

SISTEM PELAPORAN INSPEKSI DAN PERAWATAN VENTEX EXPLOSION ISOLATION VALVE BERBASIS ANDROID

¹Agung Yoke Basuki, ²Imelda Uli Vistalin Simanjuntak, ³Bagas Febryan Dwi Cahyo

^{1,2,3}Fakultas Teknik Elektro Universitas Mercu Buana,

Jl. Meruya Selatan Kembangan Jakarta Barat 11650 DKI Jakarta

¹agung.yoke@mercubuana.ac.id, ²imelda.simanjuntak@mercubuana.ac.id,

bagasfdc@gmail.com

Abstrak

PT X ditunjuk sebagai perusahaan representatif untuk unit Ventex Explosion Isolation Valve dari Rico Sicherheitstechnik AG dalam melakukan aktivitas jasa inspeksi dan perawatan secara berkala. Akan tetapi karena sistem pelaporan masih dilakukan secara manual, maka sering terjadinya malfungsi yang akhirnya menyebabkan kerugian pada perusahaan. Berdasarkan pelaporan pekerjaan inspeksi dan perawatan unit Ventex Explosion Isolation Valve pada salah satu pelanggan PT X sepanjang tahun 2020 – 2021, terdapat selisih waktu antara pekerjaan inspeksi dan perawatan dengan tanggal pelaporan pekerjaan tersebut sebesar 1 sampai dengan 7 hari. Oleh karena itu, perlunya dilakukan pelaporan dengan waktu yang lebih singkat untuk mempercepat pengambilan keputusan terhadap tindakan selanjutnya, sehingga kerugian dapat diminimalisir. Penelitian ini membuat pelaporan otomatis hanya menggunakan smartphone dan menyimpan hasil laporan pada sistem database menggunakan Android Studio. Setelah diimplementasikan aplikasi pembuatan laporan inspeksi dan perawatan ventex otomatis, durasi pelaporan menjadi lebih efisien dan hanya membutuhkan waktu sekitar 2 – 3 menit. Pengiriman dan penerimaan notifikasi laporan inspeksi dan perawatan ventex menjadi lebih cepat dari yang minimal membutuhkan waktu sehari menjadi 3 – 4 menit. Hal ini terbukti berhasil meminimalisir waktu pelaporan menjadi jauh lebih signifikan dan kerugian yang disebabkan terlambatnya mengambil keputusan selanjutnya dapat ditekan.

Kata Kunci: *Android, Inspeksi dan Perawatan, Ventex Explosion Isolation Valve*

Abstract

PT X has been assigned as the representative company for Rico Sicherheitstechnik AG's Ventex Explosion Isolation Valve unit, in charge of regular inspection and maintenance services. However, because the reporting system is still done manually, malfunctions frequently occur, which causes losses for the company. Based on the reporting of inspection and maintenance work on the Ventex Explosion Isolation Valve unit for one of PT X's clients from 2020 - 2021, there is a time difference of 1-7 days between the inspection and maintenance work and the reporting date. To accelerate decision-making and minimize losses, there is a need to shorten the reporting time for inspection and maintenance work on the Ventex Explosion Isolation Valve unit. This research creates an automatic reporting system using a smartphone, with report results stored in a database system developed using Android Studio. Once the Ventex automatic inspection and maintenance report application is implemented, reporting duration is reduced to 2-3 minutes. The sending and receiving of notifications for ventex inspection and maintenance reports now only takes 3-4 minutes, as opposed to the previous minimum of a day. The successful implementation of this system has significantly minimized reporting time and reduced losses caused by delays in making subsequent decisions.

Keywords: *Android, Inspection and Maintenance, Ventex Explosion Isolation Valve*

PENDAHULUAN

PT X ditunjuk sebagai perusahaan representatif untuk unit *Ventex Explosion Isolation Valve* dari Rico Sicherheitstechnik AG dalam melakukan aktivitas jasa inspeksi dan perawatan secara berkala. Unit *ventex* ini berfungsi sebagai proteksi pada mesin maupun lingkungan sekitar jika terjadi ledakan dalam proses produksi yang sedang berlangsung. *Ventex* umumnya dipasang pada jalur udara masuk dan jalur udara keluar mesin-mesin industri yang memiliki resiko ledakan dalam proses produksinya, salah satu contohnya adalah mesin granulasi. Untuk mencegah terjadinya malfungsi atau abnormalitas pada kinerja *ventex*, maka dilakukan tahapan inspeksi dan perawatan berkala yang dilakukan oleh teknisi yang telah dilatih oleh pihak Rico Sicherheitstechnik AG, dalam hal ini dilakukan oleh tim *service* PT X.

Saat inspeksi dan perawatan berkala, diperlukan pembuatan laporan yang akan

digunakan sebagai bukti pekerjaan telah dilakukan. Laporan ini juga dapat digunakan sebagai dasar pengambilan langkah selanjutnya, misalnya dalam pemesanan *spare part* untuk periode perawatan berikutnya. Berdasarkan pelaporan pekerjaan inspeksi dan perawatan unit *Ventex Explosion Isolation Valve* pada salah satu pelanggan PT X sepanjang tahun 2020-2021, diperoleh data seperti pada Tabel 1.

Berdasarkan selisih waktu pada Tabel 1, tim *service* PT X telah melakukan pekerjaan inspeksi dan perawatan unit *Ventex Explosion Isolation Valve* sejumlah 2 kali pada setiap *line* produksi PT X, dan terdapat selisih waktu minimal 1-7 hari karena masih dilakukan secara manual. Hal ini menjadikan waktu pengambilan keputusan selanjutnya menjadi lebih lamban dan menyebabkan kerugian terhadap perusahaan. Oleh karena itu, durasi ini diharapkan menjadi lebih cepat dengan pelaporan otomatis agar pengambilan keputusan selanjutnya menjadi lebih cepat dan efisien.

