

# ED06\_3 - Refactorización y documentación de proyecto Java

---

## 1. Introducción

En esta actividad se te proporciona un proyecto de Java que sirve para gestionar las reservas de un hotel. El objetivo es **refactorizar el código** corrigiendo problemas de diseño y documentarlo adecuadamente utilizando JavaDoc.

## 2. Objetivos

- Identificar code smells en el código Java proporcionado.
- Aplicar técnicas de refactorización para mejorar la calidad del código proporcionado.
- Documentar el código utilizando JavaDoc
- Generar la documentación en formato HTML utilizando JavaDoc.

## 3. Descripción del Proyecto

El proyecto proporcionado es un sistema de gestión de reservas de un hotel. El sistema permite la gestión de **habitaciones, clientes, facturación y reservas**. Puedes encontrar el proyecto en este [repositorio de Github](#).

## 4. Descripción de la Actividad

Debes analizar el proyecto en busca de code smells y aplicar técnicas de refactorización para mejorar la calidad del código. A continuación se detallan los pasos a seguir:

1. **Hazte un fork del repositorio:** Realiza un fork del repositorio proporcionado en tu cuenta de GitHub. Esto te permitirá trabajar en una copia del proyecto sin afectar al original.
  - Puedes encontrar información sobre cómo hacer un fork en la [documentación de GitHub](#).
2. **Clonar el repositorio:** Clona el repositorio del que acabas de hacer fork en tu máquina local.
3. **Abrir el proyecto en IntelliJ IDEA:** Abre el proyecto clonado en IntelliJ IDEA.
  - Puedes clonar directamente el proyecto desde IntelliJ IDEA
4. **Identificar code smells:** Utiliza las herramientas de análisis de código de IntelliJ IDEA para identificar code smells en el proyecto. Presta atención a los siguientes aspectos:
  - Clases demasiado grandes o con demasiadas responsabilidades.
  - Métodos demasiado largos o complejos.
  - **if** anidados o complejos. Posibilidad de aplicar **Early Return**.
  - Métodos demasiado largos.
  - Nombres de variables y métodos poco descriptivos.
  - Código duplicado.

- Código inalcanzable
  - Comentarios poco adecuados o redundantes.
  - Uso de *Magic Numbers*
  - Cualquier otros aspecto de los descritos en los apuntes de la unidad 6.
  - Puedes revisar los code smells más comunes en Refactoring Guru: [Code Smells](#).
5. **Refactorizar el código:** Aplica las técnicas de refactorización adecuadas para corregir los problemas identificados. Utiliza las facilidades proporcionadas por el IDE para realizar la refactorización de manera segura. **Realiza capturas de pantalla de los procesos de refactorización aplicados.**
6. **Documentar el código:** Utiliza JavaDoc para documentar el código refactorizado. Asegúrate de incluir descripciones adecuadas para clases, métodos y atributos.
7. **Generar la documentación** en formato HTML utilizando JavaDoc.
8. **Subir el proyecto a GitHub:** Sube el proyecto refactorizado y documentado a un nuevo repositorio de GitHub. Asegúrate de que el código esté bien organizado.

## 5. Evaluación

A continuación se detallan los criterios de evaluación:

- Identificación de code smells (3 puntos).
- Aplicación de técnicas de refactorización (3 puntos).
- Documentación del código utilizando JavaDoc (1.5 puntos).
- Generación de la documentación en formato HTML (1 punto).
- Subir los cambios a un nuevo repositorio de GitHub (0.5 puntos).
- Generación del informe (1 punto).

## 6. Entrega

Deberás entregar:

- Un informe en format PDF con el nombre **apellidos\_nombre\_ed06\_3.pdf** que contenga:
  - Descripción de los cambios realizados en el código.
  - Capturas de pantalla del proceso de refactorización con IntelliJ IDEA.
  - Capturas de pantalla del proceso de generación de la documentación con JavaDoc.
- Documentación generada a partir de comentarios Javadoc en formato HTML.
- Enlace a GitHub del proyecto con el código refactorizado y documentado.

El informe en formato pdf junto a la documentación generada en formato HTML y el enlace al repositorio de GitHub deben comprimirse en un archivo zip con el nombre **apellidos\_nombre\_ed06\_3.zip**