Universitatea Ovidius din Constanta

Facultatea de Matematica- Informatica

Specializarea Informatica

Proiect SGBD

Sisteme de gestiune al bazelor de date

Munteanu Ioan

Nastase Marius

Nichifor Vlad

Contents

[1. Descrierea sistemului 3](#_Toc63619694)

[2. Dictionarul de date al SGBD 4](#_Toc63619695)

[1. Tabelele utilizatorului 4](#_Toc63619696)

[2. Coloanele tabelelor 4](#_Toc63619697)

[3. Constrangeri 4](#_Toc63619698)

[3. Librariile limbajului de programare ales 4](#_Toc63619699)

[4. Diagrama cazurilor de utilizare 4](#_Toc63619700)

[5. Diagrama de context Functionala 5](#_Toc63619701)

[6. Diagrama de context structurala 6](#_Toc63619702)

# 1. Descrierea sistemului

Sa se construiasca un sistem informatic care sa verifice daca bazele de date sunt proiectate corect, iar in cazul in care acestea nu sunt, sistemul sa poata rezolva aceste probleme.

Pentru utilizarea sistemului, utilizatorul trebuie sa se logheze in aplicatie. Datele introduse vor fi numele bazei de date si parola acesteia.

Utilizatorul poate verifica daca baza de date in care s-a logat este corect proiectata, putand sa verifice: daca BD este normalizat din punct de vedere al constrangerilor de existenta, daca BD este normalizat din punct de vedere al cheilor primare si daca BD este normalizat din punct de vedere al cheilor semantice.

**Normalizarea constrangerilor de existenta**

* un tabel trebuie sa aiba cel putin o coloana (un atribut), in afara de cheia primara, care sa NU admita valori null.
* implementare: se scaneaza fiecare tabel din baza de date, iar daca nu se gasesc coloane cu constrangerea de existenta, se cauta coloane in care nu sunt NULL-uri in acel moment. Daca nu exista nicio coloana de acest gen, trebuie informat utilizatorul ca acel tabel este gresit proiectat (se pot introduce linii numai cu null-uri) si rugat sa elimine null-uri macar dintr-o coloana. Altfel, se da utilizatorului lista de coloane pe care se poate pune constrangerea not null si rugat sa aleaga coloana la care sa se adauge in mod automat aceasta constrangere.

**Normalizare chei primare**

* Pentru fiecare tabela in care nu este cheie primara, se adauga una surogat.
* Pentru fiecare tabela in care
  + exista o cheie primara pe o singura coloana dar care nu este numerica
  + exista o cheie primara numerica pe mai mai multe atribute (cel putin 2)
* implementare: creeaza o cheie primara standard surogat cu autonumarare dupa ce o decade pe cea existenta din aceasta postura, ceea ce inseamna si stergerea cheilor straine care o refera si refacerea lor catre noua cheie surogat.

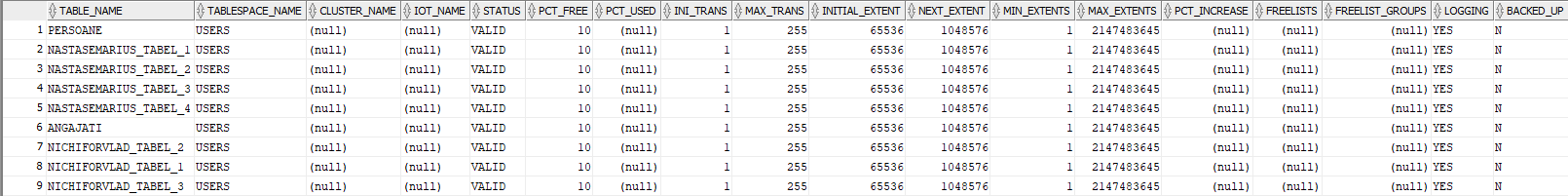
**Normalizare chei semantice**

* Se cauta mai intai tabele care nu au nici o cheie in afara de cea primara surogat.
* Apoi sa cauta pentru fiecare dintre coloane daca exista sau nu duplicate.
* Cand gaseste o coloana in care nu exista duplicat, o propune pentru a o transforma in cheie semantica (dialog cu utilizatorul)
* Daca nu gaseste nicio o coloana fara valori duplicate, informeaza utilizatorul ca tabelul este gresit proiectat si solicita introducerea/setarea unei chei semantice.

