INDUSTRIAL WORK PRACTICE REPORT

CREATE A SCHEDULING SYSTEM USING VBA IN MICROSOFT EXCEL AT PT. PANASONIC INDUSTRIAL DEVICE BATAM

This report written to show the author responsibility while doing the Field Work Practice at **PT. Panasonic Industrial Device Batam** On January 18th until May, 2021

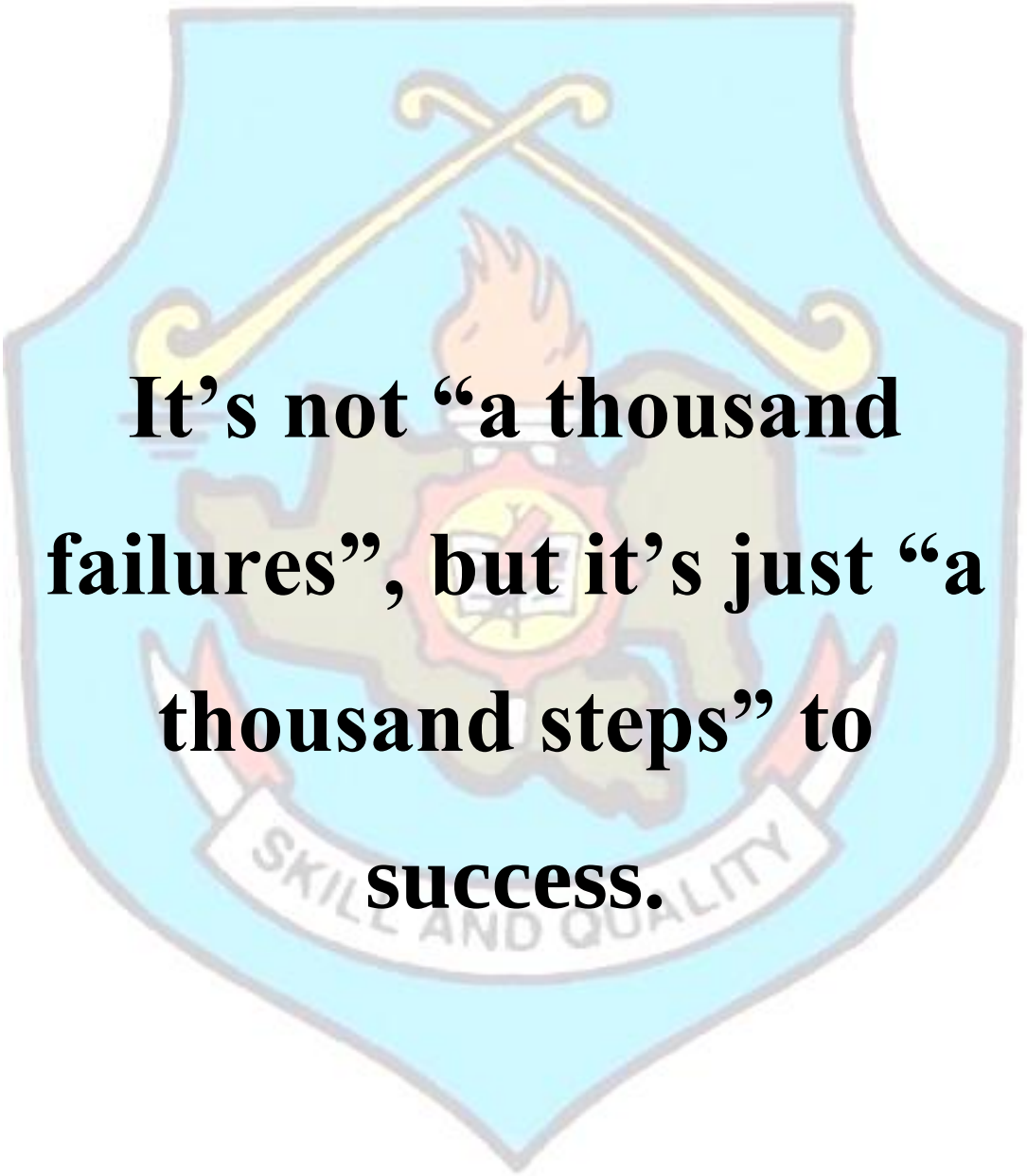
Arranged by **:**

Name	: Peter Prigel Wicaksono
NIS	: 19-9069
Skill Competency	: Computer & Network Engineering

EDUCATION PROVINCE OF RIAU ISLANDS VOCATIONAL HIGH SCHOOL 1 BATAM Jl. Prof. Dr. Hamka No. 1 Batu Aji Batam, Tlp. 0778 365093 Email. humas@smkn1batam.sch.id

2021

MOTTO



It’s not “a thousand failures”, but it’s just “a thousand steps” to success.

COMPANY APPROVAL PAGE

CREATE A SCHEDULING SYSTEM USING VBA IN MICROSOFT EXCEL AT PT. PANASONIC INDUSTRIAL DEVICE BATAM

This report has been validated and approved

On May 17th 2021 in Batam by:

Industrial Coordinator,	Mentor,
Anwar Subekti	**Purwanto**
HOS Employee Development	HOS Resistor Department

Head of the Personally Department
PT. Panasonic Industrial Devices Batam

Budisila Hutasuhut
HR Manager

SCHOOL APPROVAL PAGE

CREATE A SCHEDULING SYSTEM USING VBA IN MICROSOFT EXCEL AT PT. PANASONIC INDUSTRIAL DEVICE BATAM

This report has been validated and approved

On May 17th 2021 in Batam by:

Industrial Relations Coordinator,	Report Counselor,
Yarliansyah Agustian, S.ST	**Mike Musna, Gr, S.Pd**
NIP. 19860802 200902 002	NIP.-

Principal of Vocational High School 1 Batam,

Lea Lindrawijaya Suroso, M.Pd
NIP. 19690124 199503 2 005

PREFACE

I thank God Almighty, because thanks to the abundance of grace and grace I can compile this report properly and correctly, and on time. In this report, I describe the results of creating an automated scheduling system. This report was prepared to fulfill the practice of field work, therefore I would like to thank all parties concerned.

This report is a form of the writer's responsibility in completing field work practices at PT. Panasonic Industrial Device Batam from 18 January 2021 to 17 May 2021 in the framework of dual system education.

In completing this report, the writer received assistance and guidance from various parties. The writer would like to thank:

1. My parents that always pray incessantly for me.
2. Mrs. Lea Lindrawijaya Suroso, M. Pd as Principal of Vocational High School 1 Batam.
3. Mr. Adi Kurniawan, S.Pd as Head of Computer and Network Engineering Department.
4. Mrs. Suci Rahmadhani, S.Pd as Coordinator of Industrial Relations.
5. Mrs. Mike Musna, Gr, S.Pd as Report Counselor.
6. Mrs. Yeni Rifma Yona, S.Pd as Homeroom Teacher.
7. Mr. M. Yoseph Ezra as Head of Field Work Practice Resistor Department.
8. Mr. Purwanto as Supervisor at Resistor Detpartment.
9. Mrs. Sulastry Manurung as Industry Mentor.
10. Mrs. Betty Nurhaida as Career & Talent Staff PT. Panasonic Industrial Device Batam.
11. Mrs. Nechya Einsfrianas as my Leader.
12. All Leaders and Maintenence staff at Resistor Department.
13. All staff and employees at PT. Panasonic Industrial Device Batam.

14. All friends of Computer and Networking that are always providing support and support in the writing of this report.

The Writer also deeply apologized, if ini this report still many inappropriate words are used. The Writer hopes this report can be useful for students of Vocational High School 1 Batam.

Batam, 17th May 2021

Peter Prigel Wicaksono
NIS. 19-9069

TABLE OF CONTENTS

TABLE OF PICTURES

TABLE OF TABLES

CHAPTER I
PRELIMINARY

A. Issue Background

Industrial work practices aims to determine students' ability to work directly in the real world. So that students can know how to adapt to the work environment, as well as train and apply the skills gained from learning at school. In the current industrial work practice, I have the opportunity to practice at the Panasonic Industrial Device Batam Company. The Panasonic Industrial Device Batam Company is a large company that produces electronic goods. In its management, Panasonic Industrial Device Batam company has implemented a series of automation systems. Some of them implement automation such as automatic machines, automatic data management, and others.

The use of this automation system aims to increase efficiency and so that the Panasonic Company can compete with other companies' products. Although the use of this automation system is not comprehensive, the Panasonic Company has already implemented this system on several machines and data processing. Automated jobs make work easier and save more time. That way the workers only monitoring the quality of the goods and also monitoring the health of the machines.

In this industrial practice activity, I got a project to create a scheduling system using Visual Basic for Application that runs on Microsoft Excel application. Concerned with what I explain in the third paragraph, where monitoring of machine health is enough attention in this automation system. This is so that something unwanted does not occur, such as work accidents, substandard quality, and so on. Therefore, to make it easier to carry out maintenance we need a system that can load maintenance schedules and remind technicians to carry out maintenance in advance automatically. Therefore, I chose the title "**CREATE A SCHEDULING SYSTEM USING VBA IN MICROSOFT EXCEL**".

B. Formulation of the Problem

Based on the background, the problem formulations to be described are :

1. The basic theory of the Visual Basic programming language.
2. The process of creating a scheduling system using Visual Basic for Application.
3. Checking the result of creating a scheduling system using Visual Basic for Application.

CHAPTER II
COMPANY OVERVIEW

A. Company Profile

Picture 2.1 Panasonic Industrial Device Batam

Panasonic company previously named Matsushita Electric Industry is an electronics manufacturer based in Kodama, Osaka Prefecture, Japan. The company was founded by Konosuke Matsushita in 1918. With its first product was a light socket, and it continues to grow until today is famous for its electronic products. In batam, the Panasonic company was first founded on October 19, 1995, located at Muka Kuning Industrial Device. Then the Panasonic company moved to Puri Industrial 2000, Batam Center to be precise on October 1, 2001.

Konosuke Matsushita is a founder of the Panasonic Company which has international scale in 1 generation. Since childhood, Konosuke spent his time studying business and honing his skills to survive. He has a dream to help others improve their lives and make the world a better place. When entering the electric age, Konosuke worked as an electrician, he had a stable job. At that time Konosuke had the idea to make his own socket. Then he started to build his new business called Matsushita Electric Appliance

Manufacturing Works in a small house he rented. People around him are attracted to and follow Konosuke's passion and ideas. And from there the business started getting bigger.

In 1929, there was a wave of major economic crises that hit Japan. Which led Panasonic company executives to suggest laying off half of its employees. But Konosuke refused, he didn't want to lay off his workers, not even one person. This is due to the company's mission, which is "The company's mission is to alleviate poverty." To solve these problems, Konosuke invested in his business and began to anticipate the upcoming social, technological and industrial advances.

After World War 2 ended, in 1945, Konosuke demonstrated his strength in solving problems for national peace and recovery for employees and the public who felt afraid and sad after the world war and continued his business. Konosuke traveled to United State. His building relationships with people who had a background in industry and economics, led him to reflect more on the true meaning of wealth. Then Konosuke returned to Japan, determined to enrich and prosper the people. He built alliances with Philips and brought reforms in many disciplines.

Konosuke's leadership and the vision he created for household electrical appliances, exploded in the Japanese market. This made Konosuke's dream come true. Konosuke's next target is to contribute to improving the lives of people around the world. Matsushita Electric is increasingly developing into a world-class electrical industry. The vision of the Konosuke company spreads throughout the country and makes Konosuke continue to receive support and build friendly and business relationships.

B. Company Vision and Mission

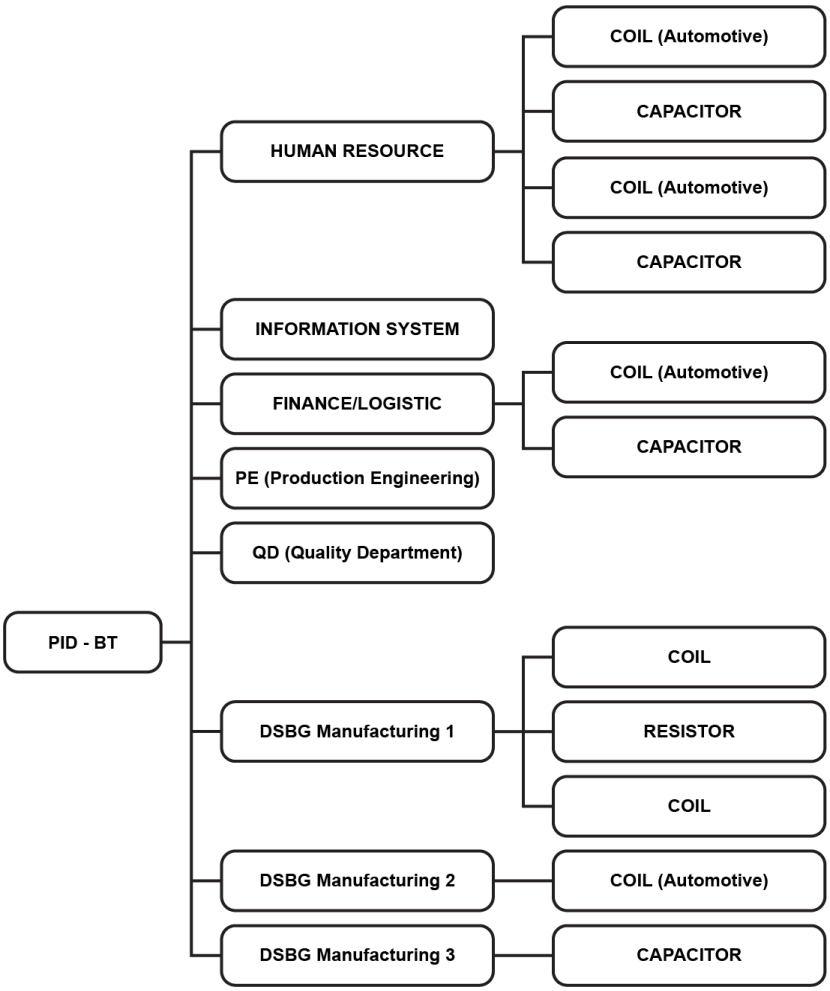
1. Vision

Given our responsibility as industrialists, we will devote oneself to the progress and development of welfare society through our business activities, by improving quality live all over the world.

2. Mission

Basic Management Objectives of Panasonic Corporation, formulated in 1929 by its founder, Konosuke Matsushita. Management's Basic Purpose is Panasonic's business philosophy realize our mission and service for the betterment of society and welfare of people around the world through our business activities.This Basic Management Purpose is Panasonic's business philosophy of realizing our mission and dedication to the betterment of society and the welfare of people around the world through our business activities.

C. Organization Chart



Picture 2.2 Industrial Organization Structure

D. Work Order and Discipline

1. Working Hours

a. Due to corona (covid – 19) normal working hours are replaced and start working from 08.00 until 17.00
b. Arrive on time, at least five minutes before than the starting hours.
c. Scan badge before entering the building, and get used to queuing up.
d. Say greetings to colleagues to boost up morale before starting work.
e. Get used to warm up before starting to work.

2. Uniform

a. Do not wear uniform too fit, too short, and not too loose.
b. Always bring badge or name tag during works.
c. Wearing uniform correctly.

3. Attitudes

a. Before Working
    1) Do the 5S before, while doing, and after works.
        a) Seiri, separating unnecessary items.
        b) Seiton, arranging everything neatly and in its place.
        c) Seiso, clean the work environment.
        d) Seiketsu, keep things in order, clean, and neat.
        e) Shitsuke, diligence and discipline to carry out 5S.
    2) Understand the operation instruction before begin to work.
    3) Listen, understand, and follow instructions and rules delivered by the leader or supervisor.
    4) Cell phones are prohibited during the working hours.
    5) No smoking at work, or in toilets. Smoking is only allowed in the smoking area.
    6) Not allowed to eat or drink in production area, locker room, training room or working area. Eat or drink only allowed in canteen.

7) Prohibited to damage or steal company's stuffs.
8) Prohibited from committing a disgraceful act.
9) Throw garbage in its place.
10) Prohibited from carrying sharp weapons and illegal drugs in the company area.
11) Do not fight in the company area.

b. While Working
1) Be polite and respect each other colleagues.
2) Always pay attention to operation instruction before begin to work.
3) Communicate well with colleagues.
4) Do not disturb other colleagues while working.
5) Do not take action that can provoke a commotion.

c. After Working
1) Fill the record sheet before going home.
2) Keep things in working area in order.
3) Make sure the machine power is off.
4) Do not forget to scan badge before exiting the building.

4. Work Safety

Safety means protecting yourself and others from accidents and avoiding the risk of damage to company facilities. Some of the most common accidents that occur at work are:

a. Electric shock.
b. Limbs squeezed by machine.
c. Accident for not paying attention to 5S.
1) Cause of Accident
a) Not following operation instruction when working.
b) Joking and not being serious in work.
c) Sleepy while working.
d) Not concentrating when working.

2) Safety Rules

e) Check equipment and machine before use, and immediately replace defective equipment.

f) For your safety, wear personal protective equipment correctly.

g) Listen, understand, and follow instructions and rules work delivered by the leader or your supervisor.

h) Check and maintain machine regularly.

i) Do not running machine, if you don't have authority.

5. Work Security

a. Asset Security

Asset security is security in the form of protection to employees and company properties. Here are some examples of asset security:

1) Prohibited to bring company properties out without permitted.

2) Any employees can't leave company during the working hour without permit from head of the department.

b. Information Security

Information security is security in the form to protect information in company so not leaked to public.

1) Do not discuss confidential information in any place.

2) Prohibited to bring any electronic devices, such as laptop and flash drive inside to company.

6. Sanctions

Disciplinary sanctions are consequences that have been agreed upon by both parties. Here are some explanations of the disciplinary crimes that exist in the company:

a. Termination of employment for serious offenses.

b. A proper investigation will be done to find out the size of the case before a decision is made for sanctions to be given.

7. Quality Policies of Panasonic Industrial Devices Batam

a. Making customer choice number one in quality is a top priority.
b. Provide high-quality, safe, secure and reliable product through 5S implementation and process control.
c. Continuous quality improvement on processes and systems for "0" defect.
d. Comply with applicable statutes and regulations.
e. Motivate all colleagues to earnestly to achieve quality goals with full dedication and teamwork.
f. Implement a systematic problem-solving method through the 5 GEN principles:
    1) Genri: Following, troubleshooting or continuous improvement processes.
    2) Gensoku: Following the Rules or Procedures in troubleshooting and improving the process.
    3) Genba: Where the work process running.
    4) Genbutsu: See an event or object with its own eyes or feel it yourself and touch it with your own hands.
    5) Genjitsu: Facts that must be felt, such as symptoms of problems and their effects on production processes and work processes in an industry.

CHAPTER III
CREATE A SCHEDULING SYSTEM USING VBA IN MICROSOFT EXCEL AT PT. PANASONIC INDUSTRIAL DEVICE BATAM

A. General Theory

1. Microsoft Excel

Picture 3.1 Microsoft Excel

Microsoft Excel is a program or application that is part of the Microsoft Office installation package. Has a function to process numbers using a spreadsheet consisting of rows and columns to execute commands. Microsoft Excel has become the best data / number processing software in the world, besides Microsoft Excel has been distributed multi-platform. Microsoft Excel can run on Windows, Android, MacOS and Apple.

Microsoft Excel fundamentally uses spreadsheets for data management and performs Excel functions known as "Excel formulas". And also there is Macro feature. The tools used in making commands and recording Macros are Visual Basic. The purpose of making Macros is so that all commands that the user gives will be recorded by the Excel application in the Visual Basic language and displayed in the Microsoft Visual Basic Editor program.

2. Visual Basic

Picture 3.2 Visual Basic

Visual Basic is a programming language. Programming languages are the commands understood by the computer to perform certain tasks. Visual Basic is a programming language derived from the BASIC language. This programming language was developed by the Microsoft company since 1991. Visual Basic is one of the Development Tools which is a tool for creating various kinds of computer programs, especially those using the Windows operating system.

