**COMintel**

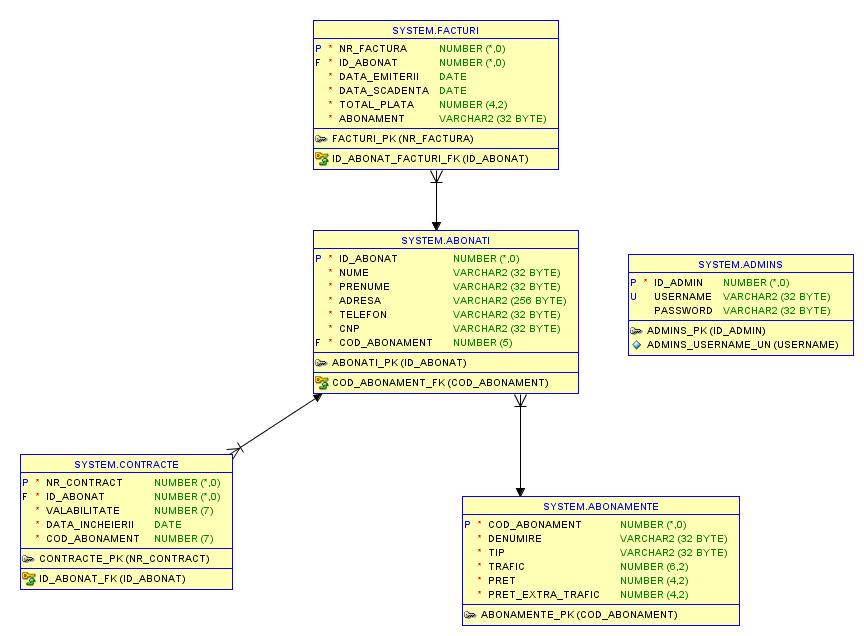
**Documentatie proiect**

**Nicoalescu Gheorghe-Mihaita**

**Grupa 233 Informatica**

**Descrierea bazei de date**

Baza de date este compusa din urmatoarele tabele: facturi, abonati, admins, contracte, abonamente.



