BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC NHA TRANG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP NHÓM CÔNG NGHỆ PHẦN MỀM**

**ĐỀ TÀI: QUẢN LÝ QUÁN CÀ PHÊ**

**Sinh viên thực hiện:**

**1. Cao Nguyễn Quốc Lâm**

**2. Trịnh Đăng Khoa**

**3. Lê Minh Thành**

**Lớp : 63.CNTT-3**

**Giảng viên : Bùi Chí Thành**

KHÁNH HÒA – 2024

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Ngày** | **Nội dung** | **Phân công** |
| 1 | 7/03/2024 – 12/03/2024 | Khảo sát và thu thập thông tin | Cả nhóm |
| 2 | 14/03/2024 – 19/03/2024 | Phân tích và đặc tả yêu cầu | * *Phương pháp xác định yêu cầu*: Khoa, Thành * *Đặc tả yêu cầu hệ thống*: Khoa, Lâm |
| 3 | 21/03/2024 – 24/03/2024 | Vẽ biểu mẫu | Cả nhóm |
| 4 | 26/03/2024 – 5/04/2024 | Vẽ sơ đồ luồng dữ liệu | Khoa |
| 5 | 20/4/2024 – 3/05/2024 | Thiết kế phần mềm | * *Thiết kế CSDL:* Khoa Lâm * *Thiết kế giao diện:* Cả nhóm * *Thiết kế xử lý:* Cả nhóm |
| 6 | 5/05/2024 – 25/05/2024 | Cài đặt phần mềm | Cả nhóm |
| 7 | 27/05/2024 – 30/05/2024 | Hoàn thành báo cáo | Khoa, Lâm |

# **BẢNG PHÂN CÔNG THỰC HIỆN BÀI TẬP NHÓM**

# **MỤC LỤC**

[**BẢNG PHÂN CÔNG THỰC HIỆN BÀI TẬP NHÓM** 1](#_Toc167887963)

[**MỤC LỤC** 2](#_Toc167887964)

[**CHƯƠNG 1. ĐẶC TẢ VÀ PHÂN TÍCH YÊU CẦU** 5](#_Toc167887965)

[**1.1. Đặc tả yêu cầu** 5](#_Toc167887966)

[**1.2 Phân tích yêu cầu** 6](#_Toc167887967)

[1.2.1 Danh sách các yêu cầu 6](#_Toc167887968)

[1.2.2 Sơ đồ luồng dữ liệu 19](#_Toc167887969)

[**CHƯƠNG 2: Thiết kế phần mềm** 29](#_Toc167887970)

[**1. Thiết kế dữ liệu** 29](#_Toc167887971)

[1.1. Bộ phận: Quản lý 29](#_Toc167887972)

[1.2. Bộ phận: Quản lý kho 33](#_Toc167887973)

[1.3. Bộ phận: Thu ngân 35](#_Toc167887974)

[1.4. Một số thành phần khác 40](#_Toc167887975)

[**2. Thiết kế giao diện** 41](#_Toc167887976)

[2.1 Danh sách các màn hình 41](#_Toc167887977)

[2.2 Sơ đồ liên kết các màn hình 43](#_Toc167887978)

[2.3 Mô tả chi tiết từng màn hình 44](#_Toc167887979)

[2.3.1. Bộ phận: Quản lý 45](#_Toc167887980)

[2.3.2. Bộ phận: Quản lý kho 65](#_Toc167887981)

[2.3.3. Bộ phận: Thu ngân 73](#_Toc167887982)

[2.3.4. Một số biến cố khác 77](#_Toc167887983)

[**3. Thiết kế xử lý** 78](#_Toc167887984)

[3.1. Bộ phận quản lý 78](#_Toc167887985)

[3.1.1. Thêm, xóa sửa loại sản phẩm 78](#_Toc167887986)

[3.1.2. Tìm kiếm loại sản phẩm 78](#_Toc167887987)

[3.1.3. Thêm, xóa, sửa sản phẩm 78](#_Toc167887988)

[3.1.4. Tìm kiếm sản phẩm 78](#_Toc167887989)

[3.1.4.1. Tìm kiếm sản phẩm theo loại 78](#_Toc167887990)

[3.1.4.2. Tìm kiếm sản phẩm theo tên 78](#_Toc167887991)

[3.1.4.3. Tìm kiếm sản phẩm theo giá 78](#_Toc167887992)

[3.1.5. Xem, xóa và in hóa đơn 78](#_Toc167887993)

[3.1.6. Tìm kiếm hóa đơn 78](#_Toc167887994)

[3.1.7. Xem, xóa và in báo cáo kết ca 78](#_Toc167887995)

[3.1.8. Tìm kiếm báo cáo 78](#_Toc167887996)

[3.1.9. Thêm, xóa, sửa loại người dùng 78](#_Toc167887997)

[3.1.10. Thêm, xóa, sửa người dùng 78](#_Toc167887998)

[3.1.11. Tìm kiếm người dùng 78](#_Toc167887999)

[3.1.11.1. Tìm kiếm người dùng theo tên 78](#_Toc167888000)

[3.1.11.2. Tìm kiếm người dùng theo loại 78](#_Toc167888001)

[3.1.11.3. Tìm kiếm người dùng theo giới tính 78](#_Toc167888002)

[3.1.12. Phân quyền 78](#_Toc167888003)

[3.2. Bộ phận quản lý kho 78](#_Toc167888004)

[3.2.1. Thêm, xóa, sửa loại mặt hàng 78](#_Toc167888005)

[3.2.2. Tìm kiếm loại mặt hàng 78](#_Toc167888006)

[3.2.3. Thêm, xóa, sửa mặt hàng 78](#_Toc167888007)

[3.2.4. Tìm kiếm mặt hàng 78](#_Toc167888008)

[3.2.4.1. Tìm kiếm mặt hàng theo tên 78](#_Toc167888009)

[3.2.4.2. Tìm kiếm mặt hàng theo loại 78](#_Toc167888010)

[3.2.4.3. Tìm kiếm mặt hàng theo đơn vị 78](#_Toc167888011)

[3.2.4.4. Tìm kiếm mặt hàng theo giá 78](#_Toc167888012)

[3.2.4.5. Tìm kiếm mặt hàng theo số lượng 78](#_Toc167888013)

[3.2.5. Thống kê mặt hàng sắp hết 78](#_Toc167888014)

[3.3. Bộ phận thu ngân 78](#_Toc167888015)

[3.3.1. Lập hóa đơn 78](#_Toc167888016)

[3.3.2. Thanh toán 78](#_Toc167888017)

[3.3.3. Tìm kiếm hóa đơn 78](#_Toc167888018)

[3.3.4. Kết ca 78](#_Toc167888019)

[**CHƯƠNG 3: Cài đặt phần mềm** 79](#_Toc167888020)

[**1. Bộ phận quản lý** 79](#_Toc167888021)

[1.1. Số bàn tối đa 79](#_Toc167888022)

[1.2. Thêm, xóa, sửa loại đồ uống 79](#_Toc167888023)

[1.3. Tìm kiếm loại đồ uống 80](#_Toc167888024)

[1.4. Thêm, xóa, sửa đồ uống 80](#_Toc167888025)

[1.5. Tìm kiếm đồ uống 83](#_Toc167888026)

[1.6. Xem, xóa và in hóa đơn 83](#_Toc167888027)

[1.7. Tìm kiếm hóa đơn theo ngày 84](#_Toc167888028)

[1.8. Xem, xóa và in bc kết ca 85](#_Toc167888029)

[1.9. Tìm kiếm báo cáo kết ca theo ngày 86](#_Toc167888030)

[1.10. Thêm xóa sửa loại người dùng 86](#_Toc167888031)

[1.11. Tìm kiếm loại người dùng 88](#_Toc167888032)

[1.12. Thêm xóa sửa người dùng 88](#_Toc167888033)

[1.13. Tìm kiếm người dùng 90](#_Toc167888034)

[1.14. Phân quyền 91](#_Toc167888035)

[**2. Bộ phận quản lý kho** 91](#_Toc167888036)

[2.1. Thêm xóa sửa loại mặt hàng 91](#_Toc167888037)

[2.2. Tìm kiếm loại mặt hàng 93](#_Toc167888038)

[2.3. Thêm xóa sửa mặt hàng 93](#_Toc167888039)

[2.4. Tìm kiếm mặt hàng 94](#_Toc167888040)

[2.5. Thống kê số lượng mặt hàng sắp hết 95](#_Toc167888041)

[**3. Bộ phận thu ngân** 96](#_Toc167888042)

[3.1. Nhập tiền đầu ca 96](#_Toc167888043)

[3.2. Bán hàng 97](#_Toc167888044)

[3.3. Lập hóa đơn tạm tính 100](#_Toc167888045)

[3.4. Thanh toán 101](#_Toc167888046)

[3.5. Kết ca và in báo cáo 101](#_Toc167888047)

[**KẾT LUẬN** 102](#_Toc167888048)

[**TÀI LIỆU THAM KHẢO** 103](#_Toc167888049)

# **CHƯƠNG 1. ĐẶC TẢ VÀ PHÂN TÍCH YÊU CẦU**

## **1.1. Đặc tả yêu cầu**

Hiện tại, quán cà phê “Lá” chỉ mới có phần mềm nhập và xuất hóa đơn. Quán cần một hệ thống mới với các chức năng như quản lý kho, quản lý hóa đơn, quản lý việc bán hàng, thống kê doanh thu và lợi nhuận hàng tháng. Sau khi xác định yêu cầu và khảo sát hiện trạng bằng phương pháp phỏng vấn, quan sát trực tiếp thì nhóm chúng tôi có được những thông tin sau:

* *Quản lý kho:*

- Nhân viên quản lý kho sẽ kiểm tra các mặt hàng sắp hết rồi lập báo cáo các mặt hàng sắp hết đó cho quản lý.

- Khi có mặt hàng nhập về, nhân viên quản lý kho sẽ kiểm tra phiếu nhập hàng với phiếu xuất hàng từ nhà cung cấp. Nếu đúng thì tiến hành xem mặt hàng đã có trên hệ thống chưa, nếu chưa thì nhập vào các mặt hàng mới, nếu có rồi thì cập nhật số lượng mặt hàng đó. Các thông tin của mặt hàng bao gồm: loại mặt hàng, tên mặt hàng, số lượng, đơn vị, giá nhập.

- Vào mỗi ngày nhân viên quản lý kho sẽ kiểm tra và đưa các mặt hàng còn thiếu vào trong quầy pha chế và cập nhật lại số lượng mặt hàng.

* *Quản lý việc bán hàng:*

*-* Khi mỗi lần vào ca nhân viên thu ngân sẽ đếm số tiền có được rồi nhập vào hệ thống (bắt buộc).

*-* Sau khi khách hàng gọi nước nhân viên phục vụ sẽ ghi tên đồ uống, số bàn vào giấy và đưa vào quầy pha chế để làm. Sau khi làm đồ uống xong, nhân viên pha chế sẽ đưa cho nhân viên thu ngân để nhập vào hệ thống.

*-* Nhân viên thu ngân sẽ chọn số bàn trong danh sách số bàn có sẵn trong hệ thống rồi chọn đồ uống trong danh sách đồ uống có sẵn trong hệ thống rồi nhập số lượng của nó (nhân viên có thể tùy chỉnh số lượng khi chưa thanh toán).

*-* Khi khách hàng muốn thanh toán thì nhân viên sẽ in ra hóa đơn tạm tính để đưa cho khách hàng. Sau khi khách hàng thanh toán thì nhân viên sẽ nhấn vào nút thanh toán trên hệ thống thì hệ thống sẽ lưu hóa đơn và xóa hóa đơn đó ra khỏi bàn hiện tại. Thông tin hóa đơn bao gồm: tên các đồ uống, giá bán, số lượng, thành tiền, tổng tiền, ngày lập hóa đơn, tên nhân viên.

*-* Khi hết ca nhân viên thu ngân sẽ bấm vào kết ca trên hệ thống và nhập vào số tiền kết ca (bắt buộc). Sau khi xác nhận hệ thống sẽ in ra báo cáo kết ca và đăng xuất người dùng hiện tại. Sau đó nhân viên thu ngân sẽ đưa báo cáo kết ca cho nhân viên thu ngân tiếp theo. Nhân viên ca cuối sẽ đưa cho quản lý cùng với số tiền bán được.

* Các công việc phải tin học hóa

- Báo cáo kết ca.

- Báo cáo thống kê mặt hàng sắp hết.

* Phân quyền người sử dụng: nhân viên quản lý kho, nhân viên thu ngân, quản lý.

## **1.2 Phân tích yêu cầu**

### 1.2.1 Danh sách các yêu cầu

*Danh sách bảng yêu cầu chức năng nghiệp vụ:*

**Bộ phận: Quản lý**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Loại công việc** | **Quy định/Công thức liên quan** | **Biểu mẫu liên quan** | **Ghi chú** |
| 1 | Quản lý danh sách đồ uống | Lưu trữ | Xem, xóa, sửa các đồ uống. | QL\_BM1 |  |
| 2 | Thêm đồ uống mới | Lưu trữ | Thêm đồ uống với các thông tin sau:  - Tên loại đồ uống  - Tên đồ uống  - Hình ảnh  - Giá bán | QL\_BM2 |  |
| 3 | Tìm kiếm các đồ uống | Tra cứu | Tìm kiếm các đồ uống thông qua tên đồ uống, loại đồ uống, giá bán. | QL\_BM3 |  |
| 4 | Quản lý danh sách nhân viên | Lưu trữ | Xem, xóa, sửa các nhân viên. | QL\_BM4 |  |
| 5 | Thêm nhân viên mới | Lưu trữ | Thêm nhân viên với các thông tin sau:  - Tên nhân viên  - Giới tính  - Số điện thoại  - Loại người dùng (chức vụ)  - Tên tài khoản  - Mật khẩu | QL\_BM5 |  |
| 6 | Tìm kiếm các nhân viên | Tra cứu | Tìm kiếm các nhân viên thông qua tên nhân viên, loại người dùng, giới tính. | QL\_BM6 |  |
| 7 | Quản lý danh sách hóa đơn | Lưu trữ | Xem, xóa các hóa đơn và chi tiết hóa đơn. |  |  |
| 8 | Tìm kiếm các hóa đơn | Tra cứu | Tìm kiếm các hóa đơn thông qua ngày lập hóa đơn. | QL\_BM7 |  |
| 9 | Quản lý danh sách báo cáo kết ca | Lưu trữ | Xem, xóa các báo cáo kết ca và chi tiết báo cáo kết ca. |  |  |
| 10 | Tìm kiếm các báo cáo kết ca | Tra cứu | Tìm kiếm các báo cáo kết ca thông qua ngày kết ca. | QL\_BM8 |  |

**QL\_BM1: Biểu mẫu danh sách đồ uống**

**DANH SÁCH ĐỒ UỐNG**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên đồ uống** | **Loại đồ uống** | **Hình ảnh** | **Giá bán** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**QL\_BM2: Biểu mẫu thêm đồ uống mới**

**THÔNG TIN ĐỒ UỐNG**

**Tên loại đồ uống:………………………………...**

**Tên đồ uống:……………………………………..**

**Hình ảnh:…………………………………………**

**Giá bán:…………………………………………..**

**QL\_BM3: Biểu mẫu tìm kiếm các đồ uống**

**DANH SÁCH ĐỒ UỐNG**

**Đồ uống:……………………. Giá:………………………..**

**Loại đồ uống:………………**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên đồ uống** | **Loại đồ uống** | **Hình ảnh** | **Giá bán** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**QL\_BM4: Biểu mẫu danh sách nhân viên**

**DANH SÁCH NHÂN VIÊN**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Họ và tên** | **Loại người dùng** | **Giới tính** | **Hình ảnh** | **Số điện thoại** | **Tên tài khoản** | **Mật khẩu** |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

