***Abdullah Gul University***

28.03.2019

Melik Bozkurt 110510059

Alperen Çınar 110510048

Yasin Enes Deliak 110510060

DATABASE MANAGEMENT SYSTEMS (COMP 204)

Clothing Store Automation Project

Index

[1 Name of the Project 2](#_Toc8574737)

[2 Members of the Project 2](#_Toc8574738)

[3 Purpose of the Project 2](#_Toc8574739)

[4 User Roles of the Project 2](#_Toc8574740)

[5 Specifications of the Project 2](#_Toc8574741)

[6 Tools/IDEs (Will Be) Used in the Project 3](#_Toc8574742)

[7 Summary of the Project 3](#_Toc8574744)

[8 Logical Diagram/E-R Model and Explanation 3](#_Toc8574745)

[9 UML Use Case Diagram 7](#_Toc8574746)

[10 Table Creation Scripts and Constraints 7](#_Toc8574747)

[11 Insertion Scripts 12](#_Toc8574748)

[12 Screenshots 14](#_Toc8574749)

## Name of the Project

Clothing Store Automation

## Members of the Project

Melik BOZKURT

Alperen ÇINAR

Yasin Enes DELİAK

## Purpose of the Project

It is a software that will be created to meet the requirements of a clothing store and to speed up the sales process.

## User Roles of the Project

Clothing store automation allows the collection and sharing of necessary information and data to provide better product sales. This automation, keeps the product sales operations, personnel registration and operations, product stock information, store information, customer information and uses that information when it is needed.

## Specifications of the Project

Our system which is designed for convenience:

-Allows personnel to make sales process quickly.

-Allows to make stock update operations easily.

-Makes it easy to manage customer information.

-Provides personnel management.

-It provides an easy and understandable interface for personnel to use the system.

-The product performs billing.

## Tools/IDEs (Will Be) Used in the Project

## Firstly, we used dia program to create diagrams. We will use java-eclipse for user interface. In addition, we will use MySQL-Workbench to create and manage data. Wamp server is used to test in local network.

## Summary of the Project

Our project is a software that responds to the store and customer needs, keeping the information of products, personnel, store and customers.

