TRƯỜNG ĐẠI HỌC BÁCH KHOA

KHOA CÔNG NGHỆ THÔNG TIN

──────── \* ───────

**BÀI TẬP LỚN**

MÔN: CÔNG NGHỆ PHẦN MỀM

**<HỆ THỐNG QUẢN LÝ THÔNG TIN BÁN VÉ MÁY BAY CHO DOANH NGHIỆP >**

Nhóm : **11**

Mã lớp học : **1020252.2420.23.11**

Giáo viên hướng dẫn : TS. Võ Đức Hoàng

Danh sách sinh viên thực hiện:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Họ tên** | **Mã sinh viên** | **Email** | **Lớp** |
| 01 | Nguyễn Thanh Hiếu | 102230239 | nguyenthanhhieu17022005@gmail.com | 23T\_DT2 |
| 02 | Nguyễn Mạnh Kiên | 102230248 | kien56162@gmail.com | 23T\_DT2 |
| 03 | Nguyễn Văn Tiến | 102230271 | Nguyenvantien080174@gmail.com | 23T\_DT2 |
|  |  |  |  |  |
|  |  |  |  |  |

***Đà Nẵng, tháng 5 năm 2025***

MỤC LỤC

[MỤC LỤC i](#_Toc54813310)

[LỜI NÓI ĐẦU 1](#_Toc54813311)

[PHÂN CÔNG THÀNH VIÊN TRONG NHÓM 2](#_Toc54813312)

[CHƯƠNG 1. KHẢO SÁT BÀI TOÁN 3](#_Toc54813313)

[1.1. Mô tả yêu cầu bài toán 3](#_Toc54813314)

[1.2. Khảo sát bài toán 3](#_Toc54813315)

[1.3. Xác định thông tin cơ bản cho nghiệp vụ của bài toán 4](#_Toc54813316)

[1.4. Xây dựng biểu đồ mô tả nghiệp vụ và phân cấp chức năng 5](#_Toc54813317)

[1.5. Xây dựng kế hoạch dự án đơn giản 6](#_Toc54813318)

[CHƯƠNG 2. ĐẶC TẢ YÊU CẦU BÀI TOÁN 8](#_Toc54813319)

[2.1. Giới thiệu chung 8](#_Toc54813320)

[2.2. Biểu đồ use case 9](#_Toc54813321)

[2.2.1. Biểu đồ use case tổng quan 9](#_Toc54813322)

[2.2.2. Biểu đồ use case phân rã mức 2 10](#_Toc54813323)

[2.3. Đặc tả use case 11](#_Toc54813324)

[2.4. Các yêu cầu phi chức năng 13](#_Toc54813325)

[CHƯƠNG 3. PHÂN TÍCH YÊU CẦU 15](#_Toc54813326)

[3.1. Xác định các lớp phân tích 15](#_Toc54813327)

[3.2. Xây dựng biểu đồ trình tự 15](#_Toc54813328)

[3.3. Xây dựng biểu đồ lớp phân tích 16](#_Toc54813329)

[3.4. Xây dựng biểu đồ thực thể liên kết (ERD) 16](#_Toc54813330)

[CHƯƠNG 4. THIẾT KẾ CHƯƠNG TRÌNH 17](#_Toc54813331)

[4.1. Thiết kế kiến trúc 17](#_Toc54813332)

[4.2. Thiết kế cơ sở dữ liệu 17](#_Toc54813333)

[4.3. Thiết kế chi tiết các gói 19](#_Toc54813334)

[4.4. Thiết kế chi tiết lớp 20](#_Toc54813335)

[4.5. Sơ đồ lớp chi tiết 21](#_Toc54813336)

[4.6. Thiết kế giao diện 21](#_Toc54813337)

[CHƯƠNG 5. XÂY DỰNG CHƯƠNG TRÌNH MINH HỌA 24](#_Toc54813338)

[5.1. Thư viện và công cụ sử dụng 24](#_Toc54813339)

[5.2. Kết quả chương trình minh họa 24](#_Toc54813340)

[5.3. Giao diện minh hoạ các chức năng của chương trình 24](#_Toc54813341)

[CHƯƠNG 6. KIỂM THỬ CHƯƠNG TRÌNH 28](#_Toc54813342)

[6.1. Kiểm thử các chức năng đã thực hiện 28](#_Toc54813343)

[6.1.1. Kiểm thử cho chức năng 1 28](#_Toc54813344)

[6.1.2. Kiểm thử cho chức năng 2 29](#_Toc54813345)

[6.1.3. Kiểm thử yêu cầu phi chức năng 29](#_Toc54813346)

[CHƯƠNG 7. HƯỚNG DẪN CÀI ĐẶT VÀ SỬ DỤNG 30](#_Toc54813347)

[7.1. Hướng dẫn cài đặt 30](#_Toc54813348)

[7.2. Đối tượng, phạm vi sử dụng 30](#_Toc54813349)

[7.3. Xác định các yêu cầu cài đặt 30](#_Toc54813350)

[7.4. Hướng dẫn chi tiết các bước cài đặt 30](#_Toc54813351)

[7.5. Hướng dẫn sử dụng phần mềm 30](#_Toc54813352)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 31](#_Toc54813353)

[TÀI LIỆU THAM KHẢO 32](#_Toc54813354)

[PHỤ LỤC 33](#_Toc54813355)

LỜI NÓI ĐẦU

***<Các chú thích nằm trong cặp dấu ngoặc nhọn hoặc có màu ký tự mục đích là để giải thích thêm hoặc lấy ví dụ minh hoạ. Khi sinh viên sử dụng tài liệu này, cần xoá các phần chú thích hoặc ví dụ minh hoạ trong bài làm của mình và điền nội dung theo chỉ dẫn>***

*<Nêu lý do chọn đề tài này, tầm quan trọng của đề tài này>*

*<Mô tả sơ qua về đề tài, quá trình tiếp cận và hoàn thiện đề tài>*

*Trong thời đại công nghệ số phát triển mạnh mẽ, ngành hàng không đã và đang đóng vai trò quan trọng trong việc kết nối các quốc gia, thúc đẩy giao thương và phát triển kinh tế. Đi cùng với sự phát triển này, nhu cầu đi lại bằng đường hàng không của hành khách ngày càng tăng cao, kéo theo yêu cầu về một hệ thống quản lý bán vé máy bay hiện đại, tiện lợi, nhanh chóng và chính xác. Việc đặt vé không chỉ đơn thuần là chọn chuyến bay và mua vé, mà còn liên quan đến quá trình xác thực thông tin, lựa chọn phương thức thanh toán, theo dõi lịch trình, hỗ trợ khách hàng, cùng nhiều chức năng khác nhằm mang lại trải nghiệm tốt nhất cho người sử dụng.*

*Tuy nhiên, trong thực tế, không ít doanh nghiệp vẫn sử dụng các phương thức quản lý thủ công hoặc hệ thống chưa tối ưu, dẫn đến nhiều bất cập. Việc lưu trữ và xử lý thông tin bằng cách truyền thống có thể gây sai sót trong khâu đặt vé, khó khăn trong việc kiểm soát số lượng chuyến bay, chỗ ngồi, cũng như hạn chế khả năng mở rộng quy mô. Hơn nữa, trong bối cảnh cạnh tranh ngày càng gay gắt, các hãng hàng không và đại lý vé máy bay buộc phải đổi mới, áp dụng công nghệ để tối ưu hóa hoạt động kinh doanh, nâng cao hiệu suất làm việc và đáp ứng tốt hơn nhu cầu của khách hàng. Chính vì vậy, việc xây dựng một* ***Hệ thống Quản lý Thông tin Bán vé Máy bay*** *là một yêu cầu cấp thiết, giúp doanh nghiệp vận hành hiệu quả hơn, giảm thiểu rủi ro và nâng cao chất lượng dịch vụ.*

*Nhận thấy tầm quan trọng của vấn đề này, nhóm nghiên cứu đã quyết định lựa chọn đề tài “Hệ thống Quản lý Thông tin Bán vé Máy bay cho Doanh nghiệp” nhằm xây dựng một giải pháp phần mềm có tính ứng dụng cao, đáp ứng được các yêu cầu thực tiễn. Đề tài tập trung vào việc thiết kế và triển khai một hệ thống có thể quản lý toàn diện quy trình đặt vé, bao gồm từ việc cập nhật thông tin chuyến bay, quản lý khách hàng, xử lý thanh toán đến báo cáo doanh thu và hỗ trợ kỹ thuật. Hệ thống này không chỉ giúp doanh nghiệp dễ dàng theo dõi hoạt động kinh doanh, mà còn mang đến cho khách hàng trải nghiệm thuận tiện, an toàn và hiệu quả hơn khi đặt vé máy bay.*

*Quá trình tiếp cận và thực hiện đề tài được tiến hành theo các bước khoa học và có hệ thống. Trước tiên, nhóm nghiên cứu tiến hành khảo sát thực tế, thu thập thông tin từ các doanh nghiệp kinh doanh vé máy bay, nghiên cứu các hệ thống hiện có để xác định những hạn chế và yêu cầu cải tiến. Từ đó, nhóm xây dựng mô hình hệ thống, thiết kế các chức năng cần thiết, bao gồm cả giao diện người dùng và kiến trúc cơ sở dữ liệu. Trong quá trình triển khai, nhóm sử dụng các công nghệ lập trình phù hợp để đảm bảo hệ thống vận hành mượt mà, bảo mật tốt và dễ dàng mở rộng trong tương lai. Cuối cùng, hệ thống được kiểm thử nghiêm ngặt để phát hiện và khắc phục lỗi, đảm bảo mang đến một sản phẩm hoàn thiện trước khi đưa vào sử dụng.*

*Với những nỗ lực và tâm huyết trong quá trình nghiên cứu, nhóm hy vọng rằng đề tài này sẽ mang lại giá trị thực tiễn cho các doanh nghiệp hoạt động trong lĩnh vực bán vé máy bay, giúp họ cải thiện quy trình làm việc, nâng cao hiệu suất kinh doanh và tạo ra một nền tảng dịch vụ chuyên nghiệp hơn. Đồng thời, đề tài cũng là cơ hội để nhóm nghiên cứu áp dụng những kiến thức đã học vào thực tế, rèn luyện kỹ năng phân tích, thiết kế và lập trình, từ đó có thêm kinh nghiệm trong việc xây dựng các hệ thống phần mềm thực tiễn.*

PHÂN CÔNG THÀNH VIÊN TRONG NHÓM

*<Danh sách các thành viên trong nhóm>*

*<Phân công nhiệm vụ cụ thể của mỗi thành viên trong việc thực hiện bài tập lớn môn học và đánh giá mức độ hoàn thành>*

|  |  |  |  |
| --- | --- | --- | --- |
| **Họ và tên** | **Email** | **Tổng hợp công việc thực hiện** | **Đánh giá** |
| Nguyễn Thanh Hiếu | nguyenthanhhieu17022005@gmail.com | Mục 1.1, 1.2 | Hoàn thành |
| Nguyễn Mạnh Kiên | kien56162@gmail.com | Mục 1.3 | Hoàn Thành |
| Nguyễn Văn Tiến | Nguyenvantien080174@gmail.com | Mục 1.4 | Hoàn Thành |

# KHẢO SÁT BÀI TOÁN

<Trình bày, mô tả chi tiết về các kết quả khảo sát về bài toán – thông qua đề bài và những yêu cầu từ giáo viên sau những lần trao đổi, gặp mặt>

## Mô tả yêu cầu bài toán

<Mô tả bằng lời về bài toán được giao>

Hiện nay, nhu cầu đi lại bằng đường hàng không ngày càng tăng, kéo theo sự phức tạp trong việc quản lý đặt vé, chuyến bay và thông tin hành khách. Doanh nghiệp cần một hệ thống quản lý bán vé máy bay **tự động, chính xác và hiệu quả**, giúp giảm sai sót, tối ưu doanh thu và nâng cao trải nghiệm khách hàng.

