

Cal Poly Pomona Farm Store Online Order Database

Summary

Cal Poly Pomona Farm Store is running by the Cal Poly Pomona as a campus farm store. Due to the COVID-19, more and more customer prefers to order online, and the flooded online order cause the need in keep track of online order and the need of inventory check. Thus, the staff want to build a database to monitor the online order.

Stakeholders

* Customer
* People who purchase goods from the farm store online.
* Staff
* Staff members who are monitoring the online order for the farm store.

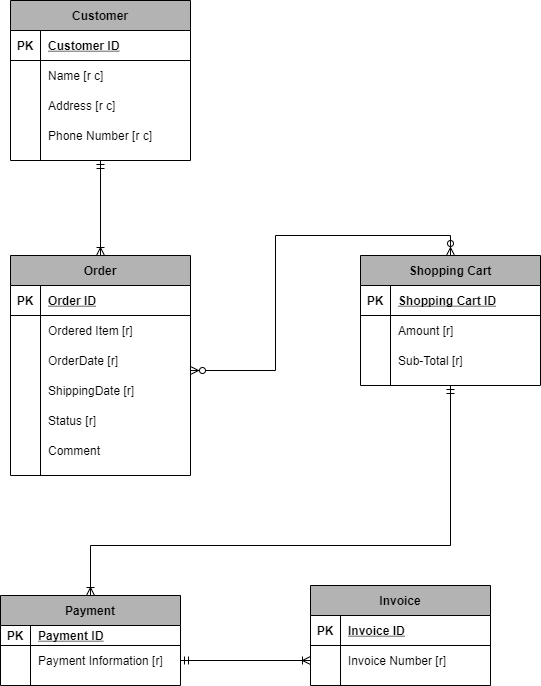
Business Rules

* Customer can have multiple orders.
* Shopping cart can store multiple items per purchase.
* Customer can only have one shopping cart per purchase.
* Invoice are created and sent to our customers individually.

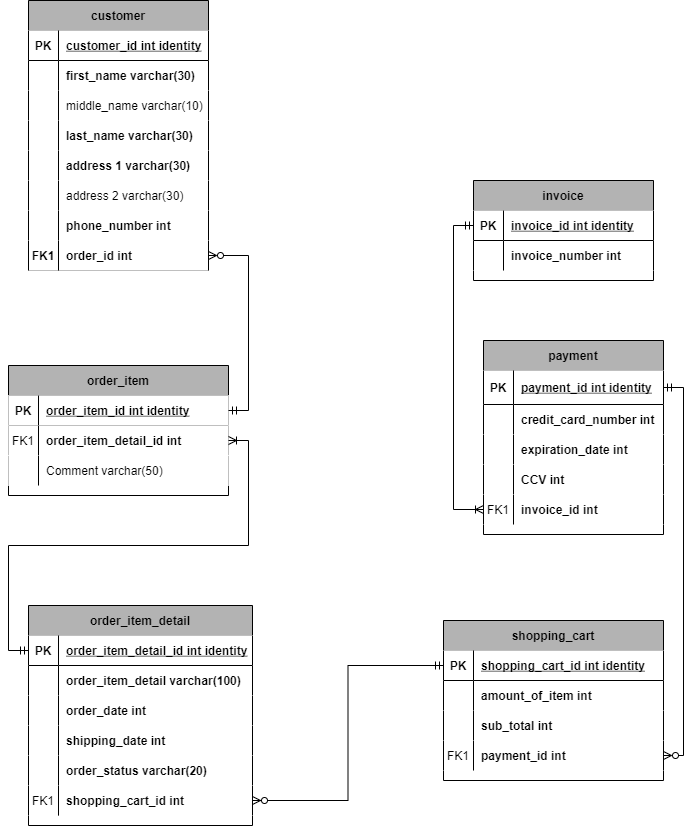
Data Questions

* Who is our customer and what is their contact information?
* Are there any errors on the shipping date?
* What kind of credit card our customer used when shopping online farm store?
* What kind of comments are the customer giving?

