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

Nhóm 05 - Thành viên nhóm:

1. Trần Quốc Đảm (Facilitacator)

2. Nguyễn Danh Minh Toàn (Reporter)

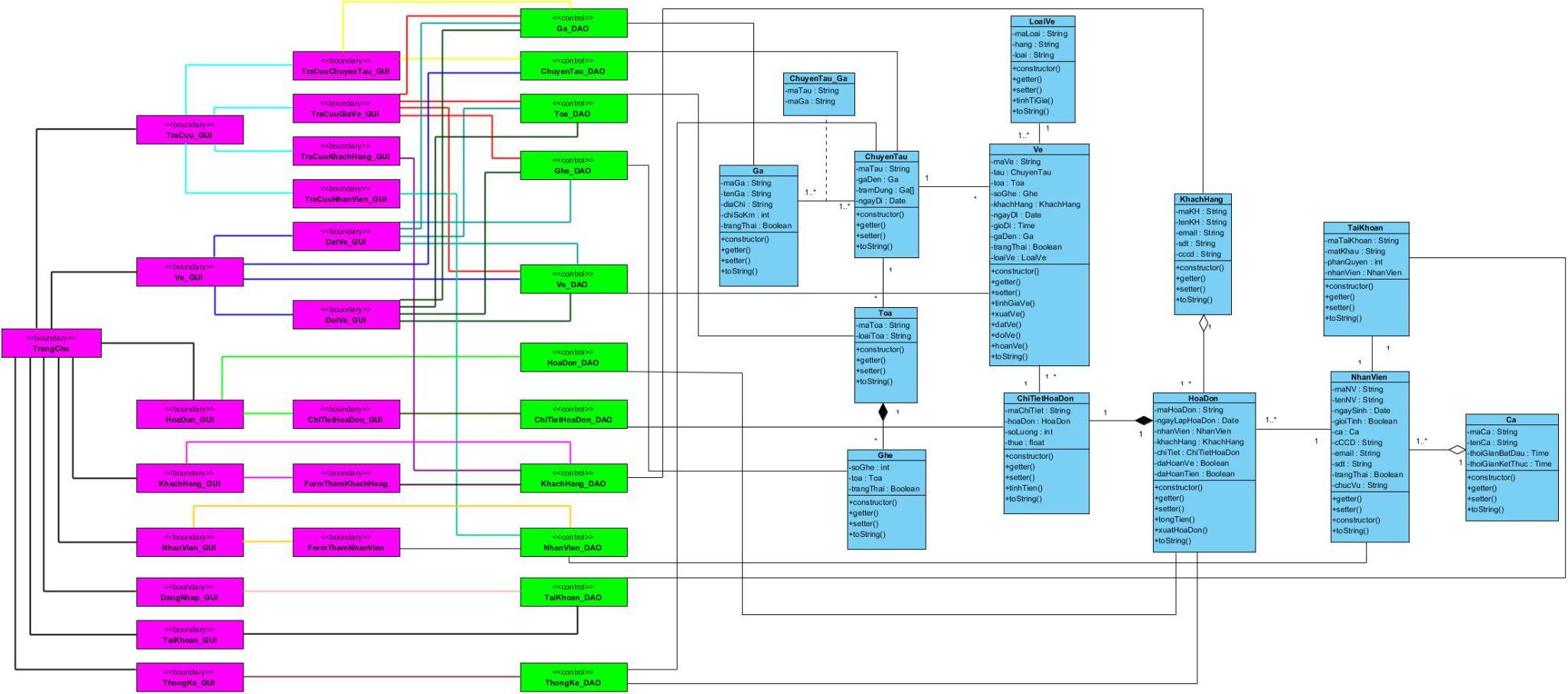
3. Lê Tấn Phong (Timekeeper)

4. Vương Ngọc Huệ (Note tasker)

*Tên ứng dụng:* **CHƯƠNG TRÌNH QUẢN LÝ BÁN VÉ TẠI GA ĐTHP**

Thời gian thực hiện: Từ 16/8/2024 đến 21/10/2024 (10 tuần)

