Assignment

- **Briefly describe the about the hypothetical scenario you have developed for the system of the selected topic.**

  The system (Online Transport Booking) is used by 4 types of different users which includes the users Customer/User, OTB Manager, Database Administrator, and System Administrator. Here the main user been the customer who has to register themselves in order to use the system. A registered user has attributes such as **UserID**, Name (including first name, last name), **NIC** (which is used as the primary key), address and email and is able to various things which includes selecting convenient packages, payment details, and the relevant Bus/Train details. The different packages also have a unique ID (**PID**) while the payment details too are identified uniquely by its unique ID (**PayID**). The System Administrator is occupied with managing the users, monitoring traffic and updating the system while the Database Administrator is allocated with backing up the system. Last but no least the OTB Manager is occupied with checking confirmed transactions and most importantly informing the relevant parties of the confirmed booking.

- **Identify the main requirements for the system and develop a draft Requirements Analysis document.**

  - Unregistered users can view in brief what are the packages available, the different payment methods, destinations, times, dates, sources, and the seat availability.
  - Registered users can log in to their specific account and refer to different packages available in the system.
  - They have the option to select any of these packages.
  - If they aren't interested in such packages, they can select the relevant bus/train details. (Source, destination, date, available seats, reserved seats)
  - Once selected they can select a relevant payment method available for their selected package or their selected train/bus booking.
  - Once a payment method is selected, users are prompted to add payment details.
  - If the payment is successful, the system displays a payment successful message and the details with regarding the specific booking while if the payment is unsuccessful the users are re-prompted to enter payment details.
  - The confirmed transactions are sent to the OTB Manager, where relevant accounts and detailed reports are made.
  - He informs the government and the other relevant transportation parties associated with the trains/buses.
  - OTB Manager also replies back to feedback and comments.
  - The System Administrator updates the system, manage users and monitor the on-going traffic in the system.
  - The Database Administrator maintains the constant backups in the system.

- **Data requirements to develop the Database**

The database should have 5 main tables which includes the user, Account, Payment details, Package details, and Bus/Train details tables' and then we have 2 tables for multivalued attributes and 4 more additional tables which includes the select, choose, confirm and order tables identified through the ER diagram.

The User table contains the details of the User, which includes NIC, Name, Address, and UserId.

The Account table contains the UserId, UserName and PW and is connected with the User table.

The Payment details table is connected to User, Package details, Bus/Train details but not the Account table. It contains the attributes PayId, and different payment types available.

The Package details table is connected to only User table and Payment details account. It contains PId and PName.

The Bus/Train details is connected to User table and Payment details and contains SeatId, Destination, Source, Date, ReservedSeats, and AvailableSeats.

