JavaScript Test

engin

You are required to build a simple web application using AngularJS 1.x that will show the Top 25 Articles from the Y Combinator Hacker News feed. All the resources you will require will be listed below.
The purpose of this test is to demonstrate your JavaScript abilities in unfamiliar territory and problem solving ability, and providing a well formed solution to a basic problem.
Majority of our microservices have an Angularjs front end. We are gradually moving onto React.

Requirements

There are very few requirements to this task, and it is expected that you take ownership of the majority of the implementation.
When you send through your solution, you will need to provide a link to your private repository, it could be GitHub, GitLab or any repository of your choice.
Your solution must

- Use AngularJS 1.x
- Make use of at least one Angular Factory/Service written by yourself
- Make use of the built in $q service **defer** or **Promise** pattern(s)

Things to Consider

Here are some points to note:

- All logic should be on the Front End (HTML/CSS/JS). No server side code is allowed.
- You can use Yeoman generators and/or any task runners/helper scripts to aid in the setup of your test, but the logic must be written by you.
- You may use external JS libraries to help you (Lodash/Underscore, etc), but the majority of logic must be written by you.
- Each item you show needs to include the title, link to the url, the author’s username and the score (see below)

Hacker News API Endpoints:

Below are the two API endpoints for Hacker News you will need for this project.

Top 500 Stories (There is no top 100 endpoint)

Request Type: GET
Endpoint: https://hacker-news.firebaseio.com/v0/topstories.json
Returns: An Array of story_ids for the Top 500 articles.

Item Details Request Type: GET
Endpoint: https://hacker-news.firebaseio.com/v0/item/{story_id}.json
Returns: An Object with details of the story matching the ID or *null* if there is no story with that
Example Object:

```
{
    "by":"anacleto",
    "descendants":139,
    "id":21126641,

"kids":[21126850,21129157,21127413,21128013,21127199,21130125,21131051,2
1127623,21129206,21130810,21128053,21127207,21131862,21129127,21127543,2
1128505,21127296,21128577,21128723,21129397,21128867,21127680,21130566,2
1128413,21127558,21128835,21127574,21131297,21127681,21127507],
    "score":470,
    "time":1569944569,
    "title":"The 2002 mandate for internal communication systems at
Amazon",
    "type":"story",
    "url":"https://www.sametab.com/blog/frameworks-for-remote-working"
}
```

Remember to append .json (dot json) at the end of your endpoints to ensure you get a JSON response.

AngularJS: Helpful hints

- The Docs for Angular live at https://docs.angularjs.org/api
- StackOverflow has a huge Angular community, and most of your questions will already be answered there
- Make use of **ng-repeat** to render your list.
- the **$q** and **$http** services work really well together.
- Angular Factories are global singletons, and are useful for sharing data across multiple Controllers.

It should take you a few hours to complete this task.

Have Fun and Good Luck!