



# HUST

**ĐẠI HỌC BÁCH KHOA HÀ NỘI**  
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

ONE LOVE. ONE FUTURE.





**ĐẠI HỌC  
BÁCH KHOA HÀ NỘI**  
HANOI UNIVERSITY  
OF SCIENCE AND TECHNOLOGY

# **BÁO CÁO PROJECT DATABASE**

## **Project: Smart ecommerce – HUSTER STORE**

Nhóm – IT3290 – Mã lớp BT: 156784

1. Nguyễn Tài Hưng – 20236034 – Nhóm trưởng
2. Trần Việt Gia Khánh – 20235756
3. Nguyễn Trung Kiên - 20235759

**GVHD: TS. Trần Văn Đăng**

**ONE LOVE. ONE FUTURE.**



# 1. ĐẶT VẤN ĐỀ

Trong bối cảnh thị trường thương mại điện tử ngày càng phát triển mạnh mẽ, đặc biệt là lĩnh vực thời trang và phụ kiện, việc sở hữu một trang web bán giày trực tuyến hiệu quả đã trở thành một yếu tố then chốt để các doanh nghiệp tiếp cận khách hàng tiềm năng và gia tăng doanh số. Tuy nhiên, sự đa dạng về mẫu mã, kích cỡ, nhà cung cấp và thông tin khách hàng đặt ra một thách thức không nhỏ trong việc quản lý dữ liệu một cách khoa học và hệ thống. Chính vì lẽ đó, dự án "Xây dựng cơ sở dữ liệu cho trang web bán giày" được đề xuất nhằm giải quyết bài toán này, tạo nền tảng vững chắc cho việc vận hành và phát triển trang web một cách bền vững.



**HUST**

## 2. PHÂN TÍCH CHỨC NĂNG VÀ NGHIỆP VỤ



## 2. PHÂN TÍCH CHỨC NĂNG VÀ NGHIỆP VỤ

### Phân tích nghiệp vụ

#### a. Quản lý sản phẩm

- Nhập giày từ các thương hiệu, phân loại theo danh mục.
- Mỗi sản phẩm có nhiều biến thể (kích cỡ, màu sắc).
- Nhân viên có thể thêm/sửa sản phẩm: tên, giá, mô tả...
- Hệ thống theo dõi tồn kho từng biến thể.
- Chỉ bán hàng khi còn tồn kho.

#### b. Quản lý khách hàng

- Thêm, sửa, xóa thông tin khách hàng.
- Hỗ trợ cấp lại mật khẩu, chỉnh sửa thông tin.

## 2. PHÂN TÍCH CHỨC NĂNG VÀ NGHIỆP VỤ

### Phân tích nghiệp vụ

#### c. Quản lý doanh thu, lợi nhuận

- Thống kê doanh thu theo ngày/tuần/tháng/quý/năm.
- Ghi nhận doanh thu từng đơn hàng (trước & sau khi áp dụng voucher).
- Cập nhật tiền thanh toán theo thời gian thực.

#### d. Quá trình bán hàng

- Khách tìm kiếm và duyệt sản phẩm.
- Chọn biến thể (kích cỡ, màu) và thêm vào giỏ hàng.
- Đặt hàng: nhập thông tin giao hàng, chọn thanh toán.
- Hệ thống tạo đơn hàng, trừ tồn kho tương ứng.
- Khách hàng thanh toán qua nhiều phương thức.



## 2. PHÂN TÍCH CHỨC NĂNG VÀ NGHIỆP VỤ

### Phân tích nghiệp vụ

#### e. Xử lý & thực hiện đơn hàng

- Nhân viên tiếp nhận và xác nhận đơn hàng.
- Kiểm tra thanh toán, chuẩn bị hàng (lấy từ kho, đóng gói).
- Tạo đơn vận chuyển, cập nhật mã theo dõi.
- Cập nhật trạng thái đơn hàng:
  - Đang xử lý → Đã đóng gói → Đang vận chuyển → Đã giao.

## 2. PHÂN TÍCH CHỨC NĂNG VÀ NGHIỆP VỤ

### Mô tả chức năng

Hệ thống được thiết kế với ba nhóm người dùng chính, mỗi nhóm có những vai trò và quyền hạn riêng biệt để đảm bảo hoạt động hiệu quả và an toàn.

