

# **Documentatie**

## **PROIECT 2**

### *1. Conturi financiare*

Subpuncte:

1. Creare cont financiar
2. Editare cont financiar
3. Sterge cont financiar
4. Vizualizare cont financiar

#### *1.1. Creare cont financiar*

Se citesc si se valideaza datele necesare pentru crearea contului financiar, iar daca datele nu sunt valide utilizatorul este anuntat printr-un mesaj specific. Datele sunt adaugate in fisierul "conturiFinanciare.txt" .

FORMAT: cont;denumire;tip.

#### *1.2. Editare cont financiar*

Se citeste numarul contului, daca contul nu exista utilizatorul este anuntat printr-un mesaj. Daca se gaseste contul care se doreste sa se modifice, se reintroduc datele contului modificate si sunt salvate in fisierul "conturiFinanciare.txt", iar vechea inregistrare este stearsa.

### *1.3. Stergere cont financiar*

Se citeste numarul contului, daca contul nu exista utilizatorul este anuntat printr-un mesaj. Daca se gaseste contul care se doreste sa fie sters, acesta se sterge din fisierul "conturiFinanciare.txt".

### *1.4. Vizualizare conturi financiare*

Se citesc datele din fisierul "conturiFinanciare.txt" si se afiseaza in consola, iar daca nu sunt conturi financiare de vizualizat este afisat un mesaj.

## *2. Tranzactii*

Subpuncte:

1. Inregistrare tranzactii
2. Vizualizare tranzactii

### *2.1. Inregistrare tranzactii*

Se citesc datele necesare pentru a inregistra o noua tranzactie, se valideaza datele, iar daca acestea nu sunt valide utilizatorul este instiintat printr-un mesaj specific. Tranzactia este salvata in fisierul "tranzactii.txt".

FORMAT: numar tranzactie; tip tranzactie; numar cont; operatie(+/-); beneficiar; suma; data

### *2.2. Vizualizare tranzactii*

Se citesc datele din fisierul "tranzactii.txt" si se afiseaza in consola, iar daca nu sunt tranzactii de vizualizat se afiseaza un mesaj.

## *3. Calcul sold*

Se citeste contul financiar pentru care se doreste calculul soldului, iar daca acesta nu exista, utilizatorul este anuntat printr-un mesaj. Daca contul exista, se calculeaza soldul contului cu ajutorul datelor din fisierul "tranzactii.txt" si se afiseaza soldul contului in consola.

## *4. Rapoarte financiare*

Subpuncte:

1. Extras de cont
2. Solduri
3. Registre de tranzactii
4. Rapoarte de cheltuieli

### *4.1. Extras de cont*

Se citește contul pentru care se dorește să se genereze extrasul de cont, iar dacă acesta nu există utilizatorul este anunțat printr-un mesaj. Dacă contul este existent, se citește perioada (luna/an) pentru care se dorește generarea extrasului de cont, se validează data, iar dacă aceasta este invalidă utilizatorul este anunțat printr-un mesaj. Dacă perioada introdusă este validă, se afișează toate tranzacțiile din perioada respectivă și soldul curent al contului, datele fiind citite din fișierul “tranzactii.txt”, iar dacă nu există tranzacții din acea perioadă se va afișa un mesaj sugestiv.

### *4.2. Solduri*

Se calculează soldul fiecărui cont (din fișierul “conturi.txt”) cu ajutorul datelor din fișierul “tranzactii.txt” și se afișează soldurile în consolă. Dacă nu există solduri de afișat, utilizatorul va fi anunțat printr-un mesaj.

#### *4.3. Registre de tranzactii*

Se citeste perioada (luna/an) pentru care sa se afiseze registrul de tranzactii. Perioada este validata, iar daca aceasta este invalida se va afisa un mesaj sugestiv, iar in cazul in care perioada este valida se vor afisa tranzactiile din perioada respectiva, indiferent de cont. Tranzactiile sunt citite din fisierul "tranzactii.txt". Daca nu exista tranzactii de afisat pentru perioada aleasa, se va afisa un mesaj specific.

#### *4.4. Rapoarte de cheltuieli*

Se citeste perioada (luna/an) pentru care sa se genereze raportul de cheltuieli. Perioada este validata, iar daca aceasta este invalida se va afisa un mesaj sugestiv, iar in cazul in care perioada este valida se vor afisa toate tranzactiile (din fisierul "tranzactii.txt") care tin de cheltuieli, indiferent de cont, iar la final se afiseaza suma totala cheltuita in perioada aleasa.

## *5. Managementul datelor clientilor*

Subpuncte:

1. Adauga client
2. Modifica client
3. Sterge client
4. Vizualizare clienti

### *5.1. Adauga client*

Se citesc si se valideaza datele necesare pentru adaugarea unui nou client, iar daca datele sunt invalide utilizatorul va fi anuntat printr-un mesaj specific. Daca datele introduse sunt valide, acestea vor fi salvate in fisierul "clienti.txt".

