AN AIRLINE COMPANY DATABASE SYSTEM

Database Systems Assignment

Student Name: Ciara Power

Student Number: 20072488

Student Name: Kacper Wołoszyn

Student Number: 20071494

Table of Contents

Introduction

The topic chosen for the database design and development is a database system for an Airline company, for example Aer Lingus or Ryanair.

An airline company has various uses for a database, ranging from storing employee details, to scheduling a flight with components integrated from other sections of the database such as planes owned to be used for the flight, flight path to be taken for the flight, and the staff scheduled to work on the flight. These uses, amongst the many others, evidently emphasised the importance of an efficient and well-structured database system.

The model shows a variety of relationships between the tables in the database, as each relationship between areas of an airline is complex and has specific requirements involved. Each table in this model was carefully designed and implemented, with caution applied to constraint conditions and the many different specifications involved with an airline company's components.

Project Description

The database is created for the use of an individual airline, and contains tables documenting many areas such as planes owned by the company, scheduled flights by the company, flight paths set out by the company, employees of the company, and many more.

The connections between data in an airline database system are vital in maintaining the efficient database, which is key to a successful airline company. For example, an employee working for the company must have a contract associated with them, in order to ensure contract dates and proposed weekly hours are upheld.

Information must be stored about each plane, employee, airport, supply company, to name but a few. The details entered into the database for this project for items such as plane model specifications, airport details and flight path details, are as accurate as possible, with research having been done on each plane type and airport included for this demonstration.

Function

This database design has many functions for an airline company.

The database needs to be used to schedule flights, in order to connect a plane owned to each flight, assign a flight path to each flight, assign certain staff members to work on the scheduled flight, and to set out times and the duration of the flight scheduled.

Another major function of the database would be in regards to employment. As each employee's details is stored in the system, along with contract details and position details associated with them, the company can easily see the amount employed for each position. For example, if the company sees only two Pilots are employed, they may aim to hire more Pilots as a priority over other positions.

The database functions as storage for plane details of the planes owned, and the specifications of each possible plane model, which is vital when purchasing new planes (to ensure a suitable number of each plane model is owned). In this area of the company, details such as the maintenance date of each plane owned, and the capacity of each plane model are required to enable the company to upkeep health and safety standards and certificates.

In addition to the above, the database stores information regarding partner airports the company is linked with. Important details such as radio frequency and location are monumental in ensuring full cooperation and communication between the company and the airport. Each airline has associated supply companies which provide a variety of products, relevant to location. The airport connects to these companies, as there are certain companies located near certain airports, used for refuelling the airline's planes as they rest at each airport.

In order to successfully maintain the profitability of the airline as a business, a major function of the database involves dealing with bookings and customer details. Bookings must be carefully stored to ensure no mishaps occur, and each paying customer does not experience any difficulties or miscommunications in the process of booking a flight. In addition to this, each booking must be linked to a scheduled flight, and the passengers included in the booking must have details such as their passport number stored, because by law the airline must have these details before allowing a passenger board a flight.

Benefits

The benefits of implementing this database design for an airline company varies over a range of areas.

