**TRƯỜNG ĐẠI HỌC KHOA HỌC HUẾ**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP NHÓM**

**KIỂM ĐỊNH PHẦN MỀM**

**Đề tài: Kiểm thử phần mềm quản lý quán Cafe**

**NHÓM 2**

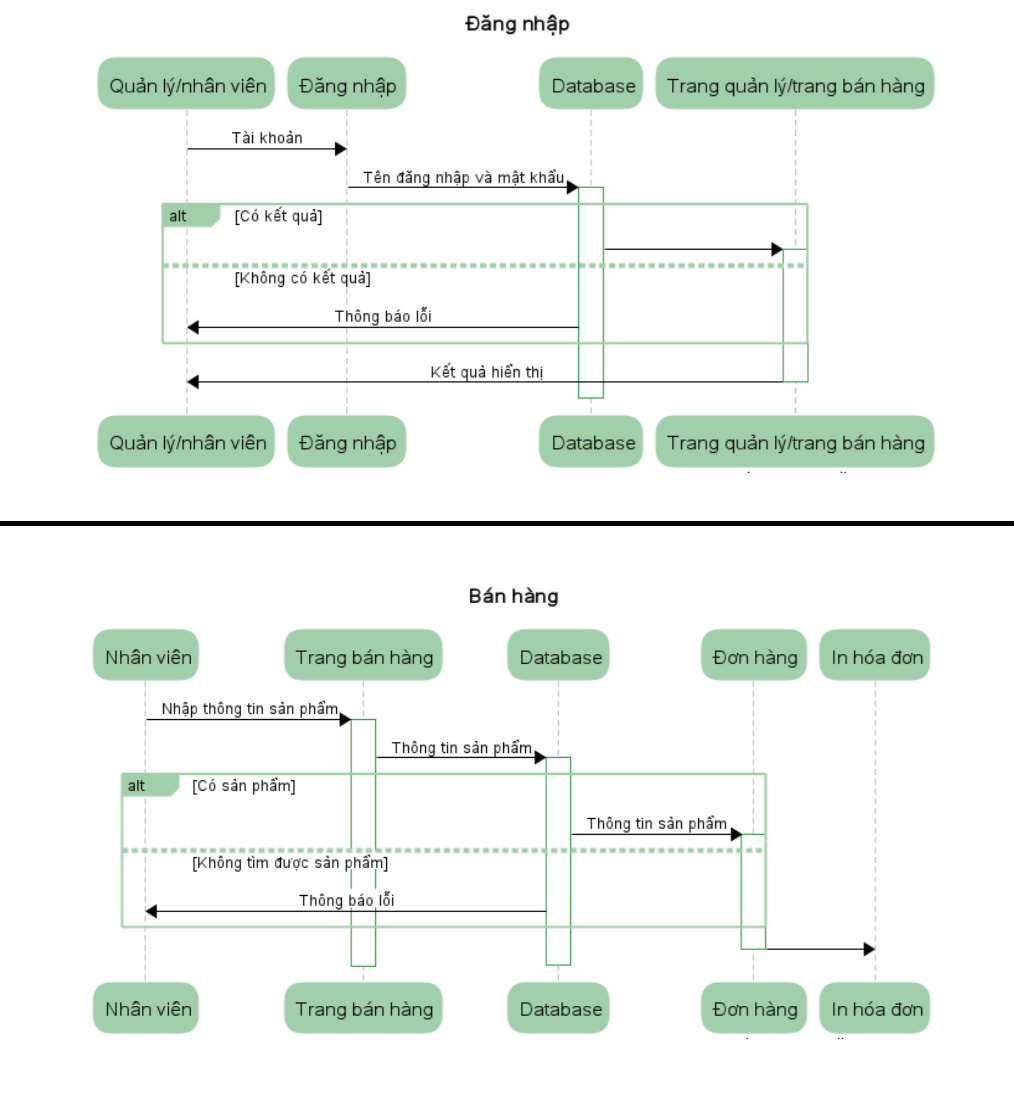
**Giáo viên hướng dẫn: Lê Văn Tường Lân**

