

CHAPTER 1

INTRODUCTION

1.1 Database Management System

A database management system (DBMS) is system software for creating and managing databases. The DBMS provides users and programmers with a systematic way to create, retrieve, update and manage data.

A DBMS makes it possible for end users to create, read, update and delete data in a database. The DBMS essentially serves as an interface between the database and end users or application programs, ensuring that data is consistently organized and remains easily accessible.

DBMS include change management, performance monitoring/tuning and backup and recovery. Many database management systems are also responsible for automated rollbacks, restarts and recovery as well as the logging and auditing of activity.

The DBMS can offer both logical and physical data independence. That means it can protect users and applications from needing to know where data is stored or having to be concerned about changes to the physical structure of data (storage and hardware). As long as programs use the application programming interface (API) for the database that is provided by the DBMS, developers won't have to modify programs just because changes have been made to the database. With relational DBMSs (RDBMSs), this API is SQL, a standard programming.

1.2 INTRODUCTION TO MICROSOFT VISUAL STUDIO 2019

Visual Studio is an **Integrated Development Environment (IDE)** developed by Microsoft to develop GUI (Graphical User Interface), console, Web applications, web apps, mobile apps, cloud, and web services, etc. With the help of this IDE, you can create managed code as well as native code. It uses the various platforms of Microsoft software development software like Windows store, Microsoft Silverlight, and Windows API, etc. It is not a language-specific IDE as you can use this to write code in C#, C++, VB (Visual Basic), Python, JavaScript, and many more languages. It provides support for 36 different programming languages. It is available for Windows as well as for macOS.

Community: It is a **free** version which is announced in 2014. *All other editions are paid.* This contains the features similar to Professional edition. Using this edition, any individual developer can develop their own free or paid apps like *.Net applications*, Web applications and many more. In an enterprise organization, this edition has some limitations. For example, if your organization have more than 250 PCs and having annual revenue greater than \$1 Million (US Dollars) then you are not permitted to use this edition. In a non-enterprise organization, up to five users can use this edition. Its main purpose is to provide the Ecosystem(Access to thousands of extensions) and Languages(You can code in C#, VB, F#, C++, HTML, JavaScript, Python, etc.) support.

1.3 INTRODUCTION TO C#

C# is a general purpose modern and object-oriented programming language pronounced as “C#” It was developed by Microsoft led by Anders Hejlsberg and his team within the .Net initiative and was approved by European Computer Manufacturers Association (ECMA) and International Standards Organization (ISO). C# is among the languages for Common Language Infrastructure.

1.4 PROBLEM STATEMENT

Coffee database is a database software where user purchases coffee in a shop, coffee's have many category customers pays money and gets billing information where employee sales Coffee and produce bills to customer with customer details and the employee details who served it along with the coffee details and its price. This motivated us to develop an application to ease the administrative activities for these kinds of business section.

1.5 APPLICATION AND ADVANTAGES

This software can be used to monitor and store the business data of a BlueBay Coffee store.

The main advantage of this application is that it reduces the use of papers and also reduces the time of recording the data. This is the efficient way of storing the data in the database.

Advantage of using C# and Visual Studio are

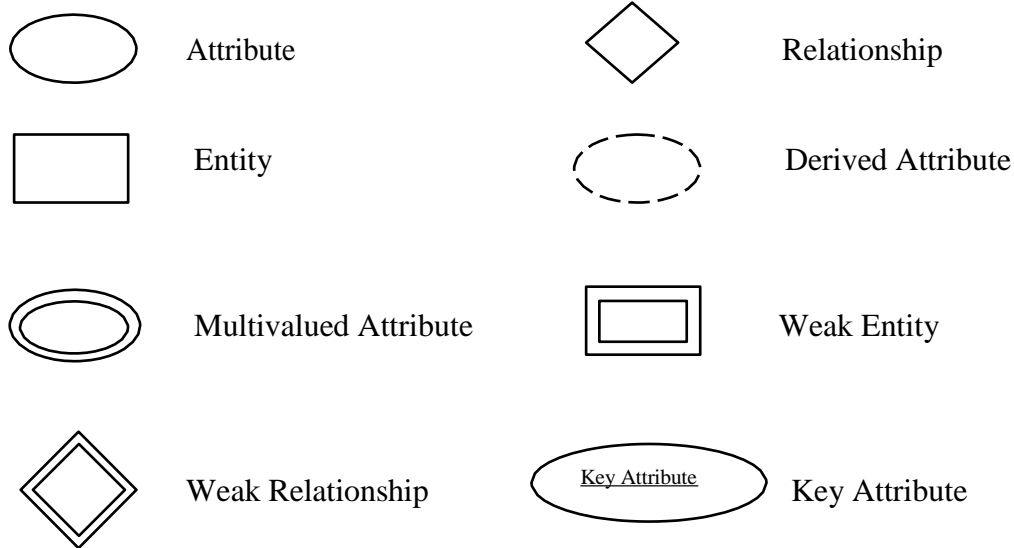
- Easy to write c# code and navigate between the database as it is Inbuilt plugin.
- The BlueBay Coffee Project can be used by more than 100 devices all around the world, while the admin system is connected to non-stop internet and server.
- Visual Studio is considered has best graphic designer user interface which gives it an upper hand compared to all other software.
- It's lightweight, fast, open source and cross-platform nature along with other cool features gives it an extra edge over any other editor.

