**Group 8**

**CMPG223 Documentation for Arcadia Hotel**

**Members:**

Ian Stolz 37251244

Albertus van der Walt 37073710

Bernard Swanepoel 39909476

Ashton du Plessis 34202676

Jacobus Koorts 35560517

Franco Burger 24904635

Table of Contents

[Physical Data Model 3](#_Toc113811644)

[Physical Process Model 3](#_Toc113811645)

[Database Schema Created 6](#_Toc113811646)

[SQL Used 7](#_Toc113811647)

[Example code for maintaining a child entity and reuse of code 10](#_Toc113811648)

[Output of two reports 12](#_Toc113811649)

[User Manual 13](#_Toc113811650)

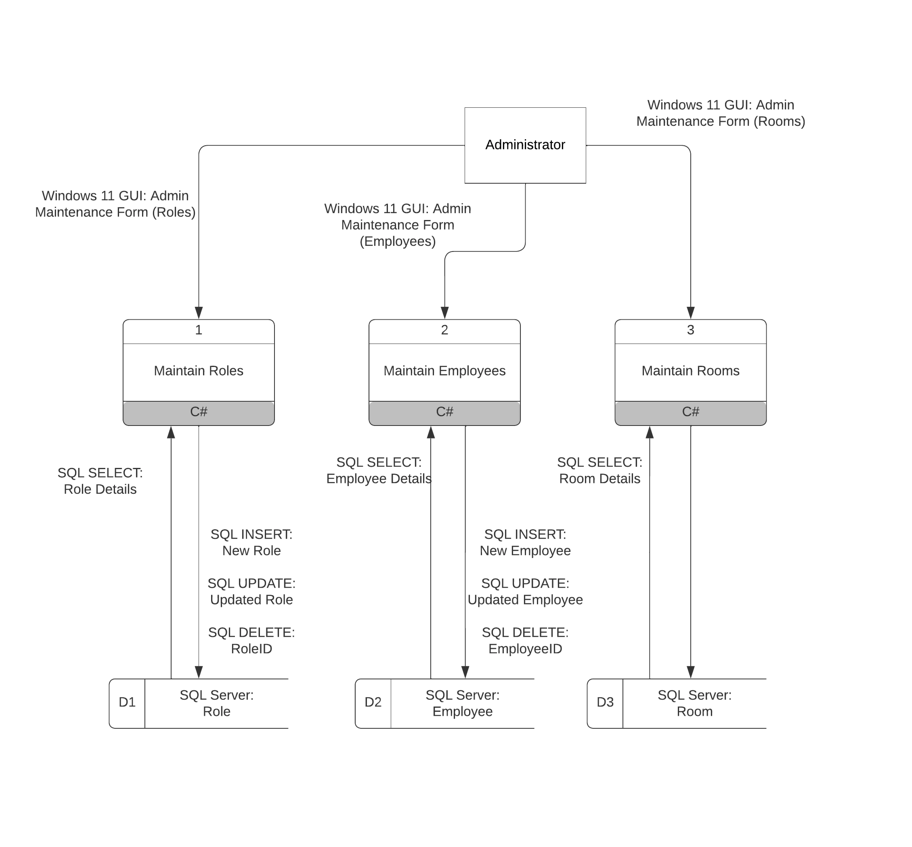
[Diary 14](#_Toc113811651)

