```
<!DOCTYPE html>

<html>

<head>

    <title>TRAFFIC CONTROL SYSTEM</title>

    <style>

        body {

            background-image: url('traffic_mngt.jpg');

            background-repeat: no-repeat;

            background-size: cover;

            background-attachment: fixed;

            margin: 0;

        }


        .header {

            padding: 15px;

            background-color: BLACK;

            text-align: center;

        }


        .header h3 {

            color: WHITE;

            font-size: 20px;

        }


        .nav {
```

```
    background-color: gray;

    padding: 5px;

}


.nav li {

    list-style: none;

    display: inline-block;

    padding: 8px;

}


.nav a {

    color: white;

}


.nav ul li a:hover {

    text-decoration: none;

    color: blue;

}


.lside {

    float: right;

    width: 50%;

    min-height: 500px;

    background: none; /* Remove background color */

    text-align: center;
    color: white; /* Set text color */

}


.rside {

    float: left;

    width: 50%;

    min-height: 500px;

    background: none; /* Remove background color */

    text-align: center;/* Add padding if needed */

    color: white; /* Set text color */

}


.footer {

    background-color: #455e64;

    text-align: center;

    padding: 10px;

}


.footer p {

    color: white;

}

marquee {

    background-color: transparent;

    border: none;

    padding: 0;
```

```
            margin: 0;
            float: bottom;
        }
    </style>
</head>
<body>
    <div>
        <div class="header">
            <h3><B>TRAFFIC CONTROL SYSTEM</B></h3>
            <p style="font-weight:bold;"></p>
        </div>

        <!-- Nav -->
        <div class="nav">
            <ul>
                <li><a href="#">Home</a></li>
                <li><a href="#">Services</a></li>
                <li><a href="ContactUs.html">Contact Us</a></li>
                <a href="user.html">Login</a>
            </ul>
        </div>

        <!-- main -->
        <div style="height:350px">
            <div class="lside">
        <marquee>
          <h2> TRAFFIC MANAGEMENT </h2>
          <h2> Traffic control system software</h2>
          <h2> has the capacity to plan and control </h2>
          <h2> traffic effectively providing users with </h2>
          <h2> the most comfortable and convenient</h2>
          <h2> experience to combat traffic congestion. </h2>

        </marquee>
      </div>
      <!-- side -->
      <div class="rside">
        <marquee>
          <h2> ROUTE MANAGEMENT</h2>
          <h2> It is a one-stop solution to effectively </h2>
          <h2> solve problems related to traffic congestion. </h2>
          <h2> this application is designed to meet all the </h2>
          <h2> needs of traffic control system while </h2>
          <h2> providing better quality and accuracy. </h2>

        </marquee>
      </div>
    </div>
  </div>
</body>
```

```
</html>


<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Full-Screen Background Color</title>
    <style>
        body {
            margin: 0;
            padding: 0;
            overflow: hidden;
            background-color: #c3edeb; /* Change to your desired background color */
            font-family: Arial, sans-serif;
        }

        /* Optional: Add more styles as needed for your content */
        .content {
            position: absolute;
            top: 50%;
            left: 50%;
            transform: translate(-50%, -50%);
            color: #fff;
            text-align: center;
        }
    </style>
</head>
<body>
    <div class="content">
        <!-- Your content goes here -->

    </div>
</body>
</html>
<head>

    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=2.0">
    <title>Login Page</title>
    <style>
        body {
            display: flex;
            align-items: center;
            justify-content: center;
            height: 100vh;
            margin: 0;
        }
        input {
            margin-bottom: 10px;
        }

        form {
            text-align: center;
        }


    </style>

</html>
    <form>
        <h2>Login</h2>
        <label for="username">Username:</label>
        <input type="text" id="username" name="username" required><br>

        <label for="password">Password:</label>
        <input type="password" id="password" name="password" required><br>

        <input type="submit" value="Login">
    </form>
</body>
</html>
    <script>
```

```
        // JavaScript function to automatically add a border when the page loads
        document.addEventListener("DOMContentLoaded", function() {
            var element = document.getElementById("content");
            if (element) {
                element.style.border = "2px solid #000"; // 2px solid black
