CSE 3055 DATABASE SYSTEMS PROJECT

AIRPORT

Ertuğrul Sağdıç – 150116061

Eray Ayaz – 150116053

About The Project

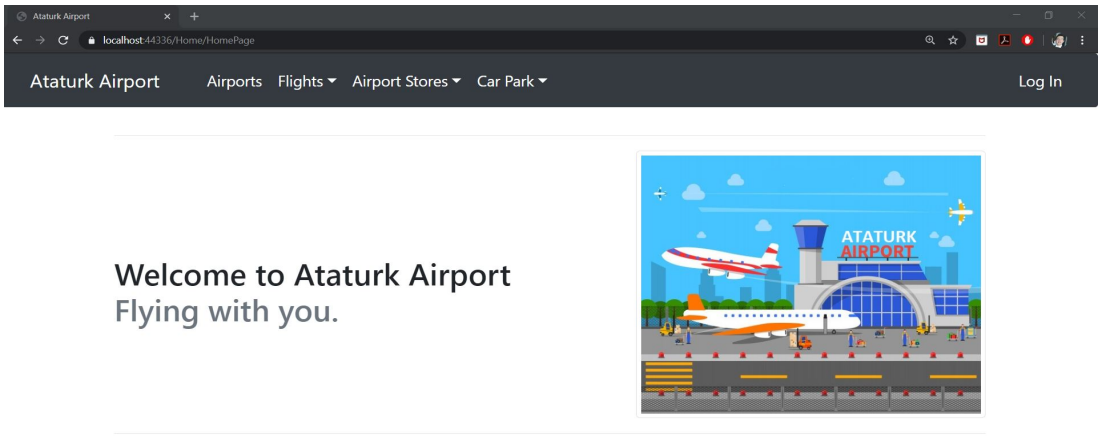
In this project step, we are designing a web application, which uses our airport database system. Our aim is to make easier management for our airport. There are several functionalities in our web application:

- List of airports.
- List of flights, search for specific flight with flight ID
- List of stores in airport, and search for a specific store with its name.
- List of cars in the car park, searching for a specific space number with using plate number of car, and calculation of the total price of the car park with plate number.
- There is a login page. You must use manager id and password, in order to log in the system. After log in, there are several more functionalities:
    - List of employees, search for an employee's salary.
    - Search for a certificate.
    - Search for a specific pilot with flight id.
    - List of passengers,search for a ticket with ticket ID.
    - Add cargo / passenger flights.

In this web application, these functions which is we created will be easy for users and managers to find what they are looking for.We use 2 layout for 2 types of user. One of them can be any human the other one is should be manager of the one department. Therefore manager layout helps him look at more things.

Web Application and Database

When we run the code, this is the first screen that welcomes us on the website. On the top of the website, there is a navigation bar. Persons can use the navigation bar according to the jobs they want. On the right side of the navigation bar is the login button for managers.

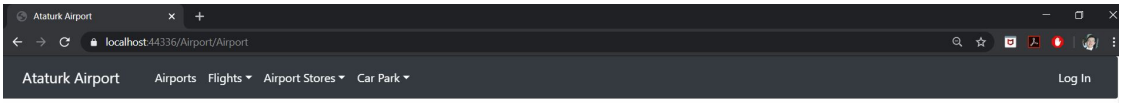


---

1)Airports Button

The airports button shows users which airports they can go from our airports. Also this part of code do the this job.

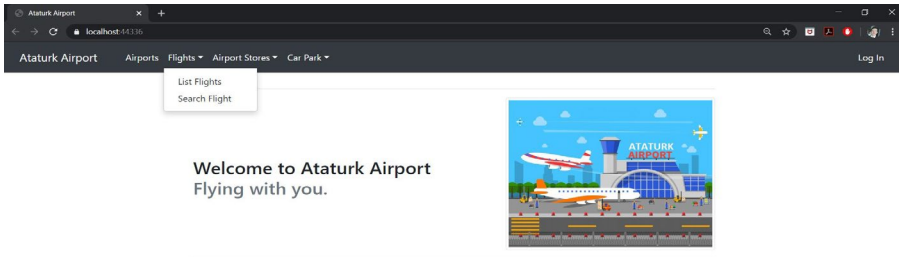
```
string cmdString = "SELECT * FROM Airport";
```



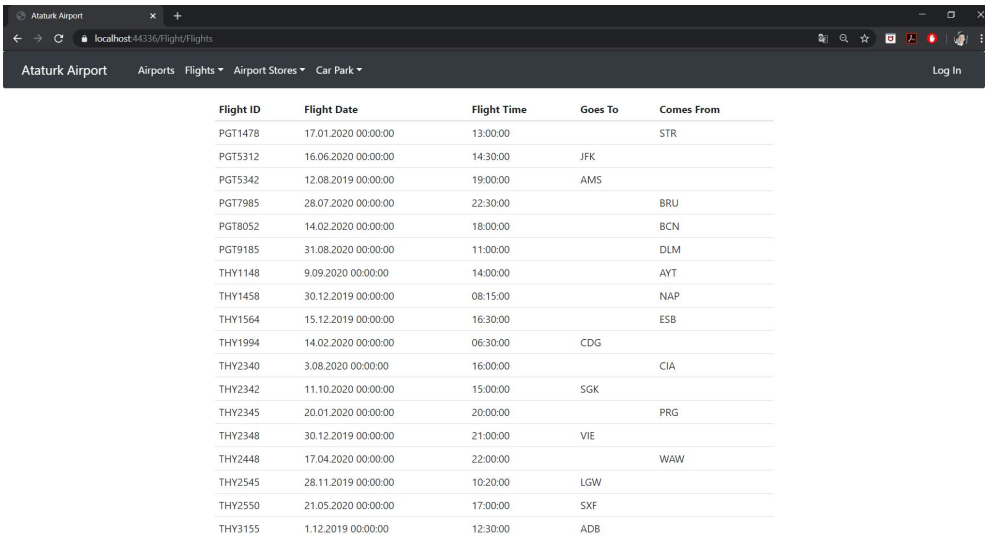
Airport ID	Name	City
ADB	Adnan Menderes Airport	Izmir
AMS	Amsterdam Schiphol Airport	Amsterdam
AYT	Antalya Airport	Antalya
BCN	Barcelona Airport	Barcelona
BRU	Brussel Airport	Brussel
BUD	Budapest Ferenc Liszt Airport	Budapest
CDG	Paris Charles Gualle Airport	Paris
CIA	Ciampino Airport	Rome
DLM	Dalaman Airport	Mugla
ESB	Esenboga Airport	Ankara
IST	Istanbul Airport	Istanbul
JFK	John F. Keneddy Airport	New York
LGW	London Gatwick Airport	London
NAP	Naples Airport	Napels
PRG	Vaclav Havel Airport	Prague
SGK	Selanik Airport	Selanik
STR	Stuttgard Airport	Stuttgard