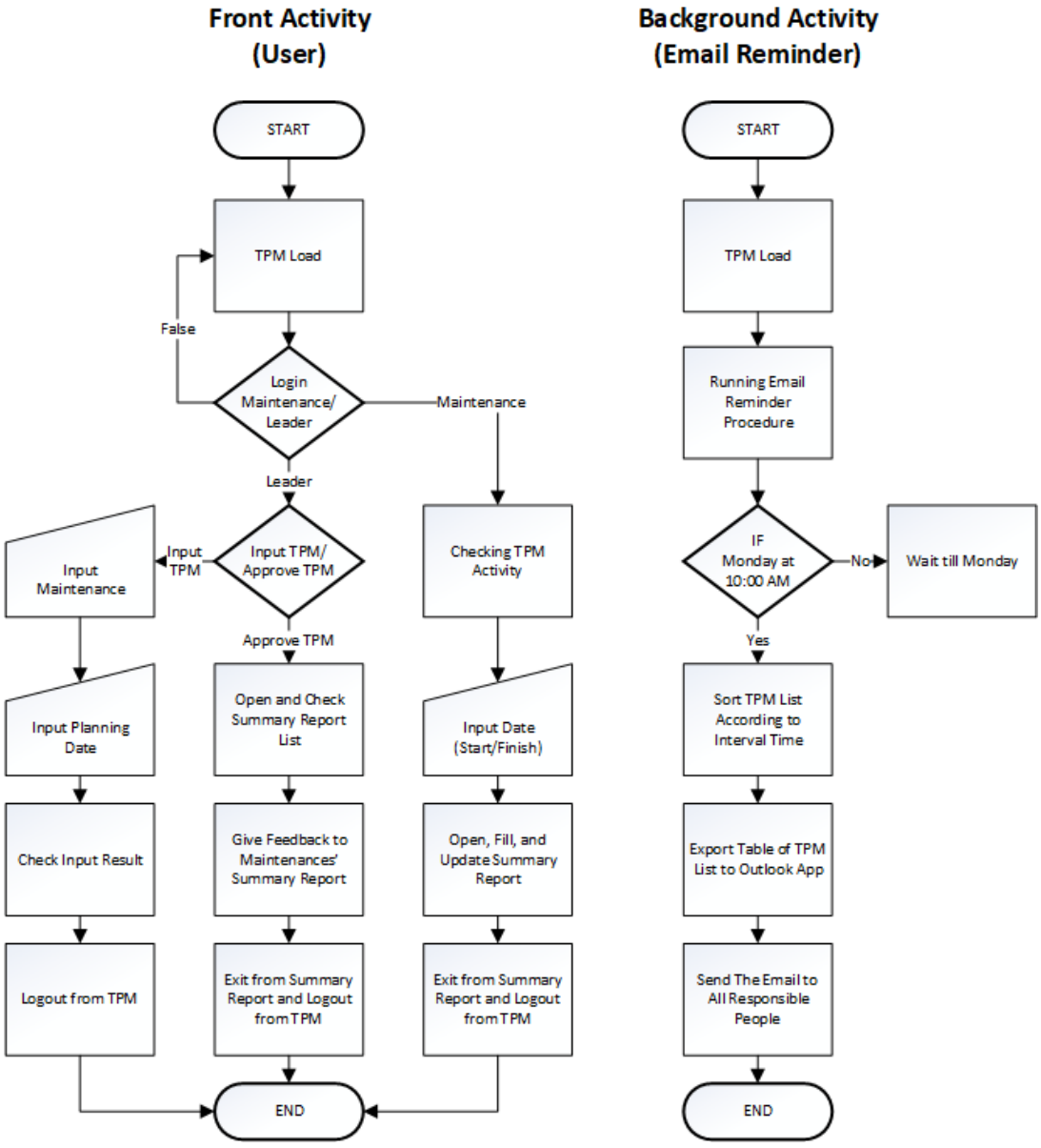
Visual Basic was created from an idea to create a programming language that is simple and easy in making scripts (Simple scripting language) for the graphic user interface developed by the Microsoft Windows operating system. Visual Basic is also a programming language that is very easy to learn, with visual programming techniques that allow users to be better creative in producing an application program. This can be seen from the basic creation in visual basic is FORM, where users can set the appearance of the form and then run it in a very easy script.

B. Tools and Materials

Table 3.1 Tool, Material, and Specification

NO	TOOL AND MATERIALS	SPECIFICATION
1	Microsoft Office Excel	Visual Basic for Application on Excel
2	Computer	Personal Computer Windows OS

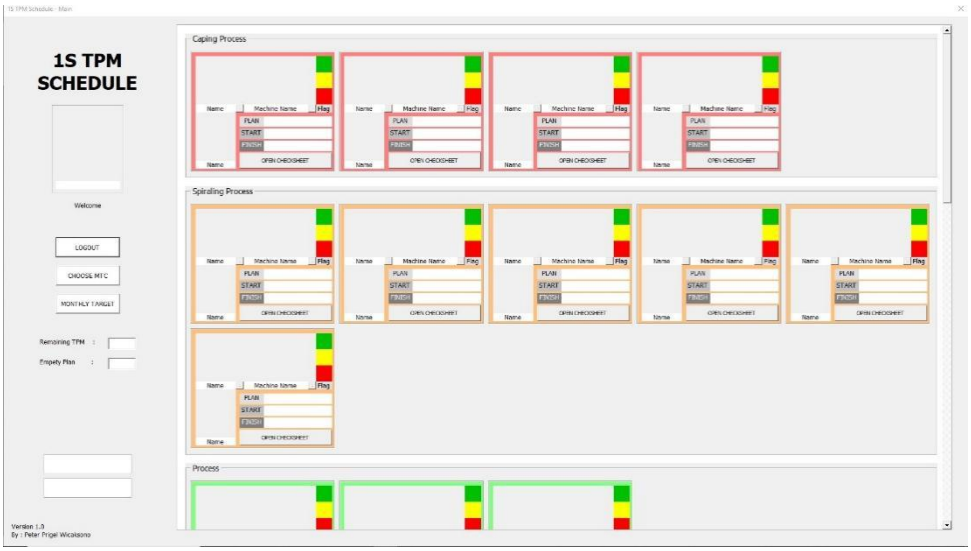
C. System Flowchart



Picture 3.3 System Flowchart

D. Work Steps

1. Main View System



Picture 3.4 Main View of The System

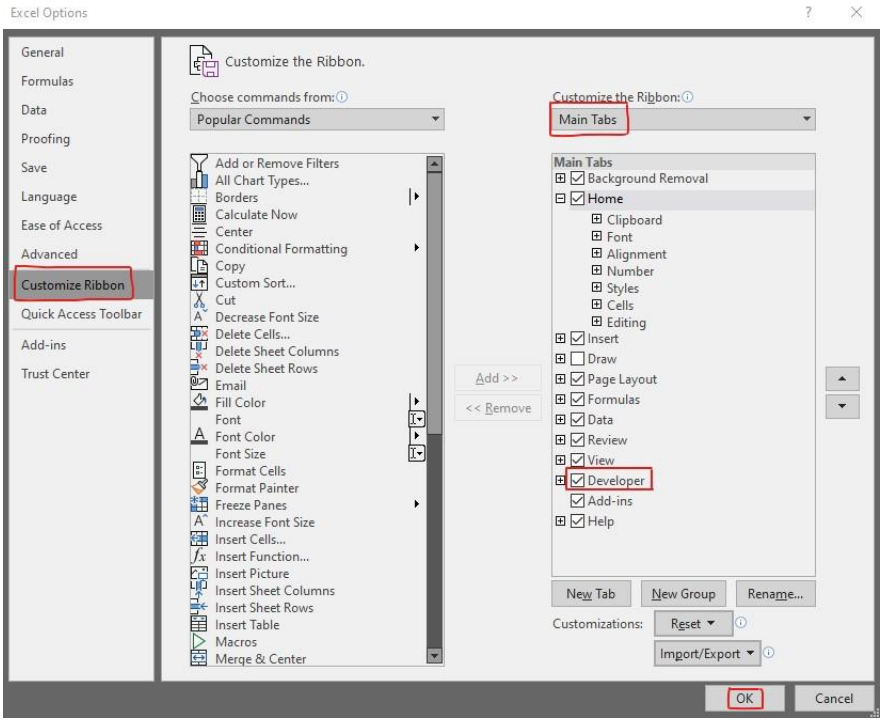
2. Preparation

Before starting to make programming, it's a good idea to activate the Developer Tab on the Ribbon menu to optimize Excel's Macro features. The steps are as follows.

a. To activate the Developer Tab, select the File menu on the Ribbon. Then select the Option menu located at the bottom.

b. After the panel appears, select Customize Ribbon.

c. Next, pay attention to the marked part. In the bottom table, look for the sentence that says Developer, then tick the sentence.

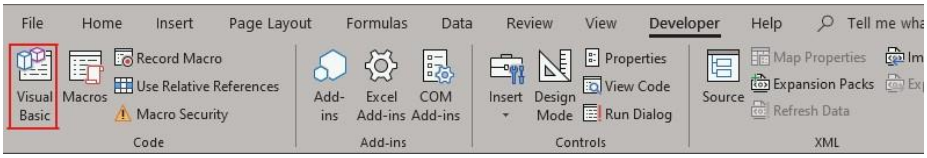


Picture 3.5 Excel Options

d. Once checked, click the OK button at the bottom.

Next, try to open the Visual Basic Editor. There are several ways to open the Visual Basic Editor. The following are among them.

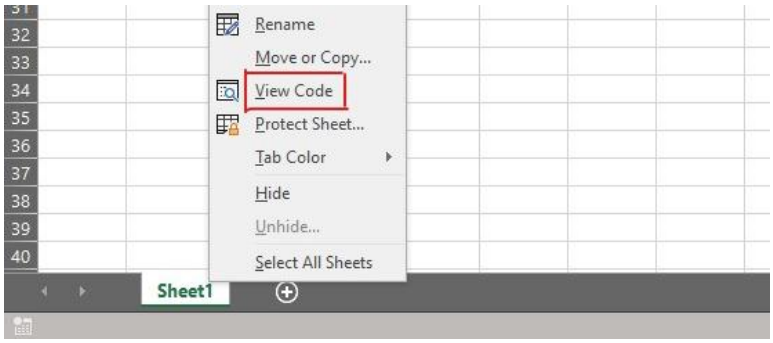
a. The first way is to go to the Developer Tool on the Ribbon. Then on the left side of the Ribbon click on the symbol and the words Visual Basic.



Picture 3.6 Developer Tab

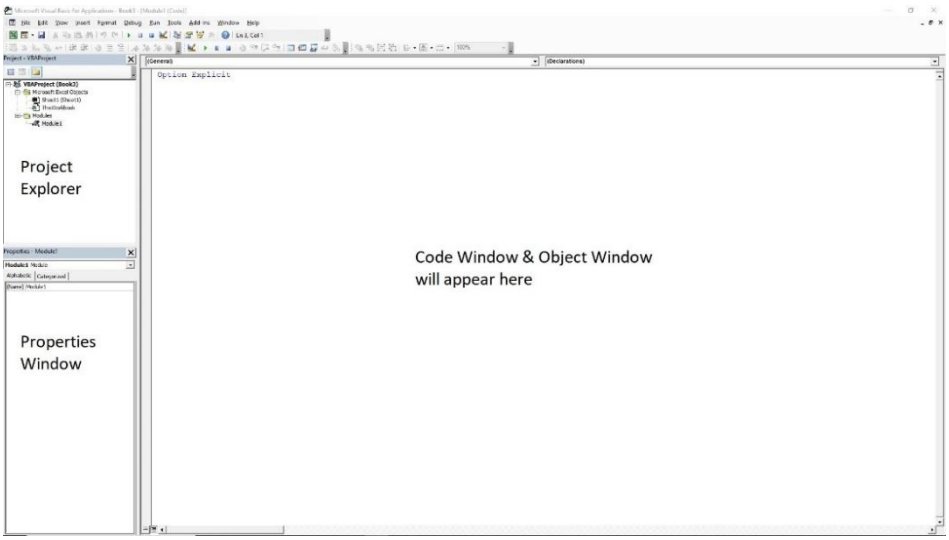
b. The second way is by using keyboard shortcuts. Press the Alt key and press F11 key at the same time to open the Visual Basic Editor (Alt + F11).

c. The third way is by right clicking on the sheet, then selecting View Code.



Picture 3.7 Right Click on Sheet

The following is an image of the part in the Visual Basic Editor display.



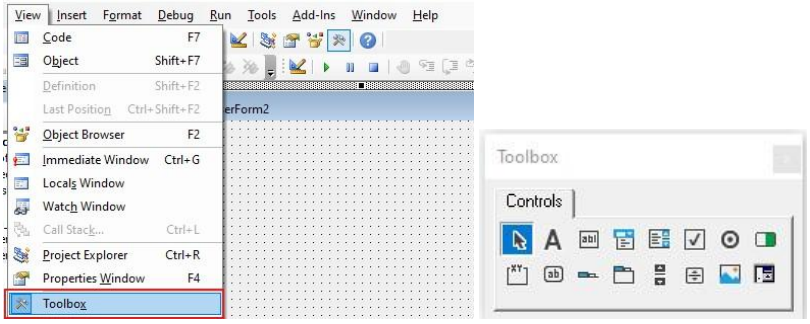
Picture 3.8 Section in VBA Editor Excel

Information :

a. Project Explorer, is used to navigate to all objects in the VBA WorkBook project. Go to the menu View - Project Explorer (Ctrl + R) to display this window.
b. Properties window, is used to display properties that are owned by an object. Go to the menu View - Properties Window (F4) to display this window.
c. Window Code, is used to view, create, or modify Macro code.

d. Window Object, is a place used to visually display, create, and organize UserForm objects and control objects in the UserForm. Go to the View - Object (Shift + F7) menu to display this window.

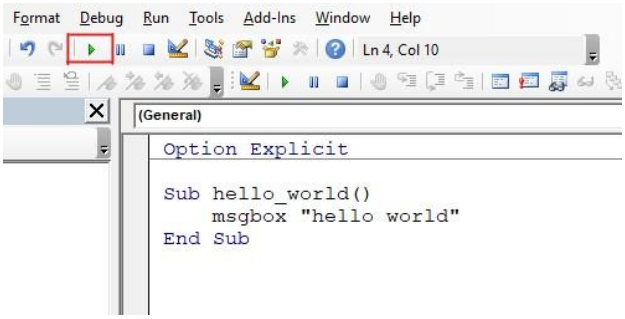
Next, to design a Form we need the ToolBox feature. Click the view menu then select ToolBox to activate this feature.



Picture 3.9 ToolBox

When we have created the code, we can run the code to be executed and done by the computer. There are several ways including the following.

a. The first way is by pressing the F5 key on the keyboard.
b. The second way is by clicking the green play icon at the top of the Visual Basic Editor window.



Picture 3.10 Code Execution

To make it easier and prevent files needed to make the program disappear, create a special folder for this project. In this folder, you can fill in the necessary files, such as images, Excel files, icons, and so on.

Also use sub folders to classify these files so that they look more neat and structured.

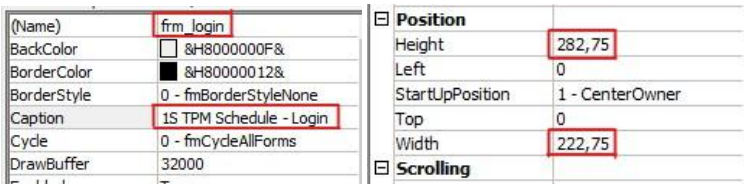
To save a file containing Macro code a special format is required. The format used is Excel Macro-Enabled Workbook (* .xlsm). If you use the regular format (* .xlsx), the WorkBook can still be opened, but the Macro cannot be run.

To make work more easy, use code name when giving a name for an object. For example, "ica1", each of alphabet has a meaning. According to the example, "i" is for "image", " c" is for "capping" which is a process name, "a" is used to giving a different between twins object (like picture "a" and picture "b"), and "1" is the number of the machine. But use this trick for the objects that has same function. For buttons can make a normal name like "btn_login" and other.

3. Designing Form

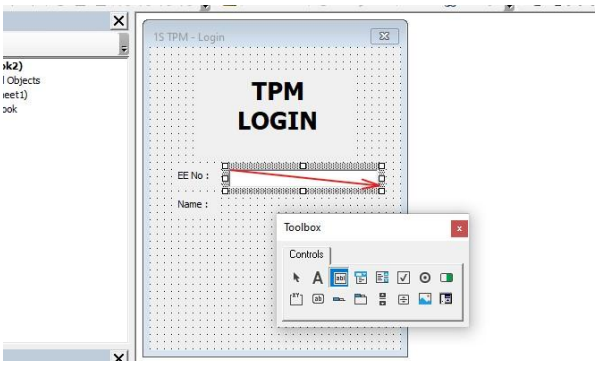
a. Login Form

1) Open Visual Basic Editor and make a new form (*Insert – UserForm*). Change the height of the form to 282,75 and width of the form to 222,75. Change name to frm_login and change the caption to "1S TPM Schedule – Login". Change them in property panel.



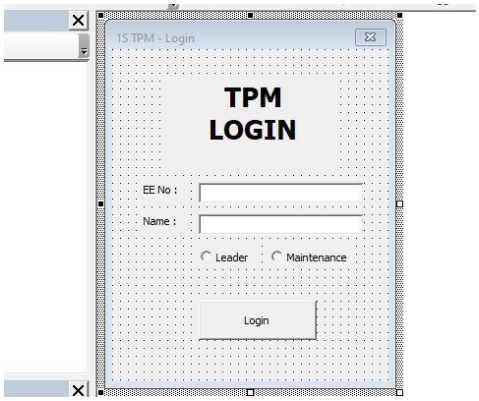
Picture 3.11 Change Properties in Login Form

2) Next, add some object. That are 2 textbox, 3 label, 2 option button and 1 button. Before that, activate the ToolBox, activate it in View menu (*View – ToolBox*). Choose one tool that inside the ToolBox. Point the cursor inside the form, then drag the cursor.



Picture 3.12 Make an Object

3) The final result of the Login Form design will look like the one below.

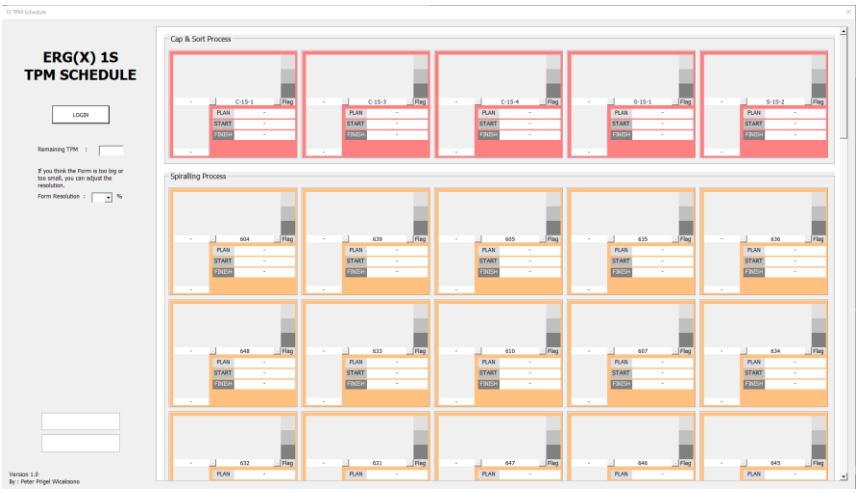


Picture 3.13 Form Login Layout

b. Cover Form

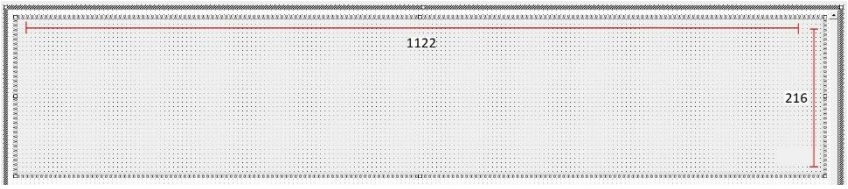
1) Insert new UserForm for the Cover Form. Change the height to 810 and width to 1440. Change name to frm_cover and give the caption “1S TPM Schedule”.

2) Cover Form will look like the picture below.



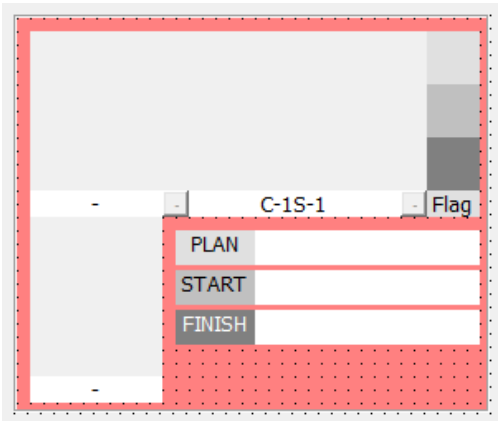
Picture 3.14 Cover Form Preview

3) Like the picture above, make some frame according to the picture below.



