

Technical Tasks

Task 1:

Write a frontend application to search and list anime characters.

Technologies: React / VueJS / Angular

- Create a search page to show search results with fields shown in the wireframe below.
 - Show Max 15 results on the single page
 - Show a warning message, if there is no result returned for a given search query
 - By default, the page shows the top 15 characters if the user did not search anything
 - Show the user pagination button to go “Next” or “Back”
 - Clicking on the next button shows next 15 records or so
 - The application searches the anime characters as user types (In sort user does not need to click on the search button or similar to see results)
- API to use:
 - Documentation:
<https://docs.api.jikan.moe/#operation/getCharactersSearch>
 - Search Endpoint:

Method: GET

URL:

`https://api.jikan.moe/v4/characters?page=<page_number>&limit=<page_limit>&q=<search_string>&order_by=favorites&sort=desc`

Examples:

To get the top 15 characters the API request would look like below

Method: GET

URL :

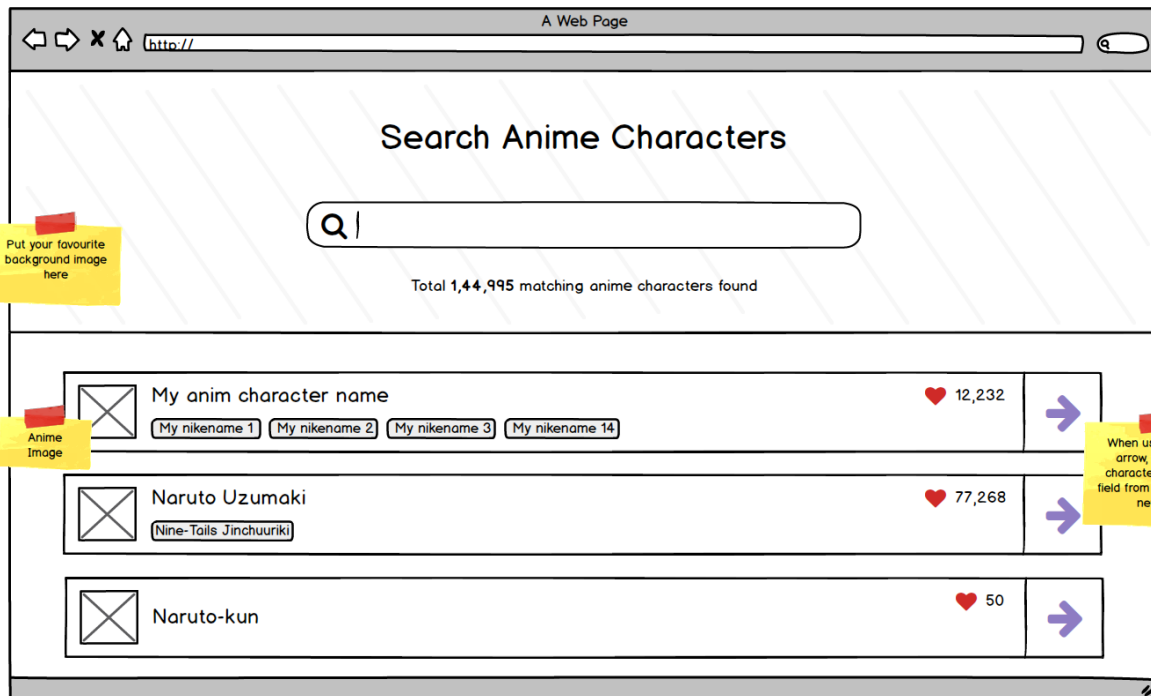
`https://api.jikan.moe/v4/characters?page=0&limit=15&q=&order_by=favorites&sort=desc`

