TRƯỜNG ĐẠI HỌC



**SƯ PHẠM KỸ THUẬT THÀNH PHỐ HỒ CHÍ MINH**

HCMC University of Technology and Education

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**



NGÀNH CÔNG NGHỆ THÔNG TIN

**BÁO CÁO ĐỒ ÁN WINDOWS**

**CHƯƠNG TRÌNH QUẢN LÝ THÀNH VIÊN TRONG LAB**



|  |  |
| --- | --- |
| **Nhóm sinh viên thực hiện:** |  |
| **Hoàng Minh Thành** | **18110196** |
| **Nguyễn Huy THế** | **18110202** |
| **Bùi Phúc Lâm** | **18110141** |
| **GVHD: TS. Huỳnh Xuân Phụng** |

**Tp. Hồ Chí Minh, tháng 07 – 2020**

ĐIỂM SỐ

|  |  |  |  |
| --- | --- | --- | --- |
| TIÊU CHÍ | NỘI DUNG | TRÌNH BÀY | TỔNG |
| ĐIỂM |  |  |  |

**NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN**

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

Giảng viên hướng dẫn

***(Ký và ghi rõ họ tên)***

**Huỳnh Xuân Phụng**

**LỜI CẢM ƠN**

Để hoàn thành tốt đề tài và bài báo cáo này, chúng em xin gửi lời cảm ơn chân thành đến giảng viên, tiến sĩ Huỳnh Xuân Phụng, người đã trực tiếp hỗ trợ chúng em trong suốt quá trình làm đề tài. Chúng em cảm ơn thầy đã đưa ra những lời khuyên từ kinh nghiệm thực tiễn của mình để định hướng cho chúng em đi đúng với yêu cầu của đề tài đã chọn, luôn giải đáp thắc mắc và đưa ra những góp ý, chỉnh sửa kịp thời giúp chúng em khắc phục nhược điểm và hoàn thành tốt cũng như đúng thời hạn Khoa đã đề ra.

Chúng em cũng xin gửi lời cảm ơn chân thành các quý thầy cô trong khoa Đào tại Chất Lượng Cao nói chung và ngành Công Nghệ Thông Tin nói riêng đã tận tình truyền đạt những kiến thức cần thiết giúp chúng em có nền tảng để làm nên đề tài này, đã tạo điều kiện để chúng em có thể tìm hiểu và thực hiện tốt đề tài. Cùng lúc đó, chúng em xin được gửi lời cảm ơn đến các bạn cùng khóa đã cung cấp nhiều thông tin và kiến thức hữu ích giúp chúng em có thể hoàn thiện hơn đề tài của minh.

Đề tài và bài báo cáo được chúng em thực hiện trong thời gian ngắn, với những kiến thức còn hạn chế cùng nhiều hạn chế khác về mặt kĩ thuật và kinh nghiệm trong việc thực hiện một dự án phần mềm. Do đó, trong quá trình làm nên đề tài có những thiếu sót là điều không thể tránh khỏi nên chúng em rất mong nhận được những ý kiến đóng góp quý báu của các quý thầy cô để kiến thức của chúng em được hoàn thiện hơn và chúng em có thể làm tốt hơn nữa trong những lần sau. Chúng em xin chân thành cảm ơn.

Cuối lời, chúng em kính chúc quý thầy, quý cô luôn dồi dào sức khỏe và thành công hơn nữa trong sự nghiệp trồng người. Một lần nữa chúng em xin chân thành cảm ơn.

**TP.HCM, ngày 10 tháng 7 năm 2020**

**Nhóm sinh viên thực hiện**

**MỤC LỤC**

[DANH MỤC CÁC HÌNH 1](#_Toc45281088)

[DANH MỤC CÁC BẢNG 2](#_Toc45281089)

[CHƯƠNG 1: TỔNG QUAN CHƯƠNG TRÌNH 3](#_Toc45281090)

[1. Giới thiệu chung 3](#_Toc45281091)

[1.1. Đồ án chương trình quản lý thành viên trong lab 3](#_Toc45281092)

[*1.1.1. Yêu cầu đồ án* 3](#_Toc45281093)

[*1.1.2. Phân tích đồ án* 3](#_Toc45281094)

[*1.1.3. Phương pháp thực hiện* 3](#_Toc45281095)

[1.2. Entity Framework, Code-first 3](#_Toc45281096)

[*1.2.1. Entity Framework* 3](#_Toc45281097)

[*1.2.2. Code-first* 4](#_Toc45281098)

[2. Đặc tả chương trình quản lý thành viên trong lab 4](#_Toc45281099)

[2.1. Chương trình quản lý quản lý thành viên trong lab 4](#_Toc45281100)

[*2.1.1. Giới thiệu về chương trình* 4](#_Toc45281101)

[*2.1.2. Dữ liệu đầu vào, ra* 4](#_Toc45281102)

[*2.1.3. Tính năng chính* 4](#_Toc45281103)

[*2.1.4. Giao diện dự kiến* 5](#_Toc45281104)

[2.2. Yêu cầu kỹ thuật 8](#_Toc45281105)

