Zewail City of Science, Technology and Innovation

University of Science and Technology

School of Computational Sciences and Artificial Intelligence

CSAI 202 - Fall 2025

Introduction to Database Systems

Supply Chain Management System

Database Schema Report

Team Members:

Mahmoud Sakr ID: [202402130]
Mohmed Tamer ID: [202402014]
Ahmed Noman ID: [202401324]

Contact:

s-mahmoud.shrief@zewailcity.edu.eg
s-mohmed.refaay@zewailcity.edu.eg
s-ahmed.nabe@zewailcity.edu.eg

1. Final ER Diagram

The Entity-Relationship diagram below represents the conceptual design of our Supply Chain Management System, showing all entities, attributes, relationships, and cardinalities.

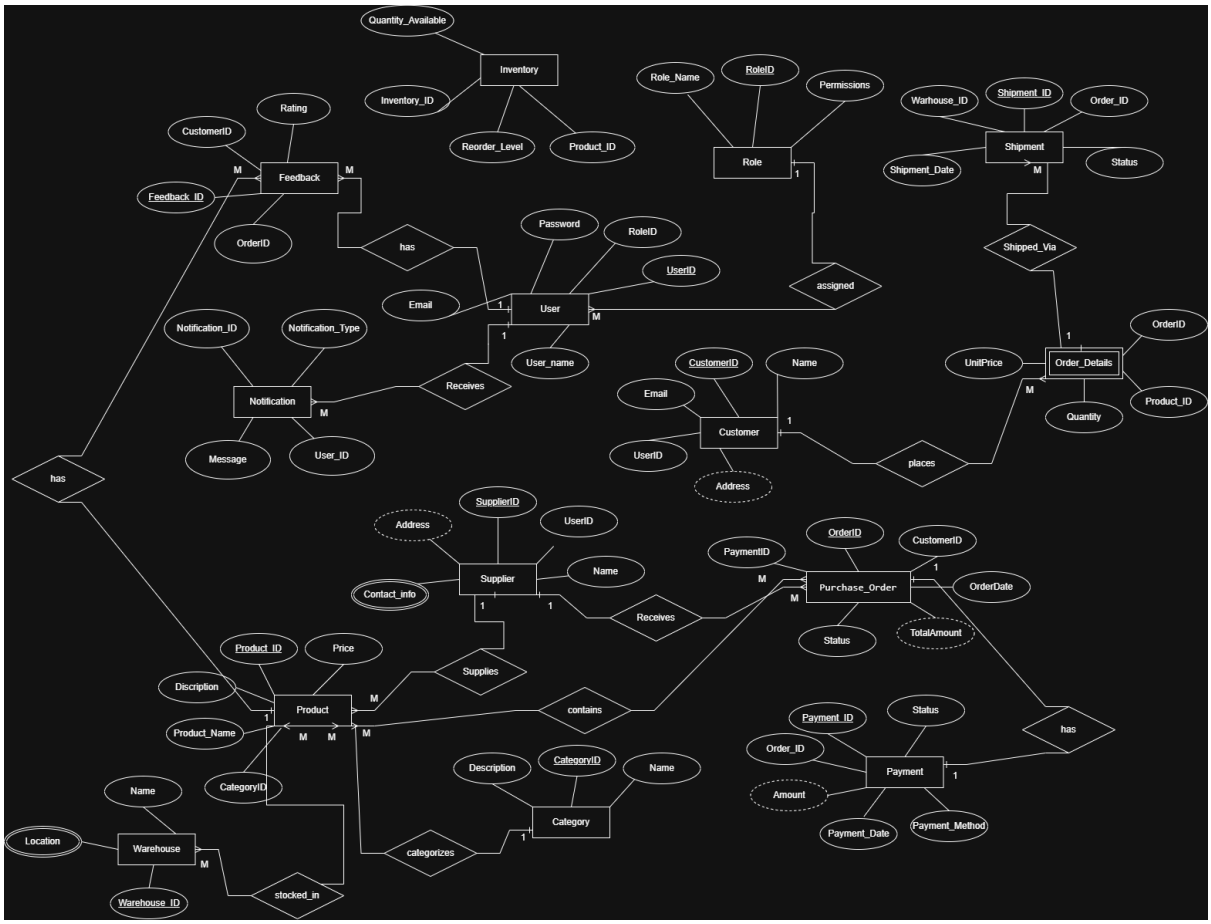

Figure 1: Entity-Relationship Diagram for Supply Chain Management System

Note: The ER diagram shows entities such as User, Role, Customer, Product, Category, Warehouse, Inventory, Supplier, Purchase Order, Payment, Shipment, Notification, and Feedback with their respective attributes and relationships.

2. Database Schema Diagram

This section presents the relational database schema derived from the ER diagram, showing all tables with their primary keys (PK) and foreign keys (FK).

