TRƯỜNG ĐẠI HỌC NAM CẦN THƠ

TRUNG TÂM ĐÀO TẠO CHUẨN ĐẦU RA VÀ PHÁT TRIỂN NGUỒN NHÂN LỰC



Đồng Lê Công Danh Nguyễn Gia Thiên

ĐỀ TÀI

# PHẦN MỀM QUẢN HÀNG HÓA

**BÁO CÁO KỸ NĂNG NGHỀ NGHIỆP**

**Cần Thơ – 2023**

TRƯỜNG ĐẠI HỌC NAM CẦN THƠ

TRUNG TÂM ĐÀO TẠO CHUẨN ĐẦU RA VÀ PHÁT TRIỂN NGUỒN NHÂN LỰC



Đồng Lê Công Danh

Mssv 190799

Nguyễn Gia Thiên

Mssv 199992

KHÓA 7

ĐỀ TÀI

# PHẦN MỀM QUẢN LÝ HÀNG HÓA

**BÁO CÁO KỸ NĂNG NGHỀ NGHIỆP**

**Cần Thơ – 2023**

# MỤC LỤC

[PHẦN MỀM QUẢN LÝ SÁCH 1](#_Toc139293883)

[PHẦN MỀM QUẢN SÁCH 2](#_Toc139293884)

[MỤC LỤC 3](#_Toc139293885)

[DANH MỤC BẢNG 4](#_Toc139293886)

[LỜI NÓI ĐẦU 6](#_Toc139293887)

[CHƯƠNG TRÌNH QUẢN LÝ 8](#_Toc139293888)

[1. Mục đích của phần mềm 8](#_Toc139293889)

[2. Chức năng chính của phần mềm 8](#_Toc139293890)

[3. Xây dựng chương trình 8](#_Toc139293891)

[3.1 Cơ sở dữ liệu 8](#_Toc139293892)

[3.2 Thiết kế các lớp 11](#_Toc139293893)

[4. Xây dựng các lớp trong chương trình 12](#_Toc139293894)

[4.1 Lớp tài khoản 12](#_Toc139293895)

[4.2 Lớp chức vụ 16](#_Toc139293896)

[4.3 Lớp Loại sản phẩm 18](#_Toc139293897)

[4.4 Lớp Sản phẩm 20](#_Toc139293898)

[4.5 Lớp Hóa Đơn 23](#_Toc139293899)

[TÀI LIỆU THAM KHẢO 27](#_Toc139293900)

[PHÂN CÔNG CÔNG VIỆC 28](#_Toc139293901)

# DANH MỤC BẢNG

[Bảng 3. 1 Thực thể Loại Sản Phẩm 8](#_Toc139293770)

[Bảng 3. 2 Thực thể Sản Phẩm 9](#_Toc139293771)

[Bảng 3. 3 Thực thể Chức Vụ 9](#_Toc139293772)

[Bảng 3. 4 Thực thể Tài Khoản 9](#_Toc139293773)

[Bảng 3. 5 Thực thể Hóa Đơn 10](#_Toc139293774)

**DANH MỤC CÁC TỪ VIẾT TẮT**

# LỜI NÓI ĐẦU

Với sự phát triển nhảy vọt của công nghệ thông tin hiện nay, Internet ngày càng giữ vai trò quan trọng trong các lĩnh vực khoa học kĩ thuật và đời sống.

Dĩ nhiên các bạn đã được nghe nói nhiều về Internet, nói một cách đơn giản, Internet là một tập hợp máy tính nối kết với nhau, là một mạng máy tính toàn cầu mà bất kì ai cũng có thể kết nối bằng máy PC của họ.

Với mạng Internet, tin học thật sự tạo nên một cuộc cách mạng trao đổi thông tin trong mọi lĩnh vực văn hóa, xã hội, chính trị, kinh tế... Vì thấy sự cần thiết phải dùng thương mại điện tử để hỗ trợ giới thiệu làng nghề và quảng bá đến bạn hàng trên mọi miền đất nước được nhanh chóng nên mong muốn sưu tầm những bài báo hay những bài viết về làng nghề, viết về các sản phẩm truyền thống làng nghề, các bài viết giới thiệu cơ sở sản xuất của làng nghề và các hình ảnh về sản phẩm làng nghề.

Với lí do đó, được sự hướng dẫn và giúp đỡ của thầy cô em đã chọn đề tài: “Xây dựng ứng dụng quản lý hàng hóa” làm đề tài của mình. Tuy nhiên mặc dù đã nỗ lực hết sức mình nhưng chắc rằng đồ án khó tránh khỏi thiếu sót. Em rất mong nhận được sự thông cảm, những lời góp ý và chỉ bảo tận tình của quý Thầy Cô và các bạn.

