MARCH 1, 2022

TeamAlpha Database

Bob's Fixit Group Case Study

Department of Information Technology, Bellevue University

CIS535-T301 Management and Design of Database

Professor Al Norman

Joshua Hudgins

Shraddha Kulkarni

Shedrick Mask

Deepa Natarajan

CONTENTS

1.PURPOSE

Team Alpha database supports the transaction of the processing scenarios for the needs of a small company -Bob's Home repairs, owned by Bob. The processing scenarios include, bid, invoice, supply order, customer transaction and supplier transactions. The scope of this project is limited to Bob's small business's needs.

2.TECHNOLOGY

Microsoft SQL Server Express or Developer Editions 2012 to 2017.

3.CASE STUDY

Bob owns a small company called Bob's Home repairs. He does the small home repair jobs that the large companies pass by.

Here is how the business works:

Someone calls Bob and asks him to bid on a job. He drives over, looks at the situation, and gives them a bid. Sometimes it is an official looking bid by mail, and sometimes it is scribbled on notebook paper. He decides how long it will take to do the job (he bills by the hour), how much wood will be needed, any odds and ends that are unique to the job, and an overall price. He moves from job to job and bills customers as he finishes the work.

Bob buys items and supplies from a variety of places, but he buys stuff only when it is needed for a particular project. A potential problem: if he gets behind on his payments to various suppliers, then they won't let him order any more. This would stop his business dead in its tracks. His biggest and most crucial supply is lumber (the price rises and falls constantly). So, he must pay all bills within 30 days of receiving them, especially the lumber companies.

Bob is pretty nice to his customers. They don't have to pay until the work is completed and they are satisfied with his work. This has occasionally led to some problems because the cash coming in is sometimes slower than the cash going out, and he would like to have a better idea of when his bills are due and when his customers will be paying.

Currently, all business records are kept in Bob's head and in one file cabinet. Sometimes he forgets which jobs he bid on and how much he bid on them. He doesn't call potential customers to ask about earlier bids, but this could increase business. He wants reports on suppliers that need to be paid and customers that are slow in paying their bills. So, he wants to computerize these aspects of the business to make things more efficient. Can you help him?

3.1ANALYSIS OF DATA NEEDS

- Bob provides bid to the customer's and estimates the overall price for the work. Customer and bid tables were created to have a traction of these information's.
- Bob buys Supplies from variety of places, a supplier table with the vendor details and supply materials will be tracked using supply item table.
- To handle the payment transactions with customer and suppliers, a supplier payment table and customer payment table have been designed.

