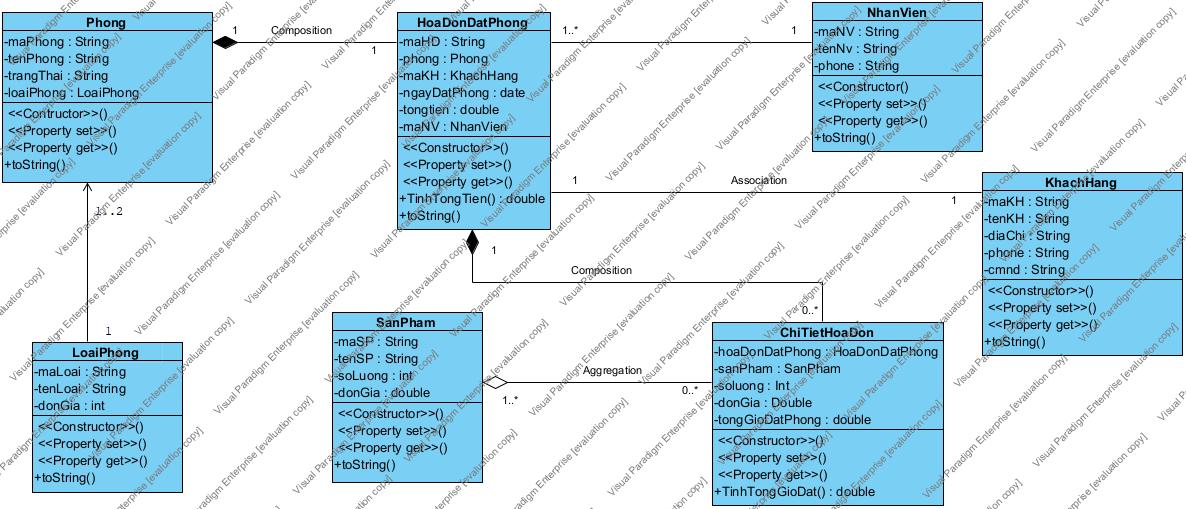
**TÀI LIỆU THIẾT KẾ HƯỚNG ĐỐI TƯỢNG (MÔ HÌNH LỚP)**

Nhóm 04 - Thành viên nhóm:

1. Đặng Phạm Thiên Khải (Nhóm trưởng)
2. Trần Hữu Nhã
3. Trần Nguyễn Hoàng Nam
4. Đinh Lê Hậu Nghĩa

*Tên ứng dụng:* **CHƯƠNG TRÌNH QUẢN LÝ KARAOKE NICE**

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

****

1. **Đặc tả sơ đồ lớp:**
   1. Thực thể Phong:
   2. Thực thể LoaiPhong:
   3. Thực thể HoaDonDatPhong
   4. Thực thể NhanVien
   5. Thực thể KhachHang
   6. Thực thể ChiTietHoaDon
   7. Thực thể SanPham

Table 1: thực thể phòng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maPhong | **String** | Gồm 2 ký tự | **{id}** |
| 1.3 | trangThai | **String** |  |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | getMaPhong() | **void** |  |  |
| 2.2 | setMaPhong(string maPhong) | **void** | Chữ ký đầu là ký tự P, ký tự tiếp theo là ký tự số. |  |
| 2.3 | getTrangThai() | **void** |  |  |
| 2.4 | setTrangThai(string trangThai) | **void** |  |  |
| **3** | **Viết các constructor** |  |  |  |
| 3.1 | Phong() |  |  |  |
| 3.2 | Phong(String maphong, String trangthai) |  |  |  |
| **4** | **Viết phương thức toString()** |  |  |  |
| 4.1 | toString() | **String** |  | Trả về chuỗi chứa giá trị các thuộc tính của nhân viên |

Table 2: thực thể loại phòng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maLoai | **String** | Gồm 2 ký tự |  |
| 1.2 | tenLoai | **String** |  |  |
| 1.3 | donGia | **Int** | > 0 |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | getMaLoai() | **void** |  |  |
| 2.2 | setMaLoai(String maLoai) | **void** | Ký tự đầu tiên là P, ký tự tiếp theo là T hoặc V. |  |
| 2.3 | getTenLoai() | **void** |  |  |
| 2.4 | setTenLoai(String tenLoai) | **void** | T là Thường, V là Vip |  |
| 2.5 | getDonGia() | **Int** |  |  |
| 2.6 | setDonGia(Int donGia) | **Int** | > 0 |  |
| **3** | **Viết các constructor** |  |  |  |
| 3.1 | LoaiPhong() |  |  |  |
| 3.2 | LoaiPhong(String maLoai, String tenLoai, Int donGia) |  |  |  |
| **4** | **Viết phương thức toString()** | **string** |  | trả về chuỗi chứa giá trị các thuộc tính của nhân viên |

