Logo, company name

Description automatically generated

**FACULTATEA DE AUTOMATICA SI CALCULATOARE**

**DEPARTAMENTUL CALCULATOARE**

**PROIECT**

**SISTEM BANCAR**

**Elena-Andreea Sand, grupa 30223**

**Maria-Simona Tivadar, grupa 30223**

**An academic: 2021-2022**

**Cuprins**

1. Introducere

2. Diagrama de tabele

3.Lista de proceduri

4. Interfata SWING: Log-in si log-out (schimbare de parola)

5. Tipuri de utilizatori si drepturile lor – meniu utilizator

6. Concluzii finale

1. **Introducere**

În cadrul acestui proiect, am realizat o aplicaţie cu interfaţă grafică care va oferi utilizatorilor funcţionalităţi, corespunzător cu drepturile pe care aceştia le deţin în sistem, descrise prin intermediul unor roluri (administrator, angajat, client).

Proiectul nostru urmareste implementarea unui sistem informatic destinat gestiunii unei **structuri bancare**. Aplicatia foloseste un sistem de gestiune pentru baze de date MySQL, iar interactiunea cu aceasta va fi realizata doar prin interfata grafica. Functionalitatile oferite de program vizeaza operatii ce tin de gestiunea clientilor, angajatilor si administrarea operatiilor curente din cadrul unei banci.

Aplicaţia poate fi accesată, pe baza unui proces de autentificare, de către mai multe tipuri de utilizatori: clienti, angajati, administratori. Pentru fiecare tip de utilizator se reţin in baza de date MySQL informații precum CNP, nume, prenume, adresa, număr de telefon, email, cont IBAN, numărul de contract.

Fiecare utilizator își poate vizualiza datele personale imediat după ce acceseaza sistemul informatic, fără a avea însă posibilitatea de a le modifica. Totodată, programul ofera și o funcționalitate pentru deautentificare, prin care se revine la fereastra care solicită datele de acces, astfel încât și un alt utilizator să îl poată folosi ulterior, fără a fi necesară repornirea sa. Mai mult decat atat, se ofera posibilitatea de schimbare de parola a contului utilizatorului, modificand totodata datele din tabele.

**Punctele atinse in realizarea proiectului sunt:**

• crearea tabelelor mentionate in cerinta

• definitivarea mai multor tipuri de utilizatori

• scrierea de porceduri in MYSQL, aplicabile fiecarui tip de utilizator

• creearea interfetei grafice pentru a facilita interactunea utilizatorului cu aplicatia

•modificarea bazei de date in functie de evenimentele generate in interfata SWING (ActionListener)

**In realizarea acestui proiect s-au folosit:**

• programarea în limbaj OOP

• manipularea bazelor de date folosind MySQL;

• folosirea JDBC - API pentru SQL Connection

1. **Diagrama de tabele**

In imaginea atasata se pot observa indeaproape tabelele folosite si conexiunile realizate intre acestea (prin relatii 1….1, 1….n, n….n). Acestea alcatuiesc baza de date pentru sistemul nostru bancar. Procedurile si operatiile efectuate au efect direct asupra modificarii informatiilor existente in tabelele de mai jos. Conexiunile dintre tabelele de mai jos este realizata prin relatiile de Primary key – Foreign Key, pentru a pastra unicitatea tuplelor si, mai mult decat atat, utilizarea eficienta a procedurilor pe datele care populeaza tabelele noastre.

