

## Trabalho 2 de JavaScript

### Nomes:

Este trabalho deve ser feito em **duplas**. Eventualmente poderão ser aceitos trios. O trabalho deve **ser entregue até 28/11/2021**. Informar os nomes das duplas por e-mail ao professor até 21/11/2021. Ao enviar o trabalho somente 1 integrante precisa enviar. O trabalho consistirá de vários arquivos e os mesmos devem ser zipados e enviados pelo AVA . Ao enviar o trabalho informar o nome de todos integrantes do trabalho.

Os alunos devem desenvolver um **Aplicativo PWA** que utilize o **localStorage** e/ou **chamadas a APIs externas**. O aplicativo PWA deve:

- Utilizar em algum momento os comandos **fetch, async e await**
- Ter um **manifesto** funcionando e com todos os tipos de ícones necessários
- Um **service worker** funcionando e que em algum momento armazene e acesse no mínimo 5 itens no **cache**.
- coloque os códigos em uma página **HTML padrão com bootstrap ou materialize** e mostre os elementos conforme o visual destes frameworks. As páginas HTML devem ter uma boa responsividade.
- as páginas devem passar pelo **validador da w3c** sem erros (<https://validator.w3.org/>).
- as páginas não devem apresentar erros pegos pelo **console do JavaScript(JS)** dos navegadores.
- Todas alterações na página feitas pelo JS devem gerar uma nova página que não apresenta erros captados pelo validador da w3c ou pelo console. (As páginas serão corrigidas por um do debug JavaScript completo)
- Na **interface auxilie o usuário a não cometer erros**, evite que ele cometa erros e caso erros aconteçam eles devem ser tratados. Teste as diversas situações que podem ocorrer com o sistema

Sugestões de aplicações:

- App de “TO DO” que armazena dados no localStorage: ou seja, um gerenciador de tarefas por fazer (permitir acessar, incluir e excluir itens no bd do navegador). Este tipo de aplicação tem muitas variantes que podem ser implementadas (lista de filmes, lista de compras,... Pode-se separar os itens em categorias e outras variações)

- Acessar APIs. Criar uma interface de acesso, buscar informações na API e mostrar os resultados. Por exemplo a API <https://api.lyrics.ovh/v1/artist/title>

Você troca a palavra artist pelo nome de um artista e title pelo título de uma música ...faz um fetch para este endereço... você obterá de volta uma string json com a letra da música solicitada ... depois é só mostrar a letra em uma interface apropriada.

- Pode-se combinar localStorage com API. Você acessa uma API, obtém informações e armazena no localStorage algumas. Por exemplo você pode armazenar localmente suas letras de música preferidas.

- Em <https://swapi.dev/documentation> tem uma documentação para uma api que conforme o endereço ela retorna a correspondente informação sobre o universo Star Wars (...é só procurar que tem api de tudo por aí....coloque no google “API game of thrones” e veja o que acha...). Aqui vai uma lista de APIs interessantes <https://devporai.com.br/9-apis-para-seu-novo-projeto/>

-----

Deverá ser possível instalar o PWA enviado em qualquer dispositivo com navegadores Web (celulares, computadores, ...). Se seguir as recomendações acima é provável que isto aconteça, mas

teste para garantir os resultados. Um bom bom visual também contará (pode ser clean ou mais produzido...mas deve ficar bonito, sem erros...).

**Opcional** (se bem sucedido obterá a simpatia do professor no momento do arredondamento de nota :D ) Disponibilize o seu projeto no Github. Ele é um servidor <https://> e uma vez disponibilizado em um endereço visível em toda internet... o seu aplicativo pode ser instalado por qualquer pessoa com acesso a internet (sem depender de playstore). E se tudo for feito de forma correta o APP ficará bem instalado em diversos dispositivos.