## Logical Diagram/E-R Model and Explanation

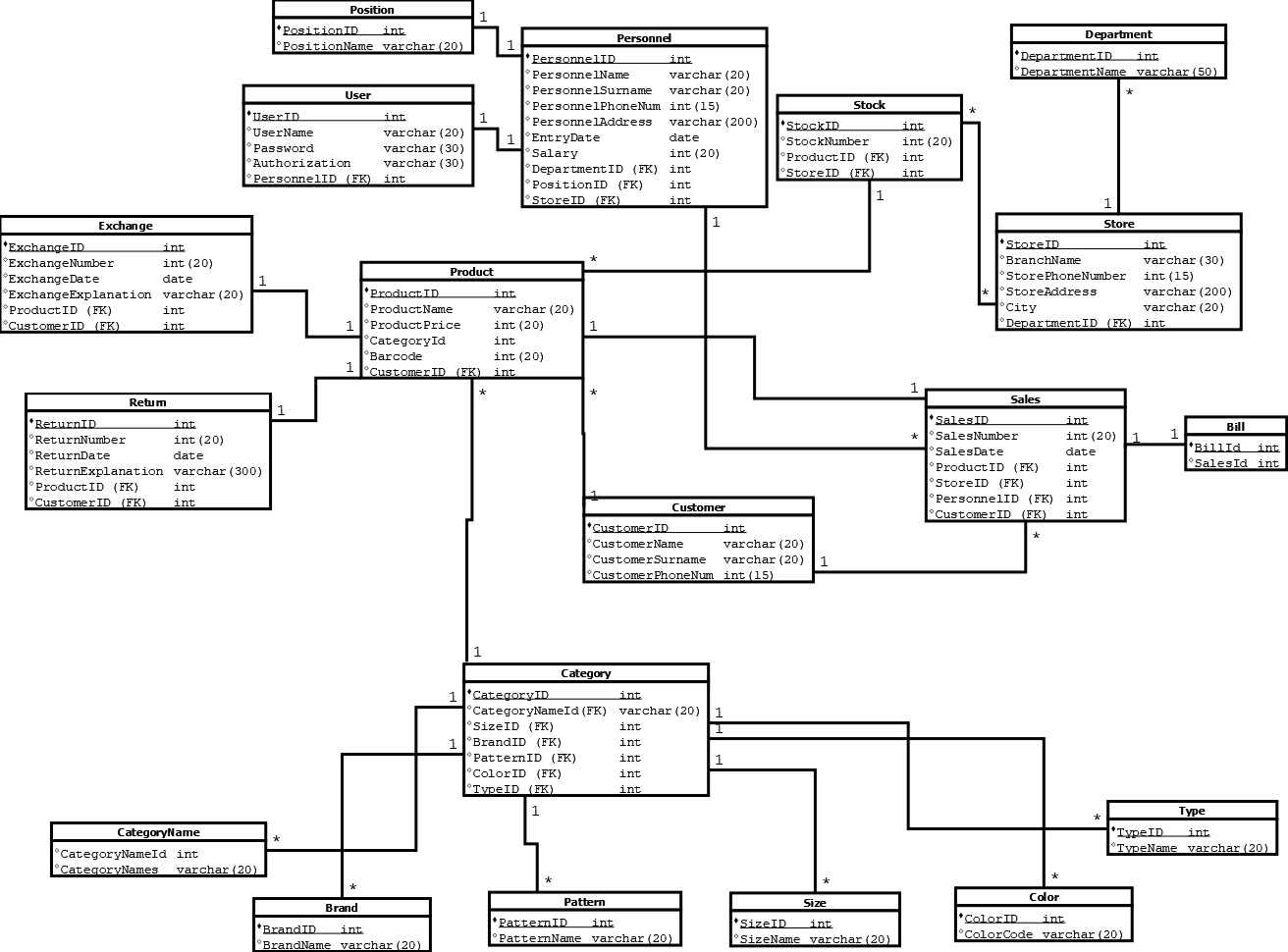


Figure 1: Logical Diagram

As shown in the figure 1 there will be 19 tables, which are needed for clothing store automation in our project. In tables, underlined parts are primary keys and we used (FK) for foreign keys. We will keep some data such as personnel information, product information and store information. You can see detailed data in figure 1. Also, the type of variables and relations between tables shown in figure 1.

Mapping:

For one to many mapping we used 1-\*,

For one to one mapping we used 1-1,

For many to one mapping we used \*-1,

For many to many mapping we used \*-\*

One personal can have one position (1-1), but can have more than one sales (1-\*). Every sale has its own bill (1-1). Every product has its own sale information (1-1), but there can be lots of product in one stock.

(1-\*). Products have their return and exchange information and if there is exchange or return, it is just for one product (1-1). There can be many stocks in one store also these stocks can be same for different stores (\*-\*). Moreover, one store can have different departments (1-\*). One category can have several products (1-\*) and it can has different brands, patterns, colors, size and types (1-\*). Finally, one customer can buy more than one product (1-\*).

You can see E-R model in fig. 2 (page 6). In E-R Model, the primary keys are written underlined and foreign keys are in double circles. The other columns are written in normal circles. Tables are written in rectangles and the relation between tables are in rhombs. Arrows show the mapping between the tables. The arrowhead show us 1 and the tail of arrow shows many. In addition, the weak entity sets and their relations are showed with double arrows and weak entity sets are in double rectangles.

All tables are normalized. For example, normally our category table contained name, brand name, pattern name, color code, type name and size name. However, now these are in different tables and all of them has a relation with category table with foreign keys. Also, bill table, department table and position table has the same situation. In advance they were in the tables that now they have a relation with. Return and exchange tables were in product table, after the normalization we made new tables for them.

