

Projeto 1:

TechMind – Organização Inteligente de Conhecimento Técnico

Setor empresarial

Educação / Tecnologia / Produtividade — plataformas de aprendizagem, comunidades técnicas e profissionais que consomem grandes volumes de conteúdo (documentação, cursos, artigos, vídeos e anotações) e precisam organizar, pesquisar e reutilizar conhecimento de forma eficiente.

Descrição do projeto

Criar uma solução que permita a organização inteligente de conteúdo técnico, facilitando sua classificação, consulta e reutilização.

A solução deve receber textos técnicos (por exemplo: descrições de artigos, documentação, anotações de estudo, conteúdos de cursos, tutoriais ou materiais de referência) e utilizar técnicas de Ciência de Dados para identificar informações relevantes sobre esse conteúdo.

As equipes poderão explorar diferentes abordagens, como:

- Classificação temática do conteúdo;
- Identificação de palavras-chave;
- Agrupamento por assuntos semelhantes;
- Recomendação de conteúdos relacionados;
- Organização automática de bases de conhecimento.

O resultado deverá ser disponibilizado em formato JSON para consumo por outras aplicações.

Esse tipo de solução pode ser utilizado por plataformas educacionais, comunidades técnicas, empresas ou profissionais que desejam construir repositórios inteligentes de conhecimento.

Opcionalmente, a solução pode ser complementada por uma interface simples para cadastro, busca ou consulta dos conteúdos processados.

A solução deverá ser integrada aos serviços OCI para armazenamento de modelos, documentos, implantação de APIs e/ou persistência de dados.

Necessidade do cliente (explicação não técnica)

Profissionais e estudantes de tecnologia consomem diariamente uma grande quantidade de conteúdo técnico, tornando difícil organizar, localizar e reutilizar essas informações posteriormente.

A solução deve permitir:

- Organizar conteúdos automaticamente;
- Facilitar buscas por temas ou tópicos;
- Encontrar conteúdos relacionados;
- Reduzir o esforço manual de catalogação;
- Construir uma base de conhecimento reutilizável.

Essa solução permite transformar grandes volumes de informação em conhecimento estruturado e facilmente acessível.

Validação de mercado

Ferramentas de gestão do conhecimento são amplamente utilizadas por empresas, equipes técnicas e plataformas educacionais.

Soluções que automatizam a organização de conteúdos podem:

- Reduzir o tempo gasto na busca por informações;
- Melhorar a produtividade;
- Facilitar o compartilhamento de conhecimento;
- Apoiar processos de aprendizagem contínua;
- Escalar repositórios de conhecimento de forma eficiente.

Mesmo soluções simples podem gerar valor ao automatizar tarefas repetitivas e melhorar o acesso à informação.

Expectativa para este Hackathon

Objetivo

Entregar um MVP funcional que permita organizar e enriquecer conteúdos técnicos utilizando técnicas de Ciência de Dados e integração com uma API.

Escopo recomendado

As equipes poderão trabalhar com uma ou mais abordagens, tais como:

- Classificação de conteúdo;
- Agrupamento de documentos;
- Extração de palavras-chave;
- Busca semântica;
- Recomendação de conteúdos relacionados.

A integração entre a aplicação e o modelo de Ciência de Dados deverá ser definida pela equipe, considerando a arquitetura adotada para a solução.

- Recomenda-se que a integração seja realizada de forma que permita o consumo do modelo por meio da API desenvolvida para o projeto.

Resultados esperados

Notebook da equipe de Ciência de Dados (Jupyter/Colab)

Contendo:

- Exploração e limpeza de dados (EDA);
- Tratamento de textos;
- Transformação de dados em formato adequado para modelagem;
- Treinamento e avaliação de modelos;
- Métricas de desempenho apropriadas para a solução proposta;
- Serialização do modelo (joblib/pickle).

Aplicação Back-End (API REST)

- API com endpoints relacionados à solução desenvolvida;
- Recebimento de conteúdo para processamento;
- Retorno dos resultados gerados pelo modelo;
- Validação de entrada;
- Tratamento de erros.

Integração com OCI

Sugestões de uso:

- Object Storage para armazenamento de modelos ou documentos;
- OCI Compute para hospedagem da aplicação;
- OCI Functions para processamento específico;
- Banco de dados opcional para persistência.

Documentação mínima (README)

- Como executar o projeto;
- Como utilizar a API;
- Exemplos de requisição e resposta;
- Dependências e versões utilizadas.

Demonstração funcional

Apresentar a solução em funcionamento e explicar como os modelos utilizados geram os resultados.

Funcionalidades exigidas (MVP)

O serviço deve expor pelo menos um endpoint capaz de receber conteúdo técnico e retornar informações processadas pela solução desenvolvida.

Exemplo

POST /conteudo

Entrada:

```
{
  "titulo": "Introdução ao Spring Boot",
  "texto": "Neste conteúdo são apresentados os conceitos básicos
para criação de APIs REST utilizando Java e Spring Boot."
}
```

Saída:

```
{
  "categoria": "Backend",
  "probabilidade": 0.89,
  "informacoes_adicionais": [
    "Java",
    "Spring Boot",
    "API REST"
  ]
}
```

A estrutura final da resposta poderá variar de acordo com a abordagem escolhida pela equipe.

Requisitos

- Modelo treinado e carregado;
- Processamento funcional do conteúdo;
- API operacional;
- Integração com OCI;
- Mínimo de três exemplos de uso;
- Documentação funcional.

Recursos opcionais

- Recomendação de conteúdos relacionados;
- Busca por palavras-chave;

- Processamento em lote (CSV);
- Dashboard simples;
- Persistência de resultados;
- Containerização com Docker;
- Testes automatizados;
- Explicabilidade do modelo;
- Busca semântica;
- Consulta por categorias.

Diretrizes técnicas para estudantes

Ciência de Dados

Cada equipe deverá construir sua própria base de conteúdos.

Os dados poderão ser:

- Obtidos de fontes públicas;
- Extraídos de documentações;
- Produzidos pela própria equipe.

Sugestões de tecnologias:

- Python;
- Pandas;
- Scikit-Learn;
- TF-IDF;
- Regressão Logística;
- Técnicas de similaridade textual.

A utilização de outras abordagens e modelos é permitida.

Back-End

Desenvolver uma API REST utilizando as tecnologias trabalhadas durante a formação.

A API deverá:

- Receber os dados de entrada da solução;
- Consumir o modelo desenvolvido pela equipe de Ciência de Dados;
- Retornar respostas em formato JSON;
- Realizar validações e tratamento de erros;
- Disponibilizar endpoints necessários para o funcionamento da aplicação.
- A arquitetura utilizada ficará a critério da equipe e deverá ser documentada.

OCI

A solução deve utilizar pelo menos um serviço OCI como parte **obrigatória** do projeto.

