

9.1. Xây dựng ứng dụng Quản lý cửa hàng Phương Perfume

✓ Mô tả ứng dụng

- Cửa hàng nước hoa Phương Perfume chuyên kinh doanh các mặt hàng nước hoa trực tiếp tại cửa hàng và online.
- Hiện tại cửa hàng có các bộ phận với các công việc chính như sau:
 - Nhân viên bán hàng: giới thiệu sản phẩm, chăm sóc khách hàng, bán hàng và giải quyết các đơn hàng của khách.
 - Nhân viên kế toán: quản lý đơn hàng và thống kê
- Cửa hàng nước hoa yêu cầu xây dựng website giúp cửa hàng quảng bá và bán sản phẩm, hỗ trợ khách hàng có thể xem và đặt hàng trực tiếp trên mạng và xây dựng một ứng dụng windows forms giúp cửa hàng quản lý thông tin sản phẩm, tạo quảng cáo, đơn hàng, thống kê, quản lý thông tin khách hàng...
- Ứng dụng được xây dựng có thể hỗ trợ nhân viên trong cửa hàng thực hiện tốt các công việc của mình và giúp cửa hàng có thể thu hút nhiều khách đến tham quan và mua hàng.

⇒ **Xây dựng windows forms quản lý cửa hàng nước hoa (làm việc ở module này)**

⇒ Xây dựng website giới thiệu, quảng bá sản phẩm và bán hàng online (làm việc ở module sau)

✓ **Thiết kế CSDL cho ứng dụng (tập tin `phuong_perfume.sql` được đính kèm): gồm có các bảng sau:**

- Bảng loai (loại)

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	<u>id</u>	int(11)			No	None	AUTO_INCREMENT
2	tenloai	varchar(200)	utf8_general_ci		No	None	

- Bảng thuonghieu (thương hiệu)

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	<u>id</u>	int(11)			No	None	AUTO_INCREMENT
2	tenthuonghieu	varchar(300)	utf8_general_ci		No	None	
3	hinhanh	varchar(300)	utf8_general_ci		No	None	

- Bảng sanpham (sản phẩm)

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	<u>id</u>	int(11)			No	None	AUTO_INCREMENT
2	tensanpham	varchar(300)	utf8_general_ci		No	None	
3	mota	text	utf8_general_ci		Yes	NULL	
4	hinhanh	varchar(300)	utf8_general_ci		No	None	
5	dongia	double			No	None	
6	dongiaKM	double			Yes	NULL	
7	soluong	int(11)			No	None	
8	ngaytao	date			No	None	
9	hienthi	int(11)			No	1	
10	id_loai	int(11)			No	None	
11	id_thuonghieu	int(11)			No	None	

– Bảng vai tro (vai trò)

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	<u>id</u>	int(11)			No	None	AUTO_INCREMENT
2	tenvaitro	varchar(100)	utf8_general_ci		No	None	
3	mota	text	utf8_general_ci		Yes	NULL	

– Bảng nguoidung (người dùng)

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	<u>id</u>	int(11)			No	None	AUTO_INCREMENT
2	email	varchar(100)	utf8_general_ci		No	None	
3	password	varchar(100)	utf8_general_ci		No	None	
4	hoten	varchar(200)	utf8_general_ci		No	None	
5	diachi	varchar(300)	utf8_general_ci		No	None	
6	dtdd	varchar(20)	utf8_general_ci		No	None	
7	id_vaitro	int(11)			No	1	

– Bảng trangthaidonhang (trạng thái đơn hàng)

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	<u>id</u>	int(11)			No	None	AUTO_INCREMENT
2	tentrangthai	varchar(100)	utf8_general_ci		No	None	

– Bảng donhang (đơn hàng)

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	<u>id</u>	int(11)			No	None	AUTO_INCREMENT
2	id_khachhang	int(11)			No	None	
3	ngaydathang	date			No	None	
4	tennguoinhanhang	varchar(200)	utf8_general_ci		No	None	
5	dienthoainguinhan	varchar(20)	utf8_general_ci		No	None	
6	diachigiaohang	varchar(300)	utf8_general_ci		No	None	
7	ghichu	text	utf8_general_ci		Yes	NULL	
8	thanhtoan	tinyint(4)			No	0	
9	id_trangthai	int(11)			No	1	

– Bảng chitietdonhang (chi tiết đơn hàng)

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	<u>id</u>	int(11)			No	None	AUTO_INCREMENT
2	id_donhang	int(11)			No	None	
3	id_sanpham	int(11)			No	None	
4	soluong	int(11)			No	None	

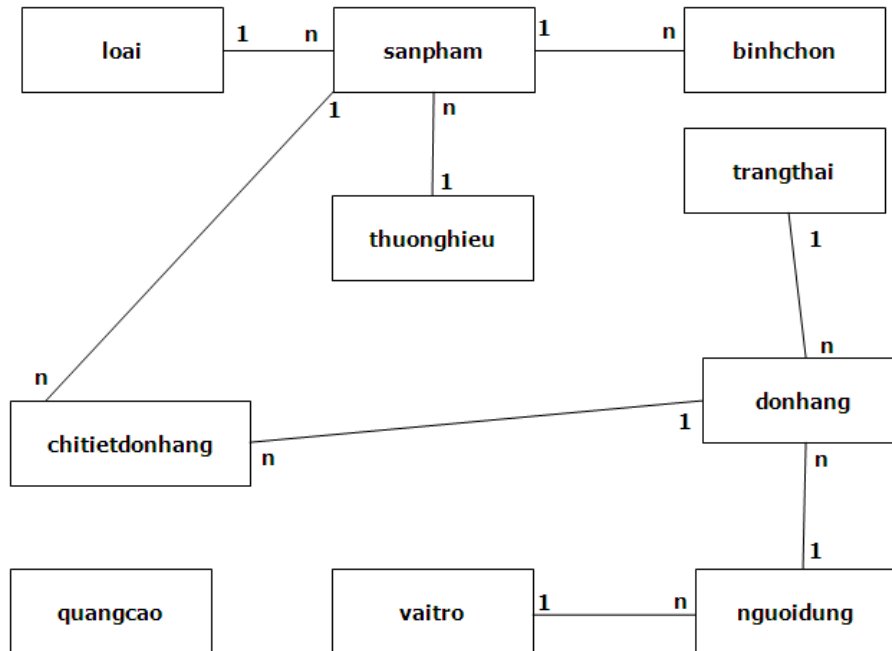
– Bảng quangcao (quảng cáo)