CHAPTER 2

2.1 ER DAIGRAM

An entity–relationship model describes interrelated things of interest in a specific domain of knowledge. The ER Diagram of our project is shown in the figure:2.2.1

Symbols in Entity Relationship:



RELATONSHIP NAME	PARTICIPATION	CARDINALITY RATIO
Ordered	COFFEE: BILLING	M: N
Services	CUSTOMER: BILLING	M: N
Services	EMPLOYEE: BILLING	M: N
Receiver	EMPLOYEE: PRODUCT	N: 1

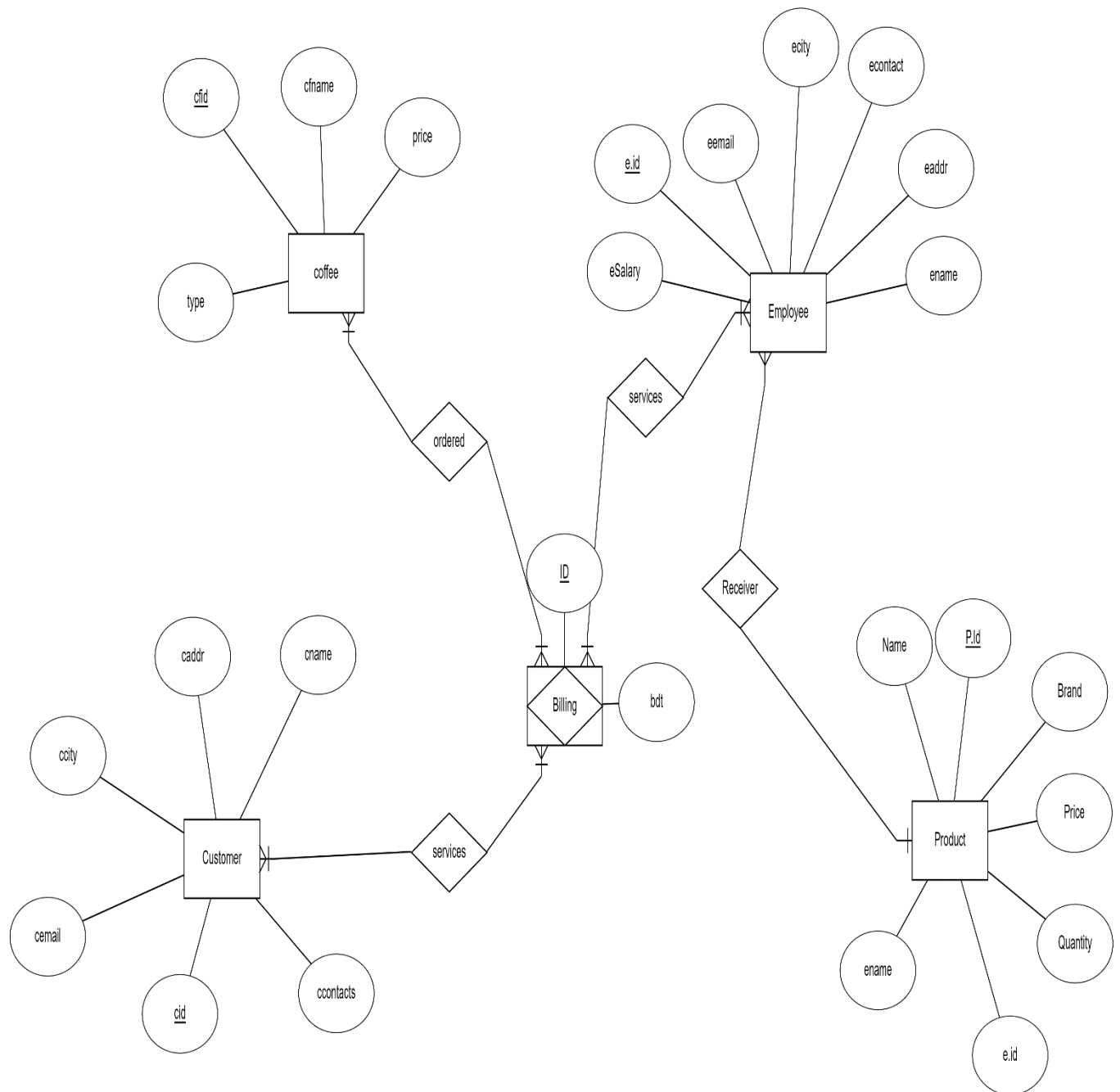
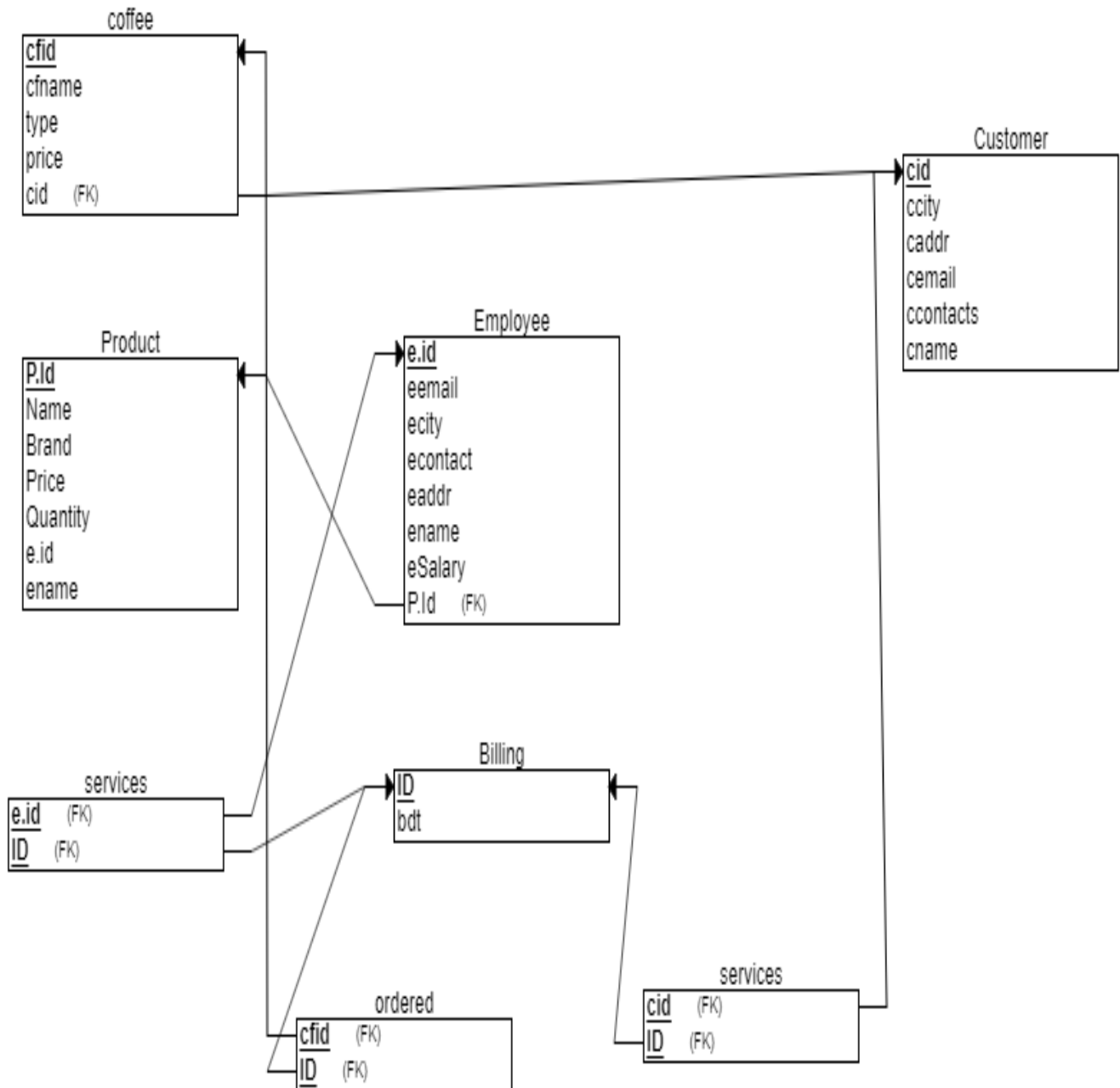


Figure:2.1.1 ER-Diagram of Coffee Database Management

2.2 Schema diagram

The Schema diagram of our project is shown in the figure:2.3.1. Coffee details, Employee details and Customer details are automatically fetched when the required id is written in the billing. Therefore, three foreign key are connected to the Billing. Connection between employee and product is mainly due to build a data of which employee handled which product in the stock.



CHAPTER 3

IMPLEMENTATION

3.1 System Specification

Operating System: Windows XP/7/8/10/MacOS/Linux.

