Project name: Tuong House

Software Requirement Specification

**Project Code: TTNTHouse**

**Document Code: MIS30071.01**

**Danang, MAY 2022**

Record of change

\*A - Added M - Modified D - Deleted

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Effective Date | Changed Items | A\*  M, D | Change Description | New Version |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

SIGNATURE PAGE

**ORIGINATOR:** MIS30071.01 - Team 1 - 46k21.3

Nguyễn Tiến Sỹ

Hồ Thanh Tuấn

Nguyễn Quốc Nguyên

Team Leader: Nguyễn Quốc Nguyên

**REVIEWERS:** Khách hàng anh Nguyễn Tường

**Table of contents.**

[**1** **Introduction** 5](#_Toc103518757)

[**1.1** **Purpose** 5](#_Toc103518758)

[**1.2** **Scope** 5](#_Toc103518759)

[**1.3** **Overview** 5](#_Toc103518760)

[**1.4** **Definitions, Acronyms, and Abbreviations** 5](#_Toc103518761)

[**1.5** **References** 6](#_Toc103518762)

[**2** **Requirement DEFINITION** 6](#_Toc103518763)

[**2.1** **Non-functional requirements** 6](#_Toc103518764)

[**2.1.1** **Operational requirements** 6](#_Toc103518765)

[**2.1.2** **Performance requirements** 6](#_Toc103518766)

[**2.1.3** **Security requirements** 7](#_Toc103518767)

[**2.1.4** **Cultural and political requirements** 7](#_Toc103518768)

[**2.1.5** **Performance Requirements** 7](#_Toc103518769)

[**2.1.6** **…** 7](#_Toc103518770)

[**3** **Functional Requirements** 7](#_Toc103518771)

[**4** **Functional model** 7](#_Toc103518772)

[**4.1** **OVERALL DESCRIPTION** 8](#_Toc103518773)

[**4.2** **Use case description** 8](#_Toc103518774)

[**4.2.1** **Thuê phòng** 8](#_Toc103518775)

[**4.2.2** **Trả phòng** 10](#_Toc103518776)

[**4.2.3 Thống kê** 12](#_Toc103518777)

[**4.3** **Open Issues** 14](#_Toc103518778)

[**5** **Structural model** 14](#_Toc103518779)

[**5.1** **Class diagram** 14](#_Toc103518780)

[**6** **Behavioral model** 19](#_Toc103518781)

[**6.1 Sequence diagram** 20](#_Toc103518782)

# **1** **Introduction**

## **1.1** **Purpose**

* Xây dựng ứng dụng quản lý phòng trọ
* Thu thập yêu cầu của Khách hàng về hệ thống.
* Phân tích yêu cầu, tập hợp thông tin được thu tập từ đó xác định được mong muốn của khách hàng cho hệ thống đó.

## **1.2** **Scope**

N/A

## **1.3** **Overview**

* Hệ thống quản lý phòng trọ. Gồm quản lý: phòng, khách hàng, dịch vụ và thắc mắc.

## **1.4** **Definitions, Acronyms, and Abbreviations**

|  |  |  |
| --- | --- | --- |
| **No** | **Abbreviation** | **Description** |

|  |  |  |
| --- | --- | --- |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |
| 6 |  |  |

## **1.5** **References**

|  |  |  |
| --- | --- | --- |
| **No** | **Abbreviation** | **Description** |

|  |  |  |
| --- | --- | --- |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |

# **2** **Requirement DEFINITION**

## **2.1** **Non-functional requirements**

### **2.1.1** **Operational requirements**

* Hệ thống hoạt động trên nền tảng web và mobile (cho điện thoại android, Iphone, ipad).

### **2.1.2** **Performance requirements**

N/A

### **2.1.3** **Security requirements**

* Hệ thống được phân quyền (có thể cấp quyền admin cho các user được chỉ định quản lý hệ thống).
* Kết hợp nhiều hình thức bảo mật:
* Web: mật khẩu, OTP.
* Mobile: mật khẩu, dấu vân tay, nhận diện khuôn mặt, OTP.

### **2.1.4** **Cultural and political requirements**

N/A

### **2.1.5** **Performance Requirements**

* Hệ thống có giao diện đơn giản, trực quan, dễ dùng nhất có thể, hài hòa, dễ đọc, không màu mè.
* Màu sắc chủ đạo: màu xanh nước biển.

