

Nombre: Ignacio Emmanuel Isaac Medina

Fecha: 09-Dic-2022

Evaluación escrita Semana 3

Programa utilizando JUnit (En colaboración con *Ángeles Teresa Mena Dzul*)

El programa se encuentra en el proyecto *JUnitUsage*, donde se creó un *POJO* correspondiente a un libro y se anotaron los datos que pudieran ser de utilidad en una librería:

- Nombre (String)
- Código (String)
- Editorial (String)
- Lista de formatos disponibles (List<String>)
- Cantidad (int)
- Precio (double)

Además, se agregaron los siguientes métodos:

- Agregar libros.
- Restar libros.
- Modificar precio.
- Obtener el importe correspondiente al total de libros.
- Obtener un descuento, el cual puede ser de consulta o permanente (modifica el precio).
- Agregar un nuevo formato.
- Eliminar un formato.

Las pruebas unitarias que se desarrollaron fueron

1. Crear un libro y que el resultado no fuera null, los formatos coincidieran con los utilizados en el constructor y que el importe total de dinero de libros (precio multiplicado por cantidad) coincidiera con el cálculo manual.
2. Crear un libro con cantidad negativa y precio negativo. Estos campos deben de tener cero.
3. Libro sin título. Se debe de inicializar este campo con la cadena "S.N."
4. Creación de un libro con los campos nulos. Se inicializa un "eBook" (formato), con los demás campos nulos para los String y 0 para los primitivos.
5. Añadir una cantidad negativa de libros. En este caso no se modifica la cantidad.
6. Restar una cantidad negativa de libros o mayor a la que se tiene registrado. En este caso no se modifica la cantidad.
7. Modificar un precio con una cantidad negativa. En este caso no se cambia el valor del precio.
8. Consultar un descuento, el precio no se modifica.
9. Aplicar un descuento, el precio sí se modifica.
10. Agregar un formato repetido, esto no modifica la lista de formatos.

Nombre: Ignacio Emmanuel Isaac Medina

Fecha: 09-Dic-2022

Evaluación escrita Semana 3

11. Eliminar un formato que no tiene el libro, esto no modifica la lista de formatos.

Entre las funciones utilizadas están:

- `assertNotNull`
- `assertLinesMatch`
- `assertEquals`
- `assertTrue`

Implementación de Scrum

- Elegir el responsable del producto, este debe de ver por los intereses generales del proyecto.
- Elegir el equipo, debe de haber personas capaces de llevar a cabo el proyecto.
- Elegir el Scrum Master, esta persona eliminará los obstáculos y le dará el enfoque Scrum al grupo.
- Crear la bitácora del producto, lista de alto nivel con lo necesario para cumplir con misión del proyecto. Puede evolucionar a lo largo del tiempo.
- Afinar y estimar la bitácora del producto, el equipo estima cuanto esfuerzo implica el cumplimiento de la bitácora. Se calcula en tamaño relativo y puede usarse la serie de Fibonacci para darle un valor puntual a las actividades.
- Planear el Sprint, primera de las reuniones de Scrum donde se planean los avances que se quieren lograr en un tiempo inferior a un mes.
- Volver visible el trabajo, mediante la creación de un tablero con las actividades pendientes, en proceso y terminadas.
- Implementar las paradas diarias, donde se explica lo que se hizo ayer, lo que se hará hoy y si hay algún obstáculo presente, todo esto en un tiempo menor a 15 minutos.
- Revisión del sprint, es una junta donde cada integrante del equipo muestra sus avances durante el sprint. Es una reunión abierta y deben asistir todos los interesados en el proyecto. Solo se muestra lo terminado.
- Retrospectiva del Sprint, se busca una retroalimentación de las funcionalidades terminadas y se piensa en lo que marchó bien, cómo se puede mejorar y el proceso que debe de implementar el equipo para mejorar.
- Se repite el ciclo del sprint usando la experiencia obtenida, impedimentos encontrado y mejoras del proceso.

Spring Batch

Se trata de un producto de Spring, cuyo propósito es la automatización del procesamiento de tareas por lotes. Se compone de un Job Repository, Job Launcher, Jobs y Steps.

Nombre: Ignacio Emmanuel Isaac Medina

Fecha: 09-Dic-2022

Evaluación escrita Semana 3

- Job repository: Se encarga de comunicarse con la persistencia de la base de datos, con el propósito de ver el estado de las tareas (Jobs). Aquí se puede ver la información pertinente de los Jobs, como si está en ejecución, su estado, mensaje de finalización, entre otros.
- Job Launcher: Invoca la ejecución de los Jobs.
- Job: Es una tarea que se va a ejecutar.
- Steps: Es una serie de pasos que debe de ejecutar un Job. A su vez se compone de una actividad de lectura, procesamiento y escritura.

La Figura 1 muestra la manera en la que se relacionan los componentes de Spring Batch.

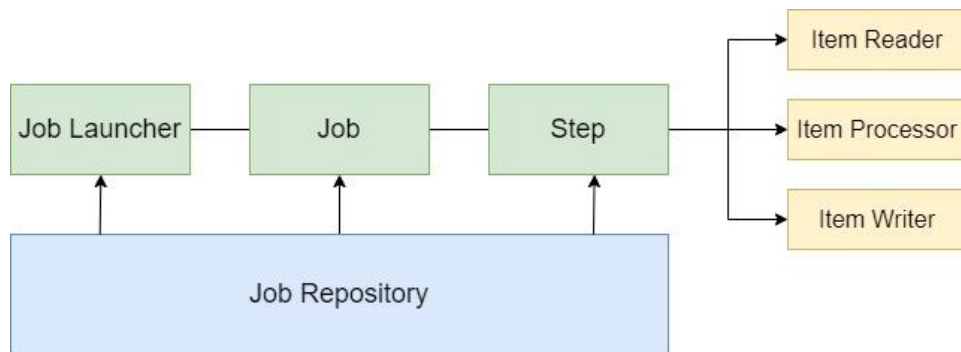


Figura 1.- Diagrama de Spring Batch.

CRUD utilizando JDBC (En colaboración con Ángeles Teresa Mena Dzul)

El programa se encuentra en el proyecto *maven-chef-tracker*.

CRUD utilizando Hibernate y Spring (En colaboración con Ángeles Teresa Mena Dzul)

El programa se encuentra en el proyecto *spring-hibernate-tiendita*. Se realizó el CRUD enfocado a una tienda, por tal motivo el Entity tuvo los siguientes campos:

- Id.
- Nombre.
- Código.
- Descripción.
- Precio.

Respecto al ejemplo proporcionado en el curso, se cambiaron los nombres de las clases, los paquetes y el contexto. Las Figuras 2 y 3 muestran unas capturas de pantalla.

Nombre: Ignacio Emmanuel Isaac Medina

Fecha: 09-Dic-2022

Evaluación escrita Semana 3



Figura 2.- Lista de productos.

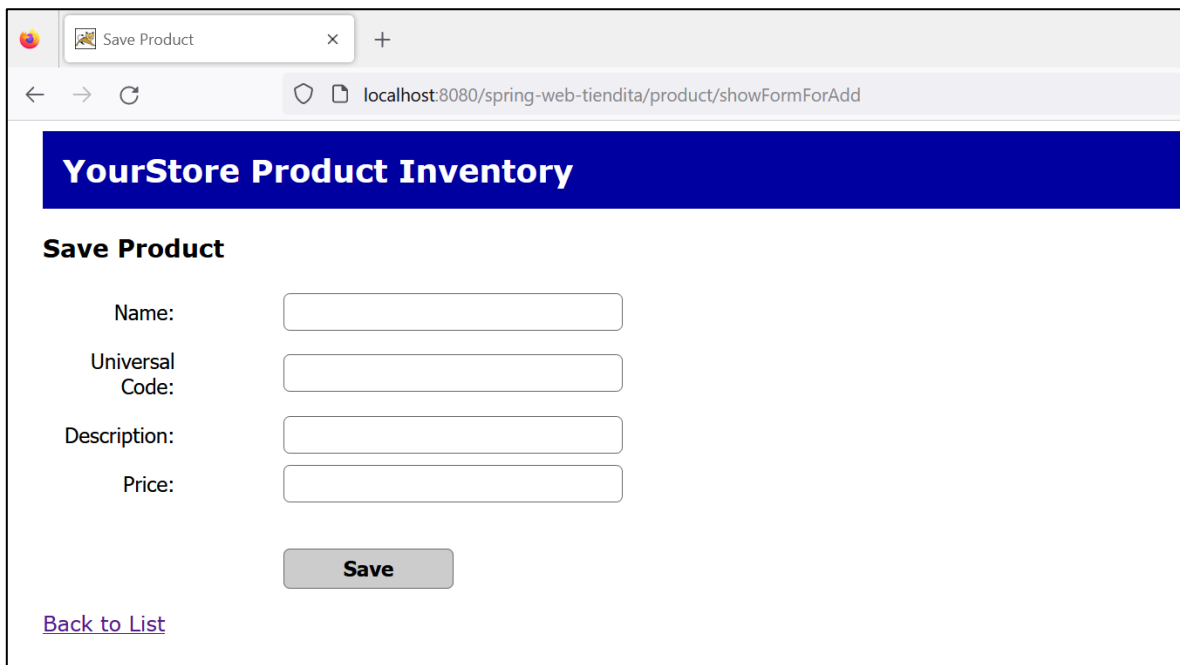


Figura 3.- Agregar un nuevo producto.

Nota: Debido a que me arrojó un problema con la longitud del nombre de algunos archivos del proyecto, se optó por adjuntar de manera complementaria, un archivo rar.