

EXAMEN DE ATESTARE A COMPETENȚELOR PROFESIONALE

ALE ABSOLVENȚILOR CLASELOR DE MATEMATICĂ-
INFORMATICĂ ȘI MATEMATICĂ-INFORMATICĂ, INTENSIV
INFORMATICĂ



Tema: *Orașul Amara*

Profesor coordonator: *Alin NICOLESCU*

Candidat: *Livia Andreea IORGA*

Clasa: *a XII-a A*

Slobozia, mai 2025

CUPRINS

| | |
|---|-----------|
| I. ARGUMENT | 3 |
| II. SCOPUL LUCRĂRII | 3 |
| III. MOTIVAȚIA ALEGERII TEMEI | 3 |
| IV. IMPORTANȚA TEMEI | 3 |
| V. I. STRUCTURA SITE-ULUI | 4 |
| V.1. Organizarea fișierelor în foldere | 4 |
| V.2. Prezentarea resurselor site-ului | 4 |
| V.3. Prezentarea generală a paginii principale (index.html) | 5 |
| V.4. Harta site-ului | 6 |
| VI. DETALII DESPRE IMPLEMENTARE | 7 |
| VI.1. HTML | 7 |
| VI.2. CSS | 10 |
| VI.3. JavaScript | 12 |
| VI.4. Utilizarea Notepad++ | 15 |
| VI.5. Utilizarea browserului (Google Chrome) | 15 |
| VII. VIZUALIZARE ÎN BROWSER | 16 |
| VIII. BIBLIOGRAFIE | 20 |

I. ARGUMENT

Lucrarea de față are ca scop realizarea unui site web de prezentare a orașului Amara, un punct important pe harta turismului balnear din România. Prin intermediul acestui proiect, am dorit să creez un site informativ, cu o interfață prietenoasă, care să ofere utilizatorilor informații detaliate despre oraș, stațiunea balneară, evenimentele locale, posibilitățile de cazare, dar și o galerie ilustrativă.

Am ales această temă deoarece combină interesul personal pentru geografie și turism cu pasiunea pentru dezvoltare web. Realizarea site-ului mi-a oferit ocazia de a-mi aprofunda cunoștințele în HTML și CSS, dar și de a înțelege cum se structurează și organizează un proiect web real.

Într-o eră digitală în care vizibilitatea online este esențială, promovarea orașelor mai mici precum Amara contribuie la dezvoltarea turismului local și la valorificarea resurselor naturale. Acest site poate reprezenta un model pentru digitalizarea informațiilor despre comunitățile locale.

II. SCOPUL LUCRĂRII

Principalul scop al lucrării este de a realiza un site de prezentare atractiv, bine structurat și ușor de utilizat, care să ofere o imagine completă a orașului Amara și a potențialului său turistic. Proiectul urmărește să:

- prezinte într-un mod clar și organizat informațiile esențiale despre oraș și stațiune;
- utilizeze principii de design responsiv pentru a fi accesibil;
- dezvolte abilități de lucru cu HTML, CSS și organizarea fișierelor într-un proiect web.

III. MOTIVAȚIA ALEGERII TEMEI

Amara este un oraș adesea subestimat, dar cu un potențial turistic important datorită lacului sărat și a nămolului sapropelic. Alegerea temei a fost motivată de dorința de a contribui, fie și într-un mod digital, la promovarea acestei localități. De asemenea, tema mi-a permis să exersez dezvoltarea unei interfețe web reale și funcționale, folosind tehnologii de bază precum HTML și CSS.

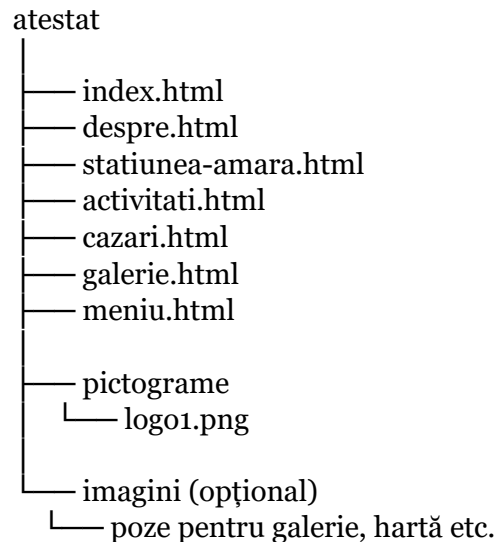
IV. IMPORTANȚA TEMEI

Această temă este importantă din două perspective: una practică și una educațională. Practic, proiectul poate servi ca punct de pornire pentru un site oficial sau turistic dedicat orașului. Educațional, lucrul la acest site mi-a oferit o experiență aplicată de lucru în front-end development, esențială pentru dezvoltarea în domeniul IT.

V. STRUCTURA SITE-ULUI

V.1. Organizarea fișierelor în foldere

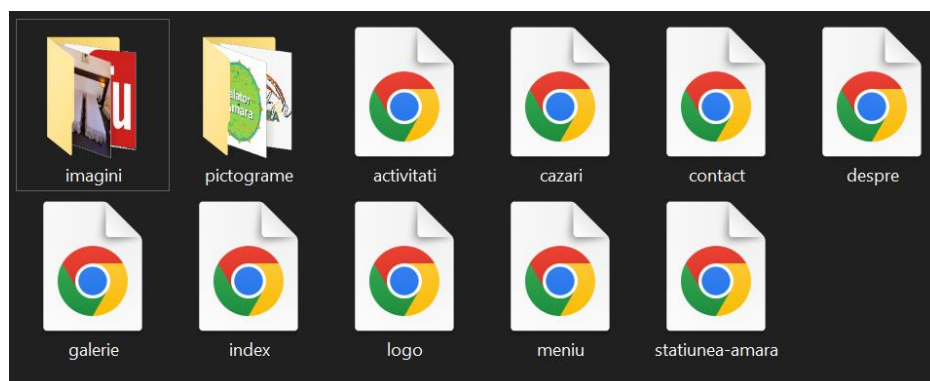
Site-ul a fost structurat într-un mod logic și eficient pentru a permite gestionarea facilă a resurselor. Structura folderelor este următoarea:



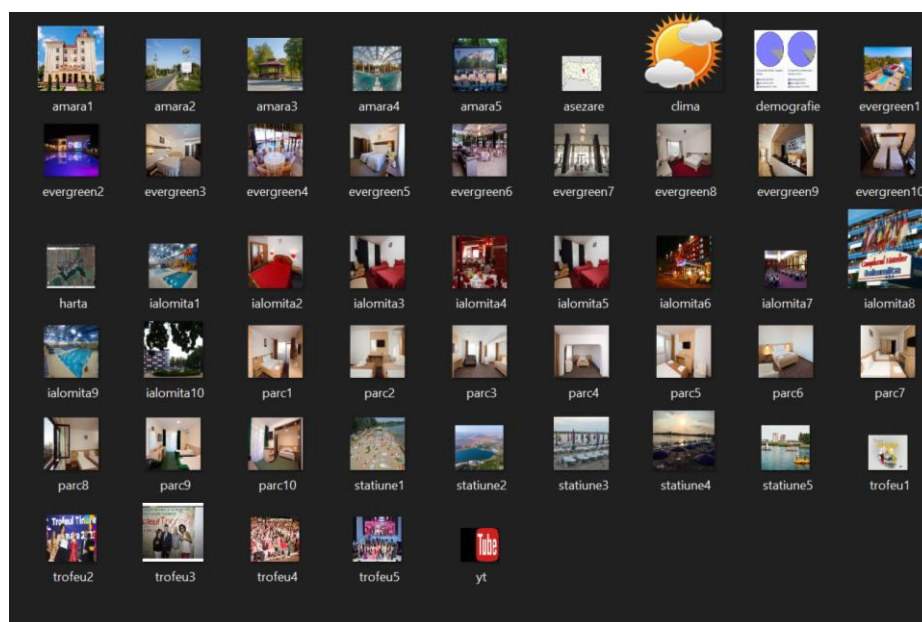
V.2. Prezentarea resurselor site-ului

Pentru acest site au fost utilizate următoarele resurse:

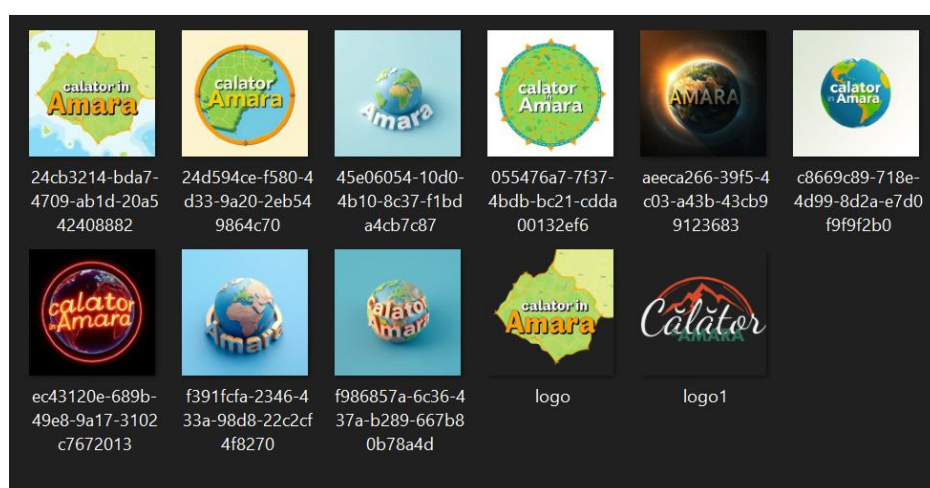
- Font implicit: Arial, din motive de lizibilitate și simplitate.
- Imagini: logo-ul site-ului este încărcat din folderul pictograme, restul imaginilor sunt adăugate din folderul imagini.
- CSS: Fără framework-uri externe
- Iframe-uri: folosite pentru afișarea conținutului în pagină fără reîncărcarea întregului site (prin *target="main-frame"*).
- *Foldere:*



(folder: atestat)



(folder: imagini)



(folder: pictograme)

V.3. Prezentarea generală a paginii principale (index.html)

Pagina principală este structurată astfel încât să ofere o interfață organizată, utilizând limbajul HTML și stilizarea CSS (internă). Aceasta servește drept "hub" pentru toate celelalte pagini din site, prin intermediul unui meniu de navigare și a unui iframe care afișează conținutul selectat, fără a încărca din nou întreaga pagină.

1. Structura generală

- Folosește doctype HTML5 (`<!DOCTYPE html>`) și are limba setată pe română (`<html lang="ro">`).
- În `<head>` sunt definite:
 - Setarea charset UTF-8

- Titlul paginii: „**Orașul Amara**”
- Stiluri CSS interne pentru navigație și layout general

2. Navigația (header)

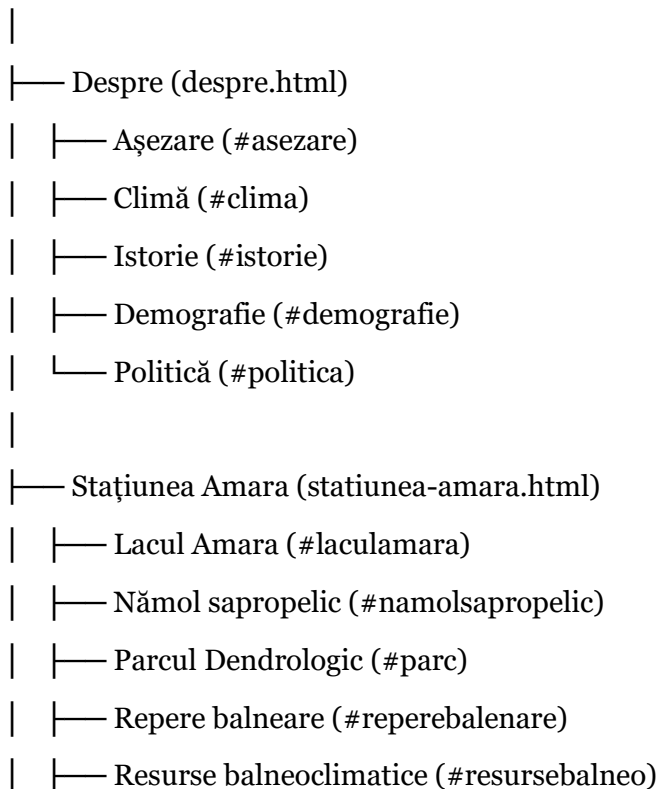
- Este împărțită în trei componente:
 - **Logo-ul** (în stânga) — imagine cu link către fișierul original (pictograme/logo1.png)
 - **Meniul principal** (centru) — listă *ul.menu* cu secțiuni și subsecțiuni pentru:
 - Despre
 - Stațiunea Amara
 - Activități și evenimente
 - Cazări
 - Galerie
 - **Meniul secundar** (dreapta) — buton „Meniu” cu legături rapide către aceleași secțiuni + “Contactează-ne!”

