

Database Development – CW2

Module Code: 7CS082

Submitted By: 100540673

Table of Contents

1	Introduction	2
2	Justification of MySQL	2
3	Connection mechanism.....	2
4	Entity Relationship Diagram	6
5	System Design	6
5.1	Navigation map of the system.....	6
5.2	User Interface design	7
5.3	Description of the system.....	10
5.4	Trigger Procedure	16
6	Report Generation.....	17
7	Conclusion	19
8	References	20

1 Introduction

The Student Flight Management system has been developed for the airline business that would serve various purposes such as to help the business to run more efficiently in order to increase the revenue, allow managers and employees to make better decisions, allows the business to stay organized and keep track of information quickly accessible. A well-designed database is vital to any company or organization. This application has been designed in a way to be simple to use for all the customers. In the first part of the report the system was developed and implemented using MySQL as the back end. Individual authentication for the admin and customers enables more comprehensive way of the design. The purpose of choosing MS Access as the front-end is it enhances flexibility, and an excellent option that can connect with various data sources easily. Furthermore, Access server allows other databases to connect with unlimited database sizes - considering this would benefit the future growth of the business to store large number of data. This report shall elaborate the connection process between MS Access and MySQL, user interface design, operations of important features of the application are discussed and the implemented trigger procedure in the database of the system is explained.

2 Justification of MySQL

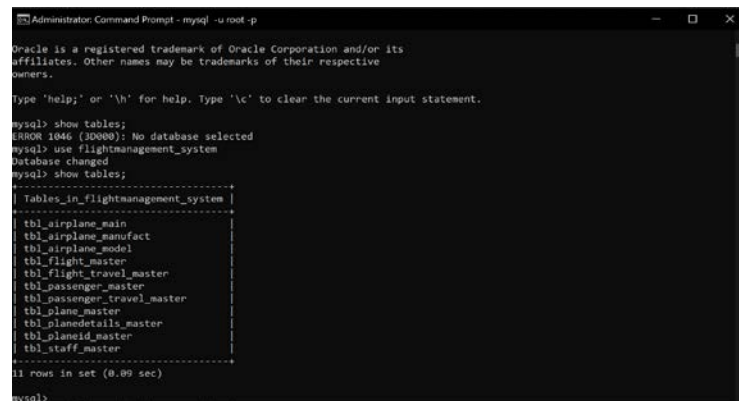
The potential reason for choosing MySQL database is because it's an open source and the company updates new features very often and considering some of the significant aspects in mind that would greatly support the business in its future growth.

- It is compatible with many operating systems such as UNIX, LINUX, Windows, Mac etc.
- Unlike other DBMS, MySQL supports various interfaces like JDBC, ODBC and other scripting languages in order to connect with the application
- The master slave replication technology of MySQL ensures high availability of database in case of failure in master database. The backup technology available in MySQL achieves full recovery of data (Ping et al, 2014).

3 Connection mechanism

To demonstrate the connection process, the WAMP SERVER 64 has been installed in the machine. Once the WAMP server is started, open the command prompt copy the path of MYSQL 8.0.21 from the bin folder then mysql will be initiated then enter the username as root and password then mysql gets connected. The database named as flightmanagement_system has been created in MySQL. In the report 1, from the raw data of the airline company there were 7 entities identified and further they are grouped as different tables with attributes containing the relevant information about each entity

