2021

# CMPG213

# Project

# Documentation

### For Murphy’s Airport

# Project Description:

The main objective of this project is to provide a system that murphy’s airlines customers can use to easily make bookings. This system will help agents look up information with ease as well as allow airport management to generate desired reports.

# Scope Definition:

Functional Requirements:

* Maintenance of bookings by users
* Maintenance of bookings by agents
* Generate reports for management
* Maintenance of members
* Maintenance of flights
* Maintenance of planes
* Maintenance of destinations

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 will have 3 types of users:
  + Customer: These are users who wish to book chartered flights and will have access to an online platform where they can make bookings and process payments.
  + 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.
  + Airport (Higher level employees): This role will primarily be given to senior staff members, managers and counter assistants who need access to customers and flight information

# Index:

Table of Contents

[CMPG213 1](#_Toc84805402)

[Project 1](#_Toc84805403)

[Documentation 1](#_Toc84805404)

[For Murphy’s Airport 1](#_Toc84805405)

[Project Description: 2](#_Toc84805406)

[Scope Definition: 2](#_Toc84805407)

[Index: 3](#_Toc84805408)

[Physical Data Model 4](#_Toc84805409)

[Physical Process Model 5](#_Toc84805410)

[Database schema 6](#_Toc84805411)

[SQL used: 6](#_Toc84805412)

[Creation script: 6](#_Toc84805413)

[Database Population 8](#_Toc84805414)

[Maintenance of tables (Insert, Update, Delete) 13](#_Toc84805415)

[Querying database (Select) 15](#_Toc84805416)

[Example of code: 16](#_Toc84805417)

[Reports Screenshots: 17](#_Toc84805418)

[User Manual: 18](#_Toc84805419)

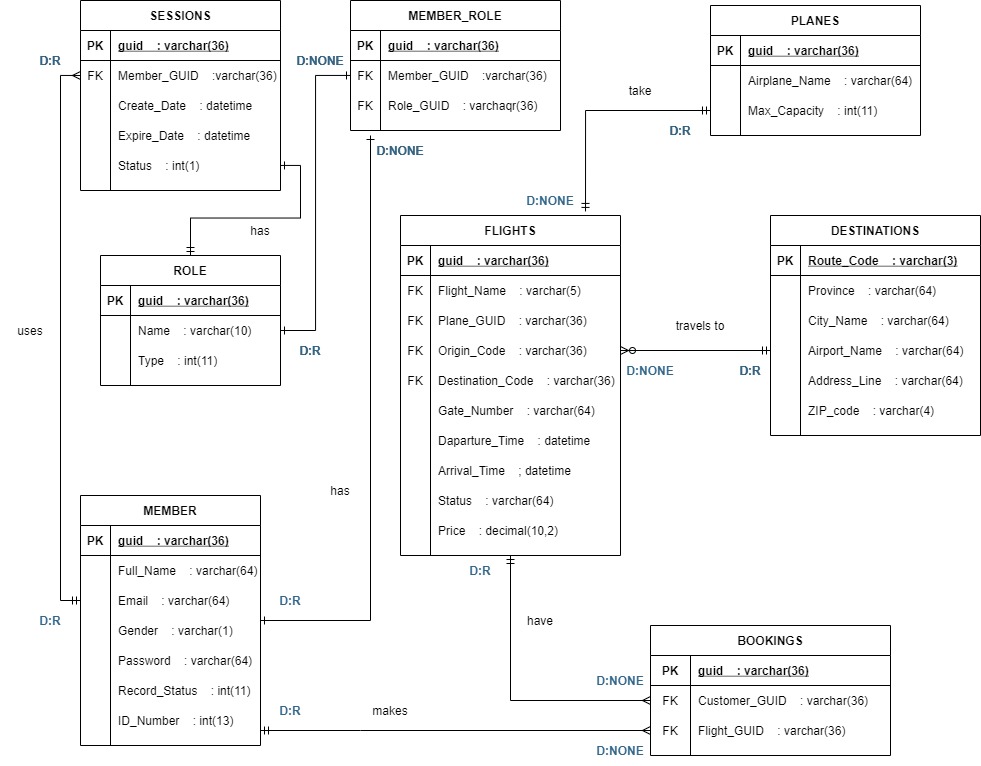
[Installation 18](#_Toc84805420)

