

**BỘ GIÁO DỤC & ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA ĐIỆN – ĐIỆN TỬ**

**BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP – Y SINH**

**🙟🙟🕮🙝🙝**

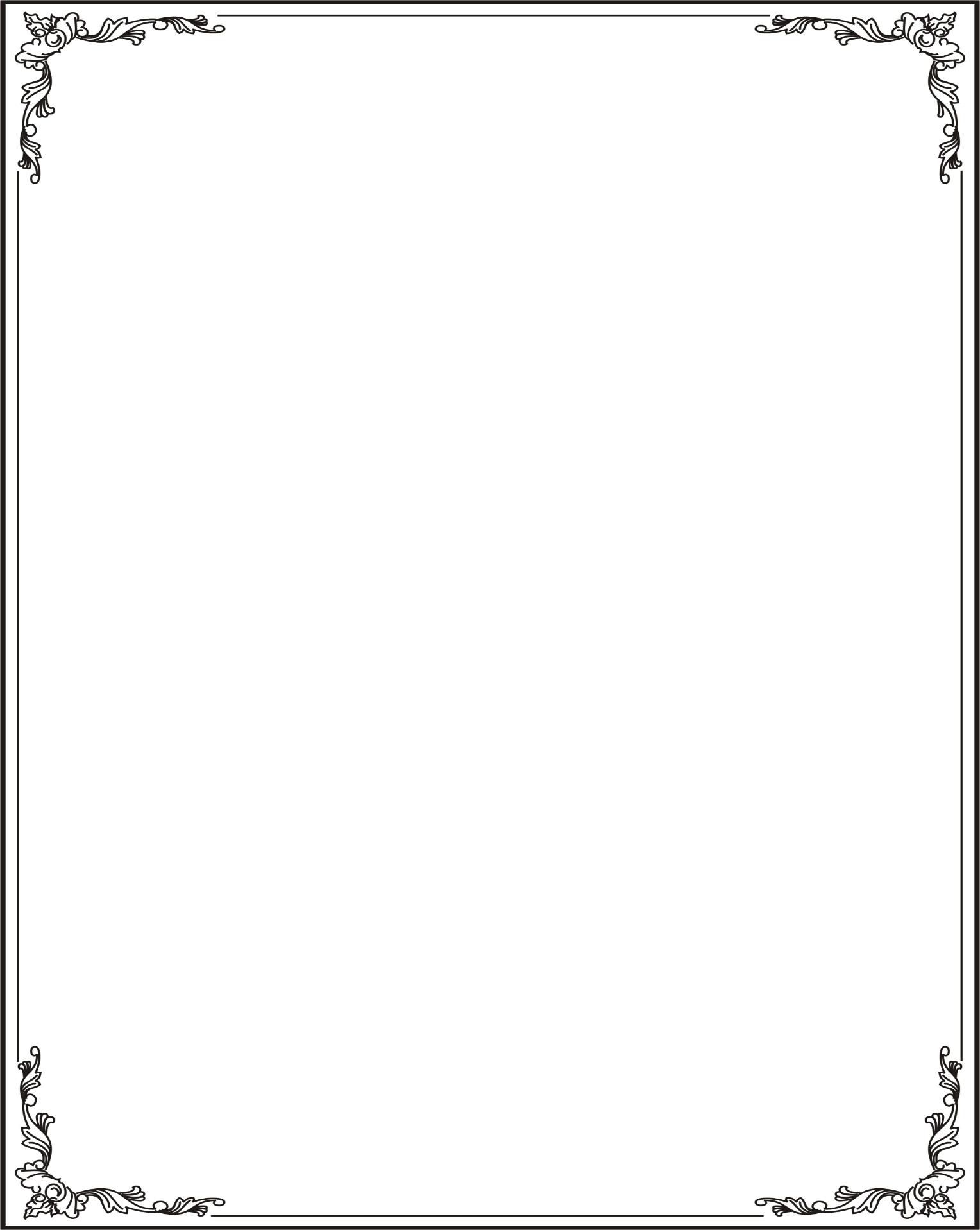
****

**ĐỒ ÁN MÔN HỌC**

**Đề tài: THIẾT KẾ VÀ THI CÔNG HỆ THỐNG CHẤM CÔNG SỬ DỤNG RFID, C# VÀ SQL SERVER**

|  |  |
| --- | --- |
| **GVDH: ThS. Nguyễn Văn Hiệp** | |
| **SVTH: Nguyễn Thành Đạt** |  |
| **MSSV: 19161219** |  |
|  |  |
|  |  |
|  |  |

**Tp. Hồ Chí Minh – 12/2022**



**BỘ GIÁO DỤC & ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA ĐIỆN – ĐIỆN TỬ**

**BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP – Y SINH**

**🙟🙟🕮🙝🙝**

**ĐỒ ÁN MÔN HỌC**

***Đề tài:* THIẾT KẾ VÀ THI CÔNG HỆ THỐNG CHẤM CÔNG SỬ DỤNG RFID, C# VÀ SQL SERVER**

|  |  |
| --- | --- |
| **GVDH: ThS. Nguyễn Văn Hiệp** | |
| **SVTH: Nguyễn Thành Đạt** |  |
| **MSSV: 19161219** |  |
|  |  |
|  |  |
|  |  |

**Tp. Hồ Chí Minh – 12/2022**

|  |  |
| --- | --- |
| **TRƯỜNG ĐH SPKT TP.HCM**  KHOA ĐIỆN – ĐIỆN TỬ  BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP Y SINH | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  ĐỘC LẬP – TỰ DO – HẠNH PHÚC |
|  | Tp.HCM, ngày 10 tháng 12 năm 2022 |

# **NHIỆM VỤ CỦA ĐỒ ÁN MÔN HỌC**

|  |  |  |
| --- | --- | --- |
| Họ và tên SVHT: | Nguyễn Thành Đạt | MSSV: 19161219 |
| Khóa: | 2019 | Lớp: 19161ĐTCNC |
| Chuyên ngành: | Điện tử công nghiệp |  |

I. Đề tài: **Thiết kế và thi công hệ thống chấm công sử dụng RFID, C# và SQL Server**

II. Nhiệm vụ

1. Các số liệu ban đầu

+ Sử dụng Arduino Uno và module RC522.

+ Sử dụng nền tảng lập trình chuyên nghiệp Visual Studio 2.

+ Sử dụng cơ sở dữ liệu SQL Server.

+ Sử dụng 3 ngôn ngữ lập trình C, C# và SQL.

+ Kết hợp các tài liệu, tư liệu,... có thể giúp ích cho việc phát triển đề tài.

2. Nội dung thực hiện

+ Tìm hiểu về Arduino Uno và module RC522.

+ Tìm hiểu về các ngôn ngữ lập trình mới.

+ Nghiên cứu việc truyễn nhận dữ liệu giữa phần cứng và phần mềm.

+ Thực hiện thiết kế hệ thống, thi công, hoàn thiện hệ thống.

+ Viết báo cáo.

+ Bảo vệ đề tài.

III. Ngày nhận nhiệm vụ: 19/09/2022.

IV. Ngày hoàn thành: 10/12/2022.

V. Họ và tên cán bộ hướng dẫn: ThS. Nguyễn Văn Hiệp

**Cán bộ hướng dẫn**

**ThS. Nguyễn Văn Hiệp**

|  |  |
| --- | --- |
| **TRƯỜNG ĐH SPKT TP.HCM**  KHOA ĐIỆN – ĐIỆN TỬ  BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP Y SINH | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  ĐỘC LẬP – TỰ DO – HẠNH PHÚC |
|  | Tp.HCM, ngày 10 tháng 12 năm 2022 |

# **LỊCH TRÌNH THỰC HIỆN ĐỒ ÁN MÔN HỌC**

**(Bản lịch trình này được đóng vào đồ án)**

Họ tên sinh viên thực hiện: **Nguyễn Thành Đạt** - MSSV: **19161219**

Đề tài: **Thiết kế và thi công hệ thống chấm công sử dụng RFID, C# và SQL server.**

|  |  |  |
| --- | --- | --- |
| ***Tuần/ngày*** | ***Nội dung*** | ***Xác nhận GVHD*** |
| Tuần 5, 6  (19/09 – 02/09) | - Nghe GVHD phổ biến yêu cầu làm đồ án, chọn đề tài.  - GVHD tiến hành xét duyệt đề tài |  |
| Tuần 7, 8  (03/10 – 16/10) | - Viết tóm tắt yêu cầu đề tài đã chọn: đề tài làm cái gì, nội dung thiết kế, các thông số của đề tài  - Thiết kế sơ đồ khối, chức năng các khối |  |
| Tuần 9, 10  (17/10 – 30/10) | - Thiết kế sơ đồ mạch, nguyên lý hoạt động của mạch, thiết kế mạch thử nghiệm bằng dây cắm  -Tìm hiểu và thiết kế cơ sở dữ liệu trên SQL server |  |
| Tuần 11, 12  (31/10 – 13/11) | - Thiết kế phần mềm cho hệ thống dựa trên C# (.NET Winforms framework) |  |
| Tuần 13  (14/11 – 20/11) | - Thi công mạch in  - Viết báo cáo những nội dung đã làm |  |
| Tuần 14  (21/11 – 27/11) | - Hoàn thiện báo cáo và gởi cho GVHD để xem xét góp ý lần cuối trước khi in và báo cáo |  |
| Tuần 15, 16, 17  (28/11 – 18/12) | - Nộp quyển báo cáo  - Làm slide, báo cáo với GVHD |  |

**GV HƯỚNG DẪN**

**ThS. Nguyễn Văn Hiệp**

# **LỜI CAM ĐOAN**

Đề tài này được em thực hiện dựa vào các tài liệu tham khảo có sẵn và hoàn toàn không sao chép từ bất kì nguồn nào.

**Người thực hiện**

**Nguyễn Thành Đạt**

# **LỜI CẢM ƠN**

Lời nói đầu tiên em xin trân trọng gửi lời cảm ơn chân thành đến các quý Thầy, Cô của Trường Đại Học Sư Phạm Kỹ Thuật TP Hồ Chí Minh nói chung và đặc biệt là các Thầy, Cô của Khoa Điện - Điện Tử nói riêng đã dành hết tâm huyết giảng dạy, truyền đạt những kinh nghiệm và những kiến thức quý báu cho chúng em để tạo tiền đề để thực hiện được đề tài này.

Em cũng xin gửi lời cảm ơn sâu sắc nhất đến Thầy Nguyễn Văn Hiệp, người đã trực tiếp hướng dẫn em một cách tận tình trong suốt học kì, luôn hỗ trợ em và cả các bạn chung lớp hết mình, đưa ra hướng giải quyết vấn đề tốt nhất để giúp em hoàn thành đồ án môn học lần này. Một lần nữa em xin được cảm ơn Thầy.

Do là lần đầu tiên thực hiện đồ án môn học nên có thể em sẽ mắc nhiều sai sót do thời gian và kiến thức của bản thân có hạn. Mong sẽ được Thầy đóng góp ý kiến để em có thể hoàn thiện hơn trong tương lai.

**Người thực hiện**

**Nguyễn Thành Đạt**

**MỤC LỤC**

[**NHIỆM VỤ CỦA ĐỒ ÁN MÔN HỌC** i](#_Toc122293667)

[**LỊCH TRÌNH THỰC HIỆN ĐỒ ÁN MÔN HỌC** ii](#_Toc122293668)

[**LỜI CAM ĐOAN** iii](#_Toc122293669)

[**LỜI CẢM ƠN** iv](#_Toc122293670)

[**MỤC LỤC** v](#_Toc122293671)

[**DANH MỤC ẢNH** vii](#_Toc122293672)

[**DANH MỤC BẢNG** viii](#_Toc122293673)

[**DANH MỤC TỪ VIẾT TẮT** ix](#_Toc122293674)

[**TÓM TẮT** x](#_Toc122293675)

[**Chương 1. DẪN NHẬP** 1](#_Toc122293676)

[**1.1. Lý do chọn đề tài** 1](#_Toc122293677)

[**1.2. Mục tiêu của đề tài** 1](#_Toc122293678)

[**1.3. Giới hạn của đề tài** 1](#_Toc122293679)

[**1.4. Kế hoạch nghiên cứu** 2](#_Toc122293680)

[**Chương 2. THIẾT KẾ** 3](#_Toc122293681)

[**2.1. Bài toán thiết kế** 3](#_Toc122293682)

[**2.1.1. Mục tiêu về phần cứng** 3](#_Toc122293683)

[**2.1.2. Mục tiêu về phần mềm** 3](#_Toc122293684)

[**2.2. Sơ đồ khối** 3](#_Toc122293685)

[**2.3. Thiết kế của từng khối** 4](#_Toc122293686)

[**2.3.1. Khối xử lý** 4](#_Toc122293687)

[**2.3.2. Thiết kế khối nguồn** 6](#_Toc122293688)

[**2.3.3. Sơ đồ nguyên lý toàn mạch (Phần cứng)** 6](#_Toc122293689)

[**2.3.3. Thiết kế phần mềm (Trên khối PC)** 7](#_Toc122293690)

[**Chương 3. THI CÔNG** 11](#_Toc122293691)

[**3.1. Thi công phần cứng** 11](#_Toc122293692)

[**3.1.1. Thi công mạch in** 11](#_Toc122293693)