3. Iframe-ul

- Afișează conținutul selectat din meniul **fără a părăsi pagina principală**
- Implicit, este încărcată pagina *despre.html*

V.4. Harta Site-ului

Orașul Amara (index.html)



| | |
|--|---|
| | └─ Bază turistică (#bazaturistica) |
| | |
| | └─ Activități și evenimente (activitati.html) |
| | └─ Hidrobiciclete (#hidrobiciclete) |
| | └─ Paddle Boards (#paddleb) |
| | └─ Trofeul Tinereții (#trofeu) |
| | └─ Proceduri (#proceduri) |
| | └─ Tabăra Pavilioane (#pavilioane) |
| | |
| | └─ Cazări (cazari.html) |
| | └─ Complex Hotelier Parc (#parc) |
| | └─ Hotel Ialomița (#ialomita) |
| | └─ Hotel Evergreen (#evergreen) |
| | |
| | └─ Galerie (galerie.html) |
| | └─ Stațiune (#statiune2) |
| | └─ Orașul Amara (#oamara) |
| | └─ Trofeul Tinereții (#trofeul) |
| | |
| | └─ Contactează-ne! (contact.html) |

VI. DETALII DESPRE IMPLEMENTARE

VI.1. HTML

HTML (HyperText Markup Language) este limbajul de bază utilizat pentru crearea structurii paginilor web. În cadrul proiectului dedicat orașului Amara, HTML a fost utilizat pentru a organiza conținutul într-o formă accesibilă, structurată și ușor de navigat.

Limbajul HTML a fost utilizat pentru a construi structura principală a site-ului web dedicat orașului Amara. Codul HTML are o arhitectură clară și este organizat în două mari secțiuni: `<head>` și `<body>`.

Documentul HTML este construit conform standardelor moderne HTML5 și cuprinde următoarele componente:

Structura generală a paginii

Documentul începe cu declarația `<!DOCTYPE html>` care specifică utilizarea standardului HTML5. Atributul `lang="ro"` din eticheta `<html>` indică faptul că pagina este scrisă în limba română.

Secțiunea `<head>` include:

- setarea setului de caractere UTF-8 (`<meta charset="UTF-8">`);
- titlul paginii (`<title>Orașul Amara</title>`);
- stiluri CSS interne, definite în cadrul etichetei `<style>`, care controlează aspectul vizual al elementelor.

Structura paginii vizibile

În interiorul etichetei `<head>` sunt incluse metadate esențiale:

1. `<meta charset="UTF-8">` setează codificarea caracterelor la UTF-8, pentru a permite afișarea corectă a caracterelor internaționale.
2. `<meta name="viewport" content="width=device-width, initial-scale=1.0">` asigură și compatibilitatea cu dispozitivele mobile, setând lățimea paginii la dimensiunea ecranului.
3. `<title>` oferă titlul paginii care va apărea în bara browserului.

Secțiunea `<body>` conține conținutul principal al site-ului, organizat astfel:

1. Header-ul și bara de navigare (`<header class="navbar">`), care include:
 - Logo-ul site-ului (un link cu o imagine);
 - Meniul principal de navigare cu submeniuri organizate ierarhic (liste `` și ``);
 - Un meniu suplimentar poziționat în partea dreaptă (meniu de acces rapid).
2. Navigația *principală*
 Meniurile sunt create cu liste neordonate (`<ul class="menu">`). Fiecare item de meniu principal conține:
 - un link (``) care încarcă o anumită pagină în cadrul iframe-ului principal;
 - un submeniu (`<ul class="submenu">`) care apare la interacțiunea cu utilizatorul (hover).

```
<li><a href="despre.html" target="main-frame">Despre</a>
  <ul class="submenu">
    <li><a href="despre.html#asezare" target="main-frame">Așezare</a></li>
    ...
  </ul>
</li>
```

3. Iframe pentru afișarea conținutului (`<iframe name="main-frame">`)
 Toate paginile secundare se încarcă în cadrul unui iframe, ceea ce permite utilizatorului să navigheze fără reîncărcarea întregului site. Acest iframe este încapsulat într-un `<div class="iframe-container">`, iar pagina implicit încărcată este `despre.html`.

```
<iframe name="main-frame" src="despre.html"></iframe>
```


4. Meniul din dreapta (<div class="right-menu">)

Include un buton „Meniu” care deschide un submeniu cu legături rapide către toate secțiunile majore ale site-ului, inclusiv o pagină de contact.

5. Harta

```
<iframe
src="https://www.google.com/maps/embed?pb=!1m14!1m8!1m3!1d11351.
287693805095!2d27.326918!3d44.61745!3m2!1i1024!2i768!4f13.1!3m3!1
m2!1sox40b7c273658ad75f%3A0xd72d2a0ff786cd9b!2sAmara%2C%20Ju
de%C8%9Bul%20Ialomi%C8%9Ba!5e0!3m2!1sro!2sro!4v1715082000000
!5m2!1sro!2sro"
width="100%" height="400" style="border:0;" allowfullscreen=""
loading="lazy"
referrerpolicy="no-referrer-when-downgrade">
</iframe>
```

6. Link către YouTube

```
<a href="https://youtu.be/R8fMM73VrXs?si=NuhLcAkdILYwtsJi"
target="_blank" style="display: block; text-align: center; margin-top:
20px;">

<p style="margin-top: 10px; font-weight: bold; color: #cc0000;">▶ Vezi
videoclipul pe YouTube</p>
</a>
```

7. Formular de contact

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

<label for="prenume">Prename:</label>
<input type="text" id="prenume" name="prenume" required>

<label for="telefon">Număr de telefon:</label>
<input type="tel" id="telefon" name="telefon" pattern="[0-9]{10}"
placeholder="07XXXXXXXX" required>

<label for="email">E-mail:</label>
<input type="email" id="email" name="email" required>

<label for="mesaj">Sugestii sau mesaje despre Stațiunea
Amara:</label>
<textarea id="mesaj" name="mesaj" placeholder="Scrie aici
sugestia ta..." required></textarea>

<button type="submit">Trimite mesajul</button>
</form>

<div class="success-message" id="successMessage">
<p>Mesajul a fost trimis cu succes!</p>
<button onclick="resetForm()">Trimite o altă solicitare</button>
```

