# Ministerul Educatiei si cercetarii al Republicii Moldova

Colegiul Universității tehnice A Republicii Moldovei

Specialitatea: „Programarea și Analiza Produselor Program”

**Raport**

Studiu Individual Nr.2

Tema:”Elaborarea unei aplicației de prelucrare a datei și orei”

Disciplina: „Asistență pentru programarea client-side a site-urilor WEB”

|  |
| --- |
| A Efectuat: |
| Baban Adrian |
| A verificat: |
| Alexandru Negară |

Chisinau 2025

**Sarcinile:**

**##1 studiu2.html.**

<!DOCTYPE html>

<html lang="ro">

<head>

    <meta charset="UTF-8">

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

    <title>Convertor Fusuri Orare</title>

    <link rel="stylesheet" href="styles.css">

    <script src="orase\_ecran.js"></script>

    <script src="conversii\_Date\_Ore.js"></script>

</head>

<body>

    <header>

        <h1>Convertor Fusuri Orare</h1>

    </header>

    <main>

        <div class="content">

            <div class="box">

                <h2>Convertire Ora intre 2 locatii</h2>

                <label for="inputTime">Introdu ora (HH:MM):</label>

                <input type="time" id="inputTime">

                <br>

                <label for="fromCity">Selectează orașul de pornire:</label>

                <select id="fromCity"></select>

                <br>

                <label for="toCity">Selectează orașul de destinație:</label>

                <select id="toCity"></select>

                <br>

                <button onclick="convertTime()">Convertește</button>

                <p id="outputTime"></p>

            </div>

            <div class="box">

                <h2>Convertire Data și Ora</h2>

                <label for="inputDateTime">Introdu data și ora:</label>

                <input type="datetime-local" id="inputDateTime">

                <br>

                <label for="fromCityDateTime">Selectează orașul de pornire:</label>

                <select id="fromCityDateTime"></select>

                <br>

                <label for="toCityDateTime">Selectează orașul de destinație:</label>

                <select id="toCityDateTime"></select>

                <br>

                <button onclick="convertDateTime()">Convertește Data și Ora</button>

                <p id="outputDateTime"></p>

            </div>

        </div>

    </main>

    <!-- Continuarea Fisierului, doar Footerul -->

    <footer>

      <div class="footer-container">

          <div class="footer-section">

              <h3>Legături rapide</h3>

              <ul>

                  <li><a href="../main.html">Acasă</a></li>

                  <li><a href="#">Despre</a></li>

                  <li><a href="#">Contact</a></li>

                  <li><a href="#">Servicii</a></li>

                  <li><a href="#">Politica de confidențialitate</a></li>

              </ul>

          </div>

          <div class="footer-section">

              <h3>Contact</h3>

              <p>Email: <a href="mailto:adry4work@gmail.com">adry4work@gmail.com</a></p>

              <p>Telefon: +373 784 00 000</p>

          </div>

          <div class="footer-section">

              <h3>Urmărește-ne</h3>

              <div class="social-icons">

                  <a href="https://www.facebook.com/Adry.baban"><img src="https://upload.wikimedia.org/wikipedia/commons/5/51/Facebook\_f\_logo\_%282019%29.svg" alt="Facebook"></a>

                  <a href="https://www.instagram.com/mr\_void26"><img src="https://upload.wikimedia.org/wikipedia/commons/9/95/Instagram\_logo\_2022.svg" alt="Instagram"></a>

                  <a href="https://github.com/MrVoid10"><img src="https://upload.wikimedia.org/wikipedia/commons/9/91/Octicons-mark-github.svg" alt="GitHub"></a>

              </div>

          </div>

      </div>

      <p class="footer-bottom">&copy; 2025 Pagina Responsivă - Toate drepturile rezervate</p>