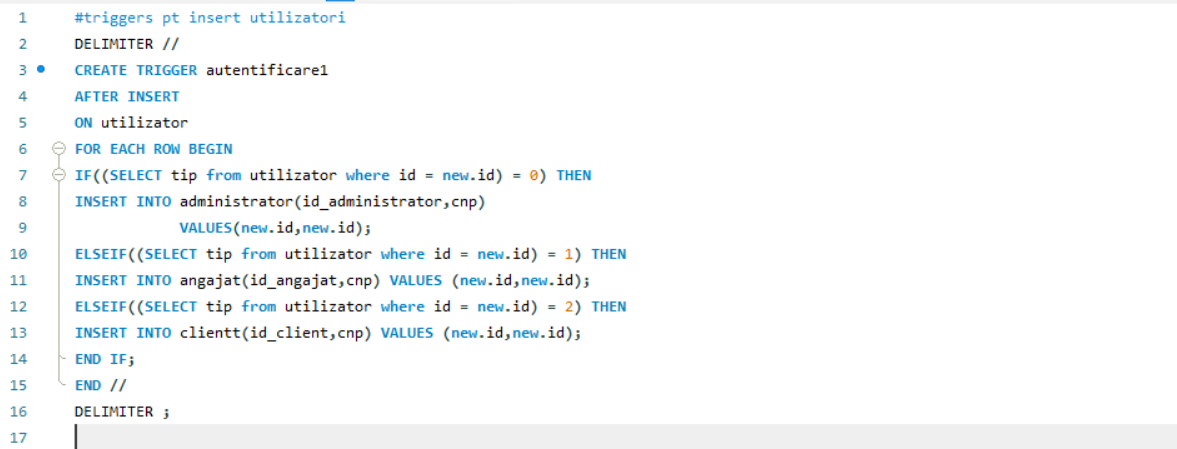
O imagine care conține text, interior

Descriere generată automat

Pentru realizarea acestei baze de date au fost construite si utilizate 16 tabele si anume:

1. Utilizator
2. Client
3. Angajat
4. Administrator
5. Cont
6. Tranzactie
7. Transfer
8. Depozit
9. Factura
10. Conturi\_favorite
11. Salarii
12. Departament
13. Cereri\_angajat
14. Cereri\_admin
15. Cont\_special
16. Card
17. **Lista de proceduri**

Pentru a atinge toate cerintele mentionate in descirerea proiectului **sistem bancar**, am utilizat numeroase proceduri scrise in MySQL, pe care le-am apelat mai apoi prin intermediul interfetei grafice. Mai mult decat atat, am folosit un trigger pentru inserarea si autentificarea user-ilor.



Acestea vin ca ajutor in ceea ce priveste popularea tabelelor – astfel incat, orice utilizator inregistrat in baza de date, apartine unuia dintre cele 3 tipuri de utilizator (administrator, angajat, client), astfel incat, popularea celor 3 tabele mentionate anterior sa se faca automat, creand facilitate si rapiditate in manipularea datelor din tabele.

Procedurile utilizate sunt:

DESCHIDERE\_CONT – prin care orice utilizator este capabil sa isi deschida un cont la banca noastra…. Se va genera un IBAN corespunzator (cu tara, moneda, si abrevierea bancii) urmat de cifrele necesare pentru validarea IBAN-ului. Mai mult decat atat, procedura nu doar ca updateaza campul IBAN din Utilizator, ci si introduce o tuple noua in tabelul CONT, specificand toate detaliile contului (tip: economii/curent, id-ul posesorului, soldul current, data crearii si data expirarii). In plus, aceasta metoda verifica si cerinta referitoare la faptul ca un client poate avea maximum 5 conturi (in cazul in care atinge limita, creearea unui cont nu este posibila).

DESCHIDERE\_DEPOZIT – permite unui client sa deshcida un depozit, pe o perioada aleasa de el, intr-un cont detinut de el,ales de asemenea. Mai mult decat atat, dupa mentionarea perioadei, dobanda este calculata automat prin procedura, respectand cerinta proiectului. Odata cu calcularea dobanzii, suma este actualizata in urma calculului ei total. Un punct important al acestei proceduri este verificarea sumei depuse in depozit – daca aceasta depaseste 500.000 de lei, clientul are nevoie de aprobarea unui angajat pentru deschiderea contului fara dificultati, si popularea tabelei DEPOZIT cu datele mentionate de client in interfata grafica.

LICHIDARE-CONT – permite clientului sa inchida unul din conturile detinute de el, indiferent de suma existenta in acesta. Procedura ofera update situatiei “numar conturi”. Mai mult decat atat, odata cu stergerea permanenta a contului, sunt sterse toate informatiile bancii care tin de cont ( din tabelele: cont, tranzactii, cereri, depozit) – un cont neexistent este invalid bancii si nu mai poate fi folosit in tranzactii viitoare.

LICHIDARE\_DEPOZIT – permite clientului sa inchida unul din depozitele acestuia, prin mentionarea id-ului de depozit. Atunci cand clientul decide sa-si inchida depozitul, suma din el va fi transferata contului current, cu o penalizare de 2% din suma totala. Daca valoarea depozitului depaseste 500.000 lei , atunci operatiunea poate fi finalizata cu success doar dupa aprobarea unui administrator. Numai atunci, celelalte funcitonalitati mentionate deja, vor fi efectuate cu success.

