**TRƯỜNG ĐẠI HỌC ĐỒNG THÁP**

**KHOA SƯ PHẠM TOÁN – TIN**

**-----oOo-----**



**BÁO CÁO BÀI TẬP LỚN MÔN HỌC  
PHÁT TRIỂN PHẦN MỀM HƯỚNG ĐỐI TƯỢNG**

**LỚP: ĐHCNTT18A – CHUYÊN NGÀNH: KHOA HỌC MÁY TÍNH**

**ĐỀ TÀI:**

**QUẢN LÝ ĐĂNG KÝ PHÒNG THỰC HÀNH**

**TRƯỜNG ĐẠI HỌC ĐỒNG THÁP**

**GVHD: ThS. Trần Kim Hương**

**Thành viên nhóm:**

**0018413046 - Nguyễn Thị Hồng Ngọc**

**0018413058 – Võ Thị Bích Trăm**

***Đồng Tháp, Ngày 24 Tháng 12 năm 2021***

# **LỜI NÓI ĐẦU**

Trong thời đại ngày nay, Công nghệ thông tin đang được phát triển mạnh mẽ, nhanh chóng và xâm nhập vào nhiều lĩnh vực khoa học, kỹ thuật cũng như trong cuộc sống. Nó trở thành công cụ đắc lực trong nhiều ngành nghề như giao thông, quân sự, y học,…và đặc biệt trong công tác quản lý nói chung và quản lý phòng máy nói riêng.

Trước đây khi máy tính chưa được sử dụng rộng rãi, các công việc quản lý được thực hiện thủ công bằng sổ sách dẫn đến việc lưu giữ rất cồng kềnh, gây tốn nhiều thời gian, công sức và thiếu độ chính xác. Do đó, công việc quản lý sử dụng phương pháp thủ công sẽ không đáp ứng được nhiệm vụ và mục tiêu của hệ thống và nó cũng gây ảnh hưởng rất lớn đến năng suất, chất lượng và hiệu quả công việc. Để khắc phục các nhược điểm trên, đồng thời nhờ vào sự phát triển nhanh chóng của Công nghệ thông tin thì xây dựng chương trình “Quản lý đăng ký phòng thực hành DThU” giúp cho công tác quản lý là một nhu cầu cần thiết hiện nay để tự động hoá việc quản lý phòng máy.

Trường Đại học Đồng Tháp là trường có số lượng phòng thực hành máy tính khá nhiều, là địa điểm học thực hành của sinh viên các khoa khác nhau trong trường, đặc biệt là sinh viên ngành công nghệ thông tin. Nhu cầu sử dụng phòng thực hành để giảng dạy của giảng viên mỗi ngày càng lớn, điều đó tạo nên sự phức tạp về mặt quản lý các phòng, chính vì vậy việc tin học hoá hệ thống phòng máy để có thể quản lý một cách hiệu quả hơn là một nhu cầu cấp thiết hiện nay.

Từ những lý do trên chúng em đã lựa chọn đề tài “Quản lý đăng ký phòng thực hành DThU” với mong muốn giúp cho việc quản lý được dễ dàng, thuận tiện và tránh sai sót.

Do thời gian có hạn và sự hiểu biết còn hạn chế nên chắc chắn bài làm không thể tránh khỏi những sai sót, rất mong được sự giúp đỡ và góp ý kiến của Cô. Chúng em chân thành cảm ơn!

**MỤC LỤC**

[**LỜI NÓI ĐẦU** 1](#_Toc91491289)

[**DANH MỤC HÌNH** 4](#_Toc91491290)

[**PHẦN MỘT: MỞ ĐẦU** 6](#_Toc91491291)

[**1.** **Đặt vấn đề** 6](#_Toc91491292)

[**2.** **Ý nghĩa thực tiễn, lý thuyết của đề tài** 6](#_Toc91491293)

[**3.** **Mục tiêu của đề tài** 7](#_Toc91491294)

[**4.** **Phạm vi của đề tài** 7](#_Toc91491295)

[**5.** **Các phương pháp nghiên cứu** 7](#_Toc91491296)

[**6.** **Kế hoạch thực hiện đề tài** 7](#_Toc91491297)

[**PHẦN HAI: NỘI DUNG** 8](#_Toc91491298)

[**CHƯƠNG 1: CƠ SỞ LÝ THUYẾT** 8](#_Toc91491299)

[**1.1** **Giới thiệu hệ quản trị SQL Server** 8](#_Toc91491300)

[**1.2** **Giới thiệu ngôn ngữ lập trình C#** 8](#_Toc91491301)

[**1.3** **Cơ bản về phân tích, thiết kế hệ thống hướng đối tượng và UML** 9](#_Toc91491302)

[**1.4** **Mô hình 3 lớp** 12](#_Toc91491303)

[**CHƯƠNG 2: PHÂN TÍCH PHẦN MỀM HƯỚNG ĐỐI TƯỢNG** 14](#_Toc91491304)

[**2.1** **Mô tả bài toán** 14](#_Toc91491305)

[**2.1.1** **Hiện trạng nghiệp vụ của bài toán** 14](#_Toc91491306)

[**2.1.2** **Mô tả hệ thống** 14](#_Toc91491307)

[**2.1.3** **Yêu cầu chức năng** 15](#_Toc91491308)

[**2.2** **Phân tích phần mềm hướng đối tượng** 15](#_Toc91491309)

[**2.2.1** **Sơ đồ Use case** 15](#_Toc91491310)

[**2.2.2** **Đặc tả Use case** 16](#_Toc91491311)

[**2.2.3** **Sơ đồ hoạt động** 23](#_Toc91491312)

[**2.2.4** **Sơ đồ lớp** 29](#_Toc91491313)

[**CHƯƠNG 3: XÂY DỰNG PHẦN MỀM HƯỚNG ĐỐI TƯỢNG** 30](#_Toc91491314)

[**3.1** **Cách tổ chức dữ liệu** 30](#_Toc91491315)

[**3.1.1** **Thiết kế dữ liệu vật lý – PMD** 30](#_Toc91491316)

[**3.1.2** **Các ràng buộc toàn vẹn trên dữ liệu** 33](#_Toc91491317)

[**3.1.3** **Diagram** 34](#_Toc91491318)

[**3.2 Công cụ sử dụng xây dựng demo** 34](#_Toc91491319)

[**3.3** **Một số giao diện chính của chương trình** 35](#_Toc91491320)

[**3.3.1** **Giao diện Đăng nhập** 35](#_Toc91491321)

[**3.3.2** **Giao diện Đăng ký tài khoản** 36](#_Toc91491322)

[**3.3.3** **Giao diện chính** 37](#_Toc91491323)

[**3.3.4** **Giao diện chức năng** 38](#_Toc91491324)

[**3.3.4.1** **Giao diện Đổi mật khẩu** 38](#_Toc91491325)

[**3.3.4.2** **Giao diện Tra cứu thông tin đăng ký Phòng thực hành** 39](#_Toc91491326)

[**3.3.4.3** **Giao diện Tra cứu thông tin máy tính hư hỏng** 40](#_Toc91491327)

[**3.3.4.4 Giao diện Quản lý cán bộ** 42](#_Toc91491328)

[**3.3.4.5** **Giao diện Quản lý lớp học phần** 43](#_Toc91491329)

[**3.3.4.6** **Giao diện Quản lý phòng thực hành** 45](#_Toc91491330)

[**3.3.4.7** **Giao diện Quản lý máy tính** 46](#_Toc91491331)

[**3.3.4.8** **Giao diện Quản lý đăng ký phòng thực hành** 47](#_Toc91491332)

[**3.3.4.9** **Giao diện Đăng ký phòng thực hành** 48](#_Toc91491333)

[**3.3.4.10** **Giao diện Phiếu thống kê tình trạng hư hỏng của máy tính** 50](#_Toc91491334)

[**3.3.4.11** **Giao diện Báo cáo danh sách đăng ký phòng thực hành** 51](#_Toc91491335)

[**PHẦN BA: KẾT LUẬN** 52](#_Toc91491336)

[**1.** **Kết quả đạt được** 52](#_Toc91491337)

[**2.** **Hạn chế** 53](#_Toc91491338)

[**3.** **Hướng phát triển** 53](#_Toc91491339)

[**TÀI LIỆU THAM KHẢO** 54](#_Toc91491340)

[**PHỤ LỤC** 55](#_Toc91491341)

[**1.** **Hướng dẫn sử dụng** 55](#_Toc91491342)

[**2.** **Bảng phân công công việc** 55](#_Toc91491343)

# **DANH MỤC HÌNH**

[Hình 2.1 Use case tổng quát 16](#_Toc91491381)

[Hình 2.2 Use case quản lý đăng ký 16](#_Toc91491382)

[Hình 2.3 Use case đăng ký phòng thực hành 17](#_Toc91491383)

[Hình 2.4 Use case quản lý phòng thực hành máy tính 18](#_Toc91491384)

[Hình 2.5 Use case quản lý cán bộ 19](#_Toc91491385)

[Hình 2.6 Use case quản lý lớp học phần 20](#_Toc91491386)

[Hình 2.7 Use case quản lý máy tính 21](#_Toc91491387)

[Hình 2.8 Use case báo cáo thống kê 22](#_Toc91491388)

[Hình 2.9 Use case chi tiết 23](#_Toc91491389)

[Hình 2.10 Hoạt động đăng ký mượn phòng thực hành 24](#_Toc91491390)

[Hình 2.11 Hoạt động trả phòng thực hành 25](#_Toc91491391)

[Hình 2.12 Hoạt động thống kê 26](#_Toc91491392)

[Hình 2.13 Hoạt động đăng nhập 27](#_Toc91491393)

[Hình 2.14 Hoạt động tạo tài khoản 28](#_Toc91491394)

[Hình 2.15 Sơ đồ lớp 29](#_Toc91491395)

[Hình 3.1 Diagram cơ sở dữ liệu 34](#_Toc91491396)

[Hình 3.2 Giao diện Đăng nhập 35](#_Toc91491397)

[Hình 3.3 Giao diện Đăng ký tài khoản 37](#_Toc91491398)

[Hình 3.4 Giao diện màn hình chính 38](#_Toc91491399)

[Hình 3.5 Giao diện Đổi mật khẩu 39](#_Toc91491400)

[Hình 3.6 Giao diện Tra cứu thông tin đăng ký phòng thực hành DThU 40](#_Toc91491401)

[Hình 3.7 Giao diện Xuất Excel danh sách đăng ký phòng thực hành 40](#_Toc91491402)

[Hình 3.8 Giao diện Tra cứu thông tin máy tính hư hỏng 41](#_Toc91491403)

[Hình 3.9 Giao diện Xuất Excel danh sách tình trạng máy tính hư hỏng 41](#_Toc91491404)

[Hình 3.10 Giao diện Quản lý thông cán bộ DThU 42](#_Toc91491405)

[Hình 3.11 Giao diện Danh sách Khoa, phòng ban DThU 43](#_Toc91491406)

[Hình 3.12 Giao diện Quản lý thông tin lớp học phần DThU 44](#_Toc91491407)

[Hình 3.13 Giao diện Quản lý thông tin các phòng thực hành DThU 45](#_Toc91491408)

[Hình 3.14 Giao diện Quản lý thông tin máy tính của các phòng thực hành 46](#_Toc91491409)

[Hình 3.15 Giao diện Quản lý thông tin đăng ký phòng thực hành DThU 47](#_Toc91491410)

[Hình 3.16 Giao diện Thông tin các lớp học phần 48](#_Toc91491411)

[Hình 3.17 Giao diện Phiếu đăng ký phòng thực hành DThU 49](#_Toc91491412)

[Hình 3.18 Giao diện Thống kê tình trạng hư hỏng của máy tính 50](#_Toc91491413)

[Hình 3.19 Giao diện Báo cáo danh sách đăng ký phòng thực hành DThU 51](#_Toc91491414)

# **PHẦN MỘT: MỞ ĐẦU**

1. **Đặt vấn đề**

Để quản lý phòng thực hành máy tính hiệu quả thì cán bộ quản lý cần phải quản lý rất nhiều giấy tờ liên quan như phiếu đăng ký, số ký nhận phòng…và điều đó làm cho công việc quản lý trở nên khó khăn hơn và tốn nhiều thời gian. Vì vậy cần phải có một hệ thống để công việc quản lý trở nên dễ dàng thuận tiện hơn. Với yêu cầu cấp thiết như vậy nhóm chúng em đã lựa chọn đề tài xây dựng phần mềm “Quản lý đăng ký phòng thực hành DThU” nhằm hỗ trợ cho quá trình quản lý được tốt hơn, tránh sai sót và thuận tiện cho việc quản lý đăng ký của các phòng.

Hệ thống có nhiệm vụ quản lý các thông tin đăng ký mượn phòng thực hành của giảng viên. Hệ thống sẽ giúp cho tổ thực hành máy tính theo dõi dễ dàng quá trình đăng ký của các giảng viên. Nó sẽ làm giảm đáng kể công sức, giấy tờ, sổ sách để lưu trữ. Ngoài việc quản lý thông tin đăng ký, hệ thống cũng cho phép cán bộ quản lý thêm thông tin cần thiết cho việc đăng ký như thông tin các lớp học phần, cán bộ, khoa, phòng thực hành, máy tính,…

Hệ thống lưu trữ kết quả đăng ký và kết xuất dữ liệu ra excel và lập báo cáo danh sách mượn khi cần thiết. Bên cạnh đó, có thể thống kê hay báo cáo tình trạng các máy tính hư hỏng.

1. **Ý nghĩa thực tiễn, lý thuyết của đề tài**

* **Ý nghĩa thực tiễn**

Với thời đại 4.0 phát triển như hiện nay thì việc quản lý các thiết bị, phòng học thực hành bằng chương trình phần mềm hỗ trợ sẽ tiện lợi hơn cách thủ công cũ rất nhiều. Vì vậy, kết quả nghiên cứu của đề tài này sẽ tài liệu giúp cho người sử dụng quản lý dễ dàng hơn, nhanh chóng và hiệu quả.

* **Ý nghĩa lý thuyết**

Qua quá trình học tập và tìm hiểu một số tư liệu tham khảo liên quan tới vấn đề đăng ký mượn phòng thực hành trong quá trình quản lý và cho thấy được những ưu và nhược điểm trong phần mềm quản lý đăng ký mượn phòng thực hành, quá trình quản lý, xử lý cần những gì từ đó đưa ra những tính toán số liệu thích hợp.

1. **Mục tiêu của đề tài**

Mục tiêu của đề tài là xây dựng một hệ thống quản lý phòng thực hành máy tính, nhằm hỗ trợ cho việc đăng ký mượn phòng thực hành, quản lý các thông tin đăng ký của giảng viên và báo cáo thống kê.

1. **Phạm vi của đề tài**

Phạm vi nghiên cứu của đề tài là dãy B4, Trường Đại học Đồng Tháp.

1. **Các phương pháp nghiên cứu**

Các phương pháp nghiên cứu chính được sử dụng trong quá trình thực hiện đề tài này gồm:

- Phương pháp phân tích hệ thống: vận dụng phương pháp này để làm sáng tỏ việc xây dựng phần mềm quản lý đăng ký phòng thực hành DThU.

- Phương pháp lập trình C-Sharp.

- Phương pháp cơ sở dữ liệu SQL.

- Phương pháp công nghệ phần mềm.

1. **Kế hoạch thực hiện đề tài**

|  |  |
| --- | --- |
| **Thời gian** | **Nội dung thực hiện** |
| 1/11/2021 – 5/11/2021 | Nghiên cứu và phân tích đề tài |
| 5/11/2021 – 10/11/2021 | Tham khảo tài liệu liên quan đề tài, phác thảo cơ sở dữ liệu |
| 11/11/2021 – 11/12/2021 | Vẽ các sơ đồ hệ thống bằng UML, thiết kế cơ sở dữ liệu trên SQL và lập trình hệ thống bằng Visual Studio |
| 12/12/2021 – 23/12/2021 | Hoàn thiện Demo, sửa những lỗi phát sinh và hoàn thiện báo cáo |
| 24/12/2021 | Báo cáo với giảng viên và nộp bài. |

# **PHẦN HAI: NỘI DUNG**

# **CHƯƠNG 1: CƠ SỞ LÝ THUYẾT**

* 1. **Giới thiệu hệ quản trị SQL Server**
* ***Khái niệm***

SQL Server (viết tắt của cụm từ Structured Query Language) là một hệ thống quản lý cơ sở dữ liệu quan hệ (Relational Database Management System, viết tắt là RDBMS). SQL Server có khả năng hỗ trợ một số lượng lớn các quy trình xử lý giao dịch, ứng dụng doanh nghiệp và ứng dụng phân tích trong các công ty hoạt động trong lĩnh vực IT.

