Computer Science Department COMPX223-19A Database Practice and Experience Project Deliverable 2

Name: \_Ryan McCarthy\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ID: \_\_\_\_\_\_1348104\_\_\_\_\_\_\_\_\_\_\_\_\_

Marking:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **sum** |
| **1** | **1** | **5** | **3** | **2** | **3** | **3** | **18** |
|  |  |  |  |  |  |  |  |

# Table of contents

[**Table of contents**](#_21m1hvac81dq) **2**

[**Introduction**](#_pdqzm89u5cfs) **3**

[**ER Diagram**](#_enjxxhxexfok) **3**

[**Relation Model and Discussion**](#_71brdtwiv4kl) **4**

[**SQL code**](#_hs4i6kdx29a2) **5**

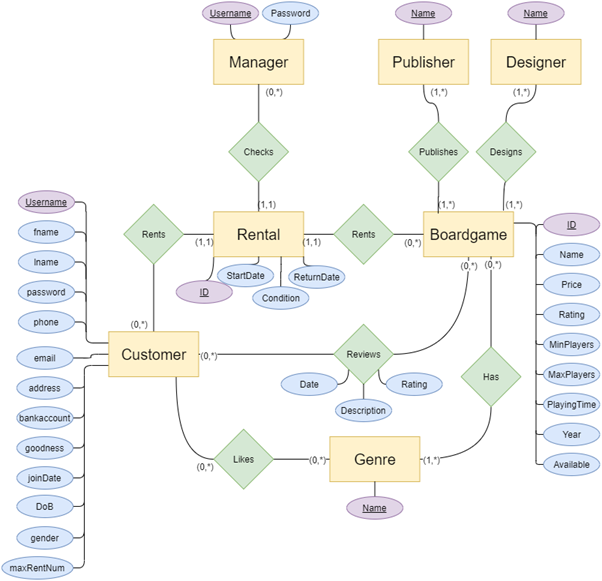
[Database Table screenshots after being Filled with Client data](#_lpjky42etbmf) 10

[**Managers Login Page cod**](#_ayrppda99fhm) **12**

[**Login walkthrough**](#_tk9xdcy1xley) **14**

# Introduction

This report contains the ER diagram and relationship model for the database that I have created for the client. It also contains relevant code to showcase the database. The database holds information on the customers who are renting board games from the clients store. Managers are able to login and see the data for customers at this stage of implementation. I have also put the complete code for both SQL and C# in the same file for the client ( or marker) to look at.



# ER Diagram

# Relation Model and Discussion

The Relation Model based off this ER diagram that I used for my SQL Table is as follows

Customer ( Username, Fname,Lname,password,phone,email,streetAddress, city,bank account,goodness,joinDate, DoB(Date of birth),gender,maxRentNum)

Rental (ID,startDate,condition,ReturnDate,customer-username,manager-username,boardgame-ID)

Genre (name)

Boardgame(ID, name, price, rating,minPlayers,maxPlayers,playingTime,yearMade, available)

Manager (username,password)

Publisher(name)

Designer(name)

Reviews(customer-username, boardgame-ID,rating,date,description)

Likes (customer-user, Genre-name)

Has (Genre-name,boardgame-ID)

Publishes(Publisher-name,boardgame-id)

Designs(designer-name, boardgame-id)

**Discussion of Relations**

For the attribute “Name” on the customer entity, I normalized it to first and last name (Fname,Lname)so they could be stored properly in the database. I also normalised address to street address and city on the customer entity.

I left the price attribute on the boardgame entity even though no data on price was provided for the client because for boardgames to be sold/rented they need a price (one has a placeholder value price for testing). This data can be filled in later by either the client or by database admin or designer (Me) when provided by the client.