### **2.1.6** **…**

N/A

# **3** **Functional Requirements**

* Khi khách hàng đến thuê phòng. Người cho thuê phòng kiểm tra còn phòng hay không. Nếu không còn phòng thì thông báo cho khách hàng, ngược lại thì cho khách xem qua phòng trọ. Nếu khách hàng đồng ý thuê phòng thì yêu cầu khách hàng xuất trình giấy CMND, thông tin cá nhân để lập phiếu đăng ký thuê phòng cho khách hàng và điền vào bản khai nhân khẩu và sổ đăng ký tạm trú tạm vắng cho khách thuê phòng. Đồng thời cho khách hàng xem bảng giá thuê phòng và các dịch vụ khác trong thời gian thuê phòng. Và liệt kê danh sách các thiết bị gồm có mã thiết bị, tên thiết bị để tiện cho việc quản lý. Khi khách hàng trả phòng. Người cho thuê phòng làm thủ tục thanh toán chi tiết với khách hàng về tiền phòng, tiền sử dụng các loại dịch vụ mà khách hàng đã sử dụng trong thời gian thuê phòng, tiền phát sinh (nếu có), tiền thuế …
* Khi khách hàng có những thắc mắc, ý kiến trong quá trình thuê trọ thì khách hàng sẽ phải đăng nhập vào hệ thống và gửi các thắc mắc. Hệ thống lưu lại các thắc mắc và chuyển đến chủ trọ để giải đáp.
* Tổng hợp, báo cáo các hóa đơn, phiếu đăng ký thuê phòng của khách hàng theo định kỳ.

# **4** **Functional model**

## **4.1** **OVERALL DESCRIPTION**

* Hệ thống quản lý phòng trọ gồm các chức năng: quản lý thông tin khách hàng, quản lý thông tin phòng trọ, quản lý sử dụng dịch vụ và quản lý thắc mắc.

Graphical user interface

Description automatically generated

## **4.2** **Use case description**

### **4.2.1** **Thuê phòng**

#### **4.2.1.1** **Overview**

* Khách thuê đến thuê phòng, chủ trọ kiểm tra số lượng phòng trống, lấy thông tin khách hàng, đồng thời lập hợp đồng cho thuê nếu khách đồng ý thuê phòng.

#### **4.2.1.2** **Actors**

* Chủ trọ

#### **4.2.1.3** **Triggers**

* Khách hàng đến gặp chủ trọ để thuê phòng

#### **4.2.1.4** **Relationships**

N/A

#### **4.2.1.5** **Main flow – Operation**

Bước 1: Khách hàng đến gặp chủ trọ để thuê phòng, chủ trọ kiểm tra còn phòng trống hay không.

Bước 2: Chủ trọ dẫn khách đi xem phòng nếu còn phòng trống.

Bước 3: Chủ trọ cho khách hàng xem bảng giá các dịch vụ kèm theo.

Bước 4: Chủ trọ nhập thông tin khách hàng vào hệ thống nếu khách hàng đồng ý thuê phòng.

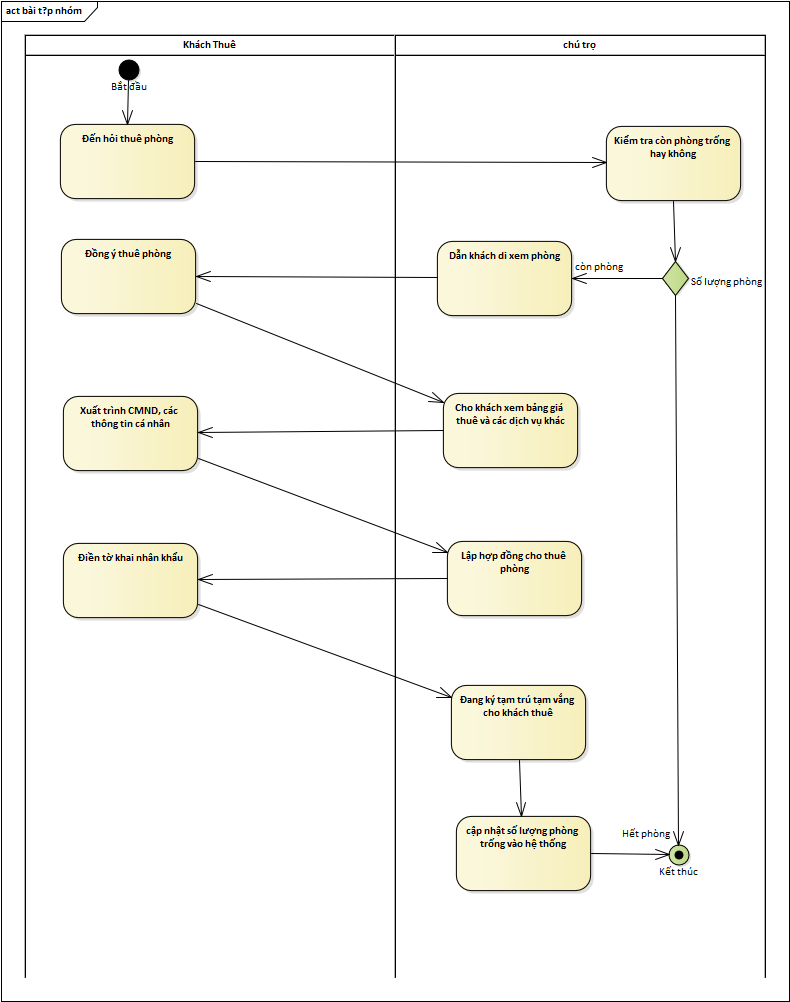
Bước 5: Chủ trọ lập hợp đồng cho thuê, đăng ký tạm trú cho khách thuê và cập nhật lại số lượng phòng vào hệ thống.

