TO DO LIST

Project report

Aashrita Kolisetty(R21EJ061)

Name of School or Institution-C&IT

Course Name-*Web Technology*

Instructor - *Prof Pavankumar Naik*

Date -24/11/23

Abstract

This project presents a simple and intuitive To-Do List application implemented using HTML. The web-based interface allows users to efficiently manage their tasks with a clean and user-friendly design. The application supports adding, editing, and deleting tasks, and utilizes HTML elements to create an interactive and responsive user experience. Through the use of HTML forms and JavaScript for dynamic behavior, users can seamlessly interact with the to-do list, marking tasks as complete and organizing them based on priority. The project showcases the fundamental principles of web development, emphasizing accessibility and ease of use. Overall, this To-Do List in HTML offers a practical solution for individuals seeking a straightforward and functional task management tool.

Introduction

In our fast-paced lives, effective task management is crucial, and what better way to stay organized than with a To-Do List? This project explores the creation of a To-Do List application using HTML, the backbone of web development. Designed with simplicity and functionality in mind, this web-based to-do list allows users to streamline their daily tasks effortlessly.

As we delve into the project, we'll leverage the power of HTML to structure the user interface and create a seamless experience for managing tasks. From adding and editing tasks to marking them as complete, our To-Do List in HTML will showcase the fundamental principles of web development, making it accessible to users of all levels.

Objectives

Here are some comprehensive objectives for developing a To-Do List website:

User-Friendly Interface:

- Design a clean and intuitive user interface that is easy to navigate.
- Ensure a straightforward and pleasant user experience for adding, editing, and managing tasks.

Task Organization and Prioritization:

- Implement features for organizing tasks based on priority, due date, or categories.
- Provide users with options to categorize and sort their tasks for better productivity.

Cross-Browser Compatibility:

- Develop a To-Do List website that functions consistently across different web browsers.
- Test and ensure compatibility with popular browsers like Chrome, Firefox, Safari, and Edge.

Responsive Design:

- Create a responsive design that adapts to various screen sizes, including desktops, tablets, and smartphones.

Problem definition

In today's fast-paced world, individuals face challenges in effectively organizing and managing their daily tasks. Traditional methods of jotting down tasks on paper or using standalone applications often lack the flexibility and accessibility needed in a digitally connected environment. Users often encounter the following issues:

Disjointed Task Management:

- Users struggle with managing tasks across various platforms, leading to disjointed and inefficient workflows.

Limited Accessibility:

- Existing solutions may not provide seamless accessibility across different devices, hindering users from accessing their task lists on the go.

Lack of Collaboration:

- Collaboration on task lists is often cumbersome, with limited options for sharing and real-time updates, especially in a team or group setting.

Ineffective Prioritization:

- Users may find it challenging to prioritize tasks effectively, leading to missed deadlines and decreased productivity.

System design

HTML:

- Structure the web page with HTML elements for headers, forms, lists, etc.

CSS:

- Style the webpage for a visually appealing and responsive design.

JavaScript:

- Implement dynamic behavior for adding, editing, and deleting tasks.
- Enable real-time updates without page refresh

System requirements

Web Browser:

- Compatible with modern web browsers such as Chrome, Firefox, Safari, and Edge.

HTML5:

- Browser support for HTML5 features.

CSS3:

- Browser support for CSS3 styling.

JavaScript:

- Enabled in the user's browser.

