

IMPLEMENTASI SISTEM MANAJEMEN PREVENTIF UNTUK PEMELIHARAAN INVENTARIS PERANGKAT IT MENGGUNAKAN METODE *RAPID APPLICATION DEVELOPMENT (RAD)*

Sabar Hanadwiputra¹, Mami Maryati², Dewi Ayu Puspitawati³, Ardhan Zuhri⁴

Universitas Bani Saleh^{1,2,3,4}

Jl. M.Hasibuan No.68 Bekasi

Email: sabar@ubs.ac.id, mami@ubs.ac.id, dewi@ubs.ac.id, ardhanzuchri@gmail.com

ABSTRAK

Penelitian ini bertujuan untuk meimplementasikan aplikasi yang memberikan untuk pegawai kemudahan melakukan pemeliharaan pada perangkat IT. Dalam proses pencatatan laporan yang belum memanfaatkan teknologi seperti terdapat sebuah sistem pencatatan dengan menggunakan sistem komputer, dengan banyak menemukan kendala dalam penyimpanan catatan berbasis kertas, yang memiliki beberapa kelemahan seperti keterbatasan ruang penyimpanan, akses dokumen yang memakan waktu, risiko kerusakan, kerentanan terhadap dokumen. Perlu adanya aplikasi sistem manajemen untuk mengatasi kesalahan pencatatan laporan, dalam analisis dan implementasi menggunakan metode pengembangan perangkat lunak, dan dalam *Rapid Application Development (RAD)* menghasilkan performance yang cukup baik untuk sistem manajemen pencatatan. Pembuatan sistem ini menggunakan bahasa pemrograman *Java* untuk *Android*, *database Firebase*.

Kata Kunci : Sistem manajemen, Pencatatan, *Rapid Application Development (RAD)*

ABSTRACT

This study aims to implement an application that provides employees with the convenience of performing maintenance on IT devices. In the process of recording reports that have not utilized technology such as a recording system using a computer system, with many obstacles in storing paper-based records, which have several weaknesses such as limited storage space, time-consuming document access, risk of damage, vulnerability to documents. There needs to be a management system application to overcome errors in recording reports, in analysis and implementation using software development methods, and in Rapid Application Development (RAD) producing quite good performance for the recording management system. The creation of this system uses the Java programming language for Android, the Firebase database.

Keyword : Management system, Record keeping, *Rapid Application Development (RAD)*

1. PENDAHULUAN

Proses digitalisasi merupakan transformasi aset analog konvensional, seperti buku dan produk kertas lainnya, menjadi salinan digital dan format elektronik. Dengan menerapkan digitalisasi, perusahaan dapat mengatasi hambatan-hambatan yang timbul dari penyimpanan dokumen berbasis kertas, dan menciptakan solusi yang efisien untuk mengelola catatan dengan lebih baik.

Pengelola memiliki aset yang lebih kompleks, pemeliharaan reaktif dapat menjadi mahal dan tidak efisien. Diperlukan strategi yang lebih proaktif dan komprehensif seperti pemeliharaan preventif. Pemeliharaan preventif menggunakan catatan pemeliharaan, daftar periksa, perintah kerja, dan metrik kinerja untuk membantu teknisi menemukan peluang melakukan pemeliharaan terencana pada aset sebelum aset tersebut rusak. Kemampuan seperti pembelajaran

2. METODE PENELITIAN

Dalam tahap perancangan penelitian sistem menggunakan RAD (*Rapid Application Development*). Dalam RAD menggunakan tiga fase pengembangan, yaitu :

