**MINISTERUL EDUCAŢIEI NAŢIONALE**

**COLEGIUL NAŢIONAL „ELENA CUZA”, CRAIOVA**

**PROIECT**

**pentru**

**EXAMENUL DE ATESTAT PROFESIONAL**

***INFORMATICĂ***

**Absolvent : Alexandru GRECU**

**Clasa : a XII-a F**

**Specializarea matematică-informatică**

**Îndrumător ştiinţific :**

**prof. Alina Petronela COTFASĂ**

**An şcolar 2021-2022**

**MINISTERUL EDUCAŢIEI NAŢIONALE**

**COLEGIUL NAŢIONAL „ELENA CUZA”, CRAIOVA**

**MAGAZIN ONLINE**

**Site**

**Absolvent :Alexandru GRECU**

**Clasa : a XII-a F**

**Specializarea matematică-informatică**

**Îndrumător ştiinţific :**

**prof. Alina Petronela COTFASĂ**

**An şcolar 2021-2022**

**CUPRINS**

|  |  |
| --- | --- |
| **JUSTIFICARE** | **3** |
| **CONSIDERAȚII TEHNICE DESPRE REALIZAREA SITE-ULUI** | **4** |
| **CAPITOLUL I. ASPECTE TEORETICE** | **5** |
| ***I.1. HTML*** | ***5*** |
| *I.1.1. Noțiuni de bază* | *5* |
| *I.1.2. Taguri de baza* | *6* |
| *I.1.3.Titlul unei pagini* | *6* |
| *I.1.4. Culoare de fond* | *6* |
| *I.1.5. Culoarea textului* | *7* |
| *I.1.6. Fonturi* | *7* |
| *I.1.7. Imagini* | *9* |
| *I.1.8. Legături* | *11* |
| *I.1.9. Liste* | *13* |
| ***I.2.CSS*** | ***14*** |
| ***I.3. JAVASCRIPT*** | ***15*** |
| ***I.4. FIREBASE (BAZA DE DATE)*** | ***16*** |
| **CAPITOLUL II. PREZENTARE SITE** | 17 |
| ***II.1. STRUCTURA SITE-ULUI*** | ***17*** |
| ***II.2. PAGINA PRINCIPALA*** | ***18*** |
| ***II.3. PAGINA MEN*** | ***18*** |
| ***II.4. PAGINA WOMEN*** | ***19*** |
| ***II.5. PAGINA SALE*** | ***20*** |
| ***II.6. PAGINA PRODUSULUI*** | ***20*** |
| ***II.7. PAGINA REGISTER*** | ***21*** |
| ***II.8. PAGINA LOG IN*** | ***22*** |
| ***II.9. PAGINA CART (COSUL DE CUMPARATURI)*** | ***22*** |
| ***II.10. PAGINA ACCOUNT (CONTUL UTILIZATORULUI)*** | ***23*** |
| ***II.11. PREZENTARE SURSA COD*** | ***24*** |
| **REFERINȚE** | **31** |

**JUSTIFICARE**

Tema aleasa este un site sub forma unui magazin online de imbracaminte. Aceasta tema a luat contur in mintea mea datorita faptului ca in zilele noastre este mult mai eficient si rapid sa iti procuri produsele de pe internet, in loc sa pierzi o multime de timp prin magazine. Din ce in ce mai multi oameni prefera aceasta metoda de achizitionare a produselor deoarece considera ca in felul acesta salveaza timp si energie. Un astfel de magazin online este mult mai profitabil fata de magazinele fizice deoarece toata lumea are acces la el indiferent de pozitia sa pe glob. Site-ul realizat de mine contine produse vestimentare si este benefic persoanelor carora nu le place sa isi petreaca timpul prin mall-uri si magazine proband produse. Condiser ca in zilele noastre timpul a devenit un lucru extrem de pretios ce trebuie folosit cu grija si din acest motiv m-am decis sa abordez aceasta tema.

Activitatea de proiectare a site-urilor Web implică: imaginaţie, capacitatea de analiză, creativitate şi o cunoaştere temeinică a limbajului HTML. Proiectarea unui site are mai multe etape:

- identificarea cu exactitate a obiectivului,

- culegerea şi selectarea informaţiilor,

- proiectarea structurii şi a aspectului grafic al site-ului,

- implementarea

- testarea,

- publicarea

- etapa de promovare.

Structura site-ului este, practic, organizarea informației și a posibilităților utilizatorilor de a interacționa cu site-ul. O structură gândită corect de la început asigură îndeplinirea cu succes a scopului pentru care s-a dezvoltat site-ul. Importante sunt și navigarea ușoară și satisfacția vizitatorilor, care preferă întotdeauna să-și rezolve problemele repede și fără complicații inutile.

Modul în care site-ul este structurat influențează comportamentul vizitatorului. Rămâne un timp mai îndelungat pe site și vizitează și celelalte produse sau secțiuni? Recomandă site-ul și celor din rețeua sa socială? Cumpără? Nu se împotmolește căutând produsul dorit? Nu se blochează la coșul de cumpărături? Revine și în viitor? Toate acestea pot fi indicatori pentru un site bine structurat.

**CONSIDERAȚII TEHNICE DESPRE REALIZAREA SITE-ULUI**

Limbajul de baza in care a fost dezvoltat acest atestat este limbajul HTML. Prin intermediul acestuia am construit baza/scheletul site-ului, structura ca sustine logica, cat si aspectul site-ului.

Aspecte legate de imaginea site-ului, precum fundal, stiluri de butoane, font, etc. au fost implementate prin itnermediul exinderii limbajului HTML cu ajutorul limbajului CSS (Cascading Style Sheets).

Limbajul care se ocupa de rularea si logica site-ului se numeste JavaScript.

Site-ul foloseste si o baza de date. Aceasta este hostata in cloud pe platforma Firebase. Prelucrarea datelor din aceasta este realizata prin intermediul API-urilor.

Site-ul realizat de mine prezinta multe functionalitati si este capabil de a fi pus in rulare pe internet. Acesta este capabil de urmatoarele:

* Creare de cont de client
* Accesare de cont de client
* Afisare catalog cu informatiile necesare despre produs (poza, pret, denumire, reducere, pret redus, marimi si culori valabile in stoc)
* Cautare dinamica a unui produs din baza de date
* Sortarea produselor afisate in functie de anumite criterii
* Alegerea marimii si a culorii produsului ales
* Adaugarea produselor in cosul de cumparaturi (chiar daca clientul nu are cont)
* Adaugarea produselor la favorite
* Schimbarea cantitatii, culorii sa a marimii produselor aflate deja in cos, inainte de checkout
* Checkout: Introducerea adresei de email, nume, numar de telfon, adresa de livrare, informatii card, urmate de plasarea comenzii
* Adaugarea produselor cumparate in istoricul de cumparari, vizibil in detaliu din contul utilizatorului
* In cazul in care apare vreo eroare, utilizatorul este intampinat de un mesaj informativ ce ii ofera posbilitatea sa raporteze problema
* Responsive design. Site-ul are un aspect placut si prietenos pe orice dispozitiv (tableta/telefon/calculator)

**CAPITOLUL I. ASPECTE TEORETICE**

**I.1. HTML**

**HyperText Markup Language** (**HTML**) este un [limbaj de marcare](https://ro.wikipedia.org/wiki/Limbaj_de_marcare" \o "Limbaj de marcare) utilizat pentru crearea paginilor web ce pot fi afișate într-un [browser](https://ro.wikipedia.org/wiki/Browser) (sau navigator). Scopul HTML este mai degrabă prezentarea informațiilor – paragrafe, fonturi, tabele ș.a.m.d. – decât descrierea semanticii documentului. În cadrul [dezvoltării web](https://ro.wikipedia.org/wiki/Dezvoltare_web) de tip [front-end](https://ro.wikipedia.org/wiki/Front-end_%C8%99i_back-end" \o "Front-end și back-end), HTML este utilizat împreună cu [CSS](https://ro.wikipedia.org/wiki/CSS" \o "CSS) și [JavaScript](https://ro.wikipedia.org/wiki/JavaScript).

Specificațiile HTML sunt dictate de [World Wide Web Consortium](https://ro.wikipedia.org/wiki/World_Wide_Web_Consortium) (W3C), care încurajează utilizarea CSS în locul a HTML explicit pentru prezentare.[[1]](https://ro.wikipedia.org/wiki/HyperText_Markup_Language#cite_note-deprecated-1)

TML este o formă de marcare orientată către prezentarea documentelor text pe o singura pagină, utilizând un software de redare specializat, numit *agent utilizator HTML*, cel mai bun exemplu de astfel de software fiind *browserul web*. HTML furnizează mijloacele prin care conținutul unui document poate fi adnotat cu diverse tipuri de metadate și indicații de redare. Indicațiile de redare pot varia de la decorațiuni minore ale textului, cum ar fi specificarea faptului că un anumit cuvânt trebuie subliniat sau că o imagine trebuie introdusă, până la scripturi sofisticate, hărți de imagini și formulare. Metadatele pot include informații despre titlul și autorul documentului, informații structurale despre cum este împărțit documentul în diferite segmente, paragrafe, liste, titluri etc. și informații cruciale care permit ca documentul să poată fi legat de alte documente pentru a forma astfel [hiperlink](https://ro.wikipedia.org/wiki/Hiperlink" \o "Hiperlink)-uri (sau [web](https://ro.wikipedia.org/wiki/Web" \o "Web)-ul).

HTML se poate genera direct utilizând tehnologii de codare din partea serverului cum ar fi [PHP](https://ro.wikipedia.org/wiki/PHP" \o "PHP), [JSP](https://ro.wikipedia.org/w/index.php?title=JavaServer_Pages&action=edit&redlink=1) sau [ASP](https://ro.wikipedia.org/wiki/ASP). Multe aplicații ca [sistemele de gestionare a conținutului](https://ro.wikipedia.org/w/index.php?title=Content_management_system&action=edit&redlink=1" \o "Content management system — pagină inexistentă), [wiki-uri](https://ro.wikipedia.org/wiki/Wiki) și [forumuri web](https://ro.wikipedia.org/w/index.php?title=Forum_web&action=edit&redlink=1) generează pagini HTML.

**I.1.1. Noțiuni de bază**

HTML este prescurtarea de la Hyper Text Mark-up Language si este codul care sta la baza paginilor web. Paginile HTML sunt formate din etichete sau tag-uri și au extensia .html sau .htm .În marea lor majoritate aceste etichete sunt pereche, una de deschidere <eticheta> și alta de închidere </eticheta>, mai există și cazuri în care nu se închid, atunci se folosește <eticheta /> browserul interpretează aceste etichete afișând rezultatul pe ecran. HTML-ul nu este un limbaj case sensitiv (nu face deosebirea între litere mici și mari). Pagina principala a unui domeniu este fisierul index.html respectiv index.htm Această pagină este setată a fi afișată automat la vizitarea unui domeniu. De exemplu la vizitarea domeniului www.nume.ro este afișată paginawww.nume.ro/index.html.

Unele etichete permit utilizarea de atribute care pot avea anumite valori: <eticheta atribut="valoare"> ... </eticheta>

Componenta unui document HTML este: 1. versiunea HTML a documentului 2. zona head cu etichetele <head> </head> 3. zona body cu etichetele <body> </body> sau <frameset> </frameset>

Toate paginile HTML încep și se termină cu etichetele <html> și </html>. În interiorul acestor etichete găsim perechile <head>, </head> și <body>, </body>.

Tag-uri.

Orice fișier html are următoarea structură:

<html>

<head>

<title>Titlul</title>

</head>

<body>

………………………

</body>

</html>

Între tagurile <body></body> se scriu celelalte "instrucțiuni". Tagul <body> permite stabilirea fundalului și a culorii hiperlinkurilor.

<body bgcolor="#ff0000"> - culoarea fundalului paginii devine rosie

<body background="fundal.jpg"> - pune o imagine în fundalul paginii

<body link="#ff0000" alink="#00ff00" vlink="#0000ff"> - stabilește culoarea linkurilor nevizitate și vizitate.