2)Flight Button

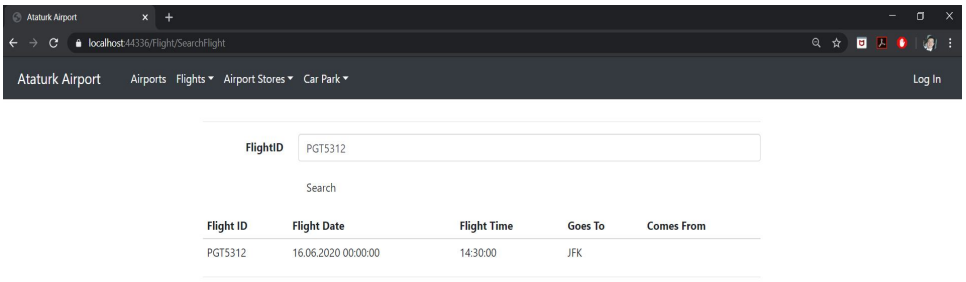
Flights button provide a two option to user. First one is list flights button and it will list the flights and user can search the flights with the second one.




```
string cmdString = "SELECT * FROM Airport_Store";
```



Flight ID	Flight Date	Flight Time	Goes To	Comes From
PGT1478	17.01.2020 00:00:00	13:00:00		STR
PGT5312	16.06.2020 00:00:00	14:30:00	JFK	
PGT5342	12.08.2019 00:00:00	19:00:00	AMS	
PGT7985	28.07.2020 00:00:00	22:30:00		BRU
PGT8052	14.02.2020 00:00:00	18:00:00		BCN
PGT9185	31.08.2020 00:00:00	11:00:00		DLM
THY1148	9.09.2020 00:00:00	14:00:00		AYT
THY1458	30.12.2019 00:00:00	08:15:00		NAP
THY1564	15.12.2019 00:00:00	16:30:00		ESB
THY1994	14.02.2020 00:00:00	06:30:00	CDG	
THY2340	3.08.2020 00:00:00	16:00:00		CIA
THY2342	11.10.2020 00:00:00	15:00:00	SGK	
THY2345	20.01.2020 00:00:00	20:00:00		PRG
THY2348	30.12.2019 00:00:00	21:00:00	VIE	
THY2448	17.04.2020 00:00:00	22:00:00		WAW
THY2545	28.11.2019 00:00:00	10:20:00	LGW	
THY2550	21.05.2020 00:00:00	17:00:00	SXF	
THY3155	1.12.2019 00:00:00	12:30:00	ADB	



Flight ID	Flight Date	Flight Time	Goes To	Comes From
PGT5312	16.06.2020 00:00:00	14:30:00	JFK	

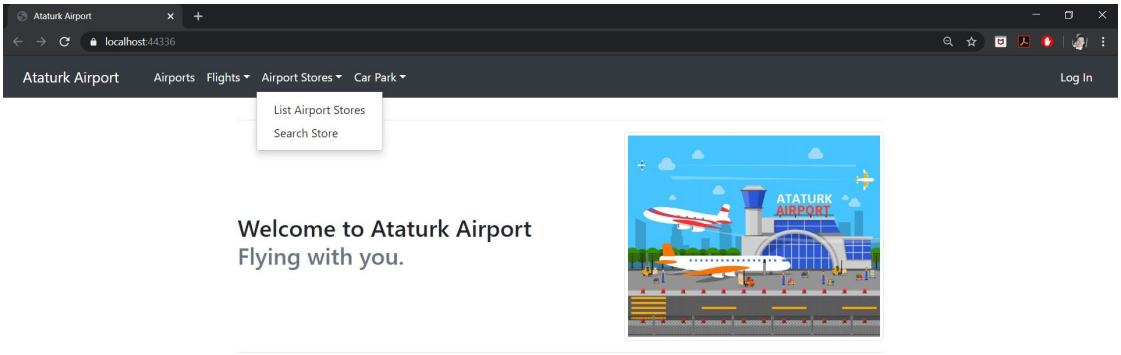
We created a stored procedure which name is a SearchFlight and it provide to find the flights with the flight id and this part of the code.

```
string cmdString1 = "SearchFlight";
SqlCommand cmd1 = new SqlCommand(cmdString1, connection);
cmd1.CommandType = System.Data.CommandType.StoredProcedure;
```