which are staff, airplanes, flight and passengers. For instance, the staff table contains the information about all the staffs and the pilot's eligibility of planes they are allowed to fly. All the entities are identified, normalized and different tables were created associating the relationship between each other. The below figure shows the tables within the flightmanagement_system.



```

Administrator: Command Prompt - mysql -u root -p
mysql> show tables;
ERROR 1046 (3D000): No database selected
mysql> use flightmanagement_system
Database changed
mysql> show tables;
+-----+
| Tables_in_flightmanagement_system |
+-----+
| tbl_airplane_main                  |
| tbl_airplane_manufact              |
| tbl_airplane_model                 |
| tbl_flight_master                  |
| tbl_flight_travel_master           |
| tbl_passenger_master               |
| tbl_passenger_travel_master        |
| tbl_plane_master                   |
| tbl_planedetails_master            |
| tbl_planeid_master                 |
| tbl_staff_master                   |
+-----+
11 rows in set (0.09 sec)

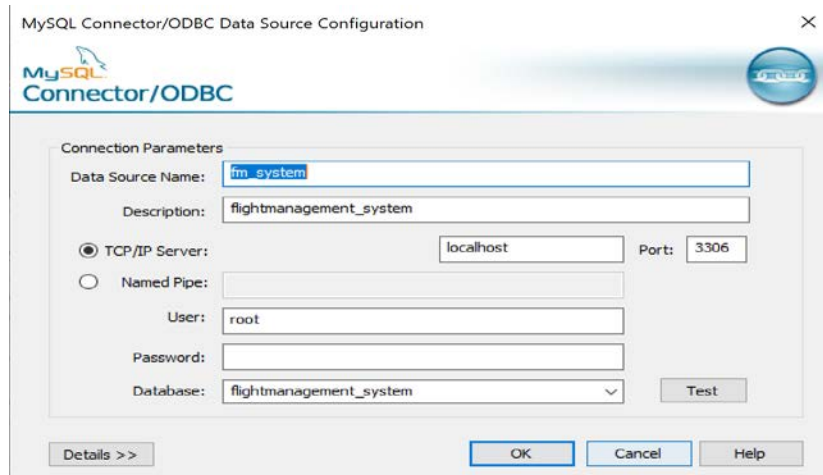
mysql>

```

Fig 1: Tables created in flightmanagement_system

In order to connect the tables with the front-end application, the Open Database Connectivity (ODBC) that permits interoperability between MS Access and MySQL has been successfully established. The step by step procedure for connection is as follows.

- Step 1: Download MySQL connector/J based on the type of operating system in the machine and visual C++ redistributable package for visual studio 2016
- Step 2: Follow the steps and install both drivers in the machine
- Step 3: Open ODBC data source -> User DSN -> Create New Data Source -> select MySQL ODBC 5.3 Unicode Driver -> click Finish
- Step 4: In MySQL Connector/ODBC Data Source Configuration wizard -> under Data Source Name -> fm_system
- The following figure shows the entries to be filled in the ODBC Data Source Connection wizard



- Step 5: The created User Data Source appears in the User DSN tab. Select fm_system and click OK.
- Step 6: Open MS Access -> create a blank database -> select External Data -> choose Link to the data source by creating a linked table -> click OK.
- Step 7: Select Data source -> select Machine data Source -> select fm_system -> press OK
- Step 8: Link Tables -> Choose flightmanagement_system in order to link the database tables and click OK.

All the tables under the flightmanagement_system database from MySQL has been imported to MS Access successfully.

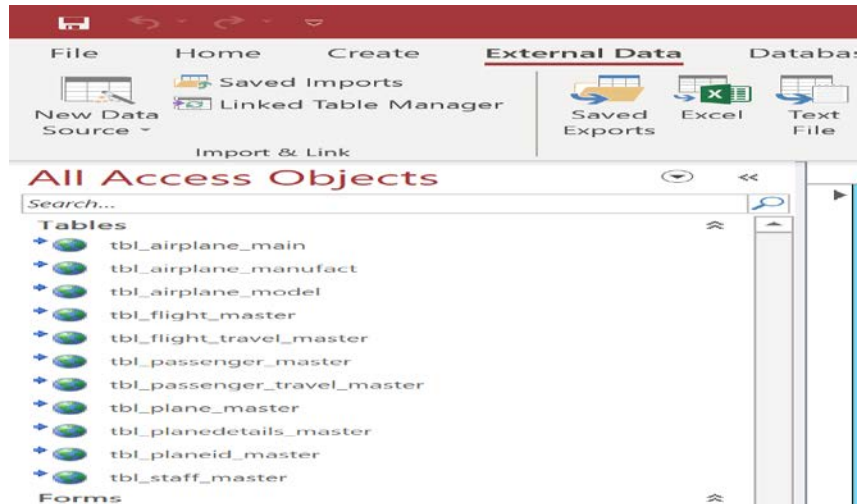


Fig 2: Tables imported in MS Access

The below table shows the MS Access view of imported staff_master table and its attributes.