CERERE\_CARD – aceasta procedura permita clientului sa ceara un card fizic pentru unul din conturile existente. Aceasta procedura poate fi dusa la bun sfarsit doar dupa aprobarea atat a angajatului cat si a adminului. Dupa ce clientul primeste aprobarea necesara, cererea va fi efectuata cu succes.

PLATA\_FACTURI – permite clientului sa plateasca diferite facturi, la diferiti furnizori, prin mentionarea unui ID de factura si a IBAN-ului unui cont personal, din care doreste sa fie retrasa suma de platit factura. Procedura verifica daca soldul clientului este suficient pentru plata facturii. In caz favorabil, plata este efectuata, updatandu-se sold-ul curent din CONT, iar apoi se genereaza niste detalii ale facturii platite.

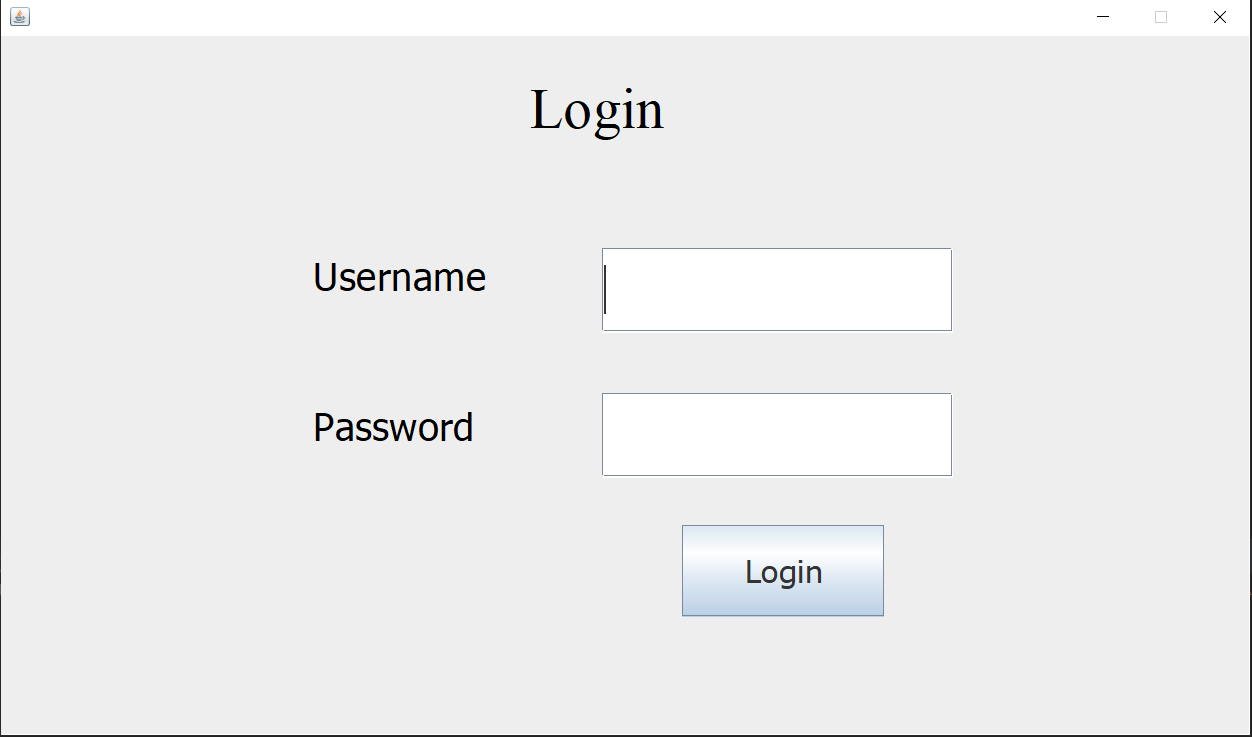
INITIERE\_TRANSFER – ofera functionalitatea de transfer intre 2 conturi (acestea pot sau nu sa fie create la aceeasi banca). Un utilizator decide sa efectueze un transfer, acesta se iniializeaza cu statusul de CREATED. Se verifica daca soldul curent este suficient pentru a efectua transferul. In caz contrat, statusul transferului trece pe ERROR. Dupa aceea, se verifica apartenta bancii pentru fiecare cont. Daca cele doua conturi sunt create la banci diferite, banca noastra va extrage un commission de 1%, astfel incat suma care ajunge la receptor reprezinta 99% din suma transferata de emitator. Aceasta situatie este evitata pentru transferul intre conturi de la aceeasi banca. Odata initiat transferul, se inregistreaza o cerere la angajatul functionar care, indiferent de suma transferata, trebuie sa aprobe aceasta tranzactie. Daca tranzactia este aprobata iar soldul este suficient, statusul transferului va trece pe SUCCESSFUL. Daca nu, statusul transferului este schimbat in ERROR.