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



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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| 1 | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maGa | **String** | Dãy số gồm 5 ký tự, 2 ký tự đó là GA, 3 ký tự sau là XXX, 3 ký tự sau là ký tự số được đánh thứ tự, ví dụ: GA001, GA002 | {id}: |
| 1.2 | tenGa | **String** | Ký tự đầu tiên sau dấu cách viết hoa. Không có ký tự đặc biệt. Không được bỏ trống |  |
| 1.3 | diaChi | **String** | Là dãy ký tự là địa chỉ của ga đến. |  |
| 1.4 | trangThai | **Boolean** |  | False: không khả dụng  True: khả dụng |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaGa (String maGa) | **void** | Phát sinh theo quy luật: Dãy số gồm 5 ký tự, 2 ký tự đó là GA, 3 ký tự sau là XXX, 3 ký tự sau là ký tự số được đánh thứ tự, ví dụ: GA001, GA002 | Tự động phát sinh |
| 2.2 | setTenGa (String tenGa) | **void** | Kiểm tra tham chiếu TenGa có trong cơ sở dữ liệu Ga hay không | Throw exception “Tên ga không tồn tại” nếu tham số tenGa không hợp lệ. |
| 2.3 | setDiaChi (String diaChi) | **void** | Kiểm tra tham chiếu DiaChi có trong cơ sở dữ liệu Ga hay không | Throw exception “Địa chỉ không tồn tại” nếu tham số diaChi không hợp lệ. |
| 2.4 | setTrangThai (boolean trangThai) | **void** |  |  |
| **3** | **Viết các constructor** |  | * Constructor 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 ga |

* 1. Thực thể ChuyenTau:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| 1 | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maTau | **String** | Dãy số gồm 5 ký tự, 2 ký tự đầu là TA, 3 ký tự còn lại là XXX là 1 số được đánh theo thứ tự của tàu, ví dụ: TA001,TA002,... | {id}: thuộc tính định danh |
| 1.2 | gaDen | **Ga** | Là 1 dãy kí tự là tên ga sẽ đến, Sẽ được lấy từ thực thể Ga và chỉ được lưu khi dữ liệu có trong thực thể Ga. |  |
| 1.3 | tramDung | **Ga[]** |  |  |
| 1.4 | ngayDi | **Date** | Là thời gian được quy định trước. |  |
| 1.5 | gioDi | **Time** |  |  |
| 2 | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | **setMaTau (String maTau)** | **void** | Phát sinh theo quy luật: Dãy số gồm 5 ký tự, 2 ký tự đầu là TA, 3 ký tự còn lại là XXX là 1 số được đánh theo thứ tự của tàu, ví dụ: TA001,TA002,... | Tự phát sinh. |
| 2.2 | **setGaDen (Ga gaDen)** | **void** | Kiểm tra tham chiếu GaDen có trong cơ sở dữ liệu ChuyenTau hay không. | Throw exception “Ga đến không tồn tại” nếu tham số gaDen không hợp lệ. |
| 2.3 | **setTramDung (Ga** |  | Kiểm tra tham số TramDung xem có trong cơ sở dữ liệu Ga hay không | Throw exception “Trạm dừng không tồn tại” nếu tham số tramDung không hợp lệ. |
| 2.4 | **setNgayDi (date ngayDi)** | **void** | Kiểm tra tham chiếu NgayDi có trong cơ sở dữ liệu ChuyenTau không.  Sau ngày hiên tại. | Throw exception “Ngày đi không tồn tại” nếu tham số ngayDi không hợp lệ. |
| 2.5 | **setGioDi (Time gioDi)** | **void** | Kiểm tra tham chiếu GioDi có trong cơ sở dữ liệu ChuyenTau không. |  |
| **3** | **Viết các constructor** |  | * Constructor 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 chuyến tàu |