[System Requirements for server: 18](#_Toc84805421)

[System Requirements for user: 19](#_Toc84805422)

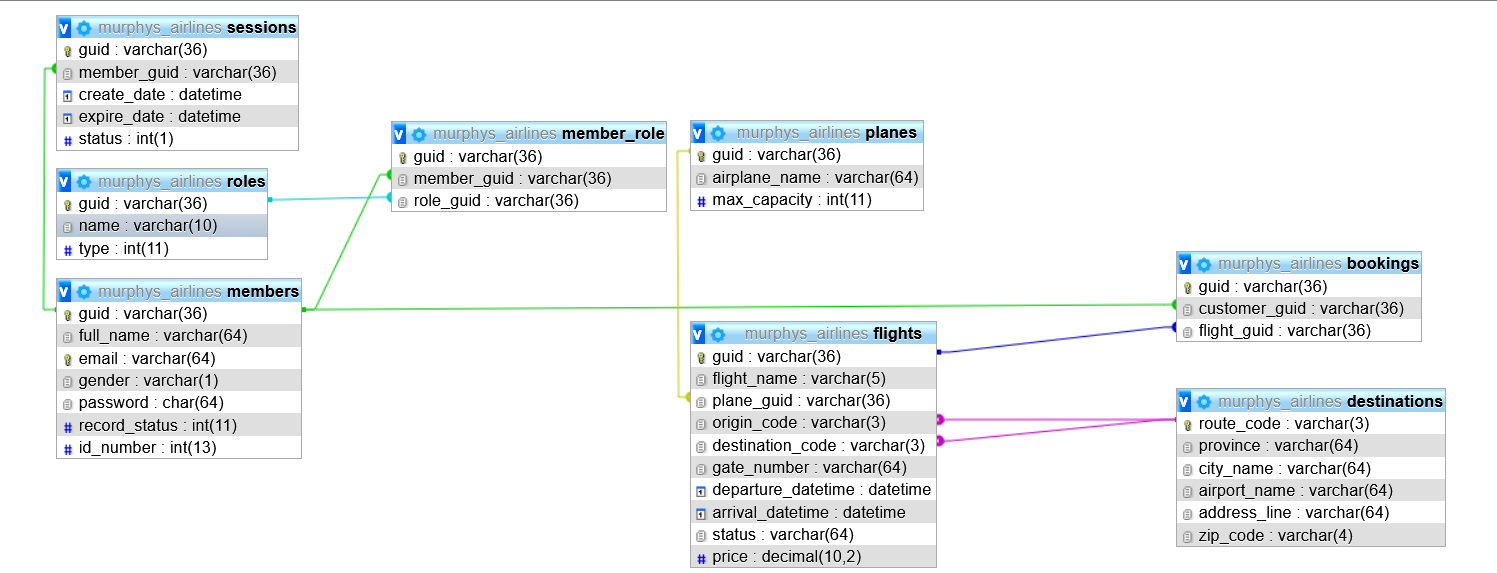
[Diary: 19](#_Toc84805423)

# Physical Data Model



# Physical Process Model

# Database schema



# SQL used:

## Creation script:

CREATE DATABASE `murphys\_airlines`;

USE `murphys\_airlines`;

CREATE TABLE `members` (

`guid` varchar(36),

`full\_name` varchar(64),

`email` varchar(64),

`gender` varchar(1),

`password` char(64),

`record\_status` int(11),

`id\_number` int(13),

Primary Key (`guid`),

Unique Key (`email`));