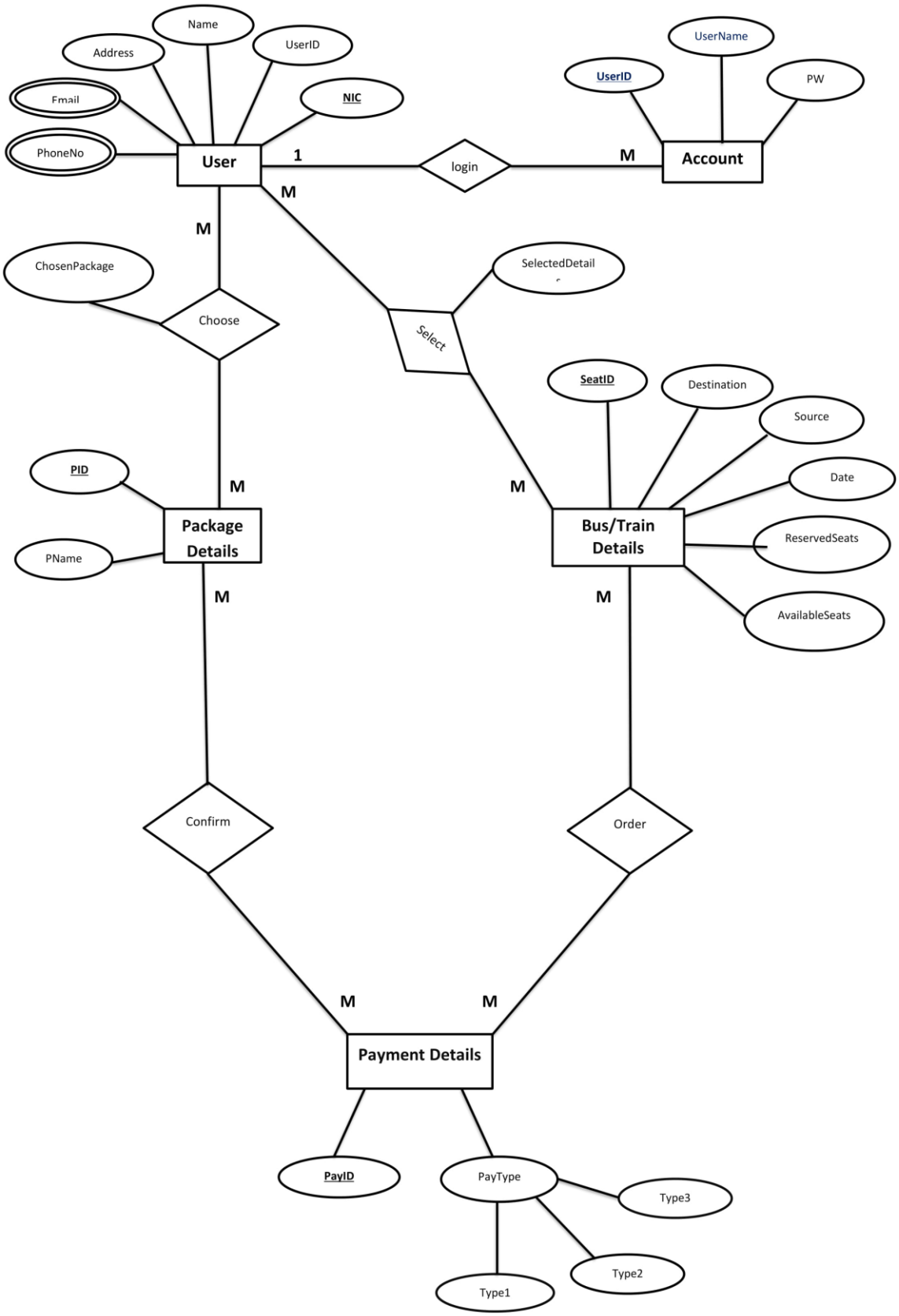
The Select table consists of NIC and SeatId.

The Choose table consists of NIC and PackageId.

The Confirm table consists of PackageId and PaymentId.

