Global Shop

Team:

Project Manager: Dan Chiorean

Database Developer: Iulian Gliga

Software Developer: Roxana Dodea

Ruxandra Ierima

Tester: Alina Bentan

**Cuprins**

1. **Continutul Proiectului**
   1. Obiectivul Proiectului
   2. Functiile Aplicatiei
2. **Design-ul Aplicatiei**
   1. Tipurile de Utilizatori si functiile lor
   2. Design-ul Bazei de Date
   3. Descrierea Interfetelor Grafice si structura codului

1. **Implementarea Aplicatiei**
   1. Distribuirea atributiilor echipei si procedura de lucru
   2. Implementarea Bazei de Date
   3. Implementarea detaliilor
      1. Autentificare
      2. Modelarea functionalitatii aplicatiei
2. **Testarea Aplicatiei**
   1. Procedura de testare
   2. Rezultate

**1.1 Obiectivul Proiectului**

Global Shop este o companie din Romania, care se ocupa cu vanzarea online de echipamente si component IT, electronice, electrocasnice, prodese auto, articole de ingrijire personala, imbracaminte si incaltaminte, carti, muzica si filme.

Cu aplicatia Global Shop cumperi rapid si in siguranta produsele dorite, te bucuri de promotiile si ofertele zilnice oferite.

Cel mai important obiectiv al acestui proiect este realizarea si implementareaunei aplicatii web, aceasta fiind scrisa in limbajul de programare C# folosind o baza de date creata in Microsoft SQL. Aplicatia complexa este proiectata in format Windows Forms Application. Aceasta este realizata de tipul aplicatiilor multinivel:

* Nivelul de gestionare a datelor in baza de date
* Nivelul logic implementat in interiorul programului
* Nivelul de prezentare in format Windows Forms Application

**1.2 Functiile Aplicatiei**

Aplicatia web gestioneaza si afiseaza date folosind SQL Server. Accesul la aceste informatii gestionate de baza de date este permis in functie de tipul de utilizator autentificat si de nivelul de informatie.

Permisiunea de a avea acces la operatiile CRUD (Create, Read, Update, Delete) realizate asupra aplicatiei depinde de cel mai important criteriu: tipul de utilizator. De exemplu, un simplu utilizator (cumparator) nu poate sterge sau edita stocul produselor, dar administratorul de vanzari poate realiza toate acestea. Fiecare entitate implementeaza in baza de date toate aceste operatii multiple cu diferite nivele de acces. Posibilitatea de a intra in cont si de a iesi din cont este implementata la fel ca si celelalte entitati, unde fiecare utilizator neautentificat poate intra in cont in functie de statutul pe care il are.

**2.1 Tipurile de utilizatori si functiile lor**

In aplicatia Global Shop avem trei tipuri de utilizatori:

1. Utilizatorul (cumparatorul)-> acest tip de utilizator poate sa realizeze urmatof icntarele operatii:

-poate sa se autentifice pe site;

-poate vedea gama larga de produse pe care le ofera site-ul;

-poate cauta produsele dorite;

-poate sa isi creeze propriul cos de cumparaturi cu produsele alese de el;

-poate sa modifice produsele din cos prin stergere sau adaugare;

-poate sa adauge la fovorit produsele preferate;

-poate sa isi plaseze comanda.

1. Administrator vanzari-> acest tip de utilizator poate efectua urmatoarele operatii:

-poate sa se autentifice;

-poate realiza operatii de adaugare, stergere, editare asupra stocului de produse;

-poate primi comenzile date de comparator;

-poate sa urmareasca cosurile de cumparaturi ale cumparatorilor.

3. Administrator de gestionare-> acest tip de utilizator efectueaza urmatoarele operatii:

-poate gestiona administratorii de vanzari si cumparatorii;

-poate sa adauge vanzatori.

