**Task 1:**

● Display the search results related to the searched term

Implemented the search functionality using Fetch API in Vanilla JavaScript and then rendered the search results to the application.

● Please consider what happens if you have more than 10 results.

Show only 10 ids or results for the page and the rest to be accessed by clicking the next and previous buttons at the bottom of the page.

● Please consider how you would handle it if there were no results for a given query.

Implemented the error handling, if no response from the API or the word given is not found in the Shakespeare playlist throws an error.

● Please consider that your website would be consumed by different devices.

Created a web application using Bootstrap which helps in making the responsive website which can be accessed on mobiles, tablets, laptop, PC and large screens.

**Task 2:**

If you wanted to improve the user experience or functionality of your website, what would you add? Without writing any implementations in code, please provide a paragraph or two in the documentation to explain your thoughts.

I have implemented the current solution using vanilla JavaScript. If we need to implement any robust functionalities, we can do it using various frameworks/libraries in which few functionalities are already built in. we just need to integrate the built-in functionalities in our application. Elastic-search API is one of the frameworks which helps in building search functionality using elastic-search.

Graphical user interface, text, application, email

Description automatically generated

SORT BY

FILTER

SHOW QTY

<< Showing 1 of 3 >>

* By using elastic-search library we can easily implement the sort, filter, relevant, show current previous and next page results which are in-built in Elastic.
* Auto completion of words using fuzzy search in elastic. It returns documents containing terms similar to that of the search element.

GET shakespeare/\_search

{

"query": {

"fuzzy": {

"speaker":{

"value":"Halmet", Instead of HAMLET but results the same

"fuzziness": "2"

}

}

}

}

* Boosting of words which is used to rank words and demote them using positive, negative and negative\_boost.
* Get Size from drop down and show no. of results based on selected value.
* Tracking performance of the application using Application Performance Monitoring service provided by Elastic Search.
* Track which words are searched more and provide these words in the popular section.
* Visualize the statistics of the results using Kibana service offered by Elastic search.

**Task 3**

What strategies would you employ to test the application you have just written? Without writing any implementations in code, please provide a paragraph or two about how you would test your website during both the development stage and also when deployed to a live, production environment.

* Searching for both lower-case and upper-case words leads to same results.
* Clear the search term and the results when clicked on clear button.
* Suggest the previously searched terms using
* Both mobile and web versions of the application show the results in right way. Responsive Web Application using Bootstrap. Tested it Using browser DevTools Dimensions.
* Check the application works well and in the same manner in different browsers.
* What if the search term has too many results to fetch? Asynchronous programming.
* What if the search term doesn’t exist in the playlists? Error message using catch and throw.
* What if user searches for blank spaces and then the word? Use trim to remove spaces before and after the text.
* What if the user misspells the search term? Use Fuzzy query.
* What if the user tries to search for a sentence instead of a word? Use setTimeout and call search function after some time when event key up happens.
* Show Errors based on the HTTP status codes.
* Considering CORS
* Consider words with same letters in it. Like when typing Hen in the search bar should return results of both Herny and Henrick as auto complete suggestion using Match Phrase prefix query.
* When clicked on Show it should display dropdown how many results to be shown for each page. Show Dropdown using datalist tag and options 10 20 50 100 in it.