Front End Engineering-II

Project Report

Semester-IV (Batch-2022)

Character Counter

Supervised By:

Mr. Raveesh Samkaria

Submitted By:

Sambhav Yadav

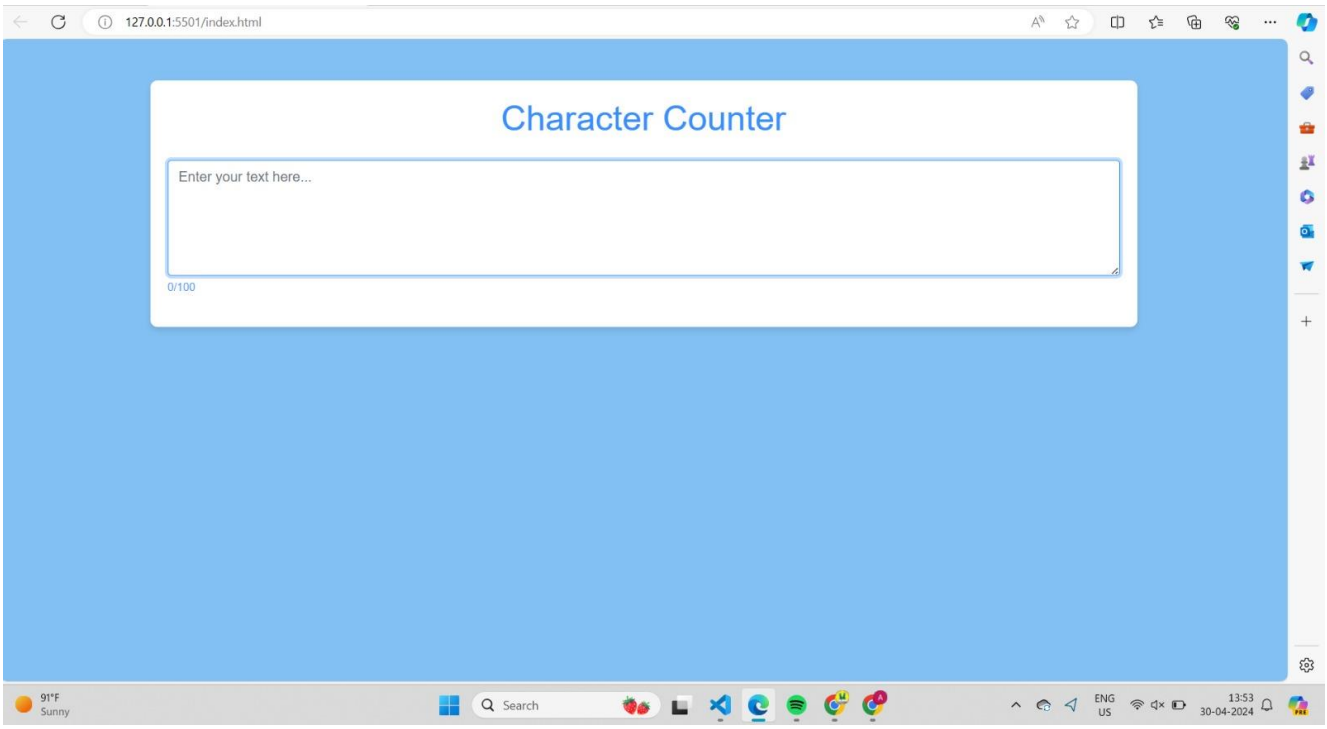
2210990774

G-12

Department of Computer Science and Engineering Chitkara University Institute of Engineering & Technology,
Chitkara University, Punjab

Abstract

"Character Counter" is a web application designed to help users count the number of characters in a piece of text. Built using HTML, CSS, and JavaScript, the application provides a user-friendly interface where users can input their text and instantly see the character count displayed on the screen. With its responsive design and intuitive functionality, Character Counter offers a convenient tool for writers, students, and anyone who needs to track the length of their text in real-time. Whether you're composing a tweet, writing an essay, or crafting an email, Character Counter simplifies the task of character counting, making it efficient and effortless.



INDEX

1. Introduction

"Character Counter" is a simple yet powerful web tool designed to provide users with instant character count feedback for any text input. Developed using HTML, CSS, and JavaScript, this application offers a minimalist interface where users can paste or type their text and instantly view the character count. With its responsive design and real-time updates, Character Counter is ideal for writers, students, and professionals who need to monitor text length for various purposes. Whether composing tweets, essays, or emails, Character Counter streamlines the process, making it efficient and hassle-free.