Memory: Minimum of 2GB of RAM, Minimum of 2GB hard disk space.

Backend: Microsoft Visual Studio 2019 SQL Server Express

Frontend: Microsoft Visual Studio 2019

3.2 +Table Structure

3.2.1.Coffee Relation

```
CREATE TABLE [dbo].[acoffee] (  
  [Id] INT NOT NULL,  
  [name] NVARCHAR (50) NOT NULL,  
  [type] NVARCHAR (50) NULL,  
  [price] NVARCHAR (50) NULL,  
  PRIMARY KEY CLUSTERED ([Id])  
);
```

<u>Name</u>	<u>Type</u>
<u>Id</u>	INT
Name	NVARCHAR(50)
Type	NVARCHAR(50)
Price	NVARCHAR(50)

Figure:3.2.1 Coffee Table Structure

3.2.2.Customer Relation

```
CREATE TABLE [dbo].[cust] (  
    [Id]INT NOT NULL,  
    [name] NVARCHAR (50) NULL,  
    [addr] NVARCHAR (50) NULL,  
    [city] NVARCHAR (50) NULL,  
    [contact] NVARCHAR (50) NULL,  
    [email] NVARCHAR (50) NULL,  
    PRIMARY KEY CLUSTERED ([Id] ASC)  
)
```

<u>Name</u>	<u>Type</u>
Name	VARCHAR(50)
<u>ID</u>	INT
Addr	VARCHAR(50)
City	VARCHAR(50)
contact	VARCHAR(50)
Email	VARCHAR(50)

Figure:3.2.2 Customer Table Structure

3.2.3.Employee Relation

```
CREATE TABLE [dbo].[emp] (  
  [Id]    INT NOT NULL,  
  [name]  NVARCHAR (50) NULL,  
  [addr]  NVARCHAR (50) NULL,  
  [city]  NVARCHAR (50) NULL,  
  [contact] NVARCHAR (50) NULL,  
  [gender] NVARCHAR (50) NULL,  
  [email] NVARCHAR (50) NULL,  
  [doj]   NVARCHAR (50) NULL,  
  [salary] NVARCHAR (50) NULL,  
  PRIMARY KEY CLUSTERED ([Id] ASC)  
);
```

<u>Name</u>	<u>Type</u>
<u>ID</u>	INT
Name	NVARCHAR(50)
Addr	NVARCHAR(50)
City	NVARCHAR(50)
Contact	NVARCHAR(50)
Gender	NVARCHAR(50)
Email	NVARCHAR(50)
Doj	NVARCHAR(50)
Salary	NVARCHAR(50)

Figure:3.2.3 Employee Table Structure

3.2.4.Product Relation

```
CREATE TABLE [dbo].[product] (  
  [Id]    INT      IDENTITY (1, 1) NOT NULL,  
  [p_name] NVARCHAR (50) NULL,  
  [brand]  NVARCHAR (50) NULL,  
  [price]  NVARCHAR (50) NULL,  
  [quantity] NVARCHAR (50) NULL,  
  PRIMARY KEY CLUSTERED ([Id] ASC)  
);
```

<u>Name</u>	<u>Type</u>
<u>ID</u>	INT
P_name	NVARCHAR(50)
Brand	NVARCHAR(50)
Price	NVARCHAR(50)
Quantity	NVARCHAR(50)

Figure :3.2.4 Product Table Structure

3.2.5. Order Relation

```
CREATE TABLE [dbo].[p_order] (  
    [Id] INT NOT NULL,  
    [cust_no] NVARCHAR (50) NULL,  
    [name] NVARCHAR (50) NULL,  
    [addr] NVARCHAR (50) NULL,  
    [city] NVARCHAR (50) NULL,  
    [c_name] NVARCHAR (50) NULL,  
    [price] NVARCHAR (50) NULL,  
    [date] NVARCHAR (50) NULL,  
    [Eid] INT NULL,  
    PRIMARY KEY CLUSTERED ([Id] ASC)
```

<u>Name</u>	<u>Type</u>
<u>ID</u>	NUMBER(3)
Cust_no	NVARCHAR(50)
Name	NVARCHAR(50)
Addr	NVARCHAR(50)
City	NVARCHAR(50)
C_name	NVARCHAR(50)
Price	NVARCHAR(50)
Date	NVARCHAR(50)
Eid	INT

Figure:3.2.5 Order Table Structure

Functionalities

3.3.1 Connecting to Database

The “BlueBay Coffee Database System” has been developed C#. It is used by Microsoft SQL Server Express database for storing the data. It is connected by the following syntax:

```
SqlConnection con = new
    SqlConnection(@"Data
    Source=(LocalDB)\MSS
    QLLocalDB;AttachDbF
    ilename=C:\Users\Sudh
    vina
    A.S\Downloads\Coffee
    Bluebay\CoffeShopMan
    egementSystemCSharp\
    CoffeShopManegement
    SystemCSharp\coffee.m
    df;Integrated
    Security=True");
```

