**Universitatea Tehnica a Moldovei**

**Facultatea Calculatoare, Informatica si Microelectronica**

**Lucrare de totalizare curs la**

**Disciplina:**Baze de date

**Tema:**Proiectarea unei Baze de date si elaboarea unei Aplicatii Web pentru domeniul: “**Serviciu schimb de apartamente”**

**A elaborat:** Studentul grupei C-162 Marjina Alexandru

**A verificat:** Conf. univ. Prebinos Mihail

Chisinau 2017

**Lucrarea urmează să fie executată în conformitate cu formatul ce urmează:**

**1**. **Capitolul I (Introducere în SGBD)**

1.1 Conceptul Bazelor de Date

1.2 Gestiunea Bazelor de Date

**2.Capitolul II (Modelul Conceptual, Modelul Logic)**

2.1 Scopul si obiectivele sarcinii:

2.2 Descrierea domeniului de studiu:

2.3 Schema legăturilor intre subdomeniilor domeniului de studiu:

2.4 Caracteristicile informaționale ale *Paginii Web* sau *o aplicației Desktop* “***Activitatea companiei de producere şi realizare a materialelor de construcţie***”.

2.4Tabelul de acoperire:

2.6Legaturi intre obiectele Domeniului de Studiu:

2.7 Legăturile intre atributele obiectelor Domeniului de Studiu ER –schema BD:

2.8 Modelul logic al problemei./Prezentarea schemei ER/

2.9 Mochetul bazei de date

**3.Capitolul III (Introducere Programare WEB)**

3.1 HTML Prezentare

3.2 Limbaje de Scriptare

3.3 PHP Reprezentare

3.4 Serverul Web Apache

3.5 Administrarea Bazelor de Date cu MySQL

**4 Capitolul IV (Structura Bazei de date)**

4.1 Machetul Bazei de Date:

4.2 Tabelul «… » Nu mai puțin de 7 tabele !!!

4.3 ….

4.4 SCHEMA ER A BD – SCREENSHOT-UL ANEXAT

**5Capitolul V (Structura site-ului)**

5.1 Scopul site-ului

5.2 Descrierea site-ului

5.3 Componenta ”Admin”

5.3.1 “Modificare înregistrări”

5.3.2 “Vizualizarea datelor ….”

5.3.3 “Totalizarea datelor”

5.3.4 “Datele de ieșire ale sarcinii”

5.4 “Căutări” informații suplimentare

5.5 Componenta “Contact“

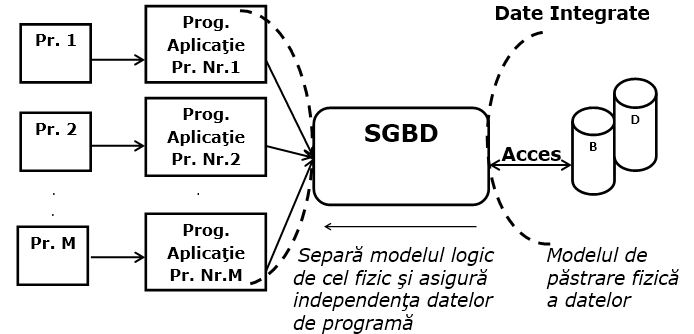
**Capitolul VI (Listing Program)**

**Concluzie**

**Bibliografie**

**Anexe**

1. **Capitolul I: Introducere în SGBD**



* 1. **Conceptul Bazelor de Date**

O *bază de date* reprezintă un ansamblu de date integrat, anume structurat şi dotat cu o descriere a acestei structuri. Descrierea structurii poartă numele de *dicţionar de date* sau *metadate* şi crează o interdependenţă între datele propriu-zise şi programe.

Datele dintr-o bază de date sunt organizate pe 3 nivele:

* **Nivelul intern** (nivel de stocare) aflat cel mai aproape de mediul de stocare fizic – se referă la modul în care sunt stocate datele în sistem.
* **Nivelul extern** (nivel logic al utilizatorului) - aflat cel mai aproape de utilizatori – se referă la modul în care sunt vizualizate datele de către utilizatorii individuali.
* **Nivelul conceptual** (nivel logic colectiv sau, nivel logic) - un nivel intermediar între celelalte două. Acestui nivel îi corespunde structura conceptuală (schema) a bazei de date, prin care se realizează o descriere a tuturor datelor, într-un mod independent de aplicaţii, ce face posibilă administrarea datelor.

Se cunosc mai multe tipuri de baze de date după modul de organizare, modul de dispunere pe suport magnetic a informaţiei şi a elementelor componente:

* **modele primitive**– datele sunt organizate la nivel logic în fişiere, structura de bază este înregistrarea, mai multe înregistrări fiind grupate în structuri de tip fişier;
* **baze de date ierarhice**– legăturile dintre date sunt ordonate unic, accesul se face numai prin vârful ierarhiei, un subordonat nu poate avea decât un singur superior direct şi nu se poate ajunge la el decât pe o singură cale;
* **baze de date în reţea**– datele sunt reprezentate ca într-o mulţime de ierarhii, în care un membru al ei poate avea oricâţi superiori, iar la un subordonat se poate ajunge pe mai multe căi;
* **baze de date relaţionale**– structura de bază a datelor este aceea de *relaţie – tabel*, limbajul SQL (Structured Query Language) este specializat în comenzi de manipulare la nivel de tabel. Termenul relaţional a fost introdus de un cercetător al firmei IBM dr. E. F. Codd în 1969 cel care a enunţat cele 13 reguli de bază necesare pentru definerea unei baze de date relaţionale. Baza de date relaţională reprezintă o mulţime structurată de date, accesibile prin calculator, care pot satisface în timp minim şi într-o manieră selectivă mai mulţi utilizatori. Această mulţime de date modelează un sistem sau un proces din lumea reală şi serveşte ca suport unei aplicaţii informatice;
* **baze de date distribuite**– sunt rezultatul integrării tehnologiei bazelor de date cu cea a reţelelor de calculatoare. Sunt baze de date logic integrate, dar fizic distribuite pe mai multe sisteme de calcul. Integrarea bazei de date distribuite se face cu ajutorul celor 3 tipuri de de scheme care sunt implementate:
  + 1. **schema globală**– defineşte şi descrie toate informaţiile din baza de date distribuită în reţea;
    2. **schema de fragmentare**– descrie legăturile dintre o colecţie globală şi fragmentele sale. Ea este de tipul unu la mai mulţi şi are forma unei ierarhii;
    3. **schema de alocare**– descrie modul de distribuire a segmentelor pe calculatoarele (nodurile) din reţea. Fiecare segment va avea o alocare fizică pe unul sau mai multe calculatoare. Schema de alocare introduce o redundanţă minimă şi controlată: un anumit segment se poate regăsi fizic pe mai multe calculatoare.
  1. **Gestiunea Bazelor de Date**

*Sistemul de gestiune a bazalor de date* (SGBD) sunt un ansamblu de programe prin care se asigură gestionarea şi prelucrarea complexă a datelor şi care reprezintă componenta software a sistemului de baze de date.

De obicei o baza de date este memorată într-unul sau mai multe fişiere, fiind manipulate cu ajutorul sistemelor de gestiune a bazelor de date

Funcţiile unui SGBD :

* **definirea** — crearea bazei de date;
* **introducerea** (adăugarea) datelor în baza de date;
* **modificarea** unor date deja existente în baza de date;
* **ştergerea** datelor din baza de date;

Un SGBD trebuie să asigure următoarele activităţi:

* *definirea şi descrierea structurii bazei de date* – se realizează printr-un limbaj propriu, limbaj de definire a datelor (LDD), conform unui anumit model de date;
* *încărcarea datelor în baza de date* – se realizează prin comenzi în limbaj propriu, limbaj de manipulare a datelor (LMD);
* *accesul la date* – se realizează prin comenzi specifice din limbajul de manipulare a datelor. Accesul la date se referă la operaţiile de *interogare şi actualizare*.

1. **Capitolul II: Modelul Conceptual, Modelul Logic**

**I *Proiectarea bazei de date Input-Output***

**Descrierea domeniului de studiu:**

Serviciul schimb de apartamente mereu a fost un lucru necesar in societate, odata cu aparitia numarului mare de tehnolgii viata noastra a devenit mai simpla astfel noi putem sa efectuam un sir de operatiuni economisind o multime de timp. Astfel a aprut si serviciul schimb de apartamente care acum poate fi gestionat si automatizat online, clientul ne avind nevoie sa vina la oficiul companiei pentru a utiliza unul din serviciile disponibile, totul este realizat online avind nevoie doar de un dispozitiv cu acces la internet.

1. **Modelul conceptual**

Utilizator

Serviciu schimb de apartamente

Agentie

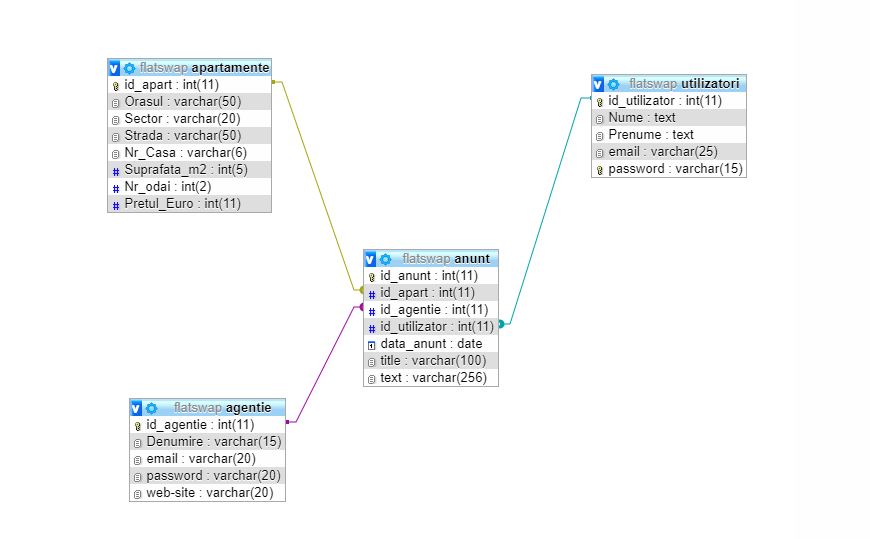
Anunt

Apartament

**Machetul bazei de date**

|  |  |  |  |
| --- | --- | --- | --- |
| Tabel | Cimp | Tip(Lung) | Comentariu |
| Agentie | id\_agentie | Int(11) | Primary key |
| Denumire | Varchar(20) |  |
| Email | Varchar(20) |  |
| Password | Varchar(20) |  |
| Web-site | Varchar(20) |  |
| Apartament | Id\_apart | Int(11) | Primary key |
| Orasul | Varchar(50) |  |
| Sector | Varchar(20) |  |
| Strada | Varchar(50) |  |
| Nr\_casa | Varchar(6) |  |
| Suprafata\_m2 | Int(5) |  |
| Nr\_odai | Int(2) |  |
| Pretul\_euro | Int(11) |  |
| Anunt | Id\_anunt | Int(11) | Primary key |
| Id\_apart | Int(11) | Foreign key |
| Id\_agentie | Int(11) | Foreign key |
| Id\_utilizator | Int(11) | Foreign key |
| Data\_anunt | Date |  |
| Title | Varchar(100) |  |
| Text | Varchar(256) |  |
| Utilizatori | Id\_utilizator | Int(11) | Primary key |
| Nume | text |  |
| Prenume | Text |  |
| Email | Varchar(25) |  |
| Password | Varchar(15) |  |

1. **Modelul logic / Schema ER in phpMyAdmin Designer**



**II *Proiectarea bazei de date Output-Input***

* 1. **Scopul si obiectivele sarcinii**

**Scopul:**

Crearea unui SAPD (Sistem Automatizat de Prelucrare a Datelor) ce ar permite utilizatorului să se foloseasca de servicii de cumparare/chirie imobile, vinzare/dare in arenda , toate operatiile constituinduse online.

**Obiectivele**

***Obiectiv nr.1***

Gestionarea datelor privind ofertele si cererile a serviciilor imobile in Chisinau .

***Obiectiv nr. 2***

Asigurarea Legaturilor intre cumparator, vinzator si companiei.

***Obiectiv nr. 3***

În baza datelor și informațiilor despre apartamente, despre serviicle acordate de catre companie promovarea pe piata a ofertelor companiei si obtinerea de profit.

***Obiectiv nr. 4***

Realizind ofertele propuse si promovate administratia companiei are posibilitatea de a vedea eficienta companiei si a angajatilor sai.

* 1. **Descrierea domeniului de studiu:**

Serviciul schimb de apartamente mereu a fost un lucru necesar in societate, odata cu aparitia numarului mare de tehnolgii viata noastra a devenit mai simpla astfel noi putem sa efectuam un sir de operatiuni economisind o multime de timp. Astfel a aprut si serviciul schimb de apartamente care acum poate fi gestionat si automatizat online, clientul ne avind nevoie sa vina la oficiul companiei pentru a utiliza unul din serviciile disponibile, totul este realizat online avind nevoie doar de un dispozitiv cu acces la internet.

**Din domeniul de studiu’’Serviciu schimb de apartamente’’ putem desprinde urmatoare subdomenii si obiecte ce il descriu:**

* **Vinzator**

Acesta este subdomeniu care reprezinta clientul care doreste sa vinda sau dea in arenda un imobil.

* **Cumparator**

Acesta este subdomeniu care reprezinta clientul care doreste sa cumpere sau sa inchirieze un imobil.

* **Oferta**

Reprezinta imobilul care este pus in vinzare de catre vinzator.

* **Cererea**

Reprezinta imobilul care este cerut de catre cumparatori.

* **Contract Vinzare**

Este subdomeniul care reprezinta legatura dintre servciul acordat de companie, de vinzator si de oferta acestuia.

* **Contract Cumpare**

Este subdomeniul care reprezinta legatura dintre servciul acordat de companie, de cumparator si de cererea acestuia.

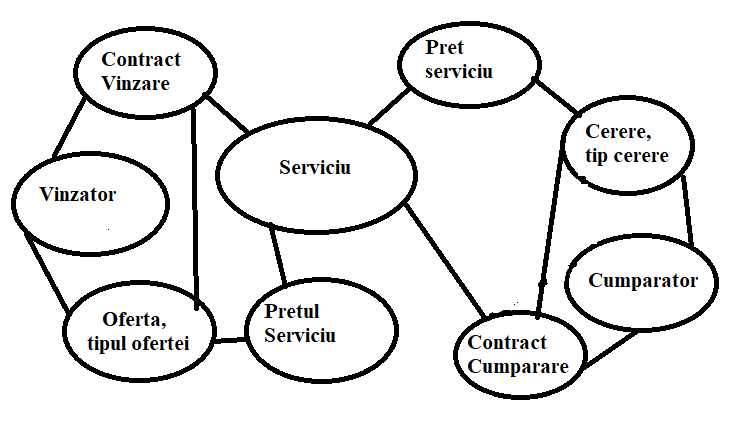
* **Serviciu**

Este subdomeniul care reprezinta comnia care acorda serviciul de schimb de apartamente, si face legatura dintre vinzator si cumpartor prin contractele respective.

* **Pret Servicu**

Este subdomeniul care reprezinta preturile stabilite de catre companie in conformitate cu serviciile acordate.

* 1. **Schema legăturilor între subdomeniilor domeniului de studiu**



* 1. **Caracteristicile informaționale ale *Paginii Web***

**a)Iesiri informationale:**

**Pentru administratia companiei :**

1)Lista contract vinzare/dare in arenda

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nr.ContrV | Data inc. | Data sf. | Id\_serv | Serviciu | Id\_Angajat | Pret Serv |

2)Lista contract cumparare/chirie

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nr.ContrC | Data inc. | Data sf. | Id\_serv | Serviciu | Id\_Angajat | Pret Serv |

**Pentru utilizatorii sitului :**

3)Lista Servicii vinzari/dare in arenda

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr serv | Grupa Serv | Nume Serv | Pret serv | Descriere |

4)Lista Servicii cumparari/chirie

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr serv | Grupa Serv | Nume Serv | Pret serv | Descriere |

5)Lista ofertelor disponibile

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nr oferta | Tip oferta | Oferta | Data inc. | Data sf. | Id\_Angajat |

6)Lista cererilor disponibile

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nr cerere | Tip cerere | Cerere | Data inc. | Data sf. | Id\_Angajat |

**b)Interogari**

**Pentru Administratie:**

**Input:**Data expirare serviciu xxxx

1)Lista Contract vinzari/dare in arenda care se termina la o data respective:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr.ContrV | Id\_serv | Serviciu | Id\_Angajat | Pret serv |

2)Lista Contract cumparare/chirie care se termina la o data respective:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr.ContrtC | Id\_serv | Serviciu | Id\_Angajat | Pret serv |

3)Lista servicii cumparare/chirie, cele mai solicitate pina la data curenta

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr ContrC | Grupa serv | Nume serv | Pret serv | Nr de solicitari |

**c)Servicii**

**Pentru Administratie:**

**Input:** [Data min, Data max]

1)Nr total contracte vinzari/dare in arenda si venitul de la acestea

|  |  |
| --- | --- |
| Nr total ContrV | Suma total venit ContrV |

2) Nr total contracte cumparare/chirie si venitul de la acestea

|  |  |
| --- | --- |
| Nr total ContrC | Suma total venit ContrC |

3)Lista angajatilor si eficienta lor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id\_Angajat | Nume | Prenume | Sarcini\_active | Sarcini\_realizate |

4)Cele mai solicitate servicii

|  |  |  |  |
| --- | --- | --- | --- |
| Id\_serv | Nume\_Serviciu | Pret\_serviciu | Nr\_solicitari |

**Restrictii**

1)Modele vinzari si modelul cumpararii sunt acelasi.

2)Modele dare in arenda si chirie sunt la fel .

3)In contract poate fi stipulate doar un singur obiect.

4)Informatii despre eficienta companiei le are doar conducerea companii avind un acces personalizat la aceasta.

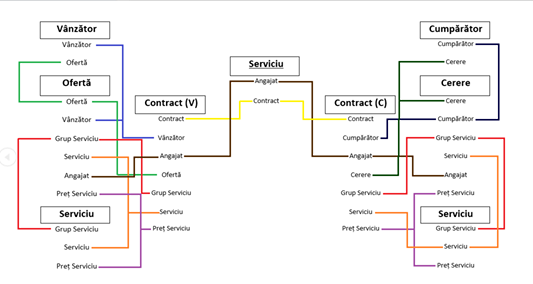
5)Vinzatorul si cumparatorul nu contacteaza direct.