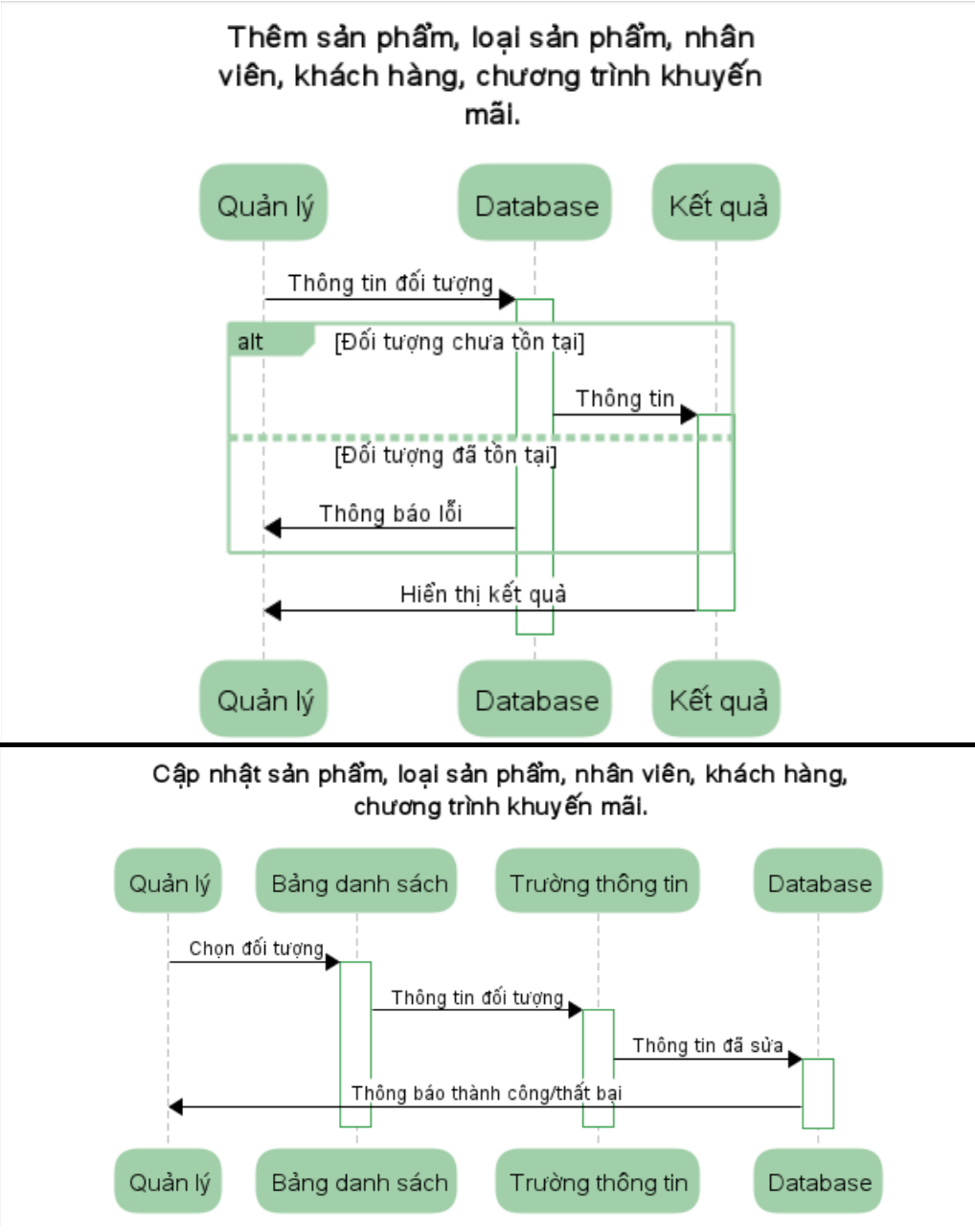
**Sinh viên thực hiện:**

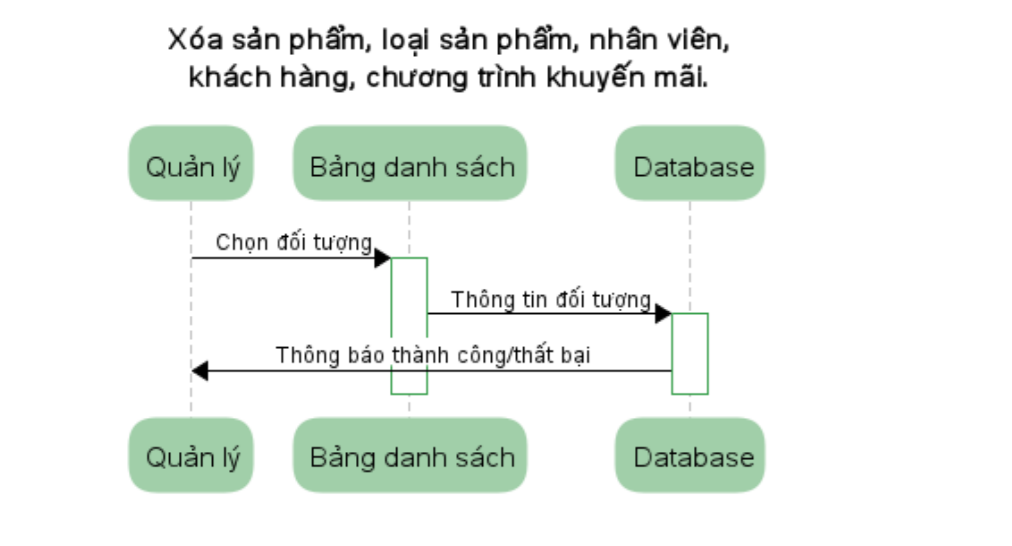
1. **Phạm Minh Đức (Nhóm trưởng)**
2. **Phan Quốc Hùng**
3. **La Hữu Đạt**
4. **Nguyễn Xuân Ý**
5. **Bùi Khắc Nhật**
6. **Nguyễn Lương Hiếu**
7. **Cao Văn Thanh Tuấn**