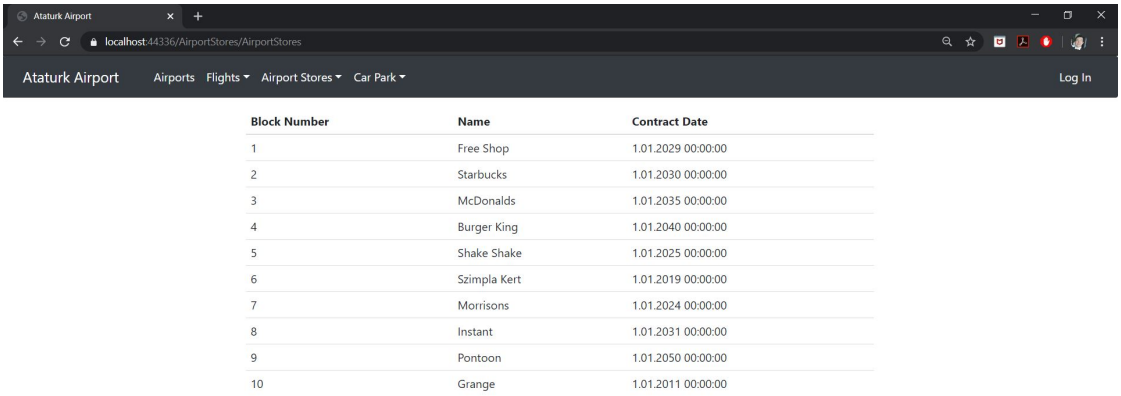
```
create procedure SearchFlight
    @flightID varchar(7)
as
    select *
    from Flight f where f.FlightID = @flightID
```

3)Airport Store Button

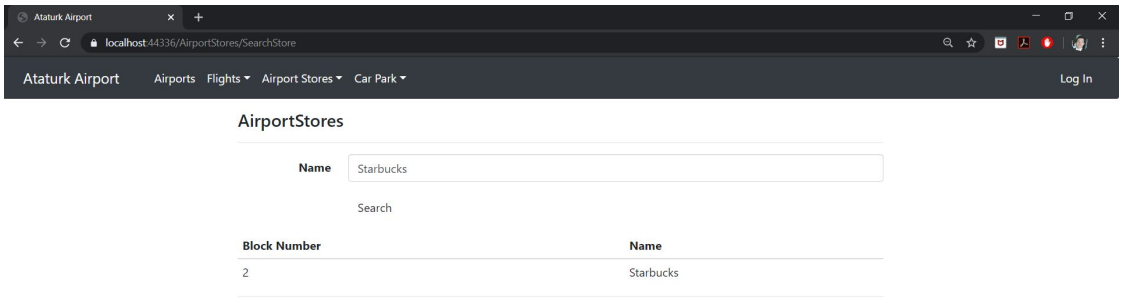
Airport Stores button functions same as flights button. It can list the stores and users can search the check where the stores are.




```
string cmdString = "SELECT * FROM Airport_Store";
```



Block Number	Name	Contract Date
1	Free Shop	1.01.2029 00:00:00
2	Starbucks	1.01.2030 00:00:00
3	McDonalds	1.01.2035 00:00:00
4	Burger King	1.01.2040 00:00:00
5	Shake Shake	1.01.2025 00:00:00
6	Szimpla Kert	1.01.2019 00:00:00
7	Morrisons	1.01.2024 00:00:00
8	Instant	1.01.2031 00:00:00
9	Pontoon	1.01.2050 00:00:00
10	Grange	1.01.2011 00:00:00



Also we created a stored procedure again and it's name is Search

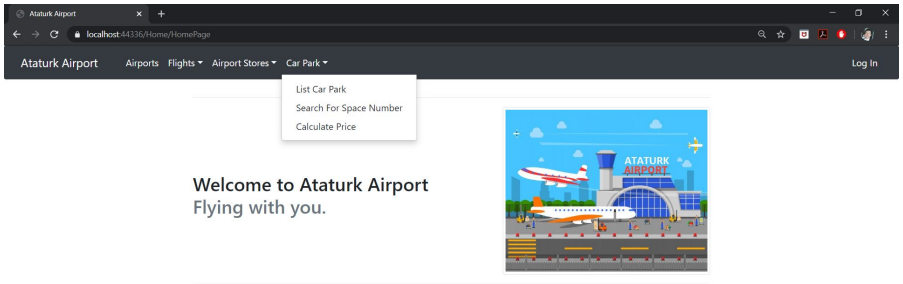
```
string cmdString1 = "SearchStore";
SqlCommand cmd1 = new SqlCommand(cmdString1, connection);
cmd1.CommandType = System.Data.CommandType.StoredProcedure;
```

```
create procedure SearchStore
    @name varchar(15)
as
begin
    select *
    from Airport_Store a where a.name = @name
end
```

Store. it provide to find the stores with the store name and this part of the code.

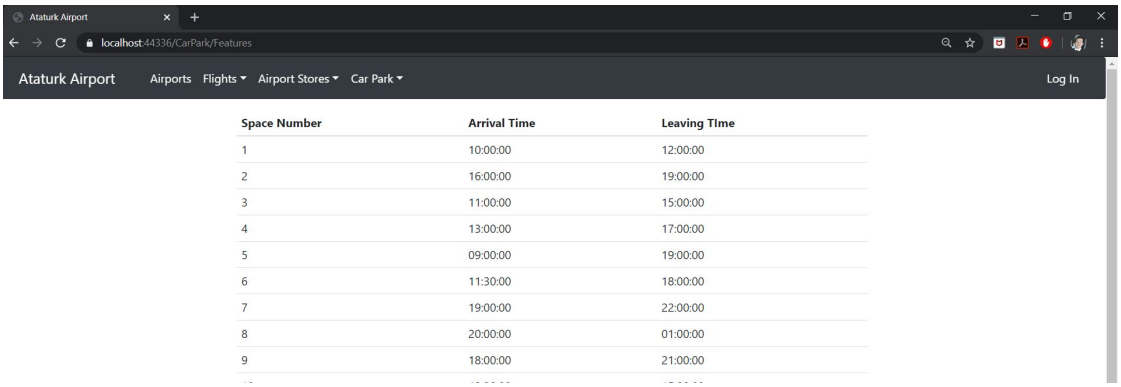
4)Car Park Button

Users can list the car with the car park button and they can see the how much parking lot is full. Also can search the their car with enter the car's plate number and owner's of the car can calculate the parking fee.



