

Gestiune centru inchirieri biciclete

1. Descrierea cerintelor:

Am decis sa realizez un site de inchirieri biciclete. Am ales ca tehnologie de realizare a acestei aplicatii ASP.NET MVC C# pentru partea de programarea si SQL SERVER Management Studio pentru partea de SQL Server.

2. Descrierea tabelelor din baza de date:

Tabelele folosite si campurile fiecarei tabele:

Centru (IdCentru, Nume, Nr_telefon, Program, IdAdresa)

Adresa (IdAdresa, Judet, Oras, Strada, Nr., Bloc)

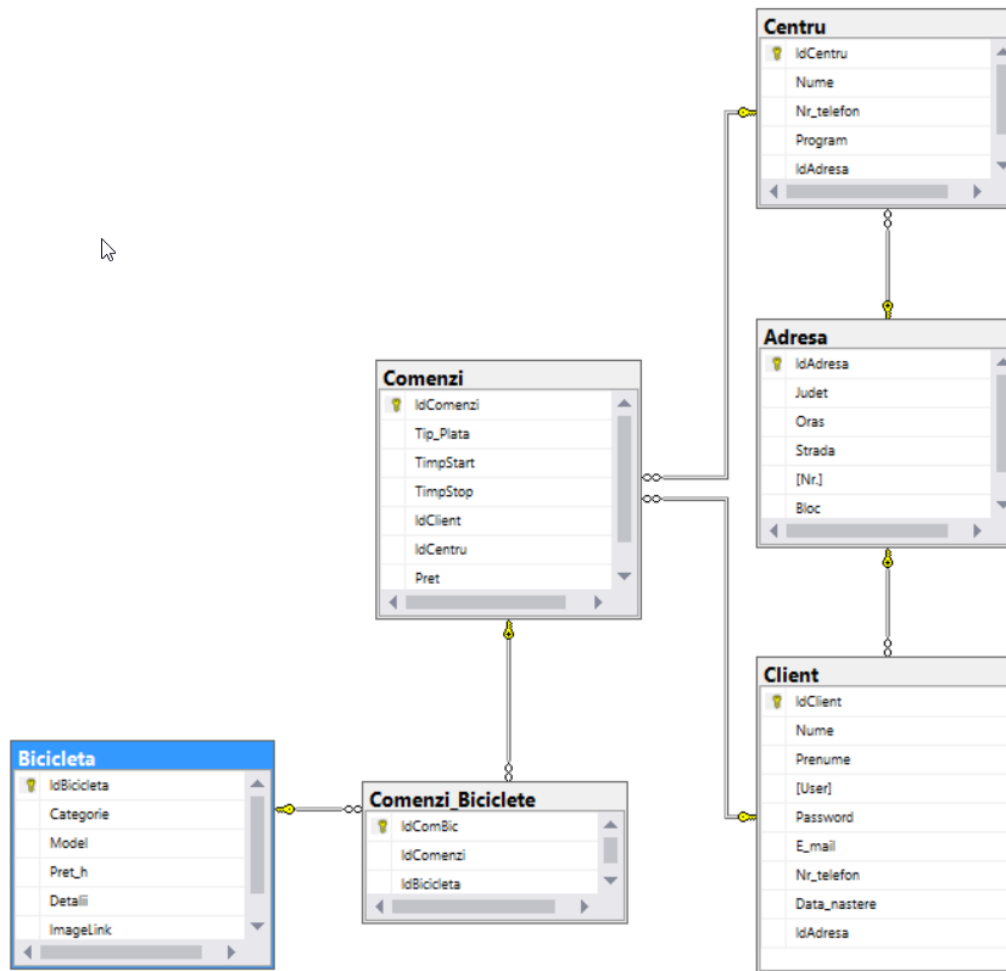
Client (IdClient, Nume, Prenume, User, Password, E_mail, Nr_telefon, Data_nastere, IdAdresa)

Comenzi (IdComenzi, Tip_Plata, TimpStart, TimpStop, IdClient, IdCentru, Pret)

Bicicleta (IdBicicleta, Categorie, Model, Pret_h, Detalii, ImageLink)

Comenzi_Biciclete (IdComBic, IdComenzi, IdBicicleta)

Relatiile dintre tabele:



Dupa cum se poate observa si in diagrama de mai sus, intre cele 6 tabele de mai sus avem doua relatii 1:N si doua relatii N:N

Prima relatie 1:N este intre tabelul Client si Adresa (fiecare client avand adresa sa)
Pe acelasi principiu a 2-a relatie 1:N este intre tabelul Centru si tabela Adresa (fiecare centru se afla la adresa sa)

Relatiile N:N se afla la tabela Comenzi, care uneste tabelele Client si Centru, iar a 2-a relatie N:N se regaseste la tabela Comenzi_Biciclete care uneste tabela de Biciclete cu Comenzi, dand astfel posibilitatea ca un client sa comande mai multe biciclete in aceeasi comanda.