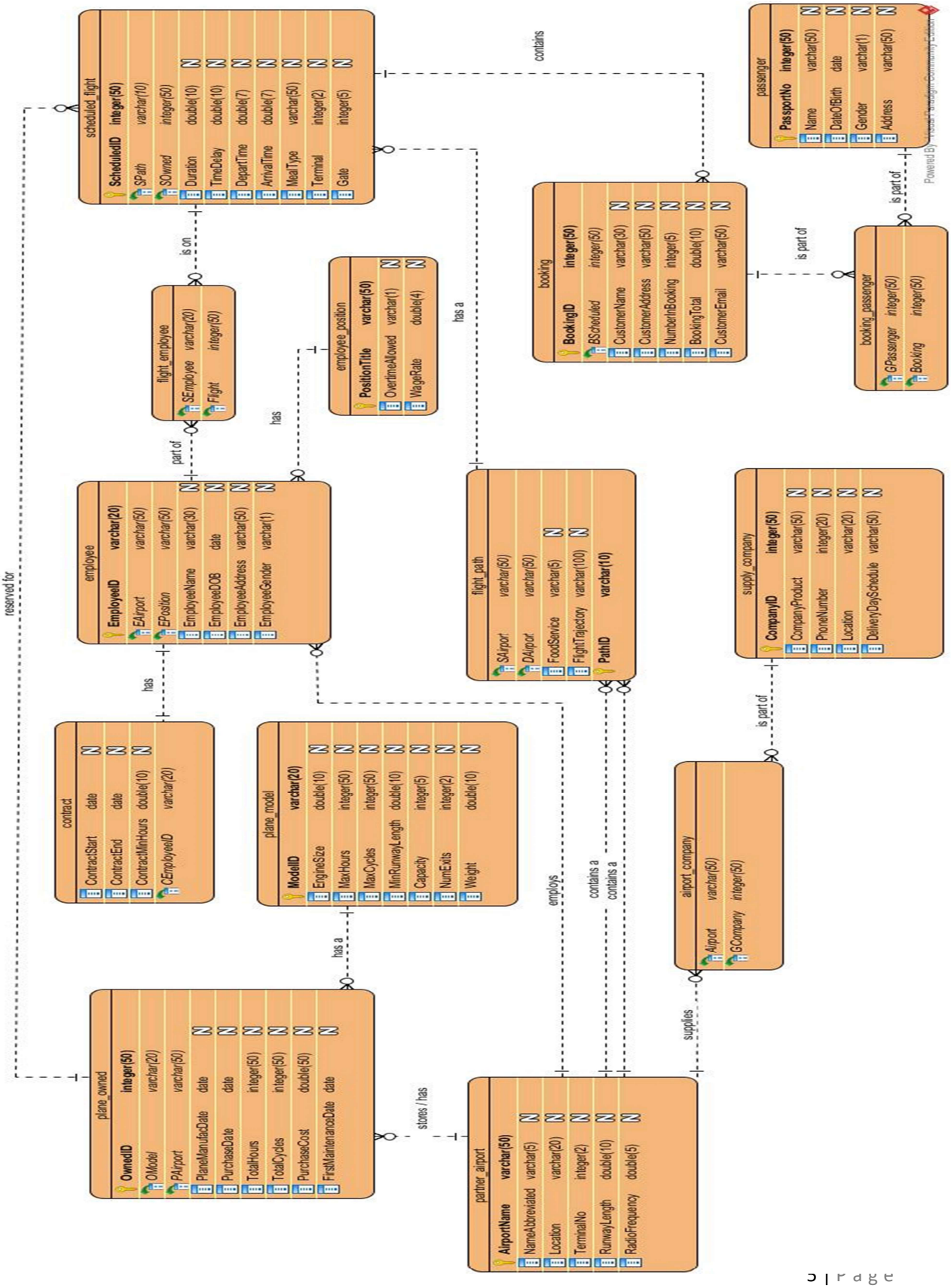
All expenditure for the company will be carefully planned out according to the current assets of the business, referring to aircraft, employees, supply company products, and partner airports. This will benefit the airline because expenditure will be monitored, which will lead to less unnecessary payments and increased profitability.

An airline company is highly dependent on accurate data and efficiently designed database systems to ensure no mistakes are made which potentially result in fatalities. As airlines work within a very high responsibility demanding sector, care must be taken to upkeep data entries and correctly use the database for the correct data retrieval. The list of benefits of this database system to the airline in this respect is endless, including facts such as the capacity of each plane for a flight being stored, so to ensure there is no overcrowding on planes which is very unsafe. Details such as radio frequency of an airport being stored is a huge benefit also as the plane will maintain contact with the destination and source airports, for inflight updates and resolving any issues that may occur during the flight.