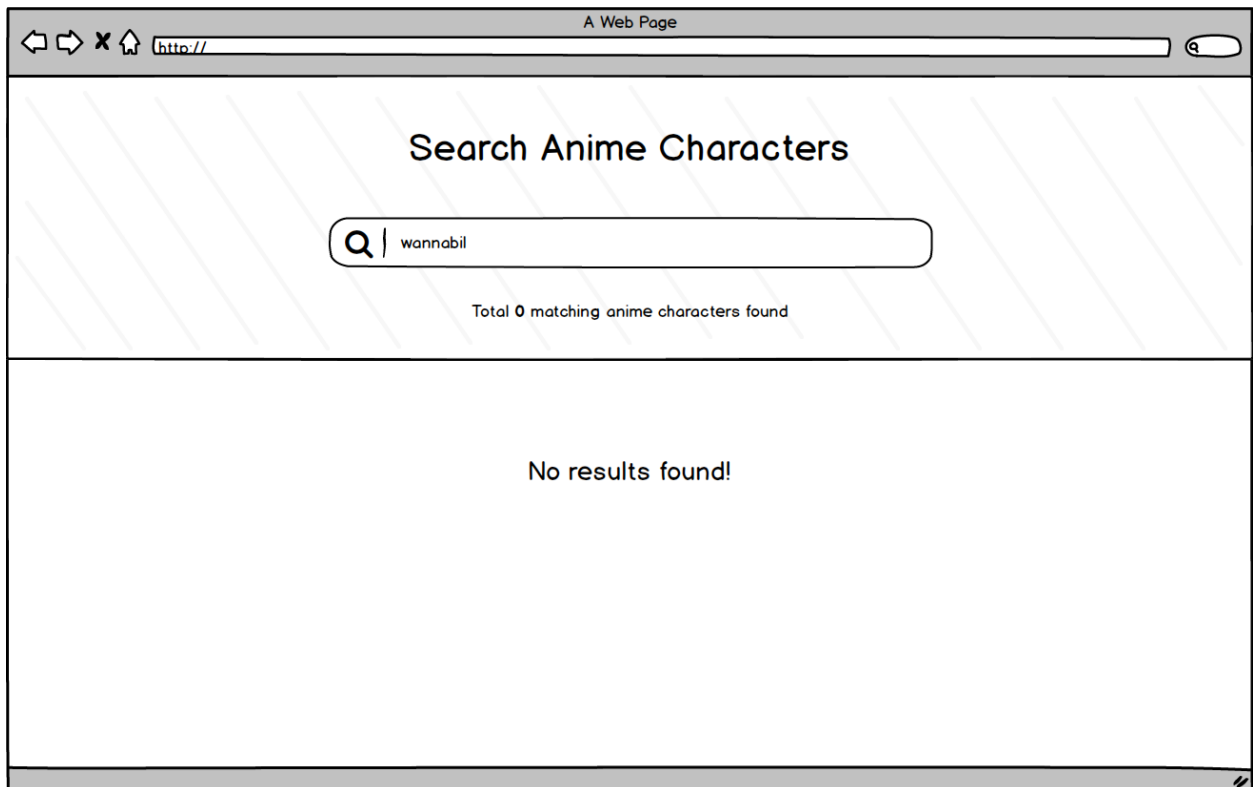
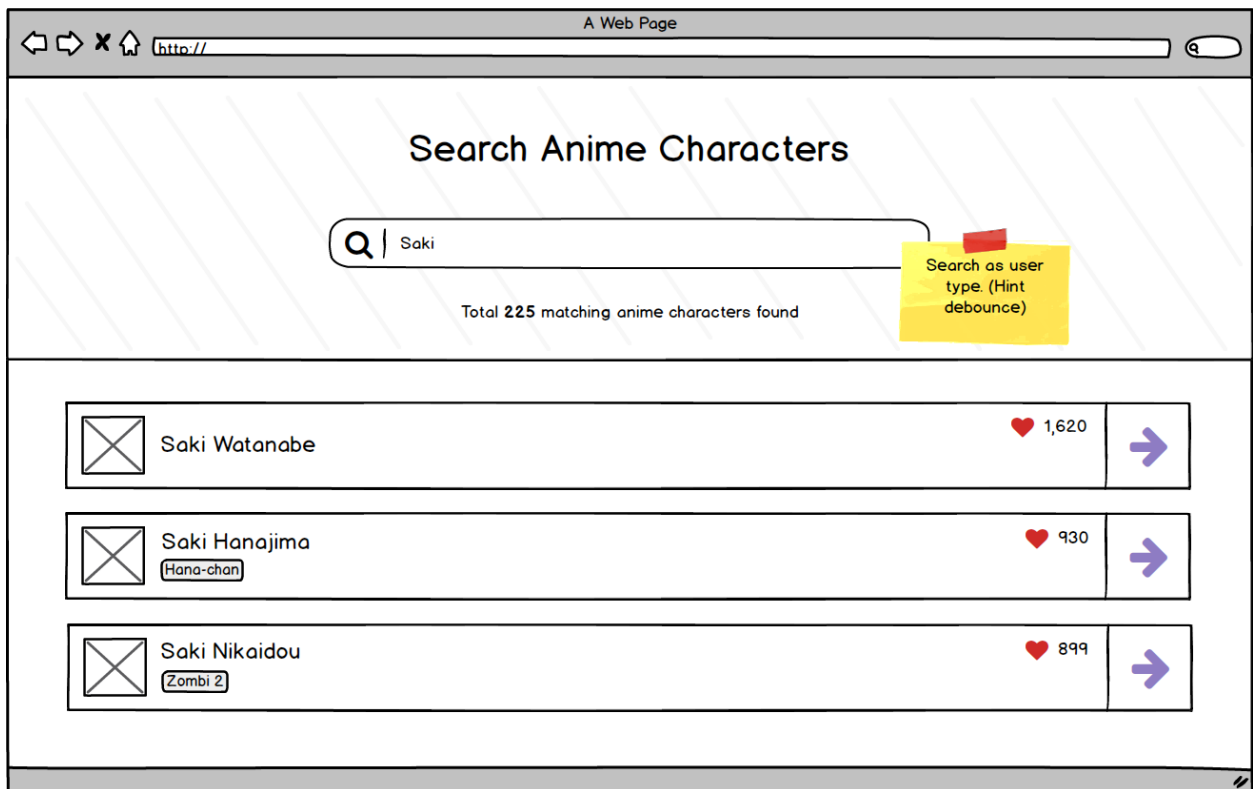
To search “saki” the API request looks like below

