**Assignment #2**

**CIDM 4390**

Aaron Acosta

Krystal Arney

Gerardo Herrera Gonzalez

Ashley Reynolds

Caleb White

1. **NARRATIVE**
   1. **Use Case Diagram**

The use case diagram depicts the main functions of the Delivery Cart Application. The subject boundary houses 9 use cases, 3 actors, and their relationships. The initial use case, “Login to Delivery Cart,” begins the process of completing an item order and having it delivered to the customer’s location. From there, other functions are included as part of a natural flow to the initial use case and some functions can be extended to provide additional options for new and existing users. This diagram has been created to provide us with a simple, straightforward way of communicating the actions and intent of the application.

* 1. **Use Case Descriptions**

The use case descriptions are based on identified requirements and the use-case diagram. In an effort to fully document the different aspects of each use case, each of the descriptions simplify a process by listing out the requirements and actions of a use case in a step by step fashion. One description describes the item selection and payment functions, which includes functionalities from multiple use cases, and the second describes the item collection and delivery process.

* 1. **Class Diagram**  
     The class diagram design chosen for the DeliveryCart system contains twelve classes that allow you to visualize the different objects within the system, their static relationships that exist, and details the different attributes and functions used for modeling the real-world concept of grocery shopping. The design facilitates an understanding of what elements exist through inputs and outputs of the grocery delivery process.
  2. **Activity Diagrams**

The activity diagrams represent the series of actions and decisions made while using the DeliveryCart system from start to finish. The first activity diagram demonstrates the activities of the primary actor, the Customer/User, and the second diagram demonstrates the activities of the secondary actor, the Shopper/Delivery person. The two diagrams were chosen to display the different flow of activities if you were to login to DeliverCart as the Customer or as the Shopper.

* 1. **Sequence Diagram**

The sequence diagram shows the order of the actions that the servers take as the actor moves through the process of registering, ordering, and the full delivery process, including notifications to the actors devices, and tip/feedback notifications. This diagram depicts the dispatch and return messages need to make this model function.

