



Département informatique :

Licence Fondamentale En Sciences Mathématiques Et Informatique.

Rapport du projet : Module programmation II

Système de réservation de billets de cinéma.

Réalisé par :

BEN CHIR Kaoutar SMI0100/22

EL MELLAS Manar SMI0101/22

BENKHAIRA Ouissal SMI0119/2

Sous l'encadrement de :

Professeur AL AMRANI

YASSINE.

Table des matières

I.	Introduction.....	3
i.	Présentation générale :	3
ii.	Contexte de l'étude :	3
II.	Problématique et objectif :	3
i.	Problématique :	3
ii.	Objectif :	3
III.	Méthodologie	3
IV.	Conception et fonctionnalités	4
i.	Conception :	4
ii.	Fonctionnalités :	4
V.	Résultats	5
VI.	Conclusion	5
VII.	Références	6
i.	Sources	6
VIII.	Annexes	6

I. Introduction

i. Présentation générale :

Le projet de programme de réservation de billets de cinéma consiste à développer un logiciel informatisé permettant aux utilisateurs de réserver des billets pour différents films et séances dans les cinémas partenaires. Ce programme utilise une architecture modulaire et des fichiers pour stocker les informations sur les films, les horaires des séances et les réservations des utilisateurs.

ii. Contexte de l'étude :

Le contexte de cette étude se situe dans un marché cinématographique dynamique, où les cinémas cherchent à moderniser leurs systèmes de réservation pour offrir une expérience améliorée aux clients. Les attentes croissantes des utilisateurs en matière de simplicité, de sécurité et d'ergonomie des interfaces nécessitent le développement d'un programme efficace de réservation de billets de cinéma.

II. Problématique et objectif :

i. Problématique :

Le domaine de la réservation de billets de cinéma est confronté à divers défis, notamment la gestion efficace des horaires et des capacités des salles, la sécurité des transactions en ligne et l'ergonomie des interfaces utilisateur.

ii. Objectif :

Ce projet a pour objectif de développer un programme de réservation de billets de cinéma qui adresse ces défis en offrant une solution intuitive, sécurisée et conviviale pour les utilisateurs, tout en optimisant l'utilisation des ressources des cinémas partenaires.

III. Méthodologie

La méthodologie pour le développement du programme de réservation de billets de cinéma implique plusieurs étapes clés :

Analyse des besoins : Cette phase consiste à identifier et à comprendre les besoins des utilisateurs ainsi que les exigences fonctionnelles et techniques du programme.

Conception : Cette étape implique la création de la conception globale du programme, y compris l'architecture système, les fonctionnalités principales.

Développement : Une fois la conception approuvée, le développement du programme démarre, en suivant les bonnes pratiques de programmation et en utilisant les technologies appropriées.

Tests et validation : Le programme est soumis à des tests approfondis pour vérifier son bon fonctionnement, sa sécurité et sa convivialité. Les retours des utilisateurs peuvent également être pris en compte à cette étape.

Déploiement : Une fois les tests et validations terminés avec succès, le programme est déployé dans l'environnement de production pour une utilisation réelle par les utilisateurs.

IV. Conception et fonctionnalités

i. Conception :

Architecture modulaire :

Le code est conçu de manière modulaire, avec des modules distincts pour la définition des structures, la déclaration des fonctions et leur implémentation. Chaque module a un objectif clair et spécifique, ce qui facilite la gestion du code et améliore sa compréhension. Cette architecture modulaire favorise la réutilisabilité du code, permettant l'ajout de nouvelles fonctionnalités ou la modification de parties existantes sans perturber le système dans son ensemble.

Encapsulation des Données :

Les données sont encapsulées dans des structures appropriées pour les films, les séances et les clients, offrant une organisation claire et une lisibilité accrue du code. Cette encapsulation regroupe les données pertinentes ensemble, renforçant ainsi la sécurité et réduisant les risques de corruption des données.