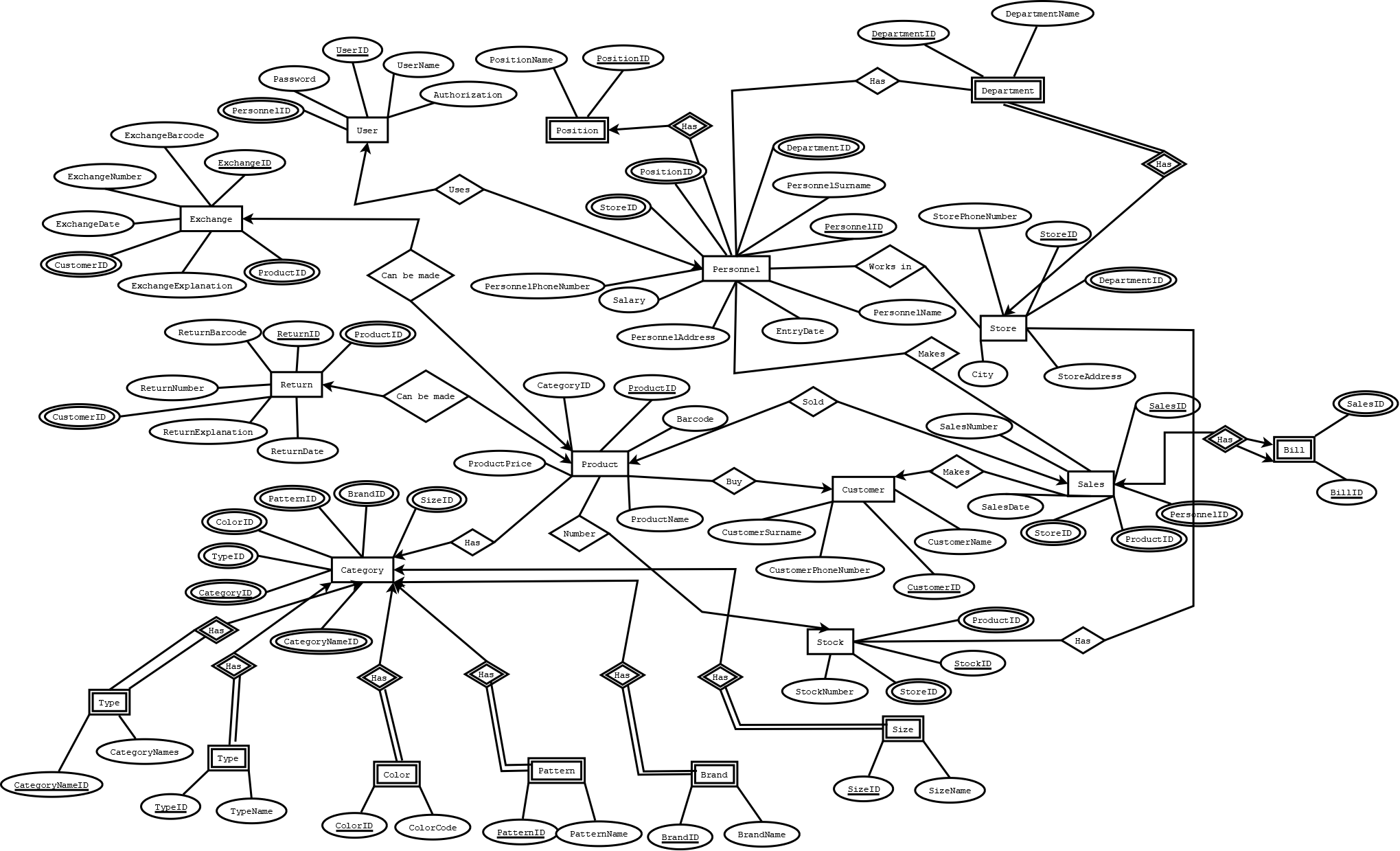


Figure 2: E-R Model

## UML Use Case Diagram

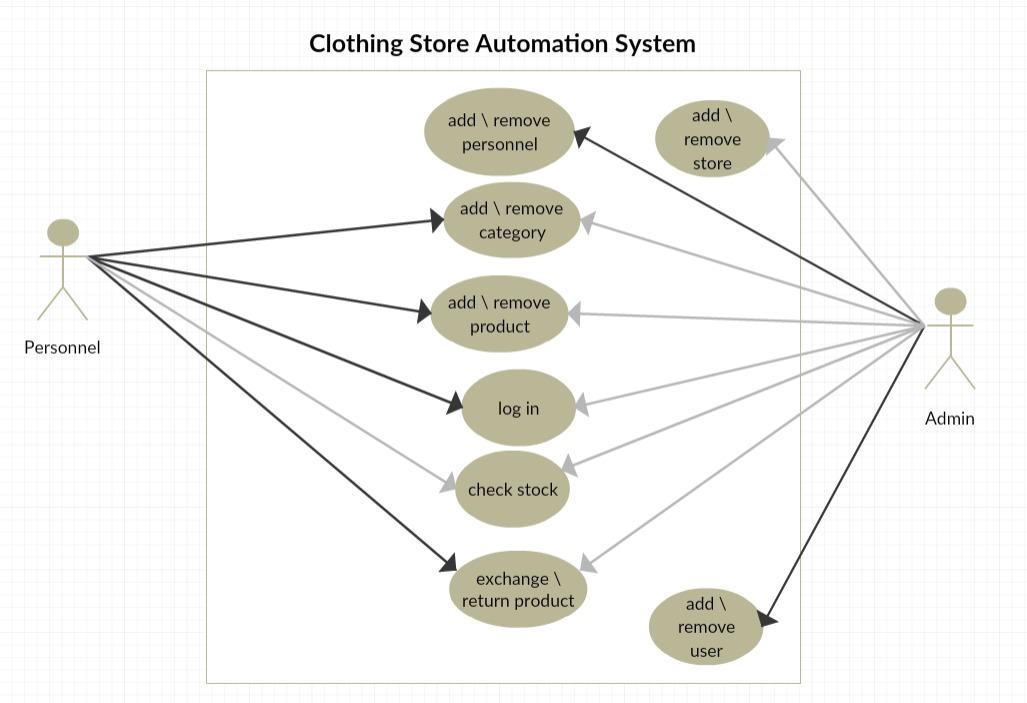


Figure 3: Use Case Diagram

## Table Creation Scripts and Constraints

The codes to create tables and their constraints are shown in the below:

**CATEGORY Table**

*CREATE TABLE `category` (*

*`CategoryID` int(11) NOT NULL,*

*`CategoryNameID` int(11) NOT NULL,*

*`SizeID` int(11) NOT NULL,*

*`BrandID` int(11) NOT NULL,*

*`PatternID` int(11) NOT NULL,*

*`ColorID` int(11) NOT NULL,*

*`TypeID` int(11) NOT NULL,*

*PRIMARY KEY (`CategoryID`),*

*KEY `CategoryNameID` (`CategoryNameID`),*

*KEY `BrandID` (`BrandID`),*

*KEY `PatternID` (`PatternID`),*

*KEY `SizeID` (`SizeID`),*

*KEY `ColorID` (`ColorID`),*

*KEY `TypeID` (`TypeID`),*

*CONSTRAINT `category\_ibfk\_1` FOREIGN KEY (`CategoryNameID`) REFERENCES `categoryname` (`CategoryNameID`),*

*CONSTRAINT `category\_ibfk\_2` FOREIGN KEY (`BrandID`) REFERENCES `brand` (`BrandID`),*

*CONSTRAINT `category\_ibfk\_3` FOREIGN KEY (`PatternID`) REFERENCES `pattern` (`PatternID`),*

*CONSTRAINT `category\_ibfk\_4` FOREIGN KEY (`SizeID`) REFERENCES `size` (`SizeID`),*

*CONSTRAINT `category\_ibfk\_5` FOREIGN KEY (`ColorID`) REFERENCES `color` (`ColorID`),*

