IE 7374 USE CASE STUDY REPORT

Group 2: Raja Muthu, Sowmya Shree Nagaraju

Executive Summary:

The primary objective of this study was to design and implement a relational database that is industry ready for application in the Airline industry for use by the properties (Employee & Customers) for better experience. Currently, the AirCare company runs only domestic services inside India. The users complained about the efficiency of the system which is time consuming and the data consistency was poor. Moreover, users were not easily able to book their tickets without the help of agency. This relational database reduces data input process time by 50% and result in huge cost saving benefits for the company who is planning on expanding their services to international flights. The database also implements a central analytics platform that has immense potential for analytics in the Airline industry, preferred air routes of travel, statistics and also maintain a tab on the employees.

The database was modelled taking requirements of data fields required by the AirCare company which has also taken feedbacks from current customers of their services and the employees. The EER and UML diagrams were modelled, followed by the mapping of the conceptual model to a relational model with the required primary and foreign keys. This database was then implemented fully using MySQL and a prototype with few tables and relationships were implemented on Neo4j NoSQL graph database to study the feasibility of the database in a NoSQL environment.

The created database is a great success, and by connecting it to R and Tableau, the analytics capabilities are immense, some of which have been shown in the study. These queries can also be helpful in fast tracking the customers and their travel plans.

I. Introduction

AirCare is a small airline company started in 1988 to facilitate domestic travel inside India. The airline was designed to handle small set of customers and today they are thinking to expand their business and facilitate thousands of passengers on a daily basis. AirCare still uses manual ticket booking system which allows only agents to book the flight tickets as in when the customers call to the agency. Customer data and flight details are kept in files and company uses inefficient system which slows down while using. Now the company wishes to transform to modern technology with flexible reservation and inventory management solutions including call Centre, travel agency, internet engine, global distribution systems.

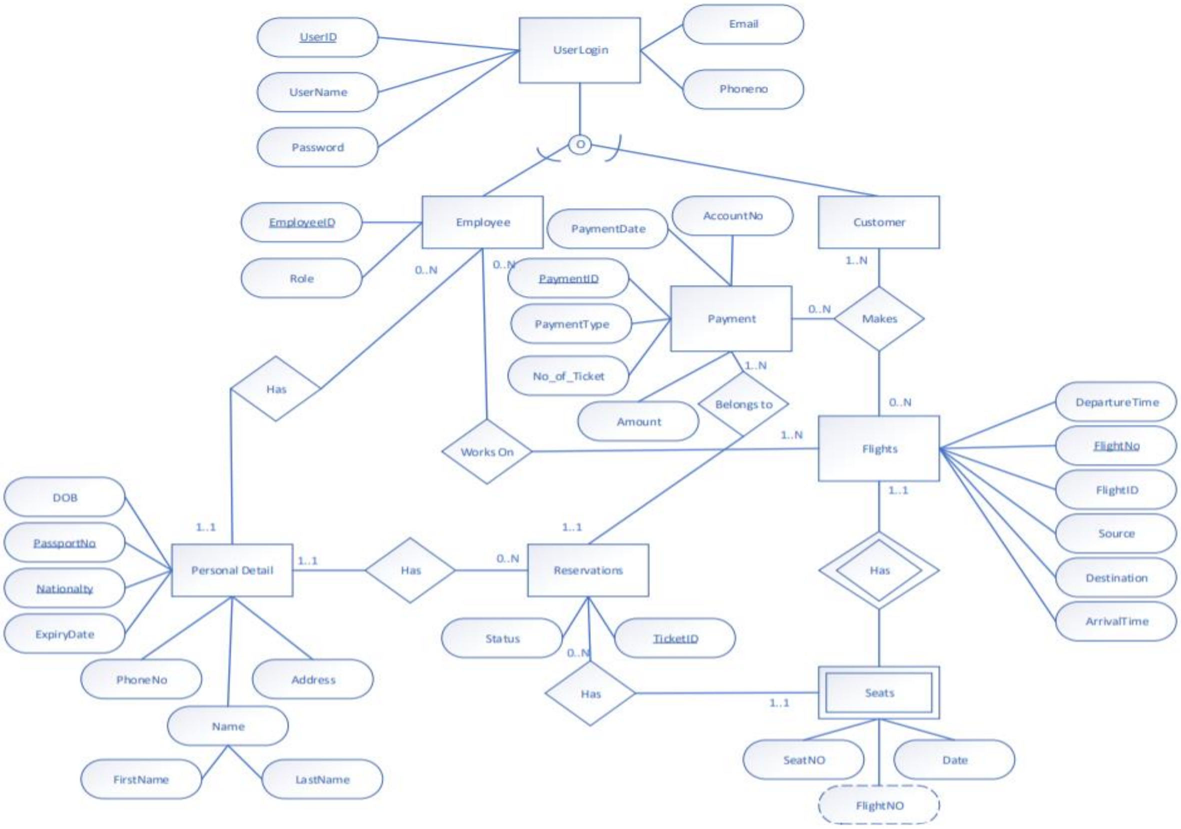
The current system is manual and slow, time consuming and it is very difficult for each person to book through office agents. Users inquire about the tickets through phone calls and it is very difficult for the customers to remember all the details that they received through phones. It is very difficult to calculate how many peoples registered and how many seats on a particular plane are vacant. This requires quite a lot of time and wastage of money as it requires quite lot of man power to do.

Hence AirCare wants to have a centralized database system for their data which can be accessed easily anytime, anywhere. This is where relational database comes into picture. Having a relational database for recording all the data of international customers and their details would solve this problem of Data duplication and can shorten the process time by over 50% as the details of the customer need not be filled every time they book the ticket unless the customer details are changed.

Another addition to the database is the employee entity that is compositely aggregated. This given an opportunity for the company to track which employee is working in which flight and which employee is responsible in registering the customer if the customer has booked the ticket via phone call.
Requirements:
Flights travel in the same route designated each day.

