# Pearson Higher Nationals in

# Computing

## Unit 04: Database Design & Development

## Assignment 01

## Higher Nationals

Internal verification of assessment decisions – BTEC (RQF)

| INTERNAL VERIFICATION – ASSESSMENT DECISIONS | | | | |
|---|---|---|---|---|
| **Programme title** | **BTEC HND in Computing** | | | |
| **Assessor** | Ms Gayani Nisansala | **Internal Verifier** | Mr Lakindu Premachandra | |
| **Unit(s)** | **Unit 04: Database Design & Development** | | | |
| **Assignment title** | **Database Solution for Polly Pipe** | | | |
| **Student's name** | Ranudi Gayathmie Kariyapperuma | | | |
| **List which assessment criteria the Assessor has awarded.** | **Pass** | **Merit** | **Distinction** | |
| | | | | |

| INTERNAL VERIFIER CHECKLIST | | | |
|---|---|---|---|
| **Do the assessment criteria awarded match those shown in the assignment brief?** | Y/N | | |
| **Is the Pass/Merit/Distinction grade awarded justified by the assessor's comments on the student work?** | Y/N | | |
| **Has the work been assessed accurately?** | Y/N | | |
| **Is the feedback to the student:**<br>Give details:<br>• Constructive?<br>• Linked to relevant assessment criteria?<br>• Identifying opportunities for improved performance?<br>• Agreeing actions? | <br><br>Y/N<br>Y/N<br>Y/N<br><br>Y/N | | |
| **Does the assessment decision need amending?** | Y/N | | |
| **Assessor signature** | ranudigk@gmail.com | **Date** | 03/04/2023 |
| **Internal Verifier signature** | | **Date** | |
| **Programme Leader signature** (if required) | | **Date** | |

| Confirm action completed | | | |
|---|---|---|---|
| **Remedial action taken**<br>Give details: | | | |
| **Assessor signature** | | **Date** | |
| **Internal Verifier signature** | | **Date** | |
| **Programme Leader signature** (if required) | | **Date** | |

## Higher Nationals - Summative Assignment Feedback Form

| | | | |
|---|---|---|---|
| **Student Name/ID** | | | |
| **Unit Title** | Unit 04: Database Design & Development | | |
| **Assignment Number** | 1 | **Assessor** | |
| **Submission Date** | | **Date Received 1st submission** | |
| **Re-submission Date** | | **Date Received 2nd submission** | |

**Assessor Feedback:**

**LO1 Use an appropriate design tool to design a relational database system for a substantial problem**

**Pass, Merit & Distinction Descripts** **P1** ☐ **M1** ☐ D1 ☐

**LO2 Develop a fully functional relational database system, based on an existing system design**

**Pass, Merit & Distinction Descripts** **P2** ☐ **P3** ☐ **M2** ☐ **M3** ☐ **D2** ☐

**LO3 Test the system against user and system requirements.**

**Pass, Merit & Distinction Descripts** **P4** ☐ **M4** ☐ **D2** ☐

**LO4 Produce technical and user documentation.**

**Pass, Merit & Distinction Descripts** **P5** ☐ **M5** ☐ **D3** ☐

| | | |
|---|---|---|
| **Grade:** | **Assessor Signature:** | **Date:** |
| **Resubmission Feedback:** | | |
| **Grade:** | **Assessor Signature:** | **Date:** |
| **Internal Verifier's Comments:** | | |
| **Signature & Date:** | | |

* Please note that grade decisions are provisional. They are only confirmed once internal and external moderation has taken place and grades decisions have been agreed at the assessment board.

## Assignment Feedback

| | | | |
|---|---|---|---|
| **Formative Feedback: Assessor to Student** | | | |
| **Action Plan** | | | |
| **Summative feedback** | | | |
| **Feedback: Student to Assessor** | | | |
| **Assessor signature** | | **Date** | |
| **Student signature** | **ranudigk@gmail.com** | **Date** | 23.01.2023 |

**General Guidelines**

1. A Cover page or title page – You should always attach a title page to your assignment. Use previous page as your cover sheet and make sure all the details are accurately filled.
2. Attach this brief as the first section of your assignment.
3. All the assignments should be prepared using a word processing software.
4. All the assignments should be printed on A4 sized papers. Use single side printing.
5. Allow 1" for top, bottom , right margins and 1.25" for the left margin of each page.

**Word Processing Rules**

1. The font size should be **12** point, and should be in the style of **Time New Roman**.
2. **Use 1.5 line spacing**. Left justify all paragraphs.
3. Ensure that all the headings are consistent in terms of the font size and font style.
4. Use **footer function in the word processor to insert Your Name, Subject, Assignment No, and Page Number on each pag**e. This is useful if individual sheets become detached for any reason.
5. Use word processing application spell check and grammar check function to help editing your assignment.

**Important Points:**

1. It is strictly prohibited to use textboxes to add texts in the assignments, except for the compulsory information. eg: Figures, tables of comparison etc. Adding text boxes in the body except for the before mentioned compulsory information will result in rejection of your work.
2. Carefully check the hand in date and the instructions given in the assignment. Late submissions will not be accepted.
3. Ensure that you give yourself enough time to complete the assignment by the due date.
4. Excuses of any nature will not be accepted for failure to hand in the work on time.
5. You must take responsibility for managing your own time effectively.
6. If you are unable to hand in your assignment on time and have valid reasons such as illness, you may apply (in writing) for an extension.
7. Failure to achieve at least PASS criteria will result in a REFERRAL grade .
8. Non-submission of work without valid reasons will lead to an automatic RE FERRAL. You will then be asked to complete an alternative assignment.
9. If you use other people's work or ideas in your assignment, reference them properly using HARVARD referencing system to avoid plagiarism. You have to provide both in-text citation and a reference list.
10. If you are proven to be guilty of plagiarism or any academic misconduct, your grade could be reduced to A REFERRAL or at worst you could be expelled from the course

**Student Declaration**

I hereby, declare that I know what plagiarism entails, namely to use another's work and to present it as my own without attributing the sources in the correct form. I further understand what it means to copy another's work.

1. I know that plagiarism is a punishable offence because it constitutes theft.
2. I understand the plagiarism and copying policy of Edexcel UK.
3. I know what the consequences will be if I plagiarise or copy another's work in any of the assignments for this program.
4. I declare therefore that all work presented by me for every aspect of my program, will be my own, and where I have made use of another's work, I will attribute the source in the correct way.
5. I acknowledge that the attachment of this document signed or not, constitutes a binding agreement between myself and Pearson, UK.
6. I understand that my assignment will not be considered as submitted if this document is not attached to the assignment.

**ranudigk@gmail.com**
**Student's Signature:** **Date: 03.04.2023**
**(*Provide E-mail ID*)** **(*Provide Submission Date*)**

## Higher National Diploma in Computing

Assignment Brief

| | |
|---|---|
| Student Name /ID Number | Ranudi Gayathmie Kariyapperuma / 00104243 |
| **Unit Number and Title** | **Unit 4: Database Design & Development** |
| Academic Year | 2021/2022 |
| Unit Tutor | Database Design & Developement |
| **Assignment Title** | **Data base system for Polly Pipe** |
| Issue Date | 22.01.2023 |
| Submission Date | 04.03.2023 |
| IV Name & Date | |

| **Submission format** |
|---|
| Part 1: The submission should be in the form of an individual written report written in a concise, formal business style using single spacing and font size 12. You are required to make use of headings, paragraphs and subsections as appropriate, and all work must be supported with research and referenced using Harvard referencing system. Please also provide in-text citation and bibliography using Harvard referencing system. The recommended word limit is 3,000–3,500 words, although you will not be penalised for exceeding the total word limit.<br>Part 2: The submission should be in the form of a fully functional relational database system demonstrated to the Tutor; and an individual written report (please see details in Part 1 above).<br>Part 3: The submission should be in the form of a witness statement of the testing completed by the Tutor; technical documentation; and a written report (please see details in Part 1 above). |
| **Unit Learning Outcomes:** |
| LO1 Use an appropriate design tool to design a relational database system for a substantial problem.<br>LO2 Develop a fully functional relational database system, based on an existing system design.<br>LO3 Test the system against user and system requirements.<br>LO4 Produce technical and user documentation. |
| **Assignment Brief and Guidance:** |

**Assignment brief**

Polly Pipe is a water sports provider and installer based in Braintree, England. They need you to design and implement a database that meets the data requirements. These necessities are defined in this scenario and below are samples of the paper records that the Polly Pipe preserves.

Polly Pipe is focused in placing aquariums at business customers. Customers can request several installations, but each installation is tailor-made for a specific customer. Facilities are classified by type. One or more employees are assigned to each facility. Because these facilities are often very large, they can include carpenters and masons as well as water installers. The facilities use equipment such as aquariums, air pumps and thermostats. There can be multiple computers in a facility.

Below are examples of paper records that Polly Pipe currently maintains.

**Staff Management Record**

| Staff Number | Name | Type |
|---|---|---|
| **SHA1** | Dave Clark | Plumber |
| **SHA8** | John Smith | Installation Manager |
| **SHA2** | Freddy Davies | Aquatics installer |
| **SHA11** | McCloud | Aquatics installer |
| **SHA23** | Satpal Singh | Plumber |
| **SHA66** | Winstn Kodogo | Aquatics installer |
| **SHA55** | Alison Smith | Brick Layer |

**Equipment Type Table**

| Type | Equipment |
|---|---|
| **Tanks** | 20 gallon tank, 50 gallon tank, 100 gallon tank, 200 gallon tank |
| **Thermostats** | Standard, Super |
| **Air Pumps** | Standard, Super |
| **Filters** | Air driven, Undergravel |

**Instillation Management Form**

| Installation ID | Installation Type | Installation Name and Address | Customer | Equipment | Types of Staff Required | Period of Staff assignment |
|---|---|---|---|---|---|---|
| **234** | Freshwater Tropical | Oak House, 17 Wroxton Road, Hertfordshire, H5 667 | Lee A. sun | 2 air pumps<br>200 gallons fish tank<br>1 x standard thermostat | 1 x Carpenter<br>1 x Aquatics installer<br>1 x Electrician | From 1st September 2012 |
| **654** | Freshwater Cold | Bayliss House, Orange Street, Kent, K7 988 | Sally Dench | 2 air pumps<br>200 gallons fish tank<br>Large Gravel Bag<br>2 x standard thermostats | 5 x Carpenters<br>1 x Installation Manager<br>1 x Aquatics installer<br>1 x Plumber<br>3 x Labourers | 1st June 2005 – 1st June 2011 |
| **767** | Marine | Eaglestone Castle, Eaglestone, Kent | Perry Vanderrune | 2 x 200 gallons fish tanks<br>500 Wood panels | 10 x Carpenters<br>2 x Installation Manager<br>1 x Aquatics installer<br>1 x Plumber<br>3 x Labourers | From 30th June 2012 |
| **943** | Marine | 23 Sackville Street, Wilts. W55 | Eric Mackintosh | 2 air pumps<br>200 gallons fish tank<br>1 x standard thermostat | No staff required | |
| **157** | Freshwater Tropical | Humbertson Castle, Kent, K8 | Perry Vanderrune | 2 air pumps<br>400 gallons fish tank<br>3 x standard thermostat | 1 x Aquatics installer | 1st September 2005 – 1st September 2012 |

## Activity 1

1.1. Identify the user and system requirements to design a database for the above scenario and design a relational database system using conceptual design (ER Model) by including identifiers (primary Key) of entities and cardinalities, participations of relationships. Convert the ER Model into logical database design using relational database model including primary keys foreign keys and referential Integrities. It should contain at least five interrelated tables. Check whether the provided logical design is normalised. If not, normalize the database by removing the anomalies.

***(Note:-It is allowed to have your own assumptions and related attributes within the scope of the case study given)***

**1.2.** Design set of simple interfaces to input and output for the above scenario using Wireframe or any interface-designing tool. Evaluate the effectiveness of the given design (ERD and Logical design) in terms of the identified user and system requirements .

## Activity 2

**Activity 2.1**

a. Develop a relational database system according to the ER diagram you have created (Use SQL DDL statements). Provide evidence of the use of a suitable IDE to create a simple interface to insert, update and delete data in the database. Implement proper security mechanisms in the developed database.

Evaluate the database solution developed and its effectiveness with relevant to the user and system requirements identified, system security mechanisms (EX: -User groups, access permissions) and the maintenance of the database.

**Activity 2.2**

a. Explain the usage of DML with below mentioned queries by giving at least one single example per each case from the developed database. Assess the usage of the below SQL statements with the examples from the developed database to prove that the data extracted through them are meaningful and relevant to the given scenario.

*Select/ Where / Update / Between / In / Group by / Order by / Having*

## Activity 3

### Activity 3.1

Provide a suitable test plan to test the system against user and system requirements. provide relevant test cases for the database you have implemented. Assess how the selected test data can be used to improve the effectiveness of testing.

***Note:- Learner needs to give expected results in a tabular format and screenshots of the actual results with the conclusion***

### Activity 3.2

Get independent feedback on your database solution from the non-technical users and some developers (use surveys, questioners, interviews or any other feedback collecting method) and make recommendations and suggestions for improvements in a separate conclusion/recommendations section.

## Activity 4

Produce a technical documentation and a user guide for the developed database system. Suitable diagrams diagrams (Use case diagram, class diagram, flow charts, DFD level 0 and 1) should be included in the technical documentation to show data movement in the system. Assess the developed database by suggesting future enhancements to ensure the effectiveness of the system.

## Acknowledgement

At last author would like to share the experience while doing the project. Author learns many new things about the networking topics. The best thing which author can share is that author developed more interest in this subject. This Project gave author a real sight into the Networking world.

A very special thanks to Ms Gayani Nisansala who teach us this subject and Author thanks for who helped author to do this kind of project. Thank you!

Regards,
The author,
Ranudi Kariyapperuma.

# TABLE OF CONTENT

Table of Figures

**Design a relational database system containing at least four interrelated tables, with clear statements of user and system requirements.**

The user's main purpose is to make a design for Polly pipe company. It is a water sport Provider and installer company. However, the company has only paper requirements. User must convert to system requirement, Entity Relationship Diagram and a Normalized Database. The Database that made for Polly Pipe company was built by Microsoft SQL Platform

The author designs the database and the entities and its attributes are below in this grid,

| Entity | Attributes |
|---|---|
| Staff | • Staff id |
| | • Staff Type |
| | • Staff Name |
| | • Staff Tel No |
| Customer | • Customer id |
| | • Customer Name |
| | • Customer Address |
| | • Customer Tel No |
| Equipment | • Equipment id |
| | • Equipment Type |
| | • Equipment Name |
| | • No of Equipment |
| Installation Management | • Installation Id |
| | • Installation Type |
| | • No of Equipment |
| | • No of Staff Member |
| | • Customer Id |
| | • Staff Id |
| | • Equipment Id |
| | • Project Start Date |
| | • Project End Date |

**Statements of User and System Requirements.**

## 1.1 User Requirements.

- The User Requirements Specification describes the business needs for what users require from the system. User Requirements Specifications are written early in the validation process, typically before the system is created.

  (Anon., n.d.)

- User Requirements are written by owner and End users.
- For an example User Requirements are,

  - ★ Secured Data
  - ★ Updated Data
  - ★ Record all the details of Staff, Customer, Equipment and Installation Management.

- Narrative text is typically used to describe user requirements in a User Requirements Document (URD).

- Finding out what a user wants a software product to achieve is a crucial and challenging phase in its creation.

## 1.2 System Requirements

- System requirements are a broad and also narrow detailed statement that the customer makes in order to achieve their requirements. The statement should clearly explain what the customer exactly wants and how they want it. A customer's need might be to satisfy a contract, solve a problem, achieve an objective, meet a standard, or to meet any other guidelines of the project.

  (Siedle, 2015)

- System requirements have Software requirements and Hardware requirements.

- In system requirement if there a packaged product users will be packing the product and If there has a downloadable product users will make a download page to it.

- In many times Hardware requirements are such as,

    - ★ Processor type
    - ★ Memory Type
    - ★ Operating system versions

**Entity Relationship Diagram (ER Diagram) For Polly Pipe Company**

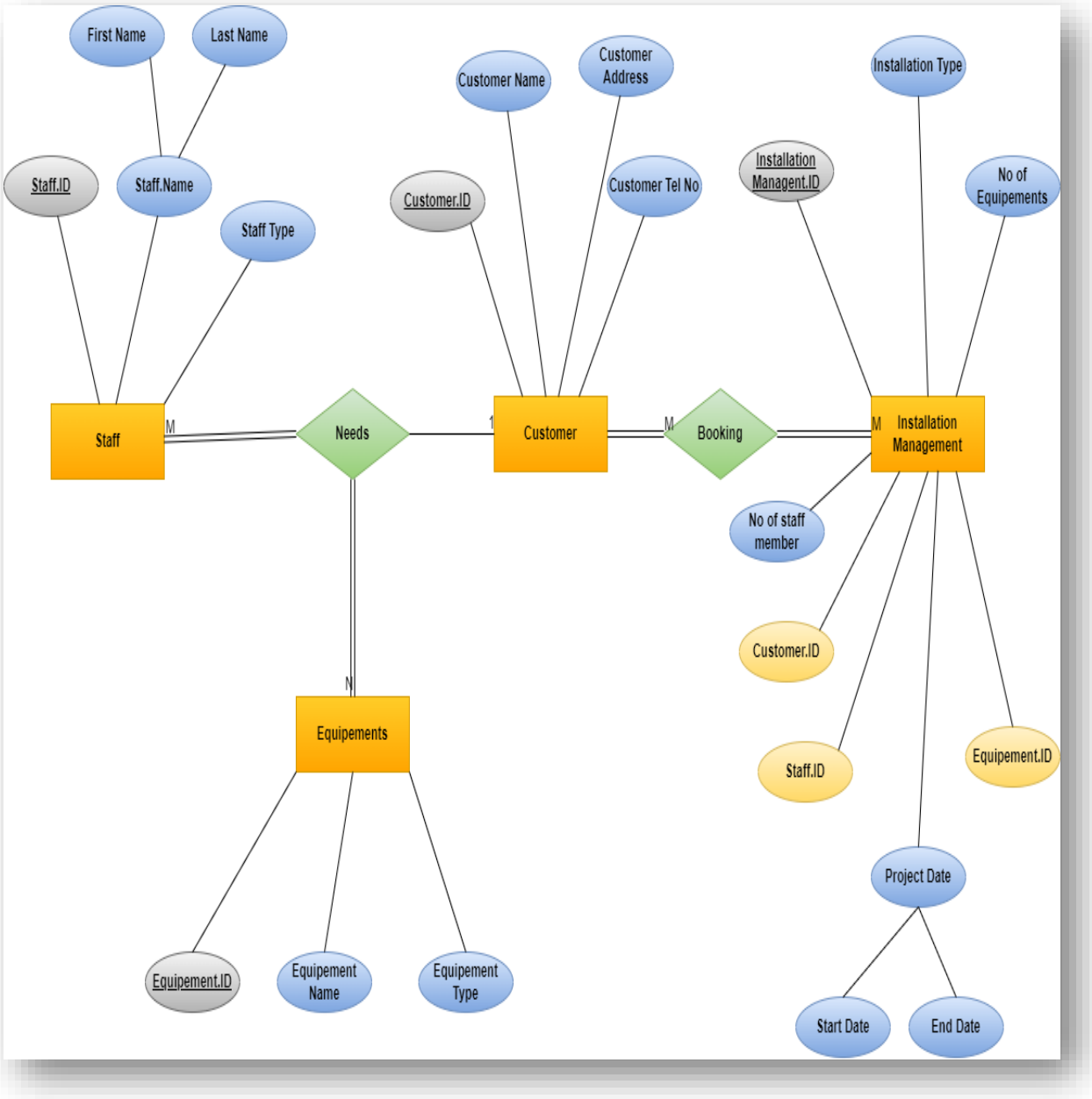



Figure 1 : 1. Entity Relationship Diagram (ER Diagram) For Polly Pipe Company

**Relational database system for Polly Pipe Company.**

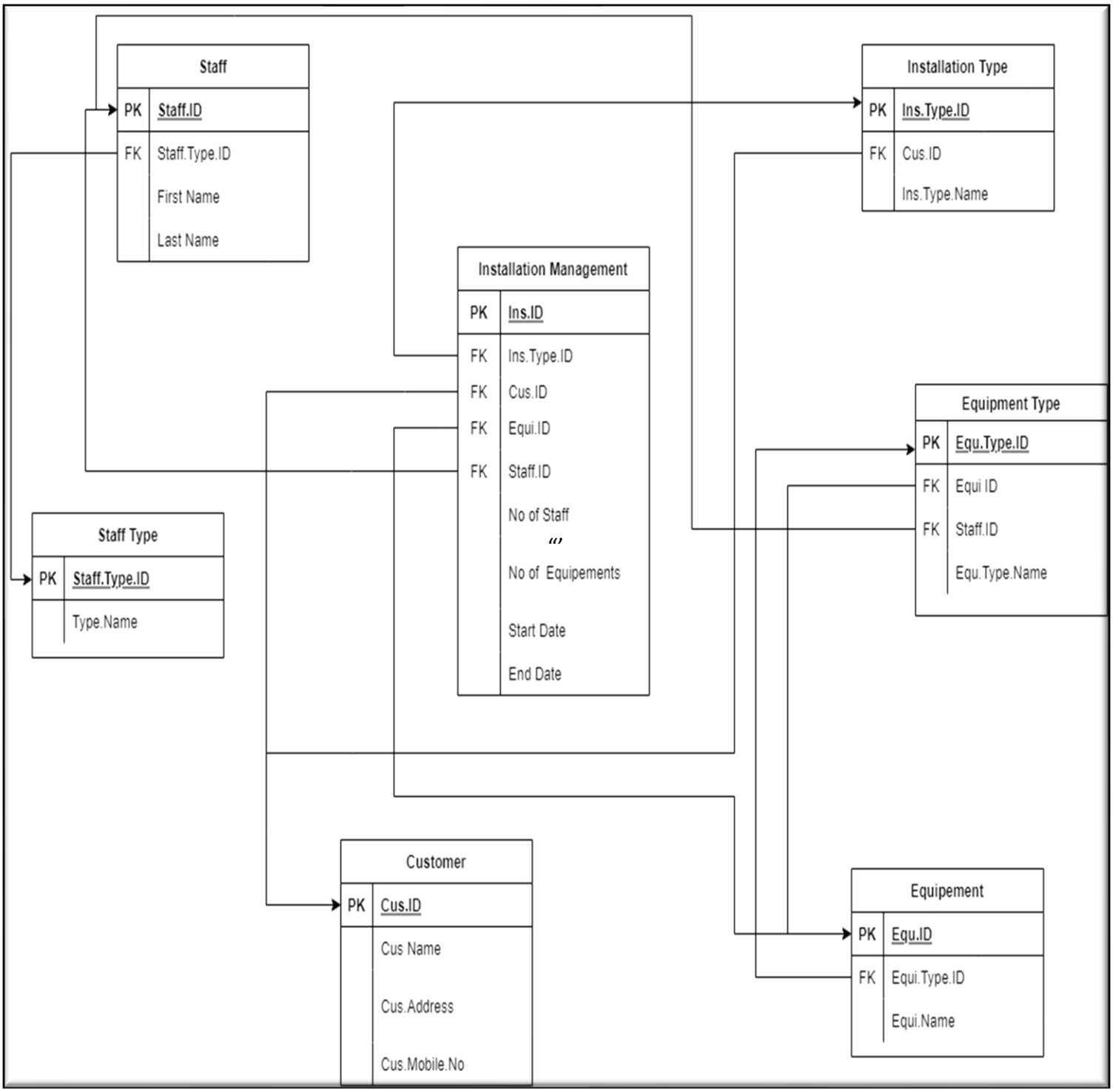



**Figure 2 :** Relational database system for Polly Pipe Company.

**Produce a comprehensive design for a fully functional system that includes interface and output designs, data validations and data normalization.**

## Normalization

In the context of data and databases, normalization is the process of arranging and structuring data in a way that reduces data abnormalities and redundancies. It is a key idea in relational database design and enhances data consistency, efficiency, and integrity. A huge table with complicated data is normalized by splitting it up into smaller, more manageable tables and connecting them through relationships. Usually, a set of formal guidelines called as normalizing forms are used to do this.

## Database Normalization Forms

- **First Normal Form (1NF):**

Assures that each table row has a distinct identifier (primary key) and that each table cell contains a single, atomic value (no repeated groups or arrays).

- **Second Normal Form (2NF)**

This form expands on the first normal form by requiring that all of the table's non-key attributes be entirely reliant on the complete primary key. Partial dependencies are removed as a result.