**QL\_BM5: Biểu mẫu thêm nhân viên mới**

**THÔNG TIN NHÂN VIÊN**

**Tên người dùng:…………………..**

**Tên loại người dùng:……………...**

**Giới tính:…………………………..**

**Số điện thoại:……………………...**

**Hình ảnh:………………. …………**

**Tên tài khoản:…………………….**

**Mật khẩu:………………………….**

**QL\_BM6: Biểu mẫu tìm kiếm các nhân viên**

**DANH SÁCH NHÂN VIÊN**

**Người dùng:……………………………..**

**Loại người dùng:………………………..**

**Giới tính:………………………………...**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Họ và tên** | **Loại người dùng** | **Giới tính** | **Hình ảnh** | **Số điện thoại** | **Tên tài khoản** | **Mật khẩu** |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

**QL\_BM7: Biểu mẫu tìm kiếm hóa đơn**

**DANH SÁCH ĐỒ UỐNG**

**Ngày lập hóa đơn:…………………….**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Mã hóa đơn** | **Tên nhân viên** | **Ngày lập** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**QL\_BM8: Biểu mẫu tìm kiếm báo cáo kết ca**

**DANH SÁCH BÁO CÁO KẾT CA**

**Ngày kết ca:…………………….**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Mã báo cáo kết ca** | **Tên nhân viên** | **Ngày lập** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Bộ phận: Nhân viên quản lý kho**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Loại công việc** | **Quy định/Công thức liên quan** | **Biểu mẫu liên quan** | **Ghi chú** |
| 1 | Quản lý danh sách mặt hàng | Lưu trữ | Xem, sửa và xóa các mặt hàng. | QLK\_BM1 |  |
| 2 | Thêm mặt hàng mới | Lưu trữ | Thêm mặt hàng với các thông tin sau | QLK\_BM2 |  |
| 3 | Tìm kiếm các mặt hàng | Tra cứu | Tìm kiếm các mặt hàng thông qua tên mặt hàng, loại mặt hàng, số lượng, ngày nhập hàng. | QLK\_BM3 |  |
| 4 | Báo cáo các sản phẩm sắp hết | Kết xuất |  | QLK\_BM4 |  |

**QLK\_BM1: Biểu mẫu danh sách mặt hàng**

**DANH SÁCH MẶT HÀNG**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Tên mặt hàng** | **Tên loại mặt hàng** | **Đơn vị** | **Số lượng** | **Giá nhập** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**QLK\_BM2: Biểu mẫu thêm mặt hàng mới**

**THÔNG TIN MẶT HÀNG**

**Tên loại mặt hàng:………………..**

**Tên mặt hàng:…………………….**

**Đơn vị:…………………………….**

**Số lượng:………………………….**

**Giá nhập:………………………….**

**QLK\_BM3: Biểu mẫu tìm kiếm mặt hàng**

**DANH SÁCH MẶT HÀNG**

**Mặt hàng:……………….. Giá:…………………**

**Loại mặt hàng:………….. Số lượng:……………**

**Đơn vị:…………………...**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Tên mặt hàng** | **Tên loại mặt hàng** | **Đơn vị** | **Số lượng** | **Giá nhập** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**QLK\_BM4: Biểu mẫu báo cáo các sản phẩm sắp hết**

**BÁO CÁO SẢN PHẨM SẮP HẾT**

**LÁ COFFEE**

Địa chỉ:…………………………

Số điện thoại:…………………..

Ngày lập:……………………….

Nhân viên:………………….......

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Tên mặt hàng** | **Tên loại mặt hàng** | **Đơn vị** | **Số lượng** | **Giá nhập** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Bộ phận: Nhân viên thu ngân**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Loại công việc** | **Quy định/Công thức liên quan** | **Biểu mẫu liên quan** | **Ghi chú** |
| 1 | Lập hóa đơn | Lưu trữ | - Thêm các loại đồ uống.  - Được sửa, xóa các đồ uống trong hóa đơn khi chưa thanh toán. |  |  |
| 2 | Xuất hóa đơn tạm tính | Kết xuất | Vẫn có thể chỉnh sửa hóa đơn được sau khi xuất. | TN\_BM1 |  |
| 3 | Thanh toán | Lưu trữ | Sau khi khách hàng thanh toán nhân viên sẽ bấm vào xác nhận thanh toán và lưu hóa đơn vào danh sách hóa đơn. |  |  |
| 4 | Xem danh sách hóa đơn | Tra cứu | Xem và tìm kiếm danh sách hóa đơn, chi tiết hóa đơn. |  |  |
| 5 | Xuất hơn đơn | Kết xuất |  | TN\_BM2 |  |
| 6 | Tính tiền đầu ca và kết ca | Tính toán | - Sau khi đăng nhập vào hệ thống phải nhập số tiền đầu ca.  - Khi kết ca phải nhập số tiền kết ca. |  |  |
| 7 | Lập phiếu kết ca | Kết xuất |  | TN\_BM3 |  |

**TN\_BM1: Biểu mẫu hóa đơn tạm tính**

**HÓA ĐƠN TẠM TÍNH**

**LÁ COFFEE**

Địa chỉ:…………………………

Số điện thoại:…………………..

Mã hóa đơn:……………………

Ngày lập:……………………….

Số bàn:………………………….

Nhân viên:………………….......

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên đồ uống** | **Số lượng** | **Đơn giá** | **Thành tiền** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Tổng tiền:…………………………….

**TN\_BM2: Biểu mẫu hóa đơn**

**HÓA ĐƠN**

**LÁ COFFEE**

Địa chỉ:…………………………

Số điện thoại:…………………..

Mã hóa đơn:……………………

Ngày lập:……………………….

Nhân viên:………………….......

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên đồ uống** | **Số lượng** | **Đơn giá** | **Thành tiền** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Tổng tiền:…………………………….

**TN\_BM3: Biểu mẫu báo cáo kết ca**

**BÁO CÁO KẾT CA**

**LÁ COFFEE**

Mã báo cáo kết ca:………

Nhân viên:……………….

Giờ vào ca:…………… Giờ kết ca:…………..

Tổng số hóa đơn:………………………………………………….

Đang phục vụ:……………………………………………………..

Doanh thu:………………………………………………………...

Tiền đầu ca:……………………………………………………….

Tiền kết ca:………………………………………………………..

Nhân viên giao ca Nhân viên nhận ca

*Bảng yêu cầu chức năng hệ thống*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Nội dung** | **Loại yêu cầu** | **Mô tả chi tiết** | **Ghi chú** |
| 1 | Phân quyền sử dụng | Phân quyền | ***- Quản lý*:** quản lý đồ uống, người dùng, thêm số bàn tối đa, quản lý hóa đơn, báo cáo kết ca và phân quyền.  ***- Nhân viên quản lý kho*:** quản lý mặt hàng, báo các mặt hàng sắp hết.  ***- Nhân viên thu ngân*:** thêm đồ uống vào hóa đơn, chỉnh sửa hóa đơn (khi chưa thanh toán, nếu thanh toán rồi không thể sửa), xem, xuất hóa đơn tạm tính, hóa đơn và báo cáo kết ca. |  |
| 2 | Sao lưu và phục hồi dữ liệu | Sao lưu | Sao lưu dữ liệu vào cơ sở dữ liệu hoặc phục hồi dữ liệu đã xóa trong cơ sở dữ liệu. |  |

*Bảng yêu cầu phi chức năng hệ thống*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Nội dung** | **Loại yêu cầu** | **Mô tả chi tiết** | **Ghi chú** |
| 1 | Giao diện đơn giản, dễ dùng | Tiện dụng | Giao diện với thiết kế đơn giản nhưng đầy đủ các chức năng cùng với các thao tác dễ dàng. |  |
| 2 | Tốc độ tra cứu đồ uống nhanh | Hiệu quả | Tra cứu đồ uống không quá 10 giây. |  |
| 3 | Các giao diện nhất quán | Tương thích | Các giao diện lưu trữ dữ liệu có cùng kiểu tổ chức. |  |
| 4 | Cho phép thay đổi số bàn tối đa trên hệ thống | Tiến hóa | Quản lý có thể thay đổi số bàn tối đa hiển thị trên hệ thống tùy theo nhu cầu của quán. |  |
| 5 | Dễ dàng bảo trì hệ thống | Bảo trì | Các dữ liệu có thể thay đổi sẽ để độc lập với các dữ liệu cố định. |  |
| 6 | Các giao diện hệ thống có cùng một layout | Tái sử dụng | Lập trình viên nếu muốn thêm 1 giao diện mới có thể sử dụng layout có sẵn trong hệ thống. |  |

### 1.2.2 Sơ đồ luồng dữ liệu

**Bộ phận: Quản lý**

*Yêu cầu thêm đồ uống mới (chức năng lưu trữ)*

|  |  |
| --- | --- |
| Quản lý  D4  1  D3  1  D2  1  D1  1 | - D1: Thông tin đồ uống bao gồm: tên đồ uống, loại đồ uống, hình ảnh, giá bán.  - D2: Danh sách đồ uống  - D3: Kết quả là thông tin đồ uống cùng với mã đồ uống.  - D4: Danh sách mã loại đồ uống. |
| Các bước xử lý:  B1: Nhận dữ liệu D1 từ quản lý.  B2: Kết nối cơ sở dữ liệu.  B3: Đọc D4 từ bộ nhớ phụ.  B4: Lưu D3 xuống bộ nhớ phụ.  B5: Hiển thị D2.  B6: Đóng kết nối cơ sở dữ liệu và kết thúc. | |

*Yêu cầu thêm nhân viên mới (chức năng lưu trữ)*

|  |  |
| --- | --- |
| Quản lý  D4  1  D3  1  D2  1  D1  1 | - D1: Thông tin nhân viên bao gồm: tên nhân viên, loại người dùng, hình ảnh, giới tính, số điện thoại, tên tài khoản, mật khẩu.  - D2: Danh sách nhân viên.  - D3: Kết quả là thông tin nhân viên cùng với mã nhân viên.  - D4: Danh sách mã loại người dùng. |
| Các bước xử lý:  B1: Nhận dữ liệu D1 từ quản lý.  B2: Kết nối cơ sở dữ liệu.  B3: Đọc D4 từ bộ nhớ phụ.  B4: Kiểm tra xem tên tài khoản đã tồn tại trong cơ sở dữ liệu chưa, nếu chưa thì xuống B6.  B4: Lưu D3 xuống bộ nhớ phụ.  B5: Hiển thị D2.  B6: Đóng cơ sở dữ liệu và kết thúc. | |

*Yêu cầu tìm kiếm đồ uống (chức năng tra cứu)*

|  |  |
| --- | --- |
| Quản lý  D4  1  D3  1  D2  1  D1  1 | - D1: Thông tin đồ uống cần tra cứu: tên đồ uống, loại đồ uống, giá bán.  - D2: Danh sách đồ uống (Nếu không có sẽ không hiển thị dữ liệu).  - D3: Không có.  - D4: Danh sách đồ uống được tìm thấy. |
| Các bước xử lý:  B1: Kết nối cơ sở dữ liệu  B2: Đọc D4 (Danh sách đồ uống)  B3: Hiển thị D2 (Danh sách đồ uống).  B4: Nhận dữ liệu D1 từ quản lý.  B5: Tìm D4 theo D1 (Danh sách đồ uống được tìm thấy).  B6: Hiển thị D2 (Danh sách đồ uống cần tra cứu).  B7: Đóng cơ sở dữ liệu và kết thúc. | |

*Yêu cầu tìm kiếm nhân viên (chức năng tra cứu)*

|  |  |
| --- | --- |
| Quản lý  D4  1  D3  1  D2  1  D1  1 | - D1: Thông tin người dùng cần tra cứu: tên nhân viên, loại người dùng, giới tính.  - D2: Danh sách người dùng (Nếu không có sẽ không hiển thị dữ liệu).  - D3: Không có.  - D4: Danh sách người dùng được tìm thấy. |
| Các bước xử lý:  B1: Kết nối cơ sở dữ liệu.  B2: Đọc D4 (Danh sách người dùng).  B3: Hiển thị D2 (Danh sách người dùng).  B4: Nhận dữ liệu D1 từ quản lý.  B5: Tìm D4 theo D1 (Danh sách người dùng được tìm thấy).  B6: Hiển thị D2 (Danh sách người dùng cần tra cứu).  B7: Đóng cơ sở dữ liệu và kết thúc. | |

*Yêu cầu tìm kiếm hóa đơn (chức năng tra cứu)*

|  |  |
| --- | --- |
| Quản lý  D4  1  D3  1  D2  1  D1  1 | - D1: Ngày lập hóa đơn.  - D2: Danh sách hóa đơn (Nếu không có sẽ không hiển thị dữ liệu).  - D3: Không có.  - D4: Danh sách hóa đơn được tìm thấy. |
| Các bước xử lý:  B1: Kết nối cơ sở dữ liệu.  B2: Đọc D4 (Danh sách hóa đơn).  B3: Hiển thị D2 (Danh sách hóa đơn).  B4: Nhận dữ liệu D1 từ quản lý.  B5: Tìm D4 theo D1 (Danh sách hóa đơn được tìm thấy).  B6: Hiển thị D2 (Danh sách hóa đơn cần tra cứu).  B7: Đóng cơ sở dữ liệu và kết thúc. | |

*Yêu cầu tìm kiếm báo cáo kết ca (chức năng tra cứu)*

|  |  |
| --- | --- |
| Quản lý  D4  1  D3  1  D2  1  D1  1 | - D1: Ngày lập báo cáo kết ca.  - D2: Danh sách báo cáo kết ca (Nếu không có sẽ không hiển thị dữ liệu).  - D3: Không có.  - D4: Danh sách báo cáo kết ca được tìm thấy. |
| Các bước xử lý:  B1: Kết nối cơ sở dữ liệu.  B2: Đọc D4 (Danh sách báo cáo kết ca).  B3: Hiển thị D2 (Danh sách báo cáo kết ca).  B4: Nhận dữ liệu D1 từ quản lý.  B5: Tìm D4 theo D1 (Danh sách báo cáo kết ca được tìm thấy).  B6: Hiển thị D2 (Danh sách báo cáo kết ca cần tra cứu).  B7: Đóng cơ sở dữ liệu và kết thúc. | |

**Bộ phận: Quản lý kho**

*Yêu cầu thêm mặt hàng mới (chức năng lưu trữ)*

|  |  |
| --- | --- |
| Quản lý kho  D4  1  D3  1  D2  1  D1  1 | - D1: Thông tin mặt hàng bao gồm: tên mặt hàng, loại mặt hàng, giá nhập, số lượng, đơn vị.  - D2: Danh sách mặt hàng.  - D3: Kết quả là thông tin mặt hàng cùng với mã mặt hàng.  - D4: Danh sách mã loại mặt hàng. |
| Các bước xử lý:  B1: Nhận dữ liệu D1 từ quản lý kho.  B2: Kết nối cơ sở dữ liệu.  B3: Đọc D4 từ bộ nhớ phụ.  B4: Lưu D3 xuống bộ nhớ phụ.  B5: Hiển thị D2.  B6: Đóng kết nối cơ sở dữ liệu và kết thúc. | |

*Yêu cầu tìm kiếm mặt hàng (chức năng tra cứu)*

|  |  |
| --- | --- |
| Quản lý  D4  1  D3  1  D2  1  D1  1 | - D1: Thông tin mặt hàng cần tra cứu: tên mặt hàng, loại mặt hàng, giá, số lượng, đơn vị.  - D2: Danh sách mặt hàng (Nếu không có sẽ không hiển thị dữ liệu).  - D3: Không có.  - D4: Danh sách mặt hàng được tìm thấy. |
| Các bước xử lý:  B1: Kết nối cơ sở dữ liệu.  B2: Đọc D4 (Danh sách mặt hàng).  B3: Hiển thị D2 (Danh sách mặt hàng).  B4: Nhận dữ liệu D1 từ quản lý kho.  B5: Tìm D4 theo D1 (Danh sách mặt hàng được tìm thấy).  B6: Hiển thị D2 (Danh sách mặt hàng cần tra cứu).  B7: Đóng cơ sở dữ liệu và kết thúc. | |

**Bộ phận: Thu ngân**

*Yêu cầu lập hóa đơn (chức năng kết xuất và lưu trữ)*

|  |  |
| --- | --- |
| Thu ngân  Máy in  D4  1  D3  1  D2  1  D1  1 | - D1: Thông tin hóa đơn bao gồm: số lượng đồ uống, tên đồ uống, giá bán, thành tiền, tổng tiền.  - D2: Thông tin hóa đơn cùng với thông tin người lập hóa đơn.  - D3: Kết quả là thông tin hóa đơn cùng với mã hóa đơn.  - D4: Danh sách các đồ uống.  - D5: D2 |
| Các bước xử lý:  B1: Nhận dữ liệu D1 từ thu ngân.  B2: Kết nối cơ sở dữ liệu.  B3: Đọc D4 từ bộ nhớ phụ.  B4: Tính toán tổng tiền và thành tiền.  B5: Lưu D3 xuống bộ nhớ phụ.  B6: Hiển thị D2.  B7: Xuất D5.  B8: Đóng kết nối cơ sở dữ liệu và kết thúc. | |

*Yêu cầu lập báo cáo kết ca (chức năng kết xuất và lưu trữ)*

|  |  |
| --- | --- |
| Thu ngân  Máy in  D4  1  D3  1  D2  1  D1  1 | - D1: Số tiền đầu ca và kết ca  - D2: Thông tin báo cáo kết ca cùng với thông tin người lập hóa đơn.  - D3: Kết quả là thông tin báo cáo kết ca cùng với mã báo cáo kết ca.  - D4: Danh sách các hóa đơn và danh sách bàn đang có người.  - D5: D2 |
| Các bước xử lý:  B1: Nhận dữ liệu D1 từ thu ngân.  B2: Kết nối cơ sở dữ liệu.  B3: Đọc D4 từ bộ nhớ phụ.  B4: Tính toán doanh thu, số hóa đơn và số tiền đang phục vụ.  B5: Lưu D3 xuống bộ nhớ phụ.  B6: Hiển thị D2.  B7: Xuất D5.  B8: Đóng kết nối cơ sở dữ liệu và kết thúc. | |

# **CHƯƠNG 2: Thiết kế phần mềm**

## **1. Thiết kế dữ liệu**

Nhóm sử dụng SQL Server và phầm mềm Microsoft SQL Server Mangament Studio để thiết kế cơ sở dữ liệu cho hệ thống.

