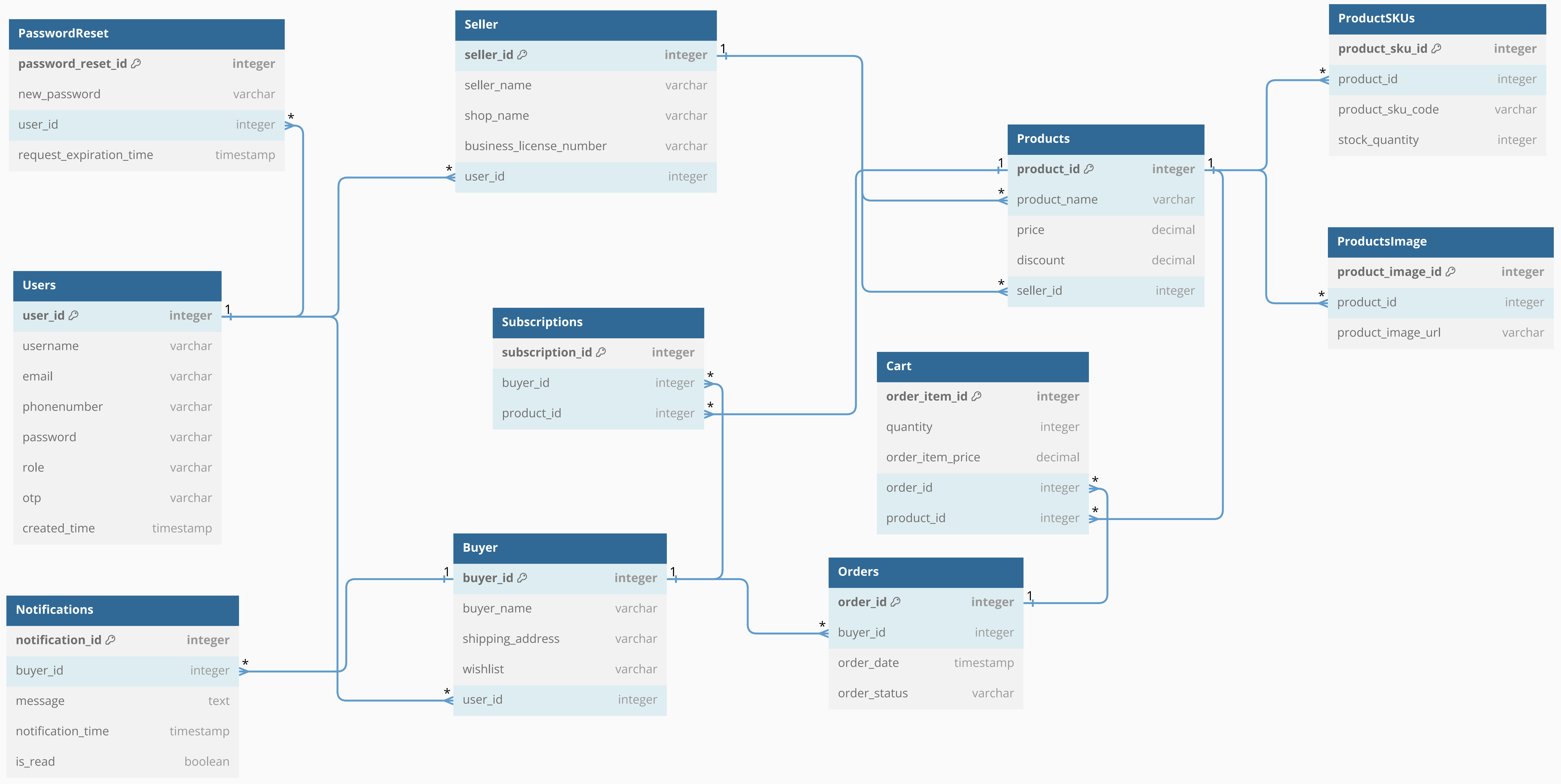
1. Physical Entity Relationship diagram of database.



PasswordReset	
password_reset_id	integer
new_password	varchar
user_id	integer
request_expiration_time	timestamp

Seller	
seller_id	integer
seller_name	varchar
shop_name	varchar
business_license_number	varchar
user_id	integer

ProductSKUs	
product_sku_id	integer
product_id	integer
product_sku_code	varchar
stock_quantity	integer

Products	
product_id	integer
product_name	varchar
price	decimal
discount	decimal
seller_id	integer

ProductsImage	
product_image_id	integer
product_id	integer
product_image_url	varchar

Users	
user_id	integer
username	varchar
email	varchar
phonenumber	varchar
password	varchar
role	varchar
otp	varchar
created_time	timestamp

Subscriptions	
subscription_id	integer
buyer_id	integer
product_id	integer

Cart	
order_item_id	integer
quantity	integer
order_item_price	decimal
order_id	integer
product_id	integer

Buyer	
buyer_id	integer
buyer_name	varchar
shipping_address	varchar
wishlist	varchar
user_id	integer

Orders	
order_id	integer
buyer_id	integer
order_date	timestamp
order_status	varchar

Notifications	
notification_id	integer
buyer_id	integer
message	text
notification_time	timestamp
is_read	boolean

2. Explain about searching performance. How will you handle replication in SQL for searching & Reporting?

- We can use Indexing to search more frequent columns like Product Name, Product Price, or Product Discount
- While fetching the data we can use pagination which will reduce massive data on database for single point of time
- We can use caching for frequently accessed data and avoid repetitive db query for same data.
- We can implement load balancing to distribute incoming fetch queries to replicate the data.

3. Explain what major factors are taken into consideration for performance.

- Latency between the application system and database server must be low to increase the performance of the whole application.
- Don't do costly operations on database for example avoid full scan queries on database rather query for columns with where clause or avoid joins and subquery over database as they are costly and take extra time to complete.
- Before taking it to real world data simulate the virtual environment to test the performance by using load and performance testing.
- Reuse the database connection over application layer don't make new database connection or object for new or different operations.

4. Mention about Indexing, Normalization and Denormalization.

- **Indexing**: It’s the process in which we use indexes (data structures) for faster data retrieval from tables. It’s like lookup the data for fast retrieval from actual data. E.g., We can create index on the created_at column in the Products table which can speed up queries which involves retrieving products from Products table based on the creation date.
- **Normalization**: It’s the process which minimize the data redundancy in database and improves data integrity. Its goal is to reduce data duplication and anomalies to make data consistent. E.g., Products, Buyers and Sellers are distinct entities with their attributes and each table serves a specific purpose.
- **Denormalization:** In this we intentionally introduce redundancy into normalised database for performance optimization. E.g., We can demoralise seller_name, seller_email and seller_phone from the Seller table into the Products table, we can now directly fetch the relevant seller information along with the product details.

5. How will you handle scaling, if required at any point of time

- By using caching, we can scale the database query. For instance, caching product details, buyer information, or seller information can speed up subsequent access to the same data.
- We can partition the big data tables for example we can portioned Products table based on date range which can help to improve search performance.
- We can use async operations for example if a user places an order, we can queue the order and handle it asynchronously, freeing up database resources for other critical operations.

6. Mention all the assumptions you are taking for solutions.

- Password reset request get expired after some time.
- Buyer first must subscribe the products to get the notification.
- This database is designed for role specific users i.e., buyer and seller.