II. Conceptual Data Modeling

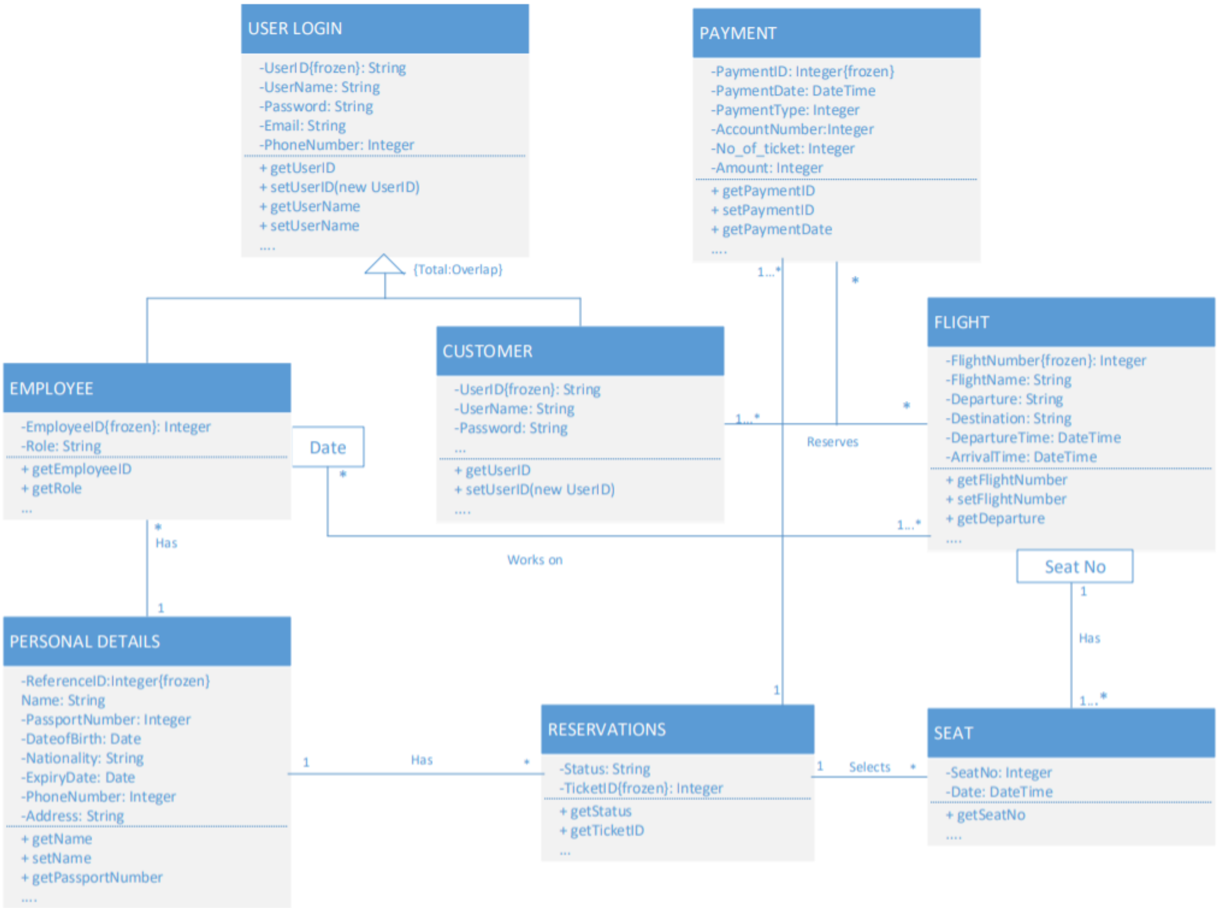
1. EER Diagram



Some of the Additional conditions which couldn’t be mapped in EER model are as follow

- A customer can reserve maximum 6 seats in one booking
- A flight will have minimum of 6 employees
- An employee works in the flight cannot be a customer at the same time

2. UML Diagram



III. Mapping Conceptual Model to Relational Model

The company plans to store all the personal details such as Name, Nationality, Passport NO, DOB of all the employees and people who are travelling in the airline. The data is stored in the system so when the customer wants to book a ticket next time it can be auto filled from the system unless there is a major change in the detail.

PERSONALDETAILS (REFRENCEID, FIRSTNAME, LASTNAME, DOB, NATIONALITY, EXPIRYDATE)
Here the primary REFRENCEID is modelled as a self-generated autoincremented value which is unique for each customer or employee detail.

USERS(USERID, USERNAME, USERPASSWORD, EMAIL, PHONENUMBER)
USERID is a unique value assigned to individual user.

FLIGHTS(FLIGHTNO, FLIGHTNAME, *SOURCE*, *DESTINATION*, DEPARTURETIME, ARRIVALTIME)
FOREIGN KEY SOURCE and DESTINATION refers to the code of Airports in AIRPORTS table.

RESERVATIONS(TICKETID, *REFERENCEID*, *PAYMENTID*,*FLIGHTNO*, DATE, SEATNO, STATUS)
TICKETID is the primary key and FOREIGN KEYs refers to REFERENCEID, PAYMENTID,FLIGHTNO in PERSONALDETAILS, PAYMENT and FLIGHTS table respectively.

PAYMENT(PAYMENTID, PAYMENTDATE, PAYMENTTYPE, AMOUNT, NOOFTICKETS, ACCOUNTNUMBER, *USERID*)
PAYMENTID is a system generated value for each transaction performed by different user.

EMPLOYEE(EMPLOYEEID, *REFRENCEID*, ROLE, *USERID*)
This table holds the Employee job function along with their unique EMPLOYEEID. The Personal details of the employee are stored in PERSONALDETAILS and referred here by making use of the REFRENCEID value.