Tabel 1. Jadwal Pekerjaan dan Pelaporan Inspeksi dan Perawatan Unit *Ventex Explosion Isolation Valve* Tahun 2020-2021.

Nomor Line	Jadwal Inspeksi	Pelaporan Inspeksi	Selisih Waktu	Jadwal Perawatan	Pelaporan Perawatan	Selisih Waktu
Line 4	3-5/02/2020	7/02/2020	2 hari	2-3/03/2020	4/03/2020	1 hari
	2/11/2020	3/11/2020	1 hari	4/11/2020	5/11/2020	1 hari
Line 5	3-5/02/2020	7/02/2020	2 hari	2-3/03/2020	4/03/2020	1 hari
	29/11/2020	6/12/2020	7 hari	4-5/01/2021	10/01/2021	5 hari
Line 6	3-5/02/2020	7/02/2020	2 hari	9/03/2020	11/03/2020	2 hari
	19/11/2020	22/11/2020	3 hari	6/01/2021	8/01/2021	2 hari

Penelitian [1], [2] mendesain sistem pelaporan proses pengecekan mesin, data pengecekan yang telah diverifikasi menggunakan bahasa pemrograman HTML, CSS, PHP, Javascript, Java dan *database* MySQL. Tampilan aplikasi menggunakan *template* SB Admin dengan dukungan Bootstrap. Kemudian penelitian [3], [4], [5] mendesain sistem penyimpanan *database spare part* dalam aplikasi Android menggunakan *framework* Android studio dan Firebase sebagai *database server*.

Penelitian [6], [7], [8], [9] mendesain sistem pelaporan menggunakan pemodelan *Unified Modelling Language* (UML) yang berorientasi pada objek. Proses pemodelan UML ini dimulai dengan menganalisis kebutuhan sistem dan diimplementasikan pada *usecase diagram* lalu dilanjutkan dengan membuat *class diagram*, *sequence diagram*, dan *activity diagram*. Penelitian [10] mendesain sistem informasi layanan tugas akhir mahasiswa, sehingga mahasiswa dapat dengan mudah memantau proses yang berjalan melalui telepon genggam secara *real time* dengan memanfaatkan *website*. Kemudian penelitian [11] mendesain suatu sistem pembelajaran untuk mendukung proses belajar mengajar jarak jauh siswa secara mandiri. Penelitian ini menggunakan metode R&D (*Research and Development*) dan model ADDIE yang disederhanakan menjadi ADD.

Dengan menggunakan sistem pelaporan berbasis Android diharapkan setiap pekerjaan dapat dilaporkan secara *real time*. Hal tersebut dapat mempercepat pengambilan

keputusan terhadap tindakan selanjutnya, yaitu penjadwalan perawatan berkala dan pemesanan suku cadang. Selanjutnya, dengan adanya sistem pelaporan berbasis Android, laporan-laporan inspeksi dan perawatan dapat dikelola dengan baik dalam sistem *database* yang telah terhubung. Hal ini dapat mempermudah proses pelacakan laporan yang terdahulu.

Kebaruan penelitian didapatkan dari mengidentifikasi masalah, hipotesis, perumusan metode, pengumpulan data-data, melakukan aplikasi pelaporan otomatis dan analisis hingga menarik Kesimpulan yang sebelumnya masih dilakukan secara manual menjadi otomatis dan terbukti dapat mempercepat proses pengambilan keputusan untuk meminimalisir kerugian perusahaan.

Sistem yang didesain terstruktur dan sistematis menggunakan Android Studio dan Firebase, kemudian setiap laporan selesai prosesnya mengirimkan notifikasi kepada setiap pengguna yang berwenang. Dengan adanya sistem pelaporan pekerjaan inspeksi dan perawatan unit *Ventex Explosion Isolation Valve* berbasis Android ini, diharapkan dapat memenuhi kebutuhan dan standar sistem pelaporan dalam durasi dan pengaruhnya dalam mengurangi kerugian perusahaan.

METODE PENELITIAN

Diagram Blok Sistem

Diagram blok sistem adalah gambaran alur pelaporan dari sistem pelaporan otomatis yang diusulkan. Diagram ini bertujuan untuk

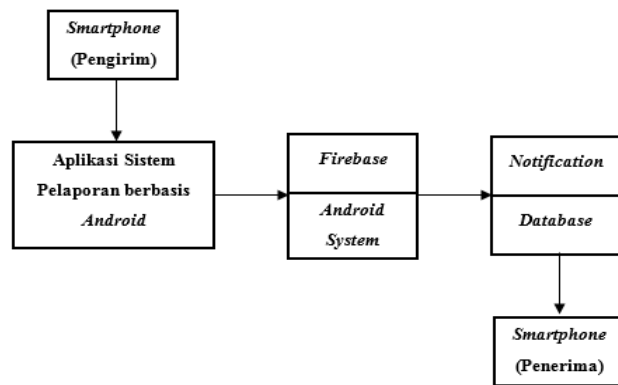
mempermudah proses pemahaman cara kerja sistem yang diusulkan. Gambar 1 adalah gambaran diagram blok sistem dari “Sistem Pelaporan Pekerjaan Inspeksi dan Perawatan Ventex Explosion Isolation Valve berbasis Android.

Pengiriman laporan menggunakan aplikasi sistem pelaporan yang sudah terinstal di *smartphone*, memasukkan, dan mengirim seluruh data progres pekerjaan, sebagai data pengambil keputusan proses

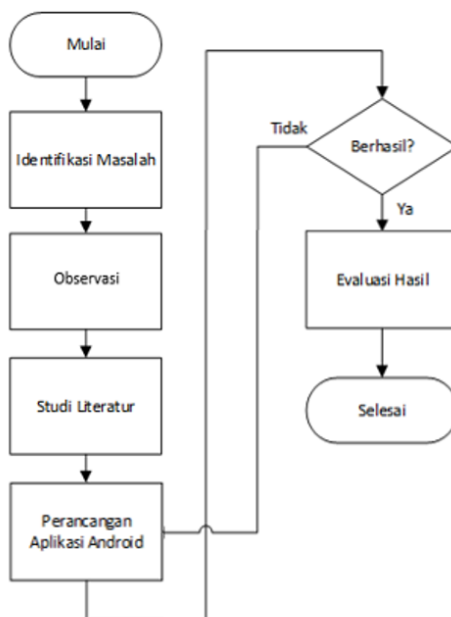
selanjutnya.

