# 4.System features

## 4. Cerinte functionale

### 4.1 – Autentificare

Cerinta functionala 1.1:

Pagina de vizitare va oferi utilizatorilor 3 optiuni de acces: acces fara cont, logare cu un cont deja existent sau crearea unui cont nou.

Cerinta functionala 1.2:

Accesul fara cont va permite utilizatorilor sa vada produsele disponibile, sa le adauge in cosul de cumparaturi, sa utilizeze butonul de “favorite”.

Cerinta functionala 1.3:

Desi orice utilizator poate folosi cosul de cumparati, pentru a trece mai departe la procesul de plata, un cont este necesar

Cerinta functionala 1.4:

Utilizatorii care au deja un cont existent se pot loga cu acesta. Ei vor putea sa cumpere produsele din cosul de cumparaturi, sa isi modifice datele contului, sa isi adauge o adresa de livrare care poate fii folosita la orice achizitii, fara a mai fii nevoie de a o introduce.

Cerinta functionala 1.5:

In cazul in care utilizatorii au un cont creat, insa au uitat datele de logare, acestia pot folosi sistemul de resetare al parolei. Vor introduce adresa de email si vor primi un cod cu ajutorul caruia isi pot reseta parola

Cerinta functionala 1.6:

Daca un utilizator nu are un cont, acesta isi poate crea unul. Va trebui sa isi introduca adresa de email, un nume de utilizator si o parola, optional utilizatorul mai poate adauga si alte informatii dupa crearea contului.

### 4.2 – Produse

Cerinta functionala 2.1:

Produsele vor fii stocate intr-o baza de date de tip MySQL.

Cerinta functionala 2.2:

Produsele vor fii afisate pe website intr-un mod dynamic din baza de date.

Cerinta functionala 2.3:

Lista cu produsele aplicatiei poate fii accesata de orice tip de utilizator.

Cerinta functionala 2.4:

Produsele vor fii impartite in mai multe tipuri de categorii, pentru a putea fii gasite mai usor de catre utilizatori

Cerinta functionala 2.5:

Aplicatia va dispune de un system de cautare, unde utilizatorii pot cauta produsele dupa nume.

Cerinta functionala 2.6:

Fiecare produs poate fii accesat in mod individual. Pagina de vizualizare a unui produs va contine toate informatiile disponibile referitoare la produsul respectiv.

Cerinta functionala 2.7:

Pe pagina de vizualizare a unui produs, utilizatorii pot adauga produsul in cosul de cumparati, sau il pot adauga la “favorit”

### 4.3 – Cosul de cumparaturi

Cerinta functionala 3.1:

Cosul de cumparaturi va putea fii utilizat de orice tip de utilizator

Cerinta functionala 3.2:

Produsele pot fii adaugate in cosul de cumparaturi printr-un buton existent in pagina de vizualizare a unui produs.

Cerinta functionala 3.3:

Cosul de cumparaturi va contine o lista cu numele, pretul, discount-ul (daca este cazul), cantitatea produselor pe care utilizatorul doreste sa le achizitioneze

Cerinta functionala 3.4:

Cosul de cumparaturi va contine functii simple, dar utile pentru utilizatori precum: stergerea a unuia sau mai multe produse din cos, vizualizarea fiecarui produs pentru mai multe detalii.

Cerinta functionala 3.5:

Cosul de cumparaturi va contine un buton, prin care utilizatorii vor fii redirectionati la procesul de achizitie al produselor din cos, numai in cazul in care acestia au un cont existent.

### 4.4 – Achizitia produselor

Cerinta functionala 4.1:

Un utilizator trebuie sa aibe un cont pentru a putea accesa acest meniu.

Cerinta functionala 4.2:

Meniul va contine lista produselor pe care utilizatorul doreste sa le achizitioneze, pretul total al acestora, modalitatea de plata si adresa de livrare.

Cerinta functionala 4.3:

Pentru modalitatea de plata, utilizatorul poate alege intre a plati cu un card sau a plati cash la momentul livrarii produselor.

Cerinta functionala 4.4:

Adresa de livrare va putea fii extrasa fie din datele de profil ale utilizatorului, fie acesta poate adauga o alta adresa.

Cerinta functionala 4.5:

Dupa validarea datelor, achizitia va fii efectuata, utilizatorul va primii un email cu datele achiziei precum si data livrarii produselor.

### 4.5 – Panoul de admin

Cerinta functionala 5.1:

Un admin se poate loga ca un utilizator normal.

Cerinta functionala 5.2:

Crearea unui cont de admin se poate face fie direct din baza de date, fie dintr-un meniu cu utilizatorii, la care au acces doar adminii de nivel mare

Cerinta functionala 5.3:

Vor exista 2 tipuri de admini: admini level 1 care va putea realiza atributii normale de administrare precum adaugarea/stergerea/editarea produselor, suspendarea conturilor de utilizatori etc. si admin level 2 care pe langa atributiile adminului level 1 va putea sa acceseze un meniu cu utilizatori.

Cerinta functionala 5.4:

Vor exista 2 tipuri de admini: admini level 1 care va putea realiza atributii normale de administrare precum adaugarea/stergerea/editarea produselor, suspendarea conturilor de utilizatori etc. si admin level 2 care pe langa atributiile adminului level 1 va putea sa acceseze un meniu cu utilizatori.

Cerinta functionala 5.5:

Va exista un meniu cu produsele din baza de date, orice tip de admin poate sa adauge/modifice/stearga produse.