TRANSFER\_FAVORIT – aceasta procedura vine cu o functionalitate in plus procedurii anterioare, si anume, permite utilizatorului sa efectueze un transfer cu o mai mare facilitate, fara a mentiona IBAN-ul receptorului. Transferul are loc doar prin mentionarea numelui persoanei care urmeaza sa primeasca suma de bani. Se face o verificare amanuntita – fiecare relatie dintre 2 conturi, care pot sa fie sau nu favorite intre ele, se verifica. Astfel, un client nu poate tansfera bani cu usurinta unui cont care nu este trecut in lista sa de favorite. Aceasta apartenta la favorite se verifica in procedura noastra de transfer favorit.

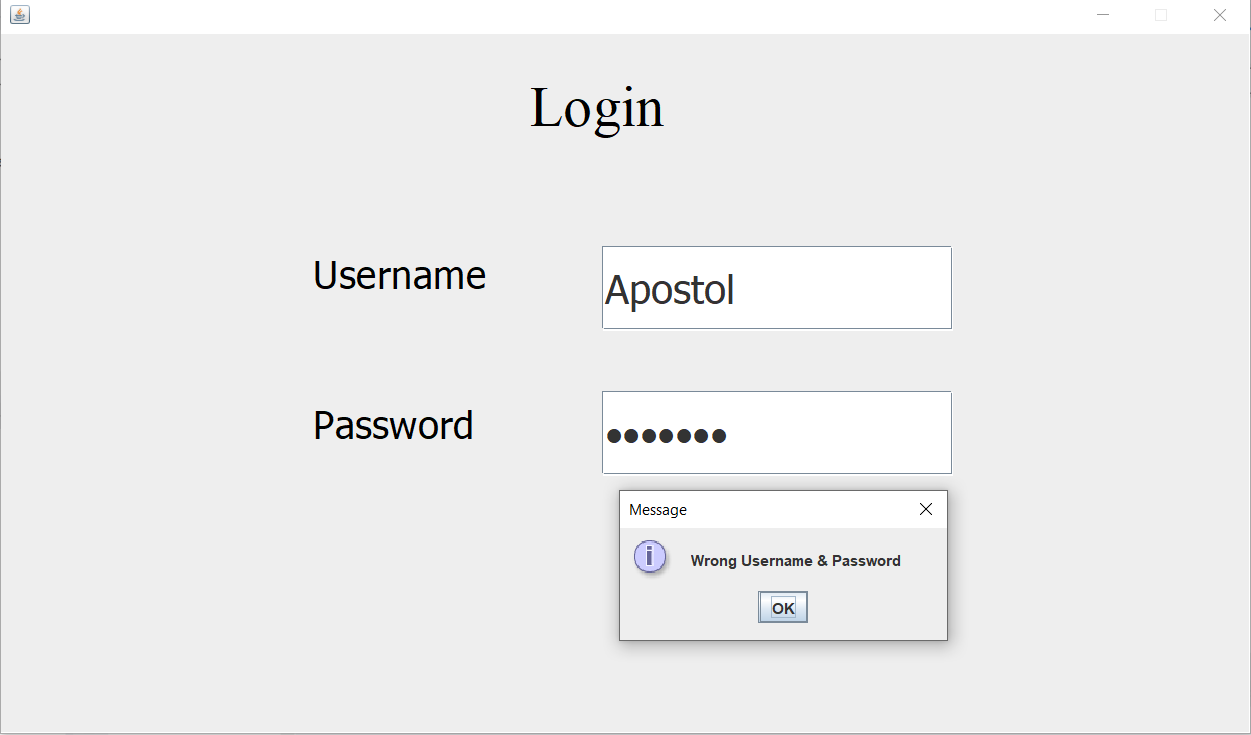
**4. Interfata SWING: Log-in si log-out**

Pentru realizarea interfectei grafice am utilizat JAVA SWING GUI – acesta permite crearea diferitelor Frame-uri, Panel-ul si ActionListener-i care ofera o mare facilitate in interactiuna cu aplicatia si manipularea datelor din baza noastra de date din MYSQL.

