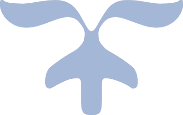


Air Force School

Viman-Nager

Pune - 411014

Computer-Science Project report





**SUBMITTED BY: SUMITTED TO:**

**Name**: Aditya Sharma **Teachers Name**: Monali Gavali

**Class:** XII’ C Signature:\_\_\_\_\_\_\_\_\_\_\_\_

# CERTIFICATE

This is to certify that **Aditya Sharma** of class: **XII C** from **Airforce School Viman Nager Pune** has done his project on the topic:

**FLIGHT MANAGEMENT SYSTEM**

He has taken interest and has shown at most sincerity in completion of this project.

I certify this project up to my expectation & as per guidelines issued by **CBSE, NEW DELHI.**

Subject teacher Signature:

\_\_\_\_\_\_\_\_\_\_\_

Principal Signature:

\_\_\_\_\_\_\_\_\_\_\_

Examiner Signature:

\_\_\_\_\_\_\_\_\_\_\_

# **ACKNOWLEDGMENT**

It is with pleasure that I acknowledge my sincere gratitude to our teacher,

***Mrs. Monali Gavali*** who taught and undertook the responsibility of teaching the subject computer science. I have been greatly benefited from his classes.

I am especially indebted to our Principal, ***Mrs. Ashwini Modhave*** who has always been a source of encouragement and support and without whose inspiration this project would not have been a successful I would like to place on record heartfelt thanks to her.

Finally, I would like to express my sincere appreciation for my parents and my friends the fine time that we all shared together make this project with perfection by learning from mistakes and the suggestions given by my parents.

## FLIGHT Booking SYSTEM(FBS)

The Airline Reservations System (ARS) was one of the earliest changes to improve efficiency. ARS eventually evolved into the Computer Reservations System (CRS), and then into Global Distribution System (GDS). The airline industry created the first GDS in the 1960s as a way to keep track of flight schedules, availability, and prices.

GDSs were actually among the first e-commerce companies in the world facilitating B-2-B electronic commerce as early as the mid-1970s, when SABRE (owned by American Airline) and Apollo (United) began installing their propriety internal reservations systems in travel agencies. Prior to this, travel agents spent an inordinate amount of time manually entering reservations. The airlines realized that by automating the reservation process fortravel agents, they could make the travel agents more productive and essentially turn into an extension of the airline’s sales force. It is these original, legacy GDSs that today provide the backbone to the Internet travel distribution system.

### NEED OF FBS:

In 21st century the world has become a global village where everything is available in a single click of mouse button. Aviation sector is one of fastest mode of travel available with us, both at domestic and international level. To maintain such a large system is a hectic job. The present system is very time consuming and in-efficient.

The definition of our problem lies in manual system and **a** fully automated system.

**Manual system**:

The system is more prone to errors and sometimes item-counters various problems which are unstructured.

**Technical system**:

With the advent of latest technology if we do not update our system then our business will suffer massive losses financially. The technical system (we have proposed) contains the tools of latest trend i.e. computers printers, fax etc. The systems with this technology are very fast, accurate, user-friendly and reliable.

### Objective

1. Minimized documentation and no duplication of records.
2. Reduced paper work.
3. Accuracy
4. Reliability
5. Faster information flow between various departments
6. Smart Revenue Management
7. Effective billing of various services
8. Exact stock information

### Tool and Platforms

#### Software Requirements

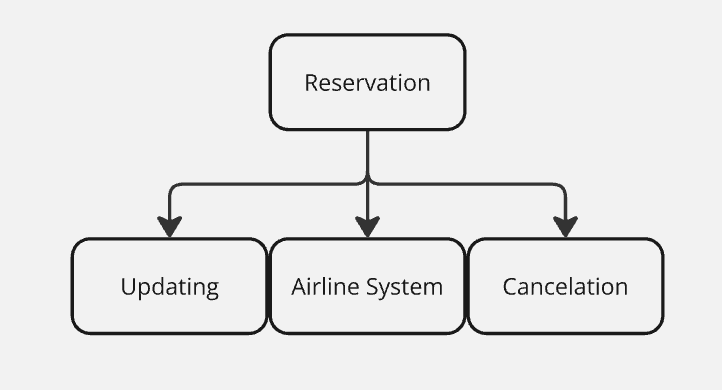
* **Programming language: -**  Python
* **Relational Database: -** MySQL

