PROJECT-CREATE A TO DO LIST

ABSTRACTION

A single page application (SPA) is a web application that uses only one HTML web page as a shell for all the application's web pages and whose end-user interactions are implemented by using JavaScript, HTML, and CSS, bootstrap. Most of the SPA development is done on the front end as opposed to traditional web applications that rely heavily on web server interactions and that reload new web pages whenever navigation occurs. SPAs resemble native applications in their behavior and development but they run inside a browser process as opposed to native applications, which run in their own process. In order to understand why SPA development is so trendy today, we need to understand the changes that happened in the web environment that led to today's SPAs. In this chapter you will learn a little bit about web development history and reasons to build SPAs in the first place.

OBJECTIVE

The objective of this create to do list website is that users can learn about the company or website without having to navigate .this website helps the users to learn about 1stop.

Sometime ,having too many pages on website can overwhelm the user . they’re not easy to browse and can take long time to find exactly what you are looking for

.the main goal was that the project website to act as the core communication tool.it is to be simple and light ,more informative as possible, maintains easiness and content update

INTRODUCTION

Web development is the work involved in developing a website for the Internet (World Wide Web) or an intranet (a private network). The webpage developed can be both static and dynamic (using more complex languages such as CGI, AJAX, ASP, ASP.NET).

Static webpages are generally written in simpler languages such as HTML, JavaScript, CSS, etc. In our project we had developed a static web page with all basic functionalities included in it.

METHODOLOGY

Depending on the project, the front-end and back-end development can go either in parallel or the back-end is followed by the front-end. A front-end developer implements all visual features and makes sure everything is pixel-perfect, and that a website is cross-browser compatible.

Our project is all about developing a static webpage, so in out project we had no work much related to the back-end development.

Why create a static website?

Static websites are quick and easy to create.

It is cheap to host.

Static websites are secure.

It is fun and you can create awesome sites with HTML, CSS, and JavaScript

front-end development:

step 1: **Create the structure with HTML.**

HTML allows users to create and structure sections, headings, links, paragraphs, and more, on a website using various tags and elements. Almost everything you want to create on a web page can be done using a specific HTML code.

```
<!DOCTYPE html>

<html>
<html lang=” en”>
```

```
<title>Page Title</title>
<body>

<h1>This is your first header</h1>
<p>This is paragraph text that pertains to your first header.</p>

<h2>This is your second header</h2>
<p>This is paragraph text that pertains to your second header.</p>

<ul>
<li>List</li>
<li>List</li>
<li>List</li>
</ul>

<img src=”Image.jpg” alt=”Image-Alt-Text” style=”width:600px;height:450px”>
```

```
Click the link <a href= “https://learn.g2crowd.com/G2”>here</a> to be taken to the G2 Learning Hub.

</body>
</html>
```

Step 2 :

Style with CSS or Bootstrap:

CSS is the language for describing the presentation of Web pages, including colors ， layout, and fonts. It allows one to adapt the presentation to different types of devices, such as large screens, small screens, or printers.

As we know that CSS is a styling language while Bootstrap is a CSS framework. Much like how jQuery is a framework of JavaScript. Bootstrap might be easier to use because it has a comprehensive grid system with many elements already built so it saves up time .

Step 3:

Make it interactive with JavaScript.

JavaScript is a scripting language that enables you to create dynamically updating content, control multimedia, animate images, and pretty much everything else.

