Project Implementation Report:

Ecommerce website

Syracuse University
School of Information Studies

Abstract:

Ecommerce is the solution to simplify shopping experience for those involved in the transaction. Ecommerce websites are being launched competitively to provide users the ease of buying and selling from the comfort of their own home. This implementation will serve as a platform for customers to sign up using an authorized login page, browse through various categories of products, get detailed information like cost, other user ratings and reviews, warranty, product specification, a page to store preferred payment method, option to purchase multiple products and preview in the cart, get seller information, an admin login privilege to the owner of the website that would enable the user to keep a track of the sellers and customers.

Table of Contents

Project Summary

The proposed implementation categorizes under the ecommerce industry. Ecommerce business outlines a platform for buying or selling of products that uses digital information of the users involved and provides the ability to perform electronic communications in transactions that establishes a relationship between the users that could be individuals or organizations. In the current global scenario, most of the transactions including buying or selling are preferred to be conducted via an online platform rather than traditional trade means, ecommerce provides the users flexibility to conduct trade without having to interact with each other, ease of placing orders at their convenience and the option to make and receive payments online, the organizations can supply products all over the world, advertise their products, selling products at competitive prices are some of the significant features of an ecommerce industry.

The current project deals with a business-to-consumer ecommerce business, i.e., the transactions involved would involve a customer and an organization. It is essential to have customer information like the shipping address and billing information, product information pertaining to the type of product, cost, and rating of a specific product, data about the organization supplying the product, its ratings, reviews and features. On a holistic level, the database would contain information about the order, products, customers, sellers, category of products, shipping methods and payment records. The most popular questions that arise while purchasing any product are related to the warranty, quality, and cost of the products which will be processed and stored in the database implementation.

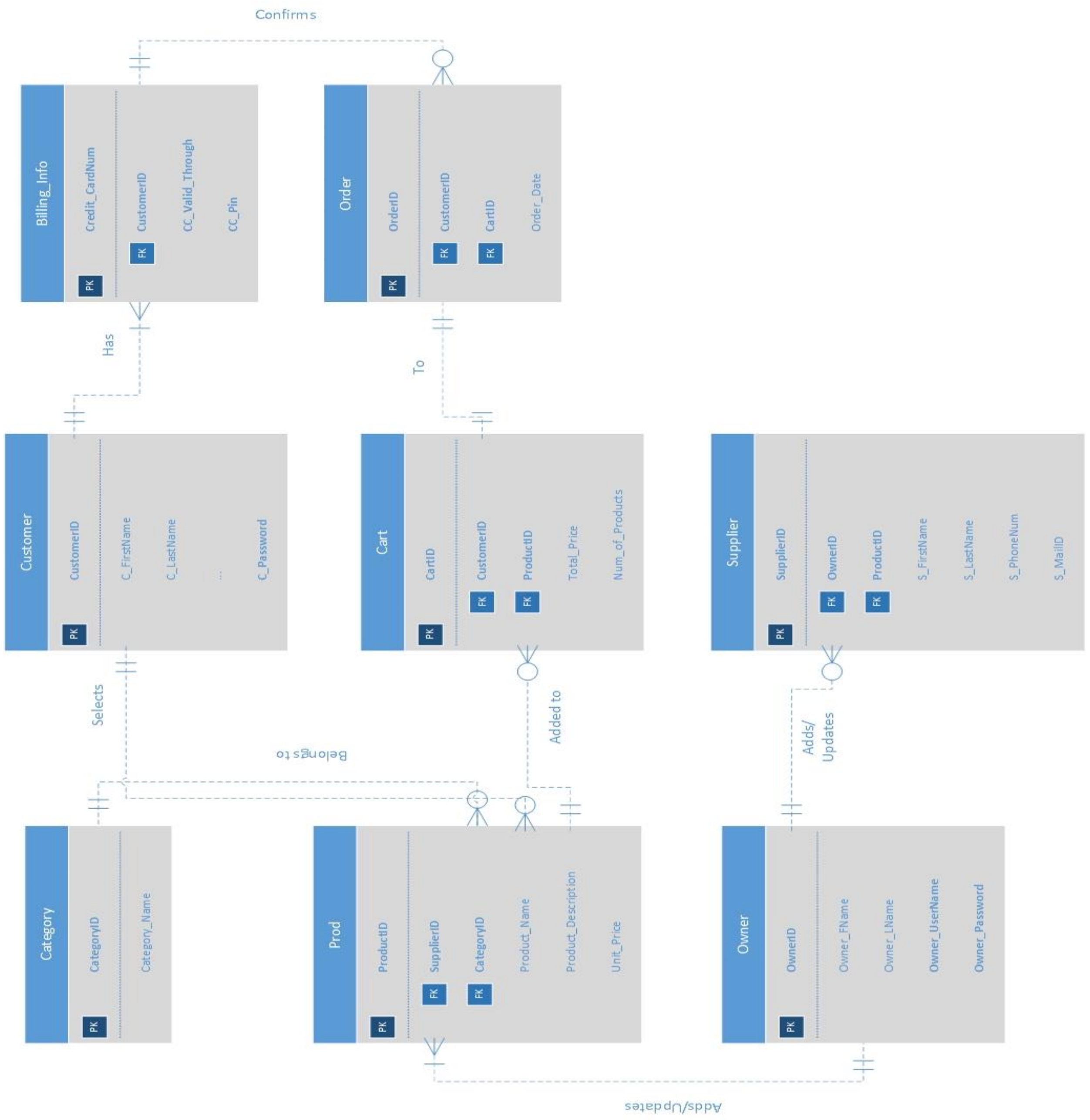
Tables and Attributes

Data object	
Category	Contains classification of products
CategoryID	PRIMARY KEY: Every category has a unique ID that identifies a specific category of products stored under it.
Category_Name	
Product	Contains detail of every product added by the owner of the ecommerce website
ProductID	PRIMARY KEY: Every product has a unique product ID which can be used to find a specific product in the list.
CategoryID	FOREIGN KEY: Associated with primary key of ‘Product’ table.
SupplierID	FOREIGN KEY: Associated with primary key of ‘Supplier’ table.
Product_Name	
Product_Description	
Unit_Price	
Owner	Maintains record of owner
OwnerID	PRIMARY KEY: There is a specific owner ID that is used as an index for the owner entity.
Owner_FName	
Owner_LName	
Owner_Password	
Owner_Username	
Customer	Maintains record ofcustomers
CustomerID	PRIMARY KEY: Every customer has a unique customer ID that is generated whilst a new account is created.
C_FirstName	
C_LastName	
C_PhoneNum	
C_MailID	
C_Address	
C_PostalCode	
C_City	
C_Country	
C_UserName	

