Documentatiom

For G18 Airlines

Group 18

CMPG 223

Group Members:

Jacques Nel – 31986595

Erika Haasbroek – 37673149

Xavier Labuschagne – 34246606

Lesedi Taunyane – 28877233

Mecaylla Beukes - 28331869

Contents

[Project Description 2](#_Toc114232687)

[Scope Definition 2](#_Toc114232688)

[Functional Requirements: 2](#_Toc114232689)

[Non-functional Requirements: 2](#_Toc114232690)

[Physical Data Model 3](#_Toc114232691)

[Physical Process Model 4](#_Toc114232692)

[Database schema 5](#_Toc114232693)

[SQL used: 6](#_Toc114232694)

[CREATE Database: 6](#_Toc114232695)

[CREATE Tables: 6](#_Toc114232696)

[INSERT Statements: 7](#_Toc114232697)

[Some Of the SQL Used in Code (C# Visual Studio): 8](#_Toc114232698)

[Adding Flights User Input: 8](#_Toc114232699)

[Adding Members User Input: 8](#_Toc114232700)

[Adding Booking User Input: 8](#_Toc114232701)

[Example Code: 9](#_Toc114232702)

[Reports Screenshots: 11](#_Toc114232703)

[Make Bookings and Add Members: 11](#_Toc114232704)

[Management Page: 12](#_Toc114232705)

[User Manual: 12](#_Toc114232706)

[Installation 12](#_Toc114232707)

[Visual Studio installation steps: 12](#_Toc114232708)

[System requirements for server 13](#_Toc114232709)

[System requirements for user 13](#_Toc114232710)

[Diary 13](#_Toc114232711)

# Project Description

The main object of this project is to provide a system that G18 Airlines employees and management can use to easily make in person bookings at the airport/counters at the airport, add and delete flights, add, and delete members (Customers who wish to become members at G18 airlines). This system will help Management look up information with ease as well as allowing the generation of reports. The System will be designed in C# and SQL Server.

# Scope Definition

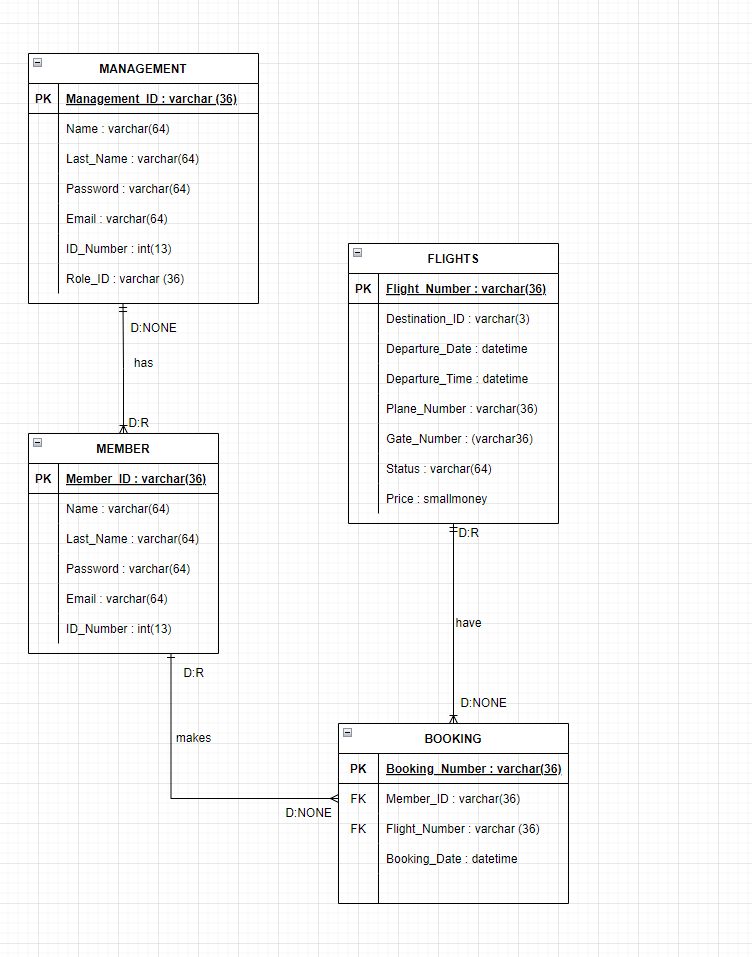
## Functional Requirements:

* Maintenance of bookings by employees.
* Generate reports for management.
* Maintenance of members.
* Maintenance of flights.