### 1.1. Bộ phận: Quản lý

*Yêu cầu quản lý đồ uống và tìm kiếm đồ uống*

A screenshot of a computer

Description automatically generated

Thành phần: DrinkCategories

Ý nghĩa: Lưu trữ thông tin về loại đồ uống

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | DrinkCategoryID | int | > 0 | Mã loại đồ uống |
| 2 | DrinkCategoryName | nvarchar | MAX | Tên loại đồ uống |
| 3 | DrinkCategoryDesciption | nvarchar | MAX | Mô tả |

Thành phần: Drinks

Ý nghĩa: Lưu trữ thông tin về đồ uống

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | DrinkID | int | > 0 | Mã đồ uống |
| 2 | DrinkCategoryID | int | > 0 | Mã loại đồ uống |
| 3 | DrinkName | nvarchar | MAX | Tên đồ uống |
| 4 | DrinkImage | nvarchar | MAX | Hình ảnh |
| 5 | DrinkPrice | decimal | > 0 | Giá bán |

*Yêu cầu quản lý người dùng và tìm kiếm người dùng*

*A screenshot of a computer

Description automatically generated*

Thành phần: UserCategories

Ý nghĩa: Lưu trữ thông tin về loại người dùng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | UserCategoryID | int | > 0 | Mã loại người dùng |
| 2 | UserCategoryName | nvarchar | MAX | Tên loại người dùng |
| 3 | UserCategoryDescription | nvarchar | MAX | Mô tả |

Thành phần: User

Ý nghĩa: Lưu trữ thông tin về người dùng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | UserID | int | > 0 | Mã người dùng |
| 2 | FullName | nvarchar | MAX | Họ và tên |
| 3 | PhoneNumber | nvarchar | MAX | Số điện thoại |
| 4 | Gender | nvarchar | MAX | Giới tính |
| 5 | AccountName | nvarchar | Tối đa 25 kí tự | Tên tài khoản |
| 6 | Password | nvarchar | Tối đa 10 kí tự | Mật khẩu |
| 7 | UserImage | nvarchar | MAX | Hình ảnh |
| 8 | UserCategoryID | int | > 0 | Mã loại người dùng |

*Yêu cầu quản lý hóa đơn và tìm kiếm hóa đơn*

*A screenshot of a computer

Description automatically generated*

Thành phần: Bills

Ý nghĩa: Lưu trữ thông tin về hóa đơn

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | BillID | nvarchar | Tối đa 128 kí tự | Mã hóa đơn |
| 2 | CreateDate | datetime |  | Ngày lập hóa đơn |
| 3 | TotalAmount | nvarchar | MAX | Tổng tiền |
| 4 | TableID | int | > 0 | Mã bàn |
| 5 | UserID | int | > 0 | Mã người dùng |

Thành phần: BillInfos

Ý nghĩa: Lưu trữ thông tin về chi tiết hóa đơn

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | BillID | nvarchar | Tối đa 128 kí tự | Mã hóa đơn |
| 2 | DrinkID | int | > 0 | Mã đồ uống |
| 3 | DrinkCount | int | > 0 | Số lượng đồ uống |
| 4 | DrinkPrice | Decimal | > 0 | Thành tiền |

*Yêu cầu quản lý báo cáo kết ca và tìm kiếm báo cáo kết ca*

*A screenshot of a computer

Description automatically generated*

Thành phần: ShiftReports

Ý nghĩa: Lưu trữ thông tin về báo cáo kết ca

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | ShiftReportID | nvarchar | Tối đa 128 kí tự | Mã báo cáo kết ca |
| 2 | Revenue | decimal | > 0 | Doanh thu |
| 3 | FirstAmount | decimal | > 0 | Số tiền đầu ca |
| 4 | LastAmount | Decimal | > 0 | Số tiền kết ca |
| 5 | BillCount | int | > 0 | Số hóa đơn |
| 6 | UncollectedAmount | decimal | > 0 | Số tiền đang phục vụ |
| 7 | UserID | int | > 0 | Mã người dùng |
| 8 | FirstTime | datetime |  | Thời gian vào ca |
| 9 | LastTime | datetime |  | Thời gian kết ca |

### 1.2. Bộ phận: Quản lý kho

*Yêu cầu quản lý mặt hàng và tìm kiếm mặt hàng*

A screenshot of a computer

Description automatically generated

Thành phần: GoodsCategories

Ý nghĩa: Lưu trữ thông tin về loại mặt hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | GoodsCategoryID | int | > 0 | Mã loại mặt hàng |
| 2 | GoodsCategoryName | nvarchar | MAX | Tên loại mặt hàng |
| 3 | GoodsCategoryDesciption | nvarchar | MAX | Mô tả |

Thành phần: Goods

Ý nghĩa: Lưu trữ thông tin về đồ uống

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | GoodsID | int | > 0 | Mã mặt hàng |
| 2 | GoodsCategoryID | int | > 0 | Mã loại mặt hàng |
| 3 | GoodsName | nvarchar | MAX | Tên mặt hàng |
| 4 | GoodsCount | int | > 0 | Số lượng |
| 5 | GoodsPrice | decimal | > 0 | Giá nhập |
| 6 | GoodsUnit | nvarchar | MAX | Đơn vị |

### 1.3. Bộ phận: Thu ngân

*A screenshot of a computer

Description automatically generatedYêu cầu lập hóa đơn*

Thành phần: DrinkCategories

Ý nghĩa: Lưu trữ thông tin về loại đồ uống

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | DrinkCategoryID | int | > 0 | Mã loại đồ uống |
| 2 | DrinkCategoryName | nvarchar | MAX | Tên loại đồ uống |
| 3 | DrinkCategoryDesciption | nvarchar | MAX | Mô tả |

Thành phần: Drinks

Ý nghĩa: Lưu trữ thông tin về đồ uống

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | DrinkID | int | > 0 | Mã đồ uống |
| 2 | DrinkCategoryID | int | > 0 | Mã loại đồ uống |
| 3 | DrinkName | nvarchar | MAX | Tên đồ uống |
| 4 | DrinkImage | nvarchar | MAX | Hình ảnh |
| 5 | DrinkPrice | decimal | > 0 | Giá bán |

Thành phần: User

Ý nghĩa: Lưu trữ thông tin về người dùng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | UserID | int | > 0 | Mã người dùng |
| 2 | FullName | nvarchar | MAX | Họ và tên |
| 3 | PhoneNumber | nvarchar | MAX | Số điện thoại |
| 4 | Gender | nvarchar | MAX | Giới tính |
| 5 | AccountName | nvarchar | Tối đa 25 kí tự | Tên tài khoản |
| 6 | Password | nvarchar | Tối đa 10 kí tự | Mật khẩu |
| 7 | UserImage | nvarchar | MAX | Hình ảnh |
| 8 | UserCategoryID | int | > 0 | Mã loại người dùng |

Thành phần: Bills

Ý nghĩa: Lưu trữ thông tin về hóa đơn

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | BillID | nvarchar | Tối đa 128 kí tự | Mã hóa đơn |
| 2 | CreateDate | datetime |  | Ngày lập hóa đơn |
| 3 | TotalAmount | nvarchar | MAX | Tổng tiền |
| 4 | TableID | int | > 0 | Mã bàn |
| 5 | UserID | int | > 0 | Mã người dùng |

Thành phần: BillInfos

Ý nghĩa: Lưu trữ thông tin về chi tiết hóa đơn

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | BillID | nvarchar | Tối đa 128 kí tự | Mã hóa đơn |
| 2 | DrinkID | int | > 0 | Mã đồ uống |
| 3 | DrinkCount | int | > 0 | Số lượng đồ uống |
| 4 | DrinkPrice | Decimal | > 0 | Thành tiền |

Thành phần: Tables

Ý nghĩa: Lưu trữ thông tin về bàn

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | TableID | int | > 0 | Mã bàn |
| 2 | TableName | nvarchar | MAX | Tên bàn |
| 3 | Status | bit | 1 và 0 | Trạng thái |

*Yêu cầu lập báo cáo kết ca*

*A screenshot of a computer

Description automatically generated*

Thành phần: User

Ý nghĩa: Lưu trữ thông tin về người dùng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | UserID | int | > 0 | Mã người dùng |
| 2 | FullName | nvarchar | MAX | Họ và tên |
| 3 | PhoneNumber | nvarchar | MAX | Số điện thoại |
| 4 | Gender | nvarchar | MAX | Giới tính |
| 5 | AccountName | nvarchar | Tối đa 25 kí tự | Tên tài khoản |
| 6 | Password | nvarchar | Tối đa 10 kí tự | Mật khẩu |
| 7 | UserImage | nvarchar | MAX | Hình ảnh |
| 8 | UserCategoryID | int | > 0 | Mã loại người dùng |

Thành phần: Bills

Ý nghĩa: Lưu trữ thông tin về hóa đơn

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | BillID | nvarchar | Tối đa 128 kí tự | Mã hóa đơn |
| 2 | CreateDate | datetime |  | Ngày lập hóa đơn |
| 3 | TotalAmount | nvarchar | MAX | Tổng tiền |
| 4 | TableID | int | > 0 | Mã bàn |
| 5 | UserID | int | > 0 | Mã người dùng |

Thành phần: ShiftReports

Ý nghĩa: Lưu trữ thông tin về báo cáo kết ca

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | ShiftReportID | nvarchar | Tối đa 128 kí tự | Mã báo cáo kết ca |
| 2 | Revenue | decimal | > 0 | Doanh thu |
| 3 | FirstAmount | decimal | > 0 | Số tiền đầu ca |
| 4 | LastAmount | Decimal | > 0 | Số tiền kết ca |
| 5 | BillCount | int | > 0 | Số hóa đơn |
| 6 | UncollectedAmount | decimal | > 0 | Số tiền đang phục vụ |
| 7 | UserID | int | > 0 | Mã người dùng |
| 8 | FirstTime | datetime |  | Thời gian vào ca |
| 9 | LastTime | datetime |  | Thời gian kết ca |

Thành phần: Tables

Ý nghĩa: Lưu trữ thông tin về bàn

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | TableID | int | > 0 | Mã bàn |
| 2 | TableName | nvarchar | MAX | Tên bàn |
| 3 | Status | bit | 1 và 0 | Trạng thái |

### 1.4. Một số thành phần khác

Thành phần: Authorizations

Ý nghĩa: Lưu trữ thông tin về phần quyền

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | UserCategoryID | int | > 0 | Mã loại người dùng |
| 2 | FunctionID | nvarchar | Tối đa 128 kí tự | Mã chức năng |
| 3 | AuthorizationDescription | nvarchar | MAX | Mô tả |

Thành phần: FunctionID

Ý nghĩa: Lưu trữ thông tin về bàn

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu | Miền giá trị | Ý nghĩa |
| 1 | FunctionID | int | > 0 | Mã chức năng |
| 2 | FunctionName | nvarchar | MAX | Tên chức năng |
| 3 | FunctionGroup | nvarchar | MAX | Nhóm chức năng |

## **2. Thiết kế giao diện**

### 2.1 Danh sách các màn hình

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Màn hình | Loại màn hình | Chức năng |
| 1 | Màn hình đăng nhập | Màn hình nhập liệu | Đăng nhập vào hệ thống và lưu thông tin phiên người dùng |
| 2 | Màn hình bán hàng | Màn hình nhập liệu | Cho phép nhập số lượng bàn |
| 3 | Màn hình danh sách đồ uống | Màn hình tra cứu | Cho phép nhập thông tin đồ uống cần tra cứu và hiển thị kết quả tra cứu được |
| 4 | Màn hình thêm đồ uống | Màn hình nhập liệu | Cho phép nhập và lưu trữ thông tin đồ uống |
| 5 | Màn hình chỉnh sửa đồ uống | Màn hình nhập liệu | Cho phép chỉnh sửa thông tin đồ uống |
| 6 | Màn hình danh sách loại đồ uống | Màn hình tra cứu | Cho phép nhập thông tin loại đồ uống cần tra cứu và hiển thị kết quả tra cứu được |
| 7 | Màn hình thêm loại đồ uống | Màn hình nhập liệu | Cho phép nhập và lưu trữ thông tin loại đồ uống |
| 8 | Màn hình chỉnh sửa loại đồ uống | Màn hình nhập liệu | Cho phép chỉnh sửa thông tin loại đồ uống |
| 9 | Màn hình danh sách người dùng | Màn hình tra cứu | Cho phép nhập thông tin người dùng cần tra cứu và hiển thị kết quả tra cứu được |
| 10 | Màn hình thêm người dùng | Màn hình nhập liệu | Cho phép nhập và lưu trữ thông tin người dùng |
| 11 | Màn hình chỉnh sửa người dùng | Màn hình nhập liệu | Cho phép chỉnh sửa thông tin người dùng |
| 12 | Màn hình danh sách loại người dùng | Màn hình tra cứu | Cho phép nhập thông tin loại người dùng cần tra cứu và hiển thị kết quả tra cứu được |
| 13 | Màn hình thêm loại người dùng | Màn hình nhập liệu | Cho phép nhập và lưu trữ thông tin loại người dùng |
| 14 | Màn hình chỉnh sửa loại người dùng | Màn hình nhập liệu | Cho phép chỉnh sửa thông tin loại người dùng |
| 15 | Màn hình danh sách hóa đơn | Màn hình tra cứu | Cho phép nhập và lưu trữ thông tin hóa đơn |
| 16 | Màn hình chi tiết hóa đơn | Màn hình tra cứu | Xem chi tiết hóa đơn và in chi tiết hóa đơn |
| 17 | Màn hình danh sách báo cáo kết ca | Màn hình tra cứu | Cho phép nhập và lưu trữ thông tin báo cáo kết ca |
| 18 | Màn hình chi tiết báo cáo kết ca | Màn hình tra cứu | Xem chi tiết hóa đơn và in chi tiết báo cáo kết ca |
| 19 | Màn hình danh sách phân quyền | Màn hình tra cứu | Cho phép nhập thông tin loại người dùng cần tra cứu và hiển thị kết quả tra cứu được |
| 20 | Màn hình phân quyền | Màn hình lưu trữ | Cho phép thêm các quyền cho từng loại người dùng |
| 21 | Màn hình danh sách mặt hàng | Màn hình tra cứu | Cho phép nhập thông tin mặt hàng cần tra cứu và hiển thị kết quả tra cứu được |
| 22 | Màn hình thêm mặt hàng | Màn hình lưu trữ | Cho phép nhập và lưu trữ thông tin mặt hàng |
| 23 | Màn hình chỉnh sửa mặt hàng | Màn hình lưu trữ | Cho phép chỉnh sửa thông tin mặt hàng |
| 24 | Màn hình danh sách loại mặt hàng | Màn hình tra cứu | Cho phép nhập thông tin loại mặt hàng cần tra cứu và hiển thị kết quả tra cứu được |
| 25 | Màn hình thêm loại mặt hàng | Màn hình lưu trữ | Cho phép nhập và lưu trữ thông tin loại mặt hàng |
| 26 | Màn hình chỉnh sửa loại mặt hàng | Màn hình lưu trữ | Cho phép chỉnh sửa thông tin loại mặt hàng |
| 27 | Màn hình lập hóa đơn | Màn hình lưu trữ | Cho phép thêm các đồ uống vào hóa đơn |
| 28 | Màn hình lập báo cáo kết ca | Màn hình lữu trữ | Cho phép nhập số tiền đầu ca và kết ca vào báo cáo kết ca |

