

**Product Dissection for Amazon**

### **Company Overview:**

Amazon is a multinational technology and e-commerce company founded by Jeff Bezos in 1994. It started as an online bookstore and has since evolved into one of the world's largest and most diverse online retailers, offering various products and services, including electronics, apparel, cloud computing, and streaming services. Amazon is also a major player in the development of artificial intelligence through products like Alexa, and its virtual assistant. The company's success is driven by its customer-centric approach, efficient logistics, and continuous innovation in various business sectors.

### **Product Dissection and Real-World Problems Solved by Amazon:**

The analysis or examination of a product sold on the Amazon platform. It involves breaking down and studying the various components, features, and aspects of a product to gain a deeper understanding of its design, functionality, quality, and overall value. This process is often undertaken by consumers, reviewers, or experts who aim to provide detailed insights into a product's strengths, weaknesses, and performance.

Such dissections can include reviews, unboxing videos, and in-depth examinations of the product's specifications, construction, and user experience. Analyzing customer feedback and ratings on Amazon is also a common aspect of product dissection, helping potential buyers make informed decisions about whether a particular product meets their needs and expectations. Overall, Amazon product dissection plays a crucial role in the online shopping experience, contributing to transparency and informed consumer decision-making.

### **Case Study: Real-World Problems and Amazon's Innovative Solutions**

Amazon offers a wide range of products and services, covering diverse industries. The primary online retail platform where customers can purchase a vast array of products, including electronics, books, apparel, and more.Services for grocery shopping and delivery, with Amazon Fresh offering fresh produce, and Whole Foods Market providing organic and high-quality food products. Amazon continues to expand and innovate across various industries. The company's diverse portfolio reflects its commitment to being a customer-centric and technology-driven organization.

**Problem 1: Unparalleled convenience in an online marketplace**

**Real-World Challenge:** Amazon offers unparalleled convenience by providing an extensive online marketplace where customers can find and purchase a wide range of products from the comfort of their homes. It eliminates the need to visit physical stores, saving time and effort.

**Amazon's Solution:**

Amazon solves several problems for both consumers and businesses. For consumers, Amazon provides a convenient and efficient way to shop for a wide range of products, often at competitive prices, and have them delivered directly to their doorstep. This saves time and effort compared to traditional retail shopping. Additionally, Amazon offers a vast selection of products, including items that may be hard to find in local stores.

**Problem 2: Delivery Network Expansion**

**Real-World Challenge:** Amazon has an amazing service. if you are based in a city then service time gets less. but if you are receiving it at a remote location then it might be a little inconvenience

**Amazon's Solution:** Amazon is known for providing good service to customers. While there may be variations in service time based on the location, Amazon has tried to expand its reach and improve delivery options, even in remote areas. Here are a few solutions that Amazon has implemented to address the inconvenience faced by customers in remote locations:

1. Delivery Network Expansion: Amazon has been expanding its delivery network to reach more remote areas. This includes establishing new fulfillment centers, and distribution hubs, and partnering with local carriers to improve delivery coverage.
2. Delivery Partnerships: Amazon has collaborated with logistics and transportation companies to enhance its delivery capabilities in remote locations. These partnerships help optimize routes and ensure efficient delivery to customers in harder-to-reach areas.
3. Amazon Hub: Amazon has installed Amazon Hub locations, such as Amazon Lockers, in select areas. These hubs serve as centralized pickup points for customers, enabling them to collect their orders at a convenient location, even in remote areas.

#### **Problem 3: Amazon Lockers**

**Real-World Challenge:** Amazon Lockers may not be available in all remote areas. The presence of lockers depends on the demand and feasibility of installing them in a particular location. As a result, customers in some remote areas may need access to this service.

**Amazon's Solution:**

The availability and accessibility of Amazon Locker services in remote areas can vary. Customers in remote areas should check the Amazon website or contact customer support to determine the availability of locker services and any specific limitations or restrictions that may apply to their location.

#### **Problem 4: Limited access to essential goods**

**Real-World Challenge:** Limited access refers to situations where individuals or communities face barriers or restrictions in obtaining essential goods, services, or opportunities.

**Amazon's Solution:** Limited access refers to situations where individuals or communities face barriers or restrictions in obtaining essential goods, services, or opportunities. Addressing limited access requires concerted efforts from governments, organizations, and communities. It involves initiatives such as improving infrastructure, expanding public services, promoting digital inclusion, implementing targeted policies, and reducing socioeconomic disparities to ensure equitable access to essential goods, services, and opportunities for all individuals, regardless of their geographical location or socioeconomic background.