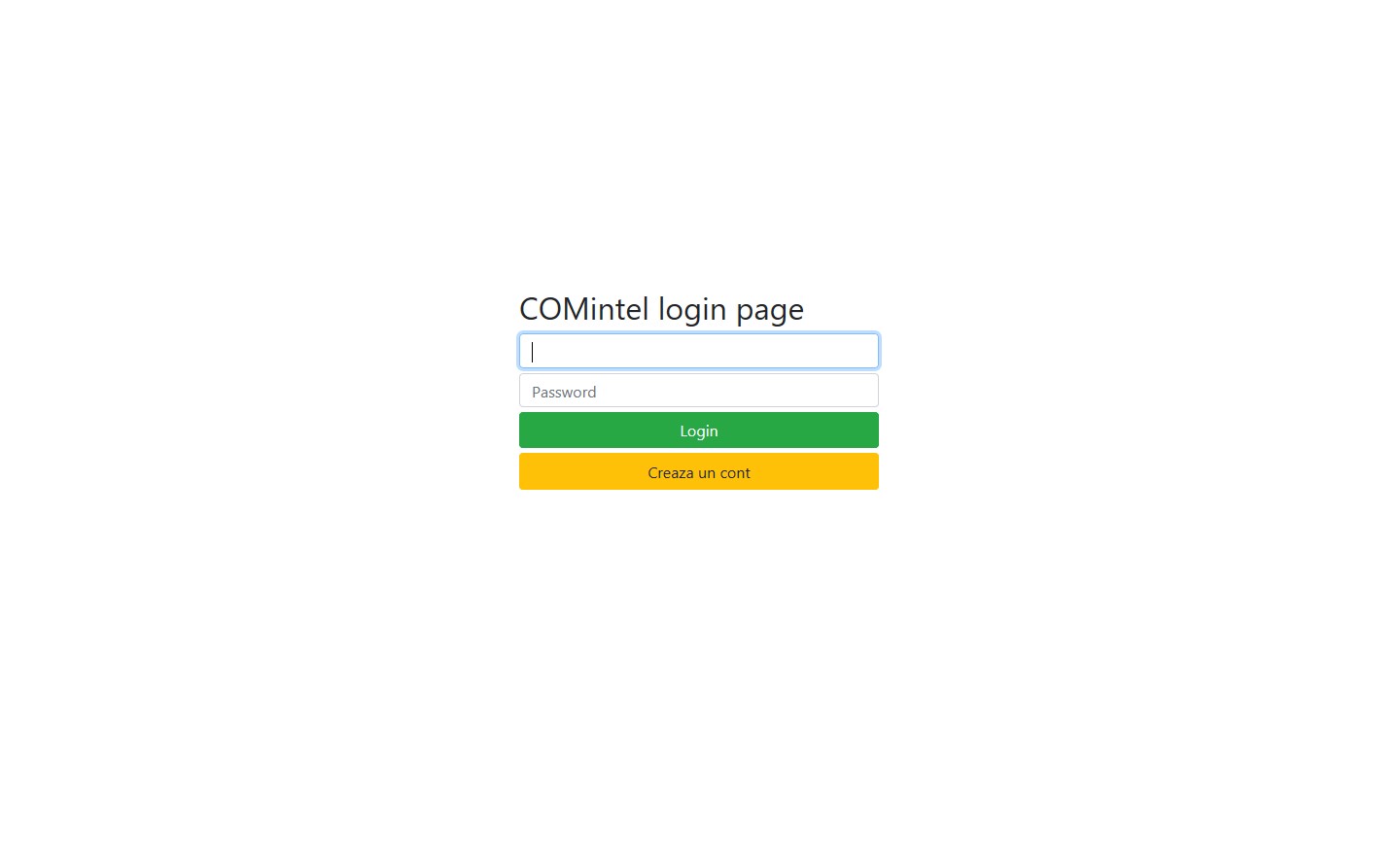
***Figura 1 – Diagrama bazei de date***

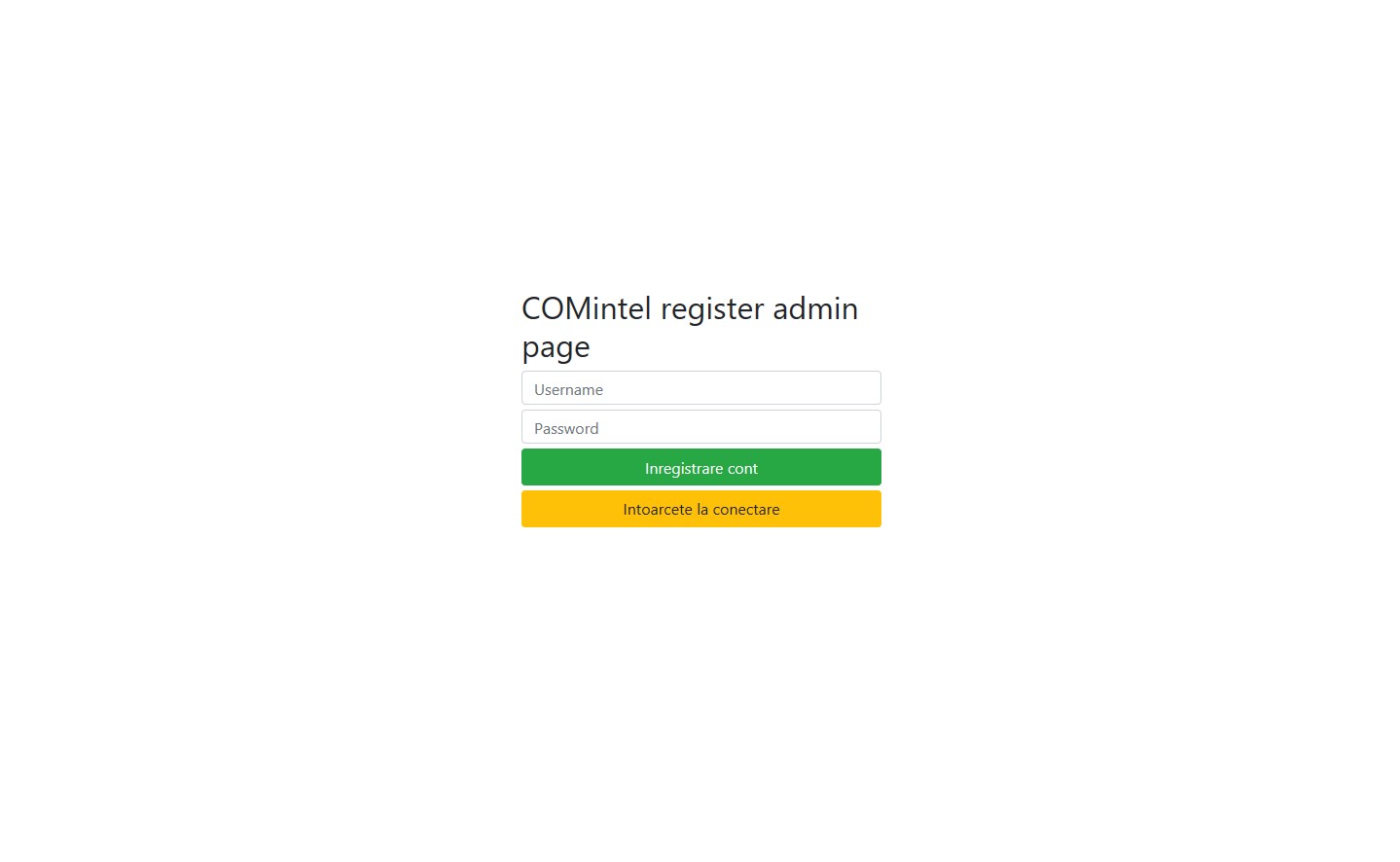
Aditional cerintelor am integrat o tabela pentru a avea o evidenta a utilizatorilor de pot accesa aplicatia. In aceasta tabela sunt prezente 3 campuri: id\_admin, username in care este retinut username-ul pentru conectare si ultimul camp password in care se retine parola folosita la conectare.