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	<u>id</u>	int(11)			No	None	AUTO_INCREMENT
2	hinhanh	varchar(200)	utf8_general_ci		No	None	
3	thongdiep	varchar(300)	utf8_general_ci		No	None	
4	thongtinchitiet	text	utf8_general_ci		Yes	NULL	
5	ngaydang	date			Yes	NULL	

– Bảng binhchon (bình chọn)

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	<u>id</u>	int(11)			No	None	AUTO_INCREMENT
2	ten	varchar(200)	utf8_general_ci		No	None	
3	email	varchar(200)	utf8_general_ci		No	None	
4	diem	int(11)			No	None	
5	id_sanpham	int(11)			No	None	

Mối quan hệ giữa các bảng:

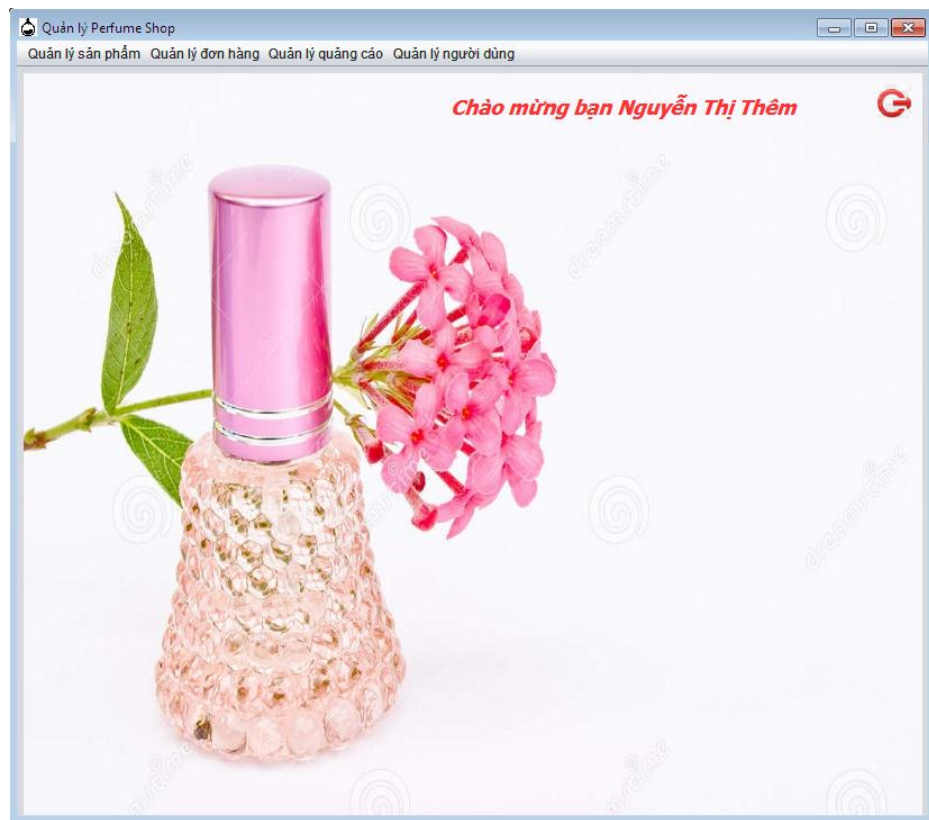


Lưu ý:

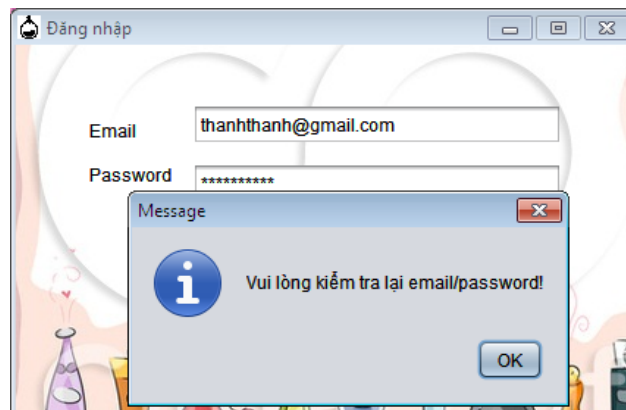
- Học viên được cung cấp CSDL **phuong_perfume.sql** => Import CSDL phuong_perfume.sql vào MySQL
- Thư mục **img** (chứa các hình ảnh cần thiết cho ứng dụng) => Học viên tạo Project và đưa vào project để sử dụng
- Trong project, ở thư mục src, học viên tạo package model, trong package model, tạo tất cả các lớp cần thiết tương ứng với các bảng của CSDL.
- Trong model, xây dựng lớp QLCSL, trong lớp này có phương thức để kết nối đến CSDL phuong_perfume