[**3.1.2. Hàn linh kiện** 12](#_Toc122293694)

[**3.1.3. Lưu đồ điều khiển khối điều khiển** 13](#_Toc122293695)

[**3.2. Thi công phần mềm** 14](#_Toc122293696)

[**3.2.1 Giới thiệu về Git và GitHub** 14](#_Toc122293697)

[**3.2.2. Tạo cơ sở dữ liệu** 15](#_Toc122293698)

[**3.2.3. Giới thiệu ngôn ngữ và kiến trúc lập trình** 15](#_Toc122293699)

[**3.2.4. Thi công phần mềm** 17](#_Toc122293700)

[**Chương 4. KẾT QUẢ VÀ ĐÁNH GIÁ** 22](#_Toc122293701)

[**4.1. Kết quả** 22](#_Toc122293702)

[**4.2. Đánh giá** 23](#_Toc122293703)

[**4.2.1. Các mặt đã đạt được** 23](#_Toc122293704)

[**4.2.2. Các mặt hạn chế** 23](#_Toc122293705)

[**4.2.3. Hướng phát triển** 23](#_Toc122293706)

[**TÀI LIỆU THAM KHẢO** 24](#_Toc122293707)

# **DANH MỤC ẢNH**

***Hình 2-1.*** *Sơ đồ khối hệ thống.................................................................................****3***

***Hình 2-2.*** *Board Arduino UNO R3..........................................................................****4***

***Hình 2-3.*** *Module RC522........................................................................................****5***

***Hình 2-4.*** *Sơ đồ nguyên lý toàn mạch......................................................................****6***

***Hình 2-5.*** *Ý tưởng thiết kế giao diện quản lý việc chấm công...................................****7***

***Hình 2-6.*** *Ý tưởng thiết kế giao diện thêm nhân viên mới vào công ty......................****8***

***Hình 2-7.*** *Ý tưởng thiết kế giao diện chỉnh sửa thông tin của nhân viên...................****9***

***Hình 2-8.*** *Ý tưởng thiết kế giao diện xóa nhân viên khỏi cơ sở dữ liệu.....................****9***

***Hình 2-9.*** *Ý tưởng thiết kế giao diện kiểm kê tình hình làm việc của nhân viên......****10***

***Hình 3-1.*** *PCB khối xử lý.......................................................................................****11***

***Hình 3-2.*** *Mạch in khối xử lý.................................................................................****12***

***Hình 3-3.*** *Phần cứng khối xử lý sau khi thi công*

***Hình 3-4.*** *Lưu đồ thuật toán khối xử lý..................................................................****13***

***Hình 3-5.*** *Version Tree – quá trình phát triển phần mềm của đồ án......................****15***

***Hình 3-6.*** *Một phần của cở sở dữ liệu trên SSMS..................................................****15***

***Hình 3-7.*** *Kiến trúc ADO.NET..............................................................................****16***

***Hình 3-8.*** *Giao diện quản lý việc chấm công.........................................................****17***

***Hình 3-9.*** *Giao diện thêm nhân viên mới vào công ty............................................****18***

***Hình 3-10.*** *Giao diện chỉnh sửa thông tin của nhân viên.......................................****18***

***Hình 3-11.*** *Giao diện xóa nhân viên khỏi cơ sở dữ liệu..........................................****19***

***Hình 3-12.*** *Giao diện kiểm kê tình hình làm việc của nhân viên.............................****19***

***Hình 3-13.*** *Lưu đồ thuật toán phần mềm quản lý chấm công.................................****20***

# **DANH MỤC BẢNG**

**Bảng 3.1.** Thông tin linh kiện sử dụng để thi công phần cứng................................**11**

**Bảng 3.2.** Sơ đồ kết nối, define chân kết nối giữa Arduino với RC522................**14**

# **DANH MỤC TỪ VIẾT TẮT**

|  |  |
| --- | --- |
| **Từ viết tắt** | **Ý nghĩa** |
| RFID | Radio Frequency Identifier |
| OOP | Object Oriented Programming |
| SSMS | SQL Server Management Studio |
| C# | C Sharp, C thăng, CSS |
| Microsoft VS | Microsoft Visual Studio |
| ADO.NET | ActiveX Data Objects |

# **TÓM TẮT**

Mục đích của đề tài là tạo ra một hệ thống giúp các doanh nghiệp, công ty vừa và nhỏ có thể tự động hóa quá trình điểm danh, chấm công của nhân viên khi đi làm. Giúp người quản lí, chủ doanh nghiệp có thể quản lí được doanh nghiệp mình tốt hơn, tối ưu lợi nhuận.

Đề tài sử dụng việc xử lí và truyền nhận tín hiệu từ thẻ RFID, đi qua khối trung tâm xử lsi tín hiệu là Arduino Uno và RC522, sau đó đưa lên máy tính xử lí qua các lớp front end, backend và cơ sở dữ liệu để lưu trữ dữ liệu lên SQL server.

Điểm nổi bậc của hệ thống là đơn giản về dễ sử dụng cho người dùng, hoạt động được 24/24, có tính ứng dụng.

***CHƯƠNG 1***

# **DẪN NHẬP**

## **1.1. Lý do chọn đề tài**

Trong thời đại hiện nay, việc tự động hóa đã góp phần rât quan trọng trong cuộc sống, không chỉ trong công nghiệp, nhà máy mà còn đến từng ngóc ngách mà chúng ta sống. Từ việc sản xuất, lắp ráp ở nhà máy, việc đi lại, mua sắm thường ngày,... và trong đó, việc chấm công cũng vậy. Thay vì phải chấm công bằng tay mỗi ngày, bây giờ con người cần một hệ thống tự động làm việc đó để không thời gian chấm công, tiện lợi, giảm đi nhân lực và tiền của cho việc quản lý việc chấm công,...

Từ những yêu cầu đó, kết hợp với việc cần hoàn thành đồ án môn học 2 về Vi Điều Khiển, em đã đi đế quyết định sẽ thiết kế một hệ thống chấm công tự động sử dụng thẻ RFID, kết hợp với lập trình Arduino, thiết kế giao diện và cơ sở dữ liệu cho hệ thống... Đề tài đã được thầy ThS. Nguyễn Văn Hiệp thông qua và góp ý, giúp em có thêm cơ sở và động lực để chọn và thực hiện đề tài.

## **1.2. Mục tiêu của đề tài**

Thiết kế được một hệ thống chấm công tự động, khi nhân viên đi làm và tan làm chỉ cần quét thẻ, việc chấm công sẽ do hệ thống tính toán, làm việc cập nhật cơ sở dữ liệu.

Về phần cứng, thiết kế được phần cứng hoạt động chính xác, ổn định, bền bỉ vì có liên quan đến công sức và tiền bạc của nhân viên, ngoài ra còn phải nhỏ gọn, đẹp mắt để không làm mất đi vẻ thẩm mỹ ở nơi văn phòng làm việc.

Về phần mềm, thiết kế được phầm mềm cho hệ thống quản lí chấm công chuẩn, hiệu quả và thông minh để giúp người quản lí có thể dễ dàng nắm bắt tình hình đi làm của nhân viên, thời gian làm việc,... đồng thời bảo vệ được quyền lợi từ nổ lực của nhân viên để từ đó đưa ra các quyết định khen thưởng, lương bổng, phạt,...

Thông qua đề tài, tự bản thân nắm vững kiến thức về lập trình vi xử lí, lập trình phần mềm, thiết kế và sử dụng cơ sở dữ liệu,...

## **1.3. Giới hạn của đề tài**

Hệ thống chấm công có quy mô sử dụng trong công ty, chủ yếu dùng chuẩn kết nối USB và mạng LAN vì sử dụng cơ sở dữ liệu SQL offline.

Nhân viên chỉ xem được tình hình chấm công của mình khi liên hệ với nhân sự do hệ thống không đăng tải lên internet.

## **1.4. Kế hoạch nghiên cứu**

Tuần 1 (03/10 – 09/10),viết tóm tắt yêu cầu đề tài đã chọn: đề tài làm cái gì, nội dung thiết kế, các thông số của đề tài.

Tuần 2 (10/10 -16/10), thiết kế sơ đồ khối, giải thích chức năng các khối.

Tuần 3 (17/10 – 23/10): Thiết kế sơ đồ mạch, nguyên lý hoạt động của mạch, thiết kế mạch thử nghiệm bằng dây cắm. Bắt đầu tìm hiểu về C# và SQL Server để thiết kế phần mềm cho hệ thống.

Tuần 10 (24/10 – 30/10), thiết kế cơ sở dữ liệu trên SQL server.

Tuần 11, 12 (31/10 – 13/11), thiết kế phần mềm cho hệ thống dựa trên C# (.NET Winforms framework).

Tuần 13 (14/11 – 20/11), thi công mạch in và viết báo cáo những nội dung đã làm.

Tuần 14 (21/11 – 27/11), hoàn thiện báo cáo và gởi cho GVHD để xem xét góp ý lần cuối trước khi in và báo cáo.

Tuần 15, 16, 17 (28/11 – 18/12), nộp quyển báo cáo, làm Slide thuyết trình và tiến hành báo cáo với GVHD.

***CHƯƠNG 2***

# **THIẾT KẾ**

## **2.1. Bài toán thiết kế**

### **2.1.1. Mục tiêu về phần cứng**

Thiết kế nhỏ gọn, tinh tế để gắn ở vị trí ra vào của nhân viên/khách hàng sẽ không gây mất thẩm mỹ.

Thiết kế đơn giản, dễ sửa chữa, bảo trì, bào dưỡng.

Truyền nhận dữ liệu tốt với máy tính.

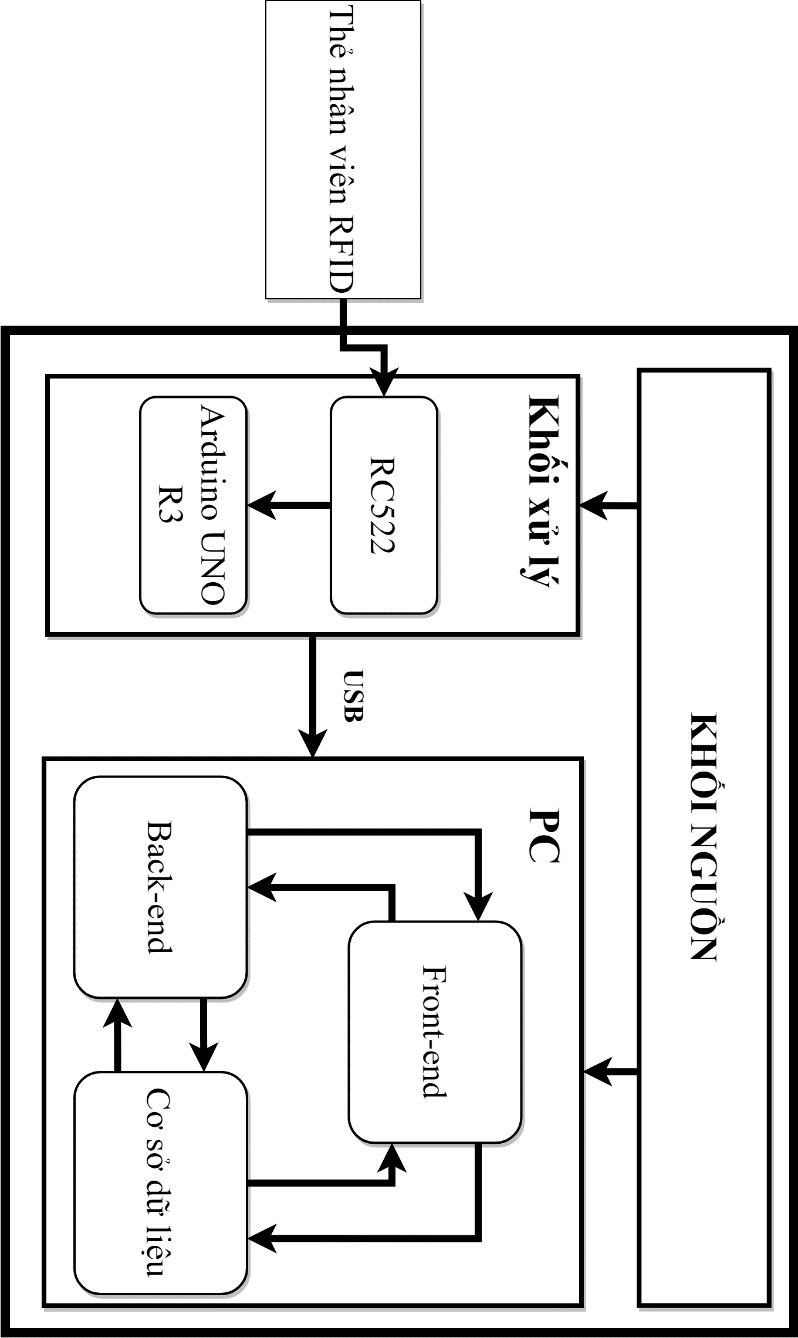
### **2.1.2. Mục tiêu về phần mềm**

Về phần mềm cho vi điều khiển, phải giúp vi điều khiển giao tiếp được và tốt với thẻ RFID, máy tính, module RC522. Ngoài ra phải ngắn gọn,

Về phần giao diện viết trên C#, giao diện phải trực quan, dễ hiểu, người quản lí có thể dễ dàng sử dụng để quản lí nhân viên của mình cũng như là quản lý, tương tác với cơ sở dữ liệu.

