

### Analisa sebab akibat

Berdasarkan analisa kondisi yang sudah dijelaskan di poin sebelumnya terdapat beberapa permasalahan yang ditemukan pada sistem SSC. Maka dari itu untuk menemukan akar permasalahan penulis menggunakan metode diagram *fishbone* (tulang ikan) yang ditunjukkan pada gambar berikut.



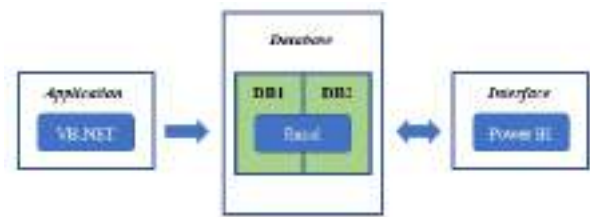
Gambar 7. Diagram *fishbone* sebab akibat

Pada gambar 7 menunjukkan diagram *fishbone* sebab akibat dengan gejala utama ialah *input data ke database* dilakukan secara manual oleh *data entry* pada *shift schedule control* DIJ. Dengan akar permasalahan yang ditemukan ialah *input data ke database* masih dilakukan manual, tidak memanfaatkan komputer yang tersedia di *line production* dan operator masih menggunakan kertas untuk mencatat produksinya. Secara keseluruhan permasalahan tersebut muncul dikarenakan belum ada sistem bagaimana data yang dicatat pada SSC langsung masuk dan tersimpan di *database* tanpa perlu memasukkan data ke *database* secara manual.

Sehingga sistem untuk *input data* yang akan dirancang dalam bentuk aplikasi berbasis *desktop* bisa digunakan langsung oleh operator dan *data entry*, dimana *data entry* hanya menggunakan aplikasi ketika terdapat data produksi yang tidak dapat tersimpan oleh operator. Karena aplikasi tersebut berbasis *desktop* (*by computer*) dan *path* penyimpanan datanya ke *server* maka aplikasi tersebut tidak dapat dibuka menggunakan *handphone* dan hanya bisa digunakan saat jaringan komputer terhubung dengan *server*. Sedangkan sebelum dan setelah adanya sistem ini, untuk melihat *report*-nya masih sama yaitu menggunakan *report* dalam bentuk *spreadsheet* dan *OEE dashboard*.

### Perancangan Sistem Aplikasi SSC

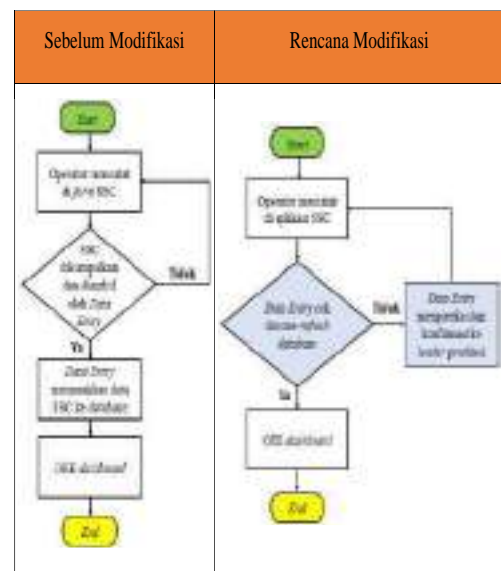
Konsep perancangan ini menggambarkan konsep dari perancangan sistem yang akan dibuat. Berikut konsep perancangan yang ditunjukkan pada gambar berikut.



Gambar 8. Konsep perancangan aplikasi SSC

Pada gambar 8 menunjukkan konsep dari perancangan sistem yang akan dibuat. Terdapat 3 bagian utama yaitu *application*, *database* dan *interface*. Pada bagian aplikasi, aplikasi berbasis *desktop* dibuat menggunakan bahasa Visual Basic .NET dengan bantuan IDE Visual Studio [5]. Bagian *database* terbagi menjadi dua bagian yaitu *database* pertama dan kedua, dibuat menggunakan Excel. Sedangkan bagian *interface* yang sebagai *OEE dashboard* dibuat menggunakan Power BI namun yang dibahas dalam penulisan hanya koneksi ke *database*.

