CS316-Final project-MS2

Zhixuan Lu, Shu Pu, Shanruo Xu, Weisheng Zhang, Jiaojiao Zhao

All Tables, Constraints, and Assumptions

Table

1. Users Guru:

Users(**user_id**, email, password, full_name, address, balance)

Transactions(**transaction_id**, user_id, transaction_type, amount, transaction_date, description)

Purchase(purchase_id, user_id, total_amount, purchase_date, status)

PurchaseItem(**purchase_item_id**, product_id, item_id, seller_id, quantity, product_price, total_price)

item(**product_id**, category_id, name, description, image, price)

category(**category_id**, category_name)

Reviews(**review_id**, user_id, product_id, seller_id, rating, review_text, created_time, updated_time)

2. Product Guru:

Product (**product_id**, category_id, name, description, image, price)

Categories Table (**category_id**, category_name)

Sellers (**seller_id**, seller_name, contact_info)

Stock_Table (**stock_id**, product_id, seller_id, quantity_in_stock, price)

review (**review_id**, reviewer_id, seller_id, rating, comment, review_date)

Users(**user_id**, fullname, password, email, balance, address)

Product tags table (**product_id,** tag_id)

3. Cart’s Guru:

Users(**user_id**, fullname, password, email, balance, address)

Product(**product_id,** category_id, name, description, image, price)

Cart(**cart_id,** **user_id**, created_time, updated_time)

CartItem(**cart_item_id**, cart_id, product_id, quantity_in_cart, saved_for_later, added_at, updated_at)

Coupon(**coupon_id**, code, description, discount_type, discount_value, applies_to, product_id, category_id, maximum_uses, used_so_far, valid_from, valid_to, active, created_at, updated_at)

Stock_Table (**stock_id,** product_id, seller_id, quantity_in_stock, price)

Order_table(**order_id**, user_id, total_amount, order_status, created_time, updated_time)

Order_items(**order_item_id**, order_id, product_id, seller_id, quantity, price_at_order, fulfillment_status, fulfilled_time)

Transactions(**transaction_id**, user_id, transaction_type, amount, transaction_date, description)

StockTransactions(**transaction_id**, product_id, quantity_change, transaction_type, order_id, description, created_at)

4.Sellers Guru: responsible for Inventory / Order Fulfillment

Users(**user_id**, email, password, full_name, address, balance)

Cart(**cart_id**, user_id, created_time, updated_time)

Cart_items(**cart_item_id**, cart_id, product_id, seller_id, quantity, price_at_added, added_time)

Order_table(**order_id**, user_id, total_amount, order_status, created_time, updated_time)

Order_items(**order_item_id**, order_id, product_id, seller_id, quantity, price_at_order, fulfillment_status, fulfilled_time)

Transactions(**transaction_id**, user_id, transaction_type, amount, transaction_date, description)

5.Social Guru:

Reviews(**review_id**, user_id, product_id, seller_id, rating, review_text, created_time, updated_time)

Messages(**message_id**, sender_id, receiver_id, order_id, content, created_ time)

ReviewLikes(**like_id**, review_id, user_id, created_ time)

ReviewImages(**image_id**, review_id, image_url, uploaded_ time)

Reports(**report_id**, reporter_id, review_id, message_id, reason, created_ time, status)

Constraints and Assumptions of the tables:

Users:

Description: This table stores user account details, including balance and personal information.

Constraints:

email must be unique across all users.

balance >= 0: A user cannot have a negative balance.

created_time and updated_time should be automatically set and updated on account creation and modification, respectively.

Assumptions:

The password will be securely hashed, and security practices will be applied to manage user authentication.

Product:

Description:

The Products table stores basic information about the products available for sale. It includes references to the product category, price, and other details.

Constraints:

Each product is uniquely identified by product_id, which serves as the primary key. It also has a foreign key, category_id, which references the Categories table to ensure that all products are associated with a valid category. Product names must be unique to prevent duplication, and fields such as name, category_id, and price are mandatory.

Assumptions:

Products can only belong to one category.

Each product has a unique name within its category.

Not every product requires an image.

Default price is set when the product is created but may vary depending on the seller

Categories:

Description:

The Categories table holds information about the different categories that products can belong to. This helps organize products into various sections.

Constraints:

Category names are constrained to be unique, ensuring that no two categories can have the same name.

Assumptions:

Category names are unique across the entire system.

Categories cannot be deleted if they still have products assigned to them.

Sellers：

Description:

The Sellers table stores information about the sellers on the platform. Each seller has a unique identifier, name, and optional contact information.

Constraints:

Seller names must be unique across the platform to avoid confusion. While contact information is optional, it should be valid if provided.

Assumptions:

It is assumed that sellers operate independently, and each has a unique identity in the system.

Product tags:

Description:

The Product Tags table establishes a many-to-many relationship between products and tags, allowing multiple tags for each product.

Constraints:

product_id: Foreign key, non-null, must reference the Products table.

tag_id: Foreign key, non-null, must reference the Tags table.

Assumptions:

Each product can have multiple tags, and a tag can apply to multiple products.

This table does not allow duplicate product-tag pairs.

Cart:

Description:

Stores information about a user's active shopping cart.

Constraints:

user_id is a foreign key referencing the users table.

A user can have only one active cart at a time.

created_time and updated_time should be automatically set on cart creation and updated when items are added/removed.

Assumptions:

The cart persists across user sessions until the user checks out or empties the cart.

Each user has exactly one active cart. The cart persists across user sessions until the order is placed.