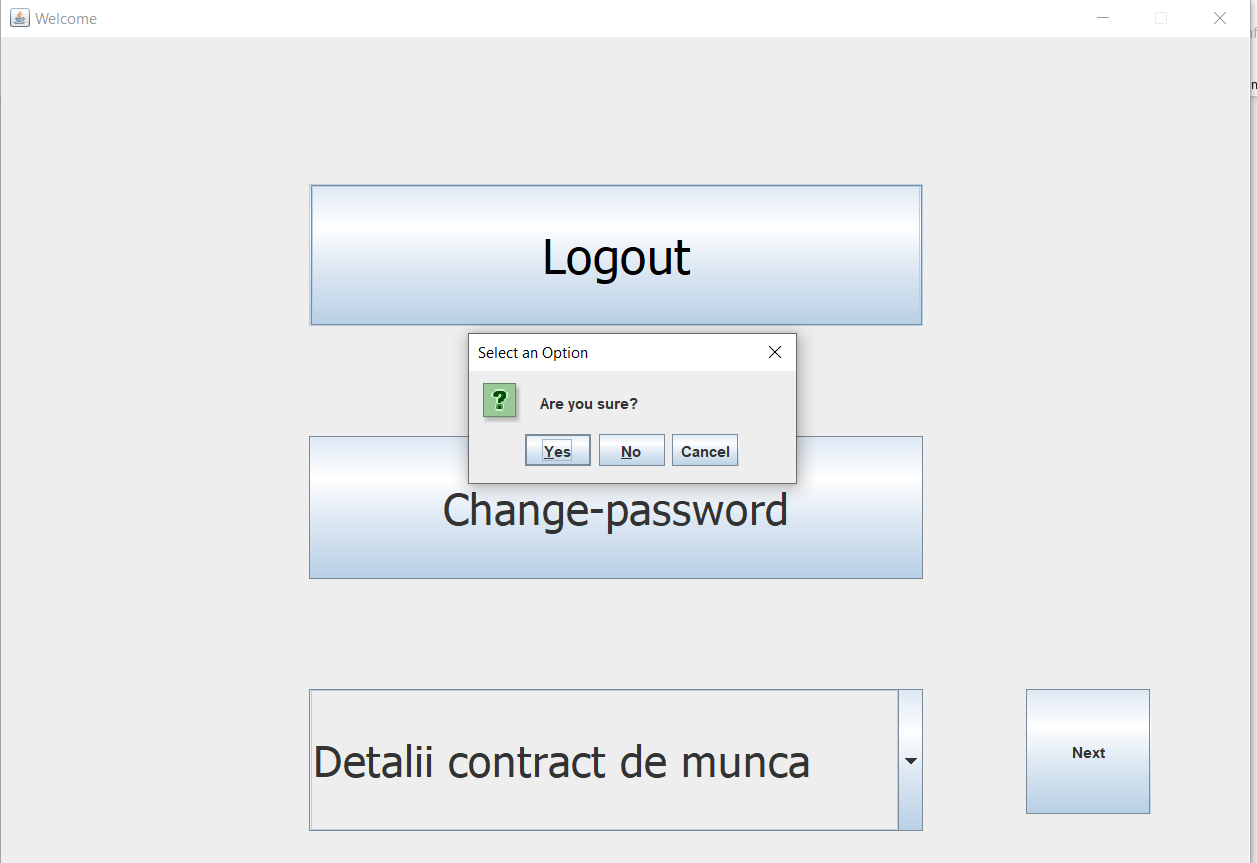
Aplicatia debuteaza cu ecranul de log-in . Utilizatorului i se cere introducerea numelui si a parolei contului sau.