**Conclusion:**

Amazon has achieved significant success as an e-commerce giant, it has also faced real-world problems that have garnered attention and criticism. Some of these problems include labor practices, concerns about market dominance, and antitrust issues.

Amazon has taken steps to address these challenges. For instance, the company has implemented measures to improve worker conditions and safety, made commitments to reduce its carbon footprint, and invested in technologies to combat counterfeit products. However, these problems highlight ongoing areas of concern and areas where Amazon continues to face scrutiny and pressure for improvement.

### **Top Features of Amazon:**

1. **Vast Product Selection:** Amazon provides an extensive selection of products across various categories, including electronics, books, clothing, home goods, and more. Customers can find a wide range of options and brands to choose from.
2. **One-Click Ordering:** Amazon's patented one-click ordering feature allows customers to make purchases quickly and easily with just a single click. This streamlined checkout process saves time and simplifies the purchasing experience.
3. **Prime Membership:** Amazon Prime is a subscription service that offers several benefits to members. It includes free two-day or even same-day shipping on eligible items, access to streaming services like Prime Video and Prime Music, exclusive deals and discounts, and more.
4. **Customer Reviews and Ratings:** Amazon provides customer reviews and ratings for most products, allowing shoppers to make informed purchasing decisions. These reviews provide valuable insights and feedback from other customers who have bought and used the product.
5. **Personalized Recommendations:** Amazon's recommendation engine analyzes customer behavior and purchase history to offer personalized product recommendations. This feature helps customers discover new products they might be interested in, enhancing the shopping experience.
6. **Subscribe & Save:** This feature allows customers to subscribe to regular deliveries of frequently used items at discounted prices. It is particularly useful for products like household essentials, pet supplies, and personal care items.

**Schema Description:**

The Amazon schema comprises various entities, each representing distinct facets of the platform. These entities include Customer, Product, Category, Order, Order\_Item, Payment, and Shipment, each characterized by specific attributes detailing its properties and relationships with other entities.

**Customer entity:** The "Customer" entity in the Amazon schema represents individuals who register accounts to make purchases on the online shopping platform. This entity encompasses attributes such as the customer's name, email address, postal address, and phone number.

* **Customer\_id(Primary Key):**  A unique identifier for each Customer.

(data\_type: int)

* **Name:** The Customer's name is displayed on their profile.

(data\_type: varchar(100))

* **Email:** The Customer's email address for account-related communication.

(data\_type: varchar(100))

* **Password:** Password of each customer for security.

(data\_type: varchar(50))

* **Address:** The address of the customer to deliver the order.

(data\_type: varchar(50))

* **Phone\_no:** A unique identifier for each customer.

(data\_type: int)

**Order entity**: Product orders placed by customers.

* **Order\_id** **(Primary Key)**: The unique order\_id of each order.

(data\_type: int)

* **Order\_date**: The date when the customer placed the order.

(data\_type: date/time)

* **price**: The price of the order placed by the customer.

(data\_type: float)

* **Customer\_id (Foreign Key referencing Customer entity):**  A uniqueCustomer\_idfor each order.

(data\_type: int)

* **Payment\_id(Foreign Key referencing Payment entity):** The unique Payment \_id of each order.

(data\_type: int)

* **Shipment\_id(Foreign Key referencing Shipment entity):** The unique Shipment \_id of each order.

(data\_type: int)

* **Total\_price:** The total price of all the orders

(data\_type: decimal (10,2)

**Order\_item entity**: Each item that is part of an order.

* **Order\_item\_id (Primary Key):** The unique order\_id of each order\_items.

(data\_type: int)

* **Product\_id** **(Foreign Key referencing product entity):** A unique Product\_id for each product.

(data\_type: int)

* **Quantity:** The quantity of the ordered\_items.

(data\_type: int)

* **Price:** The price of the ordered items placed by the customer.

(data\_type:float)

* **Order\_id** **(Primary Key, Foreign Key referencing Order entity):** The unique order\_id of each ordered\_item.

(data\_type: int)

**Product entity:** The "Product" entity within the Amazon schema signifies the collection of items accessible for buying on the platform. The product entity is defined with attributes to store Stock Keeping Unit (SKU), description, price, and stock quantity for each product.

* **Product\_id** **(Primary Key):**  A unique identifier for each product.