# Physical Data Model

# Physical Process Model

Diagram, engineering drawing, schematic

Description automatically generated

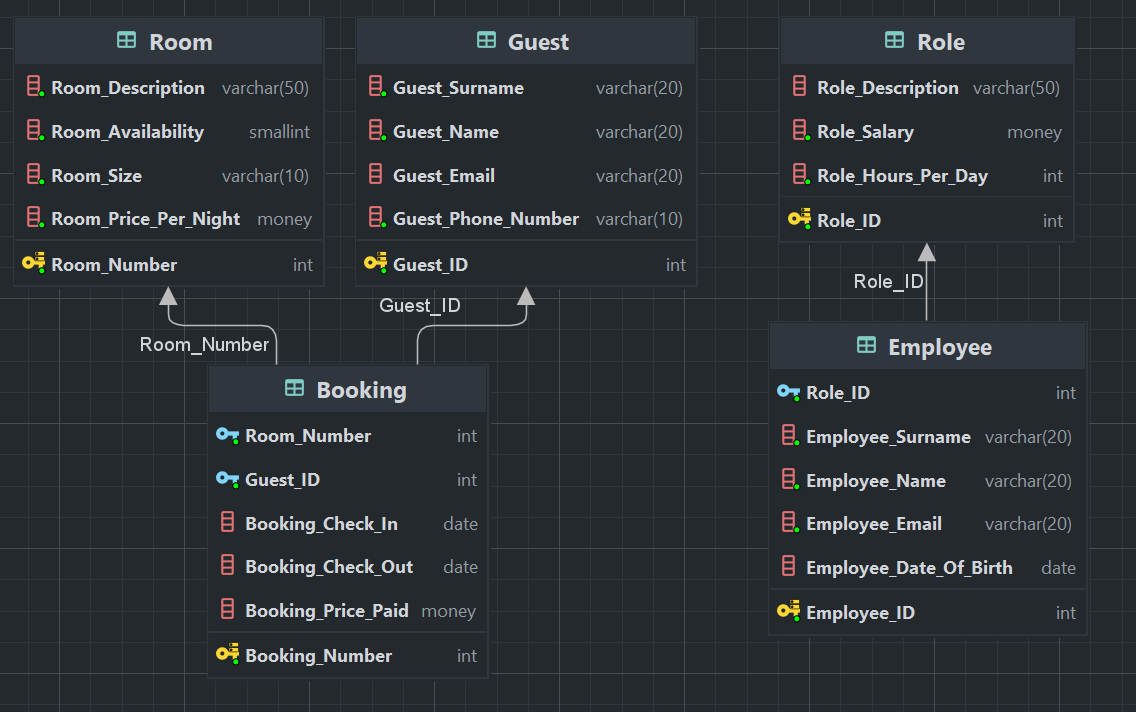


Diagram

Description automatically generated

# Database Schema Created

DataGrip:



SQL Server

Diagram

Description automatically generated

# SQL Used

## SQL Used to create database:

**Booking:**

CREATE TABLE [dbo].[Booking] (

[Booking\_Number] INT IDENTITY (1, 1) NOT NULL,

[Room\_Number] INT NOT NULL,

[Guest\_ID] INT NOT NULL,

[Booking\_Check\_Ind] DATE NULL,

[Booking\_Check\_Out] DATE NULL,

[Booking\_Price\_Paid] MONEY NULL,

CONSTRAINT [Booking\_pk] PRIMARY KEY CLUSTERED ([Booking\_Number] ASC),

CONSTRAINT [Booking\_Room\_Room\_Number\_fk] FOREIGN KEY ([Room\_Number]) REFERENCES [dbo].[Room] ([Room\_Number]),

CONSTRAINT [Booking\_Guest\_null\_fk] FOREIGN KEY ([Guest\_ID]) REFERENCES [dbo].[Guest] ([Guest\_ID])

);

**Employee:**

CREATE TABLE [dbo].[Employee] (

[Employee\_ID] INT IDENTITY (1, 1) NOT NULL,

[Role\_ID] INT NOT NULL,

[Employee\_Surname] VARCHAR (20) NOT NULL,

[Employee\_Name] VARCHAR (20) NOT NULL,

[Employee\_Email] VARCHAR (20) NOT NULL,

[Employee\_Date\_Of\_Birth] DATE NULL,

CONSTRAINT [Employee\_ID] PRIMARY KEY CLUSTERED ([Employee\_ID] ASC),

CONSTRAINT [Employee\_Role\_null\_fk] FOREIGN KEY ([Role\_ID]) REFERENCES [dbo].[Role] ([Role\_ID])

);

**Guest:**

CREATE TABLE [dbo].[Guest] (

[Guest\_ID] INT IDENTITY (1, 1) NOT NULL,

[Guest\_Surname] VARCHAR (20) NOT NULL,

[Guest\_Name] VARCHAR (20) NOT NULL,

[Guest\_Email] VARCHAR (20) NULL,

[Guest\_Phone\_Number] VARCHAR (10) NOT NULL,

CONSTRAINT [Guest\_ID] PRIMARY KEY CLUSTERED ([Guest\_ID] ASC)

);

**Role:**

CREATE TABLE [dbo].[Role] (

[Role\_ID] INT IDENTITY (1, 1) NOT NULL,

[Role\_Description] VARCHAR (50) NULL,

[Role\_Salary] MONEY NOT NULL,

[Role\_Hours\_Per\_Day] INT DEFAULT (8) NOT NULL,

CONSTRAINT [Role\_pk] PRIMARY KEY CLUSTERED ([Role\_ID] ASC)

);