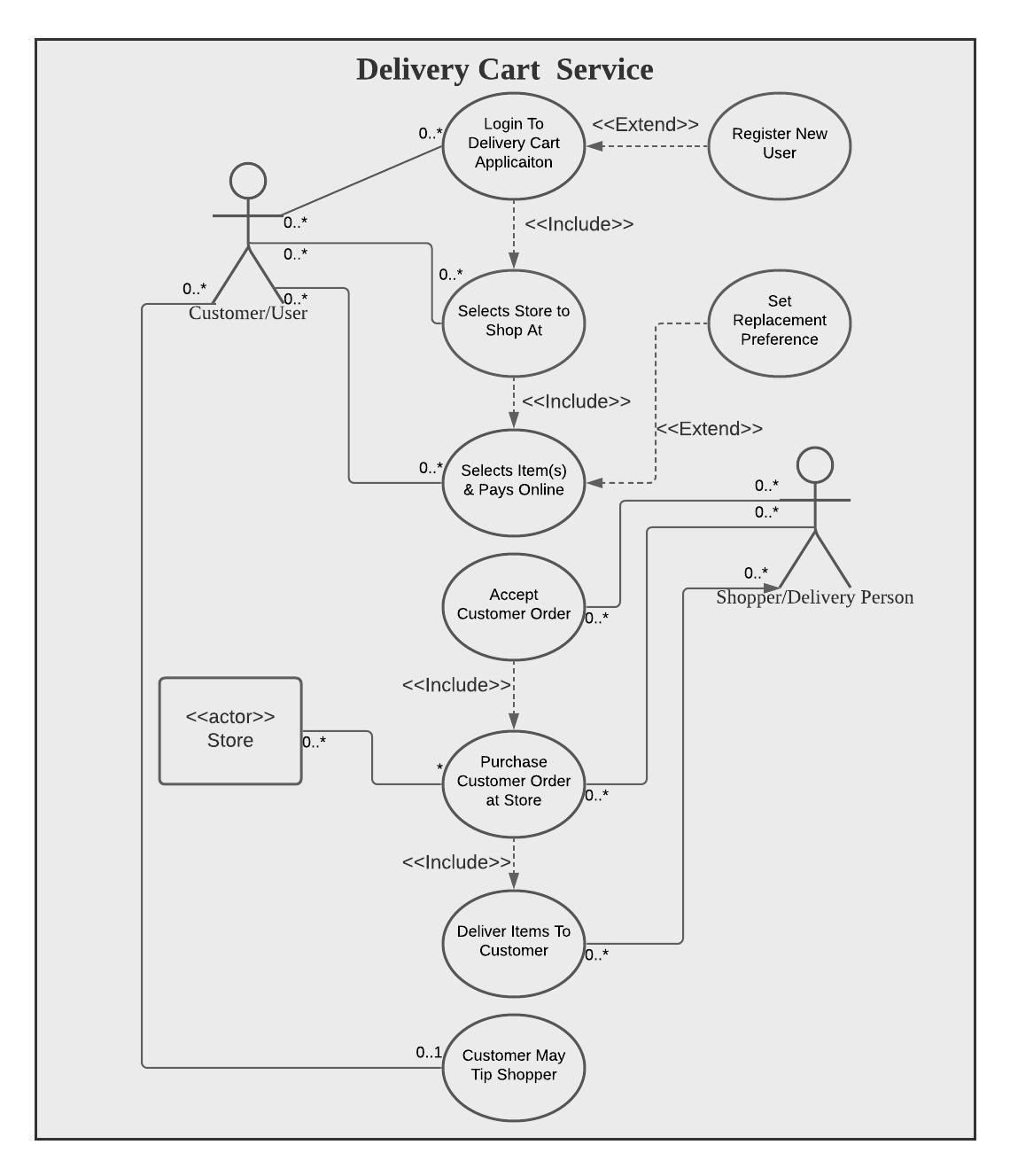
* 1. **ERD**

The ERD appropriately represents the logical structure of the database. Moreover, the diagram helps the user identify the entities which exist in the system and the relationships between the entities. The cardinal relationships in the diagram are depicted correctly.

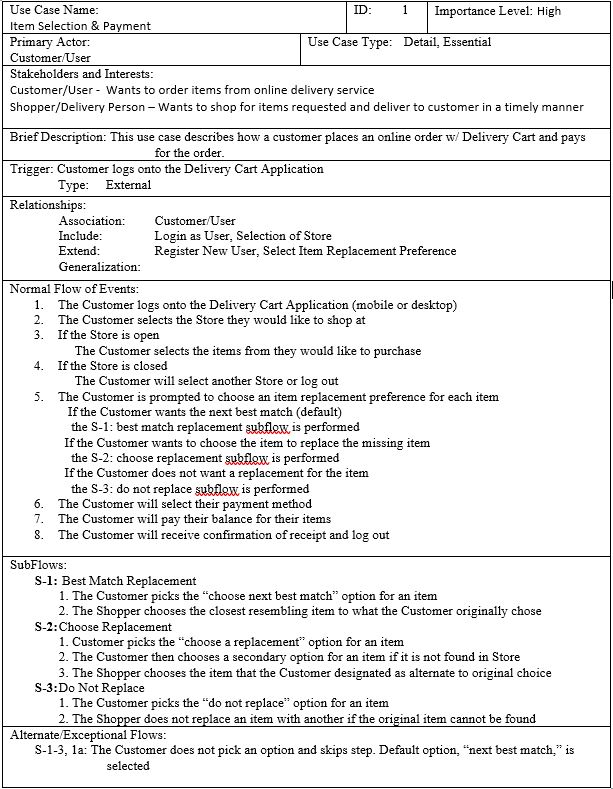
* 1. **SQL DDL Statement**

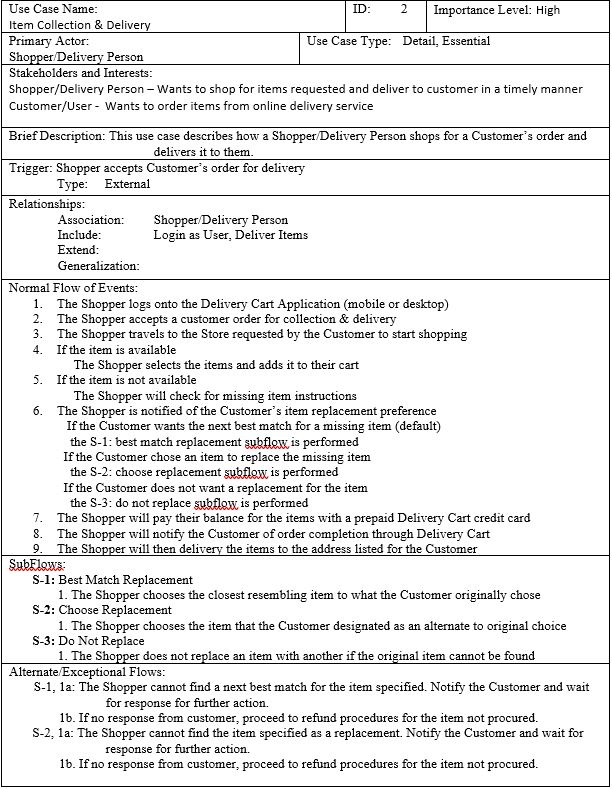
The SQL DDL statement executes without issues and correctly builds the structure of the tables and objects. Likewise, the statement accurately defines constraints.