### Analysis

#### Scope and Assumption

Scope of this system is limited to Flight booking; we assume Flight schedule are available from other service

#### High-level Data flow for Airline Reservation system



#### High-level function performed by system

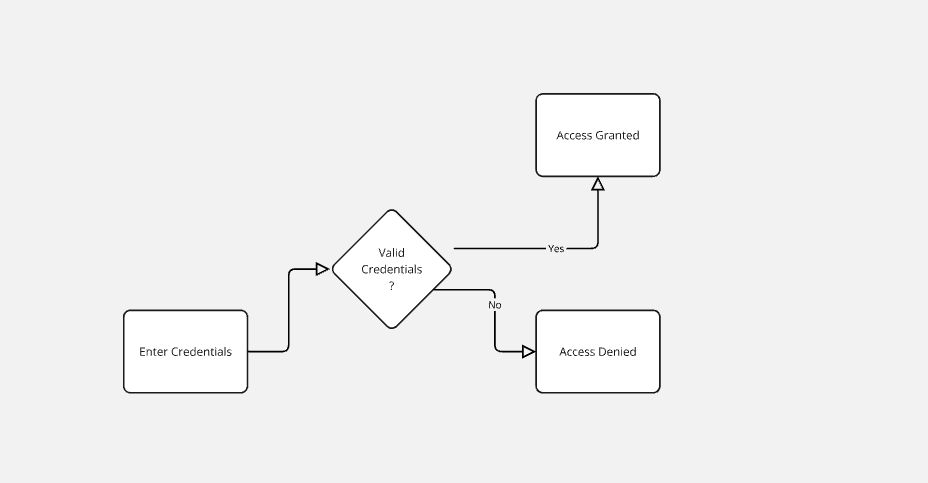
1. User should be able to login to system using credentials
2. User should be able to search Flight details based on input
3. city and Date
4. User should be able to inquire the Flight details and fair
5. User should be able to book a Flight

### Low level Design

Based on above high-level analysis, we conclude following entity need to be created

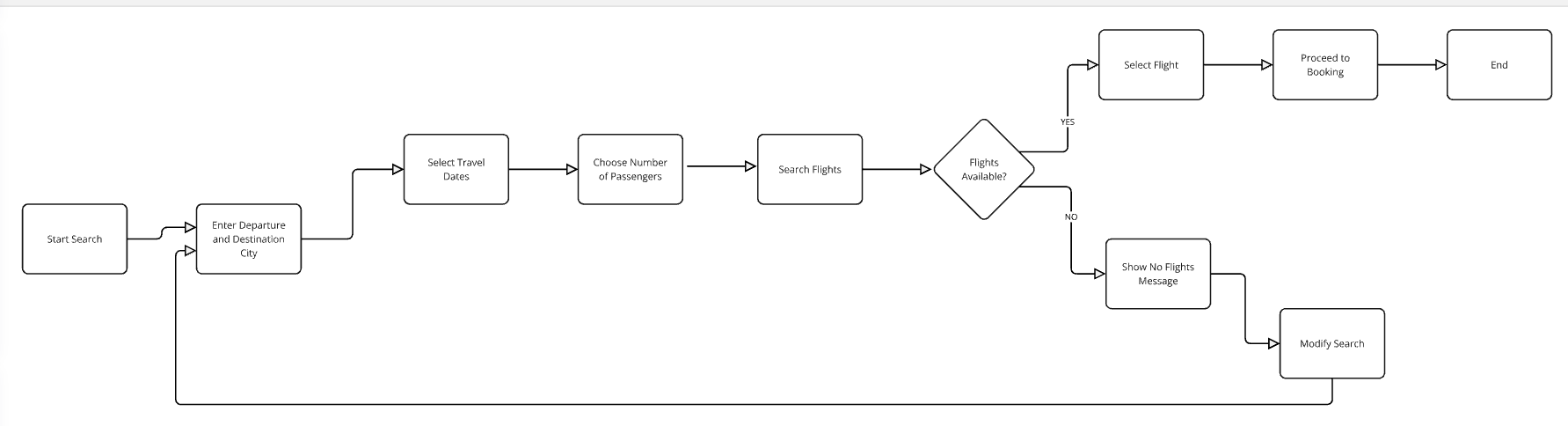
#### Login Details

In order to book a flite user needs to login to system, if login is valid access will be granted else access will be denied