**Nhóm sinh viên thực hiện**

Đồng Lê Công Danh

Nguyễn Gia Thiên

CHƯƠNG TRÌNH QUẢN LÝ

## Mục đích của phần mềm

* Việc quản lý hoạt động kinh doanh một (đơn lẻ) hoặc nhiều cửa hàng (chuỗi cửa hàng) hoàn toàn bằng thủ công cần nhiều nguồn nhân lực, tốn thời gian, chi phí và dễ xảy ra sai sót. hiểu được những khó khăn trên của các cửa hàng nhỏ và đã phát triển phần mềm quản lý sách thêm các tính năng phù hợp với người quản lý cửa hàng.
* Các phần mềm quản lý có độ chính xác cao, nhiều tiện ích sẽ là công cụ đắc lực để thúc đẩy doanh số phát triển cho cửa hàng.

## Chức năng chính của phần mềm

* Các chức năng bao gồm :

+ Tài khoản được chia làm 2 nhân viên và Admin.

+ Nhân viên thêm sửa xoá loại sản phẩm, và sản phẩm.

+ Admin có thể quản lý Nhân viên thêm xoá nhân viên.

+ Nhân viên có thể xuất hóa đơn bán sản phẩm và xem danh sách hóa đơn bán sản phẩm.

+ Mục sản phẩm thì có bao gồm thêm sửa xoá và xuất file sang excel.

## Xây dựng chương trình

### 3.1 Cơ sở dữ liệu

Bảng 3. 1 Thực thể Loại Sản Phẩm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Khoá** | **Giải thích** |
| MaLoaiSanPham | int | 10 | Khoá chính | Mã sản phẩm |
| TenLoaiSanPham | varchar | 255 |  | Tên sản phẩm |
| MoTa | varchar | 255 |  | Mô tả |

Bảng 3. 2 Thực thể Sản Phẩm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Khoá** | **Giải thích** |
| ID\_SanPham | int | 10 | Khoá chính | Mã sản phẩm |
| ID\_MaLoai | int | 10 |  | Mã loại |
| TenSanPham | varchar | 255 |  | Tên sản phẩm |
| SoLuong | int | 10 |  | Số lượng |
| Gia | int | 10 |  | Giá |
| NgayTao | date |  |  | Ngày tạo |

Bảng 3. 3 Thực thể Chức Vụ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Khoá** | **Giải thích** |
| ID\_ChucVu | int | 10 | Khoá chính | Mã chức vụ |
| TenChucVu | varchar | 255 |  | Tên chức vụ |
| MoTa | varchar | 255 |  | Mô tả |

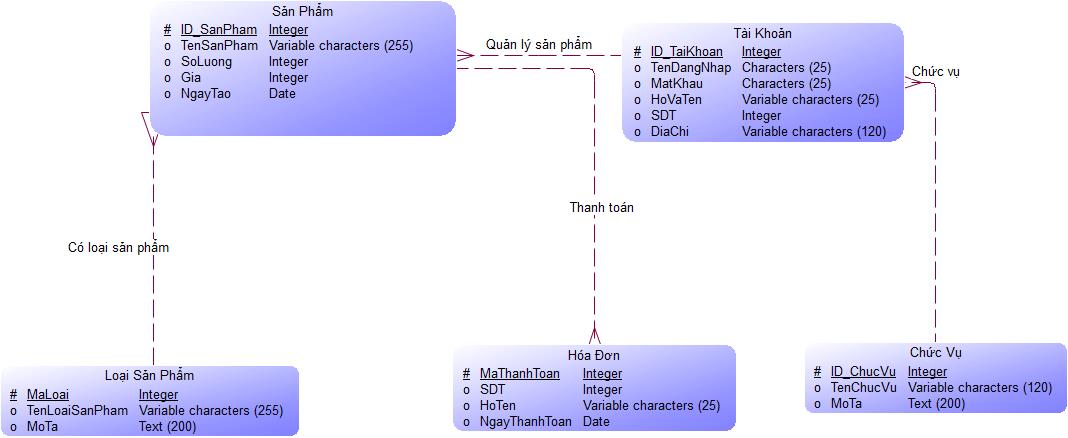
Bảng 3. 4 Thực thể Tài Khoản

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Khoá** | **Giải thích** |
| ID\_TaiKhoan | Int | 10 | Khoá chính | Mã tài khoản |
| ID\_ChucVu | Int | 10 |  | Tên tài khoản |
| TenDangNhap | char | 100 |  | Tên đăng nhập |
| MatKhau | char | 100 |  | Mật khẩu |
| HoVaTen | varchar | 200 |  | Họ và Tên |
| SDT | decimal | (10,0) |  | Số điện thoại |
| DiaChi | varchar | 250 |  | Địa chỉ |