**Room:**

CREATE TABLE [dbo].[Room] (

[Room\_Number] INT IDENTITY (1, 1) NOT NULL,

[Room\_Description] VARCHAR (50) NOT NULL,

[Room\_Availability] SMALLINT DEFAULT (0) NOT NULL,

[Room\_Size] VARCHAR (10) NOT NULL,

[Room\_Price\_Per\_Night] MONEY NOT NULL,

CONSTRAINT [Room\_Number] PRIMARY KEY CLUSTERED ([Room\_Number] ASC)

);

**SQL used:**

**Select:**

SELECT \* from Booking

SELECT \* from Employee

SELECT \* from Guest

SELECT \* from Role

SELECT \* from Room

SELECT E.Employee\_Surname, E.Employee\_Name, R.Role\_Description, E.Employee\_Email, R.Role\_Salary FROM Employee E, Role R WHERE E.Role\_ID = R.Role\_ID

$" SELECT E.Employee\_Surname, E.Employee\_Name, R.Role\_Description, E.Employee\_Email, R.Role\_Salary FROM Employee E, Role R WHERE E.Role\_ID = R.Role\_ID AND R.Role\_Description = {roleType}"

SELECT R.Room\_Description, R.Room\_Size, R.Room\_Price\_Per\_Night, COUNT(B.Room\_ID) AS "Guests", SUM(R.Room\_Price\_Per\_Night) AS "Total money received" FROM Room R, Booking B WHERE R.Room\_Number = B.Room\_Number

$"SELECT R.Room\_Description, R.Room\_Size, R.Room\_Price\_Per\_Night, COUNT(B.Room\_ID) AS "Guests", SUM(R.Room\_Price\_Per\_Night) AS "Total money received" FROM Room R, Booking B WHERE R.Room\_Number = B.Room\_Number AND R.Room\_Size = {roomSize}"

**Update:**

UPDATE Booking SET Room\_Number = @Room\_Number, Guest\_ID = @Guest\_ID, Booking\_Check\_In = @Booking\_Check\_In, Booking\_Check\_Out = @Booking\_Check\_Out, Booking\_Price\_Paid = @Booking\_Price\_Paid WHERE Booking\_Number = @Booking\_Number

UPDATE Role SET Role\_Description = @Role\_Description, Role\_Salary = @Role\_Salary, Role\_Hours\_Per\_Day = @Role\_Hours\_Per\_Day WHERE Role\_ID = @Role\_ID

UPDATE Employee SET Employee\_Name = @Employee\_Surname, Employee\_Name = @Employee\_Name, Employee\_Email = @Employee\_Email, Employee\_Date\_Of\_Birth = @Employee\_Date\_Of\_Birth WHERE Employee\_ID = @Employee\_ID

UPDATE Room SET Room\_Description = @Room\_Description, Room\_Availability = @Room\_Availibility, Room\_Size = @Room\_Size, Room\_Price\_Per\_Night = @Room\_Price\_Per\_Night WHERE Room\_Number = @Room\_Number

**Insert:**

INSERT into BOOKING(Room\_Number,Guest\_ID,Booking\_Check\_In,Booking\_Check\_Out,Booking\_Price\_Paid) VALUES (@Room\_Number,@Guest\_ID,@Booking\_Check\_In,@Booking\_Check\_Out,@Booking\_Price\_Paid)

INSERT into EMPLOYEE(Role\_ID,Employee\_Surname,Employee\_Name,Employee\_Date\_Of\_Birth,Employee\_Email) VALUES (@Role\_ID,Employee\_Surname,@Employee\_Name,@Employee\_Date\_Of\_Birth,@Employee\_Email)

INSERT into GUEST(Employee\_Surname,Employee\_Name,Employee\_Date\_Of\_Birth,Employee\_Email) VALUES (Employee\_Surname,@Employee\_Name,@Employee\_Date\_Of\_Birth,@Employee\_Email)

INSERT into GUEST(Employee\_Surname,Employee\_Name,Employee\_Date\_Of\_Birth,Employee\_Email) VALUES (Employee\_Surname,@Employee\_Name,@Employee\_Date\_Of\_Birth,@Employee\_Email)

INSERT into role(Role\_Description,Role\_Salary,Role\_Hours\_Per\_Day) VALUES (@Role\_Description@,Role\_Salary,@Role\_Hours\_Per\_Day)

**Delete:**

$"DELETE Booking WHERE {iD} = Booking\_Number"

$"DELETE Guest WHERE {iD} = Guest\_ID"

