TRƯỜNG CAO ĐẲNG NGHỀ BÁCH KHOA HÀ NỘI

**KHOA CÔNG NGHỆ THÔNG TIN**

-----🙡🕮🙣-----

**ĐỒ ÁN TỐT NGHIỆP**

**CAO ĐẲNG NGHỀ BÁCH KHOA HÀ NỘI**

**ĐỀ TÀI: XÂY DỰNG HỆ THỐNG ĐĂNG KÝ TRỰC TUYẾN**

**VÀ QUẢN LÝ KÝ TÚC XÁ**

**Sinh viên thực hiện: Nguyễn Thị Hồng Duyên**

**Lớp: LTMT 1 – K8**

**Giảng viên hướng dẫn: Kiều Đức Hạnh**

**Hà Nội, 2019**

**HÀ NỘI, NĂM 20…**

# **LỜI NÓI ĐẦU**

Ngày nay, ngành công nghệ thông tin đang phát triển vô cùng mạnh mẽ. Nó đã chứng tỏ được sức mạnh của mình trong mọi lĩnh vực của cuộc sống. Trong thực tế đã có rất nhiều cơ quan công sở, xí nghiệp, trường học, bệnh viên đã đưa tin học vào ứng dụng. Trong đó việc quản lí thông tin là một ứng dụng cần thiết và mang lại nhiều lời ích. Nó giúp cho các cơ quan, trường học, bệnh viện tiết kiệm thời gian, tiền bạc, nhân lực và có độ chính xác, tin cậy cao.

Việc áp dụng tin học, đưa máy tính vào trợ giúp quản lý ký túc xá sinh viên là rất cần thiết, nó giúp cho người quản lý thu hẹp không gian lưu trữ, tránh được thất lạc dữ liệu, xử lý nhanh một khối lượng lớn về thông tin, tra cứu tìm kiếm một cách nhanh chóng, tiết kiệm được nhiều thời gian và công sức người quản lý và sinh viên.

Xuất phát từ nhận thức trên, với mong muốn áp dụng công nghệ thông tin vào quản lý, dựa trên những hoạt động thực tiến em xin tìm hiểu và thực hiện đề tài “Hệ thống quản lý ký túc xá trường Đại học Bách Khoa Hà Nội” với các chức năng đăng ký phòng trực tuyến và quản lý ký túc xá.

Trong quá trình thực hiện đề tài, mặc dù được sự hướng dẫn rất nhiệt tình của các thầy cùng với sự nỗ lực cá nhân nhưng cũng không thể tránh khỏi những thiếu sót. Em rất mong nhận được những ý kiến đóng góp từ quý Thầy, Cô và bạn bè để em học thêm được nhiều kinh nghiệm và sẽ hoàn thành tốt hơn.

Đặc biệt em xin chân thành cảm ơn thầy Kiều Đức Hạnh, người đã nhiệt tình hướng dẫn em để em có thể hoàn thành đồ án tốt nghiệp một cách tốt nhất.

Em xin chân thành cảm ơn!

Hà Nội, ngày tháng năm 2019

|  |  |  |
| --- | --- | --- |
| BỘ LAO ĐỘNG TB & XÃ HỘI | CỘNG HÒA XÃ HÔI CHỦ NGHĨA VIỆT NAM | |
| **TRƯỜNG CĐN BÁCH KHOA HÀ NỘI** | | **Độc lập - Tự do - Hạnh phúc** |

# **NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP**

Họ và tên sinh viên: Nguyễn Thị Hồng Duyên Mã số sinh viên: CD164125

Khoá: 8 Khoa: Công nghệ thông tin Nghề: Lập trình

1. Tên đề tài: Xây dựng hệ thống đăng ký trực tuyến và quản lý ký túc xá.
2. Nội dung các phần thuyết minh và tính toán:

* Mục tiêu
* Tìm hiểu công nghệ PHP, CodeIgniter, MySQL, HTML, CSS.
* Sử dụng công nghệ đã tìm hiểu để xây dựng hệ thống đăng ký trực tuyến và quản lý ký túc xá.
* Nhiệm vụ
* Tìm hiểu về công nghệ PHP, CodeIgniter, MySQL
* Hoàn thành hệ thống đăng ký trực tuyến và quản lý ký túc xá.

1. Họ tên giảng viên hướng dẫn: Thầy Kiều Đức Hạnh
2. Ngày giao nhiệm vụ đồ án:
3. Ngày hoàn thành đồ án:

Hà Nội, ngày tháng năm 2019

Chủ nhiệm khoa Giảng viên hướng dẫn

# **BẢN TÓM TẮT ĐỒ ÁN TỐT NGHIỆP**

**TÊN ĐỀ TÀI: Xây dựng hệ thống đăng ký trực tuyến và quản lý ký túc xá.**

Sinh viên thực hiện: NGUYỄN THỊ HỒNG DUYÊN

Lớp: LTMT1-K8

Giáo viên hướng dẫn: Thầy Kiều Đức Hạnh

**TÓM TẮT ĐỀ TÀI**

Hiện nay, các website, phần mềm hỗ trợ giáo dục đã xuất hiện và phát triển mạnh mẽ, nhằm đáp ứng nhu cầu của học sinh sinh viên, cũng như hỗ trợ quản lý cho cán bộ công nhân viên của nhà trường. Bên cạnh các website học tập, nhà trường cũng cần quản lý việc lưu trú của sinh viên tại ký túc xá. Hệ thống đăng ký trực tuyến và quản lý ký túc xá được xây dựng với 2 chức năng chính là đăng ký phòng trực tuyến dành cho sinh viên và quản lý ký túc dành cho cán bộ. Hệ thống hướng tới các giải pháp để khắc phục những hạn chế của hệ thống cũ và đem lại lợi ích cho cả sinh viên và cán bộ quản lý.

**CÁC MỤC TIÊU CHÍNH**

* Tìm hiểu công nghệ PHP, CodeIgniter, MySQL, HTML, CSS.
* Ứng dụng công nghệ để xây dựng hế thống đăng ký trực tuyến và quản lý ký túc xá.

**KẾT QUẢ DỰ KIẾN**

* Xây dựng thành công hệ thống và đưa vào sử dụng trong thực tế.
* Thành thạo ngôn ngữ PHP, framework CodeIgniter, MySQL,…
* Viết báo cáo tổng kết

# **MỤC LỤC**

[**LỜI NÓI ĐẦU** 2](#_Toc12052376)

[**NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP** 3](#_Toc12052377)

[**BẢN TÓM TẮT ĐỒ ÁN TỐT NGHIỆP** 4](#_Toc12052378)

[**MỤC LỤC** 5](#_Toc12052379)

[**MỞ ĐẦU** 8](#_Toc12052380)

[**CHƯƠNG I: CƠ SỞ LÝ THUYẾT** 10](#_Toc12052381)

[**1.1** **Ngôn ngữ lập trình PHP** 10](#_Toc12052382)

[1.1.1 Khái niệm PHP 10](#_Toc12052383)

[1.1.2 Cách hoạt động của PHP 10](#_Toc12052384)

[1.1.3 Hằng và biến 11](#_Toc12052385)

[1.1.4 Các kiểu dữ liệu 11](#_Toc12052386)

[1.1.5 Toán tử trong PHP 11](#_Toc12052387)

[1.1.6 Các lệnh điều khiển 12](#_Toc12052388)

[1.1.7 Mảng 13](#_Toc12052389)

[1.1.8 Lập trình hướng đối tượng 13](#_Toc12052390)

[**1.2** **Framework CodeIgniter** 14](#_Toc12052391)

[1.2.1 CodeIgniter là gì? 14](#_Toc12052392)

[1.2.2 Cài đặt CodeIgniter 14](#_Toc12052393)

[1.2.3 Tổ chức dữ liệu trong CodeIgniter 14](#_Toc12052394)

[1.2.4 Mô hình MVC trong CodeIgniter 15](#_Toc12052395)

[1.2.5 Helpers và Libraries trong CodeIgniter 15](#_Toc12052396)

[1.2.6 Lớp Input 16](#_Toc12052397)

[1.2.7 Session trong CodeIgniter 16](#_Toc12052398)

[1.2.8 Query builder trong CodeIgniter 16](#_Toc12052399)

[**1.3** **Hệ quản trị cơ sở dữ liệu MySQL** 18](#_Toc12052400)

[1.3.1 MySQL là gì? 18](#_Toc12052401)

[1.3.2 Các kiểu dữ liệu trong MySQL 18](#_Toc12052402)

[1.3.3 Các cú pháp cơ bản 18](#_Toc12052403)

[1.3.4 Ràng buộc trong MySQL 18](#_Toc12052404)

[1.3.5 Mệnh đề cơ bản 19](#_Toc12052405)

[1.3.6 Các hàm cơ bản trong SQL 19](#_Toc12052406)

[**CHƯƠNG II: ỨNG DỤNG XÂY DỰNG HỆ THỐNG ĐĂNG KÝ TRỰC TUYẾN VÀ QUẢN LÝ KÝ TÚC XÁ** 20](#_Toc12052407)

[**2.1** **Sơ đồ Use case tổng quát hệ thống** 21](#_Toc12052408)

[**2.2** **Sơ đồ Usecase của đối tượng Sinh viên** 21](#_Toc12052409)

[2.2.1 Chức năng đăng nhập 21](#_Toc12052410)

[2.2.2 Chức năng đăng ký tài khoản 22](#_Toc12052411)

[2.2.3 Chức năng đăng ký phòng trực tuyến 22](#_Toc12052412)

[2.2.4 Chức năng xem lịch sử đăng ký phòng 24](#_Toc12052413)

[2.2.5 Chức năng xem danh sách bạn cùng phòng 24](#_Toc12052414)

[2.2.6 Chức năng xem hóa đơn 24](#_Toc12052415)

[**2.3** **Sơ đồ Usecase của đối tượng Cán bộ quản lý** 25](#_Toc12052416)

[2.3.1 Chức năng quản lý sinh viên 25](#_Toc12052417)

[2.3.2 Chức năng quản lý đơn đăng ký phòng 25](#_Toc12052418)

[2.3.3 Chức năng quản lý phòng ký túc 27](#_Toc12052419)

[2.3.4 Chức năng xem hóa đơn 27](#_Toc12052420)

[2.3.5 Chức năng tạo hóa đơn 27](#_Toc12052421)

[2.3.6 Chức năng cập nhật hóa đơn 28](#_Toc12052422)

[2.3.7 Chức năng xuất hóa đơn 28](#_Toc12052423)

[2.3.8 Chức năng gửi báo cáo 28](#_Toc12052424)

[**2.4** **Sơ đồ Usecase của đối tượng Administrator** 29](#_Toc12052425)

[2.4.1 Chức năng quản lý đơn giá 29](#_Toc12052426)

[2.4.2 Chức năng quản lý nhà ký túc xá 29](#_Toc12052427)

[2.4.3 Chức năng quản lý phòng ký túc xá 30](#_Toc12052428)

[2.4.4 Chức năng quản lý nhân viên 30](#_Toc12052429)

[2.4.5 Chức năng phân công quản lý nhà ký túc 30](#_Toc12052430)

[2.4.6 Chức năng xem báo cáo hàng tháng 32](#_Toc12052431)

[2.4.7 Chức năng xuất báo cáo hàng tháng 32](#_Toc12052432)

[**2.5** **Biểu đồ hoạt động của hệ thống** 32](#_Toc12052433)

[2.5.1 Chức năng đăng ký 32](#_Toc12052434)

[2.5.2 Chức năng đăng nhập 33](#_Toc12052435)

[2.5.3 Chức năng đăng ký phòng 33](#_Toc12052436)

[2.5.4 Chức năng tạo hóa đơn 34](#_Toc12052437)

[2.5.5 Chức năng phân công quản lý 35](#_Toc12052438)

[**2.6** **Sơ đồ thực thể liên kết (ERD)** 35](#_Toc12052439)

[**2.7** **Các bảng dữ liệu của hệ thống** 37](#_Toc12052440)

[2.7.1 Bảng “admin” 37](#_Toc12052441)

[2.7.2 Bảng “student” 37](#_Toc12052442)

[2.7.3 Bảng “building” 37](#_Toc12052443)

[2.7.4 Bảng “floor” 38](#_Toc12052444)

[2.7.5 Bảng “room” 38](#_Toc12052445)

[2.7.6 Bảng “term” 38](#_Toc12052446)

[2.7.7 Bảng “bill\_type” 38](#_Toc12052447)

[2.7.8 Bảng “price” 39](#_Toc12052448)

[2.7.9 Bảng “bill” 39](#_Toc12052449)

[2.7.10 Bảng “room\_pay” 39](#_Toc12052450)

[2.7.11 Bảng “assignment” 40](#_Toc12052451)

[2.7.12 Bảng “registration” 40](#_Toc12052452)

[2.7.13 Bảng “position” 40](#_Toc12052453)

[2.7.14 Bảng “religion” 40](#_Toc12052454)

[2.7.15 Bảng “ethnic” 40](#_Toc12052455)

[2.7.16 Bảng “group\_user” 41](#_Toc12052456)

[**CHƯƠNG III: KẾT LUẬN VÀ ĐÁNH GIÁ** 42](#_Toc12052457)

[**PHỤ LỤC** 43](#_Toc12052458)

[**TÀI LIỆU THAM KHẢO** 44](#_Toc12052459)

# **MỞ ĐẦU**

Trong thời đại công nghệ 4.0 hiện nay, các website, phần mềm hỗ trợ giáo dục đã xuất hiện và phát triển mạnh mẽ, nhằm đáp ứng nhu cầu của học sinh sinh viên, cũng như hỗ trợ quản lý cho cán bộ công nhân viên của nhà trường. Trong đó, việc quản lý sinh viên đăng ký lưu trú tại ký túc xá của nhà trường cũng là một bài toán thu hút sự quan tâm của nhiều người.