1. Yêu cầu 1: Đăng nhập vào hệ thống

Form Đăng nhập



Đăng nhập thành công => Vào Form chính của ứng dụng



Đăng nhập không thành công => Hiện thị thông báo

✓ **Hướng dẫn sử dụng:**

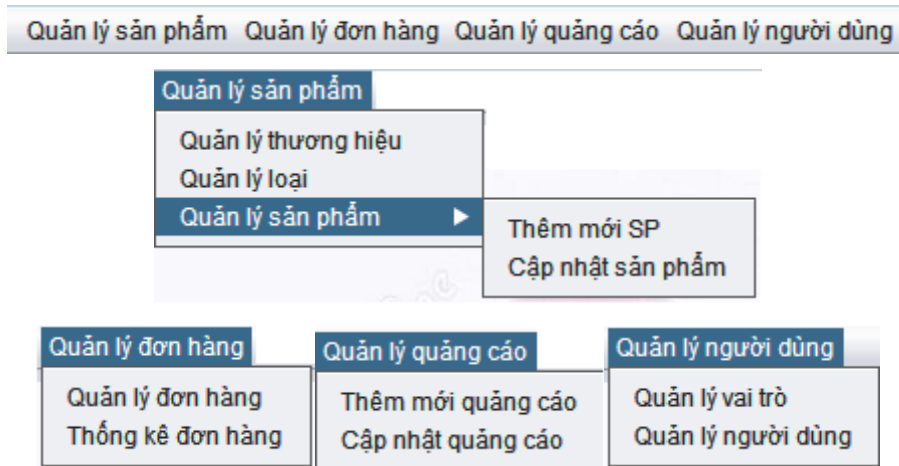
- Nhập email, password => nhấn "Đăng nhập" => hệ thống sẽ kiểm tra thông tin => nếu có tài khoản này và vai trò không phải là khách hàng thì đăng nhập thành công và chuyển vào màn hình làm việc chính của ứng dụng. Ngược lại, đăng nhập thất bại, hiển thị thông báo yêu cầu người dùng kiểm tra lại thông tin.

✓ **Tóm tắt yêu cầu**

▪ **Thiết kế giao diện người dùng:**

- Học viên thiết kế giao diện form đăng nhập phía trên

- Học viên thiết kế giao diện form chính như trên, trên form chính có menu chứa các công việc chính của ứng dụng



- **Nhập:**

- Email
- Password

- **Xuất:**

- Kết quả đăng nhập (thành công/ không thành công)

✓ **Hướng dẫn**

- Trong lớp QLCSDL, xây dựng phương thức phù hợp cho việc đăng nhập.
- Gợi ý:

```
public NgườiDung dangnhapNgườiDung(String email, String password) throws
SQLException, ClassNotFoundException {
    NgườiDung nd = null;
    try (Connection conn = this.connect()) {
        java.sql.Statement statement = conn.createStatement();
        String sql = "SELECT * FROM nguoidung WHERE email like '" + email + "' and
password like '" + password + "'";
        ResultSet resultSet = statement.executeQuery(sql);
        while (resultSet.next()) {
            nd = new NgườiDung();
            nd.setId(resultSet.getInt("id"));
            nd.setEmail(resultSet.getString("email"));
            nd.setPassword(resultSet.getString("password"));
            nd.setHoten(resultSet.getString("hoten"));
            nd.setDiachi(resultSet.getString("diachi"));
        }
    }
}
```

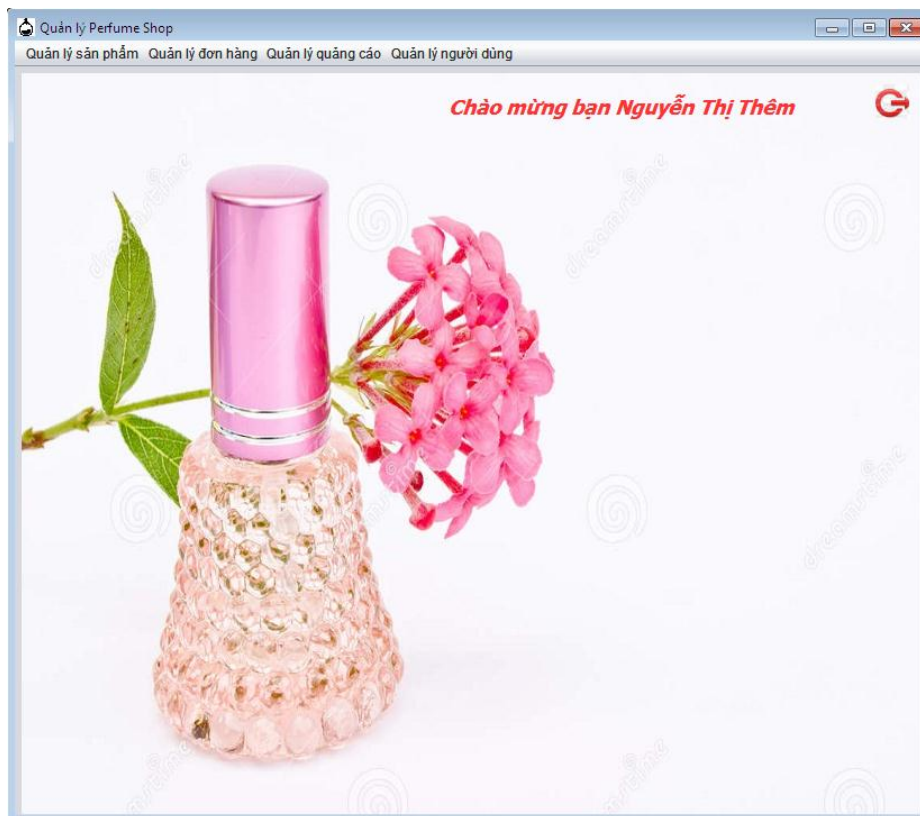
```

        nd.setDtdd(resultSet.getString("dtdd"));
        nd.setId_vaitro(resultSet.getInt("id_vaitro"));
    }
}
return nd;
}