- **Khách hàng:** Người mua sắm và sử dụng các tính năng chính của trang web.
- **Nhân viên:** Người vận hành, hỗ trợ quản lý đơn hàng và sản phẩm.
- **Quản lý:** Người có quyền cao nhất, giám sát toàn bộ hoạt động của hệ thống.

## 2. PHÂN TÍCH CHỨC NĂNG VÀ NGHIỆP VỤ

### Mô tả chức năng

#### Luồng Chức Năng - Khách Hàng

Đây là nhóm người dùng cốt lõi, trải nghiệm mua sắm được tối ưu hóa qua các chức năng sau:

- **Tìm kiếm & Khám phá:**
  - Tìm kiếm sản phẩm theo từ khóa.
  - Lọc sản phẩm (giá, kích cỡ, màu sắc, thương hiệu).
  - Sắp xếp kết quả (giá tăng/giảm, mới nhất, bán chạy).
- **Xem & Đánh giá Sản phẩm:**
  - Xem thông tin chi tiết (hình ảnh, mô tả, giá, chất liệu...).
  - Đọc review, xem rating từ người mua khác.
  - Xem các sản phẩm liên quan được gợi ý.
- **Giỏ hàng & Đặt hàng:**
  - Thêm/Sửa/Xóa sản phẩm trong giỏ hàng.
  - Cung cấp thông tin giao hàng.
  - Chọn phương thức thanh toán và xác nhận đơn hàng.
- **Quản lý Tài khoản & Đơn hàng:**
  - Đăng ký, đăng nhập, quản lý thông tin cá nhân.
  - Theo dõi lịch sử và trạng thái đơn hàng.
  - Hủy đơn hàng (khi được phép).
  - Viết đánh giá và xếp hạng cho sản phẩm đã mua.

## 2. PHÂN TÍCH CHỨC NĂNG VÀ NGHIỆP VỤ

### Mô tả chức năng

#### Luồng Chức Năng - Nhân Viên

Nhân viên là cầu nối giữa khách hàng và hệ thống, đảm bảo quy trình vận hành trôi chảy.

- **Quản lý Sản phẩm (Quyền hạn giới hạn):**
  - Xem danh sách, tìm kiếm, lọc sản phẩm.
  - Cập nhật thông tin cơ bản (số lượng tồn kho, hình ảnh).
- **Quản lý Đơn hàng:**
  - Xem và xử lý danh sách đơn hàng mới.
  - Xác nhận hoặc từ chối đơn hàng.
  - Cập nhật trạng thái (đang xử lý, đang giao, đã giao).
  - In hóa đơn.
- **Quản lý Khách hàng:**
  - Xem danh sách và thông tin chi tiết khách hàng.
  - Tra cứu lịch sử mua hàng để hỗ trợ.
- **Quản lý Tài khoản:**
  - Đăng nhập/Đăng xuất.
  - Xem và cập nhật thông tin cá nhân.

# 2. PHÂN TÍCH CHỨC NĂNG VÀ NGHIỆP VỤ

## Mô tả chức năng

### Luồng Chức Năng - Quản Lý

Quản lý có toàn quyền truy cập để giám sát, điều hành và đưa ra quyết định chiến lược.

- **Quản lý Sản phẩm (Toàn quyền):**
  - Thêm sản phẩm mới.
  - Cập nhật toàn bộ thông tin sản phẩm.
  - Xóa sản phẩm khỏi hệ thống.
- **Quản lý Kho hàng:**
  - Quản lý và điều chỉnh số lượng tồn kho.
  - Xem báo cáo tồn kho chi tiết.
- **Quản lý Đơn hàng (Toàn quyền):**
  - Xem và quản lý toàn bộ đơn hàng.
  - Có quyền hủy đơn hàng khi cần thiết.
- **Quản lý Người dùng:**
  - Tạo, sửa, xóa tài khoản nhân viên.
  - Phân quyền truy cập cho từng vai trò nhân viên.
- **Báo cáo & Thống kê:**
  - Xem báo cáo doanh thu (theo ngày, sản phẩm, khách hàng).
  - Thống kê sản phẩm bán chạy, sản phẩm tồn kho.