La tabela contracte am adaugat o coloanal reprezentand codul abonamentului pe care un abonat il detine, am considerat necesara aceasta adaugare deoarece pe perioada contractuala un utilizator poate schimba abonamentul pe care acesta in detine, schimarea abonamentului nu va infuenta afisarea facturilor.

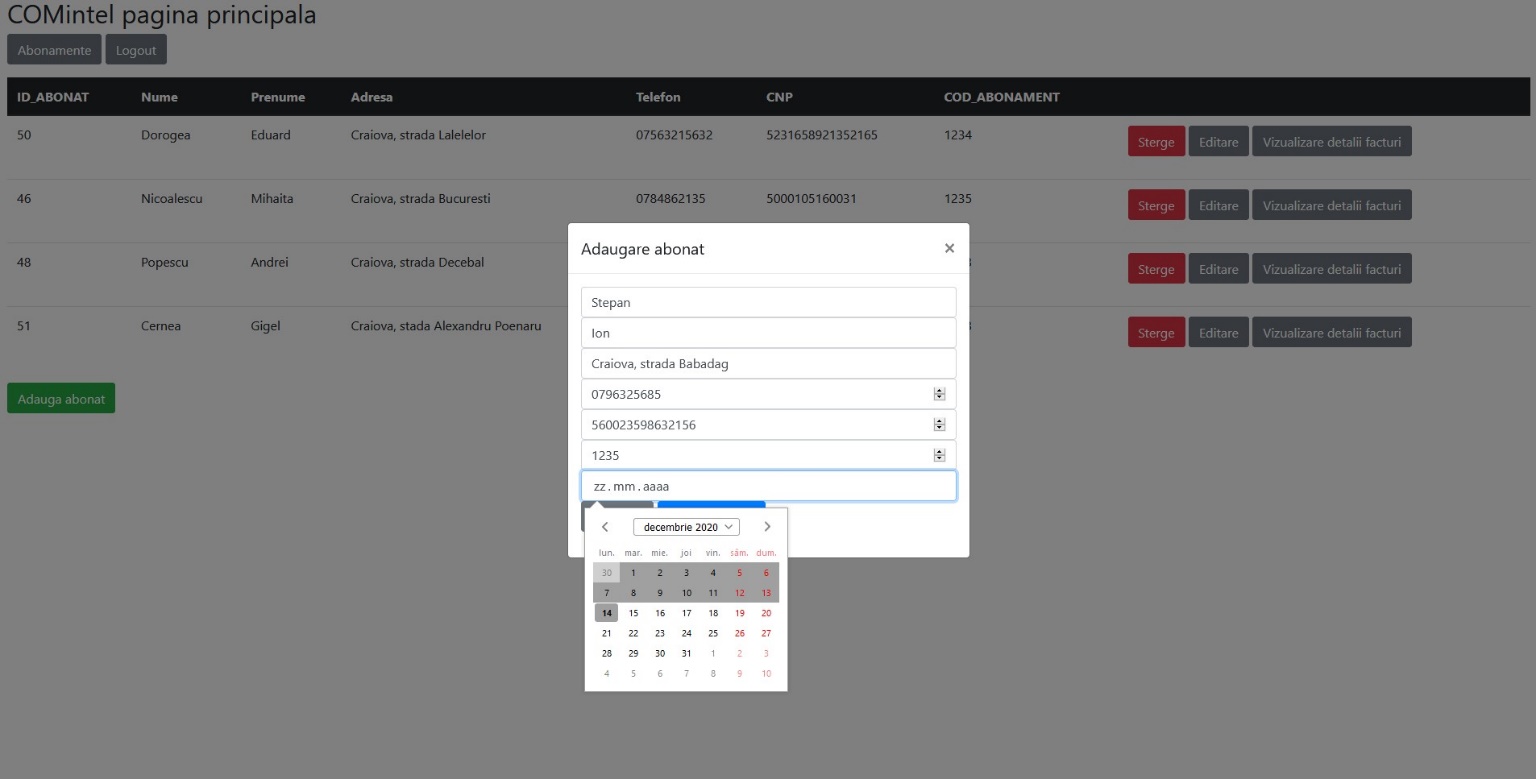
Pentru a micsora posibilitatea de eroare la introducerea datelor, pentru tabelele (Contracte, Abonati, Admins si Facturi) am creeat cate o secventa si un trigger care la inser sa incrementeze si sa introduca cheia aferenta tabelului automat, nefiind necesara interventia utilizatorului.