3.3.1 Add

Add operation is used to add an Customer details, Employee detail, Coffee detail, Product detail ,Place Order detail to the coffee database.

```
J
private void button1_Click(object sender, EventArgs e)
{
    try
    {
        new System.Net.Mail.MailAddress(this.textBox6.Text);
        // return;
    }
    catch (ArgumentException e1)
    {
        MessageBox.Show("empty");
        return;
    }
    catch (FormatException e2)
    {
        MessageBox.Show("invalid email");
        return;
    }
}
```

```
}

SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Sudhvina
A.S\Downloads\CoffeeBluebay\CoffeShopManegementSystemCSharp\CoffeShopManegementSystemCSharp
\coffee.mdf;Integrated Security=True");
con.Open();

try
{
    string str = " INSERT INTO cust(Id,name,addr,city,contact,email) VALUES(" + textBox1.Text + "," +
textBox2.Text + "," + textBox3.Text + "," + textBox4.Text + "," + textBox5.Text + "," + textBox6.Text + ")";
    ";

    SqlCommand cmd = new SqlCommand(str, con);
    cmd.ExecuteNonQuery();

    //-----//

    string str1 = "select max(Id) from cust;";

    SqlCommand cmd1 = new SqlCommand(str1, con);

    SqlDataReader dr = cmd1.ExecuteReader();
    if (dr.Read())
    {
        MessageBox.Show("Customers Information Registered Successfully..");
    }
    this.Close();
}
catch (SqlException ee)
{
    MessageBox.Show(ee.Message);
}

con.Close();
}
```

3.3.2 Delete

Delete operation is used to delete the entry made to the Customer details, Employee detail, Coffee detail, Product detail ,Place Order by pressing delete key from the firework database.

```
private void button3_Click(object sender, EventArgs e)
{
    try
    {
        SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Sudhvina
A.S\Downloads\CoffeeBluebay\CoffeShopManegementSystemCSharp\CoffeShopManegementSyste
mCSharp\coffee.mdf;Integrated Security=True");
        con.Open();

        string str = "DELETE FROM cust WHERE Id = '" + textBox1.Text + "'";

        SqlCommand cmd = new SqlCommand(str, con);
        cmd.ExecuteNonQuery();
        con.Close();
        MessageBox.Show(" Customer Information is Removed Succesfully");

        SqlCommand cmd5 = new SqlCommand(str, con);
        SqlDataAdapter da = new SqlDataAdapter(cmd5);
        DataTable dt = new DataTable();
        da.Fill(dt);

        dataGridView1.DataSource = new BindingSource(dt, null);

    }

    catch (SqlException ex)
    {
        MessageBox.Show(ex.Message);
        MessageBox.Show("Please Enter Customer Id..");
    }
}
```

3.3.3 View

View button is used to view the values entered into the table from coffee database

```

private void textBox1_TextChanged(object sender, EventArgs e)
{
    SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Sudhvina
A.S\Downloads\CoffeeBluebay\CoffeShopManegementSystemCSharp\CoffeShopManegementSyste
mCSharp\coffee.mdf;Integrated Security=True");
    con.Open();
    if (textBox1.Text != "")
    {
        try
        {
            string getCust = "select name,addr,city,contact,email from cust where Id=" +
textBox1.Text + " ";
            SqlCommand cmd = new SqlCommand(getCust, con);
            SqlDataReader dr;
            dr = cmd.ExecuteReader();
            if (dr.Read())
            {
                textBox2.Text = dr.GetValue(0).ToString();
                textBox3.Text = dr.GetValue(1).ToString();
                textBox4.Text = dr.GetValue(2).ToString();
                textBox5.Text = dr.GetValue(3).ToString();
                textBox6.Text = dr.GetValue(4).ToString();
            }
            else
            {
                textBox2.Text = "";
                textBox3.Text = "";
                textBox4.Text = "";
                textBox5.Text = "";
                textBox6.Text = "";
            }
        }
        catch (SqlException excep)
        {
            MessageBox.Show(excep.Message);
        }
        con.Close();
    }
}

```

3.3.4 Reset

Reset button is used to reset the values inserted in every attributes of each table

```
private void button5_Click(object sender, EventArgs e)
{
SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Sudhvina
A.S\Downloads\CoffeeBluebay\CoffeShopManegementSystemCSharp\CoffeShopManegementSyste
mCSharp\coffee.mdf;Integrated Security=True");
con.Open();

textBox1.Text = "";
textBox2.Text = "";
textBox3.Text = "";
textBox4.Text = "";
textBox5.Text = "";
textBox6.Text = "";
}
```

