Project Problem:

Create a train reservation system with following conditions

a. There are many trains available for each destination which can be stored in linked list.

b. Detail of each train is saved in a stack

c. The customers are organized as a Queue served FCFS.

Problem Solution:

Basic Concept:

We will have to a create online reservation system. We will have to use queue and linked list data Structure. We will need to use multiple header file so the main function looks clean and easy to understand.

The number linked list will be the total number of stations. Each station will have own linked list where all the train no will be stored.

Each train have a unique train no.

We will use two header files one for the log in function and another one for to store our data structure.

Only the menu functions will be written in the main function.

Header files and Main functions:

Login system:

In this header file we stored all necessary functions for login system. Here we have created a class named "loginManager". In that class there is a login () function which will be used for the login operation.

The usernames and passwords are saved and their class are private, so only admin have the ability to edit them.

Whenever the login () function will be called it will ask the user for the username and password. After user enters the username and password it will check if there is a username and password saved in the file. If it is then it will successfully log in. Otherwise, they won't be able to access the menu.

Code representation:

```
#include<iostream>
using namespace std;

class loginManager
 {
 public:
    string usernameAttempt;
    string passwordAttempt;
    int a=0;;


    loginManager()
    {


    }
    int login ()
    {
        cout<<"Username: "<<endl;

        cin>>usernameAttempt;
        if (usernameAttempt==username)
        {
            cout<<"Password: "<<endl;
            cin>>passwordAttempt;
            if (passwordAttempt==password)
            {
                cout<<"Login Successful\n\n"<<endl;
```

```
            a=1;



        }
        else
        {
        cout<<"Invalid Password"<<endl;
        }
      }
      else
        {
        cout<<"Invalid Username"<<endl;
        }
        return a;
  }

private:
  string username="user";
  string password="1234";


};
```

Custom DLL & Queue:

We kept the necessary functions of DLL and queue together. We have also kept function where train details are stored. The function can take a train number

And prinout all the details of the train. Only admins can see the train details.

Whenever a linked list needs to be created add function will be called.

And queue will be used for processing user request for each seat. For each seat there will be a separate queue.For each request it will check if the queue is empty or not. If its not empty then it will say that seat is unavailable. Otherwise it will queue the request and after printing ticket it will perform the deque operation.

Code representation:

```
#include<bits/stdc++.h>
using namespace std;
class Node{
public:
int data;
Node *next;
Node *previous;
Node(int data){
{
```

```
data=data;
next=NULL;
previous=NULL;
}
this -> data = data;
}
};
class DLL
{
public:
Node *head= NULL;
Node *newdata= NULL;
void add(int item)
{
newdata = new Node(item);
if(head==NULL)
{
head = newdata;
}
else
{
Node *temp = head;
while(temp -> next != NULL ){
temp = temp -> next;
}
temp -> next = newdata;
newdata -> next = NULL;
newdata->previous=temp;
}
}

void display()
{
```

```
Node *temp= head;
while(temp!= NULL)
{
cout<<temp->data<<"-> ";
temp= temp->next;
}
cout<<endl;
}
void TrainDetails (int train_num)
{
  if (train_num==1130)
        {
              cout<<"\nTrain:\t\t\tSuborno Express"<<endl;
              cout<<"\nDestination:\t\tChattogram–Dhaka"<<endl;
              cout<<"\nDeparture:\t\t12am "<<endl;
        }
        if (train_num==1131)
        {
              cout<<"\nTrain:\t\t\tGodhuli Express"<<endl;
              cout<<"\nDestination:\t\tDhaka-Chattogram"<<endl;
              cout<<"\nDeparture:\t\6am"<<endl;
        }
        if (train_num==1132)
        {
              cout<<"\nTrain:\t\t\tParabat Express"<<endl;
              cout<<"\nDestination:\t\tDhaka–Sylhet"<<endl;
              cout<<"\nDeparture:\t\t8am"<<endl;
        }
        if (train_num==1133)
        {
              cout<<"\nTrain:\t\t\tJayantika Express"<<endl;
              cout<<"\nDestination:\t\tSylhet-Dhaka"<<endl;
              cout<<"\nDeparture:\t\t1am"<<endl;
```

```
        }
        if (train_num==1230)
        {
                cout<<"\nTrain:\t\t\tSuborno Express"<<endl;
                cout<<"\nDestination:\t\tChattogram–Dhaka"<<endl;
                cout<<"\nDeparture:\t\t6am "<<endl;
        }
        if (train_num==1231)
        {
                cout<<"\nTrain:\t\t\tGodhuli Express"<<endl;
                cout<<"\nDestination:\t\tDhaka-Chattogram"<<endl;
                cout<<"\nDeparture:\t\t12pm"<<endl;
        }
        if (train_num==1232)
        {
                cout<<"\nTrain:\t\t\tParabat Express"<<endl;
                cout<<"\nDestination:\t\tDhaka–Sylhet"<<endl;
                cout<<"\nDeparture:\t\t2pm"<<endl;
        }
        if (train_num==1233)
        {
                cout<<"\nTrain:\t\t\tJayantika Express"<<endl;
                cout<<"\nDestination:\t\tSylhet-Dhaka"<<endl;
                cout<<"\nDeparture:\t\t4pm"<<endl;
        }



};
class Queue {

    public:
```

```
Node *head= NULL;

Node *newdata= NULL;

void Enque(int item)
{

   newdata = new Node(item);

   if(head==NULL) {

      head = newdata;

   } else {

      Node *temp = head;

      while(temp -> next != NULL ) {

         temp = temp -> next;

      }

      temp -> next = newdata;

      newdata -> next = NULL;

   }

}
```

```
    bool isEmpty()
    {

        if(head == NULL)return true;

        else return false;

    }

    void Deque()
    {

        Node* temp = head -> next;

        delete head;

        head = temp;

    }



};
```

