

## Desafio iOS

Olá, gostaríamos de conhecer melhor suas habilidades em desenvolvimento iOS. Para isso, estamos enviando um desafio: O desafio consiste em desenvolver um aplicativo para consumir dados de uma API REST, exibi-las e em seguida salvá-las.

O desafio deve ser feito em no máximo **2 dias**, a contar da data de envio. Caso não consiga finalizar o desafio, o teste pode ser enviado com o que foi feito até o momento.

O projeto deve estar em um provedor GIT e o repositório deve estar aberto com a opção "pública", para que possa ser acessado e revisado.

O título do e-mail deve ser: **Desafio iOS: SEU\_NOME\_COMPLETO**.

## Vamos ao Desafio

Crie um aplicativo que leia os repositórios públicos com linguagem Swift do GitHub. Utilize a API: <https://api.github.com>.

A primeira tela deve conter um **SEGMENTED CONTROL** com dois itens.

- Quando o **ITEM 1** estiver selecionado, deve ser exibido a lista de repositórios (sem salvá-los): <http://api.github.com/search/repositories?q=language:Swift&sort=stars>
- Quando o **ITEM 2** estiver selecionado, deve ser exibida uma lista de repositórios que forem salvos no Core Data (abaixo contém a explicação de como salvar);

Ao clicar em um repositório, deve-se abrir uma segunda tela com:

- Os detalhes do repositório;
- A lista de Pulls: <http://api.github.com/repos/{owner}/{repository}/pulls>;
- O botão "SALVAR", ao clicar em "SALVAR" o repositório deve ser persistido (salvo) no SQLite e o botão deve ter seu título alterado para "REMOVER";
- Ao clicar em "REMOVER" o repositório deve ser removido do SQLite e o título do botão volta a ser "SALVAR".

## Requisitos técnicos

- A linguagem utilizada deve ser Swift;
- Toda a interface deve ser criada programaticamente;
- Aplicar conceitos como View Code e MVVM.
- A aplicação deve mostrar as informações através de uma **TableView** ou uma **CollectionView** (na primeira e na segunda tela).

## Serão considerados diferenciais

- Documentação no código;
- Paginação das listas: carregar lista de repositórios por demanda na rolagem da tela;
- Testes (TDD) ;
- Cancelamento da requisição ao sair da tela onde ela foi chamada.

## Informações Importantes

- É obrigatório o uso de Git (usar qualquer provedor e que esteja acessível para revisão);
- Todos os commits devem ser escritos em inglês;
- Não será avaliado o design da aplicação;
- O projeto não precisa estar finalizado para ser enviado.

## Fique à vontade para

- Utilizar bibliotecas de terceiros que considerar importante (explique o motivo de ter utilizado);
- Utilizar sua estrutura de organização do projeto;
- Seguir o layout de tela que achar interessante para o usuário.

**Como enviar?**

- Ao finalizar o desafio, você deve enviar por e-mail o link do projeto no repositório GIT;
- O título do email deve ser: **Desafio iOS: SEU\_NOME\_COMPLETO**.

Boa sorte!