1. **DIAGRAMS**
2. **USE CASE DIAGRAM**

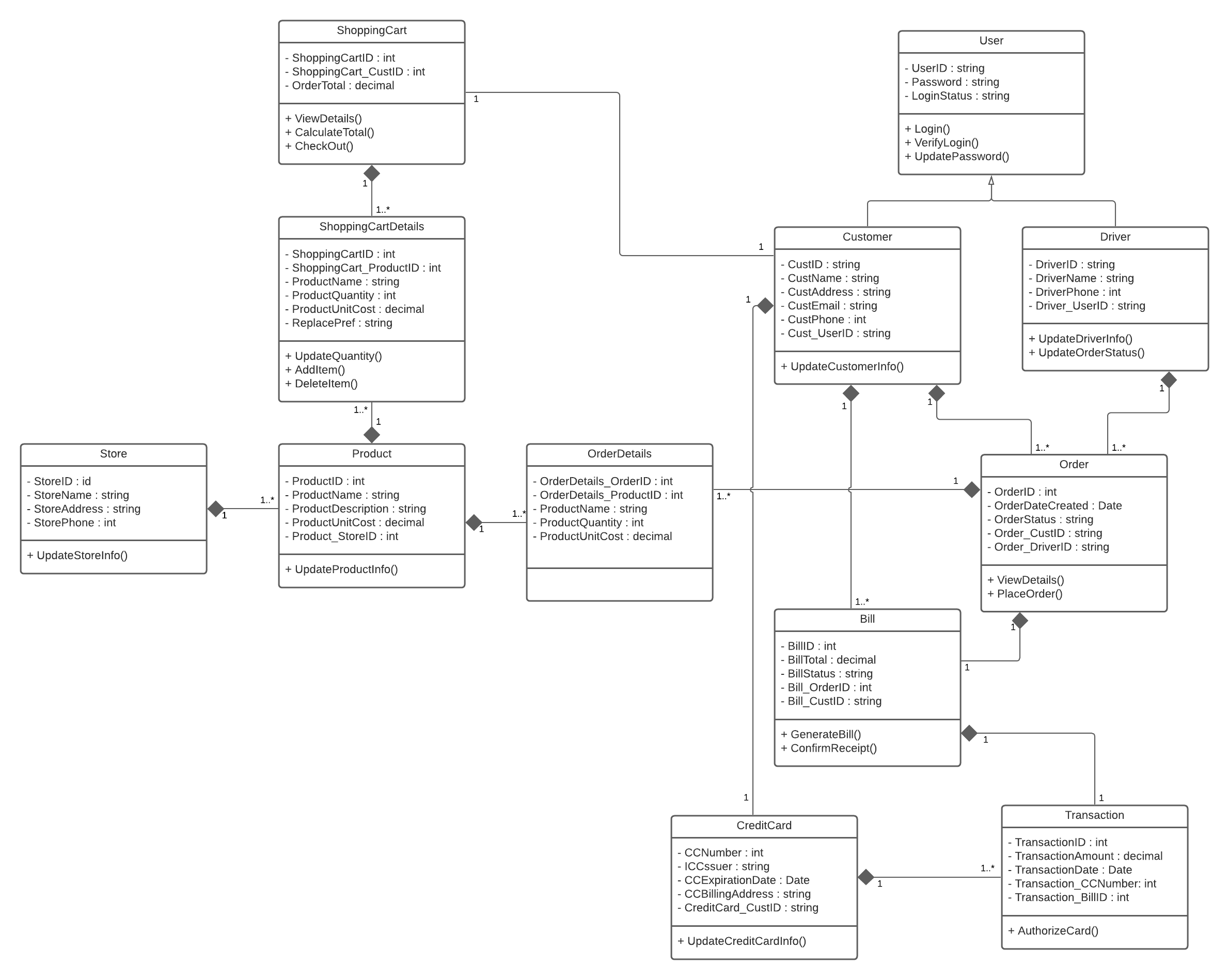
****

1. **USE CASE DESCRIPTIONS**

****

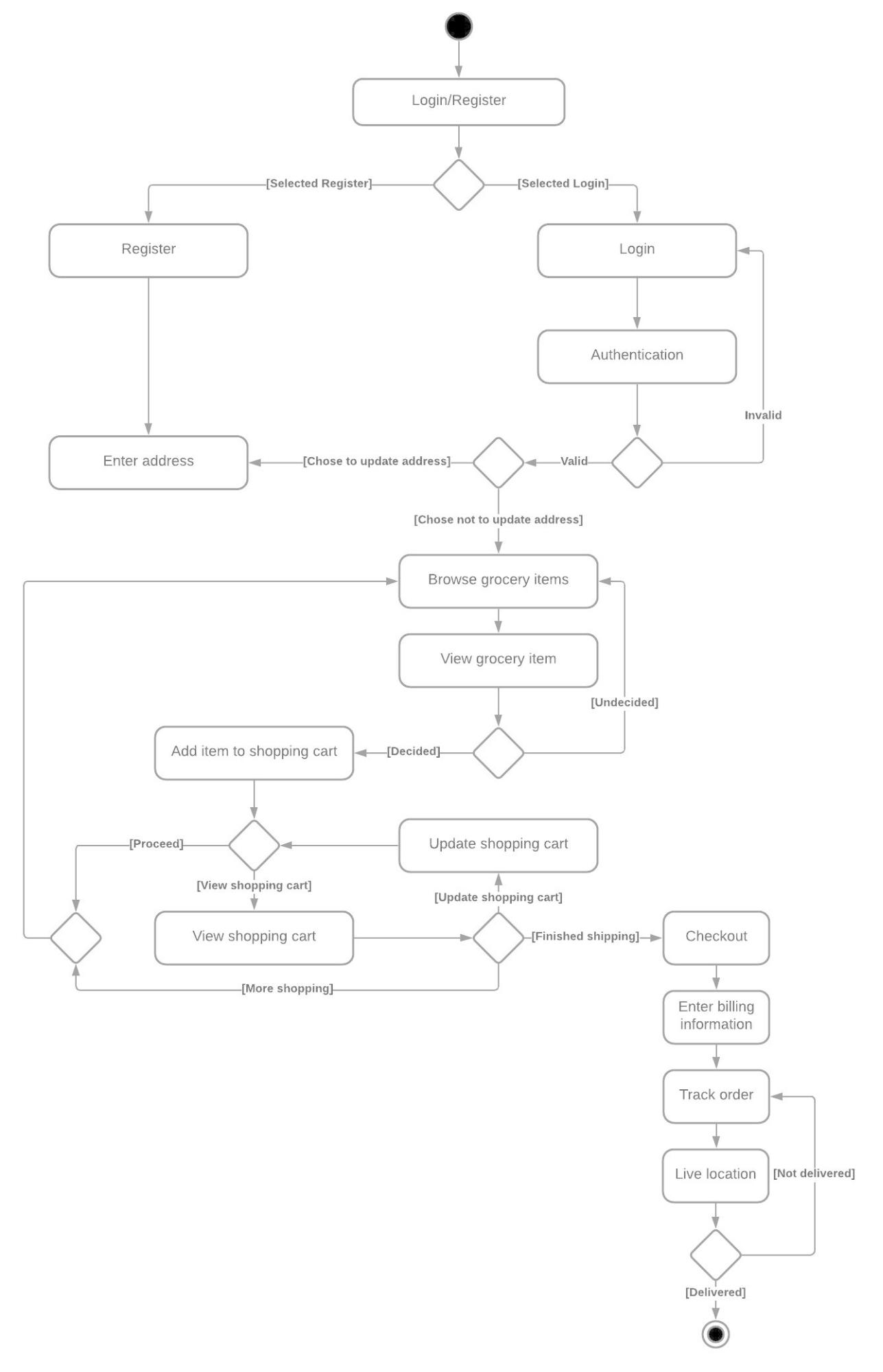
****

1. **CLASS DIAGRAM**

****

1. **ACTIVITY DIAGRAMS**

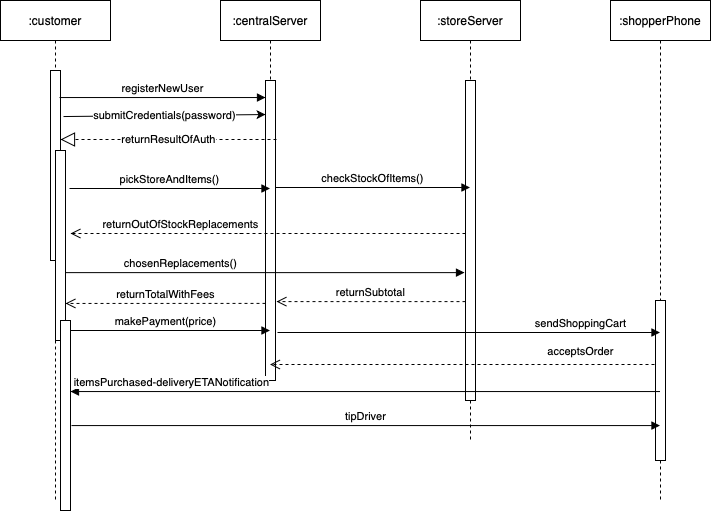
**Customer/User**



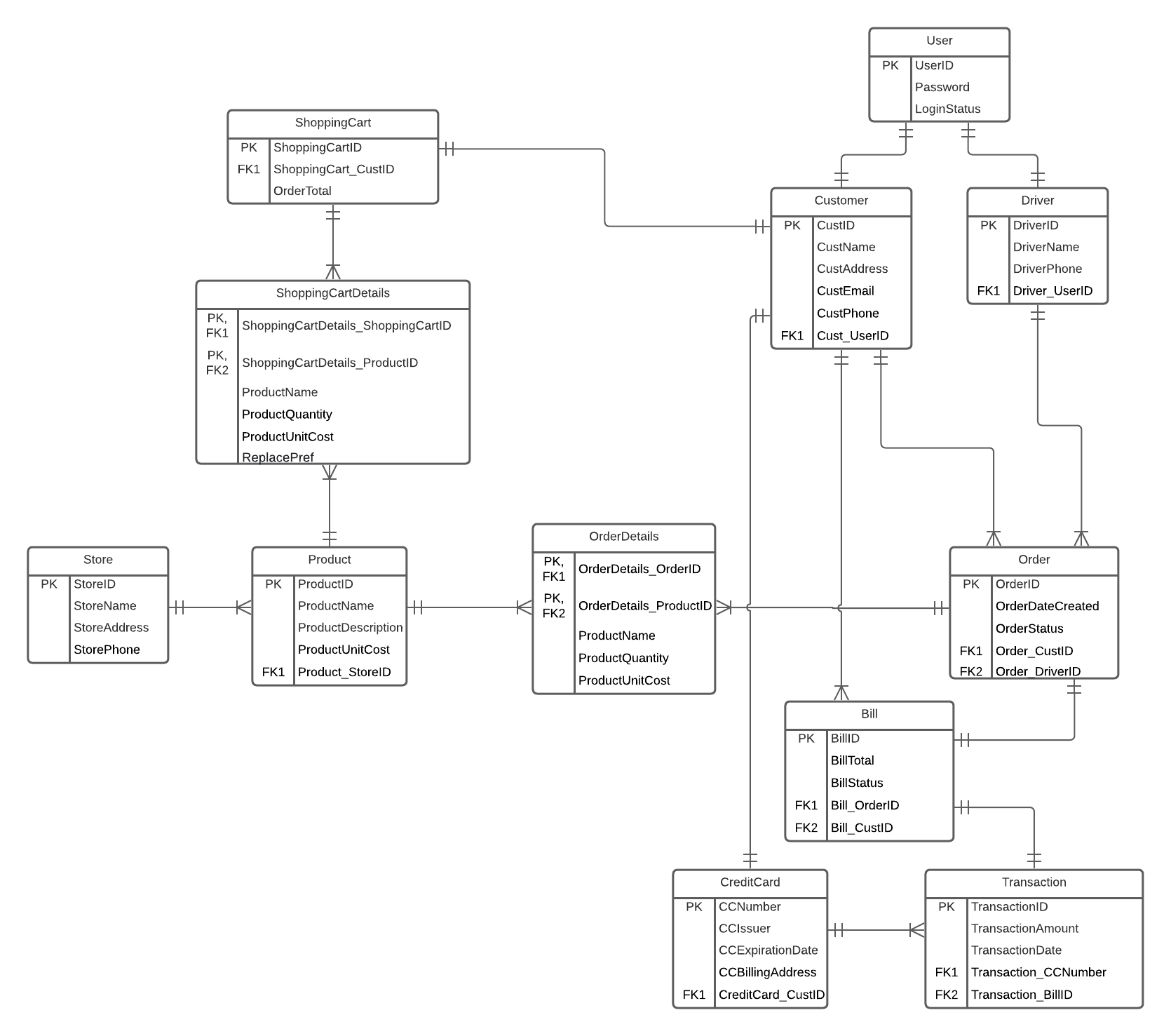
Diagram

Description automatically generated

1. **SEQUENCE DIAGRAM**



1. **ERD**



1. **SQL DDL Statement**

-- MySQL Script generated by MySQL Workbench

-- Sun Jan 24 17:37:47 2021

-- Model: New Model Version: 1.0

-- MySQL Workbench Forward Engineering

SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;

SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION';

-- -----------------------------------------------------

-- Schema GroceriesDB

-- -----------------------------------------------------

CREATE SCHEMA IF NOT EXISTS `GroceriesDB` DEFAULT CHARACTER SET utf8 ;

USE `GroceriesDB` ;

-- -----------------------------------------------------

-- Table `GroceriesDB`.`User`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `GroceriesDB`.`User` (

`UserID` VARCHAR(45) NOT NULL,

`Password` VARCHAR(45) NOT NULL,

`LoginStatus` VARCHAR(45) NOT NULL,

PRIMARY KEY (`UserID`),

UNIQUE INDEX `user\_UNIQUE` (`UserID` ASC) VISIBLE)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `GroceriesDB`.`Customer`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `GroceriesDB`.`Customer` (

