

TABLE OF CONTENT

1. PROBLEM DEFINATION
2. SYSTEM REQUIREMENT SPECIFICATION
 - PURPOSE
 - SCOPE
 - REFERENCE
 - OVERVIEW
3. OVERALL DESCRIPTION
 - HARDWARE INTERFACE
 - SOFTWARE INTERFACE
 - MEMORY CONSTRAINED
 - OPERATION
 - PRODUCT FUNCTION
 - CONSTRAINTS
4. FRONT END DESCRIPTION
5. BACK END DESCRIPTION
6. DATA FLOW DIAGRAM(DFD)
7. USE CASE DIAGRAM
8. ER DIAGRAM
9. TESTING
- 10.SAMPLE OUTPUTS
- 11.CODE
- 12.RESULT

1.PROBLEM DEFINATION:

The tourist management system automates the work of user and admin.

- User can login by entering login id and password which he gets by signing in.
- System provides information of the ticket books and packages available.
- User and admin both can change their password anytime.
- Admin can see all the tickets book by all the user

2.SYSTEM REQUIREMENT SPECIFICATION:

- PURPOSE: The purpose of this srs is to describe the requirements involved in developing tourist management system
- Scope:
 - The product is titled as tourist management system
 - The system allows user to perform following task:
 1. book ticket
 2. cancel ticket
 3. Show ticket
 4. Show packages
 5. Change password
 - The system allows admin to perform following task:
 1. Input packages
 2. Delete packages
 3. Show all tickets
 4. Show all packages
 5. Change password
 - The system allows user with sign in and login function

- REFERENCE:

IEEE standard 830-1998 recommended practice for Software Requirements Specifications-Description.

- OVERVIEW:

- The SRS contains an analysis of the requirements necessary to help easy design.
- The overall description provides interface requirements for the tourist Management System, product perspective, hardware interfaces, software interfaces, communication interface, memory constraints, product functions, user characteristics and other constraints.
- Succeeding pages illustrate the characteristics of typical naïve users accessing the system along with legal and functional constraints enforced that affect tourist Management System in any fashion.

3.OVERALL DESCRIPTION

- HARDWARE:

- Required hard disk and ram to store the information for faster running of the system and to make the process speed high.
- The system must interface with the standard output device, keyboard and mouse to interact with this software.

- SOFTWARE:

- FRONT-END: C++ AND MICROSOFT VISUALS
- BACK-END: FILE HANDLING IN C++

- MEMORY CONSTRAINED:

- No specific constraints on memory.

- OPERATION:

- BY STORING DETAILS, IT ALLOW TO SIGNIN.
- BY RETRAIVING DATA FROM TXT FILE IT ALLOW SIGNED USER TO LOGIN
- THERE IS TWO PANEL 1 FOR SIGNIN AND OTHER FOR USER
- USER CAN DO OPERATION TICKETS

- **PRODUCT FUNCTION:**

- AFTER THE VALIDATION SYSTEM ALLOWS USER TO LOGIN
- USER CAN BOOK TICKET BY ENTERING THE PACKAGE NUMBER
- USER CAN CANCEL AND SEE THE TICKET WHICH HE BOOKED.
- GIVES OPTION TO CHANGE PASSWORD FOR BOTH USER AND ADMINISTRATOR
- ADMINSTRATOR CAN EDIT ANDCHANGE THE PACKAGE,

- **CONSTRAINTS:**

THE USER HAS UNIQUE ID AND PASSWORD WHICH HE CAN CHANGE AT TIME OF FORGETING THE PASSWORD.

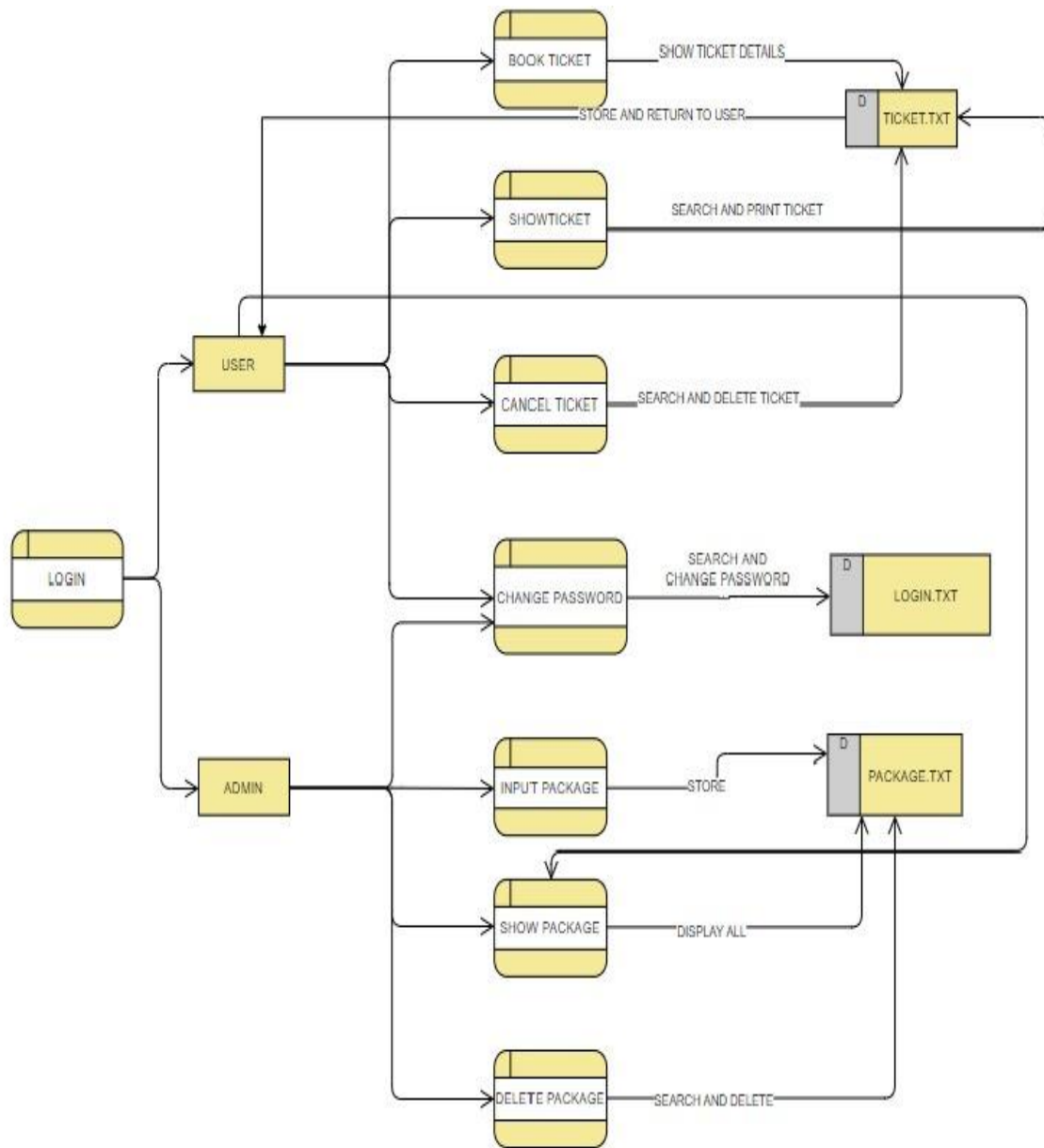
4.FRONT-END DESCRIPTION:

THE FIRST TIME ISTOR CAN SIGNIN AFTER THAT THEY ARE LOGIN AFTER AUNTHENTICATING THE USE CAN BOOK TICKET BY ENETERING THE AGE, NAME, GENDER OF THE PERSON THEN SYSTEM THE DISPLAY THE TOTAL PRICE OF THE TOUR HE HAS TO PAY AT TIME OF VISIT.ADMINSTRATOR CAN SEE ALL THE TICKETS.ADMINISTRATOR LOGIN ID IS 100000.

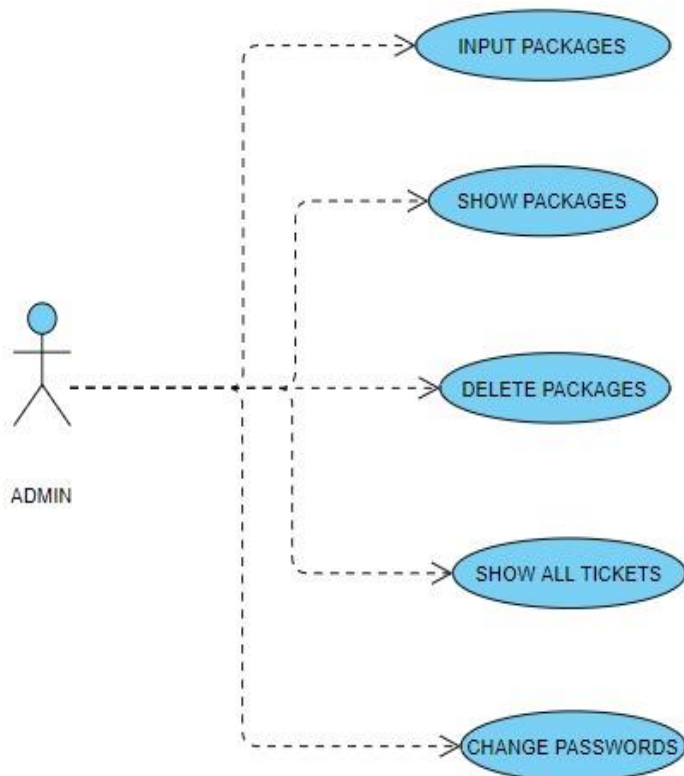
5.BACK-END DESCRIPTION:

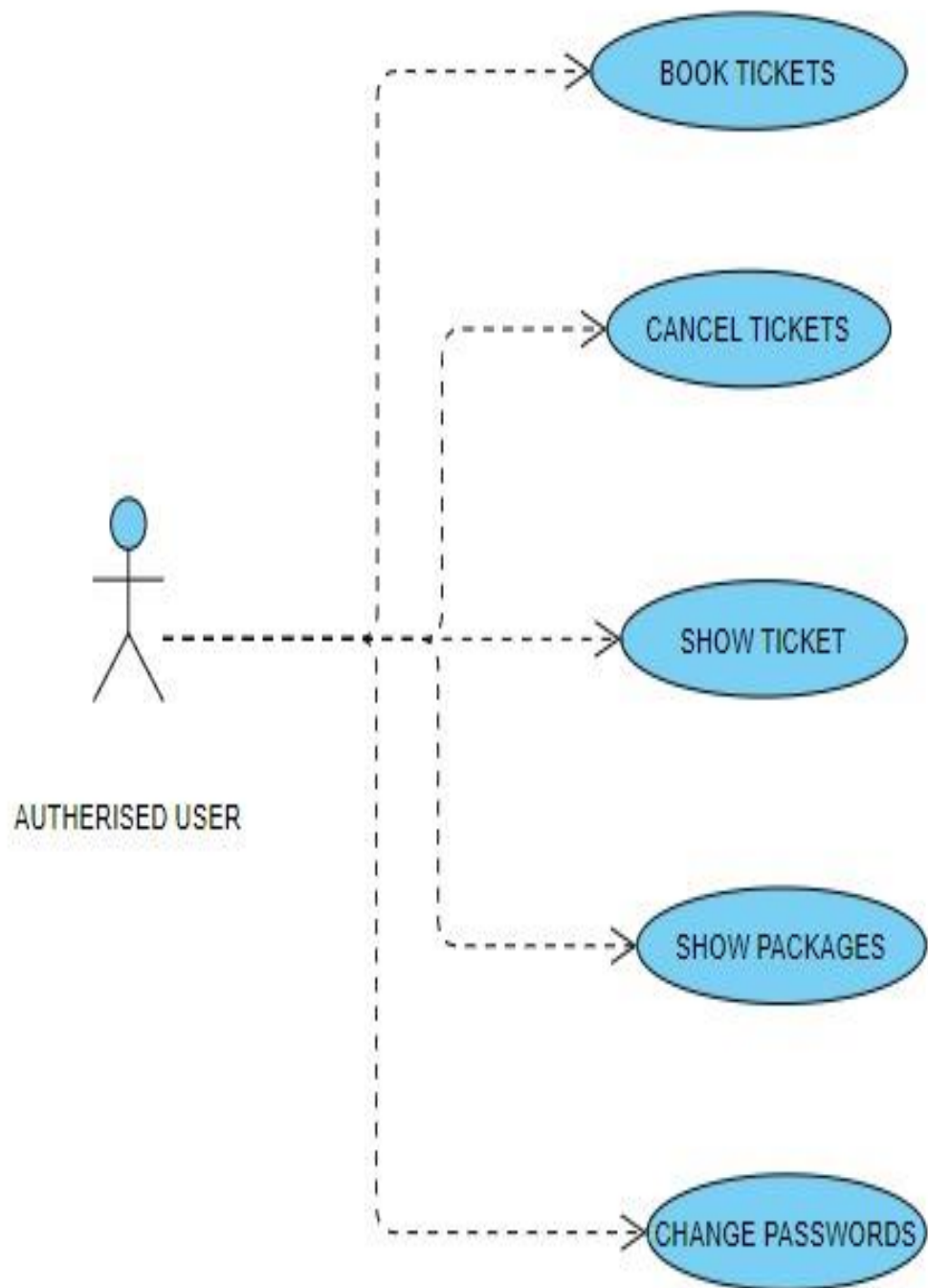
I had used file handling, in which all the login information is stored in login.txt,all the packages information is stored in package.txt and all the tickets related information is stored in ticket.txt.

DATA FLOW DIAGRAM(DFD)



USE CASE DIAGRAM

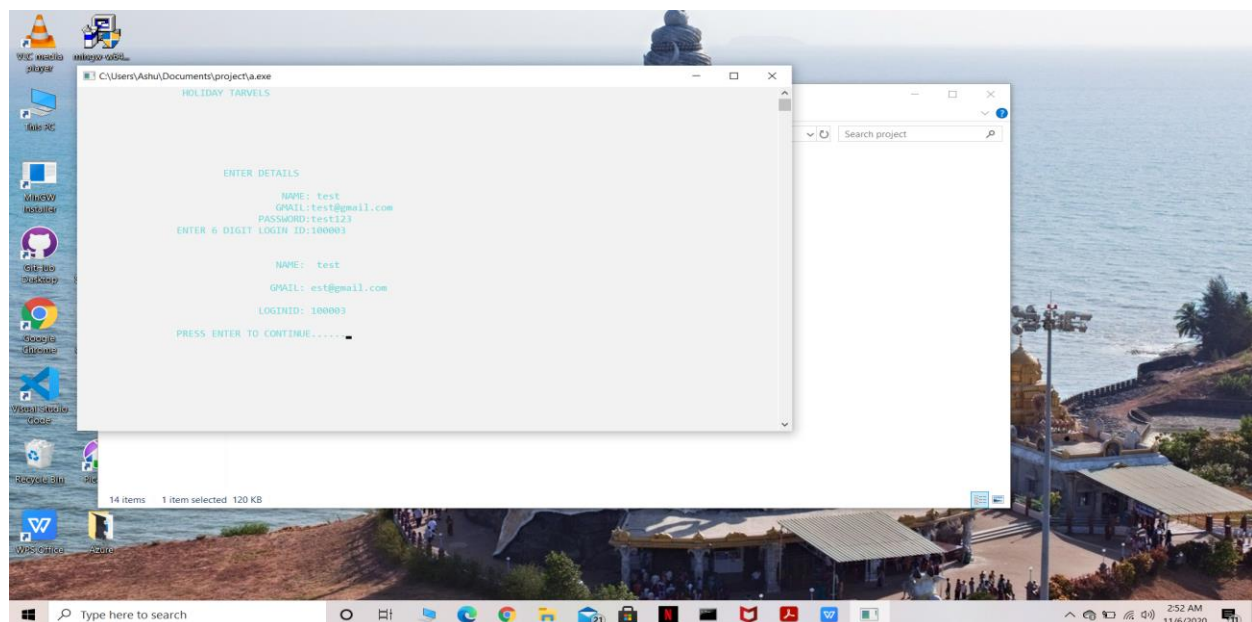
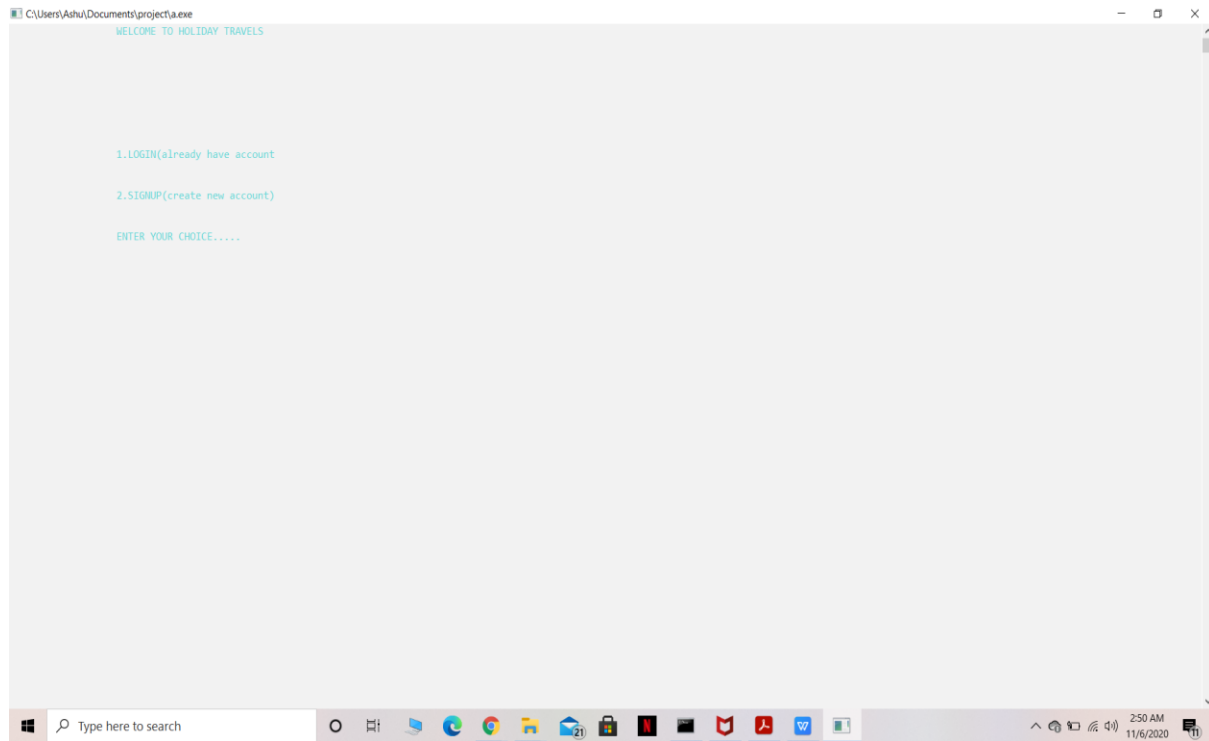