#### **4.2.1.6** **Alternative flows**

***AF1***: Số lượng phòng trống đã hết.

Bước 1: Chủ trọ báo đã hết phòng và kết thúc việc cho thuê.

#### **4.2.1.7** **Activity diagram**



#### **4.2.1.8** **Open issues**

N/A

### **4.2.2** **Trả phòng**

#### **4.2.2.1** **Overview**

#### Khách hàng trả phòng và chủ trọ làm thủ tục thanh toán.

#### **4.2.2.2** **Actors**

* Chủ trọ

#### **4.2.2.3** **Triggers**

* Khi khách hàng trả phòng.

#### **4.2.2.4** **Relationships**

#### **4.2.2.5** **Main flow – Operation**

Bước 1: Chủ trọ đăng nhập vào hệ thống.

Bước 2: Nhập thông tin khách thuê trọ.

Bước 3: Kiểm tra những dịch vụ mà khách thuê đã sử dụng trong thời gian thuê.

Bước 4: Hệ thống tính tổng tiền của tất cả dịch vụ mà khách thuê sử dụng.

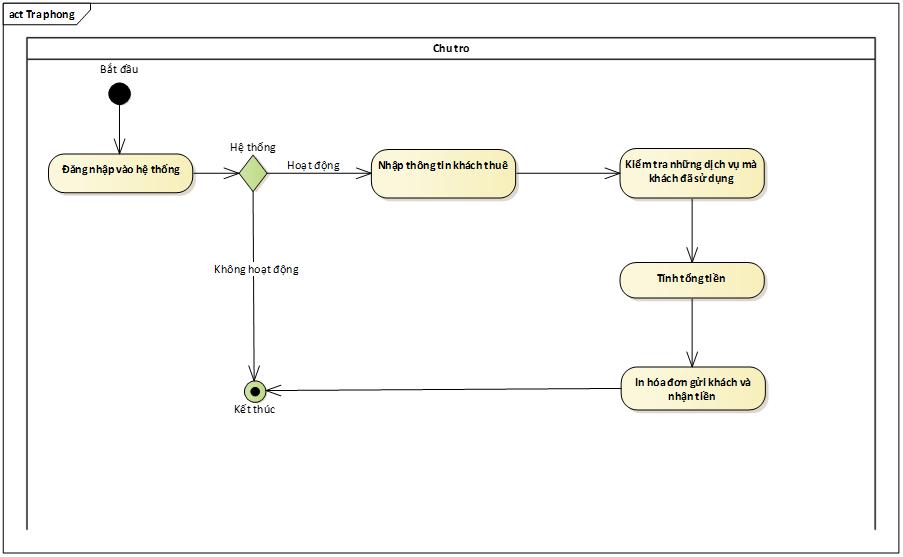
Bước 5: In hóa đơn gửi khách và nhận tiền tiền từ khách.

#### **4.2.2.6** **Alternative flows**

***AF1****: Chủ trọ không đăng nhập vào hệ thống.*

Bước 1: Thông báo với khách hàng và kết thúc.