EmpNo	Surname	Name	Address	Phone	Salary	ratings	designation	No_Of_Working
A1_01	Mark	Rose	35 Palm St,Birmingham	4567890	11000	3.8	Pilot	120
A1_05	Joy	Reena	47 Campus Road,Lo	74257788	12000		Crew	
A1_06	Stuart	Son	28 Rock avenue,Der	88765432	9000	4.1	Pilot	200
A1_07	Joseph	Reena	44 cranmer st,Lutor	778896543	8500		Crew	
A1_08	Thomson	Roy	2 abortsford Drive,I	756674324	7000		Crew	
A1_09	Crew	Monica	47 Campus Road, Lc	76543270	10000	3.8	Pilot	80
A1_12	Willis	John	36High street,Lento	775432467	15000	4.2	Pilot	100
A1_13	Tom	Riatg	76 Sunny Vale,Califc	78900754	10000	4.3	Pilot	160
A1_15	Steve	Veena	4 Avenue,Nottinghu	43456788	8000		Crew	
A1_20	Phillip	Sophie	76 Huntingdon,Subs	56789087	12700	2.8	Pilot	150

Fig 3: Staff table and its attributes

Once the connection has made, then relationships has been created between each table in MS Access. The relationships are created in a way that the tables are linked to other tables in accordance to the primary key and foreign key assigned when the tables were created.

4 Entity Relationship Diagram

As designed in the report 1, an ER diagram representing the relationship between each table in the database is encountered.

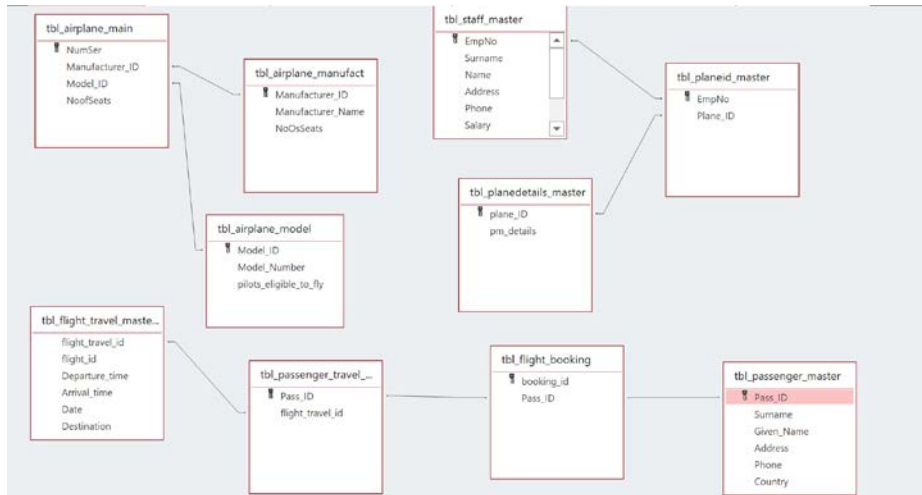
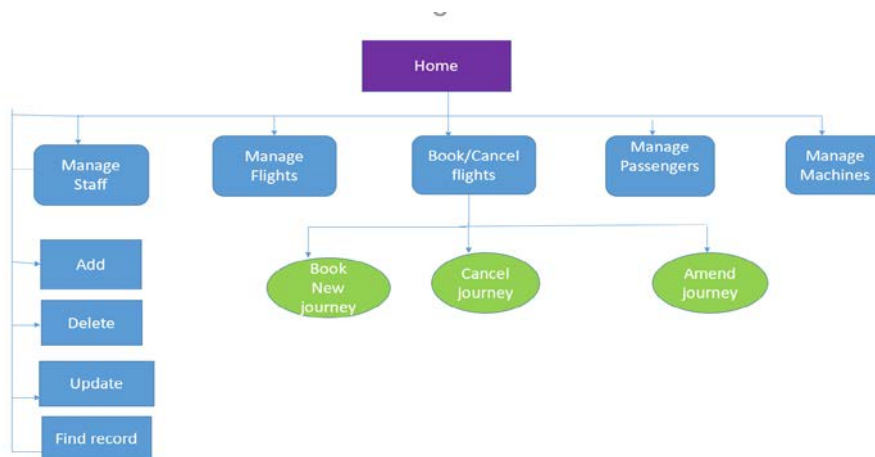


Fig 4: ER Diagram

5 System Design

5.1 Navigation map of the system



The above figure describes the navigation on how the system works and the User interface has been designed based on that.