CREATE TABLE `sessions` (

`guid` varchar(36),

`member\_guid` varchar(36),

`create\_date` datetime DEFAULT CURRENT\_TIMESTAMP,

`expire\_date` datetime,

`status` int(1),

Primary Key (`guid`),

Foreign Key (`member\_guid`) references `members` (`guid`));

CREATE TABLE `roles` (

`guid` varchar(36),

`name` varchar(10),

`type` int(11),

Primary Key (`guid`));

CREATE TABLE `destinations` (

`route\_code` varchar(3),

`province` varchar(64),

`city\_name` varchar(64),

`airport\_name` varchar(64),

`address\_line` varchar(64),

`zip\_code`varchar(4),

Primary Key (`route\_code`));

CREATE TABLE `planes` (

`guid` varchar(36),

`airplane\_name` varchar(64),

`max\_capacity` int(11),

Primary Key (`guid`));

CREATE TABLE `flights` (

`guid` varchar(36),

`flight\_name` varchar(5),

`plane\_guid` varchar(36),

`origin\_code` varchar(3),

`destination\_code` varchar(3),

`gate\_number` varchar(64),

`departure\_datetime` datetime,

`arrival\_datetime` datetime,

`status` varchar(64),

`price` decimal(10,2),

Primary Key (`guid`),

Foreign Key (`plane\_guid`) references `planes` (`guid`),

Foreign Key (`origin\_code`) references `destinations` (`route\_code`),

Foreign Key (`destination\_code`) references `destinations` (`route\_code`));

CREATE TABLE `member\_role` (

`guid` varchar(36),

`member\_guid` varchar(36),

`role\_guid` varchar(36),

Primary Key (`guid`),

Foreign Key (`member\_guid`) references `members` (`guid`),

Foreign Key (`role\_guid`) references `roles` (`guid`));

CREATE TABLE `bookings` (

`guid` varchar(36),

`customer\_guid` varchar(36),

`flight\_guid` varchar(36),

Primary Key (`guid`),

Foreign Key (`customer\_guid`) references `members` (`guid`),

Foreign Key (`flight\_guid`) references `flights` (`guid`));

## Database Population

DELETE FROM `bookings`;

DELETE FROM `member\_role`;

DELETE FROM `flights`;

DELETE FROM `planes`;

DELETE FROM `destinations`;

DELETE FROM `roles`;

DELETE FROM `sessions`;

DELETE FROM `members`;

INSERT INTO `destinations` (`route\_code`, `province`, `city\_name`, `airport\_name`, `address\_line`, `zip\_code`)

VALUES

('BFN', 'Free State', 'Bloemfontein', 'Bram Fischer International Airport', 'Bloemfontein', '9300'),

('CPT', 'Western Cape', 'Cape Town', 'Cape Town International Airport', 'Matroosfontein', '7490'),

('DUR', 'Eastern Cape', 'Durban', 'King Shaka International Airport', 'King Shaka Dr, La Mercy', '4407'),

('ELS', 'Eastern Cape', 'East London', 'East London Airport', 'East London', '5200'),

('GRJ', 'Western Cape', 'George', 'George Airport', 'George', '6530'),

('HLA', 'Gauteng', 'Johannesburg', 'Lanseria International Airport', 'Airport Rd, Lanseria', '1748'),

('JNB', 'Gauteng', 'Johannesburg', 'O.R. Tambo International Airport', '1 Jones Rd, Kempton Park', '1632'),

('KIM', 'Northern Cape', 'Kimberley', 'Kimberley Airport', 'Comp Patterson Rd, Diskobolos', '8300'),

('MQP', 'Mpumalanga', 'Mbombela', 'Kruger Mpumalanga International Airport', 'R538, Mbombela', '1200'),

('PLZ', 'Eastern Cape', 'Port Elizabeth', 'Port Elizabeth International Airport', 'Allister Miller Dr, Walmer, Gqeberha', '6070'),