*CONSTRAINT `category\_ibfk\_6` FOREIGN KEY (`TypeID`) REFERENCES `type` (`TypeID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**BRAND Table**

*CREATE TABLE `brand` (*

*`BrandID` int(11) NOT NULL,*

*`BrandName` varchar(20) NOT NULL,*

*PRIMARY KEY (`BrandID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**CATEGORYNAME Table**

*CREATE TABLE `categoryname` (*

*`CategoryNameID` int(11) NOT NULL,*

*`CategoryNames` varchar(20) NOT NULL,*

*PRIMARY KEY (`CategoryNameID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**PATTERN Table**

*CREATE TABLE `pattern` (*

*`PatternID` int(11) NOT NULL,*

*`PatternName` varchar(20) NOT NULL,*

*PRIMARY KEY (`PatternID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**SIZE Table**

*CREATE TABLE `size` (*

*`SizeID` int(11) NOT NULL,*

*`SizeName` varchar(20) NOT NULL,*

*PRIMARY KEY (`SizeID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**COLOR Table**

*CREATE TABLE `color` (*

*`ColorID` int(11) NOT NULL,*

*`ColorCode` varchar(20) NOT NULL,*

*PRIMARY KEY (`ColorID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**TYPE Table**

*CREATE TABLE `type` (*

*`TypeID` int(11) NOT NULL,*

*`TypeName` varchar(20) NOT NULL,*

*PRIMARY KEY (`TypeID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**PRODUCT Table**

*CREATE TABLE `product` (*

*`ProductID` int(11) NOT NULL,*

*`ProductName` varchar(20) NOT NULL,*

*`ProductPrice` int(20) NOT NULL,*

*`CategoryID` int(11) NOT NULL,*

*`Barcode` int(20) NOT NULL,*

*`CustomerID` int(11) NOT NULL,*

*PRIMARY KEY (`ProductID`),*

*KEY `CustomerID` (`CustomerID`),*

*CONSTRAINT `product\_ibfk\_1` FOREIGN KEY (`CustomerID`) REFERENCES `customer` (`CustomerID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**RETURNN Table**

*CREATE TABLE `returnn` (*

*`ReturnID` int(11) NOT NULL,*

*`ReturnNumber` int(20) NOT NULL,*

*`ReturnDate` date NOT NULL,*

*`ReturnExplanation` varchar(300) NOT NULL,*

*`ProductID` int(11) NOT NULL,*

*`CustomerID` int(11) NOT NULL,*

*PRIMARY KEY (`ReturnID`),*

*KEY `ProductID` (`ProductID`),*

*KEY `CustomerID` (`CustomerID`),*

*CONSTRAINT `returnn\_ibfk\_1` FOREIGN KEY (`ProductID`) REFERENCES `product` (`ProductID`),*

*CONSTRAINT `returnn\_ibfk\_2` FOREIGN KEY (`CustomerID`) REFERENCES `customer` (`CustomerID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**EXCHANGE Table**

*CREATE TABLE `exchange` (*

*`ExchangeID` int(11) NOT NULL,*

*`ExchangeNumber` int(20) NOT NULL,*

*`ExchangeDate` date NOT NULL,*

*`ExchangeExplanation` varchar(300) NOT NULL,*

*`ProductID` int(11) NOT NULL,*

*`CustomerID` int(11) NOT NULL,*

*PRIMARY KEY (`ExchangeID`),*

*KEY `ProductID` (`ProductID`),*

*KEY `CustomerID` (`CustomerID`),*

*CONSTRAINT `exchange\_ibfk\_1` FOREIGN KEY (`ProductID`) REFERENCES `product` (`ProductID`),*

*CONSTRAINT `exchange\_ibfk\_2` FOREIGN KEY (`CustomerID`) REFERENCES `customer` (`CustomerID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**CUSTOMER Table**

*CREATE TABLE `customer` (*

*`CustomerID` int(11) NOT NULL,*

*`CustomerName` varchar(20) NOT NULL,*

*`CustomerSurname` varchar(20) NOT NULL,*

*`CustomerPhoneNum` int(15) NOT NULL,*

*PRIMARY KEY (`CustomerID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**SALES Table**

*CREATE TABLE `sales` (*

*`SalesID` int(11) NOT NULL,*

*`SalesNumber` int(20) NOT NULL,*

*`SalesDate` date NOT NULL,*

*`ProductID` int(11) NOT NULL,*

*`StoreID` int(11) NOT NULL,*

*`PersonnelID` int(11) NOT NULL,*

*`CustomerID` int(11) NOT NULL,*

*PRIMARY KEY (`SalesID`),*

*KEY `ProductID` (`ProductID`),*

*KEY `PersonnelID` (`PersonnelID`),*

*KEY `CustomerID` (`CustomerID`),*

*KEY `StoreID` (`StoreID`),*

*CONSTRAINT `sales\_ibfk\_1` FOREIGN KEY (`ProductID`) REFERENCES `product` (`ProductID`),*

*CONSTRAINT `sales\_ibfk\_2` FOREIGN KEY (`PersonnelID`) REFERENCES `personnel` (`PersonnelID`),*

*CONSTRAINT `sales\_ibfk\_3` FOREIGN KEY (`CustomerID`) REFERENCES `customer` (`CustomerID`),*