6)Utilizatorul nu poate cauta un apartament cu un numar de camere mai mare de 6

* 1. **Tabel de acoperire**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Info**  **Cimpuri** | **Iesiri info** | | | | | | **Interogari** | | | | **Servicii** | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **5** |
| Nr.ContrV | **+** |  |  |  |  |  | **+** |  | **+** |  |  |  |  |  |  |
| Nr.ContrC |  | **+** |  |  |  |  |  | **+** |  | **+** |  |  |  |  |  |
| Tip oferta | **+** |  |  |  | **+** |  | **+** |  |  |  |  |  |  |  |  |
| Tip cerere |  | **+** |  |  |  | **+** |  | **+** |  |  |  |  |  |  |  |
| Oferta | **+** |  |  |  | **+** |  | **+** |  |  |  |  |  |  |  |  |
| Cererea |  | **+** |  |  |  | **+** |  | **+** |  |  |  |  |  |  |  |
| Nr\_oferta |  |  |  |  | **+** |  |  |  |  |  |  |  |  |  |  |
| Nr\_cerere |  |  |  |  |  | **+** |  |  |  |  |  |  |  |  |  |
| Nr total ContrV |  |  |  |  |  |  |  |  |  |  |  |  | **+** |  |  |
| Nr total ContrC |  |  |  |  |  |  |  |  |  |  |  |  |  | **+** |  |
| Suma total venit ContrV |  |  |  |  |  |  |  |  |  |  | **+** |  |  |  |  |
| Suma total venit ContrC |  |  |  |  |  |  |  |  |  |  |  | **+** |  |  |  |
| Data inc. | **+** | **+** |  |  | **+** | **+** |  |  |  |  |  |  |  |  |  |
| Data sf. | **+** | **+** |  |  | **+** | **+** |  |  |  |  |  |  |  |  |  |
| Id\_Angajat | **+** | **+** |  |  | **+** | **+** | **+** | **+** |  |  |  |  |  |  | **+** |
| Nume\_angajat |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **+** |
| Prenume\_angajat |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **+** |
| Pret serv | **+** | **+** | **+** | **+** |  |  | **+** | **+** | **+** | **+** |  |  |  |  |  |
| Nr serv |  |  | **+** | **+** |  |  |  |  | **+** | **+** |  |  |  |  |  |
| Grupa serv |  |  | **+** | **+** |  |  |  |  | **+** | **+** |  |  |  |  |  |
| Nume serv |  |  | **+** | **+** |  |  |  |  |  |  |  |  |  |  |  |
| Descriere |  |  | **+** | **+** |  |  |  |  |  |  |  |  |  |  |  |
| Nr de solicitari |  |  |  |  |  |  |  |  | **+** | **+** |  |  |  |  |  |
| Sarcini\_active |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **+** |
| Sarcini\_completate |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **+** |

* 1. **Legaturi intre obiectele Domeniului de Studiu:**



* 1. **Legaturile între atributele obiectelor Domeniului de Studiu ER –schema BD:**

**„Vinzator”**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id\_vinzator | Nume | Prenume | Telefon | Email |

**„Oferta”**

|  |  |  |  |
| --- | --- | --- | --- |
| Id\_oferta | Tip\_oferta | Id\_vinzator | Id\_apartament |

**„Apatament”**

|  |  |  |  |
| --- | --- | --- | --- |
| Id\_apartament | Nr. de camere | Etajul | Suprafata |

**„Servicii”**

|  |  |  |  |
| --- | --- | --- | --- |
| Id\_serv | Nume serviciu | Id\_GrServ | Pret\_serv |

**„Gr.Servicii”**

|  |  |
| --- | --- |
| Id\_GrServ | Nume GrServ |

**„ContractV”**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Id\_contrV | Id\_vinzator | Id\_oferta | Data\_inc | Data\_sf | Id\_serv | Id\_angajat | Realizat |

**„Cumparator”**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id\_cumparator | Nume | Prenume | Telefon | Email |

**„Cerere”**

|  |  |  |  |
| --- | --- | --- | --- |
| Id\_cerere | Tip\_ cerere | Id\_ cumparator | Id\_apartament |

**„ContractC”**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Id\_contrC | Id\_cumparator | Id\_cerere | Data\_inc | Data\_sf | Id\_serv | Id\_angajat | Realizat |

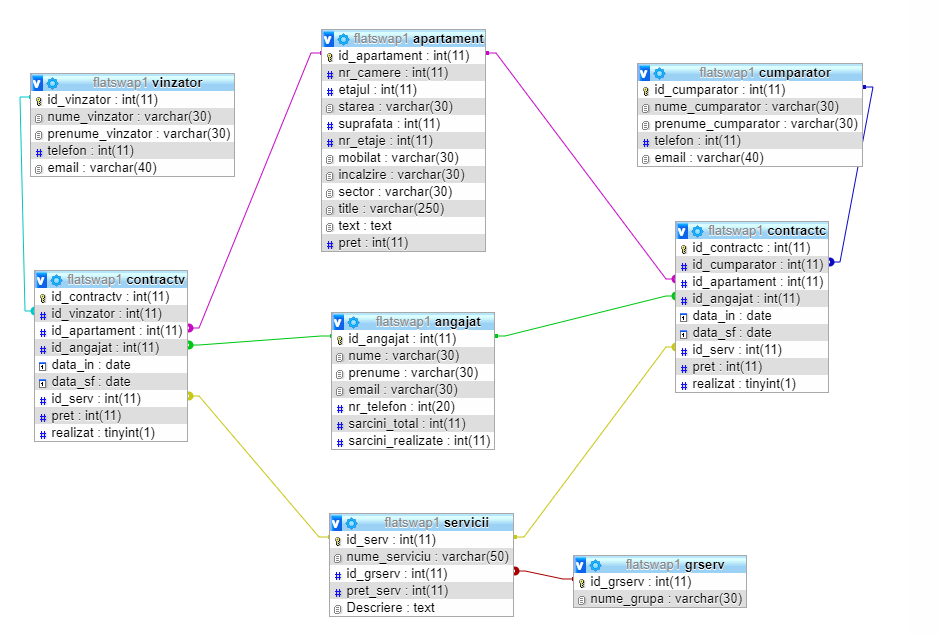
**„Angajat”**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Id\_angajat | Nume | Prenume | Email | Nr\_telefon | Sarcini\_total | Sarcini\_realizat |

* 1. **Machetul bazei de date**

|  |  |  |  |
| --- | --- | --- | --- |
| Tabel | Cimp | Tip(Lung) | Comentariu |
| Vinzator | Id\_vinzator | Int(11) | PRIMARY KEY |
| Nume\_vinzator | Text |  |
| Prenume\_vinzator | Text |  |
| Telefon | int |  |
| email | Varchar(40) |  |
| Oferta | Id\_oferta | Int(11) | Primary key |
| Id\_vinzator | Int(11) |  |
| Tip\_oferta | Text |  |
| Id\_apartament | Int(11) | Foreign Key |
| ContractV | Id\_contractV | Int(11) | Primary Key |
| Id\_vinzator | Int(11) | Foreign Key |
| Id\_oferta | Int(11) | Foreign Key |
| Id\_angajat | Int(11) | Foreign Key |
| Id\_serv | Int(11) | Foreign Key |
| Data\_inc | Date |  |
| Data\_sf | Date |  |
| Pret | Int(11) |  |
| Realizat | Int(1) |  |
| Servicii | Id\_serv | Int(11) | Primary Key |
| Nume\_serv | Varchar(20) |  |
| Id\_grserv | Int(11) | Foreign Key |
| Pret\_serv | Int(11) |  |
| GrServ | Id\_grserv | Int(11) | Primary Key |
| Nume\_grupa | Text |  |
| Angajat | Id\_angajat | Int(11) | Primary key |
| Nume | Text |  |
| Prenume | Text |  |
| Email | Varchar(30) |  |
| Nr\_telefon | Int(20) |  |
| Sarcini\_total | Int(11) |  |
| Sarcini\_realizate | Int(11) |  |
| Apartament | Id\_apartament | Int(11) | Primary key |
| Nr\_camere | Int(11) |  |
| Etajul | Int(11) |  |
| Starea | Varchar(30) |  |
| Suprafata(m2) | Int(11) |  |
| Nr\_etaje | Int(11) |  |
| Mobilat | Varchar(30) |  |
| Planificarea | Varchar(30) |  |
| Incalzirea | Varchar(30) |  |
| Sector | Varchar(30) |  |
| Title | Varchar(30) |  |
| Text | Varchar(30) |  |
| Pret | Int(11) |  |
| Cumparator | Id\_cumparator | Int(11) | PRIMARY KEY |
| Nume\_ cumparator | Text |  |
| Prenume\_ cumparator | Text |  |
| Telefon | int |  |
| email | Varchar(40) |  |
| Cerere | Id\_cerere | Int(11) | Primary key |
| Id\_cumpartor | Int(11) |  |
| Tip\_cerere | Text |  |
| Id\_apartament | Int(11) | Foreign Key |
| ContractC | Id\_contractC | Int(11) | Primary Key |
| Id\_cumparator | Int(11) | Foreign Key |
| Id\_cererea | Int(11) | Foreign Key |
| Id\_angajat | Int(11) | Foreign Key |
| Id\_serv | Int(11) | Foreign Key |
| Data\_inc | Date |  |
| Data\_sf | Date |  |
| Pret | Int(11) |  |
| Realizat | Int(1) |  |

* 1. **Modelul logic al problemei./Prezentarea schemei ER in PhpMyAdmin**



**3. Capitolul III (Introducere Programare WEB)**

* 1. **HTML Prezentare**

HyperText Markup Language (HTML) este un [limbaj de marcare](http://ro.wikipedia.org/wiki/Limbaj_de_marcare) utilizat pentru crearea paginilor web ce pot fi afișate într-un [browser](http://ro.wikipedia.org/wiki/Browser) (navigator). Scopul HTML este mai degrabă prezentarea informațiilor – paragrafe, fonturi, tabele ș.a.m.d. – decât descrierea semanticii documentului.

Specificațiile HTML sunt dictate de [World Wide Web Consortium](http://ro.wikipedia.org/wiki/World_Wide_Web_Consortium) (W3C).

HTML este un limbaj de marcare din familia SGML.

SGML - Standard Generalized Markup Language este un [ISO](http://en.wikipedia.org/wiki/International_Organization_for_Standardization)-standard pentru definirea limbajeor de marcare a documentelor.

**Documenele HTML**

Documentele HTML sunt fişerele textuale şi se salveză în fişiere cu extensia .html sau .htm

Limbajul de marcare HTML nu utilizeaza instructiuni, ci etichete, acestea fiind numite tag-uri.

**Structura unui document HTML**

<html>  
    <head>

<title>

*Titlul paginii*

</title>

</head>  
   <body>

*Continutul paginii*

  </body>  
</html>

Documentele HTML pot fi create cu un editor de texte sau cu editoare HTML care permit crearea vizuală (WYSIWYG) rezultînd însă tot documente în format text-pur. Limbajul HTML dă autorilor posibilitatea :

* Să publice documente cu headere, texte, tabele, liste, fotografii etc;
* Să regasească on-line informații prin intermediul hiperlink-urilor accesate printr-un simplu click de mouse;
* Să proiecteze formulare pentru realizarea tranzacțiilor cu servere aflate la distanță pentru căutari de informație sau pentru activități specifice comerțului;
* Să includă foi de calcul tabelar, clipuri video, sunete și alte aplicații direct în documente.

Limbajul HTML are patru caracteristici principale:

1. Folosește un marcaj descriptiv pentru a indica diversele acțiuni ce trebuie executate. Aceasta înseamnă că părți ale documentului descris de HTML sunt marcate cu nume descriptive, ca de exemplu <CHAPTER> sau <TITLE> care sunt aplicabile oricărei porțiuni de date corespunzătoare din document.
2. Definește structuri de documente ierarhice și legături întra- și inter-documente. O legătură este o relație unară între două elemente ale unui document. Structura unui document este însoțită de astfel de legături între elementele sale.
3. Limbajul HTML este guvernat de o descriere formală. HTML are o descriere a tipului documentului (Document Type Definition DTD) care stabilește specificațiile formale ale limbajului. DTD stabileste sintaxa limbajului, descrie fiecare element individual al unui document scris în limbajul HTML, definește atributele permise pentru fiecare element și descrie modelul datelor conținute în fiecare element. În corelație cu informațiile despre elemente, DTD oferă definiții pentru entitățile externe ce pot fi referite în HTML.
4. Atât specificațiile limbajului cât și limbajul însuși pot fi citite și de om dar și de computer. Datorită faptului că elementele de marcare sunt separate de text prin șiruri de delimitare alcătuite din caractere tiparibile, textul și marcajele pot coexista.

HTML este un limbaj bazat pe SGML (Standard Generalized Markup Language). SGML este un standard internațional (ISO-8879) aprobat în 1986.

SGML oferă o modalitate de a reprezenta structura documentelor și hyper-documentelor. Este totodată și o cale de a codifica hyper-documentele astfel ca acesta să poată fi interschimbate asemanator procesului de interschimbare a unor documente în cazul mai multor autori care colaborează utilizând platforme diferite aflate la distanță.

SGML este un sistem complex de descriere a documentelor. Este utilizat pentru descrierea structurii generale a diferitelor tipuri de documente fără să fie un limbaj de descriere a paginii. Principala preocupare a SGML se răsfrânge asupra conținutului documentului, nu asupra aspectului sau.

Deci, în virtutea trăsăturilor moștenite de la SGML, HTML este un limbaj pentru descrierea documentelor structurate. Teoria din spatele acestui limbaj se bazează pe faptul că majoritatea documentelor au elemente comune și că dacă definești un set de elemente, poți marca elementele documentului cu etichetele corespunzătoare.

Majoritatea etichetelor HTML arată sub forma:

<NUME ETICHET{> Textul afectat de eticheta </NUME ETICHET{> și indica navigatorului elemente de structură a documentului, formatare, hypertext sau alte elemente. Documentele HTML conțin doar textul propriu-zis și etichetele HTML iar sursa lor poate fi ușor văzută din orice navigator.

* 1. **Limbaje de Scriptare**

Baza comunicării între servere și clienții Web, la nivel de aplicație este asigurată de protocolul HTTP (HyperText Transfer Protocol). Protocolul HTTP este folosit în special pentru hipertext, dar este un protocol generic, putând susține un sistem distribuit pentru managementul obiectelor de date. O caracteristică importantă a protocolului este independența de platformă, diferitele calculatoare care comunică prin HTTP putând folosi diverse sisteme de operare și aplicații hipertext.

Principalele concepte cu care lucrează protocolul sunt cererea și raspunsul: un client Web trimite un mesaj(cererea) la un server. Mesajul conține identificatorul resursei dorite, dat sub forma unui URI(Uniform Resource Identifier), metoda de acces folosită, precum și o serie de meta-informații care pot fi utile serverului. Răspunsul serverului cuprinde un cod indicând starea serverului după interpretarea cererii, un mesaj explicativ pentru codul de stare transmis, meta-informațiile care vor fi procesate de către client și, eventual, un conținut(resursa solicitată).

Standard pentru interacțiunea clienților Web cu serverele Web, Common Gateway Interface se află în prezent la versiunea 1.1. Un program CGI, denumit în mod uzual script, se execută pe serverul WWW fie în mod explicit, apelat din cadrul paginii printr-o directivă specială, fie la preluarea informațiilor aflate în cadrul câmpurilor unui formular interactiv sau coordonatelor unei zone senzitive. CGI conferă interactivitate paginilor Web, documentele HTML putând să-și modifice în mod dinamic conținutul și să permită prelucrări sofisticate de date. Programele CGI pot oferi suport și la autentificarea utilizatorilor pe partea de server.

Programele CGI pot fi scrise în orice limbaj, fiind interpretate în cazul limbajelor PHP, Perl, Python, TCL sau compilate cum se întimplă pentru limbajul C și Delphi.

În conceperea unui script CGI trebuie respectate anumite reguli de bază:

* programul scrie datele spre a fi transmise navigatorului Web la ieșirea standard
* programul generează anteturi care permit serverului Web să interpreteze corect ieșirea scriptului

Cele mai multe scripturi CGI sunt concepute pentru a procesa datele introduse în formulare. Un formular se definește în HTML folosindu-se marcatori specifici pentru afisarea conținutului și introducerea datelor către clienți, iar scriptul invocat și executat de serverul de Web, va prelua conținutul acelui formular și-l va prelucra, returnând eventual rezultatele către navigator.

* 1. **PHP Reprezentare**

PHP (PHP: Hypertext Prepocessor), cunoscut în versiunile mai vechi şi sub numele de PHP/FI (Personal Homepage/Form Interpreter), iniţial a fost gândit a fi o simplă aplicaţie CGI pentru interpretarea formularelor definte prin HTML şi procesate de un program scris într-un limbaj Perl, script shell, executat pe server. În cazul interfeţei CGI era necesară permisiunea de a rula programe pe server, ceea ce ducea la lacune în securitate şi în plus la disocierea de documentul HTML a programului care procesa datele.

PHP (în versiunea curentă PHP 5.0) reprezintă un pachet puternic care oferă un limbaj de programare accesibil din cadrul fişierelor HTML, limbaj asemănător cu Perl sau C, plus suport pentru manipularea bazelor de date într-un dialect SQL (dBase, Informix, MySQL, mSQL, Oracle, PostgreSQL, Solid, Sybase, ODBC etc.) şi acces la sisteme hipermedia precum Hyperwave. De asemeni, PHP suportă incărcarea fişierelor de pe calculatorul client: upload (standard propus de E. Nebel şi L. Masinter de la Xerox, descris în RFC 1867) şi oferă suport pentru cookies (mecanism de stocare a datelor în navigatorul client pentru identificarea utilizatorilor, propus de Netscape). Această aplicaţie este disponibilă gratuit pe Internet, pentru medii Unix şi mai nou pentru medii Windows (inclusiv sursele), integriându-se în popularul server Apache.

Se estimează că numărul site-urilor care folosesc în prezent PHP este de peste un milion. Deja, pe Web, exista o multitudine de aplicaţii şi utilitare concepute în PHP, care se regăsesc grupate şi în aşa-numitul PEAR (PHP Extension and Add-on Repository).

PHP este un limbaj de script care funcţionează alături de un server Web.

Este de notat diferenţa cu alte limbaje script CGI scrise în limbaje precum Perl sau C: în loc de a scrie un program cu o multitudine de linii de comanda afişate în final într-o pagină HTML, veţi scrie o pagină HTML cu codul inclus pentru a realiza o acţiune precisă.

Codul PHP este inclus între tag-urile speciale de început şi de sfârşit care permit utilizatorului să treacă din "modul HTML" în "modul PHP". Faţă de alte limbaje script, precum Javascript, la PHP codul se execută pe serverul Web, nu in navigator(client Web). In consecinta, PHP poate obtine accesul la fisiere , baze de date si alte resurse inaccesibile programului JavaScript. Acestea contin bogate surse de continut dinamic care atrag vizitatorii. Dacă pe serverul Web se află un script similar, clientul nu va primi decât rezultatul execuţiei scriptului, fără a avea nici o posibilitate de acces la codul care a produs rezultatul. Vă puteţi configura serverul de Web să prelucreze (analizeze) toate fişierele HTML ca fişiere PHP. Astfel nu există nici un mijloc de a distinge paginile care sunt produse dinamic de paginile statice.

Limbajul PHP posedă aceleaşi funcţii ca alte limbaje permiţâind să se scrie scripturi CGI, să colecteze date şi să genereze dinamic pagini web sau să trimită şi să primească cookies. Marea calitate şi cel mai mare avantaj al limbajului PHP este suportul cu un număr mare de baze de date. A realiza o pagină web dinamic cu o baza de date este extrem de simplu. PHP suporta foarte multe tipuri de baze de date, ceea ce este un avantaj foarte mare pentru el. Limbajul PHP are deasemenea suport pentru diverse servicii server utilizând protocoale precum IMAP, SNMP, NNTP, POP3 şi HTTP. Fiecare program PHP include două linii speciale, care indică serverului PHP că textul cuprins între cele doua linii este alcătuit din instrucțiuni PHP.

