THE DECISION ANALYSIS AND DEVELOPMENT OF WEB-BASED OPERATION DASHBOARD PRODUCTION USING SCRUM FRAMEWORK

Syaeful Anas Aklani[1],**Jassy Lee**[2]

[1,2]Sistem Informasi Universitas Internasional Batam

Sei Ladi, Jl. Gajah Mada, Baloi Permai, Kec. Sekupang, Kota Batam, Kepulauan Riau 29442

Email: syaeful@uib.ac.id [1], 1831102.jassy@uib.edu[2]

Abstrak

Dengan pesatnya perkembangan teknologi, sistem informasi dapat dikembangkan dan digunakan untuk membantu masyarakat memperoleh informasi dengan lebih mudah. Salah satu sistem yang banyak digunakan untuk menyajikan informasi penting adalah dasbor operasional yang biasa digunakan di industri manufaktur. Manufaktur sendiri berarti suatu proses untuk mengubah bahan mentah menjadi suatu produk. Terdiri dari beberapa proses yaitu perencanaan bahan baku, pemilihan bahan, serta melalui berbagai proses mesin yang telah diatur dan dikenal sebagai lini produksi. Di setiap lini produksi, terdapat pemimpin lini untuk mengawasi dan memantau kinerja lini serta mengidentifikasi dan memecahkan masalah yang muncul. Namun, tanpa akses ke data real-time dari kinerja jalur yang ada, akan sulit untuk mengidentifikasi masalah sehingga meningkatkan kemungkinan membuat pilihan yang salah serta menambah kerugian. Oleh karena itu, penelitian ini bertujuan untuk menganalisis, merancang, dan mengembangkan dasbor operasional berbasis web yang menampilkan data yang umumnya digunakan. Dengan menggunakan PHP, JavaScript, Bootstrap serta SQL Server, penulis berhasil mengembangkan dasbor operasional berbasis web yang dapat membantu para pemimpin lini dalam mengakses data lini produksi secara real-time sehingga memungkinkan pimpinan lini untuk mengambil keputusan yang lebih baik.

Kata Kunci: *dasbor operasional, javascript, php, sql server, web*

Abstract

With the rapid development of technology, information systems can be developed and used to help people obtain information more easily. One of the systems that are popularly used to present critical information is the operational dashboard which is commonly used in manufacturing industries. Manufacture itself means a process to convert raw materials into a product. It consists of a few processes, namely planning and choosing materials, as well as going through various machine processes that is known as a production line. In each of the production lines, there are line leaders to monitor line performances as well as identify and solve the problem that arises. However, without access to the real-time data of the line performance, it will be difficult to identify the problem and increase the probability of making the wrong choice and additional losses. For that reason, this research aims to analyze, design, and develop a web-based operational dashboard that displays generally used data. Using PHP, JavaScript, Bootstrap as well as SQL Server, the writer succeeded in developing a web-based operational dashboard that could help line leaders in accessing real-time production line data hence allowing them to make a better judgement.

KeyWords : *javascript, operational dashboard, php, sql server, web*

I. INTRODUCTION

With the rapid development of technology, information systems can be developed and used to help people obtain information more easily. One of the systems that is popularly used to present critical information at a glance is dashboard [1].

According to Oxford dictionary, dashboard is a graphical summary of various information which is used to give an overview of a business. There are many types of dashboard, ranging from the dashboard that are available on vehicles, which display the controls for our vehicle to data dashboard, which displays a detailed graphic summary of the company's data. One of the types of data dashboard is operational dashboard. This type of dashboard is usually used in manufacturing industries.

Manufacturing or manufacture is a word that originated from Latin, manus and factus, which translates as a process to convert raw materials into a product [2]. Manufacture itself consists of a few processes, namely planning raw materials, choosing materials, as well as going through various machine processes or stations and operations that have been organized [3]. The various stations that the materials will go through are usually arranged in order of the manufacturing process, which is usually known as production line.

In the production lines, there are inputs, outputs, and units. A unit is any item that is being processed [4]. The unit that goes through each of the stations in production line for the first time are called inputs, and the unit that has passed the stations in production line with good quality are considered as output. In some cases where the quality of a unit failed to reach the minimum standard, the unit is then re-tested for a maximum of three times and will be considered output if it passed and sent to repair line if it failed to reach the minimum standard.