```
</div>
</div>
```

8. Formular de rezervare

```
<section id="booking">
  <h2>Rezervare pentru grupuri mari</h2>
  <p style="text-align: center;">Completează formularul pentru a primi o ofertă personalizată.</p>
  <form id="bookingForm">
    <label for="hotel">Alege hotelul:</label>
    <select id="hotel" name="hotel" required>
      <option value="">-- Selectează --</option>
      <option value="Parc">Complex Hotelier Parc</option>
      <option value="Ialomita">Complex Hotelier Ialomita</option>
      <option value="Evergreen">Hotel Restaurant Evergreen</option>
    </select>
    <label for="roomType">Tipul de cameră:</label>
    <select id="roomType" name="roomType" required>
      <option value="">-- Selectează --</option>
      <option value="Single">Single</option>
      <option value="Dublă Standard">Dublă Standard</option>
      <option value="Dublă Superioară">Dublă Superioară</option>
      <option value="Apartament">Apartament</option>
    </select>
    <label for="checkin">Check-in:</label>
    <input type="datetime-local" id="checkin" name="checkin" required>
    <label for="checkout">Check-out:</label>
    <input type="datetime-local" id="checkout" name="checkout" required>
    <label for="guests">Număr estimativ de persoane:</label>
    <input type="number" id="guests" name="guests" min="1" required>
    <label for="contactName">Nume:</label>
    <input type="text" id="contactName" name="contactName" required>
    <label for="email">E-mail:</label>
    <input type="email" id="email" name="email" required>
    <label for="telefon">Telefon:</label>
    <input type="tel" id="telefon" name="telefon" pattern="[0-9]{10}" placeholder="07XXXXXXX" required>
    <label for="mesaj">Mesaj opțional:</label>
    <textarea id="mesaj" name="mesaj" placeholder="Scrie aici orice detalii adiționale..."></textarea>
    <button type="submit">Trimite solicitarea</button>
  </form>
  <div class="success-message" id="bookingSuccess" style="display:none;">
    <p style="color: green; font-weight: bold;">Solicitarea ta a fost trimisă! Te vom contacta în cel mai scurt timp cu o ofertă personalizată.</p>
    <button onclick="resetBookingForm()" style="background-color: #FF6F00; color: white; padding: 10px 20px; border: none; border-radius: 8px;">Trimite altă solicitare</button>
  </div>
</section>
```

Observații

- Toate legăturile din meniu folosesc atributul *target="main-frame"* pentru a încărca paginile în iframe.
- Site-ul este structurat astfel încât să poată fi extins ușor, cu organizare logică.
- HTML-ul este completat de CSS intern care se ocupă de stilizarea elementelor și de afișarea dinamică a submeniurilor.

VI.2. CSS

CSS (Cascading Style Sheets) este limbajul folosit pentru a defini aspectul vizual și stilul unei pagini HTML. Dacă HTML descrie structura și conținutul paginii, CSS controlează modul în care acel conținut arată: unde este plasat pe pagină, ce culori are, ce fonturi folosește, cât spațiu îl înconjoară și cum reacționează la interacțiunea cu utilizatorul (ex: hover, click etc.).

Ce înseamnă „Cascading” în CSS?

Cuvântul „cascading” vine din engleză și înseamnă „în cascadă”, referindu-se la modul în care stilurile se aplică:

- CSS poate veni din mai multe surse: stiluri implicite ale browserului, stiluri scrise de noi, stiluri externe etc.

- Când mai multe reguli CSS afectează același element, prioritatea este determinată de ordine, specificitate și importanță.
- De exemplu, o regulă mai specifică (cum ar fi `.navbar a`) va învinge una generală (ca `a`).

Unde poate fi scris CSS-ul?

CSS poate fi adăugat în 3 moduri:

1. **Inline** – direct în tag-ul HTML (ex: `<p style="color:red">`)
2. **Intern** – într-un tag `<style>` în interiorul `<head>`
3. **Extern** – într-un fișier separat `.css`, conectat prin `<link>` în `<head>`

CSS funcționează prin reguli formate din două părți principale: selectorul și declarațiile. Selectorul stabilește ce element HTML va fi stilizat, iar declarațiile definesc ce stiluri vor fi aplicate acelui element. Declarațiile sunt compuse dintr-o proprietate (cum ar fi *color*, *font-size*, *margin* etc.) și o valoare asociată. De exemplu, dacă dorim ca tot textul din pagină să fie gri închis, putem selecta elementul *body* și să-i atribuim proprietatea *color* cu o anumită valoare de culoare.

În CSS, există o ierarhie de aplicare a stilurilor. Dacă mai multe reguli afectează același element, va fi aplicată regula mai specifică sau ultima definită. Acest mecanism se numește „cascadare” și este o caracteristică esențială a limbajului. De asemenea, stilurile CSS pot fi aplicate în moduri diferite: direct în tag-ul HTML (inline), într-un bloc `<style>` în interiorul fișierului HTML (intern), sau dintr-un fișier extern `.css`.

Un concept modern extrem de important în CSS este Flexbox, prescurtare pentru Flexible Box Layout. Acesta este un mod de a organiza elementele pe axele unei pagini, fie pe orizontală (rând), fie pe verticală (coloană). Cu ajutorul Flexbox, dezvoltatorii pot controla alinierea, spațierea și dimensiunea elementelor dintr-un container, chiar și atunci când acestea au dimensiuni variabile. De exemplu, într-un meniu de navigație orizontal, putem folosi Flexbox pentru a centra butoanele și a lăsa spațiu egal între ele, indiferent de lățimea ecranului.

Pentru a activa Flexbox, se aplică proprietatea *display: flex* pe containerul care conține elementele. Odată activat, putem folosi alte proprietăți precum *justify-content* (pentru alinierea pe axa orizontală), *align-items* (pentru alinierea pe axa verticală) și *gap* (pentru a controla distanța dintre elemente). Acest sistem este folosit adesea în paginile web moderne, inclusiv în proiectul de față, pentru organizarea barei de navigație și a conținutului din pagină.

Explicarea codului:

Prima regulă aplică stiluri universale tuturor elementelor (*). Se elimină spațiile implicite prin *margin: 0* și *padding: 0*, iar cu *box-sizing: border-box*, se stabilește că dimensiunile elementelor includ și marginile și bordura, ceea ce face ca layout-ul să fie mai previzibil.

Secțiunea *body* definește stilul general al paginii: se setează un font modern, de tip sans-serif (*Arial, sans-serif*), un fundal deschis la culoare (`#f1f1f1`) și culoarea implicită a textului (`#333`, un gri închis). Pagina este organizată cu *display: flex* și *flex-direction: column*, ceea ce înseamnă că elementele copil (precum bara de navigație și conținutul) se vor așeza unul sub altul, pe verticală. De asemenea, se setează *height: 100vh* pentru a ocupa întregul ecran, și *overflow-x: hidden* pentru a preveni apariția barei orizontale de scroll.

Clasa `.navbar` este folosită pentru bara de navigație. Se activează Flexbox pentru a alinia elementele din meniu (`display: flex`, `align-items: center`, `justify-content: space-between`), se aplică un fundal cu gradient (de la turcoaz la verde închis), se setează o înălțime fixă de 80px și padding interior.

Clasa `.logo` are `flex-shrink: 0`, ceea ce înseamnă că acest element (de obicei imaginea cu logo-ul) nu va fi redimensionat dacă spațiul din meniu devine mai mic.

Containerul `.menu-container` conține lista de navigație și este setat cu `flex-grow: 1`, adică va ocupa tot spațiul disponibil între logo și partea dreaptă a barei. Lista `.menu` este afișată orizontal (`display: flex`) și spațiată uniform cu `gap: 30px`.

Fiecare element de meniu (`.menu li`) este poziționat relativ pentru a putea afișa submeniuri. Link-urile (`.menu li a`) sunt stilizate cu text alb, fără subliniere, cu padding interior și colțuri rotunjite. De asemenea, există un efect vizual la hover, care schimbă fundalul în turcoaz deschis.

Clasa `.submenu` definește un meniu derulant ascuns inițial (`display: none`). Acesta apare sub fiecare item de meniu, fiind poziționat absolut sub elementul părinte. Fundalul galben-portocaliu și umbrele adăugate (`box-shadow`) creează un efect vizual de adâncime. Link-urile din submenu devin portocalii la hover pentru a evidenția selecția.

Prin `.menu li:hover .submenu`, meniul derulant este afișat la interacțiunea utilizatorului cu elementul principal.

Există o variantă de meniu pentru partea dreaptă a barei, denumită `.right-menu`. Aceasta are `flex-shrink: 0` pentru a nu fi comprimată. Submeniurile din această zonă sunt ajustate cu poziționare la dreapta (`right: 0`) în loc de stânga, pentru a nu ieși din ecran, și o lățime maximă mai mică (`max-width: 200px`).

Secțiunea `.iframe-container` conține un `iframe` care ocupă tot spațiul disponibil rămas din pagină (`flex-grow: 1`). Acest lucru permite încărcarea de pagini externe în partea principală a site-ului, păstrând bara de navigație vizibilă. Iar `iframe`-ul este setat să umple complet containerul, fără margini (`border: none`), asigurând o integrare elegantă.

VI.3. JavaScript

JavaScript este limbajul de programare esențial care adaugă interactivitate și dinamism unei pagini web. Spre deosebire de HTML, care definește structura conținutului, și CSS, care controlează stilul vizual, JavaScript permite utilizatorilor să interacționeze activ cu pagina, să primească feedback instant și să experimenteze o navigare fluidă, fără reîncărcarea completă a site-ului.

Pe un site modern, JavaScript este folosit pentru o varietate de funcționalități: validarea formularelor, animații, meniuri derulante, galerii de imagini (slidere), afișarea sau ascunderea unor secțiuni în funcție de acțiunile utilizatorului, manipularea dinamică a conținutului, comunicarea cu serverul (prin AJAX sau fetch), și multe altele.

JavaScript acționează așadar ca motorul logicii din spatele unei pagini web: determină „ce se întâmplă” atunci când un utilizator interacționează cu un

element. Este un limbaj orientat pe evenimente – codul este declanșat de acțiuni precum click, submit, scroll sau hover.

1. Gestionarea formularului de rezervare (cazari.html)

Codul identifică elementul cu `id="bookingForm"` (formularul) și elementul cu `id="bookingSuccess"` (mesajul de confirmare). Când utilizatorul trimite formularul, evenimentul submit este interceptat, iar comportamentul implicit (trimiterea datelor către server) este prevenit cu `e.preventDefault()`. În schimb, formularul este ascuns și se afișează un mesaj de succes, ceea ce oferă un feedback instant utilizatorului. Funcția `resetBookingForm()` este folosită pentru a reseta starea formularului: reafișează formularul și ascunde mesajul de succes. Astfel, interacțiunea utilizatorului este fluidă și controlată complet din browser.