Method: GET

URL :

`https://api.jikan.moe/v4/characters?page=0&limit=15&q=saki&order_by=favorites&sort=desc`





Task 2:

Language: Javascript / C / C++ / Java / C# / PHP / Python or any other programming language

1. Write a program to print the following pattern

Sample Input

Please enter your lucky number: 5

```
      1
    1 3 A
  1 3 5 A B
1 3 5 7 A B C
1 3 5 7 9 A B C D
  1 3 5 7 A B C
    1 3 5 A B
      1 3 A
        1
```

Please enter your lucky number: 3

```
      1
    1 3 A
  1 3 5 A B
    1 3 A
      1
```

2. Write a program to print the Fibonacci series up to the number which is lesser than the given input.

```
User input: 120
Program output:
0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89

User input: 5
Program output:
0, 1, 1, 2, 3, 5
```

User input: 10

Program output:

0, 1, 1, 2, 3, 5, 8

Rules:

- You are allowed to use the internet for your task as long as you do not use the internet to search for any direct/indirect solution for your given task.
 - You can use the internet to download any libraries you need for the task.
 - You can use the internet to search for programming language references in case you can not recall certain syntax or functionalities
- You are not allowed to take any outside help for your task or any solution
 - We believe these tasks are the best way to check compatibility for both sides
 - Not doing interview tasks with honesty can take you to the wrong position in the company, which later result in unpleasant situations
- You can attempt Tasks in any frontend framework of your choice (Angular, React, Vue, Or any).
- Make sure you test your solution well and that it does not have any bugs.
- We expect you to submit your task within **4 days** from the date and time you start.
 - It is fine if you do not finish 100% of the task. You can submit whatever is done in **4 days**.
 - Try to do your task in stages so that there is at least something to show
- We do not have any intention to gain anything from the source code you write in your task. Any code you write, you have full ownership of that code. It is your IP. Any sample files or material we share with you for your task is our IP. You should delete any digital copy you own from us once your interview process is done (the result is out).
- Submission:
 - Via GitHub: Push your code on the GitHub repo shared with you.
OR
 - Via Email: On completion of the task create a zip file with all code and database dump. Upload the zip on google drive and share a link with us.

- Make sure you follow code quality standards you know, folder structure, formatting, and naming conventions....
- Do not worry about CSS, Use a UI kit in your task such as
 - <https://material.angular.io/components>
 - <https://vuetifyjs.com/en/>
 - <https://material-ui.com/>
 - Or any other of your choice