#### Search a flight

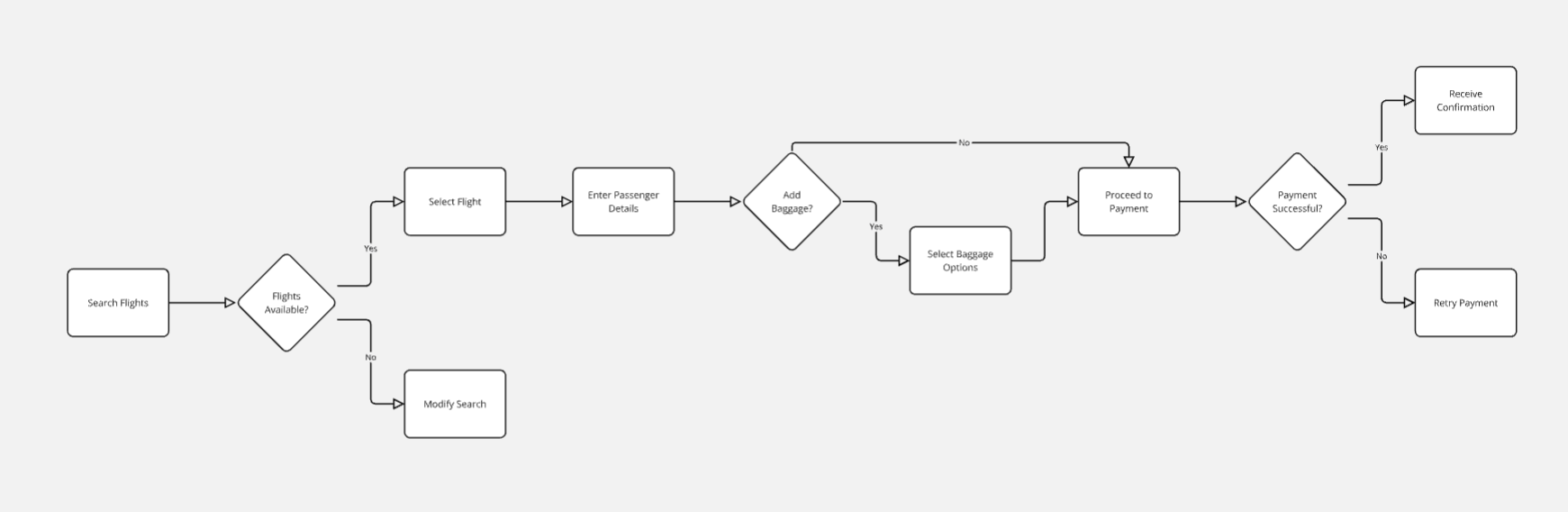
To search a flight, user need to follow below step

1. Flight’s Departure and destination city
2. Enter travail dates
3. Search flight
4. If flight available then proceed with booking
5. Else show no flights and ask user to modify the search request

#### Book a flight

To book a flight

1. Check if flight is available (from search flight)
2. Select flight
3. Enter passanger details
4. Do you have baggage?
5. Yes, Select baggage option
6. Proceed with payment
7. If payment successful then send confirmation
8. Else ask user to retry