# MÔ TẢ MÔ HÌNH DỮ LIỆU CỦA HỆ THỐNG

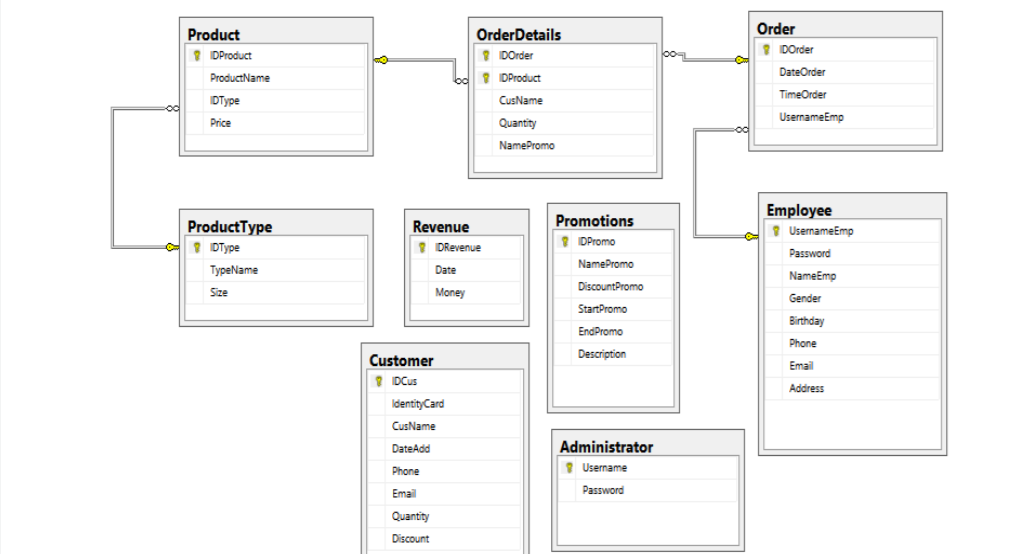
## I. Sơ đồ dữ liệu