This sql query list the car which is in the car park.

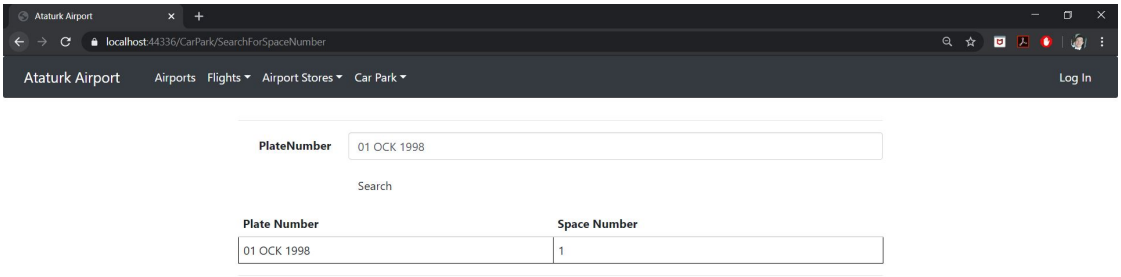
```
string cmdString = "SELECT * FROM Car_Park";
```



User can see the their car's space number with their plate number. This part of code and query provide that.

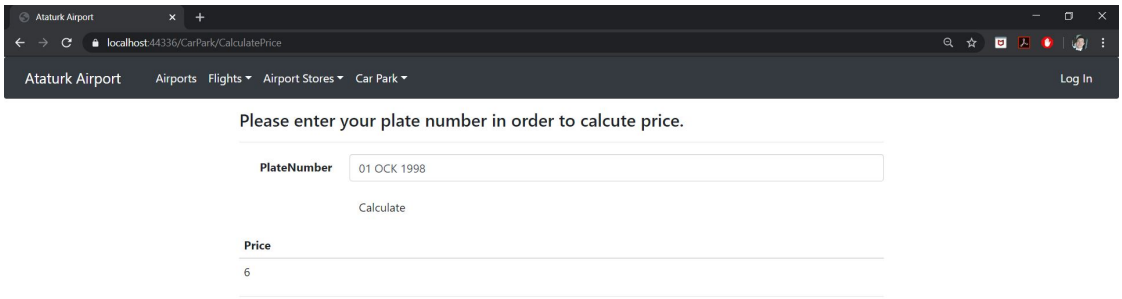
```
string cmdString1 = "CarDetail";
SqlCommand cmd1 = new SqlCommand(cmdString1, connection);
cmd1.CommandType = System.Data.CommandType.StoredProcedure;
```

```
create procedure CarDetail
    @platenumber varchar(11)
as
    select *
    from Vehicle v INNER JOIN Car c on v.VehicleID = c.CarVehicleID INNER JOIN Car_Park p on p.SpaceNumber=c.CarSpaceNumber
    where c.PlateNumber= @platenumber
```



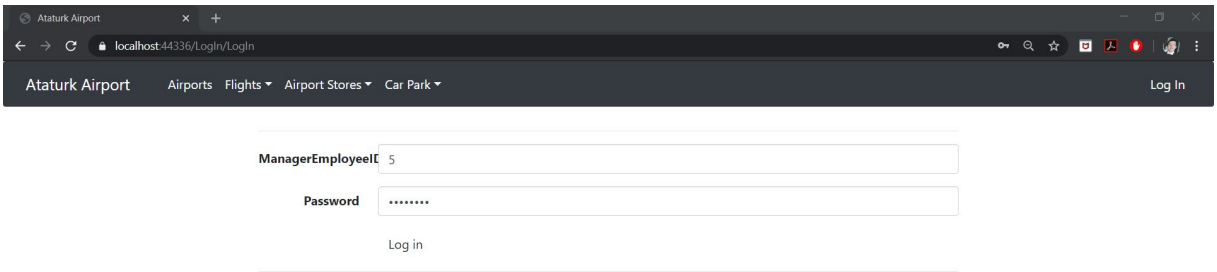
Also when owner's of the car enter the plate number of his/her car application will calculate the parking fee. This stored procedure which name is the Calculate Price do that.

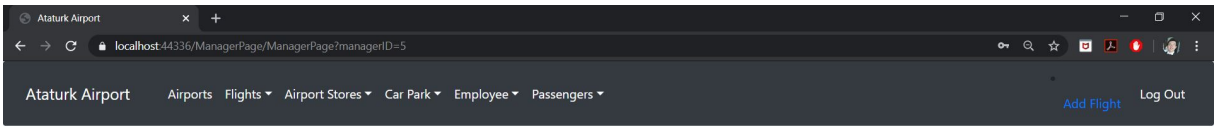
```
CREATE PROCEDURE CalculatePrice
    @PlateNumber varchar(11)
AS
BEGIN
DECLARE
    @Diff int,
    @dailyPrice int,
    @TotalPrice int,
    @arrivalTime time,
    @leavingTime time,
    @price float
set @arrivalTime = (SELECT ArrivalTime
FROM Car_Park CP INNER JOIN Car C ON CP.SpaceNumber = C.CarSpaceNumber
WHERE C.PlateNumber = @PlateNumber)
set @leavingTime = (SELECT LeavingTime
FROM  Car_Park CP INNER JOIN Car C ON CP.SpaceNumber = C.CarSpaceNumber
WHERE C.PlateNumber = @PlateNumber)
set @diff = DATEDIFF(HOUR, @arrivalTime, @leavingTime)
set @dailyPrice = (SELECT DailyPrice
FROM Car_Park CP INNER JOIN Car C ON CP.SpaceNumber = C.CarSpaceNumber
WHERE C.PlateNumber = @PlateNumber)
set @price = (@dailyPrice / 24) * @Diff
select @price
update Car
set HourlyPrice = @price
where Car.CarSpaceNumber = (SELECT SpaceNumber
FROM Car_Park CP INNER JOIN Car C ON CP.SpaceNumber = C.CarSpaceNumber
WHERE C.PlateNumber = @PlateNumber);
select Car.HourlyPrice from Car
end
```



5)Log in Button

Only managers can use this button. Because on a website, it should contaion information that is not accessible to everyone and should not be seen by others about the airport. When a manager want to enter the website with login button he/she should enter his/her manager id and password to enter the page. When somene log in the website print the welcome message to manager.





