

Sistem Informasi RAM (Reliability, Availability, Maintainability) Analisis Kinerja Peralatan Gerbang Tol Berbasis Dashboard

Restu Mahardhika*¹, Rakhmayudhi², Santi Purwanti³, Sirajun Nasihin⁴
^{1,4} Sekolah Tinggi Manajemen Ilmu Komputer LIKMI, ^{2,3} Universitas Subang
^{1,4} Jl. Ir. H. Juanda No. 96, Lebakgede, Kecamatan Coblong, Kota Bandung, Indonesia
^{2,3} Jln, R.A Kartini KM 03, Subang, Jawa Barat, Indonesia
 E-mail : *restu.mahardhika23@gmail.com

ABSTRAK- Penulisan yang berjudul Sistem Informasi RAM (*Reliability, Availability, Maintainability*) Analisis Kinerja Peralatan Gerbang Tol Berbasis *Dashboard* adalah sebuah sistem informasi yang dibangun untuk membantu PT. XYZ dalam mengukur kinerja peralatannya, khususnya di Gardu Tol. Dengan dibangunnya sistem ini perusahaan dapat melakukan pemantauan peralatan terkait kehandalan, ketersediaan dan syarat dilakukannya perawatan. Hasil pengujian dari dibangunnya sistem ini menunjukkan bahwa sistem informasi tidak memiliki kendala dari segi pengujian *blackbox*. Selain itu, pengujian Akurasi mengenai penghitungan yang dilakukan sesuai dengan teori yang digunakan. Dalam pengujian yang telah dilakukan mendapat respon yang sangat baik dari pengguna terakhir dengan nilai rata-rata yang didapatkan 84%. Dengan demikian, Sistem informasi ini layak digunakan sebagai alat untuk mengukur kinerja peralatannya.

Kata kunci : Dashboard, Sistem Informasi, RAM Analisis.

1. PENDAHULUAN

RAM (*Reliability, Availability, Maintainability*) analisis adalah suatu tinjauan untuk melakukan perhitungan yang berguna untuk mempresentasikan dan memprediksikan keandalan, ketersediaan dan kemampuan analisa perawatan suatu sistem [1]. Karena metode tersebut memiliki Tujuan yang penting dalam memastikan bahwa alat ada pada kondisi yang diinginkan seperti keselamatan, kinerja, lingkungan, batasan waktu dan tinjauan ekonomis. *Reability, Availability dan Maintainability* (RAM) merupakan bagian dalam suatu sistem yang berhubungan dan tidak bisa dipisahhkan dengan operasionalnya. Dimana *Reliability* untuk menghitung kehandalan suatu peralatan, *availability* yaitu suatu peluang peralatan dapat beroperasi dan *maintability* merupakan syarat waktu pemeliharaan akan dilakukan.

PT XYZ merupakan perusahaan yang bergerak dibidang Jalan Tol. Perusahaan tersebut memiliki 80 gardu yang tersebar di 6 daerah, yaitu: Palimanan, Subang, Kalijati, Cikedung, Sumberjaya dan Kertajati. Unit *Mechanical, Electrical dan Engineering* Divisi Pemeliharaan PT XYZ, memiliki peranan untuk menjaga kehandalan maupun memastikan ketersediaan alat penunjang kegiatan, salah satunya peralatan yang digunakan dalam kegiatan transaksi.

Untuk dapat mengelola data masuk dari berbagai gardu bukan hal yang mudah. Oleh sebab itu, Sistem Informasi berbasis *dashboard* yang dapat mengelola data dan informasi yang dapat digunakan dalam melakukan evaluasi dan memastikan tujuan dari suatu operasi [2]. Sehingga dapat membantu pengambilan

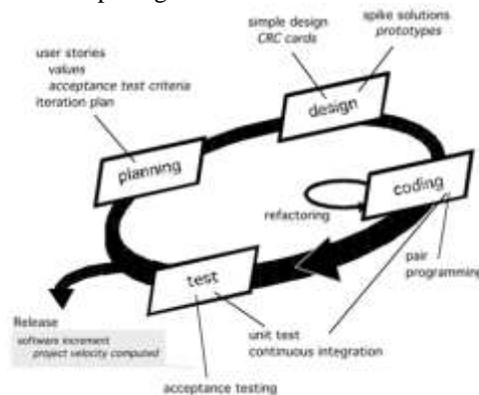
keputusan dalam mengambil strategi yang akan dilakukan.

Pada penelitian ini, Sistem Informasi berbasis Dashboard digunakan untuk monitoring dengan menekankan RAM analisis peralatan pada 6 gerbang tol yang ada. Jenis dashboard pada sistem ini memberikan analisis untuk mendukung kinerja terkait produktifitas peralatan yang ada di setiap gerbang menggunakan *tools* pengembangan *framework Codeigniter* dan Database *MySQL*.