Langkah kerja yang dibuat merupakan perancangan setelah sistem diterapkan. Langkah kerja yang dilakukan sebelum menerapkan sistem berbeda dengan setelah menerapkan sistem yang sudah dirancang. Perbedaan sebelum dan rencana modifikasi tersebut ditunjukkan pada gambar berikut.



Gambar 9. Sebelum dan rencana modifikasi

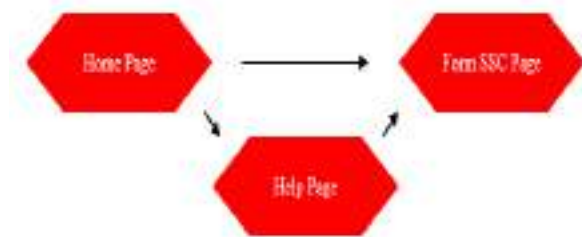
Pada gambar 9 menunjukkan perbedaan langkah kerja sebelum dan rencana modifikasi dari sistem SSC. Pada sebelum sistem terbaru diterapkan terlihat ada 4 langkah yaitu dari operator mencatat di *form* SSC, SSC dikumpulkan dan diambil oleh *Data Entry* namun jika terdapat ketidaklengkapan jumlah SSC maka operator akan mencatat ulang, *Data Entry* memasukkan data SSC ke *database* dan hingga menjadi *OEE dashboard*. Sedangkan setelah sistem terbaru diterapkan hanya menjadi 3 langkah utama yaitu operator mencatat di *form* SSC, *Data Entry* cek dan *me-refresh database*

namun jika terdapat kesalahan di *database* maka *data entry* akan mengoreksi lalu konfirmasi ke *leader* produksi, dan hingga menjadi *OEE dashboard*.

### III. HASIL DAN ANALISA

#### Pembuatan Aplikasi SSC

Pembuatan aplikasi SSC (*Shift Schedule Control*) menggunakan *software Microsoft Visual Studio* dengan bahasa *Visual Basic* dan dengan *layout* mengacu pada form SSC kertas. Namun ada beberapa yang perlu dikurangi dan ditambahkan pada *layout form*-nya. Berikut gambar 3 skema aplikasi yang akan dibuat.



Gambar 10. Skema aplikasi SSC

Pada gambar diatas menunjukkan skema / bagian aplikasi yang akan dibuat, terdiri dari home page, help page, dan form ssc page. *Home page* dibuat sebagai halaman depan dari aplikasi sebelum masuk ke halaman *form*-nya. *Help page* dibuat dengan tujuan membantu user ketika shift form SSC page yang dibuka tidak sesuai sehingga perlu manual dalam memilih shift-nya. *Form SSC page* dibuat terdiri dari 3 *form* yaitu *form* untuk shift 1, shift 2 dan shift 3. Masing-masing *form* memiliki format yang sama namun yang berbeda hanya di angka shift-nya saja.

#### Database Excel

Database *excel* dibuat dibagi menjadi dua bagian, yaitu *database 1* dan *database 2*. Pembagian database tersebut bertujuan mempermudah dalam *problem solving* sekaligus agar lebih rapih. Pembahasan mengenai pembagian *database* dijelaskan pada sub berikut.

##### Database 1 (DB1)

Database 1 atau DB1 merupakan database untuk meletakkan file excel. File excel tersebut memiliki format penamaannya masing-masing sesuai kode mesin dan shift seperti B04\_Shift1, B04\_Shift2 dan B04\_Shift3. Jadi satu mesin memiliki 3 file excel dengan tujuan data yang masuk bisa diolah dengan benar dan sekaligus mempermudah *problem solving* karena data yang masuk sangat banyak jika dalam kurun waktu yang panjang. Sehingga banyaknya jumlah file excel yang berada di database 1 sesuai jumlah mesin, jika mesin ada 10 maka file excel-nya ada 30.

Pada file excel di DB1 memiliki 5 sheet yaitu Output – SSC Form, Output Part, OEE Summary, DT dan Remarks. Sheet Output – SSC Form merupakan sheet yang menjadi tempat masuknya data dari aplikasi

sehingga tidak perlu dimasukkan rumus / formula di cell-nya. Sheet Output Part, OEE Summary, DT dan Remarks memiliki format yang sama seperti di penjelasan sebelumnya mengenai database excel.

