Test Plan for *vin_online_shopping* Schema

Test Plan ID: TP-VIN-ONLINE-SHOPPING-001

Project Name: VIN Online Shopping Database

Objective:

To validate the correct creation, insertion, deletion, and functioning of tables, foreign key constraints, triggers, and stored procedures in the vin_online_shopping database schema.

Scope:

- Creation and structure validation of tables: customers, categories, products, orders, ordered_items, and payments.
- Data insertion, deletion, and constraint checks.
- Validation of foreign key relationships and auto-increment constraints.
- Trigger functionality tests.
- Stored procedure execution tests.

Testing Strategy:

Manual testing using SQL queries and verifying outputs with expected results.

Automation testing using JAVA, TestNG and verifying outputs with expected results.

Roles and Responsibilities:

- Tester (Vincent Dlamini): Responsible for executing test cases.
- Developer (Vincent Dlamini): Fixes issues identified during testing.

Test Environment:

- MySQL database management system.
- IntelliJ IDEA

Entry Criteria:

- Schema and tables are defined in the SQL database.
- Database connection established.

Exit Criteria:

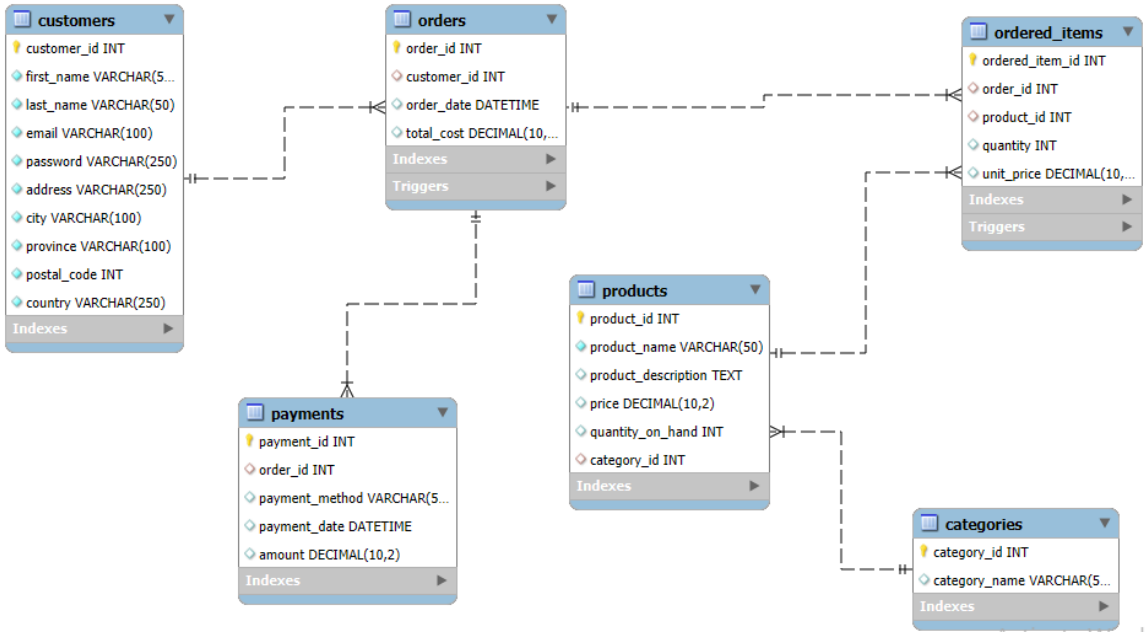
- All test cases are executed, and no critical defects remain.

About vin_online_shopping:

vin_online_shopping is a database that manages an e-commerce platform's customer information, product catalogue, orders and inventory.

The MySQL sample database schema consists of the following tables:

- Customers: Stores customer's data.
- Orders: Stores sales orders placed by customers.
- Ordered Items: Stores data on items ordered.
- Products: Stores data products supplied.
- Categories: Stores data on the categories of products.
- Payment: Stores data on payments made by customers.



Test Scenarios

1. Schema and Table Creation
    - Validate the successful creation of vin_online_shopping schema and tables.

2. Data Insertion
    - Test inserting records into each table (customers, categories, products, etc.) to ensure constraints (like NOT NULL, FOREIGN KEY, and AUTO_INCREMENT) are enforced.

3. Data Deletion
    - Verify deletion operations in tables to confirm that individual and multiple rows can be deleted as expected.

4. Foreign Key Constraints
    - Ensure referential integrity across tables (e.g., orders must link to existing customer_id in customers).

5. Trigger Functionality
    - Validate each trigger's response to events on the orders and ordered_items tables.

6. Stored Procedures Execution
    - Execute stored procedures (e.g., SelectAllCustomers, DeleteAllMultipleTablesData) to confirm they retrieve or delete data as intended.