

*Rapport du mini-projet*

8<sup>ème</sup> Mini Projet : Gestion de comptes  
bancaires en langage C



Réalisé par : GARHOUM Hafid

Encadré par : Mr. Adil HADDI

API 2 | Section 2 | groupe 3  
Année Universitaire : 2022-2023

## Résumé

Le projet de gestion de comptes bancaires en langage C a pour but de permettre aux clients de la banque de gérer leurs comptes bancaires en toute simplicité.

Le rapport comprend une introduction qui explique le contexte du projet et les motivations qui ont conduit à son développement. Il inclut également une table des matières qui décrit les différentes sections du rapport.

La description de chaque partie du code est ensuite présentée. Cela comprend une description de l'interface utilisateur, de la logique de l'application et des algorithmes utilisés pour implémenter les différentes fonctionnalités.

Enfin, le rapport décrit les tests réalisés pour vérifier le bon fonctionnement de l'application, ainsi que les perspectives d'amélioration pour le futur.

En conclusion, le mini projet de gestion de comptes bancaires en langage C a réussi à fournir une solution simple et efficace pour la gestion des comptes bancaires pour les clients de la banque. Le rapport décrit en détail les différentes parties du projet et les tests réalisés pour garantir son bon fonctionnement.

# Sommaire

I- Introduction.....	4
II-Analyse.....	5
III- Réalisation.....	6
IV- Conclusion.....	7
Annexe.....	8

# I. Introduction

La gestion des comptes bancaires consiste à surveiller les entrées et sorties d'argent et à maintenir un suivi précis des transactions financières, sans laisser aucune marge à l'erreur.

L'objectif de ce mini projet est de développer une application en langage C pour gérer les comptes bancaires, qui permet **la création, la consultation, le dépôt et le retrait d'argent sur les comptes**. Tout en gardant une interface simple et intuitive.

Les informations des utilisateurs sont enregistrées dans un fichier .DAT qui s'utilise généralement pour stocker des informations sensibles, comme des paramètres. Il s'agit d'un fichier binaire, créé par un programme et exploité uniquement par lui.

## II. Analyse

L'analyse de ce projet est fondée sur la problématique suivante :

**Comment le langage C a pu faire une révolution dans le monde de finance dans ces débuts, en l'occurrence la gestion des comptes bancaires ?**

En générale, un système simple de gestion des comptes bancaires doit être capable de faire les tâches suivantes :

- **Création de compte**
- **Consultation de compte**
- **Dépôt d'argent**
- **Retrait d'argent**
- **Suppression d'un compte**

Cela permet d'imaginer le type de menu qui va être présent dans ce programme. Il doit ressembler au menu suivant :

```
***** menu *****
```

```
1.Creer un nouveau compte
2.Voir les informations d'un compte
3.Transactions
4.Supprimer un compte
5.Quitter
```

```
Entrer votre choix :
```

### - **Choix 1 : Création de compte**

La première partie du code permet de créer un compte en entrant leur nom, numéro de compte et le solde initial et d'autre informations qui seront enregistrées dans un fichier pour une utilisation future.

### - **Choix 2 : Consultation de compte**

La seconde partie du code permet de consulter les informations de leur compte, y compris le nom, le numéro de compte et le solde.

### - **Choix 3 : Transactions**

La troisième partie du code permet de déposer de l'argent sur leur compte en entrant le montant à déposer ou de retirer de l'argent de leur compte en entrant le montant à retirer. Le solde du compte est mis à jour en conséquence.

### - **Choix 4 : Supprimer un compte**

Cette partie du code permet d'effacer un compte bancaire du système. Et faire la mise à jour des données par la création d'un nouveau fichier .DAT qui va remplacer l'ancien fichier.

### - **Choix 5 : Quitter**

Sortir du programme.

## III. Réalisation

Chaque choix du menu sera mis en œuvre sous forme de fonction distincte pour simplifier la rédaction du programme.

La conception du programme requiert les structures suivantes :

1. **Définir la structure date**

Elle va faciliter la manipulation des dates dans le programme.