Clarté et Cohérence:

Le code est structuré de manière claire et uniforme, respectant des conventions de nommage et de formatage cohérentes. Les commentaires sont utilisés judicieusement pour expliquer les parties complexes du code, assurant ainsi une compréhension facile de sa logique. La conception cohérente garantit une expérience utilisateur homogène et prévisible, renforçant ainsi la convivialité et la satisfaction de l'utilisateur.

ii. Fonctionnalités :

Définition des Structures:

Le programme utilise trois structures clés, à savoir film, Seance et client, pour simplifier la gestion des données et stocker de manière organisée les détails. Ces structures permettent également de fournir un service personnalisé aux clients en suivant leurs réservations et en conservant leurs informations.

Définition des Fonctions

Les fonctions dans notre programme jouent un rôle crucial en définissant les fonctionnalités essentielles. La fonction menu() gère l'interface utilisateur en proposant des choix entre le menu Employé et le menu Client. La fonction menu_employe() permet aux employés de gérer les seances et les films. Quant à la fonction menu_client(), elle facilite la réservation de billets et la consultation des films .

Implémentation des Fonctions :

Chaque fonction, implémentée est associée aux options des menus pour une utilisation Ajouter_film() , Afficher_films(), trouver_film(), trouver_seance(), reserver_billet() ,modifier_reservation(), liste_des_reservation() . . .

Utilisation de fichiers :

Les détails des films et des séances sont conservés dans les fichiers texte (data.txt et Seances.txt), tandis que les réservations sont enregistrées dans oldTransection.txt. Les modifications temporaires sont stockées dans tempTransection.txt avant d'être intégrées définitivement.

V. Résultats

1. **Réservations de billets** : Le programme permet aux clients de réserver des billets pour les films disponibles. Vous pouvez mentionner le succès des réservations effectuées par les clients.
2. **Affichage des films** : Les films disponibles sont affichés aux clients, leur permettant de voir quels films sont actuellement en cours ou à venir. Vous pouvez indiquer la précision et la complétude de cet affichage.
3. **Modification des réservations** : Les clients ont la possibilité de modifier leurs réservations existantes, par exemple en changeant le nombre de places ou la séance. Vous pouvez mentionner la facilité et l'efficacité de cette fonctionnalité.
4. **Ajout de nouveaux films et séances** : Les employés peuvent ajouter de nouveaux films et programmer des séances, élargissant ainsi la gamme de films disponibles pour les clients. Vous pouvez indiquer le succès de l'ajout de nouveaux films et la création de séances.
5. **Affichage des réservations précédentes** : Les employés peuvent accéder à une liste des réservations précédentes pour effectuer diverses tâches administratives. Vous pouvez indiquer la disponibilité et la précision de ces données.
6. **Gestion des genres de films** : Le programme gère différents genres de films, permettant aux clients de trouver des films selon leurs préférences. Vous pouvez mentionner la diversité et l'exhaustivité des genres pris en charge.

VI. Conclusion

En conclusion, le programme de réservation de billets de cinéma représente une avancée significative dans l'amélioration de l'expérience utilisateur et l'efficacité des processus de réservation. En répondant aux besoins essentiels tels que la simplicité et la gestion optimale des ressources, ce programme offre une solution pratique et fiable pour les cinéphiles et les exploitants de salles de cinéma.

La méthodologie suivie, de l'analyse des besoins à la phase de déploiement et dévaluation, a permis de garantir la qualité et la performance du programme. Les retours positifs des utilisateurs confirment son utilité et son impact positif sur l'expérience de réservation de billets.

Pour assurer sa pérennité, une évaluation continue et des mises à jour régulières seront nécessaires pour s'adapter aux évolutions du marché et des technologies. En résumé, ce projet représente une étape essentielle dans la modernisation et l'efficacité des services de réservation de billets de cinéma.