- a. Perencanaan syarat-syarat (*Requirement Planning*)
Dalam fase ini terdiri dari dua tahap, yaitu :
 1. Identifikasi tujuan sistem yang hendak dibangun tahap ini mengidentifikasi tujuan yang ingin dicapai dengan sistem yang hendak dibangun untuk menyelesaikan atau mengatasi kebutuhan pemilik Perusahaan dalam pencatatan dokumen tasi yang masih menggunakan buku yang sangat rawan akan hilang dan rusak.
 2. Identifikasi kebutuhan sistem yang baru tahap ini mengidentifikasi kebutuhan apa saja yang diperlukan untuk membangun sistem kebutuhan *input*, kebutuhan proses dan kebutuhan *output* yang diinginkan.
- b. *Workshop* Desain. Fase ini dilakukan untuk merancang sistem yang hendak dibangun yang dapat menyelesaikan masalah-masalah yang ada menggunakan diagram-diagram menjelaskan tentang rancangan gambar dari sistem yang hendak dibangun dengan *Unifed Modeling Language* (UML).
- c. *Implementation*. setelah *Design Workshop* dilakukan, selanjutnya sistem diimplementasikan (*coding*) ke dalam bentuk yang dimengerti oleh mesin yang diwujudkan dalam bentuk program atau unit program. Tahap implementasi system merupakan tahap meletakkan sistem supaya siap untuk dioperasikan.

Perancangan Sistem

Dalam perancangan sistem ini penulis menggunakan UML (*Unified Modeling Language*). Penulis menganggap UML ini kaya akan notasi karena memiliki banyak diagram, dimana masing-masing diagram memiliki sudut pandang berbeda-beda. Penelitian ini penulis mengidentifikasi kebutuhan dari sudut pandang user, perancangan sistem ini akan digambarkan dengan menggunakan :

- a. *Use Case* Diagram
- b. *Activity* Diagram
- c. *Squence* Diagram
- d. *Class* Diagram

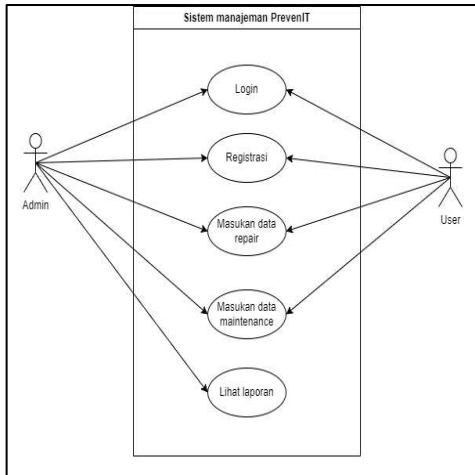
Berdasarkan analisa pada pengembangan sistem yang berjalan, maka penulis membuat aktor yang terlibat dalam perancangan dan implemetasi Aplikasi Sistem Manajemen Preventif pada perusahaan XYZ. Adapun aktor yang terlibat adalah *user*, berikut ini adalah table penjelasan aktor dan perannya dalam sistem.

Tabel 2. 1 Peranan Aktor

Aktor	Peranan Aktor
Admin/User	Mencatat dokumentasi <i>maintenance</i> pada perusahaan

2.1.1 Perancangan Use Case Diagram

Pada bagian ini adalah gambar aktivitas *user* dengan sistem, sehingga dapat terlihat alur aktivitas *user* dan akan digambar kan dengan *use case diagram*. Pada tahapan ini akan terlihat apa saja yang dilakukan oleh aktor kepada *system*.



