

Coding Test for Native Mobile Developer

This coding challenge helps us understand how you think about building great software and problem-solving skills.

Task:

Develop a production-ready mobile app that allows users to search for songs, albums, or artists using the iTunes API. The app should also provide the ability to filter and bookmark songs. Ensure that the app supports English, Traditional Chinese (Hong Kong), and Simplified Chinese (China) languages.

Requirements:

1. The app should allow users to search for songs, albums or artists by typing in search criteria.
2. The app should display the search results in a list showing the song name, artist, and album art.
3. The app should support paging the search results by providing the ability to load more records. Each page limits 20 records.
4. The app should allow users to filter the search results by country or media type.
5. The app should provide the ability to bookmark songs by adding them to a "favorites" list, which can be accessed later.
6. The app should be available in English, Traditional Chinese (Hong Kong), and Simplified Chinese (China) languages.
7. The app should have a clean and intuitive design that is easy to use.
8. The app should support iOS 12+ / Android 6.0+.

Additional Considerations:

1. Use the latest platform-specific tools or framework.
2. Use MVVM app architecture
3. Add any additional features that showcase your skills, creativity, and the potential of the app.

Instructions:

1. Create a new project with git for your work.
2. Commit your code to your branch as you work on the coding test.
4. Complete the coding test, and push it to GitHub/GitLab when you are finished.
5. Provide instructions on how to set up and run the app.

6. Provide a brief write-up (no more than two pages) describing your approach to implementing paging, solutions to problem, and any additional features you implemented.
7. Use clear and concise code documentation, and unit tests where possible.

API Information:

The endpoint to use on iTunes Search is free and publicly accessible.

PAGING EXAMPLE

To get second-page search results for all Jack Johnson audio and video content (movies, podcasts, music, music videos, audiobooks, short films, and tv shows), use the following URL: <https://itunes.apple.com/search?term=jack+johnson&offset=20&limit=20>.

Full API documentation is available at <https://developer.apple.com/library/archive/documentation/AudioVideo/Conceptual/iTuneSearchAPI/index.html>

We expect to see production-ready code, with clear and readable code documentation and unit tests where possible. Remember, we are checking to see if you have a good grasp of using Git for version control. Good luck!
