

Problem description

The scope of the database is to store data for the operation of an airport. We decided to focus on the flight information (including status), airport, airlines, passengers, seats, boarding gates, aircrafts, and employees.

Business Rules

- 1) An **AIRLINE** has one or more **FLIGHTS**.
- 2) A **FLIGHT** is operated by just one **AIRLINE** (no codesharing).
- 3) A **FLIGHT** has one **AIRCRAFT**.
- 4) An **AIRCRAFT** is assigned for one or more **FLIGHTS**.
- 5) A **FLIGHT** departs from just one **AIRPORT**.
- 6) An **AIRPORT** has one or more **FLIGHTS** departures.
- 7) A **FLIGHT** arrives at just one **AIRPORT**.
- 8) An **AIRPORT** has one or more **FLIGHTS** arrivals.
- 9) A **FLIGHT** has just one **BOARDING GATE**.
- 10) A **BOARDING GATE** can be assigned for one or more **FLIGHTS**.
- 11) A **FLIGHT** has one or more **PASSENGERS**.
- 12) A **PASSENGER** can take one or more **FLIGHTS**.
- 13) A **PASSENGER** has one **SEAT**.
- 14) A **SEAT** is assigned for one **PASSENGER**.
- 15) An **AIRLINE EMPLOYEE** works for just one **AIRLINE**.
- 16) An **AIRLINE** can have one or more **EMPLOYEE**.
- 17) An **AIRPORT EMPLOYEE** works for just one **AIRPORT**.
- 18) An **AIRPORT** can have one or more **EMPLOYEE**.
- 19) The **ARRIVAL TIME** of a flight equals the sum of scheduled departure time, delay in minutes, and duration of flight, considering the **TIME ZONE** of the **ARRIVAL AIRPORT**.
- 20) The **FLIGHT DISTANCE** is calculated using the **LONGITUDE** and **LATITUDE** distance between **AIRPORTS**.
- 21) The **BOARDING TIME** is calculated automatically. If the **FLIGHT** is **INTERNATIONAL**, the boarding time starts 90 minutes before departure time plus delay minutes. If it is **DOMESTIC**, boarding time starts 45 minutes before departure time plus delay minutes.
- 22) The **FLIGHT INFORMATION** displayed on the screen or notification is: *the flight (**FLIGHT CODE**) departing from (**DEPARTURE AIRPORT**) airport and flying to (**ARRIVAL AIRPORT**) airport will arrive at the destination at (**ARRIVAL TIME**).*
- 23) If the **FLIGHT** is delayed, the system will calculate the **DELAYED DEPARTURE TIME** automatically, which equals to scheduled departure time plus delayed minutes.

I) Passenger

1. Passengers can view the future flights under their name in the self-check-in process, but no past flights.
2. The self-check-in process: select flight under the user's name, select seat, then finish self-check-in.
 - 2.1. Passengers can select a seat 24 hours before the flight departure.
3. Passengers can check formatted information about airports (including IATA airport code), local time of destination and aircraft model that they will fly in each flight.
4. Passengers can check a panel with formatted information about all departures and arrivals in the day.

II) Airport Employee

1. Airport Employees can identify the arrival time of a flight to prepare an unloading operation.
2. Airport Employees can track delayed flights in the airport, checking FlightStatus for each Flight.
3. Airport Employees can monitor the passengers list for security reasons.
4. Airport Employees can assign a Boarding Gate for a flight.
5. Airport Employees can only view the aircrafts at the airport.

III) Airline Employee

1. Airline Employees can check passengers' full names and ages on a flight.
2. Airline Employees can check passengers who need assistance to board a flight.
3. Airline Employees can verify the number of available seats on a flight to sell tickets.
4. Airline Employees can view the data of boarding time, the arrival time, seats and the distance of the flight.
5. Only Airline Employees can add a new aircraft.
6. Only Airline Employees can add a new flight and assign an aircraft to it.
7. Airline Employees can change flight status.

User group views

I) Passenger

1. Passengers can input the passport ID to view the future flights under their name. They need to retrieve the flight data from the PassengerTour class. The data includes: flightNumber, departureTime.
2. Passengers can self-check-in. They need to change the data of SeatCode from Seat class, CheckInStatus from PassengerTour class, and also need to use the method of AbleToSelectSeat() from PassengerTour class.
3. Passengers can cancel the flight. They need to change the data of CheckInStatus from the PassengerTour class.
4. Passengers can check formatted information about airports (including IATA airport code), local time of destination and aircraft model that they will fly in each flight by using the methods DisplayAirportCodePlusCity(), DisplayLocalTime() from Airport Class and DisplayAirCraftPlusAirline() from Aircraft Class.
5. Passengers can check a panel with formatted information about all departures and arrivals in the day by using the method DisplayFlightInformation() from Flight class.