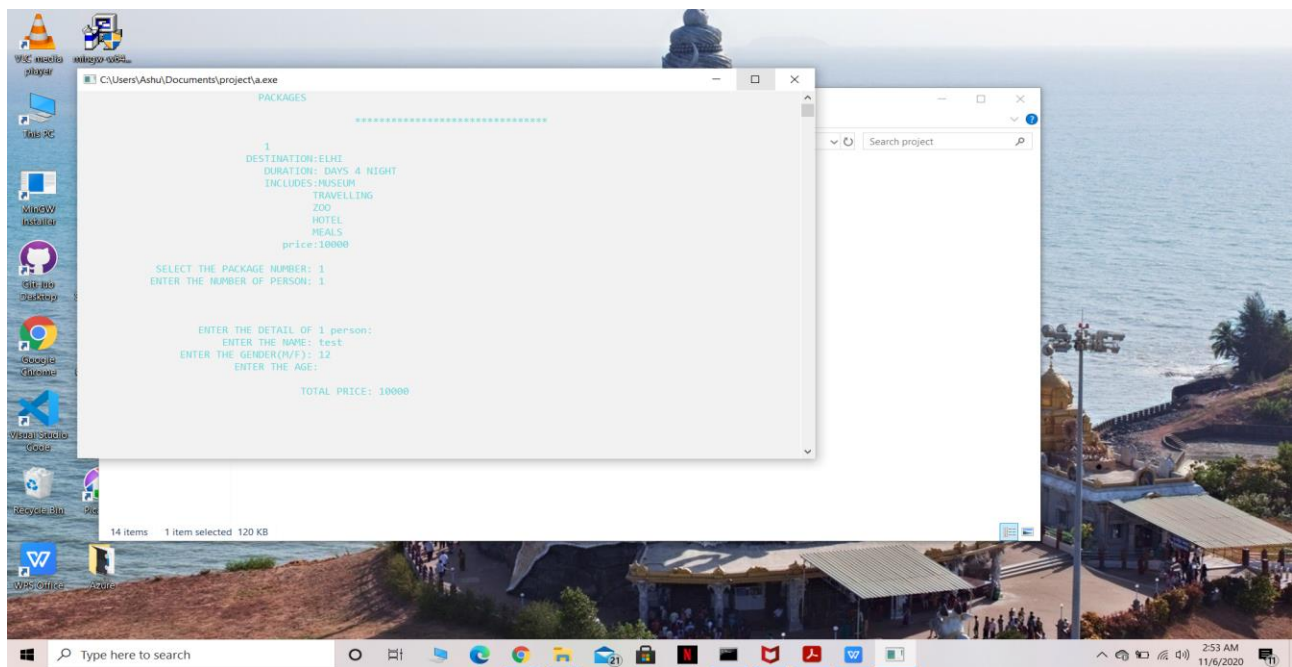
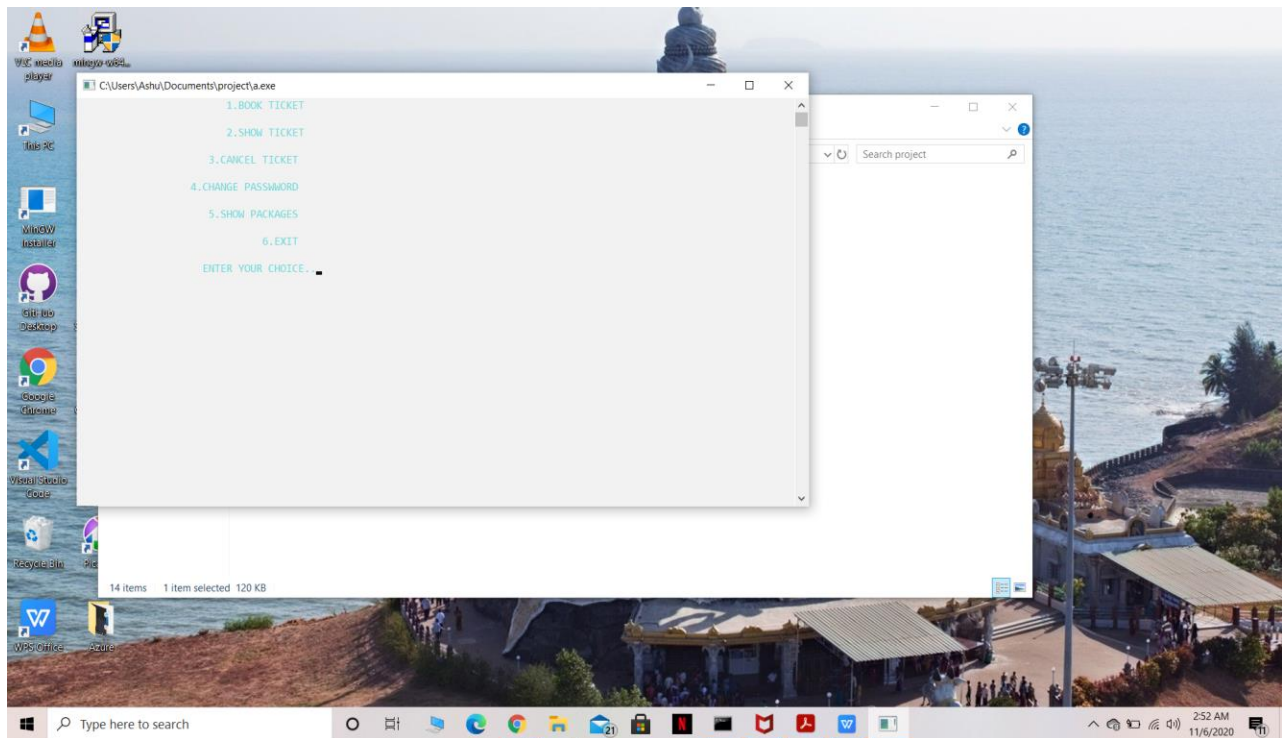