* ***Tính năng quản lý***

Về phương diện quản trị, Microsoft SQL Server gồm các dịch vụ tích hợp SQL Server, dịch vụ SQL Server Data Quality và dịch vụ SQL Server Master. Hai bộ công cụ dành riêng cho quản trị viên cơ sở dữ liệu và lập trình viên (SQL Server Data Tools) sử dụng trong việc phát triển hệ thống cơ sở dữ liệu cho SQL Server Management Studio, thực hiện nhiệm vụ triển khai, giám sát và quản lý các cơ sở dữ liệu.

* ***Vai trò của SQL server***

SQL Server không phải là một hệ quản trị cơ sở dữ liệu độc lập. Nó là thành phần với vai trò ngôn ngữ làm công cụ giao tiếp của cơ sở dữ liệu và người dùng.

* SQL là một ngôn ngữ đòi hỏi sự tương tác cao. Người dùng có thể dễ dàng trao đổi với tiện ích bằng câu lệnh SQL đến cơ sở dữ liệu và nhận lại kết quả từ đó.
* SQL là một ngôn ngữ lập trình cơ sở dữ liệu. Bằng cách nhúng các câu lệnh SQL trong ngôn ngữ lập trình, các lập trình viên có thể xây dựng được các chương trình ứng dụng giao tiếp với cơ sở dữ liệu
* SQL là một ngôn ngữ lập trình quản trị cơ sở dữ liệu. Quản trị viên cơ sở dữ liệu có thể quản lý và điều khiển các truy cập tới cơ sở dữ liệu thông qua SQL
* SQL được sử dụng như một công cụ giao tiếp với các trình ứng dụng trong hệ thống cơ sở dữ liệu khách – chủ.
  1. **Giới thiệu ngôn ngữ lập trình C#**

C# (đọc là "C thăng" hay "C sharp" *("xi-sáp")*) là một [ngôn ngữ lập trình](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh) [hướng đối tượng](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%C6%B0%E1%BB%9Bng_%C4%91%E1%BB%91i_t%C6%B0%E1%BB%A3ng) được phát triển bởi [Microsoft](https://vi.wikipedia.org/wiki/Microsoft).

Microsoft phát triển C# dựa trên [C++](https://vi.wikipedia.org/wiki/C%2B%2B) và [Java](https://vi.wikipedia.org/wiki/Java_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)). C# được miêu tả là ngôn ngữ có được sự cân bằng giữa C++, [Visual Basic](https://vi.wikipedia.org/wiki/Visual_Basic), [Delphi](https://vi.wikipedia.org/wiki/Delphi_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) và [Java](https://vi.wikipedia.org/wiki/Java_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)).

C# với sự hỗ trợ mạnh mẽ của .NET Framework giúp cho việc tạo một ứng dụng Windows Forms hay WPF ([Windows Presentation Foundatio](https://vi.wikipedia.org/wiki/Windows_Presentation_Foundation)n),…trở nên dễ dàng.

Đặc trưng của ngôn ngữ C#:

* Là một ngôn ngữ thuần hướng đối tượng.
* Là ngôn ngữ khá đơn giản, chỉ có khoảng 80 từ khóa và hơn mười mấy kiểu dữ liệu được dựng sẵn.
* C# không khuyến khích sử dụng con trỏ như trong C++ nhưng nếu bạn thực sự muốn sử dụng thì phải đánh dấu đây là mã không an toàn.
* C# đã loại bỏ đa kế thừa trong C++ mà thay vào đó C# sẽ hỗ trợ thực thi giao diện interface.

Ngoài ra, C# còn có những ưu điểm nổi bật như:

* Gần gũi với các ngôn ngữ lập trình thông dụng ( C++, Java, Pascal).
* Xây dựng dựa trên nền tảng của các ngôn ngữ lập trình mạnh nên thừa hưởng những ưu điểm của những ngôn ngữ đó.
* Cải tiến các khuyết điểm của C/C++ như con trỏ, các hiệu ứng phụ,…
* Dễ tiếp cận, dễ phát triển.
* Được sự chống lưng của .NET Framework.

Đi kèm với những điểm mạnh C# cũng có những nhược điểm:

* Nhược điểm lớn nhất của C# là chỉ chạy trên nền Windows và có cài .NET Framework.

Thao tác đối với phần cứng yếu hơn so với ngôn ngữ khác. Hầu hết phải dựa vào Windows.

## **Cơ bản về phân tích, thiết kế hệ thống hướng đối tượng và UML**

* **Khái niệm OOAD (Object Oriented Analysis and Design)**

Trong kỹ nghệ phần mềm để sản xuất được một sản phẩm phần mềm người ta chia quá trình phát triển sản phẩm ra nhiều giai đoạn như thu thập và phân tích yêu cầu, phân tích và thiết kế hệ thống, phát triển (coding), kiểm thử, triển khai và bảo trì. Trong đó, giai đoạn phân tích, thiết kế bao giờ cũng là giai đoạn khó khăn và phức tạp nhất. Giai đoạn này giúp chúng ta hiểu rõ yêu cầu đặt ra, xác định giải pháp, mô tả chi tiết giải pháp. Nó trả lời 2 câu hỏi What (phần mềm này làm cái gì?) và How (làm nó như thế nào?).

Để phân tích và thiết kế một phần mềm thì có nhiều cách làm, một trong những cách làm đó là xem hệ thống gồm những đối tượng sống trong đó và tương tác với nhau. Việc mô tả được tất cả các đối tượng và sự tương tác của chúng sẽ giúp chúng ta hiểu rõ hệ thống và cài đặt được nó. Phương thức này gọi là Phân tích thiết kế hướng đối tượng (OOAD)

* **Khái niệm về UML (Unified Modeling Language)**

UML là ngôn ngữ mô hình hóa hợp nhất dùng để biểu diễn hệ thống. Nói một cách đơn giản là nó dùng để tạo ra các bản vẽ nhằm mô tả thiết kế hệ thống. Các bản vẽ này được sử dụng để các nhóm thiết kế trao đổi với nhau cũng như dùng để thi công hệ thống (phát triển), thuyết phục khách hàng, các nhà đầu tư...

* **Tại sao lại là OOAD và UML?**

OOAD cần các bản vẽ để mô tả hệ thống được thiết kế, còn UML là ngôn ngữ mô tả các bản vẽ nên cần nội dung thể hiện. Do vậy, chúng ta phân tích và thiết kế theo hướng đối tượng và sử dụng UML để biểu diễn các thiết kế đó nên chúng thường đi đôi với nhau.

* **OOAD sử dụng UML**

UML sử dụng để vẽ cho nhiều lĩnh vực khác nhau như phần mềm, cơ khí, xây dựng,…trong phạm vi các bài viết này chúng ta chỉ nghiên cứu cách sử dụng UML cho phân tích và thiết kế hướng đối tượng trong ngành phần mềm. OOAD sử dụng UML bao gồm các thành phần sau:

* ***View (góc nhìn)***

Mỗi góc nhìn như thầy bói xem voi, nó không thể hiện hết hệ thống nhưng thể hiện rõ hệ thống ở một khía cạnh. Chính vì thế trong xây dựng có bản vẽ kiến trúc (nhìn về mặt kiến trúc), bản vẽ kết cấu (nhìn về mặt kết cấu), bản vẽ thi công (nhìn về mặt thi công).

Trong đó,

* + Use Case View: cung cấp góc nhìn về các ca sử dụng giúp chúng ta hiểu hệ thống có gì? ai dùng và dùng nó như thế nào.
  + Logical View: cung cấp góc nhìn về cấu trúc hệ thống, xem nó được tổ chức như thế nào. Bên trong nó có gì.
  + Process View: cung cấp góc nhìn động về hệ thống, xem các thành phần trong hệ thống tương tác với nhau như thế nào.
  + Component View: Cũng là một góc nhìn về cấu trúc giúp chúng ta hiểu cách phân bổ và sử dụng lại các thành phần trong hệ thống ra sao.
  + Deployment View: cung cấp góc nhìn về triển khai hệ thống, nó cũng ảnh hưởng lớn đến kiến trúc hệ thống.

Tập hợp các góc nhìn này sẽ giúp chúng ta hiểu rõ hệ thống cần phân tích, thiết kế. Trong hình 1 chúng ta thấy góc nhìn Use Case View nằm ở giữa và chi phối tất cả các góc nhìn còn lại. Chính vì thế chúng ta thường thấy các tài liệu nói về 4 view + 1 chứ không phải 5 view nhằm nhấn mạnh vai trò của UseCase View.

* ***Diagram (Bản vẽ)***

Diagram các bạn có thể dịch là sơ đồ. Tuy nhiên ở đây chúng ta sử dụng từ bản vẽ cho dễ hình dung. Các bản vẽ được dùng để thể hiện các góc nhìn của hệ thống.

Trong đó,

* Use Case Diagram: bản vẽ mô tả về ca sử dụng của hệ thống. Bản vẽ này sẽ giúp chúng ta biết được ai sử dụng hệ thống, hệ thống có những chức năng gì. Lập được bản vẽ này bạn sẽ hiểu được yêu cầu của hệ thống cần xây dựng.
* Class Diagram: bản vẽ này mô tả cấu trúc của hệ thống, tức hệ thống được cấu tạo từ những thành phần nào. Nó mô tả khía cạnh tĩnh của hệ thống.
* Object Diagram: Tương tự như Class Diagram nhưng nó mô tả đến đối tượng thay vì lớp (Class).
* Sequence Diagram: là bản vẽ mô tả sự tương tác của các đối tượng trong hệ thống với nhau được mô tả tuần tự các bước tương tác theo thời gian.
* Collaboration Diagram: tương tự như sequence Diagram nhưng nhấn mạnh về sự tương tác thay vì tuần tự theo thời gian.
* State Diagram: bản vẽ mô tả sự thay đổi trạng thái của một đối tượng. Nó được dùng để theo dõi các đối tượng có trạng thái thay đổi nhiều trong hệ thống.
* Activity Diagram: bản vẽ mô tả các hoạt động của đối tượng, thường được sử dụng để hiểu về nghiệp vụ của hệ thống.
* Component Diagram: bản vẽ mô tả về việc bố trí các thành phần của hệ thống cũng như việc sử dụng các thành phần đó.
* Deployment Diagram: bản vẽ mô tả việc triển khai của hệ thống như việc kết nối, cài đặt, hiệu năng của hệ thống v.v…
* ***Notations (các ký hiệu)***

Notations là các ký hiệu để vẽ, nó như từ vựng trong ngôn ngữ tự nhiên. Bạn phải biết từ vựng thì mới ghép thành câu, thành bài được. Chúng ta sẽ tìm hiểu kỹ các notations trong từng bản vẽ sau này.

* ***Mechanisms (Rules)***

Mechanisms là các qui tắc để lập nên bản vẽ, mỗi bản vẽ có qui tắc riêng và bạn phải nắm được để tạo nên các bản vẽ thiết kế đúng. Các qui tắc này chúng ta sẽ bàn kỹ trong các bài về các bản vẽ.

## **Mô hình 3 lớp**

* ***Khái niệm***

Gồm 3 lớp, đó là:

**GUI** Layer: Lớp này là lớp hiển thị giao diện và các chức năng để người dùng cuối sử dụng.

**Business** (BUS) Layer: Đây là lớp nhận các yêu cầu từ lớp GUI và truy xuất lên lớp Data để lấy thông tin và trả về GUI.

**Data Access** Layer: Lớp này là lớp để truy xuất với CSDL, chỉ duy nhất lớp này được làm việc với database.

(Không cần thiết) **DTO** Layer: Lớp này chỉ là phụ thôi, đây là lớp định nghĩa các table trong database của bạn, định nghĩa cột của nó cũng như để ta gán data khi query lấy dữ liệu. Các bạn có thể hiểu nôm na là 1 dạng cơ bản ORM (Object Relation Mapping).

* ***Lợi thế của mô hình 3 lớp***

Việc phân chia thành từng lớp giúp cho code được tường minh hơn. Nhờ vào việc chia ra từng lớp đảm nhận các chức năng khác nhau và riêng biệt như giao diện, xử lý, truy vấn thay vì để tất cả lại một chỗ. Nhằm giảm sự kết dính.

Dễ bảo trì khi được phân chia, thì một thành phần của hệ thống sẽ dễ thay đổi. Việc thay đổi này có thể được cô lập trong 1 lớp, hoặc ảnh hưởng đến lớp gần nhất mà không ảnh hưởng đến cả chương trình.

Dễ phát triển, tái sử dụng: khi chúng ta muốn thêm một chức năng nào đó thì việc lập trình theo một mô hình sẽ dễ dàng hơn vì chúng ta đã có chuẩn để tuân theo. Và việc sử dụng lại khi có sự thay đổi giữa hai môi trường ( Winform sang Webfrom ) thì chỉ việc thay đổi lại lớp GUI.

Dễ bàn giao. Nếu mọi người đều theo một quy chuẩn đã được định sẵn, thì công việc bàn giao, tương tác với nhau sẽ dễ dàng hơn và tiết kiệm được nhiều thời gian.

Dễ phân phối khối lượng công việc. Mỗi một nhóm, một bộ phận sẽ nhận một nhiệm vụ trong mô hình 3 lớp. Việc phân chia rõ ràng như thế sẽ giúp các lập trình viên kiểm soát được khối lượng công việc của mình.

* ***Lưu ý khi xây dựng mô hình 3 lớp***

Cần một solution riêng cho project.

Cần 3 project khác nhau để làm nên 3 lớp, tên **Project** đặt như sau:

* Lớp **GUI**: GUI\_\* (VD: GUI\_QuanLy)
* Lớp **Business**: BUS\_\* (VD: BUS\_QuanLy)
* Lớp **Data Access**: DAL\_\* (VD: DAL\_QuanLy)
* Lớp **DTO**: DTO\_\* (VD: DTO\_QuanLy)

Bên trong 3 lớp như trên các file đặt cần có các tiền tố như sau:  
Ví dụ mình có một table tên là **ThanhVien**

* Lớp **GUI**: GUI\_\* (VD: GUI**\_ThanhVien**)
* Lớp **Business**: BUS\_\* (VD: GUI**\_ThanhVien**)
* Lớp **Data Access**: DAL\_\* (VD: GUI**\_ThanhVien**)

Lớp **DTO**: DTO\_\* (VD: DTO**\_ThanhVien**)

**CHƯƠNG 2: PHÂN TÍCH PHẦN MỀM HƯỚNG ĐỐI TƯỢNG**

* 1. **Mô tả bài toán**
     1. **Hiện trạng nghiệp vụ của bài toán**

Mỗi giảng viên sẽ phụ trách một hoặc nhiều lớp học, ứng với mỗi lớp học giảng viên sẽ phải sắp xếp lịch học thực hành sao cho phù hợp nhất, và ứng với mỗi lớp giảng dạy giảng viên sẽ phải đăng ký phòng máy với một thời gian nhất định. Mỗi phòng máy sẽ có cấu hình máy tính, chất lượng khác nhau; cán bộ quản lý phòng thực hành có nhiệm vụ là phải nắm rõ tường tận tất cả các phòng máy này (thời gian, giảng viên đăng ký, số phòng, phòng máy còn trống,…) Mỗi giảng viên khi có nhu cầu đăng ký phòng học, họ sẽ phải lên lịch môn học và thời gian cụ thể sau đó liên hệ với phòng đào tạo để kiểm tra, đăng ký lịch và sau đó phòng đào tạo duyệt, bố trí phòng máy tính cho môn học đó, sao cho không xảy ra tình trạng trùng lịch hay trùng phòng máy. Từ đó, lịch sẽ được chuyển đến tổ thực hành máy tính để cán bộ quản lý phòng máy biết và trực tiếp quản lý, kiểm tra quá trình giảng dạy và học tập ở phòng máy của môn học đó. Công việc của cán bộ quản lý phòng máy ngoài việc kiểm tra tài sản ở mỗi phòng hàng năm thì còn quản lý việc giao phòng máy hay nhận trả phòng máy từ các giảng viên đăng ký mượn; sau đó sẽ phải kiểm tra các trang thiết bị tại phòng thông qua hệ thống camera được gắn ở mỗi phòng; nếu mọi kiểm tra đều bình thường thì ca đăng ký thực hành hôm đó hoàn tất. Mặt khác, bộ phận bảo trì máy tính sẽ kiểm tra định kỳ để xem máy có cần cập nhật hay sửa chữa các công cụ phần mềm trên máy nhằm phù hợp hơn với quá trình giảng dạy của các giảng viên.