Conceptual Model



Normalizer Logical Model



Data Definition Language

CREATE TABLE invoice(

invoice\_id int identity,

invoice\_number int not null,

CONSTRAINT PK\_invoice\_id PRIMARY KEY(invoice\_id)

)

CREATE TABLE payment(

payment\_id int identity,

credit\_card\_number int,

expiration\_date int,

CCV int,

invoice\_id int,

CONSTRAINT PK\_payment PRIMARY KEY(payment\_id),

CONSTRAINT FK1\_payment FOREIGN KEY(invoice\_id) REFERENCES invoice(invoice\_id)

)

CREATE TABLE shopping\_cart(

shopping\_cart\_id int identity,

amount\_of\_item int,

sub\_total int,

payment\_id int,

CONSTRAINT PK\_shopping\_cart PRIMARY KEY(shopping\_cart\_id),

CONSTRAINT FK1\_shopping\_cart FOREIGN KEY(payment\_id) REFERENCES payment(payment\_id)

)

CREATE TABLE order\_item\_detail(

order\_item\_detail\_id int identity,

order\_item\_detail varchar(100),

order\_date datetime not null default GetDate(),

shipping\_date datetime,

order\_status varchar(20),

shopping\_cart\_id int,

CONSTRAINT PK\_order\_item\_detail PRIMARY KEY(order\_item\_detail\_id),

CONSTRAINT FK1\_order\_item\_detail FOREIGN KEY(shopping\_cart\_id) REFERENCES shopping\_cart(shopping\_cart\_id)

)

CREATE TABLE order\_item(

order\_item\_id int identity,

order\_item\_detail\_id int,

comment varchar(50),

CONSTRAINT PK\_order\_item PRIMARY KEY(order\_item\_id),

CONSTRAINT FK1\_order\_item FOREIGN KEY(order\_item\_detail\_id) REFERENCES order\_item\_detail(order\_item\_detail\_id)

)

CREATE TABLE customer (

customer\_id int identity,

first\_name varchar(30) not null,

middle\_name varchar(10),

last\_name varchar(30),

address\_1 varchar(30),

address\_2 varchar(30),

phone\_number int,

order\_item\_id int,

CONSTRAINT PK\_customer PRIMARY KEY(customer\_id),

CONSTRAINT FK1\_customer FOREIGN KEY(order\_item\_id) REFERENCES order\_item(order\_item\_id)

)

Data Manipulation Language

SET IDENTITY\_INSERT invoice ON

INSERT INTO invoice (invoice\_id,invoice\_number)

VALUES

(1,734570382),

(2,369933883),

(3,252556771),

(4,33030326),

(5,670543886),

(6,254493894),

(7,831692887),

(8,898350361),

(9,581552501),

(10,856171711);

SET IDENTITY\_INSERT invoice OFF

SET IDENTITY\_INSERT payment ON

INSERT INTO payment (payment\_id,credit\_card\_number,expiration\_date,CCV,invoice\_id)

VALUES

(1,27622724,CAST(N'2021-10-21 07:14:41' AS DateTime),852,1),

(2,14654191,CAST(N'2022-10-30 04:52:37' AS DateTime),638,2),

(3,92523161,CAST(N'2021-02-21 06:53:51' AS DateTime),388,3),

(4,55292274,CAST(N'2020-12-20 04:33:21' AS DateTime),218,4),

(5,48624572,CAST(N'2022-06-13 00:47:16' AS DateTime),726,5),

(6,26732486,CAST(N'2021-05-28 22:06:25' AS DateTime),104,6),

(7,11810581,CAST(N'2022-10-08 03:39:20' AS DateTime),353,7),

(8,61366650,CAST(N'2021-06-22 21:29:35' AS DateTime),502,8),

(9,55222718,CAST(N'2022-03-11 15:41:42' AS DateTime),988,9),

(10,1188891,CAST(N'2022-02-06 21:39:14' AS DateTime),454,10);

SET IDENTITY\_INSERT payment OFF

SET IDENTITY\_INSERT shopping\_cart ON

INSERT INTO shopping\_cart (shopping\_cart\_id,amount\_of\_item,sub\_total,payment\_id)

