

Database Project – Shopee

github repo link - https://github.com/Xavier-Naing/Shopee_project_Database.git

Project Scope

This project involves the reconstruction of an e-commerce application database, developed for academic purposes as part of a study in database systems.

The system aims to provide secure and efficient access to user records, order details, shipping status, and discount coupon data.

The database contains tables that represent the following pages in Shopee, which we deemed important enough to be included:

- Profile pages – both user (buyer) and store (seller)
- Coupon page (discount vouchers)
- Terms and conditions page (for the vouchers)
- Order updates page
- Product page
- Coupon confirmation + application page

As for pages not available to the average user:

- Campaign creation page
- Discount voucher creation page

Project Objectives

- To provide secure and efficient access to data, including and not limited to user and product information.
- To have a database that is normalized to the highest possible degree, void of redundancies, and certain functional dependencies.
- To implement basic data access/control and security (user roles) when modifying the database.

Key Roles

User

- Can add products to order
- Can checkout using payment methods provided by the system (shopee)
- Can delete product from order
- Can edit personal information
- Can view (but cannot edit) shipping status
- Can use discount coupon to their orders
- Can review products and give star ratings to seller
- Can view proof of delivery once the parcel is delivered
- Can view their following + followers

Seller

- Can add and delete products from their store
- Can edit product details and specifications
- Can edit personal information
- Can view new, pending, and completed orders
- Can assign specific shipping instructions
- Can edit product variations
- Can view their following + followers

Driver

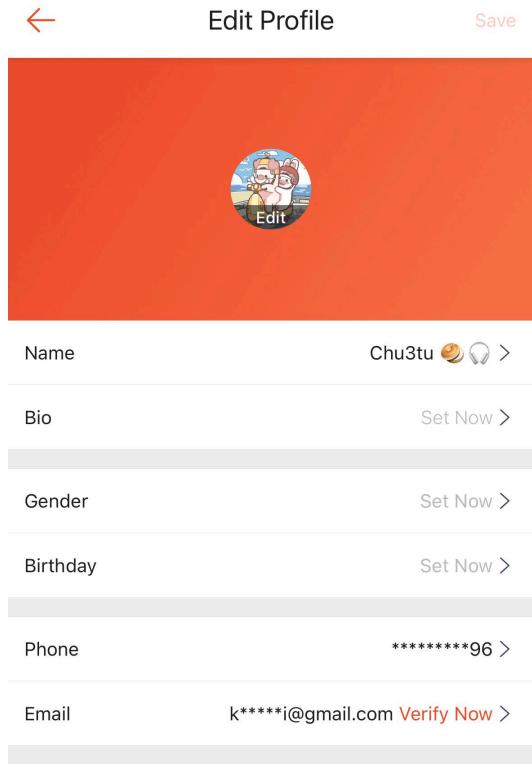
- Can read and access parcel information (order)
 - Address of delivery
 - Any instructions? (fragile, priority delivery, etc.)
 - Delivery time window
- Can edit proof of delivery

Administrator

- Can access and manage all coupon records
- Can edit terms + conditions of discount coupons

- Can add, edit, or delete discount coupons in the database
- Can add, edit, or delete campaigns in the database
- *Amongst other things, but we're only gonna consider + implement what we've said in the previous bullet points.*

