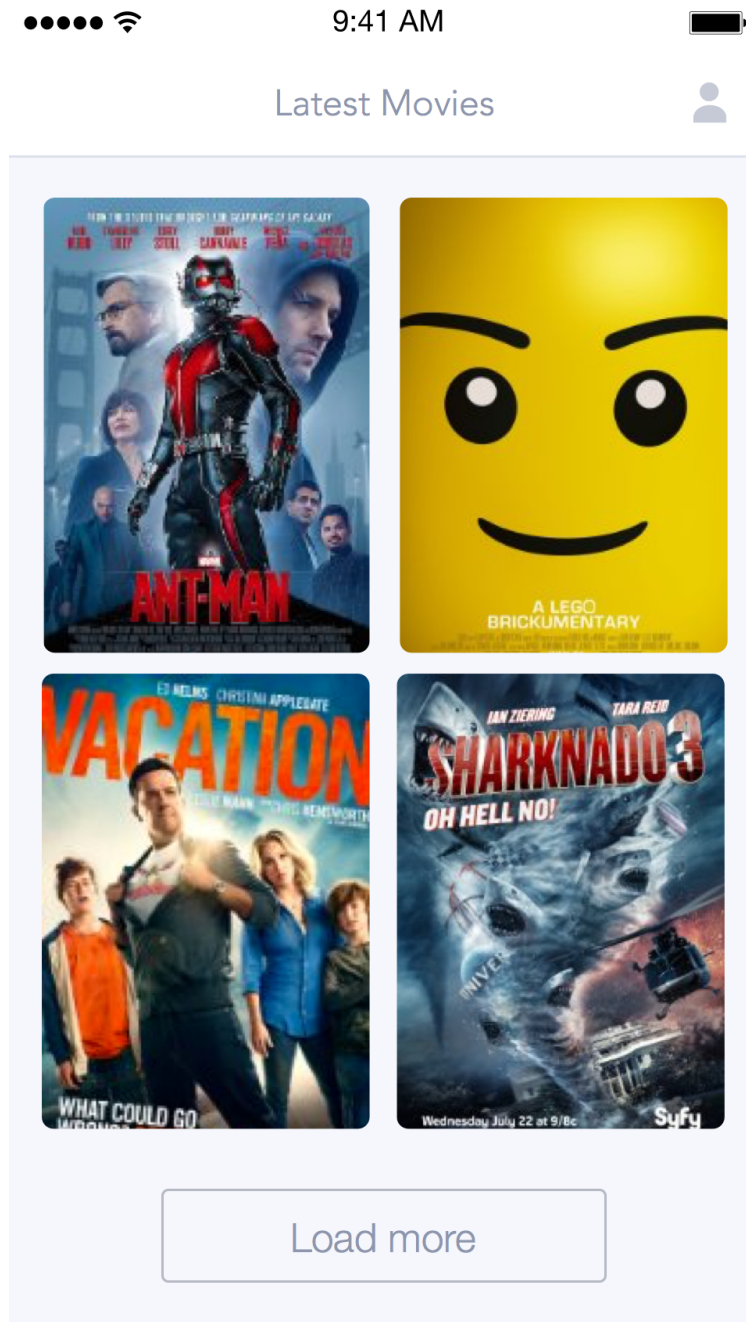


Movie Grid

Company is preparing to pivot from helping people to get fit to helping people find movies (just kidding). We want to build an app that displays movies in a grid of thumbnails.



The Movie Database API

You should use the [The Movie Database API](#) to fetch JSON movie data.

In particular, you should use the endpoint that returns [now playing movies](#).

You can use this API Key: ebea8cfca72fdff8d2624ad7bbf78e4c

Sample Request

`http://api.themoviedb.org/3/movie/now_playing?api_key=ebea8cfca72fdff8d2624ad7bbf78e4c`

Sample Response Snippet

```
{
  "dates": {
    "minimum": "2015-06-03",
    "maximum": "2015-07-15"
  },
  "page": 1,
  "results": [
    {
      "adult": false,
      "backdrop_path": "/dkMD5qlogeRMiEixC4YNPUvax2T.jpg",
      "genre_ids": [28, 12, 878, 53],
      "id": 135397,
      "original_language": "en",
      "original_title": "Jurassic World",
      "overview": "Twenty-two years after the events of Jurassic Park, Isla Nublar now features a fully functioning dinosaur theme park, Jurassic World, as originally envisioned by John Hammond.",
      "release_date": "2015-06-12",
      "poster_path": "/uXZYawqUsChGSj54wcuBtEdUJbh.jpg",
      "popularity": 54.349781,
      "title": "Jurassic World",
      "video": false,
      "vote_average": 7,
      "vote_count": 910},
    {
      "adult": false,
      "backdrop_path": "/o4I5sHdjzs29hBWzHtS2MKD3JsM.jpg",
      "genre_ids": [878, 28, 53, 12],
      "id": 87101,
      "original_language": "en",
      "original_title": "Terminator Genisys",
      "overview": "The year is 2029. John Connor, leader of the resistance continues the war against the machines. At the Los
```

```

    Angeles offensive, John's fears of the unknown future begin to
    emerge when TECOM spies reveal a new plot by SkyNet that will
    attack him from both fronts; past and future, and will ultimately
    change warfare forever.",
    "release_date": "2015-07-01",
    "poster_path": "/5JU9ytZJyR3zmClGmVm9q4Geqbd.jpg",
    "popularity": 50.258717,
    "title": "Terminator Genisys",
    "video": false,
    "vote_average": 6.4,
    "vote_count": 168
  },

```

Constructing Poster URLs

The movie api responses include a field called “poster_path” you will need to construct a full image url based on the path. The [documentation](#) explains how to do this for various image sizes. Here’s how you can build urls for 324 pixel wide poster images:

poster_url: `http://image.tmdb.org/t/p/w342 + <poster_path>`

Example Poster URL

poster_path: /5JU9ytZJyR3zmClGmVm9q4Geqbd.jpg

full_url: `http://image.tmdb.org/t/p/w342/5JU9ytZJyR3zmClGmVm9q4Geqbd.jpg`

Functionality

You should create a simple app that uses the data from the the Movie Database API and displays the movie poster thumbnails in a grid layout.

Please follow the designs above as closely as possible. A keen eye for design details is something we definitely look for.

iOS specific: Please do not use Storyboards to implement this challenge. We generally build our views and layouts programmatically so your solution should avoid using Storyboards. If you want to use nibs + code for view controllers that’s ok.

You should load an initial set of movie data for 20 movies, display those in a grid and when the user scrolls down you should progressively load another set of 20 movies, using the page query parameter:

[http://api.themoviedb.org/3/movie/now_playing?
api_key=ebea8cfca72fdff8d2624ad7bbf78e4c&page=2](http://api.themoviedb.org/3/movie/now_playing?api_key=ebea8cfca72fdff8d2624ad7bbf78e4c&page=2)

You should also implement a detail view that displays a detail view when a user taps on a poster in the grid with a back arrow to return to the grid view.



Support landscape and portrait orientation

Create a project in swift on github so we can see all the history of your commits.
Use any technology you prefer, try to show skills at making good app architecture.
Preferably to use swiftlint

Optional:

Support pull to refresh

Support of some kind of local storage so the app can be used without internet connection

Some kind of loading animation - up to you