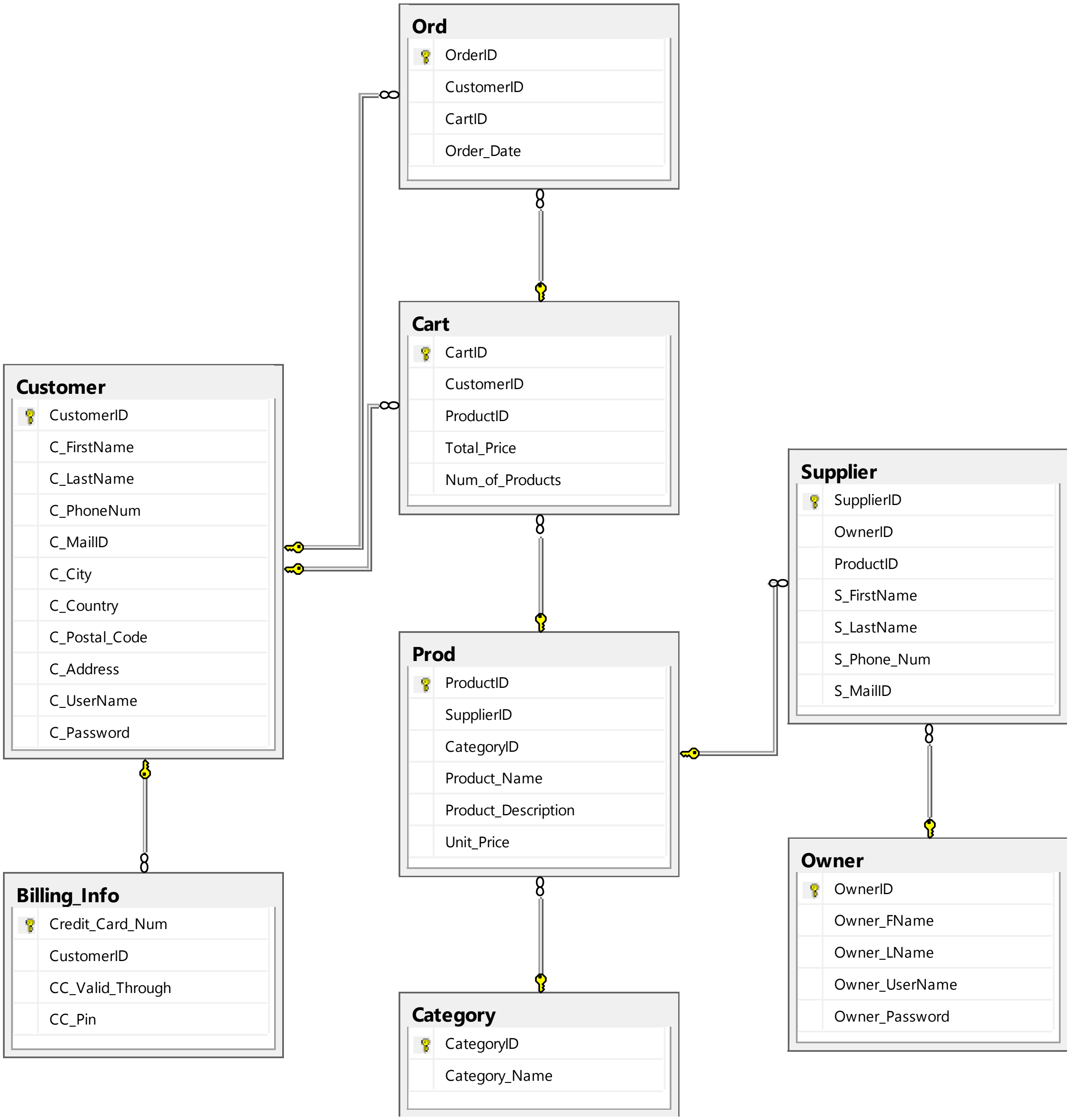
C_Password	
Supplier	Contains details of the suppliers added by the owner of the ecommerce website
SupplierID	PRIMARY KEY: Every supplier has a unique ID that is assigned by the owner of the ecommerce website.
OwnerID	FOREIGN KEY: associated with primary key of ‘Owner’ table.
S_FirstName	
S_LastName	
S_PhoneNum	
S_MailID	
Billing_Info	Contains the record of payment information of every individual customer
Credit_CardNum	PRIMARY KEY: Every customer enters a unique credit card number as a payment method.
CustomerID	FOREIGN KEY: Associated with primary key of ‘Customer’ table.
CC_ValidThrough	
CC_Pin	
Cart	Contains details of every product added by the customer for a particular order
CartID	PRIMARY KEY: Every cart has a unique ID against a specific order placed by the customer.
CustomerID	FOREIGN KEY: Associated with primary key of ‘Customer’ table.
ProductID	FOREIGN KEY: Associated with primary key of ‘Product’ table.
Total_Price	
Num_Of_Products	
Order	Contains details of every order placed by a customer
OrderID	PRIMARY KEY: Every order has a unique ID to identify a specific order placed by the customer.
CartID	FOREIGN KEY: Associated with primary key of ‘Customer’ table.
CustomerID	FOREIGN KEY: Associated with the primary key of ‘Cart’ table.
Order_Date	

Entity Relationship Diagram

Visio



SQL


Ord
OrderID
CustomerID
CartID
Order_Date
Cart
CartID
CustomerID
ProductID
Total_Price
Num_of_Products
Customer
CustomerID
C_FirstName
C_LastName
C_PhoneNum
C_MailID
C_City
C_Country
C_Postal_Code
C_Address
C_UserName
C_Password
Supplier
SupplierID
OwnerID
ProductID
S_FirstName
S_LastName
S_Phone_Num
S_MailID
Prod
ProductID
SupplierID
CategoryID
Product_Name
Product_Description
Unit_Price
Billing_Info
Credit_Card_Num
CustomerID
CC_Valid_Through
CC_Pin
Owner
OwnerID
Owner_FName
Owner_LName
Owner_UserName
Owner_Password
Category
CategoryID
Category_Name

Business Rules

- Currently the system has only two types of users 'owner' and 'customer', the 'supplier' entity table is view only.
- The customer has to create an account before purchasing anything, no guest checkout is allowed.
- The ecommerce website is supposed to have only a single owner who will manage all the products list and product supplier.
- The owner has no access on the 'category' list, it is already predefined and can be only subcategorized into new products.
- The customer cannot edit/update the products placed for a particular order.
- The owner of the ecommerce website can add/update/delete the product supplier information, the supplier has no access to the website.
- The customer can neither track the order placed nor can return or request for a refund from the owner of the ecommerce website.

Database Infrastructure

- Owner and Customer logs in using their respective login credentials.
- If the customer is new to the website s/he has to create an account, before purchasing any product.
- The owner can add/update/delete the details of product 'supplier' entity.
- The owner can add/update/delete product from the 'product' table listed as per the predefined values of 'category' entity.
- Before making the purchase, the customer has to add their billing information from which they can make a payment, it basically contains the credit card details.
- After viewing the products from the website, the customer can add the products to the cart and proceed to the final checkout step.
- After making a successful transaction for each order, the customer can view the particular order and its details.

