Mindvalley Coding Challenge

Your Task

Your task if you choose to accept it, is to recreate the Channels view that is on the Mindvalley app.

Requirements

- There are 3 sections. **New Episodes**, **Channels**, and **Categories**.
    - These 3 sections should be shown in one view as shown in the design, in that order.
- The channels section has two different design types.
    - One is for type Course and the other is for type Series. An example of a Series is “Mindvalley on Stage”. Example of a Course is “Impact at Work”
    - To differentiate between a Series and a Course, take a look at the Channels object in the JSON provided and follow the pseudocode below

      If the Series object inside Channels exists
      show the Series design
      else
      show the Course Design.

    - New Episodes and Channels should be horizontally scrollable with a maximum of 6 being shown per row
    - The Course and Series click events do not need to be handled
- All the data should be loaded from the provided API calls.
- Images and JSON should be cached efficiently, so they can be viewed offline
- API Requests/data loading and showing should be done in a way to give the user the best experience. Keep in mind the JSON provided might have some data missing in some instances, so these cases should be handled gracefully to provide a good UX

- It should be responsive to all phone sizes.
- Should have Unit Tests
- Use version control system (Git)

Bonus

- Supporting iPad/tablet sizes is an advantage
- Having some animations to make the user experience better is an advantage

Design

- You will need to Sign Up for Figma to see the design details. (Link)
- Follow the measurements/colors provided on Figma for the views.
- For the font, you can use Roboto with the provided font sizes.

API

- There is an API in JSON format for each section.
    - New Episodes Section (link)
    - Channels Section (link)
    - Categories Section (link)

Evaluation Criteria

Readability

- Class and method names should clearly show their intent and responsibility.

Design

- Should match the measurements and colors of the views in the design

Maintainability

- “SOLID” Principles and design patterns.
- Can use any design pattern the candidate is familiar with

Scalability

- Your software should easily accommodate possible future requirement changes

Testability

- Please accompany your code with test classes.

Expected Output

- The app should be able to run on any device running **iOS 13** and above. Please make sure the app can be compiled directly from the latest version of **Xcode** and the latest version of **Swift**

- You can share the code by sending it in a zipped file via email, or just sharing the link via Google Drive or Dropbox, or any other file sharing platform. **Github/Bitbucket/GitLab repos are not accepted.**

- We would be expecting a **README** file for the project with the following questions answered:
    - What parts of the test did you find challenging and why?
    - What feature would you like to add in the future to improve the project?