**Descrierea aplicatiei**

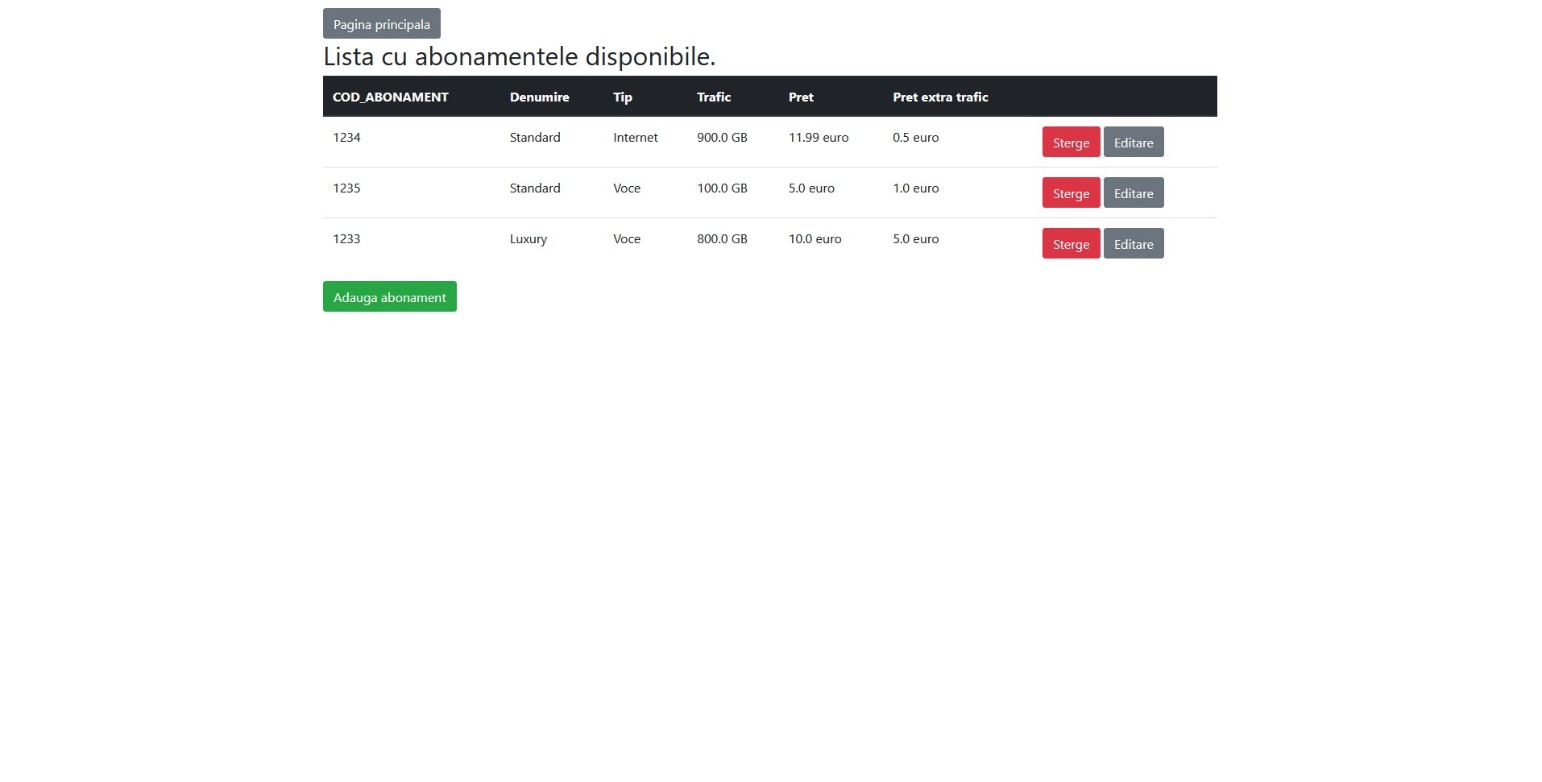
Aplicatia este implementata folosind Java Enterprise Edition aceasta fiind o aplicatie web. Pentru realizarea aplicatiei au fost folositi Java Servlets. Pentru rularea aplicatiei este necesar serverul Tomcat cu verisunea minim 8.0.