* + 1. **Mô tả hệ thống**

Mỗi cán bộ công tác tại Trường đại học Đồng Tháp sẽ được sắp xếp vào khoa/phòng ban phù hợp với chuyên môn và đảm nhận một chức vụ để làm việc tại trường thông qua những thông tin sau: *mã cán bộ, họ tên, giới tính, năm sinh, chức vụ, mã khoa, địa chỉ, email và số điện thoại.*

Vào đầu mỗi học kỳ, giảng viên sẽ được trưởng bộ môn của khoa đó phân công dạy một hoặc nhiều lớp học, đối với mỗi lớp học phần cần học thực hành phòng máy thì giảng viên phải sắp xếp đăng ký phòng thực hành với cán bộ quản lý ở một thời gian nhất định. Thông tin lớp học phần bao gồm: *ID học phần, mã học phần, tên học phần, số tín chỉ, học kì và niên khóa.*

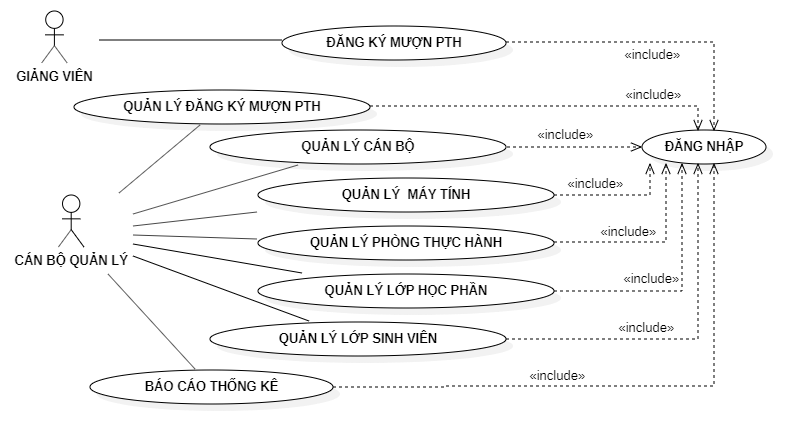
Để đáp ứng cho việc mượn phòng thực hành giảng dạy thì trường đã phân công tổ thực hành máy tính làm nhiệm vụ nắm rõ tường tận thông tin các phòng thực hành (*số phiếu, ngày đăng ký, tiết bắt đầu, số tiết,…*) Do đó, mỗi phòng thực hành tại trường sẽ được quản lý qua các thông tin sau: *mã phòng, vị trí, số lượng máy và tình trạng phòng*. Đồng thời, để tiện theo dõi các phòng máy và bảo trì máy tính thì mỗi máy tính sẽ có các thông tin như: *mã máy, hiệu máy, nhà sản xuất, tình trạng máy, mã phòng.*

Để có thể đăng ký phòng thực hành, giảng viên phải điền đầy đủ các thông tin như: *ngày đăng ký, tiết bắt đầu mượn, tổng số tiết, ID học phần, mã phòng, mã cán bộ mượn phòng.* Việc mượn phòng thực hành chỉ được thực hiện khi giảng viên cung cấp đầy đủ thông tin và không trùng lịch mượn phòng của giảng viên khác thì việc đăng ký mượn thành công, ngược lại thì sẽ thông báo tới giảng viên lý do đăng ký không thành công như trùng lịch phòng thực hành, có quá nhiều sinh viên trùng lịch học môn học khác…

Cán bộ quản lý là người có nhiệm vụ quản lý thông tin các phòng thực hành và tiếp nhận các sự cố hư hỏng thiết bị trong quá trình giảng dạy của giảng viên. Sau mỗi thời gian nhất định bộ phận quản lý tổ thực hành máy tính sẽ tổng hợp lại tình trạng của mỗi phòng xem máy tính nào còn sử dụng tốt hay cần bảo trì sửa chữa. Từ đó lập báo cáo để đề xuất sửa chữa hoặc cấp mới thiết bị với Ban giám hiệu trường.

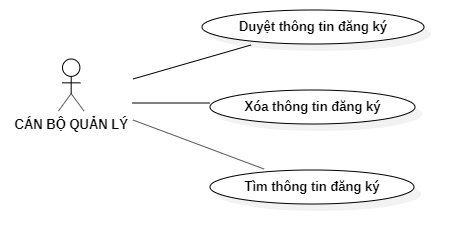
* + 1. **Yêu cầu chức năng**
* Quản lý thông tin đăng ký: hệ thống cho phép quản lý thông tin đăng ký mượn phòng thực hành, đăng ký mượn, sửa hoặc hủy đăng ký.
* Quản lý thông tin cán bộ: hệ thống cho phép quản lý thông tin cán bộ, thêm, sửa, xóa thông tin cán bộ.
* Quản lý phòng thực hành: hệ thống cho phép quản lý thông tin phòng thực hành, thêm, sửa, xóa thông tin phòng thực hành.
* Quản lý máy tính phòng thực hành: hệ thống cho phép quản lý thông tin máy tính trong phòng thực hành, thêm, sửa, xóa thông tin máy tính
* Quản lý các lớp học phần: hệ thống cho phép quản lý thông tin lớp học phần, thêm, sửa, xóa thông tin lớp học phần
* Lập báo cáo, thống kê: hệ thống cho phép thống kê thông tin đăng ký mượn phòng thực hành và thống kê tình trạng máy của các phòng thực hành
  1. **Phân tích phần mềm hướng đối tượng**
     1. **Sơ đồ Use case**

Khái quát chức năng chính của hệ thống. Use case dưới đây mô tả các chức năng chính một cách tổng quát để có thể dễ dàng nhìn thấy được trên quan điểm của các tác nhân.



Hình 2.1 Use case tổng quát

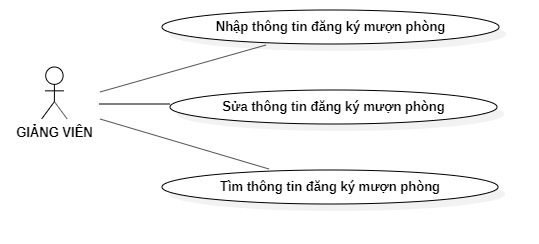
* + 1. **Đặc tả Use case**
* **Use case quản lý đăng ký**

**

Hình 2.2 Use case quản lý đăng ký

|  |
| --- |
| Tóm tắt: use case bắt đầu khi cán bộ quản lý đăng nhập vào hệ thống và chọn chức năng quản lý đăng ký phòng thực hành. |
| Tác nhân liên quan: Cán bộ quản lý |
| Dòng sự kiện chính:   1. Hệ thống sẽ tiến hành kiểm tra thông tin đăng ký 2. Hệ thống sẽ kiểm tra phòng thực hành 3. Hệ thống sẽ duyệt đăng ký 4. Cập nhật danh sách đăng ký mượn phòng và kết thúc. |
| Dòng sự kiện phụ:  A1: Kiểm tra thông tin có hợp lệ theo quy định đăng ký mượn phòng.  A2: Kiểm tra phòng thực hành có bị trùng lịch không, nếu trùng lịch sẽ đổi phòng cho giảng viên hoặc xóa đăng ký nếu không còn phòng phù hợp. |
| Tiền điều kiện: Đăng nhập thành công với quyền Cán bộ quản lý |
| Hậu điều kiện: Cập nhật tình trạng đăng ký thành công. |

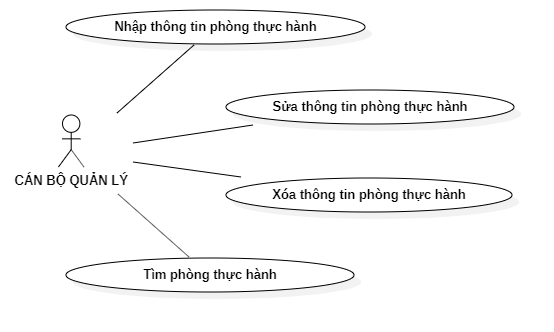
* **Use case đăng ký phòng thực hành**

**

Hình 2.3 Use case đăng ký phòng thực hành

|  |
| --- |
| Tóm tắt: use case bắt đầu khi giảng viên đăng nhập vào hệ thống và chọn chức năng đăng ký phòng thực hành. |
| Tác nhân liên quan: Giảng viên |
| Dòng sự kiện chính:   1. Chọn chức năng đăng ký phòng thực hành, hệ thống sẽ hiển thị màn hình đăng ký phòng thực hành máy tính bao gồm danh sách đăng ký phòng thực hành, các chức năng con thêm, sửa, tìm kiếm. 2. Hệ thống sẽ kiểm tra khi giảng viên cập nhật thông tin đăng ký 3. Hệ thống sẽ hiển thị thông tin đăng ký lên danh sách đăng ký. |
| Dòng sự kiện phụ:  A2: Kiểm tra thông tin có vi phạm quy định bỏ trống hoặc nhập sai kiểu dữ liệu: hệ thống sẽ hiển thị thông báo “Vui lòng nhập đúng qui định và đầy đủ thông tin!” |
| Tiền điều kiện: Đăng nhập thành công với quyền giảng viên |
| Hậu điều kiện: Cập nhật danh sách đăng ký phòng thực hành. |

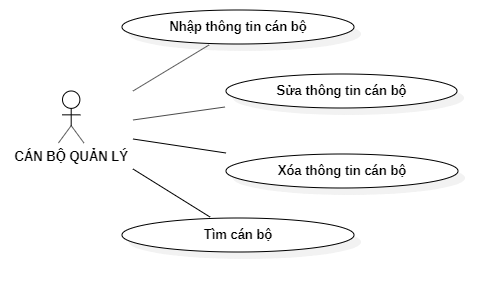
* **Use case quản lý phòng thực hành máy tính**

**

Hình 2.4 Use case quản lý phòng thực hành máy tính

|  |
| --- |
| Tóm tắt: use case bắt đầu khi cán bộ quản lý đăng nhập vào hệ thống và chọn chức năng quản lý phòng thực hành máy tính. |
| Tác nhân liên quan: Cán bộ quản lý |
| Dòng sự kiện chính:   1. Chọn chức năng quản lý phòng thực hành máy tính, hệ thống sẽ hiển thị màn hình quản lý phòng thực hành máy tính bao gồm danh sách phòng thực hành, các chức năng con thêm, sửa, xóa. 2. Hệ thống kiểm tra thông tin khi cán bộ quản lý cập nhật thông tin. 3. Hiển thị thông báo thành công tương ứng theo các chức năng thêm sửa xóa. 4. Cập nhật danh sách phòng thực hành và kết thúc. |
| Dòng sự kiện phụ:  A2: Kiểm tra thông tin có vi phạm quy định bỏ trống hoặc nhập sai kiểu dữ liệu: hệ thống sẽ hiển thị thông báo “Vui lòng nhập đúng qui định và đầy đủ thông tin!” |
| Tiền điều kiện: Đăng nhập thành công với quyền cán bộ quản lý |
| Hậu điều kiện: Cập nhật danh sách phòng thực hành và hiển thị thông báo tương ứng. |

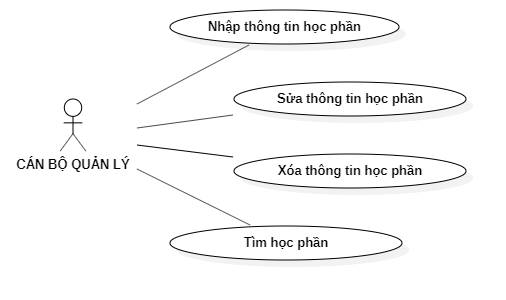
* **Use case quản lý cán bộ**

**

Hình 2.5 Use case quản lý cán bộ

|  |
| --- |
| Tóm tắt: use case bắt đầu khi cán bộ quản lý đăng nhập vào hệ thống và chọn chức năng quản lý cán bộ. |
| Tác nhân liên quan: Cán bộ quản lý |
| Dòng sự kiện chính:   1. Chọn chức năng quản lý cán bộ, hệ thống sẽ hiển thị màn hình quản lý cán bộ bao gồm danh sách cán bộ, các chức năng con thêm, sửa, xóa cán bộ. 2. Hệ thống kiểm tra thông tin khi cán bộ quản lý cập nhật thông tin. 3. Hiển thị thông báo thành công tương ứng theo các chức năng thêm sửa xóa. 4. Cập nhật danh sách cán bộ và kết thúc. |
| Dòng sự kiện phụ:  A1: Kiểm tra thông tin có vi phạm quy định bỏ trống hoặc nhập sai kiểu dữ liệu: hệ thống sẽ hiển thị thông báo “Vui lòng nhập đúng qui định và đầy đủ thông tin!” |
| Tiền điều kiện: Đăng nhập thành công với quyền cán bộ quản lý |
| Hậu điều kiện: Cập nhật danh sách cán bộ và hiển thị thông báo tương ứng. |

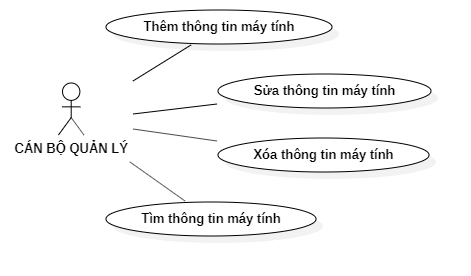
* **Use case quản lý lớp học phần**



Hình 2.6 Use case quản lý lớp học phần

|  |
| --- |
| Tóm tắt: usecase bắt đầu khi cán bộ quản lý đăng nhập vào hệ thống và chọn chức năng quản lý lớp học phần. |
| Tác nhân liên quan: Cán bộ quản lý |
| Dòng sự kiện chính:   1. Chọn chức năng quản lý lớp học phần, hệ thống sẽ hiển thị màn hình quản lý lớp học phần bao gồm danh sách học phần, các chức năng con thêm, sửa, xóa học phần. 2. Hệ thống kiểm tra thông tin khi cán bộ quản lý nhập vào. 3. Hiển thị thông báo thành công tương ứng theo các chức năng thêm sửa xóa. 4. Cập nhật danh sách học phần và kết thúc. |
| Dòng sự kiện phụ:  A1: Kiểm tra thông tin có vi phạm quy định bỏ trống hoặc nhập sai kiểu dữ liệu: hệ thống sẽ hiển thị thông báo “Vui lòng nhập đúng qui định và đầy đủ thông tin!” |
| Tiền điều kiện: Đăng nhập thành công với quyền cán bộ quản lý |
| Hậu điều kiện: Cập nhật danh sách học phần và hiển thị thông báo tương ứng. |

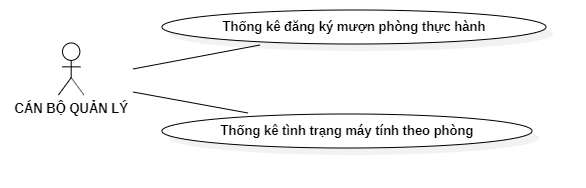
* **Use case quản lý máy tính**



Hình 2.7 Use case quản lý máy tính

|  |
| --- |
| Tóm tắt: use case bắt đầu khi cán bộ quản lý đăng nhập vào hệ thống và chọn chức năng quản lý máy tính. |
| Tác nhân liên quan: Cán bộ quản lý |
| Dòng sự kiện chính:   1. Chọn chức năng quản lý máy tính, hệ thống sẽ hiển thị màn hình quản lý máy tính bao gồm danh sách máy tính, các chức năng con thêm, sửa, xóa. 2. Hệ thống kiểm tra thông tin khi cán bộ quản lý nhập vào. 3. Hiển thị thông báo thành công tương ứng theo các chức năng thêm sửa xóa. 4. Cập nhật danh sách máy tính và kết thúc. |
| Dòng sự kiện phụ:  A1: Kiểm tra thông tin có vi phạm quy định bỏ trống hoặc nhập sai kiểu dữ liệu: hệ thống sẽ hiển thị thông báo “Vui lòng nhập đúng qui định và đầy đủ thông tin!” |
| Tiền điều kiện: Đăng nhập thành công với quyền cán bộ quản lý |
| Hậu điều kiện: Cập nhật danh sách máy tính và hiển thị thông báo tương ứng. |

* **Use case báo cáo thống kê**



Hình 2.8 Use case báo cáo thống kê

|  |
| --- |
| Tóm tắt: use case bắt đầu khi cán bộ quản lý đăng nhập vào hệ thống và chọn chức năng báo cáo, thống kê. |
| Tác nhân liên quan: Cán bộ quản lý |
| Dòng sự kiện chính:   1. Hệ thống sẽ hiển thị form thống kê 2. Sau khi cán bộ quản lý chọn hình thức, hệ thống sẽ hiển thị danh sách thống kê tương ứng 3. Hệ thống cho phép kết xuất dữ liệu ra dạng excel. |
| Tiền điều kiện: Đăng nhập thành công với quyền cán bộ quản lý |
| Hậu điều kiện: Hiển thị danh sách thống kê theo yêu cầu. |

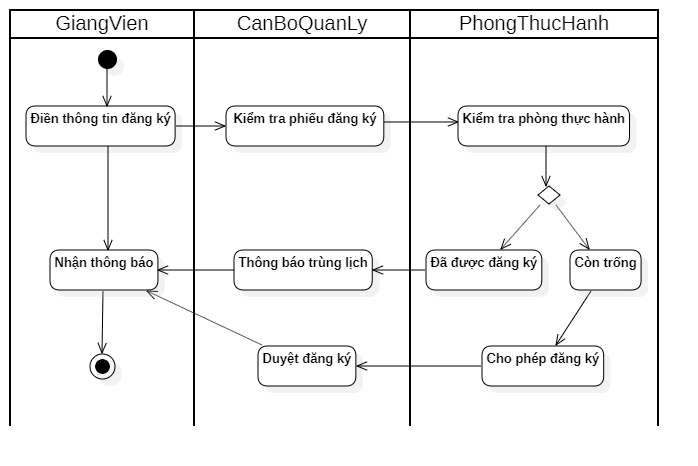
* **Use case chi tiết**



Hình 2.9 Use case chi tiết

* + 1. **Sơ đồ hoạt động**
* **Hoạt động đăng ký mượn phòng thực hành**

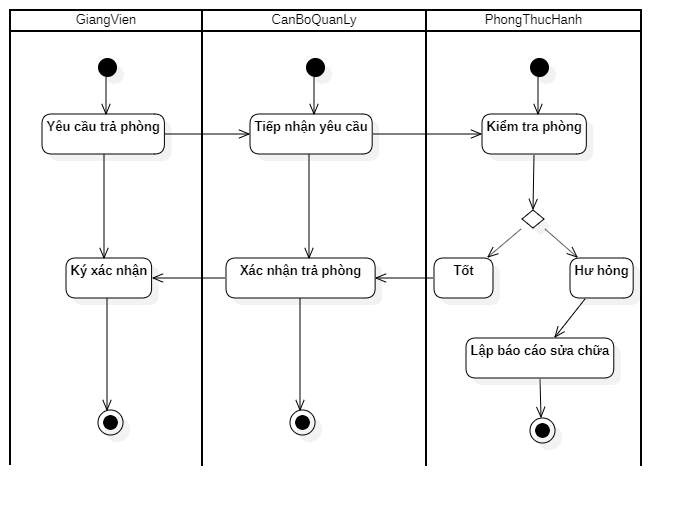
Giảng viên điền thông tin đăng ký, cán bộ quản lý tiến hành kiểm tra phiếu đăng ký sau đó tiến hành kiểm tra phòng thực hành. Phòng thực hành có hai trạng thái đã được đăng ký hoặc còn trống. Nếu phòng còn trống thì cho phép đăng ký, cán bộ quản lý xét duyệt đăng ký và thông báo cho giảng viên, kết thúc. Ngược lại, phòng đã được đăng ký thì thông báo trùng lịch và kết thúc.