2. ISI PENULISAN

2.1 Metode Pengembangan Sistem

Pada pengembangan sistem yang dilakukan menggunakan metode *Extreme Programming* yang dapat dilihat pada gambar dibawah ini:



Gambar 1. Metode *Extreme Programming*
Extreme Programming (XP) adalah metode rekayasa perangkat lunak dengan pendekatan *object*

oriented dengan sasaran dari metode ini adalah kelompok yang dibentuk dengan skala kecil sampai sedang serta metode ini juga sesuai jika kelompok dihadapkan dengan kebutuhan yang tidak jelas maupun terjadi perubahan kebutuhan yang sangat cepat. [4]

2.1.1 *Planning* (Perencanaan).

Tahap ini adalah tahap pertama dilakukannya pengembangan sistem, dalam tahapan ini penulis melakukannya dengan cara membuat *cause and effect diagram* untuk mengidentifikasi permasalahan melalui sesi *brainstorming* [3]. Selain itu, melakukan analisis kebutuhan dan penetapan jadwal pengembangan sistem.

2.1.2 *Design* (Perancangan).

Tahap ini merupakan perancangan dengan melakukan pemodelan yang dimulai dari pemodelan sistem, perancangan arsitektur sampai dengan perancangan database. Pemodelan sistem menggunakan diagram *Unified Modelling Language* (UML) sedangkan dalam perancangan database menggunakan *Physical Data Model* (PDM).

2.1.3 (Pengkodean).

Tahap *coding* merupakan implementasi dari pemodelan yang telah dilakukan ke dalam bentuk sistem yang dapat dioperasikan. Bahasa Pemrograman yang digunakan adalah PHP dengan menggunakan *framework Codeigniter* dan DBMS MySQL.

2.1.4 *Testing* (Pengujian).

Tahap terakhir yaitu *testing* untuk mengetahui kesalahan yang terjadi pada pengembangan sistem yang telah dilakukan. Metode pengujian adalah *blackbox testing* dan *performance testing*.

2.2 Tinjauan Teori

2.2.1 RAM Analisis

RAM Analisis merupakan pengamatan yang sangat penting dalam memastikan bahwa sistem berada pada tingkat kondisi yang diinginkan seperti tingkat keselamatan (*safety*), performa (*performance*), lingkungan (*environment*), batasan waktu serta tujuan ekonomis. Faktor – faktor yang mempengaruhi pentingnya metode ini antara lain adalah desain sistem, lingkungan dimana sistem pendukung, dan memelihara sistem, melakukan perawatan, serta tujuan diagnosa dan peralatan yang tersedia. Pencapaian tingkat RAM yang khas pada suatu sistem sangat penting berkaitan kesiapan, keselamatan, kesuksesan misi, alokasi biaya, dan dukungan logistik.

a. *Reliability*

Untuk mengukur kehandalan dapat diartikan berupa peluang yang memiliki fungsi dalam waktu yang telah ditentukan. Selain itu, dapat menggunakan waktu rata-rata antara kegagalan (*mean time between failures / MTBF*) [5], yaitu waktu yang diharapkan di antara

perbaikan dan kegagalan komponen, peralatan, proses, atau produk yang berikutnya. Semakin tinggi MTBFnya maka peralatan cenderung reliabel.

$$MTBF = T/R$$

Dimana :

T = Waktu total

R = jumlah kegagalan

Sedangkan untuk peralatan yang tidak bisa diperbaiki istilah dalam *reliability* adalah *Mean Time to failure* (MTTF). Dalam pembangunan aplikasi ini penulis- melakukan perhitungan terkait MTBF yang merupakan peralatan yang dapat diperbaiki.

b. *Availibility*

Ketersediaan adalah peluang suatu sistem/komponen untuk melaksanakan fungsinya berdasarkan periode waktu yang telah ditetapkan dan di maintain sesuai tatacara yang telah ditentukan. Seperti halnya kehandalan, ketersediaan menentukan peluang suatu sistem yang telah dinyatakan dalam suatu dari dua kondisi [6], yaitu :

- *Uptime* (*on*).
- *Downtime* (*off*).

Uptime berarti bahwa sistem masih berfungsi dan *downtime* berarti bahwa sistem tidak berfungsi, dalam hal ini diperbaiki atau diganti tergantung pada apakah sistem dapat diperbaiki atau tidak.