```

- Xử lý việc đăng nhập trên form

2. Yêu cầu 2: Đăng xuất khỏi hệ thống



✓ Hướng dẫn sử dụng:

- Nhấn vào icon  đăng xuất ngay sau câu chào => Hệ thống sẽ đóng form chính quay trở về form Đăng nhập

✓ Hướng dẫn

- Xử lý đóng form chính và mở form đăng nhập

3. Yêu cầu 3: Thêm sản phẩm mới

Quản lý Perfume Shop

Quản lý sản phẩm Quản lý đơn hàng Quản lý quảng cáo Quản lý người dùng

Thêm sản phẩm mới

Tên sản phẩm

Mô tả

Hình

Đơn giá

Đơn giá KM

Số lượng

Loại

Thương hiệu

From "Thêm sản phẩm mới" sẽ được mở ra khi người dùng chọn chức năng này trên menu

Thêm sản phẩm mới

Tên sản phẩm: Dolce Vita EDT 100ML

Mô tả: Dolce Vita trong tiếng Ý có nghĩa là "cuộc sống thật ngọt ngào". Đúng như tên gọi của mình, loại nước hoa này mang hương hoa thơm ngát, đầy vui tươi và hạnh phúc. Là sự kết hợp giữa hương hoa với đào và mơ, Dolce Vita giúp bạn dễ dàng lựa chọn trang phục phù hợp với nó. Với mùi hương cuối là vani và gỗ đàn hương, nó sẽ khiến bạn luôn cảm thấy nhẹ nhàng.

Hình: C:\Users\ktphuong\Desktop\Dior-Dolce-Vita.png

Đơn giá: 2113000


Đơn giá KM: 1901000

Số lượng: 10

Loại: Nước hoa nữ

Thương hiệu: Dior

Thêm mới Tiếp tục



Thêm sản phẩm mới

✓ **Hướng dẫn sử dụng:**

- Nhập các thông tin: tên sản phẩm, mô tả, hình, đơn giá, đơn giá khuyến mãi, số lượng, chọn loại và thương hiệu => nhấn "Thêm mới" để thêm sản phẩm.
- Nhấn "Tiếp tục" để xóa các thông tin đang có cho người dùng tiếp tục thêm mới sản phẩm

✓ **Tóm tắt yêu cầu**

▪ **Thiết kế giao diện người dùng:**

- Học viên thiết kế giao diện Thêm sản phẩm mới như hình phía trên, trên đó có 2 combobox có sẵn loại và thương hiệu cho người dùng chọn.
- Form này là form nằm trong form chính, được mở ra khi người dùng chọn chức năng "Thêm sản phẩm mới" trên menu.

▪ **Nhập:**

- Tên sản phẩm, mô tả, hình, đơn giá, đơn giá khuyến mãi, số lượng, chọn loại và thương hiệu

▪ **Xuất:**

- Kết quả thêm thành công/ không thành công

✓ **Hướng dẫn**

- Trong lớp QLCSDL, xây dựng thêm phương thức phù hợp cho việc hiển thị danh sách loại và thương hiệu, thêm sản phẩm.
- Gợi ý:

```
// danh sách loại
public List<Loai> dsLoai() throws SQLException, ClassNotFoundException {
    List<Loai> rs;
    try (Connection conn = this.connect()) {
        rs = new ArrayList<>();
        java.sql.Statement statement = conn.createStatement();
        String sql = "SELECT * FROM loai";
        ResultSet resultSet = statement.executeQuery(sql);
        while (resultSet.next()) {
            Loai loai = new Loai();
            loai.setId(resultSet.getInt("id"));
            loai.setTenloai(resultSet.getString("tenloai"));
            rs.add(loai);
        }
    }
    return rs;
}

// danh sách thương hiệu
public List<ThuongHieu> dsThuongHieu() throws SQLException,
    ClassNotFoundException {
    List<ThuongHieu> rs;
    try (Connection conn = this.connect()) {
        rs = new ArrayList<>();
        java.sql.Statement statement = conn.createStatement();
        String sql = "SELECT * FROM thuonghieu";
        ResultSet resultSet = statement.executeQuery(sql);
        while (resultSet.next()) {
            ThuongHieu th = new ThuongHieu();
            th.setId(resultSet.getInt("id"));
            th.setTenthuonghieu(resultSet.getString("tenthuonghieu"));
            th.setHinhanh(resultSet.getString("hinhanh"));
            rs.add(th);
        }
    }
}
```

```

    }
    return rs;
}

// thêm mới sản phẩm
public boolean themSanPham(SanPham sanpham) throws SQLException,
ClassNotFoundException, UnsupportedEncodingException {
    boolean execute = false;
    SimpleDateFormat df = new SimpleDateFormat("yyyy-MM-dd");
    try (Connection conn = this.connect()) {
        String sql = "INSERT INTO sanpham VALUES(NULL,'" + sanpham.tensanpham + "'
        ,'" + sanpham.mota + "' ,'" + sanpham.hinhanh + "' ,'" + sanpham.dongia + "' ,'" +
        sanpham.dongiaKM + "' ,'" + sanpham.soluong + "' ,'" + df.format(sanpham.ngaytao) + "' ,
        '" + 1 + "' ,'" + sanpham.id_loai + "' ,'" + sanpham.id_thuonghieu + "')";

        PreparedStatement statement = conn.prepareStatement(sql);
        execute = statement.execute();
    }
    return execute;
}