Implementation

index.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0"
/>
    <link rel="stylesheet" href="style.css" />
    <title>To do List</title>
  </head>
  <body>
    <h1>todos</h1>
    <form id="form">
      <input type="text" class="input" id="input" placeholder="Enter your
to do" autocomplete="off">

      <ul class="todos" id="todos"></ul>
    </form>
    <small>Left click to toggle completed. <br> Right click to delete
todo</small>

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

style.css

```
@import
url('https://fonts.googleapis.com/css2?family=Poppins:wght@200;400&display
=swap');

* {
  box-sizing: border-box;
}

body {
  background-image:url(background.jpg.jpg);
  color: #444;
  font-family: 'Poppins', sans-serif;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  height: 100vh;
  margin: 0;
}

h1 {
  color: rgb(49, 92, 236);
  font-size: 10rem;
  text-align: center;
  opacity: 0.4;
}

form {
  box-shadow: 0 4px 10px rgba(0, 0, 0, 0.1);
  max-width: 100%;
  width: 400px;
}

.input {
```

```
  border: none;
  color: #444;
  font-size: 2rem;
  padding: 1rem 2rem;
  display: block;
  width: 100%;
}

.input::placeholder {
  color: #d5d5d5;
}

.input:focus {
  outline-color: rgb(179, 131, 226);
}

.todos {
  background-color: #fff;
  padding: 0;
  margin: 0;
  list-style-type: none;
}

.todos li {
  border-top: 1px solid #e5e5e5;
  cursor: pointer;
  font-size: 1.5rem;
  padding: 1rem 2rem;
}

.todos li.completed {
  color: #b6b6b6;
  text-decoration: line-through;
}

small {
```

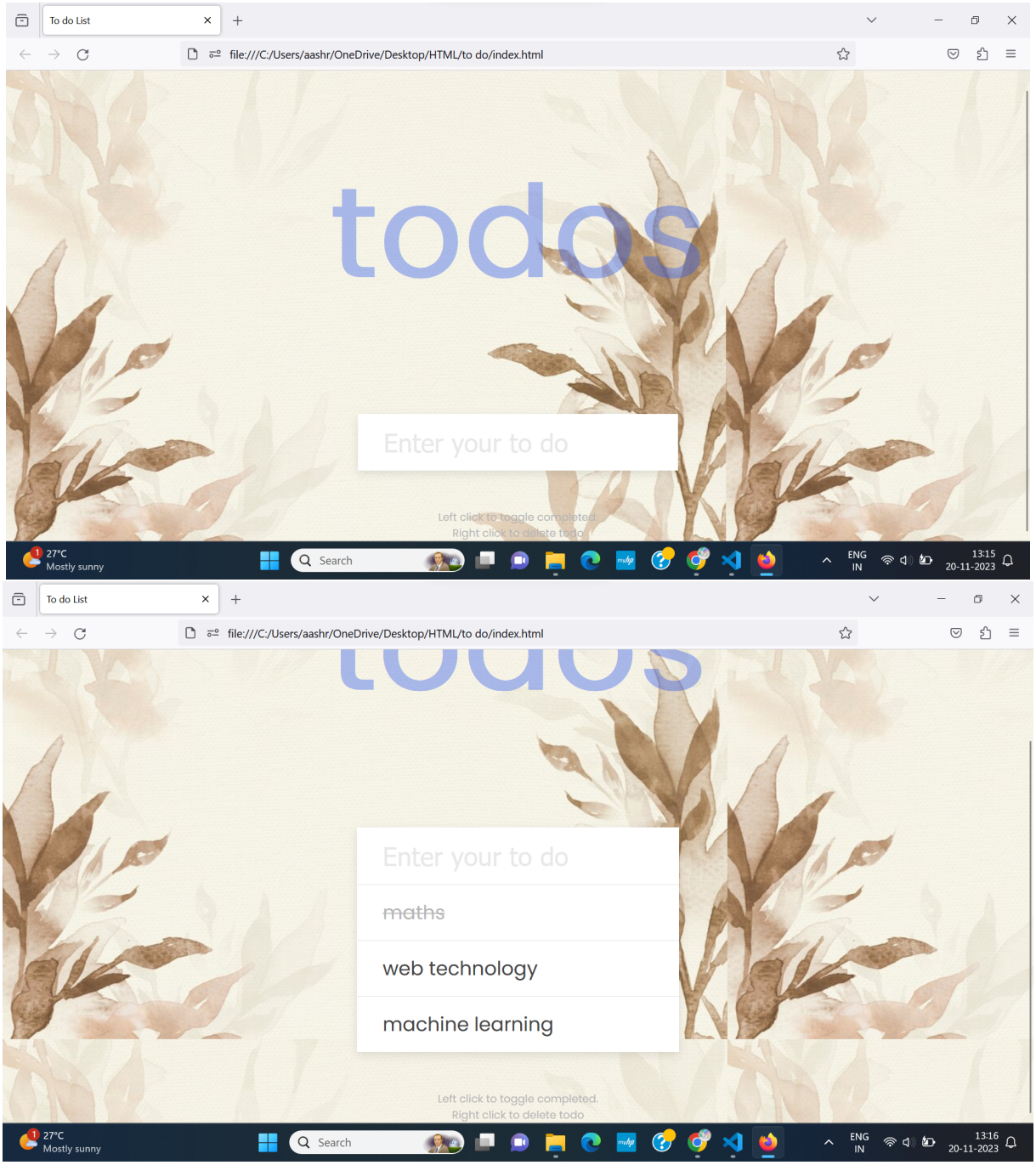
```
  color: #b5b5b5;
  margin-top: 3rem;
  text-align: center;
}
```

