**Bapuji Salunkhe Polytechnic, Kolhapur**

**Department Of Computer Science**

**Micro:**

Project Report

**Subject:**

Client Side Scripting

**Title:**

Parking System

**Submitted by:**

|  |  |  |
| --- | --- | --- |
| **ROLL NO.** | **STUDENT NAME** | **Enrollment No.** |
| 203369 | Kanishka Desai | 1800430342 |
| 203371 | Pranjal Haval | 1900430004 |
| 203374 | Adinath Sangaj | 1900430011 |

* **Introduction of micro project:**

Due to the proliferation in the number of vehicles on the road, traffic problems are bound to exist. This is due to the fact that the current transportation infrastructure and car park facility developed are unable to cope with the influx of vehicles on the road.

To alleviate the aforementioned problems, the smart parking system has been developed. With the implementation of the smart parking system, patrons can easily locate and secure a vacant parking space.

Vehicle Entry and Exit date is maintained here. So that issue cannot be occur in exit the vehicle.

Due to the rapidly increasing population the traffic congestion in the Philippines is also increasing rapidly, especially to the public places. The traffic congestions affects to those people who are working, studying and travelling just to go to their desired destinations. The causes of traffic may be classified into different problems. Example of this are big automobiles like passenger buses and Jeepneys, sidewalk vendors, under constructed roads and vehicles parked on the side of the road. But one of the most causes of traffic is improper parking.

* **Aim of micro project:**

1. Main aim of this project is to develop a software application to help the owner of vehicle.
2. Create Parking Reservation System.
3. Easy to Park your vehicle in less cost.
4. Car reservation completed display the message Car Successfully Completed.
5. Wrong Information can be added in reservation form.
6. Cannot possible to add the exit date is less than Entry date.

* **Intended course outcome:**

1. Create interactive web pages using program flow control structure.
2. Implemented Array and functions in Java Script.
3. Create event based web forms using Java Script.
4. Use JavaScript for handling cookies.
5. Create interactive webpage using regular expressions for validations.
6. Create Menus and navigations in web Pages.

* **literature Review**

TO build our project we took the from following links and books such as:

1: Client Side Scripting Language programs book we can refer and name of this book is technical publication.

2: We can visit the following websites

<http://www.1000projects.com>

In this website, I am collect various information about the client side scripting language programs. Car parking programs information available in this website & various techniques about the programming.

<http://www.w3schools.com>

In this website, different program available with related to client side scripting language programs.

3: In search different website we understood how to create our proposal or project.

* **Proposed methodology**

**Methods of project to improve it:**

This project is an automated parking system that used to reservation your car or any type of vehicle.

This will provide a safe parking zone where the drivers can leave the vehicles secured and guarded. The proposed study will help the community to control the flow of traffic due to illegal parking. This may lead to the commuters to become the cause of traffic congestion as well. This will also help the community to avoid minimal and substantial car accidents, carnapping crimes, and even becoming the reason of road arguments.

This proposed study will also help the enforcers to control and reduce the heavy traffic that is also caused by illegal parking of vehicles along highways and establishments.

This proposed study will help the researchers as a guide to improve and have further learning’s on some other related studies. This study can be used by private or public companies and establishments such as malls.

* **Problem:**
* How will the customers know if there is still available space in the parking lot if ever they want to reserve a slot?
* **Purpose:**
* The purpose of the project is to reduce the number of workers in the garage.
* reduce the prevalence of owners of cars

**Program Code:-**

**Index.html**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<link href="https://fonts.googleapis.com/css?family=Roboto&display=swap" rel="stylesheet">

<link rel="stylesheet" href="./CSS/bootstrap.min.css">

<link rel="stylesheet" href="./CSS/style.css">

<title>Parking Lot</title>

</head>

<body style="background-color:lightblue;">

<header class="shadow">

<div class="header-content d-flex justify-content-center p-2">

<imgsrc="./Images/parking.svg" alt="" id="header-logo">

<div id="header-msg" class="ml-5 align-self-center">Parking Lot</div>

</div>

</header>

<div class="form-container mt-5">

<form class="w-50 mx-auto" id="entryForm">

<h5 class="text-center">Add Car to Parking Lot</h5>

<div class="form-group">

<label for="owner">Owner:</label>

<input type="text" class="form-control rounded-0 shadow-sm" id="owner" placeholder="Owner">

</div>

<div class="form-group">

<label for="car">Car:</label>

<input type="text" class="form-control rounded-0 shadow-sm" id="car" placeholder="Car">