Picture 3.15 Size of The Process Frame

4) Make some little squares inside the Frame1. This little squares use for showing the Schedule that has been inputted before.



Picture 3.16 Little Square

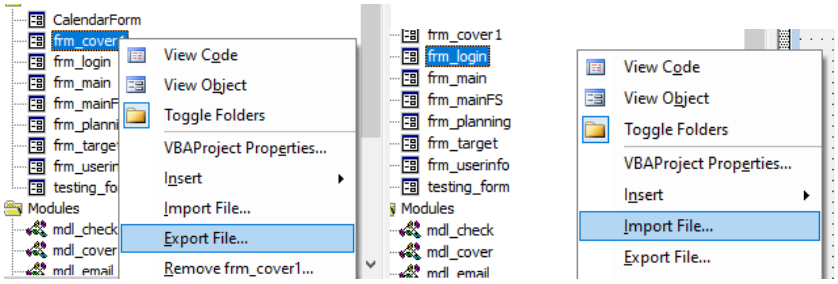
Each of the Frame will has the same size. Give the caption for each Frame. Consecutively frames 1 - 6 are given the caption Cap & Sort Process, Spiralling Process, Laser Process, Welding Process, Finishing Process, and Forming Process.

5) In this case, use code name to giving the name of the objects. Because there are a lot of objects that has same function.

c. Main Form

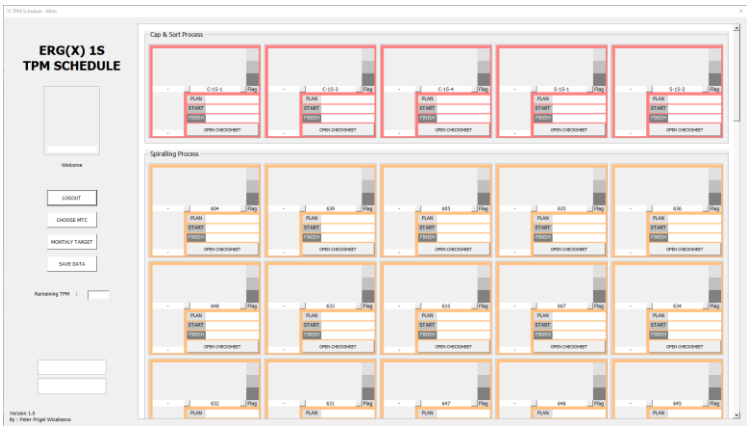
1) For the Main Form, it will more or less look like Cover Form. Export the Cover Form to a file, then we can import it to the VBA. Make sure the name of the form that will be exported must changed first (for example frm_cover1).

2) Right click in the frm_cover1 then choose export. Select/make a folder that you are going to use, then click OK. Back to the VB Editor, change the name "frm_cover1" back to the original "frm_cover", then right click inside the Project Explorer and choose import. Search the folder that contain the frm_cover1 that been exported before, choose the file, and click OK.



Picture 3.17 Export & Import Form to File

3) Change the name "frm_cover1" to "frm_main". The Main Form will be look like below. Add a button to each existing little squares, add some object, and delete some object that does not match with the picture below. Use code name to giving names of the "Open Checksheet" button.

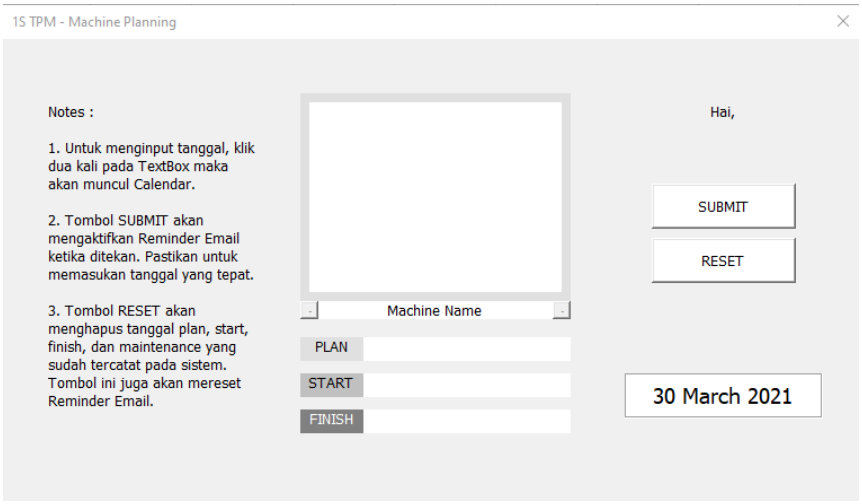


Picture 3.18 Main Form Preview

d. Planning Form

1) Insert new UserForm. Change the height to 336,75 and width to 582,75. Change name to frm_cover and give the caption "1S TPM Schedule – Machine Planning".

2) The final result will be look like the one below.

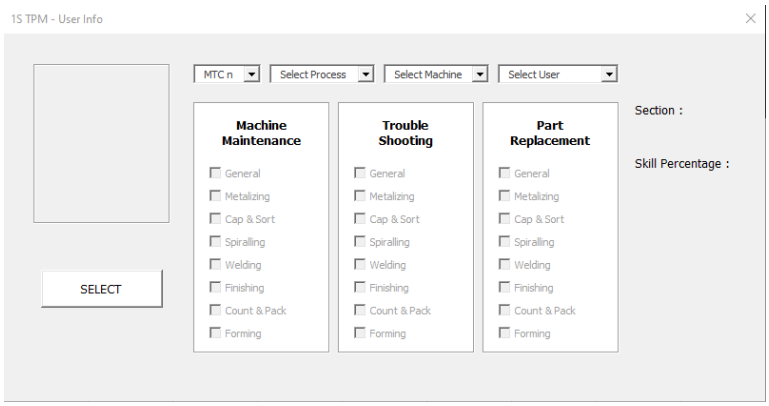


Picture 3.19 Planning Form Preview

e. User Info Form

1) Insert new UserForm. Change the height to 318,75 and width to 612,75. Change name to frm_cover and give the caption "1S TPM Schedule – User Info".

2) The final result will be look like the one below.

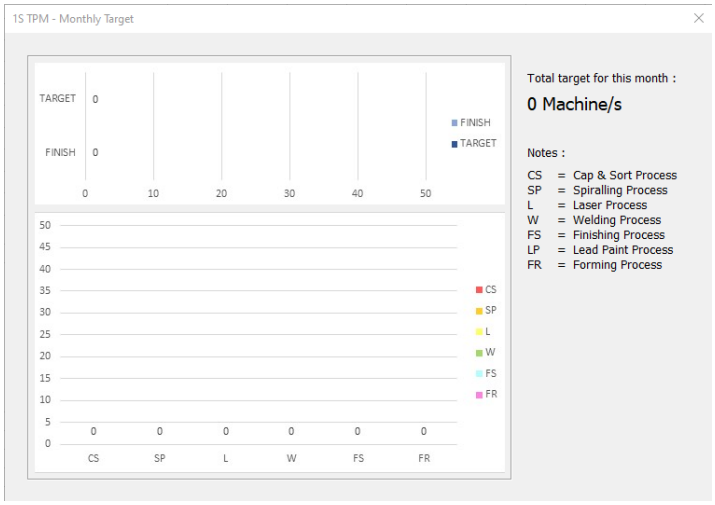


Picture 3.20 User Info Form Preview

f. Monthly Target Form

1) Insert new UserForm. Change the height to 405,75 and width to 581,25. Change name to frm_target and give the caption "1S TPM Schedule – Monthly Target".

2) The final result will be look like the one below.



Picture 3.21 Monthly Target Form Preview

4. Adding Code for Login System

Prepare some data first. These data will be link with the Login Form. The data will be the condition for Login system.

a. Go to the Sheet1. Then, rename the "Sheet1" to "userlogin". Type the data in the picture below.

Leader	
Name	EE No.
PURWANTO	002108
DARWANTO	002284
NECHYA	931678

Maintenance	
Name	EE No.
RAHMAT HIDAYAT	932074
ABDUL ROHMAN	934925
WIDI HARTOMO	934931
HARI PURNOMO	002023
KUKUH JATMIKO	931886
EDI SANTOSA	916973
SUPRIONO	002112
GR NUKE	916631
YONDRI FAIZAL	917592
MARWANTO	936457
RONI FAZLI	914874
DIDIK NURCAHYONO	916624
YULI HENDRI	919077
ARIEF IRWANTO	200144

Authorized	
Name	EE No.
PURWANTO	002108
DARWANTO	002284

Picture 3.22 Sheet 1 Data

b. Create the code on the Login Form. Give the code to btn_login, so that it can process the input entered in txt_eeno and txt_name. Double click on btn_login then type the code below. The event used in btn_login is "private sub btn_login_Click ()".

```
'login system {
ThisWorkbook.Sheets("userlogin").Activate
    If opt_leader.Value = True Then
        GoTo leader
    ElseIf opt_maintanance.Value = True Then
        GoTo maintanance
    ElseIf opt_leader.Value = False And opt_maintenance.Value = False Then
        GoTo authorized
    End If

leader:
    If txt_name.Value = Range("B4") And txt_eeno.Value = Range("C4") Then
        frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\leader\01 Purwanto.JPG")
        frm_main.lbl_welcomename.Caption = Range("B4").Value
    ElseIf txt_name.Value = Range("B5") And txt_eeno.Value = Range("C5") Then
        frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\leader\02 Darwanto.JPG")
        frm_main.lbl_welcomename.Caption = Range("B5").Value
    ElseIf txt_name.Value = Range("B6") And txt_eeno.Value = Range("C6") Then
        frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\leader\03 Nechya.JPG")
        frm_main.lbl_welcomename.Caption = Range("B6").Value
    ElseIf txt_name = "" Then
        MsgBox "Please input your Name", vbExclamation, "TPM - Login"
        Exit Sub
    ElseIf txt_eeno = "" Then
        MsgBox "Please input your EE No", vbExclamation, "TPM - Login"
        Exit Sub
    Else
        MsgBox "Incorrect" & vbCrLf & "Please check your Name, EE No, and your option", vbCritical, "TPM - Login"
        Exit Sub
    End If
    frm_main.lbl_loginas.Caption = "Leader"
    Unload Me: Unload frm_cover
    Call open_frm_main
    Exit Sub
```

Picture 3.23 System Login Code Picture 1

```
maintenance:
    If txt_name.Value = Range("E4") And txt_eeno.Value = Range("F4") Then
        frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\maintenance\01 Rahmat H.JPG")
        frm_main.lbl_welcomename.Caption = Range("E4").Value
    ElseIf txt_name.Value = Range("E5") And txt_eeno.Value = Range("F5") Then
        frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\maintenance\02 Abdul R.JPG")
        frm_main.lbl_welcomename.Caption = Range("E5").Value
    ElseIf txt_name.Value = Range("E6") And txt_eeno.Value = Range("F6") Then
        frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\maintenance\03 Widi.JPG")
        frm_main.lbl_welcomename.Caption = Range("E6").Value
    ElseIf txt_name.Value = Range("E7") And txt_eeno.Value = Range("F7") Then
        frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\maintenance\04 Hari Purnomo.JPG")
        frm_main.lbl_welcomename.Caption = Range("E7").Value
    ElseIf txt_name.Value = Range("E8") And txt_eeno.Value = Range("F8") Then
        frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\maintenance\05 Kukuh J.JPG")
        frm_main.lbl_welcomename.Caption = Range("E8").Value
    ElseIf txt_name.Value = Range("E9") And txt_eeno.Value = Range("F9") Then
        frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\maintenance\06 Edi S.JPG")
        frm_main.lbl_welcomename.Caption = Range("E9").Value
    ElseIf txt_name.Value = Range("E10") And txt_eeno.Value = Range("F10") Then
        frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\maintenance\07 Supriono.JPG")
        frm_main.lbl_welcomename.Caption = Range("E10").Value
    ElseIf txt_name.Value = Range("E11") And txt_eeno.Value = Range("F11") Then
        frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\maintenance\08 GR.Nuke.JPG")
        frm_main.lbl_welcomename.Caption = Range("E11").Value
    ElseIf txt_name.Value = Range("E12") And txt_eeno.Value = Range("F12") Then
        frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\maintenance\09 Yondri F.JPG")
        frm_main.lbl_welcomename.Caption = Range("E12").Value
    ElseIf txt_name.Value = Range("E13") And txt_eeno.Value = Range("F13") Then
        frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\maintenance\10 Marwanto.jpg")
        frm_main.lbl_welcomename.Caption = Range("E13").Value
    ElseIf txt_name.Value = Range("E14") And txt_eeno.Value = Range("F14") Then
        frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\maintenance\11 Roni Fazli.jpg")
        frm_main.lbl_welcomename.Caption = Range("E14").Value
     ElseIf txt_name.Value = Range("E15") And txt_eeno.Value = Range("F15") Then
         frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\maintenance\12 Didik N.JPG")
         frm_main.lbl_welcomename.Caption = Range("E15").Value
     ElseIf txt_name.Value = Range("E16") And txt_eeno.Value = Range("F16") Then
         frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\maintenance\13 Yuli Hendri.JPG")
         frm_main.lbl_welcomename.Caption = Range("16").Value
     ElseIf txt_name.Value = Range("E17") And txt_eeno.Value = Range("F17") Then
         frm_main.img_user.Picture = LoadPicture(ThisWorkbook.Path & "\images\maintenance\14 Arief I .JPG")
         frm_main.lbl_welcomename.Caption = Range("E17").Value
     ElseIf txt_name = "" Then
         MsgBox "Please input your Name", vbExclamation, "TPM - Login"
         Exit Sub
     ElseIf txt_eeno = "" Then
         MsgBox "Please input your EE No", vbExclamation, "TPM - Login"
         Exit Sub
     Else
         MsgBox "Incorrect" & vbCrLf & "Please check your Name, EE No, and your option", vbCritical, "TPM - Login"
         Exit Sub
     End If
     frm_main.lbl_loginas.Caption = "Maintenance"
     Unload Me: Unload frm_cover
     Call open_frm_main
     Exit Sub
```

Picture 3.24 System Login Code Picture 2

c. After giving the code to btn_login. Next, make a condition when "EE No" is typed and "EE No" is correctly matches with the data on the Excel sheet, txt_name will automatically display the user name that matches with the "EE No". With this, users can reduce typing activities on the keyboard and users can also scan barcodes on their business cards, thus shortening time and making work easier. Double click on txt_eeno. Type the code below. The event used in txt_eeno is "private sub txt_eeno_Change ()".

```
'show name when EE No has been write and true {
Private Sub txt_eeno_Change()
ThisWorkbook.Sheets("userlogin").Activate
    'leader
    If txt_eeno.Value = Range("C4") Then
        txt_name.Value = Range("B4")
    ElseIf txt_eeno.Value = Range("C5") Then
        txt_name.Value = Range("B5")
    ElseIf txt_eeno.Value = Range("C6") Then
        txt_name.Value = Range("B6")
    'maintenance
    ElseIf txt_eeno.Value = Range("F4") Then
        txt_name.Value = Range("E4")
    ElseIf txt_eeno.Value = Range("F5") Then
        txt_name.Value = Range("E5")
    ElseIf txt_eeno.Value = Range("F6") Then
        txt_name.Value = Range("E6")
    ElseIf txt_eeno.Value = Range("F7") Then
        txt_name.Value = Range("E7")
    ElseIf txt_eeno.Value = Range("F8") Then
        txt_name.Value = Range("E8")
    ElseIf txt_eeno.Value = Range("F9") Then
        txt_name.Value = Range("E9")
    ElseIf txt_eeno.Value = Range("F10") Then
        txt_name.Value = Range("E10")
    ElseIf txt_eeno.Value = Range("F11") Then
        txt_name.Value = Range("E11")
    ElseIf txt_eeno.Value = Range("F12") Then
        txt_name.Value = Range("E12")
    ElseIf txt_eeno.Value = Range("F13") Then
        txt_name.Value = Range("E13")
    ElseIf txt_eeno.Value = Range("F14") Then
        txt_name.Value = Range("E14")
    ElseIf txt_eeno.Value = Range("F15") Then
        txt_name.Value = Range("E15")
    ElseIf txt_eeno.Value = Range("F16") Then
        txt_name.Value = Range("E16")
    ElseIf txt_eeno.Value = Range("F17") Then
        txt_name.Value = Range("E17")
    Else
        txt_name.Value = ""
    End If
End Sub
'}
```

Picture 3.25 Change Event Code in EE No. Label Box

5. Adding Code for Link The Form with Other Form

Next, link the Main Form with other form when certain events are held.

a. First, the User Info Form will appear when the user click CHOOSE MTC button. To make it, double click on CHOOSE MTC button. Then write the code below.

```
Private Sub btn_mtc_Click()
    frm_userinfo.Show
End Sub
```

b. After that, the Monthly Target Form will show when the user click the MONTHLY TARGET button. Double click on MONTHLY TARGET button then type the code below.

```
Private Sub btn_target_Click()
    frm_target.Show
End Sub
```

c. Next, the Planning Form will appear if the user click the image of the machine. The Planning Form will look different when the user click other machine image. So, every machine image will take different effect to the Planning Form when it clicked. Look at the code below.

```
Private Sub img_c1_Click()
    frm_planning.f_container.BackColor = &H8080FF
    frm_planning.lbl_machinename.Caption = "C-1S-1"
    frm_planning.Show
End Sub
...
```

That is an example from button machine C-1S-1. When it get click, the Frame in the Planning Form (f_container) change to the pastel red color, and the caption of Label in the Planning Form (lbl_machinename) change to "C-1S-1". For the detail, these are the color identity for every process.

Table 3.2 Color Identity for Every Process

OBJECT	COLOUR
Cap & Sort	Pastel Red = &H008080FF&
Spiralling	Pastel Orange = &H0080C0FF&
Laser	Pastel Yellow = &H0080FFFF&
Welding	Young Green = &H0080FF80&
Finishing	Light Blue = &H00FFFF80&
Forming	Light Purple = &H00FF8080&

Continue to create the code in every image of the machine, make sure to make the code according to the specifications. For the lbl_machinename (on frm_planning), fill in the caption according to the machine name.

6. Code to Input Maintenance from User Info Form to Workbook

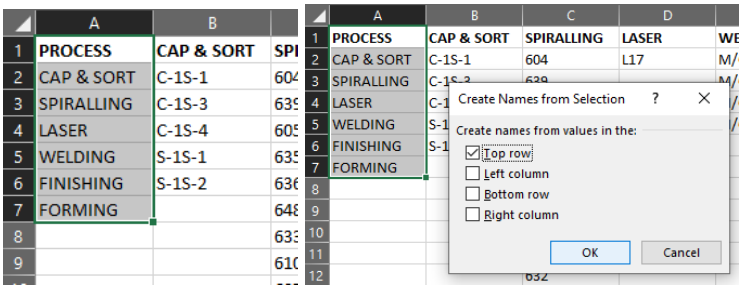
a. First, make the code for the all ComboBoxes in the User Info Form. In this case, the ComboBox will be link with a sheet called "cboxValue". So make new sheet and rename it with "cboxValue".

b. Inside the "cboxValue", type all this word.

	A	B	C	D	E	F	G	H	I	J
1	PROCESS	CAP & SORT	SPIRALLING	LASER	WELDING	FINISHING	FORMING		MAINTENANCE	MTC
2	CAP & SORT	C-1S-1	604	L17	M/C No. 2412	05 - Detaping	2		RAHMAT HIDAYAT	MTC 1
3	SPIRALLING	C-1S-3	639		M/C No. 2426	05 - Straightener 1	75		ABDUL ROHMAN	MTC 2
4	LASER	C-1S-4	605		M/C No. 2424	05 - Preheater	19		WIDI HARTOMO	
5	WELDING	S-1S-1	635		M/C No. 2419	05 - Coating	70		HARI PURNOMO	
6	FINISHING	S-1S-2	636			05 - Cooler	17		KUKUH JATMIKO	
7	FORMING		648			05 - Furnace 1	12		EDI SANTOSA	
8			633			05 - Furnace 2	46		SUPRIONO	
9			610			05 - Camera			GR NUKE	
10			607			05 - RV & THI			YONDRI FAIZAL	
11			634			05 - Printing			MARWANTO	
12			632			05 - UV Oven			RONI FAZLI	
13			631			05 - Straightener 2			DIDIK NURCAHYONO	
14			647			05 - Taping			YULI HENDRI	
15			646			02 - Detaping			ARIEF IRWANTO	
16			645			02 - Straightener 1				
17			640			02 - Preheater				
18			641			02 - Coating				
19			642			02 - Cooler				
20			643			02 - Furnace 1				
21			644			02 - Furnace 2				
22						02 - Camera				
23						02 - RV & THI				
24						02 - Printing				
25						02 - UV Oven				
26						02 - Straightener 2				
27						02 - Taping				

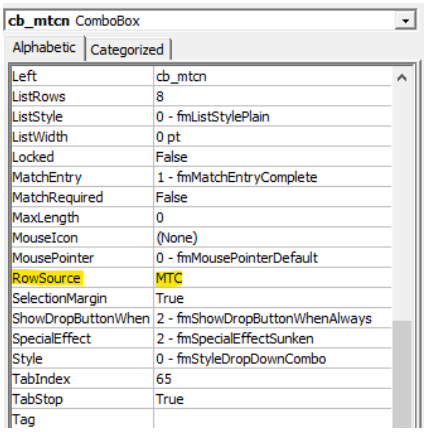
Picture 3.26 Data Inside Sheet “cboxValue”

c. Look at the “cboxValue” sheet, column A. Block range A1 to A7, then type shortcut Ctrl + Shift + F3, choose “Top Row”, and click OK. This thing is used to grouping some cell and link it with one member of the group. So, it seems like a group with one leader. Next, do it the same way for other columns.