Gambar 2.1

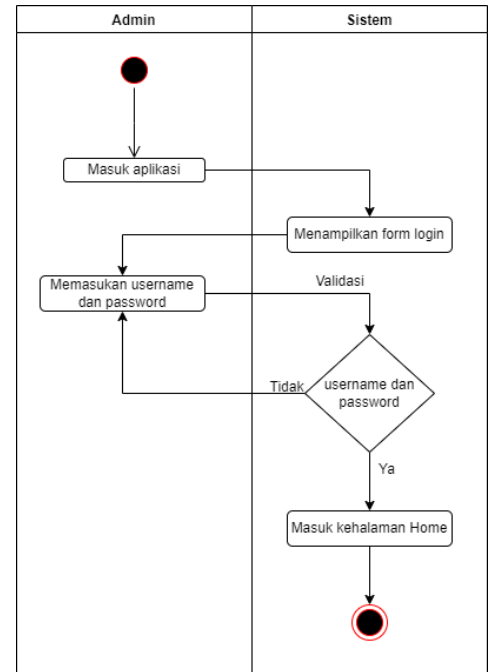
Usecase Diagram

2.1.2 Perancangan Activity Diagram

Activity Diagram menguraikan interaksi yang terjadi antara *user* dengan sistem pada masing – masing *use case*. Dalam penelitian ini ada beberapa aktivitas user terhadap sistem yaitu sebagai berikut:

A. Activity Diagram Login Admin

Activity Diagram Login ini menggambarkan dalam memahami langkah-langkah yang diambil oleh admin dan sistem selama proses *login*. Diagram ini memastikan bahwa setiap tahap dalam proses *login* dilakukan dengan benar dan sistem memberikan respon yang sesuai berdasarkan hasil validasi data *login* admin.

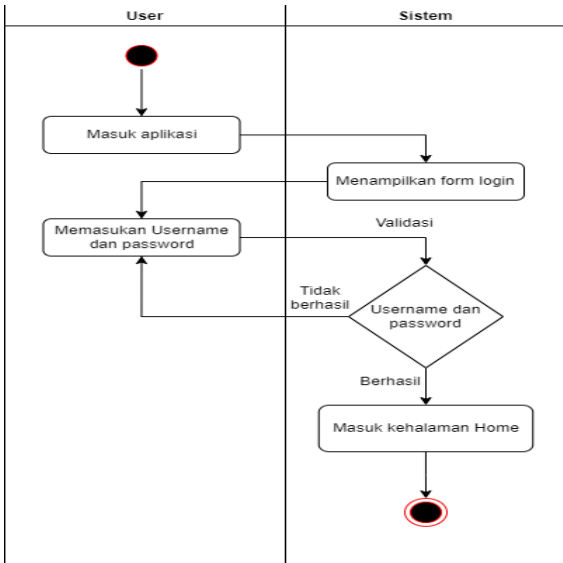


Gambar 2. 2

Activity Diagram Login Admin

B. Activity Diagram Login User

Activity Diagram Login ini menggambarkan dalam memahami langkah-langkah yang diambil oleh pengguna dan sistem selama proses *login*. Diagram ini memastikan bahwa setiap tahap dalam proses *login* dilakukan dengan benar dan sistem memberikan respon yang sesuai berdasarkan hasil validasi data *login* pengguna.

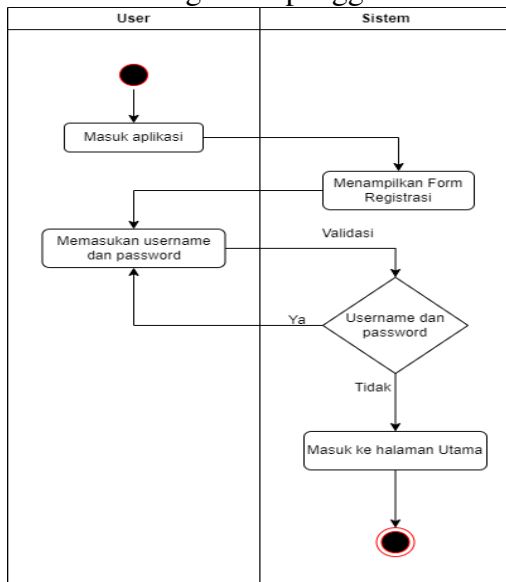


Gambar 2.3

Activity Diagram Login User

C. Activity Diagram Registrasi

Activity Diagram Registrasi ini menggambarkan dalam memahami langkah-langkah yang diambil oleh pengguna dan sistem selama proses Registrasi. Diagram ini memastikan bahwa setiap tahap dalam proses login dilakukan dengan benar dan sistem memberikan respon yang sesuai berdasarkan hasil validasi data Registrasi pengguna.