The ease of scheduling flights will be much easier with details such as airport runway lengths and the plane model's minimum required runway lengths stored. The scheduler can ensure that the airport linked to a flight has the required runway length for the associated flight's scheduled plane, and if not, easily view the planes that have a suitable requirement for the airports involved in the flight, and change the scheduled plane to be used.

As mentioned above, the database has many benefits to an airline company, mainly in areas such as profitability, health and safety, in managing the company's business.

Model


scheduled_flight
ScheduledID integer(50)
SPath varchar(10)
SOwned integer(50)
Duration double(10)
TimeDelay double(10)
DepartTime double(7)
ArrivalTime double(7)
MealType varchar(50)
Terminal integer(2)
Gate integer(5)
passenger
PassportNo integer(50)
Name varchar(50)
DateOfBirth date
Gender varchar(1)
Address varchar(50)
booking
BookingID integer(50)
BScheduled integer(50)
CustomerName varchar(30)
CustomerAddress varchar(50)
NumberInBooking integer(5)
BookingTotal double(10)
CustomerEmail varchar(50)
booking_passenger
GPassenger integer(50)
Booking integer(50)
flight_employee
SEmployee varchar(20)
Flight integer(50)
employee_position
PositionTitle varchar(50)
OvertimeAllowed varchar(1)
WageRate double(4)
employee
EmployeeID varchar(20)
EAirport varchar(50)
EPosition varchar(50)
EmployeeName varchar(30)
EmployeeDOB date
EmployeeAddress varchar(50)
EmployeeGender varchar(1)
flight_path
SAirport varchar(50)
DAirport varchar(50)
FoodService varchar(5)
FlightTrajectory varchar(100)
PathID varchar(10)
supply_company
CompanyID integer(50)
CompanyProduct varchar(50)
PhoneNumber integer(20)
Location varchar(20)
DeliveryDaySchedule varchar(50)
contract
ContractStart date
ContractEnd date
ContractMinHours double(10)
CEmployeeID varchar(20)
plane_model
ModelID varchar(20)
EngineSize double(10)
MaxHours integer(50)
MaxCycles integer(50)
MinRunwayLength double(10)
Capacity integer(5)
NumExits integer(2)
Weight double(10)
airport_company
Airport varchar(50)
GCompany integer(50)
plane_owned
OwnedID integer(50)
OModel varchar(20)
PAirport varchar(50)
PlaneManufacDate date
PurchaseDate date
TotalHours integer(50)
TotalCycles integer(50)
PurchaseCost double(50)
FirstMaintenanceDate date
parther_airport
AirportName varchar(50)
NameAbbreviated varchar(5)
Location varchar(20)
TerminalNo integer(2)
RunwayLength double(10)
RadioFrequency double(5)
contains
is on
has a
has
part of
is part of
reserved for
employs
contains a
contains a
supplies
stores / has
Powered By

Model Description

- A plane is reserved for many scheduled flights whereas a scheduled flight has only one plane.
- A plane has one specific model, a plane model has many planes.
- A partner airport has and stores many planes owned, but a plane is stored at one partner airport.
- A partner airport employs many employees but employees are employed for one partner airport.
- A partner airport contains many flights path source airports, source airport of the flight path can only have 1 airport.
- A partner airport has many destination airports, and destination airport of the flight path have only one partner airport.
- A partner airport is supplied by a group of airport companies, but only one group supplies the airport.
- An airport company group can have one company that supplies it, but a company can be part of many groups.
- A flight path has many scheduled flights, i.e. one path many flights fly on, and a scheduled flight has one specific flight path.
- A scheduled flight has many flight employees, which are grouped in flight employee, but a specific group can be part of one flight.
- An employee can be part of many flight employee groups, but a flight employee group takes in one specific employee.
- An employee has one position he is responsible for, and positions have many employees.
- An employee has one specific contract she or he is on, a contract can be linked to one employee, and each contract is different for other employees.
- A scheduled flight contains many bookings, and bookings can be made for one specific scheduled flight.
- A passenger can be part of many booking groups, there is one booking group for a specific passenger.
- A group of passengers is on one booking, whereas a booking can have many booking groups.