[2.3. Công cụ sử dụng 9](#_Toc45281106)

[CHƯƠNG 2: PHÂN CÔNG CÔNG VIỆC 9](#_Toc45281107)

[1. Kế hoạch thực hiện 9](#_Toc45281108)

[2. Phân công công việc 9](#_Toc45281109)

[CHƯƠNG 3: THIẾT KẾ CHƯƠNG TRÌNH 10](#_Toc45281110)

[1. Đặc tả lớp 10](#_Toc45281111)

[2. Đặc tả các phương thức trong lớp 10](#_Toc45281112)

[3. Thiết kế cơ sở dữ liệu 13](#_Toc45281113)

[3.1. Mối quan hệ giữa các bảng 13](#_Toc45281114)

[3.2. Các bảng của cơ sở dữ liệu 13](#_Toc45281115)

[3.3. Các trường trong bảng 14](#_Toc45281116)

[4. Thiết kế giao diện 15](#_Toc45281117)

[CHƯƠNG 4: CÀI ĐẶT VÀ KIỂM THỬ 18](#_Toc45281118)

[CHƯƠNG 5: KẾT LUẬN 24](#_Toc45281119)

[TÀI LIỆU THAM KHẢO 24](#_Toc45281120)

# DANH MỤC CÁC HÌNH

[Hình 1. Kiến trúc Entity Framework 3](#_Toc45280559)

[Hình 2. Code-first 4](#_Toc45280560)

[Hình 3. Giao diện đăng nhập 5](#_Toc45280561)

[Hình 4. Giao diện màn hình chính 5](#_Toc45280562)

[Hình 5. Giao diện thông tin nhân viên 6](#_Toc45280563)

[Hình 6. Giao diện tìm kiếm nhân viên 6](#_Toc45280564)

[Hình 7. Giao diện thông tin dự án 7](#_Toc45280565)

[Hình 8. Giao diện thông tin các công việc nhỏ trong dự án 7](#_Toc45280566)

[Hình 9. Giao diện quản lý dự án 8](#_Toc45280567)

[Hình 10. Giao diện thông tin phần mềm 8](#_Toc45280568)

[Hình 11. Mối quan hệ giữa các bảng trong cơ sở dữ liệu 13](#_Toc45280569)

# DANH MỤC CÁC BẢNG

[Table 1. Kế hoạch thực hiện 9](#_Toc45279749)

[Table 2. Phân công công công việc 9](#_Toc45279750)

[Table 3. Đặc tả lớp 10](#_Toc45279751)

[Table 4. Đặc tả các phương thức trong lớp PcController.cs 10](#_Toc45279752)

[Table 5. Đặc tả các phương thức trong lớp TaskController.cs 11](#_Toc45279753)

[Table 6. Đặc tả các phương thức trong lớp UserController.cs 11](#_Toc45279754)

[Table 7. Đặc tả các phương thức trong lớp ListUserController.cs 12](#_Toc45279755)

[Table 8. Các bảng trong cơ sở dữ liệu 13](#_Toc45279756)

[Table 9. Các trường trong bảng User 13](#_Toc45279757)

[Table 10. Các trường trong bảng Task 14](#_Toc45279758)

[Table 11. Các trường trong bảng TaskInfoes 14](#_Toc45279759)

[Table 12. Các trường trong bảng UserTasks 14](#_Toc45279760)

[Table 13. Giao diện 15](#_Toc45279761)

[Table 14. Cài đặt và kiểm thử 18](#_Toc45279762)

# CHƯƠNG 1: TỔNG QUAN CHƯƠNG TRÌNH

## Giới thiệu chung

### Đồ án chương trình quản lý thành viên trong lab

#### Yêu cầu đồ án

Thiết kế và xây dựng chương trình quản lý thành viên trong lab để quản lý các thông tin gồm: Thông tin nhân viên, thông tin dự án, quản lý tiền lương, thời gian làm việc, danh sách các dự án, các công việc cần phải làm, cho phép tìm kiếm nhân viên, tìm kiếm dự án và quản lí tiến độ thực hiện dự án.

#### Phân tích đồ án

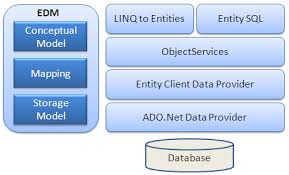
* Xây dựng mô hình theo hướng đối tượng.
* Xây dựng database để làm dữ liệu đầu vào.
* Truyền database vào giao diện đưa tới người dùng.

#### Phương pháp thực hiện

* Xây dựng chương trình trên C# sử dụng Entity Framework trong .NET Framework theo hướng tiếp cận Code-first và sử dụng mô hình MVC (Model-View-Controller).
* Sử dụng ứng dụng phần mềm SSMS (SQL Server Management Studio) để xây dựng database.
* Sử dụng Windows Forms để thiết kế giao diện người dùng.