Cơ sở dữ liệu phải trực quan, giúp người quản lí nắm được tình hình làm việc, sự chăm chỉ của nhân viên.

## **2.2. Sơ đồ khối**



***Hình 2-1.*** *Sơ đồ khối hệ thống*

***Chức năng của từng khối***

*Khối xử lý*

Khối này có chức năng nhận dữ liệu mã UID từ thẻ RFID của nhân viên, xử lý và tiến hành gửi dữ liệu đến máy tính.

*Khối máy tính PC*

Khối này là một phần mềm “full-stack”, có đầy đủ frontend, backend và cơ sở dữ liệu, có chức năng nhận dữ liệu từ khối xử lý, tiến hành xử lý ở backend, lấy và chỉnh sửa dữ liệu ở cơ sở dữ liệu, sau đó hiển thị lên frontend (nếu có).

Khối nguồn

Khối này có chức năng cung cấp nguồn hoạt động cho cả hệ thống.

## **2.3. Thiết kế của từng khối**

### **2.3.1. Khối xử lý**

Yêu cầu: Do hệ thống phải hoạt động cả ngày, cho nên khối này phải bền bỉ, dễ bảo trì, dễ sử dụng, nhỏ gọn và không tiêu thụ quá nhiều năng lượng.

Khối xử lý không cần phải có hiệu suất quá cao, nhưng phải bền bỉ, hoạt động cả ngày. Một số vi điều khiển có thể áp dụng để sử dụng như Raspberry Pi 4, STM32, PIC,... nhưng để thực hiện đề tài này, tôi chọn Board Arduino UNO R3 kết hợp với module đầu đọc thẻ RFID RC522.

Lý do là vì Board Arduino UNO R3 và module RC522 nhỏ gọn, tiêu thụ điện năng thấp, có thể cung cấp điện năng thông qua cáp USB mà không cần cắm thêm nguồn. Ngoài ra, bên phía nhà sản xuất cũng cung cấp thêm một IDE - Arduino IDE, giúp việc lập trình trở nên đơn giản, dễ sử dụng, dễ bảo trì trên tất cả mọi máy tính.



***Hình 2-2.*** *Board Arduino UNO R3*

***Thông số kỹ thuật chi tiết của Board Aruino UNO R3:***

- Vi xử lý chính: Atmega328P.

- Điện áp nguồn nuôi: 7-12V.

- Giới hạn điện áp: 20V.

- Có 14 chân tín hiệu số, 6 chân tính hiệu tương tự và 6 chân điều chế xung PWM (3, 5, 6, 9, 10 và 11).

- Dòng ra của các chân IO: 40mA.

- Dòng ra chân 3.3V: 50mA.

- Bộ nhớ Flash: 32kB.

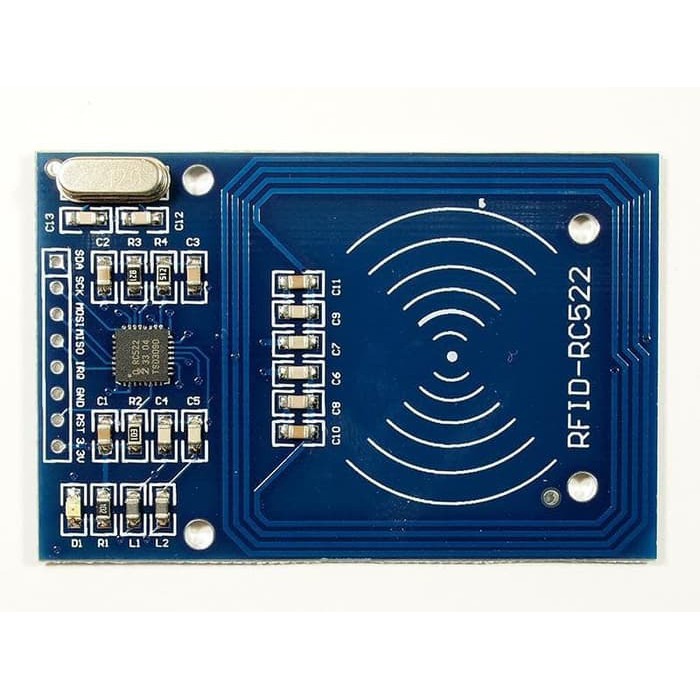
- SRAM: 2kB.

- EEPROM: 1kB.

- Tần số thạch anh: 16MHz.

- Kích thước: 68.6mm x 53.4 mm.

Bên cạnh các thông số trên, Board còn được lập trình như một bộ chuyển đổi USB sang chuẩn nối tiếp và còn có các chân giao tiếp SPI: 10 SS, 11 MOSI, 12 MISO, 13 SCK dùng để truyền phát dữ liệu bằng giao thức SPI với các thiết bị khác, đây là một trong những tính năng cần thiết để kết nối Board với Module RC522.



***Hình 2-3.*** *Module RC522*

***Thông số kỹ thuật chi tiết của module RC522:***

- Nguồn sử dụng: 3.3VDC.

- Dòng điện: 13~26mA.

- Tần số hoạt động: 13.56Mhz.

- Khoảng cách hoạt động: 0~60mm.

- Chuẩn giao tiếp: SPI

- Tốc độ truyền dữ liệu: tối đa 10Mbit/s

- Các loại card RFID hỗ trợ: mifare1 S50, mifare1 S70, mifare UltraLight, mifare Pro, mifare Desfire

- Kích thước: 40mm × 60mm.

Module RC522 cùng với Board Arduino UNO rất phổ biến trên thị trường nhờ vào giá cả, độ bền, độ tiện dụng cũng như có một cộng đồng người dùng hỗ trợ rộng lớn.

### **2.3.2. Thiết kế khối nguồn**

Đối với Arduino, chỉ cần sử dụng nguồn được lấy từ cáp USB kết nối với máy tính, cáp USB ngoài việc giúp hệ thống phần cứng giao tiếp với phần mềm, còn giúp hệ cung cấp nguồn cho toàn bộ phần cứng.

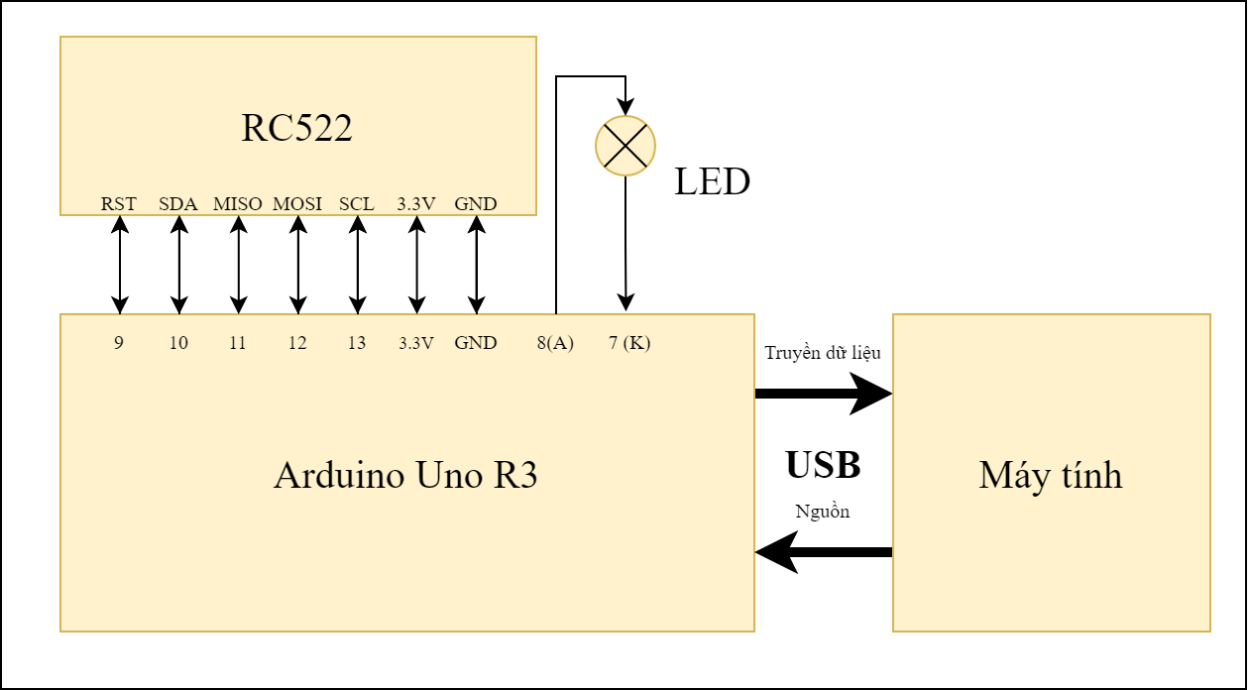
Đối với RC52, sẽ sử dụng nguồn từ chân 3.3VDC của Board Arduino.

Còn đối với máy tính, chỉ cần sử dụng Adapter sạc cùng cáp sạc do nhà sản xuất cung cấp kèm theo máy tính alaf có thể giúp máy tính hoạt động ổn định, đồng thời cung cấp nguồn cho Board Arduino thông qua cáp USB.

### **2.3.3. Sơ đồ nguyên lý toàn mạch (Phần cứng)**

Để có thể nhận dữ liệu từ RC522 và làm sáng led khi có người quét thẻ, chúng ta sẽ sử dụng các chân GPIO của Board Arduino.

Tiếp theo, để cung cấp nguồn cho toàn mạch, chúng ta sẽ sử dụng cáp USB kết nối với máy tính (như đã phân tích ở mục 2.3.2).



***Hình 2-4.*** *Sơ đồ nguyên lý toàn mạch*

### **2.3.3. Thiết kế phần mềm (Trên khối PC)**

***Thiết kế phần mềm (Phần này chỉ trình bày về frontend, còn backend và cơ sở dữ liệu sẽ được trình bày ở phần thi công).***

Phần mềm trên máy tính có chức năng quản lý việc điểm danh, quản lý nhân viên trong công ty và hiển thị được thông tin của nhân viên khi kiểm kê.

Cả phầm mềm sẽ có tổng cộng 5 giao diện làm việc chính:

+ Giao diện quản lý việc chấm công.

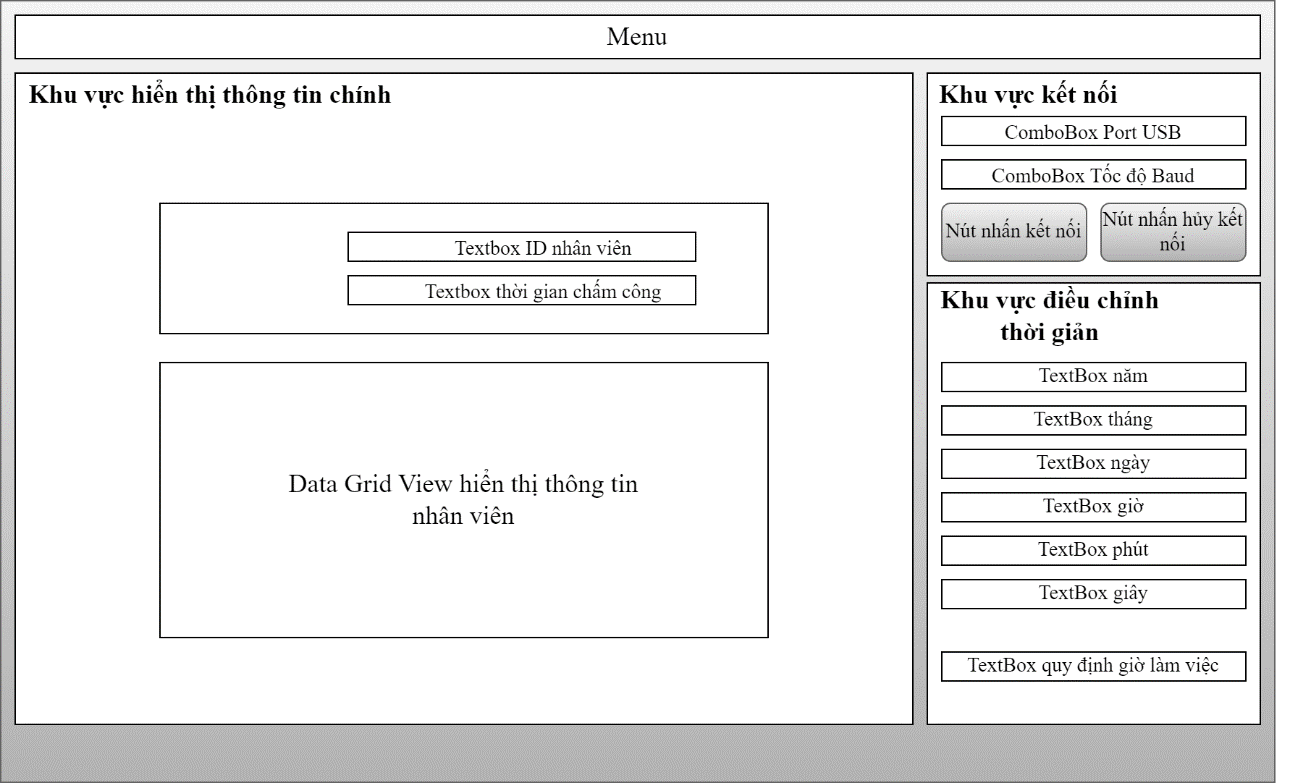
+ Giao diện thêm nhân viên mới vào công ty.

