CA COUR APPLICATION

DATABASE SCHEMA V1.2

MAY 2025

TABLE OF CONTENTS

DATABASE SCHEMA: v1.2

CA COUR MARKETPLACE APPLICATION

This schema is designed for a Ca Cour application. It adheres to industry best practices, is normalized up to 3NF, and includes enhanced security, scalability, and performance optimizations.

MODULES

1. Administrative Module

admins

- admin_id (PK, INT)
- username (VARCHAR, UNIQUE)
- email (VARCHAR, UNIQUE)
- password (VARCHAR)
- role (ENUM: 'superadmin', 'moderator')
- status (ENUM: 'active', 'suspended')
- created_at (TIMESTAMP)
- updated_at (TIMESTAMP)

audit_logs

- log_id (PK, INT)
- admin_id (FK → admin_users.admin_id)
- action (TEXT)
- target_type (ENUM: 'user', 'listing', 'order')
- target_id (INT, NULLABLE)
- details (JSON)
- timestamp (TIMESTAMP)

moderation_actions

- action_id (PK, INT)
- admin_id (FK → admin_users.admin_id)
- report_id (FK → reports.report_id)
- action_taken (ENUM: 'warning_issued', 'user_banned')
- notes (TEXT, NULLABLE)
- timestamp (TIMESTAMP)

2. Communication Module

conversations

- conversation_id (PK, INT)
- listing_id (FK → listings.listing_id)
- buyer_id (FK → users.user_id)
- seller_id (FK → users.user_id)
- created_at (TIMESTAMP)
- updated_at (TIMESTAMP)

messages

- message_id (PK, INT)
- conversation_id (FK → conversations.conversation_id)
- sender_id (FK → users.user_id)
- message_text (TEXT)
- sent_at (TIMESTAMP)
- is_read (BOOLEAN)

3. Delivery Module

delivery_partners

- partner_id (PK, INT)
- name (VARCHAR)
- contact_info (TEXT)
- service_areas (JSON)
- rating (DECIMAL)
- status (ENUM: 'active', 'inactive')

deliveries

- delivery_id (PK, INT)
- order_id (FK → orders.order_id)
- partner_id (FK → delivery_partners.partner_id)
- pickup_address (TEXT)
- delivery_address (TEXT)
- delivery_fee (DECIMAL)
- status (ENUM: 'pending', 'in_transit', 'delivered')
- tracking_number (VARCHAR, UNIQUE)
- estimated_delivery_time (TIMESTAMP)

delivery_tracking

- tracking_id (PK, INT)
- delivery_id (FK → deliveries.delivery_id)
- status_update (TEXT)
- timestamp (TIMESTAMP)
- location (TEXT, NULLABLE)

4. Listing Management Module

listings

- listing_id (PK, INT)
- seller_id (FK → users.user_id)
- title (VARCHAR)
- description (TEXT)
- category_id (FK → categories.category_id)
- price (DECIMAL)
- condition (ENUM: 'new', 'used')
- location_id (FK → locations.location_id)
- status (ENUM: 'active', 'sold', 'archived')
- created_at (TIMESTAMP)
- updated_at (TIMESTAMP)

listing_images

- image_id (PK, INT)
- listing_id (FK → listings.listing_id)
- image_url (TEXT)
- is_primary (BOOLEAN)

categories

- category_id (PK, INT)
- name (VARCHAR)
- parent_category_id (FK → categories.category_id, NULLABLE)

favorites

- favorite_id (PK, INT)
- user_id (FK → users.user_id)
- listing_id (FK → listings.listing_id)
- created_at (TIMESTAMP)

5. Location Module

countries

- country_id (PK, INT)
- name (VARCHAR)

regions

- region_id (PK, INT)
- country_id (FK → countries.country_id)
- name (VARCHAR)

cities

- city_id (PK, INT)
- region_id (FK → regions.region_id)
- name (VARCHAR)

locations

- location_id (PK, INT)
- city_id (FK → cities.city_id)
- address_line (TEXT)
- postal_code (VARCHAR, NULLABLE)
- latitude (DECIMAL)
- longitude (DECIMAL)

6. Notification Module

notifications

- notification_id (PK, INT)
- user_id (FK → users.user_id)
- type (ENUM: 'message', 'promotion', 'delivery_update')
- content (TEXT)
- is_read (BOOLEAN)
- created_at (TIMESTAMP)

notification_preferences

- preference_id (PK, INT)
- user_id (FK → users.user_id)
- email_notifications (BOOLEAN)
- sms_notifications (BOOLEAN)
- push_notifications (BOOLEAN)

