

Online Shop

Cerințe

1. HTML
 - a. Proiectul trebuie să conțină cel puțin 4 pagini web, maxim 8
2. CSS
 - a. Stilizarea trebuie să fie realizată în fișiere separate
 - b. Site-ul trebuie să fie responsive
 - c. Layout-ul să fie realizat cu ajutorul tabelelor și cadrelor
 - d. Site-ul trebuie să conțină un meniu drop-down realizat cu CSS
 - e. Site-ul trebuie să folosească transformări CSS
3. JavaScript / jQuery
 - a. Modificarea stilului unui element sau al unui grup de elemente
 - b. Folosirea funcțiilor în formulare
 - c. Folosirea evenimentelor de mouse și tastatură
 - d. Modificarea dinamică a poziției unui element
4. PHP + MySQL
 - a. Să conțină cel puțin un formular de înregistrare și login în PHP
 - b. Să folosească cel puțin o tabelă de tip MySQL pentru păstrarea și actualizarea informațiilor
 - c. Să se genereze o pagină cu conținut dinamic folosind PHP și informații luate dintr-o tabelă MySQL definind tabela cu date (> 50 de intrări) preluate din fișiere .CSV și importate în MySQL cu ajutorul interfeței PHPMyAdmin din XAMPP

Mod de funcționare

Site-ul nu va funcționa la parametrii normali decât dacă există un server (fie local, fie online) care să interpreteze codul PHP, și care să furnizeze o bază de date denumită *ewd* care conține o tabelă denumită *imagedb* de unde se vor prelua numele tuturor produselor, al imaginilor folosite de acestea, și prețul lor.

Pe lângă această tabelă, mai trebuie să existe una numită *credentials*, pentru a stoca detaliile conturilor utilizatorilor.

Prin urmare, site-ul poate fi rulat offline, daca aceste lucruri exista prin intermediul unui server local. Aplicația ideală este XAMPP, iar toate fișierele proiectului trebuie puse în directorul **htdocs** al aplicației. În folder-ul **phpmyadmin** se găsește tabela utilizată pentru informațiile actuale ale site-ului, care trebuie importate în serverul local.

Funcționalitate

Site-ul realizat este un magazin online, prin intermediul căruia se pot comanda diferite produse. El conține 6 pagini: `index.html`, `shop.php`, `contact.html`, `cart.html`, `login.html`, `register.html`.

Pagina [login.html](#) este prima pagină pe care utilizatorul o poate accesa, el fiind nevoit fie să se logheze, fie să își creeze un cont prin accesarea link-ului de înregistrare de sub formular. Validarea datelor este realizată prin compararea datelor introduse în câmpuri și datele disponibile pe server în baza de date *credentials*.

Pagina [register.html](#) este pagina folosită pentru înregistrarea unui nou utilizator. La apăsarea butonului *register*, datele sunt trimise la server și adăugate într-o tabelă numită *credentials*, unde vor fi folosite pentru a verifica datele login-ului.

Pagina [index.html](#) este pagina principală a site-ului, conținând un navbar cu numele companiei, și un meniu simplu cu 4 „butoane” pentru navigarea site-ului. Conținutul principal este un simplu div peste un background format dintr-un gradient fix între 3 culori, peste care se află un div cu numele companiei și sloganul acesteia. La sfârșitul paginii, se află un footer cu anul în care a fost realizat site-ul și numele developer-ului.

Pagina [shop.php](#) conține navbar-ul și footer-ul descrise anterior, iar ca și conținut, se generează dynamic, în funcție de intrările din tabela MySQL, toate produsele vândute de către magazin. Acestea sunt aranjate câte 4 pe rând, având evenimente de tip hover, cu ajutorul cărora se pun în evidență produsele individuale și butoanele de cumpărare. La apăsarea butonului de cumpărare a unui produs, se apelează un script JS care stochează informațiile produsului în *sessionStorage* pentru a putea fi preluate de către pagina [cart.html](#) în afișarea cart-ului.

Pagina [contact.html](#) are drept conținut un formular simplu, care cere numele, o adresă de email și mesajul care se dorește a fi trimis către companie. La apăsarea butonului *Submit* se acționează un script php care trimite email-ul folosind clientul default al sistemului.

Pagina [cart.html](#) se folosește de informațiile transmise prin intermediul *sessionStorage* pentru a alcătui cart-ul utilizatorului, având posibilitatea de a modifica numărul produselor, care va fi ținut minte și la părăsirea acestei pagini, sau de a șterge produsul din cart. Calculele din pagină sunt făcute în timp real, în funcție de numărul de produse și de cantitatea aleasă. Dacă nu sunt produse în cart, nu există posibilitatea de a face check-out, acest buton devenind ascuns până la adăugarea unor produse în cart.

Butonul *GO TO SECURE CHECKOUT* este nefuncțional, deoarece site-ul nu este legat de către un procesor de plăți în acest moment.

Stilizarea fiecărei pagini este făcută prin două fișiere .CSS, unul „style.css” care este importat la începutul fiecărui fișier .CSS corespunzător unei pagini. [style.css](#) conține informațiile despre navbar și footer, inclusiv meniul responsive. După acest fișier, fiecare pagină are stiluri, id-uri și clase proprii pentru stilizarea conținutului.

Navbar-ul este realizat drept un template, într-un fișier html separat, care este adăugat în pagină la încărcarea acesteia, el conținând nu doar meniul, ci și scriptul pentru animația meniului pentru site-ul mobil.

Paginile [shop.php](#) și [cart.html](#) au fiecare câte un fișier .JS care conține scripturile folosite în funcționalitatea acestor pagini: [shop.php](#) – funcționalitatea butonului de cumpărare; [cart.html](#) – calculele totalului, subtotalul fiecărui produs în funcție de cantitatea aleasă și funcționalitatea ștergerii produsului din *sessionStorage*, adică ștergerea sa din cart.

Tehnologii utilizate

Pentru a realiza site-ul, au fost folosite următoarele aplicații:

- Visual Studio Code – editor de text / cod
- XAMPP – server local, bază de date MySQL și interpretare PHP

- W3schools, CodePen, StackOverflow – resurse online pentru cod, soluții la probleme, referință online
- Toptal.com – subtlepatterns pentru produsele de pe site