    </footer>

    <script src="../extra\_cod/taste.js"></script>

</body>

</html>

**##2 orase\_ecran.js (locatia si Afisarea oraselor)**

const cities = [

  { "city": "Londra", "offset": 0 },

  { "city": "Chișinău", "offset": 2 },

  { "city": "București", "offset": 2 },

  { "city": "Paris", "offset": 1 },

  { "city": "Berlin", "offset": 1 },

  { "city": "Washington, D.C.", "offset": -5 },

  { "city": "Kathmandu", "offset": 5.75 },

  { "city": "Adelaide", "offset": 9.5 },

  { "city": "Beijing", "offset": 8 },

  { "city": "Moscova", "offset": 3 },

  { "city": "Tokyo", "offset": 9 },

  { "city": "New Delhi", "offset": 5.5 },

  { "city": "Ottawa", "offset": -5 },

  { "city": "Canberra", "offset": 10 },

  { "city": "Brasilia", "offset": -3 },

  { "city": "Riyadh", "offset": 3 },

  { "city": "Seoul", "offset": 9 },

  { "city": "Ankara", "offset": 3 },

  { "city": "Buenos Aires", "offset": -3 },

  { "city": "Pretoria", "offset": 2 },

  { "city": "Jakarta", "offset": 7 },

  { "city": "Mexico City", "offset": -6 },

  { "city": "Rome", "offset": 1 },

  { "city": "Madrid", "offset": 1 },

  { "city": "Bangkok", "offset": 7 },

  { "city": "Cairo", "offset": 2 },

  { "city": "Tehran", "offset": 3.5 },

  { "city": "Baghdad", "offset": 3 },

  { "city": "Nairobi", "offset": 3 },

  { "city": "Manila", "offset": 8 },

  { "city": "Hanoi", "offset": 7 }

];

// Continuarea fisierului

document.addEventListener('DOMContentLoaded', (event) => {

  const fromCitySelect = document.getElementById("fromCity");

  const toCitySelect = document.getElementById("toCity");

  const fromCityDateTimeSelect = document.getElementById("fromCityDateTime");

  const toCityDateTimeSelect = document.getElementById("toCityDateTime");

  cities.forEach(city => {

      const optionFrom = document.createElement("option");

      optionFrom.value = city.city;

      optionFrom.textContent = city.city;

      fromCitySelect.appendChild(optionFrom);

      const optionTo = document.createElement("option");

      optionTo.value = city.city;

      optionTo.textContent = city.city;

      toCitySelect.appendChild(optionTo);

      const optionFromDateTime = document.createElement("option");

      optionFromDateTime.value = city.city;

      optionFromDateTime.textContent = city.city;

      fromCityDateTimeSelect.appendChild(optionFromDateTime);

      const optionToDateTime = document.createElement("option");

      optionToDateTime.value = city.city;

      optionToDateTime.textContent = city.city;

      toCityDateTimeSelect.appendChild(optionToDateTime);

  });

  // Setează ora locală ca valoare implicită

  const now = new Date();

  const localHours = now.getHours().toString().padStart(2, '0');

  const localMinutes = now.getMinutes().toString().padStart(2, '0');

  document.getElementById("inputTime").value = `${localHours}:${localMinutes}`;

});

**##3 Conversii\_Date\_Ore.js (Logica principala)**

// functia ce converteste doar orele la doua locatii geografice

function convertTime() {

  const inputTime = document.getElementById("inputTime").value;

  const inputHour = parseInt(inputTime.split(":")[0]);

  const inputMinute = parseInt(inputTime.split(":")[1]);

  const fromCity = document.getElementById("fromCity").value;

  const toCity = document.getElementById("toCity").value;

  const fromOffset = cities.find(city => city.city === fromCity).offset;

  const toOffset = cities.find(city => city.city === toCity).offset;

  const totalMinutesOffsetFrom = fromOffset \* 60;

  const totalMinutesOffsetTo = toOffset \* 60;

  const currentTimeInFromCity = new Date();

  currentTimeInFromCity.setUTCHours(inputHour - fromOffset, inputMinute, 0, 0);

  const convertedTimeInToCity = new Date(currentTimeInFromCity.getTime() + totalMinutesOffsetTo \* 60 \* 1000);

  const hours = convertedTimeInToCity.getUTCHours().toString().padStart(2, '0');

  const minutes = convertedTimeInToCity.getUTCMinutes().toString().padStart(2, '0');

  const timeString = `${hours}:${minutes}`;

  document.getElementById("outputTime").textContent = `Ora în ${toCity}: ${timeString}`;

