

## **Project Documentation**



**Submitted By:**

**Zohra Amna 2022-SE-3**

**Muhammad Noman 2022-SE-31**

**Submitted To:**

**Dr. Umer Qasim**

**Department of Computer Science**

**University of Engineering and Technology,**

**Lahore, New Campus**

# E-Commerce Platform Backend Documentation

## Contents

<b>Project Documentation .....</b>	<b>1</b>
<b>Submitted By:.....</b>	<b>1</b>
<b>Submitted To: .....</b>	<b>1</b>
<b>Introduction.....</b>	<b>4</b>
<b>Backend Architecture .....</b>	<b>4</b>
<b>Overview .....</b>	<b>4</b>
<b>Technologies Used .....</b>	<b>4</b>
<b>Database Schema .....</b>	<b>4</b>
<b>Entities and Attributes: .....</b>	<b>4</b>
1. Users .....	4
2. Products .....	4
3. Categories .....	4
4. Orders .....	4
5. Payments.....	5
6. Reviews.....	5
7. Addresses .....	5
8. Cart.....	5
9. Coupons .....	5
10. Inventory.....	6
<b>Relationships .....</b>	<b>6</b>
1. One-to-Many Relationship:.....	6
2. Many-to-One Relationship:.....	6
3. Many-to-Many Relationship:.....	6
<b>Tables.....</b>	<b>6</b>
<b>Functionality and Features .....</b>	<b>6</b>
1. Product Management: .....	6
2. Order Processing:.....	7
3. Payment Handling: .....	7
4. Review and Rating System:.....	7
5. Inventory Management: .....	7
6. Coupon and Discount Management:.....	7
7. Search and Filtering: .....	7

# E-Commerce Platform Backend Documentation

<b>8. User Account Management:</b>	<b>8</b>
<b>9. Analytics and Reporting:</b>	<b>8</b>
Views and Triggers	8
Views	8
Triggers	8
Diagram	9
Deployment	10
Conclusion	10

## Table of Figures

Figure 1 ERD	9
Figure 2 EERD	10

# E-Commerce Platform Backend Documentation

## Introduction

This documentation provides an overview of the backend architecture and database design for the E-Commerce platform project. It outlines the functionalities implemented, the database schema, and key features of the backend system.

## Backend Architecture

### Overview

The backend architecture consists of several components responsible for handling user requests, processing business logic, and interacting with the database.

### Technologies Used

- Database: MySQL

## Database Schema

### Entities and Attributes:

#### 1. Users

- UserID (Primary Key)
- Username (String)
- Email (String)
- Password (String)
- UserType (String: customer, administrator)

#### 2. Products

- ProductID (Primary Key)
- Name (String)
- Description (Text)
- Price (Decimal)
- CategoryID (Foreign Key)

#### 3. Categories

- CategoryID (Primary Key)
- Name (String)

#### 4. Orders

- OrderID (Primary Key)

# E-Commerce Platform Backend Documentation

- UserID (Foreign Key)
- OrderDate (DateTime)
- Status (String: pending, shipped, delivered)

## 5. Payments

- PaymentID (Primary Key)
- OrderID (Foreign Key)
- Amount (Decimal)
- PaymentDate (DateTime)

## 6. Reviews

- ReviewID (Primary Key)
- ProductID (Foreign Key)
- UserID (Foreign Key)
- Rating (Integer)
- Comment (Text)

## 7. Addresses

- AddressID (Primary Key)
- UserID (Foreign Key)
- AddressLine1 (String)
- City (String)
- State (String)
- ZipCode (String)
- Country (String)

## 8. Cart

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

## 9. Coupons

- CouponID (Primary Key)
- Code (String)
- Discount (Decimal)
- ExpiryDate (Date)

# E-Commerce Platform Backend Documentation

## 10. Inventory

- InventoryID (Primary Key)
- ProductID (Foreign Key)
- StockQuantity (Integer)

## Relationships

### 1. One-to-Many Relationship:

- Users to Orders: One user can place multiple orders.
- Orders to Payments: One order can have multiple payments (e.g., partial payments or refunds).
- Products to Reviews: One product can have multiple reviews.
- Users to Addresses: One user can have multiple addresses (e.g., billing address, shipping address).
- Users to Cart: One user can have multiple items in their cart.
- Products to Inventory: One product can have multiple inventory entries (e.g., for different warehouses).