Hình 2.10 Hoạt động đăng ký mượn phòng thực hành

* **Hoạt động trả phòng thực hành**

Giảng viên yêu cầu trả phòng, cán bộ quản lý tiếp nhận yêu cầu và liên hệ phòng thực hành để kiểm tra tình trạng phòng. Nếu phòng tốt thì cán bộ quản lý xác nhận trả phòng cho giảng viên ký xác nhận. Còn nếu phòng hư hỏng thì lập báo cáo sửa chữa và kết thúc.



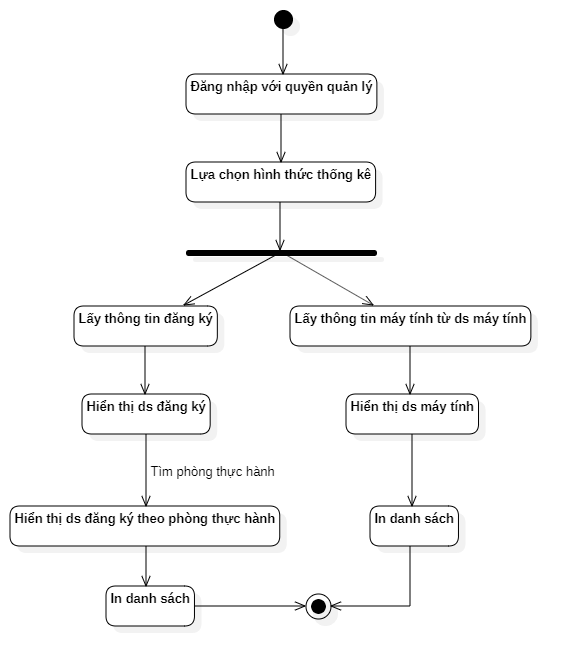
Hình 2.11 Hoạt động trả phòng thực hành

* **Hoạt động thống kê**

Người dùng đăng nhập với quyền quản lý và lựa chọn hình thức thống kê, hệ thống thực hiện đồng thời hai hình thức là lấy thông tin đăng ký và lấy thông tin máy tính từ danh sách phòng thực hành.

Khi lấy thông tin đăng ký sẽ hiển thị danh sách đăng ký, ta có thể hiển thị danh sách đăng ký hoặc lựa chọn tìm kiếm theo phòng thực hành và in danh sách đã tìm, kết thúc.

Khi lấy danh sách từ máy tính sẽ hiển thị danh sách máy tính và in danh sách máy tính để dễ dàng kiểm tra tình trạng của máy tính.



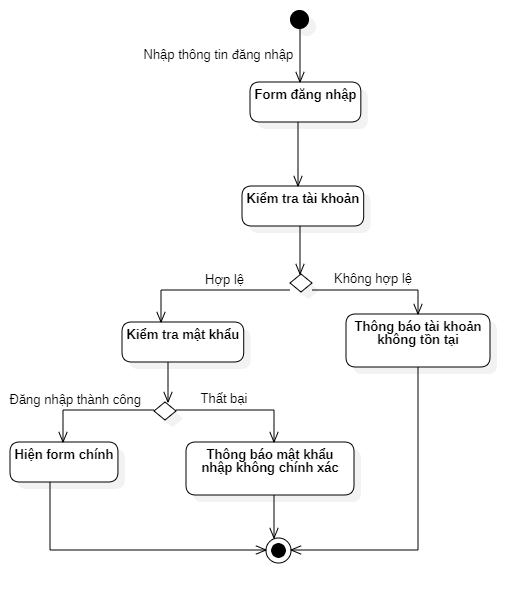
Hình 2.12 Hoạt động thống kê

* **Hoạt động đăng nhập**

Người dùng nhập thông tin đăng nhập vào form đăng nhập, hệ thống sẽ tiến hành kiểm tra thông tin đăng nhập.

Nếu thông tin hợp lệ sẽ tiến hành kiểm tra mật khẩu và đăng nhập thành công hoặc thất bại, nếu đăng nhập thành công sẽ hiện form chính của phần mềm rồi kết thúc. Ngược lại, thông báo mật khẩu không chính xác và kết thúc.

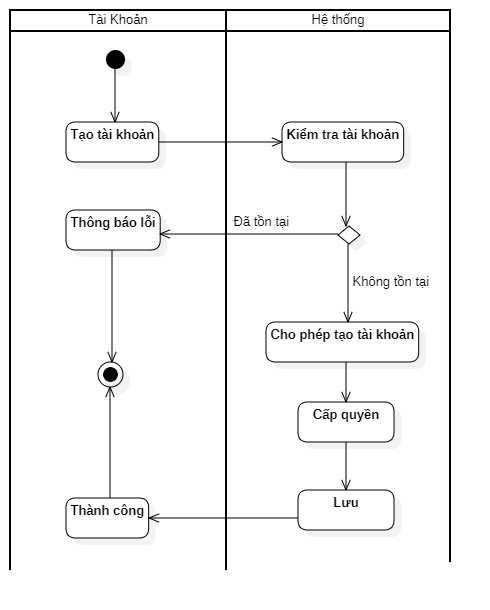
Nếu thông tin không hợp lệ thì thông báo tài khoản không tồn tại và kết thúc.



Hình 2.13 Hoạt động đăng nhập

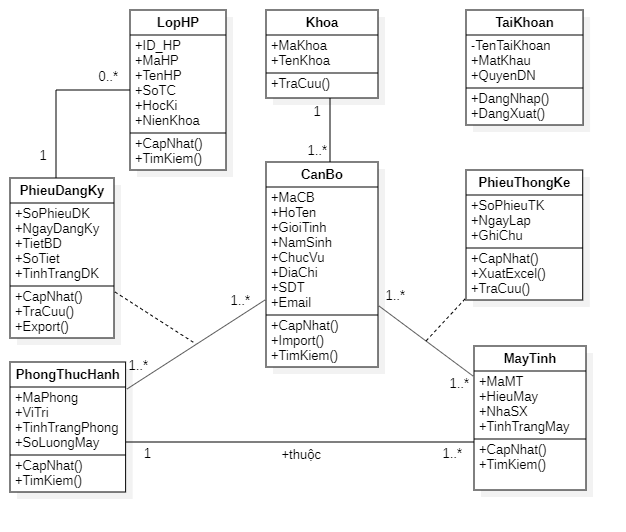
* **Hoạt động tạo tài khoản**

Cán bộ tạo tài khoản, hệ thống tiến hành kiểm tra thông tin tài khoản nếu tài khoản đã tồn tại thì thông báo lỗi và kết thúc. Ngược lại, hệ thống cho phép tạo tài khoản, cấp quyền, lưu thông tin vào cơ sở dữ liệu và thông báo thành công, kết thúc.



Hình 2.14 Hoạt động tạo tài khoản

* + 1. **Sơ đồ lớp**

****

Hình 2.15 Sơ đồ lớp

# **CHƯƠNG 3: XÂY DỰNG PHẦN MỀM HƯỚNG ĐỐI TƯỢNG**

* 1. **Cách tổ chức dữ liệu**
     1. **Thiết kế dữ liệu vật lý – PMD**
* ***Bảng CanBo****: lưu thông tin cán bộ công tác tại trường DThU*
* *Bảng lưu trữ dữ liệu:*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **MaCB** | **HoTen** | **GioiTinh** | **NamSinh** | **ChucVu** | **DiaChi** | **SDT** | **Email** | **MaKhoa** |
| CB002 | Võ Thị Bích Trăm | Nữ | 2021-12-19 | Giảng viên | Hồng Ngự | 0328420146 | tram@gmail.com | K02 |

* *Bảng cấu trúc:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | MaCB | varchar(10) | PK | Mã cán bộ |
| 2 | HoTen | nvarchar(50) |  | Họ và tên cán bộ |
| 3 | GioiTinh | nvarchar(5) |  | Giới tính cán bộ |
| 4 | NamSinh | date |  | Năm sinh cán bộ |
| 5 | ChucVu | nvarchar(100) |  | Chức vụ cán bộ |
| 6 | DiaChi | nvarchar(100) |  | Địa chỉ cán bộ |
| 7 | SDT | varchar(10) |  | Số điện thoại cán bộ |
| 8 | Email | varchar(50) |  | Email cán bộ |
| 9 | MaKhoa | varchar(10) | FK | Mã khoa cán bộ |

* ***Bảng Khoa****: lưu thông tin của các khoa trong trường DthU*
* *Bảng lưu trữ dữ liệu:*

|  |  |
| --- | --- |
| **MaKhoa** | **TenKhoa** |
| K01 | Khoa Ngoại Ngữ |

* *Bảng cấu trúc:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | MaKhoa | varchar(10) | PK | Mã khoa |
| 2 | TenKhoa | nvarchar(100) |  | Tên khoa |

* ***Bảng LopHP****: lưu thông tin của các lớp học phần*
* *Bảng lưu trữ dữ liệu:*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID\_HP** | **MaHP** | **TenHP** | **SoTC** | **HocKi** | **NienKhoa** |
| 94 | IN4119 | Lập trình DotNet | 3 | 1 | 2020-2021 |

* *Bảng cấu trúc:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | ID\_HP | int | PK | ID lớp học phần |
| 2 | MaHP | varchar(10) |  | Mã lớp học phần |
| 3 | TenHP | nvarchar(100) |  | Tên lớp học phần |
| 4 | SoTC | int |  | Số tín chỉ của học phần |
| 5 | HocKi | nvarchar(5) |  | Học kì của học phần |
| 6 | NienKhoa | varchar(15) |  | Năm học của học phần |

* ***Bảng MayTinh****: lưu thông tin của các máy tính trong phòng thực hành*
* *Bảng lưu trữ dữ liệu:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MaMT** | **HieuMay** | **NhaSX** | **Tình trạng máy** | **MaPhong** |
| MT001 | Dell | Trung Quốc | Hư hỏng | 101B4 |

* *Bảng cấu trúc:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | MaMT | varchar(10) | PK | Mã máy tính |
| 2 | HieuMay | varchar(20) |  | Hiệu máy tính |
| 3 | NhaSX | nvarchar(20) |  | Nhà sản xuất máy tính |
| 4 | TinhTrangMay | nvarchar(50) |  | Tình trạng máy tính |
| 5 | MaPhong | varchar(10) | FK | Mã phòng thực hành |

* ***Bảng PhieuDangKy****: lưu thông tin của phiếu đăng ký mượn phòng thực hành*
* *Bảng lưu trữ dữ liệu:*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **SoPhieuDK** | **NgayDangKy** | **TietBD** | **SoTiet** | **MaPhong** | **ID\_HP** | **MaCB** | **TinhTrangDK** |
| 414 | 2021-12-19 | 1 | 3 | 104B2 | 452 | CB005 | Chưa duyệt |

* *Bảng cấu trúc:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | SoPhieuDK | varchar(10) | PK | Số phiếu đăng ký |
| 2 | NgayDangKy | date |  | Ngày đăng ký |
| 3 | TietBD | int |  | Tiết bắt đầu học |
| 4 | SoTiet | int |  | Số tiết học |
| 5 | MaPhong | varchar(10) | FK | Mã phòng mượn |
| 6 | ID\_HP | int | FK | Mã học phần |
| 7 | MaCB | varchar(10) | FK | Mã cán bộ |
| 8 | TinhTrangDK | nvarchar(20) |  | Tình trạng đăng ký |

* ***Bảng PhieuThongKe****: lưu thông tin của phiếu thống kê máy tính hư hỏng*
* *Bảng lưu trữ dữ liệu:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SoPhieuTK** | **NgayLap** | **GhiChu** | **MaCB** | **MaMT** |
| 298 | 2021-12-18 | Hư main | CB007 | MT005 |

* *Bảng cấu trúc:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | SoPhieuTK | varchar(10) | PK | Mã số phiếu thống kê |
| 2 | NgayLap | date |  | Ngày lập phiếu thống kê |
| 3 | GhiChu | nvarchar(100) |  | Ghi chú |
| 4 | MaCB | varchar(10) | FK | Mã cán bộ đăng ký mượn |
| 5 | MaMT | varchar(10) | FK | Mã máy tính hư hỏng |

* ***Bảng PhongThucHanh****: lưu thông tin của các phòng thực hành*
* *Bảng lưu trữ dữ liệu:*

|  |  |  |  |
| --- | --- | --- | --- |
| **MaPhong** | **ViTri** | **TinhTrangPhong** | **SoLuongMay** |
| 101B4 | Tầng trệt dãy B4 | Tốt | 40 |

* *Bảng cấu trúc:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | MaPhong | varchar(10) | PK | Mã phòng thực hành |
| 2 | ViTri | nvarchar(50) |  | Vị trí phòng |
| 3 | TinhTrangPhong | nvarchar(20) |  | Tình trạng phòng |
| 4 | SoLuongMay | int |  | Số lượng phòng |

* ***Bảng TaiKhoan****: lưu thông tin tài khoản người dùng trong hệ thống*
* *Bảng lưu trữ dữ liệu:*

|  |  |  |
| --- | --- | --- |
| **TenTaiKhoan** | **MatKhau** | **QuyenDN** |
| CB001 | 4QrcOUm6Wau+VuBX8g+IPg== | 0 |

* *Bảng cấu trúc:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | TenTaiKhoan | varchar(10) | PK | Tên tài khoản |
| 2 | MatKhau | varchar(100) |  | Mật khẩu |
| 3 | QuyenDN | int |  | Quyền đăng nhập |

* + 1. **Các ràng buộc toàn vẹn trên dữ liệu**
* **Ràng buộc tham chiếu**

**CanBo** (MaCB) **→ PhieuDangKy** (MaCB)

**CanBo** (MaCB) **→ PhieuThongKe** (MaCB)

**Khoa** (MaKhoa) **→ CanBo** (MaKhoa)

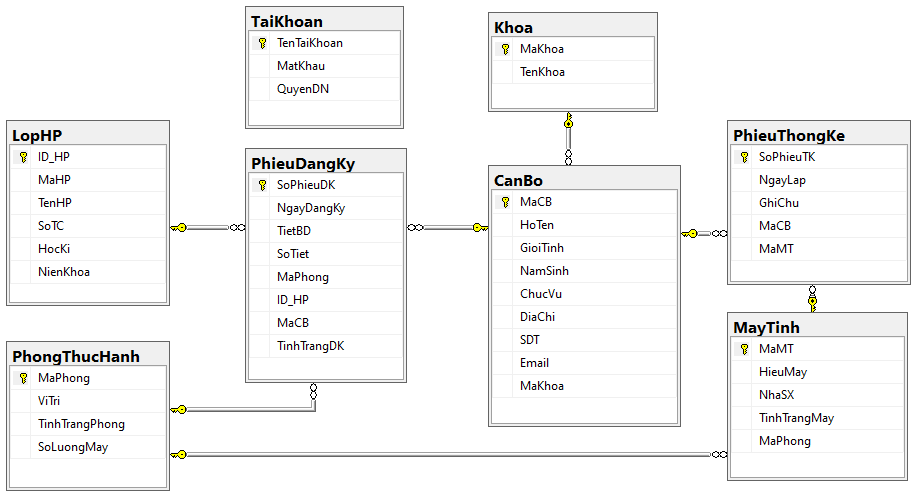
**LopHP** (ID\_HP) **→ PhieuDangKy** (ID\_HP)

**MayTinh** (MaMT) **→ PhieuThongKe** (MaMT)

**PhongThucHanh** (MaPhong) **→ MayTinh** (MaPhong)

**PhongThucHanh** (MaPhong) **→ PhieuDangKy** (MaPhong)

* **Ràng buộc miền giá trị**
* Không cho phép trùng mã.
* Các thuộc tính chỉ cho phép nhập số thì không được có chữ cái hoặc ký tự khác.
* **Các ràng buộc dữ liệu khác**
* Mỗi cán bộ chỉ thuộc một khoa
* Mỗi phiếu đăng ký tương ứng với một cán bộ, một học phần và một phòng thực hành.
* Mỗi phiếu thống kê tương ứng với một cán bộ và một máy tính.
* Mỗi máy tính chỉ thuộc một phòng thực hành.
  + 1. **Diagram**

****

Hình 3.1 Diagram cơ sở dữ liệu

## **3.2 Công cụ sử dụng xây dựng demo**

Để xây dựng chương trình Quản lý đăng ký phòng thực hành DThU, nhóm đã sử dụng một số công cụ sau:

* Công cụ StartUML để mô hình hóa sơ đồ hệ thống
* Hệ quản trị cơ sở dữ liệu SQL Server 2019 để thiết kế các bảng dữ liệu
* Phần mềm Visual Studio 2017/2019 để lập trình các chức năng hệ thống.