1. **Lý do chọn đề tài**

Ngày nay, với sự phát triển của các trường học, trường đại học, cao đẳng về quy mô diện tích sẽ dẫn đến nhu cầu lưu trú tại trường của sinh viên tăng cao. Với cách làm truyền thống, đầu năm học hoặc đầu kỳ học, sinh viên sẽ cần đến ban quản lý của ký túc xá để đăng ký lưu trú. Điều đó dẫn đến sự quá tải cho ban quản lý ký túc vì sinh viên đăng ký liên tục. Sinh viên đăng ký lưu trú cần chuẩn bị giấy tờ thủ tục đầy đủ trước khi đăng ký cũng là một vấn đề gây ra sự chậm trễ cho việc sắp xếp của ban quản lý. Khi số lượng dữ liệu tăng dẫn đến việc quản lý trên sổ sách dễ gây nhầm lẫn, thất lạc giấy tờ. Bên cạnh đó, sinh viên sẽ phải chờ đợi để đăng ký phòng, nếu gặp trường hợp ký túc xá hết phòng, hết chỗ thì phải tìm ký túc xá khác để đăng ký. Việc sắp xếp phòng phụ thuộc vào ban quản lý ký túc khiến sinh viên khá bị động.

Hệ thống đăng ký trực tuyến và quản lý ký túc xá được xây dựng nhằm giải quyết các hạn chế trên của hệ thống cũ. Từ đó có thể tiết kiệm thời gian, công sức của cả sinh viên và cán bộ quản lý ký túc.

Với mục đích trên, em chọn đề tài “Xây dựng hệ thống đăng ký trực tuyến và quản lý ký túc xá” để giải quyết vấn đề đăng ký và quản lý ký túc xá làm đề tài tốt nghiệp của mình.

1. **Đối tượng và phạm vi đề tài**

* Đối tượng: sinh viên và cán bộ quản lý ký túc xá.
* Phạm vi: đăng ký phòng trực tuyến và quản lý sinh viên lưu trú.

1. **Phương pháp nghiên cứu**

* Khảo sát nhu cầu thực tế của sinh viên
* Tham khảo hệ thống quản lý truyền thống
* Xây dựng cơ sở dữ liệu
* Lập trình website và hoàn thiện

1. **Yêu cầu của hệ thống**

Yêu cầu chức năng

* Phần mềm dễ sử dụng, người dùng ít phải thao tác.
* Giao diện đơn giản, đẹp mắt.
* Phần mềm chạy nhanh, mượt mà, không gặp lỗi.
* Tài nguyên sử dụng ít, công suất tối đa.

Yêu cầu phi chức năng

* Quản lý sinh viên đăng kí lưu trú, cán bộ phụ trách
* Quản lý các nhà, phòng ký túc
* Quản lý chi phí lưu trú.

1. **Công nghệ**

* Ngôn ngữ PHP
* Framework CodeIgniter

1. **Nội dung đồ án**

Nội dung đồ án gồm 3 chương:

***Chương 1: Cơ sở lý thuyết***

Nội dung chính của chương là tìm hiểu về PHP cơ bản, framework của PHP là CodeIgniter, hệ quản trị cơ sở dữ liệu MySQL.

***Chương 2: Phân tích và thiết kế hệ thống***

Nội dung của chương này là phân tích và thiết kế đối tượng, chức năng của hệ thống và phân tích thiết kế cơ sở dữ liệu.

***Chương 3: Kết luận và đánh giá***

Chương nêu ra kết quả của đề tài và hướng phát triển của hệ thống đã xây dựng.

# **CHƯƠNG I: CƠ SỞ LÝ THUYẾT**

* 1. **Ngôn ngữ lập trình PHP**
     1. Khái niệm PHP

PHP - viết tắt hồi quy của **"Hypertext Preprocessor"**, là một ngôn ngữ lập trình kịch bản được chạy ở phía server nhằm sinh ra mã html trên client. PHP đã trải qua rất nhiều phiên bản và được tối ưu hóa cho các ứng dụng web, với cách viết mã rõ rãng, tốc độ nhanh, dễ học nên PHP đã trở thành một ngôn ngữ [lập trình web](http://hoclaptrinhweb.org/hoc-thiet-ke-web/lap-trinh-web.html) rất phổ biến và được ưa chuộng.

* + 1. Cách hoạt động của PHP

Khi người sử dụng gọi trang PHP, Web Server sẽ triệu gọi PHP Engine để thông dịch dịch trang PHP và trả kết quả cho người dùng như hình bên dưới.



Hình 1.1: Sơ đồ hoạt động của PHP

* + 1. Hằng và biến

**Biến**: Không giống như hằng, biến không cần định nghĩa, có thể thay đổi trong chương trình. Cách đặt tên của biến cũng giống như cách đặt tên của hằng. Biến phân biệt chữ hoa và chữ thường. Tất cả biến trong PHP đều bắt đầu bằng dấu "$". PHP tự động hiểu các kiểu biến nên người dùng không cần phải khai báo kiểu biến.

**Hằng**: Tương tự như trong những ngôn ngữ lập trình khác, hằng trong PHP cũng là một giá trị được khai báo trước khi sử dụng. Nó phải bắt đầu bằng chữ cái hoặc dấu gạch dưới \_ và không được bắt đầu bằng số, nó có thể là một chuỗi. Hằng trong PHP được định nghĩa bằng hàm define(). Nó không thay đổi và không được được định nghĩa lại.

* + 1. Các kiểu dữ liệu

PHP có tám loại dữ liệu được sử dụng cho các biến, các kiểu dữ liệu này thường được hiểu ngầm dựa vào giá trị của biến mà không cần phải khai báo.

* Integers: là kiểu số nguyên, không có dấu thập phân, ví dụ 1234
* Doubles: là kiểu số thực, ví dụ 3.64576
* Booleans: chỉ có hai giá trị true hoặc false
* NULL: là một kiểu dữ liệu đặc biệt chỉ có một giá trị: NULL
* Strings: là chuỗi ký tự.
* Array: kiểu dữ liệu mảng - là một tập hợp các phần tử có cùng kiểu dữ liệu với nhau.
* Object: là 1 thể hiện của 1 lớp đã được định nghĩa trước - được sử dụng trong lập trình hướng đối tượng.
* Resource: là kiểu dữ liệu đặc biệt dùng để lưu trữ tham chiếu đến các tài nguyên bên ngoài PHP chẳng hạn như kết nối đến database.
  + 1. Toán tử trong PHP
* Toán tử số học: +, -, \*, /, %, ++, --
* Toán tử so sánh: ==, !=, >, <, >=, <=
* Toán tử gán: =, +=, -=, \*=, /=, %=
  + 1. Các lệnh điều khiển
* **Câu lệnh if else:** câu lệnh này cho phép chúng ta kiểm tra một điều kiện nào đó, nếu đúng(true) thì thực hiện tiếp những gì trong block của if, nếu sai thì thực hiện các câu lệnh trong else.

if (condition) {  
    #code;  
} else {  
    #code;  
}

* **Câu lệnh switch case**: nếu có nhiều điều kiện ràng buộc thay vì phải if else nhiều lần ta có thể sử dụng switch case để thay thế.

switch(n) {  
    case label1: #code; break;

case label2: #code; break;

. . .   
    default: #code;  
}

* **Câu lệnh while**: nếu biểu thức trong while trả về giá trị TRUE thì tiếp tục thực hiện câu lệnh, sau khi thực hiện câu lệnh thì kiểm tra lại biểu thức, cứ tiếp tục như vậy cho tới khi nào biểu thức trả về giá trị FALSE thì thoát khỏi vòng lặp.

while (condition is true) {  
    #code;  
}

* **Câu lệnh do while**: kiểm tra giá trị điều kiện vào cuối vòng lặp. Điều này có nghĩa là nó luôn luôn thực hiện đoạn lệnh cần lặp ít nhất một lần.

do {  
    #code;  
} while (condition is true);

* Câu lệnh for: được sử dụng để lặp với số lần nhất định với điều kiện được định nghĩa trước.

for(*init counter;test counter;increment counter*) {  
  *code to be executed;*  
}

* Câu lệnh foreach: Câu lệnh foreach chỉ làm việc với mảng.

foreach ($*array*as$*value*) {  
    *code to be executed;*  
}

* **Câu lệnh break**: được sử dụng để dừng việc thực thi của các vòng lặp for, foreach, while, do-while và switch.

break;

* **Câu lệnh continue**: bỏ qua lần lặp hiện thời và tiếp tục thực hiện các lần lặp tiếp theo trong các vòng lặp for, while, do-while

continue;

* + 1. Mảng

Mảng là một danh sách các phần tử có cùng kiểu dữ liệu. Mảng có thể là mảng một chiều hay nhiều chiều. Các loại mảng trong PHP:

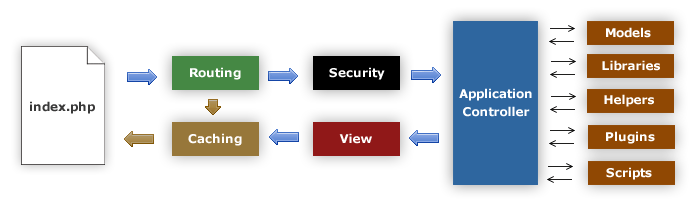
* Mảng số (numeric array)
* Mảng  kết hợp (associative array)
* Mảng đa chiều
  + 1. Lập trình hướng đối tượng

Lập trình hướng đối tượng (OOP) là kiểu viết mã sử dụng lớp, quan hệ, thuộc tính và phương thức của đối tượng nhằm giảm thời gian viết mã và giúp đơn giản hóa việc lập trình. Các đặc điểm của lập trình hướng đối tượng:

* Tính trừu tượng (Abstraction )
* Tính đóng gói (Encapsulation)
* Tính kế thừa (Inheritance)
* Tính đa hình (Polymorphism)
  1. **Framework CodeIgniter**
     1. CodeIgniter là gì?

CodeIgniter là nền tảng ứng dụng web nguồn mở được viết bằng ngôn ngữ PHP bởi Rick Ellis. Phiên bản đầu tiên được phát hành vào ngày 28/02/2006. Hiện tại CodeIgniter đang được phát triển bởi ExpressionEngine Development Team thuộc EllisLab, Inc.

