
Department of Computer Science & Engineering

National Institute of Technology, Warangal

# Database Design of an e-Commerce Platform Database

*Authors :*
(i) Sahil Tejas Shah
Roll 167255
(ii) Samarjit Karmakar
Roll 167256
(iii) Saurabh Bodhe
Roll 167257

*Supervisor :*
Dr. T. Ramakrishnudu
Assistant Professor
Dept of CSE

# Acknowledgement

We would like to thank Dr. T. Ramakrishnudu for giving us the opportunity to do this project as a part of the Database Management Systems course. Through this project we have gained deeper insight in Database Design, Relational Model, ER Model and Normalization.
We have also gained deeper insight on e-Commerce platforms such as Amazon, Flipkart, Snapdeal, etc. and the technologies and design techniques they use for their large databases.

Sahil T. Shah
Samarjit Karmakar
Saurabh Bodhe

# Contents

# Acronyms

| | |
|---|---|
| **DB** | *Database* |
| **DBMS** | *Database Management System* |
| **FD** | *Functional Dependency*' |
| **NF** | *Normal Form* |
| **ECP** | *E-Commerce Platform* |
| **IC** | *Integrity Constraints* |
| **RI** | *Relational Instance* |
| **BCNF** | *Boyce-Codd Normal Form* |

# Introduction

E-commerce is fast gaining ground as an accepted and used business paradigm. More and more business houses are implementing web sites providing functionality for performing commercial transactions over the web. It is reasonable to say that the process of shopping on the web is becoming commonplace.

The objective of this project is to develop a general purpose e-commerce store database where any product (such as books, CDs, computers, mobile phones, electronic items, and home appliances) can be bought from the comfort of home through the Internet.

An online store is a virtual store on the Internet where customers can browse the catalog and select products of interest. The selected items may be collected in a shopping cart. At checkout time, the items in the shopping cart will be presented as an order. At that time, more information will be needed to complete the transaction. Usually, the customer will be asked to fill or select a billing address, a shipping address, a shipping option, and payment information such as credit card number. An e-mail notification is sent to the customer as soon as the order is placed.

In order to design a web site, the relational database must be designed first. Conceptual design can be divided into two parts: The **logical data model** and the **implementation data model**. The logical model focuses on what data should be stored in the database while the implementation model deals with how the data is processed. To put this in the context of the relational database, the logical model is used to design the relational tables. The implementation model is used to design the queries that will access and perform operations on those tables.

# Chapter 1

# Entity Relationship Model

## 1.1 Requirement Analysis

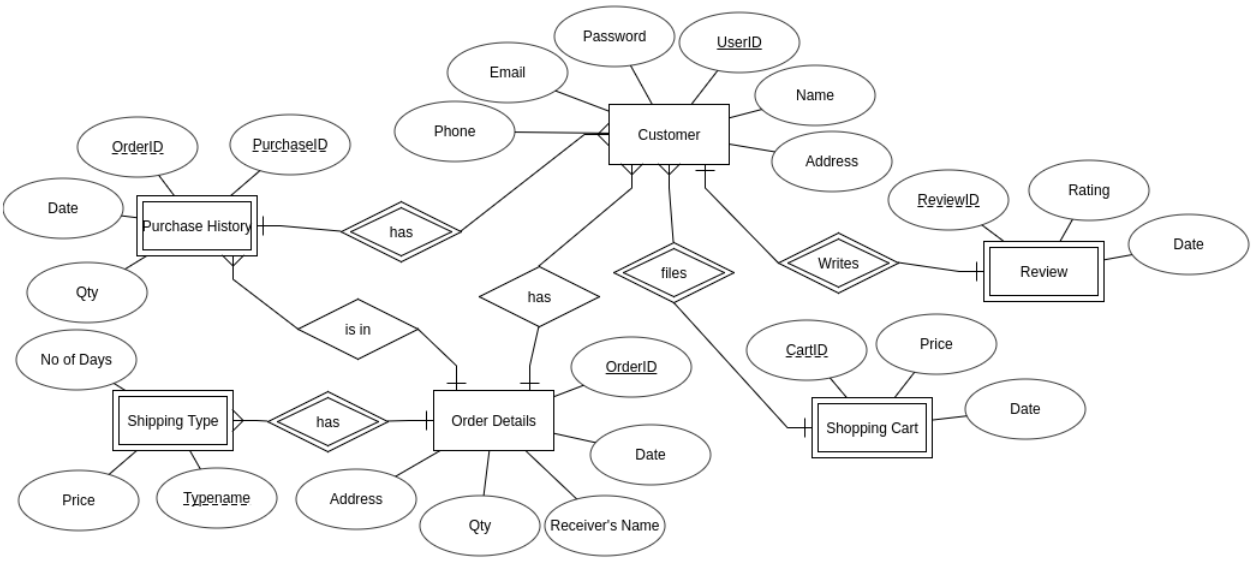
Electronic Commerce (e-commerce) applications support the interaction between different parties participating in a commerce transaction via the network, as well as the management of the data involved in the process.

A good e-commerce site should present the following factors to the customers for better usability:
• Knowing when an item was saved or not saved in the shopping cart.
• Returning to different parts of the site after adding an item to the shopping cart.
• Easy scanning and selecting items in a list.
• Effective categorical organization of products.
• Simple navigation from home page to information and order links for specific products.
• Obvious shopping links or buttons.
• Minimal and effective security notifications or messages.
• Consistent layout of product information.

Another important factor in the design of an e-commerce site is feedback. The interactive cycle between a user and a web site is not complete until the web site responds to a command entered by the user.

Finally, we must have a good database design which is consistent with the needs of the platform.