3. Functionarea aplicatiei:

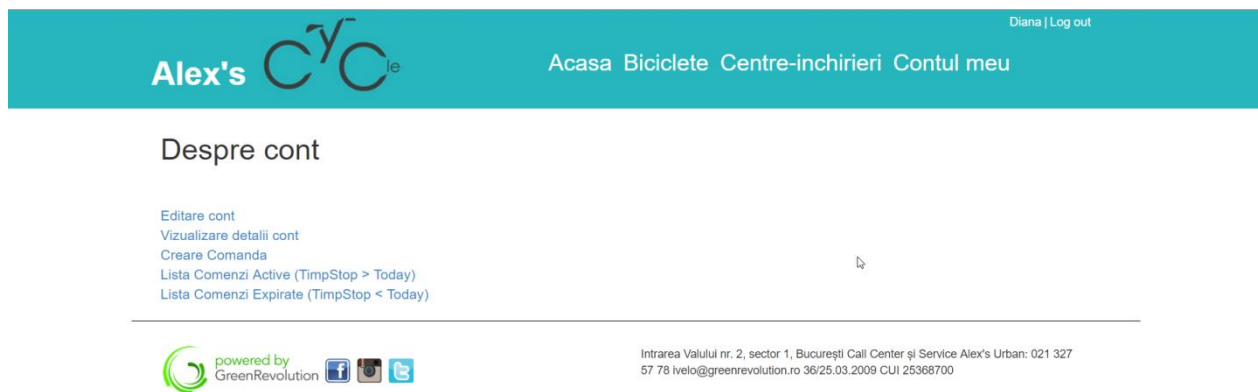
Realizat de: Stahie Alexandru

Aplicatia consta intr-un site web.

- a) Acesta are partea de client, ce ar trebui sa vada si ce ar avea nevoie un client care doreste sa inchiriezi o bicicleta sau mai multe.



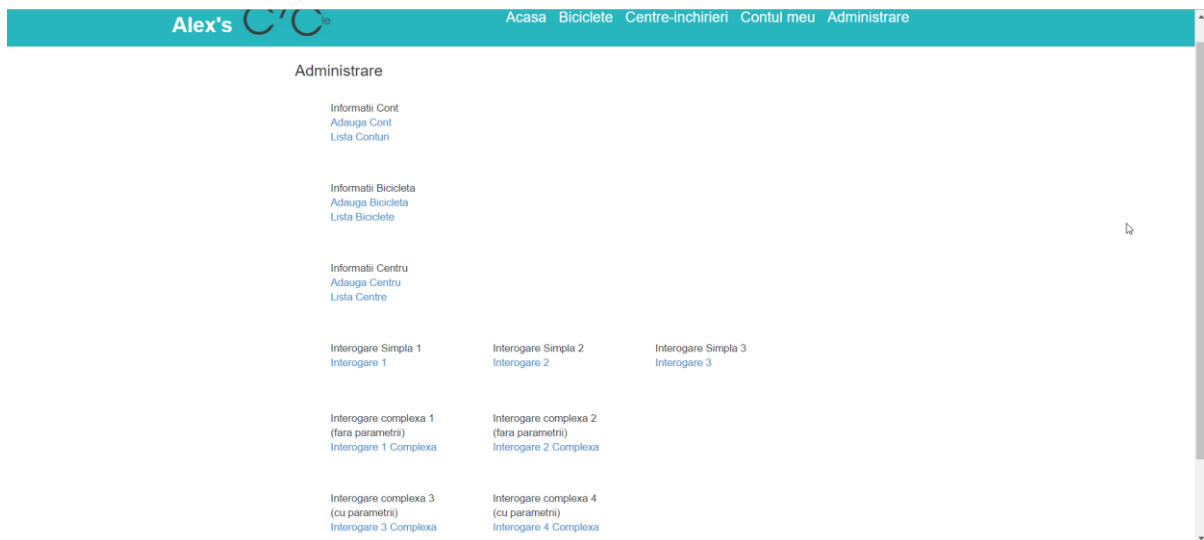
Aceasta parte de client ofera acces la pagina de Home (cea de mai sus), Vizualizarea bicicletelor disponibile (cu detaliile necesare si cu pretul pe ora afisate pentru fiecare bicicleta), centrele de inchirieri disponibile (la fel, fiecare cu detaliile necesare), si partea de log-in in pentru care clientul este nevoie sa isi face un account pentru a putea realiza o comanda.



