**Bug tracking application**

**Nume Student : Pavelescu Victor**

**Cuprins**

1. Introducere 3
2. Prezentarea functionalitatilor 3
3. Etapele proiectarii aplicatiei: 3 - 17
   * 1. **Tehnologii folosite;** 3
     2. **Diagrama cazurilor de utilizare:** 4
     3. **Descrierea cazurilor de utilizare;** 5 - 10
     4. **Diagrama de clase;** 11
     5. **Diagrama de secventa si diagrama de colaborare/comunicare;** 12 - 15
     6. **Diagrama bazei de date;** 16
     7. **Diagrama de ferestre;** 17
4. Modul de functionare al aplicatiei; 17 - 22

**Aplicatie pentru administrarea sistemului**

**de gestiune a bug-urilor**

1. **Introducere:**

Aceasta aplicatie are scopul de a administra o baza de date ce contine bug-urile din cadrul unei firme. Utilizatorii din grupul Testers au posibilitatea de a inregistra bug-uri in baza de date. Utilizatorii din grupul Programmers au posibilitatea de a verifica lista de buguri si a modifica un bug selectat.

Aceasta aplicatie este de timp web. Aceasta a fos scrisa folosind limbajul python, frontend-ul a fost creeat cu Html, Css, Bootstrap, Jquery iar baza de date a fost construita cu sqlite3.

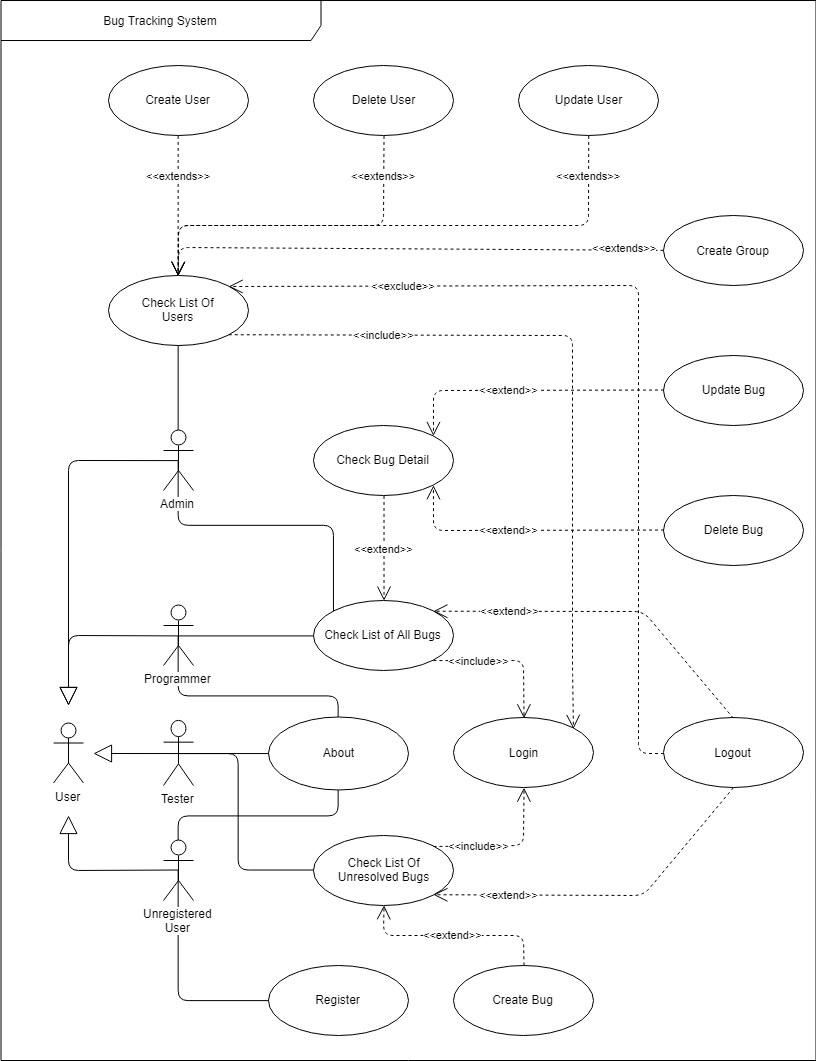
1. **Prezentarea functionalitatilor:**

**Aplicatia cuprinde realizarea urmatoarelor functionalitati:**

* + - 1. Posibilitatea de inregistrare si logare si delogare a utilizatorilor;
      2. Vizualizarea liste de bug-uri(lista cu toate bugurile/lista cu bugurile unresolved);
      3. Vizualizarea paginii about;
      4. Posibilitatea de inregistrare a unui bug nou in baza de date;
      5. Posibilitatea de modificare/sterge a unui bug selectat din baza de date;
      6. Posibilitatea de creeare/stergere/modificare a grupurilor/userilor/bugurilor de catre admin;

1. **Etapele proiectarii aplicatiei:**
2. **Tehnologii folosite:**
3. **Sqlite3:** A fost folosit pentru dezvoltarea bazei de date;
4. **Python:** Limbaj de date folosit pentru dezvoltare. Pycharm enviorement;
5. **Django:** Frameworkul web pe care a fost construita aplicatia ;
6. **ORM(Object-Relational Mapper):** Integrat in Django. A fost folosit pentru a
7. stoca, modifica si extrage date din modele in baza de date si vice versa;
8. **Html, Css, Bootstrap, JQuery:** A fost folosit la partea vizuala a aplicatiei web;
9. **Ajax:** Folosit pentru apeluri ajax;
10. **Diagrama cazurilor de utilizare:**

In continuare este prezentata diagrama cazurilor de utilizare care contine functionalitatile aplicatiei pentru programatorii si tester-ii din cadrul firmei.