</div>

<div class="form-group">

<label for="licensePlate">License Plate:</label>

<input type="text" class="form-control rounded-0 shadow-sm" id="licensePlate" placeholder="LL NNNN">

</div>

<div class="row">

<div class="col-6">

<label for="entryDate">Entry Date:</label>

<input type="date" class="form-control rounded-0 shadow-sm" id="entryDate">

</div>

<div class="col-6">

<label for="exitDate">Exit Date:</label>

<input type="date" class="form-control rounded-0 shadow-sm" id="exitDate">

</div>

</div>

<button type="submit" class="btn mx-auto d-block mt-5 rounded-0 shadow" id="btnOne">Add Car</button>

</form>

</div>

<div class="table-container mt-5 mb-5 w-75 mx-auto" >

<h5 class="text-center mb-3">List of Cars in Parking Lot</h5>

<input type="text" class="w-100 mb-3" id="searchInput" placeholder="Search...">

<table class="table table-striped shadow " id="parkingTable">

<thead class="text-white" id="tableHead">

<tr>

<th scope="col">Owner</th>

<th scope="col">Car</th>

<th scope="col">License Plate</th>

<th scope="col">Entry Date</th>

<th scope="col">Exit Date</th>

<th scope="col">Actions</th>

</tr>

</thead>

<tbody id="tableBody">

</tbody>

</table>

</div>

<scriptsrc="./JS/bootstrap.min.js"></script>

<scriptsrc="./JS/core.js"></script>

</body>

</html>

**Core.js**

//Entry Class: Represent each entry in the parking lot

class Entry{

constructor(owner,car,licensePlate,entryDate,exitDate){

this.owner = owner;

this.car = car;

this.licensePlate = licensePlate;

this.entryDate = entryDate;

this.exitDate = exitDate;

}

}

//UI Class: Handle User Interface Tasks

class UI{

staticdisplayEntries(){

const entries = Store.getEntries();

entries.forEach((entry) =>UI.addEntryToTable(entry));

}

staticaddEntryToTable(entry){

consttableBody=document.querySelector('#tableBody');

const row = document.createElement('tr');

row.innerHTML = ` <td>${entry.owner}</td>

<td>${entry.car}</td>

<td>${entry.licensePlate}</td>

<td>${entry.entryDate}</td>

<td>${entry.exitDate}</td>

<td><button class="btnbtn-danger delete">X</button></td>

`;

tableBody.appendChild(row);

}

staticclearInput(){

//Selects all the inputs

const inputs = document.querySelectorAll('.form-control');

//Clear the content of each input

inputs.forEach((input)=>input.value="");

}

staticdeleteEntry(target){

if(target.classList.contains('delete')){

target.parentElement.parentElement.remove();

}

}

staticshowAlert(message,className){

const div = document.createElement('div');

div.className=`alert alert-${className} w-50 mx-auto`;

div.appendChild(document.createTextNode(message));

constformContainer = document.querySelector('.form-container');

const form = document.querySelector('#entryForm');

formContainer.insertBefore(div,form);

setTimeout(() =>document.querySelector('.alert').remove(),3000);

}

staticvalidateInputs(){

const owner = document.querySelector('#owner').value;

const car = document.querySelector('#car').value;

constlicensePlate = document.querySelector('#licensePlate').value;

constentryDate = document.querySelector('#entryDate').value;

constexitDate = document.querySelector('#exitDate').value;

varlicensePlateRegex = /^(?:[A-Z]{2}\d{2}\d{2})$/;

if(owner === '' || car === '' || licensePlate === '' || entryDate === '' || exitDate === ''){

UI.showAlert('All fields must me filled!','danger');

return false;

}

if(exitDate<entryDate){

UI.showAlert('Exit Date cannot be lower than Entry Date','danger');

return false;

}

if(!licensePlateRegex.test(licensePlate)){

UI.showAlert('License Plate must be like LL NNNN','danger');

return false;

}

return true;

}

}

//Store Class: Handle Local Storage

class Store{

staticgetEntries(){

let entries;

if(localStorage.getItem('entries') === null){

entries = [];

}

else{

entries = JSON.parse(localStorage.getItem('entries'));

}

return entries;

}

staticaddEntries(entry){

const entries = Store.getEntries();

entries.push(entry);

localStorage.setItem('entries', JSON.stringify(entries));