Diagram Alir Penelitian

Pelaksanaan penelitian ini dilaksanakan sesuai dengan tahapan yang dijelaskan sesuai dengan diagram alir berdasarkan Gambar 2 yang meliputi identifikasi masalah, observasi alat, studi literatur, diskusi, perancangan aplikasi Android, penyusunan step permasalahan *fault*, evaluasi hasil, dan kesimpulan dan saran.



Gambar 1. Diagram Blok Sistem



Gambar 2. Diagram Alir Penelitian

Tahapan yang digunakan untuk menyelesaikan masalah pada penelitian ini adalah mengidentifikasi masalah mulai dari penyebab yang mungkin terjadi hingga dampak yang akan diberikan, kemudian observasi dilakukan dengan pengamatan langsung pada sistem pelaporan manual, pengamatan pada tahap dokumentasi, proses pembuatan laporan, serta informasi-informasi apa saja yang perlu dicantumkan di dalam laporan pekerjaan. Selanjutnya dilakukan studi literatur sebagai referensi tinjauan pustaka terkait permasalahan serupa, baik dari internet dan buku yang meneliti perancangan dan pembuatan aplikasi melalui Android.

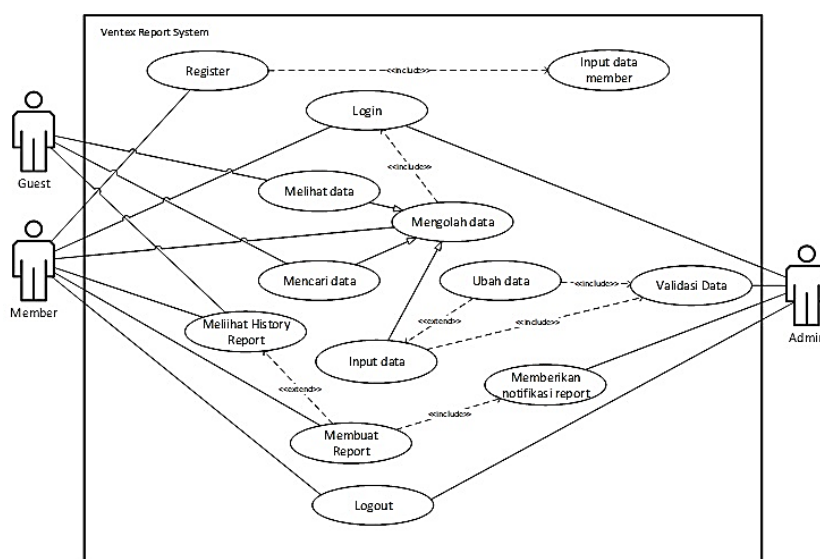
Tahap selanjutnya pembuatan aplikasi Android dilakukan menggunakan Android Studio sebagai alat *input* dan pemrosesan data yang berfungsi untuk membuat laporan pekerjaan inspeksi dan perawatan unit *Ventex Explosion Isolation Valve*. Setelah pembuatan dan uji coba simulasi selesai dilaksanakan, maka data hasil pengujian

sistem pelaporan otomatis dikaji dan evaluasi untuk menentukan tingkat keberhasilan penelitian ini.

Tahapan Penelitian

Proses tahapan *use case diagram* yang digunakan dalam penelitian ini dapat dilihat pada Gambar 3.

Pada Gambar 3 merupakan diagram *use case* [12] yang digunakan pada penelitian ini, terdapat 3 aktor yang terlibat yaitu *guest*, *member*, dan *admin*. *Guest* merupakan pengguna yang dapat membuka aplikasi tanpa perlu mengisi data pada saat *login* dan hanya dapat melihat isi dari aplikasi tanpa bisa memasukkan data. *Member* merupakan pengguna yang telah mengisi data untuk melakukan *register* dan *login*, *member* dapat melihat serta menambahkan data pada aplikasi dan setiap data yang dimasukkan perlu validasi. Validasi dilakukan oleh *admin* yang bertanggung jawab untuk menampilkan data pada aplikasi.



Gambar 3. Use Case Diagram

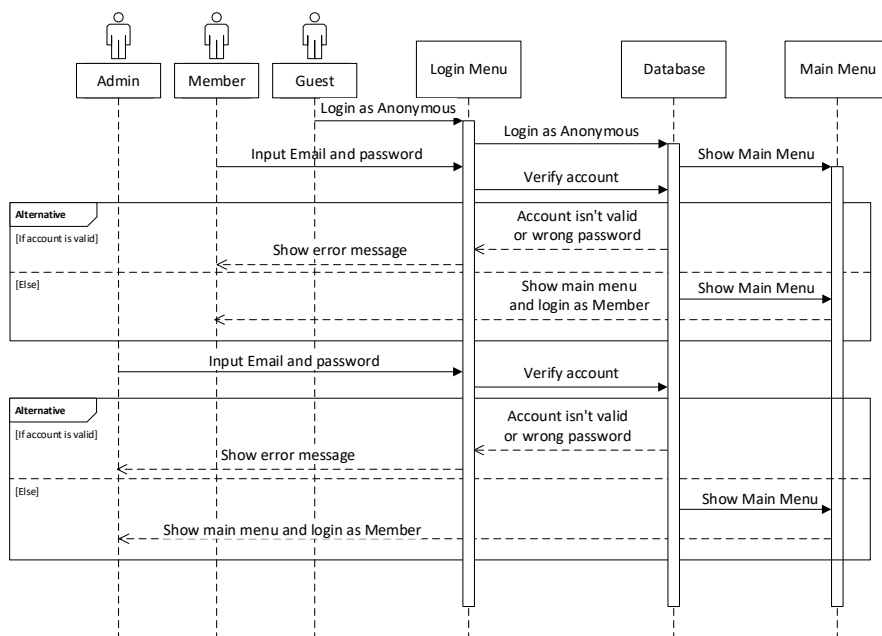
Pada sistem ini terdapat *use case register* yang berfungsi untuk mendaftarkan pengguna baru agar dapat *login* ke dalam aplikasi. Pengguna yang telah *login* dan terbaca sebagai *member* atau *admin* dapat mengolah data yang memiliki *use case* khusus seperti memantau, mencari serta memasukkan data, di mana data yang telah dimasukkan akan divalidasi oleh *admin* dan *member* yang berwenang untuk selanjutnya mengirimkan notifikasi laporan.