*CONSTRAINT `sales\_ibfk\_4` FOREIGN KEY (`StoreID`) REFERENCES `store` (`StoreID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**BILL Table**

*CREATE TABLE `bill` (*

*`BillID` int(11) NOT NULL,*

*`SalesID` int(11) NOT NULL,*

*PRIMARY KEY (`BillID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**STORE Table**

*CREATE TABLE `store` (*

*`StoreID` int(11) NOT NULL,*

*`BranchName` varchar(30) NOT NULL,*

*`StorePhoneNumber` int(15) NOT NULL,*

*`StoreAddress` varchar(200) NOT NULL,*

*`City` varchar(20) NOT NULL,*

*`DepartmentID` int(11) NOT NULL,*

*PRIMARY KEY (`StoreID`),*

*KEY `DepartmentID` (`DepartmentID`),*

*CONSTRAINT `store\_ibfk\_1` FOREIGN KEY (`DepartmentID`) REFERENCES `department` (`DepartmentID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**DEPARTMENT Table**

*CREATE TABLE `department` (*

*`DepartmentID` int(11) NOT NULL,*

*`DepartmentName` varchar(50) NOT NULL,*

*PRIMARY KEY (`DepartmentID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**STOCK Table**

*CREATE TABLE `stock` (*

*`StockID` int(11) NOT NULL,*

*`StockNumber` int(20) NOT NULL,*

*`ProductID` int(11) NOT NULL,*

*`StoreID` int(11) NOT NULL,*

*PRIMARY KEY (`StockID`),*

*KEY `ProductID` (`ProductID`),*

*KEY `StoreID` (`StoreID`),*

*CONSTRAINT `stock\_ibfk\_1` FOREIGN KEY (`ProductID`) REFERENCES `product` (`ProductID`),*

*CONSTRAINT `stock\_ibfk\_2` FOREIGN KEY (`StoreID`) REFERENCES `store` (`StoreID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**PERSONNEL Table**

*CREATE TABLE `personnel` (*

*`PersonnelID` int(11) NOT NULL,*

*`PersonnelName` varchar(20) NOT NULL,*

*`PersonnelSurname` varchar(20) NOT NULL,*

*`PersonnelPhoneNum` int(15) NOT NULL,*

*`PersonnelAddress` varchar(200) NOT NULL,*

*`EntryDate` date NOT NULL,*

*`Salary` int(20) NOT NULL,*

*`DepartmentID` int(11) NOT NULL,*

*`PositionID` int(11) NOT NULL,*

*`StoreID` int(11) NOT NULL,*

*PRIMARY KEY (`PersonnelID`),*

*KEY `PositionID` (`PositionID`),*

*KEY `DepartmentID` (`DepartmentID`),*

*KEY `StoreID` (`StoreID`),*

*CONSTRAINT `personnel\_ibfk\_1` FOREIGN KEY (`PositionID`) REFERENCES `position` (`PositionID`),*

*CONSTRAINT `personnel\_ibfk\_2` FOREIGN KEY (`DepartmentID`) REFERENCES `department` (`DepartmentID`),*

*CONSTRAINT `personnel\_ibfk\_3` FOREIGN KEY (`StoreID`) REFERENCES `store` (`StoreID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**USER Table**

*CREATE TABLE `user` (*

*`UserID` int(11) NOT NULL,*

*`UserName` varchar(20) NOT NULL,*

*`Password` varchar(20) NOT NULL,*

*`Authorization` varchar(20) NOT NULL,*

*`PersonnelID` int(11) NOT NULL,*

*PRIMARY KEY (`UserID`),*

*KEY `PersonnelID` (`PersonnelID`),*

*CONSTRAINT `user\_ibfk\_1` FOREIGN KEY (`PersonnelID`) REFERENCES `personnel` (`PersonnelID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

**POSITION Table**

*CREATE TABLE `position` (*

*`PositionID` int(11) NOT NULL,*

*`PositionName` varchar(20) NOT NULL,*

*PRIMARY KEY (`PositionID`)*

*) ENGINE=InnoDB DEFAULT CHARSET=utf8*

## Insertion Scripts

**Insertion to PERSONNEL Table**

*INSERT INTO `Personnel` (`PersonnelId`, `PersonnelName`, `PersonnelSurname`, `PersonnelPhoneNum`, `PersonnelAddress`, `EntryDate`, `Salary`, `DepartmentId`, `PositionId`, `StoreId`) VALUES*

*(1, 'yasin', 'deliak', 123, 'Yalıhuyuk', '2017-05-01', 1231, 1, 3, 1);*

**Insertion to USER Table**

*INSERT INTO `User` (‘UserID`, `UserName`, `Password`, `Authorization`, `PersonnelId`) VALUES*

*(1, 'alperen', ‘1234’, ‘enable’, 1);*

**Insertion to PRODUCT Table**

*INSERT INTO `Product` (`ProductId`, `ProductName`, `ProductPrice`, `CategroyID`, `Barcode`,’CustomerID’) VALUES*

*(1, 'WhiteTshirt', 123, 44, 2,2);*

**Insertion to DEPARTMENT Table**

*INSERT INTO `Department` (`DepartmentId`, `DepartmentName`) VALUES*

*(1, 'Human Resources');*

**Insertion to STORE Table**

*INSERT INTO `Store` (`StoreId`, `BranchName`,’StorePhoneNumber’, `StoreAddress`, `City`, `DepartmanId`) VALUES*