* 1. Thực thể Toa:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maToa | **String** | Dãy số gồm 8 ký tự, 5 ký tự đó là mã tàu, 1 ký tự tiếp theo là “\_”, 2 ký tự cuối là ký tự số được đánh thứ tự, ví dụ: TA001\_01, TA001\_02, TA002\_01, TA002\_02,... | {id}: thuộc tính định danh |
| 1.2 | loaiToa | **String** | Là dãy ký tự là loại của toa, ví dụ VIP, ghế mềm và giường nằm. |  |
| 2 | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaToa (String maToa) | **void** | Phát sinh theo quy luật: Dãy số gồm 8 ký tự, 5 ký tự đó là mã tàu, 1 ký tự tiếp theo là “\_”, 2 ký tự cuối là ký tự số được đánh thứ tự, ví dụ: TA001\_01, TA001\_02, TA002\_01, TA002\_02, … | Throw exception “Mã toa không tồn tại” nếu tham số maToa không hợp lệ. |
| 2.2 | setLoaiToa (String loaiToa) | **void** | Là 1 trong 3 dãy:   * Vip * Ghế mềm * Ghế nằm | Throw exception “Ngày đi không tồn tại” nếu tham số ngayDu không hợp lệ. |
| **3** | **Viết các constructor** |  | * Constructor 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 Toa |

* 1. Thực thể Ghe:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| 1 | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | soGhe | **String** | Là 1 ký tự số, và được nhập theo thứ tự, ví dụ: 1,2,3,4,… | {id}: thuộc tính định danh |
| 1.2 | maToa | **String** | Là dãy ký tự là mã của toa đang chứa ghế | {id}: thuộc tính định danh |
| 1.3 | trangThai | **Boolean** | Là 2 giá trị “TRUE” nếu chưa được đặt và “FALSE” nếu đã được đặt |  |
| 2 | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaGhe (String maGhe) | **void** | Phát sinh theo quy luật: Là 1 ký tự số, và được nhập theo thứ tự, ví dụ: 1,2,3,4,… | Throw exception “Mã ghế không tồn tại” nếu tham số maGhe không hợp lệ. |
| 2.2 | setMaToa (String maToa) | **void** | Kiểm tra tham chiếu MaToa có trong cơ sở dữ liệu Ghe hay không. | Throw exception “Mã toa không tồn tại” nếu tham số maToa không hợp lệ. |
| 2.5 | setTrangThai (Boolean trangThai) | **void** |  |  |
| **3** | **Viết các constructor** |  | * Constructor 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 Ghế |

* 1. Thực thể LoaiVe:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maLoai | **String** | Chỉ thuộc 1 trong 15 dạng sau:  LVGME1, LVGME2, LVGMNL, LVGMCT, LVGMSV, LVGNE1, LVGNE2, LVGNNL, LVGNCT, LVGNSV, LVVIE1, LVVIE2, LVVINL, LVVICT, LVVISV | **{id}** |
| 1.2 | hang | **String** |  |  |
| 1.3 | loai | **String** |  |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaLoai(String maLoai) | **void** | Chuỗi gồm 6 ký tự theo quy luật “LV” + “....” ( thay “....” bằng các từ viết tắt rút gọn của hạng + loại) | throw exception “Mã vé không tồn tại” nếu tham số maLoai không hợp lệ |
| 2.2 | setHang(String hang) | **void** | Kiểm tra tham số hang thuộc: Ghế mềm (GM), giường nằm (GN), VIP (VI) | throw exception “Hạng vé không tồn tại” nếu tham số hang không hợp lệ |
| 2.3 | setLoai(String loai) | **void** | Kiểm tra tham số loai thuộc: Trẻ em dưới 6 tuổi (E1), trẻ em 6 đến 10 tuổi (E2), người lớn (NL), người cao tuổi (CT), sinh viên (SV) | Throw exception “Loại vé không tồn tại” nếu tham số loai không hợp lệ |
| **3** | **Viết các constructor** |  | * Constructor 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 vé |