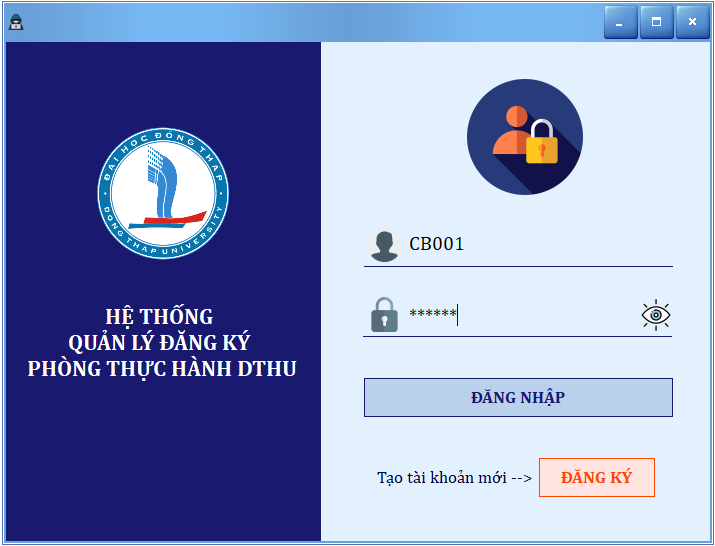
## **Một số giao diện chính của chương trình**

### **Giao diện Đăng nhập**

Bất kì hệ thống quản lý nào thì cũng đều cần phải có giao diện đăng nhập để có thể xác định được người sử dụng là Admin hay Users.

Khi Admin hoặc Users nhập đúng mật khẩu thì sẽ truy cập vào được giao diện chính của chương trình. Ngược lại sẽ báo cho người sử dụng biết là mật khẩu hoặc tên đăng nhập không chính xác.

Form đăng nhập giúp độ bảo mật về cơ sở dữ liệu được tăng lên, tránh và hạn chế tối đa các trường hợp sửa chữa làm sai dữ liệu gốc ban đầu.



Hình 3.2 Giao diện Đăng nhập

Ở đề tài “Quản lý đăng ký phòng thực hành DThU”, sẽ được phân quyền theo 2 loại tài khoản:

* Admin: CB001 (mật khẩu: 123456) sẽ có quyền cao nhất để quản lý tất cả thông tin của hệ thống.
* Users: những tài khoản tạo mới từ giảng viên để sử dụng các chức năng chính như đăng ký, cập nhật tình trạng máy tính hư hỏng,…Hạn chế một số quyền quản lý thông tin của hệ thống.
* ***Sử dụng mã hóa MD5 cho mật khẩu:***

Tại trường mật khẩu, sử dụng mã hóa MD5 để mã hóa mật khẩu của Admin và Users giúp tăng tính bảo mật. Mỗi tài khoản chỉ có người sử dụng mới biết được mật khẩu của mình, vì khi lưu vào cơ sở dữ liệu sẽ là giá trị đã mã hóa.

Khi đăng nhập, mật khẩu nhập vào sẽ được hệ thống mã hóa và lấy giá trị mã hóa so sánh với kết quả đã được lưu trong cơ sở dữ liệu. Nếu trùng khớp, hệ thống sẽ thông báo đăng nhập thành công.

* Ngoài ra, khi Users chưa có tài khoản có thể nhấn *Đăng Ký* để chuyển sang giao diện “Đăng ký tài khoản”.
  + 1. **Giao diện Đăng ký tài khoản**

Tại giao diện Đăng ký tài khoản, Users có thể tạo tài khoản mới ở đây bằng cách cung cấp tên tài khoản, mật khẩu và nhập lại mật khẩu. Sau khi hoàn thành thông tin nhấn *Đăng ký tài khoản* để tiến hành xác nhập tài khoản. Đồng thời, nhấn nút *Đăng nhập* để trở lại giao diện “Đăng nhập”.

****

Hình 3.3 Giao diện Đăng ký tài khoản

* + 1. **Giao diện chính**

Khi đăng nhập thành công hệ thống sẽ kiểm tra tài khoản đăng nhập là Admin hay là Users để hiển thị giao diện phù hợp.

Admin có đầy đủ các quyền truy cập và sửa chữa cơ sở dữ liệu. Còn đối với Users chỉ được quyền Tra cứu các thông tin đăng ký, thông tin máy tính hư hỏng, đăng ký phòng thực hành và thống kê tình trạng máy tính hư hỏng.

****

Hình 3.4 Giao diện màn hình chính

* + 1. **Giao diện chức năng** 
       1. **Giao diện Đổi mật khẩu**

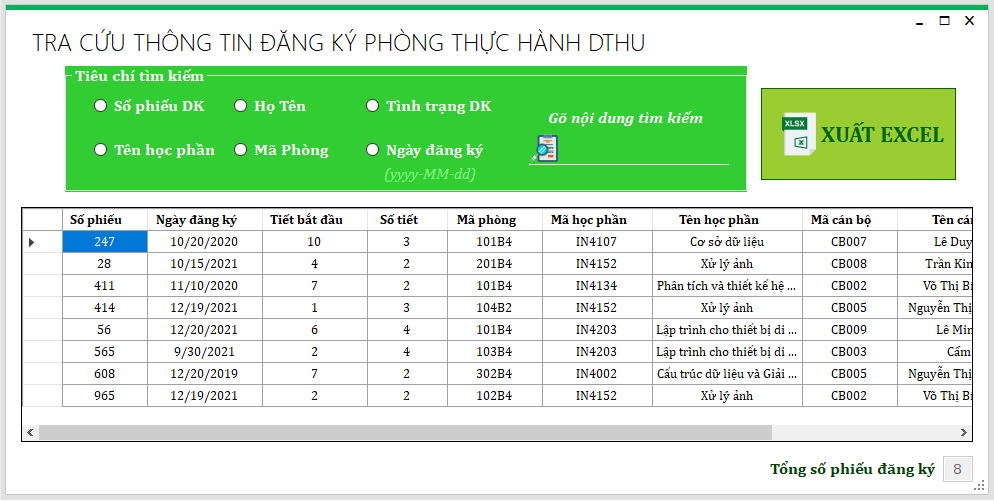
Tại giao diện Đổi mật khẩu, Admin hoặc Users được quyền đổi mật khẩu bằng cách nhập mật khẩu cũ, mật khẩu mới và nhập lại mật khẩu mới; nếu trùng khớp hệ thống sẽ thông báo “Đổi mật khẩu thành công!” ngược lại sẽ hiển thị các lỗi “Mật khẩu nhập lại không trùng khớp!” hoặc “Đổi mật khẩu thất bại!”.

****

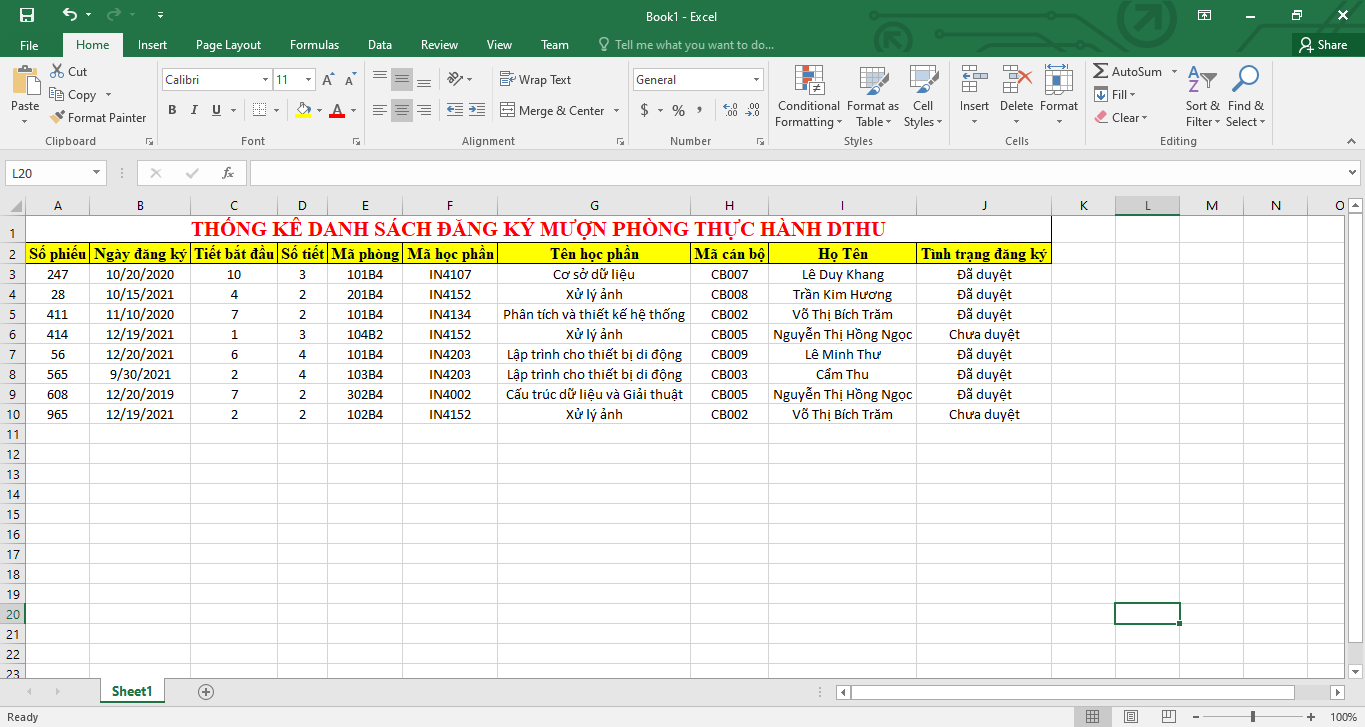
Hình 3.5 Giao diện Đổi mật khẩu

* + - 1. **Giao diện Tra cứu thông tin đăng ký Phòng thực hành**

Tại giao diện “Tra cứu thông tin đăng ký phòng thực hành”, Admin hoặc Users có thể tìm kiếm thông tin theo các tiêu chí: số phiếu đăng ký, họ tên, tình trạng đăng ký, tên học phần, mã phòng và ngày đăng ký. Ngoài ra, khi Admin hoặc Users muốn lưu trữ dữ liệu dưới dạng file Excel có thể nhấn *Xuất Excel* (như hình 3.7). Để mở form, tại màn hình chính nhấn vào *TRA CỨU* chọn *Thông tin đăng ký PTH.*

****

Hình 3.6 Giao diện Tra cứu thông tin đăng ký phòng thực hành DThU

****

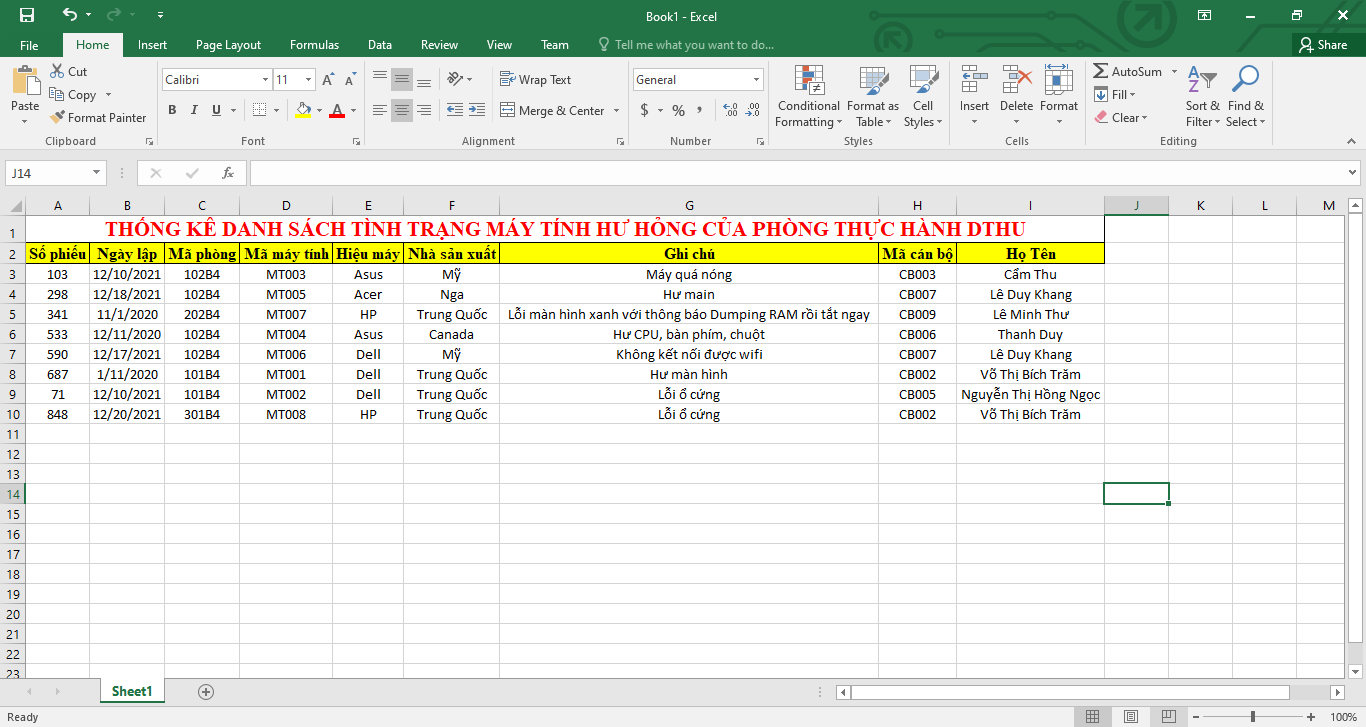
Hình 3.7 Giao diện Xuất Excel danh sách đăng ký phòng thực hành

* + - 1. **Giao diện Tra cứu thông tin máy tính hư hỏng**

Tại giao diện “Tra cứu thông tin máy tính hư hỏng”, Admin hoặc Users có thể tìm kiếm thông tin theo các tiêu chí: số phiếu thống kê, mã phòng, mã máy tính, ngày lập, họ tên cán bộ lập. Ngoài ra, khi Admin hoặc Users muốn lưu trữ dữ liệu dưới dạng file Excel có thể nhấn *Xuất Excel* (như hình 3.9). Để mở form, tại màn hình chính nhấn vào *TRA CỨU* chọn *Thông tin máy tính hư hỏng.*

****

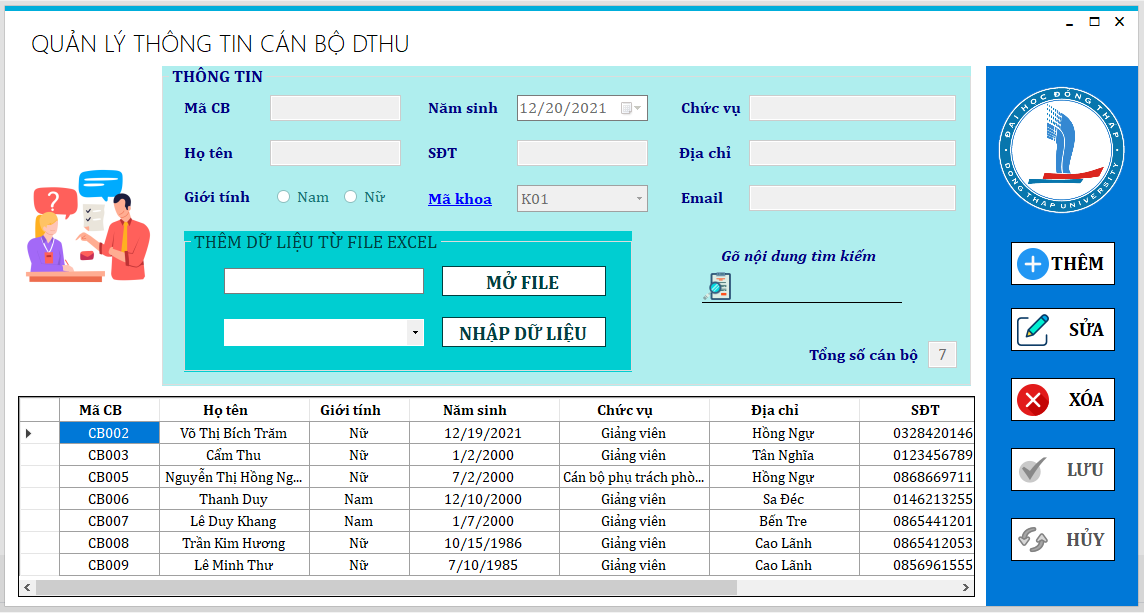
Hình 3.8 Giao diện Tra cứu thông tin máy tính hư hỏng

****

Hình 3.9 Giao diện Xuất Excel danh sách tình trạng máy tính hư hỏng

**3.3.4.4 Giao diện Quản lý cán bộ**

Tại form “Quản lý cán bộ”, chỉ Admin có quyền truy cập và cập nhật dữ liệu. Ở màn hình chính nhấn vào *DANH MỤC* chọn *Quản lý cán bộ*.

