The next step is a short development project: This exercise is meant for our dev team to get acquainted with your skills and strengths. We would like you to build a simple app using Apple’s iTunes api.

Thing we are looking for:

1.    A UI with more than one feed types from Coming Soon, Hot Tracks, New Releases, Top Albums, and/or Top Songs and including at least 50 items each from the feed generator you can see using this web page: [https://rss.itunes.apple.com/en-us](https://url.emailprotection.link/?bs20JfWQHDE_MVcJ_xrDDrIgQSu2klopPmYhown0o4hcBywzWCQRX9nYj1MaklALlv8PkluPjkInQw1ZMOFzpTA~~).

2.    The various feed types shall be displayed in multiple pages, feel free to use TabLayout or BottomNavigation view or Navigation Drawer or some PagerAdapter of your choice (These are just examples). The UX is completely your choice but must include at minimum an image of "artworkUrl100" and "artistName".

3.    Tapping on a feed item should push the user onto a new view where we see a larger image at the top of the screen and the same information that was shown in the main view, plus genre, release date, and copyright info below the image. A button should also be included on this second view. When tapped, it will fast app switch to a web browser displaying the album page using the "url" found in the JSON file.

4.    The apis needed for you to build the app is available in [https://rss.itunes.apple.com/en-us](https://url.emailprotection.link/?bs20JfWQHDE_MVcJ_xrDDrIgQSu2klopPmYhown0o4hcBywzWCQRX9nYj1MaklALlv8PkluPjkInQw1ZMOFzpTA~~). An Example: The endpoint for the Feed type of "Top Albums" could be found under the Feed Type label in the above web page. ([https://rss.itunes.apple.com/api/v1/us/apple-music/top-albums/all/5/explicit.json](https://url.emailprotection.link/?bI2kBOqiaHTtHobewNZe1DWd5Av3yGLsEmNy_NhLuXAfssiMRgEhXfWjU8boY4t41Sk1kLk1ucaKVbWJw0QqWTD7005c2fALv6tIJKkalIx5DP3RHgIYBHr4yWSk6ZFn6Bt3g83MZncfUwY9PYyv9FixebEIPm1vMOvY_B0R0YT8~)).

5.    You are free to use any library for networking; we prefer the network execution to NOT BE DONE with an AsyncTask.

6.    Use an application architecture of your choice. We want to see your understanding of an Android app architecture!

7.    Minimum Android API is 24.

Your application's code must be entirely written in Kotlin. Feel free to make use of Android's framework, as well as AndroidX libraries. Details on how the application is to be installed and the overall app design should be included in the applications README.md file.  Additionally, you may add any other information you deem to be relevant for the dev team.

When completed, send a Github link for your final submission.

Nice-to-haves:

•    Use of Kotlin Coroutines is highly encouraged.

•    Pleasing to the eye transitions/animations

•    Use of Material Components

•    We do not mandate you to write any tests, but valid tests are a plus.

In short what we are looking for is your knowledge of the Android application systems and ability to navigate through existing or new architecture.  We expect the exercise to take a couple of hours to complete but this is not a time boxed exercise.

Looking forward to continuing forward!