+ Giao diện chỉnh sửa thông tin nhân viên.

+ Giao diện xóa nhân viên khỏi cơ sở dữ liệu.

+ Giao diện kiểm kê tình hình làm việc của nhân viên.

***Giao diện quản lý việc chấm công***

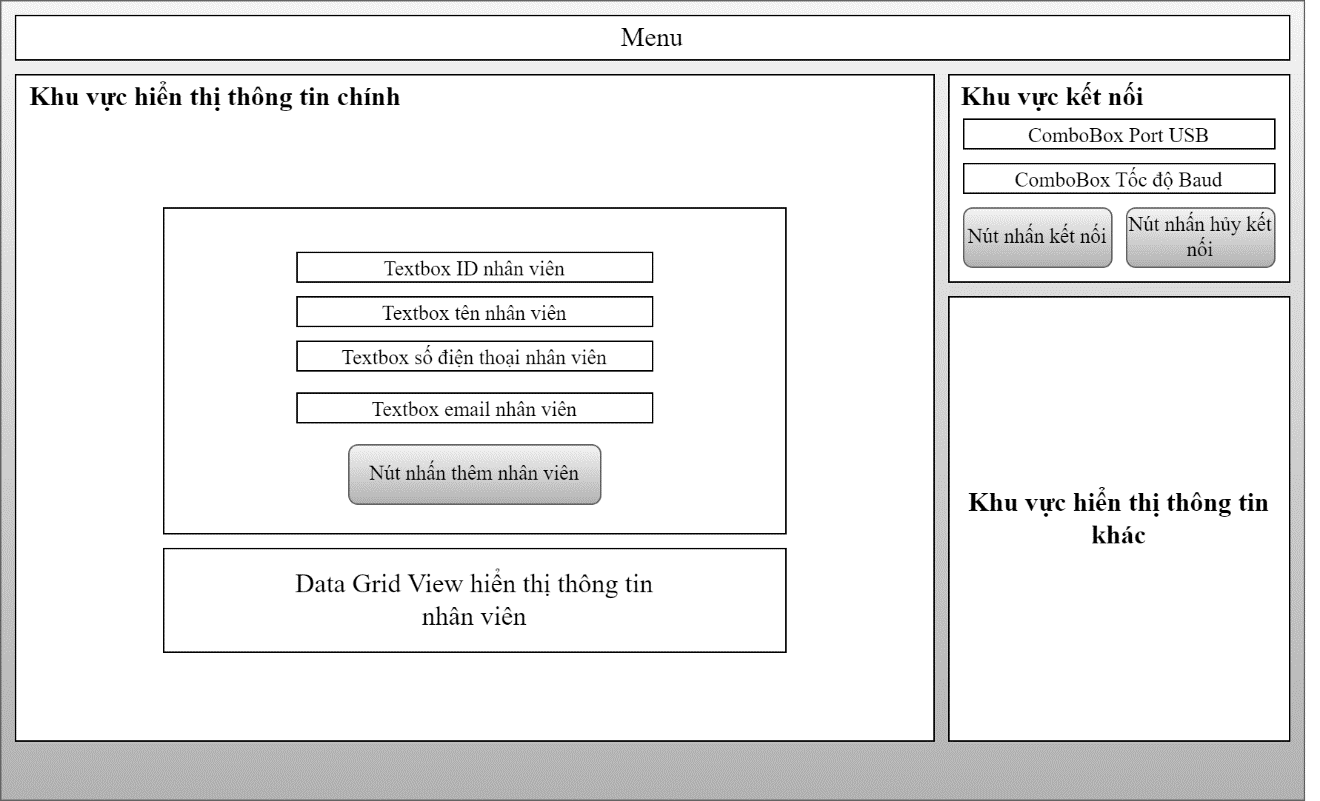


***Hình 2-5.*** *Ý tưởng thiết kế giao diện quản lý việc chấm công*

Giao diện quản lý việc chấm công có chức năng giúp người quản lý kiểm soát được thông tin của nhân viên khi nhân viên chấm công vào công ty. Khi đó thông tin của nhân viên sẽ được hiển thị chi tiết trên Data Grid View.

Giao diện còn cung cấp thêm việc kết nối giữa máy tính với bộ xử lý (Arduino) ỏ khu vực kết nối. Ngoài ra, do là phần dùng để báo cáo, để có thể điều chỉnh nhiều trường hợp thời gian, phần mêm còn phải có thêm khu vục điều chỉnh thời gian.

***Giao diện thêm nhân viên mới vào công ty***



***Hình 2-6.*** *Ý tưởng thiết kế giao diện thêm nhân viên mới vào công ty*

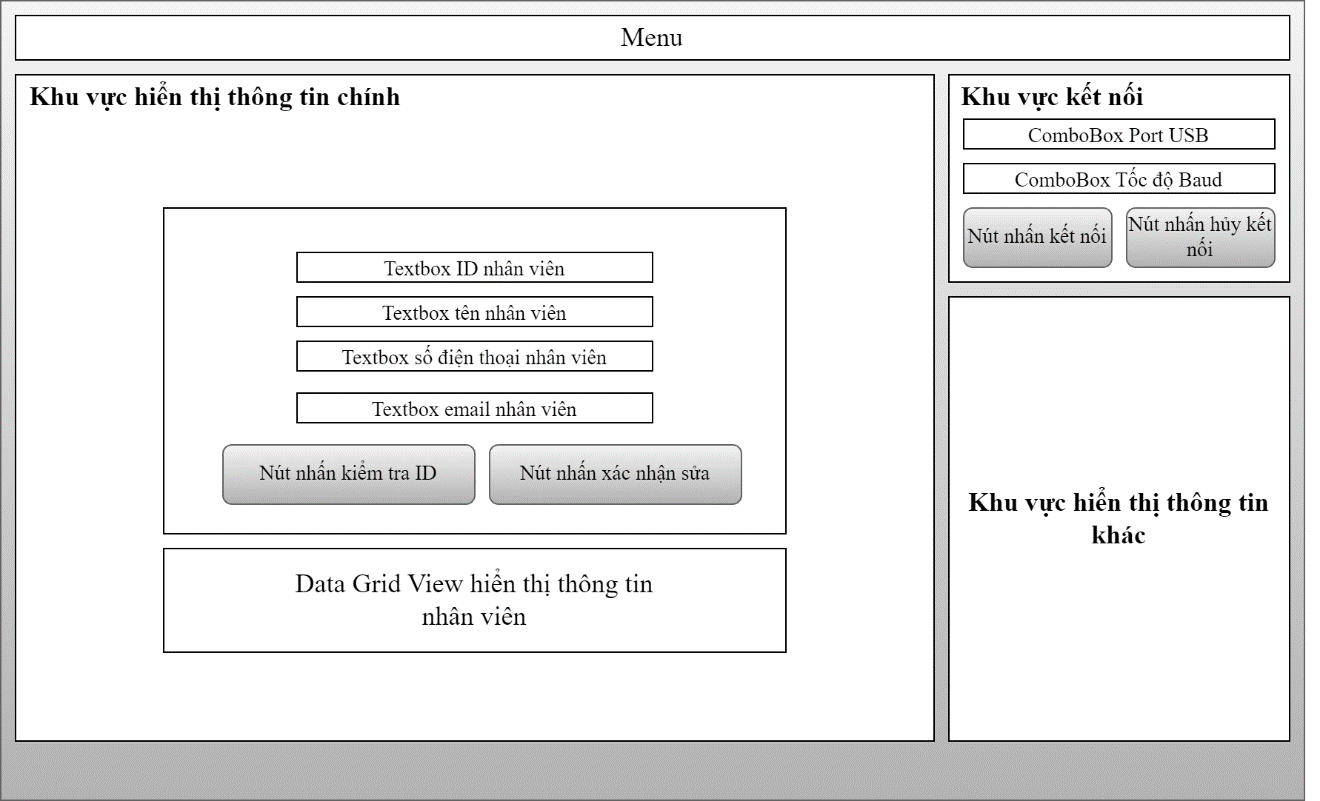
Giao diện thêm nhân viên mang đến chức năng thêm gián tiếp nhân viên mới của công ty vào cơ sở dữ ngay trên phần mềm mà không cần người dùng can thiệp trực tiếp vào cơ sở dữ liệu. Sau khi điền thông tin, nhấn nút nhấn thêm nhân viên để xác nhận việc thêm nhân viên vào cơ sở dữ liệu. Lúc này Data Grid View sẽ hiển thị chi tiết cơ sở dữ liệu để chúng ta xem nhân viên mới đã được thêm vào chưa.

Giao diện còn cung cấp thêm việc kết nối giữa máy tính với bộ xử lý (Arduino) ở khu vực kết nối.

***Giao diện chỉnh sửa thông tin của nhân viên***

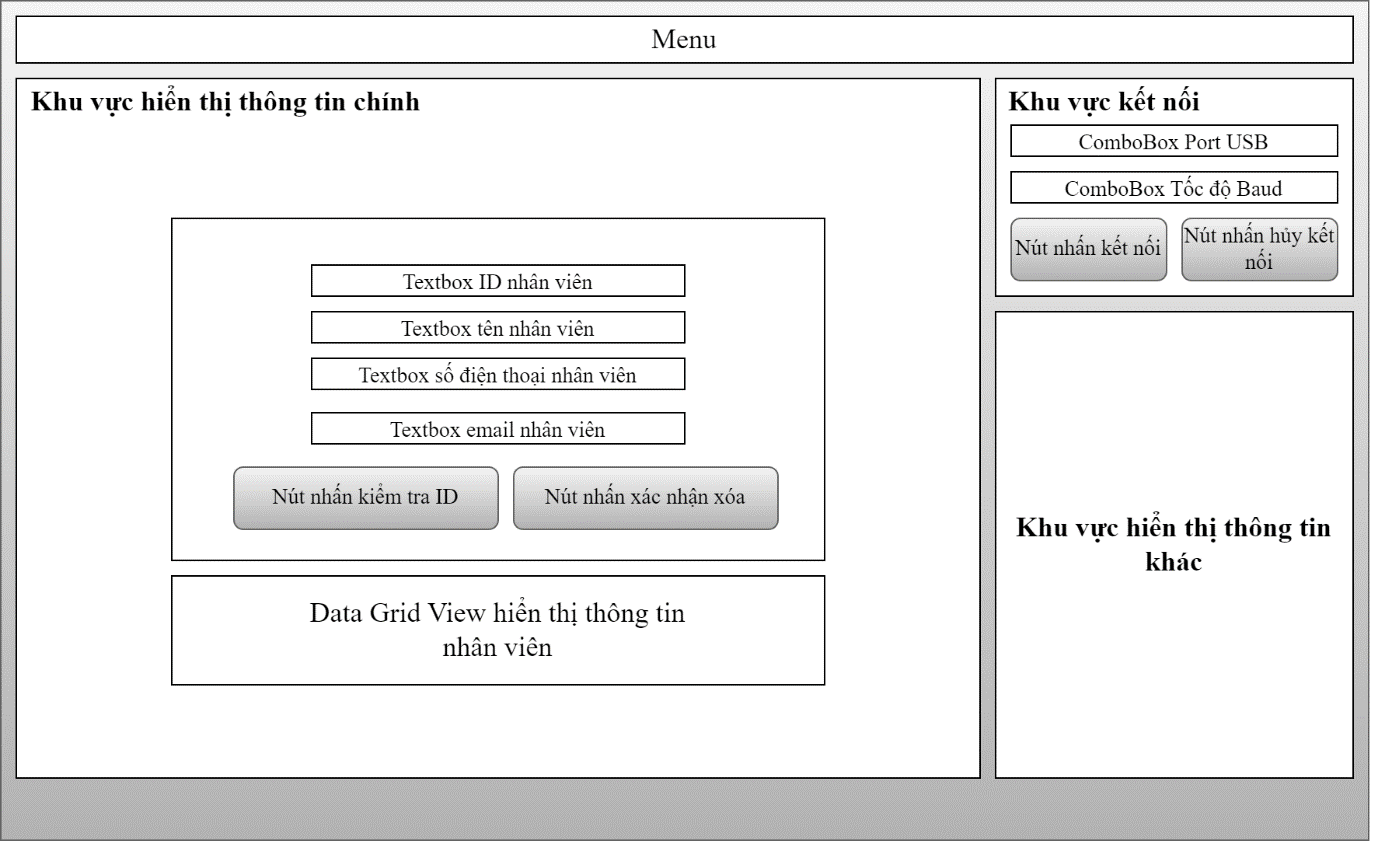
Giao diện chỉnh sửa nhân viên mang đến chức năng chỉnh sửa gián tiếp thông tin của nhân viên vào cơ sở dữ ngay trên phần mềm mà không cần người dùng can thiệp trực tiếp vào cơ sở dữ liệu. Sau khi điền thông tin, nhấn nút nhấn chỉnh sửa thông tin nhân viên để xác nhận việc chỉnh sửa. Lúc này Data Grid View sẽ hiển thị chi tiết cơ sở dữ liệu để chúng ta xem thông tin nhân viên đã được cập nhật chưa.

Giao diện còn cung cấp thêm việc kết nối giữa máy tính với bộ xử lý (Arduino) ở khu vực kết nối.