#### **4.2.2.7** **Activity diagram**



#### **4.2.2.8** **Open issues**

N/A

### **4.2.3 Thống kê**

#### **4.2.3.1** **Overview**

* Tổng hợp, báo cáo các hóa đơn của khách hàng theo khoảng thời gian chủ trọ quy định.

#### **4.2.3.2** **Actors**

* Chủ trọ

#### **4.2.3.3** **Triggers**

* Khi chủ trọ muốn xem báo cáo, thống kê.

#### **4.2.3.4** **Relationships**

N/A

#### **4.2.3.5** **Main flow – Operation**

Bước 1: Chủ trọ đăng nhập vào hệ thống.

Bước 2: Chủ trọ yêu cầu xuất thống kê.

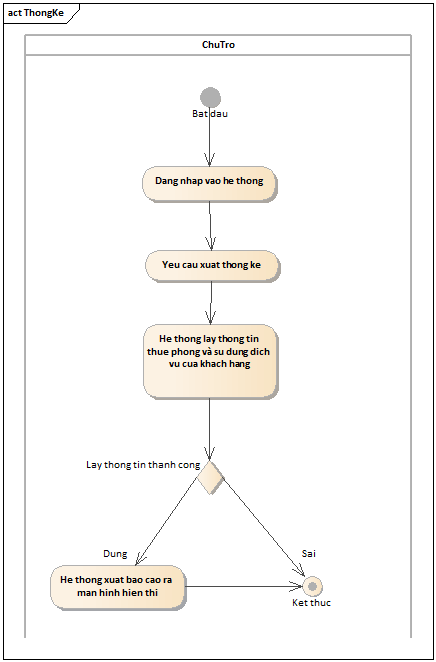
Bước 3: Hệ thống lấy thông tin thuê phòng và sử dụng dịch vụ của khách hàng để tổng hợp báo cáo.  
- Nếu hệ thống không lấy được thông tin thì huỷ toàn bộ quy trình thống kê.

Bước 4: Hệ thống xuất báo cáo ra màn hình hiển thị.

#### **4.2.3.6** **Alternative flows**

N/A

#### **4.2.3.7** **Activity diagram**



#### **4.2.3.8** **Open issues**

N/A

## **4.3** **Open Issues**

N/A

# **5** **Structural model**

## **5.1** **Class diagram**



**\*Mô tả phương thức**

Class **PhongTro**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method name: Kiemtrasoluongphong()** | | **Class name: PhongTro** | **Programmer: HT.Tuan** |
| **Events: Button X click** | | | |
| **Inputs** | **Name** | | **Data type** |
| DienTich | | Int |
| GiaPhong | | Int |
| HienTrangPhong | | Char |
| MaPhong | | Char |
| **Outputs** | KetQuaKiemTraSoluong | | Char |
|  | |  |
| **Algorithm specification** | 1. Chủ trọ đăng nhập vào hệ thống và kiểm tra số lượng phòng còn trống.  2. Nếu hết phòng thì báo về hết phòng trống  Nếu còn phòng thì báo là còn phòng   * DienTich: là diện tích của phòng * GiaPhong: là giá thuê phòng trong 1 tháng * HienTrangPhong: là tình trạng còn trống hay đã được thuê * MaPhong: Gồm 3 kí tự không chứa các kí tự đặc biệt   3. Trả về SoLuongPhong từ lớp PhongTro | | |
| **Notes** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Method name: LayGiaPhong** | | **Class name: PhongTro** | **Programmer: SyNT** |
| **Events: Button X click** | | | |
| **Inputs** | **Name** | | **Data type** |
|  | |  |
| **Outputs** | GiaPhong | | Int |
| **Algorithm specification** | 1. Hệ thống truy cập vào lớp PhongTro.  2. Trả về thông tin GiaPhong từ lớp PhongTro. | | |
| **Notes** |  | | |