**I.1.2. Taguri de baza**

Orice document HTML incepe cu notatia **<**html**>** si se termina cu notatia **</**html**>**, numite in literatura de specialitate "TAG-uri". Prin conventie, toate informatiile HTML incep cu o paranteza unghiulara deschisa **"<"** si se termina cu o paranteza unghiulara inchisa **">"**.  
Tag-urile intre aceste paranteze transmit comenzi catre browser pentru a afisa pagina intr-un anumit mod. Unele blocuri prezinta delimitator de sfarsit de bloc, in timp ce pentru alte blocuri acest delimitator este optional sau chiar interzis.

O eticheta poate fi scris atat cu litere mici, cat si cu litere mari.  
Adica **<**HTML**> = <**HtmL**> = <**html**>**. Caracterele "spatiu" si "CR/LF" ce apar intre etichete sunt ignorate de catre browser.

**I.1.3.Titlul unei pagini**

Titlul unei pagini se obtine inserand in sectiunea **<**head**>...</**head**>** a urmatoarei linii:

<title>Aceasta este prima mea pagina de Web</title>

In plus, in sectiunea **<**body**>...</**body**>** putem scrie texte cat dorim. Daca nu intalnim nici un marcaj **<** sau **>** atunci interpretorul HTML le va lua ca texte simple si le va afisa pe ecran.

Continutul blocului **<**title**>**...**</**title**>** va aparea in bara de titlu a ferestrei browser-ului. Daca acest bloc lipseste intr-o pagina HTML, atunci in bara de titlu a ferstrei browser-ului va aparea numele fisierului.

**I.1.4. Culoare de fond**

O culoare poate fi precizata in doua moduri:Printr-un nume de culoare.Sunt disponibile cel putin 16 nume de culori: aqua, black, fuchsia, gray, green lime, maroon, navy, olive, purple, red, silver, teal, white si yellow.

Prin constructia " #rrggbb " unde r (red), g (green), sau b (blue) sunt cifre hexazecimale si pot lua valorile: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, A, b, B, c, C, d, D, e, E, f, F; se pot defini astfel 65536 de culori.

Culoarea unei pagini se precizeaza prin intermediul unui atribut al etichetei <body>.

Culoarea fondului paginii Web se stabileste cu atributul bgcolor al etichetei <body>, de exemplu: <body bgcolor = culoare>. 

**I.1.5. Culoarea textului**

Acest lucru se face prin intermediul atributului text al etichetei <body> dupa sintaxa<body text=culoare>. In urmatorul exemplu textul are culorea rosie..

Textul afisat este caracterizat de urmatoarele atribute: Marime (size), Culoare (color), Font (style).

Acestea sunt atribute ale etichetei <basefont>.Este o eticheta singulara (fara delimitator de sfarsit de bloc).

<basefont size = numar color = culoare style = font>

unde:

* numar - poate fi 1, 2, 3, 4, 5, 6 sau 7; (1 pentru fontul cel mai mic si 7 pentru fontul cel mai mare);
* culoare - este o culoare precizata prin nume sau printr-o constructie RGB;
* font - poate fi un font generic ca " serif ", " san serif ", " cursive ", " monospace ", "fantasy " sau un font specific instalat pe calculatorului clientului, ca " Times New Roman ", " Helvetica " sau " Arial ".Se accepta ca valoare si o lista de fonturi separate prin virgula, de exemplu:

" Times New Roman, serif,monospace ".

Domeniul de valabilitate al caracteristicelor precizate de aceasta eticheta se intinde de la locul in care apare eticheta pana la sfarsitul paginii sau pana la urmataorea eticheta<basefont>.

Daca acest atribut lipseste atunci textul din pagina Web are atribute prestabilite sau atribute precizate de browserul utilizat.

Atributele prestabilite sunt: size = 3, color = black, si style = " Times New Roman " .

Pozitionarea continutului paginii Web fata de marginile ferestrei browserului se poate face cu ajutorul a doua atribute ale etichetei <body>:

* leftmargin (stabileste distanta dintre marginea stanga a ferstrei browserului si marginea stanga a continutului paginii);
* topmargin (stabileste distanta dintre marginea de sus a ferstrei browserului si marginea de sus a continutului paginii);

**I.1.6. Fonturi**

Un font este caracterizat de urmatoarele atribute:

* culoare (stabilita prin atributul color);
* tipul sau stilul (stabilit prin atributul face);
* marimea (definita prin atributul size);
* marimea in puncte tipografice (stabilita prin atributul point-size);
* grosime (definita prin atributul weight).

Toate aceste atribute apartin etichetei, care permite inserarea de blocuri de texte personalizate.

Culori

O culoare poate fi precizata in doua moduri:

* printr-un nume de culoare.
* printr-o constanta conform standardului de culoare RGB (Red, Green,Blue). O astfel de constanta se formeaza astfel: #rrggbb, unde r, g si b sunt cifre hexazecimale.

Culorea fontului. Pentru a scrie un fragment de text cu caractere de o anumita culoare se incadreaza acest fragment intre delimitatorii ... avand stabilit atributul color la valoarea necesara.

De exemplu:

<font color=red>fragment de text de culoare rosie</font>

Familia fontului

Pentru a scrie un text intr-o pagina pot fi folosite mai multe fonturi (stiluri de caractere). Exista cinci familii generice de fonturi care sunt de regula disponibile pe toate calculatoarele utilizatorilor: serif, sans serif, cursive, monospace si fantasy.

Tipul de font necesar poate fi stabilit prin atributul face al etichetei.   
Pot fi introduse mai multe fonturi separate prin virgula.

<font face="Arial, serif, monospace">

Marimea fontului

Pentru a stabili marimea unui font se utilizeaza atributul size al etichetei. Valorile acestui atribut pot fi:

* 1, 2, 3, 4, 5, 6, 7 ( 1 pentru cel mai mic font si 7 pentru cel mai mare);
* +1, +2, etc. pentru a mari dimensiunea fontului cu 1, 2, etc. fata de valoarea curenta;
* -1, -2, etc. pentru a micsora dimensiunea fontului cu 1, 2, etc. fata de valoarea curenta.

Marimea unui font poate fi stabilita exact cu ajutorul atributului point-size. Valorile acceptate de acest atribut pot fi orice numere naturale pozitive.Numarul astfel precizat reprezinta marimea fontului in puncte tipografice.

Grosimea unui font

Grosimea unui caracter poate fi definita cu ajutorul atributului weight al etichetei.   
Valorile posibile pentru acest atribut sunt 100, 200, 300, 400, 500, 600, 700, 800 si 900 (100 pentru fontul cel mai subtire si 900 pentru cel mai gros).

Intr-un fisier HTML, caracterele "<" si ">" au o semnificatie speciala pentru browser. Ele incadreaza comenzile si atributele de afisare a elementelor intr-o pagina. Daca dorim ca un fragment de text sa contina astfel de caractere, acest fragment trebuie incadrat de una dintre perechile de etichete:

* <xmp>...</xmp> ( 80 de caractere pe rand );
* <listing>...</listing> ( 120 de caractere pe rand ).

Aceste marcaje interpreteaza corect caracterele " spatiu ", " eticheta " si "CR/LF ". Textul afisat in pagina este monospatiat.

Blocuri paragraf

Cu ajutorul etichetei paragraf <p> este posibil trecerea la o linie noua si permite:

inserarea unui spatiu suplimentar inainte de blocul paragraf;

* inserarea unui spatiu suplimentar dupa blocul paragraf, daca se foloseste delimitatorul </p> (acesta fiind optional);
* alinierea textului cu ajutorul atributului align, avand valorile posibile " left ", " center " sau " right ".

Blocuri de titlu

Intr-un text titlurile ( headers ) de capitole pot fi introduse cu ajutorul etichetelor <h1>, <h2>, <h3>, <h4>, <h5>, <h6>.

Toate aceste etichete se refera la un bloc de text si trebuie insotite de o eticheta de incheiere similara.

Aceste etichete accepta atributul align pentru alinierea titlului blocului de text la stanga (in mod prestabilit ), in centru si la dreapta. Tag-ul <h1> permite scrierea unui titlu cu caractere mai mari si aldine, pe cand <h6> foloseste caracterele cele mai mici.

Linii orizontale

Intr-o pagina Web pot fi inserate linii orizontale.acest lucru se face cu ajutorul etichetei<hr>. Pentru a configura o linie orizontala se utilizeaza urmatorele atribute ale etichetei<hr>:

* align permite alinierea liniei orizontala. Valorile posibile sunt "left" ,"center" si "right";
* width permite alegerea lungimii liniei;
* size permite alegerea grosimii liniei;
* noshade cand este prezent defineste o linie fara umbra;
* color permite definirea culorii liniei.

Blocuri <center>

Blocul introdus de etichetele <center>...</center> aliniaza centrat toate elementele pe care le contine.

Blocuri <nobr>

Blocul de text cuprins intre etichetele <nobr>...</nobr> va fi afisat pe o singura linie.

Blocuri <div>

Modalitatea cea mai eficienta de delimitare si de formatare a unui bloc de text este folosirea delimitatorilor <div>...</div>. Un parametru foarte foarte util pentru stabilirea caracteristicilor unui bloc <div> ( diviziune ) este align ( aliniere ).

Valorile posibile ale acestui parametru sunt:

* " left " ( aliniere la stanga );
* " center " ( aliniere centrala );
* " right " ( aliniere la dreapta ).

Un bloc <div>...</div> poate include alte subblocuri. In acest caz , alinierea precizata de atributul align al blocului are efect asupra tuturor subblocurilor incluse in blocul <div>;   
Un bloc <div>...</div> admite atributul " nowrap " care interzice intreruperea randurilor de catre browser.

**I.1.7. Imagini**

Imaginile sunt stocate in fisiere cu diverse formate. Formatele acceptat de browsere pentru fisierele imagine sunt:

* GIF (Graphics Interchange Format) cu extensia .gif;
* JPEG (Joint Photographic Experts Group) cu extensia .jpeg sau .jpg;
* XPM (X PixMap) cu extensia .xmp;
* XBM (X BitMap) cu extensia .xbm;
* BMP (BitMap) cu extensia .bmp (numai cu Internet Explorer);
* TIFF (Tagged Image File Format) cu extensia .tif sau .tiff;

Cele mai raspandite formate sunt GIF(8biti pentru o culoare, 256 culori posibile) si JPEG (24biti pentru o culoare, 16777216 de culori posibile).

Adresa URL a unei imagini

URL ( "Uniform Resourse Locator" = identificator unic al resursei ) este un standard folosit in identificarea unica a unei resurse in Internet.

Toate imaginile cu care vom lucra vor avea adresa URL exprimata in functie de directorul ce contine documentul HTML care face referire la imagine.

Pentru a insera o imagine intr-o pagina, se utilizeaza eticheta <img> (de la "image"=imagine).

Pentru a putea fi identifica imaginea care va fi inserata, se utilizeaza un atribut al etichetei <img> si anume src (de la "source"=sursa).

Daca imaginea se afla in acelasi director cu fisierul HTML care face referire la imagine, atunci adresa URL a imaginii este formata numai din numele imaginii, inclusiv extensia.

<html>  
<head><title> O pagina cu imagine</title></head> <body> O pagina care contine o imagine

<img src="w3.gif"> Text dupa imagine.

</body>  
</html>

Chenarul si dimensionarea unei imagini

Daca doriti sa adaugati un chenar in jurul imaginii, folositi atributul border al etichetei<img>.  Valorile acestor atribute sunt numere intregi pozitive.

O imagine are anumite dimensiuni pe orizontala si verticala, stabilite in momentul crearii ei. Daca nu se cere altfel , aceste dimensiuni sunt respectate in momentul afisarii ei in pagina Web. Dimensiuniile prestabilite ale unei imagini pot fi modificate prin intermediul atributelor width si height.

<html>  
<head><title> Imagine cu chenar si marit</title></head> <body> O imagine cu chenar si de 200 pixeli X 15 %

<img src="w3.gif" border="5" width="350" height="25%"> Text dupa imagine.