***Hình 2-7.*** *Ý tưởng thiết kế giao diện chỉnh sửa thông tin của nhân viên*

***Giao diện xóa nhân viên khỏi cơ sở dữ liệu***

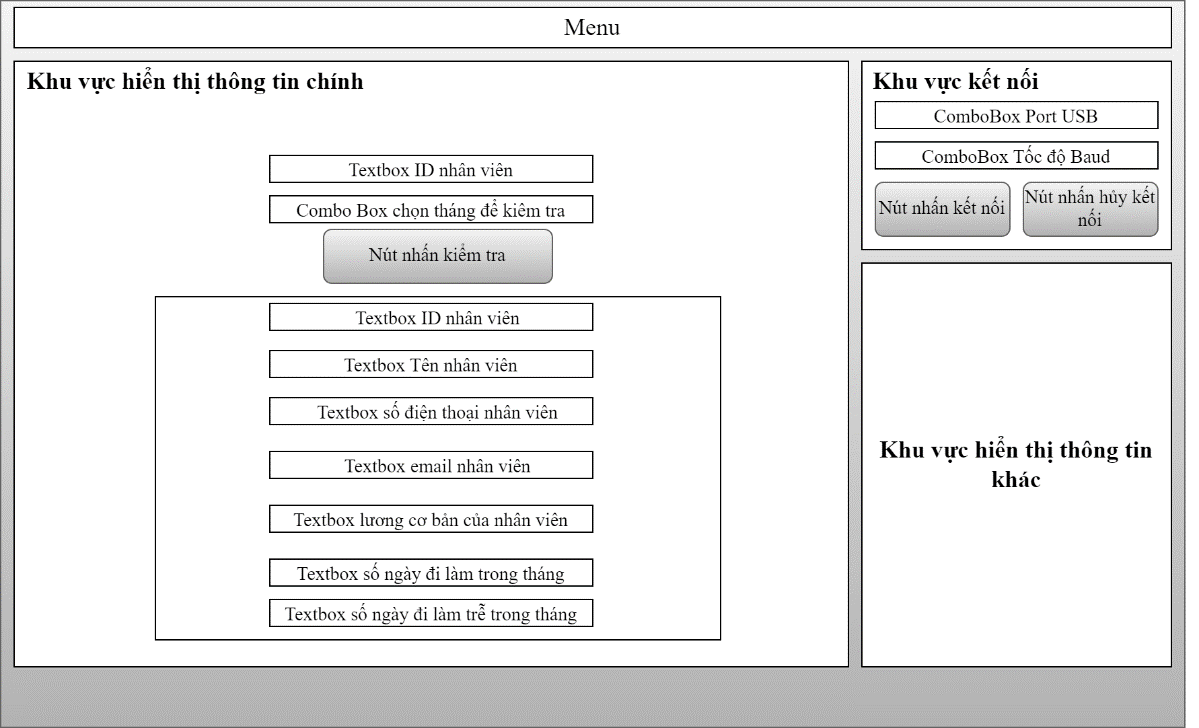


***Hình 2-8.*** *Ý tưởng thiết kế giao diện xóa nhân viên khỏi cơ sở dữ liệu*

Giao diện xóa nhân viên mang đến chức năng xóa gián tiếp một nhân viên khỏi cơ sở dữ ngay trên phần mềm mà không cần người dùng can thiệp trực tiếp vào cơ sở dữ liệu. Sau khi điền thông tin, nhấn nút nhấn xóa nhân viên để xác nhận việc xóa. Lúc này Data Grid View sẽ hiển thị chi tiết cơ sở dữ liệu để chúng ta xem nhân viên đã bị xóa chưa.

Giao diện còn cung cấp thêm việc kết nối giữa máy tính với bộ xử lý (Arduino) ở khu vực kết nối.

***Giao diện kiểm kê tình hình làm việc của nhân viên***



***Hình 2-9.*** *Ý tưởng thiết kế giao diện kiểm kê tình hình làm việc của nhân viên*

Giao diện kiểm kê tình hình làm việc của nhân viên mang đến chức năng kiểm tra sự chăm chỉ của nhân viên. Ngoài việc thể xem được thông tin chi tiết của nhân viên, còn hiển thị thêm lương cơ bản, số ngày làm việc trong tháng, số ngày đi làm trễ trong tháng.

Giao diện còn cung cấp thêm việc kết nối giữa máy tính với bộ xử lý (Arduino) ở khu vực kết nối.

***Chương 3.***

# **THI CÔNG**

## **3.1. Thi công phần cứng**

### **3.1.1. Thi công mạch in**

Đầu tiên là chuẩn bị các linh kiện cần thiết, để đảm bảo đề tài hoàn thành ở chất lượng cao, có thể hoạt động bền bỉ, không bị lỗi về phần cứng, tôi đã mua những linh kiện tốt nhất có thể trong tầm giá cũng như trong khả năng kinh tế của bản thân từ một vendor nước ngoài.

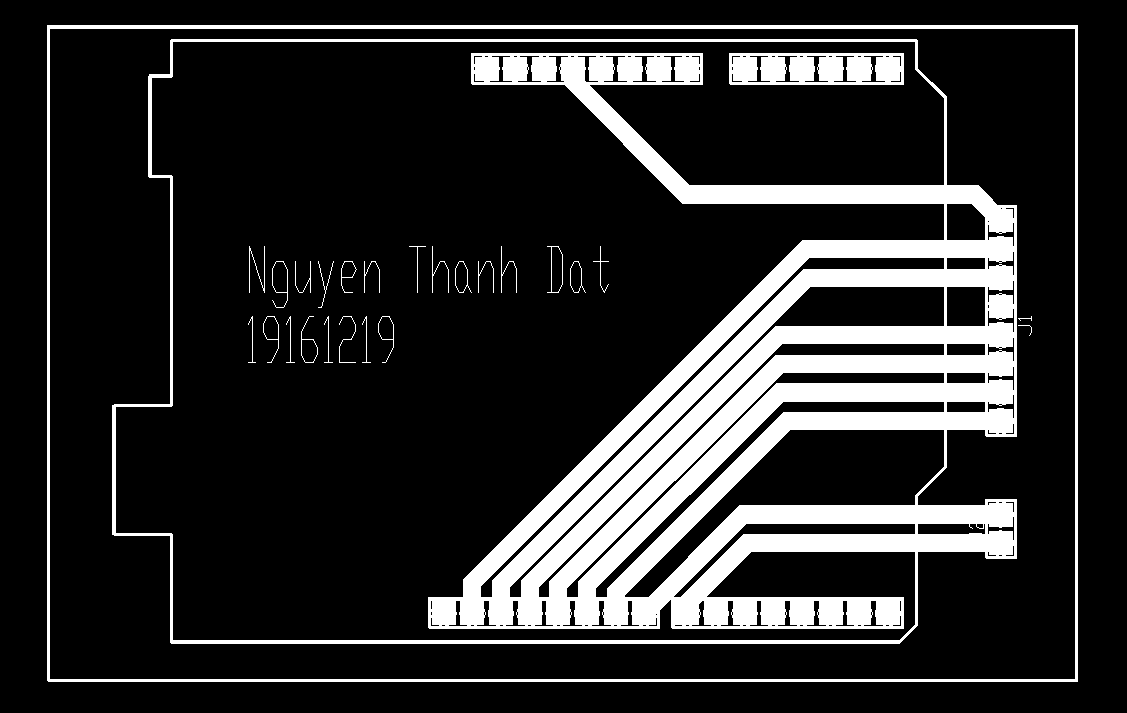
Các linh kiện được liệt kê theo bản bên dưới

**Bảng 3.1.** Thông tin linh kiện sử dụng để thi công phần cứng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên linh kiện** | **Kích thước** | **Số lượng** | **Giá** |
| 1 | Board Arduino Uno R3 | 55mm x 75mm | 1 | $10.99 |
| 2 | Module RC522 | 37.5mm x 33mm | 1 | $1.99 |
| 3 | Cáp USB Type B - A | 300mm | 1 | $0.99 |

Để phục vụ cho việc vẽ mạch in, đồ án này sử dụng phần mềm vẽ mạch in là Proteus vì sự đơn giản, tiện dụng, giao diện hiện đại.

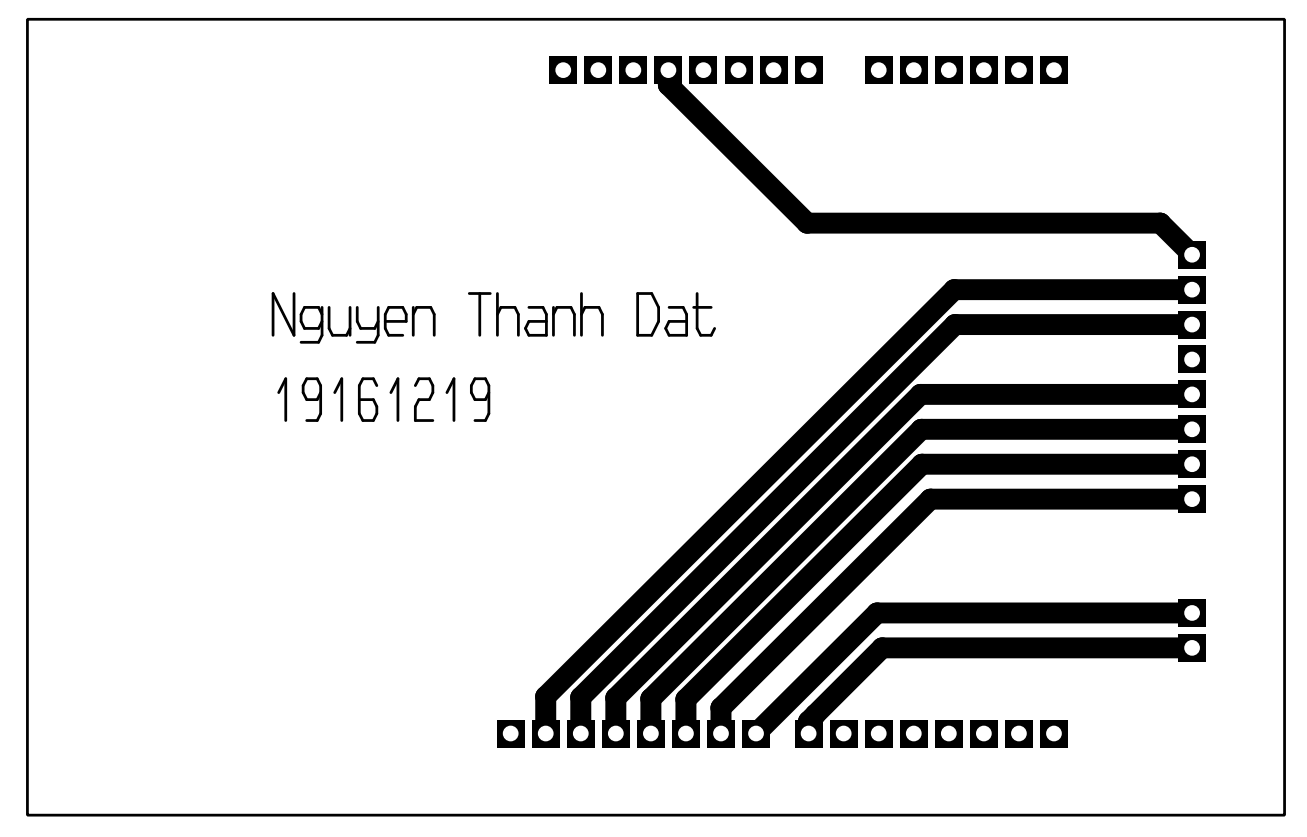
Sau khi thiết kế sơ đồ nguyên lý và đi dây, đề tài có mạch PCB như hình 3-1.



***Hình 3-1.*** *PCB khối xử lý*

### **3.1.2. Hàn linh kiện**

Bên dưới là bản mạch in trên PDF trước khi in được lên bảng đồng. Sau khi in lên bảng đồng thì tôi sẽ tiền hành khoan và hàn các linh kiện lên mạch.