}

// Continuare fisier

// Functia care calculeaza data si ora la 2 locatii geografice

function convertDateTime() {

  const inputDateTime = document.getElementById("inputDateTime").value;

  const [inputDate, inputTime] = inputDateTime.split("T");

  const [year, month, day] = inputDate.split("-");

  const [hour, minute] = inputTime.split(":");

  const fromCity = document.getElementById("fromCityDateTime").value;

  const toCity = document.getElementById("toCityDateTime").value;

  const fromOffset = cities.find(city => city.city === fromCity).offset;

  const toOffset = cities.find(city => city.city === toCity).offset;

  const fromDateTime = new Date(Date.UTC(year, month - 1, day, hour - fromOffset, minute));

  const convertedDateTime = new Date(fromDateTime.getTime() + (toOffset) \* 60 \* 60 \* 1000);

  const convertedYear = convertedDateTime.getUTCFullYear();

  const convertedMonth = (convertedDateTime.getUTCMonth() + 1).toString().padStart(2, '0');

  const convertedDay = convertedDateTime.getUTCDate().toString().padStart(2, '0');

  const convertedHour = convertedDateTime.getUTCHours().toString().padStart(2, '0');

  const convertedMinute = convertedDateTime.getUTCMinutes().toString().padStart(2, '0');

  const convertedDateTimeString = `${convertedDay}-${convertedMonth}-${convertedYear} ${convertedHour}:${convertedMinute}`;

  document.getElementById("outputDateTime").textContent = `Data și ora în ${toCity}: ${convertedDateTimeString}`;

}

**##4 style.css (formatarea grafica a site-ului web)**

body {

  font-family: Arial, sans-serif;

  margin: 0;

  padding: 0;

  background-color: #121212;

  color: #e0e0e0;

}

header {

  background-color: #1f1f1f;

  padding: 20px;

  text-align: center;

  border-bottom: 2px solid #333;

}

h1 {

  margin: 0;

  font-size: 24px;

  display: inline-block;

  background: linear-gradient(90deg, #ff00ff, #ff4500, #ffd700, #00ff00, #00ffff, #0000ff, #800080);

  background-clip: text;

  -webkit-background-clip: text;

  -webkit-text-fill-color: transparent;

  background-size: 100% 100%;

}

@keyframes gradientAnimation {

  0% { background-position: 0% 50%; }

  50% { background-position: 100% 50%; }

  100% { background-position: 0% 50%; }

}

button {

  background: linear-gradient(90deg, #bb86fc, #03dac6);

  color: #121212;

  border: none;

  padding: 10px 15px;

  cursor: pointer;

  border-radius: 5px;

  font-weight: bold;

  transition: 0.3s ease;

}

button:hover {

  background: linear-gradient(90deg, #03dac6, #bb86fc);

}

main {

  padding: 20px;

  display: flex;

  justify-content: center;

  align-items: center;

}

.content {

  display: flex;

  flex-wrap: wrap;

  gap: 20px;

  max-width: 800px;

}

.box {

  background-color: #1f1f1f;

  padding: 15px;

  border-radius: 8px;

  border: 1px solid #333;

  width: 100%;

  text-align: center;

}

ul {

  list-style: none;

  padding: 0;

  margin: 0;

  width: auto;

  max-width: none;

}

footer {

  background: linear-gradient(90deg, #1f1f1f, #222, #1f1f1f);

  text-align: center;

  padding: 20px 0;

  margin-top: 30px;

  border-top: 2px solid #333;

  font-size: 14px;

