# **VIRTUAL BACKPACK**

*A*

*Mini Project Report*

*Submitted in partial fulfilment of the*

*Requirements for the award of the Degree of*

**BACHELOR OF ENGINEERING**

IN

**INFORMATION TECHNOLOGY**

By

MOHAMMED ABDUL MOQEET – 1602-19-737-021

K. UPENDAR NAIK – 1602-19-737-051



**Department of Information Technology**

**Vasavi College of Engineering (Autonomous)**

**(Affiliated to Osmania University)**

**Ibrahimbagh, Hyderabad-31**

**2020**

**Vasavi College of Engineering (Autonomous)**

**(Affiliated to Osmania University)**

**Hyderabad-500 031**

**Department of Information Technology**



**DECLARATION BY THE CANDIDATE**

We, MOHAMMED ABDUL MOQEET and K. UPENDAR NAIK bearing hall ticket numbers, 1602-19-737-021 and 1602-19-737-051, hereby declare that the project report entitled “VIRTUAL BACKPACK” is submitted in partial fulfilment of the requirement for the award of the degree of Bachelor of Engineering in Information Technology.

This is a record of bonafide work carried out by us and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

MOHAMMED ABDUL MOQEET

1602-19-737-021

K. UPENDAR NAIK

1602-19-737-051

(Faculty In-Charge) (Head.Dept IT)

**ACKNOWLEDGEMENTS**

Our Mini Project would not have been successful without the help of several people. We are extremely thankful to our college, Vasavi College of Engineering, for providing the opportunity to implement our project, “Virtual Backpack”.

We would like to express our gratitude to Ms. Leelavathy, Assistant Professor, Department of Information Technology, Vasavi College of Engineering for their esteemed guidance, moral support and invaluable advice provided by them for the success of this Mini Project.

Sincerely,

**MOHAMMED ABDUL MOQEET 1602-19-737-021**

**K. UPENDAR NAIK 1602-19-737-051**

**ABSTRACT**

In our project ‘Virtual Backpack’, we are providing the user a virtual backpack or their own workplace . In the student’s option it will be like an inventory. our Virtual backpack is designed using C language. The transactions like login, register, add , search, view, edit, delete are provided. The backpack System stores the details like name, address, ID number , Date Of Birth of members or users who is using the app. The details of books like book name, book number, author, the total  number of books that are present in the inventory etc are also stored.

**INTRODUCTION**

**About The Project**

“Virtual Backpack” is a console-based C Project intended to be used by students to manage books details and act as an information source for various books available.

**Problem Domain**

We wanted to make an application that makes managing books and updating information easier for the students. The existing applications are usually limited to the library or the staff and is not very student friendly, we thought of making the project where the student can store the information easily and manage all the data without any problem.

**TECHNOLOGY**

**Software Requirements**

The minimum software requirements to utilize the “Virtual Backpack” are:

* Operating System: Windows XP or higher
* GNU Compiler Collection (GCC)
* Text Editor