Bảng 3. 5 Thực thể Hóa Đơn

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Khoá** | **Giải thích** |
| MaThanhToan | Int | 10 | Khoá chính | Mã thanh toán |
| MaSach | varchar | 255 | Khoá Phụ | Mã sách |
| SDT | varchar | 255 |  | Số điện thoại |
| HOVATEN | varchar | 100 |  | Họ tên |
| NGAYTHANHTOAN | Date |  |  | Ngày thanh toán |

### 3.2 Thiết kế các lớp

****

Hình 3. 2 Mô hình ERD

## Xây dựng các lớp trong chương trình

### Lớp tài khoản

Controller

1. **public** **class** TaiKhoan {
2. **public** String ID\_TaiKhoan;
3. **public** **int** ID\_ChucVu;
4. **public** String TenDangNhap;
5. **public** String MatKhau;
6. **public** String HoVaTen;
7. **public** String SDT;
8. **public** String DiaChi;
9. **public** TaiKhoan() {
10. }
11. **public** TaiKhoan(String iD\_TaiKhoan, **int** iD\_ChucVu, String tenDangNhap, String matKhau, String hoVaTen,
12. String sDT, String diaChi) {
13. **super**();
14. ID\_TaiKhoan = iD\_TaiKhoan;
15. ID\_ChucVu = iD\_ChucVu;
16. TenDangNhap = tenDangNhap;
17. MatKhau = matKhau;
18. HoVaTen = hoVaTen;
19. SDT = sDT;
20. DiaChi = diaChi;
21. }
22. **public** String getID\_TaiKhoan() {
23. **return** ID\_TaiKhoan;
24. }
25. **public** **void** setID\_TaiKhoan(String iD\_TaiKhoan) {
26. ID\_TaiKhoan = iD\_TaiKhoan;
27. }
28. **public** **int** getID\_ChucVu() {
29. **return** ID\_ChucVu;
30. }
31. **public** **void** setID\_ChucVu(**int** iD\_ChucVu) {
32. ID\_ChucVu = iD\_ChucVu;
33. }
34. **public** String getTenDangNhap() {
35. **return** TenDangNhap;
36. }
37. **public** **void** setTenDangNhap(String tenDangNhap) {
38. TenDangNhap = tenDangNhap;
39. }
40. **public** String getMatKhau() {
41. **return** MatKhau;
42. }
43. **public** **void** setMatKhau(String matKhau) {
44. MatKhau = matKhau;
45. }
46. **public** String getHoVaTen() {
47. **return** HoVaTen;
48. }
49. **public** **void** setHoVaTen(String hoVaTen) {
50. HoVaTen = hoVaTen;
51. }
52. **public** String getSDT() {
53. **return** SDT;
54. }
55. **public** **void** setSDT(String sDT) {
56. SDT = sDT;
57. }
58. **public** String getDiaChi() {
59. **return** DiaChi;
60. }
61. **public** **void** setDiaChi(String diaChi) {
62. DiaChi = diaChi;
63. }
64. }

Model

**public** **class** TaiKhoan **extends** MySQLConnect {

**public** Controllers.TaiKhoan acc;

**public** TaiKhoan() {

**this**.getConn();

}

**public** **boolean** login(String user, String password) {

acc = **new** Controllers.TaiKhoan();

acc.TenDangNhap = user;

acc.MatKhau = password;

String sql = String.*format*("SELECT COUNT(\*) FROM `taikhoan` WHERE tendangnhap = '%s' and matkhau = '%s'", user,

password);

**try** {

**this**.rs = **this**.getVal(sql);

**if** (**this**.rs.next()) {

**return** (**this**.rs.getInt(1) > 0) ? **true** : **false**;

}

} **catch** (Exception e) {

e.printStackTrace();

}

**return** **true**;

}

**public** **boolean** insert(Controllers.TaiKhoan tk) {

**this**.getConn();

String sql = String.*format*(

"INSERT INTO `taikhoan`(`id\_taikhoan`, `id\_chucvu`, `hovaten`, `sdt`, `diachi`, `tendangnhap`, `matkhau`)"

+ " VALUES (null,'%s','%s','%s','%s','%s','%s')",

tk.ID\_ChucVu, tk.HoVaTen, tk.SDT, tk.DiaChi, tk.TenDangNhap, Sys.*cre\_md5*(tk.MatKhau));

**try** {

**if** (**this**.ex\_cmd(sql)) {

**return** **true**;

}

} **catch** (Exception e) {

e.printStackTrace();

**return** **false**;

}

**return** **false**;

}

**public** **boolean** update(Controllers.TaiKhoan tk) {

String sql = String.*format*(

"UPDATE `taikhoan` SET `id\_chucvu`='%s',"

+ "`hovaten`='%s',`sdt`='%s',`diachi`='%s',`tendangnhap`='%s'"

+ " WHERE `id\_taikhoan`='%s'",

tk.ID\_ChucVu, tk.HoVaTen, tk.SDT, tk.DiaChi, tk.TenDangNhap, tk.ID\_TaiKhoan);