In each of the production lines, there are line leaders who oversee and monitor the line performances as well as identify and solve problems that arise during the manufacturing process. However, if line leaders don't have any access to the real-time data

of the line performance on hand, tackling problems that arise would be difficult as not only it would increase the probability of making the wrong choice, but also additional waiting time and line losses. In this kind of case, having an operational dashboard for each production line would be helpful for line leaders as it could greatly reduce the time needed for line leaders to acquire line's data since it displays a detailed graphic summary of the line's data, which would help line leaders in making a better decision and waiting time as well as losses could be reduced greatly. While manufacturing industries uses the same kind of operational dashboard, the types of data that is displayed on the dashboard might differ. For that reason, there is a need for an operational dashboard that displays real-time data to help line leaders in making a better choice while monitoring the line performances. The operational dashboard would be developed as web-based so that it can be hosted on a server computer, which will allow user to access it easily. The data displayed consists of general information that the writer think could assist in monitoring line performances.
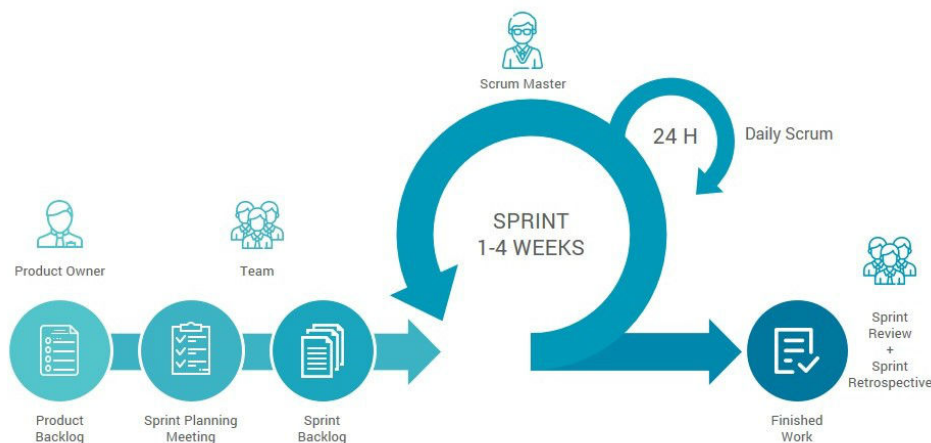
II. RESEARCH METHODOLOGY

Figure. 1: Scrum Process

The method used in developing this project is scrum methodology. Scrum is one of the commonly used software development frameworks that adopts agile methodology [5]. Scrum methodology helps programmers to develop products incrementally by aiding programmers in working coordinately by dividing their works into small tasks that are ranked according to priority which can be completed in fixed duration cycles or "sprints" while tracking their work progress and re-plan tasks in regular meetings [6]. The roles required in implementing Scrum are supervisor as scrum master, product owner and team will be fulfilled by us. Scrum requires us to be flexible in our plans, so we can always improve and deliver the most value [7]. By using Scrum, we will be able to plan and work on the project flexibly and objectively if there are changes needed as it prioritizes flexibility in its development.

As shown in Figure 1, scrum process starts with backlog. Product backlog is a list of user stories, feature that user wants in the application. In this phase, we will list out the types of data that is generally needed in the dashboard, this includes choosing the types of charts that will be used to display each type of data in the dashboard. These product backlogs will then be sorted according to priorities and taken into sprint planning. In the sprint planning phase, team will have a brief meeting to plan with scrum master to clarify what should be done in the upcoming sprint, what is the output and how to achieve it. Team will select a set of tasks from updated product backlog, according to the highest priority, estimate the time needed to finish and proceed to kick off the sprint. After planning, team will start working on the set of tasks in the current sprint from the highest priority. The tasks then will be executed into code format. The UI (User Interface) will be developed using HTML/CSS, and the backend will be using PHP and jQuery. Meanwhile, the database used to store data is SQL Server. The next process is Daily Scrum which is also known as Scrum Sprint. In this phase, team will keep track on the progress of each task, update any new information and plan for the upcoming week or month in weekly sprint meeting with scrum master. This is to ensure that all tasks are right on track. After a sprint is finished, the last phase is Sprint review. In this phase, team will review the result of the sprint together with scrum master to prove that we're moving towards the right direction and to get feedbacks from scrum master on the current sprint result so team would know what could be improved in the upcoming sprint which could help in the upcoming sprint planning. If all the product backlog is finished, the design and development of UI (User Interface) and backend of the dashboard should also be finished, and the sprint will end.