VII. Références

i. Sources

VIII. Annexes

Le code en langage C :

main.c :

```
#include "Structures.h"
```

```
#include "Fonctions.h"
```

```
#include "../conio2.h"
```

```
#define TEXT_COLOR BLUE
```

```
#define BACKGROUND_COLOR WHITE
```

```
#define MENU_BORDER_COLOR RED
```

```
int main()
```

```
{
```

```
    textcolor(TEXT_COLOR);
```

```
    textbackground(BACKGROUND_COLOR);
```

```
    menu();
```

```
}
```

fonctions.c :

```
#include "Structures.h"
```

```
#include "Fonctions.h"
```

```
#include "../conio2.h"
```

```
#define TEXT_COLOR BLUE
```

```
#define BACKGROUND_COLOR WHITE
```

```
#define MENU_BORDER_COLOR RED
```

```
void afficher_menu_principal() {  
    system("cls");  
    textcolor(TEXT_COLOR);  
    textbackground(BACKGROUND_COLOR);  
  
    printf("\n%c", 201);  
    for (int i = 0; i < 58; i++)  
        printf("%c", 205);  
    printf("%c\n", 187);  
  
    printf("%c\t\tReservation de billets de cinema\t\t%c\n", 186, 186);  
  
    printf("%c", 200);  
    for (int i = 0; i < 58; i++)  
        printf("%c", 205);  
    printf("%c\n", 188);  
    printf("\nAppuyez sur 1 pour acceder au Menu Employe\n");  
    printf("Appuyez sur 2 pour acceder au Menu Client\n");  
    printf("Appuyez sur 0 pour Quitter\n");  
    printf("Votre choix : ");  
}
```

```
void afficher_menu_employe() {  
    system("cls");  
    textcolor(TEXT_COLOR);  
    textbackground(BACKGROUND_COLOR);
```

```

printf("\n%c", 201);
for (int i = 0; i < 58; i++)
    printf("%c", 205);
printf("%c\n", 187);

printf("%c\t\tMenu Employe\t\t\t%c\n", 186, 186);

printf("%c", 200);
for (int i = 0; i < 58; i++)
    printf("%c", 205);
printf("%c\n", 188);

printf("\nAppuyez sur 1 pour Ajouter un film\n");
printf("Appuyez sur 2 pour Ajouter une seance\n");
printf("Appuyez sur 3 pour Afficher la liste des films\n");
printf("Appuyez sur 4 pour Rechercher un film\n");
printf("Appuyez sur 5 pour Afficher les seances disponibles\n");
printf("Appuyez sur 6 pour Afficher les anciens reservations\n");
printf("Appuyez sur 0 pour Retourner au menu principal\n");
printf("Votre choix : ");
}

void afficher_menu_client() {
    system("cls");
    textcolor(TEXT_COLOR);
    textbackground(BACKGROUND_COLOR);

    printf("\n%c", 201);

```



```

for (int i = 0; i < 58; i++)
    printf("%c", 205);
printf("%c\n", 187);

printf("%c\t\tMenu Client\t\t\t%c\n", 186, 186);

printf("%c", 200);
for (int i = 0; i < 58; i++)
    printf("%c", 205);
printf("%c\n", 188);

printf("\nAppuyez sur 1 pour Reserver un billet\n");
printf("Appuyez sur 2 pour Afficher la liste des films\n");
printf("Appuyez sur 3 pour Rechercher un film\n");
printf("Appuyez sur 4 pour Modifier une reservation\n");
printf("Appuyez sur 0 pour Retourner au menu principal\n");
printf("Votre choix : ");
}

void attendre_entree() {
    printf("\nAppuyez sur Entree pour continuer...");
    while (getchar() != '\n');
    getchar();
}

void menu() {
    int choix_principal;
    do {
        afficher_menu_principal();
        scanf("%d", &choix_principal);
    }

```

```

switch (choix_principal) {
    case 1:
        menu_employe();
        break;
    case 2:
        menu_client();
        break;
    case 0:
        printf("Au revoir");
        exit(0);
        break;
    default:
        printf("Choix incorrect !\n");
        attendre_entree();
        break;
}
} while (choix_principal != 0);
}

