Documentación Técnica del Proyecto
Sistema Web de Selección de Personajes Fantásticos

Juan David Mayorga López – 20232020116
Samuel Andrés Barrera Pulido – 20232020156
Mariam Betin Escobar – 20232020300

Para la clase de Modelos de Programación
Profesor: Alejandro Daza
September 22, 2025

Contents

1 Resumen del Proyecto

El proyecto consiste en un sistema web de selección de personajes fantásticos. Su objetivo es demostrar la aplicación del patrón de diseño **Abstract Factory** para la creación de personajes con atributos específicos, armas, armaduras y monturas. Los usuarios pueden elegir entre diferentes razas (Humanos, Elfos, Enanos y Orcos), y el sistema genera los objetos correspondientes, mostrando sus características y estadísticas a través de una interfaz web HTML/CSS/JavaScript.

Adicionalmente, el sistema incorpora la clase `Pool`, que implementa el patrón **Singleton**. Este componente centraliza la gestión de las fábricas, garantizando que solo exista una instancia activa de la fábrica correspondiente a la raza seleccionada. De esta manera, se mejora la consistencia y eficiencia en la creación de los personajes, ya que el `Servlet` delega en `Pool` la responsabilidad de seleccionar y mantener la instancia adecuada de cada fábrica.

2 Arquitectura General

El proyecto sigue un enfoque web cliente-servidor, con un control adicional mediante la clase `Pool`:

- **Cliente (Frontend):** Interfaz web implementada con HTML, CSS y JavaScript. Solicita los datos del personaje mediante peticiones HTTP y despliega la información visualmente.
- **Servidor (Backend):** Implementado en Java con Servlets. El `EscogerPersonajes` Servlet recibe la raza seleccionada y consulta a la clase `Pool` para obtener la instancia correspondiente de la fábrica.
- **Clase Pool:** Implementa el patrón Singleton. Centraliza la lógica de selección de la fábrica (`FabricaHumanos`, `FabricaElfos`, `FabricaEnanos`, `FabricaOrcos`) y asegura que solo exista una única instancia activa.
- **Modelo:** Clases Java que representan los personajes y sus componentes (Cuerpos, Armas, Armaduras y Monturas) organizadas en paquetes.

3 Patrones de Diseño Utilizados

Se utilizan dos patrones principales:

- **Abstract Factory:** Permite crear familias de objetos relacionados sin especificar sus clases concretas.
- **Singleton (Pool):** Garantiza que solo exista una instancia de la clase encargada de gestionar las fábricas, sirviendo como punto de acceso global al sistema de creación de personajes.

4 Flujo de Ejecución Web

1. El usuario abre la interfaz web (HTML/CSS/JS) y selecciona la raza del personaje.
2. JavaScript envía una petición HTTP GET al Servlet `EscogerPersonajes` con el parámetro de la raza seleccionada.
3. El Servlet delega en la clase `Pool`, que determina qué fábrica concreta se debe usar según la raza.

4. `Pool` garantiza que exista una única instancia de esa fábrica y devuelve la referencia al Servlet.
5. El Servlet utiliza la fábrica para generar los objetos de cuerpo, arma, armadura y montura.
6. El Servlet devuelve un objeto JSON con la información completa del personaje y sus componentes.
7. JavaScript recibe el JSON y actualiza dinámicamente la interfaz, mostrando imágenes, nombres y estadísticas de cada componente.
8. El usuario puede repetir el proceso para seleccionar otras razas y comparar personajes.

5 Conclusión

La combinación de **Abstract Factory** y **Singleton (Pool)** permite estructurar el sistema de forma flexible y eficiente. El patrón Abstract Factory asegura la creación coherente de familias de objetos relacionados, mientras que el Singleton centraliza el control, evitando duplicaciones y garantizando la consistencia en el manejo de fábricas. El enfoque web permite que los usuarios interactúen mediante una interfaz gráfica, recibiendo los datos en JSON y desplegándolos dinámicamente con HTML, CSS y JavaScript.