1.1 Background:

In an era where effective communication is paramount, the need to manage text length efficiently is ever-present. Whether crafting social media posts, composing emails, or writing essays, individuals often encounter the challenge of adhering to character limits or conveying information concisely. Recognizing this ubiquitous need, the Character Counter tool emerges as a solution, offering users a simple yet invaluable utility for tracking text length in real-time.

Amidst the digital landscape saturated with information overload, Character Counter stands out as a versatile tool that caters to a myriad of users. From writers striving for succinctness to social media enthusiasts navigating character restrictions, this tool provides a seamless experience, enabling users to focus on content creation without the distraction of manual character counting.

1.2 Objectives:

The core objective of our Character Counter project is to provide users with a reliable and user-friendly tool for accurately counting characters in their text. Our primary goals include:

Accuracy: Develop a character counting algorithm that accurately tallies the number of characters in a given text input, including spaces, punctuation, and special characters.

1.3 Significance:

Character Counter represents more than just a tool for tallying characters—it serves as a vital resource that enhances communication efficiency and effectiveness across various digital platforms. Its significance is multi-faceted:

2. Problem Statement

In the realm of digital communication, individuals frequently encounter challenges related to managing text length effectively. Existing text editing tools often lack built-in character counting functionality, leaving users to manually tally characters or rely on external resources. This can lead to inefficiencies, errors, and frustration, particularly when dealing with character-limited platforms such as social media or text input fields with specific requirements.

Furthermore, the absence of real-time character count feedback can hinder users' ability to gauge their text length accurately, resulting in overshooting character limits or unintentionally truncating their messages. This issue is exacerbated by the lack of customization options in many text editing tools, which fail to accommodate users' preferences regarding font size, spacing, or formatting.

3. Software Requirements

a) Integrated Development Environment (IDE):

- Visual Studio Code (VS Code) for code editing and project management.

b) Frontend Technologies:

- HTML: Markup language for structuring the web application.
- Boostrap: Styling language for enhancing the presentation and layout.
- JavaScript (JS): Programming language for implementing interactive features and quiz logic.

c) User Interface (UI) Framework:

- Tailwind CSS : Utility-first CSS framework for building responsive and customizable user interfaces.

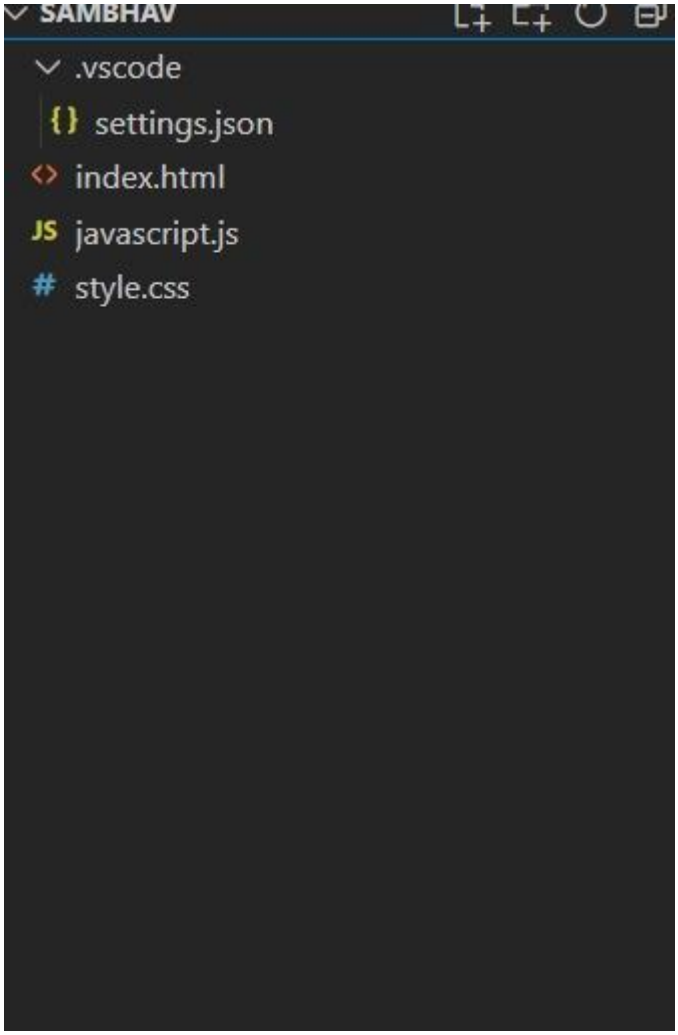
d) Version Control:

- Git: Distributed version control system for tracking changes in the project codebase and collaborating with team members effectively.