CODE:

```
<!DOCTYPE html>
<html>
    <head>
```

```
        <meta charset="utf-8">
        <meta name="viewport" content="width=device-width,initial-scale=1">
        <link href="C:\Users\ASUS\OneDrive\Desktop\Demo.java\vendor\bootstrap\css\bootstrap.css" rel="stylesheet">
        <link href="" rel="stylesheet">
        <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.10.4/font/bootstrap-icons.css">
        <title>
            HTML TodoList
        </title>
    </head>
    <body>
        <nav class="navbar navbar-expand-lg navbar-light bg-light">
            <div class="container-fluid">
                <a href="#" class="navbar-brand">
```

```
<img src="Downloads/logo.png" class="img-fluid" alt="logo" width="150">
            </a>
            <button type="button" class="navbar-toggler" data-bs-toggle="collapse" data-bs-
target="#navbar">
               <i class="bi bi-list"></i>
            </button>
            <div class="collapse navbar-collapse" id="navbar">
               <div class="navbar-nav ms-auto">

               </div>
            </div>

         </div>
      </nav>
      <div class="container p-5">
```

```
        <div class="mb-2">
            <button type="button" class="btn btn-outline-primary" onclick="showAddTaskModal()">Add Task</button>
        </div>



<div class="d-flex justify-content-center">
            <div class="col-sm-12 col-md-12 col-lg-12">
                <div class="card">
                    <div class="card-body">
                        <div class="table"></div>
                            <table class="table">
                                <thead class="text-center">
                                    <th>#</th>
                                    <th>Task/Description</th>
```

```
                        <th>Responsible</th>
                        <th>ETA</th>
                        <th>Action</th>
                    </thead>
                    <tbody class="text-center" id="taskTableBody">

                    </tbody>
                </table>
            </div>
        </div>
    </div>
</div>
```

```
  <div class="modal fade" id="AddTaskModal" data-bs-backdrop="static" data-bs-keyboard="false" tabindex="-1" aria-labelledby="AddTaskModalLabel" aria-hidden="true">
        <form id="taskInputForm">
          <div class="modal-dialog">
            <div class="modal-content">
              <div class="modal-header">
                <h5 class="modal-title" id="addTaskModalLbel">AddTask</h5>
                <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="close"></button>
              </div>
              <div class="modal-body">
                <div class="mb-1">
                  <label for="addTaskTextArea" class="form-label">Task/Description</label>
                  <textarea class="form-control" id="addTaskTextArea" name="taskDescription" row="1" placeholder="Add your Task/Description"></textarea>
                </div>
```

```
                    <div class="mb-1">
                        <label for="addTaskResponsible" class="form-label">Responsible</label>
<input type="text" class="form-control" id="addTaskResponsibleperson"
name="addTaskResponsibleperson" placeholder="Add the Responsible Person's Name"></textarea>
                    </div>
                    <div class="mb-1">
                        <label for="addTaskResponsible" class="form-label">ETA</label>
                        <input type="datetime-local" class="form-control" id="addETA" name="taskETA"
placeholder="Click to Add time"></textarea>
                    </div>
                </div>
<div class="modal-footer">
                    <button type="button" class="btn btn-secondary" data-bs-dismiss="cancel"></button>
                    <button type="button" class="btn btn-primary" onclick="addTask()">Add Task</button>
                </div>
```

```
            </div>
          </div>
        </form>
      </div>
<div class="modal fade" id="updateTaskModal" data-bs-backdrop="static" data-bs-keyboard="false" tabindex="-1" aria-labelledby="updateTaskModalLabel" aria-hidden="true">
        <form id="taskUpdateForm">
          <div class="modal-dialog">
            <div class="modal-content">
              <div class="modal-header">
                <h5 class="modal-title" id="editTaskModalLbel">AddTask</h5>
                <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="close"></button>
              </div>
              <div class="modal-body">
```

```
                    <div class="mb-1">
                        <label for="editTaskTextArea" class="form-label">Task/Description</label>