Ghi chú:

* 1. Thực thể Ve:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maVe | **String** |  | **{id}** |
| 1.2 | maTau | **ChuyenTau** |  |  |
| 1.3 | maToa | **Toa** |  |  |
| 1.4 | soGhe | **Ghe** | 0 <= soGhe <= 64 |  |
| 1.5 | khachHang | **KhachHang** |  |  |
| 1.6 | ngayDi | **Date** |  |  |
| 1.7 | gioDi | **Time** |  |  |
| 1.8 | gaDen | **Ga** |  |  |
| 1.9 | trangThai | **Boolean** |  | True: Đã hoàn thành chuyến  False: Chưa hoàn thành chuyến |
| 1.10 | loaiVe | **LoaiVe** |  |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaVe(string maVe) | **void** | Phát sinh dãy số theo quy luật:  - 2 ký tự đầu là 2 chữ số thể hiện ngày lập vé  - 2 ký tự tiếp là 2 chữ số thể hiện tháng lập vé  - 2 ký tự tiếp là 2 chữ số thể hiện hai số cuối của năm lập vé  - 4 số cuối là số thứ tự lập vé được tăng tự động từ 0001 đến 9999 | Phát sinh tự động  Ví dụ: mã vé cuối cùng trong CSDL có mã vé là 2209240001, vậy mã vé mới được thêm có mã mã vé là 2209240002. |
| 2.2 | setMaTau(ChuyenTau chuyenTau) | **void** | Kiểm tra tham số chuyenTau có tồn tại trong cơ sở dữ liệu ChuyenTau không | throw exception “Chuyến tàu không tồn tại” nếu tham số chuyenTau không hợp lệ |
| 2.3 | setMaToa(Toa toa) | **void** | Kiểm tra tham số toa có tồn tại trong cơ sở dữ liệu Toa không | throw exception “Toa không tồn tại” nếu tham số toa không hợp lệ |
| 2.4 | setSoGhe(Ghe ghe) | **void** | Kiểm tra tham số ghe có tồn tại trong cơ sở dữ liệu Ghe không | throw exception “Ghế không tồn tại” nếu tham số ghe không hợp lệ |
| 2.5 | setKhachHang(KhachHang khachHang) | **void** | Kiểm tra tham số khachHang có tồn tại trong cơ sở dữ liệu KhachHang không | throw exception “Khách hàng không tồn tại” nếu tham số khachHang không hợp lệ |
| 2.6 | setGaDen(Ga gaDen) | **void** | Kiểm tra tham số gaDen có tồn tại trong cơ sở dữ liệu Ga không | throw exception “Ga không tồn tại” nếu tham số gaDen không hợp lệ |
| 2.7 | setTrangThai(boolean trangThai) | **void** |  |  |
| 2.8 | setLoaiVe(LoaiVe loaiVe) | **void** | Kiểm tra tham số loaiVe có tồn tại trong cơ sở dữ liệu LoaiVe không | throw exception “Loại vé không hợp lệ” nếu tham số loaiVe không tồn tại |
| **3** | **Viết các constructor** |  | * Constructor 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 Vé |

