

UNIVERSIDADE ESTÁCIO DE SÁ DO CEARÁ

DISCIPLINA: DESENVOLVIMENTO RÁPIDO DE APLICAÇÕES EM PYTHON

PROFESSOR(A): CYNTHIA MAIA

ALUNOS:

MARCIO ERIK MARTINS DE OLIVEIRA (202203636219)

DAVID DO AMARAL DELFINO (202302766943)

JOÃO VICTOR DIÓGENES DE SOUSA (202303606941)

RELATÓRIO DE DESENVOLVIMENTO RAD

SISTEMA DE GERENCIAMENTO DE EXPEDIÇÃO ESPACIAL

1. Introdução

Objetivo do Projeto:

- Desenvolver uma API RESTful capaz de gerenciar missões espaciais, contendo atributos como id, nome da missão, data de lançamento, tripulantes, carga útil, status, informações adicionais da missão.

Contexto e Justificativa:

- Projeto importante para o gerenciamento e organização das expedições lançadas, para fins de armazenamento dos dados.

2. Metodologia

Fases do Desenvolvimento:

1. Planejamento de Requisitos:

- a. Criar, ler, deletar, atualizar informações da expedição.

Obs: filtrar leitura dos dados baseado no id, intervalo de tempo e ser capaz de retornar todos.

2. Design do Usuário:

- a. Desenvolvimento baseado apenas em API, feita validação por meio do Thunder Client.

3. Construção:

- a. Implementação do back-end usando Flask.

4. Teste:

- a. Testes unitários e de integração.
- b. Correção de bugs e ajuste de funcionalidades com base no feedback.

5. Implementação:

- a. Implantação do projeto em ambiente de produção.
- b. Monitoramento e manutenção contínuos.

3. Desenvolvimento Técnico

Tecnologias Utilizadas:

- **Back-end:** Flask, SQLAlchemy
- **Banco de Dados:** SQLite
- **Ferramentas de Desenvolvimento:** Visual Studio Code, Git

4. Resultados Obtidos

Funcionalidades Implementadas:

- Criação, leitura, atualização e exclusão de missões (CRUD).
- Filtro de missões por intervalo de datas.

Desafios Enfrentados:

- **Definir e Atualizar o Banco de Dados:** Garantir que o banco de dados estivesse atualizado com todas as colunas necessárias foi complicado. Desde adicionar manualmente colunas, até verificar se o contexto da aplicação estava corretamente configurado para realizar migrações.
- **Lidar com Tipos de Dados e Serialização:** Houve dificuldades com a serialização de tipos de dados como `Decimal`, que precisavam ser convertidos para um formato compatível com JSON.

- **Implementação de Funcionalidades Específicas:** Criar endpoints que lidam com operações complexas, como filtrar missões por intervalo de datas e aceitar parâmetros no corpo de requisições GET, foi um desafio técnico e exigiu bastante adaptação.

Feedback dos Usuários:

- Como usuário da API de gerenciamento de missões, minha experiência foi direta e eficaz. Através de endpoints claros, pude criar, atualizar e excluir missões sem complicações. A filtragem por intervalo de datas foi especialmente útil para monitorar missões específicas. No geral, a API oferece uma maneira eficiente de gerenciar missões com precisão e facilidade.

5. Conclusão

Resumo das Conquistas:

- A principal conquista desse projeto foi o aprendizado. Por meio dos desafios enfrentados no decorrer do projeto, tivemos muito conhecimento adquirido.