('UTN', 'Northern Cape', 'Upington', 'Upington International Airport', 'Diedericks Street', '8801');

-- Password: MurphysAirlines123

-- RecordStatus:

-- 0: Inactive

-- 1: Active

-- 2: Disabled

INSERT INTO `members`(`guid`, `full\_name`, `email`, `gender`, `password`, `record\_status`, `id\_number`)

VALUES

-- Admin

(UUID(), 'Markus Marais', 'markus.marais@outlook.com', 'M', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 0123456789123),

(UUID(), 'Arthur de Villiers', 'arthurdevilliers@gmail.com', 'M', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 3219876543210),

(UUID(), 'Marco van der Merwe', 'marcovandermerwe2000@gmail.com', 'M', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 2549863024521),

(UUID(), 'Denzil Richter', '28885198@student.g.nwu.ac.za', 'M', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 0257896543210),

(UUID(), 'Jacques Nel', 'jacquesnel85@gmail.com', 'M', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 3219876543210),

-- Moderators

(UUID(), 'Clarissa Robbins', 'clarissa@robbins.com', 'F', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 6589423150321),

(UUID(), 'Wanda Day', 'wanda@day.com', 'F', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 6589523150321),

(UUID(), 'Winston Romero', 'winston@romero.com', 'M', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 6789523150321),

-- Clients

(UUID(), 'Wallace Jenkins', 'wallace@jenkins.com', 'M', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 6589523140321),

(UUID(), 'Ned Carpenter', 'ned@carpenter.com', 'M', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 6589523140365),

(UUID(), 'Noelle Fuller', 'noelle@fuller.com', 'M', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 3569523140365),

(UUID(), 'Herbert Swanson', 'herbert@swanson.com', 'M', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 3565523140365),

(UUID(), 'Charlie McGee', 'charlie@mcgee.com', 'F', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 3269523140365),

(UUID(), 'Lonnie Hendrix', 'lonnie@hendrix.com', 'F', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 3989523140365),

(UUID(), 'Luis Bryant', 'luis@bryant.com', 'M', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 3569523143265),

(UUID(), 'Isaiah Holland', 'isaiah@holland.com', 'F', '69b13ce3c2e5081b30c399ba0bed00e56fdab4d827edc4e711da512b3dad8994', 1, 3569483140365);

INSERT INTO `roles`(`guid`, `name`, `type`)

VALUES

(UUID(), 'ADMIN', 0),

(UUID(), 'MOD', 1),

(UUID(), 'CLIENT', 2);

INSERT INTO `member\_role`(`guid`, `member\_guid`, `role\_guid`)

VALUES

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'markus.marais@outlook.com'), (SELECT `guid` FROM `roles` WHERE `name` = 'ADMIN')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'arthurdevilliers@gmail.com'), (SELECT `guid` FROM `roles` WHERE `name` = 'ADMIN')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'marcovandermerwe2000@gmail.com'), (SELECT `guid` FROM `roles` WHERE `name` = 'ADMIN')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = '28885198@student.g.nwu.ac.za'), (SELECT `guid` FROM `roles` WHERE `name` = 'ADMIN')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'jacquesnel85@gmail.com'), (SELECT `guid` FROM `roles` WHERE `name` = 'ADMIN')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'clarissa@robbins.com'), (SELECT `guid` FROM `roles` WHERE `name` = 'MOD')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'wanda@day.com'), (SELECT `guid` FROM `roles` WHERE `name` = 'MOD')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'winston@romero.com'), (SELECT `guid` FROM `roles` WHERE `name` = 'MOD')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'wallace@jenkins.com'), (SELECT `guid` FROM `roles` WHERE `name` = 'CLIENT')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'ned@carpenter.com'), (SELECT `guid` FROM `roles` WHERE `name` = 'CLIENT')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'noelle@fuller.com'), (SELECT `guid` FROM `roles` WHERE `name` = 'CLIENT')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'herbert@swanson.com'), (SELECT `guid` FROM `roles` WHERE `name` = 'CLIENT')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'charlie@mcgee.com'), (SELECT `guid` FROM `roles` WHERE `name` = 'CLIENT')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'lonnie@hendrix.com'), (SELECT `guid` FROM `roles` WHERE `name` = 'CLIENT')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'luis@bryant.com'), (SELECT `guid` FROM `roles` WHERE `name` = 'CLIENT')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'isaiah@holland.com'), (SELECT `guid` FROM `roles` WHERE `name` = 'CLIENT'));

