uso: Catalogo Productos	Iniciativa: StockManager


HU:	usuario podrá realizar movimientos en el catalogo de productos con exito (Altas, bajas y Consulta)	Sprint: N/A
------------	--	--------------------

ID	Nombre	Datos de entrada	Resultado esperado	Aprobado		Comentarios
				SI	NO	
CP1	Alta de Producto con datos completos y de tipo correcto		Se graba la informacion correctamente en la base de datos y muestra aviso de grabado exitoso			
CP2	Alta de Producto con datos incompletos		El sistema debe arrojar un aviso de que los datos estan incompletos y no debe grabar el producto hasta que este completo			
CP3	Alta de Producto con datos de tipo incorrecto		El sistema debe arrojar un aviso de que los datos son de un tipo erroneo y no debe grabar el producto hasta que esten correctos			
CP4	Consulta de Producto	Clave de producto: Iphone 14	Se encuentra el producto buscado y muestra su informacion, en caso de no encontrarlo muestra aviso de que no se encontró			
CP5	Modificacion de Producto		Modificar algun valor del producto y ver que al grabarlo, si actualice su informacion			
CP6	Baja de Producto	N/A	Eliminacion del producto en la base de datos			

	Fecha: 06/01/2025	Ambiente: Desarrollo
	Caso de Uso: Catalogo Clientes	Iniciativa: StockManager


HU:	El usuario podrá realizar movimientos en el catalogo de clientes con exito (Altas, bajas y Consulta)	Sprint: N/A
------------	--	--------------------

ID	Nombre	Datos de entrada	Resultado esperado	Aprobado		Comentarios
				SI	NO	
CP1	Alta de Cliente con datos completos y de tipo correcto		Se graba la informacion correctamente en la base de datos y muestra aviso de grabado exitoso			
CP2	Alta de Cliente con datos incompletos		El sistema debe arrojar un aviso de que los datos estan incompletos y no debe grabar el cliente hasta que este completo			
CP3	Alta de Cliente con datos de tipo incorrecto		El sistema debe arrojar un aviso de que los datos son de un tipo erroneo y no debe grabar el cliente hasta que esten correctos			
CP4	Consulta de Cliente	Clave de cliente: 01	Se encuentra el cliente buscado y muestra su informacion, en caso de no encontrarlo muestra aviso de que no se encontró			
CP5	Modificacion de Cliente		Modificar algun valor del cliente y ver que al grabarlo, si actualice su informacion			
CP6	Baja de Cliente	N/A	Eliminacion del cliente en la base de datos			

	Fecha: 06/01/2025	Ambiente: Desarrollo
	Caso de Uso: Catalogo Proveedores	Iniciativa: StockManager


HU: Usuario podrá realizar movimientos en el catalogo de proveedores con éxito (Altas, bajas y Consultas) **Sprint:** N/A

ID	Nombre	Datos de entrada	Resultado esperado	Aprobado		Comentarios
				SI	NO	
CP1	Alta de Proveedor con datos completos y de tipo correcto		Se graba la informacion correctamente en la base de datos y muestra aviso de grabado exitoso			
CP2	Alta de Proveedor con datos incompletos		El sistema debe arrojar un aviso de que los datos estan incompletos y no debe grabar el proveedor hasta que este completo			
CP3	Alta de Proveedor con datos de tipo incorrecto		El sistema debe arrojar un aviso de que los datos son de un tipo erroneo y no debe grabar el proveedor hasta que esten correctos			
CP4	Consulta de Proveedor	Clave de proveedor: 01	Se encuentra el proveedor buscado y muestra su informacion, en caso de no encontrarlo muestra aviso de que no se encontró			
CP5	Modificacion de Proveedor		Modificar algun valor del proveedor y ver que al grabarlo, si actualice su informacion			
CP6	Baja de Proveedor	N/A	Eliminacion del proveedor en la base de datos			

	Fecha: 06/01/2025	Ambiente: Desarrollo
	Caso de Uso: Cierre de sesión	Iniciativa: StockManager

HU: El usuario podrá cerrar sesión dentro del sistema **Sprint:** N/A

ID	Nombre	Datos de entrada	Resultado esperado	Aprobado		Comentarios
				SI	NO	
CP1	Cierre de sesión exitoso	N/A	El usuario cierra su sesion y en automatico se redirige a la pantalla del login			

	Fecha: 10/01/2025	Ambiente: Desarrollo
	Caso de Uso: Registro de Entrada	Iniciativa: StockManager

HU: El usuario podrá registrar entradas de productos **Sprint:** N/A

ID	Nombre	Datos de entrada	Resultado esperado	Aprobado		Comentarios
				SI	NO	
CP1	Registro de Compra con datos correctos	Proveedor: Apple Producto: Iphone 14 MacBook Pro 2018	El usuario registra una entrada de productos correctamente			
CP2	Registro de Compra con datos incorrectos	Proveedor: Xiaomi Producto: Iphone 14 Lenovo Thinkpad	El sistema arroja una alerta de que los productos no concuerdan con el proveedor			

	Fecha: 10/01/2025	Ambiente: Desarrollo
	Caso de Uso: Registro de Salida	Iniciativa: StockManager

HU: El usuario podrá registrar entradas de productos **Sprint:** N/A

ID	Nombre	Datos de entrada	Resultado esperado	Aprobado		Comentarios
				SI	NO	
CP1	Registro de Salida con datos correctos	Cliente: Edgard Parra Producto: Iphone 16 MacBook Pro M4	El usuario registra una Salida de productos correctamente			
CP2	Registro de Salida con datos incorrectos	Proveedor: Josesito Producto: Iphone 14 Lenovo Thinkpad	El sistema arroja una alerta de que los el cliente no existe			