- **Third Normal Form(3NF)**

Builds on the second normal form and gets rid of transitive dependencies. It makes sure that non-key attributes only depend on the primary key and not on other non-key attributes.

**Set of Input and Output interfaces of Polly Pipe Company.**

1. **Login Form**

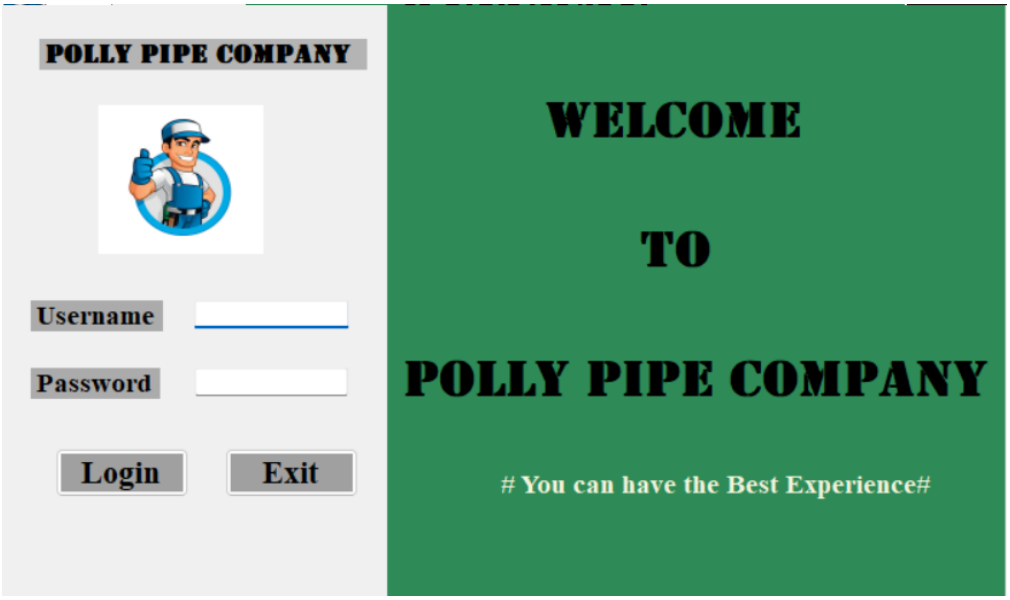



Figure 3 : Login Form

**2.Main Form**

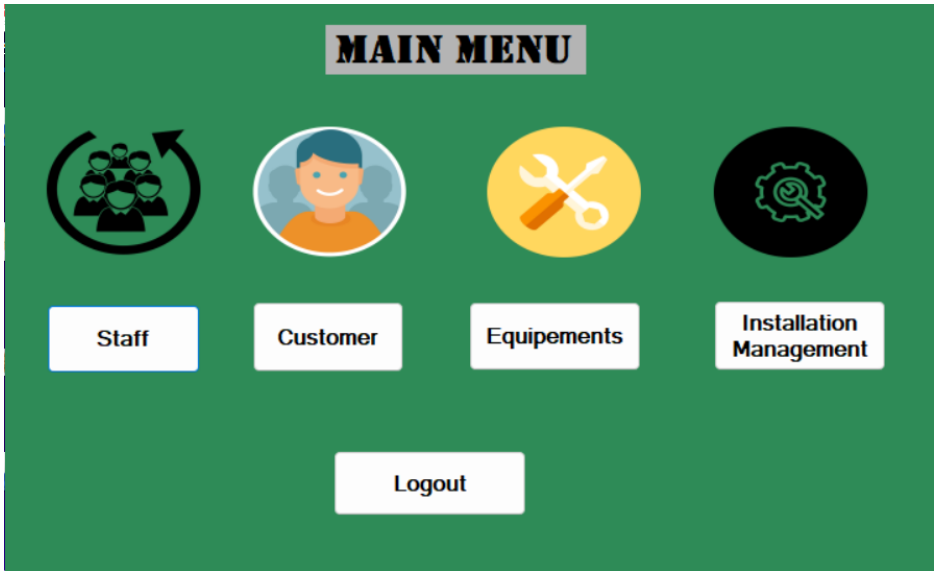



Figure 4:Main Form

### 3. Staff Form

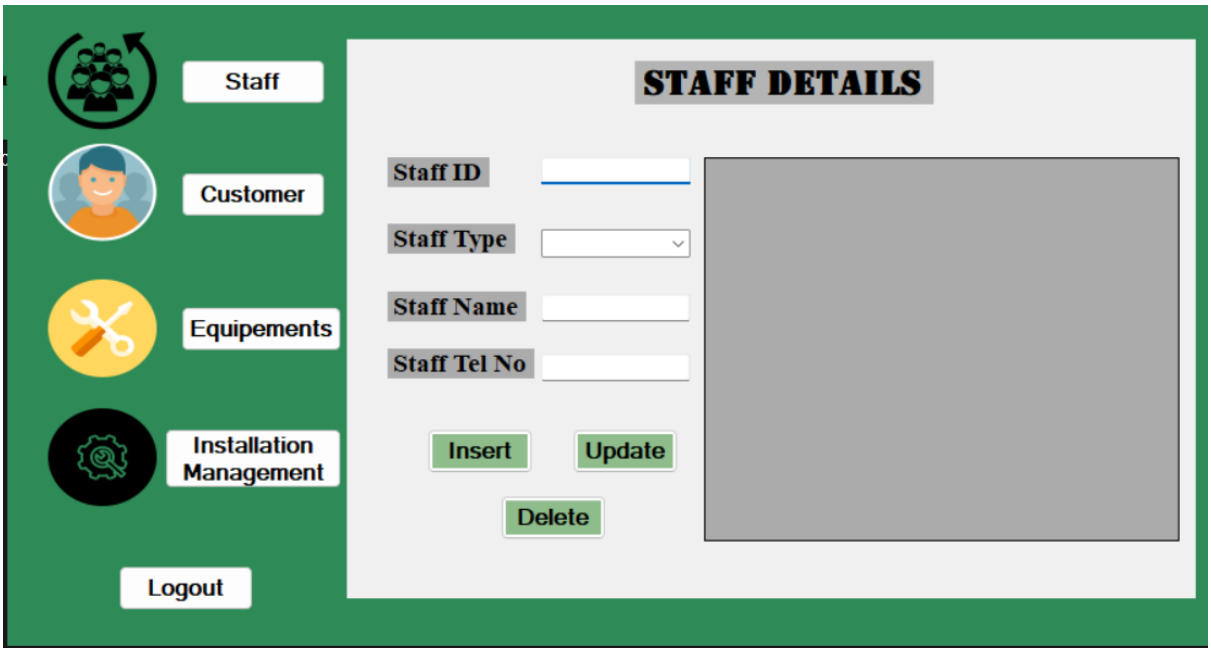



Figure 5: STAFF FORM

### 4.Customer Form

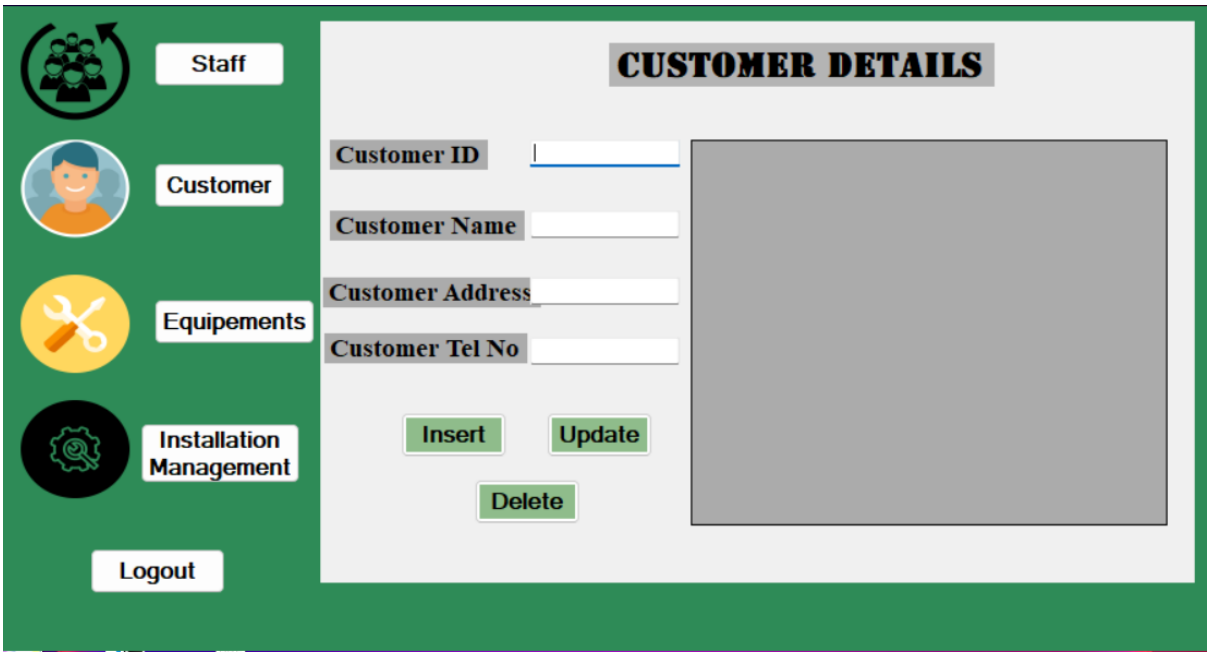



Figure 6 : Customer Form

## 5. Equipment Form

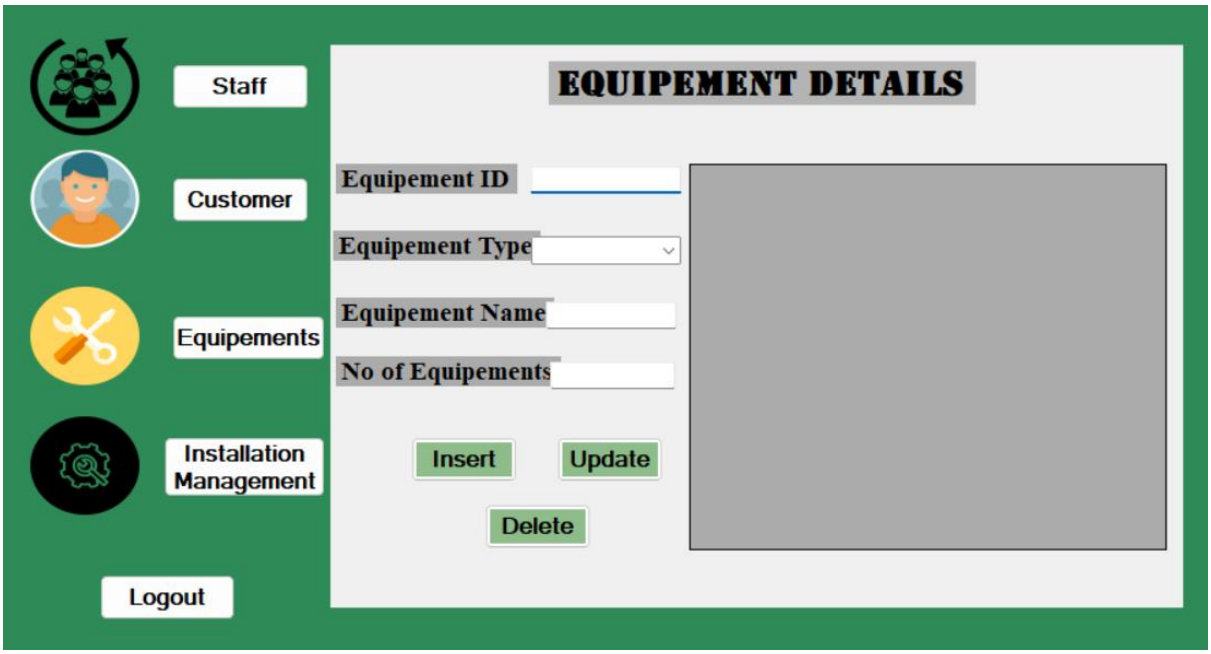



Figure 7 : Equipment Form

## 6. Installation Management Form

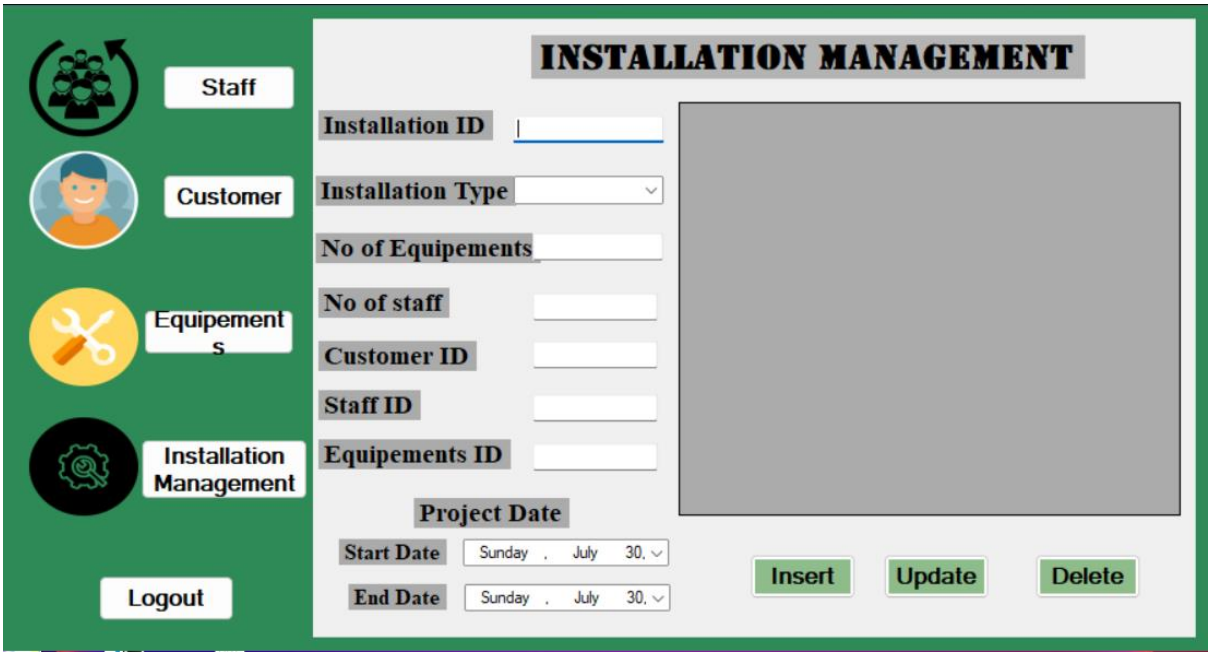



Figure 8 : Installation Management Form

**The database system with evidence of user interface, output, and data validations, and querying across multiple tables.**

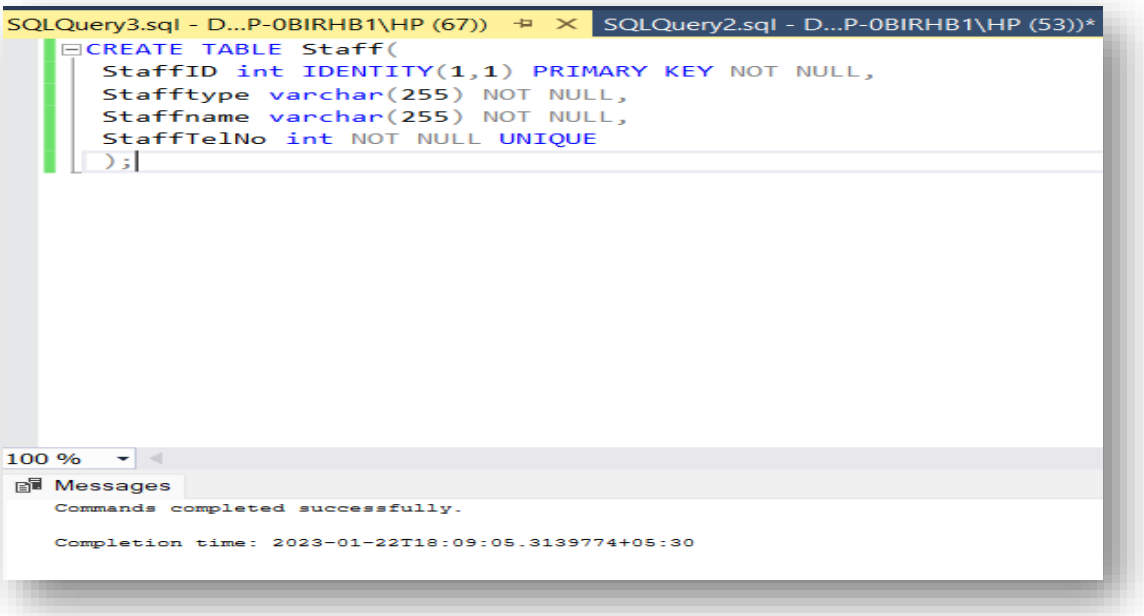
## 1 .Creating tables with SQL DDL

DDL means Data Definition Language and it is a subset of SQL . For example, DDL commands can be used to add new tables or objects, complete with all of their attributes, to the database (data type, table name, etc.). Author also mentioning the CREATE, ALTER, DROP, and TRUNCATE are used commonly in sql querying.

However the Author Creates a Database with the help of Er diagram and DDL statement are below in this with the interfaces.

```
CREATE DATABASE PollypipedDb;


CREATE TABLE Staff(
 StaffID int IDENTITY(1,1) PRIMARY KEY NOT NULL,
 Stafftype varchar(255) NOT NULL,
 Staffname varchar(255) NOT NULL,
 StaffTelNo int NOT NULL UNIQUE
 );
```



CREATE TABLE Customer(

CustomerID int IDENTITY(1,1) PRIMARY KEY NOT NULL,

Customername varchar(255) NOT NULL,

CustomerAddress varchar(255) NOT NULL,

CustomerTelNo int NOT NULL UNIQUE

);

```
CREATE TABLE Customer(
  CustomerID int IDENTITY(1,1) PRIMARY KEY NOT NULL,
  Customername varchar(255) NOT NULL,
  CustomerAddress varchar(255) NOT NULL,
  CustomerTelNo int NOT NULL UNIQUE
  );
```

CREATE TABLE Equipement(

EquipementID int IDENTITY(1,1) PRIMARY KEY NOT NULL,

Equipementtype varchar(255) NOT NULL,

Equipementname varchar(255) NOT NULL,

NoofEquipement int NOT NULL UNIQUE

);

```
CREATE TABLE Equipement(
EquipementID int IDENTITY(1,1) PRIMARY KEY NOT NULL,
Equipementtype varchar(255) NOT NULL,
Equipementname varchar(255) NOT NULL,
NoofEquipement int NOT NULL UNIQUE
);
```

CREATE TABLE InstallationManagement(

InstallationID int IDENTITY(1,1) PRIMARY KEY NOT NULL,

Installationtype varchar(255) NOT NULL,

NoofEquipements int NOT NULL,

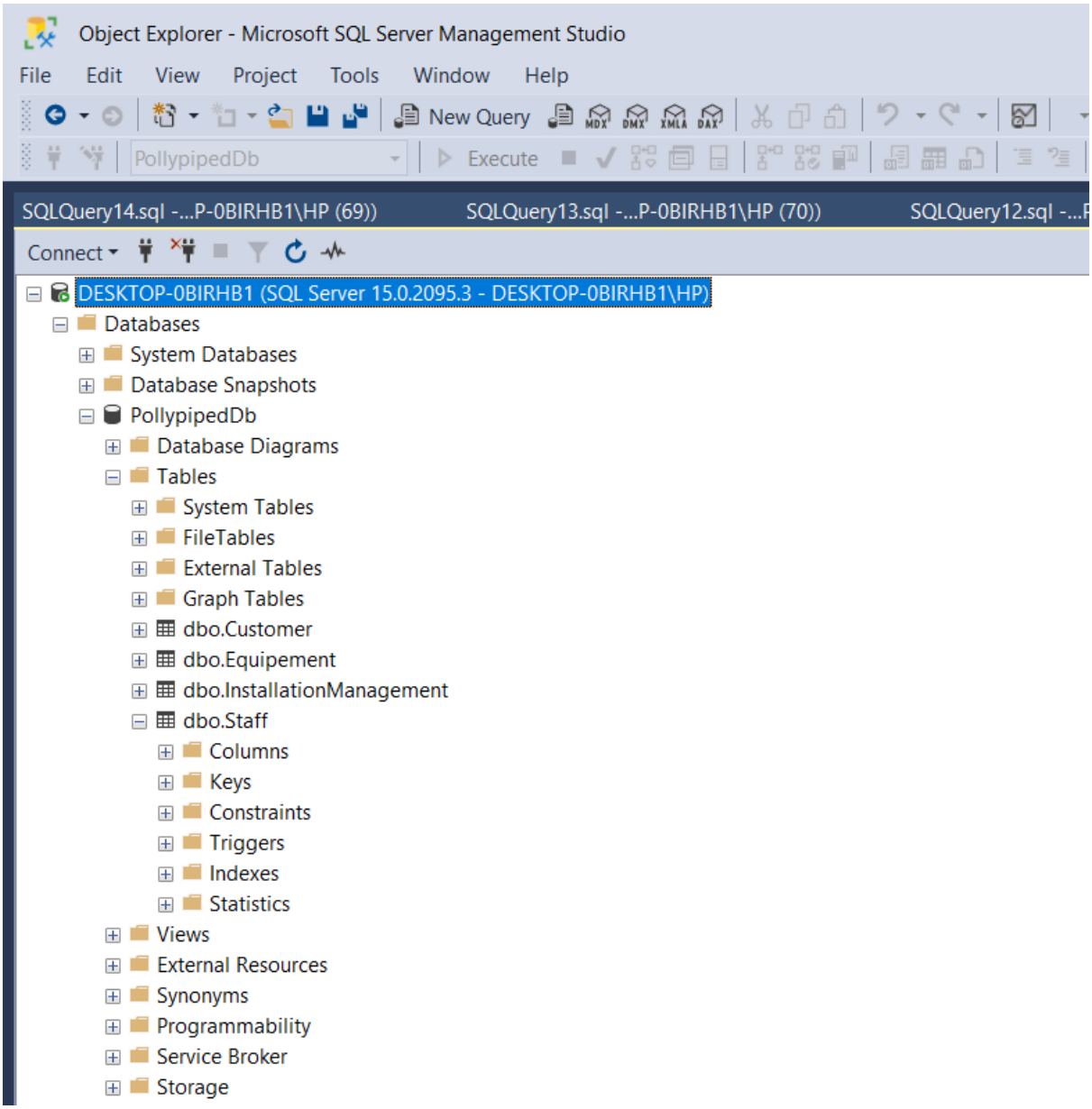
Noofstaff int NOT NULL,

CustomerID int NOT NULL,

StaffID int NOT NULL,
EquipementID int NOT NULL,
StartDate date NOT NULL,
EndDate date NOT NULL
);

```
);
CREATE TABLE Equipement(
EquipementID int IDENTITY(1,1) PRIMARY KEY NOT NULL,
Equipementtype varchar(255) NOT NULL,
Equipementname varchar(255) NOT NULL,
NoofEquipement int NOT NULL UNIQUE
);
```

**Output of these tables**



**A fully functional database system that includes system security and database maintenance.**

## The database solution and how to Develop

Database is a Structure that can stored data in a software. Database development is a complex process that can analyse goals and organized data.

There are 4 steps that follow by a database developer that are,

- Understand business requirements.
- Conceptual Modelling
- Logical Modelling
- Physical Modelling

### Maintenance of Database

Database Maintenance is a term we use to describe a set of tasks that are all run with the intention to improve your database. There are routines meant to help performance, free up disk space, check for data errors, check for hardware faults, update internal statistics, and many other obscure (but important) things.

(Support, n.d.)

There are four primary "Categories" of routines in the database maintenance program. Such as,

- Index Defragmentation
- Log File Maintenance
- File Data Complication
- Integrity Check

There also have benefits of Maintained of Database.

- Keeps Companies up to date
- Promote Efficient database.
- Saves time.
- Protect against Threats

(Anon., n.d.)

**Security of Database**

Database security refers to the policies and procedures put in place to guard against unauthorized access, abuse, damage, and theft of computer networks, systems, and data. In order to protect the privacy, integrity, and accessibility of data and resources stored on a computer system, a combination of hardware, software, processes, and policies is used.

**A query language into the relational database system**

## 1.The usage of DML

The meaning of DML is Data Manipulation Language. DML is used to Manipulate data. The basic manipulation used in DML that include adding to Database, Update records, Move data from one position to another and deleting records successfully.

The List of Commands and examples are given below.