**try** {

**this**.getConn();

**if** (**this**.ex\_cmd(sql)) {

**return** **true**;

}

} **catch** (Exception e) {

**return** **false**;

}

**return** **false**;

}

**public** **boolean** delete(Controllers.TaiKhoan tk) {

**try** {

String sql = String.*format*("delete from taikhoan WHERE `id\_taikhoan`= %s", tk.ID\_TaiKhoan);

**this**.getConn();

**if** (**this**.ex\_cmd(sql)) {

**return** **true**;

}

} **catch** (Exception e) {

**return** **false**;

}

**return** **false**;

}

**public** **void** show(JTable table, String key) {

String sql = String.*format*(

"SELECT taikhoan.id\_taikhoan as 'Mã tài khoản',chucvu.tenchucvu as 'Tên chức vụ', taikhoan.hovaten as 'Họ và tên', taikhoan.sdt as "

+ "'Số điện thoại',taikhoan.diachi as 'Địa chỉ',taikhoan.tendangnhap as 'Tên đăng nhập',taikhoan.matkhau as 'Mật khẩu' FROM taikhoan,chucvu"

+ " where taikhoan.id\_chucvu = chucvu.id\_chucvu \n %s",

(key != **null**) ? "and taikhoan.hovaten like '%" + key + "%'" : "");

**this**.getConn();

**this**.rs = **this**.getVal(sql);

DefaultTableModel dtm = (DefaultTableModel) table.getModel();

dtm.setRowCount(0);

**try** {

**while** (rs.next()) {

Object o[] = { rs.getInt("Mã tài khoản"), rs.getString("Tên chức vụ"), rs.getString("Họ và tên"),

rs.getInt("Số điện thoại"), rs.getString("Địa chỉ"), rs.getString("tên đăng nhập"),

rs.getString("mật khẩu") };

dtm.addRow(o);

}

} **catch** (Exception e) {

e.printStackTrace();

}

}

}

### 4.2 Lớp chức vụ

Controller

**public** **class** ChucVu {

**public** **int** ID\_ChucVu;

**public** String TenChucVu;

**public** String MoTa;

**public** ChucVu() {

**super**();

}

**public** ChucVu(**int** iD\_ChucVu, String tenChucVu, String moTa) {

**super**();

ID\_ChucVu = iD\_ChucVu;

TenChucVu = tenChucVu;

MoTa = moTa;

}

**public** **int** getID\_ChucVu() {

**return** ID\_ChucVu;

}

**public** **void** setID\_ChucVu(**int** iD\_ChucVu) {

ID\_ChucVu = iD\_ChucVu;

}

**public** String getTenChucVu() {

**return** TenChucVu;

}

**public** **void** setTenChucVu(String tenChucVu) {

TenChucVu = tenChucVu;

}

**public** String getMoTa() {

**return** MoTa;

}

**public** **void** setMoTa(String moTa) {

MoTa = moTa;

}

@Override

**public** String toString() {

**return** **this**.TenChucVu;

}

Model

**public** **class** ChucVu **extends** Model {

**public** List<Controllers.ChucVu> list = **new** ArrayList<>();

**public** **boolean** insert(Controllers.ChucVu cv) {

String sql = String.*format*("INSERT INTO `chucvu`(`ID\_ChucVu`, `TenChucVu`, `MoTa`) VALUES (null,'%s','%s')",

cv.TenChucVu,cv.MoTa);

**this**.getConn();

System.***out***.print(sql);

**try** {

**if**(**this**.ex\_cmd(sql)) {

**return** **true**;

}

} **catch** (Exception e) {

**return** **false**;

}

**return** **false**;

}

**public** **boolean** update(Controllers.ChucVu cv) {

**try** {

String sql = String.*format*("UPDATE `chucvu` SET `TenChucVu`='%s',`MoTa`='%s' WHERE `ID\_ChucVu`= %s", cv.TenChucVu,cv.MoTa,cv.ID\_ChucVu);

**if**(**this**.ex\_cmd(sql)) {

**return** **true**;

}

} **catch** (Exception e) {

**return** **false**;

}

**return** **false**;

}

**public** **boolean** delete(Controllers.ChucVu cv) {

**try** {

String sql = String.*format*("delete from chucvu WHERE `ID\_ChucVu`= %s",cv.ID\_ChucVu);

**if**(**this**.ex\_cmd(sql)) {

**return** **true**;

}

} **catch** (Exception e) {

**return** **false**;

}

**return** **false**;

}

**public** **void** show(JTable table,String key) {

String sql = String.*format*("SELECT ID\_ChucVu as 'Mã chức vụ',TenChucVu as 'Tên chức vụ', MoTa as 'Mô tả' FROM `chucvu` %s",

(key != **null**)?"where TenChucVu like '%"+key+"%'":"");

// System.out.print(sql);