****

**Fig 2.1 – Diagrama cazurilor de utilizare pentru aplicatie**

1. **Descrierea cazurilor de utilizare:**

Register, Login, Logout, Create Bug, Check list of all bugs, Update Bug, Delete Bug, Check Bug Detail, Check list of unresolved bugs, About, Check list of users, Create User, Delete User, Update User, Create Group

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Nume:** Register

**Participanti:** Unregistered User

**Flux evenimente:**

1. Unregistered User se conecteaza la sistem;

2. Sistemul accepta conexiunea;

3. Sistemul afiseaza formularul de inregistrare;

4. Unregistered User introduce datele de inregistrare;

5. Unregistered User apasa butonul de register;

6. Sistemul verifica daca datele sunt corecte;

7. Sistemul afiseaza un mesaj daca inregistrarea s-a efectuat cu success;

**Pre-conditii:** Unregistered User are conexiune la internet;

**Post-conditii:** Unregistered User este inregistrat in system;

**Cerinte de calitate:** Durata de steptare pentru inregistrare este mai mica de 1 secunda;

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Nume:** Login

**Participanti:** Tester/Programmer/Admin

**Flux evenimente:**

1. Tester/Programmer/Admin se conecteaza la sistem;

2. Sistemul accepta conexiunea;

3. Sistemul afiseaza formularul de logare;

4. Tester/Programmer/Admin introduce datele de inregistrare;

5. Tester/Programmer/Admin apasa butonul de login;

6. Sistemul verifica datele in baza de date;

7. Sistemul afiseaza pagina principala;

**Conditii de intrare:** Tester/Programmer/Admin are conexiune la internet;

**Conditii de iesire:** Tester/Programmer/Admin este logat in system;

**Cerinte de calitate:** Durata de asteptare pentru login este mai mica de 1 secunda

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Nume:** Logout

**Participanti:** Tester/Programmer/Admin

**Flux evenimente:**

1. Tester/Programmer/Admin activeaza functia de delogare

2. Sistemul afiseaza pagina principala

**Conditii de intrare:** Tester/Programmer/Admin este logat in system;

**Conditii de iesire:** Tester/Programmer/Admin este delogat;

**Cerinte de calitate**: Durata de asteptare pentru delogare este mai mica de 1 secunda;

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Nume:** Create Bug

**Participanti:**  Tester

**Flux evenimente:**

1. Tester activeaza functia de inregistrare bug;

2. Sistemul afiseaza un formular pentru inregistrare bug;

4. Tester introduce datele pentru inregistrare bug;

5. Sistemul actualizeaza lista de buguri;

6. Sistemul afiseaza un mesaj de confirmare;

**Conditii de intrare:** Tester este logat in sistem;

**Conditii de iesire:** Confirmarea inregistrarii bugului

**Cerinte de calitate:** Durata de asteptare pentru delogare este mai mica de 1 secunda;

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Nume:** Check list of all bugs

**Actori participanti:** Programmer

**Flux de evenimente:**

1. Programmer activeaza functia de vizualizare buguri

2. Sistemul afiseaza lista de buguri complete( solved/ unresolved );

**Conditii de intrare:** Programmer este logat in system;

**Conditii de iesire:** Programmer vede lista de buguri

**Cerinte de calitate:** Durata de asteptare pentru afisarea listei de buguri este mai mica de 1 secunda;

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Nume:** Update Bug

**Actori participanti:** Programmer

**Flux de evenimente:**

1. Programmer activeaza functia de update bug;

2. Sistemul afiseaza un formular pentru update bug;

3. Programmer introduce datele pentru update bug;

4. Programmer apasa butonul de register;

4. Sistemul actualizeaza bugul in lista de buguri;

5. Sistemul afiseaza pagina principala;

**Conditii de intrare:** Programmer este logat in sistem si a activat functia de afisare detalii bug;

**Conditii de iesire:** Programmer vede pagina principala **;**

**Cerinte de calitate:** Durata de asteptare pentru update bug este mai mica de 1 secunda;

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Nume:** Check Bug Detail

**Actori:** Programmer

**Flux de evenimente:**

1. Programmer activeaza functia de vizualizare detalii bug;

2. Sistemul afiseaza pagina de detalii bug;

**Conditii de intrare:** Programmer este inregistrar in sistem si a verificat lista de buguri;

**Codnitii de iesire:** Programmer are acces la functionalitatile bug detail;

**Cerinte de calitate:** Durata de afisare a paginii bug detail este mai mica de 1 secunda

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Nume:** Delete Bug

**Actori:** Programmer

**Flux de evenimente:**

1. Programmer activeaza functia de delete bug;

2. Sistemul afiseaza pagina de confirmare delete bug;

3. Programmer activeaza functia de confirmare delete bug;

4. Sistemul sterge bugul din lista de buguri;

5. Sistemul afiseaza pagina principala;

**Conditii de intrare:** Programmer a activat functia de vizualizare detalii bug;

**Codnitii de iesire:** Programmer are acces la pagina principala;

**Cerinte de calitate:** Durata de stergere a bugului din sistem este mai mica de 1 secunda

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Nume:** Check list of unresolved bugs

