ONLINE MOVIE BOOKING MANAGEMENT SYSTEM

PROBLEM STATEMENT:

E-Movie ticket booking providing the customer the book ,cancel ,view the seats, also person can access his booking through a reference id.

RELATED WORK:

->GOALS:

1) Main Goal is to book tickets of customer choice.

2) Our model should be able to book tickets for three days, six shows a day (two movies), and it should take the input from the customer which movie to book, which show to choose and on what day movie, show should be booked.

3) It should store the name of the user who has booked the ticket, and create a random number as ID for that particular booking, and when that ID is given

program should be able to access the details of booking i.e day, show, movie, seat number, name, contact details, etc.

4) Customer must be allowed to cancel the tickets which he had booked and also has freedom to cancel any number of tickets among his ticket bookings providing his reference id.

5) All the customer's data is to be stored using file management system, that is to store the customers details in separate files, thereby creating database enabled project since there is amount data is very large to store.

Description:-

This project is a menu driven system, using structures to store the data of the user ,which is stored in files and data is read from files whenever needed.

Key features:-

1)Enables the customer to view the seat booking pattern and select seats accordingly.

2)Also the customer can select the seats only if it is not booked.

3)Can view his respective seating pattern and his details when he enters his corresponding reference id provided to him,soon after booking.

4)Can view the seating pattern of any show ,so that to provide basic information about availability of seats.

Code related information:-

1. Created a Three Dimensional Character Array to store all the information regarding Movie code and rows and columns of seating arrangement .

2. We created Two structures to store seat information (structure seat) and information of customer (structure data).

3. Structure seat consists of data of row and column, whereas structure data consists of information, i.e name of

customer, phone number, code, number of tickets ,Invoice number and a structure array of structures of seat type to store the booked row and columns. Array is of size 6, i.e it can store information related to rows and columns of 6 tickets , so basically in our program a person can book at most 6 tickets.

4. While booking tickets we ask the costumer’s name, then his phone-number, then movie code, then number of tickets customer want to book and finally row and column of priority seat.

5.Movie codes in our Program refers to

DAY-1 , MOVIE -1 ,SHOW -1	->	1
DAY-1 , MOVIE -1 ,SHOW -2	->	2
DAY-1 , MOVIE -1 ,SHOW -3	->	3
DAY-1, MOVIE -2, SHOW -1	->	4
DAY-1, MOVIE -2, SHOW -2	->	5
DAY-1, MOVIE -2, SHOW -3	->	6
DAY-2, MOVIE -1, SHOW -1	->	7
DAY-2, MOVIE -1, SHOW -2	->	8
DAY-2 , MOVIE -1 ,SHOW -3	->	9
DAY-2 , MOVIE -2 ,SHOW -1	->	10
DAY-2, MOVIE -2,SHOW -2	->	11
DAY-2, MOVIE -2, SHOW -3	->	12

DAY-3, MOVIE -1, SHOW -1 -> 13
DAY-3, MOVIE -1, SHOW -2 -> 14
DAY-3 , MOVIE -1 ,SHOW -3 -> 15
DAY-3 , MOVIE -2 ,SHOW -1 -> 16
DAY-3 , MOVIE -2 ,SHOW -2 -> 17
DAY-3, MOVIE -2, SHOW -3 -> 18

6.We used file management to write the data we received from the customer, this file contains name, phone-number, movie-code, number of tickets, row and column(of all tickets), Invoice number and store it in a file linewise.

7.We Generated invoice number, by (multiplying movie_code*1000)+(row*100)+(column *1).Here row and column are seat numbers of his first seat irrespective of how many seats he selects to book,such that it becomes unique to every customer who had booked the tickets.

Book function:

1)Taking all the input from the customer related to information regarding name,phone number,movie code,number of seats to be booked and rows and columns numbers of seats required accordingly.3

3)Soon after taking input of name,phone number,movie_code,number of seats, the seating pattern of movie hall corresponding to given movie_code.