Class **KhachThue**

|  |  |  |  |
| --- | --- | --- | --- |
| **Methodname: Nhapthongtinkhachhang()** | | **Class name: KhachThue** | **Programmer: NguyenNQ** |
| **Events: Button X click** | | | |
| **Inputs** | **Name** | | **Data type** |
| QueQuan | | Char |
| GioiTinh | | Nam/Nu |
| NgaySinh | | Char//String |
| SoCMND | | Int |
| SoDT | | Char |
| HoTen | | Char |
| MaKhachHang | | Char |
| **Outputs** | KetQua | | Short |
|  | |  |
| **Algorithm specification** | 1. Chủ trọ thực hiện nhập thông tin cho khách hàng mới  2. Nếu không hợp lệ thì báo lỗi  Nếu hợp lệ thì báo cập nhật khách hàng mới thành công   * QueQuan: là quê quán của khách hàng * GioiTinh: là giới tính Nam/Nữ của khách hàng * NgaySinh: là ngày sinh của khách hàng, khách hàng phải đủ 18 tuổi * SoCMND: Gồm 10 kí tự số * SoDT: Gồm 10-11 số * HoTen: họ tên của khách hàng, không quá 50 ký tự * MaKhachHang: Gồm 5 ký tự ko chứa các ký tự đặc biệt | | |
| **Notes** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Method name: KiemTraTTKhachThue** | | **Classname: KhachThue** | **Programmer: SyNT** |
| **Events: Button X click** | | | |
| **Inputs** | **Name** | | **Data type** |
| MaKhachThue | | Int |
| **Outputs** | KetQua | | Short |
| HoTen | | nvarchar |
| GioiTinh | | nvarchar |
| NgaySinh | | Date |
| DiaChi | | nvarchar |
| SoDT | | Varchar |
| SoCMND | | Varchar |
| **Algorithm specification** | 1. Ghi nhận mã khách thuê.  2. Hệ thống trả về thông tin khách thuê trên màn hình.  Bao gồm: Họ tên, giới tính, ngày sinh, địa chỉ, số điện thoại, số CMND/CCCD.  3. Hệ thống đối chiếu thông tin trên hệ thống và hợp đồng thuê.   * Nếu kết quả đối chiếu không trùng khớp thì trả về 0. * Ngược lại, trả về 1. | | |
| **Notes** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Methodname: LayThongTinKhachHang()** | | **Class name: KhachThue** | **Programmer: NguyenNQ** |
| **Events: Button X click** | | | |
| **Inputs** | **Name** | | **Data type** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **Outputs** | KetQuaThongTinDichVu | | Char |
|  | |  |
| **Algorithm specification** | 1. Hệ thống lấy và xuất thông tin liên quan đến người sử dụng dịch vụ.   * QueQuan: là quê quán của khách hàng * GioiTinh: là giới tính Nam/Nữ của khách hàng * NgaySinh: là ngày sinh của khách hàng, khách hàng phải đủ 18 tuổi * SoCMND: Gồm 10 kí tự số * SoDT: Gồm 10-11 số * HoTen: họ tên của khách hàng, không quá 50 ký tự * MaKhachThue: Gồm 5 ký tự (2 chữ + 3 số, không bao gồm các ký tự đặc biệt). | | |
| **Notes** |  | | |

**Class HopDongThue**

|  |  |  |  |
| --- | --- | --- | --- |
| **Methodname: Laphopdongchothue()** | | **Class name: HopDongThue** | **Programmer: NguyenNQ** |
| **Events: Button X click** | | | |
| **Inputs** | **Name** | | **Data type** |
| BenChoThue | | Char |
| BenThue | | Char |
| GiaPhong | | Char |
| MaHopDong | | Char |
| NgayThue | | Char//String |
| NgayTra | | Char//String |
| DiaChiPhongTro | | Char |
| DieuKhoan | | Char |
| **Outputs** | KetQua | | Char |
|  | |  |
| **Algorithm specification** | 1. Chủ trọ lập hợp đồng thuê  2. Nếu không hợp lệ thì hiển thị thông báo không hợp lệ  Nếu hợp lệ thì báo đã lập hợp đồng thành công   * BenChoThue gồm: là bên cho thuê trọ * BenThue: là khách thuê trọ * GiaPhong: là giá thuê phòng trong 1 tháng * MaHopDong: gồm 10 ký tự không bao gồm các ký tự đặc biệt * NgayThue: là ngày bắt đầu thuê phòng * NgayTra: là ngày hết hợp đồng thuê phòng, ngày trả phải sau ngày thuê * DiaChiPhongTro: địa chỉ, số phòng của phòng trọ * DieuKhoan: các điều khoản phải tuân thủ khi ký hợp đồng   3. In hợp đồng cho thuê. | | |
| **Notes** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Method name: NhapTTKhachThue** | | **Class name: HopDongThue** | **Programmer: SyNT** |
| **Events: Button X click** | | | |
| **Inputs** | **Name** | | **Data type** |
| MaKhachThue | | Int |
| **Outputs** | KetQua | | Short |
|  | |  |
| **Algorithm specification** | 1. Nhập mã khách thuê vào hệ thống.  2. Nếu mã khách thuê không hợp lệ thì trả về giá trị 0  Hợp lệ:   * Mã khách thuê gồm 5 kí tự (2 chữ + 3 số, không bao gồm các kí tự đặc biệt) | | |
| **Notes** |  | | |