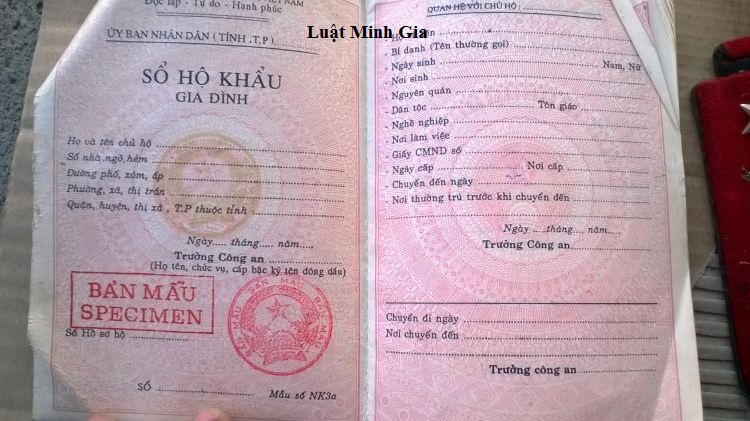
CÁC YÊU CẦU CHÍNH

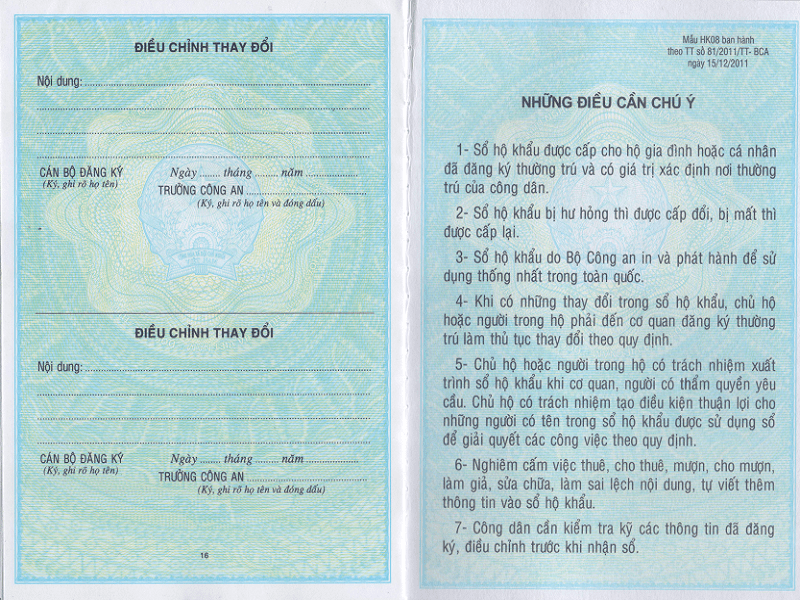
* **Quản lý chuyến bay**: Thông tin chuyến bay, số ghế trống, cập nhật lịch trình.
* **Quản lý hành khách**: Lưu thông tin khách hàng, lịch sử đặt vé.
* **Đặt vé & Vé máy bay**: Đặt, hủy vé, cấp mã đặt chỗ (PNR).
* **Thanh toán**: Xác nhận giao dịch, lưu lịch sử thanh toán.
* **Quản lý sân bay**: Danh sách sân bay, đường bay.
* **Báo cáo & Thống kê**: Doanh thu, số vé bán, tình trạng đặt chỗ.

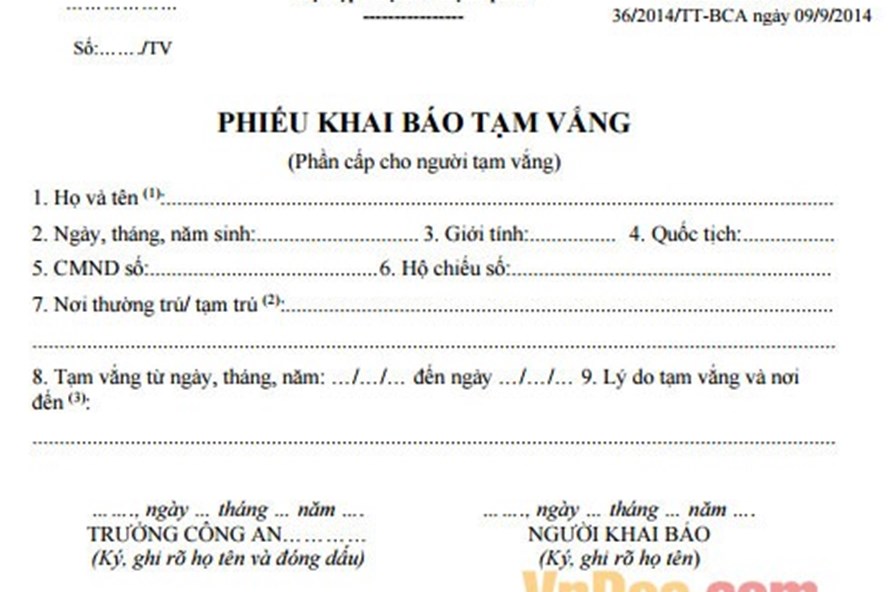
## Khảo sát bài toán

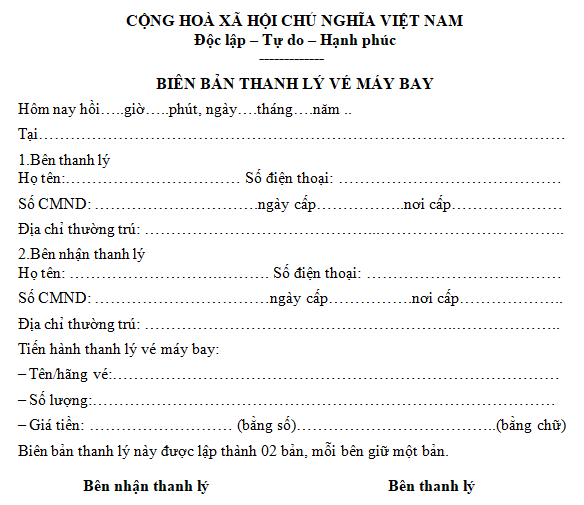
< Các nhóm tìm hiểu về yêu cầu nghiệp vụ qua việc khảo sát các biểu mẫu / tài liệu có liên quan. Trong tình huống không tìm được các biểu mẫu có sẵn, các thành viên trong nhóm có thể thảo luận để tự xây dựng biểu mẫu này theo mô tả của bài toán.>

Ví dụ: với nhóm yêu cầu nghiệp vụ số 1 có thể tìm các biểu mẫu về sổ hộ khẩu, giấy tạm vắng, tạm trú:

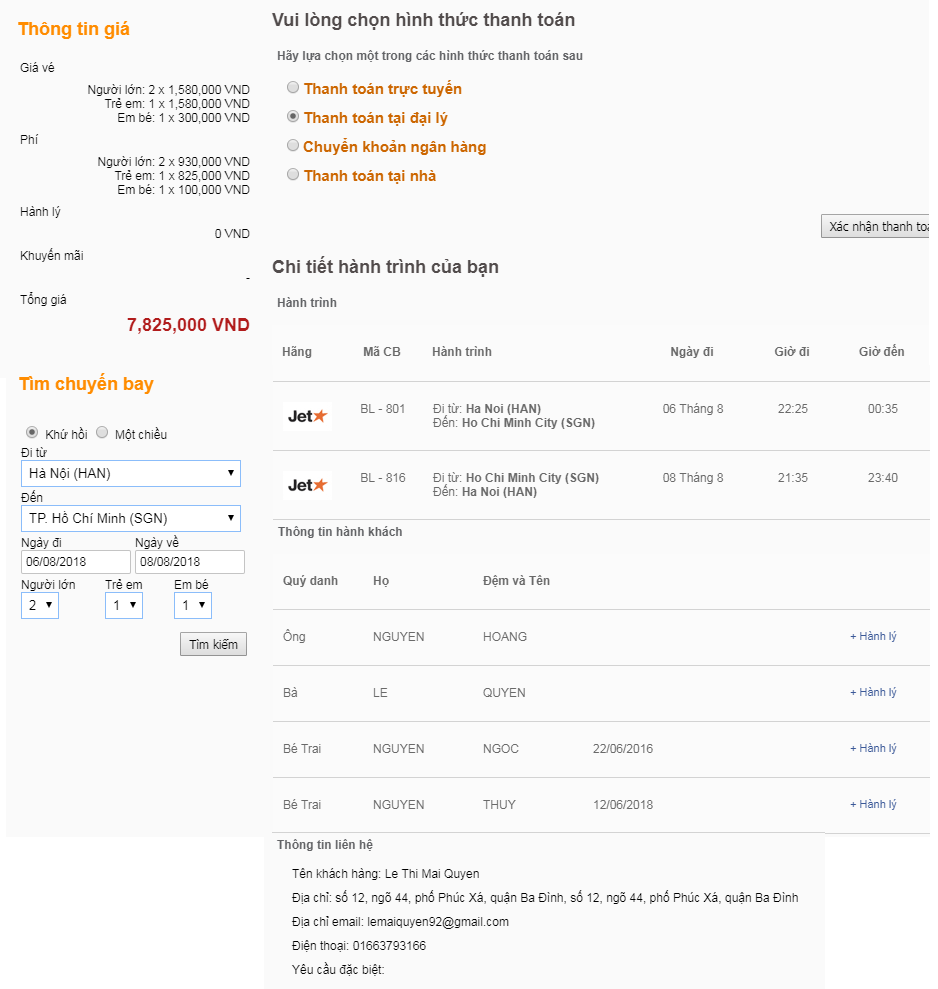








Hình 1: Biên bản Thanh Lý Vé Máy Bay



Hình 2: Biểu Mẫu Đặt Vé Máy Bay Và Thanh Toán



Hình 3. Hóa Đơn Điện Tử

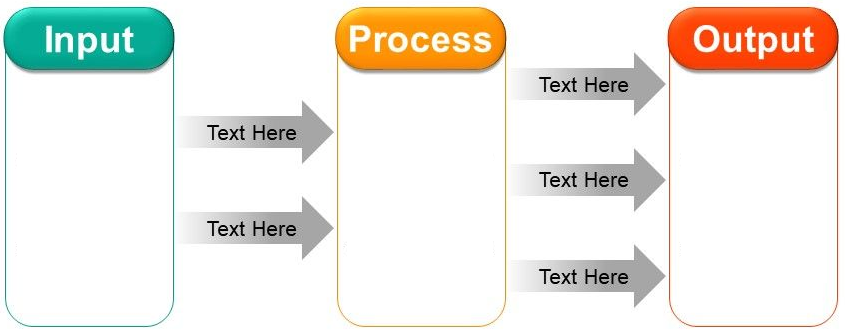
## Xác định thông tin cơ bản cho nghiệp vụ của bài toán

<Các nhóm thảo luận để xác định đầu vào (input) và kết quả (output) cho nhóm yêu cầu nghiệp vụ của bài toán (case study) mà nhóm đã lựa chọn.

Phân rã mô tả của nhóm yêu cầu nghiệp vụ thành các nghiệp vụ nhỏ hơn.