                        <textarea class="form-control" id="editTaskTextArea" name="taskDescription" row="1" placeholder="Add your Task/Description"></textarea>
                    </div>
<div class="mb-1">
                        <label for="addTaskResponsible" class="form-label">Responsible</label>
                        <input type="text" class="form-control" id="editTaskResponsiblePerson" name="addTaskResponsibleperson" placeholder="Add the Responsible Person's Name"></textarea>
                    </div>
                    <div class="mb-1">
                        <label for="addTaskResponsible" class="form-label">ETA</label>
                        <input type="datetime-local" class="form-control" id="editETA" name="taskETA" placeholder="Click to Add time"></textarea>
                    </div>
                    <input type="hidden" id="editindex">
```

```
                </div>
                <div class="modal-footer">
                    <button type="button" class="btn btn-secondary" data-bs-dismiss="cancel"></button>
                    <button type="button" class="btn btn-primary" onclick="updateTask()">Add
Task</button>
                </div>
            </div>
  </div>
        </form>
    </div>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.4/jquery.min.js"></script>
    <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.7/dist/umd/popper.min.js"
integrity="sha384-zYPOMqeu1DAVkHiLqWBUTcbYfZ8osu1Nd6Z89ify25QV9guujx43ITvfi12/QExE"
crossorigin="anonymous"></script
    <script src="vendor/bootstrap/js/bootstrap.js"></script>
    <script>
```

```
createHTMLfromStorage();
function showAddTaskModal(){
    $("#addTaskModal").modal('show')
}
function addTask(){
    console.log("Add Task clicked")
    $("#addTaskModal").modal('hide')
    var dataArr=($("#taskInputTask").serializeArray())
    var taskObject=new Object();
    var StorageObjectArr=[];
    var StorageObject=localStorage.getItem('taskStorage')
    for(var i in dataArr){
        var name=dataArr[i]['name']
        var name=dataArr[i]['value']
        taskObject[name]=value
```

```
    }
    if(StorageObject!=null && StorageObject!=undefined && StorageObject!=''){
        StorageObjectArr=JSON.parse(StorageObject)
        StorageObjectArr.push(taskObject)
    }else{
    StorageObjectArr.push(taskObject)
    }
    localStorage.setItem('taskStorage',JSON.stringify(StorageObjectArr))
    console.log(StorageObjectArr)
    createHTMLfromStorage()
    $("#taskInputForm").trigger('reset')
}
function createHTMLfromStorage(){
    var StorageObjectArr=[];
    var StorageObject=localStorage.getItem('taskStorage')
```

```
var StorageObjectArr=JSON.parse(StorageObject)
var html='';
console.log(StorageObjectArr)
if(StorageObject!=null && StorageObject!=undefined && StorageObject!=''){
   if(StorageObjectArr && StorageObjectArr.length>0){
      for(let i in StorageObjectArr){
         var date=new Date(StorageObjectArr[i]['taskETA'])
         html=html+'<tr>'
                  +'<td>'+(parseInt(i)+1)+'</td>'
                  +'<td>'+StorageObjectArr[i]['taskDescription']+'</td>'
                  +'<td>'+StorageObjectArr[i]['taskResponsiblePerson']+'</td>'
                  +'<td>'+date.toUTCString()+'</td>'
                  +'<td><i class="bi bi-check-circle-fill" onclick="markAsDone('+i+')"></i><i class="bi bi-pencil-square" onclick="editTask('+i+')"></i></td></tr>'
      }
```

```
        }else{
            html='<tr><td colspan="5">No Tasks Added yet</td></tr>'
        }
    }
    $("#taskTableBody").html(html)
}
function markAsDone(index){

    var StorageObjectArr=[];
    var StorageObject=localStorage.getItem('taskStorage');
    if(StorageObject!=null && StorageObject!=undefined && StorageObject!=''){
        StorageObjectArr=JSON.parse(StorageObject)
        StorageObjectArr.pop(index)
    }
    localStorage.setItem('taskStorage',JSON.stringify(StorageObjectArr))
```

```
    createHTMLfromStorage()



}
function editTask(index){
    var StorageObject=localStorage.getItem('taskStorage');
    var StorageObjectArr=[];
    if(StorageObject!=null && StorageObject!=undefined && StorageObject!=''){
        StorageObjectArr=JSON.parse(StorageObject)
        $("#editTaskTextArea").val(StorageObjectArr[index]['taskDescription'])
        $("#editTaskResponsiblePerson").val(StorageObjectArr[index]['taskResponsiblePerson'])
        $("#editETA").val(StorageObjectArr[index]['taskETA'])
        $("#editindex").val(index)
        $("#updateTaskModal").modal('show')
    }
```