- **Insert**

This command is used for insert data into table.

```
INSERT INTO Equipement(Equipementtype, Equipementname,NoofEquipement) VALUES
('Tanks' , '100 gallon tank' ,20) ,
('Thermostats' , 'Standard' , 30 ),
('Air Pumps' , 'Super' , 10 ),
('Filters' , 'Air driven' , 50 );
```

```
100 %
Messages

(4 rows affected)

Completion time: 2023-01-22T19:35:16.1782961+05:30
```

Figure 9 :Insert

Insert Command Output

100 %

Results | Messages

| | EquipementID | Equipementtype | Equipementname | NoofEquipement |
|---|---|---|---|---|
| 1 | 1 | Tanks | 100 gallon tank | 20 |
| 2 | 2 | Thermostats | Standard | 30 |
| 3 | 3 | Air Pumps | Super | 10 |
| 4 | 4 | Filters | Air driven | 50 |

Figure 10 : Insert Command Output

- **Update**

This command is used for update the relevant table.

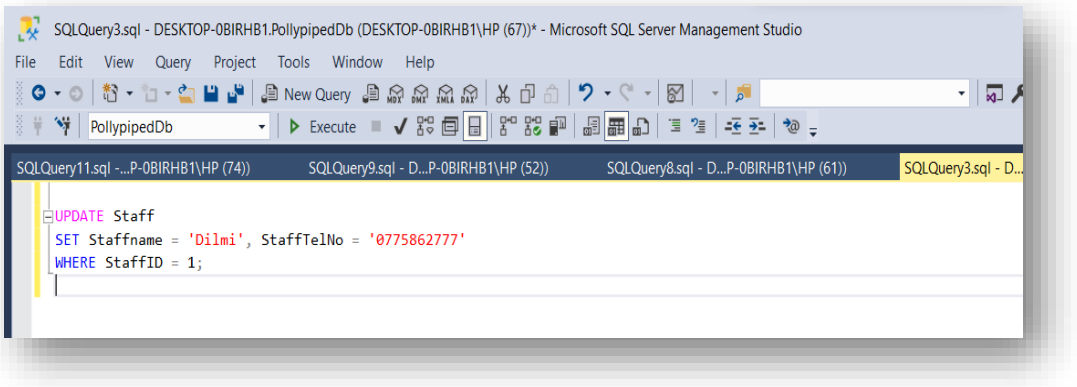



Figure 11 : Update

Update command Output

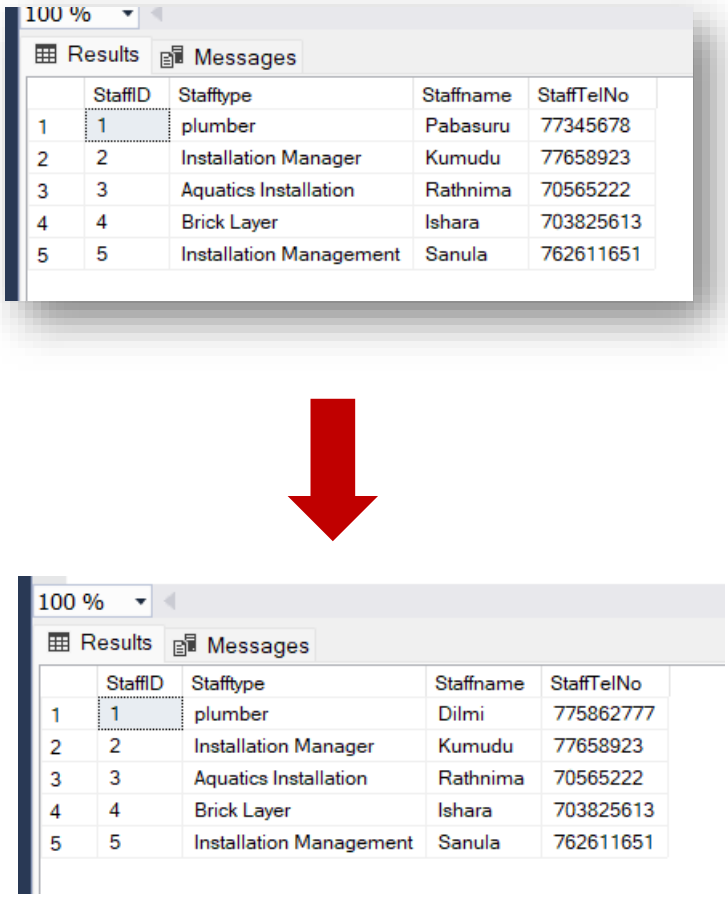



Figure 12 : Update command output

- **Delete**

This command is used for delete data in relevant table .

```
DELETE FROM Customer WHERE CustomerName='Hilmy';
```

Figure 13 : Delete

Delete command Output

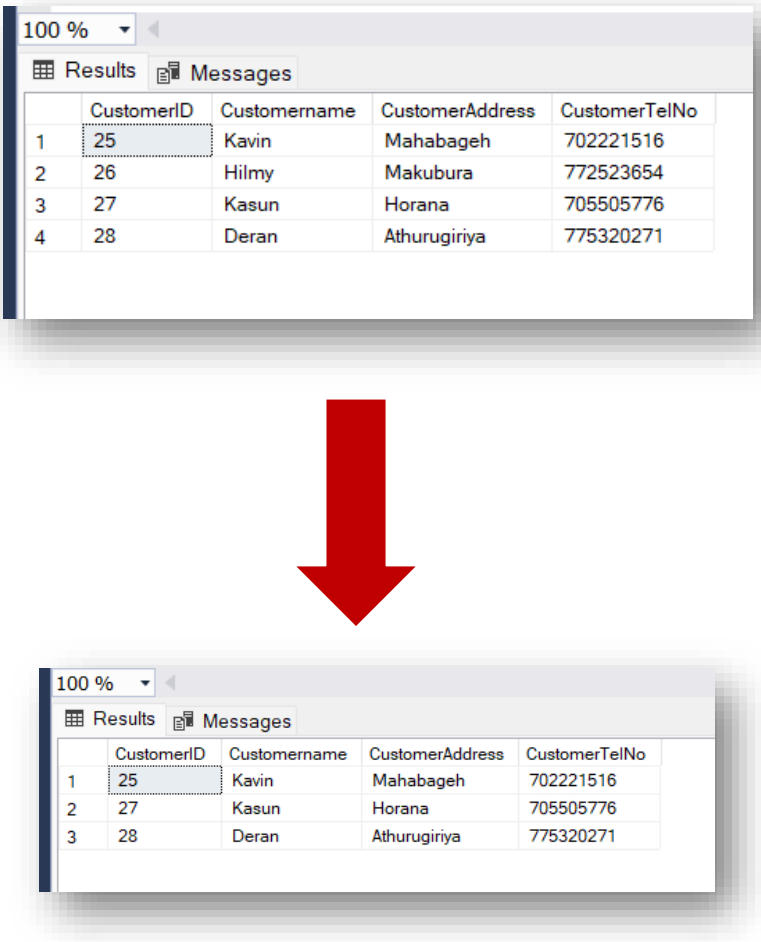



Figure 14 : Delete command Output.