<?php ?>

Practic aceste linii pot fi asimilate copertelor unei cărți, care păstrează unitatea programului PHP. Programele PHP execută trei categorii de operații elementare:

1. obțin date de la un utilizator;
2. execută prelucrări ale datelor, respectiv obțin accesul la datele stocate în fișiere și baze de date și le manipulează;
3. afisează date astfel încât un utilizator să le poată vizualiza.

Pentru executarea unui script trebuie mai întâi încărcat scriptul într-un server unde este instalat PHP. Accesul la distanță la un server Linux sau Unix se poate face prin intermediul protocoalelor Telnet sau SSH, dacă în prealabil se primește autorizare de folosire a acestor protocoale de la administatorul de sistem pentru un anumit utilizator. Dacă se folosește un server Microsoft Windows situat în aceeași rețea locală ca și stația de lucru atunci este posibilă alocarea de către adimistratorul de rețea a unei partiții de fișiere în acest scop. În vederea încărcării scriptului prin intermediul protocolului FTP adimistratorul de sistem trebuie să furnizeze utilizatorilor autorizați următoarele informații:

* numele serverului gazda;
* numele de utilizator și parola pentru deschiderea sesiunii de lucru prin intermediul protocolului FTP;
* catalogul în care trebuie să se găsească scripturile PHP;
* localizatorul uniform de resurse(URL), pe care trebuie să-l folosiți pentru a obține acces la scripturile PHP. După ce încărcarea scriptului PHP a fost realizată cu succes acesta se lansează în execuție prin dechiderea unui navigator Web(client) cu adresa URL asociată scriptului PHP.

## Serverul Web Apache :

Serverul de Web Apache stă la baza a peste 60% din domenii, potrivit unui studiu Netcraft Web Server. Providerii de Internet și companiile de găzduire de aplicații pe Web apelează deseori la Apache din cauza suportului sau pentru numeroase platforme, capabilităților de găzduire virtuală ușor de implementat și modulelor sale care îi extind capabilitățile.

Caracteristica Apache Portable Routine(APR) optimizează capabilitățile precum administrarea proceselor pentru fiecare sistem de operare, dar permite serverului propriu-zis să ignore distincțiile specifice fiecărei platforme. Acest lucru a îmbunătățit performanța și stabilitatea implementărilor Windows, prin eliminarea emulatorului. Mai mult APR este accesibil și dezvoltatorilor Web care scriu programe multi-platforme în C, care este mai rapid decît limbajele interpretative, precum PHP și Perl, folosite de obicei pentru dezvoltare Web.

La acest lucru se adaugă faptul că serverul propriu-zis este independent de protocoale. Cu toate că este în principal un server HTTP(Web), este proiectat să suporte alternative cum ar fi FTP. O astfel de structură simplifică administrarea și reduce riscurile de securitate. O parte însemnată din atractivitatea Apache o constituie versabilitatea sa. API-ul său deschis a permis dezvoltatorilor să scrie diferite module care i-au schimbat comportamentul. Dacă serverului îi lipsește o funcție de care are nevoie un site, sunt mari șanse ca undeva să fie disponibil un modul plug-in.

Serverele de Web au ca funcționalitate de baza recepționarea de cereri anonime de la clienți și furnizarea de informații într-o maniera dorită a fi eficientă și rapidă. De fapt un server Web este un daemon care accepta conexiuni conforme protocolului HTTP, răspunzând cererilor recepționate de la clienți. Pentru a asigura servicii HTTP, serverul Apache trebuie să fie instalat în sistem(în mod uzual, fiind vorba de un pachet RPM în Linux sau de un program executabil .exe în Windows), iar daemon-ul httpd pornit. Apache este un sistem modular, alcătuit dintr-un server de baza și mai multe module care sunt încărcate dinamic într-un mod similar cu funcționarea modulelor din nucleul Linux.

Apache poate fi configurat cu ajutorul interfeței grafice apache-conf(Apache Configuration Tool). Fișierul de configurare principal este http-conf și este de obicei localizat în directorul /etc/httpd (în versiunile de Linux sau Unix).

În anumite cazuri, este necesar să se restrictioneze accesul la anumite documente, prin intermediul autentificării prin nume de utilizator și parolă sau în funcție de adresa calculatorului clientului Web.

Pentru autentificarea utilizatorilor, vom parcurge doi pași:

1. se creeaza un fișier conținând numele și parolele utilizatorilor care vor avea acces la anumite date de pe serverul Web(în particular Apache);
2. se configurează serverul pentru a seta care resurse vor fi protejate și care sunt utilizatorii având permisiunea accesării lor, după introducerea unei parole valide.

Configurarea serverului se poate realiza fie prin fișierul httpd-conf, fie prin .htacces, indicând o zona protejată, de obicei în funcție de directoarele dorite a fi accesate pe bază de autentificare. Fișierul .htaccess va fi stocat în directorul asupra căruia dorim să modificăm comportamentul implicit al serverului Web. Înainte de a modifica maniera de autentificare din fișierul .htaccess, administratorul serverului Apache va specifica în httpd.conf că autentificările să se realizeze via .htaccess.

De asemenea, Apache oferă posibilitatea de a servi mai multe situri Web simultan, altfel spus, găzduire virtuală(virtual hosting). Există două metode de implementare a găzduirii virtuale: prima bazată pe nume și a doua bazată pe adrese IP. Mașinile virtuale bazate pe adresa utilizează adresa IP a conexiunii pentru a determina mașina virtuală corectă. Astfel pentru fiecare găzduire virtuală bazată pe nume, determinarea mașinii virtuale se face pe baza numelui acestuia.

Găzduirea virtuala bazată pe nume este mai simplu de implementat, și este recomandată utilizarea acesteia. Pentru a utiliza serviciul de găzduire virtuala, trebuie mai întâi stabilite adresa IP și portul pentru serverul care va accepta cereri pentru respectiva mașina virtuală.

## Administrarea Bazelor de Date cu MySQL

Dezvoltat de firma suedeză MySQL AB, MySQL este un server de baze de date disponibil gratuit, cu sursa deschisă(open-source) care oferă fiabilitate și avantaje reale. Frumusețea serverului MySQL este simplitatea sa fundamentală. Administrarea bazei de date se face folosind peste o duzină de utilitare în linia de comandă, cel mai important fiind mysql, un shell interactiv pentru controlul și interogarea bazei de date. Utilitarele rulează cel mai bine pe sistemul Linux, platforma pe care MySQL a fost dezvoltat inițial. Alte două utilitare cu sursa deschisă, oferite pe platforma Windows, care oferă un set de comenzi de administrare sunt MySql Manager, un utilitar de interogare în mod grafic similar cu mysql și WinMySQL admin, o consolă pentru administrarea detaliilor configurării lui MySQL.MySQL realizează cu ușurință importarea diverselor baze de date și exportarea sub forma fisierelor sql. Tranzacțiile nu sunt parte a tabelelor implicite(ISAM) ale lui MySQL, dar sunt incluse două tipuri noi de tabele . Berkley DB(BDB) și InnoDB, care au fost dezvoltate de alte firme. Administratorii de firme și dezvoltatorii de baze de date au ajuns la concluzia că MySQL are o dezvoltare energică și loială, ca să nu mai menționăm reputația câștigată cu trudă, de baza de date sigură.

MySQL operează în baza unui model client/server. Orice mașina care dorește să proceseze interogări asupra unei baze de date MySQL trebuie să ruleze MySQL server(mysqld), care este responsabil de tot traficul de tip intrari/iesiri(incoming/outgoing) cu baza de date. Ca orice server, mysqld "ascultă" pe un port particular (3306) eventualele cereri de conexiune ale unui "client" - orice aplicație care trimite cereri către o bază de date via mysqld. Acest client poate fi un simplu script PHP care poate trimite o cerere către baza de date prin intermediul serverului MySQL sau chiar clientului mysql. Clientul mysql este o interfață interactiva pentru transmiterea de comenzi către server.

Modelul de securitate folosit de MySQL se bazează pe nume de utilizator, parola, nume server(hostname) sau adresa de IP și privilegii, fiind similar celui generic folosit de sistemele Unix. Prin privilegii se înteleg în cazul MySQL operațiunile ce vor fi permise asupra bazei/bazelor de date, tabelelor sau indecsilor, cum sunt de exemplu SELECT, INSERT, UPDATE, DELETE, CREATE, DROP.

Datele sunt obiectul celor mai multe operații de prelucrare, iar sistemele de gestiune a bazelor de date furnizează cele mai complexe și puternice facilități pentru lucrul cu datele. PHP include o biblioteca de functii care furnizeaza o interfață cu sistemul MySQL de gestiune a bazelor de date. Folosind aceste funcții, un program PHP poate obține accesul la datele rezidente într-o bază de date MySQL și le poate modifica.

Majoritatea interacțiunilor cu o bază de date MySQL se poate desfășura după un model secvențial simplu:

* se deschide o conexiune cu serverul MySQL;
* se specifica baza de date la care se obține accesul;
* se emit interogări SQL, se obține accesul la rezultatele interogării și se execută operații non-SQL
* se închide conexiunea cu serverul MySQL

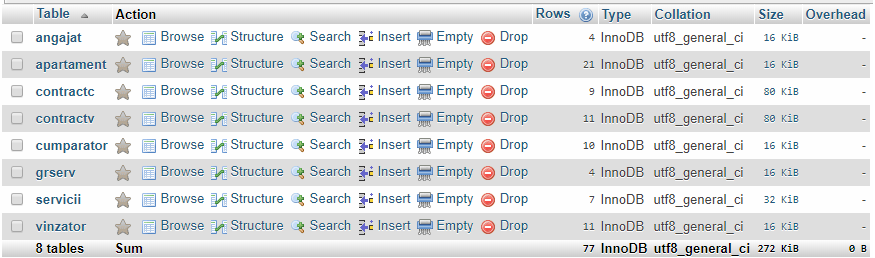
Din punct de vedere al limbajului PHP, există două categorii de interogări SQL:

1. interogările SELECT, care returnează rânduri ale unui tabel,
2. interogările UPDATE, INSERT și DELETE, care nu returnează rânduri ale unui tabel.

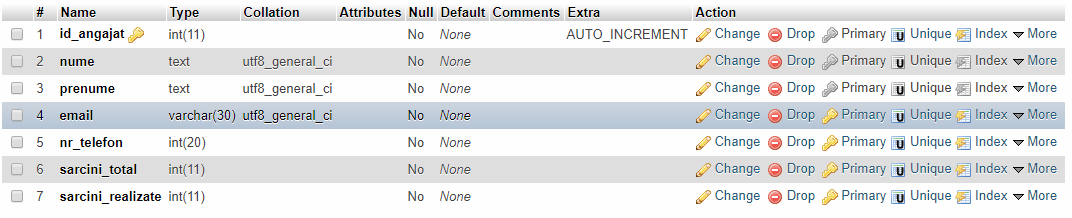
Ambele categorii de interogari sunt emise folosind funcția mysql\_query(), dar verificarea și prelucrarea celor două categorii de rezultate ale interogărilor sunt procesate destul de diferit.

**4.Capitolul IV (Structura Bazei de date)**

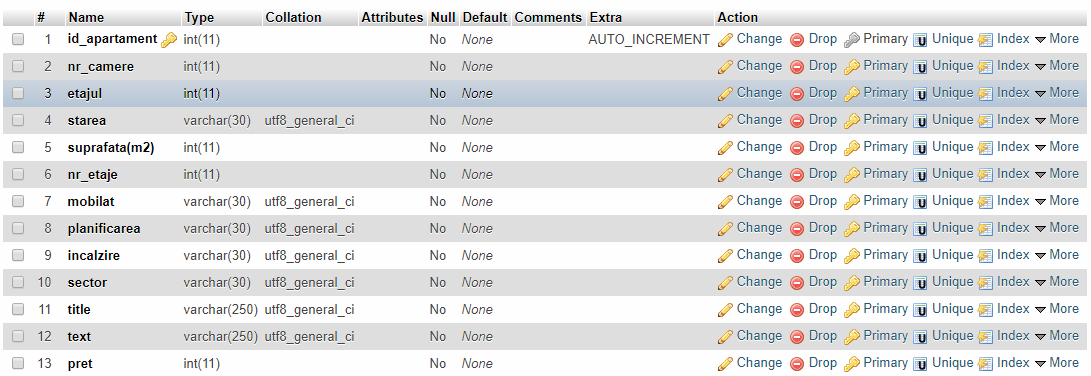
## Machetul Bazei de Date:



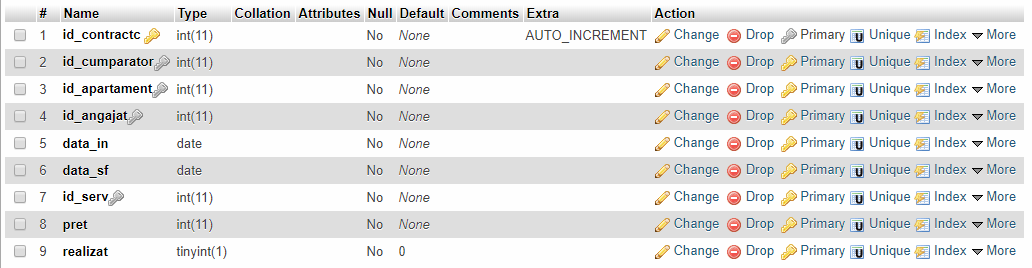
## Tabelul 1 „Angajat”:



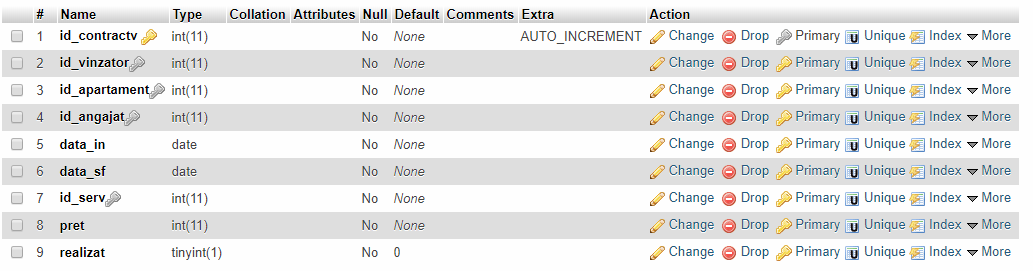
## Tabelul 2 „Apartament”:



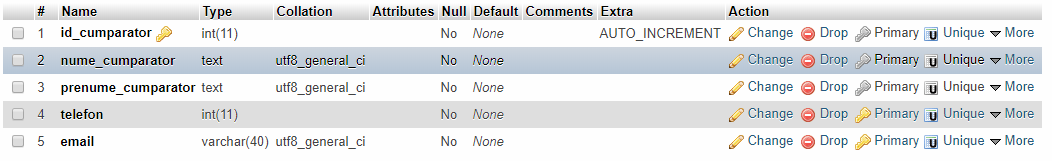
## Tabelul 3 „ContractC”:



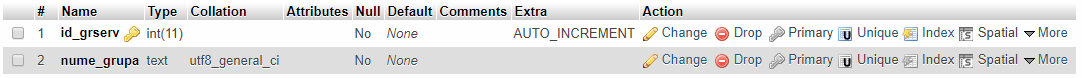
## Tabelul 4 „ContractV”:



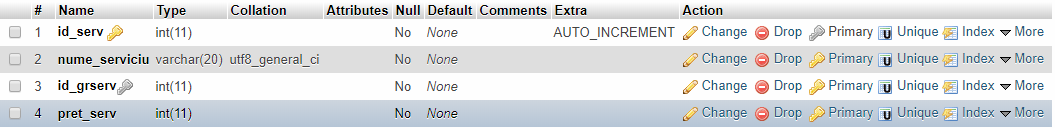
## Tabelul 5 „Cumparator”:



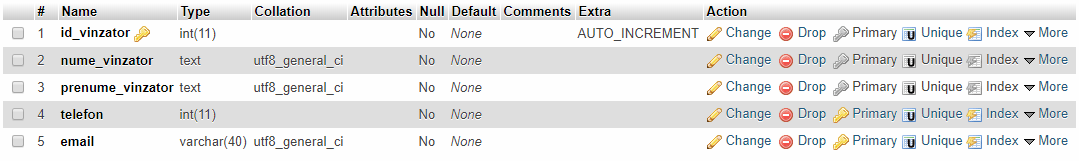
## Tabelul 6 „GrServ”:



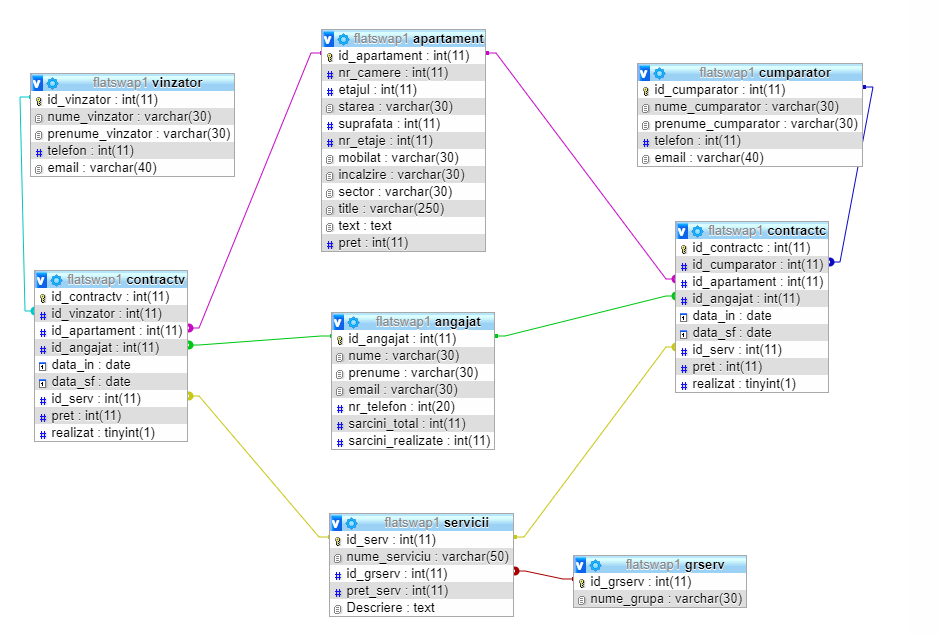
## Tabelul 7 „Serv”:



## Tabelul 8 „Vinzator”:

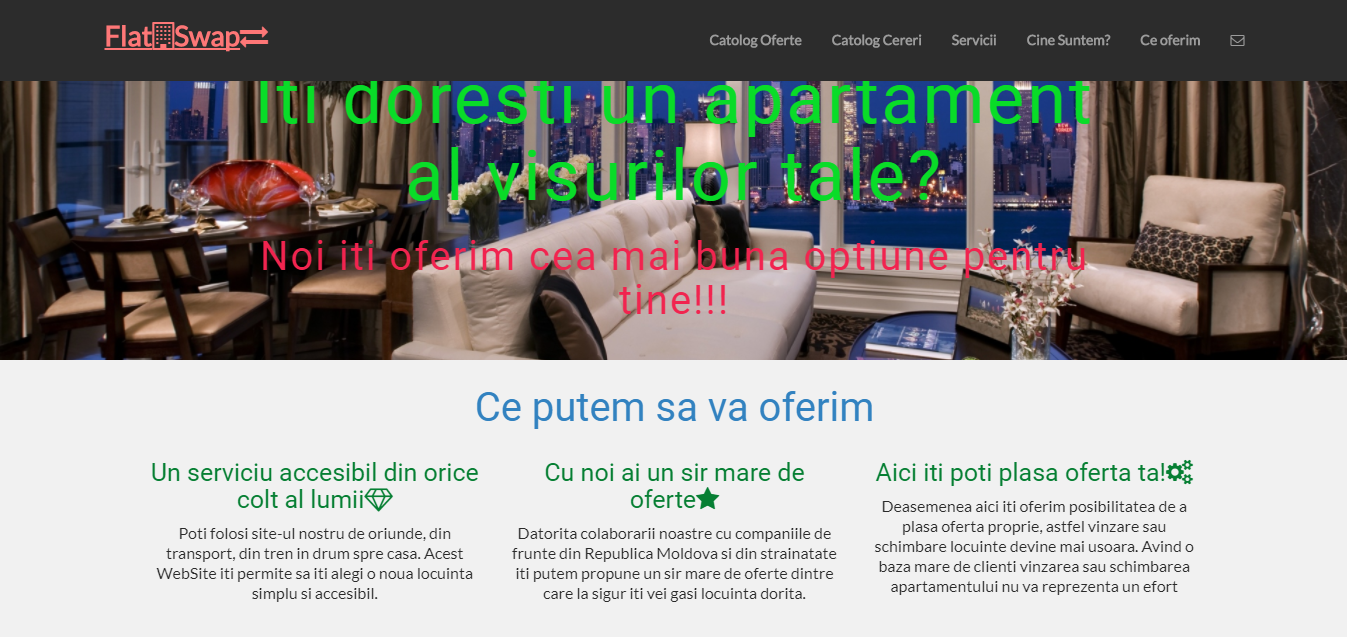


**Schema ER a bazei de date in phpMyAdmin**

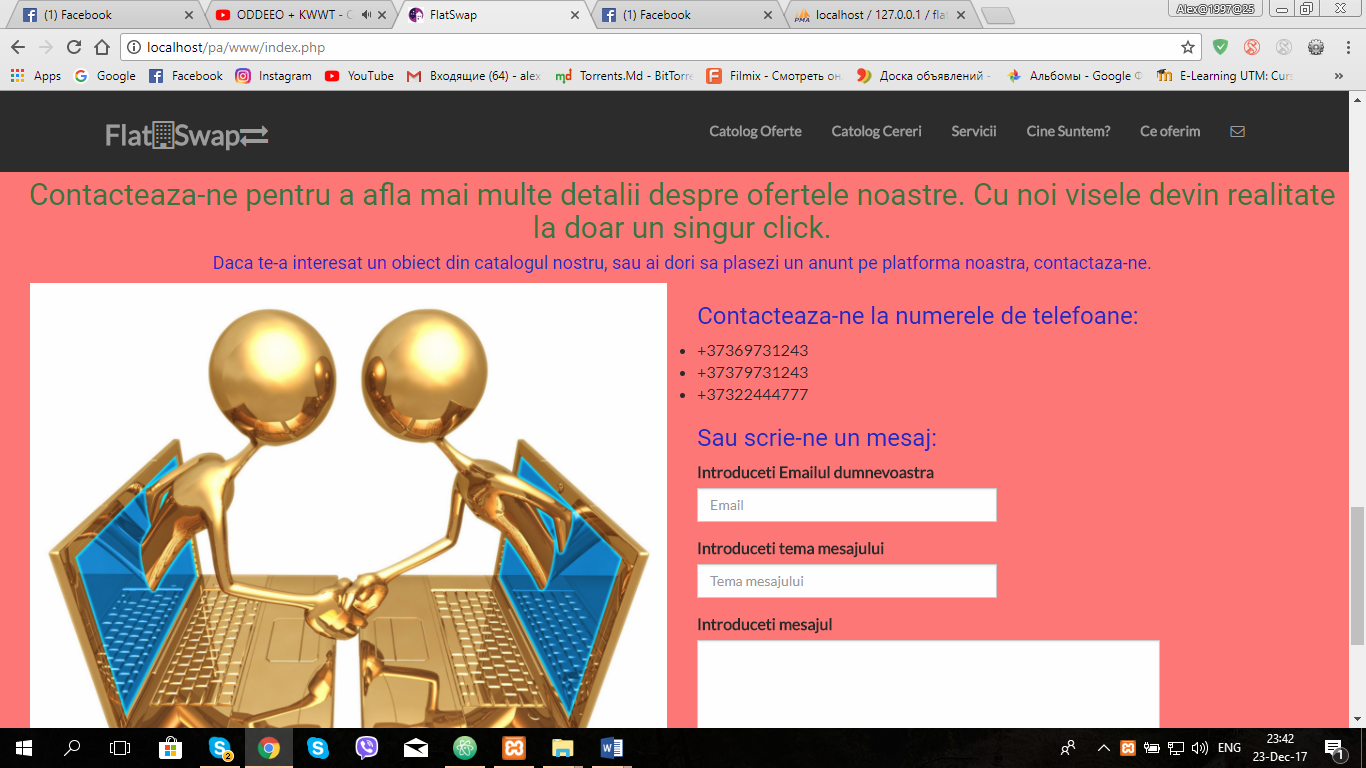


**Capitolul V (Structura SITE-ului)**

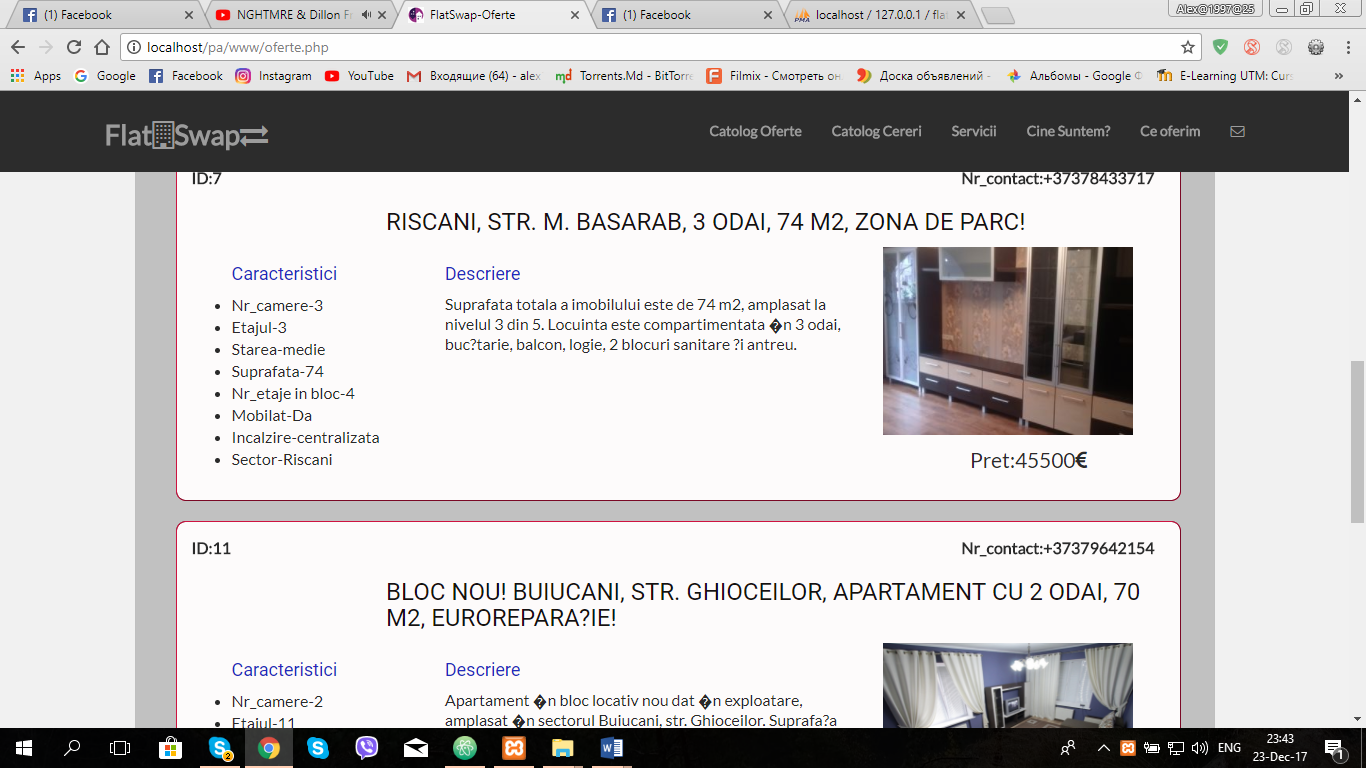
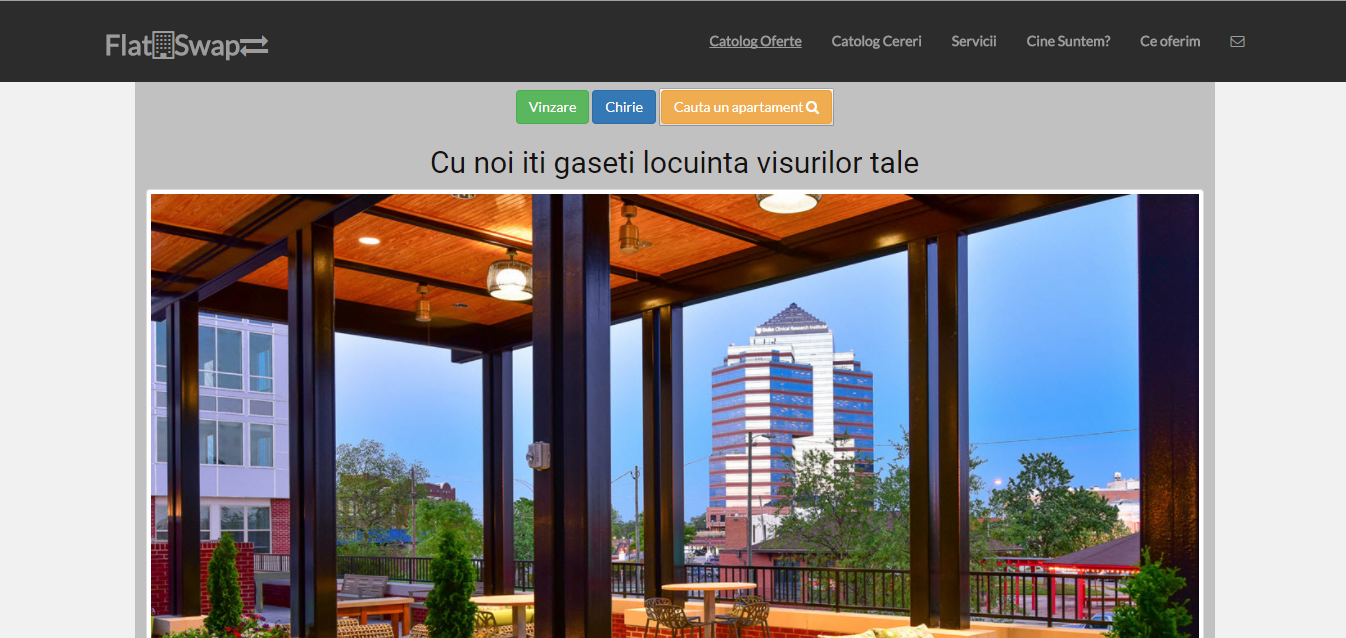
## *5.1 Scopul SITE-ului »HOME« :*



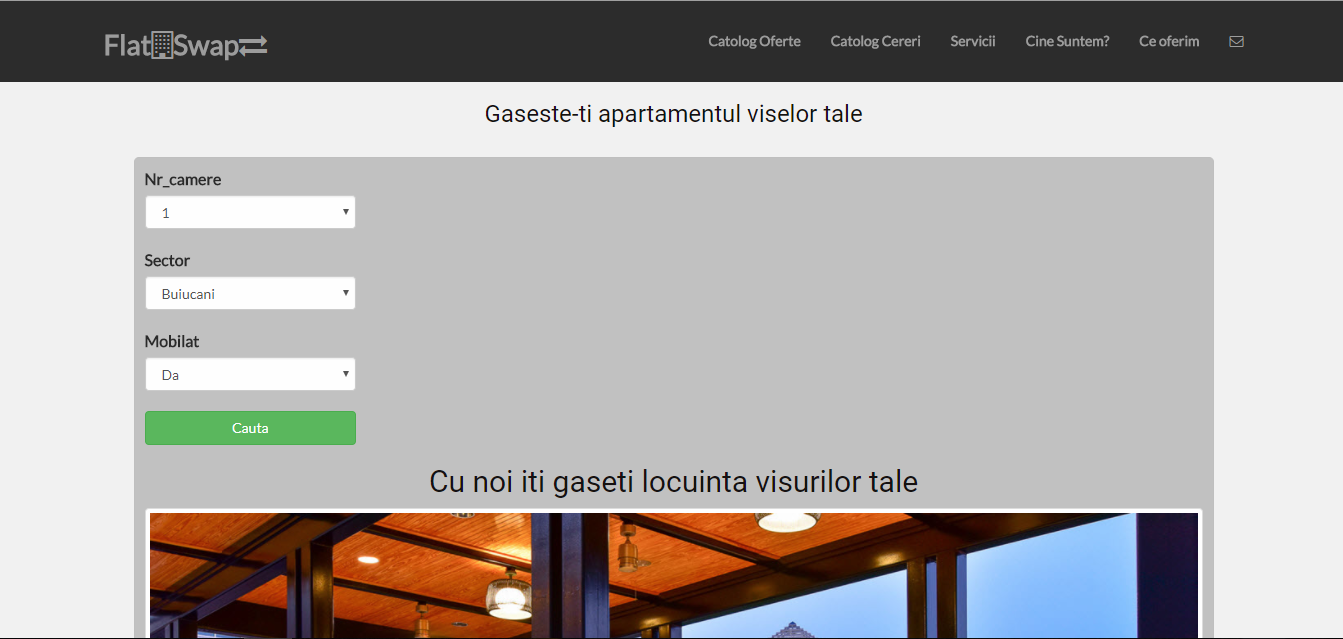
## *5.2 Blocul pentru contact cu interpriderea*