* + 1. Cài đặt CodeIgniter
* Yêu cầu hệ thống: PHP phiên bản 4.x hoặc cao hơn; hệ quản trị cơ sở dữ liệu: MySQL (4.1+), MySQLi, Mircrosoft SQL Server, Oracle, SQLite,...
* Download bộ nguồn CodeIgniter tại website <http://codeigniter.com/>
* Mở tập tin application/config/config.php, thay đổi giá trị $config['base\_url']. Đây là đường dẫn tuyệt đối đến thư mục CodeIgniter trên server.
* Thiết lập dữ liệu trong tập tin application/config/database.php
  + 1. Tổ chức dữ liệu trong CodeIgniter



Hình 1.2: Sơ đồ tổ chức dữ liệu trong CodeIgniter

* index.php đóng vai trò làm controller đầu vào, thiết lập các tài nguyên cần thiết cho hệ thống.
* Routing: Điều hướng giúp xác định các yêu cầu và hướng xử lý đối với chúng.
* Caching: Nếu dữ liệu được yêu cầu đã lưu trong bộ đệm, CodeIgniter sẽ trả dữ liệu trong bộ đệm về phía client. Quá trình xử lý kết thúc.
* Security: Dữ liệu được lọc để phòng chống XXS hoặc SQL Injection.
* Application Controller: Controller xử lý dữ liệu nhận được và trả về kết quả.
* View: Dữ liệu được chuyển qua View để hiển thị cho người dùng.
  + 1. Mô hình MVC trong CodeIgniter
  + Model: những lớp được xây dựng nhằm thực hiện việc trao đổi thông tin với cơ sở dữ liệu. Một lớp Model có thể thực hiện các tác vụ truy vấn, thêm, xóa, cập nhật dữ liệu.

**class** **Model\_name** **extends** CI\_Model {

#code;

}

* + View: những tập tin HTML được xây dựng nhằm thể hiện dữ liệu trong model thành các giao diện tương t|c với người dùng. View có thể là một trang web hoàn chỉnh, hay chỉ là một phần của trang web (header, footer, sidebar...).
* Controller: những lớp đóng vai trò trung gian giữa view và model. Controller nhận các yêu cầu từ phía người dùng, kiểm tra chúng trước khi chuyển qua cho model. Sau khi model xử lý yêu cầu và trả dữ liệu về, controller chuyển sang view để hiển thị dữ liệu cho người dùng.

**class** **Controller\_name** **extends** CI\_Controller {

#code;

}

* Route: trong CodeIgniter cũng giống như trong đa số các PHP framework khác có tác dụng điều hướng request, xác định method nào của Controller Class nào sẽ được gọi.

$route['product/:num'] **=** 'catalog/product\_lookup';

* + 1. Helpers và Libraries trong CodeIgniter
* Helpers: là các [hàm](http://hocphp.info/ham-va-pham-vi-bien-trong-php/)được xây dựng sẵn để sử dụng trong khi lập trình ví dụ như url, language, date,…

Load 1 helper trong controller:

$this**->**load**->**helper('name');

* Libraries: là một tập hợp các chức năng cố định, có tính logic cao và hợp lý, được thể hiện dưới dạng các lớp.

$this**->**load**->**library('name');

* Drivers: Là một loại thư viện đặc biệt có Class cha và nhiều class con. Các class con có thể truy cập vào class cha nhưng không thể truy cập vào class cùng cấp với nó.
  + 1. Lớp Input

Lớp Input phục vụ 2 mục đích đó là:

* + Tiền xử lý dữ liệu đầu vào để bảo mật dữ liệu
  + Cung cấp 1 số phương thức để lấy dữ liệu đầu vào và tiền xử lý dữ liệu đó.

Truy xuất dữ liệu từ form

* Phương thức post: lấy dữ liệu từ form có phương thức post, dữ liệu sẽ được ẩn đi để đảm bảo an toàn về bảo mật.

Với cú pháp: $this**->**input**->**post('some\_data');

* Phương thức get: lấy dữ liệu từ form có phương thức get, dữ liệu được hiển thị trên URL. Với cú pháp: $this**->**input**->**get('some\_data');
* Phương thức cookie: lấy dữ liệu từ cookie từ trình duyệt.

Với cú pháp: $this**->**input**->**cookie('some\_cookie');

* Phương thức server: lấy dữ liệu từ form mà không quan tâm phương thức là post hay get. Với cú pháp: $this**->**input**->**server('some\_data');
  + 1. Session trong CodeIgniter

Sesion cho phép duy trì trạng thái của người dùng đăng nhập và hệ thống và theo dõi hoạt động khi người dùng duyệt trang web.

* Ghi session: $this**->**session**->**set\_userdata('some\_name', 'some\_value');
* Kiểm tra tồn tại: $this**->**session**->**has\_userdata('some\_name');
* Truy xuất: $this**->**session**->**userdata('some\_name');
* Xóa session: $this**->**session**->**unset\_userdata('some\_name');
* Flashdata: $this**->**session**->**set\_flashdata('some\_name', 'some\_value');
  + 1. Query builder trong CodeIgniter

Kết nối cơ sở dữ liệu: để kết nối cơ sở dữ liệu với CodeIgniter, cần khai báo các tham số sau:

$config['hostname'] **=** 'localhost';

$config['username'] **=** 'myusername';

$config['password'] **=** 'mypassword';

$config['database'] **=** 'mydatabase';

$config['dbdriver'] **=** 'mysqli';

$config['dbprefix'] **=** '';

$config['pconnect'] **=** **FALSE**;

$config['db\_debug'] **=** **TRUE**;

$config['cache\_on'] **=** **FALSE**;

$config['cachedir'] **=** '';

$config['char\_set'] **=** 'utf8';

$config['dbcollat'] **=** 'utf8\_general\_ci';

$this**->**load**->**database($config);

Lớp Query builder cung cấp hàm để truy xuất và thao tác với cơ sở dữ liệu.

* Lấy dữ liệu từ bảng:

$this**->**db**->**select('title, content, date');

$this**->**db**->**from('mytable');

$query **=** $this**->**db**->**get();

* Mệnh đề join:

$this**->**db**->**join('table', 'table1.key = table2.key');

* Mệnh đề where: $this**->**db**->**where('key', 'value');
* Mệnh đề like: $this**->**db**->**like('key', 'value');
* Mệnh đề group by: $this**->**db**->**group\_by('key', 'value');
* Mệnh đề having: $this**->**db**->**having('key', 'value');
* Mệnh đề order by: $this**->**db**->**order\_by('key', 'value');
* Mệnh đề limit: $this**->**db**->**limit(value);
* Mệnh đề count: $this**->**db**->**count\_all('table');
* Câu lệnh insert: $this**->**db**->**insert('table', $data);
* Câu lệnh update:

$this**->**db**->**update('table', $data, 'key=value');

* Câu lệnh delete:

$this**->**db**->**delete('table', $data, **array**('id' **=>** $id));

* 1. **Hệ quản trị cơ sở dữ liệu MySQL**
     1. MySQL là gì?

1. SQL(Structed Query Language) Ngôn ngữ truy vấn có cấu trúc: là ngôn ngữ theo chuẩn ANSI để định nghĩa và xử lý dữ liệu trong một cơ sở dữ liệu quan hệ. Các phát biểu SQL dùng để truy tìm và cập nhật dữ liệu trong một cơ sở dữ liệu.
2. MySQL là một trong những hệ thống quản lý cơ sở dữ liệu(CSDL) mã nguồn mở quan hệ SQL sử dụng trên web phổ biến nhất hiện nay.
   * 1. Các kiểu dữ liệu trong MySQL

* Kiểu số nguyên: tinyint, smallint, mediumint, int, bigint
* Kiểu số thực: float, decimal, double, real
* Kiểu thời gian: date, datetỉme, timestamp, time, year
* Kiểu chuỗi ký tự: char, varchar, text, tinytext, blob, mediumtext,…
  + 1. Các cú pháp cơ bản
* Tạo cơ sở dữ liệu: CREATE DATABASE database\_name
* Tạo bảng: CREATE TABLE table\_name
* Thêm bản ghi: INSERT INTO table\_name VALUES(value1,…)
* Lấy bản ghi: SELECT \* FROM table\_name
* Chỉnh sửa bản ghi: UPDATE table\_name SET column\_name = value
* Xóa bản ghi: DELETE FROM table\_name WHERE column\_name = value
  + 1. Ràng buộc trong MySQL
* [NOT NULL](https://vietjack.com/sql/rang_buoc_not_null_trong_sql.jsp): Bảo đảm một cột không thể có giá trị NULL.
* [DEFAULT](https://vietjack.com/sql/rang_buoc_default_trong_sql.jsp): Cung cấp giá trị mặc định cho cột
* [UNIQUE](https://vietjack.com/sql/rang_buoc_unique_trong_sql.jsp): Bảo đảm tất cả giá trị trong một cột là khác nhau.
* [PRIMARY Key](https://vietjack.com/sql/rang_buoc_primary_key_trong_sql.jsp): Mỗi bản ghi được nhận diện một cách duy nhất trong một bảng.
* [FOREIGN Key](https://vietjack.com/sql/rang_buoc_foreign_key_trong_sql.jsp): Mỗi bản ghi được nhận diện một cách duy nhất trong bất kỳ bảng nào.
* [CHECK](https://vietjack.com/sql/rang_buoc_check_trong_sql.jsp): Bảo đảm tất cả giá trị trong một cột thỏa mãn các điều kiện nào đó.
* [INDEX](https://vietjack.com/sql/rang_buoc_index_trong_sql.jsp): Sử dụng để tạo và lấy dữ liệu từ Database một cách nhanh chóng.
  + 1. Mệnh đề cơ bản
* WHERE được sử dụng để xác định một điều kiện trong khi lấy dữ liệu từ bảng đơn hoặc nhiều bảng kết hợp.
* AND và OR được sử dụng để kết hợp nhiều điều kiện để thu hẹp phạm vi dữ liệu trong một lệnh SQL. Hai loại toán tử này được gọi là toán tử liên hợp.
* LIKE được sử dụng để so sánh một giá trị với các giá trị tương tự bởi sử dụng các toán tử wildcard(%, \_)
* TOP được sử dụng để lấy dữ liệu của TOP N số hoặc X phần trăm bản ghi từ một bảng.
* GROUP BY được sử dụng để nhóm các giá trị giống nhau từ một cột.
* HAVING dùng để xác định các điều kiện để lọc nhóm kết quả nào sẽ xuất hiện trong kết quả cuối cùng.
* BETWEEN để thay thế một tổ hợp các điều kiện "lớn hơn hoặc bằng AND nhỏ hơn hoặc bằng".
  + 1. Các hàm cơ bản trong SQL
* COUNT được sử dụng để đếm số hàng trong một bảng dữ liệu.
* MAX: lựa chọn giá trị lớn nhất (tối đa) trong một cột cụ thể.
* MIN: lựa chọn giá trị nhỏ nhất (tối thiểu) trong một cột cụ thể.
* AVG: lựa chọn giá trị trung bình trong một cột cụ thể.
* SUM: lựa chọn giá trị tổng trong một cột cụ thể.
* CONCAT: nối bất kỳ chuỗi nào trong bất kỳ lệnh MySQL nào.
* CURTIME, NOW: trả về thời gian hiện tại
* DATEDIFF: Trừ hai date cho nhau
* LOWER: trả về chuỗi ký tự in thường.
* CHAR\_LENGTH: Trả về số ký tự của tham số
* LENGTH: Trả về độ dài (số byte) của một chuỗi
* REPLACE: Thay thế một chuỗi đã cho nếu xuất hiện
* TRIM: Gỡ bỏ Leading và Trailing space

