**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

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

⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯

**KHOA CƠ KHÍ CHẾ TẠO MÁY**

****

**TÓM TẮT ĐỒ ÁN TỐT NGHIỆP**

***Đề tài***: **Xây dựng hệ thống quản lý sản xuất trong công nghiệp**

Giảng viên hướng dẫn: **Th.S NGUYỄN XUÂN QUANG**

Sinh viên thực hiện:

**LÊ VIẾT HOÀNG** MSSV:**15146166**

**CAO HỮU NHÂN** MSSV:**15146218**

**NGUYỄN TẤN TÀI** MSSV:**15146245**

Lớp: **151461A**

Khoá: **2015 - 2019**

**Tp. Hồ Chí Minh, tháng 07/2019**

MỤC LỤC

[CHƯƠNG 1: DẪN NHẬP. 4](#_Toc13774955)

[1. Lý do chọn đề tài. 4](#_Toc13774956)

[2. Ý nghĩa thực tiễn của đề tài. 5](#_Toc13774957)

[3. Mục tiêu nghiên cứu. 6](#_Toc13774958)

[4. Phương pháp nghiên cứu. 6](#_Toc13774959)

[5. Đối tượng, phạm vi nghiên cứu và giới hạn của đề tài. 7](#_Toc13774960)

[5.1 Đối tượng nghiên cứu. 7](#_Toc13774961)

[5.2 Phạm vi nghiên cứu. 7](#_Toc13774962)

[6. Giới hạn của đề tài. 8](#_Toc13774963)

[CHƯƠNG 2: QUÁ TRÌNH THỰC HIỆN 8](#_Toc13774964)

[PHẦN 1: THIẾT KẾ, LẮP RÁP PHẦN CỨNG, PLC 8](#_Toc13774965)

[1.1 Lấy thông số cường độ dòng điện của máy : 8](#_Toc13774966)

[1.2 Sơ đồ trình tự đọc giá trị cường độ dòng điện từ Module ACS712 và cài đặt thông số Modbus cho ATemega 328 bằng phần mềm Aduino: 10](#_Toc13774967)

[1.3 Chương trình đọc giá trị cân CAS 12](#_Toc13774968)

[1.4 Chương trình PLC đọc giá trị gửi lên từ ATemega328 14](#_Toc13774969)

[1.5 Chương trình C Sharp lấy dữ liệu truyền từ PLC 14](#_Toc13774970)

[PHẦN 2: XÂY DỰNG, THIẾT KẾ CƠ SỞ DỮ LIỆU TRÊN SQL SERVER, PHẦN MỀM SCADA TRÊN MÁY TÍNH 16](#_Toc13774971)

[2.1 Lưu đồ giải thuật cập nhập dữ liệu từ máy chủ lên đám mây. 16](#_Toc13774972)

[2.2 Lưu đồ cho thủ tục tính thời gian bảo hành (SQL). 17](#_Toc13774973)

[2.3 Xây dựng cơ sở dữ liệu. 18](#_Toc13774974)

[PHẦN 3: XÂY DỰNG PHẦN MỀM ĐIỆN THOẠI 23](#_Toc13774975)

[3.1 Viết class Java cho phần kết nối đến Server chứa Cơ sở dữ liệu (ConnectionClass.java). 23](#_Toc13774976)

[3.2 Tạo màn hình đăng nhập. 24](#_Toc13774977)

[3.3 Xây dựng màn hình chính. 25](#_Toc13774978)

[3.4 Xây dựng ListView cho từng bảng. 27](#_Toc13774979)

[3.5 Xây dựng TableItem cho từng bảng (TableItem.java). 28](#_Toc13774980)

[3.6 Xây dựng ListViewAdapter cho từng bảng (ListViewAdapter.java). 28](#_Toc13774981)

[3.7 Viết lệnh xử lý trong chương trình ở màn hình chính (DataActivity.java). 28](#_Toc13774982)

[3.8 Build chương trình và xuất file APK. 29](#_Toc13774983)

[CHƯƠNG 3: KẾT QUẢ VÀ KIẾN NGHỊ 30](#_Toc13774984)

[PHẦN 1: KẾT QUẢ 30](#_Toc13774985)

[1. Kết quả của phần cứng, cụm xử lý, thu thập dữ liệu 30](#_Toc13774986)

[2. Kết quả của phần mềm quản lý trên máy tính 30](#_Toc13774987)

[3. Kết quả của phần mềm giám sát trên điện thoại 34](#_Toc13774988)

[PHẦN 2: KIẾN NGHỊ 34](#_Toc13774989)

# **CHƯƠNG 1: DẪN NHẬP.**

## **Lý do chọn đề tài.**

Khi chọn đề tài, và quan sát sự vận hành của một nhà máy sản xuất lúa gạo, như các hình dưới đây, chúng em nhận thấy:

Hệ thống bồ đài, đặc điểm của hệ thống này là có chiều cao rất lớn, những động cơ thì đặt ở bên trên, người vận hành và giám sát bằng mắt thường rất khó khăn, đây là một dây chuyền hoạt động liên tục, nếu có bất kì một cụm nào ngừng hoạt động mà không được phát hiện và xử lí kịp thời sẽ gây nên sự đình trệ trong sản xuất. Hệ thống lò sấy, hoạt động với công suất vô cùng lớn, nhiệt độ toả ra rất cao. Là một môi trường quá khắt nghiệt cho những người công nhân có thể liên tục ở trong đó để giám sát và vận hành chúng. Cũng như những dây chuyền khác, dây chuyền này cần vận hành liên tục vì nó có sự kết nối với những dây chuyền trước và sau đó, chỉ cần một cụm máy ngừng hoạt động sẽ ảnh hưởng đến toàn bộ dây chuyền sản xuất lớn của nhà máy này.

 *Hình 1: Hệ thống bồ đài*



*Hình 2: Hệ thống lò sấy*

Từ những yêu cầu đặt ra như trên: môi trường hoạt động khắc nghiệt, vị trí khó khăn để giám sát bằng mắt thường (động cơ, băng tải đặt ở trên cao quá tầm mắt người…), dây chuyền lớn sức người khó quản lý chu toàn, yêu cầu hệ thống vận hành liên tục,…nên hệ thống SCADA sẽ là một giải pháp giúp tháo gỡ những vấn đề này. SCADA sẽ tự động giám sát liên tục dây chuyền, đảm bảo những phát sinh lỗi nhỏ nhất đều được phát hiện và thông báo kịp thời để xử lí, hơn nữa SCADA không cần người vận hành phải liên tục ở khu vực sản xuất độc hại, ồn ào mà chỉ cần ở trong phòng điều khiển, mọi thông tin sẽ được hệ thống này thu thập và thông báo liên tục trên giao diện phần mềm.