</body>  
</html>

Alinierea unei imagini

Alinierea unei imagini se poate face prin intermediul atributului align care poate lua urmatorele valori:

* " left " - aliniere la stanga; celelalte componente sunt dispuse pe in partea dreapta;
* " right " - aliniere la dreapta; celelalte componente sunt dispuse pe in partea stanga;
* " top " - aliniere deasupra; partea de sus a imaginii se aliniaza cu partea de sus a textului ce precede imaginea;
* " middle " - aliniere la mijloc; mijlocul imaginii se aliniaza cu linia de baza a textului ce precede imaginea.
* " bottom " - aliniere la baza; partea de jos a imaginii se aliniaza cu linia de baza a textului.

Alinierea textului in jurul imaginii

Atributele hspace si vspace precizeaza distanta in pixeli pe orizontala , respectiv pe verticala, dintre imagine si restul elementelor din pagina.   
Atributul alt admite ca valoare un text care va fi afisat in locul imaginii.

Imagini pentru fondul unei pagini

O imagine poate fi utilizata pentru a stabili fondul unei pagini Web. In acest scop se foloseste atributul background al etichetei <body>, avand ca valoare adresa URL a imaginii.   
Imaginea se multiplica pe orizontala si pe verticala pana umple intregul ecran.

<html> <head><title> Pagina cu imagine de fond</title></head> <body background="w3.gif">  
1<br>2<br>3<br>4<br> 5<br>6<br>7<br>8<br> 9<br> </body>  
</html>

Imagini folosite ca legaturi

O legatura (link) introduce in pagina Web o zona activa.   
Efectuand click cu butonul mouse-ului pe aceasta zona in browser se va incarca o alta pagina.   
Pentru a utiliza imaginea "ubm1.jpg" drept legatura catre pagina index.html utilizam sintaxa:

<a href ="index-2.html"><img src= "w3.gif"></a>

In mod prestabilit imaginea utilizata pe post de zona activa este inconjurata de un chenar avand culoarea unei legaturi.

Daca stabilim pentru atributul border al etichetei <img> 0 acest chenar dispare.

<html> <head><title> Imagini folosite ca legaturi</title></head> <body><h5>Imagini folosite ca legaturi</h5>

Text inainte de imagine.<a href="index-2.html"><img src="w3.gif"></a>

Text dupa imagine.

</body> </html>

Utilizari speciale ale imaginilor

Imaginile pot fi utilizate pentru a obtine efecte deosebite intr-o pagina web.  
Printre aceste utilizari speciale putem enumera:

* Linii orizontale formate cu ajutorul imaginilor .
* Simboluri speciale pentru elementele unei liste neordonate.

**I.1.8. Legături**

Legaturile (link-urile) reprezinta partea cea mai importanta a unei pagini Web.  
Ele transforma un text obisnuit in hipertext sau hiperlegatura, care premite trecerea rapida de la o informatie aflata pe un anumit server la alta informatie memorata pe un alt server aflat oriunde in lume.

Legaturile sunt zone active intr-o pagina Web, adica zone de pe ecran sensibile la apasarea butonului stang al mouse-ului.

* legatura catre o pagina aflata in acelasi director
* legatura catre o pagina aflata in acelasi director se formeaza cu ajutorul etichetei <a>(de la "anchor"=ancora).

Pentru a preciza pagina indicata de legatura se utilizeaza un atribut al etichetei <a>numit href, care ia valoare numele fisierului HTML aflat in acelasi director. Zona activa care devina sensibila la apasarea butonului stang al mouse-ului este formata din textul cuprins intre etichetele <a>...</a>. Prezenta etichtetei de sfarsit </a> este obligatorie.

O legatura catre o pagina aflata pe acelasi disc local

Daca pagina referita se afla pe acelasi disc local, dar intr-un alt director atunci pentru a preciza pozitia ei in structura de directoare se poate folosi adresarea relativa.

Ancore

Intr-o pagina foarte lunga pot exista puncte de reper catre care se definesc legaturi. O ancora se defineste de asemenea prin eticheta <a>. Pentru a defini ancora se utilizeaza atributul name care primeste ca valoare un nume atribuit ancorei (de exemplu "leg1").

Pentru a insera o legatura catre "leg1" definita in aceeasi pagina se utilizeaza eticheta<a> avand atributul href de valoare "#leg1".

Pentru a introduce o legatura catre o ancora definita in alt document (alta pagina) aflat in acelasi director, atributul href primeste o valoare de forma "nume\_fisier.html#nume\_ancora".

Ancore definite prin atributul id

Atributul id este un atribut universal ,adica poate fi atasat oricarui element al unei pagini Web. Acest atribut va inlocui complet atributul name, care putea fi atasat numai anumitor elemente. Atributul id primeste ca valoare un nume (de exemplu "id1") care identifica in mod unic un element. Atributul id poate fi utilizat pe post de ancora intr-o pagina Web conform sintaxei:  
<eticheta id = "id1"> ... </eticheta>

<a href = "#id1" >Link catre elementul "id1" </a>

unde "eticheta" poate fi orice element.

Alegerea culorilor pentru legaturi

In mod prestabilit se utilizeaza trei culori pentru legaturi:

* culoare pentru legaturile nevizitate (nu s-a efectuat nici un clic pe ele)
* culoare pentru legaturile vizitate (s-a efectuat cel putin un clic pe ele)
* culoare pentru legaturile active (deasupra carora se afla mouse-ul la un moment dat).

Aceste atribute pot fi stabilite cu ajutorul a trei atribute ale etichetei <body>:

* link pentru legaturile nevizitate;
* vlink pentru legaturile vizitate;
* alink pentru legaturile active.

Valorile pe care le pot lua aceste atribute sunt culori definite prin nume sau conform standardului RGB.

Utilizarea postei electronice

Intr-o pagina Web se poate afla legaturi care permit lansarea in executie a aplicatiei de expediere a mesajelor electronice. Pentru aceasta se utilizeaza in constructia adresei URL serviciul Internet mailto: urmat de o adresa e-mail valida.

Pentru ca exemplul urmator sa functioneze trebuie ca:

* pe calculator sa fie instalat o aplicatie de expediere a mesajelor electrionice (Outlook Express pe calc. Windows , Pine pe calc. Unix);
* adresa sa fie valida si calculatorul sa fie conectat la Internet.

Legaturi catre fisiere oarecare

O pagina Web poate contine legaturi catre orice tip de fisiere aflate pe orice servere din Internet. Pentru aceasta se utilizeaza eticheta <a> avand valoarea atributului href egala cu adresa URL a fisierului destinatie.

Atunci cand se efectueaza clic pe legatura din exemplul urmator browserul se deschide o caseta de dialog - File download - care va permite:

* sa salvati pe discul local fisierul
* sa lansati in executie aplicatia capabila sa interpreteze corect fisierele de tipul respectiv

Atributul title

Atributul title apartine etichetei <a> si comanda aparitia unei mici ferestre in pagina Web cand mouse-ul se afla pe o legatura, fereastra in care este afisata valoarea data acestui atribut.   
Acest atribut are astfel menirea de a furniza informatii suplimentare despre semnificatia unei legaturi.

Legaturi catre fisiere de sunet

Sunetele pot fi stocate in fisiere in diverse formate:

* AU/m-law cu extensia .au;
* AIFF/AIFC cu extensiile .aiff, .aif;
* WAVE/WAV cu extensia .wav;
* MPEG Audio cu extensia .mpeg2, sau .mp2;
* MIDI cu extensia .mid sau .midi;

O legatura catre un fisier de sunet se realizeaza folosind eticheta <a> destinata legaturilor catre orice tip de fisiere, unde atributul href va avea valoarea egala cu adresa URL a fisierului de sunet.

Legaturi catre fisiere videoclipuri

Videoclipurile sunt stocate in fisiere diverse formate.Formatele si extensiile corespunazatoare pentru fisierele utilizabile in paginile Web sunt urmatoarele:

* MPEG cu extensia .mpeg sau mpg;
* QuickTime cu extensia .mov;
* AVI cu extensia .avi.

O legatura catre un fisier de sunet se realizeaza folosind eticheta <a> destinata legaturilor catre orice tip de fisiere, unde atributul href va avea valoarea egala cu adresa URL a fisierului videoclip. 

**I.1.9. Liste**

Unul din cele mai obisnuite elemente din documentele cu mai multe pagini este un set de definitii, referinte sau indexuri. Glosarele sunt exemple clasice in acest sens; cuvintele sunt listate in ordine alfabetica, urmate de definitii ale termenilor respectivi. In HTML, intreaga sectiune a unui glosar va fi gestoinata printr-o lista de definitii, care este inclusa intr-o pereche de marcaje de lista de definitii: <dl>...</dl> (de la "definition list" = lista de definitii).

Observatii:

- Un termen al listei este initiat de eticheta <dt> (de la "definition term" = termen definit);

- Definitia unui termen este initiata de eticheta <dd> (de la "definition description" = descrierea definitiei);

- Definitia unui termen incepe pe o linie noua si este indentata;

Liste neordonate

O lista neordonata este un bloc de text delimitat de etichetele corespondente <ul>...</ul> (" ul " vine de la " unordered list " = lista neordonata). Fiecare element al listei este initiat de eticheta <li> (list item).

Lista va fi indentata fata de restul paginii Web si fiecare element al listei incepe pe un rand noua. Tag-urile <ul> si <li> pot avea un atribut type care stabileste caracterul afisat in fata fiecarui element al listei. Valorile posibile al acestui atribut sunt:

* "circle" (cerc)
* "disc" (disc plin) (valoarea prestabilita);
* "square" (patrat)

Listele neordonate pot fi imbricate pe mai multe niveluri.

Liste ordonate

O lista ordonata de elemente este un bloc de text delimitat de etichetele corespondente<ol>...</ol> (" ol " vine de la " ordered list " = lista ordonata). Fiecare element al listei este initiat de eticheta <li> (list item).

Lista va fi indentata fata de restul paginii Web si fiecare element al listei incepe pe un rand noua. Tag-urile <ol> si <li> pot avea un atribut type care stabileste tipul de caractere utilizate pentru ordonarea listei.Valorile posibile sunt:

* " A " pentru ordonare de tipul A , B , C , D etc. ( litere mari );
* " a " pentru ordonare de tipul a , b , c , d etc. ( litere mici );
* " I " pentru ordonare de tipul I , II , III , IV etc. ( cifre romane mari );
* " i " pentru ordonare de tipul i , ii , iii , iv etc. ( cifre romane mici );
* " 1 " pentru ordonare de tipul 1 , 2 , 3 , 4 etc. ( cifre arabe - optiune prestabilita );

Listele ordonate pot fi imbricate intre ele sau cu liste neordonate.

O lista de meniuri este un bloc delimitat de etichete corespondente <menu>...</menu>. Fiecare element al listei este initiat de eticheta <li> ( list item ). Cele mai multe browsere afiseaza lista de meniuri ca pe o lista neordonata.

O lista de directoare este un bloc delimitat de etichete corespondente <dir>...</dir>(de la " director "). Fiecare element al listei este initiat de eticheta <li> ( list item ). Cele mai multe browsere afiseaza lista de directoare ca pe o lista neordonata.

Utilizari speciale ale listelor

Daca intr-o lista, in loc de elementele acesteia introduse prin <li>, se insereaza un bloc de text, aceasta va fi indentat ( intocmai ca elementele unei liste).

**I.2.CSS**

CSS (Cascading Style Sheets) este un standard pentru formatarea elementelor unui document [HTML](http://ro.wikipedia.org/wiki/HTML). Stilurile se pot atașa elementelor HTML prin intermediul unor fișiere externe sau în cadrul documentului, prin elementul<style> și/sau atributul style. CSS se poate utiliza și pentru formatarea elementelor [XHTML](http://ro.wikipedia.org/wiki/XHTML), [XML](http://ro.wikipedia.org/wiki/XML) și [SVGL](http://ro.wikipedia.org/w/index.php?title=SVGL&action=edit&redlink=1).

CSS3 reprezintă un upgrade ce aduce câteva atribute noi și ajută la dezvoltarea noilor concepte in webdesign.

Unele dintre cele mai importante segmente (module) noi adăugate acestui standard pentru formatarea elementelor HTML aduc un plus considerabil in dezvoltarea activități webdesign.