```
const bookingForm = document.getElementById('bookingForm');
const bookingSuccess = document.getElementById('bookingSuccess');

bookingForm.addEventListener('submit', function (e) {
  e.preventDefault();
  bookingForm.style.display = 'none';
  bookingSuccess.style.display = 'block';
});

function resetBookingForm() {
  bookingForm.reset();
  bookingForm.style.display = 'flex';
  bookingSuccess.style.display = 'none';
}
```

2. Funcționalitatea sliderului de imagini (cazari.html)

Funcția `changeSlide(sectionId, direction)` primește două argumente: `sectionId`, care este identificatorul secțiunii în care se află sliderul, și `direction`, care indică dacă vrem să mergem la următoarea imagine (valoare pozitivă) sau la cea anterioară (valoare negativă). Sliderul identifică imaginea curentă și numărul total de imagini, apoi modifică poziția prin transformare CSS (`translateX`) în funcție de indexul imaginii. Butoanele de navigare cu clasele `.prev` și `.next` sunt legate de această funcție prin evenimente click, permițând navigarea stânga-dreapta în cadrul galeriei.

```
function changeSlide(sectionId, direction) {
  const slider = document.getElementById(sectionId + '-slider');
  const slides = slider.querySelectorAll('img');
  let currentIndex = parseInt(slider.getAttribute('data-current-slide')) || 0;
  const totalSlides = slides.length;
  currentIndex = (currentIndex + direction + totalSlides) % totalSlides;
  slider.setAttribute('data-current-slide', currentIndex);
  const slideWidth = slides[0].clientWidth;
  slider.style.transform = `translateX(-${currentIndex * slideWidth}px)`;
}
```

3. Funcționalitatea sliderului de imagini (galerie.html)

În a doua variantă de slider, este utilizat un *obiect sliders* care mapează mai multe secțiuni (precum stațiune, amara, trofeu) la câte un slider propriu. Fiecare slider are un index care determină imaginea afișată curent. Când se apelează *changeSlide(name, direction)*, funcția actualizează indexul corespunzător și aplică transformarea CSS doar pe sliderul vizat. Această abordare permite gestionarea mai clară a mai multor galerii în paralel, fără a duce la suprapuneri sau conflicte.

```
const sliders = {
  statiune: { index: 0, id: "statiune-slider" },
  amara: { index: 0, id: "amara-slider" },
  trofeu: { index: 0, id: "trofeu-slider" }
};

function changeSlide(name, direction) {
  const slider = sliders[name];
  const container = document.getElementById(slider.id);
  const total = container.children.length;

  slider.index += direction;
  if (slider.index < 0) slider.index = total - 1;
  if (slider.index >= total) slider.index = 0;

  container.style.transform = `translateX(-${slider.index * 100}%)`;
}
```

4. Gestionarea unui formular de contact (contact.html)

Logica este similară cu cea a formularului de rezervare: când utilizatorul trimite formularul, codul previne trimiterea reală și înlocuiește formularul cu un mesaj de confirmare. Funcția *resetForm()* resetează câmpurile și vizibilitatea elementelor, permițând utilizatorului să reîncerce dacă este necesar.