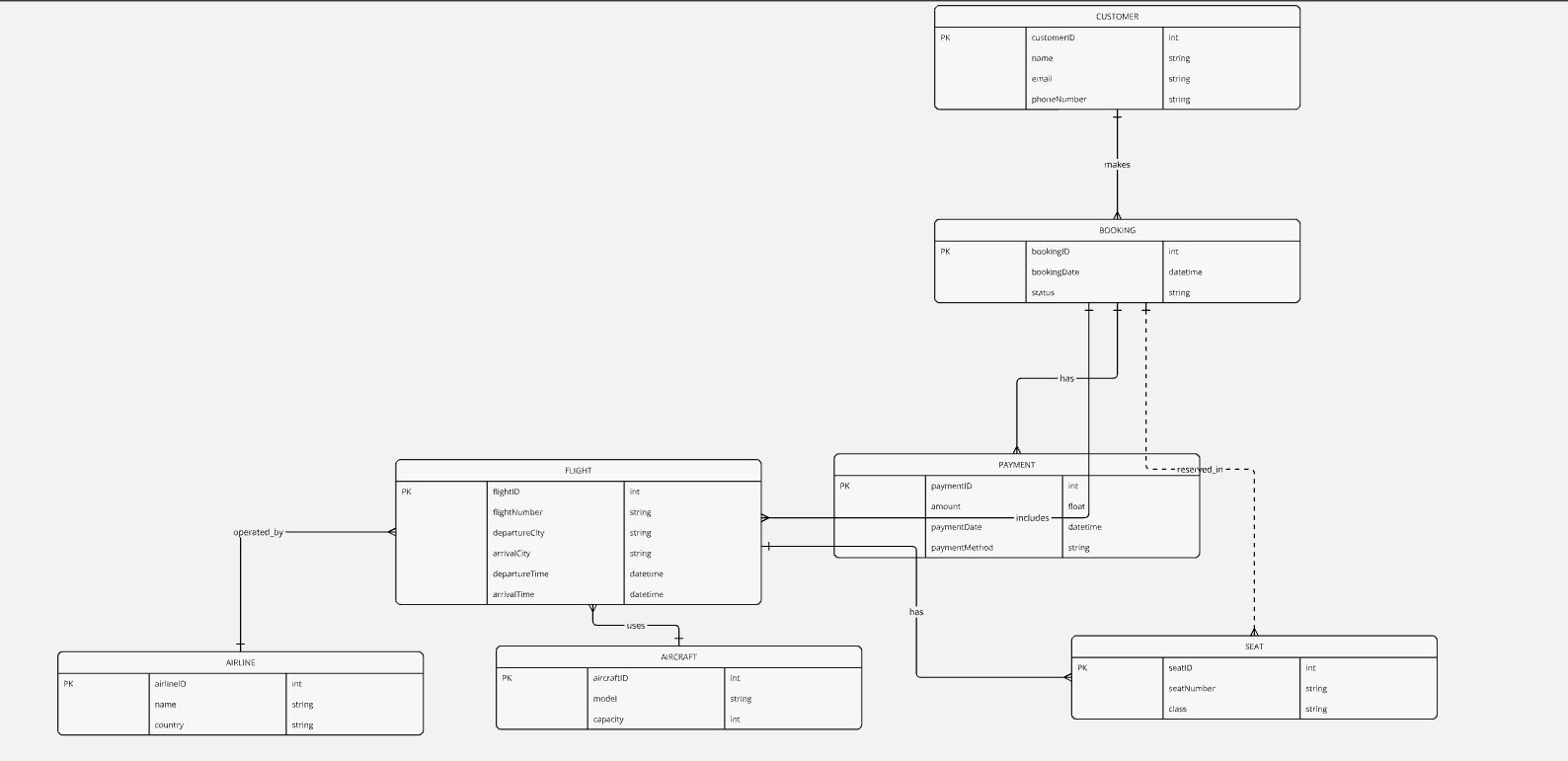


#### Data model entity Diagram

Data model entity diagram represent, relationship between different entity,

Below are entities used to book a flights

1. Customer holds passenger information
2. Booking holds booking details a confirmation to customer
3. Payments hold payment information made by customer to book a flight
4. Flight holds flight details booked by customer
5. Aircraft hold aircraft details, above flight belong to
6. Airline hold Air details that own aircraft

`

1)Customer Table

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Type | Integrity | Note |
| CustomerID | Integer | Primary\_Key | ---- |
| Name | Varchar | Not\_null | ---- |
| Email | Varchar | Not\_null | ---- |
| Phone\_Number | Integer | Not\_null | ---- |