De exemplu, puteți decide ca titlul principal de pe pagina dvs. să fie afișat ca text roșu mare. Următorul cod arată o regulă CSS foarte simplă care ar atinge stilul descris mai sus:

* În exemplul de mai sus, regula CSS se deschide cu un selector . Aceasta selectează elementul HTML pe care îl vom stila. În acest caz, stilăm titlurile de nivelul unu (<h1>).

h1 {

    color: red;

    font-size: 5em;

}

* Avem apoi un set de acolade { }.
* În interiorul acoladelor vor fi una sau mai multe declarații, care iau forma perechilor de proprietate și valoare. Specificăm proprietatea (culoarea în exemplul de mai sus) înainte de două puncte și specificăm valoarea proprietății după două puncte (roșu în acest exemplu).
* Acest exemplu conține două declarații, una pentru culoare și cealaltă pentru dimensiunea fontului. Fiecare pereche specifică o proprietate a elementului (ele) pe care le selectăm (<h1> în acest caz), apoi o valoare pe care am dori să o dăm proprietății.

Proprietățile CSS au valori permise diferite, în funcție de proprietatea specificată. În exemplul nostru, avem proprietatea culoare, care poate lua diferite valori de culoare. Avem și proprietatea font-size. Această proprietate poate lua ca valoare unități de diferite dimensiuni.

O foaie de stil CSS va conține multe astfel de reguli, scrise una după alta.

h1 {

    color: red;

    font-size: 5em;

}

p {

    color: black;

}

**I.3. JAVASCRIPT**

**JavaScript** (**JS**) este un [limbaj de programare](http://ro.wikipedia.org/wiki/Limbaj_de_programare) [orientat obiect](http://ro.wikipedia.org/wiki/Programare_orientat%C4%83_pe_obiecte) bazat pe conceptul [prototipurilor](http://ro.wikipedia.org/w/index.php?title=Prototip_(POO)&action=edit&redlink=1). Este folosit mai ales pentru introducerea unor funcționalități în paginile web, codul Javascript din aceste pagini fiind rulat de către [browser](http://ro.wikipedia.org/wiki/Browser_web). Limbajul este binecunoscut pentru folosirea sa în construirea[siturilor web](http://ro.wikipedia.org/wiki/Website" \o "Website), dar este folosit și pentru acesul la obiecte încastrate (embedded objects) în alte aplicații. A fost dezvoltat inițial de către [Brendan Eich](http://ro.wikipedia.org/wiki/Brendan_Eich" \o "Brendan Eich) de la[Netscape Communications Corporation](http://ro.wikipedia.org/wiki/Netscape_Communications_Corporation" \o "Netscape Communications Corporation) sub numele de Mocha, apoi LiveScript, și denumit în final JavaScript.

În ciuda numelui și a unor similarități în sintaxă, între JavaScript și limbajul Java nu există nicio legătură. Ca și [Java](http://ro.wikipedia.org/wiki/Java" \o "Java), JavaScript are o sintaxă apropiată de cea a [limbajului C](http://ro.wikipedia.org/wiki/Limbajul_de_programare_C), dar are mai multe în comun cu [limbajul Self](http://ro.wikipedia.org/w/index.php?title=Limbajul_Self&action=edit&redlink=1) decât cu [Java](http://ro.wikipedia.org/wiki/Limbajul_Java).

Cea mai des întâlnită utilizare a JavaScript este în scriptarea [paginilor web](http://ro.wikipedia.org/wiki/Pagin%C4%83_web). Programatorii web pot îngloba în paginile HTML script-uri pentru diverse activități cum ar fi verificarea datelor introduse de utilizatori sau crearea de meniuri și alte efecte animate.

Browserele rețin în memorie o reprezentare a unei pagini web sub forma unui [arbore](http://ro.wikipedia.org/wiki/Arbore) de [obiecte](http://ro.wikipedia.org/wiki/Obiect_(informatic%C4%83)) și pun la dispoziție aceste obiecte script-urilor JavaScript, care le pot citi și manipula. Arborele de obiecte poartă numele de [Document Object Model](http://ro.wikipedia.org/w/index.php?title=Document_Object_Model&action=edit&redlink=1)sau [DOM](http://ro.wikipedia.org/w/index.php?title=DOM&action=edit&redlink=1). Există un standard [W3C](http://ro.wikipedia.org/wiki/W3C) pentru DOM-ul pe care trebuie să îl pună la dispoziție un browser, ceea ce oferă premiza scrierii de script-uri [portabile](http://ro.wikipedia.org/w/index.php?title=Portabilitate&action=edit&redlink=1), care să funcționeze pe toate browserele. În practică, însă, standardul W3C pentru DOM este incomplet implementat. Deși tendința browserelor este de a se alinia standardului W3C, unele din acestea încă prezintă incompatibilități majore, cum este cazul [Internet Explorer](http://ro.wikipedia.org/wiki/Internet_Explorer).

O tehnică de construire a paginilor web tot mai întâlnită în ultimul timp este [AJAX](http://ro.wikipedia.org/wiki/AJAX), abreviere de la „Asynchronous JavaScript and XML”. Această tehnică constă în executarea de cereri [HTTP](http://ro.wikipedia.org/wiki/HTTP) în fundal, fără a reîncărca toată pagina web, și actualizarea numai anumitor porțiuni ale paginii prin manipularea DOM-ului paginii. Tehnica AJAX permite construirea unor interfețe web cu timp de răspuns mic, întrucît operația (costisitoare ca timp) de încărcare a unei pagini HTML complete este în mare parte eliminată.

**I.4. FIREBASE (BAZA DE DATE)**

Firebase este o platforma de dezvoltare de aplicații susținut de Google care le permite dezvoltatorilor să dezvolte aplicații iOS, Android și web. Firebase oferă instrumente pentru urmărirea analizelor, raportarea și remedierea blocărilor aplicațiilor, crearea de marketing și experimente de produs.

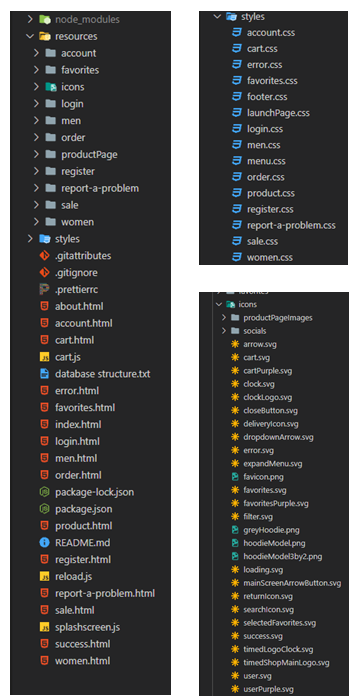
Firebase oferă o serie de servicii, inclusiv:

* Autentificare – Firebase Authentication facilitează dezvoltatorilor să construiască sisteme de autentificare sigure și îmbunătățește experiența de conectare și de integrare pentru utilizatori. Această caracteristică oferă o soluție completă de identitate, care acceptă conturi de e-mail și parole, autentificare telefonică, precum și autentificare Google, Facebook, GitHub, Twitter și multe altele.
* Baza de date în timp real – Firebase Realtime Database este o bază de date NoSQL găzduită în cloud, care permite stocarea și sincronizarea datelor între utilizatori în timp real. Datele sunt sincronizate între toți clienții în timp real și sunt încă disponibile atunci când o aplicație este offline.

**CAPITOLUL II. PREZENTARE SITE**

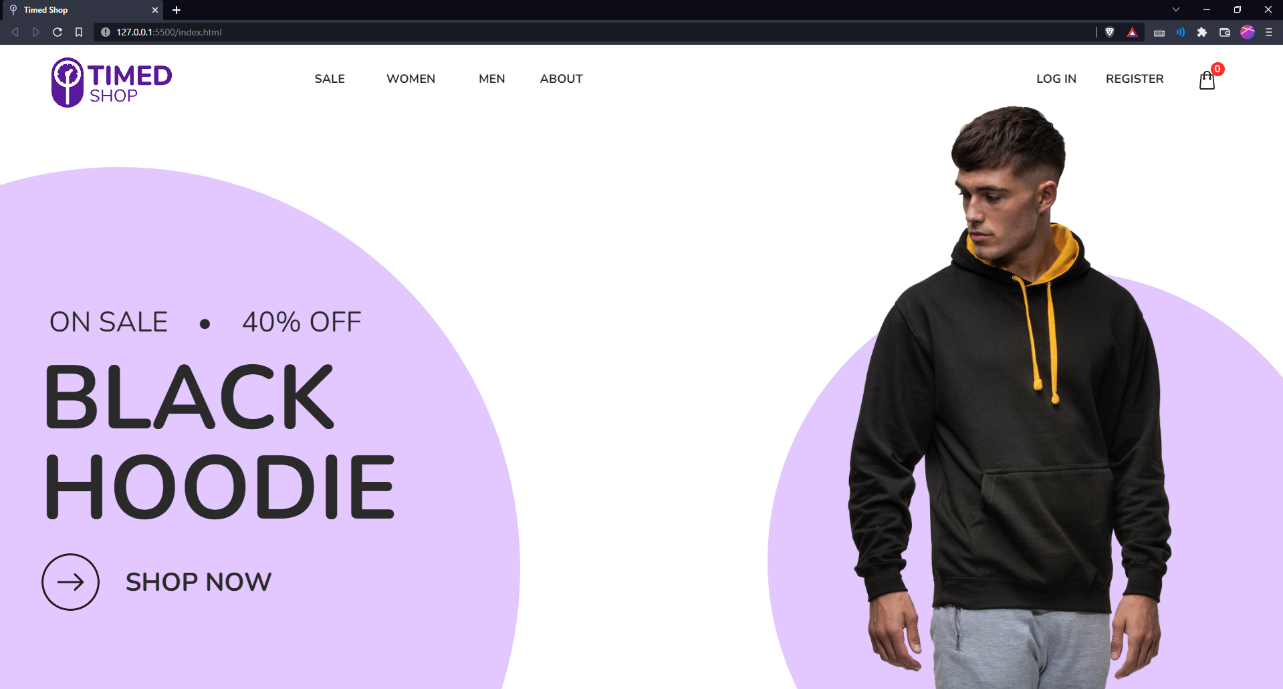
**II.1. STRUCTURA SITE-ULUI**

In urmatoarele imagini se poate vedea structura site-ului, modul in care sunt aranjate fisierele si folderele proiectului.

In imaginea din stanga putem vedea folderul `node\_modules`. Acest folder contine toate modulele necesare de JavaScript si toate librariile importate de mine.

In folderul `resources` avem structura site-ului in sine, toate fisierele acestuia. In cea de a doua jumatate a imaginii se regasesc toate fisierele de tip HTML, fiecare reprezentand cate o pagina diferita a site-ului. Tot aici, avem si cateva fisiere JavaScript care sunt utilizate in interiorul fiecarei pagini. In prima jumatate a imaginii avem folderul corespondent fiecarei pagini HTML. In acestea regasim toate fisierele JavaScript necesare pentru rularea paginii respective. In imaginea din dreapta-sus, putem vedea folderul `styles` ce contine fiecare fisier CSS al fiecarei pagini HTML. Iar in imaginea din dreapta-jos se afla folderul `icons` ce contine fiecare imagine/iconita folosita de catre site. Toate iconitele sunt create de mine in Adobe Photoshop sau Adobe Illustrator.