2. **Définir la structure Client**

Elle va contenir toutes les informations d'un client comme le nom, l'âge, le numéro du compte, la date de création du compte ...

Certaines des fonctionnalités les plus critiques du programme sont répertoriées ci-dessous :

1. **La fonction : void menu (void)**

Elle affiche un menu ou un écran d'accueil qui vous permet d'exécuter les différentes tâches bancaires énumérées ci-dessous.

2. **La fonction : void nouveaucompte (void)**

Cette fonction permet de créer un nouveau compte client en demandant le nom du client, la date de naissance, le numéro de carte d'identité, l'adresse et le numéro de téléphone, ainsi que d'autres informations personnelles et financières. Vous pouvez choisir parmi une variété de comptes d'épargne, y compris des comptes d'épargne, des comptes courants, des comptes à terme d'un an, des comptes à terme de deux ans et des comptes à terme de trois ans.

3. **La fonction : void chercher (void)**

Cette fonctionnalité vous permet d'accéder aux informations bancaires du client cherché, y compris le numéro de compte, le nom, l'adresse et le numéro de téléphone fournis lors de la création du compte.

4. **La fonction : void transact (void)**

Cette fonction vous permet de déposer et de retirer des fonds d'un compte client spécifique.

5. **La fonction : void supprimer (void)**

Un compte client peut être supprimé à l'aide de ce service en lui donnant le numéro de compte.

6. **La fonction : void close (void)**

Une fonction de la librairie <unistd.h> qui est utilisé fréquemment dans les fonctions du programme. Elle va nous permettre de le quitter.

## IV. Conclusion

En conclusion, ce projet a abouti à la réalisation d'une application de gestion de comptes bancaires fonctionnelle et efficace en utilisant le langage C tout en maintenant une interface utilisateur facile et simple à utiliser. Ce projet montre la capacité du langage C à développer des applications de gestion de comptes bancaires performantes. Ce qui explique pourquoi ce langage programmation a pris de l'envergure lors de son apparition puisqu'il a contribué à la création du monde d'aujourd'hui.

La réalisation de ce projet a également permis de développer de nombreuses compétences clés en programmation en C notamment ce qui concerne le développement et la gestion des programmes en utilisant des menus et des structures. Les connaissances acquises au cours de ce projet seront utiles dans de nombreux autres projets futurs.

En fin de compte, ce projet a été une expérience enrichissante qui a permis de développer de nombreuses compétences techniques et de mieux comprendre les enjeux de la gestion de comptes bancaires.

---

**\*note** : La dernière vacance était la seule opportunité où j'ai pu travailler sur ce mini projet, puisque je suis parti en voyage, Il était impossible de former une équipe, pour cette raison j'ai élaboré le projet tout seul.