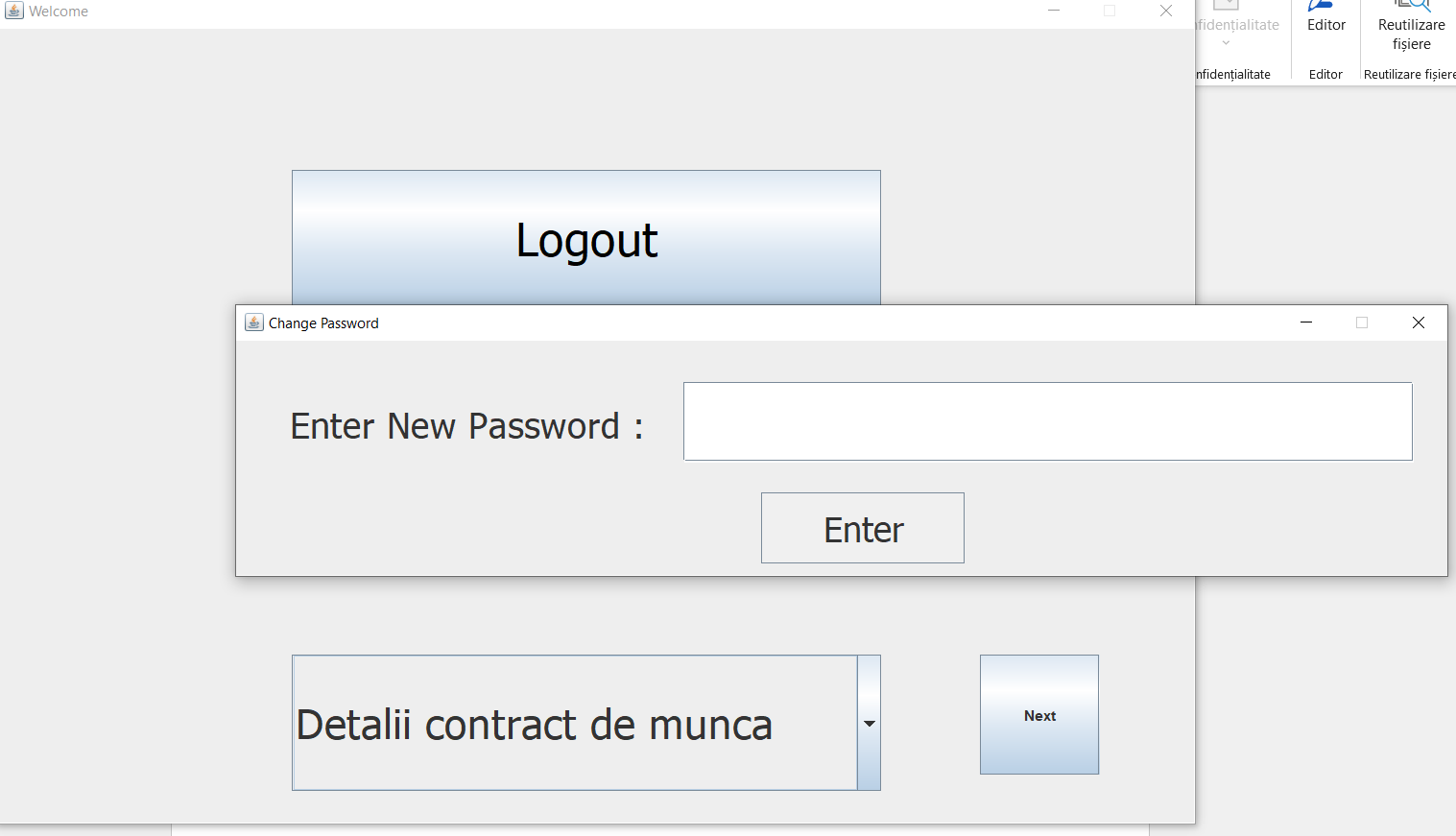


De aici, user-ul poate intampina 2 situatii…parola si numele nu se potrivesc, log-in-ul intampina o eroare, se cere repetarea sa:



In caz contrar, user-ul este autentificat cu success si are 3 optiuni: LOG-OUT, Change Password, Meniu personalizat in functie de tipul utilizatorului.