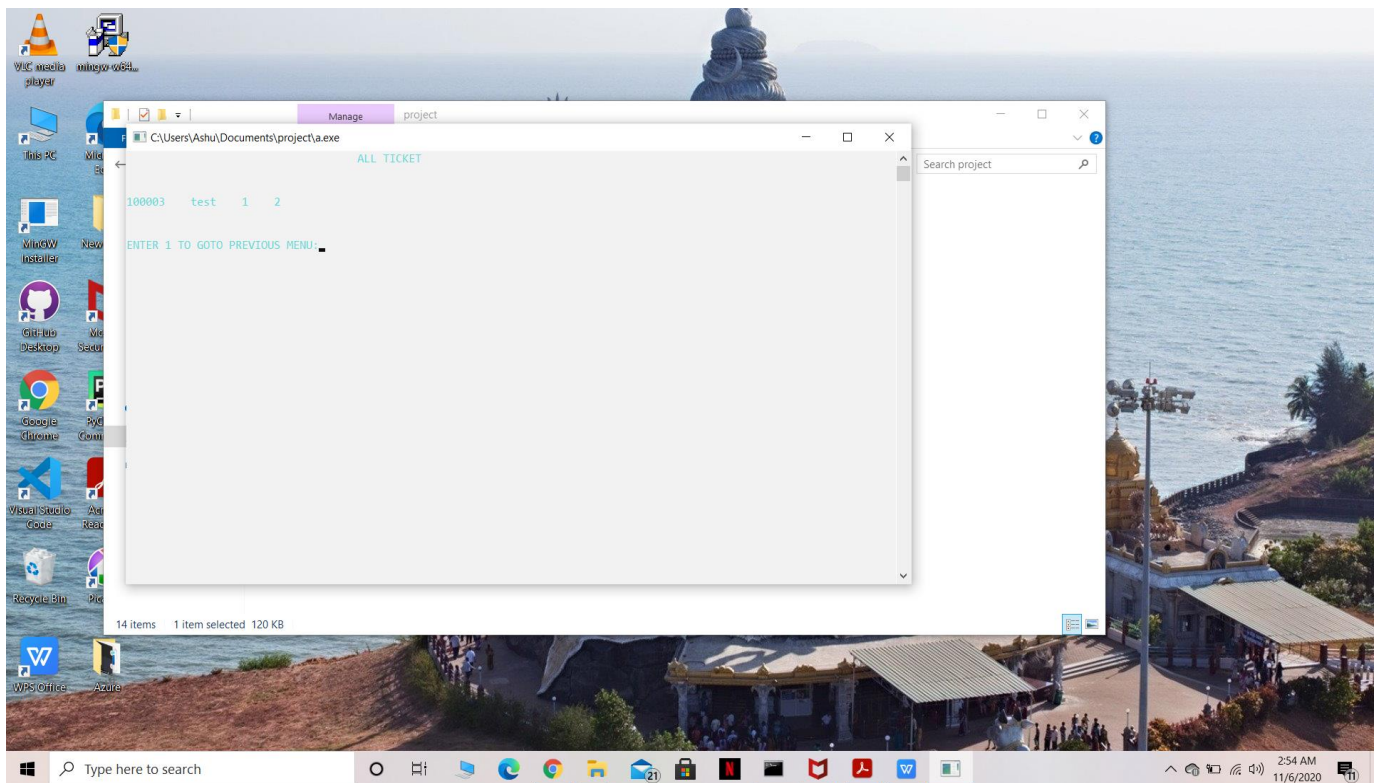
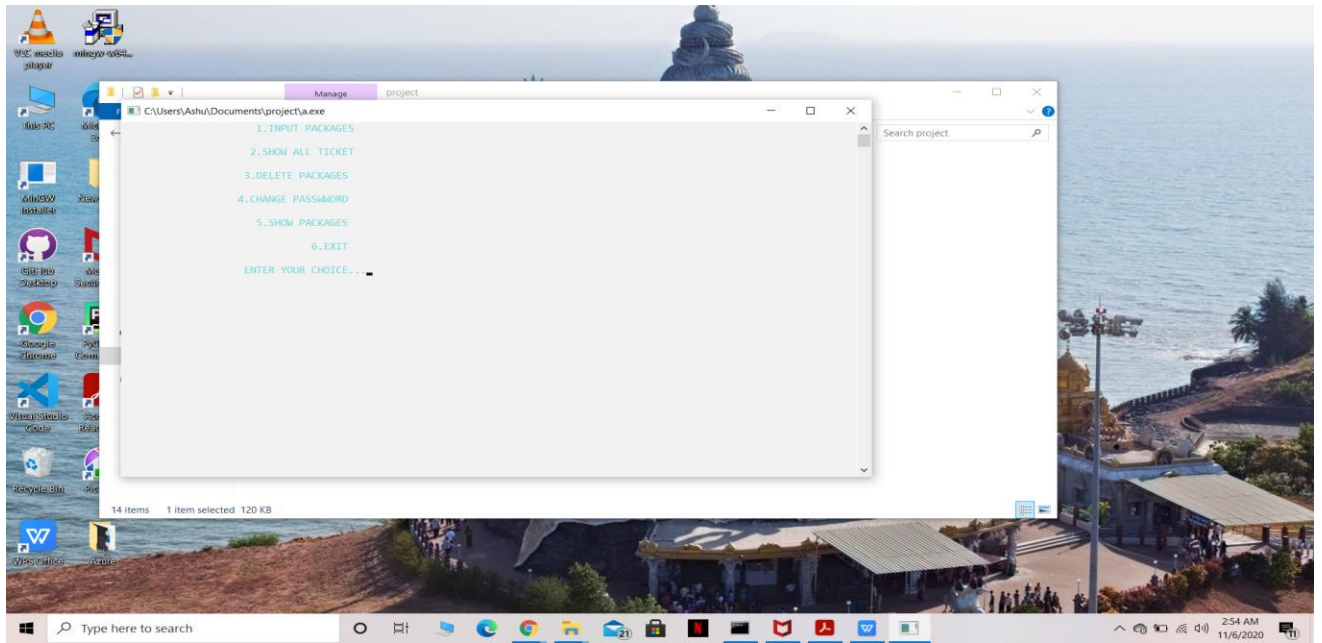
TESTING