**II.2. PAGINA PRINCIPALA**

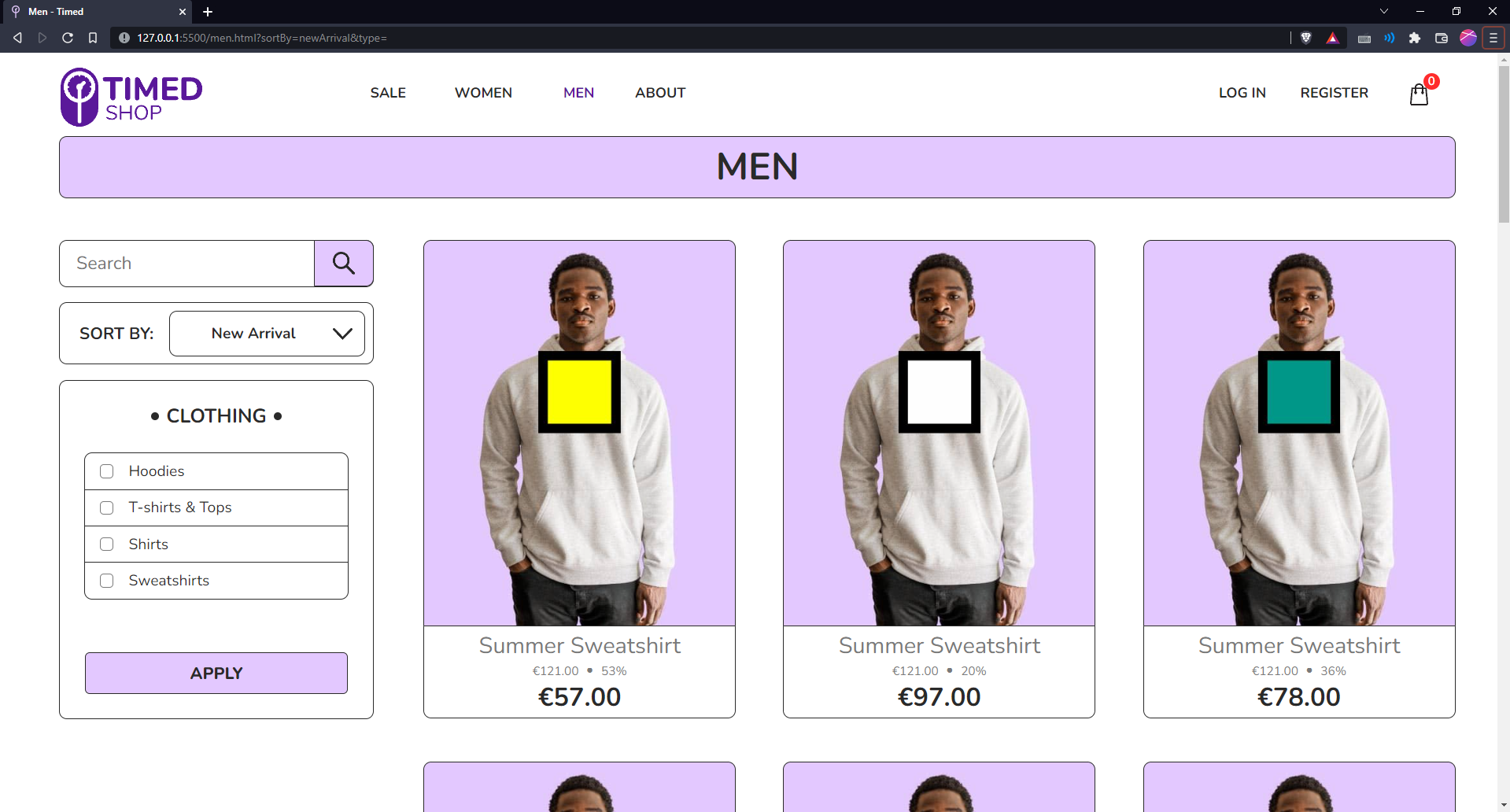


La accesarea site-ului, utilizatorul este intampinat de pagina principala. Aici putem vedea sigla website-ului, meniul de navigatie si o imagine cu informatii despre reducerea la acel produs.

Meniul de navigatie este alcatuit din:

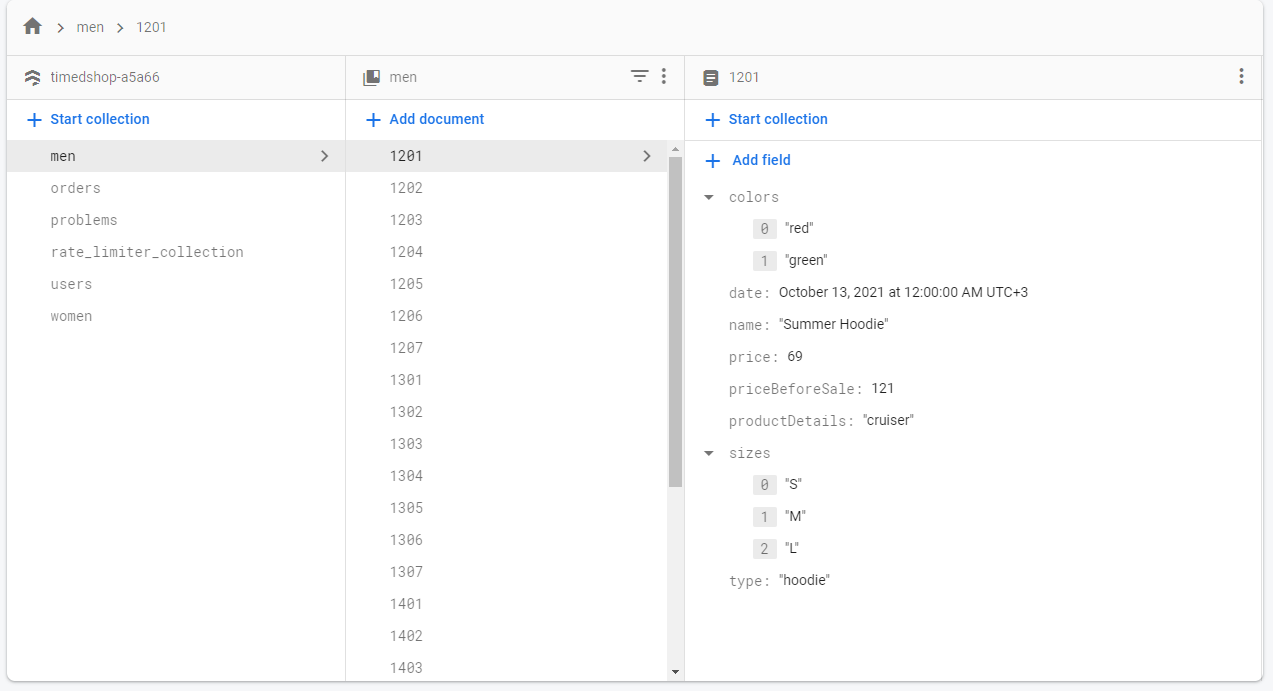
* 1. Pagina SALE
  2. Pagina WOMEN
  3. Pagina MEN
  4. Pagina ABOUT
  5. Pagina LOG IN
  6. Pagina REGISTER
  7. Cosul de cumparaturi impreuna cu numarul de produse adaugate in acesta

**II.3. PAGINA MEN**

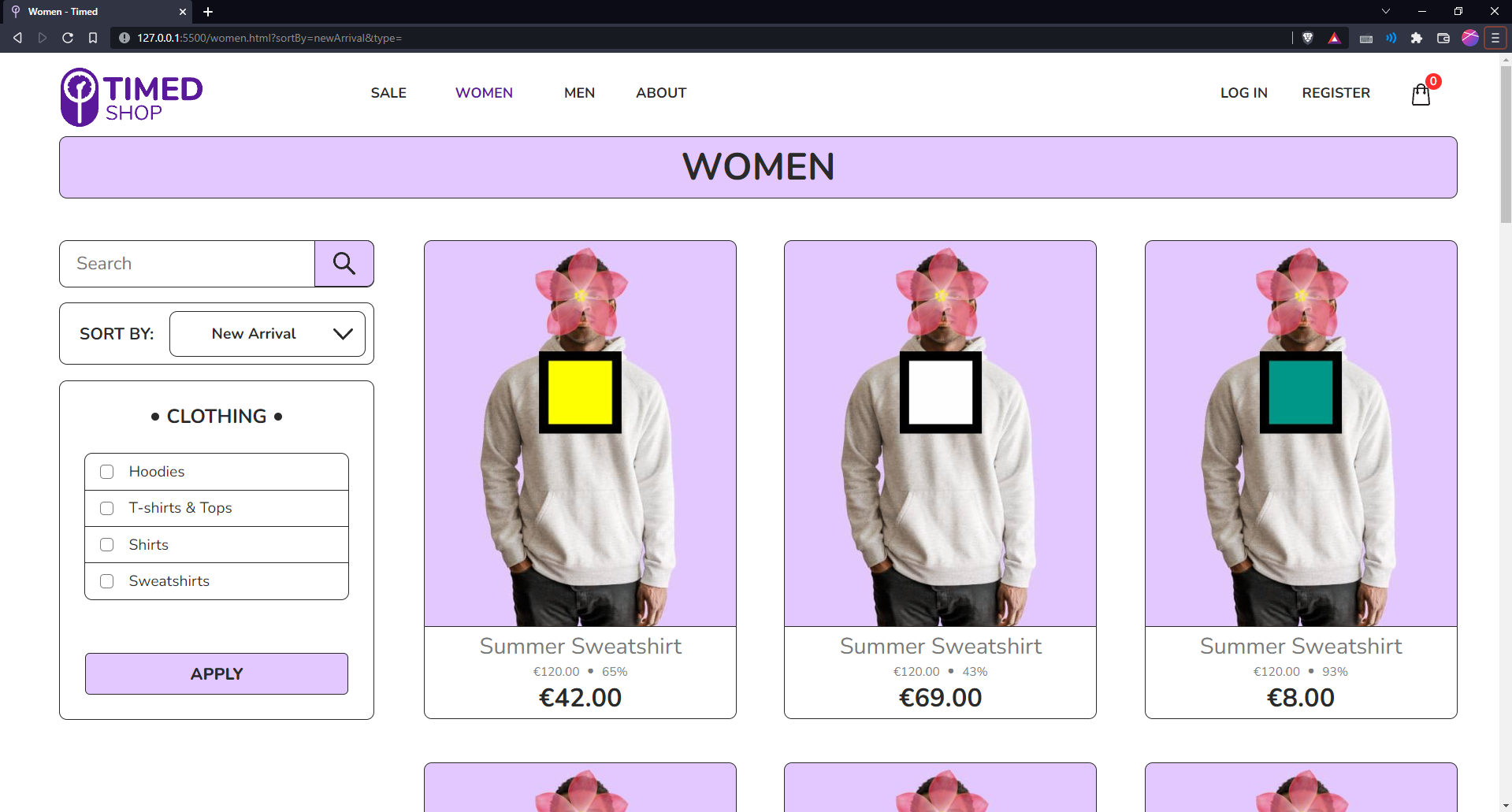


In cadrul pagini MEN se afla catalogul de produse de la sectiunea BARBATI. Aici avem o pagina de tip scroll infinit, dupa afisarea a 18 produse si atingerea sfarsitului de pagina, mai sunt incarcate alte 18 produse. In stanga, se afla o bara de cautare dinamica prin produsele din catalagoul de la barbati, impreuna cu un filtru pentru tipul de haine. Toate acestea sunt realizate cu ajutorul JavaScript-ului si a bazelor de date Firebase.