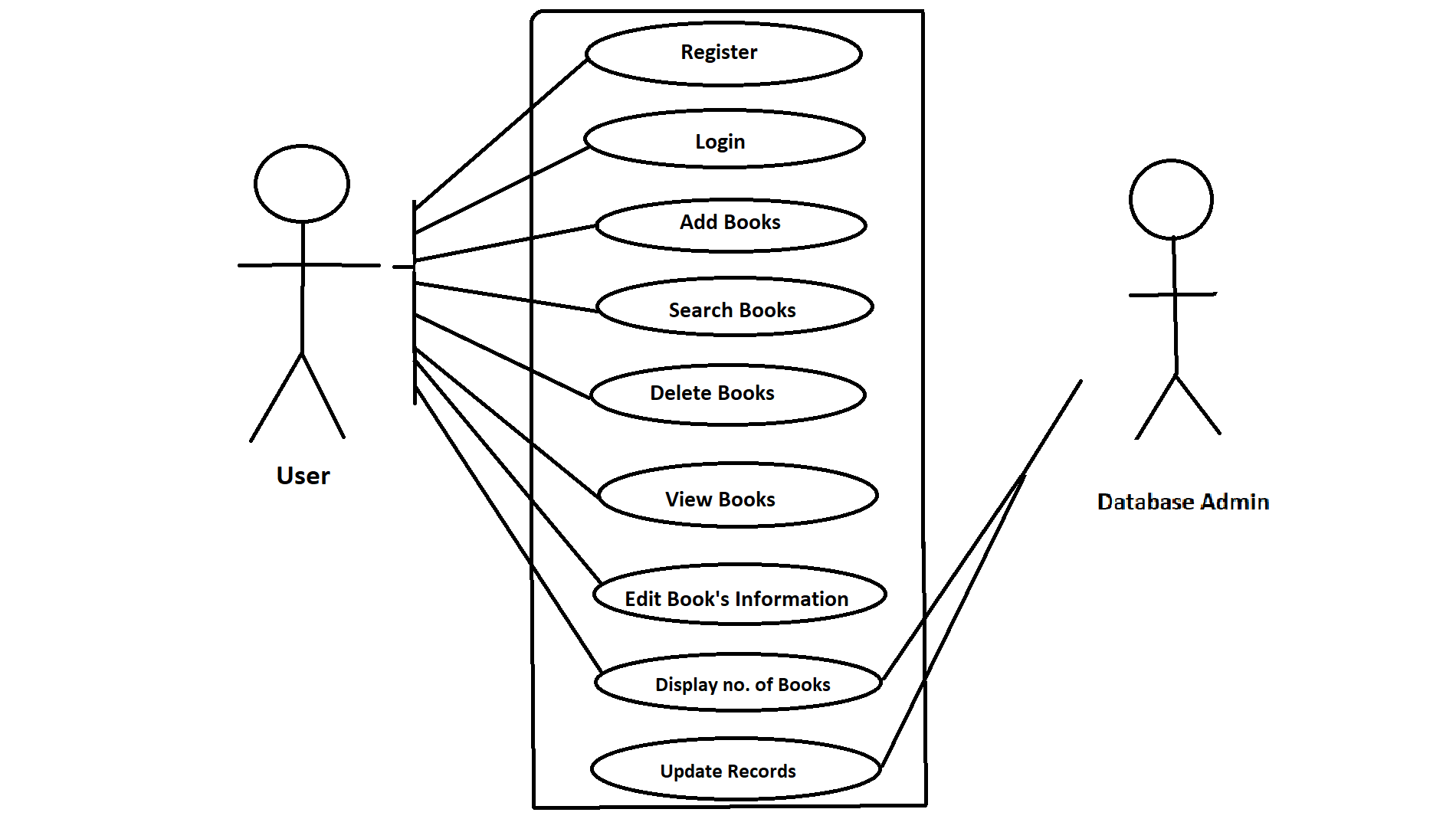
**Hardware Requirements**

The minimum hardware requirements to utilize the “Virtual Backpack” are:

* Minimum 4MB RAM
* Minimum 25MB Free Storage
* Processor: Intel 386 and its equivalent or higher

**PROPOSED WORK**

**Design :**



**Implementation:**

#include<windows.h>

#include<stdio.h>

#include<conio.h>

#include <stdlib.h>

#include<string.h>

#include<ctype.h>

#include<dos.h>

#include<time.h>

//void login();

//void regster();

void menu();

void addbooks();

void delete();

void searchbooks();

void viewbooks();

void editbooks();

void returnfunc();

void Password();

int getdata();

void Password();

int checkid();

FILE \*fp,\*ft,\*fs;

int s;

char findbook;

char password[10]={"userpass"};

struct books

{

int id;

char stname[20];

char name[20];

char Author[20];

int quantity;

float Price;

int count;

};

struct books a;

COORD coord = {0, 0};

void gotoxy (int x, int y)

{

coord.X = x; coord.Y = y; // X and Y coordinates

SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE), coord);

}

int main()

{

Password();

getch();

return 0;

}

void menu ()

{

system("cls");

printf("\n\t\t\t###########################################################################");

printf("\n\t\t\t############ ############");

printf("\n\t\t\t############ VIRTUAL BACKPACK ############");

printf("\n\t\t\t############ ############");

printf("\n\t\t\t###########################################################################");

printf("\n\n.........................................................................................................");

printf("\n\nENTER \n1 to ADD\n2 to DELETE\n3 to SEARCH\n4 to VIEW BOOKS LIST\n5 to EDIT BOOK'S RECORD\n6 to EXIT APPLICATION\n\n:");

switch(getch()){

case '1':

addbooks();

break;

case '2':

delete();

break;

case '3':

searchbooks();

break;

case '4':

viewbooks();

break;

case '5':

editbooks();

break;

case '6':

exit(0);

break;

default:{

printf("\aWrong Entry!!Please re-enter correct option");

if(getch())

menu();}

}

}

void addbooks() //funtion that add books

{

system("cls");

fp=fopen("lib.dat","ab+");

if(getdata()==1)

