Lab Task: JavaScript Project

Made by Engr. Zakria Bacha

Quote Generation Tool

1. First Create Interface

Create the below GUI using technologies HTML5, CSS3, or Bootstrap 5. The design must be responsive.

a. Make Header. (Header should be fixed).

b. Then work on body.

i. A quote box that displays generated quotes.

ii. Buttons for:

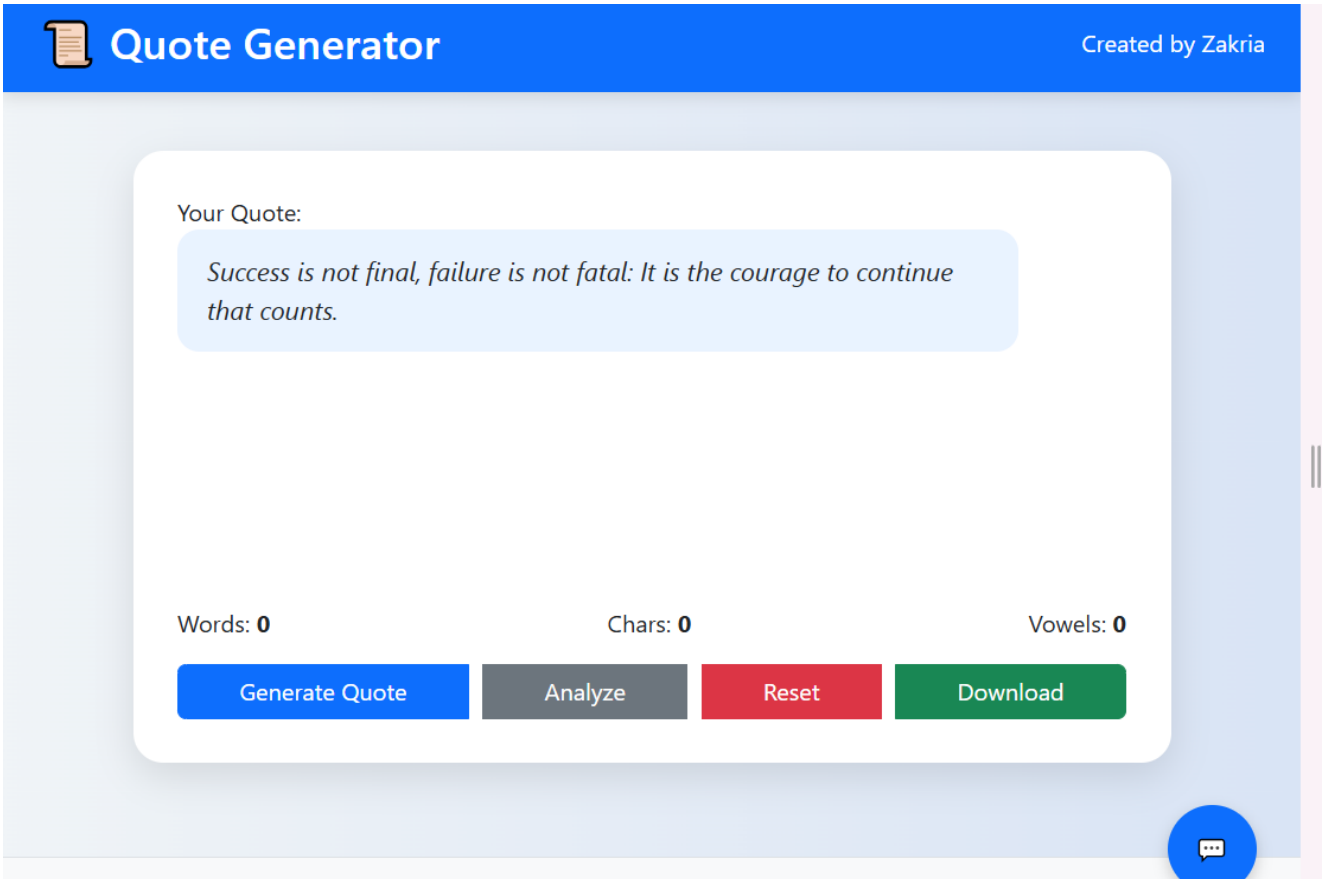
Show User History from Browser

Store User Text in local storage in Browser.

- Generate Quote
- Count Characters
- Count Words
- Count Vowels
- Reset
- Download PDF

2. Then make footer. (Footer should be fixed)

3. When user scrolls, header should not disappear.



Below figure 2 shows user history, when user click on history button. And user user first saves the quote.

