ESCUELA
POLITÉCNICA
NACIONAL

Escuela Politécnica Nacional

Facultad de Ingeniería en Sistemas

Ingeniería en Ciencias de la Computación

Programación II

Estandares de Codificación: Chapter Connect

GRUPO 2

Integrantes:

- David Cuasquer
- Alegria Farinango
- Celeste Gallardo

Fecha de entrega: 12/08/2024

Docente: Ing. Patricio Paccha

2024-A

Índice de Contenido

1. Propósito

Optimizar la gestión de inventario y ventas de una librería mediante la implementación de un sistema de lectura de códigos de barras, mejorando la eficiencia operativa y la experiencia del cliente.

2. Destinatario

Este documento está dirigido principalmente a los estudiantes universitarios que participan en el desarrollo del proyecto de clase sobre la implementación de un sistema de lectura de códigos de barras en una librería.

Además, será de utilidad para cualquier persona interesada en la aplicación de tecnologías modernas para mejorar la eficiencia en negocios minoristas, especialmente en el sector de librerías.

3. Plataforma Java

3.1. Buenas Prácticas

Integración de bibliotecas

- Para incorporar bibliotecas externas, se utiliza la palabra clave "Import" seguida del nombre del paquete. Esto permite acceder a funcionalidades adicionales sin necesidad de reinventar.

Formato del código

- La legibilidad es clave.
- Los nombres de variables y métodos deben ser descriptivos, comenzar con minúsculas y evitar abreviaturas.
- Para propiedades o constantes multipalabra, se usa el guion bajo "_" como separador.
- Las librerías mantienen sus nombres originales para facilitar su identificación.

Encapsulamiento

- Se emplean los modificadores "private", "public" o "protected" para controlar el acceso a los campos.
- Además, se proporcionan métodos getters y setters para acceder y modificar estos campos de manera controlada, manteniendo la integridad de los datos.

Herencia, interfaz y clase abstracta

- Las interfaces se utilizan para definir contratos y permitir implementaciones múltiples.
- Para clases padres o abstractas, se emplean nombres descriptivos con palabras clave como "Abstract" o "Base", clarificando su propósito en la jerarquía de clases.

Sobreescritura y sobrecarga

- Al sobrescribir métodos, se usa la anotación "@Override" para indicar explícitamente la intención.

- Se evita la sobrecarga de métodos con nombres similares para prevenir confusiones en el código.

Manejo de excepciones

- Se implementa "try-catch" para manejar excepciones de forma robusta, limitando su uso a casos estrictamente necesarios.
- Se favorece un código predecible y manejable, utilizando bloques específicos por tipo de excepción y registrando errores para facilitar la depuración.

Interfaz Gráfica

- Se aprovechan interfaces gráficas de librerías de libre uso, equilibrando funcionalidad y eficiencia en el desarrollo.

4. Base de Datos

Plataforma de Almacenamiento

- Para el almacenamiento de datos que serán ocupados en el proyecto se optó por SQL y su extensión SQLite debido a su facilidad de uso y eficiencia operativa.

Manejo de Base de Datos

- Para cada tabla, se configura tanto el DML (Data Manipulación Lenguaje) como el DDL (Data Definición Lenguaje), asegurando un control preciso sobre la estructura y los datos.

Formato de Codificación

- Se adopta el formato CamelCase para nombrar tablas y elementos clave, promoviendo consistencia y legibilidad en la base de datos.
- El uso de CamelCase mejora la legibilidad, establece uniformidad, evita problemas de compatibilidad y permite nombres descriptivos sin excesiva longitud.
- Es crucial mantener la coherencia en todo el proyecto y documentar la convención elegida para el equipo.

Comentarios y documentación

- El código incorpora anotaciones detalladas que explican la funcionalidad de las variables, las etapas lógicas del proceso de ejecución y los mecanismos de gestión de excepciones.
- Esta práctica facilita considerablemente el mantenimiento posterior y mejora la comprensibilidad del código fuente.
- La inclusión de estos comentarios significativos proporciona un contexto valioso para los desarrolladores, permitiendo una interpretación más eficiente de la estructura y el propósito del programa.

Versionamiento

- Se utiliza GitHub como plataforma de control de versiones, permitiendo colaboración eficiente y seguimiento de cambios entre los miembros del equipo.

5. Referencias

[1] Java Foundations. (2010). Java: Estándares de programación. Java Foundations Blog.

[2] Cabrera, E. (2024). Manejo de excepciones en Java. Eudris Cabrera Blog.