Database Design Document

Version: 1.0

Project: ContentKrate Digital Marketplace

# 1. Overview

This document outlines the database schema for ContentKrate, a digital products marketplace. The database supports user accounts, product listings, orders, and content management.

# 2. Database Schema

## 2.1 Tables Structure

users

Stores user account information.

sql

CREATE TABLE users (

id INT PRIMARY KEY AUTO\_INCREMENT,

username VARCHAR(50) UNIQUE NOT NULL,

email VARCHAR(100) UNIQUE NOT NULL,

password\_hash VARCHAR(255) NOT NULL,

user\_type ENUM('admin', 'editor', 'customer') DEFAULT 'customer',

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

last\_login TIMESTAMP NULL

);

products

Contains product listings.

sql

CREATE TABLE products (

id INT PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(200) NOT NULL,

category ENUM('cameras', 'audio', 'lighting', 'software', 'accessories'),

description TEXT,

base\_price DECIMAL(10,2),

amazon\_ca\_url VARCHAR(500),

image\_url VARCHAR(300),

specifications JSON,

is\_featured BOOLEAN DEFAULT FALSE,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

product\_options

Stores variant options for products.

sql

CREATE TABLE product\_options (

id INT PRIMARY KEY AUTO\_INCREMENT,

product\_id INT,

option\_name VARCHAR(100),

option\_description TEXT,

price\_modifier DECIMAL(10,2),

amazon\_url VARCHAR(500),

FOREIGN KEY (product\_id) REFERENCES products(id)

);

wishlists

Tracks user wishlists.

sql

CREATE TABLE wishlists (

id INT PRIMARY KEY AUTO\_INCREMENT,

user\_id INT,

product\_id INT,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (user\_id) REFERENCES users(id),

FOREIGN KEY (product\_id) REFERENCES products(id)

);

**orders**

Manages customer orders.

sql

CREATE TABLE orders (

id INT PRIMARY KEY AUTO\_INCREMENT,

user\_id INT NOT NULL,

order\_number VARCHAR(20) UNIQUE,

total\_amount DECIMAL(10,2) NOT NULL,

status ENUM('pending', 'processing', 'shipped', 'delivered', 'cancelled') DEFAULT 'pending',

payment\_method VARCHAR(50),

shipping\_address TEXT,

billing\_address TEXT,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

FOREIGN KEY (user\_id) REFERENCES users(id)

);

order\_items

Contains line items for orders.

sql

CREATE TABLE order\_items (

id INT PRIMARY KEY AUTO\_INCREMENT,

order\_id INT NOT NULL,

product\_id INT NOT NULL,

option\_id INT NULL,

quantity INT NOT NULL,

unit\_price DECIMAL(10,2) NOT NULL,

total\_price DECIMAL(10,2) NOT NULL,

FOREIGN KEY (order\_id) REFERENCES orders(id),

FOREIGN KEY (product\_id) REFERENCES products(id),

FOREIGN KEY (option\_id) REFERENCES product\_options(id)

);

## 2.2 Relationships

Relationship Description

users.id → wishlists.user\_id One-to-many (User can have multiple wishlist items)

products.id → product\_options.product\_id One-to-many (Product can have multiple options)

users.id → orders.user\_id One-to-many (User can place multiple orders)

# 3. Indexes

Automatically created for:

All PRIMARY KEY fields

All FOREIGN KEY fields

users.username, users.email (UNIQUE constraints)

# 4. Sample Queries

Get Featured Products

sql

SELECT \* FROM products

WHERE is\_featured = TRUE

ORDER BY created\_at DESC

LIMIT 5;

Get User Wishlist

sql

SELECT p.\* FROM wishlists w

JOIN products p ON w.product\_id = p.id

WHERE w.user\_id = [USER\_ID];