Cart_item:

Description:

Tracks products added to a user's cart, including quantities and prices at the time they were added.

Constraints:

cart_id is a foreign key referencing the cart table.

product_id is a foreign key referencing the products table.

seller_id is a foreign key referencing the sellers table.

quantity > 0: The quantity must be positive.

The added_time timestamp should automatically set when an item is added to the cart.

Assumptions:

Prices are stored as price_at_added to reflect changes in price after the item is added to the cart but before checkout.

Transactions:

Constraints:

transaction_id: Primary Key, must be unique.

user_id: Foreign Key referencing Users.user_id.

amount: Must be positive or negative depending on the transaction_type.

transaction_type: Specifies whether the transaction is a purchase, refund, etc.

Assumptions:

Transaction history is maintained for auditing user purchases, refunds, and other financial activities.

Description provides a brief summary of the transaction purpose.

StockTransaction:

Constraints:

transaction_id: Primary Key, must be unique.

product_id: Foreign Key referencing Product.product_id.

quantity_change: Reflects how stock is adjusted (negative for sale, positive for restocking).

transaction_type: Specifies if the adjustment is due to a sale, restock, or other action.

created_at: Automatically generated timestamp for when the transaction occurred.

Assumptions:

Stock changes are recorded in this table for each product transaction.

Adjustments to stock are triggered by orders or manual updates by sellers.

Orders:

Description:

Contains completed order information for each user.

Constraints:

user_id is a foreign key referencing the users table.

order_status is an enum with values such as pending, completed, and canceled.

total_amount > 0: Orders must have a positive total value.

created_time and updated_time should be automatically set on order creation and modification.

Assumptions:

The total amount is calculated when the order is placed, ensuring the inventory and prices are locked in for the transaction.

Order items:

Description:

Represents each item within an order, allowing sellers to track their part of the order fulfillment process.

Constraints:

order_id is a foreign key referencing the orders table.

Reviews:

Description:

Stores user reviews for products, including ratings and textual feedback.

Constraints:

created_time and updated_time should be automatically set on review creation and update.

rating should be within an acceptable range (e.g., 1-5).

Assumptions:

Reviews are linked directly to a product and seller to ensure feedback is relevant and traceable.

Messages:

Description:

Manages direct messages between users related to their transactions.

Constraints:

created_time should be automatically set when the message is created.

message_id must be unique to ensure each message is individually identifiable.

Assumptions:

Messages can be related to orders, providing a direct communication channel between buyers and sellers or between users.

ReviewLikes:

Description:

Tracks likes given to product reviews by users.

Constraints:

created_time should be automatically set when the like is recorded.

Each combination of review_id and user_id should be unique to prevent multiple likes from the same user on the same review.

Assumptions:

A user can like a review to show agreement or appreciation, but can only like each review once.

ReviewImages:

Description:

Allows users to upload images related to their product reviews.

Constraints:

uploaded_time should be automatically set when the image is uploaded.

image_url must be valid and accessible.

Assumptions:

Images provide visual feedback or evidence related to the review content, enhancing the review's credibility and information value.

Reports:

Description:

Enables users to report inappropriate messages, reviews, or images.

Constraints:

created_time should be automatically set when the report is created.

Each report must specify a reason and can be linked to either a message or a review.

Assumptions:

Reports are reviewed by administrators to maintain the quality and safety of the platform content.

Preliminary Design of Website

1. Homepage:

   Contents: Search bar for products, Featured products and categories, Quick links: Sign In, Register, View Cart

   User flow: Users can browse featured products, search for specific items, or navigate to other sections (login, register, cart).

2. Product Listing

   Contents: A list of products with filters, Each product shows a thumbnail, name, price, average rating, and a link to detailed product view. Description of a product with images, price, reviews & an 'Add to Cart' button.

   User Flow: Users can browse products by categories, sort products, Add To Cart, Read Reviews

3. Product Details

   Contents: Description of a product with images, price, reviews & an 'Add to

Cart' button. Sellers offering the product with their available quantity. Reviews section

User Flow: Users can view product details and choose a seller to purchase from. They can add the item to the cart and view reviews or submit their own after purchase.

4. User Registration/Login

   Contents: Email, password, name, address input fields, Login/Sign Up button

   User Flow: New users can sign up; existing users can log in

5. Shopping Cart

   Contents: Cart items, Pre checkout page, &totals etc.

   User Flow: Checkout or Remove items, Continue.

6. Checkout

   Contents: Review of cart items, Shipping details, Payment summary, Place order button

   User Flow: Users can confirm their shipping details and complete the purchase, Upon successful order placement, users are taken to an order confirmation page.

7. Order Confirmation

   Contents: Summary of order, estimated delivery time

   User Flow: Users can see their purchase summary and track fulfillment.

8. User Profile

   Contents: User account details, Purchase history with filters, Button to top-up balance or withdraw funds.

User Flow: Users can view and edit their profile details, can browse their order history or manage their balance.

9. Seller Page

   Contents: A list of products the seller has added to inventory. Buttons to update quantities, edit product details, or remove a product from inventory. Summary of fulfilled orders.

   User Flow: Sellers can manage their inventory and view order fulfillment history

10. Reviews

    Contents: Lists of product reviews and seller reviews submitted by the user.

    User Flow: Users can browse and manage their reviews, as well as rate other reviews as helpful.

11. Messages

    Contents: Message threads with sellers

    User Flow: Users can communicate with sellers about their orders.