-diagrama use case,pt fiecare sau impreuna

**2.2 Design-ul bazei de date**

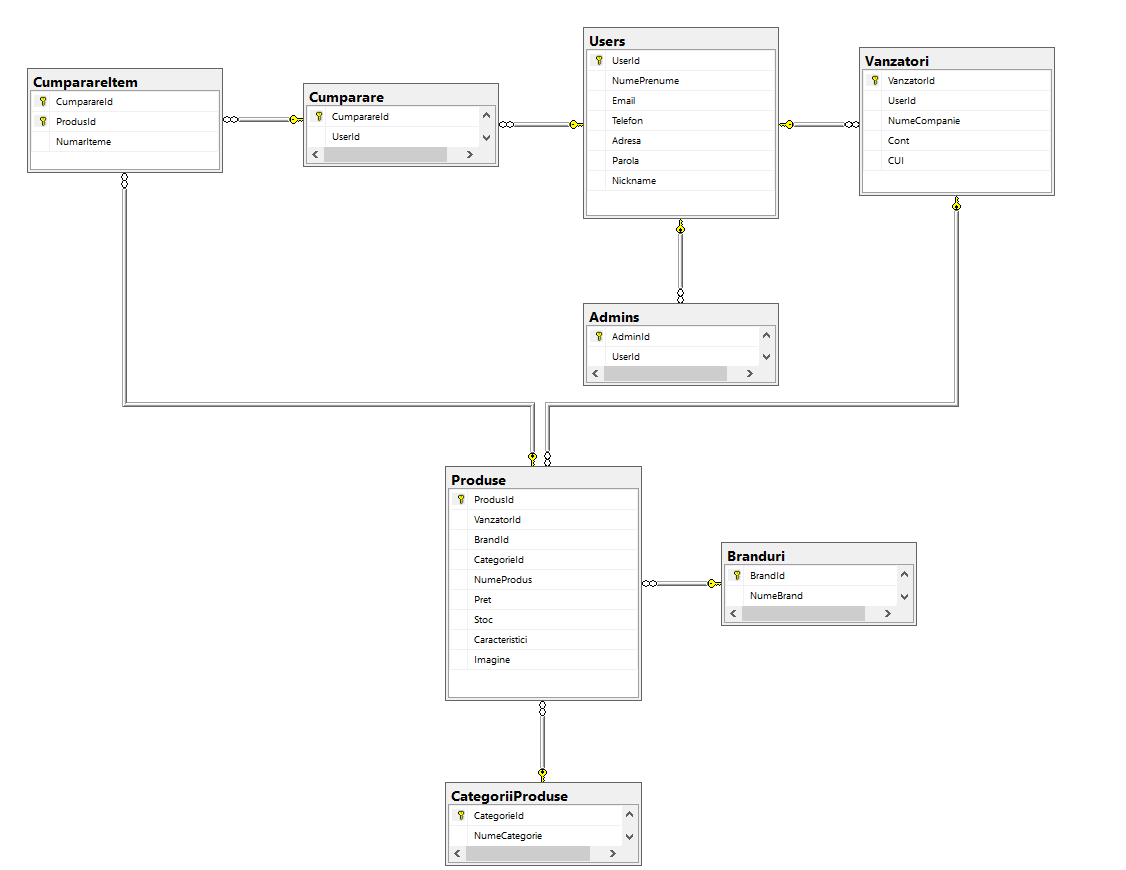


Figura 1

-numerotare pe subcapitole denumirea figurii

-mutata la implementare

-tipuri de date

-diagrama in alt mediu

Mutate relatiile

**Tabele Folosite:**

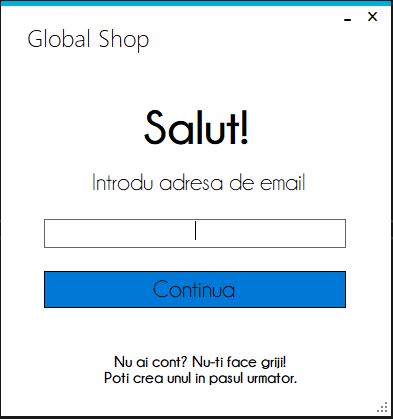
* Users -> lista tuturor utilizatorilor si a datelor personale existente in baza de date a magazinului (UserId, NumePrenume, Email, Telefon, Adresa, Parola, Nickname)
* Vanzatori -> lista tuturor vanzatorilor asociati magazinului (VanzatorId, UserId, NumeCompanie, Cont, CUI)
* Admins -> lista tuturor administratorilor magazinului (AdminId,UserId)
* Produse -> tabela cu toate produsele existente in magazine (ProdusId, VanzatorId, BrandId, CategorieId, NumeProdus, Pret, Stoc, Caracteristici, Imagine)
* Branduri -> lista brandurilor asociata produselor (BrandId, NumeBrand)
* CategoriiProduse -> lista categoriilor pe care o are un produs (CategorieId, NumeCategorie)
* CumparareItem -> tabela intermediara pentru achizitionarea mai multor produse (CumparareId, ProdusId, NumarIteme)
* Cumparare -> tabela finala pentru achizitionarea produselor (CumparareId, UserId)

**2.3 Descrierea Interfetelor Grafice si structura codului**

Interfetele grafice au fost realizate de tipul Windows Forms Application. Am folosit frameworke-ul MetroModernUI pentru o prezentare mai placuta a aplicatiei.

Pagina de pornire este o pagina care iti cere adresa de e-mail. In functie daca esti deja un utilizator inregistrat in baza de date acesta de va redirectiona catre o pagina care iti va cere e-mailul. In cazul in care nu esti un utilizator acesta te va redirectiona catre o pagina care iti va inregistra datele si se realizeaza inregistrarea utilizatorului.