```
const form = document.getElementById('contactForm');
const successMessage = document.getElementById('successMessage');

form.addEventListener('submit', function (e) {
  e.preventDefault(); // prevenim trimiterea reală
  form.style.display = 'none';
  successMessage.style.display = 'block';
});

function resetForm() {
  form.reset();
  form.style.display = 'block';
  successMessage.style.display = 'none';
}
```

VI.4. Utilizarea Notepad++

Notepad++ este un editor de text avansat, gratuit, utilizat pe scară largă pentru scrierea și editarea codului sursă. Spre deosebire de Notepad-ul clasic din Windows, Notepad++ oferă o serie de funcționalități utile pentru dezvoltarea web:

- Evidențierea sintaxei (syntax highlighting) pentru HTML, CSS, JavaScript și multe alte limbaje – ceea ce face codul mai ușor de citit și înțeles.
- Indentarea automată și colorarea codului – ajută la structurarea clară a codului, mai ales în fișiere mari.
- Numere de linie vizibile – utile pentru depanare și orientare în cod.
- Deschiderea simultană a mai multor fișiere în tab-uri – ideal atunci când lucrezi la un proiect care conține fișiere separate HTML, CSS și JS.
- Căutare avansată și înlocuire – funcții rapide pentru corectarea sau modificarea unor bucăți de cod în tot documentul sau în mai multe fișiere odată.
- Salvarea în diverse formate și fără formatare ascunsă – important pentru păstrarea compatibilității cu browserul.


În cadrul proiectului, Notepad++ a fost folosit pentru a scrie și organiza codul HTML, CSS și JavaScript într-un mod clar și eficient. Fișierele au fost salvate local cu extensiile corespunzătoare (.html pentru structură)

VI.5. Utilizarea browserului (Google Chrome)

Pentru testarea și rularea aplicației web, a fost utilizat Google Chrome, unul dintre cele mai populare și performante browsere moderne. Chrome este compatibil cu cele mai noi standarde web și oferă un mediu ideal pentru dezvoltatori:

- Randare rapidă a paginilor HTML și stilurilor CSS.
- Compatibilitate extinsă cu JavaScript, inclusiv cu cele mai recente funcționalități ECMAScript.
- DevTools (instrumente pentru dezvoltatori) – extrem de utile pentru:
 - examinarea structurii HTML în timp real;
 - testarea și modificarea stilurilor CSS;
 - vizualizarea consolei JavaScript pentru erori sau mesaje;
 - monitorizarea evenimentelor și a rețelei (network).
- Responsive Design Mode – permite simularea paginii pe diferite dimensiuni de ecran (telefon, tabletă, desktop), esențial pentru verificarea comportamentului responsive.

VII. VIZUALIZARE ÎN BROWSER




DespreStațiunea AmaraActivități și evenimenteCazăriGalerieMeniu

Despre

Amara este un oraș în județul Ialomița, Muntenia, România, format din localitatea componentă Amara (reședința), și din satul Amara Nouă. Se află în partea centrală a Câmpiei Bărăganului, la 7 km nord-vest de municipiul Slobozia din același județ. Include în arealul său administrativ Stațiunea balneoclimaterică Amara cu caracter permanent, aflată la 2 km de orașul cu același nume.


Așezare




Orașul se află în partea centrală a județului Ialomița, între comunele Grivița (Nord), Gheorghe Doja și Perieți (Vest) și municipiul Slobozia (Est și Sud). Pe plan local altitudinea variază între 23–44 m Stațiunea se află la o altitudine de 30 mpe malul lacului Amara

Climă

Climatul este continental de stepă, cu veri călduroase (temperatura medie 22-23 °C) și diferențe mari de la zi la noapte. Iernile sunt relativ reci, vânturile sunt predominante iarna dinspre nord-est, caracteristic fiind Crivățul. Temperatura medie anuală este de 10,5-11 °C precipitațiile medii însumează 450–500 mm/an. Nebulozitatea este redusă în timpul verii, cu o medie de peste 125 zile senine în timpul anului.





DespreStațiunea AmaraActivități și evenimenteCazăriGalerieMeniu

Stațiunea Amara

Lacul Amara


Lacul Amara este un lac natural - fost liman fluviatil barat cu aluviuni, situat într-o depresiune (Crivaia) - veche matcă a râului Ialomița - care nu mai are actual legătură cu râul. Se încadrează în partea estică a unității structurale Platforma Moesică.