Table Design

plane_owned			
NAME	**VARIABLE TYPE**	**CONSTRAINT**	**DESCRPTION**
OwnedID	Integer(50)	Primary key, not null	The ID of plane
OModel	Varchar(20)	Foreign key, not null	The model type
PAirport	Varchar(50)	Foreign key, not null	Where the plane is located
PlaneManufacDate	Date	PlaneManufacDate>'1974-01-01' AND PlaneManufacDate< SYSDATE()	Manufacture Date
PurchaseDate	Date	PurchaseDate > PlaneManufacDate AND PurchaseDate<= SYSDATE()	Purchase Date
TotalHours	Integer(50)		Hours spent in the air
TotalCycles	Integer(50)		Number of times it has completed a flight
FirstMaintenanceDate	Date	FirstMaintenanceDate>=PurchaseDate AND FirstMaintenanceDate<= SYSDATE()	The first maintenance date of plane
PurchaseCost	Double		How much the plane was

partner_airport			
NAME	**VARIABLE TYPE**	**CONSTRAINT**	**DESCRIPTION**
AirportName	Varchar(50)	Primary key, not null	Name of airport
NameAbbreviated	Varchar(5)		An abbreviation of the name of the airport
Location	Varchar(20)		Where the airport is located
TerminalNo	Integer(2)		Terminal Number
RunwayLength	Double		What length the runway is
RadioFrequency	Double		The radio frequency for an airport

plane_model			
NAME	**VARIABLE TYPE**	**CONSTRAINT**	**DESCRIPTION**
ModelID	Varchar(20)	Primary key, not null	The model ID of the plane
EngineSize	Double		Engine Size of the plane
MaxHours	Integer(50)		How many hours it can spend in the air
MaxCycles	Integer(50)		How many flights can a plane have
MinRunwayLength	Double		Minimum Runway length the plane can land on
Capacity	Integer(5)		How many people plane can hold
NumExits	Integer(2)		Number of exits on the plane
Weight	Double		Weight of the plane

flight_path			
NAME	**VARIABLE TYPE**	**CONSTRAINT**	**DESCRIPTION**
PathID	Varchar(10)	Primary key, not null	The Path id of a flight
SAirport	Varchar(50)	Foreign key, not null	The source airport
DAirport	Varchar(50)	Foreign key, not null	The destination airport
FoodService	Varchar(5)	FoodService="Yes" OR FoodService="No"	If food service is available on the flight path
FlightTrajectory	Varchar(100)		Which countries the flight takes place over

airport_company			
NAME	**VARIABLE TYPE**	**CONSTRAINT**	**DESCRIPTION**
Airport	Varchar(50)	Foreign key, not null	The name of the airport that a group of companies supplies to
GCompany	Integer(50)	Foreign key, not null	Group of companies that have been placed into this table

supply_company			
NAME	**VARIABLE TYPE**	**CONSTRAINT**	**DESCRIPTION**
CompanyID	Integer(50)	Primary key, not null	Company ID of the supply company
CompanyProduct	Varchar(50)		Product the company is supplying
PhoneNumber	Integer(20)		Phone number of company
Location	Varchar(20)		Location of company
DeliveryDaySchedule	Varchar(50)	DeliveryDaySchedule="Monday" OR "Tuesday" OR "Wednesday" OR "Thursday" OR "Friday" OR "Saturday" OR "Sunday"	Which day the delivery takes place

