Library management system :

The purpose of Library Management System is to automate the existing manual system by the help of computerised equipment and full-fledged computer software, fulfilling their requirements, so that their valuable data information can be stored for a Longer period with easy accessing and manipulation of the same. The required software
and hardware are easily available and easy to work with. Library Management System, as described above can lead to error free, secure
reliable and fast management system. It can assist the user to concentrate on their other activities rather than concentrating on the record keeping. Thus it will help organisations better utilise resources. The organisation can maintain computerised records without redundant entries. That means that one need not be distracted by information that is not relevant, while being able to reach the information.

CONTENTS

CHAPTER 1

INTRODUCTION

A Library Management System is a project that manages and stores the information of the books. It allows the admin or library manager to keep a constant track of all the books present in the library. It also allows admin to keep a continuous check on the books issued and returned books. This program reduces the manual work and allows smooth functioning of the library Activities. Moreover this system is designed for the particular need of the company to carry out operations in a smooth and effective manner. The application is reduced as much as possible to avoid errors while entering the data It also provides error messages While entering invalid data. No formal knowledge is needed for the user to use this system Thus by this all it proves it is user-friendly Library Management System as described above, can lead to error free secure reliable and fast management system It can assist the user to concentrate on their other activities rather than concentrating on the record keeping.

CHAPTER 2

OBJECTIVE

- Manage the information of books
- Manage the information of students to which books are issued
- Manages the records of issued books
- Editing, adding and updating of book records

CHAPTER 3

SYSTEM REQUIREMENTS

A System Requirements Specification that describes the features and behaviour of a system or software application.

3.1. SOFTWARE REQUIREMENTS:

The software requirements are descriptions of features and functionalities of the target system.

Operating system : Windows 98,Windows xp ,Windows 7,Linux

Language : c , Java 2 Runtime Environment

3.2. HARDWARE REQUIREMENTS:

Hardware requirements often specify the operating system version, processor type, memory size, available disk space and additional peripherals.

1.RAM : 128 MB

2.HARD DISK : 20GB

3.MONITOR : 15” Colour Monitor

CHAPTER 4

System Design

System design of library management system will provide the design phase for the library management system. The main aim of the design phase is to provide the solution for the specified requirements Functional decomposition can be defined as the process of dividing the functional relationship into different parts. The descriptions of the components are as follows

- **Issue Books :** Books can be issued to the user and can be added as a record in the database
- **Add Books :** This component allows you to add the new book details
- **view Book details :** this component allows you to view the recorded book details
- **Update Books** : The book details can be updated or edited through this component.
- **Delete book :** The book details can be deleted through this component

CHAPTER 5

SYSTEM IMPLEMENTATION

The project is implemented using C language as follows:

```c
#include<stdio.h>
#include<stdlib.h>
#include<string.h>

void mainmenu(void);
void addbooks(void);
void deletebooks(void);
void updatebooks(void);
void viewbooks(void);
void issuebooks(void);
typedef struct book       //structure declaration to record book details
{
    int id;
    char name[100];
    char category[100];
    char Author[100];
    float Price;
    int rackno;
}book;
typedef struct issue        //structure declaration to record student details
{
    char student[100];
    long int contact;
    int d,rd;
    int m,rm;
    int y,ry;
}issue;
```

fig 5.1

```c
void mainmenu(void)
{
    system("cls");
    do
    {
        int num;
        printf("\n\n\n\t|||||||||||||||||||||||||||||||||||||||| LIBRARY MANAGEMENT SYSTEM ||||||||||||||||||||||||||||||||||||||||\n\n\n");
        printf("\t\t\t\t 1. Add Books\n\n");
        printf("\t\t\t\t 2. Delete Books\n\n");
        printf("\t\t\t\t 3. update Book's Record\n\n");
        printf("\t\t\t\t 4. View Books List\n\n");
        printf("\t\t\t\t 5. Isuue books\n\n");
        printf("\t\t\t\t 6. Close Application\n\n\n");
        printf("\t\t\tEnter your choice:",num);
        scanf("%d",&num);
        printf("--------------------------------------------------------------------------------------------------------------------------");
        switch(num)
        {
            case 1:
                addbooks();
            break;
            case 2:
                deletebooks();
            break;
            case 3:
                updatebooks();
            break;
```