Functions and Screens

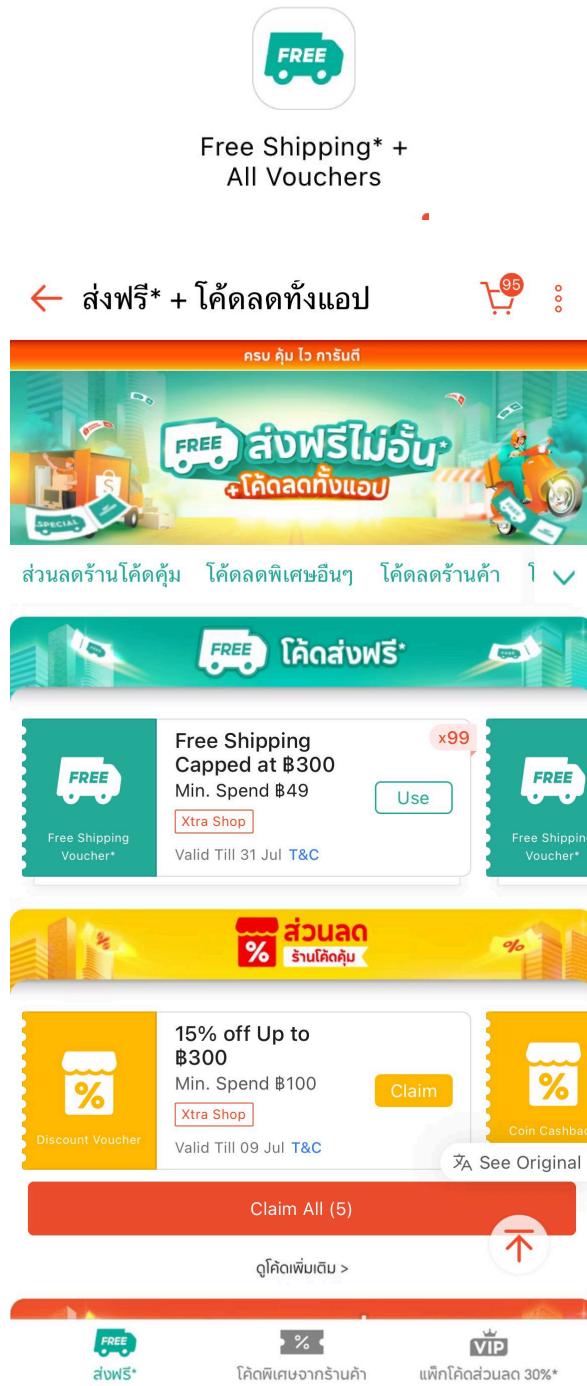


Profile Page

This page allows users to edit their personal information such as name, bio, gender, birthday, phone number, and email. The screen interacts with the User table in database. The user is able to edit the following fields:

- username
- bio
- phone_number
- email
- gender
- birth_date

A function from this page is `get_user_profiles(id)` which returns, if it exists, the record for the profile with the given id. This includes all the aforementioned fields.



Discount Voucher Page

This page provides a list of all available vouchers for users, grouped by categories such as the vouchers are issued/provided by platform, partner, or store (seller).

A function from this page is `get_vouchers()`, which returns all records that are vouchers.

Voucher T&Cs

Free Shipping Capped at ₱300
Min. Spend ₱49
Valid Till: 31.07.2025

Valid Period
1 Jul 2025 00:00 - 31 Jul 2025 23:59

Reward
Free Shipping Voucher | โค้ดส่งฟรี

Payment

- ShopeePay
- GIRO
- QR Promptpay
- Google Pay
- Credit Card Points
- Mobile Banking
- All SPaylater
- Cash on Delivery
- Credit Card/Debit Card

Device
iOS, Android

More Details
Free Shipping Xtra Shop Voucher with ₱49 min. spend

- Shipping discounts are capped as follows:
 - capped at ₱38 with ₱49 min. spend
 - capped at ₱80 with ₱150 min. spend
 - capped at ₱300 with ₱300 min. Spend

Use Voucher

Terms and Conditions Page

This screen shows the detailed terms and conditions for a particular coupon. It helps users understand restrictions like applicable payment methods, platform/device compatibility.

Data is pulled from the Term and Condition table.

The admin can edit following fields from term and condition table:

- `payment_method_id`
- `description`, `min_spend`
- `max_discount`
- `device_type`

A function from this page is `get_voucher_terms(id)` which returns the specific terms of a voucher with the given id.

The screenshot shows a user interface for creating a new discount voucher. At the top right is a 'Cancel' button. The main title is 'Create New Discount Voucher'. Below the title are several input fields and buttons:

- 'Voucher ID (can be auto-generated or manual)': A text input field containing 'type something'.
- 'Voucher Name': A text input field containing 'type something'.
- 'Voucher Type': A button with three options: 'store', 'platform', and 'partner'.
- 'Campaign': A button labeled 'pick from available campaign'.
- 'Voucher Terms' section:
 - 'Description': A text input field containing 'type something'.
 - 'Minimum Spend': A text input field containing '\$100'.
 - 'Maximum Discount': A text input field containing '15% off up to \$300'.
 - 'Valid From' and 'Valid Until': Two date picker buttons.
 - 'Payment Methods': A button labeled 'pick from available in Shopee'.
- 'Save Draft' and 'Publish' buttons at the bottom.

Discount Voucher Creation Page

This page is only available to administrators. It allows the creation and modification of discount coupons.

Admin can edit following fields, such as:

- `voucher_id`, `campaign_id`
- `voucher_name`
- `voucher_type`
- `min_spend`

- max_spend

Tables related to this page include discount_voucher, term_and_condition, payment_method, and campaign. As it is a role-limited page, there are security measures implemented (RBAC) to ensure that only administrators can access and modify this page.

The screenshot shows a 'Create New Campaign' form. At the top right is a 'Cancel' button. The main title is 'Create New Campaign'. Below it is a field for 'Campaign ID (can be auto-generated or manual)' with a placeholder 'type something'. Next is a 'Campaign Name' field with a placeholder 'type something'. Below these are 'Start Date' and 'End Date' fields, each with a 'date picker' button. Underneath is a 'Status' field with two options: 'active' and 'inactive'. A 'Select Existing Vouchers' section contains a list box with the text 'list of vouchers not yet linked to any campaign'. At the bottom are 'Save Draft' and 'Publish' buttons.

Campaign Creation Page

This is another admin page where they can create promotional campaigns which group multiple vouchers. The table used for this page is campaign.

