



De La Salle University - Manila

College of Computer Studies

Term 1, AY 2025-2026

In partial fulfillment of the course

INFORMATION MANAGEMENT - CCINFOM S17

**Pharmacy Management System
Database Application Project Proposal**

Submitted by:

Guo, Anie H.

Tan, Andrea Jadyn K.

Uy, Tara Ysabel P.

Submitted to:

Ms. Marivic Tangkeko

Submitted on:

November 11, 2025

Section 1.0 Group Composition

Group Leader (Group Member 1): Uy, Tara Ysabel P.

Group Member 2: Guo, Anie H.

Group Member 3: Tan, Andrea Jadyn K.

Section 2.0 Why is this Database System important to be developed

Pharmaceutical companies and pharmacies handle large volumes of data on medicines, customers, and suppliers. While Excel may work at first, it quickly becomes messy when tracking large inventories and restocking. A database system provides a structured way to store and connect records, making it easier to manage medicine availability, track supplier deliveries, and generate accurate reports that support decision-making and keep operations running smoothly.

Section 3.0 Records Management

Medicine Record Management	(medicine_id (PK), name, price_bought, price_for_sale, quantity_in_stock, expiration_date, discontinued, delivery_date)	assigned to Uy, Tara Ysabel P.
----------------------------	---	---------------------------------------

Viewing a Record with Other Related Records:

- Viewing of a specific medicine record and the list of suppliers providing it

Business Rules:

- When the medicine is discontinued or expired, it cannot be sold.
- price_bought, price_for_sale, quantity cannot be negative.
- When the medicine is returned to the supplier, stock levels must update.
- A medicine cannot be deleted from the database, they are marked as discontinued.

Customer Record Management	(customer_id (PK), name, contact_info, senior_pwd_id, customer_status)	assigned to Guo, Anie H.
----------------------------	--	---------------------------------

Viewing a Record with Other Related Records:

- Viewing of a customer record and the list of medicines they purchased

Business Rules:

- Senior_pwd id must be unique per customer.
- If the customer's senior_pwd id is not null, then a 20% discount is applied.
- The customer must be registered before they can purchase a medicine.
- A customer cannot be deleted from the database, they are marked as inactive.

Supplier Record Management	(supplier_id (PK), name, address, contact_info, supplier_status)	assigned to Tan, Andrea Jadyn K.
----------------------------	--	---

Viewing a Record with Other Related Records:

- Viewing of a supplier record and the medicines they provide to the pharmacy

Business Rules:

- The supplier cannot be deleted, they are marked as inactive.
 - Only active suppliers can process returns or deliveries.
-

Section 4.0 Transactions

1. Purchasing of Medicine as a Transaction will involve the following data, operations **assigned to Guo, Anie H.**

Core Records Involved: Medicine Record, Customer Record

Attributes: purchase_no, purchase_date, customer ID, customer Name, senior_pwd_id (if applicable), customer contact info, customer_status, medicine ID, medicine Name, quantity ordered, price of medicine, expiration_date, quantity_in_stock, discontinued, discount, total
Customer Record: customer_id (PK)

Associative Entity purchase: purchase_no (PK), customer_id (FK)

Associative Entity purchase_details: purchase_no (PK, FK), medicine_id (PK, FK)

Medicine Record: medicine_id (PK)

- a. Reading the customer record to verify registration (create new if not yet registered)
- b. Reading the customer record to verify if the customer is a senior citizen or PWD to apply a 20% discount
- c. Reading the customer record to verify that the customer is active
- d. Reading the medicine record to confirm the stock, expiry, and discontinued status.
- e. Recording the purchase transaction (purchase_no, medicine_id, quantity_ordered, discount, total)
- f. Updating the medicine record to deduct the quantity sold
- g. Generating a receipt for the customer

Business Rules:

- Medicine can only be purchased if they are not expired, not discontinued, and in stock.
- An inactive customer's transaction will remain for tracking purposes.