**this**.getConn();

**this**.rs = **this**.getVal(sql);

DefaultTableModel dtm = (DefaultTableModel)table.getModel();

dtm.setRowCount(0);

**try** {

**while**(rs.next()) {

Object o[] = {

rs.getInt("Mã chức vụ"),rs.getString("Tên chức vụ"),rs.getString("Mô tả")

};

dtm.addRow(o);

}

} **catch** (Exception e) {

// **TODO**: handle exception

e.printStackTrace();

}

}

**public** **void** show(JComboBox jbox,Controllers.ChucVu cv) {

String sql = String.*format*("SELECT ID\_ChucVu as 'Mã chức vụ',TenChucVu as 'Tên chức vụ', MoTa as 'Mô tả' FROM `chucvu`");

**this**.getConn();

**this**.rs = **this**.getVal(sql);

// DefaultComboBoxModel<Controllers.LoaiSanPham> dcm = (DefaultComboBoxModel)jbox.getModel();

**try** {

**while**(rs.next()) {

cv = **new** Controllers.ChucVu(rs.getInt("Mã chức vụ"),rs.getString("Tên chức vụ"),rs.getString("Mô tả"));

**this**.list.add(cv);

jbox.addItem(cv.toString());

}

} **catch** (Exception e) {

e.printStackTrace();

}

}

}

### 4.3 Lớp Loại sản phẩm

Controller

**public** **class** LoaiSanPham {

**public** **int** MaLoaiSanPham;

**public** String TenLoaiSanPham;

**public** String MoTa;

**public** LoaiSanPham() {

**super**();

}

**public** LoaiSanPham(**int** maLoaiSanPham, String tenLoaiSanPham, String moTa) {

**super**();

MaLoaiSanPham = maLoaiSanPham;

TenLoaiSanPham = tenLoaiSanPham;

MoTa = moTa;

}

**public** **int** getMaLoaiSanPham() {

**return** MaLoaiSanPham;

}

**public** **void** setMaLoaiSanPham(**int** maLoaiSanPham) {

MaLoaiSanPham = maLoaiSanPham;

}

**public** String getTenLoaiSanPham() {

**return** TenLoaiSanPham;

}

**public** **void** setTenLoaiSanPham(String tenLoaiSanPham) {

TenLoaiSanPham = tenLoaiSanPham;

}

**public** String getMoTa() {

**return** MoTa;

}

**public** **void** setMoTa(String moTa) {

MoTa = moTa;

}

@Override

**public** String toString() {

**return** **this**.TenLoaiSanPham;

}

}

Model

**public** **class** LoaiSanPham **extends** Model {

**public** List<Controllers.LoaiSanPham> list = **new** ArrayList<>();

**public** **boolean** insert(Controllers.LoaiSanPham lsp) {

String sql = String.*format*("INSERT INTO `loaisanpham`(`MaLoaiSanPham`, `TenLoaiSanPham`, `MoTa`) VALUES (null,'%s','%s')", lsp.TenLoaiSanPham,lsp.MoTa);

**this**.getConn();

System.***out***.print(sql);

**try** {

**if**(**this**.ex\_cmd(sql)) {

**return** **true**;

}

} **catch** (Exception e) {

**return** **false**;

}

**return** **false**;

}

**public** **boolean** update(Controllers.LoaiSanPham lsp) {

**try** {

String sql = String.*format*("UPDATE `loaisanpham` SET `TenLoaiSanPham`='%s',`MoTa`='%s' WHERE `MaLoaiSanPham`= %s", lsp.TenLoaiSanPham,lsp.MoTa,lsp.MaLoaiSanPham);

**if**(**this**.ex\_cmd(sql)) {

**return** **true**;

}

} **catch** (Exception e) {

**return** **false**;

}

**return** **false**;

}

**public** **boolean** delete(Controllers.LoaiSanPham lsp) {

**try** {

String sql = String.*format*("delete from loaisanpham WHERE `MaLoaiSanPham`= %s",lsp.MaLoaiSanPham);

**if**(**this**.ex\_cmd(sql)) {

**return** **true**;

}

} **catch** (Exception e) {

**return** **false**;

}

**return** **false**;

}

**public** **void** show(JTable table,String key) {

String sql = String.*format*("SELECT MaLoaiSanPham as 'Mã sản phẩm', TenLoaiSanPham as 'Tên sản phẩm',MoTa as 'Mô tả' FROM `loaisanpham` %s",

(key != **null**)?"where TenLoaiSanPham like '%"+key+"%'":"");

// System.out.print(sql);

**this**.getConn();

**this**.rs = **this**.getVal(sql);

DefaultTableModel dtm = (DefaultTableModel)table.getModel();

dtm.setRowCount(0);

**try** {

**while**(rs.next()) {

Object o[] = {

rs.getInt("Mã sản phẩm"),rs.getString("Tên sản phẩm"),rs.getString("Mô tả")

