**Analisis de Software (Sistema de ventas de un alibreria)**

1. Analisis del Sistema :

1.1Descripción del sistema: El sistema de ventas de librería permite gestionar el inventario de libros, registrar ventas y mantener información de los clientes. Este sistema se desarrollará utilizando PHP para el backend, JavaScript y DataTables para la parte frontend.

1.2Requisitos del Sistema:

Funcionales:

* Gestión de Inventario: Permitir agregar, editar y eliminar libros del inventario.
* Registro de Ventas: Registrar la venta de libros, incluyendo la información del cliente y detalles de la venta.
* Visualización de Inventario y Ventas: Permitir la visualización del inventario de libros y el historial de ventas utilizando DataTables.

No Funcionales:

* Seguridad: Proteger las operaciones con autenticación de usuario.
* Usabilidad: Interfaz intuitiva y fácil de usar.
* Rendimiento: El sistema debe ser capaz de manejar múltiples usuarios simultáneamente sin afectar el rendimiento.

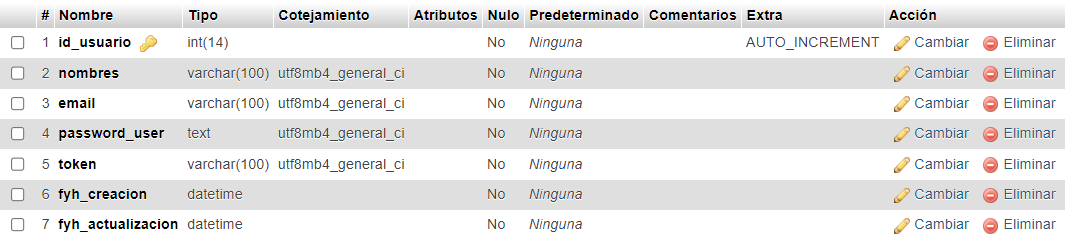
1.3Diagramas de Software:

Diagrama de Casos de Uso:

Diagrama de Clases:

1.4 Base de Datos:

Tablas Principales:



1. Documentación del Proyecto

2.1Estructura del Proyecto:

/sistema-ventas-libreria

│

├── /app

│ ├── config.php

│ ├── CarInventory.php

│ ├── Sales.php

│ ├── Customer.php

│ ├── tests

│ │ ├── UnitTest.php

│ │ └── IntegrationTest.php

│

├── /public

│ ├── index.php

│ ├── js

│ │ └── script.js

│ ├── css

│ │ └── styles.css

│

├── /temp

│ ├── header.php

│ └── footer.php

│

└── database.sql

2.2 Uso del Sistema:

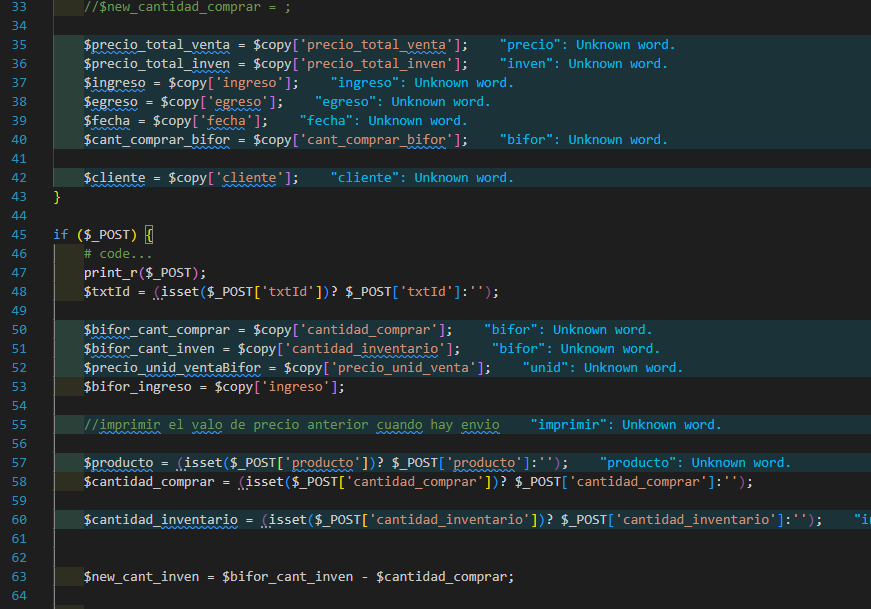
* Agregar un Libro al Inventario:

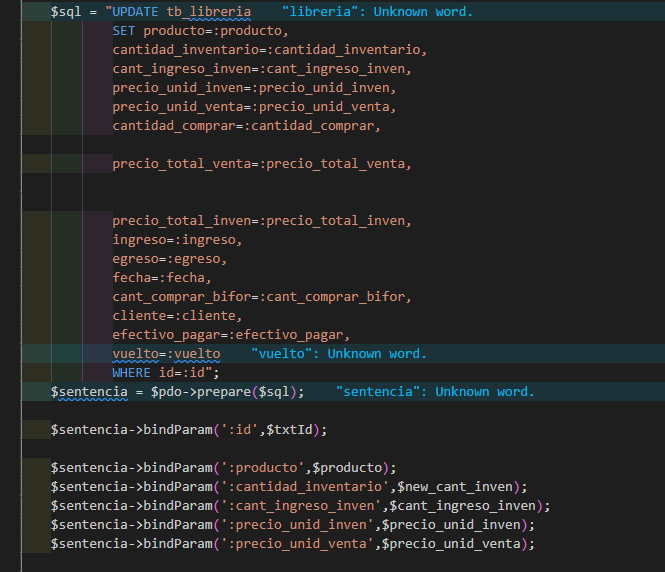
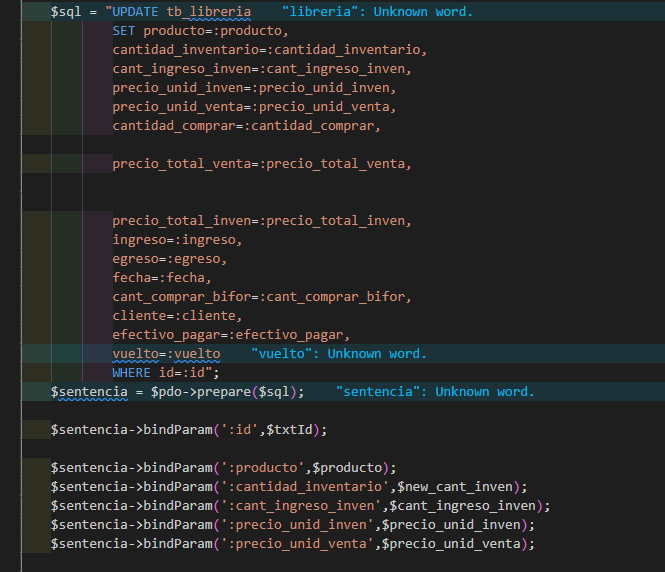
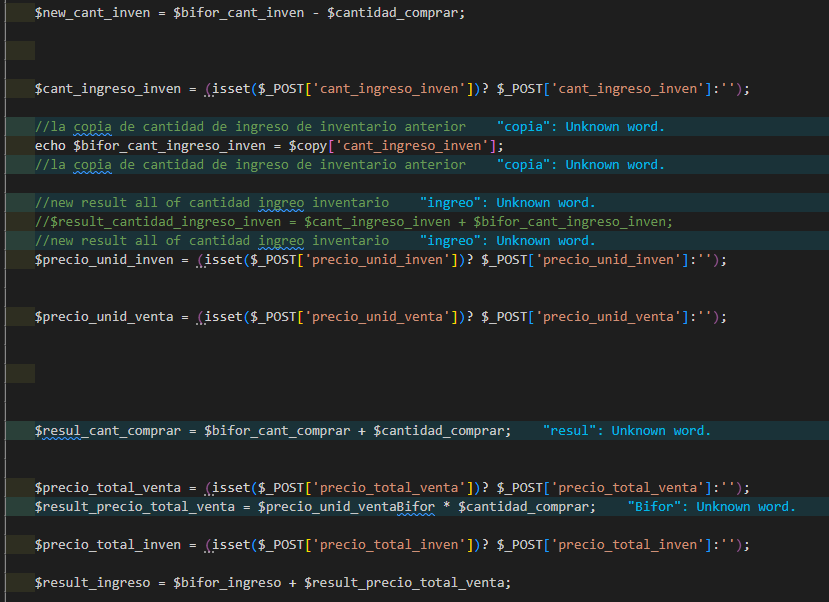
* Navegar a la sección "Agregar Libro".
* Llenar el formulario con los detalles del libro y subir una imagen.
* Hacer clic en "Agregar".
* Registrar una Venta:

* Navegar a la sección "Registrar Venta".
* Seleccionar un libro del inventario.
* Llenar el formulario con los detalles del cliente y la cantidad.
* Hacer clic en "Registrar Venta".
* Visualizar Inventario y Ventas:

* Navegar a la sección "Inventario" o "Ventas".
* Usar las funcionalidades de DataTables para filtrar y buscar registros.

3. Pruebas Unitarias y de Integración:

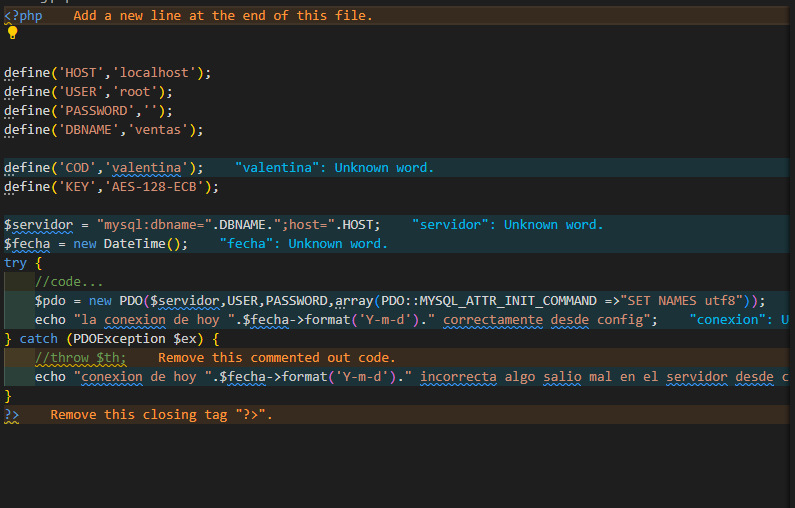




### **Pruebas de Integración**

**PHP:**

Vamos a suponer que tienes un sistema que gestiona productos y una base de datos para almacenarlos. Aquí hay un ejemplo de prueba de integración utilizando PHPUnit y una base de datos SQLite para pruebas:



### **Pruebas de Caja Negra**

* El sistema debe permitir agregar un libro al inventario.

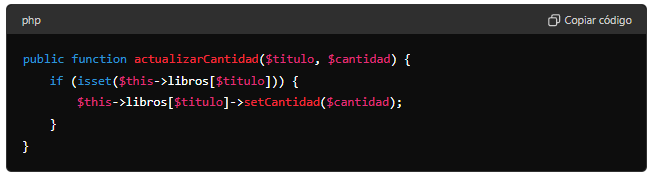
Caso de Prueba 1: - Entrada: Título: "Don Quijote", Cantidad: 10 - Resultado Esperado: El libro "Don Quijote" está en el inventario con una cantidad de 10.

Caso de Prueba 2: - Entrada: Título: "Cien Años de Soledad", Cantidad: 5 - Resultado Esperado: El libro "Cien Años de Soledad" está en el inventario con una cantidad de 5.

Caso de Prueba 3: - Entrada: Título vacío, Cantidad: 5 - Resultado Esperado: El sistema muestra un mensaje de error indicando que el título no puede estar vacío.

### **4.Pruebas de Caja Blanca**

#### **Función actualizarCantidad**



### **Pruebas de Caja Gris**

#### **Requisito Funcional:**

1. **Requisito Funcional:**
   * El sistema debe permitir actualizar la cantidad de libros en el inventario y obtener la cantidad actualizada.

#### **Casos de Prueba:**