##### Database 2 (DB2)

Database 2 merupakan *database* yang dibuat dengan tujuan gabungan dari file-file yang berada di *database 1*, *database 2* inilah yang akan dihubungkan ke *OEE dashboard*. Di *database 2* terdapat 3 file excel yaitu pertama file yang datanya diambil dari file *database* sebelumnya, kedua yaitu file gabungan dari file-file yang berada di *database 1*, dan ketiga yaitu file gabungan dari file pertama dan kedua yang file ketiga ini akan dihubungkan ke *OEE dashboard*.

#### Analisa Hasil

Pengujian aplikasi SSC dilakukan dengan trial langsung ke *line* yang dilakukan oleh operator. Pengujian bertujuan untuk mengetahui kemampuan aplikasi melakukan prosesnya. Selain itu diuji mengenai kemudahan penggunaan dan tampilannya. Aplikasi tersebut di *install* di setiap komputer sesuai kode mesinnya misalnya mesin dengan kode A03 akan di *install* aplikasi A03. Jadi tiap mesin memiliki aplikasinya masing-masing sesuai kode mesinnya. Namun tidak semua mesin memiliki satu komputer, ada juga satu komputer untuk dua mesin seperti mesin A01 dan A02 tergabung dalam satu komputer. Setelah aplikasi berhasil di *install* selanjutnya demo ke operator mengenai penggunaan aplikasinya. Setelah itu melakukan input data ke aplikasi seperti pada gambar berikut.



Gambar 11. Pengujian Input data di aplikasi SSC

Pada gambar diatas menunjukkan cara *input* data di aplikasi. Berawal dari mengisi kolom *leader*, *kpk*, nama, *part number*, *shiftly target*, *duration toy*, *target* dan *actual output*. Kolom lainnya bersifat opsional sesuai kondisinya seperti kolom *reject* diisi ketika ada *reject*. Kemudian data tersebut akan masuk ke *database 1* sesuai nama file-nya.

Pengujian keakuratan formula/rumus dilakukan dengan cara mengetahui data yang masuk berhasil ditampilkan tanpa ada yang *error*. Pengujian ini hanya

dengan cara melihat di dua sheet yaitu *Output Part* dan *OEE Summary*.

Pengujian koneksi dilakukan antara *interface report analysis* dengan *database excel* dengan cara *refresh* pada *project dashboard*-nya seperti pada gambar berikut.



Gambar 12. Pengujian koneksi dashboard

Pada gambar 12 menunjukkan menu untuk *refresh* *OEE dashboard* dengan cara menekan menu *refresh*. Ketika berhasil di *refresh* artinya koneksi dengan *database* berhasil sedangkan ketika terjadi *error* artinya koneksi gagal.

Dengan adanya sistem terbaru yang semula pencatatannya di kertas digantikan di aplikasi berbasis *desktop* yang hanya bisa dibuka menggunakan komputer. Perbandingan penggunaan kertas tersebut bisa dilihat pada gambar berikut.



Gambar 13. Perbandingan penggunaan kertas setelah *improvement*

Perbandingan penggunaan kertas antara pencatatan di kertas dan aplikasi. Pada blok berwarna hijau menunjukkan banyaknya kertas yang digunakan dalam sehari untuk mencatat data produksi dengan metode menulisnya di kertas, dengan perincian berikut.

- 1 hari => 3 shift
- 1 shift => 15 mesin (shift 3 = 9 mesin)
- 1 hari => 3 shift \* 15 = 45 – 6 = 39 lembar kertas

Sedangkan 0 lembar menunjukkan pemakaian kertas setelah mencatat data produksi menggunakan

aplikasi di komputer. Sehingga dalam pencatatan di aplikasi tidak perlu lagi menggunakan kertas untuk mencatat data produksi. Namun tetap membutuhkan kertas sebagai *backup* ketika data pada aplikasi tersebut tidak dapat tersimpan.

Pembuatan sistem mempercepat proses kerja pada proses *input* data produksi ke *database*. Perbandingan *cycle time* tersebut bisa dilihat pada gambar berikut.