III. RESULTS
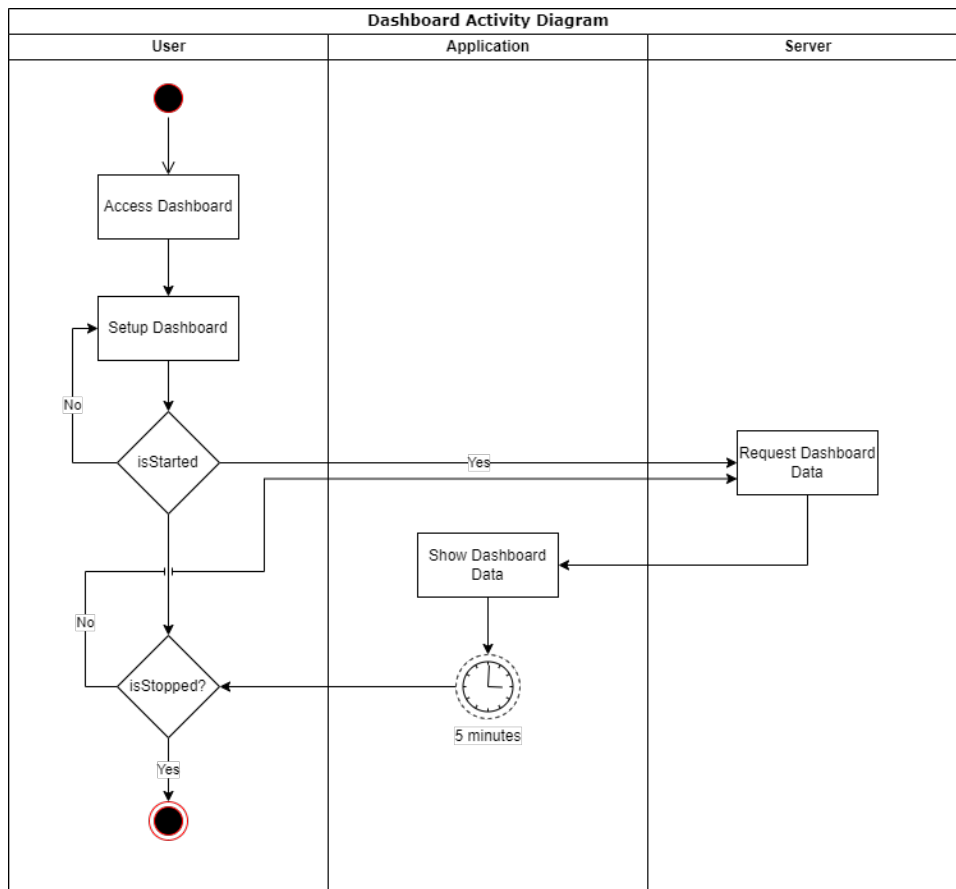
A. Activity Diagram

Figure. 2: Dashboard System Activity Diagram

The process of how this dashboard system works is shown in Figure 2. When user first access the dashboard, user would see the dashboard page directly, user then can setup the dashboard according to their needs. The dashboard will only start after user clicked on the start button with all values filled during setup. When the dashboard is started, it will then request dashboard data to the server, which the server will respond with the data that will be displayed onto the dashboard's charts. After started, the dashboard would periodically request data from the server if the system is not stopped by the user. If the user stopped the dashboard, then the process will end.