### 2. Many-to-One Relationship:

- Orders to Users: Many orders can be placed by one user.
- Payments to Orders: Many payments can be associated with one order.

### 3. Many-to-Many Relationship:

- Products to Categories: Many products can belong to multiple categories, and a category can contain multiple products.

## Tables

1. **Users:** Stores user information including username, email, password, and user type.
2. **Products:** Contains information about products such as name, description, price, and category.
3. **Orders:** Records details of orders placed by users including order date, status, and user ID.
4. **Payments:** Stores transaction details including payment ID, order ID, amount, and payment date.
5. **Categories:** Defines product categories to organize products.
6. **Reviews:** Stores product reviews provided by users including rating and comments.
7. **Addresses:** Records user addresses for shipping purposes.
8. **Cart:** Temporarily stores selected items before checkout.
9. **Coupons:** Contains information about coupons including code, discount, and expiry date.
10. **Inventory:** Tracks product availability and stock quantities.

## Functionality and Features

### 1. Product Management:

- Administrators can add, edit, and delete products from the product catalog.

# E-Commerce Platform Backend Documentation

- Products are organized into categories to facilitate navigation and browsing.

## 2. Order Processing:

- Users can add products to their cart, review their selections, and proceed to checkout.
- Orders are processed securely, and users receive confirmation emails upon successful completion.

## 3. Payment Handling:

- Integration with payment gateways enables users to make secure online payments.
- Multiple payment methods are supported to accommodate diverse user preferences.

## 4. Review and Rating System:

- Users can leave reviews and ratings for products they have purchased, helping other users make informed decisions.
- Reviews are displayed alongside product listings to provide social proof and enhance credibility.

## 5. Inventory Management:

- Inventory levels are automatically updated based on orders placed, ensuring accurate stock information.
- Administrators can track stock quantities, receive notifications for low inventory, and manage replenishment.

## 6. Coupon and Discount Management:

- Administrators can create and manage coupons offering discounts to users.
- Users can apply valid coupons during checkout to avail discounts on their orders.

## 7. Search and Filtering:

- Users can search for products using keywords and filter results based on various criteria (e.g., category, price range).
- Advanced search functionality enables users to find specific products quickly and efficiently.

# E-Commerce Platform Backend Documentation

## 8. User Account Management:

- Users can view and update their profile information, including shipping addresses and payment methods.
- Password reset functionality allows users to securely regain access to their accounts if needed.

## 9. Analytics and Reporting:

- Built-in analytics tools provide insights into sales performance, customer behavior, and product popularity.
- Administrators can generate reports on key metrics to inform strategic decision-making and marketing efforts.

## Views and Triggers

Views and Triggers are used in backend of this project and details are as follows:

### Views

1. Product Details
2. Total spent by user
3. Retrieve the details of products along with their average ratings
4. List all users along with their total spending and average order value
5. Retrieve the top 5 products with the highest number of reviews
6. Get the total revenue generated by each category
7. Retrieve the details of products that are out of stock
8. Find the top 3 categories with the highest average product prices
9. List all users along with their last order date
10. Retrieve the details of orders that contain discounted items
11. Get the total number of products in each category
12. Retrieve the details of users who have spent more than \$1000 in total

### Triggers

1. Update inventory after order
2. Max discount percentage
3. To enforce a maximum discount percentage for coupons
4. Calculate Total Order Amount
5. Send Email Notification
6. Track User Activity