{

fseek(fp,0,SEEK\_END);

fwrite(&a,sizeof(a),1,fp);

fclose(fp);

gotoxy(21,14);

printf("The Record Is Sucessfully Saved");

gotoxy(21,15);

printf("Do you want to save more?(Y / N):");

if(getch()=='n')

menu();

else

system("cls");

addbooks();

}

}

void delete() //function that delete items from file fp

{

system("cls");

int d;

char another='y';

while(another=='y') //infinite loop

{

system("cls");

gotoxy(10,5);

printf("Enter the Book ID to delete:");

scanf("%d",&d);

fp=fopen("lib.dat","rb+");

rewind(fp);

findbook=0;

while(fread(&a,sizeof(a),1,fp)==1)

{

if(a.id==d)

{

gotoxy(10,7);

printf("The Book Record Is Available");

gotoxy(10,8);

printf("Book name is %s",a.name);

findbook='t';

}

}

if(findbook!='t')

{

gotoxy(10,10);

printf("No record is found modify the search");

if(getch())

menu();

}

if(findbook=='t' )

{

gotoxy(10,9);

printf("Do you want to delete it?(Y/N):");

if(getch()=='y')

{

ft=fopen("test.dat","wb+"); //temporary file for delete

rewind(fp);

while(fread(&a,sizeof(a),1,fp)==1)

{

if(a.id!=d)

{

fseek(ft,0,SEEK\_CUR);

fwrite(&a,sizeof(a),1,ft); //write all in tempory file except that

} //we want to delete

}

fclose(ft);

fclose(fp);

remove("lib.dat");

rename("test.dat","lib.dat"); //copy all item from temporary file to fp except that

fp=fopen("lib.dat","rb+"); //we want to delete

if(findbook=='t')

{

gotoxy(10,10);

printf("The Record Is Sucessfully Deleted");

gotoxy(10,11);

printf("Delete another record?(Y/N)");

}

}

else

menu();

fflush(stdin);

another=getch();

}

}

gotoxy(10,15);

menu();

}

void searchbooks()

{

system("cls");

int d;

printf(":::::::::::::::::::::::::::::Search Books:::::::::::::::::::::::::::::");

gotoxy(20,3);

printf("\xB2\xB2\xB2\xB2 1=> Search By ID");

gotoxy(20,5);

printf("\xB2\xB2\xB2\xB2 2=> Search By Name");

gotoxy( 20,9);

printf("Enter Your Choice");

fp=fopen("lib.dat","rb+"); //open file for reading propose

rewind(fp); //move pointer at the begining of file

switch(getch())

{

case '1':

{

system("cls");

gotoxy(25,4);

printf("::::Search Books By Id::::");

gotoxy(20,6);

printf("Enter the book id:");

scanf("%d",&d);

gotoxy(20,8);

printf("Searching.....");

int t=0;

while(fread(&a,sizeof(a),1,fp)==1)

{

if(a.id==d)

{

gotoxy(20,7);

printf("The Book Is Available");

gotoxy(20,8);

printf("\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2");

gotoxy(20,9);

printf("\xB2 ID:%d",a.id);gotoxy(47,9);printf("\xB2");

gotoxy(20,10);

printf("\xB2 Name:%s",a.name);gotoxy(47,10);printf("\xB2");

gotoxy(20,11);

printf("\xB2 Author:%s ",a.Author);gotoxy(47,11);printf("\xB2");

gotoxy(20,12);

printf("\xB2 Quantity:%d ",a.quantity);gotoxy(47,12);printf("\xB2"); gotoxy(47,11);printf("\xB2");

gotoxy(20,13);

printf("\xB2 Price:Rs.%.2f",a.Price);gotoxy(47,13);printf("\xB2");

gotoxy(20,14);

printf("\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2");

t++;

}

}

if(t==0) //checks whether conditiion enters inside loop or not

{

gotoxy(20,8);

printf("\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2");

gotoxy(20,9);printf("\xB2"); gotoxy(38,9);printf("\xB2");

gotoxy(20,10);

printf("\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2");

gotoxy(22,9);printf("\aNo Record Found");