Fiecare card cu produsul contine cate 2 poze, numele produsului, pretul inainte si dupa reducere si marimile si culorile valabile in stoc. Toate produsele sunt imagini si blocuri de date preluate prin intermediul unor apeluri eficiente de API-uri din baza de date FIREBASE. Structura din baza de date este urmatoarea:

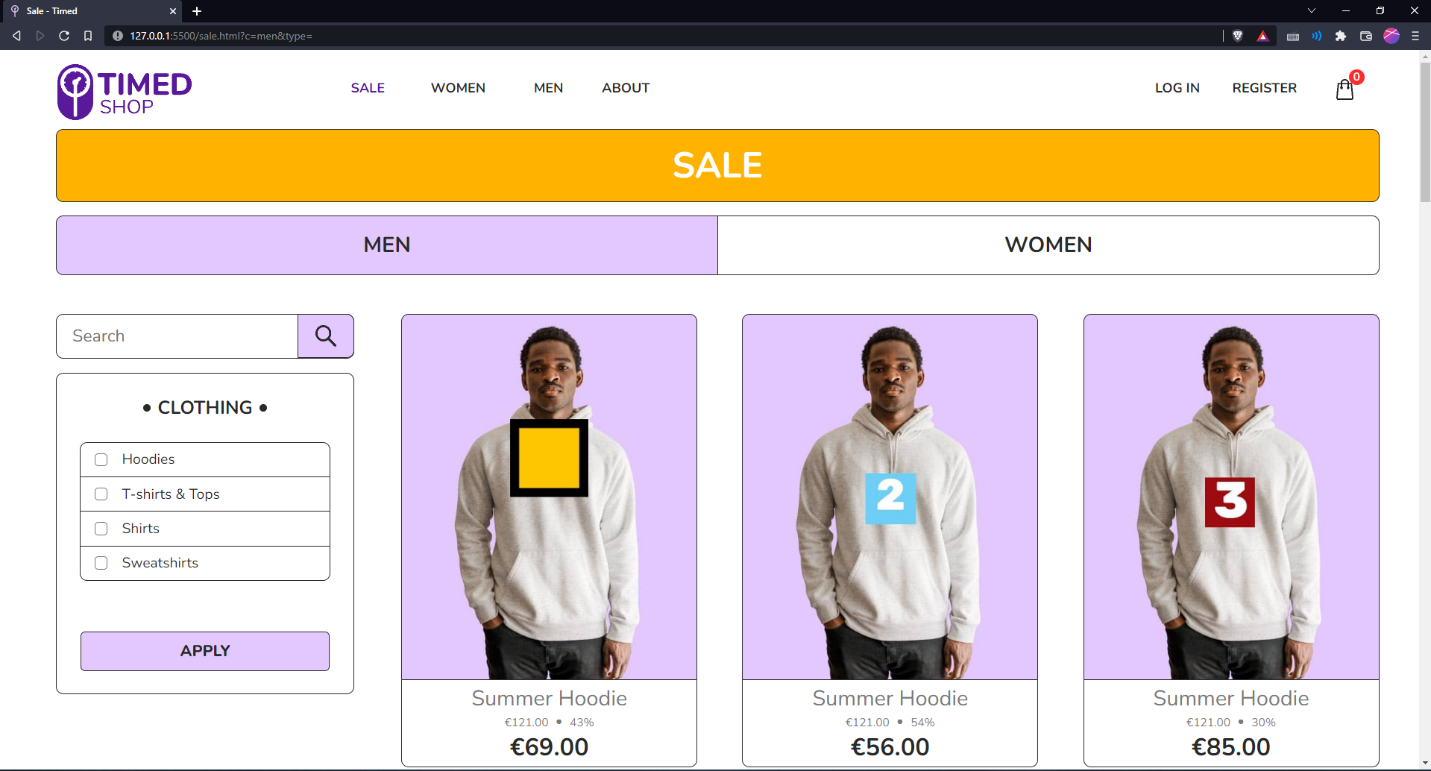


**II.4. PAGINA WOMEN**



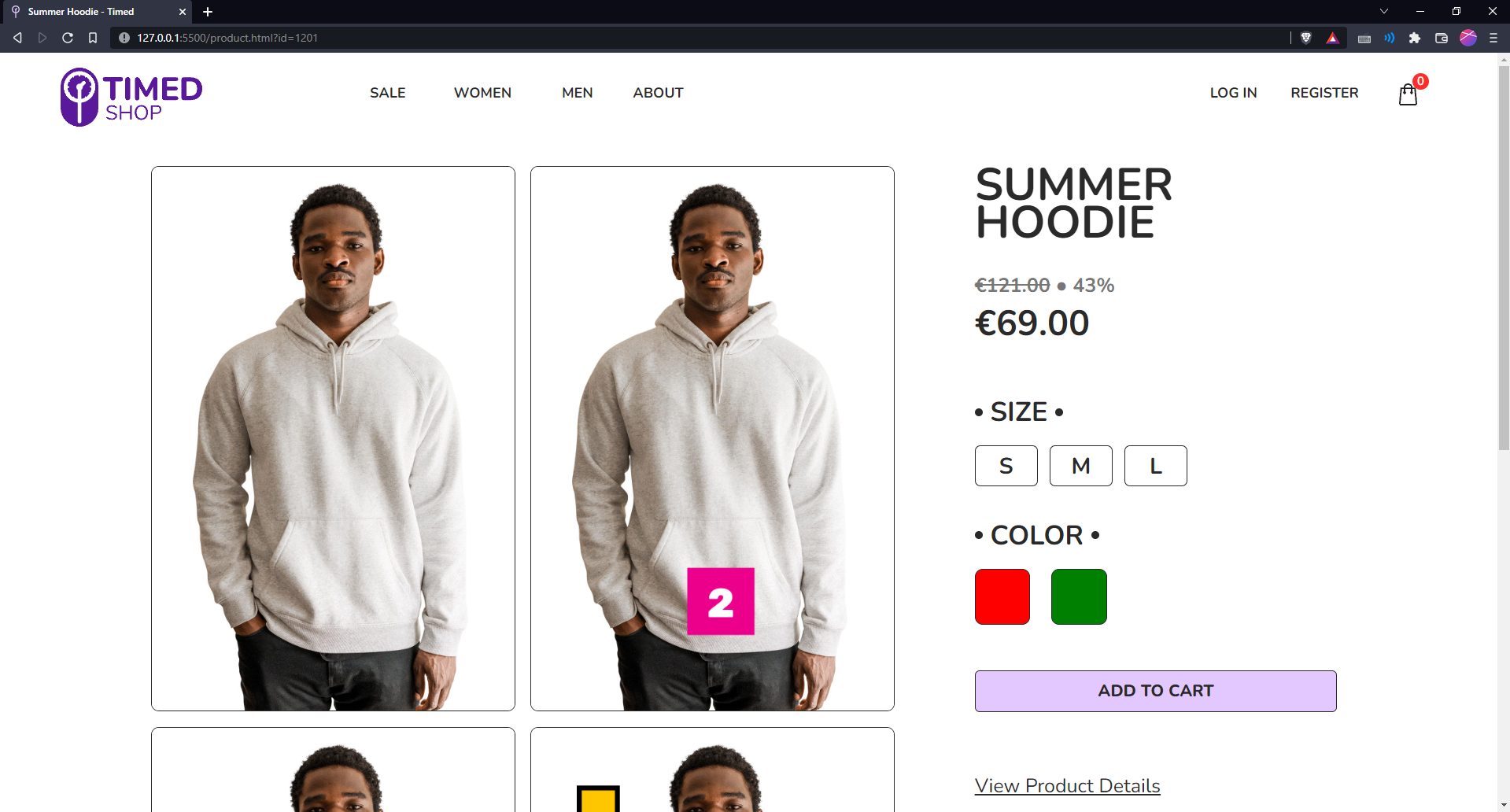
In cadrul pagini WOMEN se afla catalogul de produse de la sectiunea FEMEI. Structura paginii este exact ca cea de la MEN, doar baza de date difera cu produse pentru femei.

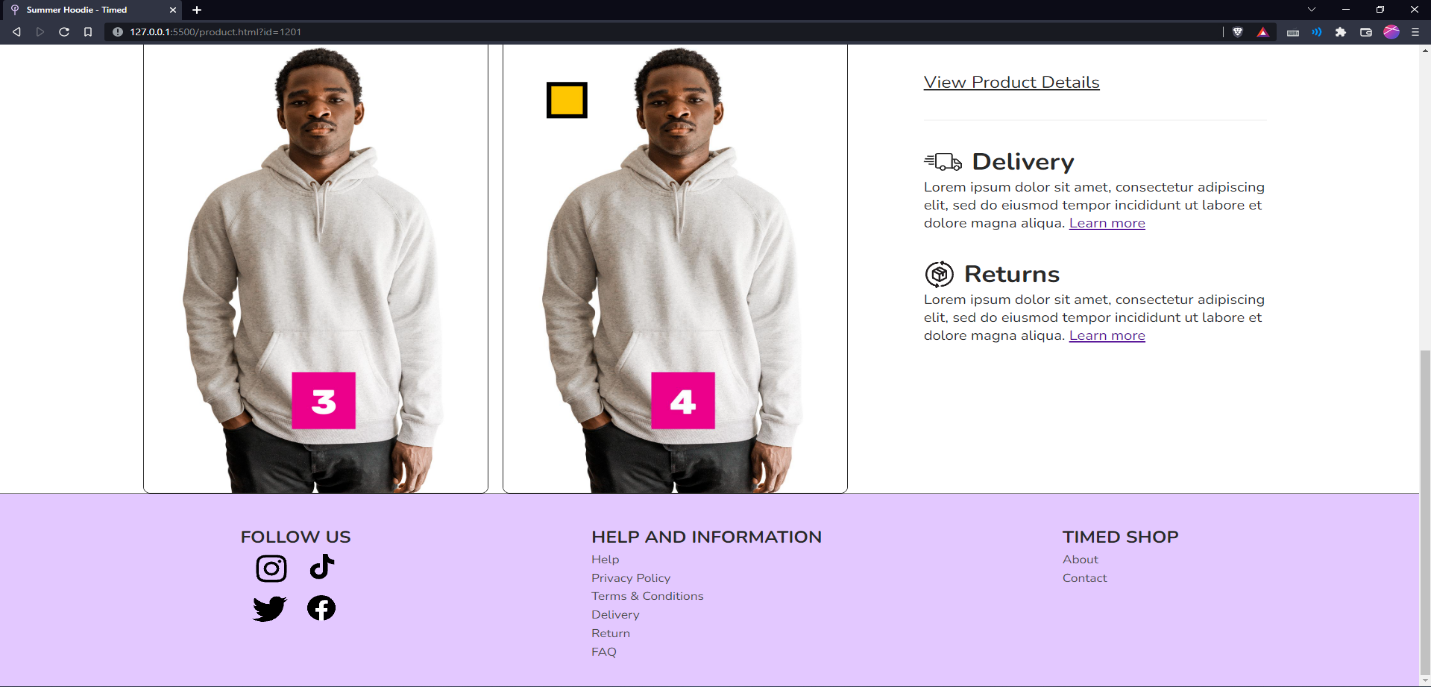
**II.5. PAGINA SALE**



Pagina SALE are aceeasi structura ca si celelalte doua pagini, WOMEN si MEN, doar ca desparte in doua parti cataloagele, alegand doar produsele care sunt la reducere. Ambele sectiuni au cate un buton pentru filtrarea produselor la reducere in functie de gen.