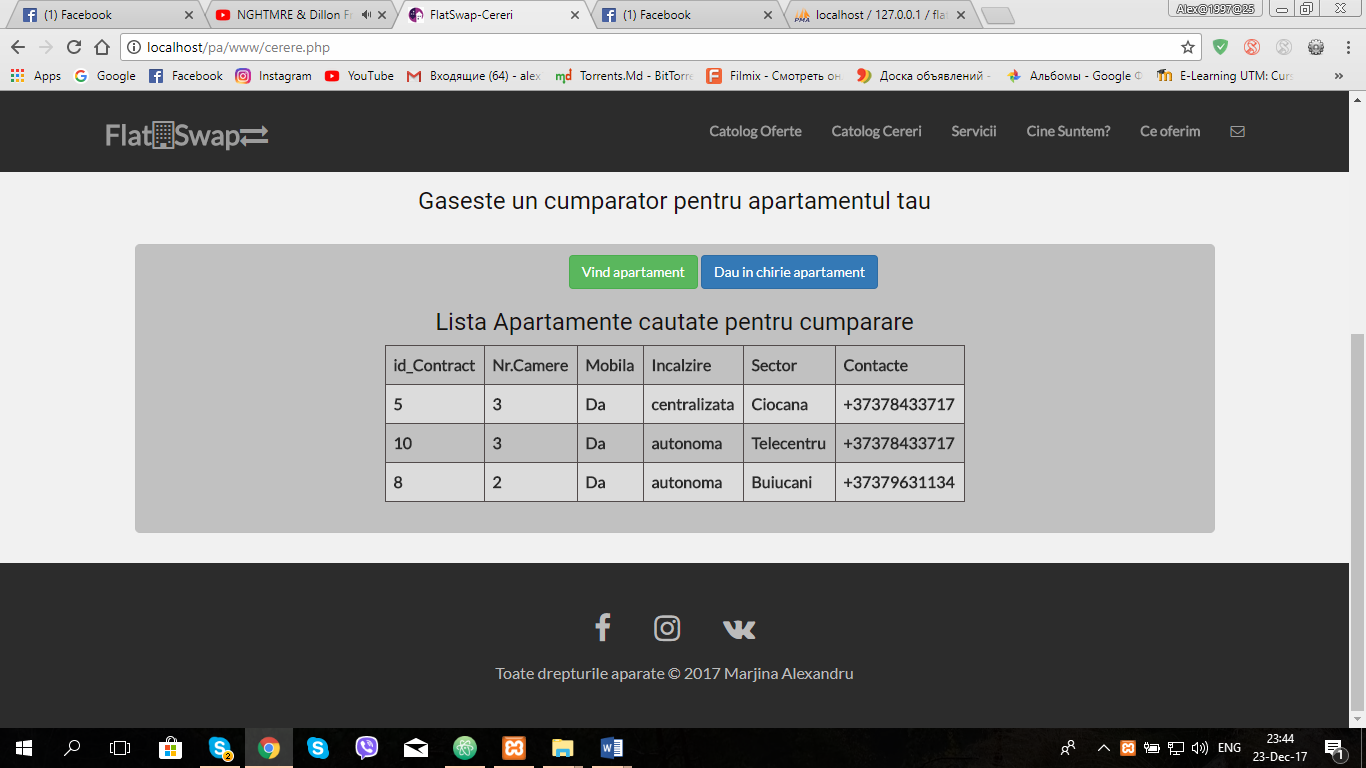
## *5.3 Pagina Catalog Oferte*



## *5.4 Pagina Catalog Cautare apartament*



## *5.5 Pagina Catalog Cereri*



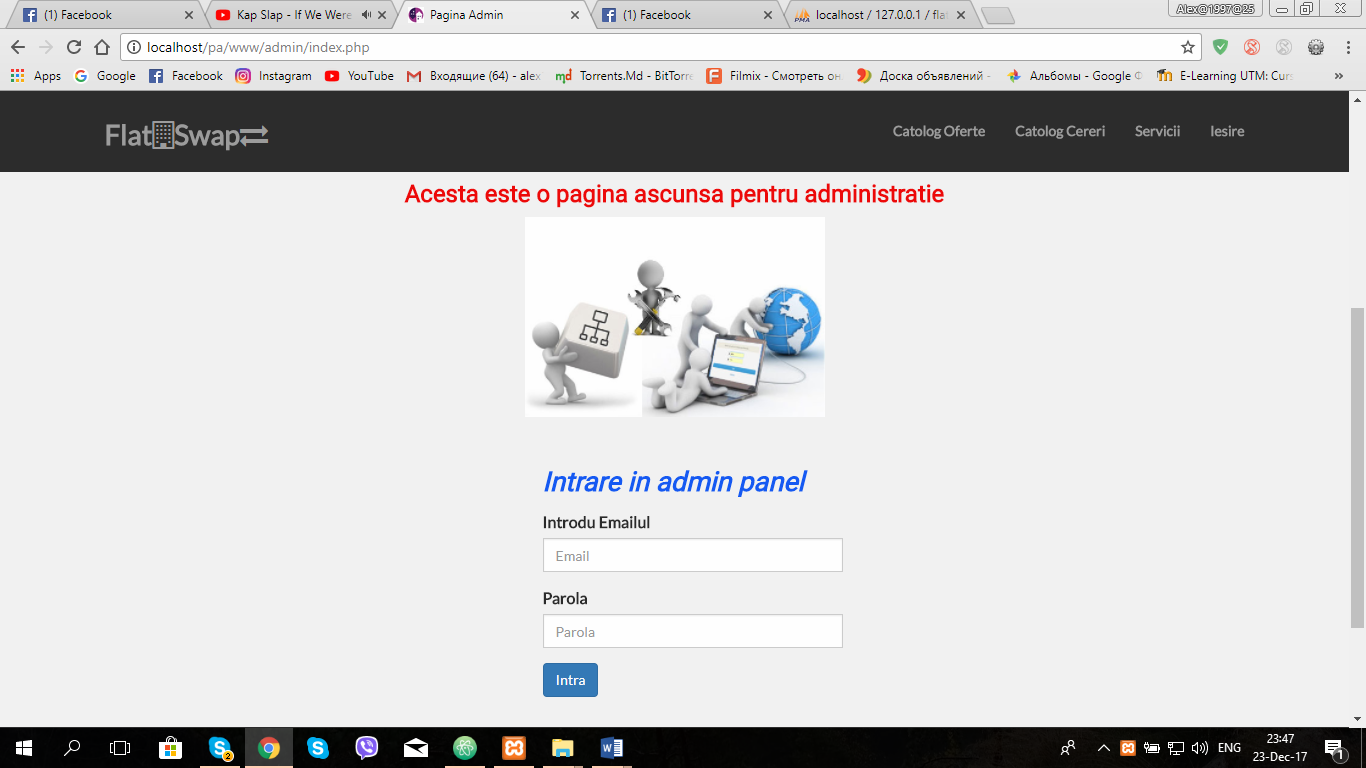
## *5.5 Pagina Servicii*



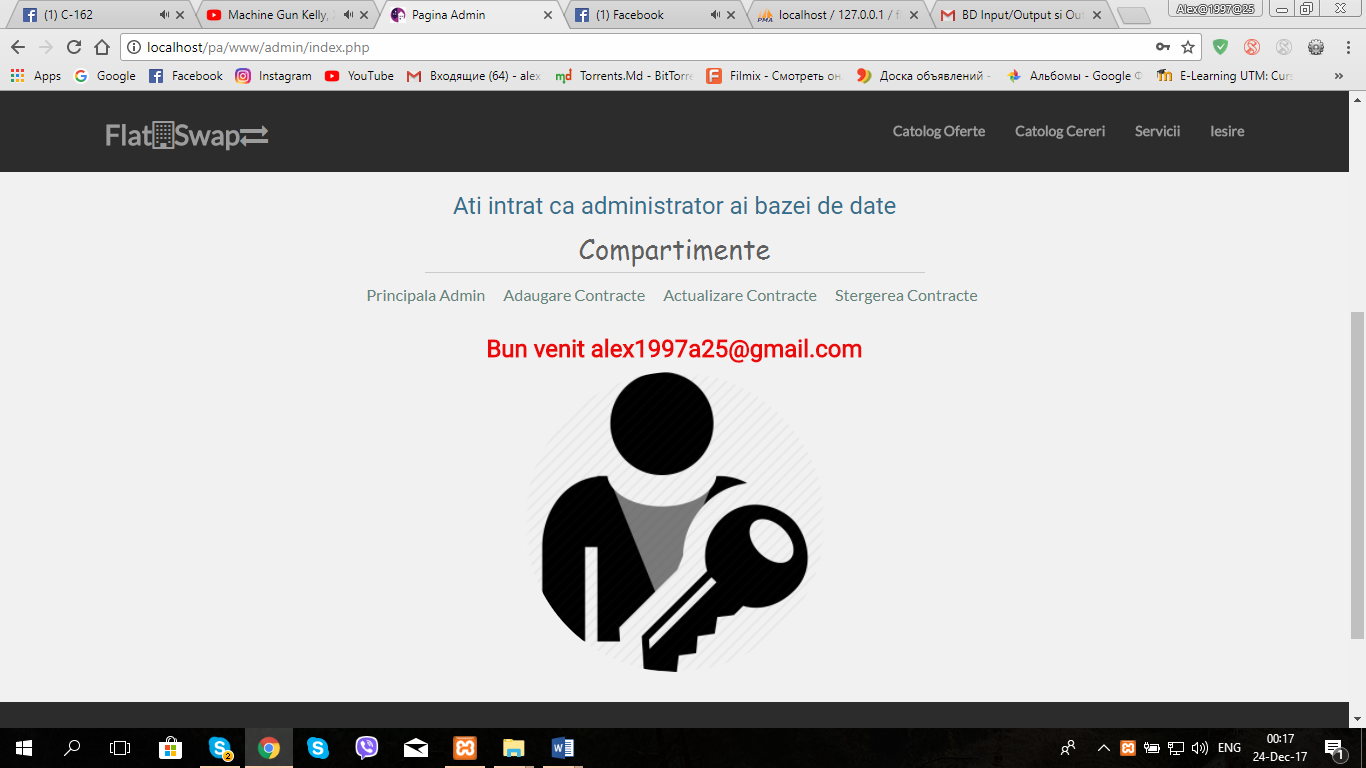
## *5.6 Intrarea in pagina admin este ascunsa in partea de jos a Sitului : este un link care rederictioneaza la admin*



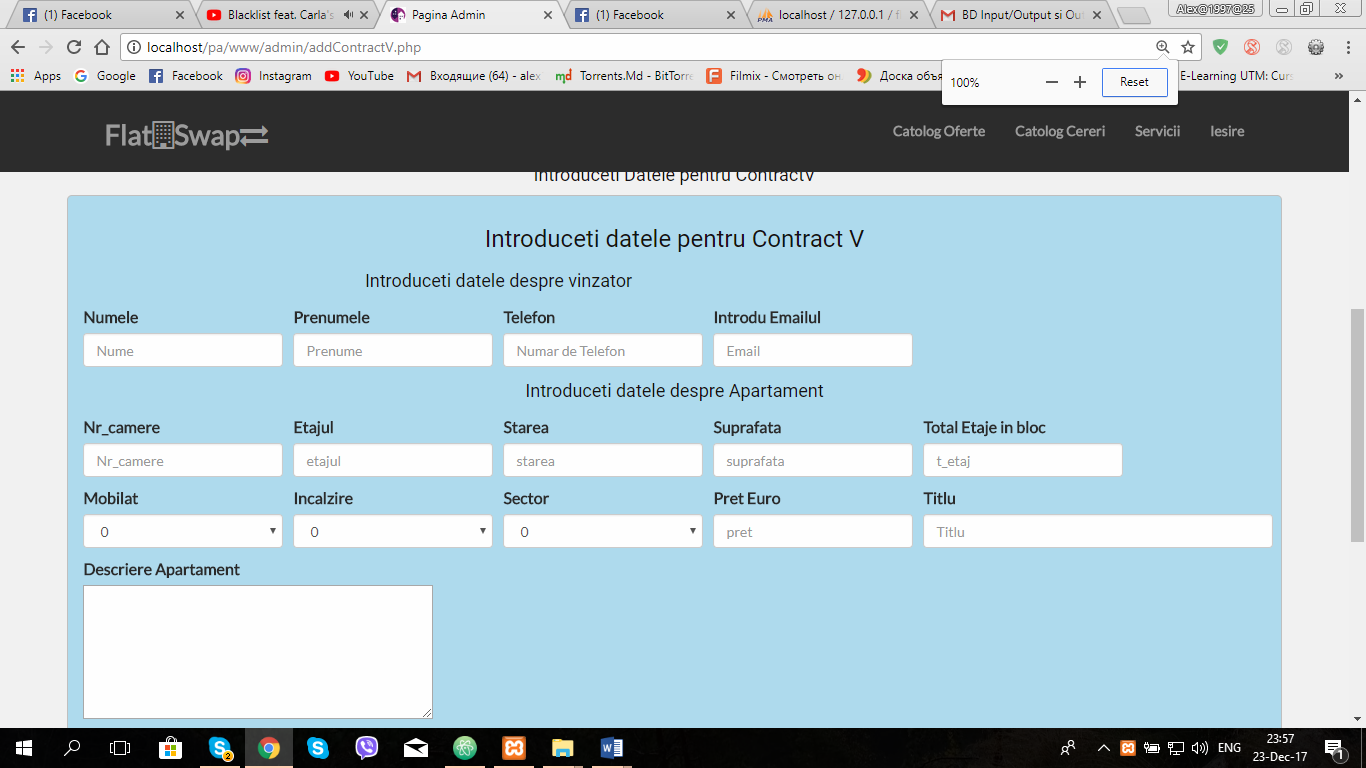
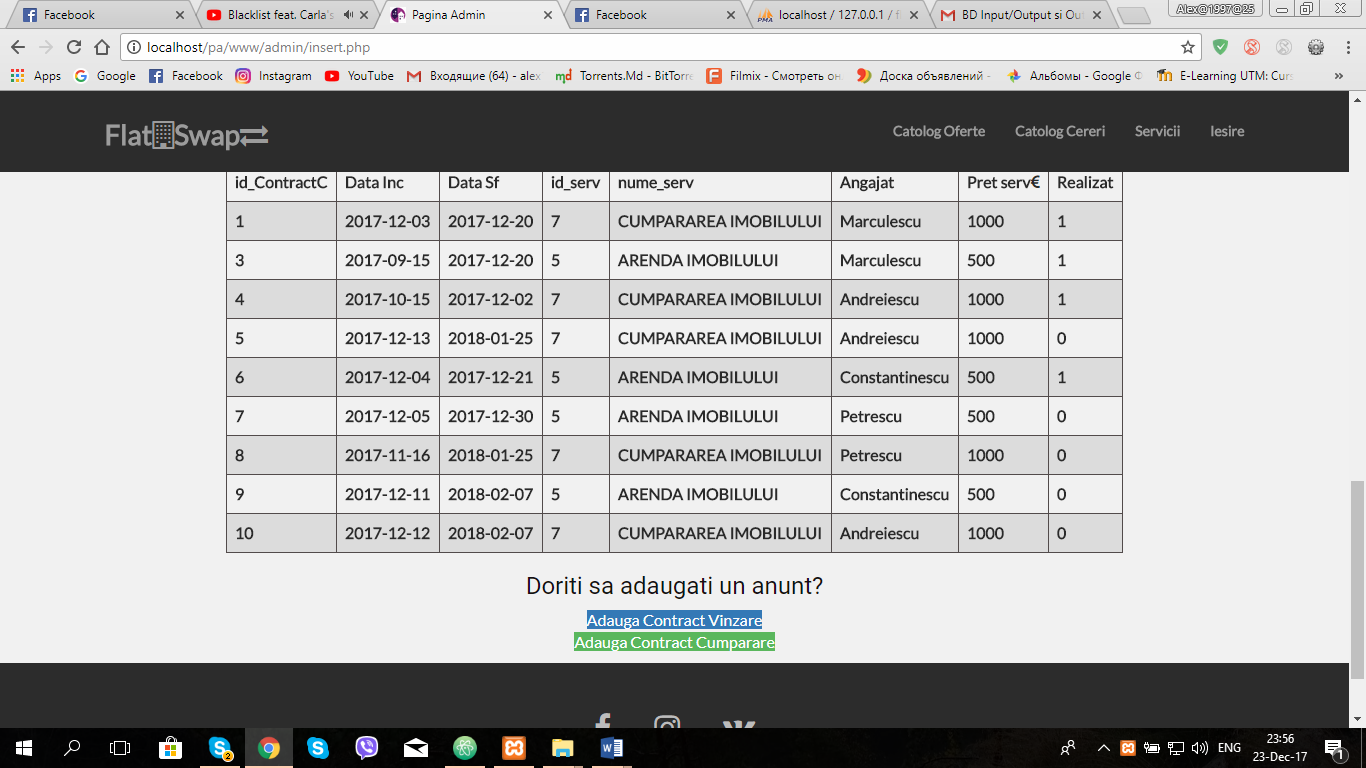
## *5.7 Pagina Access Admin*



## *5.8 Partea Administorului*



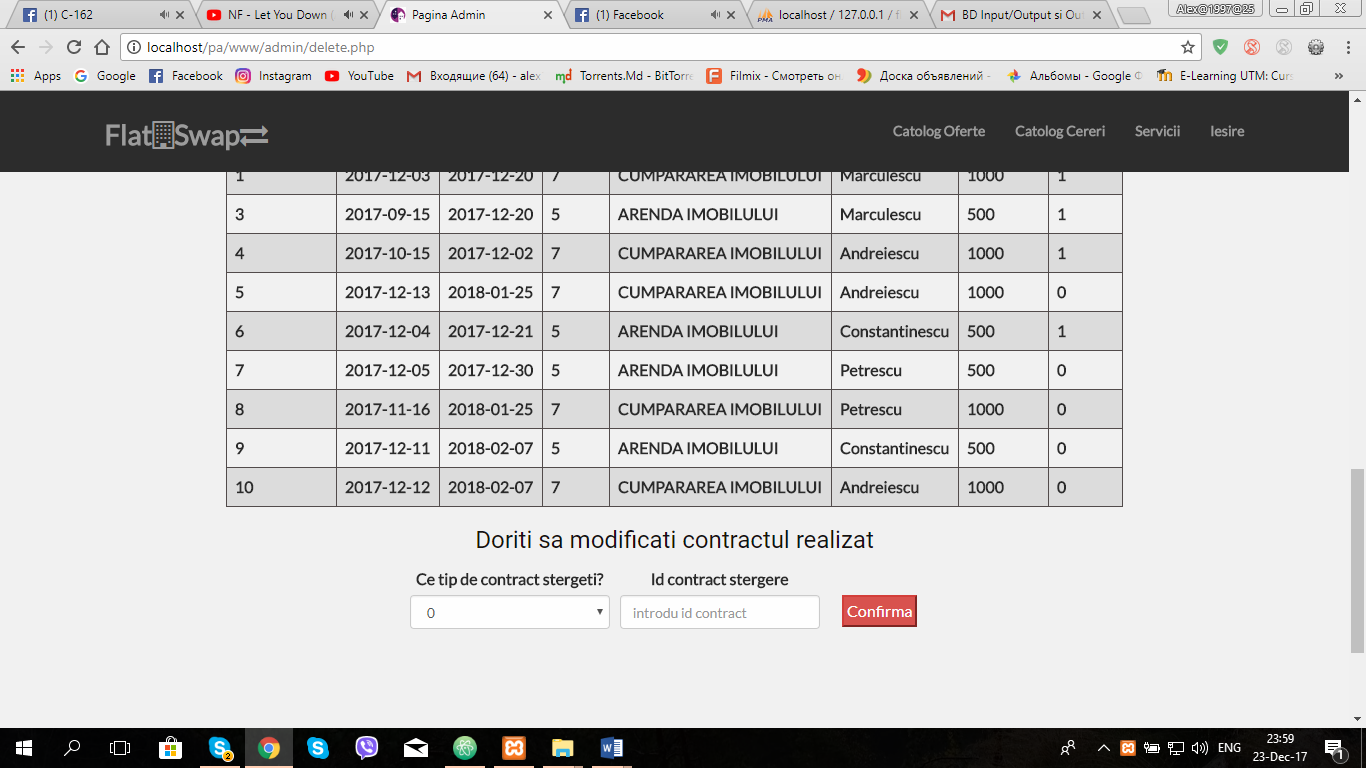
## *5.9 Partea Administorului>>Adugarea*



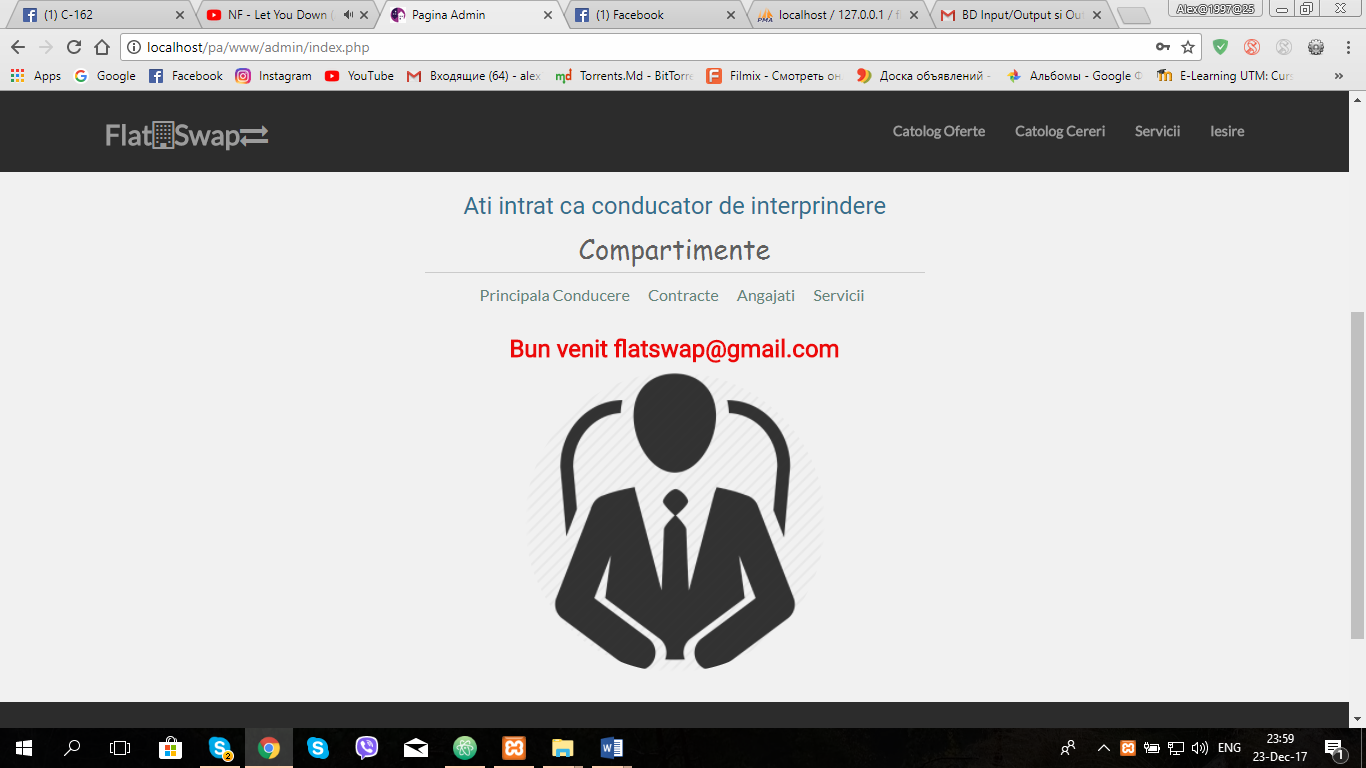
## *5.10 Partea Administorului>>Modificare*