Aceasta este o aplicatie dinamica care permite conectarea pe baza unui username si a unei parole.Daca in baza de date nu este nici un cont pentru conectare, se poate creea unul apasand butonul “Creeazun cont”, dupa apasare userul va fi redirectionat catre pagina de inregistrare a unui cont.

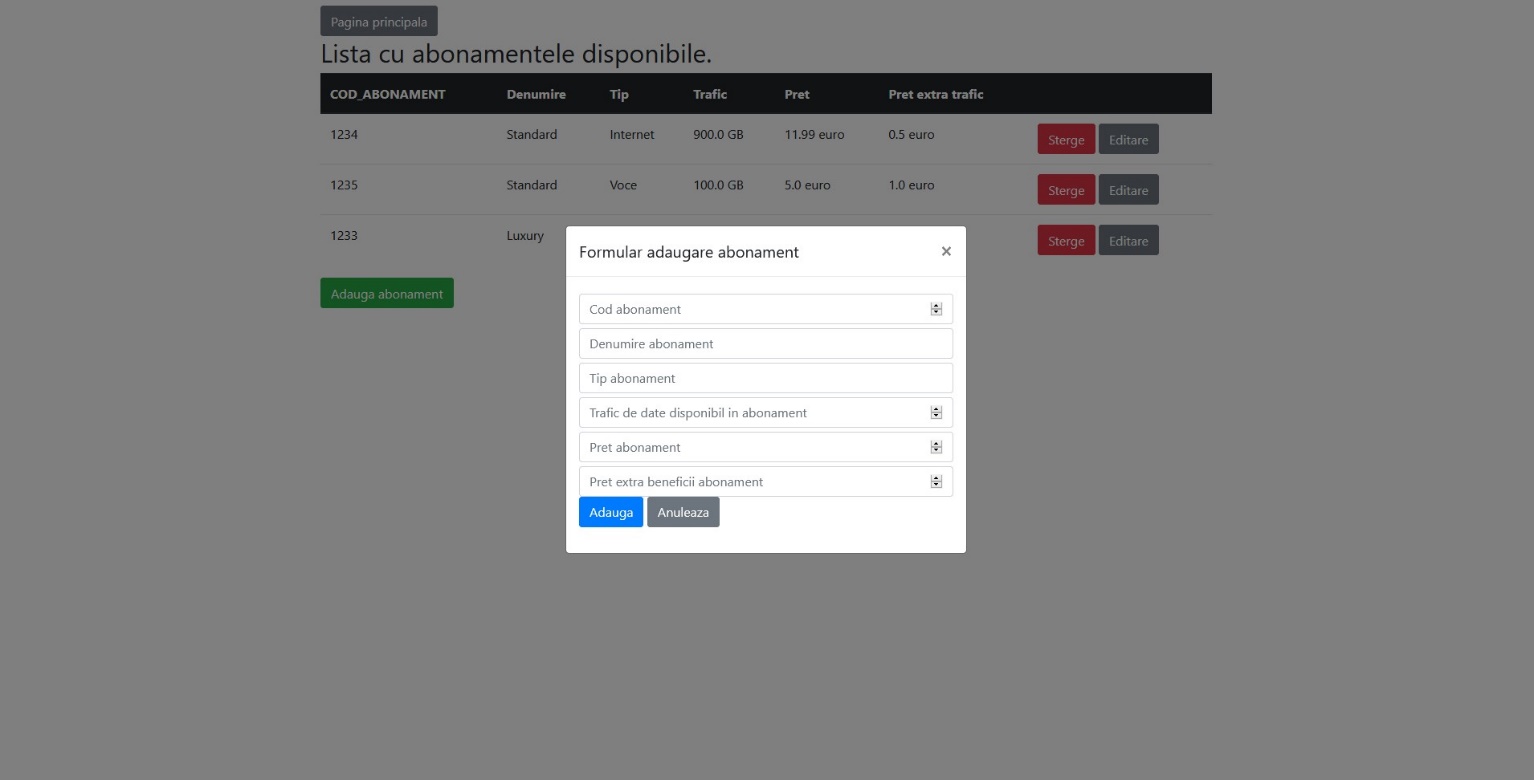
Dupa conectare utilizatorul poate adauga abonati in baza de date direct din aplicatie, cand se adauga un utilizator este necesara selectarea date pana cand este valabil abonamentul fiind valabil incepand cu data in care utilizatorul este adaugat in baza de date. Nu se poate selecta o data mai veche decat cea din prezent, acest lucru putand fi observant si in imaginea de mai jos. Zilele din calendar ce sunt colorate cu gri nu se mai pot selecta. Tot odata in aceasta pagina este posibila editarea unui utilizator sau stergerea acestuia. Data se va edita codul abonamentului pe care un abonat il detine atunci in mod automat prin intermediul unui trigger va fi modificat si contractual pe care abonatul il detine. La adaugarea in baza de date a unui abonat se va creea in mod automat si un contract care detine datele abonatului, durata acestuia si codul abonamentului pe care acesta in detine.