The admin can edit follow fields:

- campaign_id
- campaign_name
- start_date
- end_Date
- campaign_status

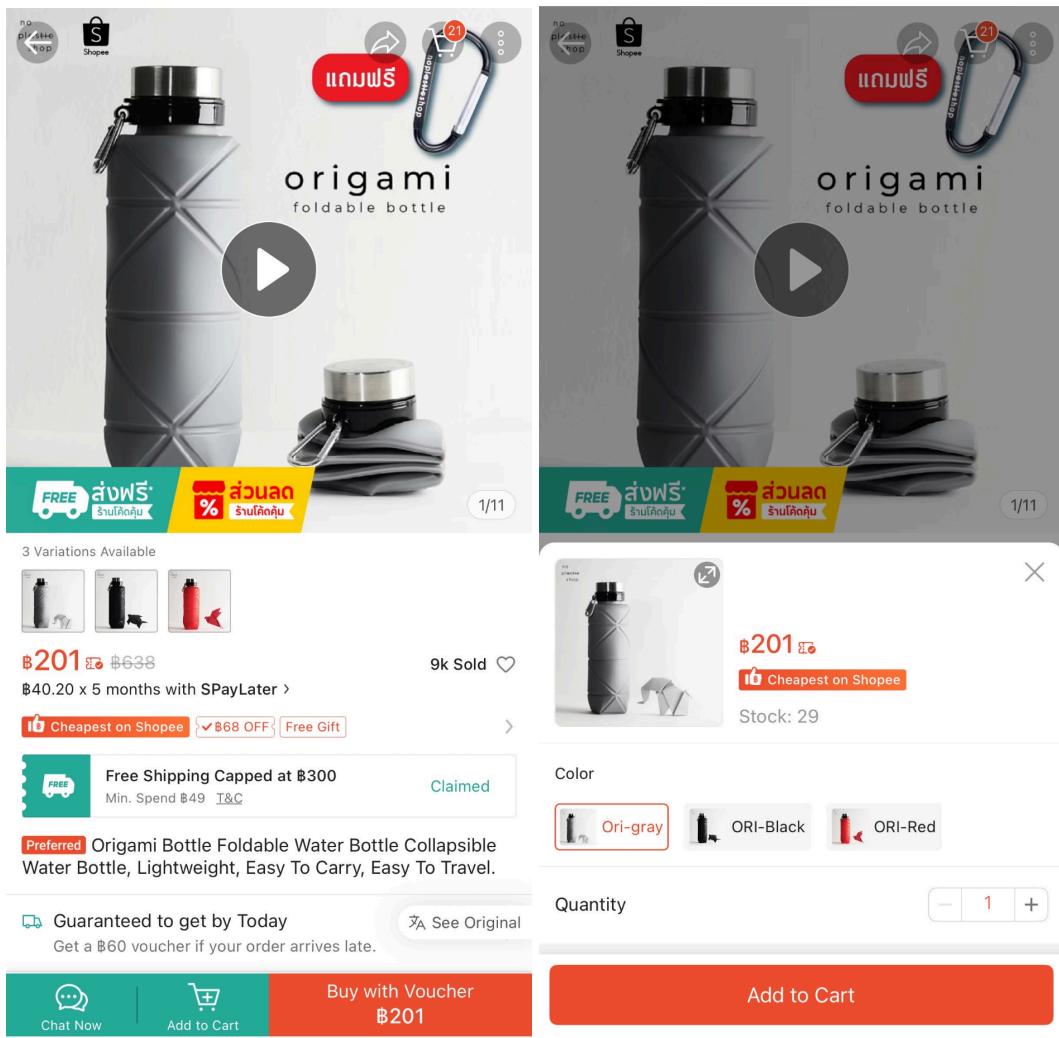
Order Updates	Read All
Parcel Delivered Parcel TH258581837989Z for your order 250630V10KUYGW has been delivered.	▼
02-07-2568 15:18	
Parcel Delivered Parcel TH15077EAK2K7A for your order 250630V10KUYGV has been delivered.	▼
02-07-2568 11:06	
Your order is completed Order 2506228W2UDARM is completed. Your feedback matters to others! Rate the products by 28-07-2025 and get up to 0.60 coins.	▼
28-06-2568 19:20	
Your order is completed Order 2506228W2UDARH is completed. Your feedback matters to others! Rate the products by 28-07-2025 and get up to 0.60 coins.	▼
28-06-2568 19:20	
Your order is completed Order 2506228W2UDARJ is completed. Your feedback matters to others! Rate the	

Order Updates Page

This page shows real-time order status to users, such as parcel delivery and order completion updates. It pulled the data from order and shipping update tables.

The driver can send the proof of delivery when the parcel is delivered or arrives at the destination.

A function from this page is `get_order_status(id)`, which returns the status, tracking number, and description of the order with the given id.