Sequences Diagram

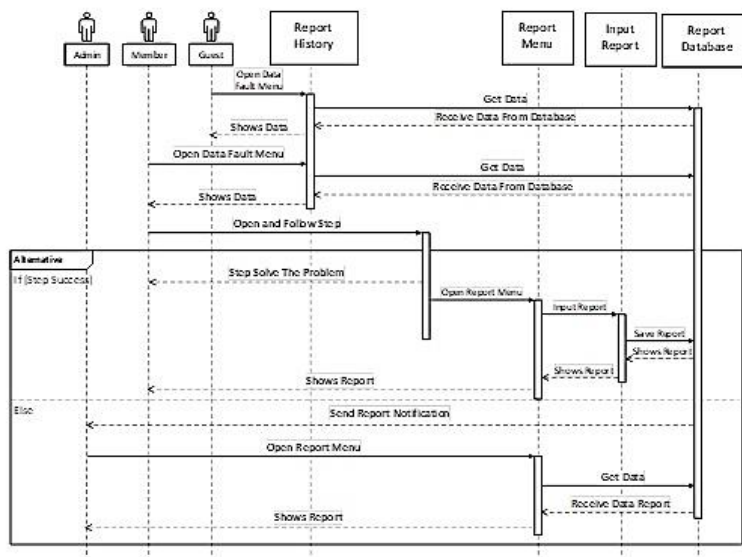
Tahap selanjutnya, untuk menjelaskan bagaimana proses sistem secara runtut dan terarah dalam penggunaannya, dijelaskan dengan *sequences diagram* [13] pada Gambar 4.

Diagram *sequences* pada Gambar 4 menjelaskan tentang urutan langkah yang dilakukan oleh setiap pengguna dalam proses *login* ke dalam aplikasi, untuk *login* sebagai *guest*, pengguna hanya perlu memilih *login as anonim* pada menu *login*. *Admin* atau *member* harus memasukkan *email* dan *password* sesuai dengan status akun pengguna yang kemudian akan divalidasi oleh *database*.

Proses validasi memiliki 2 (dua) alternatif hasil, jika data akun tersedia pada *database* maka tampilan akan langsung tertuju pada menu utama, tetapi jika data dari pengguna tersebut tidak tersedia, maka *database* akan memberikan informasi tentang penolakan akun tersebut dalam menu *login*.



Gambar 4. *Sequences Diagram* dalam Menu *Login*



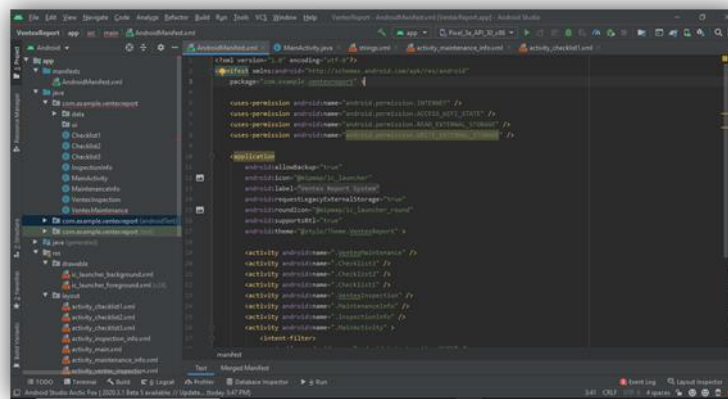
Gambar 5. *Sequences Diagram* dalam Melakukan *Input* Laporan

Diagram *sequences* pada Gambar 5 menggambarkan alur proses dalam aplikasi sistem pelaporan setelah *login* dan kemudian melakukan pelaporan pekerjaan. *Guest* hanya dapat masuk ke menu data laporan terdahulu yang datanya diambil dari *database* dan ditampilkan pada tampilan pengguna. Sama seperti *guest*, *member* memiliki langkah yang sama dalam membuka menu data laporan terdahulu, namun ketika mengikuti langkah dalam data tersebut maka *member* dapat membuat laporan terbaru, di mana data tersebut nantinya akan disimpan pada *database*. Setelah data laporan baru disimpan pada *database*, *admin* akan menerima notifikasi tentang laporan yang baru saja ditambahkan dan dapat menindaklanjuti laporan tersebut.

HASIL DAN PEMBAHASAN

Hasil Perancangan Sistem Pelaporan dengan Android Studio

Perancangan dimulai dengan menyusun program untuk perintah dan informasi yang dibutuhkan agar aplikasi sistem pelaporan yang dibuat dapat bekerja dengan baik, bahasa pemrograman yang digunakan adalah bahasa pemrograman Java dan bahasa pemrograman Kotlin. Namun, pada umumnya bahasa pemrograman Java lebih banyak digunakan ketika seseorang melakukan pemrograman aplikasi Android [14], [15], [16], [17]. Gambar 6 adalah salah satu proses pemrograman aplikasi sistem pelaporan yang dilakukan.



