**Specificații privind cerințele software(SRS)**

**Pentru Librarie**

**Studenti participanti:**

-Diaconu Ionut-Victor

-Chivu George-Daniel

**Facultate:**

Automatica,Calculatoare si Electronica

**Specializare:**

Calculatoare Romana

**Anul: IV**

**Semestrul: I**

2018-2019

Contents

[0.Revision history…………………………………………………………………………………………………………………](#_Toc509918349)………………….3

[1.Introducere 4](#_Toc509918350)

[1.1 Scopul 4](#_Toc509918351)

[1.2 Sugestii vizate si sugestii de citire 4](#_Toc509918352)

[1.3 Domeniul de aplicare al produsului……………………………………………………………………………..](#_Toc509918354)……………….4

[1.4 Referinte](#_Toc509918355)……………………………………………………………………………………………………………………………………….5

[2.Descriere generală 5](#_Toc509918356)

[2.1 Perspectiva produsului 5](#_Toc509918357)

[2.2 Functiile produsului 5](#_Toc509918358)

[2.3 Clasele si caracteristicile utilizatorului 5](#_Toc509918360)

[2.4 Mediul de functionare 6](#_Toc509918361)

[2.5 Constrangeri de proiectare si implementare 6](#_Toc509918363)

[2.6 Documentatia utilizatorului 6](#_Toc509918364)

[2.7 Ipoteze si independente 6](#_Toc509918365)

[3.Cerintele interfetei 6](#_Toc509918366)

[3.1 Interfata utilizator 7](#_Toc509918367)

[3.2 Interfata hardware 8](#_Toc509918368)

[3.3 Interfata software 8](#_Toc509918369)

[3.4 Interfata de comunicatii 9](#_Toc509918370)

[4. Cerinte functionale 10](#_Toc509918371)

[4.1 –Autentificare 10](#_Toc509918372)

[4.2 –Produse 11](#_Toc509918373)

[4.3–Cosul de cumparaturi 12](#_Toc509918374)

[4.4 –Acizitia Produselor 12](#_Toc509918376)

[4.5–Panoul de admin 13](#_Toc509918377)

[4.6 –Discount-uri si alte promotii 14](#_Toc509918378)

[5. Alte cerinte non-functionale 14](#_Toc509918391)

[5.1. Cerinte de performanta 14](#_Toc509918392)

[5.1.1. Cerinte non-functionale……………………………………………………………………………………………………..15](#_Toc509918393)

[5.1.2. Caracteritica de cautare proeminenta 15](#_Toc509918394)

[5.1.3.Utilizarea caracteristicii de cautare 15](#_Toc509918395)

[5.1.4. Utilizarea rezultatului din lista 15](#_Toc509918396)

[5.1.5Timp de raspuns. 16](#_Toc509918397)

[5.1.6. Fiabilitatea sistemului 16](#_Toc509918399)

[5.2. Cerinte de siguranta 16](#_Toc509918400)

[5.3. Cerinte de securitate 16](#_Toc509918401)

[5.4. Cerinte privind caltitatea software-ului 17](#_Toc509918402)

[5.4.1. Reguli de cod 17](#_Toc509918404)

[**5.6. Reguli Bussiness** 18](#_Toc509918405)

# 0.Istoricul editarilor

|  |  |  |  |
| --- | --- | --- | --- |
| **Nume** | **Data** | **Motivul editarii** | **Versiune** |
| Diaconu Ionut | 4.11.2018 | Adaugat 72 cerinte functionale | V.1 |
| Chivu Daniel | 5.11.2018 | Documentatia respecta standardul IEE, table of contents | V.1.1 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 1.**Introducere**

## 1.1 Scopul

Acest document descrie cerințele pentru un sistem informatic care gestionează activitățile din cadrul unei librarii online.

Sistemul realizat va fi un produs software construit pe arhitectura client/server pentru a putea cumpara produse dar totodata permite utilizatorilor accesul fara cont ,unde pot viziona produsele.

## 1.2 Sugestii vizate si sugestii de citire

|  |  |
| --- | --- |
| Termen | Definiție |
| Administrator | Are permisiunea specifică pentru gestionarea și controlul sistemului |
| Utilizator | Cine acceseaza aplicatia |
| DESC | Descriere |
| RAT | Rational |
| DEP | Dependența |

## 1.3 Domeniul de aplicare al produsului

Sistemul informatic cuprinde o interfata cu utilizatorul (navigator Web), un server de baze de date care va stoca informatii despre activitatea librariei, server de aplicatii. Utilizatorul are acces la aplicatie doar prin intermediul interfetei cu utilizatorul.

Administratorul aplicatiei gestioneaza sistemul si mentine informatii actualizate. Spre exemplu, el poate modifica datele cu privire la preturi sau pot actualiza informatiile despre produse.

## 1.4 Referinte

1. <https://www.google.ro/search?q=srs+tempalte&oq=srs+tempalte&aqs=chrome..69i57j0l5.4161j0j4&sourceid=chrome&ie=UTF->8
2. <https://stackoverflow.com/>
3. <https://www.w3schools.com/>
4. <https://www.librarie.ro/ro/primaria>

# 2.Descriere generală

## 2.1 Perspectiva produsului

Aplicația va conține o interfață web care va comunica cu un server web ce va accesa informațiile stocate într-o bază de date.

## 2.2 Functiile produsului

Libraria este o aplicație pusă la dispoziția clienților pentru a-și putea achiziționa orice produs dorit într-un interval de timp avantajos și un mod favorabil care nu implică deplasarea.