(data\_type:int)

* **SKU(stock keeping unit):** The unit of product stock.

(data\_type: varchar(100))

* **Description:** The product-related data.

(data\_type: varchar(100))

* **Price(Foreign Key referencing Order entity):** The price of the product

(data\_type: float)

* **Stock:** The product quantity represents the stocks.

(data\_type:int)

* **Customer\_id (Foreign Key referencing Customer entity):**  A uniqueCustomer\_idfor each product.

(data\_type: int)

**Category entity:** Categories in which the products are grouped. The Category entity is very simple; the only attribute it needs is the category name.

* **Category\_id** **(Primary Key):** The unique category\_id of each product.

(data\_type: int)

* **Name:** The name of the product displayed on the placed order.

(data\_type: varchar(50))

**Payment entity:** The payment made by the customer once the order is completed. The payment entity stores the date of payment, the means of payment, and the amount paid.

* **Payment\_id(primary key):** The unique Payment \_id of each ordered\_item.

(data\_type: int)

* **Payment\_date:** The date when the customer paid charges after placing an order.

(data\_type: date/time)

* **Payment\_methode:** The customer wants to select Payment\_methode for payment

(data\_type: varchar(50))

* **Amount:** The price of the ordered item.

(data\_type: float)

* **Customer\_id (Foreign Key referencing Customer entity):**  A uniqueCustomer\_idfor each payment entity.

(data\_type: int)

**Cart entity:** The "Cart" entity in the Amazon schema represents the customer's virtual basket or shopping cart, serving as a temporary storage for items before they are officially purchased and integrated into an order. Both "Cart" and "Wishlist" are dependent entities of the "Customer" entity. The "Cart" entity includes details about each product added to the shopping cart, encompassing information such as the product itself and its quantity. It's important to note that attributes constituting foreign keys or specific relationships are not included in this list.

* **Cart\_id**(**primary key):** The unique cart\_id of ordered\_item

(data\_type: int)

* **Quantity:** The quantity of the ordered\_items.

(data\_type: int)

* **Customer\_id (primary key, Foreign Key referencing Customer entity):**  A unique identifier for each cart product.

(data\_type: int)

* **Product\_id** **(Foreign Key referencing product entity):** A unique identifier for each cart product.

(data\_type: int)

**Wishlist entity:** Stores items chosen by the customer for possible future purchases. simply it contains a list of products.

* **Wishlist\_id(primary key):** The unique Wishlist \_id of liked\_items.

(data\_type: int)

* **Customer\_id (primary key, Foreign Key referencing Customer entity):**  A unique identifier for each wishlisting product.

(data\_type: int)

* **Product\_id** **(Foreign Key referencing product entity):** A unique identifier for each wishlist product.

(data\_type: int)

**Shipment entity**: Shipping information associated with an order, including delivery address and tracking information. Shipment stores the shipping date, address, city, state, country, and postal code.

* **Shipment\_id(primary key)**: The unique shipment\_id associated with an order.

(data\_type: int)

* **Shipment\_date:** The date associated with the order shipped.

(data\_type: date/time)

* **Address:** The address of the order to be delivered.

(data\_type: varchar(100))

* **State:** specified by which state.

(data\_type: varchar(50))

* **City:** specified by which city.

(data\_type: varchar(50))

* **Country:** specified by which country.

(data\_type: varchar(100))

* **Zip\_code:** The code on the shipped order.

(data\_type: int)

* **Customer\_id (Foreign Key referencing Customer entity):**  A unique identifier for each Shipment entity.

(data\_type: int)

In the Amazon schema, the following relationships exist:

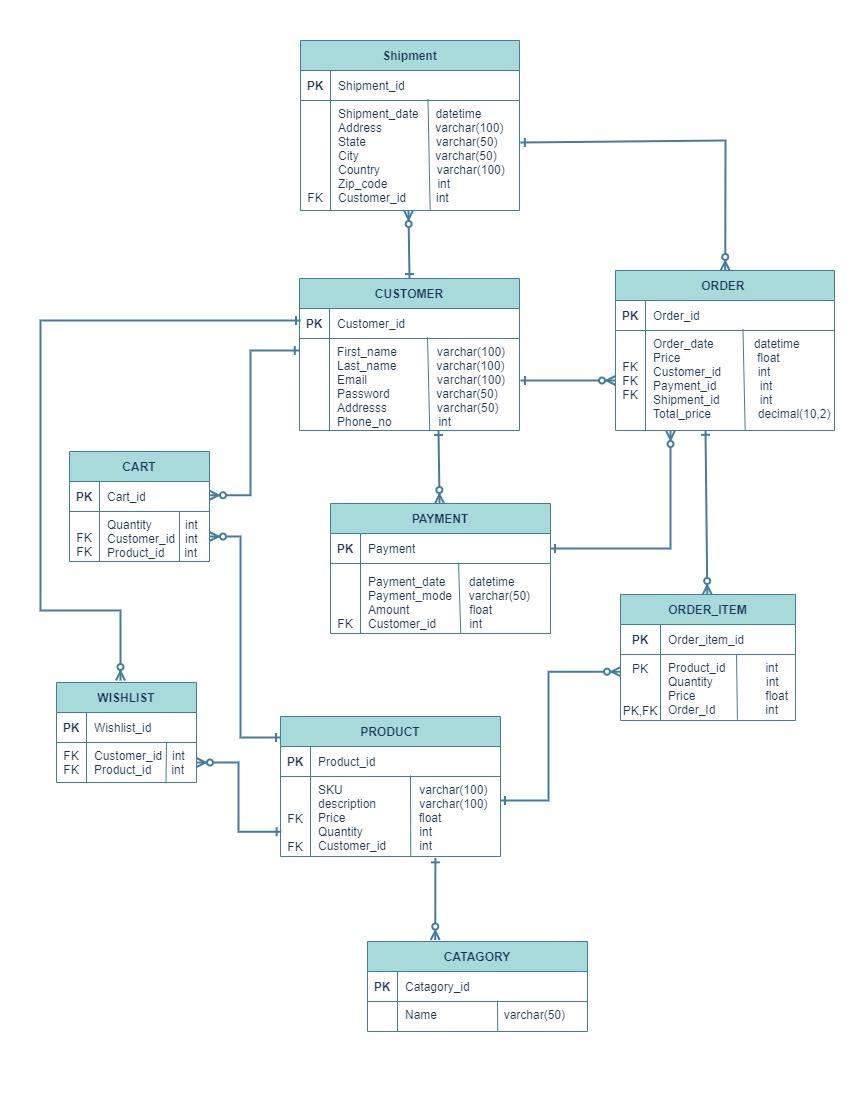
* Customer and Order:
  + One-to-many relationship: A customer can place several orders.
* Order and Order\_Item:
  + One-to-many relationship: An order can contain one or several items (Order\_Item), each representing a single product.
* Order\_Item and Product:
  + One-to-many relationship: Each Order\_Item is related to one Product, and a Product can be related to multiple Order\_Items.
* Order and Payment:
  + One-to-one relationship: An order is associated with one payment.
* Order and Shipment:
  + One-to-one relationship: An order is associated with one shipment.
* Payment and Order:
  + One-to-many relationship: Each payment can include multiple orders.
* Shipment and Order:
  + One-to-many relationship: Each shipment can include multiple orders.
* Product and Category:
  + One-to-many relationship: A product can belong to a single category.
* Customer and Cart/Wishlist:
  + Dependent entities: Both Cart and Wishlist maintain a dependency relationship with the Customer.
* Cart/Wishlist and Product:
* Many-to-one relationship: Each instance of Cart and Wishlist is related to a product.

These relationships establish the structural connections between the entities in the Amazonschema, reflecting the interactions and dependencies within the system.

**ER Diagram:**

In the context of an online shopping system, the ER diagram for Amazon's schema needs to be normalized up to the third normal form. This is crucial due to the transactional nature of the system, requiring support for constant and concurrent updates while ensuring data integrity and consistency. The database model should efficiently store customer information for those who register on the site and place orders. Additionally, it must manage product data, including price and stock details.

Customers on Amazon should have a shopping cart to store selected products before placing an order. Each customer should also be able to maintain a wish list, representing products of interest that haven't been added to a shopping cart yet. Upon order confirmation, the data model should record order details, payment information, and shipping information.



### **Conclusion**

In this case study, we delved into the design of the Amazon schema and Entity-Relationship diagram. We’ve seen the necessary steps to create a basic ER diagram for online shopping and convert it into a working database on an RDBMS. As we mentioned earlier, this model can accept many enhancements to provide more robust security or instantly updated price and stock information. By understanding this schema, we gain insight into how Amazon effectively manages the complexities of user interactions and content sharing, contributing to its widespread popularity and continued growth in the world.