$"DELETE Employee WHERE {iD} = Employee\_ID"

$"DELETE Employee WHERE {iD} = Role\_ID"

$"DELETE Room WHERE {iD} = Room\_Number"

# Example code for maintaining a child entity and reuse of code

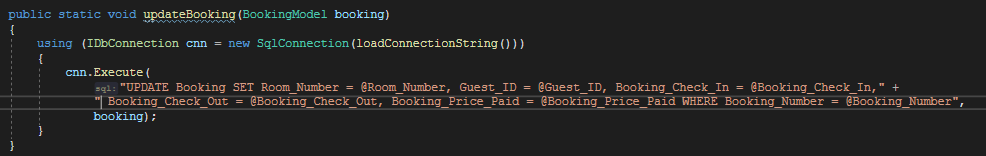
Maintain Booking

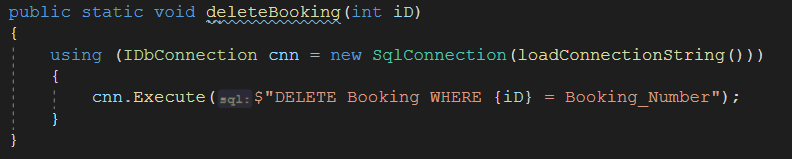
We use NuGet packages: Linq and Dapper to help us maintain the data tables.

Table data is converted to a list of the booking model.

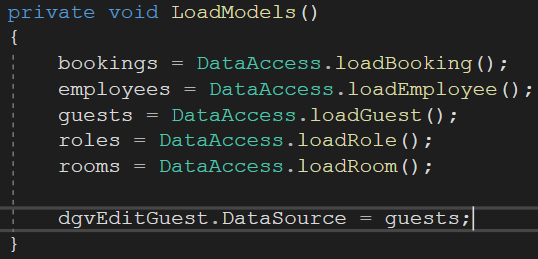
Text

Description automatically generated

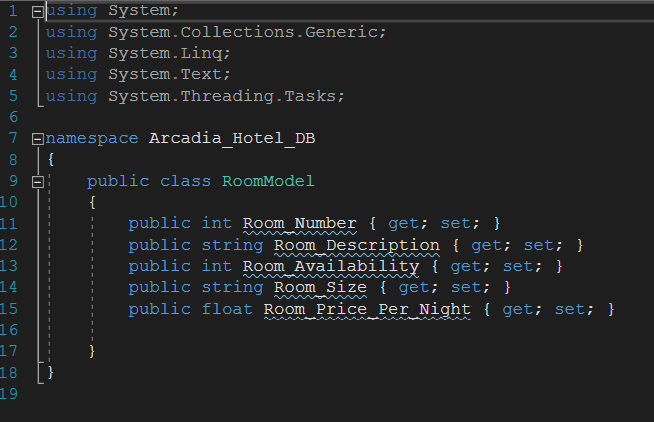




This method is called in the beginning of the program and when something is called or changed in the database.



We use the NuGet packages to read the database tables into lists of objects with the properties containing the columns of the tables.

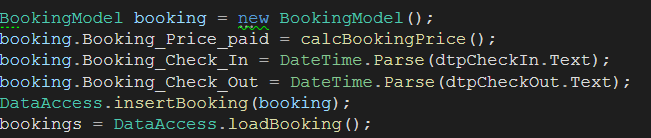


This is how the list models are declared.