Product Page

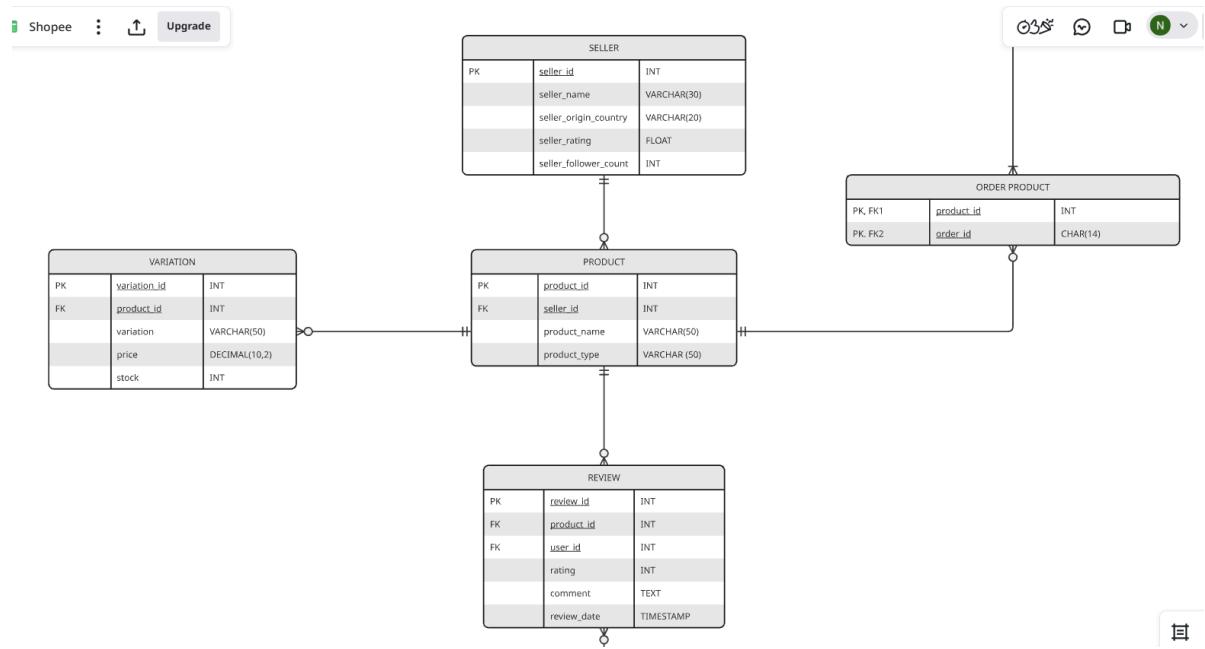
This screen shows product details including image, price, ratings, variations, stock and vouchers applied. It pulls the data from tables like product, variation, seller and review.

The seller can edit the following fields:

- seller_name
- product_name
- product_type
- stock
- price
- variation (color variation, size variation, etc.)

A function from this page is `get_product_info(product_id, variation_id)` that retrieves information of a product with a specific ID and one variation with a specific ID, provided that both of those IDs exist.

Database Design



Product

Attributes	Attribute Data Type	Required or Optional	Example Instance	
product_id	INT	Require	1234567	
seller_id	INT	Require	1528941	
product_name	VARCHAR(50)	Require	Iphone 16 pro max	
product_type	VARCHAR(10)	Require	Electronics	
+				

Seller

Attributes	Attribute Data Type	Required or Optional	Example Instance	+
seller_id	INT	Require	12778	
seller_name	VARCHAR(30)	Require	Studio 7	
seller_origin_country	VARCHAR(20)	Require	Thailand	
seller_rating	FLOAT	Require	4.7	
seller_follower_count	INT	Require	40576	

+

Variation

Attributes	Attribute Data Type	Required or Optional	Example Instance	+
variation_id	INT	Require	101	
product_id	INT	Require	1234567	
variation	VARCHAR(50)	Optional	blue	
price	DECIMAL(10,2)	Require	567	
stock	INT	Require	5	

+

Review

Attributes	Attribute Data Type	Required or Optional	Example Instance	+
review_id	INT	Require	239048203482903842	
product_id	INT	Require	1234567	
user_id	INT	Require	1	
rating	INT	Require	4	
comment	TEXT	Require	good quality, fast delivery	
review_date	DATETIME	Require	2025-07-01 00:00:00	