**Actori participanti:** Tester

**Flux de evenimente:**

1. Tester activeaza functia de vizualizare buguri;

2. Sistemul afiseaza lista de buguri unresolved ;

**Conditii de intrare:** Tester este logat in system;

**Conditii de iesire:** Tester vede lista de buguri;

**Cerinte de calitate:** Durata de asteptare pentru afisarea listei de buguri este mai mica de 1 secunda;

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Nume:** About

**Actori participanti:** Unregistered User/Tester/Programmer

**Flux de evenimente:**

1. Unregistered User/Tester/Programmer activeaza functia de vizualizare pagina about;

2. Sistemul afiseaza pagina de about;

**Conditii de intrare:** Unregistered User/Tester/Programmer are conexiune

**Conditii de iesire:** Unregistered User/Tester/Programmer are acces la pagina about;

**Cerinte de calitate:** Durata de asteptare pentru afisarea paginii about este mai mica de 1 secunda;

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Nume:** Check list of users

**Actori participanti:** Admin

**Flux de evenimente:**

1. Admin activeaza functia de vizualizare lista users;

2. Sistemul afiseaza pagina cu lista de users;

**Conditii de intrare:** Admin este logat in sistem

**Conditii de iesire:** Admin are acces la pagina cu lista de users;

**Cerinte de calitate:** Durata de asteptare pentru afisarea paginii users este mai mica de 1 secunda;

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Nume:** Create User

**Actori participanti:** Admin

**Flux de evenimente:**

1. Admin activeaza functia de create user;

2. Sistemul afiseaza pagina cu formularul pentru create user;

3. Admin introduce datele in formularul pentru create user;

4. Admin apasa butonul de save;

5. Sistemul actualizeaza lista de users;

6. Sistemul afiseaza pagina cu lista de users;

**Conditii de intrare:** Admin este logat si a verificat pagina cu users;

**Conditii de iesire:** Admin are acces la pagina cu lista de users;

**Cerinte de calitate:** Durata de asteptare pentru crearea userului este mai mica de 1 secunda;

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Nume:** Delete User

**Actori participanti:** Admin

**Flux de evenimente:**

1. Admin selecteaza userul din lista de user;

2. Sistemul afiseaza pagina cu informatiile legate de user;

2. Admin selecteaza functia de delete user;

4. Sistemul actualizeaza lista de users;

5. Sistemul afiseaza lista de users;

**Conditii de intrare:** Admin este logat si a verificat pagina cu users;

**Conditii de iesire:** Admin are acces la pagina cu lista de users;

**Cerinte de calitate:** Durata de asteptare pentru stergerea userului este mai mica de 1 secunda;

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Nume:** Update User

**Actori participanti:** Admin

**Flux de evenimente:**

1. Admin selecteaza userul

2. Sistemul afiseaza pagina cu informatiile legate de user;

3. Admin introduce datele pe care doreste sa le modifice in formular;

4. Admin apasa butonul de save;

5. Sistem actualizeaza infromatiile bugului in baza de date;

6. Sistem afiseaza pagina cu lista de buguri

**Conditii de intrare:** Admin este logat si a verificat pagina cu users;

**Conditii de iesire:** Admin are acces la pagina cu lista de users;

**Cerinte de calitate:** Durata de asteptare pentru modificarea userului este mai mica de 1 secunda;

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Nume:** Create Group;

**Actori participanti:** Admin;

**Flux de evenimente:**

1. Admin activeaza functia de add group;

2. Sistemul afiseaza pagina formul pentru add group;

3. Admin introduce datele in formular pentru add group;

4. Admin apasa butonul de save;

5. Sistem actualizeaza lista de grupuri;

6. Sistem afiseaza lista de grupuri;

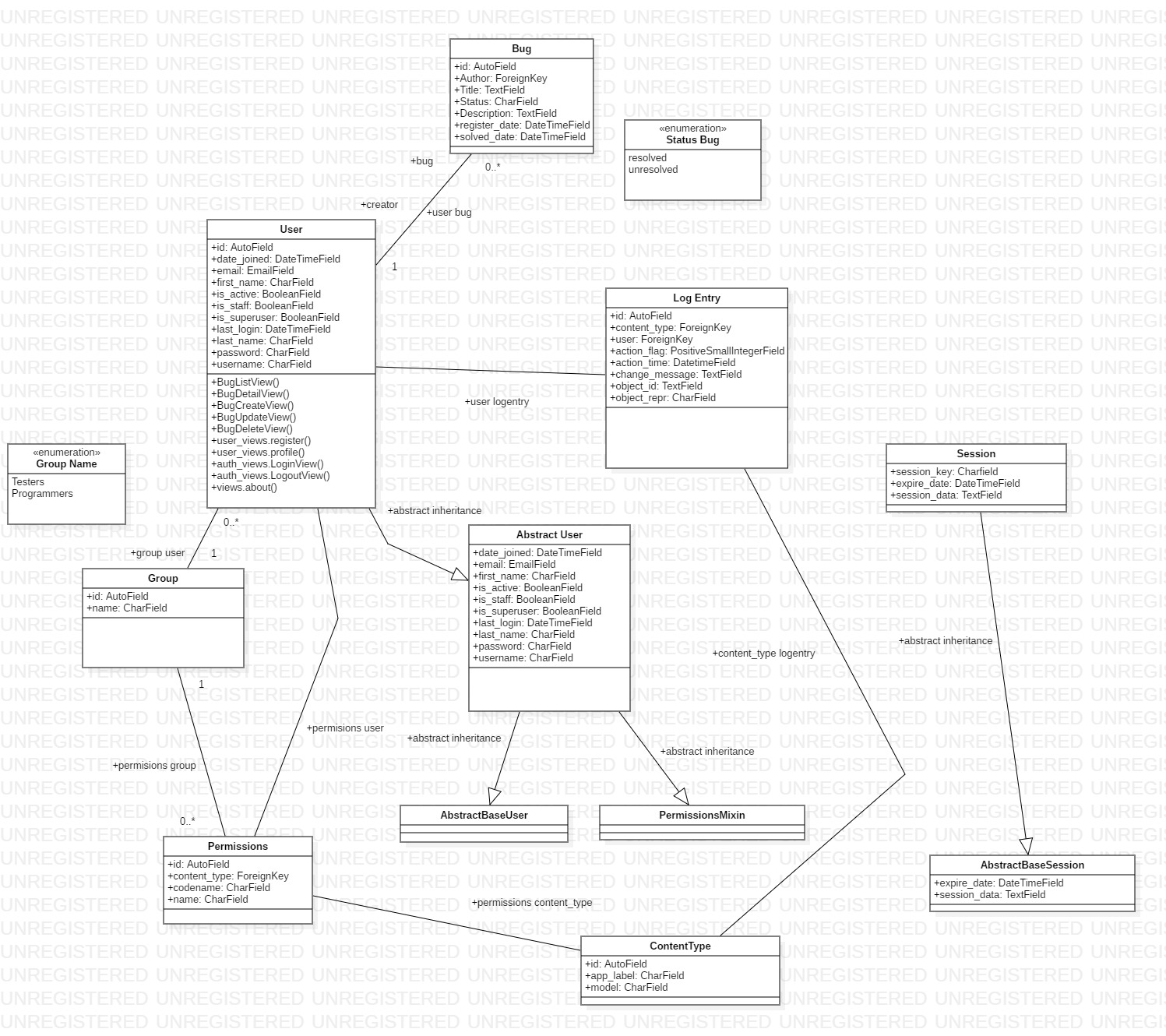
**Conditii de intrare:** Admin este logat;