4.ENTITY RELATIONSHIP (ERD) DIAGRAM

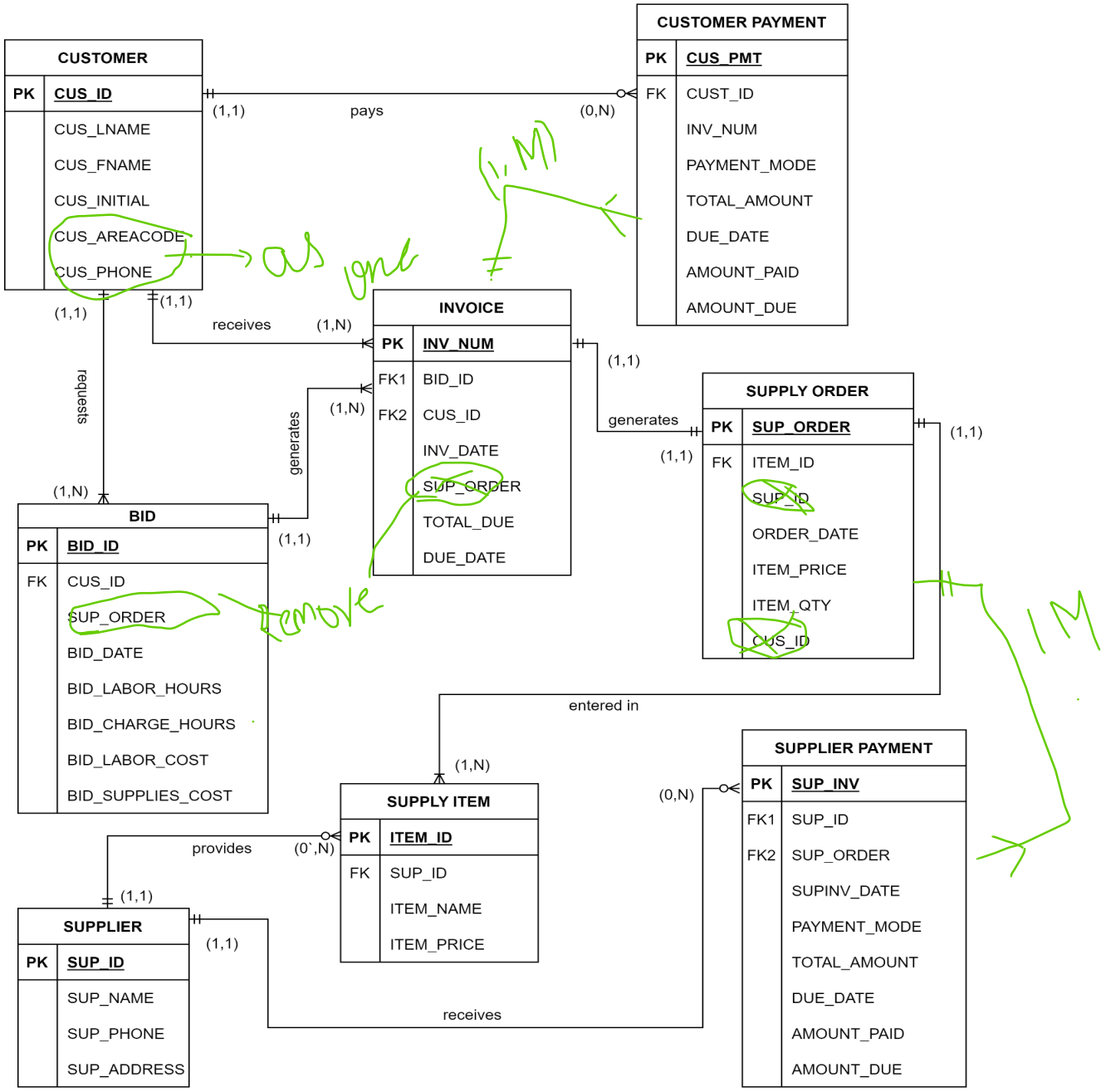


Table	ERD Modifications
customer	merged the phone number attribute
BID	supply order removed/no relationship exist
Invoice	supply order removed/no relationship exist
Supply Order	supp ID and cust id removed -no relation
supplier payment	there is a pk-fk relation , supply order table and supplier payment table, need crow foot
customer payment	pk-fk exists between invoice table and customer payment table, need crow foot

4.1 CARDINALITIES AND BUSINESS RULES

1:M relationship between customer and customer payment

There will be one row in the customer table for any given row in the customer payment table. But there may be many rows in the customer payment table for any given row in the customer table. Each payment made by one and only customer but each customer could have made many payments.

1:M relationship between customer and BID

Each bid is asked by one customer, but one customer can ask for many bids.

1:M relationship between customer and INVOICE

One customer can have many invoices. Each invoice on the other hand belong to one and only one customer.

1:M relationship between BID and invoice

For one bid there may be many invoices. It is under the assumption that customer could agree to work on the bid at different time period, and the invoices are generated accordingly.

1:1 relationship between invoice and supply order

For every invoice order, there will be one supply order. The assumption is that, the customer could buy the supplies for from different vendors, but there will be one supply order per supplier per invoice order.

1:M relationship between supply order and supply item

Each supplier order can have many supply items.

0:m relationship between supplier and supply item

One supplier can supply zero to many items.

0:m relationship between supplier and supplier payment

A supplier can receive zero or many payments from bob home repairs.