4. Proposed Design

- **User Interface Design:** Employ Tailwind CSS for a modern and responsive layout, focusing on simplicity and clarity.
- **Frontend Development:** Develop the quiz application using HTML, CSS, and JavaScript. Leverage HTML5 semantics for structural elements, CSS for styling.
- **User Experience Optimization:** Prioritize user interaction by providing real-time feedback on answer selection. Implement smooth transitions and animations using Tailwind CSS utilities to enhance the overall user experience.
- **Testing and Quality Assurance:** Conduct rigorous testing, including both manual and automated tests, to ensure the reliability and functionality of the application across different browsers and devices.
- **Documentation and Deployment:** Provide detailed documentation. Deploy on web server with domain. Maintain and update documentation regularly.

4.1 File Structure



4.2 HTML Code

These screenshots present the HTML code for our Ping Pong project, revealing the layout and content of our web pages in a code format.

index.html > html

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Character Counter</title>
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <div class="container my-5">
    <h1 class="text-center mb-4 primary-color">Character Counter</h1>
    <div class="form-group">
      <textarea class="form-control" id="textInput" rows="5" placeholder="Enter your text here..." maxlength="100"></textarea>
      <small id="counter" class="form-text counter-text">0/100</small>
    </div>
  </div>

  <script src="javascript.js"></script>
</body>
</html>
```

4.3 JavaScript Code

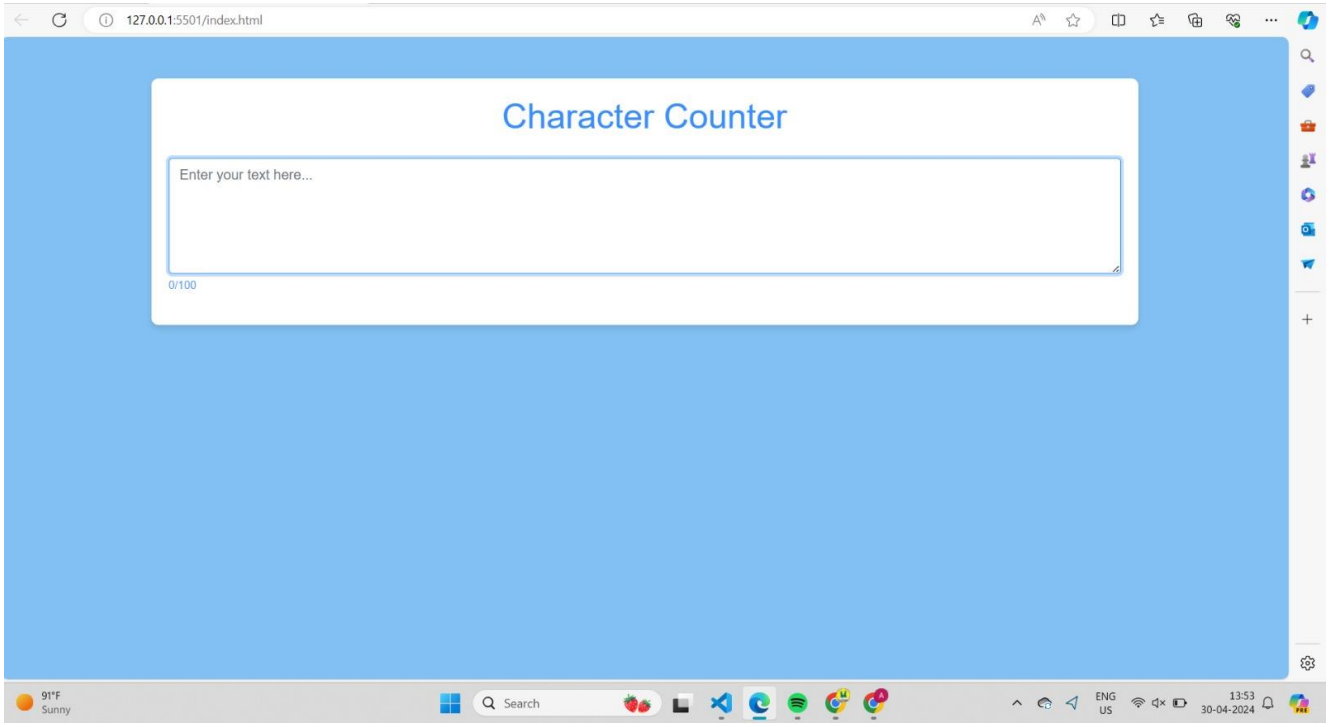
This screenshot exhibits the JS code for our Ping Pong App project, illustrating the different functions and events we trigger according to different scenarios.

javascript.js > updateCounter

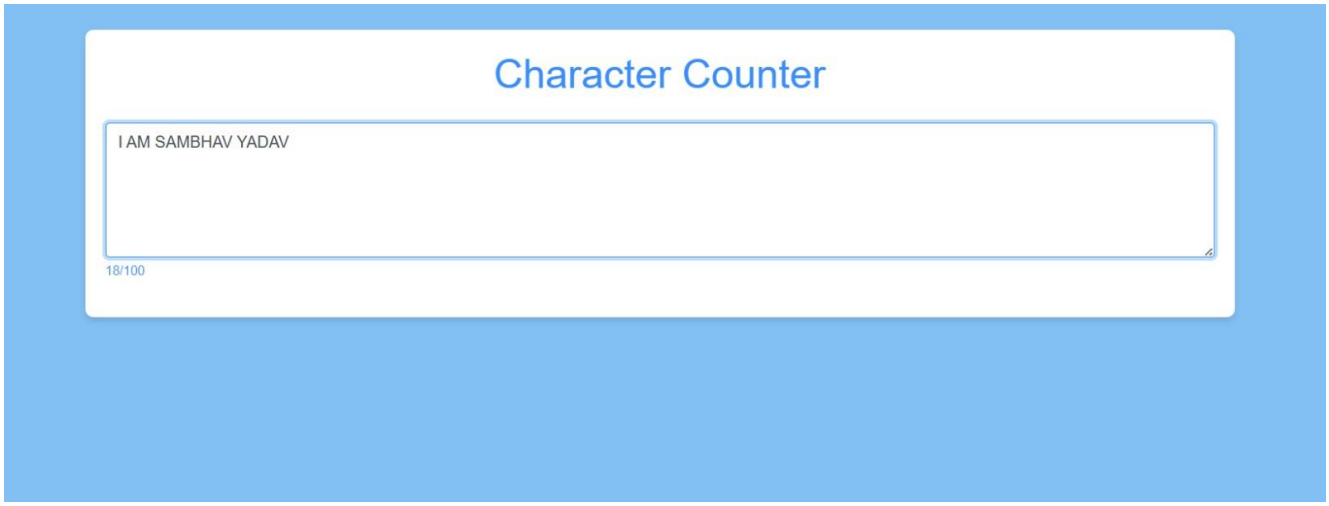
```javascript
const textInput = document.getElementById('textInput');
const counter = document.getElementById('counter');

textInput.addEventListener('input', updateCounter);

function updateCounter() {
  const textLength = textInput.value.length;
  const maxLength = textInput.getAttribute('maxlength');
  counter.textContent = `${textLength}/${maxLength}`;

  // Change counter text color based on character count
  if (textLength >= maxLength * 0.8) {
    counter.classList.add('text-danger');
  } else {
    counter.classList.remove('text-danger');
  }
}
```

5. Results

Project Screenshots for all scenarios:



Static View



Writing Paragraph

6. References

- **HTML, CSS, and JavaScript Documentation:**
  - **Mozilla Developer Network (MDN) - HTML:**

  https://developer.mozilla.org/enUS/docs/Web/HTML
  - **Mozilla Developer Network (MDN) JavaScript:**

  https://developer.mozilla.org/en-US/docs/Web/JavaScript
- **Boostrap Documentation:**
  - **Boostrap Official Documentation:**

  https://boostrap.com/
  - **GeeksForGeeks Tailwind Tutorial:**

  https://www.geeksforgeeks.org/tailwind-css/