Table 3: thực thể hóa đơn đặt phòng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maHD | **String** | Dãy gồm 4 ký tự số | **{id}** |
| 1.2 | phong | **Phong** |  |  |
| 1.3 | maKH | **khachHang** |  |  |
| 1.4 | ngayDatPhong | **Datetime** |  |  |
| 1.5 | tongTien | **double** | Tổng tiền lớn hơn 0 |  |
| 1.6 | maNV | **nhanVien** |  |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaHD(String mahd) | **void** | Phát sinh day số ngẫu nhiên | Phát sinh tự động |
| 2.2 | getMaHD() | **void** |  |  |
| 2.3 | setPhong(Phong phong) | **void** | Phong nhập vào không được trống | throw exception “Phòng không không được rỗng” nếu tham số phong không hợp lệ |
| 2.4 | getPhong() | **void** |  |  |
| 2.5 | setNgayDatPhong(datetime ngayDatPhong) | **void** | Ngày đặt phòng phải lớn hơn hoặc bằng ngày hiện tại | Throw exception “Ngày đặt phòng không được bé hơn so với ngày hiện tại “ |
| 2.6 | getNgayDatPhong() | **void** |  |  |
| 2.7 | setTongTien(double tongTien) | **void** | Tổng tiền phải lớn hơn 0 | Throw exception “Tổng tiền phải lớn hơn không” nếu tham số tongTien không hợp lệ |
| 2.8 | getTongTien() | **void** |  |  |
| **3** | **Viết các constructor** |  | * Contructor mặc nhiên * Constructor có đầy đủ tham số * Copy constructor |  |
| 3.1 | hoaDonDatPhong() |  |  |  |
| 3.2 | hoaDonDatPhong(String maHD, Phong phong,KhacHang maKH,datetime ngayDatPhong, double tongTien,NhanVien maNV) |  |  |  |
| **4** | **Viết phương thức toString()** |  |  |  |
| 4.1 | toString() | **String** |  | trả về chuỗi chứa giá trị các thuộc tính của nhân viên |
| 4.2 | TinhTongTien() | **double** |  | Phương thức tính tổng tiền |

Table 4: thực thể nhân viên

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maNV | **String** | **Gồm 5 kí tự** |  |
| 1.2 | tenNv | **String** | **Không được trống** |  |
| 1.3 | phone | **String** | **Từ 10 đến 11 kí tự số** |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaNV(string maNV) | **void** | Phát sinh theo định dạng gồm 5 kí tự bắt đầu bằng NV sau đó là 3 kí tự số tăng dần. VD: NV001 | Phát sinh tự động |
| 2.2 | getMaNV() | **void** |  |  |
| 2.3 | setTenNV(string hoten) | **void** | Kiểm tra tham số hoten không được rỗng | throw exception “Họ tên không không được rỗng” nếu tham số hoten không hợp lệ |
| 2.4 | setPhone(string phone) | **void** |  |  |
| **3** | **Viết các constructor** |  | * Contructor mặc nhiên * Constructor có đầy đủ tham số * Copy constructor |  |
| 3.1 | NhanVien() |  |  |  |
| 3.2 | NhanVien(String maNV,String tenNV,String phone) |  |  |  |
| **4** | **Viết phương thức toString()** |  |  |  |
| 4.1 | toString() | **string** |  | trả về chuỗi chứa giá trị các thuộc tính của nhân viên |

Table 5: thực thể khách hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maKH | **String** | Dãy số gồm 5 ký tự |  |
| 1.2 | tenKH | **String** | Không được trống |  |
| 1.3 | diaChi | **String** | Không được bỏ trống |  |
| 1.4 | phone | **String** | Gồm 10 đến 11 kí tự số |  |
| 1.5 | cmnd | **String** | Là dẫy số gồm 12 kí tự số |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaKH(string maKH) | **void** | Bắt đầu bằng 2 ký tự chữ KH sau đó là 3 ký tự số  VD : KH001 |  |
| 2.2 | getMaKH() | **void** |  |  |
| 2.3 | setTenKH (String tenKH) | **void** | Kiểm tra tên khách hàng không được rỗng | Throw exception “Tên khách hàng không được rỗng” nếu tham số tenKH không hợp lệ |
| 2.4 | setDiaChi(string diachi) | **void** | Kiểm tra địa chỉ không được rỗng | Throw exception “Địa chỉ khách hàng không được rỗng” nếu tham số diachi không hợp lệ |
| 2.5 | getDiaChi() | **void** |  |  |
| 2.6 | setPhone(string phone) | **void** | Kiểm tra phải bắt đầu bằng ký tự số 0 | Throw exception “ Số điển thoại phải bắt đầu bằng ký tự số 0” nếu tham số phone không hợp lệ |
| 2.7 | getPhone() | **void** |  |  |
| 2.8 | setCmnd(int cmnd) | **void** | Kiểm tra dãy số phải đủ 12 kí tự số. | Throw exception “ Số chứng minh nhân dân phải đủ 12 ký tự số ” nếu tham số cmnd không hợp lệ |
| 2.9 | getCmnd() | **void** |  |  |
| **3** | **Viết các constructor** |  | * Contructor mặc nhiên * Constructor có đầy đủ tham số * Copy constructor |  |
| 3.1 | khachHang() |  |  |  |
| 3.2 | khachHang(String maKH, String tenKH, String diaChi, String phone, String cmnd) |  |  |  |
| **4** | **Viết phương thức toString()** |  |  |  |
|  | **toString()** |  |  |  |
| 4.1 | toString() | **String** |  | trả về chuỗi chứa giá trị các thuộc tính của nhân viên |