- **Select**

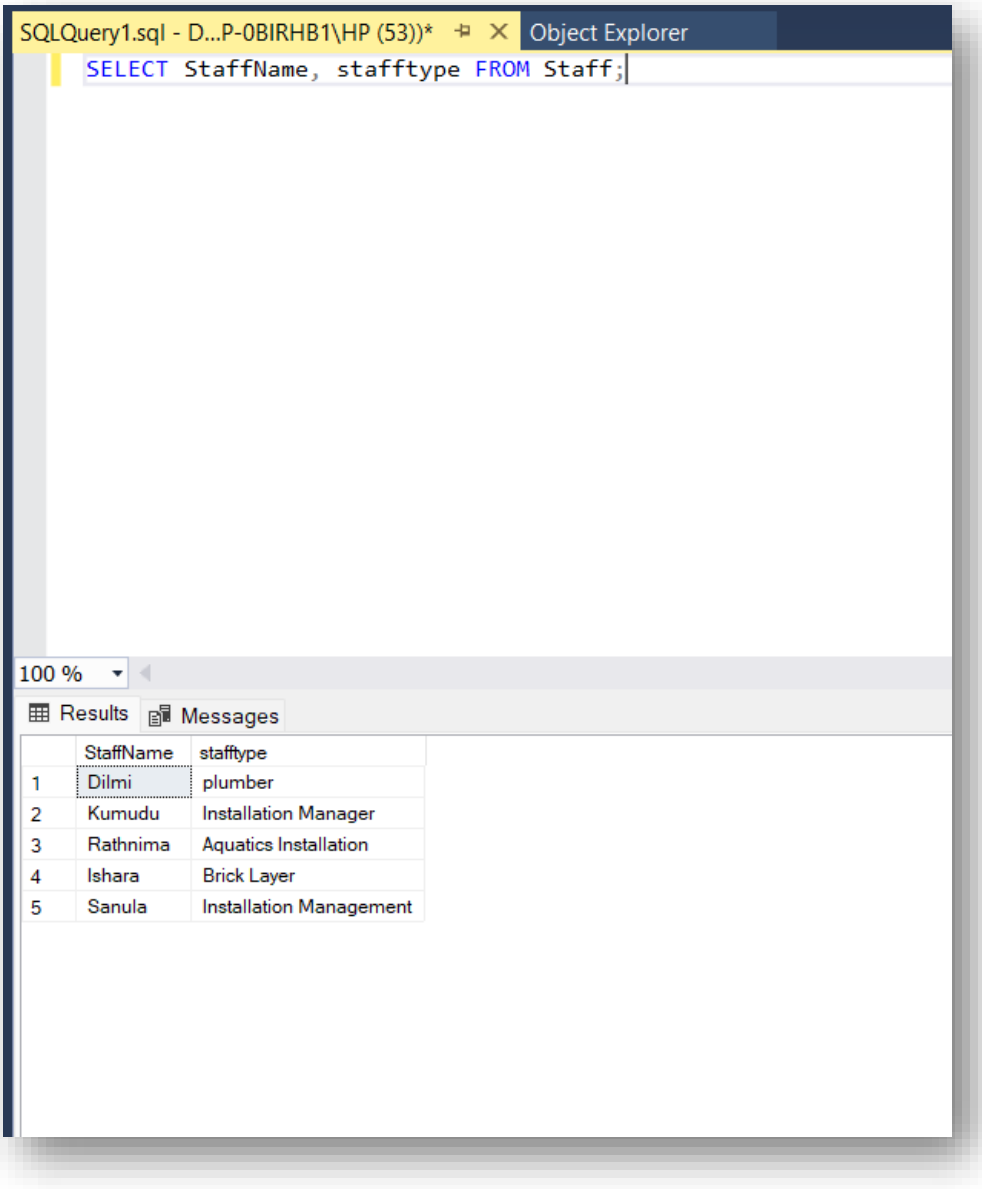



Figure 15 :Select

**Meaningful data has been extracted using query tools to produce appropriate management information.**

- **Where**

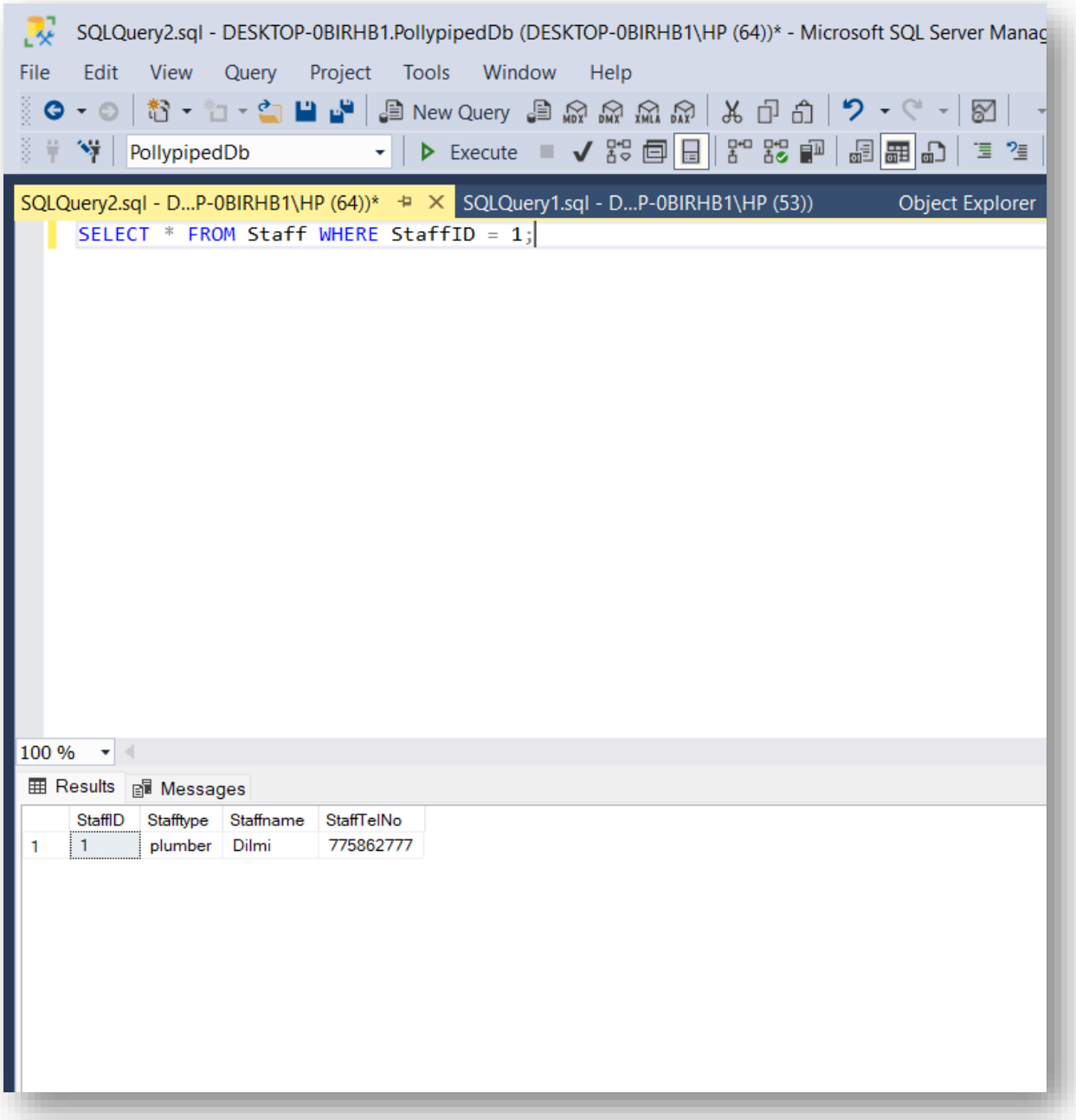



Figure 16 : Where

- **Between**

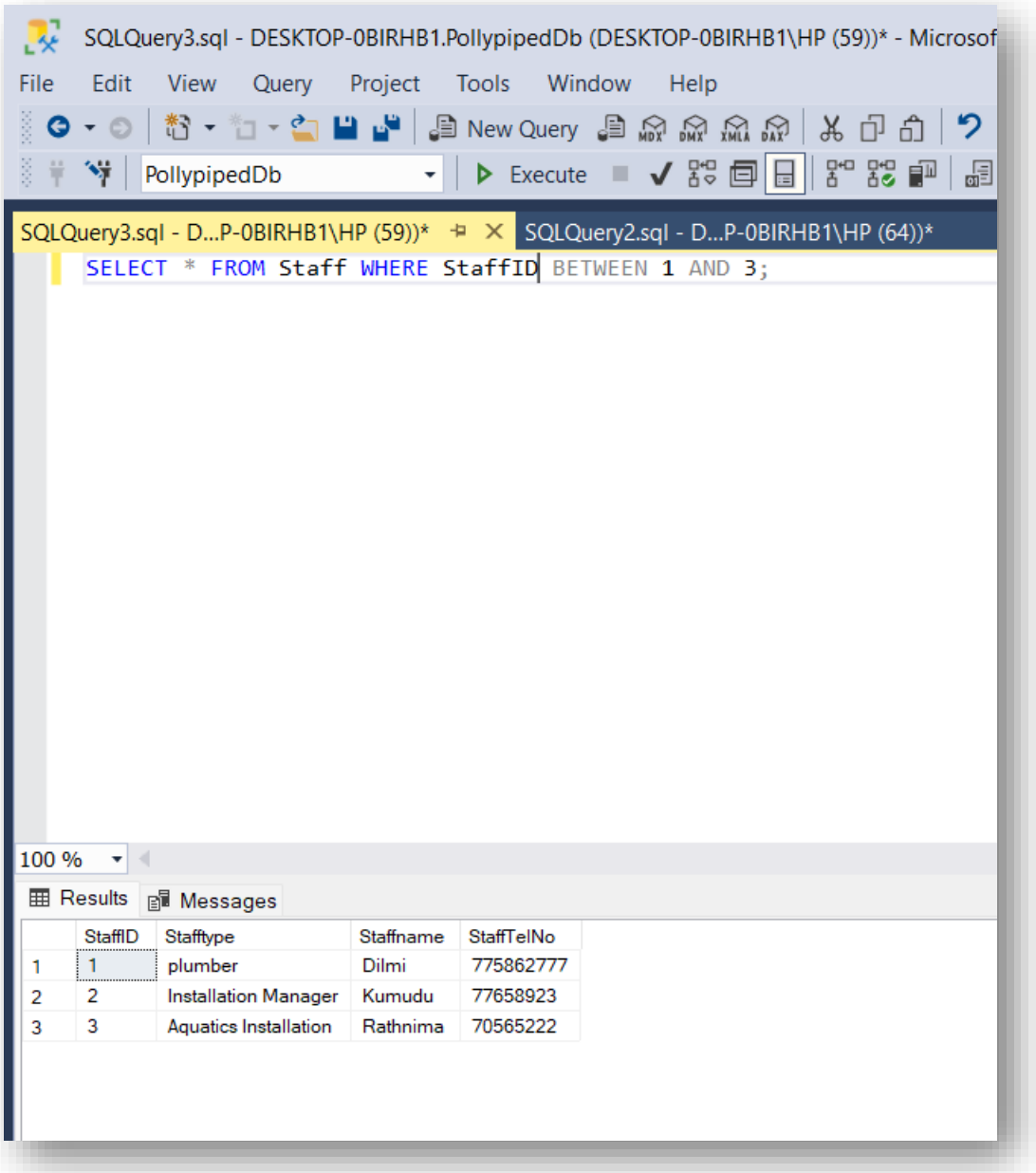



Figure 17: Between

- **In**

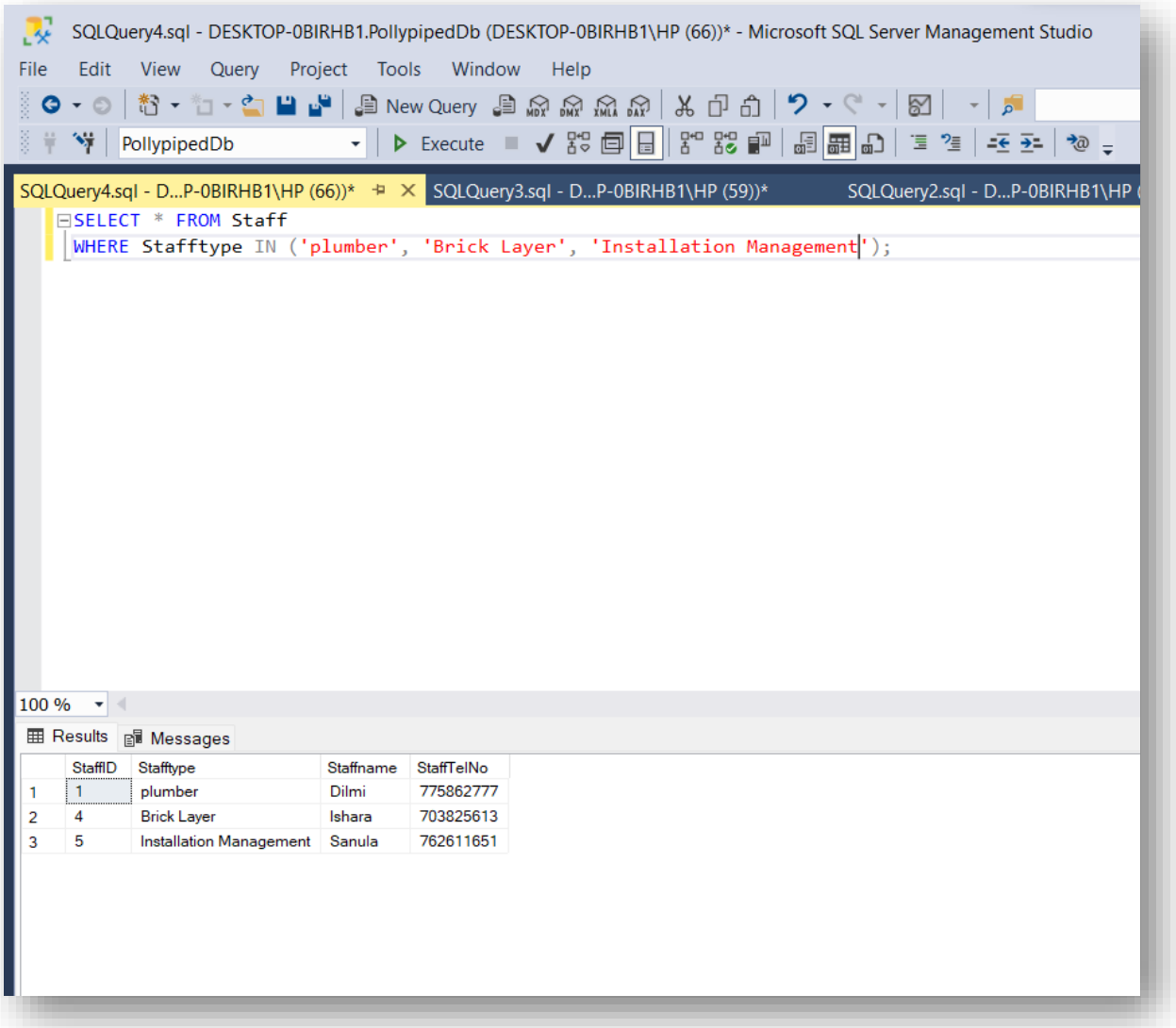



Figure 18 : In

- **Group by**

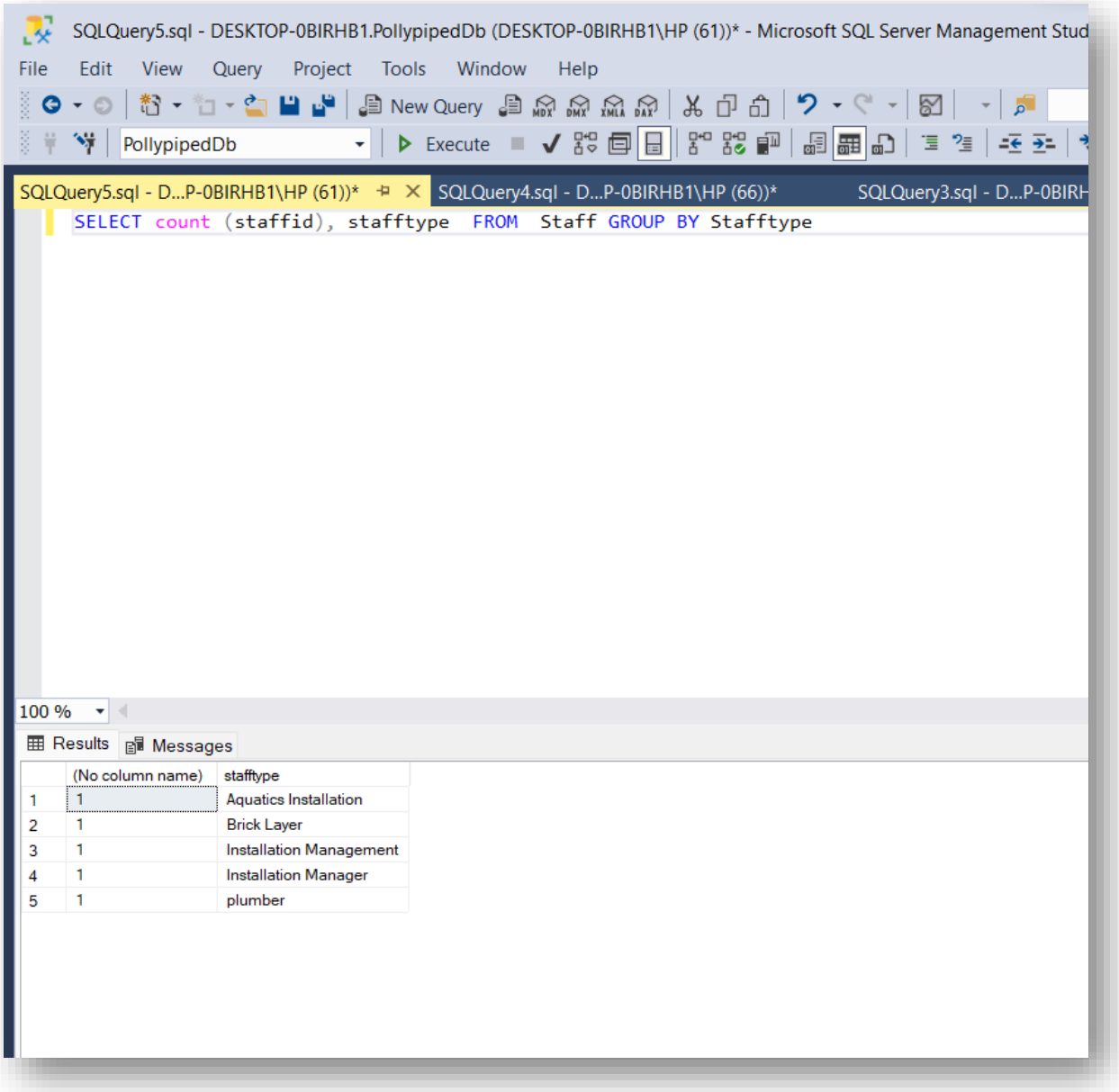



Figure 19 : Group by

- **Order by**

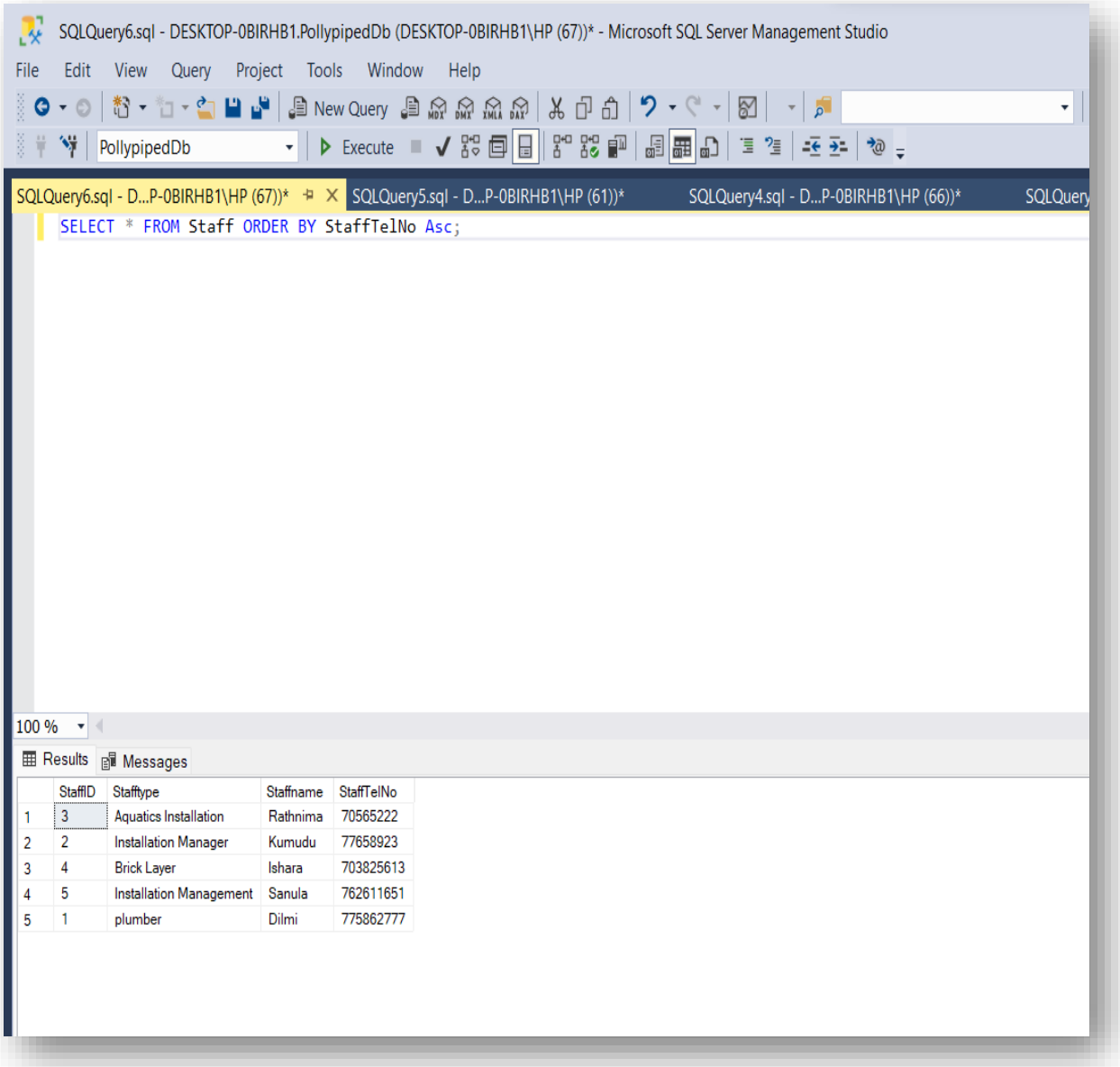



Figure 20 : Order by

- **Having**

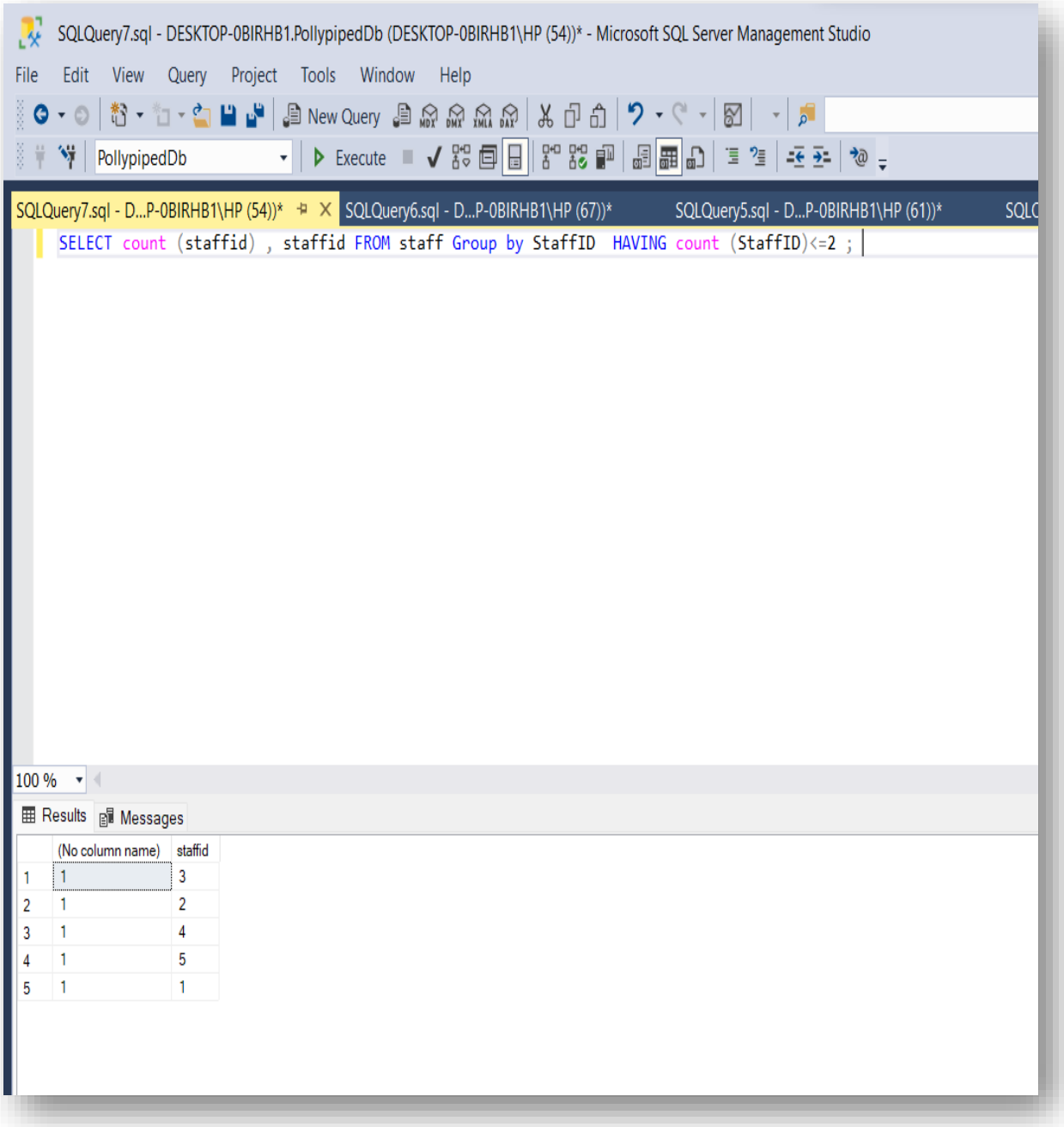



Figure 21 : Having

**The system against user and system requirements.**

## Test Plan

A test plan is a comprehensive document that lists all the resources, test objectives, timetable, estimations, and techniques that will be used to finish the project. Consider it a guide for how test managers should execute the tests necessary to make sure the product is functioning properly.

### Importance of Test Plan

- Aid folks outside the QA teams (developers, business managers, customer-facing teams) in comprehending precisely how the website or app will be tested.
- Provide QA engineers with a detailed manual on how to carry out their testing tasks.
- Go into depth about things like test scope, estimate, approach, etc.
- It is simpler for management staff to evaluate and utilize this data when it is compiled into a single document.

**Login Test 01**

| Test Case ID | | Login-01 | | | |
|---|---|---|---|---|---|
| **Test Case Description** | | Login – Positive Test Case | | | |
| **Tester Name** | | Ranudi Gayathmie Kariyapperuma | | | |
| **No** | **Action** | **Inputs** | **Expected Output** | **Actual Output** | **Test Result (Success/fail)** |
| 01 | Enter Correct Username and Password | Username – pollypipe123<br><br>Password – pollypipe1234 | Login Success! | Login Success! | Success |

Test 01 example

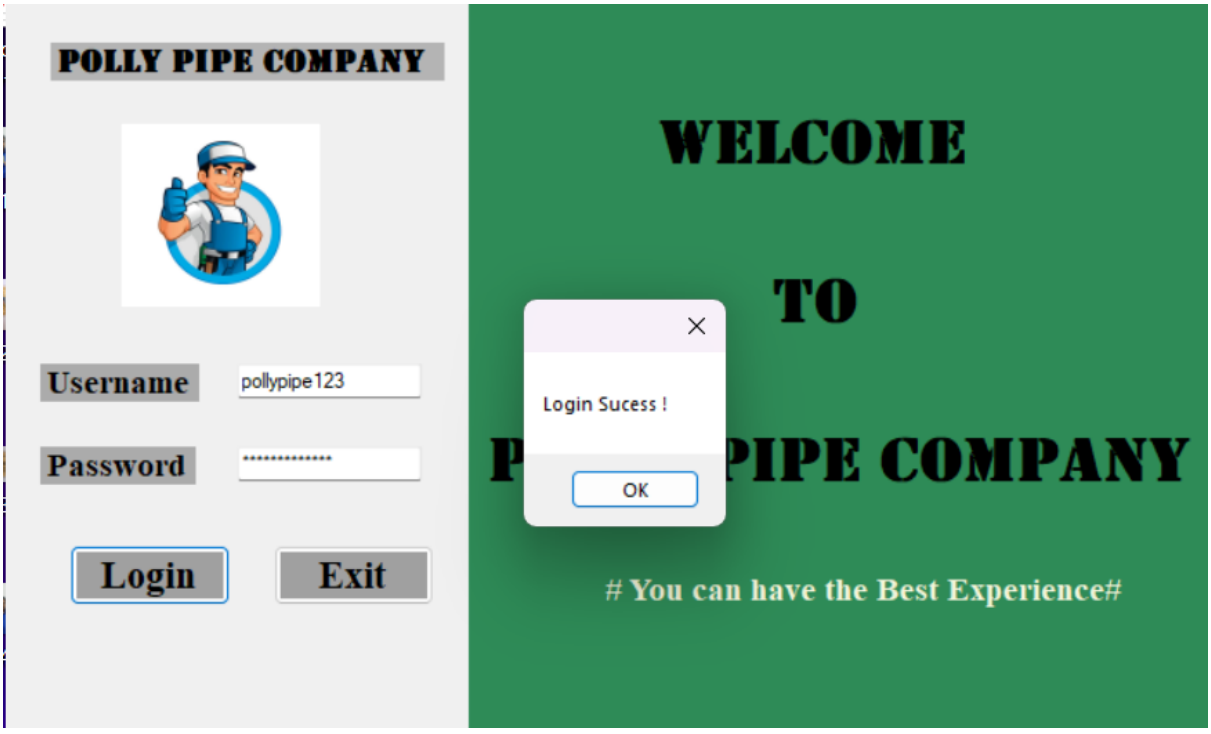



Figure 22 : Test form 1

**Login Test 02**

| Test Case ID | | Login-02 | | | |
|---|---|---|---|---|---|
| **Test Case Description** | | Login – Negative Test Case | | | |
| **Tester Name** | | Denethmi Aththanayaka | | | |
| **No** | **Action** | **Inputs** | **Expected Output** | **Actual Output** | **Test Result (Success/fail)** |
| 02 | Enter Incorrect Username and Correct Password | Username – admin123<br><br>Password – pollypipe1234 | Login Fail! | Login Fail! | Sucess |

Test 02 example


Figure 23 : Test Form 2

**Login Test 03**

| Test Case ID | | Login-03 | | | |
|---|---|---|---|---|---|
| **Test Case Description** | | Login – Negative Test Case | | | |
| **Tester Name** | | Diyana Fernando | | | |
| **No** | **Action** | **Inputs** | **Expected Output** | **Actual Output** | **Test Result (Success/fail)** |
| 03 | Enter Incorrect Username and Incorrect Password | Username – Admin123<br><br>Password – 2004 | Login Fail! | Login Fail! | Sucess |

Test 03 example

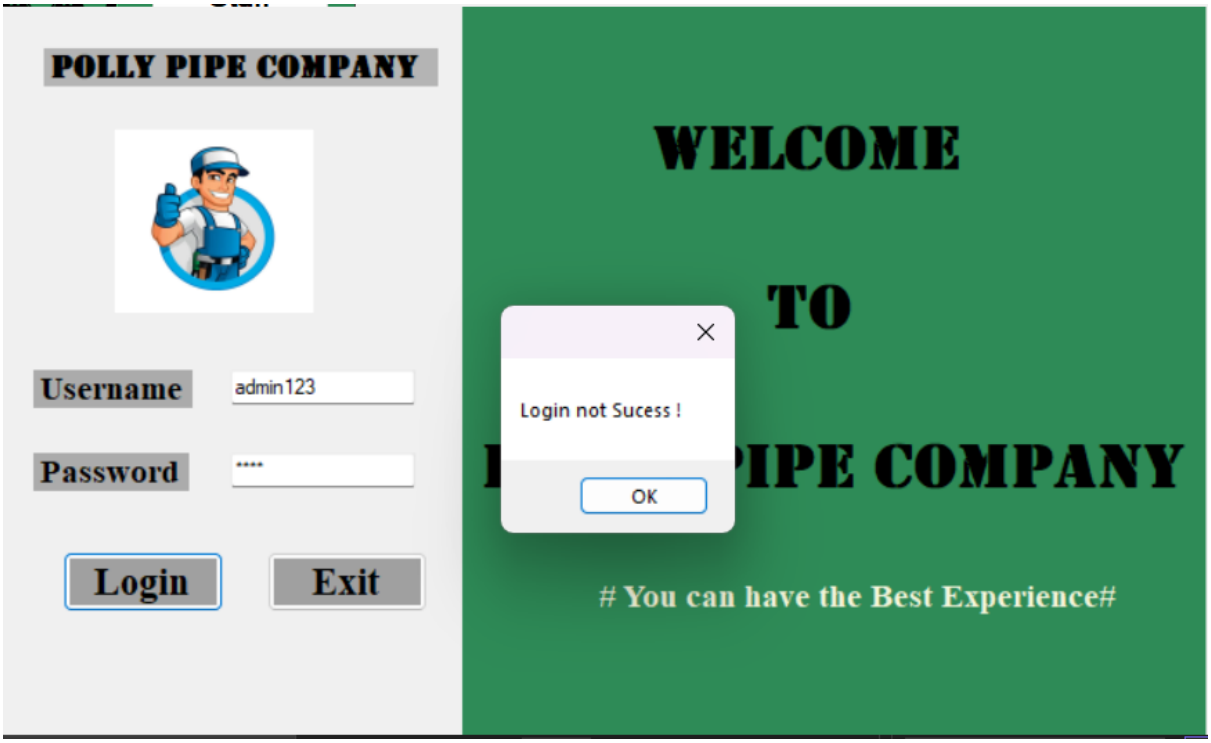



Figure 24 : Test Form 3

**Login Test 04**

| Test Case ID | | Login-03 | | | |
|---|---|---|---|---|---|
| **Test Case Description** | | Login – Negative Test Case | | | |
| **Tester Name** | | Sanula Kariyapperuma | | | |
| **No** | **Action** | **Inputs** | **Expected Output** | **Actual Output** | **Test Result (Success/fail)** |
| 03 | Enter correct Username and Incorrect Password | Username – pollypipe123<br><br>Password – 1234 | Login Fail! | Login Fail! | Sucess |

Test 04 example


Figure 25 : Test Form 4

**Insert Testing in Staff interface.**

Testers can check the database's functionality by inserting test data to add new records to the system. Benefits are that Testers can evaluate how the database handles multiple data formats, constraints, and validation rules by inputting various types of test data. This aids in identifying any problems with data management, primary key violations, or lack of data integrity.

Example -:

| **Test Case ID** | | Insert-01 | | | |
|---|---|---|---|---|---|
| **Test Case Description** | | Insert – Positive Test Case | | | |
| **Tester Name** | | Ishara dias | | | |
| **No** | **Action** | **Inputs** | **Expected Output** | **Actual Output** | **Test Result (Success/fail)** |
| 01 | Enter details of staff id, staff type, staff name, Staff Tel no | Staff id – 3<br>Staff Type – Plumber<br>Staff Name – Frash<br>Staff Tel No - 076543789 | Added successfully! | Added successfully! | Success |

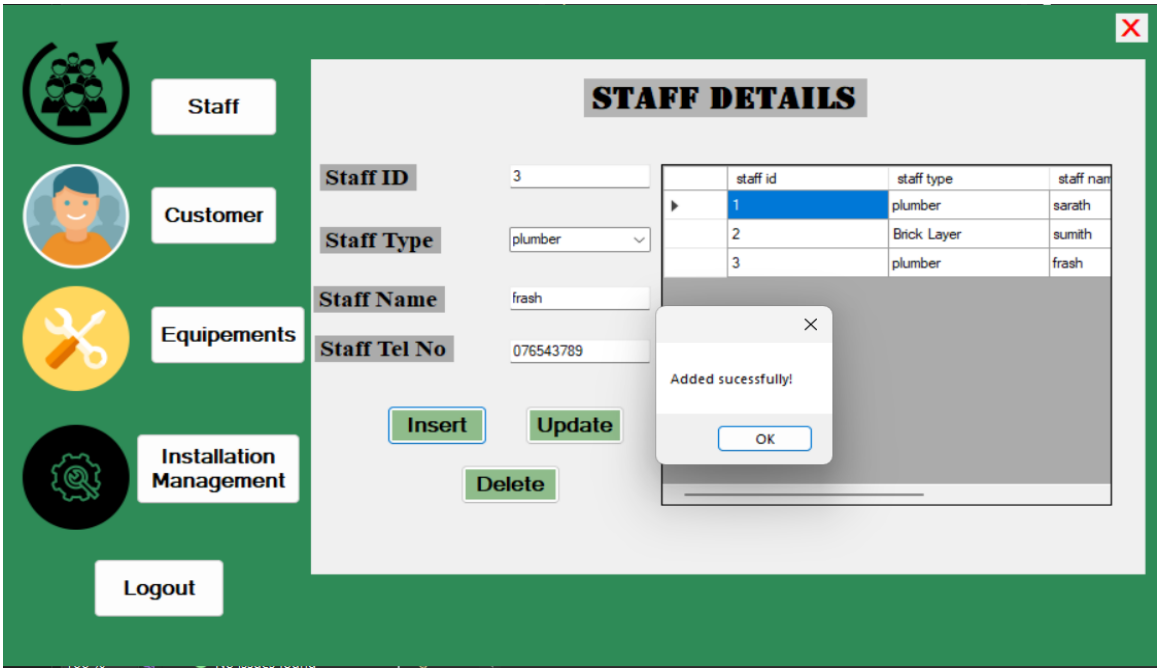



Figure 26 : Insert Output (Develop by author )

**Update Testing in Staff interface.**

A database's capacity to make changes to already-existing records and uphold data consistency is evaluated using the aid of updated test data. Benefits are that By running update operations, testers can confirm whether the database changes records correctly, initiates relevant actions, and enforces data restrictions. This check is essential to make sure that data modifications don't produce unexpected outcomes or data discrepancies.

Example -:

| **Test Case ID** | | Update-01 | | | |
|---|---|---|---|---|---|
| **Test Case Description** | | Update – Positive Test Case | | | |
| **Tester Name** | | Kumud Subash | | | |
| **No** | **Action** | **Inputs** | **Expected Output** | **Actual Output** | **Test Result (Success/fail)** |
| 01 | Update staff name as Ruwan in staff id 2 | Staff id – 2<br>Staff Type – Brick layer<br>Staff Name – Ruwan<br>Staff Tel No-<br>07768345213 | Update successfully! | Update successfully! | Success |

**Before update Interface**

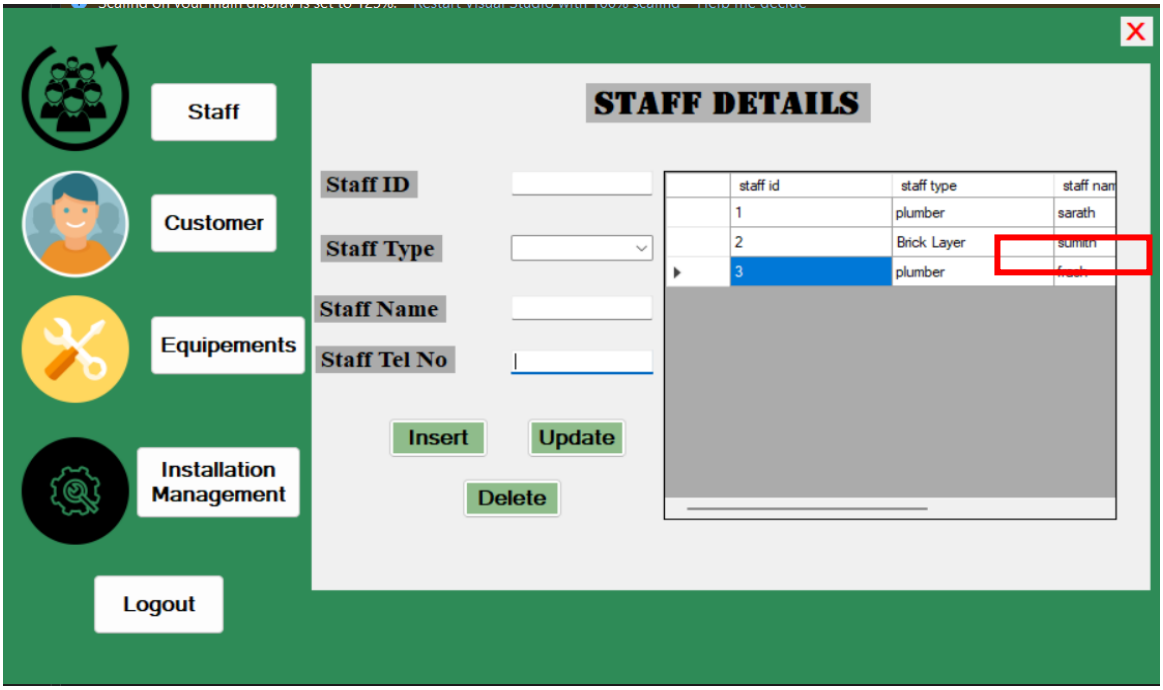



Figure 27 : Before update Interface (Made by Author)

**After update Interface**

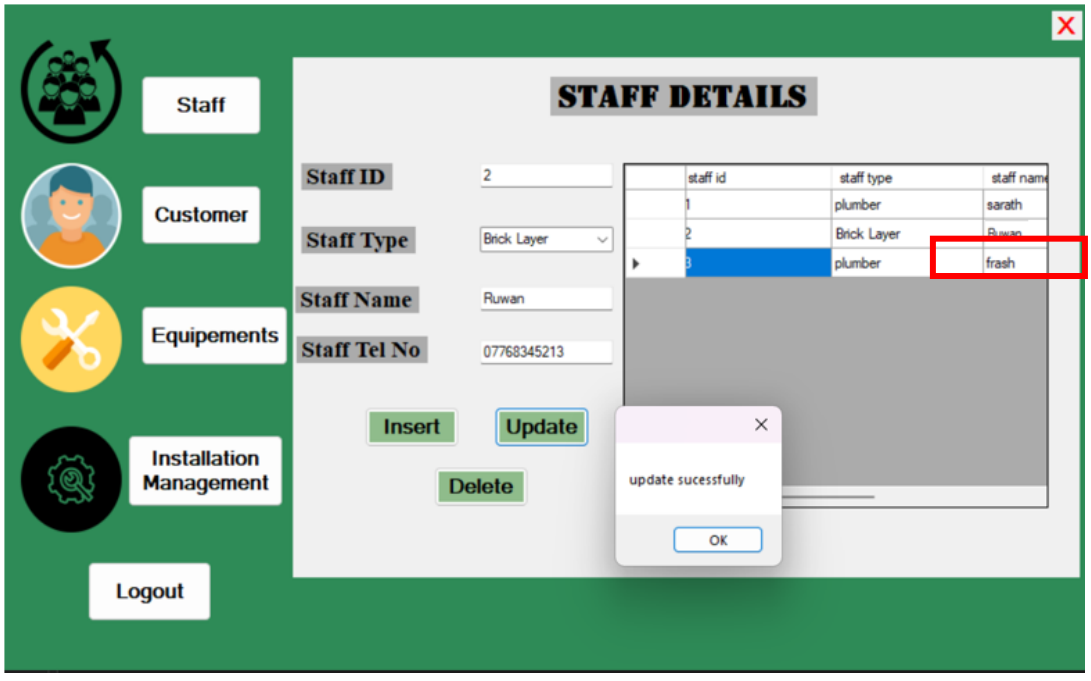



Figure 28 : After update Interface (Made by Author)

**Delete Testing in Staff interface.**

Testing the database's ability to delete records from the system while maintaining data integrity is done by deleting test data. Benefits of the delete is Executing delete operations reveals how the database manages flow deletes, foreign key restrictions, and data elimination. Additionally, it makes sure that deleting records won't have any unintended side consequences like empty data.