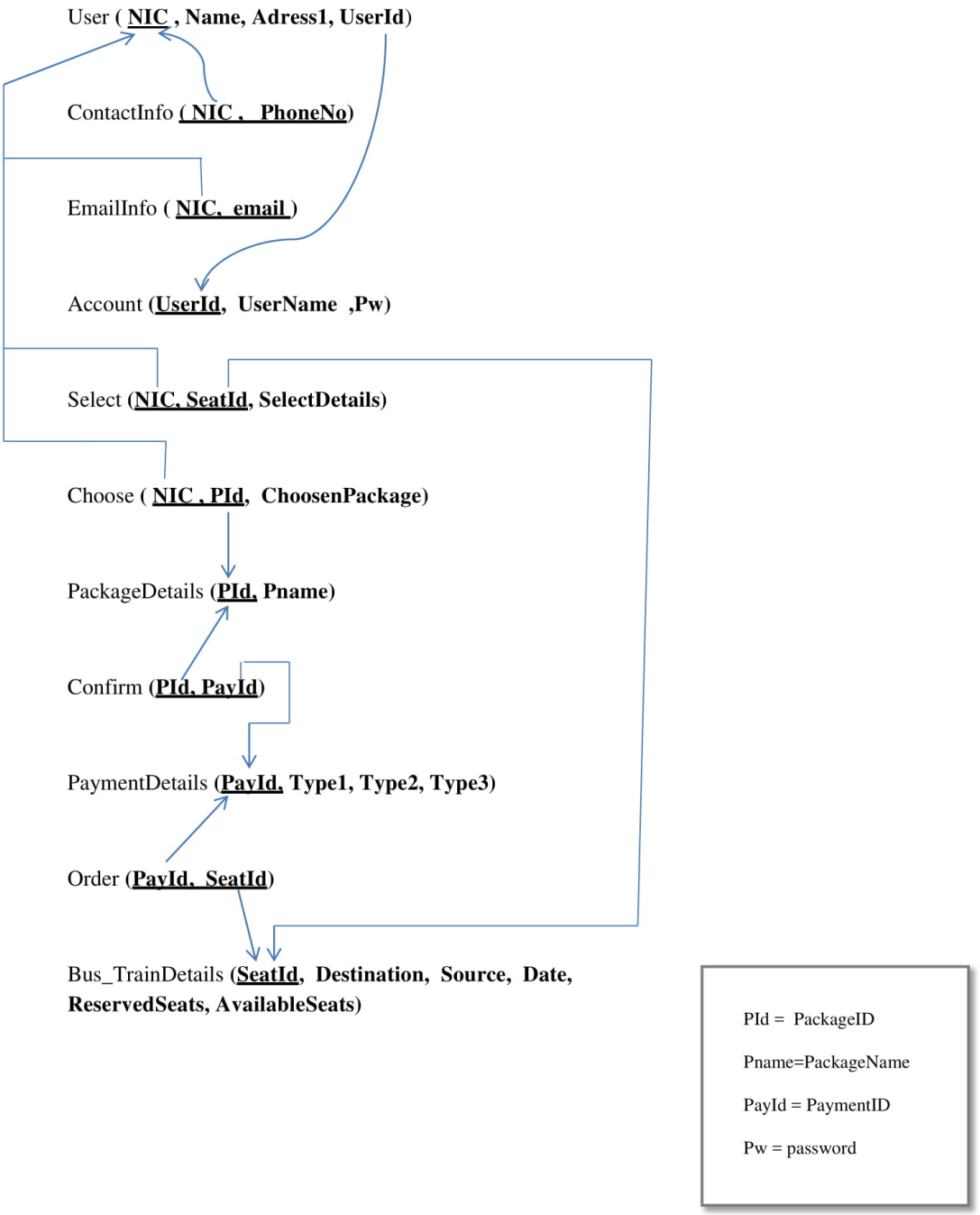
The Order table consists of PaymentId and SeatId.

- ER Diagram


Name
Address
UserID
Email
NIC
PhoneNo
User
1
login
M
Account
UserName
UserID
PW
M
M
ChosenPackage
Choose
SelectedDetail
Select
SeatID
Destination
Source
Date
ReservedSeats
AvailableSeats
PID
PName
Package Details
M
M
Bus/Train Details
M
M
Confirm
Order
M
M
Payment Details
PayID
PayType
Type3
Type2
Type1