### 2.2 Sơ đồ liên kết các màn hình

A diagram of a company

Description automatically generated

### 2.3 Mô tả chi tiết từng màn hình

*\* Màn hình đăng nhập*A screenshot of a login

Description automatically generated

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | accountName | text | Tối thiểu 2 kí tự | Nhập tên tài khoản |
| 2 | password | text | Tối thiểu 4 kí tự | Nhập mật khẩu |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Đăng nhập | Nếu đúng tên tài khoản và mật khẩu thì đăng nhập vào hệ thống của từng loại người dùng khác nhau (có 3 loại người dùng Quản lý, Thu ngân và Quản lý kho) nếu không thì báo lỗi. |

#### 2.3.1. Bộ phận: Quản lý

*A screenshot of a computer

Description automatically generated\* Màn hình bán hàng*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | tableCount | number | > 0 | Nhập số bàn muốn hiển thị trên hệ thống |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Lưu | Tạo ra danh sách bàn có độ dài bằng số bàn nhập vào và lưu vào hệ thống |

*A screenshot of a computer

Description automatically generated\* Màn hình danh sách đồ uống*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | drinkName | text |  | Nhập tên đồ uống cần tìm |
| 2 | drinkCategoryName | text |  | Nhập tên loại đồ uống cần tìm |
| 3 | minPrice | number | > 0 | Nhập giá bán nhỏ nhất cần tìm |
| 4 | maxPrice | number | > 0 | Nhập giá bán lớn nhất cần tìm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Thêm | Chuyển qua màn hình thêm đồ uống |
| 2 | Chọn nút Xóa nhiều | Xóa các đồ uống đã được chọn |
| 3 | Chọn nút Xóa | Xóa đồ uống tương ứng |
| 4 | Chọn nút Chỉnh sửa | Chuyển qua màn hình chỉnh sửa đồ uống |
| 5 | Chọn nút Tìm kiếm | Hiển thị danh sách các đồ uống theo thông tin tìm kiếm |
| 6 | Chọn nút Load trang | Xóa hết các nội dung tìm kiếm và hiển thị lại danh sách toàn bộ đồ uống |

*A screenshot of a computer

Description automatically generated\* Màn hình thêm đồ uống*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | drinkName | text |  | Nhập tên đồ uống cần thêm |
| 2 | drinkCategoryName | select |  | Chọn loại đồ uống cần thêm trong danh sách loại đồ uống |
| 3 | image | file |  | Thêm vào file hình ảnh |
| 4 | drinkPrice | number | > 0 | Nhập giá bán cần thêm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Lưu | Lưu thông tin đồ uống cần thêm vào cơ sở dữ liệu |
| 2 | Chọn nút Quay lại | Chuyển về màn hình danh sách đồ uống |

*A screenshot of a computer

Description automatically generated\* Màn hình chỉnh sửa đồ uống*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | drinkName | text |  | Nhập tên đồ uống cần chỉnh sửa |
| 2 | drinkCategoryName | select |  | Chọn loại đồ uống cần chỉnh sửa trong danh sách loại đồ uống |
| 3 | image | file |  | Thêm mới file hình ảnh |
| 4 | drinkPrice | number | > 0 | Nhập giá bán cần chỉnh sửa |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Lưu | Cập nhật lại thông tin đồ uống mới trên cơ sở dữ liệu |
| 2 | Chọn nút Quay lại | Chuyển về màn hình danh sách đồ uống |

*A screenshot of a computer

Description automatically generated\* Màn hình danh sách loại đồ uống*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | drinkCategoryName | text |  | Nhập tên loại đồ uống cần tìm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Thêm | Chuyển qua màn hình thêm loại đồ uống |
| 2 | Chọn nút Xóa nhiều | Xóa các loại đồ uống đã được chọn |
| 3 | Chọn nút Xóa | Xóa loại đồ uống tương ứng |
| 4 | Chọn nút Chỉnh sửa | Chuyển qua màn hình chỉnh sửa loại đồ uống |
| 5 | Chọn nút Tìm kiếm | Hiển thị danh sách các loại đồ uống theo thông tin tìm kiếm |
| 6 | Chọn nút Load trang | Xóa hết các nội dung tìm kiếm và hiển thị lại danh sách toàn bộ loại đồ uống |

*A screenshot of a computer

Description automatically generated\* Màn hình thêm loại đồ uống*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | drinkCategoryName | text |  | Nhập tên loại đồ uống cần thêm |
| 2 | drinkCategoryDescription | text |  | Nhập mô tả loại đồ uống cần thêm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Lưu | Lưu thông tin loại đồ uống cần thêm vào cơ sở dữ liệu |
| 2 | Chọn nút Quay lại | Chuyển về màn hình danh sách loại đồ uống |

*A screenshot of a computer

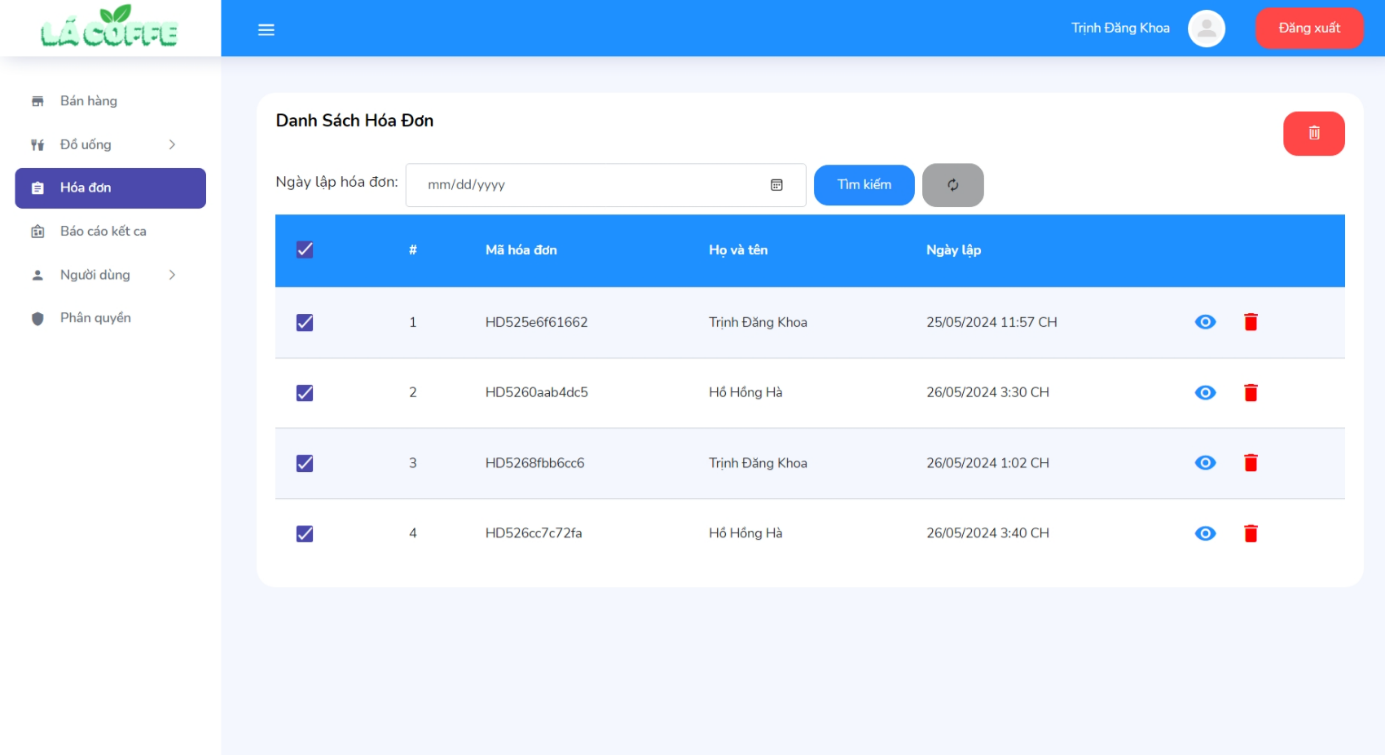
Description automatically generated\* Màn hình chỉnh sửa loại đồ uống*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | drinkCategoryName | text |  | Nhập tên loại đồ uống cần chỉnh sửa |
| 2 | drinkCategoryDescription | text |  | Nhập mô tả loại đồ uống cần chỉnh sửa |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Lưu | Cập nhật lại thông tin loại đồ uống mới trên cơ sở dữ liệu |
| 2 | Chọn nút Quay lại | Chuyển về màn hình danh sách loại đồ uống |

*\* Màn hình danh sách hóa đơn*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | createDate | date |  | Nhập số ngày lập hóa đơn cần tìm kiếm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Xem chi tiết | Chuyển sang màn hình chi tiết hóa đơn |
| 2 | Chọn nút Xóa nhiều | Xóa các hóa đơn đã được chọn |
| 3 | Chọn nút Xóa | Xóa hóa đơn tương ứng |
| 2 | Chọn nút Tìm kiếm | Hiển thị danh sách các hóa đơn theo thông tin tìm kiếm |
| 3 | Chọn nút Load trang | Xóa hết các nội dung tìm kiếm và hiển thị lại danh sách toàn bộ hóa đơn |

*A screenshot of a computer

Description automatically generated\* Màn hình chi tiết hóa đơn*

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút In | In thông tin hóa đơn (Xuất file pdf về máy) |
| 2 | Chọn nút Quay về | Chuyển về màn hình danh sách hóa đơn |

*A screenshot of a computer

Description automatically generated\* Màn hình danh sách báo cáo kết ca*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | lastTime | date |  | Nhập số ngày kết ca cần tìm kiếm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Xem chi tiết | Chuyển sang màn hình chi tiết báo cáo kết ca |
| 2 | Chọn nút Xóa nhiều | Xóa các báo cáo kết ca đã được chọn |
| 3 | Chọn nút Xóa | Xóa báo cáo kết ca tương ứng |
| 4 | Chọn nút Tìm kiếm | Hiển thị danh sách các báo cáo kết ca theo thông tin tìm kiếm |
| 5 | Chọn nút Load trang | Xóa hết các nội dung tìm kiếm và hiển thị lại danh sách toàn bộ báo cáo kết ca |

*A screenshot of a computer

Description automatically generated\* Màn hình chi tiết báo cáo kết ca*

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút In | In thông tin báo cáo kết ca (Xuất file pdf về máy) |
| 2 | Chọn nút Quay về | Chuyển về màn hình danh sách báo cáo kết ca |

*A screenshot of a computer

Description automatically generated\* Màn hình danh sách người dùng*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | userName | text |  | Nhập tên người dùng cần tìm kiếm |
| 2 | userCatergoyName | text |  | Nhập tên loại người dùng cần tìm kiếm |
| 3 | gender | radio | 2 giá trị Nam và Nữ | Chọn giới tính người dùng cần tìm kiếm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Thêm | Chuyển sang màn hình thêm người dùng |
| 2 | Chọn nút Xóa nhiều | Xóa các người dùng đã được chọn |
| 3 | Chọn nút Xóa | Xóa người dùng tương ứng |
| 4 | Chọn nút Tìm kiếm | Hiển thị danh sách các người dùng theo thông tin tìm kiếm |
| 5 | Chọn nút Load trang | Xóa hết các nội dung tìm kiếm và hiển thị lại danh sách toàn bộ người dùng |

*A screenshot of a computer

Description automatically generated\* Màn hình thêm người dùng*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | userName | text |  | Nhập tên người dùng cần thêm |
| 2 | userCatergoyName | text |  | Chọn loại người dùng cần thêm trong danh sách loại người dùng |
| 3 | phoneNumber | text | Định dạng kiểu phone | Nhập số điện thoại người dùng cần thêm |
| 4 | accountName | text | Tối thiểu 2 kí tự | Nhập tên tài khoản cần thêm |
| 5 | password | text | Tối thiểu 4 kí tự | Nhập mật khẩu cần thêm |
| 6 | image | file |  | Thêm vào file hình ảnh |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Lưu | Lưu thông tin người dùng cần thêm vào cơ sở dữ liệu |
| 2 | Chọn nút Quay về | Chuyển về màn hình danh sách người dùng |

*A screenshot of a computer

Description automatically generated\* Màn hình chỉnh sửa người dùng*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | userName | text |  | Nhập tên người dùng cần chỉnh sửa |
| 2 | userCatergoyName | text |  | Chọn loại người dùng cần chỉnh sửa trong danh sách loại người dùng |
| 3 | phoneNumber | text | Định dạng kiểu phone | Nhập số điện thoại người dùng cần chỉnh sửa |
| 4 | accountName | text | Tối thiểu 2 kí tự | Nhập tên tài khoản cần chỉnh sửa |
| 5 | password | text | Tối thiểu 4 kí tự | Nhập mật khẩu cần chỉnh sửa |
| 6 | image | file |  | Thêm mới file hình ảnh |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Lưu | Cập nhật thông tin người dùng mới trên cơ sở dữ liệu |
| 2 | Chọn nút Quay về | Chuyển về màn hình danh sách người dùng |

*A screenshot of a computer

Description automatically generated\* Màn hình danh sách loại người dùng*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | userCatergoyName | text |  | Nhập tên loại người dùng cần tìm kiếm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Thêm | Chuyển sang màn hình thêm loại người dùng |
| 2 | Chọn nút Xóa nhiều | Xóa các loại người dùng đã được chọn |
| 3 | Chọn nút Xóa | Xóa loại người dùng tương ứng |
| 4 | Chọn nút Tìm kiếm | Hiển thị danh sách các loại người dùng theo thông tin tìm kiếm |
| 5 | Chọn nút Load trang | Xóa hết các nội dung tìm kiếm và hiển thị lại danh sách toàn bộ loại người dùng |

*A screenshot of a computer

Description automatically generated\* Màn hình thêm loại người dùng*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | userCatergoyName | text |  | Nhập tên loại người dùng cần thêm |
| 2 | userCategoryDescription | text |  | Nhập mô tả loại người dùng cần thêm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Lưu | Lưu thông tin loại người dùng cần thêm vào cơ sở dữ liệu |
| 2 | Chọn nút Quay về | Chuyển về màn hình danh sách loại người dùng |

