# Online Shopping Application

## 1. Database Design

Based on the project requirements, the following tables are proposed:

### Users

* - UserID (Primary Key)
* - Name
* - Email (Unique)
* - Password
* - Address
* - Phone
* - Role (User/Admin)

### Products

* - ProductID (Primary Key)
* - Name
* - Description
* - Price
* - Stock
* - CategoryID (Foreign Key)
* - ImageURL

### Categories

* - CategoryID (Primary Key)
* - CategoryName

### Orders

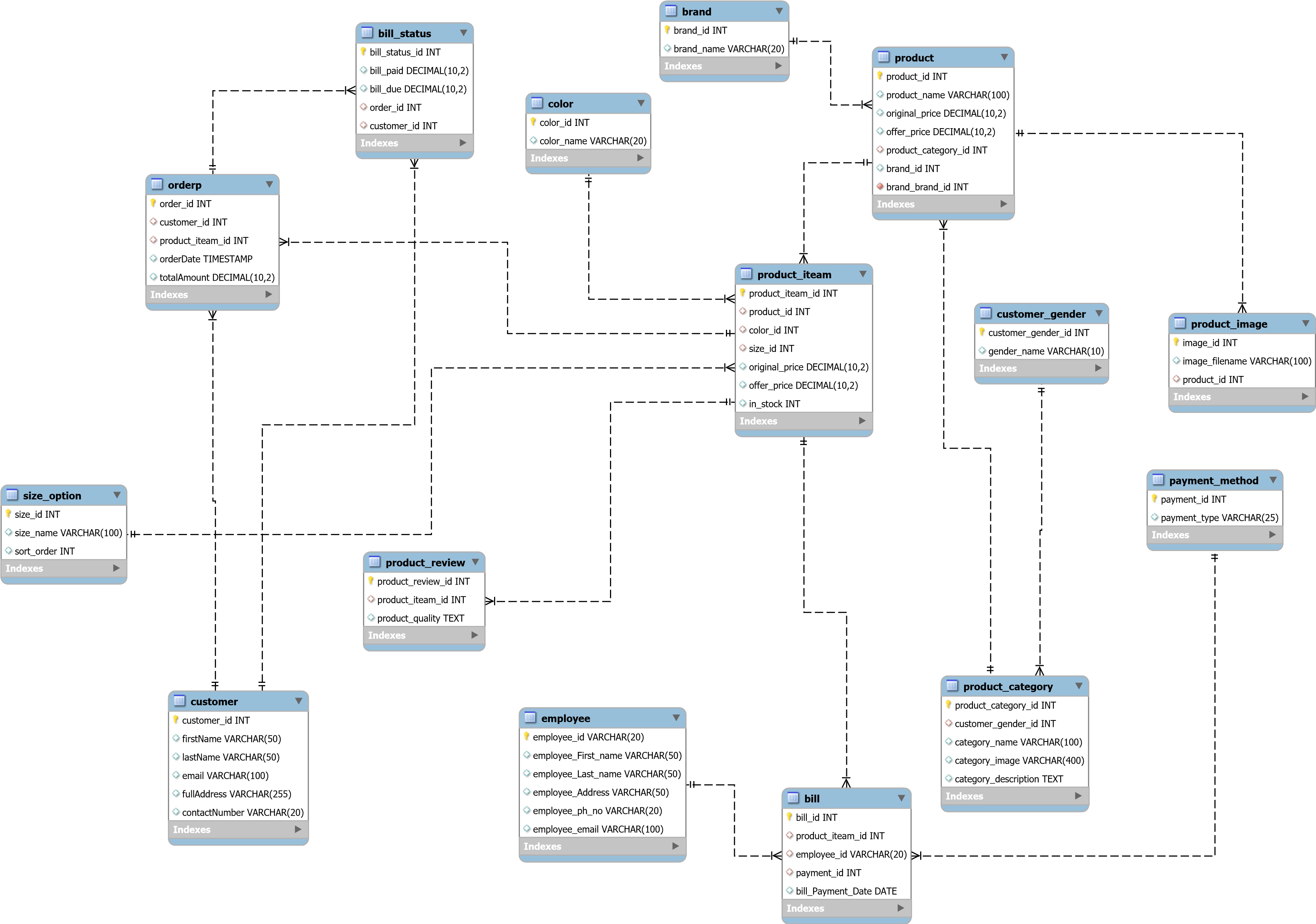
* - OrderID (Primary Key)
* - UserID (Foreign Key)
* - OrderDate
* - TotalAmount
* - Status (Pending/Shipped/Delivered)

### OrderDetails

* - OrderDetailID (Primary Key)
* - OrderID (Foreign Key)
* - ProductID (Foreign Key)
* - Quantity
* - Subtotal

### Cart

* - CartID (Primary Key)
* - UserID (Foreign Key)
* - ProductID (Foreign Key)
* - Quantity

**2. ER Diagram:** The ER diagram visually represents the relationships between the entities. The main entities include Users, Products, Categories, Orders, OrderDetails, Cart etc. Attributes and primary keys are defined for each’

**3. Normalization**

Normalization ensures that the database is structured efficiently:

**1. First Normal Form (1NF)**

- Ensure that all attributes contain atomic values.

- Example: Splitting full names into FirstName and LastName.

**2. Second Normal Form (2NF)**

- Remove partial dependencies; all non-key attributes are dependent on the entire primary key.

- Example: Separate order details into an OrderDetails table to remove duplication.

**3. Third Normal Form (3NF)**

- Remove transitive dependencies; non-key attributes should not depend on other non-key

attributes.

- Example: Store categories in a separate table to eliminate redundant data in the Products table