`CustID` VARCHAR(45) NOT NULL,

`CustName` VARCHAR(45) NOT NULL,

`CustAddress` VARCHAR(45) NOT NULL,

`CustEmail` VARCHAR(45) NOT NULL,

`CustPhone` INT NOT NULL,

`Cust\_UserID` VARCHAR(45) NOT NULL,

UNIQUE INDEX `cust\_id\_UNIQUE` (`CustID` ASC) VISIBLE,

PRIMARY KEY (`CustID`, `Cust\_UserID`),

INDEX `cust\_userID\_idx` (`Cust\_UserID` ASC) VISIBLE,

CONSTRAINT `Cust\_UserID`

FOREIGN KEY (`Cust\_UserID`)

REFERENCES `GroceriesDB`.`User` (`UserID`)

ON DELETE CASCADE

ON UPDATE CASCADE)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `GroceriesDB`.`Driver`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `GroceriesDB`.`Driver` (

`DriverID` VARCHAR(45) NOT NULL,

`DriverName` VARCHAR(45) NOT NULL,

`DriverPhone` INT NOT NULL,

`Driver\_UserID` VARCHAR(45) NOT NULL,

PRIMARY KEY (`DriverID`, `Driver\_UserID`),

UNIQUE INDEX `deliver\_id\_UNIQUE` (`DriverID` ASC) VISIBLE,

INDEX `user\_id\_idx` (`Driver\_UserID` ASC) VISIBLE,

CONSTRAINT `Deliver\_UserID`

FOREIGN KEY (`Driver\_UserID`)

REFERENCES `GroceriesDB`.`User` (`UserID`)

ON DELETE CASCADE

ON UPDATE CASCADE)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `GroceriesDB`.`Order`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `GroceriesDB`.`Order` (

`OrderID` INT NOT NULL AUTO\_INCREMENT,

`OrderDateCreated` DATE NOT NULL,

`OrderStatus` VARCHAR(45) NOT NULL,

`Order\_CustID` VARCHAR(45) NOT NULL,

`Order\_DriverID` VARCHAR(45) NOT NULL,

PRIMARY KEY (`OrderID`, `Order\_CustID`, `Order\_DriverID`),

UNIQUE INDEX `order\_id\_UNIQUE` (`OrderID` ASC) VISIBLE,

INDEX `cust\_id\_idx` (`Order\_CustID` ASC) VISIBLE,

INDEX `deliver\_id\_idx` (`Order\_DriverID` ASC) VISIBLE,

CONSTRAINT `Order\_CustID`

FOREIGN KEY (`Order\_CustID`)

REFERENCES `GroceriesDB`.`Customer` (`CustID`)

ON DELETE CASCADE

ON UPDATE CASCADE,

CONSTRAINT `Order\_DriverID`

FOREIGN KEY (`Order\_DriverID`)

REFERENCES `GroceriesDB`.`Driver` (`DriverID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `GroceriesDB`.`CreditCard`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `GroceriesDB`.`CreditCard` (

`CCNumber` INT NOT NULL,

`CCIssuer` VARCHAR(45) NOT NULL,

`CCExpirationDate` DATE NOT NULL,

`CCBillingAddress` VARCHAR(45) NOT NULL,

`CreditCard\_CustID` VARCHAR(45) NOT NULL,

PRIMARY KEY (`CCNumber`, `CreditCard\_CustID`),

INDEX `CreditCard\_CustID\_idx` (`CreditCard\_CustID` ASC) VISIBLE,

CONSTRAINT `CreditCard\_CustID`

FOREIGN KEY (`CreditCard\_CustID`)

REFERENCES `GroceriesDB`.`Customer` (`CustID`)

ON DELETE CASCADE

ON UPDATE CASCADE)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `GroceriesDB`.`Bill`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `GroceriesDB`.`Bill` (

`BillID` INT NOT NULL,

`BillTotal` DECIMAL(65,2) NOT NULL,

`BillStatus` VARCHAR(45) NOT NULL,

`Bill\_OrderID` INT NOT NULL,

`Bill\_CustID` VARCHAR(45) NOT NULL,

PRIMARY KEY (`BillID`, `Bill\_OrderID`, `Bill\_CustID`),

INDEX `Bill\_OrderID\_idx` (`Bill\_OrderID` ASC) VISIBLE,

INDEX `Bill\_CustID\_idx` (`Bill\_CustID` ASC) VISIBLE,

CONSTRAINT `Bill\_OrderID`

FOREIGN KEY (`Bill\_OrderID`)

REFERENCES `GroceriesDB`.`Order` (`OrderID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `Bill\_CustID`

FOREIGN KEY (`Bill\_CustID`)

REFERENCES `GroceriesDB`.`Customer` (`CustID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `GroceriesDB`.`Transaction`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `GroceriesDB`.`Transaction` (