# Annexe

- Le code source du mini projet :

```
#include<stdio.h>
#include<stdlib.h>
#include<windows.h>
#include<unistd.h>
#include<conio.h>
int i,j;
int main_exit;
void menu();
struct date
{
    int jour,mois,annee;
};
struct Client{

    char nom[60];
    int acc_no,age;
    char adresse[60];
    char citizenship[15];
    double phone;
    char acc_type[10];
    float amt; // amount
    struct date dob; // date of birth
    struct date deposit;
    struct date withdraw;

    }add,upd,check,rem,transaction;

void fordelay(int j)// effet de retard
{
    int i,k;
    for(i=0;i<j;i++)
        k=i;
}

void nouveaucompte()
{
    int choix;
    FILE *ptr;

    ptr=fopen("record.dat","w+");
    account_no:
    system("cls||clear");
    printf("\t\t\t***** AJOUT *****");
    printf("\n\n\nEntrer la date d\'aujourd\'hui (jj/mm/aaaa):");
    scanf("%d/%d/%d",&add.deposit.jour,&add.deposit.mois,&add.deposit.annee);
    printf("\nEntrer le numero du compte:");
    scanf("%d",&check.acc_no);
    while(fscanf(ptr,"%d %s %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",&add.acc_no,&add.nom,&add.dob.jour,&add.dob.mois,&add.dob.annee,&add.age,&add.a
dresse,&add.citizenship,&add.phone,&add.acc_type,&add.amt,&add.deposit.jour,&add.deposit.mo
is,&add.deposit.annee)!=EOF)
    {
        if (check.acc_no==add.acc_no)
        {printf("Le numero entré est déjà utilisé !");
        fordelay(1000000000);
        goto account_no;
        }
    }
    add.acc_no=check.acc_no;
```



```

        printf("\nEntrer votre nom complet:");
scanf("%s",add.nom);
printf("\nEntrer la date de naissance (jj/mm/aaaa):");
scanf("%d/%d/%d",&add.dob.jour,&add.dob.mois,&add.dob.annee);
printf("\nEntrer votre age:");
scanf("%d",&add.age);
printf("\nEntrer votre adresse:");
scanf("%s",add.adresse);
printf("\nEntrer le numero de votre carte d'identité:");
scanf("%s",add.citizenship);
printf("\nEnter the phone number: ");
scanf("%lf",&add.phone);
printf("\nEntrer le montant a deposer:DH ");
scanf("%f",&add.amt);
printf("\nType du compte:\n\t#Epargne\n\t#Courant\n\t#A terme1(pour 1 annee)\n\t#A
terme2(pour 2 annees)\n\t#A terme3(pour 3 annees)\n\n\tEntrer ton choix:");
scanf("%s",add.acc_type);

        fprintf(ptr,"%d %s %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",add.acc_no,add.nom,add.dob.jour,add.dob.mois,add.dob.annee,add.age,add.adress
e,add.citizenship,add.phone,add.acc_type,add.amt,add.deposit.jour,add.deposit.mois,add.de
posit.annee);

fclose(ptr);
printf("\nCompte créé avec succès!");
add_invalid:
printf("\n\n\t\tEntrer 1 pour afficher le menu principal et 0 pour quitter:");
scanf("%d",&main_exit);
system("cls||clear");
if (main_exit==1)
    menu();
else if(main_exit==0)
    close(1);
else
{
    printf("\nInvalide!\a");
    goto add_invalid;
}
}

void transact(void)
{
    int choix,test=0;
    FILE *old,*newrec;
    old=fopen("record.dat","r");
    newrec=fopen("new.dat","w");

    printf("Entrer le numero du compte bancaire du client :");
    scanf("%d",&transaction.acc_no);
    while (fscanf(old,"%d %s %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d",&add.acc_no,add.nom,&add.dob.jour,&add.dob.mois,&add.dob.annee,&add.age,add.adr
esse,add.citizenship,&add.phone,add.acc_type,&add.amt,&add.deposit.jour,&add.deposit.mois
,&add.deposit.annee)!=EOF)
    {

        if(add.acc_no==transaction.acc_no)
        {
            test=1;
            if(strcmpi(add.acc_type,"A terme1")==0||strcmpi(add.acc_type,"A
terme2")==0||strcmpi(add.acc_type,"A terme3")==0)
            {
                printf("\a\a\a\n\nVOUS NE POUVEZ PAS DEPOSER OU RETIRER DE L'ARGENT DANS
LES COMPTES A TERME!!!!");
                fordelay(1000000000);
                system("cls||clear");
                menu();
            }
        }
        printf("\n\nVous voulez faire un(e))\n1.Depot d'argent'\n2.Retraite