**II.6. PAGINA PRODUSULUI**

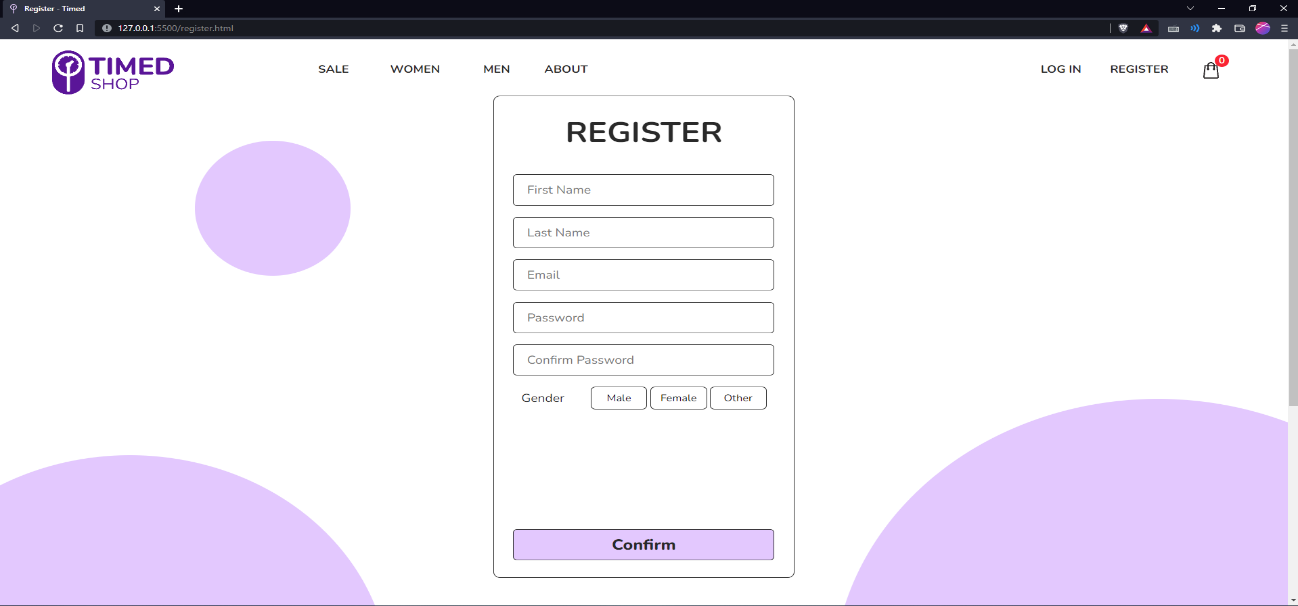




Pagina produsului contine 4 imagini diferite, de calitate mare, pentru ca utilizatorul sa isi poata vedea mai in detaliu produsul. In dreapta imaginilor se afla:

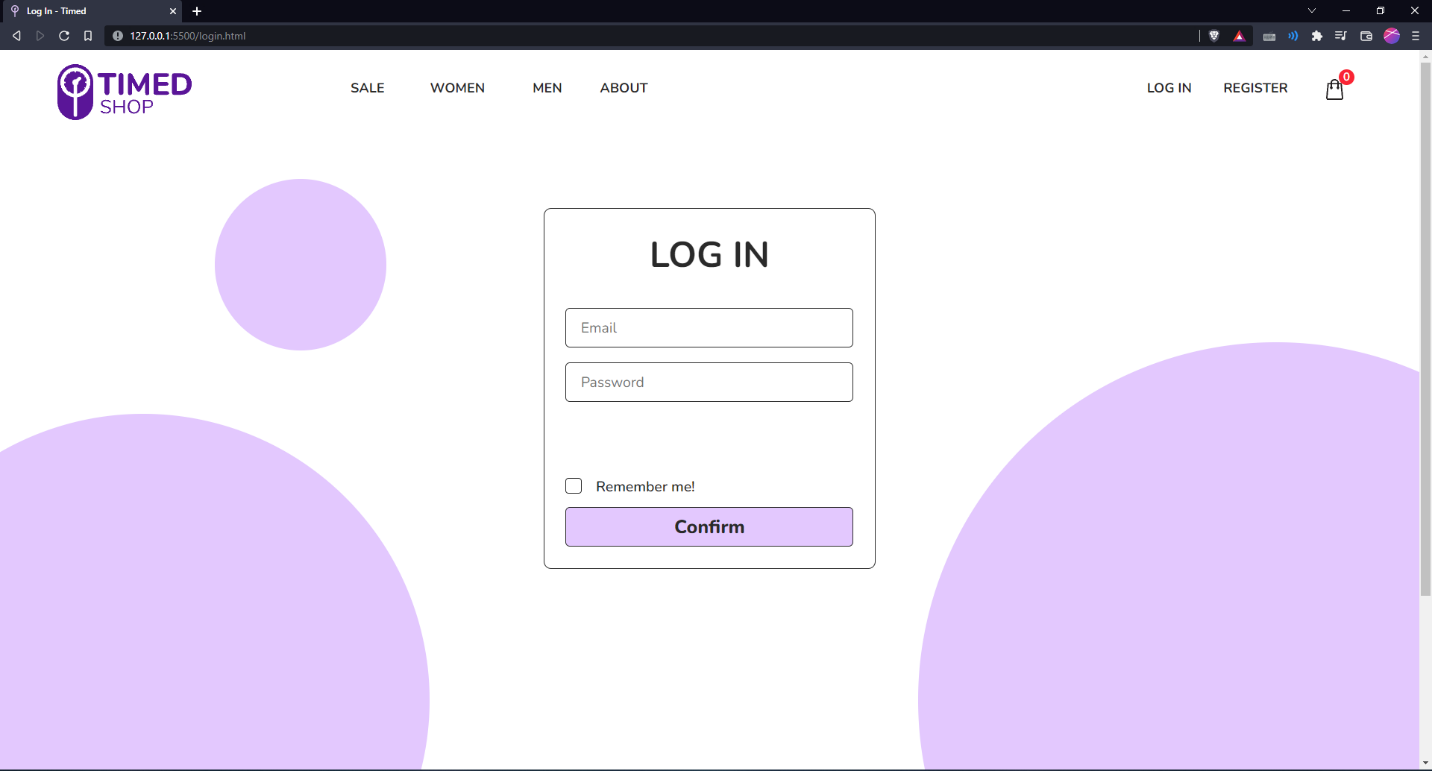
* Denumirea produsului
* Pretul produsului inainte de reducere
* Procentul de reducere
* Pretul produsului dupa reducere
* Butoane cu marimile si culorile valabile in stoc, ce pot fi selectate pentru adaugarea in cos
* Butonul de adaugare in cos
* Un buton ce afiseaza detalii referitoare la calitatea materialului din care este realizat produsul
* Informatii referitoare la livrare si retur

**II.7. PAGINA REGISTER**



Pagina REGISTER permite clientilor sa isi creeze conturi pe platforma, lucru ce le permite sa isi salveze produse la FAVORITE si sa-si vada istoricul de comenzi. In aceasta pagina putem sa vedem un formular ce permite completarea a 5 campuri text si un camp de tip selectie. In urma completarii acestor 6 campuri, datele utilizatorului sunt salvate in baza de date FIREBASE.

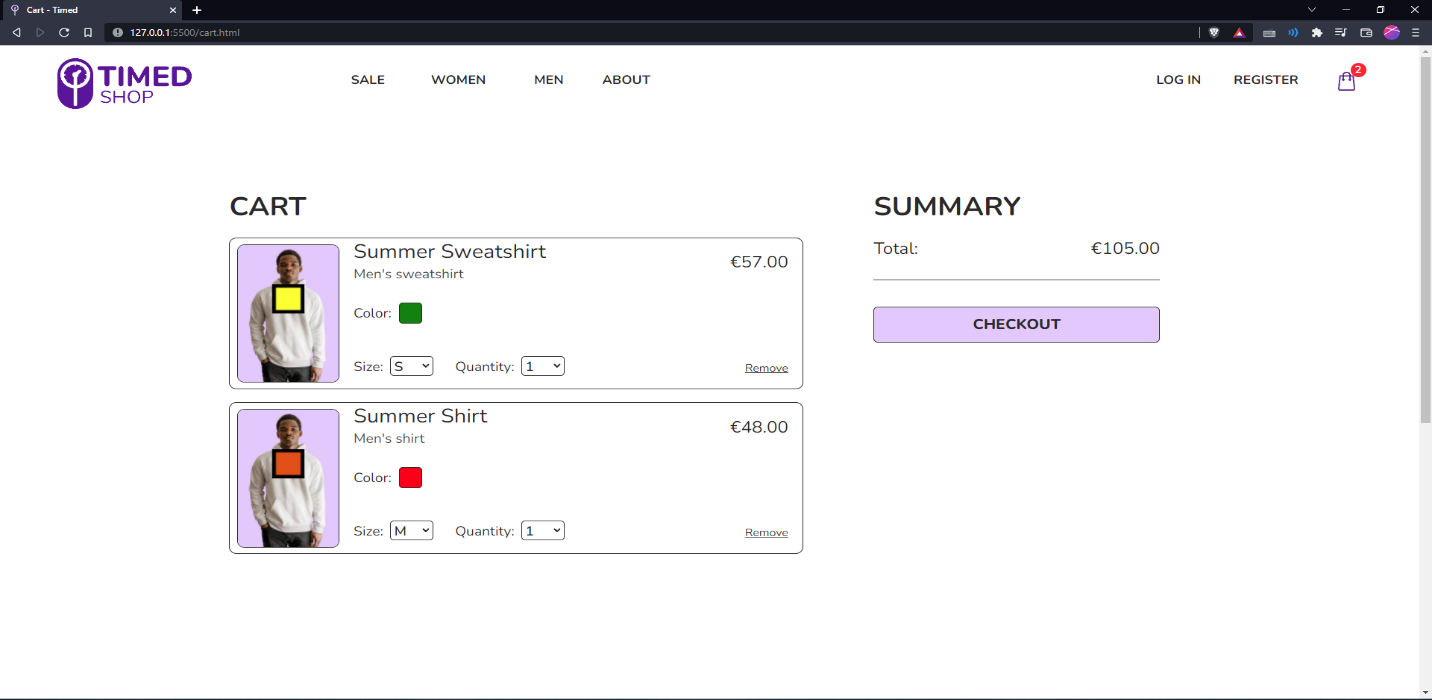
**II.8. PAGINA LOG IN**



Pagina LOG IN permite clientilor, care au deja un cont inregistrat in baza de date, sa isi acceseze acele conturi.

**II.9. PAGINA CART**

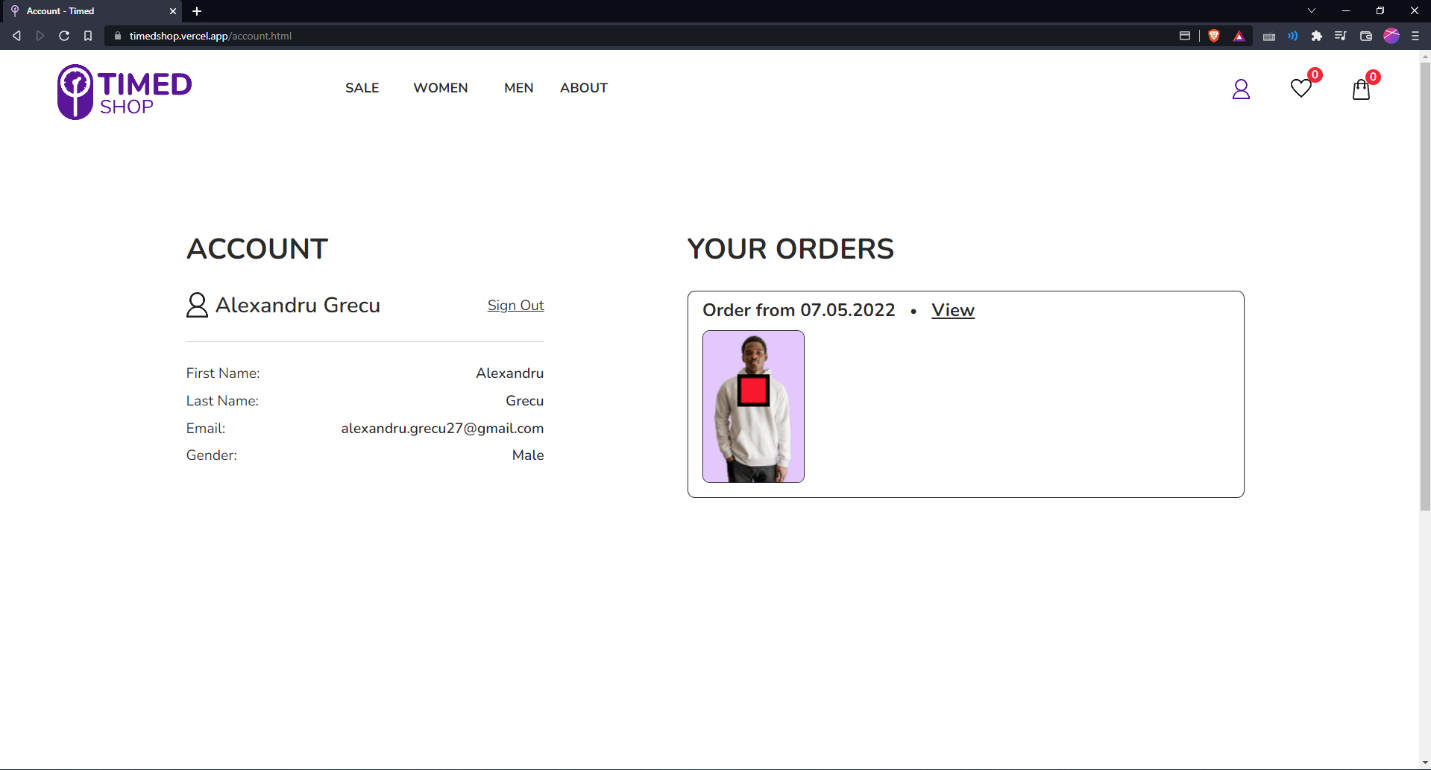
**(COSUL DE CUMPARATURI)**



Pagina CART afiseaza produsele adaugate de client in cosul propriu de cumparaturi si permite modificarea cantitatii si a marimii inainte de CHECKOUT. In stanga se afla produsele din cos, afisate cu imaginile si informatiile corespunzatoare. In dreapta este afisat pretul total al produselor, impreuna cu butonul de CHECKOUT. Dupa apasarea butonului de CHECKOUT suntem redirectionati catre pagina cu plata. In urma completarii comenzii, cosul utilizatorului se goleste si i se adauga comanda in istoricul contului.