employee			
NAME	**VARIABLE TYPE**	**CONSTRAINT**	**DESCRIPTION**
EmployeeID	Varchar(20)	Primary key, not null	PPS number of the employee
EAirport	Varchar(50)	Foreign key, not null	The airport base the employee is based in
EPosition	Varchar(50)	Foreign key, not null	The position the employee oversees
EmployeeName	Varchar(30)		Name of employee
EmployeeDOB	Date	(SYSDATE()-EmployeeDOB>=18	Date of birth of employee, must be bigger then 18
EmployeeAddress	Varchar(50)		Address of Employee
EmployeeGender	Varchar(1)	EmployeeGender="M" OR EmployeeGender="F"	Gender of Employee

contract			
NAME	**VARIABLE TYPE**	**CONSTRAINT**	**DESCRIPTION**
CEmployeeID	Varchar(20)	Foreign key, not null	Employee ID
ContractStart	Date	ContractStart <= SYSDATE()	Start date of contract
ContractEnd	Date	ContractEnd > ContractStart AND ContractEnd>SYSDATE()	End date of contract
ContactMinHours	Double		Minimum Hours on contract

flight_employee			
NAME	**VARIABLE TYPE**	**CONSTRAINT**	**DESCRIPTION**
SEmployee	Varchar(20)	Foreign key, not null	The employee Id of one employee
Flight	Integer(50)	Foreign key, not null	Employees on a specific flight

employee_postion			
NAME	**VARIABLE TYPE**	**CONSTRAINT**	**DESCRIPTION**
PositionTitle	Varchar(50)	Primary key, not null	The position the employee has in the company
OvertimeAllowed	Varchar(1)	OvertimeAllowed="Y" OR OvertimeAllowed="N"	If overtime is allowed in a specific position, i.e. pilots due to a health risk are not allowed overtimes
WageRate	Double	wageRate>= 9.15	The amount the employee will earn at a specific position

scheduled_flight			
NAME	VARIABLE TYPE	CONSTRAINT	DESCRIPTION
ScheduledID	Integer(50)	Primary key, not null	The ID of the flight
SPath	Varchar(10)	Foreign key, not null	The path the flight will take
SOwned	Integer(50)	Foreign key, not null	The plane owned that will be flown on this specific flight
Duration	Double		Duration of flight
TimeDelay	Double		If the flight has a time delay in decimal point
DepartTime	DateTime	DepartTime<SYSDATE()	Departure time of the flight
ArrivalTime	DateTime	ArrivalTime > DepartTime	The arrival time of the flight
MealType	Varchar(50)	MealType= "Breakfast" OR "Lunch" OR "Dinner" or "None"	The meal type on a scheduled flight
Terminal	Integer(2)		Terminal of the arrival airport
Gate	Integer(5)		Gate of the specified terminal
ArrivalTime	DateTime	ArrivalTime > DepartTime	The arrival time of the flight

booking			
NAME	**VARIABLE TYPE**	**CONSTRAINT**	**DESCRIPTION**
BookingID	Integer(50)	Primary key, not null	The id of the booking
BScheduled	Integer(50)	Foreign key, not null	The id of the scheduled flight
CustomerName	Varchar(30)		Name of customer
CustomerAddress	Varchar(50)		Address of customer
NumberInBooking	Integer(5)	NumberInBooking !=0	The number of people in a booking
BookingTotal	Double	BookingTotal !=0	Total amount
CustomerEmail	Varchar(50)		Email of customer

booking_passenger			
NAME	**VARIABLE TYPE**	**CONSTRAINT**	**DESCRIPTION**
GPassenger	Integer(50)	Foreign key, not null	Passport number of the booked passenger
Booking	Integer(50)	Foreign key, not null	Booking Id of the flight

passenger			
NAME	**VARIABLE TYPE**	**CONSTRAINT**	**DESCRIPTION**
PassportNo	Integer(50)	Primary Key, not Null	Passport number of passenger
Name	Varchar(50)		Name of passenger
DateOfBirth	Date	DateOfBirth<SYSDATE()	Date of birth of passenger
Gender	Varchar(1)	Gender="M" OR Gender="F"	Gender of passenger
Address	Varchar(50)		Passenger's address