Database Relations

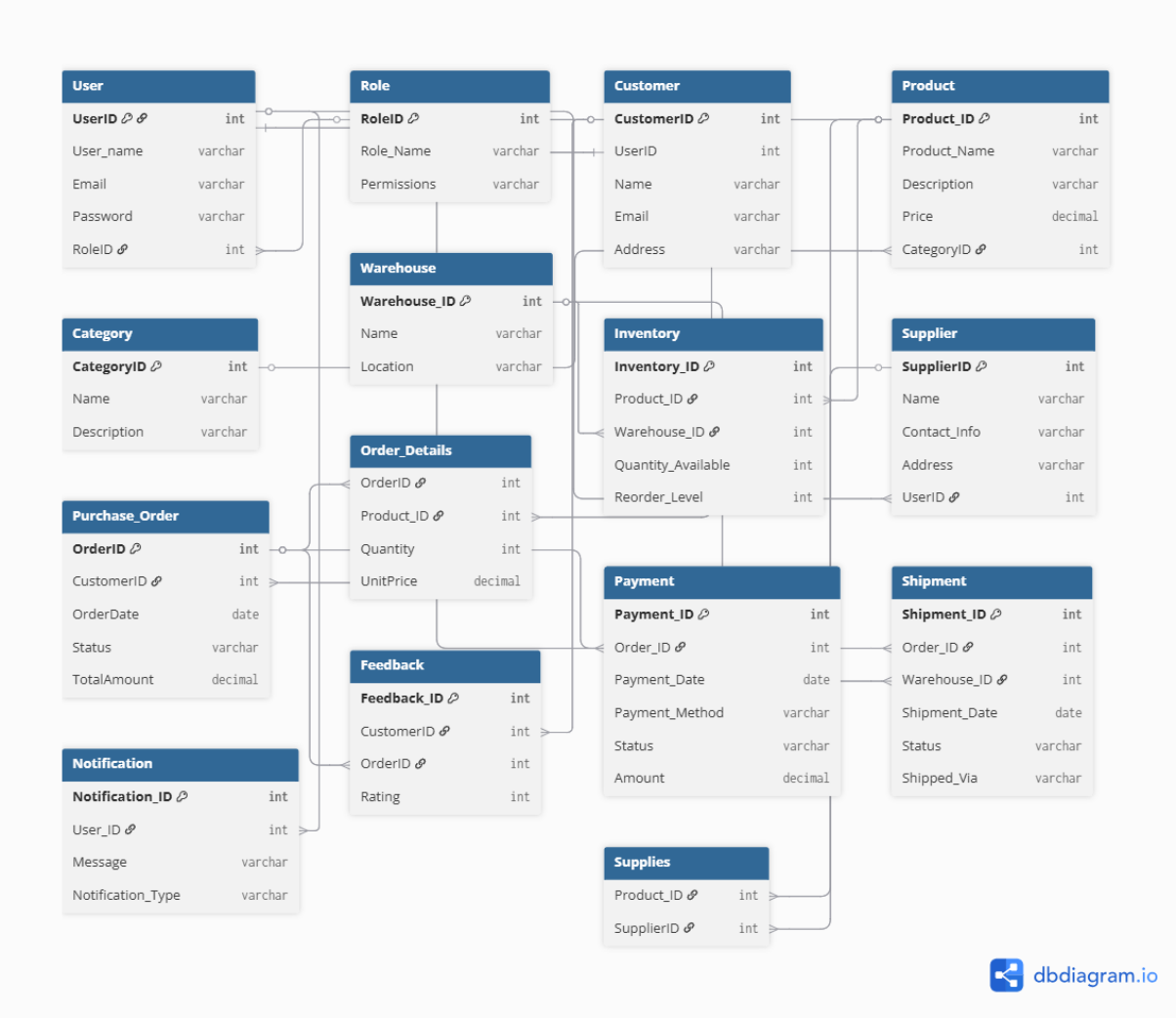

Figure 2: Database Schema - Relational Model with Keys

ER to Relational Mapping

The following mapping was applied to convert the ER diagram to relational schema:

- **Strong Entities:** Mapped to tables with primary keys (Role, User, Customer, Product, Category, Warehouse, Supplier, Purchase_Order, Payment, Shipment, Notification, Feedback)
- **Weak Entity:** Inventory depends on Product and Warehouse
- **1:M Relationships:** Foreign keys added to the "many" side (e.g., User.RoleID references Role)

- **M:N Relationships:** Created junction tables (Order_Details, Supplies)
- **1:1 Relationships:** Customer.UserID is unique, linking one customer to one user

List of Tables with Keys

Table Name	**Primary Key(s)**	**Foreign Key(s)**
Role	RoleID	-
User	UserID	RoleID → Role
Customer	CustomerID	UserID → User
Category	CategoryID	-
Product	Product_ID	CategoryID → Category
Warehouse	Warehouse_ID	-
Inventory	Inventory_ID	Product_ID → Product, Warehouse_ID → Warehouse
Supplier	SupplierID	UserID → User
Purchase_Order	OrderID	CustomerID → Customer
Order_Details	OrderID, Product_ID	OrderID → Purchase_Order, Product_ID → Product
Payment	Payment_ID	Order_ID → Purchase_Order