***Hình 3-2.*** *Mạch in khối xử lý*

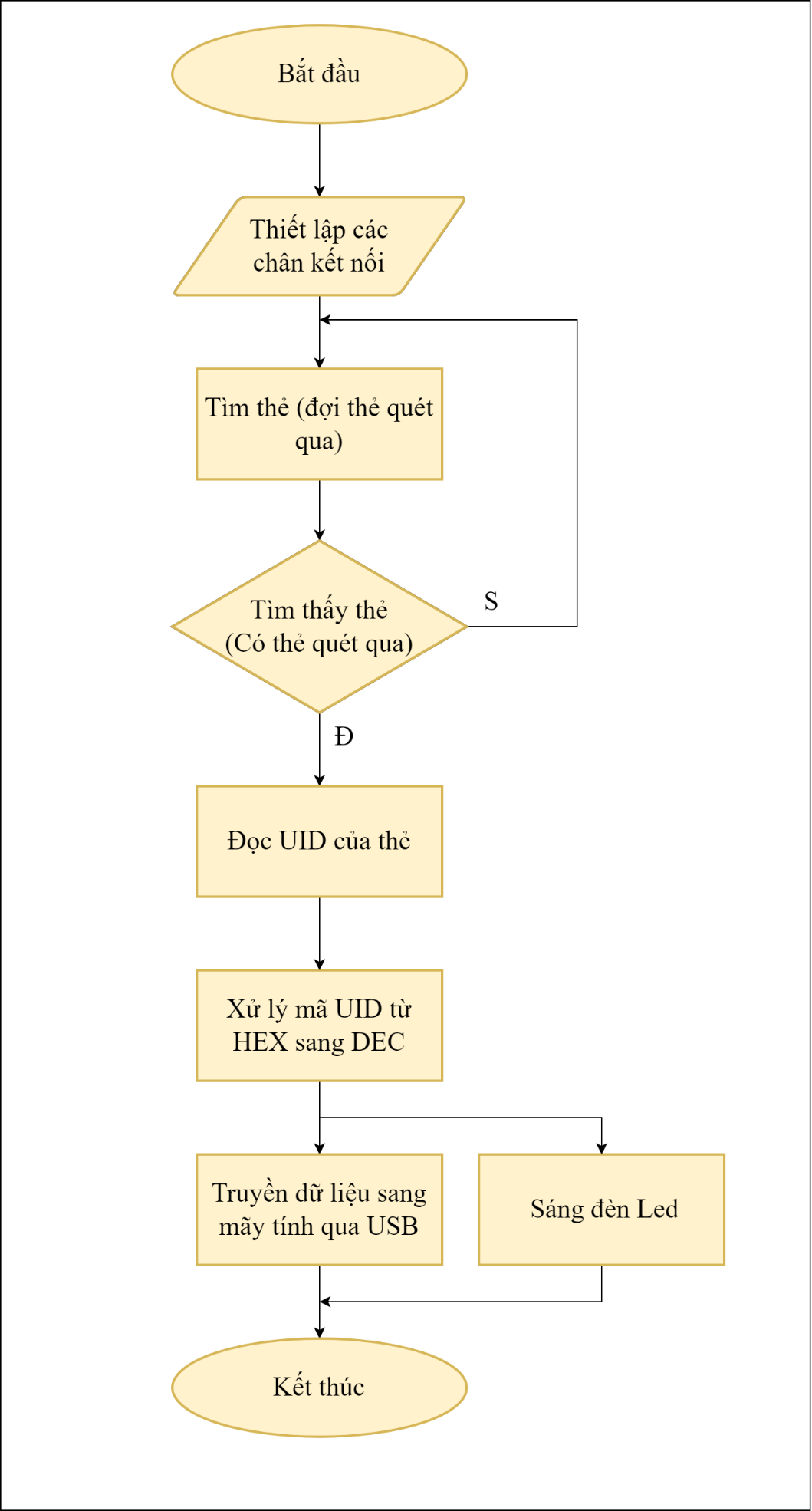
**3.1.3. Kết quả thi công phần cứng**

Sau khi in mạch, làm mạch in, khoan lỗ và hàn linh kiện, tôi đã cơ bản hoàn thành xong mô hình “hộp chấm công”, hoàn thành gần như 100% yêu cầu về khối xử lý của đề tài.



***Hình 3-3.*** *Phần cứng khối xử lý sau khi thi công*

### **3.1.3. Lưu đồ điều khiển khối điều khiển**



***Hình 3-4.*** *Lưu đồ thuật toán khối xử lý*

***Giải thích lưu đồ***

Khi bắt đầu chương trình, các chân kết nối giữa Arduino và RC522 sẽ được define như bản bên dưới. Các kết nối bên dưới được quy định theo chuẩn kết nối SPI, chuẩn kết nối được sử dụng cho module RC522.

**Bảng 3.2.** Sơ đồ kết nối, define chân kết nối giữa Arduino với RC522

|  |  |
| --- | --- |
| **Chân RC522** | **Chân Arduino** |
| Reset | 9 |
| SPI SS | 10 |
| SPI MOSI | 11 |
| SPI MISO | 12 |
| SPI SCK | 13 |

Tiếp theo, khối xử lý sẽ đợi có người quét thẻ, nếu không có thì vẫn sẽ tiếp tục đợi, nếu có thì sau khi có nhân viên quét thẻ qua đầu đọc RC522, dữ liệu sẽ được truyền từ RC522 sang Arduino, sau đó Arduino sẽ mã hóa chuỗi dữ liệu từ mã HEX (thập lục phân) sang mã DEC (thập phân), rồi truyền sang cho máy tính (lúc này đèn Led sẽ sáng) để tiếp tục thực hiện trên backend, cơ sở dữ liệu và hiển thị lên frontend.

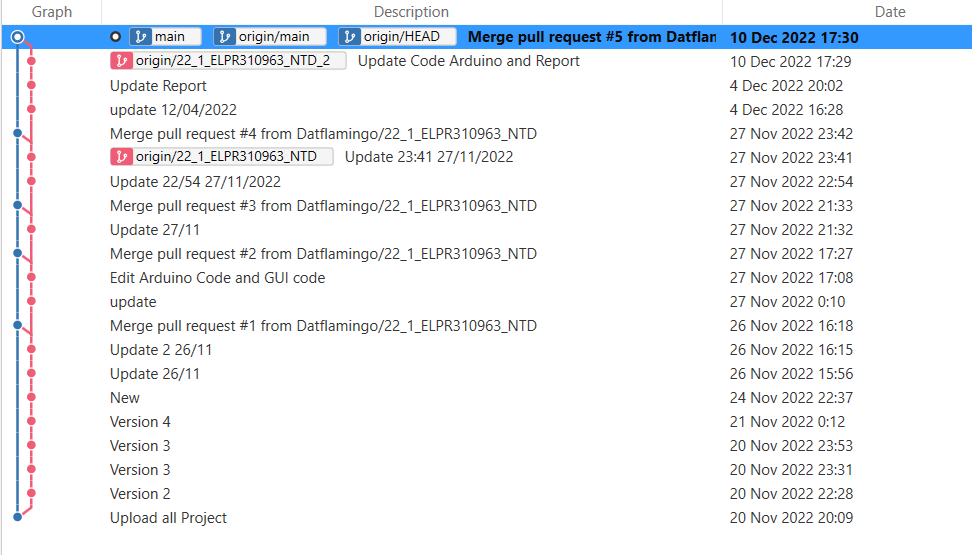
## **3.2. Thi công phần mềm**

### **3.2.1 Giới thiệu về Git và GitHub**

Git là một hệ thống quản lý phần mềm phân tán (Distributed Version Control System), được sinh ra để giúp các lập trình viên dễ dàng làm việc với các phiên bản của phần mềm từ phiên bản cũ đến phiên bản mới.

GitHub là một công cụ trợ giúp sử dụng Git, nó giúp chúng ta thay vì phải sử dụng những câu lệnh trên Command Prompt của Git, chúng ta chỉ cần sử dụng những cái click chuột lên các nút lệnh Push, Pull,...

Trong đồ án này, tôi sử dụng GitHub để phát triển phần mềm cho giao diện trên C# cũng như là mã nguồn cho khối xử lý. Nó giúp tôi có thể quản lý tới hơn 10 file code và hơn 3500 dòng code một cách khoa học và dễ dàng, giúp tôi dễ dàng tìm ra lỗi sai, bug và chỉnh sửa nó dễ dàng hơn.

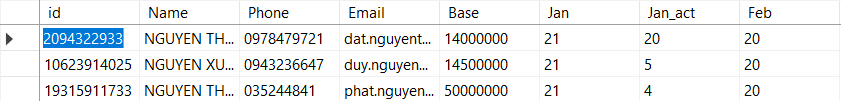
****

***Hình 3-5.*** *Version Tree – quá trình phát triển phần mềm của đồ án*

### **3.2.2. Tạo cơ sở dữ liệu**

Cơ sở dữ liệu của đề tài được thiết kế trên SQL Server Management Studio (SSMS).

Cơ sở dữ liệu được viết trên ngôn ngữ SQL và quản lý bởi SQL Server. Một nền tảng quản lý cơ sở dữ liệu được các nhà khoa học dữ liệu và nhà phân tích dữ liệu khuyên dùng bởi sự tiện dụng và có thể sẽ góp phần vào một trong những mũi nhọn của tương lai – Big Data.



***Hình 3-6.*** *Một phần của cở sở dữ liệu trên SSMS*

### **3.2.3. Giới thiệu ngôn ngữ và kiến trúc lập trình**

***Ngôn ngữ lập trình***

Để phục vụ cho việc viết phần mềm giao diện, tôi lập trình trên nền tảng .NET Framework với ngôn ngữ lập trình chính là C# (C sharp).

C# là một ngôn ngữ lập trình thuần hướng đối tượng được Microsoft thiết kế riêng để phục vụ cho việc phát triển web, phần mềm,... chủ yếu sử dụng trên nền tảng Window, MacOS. Nam 2021, C# được bình chọn là một trong những ngôn ngữ lập trình “đáng yêu nhất”, chỉ đứng sau Python.

***Kiến trúc lập trình***

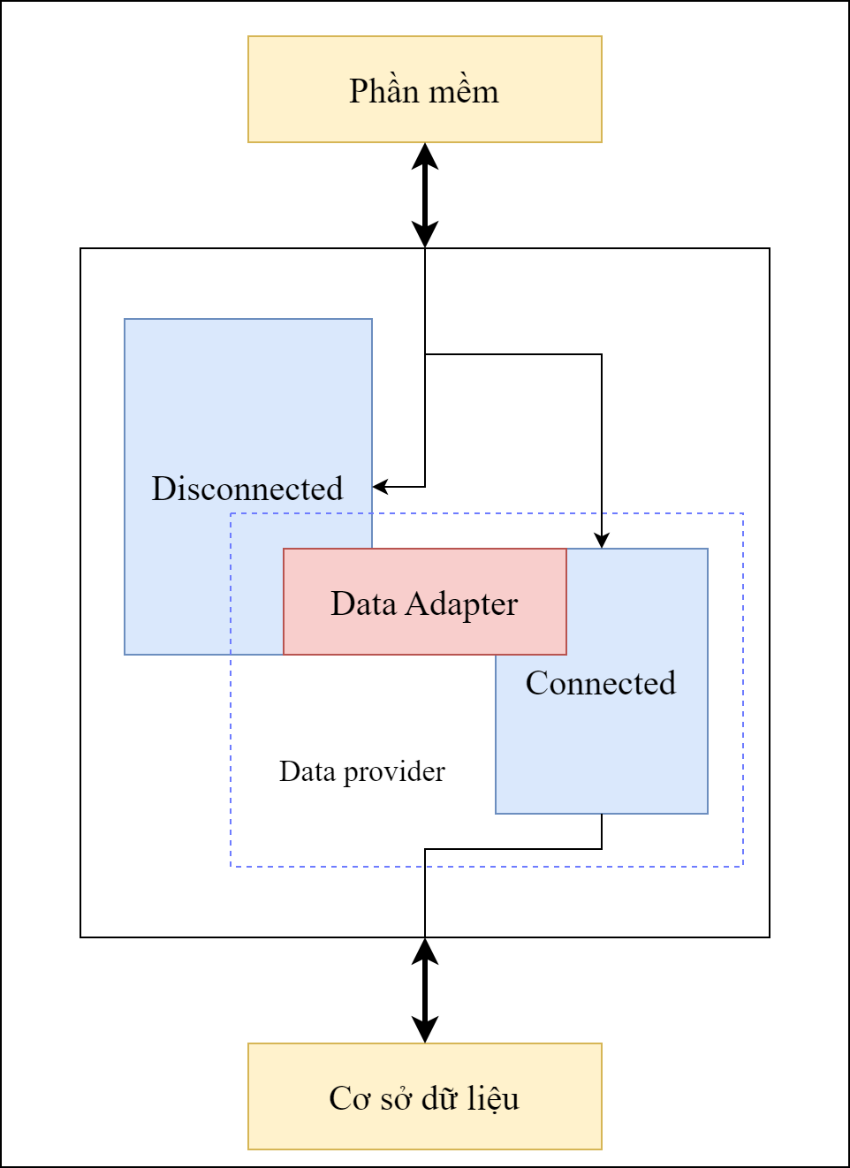
Phần mềm được phát triển dựa trên kiến trúc ADO.NET. ADO.NET là một kiến trúc 3 lớp, được xem như là người liên kết giữa phần mềm với cơ sở dữ liệu.

ADO.NET có thể xem có 3 thành phần chính (3 lớp):

+ Thành phần kết nối (Connected).

+ Thành phần cục bộ (Disconnected).

+ Thành phần đặc biệt: Data Adapter.



***Hình 3-7.*** *Kiến trúc ADO.NET*

Trong đồ án này, tôi sẽ sử dụng phương pháp chủ yếu là tương tác với thành phần Connected. Và thành phần này được phân loại theo các loại nguồn dữ liệu với nó, và sẽ cần một Class riêng để tương tác với nó và Class này sẽ được gọi là Data provider. (Đối với SQL Server sẽ là Sysrtem.Data.SqlClient).