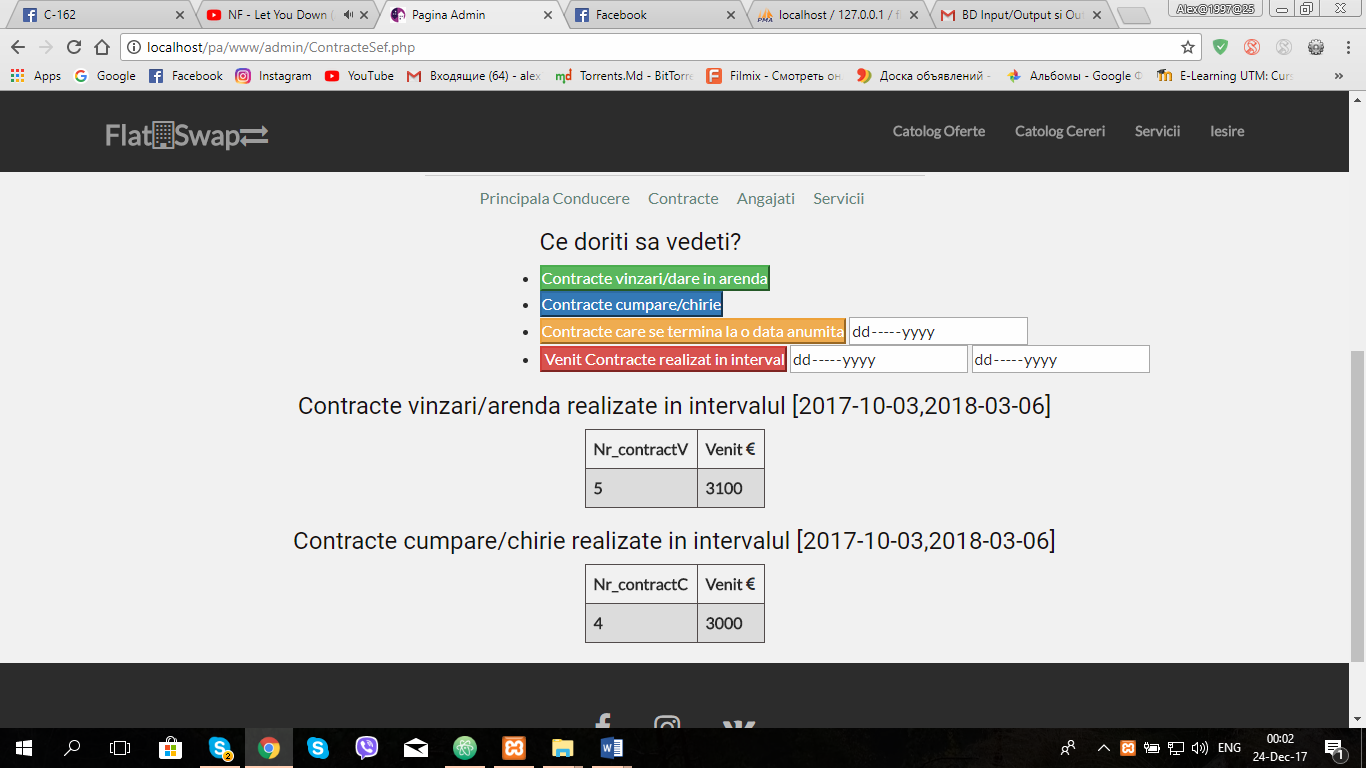
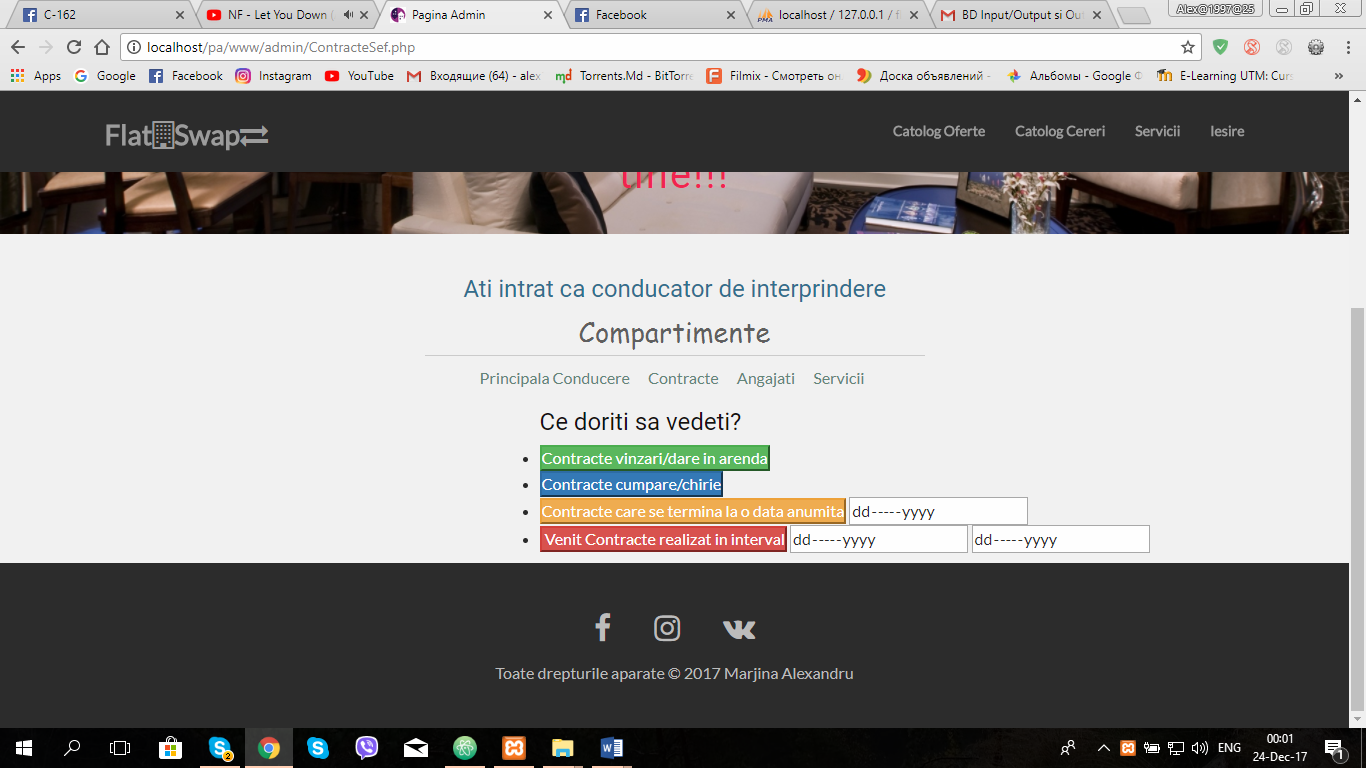
## *5.11 Partea Administorului>>Stergere*



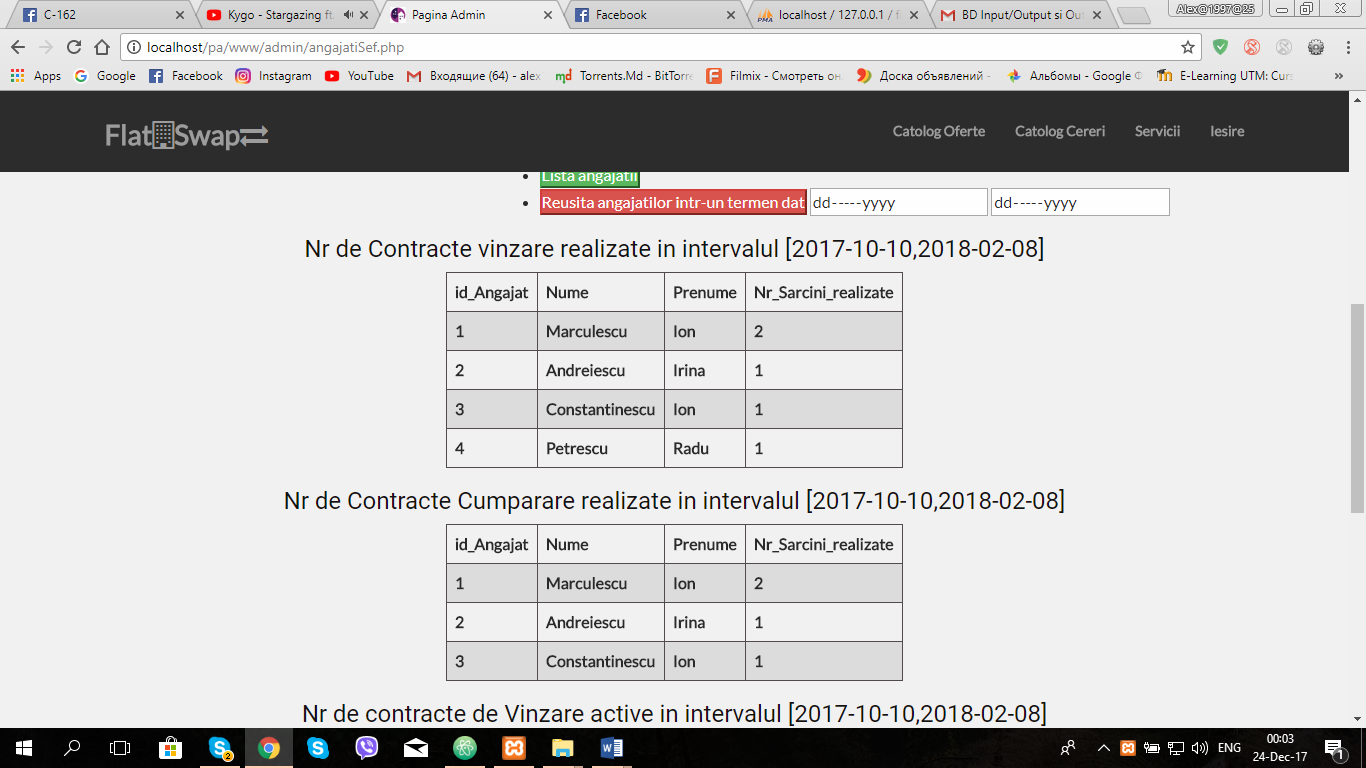
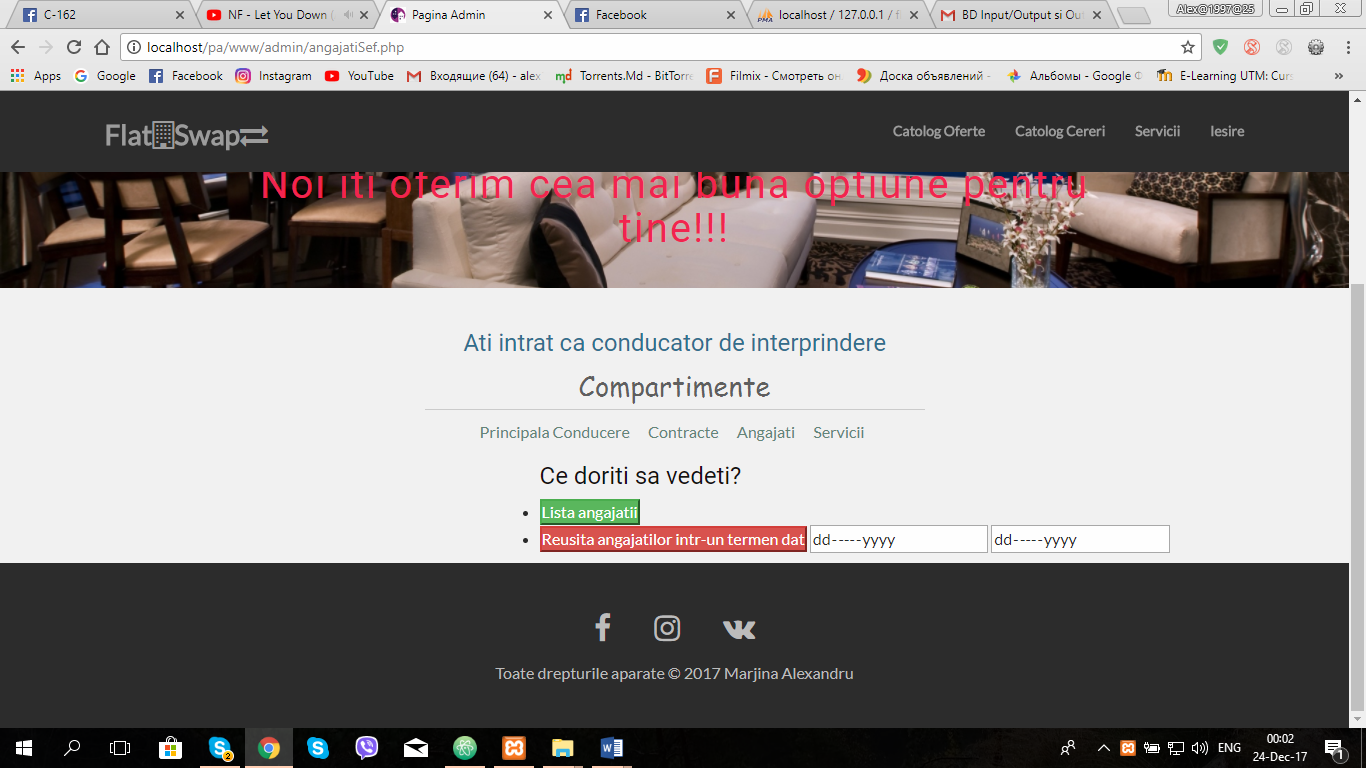
## *5.12 Partea Sefului*



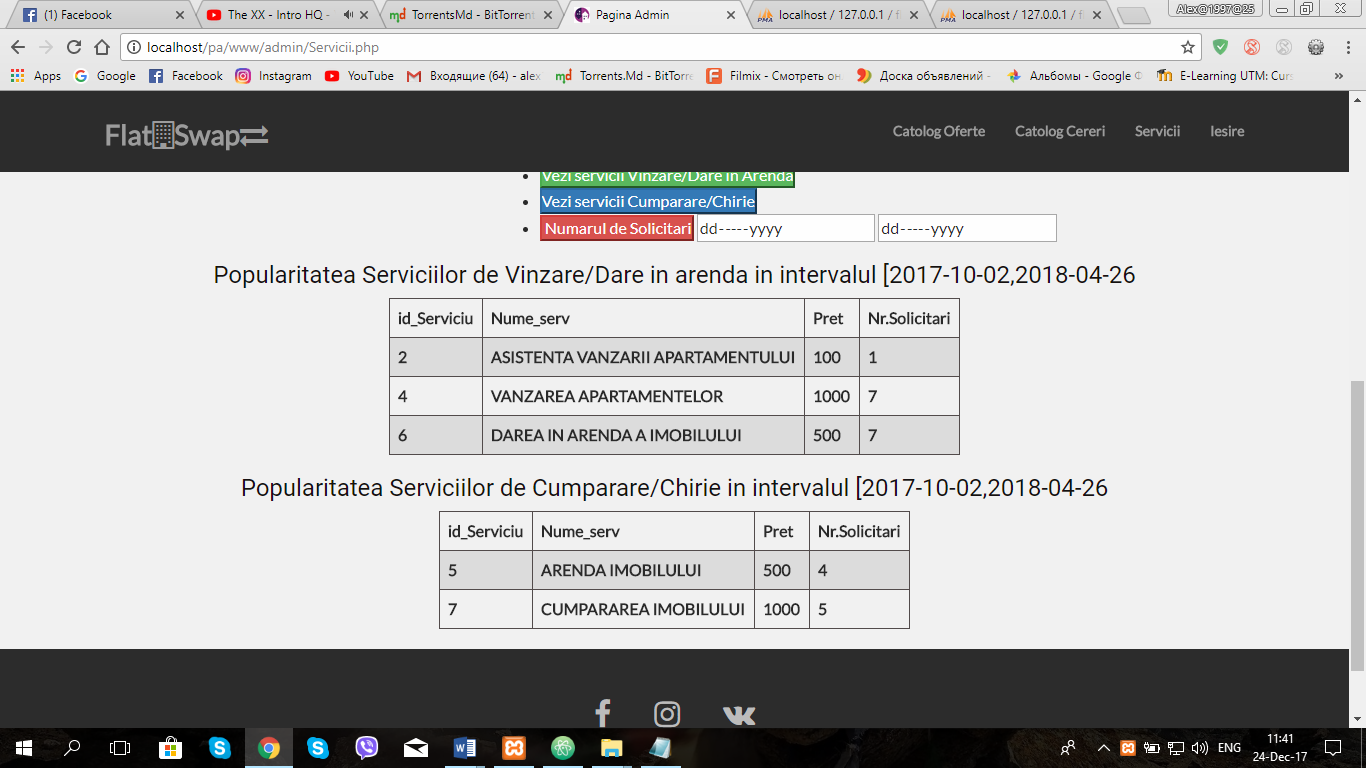
## *5.13 Partea Sefului>>Contracte*



## *5.14 Partea Sefului>>Angajati*

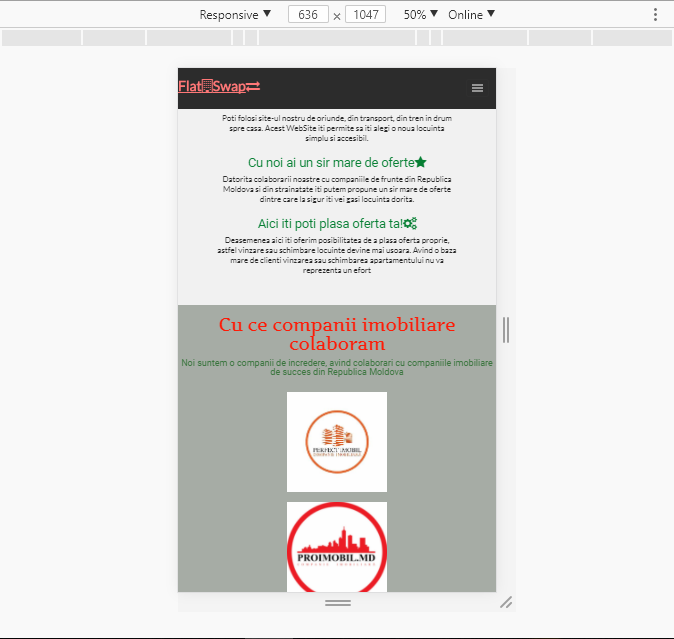


## *5.15 Partea Sefului>>Servici*



## *5.16 Efecte speciale ale sitului*

Exista imagini in pagina principala care fiind pointate cu cursorul se rotesc putin si se maresc, deasemenea situl este responsive.



## *5.17 Tehnologiile folosite*

Pentru implementarea acestei aplicatii am folosit tehnologiile HTML, CSS, BOOTSTRAP, JAVAsCRIPT, PHP, MySQL si aplicatia pentru host XAMPP.