Với mỗi nghiệp vụ nhỏ này thực hiện phân tích các thông tin cụ thể>

Có thể vẽ sơ đồ thể hiện kết quả phân tích:



Hoặc thể hiện dưới dạng bảng:

|  |  |  |
| --- | --- | --- |
| **Input** | **Process** | **Output** |
| **Mã chuyến bay, hãng hàng không, sân bay đi/đến, thời gian, số ghế** | Lưu trữ, cập nhật thông tin chuyến bay, kiểm tra số ghế trống | Danh sách chuyến bay, trạng thái chuyến bay |
| **Mã chuyến bay, thông tin hành khách, hạng vé** | Kiểm tra vé trống, tạo mã đặt chỗ, lưu thông tin đặt vé | Vé máy bay, mã đặt chỗ (PNR) |
| **Mã đặt chỗ, phương thức thanh toán, số tiền** | Xác nhận giao dịch, cập nhật trạng thái vé | Vé được xác nhận hoặc từ chối |
| **Họ tên, CMND/Hộ chiếu, ngày sinh, quốc tịch, liên hệ** | Lưu trữ, tìm kiếm, cập nhật thông tin khách hàng | Danh sách khách hàng, thông tin hành khách |
| **Mã đặt chỗ, lý do hủy, chính sách hoàn tiền** | Kiểm tra điều kiện hủy, cập nhật trạng thái vé, xử lý hoàn tiền | Vé bị hủy, tiền hoàn lại (nếu có) |
| **Ngày/tháng/năm, doanh thu, số vé bán, số vé hủy** | Tính toán tổng doanh thu, số vé bán, tỷ lệ hủy vé, thống kê theo chuyến bay | Báo cáo doanh thu, hiệu suất bán vé |

## Xây dựng biểu đồ phân rã chức năng

Biểu đồ phân rã chức năng (WBS - Work Breakdown Structure) là một sơ đồ giúp chia nhỏ một hệ thống hoặc một dự án thành các phần tử nhỏ hơn, có tổ chức, giúp dễ dàng quản lý và thực hiện từng bước.

### Mục đích chính:

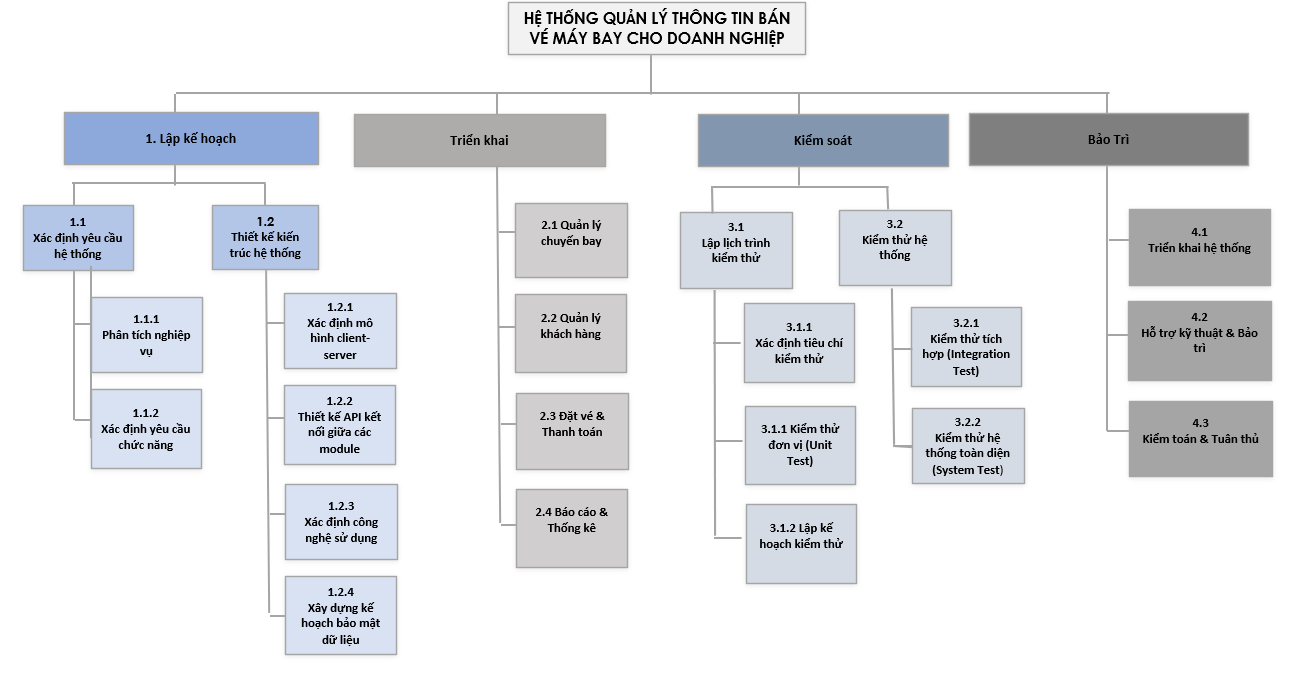
- Phân chia hệ thống/phần mềm thành các chức năng chính và các chức năng con nhỏ hơn.

- Giúp lập kế hoạch, theo dõi và quản lý công việc hiệu quả.

- Hỗ trợ nhóm phát triển dễ dàng xác định nhiệm vụ của từng bộ phận.

Tránh bỏ sót chức năng quan trọng.

### Biểu đồ phân rã chức năng



### Mô tả các chức năng trong biểu đồ:

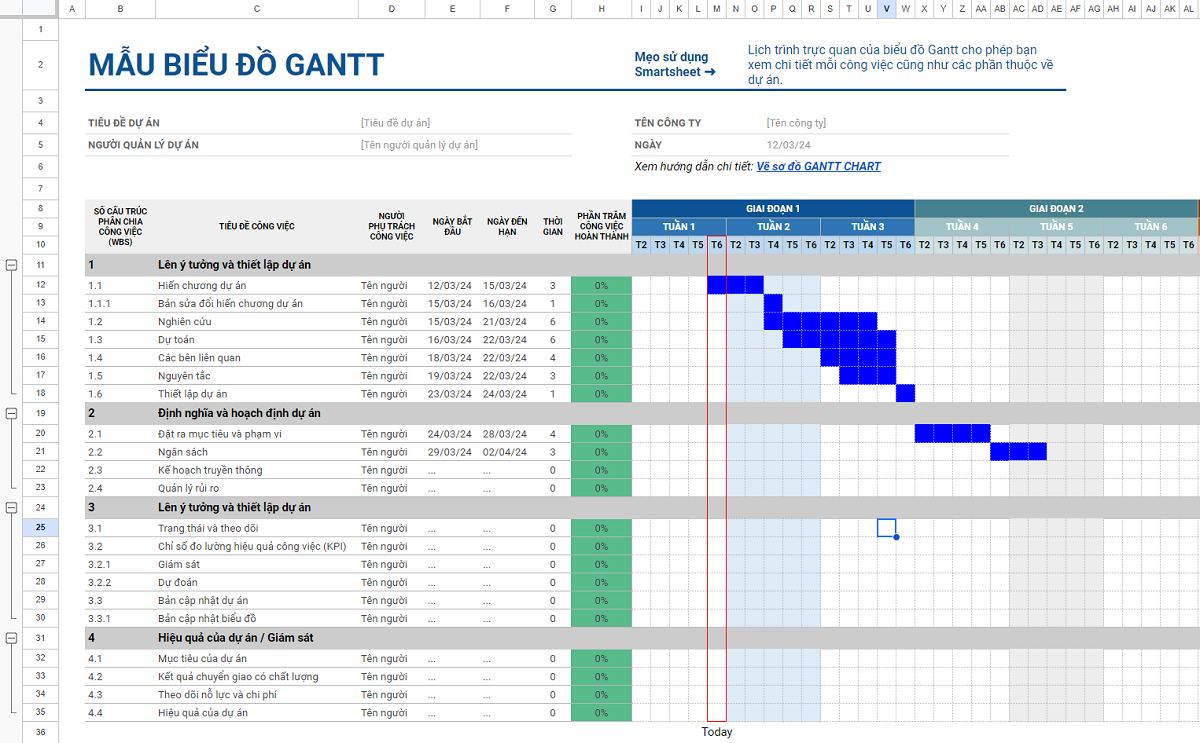
|  |  |  |
| --- | --- | --- |
| **Tên chức năng** | **Mô tả** | **Đánh giá khả năng thực hiện (nhân lực, thời gian, công nghệ, môi trường)** |
| **Quản lý khách hàng** | Quản lý thông tin khách hàng, đăng ký, đăng nhập, cập nhật thông tin | Cần 1-2 thành viên, thời gian 1-2 tuần, sử dụng Java, MySQL, môi trường web. |
| **Quản lý đặt vé** | Cho phép khách hàng tìm kiếm chuyến bay, đặt vé, hủy vé | 2-3 thành viên, thời gian 2-3 tuần, sử dụng Java hoặc C#, MySQL, API từ hãng hàng không |
| **Quản lý chuyến bay** | Cập nhật thông tin chuyến bay, tình trạng ghế ngồi, thay đổi lịch bay | 2 thành viên, 2 tuần, tích hợp API dữ liệu chuyến bay. |
| **Quản lý thanh toán** | Hỗ trợ thanh toán online, kiểm tra trạng thái thanh toán, xuất hóa đơn | 2 thành viên, 2 tuần, tích hợp cổng thanh toán (VNPay, PayPal...), cần bảo mật cao |
| **Quản lý nhân viên & hệ thống** | Quản lý tài khoản nhân viên, phân quyền, bảo mật hệ thống | 1-2 lập trình viên, 2 tuần, sử dụng C# |

## Xây dựng kế hoạch dự án đơn giản

< Xây dựng một bản kế hoạch đơn giản như sau >

|  |  |  |
| --- | --- | --- |
| **Công việc** | **Thời gian (số giờ làm việc)** | **Số người** |
| **1. Công việc 1**  **1.1 Công việc 1.1**  **1.2 Công việc 1.2**  **…** |  |  |
| **2. Công việc 2**  **2.1 Công việc 2.1**  **2.2 Công việc 2.2**  **…** |  |  |
| **…** |  |  |

Biểu đồ Gantt



< Xây dựng một bảng quản lý đơn giản các rủi ro với dự án phát triển phần mềm trong bài tập môn học >

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Công việc / Hoạt động** | **Xác định rủi ro** | | | **Quản lý rủi ro** | |
| **Mối nguy** | **Rủi ro** | **Mức độ** | **Chiến lược** | **Biện pháp** |
| Mua hàng | Hàng bị hư hại khi vận chuyển | Không có nguyên liệu sản xuất | Trung bình | Giảm thiểu | Xây dựng tồn kho tối thiểu |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

(Mức độ có thể đánh giá theo: thấp / trung bình / cao)

(Chiến lược có thể lựa chọn: Giảm thiểu rủi ro / Chuyển rủi ro / Loại bỏ rủi ro)

# ĐẶC TẢ YÊU CẦU BÀI TOÁN

<Đặc tả yêu cầu bài toán bao gồm:

* Biểu đồ use case tổng quan
* Biểu đồ use case phân rã cho các chức năng phức tạp.
* Đặc tả use case cho những use case chính, nghiệp vụ quan trọng của bài toán>

## Giới thiệu chung

<Xác định danh sách các tác nhân / actor và danh sách các usecase>

***+ Xác định các tác nhân của hệ thống:***

+ Trả lời các câu hỏi sau để xác định Actor cho hệ thống:

* Ai sử dụng hệ thống này? Ai cung cấp thông tin cho các hoạt động của hệ thống? Ai nhận kết quả từ các chức năng của hệ thống?*

* Nhóm người nào sẽ duy trì và quản trị hệ thống hoạt động?*

* Hệ thống có tương tác với các thiết bị hay phần mềm ngoại vi nào khác hay không?*

* Hệ thống nào tương tác với hệ thống này?*

* Các thiết bị có tương tác với hệ thống này?*

+ Lập bảng liệt kê các tác nhân và mô tả thông tin cho các tác nhân:

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên tác nhân** | **Mô tả tác nhân** |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |

+ Xác định quan hệ giữa các tác nhân.