JIKO (Jurnal Informatika dan Komputer)
Vol. 6, no. 1, Febuari 2022, hal. 65–74
e-ISSN : 2477-3964 — p-ISSN : 2477-4413
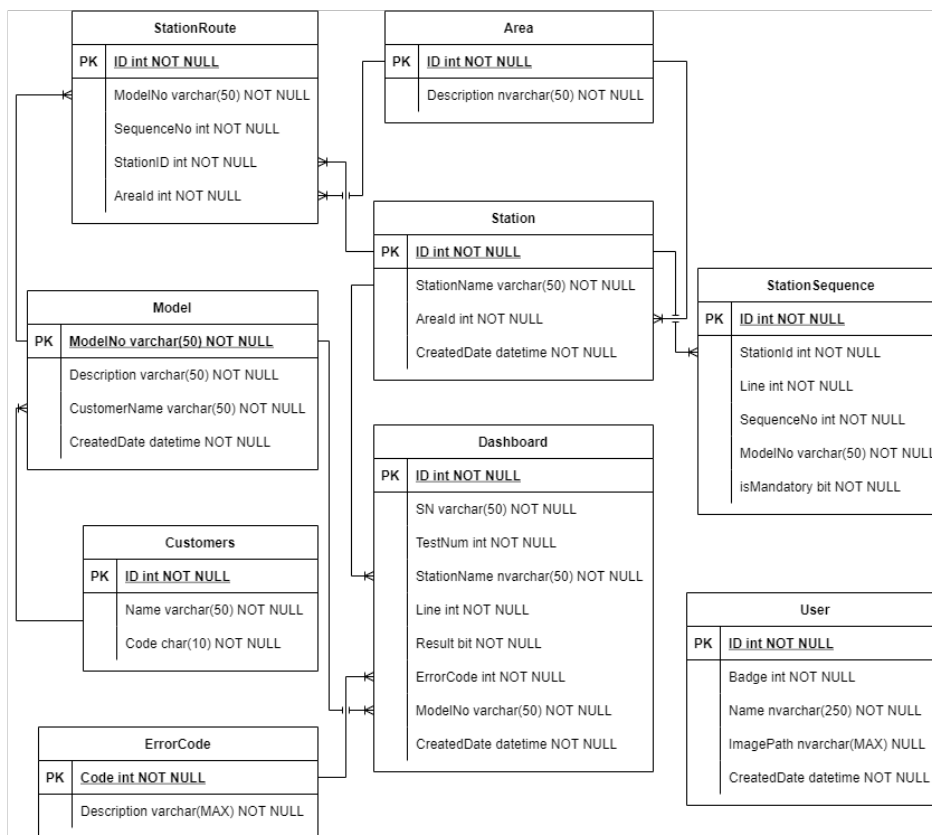
B. Entity Relationship Diagram (ERD)

Figure. 3: Dashboard System Entity Relationship Diagram

From the designed database schema show on Figure 3, table Dashboard is used as the main table for this system. The table dashboard will contain the data of units that has gone through each station. It has ID as its primary key. Table dashboard referenced to multiple tables for foreign key relation, namely column StationName from table Station which is used to store station's name of the current station or process that the unit goes through, column ErrorCode from table ErrorCode used to store error codes when a specific unit fails to reach the minimum standard, and column ModelNo from table Model which is used to store the model name of the unit. The next table is table Station. This table contains all the stations name and has ID as primary key which is referenced as a foreign key relation with multiple tables. The first table that referenced column ID from table station as foreign key is table StationSequence, which contains the organized sequence and details of the stations that a model will go through. The second table is table Dashboard to store the station name and lastly, table StationRoute, which is used to store the list of stations that a model needed to go through. The table Station also referenced to table Area for column AreaId to know which area the station is. Table Area's primary key, Id, is also referenced as foreign key in table StationRoute. In table StationRoute, column ModelNo is referenced from table Model's primary key. Table model is used to store the model of all customer, where the column customer is referenced to table Customer's column name. Lastly this database schema also has the table user that is not related to any other table. This table is used to store user's name and their image to show in the dashboard interface.

C. Discussion

The web-based operational dashboard is successfully developed using PHP, JavaScript and Bootstrap programming language as well as SQL Server as database and has successfully hosted on Internet Information Services (IIS).