};

dtm.addRow(o);

}

} **catch** (Exception e) {

// **TODO**: handle exception

e.printStackTrace();

}

}

**public** **void** show(JComboBox jbox,Controllers.LoaiSanPham loaisp) {

String sql = String.*format*("SELECT MaLoaiSanPham as 'Mã sản phẩm', TenLoaiSanPham as 'Tên sản phẩm',MoTa as 'Mô tả' FROM `loaisanpham`");

**this**.getConn();

**this**.rs = **this**.getVal(sql);

// DefaultComboBoxModel<Controllers.LoaiSanPham> dcm = (DefaultComboBoxModel)jbox.getModel();

**try** {

**while**(rs.next()) {

loaisp = **new** Controllers.LoaiSanPham(rs.getInt("Mã sản phẩm"),rs.getString("Tên sản phẩm"),rs.getString("Mô tả"));

**this**.list.add(loaisp);

jbox.addItem(loaisp.toString());

}

} **catch** (Exception e) {

e.printStackTrace();

}

}

}

### 4.4 Lớp Sản phẩm

Controller

**public** **class** SanPham {

**public** **int** ID\_SanPham;

**public** **int** ID\_MaLoai;

**public** String TenSanPham;

**public** **int** SoLuong;

**public** **int** Gia;

**public** String NgayTao;

**public** SanPham() {

**super**();

}

**public** SanPham(**int** iD\_SanPham, **int** iD\_MaLoai, String tenSanPham, **int** soLuong, **int** gia, String ngayTao) {

**super**();

ID\_SanPham = iD\_SanPham;

ID\_MaLoai = iD\_MaLoai;

TenSanPham = tenSanPham;

SoLuong = soLuong;

Gia = gia;

NgayTao = ngayTao;

}

**public** **int** getID\_SanPham() {

**return** ID\_SanPham;

}

**public** **void** setID\_SanPham(**int** iD\_SanPham) {

ID\_SanPham = iD\_SanPham;

}

**public** **int** getID\_MaLoai() {

**return** ID\_MaLoai;

}

**public** **void** setID\_MaLoai(**int** iD\_MaLoai) {

ID\_MaLoai = iD\_MaLoai;

}

**public** String getTenSanPham() {

**return** TenSanPham;

}

**public** **void** setTenSanPham(String tenSanPham) {

TenSanPham = tenSanPham;

}

**public** **int** getSoLuong() {

**return** SoLuong;

}

**public** **void** setSoLuong(**int** soLuong) {

SoLuong = soLuong;

}

**public** **int** getGia() {

**return** Gia;

}

**public** **void** setGia(**int** gia) {

Gia = gia;

}

**public** String getNgayTao() {

**return** NgayTao;

}

**public** **void** setNgayTao(String ngayTao) {

NgayTao = ngayTao;

}

}

Model

**public** **class** SanPham **extends** Model {

**public** **boolean** insert(Controllers.SanPham sp) {

**this**.getConn();

String sql = String.*format*("INSERT INTO `sanpham`(`ID\_SanPham`, `ID\_MaLoai`, `TenSanPham`, `SoLuong`, `Gia`, `NgayTao`)"

+ " VALUES (null,'%s','%s','%s','%s',CURRENT\_TIMESTAMP())",sp.ID\_MaLoai,sp.TenSanPham,sp.SoLuong,sp.Gia);

// System.out.print(sql);

**try** {

**if**(**this**.ex\_cmd(sql)) {

**return** **true**;

}

} **catch** (Exception e) {

e.printStackTrace();

**return** **false**;

}

**return** **false**;

}

**public** **boolean** update(Controllers.SanPham sp) {

String sql = String.*format*("UPDATE `sanpham` SET `ID\_MaLoai`='%s',"

+ "`TenSanPham`='%s',`SoLuong`='%s',`Gia`='%s'"

+" WHERE `ID\_SanPham`='%s'",sp.ID\_MaLoai, sp.TenSanPham,sp.SoLuong,sp.Gia,sp.ID\_SanPham);

// System.out.print(sql);

**try** {

**this**.getConn();

**if**(**this**.ex\_cmd(sql)) {

**return** **true**;

}

} **catch** (Exception e) {

**return** **false**;

}

**return** **false**;

}

**public** **boolean** delete(Controllers.SanPham sp) {

**try** {

String sql = String.*format*("delete from sanpham WHERE `ID\_SanPham`= %s",sp.ID\_SanPham);

System.***out***.print(sql);

**this**.getConn();

**if**(**this**.ex\_cmd(sql)) {

**return** **true**;

}

} **catch** (Exception e) {

**return** **false**;

}

**return** **false**;

}

**public** **void** show(JTable table,String key) {

String sql = String.*format*("SELECT sanpham.ID\_SanPham as 'Mã sản phẩm',loaisanpham.TenLoaiSanPham as 'Tên loại sản phẩm', sanpham.TenSanPham as 'Tên sản phẩm', sanpham.SoLuong as 'Số lượng',sanpham.Gia as 'Giá',sanpham.NgayTao as 'Ngày tạo'"