****

Hình 3.10 Giao diện Quản lý thông cán bộ DThU

Form “Quản lý cán bộ” sử dụng dữ liệu của bảng dữ liệu Khoa để load giá trị MaKhoa vào ComboBox và lưu trữ dữ liệu vào bảng CanBo. Ngoài ra, người sử dụng có thể nhấn vào link Mã khoa để mở form “Khoa” *(như hình 3.11)* nhằm xem chi tiết thông tin các khoa trong trường.

Chức năng của giao diện: Cập nhật thông tin các cán bộ công tác tại trường Đại học Đồng Tháp.

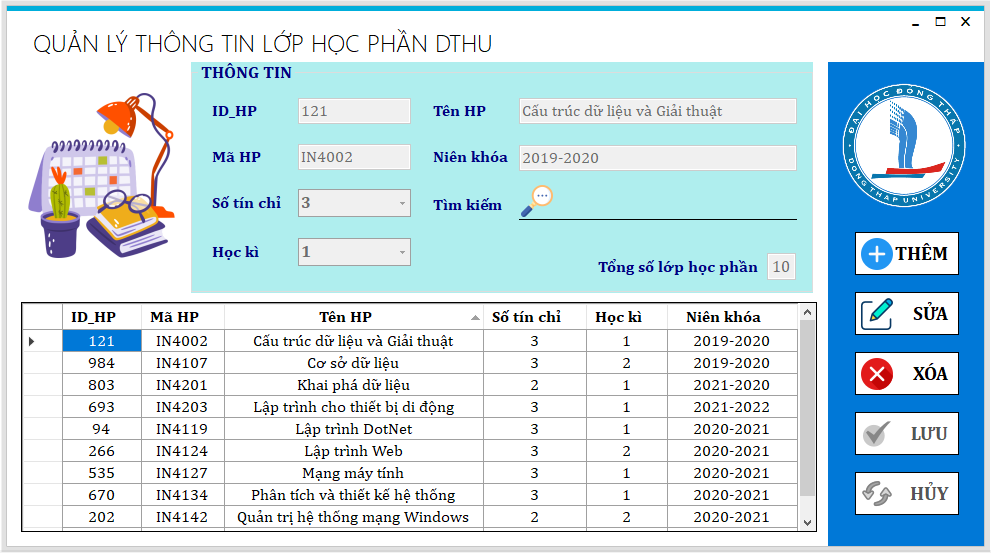
* **Thêm:** thêm thông tin cán bộ. Trong đó có các thông tin về Mã cán bộ, họ tên, giới tính, năm sinh, số điện thoại, mã khoa, chức vụ, địa chỉ và email. Khi nhấn nút *Thêm* hệ thống sẽ mở phần nhập dữ liệu, đồng thời khóa các nút Thêm, Sửa và Xóa.
* **Sửa:** dùng chuẩn bị trường cần thiết để sửa khi có sự sai sót hoặc muốn cập nhật thêm thông tin cán bộ. (Lưu ý: chọn vào cán bộ bên danh sách để hiển thị thông tin cán bộ cần sửa).
* **Xoá:** xoá toàn bộ thông tin của cán bộ đã chọn bên danh sách.
* **Lưu:** dùng để lưu thông tin cán bộ khi thêm hoặc sửa.
* **Hủy:** làm trống các trường đã nhập vào các TextBox, đồng thời mở các nút Thêm, Sửa và Xóa.
* **Tìm kiếm:** Cho phép tìm kiếm theo mã cán bộ, họ tên và mã khoa.
* **Thêm dữ liệu từ file excel:** cho phép nhập dữ liệu vào bằng file excel.
* **Duyệt dữ liệu:** dùng để mở đường dẫn đến tệp excel cần đọc.
* **Nhập dữ liệu:** dùng để thêm dữ liệu từ datagridview vào cơ sở dữ liệu.

****

Hình 3.11 Giao diện Danh sách Khoa, phòng ban DThU

* + - 1. **Giao diện Quản lý lớp học phần**

Tại form “Quản lý lớp học phần”, chỉ Admin có quyền truy cập và cập nhật dữ liệu. Ở màn hình chính nhấn vào *DANH MỤC* chọn *Quản lý lớp học phần*.

****

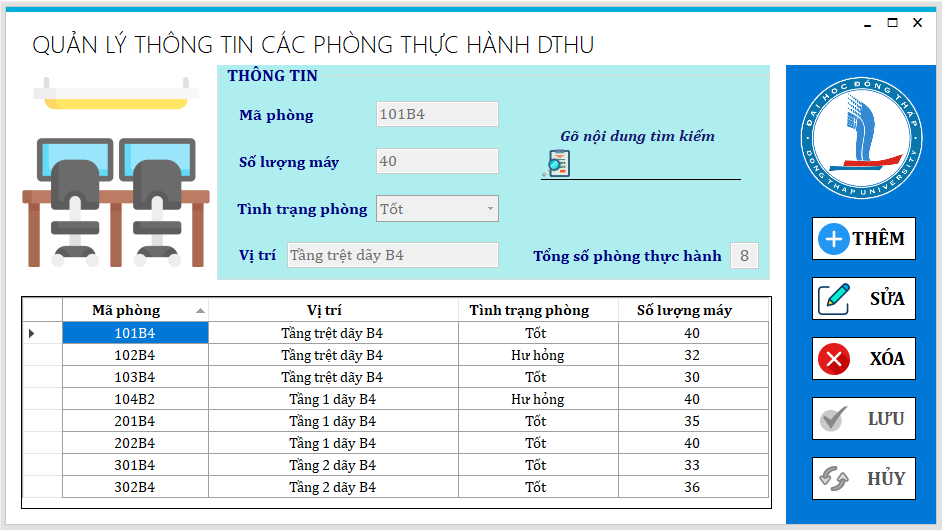
Hình 3.12 Giao diện Quản lý thông tin lớp học phần DThU

Form “Quản lý lớp học phần” sử dụng bảng dữ liệu LopHP để lưu trữ dữ liệu.

Chức năng của giao diện: Cập nhật thông tin các lớp học phần giảng dạy tại trường Đại học Đồng Tháp.

* **Thêm:** thêm thông tin học phần. Trong đó có các thông tin về ID\_HP, mã học phần, tên học phần, số tín chỉ, học kì và niên khóa. Khi nhấn nút *Thêm* hệ thống sẽ mở phần nhập dữ liệu, đồng thời khóa các nút Thêm, Sửa và Xóa.
* **Sửa:** dùng chuẩn bị trường cần thiết để sửa khi có sự sai sót hoặc muốn cập nhật thêm thông tin học phần. (Lưu ý: chọn vào học phần bên danh sách để hiển thị thông tin học phần cần sửa).
* **Xoá:** xoá toàn bộ thông tin của học phần đã chọn bên danh sách.
* **Lưu:** dùng để lưu thông tin học phần khi thêm hoặc sửa.
* **Hủy:** làm trống các trường đã nhập vào các TextBox, đồng thời mở các nút Thêm, Sửa và Xóa.
* **Tìm kiếm:** Cho phép tìm kiếm theo ID\_HP, mã học phần, tên học phần, số tín chỉ, học kì và niên khóa
  + - 1. **Giao diện Quản lý phòng thực hành**

Tại form “Quản lý phòng thực hành”, chỉ Admin có quyền truy cập và cập nhật dữ liệu. Ở màn hình chính nhấn vào *DANH MỤC* chọn *Quản lý phòng thực hành*.

****

Hình 3.13 Giao diện Quản lý thông tin các phòng thực hành DThU

Form “Quản lý phòng thực hành” sử dụng bảng dữ liệu PhongThucHanh để lưu trữ dữ liệu.

Chức năng của giao diện: Cập nhật thông tin phòng thực hành máy tính tại trường Đại học Đồng Tháp.

* **Thêm:** thêm thông tin phòng thực hành. Trong đó có các thông tin về mã phòng, số lượng máy, tình trạng phòng và vị trí. Khi nhấn nút *Thêm* hệ thống sẽ mở phần nhập dữ liệu, đồng thời khóa các nút Thêm, Sửa và Xóa.
* **Sửa:** dùng chuẩn bị trường cần thiết để sửa khi có sự sai sót hoặc muốn cập nhật thêm thông tin phòng thực hành. (Lưu ý: chọn vào phòng thực hành bên danh sách để hiển thị thông tin phòng thực hành cần sửa).
* **Xoá:** xoá toàn bộ thông tin của phòng thực hành đã chọn bên danh sách.
* **Lưu:** dùng để lưu thông tin phòng thực hành khi thêm hoặc sửa.
* **Hủy:** làm trống các trường đã nhập vào các TextBox, đồng thời mở các nút Thêm, Sửa và Xóa.
* **Tìm kiếm:** Cho phép tìm kiếm theo mã phòng.
  + - 1. **Giao diện Quản lý máy tính**

Tại form “Quản lý máy tính”, chỉ Admin có quyền truy cập và cập nhật dữ liệu. Ở màn hình chính nhấn vào *DANH MỤC* chọn *Quản lý máy tính*.

****

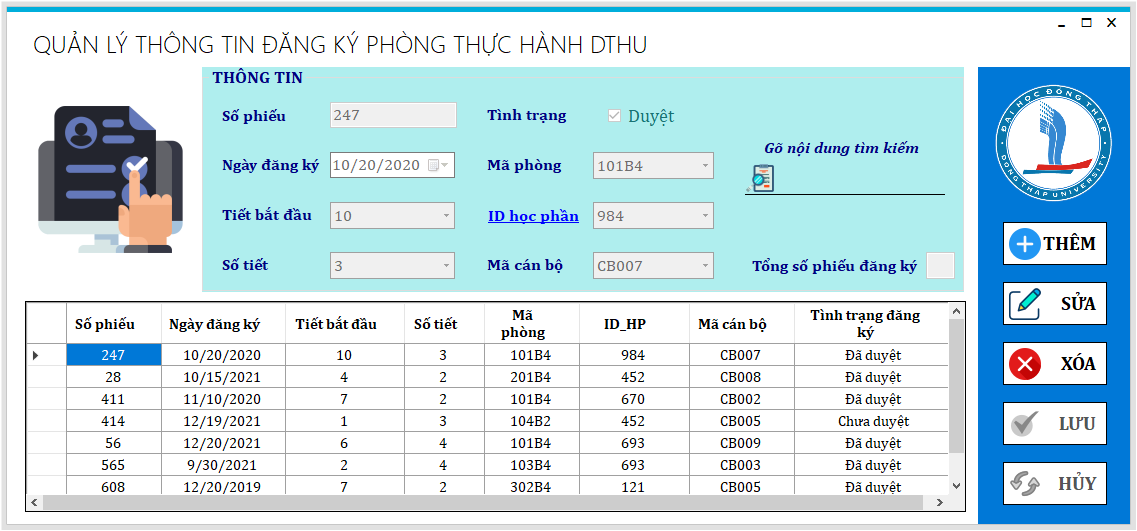
Hình 3.14 Giao diện Quản lý thông tin máy tính của các phòng thực hành

Form “Quản lý máy tính” sử dụng dữ liệu của bảng PhongThucHanh để load vào ComBoBox giá trị MaPhong, đồng thời sử dụng bảng dữ liệu MayTinh để lưu trữ dữ liệu.

Chức năng của giao diện: Cập nhật thông tin máy tính thuộc các phòng thực hành tại trường Đại học Đồng Tháp.

* **Thêm:** thêm thông tin máy tính. Trong đó có các thông tin về mã máy, hiệu máy, nhà sản xuất, tình trạng máy và mã phòng. Khi nhấn nút *Thêm* hệ thống sẽ mở phần nhập dữ liệu, đồng thời khóa các nút Thêm, Sửa và Xóa.
* **Sửa:** dùng chuẩn bị trường cần thiết để sửa khi có sự sai sót hoặc muốn cập nhật thêm thông tin máy tính. (Lưu ý: chọn vào máy tính bên danh sách để hiển thị thông tin máy tính cần sửa).
* **Xoá:** xoá toàn bộ thông tin của máy tính đã chọn bên danh sách.
* **Lưu:** dùng để lưu thông tin máy tính khi thêm hoặc sửa.
* **Hủy:** làm trống các trường đã nhập vào các TextBox, đồng thời mở các nút Thêm, Sửa và Xóa.
* **Tìm kiếm:** Cho phép tìm kiếm theo mã máy tính và mã phòng.
* **Thêm dữ liệu từ file excel:** cho phép nhập dữ liệu vào bằng file excel.
* **Duyệt dữ liệu:** dùng để mở đường dẫn đến tệp excel cần đọc.
* **Nhập dữ liệu:** dùng để thêm dữ liệu từ datagridview vào cơ sở dữ liệu.
  + - 1. **Giao diện Quản lý đăng ký phòng thực hành**

Tại form “Quản lý đăng ký phòng thực hành”, chỉ Admin có quyền truy cập và cập nhật dữ liệu. Ở màn hình chính nhấn vào *DANH MỤC* chọn *Quản lý đăng ký PTH*.

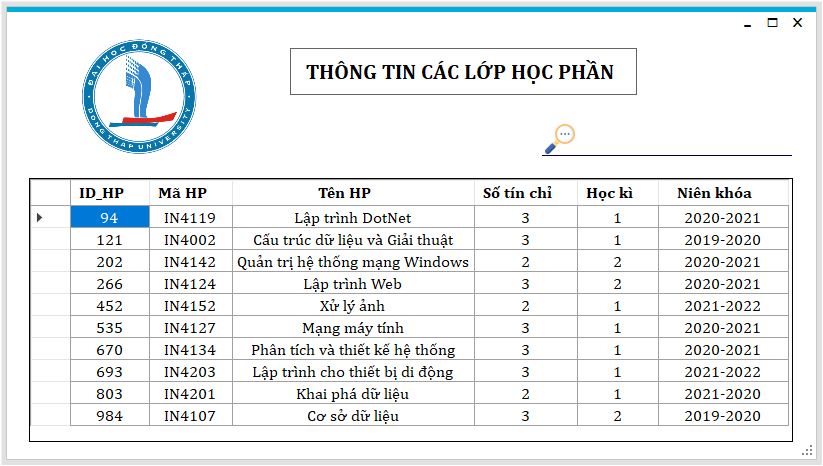
****

Hình 3.15 Giao diện Quản lý thông tin đăng ký phòng thực hành DThU

Form “Quản lý thông tin đăng ký phòng thực hành DThU” sử dụng dữ liệu của bảng dữ liệu CanBo để load giá trị MaCB, bảng dữ liệu LopHP để load giá trị ID\_HP và bảng dữ liệu PhongThucHanh để load giá trị MaPhong vào ComboBox và lưu trữ dữ liệu vào bảng PhieuDangKy. Ngoài ra, người sử dụng có thể nhấn vào link ID học phần để mở form “HocPhan” *(như hình 3.16)* nhằm xem chi tiết thông tin các lớp học phần.