Gambar 14. Perbandingan hasil *cycle time*

perbandingan hasil *cycle time* antara *input* data yang dilakukan secara manual dan menggunakan aplikasi SSC dalam rentang waktu perhari (3 *shift*). Perbandingan tersebut diambil dari waktu yang paling sering terjadi. Pada balok berwarna biru tertera 120 menit yang merupakan waktu yang diperlukan *data entry* untuk memasukkan (*input*) data ke *database*, nilai 120 menit tersebut diambil dari wawancara ke *data entry*, berikut perhitungannya.

- 1 hari => 3 shift
- 1 shift => 15 mesin (shift 3 = 9 mesin)
- 1 hari => 3 \* 15 – 6 = 39 mesin/hari
- 1 mesin => 1 lembar kertas
- 1 hari => 39 lembar kertas
- Waktu input => 120 menit / 39 = 3.08 = 3 menit/lembar

Sedangkan pada balok yang berwarna *orange* merupakan waktu waktu *loading* yang diperlukan aplikasi untuk menyimpan ke *database* dimulai dari tombol *save* pada aplikasi di tekan sampai datanya berhasil tersimpan, berikut perhitungan rata-ratanya.

- 1 toy => 15 detik
- Rata-rata perhari 2 toy / mesin=> 2 \* 15 detik = 30 detik
- 1 hari => 39 mesin
- 1 hari => 39 \* 30 detik = 1170 detik
- => 1170/60 = 19.5 menit

*Net Quality Income* (NQI) ialah keuntungan bersih yang didapat perusahaan dari sebuah *improvement*. Rumus untuk menghitung NQI yaitu:

$$NQI = \text{Keuntungan} - \text{Biaya Perbaikan}$$

Hasil NQI dapat menunjukkan seberapa besar efek *improvement* yang dilakukan terhadap keuntungan finansial yang diperoleh oleh perusahaan. Di bawah ini

rincian harga pembuatan aplikasi SSC dan metode analisis laporan otomatis pada mesin DIJ.

#### 1. Biaya Perbaikan

Biaya perbaikan merupakan biaya yang dikeluarkan untuk pembuatan *improvement*. Namun pada biaya material atau alat tidak ada karena dari *software* Visual Studio, Excel dan Power BI semuanya *free* serta alat untuk membuat dan pengujian yaitu komputer sudah tersedia di laboratorium (lab) dan *line production*. Sehingga biaya perbaikannya yaitu Rp 0.

#### 2. Cost Reduction dan Benefit

Setelah dibuat *improvement* pembuatan aplikasi SSC dan metode analisis laporan otomatis pada mesin DIJ, terdapat pengurangan penggunaan kertas dan penurunan waktu input data ke *database* yang semula 120 menit menjadi 15 menit, penjelasan lebih lanjut bisa dilihat pada poin Perbandingan Hasil *Cycle Time* Input Data. Hal ini membuat terjadinya *cost reduction / cost saving* yang akan dirincikan sebagai berikut.

##### Penggunaan kertas:

1 hari  $\Rightarrow$  3 shift  
 1 shift  $\Rightarrow$  15 mesin (shift 3 = 9 mesin)  
 1 hari  $\Rightarrow$  3 shift \* 15 = 45 – 6 = 39 lembar kertas  
 1 minggu  $\Rightarrow$  5 hari \* 39 = 195 + 10 (sabtu) = 205 lembar  
 1 tahun  $\Rightarrow$  52 minggu – 4 minggu (libur) = 48 minggu  
 1 tahun  $\Rightarrow$  48 \* 205 = 9840 lembar kertas  
 100 lembar  $\Rightarrow$  Rp 15.000  
 Sehingga:  $9840/100 * \text{Rp } 15.000 = \text{Rp } 1.476.000$  per tahun

##### Input Data:

1 hari  $\Rightarrow$  8 jam kerja  
 Input data SSC  $\Rightarrow$  2 jam  $\Rightarrow$   $2/8 = 1/4$  jam input per hari  
 UMP Kab. Bekasi  $\Rightarrow$  Rp 4.791.843,90 = Rp 4.800.000 / bulan  
 1 bulan  $\Rightarrow$   $1/4 * \text{Rp } 4.800.000 = \text{Rp } 1.200.000$   
 Sehingga:  $\text{Rp } 1.200.000 * 12 \text{ bulan} = \text{Rp } 14.400.000$  / tahun