2)Booking Table

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Type | Integrity | Note |
| Booking\_id | Integer | Primary\_Key |  |
| Booking\_Date | Datetime | Not\_null |  |
| Status | Varchar | Not\_null |  |
| Coustmer\_id | Integer | Foreign\_key | Booking records Customers detail |
| Flight\_ID | Integer | Foreign\_key | Which flight will have which customer |
| Airline\_id | Integer | Foreign\_key |  |
| Payment\_id | Integer | Foreign\_key |  |

1. Seat

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Type | Integrity | Note |
| Seat\_ID | Int | Primary\_Key |  |
| Seat\_Number | Int | Not\_null |  |
| Class | Varchar | Not\_null |  |
| Flight\_id | Int | Foreign\_key | Seat in a flight |

1. Flight

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Type | Integrity | Note |
| Flight\_ID | Int | Primary\_Key |  |
| Flight\_number | Int | Not\_null |  |
| Departure\_City | Varchar | Not\_null |  |
| Arrival\_City | Varchar | Not\_null |  |
| Departure\_Time | DateTime | Not\_null |  |
| Arrvial\_Time | DateTime | Not\_null |  |

1. Aircraft

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Type | Integrity | Note |
| Air\_Craft\_ID | Integer | Primary\_Key |  |
| Model | Varchar | Not\_null |  |
| Capacity | Integer | Not\_null |  |
| Flight\_id | Integer | Foreign\_key | Aircraft owned by an airline company |

1. Airlines

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Type | Integrity | Note |
| Airline\_id | Integer | Primary\_Key |  |
| Name | Varchar | Not\_null |  |
| Country | Varchar | Not\_null |  |
| Flight\_ID | Integer | Foreign\_key |  |

1. Customer

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Type | Integrity | Note |
| customer\_id | Integer | Primary\_Key |  |
| customer\_name | Varchar | Not\_null |  |
| Email | Varchar | Not\_null |  |
| Phone | Integer |  |  |

8)Payment

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Type | Integrity | Note |
| payment\_id | Integer | Primary\_Key |  |
| amount | Float | Not\_null |  |
| payment\_date | Datetime | Not\_null |  |
| payment\_method | Varchar |  |  |

8) Booking

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Type | Integrity | Note |
| booking\_id | Integer | Primary\_Key |  |
| booking\_date | Float | Not\_null |  |
| payment\_date | Datetime | Not\_null |  |
| booking\_status | Varchar |  |  |
| customer\_id | Int | Foreign\_key |  |
| payment\_id | Int | Foreign\_key |  |
| Flight\_id | char(15) | Foreign\_key |  |

#### The Programme code input and output screens.

C:\Users\Aditya\AppData\Local\Programs\Python\Python311\python.exe E:\ComputerScience\project\AirlineBookingSystem\Project\_Filght\_booking\_system.py

Login Details

===============================================================================

User Name: aditya

Enter password: test

True

Login successful

Search Flight details

=========================

Arrival : PNQ

Departure : ADI

Enter travail Date :2024-12-26

AVAILABLE FLIGHTS

Airline ,Flight No ,Arival , Depature Arival Time , Departure time

===============================================================================

AkashAir, QP-1508, PNQ, ADI, 15:00:00, 17:00:00

AkashAir, QP-1509, PNQ, ADI, 8:00:00, 10:00:00

SpiceJat, SG-2961, PNQ, ADI, 11:00:00, 15:00:00

SpiceJat, SG-2962, PNQ, ADI, 6:00:00, 8:00:00

===============================================================================

Choose flight to book:::SG-2962

Your flight details are as below

SpiceJat, SG-2962, PNQ, ADI, 6:00:00, 8:00:00

Enter name::Aditya Sharma

Enter Email::aditya@gmail.com

Enter Phone Number::988337878

A-001, ECONOMY

A-001, ECONOMY

A-001, ECONOMY

A-001, ECONOMY

B-003, Business

B-004, Business

===============================================================================

Please choose seat:: B-0045

Total Payment: 5000

Please confirm with payment

===============================================================================

Your Ticket, Booking Id: : 1222333

Name: : Aditya Sharma Email: aditya@gmail.com Phone 988337878

Your flite details: : SpiceJat, SG-2962, PNQ, ADI, 6:00:00, 8:00:00 Seat Number: B-0045