3)Intially every element of three dimensional character array is intialised to character '-'.But after taking input of rows and columns from the user,the corresponding elements of 3D array pertaining to input from the customer to character '#',i.e the seats after getting booked are converted from character '-' to character '#'.

4)If user tries to book seat which is already booked,then error arises indicating the error and also allowing him to book the ticket once again.

5)As soon as we are getting input from the user,we are storing that data in file so that we can use in any future requirements afterwards.

6)After storing data in 3D array and files ,the final bill and corresponding reference id is generated and printed ,and now even this is stored in file.

View function

1)Takes the input of movie code(which is mentioned earlier in this text) and prints the seating pattern corresponding to the movie code.

2)Such that to provide the user about basic information of availability of seats.

Reference function

1) This function takes the input of 3D character array and reference id , and also asks for the number of seats booked .

2)Opens the file where the entire customer data is stored and runs a loop to search for the required reference id provided to the

customer while booking
and prints his respective details regarding his name,phone number movie,number of seats booked ,row and columns of seat,bill .

3)This function returns a movie code which is used in View_details function .

4)Also this function is used in both View_details and cancel function.

View_details function

1)This is a void 'return type function',takes 3D character array and uses reference function to get movie code.

2)This prints all the details of the customer and prints his details as mentioned in above “reference function”,using the reference code provided.

3)Apart from printing details of customer this function also prints the seating pattern corresponding to movie code obtained using reference function.

Cancel function:

1)It is a void 'return type' of the function and takes the argument of 3D character array.

2)It asks user to enter the reference id and number of seats booked.

3)It reads all the details from the files including row and columns corresponding reference id entered and stored used in access elements in array to cancel.
      i.e converts '#' back to '-'.

3)Then it prints all the details of customer and cancel all his seats as mentioned above.

In main() function.........

*Declaring the 3D character array and intialising all elements to character '-'.

*Creating a menu driven system using while loop,providing options such as :

   >Book

>Cancel

>View seating pattern

>View_details

>Exit

And running the above menu driven system.