Pentru ca un abonat sa fie introdus in baza de date acesta trebuie sa aiba un cod de abonament valid, daca codul nu exista atunci utilizatorul este rugat sa recompleteze formularul introducand un cod valid de abonament.

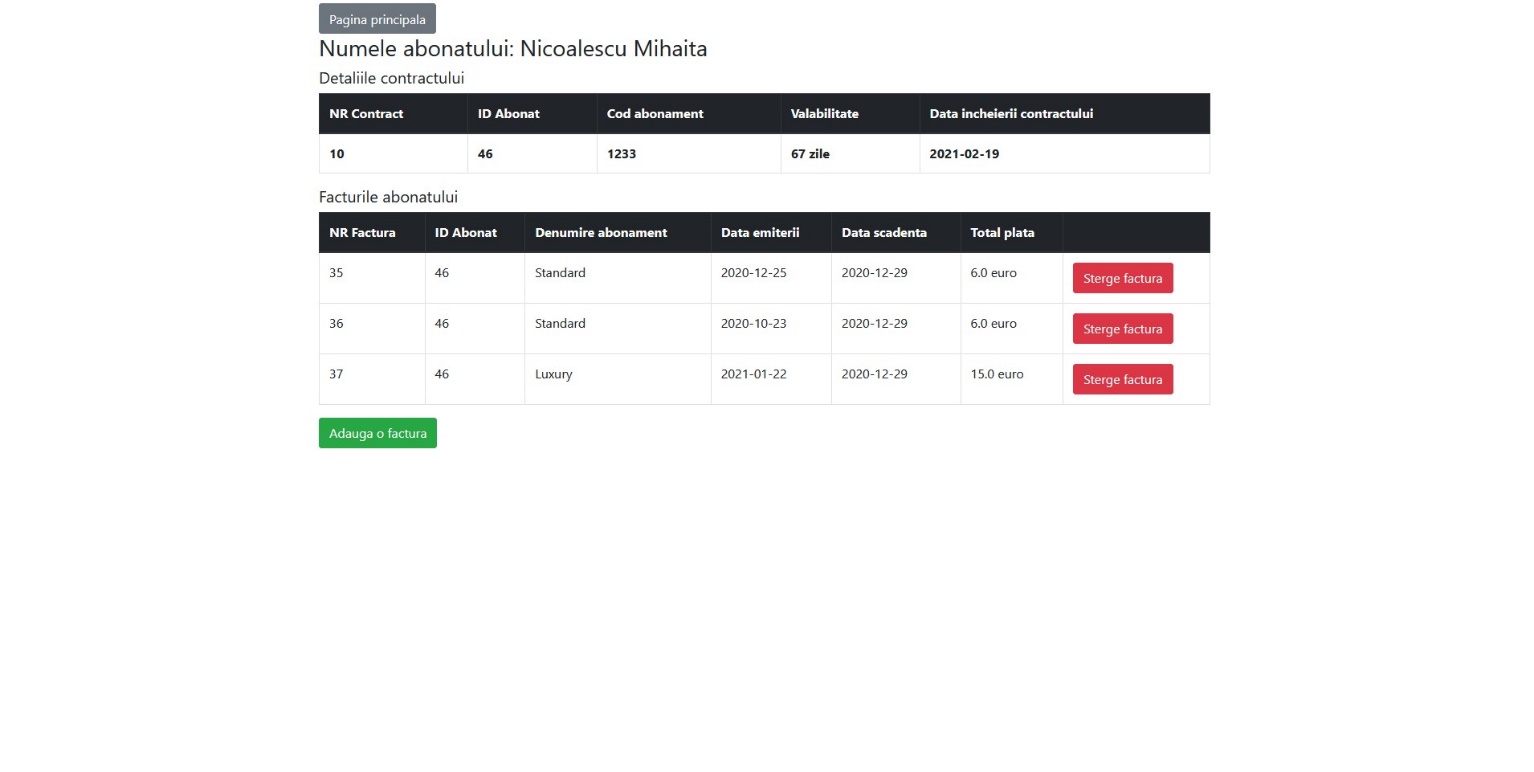
Un nou abonament poate fi introdus apanand butonul “Abonamente”, odata de utilizatorul apa acest buton va fi redirectionat catre pagina care permite adaugarea unui abonament , editarea unui abonament existent sau stergerea. Un abonament nu poate fi sters din baza de date daca exista cel putin un abonat care sa il detina.