5.DATA DICTIONARY

Table Name	CUSTOMER					
Attribute name	Content	Type	Null/not Null	PK or FK	FK Referenced table	Delete Cascade
CUS_ID	customer ID code	int	Not Null	PK		
CUS_LNAME	customer last name	Varchar (15)	Not Null			
CUS_FNAME	customer First name	Varchar (15)	Not Null			
CUS_INITIAL	customer Initial	Varchar (1)	Null			
CUS_PHONE	customer phone number	Varchar (12)	Not Null			
Table Name	**BID**					
Attribute name	Content	Type	Null/not Null	PK or FK	FK Referenced table	Delete Cascade
BID_ID	Bid ID code	int	Not Null	PK		
CUS_ID	customer ID code	int	Not null	FK	CUSTOMER	Active
BID_DATE	Bid issued date	date	Not Null			
BID_LABOR_HRS	Estimated labor hours	float	Not Null			
BID_CHARGE_HRS	Estimated cost per hour	float	Not Null			
BID_LABOR_COST	Total Est. labor cost	Money	not null			
BID_SUPPLIES_COST	Total Est supplies cost	Money	null			
Table Name	**SUPPLIER**					
Attribute name	Content	Type	Null/not Null	PK or FK	FK Referenced table	Delete Cascade
SUP_ID	Supplier ID code	int	not null	PK		
SUP_NAME	supplier Name	VARCHAR (50)	not null			
SUP_PHONE	Supplier Phone number	VARCHAR (50)	not null			
SUP_ADDRESS	Supplier Address	VARCHAR (50)	not null			
Table Name	**INVOICE**					
Attribute name	Content	Type	Null/not Null	PK or FK	FK Referenced table	Delete Cascade
INV_NUM	Invoice number	int	not null	PK		
BID_ID	BID ID code	int	not null	FK	BID	active
CUS_ID	Customer ID code	int	not null	FK	CUSTOMER	
INV_DATE	Invoice Date	date	not null			
TOTAL_DUE	Invoice total	Money	not null			
DUE_DATE	Due date	date	not null			

Table Name	SUPPLYITEM					
Attribute name	Content	Type	Null/not Null	PK or FK	FK Referenced table	Delete Cascade
ITEM_ID	Item code	int	Not Null	PK		
SUP_ID	supplier ID code	int	Not Null	FK	SUPPLIER	
ITEM_NAME	Item name	Varchar (50)	Not Null			
ITEM_PRICE	cost per item	money	Not Null			
Table Name	**SUPPLYORDER**					
Attribute name	Content	Type	Null/not Null	PK or FK	FK Referenced table	Delete Cascade
SUP_ORDER	supply order number	int	not null	PK		
ITEM_ID	Item ID number	int	not null	FK	SUPPLYITEM	
INV_NUM	Invoice number	int	not null	FK	INVOICE	active
ORDER_DATE	Supply Order date	date	not null			
ITEM_PRICE	Cost of the item	money	not null			
ITEM_QTY	number of items	int	not null			
Table Name	**SUPPLIERPAYMENT**					
Attribute name	Content	Type	Null/not Null	PK or FK	FK Referenced table	Delete Cascade
SUP_INV	supplier payment ID	int	not null	PK		
SUP_ID	Supplier ID code	int	not null	FK	SUPPLIER	
SUP_ORDER	supply order code	int	not null	FK	SUPPLYORDER	active
SUPINV_DATE	supply payment ID date	DATE	not null			
PAYMENT_MODE	mode of payment	VARCHAR (3)	not null			
TOTAL_AMOUNT	total amount to pay supplier	money	not null			
DUE_DATE	Due within 30 days	DATE	not null			
AMOUNT_PAID	due paid to supplier	Money	not null			
AMOUNT_DUE	Balance due within 30 days	Money	not null			
Table Name	**CUSTOMERPAYMENT**					
Attribute name	Content	Type	Null/not Null	PK or FK	FK Referenced table	Delete Cascade
CUS_PMT	customer payment ID	int	not null	PK		
CUS_ID	customer Id code	int	not null	FK	CUSTOMER	
INV_NUM	invoice number	int	not null	FK	INVOICE	active
PAYMENT_MODE	mode of payment	VARCHAR (3)	not null			
TOTAL_AMOUNT	Total amount to be paid by customer	money	not null			
DUE_DATE	Due within 30 days	DATE	not null			
AMOUNT_PAID	due paid by customer	Money	not null			
AMOUNT_DUE	Balance due	Money	not null			

6.TABLE DATA

Table Name	CUS_ID	CUS_LNAME	CUS_FNAME	CUS_INITIAL	CUS_PHONE
CUSTOMER	10010	Ramas	Alfred	A	615-844-2573
	10011	Dunne	Leona	k	713-894-1238
	10012	Smith	Kathy	w	615-894-2285
	10013	Ram	Fred	B	615-844-2573
	10014	Don	Lee	s	713-894-1238
	10015	Sean	Kate	z	615-894-2285

Table Name	BID_ID	CUS_ID	BID_DATE	BID_LABOR_HRS	BID_CHARGE_HRS	BID_LABOR_COST	BID_SUPPLIES_COST
BID	1	10010	12/25/2021	5.5	50	275	Null
	2	10011	11/25/2021	10	50	500	Null
	3	10012	1/25/2022	4	50	200	Null
	4	10013	12/20/2021	5.5	50	275	Null
	5	10014	1/27/2022	10	50	500	Null
	6	10015	1/2/2022	4	50	200	Null