SQL Scripts for Creating and Inserting Sample Data

```
-- Strong entities
create table Category
(
        CategoryID char(20) not null,
        Category_Name varchar(20) not null,
        constraint Category_PK primary key(CategoryID)
);

create table Owner
(
        OwnerID char(20) not null,
        Owner_FName varchar(20) not null,
        Owner_LName varchar(20) not null,
        Owner_UserName varchar(20) not null,
        Owner_Password varchar(20) not null,
        constraint Owner_PK primary key(OwnerID)
);

create table Customer
(
        CustomerID char(20) not null,
        C_FirstName varchar(20) not null,
        C_LastName varchar(20) not null,
        C_PhoneNum int not null,
        C_MailID varchar(20) not null,
        C_City varchar(20) not null,
        C_Country varchar(20) not null,
        C_Postal_Code int not null,
        C_Address varchar(20) not null,
        C_UserName varchar(20) not null,
        C_Password varchar(20) not null,
        constraint Customer_PK primary key(CustomerID)
);

-- Weak Entities
create table Billing_Info
(
        Credit_Card_Num int not null,
        CustomerID char(20) not null,
        CC_Valid_Through date not null,
        CC_Pin int not null,
        constraint Credit_Card_Num_PK primary key(Credit_Card_Num),
        constraint Billing_Info_FK_CustomerID foreign key(CustomerID) references
Customer(CustomerID)
);

create table Supplier
(
        SupplierID char(20) not null,
        OwnerID char(20) not null,
        ProductID char(20) not null,
        S_FirstName varchar(20) not null,
        S_LastName varchar(20) not null,
        S_Phone_Num int not null,
        S_MailID varchar(30) not null,
        constraint Supplier_PK primary key(SupplierID),
        constraint Supplier_FK_OwnerID foreign key(OwnerID) references Owner(OwnerID),
        constraint Supplier_FK_ProductID foreign key(ProductID) references Prod(ProductID)
);

create table Prod
(
        ProductID char(20) not null,
        CategoryID char(20) not null,
        Product_Name varchar(40) not null,
        Product_Description varchar(80),
        Unit_Price int not null,
        constraint Product_PK primary key(ProductID),
        constraint Product_FK_CategoryID foreign key(CategoryID) references
Category(CategoryID)
);
```

```
create table Cart
(
        CartID char(20) not null,
        CustomerID char(20) not null,
        ProductID char(20) not null,
        Total_Price int not null,
        Num_of_Products int not null,
        constraint Cart_PK primary key(CartID),
        constraint Cart_FK_CustomerID foreign key(CustomerID) references 
Customer(CustomerID),
        constraint Cart_FK_ProductID foreign key(ProductID) references Prod(ProductID)
);

create table Ord
(
        OrderID char(20) not null,
        CustomerID char(20) not null,
        CartID char(20) not null,
        Order_Date date not null,
        constraint Order_PK primary key(OrderID),
        constraint Order_FK_CustomerID foreign key(CustomerID) references 
Customer(CustomerID),
        constraint Order_FK_CartID foreign key(CartID) references Cart(CartID)
);

insert into Category(CategoryID, Category_Name)
values(101, 'Fashion');
insert into Category(CategoryID, Category_Name)
values(102, 'Home & Decor');
insert into Category(CategoryID, Category_Name)
values(103, 'Accesories');
insert into Category(CategoryID, Category_Name)
values(104, 'Electronics');
insert into Category(CategoryID, Category_Name)
values(105, 'Groceries');

insert into Prod(ProductID, SupplierID, CategoryID, Product_Name, Product_Description, 
Unit_Price)
values(10, 1001, 101, 'Clothes', 'Mens wear', 12.48);
insert into Prod(ProductID, SupplierID, CategoryID, Product_Name, Product_Description, 
Unit_Price)
values(11, 1002, 102, 'Home & Decor', 'Frames', 37.21);
insert into Prod(ProductID, SupplierID, CategoryID, Product_Name, Product_Description, 
Unit_Price)
values(12, 1003, 103, 'Accesories', 'Belt', 22.21);
insert into Prod(ProductID, SupplierID, CategoryID, Product_Name, Product_Description, 
Unit_Price)
values(13, 1004, 104, 'Electronics', 'Mobile Phone', 422.21);
insert into Prod(ProductID, SupplierID, CategoryID, Product_Name, Product_Description, 
Unit_Price)
values(14, 1005, 105, 'Groceries', 'Sugar', 3.25);

insert into Owner(OwnerID, Owner_FName, Owner_LName, Owner_UserName, Owner_Password)
values(1, 'Vaibhav', 'Nigam', 'vnigam', 'vnigam');

insert into Customer(CustomerID, C_FirstName, C_LastName, C_PhoneNum, C_MailID, C_City, 
C_Country, C_Postal_Code, C_Address, C_UserName, C_Password)
values(01, 'Ankita', 'Nagar', 3154671, 'anagar@gmail.com', 'Mumbai', 'India', '208025',
'1021 Lancaster Ave', 'anagar', 'anagar');
insert into Customer(CustomerID, C_FirstName, C_LastName, C_PhoneNum, C_MailID, C_City,
C_Country, C_Postal_Code, C_Address, C_UserName, C_Password)
values(02, 'Shubham', 'Bhatia', 3154637, 'shbhatia@gmail.com', 'Syracuse', 'USA',
'10301', '103 Victoria Place', 'shbhatia', 'shbhatia');
insert into Customer(CustomerID, C_FirstName, C_LastName, C_PhoneNum, C_MailID, C_City,
C_Country, C_Postal_Code, C_Address, C_UserName, C_Password)
values(03, 'Lakshay', 'Swani', 3151234, 'lswani@gmail.com', 'Noida', 'India', '201301',
'Sector 36', 'lswani', 'lswani');
insert into Customer(CustomerID, C_FirstName, C_LastName, C_PhoneNum, C_MailID, C_City,
C_Country, C_Postal_Code, C_Address, C_UserName, C_Password)
values(04, 'Dipti', 'Sikka', 3154321, 'dsikka@gmail.com', 'Delhi', 'India', '201304',
'NCC', 'dsikka', 'dsikka');

insert into Billing_Info(Credit_Card_Num, CustomerID, CC_Valid_Through, CC_Pin)
values(998877, 01, '2020-08-22', 693);
insert into Billing_Info(Credit_Card_Num, CustomerID, CC_Valid_Through, CC_Pin)
values(665544, 02, '2022-02-13', 847);

insert into Supplier(SupplierID, OwnerID, ProductID, S_FirstName, S_LastName,
S_Phone_Num, S_MailID)
values(20, 1, 10, 'Rmoil', 'Godha', 3154480, 'rgodha@gmail.com');
insert into Supplier(SupplierID, OwnerID, ProductID, S_FirstName, S_LastName,
S_Phone_Num, S_MailID)
values(21, 1, 11, 'Pranay', 'Lulla', 3154648, 'plulla@gmail.com');
insert into Supplier(SupplierID, OwnerID, ProductID, S_FirstName, S_LastName,
S_Phone_Num, S_MailID)
values(22, 1, 12, 'Tushar', 'Sharma', 312675, 'tsharma@gmail.com');

insert into Cart(CartID, CustomerID, ProductID, Total_Price, Num_of_Products)
values(201, 01, 10, 166.21, 7);
insert into Cart(CartID, CustomerID, ProductID, Total_Price, Num_of_Products)
values(202, 02, 11, 57.83, 3);
insert into Cart(CartID, CustomerID, ProductID, Total_Price, Num_of_Products)
values(203, 03, 12, 49.2, 5);
insert into Cart(CartID, CustomerID, ProductID, Total_Price, Num_of_Products)
values(204, 04, 13, 62.1, 8);

insert into Ord(OrderID, CustomerID, CartID, Order_Date)
values(301, 01, 201, '2018-04-15');
insert into Ord(OrderID, CustomerID, CartID, Order_Date)
values(302, 02, 202, '2018-11-12');
```

Major data questions:

The users using the system are of two types,

- The owner
- The customer

Why owner queries the database:

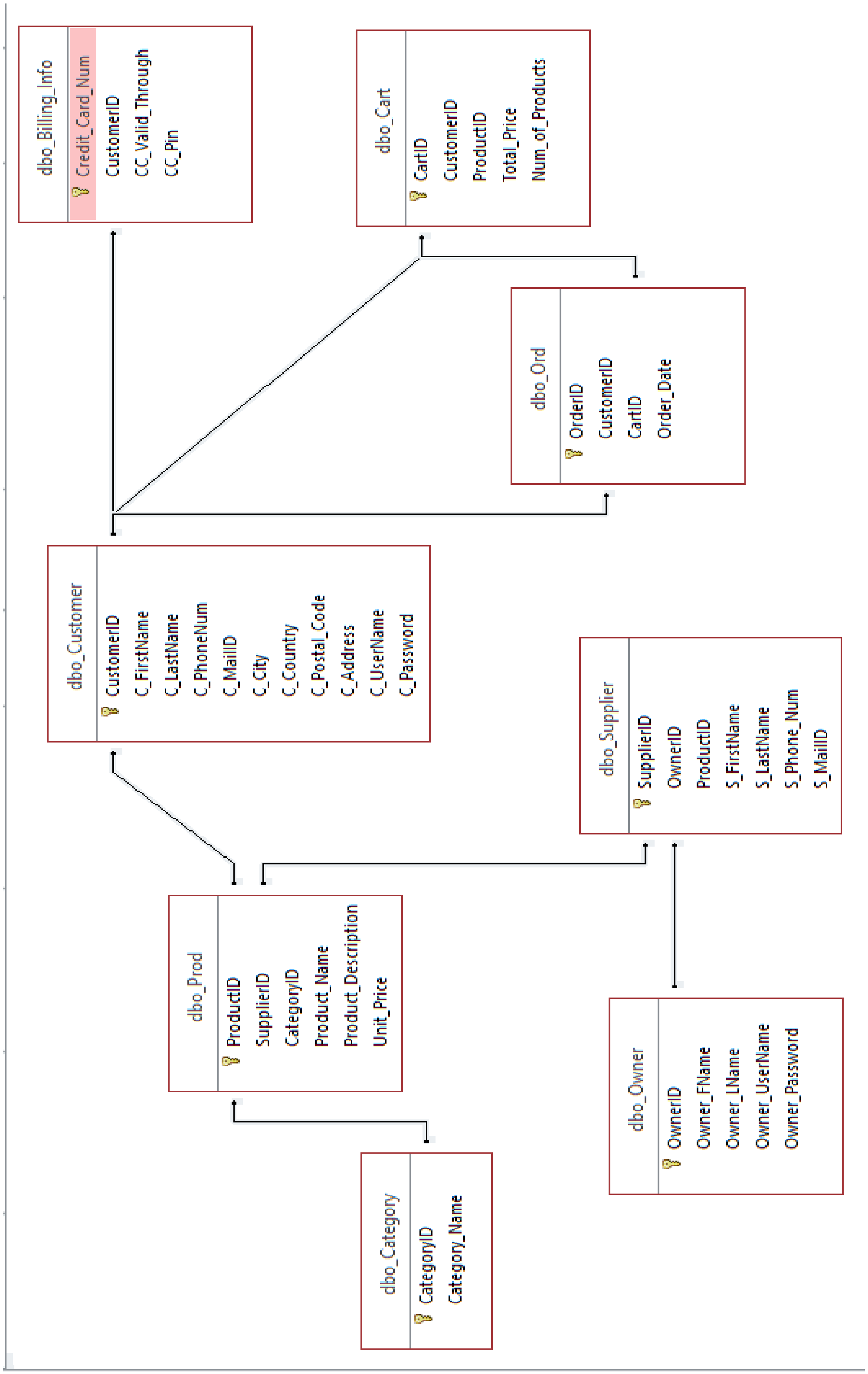
- **Add/Update supplier account**
  Whenever the owner adds a new product supplier or updates the existing supplier data, the ‘Supplier’ table is updated. The owner has no restrictions on updating or deleting the supplier account, in fact the owner can delete any supplier account.
- **Add/Update productlist**
  When a new product is added by the owner the product list is updated in the specific category,the corresponding‘productID’shouldalsobeupdatedinthecategorytable.

Why customer queries the database:

- **Add/update billing information**
  When a customer wants to place an order, s/he needs to add billing information to checkout and successfully place an order. The customer can add credit card details or update the existing card details. The ‘Billing_info’ table will be modified after a customer enters the respective billing information.
- **View and add product to the cart**
  This list can be obtained by getting a list from ‘Product’ table, this list can be further used by the customer to select products from different categories and add them to cart.
  The ‘ProductID’ will be reflected against every product added by the customer in the cart.
- **View order**
  The products added to the cart by the customer can be viewed by the customer, the ‘cartID’ will be used to fetch and view the products contained in a cart for a specific order.

