

# Documentação da Arquitetura do Projeto

## Visão Geral

Este documento fornece uma visão macro da arquitetura do projeto, destacando os principais sistemas e aplicativos, e como eles se integram através de diferentes tecnologias. A arquitetura é dividida em várias partes interconectadas, cada uma desempenhando um papel crítico na solução de análise e otimização de preços para o setor varejista.

## Fluxo de Dados e Integração entre Componentes

### 1. Entrada de Dados

- **Gestores de Negócios (Clientes da Plusoft):** Os gestores do setor comercial dos clientes da Plusoft são os principais usuários do sistema. Eles inserem dados relacionados aos produtos, vendas, custos e outras variáveis relevantes através da interface mobile (Price Whisper) conectada ao backend em Java/Spring Framework e ao backend adicional em .NET.
- **Interação com Backend Firebase:** A interface mobile desenvolvida em React Native, se comunicará com um backend feito no Firebase para a sincronização em tempo real de dados.

### 2. Processamento

- **Backend Java/Spring Framework:**
  - **Operations:** Gerencia operações de criação, leitura, atualização e exclusão para entidades como Carrinho, Categoria, Histórico de Vendas, Marca, Modelo e Produto através da API própria que será gerada à partir desse backend.
  - **Business Logic:** Implementa a lógica de negócios para processar transações, aplicar regras de precificação e manter a integridade dos dados.
  - **Comunicação com Banco de Dados Oracle:** Utiliza JPA/Hibernate para realizar operações de CRUD no banco de dados Oracle, garantindo persistência e recuperação eficiente dos dados.

- **Backend Adicional em .NET:**
  - **Gerenciamento de Entidades:** Gerencia operações CRUD em entidades como Empresa, Usuários e Backlog.
  - **Operações de Integração:** Realiza operações de integração de dados para garantir que as alterações feitas pelos gestores sejam refletidas no banco de dados Oracle e, quando necessário, comunica-se com o backend Java/Spring através da API própria que foi criada no backend principal.

### 3. Banco de Dados Oracle

- **Estrutura de Dados:**
  - **Entidades e Tabelas:** O banco de dados Oracle armazena todas as entidades do sistema, incluindo Carrinho, Categoria, Histórico de Vendas, Marca, Modelo, Produto, Empresa, Usuários e Backlog.
  - **Relacionamentos:** Implementa relacionamentos entre tabelas para garantir a integridade referencial e permitir consultas complexas. Por exemplo, cada histórico de venda está relacionado a um produto e a uma categoria.

### 4. Análise de Dados

- **Sistema de Machine Learning (Python):**
  - **Extração de Dados:** Periodicamente, o sistema de machine learning extrai dados atualizados da empresa, incluindo históricos de vendas e informações de produtos.
  - **Processamento e Análise:** Utiliza bibliotecas de machine learning como scikit-learn e TensorFlow para analisar os dados e identificar padrões, tendências de mercado e comportamento do consumidor.
  - **Geração de Recomendações:** Com base na análise, o sistema gera recomendações de preços dinâmicos, categorizando produtos em Preço Ideal, Preço Baixo e Preço Elevado.
  - **Integração com App Mobile (React Native):** As recomendações geradas pelo sistema de machine learning são integradas ao aplicativo mobile, permitindo que os gestores acessem as análises e sugestões de preços diretamente pelo app.

## 5. Distribuição e Uso dos Dados Processados

- **Backend Java/Spring Framework:**
  - **Aplicação de Preços:** Atualiza os preços e outras informações dos produtos no banco de dados Oracle e na API própria conforme as recomendações, ajustando dinamicamente conforme necessário.
- **Dashboard de Insights:**
  - **Visualização:** Os gestores têm acesso a um painel de controle intuitivo que exibe os resultados das análises, incluindo recomendações de ajuste de preços, tendências de mercado e produtos com potencial de aumento de margem.
  - **Decisões Estratégicas:** Utilizam os insights fornecidos pelo dashboard para tomar decisões informadas sobre estratégias de precificação e promoção de produtos.
- **Notificações em Tempo Real:**
  - **Alertas:** O sistema envia alertas e notificações em tempo real para gestores, informando sobre oportunidades de ajuste de preços e mudanças significativas no comportamento do mercado.
  - **Firebase:** Utiliza a capacidade de mensagens em tempo real do Firebase para garantir que todas as partes interessadas recebam informações críticas instantaneamente.

## 6. Monitoramento e Manutenção

- **DevOps e Cloud Computing:**
  - **Virtualização:** A infraestrutura de contêineres (Docker/Kubernetes) permite a execução isolada de diferentes serviços, garantindo escalabilidade e eficiência.
  - **Monitoramento:** Ferramentas de monitoramento e logging rastreiam a performance dos sistemas, ajudando a identificar e resolver problemas rapidamente, mantendo a solução robusta e disponível.

## Conclusão

Esta documentação detalhada do fluxo de dados e integração entre os componentes ilustra como as diferentes partes do sistema trabalham em conjunto para fornecer uma solução coesa e eficiente de análise e otimização

de preços para o setor varejista. Cada componente desempenha um papel específico, e sua integração através de tecnologias modernas garante uma operação suave e eficiente, permitindo que os gestores comerciais das empresas clientes da Plusoft tomem decisões informadas sobre precificação. A análise e recomendações geradas pela IA, acessíveis pelo aplicativo mobile, permitem uma visão estratégica e ajustada em tempo real dos preços, maximizando as vendas e a lucratividade.