Table Name	SUP_ID	SUP_NAME	SUP_PHONE	SUP_ADDRESS
SUPPLIER	1	Flipbug	644-474-5701	393 Vahlen Circle
	2	Youfeed	769-485-7960	6 Nobel Way
	3	Skinte	184-573-3080	941 Veith Lane

Table Name	INV_NUM	BID_ID	CUS_ID	INV_DATE	TOTAL_DUE	DUE_DATE
INVOICE	1	1	10010	1/25/2022	50	2/25/2022
	2	2	10011	1/25/2022	23.99	2/25/2022
	3	3	10012	1/25/2022	45.87	2/25/2022

Table Name	ITEM_ID	SUP_ID	ITEM_NAME	ITEM_PRICE
SUPPLYITEM	32	1	Brass	2.64
	33	2	Rubber	8.33
	34	2	Stone	0.8

Table Name	SUP_ORDER	ITEM_ID	INV_NUM	ORDER_DATE	ITEM_PRICE	ITEM_QTY
SUPPLYORDER	1	32	1	1/13/2022	2.64	10
	2	33	2	1/7/2022	8.33	10
	3	34	3	12/19/2021	0.80	100

	SUP _INV	SUP _ID	SUP _ORDER	SUPINV _DATE	PAYMENT _MODE	TOTAL _AMOUNT	DUE_DATE	AMOUNT _PAID	AMOUNT _DUE
SUPPLIER PAYMENT	1	1	1	1/1/2022	CC	26.50	1/15/2022	20.00	6.50
	3	2	2	12/12/2021	ABW	8.00	12/20/2021	8.00	0.00
	2	3	3	1/20/2020	DC	83.00	1/22/2022	53.00	30.00

Table Name	CUS _PMT	CUS _ID	INV_NUM	PAYMENT _MODE	TOTAL _AMOUNT	DUE _DATE	AMOUNT _PAID	AMOUNT _DUE
SUPPLIERPAYMENT	10	10010	1	CC	50.0	1/14/2022	50.0	0.00
	11	10011	2	ABW	23.99	1/14/2022	23.99	0.00
	12	10012	3	DC	45.87	1/14/2022	20.00	25.87

Please find the Excel for Data Dictionary and Table Data

DataDictionary_Data.xlsx

7. THE SQL SCRIPT

Find below the attached text and SQL files for the script


TeamAplha2.3_week 12


TeamAplha2.3.sql

SECTION A- CREATE DATABASE

```
-- Run the following code to create an empty database called TEAMALPHA
USE master;

-- Drop database
IF DB_ID(N'TEAMALPHA') IS NOT NULL DROP DATABASE TEAMALPHA;

--If database could not be created due to open connections, abort
IF @@ERROR = 3702
   RAISERROR(N'Database cannot be dropped because there are still open connections.', 127, 127) WITH NOWAIT, LOG;

-- Create database
CREATE DATABASE TEAMALPHA;
GO

--use database
USE TEAMALPHA;
GO

--Teamalpha uses the dbo as schema.
```

SECTION B - CREATE TABLES

```
--create table : CUSTOMER

CREATE TABLE CUSTOMER (

CUS_ID int NOT NULL PRIMARY KEY,
CUS_LNAME varchar(15) NOT NULL,
CUS_FNAME varchar(15) NOT NULL,
CUS_INITIAL varchar(1) NULL,
```

```
CUS_PHONE varchar(12)NOT NULL,

);
```

--create table : BID

```
CREATE TABLE BID(
BID_ID int not null PRIMARY KEY,
CUS_ID INT not null ,
BID_DATE DATE NOT NULL,
BID_LABOR_HRS FLOAT not null,
BID_CHARGE_HRS FLOAT not null,
BID_LABOR_COST MONEY not null,
BID_SUPPLIES_COST MONEY NULL
CONSTRAINT fk_CUS_id
   FOREIGN KEY (CUS_id)
   REFERENCES CUSTOMER (CUS_id)
   ON DELETE CASCADE
);
```

--create table : SUPPLIER

```
create table Supplier (

SUP_ID INT PRIMARY KEY not null,
SUP_NAME VARCHAR(50)NOT NULL,
SUP_PHONE VARCHAR(50) NOT NULL,
SUP_ADDRESS VARCHAR(50)NOT NULL,
);
```