**Conditii de iesire:** Admin are acces la pagina cu lista de grupuri;

**Cerinte de calitate:** Durata de asteptare pentru adaugarea grupului este mai mica de 1 secunda;

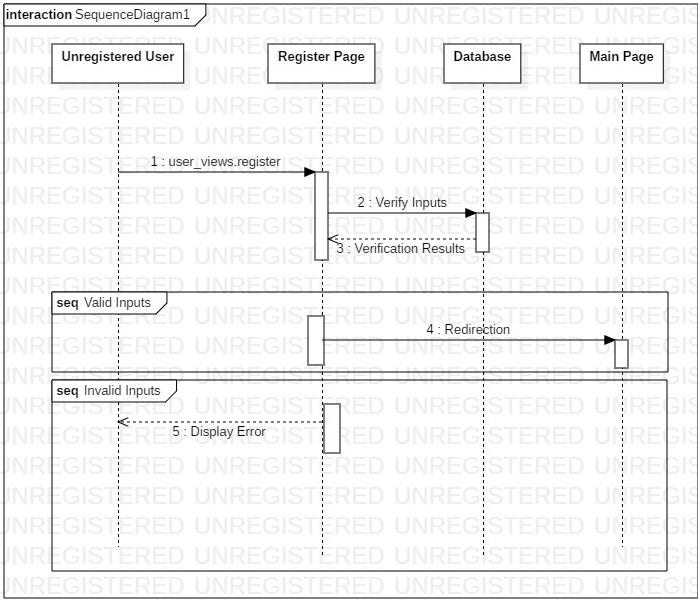
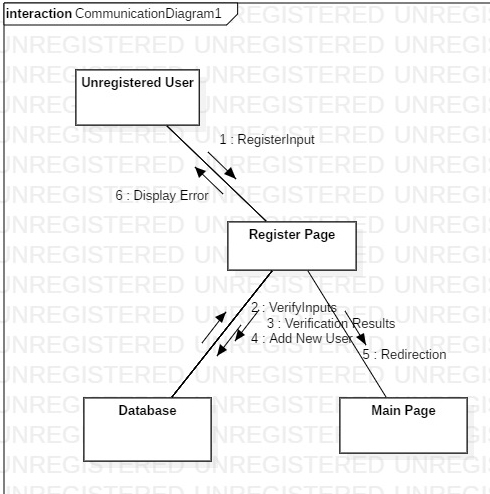
**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. **Diagrama de clase:**

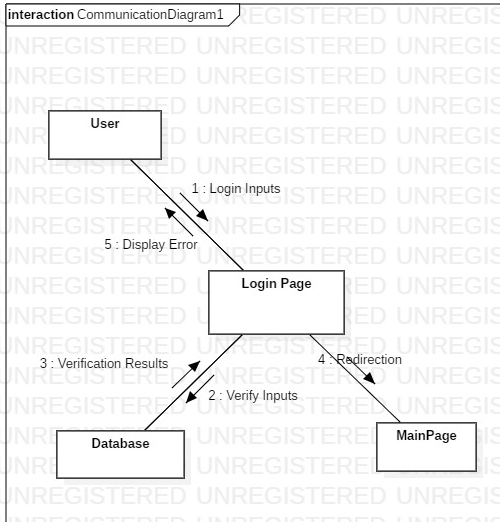
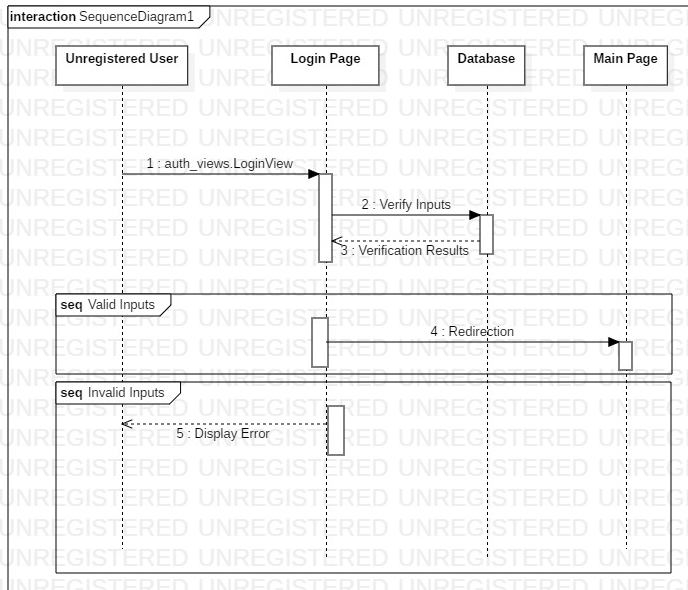
****