- **Relational Data Model**

Relational Data Model



- **SQL commands to create tables**

```
----------01--User_C table-------------------------

Create table User_C
(
NIC char(10),
Name varchar(150)not null,
Address1 varchar(200),
UserId varchar(10) not null,

constraint pk_User_C primary key(NIC),
constraint ck1_NIC check (NIC like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][V-v]'),
constraint ck1_UserId check(UserId like'[A-Z][A-Z][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]')
);



INSERT INTO User_C(NIC,Name,Address1,UserId)
VALUES ('912345678V','Teena', 'Negombo','AB00001111');

INSERT INTO User_C(NIC,Name,Address1,UserId)
VALUES ('923456781V','Shainee', 'Colombo','AB00002222');

INSERT INTO User_C(NIC,Name,Address1,UserId)
VALUES ('934567821V','Qamra', 'Halawatha','AB00003333');

INSERT INTO User_C(NIC,Name,Address1,UserId)
VALUES ('945678123V','Shehan', 'Waththala','AB00004444');

INSERT INTO User_C(NIC,Name,Address1,UserId)
VALUES ('956781234V','Abimani', 'Negombo','AB00005555');
```

```
select * from User_C;

------------------02---PackageDetails_C table-------------------------

Create table PackageDetails_C
(
PId varchar(8),
Pname varchar(100) not null,
constraint pk_PackageDetails_C primary key (PId),
constraint ck_PackageDetails_C check(PId like'[P-p][A-a][C-c][K-k][0-9][0-9][0-9][0-9]')
);

INSERT INTO PackageDetails_C(PId,Pname)
VALUES ('pack0012','Special_Bus');

INSERT INTO PackageDetails_C(PId,Pname)
VALUES ('pack0013','Special_Train');

INSERT INTO PackageDetails_C(PId,Pname)
VALUES ('pack0014','Family tour');

INSERT INTO PackageDetails_C(PId,Pname)
VALUES ('pack0015','Normal tour');

INSERT INTO PackageDetails_C(PId,Pname)
VALUES ('pack0022','One day tour');

select * from PackageDetails_C;

----------------------03----PaymentDetails_C----------------------------
```

```
Create table PaymentDetails_C
(
PayId varchar(5),
Type1 varchar(50),
Type2 varchar(50),
Type3 varchar(50),

constraint pk_PaymentDetails_C primary key (PayId),
constraint ck_PaymentDetails_C check(PayId like'[P-p][A-a][Y-y][0-9][0-9]')
);

INSERT INTO PaymentDetails_C(PayId,Type1,Type2,Type3)
VALUES ('pay01','Cash','Credit card','');

INSERT INTO PaymentDetails_C(PayId,Type1,Type2,Type3)
VALUES ('pay02',' ','Credit card','eZ-cash');


INSERT INTO PaymentDetails_C(PayId,Type1,Type2,Type3)
VALUES ('pay03','Cash','Credit card','');

INSERT INTO PaymentDetails_C(PayId,Type1,Type2,Type3)
VALUES ('pay04','Cash',' ','eZ-cash');

INSERT INTO PaymentDetails_C(PayId,Type1,Type2,Type3)
VALUES ('pay05','Cash','Credit card','eZ-cash');

select * from PaymentDetails_C;

-------------------------04----ContactInfo_C-------------------------

Create table ContactInfo_C
```

```
(
NIC char(10),
PhoneNo char(10),

constraint pk_ContactInfo_C primary key (NIC,PhoneNo),
constraint fk_ContactInfo_C foreign key (NIC) references User_C(NIC),
constraint ck_ContactInfo_C check(PhoneNo like'[0-0][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]'),
);

INSERT INTO ContactInfo_C(NIC,PhoneNo)
VALUES ('912345678V','0112233445');

INSERT INTO ContactInfo_C(NIC,PhoneNo)
VALUES ('923456781V','0123456785');

INSERT INTO ContactInfo_C(NIC,PhoneNo)
VALUES ('934567821V','0111223345');


INSERT INTO ContactInfo_C(NIC,PhoneNo)
VALUES ('945678123V','0112233445');

INSERT INTO ContactInfo_C(NIC,PhoneNo)
VALUES ('956781234V','0118333445');