Chức năng của giao diện: Cập nhật (duyệt) thông tin đăng ký phòng thực hành tại trường Đại học Đồng Tháp.

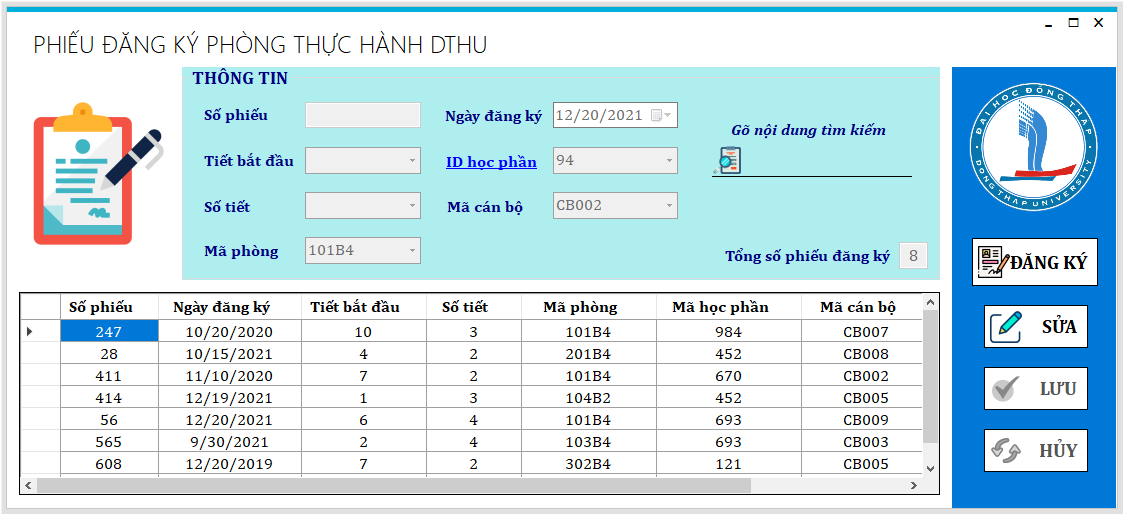
* **Thêm:** thêm thông tin đăng ký phòng thực hành. Trong đó có các thông tin về số phiếu đăng ký, ngày đăng ký, tiết bắt đầu, số tiết, tình trạng đăng ký, mã phòng, ID học phần và mã cán bộ. Khi nhấn nút *Thêm* hệ thống sẽ mở phần nhập dữ liệu, đồng thời khóa các nút Thêm, Sửa và Xóa.
* **Sửa:** dùng chuẩn bị trường cần thiết để sửa khi có sự sai sót hoặc muốn cập nhật thêm thông tin đăng ký phòng thực hành. (Lưu ý: chọn vào phiếu đăng ký bên danh sách để hiển thị thông tin đăng ký phòng thực hành cần sửa).
* **Xoá:** xoá toàn bộ thông tin của phiếu đăng ký đã chọn bên danh sách.
* **Lưu:** dùng để lưu thông tin phiếu đăng ký khi thêm hoặc sửa.
* **Hủy:** làm trống các trường đã nhập vào các TextBox, đồng thời mở các nút Thêm, Sửa và Xóa.
* **Tìm kiếm:** Cho phép tìm kiếm theo số phiếu đăng ký, mã phòng và ngày đăng ký.

****

Hình 3.16 Giao diện Thông tin các lớp học phần

* + - 1. **Giao diện Đăng ký phòng thực hành**

Tại form “Đăng ký phòng thực hành”, chỉ Users có quyền truy cập và cập nhật dữ liệu. Ở màn hình chính nhấn vào *DANH MỤC* chọn *Đăng ký Phòng thực hành*.

****

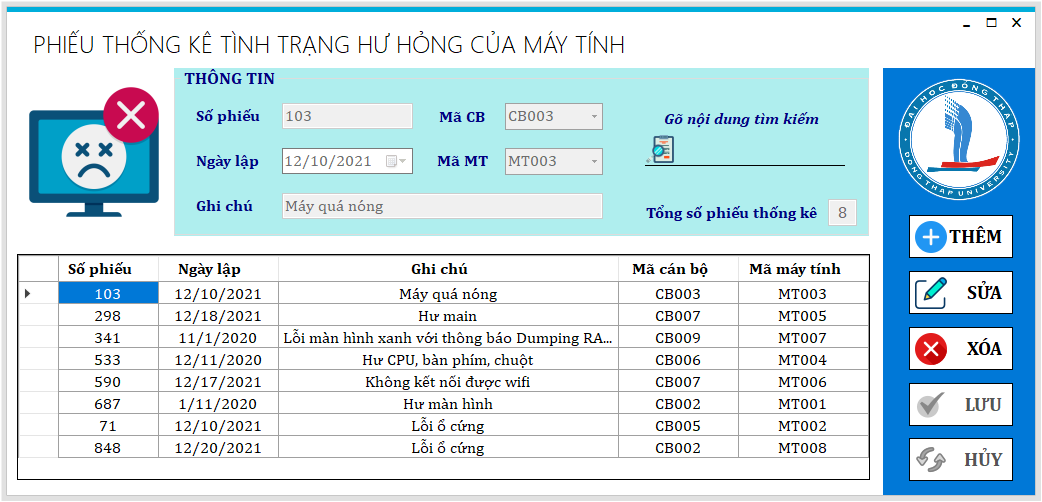
Hình 3.17 Giao diện Phiếu đăng ký phòng thực hành DThU

Form “Đăng ký phòng thực hành DThU” sử dụng dữ liệu của bảng dữ liệu CanBo để load giá trị MaCB, bảng dữ liệu LopHP để load giá trị ID\_HP và bảng dữ liệu PhongThucHanh để load giá trị MaPhong (với điều kiện TinhTrangPhong là Tốt) vào ComboBox và lưu trữ dữ liệu vào bảng PhieuDangKy. Ngoài ra, người sử dụng có thể nhấn vào link ID học phần để mở form “HocPhan” *(như hình 3.16)* nhằm xem chi tiết thông tin các lớp học phần.

Chức năng của giao diện: Đăng ký phòng thực hành tại trường Đại học Đồng Tháp.

* **Đăng ký:** đăng ký thông tin đăng ký phòng thực hành. Trong đó có các thông tin về số phiếu đăng ký, ngày đăng ký, tiết bắt đầu, số tiết, mã phòng, ID học phần và mã cán bộ. Khi nhấn nút *Đăng ký* hệ thống sẽ mở phần nhập dữ liệu, đồng thời khóa các nút Đăng ký và Sửa.
* **Sửa:** dùng chuẩn bị trường cần thiết để sửa khi có sự sai sót hoặc muốn cập nhật thêm thông tin đăng ký phòng thực hành. (Lưu ý: chọn vào phiếu đăng ký bên danh sách để hiển thị thông tin đăng ký phòng thực hành cần sửa).
* **Lưu:** dùng để lưu thông tin phiếu đăng ký khi đăng ký hoặc sửa.
* **Hủy:** làm trống các trường đã nhập vào các TextBox, đồng thời mở các nút Đăng ký và Sửa.
* **Tìm kiếm:** Cho phép tìm kiếm theo số phiếu đăng ký, mã phòng và ngày đăng ký.
  + - 1. **Giao diện Phiếu thống kê tình trạng hư hỏng của máy tính**

Tại form “Phiếu thống kê tình trạng hư hỏng của máy tính”, Admin và Users có quyền truy cập và cập nhật dữ liệu. Ở màn hình chính nhấn vào *BÁO CÁO THỐNG KÊ* chọn *Phiếu thống kê máy tính hư hỏng*.

****

Hình 3.18 Giao diện Thống kê tình trạng hư hỏng của máy tính

Form “Phiếu thống kê tình trạng hư hỏng của máy tính” sử dụng dữ liệu của bảng dữ liệu CanBo để load giá trị MaCB và bảng dữ liệu MayTinh để load giá trị MaMT (với điều kiện TinhTrangMay là Hư hỏng) vào ComboBox và lưu trữ dữ liệu vào bảng PhieuThongKe.

Chức năng của giao diện: Cập nhật thông tin máy tính hư hỏng trong phòng thực hành tại trường Đại học Đồng Tháp.

* **Thêm:** thêm thông tin tình trạng máy tính hư hỏng của phòng thực hành. Trong đó có các thông tin về số phiếu thống kê, ngày lập, ghi chú, mã cán bộ và mã máy tính. Khi nhấn nút *Thêm* hệ thống sẽ mở phần nhập dữ liệu, đồng thời khóa các nút Thêm, Sửa và Xóa.
* **Sửa:** dùng chuẩn bị trường cần thiết để sửa khi có sự sai sót hoặc muốn cập nhật thêm thông tin phiếu thống kê tình trạng hư hỏng của máy tính. (Lưu ý: chọn vào phiếu thống kê bên danh sách để hiển thị thông tin tình trạng của máy tính hư hỏng cần sửa).
* **Xoá:** xoá toàn bộ thông tin của phiếu thống kê đã chọn bên danh sách.
* **Lưu:** dùng để lưu thông tin phiếu thống kê khi thêm hoặc sửa.
* **Hủy:** làm trống các trường đã nhập vào các TextBox, đồng thời mở các nút Thêm, Sửa và Xóa.
* **Tìm kiếm:** Cho phép tìm kiếm theo số phiếu thống kê, mã cán bộ, mã máy tính và ngày lập.
  + - 1. **Giao diện Báo cáo danh sách đăng ký phòng thực hành**

Tại form “Báo cáo danh sách đăng ký phòng thực hành DThU”, chỉ Admin có quyền truy cập và xuất báo cáo. Ở màn hình chính nhấn vào *BÁO CÁO THỐNG KÊ* chọn *Báo cáo danh sách đăng ký* PTH.

****

Hình 3.19 Giao diện Báo cáo danh sách đăng ký phòng thực hành DThU

Form sử dụng bảng dữ liệu PhieuDangKy liên kết với bảng CanBo, bảng Khoa và LopHocPhan để lấy tất cả thông tin chi tiết của mỗi phiếu đăng ký theo mốc thời gian nhất định (thời gian bắt đầu và thời gian kết thúc).

Chức năng của giao diện: chọn mốc thời gian cần thống kê danh sách đăng ký phòng thực hành để tạo báo cáo, nhấn “XUẤT BÁO CÁO”.

# **PHẦN BA: KẾT LUẬN**

* + 1. **Kết quả đạt được**
* Phân quyền đăng nhập Hệ thống:
* Admin (cán bộ quản lý phòng thực hành)
* Quản lý cán bộ
* Quản lý lớp học phần
* Quản lý phòng thực hành
* Quản lý máy tính
* Quản lý đăng ký phòng thực hành
* Thống kê tình trạng hư hỏng của máy tính
* Báo cáo danh sách đăng ký phòng thực hành
* Tra cứu thông tin Đăng ký phòng thực hành
* Tra cứu thông tin Máy tính hư hỏng
* Đổi mật khẩu
* Users (giảng viên)
* Đăng ký phòng thực hành
* Thống kê máy tính hư hỏng
* Tra cứu thông tin Đăng ký phòng thực hành
* Đổi mật khẩu
* Chung: mỗi Form có chức năng thêm, sửa, xóa và tìm kiếm đa dạng. Mặc khác, có bắt lỗi người dùng tương ứng theo từng form.
* Ngoài ra ở form Cán bộ có hỗ trợ nhập dữ liệu từ file excel, các form thuộc danh mục “Tra cứu” cho phép xuất excel.
* Sử dụng ReportViewer để xuất báo cáo danh sách đăng ký phòng thực hành DThU theo mốc thời gian nhất định.
* Sử dụng Framework của Metro và DevExpress để thiết kế giao diện đẹp mắt, phù hợp và dễ sử dụng.
* Sử dụng MD5 mã hóa mật khẩu để tăng tính bảo mật hơn.
* Hiểu rõ quy trình quản lý đăng ký phòng thực hành DThU, phân tích cơ sở dữ liệu và cụ thể hóa bằng những sơ đồ vẽ bằng công cụ StarUML.
* Đóng gói phần mềm dựa vào Setup do Visual Studio hỗ trợ sẳn để có thể dễ dàng triển khai khi cần thiết.
  + 1. **Hạn chế**
* Còn giới hạn một số chức năng.
* Hệ thống có thể chưa tối ưu vì kiến thức còn hạn chế.
  + 1. **Hướng phát triển**
* Mở rộng triển khai ý tưởng cho phép quản lý cụ thể và chặt chẽ hơn.
* Sẽ nghiên cứu để phát triển thêm nhiều chức năng hỗ trợ công tác đăng ký phòng thực hành được tối ưu, hiệu quả hơn.
* Cập nhật giao diện đẹp hơn để tăng tính trải nghiệm cho người dùng.

# **TÀI LIỆU THAM KHẢO**

**[1] Nguyễn Thị Mỹ Dung**, bài giảng *“Phân tích hệ thống hướng đối tượng”*, Đại học Đồng Tháp.

**[2] TS. Phạm Nguyễn Cương, TS Hồ Tường Vinh**, Giáo trình “PT&TK HT HĐT sử dụng UML”, 2013, ĐHKHTN TP HCM.

**[3]** **Huỳnh Lê Uyên Minh, Trần Kim Hương**; *Bài giảng “Lập trình DOTNET”;* Trường Đại học Đồng Tháp; 2019.

**[4]** **Trần Kim Hương**, *Slide bài giảng “Phát triển phần mềm hướng đối tượng”.*

# **PHỤ LỤC**

## **Hướng dẫn sử dụng**

**Bước 1**: Mở file “QLPTH.sql” bằng SQL Server 2019 để nạp cơ sở dữ liệu vào máy.

**Bước 2:** Giải nén file vừa tải và mở file “BTL\_QLPTH.sln” bằng Visual Studio 2017/2019

(Chú ý sửa chuỗi kết nối dữ liệu nếu không trùng khớp)

* **Lưu ý**: tên tài khoản và mật khẩu để sử dụng hệ thống:
* **Tài khoản:** CB001/ mật khẩu: 123456 (admin)
* **Tài khoản:** CB002/ mật khẩu: 151515 (users)
* **…**

## **Bảng phân công công việc**

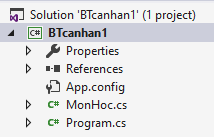
|  |  |
| --- | --- |
| **Họ tên sinh viên** | **Nội dung công việc thực hiện** |
| 0018413046 - Nguyễn Thị Hồng Ngọc | * Nghiên cứu và phân tích đề tài * Thiết kế dữ liệu và vẽ các sơ đồ * Thiết kế giao diện * Viết code * Soạn word * Hoàn chỉnh demo, sửa lỗi phát sinh.   (*Ghi chú: nhóm cùng phối hợp*  *trong từng nội dung*) |
| 0018413058 - Võ Thị Bích Trăm |

**BÀI TẬP CÁ NHÂN – Nguyễn Thị Hồng Ngọc**

**Bài tập chương 2 \_ P1: Lập trình hướng đối tượng**

1. *Bài 4 (bt cá nhân) Thiết kế class Môn học với các dữ liệu cần lưu là: mã môn, tên môn, số tín chỉ.*

Khởi tạo chương trình, tạo một lớp MonHoc:



* **Một số hàm xử lý tiêu biểu**
* *Phương thức khởi tạo không có tham số và phương thức khởi tạo có tham số*

//phương thức khởi tạo CÓ THAM SỐ

public MonHoc (string ma, string ten, int tc)

{

MaMH = ma;

TenMH = ten;

SoTC = tc;

}

//phương thức khởi tạo KHÔNG CÓ THAM SỐ

public void Nhap()

{

Console.Write("Nhap MaMH: ");

MaMH = Console.ReadLine();

Console.Write("Nhap TenMH: ");

TenMH = Console.ReadLine();

Console.Write("Nhap So tin chi: ");

SoTC = Convert.ToInt32(Console.ReadLine());

}

* *Hàm tính số tiết*

public double TinhSoTiet() {

double Sotiet = SoTC \* 22.5;

return Sotiet;

}

* *Hàm xuất thông tin ra màn hình*

public void Xuat()

{

Console.WriteLine("Ma mon hoc: {0} \tTen mon hoc: {1} \tSo tin chi: {2} \tTong so tiet: {3}", MaMH, TenMH, SoTC, TinhSoTiet());

}

* *Hàm sắp xếp giảm dần theo số tín chỉ*

MonHoc temp;

for (int i = 0; i < n; i++)

{

for (int j = i + 1; j < n; j++)

{

if (x[i].SoTC < x[j].SoTC)

{

temp = x[i];

x[i] = x[j];

x[j] = temp;

}

}

}

* *Hàm tìm môn học theo mã môn học nhập từ bàn phím*

string key;

Console.Write("\n---?----Nhap Ma mon hoc can tim: ");

key = Console.ReadLine();