Picture 3.27 Create Names from Selection

d. Then, link all the “selection” with ComboBox in the User Info Form. Go to the VB Editor, User Info Form. First, click to cb_mtcn, then look at the Property Window. Look for RowSource, then type “MTC”.



Picture 3.28 Giving The Row Source for Combo Box

e. Do the same way for cb_process and cb_name. The cb_process will be link with the selection in column A and the cb_name will be link with in column "I".

f. Give some code to the cb_process. Double click on cb_process, then type the code below.

```
Private Sub cb_process_Change()
Me.cb_machine = "Select Machine"
Select Case Me.cb_process
    Case "CAP & SORT"
        Me.cb_machine.RowSource = "capsort"
    Case "SPIRALLING"
        Me.cb_machine.RowSource = "spiralling"
    Case "LASER"
        Me.cb_machine.RowSource = "laser"
    Case "WELDING"
        Me.cb_machine.RowSource = "welding"
    Case "FINISHING"
        Me.cb_machine.RowSource = "finishing"
    Case "LEAD PAINT"
        Me.cb_machine.RowSource = "leadpaint"
    Case "FORMING"
        Me.cb_machine.RowSource = "forming"
End Select
End Sub
```

Picture 3.29 Row Source Change for Label Box in User Info Form

g. After that, make script for the SELECT button. It will use IF statement according to the value of cb_process and cb_machine. Double click the SELECT button, then type the script below. This is an example for Cap & Sort process.

```
Private Sub btn_select_Click()
ThisWorkbook.Sheets("plan").Activate
   'cap&sort
   If cb_machine = "C-1S-1" And cb_mtcn = "MTC 1" Then
       Range("U2").value = cb_name.value
   ElseIf cb_machine = "C-1S-1" And cb_mtcn = "MTC 2" Then
       Range("U3").value = cb_name.value
   ElseIf cb_machine = "C-1S-3" And cb_mtcn = "MTC 1" Then
```

```
        Range("U4").value = cb_name.value
    ElseIf cb_machine = "C-1S-3" And cb_mtcn = "MTC 2" Then
        Range("U5").value = cb_name.value
    ElseIf cb_machine = "C-1S-4" And cb_mtcn = "MTC 1" Then
        Range("U6").value = cb_name.value
    ElseIf cb_machine = "C-1S-4" And cb_mtcn = "MTC 2" Then
        Range("U7").value = cb_name.value
    ElseIf cb_machine = "S-1S-1" And cb_mtcn = "MTC 1" Then
        Range("U8").value = cb_name.value
    ElseIf cb_machine = "S-1S-1" And cb_mtcn = "MTC 2" Then
        Range("U9").value = cb_name.value
    ElseIf cb_machine = "S-1S-2" And cb_mtcn = "MTC 1" Then
        Range("U10").value = cb_name.value
    ElseIf cb_machine = "S-1S-2" And cb_mtcn = "MTC 2" Then
        Range("U11").value = cb_name.value
    '}
     ...
     'spiralling {
     ...
     End If
End Sub
```

h. Do the same way for other process with changing the value of cb_machine and cb_mtcn. Don't forget to sync data to right ranges.

i. In the end of IF statement, give this code.

```
...
Else
    MsgBox "Please Input the Right Option", vbExclamation
End If
```

7. Adding Skill Details and User Image for The User Info Form

Prepare some Images (User Image). Make new folder called "Images" then put all the images there. Next, make the table below.

RAHMAT HIDAYAT			ABDUL RAHMAN			WIDI HARTOMO			HARI PURNOMO			KUKUH JATMIKO			EDI SANTOSA		
TPM	TS	PR	TPM	TS	PR	TPM	TS	PR	TPM	TS	PR	TPM	TS	PR	TPM	TS	PR
1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	0	0	1
1	1	1	1	1	1	0	1	1	1	1	1	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	1	1	1
1	1	1	1	1	1	1	1	1	0	0	0	0	0	1	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0

SUPRIONO			GR NUKE			YONDRI FAIZAL			MARWANTO			RONI FAZLI			DIDIK NURCAHYONO		
TPM	TS	PR	TPM	TS	PR	TPM	TS	PR	TPM	TS	PR	TPM	TS	PR	TPM	TS	PR
1	0	1	1	0	1	0	0	1	1	0	1	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
0	0	1	0	0	1	0	0	0	0	0	1	1	1	1	1	1	1

YULI HENDRI			ARIEF IRWANTO		
TPM	TS	PR	TPM	TS	PR
1	0	1	1	0	1
0	0	0	0	0	0
1	0	1	1	1	1
0	0	0	0	0	0
1	0	1	0	0	0
0	0	1	0	0	0
1	1	1	1	1	1
1	1	1	1	1	1

Picture 3.30 Table Reference for Skill Details in Planning Form

a. In the User Info Form, double click to "cb_name". Be sure, the event was "_Change".

b. Look at the example below. Continue the if statement according to the user, and give each of them with their own image, name, and section.

```
Private Sub cb_name_Change()
    If Me.cb_name.value =
    ThisWorkbook.Sheets("userlogin").Range("E4").value Then
        img_user.Picture = LoadPicture(ThisWorkbook.Path &
        "\images\maintenance\01 Rahmat H.JPG")
        lbl_welcomename.Caption =
        ThisWorkbook.Sheets("userlogin").Range("E4").value
        lbl_eeno.Caption =
        ThisWorkbook.Sheets("userlogin").Range("F4").value
        lbl_section = "General"
        lbl_percen = "100%"
        skl_clm = 1
        Call loop_skillmap
    ElseIf ...
    ...
    End If
End Sub
```

c. Insert new module called “mdl_skillmap”. In the module under “Option Explicit”, type the declarations below.

```
Option Explicit
Dim ws As Worksheet
Dim skl As Integer
Public skl_clm As Integer
```

The script above it used to declare variables. Look, there is the Public variable called “skl_clm” with integer data type.

d. Next, make a procedure called “loop_skillmap”. It will contain the Looping Statement. For the script look the one below.

```
Sub loop_skillmap()
Set ws = ThisWorkbook.Sheets("skillmap")

For skl = 3 To 10
    If ws.Cells(skl, skl_clm) = 1 Then
        Call mm_satu
    ElseIf ws.Cells(skl, skl_clm) = 0 Then
        Call mm_enul
    End If
Next
skl_clm = skl_clm + 1
For skl = 3 To 10
    If ws.Cells(skl, skl_clm) = 1 Then
        Call ts_satu
    ElseIf ws.Cells(skl, skl_clm) = 0 Then
        Call ts_enul
    End If
Next
skl_clm = skl_clm + 1
For skl = 3 To 10
    If ws.Cells(skl, skl_clm) = 1 Then
        Call pr_satu
    ElseIf ws.Cells(skl, skl_clm) = 0 Then
        Call pr_enul
    End If
Next
End Sub
```

e. Next, make some procedure again, it will use Select Case Statement which is the purpose of that statement is same with the IF Statement. For the example, look the script below.

```
Sub mm_satu()
Select Case skl
    Case Is = 3
        frm_userinfo.chb_mm3.Enabled = True
        frm_userinfo.chb_mm3.value = True
    Case Is = 4
        frm_userinfo.chb_mm4.Enabled = True
        frm_userinfo.chb_mm4.value = True
    Case Is = 5
        frm_userinfo.chb_mm5.Enabled = True
        frm_userinfo.chb_mm5.value = True
    Case Is = 6
        frm_userinfo.chb_mm6.Enabled = True
        frm_userinfo.chb_mm6.value = True
    Case Is = 7
        frm_userinfo.chb_mm7.Enabled = True
        frm_userinfo.chb_mm7.value = True
    Case Is = 8
        frm_userinfo.chb_mm8.Enabled = True
        frm_userinfo.chb_mm8.value = True
    Case Is = 9
        frm_userinfo.chb_mm9.Enabled = True
        frm_userinfo.chb_mm9.value = True
    Case Is = 10
        frm_userinfo.chb_mm10.Enabled = True
        frm_userinfo.chb_mm10.value = True
End Select
End Sub
```

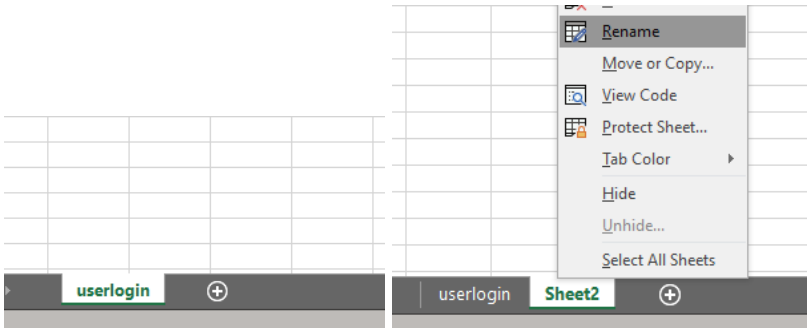
f. Then, continue to make other procedure. Copy the example above and paste it under the “mm_satu” procedure, then change a word “satu” to “enul”. Change all “True” word to “False”.

g. Next, do some copy again, but this time copy the “mm_satu” procedure and “mm_enul” procedure. Paste it, then change the “mm” word to “ts”.

h. Copy again, but this time change the “mm” word to “pr”. And the User Info Form ready to run.

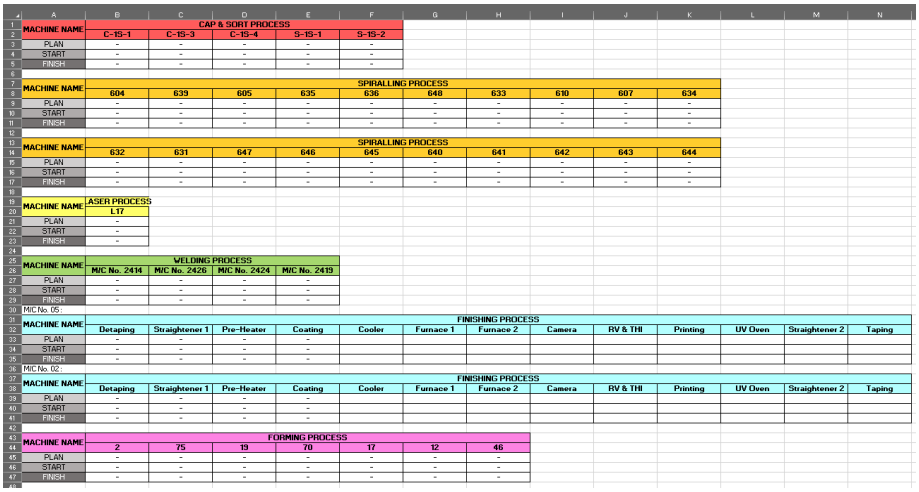
8. Code to Input Date in Planning Form

a. Prepare a Sheet to accommodate the input results. In the bottom of Excel Window, beside “userlogin” Sheet, click the Plus Button. It will insert new Sheet. Right click the new Sheet, and choose rename. Name it “plan”.



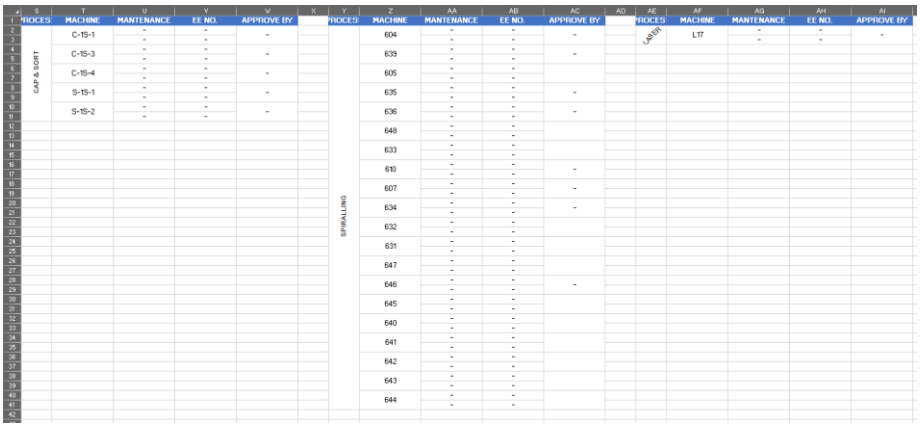
Picture 3.31 Make New Sheet & Rename Sheet

b. Then, make some table look like the picture below.

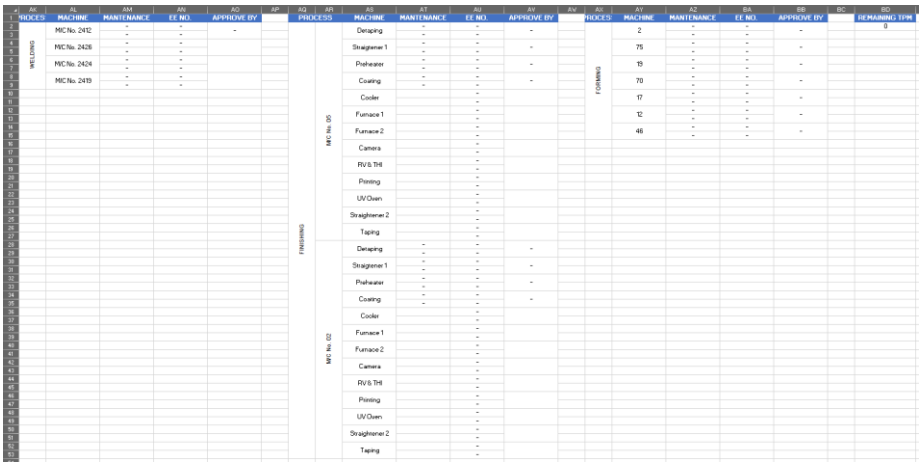


Picture 3.32 Table for Input Date in Sheet “plan”

c. Add some table again beside the previous table.



Picture 3.33 Table for Input Maintenance in Sheet “plan” Picture 1



Picture 3.34 Table for Input Maintenance in Sheet “plan” Picture 2

d. Next, go to the VB Editor to add some code for input Process. Click the frm_planning form. Double click on SUBMIT button. Type the example above.

```
If lbl_machinename.Caption = "C-1S-1" Then
    If txt_iplan = "" Then
    Else: Range("B3").value = Format(txt_iplan.value, "dd mmmm yyyy"):
    End If
    If txt_istart = "" Then
    Else: Range("B4").value = Format(txt_istart.value, "dd mmmm yyyy"):
    End If
    If txt_ifinish = "" Then
    Else: Range("B5").value = Format(txt_ifinish.value, "dd mmmm yyyy"):
    End If
ElseIf ...
    ...
End If
```

The code above will print the text in form to the sheet that has been linked.

e. After finishing the code for input process, then type this code above the IF statement that have been made before.

```
ThisWorkbook.Sheets(“plan”).Activate

If txt_iplan.value = " " Then
    txt_iplan.value = "-": End If
If txt_istart.value = " " Then
    txt_istart.value = "-": End If
If txt_ifinish.value = " " Then
    txt_ifinish.value = "-": End If
```

f. Then, in the bottom of IF statement that have been made. Type the code below.

```
txt_iplan.value = Format(txt_iplan.value, "dd mmmm yyyy")
txt_istart.value = Format(txt_istart.value, "dd mmmm yyyy")
txt_ifinish.value = Format(txt_ifinish.value, "dd mmmm yyyy")
```

g. Double click on frm_planning (the object one), then type the script below.

```
Private Sub UserForm_Activate()
ThisWorkbook.Sheets("plan").Activate
End Sub
```

9. Add Script to Check Condition of Printing Date Input

Make a condition which is if the maintenance has/have not been input, the SUBMIT button in the Planning Form Can't print the date inside the TextBox.

a. Insert new module and named it "mdl_check" (Insert – Module). Make a procedure called checkMTC_toPrintDate.

b. Type a variable with Boolean type data under "Option Explicit".

```
Option Explicit
Public stp as Boolean
```

c. Next, type the IF Statement example like the code below.

```
Sub checkMTC_toPrintDate()
'cap & sort {
    If frm_planning.lbl_machinename.Caption = "C-1S-1" Then
        If Range("U2") = "-" And Range("U3") = "-" Then
        Call Mbox: stp = False:
        Else: stp = True: End If
    ElseIf frm_planning.lbl_machinename.Caption = "C-1S-3" Then
        If Range("U4") = "-" And Range("U5") = "-" Then
        Call Mbox: stp = False:
        Else: stp = True: End If
    ...
'}
'spiralling {
...
End Sub
```

d. Continue the script for other process. Make sure the range in the script is correct.

e. Then, go back to the Planning Form. Double click on SUBMIT button. Then type the code below.

```
Call checkMTC_toPrintDate
If stp = False Then
    frm_planning.txt_iplan.value = ""
    frm_planning.txt_istart.value = ""
    frm_planning.txt_ifinish.value = ""
    Exit Sub
ElseIf stp = True Then
End If
```

10. Activating Date Picker and Giving Script for Reset Button

To provide convenience, use Date Picker for input process. So the user just using the mouse and don't need to type the date. The Date Picker in this case can be downloaded on Trevor Eyre's portofolio.

a. Download the Date Picker in https://trevoreyre.com/portfolio/excel-datepicker/ and export it.
b. Import the Date Picker file (User Form file) to the VB Editor.
c. Go to the Planning Form, double click the SUBMIT button. Under the "Option Explicit", type the code below.

```
Option Explicit
Dim tanggal As Date

Private Sub txt_iplan_DblClick(ByVal Cancel As MSForms.ReturnBoolean)
    tanggal = CalendarForm.GetDate
    txt_iplan.value = Format(tanggal, "dd/mm/yyyy")
End Sub

Private Sub txt_istart_DblClick(ByVal Cancel As MSForms.ReturnBoolean)
    tanggal = CalendarForm.GetDate
    txt_istart.value = Format(tanggal, "dd/mm/yyyy")
End Sub

Private Sub txt_ifinish_DblClick(ByVal Cancel As MSForms.ReturnBoolean)
    tanggal = CalendarForm.GetDate
    txt_ifinish.value = Format(tanggal, "dd/mm/yyyy")
End Sub
```

d. Then you can try to run the Planning Form, and double click on txt_iplan, txt_istart, and txt_ifinish. It will showing the Date Picker form.

Next, make script for the RESET button. This script will use IF statement.

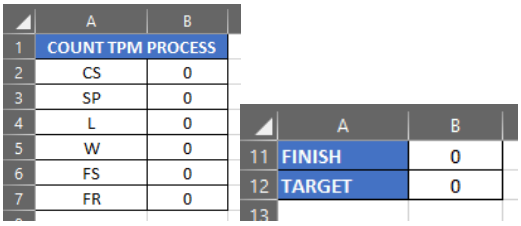
a. Double click the RESET button. Then type the code below.

```
Private Sub btn_reset_Click()
    'cap & sort {
    If lbl_machinename = "C-1S-1" Then
            Range("B3", "B5").value = "-"
            Range("U2", "U3").value = "-"
            Range("W2").value = "-"
            ThisWorkbook.Sheets("email").Range("G2", "H3") = "-"
            frm_main.ica1.Picture = LoadPicture(ThisWorkbook.Path &
            "\images\empty.jpg")
            frm_main.icb1.Picture = LoadPicture(ThisWorkbook.Path &
            "\images\empty.jpg")
            frm_main.nca1.Caption = "-"
            frm_main.ncb1.Caption = "-"
    ElseIf lbl_machinename = "C-1S-3" Then
    ...
    End If
End Sub
```

b. The code above is example from Cap & Sort process. Continue the other process base with the example above.