### **3.2.4. Thi công phần mềm**

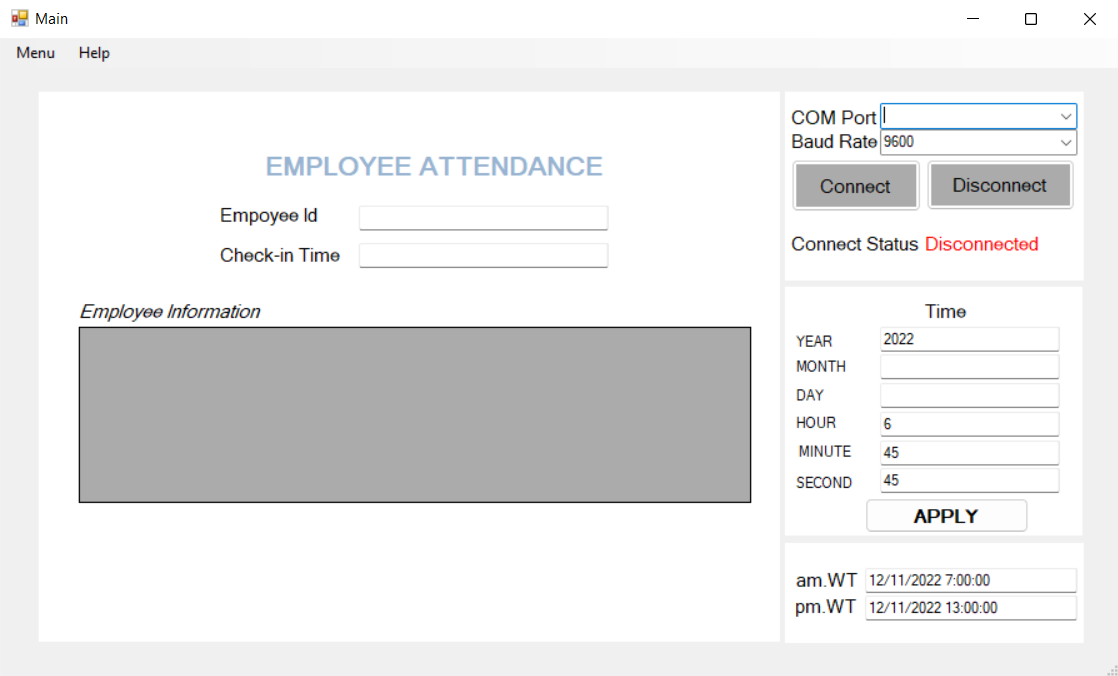
**3.2.4.1. Frontend**

Frontend là một phần của một website hoặc một phần mềm, nơi mà người dùng tương tác với nó, những gì mà bạn tương tác trên web hay một ứng dụng nào đó như các label, nút nhấn, combobox hay thậm chí là các phông chữ, cỡ chữ, hình ảnh,... nó không hề đơn giản như viết một văn bản trên Work mà nó sự một sự kết hợp hoàn hảo giữa các ngôn ngữ lập trình như HTML, C#, Java Script.

Đối với đề tài, frontend chính là phần giao diện người dùng – GUI và được lập trình chính trên ngôn ngữ C#. Một điểm hay của nền tảng .Net Framework là cả frontend và backend của nó đều được viết trên C#, nó giúp phần mềm trở nên dễ dàng hoạt động hơn, ít bị ảnh hưởng do sự bất đồng bộ giữa 2 ngôn ngữ lập trình (điều này dễ thấy với một số ứng dụng Android, khi mà frontend được viết bởi XML nhưng backend của nó lại được viết trên Java hay Kotlin).

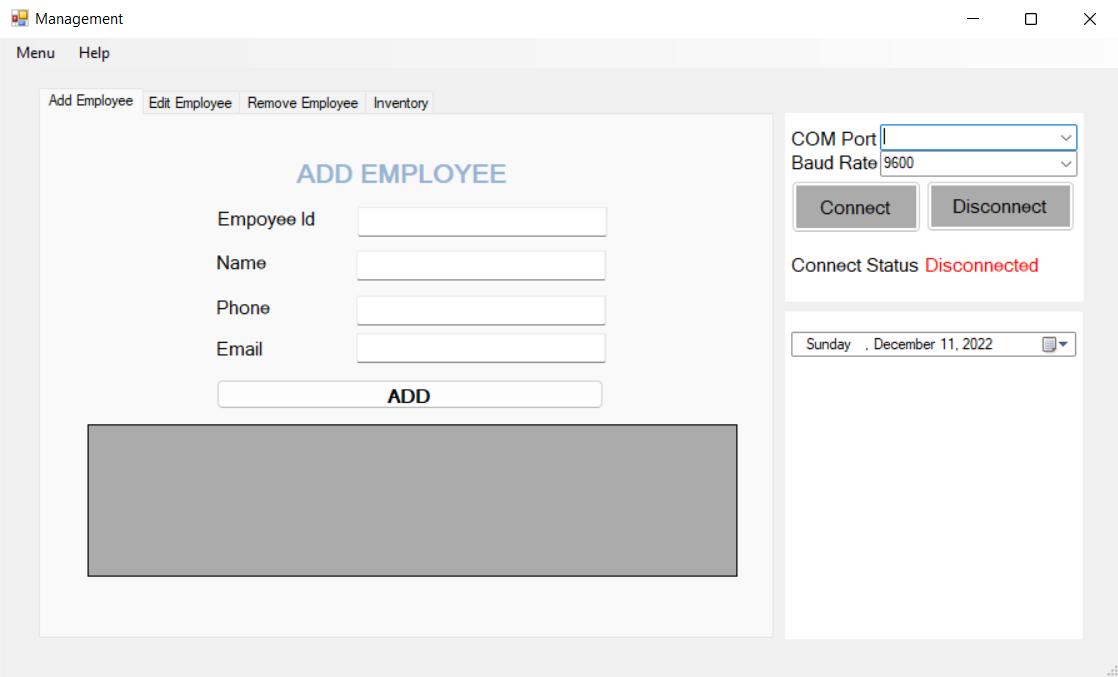
Và bên dưới đây chính là các thiết kế hoàn chỉnh về frontend của đề tài, được hoàn thiện theo như các yêu cầu ở phần thiết kế.

***Giao diện quản lý việc chấm công***

******

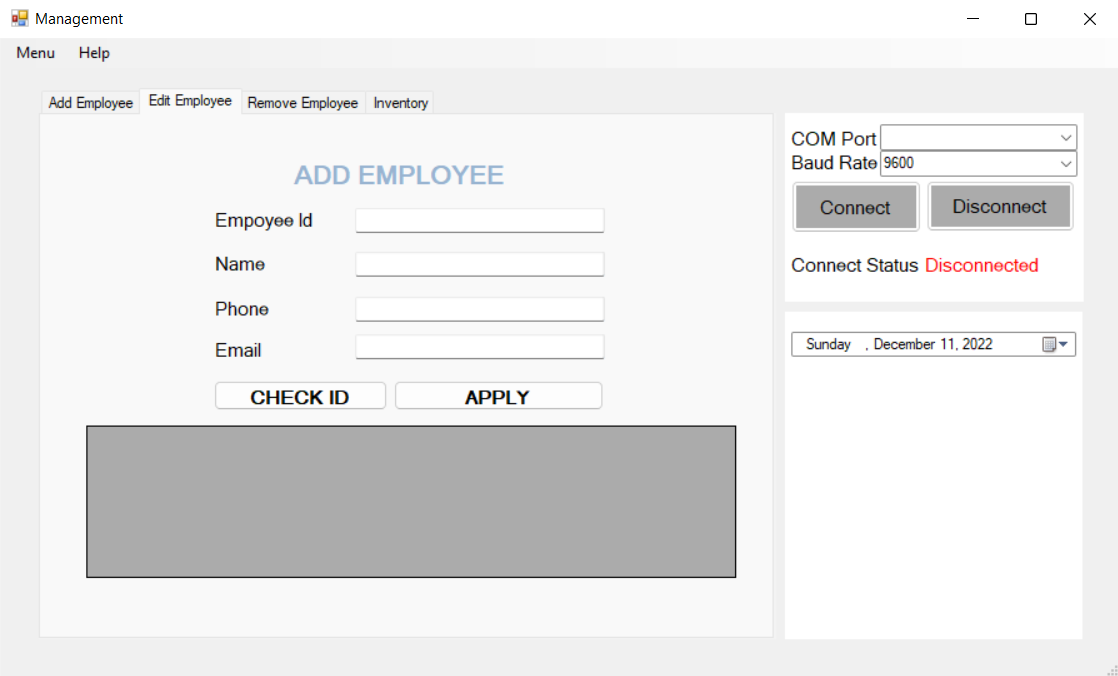
***Hình 3-8.*** *Giao diện quản lý việc chấm công*

***Giao diện thêm nhân viên mới vào công ty***

******

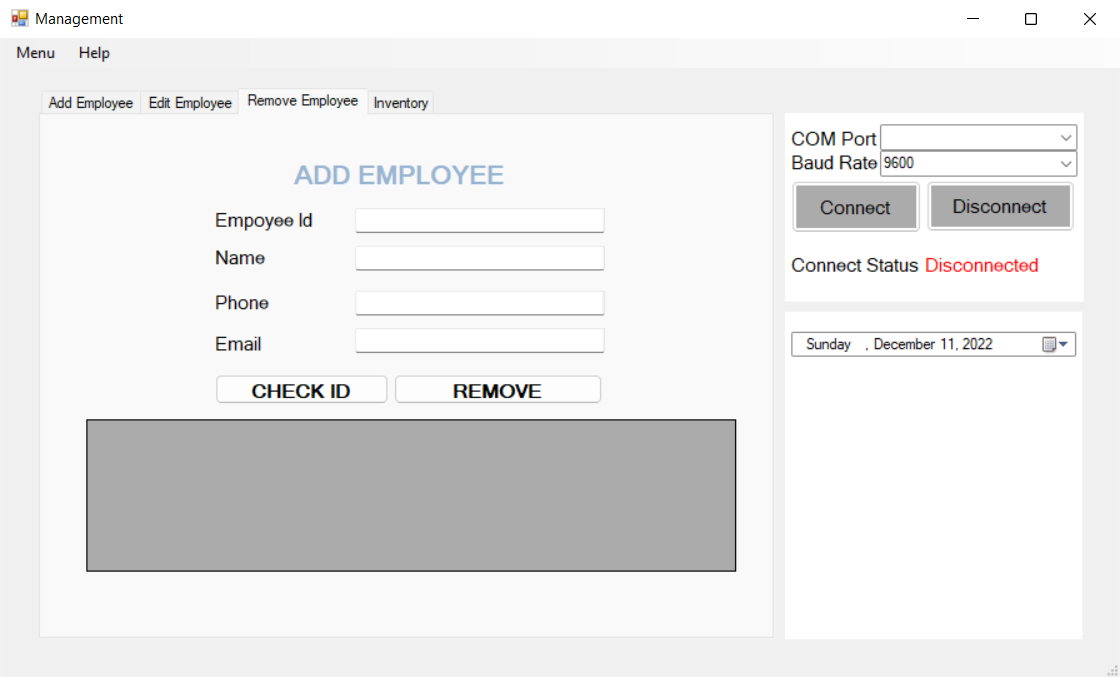
***Hình 3-9.*** *Giao diện thêm nhân viên mới vào công ty*

***Giao diện chỉnh sửa thông tin của nhân viên***

******

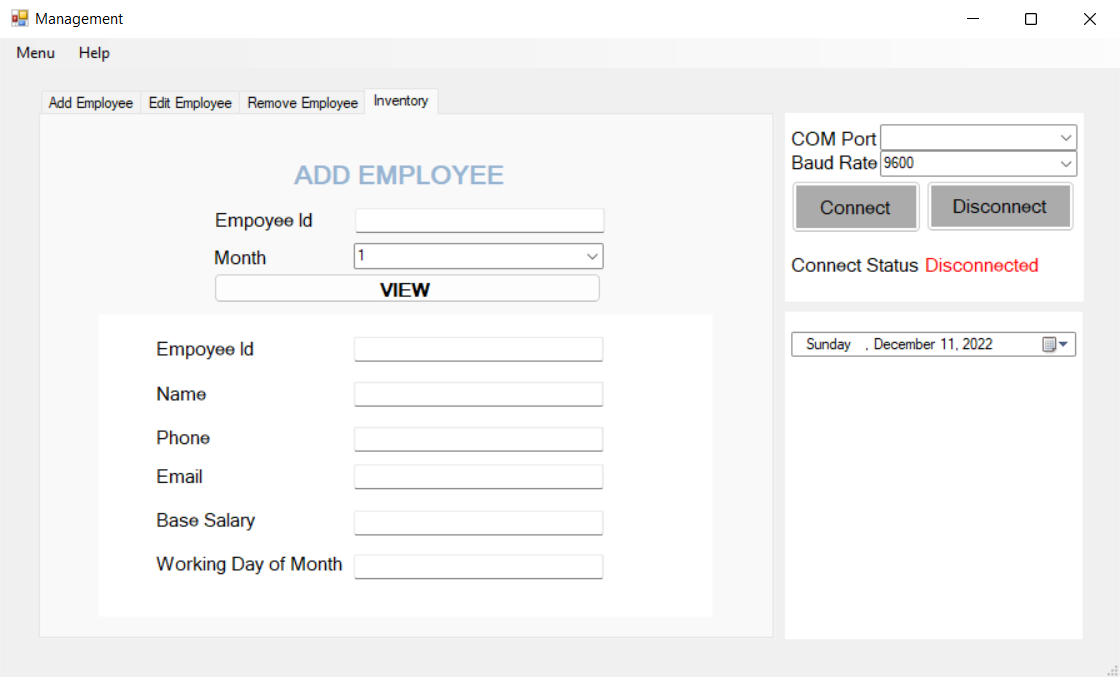
***Hình 3-10.*** *Giao diện chỉnh sửa thông tin của nhân viên*