Example -:

| **Test Case ID** | | Delete -01 | | | |
|---|---|---|---|---|---|
| **Test Case Description** | | Delete – Positive Test Case | | | |
| **Tester Name** | | Sanadi Dianayana | | | |
| **No** | **Action** | **Inputs** | **Expected Output** | **Actual Output** | **Test Result (Success/fail)** |
| 01 | Delete the staff id 3 row | Staff id – 3<br>Staff Type – plumber<br>Staff Name – frash<br>Staff Tel No- 077653829 | deleted successfully! | Deleted successfully! | Success |

**Before Deleting the staff id 3 row Interface**

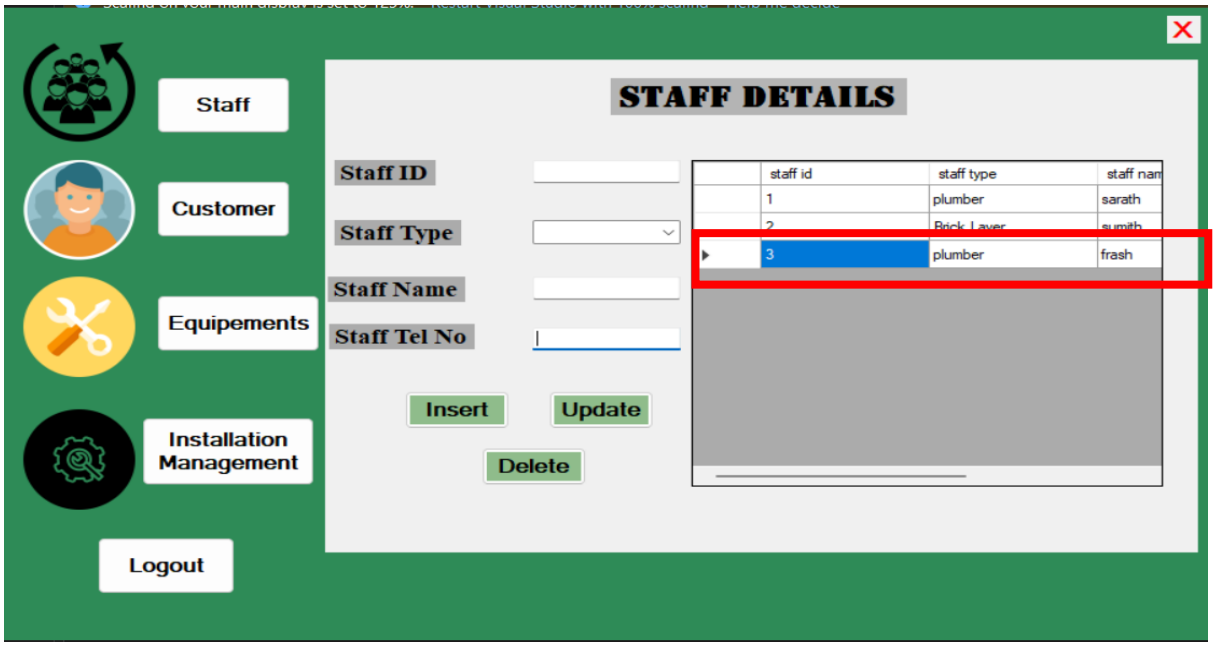



Figure 29 : Before Deleting the staff id 3 row Interface (Made by Author)

**After Deleting the staff id 3 row Interface**

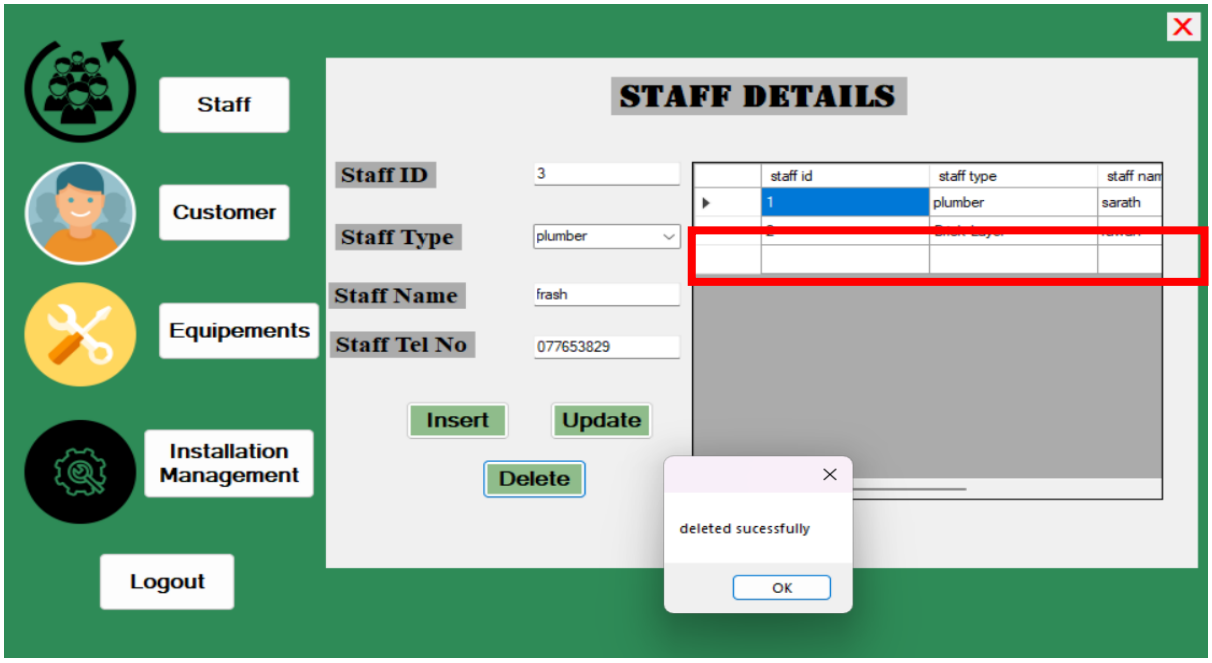



Figure 30 : After Deleting the staff id 3 row Interface ( Made by Author)

**User Documentation and Technical Documentation**

**User Manual**

Any type of documentation aimed at the final product of a good or service is referred to as user documentation or end-user documentation. This documentation's goal is to instruct users on how to set up, use, and/or troubleshoot a product correctly. Everybody has, at some time in life, read some sort of user manual. User manuals and instructions are frequently provided with devices that, among other things, include appliances, software programs, and gadgets that have a little learning curve. A typical user might not benefit fully from the product without this documentation. This might therefore lead to dissatisfied consumers, expensive customer care costs.

So author also made a user documentation to the system .

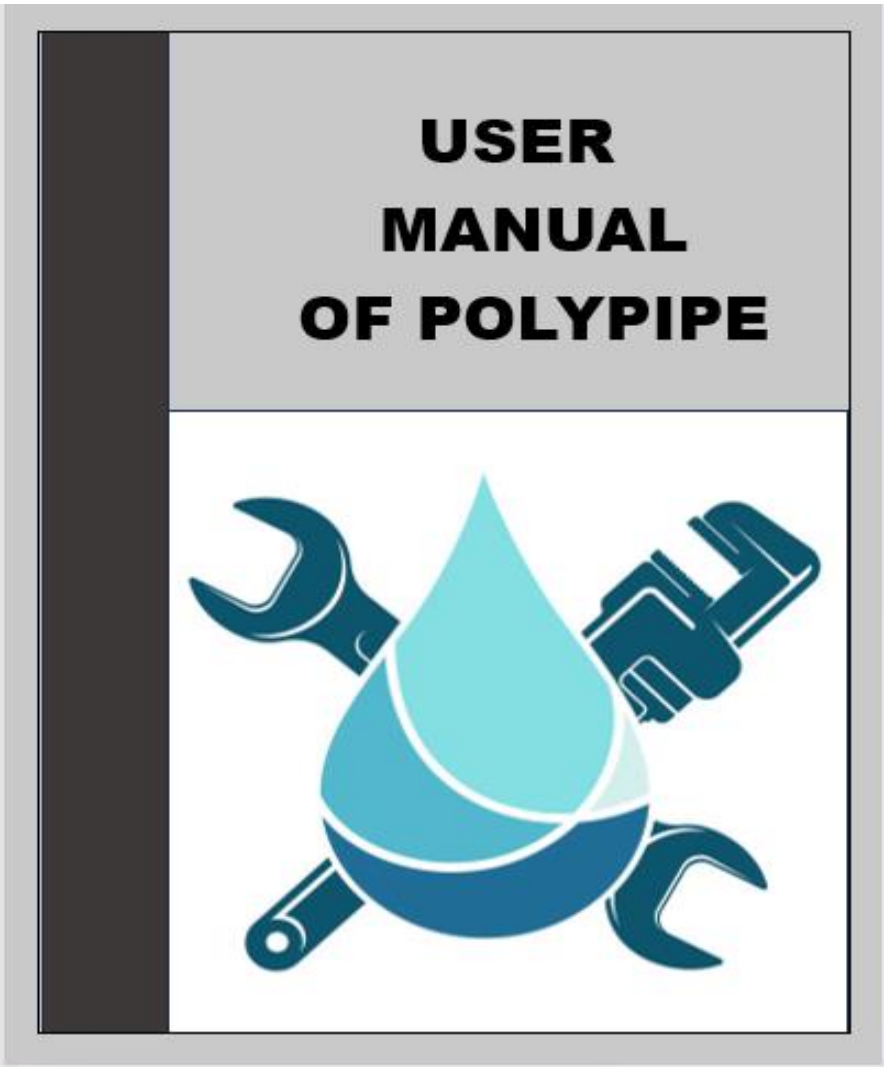
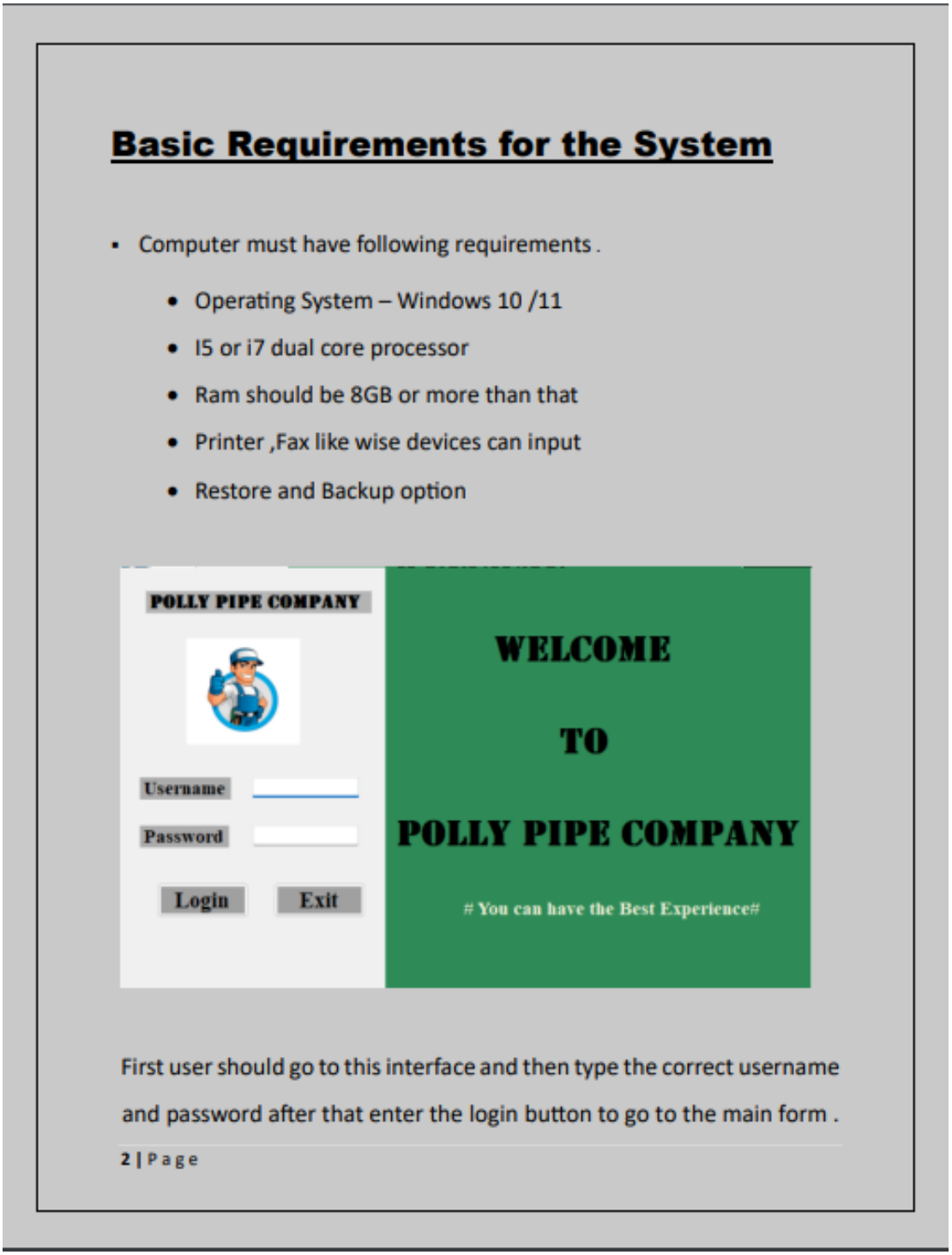



Figure 31 : User Manual interface 1



## Basic Requirements for the System

- Computer must have following requirements .
  - Operating System – Windows 10 /11
  - I5 or i7 dual core processor
  - Ram should be 8GB or more than that
  - Printer ,Fax like wise devices can input
  - Restore and Backup option

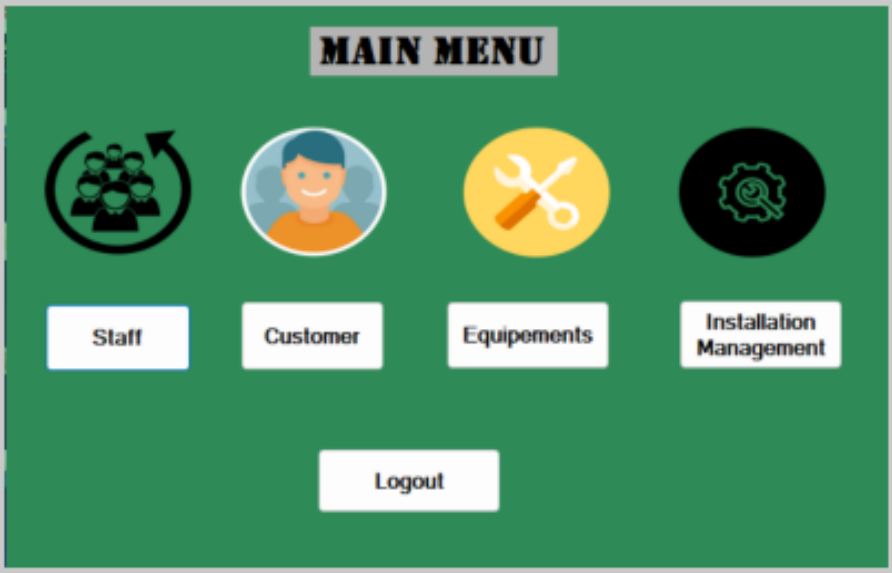


First user should go to this interface and then type the correct username and password after that enter the login button to go to the main form .

2 | Page

Figure 32 : User Manual interface 2

User cannot enter the page without having a correct username or password .

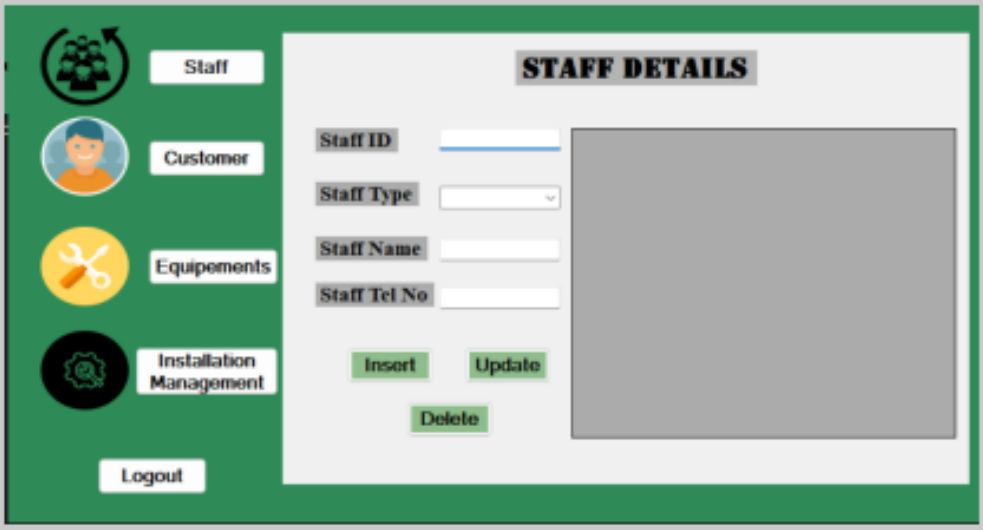



When enter the login form you can get into the Main form. In here you can select that you want then you can enter to the relevant interface to do your stuff easier. Each button that Named as Staff , Customer , Equipment's , Installation Management can go inside it. So after that you can get other stuff from this .



Figure 33 : User Manual interface 3

If you select the staff button from the Main menu, then you can go to the next interface .

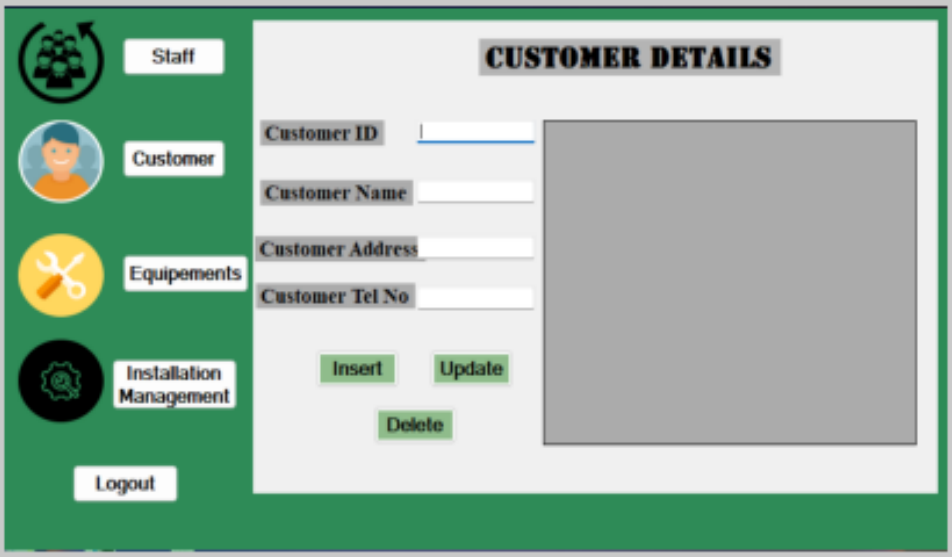



In this interface you can add staff id , can select staff type , can add staff Name and staff Tel no . If you want to update the information you can select the update button and if you want to delete some information you can click the delete button. In Left side you can select the form that you want to go .



Figure 34 : User Manual interface 4

In Main menu if you click the customer you can get the customer interface.

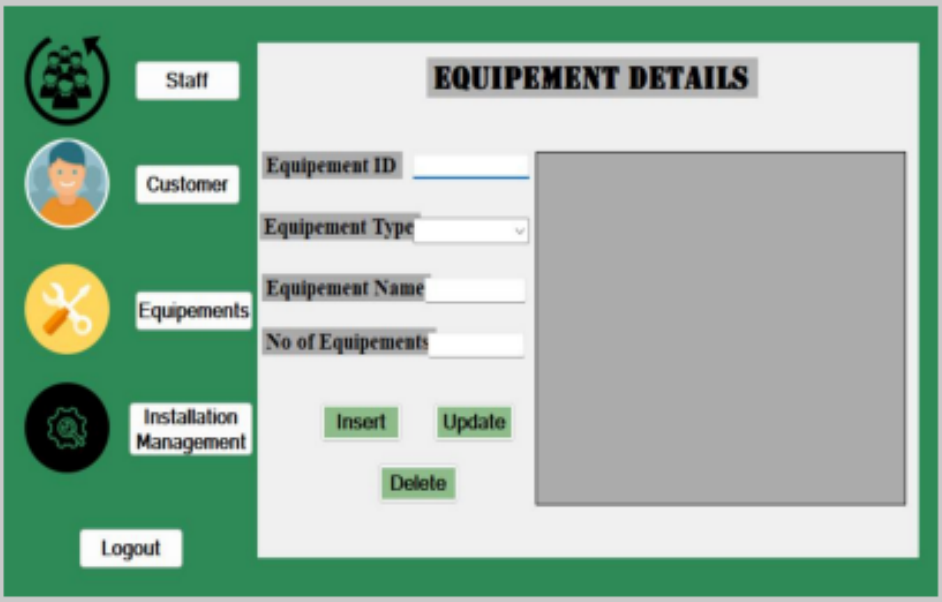



In this interface you can add customer Id , customer name , customer address and customer telephone number after that you can insert them . In here also you have the update and delete button.



Figure 35 : User Manual interface 5

When you get into the Mani menu if you click on equipment's then you can go to equipment form.




In this interface you can add equipment id , equipment type , equipment name and no of equipments after that you can insert them , update them or you can also delete that stuff by clicking the insert, update delete buttons.

6 | Page

Figure 36 : User Manual Interface 6

Finally if you click the installation management button you can go to its interface.


In this interface you can add the installation id , installation type , no of equipements, no of staff , customer id , staff id, equipment id and also you can select the project start date and end date if you want some information to get update you can click the update button and if you want to delete then click the delete button.