3.3.5 Trigger

Trigger operation is used raise an error message if the entered amount is negative in coffee table.

```
CREATE TRIGGER [dbo].[PINS]
ON [dbo].[acoffee]
INSTEAD OF INSERT
AS
BEGIN
SET NOCOUNT ON;
DECLARE @Id int, @name nvarchar(50),@type nvarchar(50), @price int
DECLARE @MESSAGE VARCHAR(100)

SELECT @Id = INSERTED.Id,
       @name = INSERTED.name,
       @type = INSERTED.type,
       @price = INSERTED.price
FROM INSERTED

IF(@price<0)
```

```
BEGIN
    SET @MESSAGE = 'INVALID PRICE'
    RAISERROR(@MESSAGE,16,1)
END
ELSE
    BEGIN
        INSERT INTO acoffee VALUES(@Id,@name,@type,@price)

    END
END
```

3.3.6 Stored Procedure

A procedure is created in our project to update the contents of table coffee.

```
CREATE PROCEDURE [dbo].UPDATECOFFEE
    @Id int,
    @name nvarchar (50),
    @type nvarchar (50),
    @price nvarchar (50)
AS
    UPDATE acoffee SET name=@name, type=@type, price=@price WHERE Id=@Id
RETURN 0
```

CHAPTER 4

RESULTS

BlueBay Coffee is a database software where user purchases coffee in shop. Our Coffee have many categories for customers and placing order is much easier as it stores entire value of coffee, customer key details and employee key details in the billing of the system

4.1 Snapshots

4.1.1 Login Page

It is a multiple user login system, where one ca



Figure: 4.1.1 Login Page

4.1.2 Home Page

In home page the tables in the database can be selected to insert, delete, display, etc.

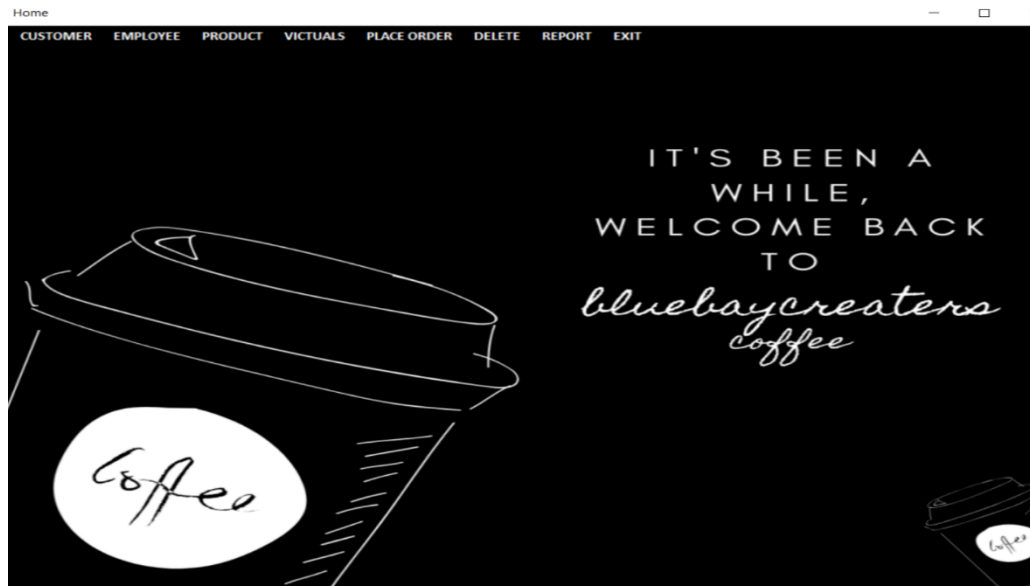


Figure: 4.1.2 Home Page

4.1.3 Coffee details

In Coffee page we can insert, delete, modify and display the details of Coffee and Snacks

