

Programación Avanzada

Taller III

Desarrollo de Software Aplicación Herencia

Beat the Rhythm

Descripción



Beat the Rhythm es una tienda de música familiar que abrió sus puertas hace unas semanas. Después de algunos días de arduo trabajo, los propietarios se dieron cuenta de que su sistema de inventario se encontraba obsoleto, ya que se basaba en lápiz y papel, por lo tanto, decidieron llamar a los estudiantes de Programación Avanzada de la Universidad Católica del Norte para que implementen un Sistema de Inventario (SINV) para su tienda de música, utilizando los conocimientos aprendidos sobre programación orientada a objetos, específicamente la aplicación de herencia.

El cliente de este sistema de inventario proporcionó varias indicaciones relacionadas con el sistema:

- El cliente necesita que el sistema tenga una interfaz que permita ver el inventario y modificarlo.
- La tienda tiene múltiples instrumentos a la venta, cada uno con un código único, un precio y un stock para saber cuántos instrumentos quedan disponibles.
- El cliente desea poder separar los instrumentos por tipo: cuerda, percusión y viento (los cuales pueden ser de metal o de madera).

El sistema debe permitir a los trabajadores de la tienda llevar un registro de inventario y acceder a información como el precio y el stock de los instrumentos de manera sencilla, es por ello que el sistema debe contar con un menú que contenga los siguientes apartados:

1. Agregar Instrumento:

- a. El sistema debe permitir al usuario agregar nuevos instrumentos a la tienda mediante un archivo CSV. En caso de no haber instrumentos estos se deben añadir a través de una lectura, pero si ya hay instrumentos registrados en el sistema, se debe actualizar el archivo y agregar los instrumentos del nuevo archivo. Los datos necesarios para ingresar instrumentos son los siguientes:
 - i. Si es un instrumento de Cuerda necesita:
 - 1. Qué instrumento es (Guitarra, Bajo, Violín, Arpa).
 - 2. Tipo de cuerda (Nylon, acero, tripa).
 - 3. Número de cuerdas.
 - 4. Material de construcción (madera, metal).
 - 5. Tipo (acústico, eléctrico).
 - ii. Si es un instrumento de Percusión necesita:
 - 1. Qué instrumento es (Bongó, Cajón, Campanas Tubulares, Bombo).
 - 2. Tipo de percusión (Membranófono, idiófono).
 - 3. Material de construcción (madera, metal, piel).
 - 4. Altura (definida, indefinida).
 - iii. Si es un instrumento de Viento necesita:
 - 1. Qué instrumento es (Trompeta, Saxofón, Clarinete, Flauta travesa).
 - 2. Material (madera, metal).

2. Vender Instrumento:

- a. El sistema debe permitir al usuario vender los instrumentos, para esto se debe ingresar el código del instrumento que se quiere vender.
- b. Si no queda stock del instrumento se debe notificar al usuario con un mensaje que indique que está agotado.
- c. Una vez vendido el instrumento se debe generar una boleta que tenga como mínimo lo siguiente:
 - i. Instrumento vendido.
 - ii. Precio.

3. Consultar Inventario:

- a. El sistema debe permitir al usuario buscar instrumentos que tengan stock en base a qué tipo son y desplegar los instrumentos en su totalidad o por tipo, es decir, cuerda, percusión o viento, también debe darse la posibilidad de buscar mediante el código para que despliegue uno en específico.
- b. Deben desplegarse todos los datos de los instrumentos al buscarlos, esto incluye:
 - i. Código.
 - ii. Precio.

- iii. Instrumento (guitarra, trompeta, Bongo, etc.)
 - iv. Detalles del instrumento en cuestión (si es una guitarra sus atributos como la cantidad de cuerdas que posee).
4. Cierre:
- a. El cierre es la forma que tienen de salir del programa, es importante destacar que al momento de cerrar el sistema y al abrirlo de nuevo, el stock debe quedar igual a como estaba antes de cerrarse el programa.

Entregables

La entrega del taller debe contener:

1. Código completo de la aplicación escrita en Java.
2. Modelo del Dominio: en el archivo **dominio.puml** ubicado en la raíz del proyecto.
3. Diagrama de clases: en el archivo **clases.puml** ubicado en la raíz del proyecto.

Condiciones de Entrega

- La fecha de entrega es a más tardar el día **domingo 21 de mayo a las 18:00 hrs.**
 - **No se aceptarán entregas fuera de plazo.**
- La resolución del taller **debe ser realizada en parejas**, no se aceptarán trabajos personales.
- El IDE utilizado debe ser IntelliJ (Community o Ultimate).
- Consultas sobre el taller serán respondida vía foro en Campus Virtual UCN.
- La entrega del taller **será por medio de GitHub** enviando la URL del repositorio privado al ayudante por medio de un correo electrónico:
 - Prof. Álvaro Castillo: Ayudante Helmer Pizarro (helmer.pizarro@alumnos.ucn.cl).
 - Prof. Pablo Salas: Ayudante Marcelo Céspedes (marcelo.cespedes@alumnos.ucn.cl).
 - Prof. Diego Urrutia: Ayudante Edgardo Ortiz (edgardo.ortiz@alumnos.ucn.cl).
- **La copia del taller será sancionada con nota 1,0 y los antecedentes del caso serán reportados a la jefatura de carrera y al registro curricular.**

La entrega del taller será evaluada por medio de la aplicación de los criterios de evaluación indicados en la tabla 1.

Criterio	Descripción	Valor
JavaDOC	Se utiliza documentación en estándar JavaDoc en cada clase y métodos del programa.	15%
Comentarios	Se utilizan comentarios de una o más líneas en el código explicando su funcionamiento.	10%
POO	El problema es resuelto por medio de la aplicación del paradigma de la orientación al objeto utilizando: clases, atributos, métodos y herencia.	30%
Codificación	El código de la solución se ajusta a los requisitos planteados en el enunciado y se ajusta completamente a los estándares de codificación de Java.	20%
Ejecución	El proyecto compila correctamente, se encuentra libre de errores de ejecución y responde a los requisitos del problema.	20%
Plataforma	Se utiliza una IDE moderna de desarrollo en la solución.	5%

Tabla 1: Criterios de Evaluación.