## Non-functional Requirements:

* Performance.
* Data integrity and information longevity.
* Finances.
* Control and/or improvement.
* Efficiency of the process.
* Role-based access system for different level users.
* Service.
* The system has 2 types of users:
  + **Employees:** These users will include airport staff such as security, cleaners, flight attendants, baggage handlers, and counter assistants who need to view client information and flight information as well as booking information.
  + **Airport (Higher level employees/Management):** This role will primarily be given to senior staff members, managers and counter assistants who need access to customers and flight information.

# Physical Data Model

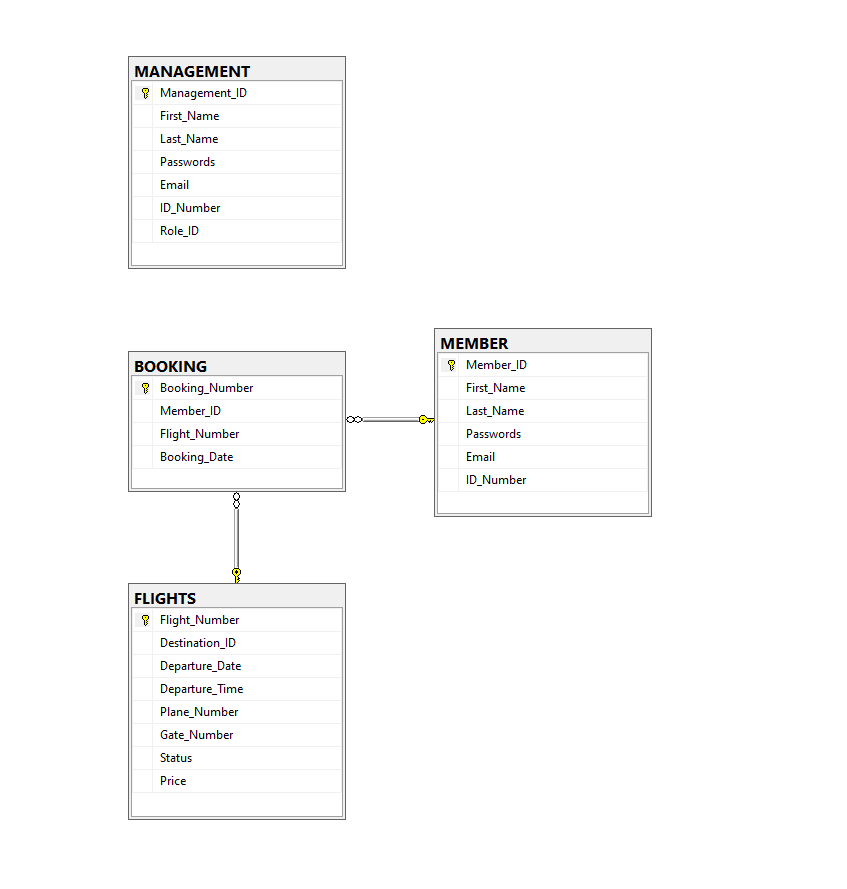


# Physical Process Model

Diagram, schematic

Description automatically generated

# Database schema



# SQL used:

## CREATE Database:

DROP DATABASE IF EXISTS AIRPORT2;

GO

CREATE DATABASE AIRPORT2;

GO

## CREATE Tables:

USE AIRPORT2

GO

CREATE TABLE MANAGEMENT (

Management\_ID VARCHAR(36) PRIMARY KEY,

First\_Name VARCHAR(64),

Last\_Name VARCHAR(64),

Passwords VARCHAR(64),

Email VARCHAR(64),

ID\_Number INT,

Role\_ID VARCHAR(36)