*A screenshot of a computer

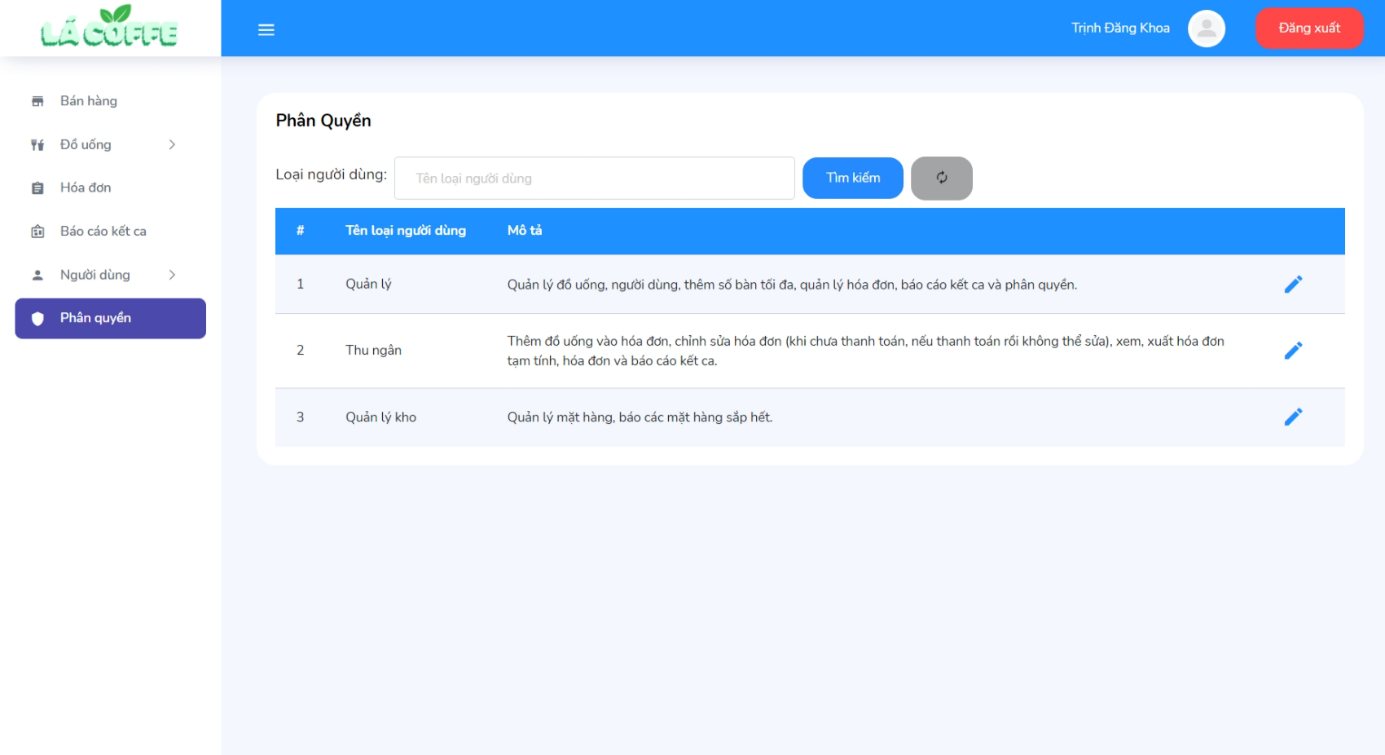
Description automatically generated\* Màn hình chỉnh sửa loại người dùng*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | userCatergoyName | text |  | Nhập tên loại người dùng cần thêm |
| 2 | userCategoryDescription | text |  | Nhập mô tả loại người dùng cần thêm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Lưu | Lưu thông tin loại người dùng cần thêm vào cơ sở dữ liệu |
| 2 | Chọn nút Quay về | Chuyển về màn hình danh sách loại người dùng |

*\* Màn hình danh sách phân quyền*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | userCatergoyName | text |  | Nhập tên loại người dùng cần tìm kiếm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Phân quyền | Chuyển sang màn hình phân quyền |
| 2 | Chọn nút Tìm kiếm | Hiển thị danh sách các loại người dùng theo thông tin tìm kiếm |
| 3 | Chọn nút Load trang | Xóa hết các nội dung tìm kiếm và hiển thị lại danh sách toàn bộ loại người dùng |

*\* Màn hình phân quyền*

*A screenshot of a computer

Description automatically generated*

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Chọn tất cả | Chọn tất cả các quyền có trong nhóm quyền tương ứng |
| 2 | Chọn nút Xóa tất cả | Xóa tất cả các quyền có trong nhóm quyền tương ứng |
| 3 | Chọn nút Cập nhật quyền hạn | Thêm các quyền được chọn cho loại người dùng tương ứng vào cơ sở dữ liệu |

#### 2.3.2. Bộ phận: Quản lý kho

*\* Màn hình danh sách mặt hàng*

*A screenshot of a computer

Description automatically generated*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | goodsName | text |  | Nhập tên mặt hàng cần tìm |
| 2 | goodsCategoryName | text |  | Nhập tên loại mặt hàng cần tìm |
| 3 | goodsUnit | text |  | Nhập đơn vị cần tìm |
| 4 | minPrice | number | > 0 | Nhập giá nhập nhỏ nhất cần tìm |
| 5 | maxPrice | number | > 0 | Nhập giá nhập lớn nhất cần tìm |
| 6 | minCount | number | > 0 | Nhập số lượng nhỏ nhất cần tìm |
| 7 | maxCount | number | > 0 | Nhập số lượng lớn nhất cần tìm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Thêm | Chuyển qua màn hình thêm mặt hàng |
| 2 | Chọn nút Xóa nhiều | Xóa các mặt hàng đã được chọn |
| 3 | Chọn nút Xóa | Xóa mặt hàng tương ứng |
| 4 | Chọn nút Chỉnh sửa | Chuyển qua màn hình chỉnh sửa mặt hàng |
| 5 | Chọn nút Tìm kiếm | Hiển thị danh sách các mặt hàng theo thông tin tìm kiếm |
| 6 | Chọn nút Load trang | Xóa hết các nội dung tìm kiếm và hiển thị lại danh sách toàn bộ mặt hàng |

*A screenshot of a computer

Description automatically generated\* Màn hình thêm mặt hàng*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | goodsName | text |  | Nhập tên mặt hàng cần thêm |
| 2 | goodsCategoryName | select |  | Chọn loại mặt hàng cần thêm trong danh sách loại mặt hàng |
| 3 | unit | text |  | Nhập đơn vị cần thêm |
| 4 | goodsPrice | number | > 0 | Nhập giá nhập cần thêm |
| 5 | goodsCount | number | > 0 | Nhập số lượng cần thêm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Lưu | Lưu thông tin mặt hàng cần thêm vào cơ sở dữ liệu |
| 2 | Chọn nút Quay lại | Chuyển về màn hình danh sách mặt hàng |

*A screenshot of a computer

Description automatically generated\* Màn hình chỉnh sửa mặt hàng*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | goodsName | text |  | Nhập tên mặt hàng cần chỉnh sửa |
| 2 | goodsCategoryName | select |  | Chọn loại mặt hàng cần chỉnh sửa trong danh sách loại mặt hàng |
| 3 | unit | text |  | Nhập đơn vị cần chỉnh sửa |
| 4 | goodsPrice | number | > 0 | Nhập giá nhập cần chỉnh sửa |
| 5 | goodsCount | number | > 0 | Nhập số lượng cần chỉnh sửa |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Lưu | Cập nhật thông tin mặt hàng mới trên cơ sở dữ liệu |
| 2 | Chọn nút Quay lại | Chuyển về màn hình danh sách mặt hàng |

*A screenshot of a computer

Description automatically generated\* Màn hình danh sách loại mặt hàng*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | goodsCategoryName | text |  | Nhập tên loại mặt hàng cần tìm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Thêm | Chuyển qua màn hình thêm loại mặt hàng |
| 2 | Chọn nút Xóa nhiều | Xóa các loại mặt hàng đã được chọn |
| 3 | Chọn nút Xóa | Xóa loại mặt hàng tương ứng |
| 4 | Chọn nút Chỉnh sửa | Chuyển qua màn hình chỉnh sửa loại mặt hàng |
| 5 | Chọn nút Tìm kiếm | Hiển thị danh sách các loại mặt hàng theo thông tin tìm kiếm |
| 6 | Chọn nút Load trang | Xóa hết các nội dung tìm kiếm và hiển thị lại danh sách toàn bộ loại mặt hàng |

*\* Màn hình thêm loại mặt hàng*

*A screenshot of a computer

Description automatically generated*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | goodsCategoryName | text |  | Nhập tên loại mặt hàng cần thêm |
| 2 | goodsCategoryDescription | text |  | Nhập mô tả loại mặt hàng cần thêm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Lưu | Lưu thông tin loại mặt hàng cần thêm vào cơ sở dữ liệu |
| 2 | Chọn nút Quay lại | Chuyển về màn hình danh sách loại mặt h |

*A screenshot of a computer

Description automatically generated\* Màn hình chỉnh sửa loại mặt hàng*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | goodsCategoryName | text |  | Nhập tên loại mặt hàng cần chỉnh sửa |
| 2 | goodsCategoryDescription | text |  | Nhập mô tả loại mặt hàng cần chỉnh sửa |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Lưu | Lưu thông tin loại mặt hàng cần chỉnh sửa vào cơ sở dữ liệu |
| 2 | Chọn nút Quay lại | Chuyển về màn hình danh sách loại mặt hàng |

*A screenshot of a computer

Description automatically generated\* Màn hình thống kê mặt hàng sắp hết*

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút In | In thông tin báo cáo các mặt hàng sắp hết (Xuất file PDF về máy) |

#### 2.3.3. Bộ phận: Thu ngân

**A screenshot of a computer

Description automatically generated***\* Màn hình nhập số tiền đầu ca*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | firstAmount | number | > 0 | Nhập số tiền đầu ca cần thêm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Xác nhận | Chuyển vào màn hình bán hàng |
| 2 | Chọn nút Hủy | Chuyển về màn hình đăng nhập |

A screenshot of a computer

Description automatically generated*\* Màn hình bán hàng*

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Bàn n | Chuyển qua màn hình thêm hóa đơn |

*A screenshot of a computer

Description automatically generated\* Màn hình lập hóa đơn*

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Đồ uống n | Thêm đồ uống vào chi tiết hóa đơn và hiển thị trên hóa đơn bên phải |
| 2 | Chọn nút Cộng | Tăng số lượng đồ uống |
| 3 | Chọn nút Trừ | Giảm số lượng đồ uống |
| 4 | Chọn nút Xóa | Xóa đồ uống tương ứng |
| 5 | Chọn nút Hóa đơn tạm tình | In thông tin hóa đơn tạm tính (Xuất file PDF về máy) |
| 6 | Chọn nút Thanh Toán | Lưu thông tin hóa đơn vào cơ sở dữ liệu và quay về màn hình bán hàng |
| 7 | Chọn nút Quay về | Chuyển về màn hình bán hàng |

*\* Màn hình danh sách hóa đơn*

*A screenshot of a computer

Description automatically generated*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | createDate | date |  | Nhập số ngày lập hóa đơn cần tìm kiếm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Xem chi tiết | Chuyển sang màn hình chi tiết hóa đơn |
| 2 | Chọn nút Tìm kiếm | Hiển thị danh sách các hóa đơn theo thông tin tìm kiếm |
| 3 | Chọn nút Load trang | Xóa hết các nội dung tìm kiếm và hiển thị lại danh sách toàn bộ hóa đơn |

*A screenshot of a computer

Description automatically generated\* Màn hình chi tiết hóa đơn*

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút In | In thông tin hóa đơn (Xuất file pdf về máy) |
| 2 | Chọn nút Quay về | Chuyển về màn hình danh sách hóa đơn |

*A screenshot of a computer

Description automatically generated\* Màn hình lập báo cáo kết ca*

Các đối tượng trên màn hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | lastAmount | number | > 0 | Nhập số tiền kết ca cần thêm |

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Kết ca | Lưu báo cáo kết ca vào cơ sở dữ liệu, sau đó xóa phiên người dùng hiện tại và chuyển sang màn hình đăng nhập |

#### 2.3.4. Một số biến cố khác

**A blue and white screen with a red and white circle

Description automatically generated**

Danh sách biến cố

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn nút Đăng xuất | Xóa phiên người dùng hiện tại và quay về trang đăng nhập |

## **3. Thiết kế xử lý**

### 3.1. Bộ phận quản lý

#### 3.1.1. Thêm, xóa sửa loại sản phẩm

#### 3.1.2. Tìm kiếm loại sản phẩm

#### 3.1.3. Thêm, xóa, sửa sản phẩm

#### 3.1.4. Tìm kiếm sản phẩm

##### 3.1.4.1. Tìm kiếm sản phẩm theo loại

##### 3.1.4.2. Tìm kiếm sản phẩm theo tên

##### 3.1.4.3. Tìm kiếm sản phẩm theo giá

#### 3.1.5. Xem, xóa và in hóa đơn

#### 3.1.6. Tìm kiếm hóa đơn

#### 3.1.7. Xem, xóa và in báo cáo kết ca

#### 3.1.8. Tìm kiếm báo cáo

#### 3.1.9. Thêm, xóa, sửa loại người dùng

#### 3.1.10. Thêm, xóa, sửa người dùng

#### 3.1.11. Tìm kiếm người dùng

##### 3.1.11.1. Tìm kiếm người dùng theo tên

##### 3.1.11.2. Tìm kiếm người dùng theo loại

##### 3.1.11.3. Tìm kiếm người dùng theo giới tính

#### 3.1.12. Phân quyền

### 3.2. Bộ phận quản lý kho

#### 3.2.1. Thêm, xóa, sửa loại mặt hàng

#### 3.2.2. Tìm kiếm loại mặt hàng

#### 3.2.3. Thêm, xóa, sửa mặt hàng

#### 3.2.4. Tìm kiếm mặt hàng

##### 3.2.4.1. Tìm kiếm mặt hàng theo tên

##### 3.2.4.2. Tìm kiếm mặt hàng theo loại

##### 3.2.4.3. Tìm kiếm mặt hàng theo đơn vị

##### 3.2.4.4. Tìm kiếm mặt hàng theo giá

##### 3.2.4.5. Tìm kiếm mặt hàng theo số lượng

#### 3.2.5. Thống kê mặt hàng sắp hết

### 3.3. Bộ phận thu ngân

#### 3.3.1. Lập hóa đơn

#### 3.3.2. Thanh toán

#### 3.3.3. Tìm kiếm hóa đơn

#### 3.3.4. Kết ca

# **CHƯƠNG 3: Cài đặt phần mềm**

## **1. Bộ phận quản lý**

### 1.1. Số bàn tối đa

public ActionResult ChangeTableCount(int? tableCount)

{

if (tableCount != null)

{

db.Tables.RemoveRange(db.Tables.ToList());

for (int i = 1; i <= tableCount; i++) // Vòng lặp để thêm số bàn mới

{

db.Tables.Add(new Tables("Bàn " + i, false)); // Thêm bàn mới với tên "Bàn " + số thứ tự và trạng thái false (có thể là trạng thái trống)

}

db.SaveChanges();

}

return RedirectToAction("Index"); // Chuyển hướng về trang "Index"

}

### 1.2. Thêm, xóa, sửa loại đồ uống

***Thêm loại đồ uống***

public ActionResult Create(DrinkCategories drinkCategories)

{

if (ModelState.IsValid) // Kiểm tra xem dữ liệu nhập vào có hợp lệ không

{

db.DrinkCategories.Add(drinkCategories); //Thêm danh mục đồ uống mới vào CSDL

db.SaveChanges();

return RedirectToAction("Index"); // Chuyển hướng đến trang Index sau khi lưu thành công

}

return View(drinkCategories); // Nếu dữ liệu không hợp lệ, hiển thị lại form với thông tin đã nhập

}

***Xóa loại đồ uống***

public ActionResult Delete(int id)

{

DrinkCategories drinkCategory = db.DrinkCategories.Find(id); // Tìm danh mục đồ uống theo id

db.DrinkCategories.Remove(drinkCategory); // Xóa danh mục đồ uống khỏi cơ sở dữ liệu

db.SaveChanges(); // Lưu thay đổi vào cơ sở dữ liệu

return Json(new { success = true }); // Trả về kết quả dưới dạng JSON, cho biết thành công

}

public ActionResult DeleteAll(string ids)

{

if (!string.IsNullOrEmpty(ids)) // Kiểm tra nếu chuỗi ids không rỗng hoặc null

{

var items = ids.Split(','); // Tách chuỗi ids thành mảng các id

if (items != null && items.Any()) // Kiểm tra nếu mảng items không rỗng và có phần tử

{

foreach (var item in items) // Duyệt qua từng id trong mảng

{

var o = db.DrinkCategories.Find(Convert.ToInt32(item)); // Tìm danh mục đồ uống theo id

db.DrinkCategories.Remove(o); // Xóa danh mục đồ uống khỏi CSDL

db.SaveChanges();

}

}

return Json(new { success = true }); // Trả về kết quả dưới dạng JSON, cho biết thành công

}

return Json(new { success = false }); // Nếu chuỗi ids rỗng hoặc null, trả về kết quả thất bại

}

***Sửa loại đồ uống***

public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

DrinkCategories drinkCategories = db.DrinkCategories.Find(id); // Tìm danh mục đồ uống theo id

if (drinkCategories == null)

{

return HttpNotFound();

}

return View(drinkCategories); // Trả về view chỉnh sửa

}

public ActionResult Edit(DrinkCategories drinkCategories)

{

if (ModelState.IsValid) // Kiểm tra xem dữ liệu nhập vào có hợp lệ không

{

db.Entry(drinkCategories).State = EntityState.Modified; // Đánh dấu đối tượng drinkCategories là đã sửa đổi

db.SaveChanges(); // Lưu các thay đổi

return RedirectToAction("Index"); // Chuyển hướng đến trang Index sau khi lưu thành công

}

return View(drinkCategories);

}

### 1.3. Tìm kiếm loại đồ uống

public ActionResult Index(string drinkCategoryName)

{

ViewBag.DrinkCategoryName = drinkCategoryName; // Lưu trữ giá trị drinkCategoryName để hiển thị lại trên view

var result = (from item in db.DrinkCategories

where (item.DrinkCategoryName.ToLower().Contains(drinkCategoryName.ToLower())

|| string.IsNullOrEmpty(drinkCategoryName))

select item).ToList(); // Truy vấn danh sách danh mục đồ uống theo tên

return View(result); // Trả về view với kết quả truy vấn

}

### 1.4. Thêm, xóa, sửa đồ uống

***Thêm đồ uống***

public ActionResult Create(Drinks drinks)

{

if (ModelState.IsValid)

{

Mở bảng Drinks

db.Drinks.Add(drinks); // Thêm đối tượng drinks vào cơ sở dữ liệu

var img = Request.Files["img"]; // Lấy tệp hình ảnh từ yêu cầu

if (img != null && img.ContentLength != 0) // Kiểm tra nếu tệp hình ảnh không null và có nội dung

{

string[] FileExtentions = new string[] { ".jpg", ".jpeg", ".png", ".gif" };

if (FileExtentions.Contains(img.FileName.Substring(img.FileName.LastIndexOf("."))))