I left the publisher name and designer names as a single attribute as they are used as foriegn keys. If broken into first and last names there could be an overlap if only one attribute is used as a foriegn key

# SQL code

NB: The sql file uploaded with the report contains this code.

create table Customer (

username varchar(32) not null,

Fname varchar (20) not null,

Lname varchar(30) not null,

Password varchar(32) not null,

Phone varchar(32) not null,

Email varchar (32),

Street\_address varchar(100) not null,

City varchar (20) not null,

BankAccount bigint not null,

goodness int,

Join\_date date,

DoB date,

Gender varchar (15),

MaxRentNum int,

primary key(username)

)

create table Boardgame(

ID int not null,

Name varchar (32),

Price int ,

Rating decimal,

MinPlayers int not null,

MaxPlayers int not null,

PlayingTime time,

YearMade int not null,

Available varchar(3) not null,

primary key(ID),

)

create table Manager(

Username varchar(32) not null,

Password varchar(32) not null,

primary key (Username),

)

create table Rental(

ID int not null,

StartDate date not null,

condition varchar(12),

ReturnDate date not null,

Customer\_username varchar(32) not null,

Manager\_username varchar(32) not null,

Boardgame\_ID int not null,

primary key (ID),

foreign key (Customer\_username) references customer,

foreign key (Manager\_username) references Manager ,

)

create table Genre (

Name varchar(32) not null,

Primary key (Name),

)

create table Publisher(

Name varchar (32) not null,

primary key (Name),

)

create table Designer(

Name varchar (32) not null,

primary key (Name),

)

create table Reviews(

Customer\_username varchar (32) not null,

Boardgame\_ID int not null,

Rating int not null,

DateReview date not null,

Description varchar(100),

primary key (Customer\_username,Boardgame\_ID),

foreign key (Customer\_username) references Customer,

foreign key (Boardgame\_ID) references BoardGame,

)

create table Likes(

Customer\_username varchar(32) not null,

Genre\_name varchar(32) not null,

primary key(Customer\_username,Genre\_name),

foreign key (Customer\_username) references Customer,

foreign key (Genre\_name) references Genre,

)

create table Has(

Genre\_name varchar(32) not null,

Boardgame\_ID int not null,

primary key(Genre\_name, Boardgame\_ID),

foreign key (Boardgame\_ID) references Boardgame,

foreign key (Genre\_name) references Genre,

)

create table publishes (

Publisher\_name varchar(32) not null,

Boardgame\_ID int not null,

primary key (Publisher\_name, Boardgame\_ID),

foreign key (Boardgame\_ID) references Boardgame,

foreign key (publisher\_name) references Publisher,

)

create table Designs (

Designer\_name varchar(32) not null,

Boardgame\_ID int not null,

primary key (Designer\_name, Boardgame\_ID),

foreign key (Designer\_name) references Designer,

foreign key (Boardgame\_ID) references Boardgame,

)

--board games info

insert into Boardgame values (31260,'Agricola',23,7.96682,1,5, '02:30:00', 2007, 'yes')

insert into Boardgame values(230802,'Azul', null ,7.85835,2,4,'00:45:00',2017,'yes')

insert into Boardgame values (102794,'Caverna: The Cave Farmers', null,8.0392,1,7,'03:30:00',2013,'yes')

insert into Boardgame values (220308,'Gaia Project',null,8.49592,1,4,'02:30:00',2017,'yes')

insert into Boardgame values (193738,'Great WesternTrail', null, 8.28423, 2, 4,'02:30:00', 2016, 'yes')

insert into Boardgame values (276025,'Maracaibo','',8.27257,1,4,'02:00:00',2019,'yes')

insert into Boardgame values (204431,'One Night Ultimate Alien',null,7.07577,4,10,'00:10:00',2017,'yes')

insert into Boardgame values (180956,'One Night Ultimate Vampire',null,6.74924,3,10,'00:10:00',2015,'yes')

insert into Boardgame values (147949,'One Night Ultimate Werewolf',null,7.16687,3,10,'00:10:00',2014,'yes')

insert into Boardgame values (162886,'Spirit Island',null,8.31674,1,4,'02:00:00',2017,'yes')

insert into Boardgame values (192291,'Sushi Go Party!',null,7.48268,2,8,'00:20:00',2016,'yes')

insert into Boardgame values (133473,'Sushi Go!',null,7.05894,2,5,'00:15:00',2013,'yes')

insert into Boardgame values (120677,'Terra Mystica',null,8.16544,2,5,'02:30:00',2012,'yes')

insert into Boardgame values (167791,'Terraforming Mars',null,8.42498,1,5,'02:00:00',2016,'yes')

insert into Boardgame values (266192,'Wingspan',null,8.1218,1,5,'01:10:00',2019,'yes')