* 1. Thực thể HoaDon:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maHoaDon | **String** | Không được bỏ trống | {id} |
| 1.2 | ngayLapHoaDon | **Date** | Ngày hiện tại. Không được bỏ trống |  |
| 1.3 | nhanVien | **NhanVien** |  |  |
| 1.4 | khachHang | **KhachHang** |  |  |
| 1.5 | chiTiet | **ChiTietHoaDon** |  |  |
| 1.6 | daHoanVe | **Boolean** |  | Mặc định là false.  +False: Không hoàn vé  +True: Đã hoàn vé |
| 1.7 | daTraVe | **Boolean** |  | Mặc định là false.  +False: Không trả vé  +True: Đã trả vé |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaHoaDon(String maHoaDon) | **void** | Phát sinh dãy số theo quy luật.  Mã hóa đơn có dạng:  “DDMMYY” + “NVXX” + “TTTTT”  với:  + DD là ngày , MM là tháng , YY là 2 số cuối năm của ngày lập hóa đơn  +NVXXX là Mã nhân viên thực hiện  +TTTTT là dãy số tăng tự động | Phát sinh tự động  VD:220924NV00100001 |
| 2.2 | setNgayLapHoaDon(LocalDateTime ngayLapHoaDon) | **void** | Ngày hiện tại | Tự động cập nhật ngày hiện tại |
| 2.3 | setNhanVien(NhanVien nhanVien) | **void** | Kiểm tra tham số nhanVien có tồn tại | Throw exception “Nhân viên không tồn tại” nếu tham số nhanVien không hợp lệ |
| 2.4 | setKhachHang(KhachHang khachHang) | **void** | Kiểm tra tham số khachHang có tồn tại | Throw exception “Khách hàng không tồn tại” nếu tham số nhanVien không hợp lệ |
| 2.5 | setChiTiet(ChiTietHoaDOn chiTiet) | **void** | Kiểm tra tham số chiTiet có tồn tại | Throw exception “Chi tiết hóa đơn không tồn tại” nếu tham số nhanVien không hợp lệ |
| 2.6 | setDaHoanTien(Boolean daHoanTien) | **void** | Kiểm tra tham số daHoanTien là false | Throw exception “Hóa đơn đã hoàn vé” nếu tham số daHoanTien là true |
| 2.7 | setDaTraVe(Boolean) | **void** | Kiểm tra tham số daTraVe là false | Throw exception “Hóa đơn đã hoàn vé” nếu tham số daTraVe là true |
| **3** | **Viết các constructor** |  | * Constructor 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 hóa đơn |

* 1. Thực thể ChiTietHoaDon:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maChiTiet | **String** | Không được bỏ trống | {id} |
| 1.2 | hoaDon | **HoaDon** |  |  |
| 1.3 | soLuong | **int** | 0 < soLuong <=4 Không được bỏ trống |  |
| 1.4 | dsVe | **ArrayList<Ve>** | Không được bỏ trống |  |
| 1.5 | thue | **float** |  |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaChiTiet(String maChiTiet) | **void** | Phát sinh dãy số theo quy luật.  Mã chi tiết hóa đơn có dạng:  “CT” + Mã hóa đơn | Phát sinh tự động |
| 2.2 | setHoaDon(HoaDon hoaDon) | **void** | Kiểm tra tham số maHoaDon có tồn tại.  theo quy luật: HDddMMYYYYxxx.Trong đó: ddMMYYYY là ngày tháng năm phát sinh hóa đơn, xxx tạo thành 1 số nguyên, khi thêm một hoá đơn mới thì lấy xxxxx trong hoá đơn cuối chuyển thành số nguyên và tăng lên 1. | Phát sinh tự động  Vd: hoá đơn trong ngày cuối cùng trong CSDL có mã hoá đơn là HD22092024001, vậy hoá đơn mới được thêm có mã hoá đơn là HD22092024002. |
| 2.3 | setSoLuong(int soLuong) | **void** | Kiểm tra tham số soLuong sao cho 0 > soLuong <=4 |  |
| 2.4 | setDsVe(ArrayList<Ve> dsVe) | **void** | Không được bỏ trống |  |
| 2.5 | setThue(float thue) | **void** | Không được bỏ trống |  |
| **3** | **Viết các constructor** |  | * Constructor 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 chi tiết hóa đơn |

