

Projeto: Sistema de Recomendação Hierárquica de Produtos (SRHP)

Mensagem do Cliente "Olá, equipe!

Nós somos uma empresa que está estruturando um novo módulo de recomendação de produtos dentro do nosso marketplace. Atualmente, temos milhares de itens cadastrados, mas nossa estrutura de categorias está desorganizada e o tempo de busca dos produtos está ficando alto à medida que o número de registros cresce.

O objetivo desse projeto é construir uma estrutura de dados eficiente para armazenar as categorias e subcategorias de produtos, permitindo buscas rápidas e sugestões automáticas de produtos relacionados.

Queremos que essa estrutura seja capaz de se reorganizar automaticamente conforme novas categorias são adicionadas — e para isso, acreditamos que árvores AVL são a melhor escolha técnica.

Também precisamos que a aplicação seja implementada com boas práticas de programação, e que a equipe saiba explicar a complexidade das operações (Big-O) e onde a recursividade foi aplicada.

Em resumo, o sistema precisa ser escalável, rápido e fácil de manter. A entrega deve vir acompanhada de um relatório técnico detalhado, explicando o raciocínio da equipe, as decisões de arquitetura e o impacto da estrutura AVL no desempenho do sistema." — Carlos Menezes, Diretor de Tecnologia e Inovação

Objetivo Geral Criar um Sistema de Recomendação Hierárquica de Produtos (SRHP) capaz de organizar e sugerir produtos de um marketplace de forma hierárquica e eficiente, usando árvores e árvores AVL como base para o armazenamento e recursividade nas operações de navegação, busca e sugestão.

Contexto de Negócio Nos grandes e-commerces, a busca eficiente de produtos é um diferencial competitivo. Estruturas de dados simples tornam-se ineficientes quando o número de produtos passa de milhares. Por isso, adotaremos árvores AVL, que garantem tempo de busca, inserção e remoção $O(\log n)$, mantendo o sistema performático mesmo sob carga elevada.

Estrutura Organizacional e Funções da Equipe - Product Owner (P.O.): Define prioridades e valida funcionalidades. - Project Manager (P.M.): Planeja entregas e gerencia prazos. - Tech Lead: Define arquitetura e padrões. - Desenvolvedores: Implementam módulos e testes. - QA / Tester: Garante a qualidade e cria casos de teste. - DevOps: Cuida de versionamento e implantação.

Descrição Técnica e Funcional 1. Estrutura de Dados: Baseada em árvore AVL com inserção, remoção e busca balanceadas, usando recursividade. 2. Módulo de Negócio: Permite cadastro e recomendação de produtos. 3. Módulo de Análise: Mede tempo de execução e compara complexidades. 4. Arquitetura: Python 3.10+, modular, interface CLI ou REST.

Requisitos Funcionais - Inserção, remoção e busca de categorias e produtos. - Impressão hierárquica da árvore. - Recomendação recursiva de produtos. - Exibição de tempos e complexidade.

Requisitos Não Funcionais - Código modular e documentado. - Uso obrigatório de árvore AVL. - Testes unitários com 80% de cobertura. - Relatório técnico completo.

Critérios de Avaliação - Estrutura de Dados: 2,0 - Recursividade: 1,5 - Complexidade: 1,5 - Arquitetura: 1,0 - Funcionalidades: 2,0 - Relatório Técnico: 2,0

Modelo de Relatório Técnico 1. Introdução: Problema e solução proposta. 2. Arquitetura: Classes, módulos e diagramas. 3. Recursividade: Onde e por que foi aplicada. 4. Análise de Complexidade: Comparação $O(n)$ vs $O(\log n)$. 5. Resultados: Testes, logs e desempenho. 6. Conclusão: Reflexão sobre ganhos e eficiência.

Extensões Possíveis - Visualização da árvore (React ou Tkinter). - API REST e persistência em SQLite. - Logs e monitoramento de performance.

Entregáveis - Código-fonte documentado. - Relatório técnico. - Protótipo opcional.