-- designers

insert into Designer values('Philippe Guérin')

insert into Designer values('Chris Quilliams')

insert into Designer values('Uwe Rosenberg')

insert into Designer values('Jens Drögemüller')

insert into Designer values('Helge Ostertag')

insert into Designer values('Alexander Pfister')

insert into Designer values('Ted Alspach')

insert into Designer values('Akihisa Okui')

insert into Designer values('R. Eric Reuss')

insert into Designer values('Phil Walker-Harding')

insert into Designer values('Jacob Fryxelius')

insert into Designer values('Elizabeth Hargrave')

--3 managers

insert into Manager values ('Greg25','52greG')

insert into Manager values ('Jeffo69','goTheAllBlacks45')

insert into Manager values ('Karen12','gregsucks4')

--10 customers

insert into Customer values('jim23','Jim','Hickey','WeatherMan1','022756573483','Jimbo@gmail.com','123 Legend ave','New Plymouth',4356231,9.86,'2019/03/16','1949/08/01','Male',2)

insert into Customer values('bigJohn5','John','Key','BlueBlood','02145638392','BigJ@gmail.com','45 Leader Street','Wellington',12321324,8.76,'2019/04/03','1961/03/09','Male',1)

insert into Customer values('sammy4','Samantha','Geer','SammyGee','027243546234','SammyG@yahoo.com','21 Spooky way','Tauranga',12312312,4.96,'2019/03/21','1995/05/12','Female',1)

insert into Customer values('Liam3','Liam','Dood','DoodBod','022171232132','LiamD@gmail.com','63 Doll Steet','Wallogong',789546532,8.74,'2019/05/22','1997/06/09','Male',1)

insert into Customer values('frank8','Frank','West','Westside4','022012121212','FrankieBoy@gmail.com','40 Manley Ave','Hamilton',54675478,9.68,'2019/06/17','1999/04/30','Male',1)

insert into Customer values('Blexzander3','Blex','Bwakes','quirky3','0227653829292','BlexBwakes@hotmail.com','25 Farm street','Hamilton',77723476,3.66,'2019/07/18','1998/03/12','Male',1)

insert into Customer values('Vicky54','Victoria','North','Vic45','0210021212','Vic9@gmail.com','36 Marine Ave','QueensTown',89234883,6.50,'2019/08/09','1997/08/01','Female',1)

insert into Customer values('PetraGee','Petra','Geer','PeeGee','0753432232','PG@gmail.com','1 Inversense Ave','Tauranga',3249832,7.40,'2019/01/20','2001/01/27','Female',1)

insert into Customer values('XX\_COOLGUY\_XX','Reece','McHot','MarginalAf','027612232123','NoodleArms@gmail.com','87 Farm Lane','Te Puke',347238423,8.65,'2019/04/15','2003/05/12','Male',1)

insert into Customer values('DudeMan28','Johnny','Loughran','MavisIsBae','02132123195','JohnnyBoi@gmail.com','1 Drac Way ','Translyvina',239042374,9.99,'2019/05/06','2000/05/03','Male',1)

--10 rentals

insert into Rental values(1,'2020/04/10','Perfect','2020/04/17','jim23','Greg25',31260)

insert into Rental values(2,'2020/04/11','Good','2020/04/18','bigJohn5','Jeffo69',230802)

insert into Rental values(3,'2020/04/12','Average','2020/04/19','sammy4','Karen12',167791)

insert into Rental values(4,'2020/04/20','Perfect','2020/04/27','Liam3','Jeffo69',220308)

insert into Rental values(5,'2020/05/10','Bad','2020/05/17','frank8','Greg25',192291)

insert into Rental values(6,'2020/05/11','Average','2020/05/18','Blexzander3','Greg25',147949)

insert into Rental values(7,'2020/05/12','Average','2020/05/19','Vicky54','Karen12',204431)

insert into Rental values(8,'2020/06/10','Perfect','2020/06/17','XX\_COOLGUY\_XX','Jeffo69',120677)