của hệ thống



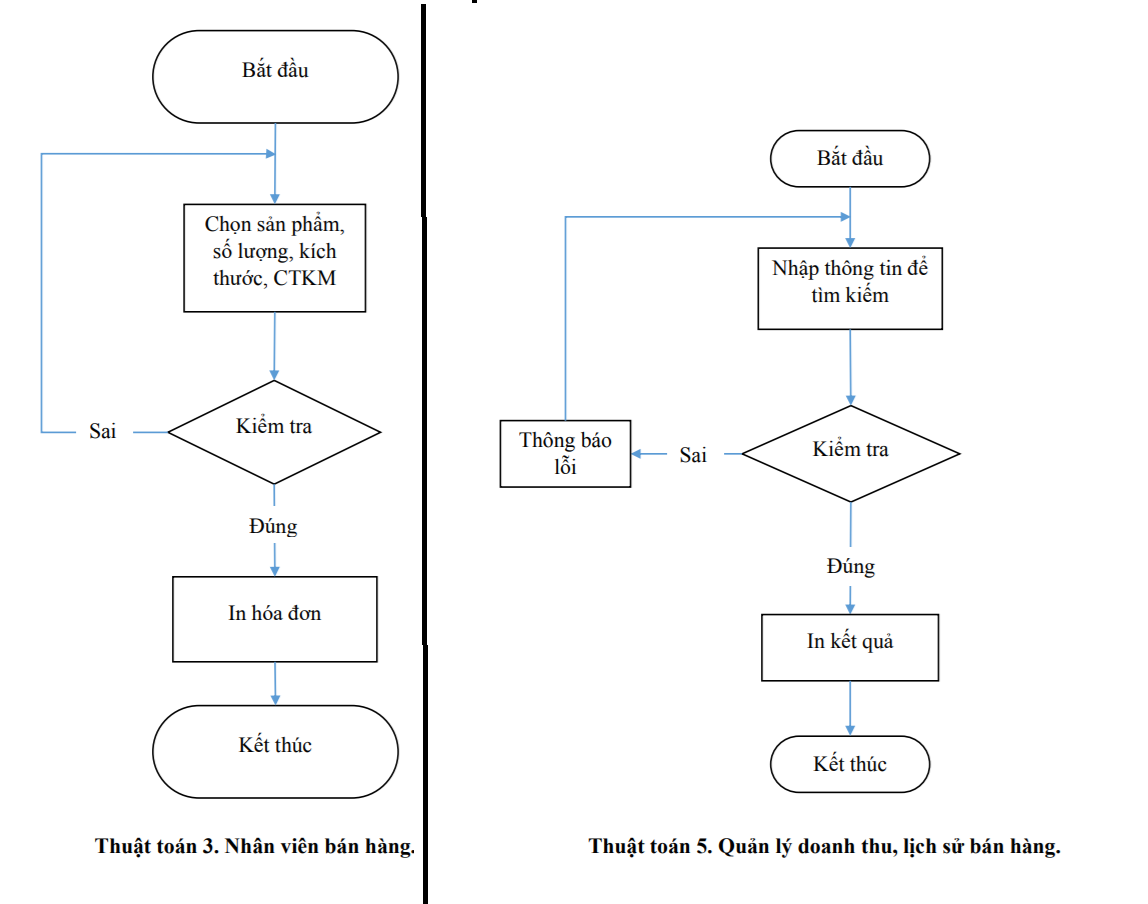
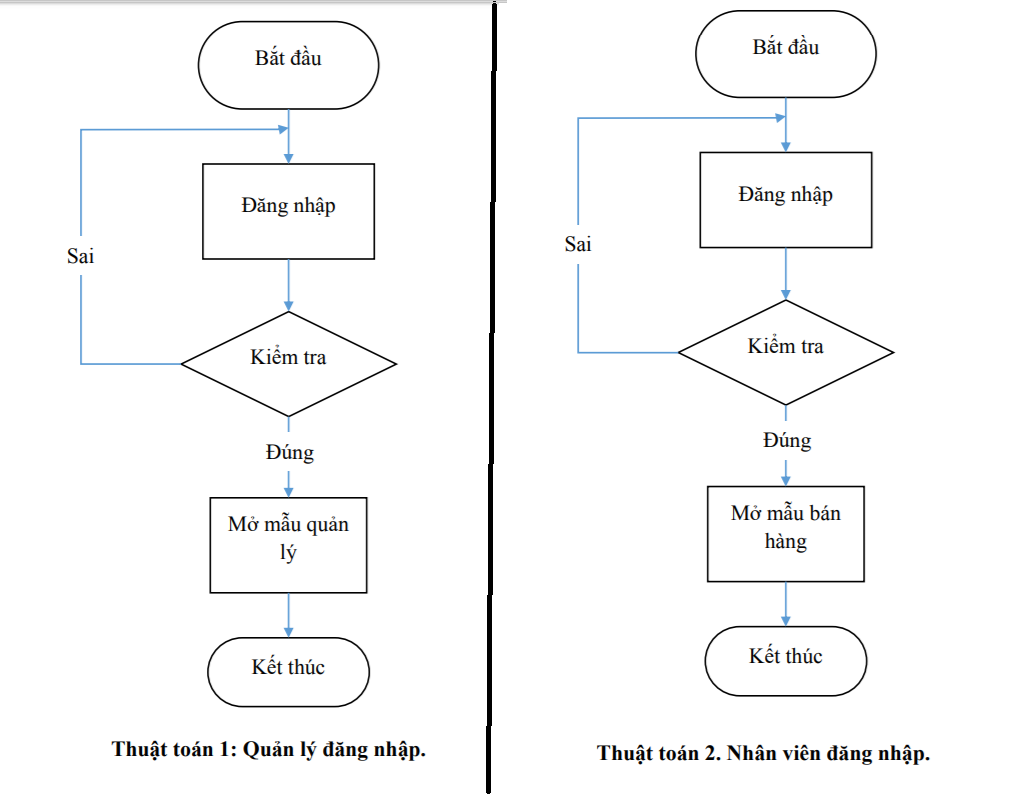


