

UNIVERSIDAD MARIANO GÁLVEZ DE GUATEMALA
FACULTAD DE INGENIERÍA EN SISTEMAS DE INFORMACIÓN



JULIO ABRAHAN CHET CHAMALE

7590-23-5400

GUATEMALA, OCTUBRE DEL 2024

Introducción

Este documento describe la estructura técnica del sistema de inventario desarrollado en Java.

Proporciona información sobre la arquitectura, los componentes clave del código, y las instrucciones para desarrolladores que quieran contribuir o mantener el proyecto.

El inventario describe como está organizado o funciona un sistema de ventas, ya que puedes gestionar, productos, categorías y existencias.

- Requisitos Técnicos

Java : Versión 8 o superior.

IDE: Cualquier entorno de desarrollo compatible con Java, Eclipse o NetBeans.

Sistema operativo: Windows, macOS o Linux.

Bibliotecas externas: Ninguna, se utilizan las bibliotecas estándar de Java.

- Arquitectura del Sistema

El sistema realiza todas las operaciones desde un único archivo ejecutable. Está basado en un modelo de consola, lo que significa que todas las interacciones con el usuario se realizan mediante la entrada y salida en terminales.

- Estructura del Código

El proyecto se organiza en un conjunto de paquetes y clases que separan las diferentes funcionalidades.

- Paquetes Principales

com.mycompany.inventario: Contiene la clase principal Inventario y el menú principal del sistema.

productos: Maneja la lógica de gestión de productos (altas, bajas, modificaciones).

existencias: Contiene la lógica para mostrar y gestionar las existencias.

categorías: Gestiona las categorías de los productos.

- Clases

Inventario: Contiene el método main() y el menú principal.

GestionProductos: Proporciona los métodos para agregar, editar y eliminar productos.

ControlExistencias: Muestra las existencias de los productos almacenados.

Categorías: Permite, agregar, modificar, eliminar categorías. También se relaciona con el producto por medio de una categoría en específico a la hora de agregar un producto.

- Archivos

El sistema utiliza archivos de texto plano para almacenar datos.

productos.txt: Almacena los productos en formato:

```
1 | Cargadores | Cargadores multicolor | 6 | Q 85.0 | Q 110.0 | 1 | Cargadores
2 | Cargadores | cargadores, multicolor | 8 | Q 85.0 | Q 110.0 | 1 | Cargadores
3 | Servilletas | Servilletas de cintura | 12 | Q 28.0 | Q 35.0 | 2 | Servilletas de 4 cuartas
4 | Servilletas | servilletas de 3 cuartas multicolor, cintura | 12 | Q 16.0 | Q 25.0 | 3 | Servilletas de 3 cuartas
```

categorias.txt: Almacena las categorías en formato de texto, una categoría por línea.

```
1 | Cargadores
2 | Servilletas de 4 cuartas
3 | Servilletas de 3 cuartas
4 | Cortes
```

- Flujo del Programa

Inicio del programa: El usuario accede al menú principal desde la clase Inventario. Se muestra un menú con las siguientes opciones:

1. Gestión de productos.
2. Control de existencias.
3. Pedidos de compra.
4. Informes y estadísticas.
5. Categorías.
6. Salir del sistema.

Gestión de productos: Desde el menú, el usuario puede agregar, editar o eliminar productos. Estos se almacenan en el archivo productos.txt.

Control de existencias: Se leen los productos del archivo productos.txt y se muestra la cantidad disponible de cada producto y su descripción.

Gestión de categorías: El usuario puede agregar nuevas categorías o listar las existentes. Estas se almacenan en categorias.txt.

Salir del sistema: Finaliza la ejecución del programa.

- Tecnologías Utilizadas

Lenguaje: Java 8 o superior.

Entrada y Salida: Uso de FileReader, BufferedReader, FileWriter, y BufferedWriter para manipular archivos.

Consola: La interacción con el usuario se maneja mediante Scanner.

- Mantenimiento del Sistema

Agregar Nuevas Funcionalidades

Para agregar nuevas funcionalidades como reportes o gestión de pedidos:

Crear una nueva clase dentro de un paquete como, reportes o gestión de pedidos.

Actualizar el menú principal.

- **Modificar Funcionalidades Existentes**

Para modificar la lógica de productos, trabajar dentro de la clase GestionProductos.

Para cambiar la lógica de las categorías, realizar cambios en Categorías.

- **Depuración**

Si el sistema arroja errores, se pueden utilizar herramientas de depuración integradas en los IDEs para trazar los errores en el código.

- **Consideraciones de Seguridad**

Actualmente, el sistema no tiene medidas de seguridad avanzadas. Para futuras versiones, se sugiere:

Implementar validaciones para evitar la entrada de datos maliciosos.

Introducir controles de acceso para restringir el uso del sistema a usuarios autorizados.