

INFORME GENERAL DEL PROYECTO — “PC Builder”

 **Nombre del proyecto:** ConfiguradorPC

– Configurador de PC por Componentes

 **Autor:**

[ABDELGHAFFAR EL AKHDAR]

 **Módulos implicados:**

- Entornos de desarrollo
 - Lenguajes de marcas
 - Bases de datos
 - Programación
-

 **Fecha de entrega:**

[26-05-2025]

1. Descripción del proyecto

He desarrollado una aplicación de escritorio en Java llamada **Configurador PC**, cuyo objetivo es permitir a los usuarios configurar un PC por componentes (CPU, GPU, RAM, Placa base, etc.) asegurando la **compatibilidad** entre ellos. La aplicación cuenta con funcionalidades tanto para el **usuario normal** como para el **administrador** que gestiona los componentes.

2. Objetivos del proyecto

- Aplicar los conocimientos adquiridos en los diferentes módulos.
- Desarrollar una aplicación con interfaz gráfica (Java Swing).
- Implementar el modelo MVC para una mejor organización del código.
- Asegurar la conexión con una base de datos MySQL.
- Permitir gestión de usuarios y componentes.
- Crear un carrito de compra con verificación de compatibilidad.
- Generar una factura en formato XML.

- Crear una pagina Web con la documentación a aportar.
 - Documentar el proyecto con buenas prácticas de Git y GitHub.
-

3. Tecnologías utilizadas

- **Java** (IDE: Eclipse)
 - **Java Swing** (interfaz gráfica)
 - **MySQL** (base de datos)
 - **HTML/CSS/JS** (web)
 - **Bootstrap** (opcional en parte web)
 - **XML** (generación de factura)
 - **Git y GitHub** (control de versiones)
 - **Trello** (gestión de tareas)
 - **Draw.io / UMLet** (diagramas)
-

4. Estructura del proyecto

He organizado todo en carpetas bajo `ProyectoFinal/` , donde cada carpeta tiene un propósito claro:

`ProyectoFinal/`

- |— 0-InformeGeneral.pdf
- |— 1-Cronograma-Kanban/
- |— 2-Especificacion-Requisitos/
- |— 3-Diagramas/
- |— 4-CodigoFuente/
- |— 5-DocumentacionTecnica/
- |— 6-Git/
- |— 7-XML-Factura/
- |— 8-Web/
- |— 9-ODOO/
- |— LEEME.txt

5. Capturas y recursos incluidos

- Capturas del cronograma y tablero Kanban.
 - Requisitos detallados del proyecto.
 - Diagramas UML actualizados.
 - Código comentado y con Javadoc generado.
 - Capturas del uso de Git (commits, push, log...).
 - XML generado tras una compra simulada.
 - Página web estática.
 - Documento de ODOO con pantallazos del alta de empresa y módulos.
-

6. Resultados obtenidos

- La aplicación funciona correctamente, con validación de compatibilidad y generación de factura.
 - Pude aplicar el patrón MVC y separar correctamente las capas del proyecto.
 - El uso de Git me permitió tener un control claro sobre los avances.
 - Documenté cada parte con comentarios, Javadoc y estructuras claras.
-

7. Dificultades encontradas

- Al principio me costó comprender cómo organizar bien el proyecto en carpetas y paquetes.
 - Integrar el DAO y la conexión con la base de datos fue complicado, sobre todo al probar errores.
 - Implementar la validación de compatibilidad entre componentes me llevó bastante lógica condicional.
 - Tuve que aprender a manejar XML en Java desde cero.
 - También tuve que entender bien Git, pero una vez dominado, me resultó útil.
-

8. Mejoras futuras

- Añadir filtros en la interfaz (por precio, por marca, etc.).
 - Mejorar el diseño visual de la interfaz con librerías externas o JavaFX.
 - Añadir persistencia del carrito.
-

9. Conclusión personal

Estoy satisfecho con el resultado. He aprendido a aplicar lo que estudié de forma práctica, a organizar un proyecto profesionalmente y a trabajar con herramientas reales como GitHub y MySQL. También he desarrollado habilidades de planificación, diseño, depuración y documentación. Este proyecto me ha ayudado a consolidar lo aprendido y me ha motivado a seguir mejorando como desarrollador.