**HUST**

### 3. THIẾT KẾ CƠ SỞ DỮ LIỆU



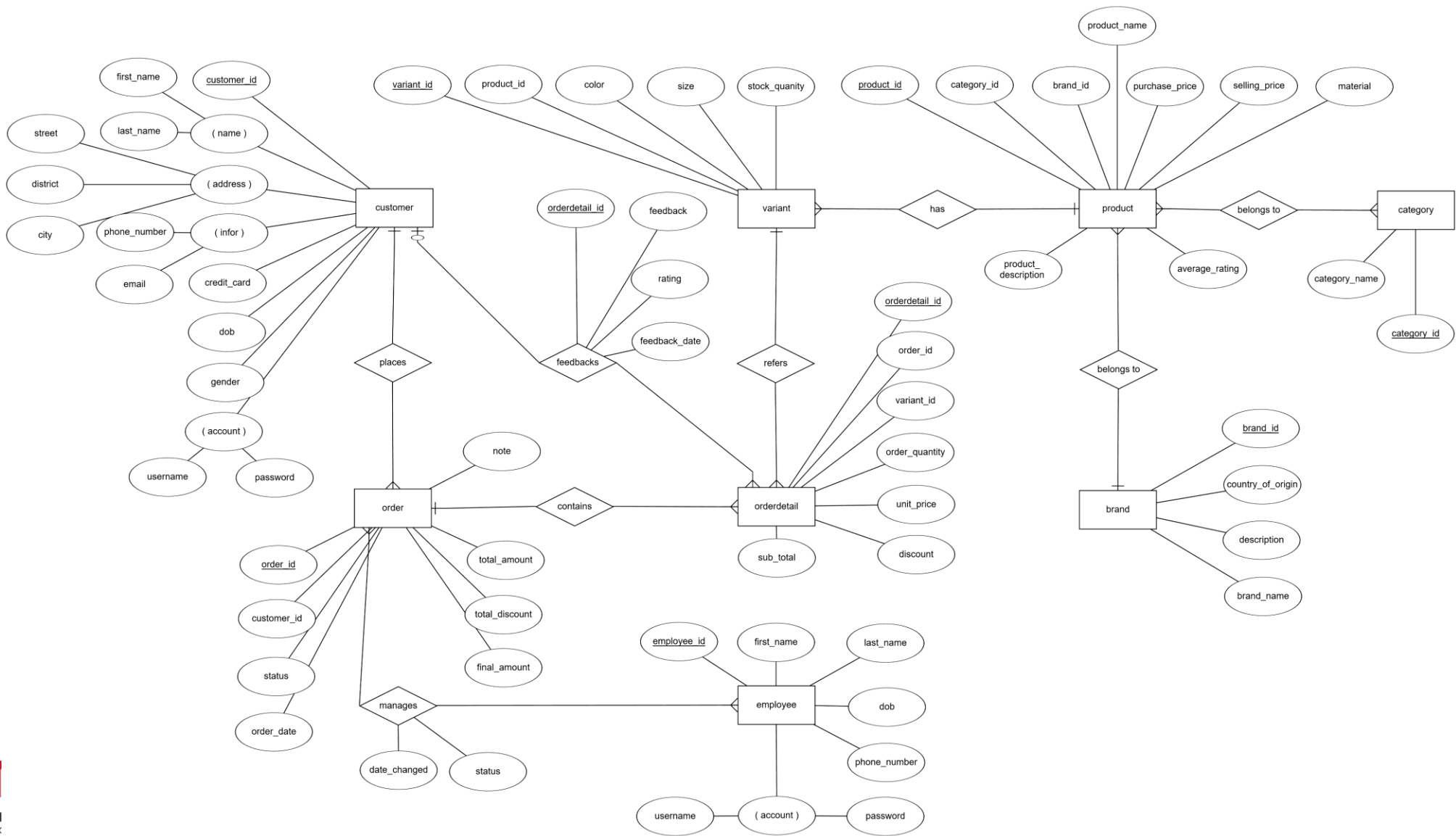
# 3. THIẾT KẾ CƠ SỞ DỮ LIỆU

## Xác định các thực thể

- Sản phẩm (product)
- Các biến thể của sản phẩm (variant)
- Phân loại (category)
- Nhãn hàng (brand)
- Đơn hàng (order)
- Chi tiết đơn hàng (orderdetail)
- Khách hàng (customer)
- Nhân viên (employee)