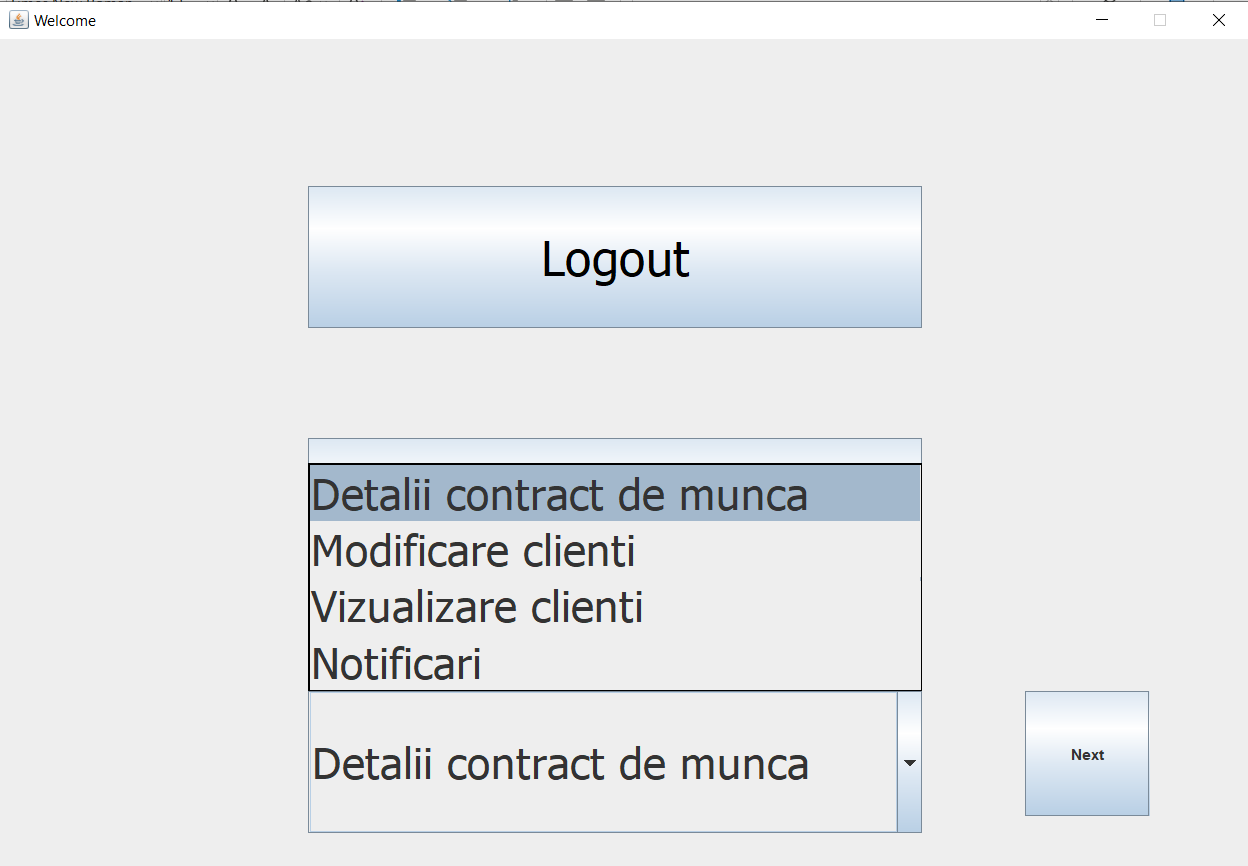
LOG-OUT: 

Change Password: 

**5. Tipuri de utilizatori si drepturile lor – meniu utilizator**

Dupa cum am mentionat anterior, exista 3 tipuri de utilizator. In functie de tipul utilizatorului, meniul ofera functionalitati diferite.

Mai exact, pentru administrator, meniul personalizat arata astfel, ingloband toate drepturile adminului, mentionate in cerinta proiectului:

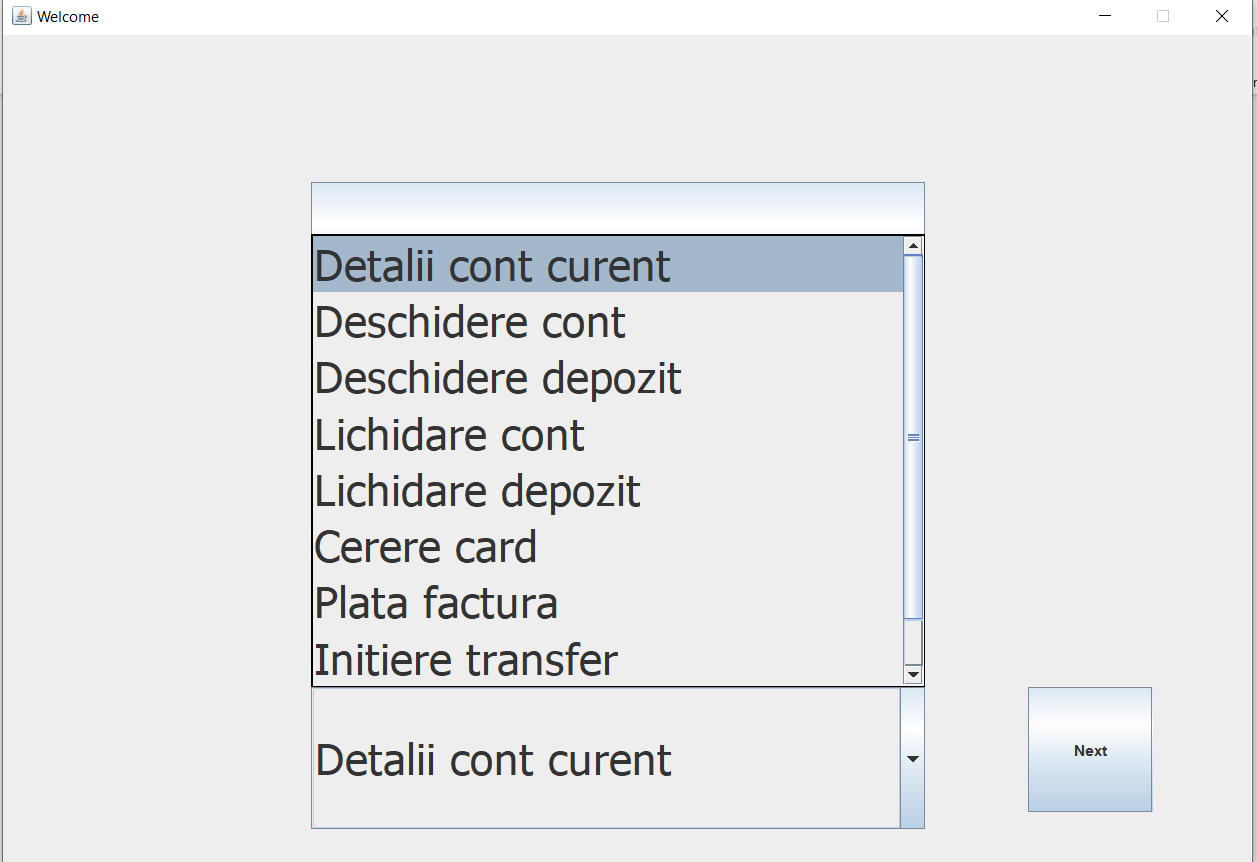


Prin apasarea butonului NEXT, optiunea utilizatorului este inregistrata de interfata grafica, si se deschide o noua fereastra, in functie de alegerea din meniu.

Pentru angajat:



Pentru client:



**6. Concluzii finale**

Asadar, sistemul nostru bancar retine informatii atat despre clientii sai, cat si despre conturile sau tranzactiile efectuate de acestia. Pe langa tranzactiile uzuale (deschidere cont, deschidere depozit, efectuare transfer, vizualizare clienti), clientul poate plati facturi, angajatul poate aproba diferite cereri care ii apar sub forma unor notificari, adminul poate modifica date despre orice utilizator.

Interfata grafica realizata in JAVA SWING ofera facilitate oricarui utilizator pentru a-si indeplini sarcinile de zi cu zi. Mai mult decat atat, autentificarea si deautentificarea permite mai multor utilizatori sa foloseasca aplicatia fara a fi necesara repornirea ei – o cerinta importanta din proiectul nostru.

Utilizatorul primeste diferite notificari / atentionari : cand suma e prea mare, cand e nevoie de aprobare, cand greseste parola si asa mai departe. Acesta nu trebuie sa lucreze deloc cu baza de date, aceasta fiind modificata si updatata fara a fi nevoie da manipularea sa directa. Conexiunea intre JAVA si MYSQL permite modificarea cu usurinta si cu acuratente a bazei de date, prin actiunile de tip ActionListener surprinse in blocuri de tip try-catch.