

Projeto da Disciplina - Sistema de Gerenciamento de Expedição Espacial

Implementarão um sistema para administrar o cadastro de missões de expedições espaciais.

1. Descrição geral

O propósito do projeto é construir um Sistema de Gerenciamento de Expedição Espacial para facilitar o planejamento, monitoramento e execução de missões espaciais, desde o lançamento até o retorno.

a. **Create (Criar):** - adicionar novas missões ao sistema, incluindo campos como nome da missão, data de lançamento, destino, tripulação, carga útil;

b. **Read (Ler):** Página de visualização que lista todas as missões registradas no sistema, exibindo detalhes básicos de cada missão.

Pontos:

Visualizar todas as missões: exibir uma lista de todas as missões registradas no sistema, permitindo aos usuários visualizar detalhes básicos de cada missão, como nome, destino e estado em **ordem decrescente de data de lançamento**;

Recuperar os detalhes de uma missão específica com base no **ID da missão**;

Pesquisar missões por intervalo de datas: permitir aos usuários pesquisar missões com base em um **intervalo de datas específico**, como todas as missões lançadas entre uma data inicial e uma data final.

c. **Update (Atualizar):** Formulário de edição para modificar os detalhes de uma missão existente, permitindo atualizações em campos como destino, tripulação, carga útil, etc.

d. **Delete(deletar):** Opção para excluir uma missão do sistema, removendo todos os dados associados. Excluir pelo ID da Missão.

Tabela Missão

Atributos da Tabela de Missões:

1. **ID da Missão:** Um identificador único para cada missão espacial. Tipo de Dados: Inteiro (Chave Primária).
2. **Nome da Missão:** O nome atribuído à missão espacial. Tipo de Dados: Texto (String).
3. **Data de Lançamento:** A data em que a missão foi lançada. Tipo de Dados: Data.
4. **Destino:** O destino final da missão, como um planeta, lua ou órbita específica. Tipo de Dados: Texto (String).
5. **Estado da Missão:** O estado atual da missão, como "Ativa", "Concluída", "Abortada", etc. Tipo de Dados: Texto (String).

6. **Tripulação:** Uma lista dos membros da tripulação envolvidos na missão, o nome do membro: Texto (String).
7. **Carga Útil:** Uma descrição da carga útil transportada pela missão, incluindo equipamentos científicos, instrumentos de medição, etc. Tipo de Dados: Texto (String).
8. **Duração da Missão:** A duração planejada da missão, desde o lançamento até o retorno. Tipo de Dados: Intervalo de Tempo (Datetime or str).
9. **Custo da Missão:** O custo total estimado da missão, incluindo custos de desenvolvimento, lançamento, operação e retorno. Tipo de Dados: Moeda (Decimal).
10. **Status da Missão:** Informações detalhadas sobre o status atual da missão. Tipo de Dados: Texto Longo (String ou Texto).

2. Lembre-se dos seguintes pontos ao entregar o resultado:

- Entregue um único arquivo zip. O nome do arquivo zip deve ser "missoes_nome_da_equipe.zip" e deve conter os códigos e o relatório em formato .pdf, identificando o(s) nome(s) do(s) aluno(s) que fez(fizeram) o projeto.

O projeto será avaliado de acordo com os seguintes critérios:

- **Total: 5 pontos.**

- **Qualidade do design (0.5 ponto).**

- **Qualidade do código (4 pontos):** consistência na codificação (padrão de codificação), os nomes de variáveis consistentes, mau tratamento de exceções, etc.

- **Relatório RAD (0.5 ponto):** explicação de cada fase RAD.

Como o projeto será corrigido:

Design: Avalia-se a qualidade do design examinando o código. Verificam-se as entidades principais e seus métodos, os tipos de dados. Atribui-se uma nota de 0 a 0.5 para esse quesito.

Código: Essa é a parte mais difícil da correção. Defeitos a observar no código: a consistência na codificação (padrão de codificação), mau tratamento de exceções. Atribui-se uma nota de 0 a 1 para esse quesito. Funcionamento: O **Read (Ler)** tem um peso maior! **Atribui-se uma nota de 0 a 1.5 para esse quesito. Os demais é retirado 0.5 ponto para cada Create (Criar); Update (Atualizar); Delete(deletar) que não funcionar.**

Relatório: se foram descritas as fases RAD utilizadas. Atribui-se uma nota de 0 a 0.5 para esse quesito.

OBS: em Flask, vou validar a API por meio do Thunder Client.

Equipes:

Tkinter: Até 3 pessoas.

Flask: Até 3 pessoas.

Implementação em Flask apenas API. Mas, se desejarem podem utilizar front-end.(0.5 ponto extra)