--create table : INVOICE

```
CREATE TABLE Invoice(

 INV_NUM INT PRIMARY KEY not null,
BID_ID int not null ,
```

```
CUS_ID  INT FOREIGN KEY REFERENCES customer(cus_id) not null,
INV_DATE DATE not null ,
TOTAL_DUE MONEY not null,
DUE_DATE DATE  not null,
CONSTRAINT fk_BID_ID
   FOREIGN KEY (BID_id)
   REFERENCES BID (BID_id)
   ON DELETE CASCADE
);
```

--create table :SUPPLYITEM

```
CREATE TABLE SupplyItem (

ITEM_ID INT PRIMARY KEY not null,
SUP_ID INT FOREIGN KEY REFERENCES SUPPLIER(SUP_ID)not null,
ITEM_NAME VARCHAR(50) not null,
ITEM_PRICE MONEY not null,
);
```

--create table : SUPPLYORDER

```
CREATE TABLE SupplyOrder
(
SUP_ORDER int PRIMARY KEY not null,
ITEM_ID INT FOREIGN KEY REFERENCES SupplyItem(ITEM_ID)not null,
INV_NUM int  not null,
ORDER_DATE DATE not null,
ITEM_PRICE MONEY not null,
ITEM_QTY INT not null,
CONSTRAINT fk_INV_NUM
   FOREIGN KEY (INV_NUM)
   REFERENCES invoice (INV_NUM)
   ON DELETE CASCADE

);
```

--create table : SUPPLIERPAYMENT

```
CREATE TABLE SupplierPayment
(
SUP_INV  INT PRIMARY KEY not null,
SUP_ID int FOREIGN KEY REFERENCES SUPPLIER(SUP_ID)not null,
SUP_ORDER int not null,
SUPINV_DATE DATE not null,
PAYMENT_MODE VARCHAR(3) not null,
TOTAL_AMOUNT MONEY not null,
DUE_DATE DATE not null,
AMOUNT_PAID MONEY not null,
AMOUNT_DUE MONEY not null,
CONSTRAINT fk_SUP_ORDER
   FOREIGN KEY (SUP_ORDER)
   REFERENCES SupplyOrder(SUP_ORDER )
   ON DELETE CASCADE
);
```

--create table : CUSTOMERPAYMENT

```
CREATE TABLE CUSTOMERPAYMENT (
CUS_PMT INT PRIMARY KEY not null,
CUS_ID int foreign key references customer(cus_ID) not null,
INV_NUM int not null,
PAYMENT_MODE VARCHAR(3),
TOTAL_AMOUNT MONEY,
DUE_DATE DATE,
AMOUNT_PAID MONEY,
AMOUNT_DUE MONEY,
CONSTRAINT fk_invoice_INVNUM
   FOREIGN KEY (INV_NUM)
   REFERENCES INVOICE (INV_NUM)
   ON DELETE CASCADE
```

SECTION C-POPULATE TABLES

--populate data in table : CUSTOMER

```
INSERT INTO CUSTOMER VALUES('10010','Ramas','Alfred','A','615-844-2573');
INSERT INTO CUSTOMER VALUES('10011','Dunne','Leona','K','713-894-1238');
INSERT INTO CUSTOMER VALUES('10012','Smith','Kathy','W','615-894-2285');
INSERT INTO CUSTOMER VALUES('10013','Ram','fred','B','615-844-2573');
INSERT INTO CUSTOMER VALUES('10014','Don','Lee','S','713-894-1238');
INSERT INTO CUSTOMER VALUES('10015','Sean','Kate','Z','615-894-2285');
```

--populate data in table :BID

```
INSERT INTO BID VALUES('001','10010','12/25/2021','5.5','50.00','275.00',NULL);
INSERT INTO BID VALUES('002','10011','11/25/2021','10.0','50.00','500.00',NULL);
INSERT INTO BID VALUES('003','10012','1/25/2022','4.0','50.00','200.00',NULL);
INSERT INTO BID VALUES('004','10013','12/20/2021','5.5','50.00','275.00',NULL);
INSERT INTO BID VALUES('005','10014','1/27/2022','10.0','50.00','500.00',NULL);
INSERT INTO BID VALUES('006','10015','1/02/2022','4.0','50.00','200.00',NULL);
```