Log In Successful
Welcome Büşra

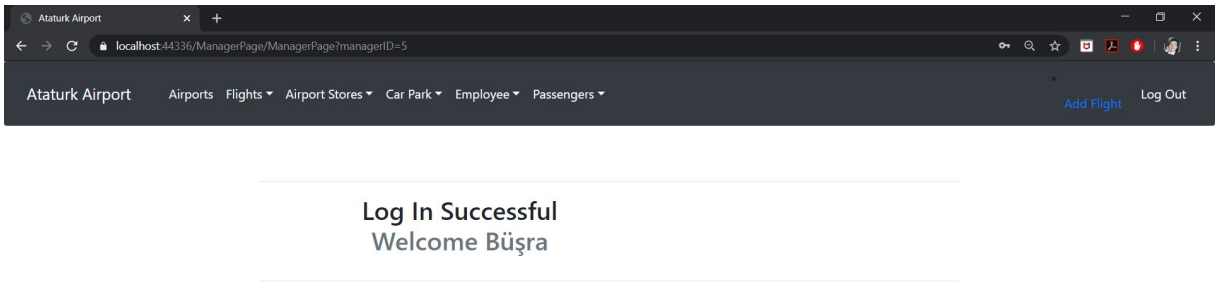
```sql
create procedure PersonInfo
    @managerID int
as
SELECT Name
FROM Manager INNER JOIN
    Employee ON Manager.ManagerEmployeeID = Employee.EmployeeID INNER JOIN
    Person ON Employee.EmployeeTCKN = Person.TCKN
    where Manager.ManagerEmployeeID = @managerID
    and Manager.ManagerEmployeeID = Employee.EmployeeID
    and Employee.EmployeeTCKN = Person.TCKN
```

```
string cmdString1 = "PersonInfo";
SqlCommand cmd1 = new SqlCommand(cmdString1, connection);
cmd1.CommandType = System.Data.CommandType.StoredProcedure;
```

Also when the manager enter the website. Employee button added to navigation bar.

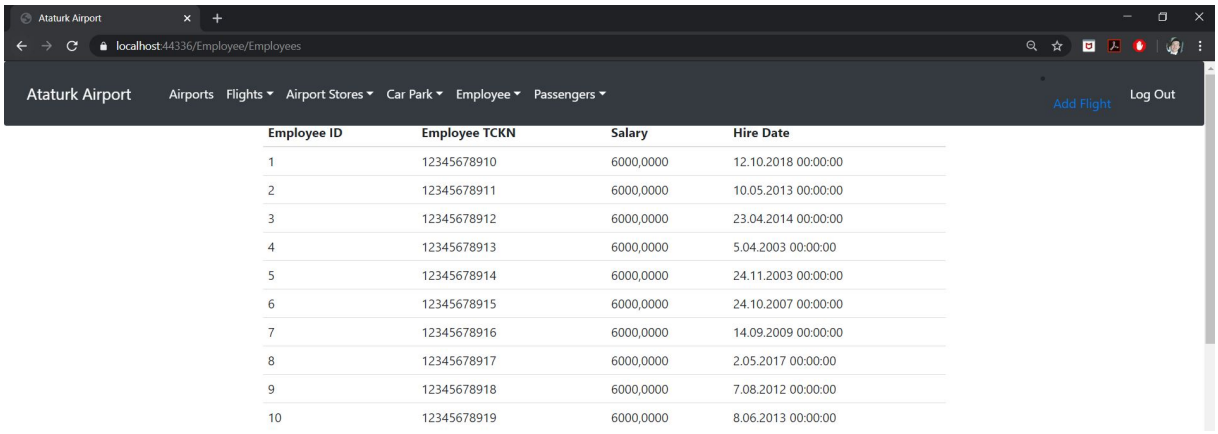
6)Employee Button

Employee button has 3 option.List the employees and search the employee with their employee id. When the manager want to search the employee he/she can see some information about employee.Last one is provide the see employee certificate.



We list the employees with this sql query.

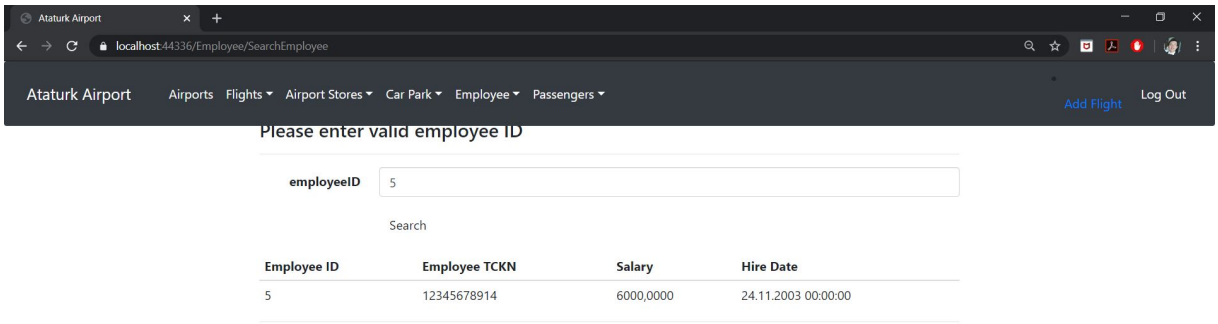
```
string cmdString = "SELECT * FROM Employee";
```



