**Personal Website: Țuluca Maria-Laura Tema 2 TW**

Am creat acest site pentru a împărtăși mai multe despre activitățile mele, pasiunile pe care le am și proiectele la care am lucrat. Site-ul este realizat cu ajutorul tehnologiilor **HTML**, **CSS și puțin Javascript**, având un design modern și adaptabil, care se ajustează pe diverse dispozitive.

Site-ul este organizat într-o structură simplă, ușor de parcurs, cu următoarele secțiuni:

* **Pagina principală (Home)**: un meniu de navigare, alături de un logo personalizat, cu link-uri către diverse secțiuni ale site-ului(acesta se găsește pe toate paginile); un mesaj introductiv de tip **typewriter** și câteva „bule” cu informații suplimentare despre preferințele mele.
* **Pagina „Despre mine” (About)**: mai multe despre cine sunt, ce studiez și ce activități am realizat de-a lungul vieții. Am inclus informații despre familia mea, dar și despre hobby-urile mele, cum ar fi dansul, kickboxing-ul și artele marțiale.
* **Pagina „Portofoliu” (Portofolio)**: studiile până în prezent, premii, proiectele la care am lucrat în timpul studiilor, activități extra-curriculare, dar și abilitățile mele tehnice și soft skill-uri acumulate.
* **Pagina „Contact”**: aici sunt incluse toate modalitățile prin care mă poate contacta cineva – de la rețelele sociale până la un email direct.

**Funcționalități:**

1. **Meniu Burger**: meniul de navigare se transformă într-un meniu burger pentru a economisi spațiu și a face site-ul mai accesibil pe dispozitivele mobile sau la redimensionarea ferestrei browser-ului.
2. **Responsive Design**: am optimizat site-ul pentru diverse dimensiuni ale ecranului, inclusiv pentru dispozitive mobile și tablete.
3. **Animații CSS**: am folosit animații subtile pentru a face site-ul mai dinamic și atractiv, cum ar fi efecte la trecerea cursorului peste imagini și texte.
4. **Link-uri externe**: am integrat link-uri către profiluri de pe rețele sociale și alte platforme externe, pentru a facilita contactul și interacțiunea cu vizitatorii.

**Elemente de CSS:**

* **font-family și font-size**: le-am folosit pentru a seta fontul și dimensiunea textului.
* **background-size: cover**;: pentru redimensionarea imaginii de fundal și acoperirea întregii fereastre, indiferent de dimensiunea acesteia.
* **transition**: pentru adăugarea unui efect de tranziție lină la imagine și text când sunt modificate. De exemplu, când trec cu mouse-ul peste o imagine, aceasta se mărește ușor și capătă o umbră subtilă, iar când trec cu mouse-ul peste text, își schimbă culoarea și se mărește dimensiunea fontului.
* **@keyframes**: pentru animații CSS, aici am folosit animații fadeIn pentru a face titlul principal (h1) să apară treptat la încărcarea paginii sau slideIn pentru a face un întreg bloc de informații să apară de sus la încărcarea paginii.
* **max-width și height**: setez dimensiunile maxime ale imaginilor pentru a le face să se potrivească în layout-ul paginii și să nu depășească dimensiunea containerului lor.
* **margin, padding, text-align**: pentru a ajusta distanțarea și alinierea elementelor. De exemplu, pentru a centra textul în pagină și pentru a adăuga spațiu între imagini și text.

**Cum am făcut pagina responsive?**

Pentru a face pagina web să fie *responsive*, adică să se poată adapta la diferite dimensiuni de ecran, am folosit:

* **media queries** în CSS. Acestea sunt reguli care permit modificarea stilurilor în funcție de dimensiunea și tipul dispozitivului pe care se vizualizează pagina.
* ***display: flex*** în mai multe părți ale site-ului, ceea ce îmi permite să organizez elementele într-un mod flexibil, distribuite uniform, iar meniul să fie ajustat ușor la dimensiunile ecranului.
* în loc de unități fixe precum *px*, am folosit **unități relative** precum *em, rem, % sau vw/vh* (unități de viewport).

Pentru telefoane mobile(max-width: 768px):

* modific dimensiunile imaginilor, ale titlurilor și ale paragrafelor pentru a face textul mai lizibil pe ecrane mici;
* tabelele cu text și imagini sunt modificate pentru a se comporta ca o listă de blocuri, adică imaginea și textul să fie așezate unul sub altul, nu pe aceeași linie ca în cazul ecranelor mari și medii, unde nu intervin probleme;
* butoanele de navigare din carousel au dimensiunea fontului și padding-ul ajustate pentru a fi mai mici;
* meniul burger devine vizibil, folosind *display: flex* pentru a-l face să apară pe mobil; link-urile din meniu sunt ascunse inițial (cu *display: none*), dar devin vizibile doar atunci când burger-ul este activat(sunt rearanjate într-o coloană (*flex-direction: column*)); meniul are un fundal diferit și un padding pentru a-l face mai ușor de navigat pe ecrane mici.

**Media Query pentru tablete (max-width: 1024px)**:

* se aplică pe dispozitivele de dimensiuni medii, precum tabletele.
* fonturile sunt ajustate pentru a arăta mai bine pe aceste dispozitive, dar nu la fel de mici ca pe telefoane.
* în mod implicit, meniul burger nu este vizibil pe tablete (ecran mediu), fiind ascuns cu *display: none;*

**Media Query pentru desktop (min-width: 1025px)**:

* sunt aplicate pe ecranele mari, de obicei pe desktopuri.
* fonturile și stilurile sunt adaptate pentru a arăta bine pe ecranele mari, cu dimensiuni mai mari pentru texte și imagini.
* în mod implicit, meniul burger nu este vizibil pe desktopuri (ecran larg), fiind ascuns cu *display: none;*

### JavaScript:

Am utilizat JavaScript pentru a adăuga interactivitate și pentru a face site-ul mai dinamic. În acest cod, JavaScript-ul se ocupă de:

* afișarea conținutului suplimentar atunci când utilizatorul face clic pe anumite elemente, pe pagina de home; cu **funcția toggleContent(id),** careeste apelată atunci când utilizatorul face clic pe un *div* cu clasa .*bubble*. Când utilizatorul face clic pe "Education", "Competences", "Movies", etc., această funcție schimbă starea vizibilității pentru conținutul asociat (de exemplu, detaliile despre educație sau competențe); la fiecare clic, clasa visible este adăugată sau eliminată pentru elementul div cu id-ul respectiv, controlând astfel afișarea conținutului.