5.2 User Interface design

In MS Access, a bound form connects directly to the data source from the table or query that can be typically used to perform all operations within the data. The system has been developed with comprehensive features to provide efficient access to the users and the business. The forms have been designed to be easy of use and provides customization with variety of features. In order to create form,

Select the required table from the navigation pane -> select Create tab -> choose Form

This application comprises of 7 forms that navigates to each other. The following figures shows the user interface design of Student Flight Management System.



Fig 5: Home Page of Students Flight Management System

When the Click here button under Covid 19 is clicked, displays the current update about travel restrictions are provided as a guideline to the business. The below figure shows the page information.

COVID-19 Travel Updates

To slow down the spread of coronavirus and protect the health and well-being of all Europeans, some travel restrictions have been necessary. The European Commission is doing its utmost to allow people to meet friends and family and to ensure free movement of citizens, goods and services – with full respect of health and safety measures.

Safely resuming travel

On 15 June 2020, the European Commission launched Re-open EU, an online platform that contains essential information about the safe relaunch of free movement and tourism across Europe. It provides information on

- 1.borders
- 2.available means of transport
- 3.travel restrictions
- 4.public health and safety measures, such as physical distancing or wearing of facemasks
- 5.other practical information for travellers

Re-open EU also brings together up-to-date information from the Commission and Member States. It allows people to browse country-specific information through an interactive map, offering updates on applicable national measures, as well as practical advice for visitors in that country. Available in the 24 official EU languages, the platform is easily accessible on desktop and mobile by following and bookmarking the Re-open EU link: <https://reopen.europa.eu/>

Go Back

Fig 6: Travel Updates

Staff Information

EmpNo

Name

Address

Salary

Surname

Phone

Ratings

Designation

Add Record

Delete Record

Find Record

Save

Home




Fig 7: Manage Staff

Flight Information

Flight Travel ID	<input type="text" value="FLT_001"/>
Flight ID	<input type="text" value="FL_001"/>
Departure Time	<input type="text" value="17:15"/>
Arrival Time	<input type="text" value="10:50"/>
Date	<input type="text" value="19/07/2018"/>
Destination	<input type="text" value="Scotland"/>




Fig 8: Manage Flights

Machine Information

NumSer	<input type="text"/>	<input type="button" value="Add New"/> <input type="button" value="Delete Machine"/> <input type="button" value="Find Record"/> <input type="button" value="Update"/>
Manufacturer_Name	<input type="text"/>	
Model_Number	<input type="text"/>	
Manufacturer_ID	<input type="text"/>	
Model_ID	<input type="text"/>	
NoofSeats	<input type="text"/>	

Fig 9: Manage Machines

Fig 10: Manage Passengers

Fig 11: Book/Cancel Flights

5.3 Description of the system

i. Manage Staff -> add new staff

- When the admin wants to add a new staff, in the staff information page the required data should be given and when the add record button is clicked then the new staff record will be added in the database.

The left screenshot shows the 'Staff Information' form with the following fields: EmpNo (AI_18), Name (Dawn), Address (78 Hill Road, Mexico), Salary (13000), Surname (Scott), Phone (88907654), Ratings (3.8), and Designation (Pilot). Buttons for 'Add Record', 'Delete Record', 'Find Record', and 'Save' are visible. The right screenshot shows the same form with a 'New record added successfully!' message box and an 'OK' button.