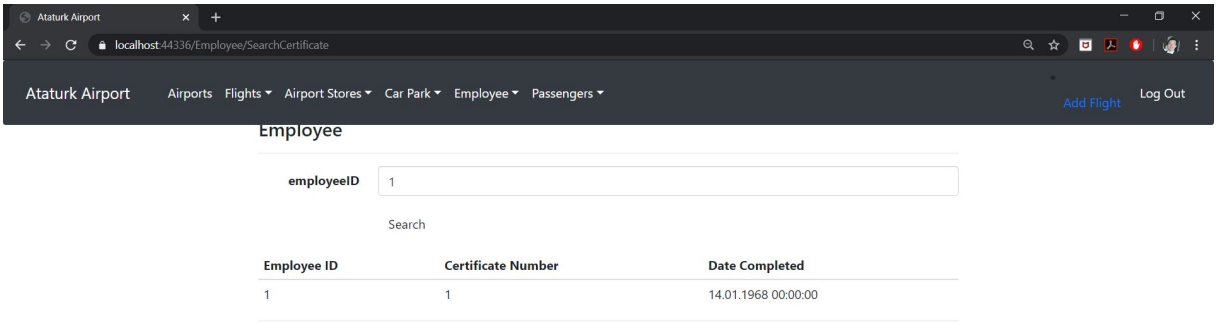
We create a stored procedure to search a employee with manager id and when you enter the manager id correctly. İt will show the some information about the employee.

```
string cmdString1 = "SearchSalary";
SqlCommand cmd1 = new SqlCommand(cmdString1, connection);
cmd1.CommandType = System.Data.CommandType.StoredProcedure;
```

```
create procedure SearchSalary
    @employeeID int
as
begin
    select *
    from Employee e where e.EmployeeID = @employeeID
end
```



Manager can see the employee's certificate with enter the employee id and we are using for this functions this sql query

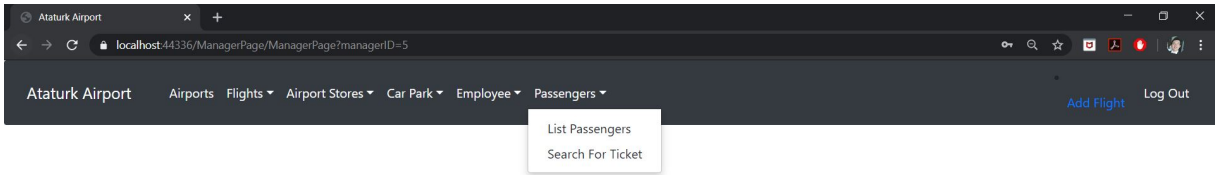



```
string cmdString = "SELECT * FROM Employee_Certificate WHERE Employee_Certificate.CEmployeeID =" + employee.employeeID;
```

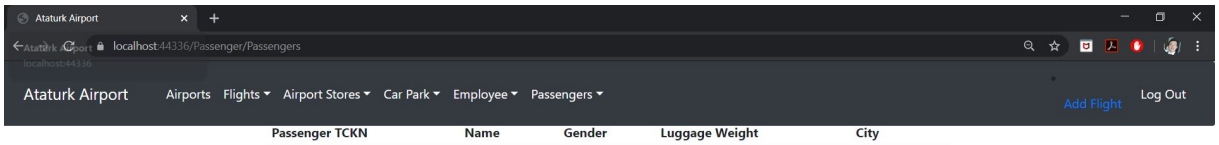
7)Passenger Button

Managers can see the list and detail about the passenger when the they enter the passenger tckn.

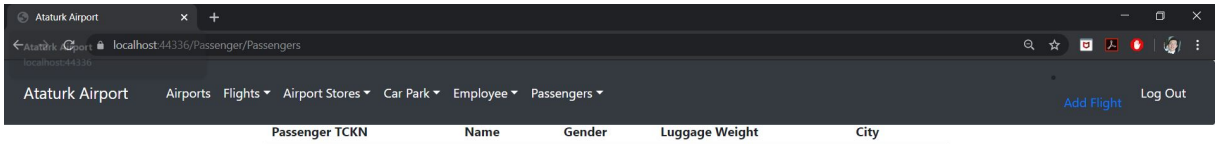
```
string cmdString = "SELECT * FROM Passenger INNER JOIN Person ON Passenger.PassengerTCKN = Person.TCKN";
```



Log In Successful
Welcome Büşra



Passenger TCKN	Name	Gender	Luggage Weight	City
12345678900	Eray	M	25	İstanbul
12345678901	Ertuğrul	M	25	İstanbul
12345678902	Evrim	M	28	İstanbul
12345678915	Çağla	F	8	İstanbul
12345678916	Ezgi	F	10	İstanbul
12345678917	Sevcan	F	15	İstanbul
12345678922	Hicran	F	27	İstanbul
12345678923	Didem	F	16	İstanbul
12345678934	Gülşah	F	5	İstanbul
12345678945	İzzy	F	32	İstanbul