# 3. THIẾT KẾ CƠ SỞ DỮ LIỆU

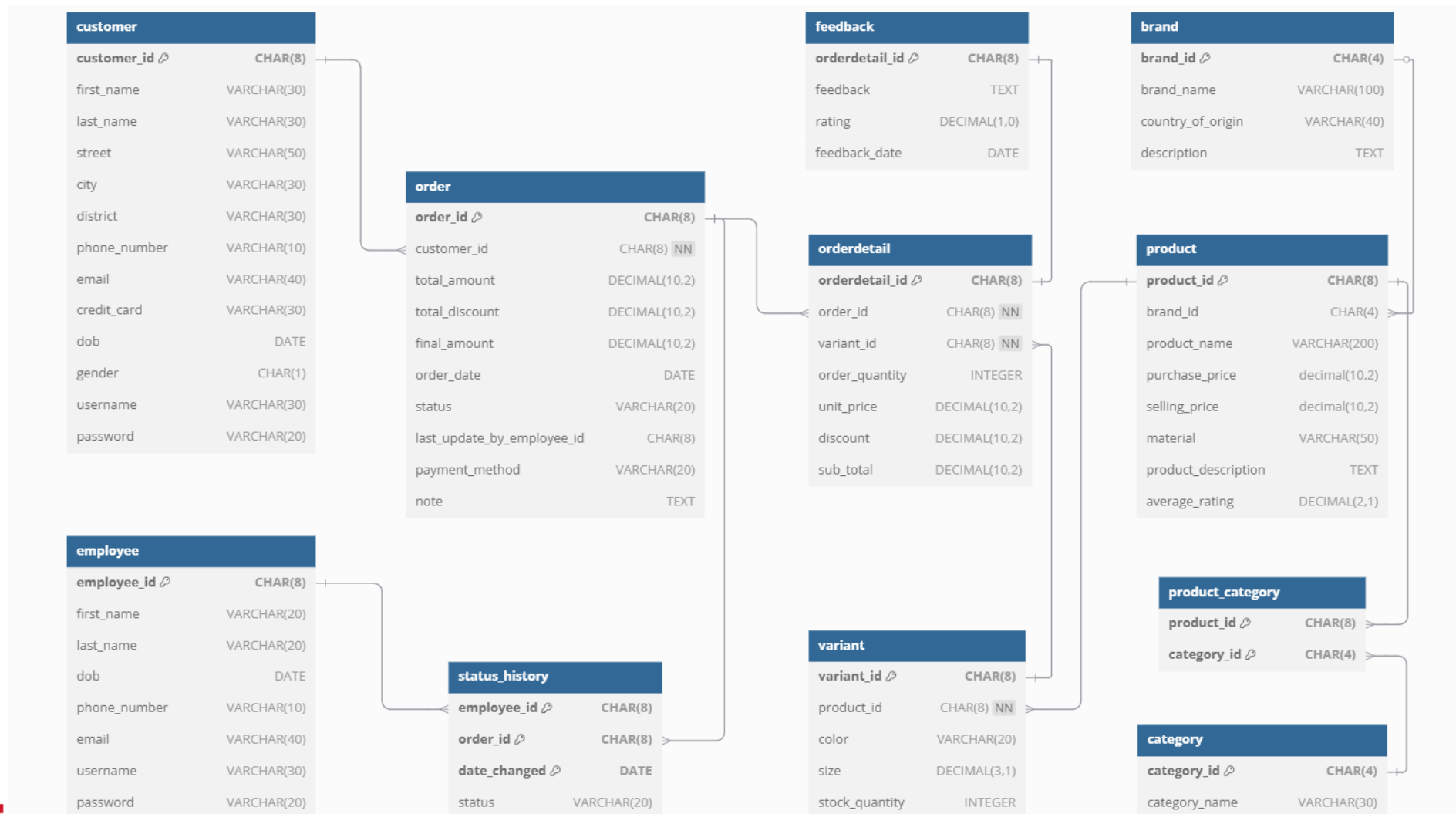
## Sơ đồ thực thể liên kết (ERD)





