



Bazar

Buscar...

Documentação Técnica: Plataforma de Artesanato Digital

1. Visão Geral

O projeto consiste em um Marketplace Multi-tenant (multi-lojas) desenvolvido para permitir que diversos artesãos gerenciem suas próprias vitrines digitais dentro de uma única federação/associação. A plataforma cobre desde a gestão de estoque até o fechamento de pedidos e análise de vendas.

2. Arquitetura do Sistema

Back-end (Java/Spring Boot)

Localizado no repositório independencia-backend, o servidor segue o padrão RESTful com as seguintes responsabilidades:

- **Segurança:** Implementação de JWT (JSON Web Token) para autenticação e autorização granulada.
- **Negócio:** Gerenciamento de entidades complexas como Loja, Produto, Pedido e Endereço.
- **Persistência:** Integração com banco de dados (provavelmente via Spring Data JPA) para manter a integridade dos dados dos artesãos.

Front-end (Angular)

Localizado no repositório associacao-moradores-java, foca na experiência do usuário (UX) e do lojista:

- **Módulos de Cliente:** Catálogo de produtos, carrinho de compras e fluxo de checkout.
- **Painel Administrativo:** Dashboard de vendas e gestão de produtos para os artesãos.
- **Integração:** Consumo de APIs via HttpClient com interceptors para anexar o token JWT.

3. Principais Funcionalidades

Módulo	Descrição
Auth & Security	Login e cadastro com proteção de rotas via JWT.
Vitrine Digital	Página personalizada para cada artesão exibir sua lista de produtos.
Gestão de Catálogo	CRUD completo de produtos (preço, descrição, estoque).
Carrinho & Checkout	Fluxo de compra e integração de endereços para entrega.
Dashboard	Visualização de métricas de vendas e histórico de pedidos.

4. Validação de Horas (O Elo Perdido)

Você mencionou que não sabia como validar as horas de trabalho. Em projetos de software, isso pode ser abordado de duas formas:

A. Validação do Esforço de Desenvolvimento

Se o objetivo é comprovar o tempo gasto para criar o projeto (para fins acadêmicos ou portfólio), utilize métricas de Git Analytics:

- **Commits por data:** Analise a linha do tempo entre o primeiro e o último commit.
- **Complexidade Ciclomática:** O volume de controladores e rotas Angular que você criou (como visto nos seus repositórios) sugere um esforço de nível Pleno, dada a quantidade de entidades integradas.

B. Implementação de "Time Tracking" no Sistema

Se você deseja que o artesão registre quanto tempo levou para produzir uma peça, a solução técnica seria:

1. **Backend:** Adicionar um campo `horas_producao` (Double) na entidade `Produto`.
2. **Lógica:** No checkout, o sistema poderia somar o "Esforço Total" do pedido para informar ao artesão o cronograma de produção.

5. Stack Tecnológica

- Linguagem Principal: Java 17+ / TypeScript
- Frameworks: Spring Boot, Angular
- Segurança: Spring Security + JWT
- Cloud/Deploy: Azure (App Service / SQL Database)
- Versionamento: Git (GitHub)

Nota do Desenvolvedor: O fato de você ter implementado um sistema de checkout com JWT e gestão de lojas indica que você domina o fluxo completo (full-stack) de uma aplicação empresarial. A estrutura de pastas nos repositórios mostra uma boa separação de responsabilidades (Clean Architecture/SoC).

Como você pretende apresentar esse projeto? Se for para um processo seletivo, posso te ajudar a refinar a seção do Dashboard de Vendas, que costuma ser o "brilho nos olhos" dos recrutadores.