Gambar 6. Tampilan Pemrograman di Android Studio



Gambar 7. Tampilan Menu Utama pada Aplikasi Sistem Pelaporan

Gambar 6 adalah tampilan pengguna aplikasi sistem pelaporan dengan bahasa XML. Pembuatan tampilan pengguna dimulai dengan membuat tampilan *login* dan *register* untuk masuk ke dalam aplikasi. Setiap pengguna melakukan *login* menggunakan akun Google untuk *member* dan *form* untuk memasukkan *email* dan *password* sebagai autentikasi *admin* atau *member* yang akan *login*. Pengguna baru yang belum memiliki akun, dapat melakukan *register* dengan

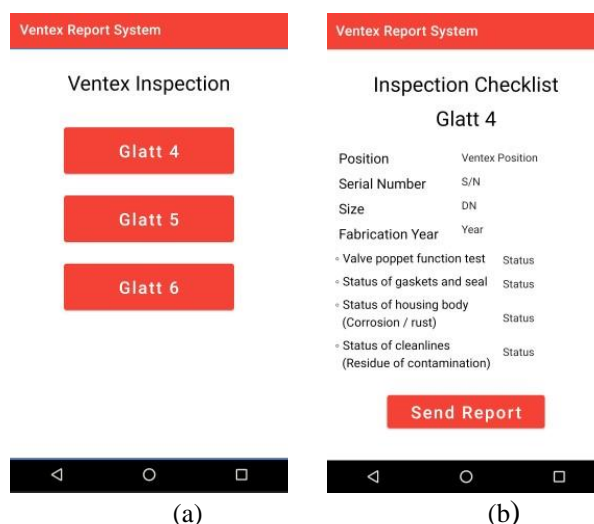
memasukkan beberapa data personal yang diperlukan.

Setelah pengguna melakukan *login* dan telah memasukkan *email* dan *password* dengan benar, pengguna akan langsung masuk ke dalam tampilan menu utama aplikasi sistem pelaporan seperti pada Gambar 7. Di dalam menu utama, terdapat dua jenis pilihan laporan yang dapat dibuat, yaitu laporan inspeksi dan laporan perawatan.

Ketika pengguna memilih tombol *inspection report* maka tampilan akan langsung diteruskan menuju menu informasi laporan inspeksi, namun jika memilih tombol *maintenance report* maka akan diteruskan menuju menu informasi laporan perawatan. Pada menu informasi laporan tersebut, berisi isian data yang perlu dilengkapi sebagai penjelasan mengenai pekerjaan yang dilakukan, seperti nama perusahaan pelanggan, kontak pelanggan, dan nama supplier mesin. Di dalam menu ini juga terdapat tombol *create report* yang dapat meneruskan pengguna ke menu selanjutnya, di mana pada menu informasi laporan inspeksi akan menuju menu *inspection checklist* dan pada menu informasi laporan perawatan akan menuju *menu maintenance details*.

Gambar 8 adalah *menu inspection*

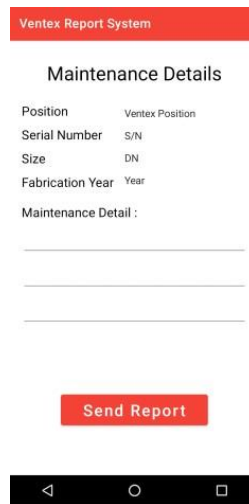
checklist, terdapat beberapa kolom informasi yang perlu diisi dan dilengkapi terkait dengan pekerjaan inspeksi yang dilakukan di lapangan. Pada kolom status, dapat dijelaskan status dari poin-poin yang perlu diinspeksi, apakah kondisinya baik atau tidak, dalam aplikasi sistem pelaporan diinisialkan dengan "OK" dan "Not OK". Kemudian pada menu ini terdapat tombol *send report* untuk dapat mengirimkan laporan inspeksi yang telah dibuat. Laporan yang telah dibuat, selanjutnya akan diteruskan kepada pengguna aplikasi sistem pelaporan yang lainnya dan tersimpan di dalam *database* yang telah tersedia. Pada menu utama terdapat dua tombol utama yaitu *inspection report* dan *maintenance report*, kemudian jika memilih *maintenance report* dan melanjutkan pengisian informasi laporan perawatan seperti pada Gambar 9 menuju ke *menu maintenance details*.



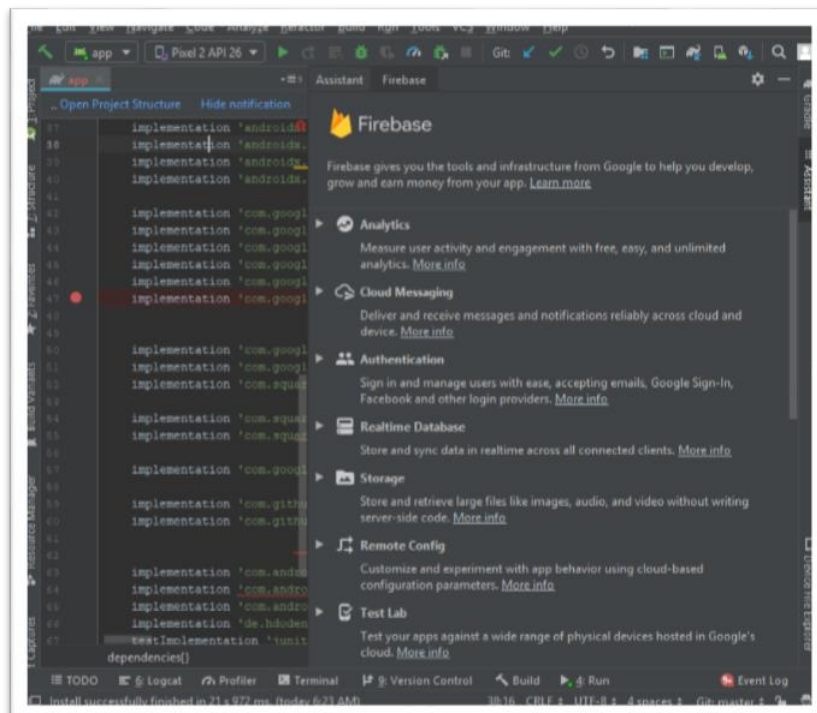
Gambar 8. Tampilan Menu (a) Pilihan Mesin, (b) *Inspection Checklist*