# **CHƯƠNG II: ỨNG DỤNG XÂY DỰNG HỆ THỐNG ĐĂNG KÝ TRỰC TUYẾN VÀ QUẢN LÝ KÝ TÚC XÁ**

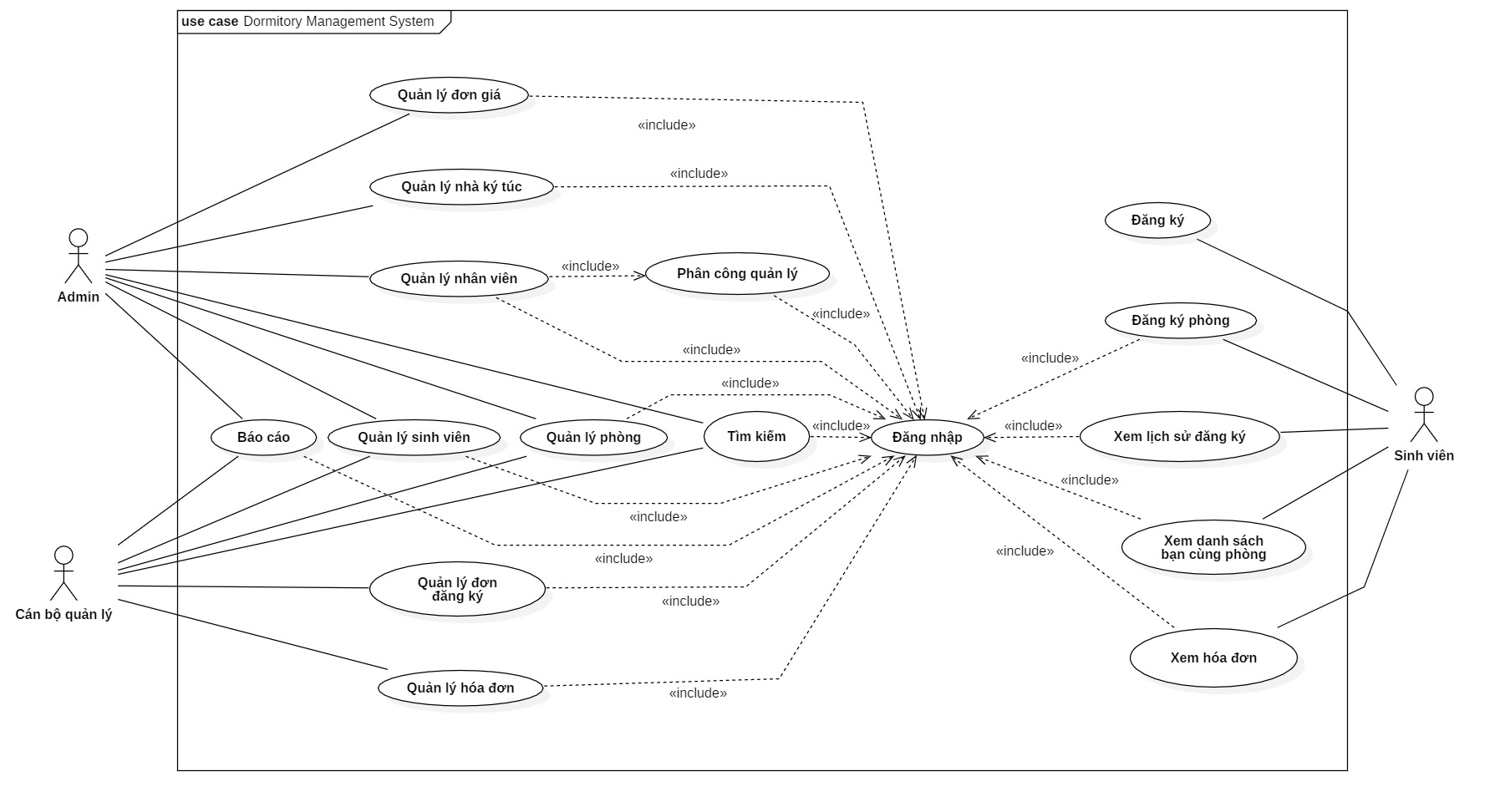
Trong chương này chúng ta sẽ phân tích đề tài để hiểu rõ yêu cầu của hệ thống cần xây dựng. Phân tích các chức năng của hệ thống, xây dựng sơ đồ phân tích thiết kế và cách tổ chức dữ liệu của hệ thống.

Hệ thống đăng ký trực tuyến và quản lý ký túc xá chia thành 2 chức năng chính đó là đăng ký phòng lưu trú trực tuyến và quản lý hệ thống ký túc xá. Với 2 nhóm đối tượng người dùng là sinh viên, cán bộ quản lý ký túc.

* Hoạt động nghiệp vụ

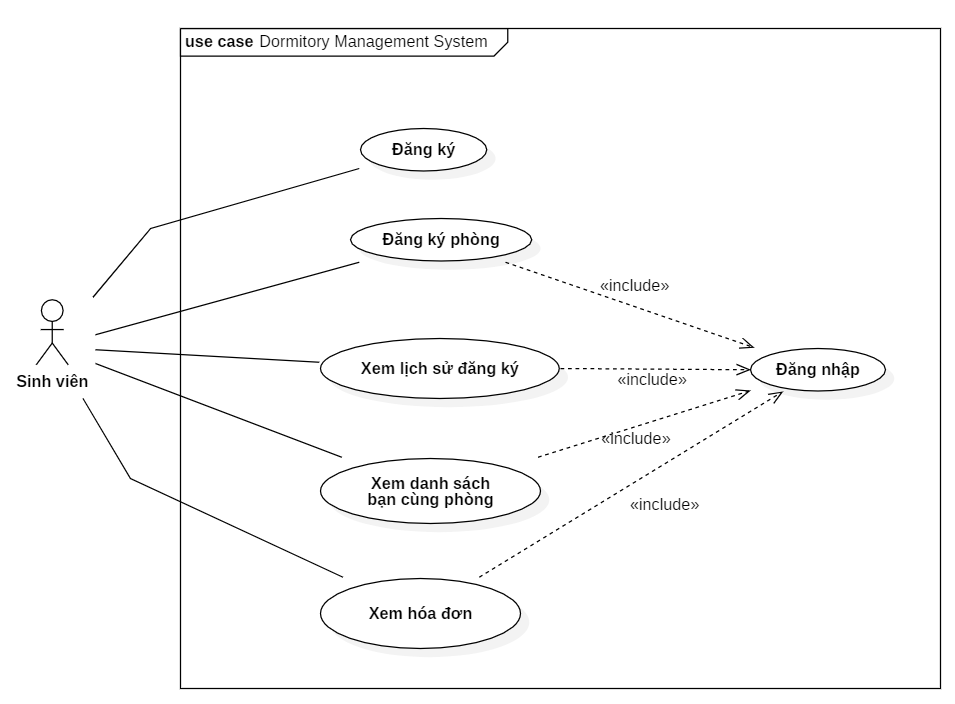
Các nghiệp vụ của hệ thống được tóm tắt như sau:

* Admin toàn hệ thống (Admin) quản lý thông tin các nhà ký túc, người quản lý (Manager), phân chia Manager cho từng nhà ký túc. Quản lý sinh viên đăng ký sử dụng hệ thống.
* Admin cập nhật bảng giá điện nước, phòng ở khi có thay đổi theo quy định chung của nhà trường hay cơ quan tổ chức có thẩm quyền.
* Mỗi Manager quản lý một nhà ký túc, quản lý thông tin, tình trạng của các phòng, sinh viên đăng ký lưu trú tại nhà thuộc quyền quản lý.
* Hàng tháng Manager cập nhật hóa đơn tiền điện nước của từng phòng, xuất hóa đơn khi sinh viên hoàn thành việc thanh toán. Xuất báo cáo lên Admin vào ngày 30 hàng tháng.
* Sinh viên đăng ký, đăng nhập hệ thống để có thể xem thông tin phòng, tình trạng phòng của ký túc xá để đăng ký lưu trú.
* Khi Manager cập nhật hóa đơn của phòng tương ứng, sinh viên có thể xem hóa đơn của phòng mình.
* Hàng tháng sinh viên thực hiện thanh toán phí điện nước với ban quản lý và sẽ nhận lại hóa đơn tương ứng.
  1. **Sơ đồ Use case tổng quát hệ thống**



Hình 2.1: Sơ đồ Use case tổng quát hệ thống

* 1. **Sơ đồ Usecase của đối tượng Sinh viên**



Hình 2.2: Sơ đồ Usecase đối tượng sinh viên

* + 1. Chức năng đăng nhập

|  |  |
| --- | --- |
| *Mô tả* | Chức năng đăng nhập giúp sinh viên truy cập vào hệ thống |
| *Kích hoạt* | Trang đăng nhập tự khởi động đầu tiên khi truy cập |
| *Đầu vào* | Thông tin tài khoản bao gồm Email, Mật khẩu |
| *Trình tự xử lý* | * Hiển thị form đăng nhập * Sinh viên nhập thông tin tài khoản * Click nút “Đăng nhập” trong form * Hiển thị trang chủ quản trị |
| *Lưu ý* | Dữ liệu nhập phải hợp lệ: không được bỏ trống |

Bảng 2.0.1: Bảng mô tả chức năng đăng nhập

* + 1. Chức năng đăng ký tài khoản

|  |  |
| --- | --- |
| *Mô tả* | Chức năng đăng ký giúp sinh viên tạo tài khoản để truy cập vào hệ thống |
| *Kích hoạt* | Chọn link “Đăng ký” trên trang đăng nhập |
| *Đầu vào* | Thông tin cá nhân bao gồm Họ tên, Mã sinh viên, Email, Mật khẩu, Số điện thoại, Địa chỉ, Ngày sinh, Giới tính, Dân tộc, Tôn giáo, Ảnh thẻ sinh viên. |
| *Trình tự xử lý* | * Hiển thị form đăng ký * Sinh viên nhập thông tin * Click nút “Đăng ký” trong form * Hiển thị trang đăng nhập và thông báo tương ứng |
| *Lưu ý* | Dữ liệu nhập phải hợp lệ: không được bỏ trống và nhập trùng dữ liệu |

Bảng 2.2: Bảng mô tả chức năng đăng ký

* + 1. Chức năng đăng ký phòng trực tuyến

|  |  |
| --- | --- |
| *Mô tả* | Chức năng đăng ký phòng cho phép sinh viên xem tình trạng các phòng của các nhà ký túc và đăng ký ngay trên website. |
| *Kích hoạt* | Chọn menu “Register” trong mục Registration |
| *Đầu vào* | Chọn nhà, tầng ký túc để xem danh sách phòng, lựa chọn phòng muốn đăng ký. |
| *Trình tự xử lý* | * Hiển thị form chọn tên nhà, tầng của ký túc xá * Sinh viên chọn tên nhà, tầng của ký túc xá * Hệ thống hiển thị danh sách các phòng phù hợp với giới tính của sinh viên. * Sinh viên chọn phòng để đăng ký ngay |
| *Lưu ý* | Dữ liệu nhập phải hợp lệ: không được bỏ trống |

Bảng 2.3: Bảng mô tả chức năng đăng ký phòng trực tuyến

* + 1. Chức năng xem lịch sử đăng ký phòng

|  |  |
| --- | --- |
| *Mô tả* | Chức năng xem lịch sử đăng ký phòng cho phép sinh viên theo dõi tình trạng của đơn đăng ký phòng vừa gửi, lịch sử đăng ký các kỳ học trước. |
| *Kích hoạt* | Chọn menu “History” trong mục Registration |
| *Đầu vào* | Menu “History” |
| *Trình tự xử lý* | * Hiển thị danh sách các đơn đăng ký và tình trạng của đơn. |

Bảng 2.4: Bảng mô tả chức năng xem lịch sử đăng ký phòng

* + 1. Chức năng xem danh sách bạn cùng phòng

|  |  |
| --- | --- |
| *Mô tả* | Chức năng xem danh sách bạn cùng phòng cho phép sinh viên xem danh sách thông tin liên lạc của bạn cùng phòng trong kỳ hiện tại. |
| *Kích hoạt* | Chọn menu “View Roommates” trong mục Registration |
| *Đầu vào* | Menu “View Roommates” |
| *Trình tự xử lý* | * Hiển thị danh sách thông tin liên lạc của bạn cùng phòng |