Untuk mengetahui tingkat ketersediaan alat yang digunakan, maka diambil metode penghitungan:

Ketersediaan (*availibility*):

$$\frac{\text{Total waktu} - (\text{waktu perbaikan} + \text{waktu perawatan})}{\text{Total Waktu}} \times 100\%$$

c. *Maintability*

Salah satu parameter *maintainability* yang umum digunakan adalah *Mean Time To Repair* (MTTR). MTTR diperoleh dengan menggunakan persamaan berikut ini [7]:

$$MTTR = \text{Total jam } \textit{breakdown unschedule} / \text{total pemberhentian}$$

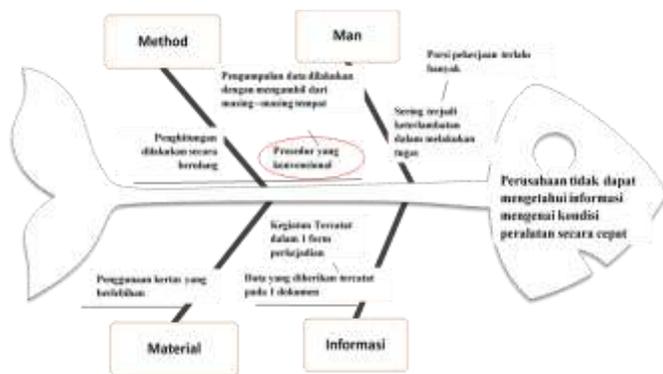
2.2.2 Dashboard

Dashboard merupakan sebuah model aplikasi sistem informasi yang disediakan bagi para manajer untuk menyajikan informasi kualitas kinerja, dari sebuah perusahaan atau lembaga organisasi [8].

2.3 Hasil Penulisan

2.3.1 *Planning* (Perencanaan)

Dari hasil Identifikasi masalah yang telah dilakukan dengan menggunakan *cause and effect diagram*:



Gambar 2. Cause and Effect Diagram

Berdasarkan hasil diatas adalah “Perusahaan tidak dapat mengetahui informasi mengenai kondisi peralatan yang digunakan dan tidak mengetahui waktu perawatan harus dilakukan” yang dikarenakan prosedur/metode yang digunakan masih menggunakan teknik konvensional.

a. Analisis Kebutuhan.

Tabel 1. Kebutuhan Fungsional

Nomor SRS	Deskripsi
Inspektor	
SRS-F-1	Sistem menyediakan login pengguna
SRS-F-2	Sistem menyediakan kelola data pengguna
SRS-F-2.1	Sistem menyediakan proses rubah <i>password</i>
SRS-F-2.2	Sistem menyediakan proses tambah pengguna
SRS-F-2.3	Sistem menyediakan proses hapus pengguna
SRS-F-3	Sistem menyediakan lihat data kejadian
SRS-F-3.1	Sistem menyediakan tampilan detail waktu <i>schedule time</i>
SRS-F-3.2	Sistem menyediakan tampilan waktu <i>Unschedule time</i>
SRS-F-3.3	Sistem menyediakan tampilan waktu <i>standby time</i>
SRS-F-4	Sistem menyediakan tampilan dashboard pengukuran kinerja berupa grafik <i>Schedule, Unscedule, Standby</i> dan table RAM
SRS-F-5	Sistem menyediakan kelola data peralatan
SRS-F-5.1	Sistem menyediakan proses tambah data peralatan
SRS-F-5.2	Sistem menyediakan proses edit data peralatan
SRS-F-5.3	Sistem menyediakan proses hapus data peralatan
Kasubag MEE	
SRS-F-6	Sistem menyediakan login pengguna
SRS-F-7	Sistem menyediakan tampilan <i>Dashboard</i>
KSLT (Operator)	
SRS-F-8	Sistem menyediakan menu login pengguna
SRS-F-9	Sistem menyediakan kelola waktu <i>schedule time</i>
SRS-F-9.1	Sistem menyediakan proses tambah waktu

	<i>schedule time</i>
SRS-F-9.2	Sistem menyediakan proses edit waktu <i>schedule time</i>
SRS-F-9.3	Sistem menyediakan proses hapus waktu <i>schedule time</i>
SRS-F-10	Sistem menyediakan kelola waktu <i>schedule time</i>
SRS-F-10.1	Sistem menyediakan proses tambah waktu <i>schedule time</i>
SRS-F-10.2	Sistem menyediakan proses edit waktu <i>schedule time</i>
SRS-F-10.3	Sistem menyediakan proses hapus waktu <i>schedule time</i>
SRS-F-11	Sistem menyediakan kelola waktu <i>standby time</i>
SRS-F-11.1	Sistem menyediakan proses tambah waktu <i>standby time</i>
SRS-F-11.2	Sistem menyediakan proses edit waktu <i>standby time</i>
SRS-F-11.3	Sistem menyediakan proses hapus waktu <i>standby time</i>