### Entity Framework, Code-first

#### Entity Framework



##### Hình . Kiến trúc Entity Framework

Entity Framework là một thư viện ORM (Object Relational Mapping)-một kỹ thuật cơ chế lập trình thực hiện ánh xạ qua lại giữa cơ sở dữ liệu và các đối tượng trong các ngôn ngữ lập trình hướng đối tượng như Java, C#...- được Microsoft phát triển từ năm 2008.

Entity Framework được chia làm các thành phần: Entity Data Model (EDM), Entity Client data provider, Object services, LINQ to Entities và Entity SQL. Và các thành phần này được chia làm 2 khối liên quan đến 2 giai đoạn làm việc với Entity Framework: Xây dựng Entity Data Model, truy vấn và xử lý dữ liệu.

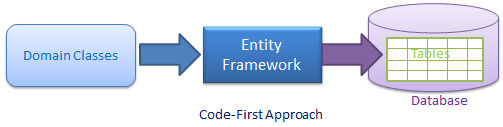
Giai đoạn xây dựng EDM có bốn cách tiếp cận khác nhau: Database-first, Model-first, Code-first, Code-first from database. Lựa chọn cách nào tùy vào từng điều kiện cụ thể. Dù việc lựa chọn hướng tiếp cận EDM là gì cũng không ảnh hưởng đến giai đoạn xử lý.

Ở giai đoạn xử lý dữ liệu, sử dụng LINQ to Entities và Object services để làm việc với dữ liệu dựa trên EDM đã xây dựng.

#### Code-first

Enity Framework đã giới thiệu cách tiếp cận Code-First với Enity Framework 4.1

Code-First chủ yếu hữu ích trong Domain Driven Design. Trong cách tiếp cận Code-First, bạn tập trung vào miền của ứng dụng của bạn và bắt đầu tạo các lớp cho thực thể miền của bạn thay vì thiết kế cơ sở dữ liệu của bạn trước rồi tạo các lớp khớp với thiết kế cơ sở dữ liệu của bạn. Hình dưới đây minh họa cách tiếp cận Code-First



##### Hình . Code-first

Có thể thấy trong hình trên, API của Enity Framework sẽ tạo cơ sở dữ liệu dựa trên cấu hình và các lớp miền của bạn. Điều này có nghĩa là bạn cần bắt đầu mã hóa trước bằng C# hoặc VB.NET và sau đó Enity Framework sẽ tạo cơ sở dữ liệu từ mã của bạn.

## Đặc tả chương trình quản lý thành viên trong lab

### Chương trình quản lý quản lý thành viên trong lab

#### Giới thiệu về chương trình

Chương trình quản lý thành viên trong lab được xây dựng trên C# giúp người dùng dễ dàng quản lý các thông tin của nhân viên, dự án, lương thông qua database đã có sẵn và kết nối với chương trình, hoặc người dùng có thể nhập trực tiếp các thông tin này trên giao diện người dùng.

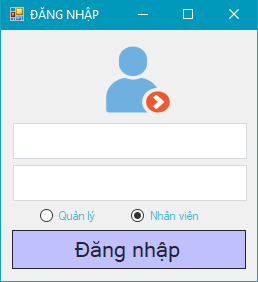
#### Dữ liệu đầu vào, ra

* Dữ liệu đầu vào: Database chứa thông tin nhân viên, dự án.
* Dữ liệu đầu ra: Hiển thị thông tin nhân viên, dự án.

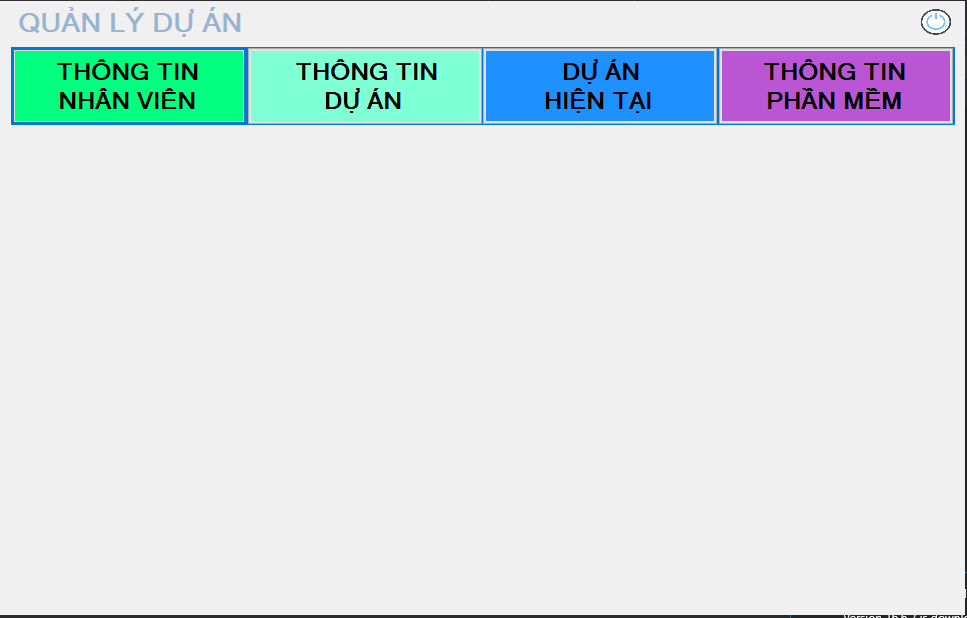
#### Tính năng chính

* Hiển thị thông tin nhân viên, dự án.
* Cho phép tìm kiếm nhân viên, dự án.
* Cho phép Admin thêm, sửa, xóa các thông tin.
* Quản lý dự án (quản lý tiến độ, nhân viên nào tham gia công việc).