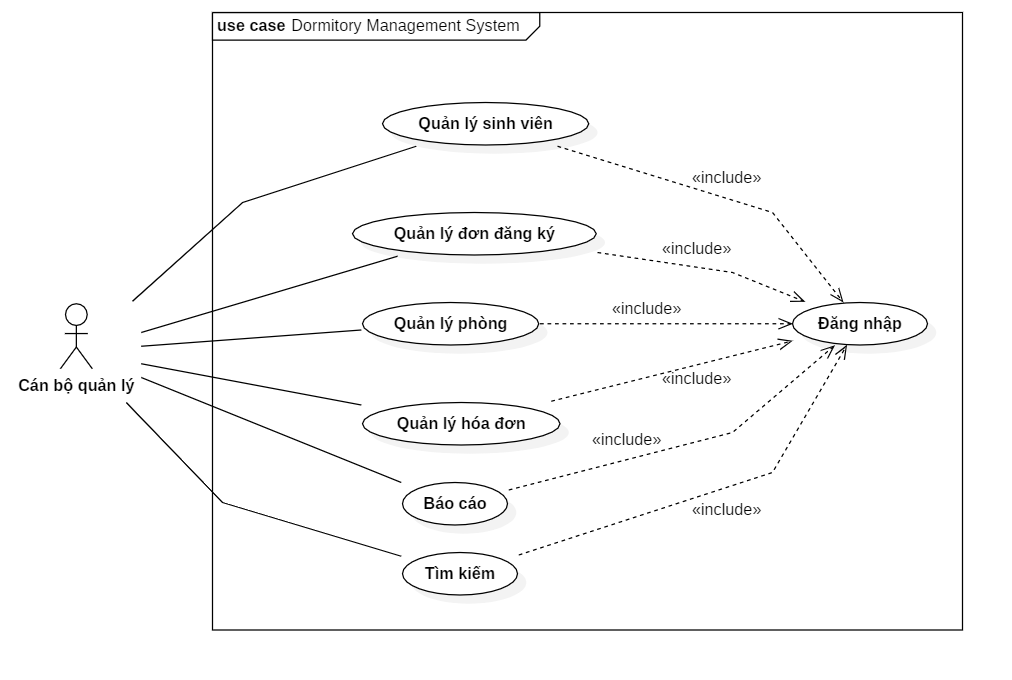
Bảng 2.5: Bảng mô tả chức năng xem danh sách bạn cùng phòng

* + 1. Chức năng xem hóa đơn

|  |  |
| --- | --- |
| *Mô tả* | Chức năng xem hóa đơn cho phép sinh viên xem danh sách hóa đơn tiền điện, nước theo tháng và tiền phòng theo kỳ học. |
| *Kích hoạt* | Chọn menu “Bills” và chọn loại hóa đơn muốn xem. |
| *Đầu vào* | Menu loại hóa đơn tương ứng |
| *Trình tự xử lý* | * Sinh viên chọn menu loại hóa đơn * Hệ thống hiển thị danh sách hóa đơn tương ứng |
| *Lưu ý* |  |

Bảng 2.6: Bảng mô tả chức năng xem hóa đơn

* 1. **Sơ đồ Usecase của đối tượng Cán bộ quản lý**



Hình 2.3: Sơ đồ Usecase đối tượng cán bộ quản lý

* + 1. Chức năng quản lý sinh viên

|  |  |
| --- | --- |
| *Mô tả* | Chức năng quản lý sinh viên cho phép Manager xem danh sách, thông tin đăng ký của sinh viên. Cập nhật tình trạng tài khoản của sinh viên. |
| *Kích hoạt* | Chọn menu “Students/ Students List”. |
| *Đầu vào* | * Xem thông tin sinh viên: chọn “View Profile” * Cập nhật tình trạng tài khoản: chọn “Verify” hoặc “Disable” |
| *Trình tự xử lý* | * Hiển thị danh sách sinh viên * Click button “View Profile” để chuyển sang trang thông tin sinh viên. * Click button trong cột Status để cập nhật tình trạng tài khoản sinh viên |

Bảng 2.7: Bảng mô tả chức năng quản lý sinh viên

* + 1. Chức năng quản lý đơn đăng ký phòng

|  |  |
| --- | --- |
| *Mô tả* | Chức năng quản lý đơn đăng ký phòng cho phép Manager xem danh sách các đơn đăng ký của sinh viên. Phê duyệt và cập nhật tình trạng của đơn đăng ký. |
| *Kích hoạt* | Chọn menu “Students/ Registration Form”. |
| *Đầu vào* | * Duyệt đơn đăng ký: chọn “Confirm” * Cập nhật tình trạng tài khoản: chọn “Disable” |
| *Trình tự xử lý* | * Hiển thị danh sách đơn đăng ký phòng * Click button “Confirm” để duyệt đơn đăng ký. * Click button “Disable” để cập nhật tình trạng tài khoản sinh viên |

Bảng 2.8: Bảng mô tả chức năng quản lý đơn đăng ký phòng

* + 1. Chức năng quản lý phòng ký túc

|  |  |
| --- | --- |
| *Mô tả* | Chức năng quản lý phòng cho phép Manager xem danh sách các phòng của nhà ký túc mình quản lý. Phê duyệt và cập nhật tình trạng của đơn đăng ký. |
| *Kích hoạt* | Chọn menu “Rooms”. |
| *Đầu vào* | * Duyệt đơn đăng ký: chọn “Confirm” * Cập nhật tình trạng tài khoản: chọn “Disable” |
| *Trình tự xử lý* | * Hiển thị danh sách đơn đăng ký phòng * Click button “Confirm” để duyệt đơn đăng ký. * Click button “Disable” để cập nhật tình trạng tài khoản sinh viên |

Bảng 2.9: Bảng mô tả chức năng quản lý phòng ký túc

* + 1. Chức năng xem hóa đơn

|  |  |
| --- | --- |
| *Mô tả* | Chức năng quản lý hóa đơn cho phép Manager xem danh sách các hóa đơn tiền điện, nước, phòng của nhà ký túc. |
| *Kích hoạt* | Chọn menu “Bills”, chọn loại hóa đơn. |
| *Đầu vào* | * Hóa đơn điện: chọn “Electricity” * Hóa đơn nước: chọn “Water” * Hóa đơn tiền phòng: “Room” |
| *Trình tự xử lý* | * Hiển thị danh sách hóa đơn của tất cả các phòng |

Bảng 2.10: Bảng mô tả chức năng quản lý hóa đơn

* + 1. Chức năng tạo hóa đơn

|  |  |
| --- | --- |
| *Mô tả* | Chức năng tạo hóa đơn cho phép Manager tạo các hóa đơn tiền điện, nước, phòng của nhà ký túc. |
| *Kích hoạt* | Chọn menu “Bills”, chọn loại hóa đơn, chọn tab “Add Bill” |
| *Đầu vào* | * Chọn tên phòng cần tạo hóa đơn * Chọn tháng của hóa đơn * Chỉ số tiêu thụ trong tháng |
| *Trình tự xử lý* | * Manager nhập thông tin hóa đơn * Hệ thống tính toán tổng số tiền của hóa đơn * Quay về trang danh sách hóa đơn đã được cập nhật |
| *Lưu ý* | Dữ liệu nhập phải hợp lệ: không được bỏ trống |

Bảng 2.11: Bảng mô tả chức năng tạo hóa đơn

* + 1. Chức năng cập nhật hóa đơn

|  |  |
| --- | --- |
| *Mô tả* | Chức năng cập nhật hóa đơn cho phép Manager cập nhật tình trạng các hóa đơn tiền điện, nước, phòng của nhà ký túc. |
| *Kích hoạt* | Chọn menu “Bills”, chọn loại hóa đơn |
| *Đầu vào* | * Chọn “Đã thanh toán” khi sinh viên thanh toán hóa đơn * Chọn “Disable” khi hóa đơn hết hạn vào cuối tháng. |
| *Trình tự xử lý* | * Hệ thống tự động cập nhật tình trạng, ngày thanh toán khi manager chọn button tương ứng. |

Bảng 2.12: Bảng mô tả chức năng cập nhật hóa đơn

* + 1. Chức năng xuất hóa đơn

|  |  |
| --- | --- |
| *Mô tả* | Chức năng xuất hóa đơn cho phép Manager xuất các hóa đơn tiền điện, nước, phòng của nhà ký túc. |
| *Kích hoạt* | Chọn menu “Bills”, chọn loại hóa đơn |
| *Đầu vào* | * Chọn icon để xuất hóa đơn đã thanh toán. |
| *Trình tự xử lý* | * Hệ thống tự động xuất hóa đơn dưới dạng file .docx |

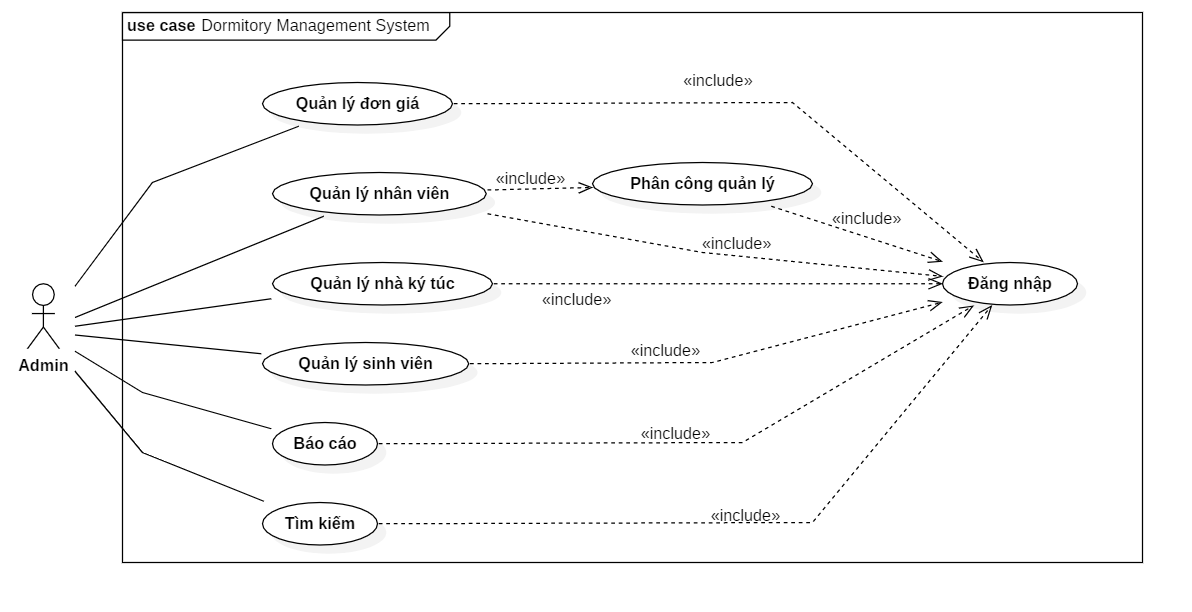
Bảng 2.13: Bảng mô tả chức năng xuất hóa đơn

* + 1. Chức năng gửi báo cáo

|  |  |
| --- | --- |
| *Mô tả* | Chức năng gửi báo cáo cho phép Manager gửi báo cáo của nhà ký túc mỗi tháng lên Admin hệ thống. |
| *Kích hoạt* | Chọn menu “Reports” |
| *Đầu vào* | * Chọn “Send Report” để gửi báo cáo theo tháng. |
| *Trình tự xử lý* | * Hệ thống tự động gửi báo cáo lên Admin và hiển thị danh sách các báo cáo của nhà ký túc. |

Bảng 2.14: Bảng mô tả chức năng gửi báo cáo

* 1. **Sơ đồ Usecase của đối tượng Administrator**



Hình 2.4: Sơ đồ Usecase của đối tượng Administrator

* + 1. Chức năng quản lý đơn giá

|  |  |
| --- | --- |
| *Mô tả* | Chức năng quản lý đơn giá cho phép Admin thêm, sửa đơn giá phòng, giá điện nước. |
| *Kích hoạt* | Chọn menu “Price List”, chọn loại đơn giá. |
| *Đầu vào* | * Tên đơn giá, mô tả và giá để thêm hoặc sửa đơn giá. |
| *Trình tự xử lý* | * Nhập tên đơn giá, mô tả và giá. * Chọn “Save” để lưu đơn giá mới. |
| *Lưu ý* | Dữ liệu nhập phải hợp lệ: không được bỏ trống |

Bảng 2.15: Bảng mô tả chức năng quản lý đơn giá

* + 1. Chức năng quản lý nhà ký túc xá

|  |  |
| --- | --- |
| *Mô tả* | Chức năng quản lý nhà ký túc cho phép Admin xem, thêm, sửa các nhà ký túc. |
| *Kích hoạt* | Chọn menu “Dormitory Facility/Buildings”. |
| *Đầu vào* | * Thông tin nhà ký túc: Name, Address, Image |
| *Trình tự xử lý* | * Nhập thông tin nhà ký túc, chọn “Save” để thêm hoặc sửa. * Chọn “View” để chuyển trang chi tiết thông tin |
| *Lưu ý* | Dữ liệu nhập phải hợp lệ: không được bỏ trống |

