iOS Software Engineer - Lightweight iTunes Catalog

We recommend that you distribute the GitHub link to your iOS project so that we may run the project and view your code.

1. Build an API that takes in a search term and uses that value to call the iTunes Search API. The API that you build will take the iTunes Search API and sort each of the results into categories based on media type. The response of this API should be a JSON object, where each field are the different media types and inside each field is an array of objects.

    - Required fields: id, name, artwork, genre, url

    ```
    {

        ‘song’: [{
            id: Integer, // trackId (ID of entity)
            name: String, // name of entity
            artwork: String, // URL of the artwork
            genre: String, // Genre of entity
            url: String // trackViewUrl
        },
        ...
        ]
        ,
        ‘feature-movie’: [{
            id: Integer, // trackId (ID of entity)
            name: String, // name of entity
            artwork: String, // URL of the artwork
            genre: String, // Genre of entity
            url: String // trackViewUrl
    }, ... ]

    }
    ```

2. Build a view that allows a user to type in a search query and will display the results on the page (do this by calling the API you created above in #1). The results should be split into different sections based on the ‘kind’ of entity. (ex. Songs will all be together sectioned off, feature-movies will be in another section,

etc). If a certain “kind” section doesn’t have any entries, do not show that section.

- Required data to be shown on the view: picture of the artwork, name, genre, link to iTunes

List of ‘kinds’ from documentation (link below): [book, album, coached-audio, feature-movie, interactive- booklet, music-video, pdf podcast, podcast-episode, software-package, song, tv- episode, artist]

iTunes Search API Documentation:

https://affiliate.itunes.apple.com/resources/documentation/itunes-store-web-service-search-api/

Example API call to iTunes Search API:

https://itunes.apple.com/search?term=jack+johnson

3. Allow items to be marked as “favorites”. These favorites can be a mix of different “kinds” of entities and should always be accessible on the page (even when no search has been entered).