+ " FROM sanpham,loaisanpham where sanpham.ID\_MaLoai = loaisanpham.MaLoaiSanPham %s",(key != **null**)?"and sanpham.tensanpham like '%"+key+"%'":"");

**this**.getConn();

**this**.rs = **this**.getVal(sql);

DefaultTableModel dtm = (DefaultTableModel)table.getModel();

dtm.setRowCount(0);

**try** {

**while**(rs.next()) {

Object o[] = {

rs.getInt("Mã sản phẩm"),rs.getString("Tên sản phẩm"),rs.getString("Tên loại sản phẩm"),

rs.getInt("Số lượng"),rs.getInt("Giá"),rs.getString("Ngày tạo")

};

dtm.addRow(o);

}

} **catch** (Exception e) {

// **TODO**: handle exception

e.printStackTrace();

}

}

**public** **void** exportExcel(JTable table) {

JFileChooser chooser = **new** JFileChooser();

**int** i = chooser.showSaveDialog(chooser);

**if** (i == JFileChooser.***APPROVE\_OPTION***) {

File file = chooser.getSelectedFile();

**try** {

BufferedWriter out = **new** BufferedWriter(

**new** OutputStreamWriter(**new** FileOutputStream(file + ".xls"), StandardCharsets.***ISO\_8859\_1***));

// BufferedWriter out = new BufferedWriter(new OutputStreamWriter(new FileOutputStream(file + ".txt"),StandardCharsets.UTF\_8));

DefaultTableModel model = (DefaultTableModel) table.getModel();

// ten Cot

**for** (**int** j = 0; j < model.getColumnCount(); j++) {

System.***out***.print(model.getColumnName(j));

out.write(model.getColumnName(j) + "\t");

}

out.write("\n");

// Lay du lieu dong

**for** (**int** j = 0; j < model.getRowCount(); j++) {

**for** (**int** k = 0; k < model.getColumnCount(); k++) {

// System.out.print(model.);

out.write(model.getValueAt(j, k).toString().replace("\n", ",") + "\t");

}

out.write("\n");

}

out.close();

JOptionPane.*showMessageDialog*(**null**, "Lưu file thành công!");

} **catch** (Exception e2) {

JOptionPane.*showMessageDialog*(**null**, "Lỗi khi lưu file!");

}

}

}

}

### 4.5 Lớp Hóa Đơn

Controller

**package** Controllers;

**public** **class** ThanhToan {

**public** **int** mathanhtoan ;

**public** **int** MaSach ;

**public** String hoten;

**public** String sdt;

**public** String ngaythanhtoan;

**public** ThanhToan() {

**super**();

}

**public** ThanhToan(**int** Mathanhtoan,**int** MaSach, String Hoten, String Sdt, String Ngaythanhtoan) {

**super**();

mathanhtoan = Mathanhtoan;

MaSach = MaSach;

hoten = Hoten;

sdt = Sdt;

ngaythanhtoan = Ngaythanhtoan;

}

**public** **int** getMathanhtoan() {

**return** mathanhtoan;

}

**public** **void** setMathanhtoan(**int** mathanhtoan) {

**this**.mathanhtoan = mathanhtoan;

}

**public** **int** getMaSach() {

**return** MaSach;

}

**public** **void** setMaSach(**int** MaSach) {

**this**.MaSach = MaSach;

}

**public** String getHoten() {

**return** hoten;

}

**public** **void** setHoten(String hoten) {

**this**.hoten = hoten;

}

**public** String getSdt() {

**return** sdt;

}

**public** **void** setSdt(String sdt) {

**this**.sdt = sdt;

}

**public** String getNgaythanhtoan() {

**return** ngaythanhtoan;

}

**public** **void** setNgaythanhtoan(String ngaythanhtoan) {

**this**.ngaythanhtoan = ngaythanhtoan;

}

}

Model

package Models;

import java.util.ArrayList;

import java.util.List;

import javax.swing.JComboBox;

import javax.swing.JTable;

import javax.swing.table.DefaultTableModel;

import Controllers.SanPham;

public class ThanhToan extends Model {

public List<Controllers.SanPham> list = new ArrayList<>();

public ThanhToan() {

}

public boolean insert(Controllers.ThanhToan tt) {

String sql = String.format("INSERT INTO `thanhtoan`(`mathanhtoan`, `MaSach`, `sdt`, `hoten`, `ngaythanhtoan`) VALUES (null,'%s','%s','%s',CURDATE())",

tt.MaSach,tt.sdt,tt.hoten);

this.getConn();

System.out.print(sql);

try {

if(this.ex\_cmd(sql)) {

return true;