OPERATION NAME	INPUT	EXPEXTED OUTPUT	ACTUAL OUTPUT	STATUS
LOGIN	LOGIN ID AND PASSSSWORD	LOGIN SUCCESFULLY	LOGIN SUCCESFULLY	PASS
SIGN IN	NAME, EMAIL, LOGIN ID	SIGN IN SUCCESFULLY	SIGN IN SUCCESFULLY	PASS
BOOK TICKET	NO OF PERSON, NAME, GENDER, AGE	TICKET BOOK SUCCESFULLY	TICKET BOOK SUCCESFULLY	PASS
CANCEL TICKET	LOGINED USER	TICKET CANCELED SUCCESFULLY	TICKET CANCELED SUCCESFULLY	PASS
CHANGE PASSWORD	ID AND NEW PASSWORD	PASSWORD CHANGED SUCCESFULLY	PASSWORD CHANGED SUCCESFULLY	PASS
INPUT PACKAGES	LOCATION INCLUDE PRICE DURATION	PACKAGED INPUT SUCCESFULLY	PACKAGED INPUT SUCCESFULLY	PASS
DELETE PACKAGE	PACKAGE NUMBER	PACKAGE DELETED SUCCESFULLY	PACKAGE DELETED	PASS

SAMPLE OUTPUT







CODE

```
#include<iostream>
#include<unistd.h>
#include<iomanip>
#include<windows.h>
#include<conio.h>
#include<stdio.h>
#include<fstream>
using namespace std;
class info
{
    public:
    char gender,name[30];
    int age,num;
    void inp(int nu)
    {
        cout<<endl;
        cin.ignore();
        cout<<setw(40)<<"ENTER THE NAME: ";
        gets(name);
        cout<<setw(40)<<"ENTER THE GENDER(M/F): ";
        cin>>gender;
```

```

        cout<<setw(40)<<"ENTER THE AGE:";

        cin>>age;

        num=nu;
    }
};

class login1
{

public:
    long no;
    char pass[20],name[20],gmail[50];
    void inp()
    {
        cout<<setw(40)<<"ENTER DETAILS\n\n";
        cout<<setw(40)<<"NAME:";
        cin.ignore();
        gets(name);
        cout<<setw(40)<<"GMAIL:";
        cin.ignore();
        gets(gmail);
        cout<<setw(40)<<"PASSWORD:";
        gets(pass);
        cout<<setw(40)<<"ENTER 6 DIGIT LOGIN ID:";
        cin>>no;

    }

```

```

};

class pack
{
public:
    int num;
    char destination[30],include[100],duration[100];
    long price;
    void inp()
    {
        cout<<setw(50)<<"ENTER DETAILS\n\n\n";

        static int ser_no=0;

        ser_no++;

        num=ser_no;

        cout<<setw(40)<<"ENTER PRICE:";

        cin>>price;

        cout<<setw(40)<<"ENTER INCLUDE:";

        cin.ignore();

        gets(include);

        cout<<setw(40)<<"ENTER DESTINATION:";

        cin.ignore();

        gets(destination);

        cout<<setw(40)<<"ENTER DURATION:";

        cin.ignore();

        gets(duration);
    }

    void disp()
    {

        cout<<setw(80)<<"*****\n\n";

        cout<<setw(32)<<num<<endl;
    }
}

```

```

    cout<<setw(40)<<"DESTINATION:"<<destination<<endl;
    cout<<setw(40)<<"DURATION:"<<duration<<endl;
    cout<<setw(40)<<"INCLUDES:";
    int i=0;
    for(i=0;include[i]!='\0';i++)
    {
        if(include[i]==' ')
        {
            cout<<endl;
            cout<<setw(40);
        }
        else
        {
            cout<<include[i];
        }
    }
    cout<<endl<<setw(40)<<"price:"<<price;
}

};

void bling(char ch[])
{

    int i;

    for(i=0;ch[i]!='\0';i++)
    {
        cout<<ch[i];
        //sleep(1);
    }
}