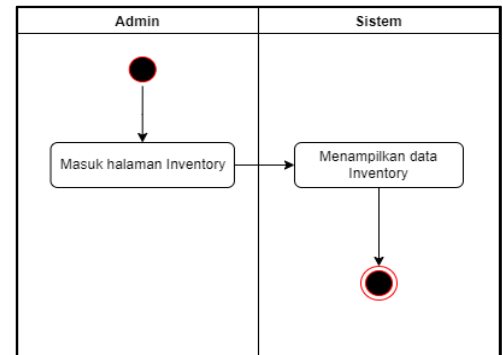


Gambar 2. 4

Activity Diagram Registrasi

D. Activity Diagram Inventory Admin

Activity diagram Inventory ini menggambarkan interaksi antara admin dan sistem dalam proses penginputan data inventory. Admin mengawali proses dengan mengakses halaman inventori dan mengisi form yang ada. Setelah data diinput oleh admin, sistem akan mengambil alih untuk menyimpan data tersebut ke dalam database, dan proses pun berakhir:

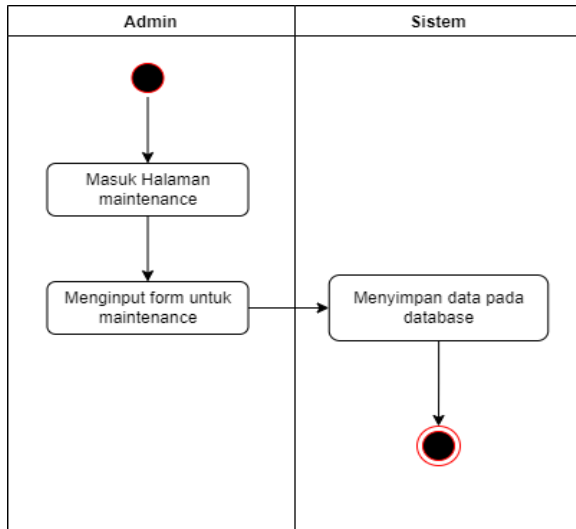


Gambar 2.5

Activity Diagram Inventory Admin

E. Activity Diagram Maintenance Admin

Activity diagram Maintenance ini menggambarkan dengan jelas interaksi antara admin dan sistem dalam proses penginputan data maintenance. Admin mengawali proses dengan mengakses halaman maintenance dan mengisi form yang ada. Setelah data diinput oleh admin, sistem akan mengambil alih untuk menyimpan data tersebut ke dalam database, dan proses pun berakhir :

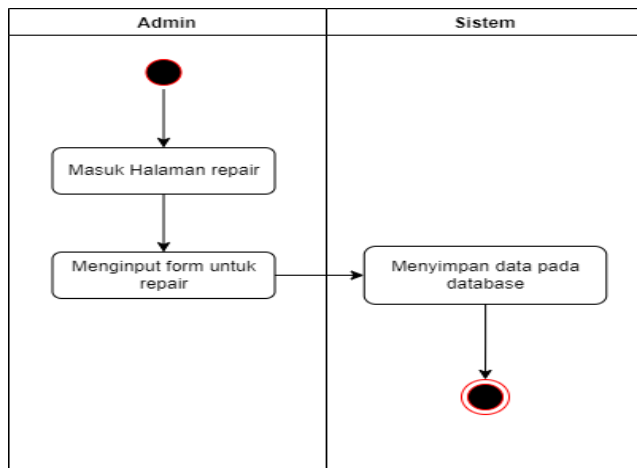


Gambar 2. 6

Activity Diagram Maintenance Admin

F. Activity Diagram Repair Admin

Activity Diagram Repair ini menggambarkan dengan jelas interaksi antara admin dan sistem dalam proses penginputan data repair. Admin mengawali proses dengan mengakses halaman repair dan mengisi form yang ada. Setelah data diinput oleh admin, sistem akan mengambil alih untuk menyimpan data tersebut ke dalam database, dan proses pun berakhir :