            }
        });
    </script>
</head>
<body>
    <div id="content">
        <!-- Your content goes here -->
    </div>
</body>
</html>
```

User name and password

```
<!DOCTYPE html>
<html lang="en">
<head>
<style>
*{
    margin:0;
    padding:0;
    box-sizing: border-box;
    font-family: sans-serif;
}
body{
    display:flex;
    align-items: center;
    justify-content: center;
    height:100vh;
}
.container{
    max-width: 650px;
    padding: 28px;
    margin:0 28px;
    box-shadow: 0 15px 20px #ABB2B9;
}
h3{
    font-size: 15px;
    font-weight: 300px;
    text-align: left;
    color:black;
    padding-bottom: 8px;
    border-bottom: 1px solid silver;
}
.content{
    display: flex;
    flex-wrap: wrap;
    justify-content: 20px 0;
}
.input-box{
    display: flex;
    flex-wrap: wrap;
    width: 50%;
    padding-bottom: 15px;
}
.input-box:nth-child(2n){
    justify-content: end;
}
.input-box label, .gender-title{
    width: 95%;
    color:#2f4f4f;
    font-weight: bold;
    margin:5px 0;
}
.gender-title{
    font-size:16px;
}
.input-box input{
    height: 40px;
    width: 95%;
    padding:0 10px;
    border-radius: 5px;
    border: 1px solid#ccc;
    outline: none;
}
.input-box input:Is(:focus,:invalid){
    box-shadow: 0 3px 6px rgba(0,0,0,0.2);
}
.alert p{
    font-size: 14px;
    font-style: italic;
    color:dimgray;
    margin: 5px 0;
    padding: 10px;
}
.alert a{
    text-decoration: none;
}
.alert a:hover{
    text-decoration: underline;
}
.button-container{
    margin: 15px 0;
}
.button-container button {
    width: 100%;
    margin-top: 10px;
    padding: 10px;

    display: block;

    font-size: 20px;

    color: #fff;

    border: none;

    border-radius: 5px;

    background: linear-gradient(to right, #aa076b, #61045f);

}


.button-container button:hover {

    background: linear-gradient(to right, #61045f, #aa076b);

}




    @media(max-width:600px){

    .container{

    min-width: 280px;

    }

    .container{

    max-height: 380px;

    overflow:auto;

    }

    .input-box{

    margin-bottom: 12px;
    width:100%;

    }

    .input-box:nth-child(2n){

    justify-content: space-between;

    }

    .content::-webkit-scrollbar{

    width:0;}

    }


</style>

    <meta charset="UTF-8">

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

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

    <link rel="StyleSheet" href="StyleSheet2.css">

    <title>Document</title>

    </head>

    <body>

        <div class="conatiner">

            <form action="sample2.jsp" method="post">

                <h3>ROUTE MANAGEMENT</h3>

                <div class="content">

                    <div class="input-box">

                        <label for="Please Select">Select Route type*</label>

                        <input type="text" placeholder="Please Select"required>

                    </div>
                    <div class="input-box">
                        <label for="Please Select">Route allowed for Vehicle*</label>
                        <input type="text"required>
                    </div>
                    <div class="input-box">
                        <label for="Please Select">From Starting Place*</label>
                        <input type="text" name="source" required>
                    </div>
                    <div class="input-box">
                        <label for="Please Select">Total KM*</label>
                        <input type="text"required>
                    </div>
                    <div class="input-box">
                        <label for="Please Select">Route Description*</label>
                        <input type="text"required>
                    </div>
                    <div class="input-box">
                        <label for="Please Select">Route Name*</label>
                        <input type="text"required>
                    </div>
                    <div class="input-box">
                        <label for="Please Select">Ending Place*</label>
                        <input type="text" name="dest" required>
                    </div>
                </div>
```

```
        <div class="input-box">
          <label for="Please Select">Route Status*</label>
          <input type="text"required>
        </div>
          <div class="alert">
            <p>By clicking sign up,you agree to our<a href="#">Terms,</a><a href="#">Privacy policy</a> and<a href="#">cookies Policy.</a>
              You may receive SMS notification from us and can get opt at any time.</p>
          </div>
        <div class="button-container">
          <input type ="submit">Save Data</button>
        </div>
      </form>
    </div>
  </body>
</html>
```

A*alogrithm

```
%@ page import="java.util.*, java.util.stream.Collectors" %>

<%!