WORKSON(*EMPLOYEEID*, FLIGHTNO, DATE)
AIRPORTS(AIRPORT, CODE)

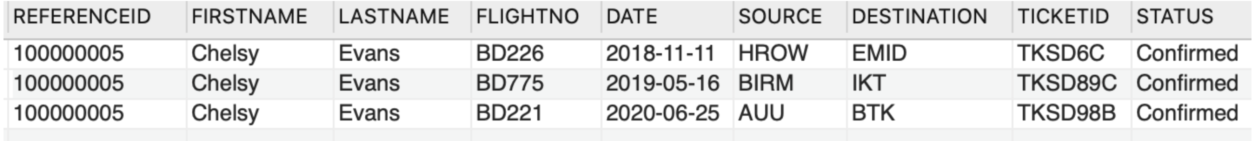
IV. Implementation of Relational Model via MySQL and NoSQL

MySQL Implementation:

The database was created in MySQL and the following queries were performed:

Query 1: Find the flight travel details of the customer with the ReferenceID as ‘100000005’.

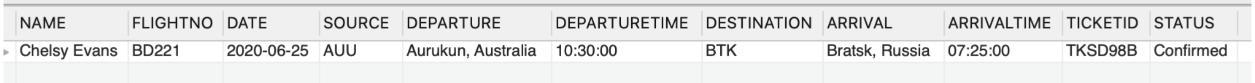
```
SELECT S.REFERENCEID, S.FIRSTNAME, S.LASTNAME, R.FLIGHTNO, R.DATE,
F.SOURCE, F.DESTINATION, R.TICKETID, R.STATUS FROM PERSONALDETAILS S
INNER JOIN RESERVATIONS R ON R.REFERENCEID = S.REFERENCEID
INNER JOIN FLIGHTS F ON F.FLIGHTNO = R.FLIGHTNO
WHERE EXISTS  (SELECT * FROM PERSONALDETAILS WHERE S.FIRSTNAME='Chelsy' AND
S.LASTNAME='Evans' AND S.PASSPORTNO='F10326771');
```



REFERENCEID	FIRSTNAME	LASTNAME	FLIGHTNO	DATE	SOURCE	DESTINATION	TICKETID	STATUS
100000005	Chelsy	Evans	BD226	2018-11-11	HROW	EMID	TKSD6C	Confirmed
100000005	Chelsy	Evans	BD775	2019-05-16	BIRM	IKT	TKSD89C	Confirmed
100000005	Chelsy	Evans	BD221	2020-06-25	AUU	BTK	TKSD98B	Confirmed

Query 2: Find the scheduled travel plans of the same above customer.

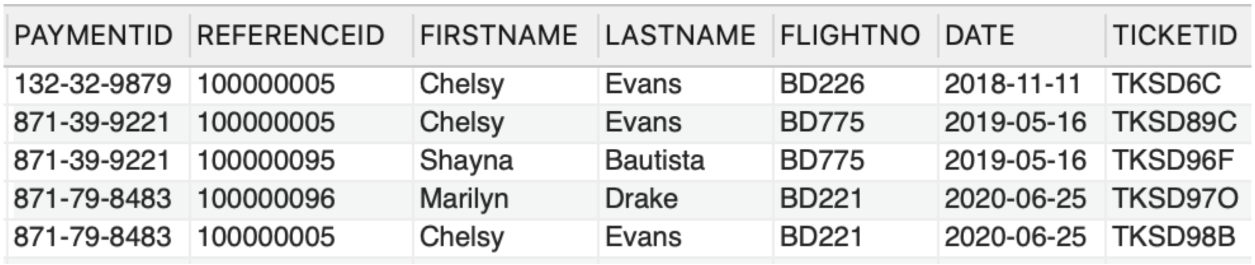
```
SELECT CONCAT(S.FIRSTNAME,' ',S.LASTNAME) AS NAME, R.FLIGHTNO, R.DATE,
F.SOURCE, A1.AIRPORT AS DEPARTURE, F.DEPARTURETIME,
F.DESTINATION, A2.AIRPORT AS ARRIVAL, F.ARRIVALTIME, R.TICKETID,
R.STATUS FROM PERSONALDETAILS S
INNER JOIN RESERVATIONS R ON R.REFERENCEID = S.REFERENCEID
INNER JOIN FLIGHTS F ON F.FLIGHTNO = R.FLIGHTNO
INNER JOIN AIRPORTDETAILS A1 ON A1.CODE = F.SOURCE
INNER JOIN AIRPORTDETAILS A2 ON A2.CODE = F.DESTINATION
WHERE S.REFERENCEID = '100000005' AND
            R.DATE >= CURDATE();
```



NAME	FLIGHTNO	DATE	SOURCE	DEPARTURE	DEPARTURETIME	DESTINATION	ARRIVAL	ARRIVALTIME	TICKETID	STATUS
Chelsy Evans	BD221	2020-06-25	AUU	Aurukun, Australia	10:30:00	BTK	Bratsk, Russia	07:25:00	TKSD98B	Confirmed

Query 3: Find the details of the payment and if any other tickets booked along with his/her bookings of the above customer.

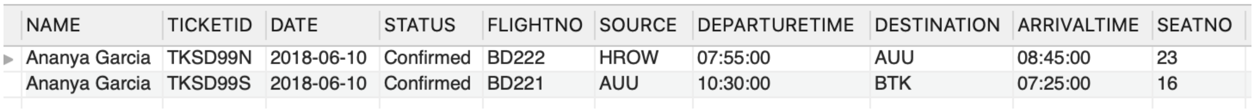
```
SELECT R.PAYMENTID, R.REFERENCEID, S.FIRSTNAME, S.LASTNAME,
R.FLIGHTNO, R.DATE, R.TICKETID FROM RESERVATIONS R
INNER JOIN PERSONALDETAILS S ON S.REFERENCEID = R.REFERENCEID
WHERE EXISTS (SELECT * FROM RESERVATIONS R1 WHERE R1.PAYMENTID =
R.PAYMENTID AND R1.REFERENCEID = '100000005');
```