Shipment	Shipment_ID	Order_ID → Purchase_Order, Warehouse_ID → Warehouse
Notification	Notification_ID	User_ID → User
Feedback	Feedback_ID	CustomerID → Customer, OrderID → Purchase_Order
Supplies	Product_ID, SupplierID	Product_ID → Product, SupplierID → Supplier

Table 1: Complete list of tables with primary and foreign keys

3. SQL Scripts for Database Creation

The following SQL scripts were used to create the database schema in Microsoft SQL Server:

Listing 1: Role and User Tables

```
CREATE TABLE [Role] (
  [RoleID] int PRIMARY KEY IDENTITY(1,1),
  [Role_Name] nvarchar(255),
  [Permissions] nvarchar(255)
);
GO

CREATE TABLE [User] (
  [UserID] int PRIMARY KEY IDENTITY(1,1),
  [User_name] nvarchar(255),
  [Email] nvarchar(255) UNIQUE NOT NULL,
  [Password] nvarchar(255),
  [RoleID] int,
  FOREIGN KEY (RoleID) REFERENCES [Role](RoleID)
);
GO
```

Listing 2: Customer and Category Tables

```
CREATE TABLE [Customer] (
  [CustomerID] int PRIMARY KEY IDENTITY(1,1),
  [UserID] int UNIQUE,
  [Name] nvarchar(255),
  [Email] nvarchar(255) UNIQUE NOT NULL,
  [Address] nvarchar(255),
  FOREIGN KEY (UserID) REFERENCES [User](UserID)
);
GO

CREATE TABLE [Category] (
  [CategoryID] int PRIMARY KEY IDENTITY(1,1),
  [Name] nvarchar(255),
  [Description] nvarchar(255)
);
GO
```

