

Android Coding Challenge

Implement a native Android app that displays photos from a REST service in a master-detail pattern.

Acceptance Criteria

- When launched, the app will display a list of album titles from the <https://jsonplaceholder.typicode.com> server using the album endpoint (See API data below).
- When any album item is tapped, the app will fetch the list of photos for that album from the server, and will display a list of the photos for that album.
- Tapping on any photo will display the details of that photo.

Technical Requirements

- Use any networking library of your choice to handle the network requests from the server. Implement a REST client to service the network calls.
- Implement the data model and the presentation layer using any design patterns of your choice. Be prepared to explain your choice of design.
- Data caching is not a requirement for this implementation.
- No prefetching is required: only fetch data for photos AFTER an album item is tapped.
- For the albums list, only display the album title.
- For the photo lists, display a list of thumbnail images (using the thumbnail url key), with each thumbnail image presented on the left of the associated metadata (only title, albumID, and ID) for the photo.
- For photo details, display the large image (using the url key) with the associated metadata (title, albumID, and ID) underneath the image.

Constraints

- Use a minimum SDK of 21, and a target SDK of 28
- Portrait mode only
- Native Android only: implement in either Java or Kotlin

Additional Information

- API data
 - There are a 100 albums available. Each album has 50 photos.
 - Sample album JSON - <https://jsonplaceholder.typicode.com/albums>
 - Sample photo JSON - <https://jsonplaceholder.typicode.com/photos>
 - URL endpoints
 - albums: **/albums**
 - photos: **/photos**

- Sample GET request URL for photos for albumId 3:
<https://jsonplaceholder.typicode.com/photos?albumId=3>
- Endeavor to adhere to best practices and ensure that the app is functional, and the code clean and efficient. Be prepared to explain any tradeoffs and design decisions made in the implementation.
- Please refer to the <https://jsonplaceholder.typicode.com> link for additional information.
- Upon completion, bundle the app as a **zip** prefixed with your name -- **<your name>_coding_challenge.zip** (for example: **JohnSmith_coding_challenge.zip**). **It should be possible to unzip the bundle, import and run it as an Android project.**

Please send the zip by email to our recruiting team:

Caitlin O'Shea - caitlin.oshea@t-3.com

Andrew Jimeno - andrew.jimeno@t-3.com