***Giao diện xóa nhân viên khỏi cơ sở dữ liệu***

******

***Hình 3-11.*** *Giao diện xóa nhân viên khỏi cơ sở dữ liệu*

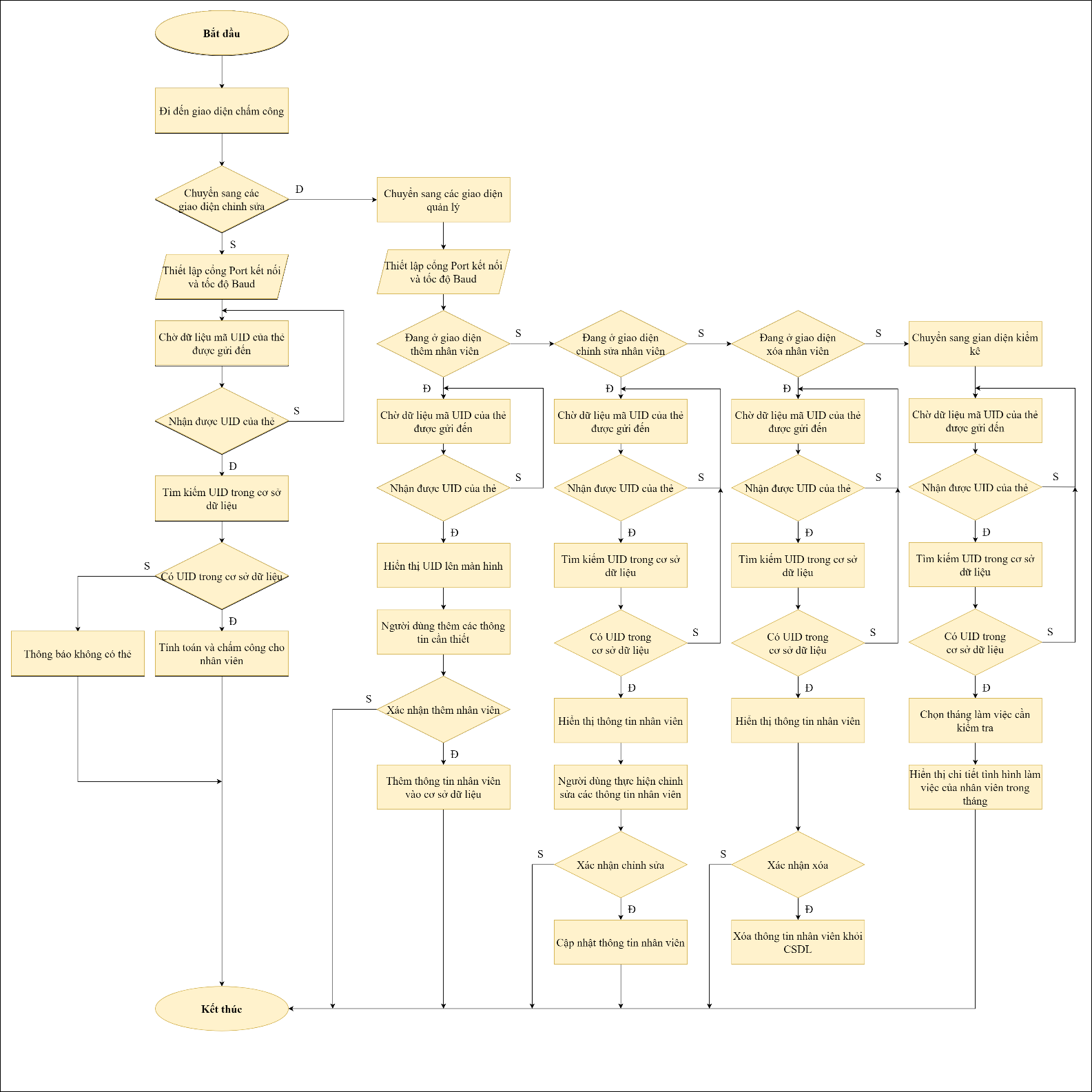
***Giao diện kiểm kê tình hình làm việc của nhân viên***



***Hình 3-12.*** *Giao diện kiểm kê tình hình làm việc của nhân viên*

**3.2.4.2. Backend**

Backend là phần bên dưới, nơi mà mọi thao tác của người dùng với frontend được xử lý, cũng như là những xử lý ngầm mà người dùng không thể nhìn thấy được. Sau những xử lý đó, backend sẽ liên kết giữa cơ sở dữ liệu với giao diện người dùng để hiển thị nó lên frontend cũng như tác động đến cơ sở dữ liệu.



***Hình 3-13.*** *Lưu đồ thuật toán phần mềm quản lý chấm công*

***Giải thích lưu đồ thuật toán phần mềm***

Khi bắt đầu khởi động phần mềm, người dùng sẽ tự động được chuyển đến giao diện quản lý chấm công. Ở giao diện này, mỗi lần nhân viên quét thẻ qua, mã thẻ sẽ được xử lý và liên kết với cơ sở dữ liệu, thực hiện tính toán chấm công cho nhân viên, sau đó hiển thị các thông tin cơ bản của nhân viên lên màn hình.

Nếu người dùng chuyển sang các giao diện quản lý, giao diện thêm nhân viên sẽ hiển thị đầu tiên. Khi quét một thẻ mới qua, UID của thẻ sẽ được hiển thị, sau đó người dùng thêm các trường như tên, số điện thoại, email cho nhân viên mới, sau đó xác nhận thêm, lúc này, thông tin nhân viên sẽ xuất hiện trong cơ sở dữ liệu.

Nếu người dùng chuyển sang giao diện chỉnh sửa thông tin nhân viên, khi này, nếu người dùng quét thẻ qua sẽ thấy UID của thẻ hiện lên, nhấn kiểm tra ID để kiểm tra xem đã có nhân viên này chưa, nếu chưa sẽ chờ lượt quét tiếp theo, nếu có sẽ hiển thị thông tin nhân viên lên màn hình, sau đó người dùng tiến hành kiểm tra thông tin nhân viên và chỉnh sửa nó, sau đó nhấn xác nhận chỉnh sửa, lúc này thông tin của nhân viên sẽ được cập nhật vào cơ sở dữ liệu.

Nếu người dùng chuyển sang giao diện chỉnh xóa nhân viên, khi này, nếu người dùng quét thẻ qua sẽ thấy UID của thẻ hiện lên, nhấn kiểm tra ID để kiểm tra xem đã có nhân viên này chưa, nếu chưa sẽ chờ lượt quét tiếp theo, nếu có sẽ hiển thị thông tin nhân viên lên màn hình, sau đó người dùng tiến hành kiểm tra thông tin nhân viên cần xóa, sau đó nhấn xác xóa, lúc này thông tin của nhân viên đó sẽ bị xóa khỏi cơ sở dữ liệu.

Nếu người dùng chuyển sang giao diện kiểm kê tình hình làm việc trong tháng của nhân viên, khi này, nếu người dùng quét thẻ qua sẽ thấy UID của thẻ hiện lên, người dùng tiếp tục chọn tháng muốn kiểm tra và nhấn xem thông tin nhân viên, lúc này nếu có sẽ hiển thị chi tiết thông tin làm việc trong tháng của nhân viên, nếu không có ID này sẽ quay lại chờ lượt quét thẻ tiếp theo.

***Chương 4.***

# **KẾT QUẢ VÀ ĐÁNH GIÁ**

## **4.1. Kết quả**

Sau một quá trình dài nghiên cứu và thực hiện, về cơ bản đề tài đã có thể đi đến kết quả cuối cùng và đã đáp ứng được những yêu cầu được đặt ra ban đầu. Thiết lập được một hệ thống chấm công hoàn chỉnh, có thể triển khai phát triển ở các doanh nghiệp, công ty ở cấp độ tư nhân vừa và nhỏ.

Kết quả của đề tài có thể được trình bày như sau:

***Về phần cứng:***

- Hệ thống hoạt động bền bỉ, ít lỗi vặt, có thể hoạt động liên tiếp 1 ngày bình thường mà không hề có lỗi và có thể hoạt động ổn định tối đa lên đến 7 ngày trước khi bị quá nhiệt (đã kiểm tra trên thực tế sử dụng)

- Tốc độ quét thẻ của “hộp chấm công” cũng ở tốc độ nhanh, không gây khó chịu cho người dùng khi phải đứng đợi nhận thẻ.

- Hệ thống phần cứng cũng truyền dữ liệu nhanh chóng đến máy tính.

- Thiết kế được một “hộp chấm công” có thể đặt ở trước cửa ra vào mà không gây mất thẩm mỹ về kiến trúc.

***Về phần mềm:***

- Thiết kế được một phần mềm quản lý chấm công ở một công ty có giao diện trực quan, dễ sử dụng, có thông tin chi tiết người phát triển để khi gặp lỗi có thể chủ động liên hệ.

- Thiết kế được một cơ sở dữ liệu trên SQL có dung lượng dường như là vô hạn. Cơ sở dữ liệu chi tiết, phong phú, đa dạng, có thể dễ dàng tương tác thông qua phần mềm.

- Thiết kế được một phần mềm có nền tảng rõ ràng và mạnh mẽ là .NET framework, sử dụng ngôn ngữ lập trình hiện đại nhất thế giới C#, có kiến trúc lập trình rõ ràng ADO.NET, giúp việc bảo quản bảo trì sẽ trở nên dễ dàng hơn và cũng như sẽ dễ dàng phát triển mạnh mẽ nếu có thêm các lập trình viên khác tham gia vì cơ bản hệ thống đã có các thông tin rõ ràng, người phát triển có thể dễ dàng tập trung vào phát triển mà không cần đi lại từ đầu tìm hiểu cách mà đễ tài được tạo nên.

## **4.2. Đánh giá**

### **4.2.1. Các mặt đã đạt được**

***Về sản phẩm***

Phát triển được một hệ thống chấm công như yêu cầu đặt ra từ ban đầu, với những kết quả về phần cứng và phần mềm như đã nêu ở mục 4.1, đề tài cơ bản có thể áp dụng sử dụng trong thực tế.

***Về con người***

Người phát triển đề tài – tôi – Nguyễn Thành Đạt sau quá trình phát triển đề tài, đã tiếp thu thêm những kiến thức mới về công nghệ RFID, củng cổ thêm kiến thức lập trình giao diện trên .NET framework (C#), lập trình Arduino, khả năng sử dụng hệ thống quản lý phiên bản (Git).

Ngoài ra còn học được thêm tính kiên nhẫn, kỹ năng làm việc độc lập trong môi trường đại học.

### **4.2.2. Các mặt hạn chế**

Ngoài các mặt đã đạt được, đề tài còn một số hạn chế như:

+ Cơ sở dữ liệu vẫn còn ở dạng offline, chưa thể đăng tải lên một hệ cơ sở dữ liệu đám mây như MySQL để phục vụ doanh nghiệp lớn, nhiều chi nhánh, đó là lý do hệ thống chỉ có thể được áp dụng ở một số công ty vừa và nhỏ hoặc một chi nhanh công ty đơn lẻ.

+ Hệ thống còn phụ thuộc vào Board Arduino UNO R3 cũng như Module RC522, chưa thể thiết kế một mạch điện riêng chỉ để áp dụng cho hệ thống.

### **4.2.3. Hướng phát triển**

Thông qua các mặt làm được và chưa làm được của đề tài, tôi có thể đưa ra một số hướng phát triển như:

+ Thiết kế một cơ sở dữ liệu trực tuyến hoặc trên các khối dữ liệu đám mây, lúc này hệ thống sẽ có thể hoạt động ở cả các tập đoàn đa quốc gia.

+ Thiết kế một mạch điện dành riêng cho hệ thống, giúp hệ thống tối ưu hơn về kinh tế cũng như là hiệu suất.

# **TÀI LIỆU THAM KHẢO**

**Sách tham khảo:**

[1] Nguyễn Văn Hiệp, Giáo trình “Công nghệ nhận dạng bằng sóng vô tuyến”, NXB ĐHQG Thành phố Hồ Chí Minh, 2014.

[2] Phạm Quang Huy, Lê Cảnh Trung, “Lập trình điều khiển Arduino”, NXB ĐHQG Thành phố Hồ Chí Minh.

**Trang web tham khảo:**

[3] “A tour of the C# language”, Microsoft, truy cập ngày 25/10/2022, đường dẫn:

<https://learn.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/>

[4] “ADO.NET”, Microsoft, truy cập ngày 25/10/2022, đường dẫn:

<https://learn.microsoft.com/vi-vn/dotnet/framework/data/adonet/>

[5] “What is .NET Framework?”, Microsoft, truy cập ngày 25/10/2022, đường dẫn:

<https://dotnet.microsoft.com/en-us/learn/dotnet/what-is-dotnet-framework>

[6] “What is Git”, Alassian, truy cập ngày 01/11/2022, đường dẫn:

<https://www.atlassian.com/git/tutorials/what-is-git>

[7] “UNO R3”, Arduino.cc, truy cập ngày 10/10/2022, đường dẫn:

<https://docs.arduino.cc/hardware/uno-rev3>