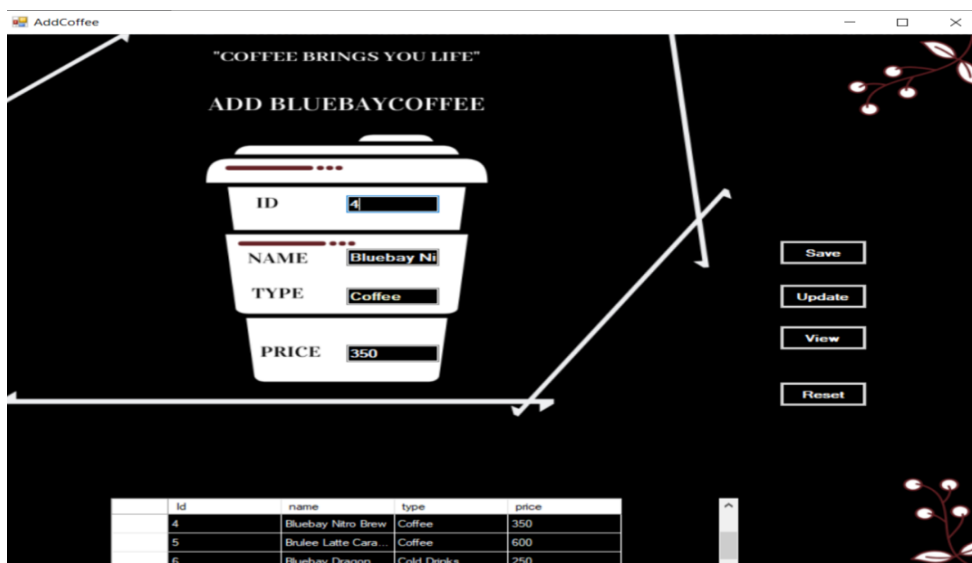


Figure: 4.1.3 Coffee details

4.1.4 Customer details

In customer page we can insert, delete, modify and display the details of customer.

	Id	name	addr	city	contact	email
▶	1	Yeshwanth	M.G.Road	Bengaluru	9741116599	yeshwanth
	2	Sudhvina	Balehonnur	Chikmangalore	9887251632	sudh7871@gmail.com
	3	Samantha	Malleshwaram	Banglore	894567825	anth@bb.coffee
	4	Sudh	Balehonnur	Bengaluru	9887251632	sudh7871@gmail.com

Figure: 4.1.4 Customer details

4.1.5 Place Order details

In billing page, we can insert, delete, modify and display the details of billing details

	Id	cust_no	name	addr	city	c_name
▶	3	3	Samantha	Malleshwaram	Banglore	7
	5	1	Yeshwanth	M.G.Road	Bengaluru	1
*						

Figure: 4.1.5 Place Order details

4.1.6 Employee details

In Employee page we can insert, delete, modify and display the details of Employee.

BLUEBAY EMPLOYEE
the best employee gets best coffee

NAME:
 ADDRESS:
 CITY:
 CONTACTS:
 ID TO DELETE:

GENDER: ☐ Male ☒ Female
 EMAIL:
 DATE OF JOINING:
 SALARY:

Buttons: Save, Reset, View

id	name	addr	city	contact
0	Steve	M.G Road	Bangalore	8799928837
1	Sanjana	Rajajinagar	Bangalore	879920398
3	Seung Hun	M.G Road	Bangalore	9267999030

Figure: 4.1.6 Employee details

4.1.7 Product details

In Product page we can insert, delete, modify and display the details of ingredients required for making coffee or snacks

BLUEBAY PRODUCT
TESTING A PRODUCT IS A LEARNING PROCESS

id:
 name:
 brand:
 price:
 quantity:

Buttons: Save, Reset, View

id	p_name	brand	price	quantity
9	Bluebay ColdBre...	Bluebaycreators	3000	1
10	Bluebay DoubleS...	Bluebaycreators	800	1
11	Bluebay Cappuc...	Bluebaycreators	200	1
12	Bluebay Pike Pla...	Bluebaycreators	2500	1

Figure: 4.1.7 Product details

4.1.8 Select * from Coffee

Id	name	type	price
1	Bluebay Americ...	Coffee	120
2	Bluebay Blonde...	Coffee	150
3	Cinnamon Dolc...	Coffee	250
4	Bluebay Nitro B...	Coffee	350
5	Brulee Latte Car...	Coffee	600
6	Bluebay Drago...	Cold Drinks	250
7	Bluebay Violet ...	Cold Drink	450
8	Mango Yogurt	Yogurt	45
NULL	NULL	NULL	NULL

4.1.9 Select * from Customer

Id	name	addr	city	contact	email
1	Yeshwanth	Mg	Bengaluru	9741116599	yeshwanth123...
2	Sudhvina	Balehonnur	Chikmangalore	9887251632	sudh7871@gm...
3	Samantha	Malleshwaram	Banglore	894567825	anth@bb.com
4	Sindu	Basvangudi	Banglore	9865523178	ssindu@bb.com
NULL	NULL	NULL	NULL	NULL	NULL

4.1.10 Select * from Employee

Id	name	addr	city	contact	gender	email	doj	salary
2	Steve	M.G.Road	Bangalore	8799928837	Male	steverocks@out...	8/8/2016	8,30,00
1	Sanjana	Rajajinagar	Bangalore	879920398	Female	sanjana@gmail...	2/3/2018	5,20,000
3	Seung Hun	M.G.Road	Bangalore	9267999030	Male	seung@yahoo.c...	19/10/2017	10,99,000
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL


4.1.11 Select * from Place Order

Id	cust_no	name	addr	city	c_name	price	date	Eld
1	1	Yeshwanth	M.G.Road	Bengaluru	1	120	4/4/2019	9
2	2	Sudhvina	Balehonnur	Chikmangalore	7	450	4/4/2019	2
3	3	Samantha	Malleshwaram	Banglore	7	450	9/9/2019	1
4	1	Yeshwanth	M.G.Road	Bengaluru	5	600	9/10/2019	2
5	1	Yeshwanth	M.G.Road	Bengaluru	1	120	9/9/2019	1
6	1	Yeshwanth	M.G.Road	Bengaluru	1	120	2/2/2019	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL


4.1.12 Select * from Product

Id	p_name	brand	price	quantity	emId
15	Mango Powder	Amul	1000	1kg	2
16	Choco Powder	Starbucks	5600	500grams	1
NULL	NULL	NULL	NULL	NULL	NULL


4.1.13 Desc Employee

	Name	Data Type	Allow Nulls
	Id	int	<input type="checkbox"/>
	name	nvarchar(50)	<input checked="" type="checkbox"/>
	addr	nvarchar(50)	<input checked="" type="checkbox"/>
	city	nvarchar(50)	<input checked="" type="checkbox"/>
	contact	nvarchar(50)	<input checked="" type="checkbox"/>
	gender	nvarchar(50)	<input checked="" type="checkbox"/>
	email	nvarchar(50)	<input checked="" type="checkbox"/>
	doj	nvarchar(50)	<input checked="" type="checkbox"/>
	salary	nvarchar(50)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>


4.1.13 Desc Coffee

	Name	Data Type	Allow Nulls
	Id	int	<input type="checkbox"/>
	name	nvarchar(50)	<input type="checkbox"/>
	type	nvarchar(50)	<input checked="" type="checkbox"/>
	price	nvarchar(50)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>


4.1.14 Desc Order

	Name	Data Type	Allow Nulls
	Id	int	<input type="checkbox"/>
	cust_no	nvarchar(50)	<input checked="" type="checkbox"/>
	name	nvarchar(50)	<input checked="" type="checkbox"/>
	addr	nvarchar(50)	<input checked="" type="checkbox"/>
	city	nvarchar(50)	<input checked="" type="checkbox"/>
	c_name	nvarchar(50)	<input checked="" type="checkbox"/>
	price	nvarchar(50)	<input checked="" type="checkbox"/>
	date	nvarchar(50)	<input checked="" type="checkbox"/>
	EId	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

4.1.15 Desc Product

	Name	Data Type	Allow Nulls	Default
	Id	int	<input type="checkbox"/>	
	p_name	nvarchar(50)	<input checked="" type="checkbox"/>	
	brand	nvarchar(50)	<input checked="" type="checkbox"/>	
	price	nvarchar(50)	<input checked="" type="checkbox"/>	
	quantity	nvarchar(50)	<input checked="" type="checkbox"/>	
	emld	int	<input checked="" type="checkbox"/>	
			<input type="checkbox"/>	

4.1.16 Desc Customer

	Name	Data Type	Allow Nulls	Default
	Id	int	<input type="checkbox"/>	
	name	nvarchar(50)	<input checked="" type="checkbox"/>	
	addr	nvarchar(50)	<input checked="" type="checkbox"/>	
	city	nvarchar(50)	<input checked="" type="checkbox"/>	
	contact	nvarchar(50)	<input checked="" type="checkbox"/>	
	email	nvarchar(50)	<input checked="" type="checkbox"/>	
			<input type="checkbox"/>	

CONCLUSION

It has been a matter of immense pleasure, honor and challenge to have this opportunity to take up this project and complete it successfully. Our project maintains the entire database of an BlueBay Coffee Management. The software can be used in any Coffee shop even any snacks shop by just changing the attribute names, to maintain its coffee, products , employee, customer and Place Order details. While developing this project we have learnt a lot about Coffee database, we have also learnt how to make it user friendly (easy to use and handle) by hiding the complicated parts of its from users. During the development process we studied more about developing a software, how to implement the backend stored database in the real time system. We have tried to implement the project making it as user friendly and error free as possible. In future this project can be improved by recording the day-to-day activities taking place in a coffee database. So our project proves that, without investing money and by providing the required efforts we can make many people life move easier.

REFERENCES

1. C# 7.0 Pocket Reference: Instant help for Beginners by Ben Albhari and Joseph Albhari.
2. Raghu Ramakrishnan, and Gehrke, “Database management systems”, McGraw Hill, 3rd Edition, 2014,
3. C# for Dummies by John Paul and Bill Sempf 1st Edition, Kindle Edition.