Este alimentat cu ape de șiroire (precipitații) care spală eflorescențele produse la suprafața rocilor din Câmpia Română și de către apele subterane (freatic) încărcate de săruri (predominant sulfat și cloruri). Sulfatul de magneziu și sulfatul de sodiu dizolvate în mare cantitate în apa lacului, îi dau gustul foarte amar care maschează gustul sau sărat. Prin lipsa alimentării constante cu apă dulce tot timpul anului și prin procesul de evaporare pe fondul climatului uscat, a crescut concentrația în săruri a apei.

Lacul a prezentat o tendință de creștere a nivelului începând cu perioada 1965-1966. Valori de peste 4 m adâncime au fost atinse în anii 1970-1971; pentru a împiedica inundarea terenurilor și a construcțiilor din apropiere a fost construit un canal pentru evacuarea apei în exces. Această măsură a avut o mare influență negativă asupra salinității apei: dacă în 1887 concentrația de săruri a apei era de 89,45 g/l, în 1976 aceasta ajunsese la 7,7 g/l, ceea ce a favorizat dezvoltarea unei faune specifice apelor dulci. Concomitent s-a produs și reducerea proprietăților curative ale apei și nămolului, precum și scăderea rezervei de nămol.


Namol Sapropelic

Este negru (în contact cu aerul devenind cenușiu), unsuros, sărat și are un miros foarte puternic de hidrogen sulfurat. El conține 41% săruri anorganice, 39% substanțe organice și 20% apă. Substanțele pe care le conține sunt: sulfat de sodiu, calciu, fier sau magneziu; clorură de sodiu; hiposulfid de sodiu; hidrogen sulfurat; hidrocarburosulfid; carbonat de sodiu sau de calciu; sulf liber; acetat de sodiu; nitrat de amoniu; acid formic, glutamic butiric, propilic, clorofil-rezorcin, pirocatechin și erin. Grosimea stratului de nămol este de 30–60 cm.



Despre
Stațiunea Amara
Activități și evenimente
Cazări
Galerie
Meniu

hidrotierapie, terapie fizică și kinetoterapie. Toate au plaje proprii. Lebăda: categoria doua stele, capacitate de 944 locuri repartizate în două corpuri de cazare, acționari sunt marile confederații sindicale. Ialomița: categoria trei stele, capacitate de 480 locuri, aflat în proprietatea Ministerului Muncii și Protecției Sociale. A beneficiat de investiții, în special după 1998. Complexul balnear Parc: este format din 2 hoteluri, de trei stele (Hotel Parc) cu 432 de locuri, respectiv patru stele (Hotel Dana) cu 78 locuri, capital integral privat. Este complet renovat. Vila Irina: capacitate 56 de locuri, aflată în administrarea Ministerului Justiției prin Administrația Națională a Penitenciarelor. Tabăra de copii Pavilioane, este amenajată din 1975, fiind situată într-o pădure de salcâm, plantație de foioase și livada de nuci. Are o capacitate de 110 locuri iar în curte se află o bază sportivă. Numeroase alte unități de cazare - moteluri, vile - completează oferta turistică. Există și un camping, cu căsuțe și loc pentru corturi. Plaje: Hoteliere: Lebăda, Splendid (Hotel Ialomița), Mircești (Complexul Parc) Plajă pentru copii - amenajată în cadrul Taberei de copii Plajă (în regim privat) pentru nudiști - amenajată cu secțiuni separate pentru femei și bărbați (Hotel Parc) Obiective turistice Locale Parcul dendrologic (care se întinde de-a lungul a 150 ha în lungul lacului Amara) și Herghelia Jegalia. De vecinătate Municipiul Slobozia Muzeul Județean (colecții de arheologie, artă plastică, istorie, artă populară) Muzeul Agriculturii Centrul Cultural „Ionel Perlea” Biserica “Sfinții Volevozi” (fostă mănăstire construită în 1628) Monumentul Eroilor din Primul Război Mondial Arealul Înconjurător: Lunca Ialomiței, Pădurea Suduți, Crângul Frumos, Pădurea Buiești (rezervații cinegetice). Comuna Balaciu: Mănăstirea Balaciu - ridicată în 1750 de familia boierilor Piteșteanu Comuna Ciulnița: Biserica din lemn “Sfântul Nicolae” Comuna Ion Roată: Muzeul sătesc de etnografie Comuna Ograda: Casa Memorială “Ionel Perlea” Comuna Sinești: Conacul Marghiloman – construcție realizată între anii 1869 – 1874



[▶ Vezi videoclipul pe YouTube](#)



Despre
Stațiunea Amara
Activități și evenimente
Cazări
Galerie
Meniu

Activități și evenimente

Hidrobiciclete
Paddle Boards
Trofeul Tineretii
Proceduri
Tabara "Pavilioane"

Hidrobiciclete

