**TÀI LIỆU THIẾT KẾ HƯỚNG ĐỐI TƯỢNG (MÔ HÌNH LỚP)**

Nhóm 02 - Thành viên nhóm

1. Lê Ngọc Dung - Leader
2. Trần Thành Đạt - Note taker
3. Trần Phúc Hưng - Time keeper
4. Phạm Khả Hào - Time keeper
5. Tô Phan Gia Bảo - Speaker

*Tên ứng dụng:*

Thời gian thực hiện: Từ ../../2018 đến ../../2018 (15 tuần)

**MỤC LỤC**

[1. Mô hình lớp thực thể (Class Diagram) 2](#_Toc4575)

[2. Đặc tả sơ đồ lớp: 2](#_Toc10328)

[2.1 Thực thể NhanVien: 2](#_Toc3456)

[2.2 Thực thể Ban: 6](#_Toc18690)

[2.3 Thực thể MonAn 8](#_Toc26490)

[2.4 Thực thể LoaiMonAn 11](#_Toc9036)

[2.5 Thực thể KhachHang 13](#_Toc19353)

[2.6 Thực thể HoaDon 16](#_Toc6935)a

[2.7 Thực thể ChiTietHoaDon 20](#_Toc90)

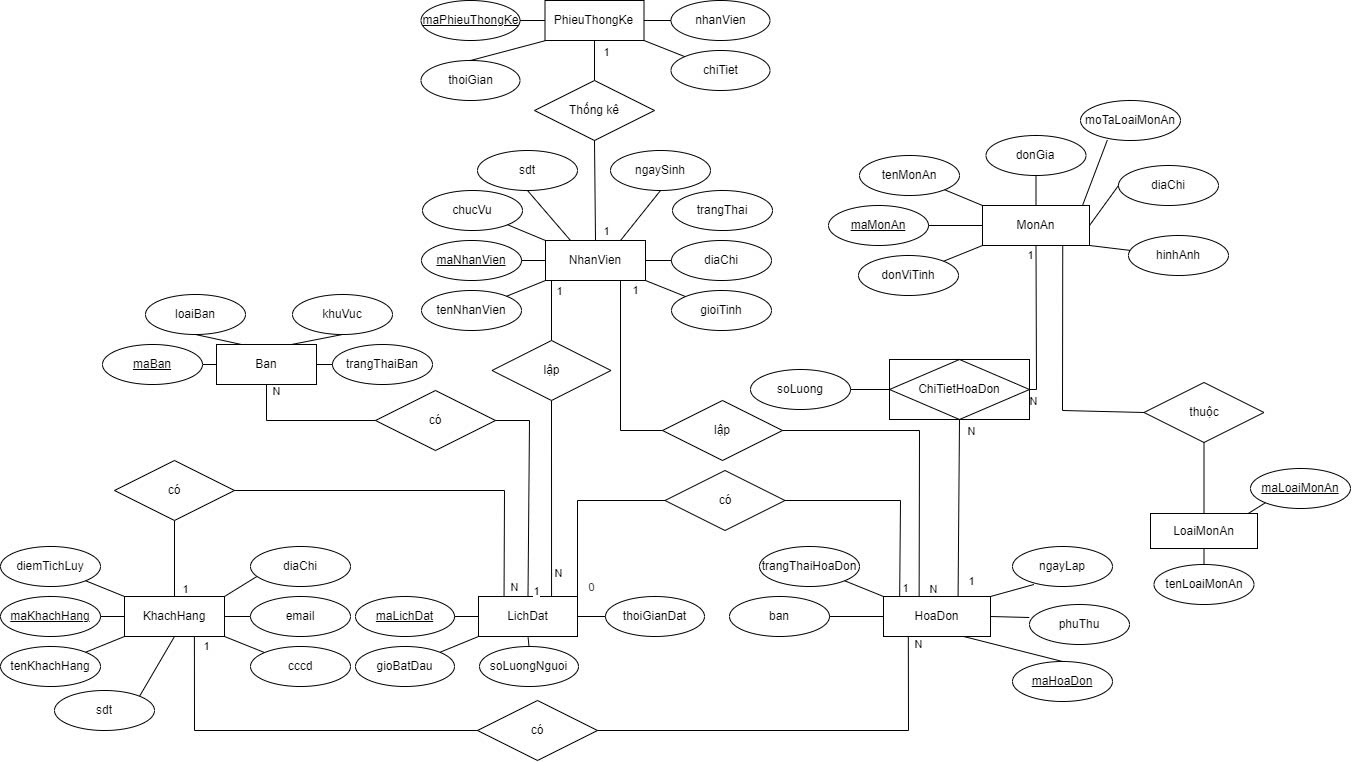
[2.8 Thực thể LichDat 22](#_Toc3723)

[2.9 Thực thể TaiKhoan 25](#_Toc15993)

[2.10 Thực thể PhienThongKe: 27](#_Toc8668)

[*Link nhật ký* 28](#_Toc22043)

1. **Mô hình lớp thực thể (Class Diagram)**

****

1. **Đặc tả sơ đồ lớp:**
   1. Thực thể NhanVien:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |  |
| 1.1 | maNhanVien | **String** |  | Mã nhân viên là một dãy số gồm 6 kí tự có dạng XXYYYY | XX có thể là NV (Nhân viên) hoặc QL(Quản lý), YYYY sẽ là một dãy số tăng dần |
| 1.2 | tenNhanVien | **String** | Họ tên nhân viên | Tên không được bỏ trống, không bao gồm ký tự đặt biệt và viết hoa chữ cái đầu tiên |  |
| 1.3 | sdt | **String** | Số điện thoại | Số điện thoại không được bỏ trống và bắt buộc phải có đủ 10 số |  |
| 1.4 | cccd | **String** | Số căn cước công dân | Gồm 12 số tự nhiên được tạo thành theo quy tắc như sau: 3 số đầu tiên là mã tỉnh, thành phố trực thuộc trung ương hoặc mã quốc gia nơi công dân đăng ký khai sinh. 1 số tiếp theo là mã thế kỷ và giới tính của công dân. 2 số kế tiếp là mã năm sinh (lấy 2 số cuối của năm sinh). Và 6 số cuối là khoảng số ngẫu nhiên. |  |
| 1.5 | diaChi | **String** | Địa chỉ | Địa chỉ không được bỏ trống |  |
| 1.6 | ngaySinh | **LocalDate** | Ngày sinh | Tuổi phải lớn hơn bằng 15 |  |
| 1.7 | gioiTinh | **boolean** | Giới tính | Chỉ có thể là True hoặc False | True là nam và False là nữ |
| 1.8 | trangThai | **TrangThaiNhanVien** |  | Trạng thái nhân viên không được bỏ trống |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |  |
| 2.1 | setMaNhanVien(String maNhanVien) | **void** |  | Mã nhân viên là một dãy số gồm 6 kí tự có dạng XXYYYY  Với XX có thể là NV (Nhân viên) hoặc QL(Quản lý), YYYY sẽ là một dãy số tăng dần | Dãy số tự động tăng |
| 2.2 | setTenNhanVien(String hoten) | **void** |  | Kiểm tra tham số hoten không được rỗng | throw exception “Họ tên không được rỗng” nếu tham số hoten không hợp lệ |
| 2.3 | setSdt(String sdt) | **void** |  | Kiểm tra số điện thoại không được rỗng và đủ 10 số | Throw exception “Số điện thoại không được rỗng và đủ 10 số” nếu tham số sdt không hợp lệ |
| 2.4 | setCccd(String cccd) | **void** |  | Kiểm tra căn cước công dân không rỗng và đủ 12 số | Throw exception “Căn cước công dân không được rỗng và đủ 12 số” nếu tham số cccd không hợp lệ |
| 2.5 | setDiaChi(String diaChi) | **void** |  | Kiểm tra địa chỉ không được bỏ trống | Throw exception “Địa chỉ không được bỏ trống” nếu tham số diaChi không hợp lệ |
| 2.6 | setGioiTinh(boolean gioiTinh) | **void** |  | Kiểm tra giới tính không được bỏ trống | Throw exception “Giới tính không được bỏ trống” nếu tham số gioiTinh không hợp lệ |
| 2.7 | setNgaySinh(LocalDate ngaySinh) | **void** |  | Kiểm tra tuổi nhân viên phải đủ 15 tuổi | Throw exception “Tuổi nhân viên phải đủ 15 tuổi” nếu tham số ngaySinh không hợp lệ |
| 2.8 | setTrangThai(TrangThaiNhanVien trangThai) | **void** |  | Kiểm tra trạng thái không được bỏ trống | Throw exception “Trạng thái không được bỏ trống ” nếu tham số trangThai không hợp lệ |
| **3** | **Viết các constructor** |  |  | * Contructor mặc nhiên * Constructor có đầy đủ tham số * Copy constructor |  |
| **4** | **Viết phương thức toString()** | **string** |  |  | trả về chuỗi chứa giá trị các thuộc tính của nhân viên |

* 1. Thực thể Ban:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |  |
| 1.1 | maBan | **String** | Mã bàn | Mã bàn là một dãy số gồm 6 kí tự có dạng: XYYZZZ | Với X là tên khu vực, YY là sức chứa của bàn và ZZZ là dãy số tự động tăng |
| 1.2 | loaiBan | **LoaiBan** | Loại bàn | Loại bàn không được bỏ trống |  |
| 1.3 | trangThaiBan | **TrangThaiBan** | Trạng thái bàn | Trạng thái bàn không được bỏ trống | Trạng thái bàn bao gồm: bàn trống, đang phục vụ, đã đặt, tạm hỏng |
| 1.4 | khuVuc | **KhuVuc** | Khu vực | Khu vực không được bỏ trống |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |  |
| 2.1 | setMaBan(string maBan) | **void** |  | Mã bàn là một dãy số gồm 6 kí tự có dạng: XYYZZZ  Với X là tên khu vực, YY là sức chứa của bàn và ZZZ là dãy số tự động tăng | Dãy số tự động phát sinh |
| 2.2 | setLoaiBan(LoaiBan loaiBan) | **void** |  | Kiểm tra tham số loaiBan không được rỗng | throw exception “Loại bàn không được rỗng” nếu tham số loaiBan không hợp lệ |
| 2.3 | setTrangThaiBan(TrangThaiBan trangThaiBan) | **void** |  | Kiểm tra tham số trangThaiBan không được rỗng và phải là một trong 3 trạng thái: bàn trống, đang phục vụ, đã đặt, tạm hỏng | Throw exception “Trạng thái bàn không được rỗng” nếu tham số trangThaiBan không hợp lệ |
| 2.4 | setKhuVuc(KhuVuc khuVuc) | **void** |  | Kiểm tra tham số khuVuc không được rỗng | throw exception “Khu vực không được rỗng” nếu tham số khuVuc không hợp lệ |
| **3** | **Viết các constructor** |  |  | * Contructor mặc nhiên * Constructor có đầy đủ tham số * Copy constructor |  |
| **4** | **Viết phương thức toString()** | **string** |  |  | trả về chuỗi chứa giá trị các thuộc tính của nhân viên |

* 1. Thực thể MonAn

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |  |
| 1.1 | loaiMonAn | **LoaiMonAn** | Loại món ăn | Loại món ăn không được để trống |  |
| 1.2 | maMonAn | **String** | Mã món ăn | Mã món ăn là dãy số ký tự có định dạng:  XXXXYYYY | Với XXXX là mã loại món ăn và YYYY sẽ được phát sinh tự động |
| 1.3 | tenMonAn | **String** | Tên món ăn | Tên không được để trống, không có ký tự đặc biệt và viết hoa chữ cái đầu |  |
| 1.4 | donGia | **float** | Đơn giá | Không được để trống, Có giá trị dương và cho phép đến 2 chữ số thập phân |  |
| 1.5 | donViTinh | **String** | Đơn vị tính | Đơn vị tính không được bỏ trống | Đơn vị tính là đơn vị tính của món ăn |
| 1.6 | hinhAnh | **String** | Hình ảnh | Định dạng hình ảnh .jpg hoặc .png |  |
| 1.7 | trangThaiMonAn | **TrangThaiMonAn** |  | Trạng thái món ăn không được để trống | Có 3 trạng thái: Có sẵn, tạm hết, ngừng bán |
| 1.8 | moTaMonAn | **String** | **Mô tả loại món ăn** | Mô tả có thể được để trống |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |  |
| 2.1 | setMaMonAn(string maMonAn) | **void** |  | Mã món ăn là dãy số ký tự có định dạng:  XXXXYYYY. Với XXXX là mã loại món ăn và YYYY sẽ được phát sinh tự động | Dãy số tự động phát sinh |
| 2.2 | setLoaiMonAn(LoaiMonAn loaiMonAn) | **void** |  | Kiểm tra loại món ăn không được rỗng | throw exception “Loại món ăn không được rỗng” nếu tham số loaiMonAn không hợp lệ |
| 2.3 | setTenMonAn(String tenMonAn) | **void** |  | Tên không được để trống, không có ký tự đặc biệt và viết hoa chữ cái đầu | throw exception “Tên món ăn không được rỗng, không ký tự đặc biệt và viết hoa chữ cái đầu ” nếu tham số tenMonAn không hợp lệ |
| 2.4 | setDonGia(float donGia) | **void** |  | Kiểm tra donGia không rỗng, lớn hơn 0 và cho phép đến 2 chữ số thập phân | throw exception “Đơn giá không được để trống ,dương và cho phép 2 chữ số thập phân” nếu tham số donGia không hợp lệ |
| 2.5 | setDonViTinh(String donViTinh) | **void** |  | Không được để trống | throw exception “Đơn vị tính không được để trống” nếu tham số donViTinh không hợp lệ |
| 2.6 | setHinhAnh(String hinhAnh) | **void** |  | Kiểm tra định dạng hình ảnh .jpg và .png | Throw exception “ Định dãng hình ảnh là jpg và .png” nếu tham số hinhAnh không hợp lệ |
| 2.7 | setTrangThaiMonAn(TrangThaiMonAn trangThaiMonAn) | **void** |  | Kiểm tra trạng thái món ăn không được rỗng | throw exception “Trạng thái món ăn không được rỗng” nếu tham số trangThaiMonAn không hợp lệ |
| 2.8 | setMoTaMonAn(String moTaLoaiMonAn) | **void** |  | Mô tả được để trống |  |
| **3** | **Viết các constructor** |  |  | * Contructor mặc nhiên * Constructor có đầy đủ tham số * Copy constructor |  |
| **4** | **Viết phương thức toString()** | **string** |  |  | trả về chuỗi chứa giá trị các thuộc tính của Món ăn |

* 1. Thực thể LoaiMonAn

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |  |
| 1.1 | maLoaiMonAn | **String** | **Mã loại món ăn** | Dãy số gồm 4 ký tự có dạng: XXYY | Với XX là viết tắt của tên loại YY sẽ được phát sinh tự động |
| 1.2 | tenLoaiMonAn | **String** | **Tên loại món ăn** | Tên không được để trống, không có ký tự đặc biệt và viết hoa chữ cái đầu |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |  |
| 2.1 | setMaLoaiMonAn(string maLoaiMonAn) | **void** |  | Dãy số gồm 4 ký tự có dạng: XXYY  Với XX là viết tắt của tên loại YY sẽ được phát sinh tự động | Dãy số tự động phát sinh |
| 2.2 | setTenLoaiMonAn(String tenLoaiMonAn) | **void** |  | Kiểm tra tên loại món ăn không được rỗng,không có ký tự đặc biệt và viết hoa chữ cái đầu | throw exception “Tên loại món ăn không được rỗng, không có ký tự đặc biệt và viết hoa chữ cái đầu” nếu tham số tenLoaiMonAn không hợp lệ |
| **3** | **Viết các constructor** |  |  | * Contructor mặc nhiên * Constructor có đầy đủ tham số * Copy constructor |  |
| **4** | **Viết phương thức toString()** | **string** |  |  | trả về chuỗi chứa giá trị các thuộc tính của Loại món ăn |

* 1. Thực thể KhachHang

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |  |
| 1.1 | maKhachHang | **String** | Mã khách hàng | Khách hàng bao gồm 10 ký tự với định dạng: KHXXXXYYYY | Với XXXX sẽ là năm lưu trữ và YYYY sẽ được phát sinh tự động |
| 1.2 | tenKhachHang | **String** | Tên khách hàng | Tên không được để trống, không có ký tự đặc biệt và viết hoa chữ cái đầu |  |
| 1.3 | sdt | **String** | Số điện thoại của khách hàng | Số điện thoại không được để trống, phải có 10 số và bắt đầu bằng (09, 03, 02, 04) |  |
| 1.4 | cccd | **String** | Căn cước công dân của khách hàng | Gồm 12 số tự nhiên được tạo thành theo quy tắc như sau: 3 số đầu tiên là mã tỉnh, thành phố trực thuộc trung ương hoặc mã quốc gia nơi công dân đăng ký khai sinh. 1 số tiếp theo là mã thế kỷ và giới tính của công dân. 2 số kế tiếp là mã năm sinh (lấy 2 số cuối của năm sinh). Và 6 số cuối là khoảng số ngẫu nhiên. |  |
| 1.5 | diaChi | **String** | Địa chỉ của khách hàng | Có thể để rỗng, nếu không rỗng thì địa chỉ phải bắt đầu bằng số hoặc chữ hoa, các kí tự còn lại có thể là chữ hoặc số hoặc kí tự ‘/’ |  |
| 1.6 | email | **String** | Email của khách hàng | Có thể để rỗng, nếu không rỗng thì email phải kết thúc bằng @gmail.com |  |
| 1.7 | diemTichLuy | **int** | Điểm tích luỹ của khách hàng | Điểm tích luỹ phải lớn hơn hoặc bằng 0. | Điểm tích luỹ sẽ tự động phát sinh khi khách hàng tiến hành order và đặt bàn tại nhà hàng |
| **2** | **Viết các phương thức getter, setter** |  |  |  |  |
| 2.1 | setMaKhachHang(String maKhachHang) | **void** |  | Khách hàng bao gồm 10 ký tự với định dạng: KHXXXXYYYY  Với XXXX sẽ là năm lưu trữ và YYYY sẽ được phát sinh tự động | Dãy số tự động phát sinh |
| 2.2 | setSdt(String sdt) | **void** |  | Kiểm tra số điện thoại không được rỗng, phải có 10 số và bắt đầu bằng (09, 03, 02, 04) | throw exception “Số điện thoại không hợp lệ” nếu không tuân thủ các ràng buộc nêu trên |
| 2.3 | setCCCD(String cccd) | **void** |  | Kiểm tra cccd có tuân thủ các ràng buộc | throw exception “CCCD không hợp lệ” nếu không tuân theo các ràng buộc của hệ thống |
| 2.4 | setDiaChi(String diaChi) | **void** |  | Kiểm tra địa chỉ có tuân theo các ràng buộc | throw exception “địa chỉ không hợp lệ” nếu không tuân theo các ràng buộc của hệ thống |
| 2.5 | setEmail(String email) | **void** |  | Kiểm tra email có tuân theo các ràng buộc | throw exception “email không hợp lệ” nếu không tuân theo các ràng buộc của hệ thống |
| **3** | **Viết các constructor** |  |  | * Contructor mặc nhiên * Constructor có đầy đủ tham số |  |
| **4** | **Viết phương thức toString()** | **string** |  |  | trả về chuỗi chứa giá trị các thuộc tính của KhachHang |

* 1. Thực thể HoaDon

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |  |
| 1.1 | maHoaDon | **String** | Mã hoá đơn | Mã hóa đơn gồm 16 ký tự với định dạng:  HDDDMMYYYYHHMMSS | Với HD là hóa đơn và DDMMYYYYHHMMSS lần lượt là ngày, tháng, năm, giờ, phút và giây lập hóa đơn |
| 1.2 | ngayLap | **LocalDateTime** | Ngày lập hoá đơn | Ngày lập hoá đơn không được trước ngày hiện tại khi tiến hành lập hoá đơn | ngayLap sẽ tự dộng phát sinh khi lập hoá đơn |
| 1.3 | ban | **Ban** | Bàn trong nhà hàng | Không được để trống |  |
| 1.4 | chiTietHoaDon | **List<ChiTietHoaDon>** | Danh sách chi tiết hoá đơn | Không được rỗng, ít nhất 1 chi tiết |  |
| 1.5 | khachHang | **KhachHang** | Thông tin khách hàng trên hoá đơn | Có thể rỗng đối với khách hàng không cần lưu thông tin |  |
| 1.6 | nhanVien | **NhanVien** | Nhân viên lập hoá đơn | Không được để trống | Nhân viên sẽ tự phát sinh khi đăng nhập thành công vào hệ thống |
| 1.7 | trangThaiHoaDon | **TrangThaiHoaDon** | Trạng thái hoá đơn | Không được để trống | Có các trạng thái gồm: đã thanh toán, chưa thanh toán |
| 1.8 | tongTien | **double** | Tổng tiền của hoá đơn | Tổng tiền phải lớn hơn 0 | được dẫn xuất từ tổng các chi tiết hoá đơn |
| 1.9 | phuThu | **double** | Phụ thu | Phụ thu phải lớn hơn hoặc bằng 0 |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |  |
| 2.1 | setMaHoaDon(String Id) | **void** |  | Mã hóa đơn gồm 16 ký tự với định dạng:  HDDDMMYYYYHHMMSS | Dãy số tự động phát sinh |
| 2.2 | setBan(Ban ban) | **void** |  | Kiểm tra trạng thái bàn | throw exception “Bàn không hợp lệ” nếu trạng thái bàn được chọn là đang phục vụ và tạm ngưng phục vụ |
| 2.3 | setKhachHang(KhachHang) | **void** |  | Khách hàng có thể để trống |  |
| 2.4 | setNhanVien(NhanVien nhanVien) | **void** |  | Nhân viên không được để rỗng, nhân viên phải đăng nhập trước khi lập hoá đơn, dữ liệu nhân viên sẽ tự động phát sinh khi đăng nhập thành công | throw exception “đăng nhập để tiếp tục” khi nhân viên chưa đăng nhập vào hệ thống |
| 2.5 | setTrangThaiHoaDon(TrangThaiHoaDon trangThai) | **void** |  | Không được để trống, mặc định là trạng thái chưa thanh toán | throw exception “trạng thái hoá đơn không được để trống” nếu tham số trangThai để trống |
| 2.6 | setPhuThu(double phuThu) | **void** |  | Kiểm tra phụ thu có lớn hơn hoặc bằng 0 | Throw exception “phụ thu phải lớn hơn hoặc bằng 0” nếu phụ thu < 0 |
| 2.7 | setNgayLap(NgayLap ngayLap) | **void** |  | Ngày lập hoá đơn phải là ngày hiện tại | Hệ thống sẽ tự động phát sinh ngày theo thời gian trên hệ thống khi hoá đơn được lập |
| 3 | **Viết các constructor** |  |  | * Contructor mặc nhiên * Constructor có đầy đủ tham số |  |
| 4 | **Viết phương thức toString()** | **String** |  |  | trả về chuỗi chứa giá trị các thuộc tính của HoaDon |
| 4.1 | Viết phương thức tinhTongTien() | **double** | Tổng tiền của hoá đơn |  | Trả về tổng tiền là tổng của các chi tiết hoá đơn |
| 4.2 | Viết phương thức tinhThueVAT() | **double** | Tính thuế VAT |  | Trả về thuế VAT |
| 4.3 | Viết phương thức tinhTongThanhTien() | **double** | Tính tổng thành tiền mà khách hàng phải trả (đã bao gồm VAT) |  | Trả về tổng thành tiền mà khách hàng phải trả |

* 1. Thực thể ChiTietHoaDon

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |  |
| 1.1 | hoaDon | **HoaDon** | Hóa đơn | Hóa đơn không được bỏ trống |  |
| 1.2 | soLuong | **int** | Số lượng | Số lượng phải lớn hơn 0 |  |
| 1.3 | monAn | **MonAn** | Món ăn | Món ăn không được bỏ trống |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |  |
| 2.1 | setHoaDon(HoaDon hoaDon) | **void** |  | Kiểm tra hóa đơn không được để trống | throw exception “Hóa đơn không được để trống ” nếu tham số hoaDon không hợp lệ |
| 2.2 | setSoLuong (int soLuong) | **void** |  | Kiểm tra số lượng không được để trống và lơn hơn 0 | throw exception “Số lượng số lượng không được để trống và >0” nếu tham số soLuong không hợp lệ |
| 2.3 | setMonAn(MonAn monAn) | **void** |  | Kiểm tra món ăn không được để trống | throw exception “Món ăn không được để trống ” nếu tham số monAn không hợp lệ |
| **3** | **Viết các constructor** |  |  | * Contructor mặc nhiên * Constructor có đầy đủ tham số |  |
| **4** | **Viết phương thức toString()** | **string** |  |  | trả về chuỗi chứa giá trị các thuộc tính của Chi tiết món ăn |
| **4.1** | **Viết phương thức layGiaBanMonAn()** | **double** |  |  | Trả về giá trị của giá bán kiểu double |

* 1. Thực thể LichDat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |  |
| 1.1 | maLichDat | **String** | Mã lịch đặt | Mã lịch đặt là một dãy số có 14 ký tự với định dạng:  LDDDMMYYYYXXXX | Với LD là lịch đặt, DDMMYYYY là ngày lập lịch đặt và XXXX sẽ được phát sinh tự động |
| 1.2 | loaiTiec | **LoaiTiec** | Loại tiệc | Chỉ có thể là tiệc cưới hoặc tiệc thường |  |
| 1.2 | thoiGianDat | **LocalDateTime** | Thời gian đặt. | Thời gian đặt không được rỗng và phải trước ngày hiện tại | Nếu là tiệc cưới thì thời gian đặt sẽ trước ngày hiện tại một tháng |
| 1.3 | khachHang | **KhachHang** | Thông tin khách hàng. | Khách hàng có thể rỗng nếu không lưu thông tin |  |
| 1.4 | nhanVien | **NhanVien** | Thông tin nhân viên. | Không được rỗng | Nhân viên sẽ tự phát sinh khi đăng nhập thành công vào hệ thống |
| 1.5 | gioNhanBan | **LocalDateTime** |  | Thời gian bắt đầu sau 7h và trước 23h. | Thời gian sẽ được cập nhật khi nhân viên chọn giờ khách hàng dùng món. |
| 1.6 | soLuongNguoi | **int** | Số lượng khách hàng. | Không được để trống |  |
| 1.7 | ban | **Ban** | Thông tin bàn khi đặt. | Không được để trống. | Hệ thống sẽ tự lấy thông tin của bàn khi mà nhân viên chọn bàn đặt. |
| **2** | **Viết các phương thức getter, setter** |  |  |  |  |
| 2.1 | setMaLichDat(String maLichDat) | **void** |  | Mã lịch đặt là một dãy số có 14 ký tự với định dạng:  LDDDMMYYYYXXXX | Phát sinh tự động |
| 2.2 | setThoiGianDat (LocalDateTime thoiGianDat) | **void** |  | Thời gian đặt không được rỗng và phải trước ngày hiện tại. Nếu là tiệc cưới thì thời gian đặt sẽ trước ngày hiện tại một tháng | throw exception “Thời gian đặt bàn không thỏa điều kiện” nếu chọn thoiGianDat không hợp lệ |
| 2.3 | setKhachHang(KhachHang tenKhachHang) | **void** |  |  | Phần này đã được ràng buộc bên thực thể KhachHang. |
| 2.4 | setNhanVien(NhanVien nhanVien) | **void** |  |  | Phần này đã được ràng buộc bên thực thể NhanVien. |
| 2.5 | setGioBatDat(LocalDateTime gioBatDau) | **void** |  | Kiểm tra tham số giờ bắt đầu phải lớn hơn bằng 7h30 và nhỏ hơn bằng 23h | Throw exception “Thời gian bắt đầu không hợp lệ” nếu như thoiGianBatDau < 7h30 hoặc gioBatDat >=23 |
| 2.6 | setSoLuongNguoi(int soLuongNguoi) | **void** |  | Kiểm tra tham số soLuongNguoi phải lớn hơn 0 | Throw exception “Số lượng người phải lớn hơn 0” nếu soLuongNguoi < 0 |
| 2.7 | setBan (Ban ban) | **void** |  | không |  |
| 3 | Viết các constructor |  |  | * Contructor mặc nhiên * Constructor có đầy đủ tham số |  |
| 4 | Viết phương thức toString() | **String** |  |  | trả về chuỗi chứa giá trị các thuộc tính của LichDat |

* 1. Thực thể TaiKhoan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |  |
| 1.1 | tenTaiKhoan | **String** | Tên tài khoản | Bắt đầu chuỗi, không cho phép bắt đầu bằng số, cho phép chữ cái (hoa và thường), số, và ký tự \_ hoặc  với độ dài từ 6 đến 20 ký tự. | Tên tài khoản là khóa chính |
| 1.2 | matKhau | **String** | Tên khách hàng | Phải có tối đa ̣9 kí tự: có 8 ký tự (chữ hoặc số)và một kí tự đặc biệt |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |  |
| 2.1 | setMaTaiKhoan (String maTaiKhoan) | **void** |  | Phát sinh day số theo quy luật….. | Phát sinh tự động. |
| 2.2 | setTenTaiKhoan (String tenTaiKhoan) | **void** |  | Bắt đầu chuỗi, không cho phép bắt đầu bằng số, cho phép chữ cái (hoa và thường), số, và ký tự \_ hoặc  với độ dài từ 6 đến 20 ký tự. | throw exception “ Tên tài khoản không hợp lệ phải bắt đầu chuỗi, không cho phép bắt đầu bằng số, cho phép chữ cái (hoa và thường), số, và ký tự \_ hoặc  với độ dài từ 6 đến 20 ký tự” |
| 2.3 | setMatKhau (String matKhau) | **void** |  | Kiểm tra tham số matKhau phải có tối đa ̣9 kí tự: có 8 ký tự (chữ hoặc số)và một kí tự đặc biệt | throw exception “Mật khẩu không hợp lệ phải bao gồm 9 ký tự, trong đó phải có ít nhất một ký tự đặc biệt.”. nếu tham số matKhau không hợp lệ |
| 3 | Viết các constructor |  |  | * Contructor mặc nhiên * Constructor có đầy đủ tham số |  |
| 4 | Viết phương thức toString() | **String** |  |  | trả về chuỗi chứa giá trị các thuộc tính của TaiKhoan |

* 1. Thực thể PhienThongKe:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |  |
| 1.1 | maPhienThongKe | **String** | Mã phiên thống kê. | Mã phiên thống kê là một dãy số có 12 ký tự với định dạng:  TKXXXXXXYYYY | Với XXXXXX là mã nhân viên và YYYY sẽ được phát sinh tự động |
| 1.2 | nhanVien | **NhanVien** | Tên nhân viên. | Không được để trống. |  |
| 1.3 | chiTiet | **String** | Chi tiết của phiên thống kê. | Không được rỗng. |  |
| 1.4 | thoiGian | **LocalDateTime** | Thời gian xuất phiên thống kê. | Thoi gian không được trước ngày hiện tại khi tien hành tạo phiên thống kê. | Thời gian sẽ tự dộng phát sinh tạo phiên thống kê. |
| **2** | **Viết các phương thức getter, setter** |  |  |  |  |
| 2.1 | setMaPhienThongKe (String maPhienThongKe) | **void** |  | Mã phiên thống kê là một dãy số có 12 ký tự với định dạng:  TKXXXXXXYYYY | Phát sinh tự động. |
| 2.2 | setNhanVien (NhanVien nhanVien) | **void** |  | Nhân viên không được để rỗng, nhân viên phải đăng nhập trước khi lập hoá đơn, dữ liệu nhân viên sẽ tự động phát sinh khi đăng nhập thành công | throw exception “đăng nhập để tiếp tục” khi nhân viên chưa đăng nhập vào hệ thống |
| 2.3 | setThoiGian (LocalDateTime thoiGian) | **void** |  | Thời gian phải là ngày hiện tại | Hệ thống sẽ tự động phát sinh ngày theo thời gian trên hệ thống tạo phiên thống kê. |
| 2.4 | setChiTiet(String chiTiet) | **void** |  | Không được rỗng. | throw exception “Chi tiết phiên thống kê không có nội dung” khi nhân viên chưa đăng chọn nội dung để thống kê. |
| 3 | Viết các constructor |  |  | * Contructor mặc nhiên * Constructor có đầy đủ tham số |  |
| 4 | Viết phương thức toString() | **String** |  |  | trả về chuỗi chứa giá trị các thuộc tính của PhieuThongKe |

***Link nhật ký***