II) Airport Employee

1. Airport Employees can track delayed flights in the airport, checking FlightStatus for each Flight.
2. Airport Employees can assign a Boarding Gate for a flight using BoardingGate class.
3. Airport Employees can monitor the passengers list for security reasons using Flight and PassengerTour classes.
4. Airport Employees can identify the aircrafts at the airport using Aircraft and Flight classes.

III) Airline Employee

1. Airline Employees can check passengers' full names and ages on a flight, they need to retrieve the data of fullname and age from the Passengers class.
2. Airline Employees can check passengers who need assistance to board a flight by selecting the attribute field IsHelpRequired from Passenger Class.
3. Airline Employees can check the number of available seats by using the method CalculateNAvailableSeats from the Flight Class.
4. Airline Employees can view the data of boarding time, the arrivalTime, numberOfAvailableSeat and the distance from Flight class.

5. Airline Employees can add a new aircraft that inherits properties from the parent class Aircraft Type.
6. Airline Employees can add a new flight and assign an aircraft to it, they need to change the data of Flight class and Aircraft class.
7. Airline Employees can change a flight's status by updating FlightStatus class and can use the CalculateDelayedDepTime method from Flight class to calculate the new departure time.

Major user views for the Airport database system

Data	Access Type	Airport Employee	Airline Employee	Passenger
Flight	Maintain		X	
	Query	X	X	X
Aircraft	Maintain		X	
	Query	X	X	X
Airport	Maintain	X		
	Query	X	X	X
Boarding Gate	Maintain	X		
	Query	X	X	X
Seat	Maintain		X	X
	Query	X	X	X

Maintain: enter, update, and delete.

Query: select

Based on Chapter 11, Figure 11.11.

Object Methods

Object Class	Method Name	Description
Flight	CalculateArrivalTime()	ScheduledDepartureTime + DelayInMinutes + FlightDuration + considering arrival airport timezone.
	CalculateDistance()	Longitude and Latitude distance between Airports using Haversine formula
	CalculateBoardingTime()	If isInternational = 1, BoardingTime starts 90 minutes before DepartureTime+DelayInMinutes. If isInternational = 0, BoardingTime starts 45 minutes before DepartureTime+DelayInMinutes.
	CalculateNAvailableSeats()	Aircraft Number of Seats - Number of Passengers
	DisplayFlightInformation()	Return String with Flight Code + Departure Airport + Arrival Airport + Delayed Departure Time + Arrival Time
	CalculateDelayedDepTime()	ScheduledDepartureTime+DelayInMinutes
Person	Age()	Today's date - DateOfBirth
	FullName()	Return String "FirstName FullName"
Airport	DisplayAirportCodePlusCity()	Return String "AirportCode City"
	DisplayLocalTime()	Uses TimeZone to return the string
Aircraft	DisplayAirCraftPlusAirline()	Return String "AirCraftType AirlineName"
PassengerTour	AbleToSelectSeat()	Calculate if 24 hours before DepartureTime, Return Boolean



Airport Database

ITEC4220 A - Milestone #1

Group #4

Mariia Barchuk

Maurílio Tiago Brüning Schmitt

Mukta Dey

Xie He

1

1. Introduction to a problem

The Airport Operational Database Management System is vital for the flawless running of the airports. It is all in one system that makes it easier to keep track of airport processes. It enforces several business rules and automates calculations and notifications to improve the overall passenger experience and operational efficiency.

- Enhances operational efficiency by automating key processes like delay calculations.
 - Provides passengers with real-time and accurate flight information.
 - Ensures compliance with business rules related to flights, aircraft, and airports.
 - Facilitates better resource allocation and management for both airports and airlines.
 - Improves the overall passenger experience by reducing confusion and uncertainty during travel.
-

2

System definition: Problem Scope

The system is consisting of 4 main parts, which can manage data related to:

Employees from airport
and airlines.