Relationship Diagram

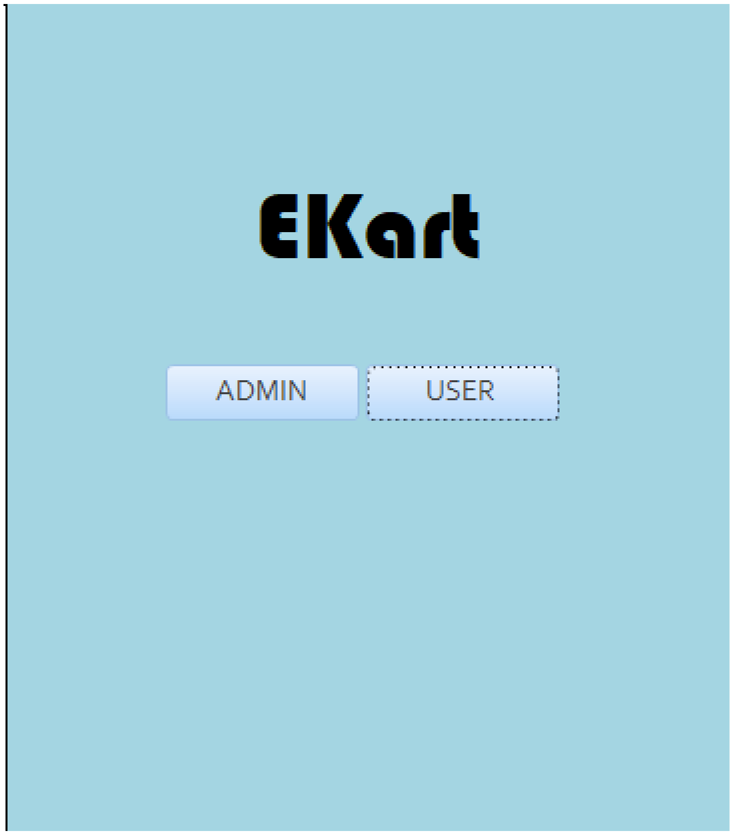
Microsoft Access



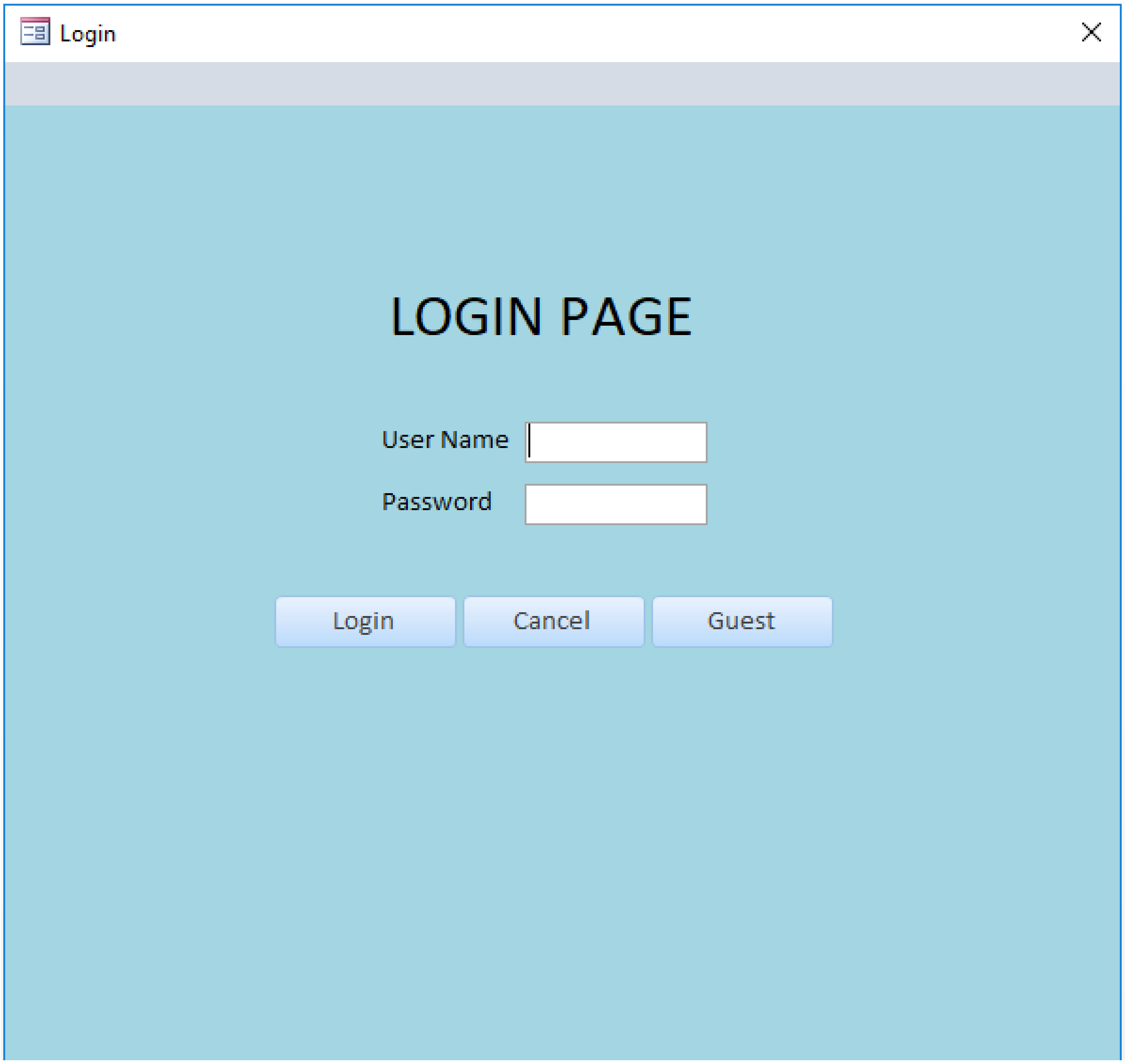
Forms

Total number of forms: 10
Total number of tables: 5
Total number of forms that I created so far: 10
Selection window: Weather the person is an admin or a user
Admin/User login page(Y): Login to the system
Add/Update product window(Y): Admin rights to add/update product
Add/Update Seller/Supplier(Y): Admin rights to add/update seller/supplier
User registration window(Y): A new user can register
View product window(Y): A user can view different products and checkout

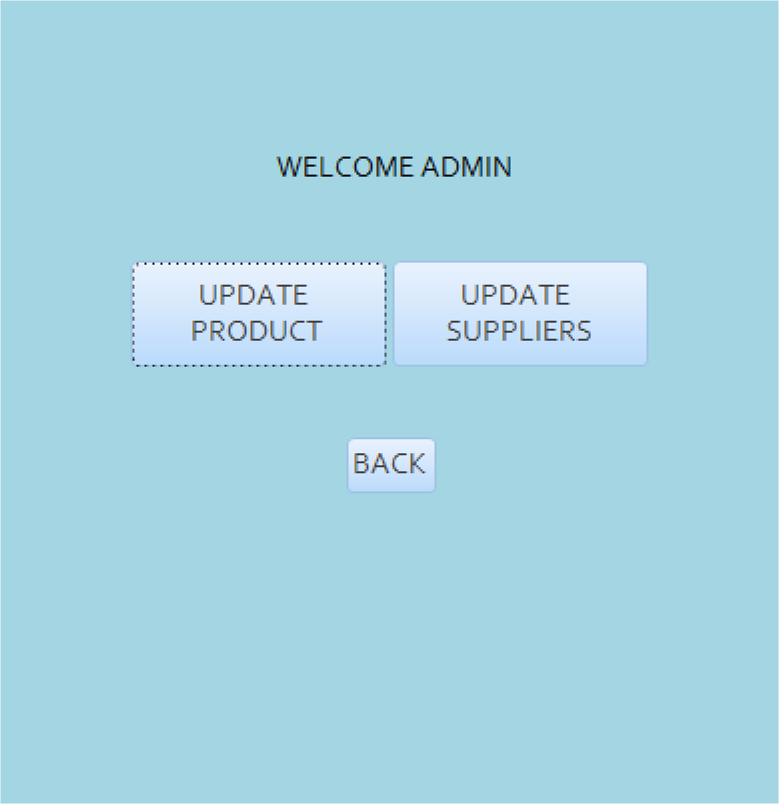
Form (Start Window)



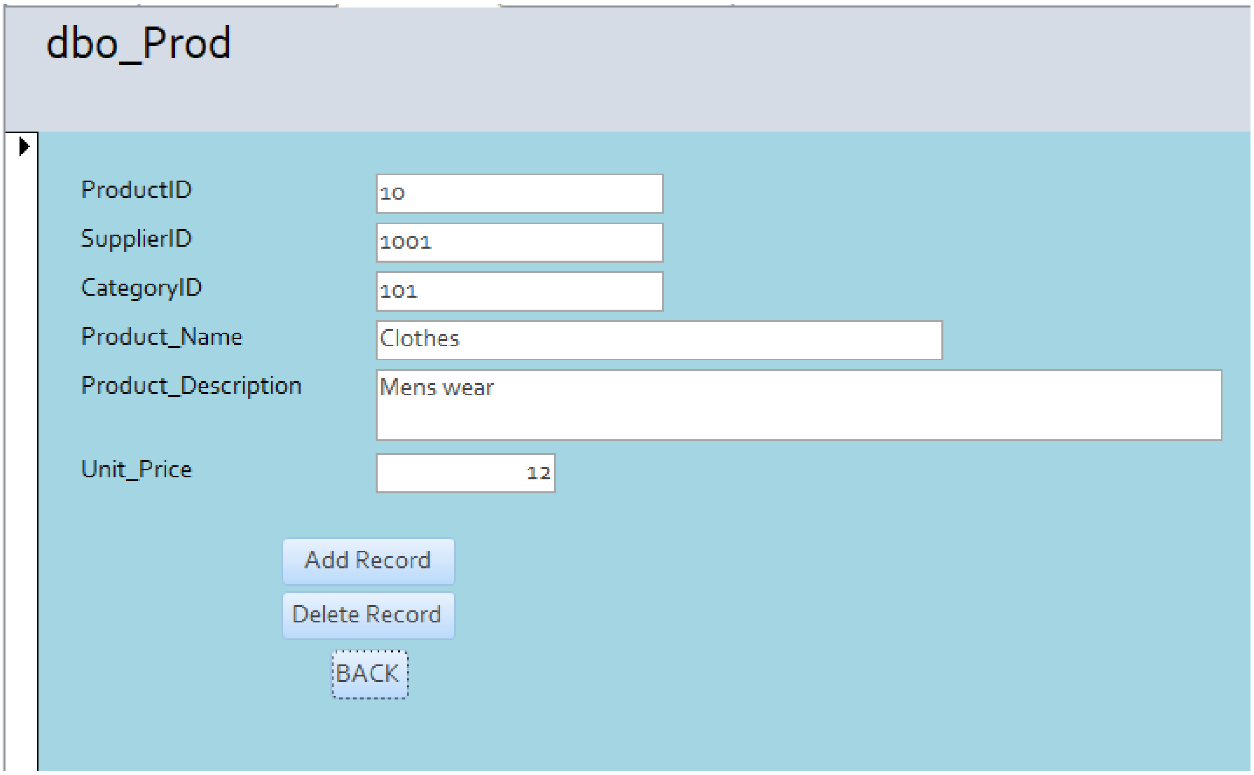
Form (Login Window)