VALUES

(1,6,8295,1),

(2,5,4832,2),

(3,5,7643,3),

(4,1,1599,4),

(5,22,7865,5),

(6,17,3199,6),

(7,2,2729,7),

(8,12,8578,8),

(9,41,1686,9),

(10,24,4680,10);

SET IDENTITY\_INSERT shopping\_cart OFF

SET IDENTITY\_INSERT order\_item\_detail ON

INSERT INTO order\_item\_detail (order\_item\_detail\_id,order\_item\_detail,order\_date,shipping\_date,order\_status,shopping\_cart\_id)

VALUES

(1,'Hfd klgbmdbldml montes nascetur ridiculus',CAST(N'2021-02-07 12:40:43' AS DateTime) , CAST(N'2021-02-10 22:00:27' AS DateTime) , 'Transition' ,1),

(2,'aliquet vel vulputate eu odio Phasellus augue id ante', CAST(N'2021-09-05 07:33:25' AS DateTime) , CAST(N'2022-11-11 15:11:21' AS DateTime) , 'Transition' ,2),

(3,'a purus Duis elementum, dui quis accumsan convallis ante lectus', CAST(N'2022-02-12 15:52:56' AS DateTime) , CAST(N'2022-08-08 03:17:18' AS DateTime) , 'Arrived' ,3),

(4,'aliquet Phasellus fermentum convallis ligula Donec luctus aliquet odio Etiam' , CAST(N'2021-12-23 12:49:21' AS DateTime) , CAST(N'2021-09-01 03:34:17' AS DateTime) , 'Arrived' ,4),

(5,'id ante Nunc maurbngfis sapien cursus in, hendrerit consectetuer, cursus' , CAST(N'2022-03-07 23:40:24' AS DateTime) , CAST(N'2022-08-25 02:07:55' AS DateTime) , 'Arrived' ,5),

(6,'ut nisi a odio semper cursus mollis tincidunt ', CAST(N'2022-09-26 12:52:00' AS DateTime) , CAST(N'2021-07-24 01:00:46' AS DateTime) , 'Transition' ,6),

(7,'ac arcu Nunc mauris Morbi non sapien molestie orci tincidunt' , CAST(N'2021-03-18 16:38:17' AS DateTime) , CAST(N'2021-10-13 04:35:03' AS DateTime) , 'Transition' ,7),

(8,'semper dui lectus rutrum urna, nec luctus felis purus ac ', CAST(N'2021-07-28 19:25:45' AS DateTime) , CAST(N'2021-10-05 11:59:17' AS DateTime) , 'Arrived' ,8),

(9,'sit amet diam eu dolor egestas rhoncus Proin nisl sem' , CAST(N'2021-03-16 03:46:29' AS DateTime) , CAST(N'2021-01-12 22:07:05' AS DateTime) , 'Pending' ,9),

(10,'ultrices mauris ipsum porta elit a feugiat tellus lorem eu' , CAST(N'2022-11-09 11:56:42' AS DateTime) , CAST(N'2021-03-03 15:05:41' AS DateTime) , 'Transition' ,10);

SET IDENTITY\_INSERT order\_item\_detail OFF

SET IDENTITY\_INSERT order\_item ON

INSERT INTO order\_item (order\_item\_id,order\_item\_detail\_id,comment)

VALUES

(1,1,'elit. Curabitur sed tortorc'),

(2,2,'Phasellus at augue i cursus. Nunc mauris elit,'),

(3,3,'fringilla, porttitor vsollicitudin'),

(4,4,'enim, sit ametarcu. Morbi sit'),

(5,5,'malesuada ut, sem. sellus in felis.'),

(6,6,'ac libero nec ligul dui, semper'),

(7,7,'iaculis odio. Nam interan eget metus.'),

(8,8,'metus vitae velit eget sed consequat'),

(9,9,'volutpat. Nulla dignissim. feugiat placerat'),

(10,10,'eros non enim tor tellus non magna.');

SET IDENTITY\_INSERT order\_item OFF

SET IDENTITY\_INSERT customer ON

INSERT INTO customer (customer\_id,first\_name,middle\_name,last\_name,address\_1,address\_2,phone\_number,order\_item\_id)