Fig 5.2

```
55                  case 4:
56                      viewbooks();
57                  break;
58                  case 5:
59                      issuebooks();
60                  break;
61                  case 6:
62                      exit(0);
63              }
64          }while(1);
65      }
66      void addbooks(void)          //funtion that add books
67      {
68          system("cls");
69          book *b;
70          FILE *fp;
71          int i,n;
72          printf("\n\n\t\t::::::::::::::::::::::::::ADDBOOKS:::::::::::::::::::::::\n\n");
73          printf("How many books you want to add:");
74          scanf("%d",&n);
75          printf("\n\n");
76          b = (book*)calloc(n,sizeof(book));
77          fp = fopen("records.txt","w");
78          for(i=0;i<n;i++)
79          {
```

Fig 5.3

```
80              printf("id:");
81              fflush(stdin);
82              scanf("%d",&b[i].id);
83              printf("Category:");
84              fflush(stdin);
85              scanf("%[^\n]s",b[i].category);
86              printf("Name:");
87              fflush(stdin);
88              scanf("%[^\n]s",b[i].name);
89              printf("Author:");
90              fflush(stdin);
91              scanf("%[^\n]s",b[i].Author);
92              printf("price:Rs. ");
93              scanf("%f",&b[i].Price);
94              printf("Rack no.:");
95              scanf("%d",&b[i].rackno);
96              printf("\n");
97              fwrite(&b[i],sizeof(book),1,fp);
98          }
99          fclose(fp);
100         printf("The Records are Sucessfully Saved\n");
101         printf("--------------------------------------------------------------------------------------------------------------------------------");
102     }
```

fig 5.4

```
103  void deletebooks(void)      //funtion that delete books
104  {
105      system("cls");
106      book bl;
107      FILE *fp, *fp1;
108      int found=0,id;
109      fp = fopen("records.txt","r");
110      fp1  = fopen("temp.txt","w");
111      printf("Enter the ID to delete: ");
112      scanf("%d",&id);
113      while(fread(&bl,sizeof(book),1,fp))
114      {
115          if(bl.id == id)
116          {
117              found=1;
118          }
119          else
120              fwrite(&bl,sizeof(book),1,fp1);
121      }
122      fclose(fp);
123      fclose(fp1);
124      if(found)
125      {
126          fp1 = fopen("temp.txt","r");
127          fp = fopen("records.txt","w");
128          while(fread(&bl,sizeof(book),1,fp1))
129          {
```

Fig 5.5

```
130              fwrite(&bl,sizeof(book),1,fp);
131          }
132          fclose(fp);
133          fclose(fp1);
134          printf("The record is deleted\n");
135          printf("-------------------------------------------------------------------------------------------------------------------------");
136      }
137      else
138          printf("The record not found\n");
139          printf("-------------------------------------------------------------------------------------------------------------------------");
140  }
141  void updatebooks(void)      //funtion that update books
142  {
143      system("cls");
144      book bl;
145      FILE *fp, *fp1;
146      int found=0,id;
147      fp = fopen("records.txt","r");
148      fp1  = fopen("temp.txt","w");
149      printf("Enter the ID to update: ");
150      scanf("%d",&id);
151      while(fread(&bl,sizeof(book),1,fp))
152      {
153          if(bl.id == id)
154          {
```

fig 5.6

```
155                 found=1;
156                 printf("The book is available\n");
157                 printf("New Name:");
158                 fflush(stdin);
159                 scanf("%[^\n]s",bl.name);
160                 printf("New Author:");
161                 fflush(stdin);
162                 scanf("%[^\n]s",bl.Author);
163                 printf("New price: Rs. ");
164                 scanf("%f",&bl.Price);
165                 printf("New Rack no.:");
166                 scanf("%d",&bl.rackno);
167             }
168             fwrite(&bl,sizeof(book),1,fp1);
169         }
170         fclose(fp);
171         fclose(fp1);
172         if(found)
173         {
174             fp1 = fopen("temp.txt","r");
175             fp = fopen("records.txt","w");
176             while(fread(&bl,sizeof(book),1,fp1))
177             {
```

fig 5.7

```
178                 fwrite(&bl,sizeof(book),1,fp);
179             }
180             fclose(fp);
181             fclose(fp1);
182             printf("The book is updated\n");
183             printf("-----------------------------------------------------------------------------------------------------------------------");
184         }
185         else
186             printf("The record not found\n");
187             printf("-----------------------------------------------------------------------------------------------------------------------");
188     }
189     void viewbooks(void)        //funtion that display recorded books details
190     {
191         system("cls");
192         char ch;
193         book bl;
194         FILE *fp;
195         fp = fopen("records.txt","r");
196 
197         printf("::::::::::::::::::::::::::::::::Book List::::::::::::::::::::::::::::::::\n");
198         printf("CATEGORY       ID     BOOK NAME       AUTHOR          PRICE        RackNo   \n");
199         while(fread(&bl,sizeof(book),1,fp))
200         {
201             printf("\n%s\t\t%d\t%s\t\t%s\t\t%.2f\t\t%d\n",bl.category,bl.id,bl.name,bl.Author,bl.Price,bl.rackno);
202         }
203         fclose(fp);
204         printf("-----------------------------------------------------------------------------------------------------------------------");
205     }
```

fig 5.8

```c
void issuebooks(void)       //funtion that issue books
{
    system("cls");
    printf("::::::::::::::::::::::::::::::::::ISSUE BOOKS::::::::::::::::::::::::::::::::::\n");
    book bl;
    issue ibl;

    FILE *fp, *fp2;
    int found=0,id;
    fp = fopen("records.txt","r");
    fp2  = fopen("issue.txt","w");
    printf("Enter the ID: ");
    scanf("%d",&id);
    while(fread(&bl,sizeof(book),1,fp))
    {
        if(bl.id == id)
        {
            found=1;
        }
        else;
    }
    fclose(fp);
    fclose(fp2);
    if(found)
    {
```

fig 5.9

```c
    fp2 = fopen("issue.txt","w");
    printf("The book is available\n");
    printf("Student name:");
    fflush(stdin);
    scanf("%[^\n]s",ibl.student);
    printf("contact details:");
    scanf("%ld",&ibl.contact);
    printf("Date of issue:");
    scanf("%d",&ibl.d);
    printf("Month of issue:");
    scanf("%d",&ibl.m);
    printf("Year of issue:");
    scanf("%d",&ibl.y);
    printf("Date of returning:");
    scanf("%d",&ibl.rd);
    printf("Month of return:");
    scanf("%d",&ibl.rm);
    printf("Year of return:");
    scanf("%d",&ibl.ry);
    fwrite(&ibl,sizeof(issue),1,fp2);
    printf("The book is issued");
    printf("----------------------------------------------------------------------------------------------------------------");
    fclose(fp2);
}
else
```

fig 5.10

```
256         printf("The record not found\n");
257         printf("-----------------------------------------------------------------------------------------------------------------------");
258 }
259 int main()
260 {
261     char user[20];
262     char pass[8];
263     printf("\t-----------------------------------------------------------------------------------------------------------\n");
264     printf("\t:::::::::::::::::::::::::::::::::::::::::: WELCOME ::::::::::::::::::::::::::::::::::::::::::\n");
265     printf("\t-----------------------------------------------------------------------------------------------------------\n\n");
266     printf("\tEnter the user ID:");
267     scanf("%s",&user);
268     printf("\tEnter the Password:");
269     scanf("%s",&pass);
270     if(strcmp(user,"cseproject")==0)
271     {
272         if(strcmp(pass,"userpass")==0)
273         {
274             printf("\nYou are logged in\n");
275             mainmenu();
276         }
277         else
278         printf("\t\tWrong password\n");
279     }
280     else
281         printf("\t\tUser doesn't exist\n\n'\n");
282 }
283
```

fig 5.11

CHAPTER 6

RESULTS

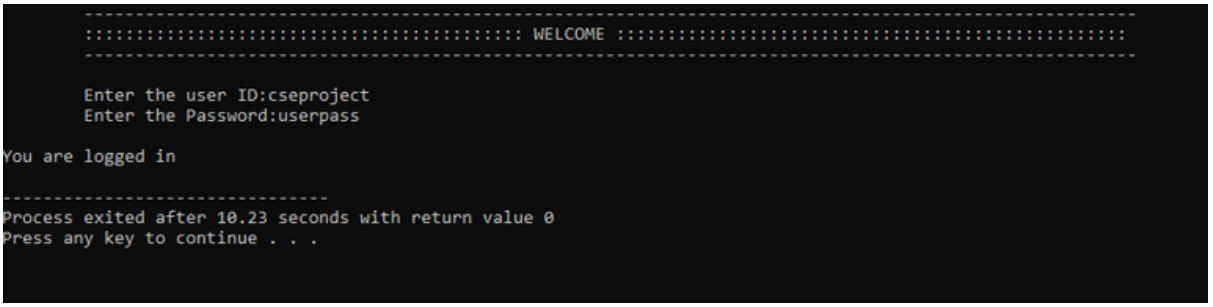

Fig 6.1 :Login page

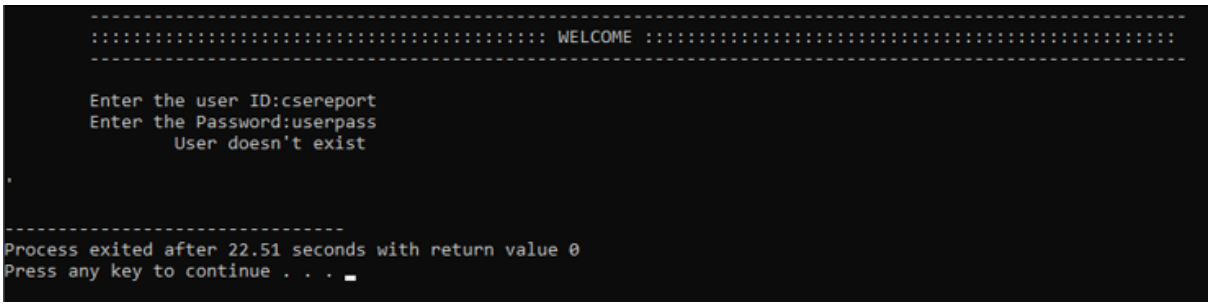

Fig 6.2

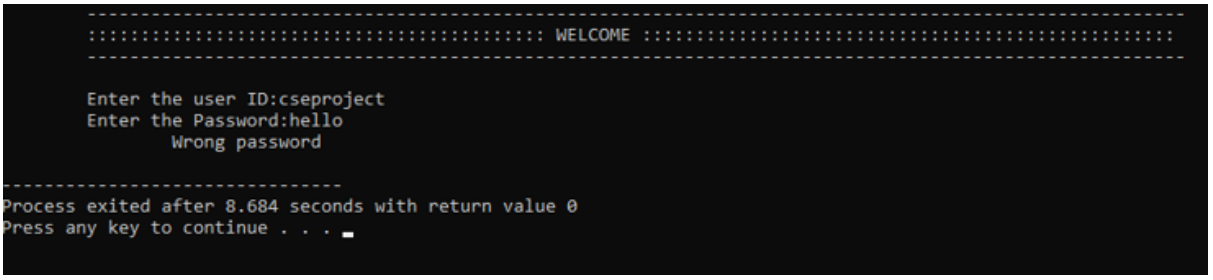