Class **DichVu**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method name: LayTenDV** | | **Class name: DichVu** | **Programmer: SyNT** |
| **Events: Button X click** | | | |
| **Inputs** | **Name** | | **Data type** |
|  | |  |
| **Outputs** | TenDV | | nvarchar |
| **Algorithm specification** | 1. Hệ thống truy cập vào lớp DichVu.  2. Trả về thông tin TenDV từ lớp DichVu. | | |
| **Notes** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Method name: LayGiaDV** | | **Class name: DichVu** | **Programmer: SyNT** |
| **Events: Button X click** | | | |
| **Inputs** | **Name** | | **Data type** |
|  | |  |
| **Outputs** | MucGia | | int |
| **Algorithm specification** | 1. Hệ thống truy cập vào lớp DichVu.  2. Trả về thông tin MucGia từ lớp DichVu. | | |
| **Notes** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Method name: XuatThongKe()** | | **Class name: DichVu** | **Programmer: NguyenNQ** |
| **Events: Button X click** | | | |
| **Inputs** | **Name** | | **Data type** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **Outputs** |  | |  |
|  | |  |
| **Algorithm specification** | 1. Hệ thống kích hoạt quy trình tạo ra báo cáo:  Thực hiện hàm LayThongTinDichVu() | | |
| **Notes** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **MethodName:LayThongTinDichVu()** | | **Class name: DichVu** | **Programmer: NguyenNQ** |
| **Events: sau hàm XuatThongBao()** | | | |
| **Inputs** | **Name** | | **Data type** |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
| **Outputs** | KetQuaThongTinDichVu | | Char |
|  |  | |  |
| **Algorithm specification** | 1. Hệ thống lọc những dịch vụ đã sử dụng trong tháng.  2. Trả về ThongTinDichVu từ lớp DichVu của khách hàng tương ứng.  Gồm:   * Khách hàng * Tên dịch vụ * Giá dịch vụ | | |
| **Notes** |  | | |

**Class HoaDon**

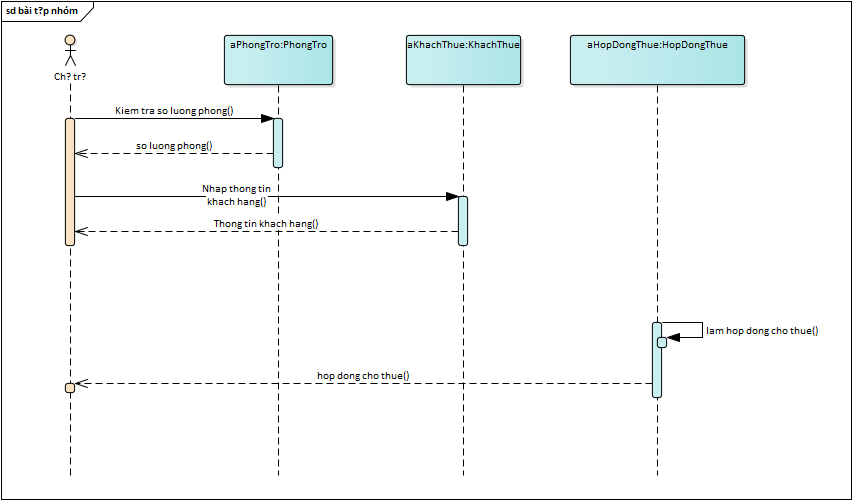
|  |  |  |  |
| --- | --- | --- | --- |
| **Method name: TinhTongTien** | | **Class name: HoaDon** | **Programmer: SyNT** |
| **Events: Button X click** | | | |
| **Inputs** | **Name** | | **Data type** |
|  | |  |
| **Outputs** | TongTien | | nvarchar |
| **Algorithm specification** | 1. Sau khi lấy TenDV và MucGia từ lớp DichVu.  2. Hệ thống tính tổng số tiền khách phải trả.   * TienPhaiTra bằng tổng MucGia của các dịch vụ sử dụng. | | |
| **Notes** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Methodname: XuatBaoCao()** | | **Class name: DichVu** | **Programmer: NguyenNQ** |
| **Events: Button X click** | | | |
| **Inputs** | **Name** | | **Data type** |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
| **Outputs** | BaoCaoSuDungDichVu | | Char |
|  | |  |
| **Algorithm specification** | 1. Sắp xếp và trình bày thông tin báo cáo theo mẫu:   Khách hàng: … Quê quán: … CCCD: …  Điện: … Số tiền: …  Nước: … Số tiền: …  Dịch vụ khác: … Số tiền: …   1. Xuất ra màn hình. | | |
| **Notes** |  | | |