***+ Xác định các ca sử dụng***

+ Với ***từng tác nhân*** ***đã xác định ở trên***, đặt và trả lời các câu hỏi:

* Các tác nhân yêu cầu những gì từ hệ thống*

* Các công việc chính mà tác nhân đó muốn hệ thống thực thi?*

* Tác nhân đó có tạo ra hay thay đổi dữ liệu gì của hệ thống?*

* Tác nhân đó có phải thông báo gì cho hệ thống?*

* Tác nhân đó có cần thông tin thông báo gì từ hệ thống?*

+ Chúng ta sẽ xác định được các Use Case cần thiết cho hệ thống. Nếu có nhiều usecase thì nên đặt mã để tham chiếu dễ dàng trong các phần sau của tài liệu.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Mã usecase** | **Tên usecase** | **Mô tả Usecase** | **Tác nhân tương tác** | **Độ phức tạp** |
| 1 |  |  |  |  |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
| 5 |  |  |  |  |  |

***+ Xác định các quan hệ***

Phân tích và các định các quan loại hệ giữa các Actor và Use Case, giữa các Actor với nhau, giữa các Use Case với nhau.

## Biểu đồ use case

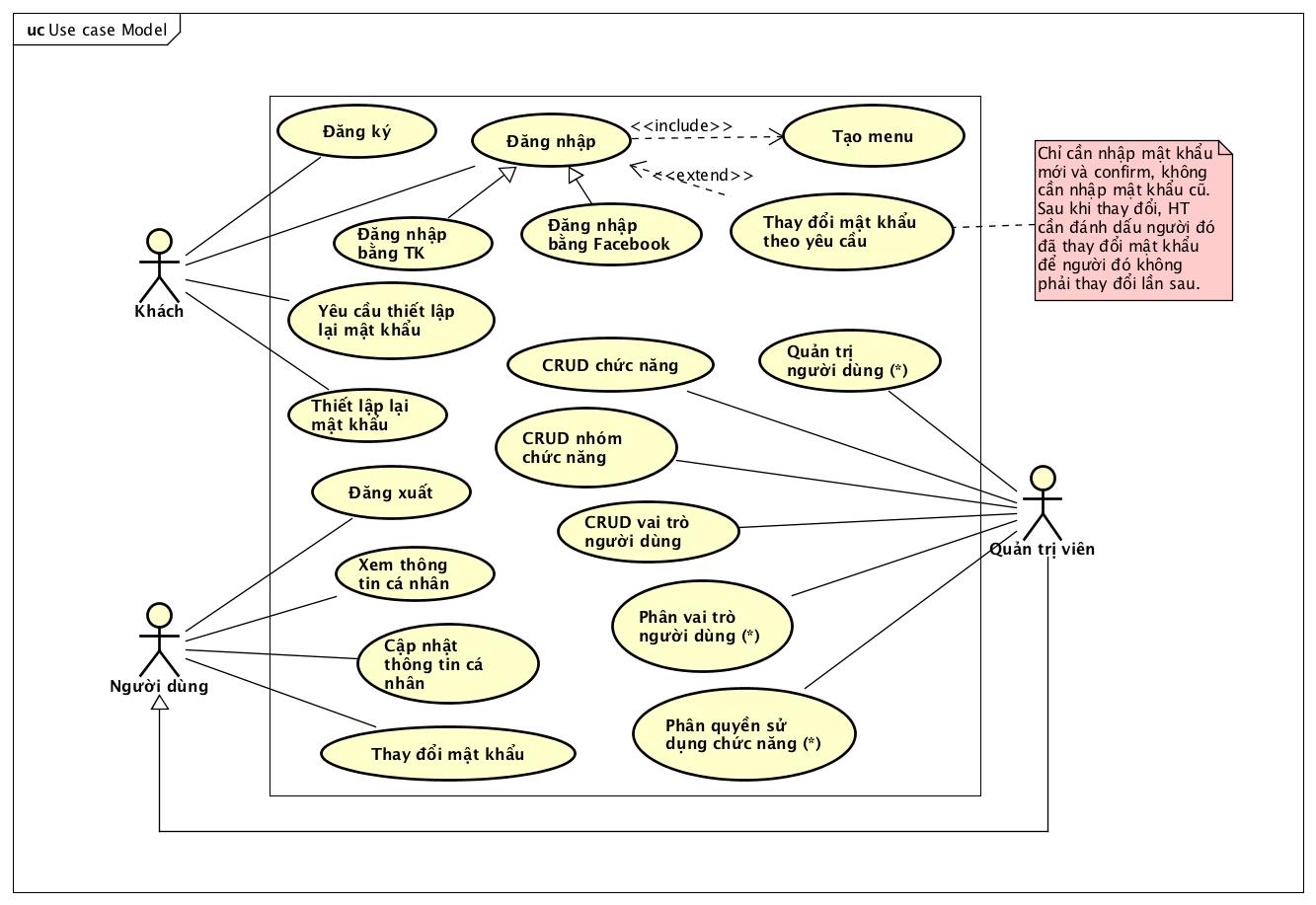
### Biểu đồ use case tổng quan

<Vẽ và giới thiệu về biểu đồ use case tổng quan>

Ví dụ:

Khi chưa đăng nhập, khách có thể đăng ký tài khoản mới, đăng nhập, yêu cầu thiết lập lại mật khẩu khi quên mật khẩu, và thiết lập lại mật khẩu khi nhận được chỉ dẫn thiết lập qua email. Khi khách đăng nhập thành công, hệ thống tạo ra menu chứa các chức năng tương ứng với nhóm người dùng mà người dùng đó thuộc về. Khi khách yêu cầu thiết lập lại mật khẩu, hệ thống thực hiện tạo token và gửi chỉ dẫn thiết lập lại mật khẩu qua email.

Sau khi đăng nhập, người dùng có thể xem và cập nhật thông tin cá nhân của mình, và có thể thay đổi thay đổi mật khẩu.



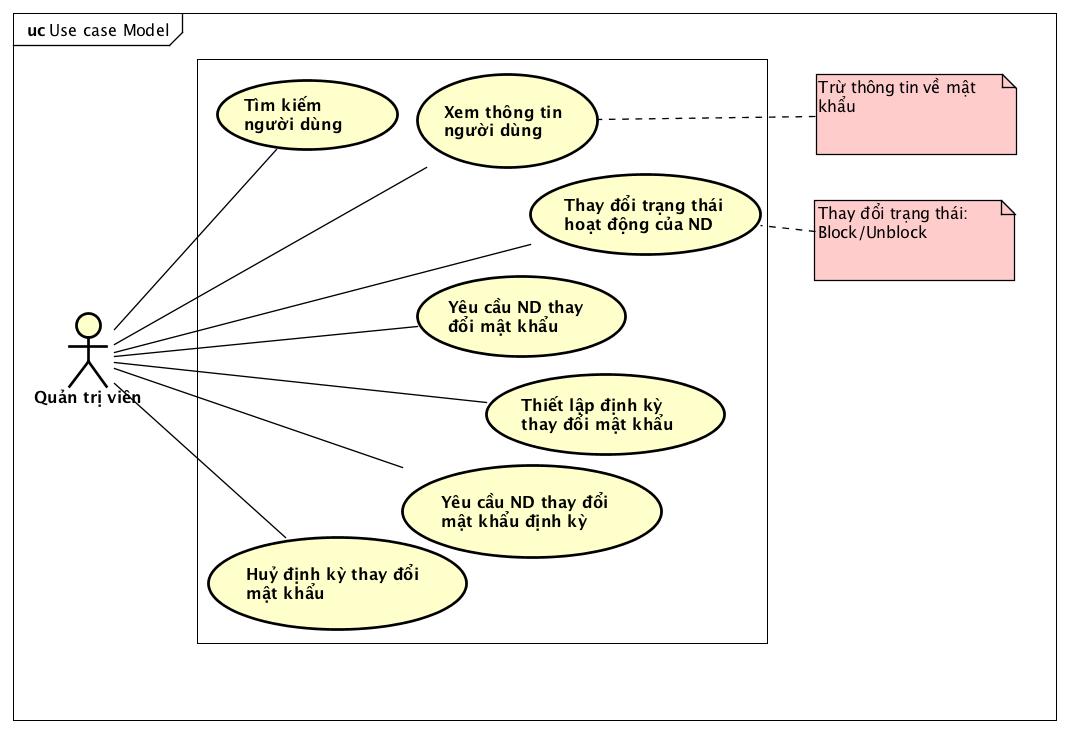
Quản trị viên có thể quản lý người dùng và nhóm người, chức năng và loại chức năng có trong hệ thống. Đồng thời, quản trị viên có thể phân nhóm cho người dùng, và phân quyền sử dụng chức năng cho nhóm người dùng. Các use case của quản trị viên trong biểu đồ use case tổng quan này là use case phức hợp của một nhóm các use case. Chi tiết về các use case phức này được đưa ra trong các biểu đồ phân rã ở phần sau.

### Biểu đồ use case phân rã mức 2

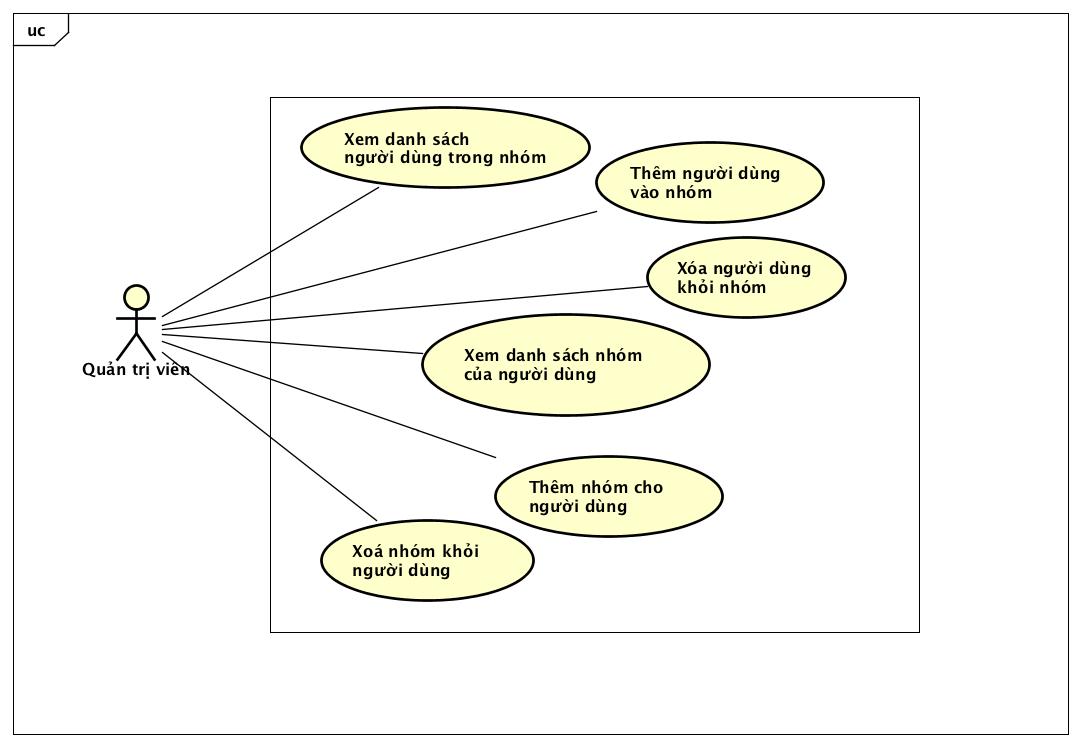
<Vẽ và giới thiệu biểu đồ use case phân rã cho những use case phức tạp>

Ví dụ:

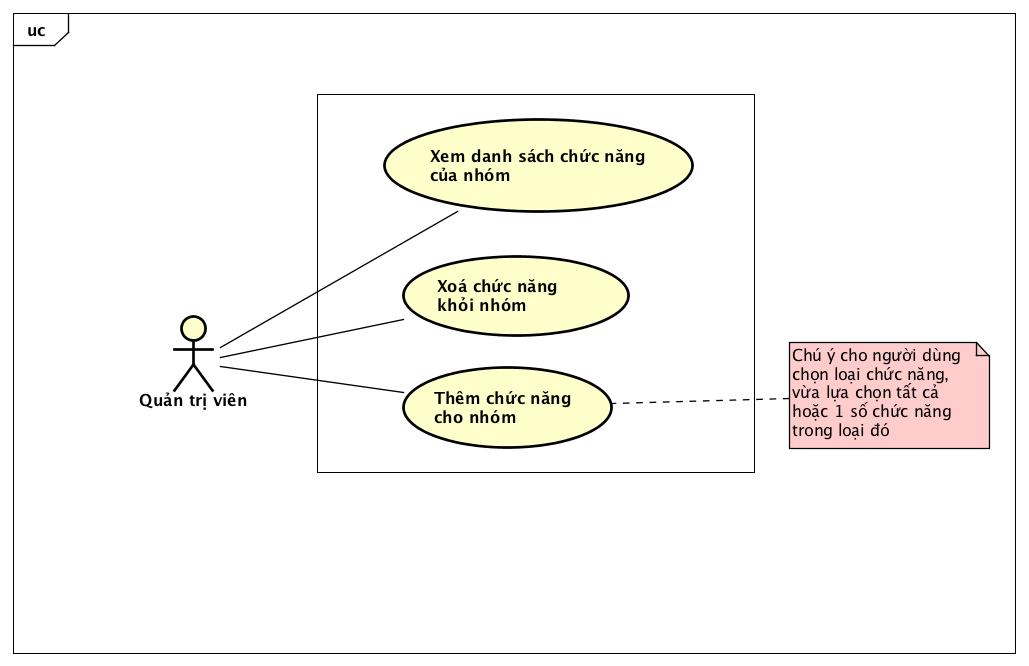
Phân rã use case “Quản trị người dùng”



Phân rã use case “Phân nhóm người dùng”



Phân rã use case “Phân quyền sử dụng chức năng”



## Đặc tả use case

<Đặc tả đầy đủ các usecase, biểu thị cho những nghiệp vụ của bài toán. Miêu tả tất cả các vấn đề liên quan đến một ca sử dụng. Dùng làm cơ sở cho các định hướng phân tích, thiết kế, mã hóa, kiểm thử theo hướng ca sử dụng ở các công đoạn tiếp theo.>

<Có thể vẽ thêm biểu đồ hoạt động mô tả rõ hơn cho quy trình nghiệp vụ trong usecase>

Ví dụ:

+ Thực hiện đặc tả ca sử dụng theo mẫu sau:

***Đặc tả use case UC001 “Đăng nhập”***

|  |  |  |  |
| --- | --- | --- | --- |
| **Mã Use case** | UC001 | **Tên Use case** | Đăng nhập |
| **Tác nhân** | Quản lý tổ dân phố | | |
| **Mô tả** | Cho phép một người dùng có tài khoản truy cập vào hệ thống | | |
| **Tiền điều kiện** | Không | | |
| **Luồng sự kiện chính**  **(Thành công)** | |  |  |  | | --- | --- | --- | | **STT** | **Thực hiện bởi** | **Hành động** | |  | Quản lý tổ dân phố | chọn chức năng Đăng nhập | |  | Hệ thống | hiển thị giao diện đăng nhập | |  | Quản lý tổ dân phố | nhập username và mật khẩu (mô tả phía dưới \*) | |  | Quản lý tổ dân phố | yêu cầu đăng nhập | |  | Hệ thống | kiểm tra xem tác nhân đã nhập các trường bắt buộc nhập hay chưa | |  | Hệ thống | kiểm tra username và mật khẩu có hợp lệ do tác nhân nhập trong hệ thống hay không | |  | Hệ thống | tạo giao diện màn hình chính | | | |
| **Luồng sự kiện thay thế** | |  |  |  | | --- | --- | --- | | **STT** | **Thực hiện bởi** | **Hành động** | | 6a. | Hệ thống | thông báo lỗi: Cần nhập các trường bắt buộc nhập nếu người dùng nhập thiếu | | 7a. | Hệ thống | thông báo lỗi: username và/hoặc mật khẩu chưa đúng nếu không tìm thấy username và mật khẩu trong hệ thống | | | |
| **Hậu điều kiện** | Không | | |

\* Dữ liệu đầu vào của thông tin cá nhân gồm các trường dữ liệu sau:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Trường dữ liệu** | **Mô tả** | **Bắt buộc?** | **Điều kiện hợp lệ** | **Ví dụ** |
|  | Username |  | Có |  | hung |
|  | Mật khẩu |  | Có |  | ToiLa12#$ |

***Đặc tả use case UC002 “Tạo mới sổ hộ khẩu”***

|  |  |  |  |
| --- | --- | --- | --- |
| **Mã Use case** | UC002 | **Tên Use case** | Tạo mới sổ hộ khẩu |
| **Tác nhân** | Quản lý tổ dân phố | | |
| **Mô tả** | Cho phép một người dùng đã đăng nhập tạo ra một sổ hộ khẩu mới | | |
| **Tiền điều kiện** | Tác nhân đã đăng nhập thành công | | |
| **Luồng sự kiện chính**  **(Thành công)** | |  |  |  | | --- | --- | --- | | **STT** | **Thực hiện bởi** | **Hành động** | |  | Quản lý tổ dân phố | Nhấn nút **tạo mới hộ khẩu** | |  | Hệ thống | hiển thị giao diện tạo mới hộ khẩu | |  | Quản lý tổ dân phố | nhập thông tin đầy đủ (thông tin về chủ hộ và các thành viên) (mô tả phía dưới \*) | |  | Quản lý tổ dân phố | nhấn nút **xác nhận** | |  | Hệ thống | kiểm tra xem tác nhân đã nhập các trường bắt buộc nhập hay chưa | |  | Hệ thống | kiểm tra số hộ khẩu có hợp lệ do tác nhân nhập trong hệ thống hay không (số hộ khẩu không được trùng với hộ khẩu đã có) | |  | Hệ thống | Gọi usecase “Thêm nhân khẩu vào hộ khẩu” và thiết lập thông tin chủ hộ | | | |
| **Luồng sự kiện thay thế** | |  |  |  | | --- | --- | --- | | **STT** | **Thực hiện bởi** | **Hành động** | | 6a. | Hệ thống | thông báo lỗi: Cần nhập các trường bắt buộc nhập nếu người dùng nhập thiếu | | 7a. | Hệ thống | thông báo lỗi: số hộ khẩu không hợp lệ | | | |
| **Hậu điều kiện** | Thông tin sổ hộ khẩu mới được lưu vào cơ sở dữ liệu | | |

\* Dữ liệu đầu vào của thông tin cá nhân gồm các trường dữ liệu sau:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Trường dữ liệu** | **Mô tả** | **Bắt buộc?** | **Điều kiện hợp lệ** | **Ví dụ** |
|  | Số hộ khẩu |  | Có |  | TQB002 |
|  | Khu vực |  | Không |  | HN03 |
|  | Địa chỉ |  | Không |  |  |
|  | Ngày lập |  | Không |  |  |
|  | Chủ hộ |  | Không | Chủ hộ có thể thiết lập sau |  |

## Các yêu cầu phi chức năng

<Đưa ra các yêu cầu khác nếu có, bao gồm các yêu cầu phi chức năng như hiệu năng, độ tin cậy, tính dễ dùng, tính dễ bảo trì; hoặc các yêu cầu về mặt kỹ thuật như về CSDL, công nghệ sử dụng…>

**Chức năng (Functionality)**

<Các yêu cầu về chức năng chung cho nhiều use case được ghi ở đây>

* Trong các chuỗi sự kiện của các use case, tất cả các bước có thao tác với CSDL, nếu có lỗi trong quá trình kết nối hoặc thao tác, cần có thông báo lỗi tương ứng để tác nhân biết là lỗi liên quan đến CSDL chứ không liên quan tới lỗi của người dùng
* Các use case do Quản trị viên và Người dùng sử dụng thì Khách cần đăng nhập với vai trò tương ứng
* Định dạng hiển thị chung như sau:
  + Số căn phải
  + Chữ căn trái
  + Font: Arial 14, màu đen
  + Nền trắng

**Tính dễ dùng (Usability)**

Các chức năng cần được thiết kế sao cho dễ thao tác. Cần có hướng dẫn cụ thể lỗi sai của người dùng để người dùng biết định vị lỗi, biết lỗi gì và biết cách sửa lỗi.

**Các yêu cầu khác**

<Mô tả các yêu cầu khác tại đây, trên mục này có thể thêm các đặc tính chất lượng khác như Hiệu năng – Efficiency/Performance, Tính tin cậy – Reliability, Tính dễ bảo trì – Maintainability, Tính khả chuyển – Portability, Yêu cầu về an toàn bảo mật, Yêu cầu về giao diện,…>

**Ví dụ minh hoạ:**

**Chức năng**

- Hỗ trợ nhiều người dùng làm việc đồng thời.

- Nếu một lớp bị hết chỗ khi một sinh viên đang đăng ký học của lớp đó thì sinh viên này phải được thông báo.

**Tính dễ dùng**

- Giao diện nguời dùng tương thích Windows 7/ Window 10. Thân thiện.

**Tính ổn định**

- Hệ thống phải hoạt động liên tục 24 giờ/ngày, 7 ngày/tuần, với thời gian ngừng hoạt động không quá 10%.

**Hiệu suất**

- Hệ thống phải hỗ trợ đến 2000 người dùng truy xuất CSDL trung tâm đồng thời bất kỳ lúc nào, và đến 500 người dùng truy xuất các server cục bộ.

- Hệ thống phải truy xuất đến CSDL danh mục học phần cũ với độ trễ không quá 10 giây.

- Hệ thống phải có khả năng hoàn tất 80% giao dịch trong vòng 2 phút.

**Sự hỗ trợ**

- Không có.

**Các ràng buộc thiết kế**

- Hệ thống tích hợp với Hệ thống danh mục học phần có sẵn, một CSDL RDBMS.

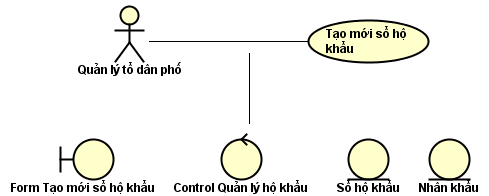
- Hệ thống phải cung cấp giao diện dựa Web.

