



FACULDADE DE INFORMÁTICA E ADMINISTRAÇÃO PAULISTA

Tecnologia Análise e Desenvolvimento de Sistemas

Front-end design Engineering

Prof. Vinícius Marcili

CHECKPOINT 06

INSTRUÇÕES PARA A ATIVIDADE

1. Esta atividade é em **trio**.
2. Não serão aceitos trabalhos exatamente iguais. Lembrem-se somos pessoas diferentes, pensando de modo diferente para a solução do mesmo problema.
3. O *layout* responsivo é obrigatório, acarretando -1 pontos na falta.
4. Não utilize imagens para criar o layout.
5. Evite o uso de CSS interno ou *inline*;
6. Envie a pasta compactada pelo sistema da FIAP sem os arquivos listados no gitignore
7. No rodapé da página, coloque o nome completo e RA do trio
8. Somente um integrante do trio precisa efetuar a entrega

Boa atividade!

CASE: Minhas cartas favoritas

Você é responsável por criar um sistema onde um usuário pode favoritar cartas de Pokémon TCG, exibindo informações uteis sobre as cartas. Apenas usuários logados devem ter acesso ao sistema. **A stack escolhida foi Next.js.**

O Sistema deve fazer a integração com a API do pokémon TCG.

URL	https://api.pokemontcg.io/v2/cards
Metodo	GET
Exemplo	https://api.pokemontcg.io/v2/cards?q=name:charizard&orderBy=number,name onde name:charizard estamos fazendo a busca por qualquer carta que tenha o nome charizard, e orderBy estamos ordenando pelo número e nome do pokémon

Critérios de avaliação:

A avaliação consistirá dos seguintes pontos:

1. Organização do projeto (1,0 pontos)

- A estrutura das pastas e dos arquivos será levada em consideração (0,3 ponto)
- Os arquivos do .gitignore não devem ser entregues (0,3 pontos)
- Typagens – Lembre de criar os tipos e usar eles de forma correta (0,4 ponto)
- O sistema DEVE ser feito usando a estrutura de app, no caso do uso da estrutura de pages penalidade de 2 pontos

2. Login (2,0 pontos)

- Criar uma página de login com campos de login e senha (0,5 pontos)
- Criar uma rota de autenticação no API Handler do Next, validar apenas a senha fixa em uma variavel de ambiente Sensível (0,5 pontos)
- Tratamento de erros e redirecionamento para área logada devem ser levados em consideração, o usuário deve permanecer logado ao atualizar a página (1,0 pontos)

3. Busca de cartas (3,0 pontos)

- Após o usuário logar, o sistema deve exibir uma tela com um campo de busca por texto e uma lista pré carregada de cartas (pegar da API) deve exibir nome, imagem, número, tipos e hp. (1,0 ponto)
- A tela deve possuir paginação (parametros **page e pageSize** na queryString) (1,0 ponto)
- Quando o usuário digitar algo, a busca deve ser feita na API e retornar os resultados desde a primeira página (1,0 ponto)

4. Favoritando cartas (3,0 pontos)

- O sistema deve possuir uma funcionalidade de favoritos, possuindo um botão que ao usuário clicar ele possa salvar esse favorito (1,0 ponto)
- Os favoritos devem ficar no localStorage, ao atualizar a página devo continuar com os mesmos favoritos (1,0 ponto)
- Ao deslogar o usuário, os favoritos devem ser limpos. (1,0 pontos)

5. Deploy na Vercel (1,0 ponto)

- Subir o projeto no GIT e fazer deploy da Vercel, colocar o link no README.md (Deixar o link como PUBLICO)
- Lembrar de configurar as variáveis de ambiente para o projeto funcionar

Extra (2,0 pontos)

- Fazer uma página ao clicar em uma carta sem favoritar e exibir informações adicionais sobre a carta (1,0 ponto)
- Favoritar e remover o favorito por essa página de informações adicionais (1,0 ponto)

Atenção:

- ✓ Evitar o uso de CSS interno ou inline.
- ✓ Responsividade obrigatória
- ✓ **No rodapé da página inicial, coloque o nome completo e RA do trio (Penalidade de 1 ponto se não estiver no HTML da página)**

Exemplo

Pokémon TCG Explorer

Descubra cartas e detalhes de Pokémon

Lista de Pokémon

Busque um Pokémon...

Buscar



#460 - Abomasnow

HP: 140

Grass



#359 - Absol

HP: 70

Darkness



#359 - Absol G

HP: 70

Darkness



#142 - Aerodactyl



#142 - Aerodactyl



#142 - Aerodactyl