Pe lacul Amara, turiștii pot închiria hidrobiciclete pentru a petrece mor în special în timpul sezonului cald. Acestea sunt disponibile în zonele amenajate de pe mal, cum ar fi plaja Splendid sau pontoane în cadrul Complexului Hotelier Ialomița. Activitatea este potrivită pentru familii și grupuri de prieteni, fiind o modalitate accesibilă și plăcută de a vă bucura de frumetea lacului într-un ritm lent și sigur.


Paddle Boards

Pasionații de activități nautice mai dinamice pot opta pentru paddle board (stand-up paddle), o formă de relaxare și exercițiu în același timp. Paddle board-urile pot fi închiriate de la „Senin pe Ape”, un furnizor local specializat în activități pe apă. Aceștia oferă echipamente moderne și, în unele cazuri, cursuri introductive pentru începători. Paddle board-ul este ideal pentru cei care doresc să combine mișcarea cu peisajul calm al lacului Amara într-o atmosferă liniștită și naturală.

Trofeul Tineretii

Trofeul Tineretii de la Amara este un festival de muzică ușoară românească, ce s-a impus ca fiind al doilea de gen din țară după Festivalul Național de Muzică Ușoară de la Mamaia.

Acesta se desfășoară în luna august la grădina de vară din orașul Amara. În trecut a mai avut loc și în Municipiul Slobozia. Este singurul care aparține genului în România, și care a ajuns la 49 de ediții fără a fi întrerupt.





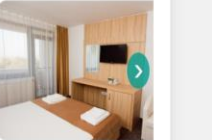


[Despre](#)
[Stațiunea Amara](#)
[Activități și evenimente](#)
[Cazări](#)
[Galerie](#)
[Meniu](#)

Cazări


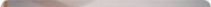



Complex Hotelier Parc


Situat în inima stațiunii Amara, Complexul Hotelier Parc oferă 104 camere duble standard, 6 camere duble superioare și 2 apartamente dotate cu jacuzzi sau cadă cu vibromasaj. Camerele sunt echipate cu aer condiționat, baie privată, mobilier modern și Wi-Fi gratuit. Complexul dispune de o bază de tratament, restaurant, trei săli de mese, sală de conferințe, bar, două terase, parcare păzită și plajă proprie amenajată cu terase, dușuri, șezlonguri, tobogan acvatic și bazin pentru copii. Plaja privată se află la doar câțiva pași de hotel, oferind acces la nămol terapeutic și apele sărate ale lacului Amara. Pentru mai multe informații și rezervări, vizitați site-ul oficial: amara-parc.ro.

Complex Hotelier Ialomita

Clasificat la categoria de trei stele, Complexul Hotelier Ialomita este situat pe malul nord-vestic al lacului Amara. Oferă 23 de camere single, 222 de camere duble și 7 apartamente, toate dotate cu aer condiționat, minibar, telefon, TV LCD și baie privată. Unele camere au balcon. Facilitățile includ un restaurant cu 240 de locuri, bar, grădină de vară cu 200 de locuri, salon de evenimente Amethyst, două săli de conferințe (Vienna și Dublin) echipate modern, bază de tratament, plajă proprie, centru SPA și cabinet de estetică medicală. Pentru detalii suplimentare și rezervări, accesați: hotelialomita.ro.



[Despre](#)
[Stațiunea Amara](#)
[Activități și evenimente](#)
[Cazări](#)
[Galerie](#)
[Meniu](#)

Rezervare pentru grupuri mari

Completează formularul pentru a primi o ofertă personalizată.

Alege hotelul:

-- Selectează --

Tipul de cameră:

-- Selectează --

Check-in:

zz . 11 . aaaa -- : --

Check-out:

zz . 11 . aaaa -- : --

Număr estimativ de persoane:

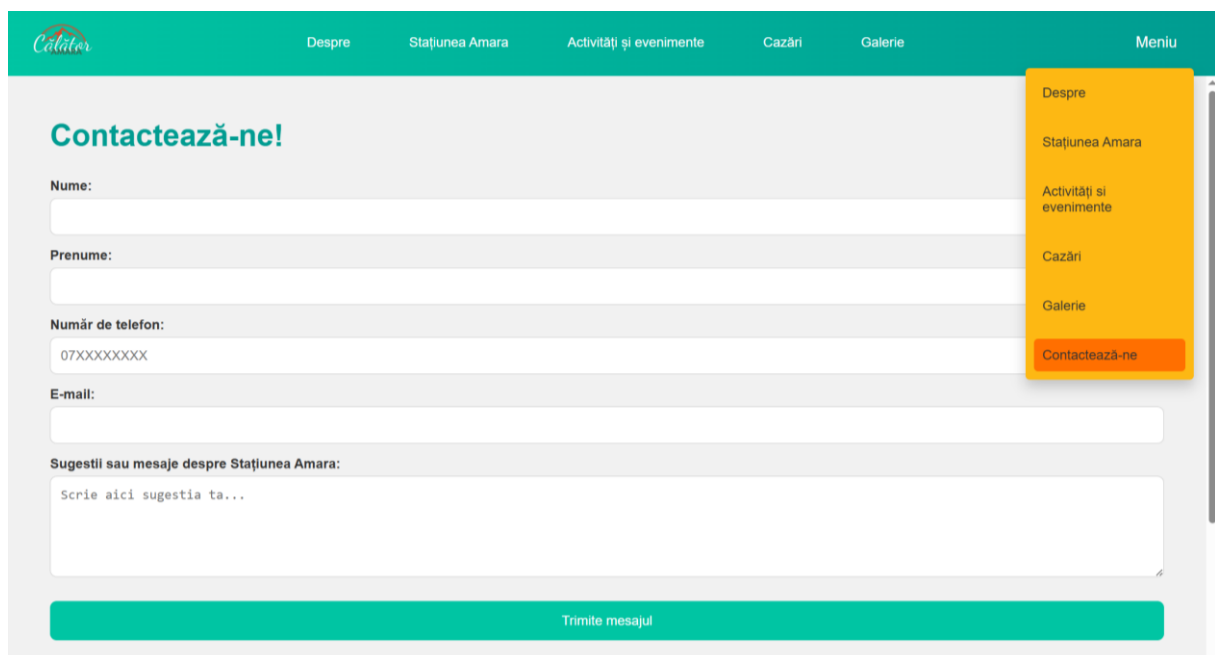
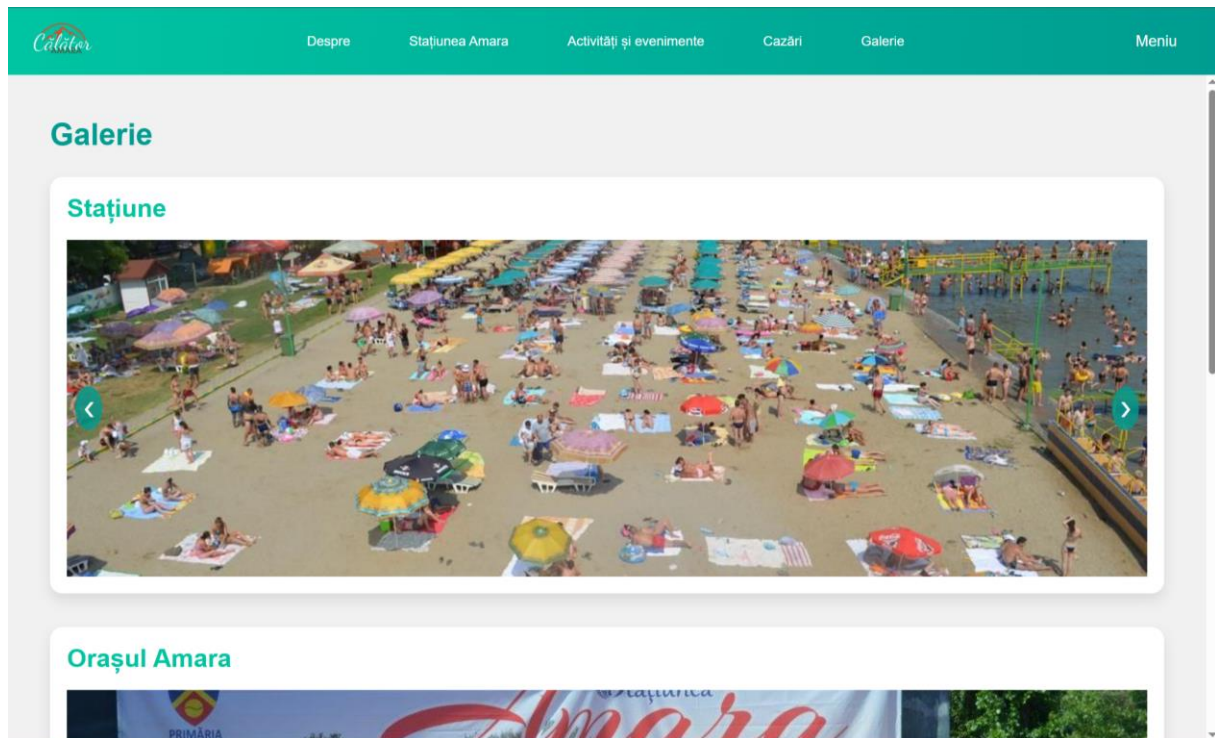
Nume:


E-mail:

Telefon:

07XXXXXXXX

Mesaj opțional:



DespreStațiunea AmaraActivități și evenimenteCazăriGalerieMeniu

E-mail:

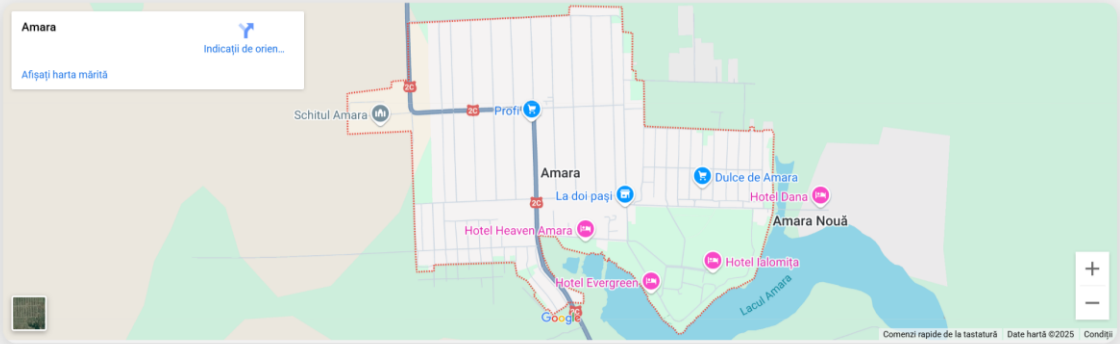
Sugestii sau mesaje despre Stațiunea Amara:

Trimite mesajul

Amara

Indicații de orien...

Afișaj harta mărită



VIII. BIBLIOGRAFIE

<https://web.ceiti.md/lesson.php?id=1#t1>

<https://www.pbinfo.ro/articole/21660/limbajul-html-legaturi-si-imagini>

https://www.euroteamtravel.ro/blog/amara-obiective-turistice/?utm_source=chatgpt.com

<https://romania.directbooking.ro/prezentare-amara-informatii-poze-imagini-18.aspx>

<https://biblioteca-amara.ro/>

<https://primaria-amara.ro/institutii-si-servicii-subordonate/>

https://web.ceiti.md/files/curs_css.pdf

<https://developer.mozilla.org>

<https://www.w3schools.com>

<https://tc39.es>

www.wikipedia.ro

<https://youtu.be/R8fMM73VrXs?si=NuhLcAkdILYwtsJi>