11. Monthly Target System

a. Insert new Sheet and named it “target”. Make some table like the one below.



	A	B
1	COUNT TPM PROCESS	
2	CS	0
3	SP	0
4	L	0
5	W	0
6	FS	0
7	FR	0

	A	B
11	FINISH	0
12	TARGET	0
13		

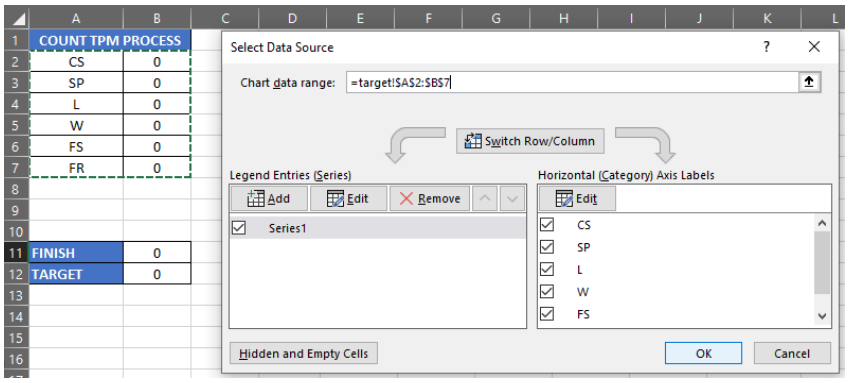
Picture 3.35 Table in Sheet “target”

b. In the cells that contain “0”, each of them fills with a function.

Table 3.3 Table of Formula for Target Sheet

OBJECT	COLOUR
CS	=COUNT(plan!B3:F3)
SP	=COUNT(plan!B9:K9;plan!B15:K15)
L	=COUNT(plan!B21)
W	=COUNT(plan!B27:E27)
FS	=COUNT(plan!B33:N33;plan!B39:N39)
FR	=COUNT(plan!B45:H45)
FINISH	=COUNT(plan!B5:F5;plan!B11:K11;plan!B17:K17;plan!B23;plan!B29:E29;plan!B35:N35;plan!B41:N41;plan!B47:H47)
TARGET	=SUM(B2:B7)

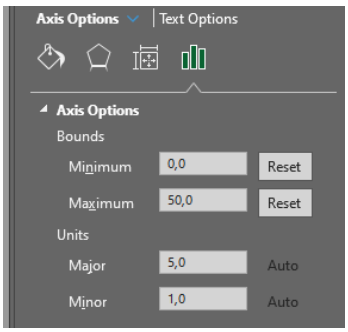
c. After that, insert a Column Chart and a Bar Chart. Right click on column chart and choose Select Data. Block A2 – B7, then click OK.



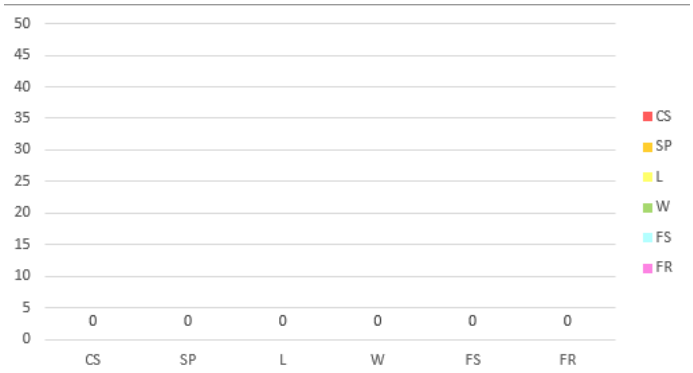
Picture 3.36 Select Data Sourve Column Chart

d. Same with column chart, right click on Bar Chart and choose Select Data. Block A11 – B12, then click OK.

e. Right click on the middle of Column Chart and choose Format Axis. Look right of the window, it will show panel. For the settings look at the picture below. Do the same way to the Bar Chart with the same settings too.

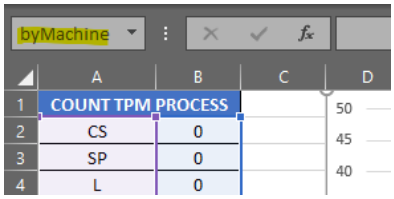


Picture 3.37 Format Exis Setting for The Chart



Picture 3.38 Column Chart Preview

f. Left Click the Column Chart and type “byMachine” on the Name Box and click Enter. Then for the Bar Chart type “byProcess”.



Picture 3.39 Giving Name for Column Chart

12. Indocator System

Use OnTime statement to make the Indicator System. Prepare some tables inside sheet called “email” look like the example picture below and prepare a module called “mdl_ontime”.

PROCESS	MACHINE NAME	MAINTENANCE	EMAIL	NOTE	DATE	STATUS	
CAP & SORT	C-1S-1	RAHMAT HIDAYAT	-	H-7 :	31-Mar-21	SEND	SEND
		-	-	H-3 :	7-Apr-21	SEND	
	C-1S-3	ABDUL ROHMAN	-	H-7 :	1-Apr-21	SEND	SEND
		-	-	H-3 :	8-Apr-21	SEND	
	C-1S-4	WIDI HARTOMO	-	H-7 :	2-Apr-21	SEND	SEND
		-	-	H-3 :	9-Apr-21	SEND	
	S-1S-1	HARI PURNOMO	-	H-7 :	3-Apr-21	SEND	SEND
		-	-	H-3 :	10-Apr-21	SEND	
	S-1S-2	KUKUH JATMIKO	-	H-7 :	4-Apr-21	SEND	SEND
		-	-	H-3 :	11-Apr-21	SEND	

Picture 3.40 Table for Indicator

Cell “C2” is contain a formula, it is “=plan!U2”. And under “C2” is same, the reference is in the Plan Sheet. For the email column, it’s okay to make it empty.

For cell “F2”, the reference is form Plan Sheet too. To make times interval by subtracting by 5 and 12, in cell “F2” fill it with “Plan Date – 12” and cell “F3” fill with “Plan Date – 5”. Do the same way for the column under it.

a. Make some variable declaration inside the module. The code looks like the one below.

```
Option Explicit
Dim date1, date2, date3, time1 As Date
Dim procedure As String
```

b. Type 3 procedure and the code below.

```
Sub StartEmailRemainder()
    Call EmailReminder
    time1 = ThisWorkbook.Sheets("email").Range("AX2")
    Application.OnTime date1 + time1, procedure
    Application.OnTime date2 + time1, procedure
    Application.OnTime date3 + time1, procedure
End Sub

Sub StopEmailReminder()
    Call EmailReminder
    On Error Resume Next
    time1 = ThisWorkbook.Sheets("email").Range("AX2")
    Application.OnTime date1 + time1, procedure, , False
    Application.OnTime date2 + time1, procedure, , False
    Application.OnTime date3 + time1, procedure, , False
End Sub

Sub re_OnTime()
    On Error Resume Next
    time1 = ThisWorkbook.Sheets("email").Range("AX2")
    Application.OnTime date1 + time1, procedure
    Application.OnTime date2 + time1, procedure
    Application.OnTime date3 + time1, procedure
End Sub
```

c. Make new module called “mdl_loadindicator”. Then make some procedure look likes below.

```
Sub setVarE_C1()
    If Date >= ThisWorkbook.Sheets("email").Range("F2") Then
      ThisWorkbook.Sheets("email").Range("G2") = "SEND": End If
    If Date >= ThisWorkbook.Sheets("email").Range("F3") Then
      ThisWorkbook.Sheets("email").Range("G3") = "SEND": End If
    If Date >= ThisWorkbook.Sheets("plan").Range("B3") Then
      ThisWorkbook.Sheets("email").Range("H2") = "SEND": End If
End Sub
```

d. The code above is used to take date value from “email” sheet and “plan” sheet. The value will be use for OnTime statement as time reference. Continue the code with make some new procedure on it.

e. Back to the “mdl_ontime”. Make a procedure called “re_EmailReminder”. It used to load all the Indicator and OnTime statement when workbook open. For the example looks like below.

```
Sub re_EmailReminder()
'cap & sort {
        date1 = ThisWorkbook.Sheets("email").Range("F2")
        date2 = ThisWorkbook.Sheets("email").Range("F3")
        date3 = ThisWorkbook.Sheets("plan").Range("B3")
        procedure = "setVarE_C1"
    Call re_OnTime
        date1 = ThisWorkbook.Sheets("email").Range("F4")
        date2 = ThisWorkbook.Sheets("email").Range("F5")
        date3 = ThisWorkbook.Sheets("plan").Range("C3")
        procedure = "setVarE_C3"
    ...
'}
'spiralling {
...
End Sub
```

f. Continue the code according to the process and procedure that has been made in “mdl_loadindicator”.

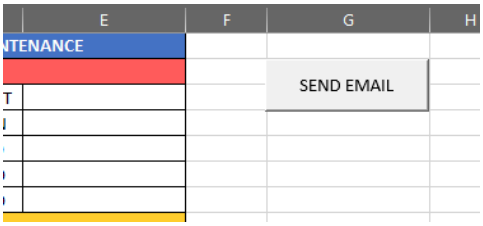
13. Email System

a. Firstly, make new sheet called “planSortir” and then make a table look like the image below. Just create an empty table, ignore the contents.

	A	B	C	D	E	F
1	MACHINE NAME	PLAN DATE	INTERVAL	MAINTENANCE		
2	CAPING & SORTING					
3	C-1S-1	12-Apr-21	-7	RAHMAT HIDAYAT		
4	C-1S-3	13-Apr-21	-6	ABDUL ROHMAN		
5	C-1S-4	14-Apr-21	-5	WIDI HARTOMO		
6	S-1S-1	15-Apr-21	-4	HARI PURNOMO		
7	S-1S-2	16-Apr-21	-3	KUKUH JATMIKO		
8	SPIRALLING					
9	604					
10	639	23-Apr-21	4	RONI FAZLI		
11	605	13-Apr-21	-6	HARI PURNOMO		
12	635					
13	636	16-Apr-21	-3	EDI SANTOSA		
14	648					
15	633					
16	610					
17	607					
18	634					
19	632					
20	631					
21	647					
22	646					
23	645					
24	640					
25	641					
26	642					
27	643					
28	644					
29	LASER					
30	L17	12-Apr-21	-7	KUKUH JATMIKO		
31	WELDING					
32	M/C No. 2412	9-Apr-21	-10	GR NUKE		
33	M/C No. 2426					
34	M/C No. 2424	13-Apr-21	-6	YONDRI FAIZAL		
35	M/C No. 2419	10-Apr-21	-9	RONI FAZLI		

Picture 3.41 The List Table of All TPM Activities

b. Beside the table, insert a button. Go to the Developer Tab, Insert – Button (Form Control). Give caption by right click – edit text. Type “SEND EMAIL”.



Picture 3.42 Make a Button on a Sheet

c. At a considerable distance, make a table just for reference for the main table at step “a”. The table look like below.

	AA	AB	AC	AD	AE	AF
1		PLAN DATE	START DATE	MAINTENANCE		
2						
3		12-Apr-21	13-Apr-21	RAHMAT HIDAYAT	-	
4		13-Apr-21	-	ABDUL ROHMAN	-	
5		14-Apr-21	-	WIDI HARTOMO	-	
6		15-Apr-21	-	HARI PURNOMO	-	
7		16-Apr-21	-	KUKUH JATMIKO	-	
8						
9		-	-	-	-	
10		23-Apr-21	-	RONI FAZLI	-	
11		13-Apr-21	-	HARI PURNOMO	-	
12		-	-	-	-	
13		16-Apr-21	16-Apr-21	EDI SANTOSA	-	
14		-	-	-	-	
15		-	-	-	-	
16		-	-	-	-	
17		-	-	-	-	
18		-	-	-	-	
19		-	-	-	-	
20		-	-	-	-	
21		-	-	-	-	
22		-	-	-	-	
23		-	-	-	-	
24		-	-	-	-	
25		-	-	-	-	
26		-	-	-	-	
27		-	-	-	-	
28		-	-	-	-	

Picture 3.43 Reference Table for The Activities Table List

For the contents, it uses a formula for each cell. In cell “AB3” it contain “=plan!B3”. It means, the cell gets its value form cell “B3” inside the Plan Sheet. And for cell “AC3” and “AD3” the concept it same likes before. Fill it according to the head of the table.

d. On the main table, give a formula for each cell. For the reference is form the table that have been made. For cell “B3” it contain “=IF(AB3="-";"";AB3)” , cell “D3” it contain “=IF(AD3="-";"";AD3)”, and cell “E3” it contain “=IF(AE3="-";"";AE3)”. For the cell “C3” is empty. Copy “B3” until “E3” to the empty cell between them. The reference will automate adapting.

e. Then, go to the VB Editor. Type the code below, under Option Explicit.

```
Option Explicit
Dim i As Integer
Dim ws As Worksheet
Public RowFind, NewRow7, NewRow3 As Integer
```

f. Make a procedure called “if_theCOLOUR”. Then type the code below.

```
Sub if_theCOLOUR()
    Set ws = ThisWorkbook.Sheets("planSortir")

    If ws.Range("AB" & i) <> "-" Then
    ws.Range("C" & i) = ws.Range("AB" & i) - DateValue(Now)
```

```
    ElseIf ws.Range("AB" & i) = "-" Then
    ws.Range("C" & i) = ""
    End If

    If ws.Range("C" & i) > 5 Then
        ws.Range("C" & i).Interior.Color = RGB(198, 239, 206)
        ws.Range("C" & i).Font.Color = RGB(0, 97, 0)
    ElseIf ws.Range("C" & i) < 6 And ws.Range("C" & i) > 0 Then
        ws.Range("C" & i).Interior.Color = RGB(255, 235, 156)
        ws.Range("C" & i).Font.Color = RGB(156, 87, 0)
    ElseIf ws.Range("C" & i) < 1 And ws.Range("AC" &i)="-" Then
        ws.Range("C" & i).Interior.Color = RGB(255, 199, 206)
        ws.Range("C" & i).Font.Color = RGB(156, 0, 6)
    ElseIf ws.Range("C" & i) < 1 And ws.Range("AC" & i) <> "-"
        ws.Range("C" & i).Interior.Color = RGB(198, 239, 206)
        ws.Range("C" & i).Font.Color = RGB(0, 97, 0)
    End If
    If ws.Range("C" & i) = "" Then
        ws.Range("C" & i).Interior.Color = vbWhite
    End If
End Sub
```

g. Then, make a new procedure called “SendEmail_Range”. Type the code below.

```
Sub SendEmail_Range()
    Dim ws As Worksheet
    Set ws = ThisWorkbook.Sheets("datetable")

    Dim emailApplication As Object
    Dim emailItem As Object
    Dim rng As Range

    Set emailApplication = CreateObject("Outlook.Application")
    Set emailItem = emailApplication.CreateItem(0)
    Set rng = ws.Range(Cells(1, 9), Cells(NewRow7, 19))

    emailItem.To = "user1email.gmail.com; user2email.gmail.com"
    emailItem.subject = "ERG(X) 1S TPM Schedule"
    emailItem.HTMLBody = "Tabel Informasi Jadwal TPM yang Akan Segera
Dilaksanakan" & vbCrLf & RangetoHTML(rng)

    emailItem.Send
'    emailItem.Display
End Sub
```

h. Make a Function to make custom function called RangetoHTML, the source code can be copied by visiting this link (https://www.rondebruin.nl/win/s1/outlook/bmail2.htm). The function is used to copy a range of cell to the Outlook apps. Which is can be used to copy the table to Outlook. The function looks like the picture below.

```
Function RangetoHTML(rng As Range) As String
' Changed by Ron de Bruin 28-Oct-2006
' Working in Office 2000-2013
    Dim fso As Object
    Dim ts As Object
    Dim TempFile As String
    Dim TempWB As Workbook
    TempFile = Environ$("temp") & "\" & Format(Now, "dd-mm-yy h-mm-ss")
& ".htm"
```

```
    'Copy the range and create a new workbook to past the data in
    rng.Copy
    Set TempWB = Workbooks.Add(1)
    With TempWB.Sheets(1)
        .Cells(1).PasteSpecial Paste:=8
        .Cells(1).PasteSpecial xlPasteValues, , False, False
        .Cells(1).PasteSpecial xlPasteFormats, , False, False
        .Cells(1).Select
        Application.CutCopyMode = False
        On Error Resume Next
        .DrawingObjects.Visible = True
        .DrawingObjects.Delete
        On Error GoTo 0
    End With
    'Publish the sheet to a htm file
    With TempWB.PublishObjects.Add( _
         SourceType:=xlSourceRange, _
         FileName:=TempFile, _
         Sheet:=TempWB.Sheets(1).Name, _
         Source:=TempWB.Sheets(1).UsedRange.Address, _
         HtmlType:=xlHtmlStatic)
        .Publish (True)
    End With
    'Read all data from the htm file into RangetoHTML
    Set fso = CreateObject("Scripting.FileSystemObject")
    Set ts = fso.GetFile(TempFile).OpenAsTextStream(1, -2)
    RangetoHTML = ts.readall
    ts.Close
    RangetoHTML = Replace(RangetoHTML,"align=center x:publishsource=",_
    "align=left x:publishsource=")
    'Close TempWB
    TempWB.Close savechanges:=False
    'Delete the htm file we used in this function
    Kill TempFile
    Set ts = Nothing
    Set fso = Nothing
    Set TempWB = Nothing
End Function
```

i. Make new procedure called “find”. It used to sort the main table to the new table according to the times intervals.

```
Sub find()
    Dim ws As Worksheet

    Application.ScreenUpdating = False
    'declare ws as planSortir sheet
    Set ws = ThisWorkbook.Sheets("planSortir")
    NewRow7 = 2
    NewRow3 = 2

    'loop start
    For RowFind = 3 To 48
        If ws.Cells(RowFind, 3) <= 12 And ws.Cells(RowFind, 3)
        > 5 Then
            ws.Cells(RowFind, 3).Select
            'copy row to table7
            ws.Range(Cells(RowFind, 1), Cells(RowFind, 5)).Copy
            ws.Cells(NewRow7, 9).PasteSpecial xlPasteValues
            ws.Cells(NewRow7, 9).PasteSpecial xlPasteFormats
            NewRow7 = NewRow7 + 1
        ElseIf ws.Cells(RowFind, 3) <= 5 And ws.Cells(RowFind,
        3) > 0 Then
            ws.Cells(RowFind, 3).Select
            'copy row to table3
            ws.Range(Cells(RowFind, 1), Cells(RowFind, 5)).Copy
            ws.Cells(NewRow3, 15).PasteSpecial xlPasteValues
            ws.Cells(NewRow3, 15).PasteSpecial xlPasteFormats
            NewRow3 = NewRow3 + 1
        End If
```

```
    Next
    ws.Range(Cells(NewRow3, 15), Cells(NewRow3, 19)).Clear
    NewRow3 = NewRow3 + 1

    For RowFind = 3 To 48
        If ws.Cells(RowFind, 3).Interior.Color = RGB(255, 199,
        206) Then
            ws.Cells(RowFind, 3).Select
            'copy row to table3
            ws.Range(Cells(RowFind, 1), Cells(RowFind, 5)).Copy
            ws.Cells(NewRow3, 15).PasteSpecial xlPasteValues
            ws.Cells(NewRow3, 15).PasteSpecial xlPasteFormats
            NewRow3 = NewRow3 + 1
        End If
    Next
    Application.ScreenUpdating = True
End Sub
```

j. After that, make new procedure again called “EmailAttachment”. Look the code below.

```
Sub EmailAttachment()
    For i = 3 To 7
        Call if_theCOLOUR
    Next
    For i = 9 To 28
        Call if_theCOLOUR
    Next
    For i = 30 To 30
        Call if_theCOLOUR
    Next
    For i = 32 To 35
        Call if_theCOLOUR
    Next
    For i = 42 To 48
        Call if_theCOLOUR
    Next

    Call find
    Call SendEmail_Range
End Sub
```

k. Back to the “planSortir” sheet. Right click to “SEND EMAIL” button. Choose “Assign Macro”. Search for macro called “EmailAttachment”.

l. To test it, click the button. The email will be sent with new table.

m. To make it automate, make new procedure like below.

```
Sub WeeklyReminder()
    Application.OnTime Weekday(Date) = vbMonday +
    TimeValue("10:00:00"), EmailAttachment
End Sub
```

14. Display Data on Main Form

a. Go to “mdl_check” module. In bottom of the module, make new procedure called “check_SENDEmail”. For the code, look at the example below.

```
'cap & sort {
    If frm_planning.lbl_machinename.Caption = "C-1S-1" Then
        If Range("B3") <> "-" Then
        frm_planning.txt_iplan.Enabled = False
```

```
        frm_planning.txt_iplan.Text = Format(Range("B3"), "dd
        mmmm yyyy"): End If
    ElseIf frm_planning.lbl_machinename.Caption = "C-1S-3" Then
        If Range("C3") <> "-" Then
        frm_planning.txt_iplan.Enabled = False
        frm_planning.txt_iplan.Text = Format(Range("C3"), "dd
        mmmm yyyy"): End If
    ...
'}
'spiralling {
...
```

b. Continue the code according to the TPM process.

c. Go to the Planning Form. Double click Submit Button. In the bottom of the procedure. Type the code below.

```
If txt_iplan.Enabled = True Then
        Call StartEmailRemainder
ElseIf txt_iplan.Enabled = False Then
End If

Call check_SENDEmail
```

d. Double click Reset Button. In the top of procedure type the code below.

```
Call StopEmailReminder
```

e. Go to the Sheet1. In Developer tab, click insert – Button (Form Control). Drag the button then click OK.

f. In VB Editor, make new module called "mdl_show". Make procedure called "RunTemplate" and type the code below.

```
Sub RunTemplate()
    If ThisWorkbook.Sheets("userlogin").Range("K50") <> "ER"
    Then
        Call re_EmailReminder
        Call WeeklyReminder
    ElseIf ThisWorkbook.Sheets("userlogin").Range("K50") = "ER"
    Then

    End If
    frm_cover.Show
    ThisWorkbook.Sheets("userlogin").Range("K50") = "ER"
End Sub
```

Back to Sheet1. Right click on Button, Edit Text and give caption "Run Template". Right click again and choose Assign Macro. Search for macro called "RunTemplate".

g. Insert new module *Insert – Module*. Named it *mdl_loadimageMain*. Inside the module, make a Sub Procedure.

```
Sub load_imgCS()
'load for c-1s-1 {
    If Range("U2") = "RAHMAT HIDAYAT" Then
        frm_main.ica1.Picture = LoadPicture(ThisWorkbook.Path &
"\images\maintenance\small\01 Rahmat H.JPG")
        frm_main.nca1.Caption = "RAHMAT H"
    ElseIf Range("U2") = "ABDUL ROHMAN" Then
        frm_main.ica1.Picture = LoadPicture(ThisWorkbook.Path &
...
```

Continue the script like the example above. And for information, a procedure has a maximum length limit for a macro. So, if a procedure crosses the limit, it will get an error. If this problem appears, try to divide the procedure. Give a number for the name of the procedure, so it will not show the error (ambiguous name). For the example look the one below.

```
Sub load_imgSP1()
...
End Sub
Sub load_imgSP2()
...
End Sub
```

h. Next, insert new module again and named it *load_textMain*. This module will use to load all text that appear on Main Form. Type the code like the one below.