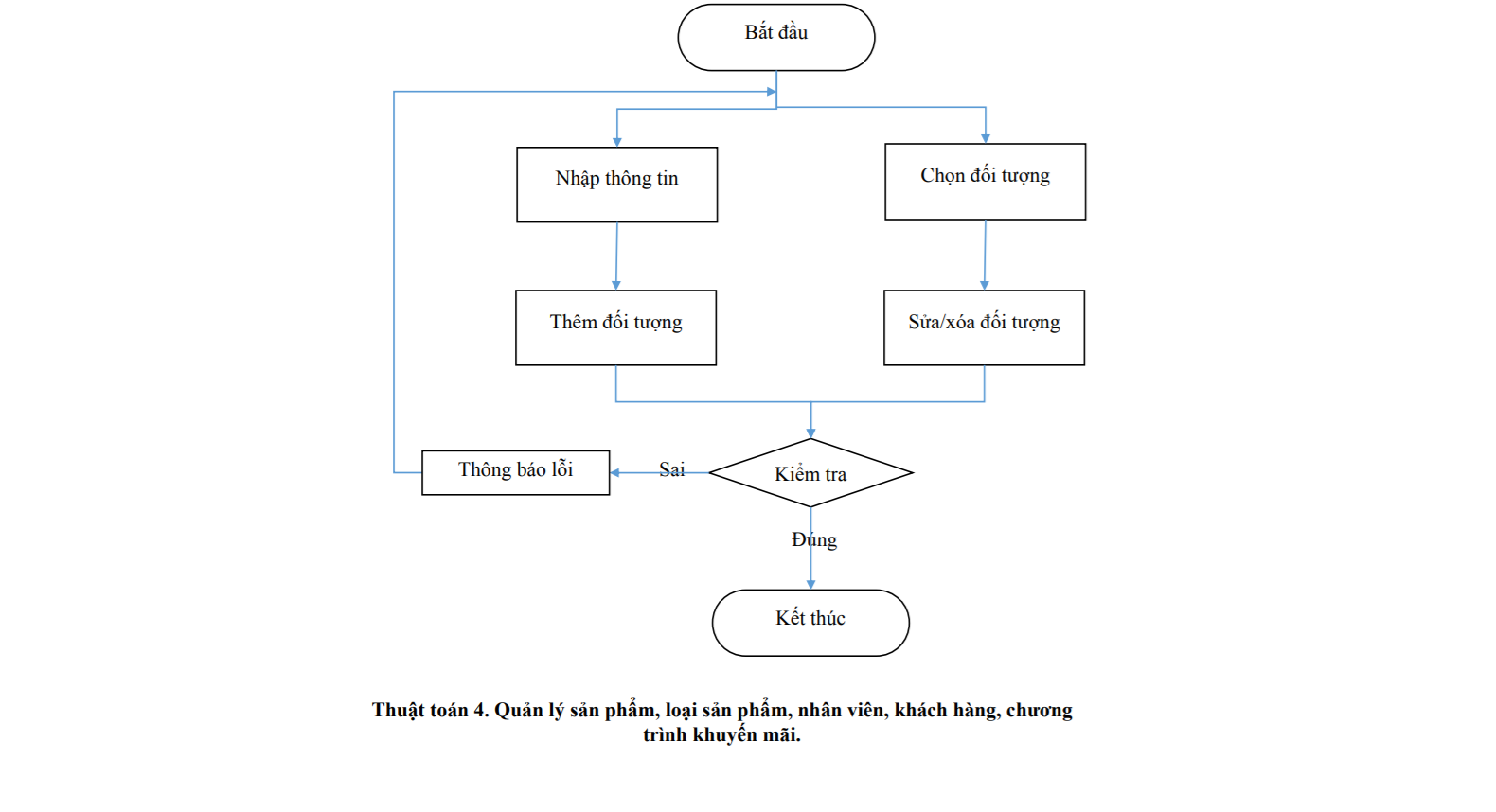


## II. Sơ đồ quan hệ các bảng cơ sở dữ liệu



## III. Sơ đồ thuật toán các chức năng cơ bản





**KIỂM THỬ CHỨC NĂNG**

## I. Chức năng đăng nhập quản lý/nhân viên

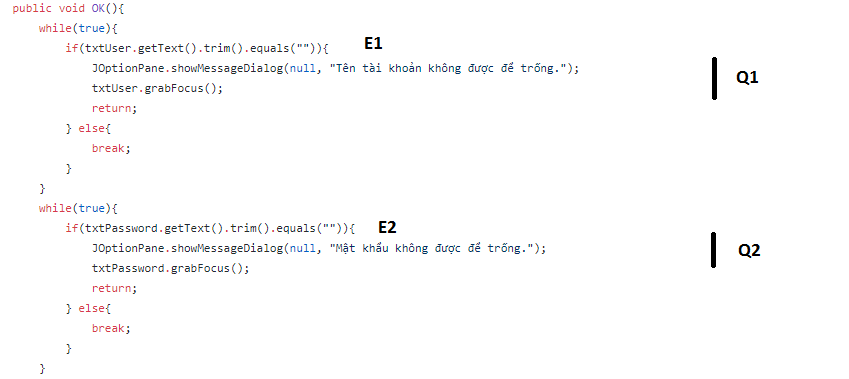
1. Mô tả yêu cầu:

- Quản lý/nhân viên đăng nhập qua cửa sổ này để vào quản lý/bán hàng.

2. Giao diện chức năng:



3. Mã nguồn và kiểm tra module chức năng:

****

****

**Theo mô tả ta có các mệnh đề**

1. Mệnh đề vào {A : database Administrator, database Employee, txtUser ∈ database Administrator, txtUser ∈ database Employee, txtPassword ∈ database Administrator, txtPassword ∈ database Employee}

Mệnh đề ra {B : thông báo đăng nhập thành công và hiển thị form tương ứng hoặc thông báo đăng nhập thất bại}

Mệnh đề bất biến {C : database Administrator, database Employee, cbRole.getSelectedIndex() == 1 ∈ database Administrator, cbRole.getSelectedIndex() == 0 ∈ database Employee }

Với : các điều kiện E1 : txtUser.getText().trim().equals("")

E2 : txtPassword.getText().trim().equals("")

E3 : if (cbRole.getSelectedIndex() == 1)

E4 : if (cbRole.getSelectedIndex() == 0)

E5 : if (rs.next())

Trong đó : cbRole: hàm sẽ kiểm tra nếu người dùng chọn vai trò là admin thì truy xuất dữ liệu ở database administrator ngược lại nếu chọn vai trò là nhân viên thì truy xuất dữ liệu ở database Employee.