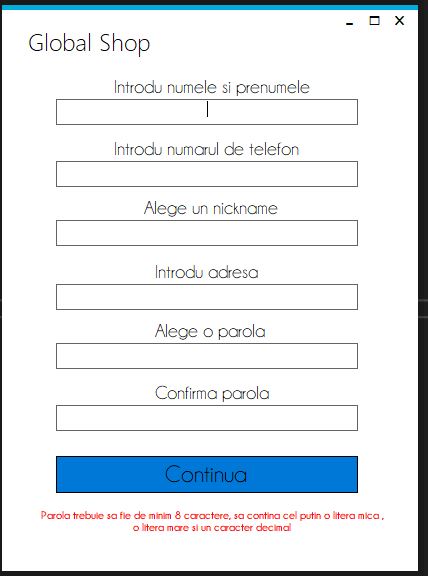
Pagina de pornire:



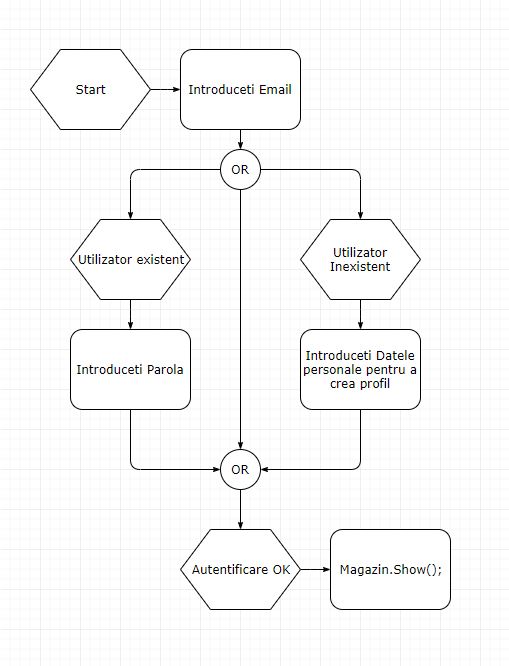
Utilizator existent:



Utilizator inexistent:



**Diagrama de evenimente a aplicatiei de Login**

****

**2.3**

**3.2 Implementarea Bazei de Date**

**-denumirea la figura + numar**

**-printuri uniforme**

Baza de date a fost implementata in MySQL Server.

Tabela Users:

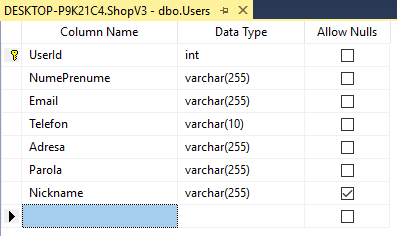


Tabela Admins:

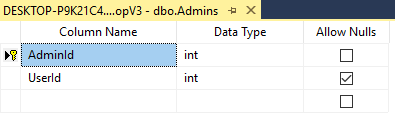


Tabela Vanzatori:

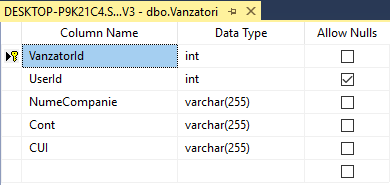


Tabela Produse:

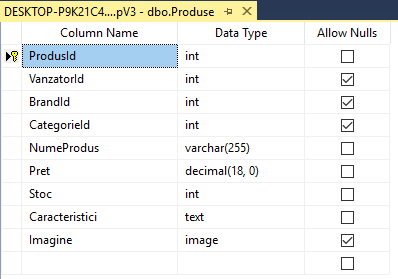


Tabela Branduri:

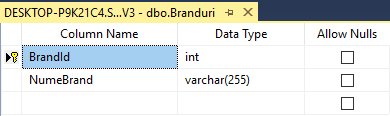


Tabela CategoriiProduse:

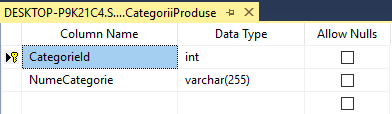


Tabela CumparareItem:

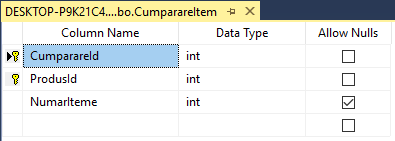
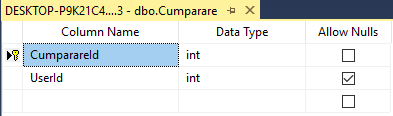


Tabela Cumparare:



Relatii de tip One-To-Many:

* UserId(Users) si UserId(Admin), UserId(Vanzatori), UserId(Cumparare)
* VanzatorId(Vanzatori) si VanzatorId(Produse)
* BrandId(Branduri) si BrandId(Produse)
* CategorieId(CategoriiProduse) si CategorieId(Produse)
* ProdusId(Produse) si ProdusId(CumparareItem)
* CumparareId(Cumparare) si CumparareId(CumparareItem)