Tabel 2. Kebutuhan Fungsional

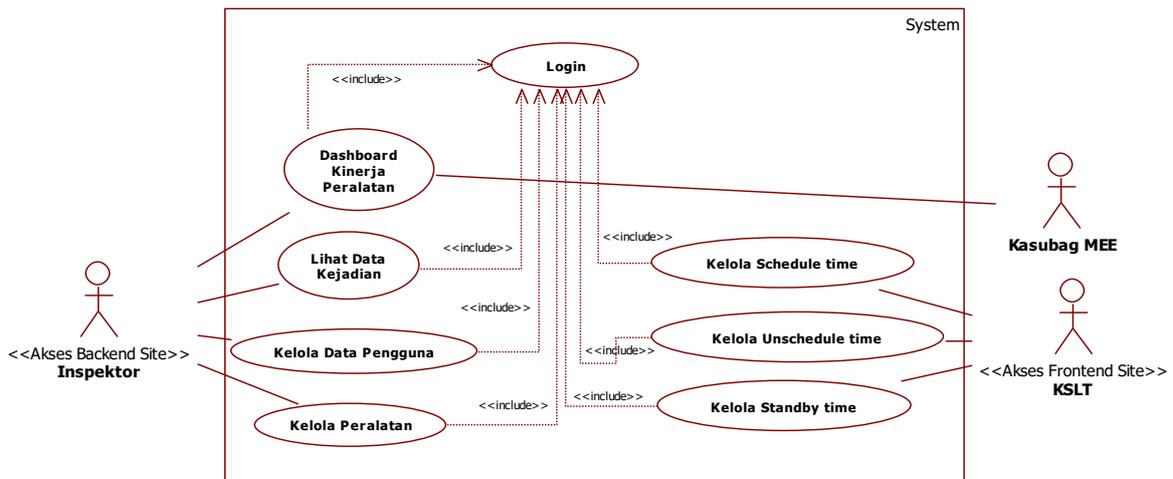
Nomor SRS	Deskripsi
SRS-NF-1	Sistem mempunyai rancangan antar muka yang <i>user-friendly</i>
SRS-NF-2	Sistem yang dibuat berupa <i>website</i> menggunakan <i>framework Codeigniter</i> dan dapat di jalankan di berbagai <i>web browser</i>
SRS-NF-3	Sistem menggunakan enkripsi <i>password</i>
SRS-NF-4	Hanya inspektor yang mempunyai peranan dalam pembuatan user sesuai dengan perannya, mengubah peranan user dan menghapus user
SRS-NF-5	Hanya inspektor yang dapat mengubah <i>password</i> user lain
SRS-NF-6	Sistem terintegrasi dengan setiap gerbang dan kantor cabang

2.3.2 Design (Perencanaan)

Untuk dapat dengan mudah dalam memahami dan mengembangkan sistem ini pada suatu instansi, maka sangat diperlukan untuk mengerti posisi, cara kerja dan rancangan umum dari sistem tersebut.

a. Use Case Diagram.

Tahap ini berguna untuk menggambarkan fungsi yang terdapat dalam sistem. *Use case diagram* memberikan gambaran aktor yang akan mempergunakan sistem dan dalam cara apa aktor tersebut berinteraksi dengan sistem yang akan dikembangkan[9].