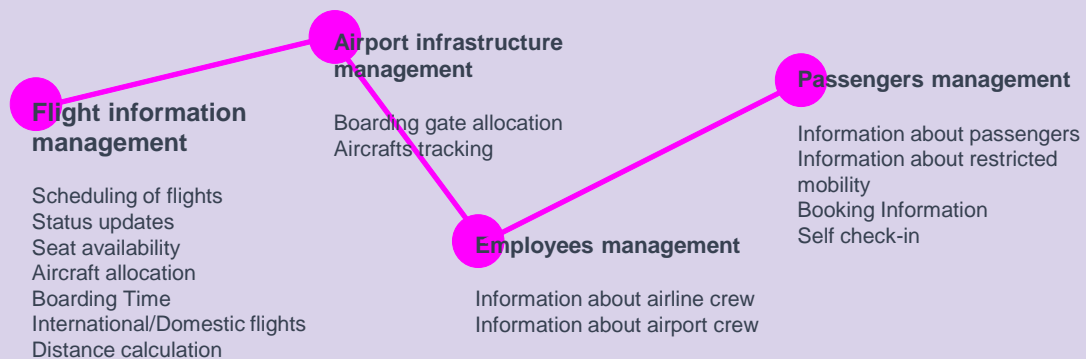
Flight information,
airlines, aircrafts

Airport infrastructure,
boarding gates

Passengers information
and the tours.

3

System definition: Problem Scope



4

Business Rules:

Passenger

1. Passengers can view the future flights under their name in the self-check-in process.
2. The self-check-in process: select flight under the user's name, select seat, then finish self-check-in.
They can select a seat 24 hours before the flight departure.
1. They can check formatted information about airports, local time of destination and aircraft model of their tour.
2. They can check formatted information about all departures and arrivals.

Airport Employee

1. Airport Employees can identify the arrival time of a flight to prepare an unloading operation.
2. They can track delayed flights in the airport, checking flight status for each Flight.
3. They can monitor the passengers list for security reasons.
4. They can assign a boarding gate for a flight.
5. They can only view the aircrafts at the airport.

Airline Employee

1. Airline Employees can view passengers' full names and ages on a flight.
2. They can check passengers who need assistance.
3. They can check the number of available seats on a flight to sell tickets.
4. They can view the data of boarding time, the arrival time, seats and the distance of the flight.
5. Only they can add new aircrafts.
6. Only they can add a new flight and assign an aircraft to it.
7. They can change flight status.

5

2. User views

I) Passenger

1. Passengers can input the passport ID to view the future flights under their name. They need to retrieve the flight data from the PassengerTour class. The data includes: flightNumber, departureTime.
2. Passengers can self-check-in. They need to change the data of SeatCode from Seat class, CheckInStatus from PassengerTour class, and also need to use the method of AbleToSelectSeat() from PassengerTour class.
3. Passengers can cancel the flight. They need to change the data of CheckInStatus from the PassengerTour class.
4. Passengers can check formatted information about airports (including IATA airport code), local time of destination and aircraft model that they will fly in each flight by using the methods DisplayAirportCodePlusCity(), DisplayLocalTime() from Airport Class and DisplayAirCraftPlusAirline() from Aircraft Class.
5. Passengers can check a panel with formatted information about all departures and arrivals in the day by using the method DisplayFlightInformation() from Flight class.

II) Airport Employee

1. Airport Employees can track delayed flights in the airport, checking FlightStatus for each Flight using FlightStatus class..
2. They can assign a Boarding Gate for a flight using BoardingGate class.
3. They can monitor the passengers list for security reasons using Flight and PassengerTour classes.
4. They can identify the aircrafts at the airport using Aircraft and Flight classes.

6

2. User views

III) Airline Employee

1. Airline Employees can check passengers' full names and ages on a flight, they need to retrieve the data of fullname and age from the Passengers class.
2. They can check passengers who need assistance to board a flight by selecting the attribute field IsHelpRequired from Passenger Class.
3. They can check the number of available seats by using the method CalculateNAvailableSeats from the Flight Class.
4. They can view the data of boarding time, the arrivalTime, numberOfAvailableSeat and the distance from Flight class.
5. They can add a new aircraft that inherits properties from the parent class Aircraft Type.
6. They can add a new flight and assign an aircraft to it, they need to change the data of Flight class and Aircraft class.
7. They can change a flight's status by updating FlightStatus class and can use the CalculateDelayedDepTime method from Flight class to calculate the new departure time.