);

CREATE TABLE MEMBER (

Member\_ID VARCHAR(36) PRIMARY KEY,

First\_Name VARCHAR(64),

Last\_Name VARCHAR (64),

Passwords VARCHAR(64),

Email VARCHAR(64),

ID\_Number INT

);

CREATE TABLE FLIGHTS (

Flight\_Number VARCHAR(36) PRIMARY KEY,

Destination\_ID VARCHAR(3),

Departure\_Date DATETIME,

Departure\_Time DATETIME,

Plane\_Number VARCHAR(36),

Gate\_Number VARCHAR(36),

Status VARCHAR(64),

Price SMALLMONEY

);

CREATE TABLE BOOKING (

Booking\_Number VARCHAR(36) PRIMARY KEY,

Member\_ID VARCHAR(36) FOREIGN KEY REFERENCES MEMBER(Member\_ID),

Flight\_Number VARCHAR(36) FOREIGN KEY REFERENCES FLIGHTS(Flight\_Number),

Booking\_Date DATETIME

);

## INSERT Statements:

USE AIRPORT2

GO

INSERT INTO MEMBER (Member\_ID, First\_Name, Last\_Name, Passwords, Email, ID\_Number)

VALUES ('MEM10', 'Kobus', 'Jan', 1234, 'kobus@gmail.com', 1478965)

INSERT INTO FLIGHTS (Flight\_Number, Destination\_ID, Departure\_Date, Departure\_Time, Plane\_Number, Gate\_Number, Status, Price)

VALUES ('FT555', 'CPT', '10-05-2022', null, 'P1', 'G1', 'Available', 2500)

INSERT INTO MANAGEMENT (Management\_ID, First\_Name, Last\_Name, Passwords, Email, ID\_Number, Role\_ID)

VALUES ('MAN1', 'Jan', 'Fourie', 2582, 'jan@gmail.com', 258741, 'Manager')

INSERT INTO BOOKING (Booking\_Number, Member\_ID, Flight\_Number, Booking\_Date)

VALUES ('B11', 'MEM10', 'FT555', 22-09-15)

## Some Of the SQL Used in Code (C# Visual Studio):

string SQL = "SELECT \* FROM FLIGHTS";

string SQL = "SELECT DISTINCT Flight\_Number FROM FLIGHTS";

string SQL = "SELECT \* FROM BOOKING";

string SQL = "SELECT DISTINCT Booking\_Number FROM BOOKING";

string SQL = "SELECT \* FROM MEMBER";

string SQL = "SELECT DISTINCT Member\_ID FROM MEMBER";

string SQL\_2 = $"DELETE FROM FLIGHTS WHERE Flight\_Number = '{ComboBox\_Val}'";

string SQL\_2 = $"DELETE FROM BOOKING WHERE Booking\_Number = '{ComboBox\_Val}'";

string SQL\_2 = $"DELETE FROM MEMBER WHERE Member\_ID = '{ComboBox\_Val}'";

### Adding Flights User Input:

string SQL = "INSERT INTO FLIGHTS (Flight\_Number, Destination\_ID, Departure\_Date, Plane\_Number, Gate\_Number, Status, Price) VALUES(@Flight\_Number, @Destination\_ID, @Departure\_Date, @Plane\_Number, @Gate\_Number, @Status, @Price)";