```

- Xử lý việc đưa danh sách loại/ thương hiệu vào combo box
- Xử lý việc chọn và hiển thị hình ảnh
- Xử lý việc đưa hình ảnh vào thư mục img chung của ứng dụng
- Xử lý việc thêm sản phẩm vào CSDL

4. Yêu cầu 4: Quản lý đơn hàng

Quản lý Perfume Shop

Quản lý sản phẩm Quản lý đơn hàng Quản lý quảng cáo Quản lý người dùng

Quản lý đơn hàng

Tìm kiếm đơn hàng Chi tiết đơn hàng

Tìm kiếm theo Tên khách hàng Tìm

Tên khách hàng thanh

Ngày đặt hàng 22/01/2016

Trạng thái đơn hàng Nhận đơn hàng

Mã ĐH	Tên khách hàng	Ngày đặt	Tính trạng	Thanh toán
14	Nguyễn Ngọc Thanh	2016-01-20	Đang giao hàng	X
15	Nguyễn Ngọc Thanh	2016-01-22	Nhận đơn hàng	

Xem chi tiết đơn hàng đang chọn

From "Quản lý đơn hàng" sẽ được mở ra khi người dùng chọn chức năng này trên menu. Tab hiển thị là tab "Tìm kiếm đơn hàng"

Quản lý đơn hàng

Tìm kiếm đơn hàng Chi tiết đơn hàng

Mã đơn hàng 15 Ngày đặt hàng 22/01/2016

Tên khách hàng Nguyễn Ngọc Thanh

Thông tin giao hàng **Tình hình đơn hàng**

Tên Nguyễn Ngọc Thanh Trạng thái Nhận đơn hàng

Địa chỉ 227 Nguyễn Văn Cừ Quận 5 ☐ Đã thanh toán

Điện thoại 0903753951

Cập nhật

ID	Tên sản phẩm	Số lượng	Đơn giá	Thành tiền
26	Chanel Coco Mademoiselle EDP 100ml	1	3,299,000	3,299,000
27	Lancôme Tresor Midnight Rose	2	1,890,000	3,780,000
28	DOLCE & GABBANA INTENSO POUR HOMME	1	1,854,000	1,854,000

Xóa chi tiết đơn hàng đang chọn

Tab "Chi tiết đơn hàng"

✓ Hướng dẫn sử dụng:

- Chọn tiêu chí tìm kiếm theo Tên khách hàng/ ngày đặt hàng/ trạng thái đơn hàng sau đó điền và chọn thông tin thích hợp > Nhấn "Tìm" > Danh sách đơn hàng phù hợp với thông tin tìm kiếm sẽ hiển thị trên table.
- Chọn một đơn hàng trên table sau đó chọn tab "Chi tiết đơn hàng" hoặc nhấn nút "Xem chi tiết đơn hàng đang chọn" > hiển thị thông tin của đơn hàng kèm theo chi tiết đơn hàng.
- Cập nhật thông tin giao hàng và chọn tình trạng giao hàng > nhấn "Cập nhật" > Thông tin đơn hàng sẽ được cập nhật.
- Chọn một chi tiết đơn hàng > Nhấn "Xóa chi tiết đơn hàng đang chọn" > Chi tiết đơn hàng sẽ được xóa khi đơn hàng còn ở trạng thái "Nhận đơn hàng"

✓ Tóm tắt yêu cầu

▪ Thiết kế giao diện người dùng:

- Học viên thiết kế giao diện Quản lý đơn hàng như hình phía trên. Sử dụng thêm điều khiển Date Chooser Combo để chọn ngày.
- Form này là form nằm trong form chính, được mở ra khi người dùng chọn chức năng "Quản lý đơn hàng" trên menu.

▪ Nhập:

- Tên khách hàng/ ngày đặt hàng/ trạng thái đơn hàng

- **Xuất:**

- Danh sách đơn hàng phù hợp với tiêu chí tìm kiếm

- ✓ **Hướng dẫn**

- Trong lớp QLCSDL, xây dựng thêm các phương thức phù hợp cho việc hiển thị danh sách trạng thái đơn hàng, tìm kiếm đơn hàng, cập nhật đơn hàng, chi tiết đơn hàng, xóa chi tiết đơn hàng
- Gợi ý:

```
// danh sách đơn hàng theo SQL
public List<DonHang> dsDonHangTheoSQL(String sql) throws SQLException,
ClassNotFoundException {
    List<DonHang> rs;
    try (Connection conn = this.connect()) {
        rs = new ArrayList<>();
        java.sql.Statement statement = conn.createStatement();
        ResultSet resultSet = statement.executeQuery(sql);
        while (resultSet.next()) {
            DonHang dh = new DonHang();
            dh.setId(resultSet.getInt("id"));
            dh.setId_khachhang(resultSet.getInt("id_khachhang"));
            dh.setDiachigiaohang(resultSet.getString("diachigiaohang"));
            dh.setDienthoanguoinhan(resultSet.getString("dienthoanguoinhan"));
            dh.setGhichu(resultSet.getString("ghichu"));
            dh.setId_trangthai(resultSet.getInt("id_trangthai"));
            dh.setNgaydathang(resultSet.getDate("ngaydathang"));
            dh.setTenkhachhang(resultSet.getString("hoten"));
            dh.setTennguoinhanhang(resultSet.getString("tennguoinhanhang"));
            dh.setThanhtoan(resultSet.getInt("thanhtoan"));
            rs.add(dh);
        }
    }
    return rs;
}

// lấy đơn hàng theo ID
public DonHang layDonHangTheoID(int ID) throws SQLException, ClassNotFoundException
{
    DonHang dh = null;
```

```

        try (Connection conn = this.connect()) {
            java.sql.Statement statement = conn.createStatement();
            ResultSet resultSet = statement.executeQuery("Select * from donhang inner join
nguoidung on nguoidung.id = donhang.id_khachhang where donhang.id = '" + ID + "'");
            while (resultSet.next()) {
                dh = new DonHang();
                dh.setId(resultSet.getInt("id"));
                dh.setId_khachhang(resultSet.getInt("id_khachhang"));
                dh.setDiachigiaohang(resultSet.getString("diachigiaohang"));
                dh.setDienthoainguoinhan(resultSet.getString("dienthoainguoinhan"));
                dh.setGhichu(resultSet.getString("ghichu"));
                dh.setId_trangthai(resultSet.getInt("id_trangthai"));
                dh.setNgaydathang(resultSet.getDate("ngaydathang"));
                dh.setTenkhachhang(resultSet.getString("hoten"));
                dh.setTennguoinhan(resultSet.getString("tennguoinhanhang"));
                dh.setThanhtoan(resultSet.getInt("thanhtoan"));
            }
        }
        return dh;
    }

    // danh sách CTĐH theo sql
    public List<ChiTietDonHang> dsCTDHTheoSQL(String sql) throws SQLException,
ClassNotFoundException {
        List<ChiTietDonHang> rs;
        try (Connection conn = this.connect()) {
            rs = new ArrayList<>();
            java.sql.Statement statement = conn.createStatement();
            ResultSet resultSet = statement.executeQuery(sql);
            while (resultSet.next()) {
                ChiTietDonHang ctdh = new ChiTietDonHang();
                ctdh.setId(resultSet.getInt("id"));
                ctdh.setId_donhang(resultSet.getInt("id_donhang"));
                ctdh.setId_sanpham(resultSet.getInt("id_sanpham"));
                ctdh.setSoluong(resultSet.getInt("soluong"));
                rs.add(ctdh);
            }
        }
    }

```

```

    }
    return rs;
}

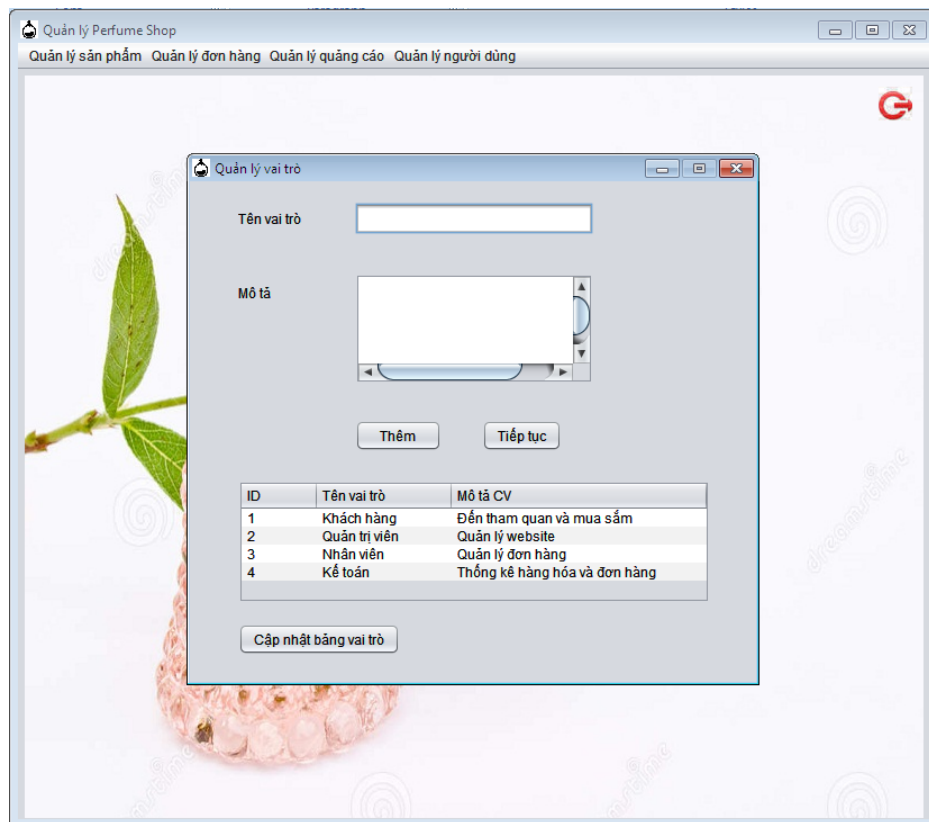
// cập nhật đơn hàng
public void capNhatDonHang(DonHang d) throws SQLException, ClassNotFoundException,
UnsupportedEncodingException {
    try (Connection conn = this.connect()) {
        String sql = "UPDATE DonHang SET tennguoinhanhang = '" +
d.getTennguoinhanhang()+ "', dienthoainguoinhan = '" + d.getDienthoainguoinhan() + "',
diachigiaohang ='" + d.getDiachigiaohang() + "', thanhtoan = '" + d.getThanhtoan() + "',
id_trangthai ='" + d.getId_trangthai() + "' WHERE id = '" + d.getId() + "'";
        PreparedStatement statement = conn.prepareStatement(sql);
        statement.execute();
    }
}

// xóa CTĐH
public void xoaCTDH(int id) throws SQLException, ClassNotFoundException,
UnsupportedEncodingException {
    try (Connection conn = this.connect()) {
        String sql = "DELETE FROM chitietdonhang WHERE id = '" + id + "'";
        PreparedStatement statement = conn.prepareStatement(sql);
        statement.execute();
    }
}
}