Từ những hiệu quả thiết thực của hệ thống SCADA, mạng truyền thông công nghiệp, yêu cầu cấp thiết của nhiều doanh nghiệp hiện nay nên chúng em quyết định chọn đề tài:

“**Xây dựng hệ thống quản lý sản xuất trong công nghiệp**” (SCADA)

## **Ý nghĩa thực tiễn của đề tài.**

Nếu tiến hành công việc một cách thủ công, đòi hỏi con người phải kiểm tra thường xuyên và giám sát hoạt động liên tục. Tuy nhiên, việc giám sát và cân chỉnh của con người còn phụ thuộc vào kinh nghiệm, trình độ chuyên môn và độ chính xác chưa cao.

Để giải quyết vấn đề trên, các hệ thống SCADA ra đời, với khả năng thu thập dữ liệu, giám sát quá trình hoạt động của hệ thống, cảnh báo, điều khiển đang được ứng dụng mạnh mẽ trong công nghiệp. Thông qua hệ thống SCADA, có thể lưu trữ dữ liệu của hệ thống trong thời gian dài, có nghĩa quan trọng trong các hệ thống tự động. Do đó, liên kết hệ thống tự động với hệ thống SCADA trung tâm sẽ giúp tối ưu hoá hệ thống tự động đồng thời nâng cao năng suất, chất lượng của các sản phẩm cũng như đảm bảo an toàn, độ bền cho toàn bộ hệ thống.

1. **Mục tiêu nghiên cứu.**

Sau khi thực hiện xong đề tài này:

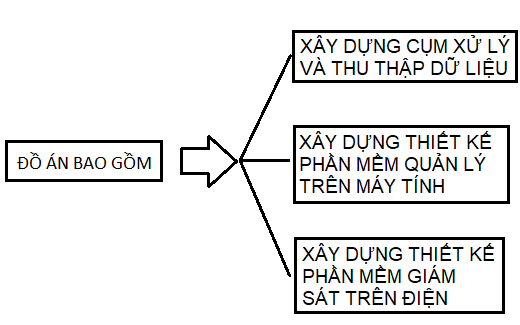
* Lắp ráp, kết nối được phần cứng PLC S7-1200, Arduino,…
* Sử dụng board truyền thông CB 1241 RS485
* Kết nối PLC theo các chuẩn RS485
* Giao tiếp PLC với máy tính qua Modbus TCP-IP
* Thiết kế hệ thống SCADA trên phần mềm Visual Studio
* Xây dựng và xử lý cơ sở dữ liệu trên phần mềm Microsoft SQL Server
* Thiết kế phần mềm trên điện thoại bằng Android Studio

1. **Phương pháp nghiên cứu.**

Phương pháp nghiên cứu tài liệu: đây là phương pháp nghiên cứu ban đầu khi bắt đầu nghiên cứu về vấn đề mới. Nhóm đã tham khảo tài liệu hãng, các thông tin trên sách vở ở thư viện, trên internet,…

Phương pháp nghiên cứu thực nghiệm, phương pháp này được nhóm nghiên cứu và sử dụng trong suốt quá trình thực hiện đề tài. Từ nghiên cứu tài liệu đến việc thực hiện, nhóm tiến hành xây dựng thiết kế mạch, lắp ráp phần cứng, lập trình, xây dựng thiết kế giao diện và chạy thử chương trình trên mô hình thực tế.

Đồ án bao gồm:



1. **Đối tượng, phạm vi nghiên cứu và giới hạn của đề tài.**
   1. **Đối tượng nghiên cứu.**

Đề tài tập trung nghiên cứu các đối tượng sau:

* PLC Siemens S7-1200
* Board truyền thông CB 1241 cho PLC S7-1200
* Mạng truyền thông công nghiệp
* Các thiết bị công nghiệp: cân, biến tần, biến dòng,...
* Truyền thông Arduino với PLC
* Phần mềm Android trên Android Studio
* Cơ sở dữ liệu trên SQL Server
  1. **Phạm vi nghiên cứu.**
* Xử lý tín hiệu analog trong Arduino và digital trong PLC
* Mạng truyền thông Modbus
* Giao tiếp với các thiết bị công nghiệp
* Kết nối PLC S7-1200 với máy tính
* Thiết kế giao diện giám sát SCADA
* Thiết kế phần mềm giám sát trên điện thoại Android

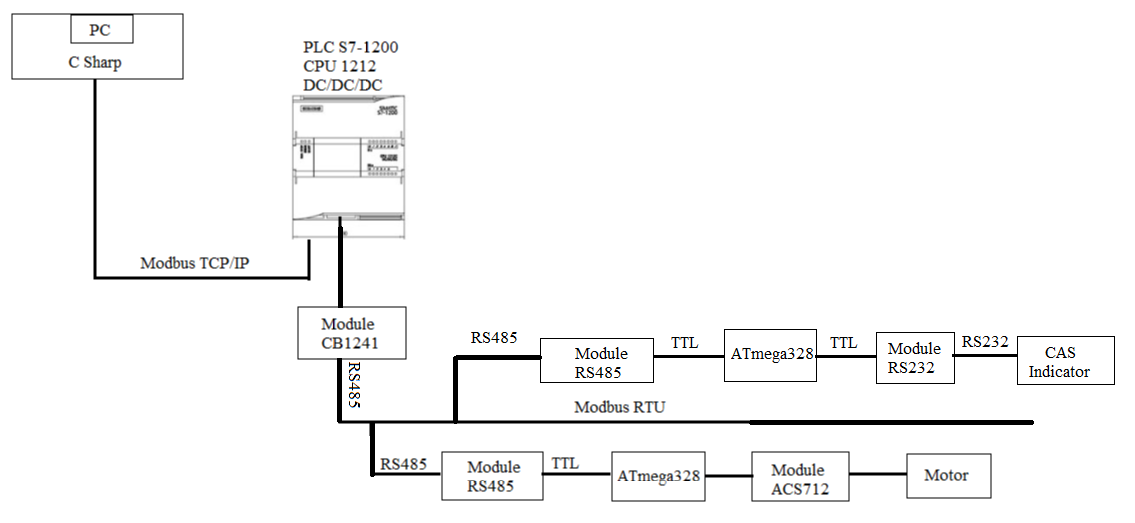
1. **Giới hạn của đề tài.**

Do thời gian và khả năng có hạn nên nhóm chúng em chỉ có thể thực hiện trong các vấn đề sau:

* Giao tiếp PLC S7-1200 với máy tính
* Master nhận dữ liệu từ các Slave gửi lên và điều khiển các Slave qua mạng Modbus. Nhận dữ liệu RS485 thông qua module CB 1241
* Điều khiển Start/Stop biến tần bằng manual
* Thiết kế giao diện phần mềm SCADA trên máy tính
* Thiết kế phần mềm giám sát cho điện thoại Android