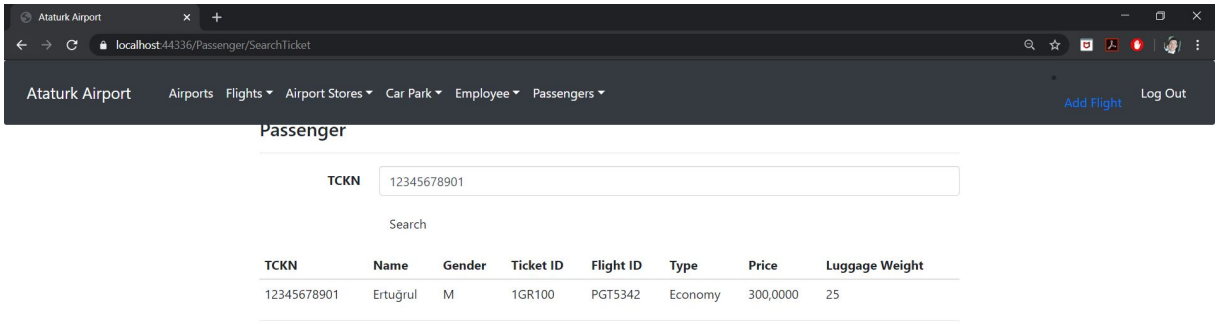


Passenger TCKN	Name	Gender	Luggage Weight	City
12345678900	Eray	M	25	İstanbul
12345678901	Ertuğrul	M	25	İstanbul
12345678902	Evrim	M	28	İstanbul
12345678915	Çağla	F	8	İstanbul
12345678916	Ezgi	F	10	İstanbul
12345678917	Sevcan	F	15	İstanbul
12345678922	Hicran	F	27	İstanbul
12345678923	Didem	F	16	İstanbul
12345678934	Gülşah	F	5	İstanbul
12345678945	İzzy	F	32	İstanbul

We used the stored procedure for the see detail of the passenger.

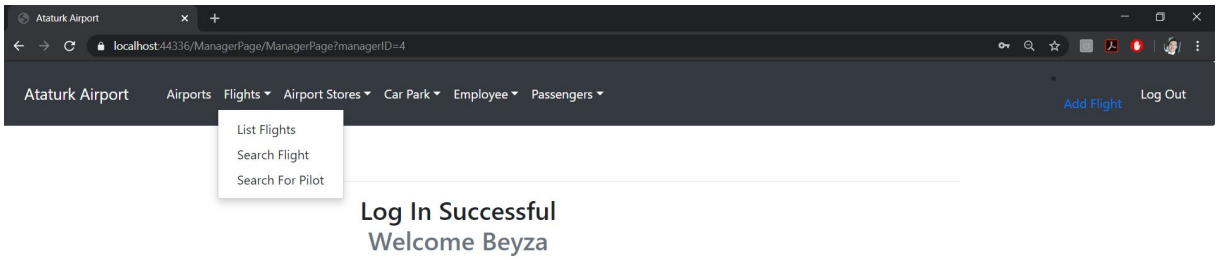
```
create procedure TicketInfo
    @TCKN numeric(11)
    as
SELECT *
FROM Passenger INNER JOIN
Flies ON Passenger.PassengerTCKN = Flies.PassengerTCKN INNER JOIN
Passenger_Flight ON Flies.PFlightID = Passenger_Flight.PFlightID INNER JOIN
Person ON Passenger.PassengerTCKN = Person.TCKN INNER JOIN
Ticket ON Passenger.PassengerTCKN = Ticket.TicketPassengerTCKN
where Passenger.PassengerTCKN = @TCKN
```

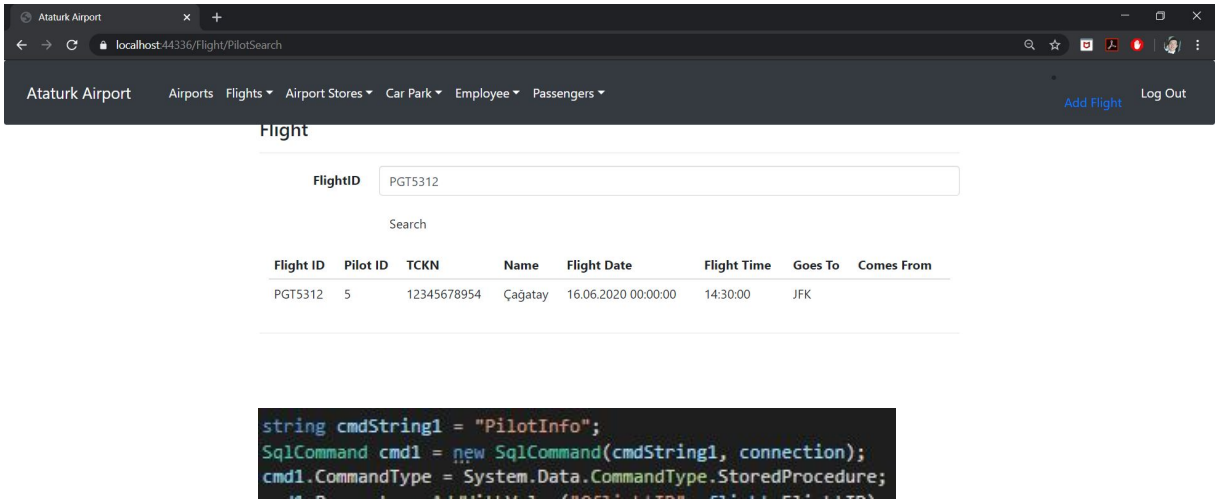
```
string cmdString1 = "TicketInfo";
SqlCommand cmd1 = new SqlCommand(cmdString1, connection);
cmd1.CommandType = System.Data.CommandType.StoredProcedure;
```



8)Seach For Pilot Button

Manager also can get the information about the pilot. When the manager enter the site