--populate data in table :SUPPLIER

```
insert into Supplier  values (1, 'Flipbug', '644-474-5701', '393 Vahlen Circle');
insert into Supplier values (2, 'Youfeed', '769-485-7960', '6 Nobel Way');
insert into Supplier values (3, 'Skinte', '184-573-3080', '941 Veith Lane');
```

--populate data in table :INVOICE

```
INSERT INTO Invoice VALUES(1, 1,'10010','1/25/2022', '50.00', '2/25/2022');
INSERT INTO Invoice VALUES(2, 2,'10011','1/25/2022', '23.99', '2/25/2022');
INSERT INTO Invoice VALUES(3, 3,'10012','1/25/2022', '45.87', '2/25/2022');
```

--populate data in table : SUPPLY ITEM

```
insert into SupplyItem values ('32',1,'Brass', '2.65');
insert into SupplyItem values ('33',2,'Rubber', '8.33');
insert into SupplyItem values ('34',2 ,'Stone', '0.80');
```

--populate data in table : SUPPLYORDER

```
INSERT INTO SupplyOrder VALUES( 1, '32',1, '1-13-2022','2.65','10');
INSERT INTO SupplyOrder VALUES( 2, '33', 2,'1-07-2022','8.33','10' );
INSERT INTO SupplyOrder VALUES( 3, '34',3, '12-19-2021','0.80','100');
```

--populate data in table :SUPPLIERPAYMENT

```
INSERT INTO SupplierPayment  VALUES('1',1,1,'2022-01-01','CC','26.5','2022-01-15','20.00', '6.5');
INSERT INTO SupplierPayment  VALUES('3',2,2,'2021-12-12','ABW','8.00','2021-12-20','8.00', '0.00');
INSERT INTO SupplierPayment VALUES('2',3,3,'2020-01-20','DC','83.00','2022-01-22','53.00', '30.00');
```

--populate data in table :CUSTOMERPAYMENT

```
INSERT INTO CUSTOMERPAYMENT VALUES (10,'10010',1,'CC','50.00','2022-01-14','50.00','0.00');
INSERT INTO CUSTOMERPAYMENT VALUES (11,'10011',2,'ABW','23.99','2022-01-14','23.99','0.00');
INSERT INTO CUSTOMERPAYMENT VALUES (12,'10012',3,'DC','45.87','2022-01-14','20.00','25.87');
```

8. THE SAMPLE QUERIES

Find below the attached SQL and text files for the queries

GP_Draft_1_week12.txt

GP_week12_queries.sql

8.1 LIST THE NAMES OF THE CUSTOMERS WHO WERE PROVIDED BIDS LAST MONTH.

```
Use TEAMALPHA
Select CUS_FNAME,CUS_LNAME, BID_ID, bid_date
from CUSTOMER, BID
where CUSTOMER.CUS_ID=BID.CUS_ID AND CAST(bid.BID_DATE as nchar)Between '2021-01-12' AND '2021-31-12';
```

Results | Messages

	CUS_FNAME	CUS_LNAME	BID_ID	bid_date
1	Alfred	Ramas	1	2021-12-25
2	Leona	Dunne	2	2021-11-25
3	fred	Ram	4	2021-12-20

8.2 LIST THE UNIQUE NAMES OF BOB'S SUPPLIERS.

```
use TEAMALPHA
select Distinct Sup_Name, Sup_ID
From Supplier
Order BY Sup_ID ASC;
```

Results | Messages

	Sup_Name	Sup_ID
1	Flipbug	1
2	Youfeed	2
3	Skinte	3

8.3A LIST THE NAMES OF THE SUPPLIERS AND THE TOTAL AMOUNT OWED TO ANY UNPAID SUPPLIERS.

```
use TEAMALPHA
select sp.Amount_due,s.sup_name,s.sup_id
from SupplierPayment as sp join SUPPLIER as s
on sp.SUP_ID= s.SUP_ID and sp.AMOUNT_DUE > 0.00;
```

Results Messages

	Amount_due	sup_name	sup_id
1	6.50	Flipbug	1
2	30.00	Skinte	3

8.3B THEN WRITE A SEPARATE QUERY TO DISPLAY THE DAYS PAST DUE FOR EACH SUPPLIER.

```
use TEAMALPHA
select s.sup_id,s.sup_name,sp.Amount_due,
sp.DUE_DATE,getDate() as TODAY_DATE ,DATEDIFF(day,sp.due_date,(getDate()-1)) AS NumDaysDue
from SupplierPayment as sp join SUPPLIER as s
on sp.SUP_ID= s.SUP_ID and
sp.AMOUNT_DUE > 0.00;
```

Results Messages

	sup_id	sup_name	Amount_due	DUE_DATE	TODAY_DATE	NumDaysDue
1	1	Flipbug	6.50	2022-01-15	2022-03-01 10:23:41.360	44
2	3	Skinte	30.00	2022-01-22	2022-03-01 10:23:41.360	37

8.4 WRITE AN INSERT STATEMENT OR SERIES OF INSERT STATEMENTS, IF NECESSARY, TO INSERT DATA INTO THE REQUIRED TABLES WHEN A BID IS ENTERED.