Pada Gambar 9 adalah tampilan *menu maintenance details*, terdapat informasi dari unit *ventex* sebagai objek yang sedang dilakukan perawatan, kemudian terdapat kolom isian mengenai detail informasi mengenai perawatan yang telah dilakukan. Kemudian sama seperti pada *menu inspection checklist*, pada menu ini juga terdapat tombol

send report untuk dapat mengirimkan laporan inspeksi yang telah dibuat. Laporan yang telah dibuat, selanjutnya akan diteruskan kepada pengguna aplikasi sistem pelaporan yang lainnya dan tersimpan di dalam *database* yang telah tersedia. *Database* dari aplikasi sistem pelaporan menggunakan *Firebase*.



Gambar 9. Tampilan Menu *Maintenance Details* pada Aplikasi Sistem Pelaporan



Gambar 10. Menu Konfigurasi *Firebase* ke *Project* di *Android Studio*

Gambar 10 adalah tampilan semua fitur Firebase yang diperlukan. Penggunaan fungsi *authentication* untuk *admin* atau *member*, *google* untuk *member*, dan *anonim* untuk *guest*. Penyimpanan data terbagi menjadi 2 (dua), yaitu gambar menggunakan *cloud storage* dan *data text* menggunakan *realtime database*, untuk selanjutnya disimpan dalam *realtime database* kemudian ditampilkan kembali ke dalam aplikasi.

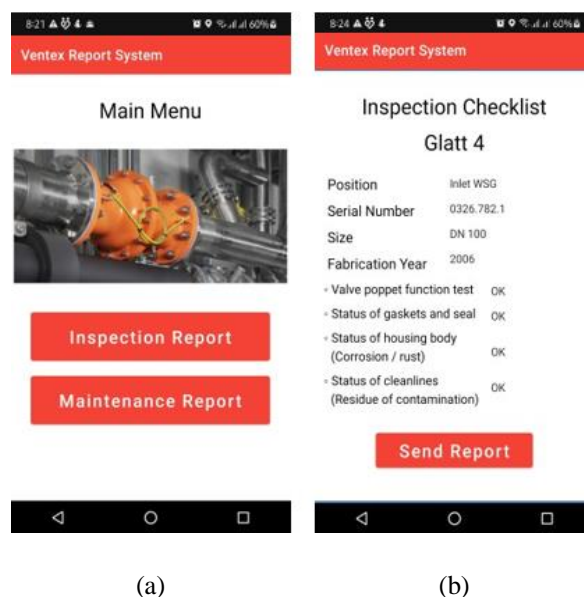
Realtime database tidak memiliki tabel atau catatan. Setelah desain dari *database* diimplementasikan, tampilan *database* akan tersusun seperti pada Gambar 10, di mana uid merupakan kode unik yang dibuat otomatis yang difungsikan sebagai node dalam satu data tersebut agar dapat membedakan dengan data lainnya untuk mencegah terjadinya penduplikasian node yang membuat data yang telah ada tergantikan dengan data yang baru.

Hasil Pengujian Aplikasi Sistem Pelaporan

Waktu pembuatan laporan

Sebelum dibuat sistem pelaporan berbasis Android, setiap pelapor melaporkan pekerjaannya di dalam laporan menggunakan *software* Microsoft Word dan mengirimkannya melalui surat elektronik (*e-mail*). Dalam membuat laporan, setidaknya membutuhkan waktu sekitar 2-3 jam untuk dapat memindahkan data dari catatan atau *smartphone* ke dalam laptop untuk membuat laporan.

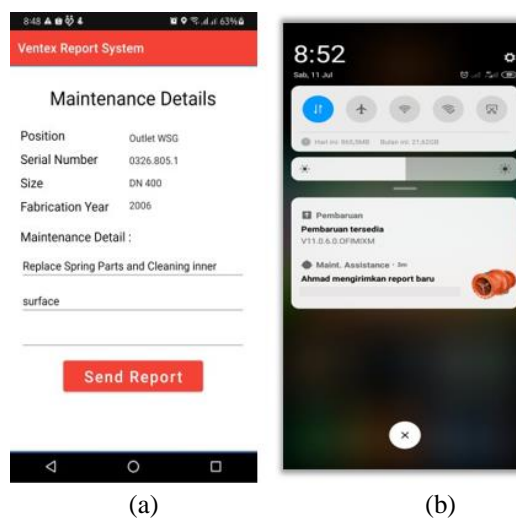
Berdasarkan Gambar 11, dengan menggunakan sistem pelaporan berbasis Android, pelapor dapat melaporkan pekerjaannya langsung menggunakan *smartphone* yang dimiliki sehingga lebih cepat, terlihat hanya membutuhkan waktu estimasi 2-3 menit untuk membuat laporan pekerjaan yang dilakukan.