7

Major user views for the Airport database system

Data	Access Type	Airport Employee	Airline Employee	Passenger
Flight	Maintain		X	
	Query	X	X	X
Aircraft	Maintain		X	
	Query	X	X	X
Airport	Maintain	X		
	Query	X	X	X
Boarding Gate	Maintain	X		
	Query	X	X	X
Seat	Maintain		X	X
	Query	X	X	X

Maintain: enter, update, and delete.
Query: select

8

```

classDiagram
    class Airport {
        Code
        Name
        Country
        City
        Longitude
        Latitude
        TimeZone
        DisplayAirportCodePlusCity()
        DisplayLocatTime()
    }
    class Employee {
        <<abstract>>
        +AirportEmployees
        +AirlineEmployees
    }
    class AirportEmployees {
        Rate
    }
    class AirlineEmployees {
        Rate
    }
    class Person {
        ID
        FirstName
        LastName
        DateOfBirth
        TelephoneNumber
        Age()
        FullName()
    }
    class Passenger {
        Solution
        IsHelpRequired
    }
    class PassengerFour {
        BookingCode
        CheckinStatus
        AbleToSelectSeat()
    }
    class Seat {
        SeatCode
    }
    class Airline {
        AirlineID
        AirlineName
    }
    class Flight {
        FlightNumber
        ScheduledDepartureTime
        International
        FlightDuration
        CalculateActualTime()
        CalculateDistance()
        CalculateLoadingTime()
        CalculateAvailableSeats()
        DisplayFlightInformation()
        CalculateDelayedDepTime()
    }
    class FlightStatus {
        Status
        DelayMinutes
        LandedTime
    }
    class AirportType {
        Type
        MaxSpeed
        MaxDistance
        MailFuel
        NumberOfSeats
    }
    class Aircraft {
        AircraftID
        Hours flown
        DisplayAirCraftPlusAirline()
    }
    class BoardingGate {
        GateNumber
    }

    Airport "1" -- "*" Employee
    Airport "1" -- "*" Employee
    Employee <|-- AirportEmployees
    Employee <|-- AirlineEmployees
    Person <|-- Passenger
    Passenger "1" -- "*" PassengerFour
    PassengerFour "1" -- "1" Seat
    Airline "1" -- "*" Flight
    Flight "1" -- "1" FlightStatus
    Flight "1" -- "*" Airport
    Flight "1" -- "*" Airport
    Flight "1" -- "*" BoardingGate
    BoardingGate "1" -- "1" Airport
    BoardingGate "1" -- "1" Aircraft
    Aircraft "1" -- "*" Flight
    Aircraft "1" -- "*" AircraftType
  
```

The diagram illustrates the relationships between various entities in an airline system. Key entities include Airport, Employee (with subclasses AirportEmployees and AirlineEmployees), Person (with subclass Passenger), PassengerFour, Seat, Airline, Flight, FlightStatus, AirportType, and Aircraft. Relationships are defined with multiplicity and direction, such as "1" to "*" for one-to-many and "1" to "1" for one-to-one. Some relationships are labeled with roles like "works for", "operates", "is assigned to", and "belongs to".

Object Class	Method Name	Description
Flight	CalculateArrivalTime()	ScheduledDepartureTime + DelayInMinutes + FlightDuration + considering arrival airport timezone.
	CalculateDistance()	Longitude and Latitude distance between Airports using Haversine formula
	CalculateBoardingTime()	If isInternational = 1, BoardingTime starts 90 minutes before DepartureTime+DelayInMinutes. If isInternational = 0, BoardingTime starts 45 minutes before DepartureTime+DelayInMinutes.
	CalculateNAvailableSeats()	Aircraft Number of Seats - Number of Passengers
	DisplayFlightInformation()	Return String with Flight Code + Departure Airport + Arrival Airport + Delayed Departure Time + Arrival Time
	CalculateDelayedDepTime()	ScheduledDepartureTime + DelayInMinutes

4. Object Methods

Object Class	Method Name	Description
Person	Age()	Today's date - DateOfBirth
	FullName()	Return String "FirstName FullName"
Airport	DisplayAirportCodePlusCity()	Return String "AirportCode City"
	DisplayLocalTime()	Uses TimeZone to return the string
Aircraft	DisplayAirCraftPlusAirline()	Return String "AirCraftType AirlineName"
PassengerTour	AbleToSelectSeat()	Calculate if 24 hours before DepartureTime, Return Boolean

11

Questions?



12

1 1