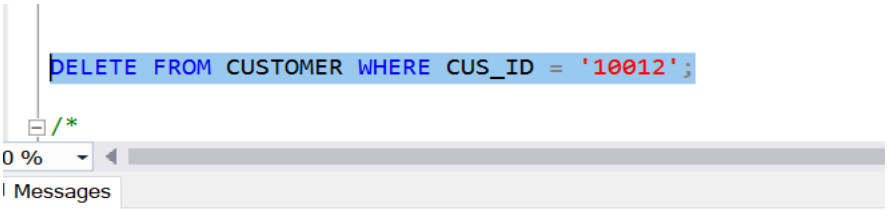
```
use TEAMALPHA

Begin Transaction
 --CUSTOMER DETAILS
INSERT INTO CUSTOMER VALUES ('10016','KENDALL','RICHARDSON','P','615-695-4545');
--BID
INSERT INTO BID VALUES('007','10016','02/01/2022',5,'10.0','50.00','300.00');
--SUPPLIER DETAILS
INSERT INTO Supplier values (4, 'HYPE', '402-699-0778', '1014 S HARBOR LANE');
--INVOICE DETAILS
INSERT INTO Invoice VALUES(4, 4,'10016','02/02/2022','50.00', '2/25/2022');
--SUPPLY ITEMS
INSERT INTO SupplyItem values ('35',4, 'GRANITE', '10.00');
--SUPPLYORDER DETAILS

INSERT INTO SupplyOrder VALUES( 4,'35', 4, '01-14-2022','10.00','5')
--SUPPLIER PAYMENT
INSERT INTO SupplierPayment values ('4',4, 4, '2022-01-20','CC','50.00', '2022-01-20','50.00', '0.00');
--CUSTOMER PAYMENT

INSERT INTO CUSTOMERPAYMENT VALUES (16,'10016','0004','CC','50.00','2022-01-14','50.00','50.00');
commit;
```

```
(1 row affected)

(1 row affected)

Completion time: 2022-03-01T10:24:14.0803939-06:00
```

8.5A WRITE A DELETE STATEMENT(S) TO DELETE A PARTICULAR CUSTOMER FROM THE CUSTOMER

use TEAMALPHA

DELETE FROM CUSTOMER WHERE CUS_ID = '10012';