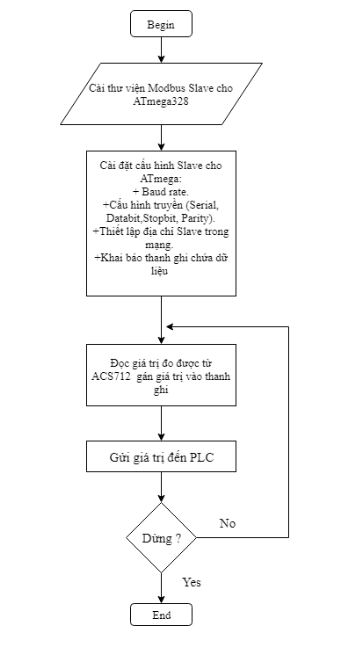
# **CHƯƠNG 2: QUÁ TRÌNH THỰC HIỆN**

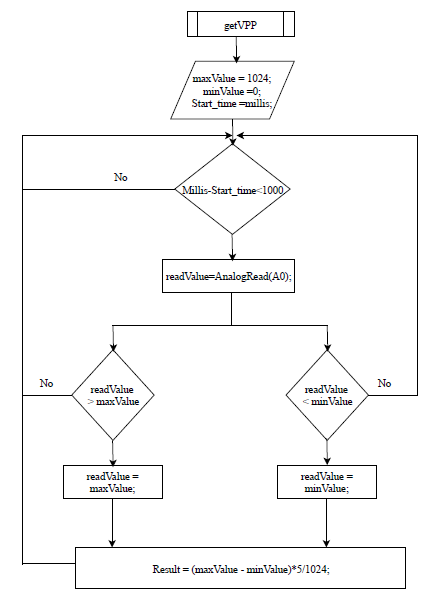
**PHẦN 1: THIẾT KẾ, LẮP RÁP PHẦN CỨNG, PLC**

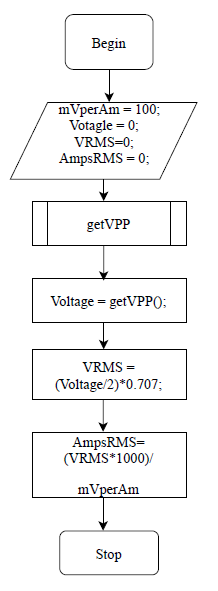
* 1. **Lấy thông số cường độ dòng điện của máy :**
* Sơ đồ khối của mô hình:
* Trình tự lấy cường độ dòng điện của máy:

Đầu tiên Module ACS712 đo dòng điện cấp cho biến tần và động cơ hoạt động (Dòng điện thay đổi khi tải trên động cơ thay đổi, động cơ ở đây tượng trưng cho hoạt động của một máy).Module ACS712 xuất ra giá trị analog đến Arduino, Vout biến đổi tuyến tính theo sự thay đổi của dòng điện Ip được lấy mẫu thứ cấp DC (hoặc AC), trong phạm vi đã cho. ATMega 328 truyền tín hiệu ra dạng chuẩn TTL, vào Module chuyển RS485 sang TTL . Nhờ Module chuyển RS485 sang TTL biến đổi chuẩn giao tiếp TTL sang RS485 nên tín hiệu tiếp tục được truyền vào broad truyền thông CB1421 gắn trên PLC. Tại đây, PLC xử lý dữ liệu và đưa lên phần mềm quản lý cơ sở dữ liệu SQL Server trên máy tính và cho phép người dùng quan sát và theo dõi.

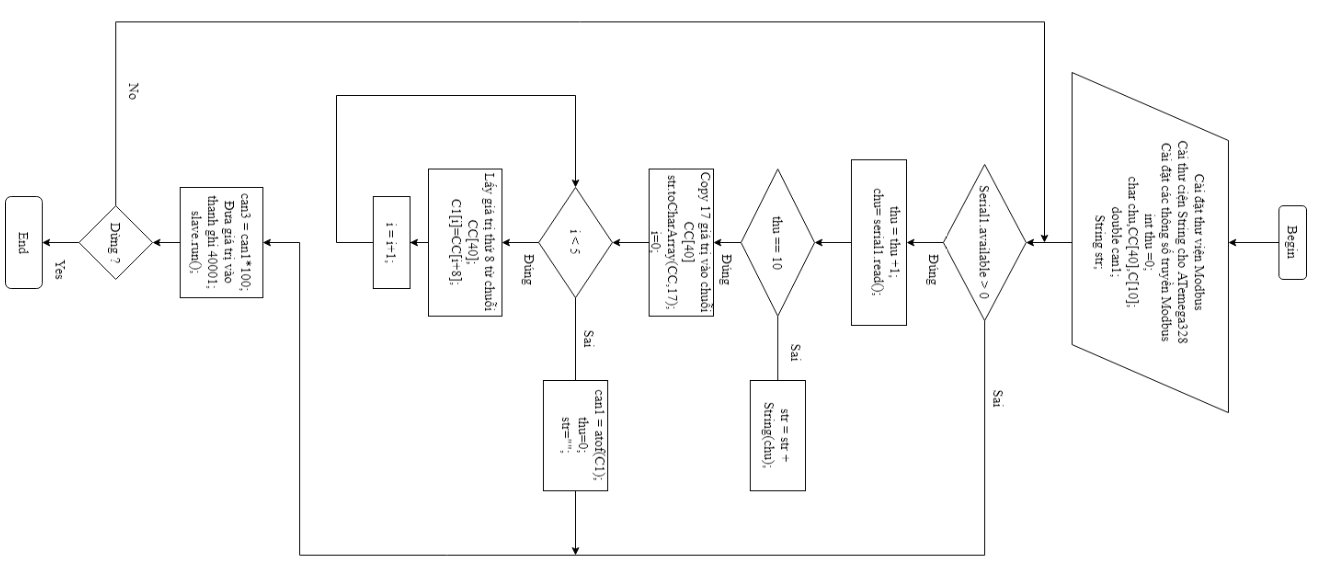
**1.2 Sơ đồ trình tự đọc giá trị cường độ dòng điện từ Module ACS712 và cài đặt thông số Modbus cho ATemega 328 bằng phần mềm Aduino:**



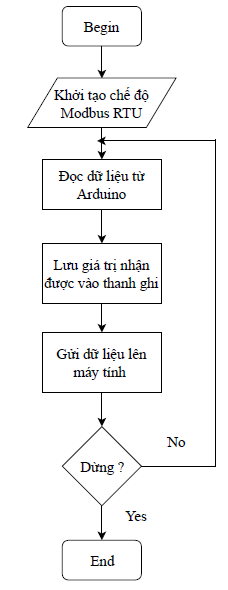
* Lưu đồ giải thuật đọc giá trị Ampare



* 1. **Chương trình đọc giá trị cân CAS**
* Lưu đồ giải thuật

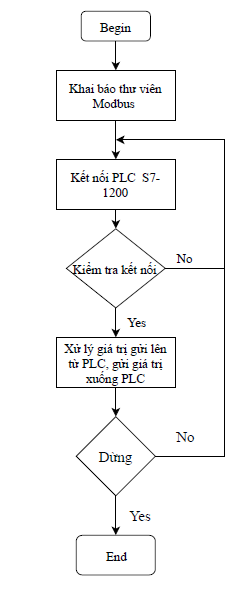


* 1. **Chương trình PLC đọc giá trị gửi lên từ ATemega328**



**1.5 Chương trình C Sharp lấy dữ liệu truyền từ PLC**

* Sơ đồ đọc giá trị từ PLC



\*\* Hàm đọc giá trị từ thanh ghi gửi lên từ PLC:

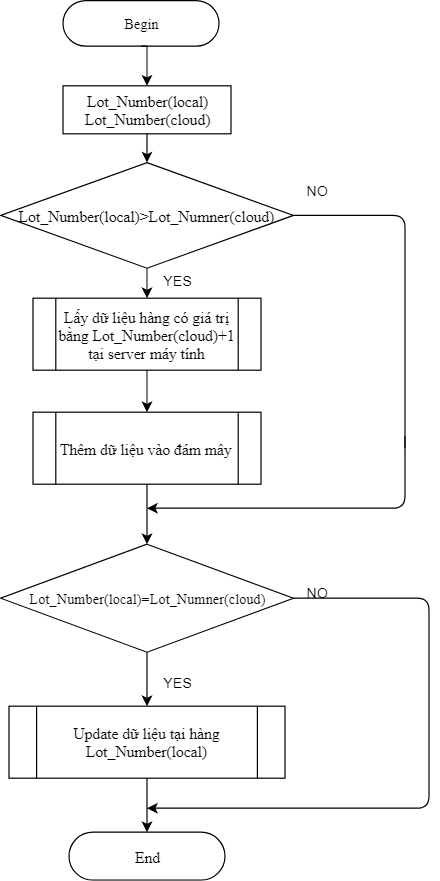


\*\* Hàm viết giá trị gửi trả về PLC

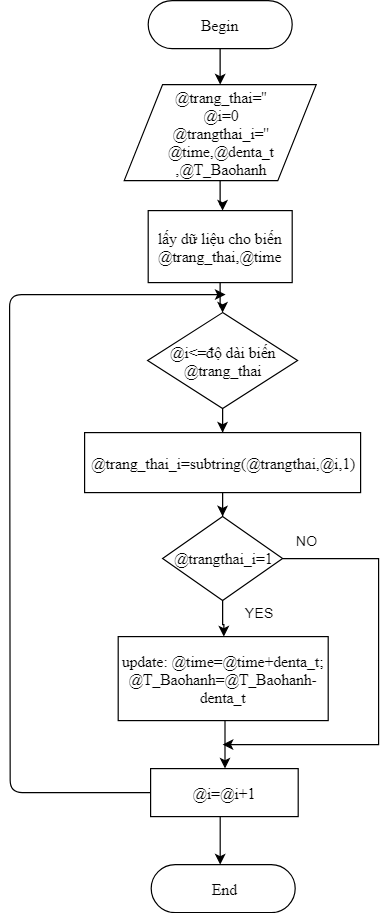


## **PHẦN 2: XÂY DỰNG, THIẾT KẾ CƠ SỞ DỮ LIỆU TRÊN SQL SERVER, PHẦN MỀM SCADA TRÊN MÁY TÍNH**

### **2.1 Lưu đồ giải thuật cập nhập dữ liệu từ máy chủ lên đám mây.**



* 1. **Lưu đồ cho thủ tục tính thời gian bảo hành (SQL).**

****

* 1. **Xây dựng cơ sở dữ liệu.**

**Bảng User:** bảng chứa tên người dùng user và password

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| **1** | id\_user | Int | Khóa chính, mã tài khoản |
| **2** | name\_user | String | Tên người dùng |
| **3** | user\_name | String | Tên tài khoản |
| **4** | pass | String | Mật khẩu người dùng |

**Bảng tbl\_permision\_detail:** bảng sẽ chứa những quyền hạn cụ thể dành cho nhóm quyền hạn

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| **1** | id\_pd | Int | Khóa chính, mã tên quyền hạn |
| **2** | name\_action | String | Tên quyền hạn |
| **3** | code\_action | String | Mã quyền hạn |
| **4** | id\_per | Int | Mã tài khoản |

**Bảng tbl\_permision:** bảng chứa nhóm quyền hạn

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| **1** | id\_per | Int | Khóa chính, mã nhóm quyền hạn |
| **2** | name\_per | String | nhóm quyền hạn |
| **3** | description | String | Mô tả nhóm quyền hạn |

**Bảng tbl\_per\_relationship:** là bảng lưu mối liên hệ giữa người dùng và nhóm quyền hạn. Mục đích của bảng này không phải là để một người dùng có nhiều nhóm quyền mà để không phải truy vấn lại bảng user chứa thông tin nhạy cảm như username và password.

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trương** | **Kiểu dữ liệu** | **Mô tả** |
| **1** | id\_rel | Int | Khóa chính |
| **2** | id\_user\_rel | Int | Mã tài khoản |
| **3** | id\_per\_rel | Int | Mã nhóm người |
| **4** | suspended | Bit | acctive |

**Bảng Line: chứa thông tin của một dây chuyền**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| **1** | Id | Int | Mã lô |
| **2** | LotNumber | Int | Số lô |
| **3** | StartTime | datetime | Thời gian bắt đầu |
| **4** | EndTime | datetime | Thời gian kết thúc |
| **5** | Name\_Material | Nvarchar | Tên nguyên liệu |
| **7** | Name\_Product\_1 | Nvarchar |  |
| **8** | Name\_Product\_2 | Nvarchar |  |
| **9** | Name\_Product\_3 | Nvarchar | Tên sản phẩm |
| **10** | Name\_Product\_4 | Nvarchar |  |
| **11** | Name\_Product\_5 | Nvarchar |  |
| **12** | Batching\_Material | Float | Số mẻ lô nguyên liệu |
| **13** | Batching\_Product\_1 | Float |  |
| **14** | Batching\_Product\_2 | Float | Số mẻ cân sản phẩm |
| **15** | Batching\_Product\_3 | Float |  |
| **16** | Batching\_Product\_4 | Float |  |
| **17** | Batching\_Product\_5 | Float |  |
| **18** | V\_Material | Float | Cân nguyên liệu |
| **19** | V\_Product\_1 | Float |  |
| **20** | V\_Product\_2 | Float |  |
| **21** | V\_Product\_3 | Float | Cân khối lượng sản phẩm |
| **22** | V\_Product\_4 | Float |  |
| **23** | V\_Product\_5 | Float |  |
| **24** | SumTimeRunLot | Float | Tổng thời gian chạy |
| **25** | Energy | Float | Công suất |
| **26** | Note | nvarchar | Ghi chú |

