Careem Assignment

Instructions - Evaluation Criteria for this Assignment

1. **Readability:** Class and method names should clearly show their intent and responsibility.
2. **Maintainability**
    1. 'SOLID' Principles.
    2. UIViewControllers should not know too much explicit knowledge about non UI Classes. - The need for domain objects.
    3. We do not like long methods or classes. Let alone huge UIViewController classes.
3. **Scalability:**
    1. Your software should easily accommodate possible future requirement changes.
    2. If you are asked to change to xml-based api instead of json
    3. If you are asked to use a different persistent store (Core Data, SQLite, iCloud, FMDB)
    4. If you are asked to use different domain for listing or images, different url configurations for listing or image.
4. **Testability:** Tests are great, but testability is more important.
    1. Please Unit Test all non-UI classes. Mocking, Stubbing, TDD if possible.
    2. Please handle all types of error, which could occur.

Other Instructions

1. Use Github - Commit Often, Perfect Later, Publish Once.
2. The solution should be a mobile application.
3. You should solve it as if you're doing it for real.
4. Document your code.
5. Code should not contain any warnings, or errors.
6. You may use external libraries or tools for building or testing purposes. If you're using anything that's not written by you, mention it.
7. Optionally, you may also include a brief explanation of your design and assumptions along with your code.
8. We want our hiring process to be fair, and for everyone to start from the same place. To enable this, we request that you do not share or publish these problems.
9. We will assess a number of things including the design aspect of your solution and your object oriented programming skills. Whilst these are small problems, we expect you to submit what you believe is “production-quality” code that you would be able to run, maintain and evolve. You don’t need to “gold plate” your solution, but we are looking for something more than a bare-bones algorithm. You should submit code that you would be happy to produce in a real project, or that you would be happy to receive from a colleague.

Requirements:

1. As a user at the search screen,
    a. When I enter a name of a movie (e.g. "Batman", "Rocky") in the search box and tap on "search button"
    b. Then I should see a new list view with the following rows
        i. Movie Poster
        ii. Movie name
        iii. Release date
        iv. Full Movie Overview
2. As a user at the Movie List Screen,
    a. When I scroll to the bottom of list
    b. Then next page should load if available
3. As a user at the search screen,
    a. When I enter a name of a movie that doesn't exist in the search box and tap on "search button",
    b. Then, an alert box should appear and display an error message.
    c. All types of error should be handled
4. As a user at the search screen,
    a. When I tap and focus into the search box
    b. Then an auto suggest list view will display below the search box showing my last 10 successful queries (exclude suggestions that return errors)
    c. Suggestions should be persisted.
5. As a user at the search screen with the auto suggest list view shown,
    a. When I select a suggestion
    b. Then the search results of the suggestion will be shown.

APIs to use:

1. http://api.themoviedb.org/3/search/movie?api_key=2696829a81b1b5827d515ff121700838&query=batman&page=1
2. Poster (size: w92, w185, w500, w780): http://image.tmdb.org/t/p/w92/2DtPSyODKWXluIRV7PVru0SSzja.jpg