EmpNo	Surname	Name	Address	Phone	Salary	ratings	designation	No_Of_Working
1234	ffhj	yffj	45 hh	54467899	2233	2.7	pilot	78
AI	Mark		35 Palm St, Birmingha	94567890	11000	3.8	Pilot	120
AI_05	Joy	Reena	47 Campus Road, Londo	74257788	12000		Crew	
AI_06	Stuart	Son	28 Rock avenue, Derby	88765432	9000	4.1	Pilot	200
AI_07	Joseph	Reena	44 cranmer st, Luton	78896543	8500		Crew	
AI_08	Thomson	Roy	2 abortsford Drive, T	56674324	7000		Crew	
AI_09	Crew	Monica	47 Campus Road, Lond	76543270	10000	3.8	Pilot	80
AI_11	Stock	Tiger	23 Rope Walk, Toronto	67789087	8000	3.4	Pilot	
AI_12	Wills	John	36 High street, Lenton	75432467	15000	4.2	Pilot	100
AI_13	Tom	Riatg	76 Sunny Vale, Califo	78900754	10000	2.9	Pilot	160
AI_15	Steve	Veena	4 Avenue, Nottingham	43456788	8000		Crew	
AI_16	Tim	Sheela	66 North St, Gao	75432123	9000		Crew	
AI_17	Palli	Monish	11 Bridge St, Derby	65678907	6800		Crew	
AI_18	Scott	Dawn	78 Hill Road, Mexico	88907654	13000	3.8	Pilot	
AI_20	Philip	Sophie	76 Huntingdon, Subway	56789087	12700	3.4	Pilot	150
AI_21	Prabhu	Wag	9 King Street, London	99865324	5000	2.8	Pilot	75

Database view of newly added record

ii. Manage Staff -> Find Record

- When an admin encounters to find a specific record from the database, then find record button should be clicked next a pop-up window appears to enter the EmpNo of the staff. When the EmpNo is submitted, the requested staff record will be retrieved from the database.

The screenshot shows a web application titled "Staff Information". It contains a form with the following fields: EmpNo (AI 16), Name (Shee), Address (66 N), Salary (9000), Surname (Tim), Phone (7543), Ratings, and Designation (Crew). A blue "Add Record" button is at the top right. A "Find and Replace" dialog box is open in the center, showing "Find What: AI 16", "Look In: Current field", "Match: Whole Field", and "Search: All". The dialog also has "Find Next" and "Cancel" buttons. A red "Home" button is at the bottom right of the form.

iii. Manage Machines -> Delete machine

- When the user intends to delete a record in the machines table which is named in my database as airplane, the delete button should be clicked to delete the required record then the appropriate record will be deleted in the database.

The screenshot shows a web application titled "Machine Information". It contains a form with the following fields: NumSer (100301), Manufacturer_Name (Boeing), Model_Number (E171), Manufacturer_ID (M002), Model_ID (MQ_007), and NoofSeats (300). There are buttons for "Add New", "Delete Machine", "Find Record", and "Update". A "Home" button and a "more" link are at the bottom. A Microsoft Access dialog box is open, displaying a warning: "You are about to delete 1 record(s). If you click 'Yes', you won't be able to undo this Delete operation. Are you sure you want to delete these records?" with "Yes" and "No" buttons.

NumSer	Manufacturer_ID	Model_ID	NoofSeats
100302	M003	MO_011	350
100303	M003	MO_012	25
100402	M004	MO_015	300
100403	M004	MO_016	700
100501	M005	MO_018	25
100502	M005	MO_019	95
100503	M005	MO_020	220
100504	M005	MO_021	300
*			

The database view of deleted record

iv. Manage Passengers -> Print page

- When a passenger wishes to print the page the print option from the passenger form allows to do it.

The screenshot shows a web application interface for managing passengers. On the left, there is a 'Passenger' form with the following fields: Pass_ID (PS_006), Surname (Judes), Given_Name (Robinson), Address (467 Kings Road, Port), and Phone (735687664). At the bottom of the form are buttons for 'Search' and 'Go Back'. Overlaid on the right side of the form is a 'Print' dialog box. The dialog box has a 'Printer' section showing 'OneNote for Windows 10' as the selected printer, with status 'Ready' and type 'Microsoft Software Printer Driver'. Below this, there are options for 'Print Range' (All, Pages, Selected Record(s)) and 'Copies' (Number of Copies: 1, Collate). The 'Print to File' checkbox is unchecked. At the bottom of the dialog are 'Setup...', 'OK', and 'Cancel' buttons.

v. Book / Cancel Flight -> Book a new journey

- When the admin wants to book for a new flight, after entering the travelling details, the Book Now button should be pressed then the page enables the user to enter the name and age of the passengers and click on Book Flight, the new passenger information will be updated into tbl_passenger_master