**Fig 2.2 – Diagrama de clase**

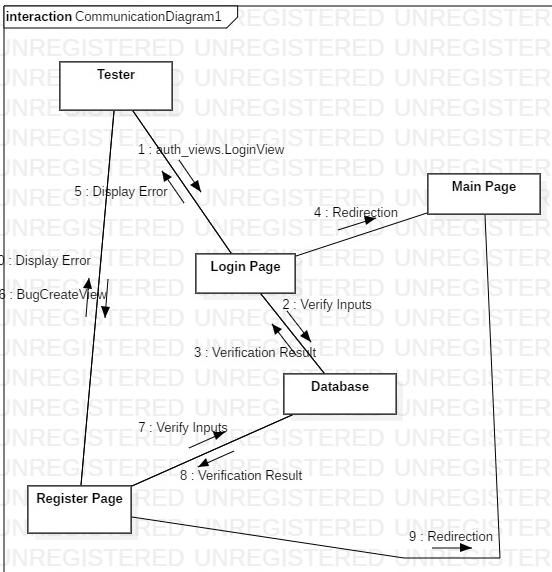
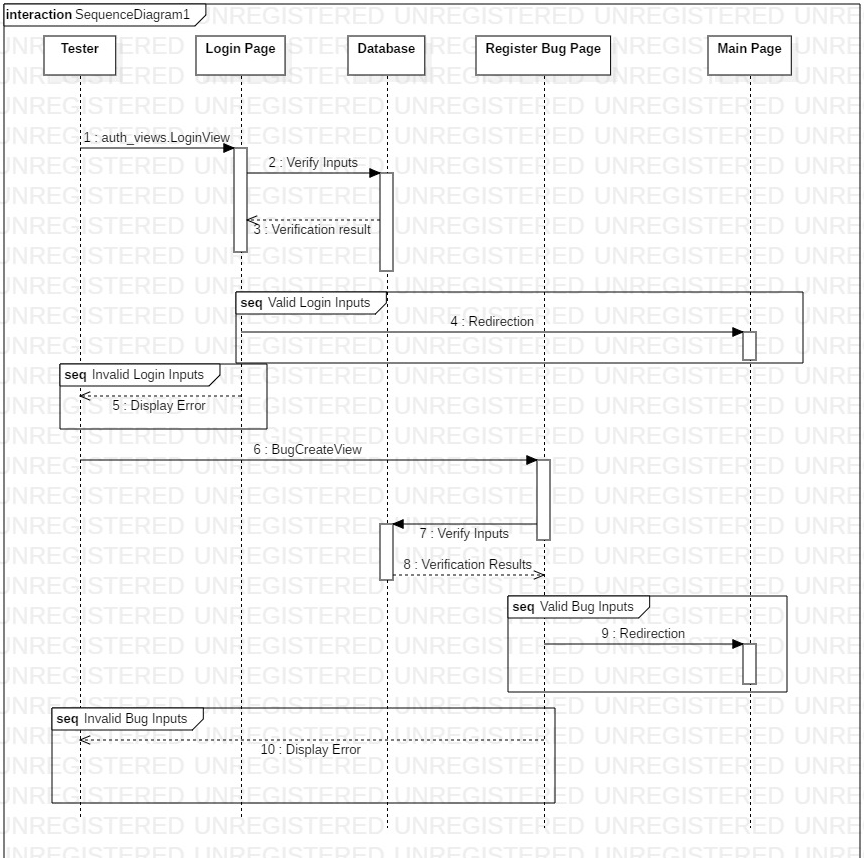
1. **Diagrama de secventa si colaborare:**
2. **Diagrama de secventa si colaborare pentru register user:**