Bảng 2.0.36: Bảng mô tả chức năng quản lý nhà ký túc xá

* + 1. Chức năng quản lý phòng ký túc xá

|  |  |
| --- | --- |
| *Mô tả* | Chức năng quản lý phòng ký túc cho phép Admin xem, thêm, sửa các phòng ký túc. |
| *Kích hoạt* | Chọn menu “Dormitory Facility/Rooms”. |
| *Đầu vào* | * Thông tin phòng ký túc: Building, Floor, Room name, Room type |
| *Trình tự xử lý* | * Nhập thông tin phòng, chọn “Save” để thêm hoặc sửa. * Chọn “View Students” để chuyển trang xem danh sách sinh viên tại phòng. |
| *Lưu ý* | Dữ liệu nhập phải hợp lệ: không được bỏ trống |

Bảng 2.17: Bảng mô tả chức năng quản lý phòng ký túc xá

* + 1. Chức năng quản lý nhân viên

|  |  |
| --- | --- |
| *Mô tả* | Chức năng quản lý nhân viên cho phép Admin xem, cập nhật trạng thái tài khoản của nhân viên quản lý |
| *Kích hoạt* | Chọn menu “Officers”. |
| *Đầu vào* | * Các button trạng thái “Active”, “Deactive”, “Delete” |
| *Trình tự xử lý* | * Click các button trạng thái “Active”, “Deactive”, “Delete” * Hệ thống tự cập nhật trạng thái |

Bảng 2.18: Bảng mô tả chức năng quản lý nhân viên

* + 1. Chức năng phân công quản lý nhà ký túc

|  |  |
| --- | --- |
| *Mô tả* | Chức năng phân công quản lý nhà ký túc cho phép Admin phân công người quản lí cho các nhà ký túc. |
| *Kích hoạt* | Chọn menu “Officers”. Chọn tab “Assign to building” |
| *Đầu vào* | * Manager, Building, Position |
| *Trình tự xử lý* | * Admin chọn người quản lý, nhà, vị trí * Chọn “Save” để hoàn thành form phân công vị trí |
| *Lưu ý* | Dữ liệu nhập phải hợp lệ: không được bỏ trống |

Bảng 2.19: Bảng mô tả chức năng phân công quản lý nhà ký túc

* + 1. Chức năng xem báo cáo hàng tháng

|  |  |
| --- | --- |
| *Mô tả* | Chức năng xem báo cáo hàng tháng cho phép Admin theo dõi báo cáo của các nhà ký túc theo tháng. |
| *Kích hoạt* | Chọn menu “Reports” |
| *Đầu vào* | * Xem báo cáo chi tiết: Chọn “View Report” |
| *Trình tự xử lý* | * Chọn “View Report” để chuyển sang trang báo cáo chi tiết |

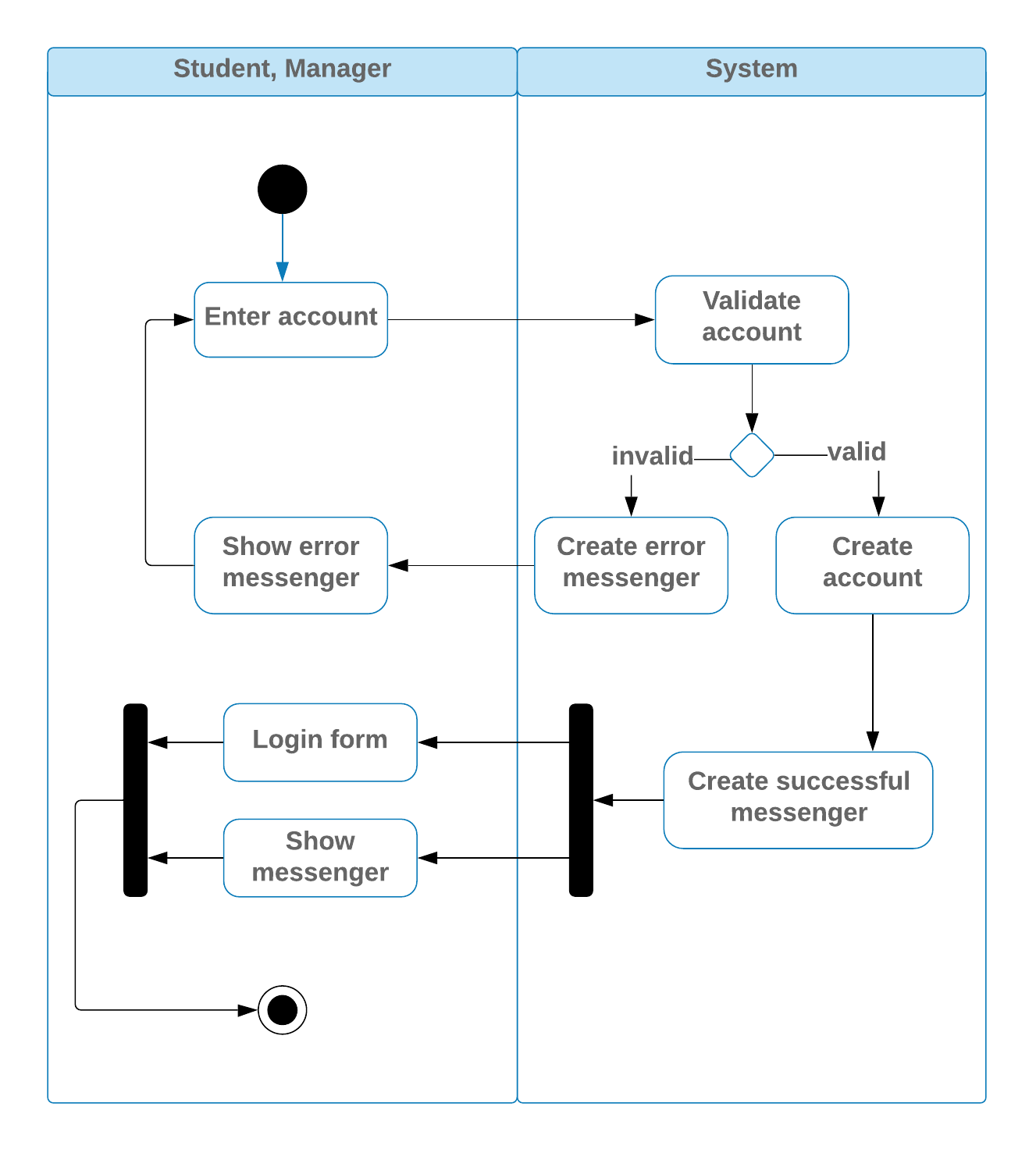
Bảng 2.0.40: Bảng mô tả chức năng xem báo cáo hàng tháng

* + 1. Chức năng xuất báo cáo hàng tháng

|  |  |
| --- | --- |
| *Mô tả* | Chức năng xuất báo cáo hàng tháng cho phép Admin xuất danh sách báo cáo của các nhà ký túc ra file Excel |
| *Kích hoạt* | Chọn menu “Reports”. |
| *Đầu vào* | * Click icon để xuất báo cáo |
| *Trình tự xử lý* | * Hệ thống tự động xuất và tải xuống báo cáo dưới dạng file excel |

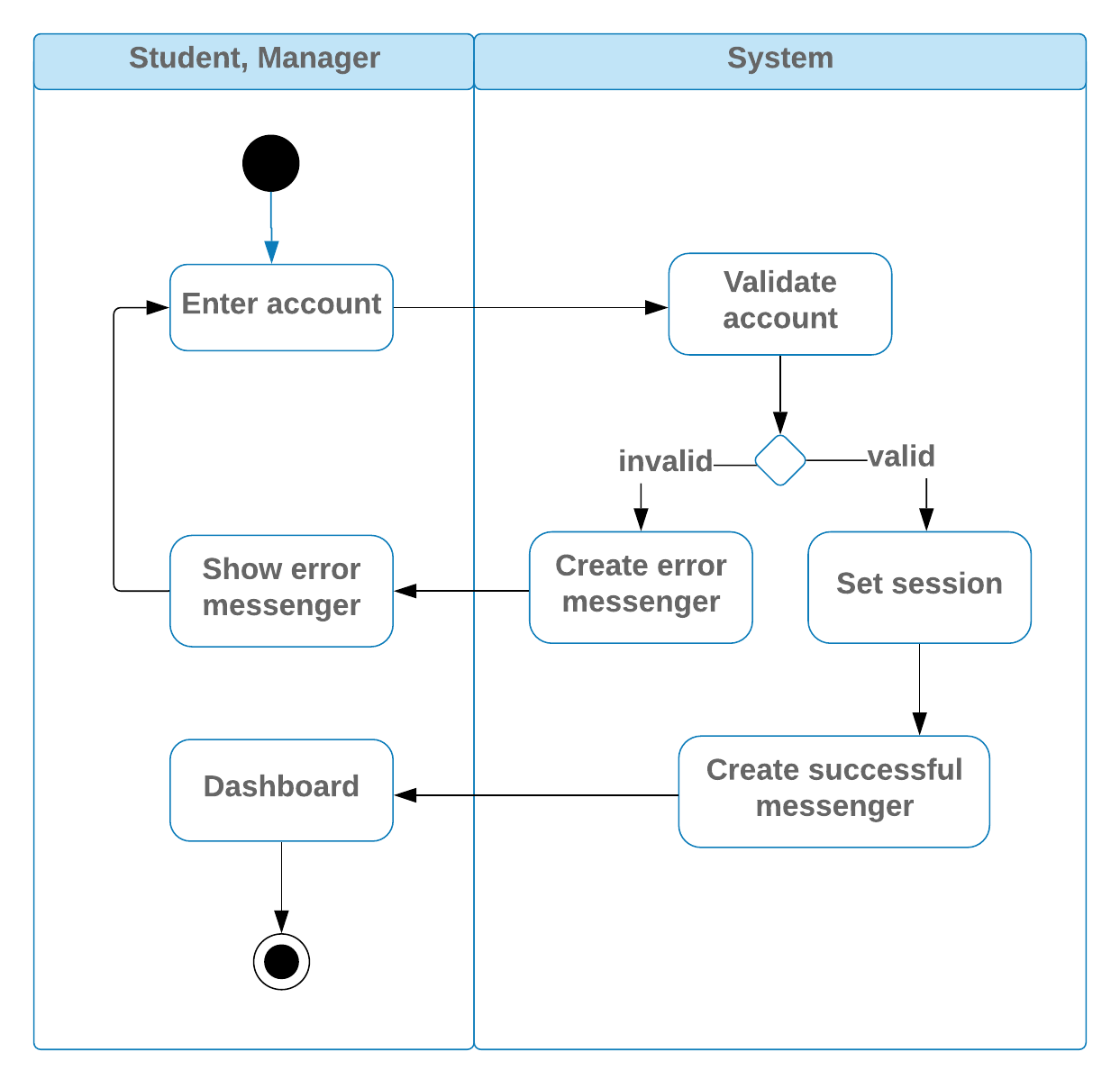
Bảng 2.21: Bảng mô tả chức năng xuất báo cáo hàng tháng