PAYMENTID	REFERENCEID	FIRSTNAME	LASTNAME	FLIGHTNO	DATE	TICKETID
132-32-9879	100000005	Chelsy	Evans	BD226	2018-11-11	TKSD6C
871-39-9221	100000005	Chelsy	Evans	BD775	2019-05-16	TKSD89C
871-39-9221	100000095	Shayna	Bautista	BD775	2019-05-16	TKSD96F
871-79-8483	100000096	Marilyn	Drake	BD221	2020-06-25	TKSD97O
871-79-8483	100000005	Chelsy	Evans	BD221	2020-06-25	TKSD98B

Query 4: Find the connecting flight details of the customer with ticket ID TKSD99N.

```
SELECT DISTINCT CONCAT(S.FIRSTNAME,' ',S.LASTNAME) AS NAME,
 R.TICKETID,R.DATE,R.STATUS,R.FLIGHTNO,
 (SELECT AIRPORT FROM AIRPORTDETAILS WHERE CODE=F.SOURCE) AS SOURCE,
F.DEPARTURETIME, (SELECT AIRPORT FROM AIRPORTDETAILS WHERE
CODE=F.DESTINATION) AS DESTINATION, F.ARRIVALTIME, R.SEATNO FROM
PERSONALDETAILS S INNER JOIN RESERVATIONS R ON R.REFERENCEID =
S.REFERENCEID INNER JOIN FLIGHTS F ON F.FLIGHTNO = R.FLIGHTNO
INNER JOIN AIRPORTDETAILS FD ON FD.CODE=F.SOURCE OR FD.CODE=F.DESTINATION
WHERE S.REFERENCEID = (SELECT REFERENCEID FROM RESERVATIONS R1
 WHERE R1.TICKETID = ' TKSD99N ') AND
EXISTS (SELECT * FROM RESERVATIONS R1
 WHERE R1.PAYMENTID = R.PAYMENTID) ORDER BY TICKETID;
```

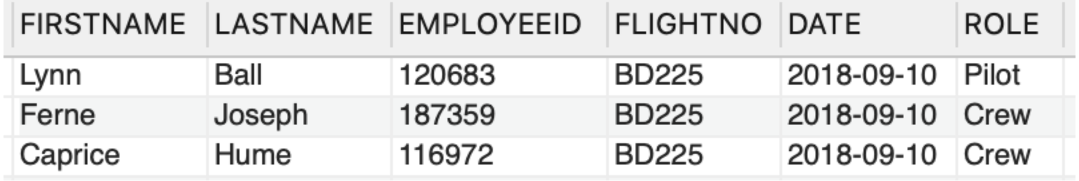


NAME	TICKETID	DATE	STATUS	FLIGHTNO	SOURCE	DEPARTURETIME	DESTINATION	ARRIVALTIME	SEATNO
Ananya Garcia	TKSD99N	2018-06-10	Confirmed	BD222	HROW	07:55:00	AUU	08:45:00	23
Ananya Garcia	TKSD99S	2018-06-10	Confirmed	BD221	AUU	10:30:00	BTK	07:25:00	16

Query 5 Get the list of employees working in the flight BD225 on Sep 10th, 2018.

```
SELECT S.FIRSTNAME,S.LASTNAME,W.EMPLOYEEID,W.FLIGHTNO,W.DATE,E.ROLE
FROM WORKSON W
```

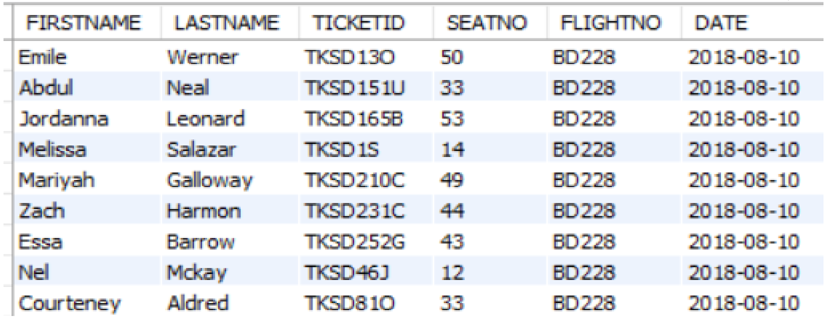
```
INNER JOIN EMPLOYESS E ON E.EMPLOYEEID = W.EMPLOYEEID
INNER JOIN PERSONALDETAILS S ON S.REFERENCEID = E.REFERENCEID
WHERE W.FLIGHTNO = 'BD225' AND
            W.DATE = '2018-09-10';
```



FIRSTNAME	LASTNAME	EMPLOYEEID	FLIGHTNO	DATE	ROLE
Lynn	Ball	120683	BD225	2018-09-10	Pilot
Ferne	Joseph	187359	BD225	2018-09-10	Crew
Caprice	Hume	116972	BD225	2018-09-10	Crew

Query 6 Get the list of Passenger Name, Seat no, Flightno, Travel date of passengers who have made a reservation on flight BD228 on Aug 10th 2018.

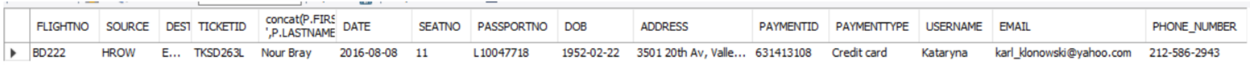
```
SELECT S.FIRSTNAME,S.LASTNAME, R.TICKETID, R.SEATNO, F.FLIGHTNO, R.DATE
FROM PERSONALDETAILS S
INNER JOIN RESERVATIONS R ON S.REFERENCEID=R.REFERENCEID
INNER JOIN FLIGHTS F ON R.FLIGHTNO=F.FLIGHTNO
WHERE R.FLIGHTNO='BD228' AND DATE='2018-08-10';
```



FIRSTNAME	LASTNAME	TICKETID	SEATNO	FLIGHTNO	DATE
Emile	Werner	TKSD13O	50	BD228	2018-08-10
Abdul	Neal	TKSD151U	33	BD228	2018-08-10
Jordanna	Leonard	TKSD165B	53	BD228	2018-08-10
Melissa	Salazar	TKSD1S	14	BD228	2018-08-10
Mariyah	Galloway	TKSD210C	49	BD228	2018-08-10
Zach	Harmon	TKSD231C	44	BD228	2018-08-10
Essa	Barrow	TKSD252G	43	BD228	2018-08-10
Nel	Mckay	TKSD46J	12	BD228	2018-08-10
Courteney	Aldred	TKSD81O	33	BD228	2018-08-10

Query 7 Get the flight details to travel from Merida to Heathrow.