Console.WriteLine("\nMON HOC CAN TIM LA: ");

for (int i = 0; i < n; i++)

{

if(x[i].MaMH == key)

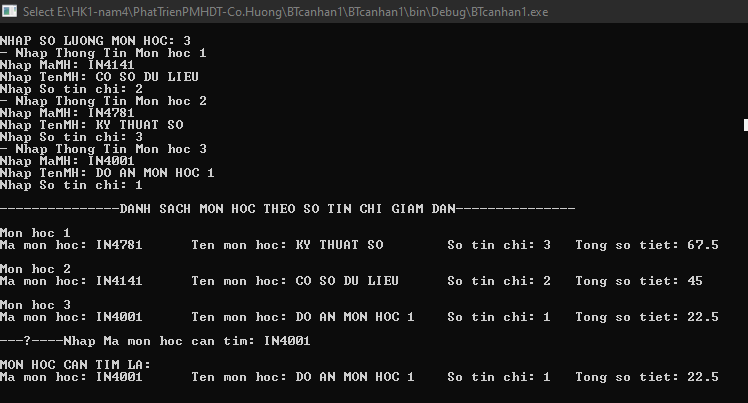
{

x[i].Xuat();

}

}

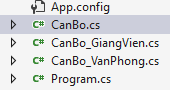
* **Giao diện demo kết quả**



**Bài tập chương 2 \_ P2: Tính kế thừa – đa hình trong HĐT**

1. *Bài 4 (cá nhân) Viết chương trình tính lương cán bộ của một trường học.*

Chương trình sẽ có 4 lớp, trong đó: lớp CanBo\_GiangVien và CanBo\_VanPhong sẽ kế thừa từ lớp CanBo, còn lại là lớp Program



* **Một số hàm xử lý tiêu biểu**
* *Hàm nhập thông tin của lớp CanBo\_VanPhong*

public override void Nhap()

{

Console.Write("Nhap Ma can bo: ");

MaCB = Console.ReadLine();

Console.Write("Nhap Ho ten: ");

HoTen = Console.ReadLine();

Console.Write("Nhap Nam sinh: ");

NamSinh = Console.ReadLine();

Console.Write("Nhap Que Quan: ");

QueQuan = Console.ReadLine();

Console.Write("Nhap Luong co ban: ");

LuongCB = double.Parse(Console.ReadLine());

Console.Write("Nhap He so luong: ");

HSluong = double.Parse(Console.ReadLine());

}

* *Hàm nhập thông tin của lớp CanBo\_GiangVien*

public override void Nhap()

{

Console.Write("Nhap Ma can bo: ");

MaCB = Console.ReadLine();

Console.Write("Nhap Ho ten: ");

HoTen = Console.ReadLine();

Console.Write("Nhap Nam sinh: ");

NamSinh = Console.ReadLine();

Console.Write("Nhap Que Quan: ");

QueQuan = Console.ReadLine();

Console.Write("Nhap Luong co ban: ");

LuongCB = double.Parse(Console.ReadLine());

Console.Write("Nhap He so luong: ");

HSluong = double.Parse(Console.ReadLine());

Console.Write("Nhap Tham nien: ");

ThamNien = Int32.Parse(Console.ReadLine());

}

* *Hàm xuất thông tin:*

public override void Xuat()

{

Console.WriteLine("MaCB: {0} \tHo ten: {1} \tNam sinh: {2} \tQue Quan: {3} \tLuong: {4}", MaCB, HoTen, NamSinh, QueQuan, Luong);

}

* *Hàm tính lương của CanBo\_GiangVien:*

public override void TinhLuong()

{

If (ThamNien > 5)

Luong = (LuongCB \* HSluong) + 0.4 \* LuongCB + 0.1 \* LuongCB;

else

Luong = (LuongCB \* HSluong) + 0.4 \* LuongCB;

}

* *Hàm tính lương của CanBo\_VanPhong:*

public override void TinhLuong()

{

Luong = (LuongCB \* HSluong) + (0.1 \* LuongCB);

}

* *Hàm main:*

static void Main(string[] args)

{

string a;

do

{

Console.WriteLine("\n-----------------\*MENU\*------------------");

Console.WriteLine(" 1. Can bo Van Phong!");

Console.WriteLine(" 2. Giang Vien!");

Console.WriteLine(" 3. Thoat!");

Console.WriteLine("-------------------------------------------------");

Console.Write("Chon Can bo: ");

a = Console.ReadLine();

switch (a)

{

case "1":

Console.Write("\nNhap vao so luong Can bo Van Phong: ");

int n = Convert.ToInt32(Console.ReadLine());

CanBo\_VanPhong[] x = new CanBo\_VanPhong[n];

for (int i = 0; i < n; i++)

{

Console.WriteLine("\nNhap Thong Tin can bo Van phong " + (i + 1));

x[i] = new CanBo\_VanPhong();

x[i].Nhap();

x[i].TinhLuong();

}

//Hàm sắp xếp giảm dần

CanBo\_VanPhong tam;

for (int i = 0; i < n; i++)

{

for(int j = i+1; j<n; j++)

{

if(x[i].Luong < x[j].Luong)

{

tam = x[i];

x[i] = x[j];

x[j] = tam;

}

}

}

Console.WriteLine("\n-------DANH SACH CAN BO VAN PHONG THEO SO TIEN LUONG GIAM DAN-------");

for (int i = 0; i < n; i++)

{

Console.Write("STT " + (i + 1) + "\t");

x[i].Xuat();

}

break;

case "2":

Console.Write("Nhap vao so luong Giang Vien: ");

int m = Convert.ToInt32(Console.ReadLine());

CanBo\_GiangVien[] z = new CanBo\_GiangVien[m];

for (int i = 0; i < m; i++)

{

Console.WriteLine("\nNhap Thong Tin Giang vien "+(i+1));

z[i] = new CanBo\_GiangVien();

z[i].Nhap();

z[i].TinhLuong();

}

//Hàm sắp xếp giảm dần

CanBo\_GiangVien temp;

for (int i = 0; i < m; i++)

{

for (int j = i + 1; j < m; j++)

{

if (z[i].Luong < z[j].Luong)

{

temp = z[i];

z[i] = z[j];

z[j] = temp;

}

}

}

Console.WriteLine("\n-------DANH SACH GIANG VIEN THEO SO TIEN LUONG GIAM DAN-------");

for (int i = 0; i < m; i++)

{

Console.Write("STT " + (i + 1) +"\t");

z[i].Xuat();

}

break;

case "3":

Console.WriteLine("Thoat chuong trinh! Byeee.");

break;

default:

Console.WriteLine("Vui long nhap lai!!!");

break;

}

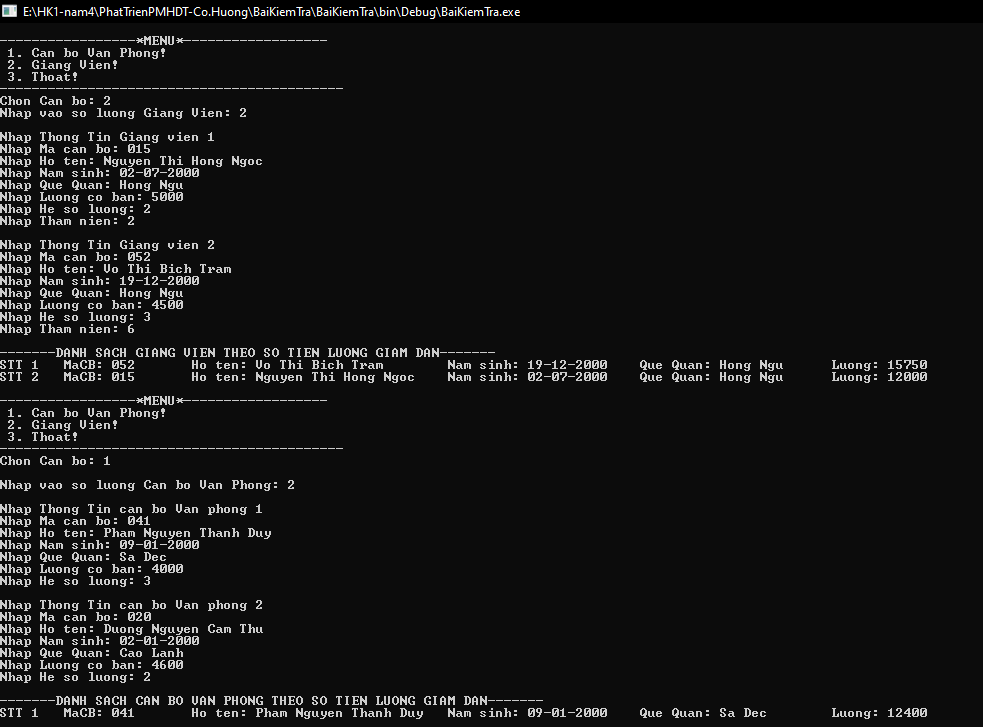
} while (a != "3");

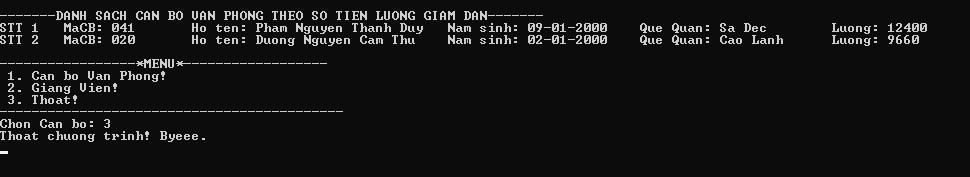
Console.ReadLine();

}

}

* **Giao diện demo kết quả**

****

****

**BÀI TẬP CÁ NHÂN – Võ Thị Bích Trăm**

**Bài tập chương 2 \_ P1: Lập trình hướng đối tượng**

1. *Bài**1: Cài đặt Class Hình Tròn thực hiện việc tính chu vi và diện tích hình tròn.*

Khởi tạo chương trình, tạo một lớp MonHoc:



* **Một số hàm xử lý tiêu biểu**
* *Khai báo các thuộc tính trong chương trình*

public double R, S, CV;

public double PI = 3.1416;

public double bk;

* *Phương thức khởi tạo không có tham số và phương thức khởi tạo có tham số*

//phương thức khởi tạo CÓ THAM SỐ

public HinhTron(double bk)

{

R = bk;

}

//phương thức khởi tạo KHÔNG CÓ THAM SỐ

public void Nhap()

{

Console.Write("Nhap Ban kinh R: ");

bk = Convert.ToDouble(Console.ReadLine());

}

* *Hàm tính Chu vi*

public double ChuVi() {

CV = 2 \* PI \* R;

return CV;

}

* *Hàm tính Diện tích*

public double ChuVi() {

S = PI \* PI \* R;

return S;

}

* *Hàm xuất thông tin ra màn hình*

public void XuatKQ()

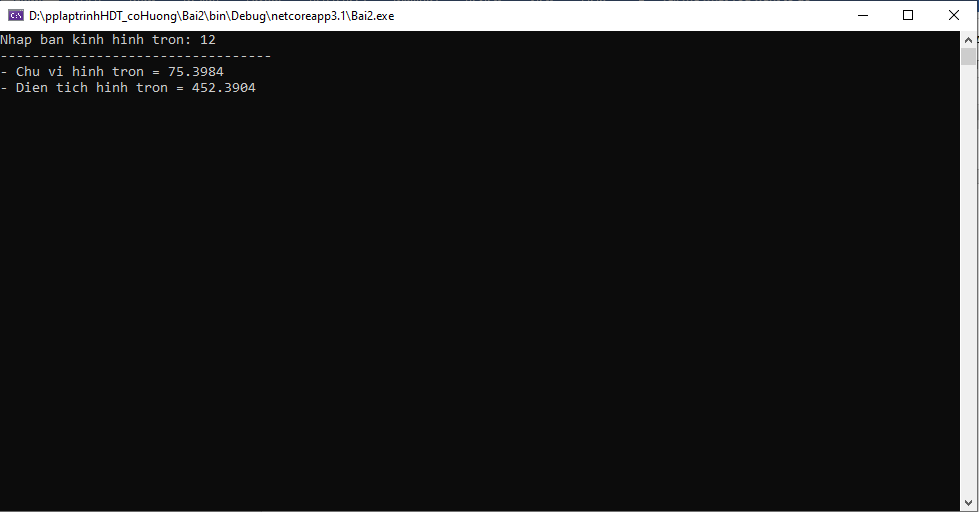
{

Console.WriteLine("Chu vi hinh tron = {0}", CV);

Console.WriteLine("Dien tich hinh tron = {0}", S);

}

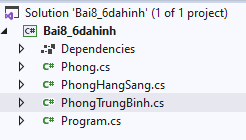
* **Giao diện demo kết quả**



**Bài tập chương 2 \_ P2: Tính kế thừa – đa hình trong HĐT**

1. *Bài 6:* Viết chương trình tính diện tích và chi phí xây dựng căn phòng theo hướng  
   đối tượng, sử dụng tính đa hình.

Chương trình sẽ có 4 lớp, trong đó: lớp PhongHangSang và PhongTrungBinh sẽ kế thừa từ lớp Phong, còn lại là lớp Program



* **Một số hàm xử lý tiêu biểu**
* *Hàm nhập thông tin của lớp PhongHangSang*

public override void Nhap()

{

Console.Write("- Nhap chieu dai can phong: ");

chieudai = double.Parse(Console.ReadLine());

Console.Write("- Nhap chieu rong can phong: ");

chieurong = double.Parse(Console.ReadLine());

Console.Write("- Nhap don gia: ");

DonGia = double.Parse(Console.ReadLine());

Console.Write("- Nhap gia trang tri: ");

GiaTT = double.Parse(Console.ReadLine());

}

* *Hàm nhập thông tin của lớp PhongTrungBinh*

public override void Nhap()

{

Console.Write("- Nhap chieu dai can phong: ");

chieudai = double.Parse(Console.ReadLine());

Console.Write("- Nhap chieu rong can phong: ");

chieurong = double.Parse(Console.ReadLine());

Console.Write("- Nhap don gia: ");

DonGia = double.Parse(Console.ReadLine());

}

* *Hàm xuất thông tin lớp PhongHangSang:*

public override void ThongTin()

{

Console.WriteLine("Chieu dai la :{0} ", chieudai);

Console.WriteLine("Chieu rong la :{0} ", chieurong);

Console.WriteLine("Dien tich la :{0} ", dientich());

Console.WriteLine("Chi phi la :{0} ", cpSang);

}

* *Hàm xuất thông tin lớp PhongTrungBinh:*

public override void ThongTin()

{

Console.WriteLine("Chieu dai la :{0} ", chieudai);

Console.WriteLine("Chieu rong la :{0} ", chieurong);

Console.WriteLine("Dien tich la :{0} ", dientich());

Console.WriteLine("Chi phi la :{0} ", cp);

}

* *Hàm tính diện tích phòng của lớp Phong:*

public double dientich()

{

double S = chieudai \* chieurong;

return S;

}

* *Hàm tính Chi phí của PhongHangSang:*

public override void TinhChiPhi()

{

cpSang = dientich() \* Dongiasang \* GiaTT;

}

* *Hàm tính Chi phí của PhongTrungBinh:*

public override void TinhChiPhi()

{

cp = dientich() \* dongia;

}

* *Hàm main:*

static void Main(string[] args)

{

int chon;

do

{

Console.WriteLine("\*\*\*\*\*CHON PHONG MUON QUAN LY\*\*\*\*\*");

Console.WriteLine("1. Phong trung binh");

Console.WriteLine("2. Phong hang sang");

Console.Write("\nNhap lua chon cua ban: ");

chon = int.Parse(Console.ReadLine());

switch (chon)

{

case 1:

Console.Write("\nNhap vao so phong TB:");

int n = Convert.ToInt32(Console.ReadLine());

PhongTrungBinh[] a = new PhongTrungBinh[n];

for (int i = 0; i < n; i++)

{

Console.WriteLine("\nNhap phong " + (i + 1));

a[i] = new PhongTrungBinh();

a[i].nhapTB();

a[i].TinhChiphi();

}

for (int i = 0; i < n; i++)

{

Console.WriteLine("\nThong Tin Phong " + (i + 1));

a[i].ThongTin();

}

break;

case 2:

Console.Write("\nNhap vao so phong HS:");

int m = Convert.ToInt32(Console.ReadLine());

PhongHangSang[] ar = new PhongHangSang[m];

for (int i = 0; i < m; i++)

{

Console.WriteLine("\nNhap phong " + (i + 1));

ar[i] = new PhongHangSang();

ar[i].nhapHS();

ar[i].TinhChiphi();

}

for (int i = 0; i < m; i++)

{

Console.WriteLine("\nThong Tin Phong " + (i + 1));

ar[i].ThongTin();

}

break;

}

Console.WriteLine("\n");

} while (chon != 2);

}

* **Giao diện demo kết quả**