# 3. THIẾT KẾ CƠ SỞ DỮ LIỆU

## Sơ đồ Quan hệ (Relational Schema)



# 3. THIẾT KẾ CƠ SỞ DỮ LIỆU

## Tạo bảng (Create Table)

```
CREATE TABLE customer (  
    customer_id CHAR(8) PRIMARY KEY,  
    first_name VARCHAR(30) NOT NULL,  
    last_name VARCHAR(30),  
    street VARCHAR(50),  
    city VARCHAR(30),  
    district VARCHAR(30),  
    phone_number VARCHAR(10) UNIQUE NOT NULL,  
    email VARCHAR(40) UNIQUE,  
    credit_card VARCHAR(30),  
    dob DATE,  
    gender CHAR(1) CHECK (gender IN ('F', 'M')),  
    username VARCHAR(30) UNIQUE NOT NULL,  
    password VARCHAR(20) NOT NULL  
);
```

```
CREATE TABLE employee (  
    employee_id CHAR(8) PRIMARY KEY,  
    first_name VARCHAR(20),  
    last_name VARCHAR(20),  
    dob DATE,  
    phone_number CHAR(10),  
    email VARCHAR(40),  
    username VARCHAR(30) UNIQUE NOT NULL,  
    password VARCHAR(20) NOT NULL  
);
```

# 3. THIẾT KẾ CƠ SỞ DỮ LIỆU

## Tạo bảng (Create Table)

```
CREATE TABLE category (  
    category_id CHAR(4) PRIMARY KEY,  
    category_name VARCHAR(30) NOT NULL  
);
```

```
CREATE TABLE brand (  
    brand_id CHAR(4) PRIMARY KEY,  
    brand_name VARCHAR(100) NOT NULL,  
    country_of_origin VARCHAR(40),  
    brand_description TEXT  
);
```

# 3. THIẾT KẾ CƠ SỞ DỮ LIỆU

## Tạo bảng (Create Table)

```
CREATE TABLE product (  
    product_id CHAR(8) PRIMARY KEY,  
    brand_id CHAR(4),  
    product_name VARCHAR(200) NOT NULL,  
    purchase_price DECIMAL(10,2) NOT NULL,  
    selling_price DECIMAL(10,2) NOT NULL,  
    material VARCHAR(50),  
    product_description TEXT,  
    average_rating DECIMAL(2,1) DEFAULT 0.0,  
    CONSTRAINT fk_brand_id FOREIGN KEY (brand_id) REFERENCES brand(brand_id)  
);  
  
CREATE TABLE product_category (  
    product_id CHAR(8) NOT NULL,  
    category_id CHAR(4) NOT NULL,  
    CONSTRAINT pk_product_category PRIMARY KEY (product_id, category_id),  
    CONSTRAINT fk_product_id FOREIGN KEY (product_id) REFERENCES product(product_id),  
    CONSTRAINT fk_category_id FOREIGN KEY (category_id) REFERENCES category(category_id)  
);
```

# 3. THIẾT KẾ CƠ SỞ DỮ LIỆU

## Tạo bảng (Create Table)

```
CREATE TABLE variant (  
    variant_id CHAR(8) PRIMARY KEY,  
    product_id CHAR(8) NOT NULL,  
    color VARCHAR(20) NOT NULL,  
    size DECIMAL(3,1) NOT NULL,  
    stock_quantity INTEGER NOT NULL DEFAULT 0 CHECK (stock_quantity >= 0),  
    CONSTRAINT fk_product_id FOREIGN KEY (product_id) REFERENCES product(product_id)  
);
```

# 3. THIẾT KẾ CƠ SỞ DỮ LIỆU

## Tạo bảng (Create Table)

```
CREATE TABLE "order" (  
    order_id CHAR(8) PRIMARY KEY,  
    customer_id CHAR(8),  
    total_amount DECIMAL(10,2) DEFAULT 0.0,  
    total_discount DECIMAL(10,2) DEFAULT 0.0,  
    final_amount DECIMAL(10,2) DEFAULT 0.0,  
    order_date DATE DEFAULT CURRENT_DATE,  
    status VARCHAR(20) CHECK  
        (status IN ('PENDING', 'PACKAGING', 'ON DELIVERY', 'DELIVERED', 'CANCELLED')) DEFAULT 'PENDING',  
    last_updated_by_employee_id CHAR(8),  
    payment_method VARCHAR(20),  
    note TEXT,  
    CONSTRAINT fk_customer_id FOREIGN KEY (customer_id) REFERENCES customer(customer_id),  
    CONSTRAINT fk_employee_id FOREIGN KEY (last_updated_by_employee_id) REFERENCES employee(employee_id)  
);
```