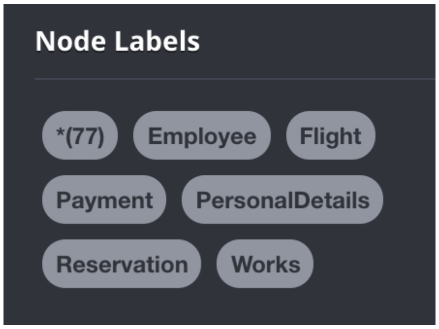
```
SELECT F.FLIGHTNO,F.SOURCE,F.DESTINATION,R.TICKETID,
CONCAT(P.FIRSTNAME, ' ', P.LASTNAME),R.DATE,R.SEATNO,P.PASSPORTNO,
P.DOB,P.ADDRESS, PA.PAYMENTID,PA.PAYMENTTYPE,
U.USERNAME,U.EMAIL,U.PHONE_NUMBER
FROM FLIGHTS F INNER JOIN RESERVATIONS R ON F.FLIGHTNO=R.FLIGHTNO
INNER JOIN PERSONALDETAILS P ON R.REFERENCEID=P.REFERENCEID
INNER JOIN PAYMENT PA ON R.PAYMENTID=PA.PAYMENTID
INNER JOIN USERS U ON PA.USERID=U.USERID
WHERE R.TICKETID='TKSD263L' ;
```

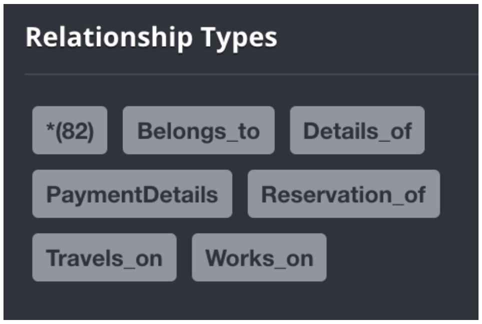


FLIGHTNO	SOURCE	DEST	TICKETID	concat(P.FIRS ',P.LASTNAME	DATE	SEATNO	PASSPORTNO	DOB	ADDRESS	PAYMENTID	PAYMENTTYPE	USERNAME	EMAIL	PHONE_NUMBER
BD222	HROW	E...	TKSD263L	Nour Bray	2016-08-08	11	L10047718	1952-02-22	3501 20th Av, Valle...	631413108	Credit card	Kataryna	karl_klonowski@yahoo.com	212-586-2943

NoSQL Implementation:

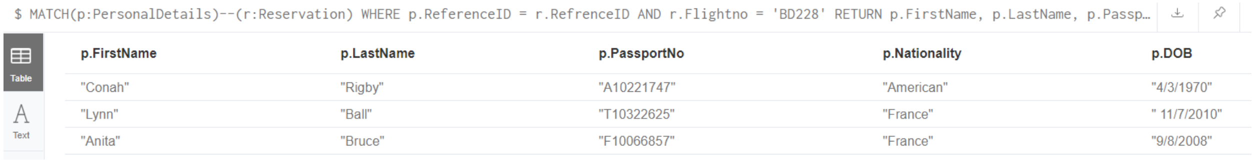
Six tables and six relationship types were created. The following Cypher queries were done.





Query 1: Get the details of the passengers traveling in the flight BD228.

```
MATCH(p:PersonalDetails)--(r:Reservation)
WHERE p.ReferenceID = r.RefrenceID AND
            r.Flightno = 'BD228'
RETURN p.FirstName, p.LastName, p.PassportNo, p.Nationality, p.DOB
```

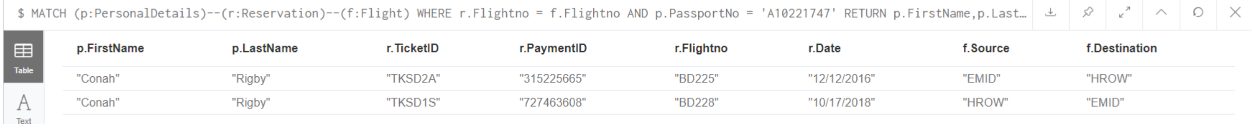


```
$ MATCH(p:PersonalDetails)--(r:Reservation) WHERE p.ReferenceID = r.RefrenceID AND r.Flightno = 'BD228' RETURN p.FirstName, p.LastName, p.Passp…
```

p.FirstName	p.LastName	p.PassportNo	p.Nationality	p.DOB
"Conah"	"Rigby"	"A10221747"	"American"	"4/3/1970"
"Lynn"	"Ball"	"T10322625"	"France"	" 11/7/2010"
"Anita"	"Bruce"	"F10066857"	"France"	"9/8/2008"

Query 2: Get all the travel details of the passenger 'Conah Rigby' with PassportNo = 'A10221747'

```
MATCH (p:PersonalDetails)--(r:Reservation)--(f:Flight)
WHERE r.Flightno = f.Flightno AND
           p.PassportNo = 'A10221747'
RETURN p.FirstName,p.LastName,r.TicketID,r.PaymentID,r.Flightno,
r.Date,f.Source,f.Destination
```