```

- Xử lý việc đưa danh sách trạng thái đơn hàng vào combo box
- Xử lý việc tìm kiếm và in xuất danh sách đơn hàng vào table
- Xử lý việc hiển thị đơn hàng
- Xử lý việc cập nhật đơn hàng
- Xử lý việc hiển thị danh sách chi tiết đơn hàng của đơn hàng
- Xử lý việc xóa chi tiết đơn hàng của đơn hàng

5. Yêu cầu 5: Quản lý vai trò



From "Quản lý vai trò" sẽ được mở ra khi người dùng chọn chức năng này trên menu.

Quản lý vai trò

Tên vai trò:

Mô tả:

ID	Tên vai trò	Mô tả CV
1	Khách hàng	Đến tham quan và mua sắm
2	Quản trị viên	Quản lý website
3	Nhân viên	Quản lý đơn hàng
4	Kế toán	Thông kê hàng hóa và đơn hàng

Cập nhật vai trò trực tiếp trên table

✓ **Hướng dẫn sử dụng:**

- Nhập tên vai trò, mô tả > Nhấn “Thêm” > Vai trò mới sẽ được thêm vào CSDL và đồng thời hiển thị trên table.
- Nhấn “Tiếp tục” để xóa nội dung đang có trên các điều khiển
- Cập nhật tên vai trò, mô tả CV trực tiếp trên table > Nhấn “Cập nhật bảng vai trò” để cập nhật nhiều vai trò một lần.

✓ **Tóm tắt yêu cầu**

▪ **Thiết kế giao diện người dùng:**

- Học viên thiết kế giao diện Quản lý vai trò như hình phía trên. Form này vừa có thể thêm mới vừa có thể cập nhật trực tiếp trên table.
- Form này là form nằm trong form chính, được mở ra khi người dùng chọn chức năng “Quản lý vai trò” trên menu.

▪ **Nhập:**

- Tên vai trò, mô tả

▪ **Xuất:**

- Các vai trò trên table

✓ Hướng dẫn

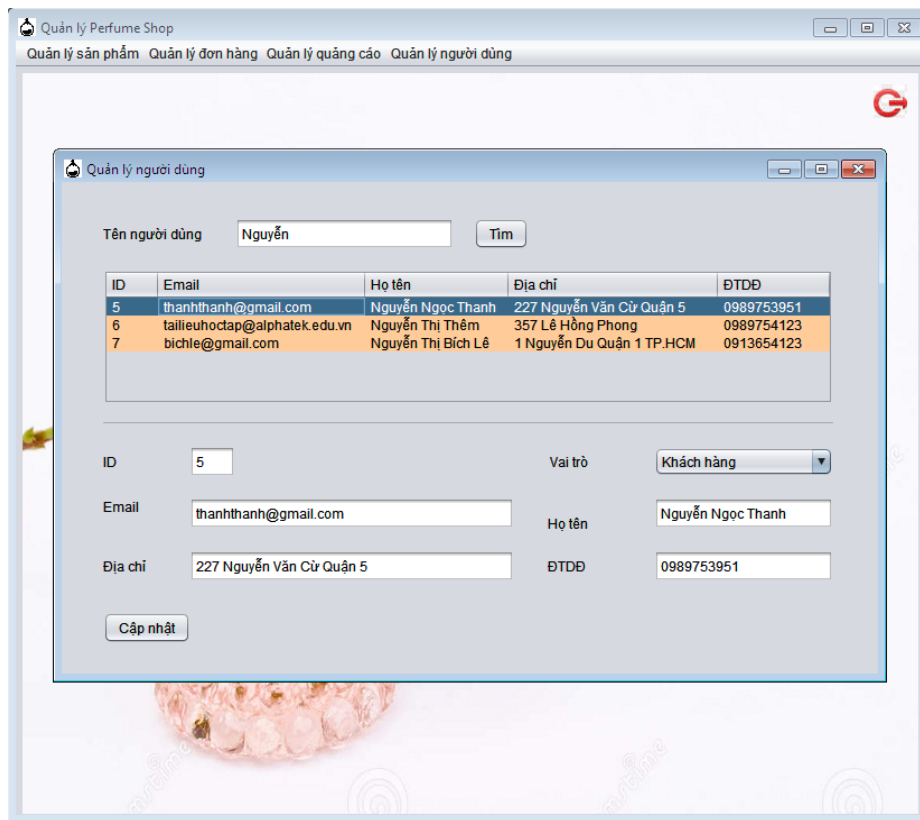
- Trong lớp QLCSDL, xây dựng thêm phương thức thêm mới và cập nhật vai trò
- Gợi ý:

```
// thêm vai trò
public int themVaiTro(VaiTro vt) throws SQLException, ClassNotFoundException,
UnsupportedEncodingException {
    int lastInsertID = 0;
    try (Connection conn = this.connect()) {
        java.sql.Statement statement = conn.createStatement();
        String sql = "INSERT vaiTro VALUES(null, '" + vt.getTenvaitro() + "', '" +
vt.getMota() + "')";
        statement.executeUpdate(sql);
        ResultSet rs= statement.executeQuery("select id from vaiTro order by id desc limit
0,1");
        while (rs.next()) {
            lastInsertID = rs.getInt("id");
            break;
        }
    }
    return lastInsertID;
}

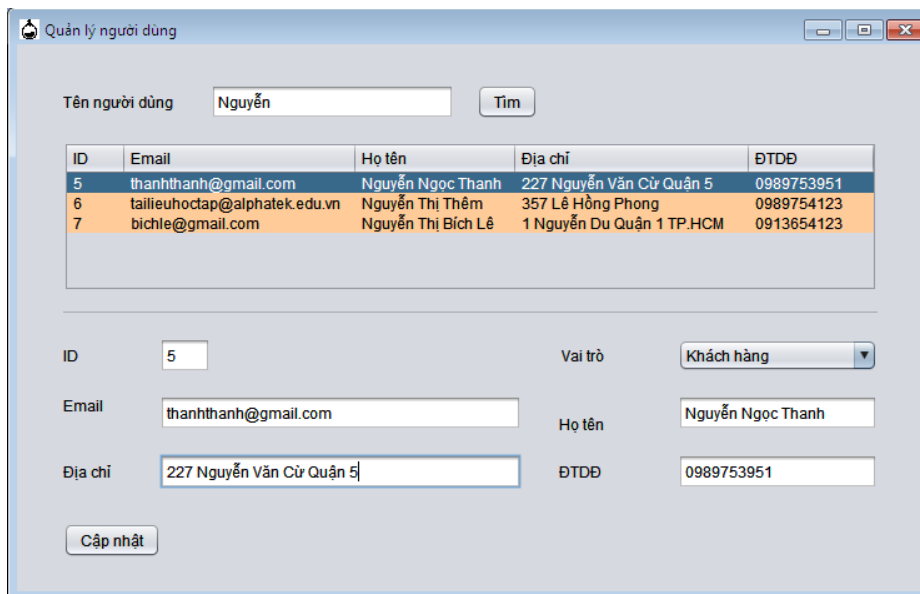
// Cập nhật vai trò
public void capNhatVaiTro(VaiTro vt) throws SQLException, ClassNotFoundException,
UnsupportedEncodingException {
    try (Connection conn = this.connect()) {
        String sql = "UPDATE vaiTro SET tenvaitro = '" + vt.tenvaitro + "', mota = '" +
vt.mota + "' WHERE id = '" + vt.getId() + "'";
        PreparedStatement statement = conn.prepareStatement(sql);
        statement.execute();
    }
}
```