**Bảng Session:** chứa thông tin của chi tiết do người giám sát nhập vào thuận tiện cho viêc báo cáo.

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu đữ liệu** | **Mô tả** |
| **1** | Id | Int | Mã lô |
| **2** | LotNumber | Int | Số lô |
| **3** | StartTime | Datetime | Thời gian bắt đầu |
| **4** | EndTime | Datetime | Thời gian kết thúc |
| **5** | Line\_1 | Int |  |
| **6** | Line\_2 | Int |  |
| **7** | Line\_3 | Int | Line chạy |
| **8** | Line\_4 | Int |  |
| **9** | Line\_5 | Int |  |
| **10** | SumTimeRunLot | Float | Thời gian chạy |
| **11** | Bui | Float |  |
| **12** | Buibui | Float |  |
| **13** | San | Float |  |
| **14** | Buibuikho | Float | Các loại sản phẩm |
| **15** | Buibuiuot | Float |  |
| **16** | Tạpchat | Flaot |  |
| **17** | Gaohong | Flaot |  |
| **18** | Energy | Float | Công suất |
| **19** | Note | Nvarchar | Ghi chú |

**Bảng machine:** chứa thông tin trạng thái máy cho điện thoại

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| **1** | Id | Int | Khóa chính |
| **2** | MAY | Nvarchar(50) | Tên máy |
| **3** | Trang\_thai | Nvarchar(50) | Trạng thái máy |
| **4** | Dong\_dien | Float | Đòng điện |
| **5** | Thoi\_gian\_chay | Float | Thời gian chạy |
| **6** | Thoi\_gian\_BH | Glaot | Bảo hành |

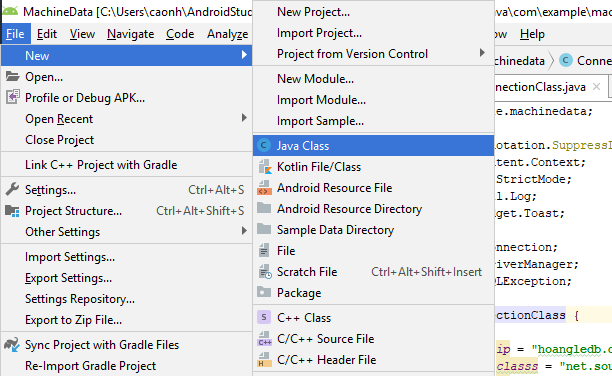
## **PHẦN 3: XÂY DỰNG PHẦN MỀM ĐIỆN THOẠI**

### **3.1 Viết class Java cho phần kết nối đến Server chứa Cơ sở dữ liệu (ConnectionClass.java).**

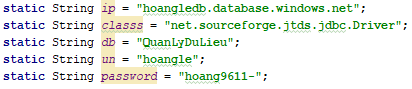
Android app không thể giao tiếp một cách trực tiếp với SQL Server, bởi đơn giản là do SQL Server thuộc về Microsoft còn Android là của Google chúng. Nhưng bên cạnh đó, chúng ta có thể viết một ứng dụng bằng Java, mà Java có lại hỗ trợ kỹ thuật cho phép người dùng có thể sử dụng ứng dụng Android để thao tác với cơ sở dữ liệu trên SQL Server.

**Tạo một file java:**

Chọn File => New => Java Class

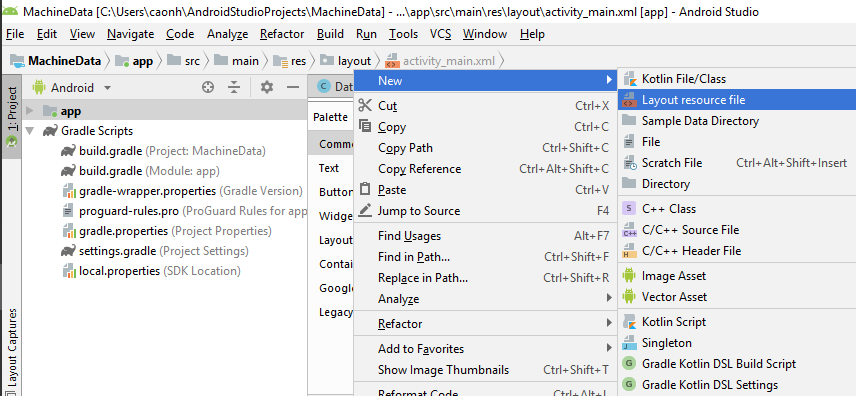
****

**Với các thông số cần lưu ý:**

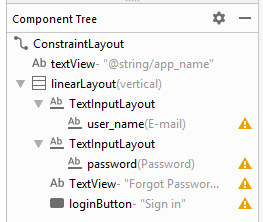
* ip (địa chỉ ip của server lưu trữ cơ sở dữ liệu, ở đồ án này dùng server Azure của Microsoft cung cấp)
* classs
* db (tên database)
* un (tên đăng nhập vào server)
* password (mật khẩu đăng nhập vào server**)**
  1. **Tạo màn hình đăng nhập.**
     1. **Thiết kế giao diện màn hình đăng nhập (activity\_main.xml).**

Tạo một file Layout bằng cách:

* + Dẫn địa chỉ đến: MachineData/app/src/main/res/layout
  + Chuột phải trên thư mục layout và chọn:
  + New => Layout resource file

****

**Thiết kế màn hình đăng nhập với các thành phần dưới đây:**

****

1 TextView, 2 khung TextInputLayout dùng cho username và password, 1 button để đăng nhập.

* + 1. **Viết code để “Sign in” vào màn hình chính ở file MainActivity.**

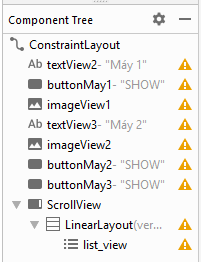
Màn hình đăng nhập gồm ba đối tượng chính: User, Password và Button “Sign in”. Khi đăng nhập đúng User và Password thì màn hình sẽ chuyển sang màn hình giao diện chính (Lúc đó DataActivity.class sẽ hoạt động)

### **3.3 Xây dựng màn hình chính.**

Tương tự như thiết kế màn hình đăng nhập tạo một file trong thư mục layout để thiết kế màn hình chính cho phần mềm.

Dùng ListView kết hợp ScrollView để tạo chỗ cho hiện thị dữ liệu sau này.

Các thành phần trong màn hình chính như sau:

****

Ngoài những chú ý những thuộc tính của Button, TextBox,… ta cần chú ý để các thuộc tính của ListView, ScrollView, chi tiết là:

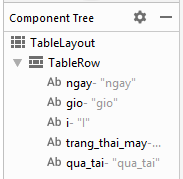
****

Thuộc tính cần chú ý nhất ở đây là layout\_height và layout\_width – cần được thiết lập đúng để có thể hiện thị đầy đủ cơ sở dữ liệu được lấy từ SQL Server.