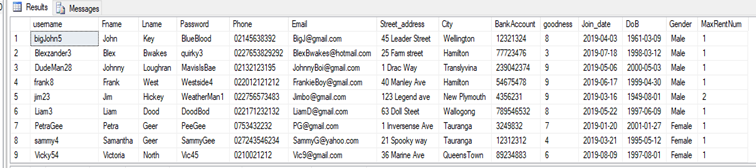
insert into Rental values(9,'2020/06/14','Good','2020/06/21','PetraGee','Karen12',204431)

insert into Rental values(10,'2020/06/14','Bad','2020/06/21','DudeMan28','Jeffo69',147949)

SELECT \* FROM Customer

## Database Table screenshots after being Filled with Client data

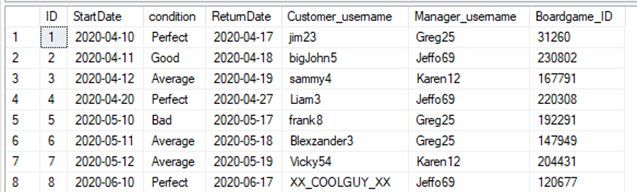
Filled Customer Data Table (showing 9/10)



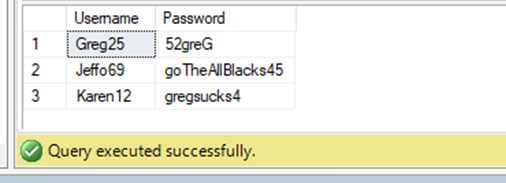
Filled Boardgame data Table (showing 8/10)



Filled Rental Table (Showing 8/10)



Filled Manager Table



# Managers Login Page cod

NB: the project folder for the C# application for accessing the database has been included in the upload.Will need to change SQL connection in sql class as I used my own machine not a remote connection to a lab machine.

private void buttonLogin\_Click(object sender, EventArgs e)

{

string username;

string password;

try

{

// if both textboxes have text in them

if (textBoxUser.Text != "" && textBoxPassword.Text != "")

{

// get the username and password

username = textBoxUser.Text.Trim();

password = textBoxPassword.Text.Trim();

// GET THE DATA FROM THE MANAGERS TABLE IN THE DATABASE

SQL.selectQuery("SELECT\*FROM MANAGER");

// If there is data in the data base

if (SQL.read.HasRows)

{

while (SQL.read.Read())

{

// if the username and password are in the data base

if (username.Equals(SQL.read[0].ToString()) && password.Equals(SQL.read[1]))

{

// log them in

MessageBox.Show("Welcome" + " " + username);

// switch to data screen

Hide();

DisplayPage displayPage = new DisplayPage();

displayPage.ShowDialog();

Close();

}

}

// show if username or password are incorrect/not in the database

MessageBox.Show("Please Check Username and Password are correct");

}

// else database must be empty

else

{

MessageBox.Show("The Data base has no Data");

}

}

// else there must be an empty field

else

{

MessageBox.Show("Please enter a Username and Password");

}

}

catch (Exception ex)

{

MessageBox.Show(ex.ToString());