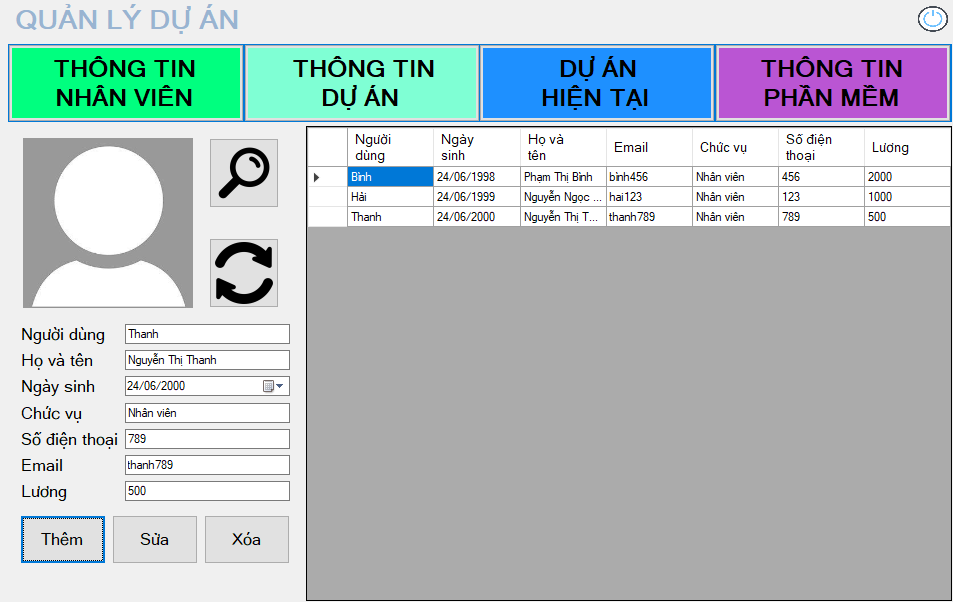
#### Giao diện dự kiến



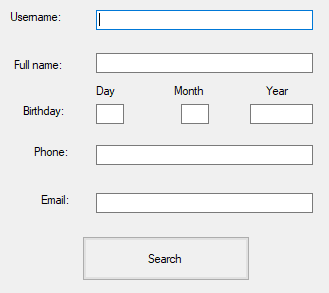
##### Hình . Giao diện đăng nhập



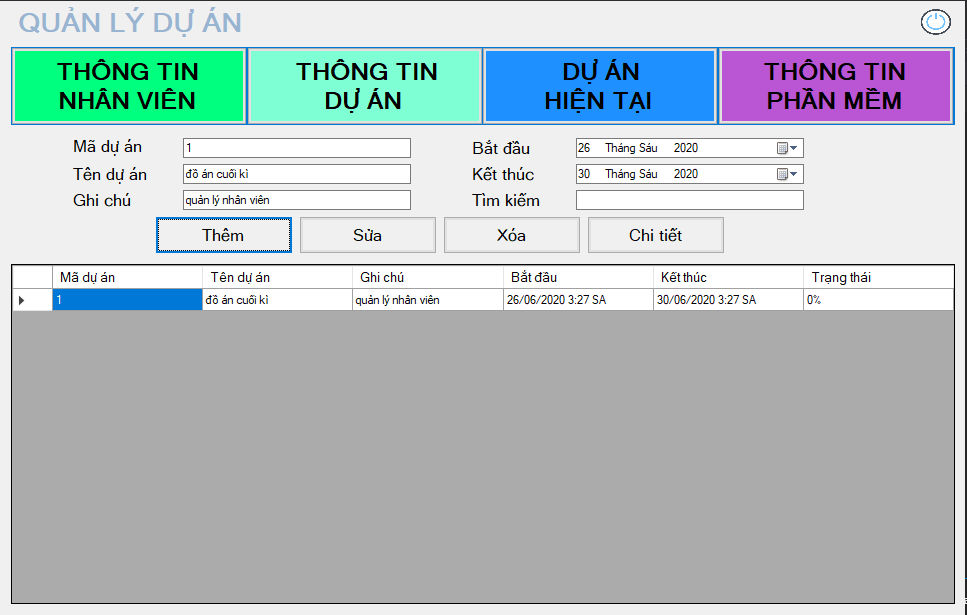
*Hình 4. Giao diện màn hình chính*



*Hình 5. Giao diện thông tin nhân viên*



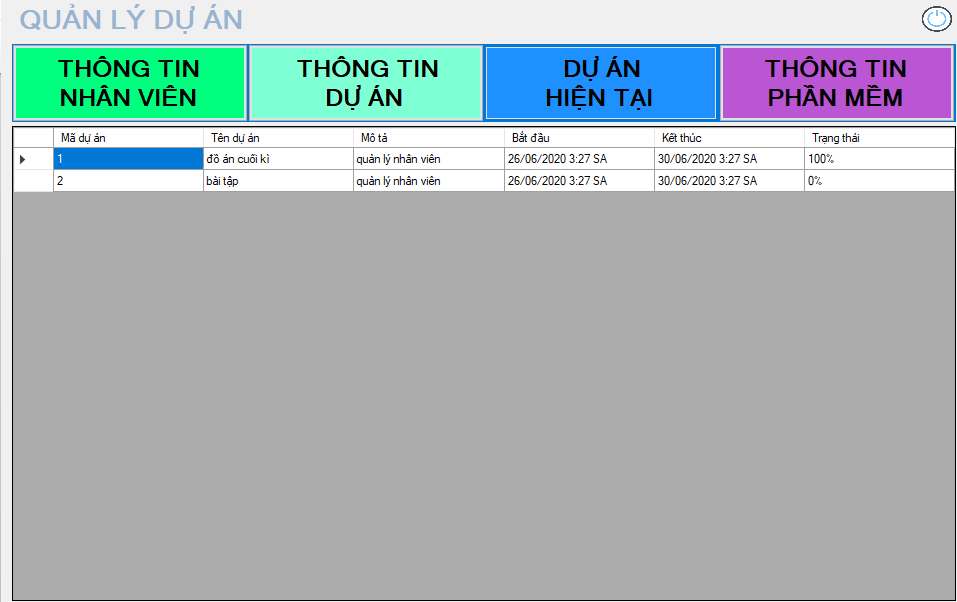
##### Hình . Giao diện tìm kiếm nhân viên



##### Hình . Giao diện thông tin dự án



##### Hình . Giao diện thông tin các công việc nhỏ trong dự án



##### Hình . Giao diện quản lý dự án



##### Hình . Giao diện thông tin phần mềm

### Yêu cầu kỹ thuật

* Hoàn thành các yêu cầu đề ra
* Chương trình thực thi được
* Áp dụng các kiến thức đã học

### Công cụ sử dụng

* Xây dựng database bằng phần mềm SQL Server Management Studio
* Thiết kế giao diện người dùng trên Windows form
* Xây dựng chương trình trên Visual Studio 2019 (C#)

# CHương 2: Phân công công việc

## Kế hoạch thực hiện

Thời gian dự kiến: 29/04/2020 – 01/07/2020

*Table 1. Kế hoạch thực hiện*

|  |  |
| --- | --- |
| Tuần | Công việc |
| 5 | Tìm hiểu về Entity Framework và Code-first, cách xây dựng database |
| 6 | Phân tích dữ liệu đầu vào, đầu ra, xây dựng database |
| 7 | Thiết kế giao diện ban đầu |
| 8 | Viết thuật toán kết nối Database vào chương trình, xây dựng mô hình MVC |
| 9 | Viết thuật toán chỉnh sửa thông tin trực tiếp trên giao diện và lưu vào database |
| 10 | Hoàn thiện chương trình. Soát lỗi, kiểm thử chương trình |