**Capitolul VI (Listing Program)**

**Putin din codul sitului**

**Index.php**

**<?php**

**session\_start();**

**unset($\_SESSION["to"]);**

**unset($\_SESSION["from"]);**

**unset($\_SESSION["subject"]);**

**unset($\_SESSION["message"]);**

**$\_SESSION["from"]="";**

**$\_SESSION["subject"]="";**

**$\_SESSION["message"]="";**

**if(isset($\_POST["send"])){**

**$from=htmlspecialchars($\_POST["from"]);**

**$to="alex1997a25@gmail.com";**

**$subject=htmlspecialchars($\_POST["subject"]);**

**$message=htmlspecialchars($\_POST["message"]);**

**$\_SESSION["to"]=$to;**

**$\_SESSION["from"]=$from;**

**$\_SESSION["subject"]=$subject;**

**$\_SESSION["message"]=$message;**

**$error=false;**

**if($from==""){**

**echo "<script>alert('Nu ati introdus adresa de email')</script>";**

**$error=true;**

**}**

**if(strlen($subject)==0){**

**echo "<script>alert('Nu ati introdus tema mesajului')</script>";**

**$error=true;**

**}**

**if(strlen($message)==0){**

**echo "<script>alert('Introduceti mesajul')</script>";**

**$error=true;**

**}**

**if(!$error){**

**$subject="?utf-8?B?".base64\_encode($subject)."?=";**

**$headers="From: $from\r\nReply-to: $from\r\nContent-type:text/plain;**

**charset=utf-8\r\n";**

**$test=mail($to,$subject,$message,$headers);**

**if($test){**

**echo "<script>alert('Ati trimis mesajul cu succes')</script>";**

**header("Location: index.php");**

**exit;**

**}**

**else {**

**echo "<script>alert('A aparut o eroare la trimiterea mesajului')</script>";**

**header("Location: index.php");**

**exit;**

**}**

**}**

**}**

**?>**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<?php**

**$title="FlatSwap";**

**require\_once "blocks/head.php";**

**?>**

**</head>**

**<body>**

**<?php**

**require\_once "blocks/header.php";**

**?>**

**<div class="container of" id="offer">**

**<div class="row centered">**

**<h3>Ce putem sa va oferim</h3>**

**<div class="col-lg-4">**

**<h5>Un serviciu accesibil din orice colt al lumii<i class="fa fa-diamond" aria-hidden="true"></i></h5>**

**<p> Poti folosi situl nostru de oriunde, din transport, din tren in drum spre casa. Acest WebSite iti permite sa iti alegi o noua locuinta simplu si accesibil. </p>**

**</div>**

**<div class="col-lg-4 ">**

**<h5>Cu noi ai un sir mare de oferte<i class="fa fa-star" aria-hidden="true"></i></h5>**

**<p>Datorita colaborarii noastre cu companiile de frunte din Republica Moldova si din strainatate iti putem propune un**

**sir mare de oferte dintre care la sigur iti vei gasi locuinta dorita.</p>**

**</div>**

**<div class="col-lg-4">**

**<h5>Aici iti poti plasa oferta ta!<i class="fa fa-cogs" aria-hidden="true"></i></h5>**

**<p>Deasemenea aici iti oferim posibilitatea de a plasa oferta proprie, astfel vinzare sau schimbare locuinte devine mai usoara.**

**Avind o baza mare de clienti vinzarea sau schimbarea apartamentului nu va reprezenta un efort</p>**

**</div>**

**</div>**

**</div>**

**<div class="container pr">**

**<div class="row centered">**

**<h3>Cu ce companii imobiliare colaboram </h3>**

**<h4 class="text-success">Noi suntem o companii de incredere, avind colaborari cu companiile imobiliare de succes din Republica Moldova</h4>**

**<div class="col-lg-2">**

**<div class="tilt">**

**<img src="img\perfect.jpg" alt="perfectimobil">**

**</div>**

**</div>**

**<div class="col-lg-2">**

**<div class="tilt">**

**<img src="img\proimobil.jpg" alt="proimobil">**

**</div>**

**</div>**

**<div class="col-lg-2">**

**<div class="tilt">**

**<img src="img\accesimobil.jpg" alt="accesimobil">**

**</div>**

**</div>**

**<div class="col-lg-2">**

**<div class="tilt">**

**<img src="img\axaimobil.jpg" alt="axaimobil">**

**</div>**

**</div>**

**<div class="col-lg-2">**

**<div class="tilt">**

**<img src="img\ipremier.jpg" alt="ipremier">**

**</div>**

**</div>**

**<div class="col-lg-2">**

**<div class="tilt">**

**<img src="img\dansicons.png" alt="dansicons">**

**</div>**

**</div>**

**</div>**

**</div>**

**<div class="container r" id="about">**

**<div class="row centered">**

**<h3>Cine suntem ?</h3>**

**<p class="text-primary">**

**Suntem o companie imobiliara care iti indeplinim visele. Alaturi de noi iti poti gasi apartamentul visurilor tale .**

**Noi iti oferim o multime de oferte, in care fiecare i-si va gasi apartamentul pe care mereu si l-a dorit.**

**Datorita colaborarii cu noi, cautarea unei locuinte dorite se rezuma la contactarea noastra, restul lucrului il facem noi,**

**de la cautarea apartamentului cu caracteristicile dorite pina la colectarea tuturor documentelor necesare.**

**</p>**

**<img class="img-responsive img-rounded" src="img\comunity.jpg" alt="Our Team">**

**</div>**

**</div>**

**<div class="container con" id="contact">**

**<div class="row">**

**<h2 class="text-success centered">Contacteaza-ne pentru a afla mai multe detalii despre ofertele noastre.**

**Cu noi visele devin realitate la doar un singur click.</h2>**

**<h4 class="centered">Daca te-a interesat un obiect din catalogul nostru, sau ai dori sa plasezi un anunt pe platforma noastra, contactaza-ne.</h4>**

**<div class="col-lg-6">**

**<img class="img-responsive" src="img\deal.jpg" alt="">**

**</div>**

**<div class="col-lg-6">**

**<h3>Contacteaza-ne la numerele de telefoane:</h3>**

**<ul>**

**<li>+37369731243</li>**

**<li>+37379731243</li>**

**<li>+37322444777</li>**

**</ul>**

**<h3>Sau scrie-ne un mesaj:</h3>**

**<form role="form" class="form-vertical fc" method="post">**

**<div class="form-group">**

**<label for="InputEmailFirst">Introduceti Emailul dumnevoastra</label>**

**<div class="input-group">**

**<input type="email" class="form-control" id="from" name="from" placeholder="Email"**

**value="<?=$\_SESSION['from']?>">**

**</div>**

**</div>**

**<div class="form-group">**

**<label for="InputSubject">Introduceti tema mesajului</label>**

**<div class="input-group">**

**<input type="text" class="form-control" id="subject" name="subject" placeholder="Tema mesajului"**

**value="<?=$\_SESSION['subject']?>">**

**</div>**

**</div>**

**<div class="form-group">**

**<label for="message">Introduceti mesajul</label>**

**<div class="input-group">**

**<textarea name="message" id="message" class="form-control" rows="5" cols="60">**

**<?=$\_SESSION['message']?></textarea>**

**</div>**

**</div>**

**<input type="submit" name="send" id="send" value="Trimite" class="btn btn-success ">**

**</form>**

**</div>**

**</div>**

**</div>**

**<?php**

**require\_once "blocks/footer.php" ;**

**?>**

**Head.php**

**<meta http-equiv="Content-Type" content="text/html; charset=utf-8">**

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

**<title><?php echo $title ?></title>**

**<?php require\_once 'function/function.php' ?>**

**<link rel="stylesheet" href="css/bootstrap.css">**

**<link rel="stylesheet" href="css/font-awesome.min.css">**

**<link rel="stylesheet" href="css/main1.css" type="text/css">**

**<link href="img/favicon.ico" rel="shortcut icon" type="image/x-icon" />**

**<script type="text/javascript" src="js/jquery-3.2.1.min.js"></script>**

**<script src="js/bootstrap.min.js" integrity="sha384-Tc5IQib027qvyjSMfHjOMaLkfuWVxZxUPnCJA7l2mCWNIpG9mGCD8wGNIcPD7Txa" crossorigin="anonymous"></script>**

**<script type="text/javascript">**

**</script>**

**Header.php**

**<div class="navbar navbar-inverse navbar-fixed-top">**

**<div class="container">**

**<div class="navbar-header">**

**<button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">**

**<span class="icon-bar"></span>**

**<span class="icon-bar"></span>**

**<span class="icon-bar"></span>**

**</button>**

**<a id="logo" href="index.php" class="navbar-brand" title="Pagina Principala">Flat<i class="fa fa-building-o" aria-hidden="true"></i>Swap<i class="fa fa-exchange" aria-hidden="true"></i></a>**

**</div>**

**<div class="navbar-collapse collapse">**

**<ul id="ch" class="nav navbar-nav navbar-right">**

**<li><a href="oferte.php" title="Pagina oferte">Catolog Oferte</a></li>**

**<li><a href="cerere.php" title="Pagina cereri">Catolog Cereri</a></li>**

**<li><a href="servicii.php" title="Pagina Servicii">Servicii</a></li>**

**<li><a href="index.php#about" title="Informatie despre noi">Cine Suntem?</a></li>**

**<li><a href="index.php#offer">Ce oferim</a></li>**

**<li><a href="index.php#contact"><i class="fa fa-envelope-o" aria-hidden="true"></i></a></li>**

**</ul>**

**</div>**

**</div>**

**</div>**

**<div id="headerwrap">**

**<div class="container">**

**<div class="row centered">**

**<div class="col-lg-8 col-lg-offset-2">**

**<h1>Iti doresti un apartament al visurilor tale </h1>**

**<h2>Noi iti oferim cea mai buna optiune pentru tine!!!</h2>**

**</div>**

**</div>**

**</div>**

**</div>**

**Footer.php**

**<div id="f">**

**<div class="container">**

**<div class="row centered">**

**<a target="\_blank" href="https://www.facebook.com/alexandru.marjina"><i class="fa fa-facebook"></i></a>**

**<a href="#"><i class="fa fa-instagram"></i></a>**

**<a href="#"><i class="fa fa-vk"></i></a>**

**</div>**

**<center>**

**<a href="admin/index.php"><p>Toate drepturile aparate &copy; <?=date('Y')?> Marjina Alexandru</p></a>**

**</center>**

**</div>**

**</div>**

**</body>**

**</html>**

**Function.php**

**<?php**

**$mysqli=false;**

**function connectDB(){**

**global $mysqli;**

**$host = 'localhost'; // locatia-gazda serverului**

**$database = 'flatswap1'; // numele BD**

**$user = 'root'; // numele utilizatorului**

**$password = ''; // parola**

**$mysqli=new mysqli($host,$user,$password,$database);**

**if (!$mysqli) die ("Nu se poate conecta la MySQL");**

**return $mysqli;**

**}**

**function closeDB($mysqli){**

**global $mysqli;**

**$mysqli->close();**

**}**

**//Functii pentru administratorul**

**function isAdmin($login=false,$password=false){**

**if(!$login) $login=isset($\_SESSION["login"])? $\_SESSION["login"]:false;**

**if(!$password) $password=isset($\_SESSION["password"])? $\_SESSION["password"]:false;**

**return mb\_strtolower($login)===mb\_strtolower(ADM\_LOGIN) && $password===ADM\_PASSWORD;**

**}**

**function isSef($login=false,$password=false){**

**if(!$login) $login=isset($\_SESSION["login"])? $\_SESSION["login"]:false;**

**if(!$password) $password=isset($\_SESSION["password"])? $\_SESSION["password"]:false;**

**return mb\_strtolower($login)===mb\_strtolower(SEF\_LOGIN) && $password===SEF\_PASSWORD;**

**}**

**function login($login,$password){**

**$password=hashSecret($password);**

**if(isAdmin($login,$password)){**

**$\_SESSION["login"]=$login;**

**$\_SESSION["password"]=$password;**

**return true;**

**}**

**if(isSef($login,$password)){**

**$\_SESSION["login"]=$login;**

**$\_SESSION["password"]=$password;**

**return true;**

**}**

**return false;**

**}**

**function hashSecret($str){**

**return md5($str.SECRET);**

**}**

**function read($data){**

**if(is\_array($data)){**

**$escaped=array();**

**foreach ($data as $key => $value) {**

**$escaped[$key]=read($value);**

**}**

**return $escaped;**

**}**

**return trim(htmlspecialchars($data));**

**}**

**?>**

**Request.php**

**<?php**

**$request=read($\_REQUEST);**

**if (isset($request["auth"])){**

**if(login($request["login"],$request["password"])) {**

**header("Location: index.php");**

**}**

**else $message ="Nu corespunde numele si/sau parola";**

**}**

**?>**

**addContractC.php**

**<?php**

**require\_once 'blocks/top.php';**

**?>**

**<center><h4>Introduceti Datele pentru ContractC</h4></center>**

**</center>**

**<div class="container-ContractV">**

**<center><h3>Introduceti datele pentru Contract C</h3></center>**

**<form class="AddContractV" action="addContractC.php" method="post">**

**<div class="Vinzator">**

**<center><h4>Introduceti datele despre cumparator</h4></center>**

**<div class="spat">**

**<label for="nume">Numele</label></br>**

**<input type="text" class="form-control" id="nume" name="nume" placeholder="Nume">**

**</div>**

**<div class="spat">**

**<label for="prenume">Prenumele</label></br>**

**<input type="text" class="form-control" id="prenume" name="prenume" placeholder="Prenume">**

**</div>**

**<div class="spat">**

**<label for="nr">Telefon</label></br>**

**<input type="text" class="form-control" id="nr" name="nr" placeholder="Numar de Telefon">**

**</div>**

**<div class="spat">**

**<label for="email">Introdu Emailul</label></br>**

**<input type="email" class="form-control" id="email" name="email" placeholder="Email">**

**</div>**

**</div>**

**<div class="ApartamentV">**

**<center><h4>Introduceti datele despre Apartament dorit</h4></center>**

**<div class="spat">**

**<label for="Nr\_camere">Nr\_camere</label>**

**<input type="text" class="form-control" id="Nr\_camere" name="Nr\_camere" placeholder="Nr\_camere">**

**</div>**

**<div class="spat">**

**<label for="mob">Mobilat</label>**

**<select class="form-control" name="mob">**

**<option value="NULL">0</option>**

**<option value="Da">Da</option>**

**<option value="Nu">Nu</option>**

**</select>**

**</div>**

**<div class="spat">**

**<label for="incal">Incalzire</label>**

**<select class="form-control" name="incal">**

**<option value="NULL">0</option>**

**<option value="centralizata">centralizata</option>**

**<option value="autonoma">autonoma</option>**

**</select>**

**</div>**

**<div class="spat">**

**<label for="sector">Sector</label>**

**<select class="form-control" name="sector">**

**<option value="NULL">0</option>**

**<option value="Buiucani">Buiucani</option>**

**<option value="Botanica">Botanica</option>**

**<option value="Ciocana">Ciocana</option>**

**<option value="Centru">Centru</option>**

**<option value="Riscani">Riscani</option>**

**<option value="Telecentru">Telecentru</option>**

**<option value="Posta Veche">Posta Veche</option>**

**<option value="Sculeanca">Sculeanca</option>**

**</select>**

**</div>**

**</div>**

**<div id="Contract">**

**<center><h4>Introduceti datele Contractc</h4></center>**

**<div class="spat">**

**<label for="angaj">Angajatul</label>**

**<select class="form-control" name="angaj">**

**<option value="1">Marculescu</option>**

**<option value="2">Andreiescu</option>**

**<option value="3">Constantinescu</option>**

**<option value="4">Petrescu</option>**

**</select>**

**</div>**

**<div class="spat">**

**<label for="servV">Serviciu\_Cumparare</label>**

**<select class="form-control" name="servV">**

**<option value="5">ARENDA IMOBILULUI</option>**

**<option value="7">CUMPARAREA IMOBILULUI</option>**

**</select>**

**</div>**

**<div class="spat">**

**<label for="dataIn">Data intocmirii contractului</label></br>**

**<input type="date" class="form-control" id="dataIn" name="dataIn" placeholder="pretServiciu">**

**</div>**

**<div class="spat">**

**<label for="dataSF">Data sfirsit contract</label></br>**

**<input type="date" class="form-control" id="dataSF" name="dataSF" placeholder="dataSF">**

**</div>**

**<div class="spat">**

**<label for="pretSV">Pret Serviciu euro</label></br>**

**<input type="text" class="form-control" id="pretSV" name="pretSV" placeholder="pretServiciu">**

**</div>**

**<div class="spat">**

**<label for="realizat">Realizat</label>**

**<select class="form-control" name="realizat">**

**<option value="NULL">0</option>**

**<option value="0">Nu</option>**

**<option value="1">Da</option>**

**</select>**

**</div>**

**</div>**

**<input type="submit" id="AddC" name="AddC" value="Aduga" class="btn btn-primary ">**

**</form>**

**</div>**

**<?php**

**if(isset($\_POST['AddC'])){**

**if(($\_POST['nume'])=="" || ($\_POST['prenume'])=="" || ($\_POST['nr'])=="" || ($\_POST['email'])=="" || ($\_POST['Nr\_camere'])==""**

**||($\_POST['mob'])=="" || ($\_POST['incal'])=="" || ($\_POST['sector'])=="" ||($\_POST['angaj'])==""**

**|| ($\_POST['servV'])=="" || ($\_POST['dataIn'])=="" || ($\_POST['dataSF'])=="" || ($\_POST['pretSV'])=="" || ($\_POST['realizat'])=="")**

**{**

**echo "<script>alert('Nu au fost introduse toate datele')</script>";**

**}**

**else {**

**//adaugarea in tabelul vinzator**

**$nume=$\_POST['nume'];**

**$prenume=$\_POST['prenume'];**

**$nr=$\_POST['nr'];**

**$email=$\_POST['email'];**

**connectDB();**

**$sql = "INSERT INTO cumparator (cumparator.nume\_cumparator,cumparator.prenume\_cumparator,cumparator.telefon,cumparator.email)\n"**

**. "VALUES ('$nume','$prenume','$nr','$email')";**

**$result = mysqli\_query($mysqli,$sql);**

**if(!$result){**

**echo "<script>alert('Nu a fost introdus vinzator')</script>";**

**}**

**//adaugarea in tabelul apartament**

**$Nr\_camere=$\_POST['Nr\_camere'];**

**$mob=$\_POST['mob'];**

**$incal=$\_POST['incal'];**

**$sector=$\_POST['sector'];**

**$sql = "INSERT INTO apartament (apartament.nr\_camere,apartament.mobilat,apartament.incalzire,apartament.sector)\n"**

**. "VALUES ('$Nr\_camere','$mob','$incal','$sector')";**

**$result = mysqli\_query($mysqli,$sql);**

**if(!$result){**

**echo "<script>alert('Nu a fost introdus apartament')</script>";**

**}**

**//introduce in tabelul contracte**

**//intoarce ultimul id\_vinzator introdus**

**$sql = "SELECT cumparator.id\_cumparator FROM cumparator ORDER BY cumparator.id\_cumparator DESC LIMIT 1";**

**$result=mysqli\_query($mysqli,$sql);**

**$value=mysqli\_fetch\_object($result);**

**$id\_cumparator=$value->id\_cumparator;**

**$sql = "SELECT apartament.id\_apartament FROM apartament ORDER BY apartament.id\_apartament DESC LIMIT 1";**

**$result=mysqli\_query($mysqli,$sql);**

**$value=mysqli\_fetch\_object($result);**

**$id\_apartament=$value->id\_apartament;**

**$angaj=$\_POST['angaj'];**

**$servV=$\_POST['servV'];**

**$dataIn=$\_POST['dataIn'];**

**$dataSF=$\_POST['dataSF'];**

**$pretSV=$\_POST['pretSV'];**

**$realizat=$\_POST['realizat'];**

**$sql = "INSERT INTO contractc (contractc.id\_cumparator,contractc.id\_apartament,contractc.id\_angajat,contractc.id\_serv,contractc.data\_in,contractc.data\_sf,contractc.pret,contractc.realizat)\n"**