}

staticremoveEntries(licensePlate){

const entries = Store.getEntries();

entries.forEach((entry,index) => {

if(entry.licensePlate === licensePlate){

entries.splice(index, 1);

}

});

localStorage.setItem('entries', JSON.stringify(entries));

}

}

//Event Display

document.addEventListener('DOMContentLoaded',UI.displayEntries);

//Event Add

document.querySelector('#entryForm').addEventListener('submit',(e)=>{

e.preventDefault();

//Declare Variables

const owner = document.querySelector('#owner').value;

const car = document.querySelector('#car').value;

constlicensePlate = document.querySelector('#licensePlate').value;

constentryDate = document.querySelector('#entryDate').value;

constexitDate = document.querySelector('#exitDate').value;

if(!UI.validateInputs()){

return;

}

//Instatiate Entry

const entry = new Entry(owner, car, licensePlate, entryDate, exitDate);

//Add the entry do de UI table

UI.addEntryToTable(entry);

Store.addEntries(entry);

//Delete content of input's

UI.clearInput();

UI.showAlert('Car successfully added to the parking lot','success');

});

//Event Remove

document.querySelector('#tableBody').addEventListener('click',(e)=>{

//Call to UI function that removes entry from the table

UI.deleteEntry(e.target);

//Get license plate to use as unique element of an entry

varlicensePlate = e.target.parentElement.previousElementSibling.previousElementSibling.previousElementSibling.textContent;

//Call to Store function to remove entry from the local storage

Store.removeEntries(licensePlate);

//Show alert that entry was removed

UI.showAlert('Car successfully removed from the parking lot list','success');

})

//Event Search

document.querySelector('#searchInput').addEventListener('keyup', function searchTable(){

//Get value of the input search

constsearchValue = document.querySelector('#searchInput').value.toUpperCase();

//Get all lines of table body

consttableLine = (document.querySelector('#tableBody')).querySelectorAll('tr');

//for loop #1 (used to pass all the lines)

for(let i = 0; i <tableLine.length; i++){

var count = 0;

//Get all collumns of each line

constlineValues = tableLine[i].querySelectorAll('td');

//for loop #2 (used to pass all the collumns)

for(let j = 0; j <lineValues.length - 1; j++){

//Check if any collumn of the line starts with the input search string

if((lineValues[j].innerHTML.toUpperCase()).startsWith(searchValue)){

count++;

}

}

if(count > 0){

//If any collumn contains the search value the display block

tableLine[i].style.display = '';

}else{

//Else display none

tableLine[i].style.display = 'none';

}

}

});

**Style.css**

\*{

font-family: 'Roboto', sans-serif;

padding:0;

margin:0;

border:0

}

header{

background-color: rgb(25, 120, 197);

}

#header-logo{

width:64px;

height:64px;

}

#header-msg{

font-size: 25px;

color: white;

}

#btnOne{

background-color:rgb(25, 120, 197);

color:white;

}

#btnOne:focus,#btnOne:hover{

box-shadow: none;

outline:none;

}

#tableHead{

background-color:rgb(25, 120, 197);

}

#searchInput{

height: calc(1.5em + .75rem + 2px);

padding: .375rem .75rem;

border-bottom: 1px solid #ced4da;