Book a new flight

Cancel a booking

Origin

London

Destination

Singapore

Start Date

03/01/2021

Return Date

09/01/2021

Journey Type

Return

Number of Passengers

2

Book now

Passenger Details

Passenger 1 First Name

Last Name

Age

Passenger 2 First Name

Last Name

Age

Book Flight

Book a new flight

Cancel a booking

Origin

London

Destination

Singapore

Start Date

03/01/2021

Return Date

09/01/2021

Journey Type

Return

Number of Passengers

2

Book now

Passenger Details

Passenger 1 First Name

Divu

Last Name

Kumar

Age

23

Passenger 2 First Name

Samar

Last Name

Sachin

Age

28

Book Flight

Book flight button vb.net code

```

Private Sub Command291_Click()
    Dim conn As ADODB.Connection
    Dim rs As ADODB.Recordset
    Dim fld As ADODB.Field
    Dim lastvalue As String

    Set conn = New ADODB.Connection
    Dim sql As String
    conn.ConnectionString = "Driver={MySQL ODBC 8.0 Unicode Driver};Server=localhost;Database=flightmanagement_system;Uid=root;
    conn.Open

    Dim Origin, Destin, JournTy As String
    Dim NoOfPassengers As Integer
    Dim FromDate, ReturnDate As Date

    Set rs = New ADODB.Recordset
    conn.Execute ("Insert into tbl_passenger_master(Given_Name, Surname) values ('" & Text179.Value & "','" & Text192.Value &
    rs.Open "Select Pass_ID from tbl_passenger_master order by Pass_ID Desc limit 1", conn
    For Each fld in rs.Fields
        lastvalue = fld.Value
    Next
    rs.MoveNext
    rs.Close

    MsgBox "Your flight booking is confirmed", vbOKOnly, "Flight Booking Confirmation"
    Command434.SetFocus
End Sub

```

vi. Cancel Booking

- When the admin wants to cancel a booking that has been made, then cancel a booking on the add passengers page should be clicked. Then displayed with the booking ID to cancel field -> when the booking id is entered the specific booking will be cancelled in the database.

The screenshot shows a web application interface for flight management. At the top, there are two buttons: "Book a new flight" and "Cancel a booking". Below these, there are several input fields: "Origin", "Destination", "Start Date", "Return Date", and "Journey Type". A "CancelBooking" button is located below the "Journey Type" field. A modal dialog box titled "Booking Cancellation confirmation" is open in the center, displaying the message "Your flight booking 503 has been cancelled" and an "OK" button. Below the dialog, there is a text input field labeled "Enter your booking ID to cancel" with the value "503" entered, and a "Confirm Cancel" button.

booking_id	Pass_ID		
502	108		
500	110		
504	110		
505	112		
506	113		
*			

The database view of cancelled booking id of the passenger.

Confirm Cancel button.vb code



5.4 Trigger Procedure

Through the flight booking form, the flight information that customer wishes to fly is selected along with travel dates, travel type and number of passengers. On entering the passenger(s) details in the page, the confirmation button inserts the relevant data into tbl_passenger_master table. In this instance, an auto incremented Pass_ID is created for the new record. On such trigger, the newly created Pass_ID from tbl_passenger_master is inserted into another table tbl_flight_booking through foreign reference key Pass_ID. This occurrence is achieved through triggers directly defined in MySQL

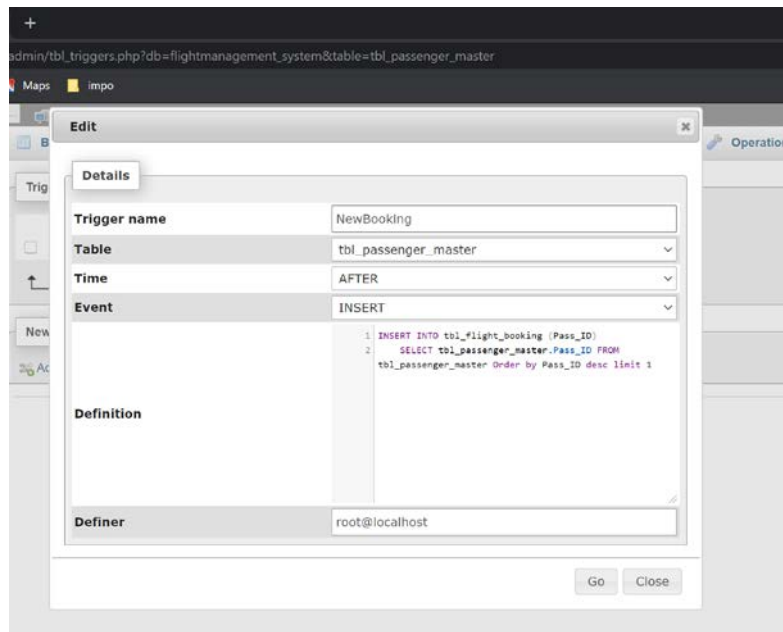
```

DROP TRIGGER IF EXISTS `NewBooking`;
CREATE DEFINER='root'@'localhost' TRIGGER `NewBooking` AFTER INSERT ON
`tbl_passenger_master`
FOR EACH ROW