Utilizatorii si administratorul se autentifica introducand informațiile de log in pagina web care vor fi comunicate catre server.

## 2.3 Clasele si caracteristicile utilizatorului

Vor fi doua tipuri de utilizatori pentru acest sistem: utilizatorii aplicatiei si administratorul.

Utilizatorii pot accesa aplicația pentru a afla informații despre produsele dorite, având posibilitatea de a le adăuga în coș pt a lansa comanda,pot adauga produse la favorite.

Administratorul gestionează informații despre fiecare produs, având grijă să nu existe informații incorecte,poate reduce preturile produselor ,verifica stocul acestora si mai pe scurt poate aduce orice modificare in aplicatie.

## 2.4 Mediul de functionare

Sistemul informatic va fi accesibil atat pe sistemele de operare Windows 7, Windows 8, Windows 10, cat si pe mobil: Android(versiunile 6.0, 7.0, 8.0), iOS(versiunile 10.3,11.0).

## 2.5 Constrangeri de proiectare si implementare

Aplicatia va avea nevoie de o comunicatie la Internet, deja stabilita inainte de rulare care va permite sa se configureze la un server de baze de date pentru stocarea informatiilor si manegementul acestora.

## 2.6 Documentatia utilizatorului

Utilizatorul are la dispozitie acest document drept manual de utilizator.

## 2.7 Ipoteze si independente

-O prima ipoteza a acestei aplicatii este ca ofera posibilitatea de logare pt a cumpara produse sau exista posibilitatea de vizionarea a produselor fara cont

# 3.Cerintele interfetei

Scopul interfetei este de a oferi utilizatorului un mediu de lucru ordonat,accesand cu usurinta toate informatiile de care are nevoie si creeaza totodata o interactiune cu acesta.

## 3.1 Interfata utilizator

Fiecare utilizator ar trebui sa vadă pagina principală la deschiderea aplicației. Acesta are acces liber la toate informațiile aplicației, fiind necesară crearea unui cont doar la cumpărarea produselor dorite.

In cazul in care doresti sa plasezi o comanda ai nevoie de campurile inregistrare si logare.

Inregistrarea

Pentru inregistrare, ei vor fi nevoiti sa isi aleaga un nume de utilizator, o parola si sa ofere email-ul pe care se va inregistra. La alegerea utilizatorului, acesta mai poate adauga si cateva date de contact.

Logarea este optiunea obligatorie pentru trimiterea unei comenzi

**Log in**

Inregistrare

Obligatoriu

Nume utilizator:

Utilizator

Parola:

Admin

Nume utilizator:

Email:

Parola:

Optional

Date de contact

Telefon:

Date personale

Nume:

Prenume:

Adresa:

Odata logat pe site, utilizatorul va putea sa adauge produse in cos,sa adauge la favorite,sa trimita comenzi. Aceasta ofera o interfata grafica simpla si intuitiva.

După crearea contului, utilizatorii autentificați vor avea o pagină de profil în care își pot edita adresa de email, numărul de telefon, parola și adresa personală.

## 3.2 Interfata hardware

- Nu se aplica

## 3.3 Interfata software

Aplicatia dezvoltata va fi un portal web care va fii intretinuta pe un server local, oferind astfel utilizatorilor o navigare simpla si rapida.

Serverul local va avea acces in permanenta la baza noastra de date. Urmatoarele *functii prototip* vor arata rolul bazei de date in aplicatie:

* dbGet(criteriu) – returneaza un element din baza de date in functie de criteriul specificat
* dbAdd(element, tabel) – adauga in tabelul specificat din baza de date, un element
* dbRemove(element, tabel) – sterge in tabelul specificat, un element
* dbAverage(criteriu) – returneaza o medie aritmetica bazata pe un criteriu specificat
* dbModify(element,tabel,valoareNoua) – modifica valoarea unui element cu valoarea noua specificata
* dbEdit(tabel) – editeaza structura unui tabel (se pot adauga/sterge campuri noi)

Comunicarea intre baza de date si aplicatie consta in operarea privind citirea si modificarea datelor. SGBD-ul folosit va fi MySQL

## 3.4 Interfata de comunicatii

In momentul crearii unui cont de utilizator, aplicatia va comunica cu serviciul de e-mail cu care utilizatorul sa inregistrat, pentru a confirma inregistrarea. De asemenea, in cazul in care utilizatorul si-a uitat datele de logare, el poate solicita o recuperare a acestora. Recuperarea lor va consta in comunicarea acestora pe email-ul oferit de utilizator.

Login

Inregistrare

Confirm email

Salveaza date cont

Confirma identitatea

Trimite confirmarea-email

Access website-user

User

5.Actualizeaza baza de date

1.Introduce date de logare

2.Se verifica datele

3.Valideaza integritatea conturilor

4.Manageriaza aplicatia

Access

website

-admin

Administrator

Protocolul HTTP este un protocol de tip text, fiind protocolul "implicit" al WWW. Adică, dacă un URL  nu conține partea de protocol, aceasta se consideră ca fiind http. De aceea si noi il vom utiliza in aplicatia noastra.

## 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

### 5.1.2. Caracteristică de căutare proeminentă

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

Dependinte: Niciuna

### 5.1.3. Utilizarea caracteristicii de căutare

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

Dependinte: Niciuna

### 5.1.4. 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

### 5.1.5. 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.