}

} catch (Exception e) {

return false;

}

return false;

}

public boolean update(Controllers.ThanhToan th) {

try {

String sql = String.format("UPDATE `thanhtoan` SET `MaSach`='%s', `sdt`='%s', `hoten`='%s' WHERE `mathanhtoan`= %s", th.MaSach, th.sdt, th.hoten, th.mathanhtoan);

if(this.ex\_cmd(sql)) {

return true;

}

} catch (Exception e) {

return false;

}

return false;

}

public boolean delete(Controllers.ThanhToan th) {

try {

String sql = String.format("delete from thanhtoan WHERE `mathanhtoan`= %s", th.mathanhtoan);

if(this.ex\_cmd(sql)) {

return true;

}

} catch (Exception e) {

return false;

}

return false;

}

public void show(JTable table,String key) {

String sql = String.format("SELECT th.mathanhtoan as 'Mã Thanh Toán' , sp.TenSanPham as 'Tên Sách' , th.sdt as 'Số Điện Thoại' , th.hoten as 'Họ Tên', sp.gia as 'Giá', th.ngaythanhtoan as 'Ngày tạo' FROM thanhtoan th, sanpham sp WHERE sp.ID\_SanPham = th.MaSach %s",

(key != null)?"where sdt like '"+key+"'":"");

// System.out.print(sql);

this.getConn();

this.rs = this.getVal(sql);

DefaultTableModel dtm = (DefaultTableModel)table.getModel();

dtm.setRowCount(0);

try {

while(rs.next()) {

Object o[] = {

rs.getInt("Mã Thanh Toán"),rs.getString("Tên Sách"),rs.getString("Số Điện Thoại"),rs.getString("Họ Tên"),rs.getString("Giá"),rs.getString("Ngày tạo")

};

dtm.addRow(o);

}

} catch (Exception e) {

// TODO: handle exception

e.printStackTrace();

}

}

public void show(JComboBox jbox,Controllers.SanPham sp) {

String sql = String.format("SELECT sp.ID\_SanPham as 'Mã Sản Phẩm' , sp.TenSanPham as 'Tên Sản Phẩm' FROM sanpham sp");

this.getConn();

this.rs = this.getVal(sql);

// DefaultComboBoxModel<Controllers.SanPham> dcm = (DefaultComboBoxModel)jbox.getModel();

try {

while(rs.next()) {

sp = new Controllers.SanPham(rs.getInt("Mã Sản Phẩm"),rs.getString("Tên Sản Phẩm"));

this.list.add(sp);

jbox.addItem(sp.toString());

}

} catch (Exception e) {

e.printStackTrace();

}

}

public void showcb(JComboBox table) {

String sql = String.format("SELECT sp.ID\_SanPham as 'Mã Sản Phẩm' , sp.TenSanPham as 'Tên Sản Phẩm' FROM sanpham sp", null);

this.getConn();

this.rs = this.getVal(sql);

DefaultTableModel dtm = (DefaultTableModel)table.getModel();

dtm.setRowCount(0);

try {

while(rs.next()) {

Object o[] = {

rs.getInt("Mã Sản Phẩm"),rs.getString("Tên Sản Phẩm")

};

dtm.addRow(o);

}

} catch (Exception e) {

// TODO: handle exception

e.printStackTrace();

}

}

}

# TÀI LIỆU THAM KHẢO

[1]. Giáo trình hệ quản trị cơ sở dữ liệu, Khoa Công nghệ thông tin, Trường Đại học Nam Cần Thơ.

[2]. Giáo trình Phân tích và thiết kế hệ thống thông tin. Khoa Công nghệ thông tin, Trường Đại học Nam Cần Thơ.

[3]. vi.wikipedia,“ MySQL là gì? ” truy cập ngày 01/07/2022, <https://vi.wikipedia.org/wiki/MySQL>.

[4].vi.wikipedia,”Eclipse là gì?” truy cập ngày 01/07/2022,

< https://vn.got-it.ai/blog/eclipse-la-gi-huong-dan-cai-dat-eclipse-chi-tiet-nhat>.

[5]. vi.wikipedia,”Java là gì?” truy cập ngày 01/07/2022, <<https://quantrimang.com/java-la-gi-tai-sao-nen-chon-java-143130>>.

# PHÂN CÔNG CÔNG VIỆC

|  |  |  |  |
| --- | --- | --- | --- |
| **MSSV** | **Họ Tên** | **Lớp** | **Nội dung công việc tham gia thự hiện** |
| 190799 | Đồng Lê Công Danh | DH19TIN01 | Tạo project,Sữa lỗi,tìm kiếm thông tin tài liệu,lên ý tưởng |
| 199992 | Nguyễn Gia Thiên | DH19TIN01 | Tìm kiếm tài liệu,tạo file,vẽ mô hình efd,erd,khuyên răn |