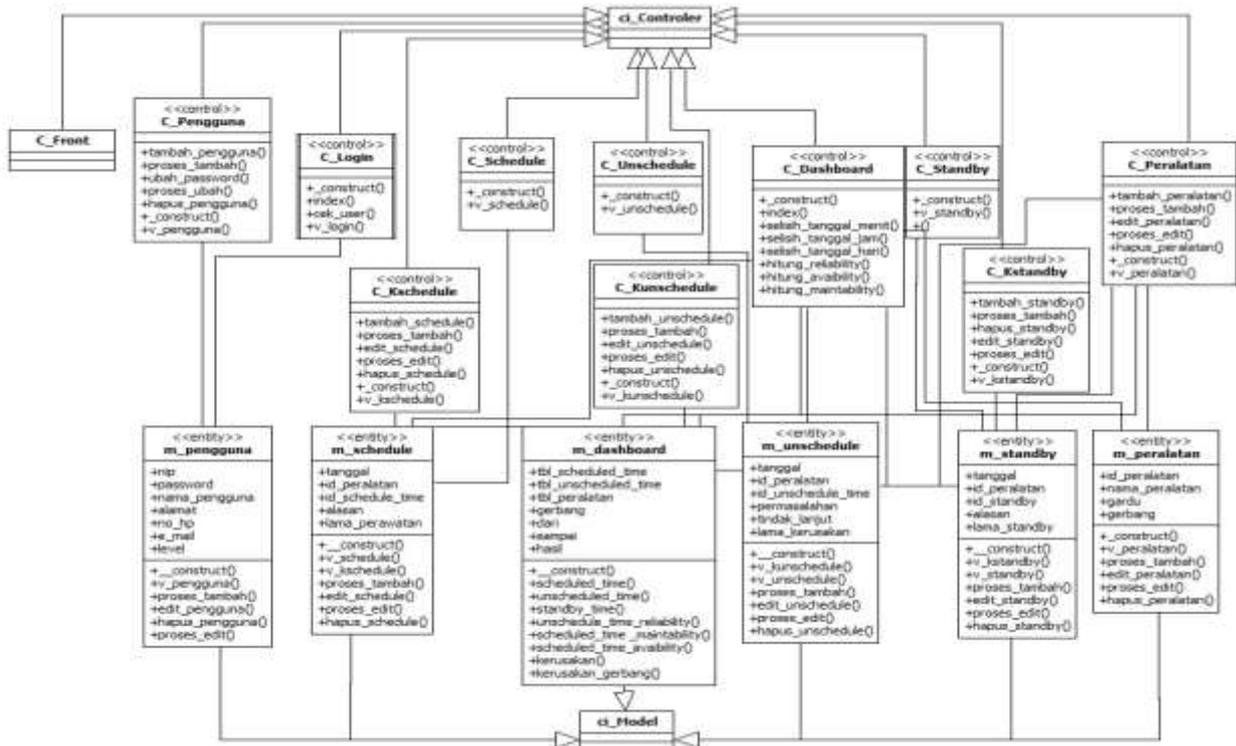


Gambar 3. Usecase Diagram Sistem Informasi RAM Analisis

Usecase yang dirancang terdiri dari 8 dan tiga aktor yang terdiri dari Kasubag MEE yang memiliki kewenangan dalam melihat kinerja peralatan melalui *dashboard*, KSLT merupakan orang yang bertanggungjawab dalam mengelola data peralatan dan Inspektor sebagai super user.

b. Class Diagram.

Tahap ini menjelaskan hubungan dan interaksi antar kelas-kelas yang terdapat pada sistem pendukung keputusan ini. Kelas-kelas tersebut berisi atribut dan fungsi yang dapat digunakan oleh sistem sehingga dapat menyimpan data dan menghasilkan apa yang menjadi tujuan dari terciptanya sistem ini.

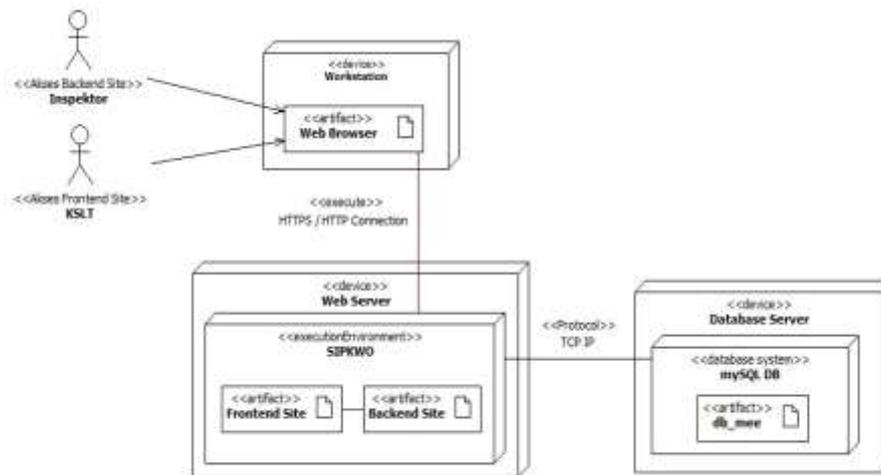


Gambar 4. Class Diagram Sistem Informasi RAM Analisis

Class diagram diatas yang dibuat untuk mendukung dalam melakukan eksekusi pada sistem yang dibuat ditujukan teknik pemrograman dengan konsep MVC (*Model, View dan Control*) dengan penggunaan *framework Codeigniter*. Dengan demikian eksekusi pengkodean dalam melakukan implementasi dapat dilakukan sesuai dengan *class diagram* yang telah dibuat.

c. *Deployment Diagram.*

Tahap ini digunakan untuk mewakili arsitektur eksekusi suatu sistem. Diagram ini memperlihatkan konfigurasi komponen ketika akan dilakukannya proses eksekusi pada sistem [10].