Dupa crearea contului si logarea clientului, in pagina “Contul meu” vor aparea urmatoarele informatii, care iti permit sa editezi detalii contului sau doar sa le vizualizezi, de asemenea poti da comenzi sau poti vedea lista comenzilor active sau expirate din trecut.

b) Pe de alta parte aplicatia are si o parte de administrare, pe care are acces doar user-ul “admin”.

La logarea cu acest user va aparea automat un nou link in meniu numit “Administrare”, de unde poti modifica detaliile despre bicicletele disponibile, centre, chiar si useri. Insert, Update, Delete pe oricare din tabelele anterioare. De asemenea aici apar si link-urile catre paginile cu interogari simple si cele complexe.



4. Interogările simple folosite:

Primele 2 interogari simple sunt folosite pentru a aduce informatii din baza si a le afisa in campurile de update, pentru a stii campurile ce urmeaza a fi modificate. Operatia de update, dupa cum am explicat mai sus a fost realizata pentru tabelele, centre, client (si biciclete dar la aceasta tabela nu s-a folosit o interogare simpla)

- a) Update Centru (CentruController – HttpGet – EditareCentru)

```
“select * from Centru c  
inner join Adresa a on a.IdAdresa = c.IdAdresa  
where C.IdCentru = " + id
```

- b) Update Client (ClientController – HttpGet – EditareCont)

```
“select * from Client c  
inner join Adresa a on a.IdAdresa = c.IdAdresa  
where C.IdClient = @IdClient ”
```

- c) Urmatoarea interogarea simpla este folosita la afisarea detaliilor unei comenzi, aici se afiseaza informatiile despre comanda, centru si bicicletele comandate:

DetaliiComanda (ComenziController – HttpGet)

```
“select * from Comenzi c  
inner join Centru ce on c.IdCentru = ce.IdCentru  
where c.IdComenzi = " + id
```

Umatoarele 3 interogari simple se gasesc in SimpleQueryController:

- d) Clientii care au comenzi de peste 90 lei

```
select co.IdComenzi, c.Nume, c.Prenume, co.Pret from Client c  
inner join Comenzi co on co.IdClient = c.IdClient  
inner join Comenzi_Biciclete cb on cb.IdComenzi = co.IdComenzi  
where co.Pret > 90  
group by c.Nume, c.Prenume, co.Pret, co.IdComenzi
```

- e) Top 2 biciclete care apar cele mai des in comenzi

```
select top 2 cb.IdBicicleta, b.Model, b.Categorie, COUNT(cb.IdBicicleta) from Comenzi_Biciclete cb  
left join Bicicleta b on b.IdBicicleta = cb.IdBicicleta  
group by cb.IdBicicleta, b.Model, b.Categorie  
order by count(cb.IdBicicleta) desc
```

- f) Clientul cu cele mai multe comenzi

```
select top 1 COUNT(c.IdClient), cl.Nume, cl.Prenume, cl.IdClient from Comenzi c
inner join Client cl on cl.IdClient = c.IdClient
group by cl.IdClient, cl.Prenume, cl.Nume
order by COUNT(c.IdClient) desc
```

5. Interogari complexe folosite

- a) Interogari complexe fara parametrui:

- Bicicletele care nu au fost comandate niciodata

```
select b.IdBicicleta, b.Model, b.Categorie, b.Pret_h
from Bicicleta b
where b.IdBicicleta not in (select cb.IdBicicleta from Comenzi_Biciclete cb)
```

- Clientii cu pretul comenzii > media tuturor comenzilor

```
select cl.Nume, cl.Prenume, c.IdComenzi, c.TimpStart, c.TimpStop, c.Pret
from Comenzi c
inner join Client cl on cl.IdClient = c.IdClient
where c.Pret > (select avg(Pret) from Comenzi )
```

- b) Interogari complexe cu parametrui:

- Top x client cu comenzile cele mai scumpe

```
select c1.Nume, c1.Prenume, cl1.Pret from Client c1
inner join Comenzi cl1 on c1.IdClient = cl1.IdClient
where " + x[0].count + @">(select count(*)
                        from Client c2
                        inner join Comenzi cl2 on cl2.IdClient = c2.IdClient
                        where cl1.Pret < cl2.Pret)
```

- Comenzile ce contin biciclete cu pretul pe ora > x

```
Select c.IdComenzi, c.TimpStart, c.TimpStop
from Comenzi c
where c.IdComenzi in (select cb.IdComenzi
                    from Comenzi_Biciclete cb
                    inner join Bicicleta b on b.IdBicicleta = cb.IdBicicleta
                    where b.Pret_h > " + x[0].pret + " group by cb.IdComenzi)
```