+

Order Product

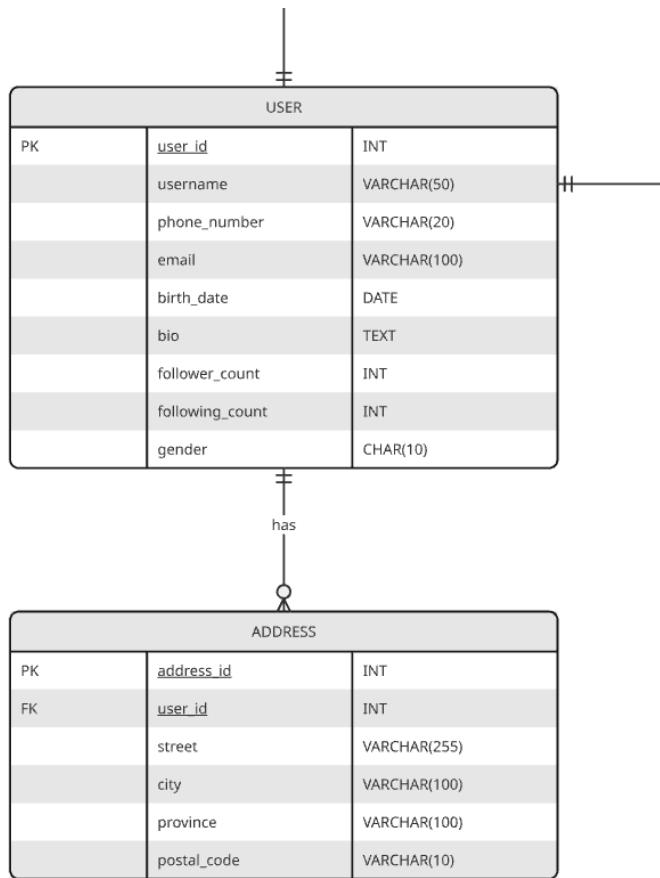
The screenshot shows a database schema editor interface with a toolbar at the top. Below the toolbar is a table with four columns: 'Attributes', 'Attribute Data Type', 'Required or Optional', and 'Example Instance'. There are two rows in the table. The first row contains 'product_id' (INT type, required, example value 1234567). The second row contains 'order_id' (CHAR(14) type, required, example value 250630V10KUYGW). A '+' button is located at the bottom left of the table.

Attributes	Attribute Data Type	Required or Optional	Example Instance	+
product_id	INT	Require	1234567	
order_id	CHAR (14)	Require	250630V10KUYGW	

This part of the database covers product management, including sellers, product variations or product choices, and reviews. It supports Shopee pages like product detail pages.

This part contains entities:

- Seller: stores seller details has seller_id as a primary key
- Product: stores product details has product_id as a primary key and seller_id as a foreign key
- Variation: stores product variants (e.g. size, color) has variation_id as a primary key and product_id as a foreign key
- Review: stores user reviews of products has review_id as a primary key, product_id and user_id as a foreign key



User

Attributes	Attribute Data Type	Required or Optional	Example Instance	
user_id	INT	Require	1023456	
username	VARCHAR(50)	Require	churose	
phone_number	VARCHAR(20)	Require	0812345678	
email	VARCHAR(100)	Require	churose@gmail.com	
birth_date	DATE	Require	2022-07-25	
bio	TEXT	Optional	loves chicken food	
follower_count	INT	Optional	78	
following_count	INT	Optional	152	
gender	CHAR(1)	Require	M	

