Anywhere Mobile Engineer Candidate Code Exercise

Overview

In order to best assess your fit with Anywhere Real Estate, we'd like to get an idea of how you approach application development. There is no time limit for this exercise, and **the time you take to complete it does not affect our consideration.**

Your goal is not simply to finish it -- we consider it a given that you can meet the requirements; rather**, the goal is to complete the exercise with the highest possible quality so that we understand what you will bring to our team.** Regardless of the complexity of the requirements, you should try to demonstrate your knowledge of large-scale production quality app-building, including robust architecture and infrastructure and correct usage of best-practice design patterns.

Finally, adherence to the requirements demonstrates your attention to detail. *Read carefully.*

App Requirements

Your app should be written entirely in either **Swift** for iOS, or **Kotlin** for Android. Try to demonstrate your knowledge of both basic and advanced language semantics. We are not yet using SwiftUI or Compose on our mobile projects, so please complete the exercise using UIKit for iOS or XML layouts for Android.

Write a sample app that fetches and displays data from a RESTful Web API. The app should be comprised of two parts, a **list** and a **detail**. Your app should support both portrait and landscape orientations on both phones and tablets.

On Phones, the list and detail should be separate screens, on Tablets, list and detail should appear on the same screen.

·   For the list view, data should be displayed as a text only, vertically scrollable list of character names.

·   The app should offer **search** functionality that filters the character list according to characters whose titles or descriptions contain the query text

·   Clicking on an item should load the detail view of that character, including the character's image, title, and description. You choose the layout of the detail.

·   For the image in the detail view, use the URL in the "Icon" field of the API JSON response. For items with blank or missing image URLs, use a placeholder image of your choice.

·   Two variants of the app should be created, using a single shared codebase. Each variant should have a different name, package-name, and url that it pulls data from. (We're interested in your methodology for creating multiple apps from a shared codebase)

Variant One

Name: Simpsons Character Viewer

Data API: http://api.duckduckgo.com/?q=simpsons+characters&format=json

Package/Bundle name: com.sample.simpsonsviewer

Version Two

Name: The Wire Character Viewer

Data API: http://api.duckduckgo.com/?q=the+wire+characters&format=json

Package/Bundle name: com.sample.wireviewer

Miscellaneous

Using Libraries To Complete The Task

Use open-source libraries as you see fit, but we must be able to build and run your project in the IDE. Before sending, consider building and running your project from a clean environment.

Aside from the libraries you use, please disclose any other sources from which you have taken code or used as a bootstrap, other than the templates provided by the IDE. This includes sample apps from other projects or your own pre-written bootstrap app code.

Reaching Out For More Information

Please reach out with any questions you have, or if any part of the above is unclear or ambiguous.

Note

This code is purely an exercise to assess your software engineering skill level. We will never use any of your code in any of our products, and you retain full ownership of anything you submit.