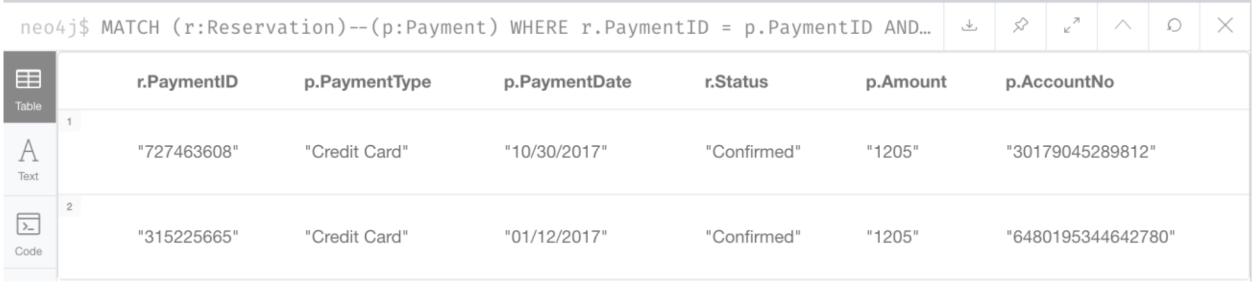


```
$ MATCH (p:PersonalDetails)--(r:Reservation)--(f:Flight) WHERE r.Flightno = f.Flightno AND p.PassportNo = 'A10221747' RETURN p.FirstName,p.Last…
```

p.FirstName	p.LastName	r.TicketID	r.PaymentID	r.Flightno	r.Date	f.Source	f.Destination
"Conah"	"Rigby"	"TKSD2A"	"315225665"	"BD225"	"12/12/2016"	"EMID"	"HROW"
"Conah"	"Rigby"	"TKSD1S"	"727463608"	"BD228"	"10/17/2018"	"HROW"	"EMID"

Query 3: Get all the payment details of the above shown passenger

```
MATCH (r:Reservation)--(p:Payment)
WHERE r.PaymentID = p.PaymentID AND
           r.RefrenceID = '100000001'
RETURN r.PaymentID,p.PaymentType,p.PaymentDate,r.Status,p.Amount,p.AccountNo
```

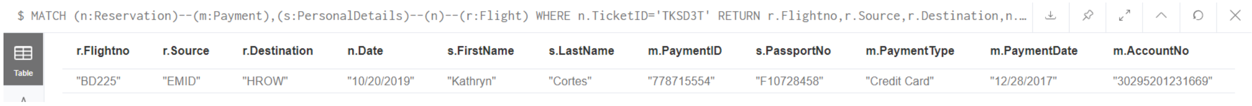


```
neo4j$ MATCH (r:Reservation)--(p:Payment) WHERE r.PaymentID = p.PaymentID AND…
```

r.PaymentID	p.PaymentType	p.PaymentDate	r.Status	p.Amount	p.AccountNo
"727463608"	"Credit Card"	"10/30/2017"	"Confirmed"	"1205"	"30179045289812"
"315225665"	"Credit Card"	"01/12/2017"	"Confirmed"	"1205"	"6480195344642780"

Query 4: Get the Flight, passenger and payment details of ticket TKSD3T

```
MATCH (n:Reservation)--(m:Payment),(s:PersonalDetails)--(n)--(r:Flight)
WHERE n.TicketID='TKSD3T'
RETURN r.Flightno,r.Source,r.Destination,n.Date,s.FirstName,s.LastName,m.PaymentID,
s.PassportNo,m.PaymentType,m.PaymentDate,m.AccountNo
```