```

```

d'argent'?\n\nEntrer votre choix(1 pour le depot et 2 pour la retraite:");
scanf("%d",&choix);
if (choix==1)
{
    printf("Entrer le montant a depoter :DH ");
    scanf("%f",&transaction.amt);
    add.amt+=transaction.amt;
    fprintf(newrec,"%d %s %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",add.acc_no,add.nom,add.dob.jour,add.dob.mois,add.dob.annee,add.age,add.adress
e,add.citizenship,add.phone,add.acc_type,add.amt,add.deposit.jour,add.deposit.mois,add.de
posit.annee);
    printf("\n\nDepot avec success!");
}
else
{
    printf("Entrer le montant que vous voulez retirer :DH ");
    scanf("%f",&transaction.amt);
    add.amt-=transaction.amt;
    fprintf(newrec,"%d %s %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",add.acc_no,add.nom,add.dob.jour,add.dob.mois,add.dob.annee,add.age,add.adress
e,add.citizenship,add.phone,add.acc_type,add.amt,add.deposit.jour,add.deposit.mois,add.de
posit.annee);
    printf("\n\nReatrait avec succes!");
}

}
else
{
    fprintf(newrec,"%d %s %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",add.acc_no,add.nom,add.dob.jour,add.dob.mois,add.dob.annee,add.age,add.adress
e,add.citizenship,add.phone,add.acc_type,add.amt,add.deposit.jour,add.deposit.mois,add.de
posit.annee);
}
}
fclose(old);
fclose(newrec);
remove("record.dat");
rename("new.dat","record.dat");
if(test!=1)
{
    printf("\n\nCompte non trouvé !!");
    transact_Invalid:
    printf("\n\nEntrer 0 pour reesayer,1 pour revenir au menu principale et 2 pour
Quitter:");
    scanf("%d",&main_exit);
    system("cls||clear");
    if (main_exit==0)
        transact();
    else if (main_exit==1)
        menu();
    else if (main_exit==2)
        close(1);
    else
    {
        printf("\nInvalide!");
        goto transact_Invalid;
    }
}
else
{
    printf("\nEntrer 1 pour afficher le menu principal et 0 pour quitter:");
    scanf("%d",&main_exit);
    system("cls||clear");
    if (main_exit==1)
        menu();
    else

```

```

        close(1);
    }

}

void supprimer(void)
{
    FILE *old,*newrec;
    int test=0;
    old=fopen("record.dat","r");
    newrec=fopen("new.dat","w");
    printf("Entrer le numero du compte bancaire que vous voulez supprimer :");
    scanf("%d",&rem.acc_no);
    while (fscanf(old,"%d %s %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d",&add.acc_no,&add.nom,&add.dob.jour,&add.dob.mois,&add.dob.annee,&add.age,&add.adr
esse,&add.citizenship,&add.phone,&add.acc_type,&add.amt,&add.deposit.jour,&add.deposit.mois
,&add.deposit.annee)!=EOF)
    {
        if(add.acc_no!=rem.acc_no)
            fprintf(newrec,"%d %s %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d\n",add.acc_no,&add.nom,&add.dob.jour,&add.dob.mois,&add.dob.annee,&add.age,&add.adr
esse,&add.citizenship,&add.phone,&add.acc_type,&add.amt,&add.deposit.jour,&add.deposit.mois,&add.de
posit.annee);

        else
            {test++;
            printf("\nCompte supprimé avec succes !\n");
            }
    }
    fclose(old);
    fclose(newrec);
    remove("record.dat");
    rename("new.dat","record.dat");
    if(test==0)
    {
        printf("\nCompte non trouvé!!\a\a\a");
        supprimer_Invalide:
        printf("\nEntrer 0 pour reesayer,1 pour revenir au menu principale et 2 pour
Quitter:");
        scanf("%d",&main_exit);

        if (main_exit==1)
            menu();
        else if (main_exit==2)
            close(0);
        else if(main_exit==0)
            supprimer();
        else
            {printf("\nInvalide!\a");
            goto supprimer_Invalide;}
    }
    else
    {printf("\nEntrer 1 pour afficher le menu principal et 0 pour quitter:");
    scanf("%d",&main_exit);
    system("cls||clear");
    if (main_exit==1)
        menu();
    else
        close(0);
    }
}

void chercher(void)
{
    FILE *ptr;
    int test=0,rate;
    int choix;