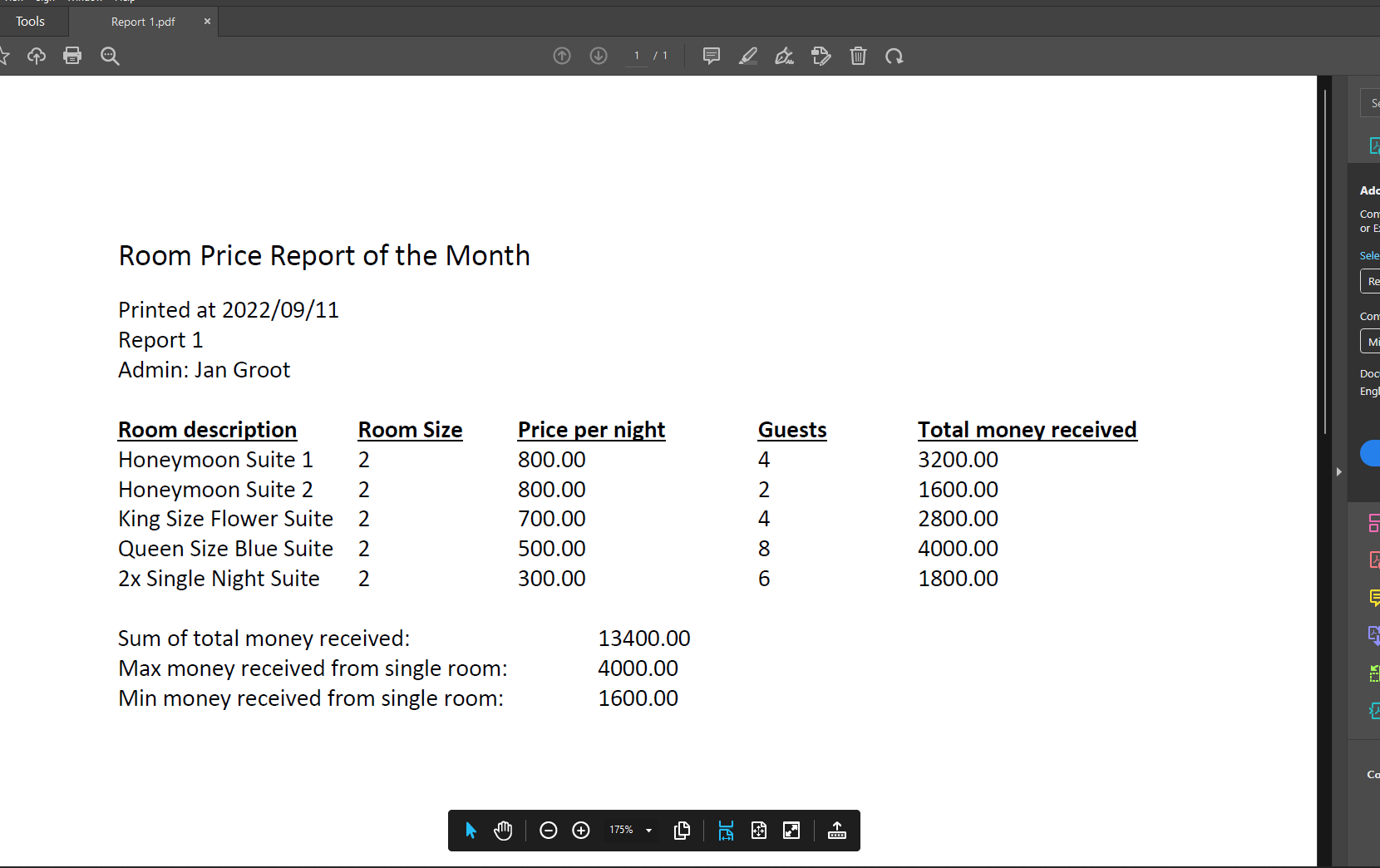
Text

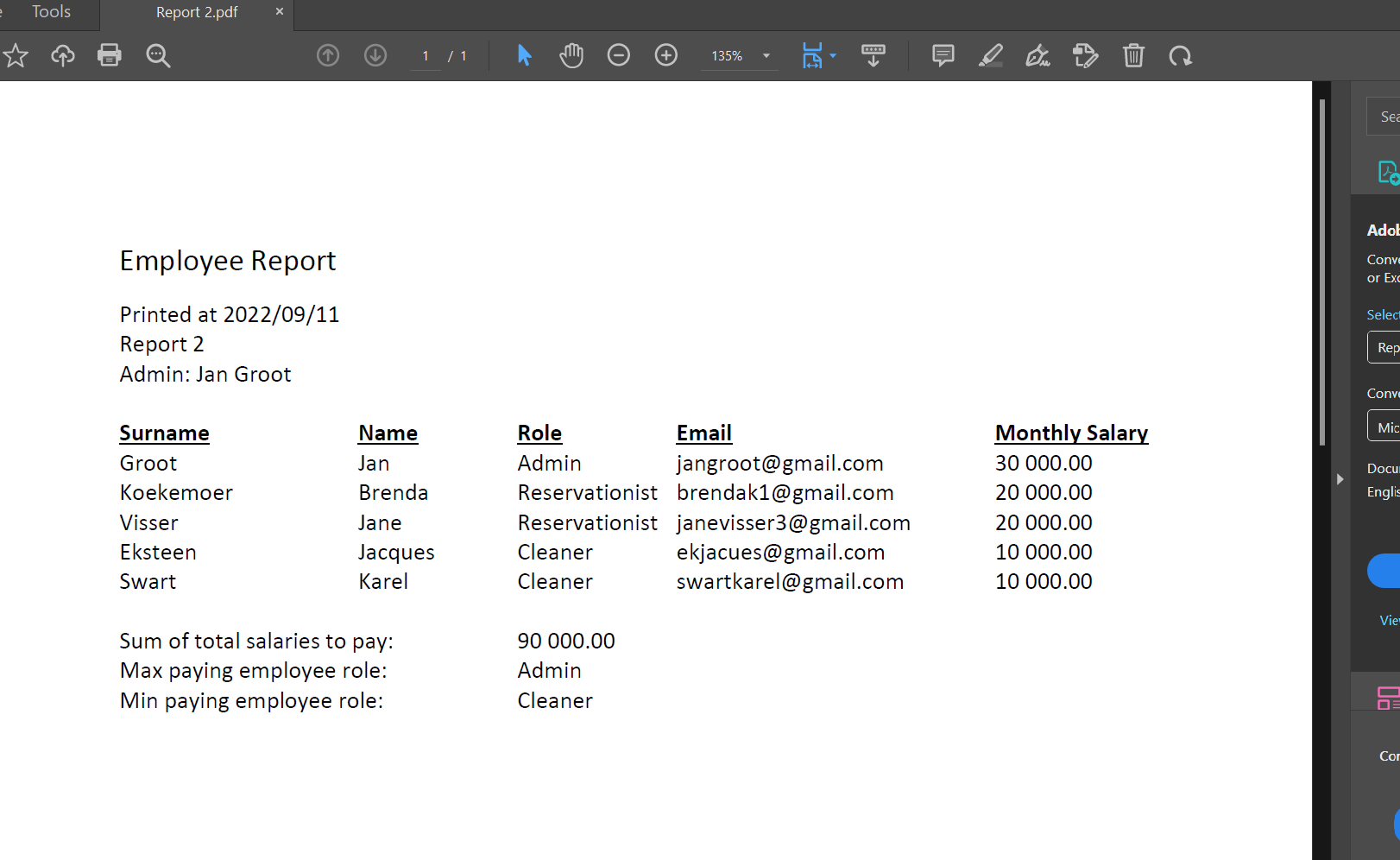
Description automatically generated

This is how bookings are added.



# Output of two reports





# User Manual

**Where to download the system.**

Use the following gitlink to access the files of the system https://github.com/naughty00shortie/Arcadia\_Hotel\_Installation/releases. Download the files from the repository. After downloading the files, navigate to the folder of the system and run the .exe file. Now you are ready and set to go.