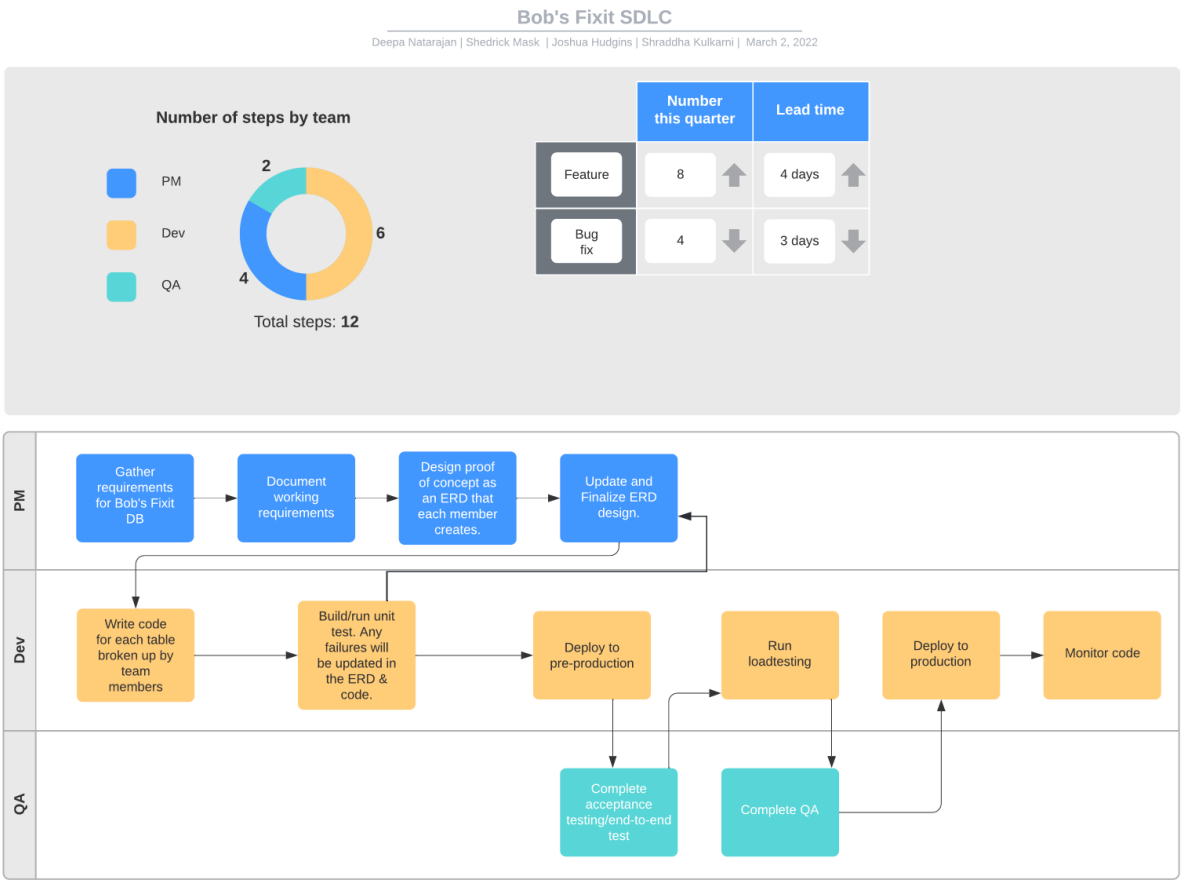
8.5B MAKE SURE TO DELETE ANY CORRESPONDING ROWS OTHER TABLES

use TEAMALPHA
select c.cus_ID ,c.CUS_FNAME,c.CUS_LNAME
from customer as c join bid as b on c.CUS_ID =b.CUS_ID
join invoice as i on b.CUS_ID=i.CUS_ID
join customerpayment as cp on i.CUS_ID=cp.CUS_ID and c.CUS_ID='10012'

9.SUMMARY OF THE OVERALL GROUP PROJECT WORK DONE

To summarize, we created the ERD of the with the individual tables to be used for our design based off the process and need of Bob's Home repair business. We divided the task of the Model design amongst us. Once the tables were combined, team proceeded to script each table so that it could be added to the design. With the script created, we ran tests on our own individual instances using sample data as entities for the tables. Here we encountered issues with missing entities in the tables. We corrected the issue by creating another table and updating the ERD design.

After the major hurdle we worked on fine tuning the workflows, so they reacted according. A last tweak was made to grant the ability to properly remove data and insert them manually. Additional tweaks we created to the model and the database until we felt confident that we have a product ready for deployment.



10.MEMBER EXPERIENCE WITH THE PROJECT

Shraddha- This project gave practical insights towards database design and management. Structure for the database was formed from basics like creating entities, adding Primary and Foreign Keys. SQL queries were inserted in reference to ERD to understand and display the relationship between entities. This was also an opportunity for us to try different SQL queries to get desired results from the database. For me as a part of team, it was an excellent team effort in terms of co-ordination and co-operation. Where there were difficulties other team members were ready to help with the issue and exchange their ideas.

Joshua- I could not imagine completing this project without the team that we had. I feel that as the weeks went on, we really bounced ideas off each other to take what we learned and apply them to make this project work. This project helped me to build upon the basic concepts we learned each week and understand how they fit together to create a working database. I have to say this has been one of the best group projects I have been a part of.

Shedrick- This project helped me grow in my abilities of SQL creation and tackling problems as a team. Everyone was eager to jump in and handle tasks or small hiccups as they came up. That level of teamwork made it very easy to design and deploy the project with very few issues or downtime.

We, as a team were very effective in working together, in communication, organizing the meeting, showing the skills, and completing the work on time. I personally learnt the concepts of DBMS, it has increased my confidence in writing sql queries and designing ERD. The main factor for success of the project is the team's communication, we had zoom meeting, email communications and effective usage of the groups discussion board. My team members were reliable and responsible!!

Thank you all!!