VALUES

(1,'Russell','Williams','Jescie','9136 Facilisis Rd.','Ap #207-6227 Erat St.','54686702',1),

(2,'Virginia','Long','Samson','2585 Eget St.','508-5772 Ac Street','12441495',2),

(3,'Kellie','Logan','Martena','Ap #483-8611 Et Ave','Ap #401-9651 Interdum Rd.','185274593',3),

(4,'Rebekah','Whitaker','Mark','777-7516 Integer Av.','Ap #371-6277 Maecenas Av.','37522153',4),

(5,'Dante','Harper','Yoshi','Ap #943-4026 Dictum Av.','3217 Quisque St.','55583423',5),

(6,'Dacey','Yang','Jacob','771-2999 Risus. St.','Ap #367-9656 Enim, St.','85714352',6),

(7,'Ray','Price','Burton','2835 Dictum Road','5687 Lobortis Rd.','172805913',7),

(8,'Tana','Henson','Mechelle','Ap #399-8227 Scelerisque Road','555946 Consectetuer Avenue','2632641',8),

(9,'Clark','Curry','Jacob','9020 Cum St.','690-9295 Odio, Avenue','194287377',9),

(10,'Christian','Peters','Lewis','8442 Nullam Road','Ap #464-8391 Lorem. St.','162348662',10);

SET IDENTITY\_INSERT customer OFF

Programming Objects

This is a view where it view how much invoices are in the database.

CREATE OR ALTER VIEW invoiceCount AS

SELECT COUNT(invoice\_id) as invoice\_number FROM invoice

SELECT \* FROM invoiceCount



* This procedure is made to help change customer change their new phone number.

CREATE PROCEDURE PhoneNumberChange (@firstname varchar(30), @phonenum int)

AS

BEGIN

UPDATE customer SET phone\_number = @phonenum

WHERE first\_name = @firstname

END

GO

* In the below code, I change Rusell’s phone number using the procedure I made.

EXEC PhoneNumberChange 'Russell', '9238472'

SELECT \* FROM customer WHERE first\_name = 'Russell'



CREATE FUNCTION dbo.commentlookup(@itemid int)