Sub load_textCS()

```
'load for c-1s-1. {
    frm_main.tcp1.Caption = Format(Range("B3"), "dd mmmm yyyy")
    frm_main.tcs1.Caption = Format(Range("B4"), "dd mmmm yyyy")
    frm_main.tcf1.Caption = Format(Range("B5"), "dd mmmm yyyy")
'}
...
```

15. Display Data on Cover Form

a. Make new module called “mdl_cover”.

b. To display the data on Cover Form, use the Main Form as reference. So, it will look like below.

```
Sub load_dateFORMCover()
    'c1
    frm_cover.tcp1.Caption = frm_main.tcp1.Caption
    frm_cover.tcs1.Caption = frm_main.tcs1.Caption
    frm_cover.tcf1.Caption = frm_main.tcf1.Caption
    ...
```

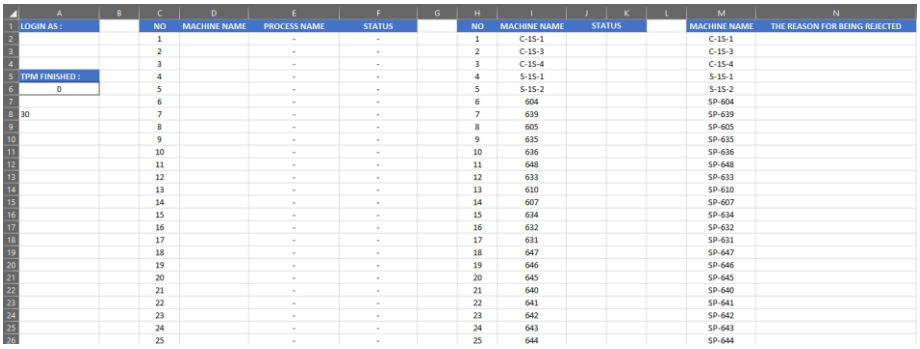
c. Continue the example above. Use that concept to show the images, dates, remaining TPM, and indicators.

16. Raw Summary Report Workbook

a. Make new workbook called “Raw_SummaryReport” and give Macro Enabled extension (*.xlsm).

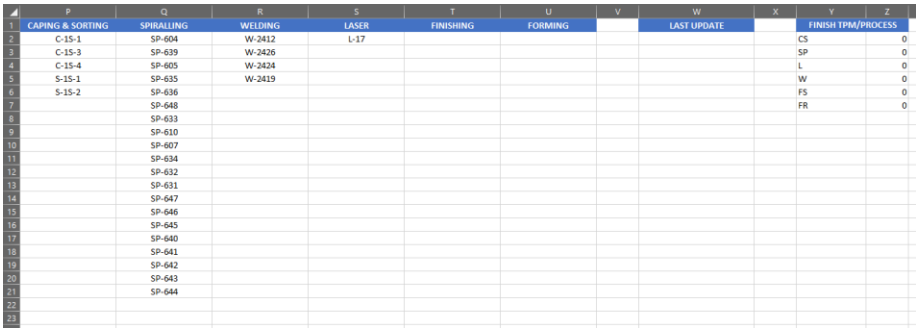
b. Inside the workbook make 5 main sheets. Which is Database sheet, Home Sheet, Rejected Sheet, “C-1S-1” sheet, and Sheet1.

c. Inside Database sheet make some table look like the picture below.



LOGIN AS :		NO	MACHINE NAME	PROCESS NAME	STATUS		NO	MACHINE NAME	STATUS		MACHINE NAME	THE REASON FOR BEING REJECTED
		1		-	-		1	C-1S-1			C-1S-1	
		2		-	-		2	C-1S-3			C-1S-3	
		3		-	-		3	C-1S-4			C-1S-4	
TPM FINISHED :		4		-	-		4	S-1S-1			S-1S-1	
0		5		-	-		5	S-1S-2			S-1S-2	
		6		-	-		6	604			SP-604	
30		7		-	-		7	639			SP-639	
		8		-	-		8	605			SP-605	
		9		-	-		9	635			SP-635	
		10		-	-		10	636			SP-636	
		11		-	-		11	648			SP-648	
		12		-	-		12	633			SP-633	
		13		-	-		13	610			SP-610	
		14		-	-		14	607			SP-607	
		15		-	-		15	634			SP-634	
		16		-	-		16	632			SP-632	
		17		-	-		17	631			SP-631	
		18		-	-		18	647			SP-647	
		19		-	-		19	646			SP-646	
		20		-	-		20	645			SP-645	
		21		-	-		21	640			SP-640	
		22		-	-		22	641			SP-641	
		23		-	-		23	642			SP-642	
		24		-	-		24	643			SP-643	
		25		-	-		25	644			SP-644	

Picture 3.44 Database Sheet Table Picture 1



CAPING & SORTING	SPIRALLING	WELDING	LASER	FINISHING	FORMING		LAST UPDATE		FINISH TPM/PROCESS	
C-1S-1	SP-604	W-2412	L-17						CS	0
C-1S-3	SP-639	W-2426							SP	0
C-1S-4	SP-605	W-2424							L	0
S-1S-1	SP-635	W-2419							W	0
S-1S-2	SP-636								FS	0
	SP-648								FR	0
	SP-633									
	SP-610									
	SP-607									
	SP-634									
	SP-632									
	SP-631									
	SP-647									
	SP-646									
	SP-645									
	SP-640									
	SP-641									
	SP-642									
	SP-643									
	SP-644									

Picture 3.45 Database Sheet Table Picture 2

d. In Machine Name column, fill it according to the all machines name. Except, in column “D”, don’t fill it with any value.

e. In cell column “E” and cell column “F” are use formulas. Look the example below.

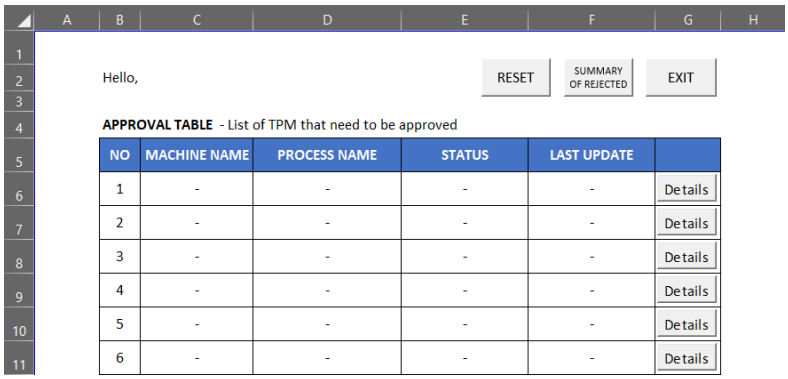
1) Inside cell “E2”

```
=IF(D2<>"";"CAPING & SORTING";"-")
```

2) Inside cell “F2”

```
=IF(AND(D2<>"";J2="F");"FINISHED";IF(AND(D2<>"";J2="R";K2="
U");"REVISION";IF(AND(D2<>"";J2="R");"REJECTED";IF(D2<>"";"
NEED APPROVAL";"-"))))
```

f. In the Home Sheet, make the table look like the picture below.



Hello,

RESET | SUMMARY OF REJECTED | EXIT

APPROVAL TABLE - List of TPM that need to be approved

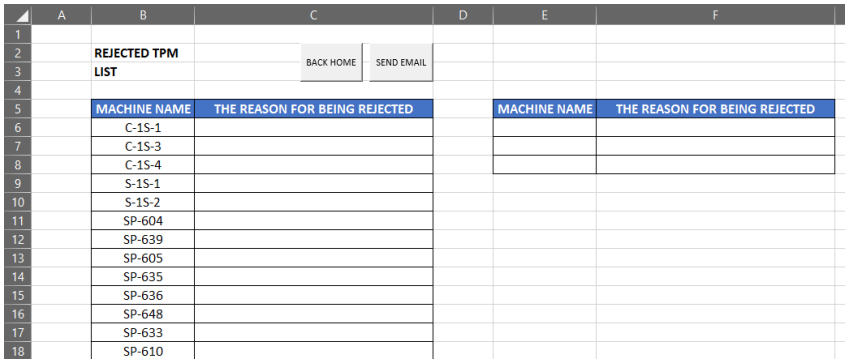
NO	MACHINE NAME	PROCESS NAME	STATUS	LAST UPDATE	
1	-	-	-	-	Details
2	-	-	-	-	Details
3	-	-	-	-	Details
4	-	-	-	-	Details
5	-	-	-	-	Details
6	-	-	-	-	Details

Picture 3.46 Home Sheet Table

g. In each cell, make a formula to get value reference from Database Sheet. Example like below (cell "C6").

```
=IF(Database!D3="";"-";Database!D3)
```

h. In Rejected Sheet, fill with tables inside the picture below.



REJECTED TPM LIST

BACK HOME | SEND EMAIL

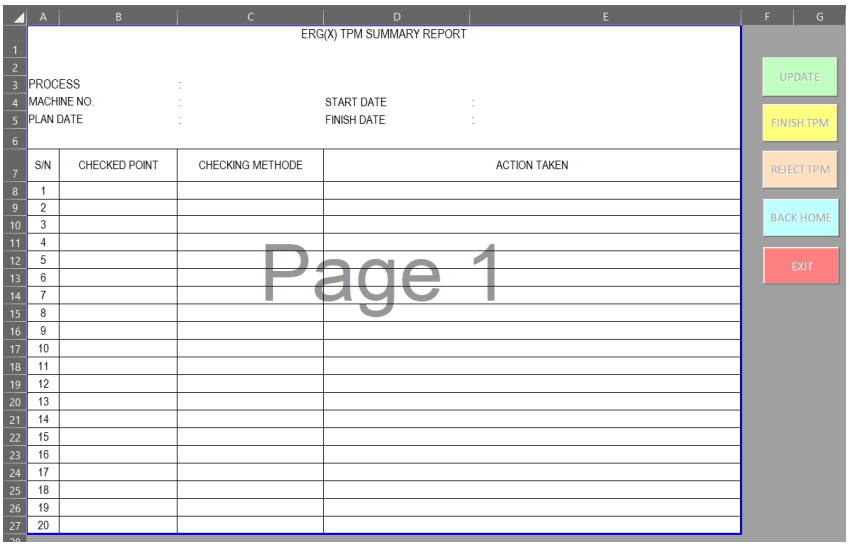
MACHINE NAME	THE REASON FOR BEING REJECTED
C-1S-1	
C-1S-3	
C-1S-4	
S-1S-1	
S-1S-2	
SP-604	
SP-639	
SP-605	
SP-635	
SP-636	
SP-648	
SP-633	
SP-610	

MACHINE NAME	THE REASON FOR BEING REJECTED

Picture 3.47 Rejected Sheet Table

i. Fill the Machine Name column according to the all machines name.

j. Inside the "C-1S-1" Sheet. Fill it with machine check sheet. Look like the picture below.



Picture 3.48 Machine Sheet Table

k. Also make the button too in each sheet. Use the ActiveX Control. For the Sheet1 don't fill it with anything.

l. Go to the VB Editor. Make new module. And make some procedures like the picture below.

```
Option Explicit
Public wb As Workbook
Public ws, wsD As Worksheet
Public mName, folderName As String
Public rangeAddr, folderAddr As Range

Sub updateButton()
    'declare variable
    Set wsD = ThisWorkbook.Sheets("Database")
    mName = ActiveSheet.name
    Set rangeAddr = wsD.Range("M2", "M31").Find(mName)

    wsD.Cells(rangeAddr.Row, 4) = mName
    If wsD.Cells(rangeAddr.Row, 10) = "R" Then
        wsD.Cells(rangeAddr.Row, 11) = "U"
    End If
    ThisWorkbook.Save
End Sub

Sub finishButton()
    Dim pesan As String
    pesan = MsgBox("Are you sure want to finish this TPM?", vbQuestion & vbYesNo)
    If pesan = vbYes Then
    ElseIf pesan = vbNo Then
        Exit Sub
    End If

    'declare variable
    Set wsD = ThisWorkbook.Sheets("Database")
    mName = ActiveSheet.name
    Set rangeAddr = wsD.Range("M2", "M31").Find(mName)
    Set folderAddr = wsD.Range("P2", "U21").Find(mName)

    Call checkSheetNameSave
    Dim wbN As Workbook
    Set wbN = Workbooks.Add

    ThisWorkbook.ActiveSheet.Copy before:=wbN.Worksheets(1)
    wbN.SaveAs (ThisWorkbook.Path & folderName)
    wbN.Close

    wsD.Cells(rangeAddr.Row, 10) = "F"
'    ThisWorkbook.ActiveSheet.Range("B8", "E27").ClearContents
    wsD.Cells(rangeAddr.Row, 14).ClearContents
End Sub
Sub rejectButton()
    Set wsD = ThisWorkbook.Sheets("Database")
    mName = ActiveSheet.name
    Set rangeAddr = wsD.Range("M2", "M31").Find(mName)

    wsD.Cells(rangeAddr.Row, 10) = "R"
    wsD.Cells(rangeAddr.Row, 14) = frm_rejected.TextBox1.Value
End Sub

Sub backButton()
    ThisWorkbook.ActiveSheet.Visible = False
    ThisWorkbook.Sheets("Home").Activate
    ThisWorkbook.Sheets("Home").Visible = True
End Sub

Sub exitButton()
    Dim pesan As String
    pesan = MsgBox("Are you sure want to exit?", vbQuestion & vbYesNo)
    If pesan = vbYes Then
    ElseIf pesan = vbNo Then
        Exit Sub
    End If

    'hide active sheet, clear A2, A3 database sheet & save local workbook
    Call hideAllSheet
    ThisWorkbook.Sheets("Database").Range("A2", "A3").ClearContents
    ThisWorkbook.Save

    'run the macro in neighbor workbook and close local workbook
    Workbooks.Open (ThisWorkbook.Path & "\TPM Schedule-1S_v1.0.xlsm")
    ThisWorkbook.Sheets("Database").Range("A6").Copy
    Workbooks("TPM Schedule-1S_v1.0.xlsm").Sheets("target").Range("B11").PasteSpecial xlPasteValues
    ThisWorkbook.Sheets("Database").Range("Z2:Z7").Copy
    Workbooks("TPM Schedule-1S_v1.0.xlsm").Sheets("target").Range("B17").PasteSpecial xlPasteValues
    Application.Run "'TPM Schedule-1S_v1.0.xlsm'!Run_LoginForm"
    ThisWorkbook.Close
End Sub
```

Picture 3.49 Code Inside Raw_SummaryReport Picture 1

```
Sub resetButton()
    Dim pesan As String
    Set wsD = ThisWorkbook.Sheets("Database")
    If wsD.Range("A8") = 30 Then
        Exit Sub
    End If

    pesan = MsgBox("Are you sure want to reset all History?", vbQuestion & vbYesNo)
    If pesan = vbYes Then
    ElseIf pesan = vbNo Then
        Exit Sub
    End If

    Set wb = ThisWorkbook
    wb.Sheets("BackupDatabase").Range("C2") = DateValue(Now)
    wb.Sheets("Database").Range("C1", "N31").Copy
    wb.Sheets("BackupDatabase").Range("B3").PasteSpecial xlPasteAll

    wsD.Range("D2", "D31").ClearContents
    wsD.Range("J2", "K31").ClearContents
    wsD.Range("N2", "N31").ClearContents
    wsD.Range("W2", "W31").ClearContents
    Call resetSheet
End Sub
```

```
Sub openRejectedTPM()
    Set ws = ThisWorkbook.Sheets("Rejected")
    ws.Visible = True
    ws.Activate
End Sub
```

Picture 3.50 Code Inside Raw_SummaryReport Picture 2

m. And add some add on procedure above it.

```
Sub tool()
    Set wsD = ThisWorkbook.Sheets("Database")
    mName = ActiveSheet.name
    Set rangeAddr = wsD.Range("M2", "M31").Find(mName)
End Sub
```

```
Sub disable()
    'declare worksheet and workbook
    Dim ws As Worksheet
    Dim wb As Workbook
    Set ws = ThisWorkbook.Sheets("Database")
    Set wb = ThisWorkbook

    'if statement to disable/enable CmdButton2
    If ThisWorkbook.Sheets("Database").Range("A2") = "Maintenance" Then
        Sheet5.Shapes.Item("CommandButton2").ControlFormat.Enabled = False
    ElseIf ThisWorkbook.Sheets("Database").Range("A2") = "Leader" Then
        Sheet5.Shapes.Item("CommandButton2").ControlFormat.Enabled = True
    End If
End Sub
```

```
Sub hideAllSheet()
    Dim sh As Object
    Application.ScreenUpdating = False
    For Each sh In Sheets
        If sh.name <> "Sheet1" Then
            sh.Visible = False
        End If
    Next sh
    Application.ScreenUpdating = True
End Sub
```

```
Sub checkSheetNameSave()
    Select Case folderAddr.Column
        Case Is = 16
            Call lokasiFile
        Case Is = 17
            Call lokasiFile
        Case Is = 18
            Call lokasiFile
        Case Is = 19
            Call lokasiFile
        Case Is = 20
            Call lokasiFile
        Case Is = 21
            Call lokasiFile
    End Select
End Sub
```