Form (Admin Window)



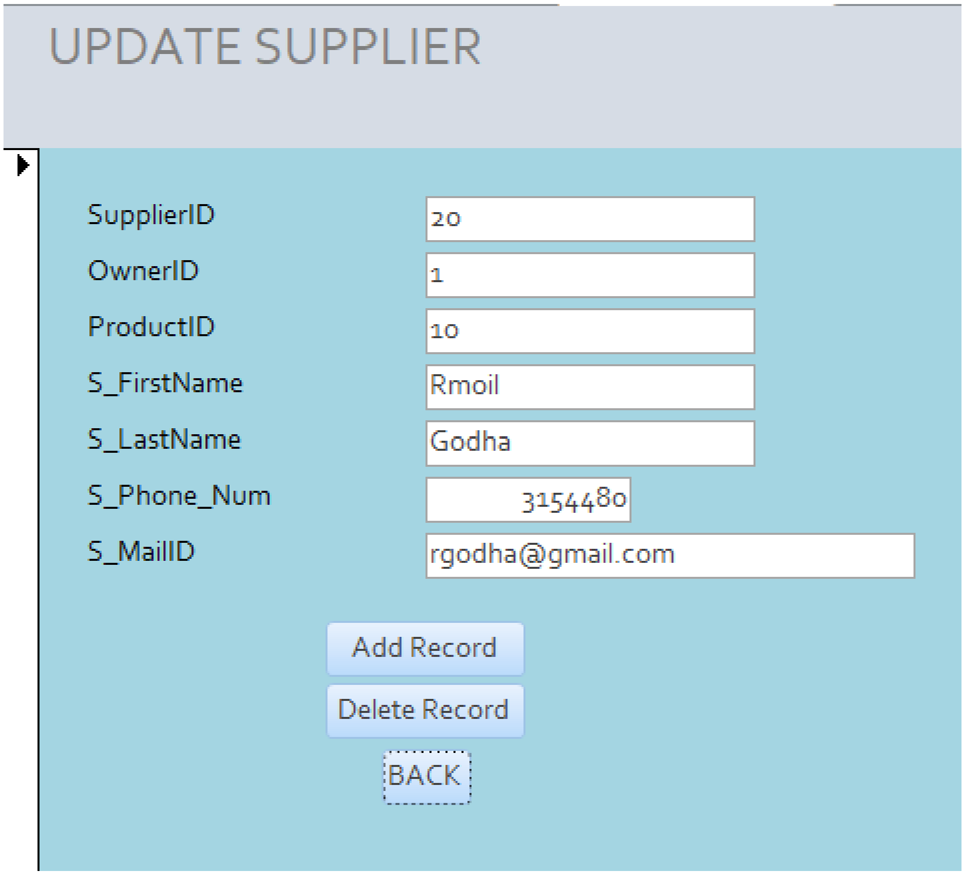
Form (Add/Update Product Window)



dbo_Prod

ProductID	10
SupplierID	1001
CategoryID	101
Product_Name	Clothes
Product_Description	Mens wear
Unit_Price	12

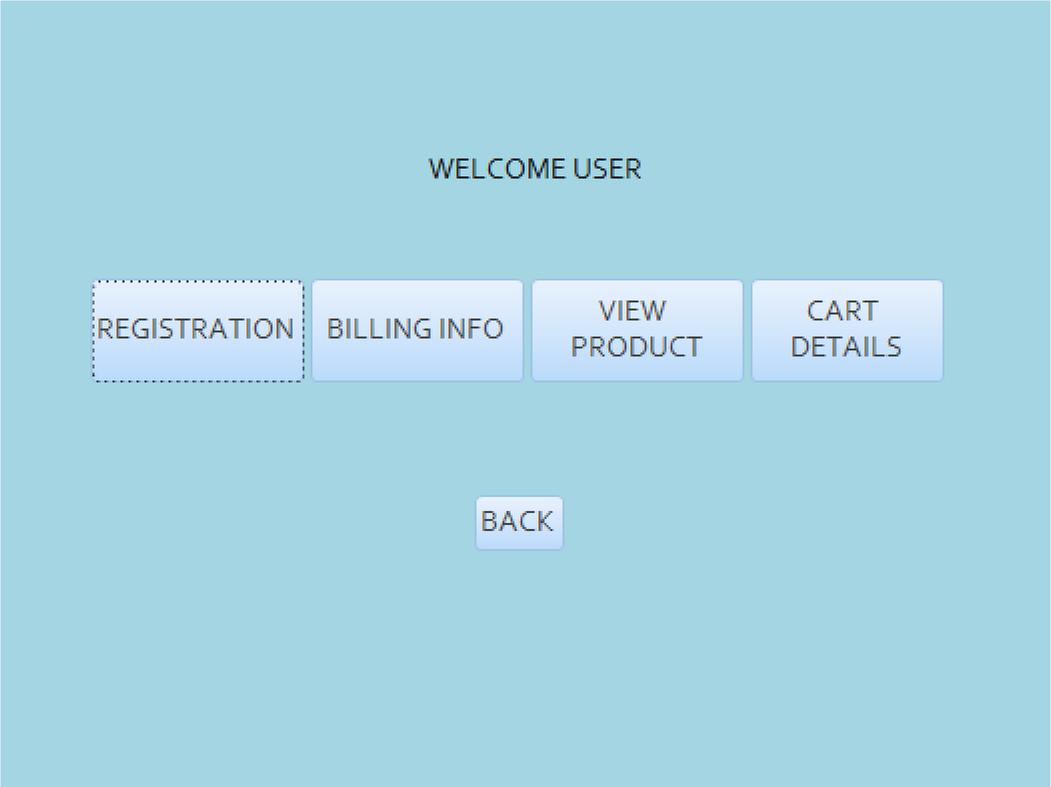
Form (Add/Update Supplier)