JIKO (Jurnal Informatika dan Komputer)
Vol. 6, no. 1, Febuari 2022, hal. 65–74
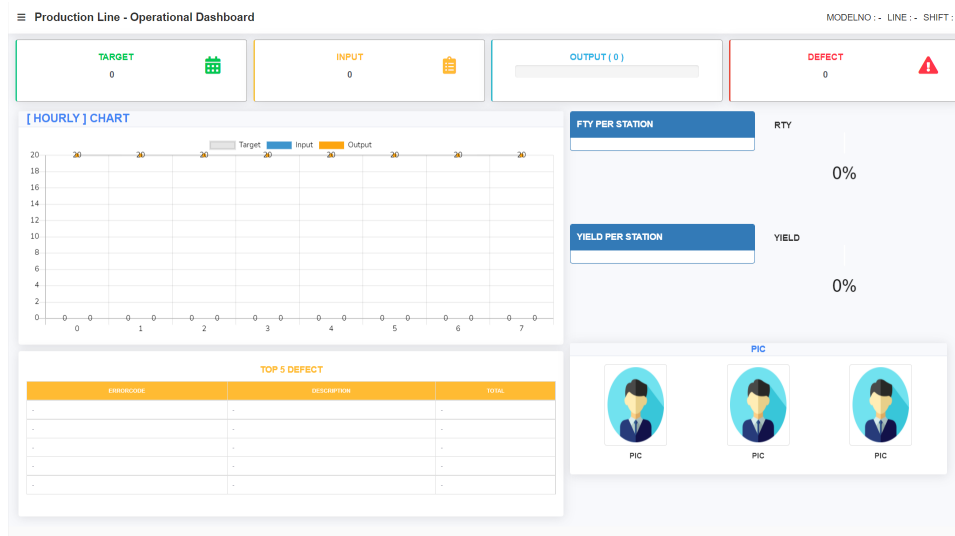e-ISSN : 2477-3964 — p-ISSN : 2477-4413

Figure. 4: Dashboard Interface

When user first access the webpage, the dashboard interface will be displayed directly as shown in Figure 4. Following will be detailed explanation of the dashboard interface by parts.

Figure. 5: Dashboard Interface

First part of this dashboard is shown in Figure 5, which is the navigation bar. The navigation bar consists of a hamburger menu and title on the left part, and details of the dashboard that has been set up on the right part. When clicked on the hamburger menu, a sidebar to set up this dashboard will show up. Dashboard setup sidebar is broken down into two parts and shown in Figure 6. and Figure 7.

Figure. 6: Dashboard Setup Part 1 of 2

In the first half of the sidebar consists of 6 combo boxes for user to choose. In Figure 6, the combo boxes have been filled up for testing purposes. The first combo box, AREA, is for user to choose which area of production line will the dashboard

be used. The next combo box is CUSTOMER, user will choose for which customer is this production line running for, this combo box is also used to filter model by customer chosen. After choosing customer, user will be able to select the model of product that will be running on the production line in the MODELNO combo box. In the LINE combo box, user will be able to choose which line is this dashboard for, as this dashboard will only show information of the line selected. The next combo box is INPUT STATION, this combo box will allow user to select a station for the dashboard to get input data according to the station selected. Same goes for the OUTPUT STATION combo box. With the input station and output station chosen, the dashboard will display data of the two chosen station and the station between them according to the station sequence.

Figure. 7: Dashboard Setup Part 2 of 2

In the second half of the sidebar, user will be able to choose the SHIFT TIME and BREAK TIME with the date time picker for the dashboard. This is to ensure that the dashboard is flexible enough to be used because not all manufacturing company has the same shift schedule. The next three combo box is for Person-In-Charge (PIC). This is where user select the PIC of the line that will be displayed in the dashboard. If there is only one PIC, then user can just leave the other two empty. Currently, only three PIC is available in this dashboard. The last text box that user needs to fill is TARGET. This is to show the output target of the line. After filling in all the values, there are two buttons in the bottom most of the sidebar. The default text for these two buttons is START and RESET. When button start is clicked, the system will start getting data from database and then the text of the start button would change to STOP. Lastly, the RESET button is to reset all remove all values that has been filled in. The following figure is an example of the dashboard interface when dashboard has been setup and started successfully.
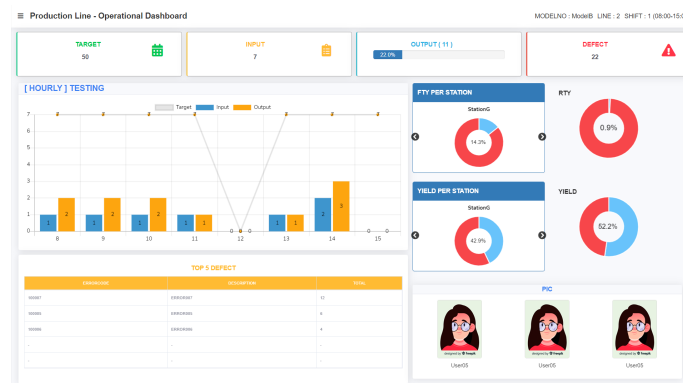