INSERT INTO `planes`(`guid`, `airplane\_name`, `max\_capacity`)

VALUES

(UUID(), 'Frostfire', 15),

(UUID(), 'Swiftcharge', 10),

(UUID(), 'Dreamlance', 20),

(UUID(), 'Tiny Dragonfly', 15),

(UUID(), 'Dragonpyre', 30);

-- Status:

-- 0: Scheduled

-- 1: En Route

-- 2: Check-In

-- 3: Boarding

-- 4: Landed

INSERT INTO `flights`(`guid`, `flight\_name`, `plane\_guid`, `origin\_code`, `destination\_code`, `gate\_number`, `departure\_datetime`, `arrival\_datetime`, `status`, `price`)

VALUES

(UUID(), 'FT123', (SELECT `guid` FROM `planes` WHERE `airplane\_name` = 'Frostfire'), 'BFN', 'CPT', 'A01', '2020-01-01 07:00:00', '2020-01-01 12:00:00', 0, 5000.00),

(UUID(), 'SA237', (SELECT `guid` FROM `planes` WHERE `airplane\_name` = 'Swiftcharge'), 'ELS', 'CPT', 'C23', '2020-05-04 02:00:00', '2020-05-05 03:00:00', 1, 4500.00),

(UUID(), 'KJ358', (SELECT `guid` FROM `planes` WHERE `airplane\_name` = 'Dreamlance'), 'BFN', 'DUR', 'B03', '2021-02-02 20:00:00', '2021-02-03 21:30:00', 2, 7500.00),

(UUID(), 'JY098', (SELECT `guid` FROM `planes` WHERE `airplane\_name` = 'Tiny Dragonfly'), 'UTN', 'JNB', 'D21', '2019-12-24 15:45:00', '2019-12-25 00:00:00', 3, 2300.00),

(UUID(), 'LO034', (SELECT `guid` FROM `planes` WHERE `airplane\_name` = 'Dragonpyre'), 'KIM', 'PLZ', 'A10', '2021-10-31 13:15:00', '2020-11-01 00:00:00', 4, 1500.00);

INSERT INTO `bookings`(`guid`, `customer\_guid`, `flight\_guid`)

VALUES

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'wallace@jenkins.com'), (SELECT `guid` FROM `flights` WHERE `flight\_name` = 'FT123')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'ned@carpenter.com'), (SELECT `guid` FROM `flights` WHERE `flight\_name` = 'SA237')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'noelle@fuller.com'), (SELECT `guid` FROM `flights` WHERE `flight\_name` = 'SA237')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'luis@bryant.com'), (SELECT `guid` FROM `flights` WHERE `flight\_name` = 'KJ358')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'charlie@mcgee.com'), (SELECT `guid` FROM `flights` WHERE `flight\_name` = 'LO034')),

(UUID(), (SELECT `guid` FROM `members` WHERE `email` = 'isaiah@holland.com'), (SELECT `guid` FROM `flights` WHERE `flight\_name` = 'KJ358'));

## Maintenance of tables (Insert, Update, Delete)

#### Table Flights:



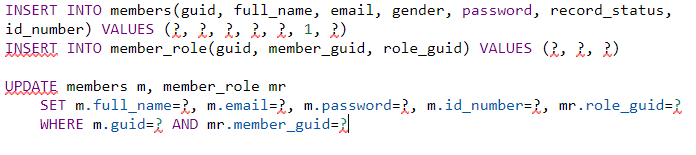
#### Table Planes:

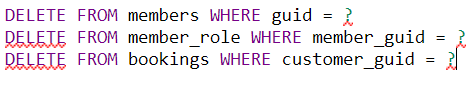


#### Table Destinations:

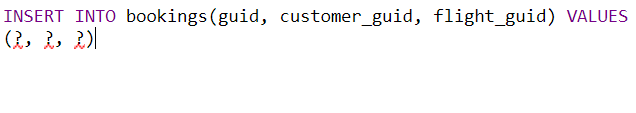


#### Table Members:





#### Table bookings:





## Querying database (Select)

#### Table Flights:



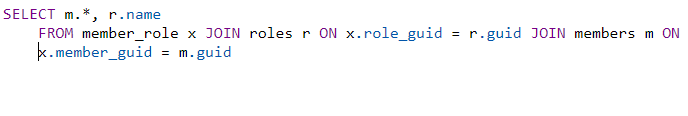
#### Table Planes:



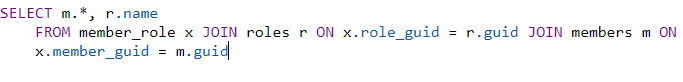
#### Table Destinations:



#### Table bookings:



#### Table Members:



# Example of code:

class StandardResponse {

    /\*\*

     \* Always use this object when sending a response to the user

     \* @constructor

     \* @param {string} pResult - 0:error 1:success

     \* @param {string} pData - Any data for the front end to use.

     \* @param {string} pProgrammerMessage - A message for the programmer ('The instance on node 152 has lost contact').

     \* @param {string} pFrontEndMessage - A message for the front end user ('The service has lost its internet connection').

    \*/

    constructor(pResult, pData, pProgrammerMessage, pFrontEndMessage) {

        if (pResult == 0) this.result = 'error';

        if (pResult == 1) this.result = 'success';

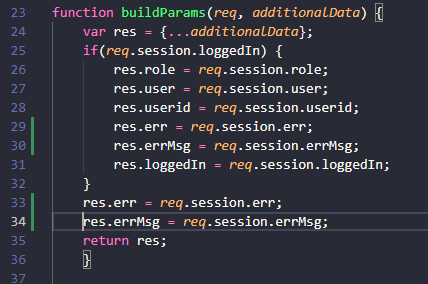
        this.data = pData;

        this.programmerMessage = pProgrammerMessage;