* 1. Thực thể KhachHang:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maKH | **String** | Không được bỏ trống. | {id} |
| 1.2 | tenKH | **String** | Ký tự đầu tiên sau dấu cách viết hoa. Không có ký tự đặc biệt. Không được bỏ trống |  |
| 1.3 | email | **String** | Ký tự bắt buộc @.Trước ký tự @ ,cho phép bắt đầu bằng các ký tự chữ cái thường và hoa (a-zA-Z), chữ số (0-9), dấu chấm (.). Sau ký tự @, kết thúc bằng “gmail.com” |  |
| 1.4 | sdt | **String** | Dãy 10 ký tự số. Bắt đầu bằng 2 ký tự “03”,”05”,”08”,”07”,”09”.Tiếp đó là 8 ký tự số từ (0-9) |  |
| 1.5 | cccd | **String** | Dãy 12 ký tự số có dạng XXXYZZTTTTTT với:  +XXX là mã tỉnh/thành phố. VD: 079( thành phố Hồ Chí Minh)  + Y là mã giới tính:  VD:  0: Nam sinh trong thế kỷ 20 (1900 - 1999)  1: Nữ sinh trong thế kỷ 20  2: Nam sinh trong thế kỷ 21 (2000 - 2099)  3: Nữ sinh trong thế kỷ 21  +ZZ: hai số cuối năm sinh. VD: 04(2004)  +TTTTTT: sáu ký tự số ngẫu nhiên |  |
| **2** | **Viết các phương thức getter, setter** |  | * Constructor mặc nhiên * Constructor có đầy đủ tham số. * Copy constructor |  |
| 2.1 | setMaKH(String maKH) | **void** | Phát sinh day số theo quy luật KHXXXX với XXXX là 4 chữ số tăng dần. | Phát sinh tự động.  VD: KH0001,KH0002 |
| 2.2 | setTenKH(String tenKH) | **void** | Kiểm tra tham số tenKH không được rỗng | throw exception “Họ tên không được rỗng” nếu tham số tenKH không hợp lệ |
| 2.3 | setEmail(String email) | **void** | Kiểm tra tham số email theo quy luật.Trước ký tự @ ,cho phép bắt đầu bằng các ký tự chữ cái thường và hoa (a-zA-Z), chữ số (0-9), dấu chấm (.).Ký tự @ bắt buộc. Sau ký tự @, kết thúc bằng “gmail.com”. | Throw exception “Email không được rỗng ” nếu tham số email không hợp lệ |
| 2.4 | setCccd(String cccd) | **void** | Kiểm tra tham số cccd gồm 12 chữ số | Throw exception “Cccd không được rỗng ” nếu tham số cccd không hợp lệ |
| **3** | **Viết các constructor** |  |  |  |
| **4** | **Viết phương thức toString()** | **String** |  | trả về chuỗi chứa giá trị các thuộc tính của Khách hàng |

* 1. Thực thể TaiKhoan:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maTaiKhoan | **String** |  | **{id}** |
| 1.2 | matKhau | **String** |  |  |
| 1.3 | phanQuyen | **Int** |  | 0: quản lý.  1: nhân viên. |
| 1.4 | maNV | **NhanVien** |  |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaTaiKhoan(String maTaiKhoan) | **void** | TKQL001  TKNV001 |  |
| 2.2 | setMatKhau(String matKhau) | **void** | Kiểm tra tham số mật khẩu không được rỗng. Sử dụng tối thiểu 8 ký tự, và tối đa 15 ký tự. Bao gồm số, chữ thường, chữ in hoa và ký tự đặc biệt. |  |
| 2.3 | setPhanQuyen(Int phanquyen) | **void** |  |  |
| 2.4 | setMaNhanVien(NhanVien maNhanVien) | **void** | Kiểm tra tham số mã nhân viên không được rỗng |  |
| **3** | **Viết các constructor** |  | * Constructor 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 Tài Khoản |