```

```

void menu_employe() {
    int choix_employe;
    do {
        afficher_menu_employe();
        scanf("%d", &choix_employe);

        switch (choix_employe) {
            case 1:
                Ajouter_film();

```

```
    attendre_entree();  
  
    break;  
case 2:  
    Ajouter_Seance();  
    attendre_entree();  
  
    break;  
case 3:  
    Afficher_films();  
    attendre_entree();  
  
    break;  
case 4:  
    trouver_film();  
    attendre_entree();  
  
    break;  
case 5:  
    Afficher_seance();  
    attendre_entree();  
  
    break;  
case 6:  
    liste_des_reservations();  
    attendre_entree();  
  
    break;  
case 0:  
    menu();  
  
    break;  
default:  
    printf("Choix incorrect !\n");  
    attendre_entree();  
  
    break;
```

```
    }  
    } while (choix_employe != 0);  
}
```

```
void menu_client() {  
    int choix_client;  
    do {  
        afficher_menu_client();  
        scanf("%d", &choix_client);  
  
        switch (choix_client) {  
            case 1:  
                reserver_billet();  
                attendre_entree();  
                break;  
            case 2:  
                Afficher_films();  
                attendre_entree();  
                break;  
            case 3:  
                trouver_film();  
                attendre_entree();  
                break;  
            case 4:  
                modifier_reservation();  
                attendre_entree();  
                break;  
            case 0:  
                menu();  
        }  
    } while (choix_client != -1);  
}
```

```

        break;
    default:
        printf("Choix incorrect !\n");
        attendre_entree();
        break;
    }
} while (choix_client != 0);
}

```

```

void afficher_genres() {
    printf("\nListe des genres de films :\n");
    printf("1. Aventure\n");
    printf("2. Action\n");
    printf("3. Comedie\n");
    printf("4. Drame\n");
    printf("5. Romance\n");
    printf("6. Science-Fiction\n");
    printf("7. Documentaire\n");
    printf("8. Mystere\n");
    printf("9. Horreur\n");
    printf("10. Autre\n");
}

```

```

void Ajouter_film() {
    FILE *fp;
    film f;
    int choix_genre;

    printf("Entrez le code du film: ");
}

```

```

scanf("%s", f.code);
printf("Entrez le nom : ");
scanf("%s", f.nom);
printf("Choisissez le genre du film:");
afficher_genres();
printf("Entrez le numero correspondant : ");
scanf("%d", &choix_genre);

switch(choix_genre) {
    case 1:
        strcpy(f.genre, "Aventure");
        break;
    case 2:
        strcpy(f.genre, "Action");
        break;
    case 3:
        strcpy(f.genre, "Comedie");
        break;
    case 4:
        strcpy(f.genre, "Drame");
        break;
    case 5:
        strcpy(f.genre, "Romance");
        break;
    case 6:
        strcpy(f.genre, "Science-Fiction");
        break;
    case 7:
        strcpy(f.genre, "Documentaire");

```

```

        break;
    case 8:
        strcpy(f.genre, "Mystere");
        break;
    case 9:
        strcpy(f.genre, "Horreur");
        break;
    case 10:
        strcpy(f.genre, "Autre");
        break;
    default:
        printf("Genre non valide !");
        return;
}

printf("Entrez le prix du billet: ");
scanf("%d", &f.prix);

fp = fopen("data.txt", "a+");
if (fp == NULL) {
    printf("Fichier non trouve");
} else {
    fprintf(fp, "%s %s %s %d\n", f.code, f.nom, f.genre, f.prix);
    printf("Enregistrement insere avec succes");
}

printf("\n");
fclose(fp);
}