1. CM {A} => P => {C}

{A,E1,E2} Q1,Q2 =logic=>{C} (1)

{A,!E1,!E2,E3} Q3 =logic=>{C} (2)

{A,!E1,!E2,E3,E5} Q4 =logic=>{C} (3)

{A,!E1,!E2,E4} Q5 =logic=>{C} (4)

{A,!E1,!E2,E4,E5} Q6 =logic=>{C} (5)

(1)(2)(3)(4)(5)=>{A} = P => {C}

* 1. Dễ dàng ta có : {C,E5,E1,E2} qua đoạn trình Q1,Q2 :

|  |
| --- |
| JOptionPane.showMessageDialog(null, Tên tài khoản không được để trống.");  txtUser.grabFocus(); |
|  |  |

Logic => {C} (6)

* 1. Và {C,E5,!E1,!E2,E3} qua đoạn trình Q4 :

|  |
| --- |
| new AdminForm().setVisible(true);  this.setVisible(false); |
|  |  |

Logic => {C} (7)

* 1. Với {C,E5,!E1,!E2,E4} qua đoạn trình Q6 :

|  |
| --- |
| new BillForm(rs.getString("NameEmp")).setVisible(true);  this.setVisible(false);  {C,E5,!E1,!E2,E3} Q3 {C} (8)  {C,E5,!E1,!E2,E4} Q4 {C} (9)  Từ (6)(7)(8)(9) ta có mệnh đề {C} là bất biến của đoạn trình P.   1. Nên kết thúc đoạn trình P ta có {C,!E5}   Ta thấy {C,!E5}=Logic=>{B} hay chương trình trên đúng. |

4. Kiểm thử hộp đen:

* **Phương pháp tương đương**
* **Xác định các lớp tương đương:**

|  |  |  |
| --- | --- | --- |
| **Các giá trị đầu vào** | **Lớp tương đương hợp lệ** | **Lớp tương đương không hợp lệ** |
| Nhập tên đăng nhập | Đúng với tên đăng nhập trong dữ liệu (SQL). (1) | * Chưa nhập tên đăng nhập. (2) * Nhập thiếu ký tự. (3) * Nhập thừa ký tự. (4) * Nhập sai tên đăng nhập người dùng. (5) * Nhập có ký tự space đầu và cuối tên đăng nhập. (6) * Có ký tự đặc biệt. (7) |
| Nhập mật khẩu | Đúng với mật khẩu trong dữ liệu (SQL). (8) | * Chưa nhập mật khẩu. (9) * Nhập thiếu ký tự. (10) * Nhập thừa ký tự. (11) * Nhập sai mật khẩu người dùng. (12) * Nhập có ký tự space đầu và cuối mật khẩu. (13) * Có ký tự đặc biệt. (14) |
| Chọn quyền | * Nhân viên chọn quyền nhân viên. (15) * Admin chọn quyền quản lý. (16) | * Admin nhưng chọn quyền nhân viên. (17) * Nhân viên chọn quyền quản lý. (18) |

* **Xác định các ca kiểm thử:**

+ Các ca kiểm thử bao phủ các lớp tương đương hợp lệ là: các lớp (1), (8), (15), (16).

+ Các ca kiểm thử tương ứng với từng ca kiểm thử không hợp lệ:

* (2) Để trống ô đăng nhập.
* (3) Tên đăng nhập là “admin” mà nhập “amin” hay “dmin”, “admi”, vv…
* (4) Tên đăng nhập là “admin” mà nhập “admin1” hay “ admina”, vv...
* (5) Tên đăng nhập là “admin” mà nhập “madin” hay “minda”, “12345”,…
* (6) Tên đăng nhập là “admin” mà nhập “ admin” hay “admin “.
* (7) Mật khẩu là “admin” mà nhập “@dmin”,…
* (9) Để trống ô mật khẩu.
* (10) Mật khẩu là “admin” mà nhập “amin” hay “dmin”, “admi”, vv…
* (11) Mật khẩu là “admin” mà nhập “admin1” hay “ admina”, vv...
* (12) Mật khẩu là “admin” mà nhập “madin” hay “minda”, “12345”,…
* (13) Mật khẩu là “admin” mà nhập “ admin” hay “admin “.
* (14) Mật khẩu là “admin” mà nhập “@dmin”,…
* (17) Đăng nhập tài khoản, mật khẩu là admin mà chọn quyền nhân viên.
* (18) Đăng nhập tài khoản, mật khẩu là nhân viên mà chọn quyền quản lý.
* **Phương pháp đồ thị nguyên nhân-kết quả**

Tạo các TeseCase cho chức năng: Đăng nhập.

Từ đặt tả chương trình ta xác định được các nguyên nhân và kết quả sau:

• Nguyên nhân là:

1. Tên đăng nhập là chuỗi kí tự dạng chữ và số có độ dài khác rỗng và <= 50.

2. Mật khẩu là chuỗi kí tự dạng chữ và số có độ dài khác rỗng và độ dài > 6 và <= 20.

3. Quyền đăng nhập là nhân viên nếu là nhân viên hoặc quản lý nếu là quản lý.

• Kết quả là:

R1. Thông báo: Tên tài khoản không được để trống.

R2. Thông báo: Mật khẩu không được để trống.

R3. Lỗi :: Sai tên đăng nhập hoặc mật khẩu Nhân viên.

