

Hey so you are 1 step closer to working with us. But between you & us, stands an assignment.

Video recording: <https://www.loom.com/share/2683ecef3042b599ca8672880fcc15>

As part of the assignment, below is the Problem Statement from **Parspec**. We request you to complete the assignment within **1 day**. In case of any query please get in touch with us.

Problem Statement:

You have to make a search input box which will search over a list of users.

The user object has the following fields

id: a unique id

name: user's name

items: list of items ordered by user

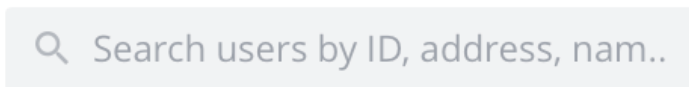
address: address of the user

pincode: user address pin code

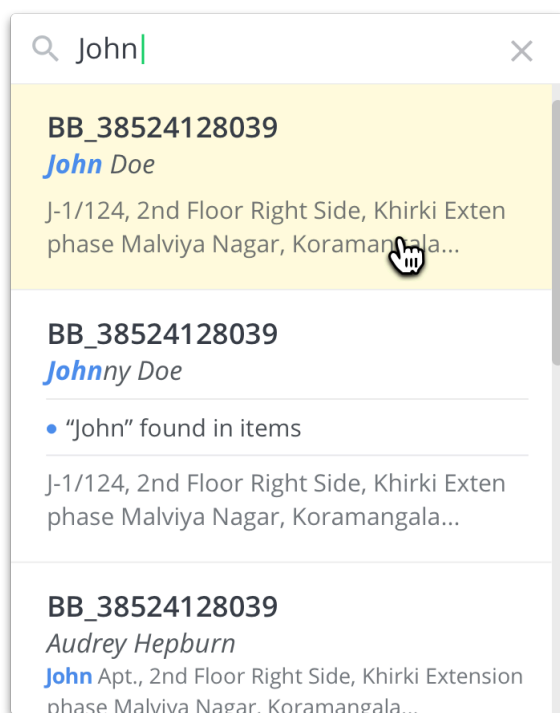
You have to implement search on all of the fields.

The search results will show up as a list of User Cards. Refer to the screenshots for how the UI would look like

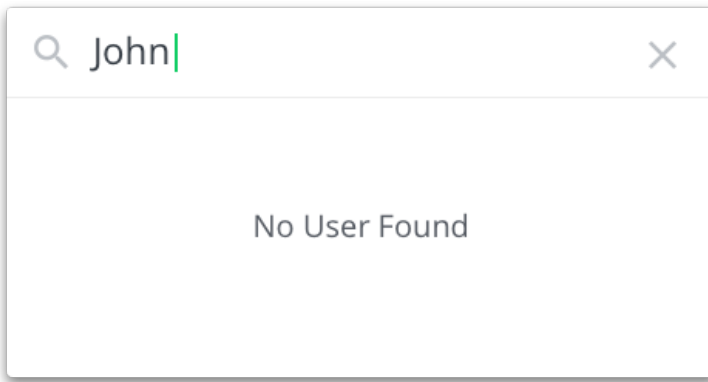
1. Placeholder (empty input)



2. Search Result List



### 3. No Results Card



## To Summarize

1. You have to implement search on all of the fields available in the JSON <http://www.mocky.io/v2/5ba8efb23100007200c2750c> . On typing in the search input box, the search results list opens up. The search could be just a string matching search.
2. Matching text should be highlighted in blue color as displayed in the search result list UI
3. If a search query is found in items, then it should show “{{queryValue}} found in items”, as displayed in the search result list UI. Eg: “john” found in items
4. The list of cards can be navigated through keyboard or mouse
5. Only one card should highlight at a time if both mouse and keyboard are used for navigation
6. Keyboard will take preference if mouse is kept hovered on the list, similarly mouse will take preference if keyboard navigation is not used
7. This behavior is similar to how **youtube** search works <https://www.youtube.com/>
8. When no search results are found, an empty card is displayed
9. The card list would be scrollable
10. The highlighted card (via keyboard/mouse) will scroll into view

[You can find the mock data here](#)

## Evaluation Criteria

1. Modularity of code
2. Handling of Keyboard navigation with mouse.
3. Handling ScrollIntoView

You have to use **React** for the webapp.

1. No jQuery.
2. No need for fancy CSS - We know you can do it if you can code this stuff. (But the card structure).
3. To submit the solution, please upload the entire folder on Google Drive / Dropbox, share the folder as 'anybody with a link can view' & then reply back with the link to this email.
4. **Do not use a library for handling ScrollIntoView.**