`TransactionID` INT NOT NULL AUTO\_INCREMENT,

`TransactionAmount` DECIMAL(65,2) NOT NULL,

`TransactionDate` DATE NOT NULL,

`Transaction\_CCNumber` INT NOT NULL,

`Transaction\_BillID` INT NOT NULL,

PRIMARY KEY (`TransactionID`, `Transaction\_BillID`, `Transaction\_CCNumber`),

UNIQUE INDEX `transaction\_id\_UNIQUE` (`TransactionID` ASC) VISIBLE,

INDEX `cardNumber\_idx` (`Transaction\_CCNumber` ASC) VISIBLE,

INDEX `Transaction\_BillID\_idx` (`Transaction\_BillID` ASC) VISIBLE,

CONSTRAINT `Transaction\_CardNumber`

FOREIGN KEY (`Transaction\_CCNumber`)

REFERENCES `GroceriesDB`.`CreditCard` (`CCNumber`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `Transaction\_BillID`

FOREIGN KEY (`Transaction\_BillID`)

REFERENCES `GroceriesDB`.`Bill` (`BillID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `GroceriesDB`.`Store`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `GroceriesDB`.`Store` (

`StoreID` INT NOT NULL,

`StoreName` VARCHAR(45) NOT NULL,

`StoreAddress` VARCHAR(45) NOT NULL,

`StorePhone` INT NOT NULL,

PRIMARY KEY (`StoreID`),

UNIQUE INDEX `store\_id\_UNIQUE` (`StoreID` ASC) VISIBLE)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `GroceriesDB`.`Product`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `GroceriesDB`.`Product` (

`ProductID` INT NOT NULL,

`ProductName` VARCHAR(45) NOT NULL,

`ProductDescription` VARCHAR(45) NOT NULL,

`ProductUnitCost` VARCHAR(45) NOT NULL,

`Product\_StoreID` INT NOT NULL,

PRIMARY KEY (`ProductID`, `Product\_StoreID`),

UNIQUE INDEX `product\_id\_UNIQUE` (`ProductID` ASC) VISIBLE,

INDEX `store\_id\_idx` (`Product\_StoreID` ASC) VISIBLE,

CONSTRAINT `Product\_StoreID`

FOREIGN KEY (`Product\_StoreID`)

REFERENCES `GroceriesDB`.`Store` (`StoreID`)

ON DELETE CASCADE

ON UPDATE CASCADE)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `GroceriesDB`.`OrderDetails`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `GroceriesDB`.`OrderDetails` (

`OrderDetails\_OrderID` INT NOT NULL,

`OrderDetails\_ProductID` INT NOT NULL,

`ProductName` VARCHAR(45) NOT NULL,

`ProductQuantity` INT NOT NULL,

`ProductUnitCost` FLOAT NOT NULL,

INDEX `order\_id\_idx` (`OrderDetails\_OrderID` ASC) VISIBLE,

INDEX `product\_id\_idx` (`OrderDetails\_ProductID` ASC) VISIBLE,

PRIMARY KEY (`OrderDetails\_OrderID`, `OrderDetails\_ProductID`),

CONSTRAINT `OrderDetails\_OrderID`

FOREIGN KEY (`OrderDetails\_OrderID`)

REFERENCES `GroceriesDB`.`Order` (`OrderID`)

ON DELETE CASCADE

ON UPDATE CASCADE,

CONSTRAINT `OrderDetails\_ProductID`

FOREIGN KEY (`OrderDetails\_ProductID`)

REFERENCES `GroceriesDB`.`Product` (`ProductID`)

ON DELETE CASCADE

ON UPDATE CASCADE)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `GroceriesDB`.`ShoppingCart`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `GroceriesDB`.`ShoppingCart` (

`ShoppingCartID` INT NOT NULL AUTO\_INCREMENT,

`ShoppingCart\_CustID` VARCHAR(45) NOT NULL,

`OrderTotal` DECIMAL(65,2) NOT NULL,

PRIMARY KEY (`ShoppingCartID`, `ShoppingCart\_CustID`),

UNIQUE INDEX `cart\_id\_UNIQUE` (`ShoppingCartID` ASC) VISIBLE,

INDEX `cust\_id\_idx` (`ShoppingCart\_CustID` ASC) VISIBLE,

CONSTRAINT `ShoppingCart\_CustID`

FOREIGN KEY (`ShoppingCart\_CustID`)

REFERENCES `GroceriesDB`.`Customer` (`CustID`)

ON DELETE CASCADE

ON UPDATE CASCADE)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `GroceriesDB`.`ShoppingCartDetails`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `GroceriesDB`.`ShoppingCartDetails` (

`ShoppingCartDetails\_ShoppingCartID` INT NOT NULL,

`ShoppingCartDetails\_ProductID` INT NOT NULL,

`ProductName` VARCHAR(45) NOT NULL,

`ProductQuantity` VARCHAR(45) NOT NULL,

`ProductUnitCost` VARCHAR(45) NOT NULL,

`ReplacePref` VARCHAR(45) NOT NULL,

PRIMARY KEY (`ShoppingCartDetails\_ShoppingCartID`, `ShoppingCartDetails\_ProductID`),