7 | Page

Figure 37 : User Manual interface 7

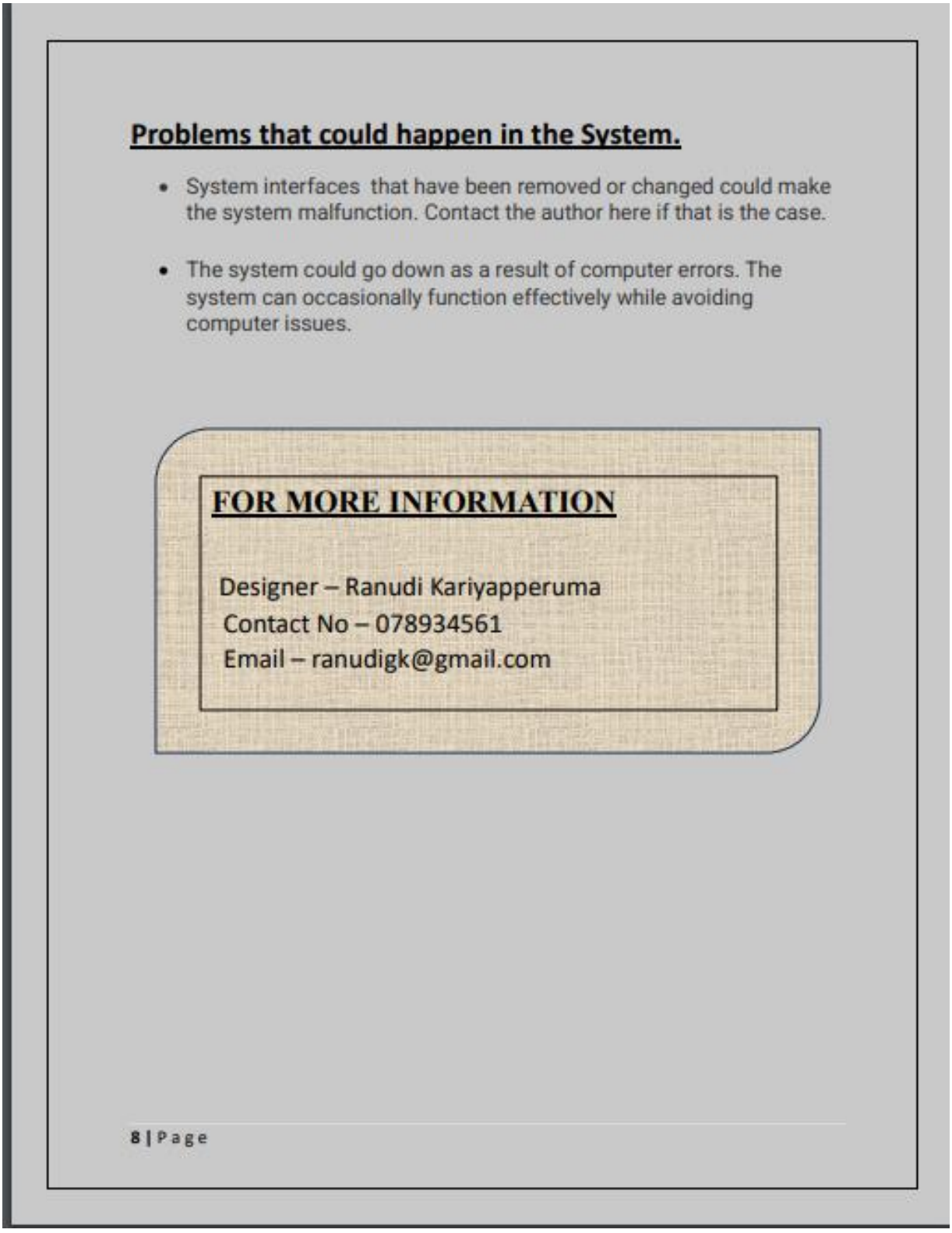



Figure 38 : User Manual Interface 8

## ER Diagram

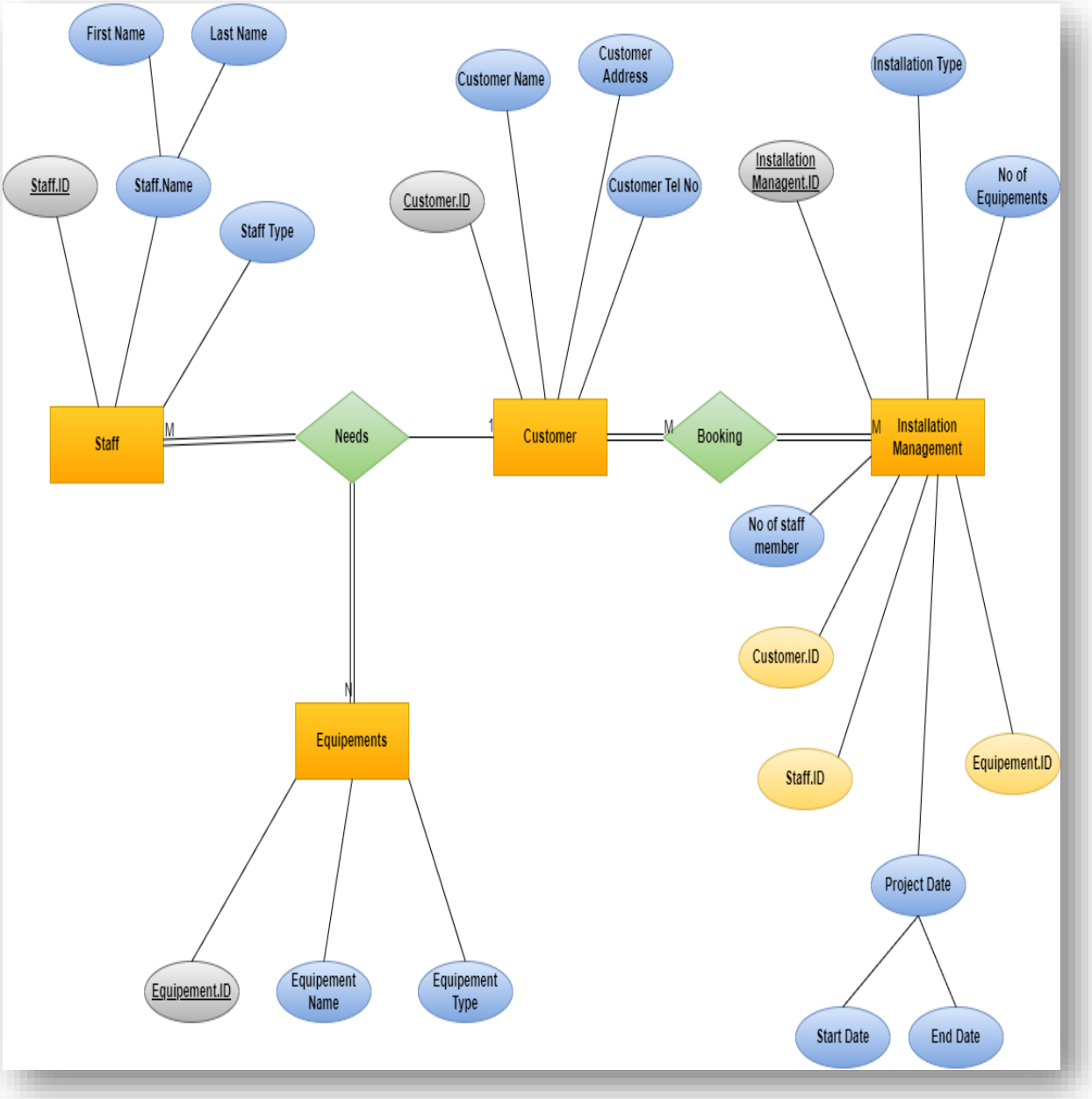



Figure 39 : Er diagram

In Er diagrams it shows the relationship between attributes and entities So

This diagram was produced by the writer in accordance with user and system requirements. The ER diagram provides the backbone for this database building. The next stage of system building is done by developers following the creation of this ER diagram.

**Logical Database Diagram**

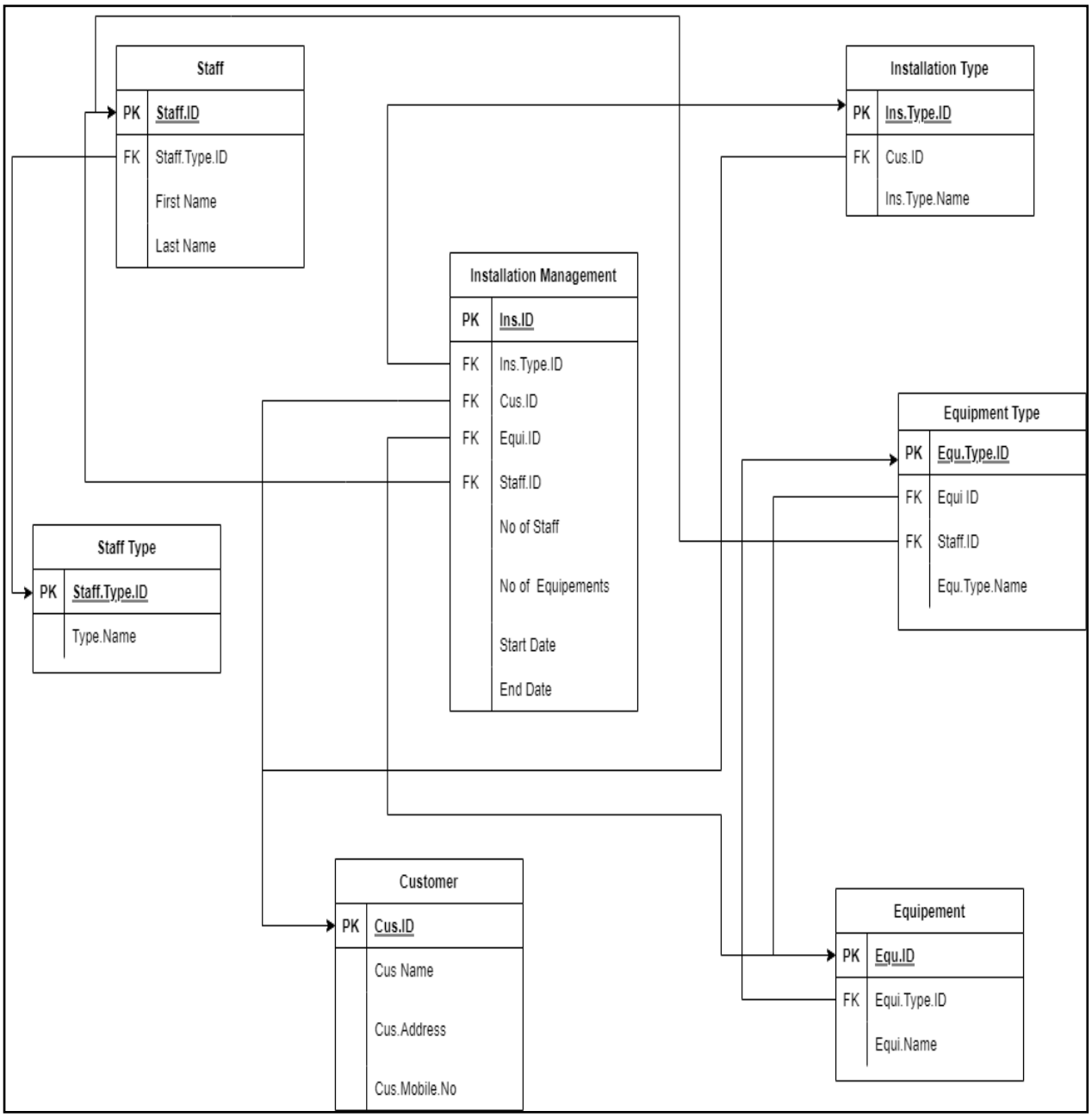
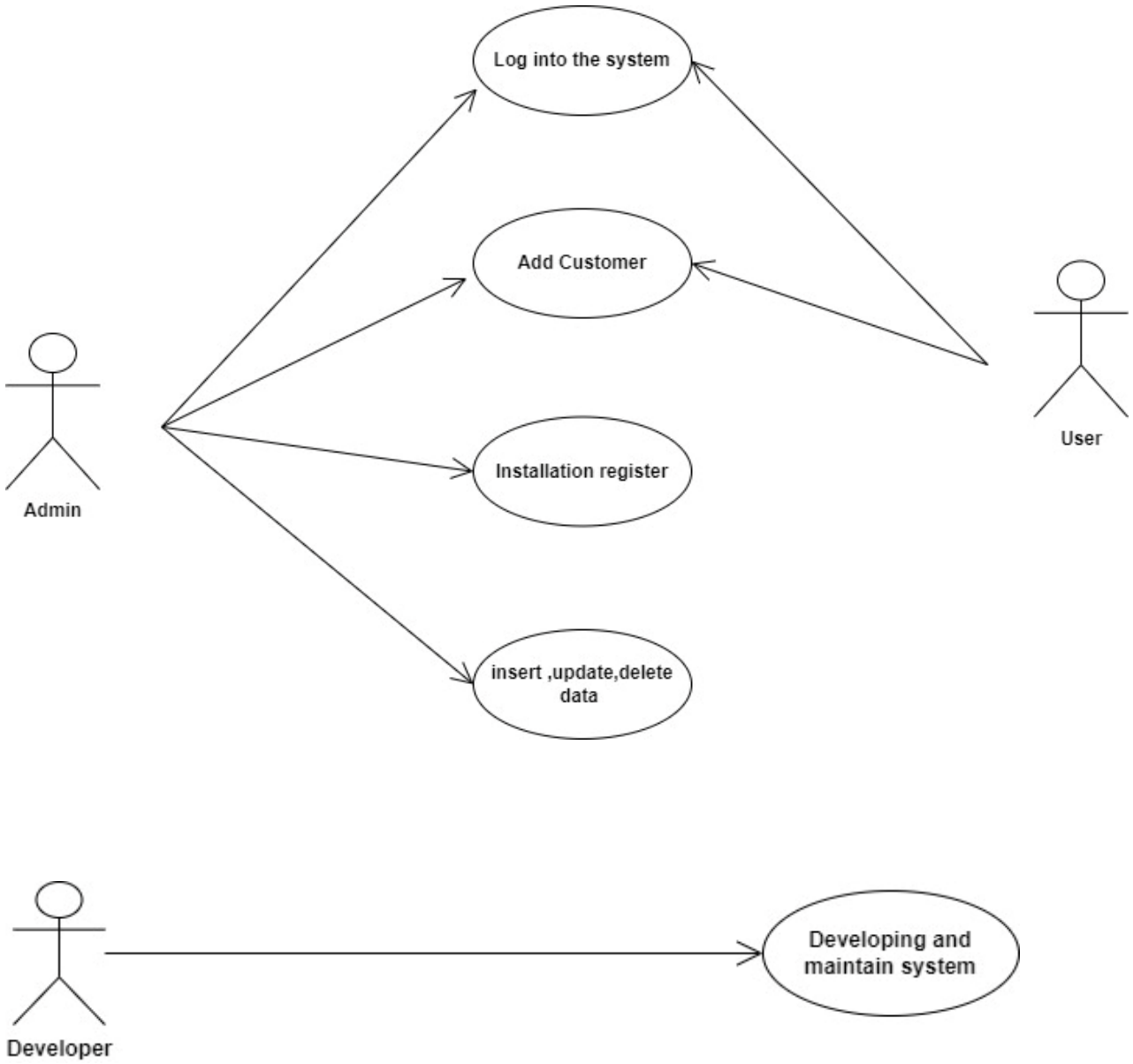



Figure 40 : Logical database

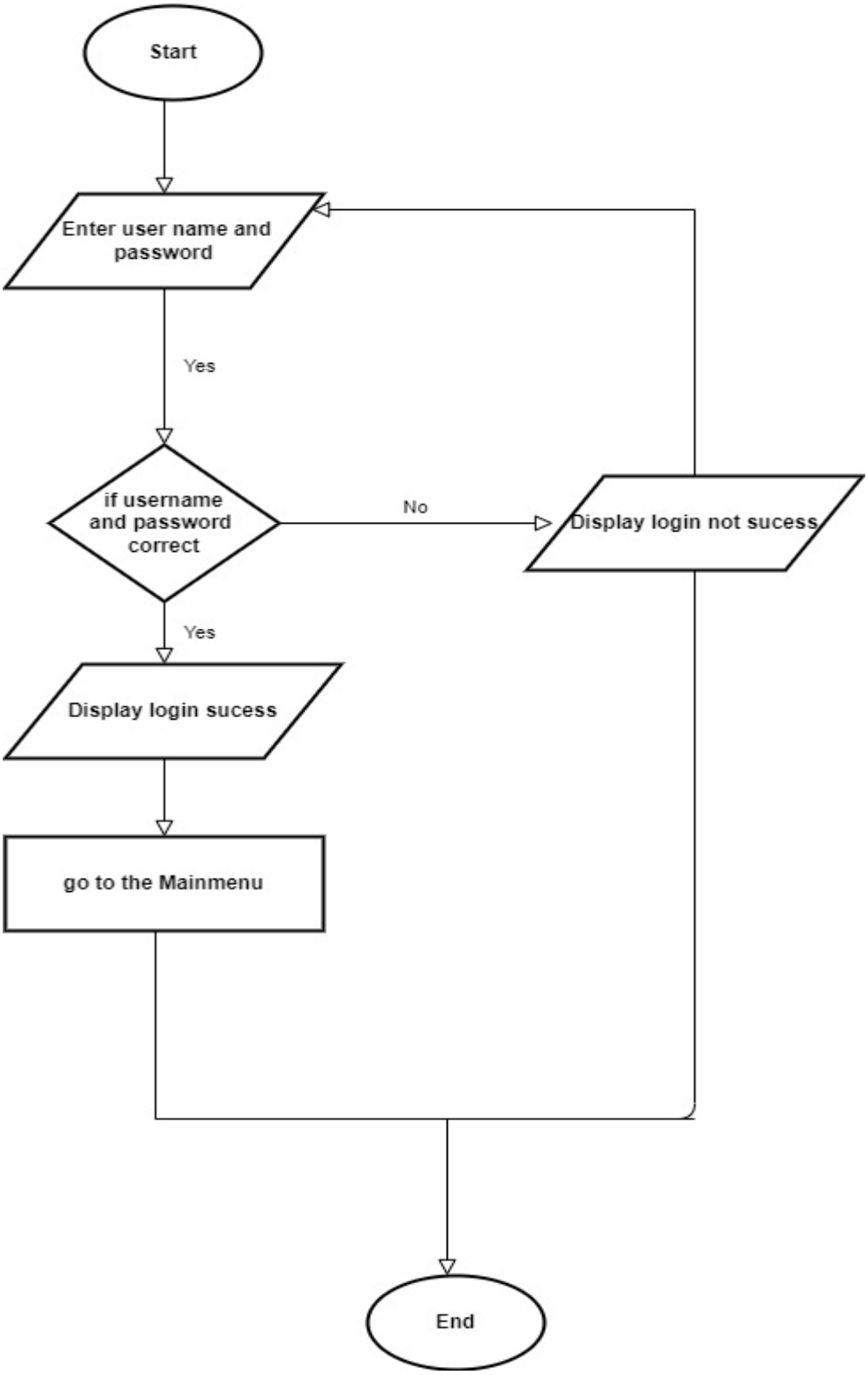
This logical database was created from looking the Er diagram and in here it can identify all the data that want for develop the database.

**Use case diagram**

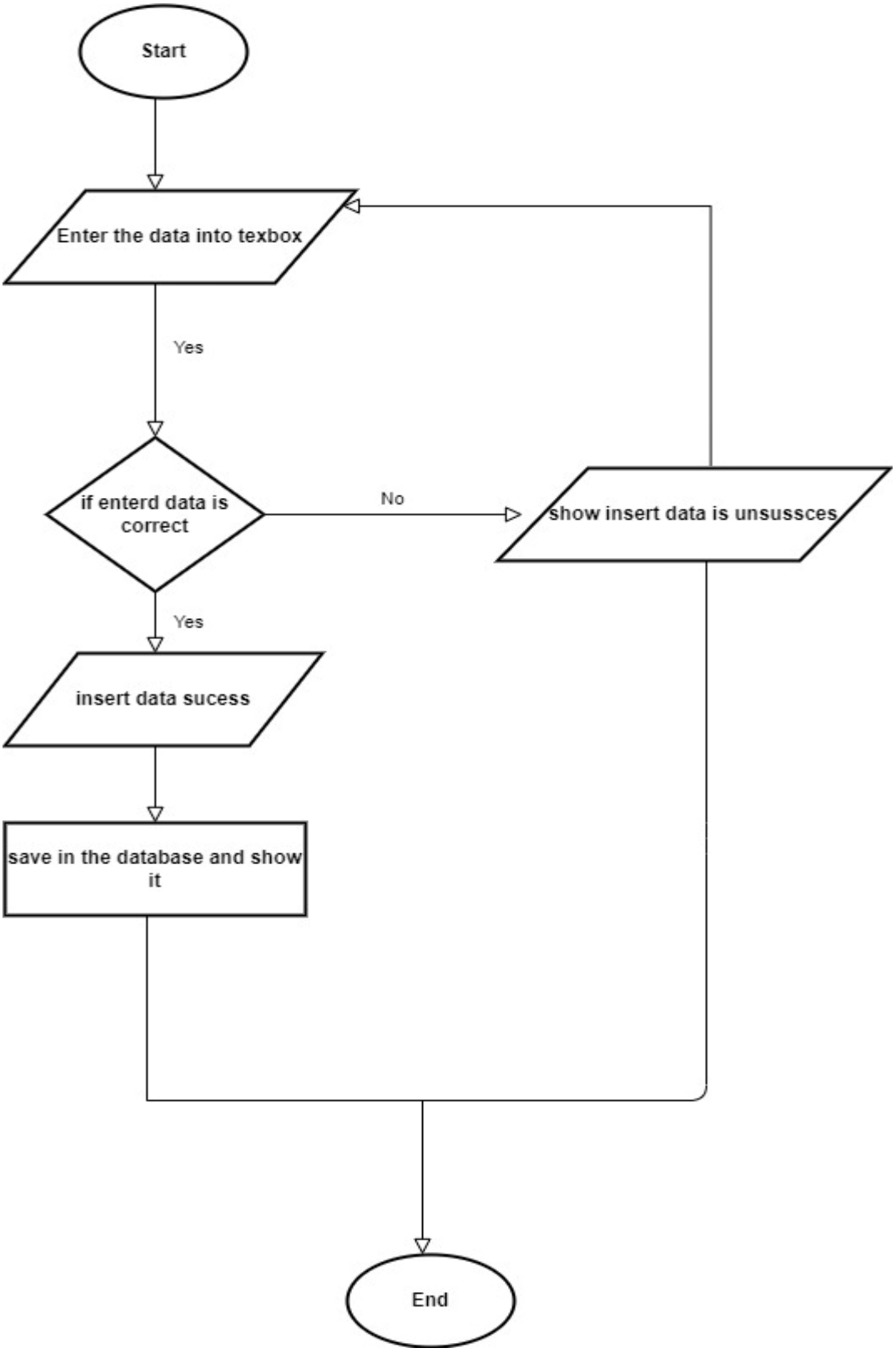


**Flow chart**

- **Login form**



- **Insert**


Start
Enter the data into texbox
Yes
if enterd data is correct
No
show insert data is unsussces
Yes
insert data sucess
save in the database and show it
End

**The effectiveness of the database solution in relation to user and system requirements, and suggest improvements.**

To check the database for users is it compatible to users author create a feedback form that can get the results how successful about the pollypipe database. This feedback form was made by the Author using google forms in their author implement seven questions and give to select multiple answers to users. For users it is very easy to select the answers through the google form. Also, author add a section called suggestions to get user suggestions and what should improve in the system.