Table 6: thực thể chi tiết hóa đơn

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | hoaDonDatPhong | **HoaDonDatPhong** |  |  |
| 1.2 | sanPham | **SanPham** |  |  |
| 1.3 | soluong | **Int** | **Là số tự nhiên ,không được bỏ trống** |  |
| 1.4 | donGia | **Double** |  |  |
| 1.5 | tongGioDatPhong | **Double** |  |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setHoaDonDatPhong(HoaDonDatPhong hoaDonDatPhong) | **void** |  |  |
| 2.2 | getHoaDonDatPhong() | **void** |  |  |
| 2.3 | setSanPham(SanPham sanPham) | **void** |  |  |
| 2.4 | getSanPham() | **void** |  |  |
| 2.5 | setSoLuong(int soLuong) | **void** |  |  |
| 2.6 | getSoLuong() | **void** |  |  |
| 2.7 | setTongGioDatPhong(double tongGioDatPhong) | **void** | kiểm tra tham số: 1.84<=hesoluong<=6.5 | Throw exception “1.84<=hệ số lương <=6.5” nếu tham số hesoluong không hợp lệ |
| 2.8 | getTongGioDatPhong() | **void** |  |  |
| **3** | **Viết các constructor** |  | * Contructor mặc nhiên * Constructor có đầy đủ tham số * Copy constructor |  |
| **3.1** | ChiTietHoaDon() |  |  |  |
| **3.2** | ChiTietHoaDon(HoaDonDatPhong hoaDonDatPhong, SanPham sanPham, int soLuong, Double donGia, Double tongGioDatPhong) |  |  |  |
| **4** | **Viết phương thức toString()** |  |  |  |
| **4.1** | **toString()** | **String** |  | trả về chuỗi chứa giá trị các thuộc tính của ChiTietHoaDon |
| **4.1** | **TinhTongGioDat()** | **Double** |  | Trả về giá trị của TongGioDatPhong |

Table 7: thực thể sản phẩm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maSP | **String** | Dãy số gồm 5 ký tự |  |
| 1.2 | tenSP | **String** | Không được bỏ trống |  |
| 1.3 | soLuong | **int** | Không bỏ trống |  |
| 1.4 | donGia | **Double** |  |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaSP(string maSP) | **void** | Phát sinh day số theo cách 2 kí tự đầu là SP sau đó là 3 kí tự số. | Phát sinh tự động |
| 2.2 | getMaSP() | **void** |  |  |
| 2.3 | setTenSP(String tenSP) | **void** |  | Throw exception “Tên sản phẩm không được bỏ trống” nếu tham số tenSP không hợp lệ |
| 2.4 | getTenSP() | **void** |  |  |
| 2.5 | setSoLuong (int soLuong) | **void** | Là số tự nhiên và không được bỏ trống | Throw exception”Số lượng không được để trống” nếu tham số soLuong không hợp lệ |
| 2.6 | getSoLuongI() | **void** |  |  |
| 2.7 | setDonGia(double donGia) | **void** |  |  |
| 2.8 | getDonGia() | **void** |  |  |
| **3** | **Viết các constructor** |  | * Contructor mặc nhiên * Constructor có đầy đủ tham số * Copy constructor |  |
| 3.1 | SanPham() |  |  |  |
| 3.2 | SanPham(String maSP, String tenSP, int soLuong, Double donGia) |  |  |  |
| **4** | **Viết phương thức toString()** |  |  |  |
|  | **toString()** |  |  |  |
| 4.1 | toString() | **String** |  | trả về chuỗi chứa giá trị các thuộc tính của sản Phẩm. |