### Adding Members User Input:

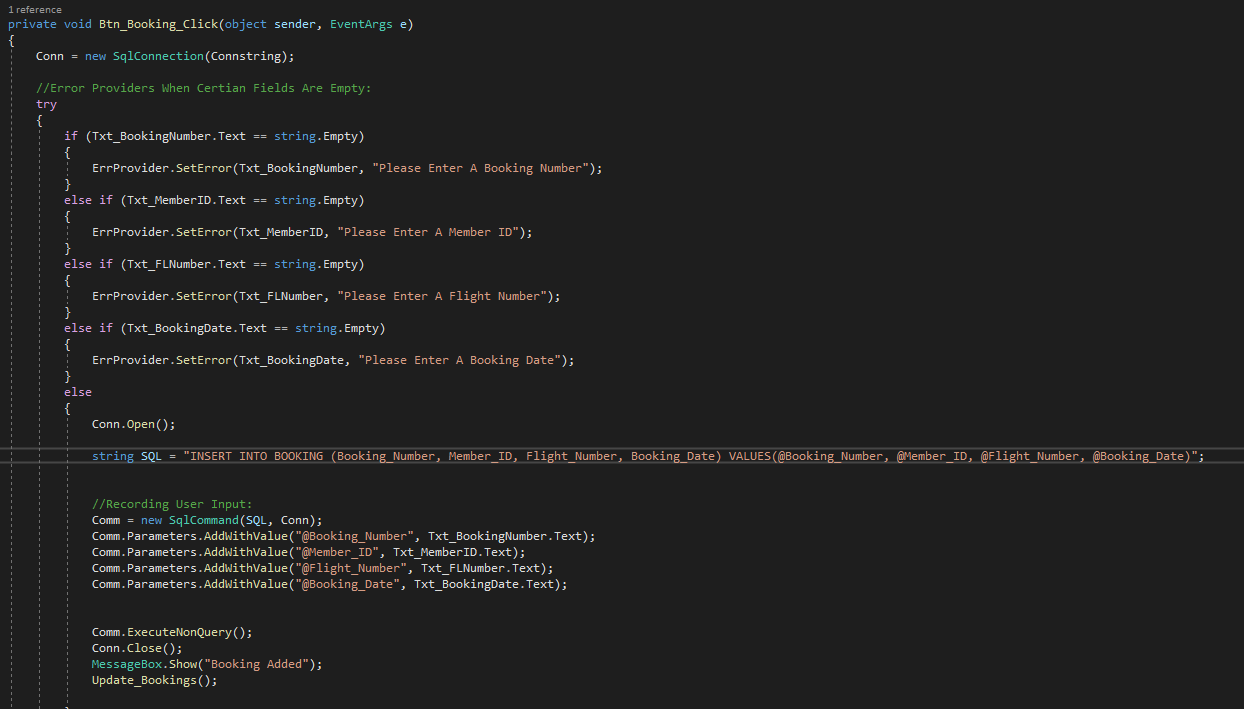
string SQL = "INSERT INTO MEMBER (Member\_ID, First\_Name, Last\_Name, Email, ID\_Number) VALUES (@Member\_ID, @First\_Name, @Last\_Name, @Email, @ID\_Number)";

### Adding Booking User Input:

string SQL = "INSERT INTO BOOKING (Booking\_Number, Member\_ID, Flight\_Number, Booking\_Date) VALUES(@Booking\_Number, @Member\_ID, @Flight\_Number, @Booking\_Date)";