}

.footer-container {

  display: flex;

  justify-content: space-around;

  flex-wrap: wrap;

  padding: 20px;

  max-width: 100%;

  margin: 0 auto;

}

.footer-section {

  flex: 1;

  display: inline-block;

  min-width: auto;

  margin: 10px;

}

.footer-section h3 {

  font-size: 18px;

  margin-bottom: 10px;

  color: #bb86fc;

}

.footer-section a {

  color: #e0e0e0;

  text-decoration: none;

  transition: color 0.3s ease;

}

.footer-section a:hover {

  color: #03dac6;

  text-decoration: underline;

}

.social-icons {

  display: flex;

  justify-content: center;

  gap: 15px;

  margin-top: 10px;

}

.social-icons img {

  width: 32px;

  height: 32px;

  transition: transform 0.3s ease, filter 0.3s ease;

  filter: brightness(0.8);

}

.social-icons img:hover {

  transform: scale(1.1);

  filter: brightness(1);

}

.footer-bottom {

  margin-top: 20px;

  font-size: 12px;

  color: #aaa;

  padding-top: 10px;

  border-top: 1px solid #444;

}

/\* iPhone-uri mici (SE, 5, 5S) \*/

@media only screen and (max-width: 375px) {

  body {background-color: #44ff44;}

}

/\* iPhone-uri medii (6, 7, 8, X, 11, 12 Mini) \*/

@media only screen and (max-width: 430px) and (max-height: 932px) {

  body {background-color: #00ff00;}

}

/\* iPhone-uri mari (Pro Max, 14 Plus) \*/

@media only screen and (min-width: 431px) and (max-width: 480px) {

  body {background-color: #cc0000;}

}

/\* iPad-uri (portret) \*/

@media only screen and (min-width: 768px) and (max-width: 1024px) and (orientation: portrait) {

  body {background-color: #ff9900;}

}

/\* iPad-uri (landscape) \*/

@media only screen and (min-width: 768px) and (max-width: 1366px) and (orientation: landscape) {

  body {background-color: #ffcc00;}

}

/\* Telefoane (portrait) - Android & iOS \*/

@media only screen and (max-width: 600px) and (orientation: portrait) {

  body {background-color: #ff0000;}

}

/\* Telefoane (landscape) - Android & iOS \*/

@media only screen and (max-width: 900px) and (orientation: landscape) {

  body {background-color: #ff66b2;}

}

/\* Tablete Android & iPad (portrait) \*/

@media only screen and (min-width: 601px) and (max-width: 900px) and (orientation: portrait) {