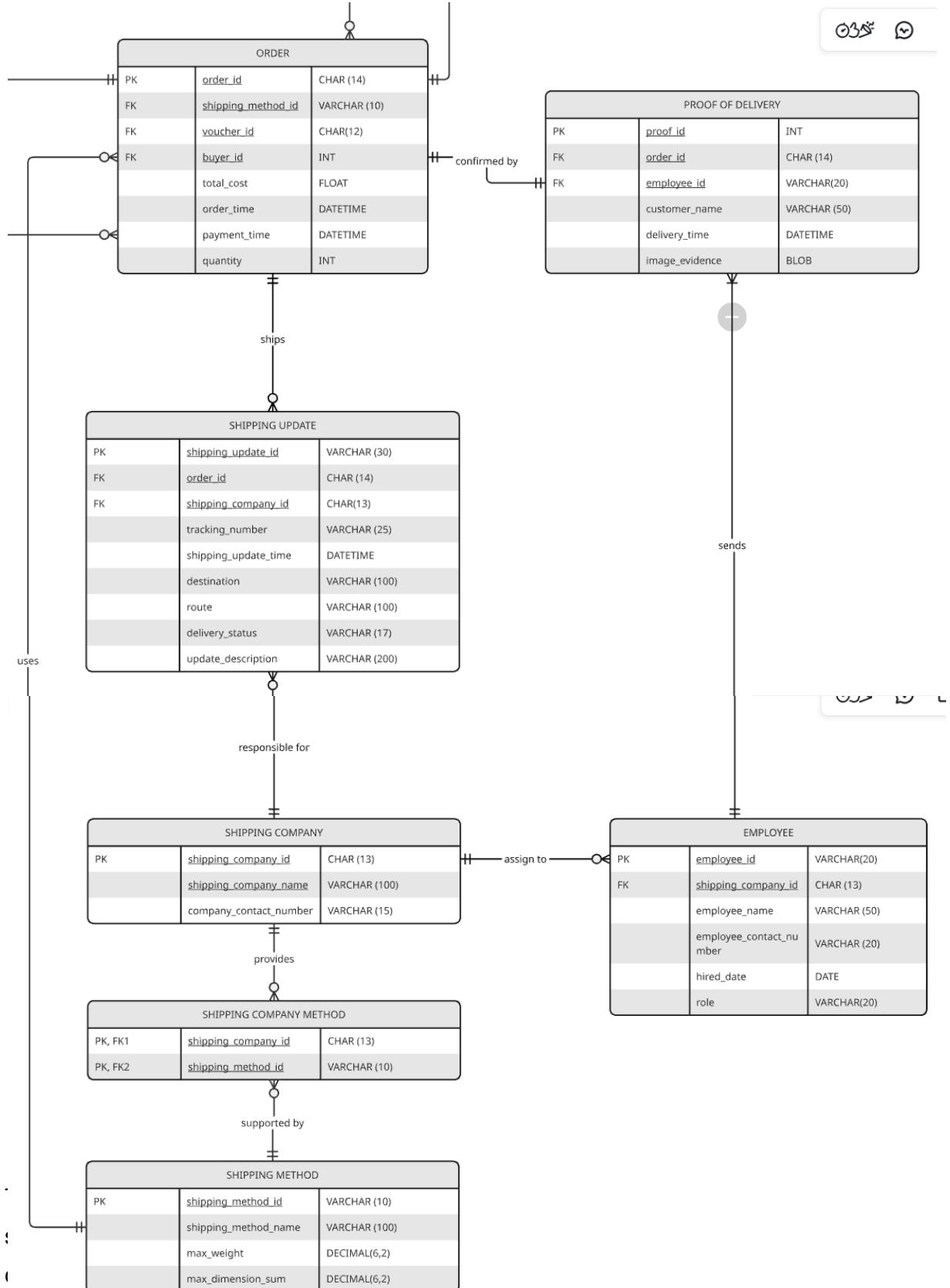
Address

Attributes	Attribute Data Type	Required or Optional	Example Instance	+
address_id	INT	Require	256489	
user_id	INT	Require	1023456	
street	VARCHAR(255)	Require	123 Main Street	
city	VARCHAR(100)	Require	Bangkok	
province	VARCHAR(100)	Require	Bangkok	
postal_code	VARCHAR(10)	Require	10110	

This part of the database covers user details and user addresses. It supports Shopee pages like user profiles.

This part contains entities:

- user: stores user details has user_id as a primary key
- address: stores user addresses has address_id as primary key and user_id as foreign key



Order

<input type="checkbox"/> Attributes	<input type="checkbox"/> Attribute Data Type	<input type="checkbox"/> Required or Optional	<input type="checkbox"/> Example Instance	<input type="checkbox"/>
order_id	CHAR (14)	Require	250630V10KUYGW	
shipping_method_id	VARCHAR (10)	Require	4569871	
voucher_id	CHAR(12)	Optional	VC20250701A	
buyer_id	INT	Require	1024537	
total_cost	FLOAT	Require	153.50	
order_time	DATETIME	Require	2025-07-01 10:15:00	
payment_time	DATETIME	Require	2025-07-01 10:15:01	
quantity	INT	Require	1	

Shipping Update

<input type="checkbox"/> Attributes	<input type="checkbox"/> Attribute Data Type	<input type="checkbox"/> Required or Optional	<input type="checkbox"/> Example Instance	<input type="checkbox"/>
shipping_update_id	VARCHAR (30)	Require	23456	
order_id	CHAR (14)	Require	12345	
shipping_company_id	CHAR (13)	Require	0105560159254	
tracking_number	VARCHAR (25)	Require	TH258581837989Z	
shipping_update_time	DATETIME	Require	2025-07-02 11:42:00	
destination	VARCHAR (100)	Require	APUTM - Phutthamonthon	
route	VARCHAR (100)	Require	From ABKPI-B to APTM	
delivery_status	VARCHAR (17)	Require	Out for Delivery	
update_description	VARCHAR (200)	Require	Parcel is in the process of being delivered. You should receive your parcel by 8:00 p.m.	

Employee

<input type="checkbox"/> Attributes	<input type="checkbox"/> Attribute Data Type	<input type="checkbox"/> Required or Optional	<input type="checkbox"/> Example Instance	<input type="checkbox"/>
employee_id	VARCHAR (20)	Require	1122334455	
shipping_company_id	CHAR (13)	Require	0105560159254	
employee_name	VARCHAR (50)	Require	James John	
employee_contact_number	VARCHAR (20)	Require	0815555555	
hired_date	DATE	Require	2025-07-05	
role	VARCHAR(20)	Require	driver	

Shipping Company

<input type="checkbox"/> Attributes	<input type="checkbox"/> Attribute Data Type	<input type="checkbox"/> Required or Optional	<input type="checkbox"/> Example Instance	<input type="checkbox"/> Example Instance	<input style="border: none; background-color: transparent;" type="button" value="+"/>
shipping_company_id	CHAR (13)	Require	0105560159254	0105548000160	
shipping_company_name	VARCHAR (100)	Require	Flash Express	DHL	
company_contact_number	VARCHAR (15)	Require	02-168-2111	02-017-8333	

Shipping Method

<input type="checkbox"/> Attributes	<input type="checkbox"/> Attribute Data Type	<input type="checkbox"/> Required or Optional	<input type="checkbox"/> Example Instance	<input type="checkbox"/> Example Instance	<input style="border: none; background-color: transparent;" type="button" value="+"/>
shipping_method_id	VARCHAR (10)	Require	STD	BULKY	
shipping_method_name	VARCHAR (100)	Require	Standard Delivery	Standard Delivery Bulky	
max_weight	DECIMAL(6,2)	Require	20	100	
max_dimension_sum	DECIMAL(6,2)	Require	180	600	