| 11 | Viết báo cáo |
| 12 | Hoàn thành báo cáo |
| 13 | Kiểm tra chương trình lần cuối, kiểm tra bài báo cáo |
| 14 | Báo cáo cuối kỳ |

## Phân công công việc

*Table 2. Phân công công công việc*

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên SV | Mô tả công việc | Phần trăm đóng góp |
| 1 | Bùi Phúc Lâm | - Code chính: tạo thông tin nhân viên, dự án, lương  - Viết thuật toán kết nối Database vào chương trình, xây dựng mô hình MVC  - fix bug  - Hoàn thiện chương trình | 40% |
| 2 | Nguyễn Huy Thế | - Thiết kế giao diện ban đầu  - Chỉnh sửa và hoàn thành giao diện  - Hỗ trợ code phụ  - Kiểm tra lần cuối | 30% |
| 3 | Hoàng Minh Thành | -Xây dựng database  - Viết thuật toán kết nối Database vào chương trình, xây dựng mô hình MVC  - Viết thuật toán chỉnh sửa thông tin trực tiếp trên giao diện và lưu vào database  - Viết báo cáo, cài đặt và kiểm thử | 30% |

# Chương 3: thiết kế chương trình

## Đặc tả lớp:

* Sinh viên phụ trách: Hoàng Minh Thành

*Table 3. Đặc tả lớp*

|  |  |  |
| --- | --- | --- |
| TT | Tên lớp | Mục đích |
| 1 | PcController.cs | Xử lý cơ sở dữ liệu của bảng phân công công việc dự án |
| 2 | TaskController.cs | Xử lý cơ sở dữ liệu của bảng quản lý dự án |
| 3 | UserController.cs | Xử lý cơ sở dữ liệu của bảng quản lý nhân viên |
| 4 | ListUserController.cs | Xử lý các tác vụ liên quan tới ListUser và ListTask |

