

Selección de Herramientas y Configuración del Entorno de Desarrollo

Para el desarrollo del MVP he decidido utilizar IntelliJ J como IDE de desarrollo y Git combinado con GitHub como control de versiones.

En cuanto al IntelliJ J como IDE, este es ampliamente reconocido por su robustez y características avanzadas que optimizan la productividad del desarrollo. Entre sus ventajas se encuentran:

- Un soporte avanzado para Java, ofrece funcionalidades como la finalización inteligente de código, refactorización automática y detección de errores. Mejorando y agilizando el código.
- Integración con Git y GitHub, ya que este permite la gestión de versiones del código sin salir del IDE y facilita la sincronización del proyecto con un repositorio remoto. Además de que posee una interfaz grafica facilitando el control del proyecto.
- Depuración, su depurador permite identificar errores de manera rápida y eficiente. Además, cuenta con herramientas para ejecutar pruebas unitarias.

Se usará Git como sistema de control de versiones y GitHub como plataforma de almacenamiento remoto. Esto nos permitirá, tener un historial de versiones para monitorizar cualquier cambio en el código, evitar la perdida de datos gracias a su almacenaje en un repositorio remoto además permitiendo restaurar versiones anteriores. GitHub además integra herramientas de CI/CD que facilita la automatización de pruebas de código y despliegue, asegurando que las versiones publicadas sean mas estables y funcionales. Por último al trabajar con estas herramientas se facilita el trabajo en equipos ya que proporciona herramientas para la gestión de ramas y fusiones, permitiendo así un proyecto con diferentes desarrolladores sin afectar la estabilidad del código base.

En conclusión estas herramientas optimizaran la productividad, permitirán un desarrollo estructurado y garantizaran la seguridad del código, facilitando el desarrollo del MVP.