Proof of Delivery

<input type="checkbox"/> Attributes	<input type="checkbox"/> Attribute Data Type	<input type="checkbox"/> Required or Optional	<input type="checkbox"/> Example Instance	<input style="border: none; background-color: transparent;" type="button" value="+"/>
proof_id	INT	Require	112233	
order_id	CHAR(14)	Require	250630V10KUYGW	
customer_name	VARCHAR (50)	Require	churose	
delivery_time	DATETIME	Require	2025-07-05 15:32:00	
image_evidence	BLOB	Require	-	

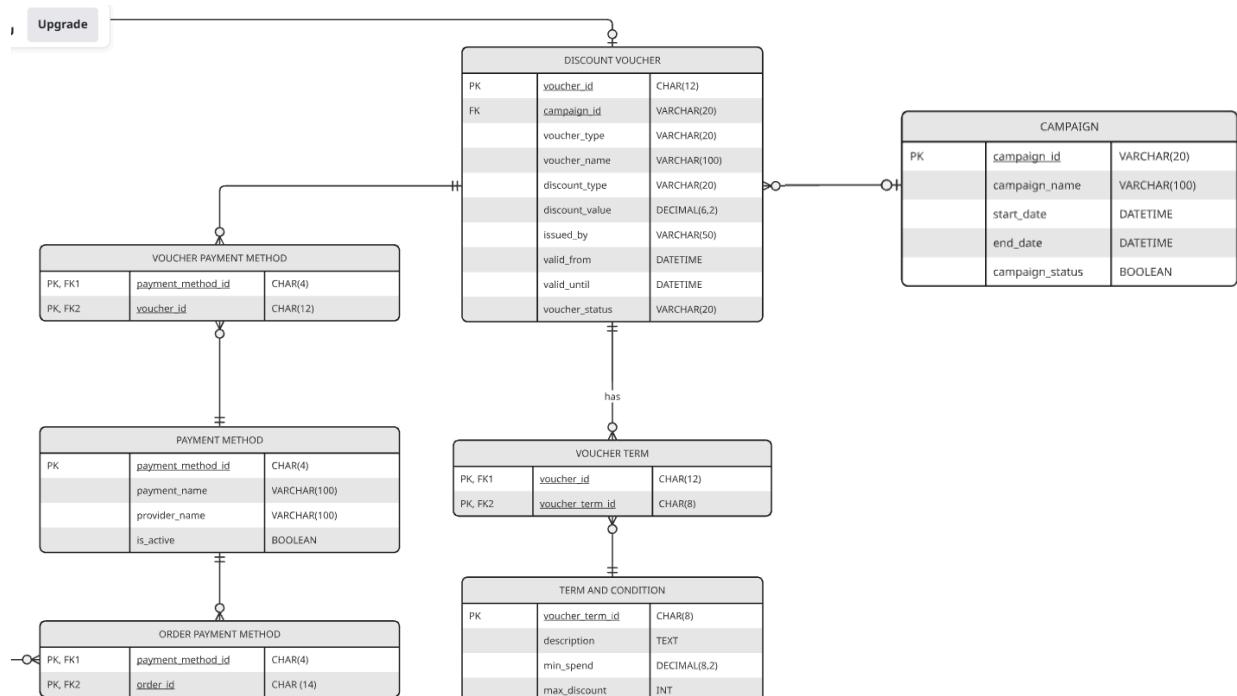
Shipping Company Method

<input type="checkbox"/> Attributes	<input type="checkbox"/> Attribute Data Type	<input type="checkbox"/> Required or Optional	<input type="checkbox"/> Example Instance	<input type="checkbox"/> Example Instance	<input style="border: none; background-color: transparent;" type="button" value="+"/>
company_id	CHAR (13)	Require	0105560159254	0105548000160	
method_id	VARCHAR (10)	Require	15234	56478	

This part contains entities:

- Order: stores user orders and linked to payment and shipping methods, has order_id as a primary key. shipping_method_id, user_id, voucher_id, buyer_id as foreign keys
- Shipping Update: tracks delivery status and updates with tracking numbers, has shipping_update_id as a primary key, order_id and shipping_company as foreign keys
- Shipping Company: stores shipping service providers and their informations, has shipping_company_id as a primary key

- Shipping Method: details various shipping methods supported (e.g., standard delivery, bulky) has `shipping_method` as a primary key
- Employee: employees assigned to delivery has `employee_id` as a primary key and `shipping_company_id` as a foreign key



Payment Method

	Attributes	Attribute Data Type	Required or Optional	Example Instance	
	payment_method_id	CHAR(4)	Require	SPAY	
	payment_name	VARCHAR(100)	Require	QR PromptPay	
	provider_name	VARCHAR(100)	Require	Shopee	
	is_active	BOOLEAN	Require	TRUE	