Figure. 8: Dashboard Interface Started

As seen in Figure 8, with the dashboard started, the charts are filled up with data. The details of the dashboard on the right part of the navigation bar are also filled with the data user has chosen. This can be seen in Figure 9.

MODELNO : ModelB LINE : 2 SHIFT : 1 (08:00-15:00)
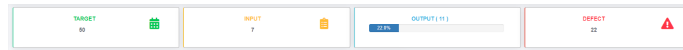
Figure. 9: Dashboard Navigation Bar Details

Figure. 10: Dashboard Information Cards

The second part of this dashboard is shown in Figure 10, which is a collection of cards that consist of target, input, output, and defect quantity of the specific line. From left to right, TARGET is taken from the quantity that user inputted as target during setup, INPUT is total quantity of every unit that passed through the INPUT STATION chosen for the first time, while OUTPUT is total quantity of unit that has passed the OUTPUT STATION with no defects. Lastly, DEFECT shows the quantity of component that is found defect during the production process in the shift chosen on that line.
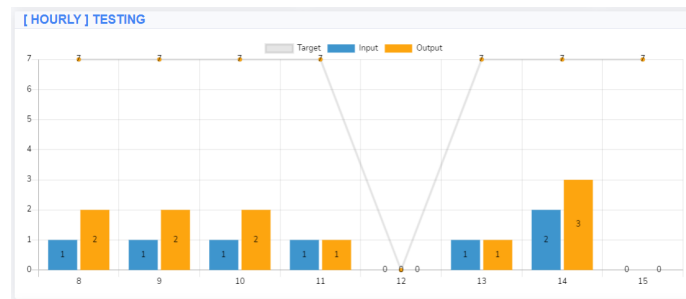

Figure. 11: Dashboard Hourly Chart

The next part shown in Figure 11 is a chart. This chart is used to show hourly data and target of the line between the SHIFT TIME that has been set previously. The blue colored bar chart indicates input, while the orange bar chart indicates output. The quantity of input and output of the chart is taken according to the chosen INPUT STATION and OUTPUT STATION. Other than bar chart, there is also line chart to indicate target per hour, these target per hour is divided from the total target that has been inputted during set up.
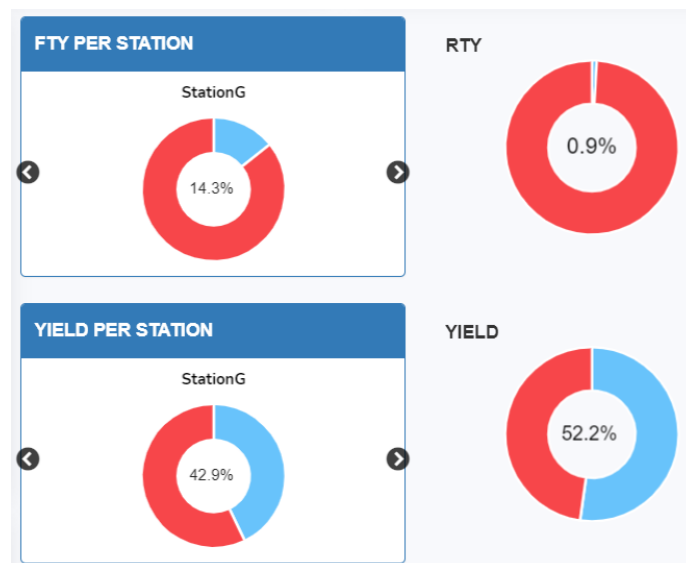

Figure. 12: Dashboard Pie Charts

he fourth part shown in Figure 12 is located on the right side of hourly chart. It consists of four pie charts, which are First Time Yield (FTY) per station chart, Rolled Throughput Yield (RTY) chart, Yield per station chart, and overall yield chart.