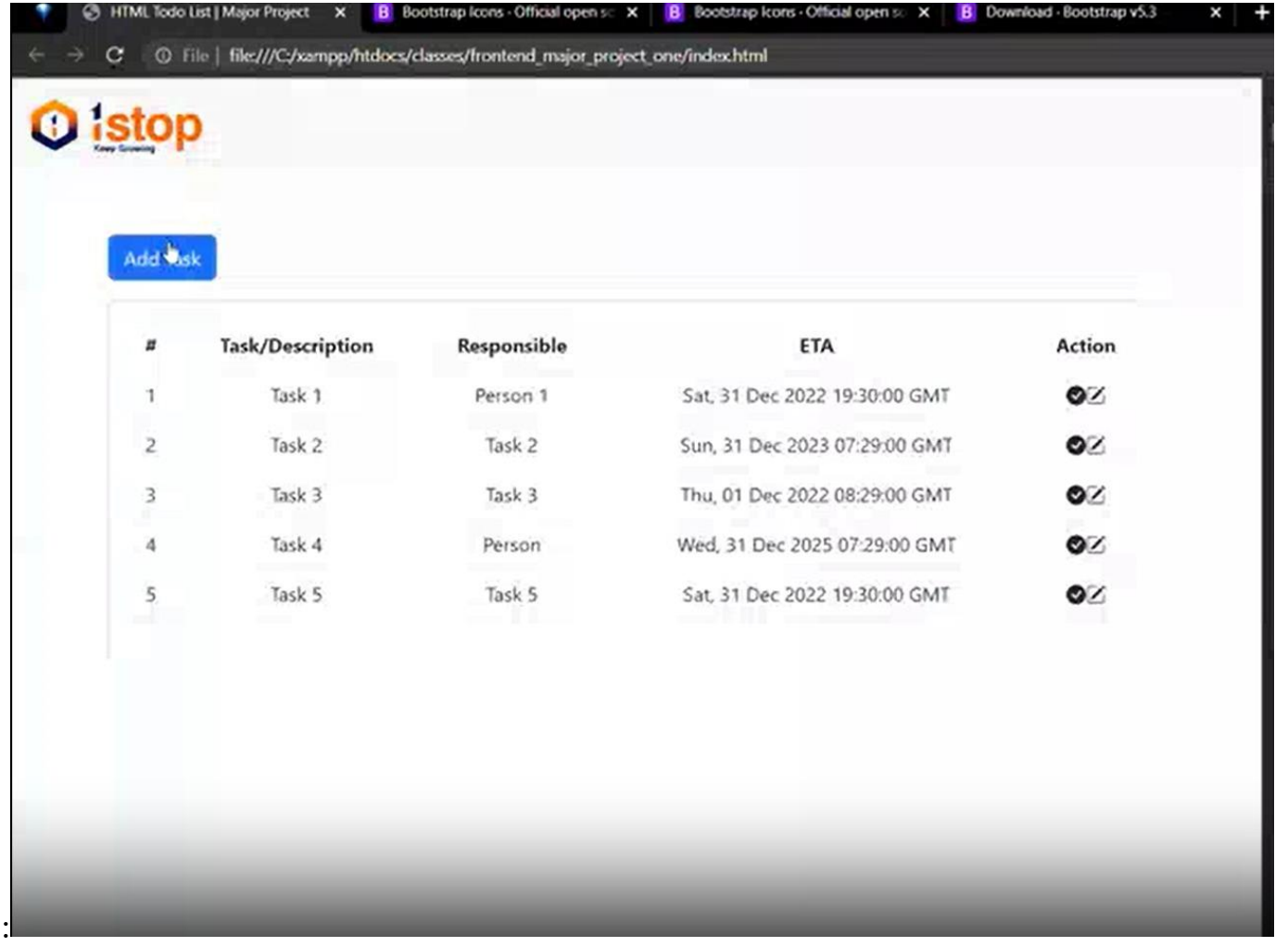
```
}
function updateTask(){
    $("#updateTaskModal").modal('hide')
    var dataArr=($("#taskUpdateForm").serializeArray())
    console.log(dataArr)
    var taskObject=new Object();
    var StorageObjectArr=[];
    var StorageObject=localStorage.getItem('taskStorage')
    for(var i in dataArr){
        var name=dataArr[i]['name']
        var name=dataArr[i]['value']
        taskObject[name]=value
    }
    if(StorageObject!=null && StorageObject!=undefined && StorageObject!=''){
        StorageObjectArr=JSON.parse(StorageObject)
```

```
            StorageObjectArr(taskObject['taskindex'])=taskObject
        }

        localStorage.setItem('taskStorage',JSON.stringify(StorageObjectArr))
        createHTMLfromStorage()


    }
  </script>
 </body>
</html>
```

OUTPUT:


HTML Todo List | Major Project
Bootstrap Icons · Official open s
Bootstrap Icons · Official open s
Download · Bootstrap v5.3
File | file:///C:/xampp/htdocs/classes/frontend_msjor_project_one/index.html
istop
Add Task
Task/Description Responsible ETA Action
1 Task 1 Person 1 Sat, 31 Dec 2022 19:30:00 GMT
2 Task 2 Task 2 Sun, 31 Dec 2023 07:29:00 GMT
3 Task 3 Task 3 Thu, 01 Dec 2022 08:29:00 GMT
4 Task 4 Person Wed, 31 Dec 2025 07:29:00 GMT
5 Task 5 Task 5 Sat, 31 Dec 2022 19:30:00 GMT

CONCLUSION:

The purpose and objective of create to do list website is achieved . Web page designing is easy and in an aesthetic form.

Flexibility in designing makes user explore their imagination.

The user can learn information about the particular website without having to navigate to other web pages ,which is easy to users.