

Challenge Job-Readiness

O objetivo desta etapa é desenvolver um app utilizando as APIs do Mercado Libre, que permita ao usuário ver os detalhes de um produto. Para isso, o Mercado Libre possui APIs abertas à comunidade para que qualquer desenvolvedor possa consumi-las e ter pesquisas e compras no seu aplicativo.

Layout do app

O App a ser desenvolvido deve ter duas telas:

1. Campo de pesquisa e Exibição dos resultados da pesquisa.
2. Detalhe de um produto.

Você pode entregar uma lista e detalhes de produtos que são somente texto, ou um mecanismo de pesquisa com imagens, ícones e texto, e um detalhe completo do produto, como o mostrado na web.

Etapas

Para utilizar os endpoints fornecidos, devemos realizar uma série de etapas para registrar o nosso aplicativo e obter uma chave de acesso:

1. Para isso, você deverá criar uma conta como desenvolvedor dentro da plataforma Mercado Libre e criar um novo aplicativo ali mesmo. Ao preencher o formulário, um endereço web como "Redirect URL" será solicitado. Nesse campo colocar a URL do Mercado Libre (<https://mercadolibre.com.ar>).
2. Uma vez criado o aplicativo, devemos acessá-lo através da opção "Editar" e dentro dele, teremos que copiar os valores "App ID" e "Client Secret".
3. Usando o navegador da nossa preferência deveremos acessar a seguinte URL (adicionando em \$APP_ID o código do nosso aplicativo):
[https://auth.mercadolibre.com.ar/authorization?response_type=code&client_id=\\$APP_ID&redirect_uri=https://mercadolibre.com.ar](https://auth.mercadolibre.com.ar/authorization?response_type=code&client_id=$APP_ID&redirect_uri=https://mercadolibre.com.ar)
4. Essa ação pedirá permissões para redirecionar para o site do Mercado Libre. Pressionaremos o botão "Permitir" e o navegador redirecionará para o site indicado

acima. Observe que a URL terá uma modificação e terá o seguinte formato: [https://www.mercadolibre.com.ar/?code=\\$CODE](https://www.mercadolibre.com.ar/?code=$CODE). Copiar esse valor.

5. Abrir o terminal do computador e acessar o seguinte CURL. Observe que devemos inserir o "App Id", "Client Secret" e "Code" gerenciados anteriormente:

```
curl -X POST \
-H 'accept: application/json' \
-H 'content-type: application/x-www-form-urlencoded' \
'https://api.mercadolibre.com/oauth/token' \
-d 'grant_type=authorization_code' \
-d 'client_id=APP_ID' \
-d 'client_secret=CLIENT_SECRET' \
-d 'code=CODE' \
-d 'redirect_uri=https://mercadolibre.com.ar'
```

O referido endpoint responderá com um JSON, do qual devemos pegar a variável "access_token" que usaremos para acessar os endpoints do Mercado Libre.

Importante: Esse token possui expiração, portanto, se a API retornar um código 401, devemos repetir as etapas para gerenciar um novo "access_token".

Descrição

O usuário utilizará o app para procurar um **listado dos produtos mais vendidos** de acordo com a categoria da pesquisa. Ao abrir o aplicativo deverá ser apresentado um campo de pesquisa, no qual pode ser inserida uma categoria. O valor inserido deve ser validado pelo preditor de categorias fornecido na documentação. Esse endpoint retornará uma lista de opções possíveis, e usaremos apenas a primeira delas para obter os produtos mais vendidos desta categoria.

Cada produto terá a opção de ser selecionado e abrirá uma nova tela com mais descrições do item escolhido. Nesta seção, além da descrição do produto, haverá também um botão para salvar sua ID como favorita.

Exigências técnicas

1. **Criar componentes visuais com estilos de design de acordo com o Mercado Libre.**
2. **Tratar casos de erro do ponto de vista do desenvolvedor.** Ou seja, como os casos de erro inesperados são gerenciados, consistência em todo o aplicativo, uso de logs, etc.
3. **Tratar casos de erro do ponto de vista do usuário.** Priorizar uma experiência fluida dando **feedback ao usuário** de forma adequada.
4. **Gerenciar fluxos de navegação do app de pesquisa.**
5. Controlar o fluxo de vida de:
 - a. **Activities e Fragments** (Android)
 - b. **Controllers** (iOS)
6. Interagir com o gerenciador de **dependências** Gradle (Android-) ou Cocoapods (IOS) de acordo com o caso.
7. Armazenar IDs dos produtos favoritos usando Shared Preferences (Android) ou Userdefaults (IOS) conforme apropriado.
8. Realizar o **pedido aos endpoints** de forma adequada.
9. Obter o JSON através do pedido de forma correta.
10. Mapear o JSON para modelos do aplicativo.
11. Criar um **serviço reutilizável** para o tratamento de solicitações.
12. Implementar **testes unitários** utilizando JUnit (Android) ou XCTest (IOS) dependendo do caso. Testar que as validações na barra de pesquisa sejam cumpridas.

¡Bônus track!

- Documentar o código.
- Destacar nas boas práticas de código.
- Uso de Clean Architecture e Padrões de Arquitetura Mobile

Critérios de entrega

- Entrega do projeto: Repositório de código (Fury, GitHub ou similar).
- Respeitar as guidelines oficiais da plataforma para garantir a legibilidade do código.
- O App deve ser user friendly para garantir uma ótima experiência de usuário.

**Documentação:**

- Registro do aplicativo: [Registre o seu aplicativo](#)
- Preditor de categorias: [Categorização dos produtos](#)
- Mais vendidos por categoria: [Mais vendidos no Mercado Libre](#)
- Pesquisa de itens: [developers.mercadolibre.com.ar/es_ar/items-y-busquedas](#)