* 1. **Xây dựng ListView cho từng bảng.**

Tạo một file layout, và đặt tên listview\_row\_may\_1.xml

Dùng các công cụ để thiết kế list view này, các thành phần của list view cho máy 1 cơ bản gồm:

****

ListView này chứa tên những cột sẽ được hiển thị trên màn hình chính, những tên biến khai báo được đặt trùng tên biến trên cơ sở dữ liệu.

### **3.5 Xây dựng TableItem cho từng bảng (TableItem.java).**

Tạo một file java, trong thư mục machinedata, đặt tên TableItem.java

Ở phần này, ta khai báo kiểu dữ liệu của các cột trong bảng, kiểu dữ liệu của từng cột. Tạo câu lệnh lấy dữ liệu từng cột cho chương trình.

### **Xây dựng ListViewAdapter cho từng bảng (ListViewAdapter.java).**

Tạo một file java, trong thư mục machinedata, đặt tên ListViewAdapter.java

Phần này hỗ trợ hiển thị những giá trị ta lấy được từ TableItem lên ListView.

### **3.7 Viết lệnh xử lý trong chương trình ở màn hình chính (DataActivity.java).**

Lệnh xử lí viết trên event nhấn nút ứng với từng bảng thích hợp. Tại đây ta cũng xử lý các biến của TableItem thành những nhãn tiếng Việt cho phù hợp. Đồng thời gọi các lệnh từ TableItem và ListViewAdapter để đạt được kết quả cuối cùng – hiển thị nội dung lên màn hình. Xử lý nhãn của nút nhấn – khi chưa hiện bảng, nhãn nút sẽ là chữ “SHOW”, sau khi nhấn nút và bảng được hiện, nhãn nút sẽ là chữ “HIDE”.

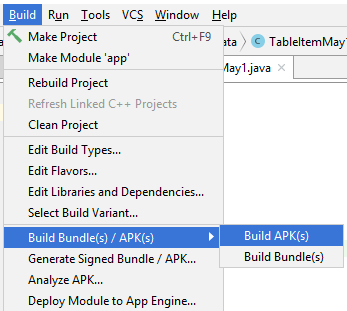
* 1. **Build chương trình và xuất file APK.**
* Để build chương trình ta nhấn Ctrl+F9
* Tiến hành chạy thử trên máy ảo được cài đặt sẵn trên máy tính:

+ Nhấn tổ hợp Shift + F10

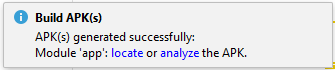
+ Cửa sổ Select Deployment Target xuất hiện, tại đây ta có thể chọn máy ảo đã được tạo sẵn (nếu có) hoặc tiến hành tạo một máy ảo mới bằng cách chọn Create New Virtual Device. Lưu ý cần chọn hoặc tạo máy ảo với phiên bản hệ điều hành Android phù hợp với phần mềm, ở đây là Android 8.0

* Xuất file APK để cài đặt vào điện thoại

+ Chọn Build => Build Bundle(s)/APK(s) => Build APK(s)

****

Chờ cho chương trình chạy đến khi có thông báo này ở góc dưới bên phải màn hình, có nghĩa là file APK đã được tạo thành công:

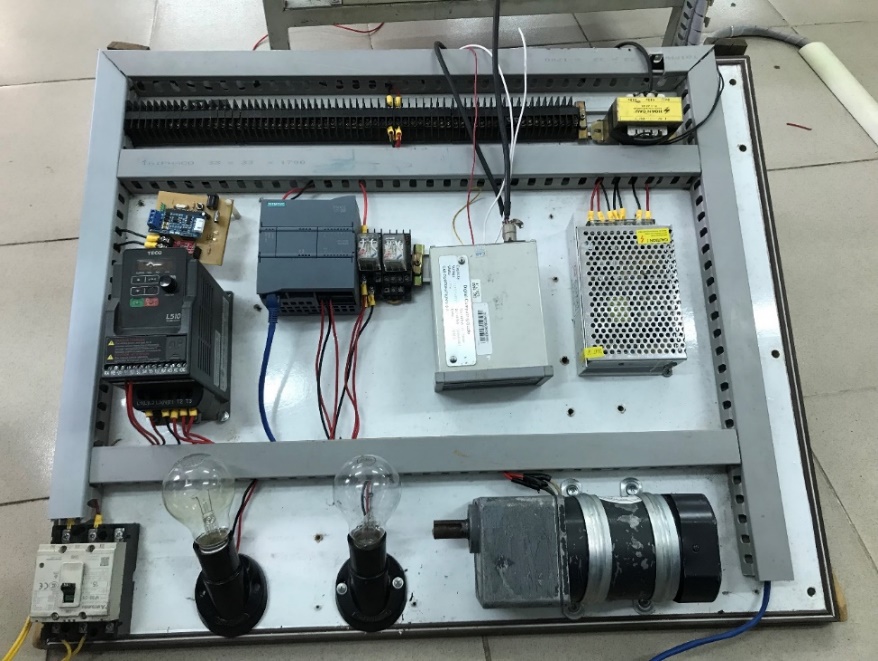


Chọn “locate” chương trình sẽ mở cửa sổ thư mục chứa file APK vừa được tạo với tên “app-debug.apk”