{

string imgName = drinks.DrinkName + img.FileName.Substring(img.FileName.LastIndexOf(".")); // Tạo tên tệp hình ảnh mới

drinks.DrinkImage = imgName; // Lưu tên tệp hình ảnh vào thuộc tính DrinkImage

string PathDir = "~/Public/images/drinks/"; // Đường dẫn thư mục lưu trữ hình ảnh

string PathFile = Path.Combine(Server.MapPath(PathDir), imgName); // Tạo đường dẫn đầy đủ cho tệp hình ảnh

img.SaveAs(PathFile); // Lưu tệp hình ảnh vào đường dẫn chỉ định

}

}

db.SaveChanges(); // Lưu các thay đổi vào cơ sở dữ liệu

return RedirectToAction("Index"); // Chuyển hướng đến trang Index sau khi lưu thành công

}

ViewBag.DrinkCategoryID = new SelectList(db.DrinkCategories, "DrinkCategoryID", "DrinkCategoryName", drinks.DrinkCategoryID); // Tạo danh sách các danh mục đồ uống cho DropDownList

return View(drinks);

}

***Sửa đồ uống***

public ActionResult Edit(Drinks drinks)

{

if (ModelState.IsValid)

{

var img = Request.Files["img"];

if (img != null && img.ContentLength != 0

{

string[] FileExtentions = new string[] { ".jpg", ".jpeg", ".png", ".gif" };

if (FileExtentions.Contains(img.FileName.Substring(img.FileName.LastIndexOf("."))))

{

string imgName = drinks.DrinkName + img.FileName.Substring(img.FileName.LastIndexOf("."));

drinks.DrinkImage = imgName;

string PathDir = "~/Public/images/drinks/";

string PathFile = Path.Combine(Server.MapPath(PathDir), imgName);

img.SaveAs(PathFile);

}

}

db.Entry(drinks).State = EntityState.Modified; // Đánh dấu đối tượng drinks là đã sửa đổi

db.SaveChanges(); // Lưu các thay đổi vào cơ sở dữ liệu

return RedirectToAction("Index");

}

ViewBag.DrinkCategoryID = new SelectList(db.DrinkCategories, "DrinkCategoryID", "DrinkCategoryName", drinks.DrinkCategoryID); // Tạo danh sách các danh mục đồ uống cho DropDownList

return View(drinks);

}

***Xóa đồ uống***

public ActionResult Delete(int id)

{

Drinks drinks = db.Drinks.Find(id); // Tìm mục đồ uống cần xóa

string PathDir = "~/Public/images/drinks/"; // Đường dẫn thư mục lưu trữ hình ảnh

if (drinks.DrinkImage != null) // Kiểm tra xem có tệp hình ảnh được liên kết với mục đồ uống hay không

{

string fileName = System.IO.Path.GetFileNameWithoutExtension(drinks.DrinkImage); // Lấy tên tệp hình ảnh mà không có phần mở rộng

string[] DelPath = Directory.GetFiles(Server.MapPath(PathDir), fileName + ".\*"); // Tìm tất cả các tệp liên quan đến tên tệp hình ảnh

foreach (string dp in DelPath)

{

System.IO.File.Delete(dp); // Xóa tệp hình ảnh từ đường dẫn trên máy chủ

}

}

db.Drinks.Remove(drinks); // Xóa mục đồ uống khỏi cơ sở dữ liệu

db.SaveChanges(); // Lưu các thay đổi vào cơ sở dữ liệu

return Json(new { success = true }); // Trả về kết quả thành công dưới dạng JSON

}

public ActionResult DeleteAll(string ids)

{

if (!string.IsNullOrEmpty(ids))

{

var items = ids.Split(',');

if (items != null && items.Any())

{

foreach (var item in items) // Duyệt qua từng id trong mảng items

{

var o = db.Drinks.Find(Convert.ToInt32(item)); // Tìm mục đồ uống có id tương ứng

string PathDir = "~/Public/images/drinks/";

if (o.DrinkImage != null)

{

string fileName = System.IO.Path.GetFileNameWithoutExtension(o.DrinkImage);

string[] DelPath = Directory.GetFiles(Server.MapPath(PathDir), fileName + ".\*");

foreach (string dp in DelPath) // Duyệt qua từng tệp tìm thấy

{

System.IO.File.Delete(dp); // Xóa tệp hình ảnh từ đường dẫn trên máy chủ

}

}

db.Drinks.Remove(o);

db.SaveChanges();

}

}

return Json(new { success = true });

}

return Json(new { success = false });

}

### 1.5. Tìm kiếm đồ uống

***Tìm kiếm theo tên, loại và giá nhỏ nhất, lớn nhất***

public ActionResult Index()

{

var drinks = db.Drinks.Include(d => d.DrinkCategories); // Truy vấn cơ sở dữ liệu để lấy danh sách đồ uống và danh mục tương ứng

return View(drinks.ToList());

}

[HttpPost]

[RoleUser(FunctionID = "DU\_Xem")]

public ActionResult Index(string drinkName, string drinkCategoryName, decimal? minPrice, decimal? maxPrice)

{

ViewBag.DrinkName = drinkName; // Truyền giá trị biến DrinkName vào ViewBag để sử dụng trong view

ViewBag.DrinkCategoryName = drinkCategoryName;

ViewBag.MinPrice = minPrice;

ViewBag.MaxPrice = maxPrice;

var result = (from item in db.Drinks // Bắt đầu truy vấn LINQ từ bảng Drinks trong cơ sở dữ liệu

where (item.DrinkName.ToLower().Contains(drinkName.ToLower()) || string.IsNullOrEmpty(drinkName)) && // Điều kiện lọc theo tên đồ uống

(item.DrinkCategories.DrinkCategoryName.ToLower().Contains(drinkCategoryName.ToLower()) || string.IsNullOrEmpty(drinkCategoryName)) && // Điều kiện lọc theo tên danh mục đồ uống

(item.DrinkPrice >= minPrice || minPrice == null) && // Điều kiện lọc theo giá tối thiểu

(item.DrinkPrice <= maxPrice || maxPrice == null) // Điều kiện lọc theo giá tối đa

select item).ToList(); //Chuyển kết quả truy vấn thành list

return View(result); // Trả về view "Index" với danh sách đồ uống đã lọc được theo các điều kiện

***}***

### 1.6. Xem, xóa và in hóa đơn

***Xem hóa đơn***

public ActionResult Details(String id)

{

TempData["BillID"] = id; // Lưu trữ ID của hóa đơn vào TempData để sử dụng sau này

Bills bills = db.Bills.Find(id); // Tìm hóa đơn dựa trên ID được chỉ định

return View(bills.BillInfos.ToList()); // Trả về view "Details" với danh sách các mục hóa đơn của hóa đơn được chỉ định

}

***Xóa hóa đơn***

[HttpPost]

[RoleUser(FunctionID = "HD\_Xoa")]

public ActionResult Delete(String id)

{

// Tìm hóa đơn dựa trên ID được chỉ định

Bills bill = db.Bills.Find(id);

db.Bills.Remove(bill); // Xóa hóa đơn

db.SaveChanges();

return Json(new { success = true }); // Trả về kết quả JSON cho phía máy khách, thông báo thành công

}

[HttpPost]

[RoleUser(FunctionID = "HD\_Xoa")]

public ActionResult DeleteAll(string ids)

{

if (!string.IsNullOrEmpty(ids))

{

var items = ids.Split(',');

// Nếu danh sách ID không rỗng

if (items != null && items.Any())

{

foreach (var item in items)

{

var o = db.Bills.Find(item.ToString());// Tìm hóa đơn dựa trên ID

db.Bills.Remove(o);

db.SaveChanges();

}

}

return Json(new { success = true });

}

return Json(new { success = false });

}

***In hóa đơn***

public ActionResult ExportPDF()

{

// Khởi tạo đối tượng chuyển đổi từ HTML sang PDF

HtmlToPdf converter = new HtmlToPdf();

// Cấu hình kích thước trang PDF và cách đặt trang

converter.Options.PdfPageSize = PdfPageSize.A6;

converter.Options.PdfPageOrientation = PdfPageOrientation.Portrait;

converter.Options.MarginLeft = 10;

converter.Options.MarginRight = 10;

converter.Options.MarginTop = 20;

converter.Options.MarginBottom = 20;

String billID = TempData["BillID"] as String;

Bills bills = db.Bills.Find(billID);

// Render view "HoaDon.cshtml" thành một chuỗi HTML

var htmlPDF = base.RenderPartialToString("~/Areas/Admin/Views/Bills/HoaDon.cshtml", bills.BillInfos.ToList());

PdfDocument doc = converter.ConvertHtmlString(htmlPDF); // Chuyển đổi chuỗi HTML thành tài liệu PDF

string fileName = string.Format("{0}.pdf", bills.BillID); // Tạo tên file PDF dựa trên ID của hóa đơn

string pathFile = Path.Combine(Server.MapPath("~/Public/PDF"), fileName); // Đường dẫn lưu trữ tập tin PDF

doc.Save(pathFile);

doc.Close();

return Json(new { success = true }, JsonRequestBehavior.AllowGet);

}

### 1.7. Tìm kiếm hóa đơn theo ngày

public ActionResult Index()

{

return View(db.Bills.Where(b => b.TableID == null).ToList()); // Trả về view "Index" với danh sách các hóa đơn chưa được gắn với bàn.

}

[HttpPost]

[RoleUser(FunctionID = "HD\_Xem")]

public ActionResult Index(DateTime? createDate)

{

if (createDate != null)

{

// Lọc và hiển thị danh sách các hóa đơn chưa được gắn với bàn dựa trên ngày tạo.

var result = (from item in db.Bills

where (item.CreateDate.Day == createDate.Value.Day) &&

(item.CreateDate.Month == createDate.Value.Month) &&

(item.CreateDate.Year == createDate.Value.Year) &&

(item.TableID == null)

select item).ToList();

return View(result);

}

return RedirectToAction("Index");

}

### 1.8. Xem, xóa và in bc kết ca

***Xem báo cáo kết ca***

public ActionResult Details(String id)

{

TempData["ShiftReportID"] = id;

ShiftReports shiftReports = db.ShiftReports.Find(id); // Tìm báo cáo ca làm việc dựa trên ID được chỉ định

return View(shiftReports);

}

***Xóa báo cáo kết ca***

public ActionResult Delete(String id)

{

ShiftReports shiftReports = db.ShiftReports.Find(id); // Tìm báo cáo ca làm việc dựa trên ID được chỉ định

db.ShiftReports.Remove(shiftReports);

db.SaveChanges();

return Json(new { success = true });

}

[HttpPost]

[RoleUser(FunctionID = "BCKC\_Xoa")]

public ActionResult DeleteAll(string ids)

{

if (!string.IsNullOrEmpty(ids))

{

var items = ids.Split(',');

if (items != null && items.Any())

{

foreach (var item in items)

{

var o = db.ShiftReports.Find(item.ToString());// Tìm báo cáo ca làm việc dựa trên ID

db.ShiftReports.Remove(o);

db.SaveChanges();

}

}

return Json(new { success = true });

}

return Json(new { success = false });

}

***In báo cáo kết ca***

public ActionResult ExportPDFCT()

{

HtmlToPdf converter = new HtmlToPdf();

// Cấu hình kích thước trang PDF và cách đặt trang

converter.Options.PdfPageSize = PdfPageSize.A6;

converter.Options.PdfPageOrientation = PdfPageOrientation.Portrait;

converter.Options.MarginLeft = 10;

converter.Options.MarginRight = 10;

converter.Options.MarginTop = 20;

converter.Options.MarginBottom = 20;

String shiftReportID = TempData["ShiftReportID"] as String;

// Tìm báo cáo ca làm việc trong cơ sở dữ liệu dựa trên ID

ShiftReports shiftReports = db.ShiftReports.Find(shiftReportID);

// Render view "BaoCaoKetCa.cshtml" thành một chuỗi HTML

var htmlPDF = base.RenderPartialToString("~/Areas/Admin/Views/ShiftReports/BaoCaoKetCa.cshtml", shiftReports);

PdfDocument doc = converter.ConvertHtmlString(htmlPDF);

string fileName = string.Format("{0}.pdf", shiftReports.ShiftReportID);

string pathFile = Path.Combine(Server.MapPath("~/Public/PDF"), fileName);

doc.Save(pathFile);

doc.Close();

return Json(new { success = true }, JsonRequestBehavior.AllowGet);

}

### 1.9. Tìm kiếm báo cáo kết ca theo ngày

public ActionResult DanhSach()

{

// Trả về view "DanhSach" với danh sách các báo cáo ca làm việc

return View(db.ShiftReports.ToList());

}

[HttpPost]

[RoleUser(FunctionID = "BCKC\_Xem")]

public ActionResult DanhSach(DateTime? createDate)

{

if (createDate != null)

{

// Lọc và trả về danh sách các báo cáo ca làm việc dựa trên ngày tạo

var result = (from item in db.ShiftReports

where (item.LastTime.Day == createDate.Value.Day) &&

(item.LastTime.Month == createDate.Value.Month) &&

(item.LastTime.Year == createDate.Value.Year)

select item).ToList();

return View(result);

}

// Nếu không có ngày tạo được chỉ định, chuyển hướng lại đến action "DanhSach" để hiển thị toàn bộ danh sách

return RedirectToAction("DanhSach");

}

### 1.10. Thêm xóa sửa loại người dùng

***Thêm loại người dùng***

public ActionResult Create()

{

return View();

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Create(UserCategories userCategories)

{

if (ModelState.IsValid)

{

// Thêm một danh mục người dùng mới vào cơ sở dữ liệu nếu dữ liệu hợp lệ

db.UserCategories.Add(userCategories);

db.SaveChanges();

return RedirectToAction("Index");

}

return View(userCategories);

}

***Xóa loại người dùng***

public ActionResult Delete(int id)

{

// Tìm danh mục người dùng dựa trên ID được chỉ định

UserCategories userCategory = db.UserCategories.Find(id);

db.UserCategories.Remove(userCategory);

db.SaveChanges();

return Json(new { success = true });

}

[HttpPost]

public ActionResult DeleteAll(string ids)

{

if (!string.IsNullOrEmpty(ids))

{

// Chia các ID thành một mảng

var items = ids.Split(',');

if (items != null && items.Any())

{

foreach (var item in items)

{

// Tìm danh mục người dùng dựa trên ID

var o = db.UserCategories.Find(Convert.ToInt32(item));

db.UserCategories.Remove(o);

db.SaveChanges();

}

}

return Json(new { success = true });

}

return Json(new { success = false });

}

***Sửa loại người dùng***

public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

// Tìm danh mục người dùng dựa trên ID được cung cấp

UserCategories userCategories = db.UserCategories.Find(id);

// Kiểm tra xem danh mục người dùng có tồn tại hay không

if (userCategories == null)

{

return HttpNotFound();

}

return View(userCategories);

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Edit(UserCategories userCategories)

{

if (ModelState.IsValid)

{

// Cập nhật trạng thái của đối tượng userCategories thành Modified để chỉ ra rằng nó đã được thay đổi

db.Entry(userCategories).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index");

}

return View(userCategories);

}

### 1.11. Tìm kiếm loại người dùng

public ActionResult Index()

{

return View(db.UserCategories.ToList());

}

[HttpPost]

public ActionResult Index(string userCategoryName)

{

ViewBag.UserCategoryName = userCategoryName;

// Lấy danh sách các danh mục người dùng từ cơ sở dữ liệu và lọc dựa trên tên nếu có

var result = (from item in db.UserCategories

where (item.UserCategoryName.ToLower().Contains(userCategoryName.ToLower()) || string.IsNullOrEmpty(userCategoryName))

select item).ToList();

// Trả về view "Index" với danh sách danh mục người dùng đã lọc được

return View(result);

}

### 1.12. Thêm xóa sửa người dùng

***Thêm người dùng***

public ActionResult Create()

{

ViewBag.UserCategoryID = new SelectList(db.UserCategories, "UserCategoryID", "UserCategoryName");

return View();

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Create(User user)

{

if (ModelState.IsValid)

{

var img = Request.Files["img"];

if (img != null && img.ContentLength != 0)

{

string[] FileExtentions = new string[] { ".jpg", ".jpeg", ".png", ".gif" };

if (FileExtentions.Contains(img.FileName.Substring(img.FileName.LastIndexOf("."))))

{

string imgName = user.UserImage + img.FileName.Substring(img.FileName.LastIndexOf("."));

user.UserImage = imgName;

string PathDir = "~/Public/images/users/";

string PathFile = Path.Combine(Server.MapPath(PathDir), imgName);

img.SaveAs(PathFile);

}

}

if (db.Users.Count(u => u.AccountName == user.AccountName) > 0)

{

ViewBag.Error = "Tên đăng nhập đã tồn tại";

ViewBag.UserCategoryID = new SelectList(db.UserCategories, "UserCategoryID", "UserCategoryName", user.UserCategoryID);

return View();

}

db.Users.Add(user);

db.SaveChanges();

return RedirectToAction("Index");

}

ViewBag.UserCategoryID = new SelectList(db.UserCategories, "UserCategoryID", "UserCategoryName", user.UserCategoryID);

return View(user);

}

***Xóa người dùng***

public ActionResult Delete(int id)

{

User user = db.Users.Find(id);

string PathDir = "~/Public/images/users/";

if (user.UserImage != null)

{

string fileName = System.IO.Path.GetFileNameWithoutExtension(user.UserImage);

string[] DelPath = Directory.GetFiles(Server.MapPath(PathDir), fileName + ".\*");

foreach (string dp in DelPath)

{

System.IO.File.Delete(dp);

}

}

db.Users.Remove(user);

db.SaveChanges();

return Json(new { success = true });

}

[HttpPost]

[RoleUser(FunctionID = "ND\_Xoa")]

public ActionResult DeleteAll(string ids)

{

if (!string.IsNullOrEmpty(ids))

{

var items = ids.Split(',');

if (items != null && items.Any())

{

foreach (var item in items)

{

var o = db.Users.Find(Convert.ToInt32(item));

string PathDir = "~/Public/images/users/";

if (o.UserImage != null)

{

string fileName = System.IO.Path.GetFileNameWithoutExtension(o.UserImage);

string[] DelPath = Directory.GetFiles(Server.MapPath(PathDir), fileName + ".\*");

foreach (string dp in DelPath)

{

System.IO.File.Delete(dp);

}

}

db.Users.Remove(o);

db.SaveChanges();

}

}

return Json(new { success = true });

}

return Json(new { success = false });

}

***Sửa người dùng***

public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

User user = db.Users.Find(id);

if (user == null)

{

return HttpNotFound();

}

ViewBag.UserCategoryID = new SelectList(db.UserCategories, "UserCategoryID", "UserCategoryName", user.UserCategoryID);

return View(user);

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Edit(User user)

{

if (ModelState.IsValid)

{

var img = Request.Files["img"];

if (img != null && img.ContentLength != 0)

{

string[] FileExtentions = new string[] { ".jpg", ".jpeg", ".png", ".gif" };

if (FileExtentions.Contains(img.FileName.Substring(img.FileName.LastIndexOf("."))))

{

string imgName = user.UserImage + img.FileName.Substring(img.FileName.LastIndexOf("."));

user.UserImage = imgName;

string PathDir = "~/Public/images/users/";

string PathFile = Path.Combine(Server.MapPath(PathDir), imgName);

img.SaveAs(PathFile);

}

}

db.Entry(user).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index");

}

ViewBag.UserCategoryID = new SelectList(db.UserCategories, "UserCategoryID", "UserCategoryName", user.UserCategoryID);

return View(user);

}

### 1.13. Tìm kiếm người dùng

public ActionResult Index()

{

// Trả về view "Index" với danh sách tất cả người dùng

return View(db.Users.ToList());

}

[HttpPost]

public ActionResult Index(string fullName, string gender, string userCategoryName)

{

// Lưu trữ các thông tin tìm kiếm vào ViewBag để sử dụng trong view

ViewBag.FullName = fullName;

ViewBag.Gender = gender;

ViewBag.UserCategoryName = userCategoryName;

// Lấy danh sách người dùng từ cơ sở dữ liệu và lọc dựa trên các thông tin tìm kiếm

var result = (from item in db.Users

where (item.FullName.ToLower().Contains(fullName.ToLower()) || string.IsNullOrEmpty(fullName)) && // Lọc theo tên

(item.Gender.ToLower().Contains(gender.ToLower()) || string.IsNullOrEmpty(gender)) && // Lọc theo giới tính

(item.UserCategories.UserCategoryName.ToLower().Contains(userCategoryName.ToLower()) || string.IsNullOrEmpty(userCategoryName))

select item).ToList(); // Lọc theo loại người dùng

// Trả về view "Index" với danh sách người dùng đã lọc được

return View(result);

}

### 1.14. Phân quyền

***Chỉnh sửa quyền***

public ActionResult Authorize(int UserCategoryID)

{

ViewBag.ListFunctions = db.Functions.ToList();

UserCategories userCategories = db.UserCategories.Find(UserCategoryID);

return View(userCategories);

}

[HttpPost]

public ActionResult Authorize(int UserCategoryID, List<string> functionIDs)

{

// Xóa các quyền hiện tại

var existingAuthorizations = db.Authorizations.Where(a => a.UserCategoryID == UserCategoryID).ToList();

db.Authorizations.RemoveRange(existingAuthorizations);

db.SaveChanges();

// Thêm các quyền mới

if (functionIDs != null && functionIDs.Count > 0)

{

foreach (var functionID in functionIDs)

{

var authorization = new Authorizations

{

UserCategoryID = UserCategoryID,

FunctionID = functionID

};

db.Authorizations.Add(authorization);

}

db.SaveChanges();

}

return RedirectToAction("Authorize", new { UserCategoryID = UserCategoryID});

}

## **2. Bộ phận quản lý kho**

### 2.1. Thêm xóa sửa loại mặt hàng

***Thêm loại mặt hàng***

public ActionResult Create()

{

return View();

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Create(GoodsCategories goodsCategories)

{

if (ModelState.IsValid)

{

db.GoodsCategories.Add(goodsCategories);

db.SaveChanges();

return RedirectToAction("Index");

}

return View(goodsCategories);

}

***Xóa loại mặt hàng***

public ActionResult Delete(int id)

{

GoodsCategories goodsCategory = db.GoodsCategories.Find(id);

db.GoodsCategories.Remove(goodsCategory);

db.SaveChanges();

return Json(new { success = true });

}

[HttpPost]

public ActionResult DeleteAll(string ids)

{

if (!string.IsNullOrEmpty(ids))

{

var items = ids.Split(',');

if (items != null && items.Any())

{

foreach (var item in items)

{

var o = db.GoodsCategories.Find(Convert.ToInt32(item));

db.GoodsCategories.Remove(o);

db.SaveChanges();

}

}

return Json(new { success = true });

}

return Json(new { success = false });

}

***Sửa loại mặt hàng***

public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

GoodsCategories goodsCategories = db.GoodsCategories.Find(id);

if (goodsCategories == null)

{

return HttpNotFound();

}

return View(goodsCategories);

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Edit(GoodsCategories goodsCategories)

{

if (ModelState.IsValid)

{

db.Entry(goodsCategories).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index");

}

return View(goodsCategories);

}

### 2.2. Tìm kiếm loại mặt hàng

public ActionResult Index()

{

// Trả về view "Index" với danh sách tất cả các danh mục hàng hóa

return View(db.GoodsCategories.ToList());

}

[HttpPost]

public ActionResult Index(string goodsCategoryName)

{

// Lưu trữ tên danh mục hàng hóa được nhập vào ViewBag để sử dụng trong view

ViewBag.GoodsCategoryName = goodsCategoryName;

// Lấy danh sách các danh mục hàng hóa từ cơ sở dữ liệu và lọc dựa trên tên nếu có

var result = (from item in db.GoodsCategories

where (item.GoodsCategoryName.ToLower().Contains(goodsCategoryName.ToLower()) || string.IsNullOrEmpty(goodsCategoryName))

select item).ToList();

// Trả về view "Index" với danh sách danh mục hàng hóa đã lọc được

return View(result);

}

### 2.3. Thêm xóa sửa mặt hàng

***Thêm mặt hàng***

public ActionResult Create()

{

ViewBag.GoodsCategoryID = new SelectList(db.GoodsCategories, "GoodsCategoryID", "GoodsCategoryName");

return View();

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Create(Goods goods)

{

if (ModelState.IsValid)

{

db.Goods.Add(goods);

db.SaveChanges();

return RedirectToAction("Index");

}

ViewBag.GoodsCategoryID = new SelectList(db.GoodsCategories, "GoodsCategoryID", "GoodsCategoryName", goods.GoodsCategoryID);

return View(goods);

}

***Xóa mặt hàng***

public ActionResult Delete(int id)

{

Goods goods = db.Goods.Find(id);

db.Goods.Remove(goods);

db.SaveChanges();

return Json(new { success = true });

}

[HttpPost]

[RoleUser(FunctionID = "MH\_Xoa")]

public ActionResult DeleteAll(string ids)

{

if (!string.IsNullOrEmpty(ids))

{

var items = ids.Split(',');

if (items != null && items.Any())

{

foreach (var item in items)

{

var o = db.Goods.Find(Convert.ToInt32(item));

db.Goods.Remove(o);

db.SaveChanges();

}

}

return Json(new { success = true });

}

return Json(new { success = false });

}

***Sửa mặt hàng***

public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Goods goods = db.Goods.Find(id);

if (goods == null)

{

return HttpNotFound();

}

ViewBag.GoodsCategoryID = new SelectList(db.GoodsCategories, "GoodsCategoryID", "GoodsCategoryName", goods.GoodsCategoryID);

return View(goods);

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Edit(Goods goods)

{

if (ModelState.IsValid)

{

db.Entry(goods).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index");

}

ViewBag.GoodsCategoryID = new SelectList(db.GoodsCategories, "GoodsCategoryID", "GoodsCategoryName", goods.GoodsCategoryID);

return View(goods);

}

### 2.4. Tìm kiếm mặt hàng

***Tìm kiếm mặt hàng theo loại, tên, đơn vị, giá và số lượng***

public ActionResult Index()

{

// Lấy danh sách hàng hóa từ cơ sở dữ liệu bao gồm cả thông tin về danh mục hàng hóa

var goods = db.Goods.Include(g => g.GoodsCategories);

// Tạo SelectList chứa danh sách ID và tên của các hàng hóa để sử dụng trong view

ViewBag.GoodsID = new SelectList(db.Goods, "GoodsID", "GoodsName");

// Trả về view "Index" với danh sách hàng hóa và SelectList

return View(goods.ToList());

}

[HttpPost]

public ActionResult Index(string goodsName, string goodsCategoryName, string goodsUnit, decimal? minPrice, decimal? maxPrice, int? minCount, int? maxCount)

{

// Lưu các thông tin tìm kiếm vào ViewBag để sử dụng trong view

ViewBag.GoodsName = goodsName;

ViewBag.GoodsCategoryName = goodsCategoryName;

ViewBag.GoodsUnit = goodsUnit;

ViewBag.MinPrice = minPrice;

ViewBag.MaxPrice = maxPrice;

ViewBag.MinCount = minCount;

ViewBag.MaxCount = maxCount;

// Lấy danh sách hàng hóa từ cơ sở dữ liệu và lọc dựa trên các thông tin tìm kiếm

var result = (from item in db.Goods

where (item.GoodsName.ToLower().Contains(goodsName.ToLower()) || string.IsNullOrEmpty(goodsName)) && // Lọc theo tên hàng hóa

(item.GoodsCategories.GoodsCategoryName.ToLower().Contains(goodsCategoryName.ToLower()) || string.IsNullOrEmpty(goodsCategoryName)) && // Lọc theo loại

(item.GoodsUnit.ToLower().Contains(goodsUnit.ToLower()) || string.IsNullOrEmpty(goodsUnit)) &&

(item.GoodsCount >= minCount || minCount == null) &&

// Lọc theo đơn vị nhỏ nhất

(item.GoodsCount <= maxCount || maxCount == null) &&

// Lọc theo đơn vị lớn nhất

(item.GoodsPrice >= minPrice || minPrice == null) &&

// Lọc theo giá nhỏ nhất

(item.GoodsPrice <= maxPrice || maxPrice == null)

// Lọc theo giá lớn nhất

select item).ToList();

// Trả về view "Index" với danh sách hàng hóa đã lọc được

return View(result);

}

### 2.5. Thống kê số lượng mặt hàng sắp hết

***Sửa thống kê mặt hàng sắp hết***

public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Goods goods = db.Goods.Find(id);

if (goods == null)

{

return HttpNotFound();

}

ViewBag.GoodsCategoryID = new SelectList(db.GoodsCategories, "GoodsCategoryID", "GoodsCategoryName", goods.GoodsCategoryID);

return View(goods);

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Edit(Goods goods)

{

if (ModelState.IsValid)

{

db.Entry(goods).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index");

}

ViewBag.GoodsCategoryID = new SelectList(db.GoodsCategories, "GoodsCategoryID", "GoodsCategoryName", goods.GoodsCategoryID);

return View(goods);

}

***In PDF thống kê mặt hàng sắp hết***

public ActionResult ExportPDF()

{

// Khởi tạo đối tượng converter để chuyển đổi HTML thành PDF

HtmlToPdf converter = new HtmlToPdf();

// Cấu hình kích thước và hướng của trang PDF

converter.Options.PdfPageSize = PdfPageSize.A6;

converter.Options.PdfPageOrientation = PdfPageOrientation.Portrait;

// Cấu hình các margin của trang PDF

converter.Options.MarginLeft = 10;

converter.Options.MarginRight = 10;

converter.Options.MarginTop = 20;

converter.Options.MarginBottom = 20;

// Lấy danh sách các mặt hàng có số lượng tồn kho nhỏ hơn hoặc bằng 5

var result = db.Goods.Where(g => g.GoodsCount <= 5).ToList();

// Render view thành chuỗi HTML sử dụng RenderPartialToString

var htmlPDF = base.RenderPartialToString("~/Areas/Admin/Views/ThongKe/BaoCao.cshtml", result);

// Chuyển đổi chuỗi HTML thành tài liệu PDF

PdfDocument doc = converter.ConvertHtmlString(htmlPDF);

// Tạo tên tệp PDF dựa trên tháng và ngày hiện tại

string fileName = string.Format("{0}.pdf", "BC" + DateTime.Now.Month + DateTime.Now.Day);

// Đường dẫn lưu trữ tệp PDF

string pathFile = Path.Combine(Server.MapPath("~/Public/PDF"), fileName);

doc.Save(pathFile);

doc.Close();

return Json(new { success = true }, JsonRequestBehavior.AllowGet);

}

## **3. Bộ phận thu ngân**

### 3.1. Nhập tiền đầu ca

public ActionResult StartShift(decimal firstAmount)

{

ShiftReports shiftReports = new ShiftReports();

// Tạo ID cho báo cáo ca làm việc dựa trên tháng, ngày và một phần GUID

shiftReports.ShiftReportID = "KC" + DateTime.Now.Month + DateTime.Now.Day + Guid.NewGuid().ToString().Substring(0, 8);

// Gán giá trị số tiền đầu ca

shiftReports.FirstAmount = firstAmount;

// Khởi tạo giá trị cuối ca, số lượng hóa đơn, doanh thu và mã người dùng

shiftReports.LastAmount = 0;

shiftReports.BillCount = 0;

shiftReports.Revenue = 0;

shiftReports.UserID = SessionConfig.GetUser().UserID;

// Kiểm tra nếu có hóa đơn chưa thanh toán

if (db.Bills.Where(b => b.Tables.Status == true).Count() > 0)

{

// Tính tổng số tiền chưa thu được từ các hóa đơn chưa thanh toán

shiftReports.UncollectedAmount = db.Bills.Where(b => b.Tables.Status == true).Sum(b => b.TotalAmount);

}

else

{

// Nếu không có hóa đơn chưa thanh toán, gán giá trị 0 cho số tiền chưa thu được

shiftReports.UncollectedAmount = 0;

}

// Gán thời gian bắt đầu và thời gian cuối ca làm việc

shiftReports.FirstTime = DateTime.Now;

shiftReports.LastTime = DateTime.Now;

// Thêm báo cáo ca làm việc vào cơ sở dữ liệu và lưu thay đổi

db.ShiftReports.Add(shiftReports);

db.SaveChanges();

// Lưu báo cáo ca làm việc vào session để sử dụng sau này

SessionConfig.SetShiftReport(shiftReports);

return RedirectToAction("Index");

}

### 3.2. Bán hàng

***Thêm đồ uống***

public ActionResult AddDrinkToBill(int id)

{

// Lấy ID bàn từ TempData

int? tableID = TempData["TableID"] as int?;

// Tìm thông tin về đồ uống từ cơ sở dữ liệu

Drinks drink = db.Drinks.Find(id);

// Tạo mới một thông tin chi tiết hóa đơn cho đồ uống này

BillInfos billInfo = new BillInfos();

// Tìm thông tin về bàn từ cơ sở dữ liệu

Tables table = db.Tables.Find(tableID);

// Thiết lập các thông tin cho thông tin chi tiết hóa đơn mới

billInfo.DrinkID = id;

billInfo.DrinkCount = 1;

billInfo.DrinkPrice = drink.DrinkPrice \* billInfo.DrinkCount;

// Nếu bàn đang trống và chưa có hóa đơn nào được tạo, tạo mới một hóa đơn

if (table.Status == false && !table.Bills.Any())

{

Bills bills = new Bills();

bills.BillID = "HD" + DateTime.Now.Month + DateTime.Now.Day + Guid.NewGuid().ToString().Substring(0, 8);

Session["Bills"] = bills.BillID;

bills.CreateDate = DateTime.Now;

bills.TotalAmount = billInfo.DrinkPrice;

bills.TableID = (int)tableID;

bills.UserID = SessionConfig.GetUser().UserID;

billInfo.BillID = bills.BillID;

db.Bills.Add(bills);

db.BillInfos.Add(billInfo);

table.Status = true;

table.Bills.Add(bills);

db.Entry(table).State = EntityState.Modified;

db.SaveChanges();

}

else

{

// Nếu bàn đã có hóa đơn, thêm đồ uống vào hóa đơn đó hoặc tăng số lượng nếu đồ uống đã tồn tại trong hóa đơn

Bills bills = db.Bills.Where(b => b.TableID == tableID).FirstOrDefault();

BillInfos existingBillInfo = db.BillInfos.FirstOrDefault(bi => bi.BillID == bills.BillID && bi.DrinkID == id);

if (existingBillInfo != null)

{

existingBillInfo.DrinkCount += 1;

existingBillInfo.DrinkPrice = existingBillInfo.DrinkCount \* drink.DrinkPrice;

db.Entry(existingBillInfo).State = EntityState.Modified;

db.SaveChanges();

}

else

{

billInfo.BillID = bills.BillID;

db.BillInfos.Add(billInfo);

db.SaveChanges();

}

// Cập nhật tổng tiền của hóa đơn

bills.TotalAmount = db.BillInfos

.Where(bi => bi.BillID == bills.BillID)

.Sum(bi => bi.DrinkPrice);

db.Entry(bills).State = EntityState.Modified;

db.SaveChanges();

}

// Chuyển hướng đến trang Order của bàn đó

return RedirectToAction("Order", new { id = tableID });

}

***Tăng số lượng đồ uống***

public ActionResult TangSoLuong(int idDrink, String idBill)

{

int? tableID = TempData["TableID"] as int?;

BillInfos billInfos = db.BillInfos.Find(idBill, idDrink);

Drinks drinks = db.Drinks.Find(idDrink);

Bills bills = db.Bills.Find(idBill);

billInfos.DrinkCount++;

billInfos.DrinkPrice = drinks.DrinkPrice \* billInfos.DrinkCount;

db.Entry(billInfos).State = EntityState.Modified;

db.SaveChanges();

bills.TotalAmount = db.BillInfos

.Where(bi => bi.BillID == bills.BillID)

.Sum(bi => bi.DrinkPrice);

db.Entry(bills).State = EntityState.Modified;

db.SaveChanges();

if (billInfos.DrinkCount == 0)

{

db.BillInfos.Remove(billInfos);

db.SaveChanges();

if (!db.BillInfos.Any(b => b.BillID.Equals(idBill)))

{

db.Bills.Remove(bills);

db.SaveChanges();

}

}

return RedirectToAction("Order", new { id = tableID });

}

***Giảm số lượng đồ uống***

public ActionResult GiamSoLuong(int idDrink, String idBill)

{

// Lấy ID bàn từ TempData

int? tableID = TempData["TableID"] as int?;

// Tìm thông tin chi tiết hóa đơn tương ứng với ID hóa đơn và ID đồ uống

BillInfos billInfos = db.BillInfos.Find(idBill, idDrink);

// Tìm thông tin về đồ uống từ cơ sở dữ liệu

Drinks drinks = db.Drinks.Find(idDrink);

// Tìm thông tin về hóa đơn từ cơ sở dữ liệu

Bills bills = db.Bills.Find(idBill);

// Giảm số lượng đồ uống trong hóa đơn đi 1 và cập nhật lại giá tiền của đồ uống đó

billInfos.DrinkCount--;

billInfos.DrinkPrice = drinks.DrinkPrice \* billInfos.DrinkCount;

// Đánh dấu thông tin chi tiết hóa đơn đã được sửa đổi và lưu vào cơ sở dữ liệu

db.Entry(billInfos).State = EntityState.Modified;

db.SaveChanges();

// Tính lại tổng tiền của hóa đơn dựa trên thông tin chi tiết hóa đơn và cập nhật vào hóa đơn

bills.TotalAmount = db.BillInfos

.Where(bi => bi.BillID == bills.BillID)

.Sum(bi => bi.DrinkPrice);

db.Entry(bills).State = EntityState.Modified;

db.SaveChanges();

// Nếu số lượng đồ uống trong hóa đơn giảm xuống 0, xóa thông tin chi tiết hóa đơn đó

if (billInfos.DrinkCount == 0)

{

db.BillInfos.Remove(billInfos);

db.SaveChanges();

// Nếu không còn thông tin chi tiết hóa đơn nào cho hóa đơn này, xóa luôn hóa đơn đó

if (!db.BillInfos.Any(b => b.BillID.Equals(idBill)))

{

db.Bills.Remove(bills);

db.SaveChanges();

}

}

// Chuyển hướng đến trang Order của bàn đó

return RedirectToAction("Order", new { id = tableID });

}

***Xóa đồ uống***

public ActionResult Delete(int idDrink, String idBill)

{

BillInfos billInfos = db.BillInfos.Find(idBill, idDrink);

Bills bills = db.Bills.Find(idBill);

db.BillInfos.Remove(billInfos);

db.SaveChanges();

if (!db.BillInfos.Any(b => b.BillID.Equals(idBill)))

{

db.Bills.Remove(bills);

db.SaveChanges();

}

else

{

bills.TotalAmount = db.BillInfos

.Where(bi => bi.BillID == bills.BillID)

.Sum(bi => bi.DrinkPrice);

db.Entry(bills).State = EntityState.Modified;

db.SaveChanges();

}

;

return Json(new { success = true });

}

### 3.3. Lập hóa đơn tạm tính

public ActionResult Order(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Tables table = db.Tables.Find(id);

if (table == null)

{

return HttpNotFound();

}

ViewBag.TableName = table.TableName;

ViewBag.ListDrinks = db.Drinks.ToList();

ViewBag.ListDrinkCategories = db.Drinks.Select(p => p.DrinkCategories).Distinct().ToList();

TempData["TableID"] = table.TableID;

Bills bills = db.Bills.Where(b => b.TableID == id).FirstOrDefault();

if (bills != null)

{

ViewBag.check = true;

return View(bills.BillInfos.ToList());

}

else

{

ViewBag.check = false;

table.Status = false;

db.Entry(table).State = EntityState.Modified;

db.SaveChanges();

return View(db.BillInfos.ToList());

}

}

### 3.4. Thanh toán

public ActionResult ThanhToan()

{

// Lấy ID bàn từ TempData

int? tableID = TempData["TableID"] as int?;

// Tìm bàn trong cơ sở dữ liệu bằng ID

Tables table = db.Tables.Find(tableID);

// Đặt trạng thái của bàn thành không sử dụng (false)

table.Status = false;

// Tìm hóa đơn tương ứng với bàn và đặt TableID của hóa đơn thành null, đồng thời cập nhật ngày tạo hóa đơn

Bills bills = db.Bills.Where(b => b.TableID == tableID).FirstOrDefault();

bills.TableID = null;

bills.CreateDate = DateTime.Now;

// Đánh dấu hóa đơn đã thay đổi và lưu vào cơ sở dữ liệu

db.Entry(bills).State = EntityState.Modified;

db.SaveChanges();

// Trả về kết quả JSON báo hiệu rằng quá trình thanh toán đã hoàn thành thành công

return Json(new { success = true }, JsonRequestBehavior.AllowGet);

}

### 3.5. Kết ca và in báo cáo

***Kết ca***

[HttpPost]

public ActionResult Index(decimal? lastAmount) // Phương thức Index với tham số lastAmount kiểu decimal có thể là null

{

var shiftReports = SessionConfig.GetShiftReport(); // Lấy thông tin về ca làm việc từ session

if (lastAmount == null) // Kiểm tra xem lastAmount có giá trị hay không

{

ViewBag.Check = 1; // Thiết lập ViewBag.Check thành 1 để đánh dấu có lỗi

ViewBag.Error = "Số tiền kết ca không được để trống"; // Gán thông báo lỗi

return View(shiftReports); // Trả về view Index với thông tin của ca làm việc

}

else

{

shiftReports.LastAmount = (decimal)lastAmount; // Gán giá trị lastAmount vào thuộc tính LastAmount của ca làm việc

shiftReports.LastTime = DateTime.Now; // Gán thời gian kết ca là thời điểm hiện tại

db.Entry(shiftReports).State = EntityState.Modified; // Đánh dấu ca làm việc là đã được sửa đổi trong cơ sở dữ liệu

db.SaveChanges(); // Lưu các thay đổi vào cơ sở dữ liệu

}

return RedirectToAction("ExportPDF"); // Chuyển hướng đến hành động ExportPDF để tạo báo cáo PDF

}

***In báo cáo***

public ActionResult ExportPDF()

{

HtmlToPdf converter = new HtmlToPdf();

converter.Options.PdfPageSize = PdfPageSize.A6;

converter.Options.PdfPageOrientation = PdfPageOrientation.Portrait;

converter.Options.MarginLeft = 10;

converter.Options.MarginRight = 10;

converter.Options.MarginTop = 20;

converter.Options.MarginBottom = 20;

int? tableID = TempData["TableID"] as int?;

Bills bills = db.Bills.Where(b => b.TableID == tableID).FirstOrDefault();

bills.CreateDate = DateTime.Now;

db.Entry(bills).State = EntityState.Modified;

db.SaveChanges();

var list = bills.BillInfos.ToList();

var htmlPDF = base.RenderPartialToString("~/Areas/Admin/Views/Sell/HoaDonTamTinh.cshtml", list);

PdfDocument doc = converter.ConvertHtmlString(htmlPDF);

string fileName = string.Format("{0}.pdf", bills.BillID + "temp");

string pathFile = Path.Combine(Server.MapPath("~/Public/PDF"), fileName);

doc.Save(pathFile);

doc.Close();

return Json(new { success = true }, JsonRequestBehavior.AllowGet);

}

# 

# **KẾT LUẬN**

Trong quá trình thực hiện đề tài "Quản lý quán cà phê", tôi đã đạt được những kết quả quan trọng và đáng chú ý. Dưới đây là một số kết quả đáng kể mà tôi đã đạt được:

* Thiết kế và triển khai giao diện người dùng: đã xây dựng giao diện người dùng hấp dẫn, dễ sử dụng. Giao diện được tối ưu hóa để cung cấp trải nghiệm mua sắm thuận tiện và thú vị cho khách hàng.
* Quản lý sản phẩm: Hệ thống cho phép quản lý thông tin về các sản phẩm trong cửa hàng, bao gồm cập nhật, thêm mới và xóa sản phẩm. Điều này giúp quản lý kho hàng một cách hiệu quả và dễ dàng.
* Quản lý đơn hàng: Hệ thống cung cấp chức năng quản lý đơn hàng từ khi khách hàng đặt hàng cho đến khi hoàn thành thanh toán. Quản lý đơn hàng giúp theo dõi quá trình giao hàng và xử lý các yêu cầu của khách hàng một cách chính xác và nhanh chóng.
* Phân quyền: Người quản lý được phép phân quyền xử dụng với các nhân viên cấp dưới làm đơn giản hóa việc sử dụng ứng dụng, phân quyền dễ dàng và chuẩn xác.
* Báo cáo thống kê: Xử lý được xuất file PDF để xem báo cáo thuận tiện cho việc kiểm kê, so sánh số liệu và so sánh được doanh thu của từng ngày, từng tháng.

# 

# **TÀI LIỆU THAM KHẢO**

|  |  |
| --- | --- |
| [1] | Phạm Thị Kim Ngoan, *Slides Bài giảng Phát triển ứng dụng Web*, Đại học Nha Trang, 2023. |
| [2] | Hồ Thuần, Hồ Cẩm Hà, Trần Thiên Thành, *Cấu trúc dữ liệu, Phân tích thuật toán và Phát triển phần mềm*, NXB Giáo dục, 2008. |
| [3] | Nguyễn Đức Thuần, *Bài giảng Cơ sở dữ liệu*, Đại học Nha Trang, 2008. |
| [4] | N. H. K. Phạm Thị Thu Thuý, *Bài giảng hệ quản trị cơ sở dữ liệu*, Đại học Nha Trang, 2019. |
| [5] | Bùi Chí Thành, Slides Bài giảng Công nghệ phần mềm, Đại học Nha Trang, 2024. |
| [6] | Bùi Chí Thành, Slides Project tham khảo, Đại học Nha Trang, 2024. |