Both FTY per station and Yield per station chart displays data of all station by implementing a carousel to slide and show each station's data periodically. The data for these pie charts are data that has been calculated using formulas by referencing various sources. Starting from FTY per station chart, FTY is the ratio of the number of good units that comes out of a process to the number of units that goes in the process [8]. The data for this chart uses the following formula.

$$Fist\ Time\ Yield = \left[\frac{Number\ of\ Product\ Without\ Error}{Total\ Numbers\ of\ Products}\right] \quad (1)$$

From the formula given in 1, number of products without error means total of units that went under the production line process and coming out of the process without errors, while total number of products is the total quantity of units that went in the process. An example of calculation of FTY is as follow. The total quantity of units that goes into process A is 100, and from 100 units, 20 of them has errors. While process B has 80 units going into the process and has 10 errors. This means for process A, the number of products without error is 80, and total number of products is 100. So, the FTY for process A is 80/100. Meanwhile for process B, number of products without error is 70 and total number of products is 80. So, the FTY for process B is 70/80.

$$RTY = [(FTY_1) * (FTY_2) * (FTY_3) * (FTY_4) * (FTY_5)] \quad (2)$$

Using the previous FTY calculation of process A as FTY1 and process B as FTY2, the RTY can be calculated as following.
RTY = (FTY1) * (FTY2)
RTY = (80/100) * (70/80)
RTY = 70/100

$$Yield = \left[\frac{Good\ Units\ Produced}{The\ Numbers\ of\ Units\ Entering\ The\ Process}\right] \quad (3)$$

The two last charts are yield per station and overall yield chart. Both yield per station and overall yield chart uses formula shown in 3. As the formula wrote, yield is the ratio of good units produced to the number of units entering the station [4]. From the above formula, good units produced means units that total unit that comes out of the station without error, unlike FTY, which only counts total units that passed without the ones after going to re-tests, the good units in yield includes all the units that comes out without error, whether if it has been re-tested or not. Next is the number of units entering the process, this means the total quantity of units that goes in all the stations.

ERRORCODE	DESCRIPTION	TOTAL
	TOP 5 DEFECT	
100007	ERROR007	12
100005	ERROR005	6
100006	ERROR006	4
-	-	-
-	-	-

Figure. 13: Dashboard Top 5 Defect Table

The fifth part of this dashboard is the top 5 defect table. As the title of the table suggests, this table shows the top 5 defects in all the station. It will show the error code, description of the error and total quantity of the error which will be ordered by the most amount of error.
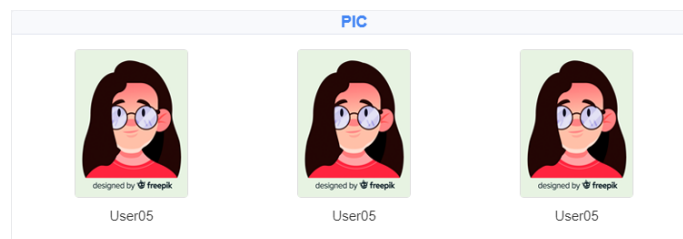
PIC

User05   User05   User05

Figure. 14: Dashboard Person-In-Charge Interface

The last part of this dashboard is the Person-In-Charge (PIC). Currently, the dashboard can display a maximum of three PIC that has been chosen during setup. The PIC's image and name will be displayed directly when their name is chosen during setup.

JIKO (Jurnal Informatika dan Komputer)
Vol. 6, no. 1, Febuari 2022, hal. 65–74
e-ISSN : 2477-3964 — p-ISSN : 2477-4413

D. Black Box Testing

1) Form Functionality

Table I: Form Functionality Test

No.	Test	Target	Result
1	Selecting PIC	Image and name of selected PIC will show directly in the PIC interface	OK

2) Start Button Functionality

Table II: Start Button Functionality Test

No.	Test	Target	Result
1	Starting dashboard without filling all values with exception of PIC and break time	A pop-up would be shown asking user to fill all values.	OK
2	Starting dashboard with break time outside the time range of shift time	A pop-up would be shown warning user to select a break time inside time range of shift time	OK
3	Starting dashboard with quantity 0 as target	A pop-up would be shown warning the user of invalid target quantity	OK
4	Starting dashboard with all values filled	Dashboard starts and display data according to the setup and start button text changes to **STOP**	OK