Listing 3: Product, Warehouse, and Inventory Tables

```
CREATE TABLE [Product] (
  [Product_ID] int PRIMARY KEY IDENTITY(1,1),
  [Product_Name] nvarchar(255),
  [Description] nvarchar(255),
  [Price] decimal(10,2),
  [CategoryID] int,
  FOREIGN KEY (CategoryID) REFERENCES [Category](CategoryID)
);
GO

CREATE TABLE [Warehouse] (
  [Warehouse_ID] int PRIMARY KEY IDENTITY(1,1),
  [Name] nvarchar(255),
  [Location] nvarchar(255)
```

```
);
GO

CREATE TABLE [Inventory] (
  [Inventory_ID] int PRIMARY KEY IDENTITY(1,1),
  [Product_ID] int,
  [Warehouse_ID] int,
  [Quantity_Available] int,
  [Reorder_Level] int,
  FOREIGN KEY (Product_ID) REFERENCES [Product](Product_ID),
  FOREIGN KEY (Warehouse_ID) REFERENCES [Warehouse](Warehouse_ID)
);
GO
```

Listing 4: Supplier and Purchase Order Tables

```
CREATE TABLE [Supplier] (
  [SupplierID] int PRIMARY KEY IDENTITY(1,1),
  [Name] nvarchar(255),
  [Contact_Info] nvarchar(255),
  [Address] nvarchar(255),
  [UserID] int,
  FOREIGN KEY (UserID) REFERENCES [User](UserID)
);
GO

CREATE TABLE [Purchase_Order] (
  [OrderID] int PRIMARY KEY IDENTITY(1,1),
  [CustomerID] int,
  [OrderDate] date,
  [Status] nvarchar(255),
  [TotalAmount] decimal(10,2),
  FOREIGN KEY (CustomerID) REFERENCES [Customer](CustomerID)
);
GO

CREATE TABLE [Order_Details] (
  [OrderID] int,
  [Product_ID] int,
  [Quantity] int,
  [UnitPrice] decimal(10,2),
  CONSTRAINT PK_OrderDetails PRIMARY KEY (OrderID, Product_ID),
  FOREIGN KEY (OrderID) REFERENCES [Purchase_Order](OrderID),
  FOREIGN KEY (Product_ID) REFERENCES [Product](Product_ID)
);
GO
```

Listing 5: Payment, Shipment, and Supporting Tables

```
CREATE TABLE [Payment] (
  [Payment_ID] int PRIMARY KEY IDENTITY(1,1),
  [Order_ID] int,
  [Payment_Date] date,
  [Payment_Method] nvarchar(255),
  [Status] nvarchar(255),
  [Amount] decimal(10,2),
  FOREIGN KEY (Order_ID) REFERENCES [Purchase_Order](OrderID)
);
```

```
GO

CREATE TABLE [Shipment] (
  [Shipment_ID] int PRIMARY KEY IDENTITY(1,1),
  [Order_ID] int,
  [Warehouse_ID] int,
  [Shipment_Date] date,
  [Status] nvarchar(255),
  [Shipped_Via] nvarchar(255),
  FOREIGN KEY (Order_ID) REFERENCES [Purchase_Order](OrderID),
  FOREIGN KEY (Warehouse_ID) REFERENCES [Warehouse](Warehouse_ID)
);
GO

CREATE TABLE [Notification] (
  [Notification_ID] int PRIMARY KEY IDENTITY(1,1),
  [User_ID] int,
  [Message] nvarchar(255),
  [Notification_Type] nvarchar(255),
  FOREIGN KEY (User_ID) REFERENCES [User](UserID)
);
GO

CREATE TABLE [Feedback] (
  [Feedback_ID] int PRIMARY KEY IDENTITY(1,1),
  [CustomerID] int,
  [OrderID] int,
  [Rating] int,
  FOREIGN KEY (CustomerID) REFERENCES [Customer](CustomerID),
  FOREIGN KEY (OrderID) REFERENCES [Purchase_Order](OrderID)
);
GO

CREATE TABLE [Supplies] (
  [Product_ID] int,
  [SupplierID] int,
  CONSTRAINT PK_Supplies PRIMARY KEY (Product_ID, SupplierID),
  FOREIGN KEY (Product_ID) REFERENCES [Product](Product_ID),
  FOREIGN KEY (SupplierID) REFERENCES [Supplier](SupplierID)
);
GO
```