Discount Voucher

	Attributes	Attribute Data Type	Required or Optional	Example Instance	+
	voucher_id	CHAR(12)	Require	VC20250701A	
	campaign_id	VARCHAR(20)	Optional	CAMP_202507_77SALE	
	voucher_type	ENUM (STORE, PLATFORM, PARTNER)	Require	PLATFORM	
	voucher_name	VARCHAR(100)	Require	50% off Up to ₱20	
	discount_type	ENUM (PERCENT, FIXED)	Require	PERCENT	
	discount_value	DECIMAL(6,2)	Require	10	
	issued_by	VARCHAR(50)	Require	Shopee Food	
	valid_from	DATETIME	Require	2025-07-01 00:00:00	
	valid_until	DATETIME	Require	2025-07-31 23:59:00	
	voucher_status	ENUM (available, fully redeemed)	Require	available	

Term and Condition

	Attributes	Attribute Data Type	Required or Optional	Example Instance	+
	voucher_term_id	CHAR(8)	Require	VT000123	
	description	TEXT	Require	Save ₱200 on orders over ₱500	
	min_spend	DECIMAL(8,2)	Require	500	
	max_discount	DECIMAL(8,2)	Require	200	

Voucher Term

	Attributes	Attribute Data Type	Required or Optional	Example Instance	+
	voucher_id	CHAR(12)	Require	VC20250701A	
	voucher_term_id	CHAR(8)	Require	VT000123	

Order Payment Method

	Attributes	Attribute Data Type	Required or Optional	Example Instance	+
	payment_method_id	CHAR(4)	Require	SPAY	
	order_id	CHAR (14)	Require	250630V10KUYGW	
+					

campaign

	Attributes	Attribute Data Type	Required or Optional	Example Instance	+
	campaign_id	VARCHAR(20)	Require	CAMP_202507_77SALE	
	campaign_name	VARCHAR(100)	Require	7.7 Big Sale	
	start_date	DATETIME	Require	2025-07-31 00:00:00	
	end_date	DATETIME	Require	2025-07-31 02:00:00	
	campaign_status	BOOLEAN	Require	TRUE	
+					

Voucher Payment Method

	Attributes	Attribute Data Type	Required or Optional	Example Instance	+
	voucher_id	CHAR(12)	Require	VC20250701A	
	payment_method_id	CHAR(4)	Require	SPAY	
+					

This part of the database manages discount vouchers, payment methods, and campaigns in the Shopee system. It supports Shopee pages like coupon creation page, campaign creation page, and payment page.

This part contains entities:

- Discount Voucher: stores voucher details like coupon type, coupon name, and their campaigns, has voucher_id as a primary key and campaign_id as a foreign key
- Campaign: stores events in Shopee, has campaign_id as a primary key

- Term and Condition: stores detailed rules for particular vouchers such as description, minimum spending. Has voucher_term_id as a primary key
- Payment Method: lists payment options supported by Shopee, has payment_method_id as a primary key

Security Mindset

We implemented role-based permissions by creating normal user (buyer), seller, driver and admin roles (as mentioned briefly in the “Key Roles” section of the report). We do this to ensure data integrity and security, meaning that only authorized users can access and modify sensitive records. The four roles have the following permissions:

- A user can either be a buyer or seller:
 - Buyer: can create, update, and delete orders, user profile.
 - Seller: can create, update, and delete products and variations.
- Driver: can create, update, and delete proof of delivery.
- Admin: has full access to the discount_voucher and campaign table, including creating, updating, and deleting vouchers + campaigns.

Performance Optimization

1. Trigger

```
update_seller_rating()
```

We implement a trigger in this function to automatically calculate and update a seller's rating whenever a new review (with a rating) is added to said seller.

2. Indexes

We created indexes on these attributes to enhance searching performance, as we feel like these will be frequently accessed:

- idx_users_username: searching for users by username
- idx_orders_buyer_id: searching for orders by buyer id
- idx_review_product_id: searching for reviews by product id
- idx_review_user_id: searching for reviews by user id
- idx_shipping_update_order_id: searching for shipping updates by order id
- idx_product_seller_id: searching for products by seller id

Deployment

- Link to our Github repository
- Link to our Miro (ER Diagrams)

https://miro.com/welcomeonboard/aC9MLy9FOU90VUwrWmpPOVdCOFF3NXpVWHUvd3ZIVG5oanISaDFnWmVyeEZLM3hpWWtxSXI5Y2JseGEyUXgyVHVtMS9ycjNtNXJYNGlaTHpFTmFRYytObTZiZ1ZITE1oYmRky2V2UDM5SkM0QS9KVi9YdW1TZN0RXZwTitKWWI3VHhHVHd5UWtSM1BidUtUYmxycDRnPT0hdjE=?share_link_id=734351642687

Conclusion

This project is a recreation of Shopee's database, focusing mainly on pages that we deemed were integral enough to be included. We chose to focus on deploying essential features such as user (buyer) and store (seller) profiles, product and variation management, order tracking, discount voucher application, and campaign creation. We also included convenience functions, allowing users and admins to modify the database based on their permission level. We also implemented security and performance optimizations to compensate for larger queries, allowing for better efficiency.