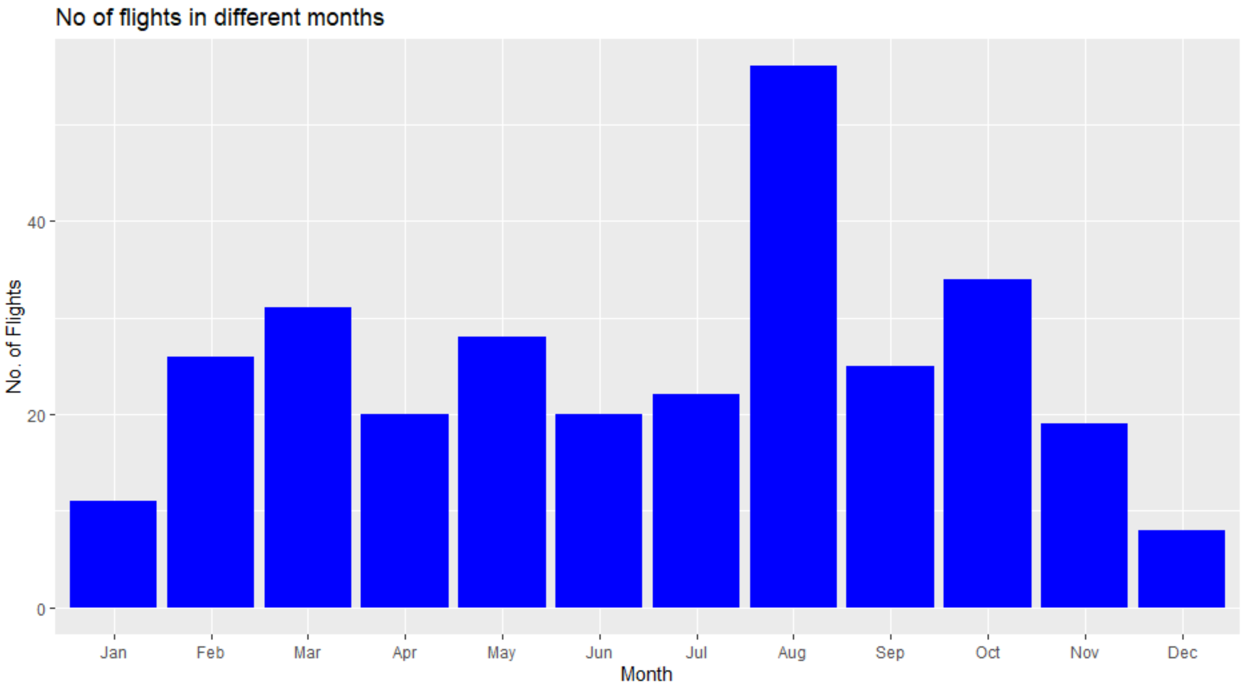


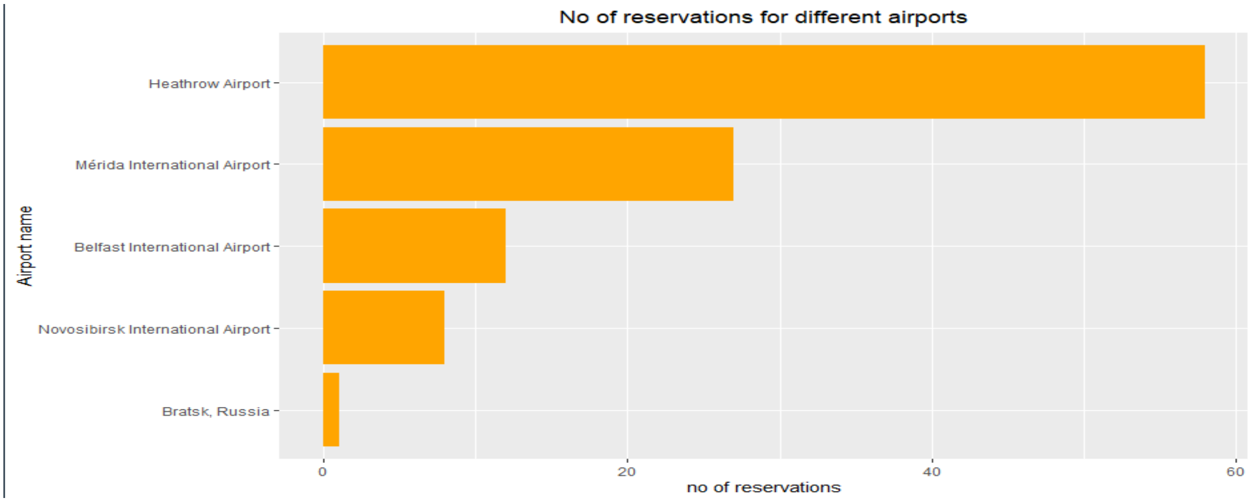
```
$ MATCH (n:Reservation)--(m:Payment),(s:PersonalDetails)--(n)--(r:Flight) WHERE n.TicketID='TKSD3T' RETURN r.Flightno,r.Source,r.Destination,n.…
```

r.Flightno	r.Source	r.Destination	n.Date	s.FirstName	s.LastName	m.PaymentID	s.PassportNo	m.PaymentType	m.PaymentDate	m.AccountNo
"BD225"	"EMID"	"HROW"	"10/20/2019"	"Kathryn"	"Cortes"	"778715554"	"F10728458"	"Credit Card"	"12/28/2017"	"30295201231669"

V. Database access via R and Tableau

The Airline database is accessed using R and the visualization of the analyzed data is shown below. The connection to MySQL database is achieved using the function dbConnect() of RMySQL library. Then the SQL query is sent using the function dbSendQuery() and the visualization is done with the help of ggplot2 library.

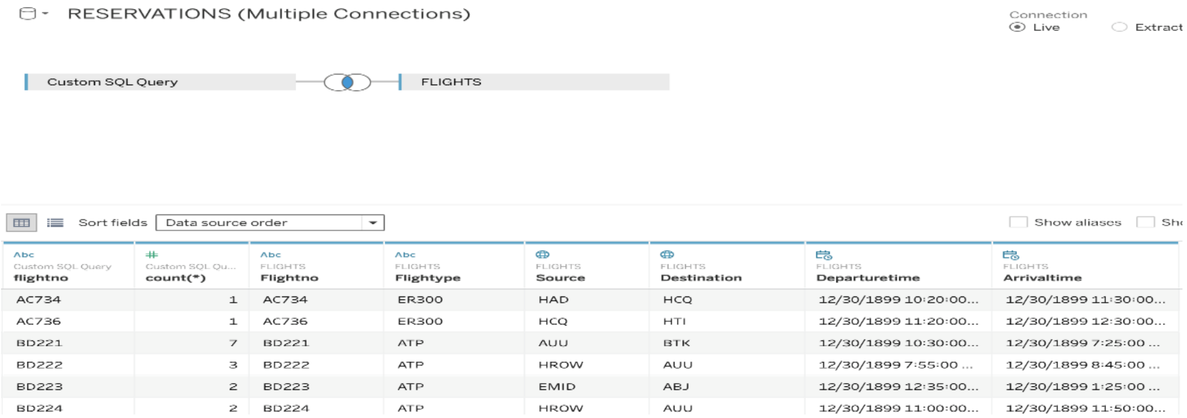



No of reservations for different airports

The above plot gives the plot number of reservations made to different airports, we can based on the observation the company can get to know the travelling trend of people by adding more constraint such as date in the plot. With this knowledge important business decision can be formulated.

Tableau:

The visualization is achieved from the Custom SQL Query along with inner join as shown below.


RESERVATIONS (Multiple Connections)

