**Building an Event Finder App in React**

Project 1

Your project is under assessment. You will hear from us soon.

DESCRIPTION

Build a React app where users can view and promote upcoming events.

**Background of the problem statement:**

You are a web developer at a company called FindMyMovies. The company has decided to launch a new app called FindMyEvents where users can view and add upcoming events in their region. bBackend development has been outsourced and frontend will be developed in-house. During sprint planning, you agree to lead this project and develop an app that will let users find events using APIs that backend engineers will provide. The tasks you are responsible for:

* Building the application in React
* Setting up a JSON server for testing until backend APIs are ready
* Managing the state using Redux

**You must use the following tools:**

* React: To build the application
* Redux: To manage the state of components in the application
* Redux Thunk: To manage the async flow
* JSON Server: To build a placeholder backend for app development and testing
* React Router: To navigate within the app using URL links
* Bootstrap: To accelerate the development

**Following requirements should be met**:

* The app should be responsive.
* There should only be one source of truth for state management.
* The app should have functionalities for users to create, view, and delete events.

2.

**Pokemon Search Application**

**DESCRIPTION**

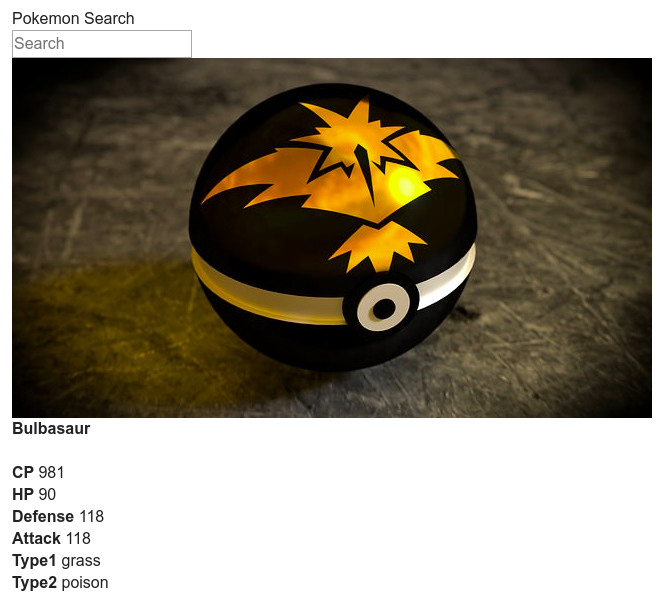
In the world of pokemon, where the user should be able to search for different pokemons, you have to create an application to search them with the following functionalities:

1. The ***Pokemon*** application should have the list of pokemon details which will be provided in the form of JSON data i.e., it should have name, image, CP, attack, defense type etc.
2. It should have a search bar so that user can search the pokemon based on the name of pokemon.
3. Since we have many pokemon, apply **pagination**, so that user can only see 10 pokemon at a time.

**Pagination**



**Search Bar**



**Note:**

1. Pagination should have “*page current-page of Total pages*”. For example: page 50 of 100

* The total pages should be calculated based on the JSON data provided.
* And the current page is calculated based on the page, the user is in, currently.
* Pagination should have id=”*pagination* ”, which should contain the details as listed above.
* The previous button should be disabled if it is in the first page and it should have id=”*previous*”.
* Next button, should show next set of pokemon (which is 10 pokemon on a click) and should have id=”*next*”.

1. Search bar should have an id=”*search* ”.
2. Pokemon's name should be inside the span tag (or any other tag) which should have id=”*name*”.