Figure:01

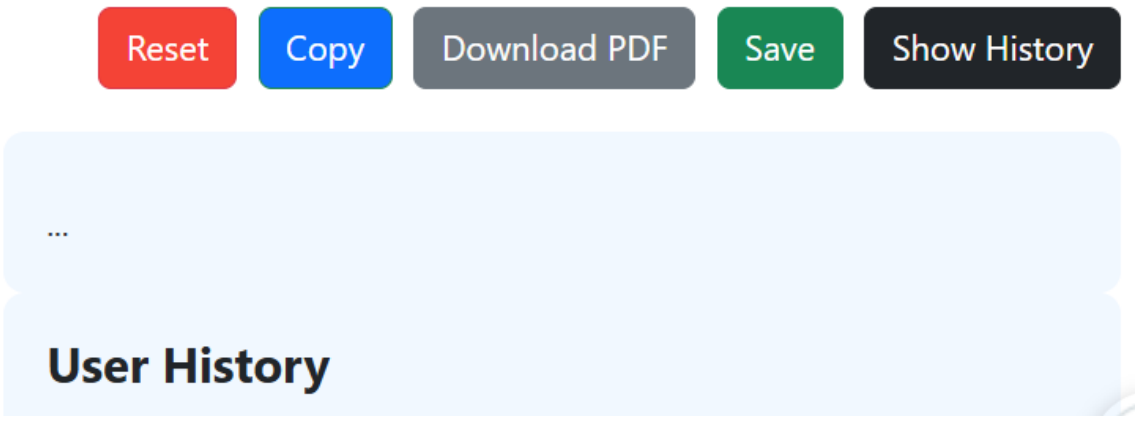


Figure:02

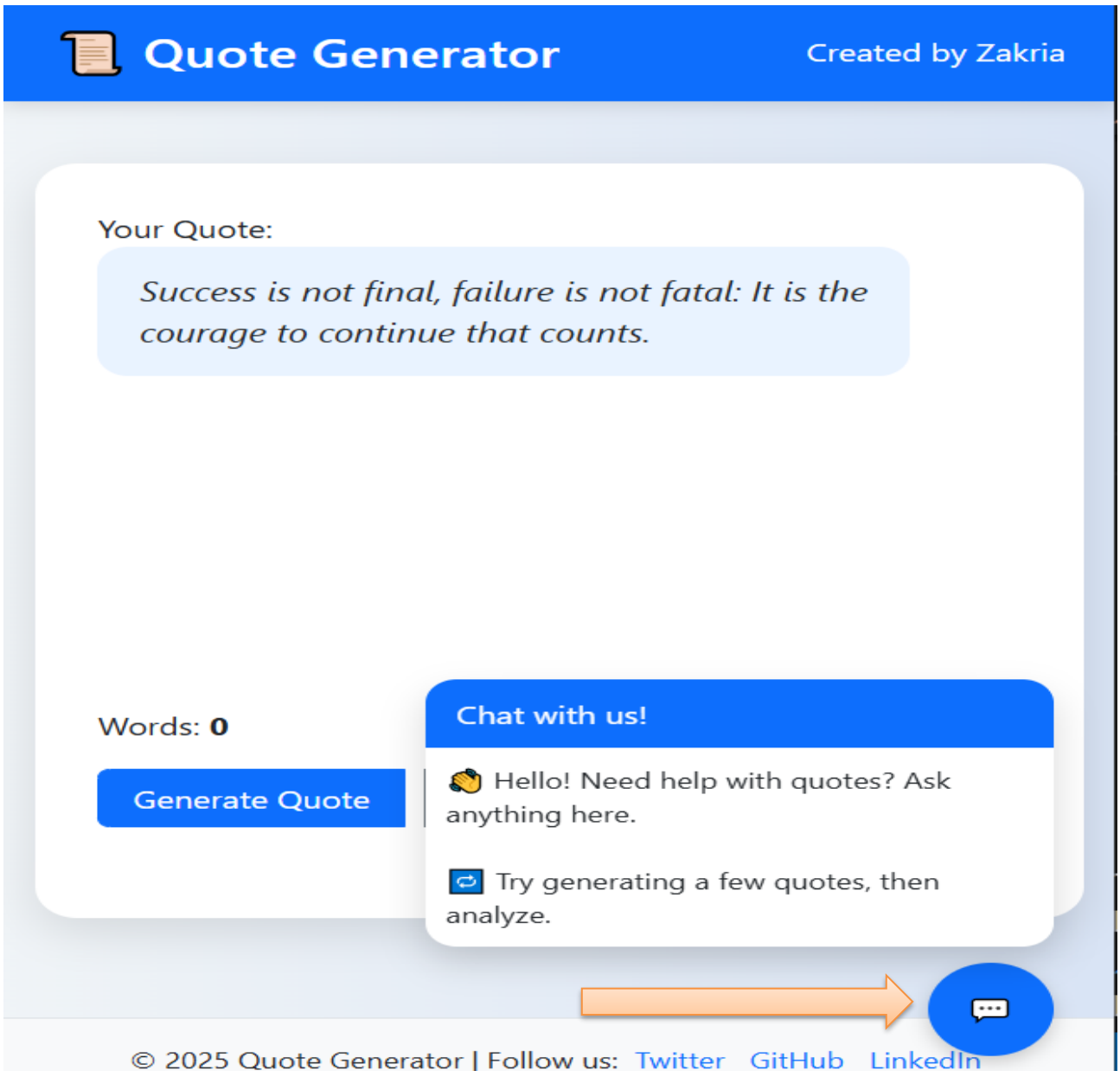

Quote Generator
Created by Zakria
Your Quote:
Success is not final, failure is not fatal: It is the courage to continue that counts.
Words: 0
Generate Quote
Chat with us!
Hello! Need help with quotes? Ask anything here.
Try generating a few quotes, then analyze.
© 2025 Quote Generator | Follow us: Twitter GitHub LinkedIn

Figure:03

2. Make It Interactive Using JavaScript

First, work on API. Because the quote will be generated by the server every time the user clicks the button.

Follow the steps for API integration:

a. Click on https://www.api-ninjas.com/api

b. Register yourself on the site.

c. Select Quotes API.

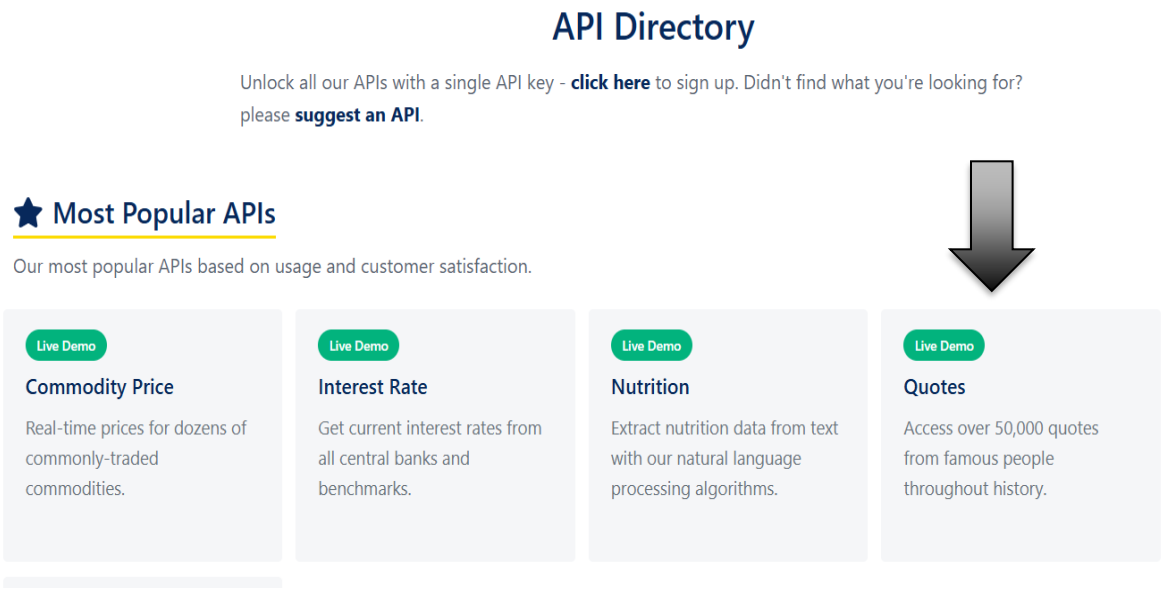


Figure:03

Secondly, generate API Key and API URL and use it in your code to receive quotes.

Quote Generator Software Function Points:

1. Generate Quote
2. Count Vowels
3. Count Words
4. Count Characters
5. Download Quote

6. Dummy chat window

7. Reset Quote

8. Store User Quote

9. Show User History (Quote)

Best Practices:

1. Use class and object concepts while applying JavaScript scripting language. API Key and URL should be private.

2. Follow the correct flow for writing code for each functionality, such as:

- First, the user can click on Generate button

- Then use character count functionality

- Vowel count functionality

- Word count functionality

- Then download, and so on.

3. Separate each section with a long comment line for better understandability of the code.

4. Divide your JavaScript code into three sections for DOM manipulation:

a. Target the DOM

b. Add Event Listener

c. Make Event Handler

5. Additionally, web app should be reliable that is any error or issues will be handle by software. ***For example, when there is no network, so network error message should be displayed to the user.***

6. It should be maintainable, any other changes in future should be accommodate easily.

7. You should call API via AJAX and then run your program.

8. You should call API via Fetch modern method of JavaScript.

9. Arrow function and add event listener should be used instead of traditional function for event handling.

10. Your code should contain JavaScript function such as spread operator, map, slice.