Code-:

```
#include<stdio.h>
#include<string.h>
struct seat                    /*structure which stores
information of a seat i.e row and column*/
{
int row;
int column;
};
struct data                    /*structure which stores
information of name,phone number,movie
code,num of tickets customer booked,invoice
num,bill*/
{
char name[15];
int phone_num;
int code;
int num_tickets;
int invoice_num;
int amount;
struct seat seat_arr[6];     /*this is a struture
array,which helps in storing information of more
than one seat,i.e if a person wants to book more
than 2 seats,this helps in storing information of
more than one seat's row and column*/
};

struct data booking_data;

int book(char arr[18][15][7])       /*this is a
function which books data,it takes an address of
the main array as arguement*/
{
int num,code,r,c,i,j,kk;
FILE *fp;
fp=fopen("movie.txt","a");          /*the file
movie.txt is used to store all the information of
the customer for future use.*/
printf("                                    ENTER YOUR
NAME AND MOBILE NUMBER TO
REGISTER  :) \n ");
printf("Name:");
fscanf(stdin,"%s",booking_data.name);
printf("\n");
printf("Mobile no:");
fscanf(stdin,"%d",&booking_data.phone_num);
fprintf(fp,"%s\t
%d\t",booking_data.name,booking_data.phone_n
um);
printf("enter your movie code\n");
printf(" DAY\tMOVIE \t\tSHOW \t\tcode\n");
printf(" DAY-1\tMOVIE -1\tSHOW -1\t\t1\n");
printf(" DAY-1\tMOVIE -1\tSHOW -2\t\t2\n");
printf(" DAY-1\tMOVIE -1\tSHOW -3\t\t3\n");
printf(" DAY-1\tMOVIE -2\tSHOW -1\t\t4\n");
printf(" DAY-1\tMOVIE -2\tSHOW
-2\t\t5\n");     //printing codes for corresponding
day,movie,show.
printf(" DAY-1\tMOVIE -2\tSHOW -3\t\t6\n");
printf(" DAY-2\tMOVIE -1\tSHOW -1\t\t7\n");
printf(" DAY-2\tMOVIE -1\tSHOW -2\t\t8\n");
printf(" DAY-2\tMOVIE -1\tSHOW -3\t\t9\n");
printf(" DAY-2\tMOVIE -2\tSHOW -1\t\t10\n");
printf(" DAY-2\tMOVIE -2\tSHOW -2\t\t11\n");
printf(" DAY-2\tMOVIE -2\tSHOW -3\t\t12\n");
printf(" DAY-3\tMOVIE -1\tSHOW -1\t\t13\n");
printf(" DAY-3\tMOVIE -1\tSHOW -2\t\t14\n");
printf(" DAY-3\tMOVIE -1\tSHOW -3\t\t15\n");
printf(" DAY-3\tMOVIE -2\tSHOW -1\t\t16\n");
printf(" DAY-3\tMOVIE -2\tSHOW -2\t\t17\n");
printf(" DAY-3\tMOVIE -2\tSHOW -3\t\t18\n");
```

```
fscanf(stdin,"%d",&booking_data.code);
fprintf(fp,"%d\t",booking_data.code);
printf("enter number of tickets you want to book
(max 6 only)\n");   /*dialouge for telling that max
no of tickets that can be booked is 6*/
fscanf(stdin,"%d",&num);
fprintf(fp,"%d\t",num);

printf("************************************
************** LAYOUT OF MOVIE HALL
*******************************************
*****\n\n");
for(kk=0;kk<7;kk++)
{
printf("%d\t",kk+1);
}
printf("\n");
for(i=0;i<15;i++)
  {
   if(i==5 || i==10)
    {
      printf("\n\n");
    }
   for (j = 0;j < 7;j++)
      {
        printf("%c\t",arr[booking_data.code-1][i][j]);

      }
```

```
     printf("   %c",65+i);
    printf("\n");
  }


for(i=0;i<num;i++)
{
printf("enter row of seat\n");
fscanf(stdin,"%d",&booking_data.seat_arr[i].row)
;
fprintf(fp,"%d\t",booking_data.seat_arr[i].row);
printf("enter column of seat\n");
fscanf(stdin,"%d",&booking_data.seat_arr[i].colu
mn);
fprintf(fp,"%d\t",booking_data.seat_arr[i].column
);

if(arr[booking_data.code]
[booking_data.seat_arr[i].row]
[booking_data.seat_arr[i].column]=='-')   /*this if-
else statement is used to check whether the seat
was already booked before,if it is booke before,an
error message saying seat is already booked is
printed,if no error it books the seat accordingly*/
{
arr[booking_data.code-1]
[booking_data.seat_arr[i].row-1]
[booking_data.seat_arr[i].column-1]='#';
}
  else{printf("seat is already booked! \n");
  i-=1;
}
}

