

Sprint 1 - Inicialização e Estrutura Base

Objetivo:

Configurar o ambiente de desenvolvimento, estruturar o projeto e preparar o versionamento.

Tarefas Principais e Subtarefas:

1.1 - Preparar o repositório Git e versionamento

- Criar repositório Git
- Definir arquivos .gitignore
- Configurar GitHub com GitHub Actions

1.2 - Estruturação base do Backend com ASP.NET Core

- Criar solução com ``dotnet new sln``
- Criar projetos: Domain, Application, Infrastructure, API
- Aplicar arquitetura Clean Architecture com DDD
- Instalar pacotes: Entity Framework Core, Swagger, JWT

1.3 - Estruturação base do Frontend com React

- Criar app com Vite: ``npm create vite@latest esellers-frontend --template react-ts``
- Instalar dependências essenciais (React Router, Axios)
- Configurar estrutura de pastas: ``pages``, ``components``, ``styles``, ``hooks``, ``services``
- Configurar CSS Modules com SCSS

1.4 - Configuração do Docker e Docker Compose

- Criar ``Dockerfile`` para backend e frontend
- Criar ``docker-compose.yml`` com PostgreSQL
- Testar subida completa: ``docker-compose up --build``

1.5 - Documentação inicial

- Criar README com estrutura inicial do projeto
- Documentar setup e instruções de uso dos containers

Pré-requisitos antes da Sprint 2

- Backend e Frontend devem rodar via Docker sem erros
- Estrutura de pastas deve estar definida e funcional
- Commit inicial realizado e documentado no GitHub
- Swagger funcional no backend
- Ambiente de desenvolvimento documentado no README

Referências Técnicas

- Clean Architecture com ASP.NET Core: <https://bit.ly/clean-architecture-dotnet>
- Swagger com .NET: <https://learn.microsoft.com/aspnet/core/tutorials/getting-started-with-swashbuckle>
- React + Vite com TypeScript: <https://vitejs.dev/guide/>
- Docker Compose + PostgreSQL: <https://docs.docker.com/compose/>
- GitHub Actions para CI/CD: <https://docs.github.com/en/actions>