## 1.2 Building the ER Model




The main entity around which the model is centered is the **Customer**. A customer can perform various functions such as order products, save products in his/her shopping cart and perform a review.

Each of these functions can be an entity, as an entity can have real-world existence, physically present (eg. a person), as well as a virtual existence, not physically present (eg. an order).

Every **Order** which gets dispatched to serve a specific customer has a **Shipping Type** (eg. Fast Delivery, Normal Delivery).

A **Customer** also has a **Purchase History** which he/she may view for previous orders or repeating an order.

# Chapter 2

# Relational Schema & Normal Forms

After building the **Logical Data Model** (here ER Model), the next step is to map it to an **Implementation Data Model** which is present in most commercially viable databases. For this purpose, we will choose the **Relational Model**. The Relational Model will consist of multiple **Relational Schemas** which will better describe our data.
**Note:** We will not create **Relational Instances**, but end our discussion with the Relational Schemas.

## 2.1 Relational Schemas

The following relational schemas have been created by mapping our previously created ER Model (Chapter 1) into the Relational model.

**(i)** ***Customer*****(User ID: INT PRIMARY KEY, Name: VARCHAR, Password: VARCHAR, Email: VARCHAR, Phone: CHAR(10), Address: VARCHAR)**

**(ii)** ***Review*****(ReviewID: INT, UserID: INT FOREIGN KEY REFERENCES *Customer*, Rating: INT, Date: DATE, PRIMARY KEY(ReviewID, UserID))**

**(iii)** ***Shopping Cart*****(CartID: INT, UserID: INT FOREIGN KEY REFERENCES *Customer*, Price: DECIMAL, Date: DATE, PRIMARY KEY(CartID, UserID))**

**(iv)** ***Order Details*****(OrderID: INT PRIMARY KEY, Receivers Name: VARCHAR, Date: DATE, Qty: INT, Address: VARCHAR, UserID: INT FOREIGN KEY REFERENCES *Customer*)**

**(v)** ***Shipping Type*****(Typename: VARCHAR, OrderID: INT FOREIGN KEY REFERENCES** ***Order Details*****, Price: DECIMAL, No of days: INT, PRIMARY KEY(Typename, OrderID))**

**(vi)** ***Purchase History*****(PurchaseID: INT, UserID: INT FOREIGN KEY REFERENCES** ***Customer*****, Date: DATE, Qty: INT, PRIMARY KEY(PurchaseID, UserID))**

**(vii)** ***Purchase_ Order*****(PurchaseID: INT, UserID: INT, OrderID: INT, PRIMARY KEY (PurchaseID, UserID, OrderID))**

We have performed shifting of attributes to support weak entities.

For the benefit of the next section, we declare the schemas as follows:

*Customer* as **R1(UNPEOA)**
*Review* as **R2(RUAD)**
*Shopping Cart* as **R3(CUPD)**
*Order Details* as **R4(ORDQAU)**
*Shipping Type* as **R5(TOPN)**
*Purchase History* as **R6(PUDQ)**
*Purchase_ Order* as **R7(PUO)**
(where each attribute is abbreviated intuitively)

## 2.2 Normal Forms

By analysis of our relational schemas, we can procure various **Functional Dependencies** in them. These FDs will give us the specific **Normal Form** our relational schema is in.

(i) ***Customer***

We have UserID (U) as the primary key. Hence, $U \rightarrow NPEOA$.
No other functional dependencies exist.
Hence, *Customer* is in *Third Normal Form (3NF)* as well as in *BCNF*.

(ii) ***Review***

As *Review* is a weak entity in our ER Model, (ReviewID, UserID) acts as the primary key. Hence, $RU \rightarrow AD$. No other functional dependencies exist.
Hence, *Review* is in *Third Normal Form (3NF)* as well as in *BCNF*.

(iii) ***Shopping Cart***

As *Shopping Cart* is a weak entity in our ER Model, (CartID, UserID) acts as the primary key. Hence, $CU \rightarrow PD$.
The FD $D \rightarrow P$ exists as the prices changes with date, as some products have higher price in peak season and vice versa.
A transitive dependency exists but no partial dependency exists.
Hence, *Shopping Cart* is in *Second Normal Form (2NF).*

(iv) ***Order Details***

We have OrderID (O) as the primary key. Hence, $O \rightarrow RDQAU$.
The FD $R \rightarrow A$ as the address where the order is to be delivered is determined by the receiver.
A transitive dependency exists but no partial dependency exists.
Hence, *Order Details* is in *Second Normal Form (2NF).*

(v) ***Shipping Type***

As *Shipping Type* is a weak entity in our ER Model, (Typename, OrderID) acts as the primary key. Hence, $TO \rightarrow PN$.
No other functional dependencies exist.
Hence, *Shipping Type* is in *Third Normal Form (3NF)* as well as in *BCNF.*

(vi) ***Purchase History***

As *Purchase History* is a weak entity in our ER Model, (PurchaseID, UserID) acts as the primary key. Hence, $PU \rightarrow DQ$.
The FD $U \rightarrow D$ exists as the user solely determines the date when he/she placed his/her order.
A partial dependency exists.
Hence, *Purchase* is in *First Normal Form (1NF).*

(vii) ***Purchase_ Order***

We have (PurchaseID, UserID, OrderID) as the primary key and no non-key attributes.
Hence, *Purchase_ Order* is in *Third Normal Form (3NF).*

# Conclusion

From this project, we can get a thorough understanding of database design and how it can be applied to a real life scenario such as an e-Commerce Platform. We have used the theory developed on commercial databases to intuitively design this database. Although a practical e-Commerce Database will have many more dependencies and a much more complex design, it will be centered around our basic design.

Lastly, we have briefly created some relational schemas and normal forms for our database, which gives better understanding of implementation level design.