Picture 3.51 Add On Procedure in Raw_SummaryReport Picture 1

```
Sub lokasiFile()
    Dim tanggal, subFolder As String
    tanggal = Format(Date, "d-mmm-yy")
    subFolder = wsD.Cells(1, folderAddr.Column)
    folderName = "\SummaryReport\" & subFolder & "\" & tanggal & " (" & mName & ")" & ".xlsx"
End Sub

Sub copyFinishedTPM()
'    Set wb = Workbooks.Open(ThisWorkbook.Path & "\TPM Schedule-1S_v1.0.xlsm")
    ThisWorkbook.Sheets("Database").Range("A6").Copy Workbooks("TPM Schedule-1S_v1.0.xlsm").Sheets("target").Range("B11")
End Sub

Sub sortir()
    Set ws = ThisWorkbook.Sheets("Rejected")
    Dim cRow, nextRow As Integer

    nextRow = 6
    For cRow = 6 To 35
        If ws.Cells(cRow, 3) <> "" Then
            ws.Range(Cells(cRow, 2), Cells(cRow, 3)).Copy
            ws.Cells(nextRow, 5).PasteSpecial xlPasteValues
            ws.Cells(nextRow, 5).PasteSpecial xlPasteFormats
            nextRow = nextRow + 1
        End If
    Next
'    call sendEmail
End Sub

Sub sendEmail()
    Dim ws As Worksheet
    Dim rowTo As Integer
    Set ws = ThisWorkbook.Sheets("Rejected")

'    On Error Resume Next
    Dim emailApplication As Object
    Dim emailItem As Object
    Dim rng As Range

'    NewRow7 = NewRow7 - 1
'    NewRow3 = NewRow3 - 1
    Set emailApplication = CreateObject("Outlook.Application")
    Set emailItem = emailApplication.CreateItem(0)
    rowTo = nextRow - 1
    Set rng = ws.Range(Cells(5, 5), Cells(rowTo, 6))

    emailItem.To = "nechya.einsfrianas@sg.panasonic.com"
    emailItem.Subject = "ERG(X) 1S TPM Schedule | Rejected Machine List"
    emailItem.HTMLBody = "Tabel Rejected TPM" & vbCrLf & RangetoHTML(rng)

'    emailItem.Send
    emailItem.Display
End Sub

Sub resetSheet()
    Dim wsCount As Integer
    Dim i As Integer
    Dim sheetName As String
    Set wsD = ThisWorkbook.Sheets("Database")

    For i = 2 To 31
        sheetName = wsD.Cells(i, 13)
'        MsgBox sheetName
        On Error Resume Next
        ThisWorkbook.Worksheets(sheetName).Range("B8", "E27").ClearContents
    Next
End Sub
```

Picture 3.52 Add On Procedure in Raw_SummaryReport Picture 2

n. Make new module again, and fill it with the codes below.

```
Option Explicit

Sub loadAwal()
    Dim ws As Worksheet
    Dim wb As Workbook
    Set ws = ThisWorkbook.Sheets("Database")
    Set wb = ThisWorkbook
    Dim addr As Range

    Application.ScreenUpdating = False
    On Error Resume Next
    If ThisWorkbook.Sheets("Database").Range("A2") = "Maintenance" Then
        Set addr = ws.Range("M2", "M31").Find(ActiveSheet.name)
        If ws.Cells(addr.Row, 10) = "F" Then
            ActiveSheet.Shapes.Item("btn_update").ControlFormat.Enabled = False
        Else
            ActiveSheet.Shapes.Item("btn_update").ControlFormat.Enabled = True
        End If
        ActiveSheet.Shapes.Item("btn_finish").ControlFormat.Enabled = False
        ActiveSheet.Shapes.Item("btn_back").ControlFormat.Enabled = False
        ActiveSheet.Shapes.Item("btn_reject").ControlFormat.Enabled = False
        ActiveSheet.Shapes.Item("btn_exit").ControlFormat.Enabled = True
    ElseIf ThisWorkbook.Sheets("Database").Range("A2") = "Leader" Then
        ActiveSheet.Shapes.Item("btn_update").ControlFormat.Enabled = False
        ActiveSheet.Shapes.Item("btn_finish").ControlFormat.Enabled = True
        ActiveSheet.Shapes.Item("btn_back").ControlFormat.Enabled = True
        ActiveSheet.Shapes.Item("btn_reject").ControlFormat.Enabled = True
        ActiveSheet.Shapes.Item("btn_exit").ControlFormat.Enabled = True
    Else
        ActiveSheet.Shapes.Item("btn_update").ControlFormat.Enabled = True
        ActiveSheet.Shapes.Item("btn_finish").ControlFormat.Enabled = False
        ActiveSheet.Shapes.Item("btn_back").ControlFormat.Enabled = False
        ActiveSheet.Shapes.Item("btn_reject").ControlFormat.Enabled = False
    End If
    Application.ScreenUpdating = True
End Sub
```

```
Sub ontime()
    Application.ontime TimeValue(Now) + TimeValue("00:00:02"), "loadAwal"
End Sub
```

```
Function RangetoHTML(rng As Range)
' Changed by Ron de Bruin 28-Oct-2006
' Working in Office 2000-2016
    Dim fso As Object
    Dim ts As Object
    Dim TempFile As String
    Dim TempWB As Workbook

    TempFile = Environ$("temp") & "\" & Format(Now, "dd-mm-yy h-mm-ss") & ".htm"

    'Copy the range and create a new workbook to past the data in
    rng.Copy
    Set TempWB = Workbooks.Add(1)
    With TempWB.Sheets(1)
        .Cells(1).PasteSpecial Paste:=8
        .Cells(1).PasteSpecial xlPasteValues, , False, False
        .Cells(1).PasteSpecial xlPasteFormats, , False, False
        .Cells(1).Select
        Application.CutCopyMode = False
        On Error Resume Next
        .DrawingObjects.Visible = True
        .DrawingObjects.Delete
        On Error GoTo 0
    End With

    'Publish the sheet to a htm file
    With TempWB.PublishObjects.Add( _
         SourceType:=xlSourceRange, _
         Filename:=TempFile, _
         Sheet:=TempWB.Sheets(1).name, _
         Source:=TempWB.Sheets(1).UsedRange.Address, _
         HtmlType:=xlHtmlStatic)
        .Publish (True)
    End With

    'Read all data from the htm file into RangetoHTML
    Set fso = CreateObject("Scripting.FileSystemObject")
    Set ts = fso.GetFile(TempFile).OpenAsTextStream(1, -2)
    RangetoHTML = ts.readall
    ts.Close
    RangetoHTML = Replace(RangetoHTML, "align=center x:publishsource=", _
                          "align=left x:publishsource=")

    'Close TempWB
    TempWB.Close savechanges:=False

    'Delete the htm file we used in this function
    Kill TempFile

    Set ts = Nothing
    Set fso = Nothing
    Set TempWB = Nothing
End Function
```

Picture 3.53 Code in Raw_SummaryReport Module 2

o. Give a reference procedure in each button.

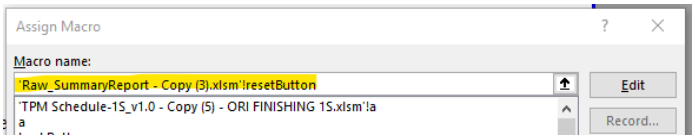
1) Exit Button in each Sheet

```
Private Sub btn_exit_Click()
    Call exitButton
End Sub
```

2) Back Home Button in each Sheet

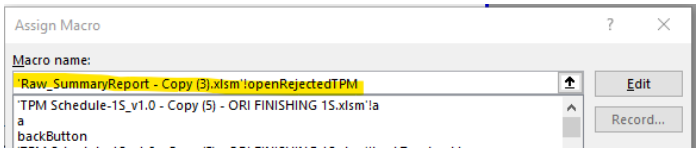
```
Private Sub btn_back_Click()
    Call backButton
End Sub
```

3) Reset Button in Home Sheet



Picture 3.54 Reset Button Reference

4) Summary of Rejected TPM Button in Home Sheet



Picture 3.55 Rejected TPM Button Reference

5) Update, Finish, and Reject Button in "2 – Detaping" Sheet

```
Option Explicit

Private Sub btn_update_Click()
    Call updateButton
    Set wsD = ThisWorkbook.Sheets("Database")
    mName = ActiveSheet.name
    Set rangeAddr = wsD.Range("M2", "M31").Find(mName)

    wsD.Cells(rangeAddr.Row, 23) = DateValue(Now) + TimeValue(Now)
End Sub

Private Sub btn_finish_Click()
    Call tool
    If wsD.Cells(rangeAddr.Row, 6) = "FINISHED" Then
        Exit Sub
    ElseIf wsD.Cells(rangeAddr.Row, 4) = "" Then
        Exit Sub
    ElseIf wsD.Cells(rangeAddr.Row, 6) <> "FINISHED" Then
        Call finishButton
    End If

    Set wsD = ThisWorkbook.Sheets("Database")
    mName = ActiveSheet.name
    Set rangeAddr = wsD.Range("M2", "M31").Find(mName)

    wsD.Cells(rangeAddr.Row, 23) = DateValue(Now) + TimeValue(Now)
End Sub

Private Sub btn_reject_Click()
    Call tool
'    If wsD.Cells(rangeAddr.Row, 6) <> "NEED APPROVAL" Or wsD.Cells(rangeAddr.Row, 6) <> "REJECTED" Then
'        Exit Sub
    If wsD.Cells(rangeAddr.Row, 6) = "NEED APPROVAL" Or wsD.Cells(rangeAddr.Row, 6) = "REJECTED" Then
        Dim pesan As String
        pesan = MsgBox("Are you sure want to reject this TPM?", vbQuestion & vbYesNo)
        If pesan = vbYes Then
        ElseIf pesan = vbNo Then
            Exit Sub
        End If
        frm_rejected.Show
    End If
End Sub
```

Picture 3.56 Update, Finish, and Reject Button Reference

17. Code for Button Summary Report on Main Form

a. Make a procedure inside a module. Named the procedure with "open_summaryReport". Then type the code below.

```
Sub open_summaryReport()
    Dim rangeAddr As Range
    Dim asName As String
    Dim ws, wsOW As Worksheet
    ActiveSheet.Visible = True
    ActiveWorkbook.Sheets("Database").Range("A2").value = frm_main.lbl_loginas
    ActiveWorkbook.Sheets("Database").Range("A3").value = frm_main.lbl_welcomename

    asName = ActiveWorkbook.ActiveSheet.Name
    Set ws = ThisWorkbook.Sheets("plan")
    Set rangeAddr = ws.Range("B6", "N48").find(asName)
    Set wsOW = ActiveWorkbook.ActiveSheet
    If wsOW.Range("E5") = ": " & ws.Cells(rangeAddr.Row - 1, rangeAddr.Column) Then
    Else
        wsOW.Range("C3") = ": " & ws.Cells(rangeAddr.Row, 1)
        wsOW.Range("C4") = ": " & ws.Cells(rangeAddr.Row - 4, rangeAddr.Column)
        wsOW.Range("C5") = ": " & ws.Cells(rangeAddr.Row - 3, rangeAddr.Column)
        wsOW.Range("E4") = ": " & ws.Cells(rangeAddr.Row - 2, rangeAddr.Column)
        wsOW.Range("E5") = ": " & ws.Cells(rangeAddr.Row - 1, rangeAddr.Column)
    End If
    ThisWorkbook.Sheets("userlogin").Range("K47") = frm_main.lbl_welcomename.Caption
    ThisWorkbook.Sheets("userlogin").Range("K48") = frm_main.lbl_loginas.Caption
    Unload frm_main
End Sub
```

Picture 3.57 Code for open_summaryReport Procedure

b. Go to the Main Form. Double click the Summary Report button. Then type the code inside the procedure.

```
Private Sub btn_c1_Click()
    Workbooks.Open ThisWorkbook.Path &"\Raw_SummaryReport.xlsm"
    ActiveWorkbook.Sheets("FR-46").Activate
    Call open_summaryReport
End Sub
```

c. Then try to click it. It will open the workbook and activate the sheet according the machine. Continue the code according to the machine unit.

d. Last, go to Raw_SummaryReport Workbook then open VB Editor and go to Module 1. Type the code below. End run it.

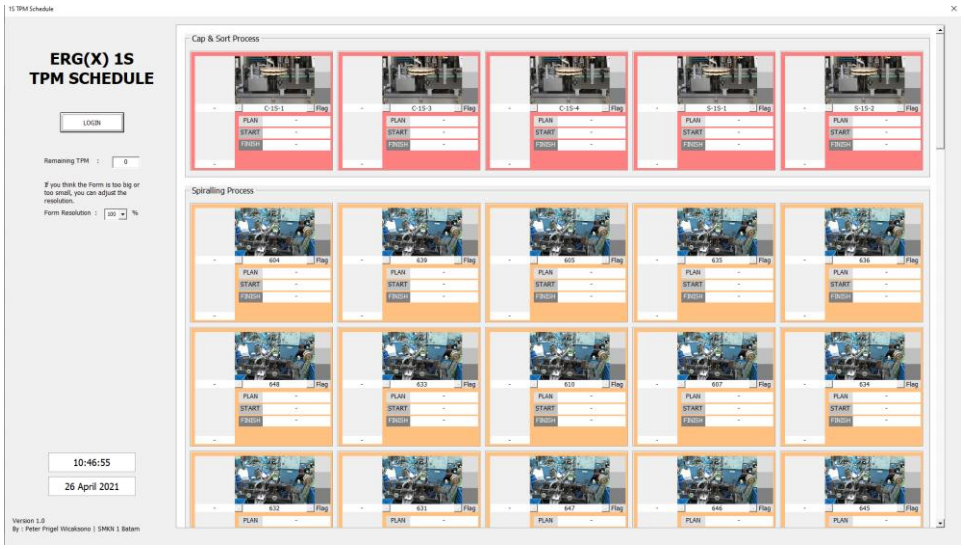
```
Sub copySheet()
Dim i As Integer
Dim name As String
Dim before As String

For i = 3 To 31
    name = ThisWorkbook.Sheets("Database").Cells(i, 13)
    before = ThisWorkbook.Sheets("Database").Cells(i - 1, 13)
    Worksheets("C-1S-1").Copy before:=ThisWorkbook.Sheets(before)
    ActiveSheet.name = name
    MsgBox name
Next
End Sub
```

Picture 3.58 Code to Copy "C-1S-1" Sheet

E. System Result and Testing

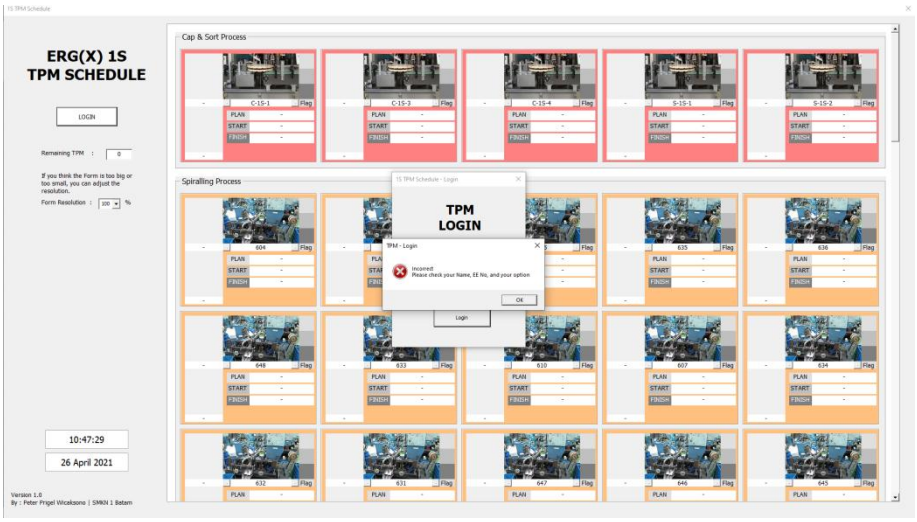
1. Cover Form



Picture 3.59 Cover Form Running Testing

2. Login Form

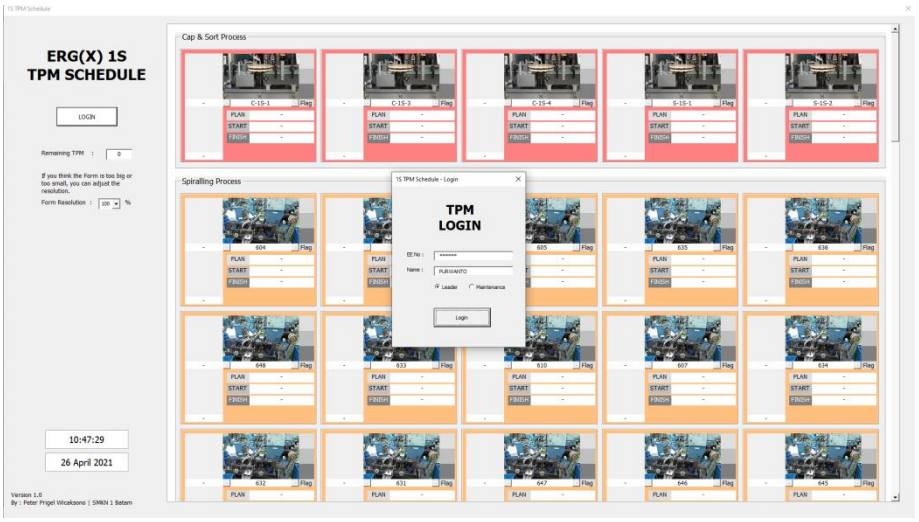
   a. If False



Picture 3.60 Login Form – If Login False

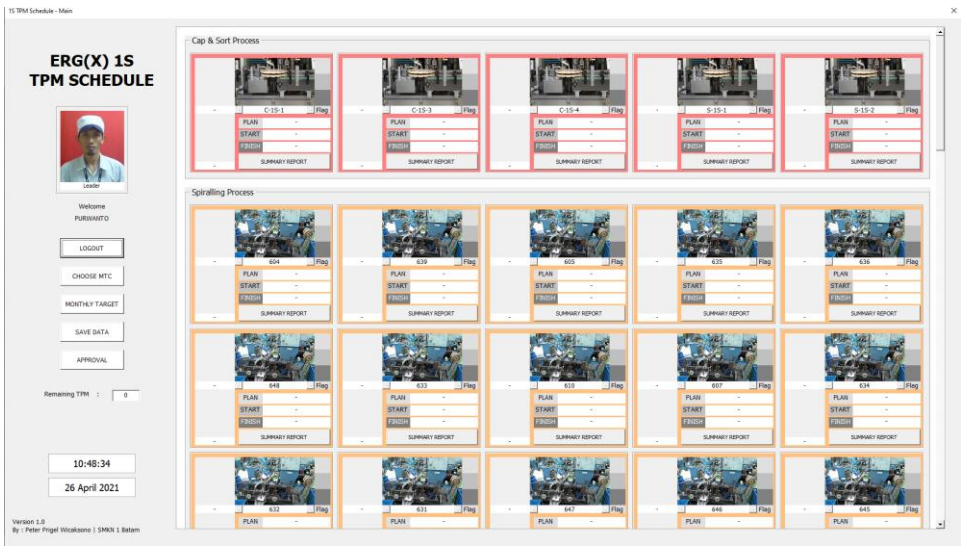
b. If True

If True, Main Form will show.



Picture 3.61 Login Form – If Login True

3. Main Form



Picture 3.62 Main Form

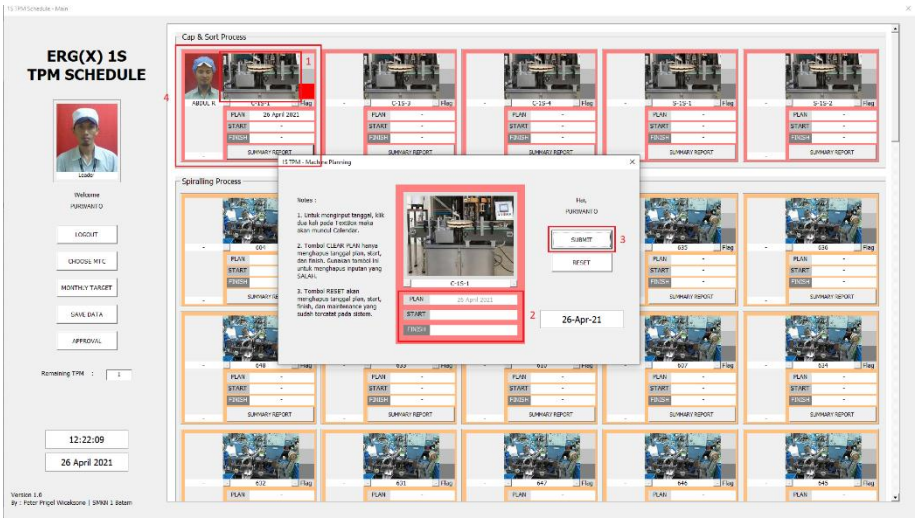
4. User Info Form

   Choosing MTC on User Info Form.