Total keuntungan = penggunaan kertas + input data  
 $= \text{Rp } 1.476.000 + \text{Rp } 14.400.000$   
 $= \text{Rp } 15.876.000$  per tahun

Dari perhitungan di atas, maka nilai NQI (*Net Quality Income*) yang didapat ialah:

NQI = Total keuntungan - Total biaya *improvement*  
 $= \text{Rp } 15.876.000 - \text{Rp } 0$   
 $= \text{Rp } 15.876.000$  per tahun

## IV. KESIMPULAN

Berdasarkan data dan analisa hasil pada penelitian ini maka dapat disimpulkan, diantaranya adalah:

- Pembuatan aplikasi SSC menggunakan software Microsoft Visual Studio dengan bahasa pemrograman Visual Basic .NET dengan tujuan menggantikan SSC kertas sehingga tidak perlu menggunakan kertas yang dalam sehari membutuhkan 39 lembar kertas menjadi 0 lembar.
- Rumus untuk mengolah data secara otomatis yaitu pengambilan data dari sheet lain pada database excel yang dibuat dengan Microsoft Excel dengan memasukkan rumus di cell-nya dan digabungkan menjadi satu sesuai id (tanggal, mesin dan shift), yang secara otomatis terambil dan terhitung sendiri tanpa harus input manual maupun drag baris atau cell-nya. Sehingga mampu menurunkan cycle time saat input data ke database yang semula dilakukan oleh data entry memerlukan 120 menit menjadi 19,5 menit ketika menggunakan aplikasi.
- Koneksi interface report analysis atau OEE dashboard dengan database excel yaitu bagaimana mengambil data dari interface menggunakan software Microsoft Power BI ke source berupa database excel. Pengambilan data tersebut dilakukan dengan memasukkan path database ke source pada Power BI. Sehingga ketika di refresh pada Power BI berhasil, maka koneksi antara database dan interface berhasil.

## V. DAFTAR PUSTAKA

- [1] A. Alabdullaali, A. Asif, S. Khatoon, and M. Alshamari, "Designing Multimodal Interactive Dashboard of Disaster Management Systems," *Sensors*, vol. 22, no. 11, 2022, doi: 10.3390/s22114292.
- [2] C. Oktafiani, "Sistem Manajemen Basis Data," *Sist. Inf. Manaj. Basis Data*, no. April, p. 18, 2020, [Online]. Available: [https://www.researchgate.net/publication/344662419\\_ARTIKEL\\_SISTEM\\_MANAJEMEN\\_BASI\\_S\\_DATA](https://www.researchgate.net/publication/344662419_ARTIKEL_SISTEM_MANAJEMEN_BASI_S_DATA).
- [3] C. Bakti and H. Kartika, "Perawatan Mesin Dengan Metode Overall Equipment Effectiveness (Oee)," *J. Ilmu Tek. dan Komput.*, vol. 3, no. 1, pp. 31–38, 2019.
- [4] A. Wicaksana, "濟無No Title No Title No Title," *Https://Medium.Com/*, 2016, [Online]. Available: <https://medium.com/@arifwicaksanaa/pengertian-use-case-a7e576e1b6bf>.
- [5] R. R. Saragih, "Pemrograman dan bahasa Pemrograman," *STMIK-STIE Mikroskil*, no. December, pp. 1–91, 2016.



## MENURUNKAN LEADTIME GENERAL OVERHAUL UNIT EXCAVATOR PC1250SP-8R DI PT UT SITE LOA JANAN DENGAN METODE GOH SMART SOLUTION

Vuko T Manurung<sup>a,1</sup>, M Jamaluddien Jumrah<sup>a,2</sup>, Yohanes T. Wibowo<sup>b,3</sup>

<sup>a</sup>Program Studi Mesin Otomotif, Politeknik Astra Jakarta Indonesia

<sup>b</sup>Program Studi Pembuatan Peralatan dan Perkakas Produksi, Politeknik Astra Jakarta Indonesia

E-mail: vuko.manurung@polytechnic.astra.ac.id<sup>1</sup>, jamal.ddien@gmail.com<sup>2</sup>, yohanes.trijoko@polytechnic.astra.ac.id<sup>3</sup>