# **CHƯƠNG 3: KẾT QUẢ VÀ KIẾN NGHỊ**

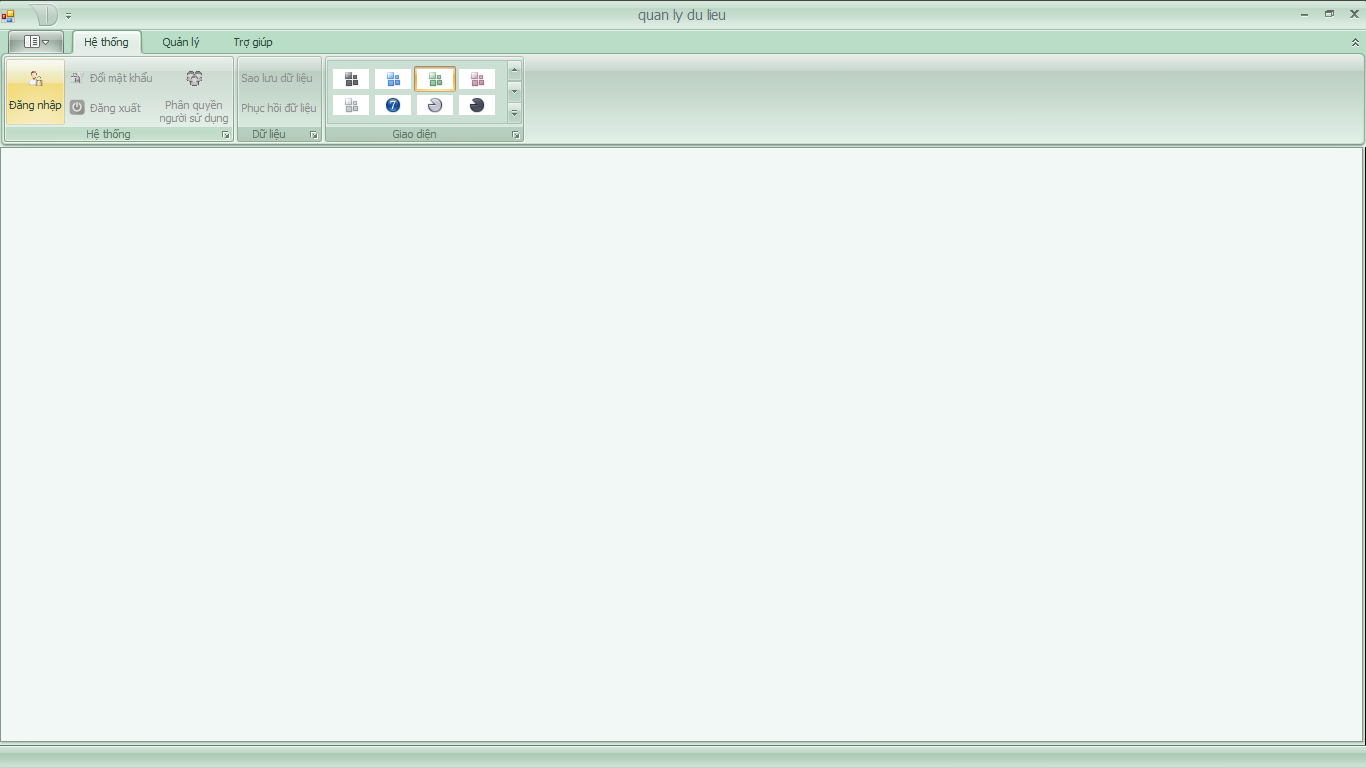
## **PHẦN 1: KẾT QUẢ**

### **Kết quả của phần cứng, cụm xử lý, thu thập dữ liệu**

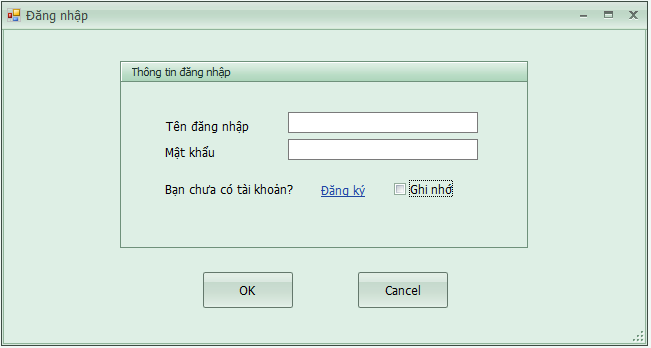
****

1. **Kết quả của phần mềm quản lý trên máy tính**

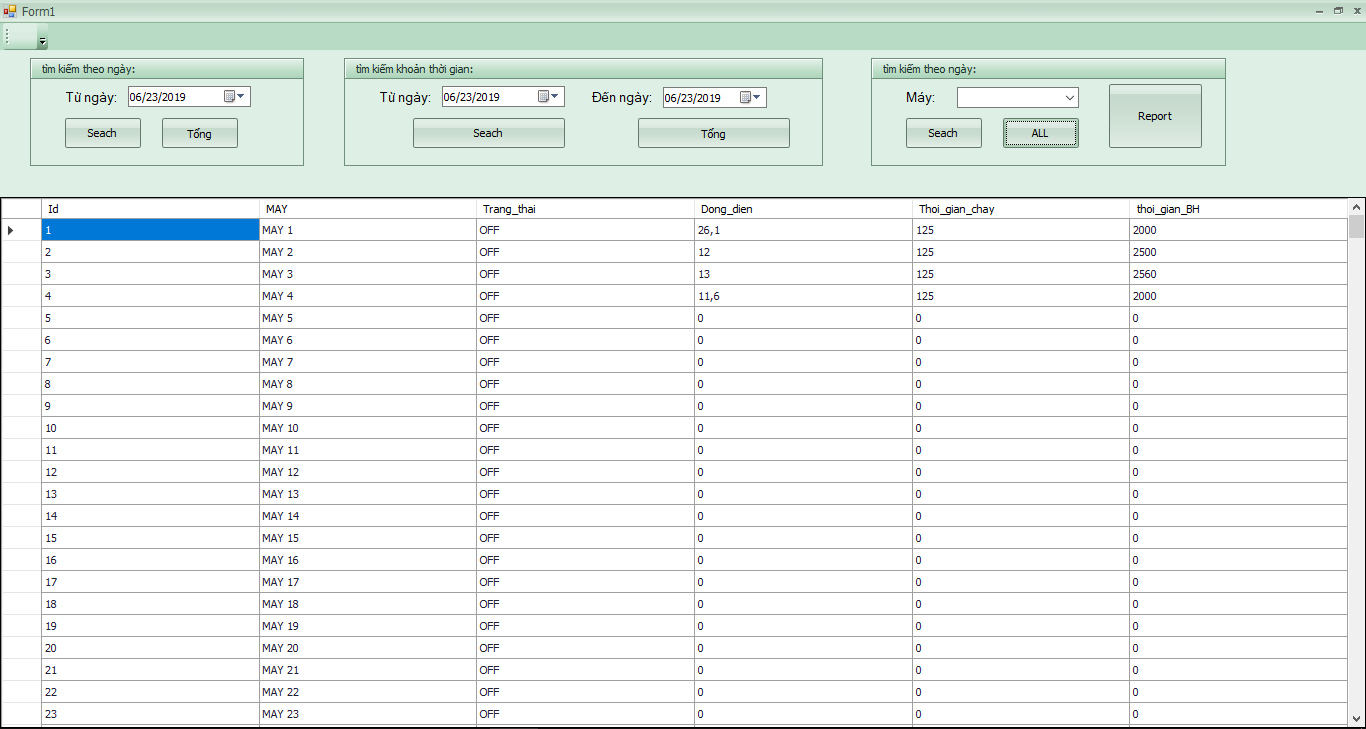
* Giao diện chính

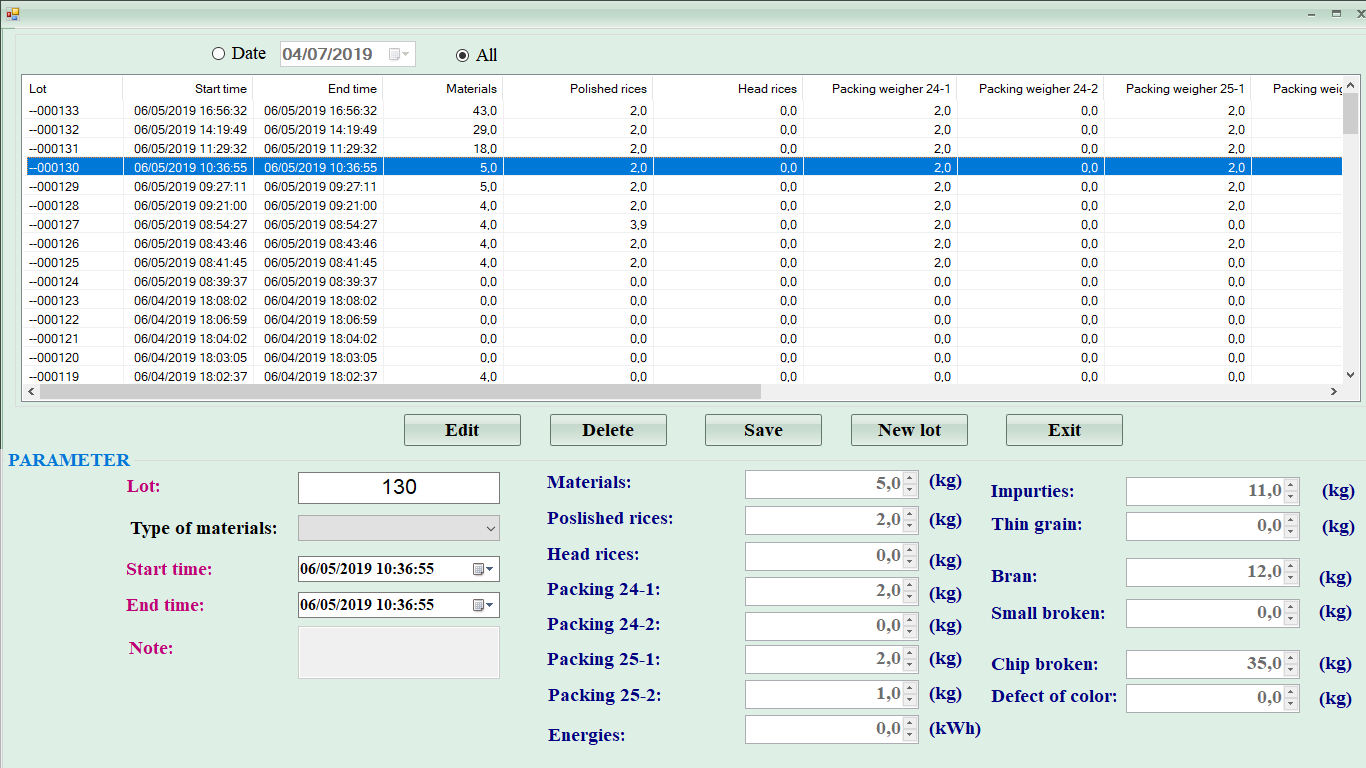


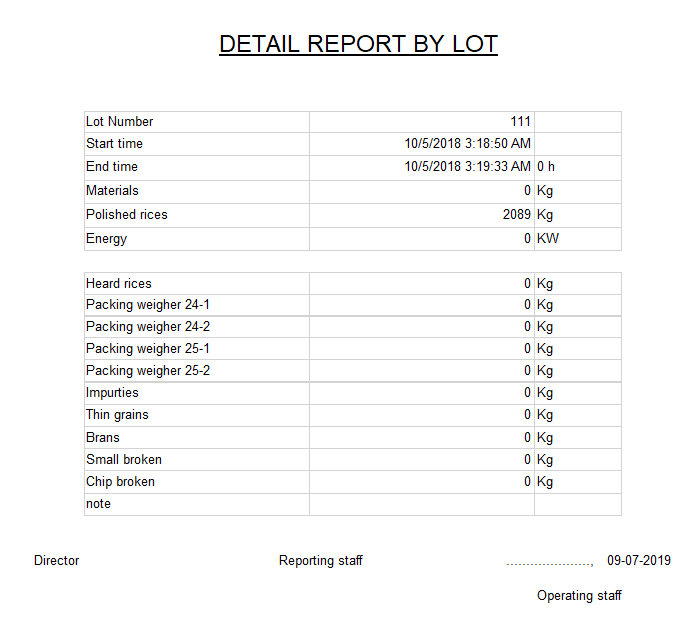
* Giao diện đăng nhập



* Giao diện data



* Giao diện form nhập data lên cơ sở dữ liệu dám mây:
* Giao diện form Report



**Kết quả tóm tắt:**

* Cho phép người dùng có tài khoảng có thể đăng nhập vào hệ thống và sử dụng hệ thống với một số chức năng: điều khiển và giám sát quản lý dữ liệu tạo report
* Ứng dụng C# và các công cụ Azure để lập trình.
* Áp dụng được phương pháp và công nghệ hiện đại để ứng dụng vào thực tế.
* Tuy nhiên việc lấy dữ liệu được đặt tại server, đôi khi đường truyền internet không ổn định.
* Phần mềm SCADA này có những phát triển hơn những phần mềm trên thị trường: có hình ảnh trực quang sinh động, hiển thị rõ sự hoạt động của từng cụm máy, có khả năng xuất báo cáo ra file word hoặc excell, pdf,…