int totalamount=0;                                     /*this is
a loop for caluculating the cost of tickets,since
tickets are divided into three classes,there would
be different costs for all the three classes*/
for(i=0;i<num;i++)
{
      if(booking_data.seat_arr[i].row<=4)
      {
            totalamount=totalamount+ 75;
      }
      else if(booking_data.seat_arr[i].row>4 &&
booking_data.seat_arr[i].row<=9)
      {
       totalamount=totalamount+100;
      }
      else if(booking_data.seat_arr[i].row>9)
      {
       totalamount=totalamount+150;
      }
}
booking_data.amount=totalamount;
fprintf(fp,"%d\t",booking_data.amount);
```

```
/*this writes the caluculated ticket cost into the
file,for future references*/
booking_data.invoice_num=booking_data.code*1
000+booking_data.seat_arr[0].row*100+booking_
data.seat_arr[0].column;
fprintf(fp,"%d\n",booking_data.invoice_num);
/*this writes the caluculated invoice number into
the file*/
printf("your booking is done\nREFERENCE ID:
%d\nBILL:
%d\n\n",booking_data.invoice_num,booking_data
.amount);
fclose(fp);
}

int cancel(char arr[18][15][7])          /*this
fucntion is to cancel the tickets,when user selects
cancel option*/
{
FILE *fp;
fp=fopen("movie.txt","r");
int i,n,ref;
printf("Enter your reference id:\n");
scanf("%d",&ref);
printf("Enter number of seats you have
booked:\n");
scanf("%d",&n);
while(!feof(fp))
```

```
{
     fscanf(fp,"%s %d %d
%d",booking_data.name,&booking_data.phone_n
um,&booking_data.code,&booking_data.num_tic
kets);
      for(i=0;i<n;i++)  /*here,in this loop what we
are doing each and every line of code in the
file,where we previously stored information..we
get invice number frm the file by advancing from
the staring point of the file for every customer*/
   {
     fscanf(fp,"%d
%d",&booking_data.seat_arr[i].row,&booking_da
ta.seat_arr[i].column);
   }

fscanf(fp,"%d
%d",&booking_data.amount,&booking_data.invoi
ce_num);
if(booking_data.invoice_num==ref) /*from the
above loop we get invoice number,now we check
whether the invoice number provided by the
customer is equal to that of our iinvoice number
from the file,if thats the case then again from
starting we store all the row and column numbers
by writing anther loop,and one by one we mark
them to booked status,if invoice number provided
by customer is not equal to any of the invoice
numbers of the file,then no change would take
place showing that the number entered is invalid
invoice number*/
{
     for(i=0;i<n;i++)
    {
       fscanf(fp,"%d
%d",&booking_data.seat_arr[i].row,&booking_da
ta.seat_arr[i].column);
       arr[booking_data.code-1]
[booking_data.seat_arr[i].row-1]
[booking_data.seat_arr[i].column-1]='-';
    }
    fclose(fp);
    printf("Your booking is cancelled\n\n");
    break;
}
}
}


int reference(int r,int n)  /*this is like a
subfunction to view-details function, this takes in
invoice number as arguement,it then searches the
file for invoice number,if it finds a invoice
number matching with arguement give,it then
prints details of the customer with help of that
invoice number,i.e it prints
```

```
name,mobileno,day,movie,show,and also all the
rows and columns of tickets the customer has
booked.In the last it prints the bill.The function
returns booking_data code to view-details
function .*/
{
int i;
FILE *fp;
fp=fopen("movie.txt","r");
while(!feof(fp))
{
     fscanf(fp,"%s %d %d
%d",booking_data.name,&booking_data.phone_n
um,&booking_data.code,&booking_data.num_tic
kets);
       printf("%d\n",n);
      for(i=0;i<n;i++)
   {
      fscanf(fp,"%d
%d",&booking_data.seat_arr[i].row,&booking_da
ta.seat_arr[i].column);
   }
fscanf(fp,"%d
%d",&booking_data.amount,&booking_data.invoi