## Đặc tả các phương thức trong lớp

*Table 4. Đặc tả các phương thức trong lớp PcController.cs*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Tên SV phụ trách |
| 1 | public static void AddPc(TaskInfo taskinfo) | Phân công các công việc nhỏ trong dự án | PcController.cs  (14) | Bùi Phúc Lâm |
| 2 | public static List<TaskInfo> getAllInfo() | Lấy tất cả danh sách của công việc được phân công | PcController.cs  (22) | Bùi Phúc Lâm |
| 3 | public static TaskInfo GetTaskInfo(int ID, string name) | Lấy thông tin của công việc tương ứng với ID và name | PcController.cs  (45) | Bùi Phúc Lâm |
| 4 | public static bool UpdateInfo(TaskInfo task) | Cập nhật thông tin của công việc | PcController.cs  (62) | Hoàng Minh Thành |

*Table 5. Đặc tả các phương thức trong lớp TaskController.cs*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Tên SV phụ trách |
| 1 | public static List<Models.Task> GetListTask() | Lấy dữ liệu từ các dự án | TaskController.cs  (14) | Bùi Phúc Lâm |
| 2 | public static void AddTask(Models.Task task) | Thêm dự án | TaskController.cs  (23) | Hoàng Minh Thành |
| 3 | public static void DeleteTask(Models.Task task) | Xóa dự án | TaskController.cs  (31) | Hoàng Minh Thành |
| 4 | public static int UpdateTask(Models.Task task) | Cập nhật dự án | TaskController.cs  (42) | Hoàng Minh Thành |
| 5 | public static Models.Task GetTask(int id) | Lấy thông tin của 1 dự án | TaskController.cs  (56) | Hoàng Minh Thành |

##### **Table 6. Đặc tả các phương thức trong lớp UserController.cs**

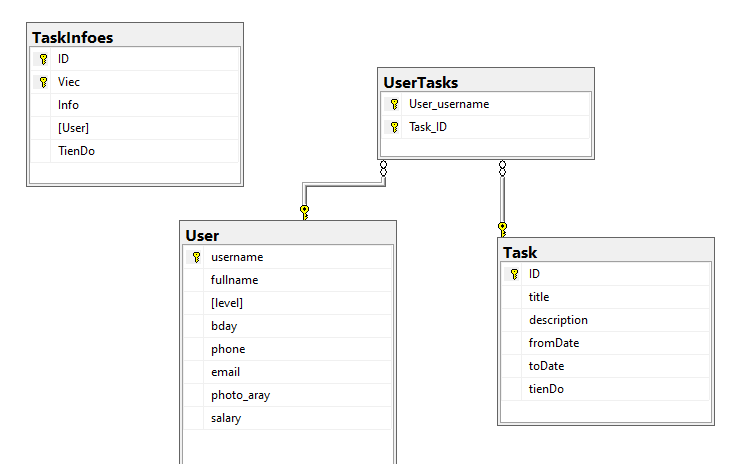
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Tên SV phụ trách |
| 1 | public static List<User> GetListUser() | Lấy danh sách của tất cả User | UserController.cs  (22) | Bùi Phúc Lâm |
| 2 | public static void AddUser(User user) | Thêm User | UserController.cs  (31) | Hoàng Minh Thành |
| 3 | public static void DeleteUser(User user) | Xóa User | UserController.cs  (39) | Hoàng Minh Thành |
| 4 | public static int UpdateUser(User user) | Cập nhật User | UserController.cs  (50) | Hoàng Minh Thành |
| 5 | public static User GetUser(string username) | Lấy thông tin của User có username | UserController.cs  (64) | Bùi Phúc Lâm |
| 6 | public static List<User> getListUser(string usernamesearch) | Lấy danh sách những User có username chứa usernamesearch | UserController.cs  (72) | Bùi Phúc Lâm |

##### **Table 7. Đặc tả các phương thức trong lớp ListUserController.cs**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Tên SV phụ trách |
| 1 | public static List<User> GetListSearch(string username, string fullname, string day, string month, string year, string phone, string email) | Tìm kiếm User theo thông tin | ListUserController.cs  (13) | Bùi Phúc Lâm |
| 2 | public static List<Models.Task> GetTaskSearch(string search) | Tìm kiếm dự án theo thông tin | ListUserController.cs  (33) | Bùi Phúc Lâm |
| 3 | public static List<Models.Task> GetTaskSearchRecent(DateTime day) | Liệt kê dự án đang trong quá trình thực hiện | ListUserController.cs  (60) | Bùi Phúc Lâm |

## Thiết kế cơ sở dữ liệu:

### Mối quan hệ giữa các bảng:



##### Hình . Mối quan hệ giữa các bảng trong cơ sở dữ liệu

### Các bảng của cơ sở dữ liệu:

*Table 8. Các bảng trong cơ sở dữ liệu*

|  |  |  |
| --- | --- | --- |
| TT | Tên bảng | Mục đích |
| 1 | User | Chứa thông tin của các nhân viên |
| 2 | Task | Chứa thông tin của các dự án |
| 3 | TaskInfoes | Chứa các công việc nhỏ trong dự án |
| 4 | UserTasks | Bảng trung gian liên kết User và Task |