```

```

float time;
float intrst;
ptr=fopen("record.dat","r");
printf("Vous voulez faire la recherche par\n1.numero du compte\n2.nom\nEntrer votre
choix:");
scanf("%d",&choix);
if (choix==1)
{   printf("Entrer le numero du compte:");
    scanf("%d",&check.acc_no);

    while (fscanf(ptr,"%d %s %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d",&add.acc_no,&add.nom,&add.dob.jour,&add.dob.mois,&add.dob.annee,&add.age,&add.adr
esse,&add.citizenship,&add.phone,&add.acc_type,&add.amt,&add.deposit.jour,&add.deposit.mois
,&add.deposit.annee)!=EOF)
    {
        if(add.acc_no==check.acc_no)
        {   system("cls||clear");
            test=1;

            printf("\nNumero du compte: %d\nnom: %s \nDOB: %d/%d/%d \nAge: %d \nAdresse: %s
\nCNI: %s \nNumero de telephone: %.0lf \nType Of Account: %s \nMontant depose: $ %.2f \nDate
de
Depot: %d/%d/%d\n\n", add.acc_no, add.nom, add.dob.jour, add.dob.mois, add.dob.annee, add.age, ad
d.adresse, add.citizenship, add.phone,
            add.acc_type, add.amt, add.deposit.jour, add.deposit.mois, add.deposit.annee);
        }
    }
}
else if (choix==2)
{   printf("Enter the nom:");
    scanf("%s",&check.nom);
    while (fscanf(ptr,"%d %s %d/%d/%d %d %s %s %lf %s %f
%d/%d/%d",&add.acc_no,&add.nom,&add.dob.jour,&add.dob.mois,&add.dob.annee,&add.age,&add.adr
esse,&add.citizenship,&add.phone,&add.acc_type,&add.amt,&add.deposit.jour,&add.deposit.mois
,&add.deposit.annee)!=EOF)
    {
        if(strcmpi(add.nom,check.nom)==0)
        {   system("cls||clear");
            test=1;
            printf("\nNumero de compte: %d\nnom: %s \nDOB: %d/%d/%d \nAge: %d \nAdresse: %s
\nCNI: %s \nNumero de telephone: %.0lf \nType de compte: %s \nMontant depose: $ %.2f \nDate de
Depot: %d/%d/%d\n\n", add.acc_no, add.nom, add.dob.jour, add.dob.mois, add.dob.annee, add.age, add
.adresse, add.citizenship, add.phone,
            add.acc_type, add.amt, add.deposit.jour, add.deposit.mois, add.deposit.annee);
        }
    }
}

fclose(ptr);
if(test!=1)
{   system("cls||clear");
    printf("\ncompte no trouvé!!\a\a\a");
    chercher_Invalide:
    printf("\nEntrer 0 pour reesayer, 1 pour revenir au menu principale et 2 pour
quitter:");
    scanf("%d",&main_exit);
    system("cls||clear");
    if (main_exit==1)
        menu();
    else if (main_exit==2)
        close(1);
    else if (main_exit==0)
        chercher();
    else
    {
        system("cls||clear");
    }
}

```

```

        printf("\nInvalide!\a");
        goto chercher_Invalide;}
    }
else
{printf("\nEntrer 1 pou aller au menu principale et 0 pour quitter:");
scanf("%d",&main_exit);}
if (main_exit==1)
{
    system("cls||clear");
    menu();
}

else
{
    system("cls||clear");
    close(0);
}

}
void menu(void)
{
    int choix;
    system("cls||clear");
    system("color 2");
    printf("\n\n\t\t\tSYSTEME DE GESTION BANCAIRE");
    printf("\n\n\n\t***** BIENVENU DANS LE MENU PRINCIPALE *****");
    printf("\n\n\t\t1.Creer un nouveau compte\n\t\t2.Voir les informations d'un
compte\n\t\t3.Transactions\n\t\t4.Supprimer un compte\n\t\t5.Quitter\n\n\t\t Entrer votre
choix :");
    scanf("%d",&choix);

    system("cls||clear");
    switch(choix)
    {
        case 1:nouveaucompte();
        break;
        case 2:chercher();
        break;
        case 3:transact();
        break;
        case 4:supprimer();
        break;
        case 5:close(0);
        break;
    }
}
int main()
{
    menu();
    return 0;
}

```