```



```
INSERT INTO tbl_flight_booking (Pass_ID) SELECT tbl_passenger_master.
Pass_ID FROM tbl_passenger_master Order by Pass_ID desc limit 1;
```



6 Report Generation

Creating an effective database system is crucial for any business, like wise generating reports for the designed database allows the business to view the summary of information, easy for printing, produces clear and attractive view on the data, highlights all the major facts and trends. MS Access produces sophisticated ready to use reports in which the data can be retrieved from one or more tables and queries. The motive of creating reports is to let the business admins to take decisions, helps to achieve goals of the business by analyzing the current status, useful to produce business information during conferences, helps to analyze in making changes over the business process management. The reports in MS Access can be created in either selecting the table choosing report wizard and customizing the required field from various tables or using queries.

The reports for this application have been created using the report wizard. Reports are generated for the following conditions.

- a) No of passengers by flight

No of Passengers

NoOfPassengers	Flight_Number
233	FL_008
321	AI 161
189	JA 0567
325	JA 0567
210	AI 161
300	TI A256
147	TI A256
387	AI 323
177	ER 2019
300	ER 1098
222	AF 1007
400	ER 1098
288	AI 161
411	ER 2019
290	AI 161

b) List of company destinations and their schedule

flights schedule

Departure_time	Arrival_time	Date	Destination	Flight_Number	Origin
17:15	10:50	19/07/2018	Scotland	FL_008	California
12:15	03:05	30/05/2017	Singapore	AI 161	India
19:05	16:00	06/10/2019	Malaysia	JA 0567	UAE
20:05	23:05	12/08/2018	India	JA 0567	Malaysia
04:05	21:03	11/10/2019	Boston	AI 161	China
12:03	07:03	15/03/2015	Thailand	TI A256	Singapore
10:05	13:55	20/04/2017	Mauritius	TI A256	Singapore
01:03	17:05	20/06/2016	Singapore	AI 323	India
02:01	08:15	17/10/2014	London	ER 2019	California
08:05	12:45	12/01/2009	Boston	ER 1098	London
14:15	20:02	22/07/2011	Canada	AF 1007	Boston
12:03	16:01	10/03/2018	London	ER 1098	Scotland
01:15	07:02	02/04/2020	Chennai	AI 161	Boston
12:03	18:05	05/06/2017	Germany	ER 2019	London

c) Number of working hours of all pilots

Working Hours for all pilots			
Name	Surname	No_Of_Working_hours	designation
Dawn	Scott		Pilot
John	Wills	100	Pilot
Jones	Mark	120	Pilot
Monica	Crew	80	Pilot
Manish	Palli		Crew
Reena	Joseph		Crew
Reena	Joy		Crew
Riatg	Tom	160	Pilot
Roy	Thomson		Crew
Sheela	Tim		Crew
Son	Stuart	200	Pilot
Sophie	Philip	150	Pilot
Tiger	Stock		Pilot
Veena	Steve		Crew
Wag	Prabhu	75	Pilot

7 Conclusion

To conclude, a well-designed efficient system has built for the business and the data are organized using normalization, describing the relations between tables through ER diagrams and the queried using MySQL- connected the front-end and backend using the Open Database Connectivity in order to build a complete working system of the business. Furthermore, reports have been generated for analysis and improvement of potential and future development.

8 References

1. Ping, Y., Hong-Wei, H. and Nan, Z. (2015a). BIG-IP logout page. [online] ieeexplore-ieee-org.ezproxy.derby.ac.uk. Available at: <https://ieeexplore-ieee-org.ezproxy.derby.ac.uk/document/7053638> [Accessed 18 Dec. 2020].