  class Node {
    String name;
    int xCoordinate;
    int yCoordinate;
    List<Edge> neighbors;

    public Node(String name, int xCoordinate, int yCoordinate) {
        this.name = name;
        this.xCoordinate = xCoordinate;
        this.yCoordinate = yCoordinate;
        this.neighbors = new ArrayList<>();
    }

    public void addNeighbor(Node neighbor, int distance) {
        this.neighbors.add(new Edge(neighbor, distance));
    }

    public int getXCoordinate() {
        return xCoordinate;
    }

    public int getYCoordinate() {
        return yCoordinate;
    }
}

class Edge {
    Node destination;
    int distance;
    public Edge(Node destination, int distance) {
        this.destination = destination;
        this.distance = distance;
    }
}

class Alg {
    public List<Node> findPath(Node start, Node goal) {
        Map<Node, Integer> gScore = new HashMap<>();
        Map<Node, Integer> fScore = new HashMap<>();
        Map<Node, Node> cameFrom = new HashMap<>();

        PriorityQueue<Node> openSet = new PriorityQueue<>(Comparator.comparingInt(fScore::get));
        Set<Node> closedSet = new HashSet<>();

        gScore.put(start, 0);
        fScore.put(start, heuristicCostEstimate(start, goal));
        openSet.add(start);

        while (!openSet.isEmpty()) {
            Node current = openSet.poll();

            if (current == goal) {
                return reconstructPath(cameFrom, goal);
            }

            closedSet.add(current);

            for (Edge neighborEdge : current.neighbors) {
                Node neighbor = neighborEdge.destination;

                if (closedSet.contains(neighbor)) {
                    continue;
                }

                int tentativeGScore = gScore.get(current) + neighborEdge.distance;

                if (!openSet.contains(neighbor) || tentativeGScore < gScore.get(neighbor)) {
                    cameFrom.put(neighbor, current);
                    gScore.put(neighbor, tentativeGScore);
                    fScore.put(neighbor, tentativeGScore + heuristicCostEstimate(neighbor, goal));

                    if (!openSet.contains(neighbor)) {
                        openSet.add(neighbor);
                    }
                }
            }
        }
```

```
    return Collections.emptyList();
}

private int heuristicCostEstimate(Node node, Node goal) {
    int x1 = node.getXCoordinate();
    int y1 = node.getYCoordinate();

    int x2 = goal.getXCoordinate();
    int y2 = goal.getYCoordinate();

    return (int) Math.sqrt(Math.pow(x2 - x1, 2) + Math.pow(y2 - y1, 2));
}

private List<Node> reconstructPath(Map<Node, Node> cameFrom, Node current) {
    List<Node> path = new ArrayList<>();
    path.add(current);

    while (cameFrom.containsKey(current)) {
        current = cameFrom.get(current);
        path.add(current);
    }

    Collections.reverse(path);
    return path;
}
```

```
    }

    private Node findNodeByName(String nodeName, Node... nodes) {
        for (Node node : nodes) {
            if (node.name.equalsIgnoreCase(nodeName)) {
                return node;
            }
        }
        return null;
    }
%>

<html lang="en">
<head>
    <!-- Your head content goes here -->
</head>
<body>

<%
    String sourceName = request.getParameter("source");
    String destinationName = request.getParameter("dest");

    Node a = new Node("A", 0, 0);
    Node b = new Node("B", 1, 1);
    Node c = new Node("C", 2, 2);
    Node d = new Node("D", 3, 3);
    Node e = new Node("E", 4, 4);

    a.addNeighbor(b, 1);
    b.addNeighbor(c, 3);
    c.addNeighbor(d, 2);
    d.addNeighbor(e, 4);

    Alg aStar = new Alg();
    Node sourceNode = findNodeByName(sourceName, a, b, c, d, e);
    Node destinationNode = findNodeByName(destinationName, a, b, c, d, e);

    if (sourceNode != null && destinationNode != null) {
        List<Node> path = aStar.findPath(sourceNode, destinationNode);

        if (!path.isEmpty()) {
            out.println("Optimal Path: " + path.stream().map(node -> node.name).collect(Collectors.joining("
-> ")));
        } else {
            out.println("No path found.");
        }
    } else {
        out.println("Invalid source or destination node name.");
    }
%>
<img src="MAPS.jpg">
```

```
</body>

</html>
```

Contact us page

```
<!DOCTYPE html>

<html lang="en">

<head>

<style>

body {

    font-family: Arial, sans-serif;

    margin: 0;

    padding: 0;

    display: flex;

    align-items: center;

    justify-content: center;

    height: 100vh;

    background-color: #f4f4f4;

}

.container {

    background-color: #fff;

    padding: 20px;

    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

    border-radius: 8px;

}
```

```css
h2 {
    text-align: center;
}
.contact-info,
.contact-form {
    max-width: 400px;
    margin: 0 auto;
    margin-top: 20px;
}
.contact-info p {
    margin-bottom: 10px;
}
label {
    display: block;
    margin-bottom: 8px;
}
input,
textarea {
    width: 100%;
    padding: 8px;
    margin-bottom: 16px;
    box-sizing: border-box;
}
button {
    background-color: #007BFF;
    color: #fff;
    padding: 10px 15px;
    border: none;
    border-radius: 4px;
    cursor: pointer;
}
button:hover {
    background-color: #0056b3;
}
</style>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="website.css">
    <title>Reach Us</title>
</head>
<body>

<div class="container">
    <h2>TRAFFIC CONTROL SYSTEM</h2>
    <h3>For further details contact us:</h3>

    <div class="contact-info">
        <p>Email: trafficcontrolsystem@gmail.com</p>
        <p>Phone number: +91 123-456-7890</p>
        <p>Telephone number: 080123345 </p>
```

```
  </div>

  <div class="contact-form">
    <form id="contactForm">
      <label for="name">Name:</label>
      <input type="text" id="name" name="name" required>

      <label for="email">Email:</label>
      <input type="email" id="email" name="email" required>

      <label for="message">Message:</label>
      <textarea id="message" name="message" rows="4" required></textarea>

      <button type="submit">Submit</button>
    </form>
  </div>
</div>
</body>
</html>
```