*(1, 'Sivas', 1234, 'hebeke', 'sivas', 2);*

**Insertion to POSITION Table**

*INSERT INTO `Position` (`PositionId`, `PositionName`) VALUES*

*(1, 'Cashier');*

**Insertion to CATEGORY Table**

*INSERT INTO `Category` (`CategoryId`, `CategoryNameId`, `SizeId`, `BrandId`, `PatternId`, `ColorId`, `TypeId`) VALUES*

*(5, 1, 2, 2, 2, 1, 1),*

*(6, 1, 2, 2, 2, 1, 1),*

*(7, 1, 2, 2, 2, 1, 1),*

*(8, 1, 2, 2, 2, 1, 1);*

**Insertion to CATEGORYNAME Table**

*INSERT INTO `CategoryName` (`CategoryNameId`, `CategoryName`) VALUES*

*(1, 'Trouser');*

**Insertion to SIZE Table**

*INSERT INTO `Size` (`SizeId`, `SizeName`) VALUES*

*(2, 'Small'),*

*(3, 'Medium');*

**Insertion to BRAND Table**

*INSERT INTO `Brand` (`BrandId`, ‘BrandName`) VALUES*

*(2, 'CalvinKlein');*

**Insertion to COLOR Table**

*INSERT INTO `Color` (`ColorID`, `ColorCode`) VALUES*

*(1, 3),*

*(2, 33);*

**Insertion to STOCK Table**

*INSERT INTO `Stock` (`StockId`, `StockNumber`, `ProductId`, `StoreId`) VALUES*

*(1, 3123, 2, 1);*