Gambar 11. Tampilan (a) Menu Utama (catatan jam 08.21) dan (b) *Inspection Checklist* yang Sudah Diisi (catatan jam 08.24)

Tabel 2. Jadwal Inspeksi Sebelum dan Sesudah Pembuatan Sistem Pelaporan Otomatis

Nomor Line	Jadwal Inspeksi Sebelum	Pelaporan Inspeksi Sebelum	Selisih Waktu (Hari)	Jadwal Inspeksi Sesudah	Pelaporan Inspeksi Sesudah	Selisih Waktu (Menit)
Line 4	3-5/02/2020	7/02/2020	2 hari	2-3/03/2021	4/03/2021	2
	2/11/2020	3/11/2020	1 hari	4/11/2021	5/11/2021	3
Line 5	3-5/02/2020	7/02/2020	2 hari	2-3/03/2021	4/03/2021	2
	29/11/2020	6/12/2020	7 hari	4-5/02/2022	10/01/2022	2
Line 6	3-5/02/2020	7/02/2020	2 hari	9/03/2021	11/03/2021	3
	19/11/2020	22/11/2020	3 hari	6/01/2022	8/01/2022	3



Gambar 12. Tampilan (a) Menu *Maintenance Details* (catatan jam 08.48) dan (b) Layar Notifikasi (catatan jam 08.52)

Tabel 2 merupakan data simulasi pengambilan data secara rata-rata pada tiap *line* dan terlihat waktu rata-rata pelaporan inspeksi otomatis kisaran 2-3 menit. Hal ini sekaligus menyatakan signifikansi durasi dari yang hitungan hari menjadi hitungan menit.

Waktu penerimaan notifikasi

Setelah melakukan pengujian waktu pembuatan laporan, maka dilakukan pengujian terhadap waktu penerimaan notifikasi laporan yang telah dibuat dalam aplikasi sistem pelaporan berbasis Android ini.

Berdasarkan Gambar 12, perkiraan waktu yang dibutuhkan dalam penerimaan notifikasi dari setelah laporan dikirim adalah 3-4 menit. Notifikasi diterima oleh *smartphone* dari pengguna aplikasi yang lain saat pelapor mengirimkan laporan dari *smartphone* pengguna.

Tabel 3 merupakan data simulasi durasi penerimaan notifikasi secara rata-rata pada tiap *line* dan terlihat waktu rata-rata durasi yang diperlukan adalah kisaran 3- 4 menit dan hal ini juga menunjukkan signifikansi durasi dari yang hitungan hari menjadi hitungan menit.

Tabel 3. Penerimaan Notifikasi Sebelum dan Sesudah Pembuatan Sistem Pelaporan Otomatis

Nomor <i>Line</i>	Jadwal Perawatan Sebelum	Pelaporan Perawatan Sebelum	Selisih Waktu (Hari)	Jadwal Perawatan Sesudah	Pelaporan Perawatan Sesudah	Selisih Waktu (Menit)
<i>Line 4</i>	2-3/03/2020	4/03/2020	1 hari	2-3/03/2021	4/03/2021	3
	4/11/2020	5/11/2020	1 hari	4/11/2021	5/11/2021	4
<i>Line 5</i>	2-3/03/2020	4/03/2020	1 hari	2-3/03/2021	4/03/2021	4
	4-5/01/2021	10/01/2021	5 hari	4-5/02/2022	10/01/2022	4
<i>Line 6</i>	9/03/2020	11/03/2020	2 hari	9/03/2021	11/03/2021	3
	6/01/2021	8/01/2021	2 hari	6/01/2022	8/01/2022	4

Tabel 4. Hasil Pengujian Aplikasi dengan Metode *Blackbox Testing*

No	Nama Fungsi	Luaran yang diharapkan	Berhasil
1.	<i>Sign In / Register</i>	<i>Email</i> dan <i>Password</i> tercatat dan sesuai dengan data	√
2.	<i>Inspection Report</i>	Masuk ke menu <i>ventex inspection</i>	√
3.	<i>Maintenance Report</i>	Masuk ke menu <i>report information (maintenance)</i>	√
4.	<i>Cust. Name</i>	<i>Edit text</i> untuk nama pelanggan	√
5.	<i>Cust. Contact</i>	<i>Edit text</i> untuk PIC pelanggan	√
6.	<i>Machine Supplier</i>	Muncul pilihan beberapa supplier mesin	√
7.	<i>Machine Type</i>	<i>Edit text</i> untuk tipe mesin	√
8.	<i>Serial Number</i>	<i>Edit text</i> untuk nomor serial mesin.	√
9.	<i>Date of Visit</i>	<i>Edit text</i> untuk tanggal <i>visit</i>	√
10.	<i>Service Engineer</i>	Muncul pilihan nama <i>engineer</i>	√
11.	<i>Create Report</i>	Masuk ke menu <i>inspection checklist</i> pada laporan inspeksi dan masuk ke menu <i>maintenance details</i> pada laporan perawatan.	√
12.	<i>Position</i>	Muncul pilihan lokasi <i>ventex</i> dipasang.	√
13.	<i>Serial Number</i>	Muncul pilihan nomor serial <i>ventex</i> .	√
14.	<i>Size</i>	Muncul pilihan ukuran diameter <i>ventex</i> .	√
15.	<i>Fabrication Year</i>	Muncul pilihan tahun produksi dari unit <i>ventex</i> .	√
16.	<i>Status</i>	Muncul pilihan "OK" atau "Not OK"	√
17.	<i>Maintenance Detail</i>	<i>Edit text</i> untuk detail perawatan yang dilakukan	√
18.	<i>Send Report</i>	Mengirimkan laporan yang telah dibuat.	√

Pengujian aplikasi dengan metode *blackbox testing*

Selain pengujian yang berkaitan dengan waktu, pengujian juga dilakukan terhadap aplikasi sistem pelaporan yang telah dibuat menggunakan metode *blackbox testing*. Metode ini melakukan pengujian aplikasi dari segi fungsional tanpa menguji desain dan kode program, untuk mengetahui apakah fungsi, masukan, dan keluaran dari aplikasi sesuai dengan apa yang telah diprogram. Hasil pengujian dengan metode *blackbox testing* terhadap aplikasi sistem pelaporan inspeksi dan perawatan *ventex* dapat dilihat pada Tabel 4.

Berdasarkan hasil pengujian pada Tabel 4 dapat diketahui bahwa aplikasi sistem pelaporan yang dibuat sudah bekerja sesuai fungsi dan program yang telah direncanakan, sehingga menghasilkan luaran yang sesuai dengan yang diharapkan.