Fig 6.3

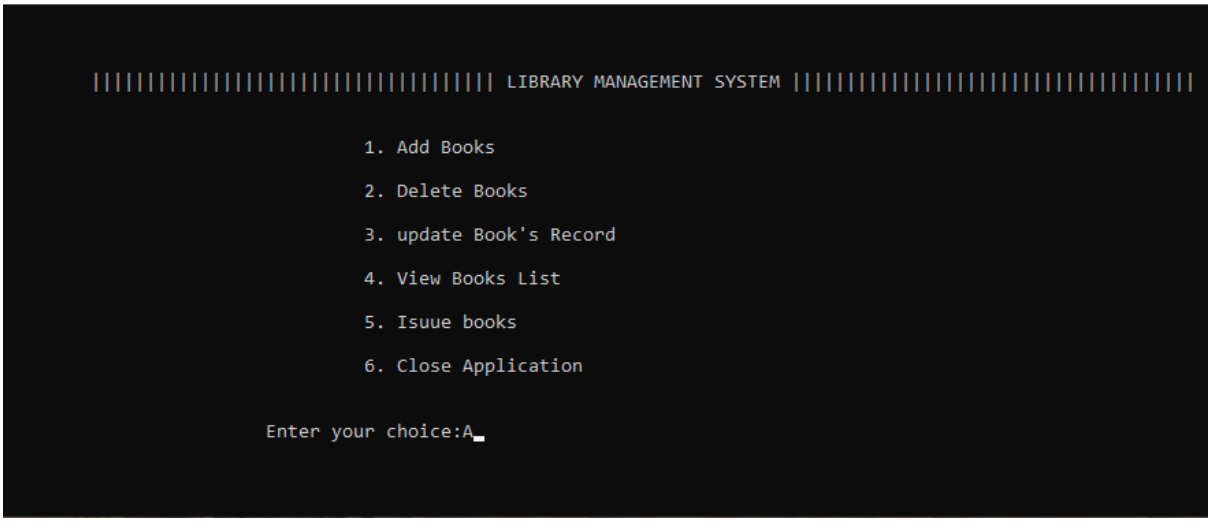

Fig 6.4 : Main menu

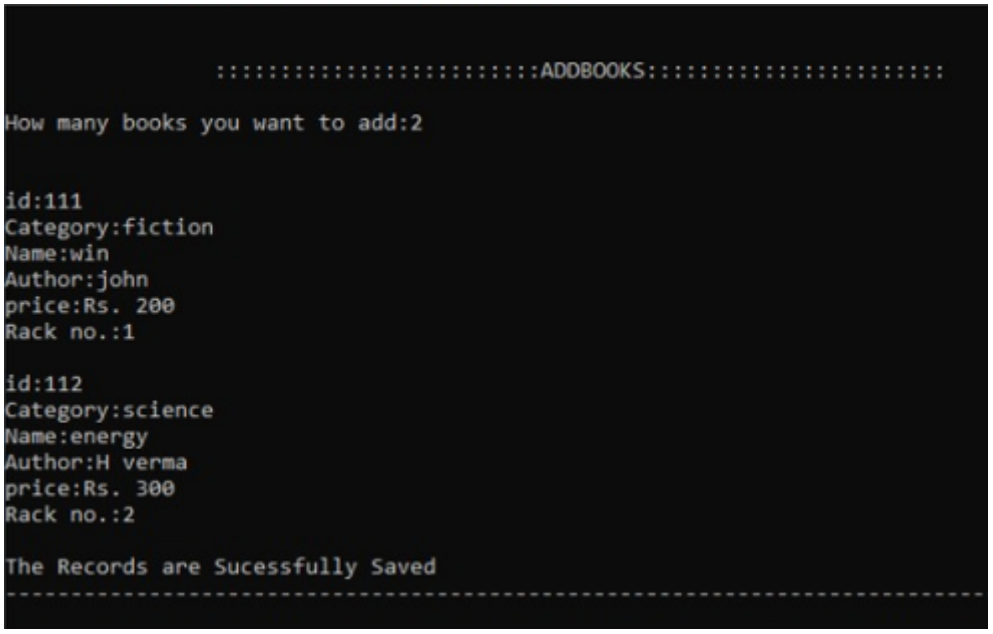

Fig 6.5 :Add Books

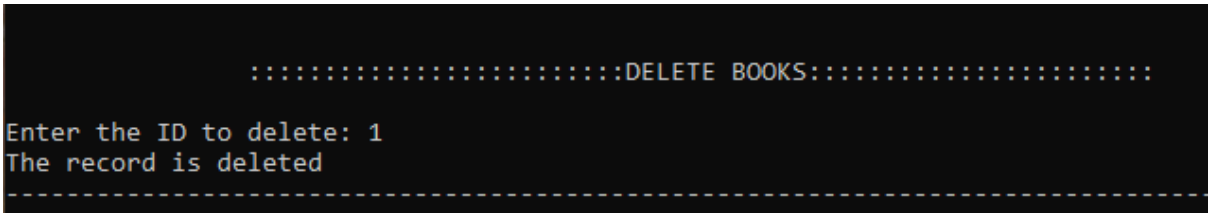

Fig 6.6 : Delete books