# PHÂN TÍCH YÊU CẦU

## Xác định các lớp phân tích

<**Phân rã** các yêu cầu phức tạp được trình bày trong pha xác định yêu cầu thành các nhân tố chính cùng mối quan hệ giữa chúng để làm cơ sở cho giải pháp được trình bày trong pha thiết kế sau này. Kết quả của quá trình phân rã là **các lớp phân tích**. *Tất cả các hoạt động trong kịch bản của Use-Case phải được phản ánh đầy đủ trong các lớp phân tích*>

- Ví dụ: **Kết quả quá trình phân rã bước đầu của usecase “Tạo mới sổ hộ khẩu”**

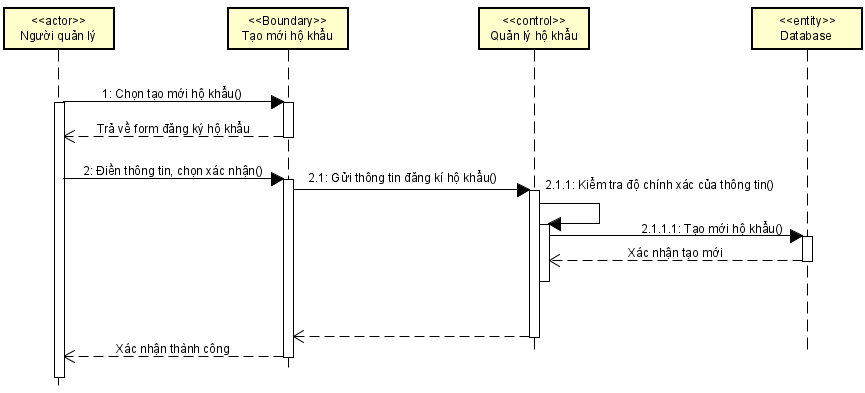


< Thực hiện tương tự với các usecase khác, chúng ta thu được một mô hình phân tích bao gồm các lớp phân tích trong đó. Chú ý: nếu trong các kết quả có các lớp phân tích trùng nhau (ví dụ trong một usecase khác cũng xuất hiện lớp thực thể “Sổ hộ khẩu”) thì cần hợp nhất các lớp phân tích này.>

## Xây dựng biểu đồ trình tự

< **Với mỗi usecase**: chúng ta cần phân bổ trách nhiệm ca sử dụng cho các đối tượng của các lớp phân tích. Đây là một hoạt động quan trọng và đôi khi khó khăn, nó là cơ sở để chúng ta xác định các dữ liệu thành phần (phương thức + thuộc tính) cho mỗi lớp. Kết quả của quá trình này có thể biểu diễn bằng **biểu đồ trình tự (sequence diagram)** hoặc **biểu đồ giao tiếp (communication diagram)** trong UML.>

- **Ví dụ: biểu đồ trình tự cho usecase “Tạo mới sổ hộ khẩu” phân bổ trách nhiệm ca sử dụng cho các đối tượng của các lớp phân tích**



## Xây dựng biểu đồ lớp phân tích

<Xây dựng biểu đồ lớp phân tích từ các kết quả thu được ở trên>

## Xây dựng biểu đồ thực thể liên kết (ERD)

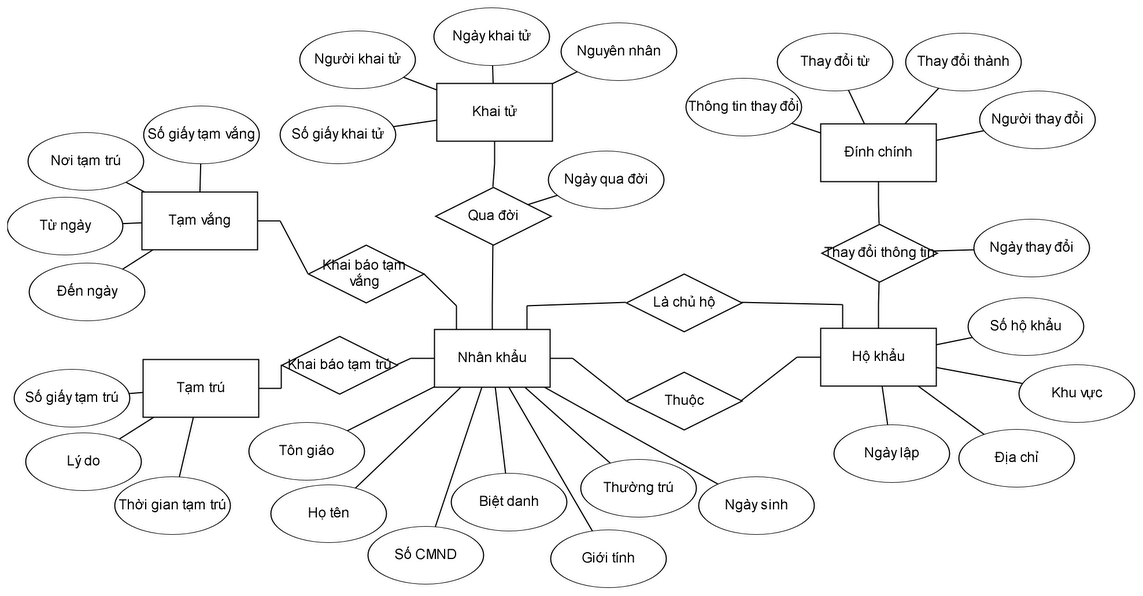
< Thực hiện theo các bước sau:

- Xác định các đối tượng dữ liệu

- Xác định các đặc tính của các đối tượng dữ liệu

- Xác định các mối quan hệ giữa các đối tượng dữ liệu>

- **Ví dụ: biểu đồ ERD cho nhóm chức năng số 1:**



Viết mô tả cho các mối quan hệ giữa các đối tượng dữ liệu

# THIẾT KẾ CHƯƠNG TRÌNH

<Trình bày các nội dung về thiết kế chương trình>

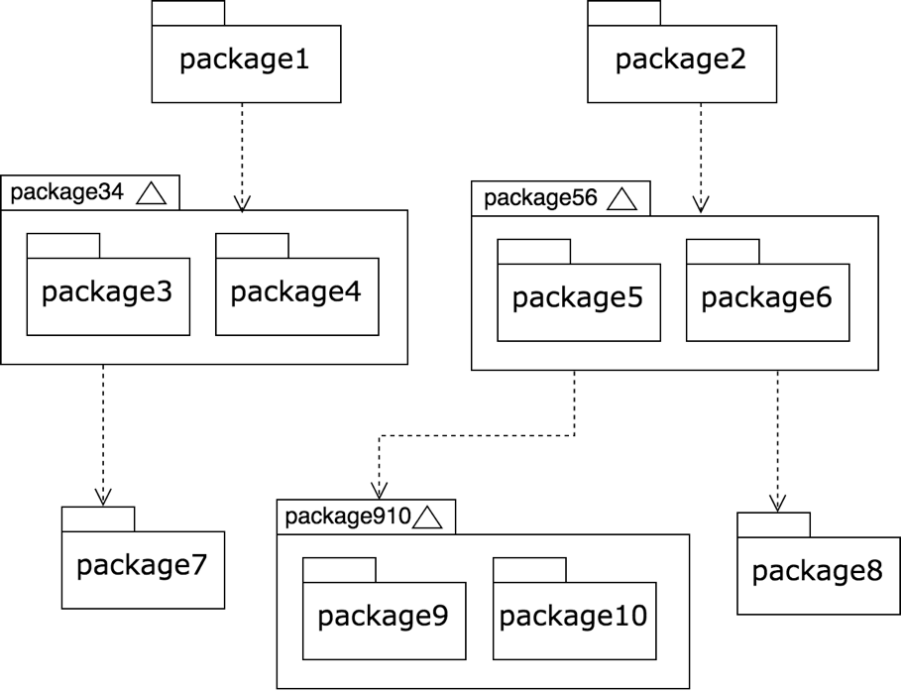
## Thiết kế kiến trúc

< Sinh viên lựa chọn kiến trúc phần mềm cho ứng dụng của mình như: kiến trúc ba lớp MVC, MVP, SOA, Microservice, v.v. rồi giải thích sơ bộ về kiến trúc đó >

< Sử dụng kiến trúc phần mềm đã chọn ở trên, sinh viên mô tả kiến trúc cụ thể cho ứng dụng của mình. Gợi ý: sinh viên áp dụng lý thuyết chung vào hệ thống/sản phẩm của mình như thế nào, có thay đổi, bổ sung hoặc cải tiến gì không. Ví dụ, thành phần M trong kiến trúc lý thuyết MVC sẽ là những thành phần cụ thể nào (ví dụ: là interface I + class C1 + class C2, v.v.) trong kiến trúc phần mềm của sinh viên.>

< Sinh viên vẽ biểu đồ gói UML (UML package diagram), nêu rõ sự phụ thuộc giữa các gói (package). SV cần vẽ các gói sao cho chúng được phân theo các tầng rõ ràng, không được sắp đặt package lộn xộn trong hình vẽ. Sinh viên chú ý các quy tắc thiết kế (Các gói không phụ thuộc lẫn nhau, gói tầng dưới không phụ thuộc gói tầng trên, không phụ thuộc bỏ qua tầng, v.v.) và cần giải thích sơ lược về mục đích/nhiệm vụ của từng package.>

Ví dụ:



## Thiết kế cơ sở dữ liệu

< Xây dựng **các bảng dữ liệu**, xác định **quan hệ giữa các bảng** và **thiết kế chi tiết cho mỗi trường** (cột) trong bảng.

Một số lưu ý khi thiết kế các trường dữ liệu: Field là đơn vị nhỏ nhất của dữ liệu 🡪 Field tương ứng với 1 thuộc tính (attribute) trong mô hình dữ liệu logic. Quyết định cần làm khi thiết kế là phải **chọn kiểu dữ liệu cho field**, **kiểm soát tính toàn vẹn dữ liệu** và Hệ quản trị dữ liệu sẽ quản lý các giá trị bị thiếu cho field như thế nào?>

|  |  |
| --- | --- |
| **Loại đặc tả** | **Mô tả nội dung** |
| Tên trường (field name) | Theo quy định về cách đặt tên hoặc quy ước viết code của nhóm |
| Kiểu trường (data type) | Kiểu dữ liệu phù hợp với các giá trị cần lưu trữ trong các trường |
| Kích cỡ (size) | Kích thước / Khoảng giá trị tối đa và tối thiểu |
| Mã hóa (Coding) | Mã hoá các thông tin lưu trữ / Khuôn dạng của dữ liệu |
| Các quy tắc toàn vẹn dữ liệu (data integrity rules) | Đặc tả về các hạn chế đặt lên giá trị của trường: NULL / NOT NULL/ KHOÁ CHÍNH,… |
| Các kiểm soát bảo trì (maintenance controls) | Chỉ ra những giá trị nào được phép thay đổi |
| Công thức (Formular) | Mô tả công thức tính toán giá trị với những trường cần tính toán. |
| Toàn vẹn tham chiếu (references integrity) | Đặc tả giá trị của trường có liên quan đến giá trị của trường khác, ví dụ: KHOÁ NGOẠI,… |
| Sở hữu (Ownership) | Ai có quyền đối với dữ liệu. |

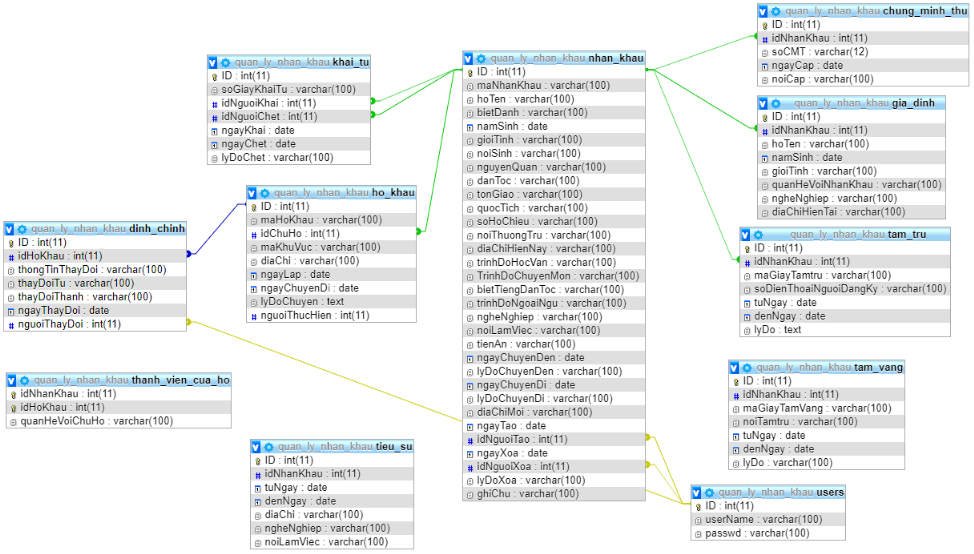
Đặc tả thiết kế dữ liệu cho từng bảng theo mẫu sau:

**Tên bảng:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc toàn vẹn** | **Khuôn dạng** | **Ghi chú** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Ví dụ: Thiết kế cơ sở dữ liệu cho nhóm chức năng số 1**

* Sơ đồ quan hệ giữa các bảng:



* Đặc tả thiết kế dữ liệu cho từng bảng:

**Bảng HoKHau:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Ràng buộc toàn vẹn** | **Khuôn dạng** | **Ghi chú** |
| **ID** | Int |  | Khoá chính | Số nguyên dương |  |
| maHoKhau | Varchar(100) | 100 ký tự |  | Văn bản |  |
| idChuHo | Int |  | Khoá tham chiếu từ bảng NhanKhau | Số nguyên dương |  |
| maKhuVuc | Varchar(100) | 100 ký tự |  | Văn bản |  |
| diaChi | Varchar(100) | 100 ký tự |  | Văn bản |  |
| ngayLap | Date |  |  | Ngày tháng năm |  |
| ngayChuyenDi | Date |  |  | Ngày tháng năm |  |
| lyDoChuyen | Text | Văn bản dài |  | Văn bản |  |
| nguoiThucHien | Int |  | Mã tài khoản thêm thông tin | Số nguyên dương |  |

Các bảng còn lại đặc tả tương tự.