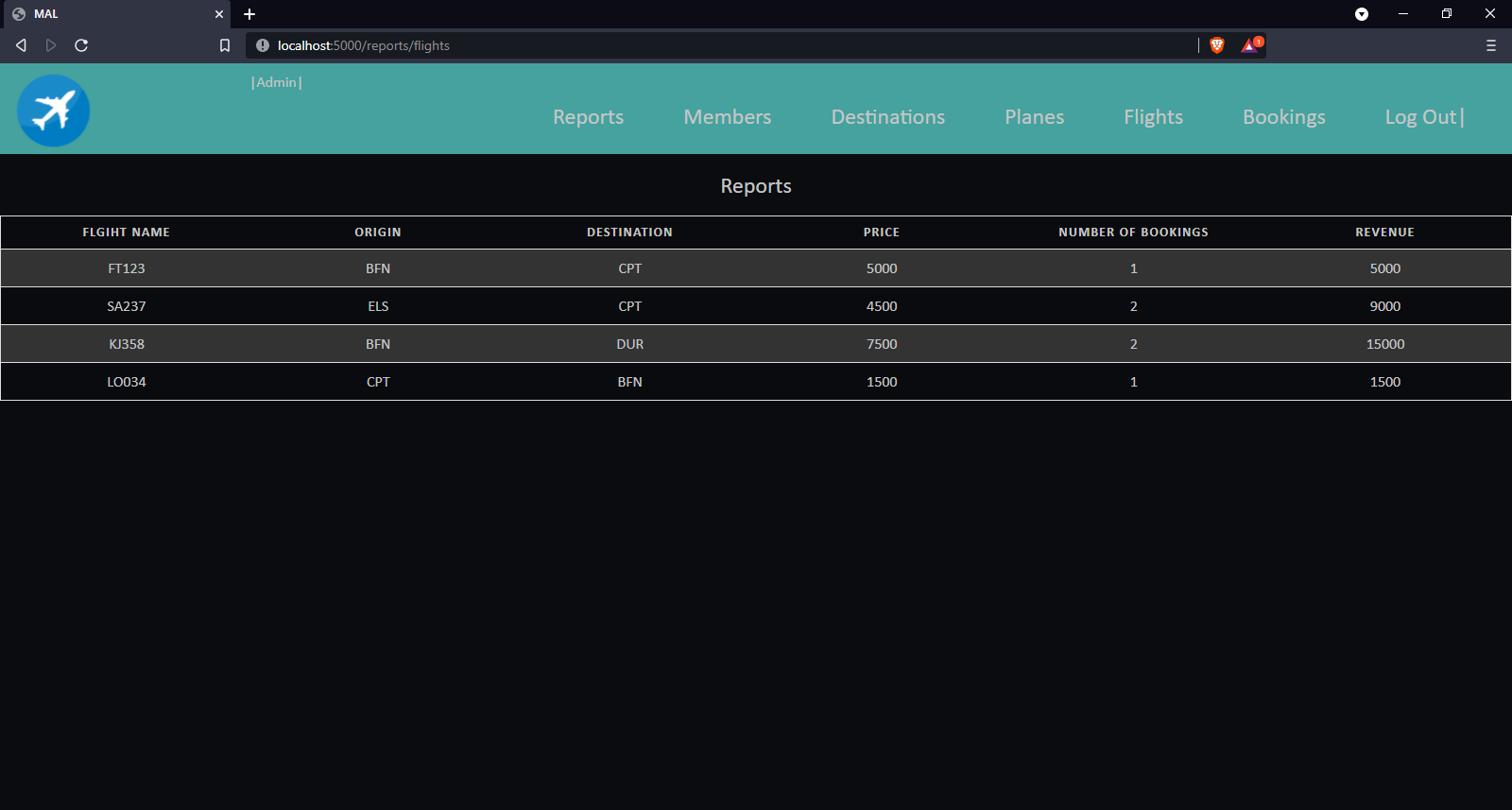
        this.frontEndMessage = pFrontEndMessage;