**. "VALUES ('$id\_cumparator','$id\_apartament','$angaj','$servV','$dataIn','$dataSF','$pretSV','$realizat')";**

**$result=mysqli\_query($mysqli,$sql);**

**echo "<script>alert('Success')</script>";**

**closeDB($mysqli);**

**exit;**

**}**

**}**

**?>**

**<?php**

**require\_once 'blocks/footer.php';**

**?>**

**addContractV.php**

**<?php**

**require\_once 'blocks/top.php';**

**?>**

**<center><h4>Introduceti Datele pentru ContractV</h4></center>**

**</center>**

**<div class="container-ContractV">**

**<center><h3>Introduceti datele pentru Contract V</h3></center>**

**<form class="AddContractV" action="addContractV.php" method="post">**

**<div class="Vinzator">**

**<center><h4>Introduceti datele despre vinzator</h4></center>**

**<div class="spat">**

**<label for="nume">Numele</label></br>**

**<input type="text" class="form-control" id="nume" name="nume" placeholder="Nume">**

**</div>**

**<div class="spat">**

**<label for="prenume">Prenumele</label></br>**

**<input type="text" class="form-control" id="prenume" name="prenume" placeholder="Prenume">**

**</div>**

**<div class="spat">**

**<label for="nr">Telefon</label></br>**

**<input type="text" class="form-control" id="nr" name="nr" placeholder="Numar de Telefon">**

**</div>**

**<div class="spat">**

**<label for="email">Introdu Emailul</label></br>**

**<input type="email" class="form-control" id="email" name="email" placeholder="Email">**

**</div>**

**</div>**

**<div class="ApartamentV">**

**<center><h4>Introduceti datele despre Apartament</h4></center>**

**<div class="spat">**

**<label for="Nr\_camere">Nr\_camere</label>**

**<input type="text" class="form-control" id="Nr\_camere" name="Nr\_camere" placeholder="Nr\_camere">**

**</div>**

**<div class="spat">**

**<label for="etajul">Etajul</label>**

**<input type="text" class="form-control" id="etajul" name="etajul" placeholder="etajul">**

**</div>**

**<div class="spat">**

**<label for="starea">Starea</label>**

**<input type="text" class="form-control" id="starea" name="starea" placeholder="starea">**

**</div>**

**<div class="spat">**

**<label for="suprafata">Suprafata</label>**

**<input type="text" class="form-control" id="suprafata" name="suprafata" placeholder="suprafata">**

**</div>**

**<div class="spat">**

**<label for="t\_etaj">Total Etaje in bloc</label>**

**<input type="text" class="form-control" id="t\_etaj" name="t\_etaj" placeholder="t\_etaj">**

**</div>**

**<div class="spat">**

**<label for="mob">Mobilat</label>**

**<select class="form-control" name="mob">**

**<option value="NULL">0</option>**

**<option value="Da">Da</option>**

**<option value="Nu">Nu</option>**

**</select>**

**</div>**

**<div class="spat">**

**<label for="incal">Incalzire</label>**

**<select class="form-control" name="incal">**

**<option value="NULL">0</option>**

**<option value="centralizata">centralizata</option>**

**<option value="autonoma">autonoma</option>**

**</select>**

**</div>**

**<div class="spat">**

**<label for="sector">Sector</label>**

**<select class="form-control" name="sector">**

**<option value="NULL">0</option>**

**<option value="Buiucani">Buiucani</option>**

**<option value="Botanica">Botanica</option>**

**<option value="Ciocana">Ciocana</option>**

**<option value="Centru">Centru</option>**

**<option value="Riscani">Riscani</option>**

**<option value="Telecentru">Telecentru</option>**

**<option value="Posta Veche">Posta Veche</option>**

**<option value="Sculeanca">Sculeanca</option>**

**</select>**

**</div>**

**<div class="spat">**

**<label for="pret">Pret Euro</label></br>**

**<input type="text" class="form-control" id="pret" name="pret" placeholder="pret">**

**</div>**

**<div class="spat">**

**<label for="title">Titlu</label></br>**

**<input type="text" class="form-control" id="title" name="title" placeholder="Titlu">**

**</div>**

**<div class="spat">**

**<label for="descriere">Descriere Apartament</label></br>**

**<textarea name="descriere" rows="6" cols="100"></textarea>**

**</div>**

**</div>**

**<div id="Contract">**

**<center><h4>Introduceti datele ContractV</h4></center>**

**<div class="spat">**

**<label for="angaj">Angajatul</label>**

**<select class="form-control" name="angaj">**

**<option value="1">Marculescu</option>**

**<option value="2">Andreiescu</option>**

**<option value="3">Constantinescu</option>**

**<option value="4">Petrescu</option>**

**</select>**

**</div>**

**<div class="spat">**

**<label for="servV">Serviciu\_Vinzare</label>**

**<select class="form-control" name="servV">**

**<option value="1">EVALUAREA IMOBILULUI</option>**

**<option value="2">ASISTENTA VANZARII APARTAMENTULUi</option>**

**<option value="3">ASISTENTA JURIDICA</option>**

**<option value="4">VANZAREA APARTAMENTELOR</option>**

**<option value="6">DAREA IN ARENDA A IMOBILULUI</option>**

**</select>**

**</div>**

**<div class="spat">**

**<label for="dataIn">Data intocmirii contractului</label></br>**

**<input type="date" class="form-control" id="dataIn" name="dataIn" placeholder="pretServiciu">**

**</div>**

**<div class="spat">**

**<label for="dataSF">Data sfirsit contract</label></br>**

**<input type="date" class="form-control" id="dataSF" name="dataSF" placeholder="dataSF">**

**</div>**

**<div class="spat">**

**<label for="pretSV">Pret Serviciu euro</label></br>**

**<input type="text" class="form-control" id="pretSV" name="pretSV" placeholder="pretServiciu">**

**</div>**

**<div class="spat">**

**<label for="realizat">Realizat</label>**

**<select class="form-control" name="realizat">**

**<option value="NULL">0</option>**

**<option value="0">Nu</option>**

**<option value="1">Da</option>**

**</select>**

**</div>**

**</div>**

**<input type="submit" id="AddV" name="AddV" value="Aduga" class="btn btn-primary ">**

**</form>**

**</div>**

**<?php**

**if(isset($\_POST['AddV'])){**

**if(($\_POST['nume'])=="" || ($\_POST['prenume'])=="" || ($\_POST['nr'])=="" || ($\_POST['email'])=="" || ($\_POST['Nr\_camere'])==""**

**|| ($\_POST['etajul'])=="" || ($\_POST['starea'])==""|| ($\_POST['suprafata'])=="" || ($\_POST['t\_etaj'])=="" || ($\_POST['mob'])=="" ||**

**($\_POST['incal'])=="" || ($\_POST['sector'])=="" || ($\_POST['pret'])==""|| ($\_POST['title'])=="" || ($\_POST['descriere'])=="" ||**

**($\_POST['angaj'])=="" || ($\_POST['servV'])=="" || ($\_POST['dataIn'])=="" || ($\_POST['dataSF'])=="" || ($\_POST['pretSV'])=="" || ($\_POST['realizat'])=="")**

**{**

**echo "<script>alert('Nu au fost introduse toate datele')</script>";**

**}**

**else {**

**//adaugarea in tabelul vinzator**

**$nume=$\_POST['nume'];**

**$prenume=$\_POST['prenume'];**

**$nr=$\_POST['nr'];**

**$email=$\_POST['email'];**

**connectDB();**

**$sql = "INSERT into vinzator (vinzator.nume\_vinzator,vinzator.prenume\_vinzator,vinzator.telefon,vinzator.email)\n"**

**. "VALUES ('$nume','$prenume','$nr','$email')";**

**$result = mysqli\_query($mysqli,$sql);**

**if(!$result){**

**echo "<script>alert('Nu a fost introdus vinzator')</script>";**

**}**

**//adaugarea in tabelul apartament**

**$Nr\_camere=$\_POST['Nr\_camere'];**

**$etajul=$\_POST['etajul'];**

**$starea=$\_POST['starea'];**

**$suprafata=$\_POST['suprafata'];**

**$t\_etaj=$\_POST['t\_etaj'];**

**$mob=$\_POST['mob'];**

**$incal=$\_POST['incal'];**

**$sector=$\_POST['sector'];**

**$pret=$\_POST['pret'];**

**$title=$\_POST['title'];**

**$descriere=$\_POST['descriere'];**

**$sql = "INSERT INTO apartament (apartament.nr\_camere,apartament.etajul,apartament.starea,apartament.suprafata,apartament.nr\_etaje,apartament.mobilat,apartament.incalzire,apartament.sector, apartament.title,apartament.text,apartament.pret)\n"**

**. "VALUES ('$Nr\_camere','$etajul','$starea','$suprafata','$t\_etaj','$mob','$incal','$sector','$title','$descriere','$pret')";**

**$result = mysqli\_query($mysqli,$sql);**

**if(!$result){**

**echo "<script>alert('Nu a fost introdus apartament')</script>";**

**}**

**//introduce in tabelul contracte**

**//intoarce ultimul id\_vinzator introdus**

**$sql = "SELECT vinzator.id\_vinzator FROM vinzator ORDER BY vinzator.id\_vinzator DESC LIMIT 1";**

**$result=mysqli\_query($mysqli,$sql);**

**$value=mysqli\_fetch\_object($result);**

**$id\_vinzator=$value->id\_vinzator;**

**$sql = "SELECT apartament.id\_apartament FROM apartament ORDER BY apartament.id\_apartament DESC LIMIT 1";**

**$result=mysqli\_query($mysqli,$sql);**

**$value=mysqli\_fetch\_object($result);**

**$id\_apartament=$value->id\_apartament;**

**$angaj=$\_POST['angaj'];**

**$servV=$\_POST['servV'];**

**$dataIn=$\_POST['dataIn'];**

**$dataSF=$\_POST['dataSF'];**

**$pretSV=$\_POST['pretSV'];**

**$realizat=$\_POST['realizat'];**

**$sql = "INSERT INTO contractv (contractv.id\_vinzator,contractv.id\_apartament,contractv.id\_angajat,contractv.id\_serv,contractv.data\_in,contractv.data\_sf,contractv.pret,contractv.realizat)\n"**

**. "VALUES ('$id\_vinzator','$id\_apartament','$angaj','$servV','$dataIn','$dataSF','$pretSV','$realizat')";**

**$result=mysqli\_query($mysqli,$sql);**

**echo "<script>alert('Success')</script>";**

**closeDB($mysqli);**

**exit;**

**}**

**}**

**?>**

**<?php**

**require\_once 'blocks/footer.php';**

**?>**

**anuntV.php**

**<center><h2>Anunturi Vinzare</h2></center>**

**<?php**

**connectDB();**

**$sql = "SELECT contractv.id\_contractv ,apartament.nr\_camere,apartament.etajul,apartament.starea,apartament.suprafata,apartament.nr\_etaje,apartament.mobilat,apartament.incalzire, apartament.sector, apartament.title,apartament.text,apartament.pret, angajat.nr\_telefon,contractv.id\_apartament**

**FROM contractv, apartament,angajat, servicii WHERE contractv.id\_apartament=apartament.id\_apartament**

**AND contractv.id\_serv=servicii.id\_serv AND contractv.id\_angajat=angajat.id\_angajat AND contractv.realizat='0' AND servicii.id\_serv ='4'";**

**$result = mysqli\_query($mysqli,$sql);**

**while($row=mysqli\_fetch\_array($result))**

**{**

**$id\_Contract=$row[0];**

**$nr\_camere=$row[1];**

**$etajul=$row[2];**

**$starea=$row[3];**

**$suprafata=$row[4];**

**$t\_etaj=$row[5];**

**$mobilat=$row[6];**

**$incalzire=$row[7];**

**$sector=$row[8];**

**$title=$row[9];**

**$descriere=$row[10];**

**$pret=$row[11];**

**$nr\_angajat=$row[12];**

**$id\_app=$row[13];**

**?>**

**<div class="container-anun">**

**<span><b>ID:<?=$id\_Contract?></b></span>**

**<span id="cont"><b>Nr\_contact:+373<?=$nr\_angajat?></b></span>**

**<h3><?=$title?></h3>**

**<div id="caract">**

**<h4>Caracteristici</h4>**

**<ul>**

**<li>Nr\_camere-<?=$nr\_camere?></li>**

**<li>Etajul-<?=$etajul?></li>**

**<li>Starea-<?=$starea?></li>**

**<li>Suprafata-<?=$suprafata?></li>**

**<li>Nr\_etaje in bloc-<?=$t\_etaj?></li>**

**<li>Mobilat-<?=$mobilat?></li>**

**<li>Incalzire-<?=$incalzire?></li>**

**<li>Sector-<?=$sector?></li>**

**</ul>**

**</div>**

**<div id="descriere">**

**<h4>Descriere</h4>**

**<p><?=$descriere?></p>**

**</div>**

**<div id="photo">**

**<img src="articole/<?=$id\_app?>.jpg" alt="">**

**</div>**

**<div id="pret">**

**<span>Pret:<?=$pret?><i class="fa fa-eur"></i></span>**

**</div>**

**</div>**

**<?php }**

**closeDB($mysqli);**

**?>**

**AngajReusit.php**

**<center>**

**<h3>Nr de Contracte vinzare realizate in intervalul [<?=$dataIN?>,<?=$dataSF?>] </h3>**

**<table class="out">**

**<tr>**

**<th>id\_Angajat</th>**

**<th>Nume</th>**

**<th>Prenume</th>**

**<th>Nr\_Sarcini\_realizate</th>**

**</tr>**

**<?php**

**connectDB();**

**$sql = "SELECT contractv.id\_angajat,angajat.nume,angajat.prenume,**

**COUNT(contractv.realizat) as realizate from contractv,angajat WHERE contractv.realizat='1' AND contractv.id\_angajat=angajat.id\_angajat**

**AND contractv.data\_sf BETWEEN '$dataIN' AND '$dataSF' GROUP by contractv.id\_angajat";**

**$result = mysqli\_query($mysqli,$sql);**

**while($row=mysqli\_fetch\_array($result))**

**{**

**$id\_Angajat=$row[0];**

**$Nume=$row[1];**

**$Prenume=$row[2];**

**$Sarcini\_realizate=$row[3];**

**?>**

**<tr>**

**<th><?php echo $id\_Angajat;?></th>**

**<th><?php echo $Nume;?></th>**

**<th><?php echo $Prenume;?> </th>**

**<th><?php echo $Sarcini\_realizate;?></th>**

**</tr>**

**<?php }**

**closeDB($mysqli);**

**?>**

**</table>**

**<h3>Nr de Contracte Cumparare realizate in intervalul [<?=$dataIN?>,<?=$dataSF?>] </h3>**

**<table class="out">**

**<tr>**

**<th>id\_Angajat</th>**

**<th>Nume</th>**

**<th>Prenume</th>**

**<th>Nr\_Sarcini\_realizate</th>**

**</tr>**

**<?php**

**connectDB();**

**$sql = "SELECT contractc.id\_angajat,angajat.nume,angajat.prenume, COUNT(contractc.realizat) as realizate from contractc,angajat WHERE contractc.realizat='1' AND contractc.id\_angajat=angajat.id\_angajat**

**AND contractc.data\_sf BETWEEN '$dataIN' AND '$dataSF' GROUP by contractc.id\_angajat";**

**$result = mysqli\_query($mysqli,$sql);**

**while($row=mysqli\_fetch\_array($result))**

**{**

**$id\_Angajat=$row[0];**

**$Nume=$row[1];**

**$Prenume=$row[2];**

**$Sarcini\_realizate=$row[3];**

**?>**

**<tr>**

**<th><?php echo $id\_Angajat;?></th>**

**<th><?php echo $Nume;?></th>**

**<th><?php echo $Prenume;?> </th>**

**<th><?php echo $Sarcini\_realizate;?></th>**

**</tr>**

**<?php }**

**closeDB($mysqli);**

**?>**

**</table>**

**<h3>Nr de contracte de Vinzare active in intervalul [<?=$dataIN?>,<?=$dataSF?>] </h3>**

**<table class="out">**

**<tr>**

**<th>id\_Angajat</th>**

**<th>Nume</th>**

**<th>Prenume</th>**

**<th>Nr\_Sarcini\_active</th>**

**</tr>**

**<?php**

**connectDB();**

**$sql = "SELECT contractv.id\_angajat,angajat.nume,angajat.prenume,**

**COUNT(contractv.realizat) as realizate from contractv,angajat WHERE contractv.realizat='0' AND contractv.id\_angajat=angajat.id\_angajat**

**AND contractv.data\_sf BETWEEN '$dataIN' AND '$dataSF' GROUP by contractv.id\_angajat";**

**$result = mysqli\_query($mysqli,$sql);**

**while($row=mysqli\_fetch\_array($result))**

**{**

**$id\_Angajat=$row[0];**

**$Nume=$row[1];**

**$Prenume=$row[2];**

**$Sarcini\_active=$row[3];**

**?>**

**<tr>**

**<th><?php echo $id\_Angajat;?></th>**

**<th><?php echo $Nume;?></th>**

**<th><?php echo $Prenume;?> </th>**

**<th><?php echo $Sarcini\_active;?></th>**

**</tr>**

**<?php }**

**closeDB($mysqli);**

**?>**

**</table>**

**<h3>Nr de contracte de Cumparare active in intervalul [<?=$dataIN?>,<?=$dataSF?>] </h3>**

**<table class="out">**

**<tr>**

**<th>id\_Angajat</th>**

**<th>Nume</th>**

**<th>Prenume</th>**

**<th>Nr\_Sarcini\_active</th>**

**</tr>**

**<?php**

**connectDB();**

**$sql = "SELECT contractc.id\_angajat,angajat.nume,angajat.prenume, COUNT(contractc.realizat) as realizate from contractc,angajat WHERE contractc.realizat='0' AND contractc.id\_angajat=angajat.id\_angajat**

**AND contractc.data\_sf BETWEEN '$dataIN' AND '$dataSF' GROUP by contractc.id\_angajat";**

**$result = mysqli\_query($mysqli,$sql);**

**while($row=mysqli\_fetch\_array($result))**

**{**

**$id\_Angajat=$row[0];**

**$Nume=$row[1];**

**$Prenume=$row[2];**

**$Sarcini\_active=$row[3];**

**?>**

**<tr>**

**<th><?php echo $id\_Angajat;?></th>**

**<th><?php echo $Nume;?></th>**

**<th><?php echo $Prenume;?> </th>**

**<th><?php echo $Sarcini\_active;?></th>**

**</tr>**

**<?php }**

**closeDB($mysqli);**

**?>**

**</table>**

**</center>**

**Concluzie:**

Aplicatia descrisa in acest raport are ca scop crearea unui SAPD (un sistem automatizat de prelucrare a datelor) intr-un serviciu de schimb de apartamente, acest sistem descrie lucru serviciului shimb de apartamente, astfel vin cumparatori si vinzatori care fac legatura unul cu altul prin intermediul serviciului acordat companie , vinzatori intocmesc un contract de vinzare expunind apartamentul pe platforma interprinderii iar cumparatori vin cu o cerere de cumparare si angajatii companiei fac legatura intre acestea, deasemenea compania acorda un spectru mare de servicii de la asistenta in vazare, asistenta juridica, gasirea arenda imobil , si dare in arenda a imobilului.

Deasemenea aplicatia permite un acces special administratorului de site care poate aduaga, modifica, sterge contracte. Deasemenea exista un acces ca conducator a interprinderii care poate vizualiza venitul companiei, reusita angajatilor, contractele care se termina la o data anumita , veniturile de la contracte intr-un interval de timp si cele mai solicitate servicii de catre utilizatori. Astfel aplicatia data reprezinta un sistem complex automatizat care poate servi o companie imobiliara ce se ocupa in domeniu.

**Bibliografie:**

* Conspectul la BD 2017 Conf. Univ. Perebinos Mihail;
* <https://www.tutorialspoint.com/web_development_tutorials.htm>
* <https://www.w3schools.com/>