===============================================================================

Process finished with exit code 0

#### SQL table and Master data.

CREATE DATABASE flight\_booking\_system;  
use flight\_booking\_system;  
  
CREATE TABLE account (  
 user\_name varchar(10),  
 password varchar(20) NOT NULL,  
 PRIMARY KEY (user\_name)  
);  
  
INSERT INTO account value ('aditya', 'test');  
  
  
create table seat(  
 seat\_id char(5) primary key,  
 seat\_number char(10),  
 class varchar(10)  
  
 );  
  
  
create table airline(  
 airline\_id char(5) primary key,  
 name char(10),  
 country varchar(10)  
 );  
  
insert INTO AIRLINE VALUE ('00001', 'AkashAir', 'IN');  
insert INTO AIRLINE VALUE ('00003', 'SpiceJat', 'IN');  
  
drop table flight;  
create table flight(  
 Flight\_id char(15) primary key,  
 Flight\_number char(15),  
 Departure varchar(10),  
 Arival varchar(10),  
 Departure\_time datetime,  
 Arival\_time datetime,  
 airline\_id char(5),  
 FOREIGN KEY (airline\_id) REFERENCES airline(airline\_id)  
 );  
  
insert into flight  
value ('SG-2962', 'SG-2962', 'PNQ', 'ADI', '2024-12-26 06:00:00','2024-12-26 08:00:00', '00003');  
  
insert into flight  
value ('SG-2961', 'SG-2961', 'PNQ', 'ADI', '2024-12-26 11:00:00','2024-12-26 15:00:00', '00003');  
  
insert into flight  
value ('QP-1509', 'QP-1509', 'PNQ', 'ADI', '2024-12-26 08:00:00','2024-12-26 10:00:00', '00001');  
  
insert into flight  
value ('QP-1508', 'QP-1508', 'PNQ', 'ADI', '2024-12-26 15:00:00','2024-12-26 17:00:00', '00001');  
  
select \* from flight;  
  
drop table seat;  
create table seat(  
seat\_id char(15) primary key,  
seat\_number char(10),  
class varchar(10),  
Flight\_id char(15),  
FOREIGN KEY (Flight\_id) REFERENCES flight(Flight\_id)  
);  
  
insert into seat value('SG-A-001','A-001', 'ECONOMY', 'SG-2962');  
insert into seat value('SG-A-002','A-001', 'ECONOMY', 'SG-2962');  
insert into seat value('SG-B-003','B-003', 'Business', 'SG-2962');  
insert into seat value('SG-B-004','B-004', 'Business', 'SG-2962');  
insert into seat value('SG-A-005','A-001', 'ECONOMY', 'SG-2962');  
insert into seat value('SG-A-006','A-001', 'ECONOMY', 'SG-2962');  
  
  
  
insert into seat value('SG-62-A-001','A-001', 'ECONOMY', 'SG-2961');  
insert into seat value('SG-62-A-002','A-001', 'ECONOMY', 'SG-2961');  
insert into seat value('SG-62-B-003','B-003', 'Business', 'SG-2961');  
insert into seat value('SG-62-B-004','B-004', 'Business', 'SG-2961');  
insert into seat value('SG-62-A-005','A-001', 'ECONOMY', 'SG-2961');  
insert into seat value('SG-62-A-006','A-001', 'ECONOMY', 'SG-2961');  
  
insert into seat value('QP-1509-A-001','A-001', 'ECONOMY', 'QP-1509');  
insert into seat value('QP-1509-A-002','A-001', 'ECONOMY', 'QP-1509');  
insert into seat value('QP-1509-B-003','B-003', 'Business', 'QP-1509');  
insert into seat value('QP-1509-B-004','B-004', 'Business', 'QP-1509');  
insert into seat value('QP-1509-A-005','A-001', 'ECONOMY', 'QP-1509');  
insert into seat value('QP-1509-A-006','A-001', 'ECONOMY', 'QP-1509');  
  