**Design of Feedback Form**

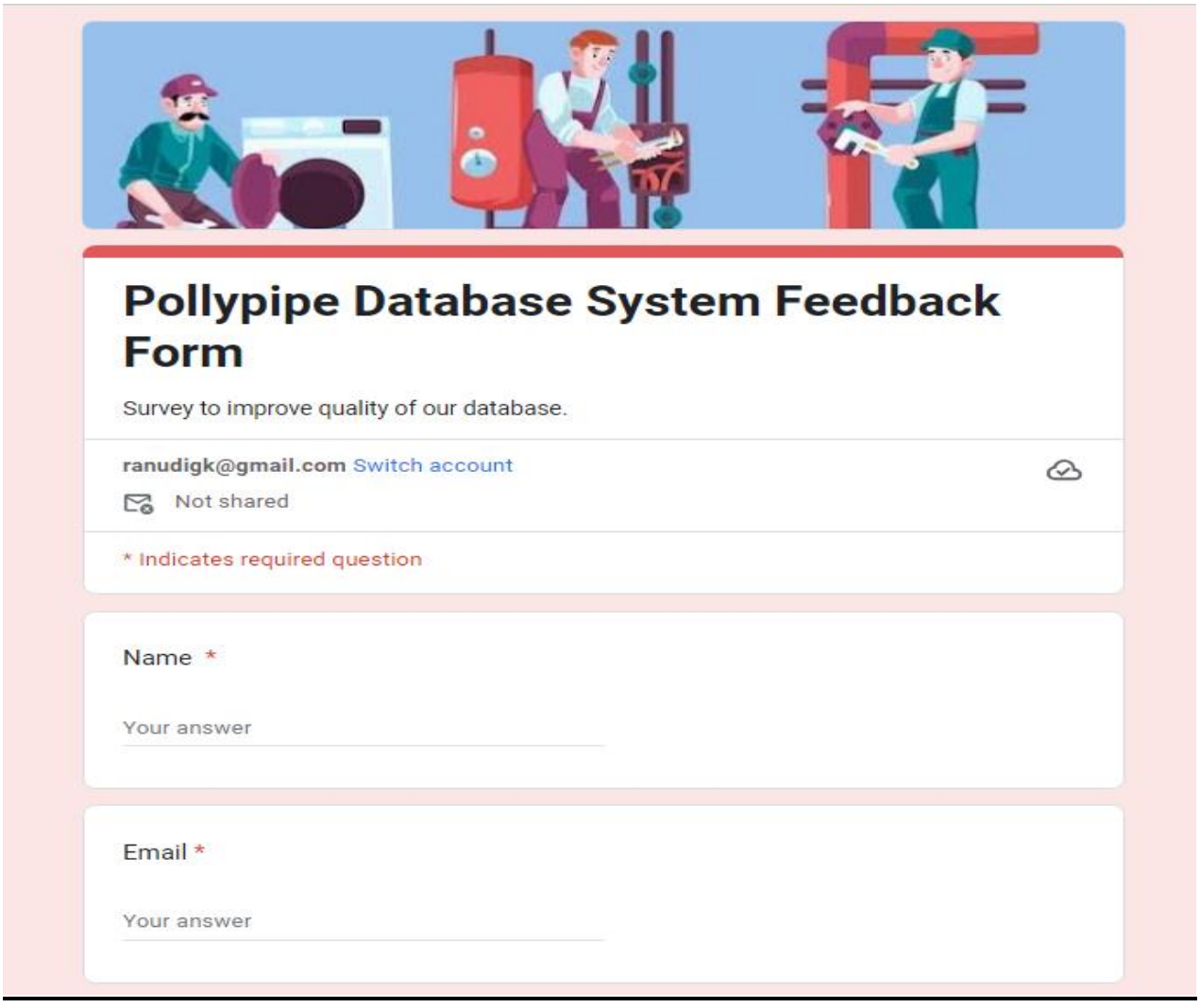



Figure 41 : feedback form

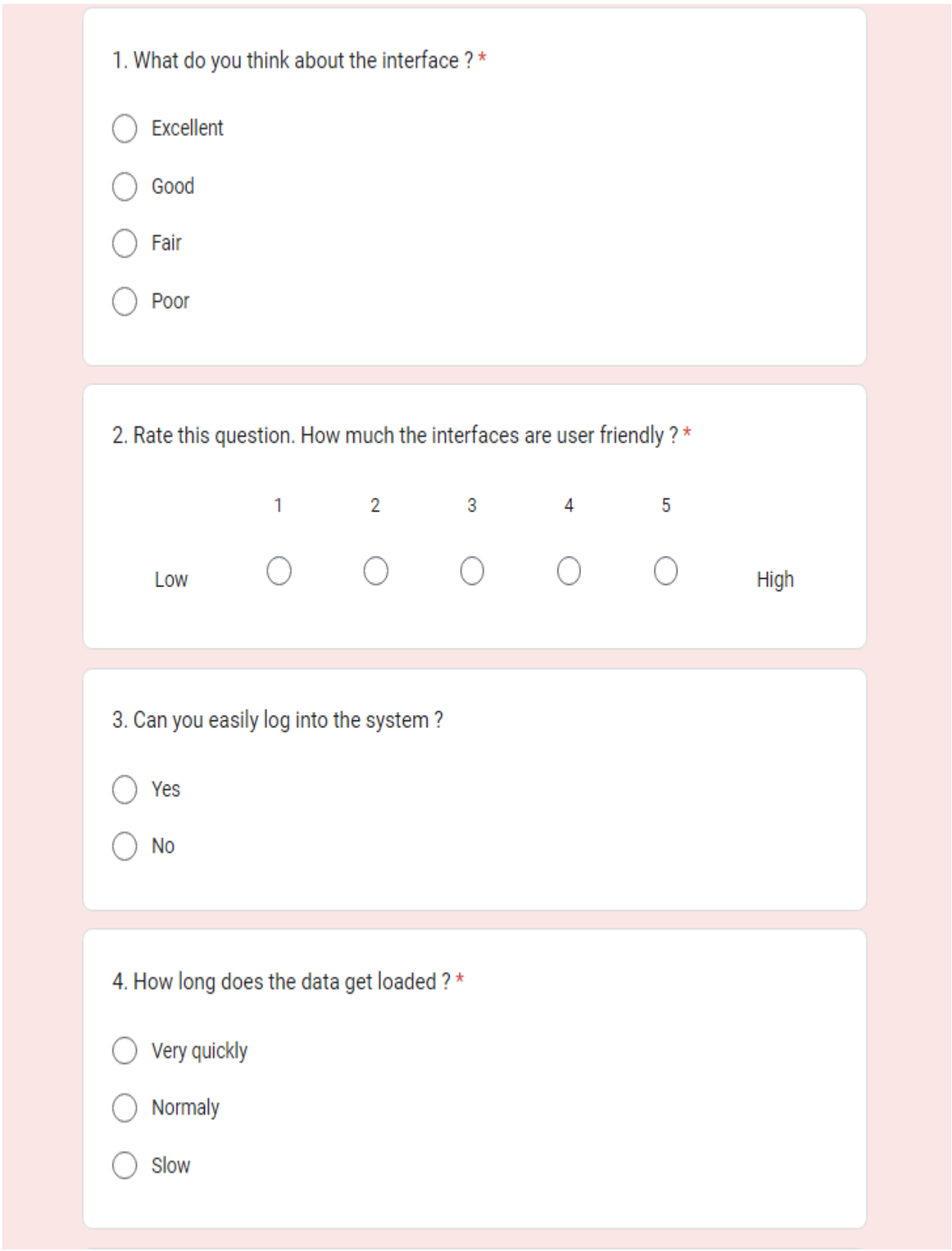



Figure 42 : feedback form 2

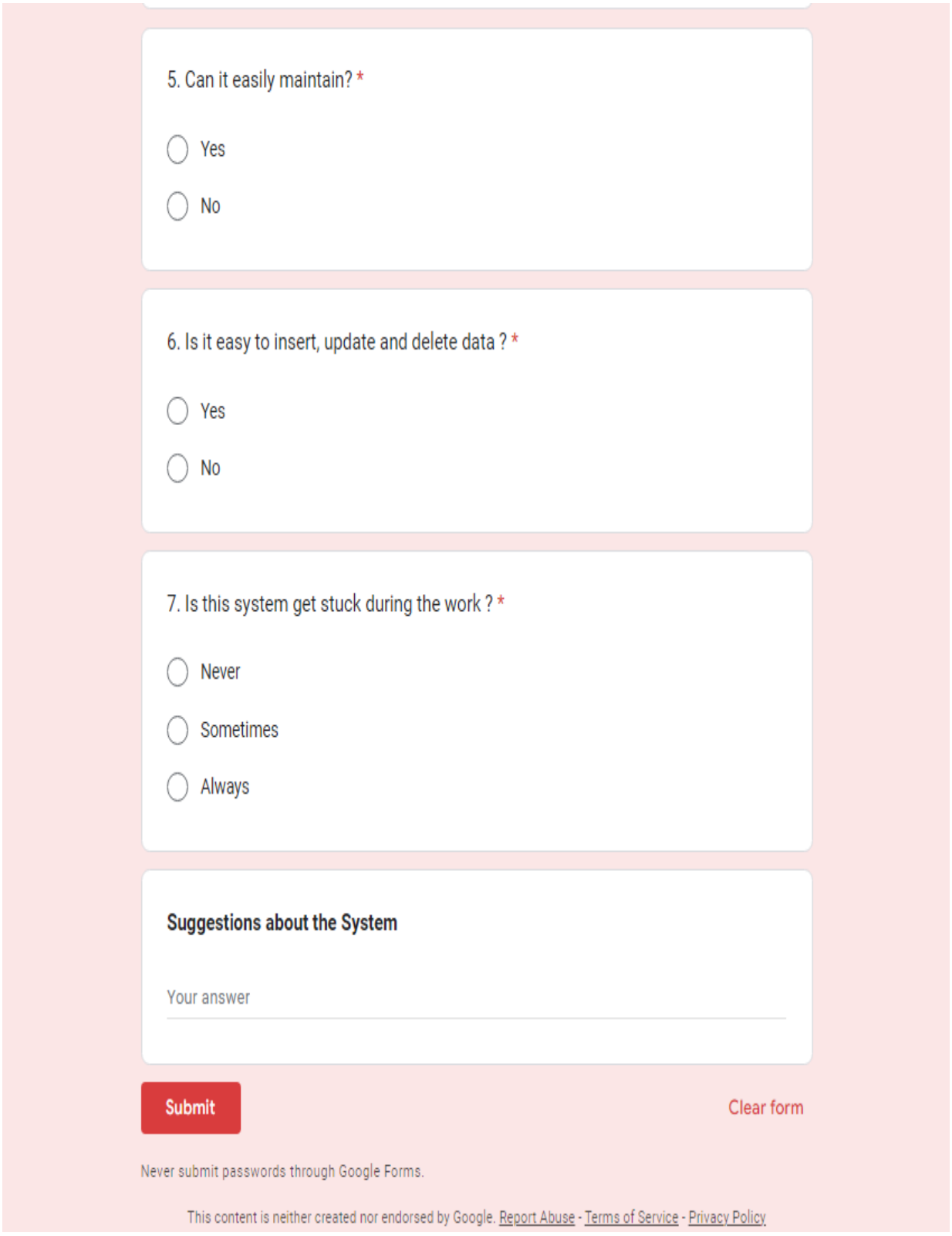



Figure 43 :feedback form 3

**Example of Feedback form**

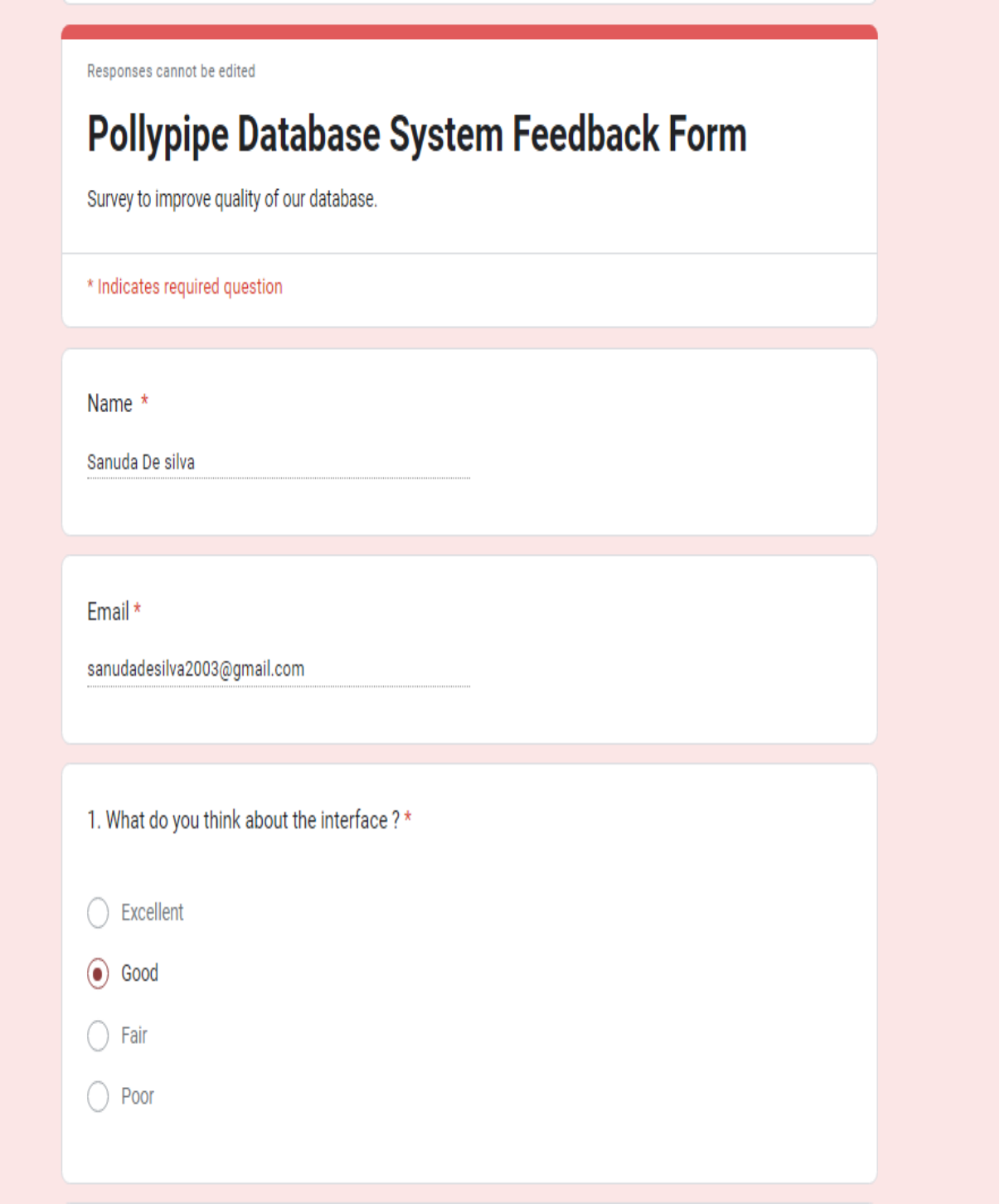



Figure 44 : feedback form 4

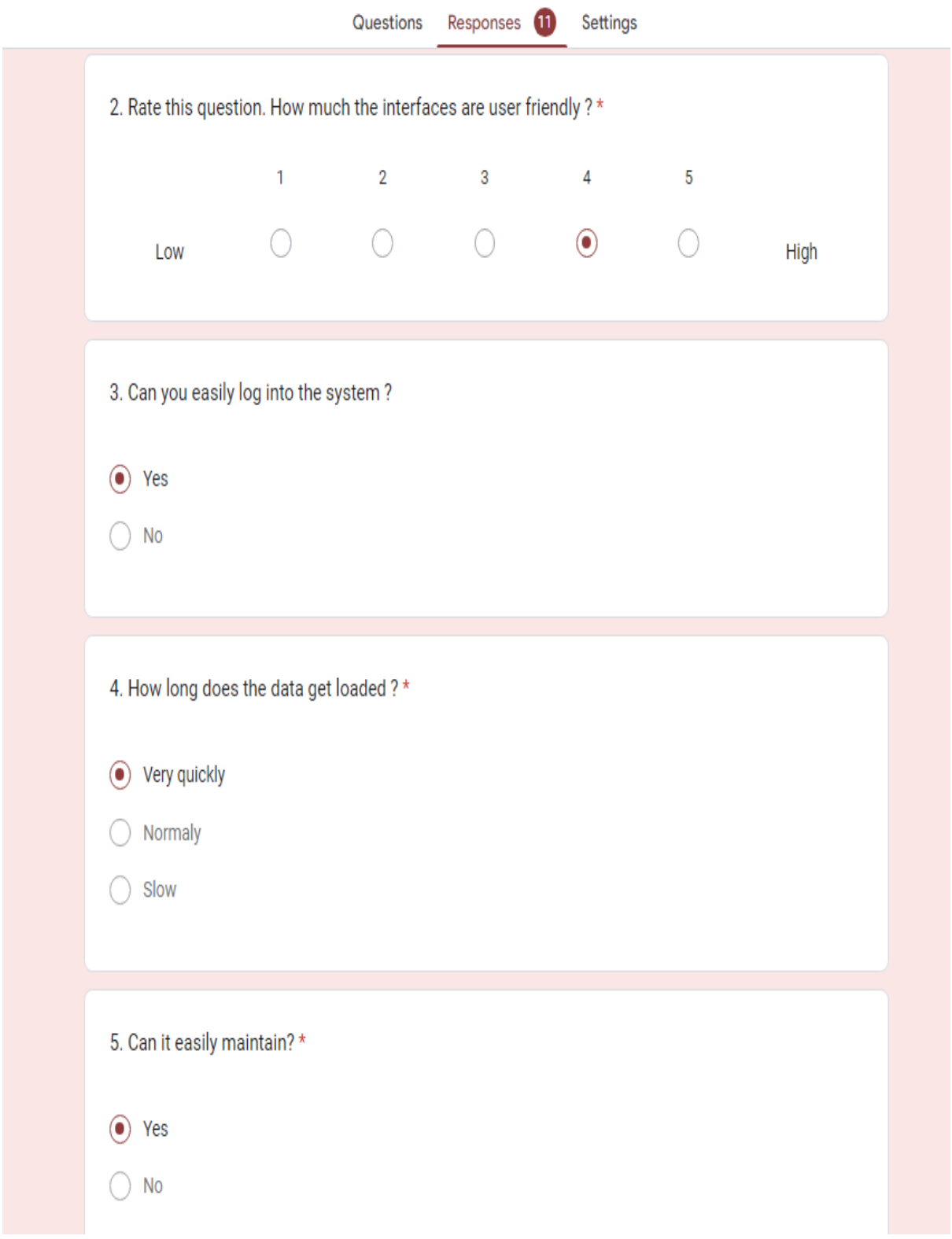



Figure 45 : feedback form 5

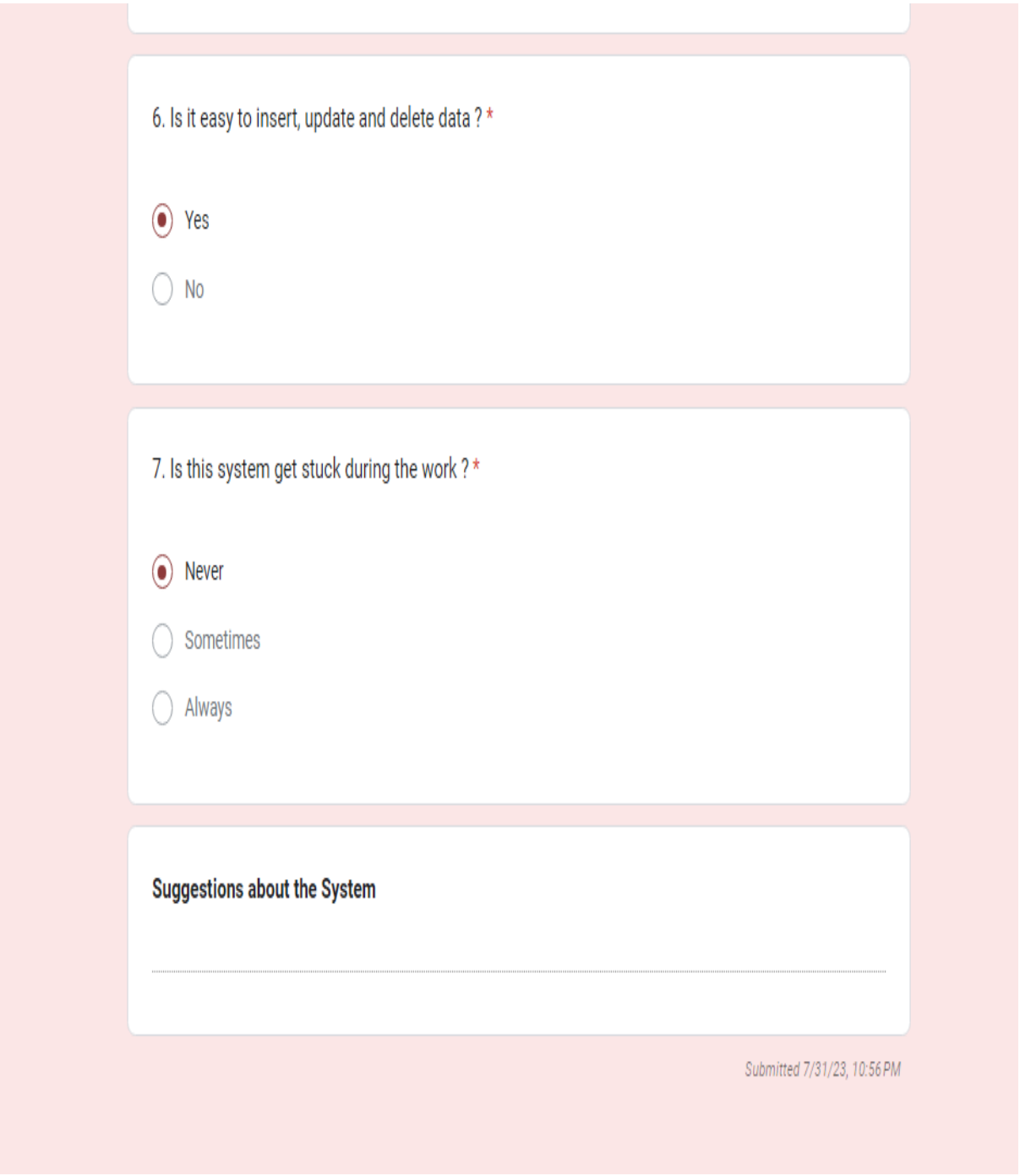
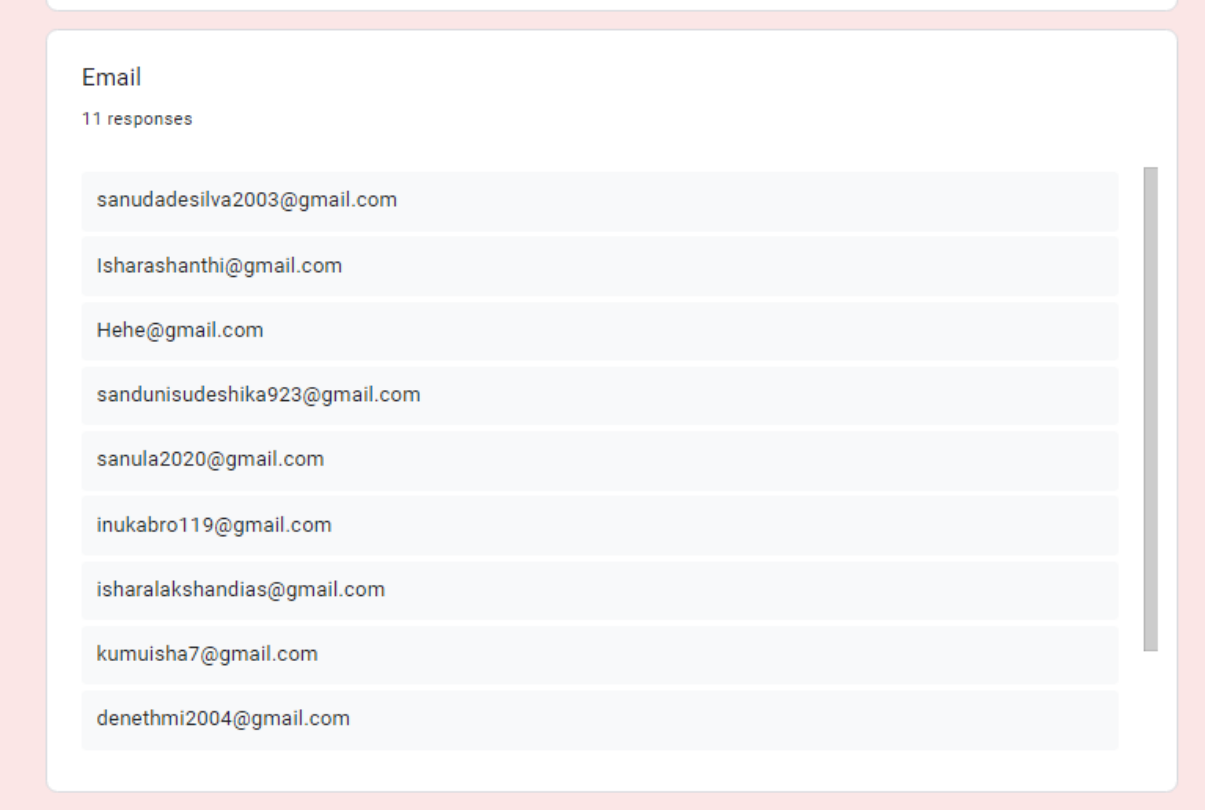