# 2. Dictionarul de date al SGBD

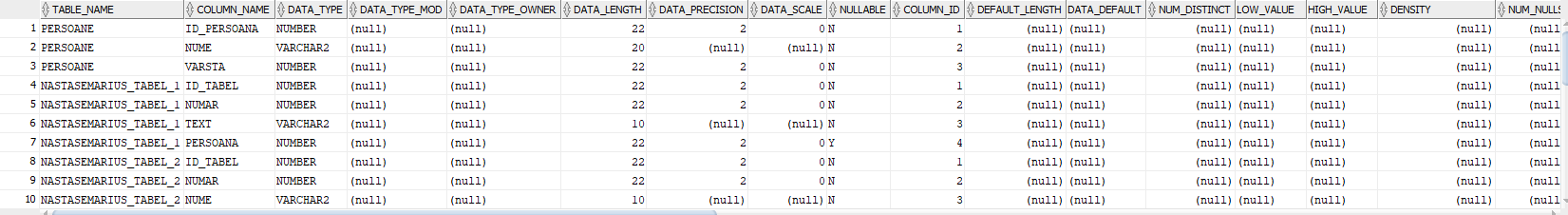
## 1. Tabelele utilizatorului

SELECT \* FROM user\_tables;



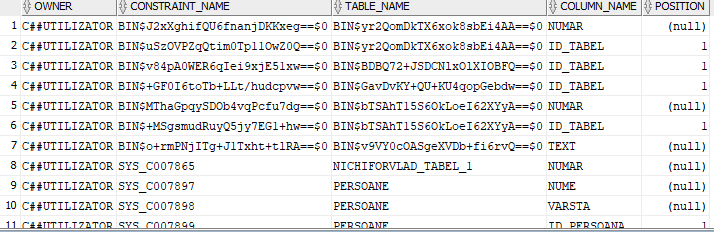
## 2. Coloanele tabelelor

SELECT \* FROM USER\_TAB\_COLUMNS;

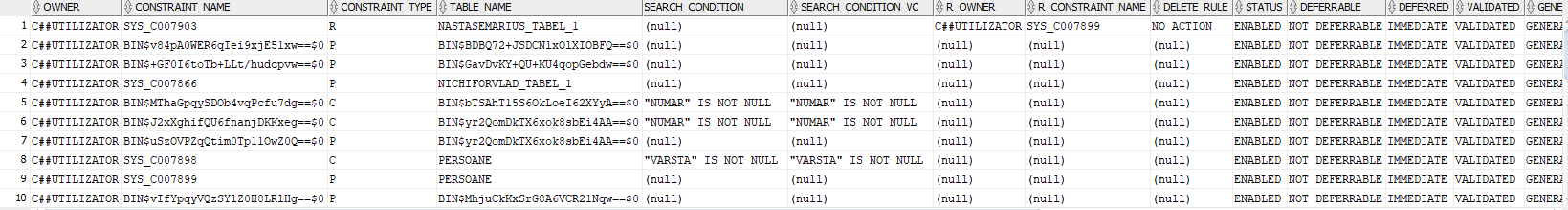


## 3. Constrangeri

SELECT \* FROM USER\_CONS\_COLUMNS;



