**Arquitetura de Software:**

O sistema que estamos desenvolvendo será baseado no modelo **Cliente-Servidor** permitindo que os usuários acessem suas funcionalidades por meio de páginas web, sem a necessidade de instalação local. Esse modelo garante maior escalabilidade, facilidade de manutenção e flexibilidade na distribuição de atualizações.

Na camada de servidor da aplicação, adotaremos uma arquitetura baseada em **Monolito**, consolidando toda a lógica de negócios dentro de uma única aplicação. Essa abordagem simplifica o desenvolvimento inicial, reduz a complexidade da infraestrutura e facilita a manutenção, além de proporcionar um ambiente de execução mais previsível.

Apesar da escolha do modelo monolítico, manteremos boas práticas de desenvolvimento para garantir que o sistema seja modular e possa evoluir para uma arquitetura de microsserviços no futuro, caso seja necessário.

**Ferramentas de Desenvolvimento/Implementação:**

**Backend (Servidor)**

A camada de backend será responsável pelo processamento da lógica de negócios, comunicação com o banco de dados e exposição de APIs para o frontend. Para isso, utilizaremos:

* **Linguagem:** **Java** – Uma das linguagens mais populares e confiáveis para aplicações empresariais, proporcionando alta performance, segurança e uma vasta comunidade de suporte.
* **Framework:** **Spring** - O Spring nos ajuda a deixar o código mais organizado e facilita o desenvolvimento, principalmente com o Spring Boot, que agiliza configurações e integrações.
* **IDE**: **IntelliJ IDEA**

**Frontend (Interface do Usuário)**

Essa é a parte do sistema que os usuários interagem diretamente. Nosso objetivo é criar uma interface moderna, intuitiva e rápida. Para isso, vamos usar:

* **Linguagem:** **HTML, CSS e JS** – As bases do desenvolvimento web, garantindo compatibilidade com diferentes dispositivos e navegadores**.**
* **Framework:** **Angular** – Nos ajuda a criar aplicações web dinâmicas e organizadas, além de facilitar a manutenção do código.
* **IDE**: **Visual Studio Code** – Uma IDE robusta e cheia de recursos que melhora a produtividade no desenvolvimento em Java.

**Banco de dados**

O banco de dados será responsável por armazenar todas as informações do sistema, garantindo segurança e organização. Para essa parte, escolhemos:

* **Sistema de Dados:** **PostgreSQL** – Um dos bancos de dados mais poderosos e confiáveis disponíveis, perfeito para aplicações que exigem alto desempenho.
* **IDE:** **DBeaver** – Um gerenciador de banco de dados que facilita a administração, execução de consultas e organização dos dados.

**Versionamento de código**

Para manter o código organizado e facilitar o trabalho em equipe, vamos usar um sistema de controle de versões, garantindo que todas as alterações sejam registradas e que possamos voltar atrás caso necessário.

* **Ferramenta: Git** – Permite acompanhar todas as mudanças no código, facilitando o trabalho em equipe.
* **Repositório: GitHub** – Nossa central para armazenar o código, colaborar com outros desenvolvedores e integrar ferramentas de automação.