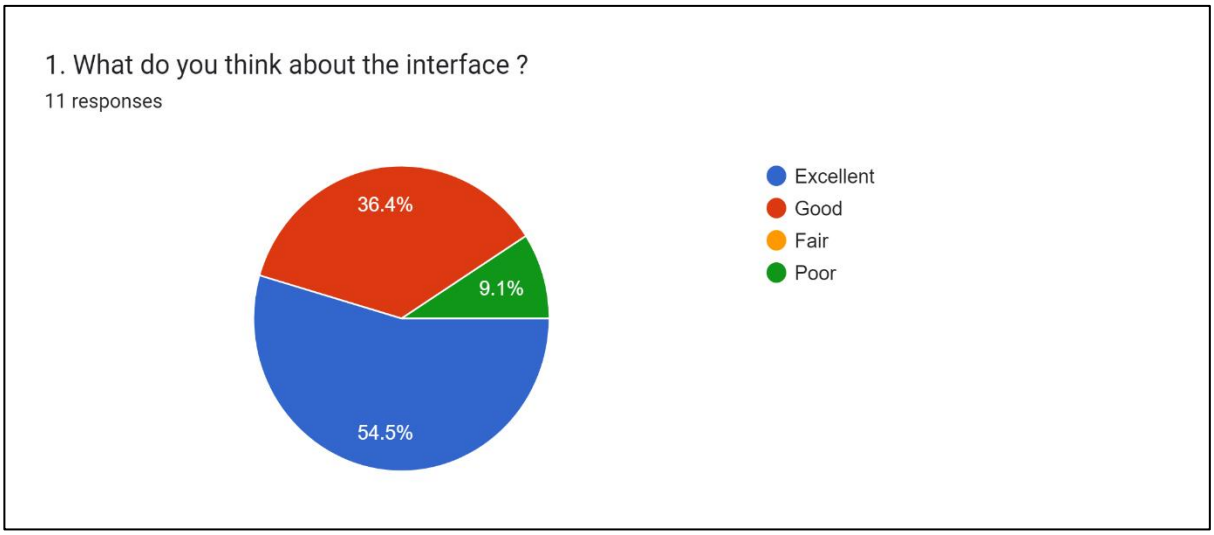

Figure 46 :feedback form 6

**Responses Persons details**

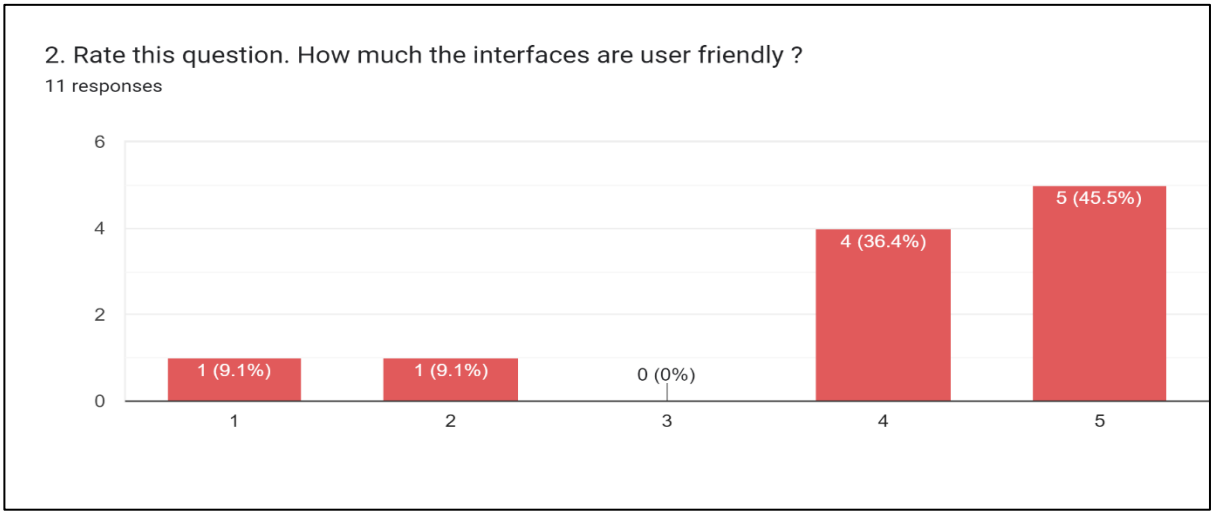
**Feedback Review**

**Question 01**



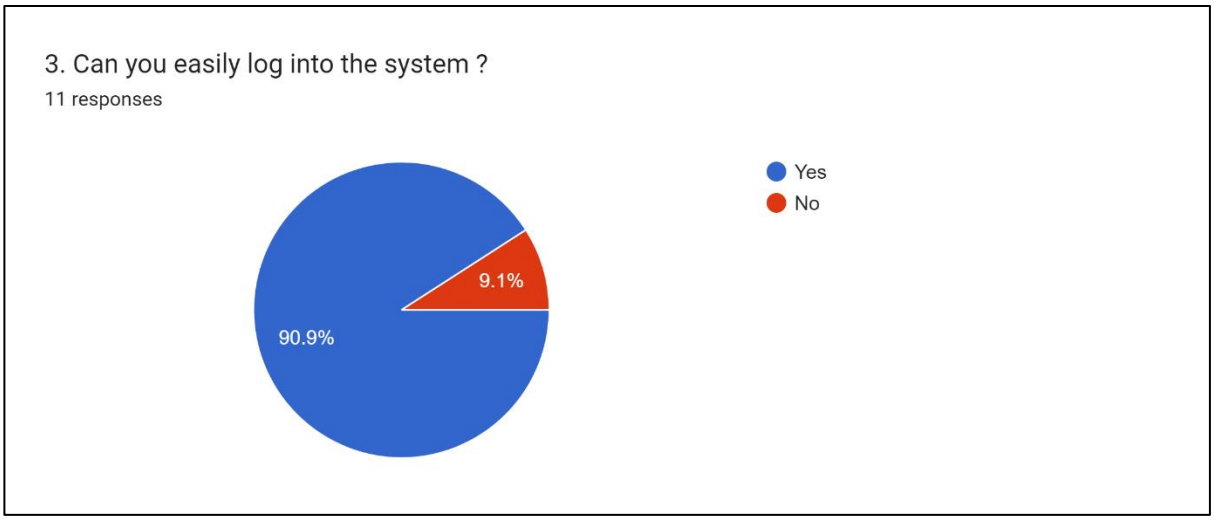

Author wants to check whether the interface is okay with the users . In this Chart first question 54.5% users are satisfied about the interface appearances.36.4% users are selected the interface is good and 9.1% users are response as poor. As the chart it shows that the interface is good for most of the users.

**Question 02**



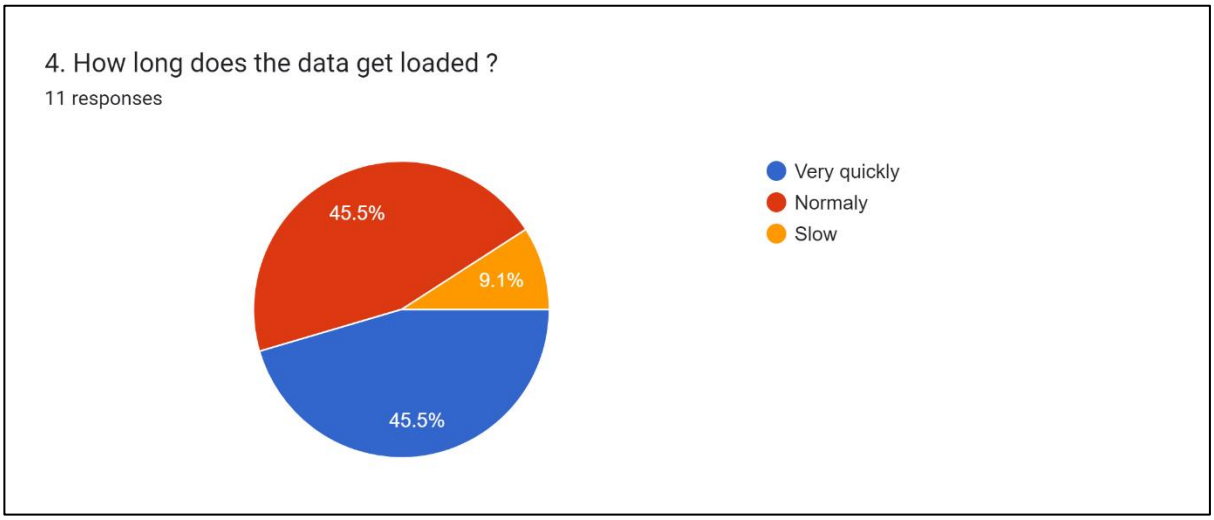

In this question author need to ask about the friendliness of the interface so as the bar chart it shows the highest number of users are rated that it is user-friendly and it is 45.5% and some rate 4 that means 36.4% users rated it . and equally users rated 1 and 2 that is 9.1%.

**Question 03**



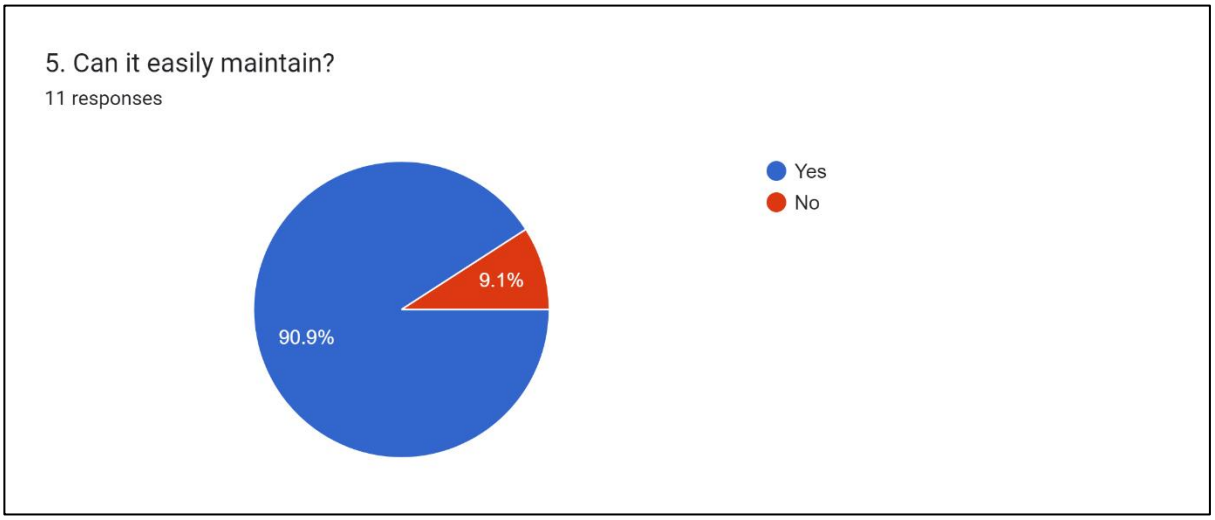

In this chart author need about the how effectively can log into the system so most of users that means 90.9% users are satisfied about the login of the system and minimum is 9,1% users tell that it is hard to login in to the system.

**Question 04**



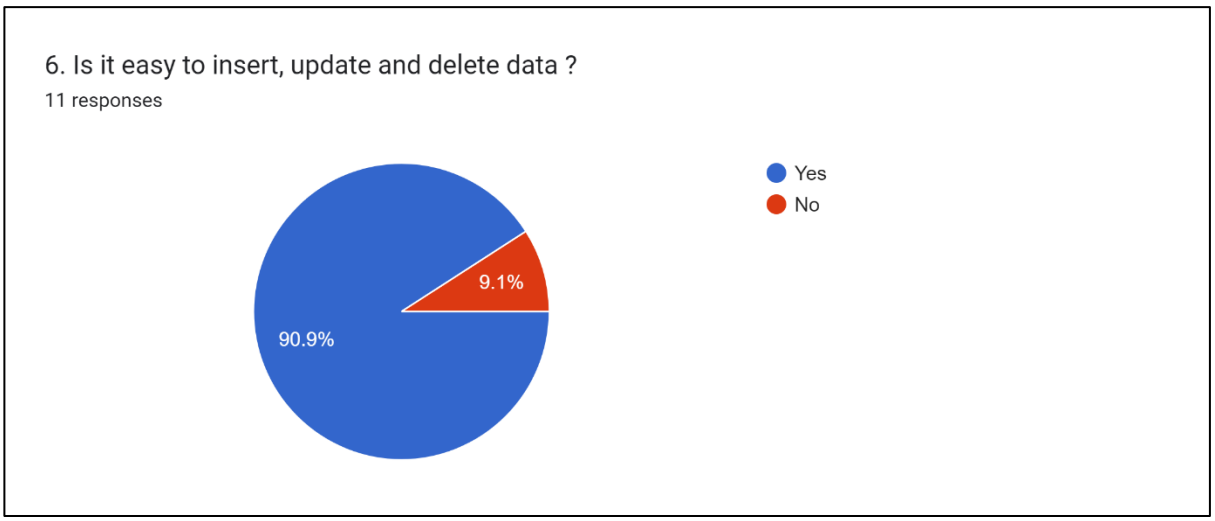

This 4th question author asked about how long that the data is loaded to the data so in here equal number of users selected very quickly and normally in the feedback form, so its 45.5% and 9.1% users selected slow. Overall, the data are loaded pretty quickly to the database.

**Question 05**



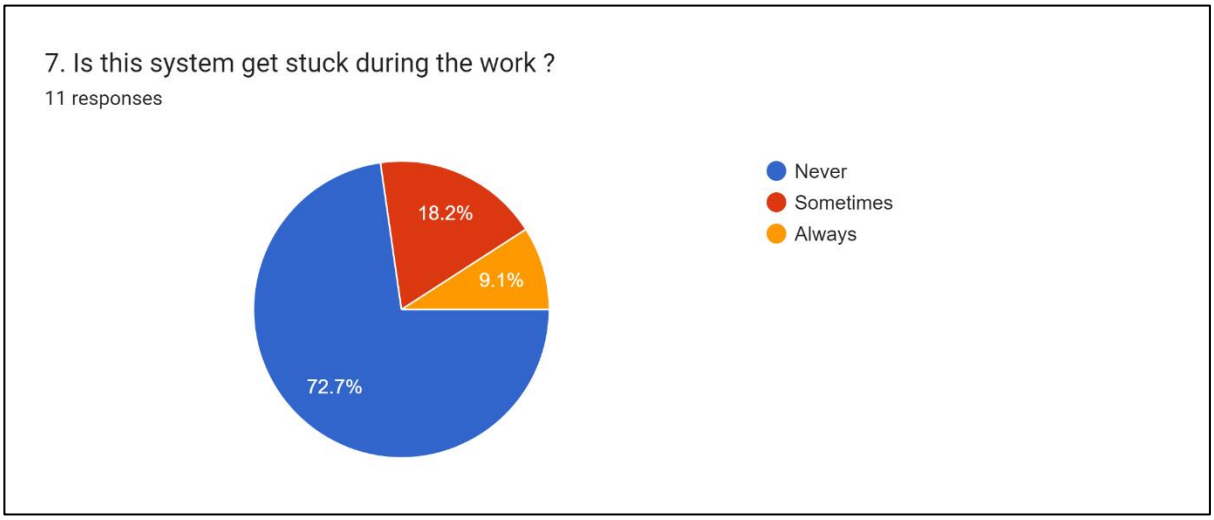

In 5th question author asked about the maintainers of the database that is it easy. so users need to select answer as yes or no . so Highest number of users that mean 90.9% users selected yes for this question and 9.1% users selected no for this question. As the conclusion it shows that database is easy to main.

**Question 06**




In this chart it asked about in database is it can easily to insert data, update data and delete. data. So, most of users select yes to this question in percentage it is 90.9% and 9.1% select no for this question. Overall in the database it is easy to insert, update and delete data.

**Question 07**



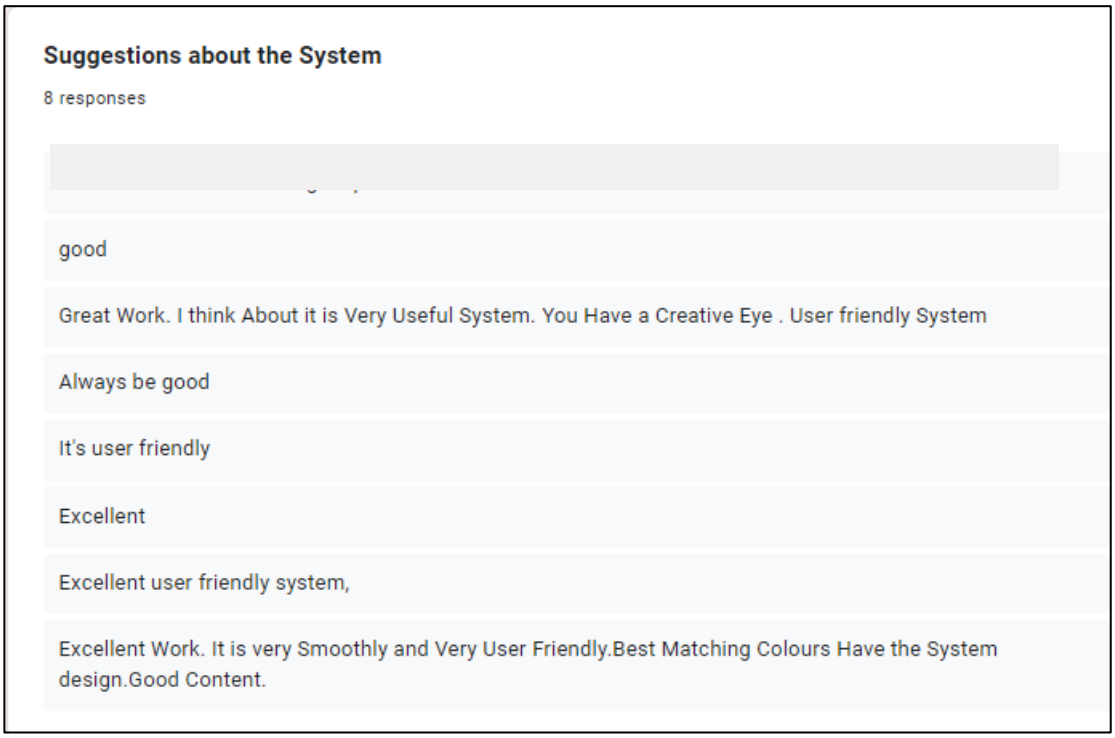

So finally, the in the last question author asked about the when user work with the database is it stuck as the chart it shows that 72.7% users say that it never get stuck while working with the database and 18.2% users are selected as sometimes. Also 9.1% users choose always as the answer. It shows that it is good database for user purposes.

**Summary of User feedback**

| Question | Answers |
|---|---|
| 1. What do you think about the interface ? | Excellent<br>Good<br>Fair<br>Poor |
| 2. Rate this question. How much the interfaces are user friendly ? | Low<br>1<br>2<br>3<br>4<br>5<br>High |
| 3. Can you easily log into the system ? | Yes<br>No |
| 4. How long does the data get loaded ? | Very quickly<br>Normaly<br>Slow |

| | |
|---|---|
| 5. Can it easily maintain? | Yes<br>No |
| 6. Is it easy to insert, update and delete data ? | Yes<br>No |
| 7. Is this system get stuck during the work ? | Never<br>Sometimes<br>Always<br>Suggestions about the System |

**Suggestions and improvements that users give for the Pollypipe database system**




To get the suggestions from users author add suggestion part to the Feedback Form. Through this form developer can get the idea how about the system that developer developed. Also it is easily to get suggestions from users but author cant reply for the users but developer can get a idea about the system and problems of it.

**The database in terms of improvements needed to ensure the continued effectiveness of the system.**

## Database Improvements

The design, functionality, and security of a database can all be improved. In order to lower the cost of storage and access, as well as to decrease downtime, these innovations make sure that data is accurate, safe, accessible , and efficiently kept.

## Future improvements that need to do in the pollypipe database system.

1. Improved more data Security.

   In database it is very effective to add more security because in a database sometimes there are sensitive data so it can be a risk while having lack of security. For improving security developer can implement encryption and access controls, two step verification methods like wise.

2. Faster page Loads

   It is important to have fast page loader because sometimes user get angry when it get more time for load so it is better to have fast in page loading. For that there should be normalised database to store data fast and access for customers to get information quickly.

3. User Retention

   Maintaining a user's use of a service or feature is known as user retention. It's a crucial metric for assessing the accomplishment of digital and SaaS products. It calculates the proportion of first-time users who return over time. Can investigate a specific period of time or frequently assess user retention. Can examine the number of logins over time if you're researching user retention for user product

4. Improving Scalability

When a business needs more resources, a database's capacity to increase its availability and behavior is known as scalability.In scalability there two types of scalability. That are named as Horizontal scalability and vertical scalability. So to improve scalability in polypilpe database it can replication it means that it forms creates copies of database , it can sharding it means that moving data across the node known as portioning like wise.

5. Reduce Database Size

The cost of storing and maintaining huge databases prevents them from being useful for professionals. The performance of database operations decreases as the size of the database grows because the SQL server needs more memory and computing power to get data from the tables. Additionally, it's possible that the growing databases will necessitate the need for extra storage space. So in Pollypipe database system developer think that it need to be more increased and it can be easy to handle customer and the developer.

## References


- Anon., n.d. [Online]
  Available at: https://www.businesstechweekly.com/operational-efficiency/datmanagement/databases-advantages-benefits/

- Anon., n.d. *Couch Base.* [Online]
  Available at: https://www.couchbase.com/resources/concepts/database-scalability/#:~:text=Database%20scalability%20is%20not%20just,also%20down%20if%20demand%20decreases.

- Anon., n.d. *Dopinger Blog.* [Online]
  Available at: https://www.dopinger.com/blog/database-size-what-it-is-how-to-reduce-it

- Anon., n.d. *MUO.* [Online]
  Available at: https://www.makeuseof.com/

- Anon., n.d. *NightinggaleHQ.* [Online]
  Available at: https://nightingalehq.ai/knowledgebase/glossary/what-are-database-improvements/

- Anon., n.d. *Ofni Systems.* [Online]
  Available at: http://www.ofnisystems.com/

- Siedle, J., 2015. *Information Systems Analysis Fall.* [Online]
  Available at: https://www.umsl.edu

- Support, T., n.d. *OfficeTools.* [Online]
  Available at: https://www.officetools.com/

- Villegas, F., n.d. *Question Pro.* [Online]
  Available at: https://www.questionpro.com/blog/user-retention/#:~:text=What%20is%20user%20retention%3F,who%20return%20in%20successive%20periods.

| Grading Criteria | Achieved | Feedback |
|---|---|---|
| **LO1** Use an appropriate design tool to design a relational database system for a substantial problem | | |
| **P1** Design a relational database system using appropriate design tools and techniques, containing at least four interrelated tables, with clear statements of user and system requirements. | | |
| **M1** Produce a comprehensive design for a fully functional system that includes interface and output designs, data validations and data normalization. | | |
| **D1** Evaluate the effectiveness of the design in relation to user and system requirements. | | |

| | | |
|---|---|---|
| **LO2** Develop a fully functional relational database system, based on an existing system design | | |
| **P2** Develop the database system with evidence of user interface, output, and data validations, and querying across multiple tables. | | |
| **P3** Implement a query language into the relational database system | | |
| **M2** Implement a fully functional database system that includes system security and database maintenance. | | |
| **M3** Assess whether meaningful data has been extracted using query tools to produce appropriate management information. | | |

| | | |
|---|---|---|
| **LO3** Test the systems against user and system requirements | | |
| **P4** Test the system against user and system requirements. | | |
| **M4** Assess the effectiveness of the testing, including an explanation of the choice of test data used. | | |
| **LO2 & LO3**<br>**D2** Evaluate the effectiveness of the database solution in relation to user and system requirements, and suggest improvements. (Anon., n.d.) | | |
| **LO4** Produce technical and user documentation | | |
| **P5** Produce technical and user documentation. | | |
| **M5** Produce technical and user documentation for a fully functional system, including diagrams showing movement of data through the system, and flowcharts describing how the system works. | | |
| **D3** Evaluate the database in terms of improvements needed to ensure the continued effectiveness of the system. | | |