R4. Kiểm tra thông tin và vào màn hình chính.

1

R1

12

R3

123

3

2

R2

R4

Bảng quyết định:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** |
| **1** | 0 |  | 0 | 1 |
| **2** |  | 0 | 0 | 1 |
| **3** |  |  |  | 1 |
| **R1** | 1 | 0 | 0 | 0 |
| **R2** | 0 | 1 | 0 | 0 |
| **R3** | 0 | 0 | 1 | 0 |
| **R4** | 0 | 0 | 0 | 1 |

Các trường hợp kiểm thử theo phương pháp đồ thị nguyên nhân-kết quả:

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Các điều kiện** | **Ca kiểm thử** | **Hành động** |
| 1 | Tên đăng nhập là chuỗi kí tự dạng chữ và số có độ dài bằng rỗng. | Tên đăng nhập= “”. | **R1** |
| 2 | Mật khẩu là chuỗi kí tự dạng chữ và số có độ dài bằng rỗng. | Mật khẩu= “”. | **R2** |
| 3 | Mật khẩu và Tên đăng nhập là chuỗi kí tự dạng chữ và số:   * Tên đăng nhập có độ dài >50. | Tên đăng nhập= “a…dmin”,… (… là nhiều hơn 46 ký tự) | **R3** |
| * Mật khẩu có độ dài <6 hoặc >20. | Mật khẩu= “a” hoặc Mật khẩu= “a…dmin”,…  (…là nhiều hơn16 ký tự) |
| * Tên đăng nhập hoặc Mật khẩu không đúng trong dữ liệu SQL. | Mật khẩu= “ndmin”,…  hoặc Tên đăng nhập= “ndmin”,… |
| * Tên đăng nhập hoặc Mật khẩu có ký tự space trước hoặc sau. | Tên đăng nhập= “ admin”,… hoặc  Mật khẩu= “ admin”,… |
| 4 | Mật khẩu và Tên đăng nhập đúng . | Tên đăng nhập= “admin”  Mật khẩu= “admin”. | **R4** |

5. Kiểm thử hộp trắng:

**a, Xác định các nút**:

**if** (cbRole.getSelectedIndex() == 1) { **(1)**

**try** {**(2)**

String admin = "select \* from Administrator where Username=? COLLATE SQL\_Latin1\_General\_CP1\_CS\_AS and Password=? COLLATE SQL\_Latin1\_General\_CP1\_CS\_AS";**(3)**

ps = con.prepareStatement(admin); **(4)**

ps.setString(1, txtUser.getText()); **(5)**

ps.setString(2, txtPassword.getText()); **(6)**

rs = ps.executeQuery();**(7)**

**if** (rs.next()) **(8)**{ **(9)**

**new** AdminForm().setVisible(**true**);**(10)**

**this**.setVisible(**false**);**(11)**

} **(12)else** { **(13)**

JOptionPane.showMessageDialog(**null**, "Lỗi:: Sai tên tài khoản hoặc mật khẩu Quản lý.");

txtUser.grabFocus();**(14)**

}**(15)**

} **catch** (SQLException ex) {

JOptionPane.showMessageDialog(**null**, "Lỗi:: Không thể kết nối đến SQL");

}**(16)**

}

**(17)**

**if** (cbRole.getSelectedIndex() == 0) { **(18)**

**try** { **(19)**

String Emp = "select \* from Employee where UsernameEmp=? COLLATE SQL\_Latin1\_General\_CP1\_CS\_AS and Password=? COLLATE SQL\_Latin1\_General\_CP1\_CS\_AS";**(20)**

ps = con.prepareStatement(Emp); **(21)**

ps.setString(1, txtUser.getText());**(22)**

ps.setString(2, txtPassword.getText());**(23)**

rs = ps.executeQuery();**(24)**

**if** (rs.next()) **(25)**{**(26)**

**new** BillForm(rs.getString("NameEmp")).setVisible(**true**);**(27)**

**this**.setVisible(**false**);**(28)**

} **(29)else** {**(30)**

JOptionPane.showMessageDialog(**null**, "Lỗi:: Sai tên tài khoản hoặc mật khẩu Nhân viên.");**(31)**