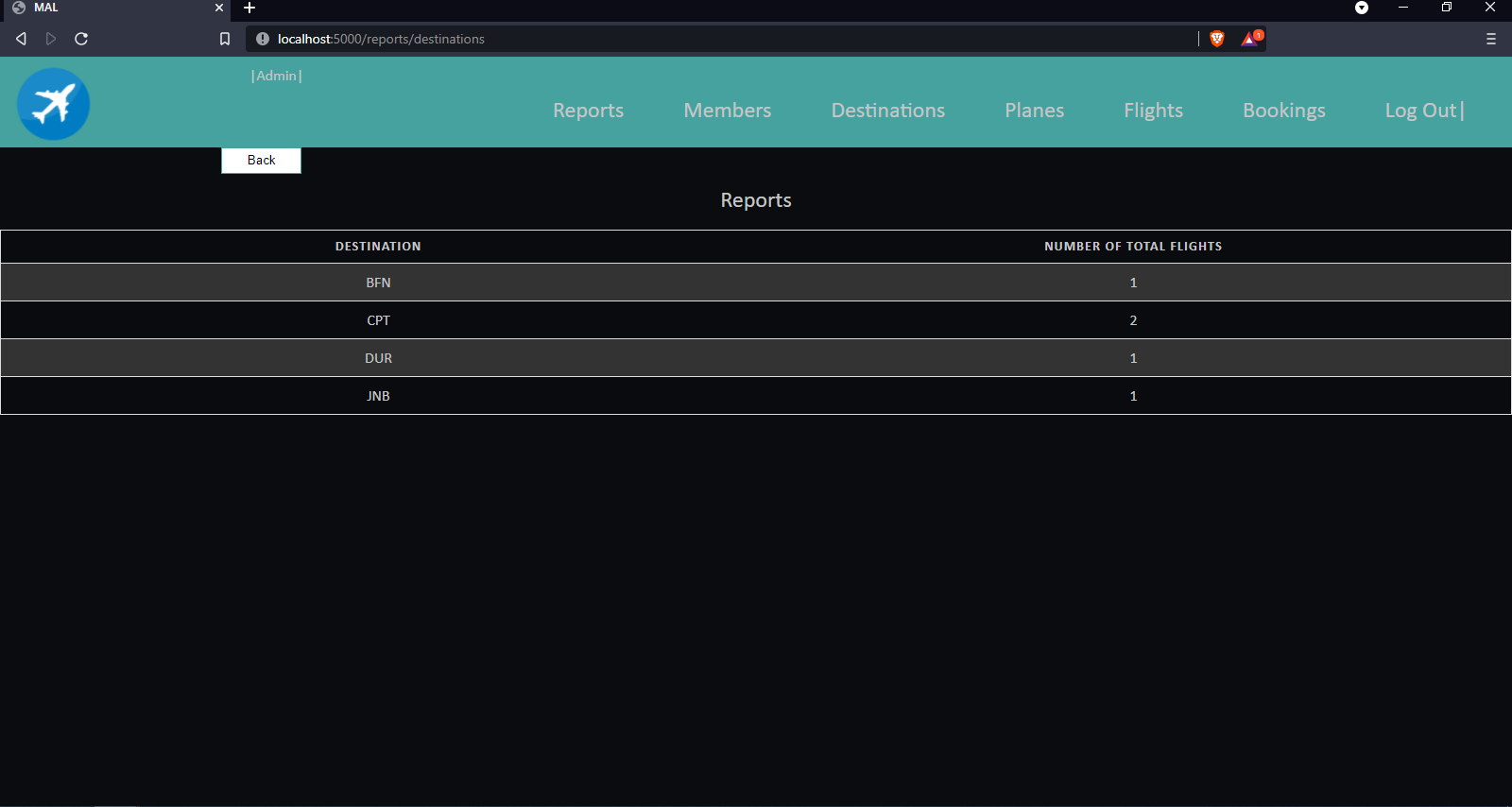
    }

}



# Reports Screenshots:

Number of bookings per flight and total revenue

Total flights to destinations

# User Manual:

## Installation

For production the software is web based therefore it will be preinstalled on the server by the server administrator and one of the senior developers. After the installation the system can be accessed through a browser using the domain name to connect to the site.

For development the user will need to install Node.js downloaded from (url) and run the usbwebserver.exe (supplied along source code) to run the sandbox database.

Now the user will need to use the command-line or terminal, navigate to the project file and in the active directory type ‘npm install’ and wait for the dependencies to finish installing. There after the user will need to run the command node index.js in the same directory. The user can now open their web browser and type localhost:5000 in the URL and the system should load.

## System Requirements for server:

**Processor:** Any Intel/AMD processors.

**Memory:** 2GB

**OS:** Windows server or any server based Linux distro (Debian/CentOS/Ubuntu)

**Graphics:** Intel Based On-Board Graphics

**Storage:** 100 GB SSD

## System Requirements for user:

**Processor:** Any Intel/AMD processors.

**Memory:** 8GB

**OS:** Windows 10+ or macOS or Linux

**Graphics:** Intel Based On-Board Graphics

**Storage:** No need, Software is Web-Based

**Supported Browsers:** Mozilla Firefox/Google Chrome/Internet Explorer/Opera/Edge

# Diary: