

# *Inventario CFP Pere de Son Gall*



Actividad evaluable 1-Entornos de desarrollo

Marc Marchal ; Joan Bonet ; Miquel Bonet ; Leonie Kleindorp

Día de entrega : 9 de febrero de 2025

# Índice

<b>Inventario CIFP Pere de Son Gall.....</b>	<b>1</b>
Índice.....	2
Descripción del Proyecto.....	3
Distribución y roles del Trabajo (4 Personas):.....	3
Tecnologías utilizadas.....	4
Uso de Git en el Proyecto.....	4
Instrucciones de Uso.....	6
Registro de Reuniones y Decisiones Tomadas.....	6
Sprint 1: Planificación del proyecto.....	6
Sprint 2: Desarrollo del menú principal y funcionalidades básicas.....	6
Sprint 3: Funcionalidades avanzadas del inventario.....	7
Sprint 4: Persistencia de datos, optimización y documentación (5-9 de febrero).....	7

## Descripción del Proyecto

Este proyecto tiene como objetivo desarrollar un sistema de inventario utilizando Java. Durante su desarrollo, hemos implementado distintas funcionalidades, incluyendo la gestión de productos, eliminación, modificación y visualización del inventario. Se ha seguido una metodología ágil con sprints bien definidos para una organización eficiente.

### Distribución y roles del Trabajo (4 Personas):

- **Product Owner (Leonie Kleindorp)**: Responsable de maximizar el valor del producto.
  - Desarrollo del menú principal y funcionalidades básicas.
  - Implementación de funciones avanzadas (modificar, sustituir productos).
  - Pruebas, optimización, documentación y gestión del repositorio.
- **Scrum Master (Marc Marchal)**: Facilita el proceso Scrum i elimina obstacles.
  - Implementación de funciones avanzadas (modificar, sustituir productos).
  - Redimensionamiento de arrays, persistencia de datos.
  - Pruebas, optimización, documentación y gestión del repositorio.
- **Equip de desenvolupament (Joan Bonet y Miquel Bonet)** : Professionals que creen el producte.
  - Implementación de funciones avanzadas (modificar, sustituir productos).
  - Pruebas, optimización, documentación y gestión del repositorio.
  - Mantenimiento de GitHub.



## Tecnologías utilizadas

En este proyecto hemos utilizado los siguientes herramientas y lenguajes para poder trabajar con la máxima eficiencia y organización :

### 1. Herramientas:

- Git
- GitHub
- Trello
- IntelliJ

### 2. Lenguajes:

- Java

## Uso de Git en el Proyecto

Para gestionar el control de versiones y la colaboración en el desarrollo del proyecto, se ha utilizado Git y GitHub.

### 1. Repositorio en GitHub:

- Se ha creado un repositorio en GitHub para alojar el código fuente.
- La estructura inicial incluye carpetas organizadas y un archivo **README.md**.

### 2. Flujo de trabajo con ramas:

- Se ha definido una rama principal (**main**) donde se almacena el código estable.
- Cada nueva funcionalidad o corrección de errores se desarrolla en una rama específica (**feature/nombre-funcion** o **bugfix/nombre-fix**).
- Se realizan pull requests (PR) para fusionar cambios a **main** después de revisión y pruebas.

### 3. Reglas de contribución:

- Se han establecido normas para los commits, siguiendo una estructura clara (**feat:**, **fix:**, **docs:**, **refactor:**).
- Se exige revisión de código antes de fusionar cambios en **main**.
- Se ha implementado la asignación de issues y milestones para una mejor organización.

### 4. Uso de GitHub Actions:

- Se han configurado acciones para ejecutar pruebas automáticas antes de fusionar cambios.
- Verificación de calidad del código con herramientas de análisis estático.

## Estructura del Proyecto

El proyecto está compuesto por los siguientes archivos:

- **Main.java**: Contiene el menú principal y la lógica para interactuar con el usuario.
- **Inventory.java**: Implementa la gestión del inventario con funciones como agregar, eliminar y modificar productos.
- **Helper.java**: Contiene funciones auxiliares para facilitar ciertas operaciones.
- **.git/**: Carpeta interna gestionada por Git para el control de versiones.
- **.gitignore**: Archivo que define qué archivos y carpetas deben ser ignorados por Git.
- **README.md**: Documentación del proyecto con instrucciones de uso y detalles técnicos.
- **docs/**: Carpeta que contiene documentación adicional, incluyendo minutas de reuniones y decisiones tomadas.

El código fuente se encuentra en la carpeta **src**, la documentación en **docs**, y las pruebas en **tests**. Además, Git ha sido parte fundamental de la estructura para el control de versiones y la colaboración.

```
/proyecto-inventario
|-- /src
|   |-- Main.java
|   |-- Inventory.java
|   |-- Helper.java
|-- /docs
|   |-- README.md
|   |-- minutas.md
|-- /tests
|   |-- InventoryTest.java
|-- .gitignore
|-- .github/workflows
```

## Instrucciones de Uso

Al ejecutar el programa, se mostrará un menú con las siguientes opciones:

1. **Añadir producto:** Permite agregar un nuevo producto al inventario.
2. **Eliminar producto por nombre:** Elimina un producto según el nombre ingresado.
3. **Eliminar producto por posición:** Elimina un producto según su índice en el inventario.
4. **Sustituir producto:** Permite reemplazar un producto en una posición específica.
5. **Modificar producto:** Permite cambiar el nombre de un producto existente.
6. **Mostrar productos:** Lista todos los productos en el inventario.
7. **Mostrar resumen del inventario:** Muestra un listado con la cantidad total de cada producto.
8. **Salir:** Finaliza la ejecución del programa.

## Registro de Reuniones y Decisiones Tomadas

Hemos realizado reuniones periódicas para la organización y revisión del proyecto:

- **Reuniones en clase:** Todos los miércoles de 15:00 a 15:55 h, donde discutimos el progreso y planificamos las siguientes tareas de la semana nueva.
- **Reuniones en casa:** Durante cada sprint, realizamos al menos dos reuniones adicionales entre semana de aproximadamente 1h - 1h 15min. En estas reuniones seguimos trabajando en el proyecto, mejoramos funcionalidades y compartimos nuestro avance para coordinarnos mejor como equipo.

### Sprint 1: Planificación del proyecto

**Fecha:** 15-21 de enero **Objetivo:** Definir el alcance, estructura del equipo, roles, metodología, herramientas y organización del trabajo.

#### Tareas:

- Definir roles en el equipo(asignar y establecer responsables y roles del equipo).
- Seleccionar herramienta de gestión de tareas (Trello).
- Definir backlog de tareas(desglosar las funcionalidades requeridas en tareas específicas y estimar cada tarea).
- Crear repositorio en GitHub(configurar estructura inicial y añadir reglas de contribución).
- Configurar el tablero Kanban(con sus columnas y tarjetas correspondientes).
- Planificación del primer sprint de desarrollo (Sprint 2).

### Sprint 2: Desarrollo del menú principal y funcionalidades básicas

**Fecha:** 22-28 de enero **Objetivo:** Crear la estructura básica del programa y las funciones iniciales del inventario.

#### Tareas:

- Implementación del menú principal.
- Funcionalidad para mostrar todos los productos.
- Funcionalidad para añadir un producto.
- Funcionalidad para eliminar un producto por nombre.
- Revisión de código y pruebas iniciales.

### **Sprint 3: Funcionalidades avanzadas del inventario**

**Fecha:** 29 de enero - 4 de febrero **Objetivo:** Completar las funciones avanzadas y garantizar el correcto manejo de datos.

#### Tareas:

- Implementación de eliminación de un producto por posición.
- Funcionalidad para modificar el nombre de un producto.
- Sustitución de producto en una posición específica.
- Función para mostrar un resumen del inventario.
- Revisión y pruebas.

### **Sprint 4: Persistencia de datos, optimización y documentación (5-9 de febrero)**

**Fecha:** 5-9 de febrero. **Objetivo:** Añadir persistencia de datos, optimizar el programa y preparar la entrega final.

#### Tareas:

- Implementación de persistencia de datos.
- Optimización y mejora del código.
- Pruebas finales.
- Redacción de la documentación.
- Revisión final y entrega.