insert into seat value('QP-1508-A-001','A-001', 'ECONOMY', 'QP-1508');  
insert into seat value('QP-1508-A-002','A-001', 'ECONOMY', 'QP-1508');  
insert into seat value('QP-1508-B-003','B-003', 'Business', 'QP-1508');  
insert into seat value('QP-1508-B-004','B-004', 'Business', 'QP-1508');  
insert into seat value('QP-1508-A-005','A-001', 'ECONOMY', 'QP-1508');  
insert into seat value('QP-1508-A-006','A-001', 'ECONOMY', 'QP-1508');

#### Technical Detail of code.

##### Manage connection

import mysql.connector as sql\_connector  
  
def connect():  
 connection = sql\_connector.connect(  
 host="localhost",  
 user="root",  
 passwd="Aditya@2008",  
 database="flight\_booking\_system"  
 )  
 return connection

##### Login service

from connection\_manager import connect  
  
def login(user\_name, password):  
  
 query = """SELECT \* FROM ACCOUNT WHERE USER\_NAME = %s AND PASSWORD = %s"""  
 connection = connect()  
 cursor = connection.cursor()  
  
 criteria = ( user\_name, password)  
 cursor.execute(query, criteria)  
 data=cursor.fetchall()  
 if len(data)==0:  
 return False  
 else:  
 return True

##### Flight Search service

def search\_flight(departure, arival, date):  
 query = """SELECT A.NAME, F.Flight\_number, F.Departure, F.Arival, TIME(F.Departure\_time), TIME(F.Arival\_time)  
 FROM FLIGHT F, AIRLINE A  
 WHERE F.airline\_id=A.airline\_id   
 AND F.Departure= %s AND F.Arival = %s AND DATE(F.Departure\_time)= %s"""  
 connection = connect()  
 cursor = connection.cursor()  
 criteria = ( departure, arival, date)  
 cursor.execute(query, criteria)  
 data=cursor.fetchall()  
 return data

##### Flight Detail service

def get\_flight\_detail(flight\_number):  
 query = """SELECT A.NAME, F.Flight\_number, F.Departure, F.Arival, TIME(F.Departure\_time), TIME(F.Arival\_time)  
 FROM FLIGHT F, AIRLINE A  
 WHERE F.airline\_id=A.airline\_id   
 AND F.Flight\_number= %s"""  
 connection = connect()  
 cursor = connection.cursor()  
 criteria = (flight\_number,)  
 cursor.execute(query, criteria)  
 flight\_details=cursor.fetchone()  
  
 seat\_query = """SELECT seat\_number, class from seat WHERE Flight\_id= %s"""  
 connection = connect()  
 cursor = connection.cursor()  
 criteria = (flight\_number,)  
 cursor.execute(seat\_query, criteria)  
 available\_seats=cursor.fetchall()  
 return (flight\_details, available\_seats)

**Note:** **Following Service are work in progress**

##### Customer service

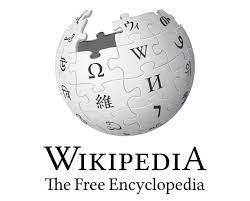
##### Payment Service

##### Booking Service

**BIBLOGRAPHY**



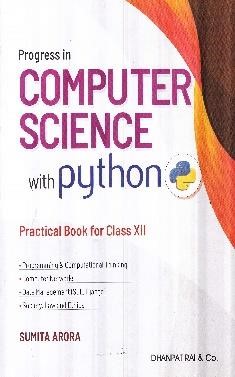
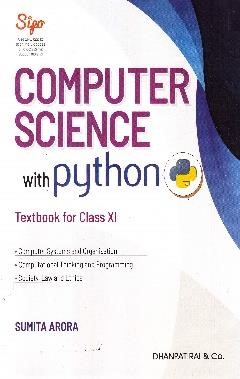
* [www.google.com](http://www.google.com/)



* [www.wikipedia.org](http://www.wikipedia.org/)



* [www.yahoo.com](http://www.yahoo.com/)



* Class XI & XII NCERT Book