Main Function:

In the main function we have only kept the main function. We used two while loops one for the number attempt a user an have another one is for the menu.

When user successfully login they can either reserve or cancel tickect. For cancelling the ticket you will only need to enter the ticket no.

For reservation user can choose from where they wanna aboard, their destination. Then they will be shown available trains. From their they can choose which train tickect they want to book. Then they will choose the seast class and bogey na ,the number of seats they want to book and seat no they want to book.

Code representation:

```
#include<bits/stdc++.h>

#include"customDLL&Queue.h"

#include"login.h"

int main ()
{
   int choice;
```

```
int a=0;
int c=0;
srand (time(NULL));
time_t now=time(0);
char* dt=ctime(&now);

DLL *Dhaka=new DLL ();
DLL *Rajshahi=new DLL();

Dhaka->add (1130);
Dhaka->add (1131);
Dhaka->add (1132);
Dhaka->add (1133);



Rajshahi->add (1230);
Rajshahi->add (1231);
Rajshahi->add (1232);
Rajshahi->add (1233);

Queue *q=new Queue();

int Parabat_Express[20];
int Godhuli_express[20];

for (int i=0;i<20;i++)
{
 Parabat_Express[i]=0;
 Godhuli_express[i]=0;
}
```

```
    while (a!=3)
    {
    int b;

    loginManager log;
    b=log.login();
    a++;
    c++;

 if (b==1)
 {
    while (choice!=3)
{
    cout<<"1.Reserve a ticket"<<endl;
    cout<<"2.Cancel ticket"<<endl;
    cout<<"3.Exit"<<endl;
    cin>>choice;

    if (choice ==1)
    {
        int x;
        cout<<"From:\n1.Dhaka \n2.Rajshahi "<<endl;
        cin>>x;

        if (x==1)
        {
          int y;
         cout<<"Destination:\n1.Rajshahi \n2.Chottogram"<<endl;

         cin>>y;

         if (y==1)
         {   int z;
```

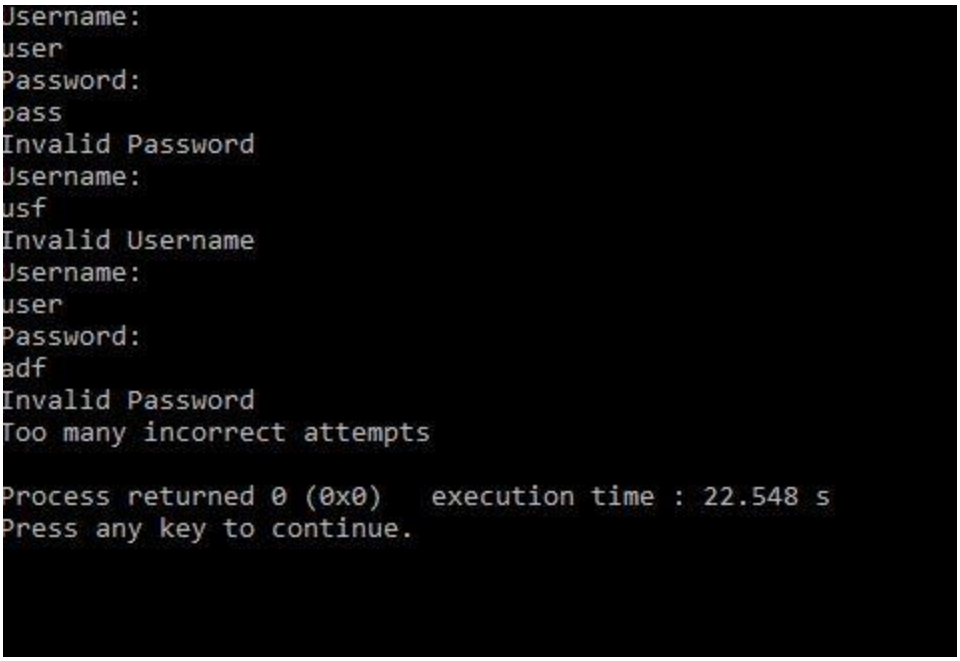
```
cout<<"Available Trains: \n1.Godhuli Express"<<endl;

cin>>z;
if (z==1)
{
   int e;
   cout<<"Select class: \n1.Shovon \n2.Ac coach \n3.Non-Ac coach" <<endl;
   cin>>e;


   if (e==1)
   {
     int f;
     cout<<"Select Bogey: \n1.A \n2.D" <<endl;
     cin>>f;


     if (f==1)
     {
        int g, m;

        cout<<"How many tikets do you want to buy"<<endl;
        cin>>m;

        for(int n=0;n<m;n++)
        {


        cout<<"Available seats"<<endl;
        for (int k=0;k<5;k++)
        {
           if(Godhuli_express[k]==0)
           {
```

```
            cout<<" "<<k+1;
          }
        }
        cout<<"\n\n Enter seat No:"<<endl;
        cin>>g;
        bool l= q->isEmpty();
        if (l='True')
        {
          q->Enque(g);
        }
        else
        {
        cout<<"Seat is unavailable"<<endl;
        }
        Godhuli_express[g-1]=1;
        }

        cout<<"                Ticket      "<<endl;
        cout<<"Name:User"<<endl;
        cout<<"Train:Godhuli Express"<<endl;
        cout<<"From:Dhaka                Destination:Rajshahi"<<endl;
        cout<<"Departure time: 4 pm          Arrival time:12 am"<<endl;
        cout<<"number of seats: "<<m<<"          Price : "<<350*m<<" Taka"<<endl;
        cout<<"Date &Time: "<<dt<<endl;
        cout<<"Ticket no: "<<rand()%100000<<endl;
        cout<<"\n\n";
        q->Deque();
      }

    }
  }

}
```

```
        }

    }
    if (choice==2)
    {
        int o;
        cout<<"Enter tickect No: "<<endl;
        cin>>o;
        cout<<"Ticket cancelled"<<endl;
    }
    else if (choice==3)
        {
            a=3;
            cout<<"Program ended successfully"<<endl;
        }

}
 }

    }
    if (c==3)
    {cout<<"Too many incorrect attempts"<<endl;}

}
```

Output:

```
Username:
user
Password:
1234
Login Successful


1.Reserve a ticket
2.Cancel ticket
3.Exit
1
From:
1.Dhaka
2.Rajshahi
1
Destination:
1.Rajshahi
2.Chottogram
1
Available Trains:
1.Godhuli Express
1
Select class:
1.Shovon
2.Ac coach
3.Non-Ac coach
1
Select Bogey:
1.A
2.D
1
How many tikets do you want to buy
2
Available seats
 1 2 3 4 5

 Enter seat No:
1
Available seats
 2 3 4 5

 Enter seat No:
4
                        Ticket
Name:User
Train:Godhuli Express
From:Dhaka                          Destination:Rajshahi
Departure time: 4 pm                Arrival time:12 am
number of seats: 2                  Price : 700 Taka
Date &Time: Fri May 20 12:27:53 2022

Ticket no: 24921


1.Reserve a ticket
2.Cancel ticket
3.Exit
```


Username:
user
Password:
pass
Invalid Password
Username:
usf
Invalid Username
Username:
user
Password:
adf
Invalid Password
Too many incorrect attempts
Process returned 0 (0x0) execution time : 22.548 s
Press any key to continue.