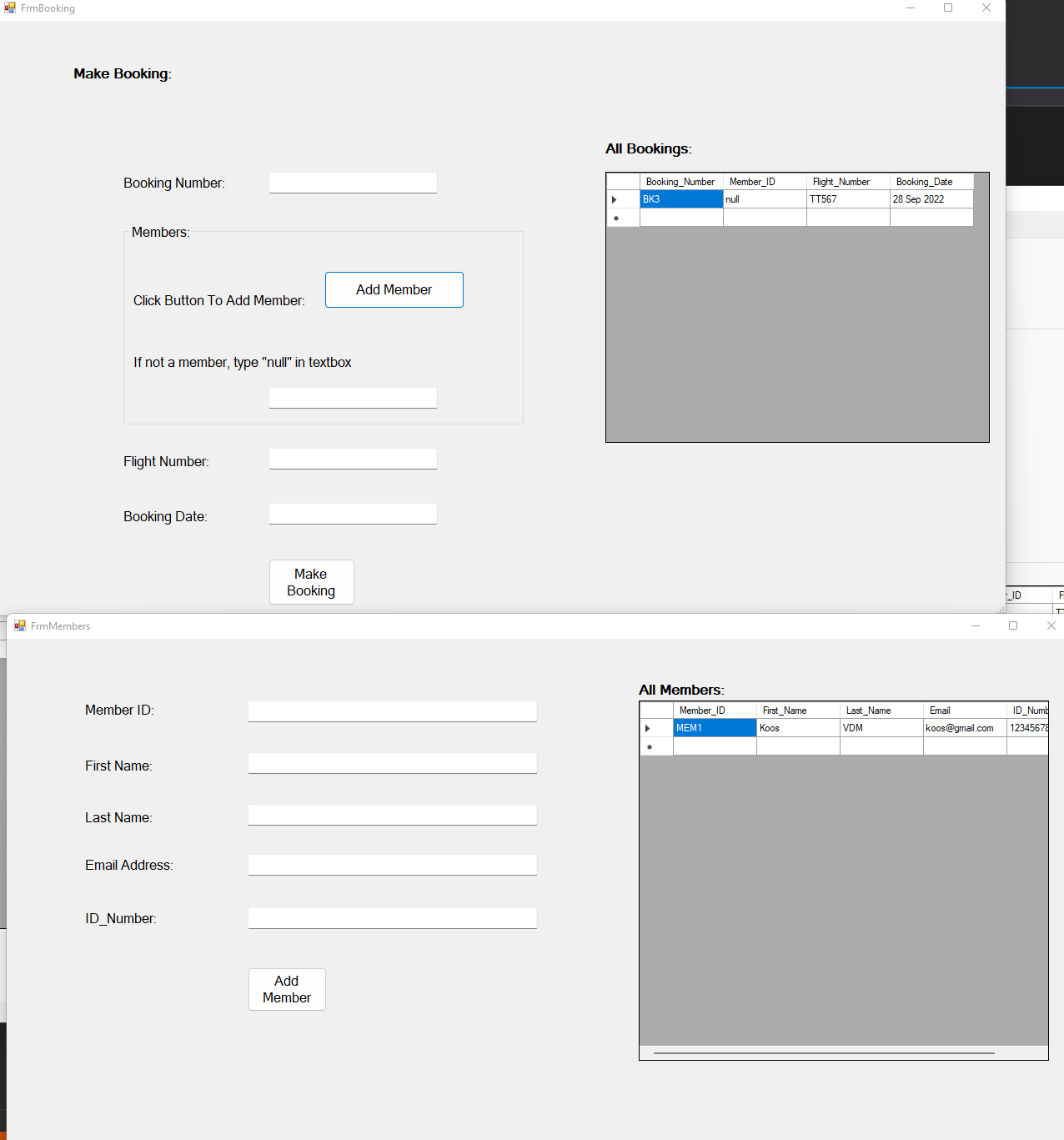
# Example Code:



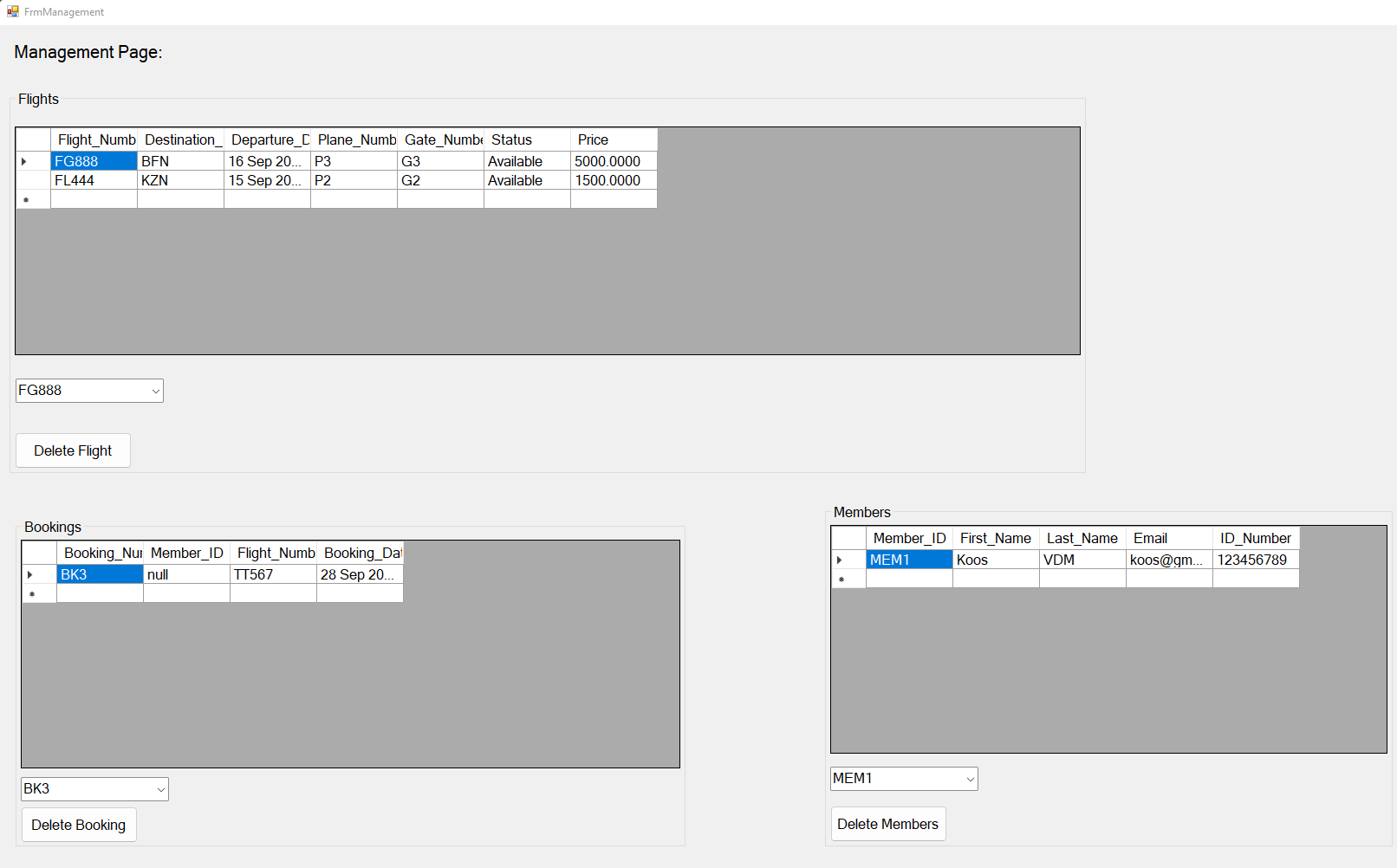


# Reports Screenshots:

## Make Bookings and Add Members:



## Management Page:



# User Manual:

## Installation

For the installation of this system the user will need to download Visual Studio for Windows. After the installation of Visual Studio, the user should open the folder “G18 Airlines” in Visual Studio so that the system can start.

### Visual Studio installation steps:

1. Download Visual Studio
2. Open the .exe file
3. Start the installation
4. Wait for installation to finish
5. Choose any software version and click install
6. Select desktop version on the screen after you have chosen the software version
7. Wait for the files to finish downloading
8. Restart the PC
9. Open Visual Studio
10. Open “G18 Airlines” folder in Visual Studio

### System requirements for server

**Processor:** Any Intel or AMD processors

**Memory:** 4gb ram

**Operating System:** Windows 10 or newer

**Graphics card:** Intel based on board graphics card

**Storage:** 100GB SSD, 2TB HDD.

### System requirements for user

**Processor:** Any Intel or AMD processors

**Memory:** 8GB ram

**Operating System:** Windows 10 or newer

**Graphics card:** Intel based on board graphics card

**Storage:** 100GB SSD

# Diary

GitHub Commits: On the the next page.

Graphical user interface, text, application

Description automatically generated