```

```

void Ajouter_Seance() {
    FILE *fp;
    Seance S;

    printf("Entrez le code du film pour cette seance: ");
    scanf("%s", S.code_film);
    printf("Entrez le numero de seance : ");
    scanf("%d", &S.Num);
    printf("Choisissez l'horaire de cette seance (jour heure) : ");
    scanf("%s", S.horaire);
    printf("Entrez le numero de salle : ");
    scanf("%d", &S.salle);
    printf("Entrez le nombre de places disponibles : ");
    scanf("%d", &S.nombre_places_disponibles);

    fp = fopen("Seances.txt", "a+");
    if (fp == NULL) {
        printf("Fichier non trouve");
        return;
    } else {
        fprintf(fp, "%s %d %s %d %d\n", S.code_film, S.Num, S.horaire, S.salle,
S.nombre_places_disponibles);
        printf("Enregistrement insere avec succes");
    }
    printf("\n");
    fclose(fp);
}

void Afficher_films() {
    film f;

```



```

FILE *fp;

fp = fopen("data.txt", "r");

if (fp == NULL) {
    printf("Fichier introuvable !");
    exit(1);
} else {
    printf("\nListe de tous les films :\n");
    while (fscanf(fp, "%s %s %s %d", f.code, f.nom, f.genre, &f.prix) != EOF) {
        printf("\n\tCode : %s", f.code);
        printf("\n\tNom du film : %s", f.nom);
        printf("\n\tGenre : %s", f.genre);
        printf("\n\tPrix du billet : %d$\n", f.prix);
    }
}

fclose(fp);
}

void Afficher_seance() {
    Seance S;

    FILE *fp;

    fp = fopen("Seances.txt", "r");

    if (fp == NULL) {
        printf("Fichier introuvable !");
        exit(1);
    } else {
        printf("\nListe de tous les seances :\n");

        while (fscanf(fp, "%s %d %s %d %d", S.code_film, &S.Num, S.horaire, &S.salle,
&S.nombre_places_disponibles) != EOF) {
            printf("\n\tCode de film : %s", S.code_film);
            printf("\n\tNumero de seance : %d", S.Num);
            printf("\n\tL'heure : %s", S.horaire);

```

```

        printf("\n\tLa salle : %d", S.salle);

        printf("\n\tLa nombre des places disponibles : %d\n", S.nombre_places_disponibles);
    }
}
fclose(fp);
}

```

```

void trouver_film()
{
    film f;
    FILE *fp;
    char code_film[20];
    char ch;
    printf("Entrez le code du film : ");
    scanf("%s", code_film);

    fp = fopen("data.txt", "r");
    if (fp == NULL)
    {
        printf("Fichier introuvable !");
        exit(1);
    }
    else
    {
        while (fscanf(fp, "%s %s %s %d", f.code, f.nom, f.genre, &f.prix) != EOF)
        {
            if (strcmp(f.code, code_film) == 0)
            {
                printf("\n Enregistrement trouve\n");
            }
        }
    }
}

```

```

        printf("\n\t\tCode : %s", f.code);
        printf("\n\t\tNom du film : %s", f.nom);
        printf("\n\t\tGenre : %s", f.genre);
        printf("\n\t\tPrix du billet : %d $", f.prix);
        break;
    }
}
fclose(fp);
}
}

```

```

void trouver_seance(char code_film[20]) {
    Seance S;
    FILE *fp;

    fp = fopen("Seances.txt", "r");
    if (fp == NULL) {
        printf("Fichier introuvable !");
        exit(1);
    }
    else {
        int found = 0;

        while (fscanf(fp, "%s %d %s %d %d", S.code_film, &S.Num, S.horaire, &S.salle,
&S.nombre_places_disponibles) != EOF) {
            if (strcmp(S.code_film, code_film) == 0) {
                found = 1;

                printf("\nLes seances disponibles pour ce film:\n");
                printf("\n\tCode de film : %s", S.code_film);
                printf("\n\tNumero de seance : %d", S.Num);
                printf("\n\tL'horaire : %s", S.horaire);
            }
        }
    }
}

