

# Electronic Web Store Database Design

## Core Entities and Relationships

### 1. User Management

#### users

- user\_id (PK, INT, AUTO\_INCREMENT)
- username (VARCHAR(50), UNIQUE)
- email (VARCHAR(100), UNIQUE, NOT NULL)
- password\_hash (VARCHAR(255), NOT NULL)
- first\_name (VARCHAR(50))
- last\_name (VARCHAR(50))
- phone (VARCHAR(15))
- date\_registered (DATETIME, DEFAULT CURRENT\_TIMESTAMP)
- last\_login (DATETIME)
- is\_active (BOOLEAN, DEFAULT TRUE)
- is\_admin (BOOLEAN, DEFAULT FALSE)

#### user\_addresses

- address\_id (PK, INT, AUTO\_INCREMENT)
- user\_id (FK, INT, NOT NULL)
- address\_type (ENUM('billing', 'shipping'), NOT NULL)
- street\_address (VARCHAR(255), NOT NULL)
- city (VARCHAR(100), NOT NULL)
- state\_province (VARCHAR(100))
- postal\_code (VARCHAR(20))
- country (VARCHAR(100), NOT NULL)
- is\_default (BOOLEAN, DEFAULT FALSE)
- created\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP)

### 2. Product Catalog

#### categories

- category\_id (PK, INT, AUTO\_INCREMENT)
- category\_name (VARCHAR(100), NOT NULL)
- parent\_category\_id (FK, INT, NULL) -- for subcategories
- description (TEXT)
- image\_url (VARCHAR(255))
- is\_active (BOOLEAN, DEFAULT TRUE)
- sort\_order (INT, DEFAULT 0)

## **brands**

- brand\_id (PK, INT, AUTO\_INCREMENT)
- brand\_name (VARCHAR(100), NOT NULL, UNIQUE)
- description (TEXT)
- logo\_url (VARCHAR(255))
- website\_url (VARCHAR(255))
- is\_active (BOOLEAN, DEFAULT TRUE)

## **products**

- product\_id (PK, INT, AUTO\_INCREMENT)
- product\_name (VARCHAR(255), NOT NULL)
- sku (VARCHAR(50), UNIQUE, NOT NULL)
- brand\_id (FK, INT)
- category\_id (FK, INT, NOT NULL)
- description (TEXT)
- specifications (JSON) -- for technical specs
- price (DECIMAL(10,2), NOT NULL)
- compare\_price (DECIMAL(10,2)) -- original price for discounts
- cost\_price (DECIMAL(10,2)) -- for profit calculations
- weight (DECIMAL(8,2))
- dimensions (VARCHAR(50)) -- e.g., "10x5x3 inches"
- warranty\_period (INT) -- in months
- is\_active (BOOLEAN, DEFAULT TRUE)
- is\_featured (BOOLEAN, DEFAULT FALSE)
- created\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP)
- updated\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP)

### **product\_images**

- image\_id (PK, INT, AUTO\_INCREMENT)
- product\_id (FK, INT, NOT NULL)
- image\_url (VARCHAR(255), NOT NULL)
- alt\_text (VARCHAR(255))
- is\_primary (BOOLEAN, DEFAULT FALSE)
- sort\_order (INT, DEFAULT 0)

### **product\_variants**

- variant\_id (PK, INT, AUTO\_INCREMENT)
- product\_id (FK, INT, NOT NULL)
- variant\_name (VARCHAR(100)) -- e.g., "64GB Silver"
- sku (VARCHAR(50), UNIQUE, NOT NULL)
- price (DECIMAL(10,2), NOT NULL)
- stock\_quantity (INT, DEFAULT 0)
- variant\_attributes (JSON) -- color, size, storage, etc.
- is\_active (BOOLEAN, DEFAULT TRUE)

### 3. Inventory Management

#### inventory

- inventory\_id (PK, INT, AUTO\_INCREMENT)
- product\_id (FK, INT)
- variant\_id (FK, INT)
- quantity\_available (INT, DEFAULT 0)
- quantity\_reserved (INT, DEFAULT 0) -- for pending orders
- reorder\_level (INT, DEFAULT 0)
- last\_updated (DATETIME, DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP)

#### inventory\_movements

- movement\_id (PK, INT, AUTO\_INCREMENT)
- product\_id (FK, INT)
- variant\_id (FK, INT)
- movement\_type (ENUM('in', 'out', 'adjustment'), NOT NULL)
- quantity (INT, NOT NULL)
- reference\_type (ENUM('purchase', 'sale', 'return', 'adjustment'))
- reference\_id (INT) -- order\_id, return\_id, etc.
- notes (TEXT)
- created\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP)
- created\_by (FK, INT) -- user\_id

### 4. Shopping Cart

#### shopping\_cart

- cart\_id (PK, INT, AUTO\_INCREMENT)
- user\_id (FK, INT)
- session\_id (VARCHAR(255)) -- for guest users
- created\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP)
- updated\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP)

### **cart\_items**

- cart\_item\_id (PK, INT, AUTO\_INCREMENT)
- cart\_id (FK, INT, NOT NULL)
- product\_id (FK, INT, NOT NULL)
- variant\_id (FK, INT)
- quantity (INT, NOT NULL, DEFAULT 1)
- unit\_price (DECIMAL(10,2), NOT NULL)
- added\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP)

## **5. Order Management**

### **orders**

- order\_id (PK, INT, AUTO\_INCREMENT)
- order\_number (VARCHAR(50), UNIQUE, NOT NULL)
- user\_id (FK, INT)
- order\_status (ENUM('pending', 'confirmed', 'processing', 'shipped', 'delivered', 'cancelled', 'refunded'), DEFAULT 'pending')
- payment\_status (ENUM('pending', 'authorized', 'captured', 'failed', 'refunded'), DEFAULT 'pending')
- subtotal (DECIMAL(10,2), NOT NULL)
- tax\_amount (DECIMAL(10,2), DEFAULT 0)
- shipping\_amount (DECIMAL(10,2), DEFAULT 0)
- discount\_amount (DECIMAL(10,2), DEFAULT 0)
- total\_amount (DECIMAL(10,2), NOT NULL)
- currency (VARCHAR(3), DEFAULT 'USD')
- billing\_address (JSON) -- store address as JSON
- shipping\_address (JSON)
- notes (TEXT)
- created\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP)
- updated\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP)

### **order\_items**

- order\_item\_id (PK, INT, AUTO\_INCREMENT)
- order\_id (FK, INT, NOT NULL)
- product\_id (FK, INT, NOT NULL)
- variant\_id (FK, INT)
- product\_name (VARCHAR(255), NOT NULL) -- snapshot at time of order
- sku (VARCHAR(50), NOT NULL)
- quantity (INT, NOT NULL)
- unit\_price (DECIMAL(10,2), NOT NULL)
- total\_price (DECIMAL(10,2), NOT NULL)

## **6. Payment Processing**

### **payments**

- payment\_id (PK, INT, AUTO\_INCREMENT)
- order\_id (FK, INT, NOT NULL)
- payment\_method (ENUM('credit\_card', 'debit\_card', 'paypal', 'stripe', 'bank\_transfer'), NOT NULL)
- payment\_status (ENUM('pending', 'completed', 'failed', 'refunded'), DEFAULT 'pending')
- transaction\_id (VARCHAR(255)) -- from payment gateway
- amount (DECIMAL(10,2), NOT NULL)
- currency (VARCHAR(3), DEFAULT 'USD')
- gateway\_response (JSON) -- store gateway response
- processed\_at (DATETIME)
- created\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP)

## 7. Shipping and Fulfillment

### shipping\_methods

- shipping\_method\_id (PK, INT, AUTO\_INCREMENT)
- method\_name (VARCHAR(100), NOT NULL)
- carrier (VARCHAR(100))
- estimated\_days (INT)
- base\_cost (DECIMAL(10,2), NOT NULL)
- cost\_per\_kg (DECIMAL(10,2), DEFAULT 0)
- free\_shipping\_threshold (DECIMAL(10,2))
- is\_active (BOOLEAN, DEFAULT TRUE)

### shipments

- shipment\_id (PK, INT, AUTO\_INCREMENT)
- order\_id (FK, INT, NOT NULL)
- shipping\_method\_id (FK, INT, NOT NULL)
- tracking\_number (VARCHAR(255))
- carrier (VARCHAR(100))
- status (ENUM('pending', 'shipped', 'in\_transit', 'delivered', 'returned'), DEFAULT 'pending')
- shipped\_at (DATETIME)
- delivered\_at (DATETIME)
- shipping\_address (JSON)
- notes (TEXT)
- created\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP)

## 8. Reviews and Ratings

### product\_reviews

- review\_id (PK, INT, AUTO\_INCREMENT)
- product\_id (FK, INT, NOT NULL)
- user\_id (FK, INT, NOT NULL)
- order\_id (FK, INT) -- to verify purchase
- rating (INT, NOT NULL, CHECK(rating >= 1 AND rating <= 5))
- title (VARCHAR(255))
- review\_text (TEXT)
- is\_verified\_purchase (BOOLEAN, DEFAULT FALSE)
- is\_approved (BOOLEAN, DEFAULT TRUE)
- helpful\_count (INT, DEFAULT 0)
- created\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP)
- updated\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP)

## 9. Discounts and Promotions

### coupons



- coupon\_id (PK, INT, AUTO\_INCREMENT)
- coupon\_code (VARCHAR(50), UNIQUE, NOT NULL)
- discount\_type (ENUM('percentage', 'fixed\_amount', 'free\_shipping'), NOT NULL)
- discount\_value (DECIMAL(10,2), NOT NULL)
- minimum\_order\_amount (DECIMAL(10,2), DEFAULT 0)
- usage\_limit (INT) -- total usage limit
- usage\_limit\_per\_user (INT, DEFAULT 1)
- usage\_count (INT, DEFAULT 0)
- valid\_from (DATETIME, NOT NULL)
- valid\_until (DATETIME, NOT NULL)
- is\_active (BOOLEAN, DEFAULT TRUE)
- created\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP)

#### **coupon\_usage**

- usage\_id (PK, INT, AUTO\_INCREMENT)
- coupon\_id (FK, INT, NOT NULL)
- user\_id (FK, INT, NOT NULL)
- order\_id (FK, INT, NOT NULL)
- discount\_amount (DECIMAL(10,2), NOT NULL)
- used\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP)

## **10. Wishlist**

#### **wishlists**

- wishlist\_id (PK, INT, AUTO\_INCREMENT)
- user\_id (FK, INT, NOT NULL)
- product\_id (FK, INT, NOT NULL)
- variant\_id (FK, INT)
- added\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP)

## **11. Analytics and Reporting**

#### **page\_views**

- view\_id (PK, INT, AUTO\_INCREMENT)
- product\_id (FK, INT)
- user\_id (FK, INT)
- session\_id (VARCHAR(255))
- ip\_address (VARCHAR(45))
- user\_agent (TEXT)
- viewed\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP)

### **search\_queries**

- query\_id (PK, INT, AUTO\_INCREMENT)
- search\_term (VARCHAR(255), NOT NULL)
- user\_id (FK, INT)
- session\_id (VARCHAR(255))
- results\_count (INT, DEFAULT 0)
- clicked\_product\_id (FK, INT)
- searched\_at (DATETIME, DEFAULT CURRENT\_TIMESTAMP)

## **Key Indexes for Performance**

sql

*-- User indexes*

```
CREATE INDEX idx_users_email ON users(email);  
CREATE INDEX idx_users_username ON users(username);
```

*-- Product indexes*

```
CREATE INDEX idx_products_category ON products(category_id);  
CREATE INDEX idx_products_brand ON products(brand_id);  
CREATE INDEX idx_products_sku ON products(sku);  
CREATE INDEX idx_products_active ON products(is_active);
```

*-- Order indexes*

```
CREATE INDEX idx_orders_user ON orders(user_id);  
CREATE INDEX idx_orders_status ON orders(order_status);  
CREATE INDEX idx_orders_date ON orders(created_at);  
CREATE INDEX idx_order_items_order ON order_items(order_id);  
CREATE INDEX idx_order_items_product ON order_items(product_id);
```

*-- Inventory indexes*

```
CREATE INDEX idx_inventory_product ON inventory(product_id);  
CREATE INDEX idx_inventory_variant ON inventory(variant_id);
```

*-- Review indexes*

```
CREATE INDEX idx_reviews_product ON product_reviews(product_id);  
CREATE INDEX idx_reviews_user ON product_reviews(user_id);  
CREATE INDEX idx_reviews_approved ON product_reviews(is_approved);
```

*-- Analytics indexes*

```
CREATE INDEX idx_page_views_product ON page_views(product_id);  
CREATE INDEX idx_page_views_date ON page_views(viewed_at);  
CREATE INDEX idx_search_queries_term ON search_queries(search_term);
```

## Business Rules and Constraints

### Data Integrity Rules

1. **Stock Management:** Ensure inventory quantities cannot go negative
2. **Price Validation:** All prices must be positive values
3. **Order Totals:** Order totals must equal sum of items + tax + shipping - discounts
4. **Unique Constraints:** SKUs must be unique across products and variants
5. **Email Validation:** User emails must be unique and valid format

### Security Considerations

1. **Password Storage:** Always hash passwords using bcrypt or similar
2. **PII Protection:** Encrypt sensitive data like credit card information
3. **Session Management:** Implement secure session handling for shopping carts
4. **Input Validation:** Validate all user inputs to prevent SQL injection
5. **Access Control:** Implement role-based permissions for admin functions

## Performance Optimizations

1. **Caching:** Implement Redis/Memcached for frequently accessed product data
2. **Database Partitioning:** Consider partitioning large tables like orders by date
3. **Read Replicas:** Use read replicas for analytics and reporting queries
4. **CDN:** Store product images and static assets on CDN
5. **Database Connection Pooling:** Use connection pooling for better performance

This database design provides a solid foundation for an electronic web store with room for scalability and additional features as the business grows.