Fig 6.7

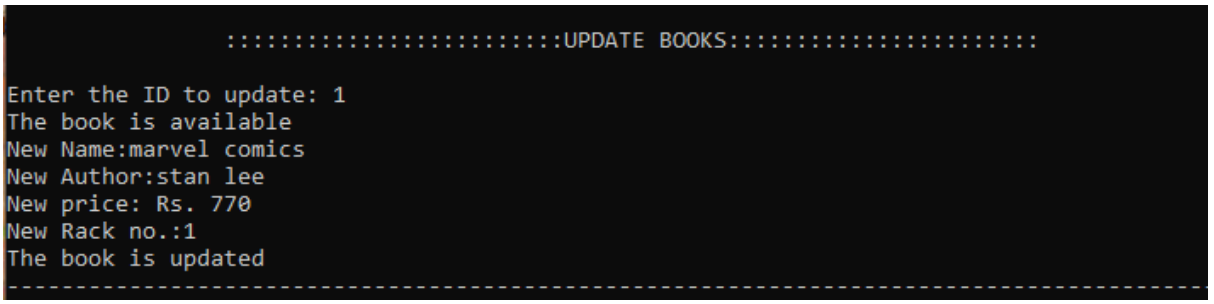

Fig 6.8 : Update Books

Fig 6.9

Fig 6.10 : View Books List

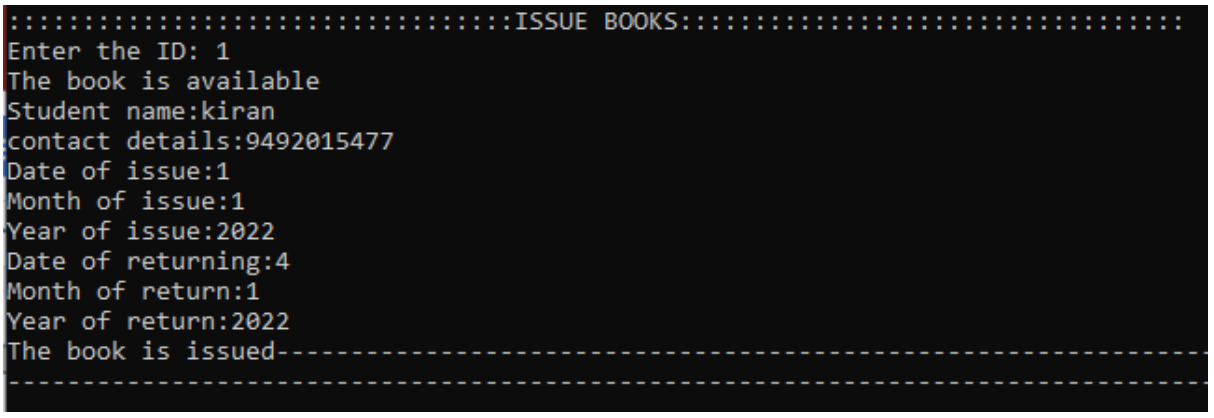

Fig 6.11 : Issue Books