****

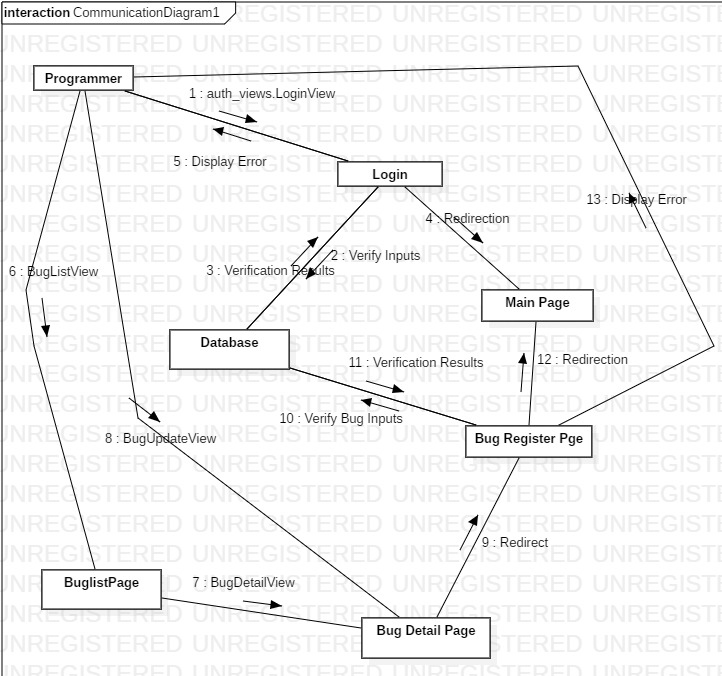
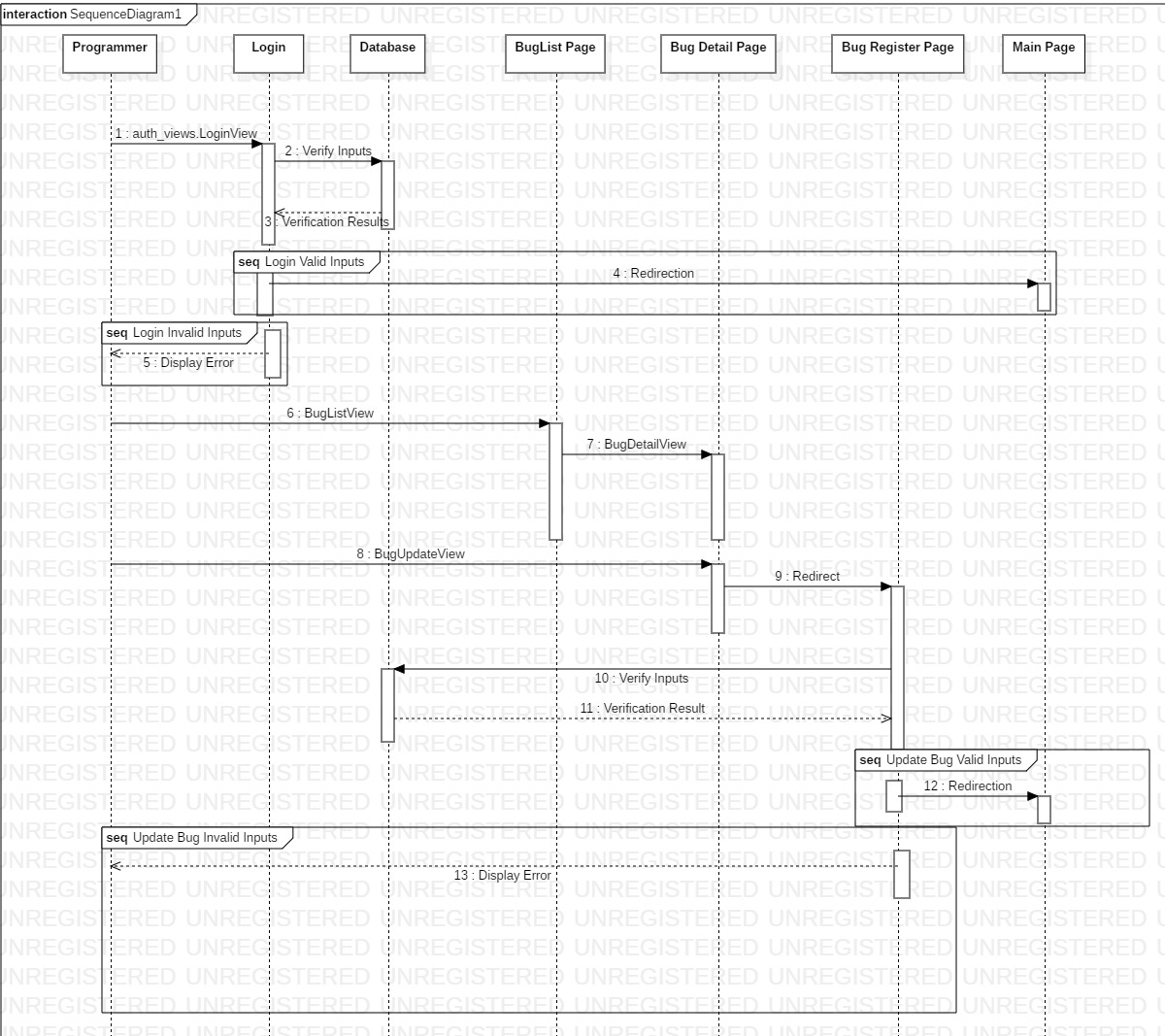
1. **Diagrama de secventa si colaborare pentru login user:**