# 3. THIẾT KẾ CƠ SỞ DỮ LIỆU

## Tạo bảng (Create Table)

```
CREATE TABLE orderdetail (  
    orderdetail_id CHAR(8) PRIMARY KEY,  
    order_id CHAR(8) NOT NULL,  
    variant_id CHAR(8) NOT NULL,  
    order_quantity INTEGER NOT NULL,  
    unit_price DECIMAL(10,2) DEFAULT 0.0,  
    discount INTEGER DEFAULT 0,  
    sub_total DECIMAL(10,2) DEFAULT 0.0,  
    CONSTRAINT fk_order_id FOREIGN KEY (order_id) REFERENCES "order"(order_id),  
    CONSTRAINT fk_variant_id FOREIGN KEY (variant_id) REFERENCES variant(variant_id)  
);
```

```
CREATE TABLE feedback (  
    orderdetail_id CHAR(8) PRIMARY KEY,  
    feedback TEXT,  
    rating DECIMAL(1,0),  
    feedback_date DATE,  
    CONSTRAINT fk_orderdetail_id FOREIGN KEY (orderdetail_id) REFERENCES orderdetail(orderdetail_id)  
);
```

# 3. THIẾT KẾ CƠ SỞ DỮ LIỆU

## Tạo bảng (Create Table)

```
CREATE TABLE status_history (  
    employee_id CHAR(8) NOT NULL,  
    order_id CHAR(8) NOT NULL,  
    date_changed DATE NOT NULL DEFAULT CURRENT_DATE,  
    status VARCHAR(20) CHECK  
        (status IN ('PENDING', 'PACKAGING', 'ON DELIVERY', 'DELIVERED', 'CANCELLED')) DEFAULT 'PENDING',  
    CONSTRAINT pk_status_history PRIMARY KEY (employee_id, order_id, date_changed),  
    CONSTRAINT fk_employee_id FOREIGN KEY (employee_id) REFERENCES employee(employee_id),  
    CONSTRAINT fk_order_id FOREIGN KEY (order_id) REFERENCES "order"(order_id)  
);
```



# HUST

## 4. INDEX, TRIGGER



## 4.1 INDEX

Cú pháp:

```
CREATE INDEX idx_order_status_date ON "order"(status, order_date ASC);  
CREATE INDEX idx_order_customer_date ON "order"(customer_id, order_date DESC);  
CREATE INDEX idx_variant_product_stock ON variant(product_id, stock_quantity);  
CREATE INDEX idx_variant_size ON variant(size);
```

## 4.2 TRIGGER

Cú pháp:

```
CREATE TRIGGER trg_set_orderdetail_prices  
BEFORE INSERT ON orderdetail  
FOR EACH ROW  
EXECUTE FUNCTION calculate_orderdetail_prices_trigger_function();
```

**Chức năng:** Tự động cập nhật các thuộc tính của orderdetail khi tạo orderdetail mới.

## 4.2 TRIGGER

Cú pháp:

```
CREATE TRIGGER trg_after_orderdetail_change  
AFTER INSERT OR UPDATE OR DELETE ON orderdetail  
FOR EACH ROW  
EXECUTE FUNCTION update_order_totals_function();
```

Chức năng: Tự động cập nhật lại order khi có các orderdetail.

## 4.2 TRIGGER

Cú pháp:

```
CREATE TRIGGER trg_log_order_status_change  
AFTER UPDATE ON "order"  
FOR EACH ROW  
EXECUTE FUNCTION log_order_status_change();
```

Chức năng: Tự động lưu lại các lịch sử chỉnh status của order.

## 4.2 TRIGGER

Cú pháp:

```
CREATE TRIGGER trg_before_orderdetail_change_update_stock  
BEFORE INSERT OR UPDATE OR DELETE ON orderdetail  
FOR EACH ROW  
EXECUTE FUNCTION update_stock_quantity_function();
```

**Chức năng:** Tự động cập nhật số lượng tồn kho khi có thay đổi trong chi tiết đơn hàng

## 4.2 TRIGGER

**Cú pháp:**

```
CREATE TRIGGER trg_after_feedback_change  
AFTER INSERT OR UPDATE OR DELETE ON feedback  
FOR EACH ROW  
EXECUTE FUNCTION update_product_average_rating_function();
```

**Chức năng:** Tự động cập nhật điểm đánh giá trung bình của sản phẩm .

**HUST**

## 5. VIẾT TRUY VẤN CÁ NHÂN





A large graphic on the left side of the slide. It features a dark blue background with a circular pattern of red dots of varying sizes, creating a sense of depth and movement. The word "HUST" is centered within this graphic in a bold, white, sans-serif font.

# HUST

# THANK YOU !