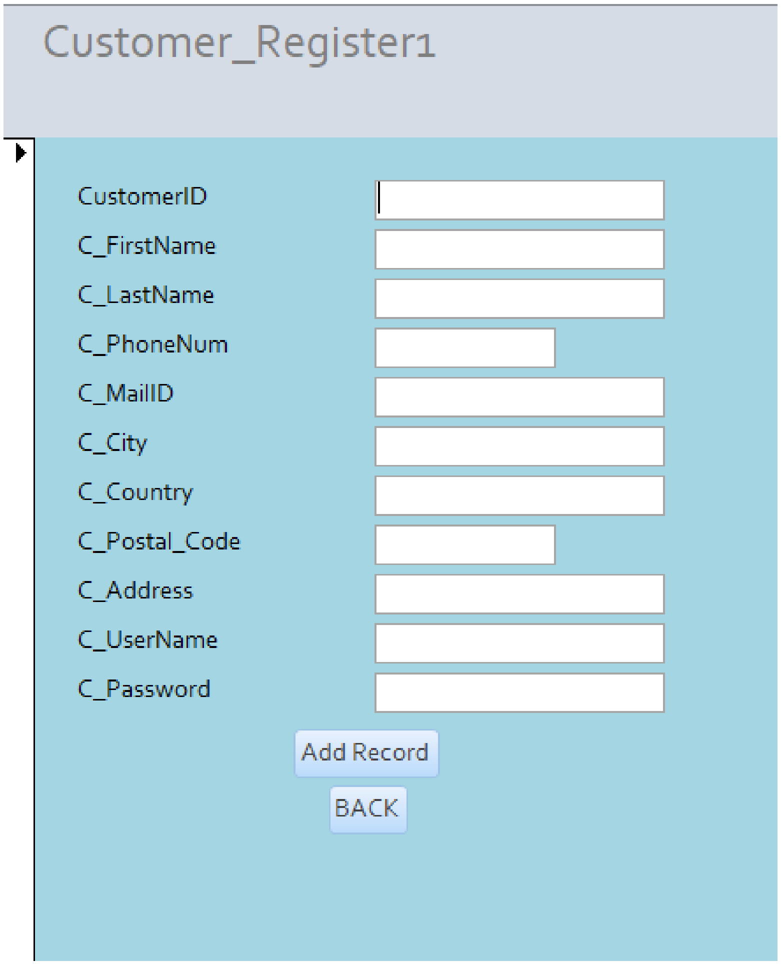
UPDATE SUPPLIER

SupplierID	20
OwnerID	1
ProductID	10
S_FirstName	Rmoil
S_LastName	Godha
S_Phone_Num	3154480
S_MailID	rgodha@gmail.com

Form (User Window)



Form (User Registration Window)



Customer_Register1

CustomerID

C_FirstName

C_LastName

C_PhoneNum

C_MailID

C_City

C_Country

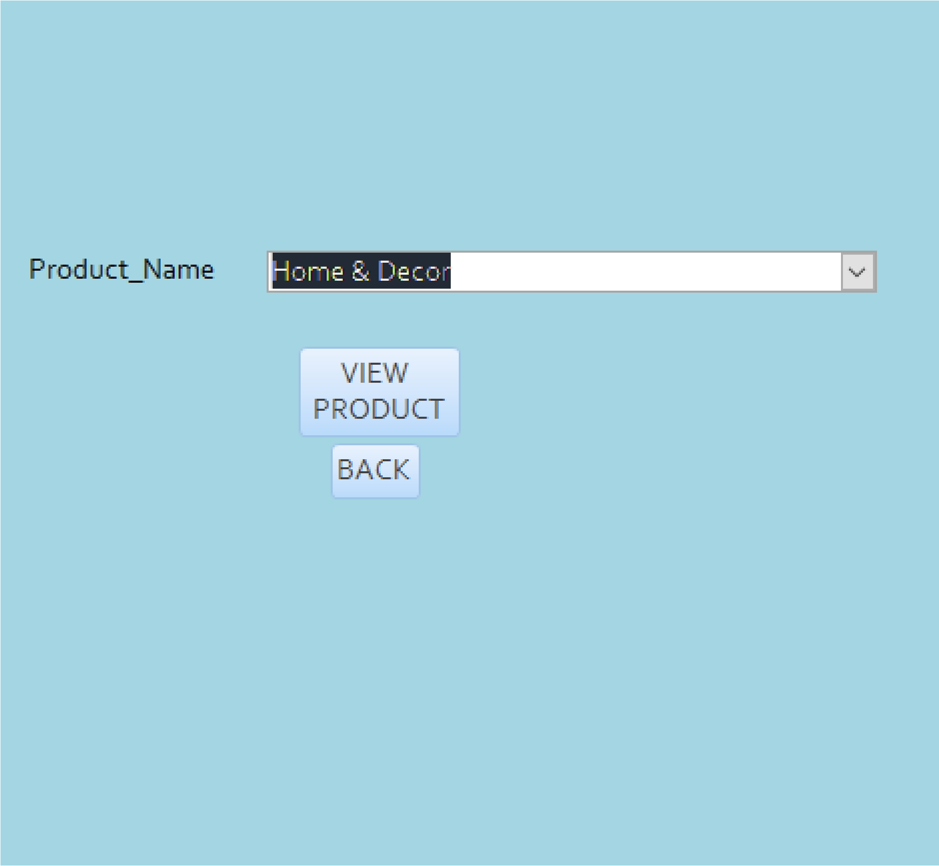
C_Postal_Code

C_Address

C_UserName

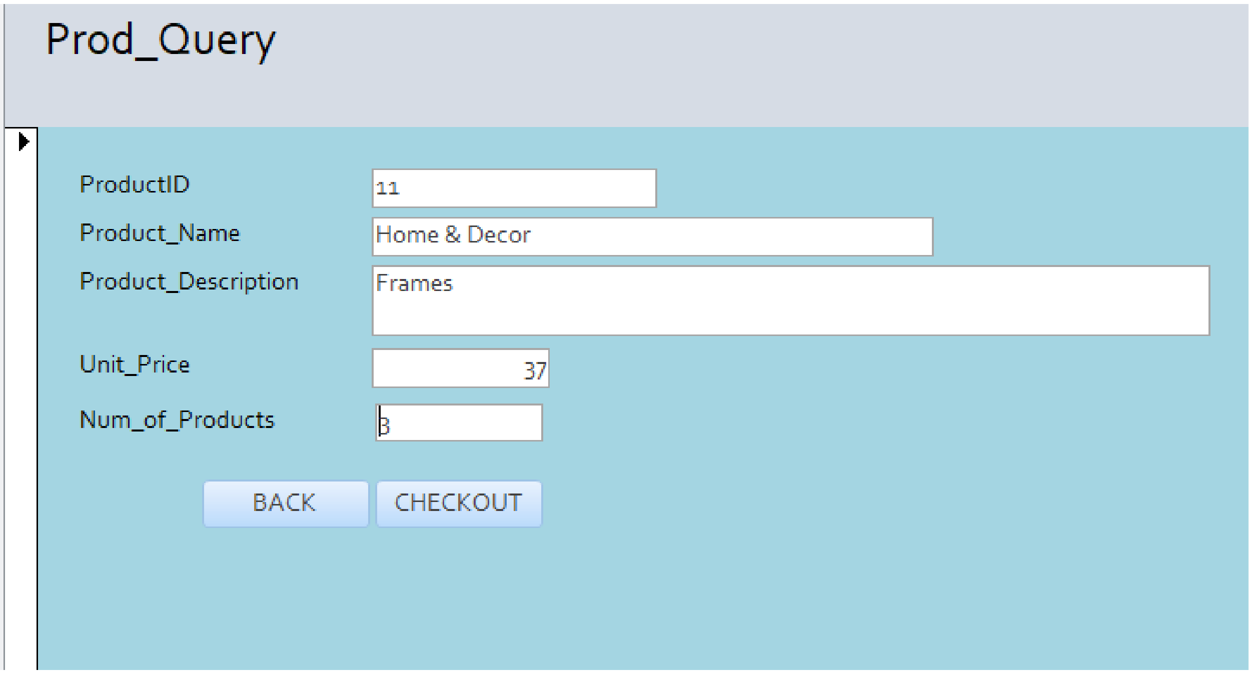
C_Password

Form (Product View Window)