**The pc specs that you are going to need to run the system.**

Operating System: Windows 7, or latter.

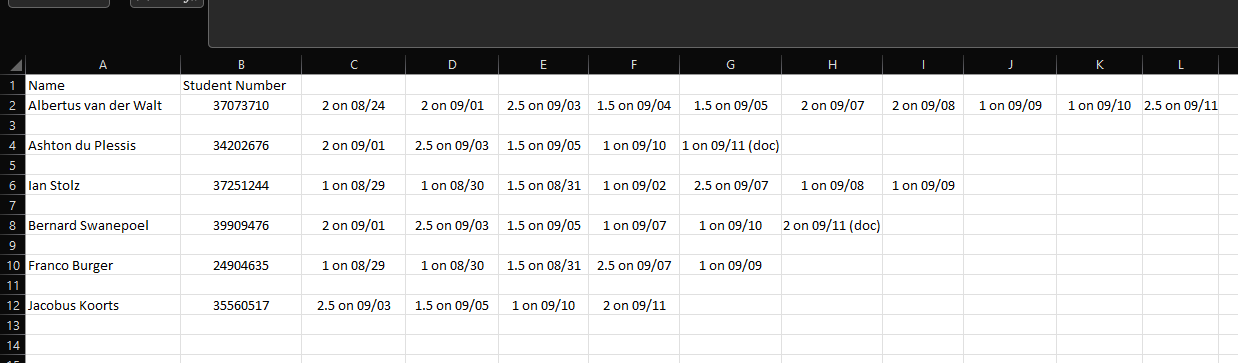
Processor: 2.3 GHz Intel Core 2 Duo or better.

Memory: 1 GB RAM or better.

Graphics: Not needed

Storage: 1 GB of storage space available

# Diary

Diary recorded at 09/11 at 22:36 of the project. The final Diary document will be given when the project will be presented.