```

```

    }
}
void begin1();
void front(int);
void admin();
void login()
{
    system("cls");
    cout<<setw(40)<<"HOLIDAY TARVELS\n\n\n\n\n\n\n";
    long num;
    char passw[20];
    cout<<setw(40)<<"LOGINID:";
    cin>>num;
    cout<<setw(40)<<"PASSWORD:";
    cin.ignore();
    gets(passw);
    fstream file;
    file.open("login.txt",ios::in|ios::binary);
    login1 ob;
    file.read((char*)&ob,sizeof(ob));
    int flag=0;
    while(!file.eof())
    {
        if(num==ob.no)
        {
            if(strcmp(ob.pass,passw)==0)
            {
                flag=1;
                if(num==100000)

```



```

        admin();
    else
        front(num);
    }
}

file.read((char*)&ob,sizeof(ob));
}

if(flag==0)
{
    cout<<setw(40)<<"WRONG ID OR PASSWORD TRY AGAIN";
    sleep(2);
    login();
}

file.close();
}

void disp(login1 ob)
{
    int n;
    cout<<setw(40)<<"NAME: "<<ob.name<<"\n\n";
    cout<<setw(40)<<"GMAIL: "<<ob.gmail<<"\n\n";
    cout<<setw(40)<<"LOGINID: "<<ob.no<<"\n\n";
    cout<<setw(40)<<"PRESS ENTER TO CONTINUE";
    char ch[10]=".....";
    bling(ch);
    cin>>n;
}

void signup()
{
    system("cls");

```

```

    cout<<setw(40)<<"HOLIDAY TARVELS\n\n\n\n\n\n\n";
    ofstream file;
    file.open("login.txt",ios::app|ios::binary);
    login1 ob;
    ob.inp();
    file.write((char*)&ob,sizeof(ob));
    cout<<"\n\n";
    disp(ob);
    cout<<setw(40)<<"REGISTERED";
    sleep(2);
    file.close();
    begin1();

}

void show_pack();
void inp_pack();
void change_passw(int);
void del_pack();
//void update_pack();
void book_ticket(int);
void show_ticket(int);
void cancel_ticket(int);
void disp_tick();
void inpu(int);
int main()
{
    system("color FB");
    begin1();

```

```

    return 0;
}
void inp_pack()
{
    system("cls");
    pack ob;
    ob.inp();
    fstream file;
    file.open("packages.txt",ios::app|ios::binary);
    file.write((char*)&ob,sizeof(ob));
    file.close();
}
void inpu(int n)
{
    info ob;
    ob.inp(n);
    fstream file;
    file.open("ticket.txt",ios::app|ios::binary);
    file.write((char*)&ob,sizeof(ob));
    file.close();
}
void show_pack()
{
    cout<<setw(40)<<"PACKAGES\n\n";
    fstream file;
    file.open("packages.txt",ios::in|ios::binary);
    pack ob;
    file.read((char*)&ob,sizeof(ob));

```

```

while(!file.eof())
{
    ob_disp();
    file.read((char*)&ob,sizeof(ob));
}
sleep(20);
file.close();
}

void begin1()
{
    system("cls");
    char ch[100]="WELCOME TO HOLIDAY TRAVELS";
    cout<<setw(20);
    bling(ch);
    cout<<"\n\n\n\n\n\n\n\n\n";
    cout<<setw(50);
    cout<<"1.LOGIN(already have account\n\n\n";
    cout<<setw(50);
    cout<<"2.SIGNUP(create new account)\n\n\n";
    cout<<setw(36);
    cout<<"ENTER YOUR CHOICE";
    char a[100]=".....";
    bling(a);
    cout<<"\b\b\b\b\b\b";
    bling(a);
    cout<<"    ";
    int n;
    cin>>n;
    if(n==1)

```

```

    login();

    else

        signup();
}

void admin()
{
    system("cls");

    cout<<setw(40)<<"welcome to admin pannel";

    int n=0;

    while(n!=6)
    {
        system("cls");

        cout<<setw(40)<<"1.INPUT PACKAGES\n\n";
        cout<<setw(40)<<"2.SHOW ALL TICKET\n\n";
        cout<<setw(40)<<"3.DELETE PACKAGES \n\n";
        cout<<setw(40)<<"4.CHANGE PASSWORD \n\n";
        cout<<setw(40)<<"5.SHOW PACKAGES \n\n";
        cout<<setw(40)<<"6.EXIT \n\n";
        cout<<setw(40)<<"ENTER YOUR CHOICE...";

        cin>>n;

        switch(n)
        {
            case 1:inp_pack();

                break;

            case 5: show_pack();

                break;

            case 4: change_passw(1);

                break;

            case 3: del_pack();

```

```

        break;

    case 2: disp_tick();

        break;

    }

}

}

void change_passw(int l)
{
    int n;

    system("cls");

    cout<<setw(50)<<"CHANGE YOUR PASSWORD\n\n";

    cout<<"\n\n\n";

    cout<<setw(30)<<"ENTER LOGIN ID:";

    cin>>n;

    char num[50];

    cout<<setw(30)<<"ENTER NEW PASSWORD:";

    cin.ignore();

    gets(num);

    fstream file;

    file.open("login.txt",ios::in|ios::binary|ios::out);

    login1 ob;

    file.read((char*)&ob,sizeof(ob));

    fstream file1;

    file1.open("xyz.txt",ios::out|ios::binary|ios::in);

    while(!file.eof())

    {

        if(ob.no!=n)

        {

            file1.write((char*)&ob,sizeof(ob));