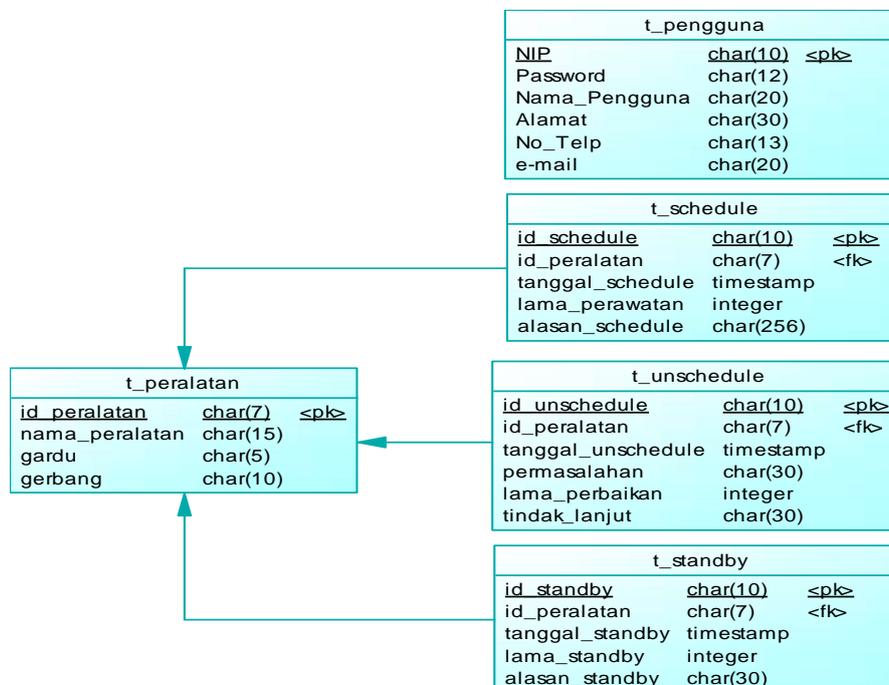


Gambar 5. *Deployment Diagram Sistem Informasi RAM Analisis*

Pada *deployment diagram* diatas menunjukkan perangkat lunak yang berada pada perangkat keras yang digunakan saling terintegrasi. Diagram diatas memperlihatkan ketika perangkat lunak beroperasi pada masing-masing konfigurasi yang berbeda.

4) *Physical Data Model*

Tahap ini adalah membuat model untuk mendeskripsikan data dan hubungan antar data. Setiap tabel memiliki sekumpulan kolom, dan setiap kolom memiliki nama yang unik. Perancangan dan tipe data database fisik *physical data model* dapat dilihat pada gambar dibawah:

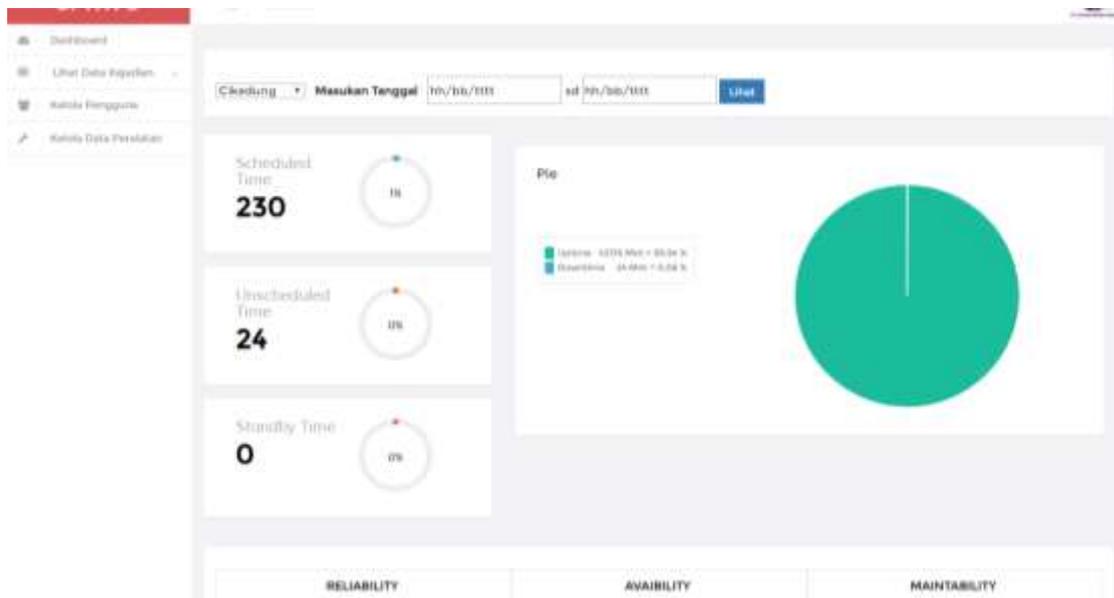


Gambar 6. *Physical Data Model Sistem Informasi RAM Analisis*

Pada gambar diatas terlihat bahwa sistem yang dibuat terdiri dari lima tabel yang kemudian akan diimplementasikan pada *MySQL*. Masing-masing tabel entitas tersebut adalah *t_peralatan*, *t_pengguna*, *t_schedule*, *t_unschedule* dan *t_standby*.

2.3.3 Coding (Pengkodean)

Pada tahap coding ini penulis melakukan pengkodean menggunakan bahasa pemrograman *php* dengan *dbms* menggunakan *MySQL*. Adapun hasil dari tahap ini adalah sebagai berikut:



Gambar 7. Dashboard Sistem Informasi RAM Analisis

Gambar diatas menunjukkan hasil dari implementasi perancangan yang dilakukan. *Dashboard* yang dibuat memiliki fungsi penghitungan yang sudah sesuai dengan kebutuhan perusahaan untuk mengukur kinerja peralatan yang berada pada gerbang tol.