2. Delivering of Medicine from Supplier as a Transaction will involve the following data, operations **assigned to Uy, Tara Ysabel P.**

Core Records Involved: Medicine Record, Supplier Record

Attributes: delivery_no, shipped_date, request_date, status, supplier_id, supplier_name, contact_info, supplier_address, supplier_status, medicine_id, medicine_name, medicine_quantity_in_stock, price_bought, medicine_delivery_date, discontinued, expiration_date, quantity_bought, total
Supplier Record: supplier_id (PK)

Associative Entity delivery: delivery_no (PK), supplier_id (FK)

Associative Entity delivery_details: delivery_no (PK, FK), medicine_id (PK, FK)

Medicine Record: medicine_id (PK)

- a. Reading the medicine records to check stock levels
- b. Reading the supplier record to check that the supplier is still active
- c. Recording the delivery request (delivery_no, medicine_id, quantity_bought, total)
- d. Updating the medicine stock once the restock is completed
- e. Recording the supplier delivery for tracking

Business Rules:

- Only active suppliers can supply medicines.
- Inactive supplier remains for tracking purposes.

3. Returning of Medicine (discontinued or expired) to Supplier as a Transaction will involve the following data, operations **assigned to Tan, Andrea Jadyn K.**

Core Records Involved: Medicine Record, Supplier Record

Attributes: return_no, reason, shipped_date, request_date, return_status, supplier_id, supplier_name, contact_info, address, supplier_status, medicine_id, medicine_name, discontinued, expire_date, medicine_delivery_date, quantity_returned

Supplier Record: supplier_id (PK)

Associative Entity return: return_no (PK), supplier_id (FK)

Associative Entity return_details: return_no (PK, FK), medicine_id (PK, FK)

Medicine Record: medicine_id (PK)

- a. Reading the medicine records to identify expired or discontinued medicines
- b. Reading the medicine records to verify if the medicine delivery date is within 7 days
- c. Reading the supplier record to verify supplier, delivery, and supplier status
- d. Recording return (return_no, medicine_id, quantity_returned, price_returned)
- e. Updating medicine record (send back to supplier)
- f. Generating a receipt for the return

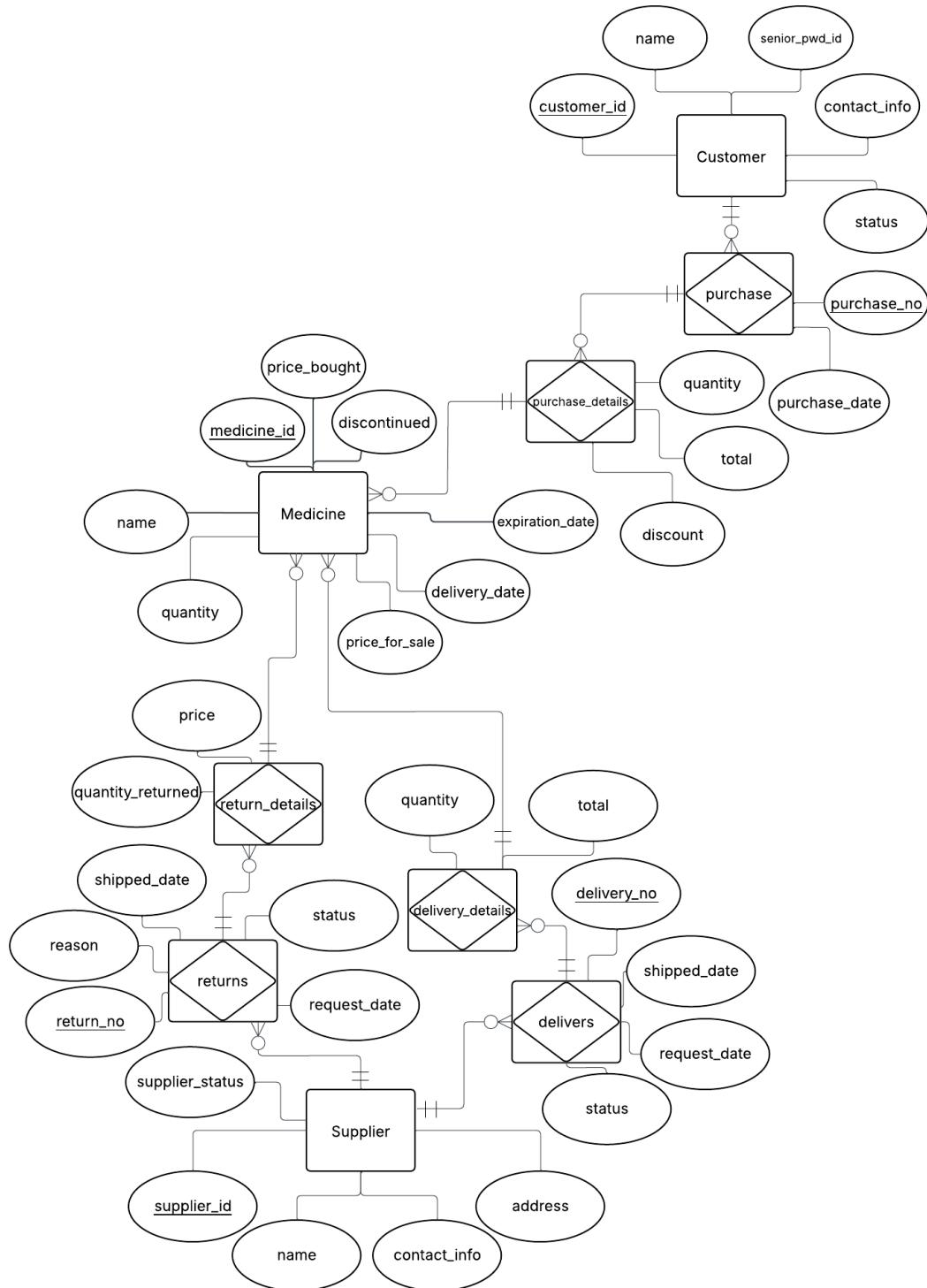
Business Rules:

- Only medicines supplied by that supplier may be returned.
- Returns must be initiated within 7 days of delivery.
- When a medicine is discontinued or expired, they can be returned to the supplier.

Section 5.0 Reports to be Generated

Customer Purchase Report	Core/Transaction Used: Medicine Record, Customer Record, Selling of Medicine Transaction (number and total amount of sales transactions) per customer for a given Month and Year	assigned to Guo, Anie H.
Medicine Return Report	Core/Transaction Used: Medicine Record, Supplier Record, Return Transaction (number and total amount of returns) per medicine for a given Month and Year	assigned to Uy, Tara Ysabel P.
Procurement Report	Core/Transaction Used: Medicine Record, Supplier Record, Restocking of Medicine from Supplier Transaction (number and total amount of restocks) per supplier for a given Month and Year	assigned to Tan, Andrea Jadyn K.

Entity Relationship Diagram



Transform to Relations



Normalize

Purchasing of Medicine

`purchase_no, medicine_id` → `quantity_ordered, discount, total` (Full Dependency)

`purchase_no` → `purchase_date, customer_id, customer_name, senior_pwd_id, contact_info, customer_status` (Partial Dependency)

`medicine_id` → `medicine_name, price_for_sale, expiration_date, quantity_in_stock, discontinued` (Partial Dependency)

`customer_id` → `customer_name, senior_pwd_id, contact_info, customer_status` (Transitive Dependency)

UNF

`purchasing(purchase_no, purchase_date, customer_id, customer_name, senior_pwd_id, contact_info, customer_status, {medicine_id, medicine_name, quantity_ordered, price_for_sale, expiration_date, quantity_in_stock, discontinued, discount, total})`

1NF

`purchasing(purchase_no, purchase_date, customer_id, customer_name, senior_pwd_id, contact_info, customer_status, medicine_id, medicine_name, quantity_ordered, price_for_sale, expiration_date, quantity_in_stock, discontinued, discount, total)`