```

```

    }
    else
    {
        strcpy(ob.pass,num);
        file1.write((char*)&ob,sizeof(ob));
    }
    file.read((char*)&ob,sizeof(ob));
}
file.close();
file1.close();
file.open("login.txt",ios::out|ios::binary);
file1.open("xyz.txt",ios::in|ios::binary);
file1.read((char*)&ob,sizeof(ob));
while(!file1.eof())
{
    file.write((char*)&ob,sizeof(ob));
    file1.read((char*)&ob,sizeof(ob));
}
file1.close();
cout<<setw(50)<<"PASSWORD CHANGED SUCCESSFULLY";
sleep(5);
if(l==1)
    admin();
else
{
    front(ob.no);
}

```

```

}

void front(int nu)
{
system("cls");

cout<<setw(40)<<"WELCOME TO HOLIDAY TRAVELS";

int n=0;
while(n!=6)
{
    system("cls");
    cout<<setw(40)<<"1.BOOK TICKET\n\n";
    cout<<setw(40)<<"2.SHOW TICKET\n\n";
    cout<<setw(40)<<"3.CANCEL TICKET \n\n";
    cout<<setw(40)<<"4.CHANGE PASSWORD \n\n";
    cout<<setw(40)<<"5.SHOW PACKAGES \n\n";
    cout<<setw(40)<<"6.EXIT \n\n";
    cout<<setw(40)<<"ENTER YOUR CHOICE..";
    cin>>n;
    switch(n)
    {
        case 1:book_ticket(nu);
        break;
        case 2:show_ticket(nu);
        break;
        case 3: cancel_ticket(nu);
        break;
        case 4: change_passw(0);
        break;
        case 5: show_pack();

```



```

        break;

    }

}

void show_ticket(int nu)
{
    system("cls");
    fstream new1;
    info u ;
    new1.open("ticket.txt",ios::binary|ios::in);
    new1.read((char*)&u,sizeof(u));
    while(!new1.eof())
    {
        if(u.num==nu)
            cout<<setw(50)<<" "<<u.name<<" "<<u.gender<<" "<<u.age<<endl;
        new1.read((char*)&u,sizeof(u));
    }
    new1.close();
    int n;
    cout<<setw(50)<<"ENTER ANY NUMBER TO EXIT.....";
    cin>>n;
}

void cancel_ticket(int nu)
{
    fstream fin;
    fin.open("ticket.txt",ios::binary|ios::in);
    fstream temp;
    temp.open("temp1.txt",ios::binary|ios::out);

```

```

info ob;

fin.read((char*)&ob,sizeof(ob));

while(!fin.eof())
{
    if(nu!=ob.num)
        temp.write((char*)&ob,sizeof(ob));
    fin.read((char*)&ob,sizeof(ob));
}

fin.close();

temp.close();

fin.open("ticket.txt",ios::binary|ios::out);

temp.open("temp1.txt",ios::binary|ios::in);

temp.read((char*)&ob,sizeof(ob));

while(!temp.eof())
{
    fin.write((char*)&ob,sizeof(ob));
    temp.read((char*)&ob,sizeof(ob));
}

fin.close();

temp.close();
}

void book_ticket(int nu)
{
    system("cls");
    show_pack();

    int n,i,j;

    cout<<endl<<endl;

    cout<<setw(40)<<"SELECT THE PACKAGE NUMBER: ";

    cin>>n;

```

```

cout<<setw(40)<<"ENTER THE NUMBER OF PERSON: ";
cin>>j;
info ob[j];
cout<<"\n\n";
fstream new1;
new1.open("ticket.txt",ios::app|ios::binary);
for(i=0;i<j;i++)
{
    cout<<endl<<setw(40)<<"ENTER THE DETAIL OF "<<i+1<<" person: ";
    inpu(nu);
}
nu++;
fstream fin;
fin.open("packages.txt",ios::binary|ios::in);
pack ib;
int price;
fin.read((char*)&ib,sizeof(ib));
while(!fin.eof())
{
    if(ib.num==n)
        price=ib.price;
    fin.read((char*)&ib,sizeof(ib));
}
fin.close();
cout<<"\n\n"<<setw(50)<<"TOTAL PRICE: "<<price*j;
sleep(5);
}
void del_pack()
{

```

```

system("cls");
show_pack();
int n;
cout<<setw(40)<<"ENTER THE NUMBER: ";
cin>>n;
fstream xyz;
xyz.open("packages.txt",ios::binary|ios::in);
pack ob;
xyz.read((char*)&ob,sizeof(ob));
fstream new1;
new1.open("temp.txt",ios::binary|ios::out);
while(!xyz.eof())
{
    if(ob.num!=n)
    {
        new1.write((char*)&ob,sizeof(ob));
    }
    xyz.read((char*)&ob,sizeof(ob));
}
new1.close();
xyz.close();
xyz.open("packages.txt",ios::out|ios::binary);
new1.open("temp.txt",ios::in|ios::binary);
new1.read((char*)&ob,sizeof(ob));
while(!new1.eof())
{
    xyz.write((char*)&ob,sizeof(ob));
    new1.read((char*)&ob,sizeof(ob));
}

```

```

xyz.close();

new1.close();

admin();

}

void disp_tick()
{
    system("cls");
    cout<<setw(50)<<"ALL TICKET \n\n\n";
    fstream fin;
    fin.open("ticket.txt",ios::binary|ios::in);
    info ob;
    fin.read((char*)&ob,sizeof(ob));
    while(!fin.eof())
    {
        cout<<ob.num<<" "<<ob.name<<" "<<ob.gender<<" "<<ob.age<<endl;
        fin.read((char*)&ob,sizeof(ob));
    }
    fin.close();

    int n;
    cout<<"\n\n"<<"ENTER 1 TO GOTO PREVIOUS MENU:";

    cin>>n;

    if(n==1)
        admin();
}

```

RESULT

Thus the TOURIST MANAGMENT SYSTEM was implemented using the specified front end and back end tools.