**Abstrak--General overhaul (GOH) adalah proses overhaul seluruh komponen yang ada di unit dengan tujuan untuk mengembalikan unjuk kerja (performance) yang mendekati atau sama seperti unit yang baru. PT RPP Contractors pertambangan di Kaltim melakukan general overhaul alat beratnya, yakni PC1250SP-8R sebanyak 4 unit di PT UT cabang site Loa Janan. Dalam prosesnya waktu pelaksanaan general overhaul PC1250SP-8R melebihi dari target yang telah disepakati, yakni 45 hari. Proses GOH yang terjadi ini tidak diimbangi dengan adanya sistem manajemen yang terukur dan terencana mulai dari planning, executing, controlling, dan evaluating sehingga target leadtime dalam pengerjaan GOH tidak tercapai. Perbaikan yang dilakukan adalah menggunakan smart (specific measurable achievable realistic timeline) solution management sehingga permasalahan keterlambatan dari sisi dukungan suku cadang (spare part) kesalahan dalam proses pemasangan, kemampuan mekanik baru dan keterlambatan lainnya dapat diatasi dengan baik. Dengan demikian estimasi penyelesaian GOH dengan customer dapat mendekati waktu yang telah ditetapkan.**

**Kata Kunci : General Overhaul, Leadtime, PC1250SP-8R**

### I. PENDAHULUAN

PT UT Tbk, merupakan perusahaan penyedia peralatan alat berat untuk pertambangan dan konstruksi, dalam melaksanakan bisnisnya menerapkan prinsip menjadi mitra bisnis yang handal untuk pelanggan, PT UT memiliki beberapa pilihan layanan program purna jual. Salah satunya adalah program servis. Program servis yang diberikan, antara lain, UT PTA (Program Tukar Alat) Services, UT PAP (Program Analisa Pelumas) Services, UT PPM (Program Pemeriksaan Mesin) Services, UT PMP (Preventive Maintenance Package) Plus, UT PSR (Paket Service Regular) dan UT GOH (General Overhaul) Services. GOH merupakan proses overhaul seluruh komponen yang terorganisir pada unit yang bertujuan untuk mengembalikan performa mesin dengan melakukan perbaikan maupun penggantian komponen agar performa mesin menjadi mendekati baru sehingga mempunyai umur pakai yang lebih panjang, serta dapat meminimalisir kerusakan yang tidak terjadwal (*unscheduled breakdown*).

Penelitian sebelumnya menunjukkan bahwa dengan terjadinya kerusakan yang tidak terjadwal (*unscheduled breakdown*) [1-3], mengakibatkan proses produksi di area pertambangan menjadi terhambat dan merugikan pemilik unit alat berat maupun pengelola pertambangan. Oleh karena itu GOH memastikan agar unit memiliki unjuk kerja yang sama atau mendekati unit yang baru dengan biaya proses perbaikannya yang jauh lebih murah dibandingkan dengan membeli unit baru.

GOH sendiri merupakan proses perbaikan secara menyeluruh sebagai tindakan *preventive maintenance* unit alat berat dengan maksud untuk merekondisi unit sehingga tetap dalam unjuk kerja yang baik dan dapat kembali pada standard spesifikasi pabrik meskipun unit sudah tua, dengan menggunakan kombinasi suku cadang yang digunakan kembali (*reuse*), diperbaiki (*repair*), dan diganti baru (*renew/replace*), dengan biaya yang lebih murah dibandingkan dengan membeli unit baru sehingga dapat mencapai nilai keekonomiannya.

### II. PENGUMPULAN DATA

Gambar 1 merupakan data proses GOH sebelumnya untuk unit Excavator PC1250-8R milik PT RPP periode 2017-2022 dengan total sebanyak 6 unit. Dari data tersebut terlihat bahwa *leadtime* dalam proses pelaksanaan GOH masih cenderung melewati dari target yang telah disepakati antara pihak PT UT dan PT RPP Contractors Indonesia, yakni 45 hari, akan tetapi aktualnya melebihi waktu yang disepakati.

Untuk itu perlu dilakukan perbaikan (*improvement*) sehingga waktu pengerjaan GOH kembali ke waktu yang sudah disepakati dengan pihak customer dapat ditepati.