ce_num);

if(booking_data.invoice_num==r)
{
```

```
printf("Name: %s\n",booking_data.name);
printf("Mobile no:
%d\n",booking_data.phone_num);

if(booking_data.code==1)
{
printf("Day: Day 1\n");
printf("Movie: Movie 1\n");
printf("Show: Morning show\n");
}
if(booking_data.code==2)
{
printf("Day: Day 1\n");
printf("Movie: Movie 1\n");
printf("Show: Matinee show\n");
}
if(booking_data.code==3)
{
printf("Day: Day 1\n");
printf("Movie: Movie 1\n");
printf("Show: Evening show\n");
}
if(booking_data.code==4)
{
printf("Day: Day 1\n");
printf("Movie: Movie 2\n");
printf("Show: Morning show\n");
}
```

```
if(booking_data.code==5)
{
printf("Day: Day 1\n");
printf("Movie: Movie 2\n");
printf("Show: Matinee show\n");
}
if(booking_data.code==6)
{
printf("Day: Day 1\n");
printf("Movie: Movie 2\n");
printf("Show: Evening show\n");
}
if(booking_data.code==7)
{
printf("Day: Day 2\n");
printf("Movie: Movie 1\n");
printf("Show: Morning show\n");
}
if(booking_data.code==8)
{
printf("Day: Day 2\n");
printf("Movie: Movie 1\n");
printf("Show: Matinee show\n");
}
if(booking_data.code==9)
{
printf("Day: Day 2\n");
printf("Movie: Movie 1\n");
printf("Show: Evening show\n");
}
if(booking_data.code==10)
{
printf("Day: Day 2\n");
printf("Movie: Movie 2\n");
printf("Show: Morning show\n");
}
if(booking_data.code==11)
{
printf("Day: Day 2\n");
printf("Movie: Movie 2\n");
printf("Show: Matinee show\n");
}
if(booking_data.code==12)
{
printf("Day: Day 2\n");
printf("Movie: Movie 2\n");
printf("Show: Evening show\n");
}
if(booking_data.code==13)
{
printf("Day: Day 3\n");
printf("Movie: Movie 1\n");
printf("Show: Morning show\n");
}
if(booking_data.code==14)
{
```

```
printf("Day: Day 3\n");
printf("Movie: Movie 1\n");
printf("Show: Matinee show\n");
}
if(booking_data.code==15)
{
printf("Day: Day 3\n");
printf("Movie: Movie 1\n");
printf("Show: Evening show\n");
}
if(booking_data.code==16)
{
printf("Day: Day 3\n");
printf("Movie: Movie 2\n");
printf("Show: Morning show\n");
}
if(booking_data.code==17)
{
printf("Day: Day3\n");
printf("Movie: Movie 2\n");
printf("Show: Matinee show\n");
}
if(booking_data.code==18)
{
printf("Day: Day 3\n");
printf("Movie: Movie 2\n");
printf("Show: Evening show\n");
}
```

```
printf("Seat positions:\n");

        for(i=0;i<n;i++)
    {
        printf("row=%c\tcolumn=
%d\n",64+booking_data.seat_arr[i].row,booking_
data.seat_arr[i].column);
    }

printf("Your Total Bill=%d\nYour reference code=
%d\n",booking_data.amount,booking_data.invoic
e_num);


fclose(fp);
return booking_data.code;
break;
}
}
}
void view(char arr[18][15][7])    /*this function
takes input of movie code and prints the tickets
for that movie*/
{
int kk,i,j,Movie_code;
printf(" DAY\tMOVIE \t\tSHOW \t\tcode\n");
printf(" DAY-1\tMOVIE -1\tSHOW -1\t\t1\n");
printf(" DAY-1\tMOVIE -1\tSHOW -2\t\t2\n");
printf(" DAY-1\tMOVIE -1\tSHOW -3\t\t3\n");
printf(" DAY-1\tMOVIE -2\tSHOW -1\t\t4\n");
printf(" DAY-1\tMOVIE -2\tSHOW
-2\t\t5\n");     //printing codes for corresponding
day,movie,show.
printf(" DAY-1\tMOVIE -2\tSHOW -3\t\t6\n");
printf(" DAY-2\tMOVIE -1\tSHOW -1\t\t7\n");
printf(" DAY-2\tMOVIE -1\tSHOW -2\t\t8\n");
printf(" DAY-2\tMOVIE -1\tSHOW -3\t\t9\n");
printf(" DAY-2\tMOVIE -2\tSHOW -1\t\t10\n");
printf(" DAY-2\tMOVIE -2\tSHOW -2\t\t11\n");
printf(" DAY-2\tMOVIE -2\tSHOW -3\t\t12\n");
printf(" DAY-3\tMOVIE -1\tSHOW -1\t\t13\n");
printf(" DAY-3\tMOVIE -1\tSHOW -2\t\t14\n");
printf(" DAY-3\tMOVIE -1\tSHOW -3\t\t15\n");
printf(" DAY-3\tMOVIE -2\tSHOW -1\t\t16\n");
printf(" DAY-3\tMOVIE -2\tSHOW -2\t\t17\n");
printf(" DAY-3\tMOVIE -2\tSHOW -3\t\t18\n");
printf("Enter the movie code:\n");
scanf("%d",&Movie_code);

printf("**********************************