}

gotoxy(20,17);

printf("Try another search?(Y/N)");

if(getch()=='y')

searchbooks();

else

menu();

break;

}

case '2':

{

char s[15];

system("cls");

gotoxy(25,4);

printf("::::Search Books By Name::::");

gotoxy(20,5);

printf("Enter Book Name:");

scanf("%s",s);

int d=0;

while(fread(&a,sizeof(a),1,fp)==1)

{

if(strcmp(a.name,(s))==0) //checks whether a.name is equal to s or not

{

gotoxy(20,7);

printf("The Book Is Available");

gotoxy(20,8);

printf("\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2");

gotoxy(20,9);

printf("\xB2 ID:%d",a.id);gotoxy(47,9);printf("\xB2");

gotoxy(20,10);

printf("\xB2 Name:%s",a.name);gotoxy(47,10);printf("\xB2");

gotoxy(20,11);

printf("\xB2 Author:%s",a.Author);gotoxy(47,11);printf("\xB2");

gotoxy(20,12);

printf("\xB2 Qantity:%d",a.quantity);gotoxy(47,12);printf("\xB2");

gotoxy(20,13);

printf("\xB2 Price:Rs.%.2f",a.Price);gotoxy(47,13);printf("\xB2");

gotoxy(20,14);

printf("\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2");

d++;

}

}

if(d==0)

{

gotoxy(20,8);

printf("\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2");

gotoxy(20,9);printf("\xB2"); gotoxy(38,9);printf("\xB2");

gotoxy(20,10);

printf("\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2");

gotoxy(22,9);printf("\aNo Record Found");

}

gotoxy(20,17);

printf("Try another search?(Y/N)");

if(getch()=='y')

searchbooks();

else

menu();

break;

}

default :

getch();

searchbooks();

}

fclose(fp);

}

void viewbooks(void) //show the list of book persists in library

{

int i=0,j;

system("cls");

gotoxy(1,1);

printf("::::::::::::::::::::::::::::::::Book List:::::::::::::::::::::::::::::::::");

gotoxy(2,2);printf(" ID BOOK NAME AUTHOR QTY PRICE ");

j=4;

fp=fopen("lib.dat","rb");

while(fread(&a,sizeof(a),1,fp)==1)

{

gotoxy(16,j);

printf("%d",a.id);

gotoxy(22,j);

printf("%s",a.name);

gotoxy(36,j);

printf("%s",a.Author);

gotoxy(50,j);

printf("%d",a.quantity);

gotoxy(57,j);

printf("%.2f",a.Price);

printf("\n\n");

j++;

i=i+a.quantity;

}

gotoxy(3,25);

printf("Total Books =%d",i);

fclose(fp);

gotoxy(35,26);

returnfunc();

}

void editbooks(void) //edit information about book

{

system("cls");

int c=0;

int d,e;

gotoxy(20,4);

printf("\*\*\*\*Edit Books Section\*\*\*\*");

char another='y';

while(another=='y')

{

system("cls");

gotoxy(15,6);

printf("Enter Book Id to be edited:");

scanf("%d",&d);

fp=fopen("lib.dat","rb+");

while(fread(&a,sizeof(a),1,fp)==1)

{

if(checkid(d)==0)

{

gotoxy(15,7);

printf("The Book Is Available");

gotoxy(15,8);

printf("The Book ID:%d",a.id);

gotoxy(15,9);

printf("Enter New Name:");scanf("%s",a.name);

gotoxy(15,10);

printf("Enter New Author:");scanf("%s",a.Author);

gotoxy(15,11);

printf("Add New Quantity:");scanf("%d",&a.quantity);

gotoxy(15,12);

printf("Enter New Price:");scanf("%f",&a.Price);

gotoxy(15,14);

printf("The record is modified");

fseek(fp,ftell(fp)-sizeof(a),0);

fwrite(&a,sizeof(a),1,fp);

fclose(fp);

c=1;

}

if(c==0)

{

gotoxy(15,9);

printf("No record found");

}

}

gotoxy(15,16);

printf("Modify another Record?(Y/N)");

fflush(stdin);

another=getch() ;

}

printf("\n");

returnfunc();

}

int getdata()