### Các trường trong bảng:

##### **Table 9. Các trường trong bảng User**

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | username | nvarchar(128) | Chứa tên nhân viên |
| 2 | fullname | nvarchar(MAX) | Chứa họ và tên nhân viên |
| 3 | [level] | nvarchar(MAX) | Chứa chức vụ nhân viên |
| 4 | bday | datetime | Chứa ngày sinh nhân viên |
| 5 | phone | nvarchar(MAX) | Chứa số điện thoại nhân viên |
| 6 | email | nvarchar(MAX) | Chứa email nhân viên |
| 7 | photo\_aray | varbinary(MAX) | Chứa ảnh nhân viên |
| 8 | salary | bigint | Chứa lương nhân viên |

*Table 10. Các trường trong bảng Task*

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | ID | int | Chứa mã dự án |
| 2 | title | nvarchar(MAX) | Chứa tên dự án |
| 3 | description | nvarchar(MAX) | Mô tả nội dung dự án |
| 4 | fromDate | datetime | Ngày bắt đầu dự án |
| 5 | toDate | datetime | Ngày kết thúc dự án |
| 6 | tienDo | nvarchar(MAX) | Thể hiện tiến dộ công việc |

***Table 11. Các trường trong bảng TaskInfoes***

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | ID | int | Chứa mã dự án |
| 2 | Viec | nvarchar(128) | Chứa những công việc nhỏ trong dự án |
| 3 | Info | nvarchar(MAX) | Chứa thông tin công việc |
| 4 | [User] | nvarchar(MAX) | Người thực hiện công việc |
| 5 | TienDo | nvarchar(MAX) | Tiến độ công việc |

##### **Table 12. Các trường trong bảng UserTasks**

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | User\_username | nvarchar(128) | Chứa tên nhân viên |
| 2 | Task\_ID | int | Chứa mã dự án |

## Thiết kế giao diện:

* Sinh viên phụ trách: Nguyễn Huy Thế

*Table 13. Giao diện*

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Màn hình/Cửa sổ/Dialog (Tên màn hình trên 1 dòng; dưới dòng này là hình chụp màn hình) | Mục đích | Giải thích |
| 1 | - Giao diện đăng nhập | Giúp quản lý người dùng chương trình và phân quyền và tránh người lạ không phải nhân viên hay quản lý vào phầm mềm. | Giao diện tương đối dễ nhìn, đơn giản dễ sử dụng |
| 2 | - Giao diện màn hình chính | Giúp người dùng chọn nội dung muốn sử dụng | Giao diện đơn giản dễ sử dụng |
| 3 | - Giao diện thông tin nhân viên | Giúp xem được danh sách nhân viên và thông tin cá nhân, có thể thêm, sửa hoặc xóa thông tin | Giao diện đơn giản dễ sử dụng |
| 4 | - Giao diện tìm kiếm nhân viên | Khi muốn tìm kiếm thông tin của một nhân viên nào đó khi biết Username | Giao diện đơn giản dễ sử dụng |
| 5 | - Giao diện thông tin dự án | Thêm và xem được thông tin dự án đó, giúp người quản lý chỉnh sửa thông tin dự án và tìm kiếm được dự án nhanh hơn bằng mã dự án | Giao diện đơn giản dễ sử dụng |
| 6 | - Giao diện thông tin các công việc nhỏ trong dự án | Giúp chia dự án ra thành nhiều công việc nhỏ để dễ thực hiện và tăng thời gian hoàn thành và hiển thị tiến độ của từng công việc | Giao diện đơn giản dễ sử dụng |
| 7 | - Giao diện quản lý dự án | Hiển thị các dự án đã lên kế hoạch | Giao diện đơn giản dễ sử dụng |
| 8 | - Giao diện thông tin phần mềm | Giúp hiển thị thông tin phần mềm, ai làm ra | Giao diện đơn giản dễ sử dụng |