}

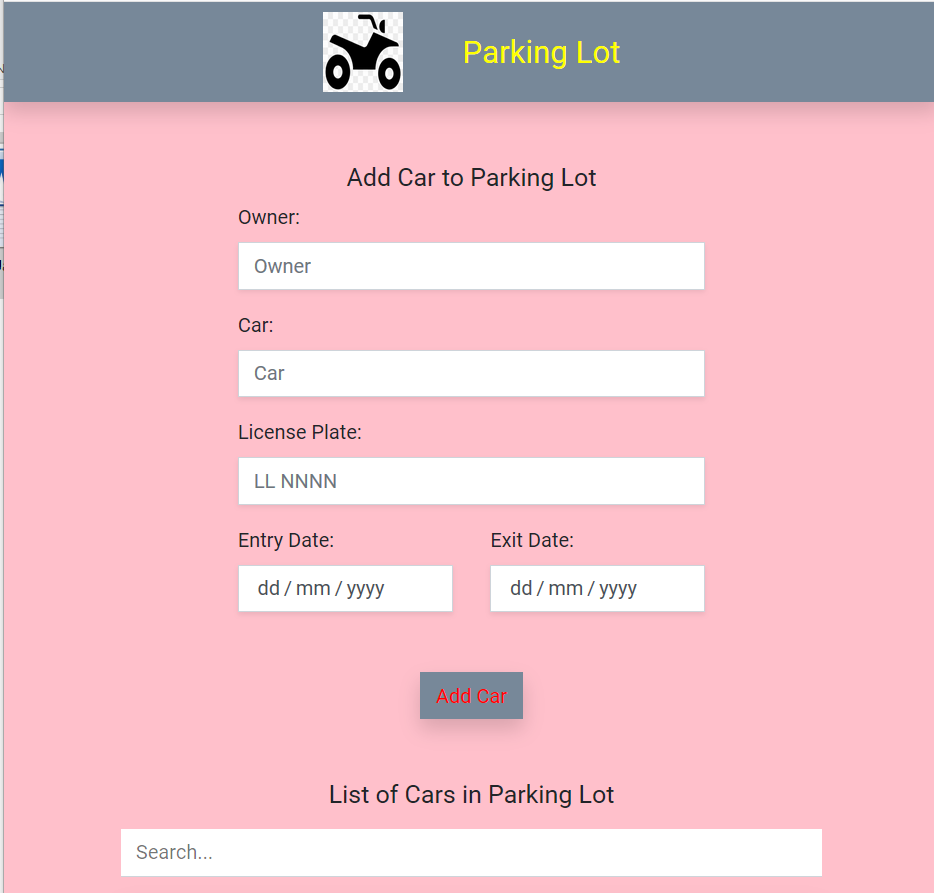
#searchInput:hover,#searchInput:focus{

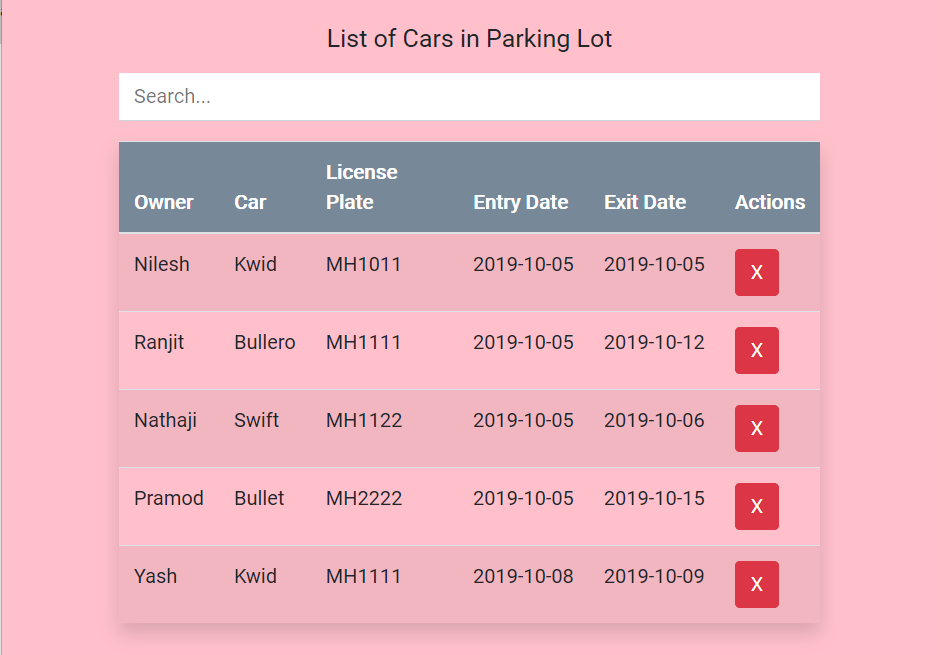
box-shadow: none;

outline:none;

}

**Output:-**

****

****

* **Resources Used:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr.no** | **Name of Resource/material** | **Specifications** | **Qty.** | **Remarks** |
| 1 | **Computer system** | HP Intel Core i3 7th Gen,  4GB Ram, 1TB HDD | 1 |  |
| 2 | **Software Used** | Google Chrome, MS world 2016, Eclipse Neon 3, Notepad, command prompt | 1 |  |
| 3 | **Computer** | IP Address | 1 |  |
| 4 | **Other Resources** | Printer | 1 |  |

**Skills Development :-**

1. Able to develop JavaScript to implement functions.

2. Able to develop JavaScript to implemen Form events.

3. protect webpage using JS.

**Application Of This Micro Project :-**

1) For parking system to maintain their alumnus database

2) To reduce server load by validating data on client machines.

3) To produce dynamic and responsive websites.