{

int t;

gotoxy(20,3);printf("Enter the Information Below");

gotoxy(20,4);printf("\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2");

gotoxy(20,5);

printf("\xB2");gotoxy(46,5);printf("\xB2");

gotoxy(20,6);

printf("\xB2");gotoxy(46,6);printf("\xB2");

gotoxy(20,7);

printf("\xB2");gotoxy(46,7);printf("\xB2");

gotoxy(20,8);

printf("\xB2");gotoxy(46,8);printf("\xB2");

gotoxy(20,9);

printf("\xB2");gotoxy(46,9);printf("\xB2");

gotoxy(20,10);

printf("\xB2");gotoxy(46,10);printf("\xB2");

gotoxy(20,11);

printf("\xB2");gotoxy(46,11);printf("\xB2");

gotoxy(20,12);

printf("\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2\xB2");

gotoxy(21,6);

printf("Book ID:\t");

gotoxy(30,6);

scanf("%d",&t);

if(checkid(t) == 0)

{

gotoxy(21,13);

printf("\aThe book id already exists\a");

getch();

menu();

return 0;

}

a.id=t;

gotoxy(21,7);

printf("Book Name:");gotoxy(33,7);

scanf("%s",a.name);

gotoxy(21,8);

printf("Author:");gotoxy(30,8);

scanf("%s",a.Author);

gotoxy(21,9);

printf("Quantity:");gotoxy(31,9);

scanf("%d",&a.quantity);

gotoxy(21,10);

printf("Price:");gotoxy(28,10);

scanf("%f",&a.Price);

return 1;

}

int checkid(int t) //check whether the book is exist or not

{

rewind(fp);

while(fread(&a,sizeof(a),1,fp)==1)

if(a.id==t)

return 0; //returns 0 if book exits

return 1; //return 1 if it not

}

void Password(void) //for password option

{

system("cls");

char ch,pass[10];

int i=0,j;

printf("Enter The Password:");

while(ch!=13)

{

ch=getch();

if(ch!=13 && ch!=8)

{

putch('\*');

pass[i] = ch;

i++;

}

}

pass[i] = '\0';

if(strcmp(pass,password)==0)

{

printf("\n\nNow You Are Logged In!");

printf("\nPress any key to countinue...");

getch();

menu();

}

else

{

printf("\n\n\aWarning!! Incorrect Password");

getch();

Password();

}

}

void returnfunc(void)

{

{

printf(" Press ENTER to return to main menu");

}

a:

if(getch()==13) //allow only use of enter

menu();

else

goto a;

}

**GITHUB LINK**

<https://github.com/Macro-tech>

**Folder Structure of the Project**

The main code can be placed in any file and a folder.

We have a lib.dat File will be created to store the books information in the encrypted form.

These are the folder structures for this project.

**TESTING:**

If the user enters wrong password, then the application displays invalid password and logs in only when the correct password is entered.

Graphical user interface, application

Description automatically generated

If the user searches for a record that does not exist or tries to delete a record that does not exist then they will get a no record found prompt