}

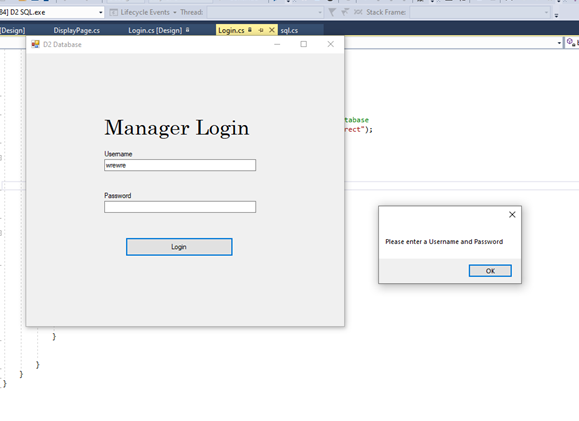
}

# 

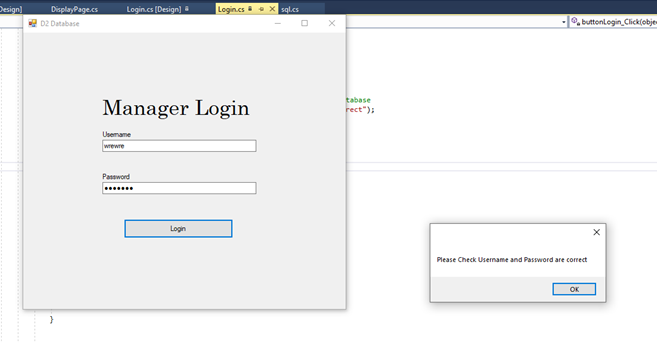
# Login walkthrough

1. Manager enters username and password (if not entered correctly they are notified)

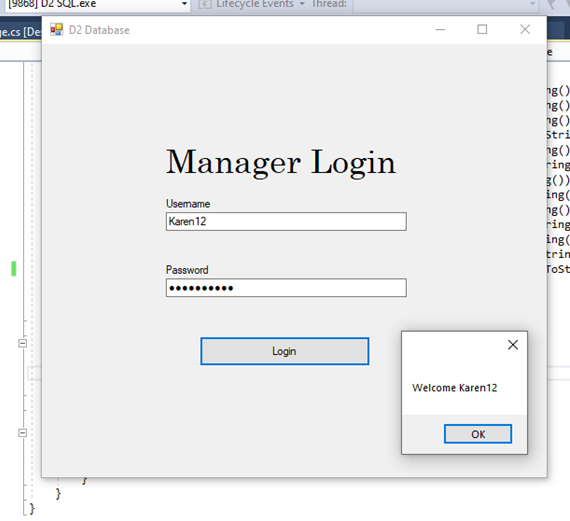
Blank fields



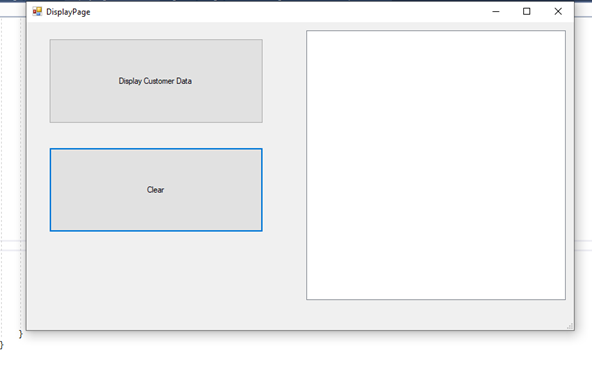
Incorrect login details

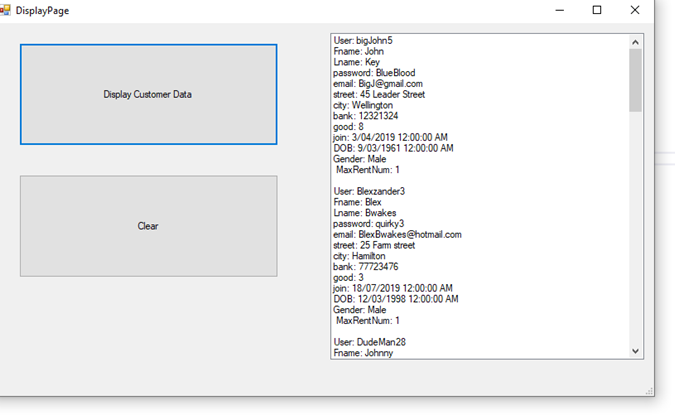


Manager enters correct details



2.Manager is taken to the Customer Data page on clicking ok



3.Manager loads Customer Data by clicking highlighted button (if database is empty they will be notified)