txtUser.grabFocus();**(32)**

}**(33)**

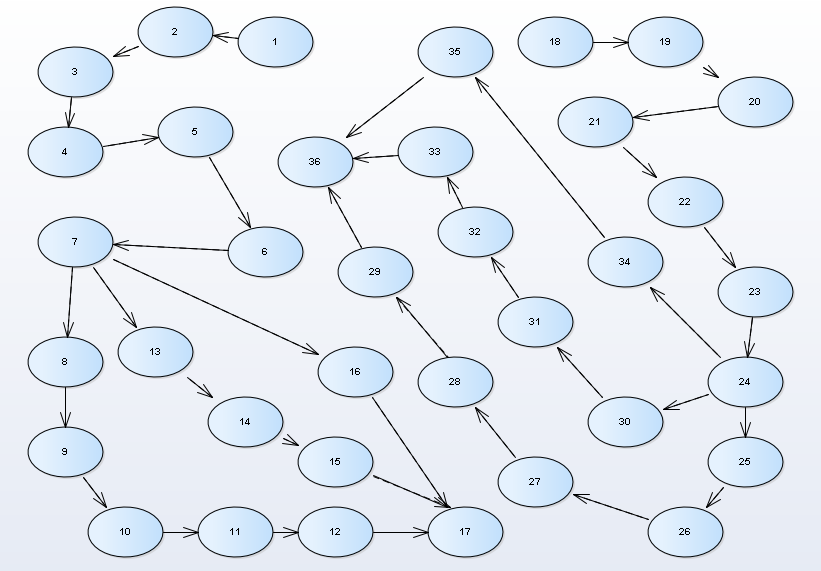
} **catch** (SQLException ex) { **(34)**

JOptionPane.showMessageDialog(**null**, "Lỗi:: Không thể kết nối đến máy chủ");**(35)**

}

} **(36)**

**b, Đồ thị đường diễn tiến:**

****

**c, Xác định số đường kiểm thử**

V(G) = 38 - 36 +2= 4

Các đường kiểm thử là

1. 1.2.3.4.5.6.7.8.9.10.11.12.17
2. 1.2.3.4.5.6.7.13.14.15.17
3. 1.2.3.4.5.6.7.16.17
4. 18.19.20.21.22.23.24.25.26.27.28.29.36
5. 18.19.20.21.22.23.24.30.31.32.33.36
6. 18.19.20.21.22.23.24.34.35.36

**d, Xác định test case**

|  |  |  |
| --- | --- | --- |
| **Đường kiểm thử** | **Giá trị đầu vào** | **Kết quả mong muốn** |
| **1** | txtuser.Text = "admin";  txtpass.Text = "admin"; | “Đăng Nhập Quản lý Thành Công" |
| **2** | txtuser.Text = "123";  txtpass.Text = "123"; | Lỗi:: Sai tên tài khoản hoặc mật khẩu Quản lý |
| **3** | txtuser.Text = "";  txtpass.Text = ""; | Lỗi:: Không thể kết nối đến SQL |
| **4** | txtuser.Text = "nguyenhue";  txtpass.Text = "123456"; | “Đăng Nhập Nhân viên Thành Công" |
| **5** | txtuser.Text = "123";  txtpass.Text = "123"; | Lỗi:: Sai tên tài khoản hoặc mật khẩu nhân viên |
| **6** | txtuser.Text = "";  txtpass.Text = ""; | Lỗi:: Không thể kết nối đến SQL |

## II. Chức năng SetImage cho Button

1. Mô tả chức năng:

- Gán hình ảnh vào một Button

2. Mô tả thiết kế chức năng:



3. Mã nguồn và kiểm tra module chức năng:

4. Kiểm thử hộp đen:

* **Phương pháp tương đương**

|  |  |  |
| --- | --- | --- |
| **Các giá trị đầu vào** | **Lớp tương đương hợp lệ** | **Lớp tương đương không hợp lệ** |
| Kiểm tra khi gõ tên ảnh trực tiếp trên TextBox FileName | * Chiều rộng/Chiều cao khung ảnh > chiều rộng/chiều cao của ảnh. (1) * Chiều rộng/Chiều cao khung ảnh <= chiều rộng/chiều cao của ảnh. (2) | Không có. |
| Kiểm tra khi chọn ảnh và nhấn nút Open | * Chiều rộng/Chiều cao khung ảnh > chiều rộng/chiều cao của ảnh. (3) * Chiều rộng/Chiều cao khung ảnh <= chiều rộng/chiều cao của ảnh. (4) | Không có. |

* **Xác định các lớp tương đương:**
* **Xác định các ca kiểm thử:**

+ Các ca kiểm thử bao phủ các lớp tương đương hợp lệ là: các lớp (1), (2), (3), (4).

+ Các ca kiểm thử tương ứng với từng ca kiểm thử không hợp lệ: Không có.

* **Phương pháp đồ thị nguyên nhân-kết quả**

Tạo các TeseCase cho chức năng: Kiểm tra ảnh.

Từ đặt tả chương trình ta xác định được các nguyên nhân và kết quả sau:

• Nguyên nhân là:

1. File ảnh là file ảnh có trong máy và có các định dạng .png, .jpg, .gif, .tiff và .bmp

* Kết quả là:

R1: Xử lý và đưa ảnh ra màn hình.

Bảng quyết định:

|  |  |
| --- | --- |
|  | 1 |
| 1 | 1 |
| R1 | 1 |

Các trường hợp kiểm thử theo phương pháp đồ thị nguyên nhân-kết quả:

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Các điều kiện** | **Ca kiểm thử** | **Hành động** |
| 1 | File ảnh là file ảnh có trong cơ sở dữ liệu SQL hoặc trong file image của ứng dụng và có các định dạng .png, .jpg, .gif, .tiff và .bmp | File: “1.jpg, 2.jpg”,… | **R1** |

5. Kiểm thử hộp trắng: