# **Documentación Inicial del TFG**

## **1. Introducción**

Este proyecto de Trabajo de Fin de Grado (TFG) tiene como objetivo el desarrollo de una API REST utilizando Spring Boot, la cual estará conectada a un frontend para ofrecer una aplicación web funcional. La temática específica del proyecto está por definirse, pero se centrará en la creación de un servicio escalable, seguro y bien documentado.

## **2. Objetivos**

* Diseñar y desarrollar una API REST con Spring Boot.
* Implementar una base de datos adecuada para la aplicación.
* Integrar el backend con un frontend mediante peticiones HTTP.
* Garantizar buenas prácticas en desarrollo de software, incluyendo seguridad, documentación y pruebas.
* Evaluar el rendimiento y la escalabilidad de la API.

## **3. Tecnologías y Herramientas**

* **Backend:** Spring Boot, Spring Data JPA, Spring Security, JWT (si aplica), Swagger (para documentación).
* **Base de Datos:** MySQL / PostgreSQL / MongoDB (según necesidades del proyecto).
* **Frontend:** React / Angular / Vue.js (a definir según requerimientos).
* **Otros:** Docker, Postman, Git/GitHub para control de versiones.

## **4. Arquitectura**

El proyecto seguirá una arquitectura basada en:

* **Modelo Cliente-Servidor:** El frontend consumirá la API REST para la gestión de datos.
* **Patrón MVC:** Para la organización del código en el backend.
* **Autenticación y Autorización:** Implementación de seguridad con JWT o sesiones.

## **5. Desarrollo y Metodología**

* **Metodología de desarrollo:** Se utilizará una metodología ágil (Scrum o Kanban).
* **Planificación:** División en fases (análisis, diseño, implementación, pruebas y despliegue).
* **Herramientas de gestión:** Trello, Jira o GitHub Projects.

## **6. Evaluación y Pruebas**

* Pruebas unitarias con JUnit y Mockito.
* Pruebas de integración para verificar la comunicación entre componentes.
* Pruebas funcionales del frontend y API con Postman o Cypress.

## **7. Despliegue**

* Se estudiará el despliegue en un servidor en la nube (AWS, Heroku, DigitalOcean, etc.).
* Uso de Docker para la contenerización de la aplicación.

## **8. Conclusiones y Futuras Mejoras**

Este apartado se completará al finalizar el proyecto, incluyendo reflexiones sobre el desarrollo, resultados obtenidos y posibles mejoras futuras.