script.js

```
const form = document.getElementById('form')
const input = document.getElementById('input')
const todosUL = document.getElementById('todos')

const todos = JSON.parse(localStorage.getItem('todos'))

if(todos) {
    todos.forEach(todo => addTodo(todo))
}

form.addEventListener('submit', (e) => {
    e.preventDefault()

    addTodo()
})

function addTodo(todo) {
    let todoText = input.value

    if(todo) {
        todoText = todo.text
    }

    if(todoText) {
        const todoEl = document.createElement('li')
        if(todo && todo.completed) {
            todoEl.classList.add('completed')
        }
```

```
        todoEl.innerText = todoText

        todoEl.addEventListener('click', () => {
            todoEl.classList.toggle('completed')
            updateLS()
        })

        todoEl.addEventListener('contextmenu', (e) => {
            e.preventDefault()

            todoEl.remove()
            updateLS()
        })

        todosUL.appendChild(todoEl)

        input.value = ''

        updateLS()
    }
}

function updateLS() {
    todosEl = document.querySelectorAll('li')

    const todos = []

    todosEl.forEach(todoEl => {
        todos.push({
            text: todoEl.innerText,
            completed: todoEl.classList.contains('completed')
        })
    })

    localStorage.setItem('todos', JSON.stringify(todos))
}Results
```

Results


To do List
file:///C:/Users/aashr/OneDrive/Desktop/HTML/to do/index.html
todos
Enter your to do
Left click to toggle completed.
Right click to delete todo
27°C
Mostly sunny
Search
ENG
IN
13:15
20-11-2023
To do List
file:///C:/Users/aashr/OneDrive/Desktop/HTML/to do/index.html
Enter your to do
maths
web technology
machine learning
Left click to toggle completed.
Right click to delete todo
27°C
Mostly sunny
Search
ENG
IN
13:16
20-11-2023

Conclusion

In conclusion, the creation of a To-Do List website provides a practical and user-centric solution to the challenges individuals face in managing their tasks efficiently. By leveraging fundamental web development technologies such as HTML, CSS, and JavaScript, we've crafted a simple yet effective platform for users to organize their daily responsibilities.

The intuitive user interface ensures ease of use, allowing users to seamlessly add, edit, and delete tasks. The real-time updates enhance the user experience by providing instant feedback without the need for page refreshes. This streamlined approach prioritizes accessibility, making the To-Do List website compatible with various devices and browsers, empowering users to stay organized on the go.

.

References

1. https://freshman.tech/to-do-list/
2. https://codepen.io/rajdgreat007/pen/edvZpx
3. https://www.geeksforgeeks.org/productivity-improvement-tips/