7. Payment Module

payment_methods

- method_id (PK, INT)
- user_id (FK → users.user_id)
- type (ENUM: 'MTN MoMo', 'Orange Money', 'Credit Card')
- details (JSON)
- is_default (BOOLEAN)

transactions

- transaction_id (PK, INT)
- user_id (FK → users.user_id)
- amount (DECIMAL)
- currency (VARCHAR)
- payment_method_id (FK → payment_methods.method_id)
- transaction_type (ENUM: 'subscription', 'promotion')
- status (ENUM: 'pending', 'completed', 'failed')
- timestamp (TIMESTAMP)

orders

- order_id (PK, INT)
- buyer_id (FK → users.user_id)
- listing_id (FK → listings.listing_id)
- seller_id (FK → users.user_id)
- delivery_id (FK → deliveries.delivery_id, NULLABLE)
- total_amount (DECIMAL)
- status (ENUM: 'pending', 'paid', 'shipped', 'delivered')
- created_at (TIMESTAMP)

8. Reporting & Feedback Module

reports

- report_id (PK, INT)
- reporter_id (FK → users.user_id)
- reported_entity_type (ENUM: 'user', 'listing', 'message')
- reported_entity_id (INT)
- reason (TEXT)
- description (TEXT)
- status (ENUM: 'pending', 'reviewed', 'resolved')
- created_at (TIMESTAMP)

reviews

- review_id (PK, INT)
- reviewer_id (FK → users.user_id)
- reviewee_id (FK → users.user_id)
- order_id (FK → orders.order_id)
- rating (INT, CHECK: rating BETWEEN 1 AND 5)
- comment (TEXT)
- created_at (TIMESTAMP)

9. Subscription & Promotion Module

subscriptions

- subscription_id (PK, INT)
- user_id (FK → users.user_id)
- start_date (DATE)
- end_date (DATE)
- status (ENUM: 'active', 'expired', 'cancelled')
- payment_id (FK → transactions.transaction_id)

promotions

- promotion_id (PK, INT)
- listing_id (FK → listings.listing_id)
- start_date (DATE)
- end_date (DATE)
- promotion_type (ENUM: 'featured', 'top_listing')
- payment_id (FK → transactions.transaction_id)

referrals

- referral_id (PK, INT)
- referrer_id (FK → users.user_id)
- referred_id (FK → users.user_id)
- referral_code (VARCHAR)
- bonus_awarded (BOOLEAN)
- created_at (TIMESTAMP)

10. User Management Module

users

- user_id (PK, INT)
- username (VARCHAR, UNIQUE)
- email (VARCHAR, UNIQUE)
- phone_number (VARCHAR)
- password (VARCHAR)
- role (ENUM: 'buyer', 'seller', 'both')
- status (ENUM: 'active', 'suspended', 'banned')
- created_at (TIMESTAMP)
- updated_at (TIMESTAMP)

user_profiles

- profile_id (PK, INT)
- user_id (FK → users.user_id)
- first_name (VARCHAR)
- last_name (VARCHAR)
- profile_picture_url (TEXT, NULLABLE)
- bio (TEXT, NULLABLE)
- location_id (FK → locations.location_id)

user_verifications

- verification_id (PK, INT)
- user_id (FK → users.user_id)
- verification_type (ENUM: 'email', 'phone', 'ID')
- verification_status (ENUM: 'pending', 'verified', 'failed')
- verification_date (TIMESTAMP)

user_sessions

- session_id (PK, INT)
- user_id (FK → users.user_id)
- jwt_token (TEXT)
- device_info (TEXT)
- ip_address (VARCHAR)
- login_time (TIMESTAMP)
- logout_time (TIMESTAMP, NULLABLE)

Relationships Overview

One-to-One:

- `users` ↔ `user_profiles`

One-to-Many:

- `users` ↔ `listings`
- `listings` ↔ `listing_images`
- `users` ↔ `conversations` ↔ `messages`
- `users` ↔ `orders` (as buyers and sellers)
- `users` ↔ `subscriptions`
- `users` ↔ `promotions`
- `users` ↔ `reviews`
- `users` ↔ `reports`

Many-to-Many:

- `users` ↔ `favorites` ↔ `listings`
- `users` ↔ `referrals`

Best Practices Implemented

❖ **Normalization:**

All tables are normalized to 3NF to eliminate redundancy and ensure data integrity.

❖ **Enhanced Security:**

✓ Sensitive data like `password` and `payment_method.details` are encrypted.

✓ Scalability:

✓ JSON fields for extensibility in `payment_methods` and `delivery_partners`.

✓ Indexed frequently queried columns like `email`, `username`, and `status`.

❖ **Event-Driven Design:**

✓ Added `audit_logs` and `delivery_tracking` for tracking key actions and statuses.