  body {background-color: #ff8800;}

}

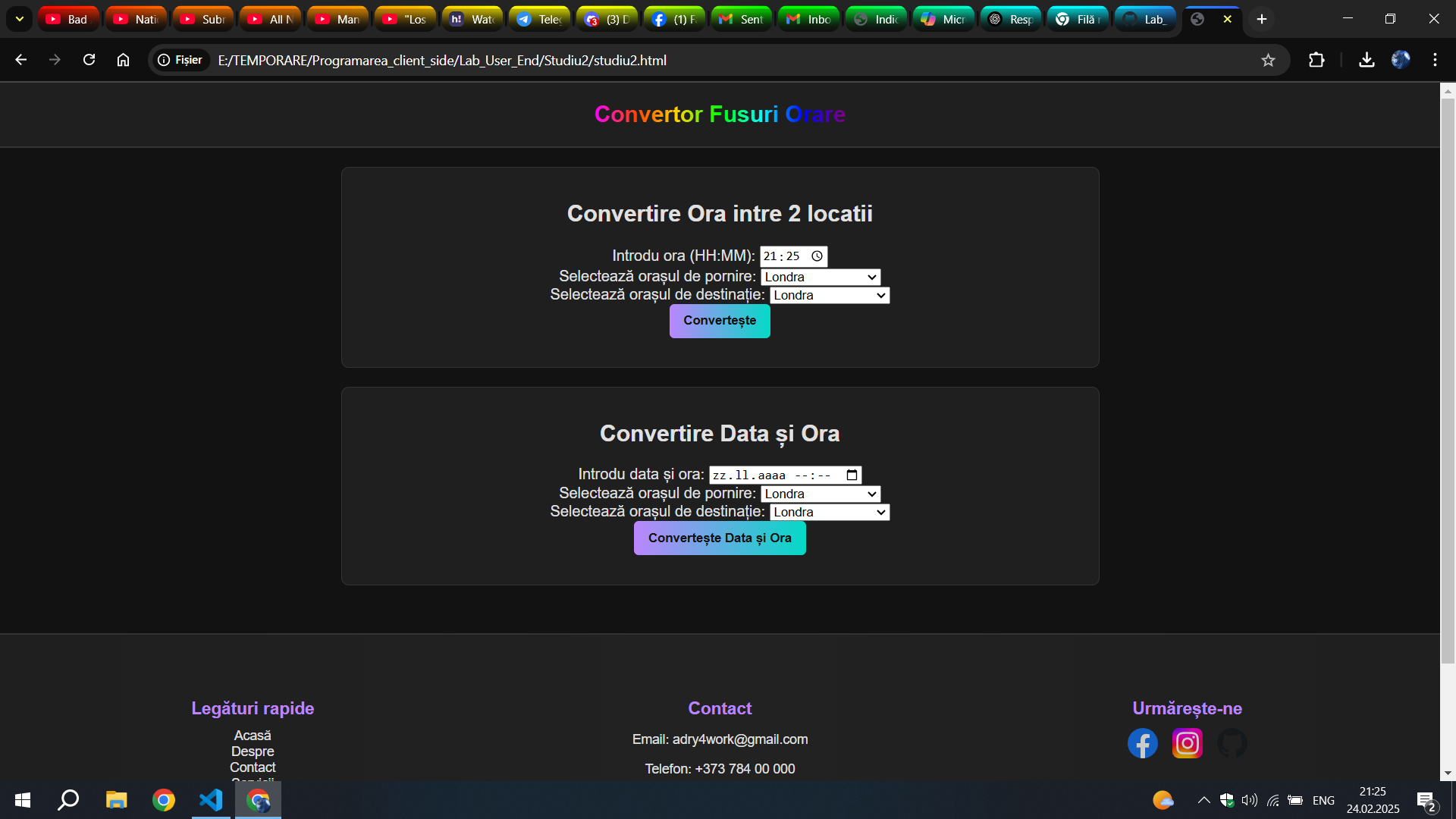
/\* Tablete Android & iPad (landscape) \*/

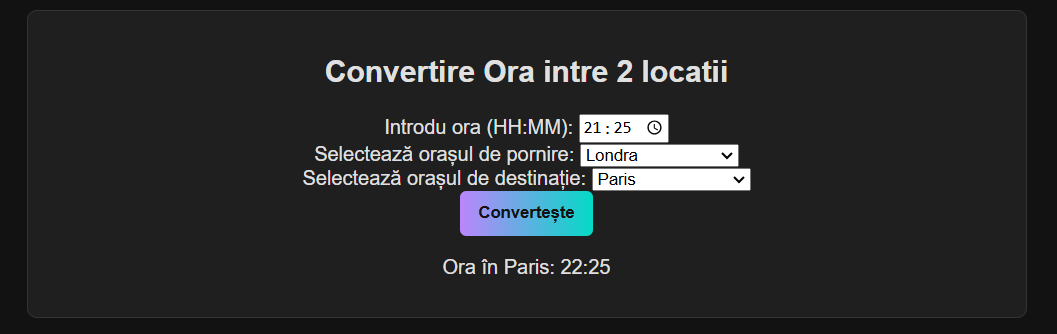
@media only screen and (min-width: 601px) and (max-width: 1200px) and (orientation: landscape) {