* 1. **Biểu đồ hoạt động của hệ thống**
     1. Chức năng đăng ký



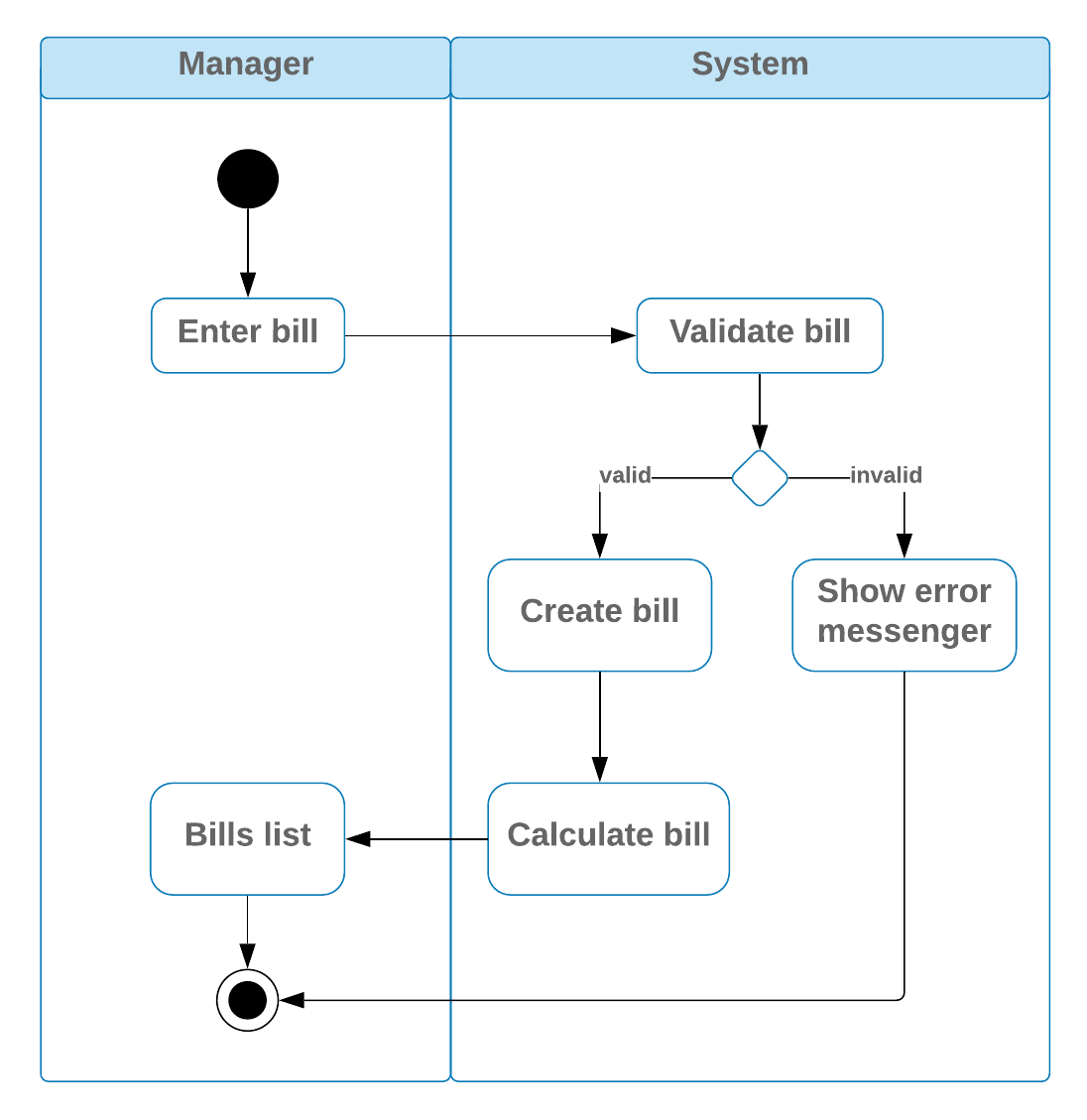
Hình 2.5: Biểu đồ hoạt động chức năng đăng ký

* + 1. Chức năng đăng nhập



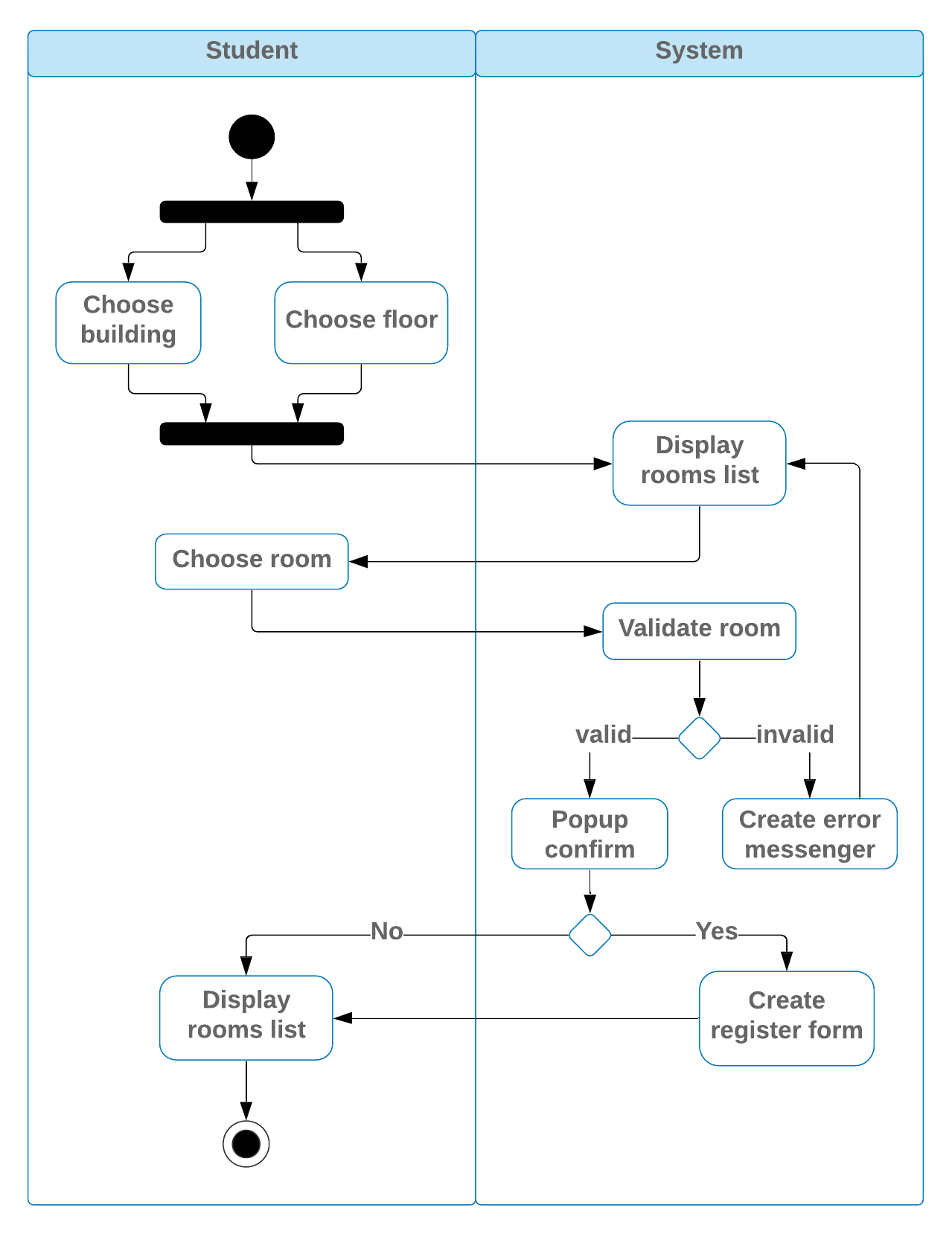
Hình 2.6: Biểu đồ hoạt động chức năng đăng nhập

* + 1. Chức năng tạo hóa đơn



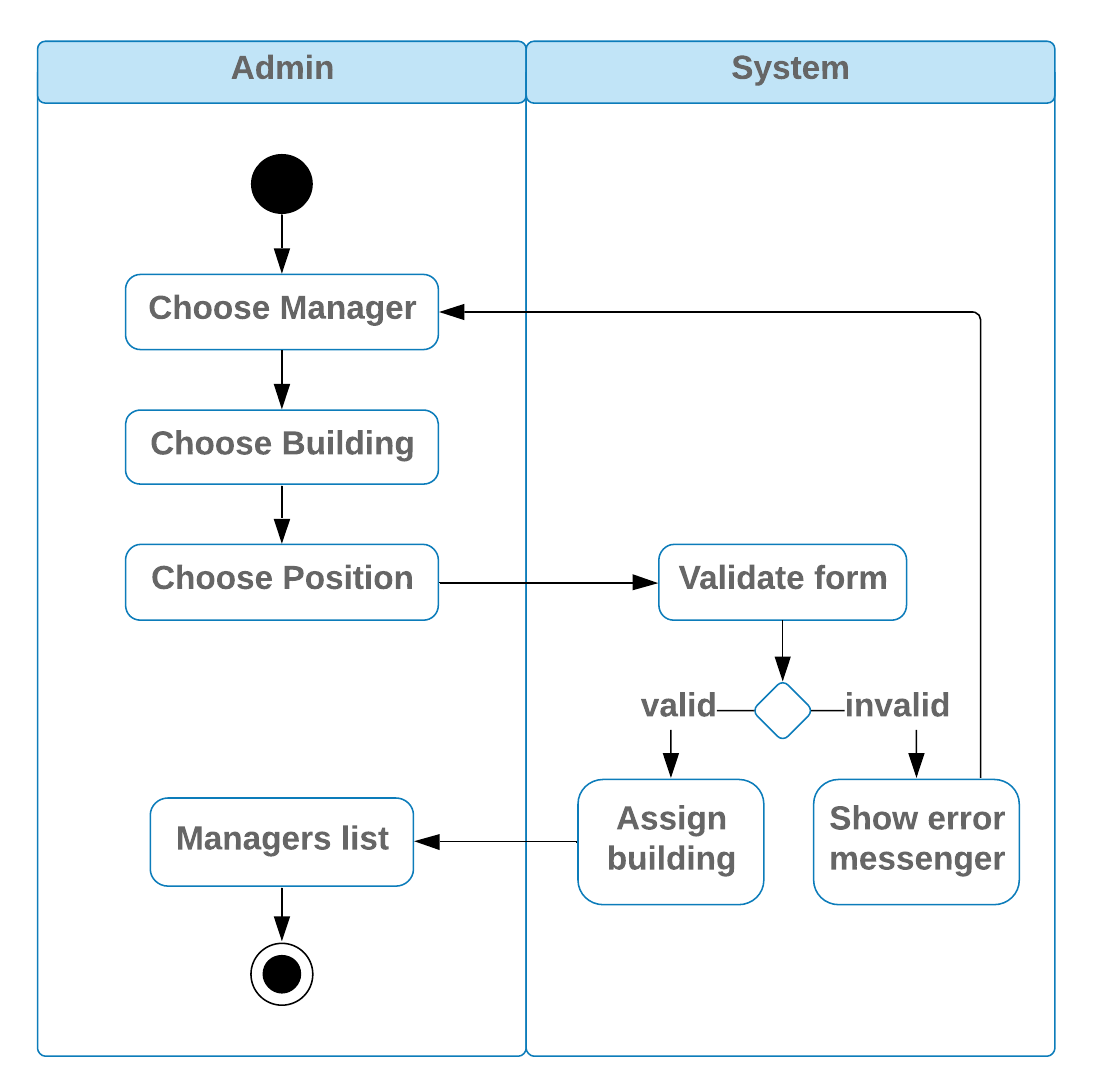
Hình 2.7: Biểu đồ hoạt động chức năng tạo hóa đơn