### https://scontent.fsgn8-1.fna.fbcdn.net/v/t1.15752-9/65889514_1470795196395495_1539044330147151872_n.jpg?_nc_cat=110&_nc_oc=AQlMaOGDwOcZXSFJeERGijpKDP2cmbQrMsklQRF2eVSI2DhJ4sr0UqsCbv6k6ll8feQ&_nc_ht=scontent.fsgn8-1.fna&oh=3b1813ced4bb07253b7c75aac2c41b41&oe=5DAE768A**Kết quả của phần mềm giám sát trên điện thoại**

## **PHẦN 2: KIẾN NGHỊ**

Phần mềm điện thoại còn đơn giản do giới hạn về phần cứng và kiến thức chuyên ngành. Nên có thêm hình ảnh động sẽ giúp giao diện trực quan và sinh động hơn.

Cụm xử lý và thu thập dữ liệu: Khi sử dụng càng nhiều nút Slave thì độ trễ giữa các nút càng cao, dễ gây ra mất tín hiệu. Kết nối giữa các thiết bị phải chắc chắn (nên hàn các chân thiết bị với nhau) để đảm bào các thiết bị luôn được kết nối với nhau thông suốt. Trong quá trình vận hành Arduino bị Disconnect với hệ thống cần phải Reset Module vì vậy nên cải tiến lại thư viện ModbusSlave .

Phần mềm quản lý trên máy tính: Phần giám sát trạng thái máy phụ thuộc vào máy chủ nếu máy chủ tắt thì dữ liệu sẽ không được gửi lên mặc dù dây chuyền còn đang hoặc động, đễ bị ảnh hưởng bởi sự mất kết nối của mạng Internet. Khắc phục: sử dụng PLC hoặc vi điều khiển đưa trạng thái máy lên Enternet sẽ khắc phục được điểm thiếu sót này.