RETURNS int AS

BEGIN

DECLARE @returnValue int

SELECT @returnValue = customer\_id from customer

WHERE order\_item\_id = @itemid

RETURN @returnValue

END

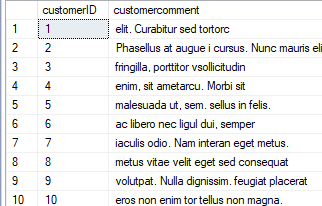
GO

SELECT

dbo.commentlookup(order\_item\_id) as customerID

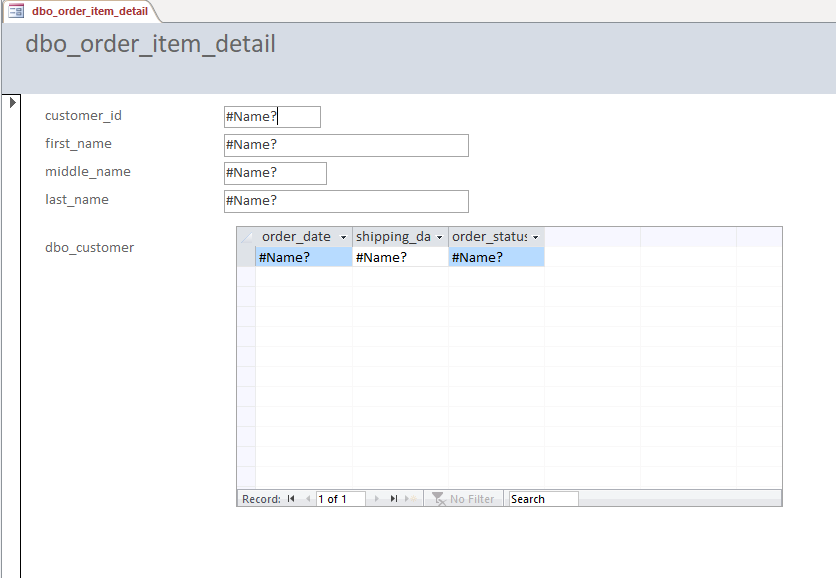
,order\_item.comment as customercomment

FROM order\_item

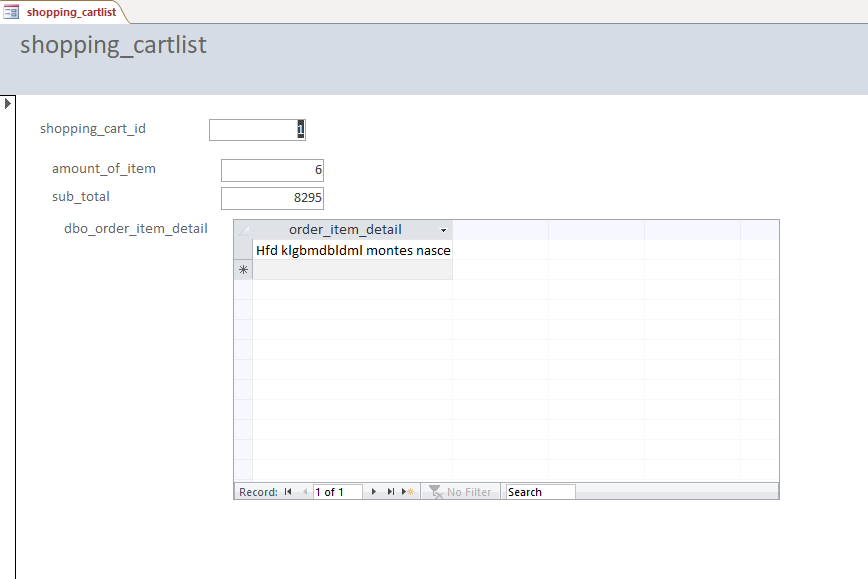


* This function is made for looking up the comment for the order.

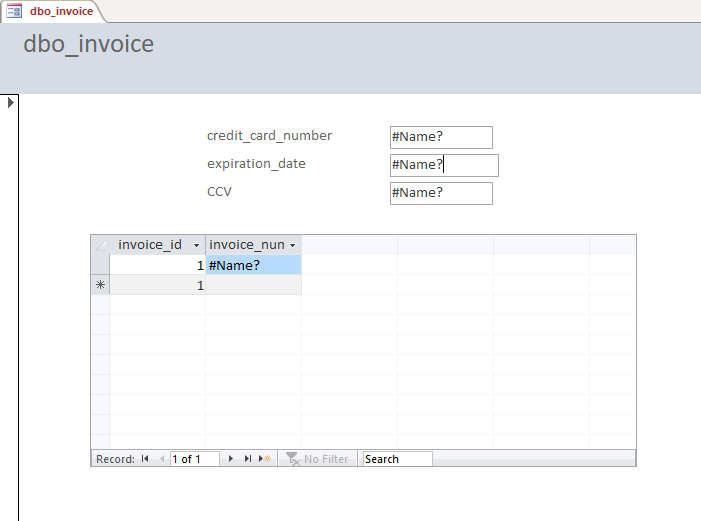
User Interface



* This is an interface for customer to search their ordered items.



* This is an interface for customer to search their shopping cart.



* This is the paying interface for customer to use.

Reflection

I assume that the data base coding does not take any credit of it, but after coding and using the knowledge in class. I realize that the database actually related to the database and the design of it can de determine by the database relationship. If I have a chance to go through another SQL project, I will be more prepared and experienced in the coding and exporting to other platform.

Summary

The Cal Poly Online Farm Store database is made for customer that still want to purchase the grocery from the Cal Poly Farm store during COVID-19 period. In the database, there are some functions are added to make the staff and customer access the database faster and more convenient. We can now change customer’s phone number using the procedure PhoneNumberChange, this will change customer’s phone number by their first name. Also, the function dbo.commentlookup allow us to look up customer’s comment when they order the item online. These change help the staff and customer to have a greater experience when using our database or accessing our database.