## Thiết kế chi tiết các gói

< Sinh viên thiết kế và lần lượt vẽ biểu đồ thiết kế cho từng package, hoặc một nhóm các package liên quan để giải quyết một vấn đề gì đó. Khi vẽ thiết kế gói, sinh viên chỉ cần đưa tên lớp, không cần chỉ ra các thành viên phương thức và thuộc tính. SV tham khảo ví dụ minh họa.

Sinh viên cần vẽ rõ ràng quan hệ giữa các lớp trong biểu đồ. Các quan hệ bao gồm: phụ thuộc (dependency), kết hợp (association), kết tập (aggregation), hợp thành (composition), kế thừa (inheritance), và thực thi (implementation). Các quan hệ này đều đã được minh họa.

Sau khi vẽ hình minh họa, sinh viên cần giải thích ngắn gọn về thiết kế của mình.>

Ví dụ minh hoạ:



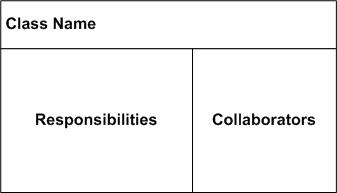
## Thiết kế chi tiết lớp

<Mô tả chi tiết về các lớp>

<Có thể mô tả theo mẫu CRC (class responsibility card)>

<Mô tả các thuộc tính và phương thức trong các lớp>

**Xây dựng mô tả chi tiết cho class theo mẫu:** **“CRC Card Layout”**



**Ví dụ:**

|  |  |
| --- | --- |
| **Class HoKhauModel** | |
| Chứa các thông tin về sổ hộ khẩu:  private int ID;  private String maHoKhau;  private int idChuHo;  private String maKhuVuc;  private String diaChi;  private Date ngayLap;  private Date ngayChuyDi;  private String lyDoChuyen;  private int nguoiThucHien; | Class HoKhauController |



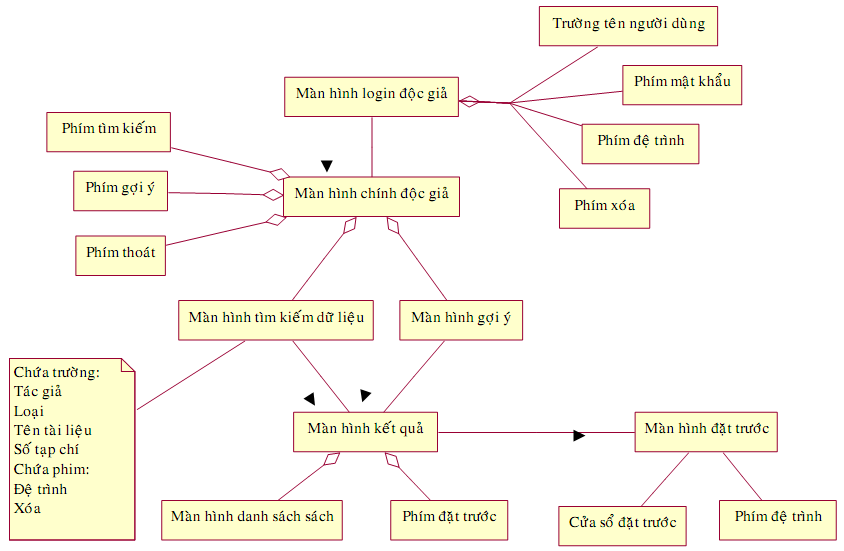
## Sơ đồ lớp chi tiết

<Xây dựng sơ đồ lớp – Class Diagram>

## Thiết kế giao diện

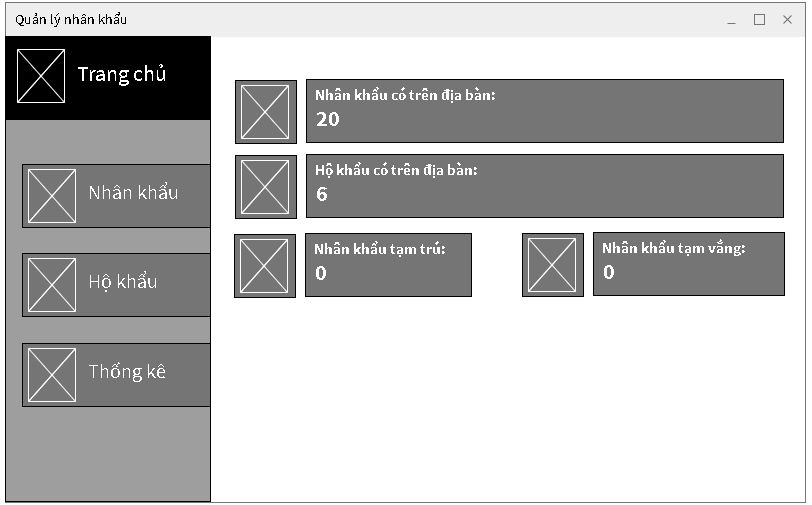
< Xây dựng biểu đồ chuyển giữa các giao diện.>

Ví dụ:



<Thiết kế giao diện mock-up cho từng màn hình>

**Ví dụ: Xây dựng mock-up cho màn hình chính của bài toán (casestudy)**



< Đặc tả thiết kế giao diện cho các màn hình.>

Đặc tả thiết kế giao diện theo mẫu như sau:

**Tên màn hình**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Điều khiển** | **Thông tin dữ liệu** | **Loại** | **Thuộc tính** | **Ghi chú** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Ví dụ: Đặc tả thiết kế màn hình chính của bài toán (casestudy)**

**Màn hình chính của ứng dụng (main screen):** màn hình làm việc sau khi người dùng đăng nhập thành công. Màn hình chứa menu bên trái với các nút liên kết để mở các chức năng chính. Khung bên phải chứa các thông tin thống kê chung về các số liệu (nhân khẩu, hộ khẩu, tạm vắng, tạm trú) trong hệ thống.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Điều khiển** | **Thông tin dữ liệu** | **Loại** | **Thuộc tính** | **Ghi chú** |
| **Tiêu đề màn hình** | Chứa ảnh đại diện và tên màn hình | Image + text label | Hiển thị  Kích thước image  Font chữ |  |
| **Nút menu "Nhân khẩu"** | Khi người dùng click sẽ mở màn hình "Quản lý nhân khẩu" | Image + Button | Hiển thị  Kích thước image  Font chữ  Sự kiện click |  |
| **Nút menu "Hộ khẩu"** | Khi người dùng click sẽ mở màn hình "Quản lý hộ khẩu" | Image + Button | Hiển thị  Kích thước image  Font chữ  Sự kiện click |  |
| **Nút menu "Thống kê"** | Khi người dùng click sẽ mở màn hình "Thống kê" | Image + Button | Hiển thị  Kích thước image  Font chữ  Sự kiện click |  |
| **Khung hiển thị thông tin thống kê nhân khẩu** | HIển thị số nhân khẩu có trên địa bàn | Image + text label + text label (số liệu thống kê từ database) | Hiển thị  Kích thước image  Font chữ |  |
| **…** |  |  |  |  |

# XÂY DỰNG CHƯƠNG TRÌNH MINH HỌA

## Thư viện và công cụ sử dụng

<Sinh viên liệt kê các công cụ, ngôn ngữ lập trình, API, thư viện, IDE, công cụ kiểm thử, v.v. mà mình sử dụng để phát triển ứng dụng. Mỗi công cụ phải được chỉ rõ phiên bản sử dụng. SV nên kẻ bảng mô tả tương tự như Bảng 1. Nếu có nhiều nội dung trình bày, sinh viên cần xoay ngang bảng.>

Danh sách thư viện và công cụ sử dụng

|  |  |  |
| --- | --- | --- |
| **Mục đích** | **Công cụ** | **Địa chỉ URL** |
| IDE lập trình | Eclipse Oxygen 64 bit | http://www.eclipse.org/ |
| v.v. | v.v. | v.v. |

## Kết quả chương trình minh họa

< Sinh viên trước tiên mô tả kết quả đạt được của mình là gì, ví dụ như các sản phẩm được đóng gói là gì, bao gồm những thành phần nào, ý nghĩa, vai trò?

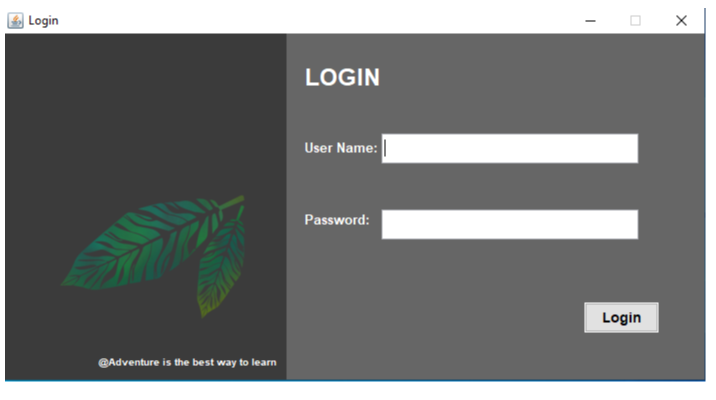
Sinh viên cần thống kê các thông tin về ứng dụng của mình như: số dòng code, số lớp, số gói, dung lượng toàn bộ mã nguồn, dung lượng của từng sản phẩm đóng gói, v.v. Tương tự như phần liệt kê về công cụ sử dụng, sinh viên cũng nên dùng bảng để mô tả phần thông tin thống kê này.>

## Giao diện minh hoạ các chức năng của chương trình

<Chụp các màn hình cho các chức năng quan trọng. Mô tả chi tiết từng chức năng>

Ví dụ:

* Giao diện trang đăng nhập



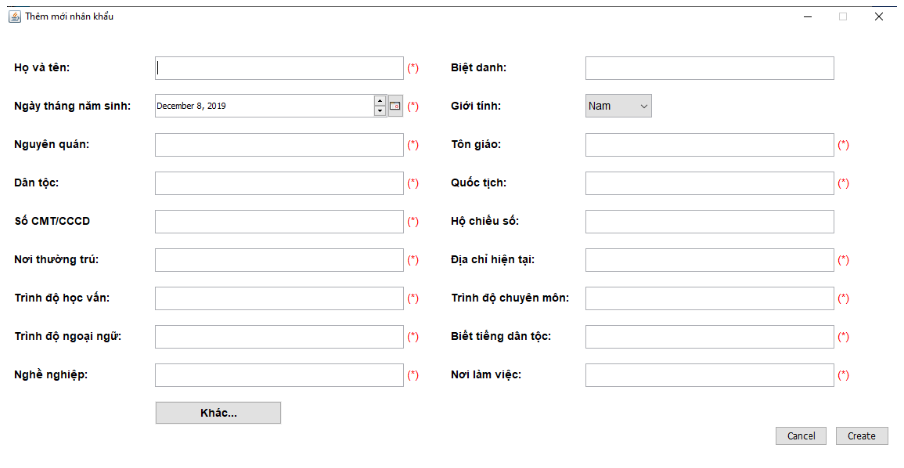
* Giao diện trang chính



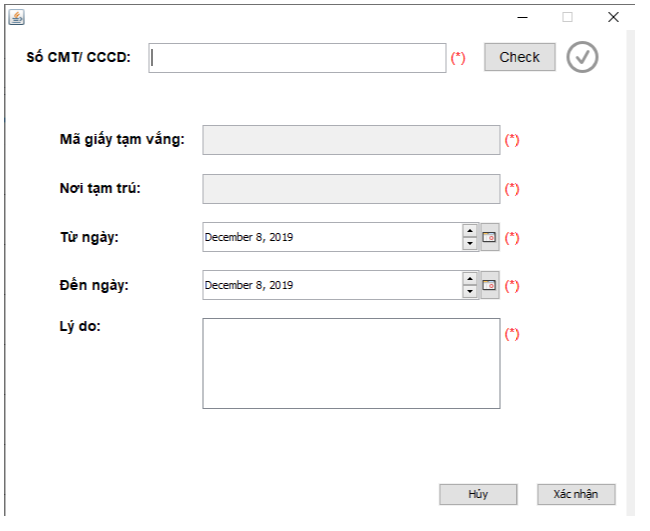
* Giao diện quản lý nhân khẩu



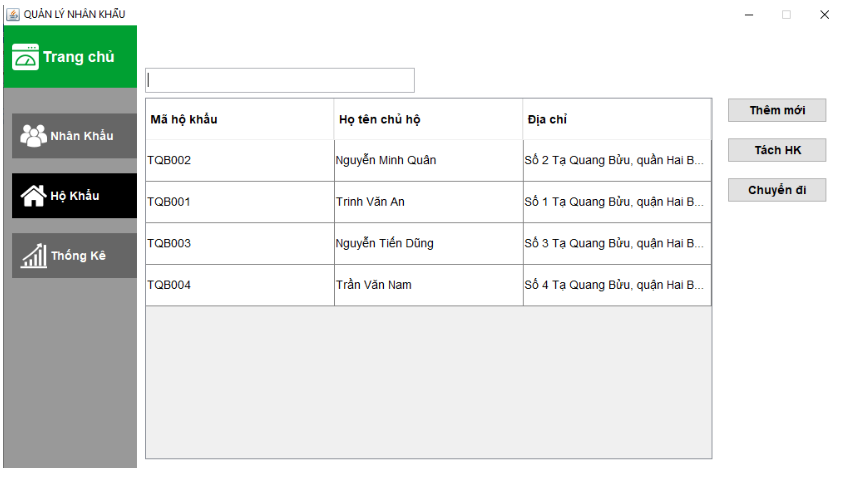
* Giao diện thêm mới nhân khẩu



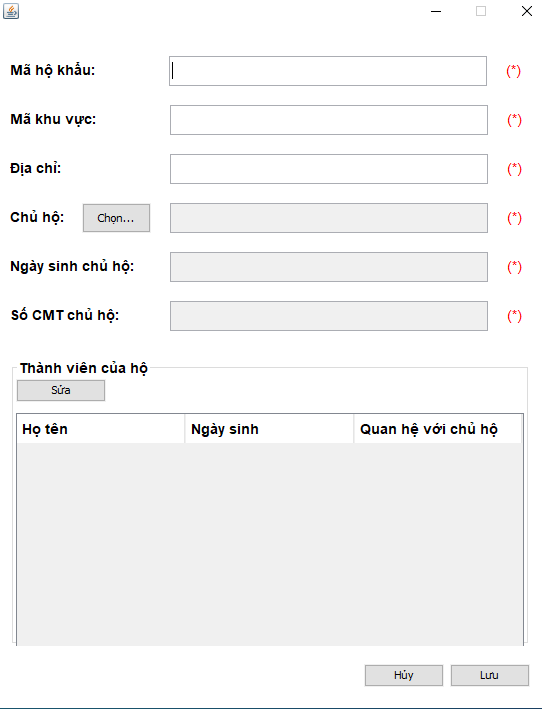
* Giao diện đăng ký tạm vắng



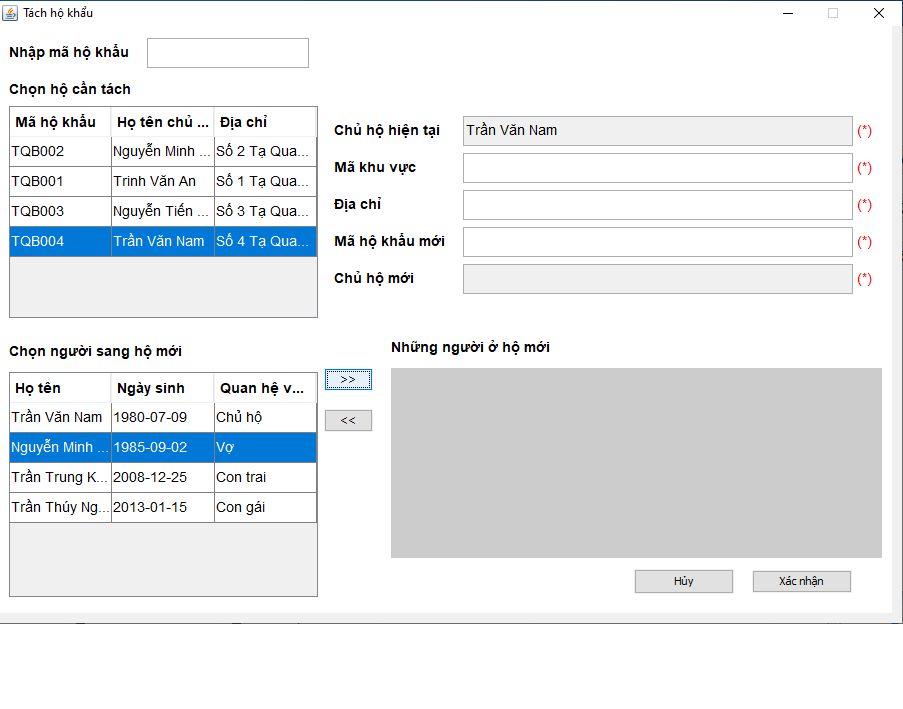
* Giao diện quản lý hộ khẩu



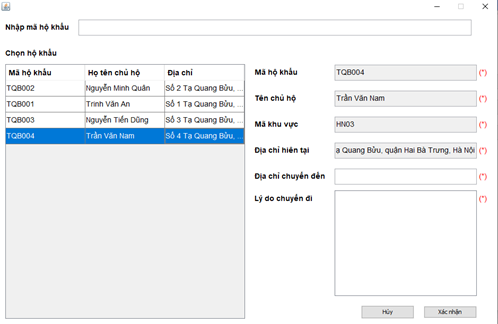
* Giao diện thêm mới hộ khẩu



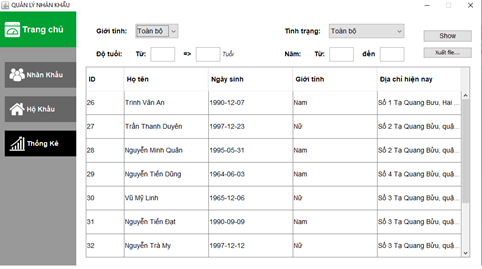
* Giao diện tách hộ khẩu



* Giao diện chuyển hộ khẩu



* Giao diện thống kê



# KIỂM THỬ CHƯƠNG TRÌNH

## Kiểm thử các chức năng đã thực hiện

< Sinh viên thiết kế các trường hợp kiểm thử cho các chức năng. Sinh viên cần chỉ rõ các kỹ thuật kiểm thử đã sử dụng. Sinh viên sau cùng tổng kết về số lượng các trường hợp kiểm thử và kết quả kiểm thử. Sinh viên cần phân tích lý do nếu kết quả kiểm thử không đạt.>

### Kiểm thử cho chức năng 1

<Tên chức năng>

<Đưa ra các trường hợp kiểm thử. Cần cố gắng đưa ra các ngoại lệ. Ví dụ, khi tạo mới dữ liệu, không nhập dữ liệu nào, hoặc chỉ nhập 1 số trường, …>

<Lưu ý: Với mỗi trường hợp kiểm thử, mô tả input, output, exception (không xử lý/xử lý sai/xử lý chuẩn) nếu có. Kết luận có pass hay không>

Mẫu thiết kế testcase:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test No. | *<unique test ID>* | | | |
| Current status |  Passed  Failed  Pending | | | |
| Title | <*unique test tile*> | | | |
| Description | <*short description of this test*> | | | |
| Approach | <*strategy that is being used on this test*> | | | |
| Step No. | Action | Purpose | Expected result | Comment |
|  |  |  |  |  |
|  |  |  |  |  |
| Concluding remark |  | | | |
| Testing team | <*lead and members of testing team*> | | | |
| Date completed |  | | | |

**Ví dụ**

**Chức năng: Thêm mới Book.**

**Bảng 1**: Kết quả kiểm thử chức năng thêm mới Book

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | input | output | Exception | Kết quả |
| 1 | First name = Hùng  Last name = null  Age = xy | Thông báo tạo thành công. Cập nhật vào CSDL | Không xử lý | FAIL |
| 2 | First name = Hùng  Last name = Nam  Age = 12 | Thông báo tạo thành công. Cập nhật vào CSDL | Không xử lý | OK |

### Kiểm thử cho chức năng 2

…

…

### Kiểm thử yêu cầu phi chức năng

<Đánh giá chương trình chạy ổn định, không lỗi/tạm được/kém ổn định>

# HƯỚNG DẪN CÀI ĐẶT VÀ SỬ DỤNG

## Hướng dẫn cài đặt

< Hướng dẫn cài đặt phần mềm nhằm mục đích chỉ ra các yêu cầu và các bước cài đặt, giúp tránh được sai sót trong quá trình tiến hành cài đặt phần mềm>

## Đối tượng, phạm vi sử dụng

<Nêu đối tượng, phạm vi sử dụng của phần mềm>

## Xác định các yêu cầu cài đặt

<Liệt kê cấu hình máy: phần cứng và phần mềm cần thiết>

Ví dụ:

 Yêu cầu phần cứng

- CPU: 1.1 GHz trở lên;

- Bộ nhớ trong (RAM): tối thiểu 2 GB;

 Yêu cầu phần mềm

- Hệ điều hành: Windows 7 trở lên;

- Phần mềm MS Office 2007 trở lên: Bản cài đầy đủ;

- .NET Framework: .NET Framework 4.0 trở lên.

## Hướng dẫn chi tiết các bước cài đặt

< Hướng dẫn chi tiết các bước cài đặt>

## Hướng dẫn sử dụng phần mềm

< Hướng dẫn sử dụng phần mềm>

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

<Phần này nêu kết luận đã làm được gì trong lần quá trình thực hiện bài tập lớn.

Ưu, nhược điểm. Sau đó đưa ra hướng phát triển cho đề tài, chủ yếu để khắc phục các nhược điểm đã nêu>

TÀI LIỆU THAM KHẢO

<Liệt kê thông tin chi tiết về các tài liệu tham khảo đã sử dụng trong quá trình làm bài tập lớn – kể cả tài liệu tiếng Anh, tiếng Việt, trên Internet hay sách, báo…>

[1] Phân tích và thiết kế hệ thống thông tin - Nguyễn Văn Ba - 2003.

[2] Vở ghi môn Phân tích thiết kế hệ thống do thầy Đỗ Văn Uy giảng dạy.

PHỤ LỤC

<Phần này đưa ra các nội dung bổ sung thêm mà trong báo cáo chưa trình bày, một số các vấn đề khác muốn trình bày…>