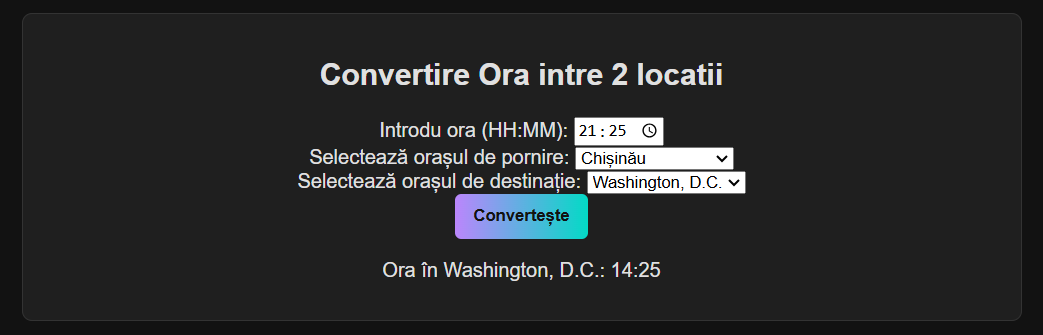
  body {background-color: #ffff00;}

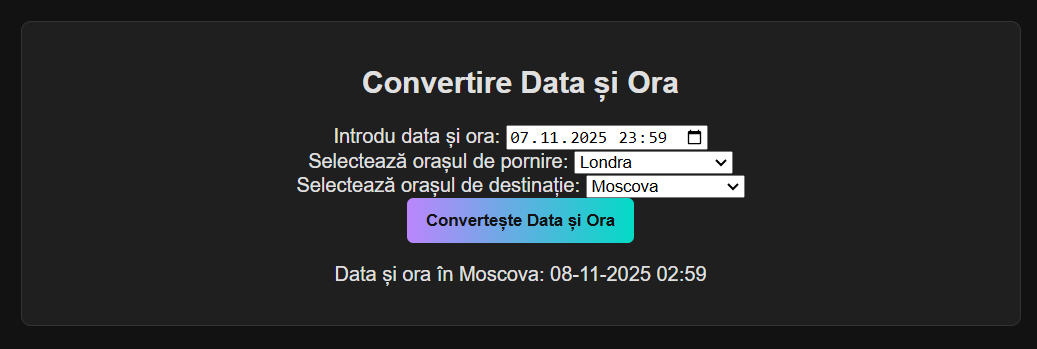
}

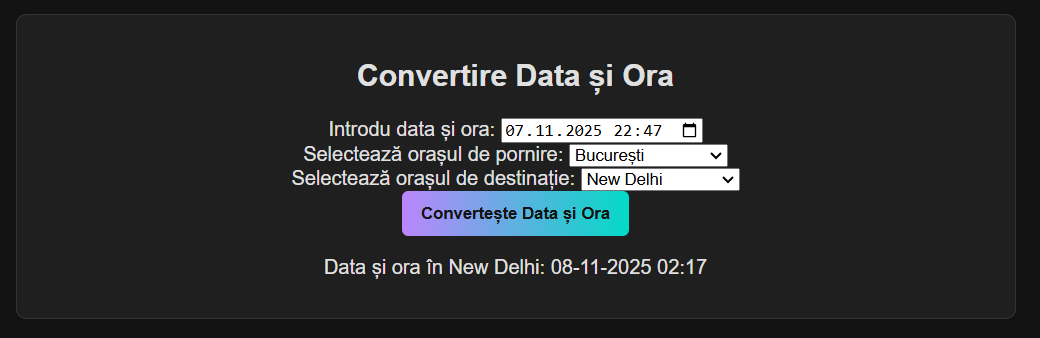
**##5 Navigarea unui utilizator.**



****

****

****

****

**Concluzie:**

La aceast Studiu Individual, am învățat să creez un site web care face diferite manipulari asupra formatului de date de timp.

Calculeaza diferentele dintre timpul unor regiuni in urmand fusul orar local.