3) Stop Button Functionality

Table III: Stop Button Functionality Test

No.	Test	Target	Result
1	Click on stop button while dashboard is started	A pop-up would be shown to notify user that the dashboard has been stopped, the dashboard stops loading new data and the text on stop button would change into **CONTINUE**	OK

4) Continue Button Functionality

Table IV: Stop Button Functionality Test

No.	Test	Target	Result
1	Click on continue button when the dashboard is stopped	Dashboard will continue to get data	OK

5) Reset Button Functionality

Table V: Stop Button Functionality Test

No.	Test	Target	Result
1	Click on reset button while dashboard is started	A pop-up would be shown warning the user that the dashboard system is currently running	OK
2	Click on reset button when dashboard is stopped	All filled values will be cleared, and the dashboard user interface data will be cleared and return to default	OK

The dashboard system is tested using two popularly used browsers which are Google Chrome and Microsoft Edge. The above test tables indicates that it is compatible with both browsers.

IV. CONCLUSION AND SUGGESTIONS

A. Conclusion

From the result of developing and testing, the author has drawn several conclusions, namely:

1) The operational dashboard is compatible with Google Chrome and Microsoft Edge which is two of the popularly used browser.
2) The operational dashboard is believed to have display data that is needed by line leaders.
3) With this operational dashboard, the losses caused by unavailability of real-time data for line leaders to identify and solve problems can be suppressed.
4) With the dashboard hosted on a server computer's IIS, it can be accessed by multiple computers with the same network at the same time.

B. Suggestions

From the result of developing and testing, there are a few suggestions and aspects which can be used as a guide to improve and taken in the future, such as:

1) Adding an accumulation input and output chart interface.
2) Making the interface more responsive and user-friendly.
3) Allowing the dashboard to be accessed through mobile phones.

REFERENCES

[1] W. Noonpakdee, T. Khunkornsiri, A. Phothichai, and K. Danaisawat, "A framework for analyzing and developing dashboard templates for small and medium enterprises," 2018 5th Int. Conf. Ind. Eng. Appl. ICIEA 2018, pp. 479–483, 2018, doi: 10.1109/IEA.2018.8387148.
[2] N. H. Pranata, E. Suryono, and B. Margono, "PENGARUH MODIFIKASI SPLITTING NICKS TOOL DRILL TERHADAP SPINDLE LOADS DAN WAKTU PROSES DRILLING MATERIAL S45C," pp. 6–11, 2020.
[3] Y. Gasela, Sujadmi, and L. Febriani, "PEMBERDAYAAN NARAPIDANA DI LEMBAGAPEMASYARAKATAN KELAS II A PANGKALPINANG," vol. 1, pp. 654–661, 2021.
[4] R. Walters, "First Pass Yield vs," 2021. [Online]. Available: https://blog.masterofproject.com/rty/. [Accessed: 01-Jan-2022].
[5] D. Alfath, "Diajukan sebagai salah satu syarat untuk memperoleh Gelar Sarjana Pendidikan Oleh:," 2017.
[6] E. Senabre Hidalgo, "Corrigendum to 'Adapting the scrum framework for agile project management in science: case study of a distributed research initiative' [Heliyon 5 (3) (March 2019) e01447] (Heliyon (2019) 5(3), (S2405844018340635), (10.1016/j.heliyon.2019.e01447))," Heliyon, vol. 5, no. 4, p. e01542, 2019, doi: 10.1016/j.heliyon.2019.e01542.
[7] M. Dalmijn, "If you are not flexible, you are not doing Scrum _ by Maarten Dalmijn _ Serious Scrum _ Medium," 2019. [Online]. Available: https://medium.com/serious-scrum/if-you-are-not-flexible-you-are-not-doing-scrum-ebf41f47d151#: :text=The essence of Scrum is,be able to improve them.
[8] E. Kosven, "First Time Yield and Rolled Throughput Yield — ISSSP for Lean Six Sigma." 2021.