Adaugarea unui abonament poate fi observata mai jos.



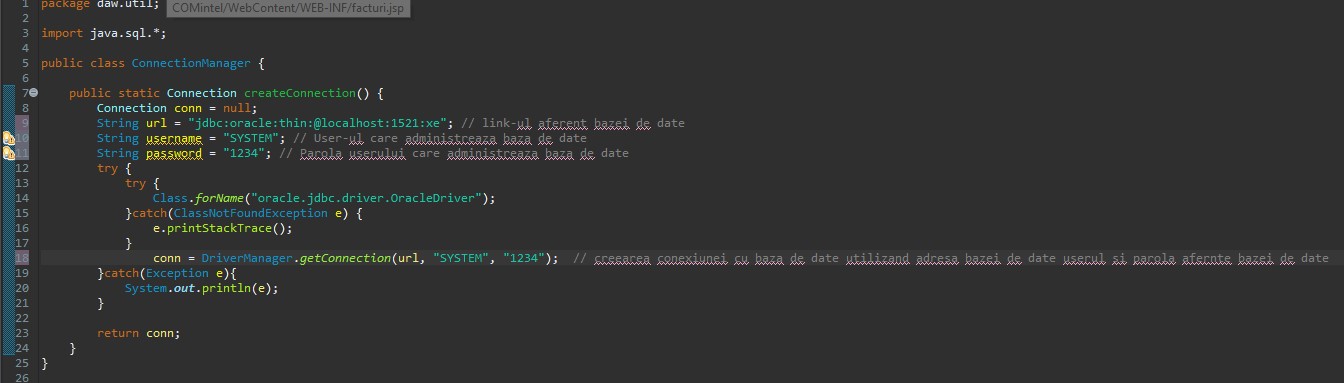
Userul poate naviga la pagina principala prin intermediul butonului “Pagina principala”. In continuare utilizatorul poate accesa facturile unui abonat prin intermediul butonului “Vizualizare detalii factura”.

Odata apasat acel buton utilizatorul va fi redirectionat catre pagina unde vor di afisare facturile pentru fiecare abonat individual.