* 1. Thực thể NhanVien:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maNV | **String** |  | **{id}** |
| 1.2 | tenNV | **String** |  |  |
| 1.3 | ngaySinh | **Date** |  |  |
| 1.4 | gioiTinh | **Boolean** |  | False: nam  True: nữ |
| 1.5 | maCa | **Ca** |  |  |
| 1.6 | CCCD | **String** |  |  |
| 1.7 | email | **String** | Ký tự bắt buộc @.Trước ký tự @ ,cho phép bắt đầu bằng các ký tự chữ cái thường và hoa (a-zA-Z), chữ số (0-9), dấu chấm (.). Sau ký tự @, kết thúc bằng “gmail.com” |  |
| 1.8 | sDT | **String** |  |  |
| 1.9 | trangThai | **Boolean** |  | True: đang làm.  False: nghỉ việc |
| 1.10 | chucVu | **int** |  | 1: quản lý  2: nhân viên  ... (nhiều hơn nếu còn các chức vụ khác) |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaNV(String manv) | **void** | theo quy luật: NVxxx  Trong đó: xxx tạo thành 1 số nguyên, khi thêm một nhân viên mới thì lấy xxx trong nhân viên cuối chuyển thành số nguyên và tăng lên 1. | Phát sinh tự động  Ví dụ: Nhân viên cuối cùng trong CSDL có mã nhân viên là NV001, vậy nhân viên mới được thêm có mã nhân viên là NV002. |
| 2.2 | setTenNV(String tennv) | **void** | Kiểm tra tham số TenNV không được rỗng | Throw exception “tên nhân viên không không được rỗng” nếu tham số tenNV không hợp lệ. |
| 2.3 | setNgaySinh(date ngaysinh) | **void** | Nam: 18 <= ngayHienTai - ngaySinh < 61  Nữ: 18 <= ngayHienTai - ngaySinh < 56 | Throw exception “Nhân viên phải >=18 tuổi” nếu tham số ngaysinh không hợp lệ |
| 2.4 | setGioiTinh(Boolean gioiTinh) | **void** |  |  |
| 2.5 | setMaCa(Ca maCa) | **void** | Kiểm tra tham số Ca không được rỗng |  |
| 2.6 | SetCCCD(String cccd) | **void** | Chuỗi ký tự gồm 12 chữ số, bắt đầu bằng số 0 |  |
| 2.7 | setEmail(String email) | **void** | Kiểm tra tham số Email không được rỗng |  |
| 2.8 | setSDT(String sdt) | **void** | Bắt đầu bằng số 0, gồm 10 chữ số |  |
| 2.9 | setTrangThai(boolean trangthai) | **void** | Kiểm tra tham chiếu trangThai trong cơ sở dữ liệu TRUE hay FALSE |  |
| 2.10 | setChucVu(int chucvu) | **void** |  |  |
| **3** | **Viết các constructor** |  | * Constructor 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 Ca:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maCa | **String** |  | {id}, có 3 ca:  Ca 1, Ca 2, Ca 3 |
| 1.2 | tenCa | **String** |  |  |
| 1.3 | thoiGianBatDau | **Time** |  | Ca 1: Bắt đầu 6 giờ  Ca 2: Bắt đầu 14 giờ  Ca 3: Bắt đầu 22 giờ |
| 1.4 | thoiGianKetThuc | **Time** |  | Ca 1: Kết thúc 14 giờ Ca 2: Kết thúc 22 giờ  Ca 3: Kết thúc 6 giờ |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaCa(String maCa) | **void** | Theo quy luật: CAxx  Trong đó: xx tạo thành 1 số nguyên, khi thêm một ca mới thì lấy xx trong ca cuối chuyển thành số nguyên và tăng lên 1. |  |
| 2.2 | setTenCa(String tenCa) | **void** | Kiểm tra tham số TenCa không được rỗng |  |
| 2.3 | setThoiGianBatDau(Time thoigianbatdau) | void |  |  |
| 2.4 | set thoiGianKetThuc (Time thoigianketthuc) | void |  |  |
| **3** | **Viết các constructor** |  | * Constructor 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 Ca |

***Link nhật ký:*** [**05\_1 Application Development NHATKY**](https://docs.google.com/document/d/18TTzJDMYPDMKaH2OqKSEt167H9DLMVZQ2oSEHbF1XgM/edit)