2NF

`purchase_detail(purchase_no, medicine_id, quantity_ordered, discount, total)`

medicine(medicine_id, medicine_name, price_for_sale, expiration_date, quantity_in_stock, discontinued)
 purchase(purchase_no, purchase_date, customer_id, customer_name, senior_pwd_id, contact_info, customer_status)
 3NF
 purchase_detail(purchase_no, medicine_id, quantity_ordered, discount, total)
 medicine(medicine_id, medicine_name, price_for_sale, expiration_date, quantity_in_stock, discontinued)
 purchase(purchase_no, purchase_date, customer_id, customer_status)
 customer(customer_id, customer_name, senior_pwd_id, contact_info, customer_status)

Delivering of Medicine

delivery_no, medicine_id -> quantity_bought, total (Full Dependency)
 delivery_no -> shipped_date, request_date, delivery_status, supplier_id, supplier_name, contact_info, supplier_address, supplier_status (Partial Dependency)
 medicine_id -> medicine_name, medicine_quantity_in_stock, price_bought, discontinued, expiration_date, medicine_delivery_date (Partial Dependency)
 supplier_id -> supplier_name, contact_info, supplier_address, supplier_status (Transitive Dependency)
 UNF

delivery(delivery_no, shipped_date, request_date, delivery_status, supplier_id, supplier_name, contact_info, supplier_address, supplier_status, {medicine_id, medicine_name, medicine_quantity_in_stock, price_bought, discontinued, expiration_date, medicine_delivery_date, quantity_bought, total})
 1NF

delivery(delivery_no, shipped_date, request_date, delivery_status, supplier_id, supplier_name, contact_info, supplier_address, supplier_status, medicine_id, medicine_name, medicine_quantity_in_stock, price_bought, discontinued, expiration_date, medicine_delivery_date, quantity_bought, total)

2NF

delivery_details(delivery_no, medicine_id, quantity_bought, total)
 medicine(medicine_id, medicine_name, medicine_quantity_in_stock, price_bought, discontinued, expiration_date, medicine_delivery_date)
 delivery(delivery_no, shipped_date, request_date, delivery_status, supplier_id, supplier_name, contact_info, supplier_address, supplier_status)

3NF

delivery_details(delivery_no, medicine_id, quantity_bought, total)
 medicine(medicine_id, medicine_name, medicine_quantity_in_stock, price_bought, discontinued, expiration_date, medicine_delivery_date)
 delivery(delivery_no, shipped_date, request_date, delivery_status, supplier_id)
 supplier(supplier_id, supplier_name, contact_info, supplier_address, supplier_status)

Returning of Medicine

return_no, medicine_id -> quantity_returned, price_returned (Full Dependency)
 return_no -> reason, shipped_date, request_date, status, supplier_id, supplier_name, contact_info, address, supplier_status (Partial Dependency)
 medicine_id -> medicine_name, discontinued, expire_date, price_bought (Partial Dependency)
 supplier_id -> supplier_name, contact_info, address, supplier_status (Transitive Dependency)
 UNF

```
return(return_no, reason, shipped_date, request_date, status, supplier_id, supplier_name,
contact_info, address, supplier_status, {medicine_id, medicine_name, discontinued, expire_date,
price_bought, quantity_returned, price_returned})  
1NF  
return(return_no, reason, shipped_date, request_date, status, supplier_id, supplier_name,
contact_info, address, supplier_status, medicine_id, medicine_name, discontinued, expire_date,
price_bought, quantity_returned, price_returned)  
2NF  
return_details(return_no, medicine_id, quantity_returned, price_returned)  
return(return_no, reason, shipped_date, request_date, status, supplier_id, supplier_name,
contact_info, address, supplier_status)  
medicine(medicine_id, medicine_name, discontinued, expire_date, price_bought)  
3NF  
return_details(return_no, medicine_id, quantity_returned, price_returned)  
medicine(medicine_id, medicine_name, discontinued, expire_date, price_bought)  
return(return_no, reason, shipped_date, request_date, status, supplier_id)  
supplier(supplier_id, supplier_name, contact_info, address, supplier_status)
```

Section 6.0 Declaration of Generative AI Use

The group used ChatGPT (OpenAI, GPT-5) to assist in brainstorming project ideas and to help brainstorm possible transactions and reports. The group members reviewed, edited, and customized the content to ensure accuracy and alignment with project proposal guidelines. No part of the proposal was fully generated or rewritten solely by AI.