select* from ContactInfo_C;
drop table ContactInfo_C;
----------------------------05---EmailInfo_C-----------------------------

Create table EmailInfo_C
(
```

```
NIC char(10),
Email varchar(50),

constraint pk_EmailInfo_C primary key (NIC, Email),
constraint fk_EmailInfo_C foreign key (NIC) references User_C(NIC)
);

INSERT INTO EmailInfo_C(NIC,Email)
VALUES ('912345678V','teena123@gmail.com');

INSERT INTO EmailInfo_C(NIC,Email)
VALUES ('923456781V','shainee123@gmail.com');

INSERT INTO EmailInfo_C(NIC,Email)
VALUES ('934567821V','qamra321@gmail.com');

INSERT INTO EmailInfo_C(NIC,Email)
VALUES ('945678123V','shehan321@gmail.com');


INSERT INTO EmailInfo_C(NIC,Email)
VALUES ('956781234V','abimani104@gmail.com');

select * from EmailInfo_C;
drop table EmailInfo_C;
--------------------------06---Bus_TrainDetails_C  ------------------------------------

Create table Bus_TrainDetails_C
(
SeatId varchar(9),
Destination varchar(100) not null,
Source1 varchar(50),
```

```
Date1 date not null,
ReservedSeats int,
AvailableSeats int,

constraint pk_Bus_TrainDetails_C primary key (SeatId),
constraint ck_Bus_TrainDetails_C check(SeatId like'[S-s][E-e][A-a][T-t][B-T][0-9][0-9][0-9][0-9]')
);

INSERT INTO Bus_TrainDetails_C(SeatId,Destination,Source1,Date1,ReservedSeats,AvailableSeats)
VALUES ('SeatT0011','Kollupitiya to Negombo',' Train','2018-10-09',101,70);

INSERT INTO Bus_TrainDetails_C(SeatId,Destination,Source1,Date1,ReservedSeats,AvailableSeats)
VALUES ('SeatB0008','Colombo Fort to Kandy',' Bus','2018-10-12',10,50);

INSERT INTO Bus_TrainDetails_C(SeatId,Destination,Source1,Date1,ReservedSeats,AvailableSeats)
VALUES ('SeatT0055','Colombo to Galle',' Train','2018-10-15',120,50);


INSERT INTO Bus_TrainDetails_C(SeatId,Destination,Source1,Date1,ReservedSeats,AvailableSeats)
VALUES ('SeatB0701','Colombo to Galle',' L-Bus','2018-10-22',41,25);

INSERT INTO Bus_TrainDetails_C(SeatId,Destination,Source1,Date1,ReservedSeats,AvailableSeats)
VALUES ('SeatT3001','Kollupitiya to Chilaw',' Train','2018-11-04',181,55);

INSERT INTO Bus_TrainDetails_C(SeatId,Destination,Source1,Date1,ReservedSeats,AvailableSeats)
VALUES ('SeatB3001','Kollupitiya to Jaffna',' L-Bus','2018-11-27',70,15);

select * from Bus_TrainDetails_C;

--------------------------------07---Account_B-----------------------------------

Create table Account_C
```

```
(
UserId varchar(10),
UserName varchar(200),
Pw varchar(5),
constraint pk_Account_C primary key (UserId),
);

INSERT INTO Account_C(UserId,UserName,Pw)
VALUES ('AB00001111','Teena','gc521');

INSERT INTO Account_C(UserId,UserName,Pw)
VALUES ('AB00002222','Shainee','vhr52');

INSERT INTO Account_C(UserId,UserName,Pw)
VALUES ('AB00003333','Qamra','30abc');

INSERT INTO Account_C(UserId,UserName,Pw)
VALUES ('AB00004444','Shehan','saj49');

INSERT INTO Account_C(UserId,UserName,Pw)
VALUES ('AB00005555','Abimani','12345');

select * from Account_C;
drop table Account_C;
----------------------------------08---choose_C-----------------------------------------

Create table choose_C
(
NIC char(10),
PId varchar(8),
ChoosenPackage varchar(100) not null,
```

```
constraint pk_choose_C primary key (NIC, PId),
constraint fk_choose_C_1 foreign key (NIC) references User_C(NIC),
constraint fk_choose_C_2 foreign key (PId) references PackageDetails_C(PId),
);

INSERT INTO choose_C(NIC,PId,ChoosenPackage)
VALUES ('956781234V','pack0022','Package05');