**II.10. PAGINA ACCOUNT**

**(CONTUL UTILIZATORULUI)**



Pagina ACCOUNT (contul utilizatorului) afiseaza informatiile contului precum datele personale si istoricul de comenzi. Aici ai posibilitatea sa te deconectezi de pe contul curent.

**II.11. PREZENTARE SURSA COD**

Mai jos avem sursa HTML a paginii principale:

<html lang="en">

    <head>

        <link rel="preconnect" href="https://fonts.googleapis.com">

        <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

        <link href="https://fonts.googleapis.com/css2?family=Nunito:ital,wght@0,200;0,300;0,400;0,600;0,700;0,800;0,900;1,200;1,300;1,400;1,600;1,700;1,800;1,900&display=swap" rel="stylesheet">

        <link rel="icon" href="resources/icons/favicon.png" />

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

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

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

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

        <meta charset="UTF-8" />

        <title>Account - Timed</title>

        <script src="reload.js"></script>

        <script src="resources/productPage/productDetails.js"></script>

        <script src="https://www.gstatic.com/firebasejs/8.10.0/firebase-app.js"></script>

        <script src="https://www.gstatic.com/firebasejs/8.10.0/firebase-firestore.js"></script>

        <script src="https://www.gstatic.com/firebasejs/8.10.0/firebase-auth.js"></script>

        <script src="https://js.stripe.com/v3/"></script>

    </head>

    <body>

        <div id="bigdiv">

            <div id="THEDIV">

                <div id="TimedShopMainLogo">

                    <a href="./index.html"><img src="resources/icons/timedShopMainLogo.svg" alt="TIMED SHOP" /></a>

                </div>

                <div id="clockLogo">

                    <a href="./index.html"><img src="/resources/icons/clockLogo.svg" alt="TIMED SHOP"/></a>

                </div>

                <div id="menu">

                    <div id="row">

                        <a href="account.html" id="account" class="icons">

                            <img

                                alt="ACCOUNT"

                                src="/resources/icons/userPurple.svg"

                                onmouseover="this.src='resources/icons/userPurple.svg'"

                                onmouseout="this.src='resources/icons/user.svg'"

                            />

                        </a>

                        <a href="favorites.html" id="favorites" class="icons">

                            <img

                                class="icon-badge-icon-favorites"

id="favoritesIMG"

                                alt="FAVORITES"

                                src="/resources/icons/favorites.svg"

                                onmouseover="this.src='resources/icons/favoritesPurple.svg'"

                                onmouseout="this.src='resources/icons/favorites.svg'"

                            />

                            <div id="favoritesCount" class="icon-badge-favorites">6</div>

                        </a>

                        <a href="cart.html" id="cart" class="icon-badge-container" class="icons">

                            <img

                                class="icon-badge-icon"

                                alt="CART"

                                src="/resources/icons/cart.svg"

                                onmouseover="this.src='resources/icons/cartPurple.svg'"

                                onmouseout="this.src='resources/icons/cart.svg'"

                            />

                            <div id="cartCount" class="icon-badge">6</div>

                        </a>

                        <a href="javascript:void(0);" id="expandMenu" class="icon" onclick="expandMenu()">

                            <img src="/resources/icons/expandMenu.svg">

                        </a>

                    </div>

                    <nav class="navMenu" id="navMenu">

                        <a href="sale.html?c=men&type=">SALE</a>

                        <a href="women.html?sortBy=newArrival&type=" id="women">WOMEN</a>

                        <a href="men.html">MEN</a>

                        <a>ABOUT</a>

                        <a href="register.html" id="register">REGISTER</a>

                        <a href="login.html" id="LogIn">LOG IN</a>

                    </nav>

                </div>

                <div id="mainDiv">

                    <div id="leftDiv">

                        <a id="accountTitle">ACCOUNT</a>

                        <div id="accountImgDiv">

                            <img src="resources/icons/user.svg" alt="userIcon" id="accountUserIcon">

                            <a id="accountName"></a>

                            <a id="signOut" onclick="signOut()">Sign Out</a>

                        </div>

                        <div id="divider"></div>

                        <div id="accountInfo">

                            <div id="firstNameDiv">

                                <a id="firstNameText">First Name: </a>

                                <a id="firstName"></a>

                            </div>

                            <div id="lastNameDiv">

                                <a id="lastNameText">Last Name: </a>

                                <a id="lastName"></a>

                            </div>

                            <div id=emailDiv>

                                <a id="emailText">Email: </a>

                                <a id="email"></a>

                            </div>

                            <div id="genderDiv">

                                <a id="genderText">Gender: </a>

                                <a id="gender"></a>

                            </div>

                        </div>

                    </div>

                    <div id="orders">

                        <a id="ordersTitle">YOUR ORDERS</a>

                        <div id="ordersDiv">

                        </div>

                    </div>

                </div>

                <div id="footer">

                    <div id="followUsDiv">

                        <a id="followUsText"> FOLLOW US </a>

                        <div id="iconsDiv">

                            <div id="iconsDivTop">

                                <a id="instagram" href="https://www.instagram.com">

                                    <img

                                        src="resources/icons/socials/instagram.svg"

                                        alt="instagram"

                                    />

                                </a>

                                <a id="tiktok">

                                    <img

                                        src="resources/icons/socials/tiktok.svg"

                                        alt="tiktok"

                                    />

                                </a>

                            </div>

                            <div id="iconsDivBottom">

                                <a id="twitter">

                                    <img

                                        src="resources/icons/socials/twitter.svg"

                                        alt="twitter"

                                    />

                                </a>

                                <a id="facebook">

                                    <img

                                        src="resources/icons/socials/facebook.svg"

                                        alt="facebook"

                                    />

                                </a>

                            </div>

                        </div>

                    </div>

                    <div id="helpAndInfoDiv">

                        <a id="helpAndInfoText"> HELP AND INFORMATION </a>

                        <div id="HILinks">

                            <a class="alink">Help</a>

                            <a class="alink">Privacy Policy</a>

                                        alt="twitter"

                                    />

                                </a>

                                <a id="facebook">

                                    <img

                                        src="resources/icons/socials/facebook.svg"

                                        alt="facebook"

                                    />

                                </a>

                            </div>

                        </div>

                    </div>

                    <div id="helpAndInfoDiv">

                        <a id="helpAndInfoText"> HELP AND INFORMATION </a>

                        <div id="HILinks">

                            <a class="alink">Help</a>

                            <a class="alink">Privacy Policy</a>

                            <a class="alink">Terms & Conditions</a>

                            <a class="alink">Delivery</a>

                            <a class="alink">Return</a>

                            <a class="alink">FAQ</a>

                        </div>

                    </div>

                    <div id="timedShopDiv">

                        <a id="timedShopText">TIMED SHOP</a>

                        <div id="timedShopLinks">

                            <a class="alink">About</a>

                            <a class="alink">Contact</a>

                        </div>

                    </div>

                </div>

            </div>

            <div id="loading">

                <div class="wrapper">

                    <svg class="hourglass" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 120 206" preserveAspectRatio="none">

                    </svg>

                </div>

            </div>

        </div>

        <script>

            function loader(loaded)

            {

                if(loaded)

                {

                    setTimeout(()=>{

                        document.getElementById("loading").style.display = "none";

                        document.getElementById("THEDIV").style.display =

"block";

                    }, 1000);

                }

                else

                {

                    setTimeout(()=>{

                        document.getElementById("THEDIV").style.display = "none";

                        document.getElementById("loading").style.display = "flex";

                    }, 1000);

                }

            }

            loader(false);

        </script>

        <script>

            // Initialize Firebase

            // TODO: Replace with your project's customized code snippet

            var config = {

                \*\*\*\*\*

            };

            firebase.initializeApp(config);

            const db = firebase.firestore();

            let cartCount = document.getElementById("cartCount");

            var signOutPressed = false;

            var w = window.innerWidth

                || document.documentElement.clientWidth

                || document.body.clientWidth;

            firebase.auth().onAuthStateChanged((user) => {

                if (user) {

                    var uid = user.uid;

                    console.log("logged in user: ", user.email);

                    document.getElementById("LogIn").setAttribute("style","display: none !important")

                    document.getElementById("register").setAttribute("style","display: none !important")

                    document.getElementById("account").style.display = "flex";

                    if(w > 768 && w <= 1024)

                    {

                        document.getElementById("navMenu").setAttribute("style", "width:50vw !important");

                    }

                    document.getElementById("favorites").style.display = "inline";

                    db.collection("users")

                        .doc(uid)

                        .get()

                        .then((data) => {

                            if(data.data().cart)

                            {

                                cartCount.innerText = data.data().cart.length;

                            }

                            else

                            {

                                cartCount.innerText = 0;

                            }

                            if(data.data().favorites)

                            {

                                favoritesCount.innerText = data.data().favorites.length;

                            }

                            else

                            {

                                favoritesCount.innerText = 0;

                            }

                        })

                } else {

                    // User is signed out

                    setTimeout(()=>{

                        if(!signOutPressed)

                            window.open("/error.html","\_self");

                    }, 500);

                    document.getElementById("account").setAttribute("style","display: none !important")

                    document.getElementById("favorites").setAttribute("style","display: none !important")

                    document.getElementById("LogIn").style.display = "flex";

                    console.log(w);

                    if(w > 768)

                    {

                        document.getElementById("row").setAttribute("style","margin-left: 0 !important; width: 10% !important");

                    }

                    document.getElementById("register").style.display = "flex";

                    if(localStorage.getItem("cartCount"))

                    {

                        document.getElementById("cartCount").innerText = localStorage.getItem("cartCount");

                    }

                    else

                    {

                        document.getElementById("cartCount").innerText = "0";

                    }

                }

            });

            function signOut() {

                signOutPressed = true;

                firebase

                    .auth()

                    .signOut()

                    .then(() => {

                        // Sign-out successful.

                        loader(false);

                        setTimeout(()=>{

                            window.open("/login.html","\_self");

                        }, 1000);

                    })

                    .catch((error) => {

                        window.open("/error.html","\_self");

                    });

            }

            function expandMenu() {

                console.log('coco')

                var x = document.getElementById("navMenu");

                if (x.className === "navMenu") {

                    x.className += " responsive";

                } else {

                    x.className = "navMenu";

                }

            }

        </script>

        <script src="/resources/account/account.js"></script>

    </body>

</html>

**REFERINȚE**

## BIBLIOGRAFIE

* Miloşescu, M., *Manual de informatica, clasa a IX-a,* Ed. Didactică şi Pedagogică, Bucureşti, 2005.
* Miloşescu, M., *Manual de informatica, clasa a X-a,* Ed. Didactică şi Pedagogică, Bucureşti, 2006.
* Miloşescu, M., *Manual de informatica, clasa a XI-a,* Ed. Didactică şi Pedagogică, Bucureşti, 2007.

**WEBOGRAFIE**

* <https://www.w3schools.com/html/>
* <https://www.w3schools.com/css/default.asp>
* <https://www.w3schools.com/js/default.asp>
* <https://firebase.google.com/>
* <https://firebase.google.com/products/auth>
* <https://firebase.google.com/docs/firestore>
* <https://firebase.google.com/docs/auth>