Product_Name: Home & Decor

Form (Checkout Window)



Prod_Query

ProductID: 11

Product_Name: Home & Decor

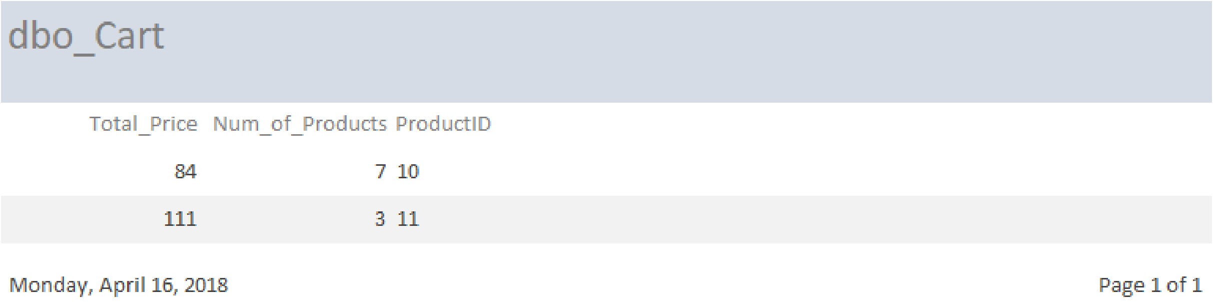
Product_Description: Frames

Unit_Price: 37

Num_of_Products: 3

Report 1

This report represents the total price of products having same productID in accordance to the number of products.



dbo_Cart

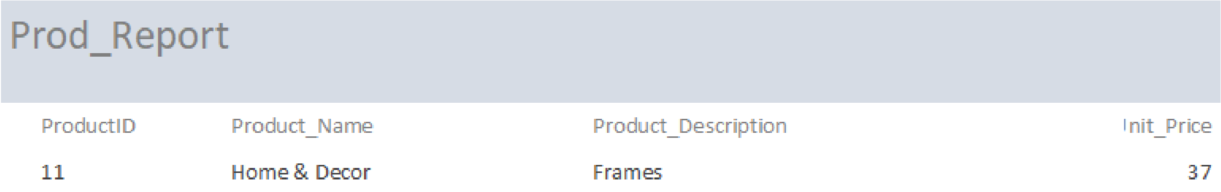
Total_Price	Num_of_Products	ProductID
84	7	10
111	3	11

Monday, April 16, 2018

Page 1 of 1

Report 2

This report represents the view product window.

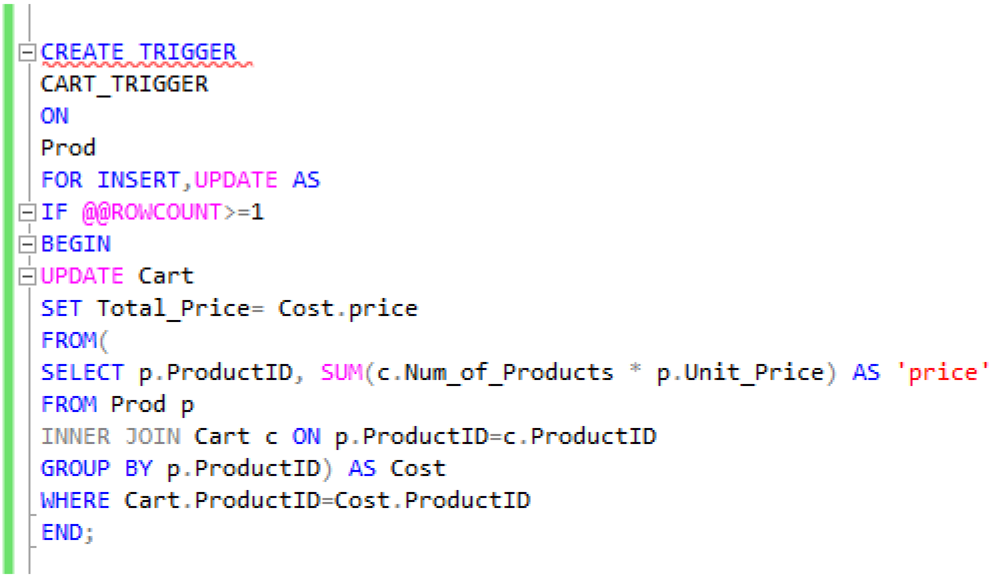


Prod_Report

ProductID	Product_Name	Product_Description	Init_Price
11	Home & Decor	Frames	37

Trigger

This trigger updates the total amount according to the number of products (of the same productID) added in the cart.



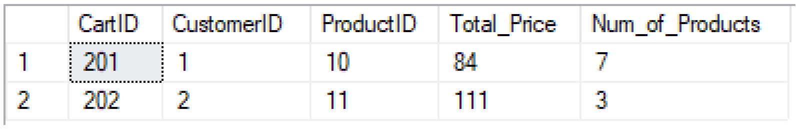
```
CREATE TRIGGER
CART_TRIGGER
ON
Prod
FOR INSERT,UPDATE AS
IF @@ROWCOUNT>=1
BEGIN
UPDATE Cart
SET Total_Price= Cost.price
FROM(
SELECT p.ProductID, SUM(c.Num_of_Products * p.Unit_Price) AS 'price'
FROM Prod p
INNER JOIN Cart c ON p.ProductID=c.ProductID
GROUP BY p.ProductID) AS Cost
WHERE Cart.ProductID=Cost.ProductID
END;
```

Trigger

```
drop trigger CART_TRIGGER;

CREATE TRIGGER
CART_TRIGGER
ON
Prod
FOR INSERT,UPDATE AS
IF @@ROWCOUNT>=1
BEGIN
UPDATE Cart
SET Total_Price= Cost.price
FROM(
SELECT p.ProductID, SUM(c.Num_of_Products * p.Unit_Price) AS 'price'
FROM inserted p
INNER JOIN Cart c ON p.ProductID=c.ProductID
GROUP BY p.ProductID) AS Cost
WHERE Cart.ProductID=Cost.ProductID
END;

SELECT * FROM Cart

insert into Prod(ProductID, SupplierID, CategoryID, Product_Name, Product_Description,
Unit_Price)
values(10, 1001, 101, 'Clothes', 'Mens wear', 12.48);
insert into Prod(ProductID, SupplierID, CategoryID, Product_Name, Product_Description,
Unit_Price)
values(11, 1002, 102, 'Home & Decor', 'Frames', 37.21);

insert into Cart(CartID, CustomerID, ProductID, Total_Price, Num_of_Products)
```

```
values(201, 01, 10, 166.21, 7);
insert into Cart(CartID, CustomerID, ProductID, Total_Price, Num_of_Products)
values(202, 02, 11, 57.83, 3);
```

Updated table after trigger



	CartID	CustomerID	ProductID	Total_Price	Num_of_Products
1	201	1	10	84	7
2	202	2	11	111	3