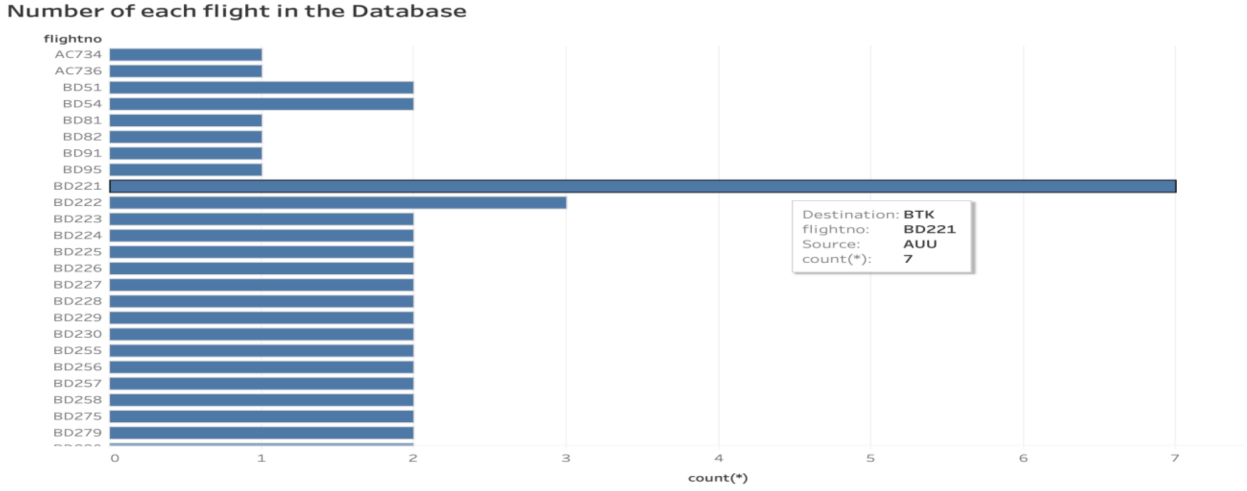
Connection ● Live ○ Extract

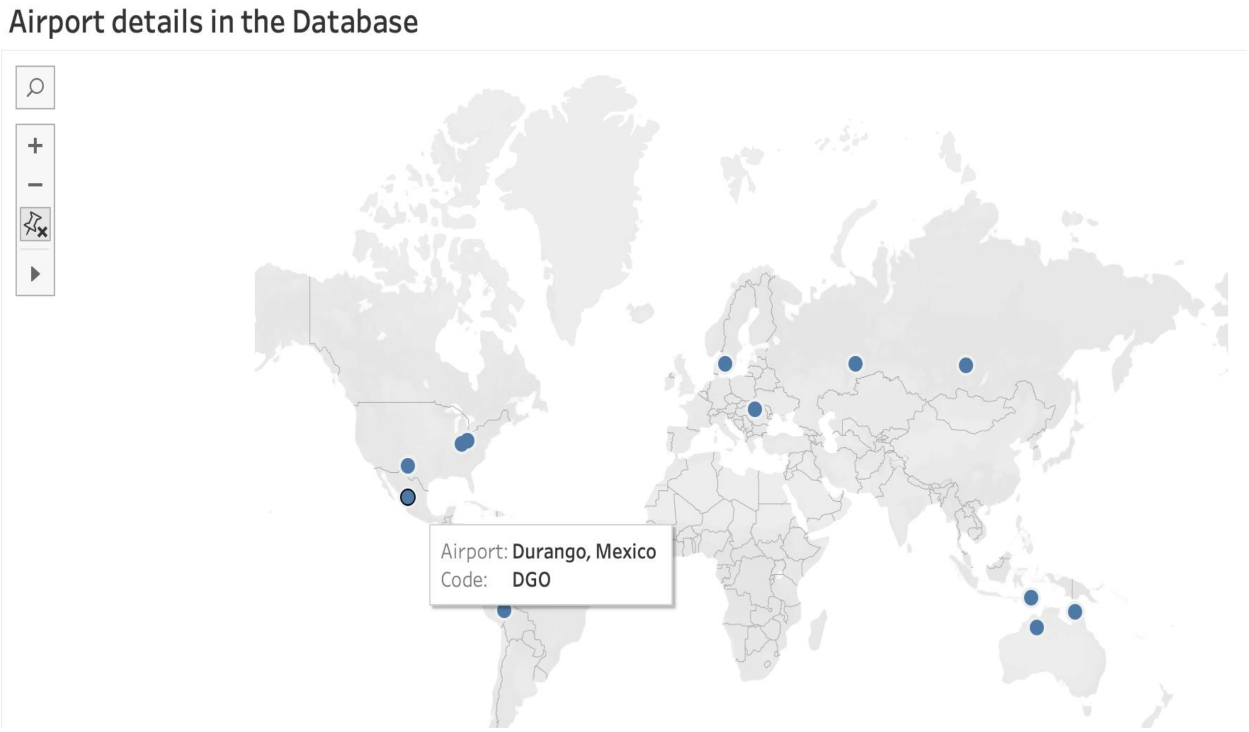
Custom SQL Query — FLIGHTS

Sort fields: Data source order

□ Show aliases □ Sh

Custom SQL Query flightno	Custom SQL Qu... count(*)	FLIGHTS Flightno	FLIGHTS Flightype	FLIGHTS Source	FLIGHTS Destination	FLIGHTS Departuretime	FLIGHTS Arrivaltime
AC734	1	AC734	ER300	HAD	HCQ	12/30/1899 10:20:00...	12/30/1899 11:30:00...
AC736	1	AC736	ER300	HCQ	HTI	12/30/1899 11:20:00...	12/30/1899 12:30:00...
BD221	7	BD221	ATP	AUU	BTK	12/30/1899 10:30:00...	12/30/1899 7:25:00 ...
BD222	3	BD222	ATP	HROW	AUU	12/30/1899 7:55:00 ...	12/30/1899 8:45:00 ...
BD223	2	BD223	ATP	EMID	ABJ	12/30/1899 12:35:00...	12/30/1899 1:25:00 ...
BD224	2	BD224	ATP	HROW	AUU	12/30/1899 11:00:00...	12/30/1899 11:50:00...


Number of each flight in the Database



Airport details in the Database

Airport: Durango, Mexico
Code: DGO

VII. Summary and Recommendation

The Airline database designed using MySQL is an industry ready relational database that can be implemented by the AirCare company. It will result in great cost savings in the data process aspect and provides great analytics capabilities, a small part of the visualization is shown in the report using R and Tableau.

- There will more than hundreds of users at any given time accessing the database who may retrieve the flight details or perform a reservation. The system must be able to handle the load of multiple inputs simultaneously.
- There will be gigs of transaction and ticket reservation details generated on a daily basis, we can proceed to build tables on a monthly basis to handle data.
- By implementing the above monthly table for tickets and payment transaction, it is possible to easily search and retrieve data.
- The employee wages, timesheet and supervisor details can be modeled into another table.

Recommendations:

We feel it’s more convenient to store certain type of data detail in NoSQL and few in SQL. The Database system can be built as a combination of both SQL and NoSQL tables.
Creating interactive dashboards would be a more feasible option to perform the business decisions to add more services etc.,