KESIMPULAN DAN SARAN

Hasil pengujian sistem pelaporan inspeksi dan perawatan *ventex explosion isolation valve* berbasis Android menunjukkan bahwa sistem bekerja sudah sesuai. Durasi waktu pelaporan dari 1-7 hari menjadi sekitar 2-3 menit. Pengiriman dan penerimaan notifikasi laporan inspeksi dan perawatan *ventex* menjadi lebih efisien dari 1-5 hari menjadi 3-4 menit. Hal ini menjadi solusi atas masalah sistem yang manual dan lamban menjadi otomatis dan lebih cepat prosesnya,

sehingga proses pengambilan keputusan selanjutnya dapat dilakukan dengan cepat dan meminimalisir kerugian perusahaan. Saran untuk pengembangan dari sistem pelaporan inspeksi dan perawatan *ventex* otomatis adalah mengembangkan tampilan aplikasi dengan lebih lengkap dan menarik, dan dapat beroperasi pada sistem operasi lain seperti iOS, sehingga lebih luas penggunaan sistem ini tidak pada Android saja.

DAFTAR PUSTAKA

- [1] R. Winson, H. Palit, and K. Purba, "Perancangan dan pembuatan aplikasi pemeliharaan untuk memantau kondisi mesin pabrik berbasis android di PT. X," *Jurnal Infra*, vol. 5, no. 2, pp. 275–281, 2017.
- [2] N. Agustina, D. A. Rismayadi, A. I. Aji, and N. Salsabila, "Rancang bangun sistem pelaporan dan pemantauan jalan berlubang menggunakan cloud dan pendeteksi objek," *Prosiding Seminar Hasil Penelitian Teknik Informatika (SehaTIF)*, 2021.
- [3] T. L. Fachruti and T. Wijatmaka, "Rancang bangun aplikasi stock di warehouse PT. Solusi Bangun Indonesia," in *Seminar Nasional Teknik Mesin*, 2021, pp. 965–973.
- [4] A. Ayuningtyas, A. Pujiastuti, A. Kusumaningrum, N. D. Retnowati, and G. N. Sorateleng, "Aplikasi ngesiki untuk pelaporan penumpukan sampah memanfaatkan Global Positioning

- System (GPS) dan firebase,” *Angkasa: Jurnal Ilmiah Bidang Teknologi*, vol. 13, no. 2, pp. 153–162, 2021.
- [5] J. Panjaitan and A. F. Pakpahan, “Perancangan sistem e-reporting menggunakan ReactJS dan firebase,” *Jurnal Teknik Informatika Dan Sistem Informasi*, vol. 7, no. 1, 2021.
- [6] S. Setiaji and R. Sastra, “Implementasi diagram UML (Unified Modelling Language) pada perancangan sistem informasi penggajian,” *Jurnal Teknik Komputer*, vol. 7, no. 1, pp. 106–111, 2021.
- [7] M. Purnasari, Y. Hartiwi, and N. Nurhayati, “Perancangan sistem informasi pengelolaan dana masjid berbasis web menggunakan Unified Modeling Language (UML),” *Resolusi: Rekayasa Teknik Informatika dan Informasi*, vol. 2, no. 6, pp. 258–264, 2022.
- [8] F. Sonata, “Pemanfaatan UML (Unified Modeling Language) dalam perancangan sistem informasi e-commerce jenis customer-to-customer,” *Jurnal Komunika: Jurnal Komunikasi, Media Dan Informatika*, vol. 8, no. 1, pp. 22–31, 2019.
- [9] F. Fatmasari and S. Sauda, “Pemodelan unified modeling language sistem informasi enterprise resource planning,” *Jurnal Media Informatika Budidarma*, vol. 4, no. 2, pp. 429–436, 2020.
- [10] D. Saputra, H. Haryani, A. Surniandari, M. Martias, and F. Akbar, “Sistem informasi bimbingan tugas akhir mahasiswa berbasis website menggunakan metode waterfall,” *MATRIK: Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 21, no. 2, pp. 403–416, 2022.
- [11] T. Feriatna, S. A. Pramuditya, and N. Aminah, “Pengembangan aplikasi android sebagai media pembelajaran matematika pada materi peluang untuk siswa SMA kelas X,” *Lemma: Letters of Mathematics Education*, vol. 4, no. 1, 2017.
- [12] S. Riyadi and I. Nurhaida, “Aplikasi sistem virtual tour e-panorama 360 derajat berbasis android untuk pengenalan kampus Mercu Buana,” *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 9, no. 1, pp. 17–24, 2022.
- [13] M. Naomi, “Analisa dan perancangan sistem pengaduan mahasiswa berbasis web (studi kasus: Universitas Mercu Buana Kranggan),” *JUSIBI (Jurnal Sistem Informasi dan Bisnis)*, vol. 1, no. 5, 2019.
- [14] Y. A. Syakura, “Aplikasi pelaporan kegiatan mentoring di Universitas Teknologi Sumbawa berbasis android,” *Jurnal Informatika Teknologi dan Sains (Jinteks)*, vol. 2, no. 2, pp. 120–128, 2020.

- [15] F. Mohamad and F. Tupamahu, “Rancangan sistem integrasi monitoring dan pelaporan sampah di kota Gorontalo berbasis android,” in *SemanTECH (Seminar Nasional Teknologi, Sains dan Humaniora)*, 2019, pp. 50–57.
- [16] N. Harahap, H. Santoso, and M. Alda, “Aplikasi pelaporan tindakan kekerasan terhadap perempuan dan anak di DP3APMP2KB Medan berbasis android,” *Jurnal Responsif: Riset Sains dan Informatika*, vol. 6, no. 1, pp. 92–102, 2024.
- [17] N. Hidayanti, W. Widyawati, R. Fatullah, and B. Budiono, “Rancang bangun aplikasi monitoring kegiatan kuliah kerja mahasiswa berbasis android di Universitas Banten Jaya,” *Teknika: Jurnal Sains Dan Teknologi*, vol. 16, no. 2, pp. 267–278, 2020.