INDEX `ShoppingCartDetails\_ProductID\_idx` (`ShoppingCartDetails\_ProductID` ASC) VISIBLE,

CONSTRAINT `ShoppingCartDetails\_ShoppingCartID`

FOREIGN KEY (`ShoppingCartDetails\_ShoppingCartID`)

REFERENCES `GroceriesDB`.`ShoppingCart` (`ShoppingCartID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `ShoppingCartDetails\_ProductID`

FOREIGN KEY (`ShoppingCartDetails\_ProductID`)

REFERENCES `GroceriesDB`.`Product` (`ProductID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

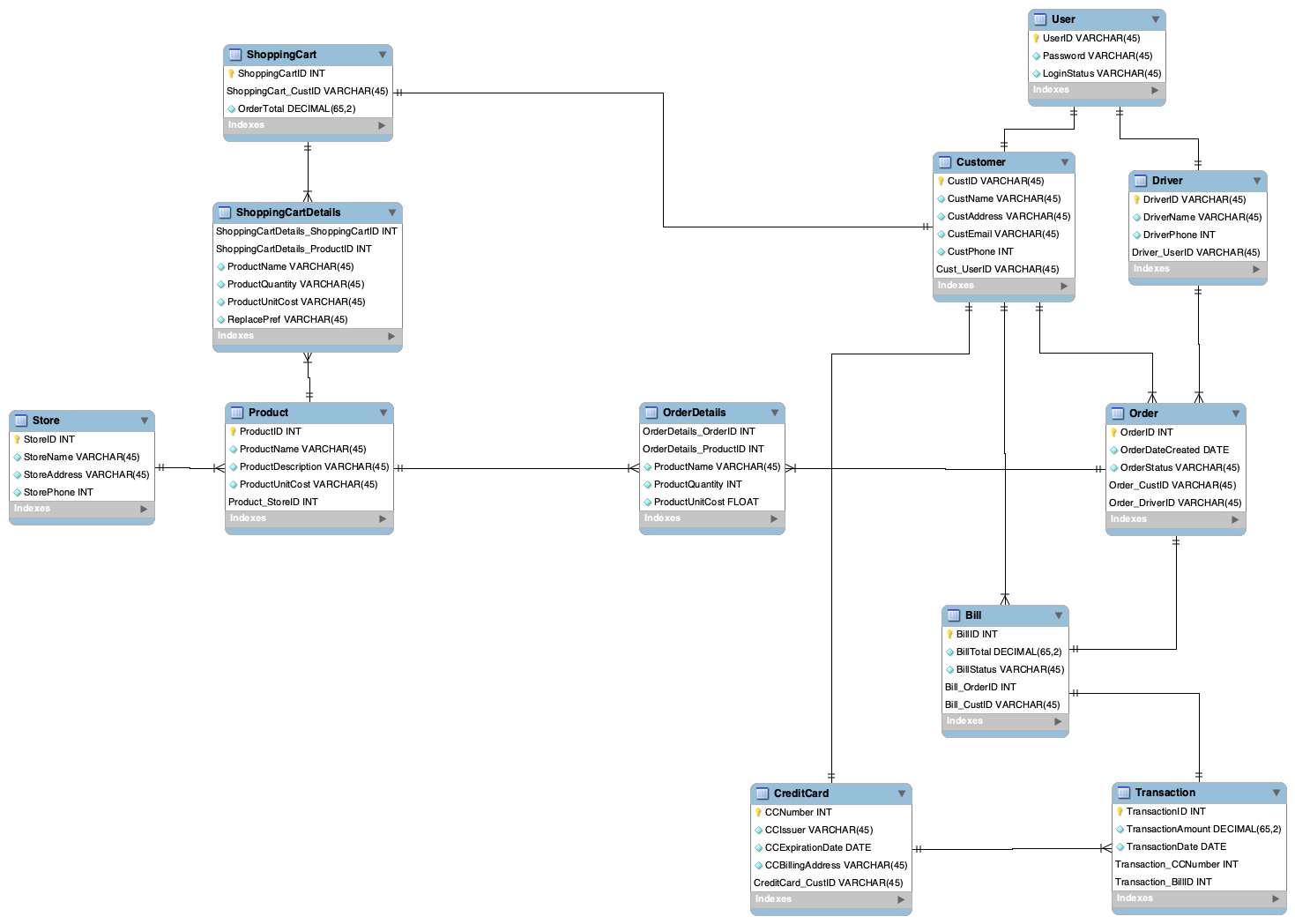
ENGINE = InnoDB;

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

1. **EER DIAGRAM (Included to complement the SQL DDL Statement)**

****