## Diagram

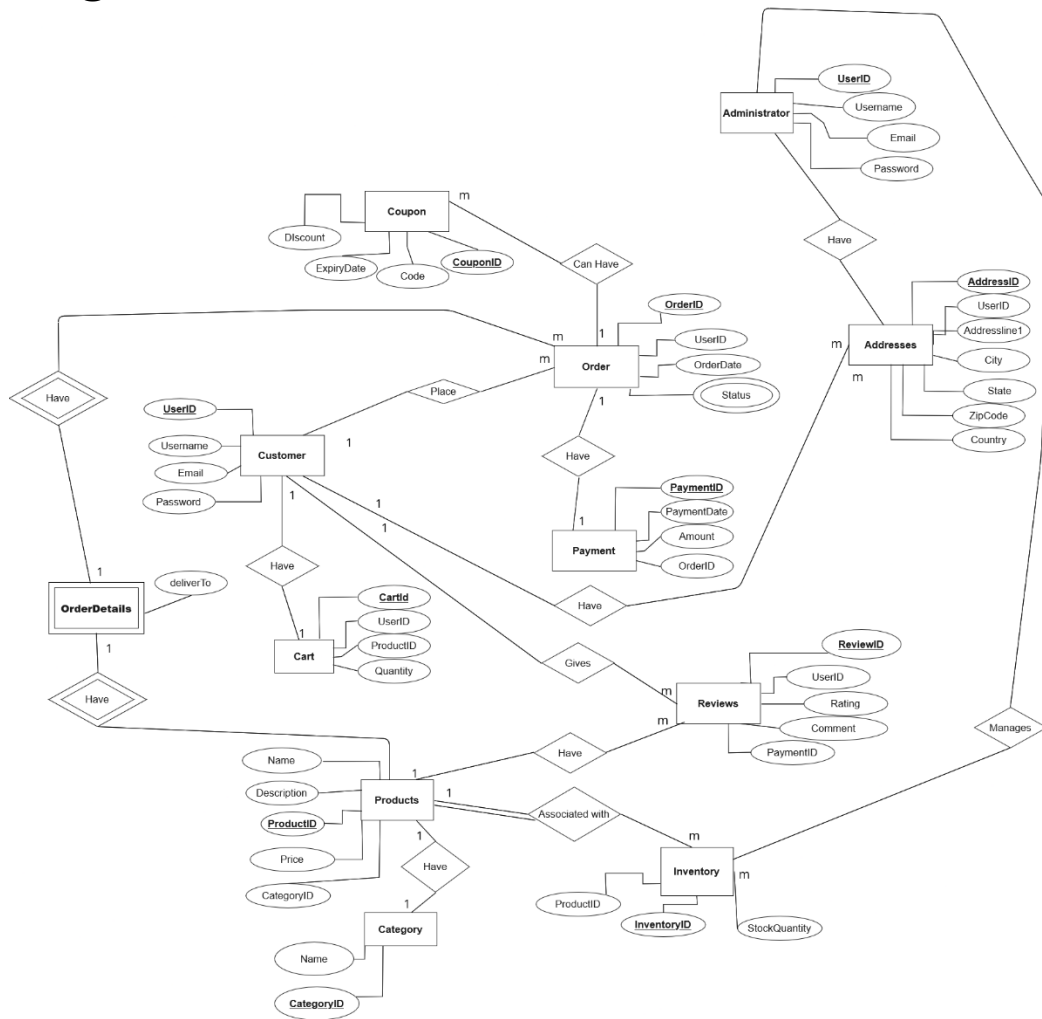


Figure 1 ERD

# E-Commerce Platform Backend Documentation

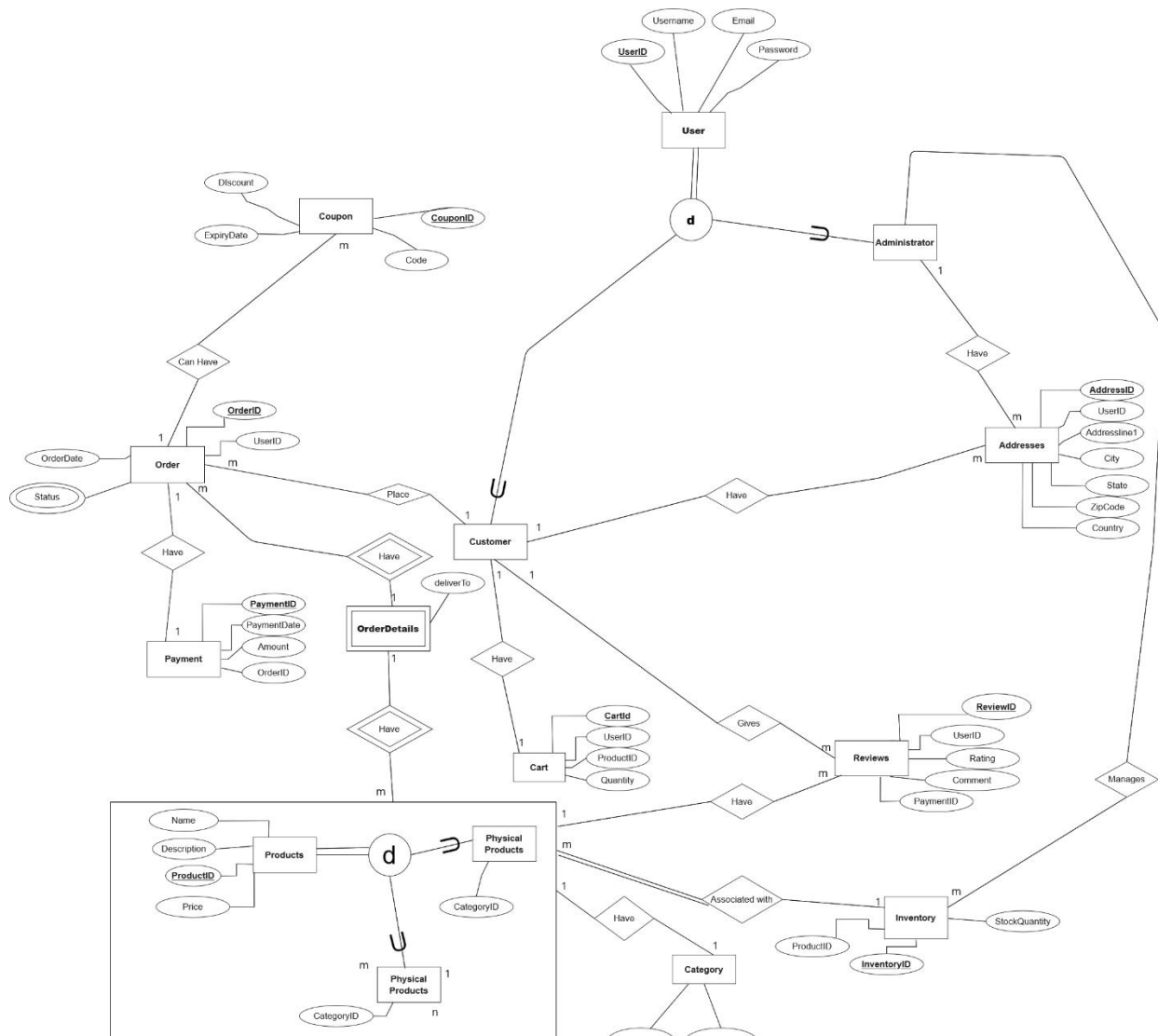


Figure 2 EERD

## Deployment

After completing with the front end well will be connecting this database and deploy to customer.

## Conclusion

In conclusion, this documentation provides a comprehensive overview of the backend architecture and database design for the e-commerce platform project. Through the detailed explanation of the backend components, database schema, functionality, and features, this documentation aims to facilitate a deeper understanding of the system's architecture and operation. MySQL serves as the database management system.

# E-Commerce Platform Backend Documentation

The database schema is structured to efficiently store and manage data related to users, products, orders, payments, categories, reviews, addresses, cart, coupons, and inventory. The Entity-Relationship Diagram (ERD) visually represents the relationships between different entities, aiding in understanding the database structure.

Key functionalities and features of the backend include product management, order processing, payment handling, review and rating system, inventory management, and coupon management. These functionalities are implemented with the help of views, triggers, subqueries, and joins to ensure efficient data retrieval, manipulation, and enforcement of business rules.

Triggers play a crucial role in maintaining data integrity and enforcing business logic within the database. Examples of triggers include updating inventory after orders, enforcing maximum discount percentages for coupons, calculating total order amounts, sending email notifications, and tracking user activity etc.

Overall, this documentation serves as a valuable resource for developers, stakeholders, and users alike, providing insights into the backend architecture, database design, and functionalities of the e-commerce platform project. It lays the foundation for building a robust and scalable system that meets the needs of modern e-commerce businesses.