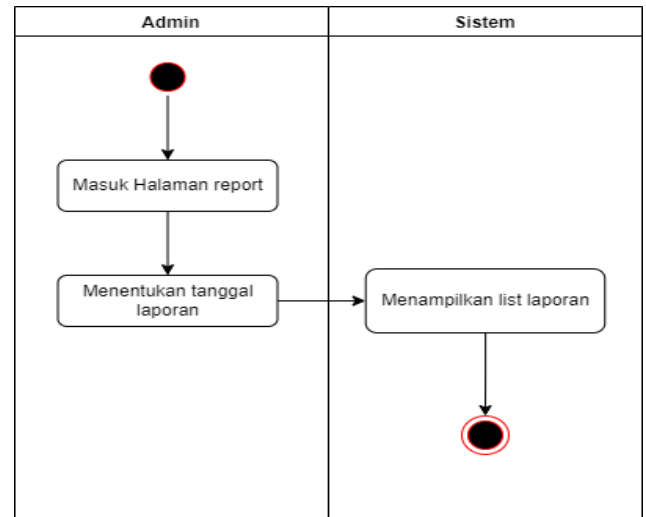
Gambar 2.7

Activity Diagram Repair Admin

G. Activity Diagram Report Admin

Activity Diagram Report

menggambarkan alur dalam memeriksa Report pencatatan. Dalam menampilkan laporan tersebut, admin dapat memilih jangka waktu dari laporan yang akan ditampilkan:

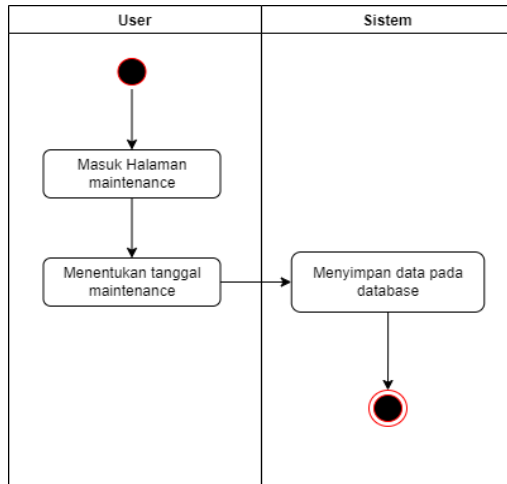


Gambar 2.8

Activity Diagram Report Admin

H. Activity Diagram Maintenance Admin

Activity diagram Maintenance ini menggambarkan dengan jelas interaksi antara user dan sistem dalam proses penginputan data maintenance. User mengawali proses dengan mengakses halaman maintenance dan mengisi form yang ada. Setelah data diinput oleh user, sistem akan mengambil alih untuk menyimpan data tersebut ke dalam database, dan proses pun berakhir :