Picture 3.63 User Info Form

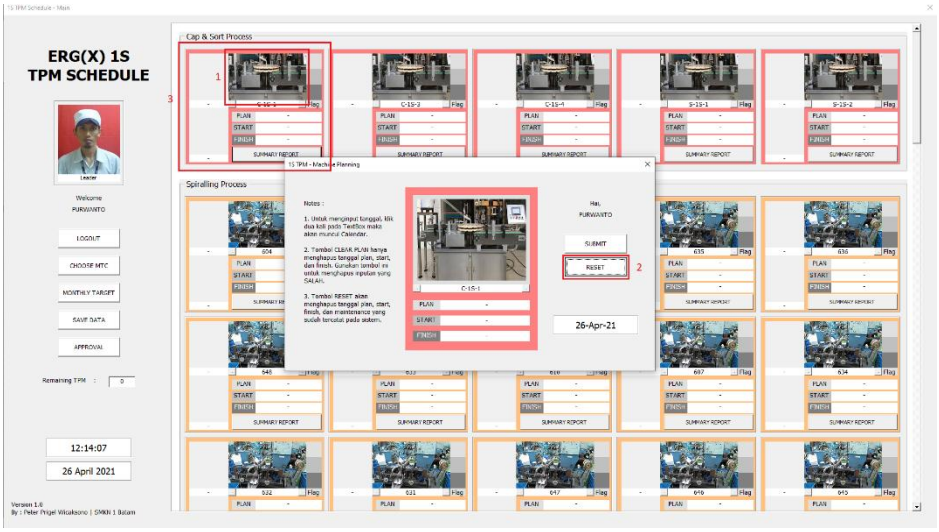
5. Planning Form

   a. Submit Button Click



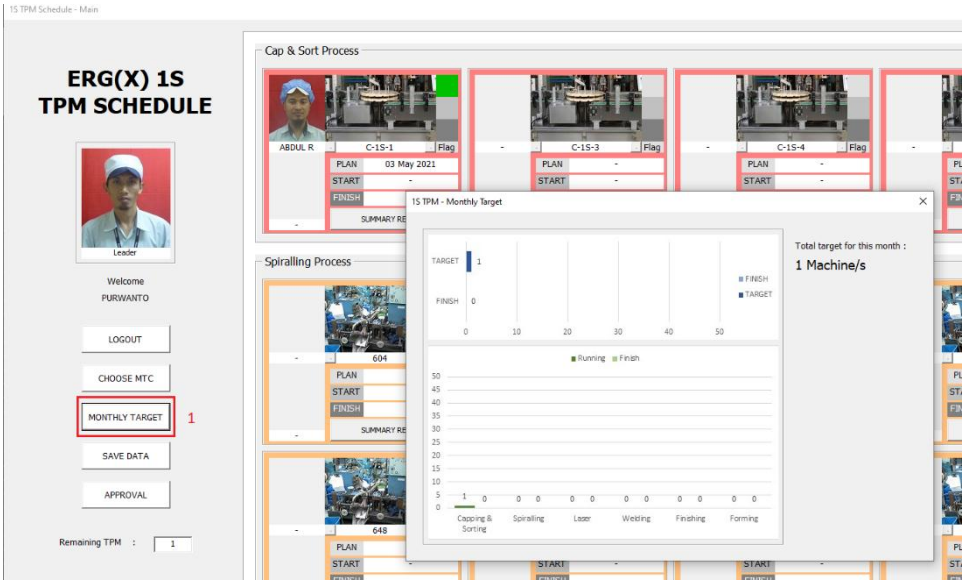
Picture 3.64 Planning Form – Submit

b. Reset Button Click



Picture 3.65 Planning Form - Reset

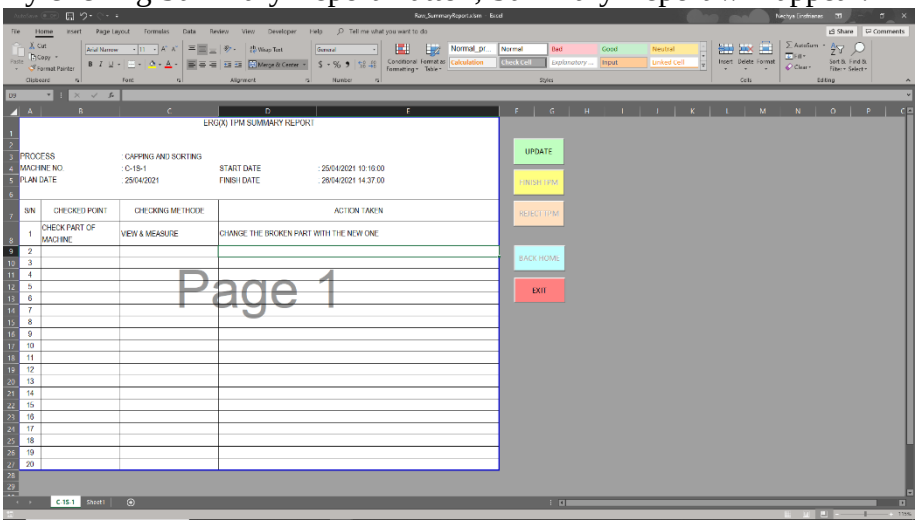
6. Monthly Target Form



Picture 3.66 Monthly Target Form

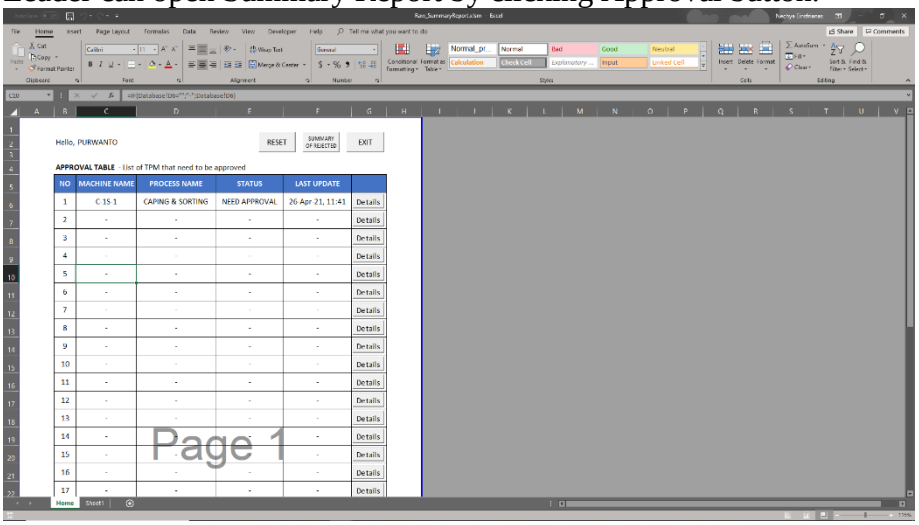
7. Summary Report

   a. Login as Maintenance – Update Machine Summary Report
   By clicking Summary Report Button, Summary Report will appear.



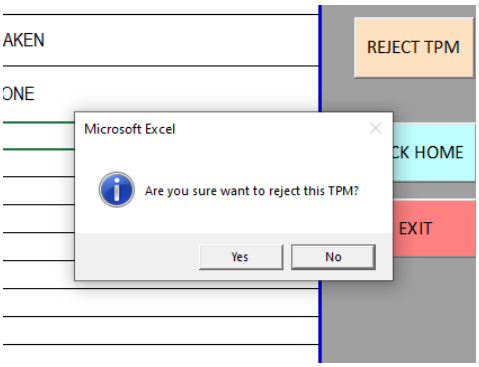
Picture 3.67 Summary Report as Maintenance

   b. Login as Leader – Home Sheet
   Leader can open Summary Report by clicking Approval button.

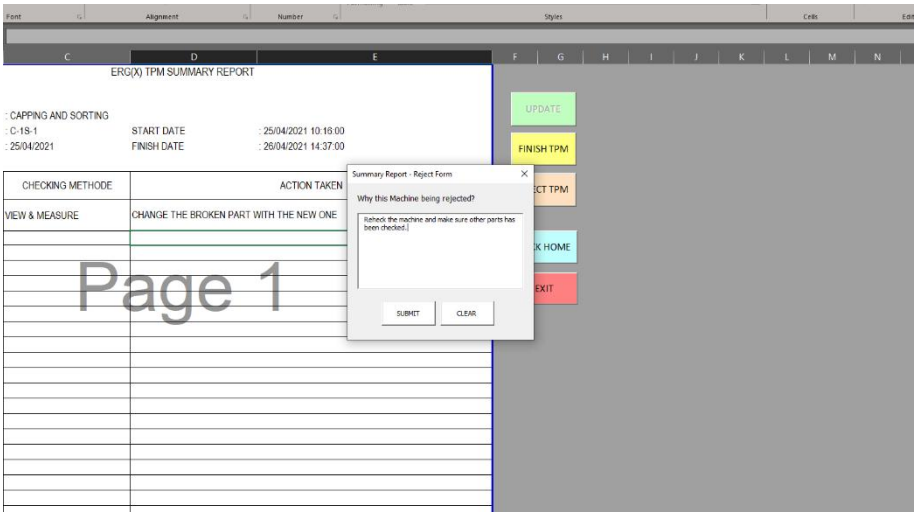


Picture 3.68 Summary Report as Leader

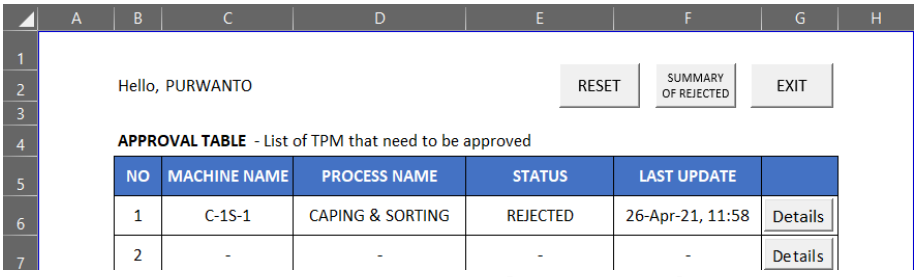
c. Login as Leader – Reject The TPM
By clicking Details button in Home Sheet, Machine Check Sheet will active.



Picture 3.69 Reject Pop-up

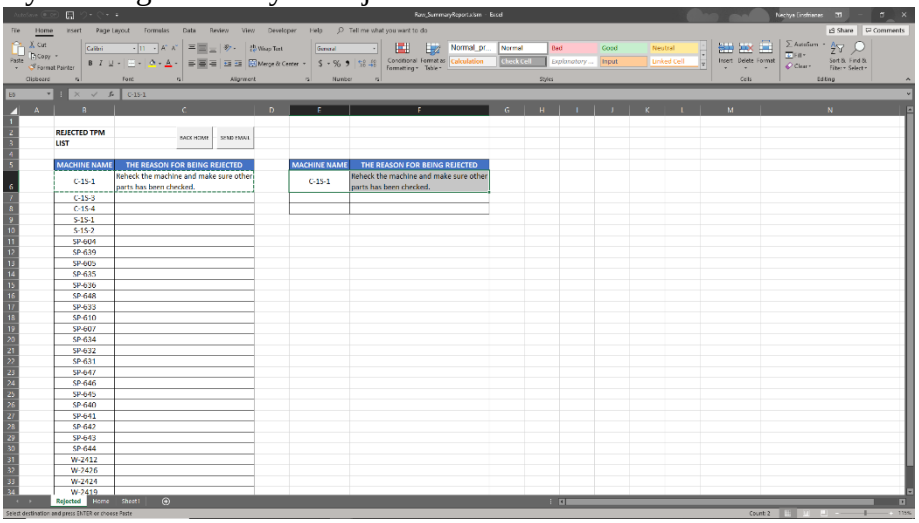


Picture 3.70 Reject Form



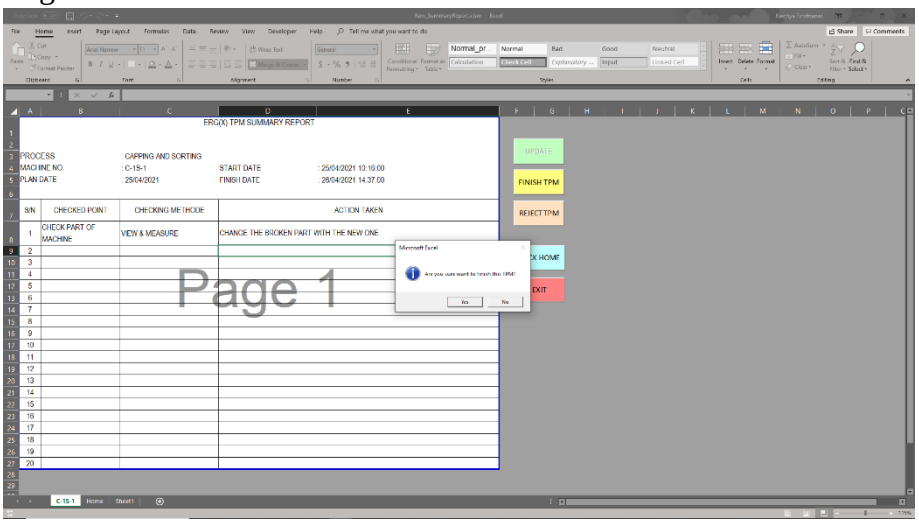
Picture 3.71 Reject Result on Home Sheet

d. Login as Leader – Send Email for Rejected TPM
By clicking Summary of Rejected button.

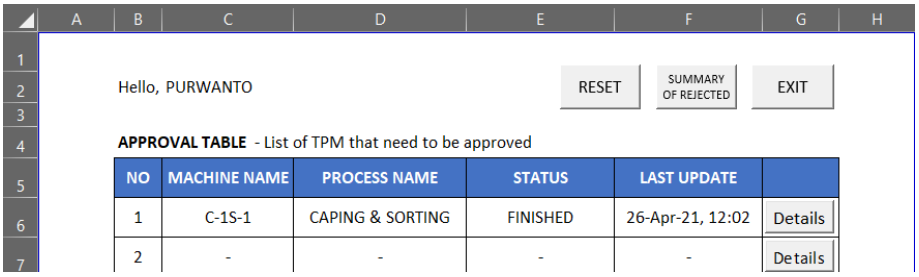


Picture 3.72 Rejected TPM List at Reject Sheet

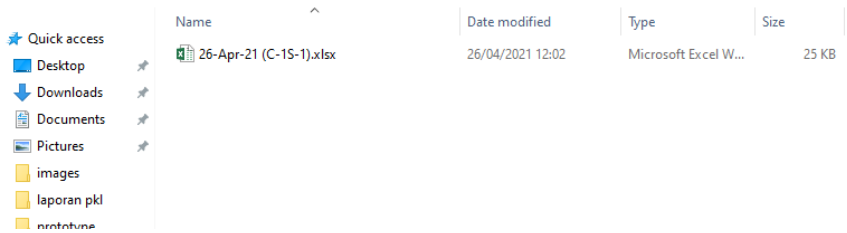
e. Login as Leader – Finish The TPM



Picture 3.73 Finishing the TPM

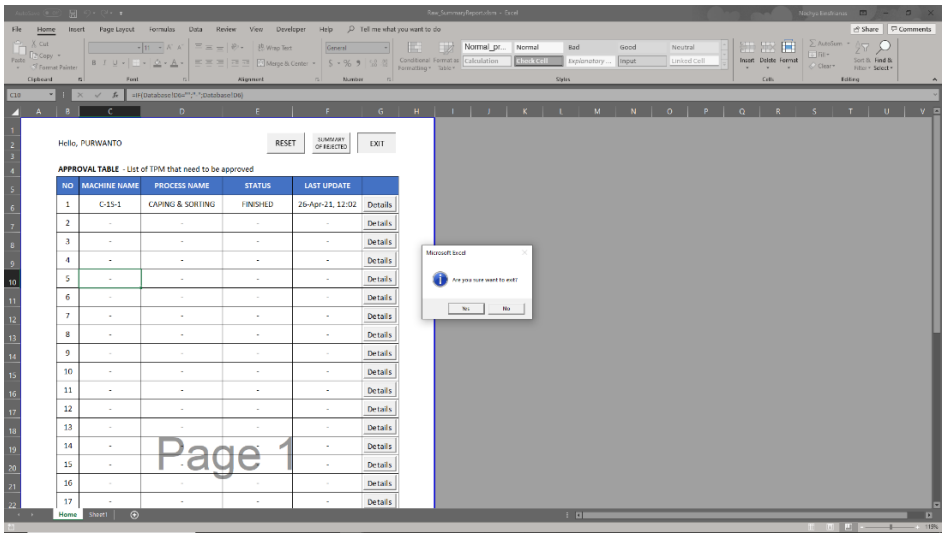


Picture 3.74 Finish Result on Home Sheet



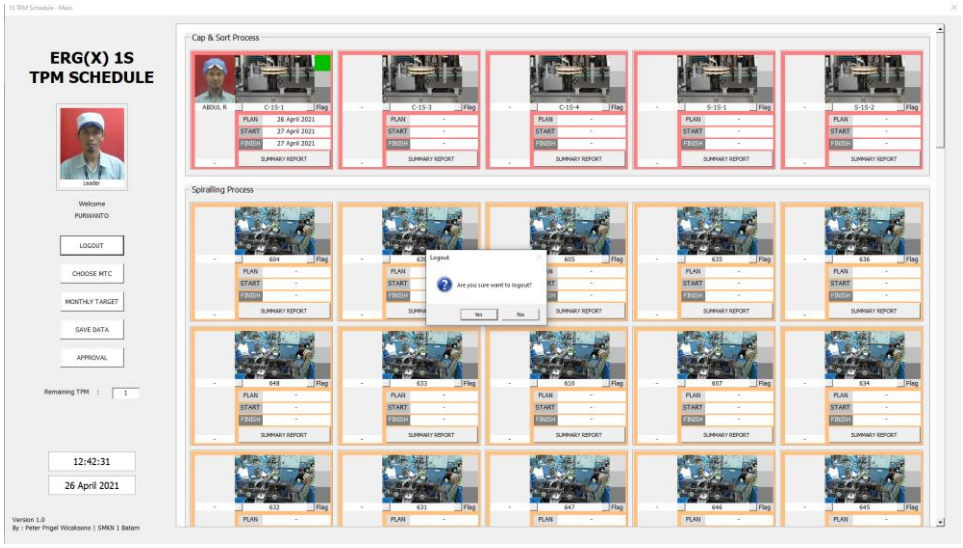
Picture 3.75 Export Check Sheet Result

8. Exit from Summary Report



Picture 3.76 Exit Summary Report Pop-up

9. Logout



Picture 3.77 Logout From TPM Form

CHAPTER IV
CLOSING

A. Conclusion

Based on the discussion, a series of stages in the process of making a project, to carrying out several system tests, conclusions can be drawn including.

1. This scheduling system is quite helpful in carrying out some machine maintenance. Make maintenance activities more regular and clear.
2. The making of this scheduling system also helps in reminding people who carry out maintenance, minimizing schedule build up and un-running of machine maintenance activities.
3. This system also makes it easier for leaders to check and approve machine maintenance activities that are being carried out whether it is going well or not.

B. Suggestion

This machine maintenance schedule system is highly recommended because it can help support ease of work. What's more, PT. Panasonic Industrial Device Batam is very supportive in terms of automation which can increase work efficiency and effectiveness. For the future, maybe the concept of this scheduling system can be developed even better. For example, we can use the Python & MySQL programming language or use the Objective-C & MySQL programming language with the Visual Studio Editor. So, the system can be more powerfull than before. The author hopes that this system can be used properly and optimally and runs in accordance with the main purpose of making this system.

REFERENCES

Adib Mubarok. 2015. *Rumus Excel*. Jakarta: PT. Alex Media Komputindo.

Kelas Excel (2018, June 11). *Tutorial Macro VBA Indonesia* on: https://www.kelasexcel.id/p/tutorial-macro-vba-excel-indonesia.html

Ron de bruin (2017, November 1), *Mail Range/Selection in the body of the mail* on https://www.rondebruin.nl/win/s1/outlook/bmail2.htm

Rumus Rumus (2021, March 23), *Rumus Macro Excel dan VBA Dasar Untuk Pemula* on https://rumusrumus.com/macro-excel/

Stackoverflow (2019, April 2), *Excel Macro VBA* on: https://stackoverflow.com/questions/31466464/excel-macro-vba

Trevor Eyre (2020, January 3), *Excel VBA Date Picker* on https://trevoreyre.com/portfolio/excel-datepicker/

Yudhy Wicaksono. 2020. *Kolaborasi Macro Excel & MySQL untuk Membuat Aplikasi Database*. Jakarta: PT. Alex Media Komputindo.