****

1. **Diagrama de secventa si colaborare pentru register bug:**

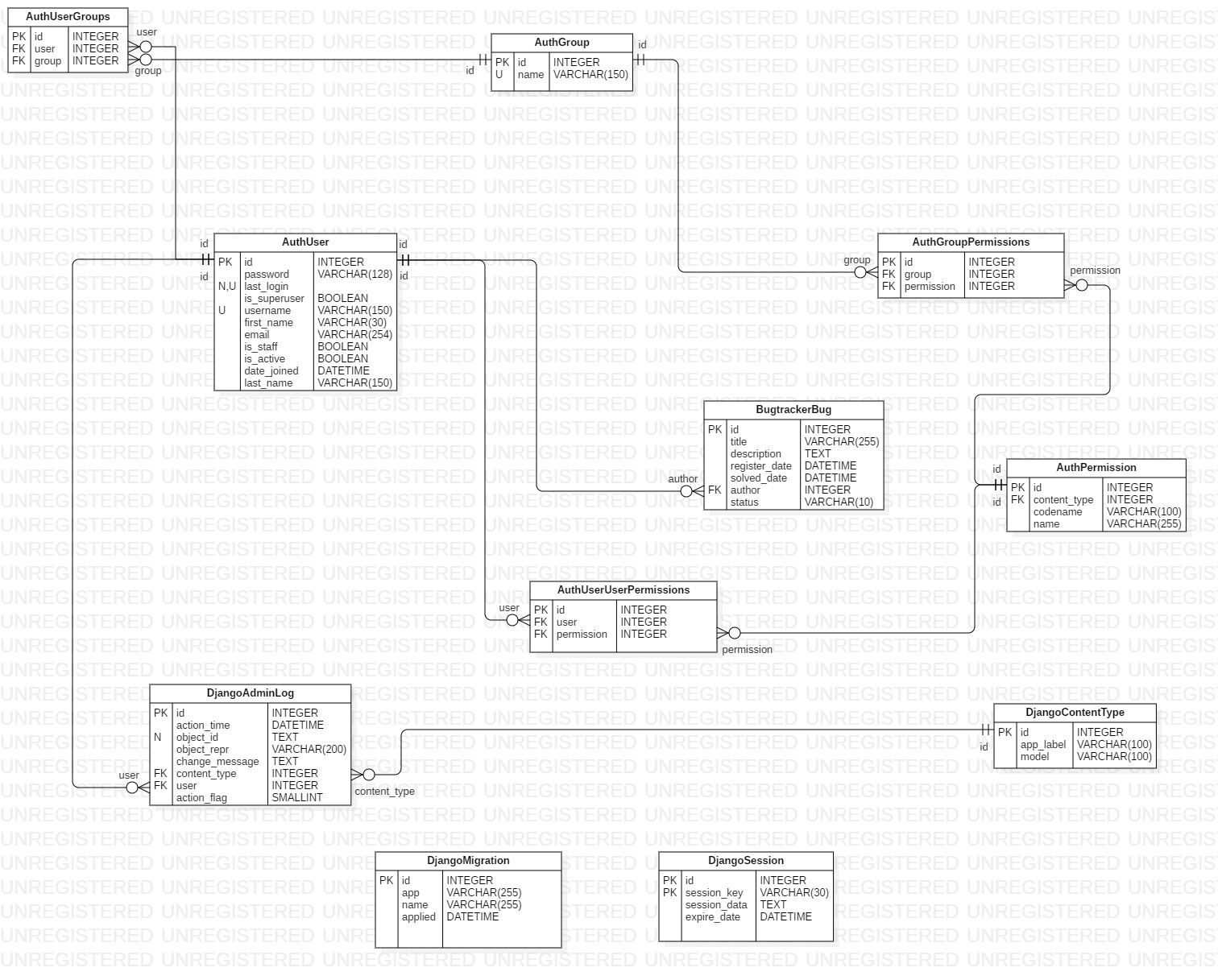
****

1. **Diagrama de secventa si colaborare pentru update bug:**

****

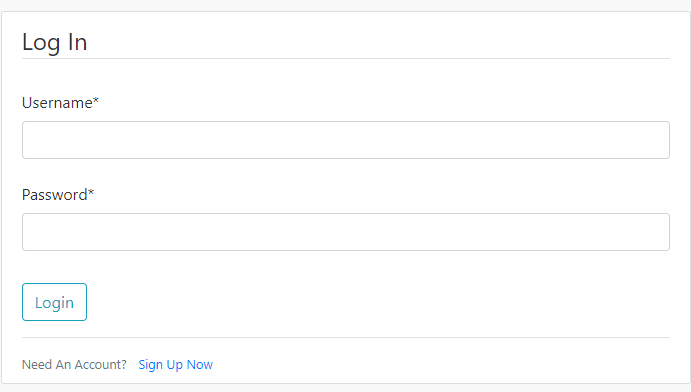
1. **Diagrama bazei de date:**

**Baza de data este formata din 11 tabele. Aceasta a fost realizata in Sqlite3.**

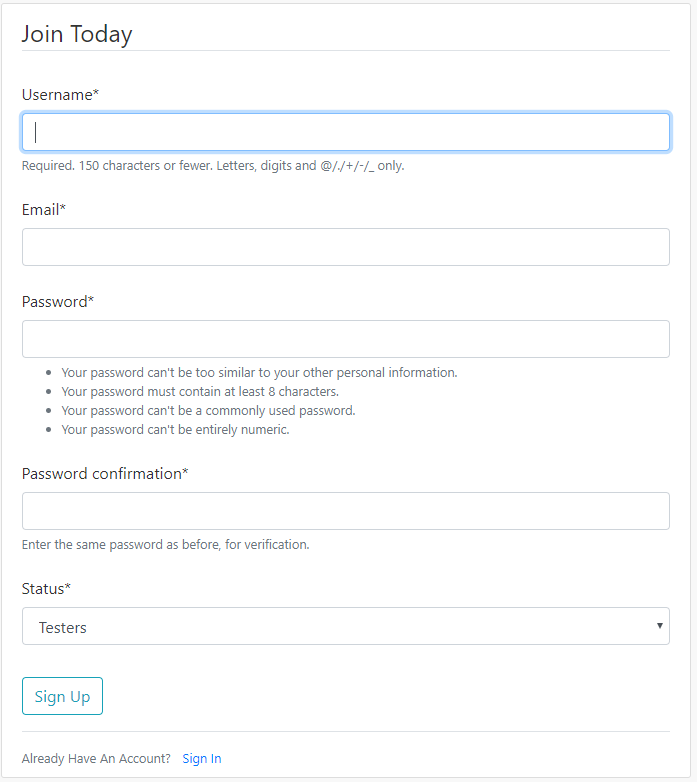
****

1. **Diagrama de ferestre:**
2. **Modul de functionare al aplicatiei:**

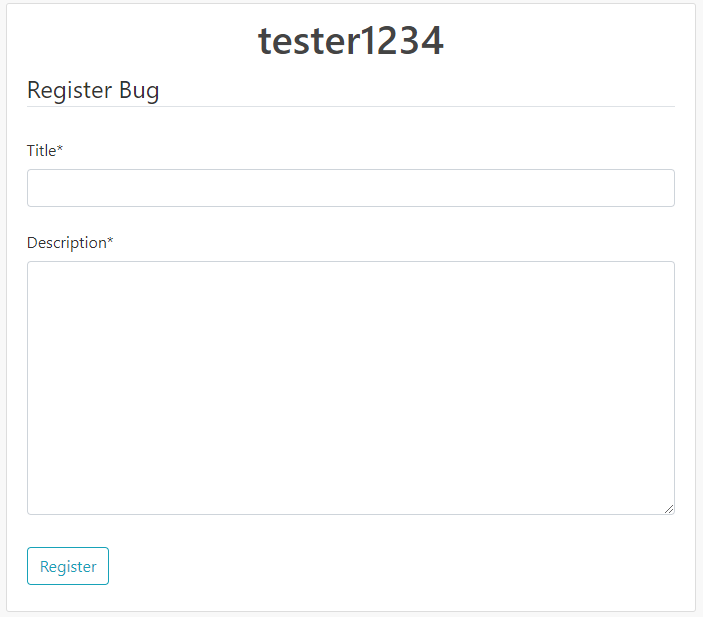
* La executarea si accesarea paginii Home a aplicatiei web va aparea pagina principala. Meniul principal contine butoanele Home, About, Register, Login catre paginile aferente;
* Pagina de Login contine formularul de logare:



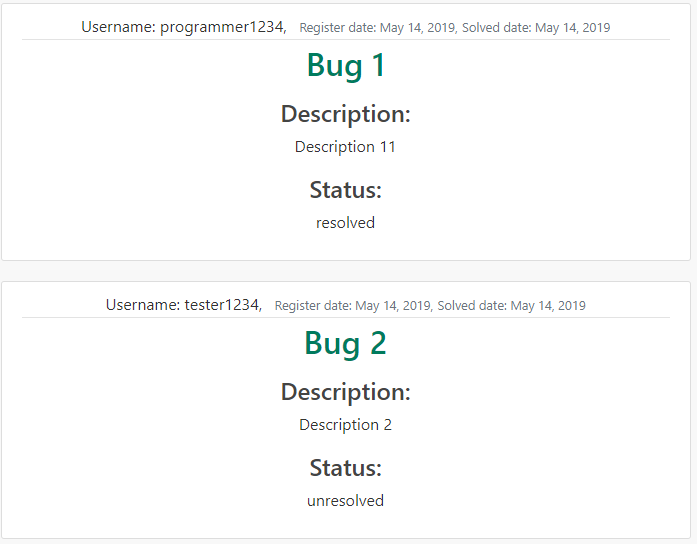
* Utilizatorul va trebui sa introduca numele de utilizator si o parola, dupa care se va loga in sistem apasand butonul Login. In cazul in care utilizatorul se autentifica cu succes, va fi redirecitonat catre pagina Home. In caz contrar sistemul va afisa mesaje legate de inputurile incorecte;
* Pagina de Register contine formularul de inregistrare:

****

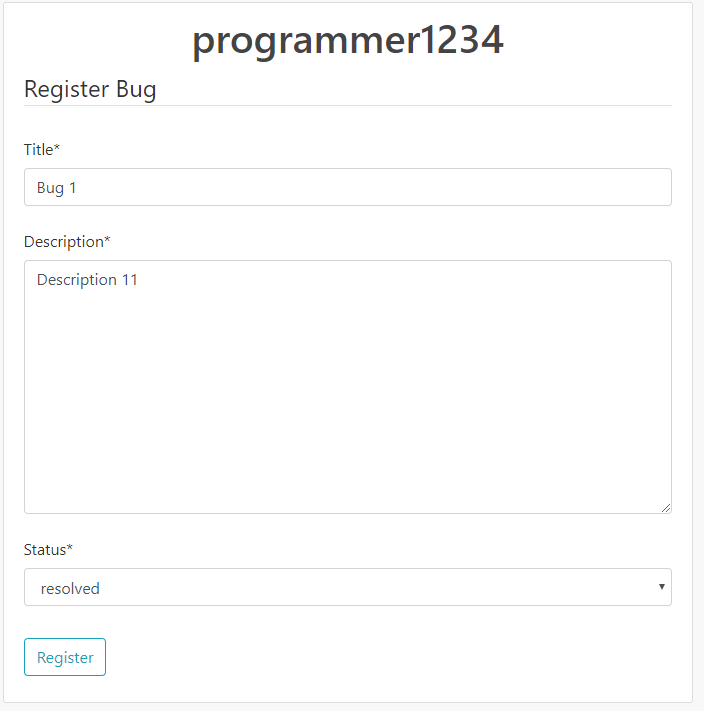
* Utilizatorul va trebui sa introduca numele de utilizator, emailul, parola, confirmarea parolei si sa selecteze din lista de statusuri(Testers/Programmers) un grup. In cazul in care utilizatorul se inregistreaza cu success, va fi redirectionat catre pagina de login iar sistemul va afisa un mesaj de confirmare a inregistrarii. In caz contrar sistemul va afisa mesaje legate de inputurile incorecte;
* Daca utilizatorul este logat ca “Testers” atunci in meniul principal va aparea butonul de Register Bug, iar daca este logat ca “Programmers”, butonul Bug List;
* Utilizatorii din grupul “Testers” au pe pagina principala “Home” lista bugurilor unresolved iar utilizatorii din grupup “Programmers” lista bugurilor resolved si unresolved;
* Pagina Register Bug contine formularul de inregistrare bug:

****

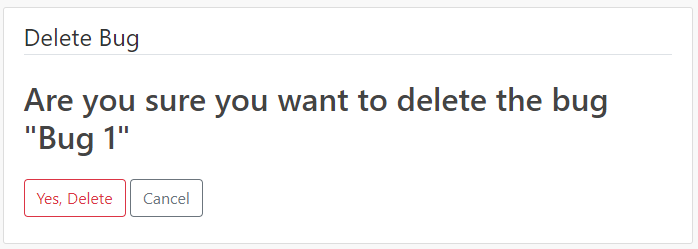
* Utilizatorul va trebui sa introduca titlul bugului si descrierea acestuia. In cazul in care se inregistreaza cu success, va fi redirectionat catre pagina principala iar sistemul va afisa un mesaj de confirmare a inregistrarii. In caz contrar sistemul va afisa mesaje legate de inputurile incorecte;
* Pagina Bug List contine lista tuturor bugurilor din sistem asezate pe orizontala:



* La apasarea titlului unui bug sistemul va redirecitona utilizatorul catre pagina de detalii bug.
* Pagina Bug Detail contine detaliile legate de bug cat si 2 butoane(Delete, Update) in cazul in care bugul este resolved sau 1 buton(Update) in cazul in care bugul este unresolved.
* Butonul de Update redirectioneaza utilizatorul catre pagina de Update Bug:



* Utilizatorul poate modifica titlul, descrierea si statusul bugului. In cazul in care se inregistreaza cu success, va fi redirectionat catre pagina principala. In caz contrar sistemul va afisa mesaje legate de inputurile incorecte;
* Butonul de Delete redirectioneaza utilizatorul catre pagina de confirmare Delete:



* Pagina de Confirmare Delete contine un mesaj si 2 butoane(Yes, Delete, Cance). In cazul in care utilizatorul apasa butonul “Yes, Delete” atunci sistemul sterge bugul din lista de buguri si redirectioneaza catre pagina principala. In cazul in care utilizatorul apasa butonul cancel este redirectionat catre pagina de detalii bug;