Graphical user interface, text

Description automatically generated

**RESULT:**

Starting Page

Graphical user interface, application

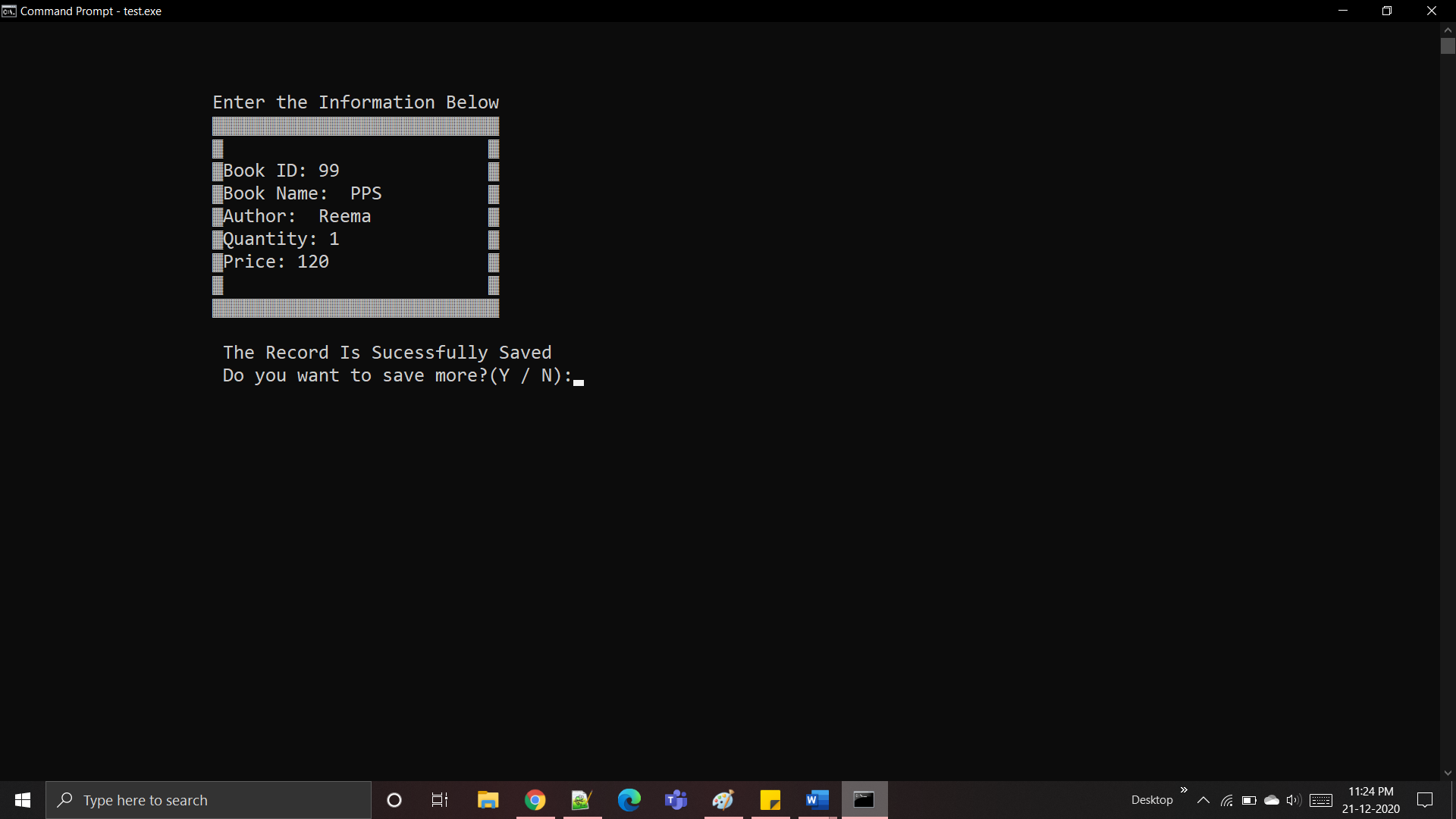
Description automatically generated

Menu

A picture containing text, screenshot, electronics, computer

Description automatically generated

1.Adding a book



2.Deleting a record

A picture containing text, screenshot, electronics, display

Description automatically generated

3. Searching a Book

Graphical user interface, text, application

Description automatically generated

3.1 By ID

Graphical user interface, text

Description automatically generated

3.2 By Name

Graphical user interface, text

Description automatically generated

4. Displaying all the Books

A picture containing text, monitor, screenshot, electronics

Description automatically generated

5. Edit records

A picture containing text, screenshot, electronics, display

Description automatically generated

6. Exit

A picture containing text, screenshot, monitor, computer

Description automatically generated

**Additional Learning:**

This project is a great experience to learn many new things. It was very helpful in improving our coding skills and concepts related to C. Teamwork is another important skill in which we improved ourselves from sharing our ideas to presenting the final project. It helped us in organizing our thoughts, planning our work, time management, communication skills and learning together. Working as a team also helped in completing the work in less time and in more efficient way and sharing the workload eases the burnout. Self-learning is another important aspect which we learnt while working for this project. We learnt many new concepts and different ways of approach to the problem, fixing the errors, debugging. We also had a thorough revision of the concepts and made our basic even stronger It also helped us to be more creative in various steps of its development. It is an opportunity to enhance our skills.

**CONCLUSION AND FUTURE WORK**

In conclusion, we have successfully developed a project which can be utilized by students to manage their inventory. The project also makes the library like experience much more student friendly.

The Project has opened up the possibilities for a much bigger and more comprehensive project for us. We could add different functions like assignments tab, marks tab, suggestions etc. In the future, we can rebuild this project using databases and a graphical user interface to give a better feel to the user and make inventory management much easier.

**REFERENCES**

1. <http://geeksforgeeks.com/>
2. <https://stackoverflow.com/>
3. <https://www.gsmarena.com/>
4. <https://www.amazon.in/>