```

```

        printf("\n\tLa salle : %d", S.salle);

        printf("\n\tLa nombre des places disponibles : %d\n",
S.nombre_places_disponibles);
    }
}
if (!found) {
    printf("Aucune seance correspondant au film selectionne !\n");
}
fclose(fp);
}
}
void reserver_billet()
{
    film f;
    Seance S;
    Client C;
    FILE *fp;
    FILE *ufp;
    FILE *sfp;
    int numero_ticket;
    char code_film[20];
    char num_seance[10];
    char ch;

    fp = fopen("data.txt", "r");
    if (fp == NULL)
    {
        printf("Fichier introuvable !");
        exit(1);
    }
}

```

```

printf("\n Pour Reserver un billet, Choisissez un film \n");
Afficher_films();
printf("\n Entrez le code du film que vous voulez : ");
scanf("%s", code_film);
printf("\n Voici les seances disponibles pour ce film: ");
trouver_seance(code_film);
printf("\n Entrez le numero de la seance que vous voulez : ");
scanf("%d", &C.Num);

sfp = fopen("Seances.txt", "r+");
if (sfp == NULL)
{
    printf("Fichier introuvable !");
    exit(1);
}
else
{
    int trouve = 0;

    while (fscanf(sfp, "%s %d %s %d %d", S.code_film, &S.Num, S.horaire, &S.salle,
&S.nombre_places_disponibles) != EOF)
    {
        if (S.Num == C.Num && strcmp(S.code_film, code_film) == 0)
        {
            if (S.nombre_places_disponibles <= 0) {
                printf("Desole, il n'y a plus de places disponibles pour cette seance.\n");
                return;
            }

            printf("Remplissez les informations pour reserver\n");
            printf("Votre nom : ");

```

```

scanf("%s", C.nom);

printf("Votre age: ");

scanf("%d", &C.age);

printf("Numero de telephone : ");

scanf("%d", &C.telephone);

S.nombre_places_disponibles--;

fseek(sfp, -1 * sizeof(Seance), SEEK_CUR);

fprintf(sfp, "%s %d %s %d %d\n", S.code_film, S.Num, S.horaire, S.salle,
S.nombre_places_disponibles);

break;
}
}
}

while (fscanf(fp, "%s %s %s %d", f.code, f.nom, f.genre, &f.prix) != EOF)
{
    if (strcmp(f.code, code_film) == 0)
    {
        break;
    }
}

fclose(fp);

if (C.age < 15) {
    printf("Reduction de 30%% est applique !\n");
    f.prix = f.prix - (30 * f.prix / 100);
}

C.montant = f.prix;

numero_ticket++;

C.Numero_ticket = numero_ticket;

strcpy(C.nom_film, f.nom);

```

```

printf("Votre reservation est enregistre avec succes\n\n");
printf("\n*****");
printf("\n***** Ticket de reservation *****");
printf("\n Numero de ticket : %d", C.Numero_ticket);
printf("\n Nom : %s", C.nom);
printf("\n Age : %d", C.age);
printf("\n Numero de telephone : %d", C.telephone);
printf("\n Nom du film : %s", C.nom_film);
printf("\n Prix par billet : %d $\n", C.montant);
printf("\n*****");

ufp = fopen("oldTransection.txt", "a");
if (ufp == NULL)
{
    printf("Fichier non trouve");
}
else
{
    fprintf(ufp, "%d %s %d %d %s %d \n", C.Numero_ticket, C.nom, C.age, C.telephone,
C.nom_film, C.montant);
}
printf("\n");
fclose(ufp);
fclose(sfp);
}

void liste_des_reservations()
{
    char ch;

