Test Task

Introduction

Please create a simple application with the list of pokemons (pokedex). You can pick either the front-end or the full-stack requirements. Full stack variant is more complex (has more requirements) and will also demonstrate your backend skills.

Requirements

You have to meet the following requirements:

1. Source code should be open sourced in GitHub
2. The design of the app is not important, but the overall user experience and usability of the application should be good. You can use the existing UI libraries like material ui or ant.design
3. Application should be responsive (look good both on mobile and desktop)
4. Code should be linted. Preferably with the eslint standard config (https://github.com/feross/eslint-config-standard)
5. The application should be deployed online. Heroku, github pages are a few of free alternatives for deployment
6. The app must be developed with React.js and MobX (or a MobX alternative with the same reactivity concept, but not Redux) in ES6. For full stack developers the server has to be in Node.js

Evaluation Criterias

We do not expect to receive a production ready application from you, but it's important for us to understand whether you have the following skills:

1. Deliver a minimal viable product in a short period of time
2. Follow good coding conventions and style guides
3. Know what is important to implement, and what can be left out to be implemented in future
4. Ability to properly integrate libraries into the app and use them
5. ... more stuff depending on your skill level (DRY, architecture skills, etc.)

Bonuses

Following are not a requirement, but will be a bonus for us:

1. Stylus or any other css preprocessor
2. Webpack if applicable
3. CSS animations
4. Unit testing

Pokedex

Using the open pokemon API (http://pokeapi.co) build a pokedex with the following functionality:

Front End Candidate Requirements:

1. List pokemons in a table view (or cards) with their name, avatar, type (should visually look as a colored tag) and the main pokemon stats (whichever additional pokemon info you want to show)
2. The list must have a pagination with an ability to select how many items to show per page (10 / 20 / 50)
3. Filter the pokemons by name with a search box
4. Filter the pokemons by type using tags (multiple selected tags have to show pokemons with any of the types selected)
5. * Show your skills

Full Stack Candidate Requirements:

1. Everything from the list of the front end requirements
2. User management system with at least one social service login (like facebook)
3. A profile page where the user can see a list of his favourite pokemons (each pokemon needs to have a star icon to make it favourite)
4. * Show your skills

** Show your skills, In addition to the requirements above you can implement anything you want using the API if you want to demonstrate more of your skills*