FORMAT: id client; nume client; adresa client

### *5.2. Modifica client*

Se citeste id-ul clientului pentru care se doreste modificarea datelor, iar in cazul in care acesta nu exista utilizatorul va fi instiintat printr-un mesaj. Daca id-ul clientului exista, se vor citi datele clientului si se vor salva modificarile in fisierul "clienti.txt", iar datele vechi vor fi sterse.

### *5.3. Sterge client*

Se citeste id-ul clientului pentru care se doreste modificarea datelor, iar in cazul in care acesta nu exista utilizatorul va fi instiintat

printr-un mesaj. Daca id-ul clientului exista, datele clientului vor fi sterse din fisierul "clienti.txt".

#### *5.4. Vizualizare clienti*

Se incarca datele despre clienti din fisierul "clienti.txt" si se afiseaza in consola.

## *6. Securitate si autentificare*

Subpuncte:

1. Cont nou
2. Autentificare

### *6.1. Cont nou*

Se citeste user-ul si parola pentru crearea unui cont nou. User-ul trebuie sa fie unic, iar daca cel introdus exista deja va aparea un mesaj pe ecran pentru instiintarea utilizatorului. Parola trebuie sa aiba minim 8 caractere, iar daca aceasta nu este valida va aparea un mesaj specific pe ecran. Daca user-ul si parola sunt valide, acestea vor fi salvate in fisierul "conturi.txt".

FORMAT: user;parola

## *6.2. Autentificare*

Se citeste user-ul si parola si se verifica daca datele coincid cu cele din fisierul "conturi.txt". Daca datele nu sunt valide, se va afisa un mesaj pentru instiintarea utilizatorului, iar in cazul in care datele coincid utilizatorul va fi logat in contul sau.

## *7. Audit si logging*

La fiecare autentificare se salveaza user-ul, data si ora in fisierul "operatiiLogare.txt", cat si fiecare operatie efectuata de utilizator pe perioada cat este logat. La delogare (iesire din program) se salveaza in acelasi fisier user-ul, data si ora delogarii, iar apoi se salveaza toata activitatea in fisierul "audit.txt".

La alegerea optiunii de audit si logging din meniu se va afisa in consola user-ul, data si ora logarii, cat si a delogarii, impreuna cu fiecare operatiune efectuata pentru toate datele salvate in fisierul "audit.txt".



## 8. Structuri de date

### 1. Structura de date pentru conturile utilizatorilor

```
struct Clienti  
{  
    char user[50], password[50];  
} client;
```

user - user-ul utilizatorului

password - parola utilizatorului

### 2. Structura de date pentru conturi financiare

```
struct conturiFinanciare  
{  
    char cont[20], denumire[50], tip[15];  
} contFinanciar;
```

cont - contul financiar

denumire - denumirea contului financiar

tip - tipul contului financiar

### 3. Structura de date pentru tranzactii

```
struct dateFisier
{
    char text[1000];
} text[1000];
```

text - textul salvat in fisierul "tranzactii.txt"

Cu ajutorul acestei structuri se rescrie fisierul cand se doreste modificarea/stergerea unei tranzactii.

### 4. Structura de date pentru calcularea soldului

```
struct solduri
{
    char cont[20];
    double sold;
} sold[1000];
```

cont - contul financiar

sold - soldul contului

## 5. Structura de date pentru extrase de cont

```
struct Extrase
{
    char luna[3], an[5];
} extras;
```

luna - luna pentru care sa se genereze extras  
an - anul pentru care sa se genereze extrasul

## 6. Structura de date pentru clienti

```
struct Clienti2
{
    char id[10], nume[50], adresa[100];
} client2;
```

id - id client  
nume - nume client  
adresa - adresa client

## 7. Structura de date pentru operatii clienti (modificare/stergere)

```
struct OperatiiClienti
{
    char text[1000];
} opClient[1000];
```

text - textul din fisierul "clienti.txt"

Cu ajutorul acestei structuri de date se rescrie fisierul "clienti.txt" pentru modificarea/stergere a unui client.

## 8. Structura de date pentru operatiile efectuate

```
struct Operatii
{
    char operatie[5];
    int cate;
} op;
```

operatie - numarul operatiei din meniu efectuate

cate - numarul operatiilor din meniu efectuate

Cu ajutorul acestei structuri se tine evidenta operatiilor efectuate de catre user cat timp este logat, pentru a le putea salva in cele din urma in fisierul "audit.txt", astfel tinand o evidenta a tuturor operatiilor efectuate de catre fiecare user, la fiecare logare.

## 9. Structura de date pentru audit

```
struct Audit
{
    char op[10];
} audit;
```

op - operatia efectuata de catre utilizator

Cu ajutorul acestei structuri se afiseaza mesajul specific fiecarei operatii efectuate de catre utilizator, fiind utila la afisarea in consola a datelor pentru optiunea Audit si Logging.