Cerinta functionala 5.6:

Va exista un meniu cu utilizatorii create in baza de date, orice tip de admin poate sa suspende diferite conturi de utilizatori, iar adminii de level 2 vor putea crea alti admini.

### 4.6. – Discount-uri si alte promotii

Cerinta functionala 6.1:

Va exista un meniu pe pagina principala in care anumite produse vor fii la reducere.

Cerinta functionala 6.2:

Orice utilizator poate vizualiza produsele aflate la reduceri, promotiile existente, promotiile care urmeaza sa fie adaugata, timpul ramas din reduceri/promotii.

Cerinta functionala 6.3:

Fiecare produs aflat la reducere va putea fii adaugat in cosul de cumparaturi sau la “favorit” prin doua butoane existente.

# 5. Alte cerinte non-functionale

## 5.1. Cerințe de performanță

Cerinţele din această secţiune oferă o specificaţie detaliată a interacţiunii cu utilizatorul cu software-ul şi măsurătorile plasate pe performanţa sistemului.

### 5.1.1. Cerinte non-functionale

Această secțiune include constrângerile de proiectare asupra software-ului cauzate de hardware.

Caracteristicile sistemelor de calcul pe care va fi instalat sistemul informatic sunt:

**-** **procesor** : 1 GHz sau mai rapid;

- **memorie** : cel putin 128 Mb;

- **spatiu pe hard disk** : minim 10 Mb.

-**network :** conexiune activa la internet

### 5.1.3. Caracteristică de căutare proeminentă

Descriere: Caracteristica de căutare ar trebui să fie proeminenta şi uşor de găsit pentru utilizator.

Dependinte: Niciuna

### 5.1.4. Utilizarea caracteristicii de căutare

Descriere: Diferitele opţiuni de căutare ar trebui să fie evidente, simplu şi uşor de înţeles.

Dependinte: Niciuna

### 5.1.5. Utilizarea rezultatului din lista

Descriere: Rezultatele afişate în vizualizarea listă ar trebui să fie uşor de utilizat şi de înţeles. Selectarea unui element din lista de rezultate ar trebui să ia doar un singur click. Pentru ca un utilizator să utilizeze cu ușurință vizualizarea listei.

Dependinte: Niciuna

### 5.1.6. Timp de raspuns

- Rapiditatea căutării

- Timpul de răspuns al unei căutări

Masurare: Măsurători obținute din 100 de căutări în timpul testării.

Obligatoriu: Nu mai mult de 5 secunde 100% din timp

Dorinţa: Nu mai mult de 2 secunde 100% din timp.

### 5.1.7. Fiabilitatea sistemului

- Toleranţa la defect a sistemului

- În cazul în care sistemul pierde conexiunea la internet, utilizatorul ar trebui să fie informat.

Masurare: Măsurători obținute din 10 ore de utilizare în timpul testelor.

Obligatoriu: 90% din timp.

Dorinţa: 100% din timp.

## 5.2. Cerinte de siguranta

-Nu se aplica

## 5.3. Cerinte de securitate

- Baze de date si informatiile utilizizatorilor sa fie protejate.

- Dacă un utilizator încearcă să se conecteze la portalul web cu un cont non-existent, atunci utilizatorul nu ar trebui să fie autentificat. Utilizatorul ar trebui să fie notificat despre esecul la logare.

- Dacă un administrator încearcă să se conecteze la portalul web cu un cont non-existent, atunci administratorul nu ar trebui să fie autentificat. Administratorul ar trebui să fie notificat despre esecul la logare.

- Atunci cand un utilizator se conecteaza acesta trebuie sa fie conectat ca si utilizator in nici un caz ca un admin.

- Daca logarea esueaza de mai multe ori, atunci IP-ul ar trebuii sa fie blocat o perioada de la functia de logare.

- Dacă un utilizator dorește să creeze un cont și numele de utilizator dorit este ocupat, utilizatorului i se va solicita să aleagă un alt nume de utilizator.

## 5.4. Cerinte privind caltitatea software-ului

### 5.4.1 Reguli de cod

1. Parantezele sunt folosite cu instrucțiunile (if, else, for, do and while), chiar dacă corpul este gol sau conține doar o singură instrucțiune.

2. Identare trebuie sa aiba cel putin 4 spatii

3. Un bloc gol sau construcție asemănătoare blocului poate fi în stil K & R . Alternativ, aceasta poate fi închisă imediat după ce este deschisă, fără caractere sau linie între ({}), cu excepția cazului în care face parte dintr-o instrucțiune cu mai multe blocuri (una care conține mai multe blocuri: if / else sau try / catch/finally)

4. Identarea blocului se face cu 2 spatii

5. O singura declaratie de variabila pe linie.

6. Limita coloanei : 100 caractere

7. Se vor folosi lini goale pentru a face codul mai usor de citit

8. Un nume de parametru format dintr-un singur caracter ar trebuii evitat in metodele publice.

9. Variabilele locale nu trebuiesc declarate in corpurile (if,for...).

Ex: for(int i=0;i<n;i++) gresit

10. Varaibilele locale se declara in apropierea locului unde se utilizeaza

**5.5. Reguli Bussiness**

Proiectul trebuie utilizat doar in scopuri pur didactice acesta nu trebuie monetizat, distribuit etc. , nu ne asumam nici o responsabilitate daca aceste reguli sunt incalcate.