**Insertion to TYPE Table**

*INSERT INTO `Type` (`TypeId`, `TypeName`) VALUES*

*(1, 'Jean');*

**Insertion to PATTERN Table**

*INSERT INTO `Pattern` (`PatternID`, `PatternName`) VALUES*

*(2, 'Symetrical');*

## Screenshots

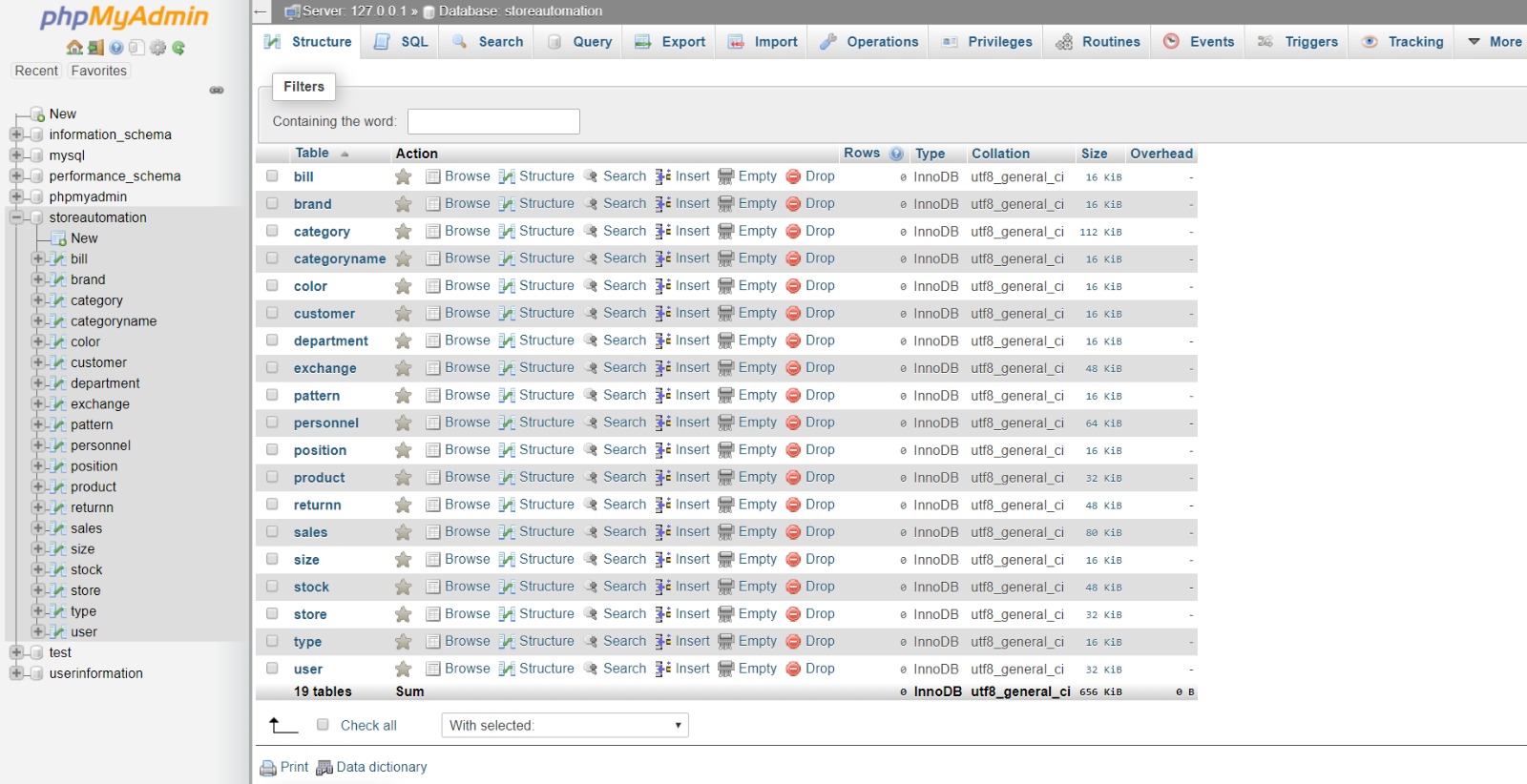
****

Figure 4: Tables

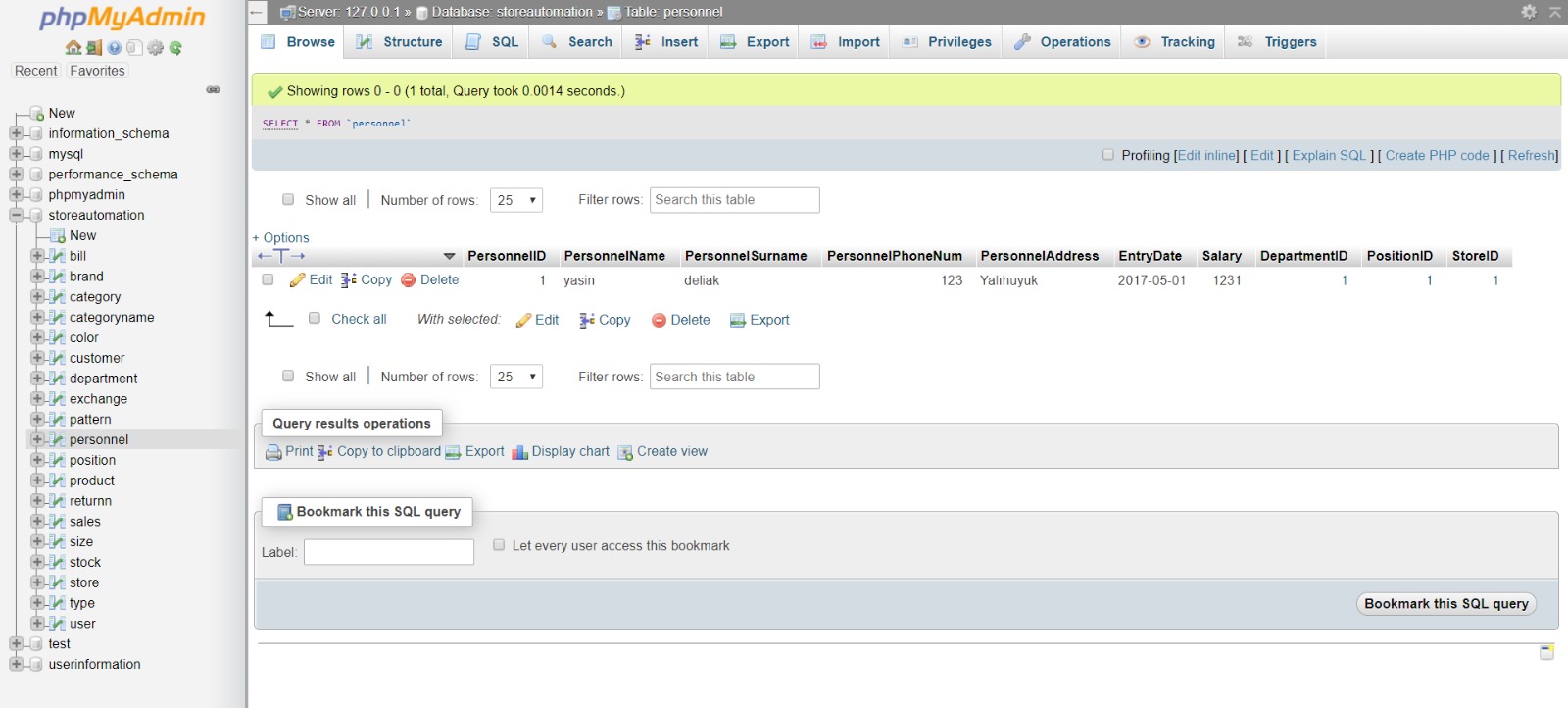


Figure 5: Inserted Personnel Table

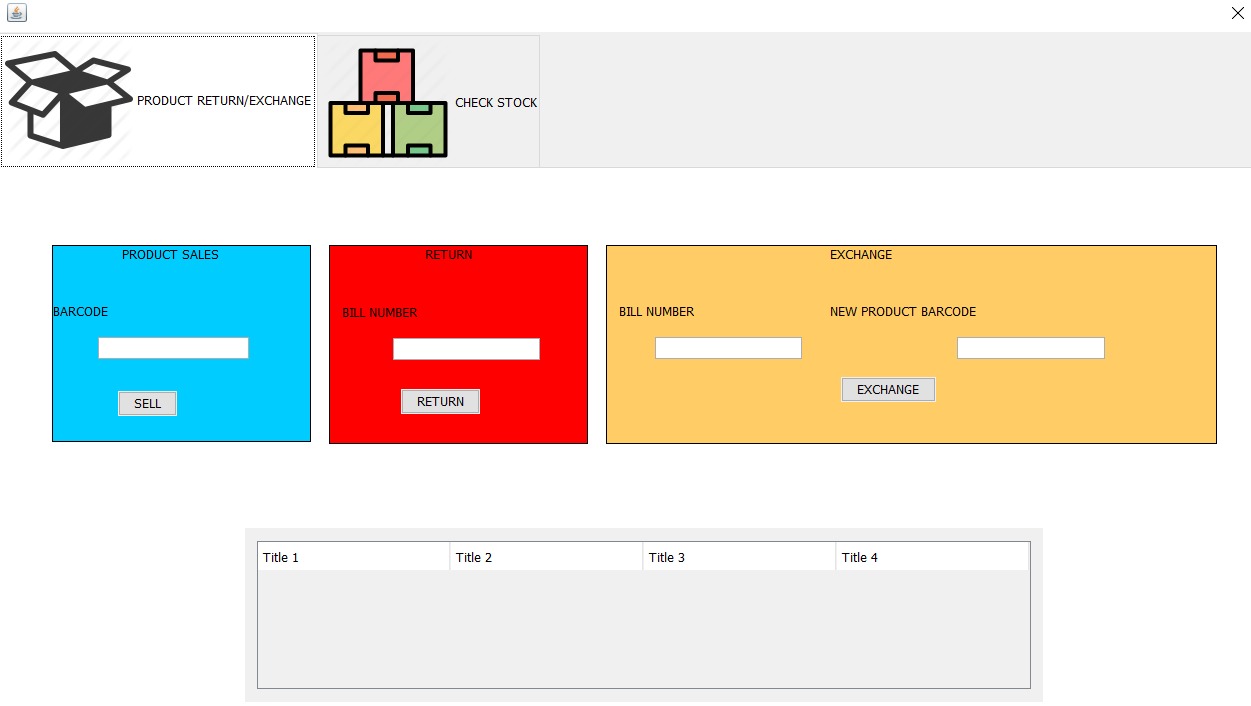


Figure 6: User Panel Interface

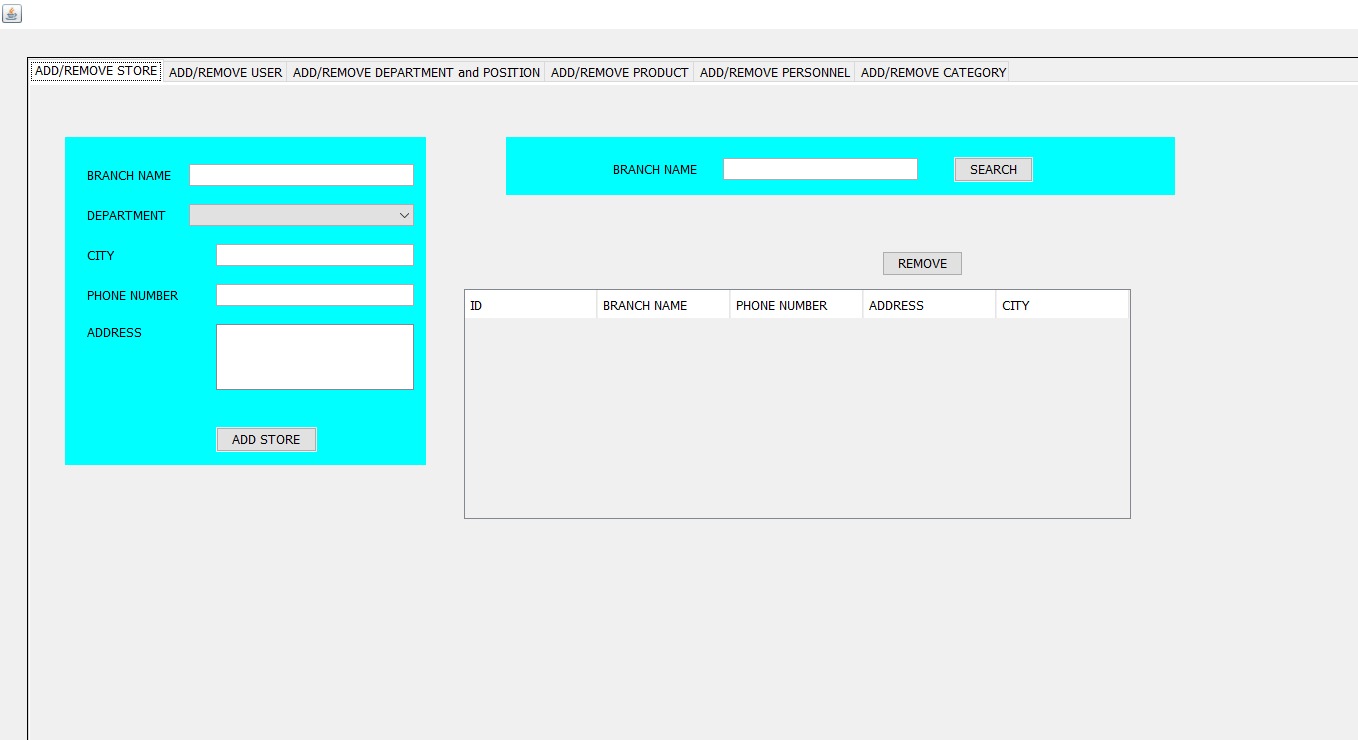


Figure 7: Admin Panel Interface