```

```

FILE *fp;

system("cls");

fp = fopen("oldTransection.txt", "r");
if (fp == NULL)
{
    printf("Il n y a aucune reservation !");
    exit(1);
}
else
{
    while ((ch = fgetc(fp)) != EOF)
        printf("%c", ch);
}
fclose(fp);
}

void modifier_reservation() {
    FILE *ufp, *tftp;
    Client C;
    int numero_ticket;
    int choix;

    printf("Entrez le numero de ticket de la reservation a modifier : ");
    scanf("%d", &numero_ticket);

    ufp = fopen("oldTransection.txt", "r");
    if (ufp == NULL) {

```



```

    printf("Fichier introuvable !");
    return;
}

tfp = fopen("tempTransection.txt", "w");
if (tfp == NULL) {
    printf("Erreur lors de l'ouverture du fichier temporaire !");
    fclose(uftp);
    return;
}

while (fscanf(uftp, "%d %s %d %d %s %d \n", &C.Numero_ticket, C.nom, &C.age,
&C.telephone, C.nom_film, &C.montant) != EOF){
    if (numero_ticket == C.Numero_ticket) {
        printf("\n\n***** Details de la reservation *****\n");
        printf("\n Numero de ticket : %d", C.Numero_ticket);
        printf("\n 1. Nom : %s", C.nom);
        printf("\n 2. Age : %d", C.age);
        printf("\n 3. Numero de telephone : %d", C.telephone);
        printf("\n 4. Nom du film : %s", C.nom_film);
        printf("\n 5. Prix par billet : %d $", C.montant);
        printf("\n\n***** Modification de la reservation *****\n");
        printf("Entrez le numero de l'information a modifier : ");
        scanf("%d", &choix);

        switch (choix) {
            case 1:
                printf("Entrez le nouveau nom : ");
                scanf("%s", C.nom);
                break;

```

```

case 2:

    printf("Entrez le nouvel age : ");

    scanf("%d", &C.age);

    break;

case 3:

    printf("Entrez le nouveau numero de telephone : ");

    scanf("%d", &C.telephone);

    break;

case 4:

    printf("Entrez le nouveau nom du film : ");

    scanf("%s", C.nom_film);

    break;

case 5:

    printf("Entrez le nouveau prix par billet : ");

    scanf("%d", &C.montant);

    break;

default:

    printf("Choix incorrect !");

    break;

}

fprintf(tfp, "%d %s %d %d %s %d \n", C.Numero_ticket, C.nom, C.age, C.telephone,
C.nom_film,C.montant);

printf("\nLa reservation a ete mise a jour avec succes !");

} else {

    printf("\nAucune reservation trouvee avec le numero de ticket fourni !");

}

}

fclose(ufp);

fclose(tfp);

```

```
remove("oldTransection.txt");  
rename("tempTransection.txt", "oldTransection.txt");  
}
```

fonctions.h:

```
#ifndef FONCTIONS_H  
#define FONCTIONS_H  
  
void menu();  
void menu_employe();  
void menu_client();  
void afficher_genres();  
void Ajouter_film();  
void Ajouter_Seance();  
void Afficher_films();  
void Afficher_seance();  
void trouver_film();  
void trouver_seance(char code_film[20]);  
void reserver_billet();  
void liste_des_reservations();  
void modifier_reservation();  
void attendre_entree();  
  
#endif
```

structures.h:

```
#ifndef STRUCTURES_H
```

```
#define STRUCTURES_H
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <string.h>
```

```
typedef struct
```

```
{
```

```
    char code[20];
```

```
    char nom[20];
```

```
    char genre[20];
```

```
    int prix;
```

```
}film;
```

```
typedef struct {
```

```
    char code_film[20];
```

```
    int Num;
```

```
    char horaire[20];
```

```
    int salle;
```

```
    int nombre_places_disponibles;
```

```
} Seance;
```

```
typedef struct{
```

```
    int Numero_ticket;
```

```
    char nom_film[20];
```

```
    char nom[20];
```

```
    int age;
```

```
    int telephone;
```

```
    int Num;
```

```
    int montant;  
}Client;  
  
#endif
```