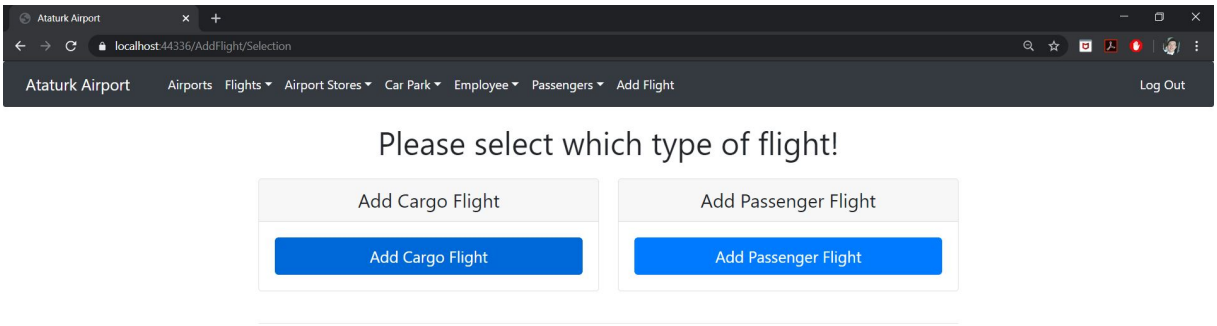

```
string cmdString1 = "PilotInfo";
SqlCommand cmd1 = new SqlCommand(cmdString1, connection);
cmd1.CommandType = System.Data.CommandType.StoredProcedure;
```

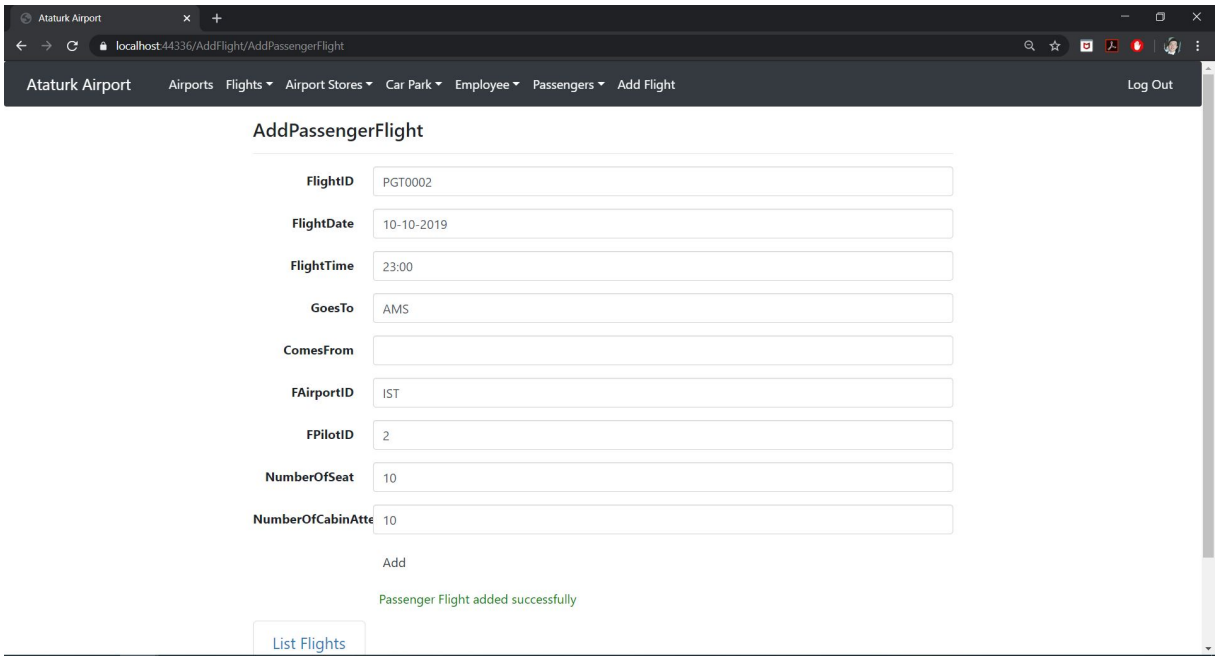
```
create procedure PilotInfo
    @flightID varchar(7)
as
    SELECT *
    FROM Pilot INNER JOIN Flight ON Pilot.PilotID = Flight.FPilotID INNER JOIN
    Person ON Pilot.PTCKN = Person.TCKN
    where Flight.FlightID = @flightID
```

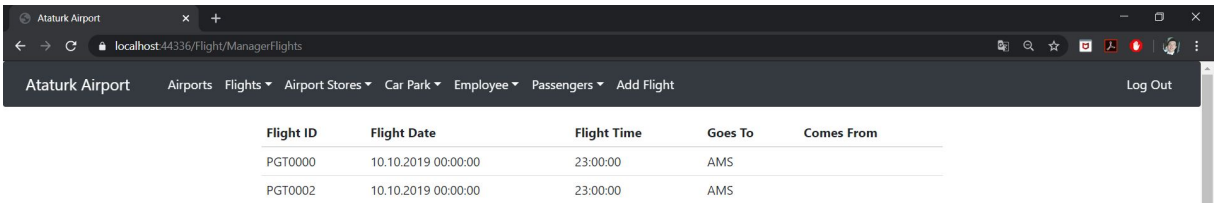
Manager can see the detail about the pilot with this query and part of the code.

9)Add Flight Button

Manager can add a passenger or cargo flights. When manager enter the manager screen. Add flight button added to navigation bar.
Manager need the choose the flight type then should enter the some information about the flights. After this jobs manager can see the list of all flights.








```
string cmdString1 = "INSERT INTO Cargo_Flight VALUES('" + flight.FlightID + "', " + flight.numberOfCargo + ")";
```

```
string cmdString1 = "INSERT INTO Passenger_Flight VALUES('" + flight.FlightID + "', " + flight.NumberOfSeat + ", " + flight.NumberOfCabinAttendant + ")";
```

This parts of codes provide the add a flight.

Used Technologies

- MVC
- MSSQL