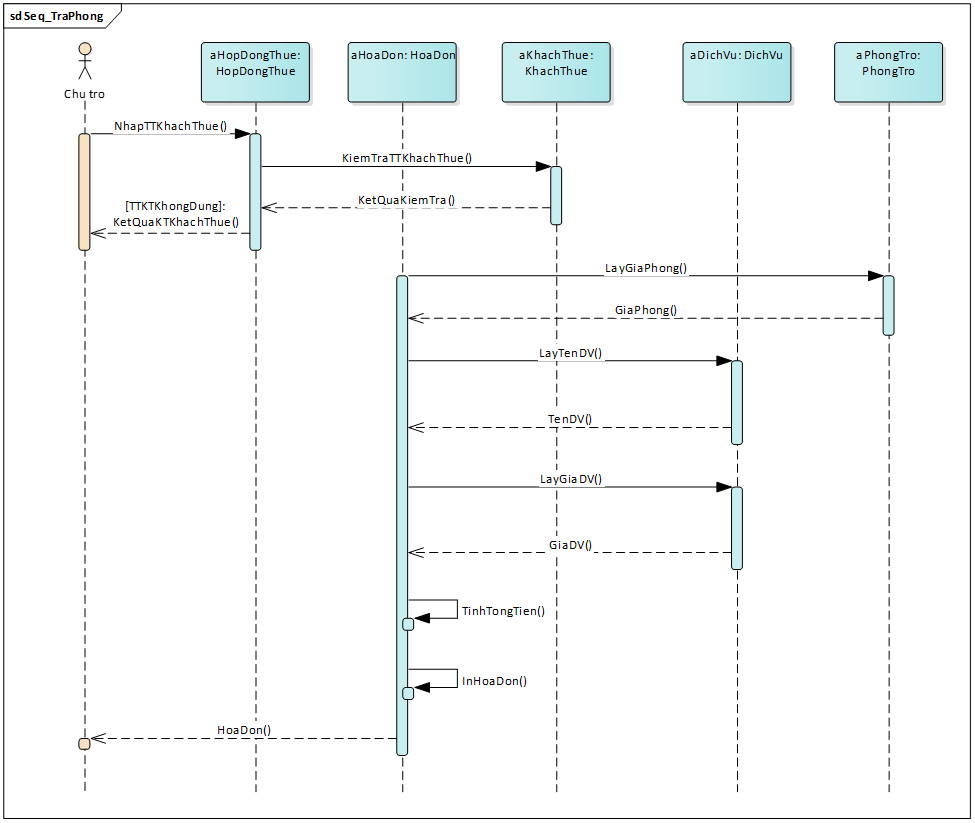
# **6** **Behavioral model**

## **6.1 Sequence diagram**

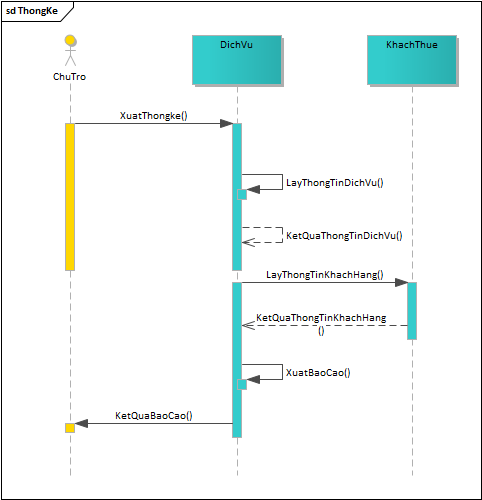
***\* Thuê phòng***



***\* Trả phòng***



***\*Thống kê***

******

## 