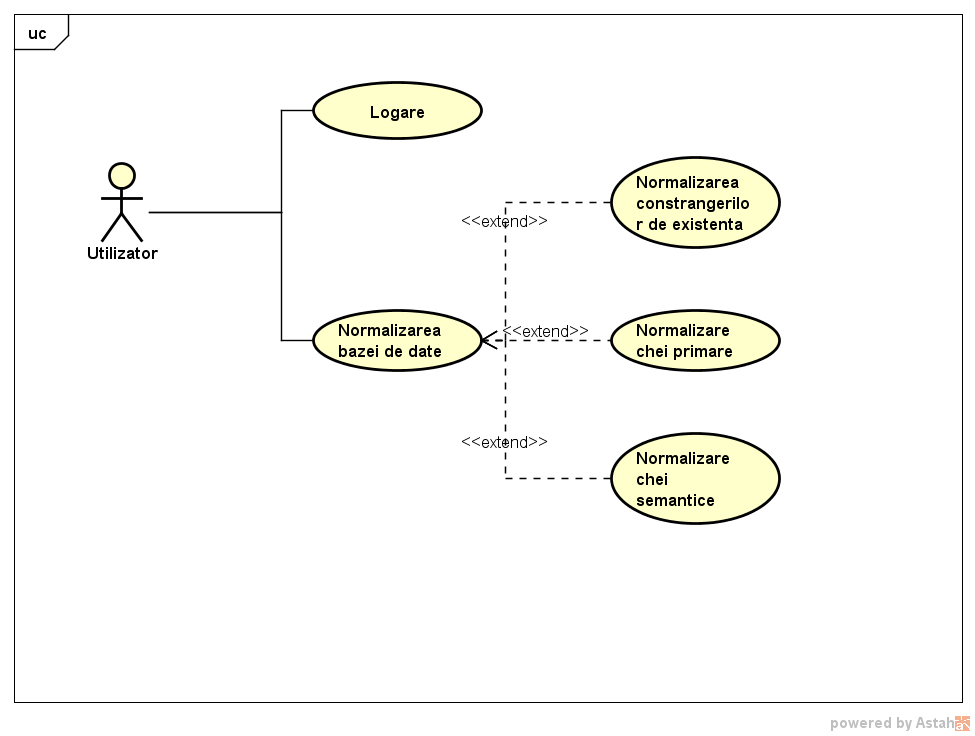
SELECT \* FROM USER\_CONSTRAINTS;



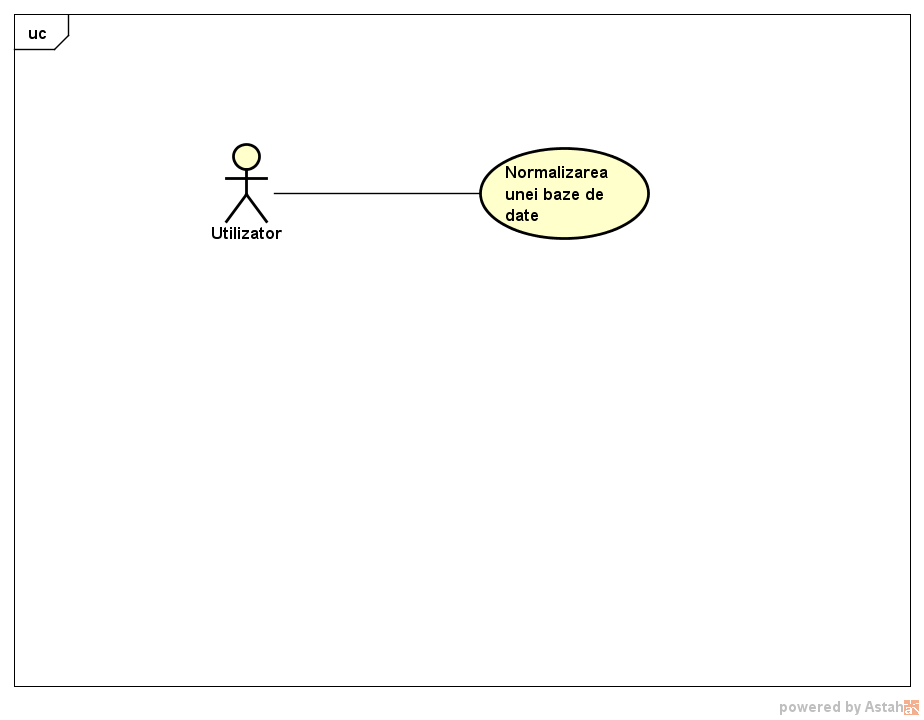
# 3. Librariile limbajului de programare ales

API JDBC, libraria ojdbc8.jar

# 4. Diagrama cazurilor de utilizare



# 5. Diagrama de context Functionala



# 6. Diagrama de context structurala