# Chương 4: cài đặt và kiểm thử

*Table 14. Cài đặt và kiểm thử*

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tình huống | Mục đích | Mô tả |
| Kiểm thử 1:  Thông tin chi tiết và danh sách nhân viên |  | Mục đích: Xem được danh sách nhân viên và thông tin cá nhân | **-Giúp người quản lý có thể biết được danh sách nhân viên của mình.**  Dữ liệu đầu vào là database chứa thông tin nhân viên |
| Kiểm thử 2:  Chỉnh sửa thông tin nhân viên( thêm, sửa, xóa) | - Thêm nhân viên: (thêm nhân viên Long)    - Sửa thông tin nhân viên: (Lương của nhân viên Long từ 700 sửa thành 1000)    - Xóa nhân viên: ( xóa nhân viên Bình) | Mục đích: Giúp người quản lý chỉnh sửa thông tin | **-Để cập nhật thêm nhân viên mới hoặc bỏ bớt nhân viên, sửa trong việc nhập sai thông tin.**  Dữ liệu đầu vào là database chứa thông tin nhân viên. |
| Kiểm thử 3:  Tìm kiếm nhân viên | - Tìm nhân viên Long: (nhập username và bấm search)    - Thông tin về nhân viên Long cần tìm kiếm đã hiển thị | Mục đích: Khi muốn tìm kiếm thông tin của một nhân viên nào đó khi biết Username | **-Giúp tra cứu thông tin của một nhân viên cụ thể khi biết tên.**  Dữ liệu đầu vào là database chứa thông tin nhân viên |
| Kiểm thử 4:  Thông tin chi tiết và danh sách dự án | - Thông tin chi tiết dự án:    - Danh sách dự án hiện tại: | Mục đích:  Xem được danh sách các dự án và thông tin dự án đó | **-Giúp người quản lý có thể biết được danh sách dự án của mình.**  Dữ liệu đầu vào là database chứa thông tin dự án. |
| Kiểm thử 5:  Chỉnh sửa thông tin dự án ( thêm, sửa, xóa) | - Thêm dự án: (thêm dự án 3)    - Sửa thông tin dự án: (sửa ngày bắt đầu và kết thúc của dự án 3)    - Xóa dự án: (xóa dự án 2) | Mục đích: Giúp người quản lý chỉnh sửa thông tin dự án | **-Để cập nhật thêm dự án mới hoặc bỏ bớt dự án, sửa trong việc nhập sai thông tin.**  Dữ liệu đầu vào là database chứa thông tin dự án. |
| Kiểm thử 6:  Tìm kiếm dự án | - Tìm kiếm dự án 3: (nhập tên dự án ở chỗ tìm kiếm) | Mục đích: Khi muốn tìm kiếm thông tin của một dự án nào đó qua mã dự án | **-Giúp tra cứu thông tin của một dự án cụ thể khi biết mã dự án.**  Dữ liệu đầu vào là database chứa thông tin dự án |
| Kiểm thử 7:  Phân công công việc trong mỗi dự án | - Phân công: (chọn dự án 3 và bấm chi tiết)    - Thêm công việc: (thêm công việc mới và chọn người thực hiện công việc đó) | Mục đích: Giúp chia dự án ra thành nhiều công việc nhỏ để dễ thực hiện và tăng thời gian hoàn thành | **-Giúp hoàn thành dự án nhanh hơn khi chia ra các công việc nhỏ.**  Dữ liệu đầu vào là database chứa thông tin dự án |
| Kiểm thử 8:  Tiến độ thực hiện của dự án và từng công việc phân công trong dự án | - Tiến độ từng công việc được phân công trong dự án: (việc sửa báo cáo của dự án 3 đã hoàn thành)    - Tiến độ của dự án: ( dự án 3 sau khi hoàn thành 1 công việc đã đạt được 50% tiến độ) | Mục đích:  Giúp theo dõi được tiến độ thực hiện được đến đâu để điều chỉnh lại cho kịp tiến độ dự án | **-Giúp theo dõi tiến độ thực hiện và điều chỉnh cho phù hợp với thời gian kết thúc dự án.**  Dữ liệu đầu vào là database chứa thông tin dự án |
| Kiểm thử 9:  Thông tin về phần mềm đang sử dụng |  | Mục đích: Giúp người xem biết được phần mềm gì và ai làm | **-Giúp biết thông tin phần mềm.**  Dữ liệu đầu vào là database chứa thông tin nhân viên |
| Kiểm thử 10:  Đăng nhập chương trình |  | Mục đích: Quản lý người dùng chương trình và phân quyền | **-Giúp bạn đăng nhập được tránh người khác không phải nhân viên hay người quản lý vào.**  Dữ liệu đầu vào là database chứa thông tin nhân viên |

# Chương 5: kết luận

*Tự đánh giá mức độ hoàn thành mục tiêu:* 9/10.

*Các khó khăn gặp phải:* do làm riêng code với giao diện nên khi kết hợp gặp trục trặc, bị một số lỗi lặt vặt.

*Cách khắc phục:* xem lại video bài giảng của thầy, stack overflow.

*Ưu điểm:* giao diện đơn giản, dễ nhìn; đáp ứng đủ yêu cầu cần thiết

*Hạn chế:* chức năng ở mức đơn giản chưa có nhiều nâng cao.

*Ý tưởng phát triển đồ án:* thêm nhiều ràng buộc vào thông tin nhân viên, bổ sung cơ sở dữ liệu đầy đủ hơn, các chức năng nâng cao hơn: tự chấm công lương theo giờ làm, tiến độ hoàn thành công việc được tốt hơn như nhập thời gian sẽ hiển thị tiến độ rõ hơn.

# Tài liệu tham khảo

[1].Các video dạy học online của thầy Huỳnh Xuân Phụng.

<https://www.youtube.com/channel/UCWwPuel7ja3fi1TyXWpHzXg>

[2]. <https://stackoverflow.com/>