************** LAYOUT OF MOVIE HALL
****************************************
*****\n\n");
for(kk=0;kk<7;kk++)
{
 printf("%d\t",kk+1);
}
printf("\n");

for(i=0;i<15;i++)
  {
   if(i==5 || i==10)
    {
       printf("\n\n");
    }
   for (j = 0;j < 7;j++)
      {
        printf("%c\t",arr[Movie_code-1][i][j]);
      }
       printf("   %c",65+i);
      printf("\n");
  }
}
void View_Details(char arr[18][15][7])
/*this function takes input reference id,no of
tickets he has booked as input he then prints out
layout of movie-hall,it has a fucntion called
reference,which takes reference id given as
arguement(this function reads data from the file
using reference id and prints all the data from the
file).*/
{
int i,j,ref,movie_code,n;
printf("Enter your reference id:\n");
```

```
scanf("%d",&ref);
printf("Enter Number of seats you have
booked:\n");
scanf("%d",&n);
movie_code=reference(ref,n);
printf("%d\n",movie_code);


printf("*********************************
************** LAYOUT OF MOVIE HALL
***************************************
*****\n\n");
for(i=0;i<15;i++)
  {
   if(i==0 || i==5 || i==10)
    {
      if (i/5 == 0)
      {
        printf("                    ****SILVER****
                \n");
      }
      if  (i/5 == 1)
      {
        printf("                    ****GOLD****
                     \n");
      }
      if  (i/5 == 2)
      {
        printf("
      ****EXECUTIVE****                 \n");
       }
       printf("\n");
    }
   for (j = 0;j < 7;j++)
      {
        printf("%c\t",arr[movie_code-1][i][j]);
      }
        printf("   %c",65+i);
      printf("\n");

  }}


int main()    /*this is the main function which
prints a menu,which has the corresponding
functions and until the user opts for the exit option
from the menu,user has the chance to
cancel,book,view-details of tickets in our ticket
management system!*/
{
void View_Details(char arr[18][15][7]);

struct data booking_data;
int i,j,code,r,c;
char arr[18][15][7];
for(code=0;code<18;code++){
```

```
for(r=0;r<15;r++){
for(c=0;c<7;c++){
arr[code][r][c]='-';
}}}
int check,mm=1;
printf("************************************
*********  WELCOME TO ONLINE MOVIE
BOOKING PORTAL
******************************************
**********\n");
while(mm==1)
{
printf("ENTER 1 to book \nEnter 2 to
View_Details\nEnter 3 to cancel\nEnter 4 to
view\nEnter 0 to exit\n");

scanf("%d",&check);
if(check==0)
{
printf("************************************
**********THANK
YOU**************************************
********\n");
mm=2;
}
if(check==1)
{
book(arr);
}
if(check==2)
{
View_Details(arr);
}
if(check==3)
{
cancel(arr);
}
if (check == 4)
{
view(arr);
}
}
FILE *fp;
fp= fopen("movie.txt","r");
}
```

Future advancements:

1)Booking all seats at a time as we do in real life(if person needs all the seats at a place).

2)Deleting persons details from the text document when cancel the seats.
3)Retaining the status of seats when the terminal is closed and can book or cancel seats corresponding to the old status again.

4)cancelling and viewing details without taking number of tickets booked as input.

5)cancelling only few tickets by taking input from user.

Bibilography:

There is no specific sources from which it is extracted used most of which is taught in class.

No other library is used exept <stdio.h>
such that to make it simplified and compact.

*But File management in c has its contibuted sources from :-
=> C programming By Reema thareja.
=> ANSI C By E Balaguruswamy
=>www.tutorialspoint.com

Done by-:

Srujan Swaroop Gajula(IMT2016033)

Nizam Manideep(IMT2016026)
Yelugoti Mohana Datta(IMT2016012)
Siddarth Reddy Desu(IMT2016037)
Sapparapu Rahul(IMT2016037)