2.3.4 Testing (Pengujian)

Pada tahap ini dilakukan pengujian pada sistem ini yang bertujuan untuk mengetahui pengembangan perangkat lunak yang dilakukan memenuhi persyaratan.

a. Pengujian *Blackbox*

Tabel 3. Hasil Pengujian *Blackbox*

Requirement	Hasil Pengujian	
SRS-F-1	Pengguna dapat masuk sesuai dengan hak akses yang diberikan	ok
SRS-F-2	Inspektur dapat melakukan pengelolaan pengguna, sekaligus menentukan hak akses pada sistem	ok
SRS-F-2.1	Inspektur dapat melakukan perubahan <i>password</i> pada semua pengguna	ok
SRS-F-2.2	Inspektur dapat melakukan penambahan pengguna terkait hak akses pada sistem	ok
SRS-F-2.3	Inspektur dapat melakukan hapus pengguna	ok
SRS-F-3	Inspektur dapat melihat data kejadian yang telah di inputkan di halaman <i>frontend</i>	ok

SRS-F-3.1	Inspektur dapat melakukan melihat data kejadian <i>schedule time</i> yang telah di inputkan di halaman <i>frontend</i>	ok
SRS-F-3.2	Inspektur dapat melakukan melihat data kejadian <i>unschedule time</i> yang telah di inputkan di halaman <i>frontend</i>	ok
SRS-F-3.3	Inspektur dapat melakukan melihat data kejadian <i>standby time</i> yang telah di inputkan di halaman <i>frontend</i>	ok
SRS-F-4	Inspektur dapat melihat data kerusakan alat, gardu dan <i>reliability</i> , <i>avaibility</i> serta <i>maintability</i> peralatan	ok
SRS-F-5	Inspektur dapat melakukan kegiatan pengelolaan peralatan	ok
SRS-F-5.1	Data peralatan tersimpan di database	ok
SRS-F-5.2	Data peralatan berhasil dirubah database	ok
SRS-F-5.3	Data berhasil peralatan berhasil dihapus di database	ok
SRS-F-9	KSLT dapat melakukan kegiatan pengelolaan <i>schedule time</i>	ok

SRS-F-9.1	Data <i>schedule time</i> tersimpan di database	ok
SRS-F-9.2	Data <i>schedule time</i> berhasil dirubah database	ok
SRS-F-9.3	Data <i>schedule time</i> berhasil dihapus di database	ok
SRS-F-10	KSLT dapat melakukan kegiatan pengelolaan <i>unschedule time</i>	ok
SRS-F-10.1	Data <i>unschedule time</i> tersimpan di database	ok
SRS-F-10.2	Data <i>unschedule time</i> berhasil dirubah database	ok
SRS-F-10.3	Data <i>unschedule time</i> berhasil dihapus di database	ok
SRS-F-11	KSLT dapat melakukan kegiatan pengelolaan <i>standby time</i>	ok
SRS-F-11.1	Data <i>standby time</i> tersimpan di database	ok
SRS-F-11.2	Data <i>standby time</i> berhasil dirubah database	ok
SRS-F-11.3	Data <i>standby time</i> berhasil dihapus di database	ok

b. Pengujian Performance

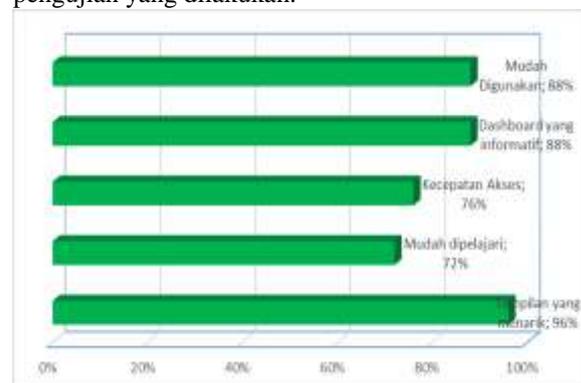
Tabel 4. Hasil Pengujian Performance

No	Butir pengujian	Objek	
1	Melihat kerusakan alat pada gerbang	Informasi Alat 	ok
2	Melihat gardu yang sering mengalami kerusakan alat	Informasi Gardu 	ok
3	Melihat scheduled time pada gerbang yang dipilih	Scheduled time 270 	ok
4	Melihat unscheduled time pada gerbang yang dipilih	Unscheduled Time 124 	ok
5	Melihat produktivas gerbang terkait persentasi uptime dan downtime	Produktivitas Gerbang 	ok

6	Melihat kehandalan peralatan	RELIABILITY 	ok
7	Melihat ketersediaan peralatan	AVAILABILITY 	ok
8	Melihat syarat dilakukannya perawatan alat	MAINTABILITY 	ok

c. Pengujian Beta

Pengujian beta yang dilakukan adalah melakukan penyebaran kuisioner kepada pengguna sistem sebanyak 10 orang dengan pengukuran menggunakan skala *Likert*. Pengujian ini bertujuan untuk mengetahui dan menilai dari segi pengoperasian sistem. Berikut adalah hasil pengujian yang dilakukan.



Gambar 8. Grafik Hasil Pengujian Beta

Dari hasil pengujian yang telah dilakukan, sistem yang dibuat memiliki respon yang baik dari pengguna sistem dengan kriteria Tampilan yang Menarik dengan nilai 96%, Mudah digunakan dan *Dashboard* yang Informatif memiliki nilai 88%, Kecepatan Akses 76% dan Mudah dipelajari. Dengan demikian sistem respon dari pengguna memiliki nilai 84% yang berarti sistem sangat baik untuk digunakan.

3. KESIMPULAN

Berdasarkan pembahasan pada analisis kebutuhan, perancangan implementasi dan pengujian yang telah dilakukan, maka dapat ditarik kesimpulan yaitu:

1. Sistem Informasi RAM (*Reliability, Availability, Maintainability*) Analisis Kinerja Peralatan Gerbang Tol Berbasis Dashboard yang dilakukan di PT. XYZ berhasil dibangun yang berguna untuk mengetahui kondisi peralatan yang ada di dalam gardu tol.
2. Dengan adanya sistem ini, dapat mempercepat proses pengerjaan dalam pengumpulan data dan proses penghitungan kinerja peralatan oleh karyawan yang bersangkutan serta berimplikasi pada pengambilan keputusan juga pelaporan yang lebih cepat.
3. Dengan adanya sistem ini, penyimpanan data yang telah dikumpulkan dari masing – masing gerbang tol lebih tertetukur sehingga dapat memudahkan dalam pencarian data apabila diperlukan.

4. PENUTUP

Kesadaran penulis dalam pembuatan sistem ini masih banyak kekurangan, untuk penyempurnaan sistem ini dapat dilakukan. Adapun penyempurnaan sebagai berikut.

1. Sistem ini di integrasikan dengan sistem informasi pengelolaan asset peralatan yang ada, sehingga pengelolaan peralatan dapat terpantau dengan baik.
2. Sistem ini dapat diimplementasikan pada pengelolaan seluruh aset elektronik yang ada di lingkungan perusahaan.

Demikian saran yang dapat penulis berikan, semoga saran tersebut bisa dijadikan sebagai rujukan untuk penulis khususnya untuk PT. XYZ.

DAFTAR PUSTAKA

- [1] A. F. Yusra, "Analisis Performance Mesin Weaving Pada PT. ABC Menggunakan Metode Reliability Availability Maintainability (RAM) Dan Overall Equipment Effectiveness (OEE)," in *e-Proceeding of Engineering.*, , 2018.
- [2] R. Rajagukguk, *Panduan Praktis SharePoint Server 2010*, Jakarta: PT. Elex Media Computindo, 2010.
- [3] D. Krisbiantoro, A. Azis and Z. Mustaqiem, "Implementasi XP Programming Pada Sistem Informasi Rukun Tetangga Berbasis Website Menggunakan Framework Codeigniter," *Jurnal Akrab Juara*, vol. Volume 4 Nomor 4 Edisi November, pp. (249-258, 2019).
- [4] N. R. Tague, *The Quality Toolbox*, Milwaukee, Wis.: ASQ Quality Press,, 2005.
- [5] I. N. Daulay, S. S. Nurutami and D. D. Daniel, "Analisis Maintenance Reliability terhadap

MTBF (Mean Time Between Failures) Facilities pada Industri Pulp & Paper," Jurnal Ekonomi, vol. Volume 21 Nomor 4, pp. 1-18, 2013.

- [6] N. F. Alamie, *Analisis Optimalisasi Tingkat Operasional (Availability) Pesawat C-130 Hercules Versi Militer*, Bandung: ITB FTMD - Teknik Dirgantara, 2007.
- [7] S. D. Mutiara, A. Rahman and I. Hamdala, "Perencanaan Preventive Maintenance Komponen Cane Cutter I Dengan Pendekatan Age Replacement," *Jurnal Rekayasa dan Manajemen Sistem Industri*, vol. Vol 2 Nomor 2, pp. 396-405, 2014.
- [8] R. L. Salam, *BI Platform Users Survey, 2011: Customers Rate their BI Platform Functionality.*, available at <http://bit.ly/GY3XgB.>, 2011.
- [9] Whitten, L. Jeffery, Bentley, D. Loney, Ditman and D. Kevin, *Metode desain dan analisis sistem*, Yogyakarta: Andi, 2004.
- [10] R. A. Sukanto and M. Salahuddin, *Rekayasa perangkat lunak : Terstruktur dan berorientasi objek*, Informatika: Bandung, 2013.