INSERT INTO choose_C(NIC,PId,ChoosenPackage)
VALUES ('912345678V','pack0015','Package04');

INSERT INTO choose_C(NIC,PId,ChoosenPackage)
VALUES ('956781234V','pack0014','Package03');

INSERT INTO choose_C(NIC,PId,ChoosenPackage)
VALUES ('934567821V','pack0013','Package01');


INSERT INTO choose_C(NIC,PId,ChoosenPackage)
VALUES ('934567821V','pack0012','Package05');

select * from choose_C;
drop table choose_C;
-----------------------------------09---SelectBT_C-------------------------------------

Create table SelectBT_C
(
NIC char(10),
SeatId varchar(9),
SelectedDetails varchar(150),

constraint pk_SelectBT_C primary key (NIC, SeatId),
```

```
constraint fk_SelectBT_C_1 foreign key (NIC) references User_C(NIC),
constraint fk_SelectBT_C_2 foreign key (SeatId) references Bus_TrainDetails_C(SeatId),
);

INSERT INTO SelectBT_C(NIC,SeatId,SelectedDetails)
VALUES ('912345678V','SeatT0011','pack0012 and pay05');

INSERT INTO SelectBT_C(NIC,SeatId,SelectedDetails)
VALUES ('923456781V','SeatB0008','pack0022 and pay01');

INSERT INTO SelectBT_C(NIC,SeatId,SelectedDetails)
VALUES ('934567821V','SeatB3001','Colombo to Galle and pay03');

INSERT INTO SelectBT_C(NIC,SeatId,SelectedDetails)
VALUES ('945678123V','SeatB0701','package14');



INSERT INTO SelectBT_C(NIC,SeatId,SelectedDetails)
VALUES ('956781234V','SeatT0055','Kollupitiya to Negombo');

select * from SelectBT_C;

----------------------------------10---confirm_C---------------------------------------------

Create table confirm_C
(
PId varchar(8),
PayId varchar(5),

constraint pk_confirm_C primary key (PId, PayId),
constraint fk_confirm_C_1 foreign key (PId) references PackageDetails_C(PId),
```

```
constraint fk_confirm_C_2 foreign key (PayId) references PaymentDetails_C(payId),
);

INSERT INTO confirm_C(PId,payId)
VALUES ('pack0022','pay01');

INSERT INTO confirm_C(PId,payId)
VALUES ('pack0013','pay02');

INSERT INTO confirm_C(PId,payId)
VALUES ('pack0012','pay05');

INSERT INTO confirm_C(PId,payId)
VALUES ('pack0015','pay03');


INSERT INTO confirm_C(PId,payId)
VALUES ('pack0022','pay03');
select * from confirm_C;

---------------------------------11---order_C---------------------------------------

Create table order_C
(
PayId varchar(5),
SeatId varchar(9),

constraint pk_order_C primary key (PayId, SeatId),
constraint fk_order_C_1 foreign key (PayId) references PaymentDetails_C(PayId),
constraint fk_order_C_2 foreign key (SeatId) references Bus_TrainDetails_C (SeatId),
);
```

```
INSERT INTO order_C(payId,SeatId)
VALUES ('pay03','SeatB0008');

INSERT INTO order_C(payId,SeatId)
VALUES ('pay04','SeatB3001');

INSERT INTO order_C(payId,SeatId)
VALUES ('pay04','SeatB0701');

INSERT INTO order_C(payId,SeatId)
VALUES ('pay05','SeatT0055');

INSERT INTO order_C(payId,SeatId)
VALUES ('pay03','SeatT0011');

select * from order_C;
```

- Special performance considerations for the system

    1. Workload

       The system should be able to support and withstand more than 11,000 users at a time and should be also able to acquire the given information and support concurrently for the requests made by users simultaneously. The system should be able to match the specific user needs and supply it to the user.

    2. Efficient forms to add data accurately and timely to the database

- ER Model derived from the Schema of the Database according to the Relational Data Model