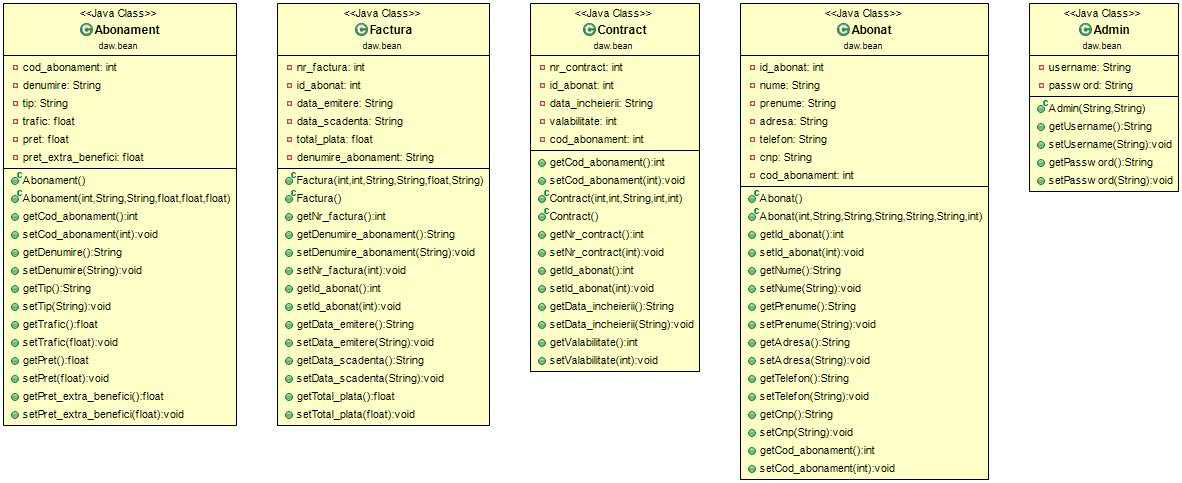


In aceasta pagina utilizatorul poate adauga noi facturi pentru abonat sau le poate sterge. La adaugarea unei facturi este necesara sa fie selectat doar data pentru emiterea facturii. Pretul total reprezinta pretul abonamentului plus pretul beneficiilor aferente abonamentului, data de scadenta este calculate ca fiind data de dupa 15 zile de la data emiterii facturii. In facture este inregistrat si numele abonamentului pentru a se identifica abonamentul pentru facture respective deoarece un abonat isi poate schimba abonamentul pe perioada contractuala. Tot in aceasta pagina in partea de sus sunt afisate datele contractului.

**Prezentarea modului in care se face conexiulea la baza de date**

 Conectarea la baza de date se realizeaza prin intermediul limbajului Java, in care am creeat o clasa separata care realizeaza conexiunea catre baza de date.

Clasele pe care le-am folosit la realizarea aplicatiei pot fi observate in figura de mai jos.

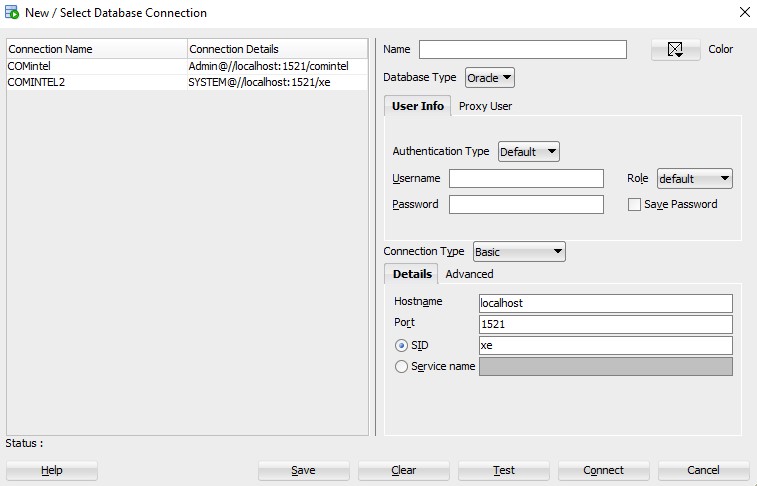


***Figura 2 – Diagrama de clase***

Pentru conectarea aplicatiei la o baza de date sunt necesare urmatoarele schimbari.

* In clasa *ConnectionManager* se vor modifica *username*, *parola* si *url* cu detaliile aferente bazei de date
* Partea din url ce trebuie modificata este : jdbc:oracle:thin:@localhost:1521:xe

Datele care sunt necesare pentru modificarea lincului pot fi regasite in SQL Developer la crearea unei noi conexiuni catre baza de date, cum se poate vedea si in figura de mai jos, localhost reprezentand HostName-ul, 1521 reprezinta portul pe care il foloseste baza de date iar xe reprezinta SID (identificatiorul bazei de date).



**Concluzii**

In concluzie acest proiect a fost o incercare pentru mine deoarece nu am mai lucrat cu baze de date Oracle in alte proiecte, iar acest lucru ma ajutat sa imi imbunatatesc gandirea si abilitatea de a lucra cu baze de date intr-o aplicatie. Asemanarea in folosirea unei baze de date Oracle in aplicatii este foarte mare cu PHP MySQL, existand o diferenta mica in modalitatea de conectare la baza de date, totodata am observant ca nu exista diferente de sintaxa intr-e o baza de date Oracle si o baza de date realizata pe PHP MySQL. Greutati nu pot spune ca am intampinat deoarece lucrand cu PHP MySQL in trecut mi-a oferit un avantaj acum, deoarece nu mi-a fost greu sa ma adaptez modului de lucru cu o baza de date Oracle.