Upcoming Movies Mobile App

(Mobile Dev Code Challenge)

Project description

As a mobile engineer you've been tasked with the development of an app for cinephiles and movie hobbyists. This first version (MVP) of the app will be very simple and limited to show the list of upcoming movies. The app will be fed with content from The Movie Database (TMDb). No design specs were given, so you're free to follow your UX and UI personal preferences. However the app should work both in landscape and portrait orientations. The choice of platform (iOS, Android, Xamarin) and development approach is also for you to decide based on previous experience and/or personal interest.

Functional Requirements

The first release of the app will be very limited in scope, but will serve as the foundation for future releases. It's expected that user will be able to:

- Scroll through the list of upcoming movies - including movie name, poster or backdrop image, genre and release date. List should not be limited to show only the first 20 movies as returned by the API.
- Select a specific movie to see details (name, poster image, genre, overview and release date).
- (Optional) Search for movies by entering a partial or full movie name.

Technical Requirements

You should see this project as an opportunity to create an app following modern development best practices (given your platform of choice), but also feel free to use your own app architecture preferences (coding standards, code organization, third-party libraries, etc).

A TMDb API key is already available so you don't need to request your own:
1f54bd990f1cdfb230adb312546d765d.
The API documentation and examples of use can be found here:

- https://developers.themoviedb.org/3

Feel free to use package/dependency managers (ex: Maven, CocoaPods, etc) if you see fit.

Deliverables

The project source code and dependencies should be made available in GitHub. Here are the steps you should follow:

1. Create a public repository on GitHub (create an account if you don't have one).
2. Create a "development" branch and commit the code to it. Do not push the code to the master branch.
3. Create a "screenshots" sub-folder and include at least two screenshots of the app.
4. Include a README file that describes:
    - Special build instructions, if any
    - List of third-party libraries used and short description of why/how they were used
5. Once the work is complete, create a pull request from "development" into "master" and send us the link. We will review your code and might ask you some questions directly on your pull request.

Notes

Here at ArcTouch we're big believers of collective code ownership, so remember that you're writing code that will be reviewed, tested, maintained by other developers. Things to keep in mind:

- First of all, it should compile and run without errors
- Be as clean and consistent as possible
- **Despite the project simplicity, don't ignore development and architecture best practices. It's expected that code architecture will be structured to support project growth.**

This project description is intentionally vague in some aspects, but if you need assistance feel free to ask for help.

We wish you good luck!