* + 1. Chức năng đăng ký phòng



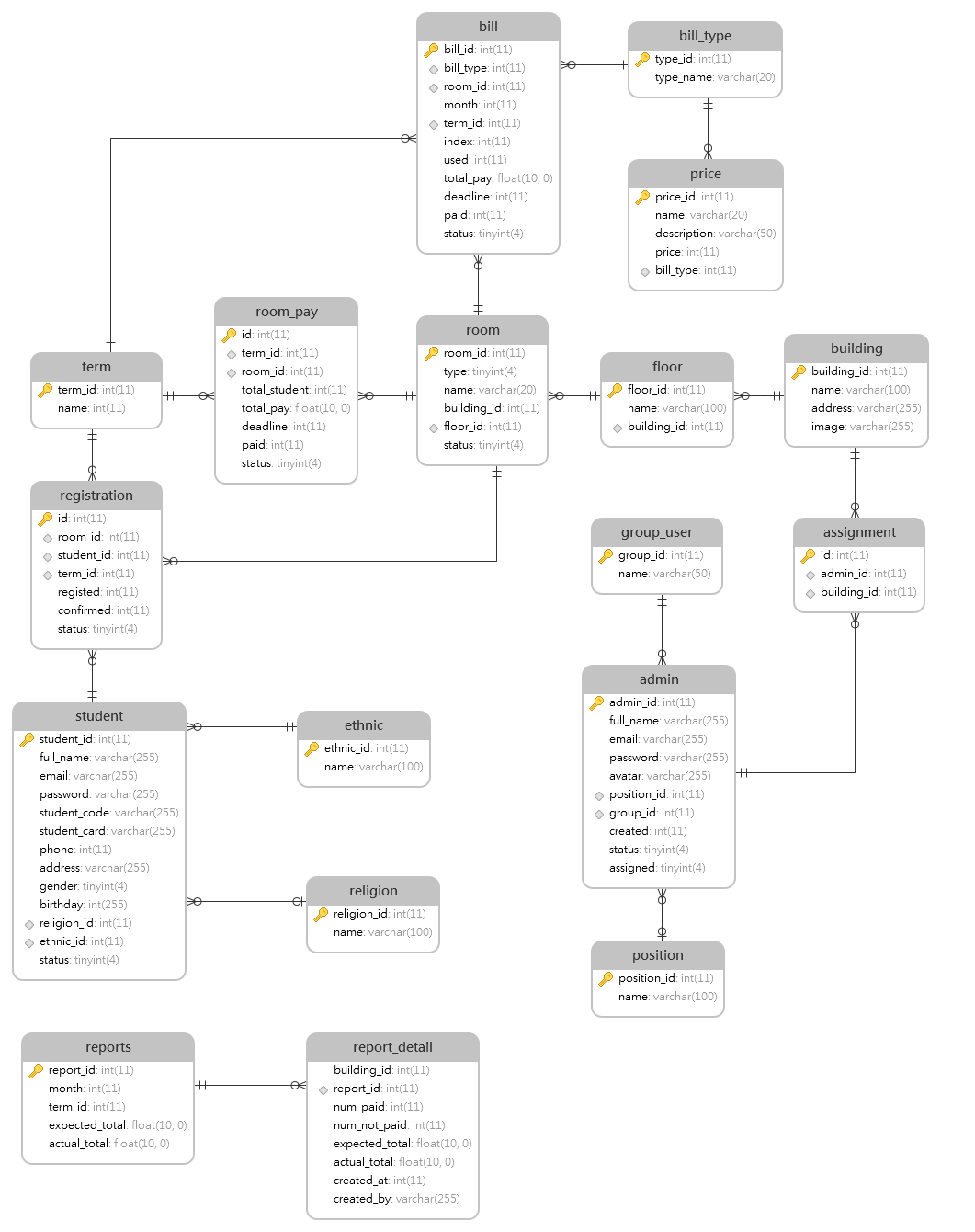
Hình 2.8: Biểu đồ hoạt động chức năng đăng ký phòng

* + 1. Chức năng phân công quản lý



Hình 2.9: Biểu đồ hoạt động chức năng phân công quản lý

* 1. **Sơ đồ thực thể liên kết (ERD)**



Hình 2.10: Sơ đồ thực thể liên kết (ERD)

* 1. **Các bảng dữ liệu của hệ thống**
     1. Bảng “admin”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | admin\_id | int(11) | Không | Mã admin |
|  | full\_name | varchar(255) | Không | Tên đầy đủ |
|  | email | varchar(255) | Không | Email |
|  | password | varchar(255) | Không | Mật khẩu |
|  | avatar | varchar(255) | Có | Ảnh đại diện |
| FK | position\_id | int(11) | Có | Vị trí |
| FK | group\_id | int(11) | Không | Nhóm |
|  | created\_at | int(11) | Không | Ngày tạo |
|  | status | tinyint(4) | Không | Trạng thái |
|  | assigned | tinyint(4) | Không | Trạng thái phân công |

Bảng 2.22: Bảng admin

* + 1. Bảng “student”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | student\_id | int(11) | Không | Số thứ tự |
|  | full\_name | varchar(255) | Không | Tên đầy đủ |
|  | email | varchar(255) | Không | Email |
|  | password | varchar(255) | Không | Mật khẩu |
|  | student\_code | varchar(255) | Không | Mã sinh viên |
|  | student\_card | varchar(255) | Không | Thẻ sinh viên |
|  | phone | int(11) | Không | Số điện thoại |
|  | address | varchar(255) | Không | Địa chỉ |
|  | gender | tinyint(4) | Không | Giới tính |
|  | birthday | int(11) | Không | Ngày sinh |
| FK | religion\_id | int(11) | Có | Tôn giáo |
| FK | ethnic\_id | int(11) | Không | Dân tộc |
|  | status | tinyint(4) | Không | Trạng thái |

Bảng 2.23: Bảng student

* + 1. Bảng “building”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | building\_id | int(11) | Không | Số thứ tự |
|  | name | varchar(100) | Không | Tên nhà |
|  | address | varchar(255) | Không | Địa chỉ |
|  | image | varchar(255) | Không | Ảnh |

Bảng 2.24: Bảng building

* + 1. Bảng “floor”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | floor\_id | int(11) | Không | Số thứ tự |
|  | name | varchar(100) | Không | Tên tầng |
| FK | building\_id | int(11) | Không | Số thứ tự nhà |

Bảng 2.25: Bảng floor

* + 1. Bảng “room”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | room\_id | int(11) | Không | Số thứ tự |
|  | type | tinyint(4) | Không | Loại phòng |
|  | name | varchar(20) | Không | Tên phòng |
| FK | building\_id | int(11) | Không | Số thứ tự nhà |
| FK | floor\_id | int(11) | Không | Số thứ tự tầng |
|  | status | tinyint(4) | Không | Trạng thái |

Bảng 2.26: Bảng room

* + 1. Bảng “term”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | term\_id | int(11) | Không | Số thứ tự kỳ |
|  | name | int(11) | Không | Tên kỳ |

Bảng 2.27: Bảng term

* + 1. Bảng “bill\_type”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | type\_id | int(11) | Không | Số thứ tự |
|  | type\_name | varchar(20) | Không | Tên loại hóa đơn |

Bảng 2.28: Bảng bill\_type

* + 1. Bảng “price”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | price\_id | int(11) | Không | Số thứ tự |
|  | name | varchar(20) | Không | Tên giá |
|  | description | varchar(50) | Không | Mô tả |
|  | price | int(11) | Không | Giá |
|  | bill\_type | int(11) | Không | Loại hóa đơn |

Bảng 2.29: Bảng price

* + 1. Bảng “bill”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | bill\_id | int(11) | Không | Số thứ tự |
|  | bill\_type | int(11) | Không | Loại hóa đơn |
|  | room\_id | int(11) | Không | Mã số phòng |
|  | month | int(11) | Không | Tháng |
|  | term\_id | int(11) | Không | Mã số kỳ |
|  | index | int(11) | Không | Chỉ số tháng |
|  | used | int(11) | Không | Tiêu thụ |
|  | total\_pay | float(10) | Không | Tổng tiền |
|  | deadline | int(11) | Không | Hạn thanh toán |
|  | paid | int(11) | Có | Ngày thanh toán |
|  | status | tinyint(4) | Không | Trạng thái |

Bảng 2.0.50: Bảng bill

* + 1. Bảng “room\_pay”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | id | int(11) | Không | Số thứ tự |
|  | term\_id | int(11) | Không | Mã số kỳ |
|  | room\_id | int(11) | Không | Mã số phòng |
|  | total\_student | int(11) | Không | Tổng số sinh viên |
|  | total\_pay | float(10) | Không | Tổng tiền |
|  | deadline | int(11) | Không | Hạn thanh toán |
|  | paid | int(11) | Không | Ngày thanh toán |
|  | status | tinyint(4) | Không | Trạng thái |

Bảng 2.31: Bảng room\_pay

* + 1. Bảng “assignment”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | id | int(11) | Không | Số thứ tự |
|  | admin\_id | int(11) | Không | Mã số admin |
|  | building\_id | int(11) | Không | Mã số nhà |

Bảng 2.32: Bảng assignment

* + 1. Bảng “registration”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | id | int(11) | Không | Số thứ tự |
|  | room\_id | int(11) | Không | Mã số phòng |
|  | student\_id | int(11) | Không | Mã số sinh viên |
|  | term\_id | int(11) | Không | Mã số kỳ |
|  | registed | int(11) | Không | Ngày đăng ký |
|  | confirmed | int(11) | Có | Ngày xác nhận |
|  | status | tinyint(4) | Có | Trạng thái |

Bảng 2.33: Bảng registration

* + 1. Bảng “position”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | position\_id | int(11) | Không | Số thứ tự |
|  | name | varchar(100) | Không | Tên vị trí |

Bảng 2.34: Bảng position

* + 1. Bảng “religion”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | religion\_id | int(11) | Không | Số thứ tự |
|  | name | varchar(100) | Không | Tên tôn giáo |

Bảng 2.35: Bảng religion

* + 1. Bảng “ethnic”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | ethnic\_id | int(11) | Không | Số thứ tự |
|  | name | varchar(100) | Không | Tên dân tộc |

Bảng 2.36: Bảng ethnic

* + 1. Bảng “group\_user”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khóa | Tên cột | Kiểu | Null | Mô tả |
| PK | group\_id | int(11) | Không | Số thứ tự |
|  | name | varchar(100) | Không | Tên nhóm |

Bảng 2.37: Bảng group\_user

# **CHƯƠNG III: KẾT LUẬN VÀ ĐÁNH GIÁ**

* 1. **Kết quả đạt được**

Sau quá trình tìm hiểu và xây dựng, phần mềm quản lý sách cho sinh viên Bkacad đã tương đối đầy đủ yêu cầu chức năng đã đề ra. Giao diện đơn giản dễ sử dụng.

Hiểu thêm về phân tích thiết kế hệ thống, ngôn ngữ lập trình hướng đối tượng, làm quen với framework Laravel và một số thư viện hỗ trợ như Bootstrap, Jquery, kỹ thuật Ajax.

* 1. **Hạn chế**

Phần mềm còn những hạn chế như

* Chưa có chức năng đăng nhập bằng email, Facebook.
* Chưa thể gửi danh sách sinh viên chưa đăng ký lên các nhóm lớp trên Facebook.
* Giao diện thiết kế chưa chuyên nghiệp.
  1. **Hướng phát triển**
* Sử dụng API của Facebook để khắc phục những hạn chế trên.
* Mở rộng thống kê để có thể thống kê theo tháng, năm, học kỳ.
* Mở rộng quản lý thêm các khóa học, chương trình học tương ứng của Bkacad.
  1. **Phân công công việc**

|  |  |  |
| --- | --- | --- |
| **Công việc** | **Sinh viên thực hiện** | **Đánh giá** |
| Thiết kế giao diện | Nguyễn Thị Hồng Duyên | 100% |
| Thiết kế cơ sở dữ liệu | 100% |
| Lập trình | 90% |

# **PHỤ LỤC**

DANH MỤC HÌNH ẢNH

[Hình 1.1: Sơ đồ hoạt động của PHP 10](#_Toc12054860)

[Hình 1.2: Sơ đồ tổ chức dữ liệu trong CodeIgniter 14](#_Toc12054861)

[Hình 2.1: Sơ đồ Use case tổng quát hệ thống 21](#_Toc12054862)

[Hình 2.2: Sơ đồ Usecase đối tượng sinh viên 21](#_Toc12054863)

[Hình 2.3: Sơ đồ Usecase đối tượng cán bộ quản lý 26](#_Toc12054864)

[Hình 2.4: Sơ đồ Usecase của đối tượng Administrator 30](#_Toc12054865)

[Hình 2.5: Biểu đồ hoạt động chức năng đăng ký 34](#_Toc12054866)

[Hình 2.6: Biểu đồ hoạt động chức năng đăng nhập 34](#_Toc12054867)

[Hình 2.7: Biểu đồ hoạt động chức năng đăng ký phòng 35](#_Toc12054868)

[Hình 2.8: Biểu đồ hoạt động chức năng tạo hóa đơn 36](#_Toc12054869)

[Hình 2.9: Biểu đồ hoạt động chức năng phân công quản lý 36](#_Toc12054870)

[Hình 2.10: Sơ đồ thực thể liên kết (ERD) 37](#_Toc12054871)

DANH MỤC BẢNG BIỂU

# **TÀI LIỆU THAM KHẢO**

**Tiếng Anh:**

1. Website: “<https://www.php.net/manual/en/index.php>”
2. Website: “<https://www.codeigniter.com/docs>”
3. Website: “<https://www.tutorialspoint.com/uml>”

**Tiếng Việt:**

1. Website: “[https://viblo.asia](https://viblo.asia/)”
2. Website: “https://www.evn.com.vn”
3. Website: “<https://nuocsachhanoi.com/>”