- Xử lý việc thêm mới vai trò vào CSDL
- Xử lý việc hiển thị danh sách vai trò trên table
- Xử lý việc cập nhật các vai trò trực tiếp trên table

6. Yêu cầu 6: Quản lý người dùng



From "Quản lý người dùng" sẽ được mở ra khi người dùng chọn chức năng này trên menu.



Cập nhật thông tin người dùng

✓ Hướng dẫn sử dụng:

- Nhập tên người dùng (tìm tương đối) => Nhấn nút "Tìm" => Danh sách người dùng tìm thấy sẽ hiển thị trên table

- Chọn một người dùng trên table => Thông tin người dùng sẽ hiển thị trên các điều khiển
=> Cập nhật địa chỉ, số điện thoại và vai trò => Nhấn cập nhật.

✓ **Tóm tắt yêu cầu**

▪ **Thiết kế giao diện người dùng:**

- Học viên thiết kế giao diện Quản lý người dùng như hình phía trên.
- Form này là form nằm trong form chính, được mở ra khi người dùng chọn chức năng "Quản lý người dùng" trên menu.

▪ **Nhập:**

- Tên người dùng

▪ **Xuất:**

- Danh sách người dùng thỏa tiêu chí tìm kiếm

✓ **Hướng dẫn**

- Trong lớp QLCSDL, xây dựng thêm phương thức phù hợp cho việc hiển thị danh sách người dùng và cập nhật người dùng.
- Gợi ý:

```
// danh sách người dùng theo SQL
public List<NguoiDung> dsNguoiDung(String sql) throws SQLException,
ClassNotFoundException {
    List<NguoiDung> rs;
    try (Connection conn = this.connect()) {
        rs = new ArrayList<>();
        java.sql.Statement statement = conn.createStatement();
        ResultSet resultSet = statement.executeQuery(sql);
        while (resultSet.next()) {
            NguoiDung nd = new NguoiDung();
            nd.setId(resultSet.getInt("id"));
            nd.setEmail(resultSet.getString("email"));
            nd.setPassword(resultSet.getString("password"));
            nd.setHoten(resultSet.getString("hoten"));
            nd.setDiachi(resultSet.getString("diachi"));
            nd.setDtdd(resultSet.getString("dtdd"));
            nd.setId_vaitro(resultSet.getInt("id_vaitro"));
            rs.add(nd);
        }
    }
}
```

```

        return rs;
    }
    // cập nhật người dùng
    public void capNhatNguoiDung(NguoiDung nd) throws SQLException,
        ClassNotFoundException, UnsupportedEncodingException {
        try (Connection conn = this.connect()) {
            String sql = "UPDATE nguoidung SET diachi = " + nd.getDiachi() + ", dtdd = " +
            nd.getDtdd() + ", id_vaitro = " + nd.getId_vaitro() + " WHERE id = " + nd.getId() + """;
            PreparedStatement statement = conn.prepareStatement(sql);
            statement.execute();
        }
    }
}

```

- Xử lý việc tìm kiếm và hiển thị người dùng
- Xử lý việc hiển thị danh sách vai trò vào combobox khi chọn một người dùng trên table
- Xử lý việc cập nhật thông tin người dùng vào CSDL

Lưu ý: Các yêu cầu chứng năng còn lại, học viên tự thiết kế và xử lý tương tự như những chức năng trên