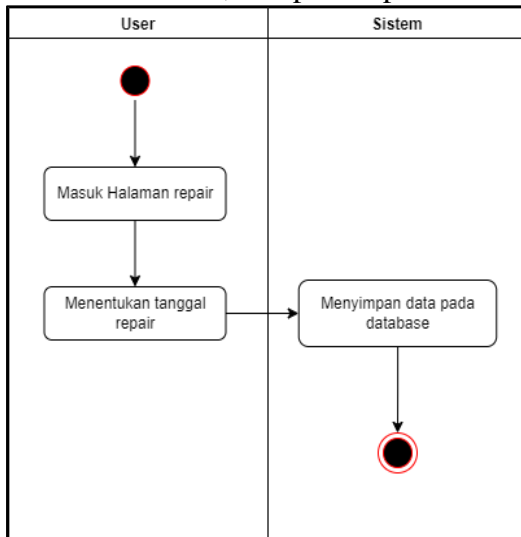


Gambar 2.9

Activity Diagram Maintenance User

I. Activity Diagram Repair Admin

Activity Diagram Repair ini menggambarkan dengan jelas interaksi antara user dan sistem dalam proses penginputan data repair. User mengawali proses dengan mengakses halaman repair dan mengisi form yang ada. Setelah data diinput oleh user, sistem akan mengambil alih untuk menyimpan data tersebut ke dalam database, dan proses pun berakhir :



Gambar 2. 10

Activity Diagram Repair User

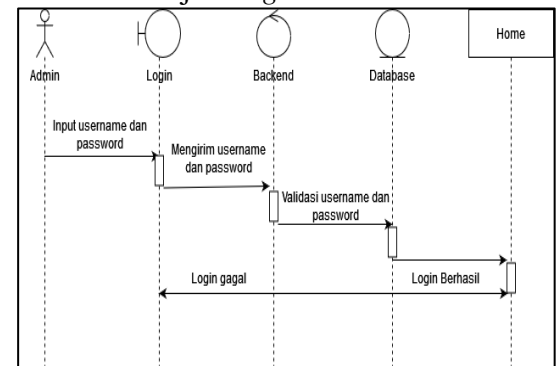
2.1.2 Perancangan Sequence Diagram

Sequence Diagram merupakan salah

satu jenis diagram pada UML yang menjelaskan interaksi objek yang berdasarkan urutan waktu, Sequence Diagram juga dapat menggambarkan tahapan yang harus dilakukan untuk dapat menghasilkan sesuatu seperti pada usecase diagram.

A. Sequence Diagram Login Admin

Pada Diagram sequence ini membantu dalam memahami alur proses login dengan lebih jelas dan memastikan bahwa setiap komponen dalam sistem berinteraksi dengan benar untuk mencapai tujuan akhir yaitu autentikasi pengguna dan pengalihan ke halaman beranda jika login berhasil.

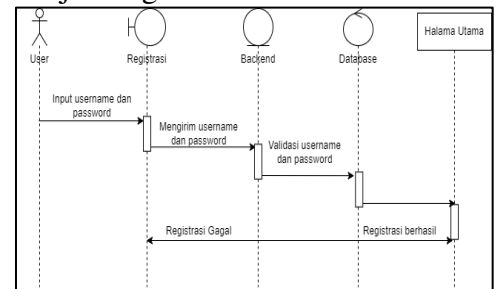


Gambar 2. 11

Sequence Diagram Login Admin

B. Sequence Diagram Registrasi

Pada Diagram sequence ini membantu dalam memahami alur proses Registrasi dengan lebih jelas dan memastikan bahwa setiap komponen dalam sistem berinteraksi dengan benar untuk mencapai tujuan akhir yaitu autentikasi pengguna dan pengalihan ke halaman beranda jika registrasi berhasil.

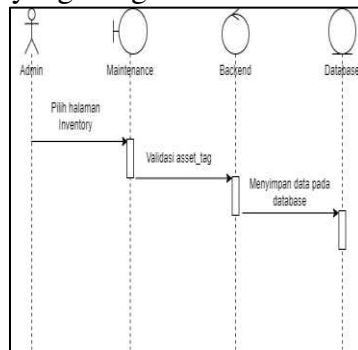


Gambar 2. 12

Sequence Diagram Registrasi

C. Sequence Diagram Inventory Admin

Pada Diagram ini menggambarkan alur proses validasi dan penyimpanan data inventaris secara rinci, mulai dari interaksi awal oleh pengguna hingga penyimpanan data di *database*. Diagram ini menunjukkan bagaimana setiap komponen sistem berinteraksi satu sama lain untuk menyelesaikan tugas yang diinginkan :

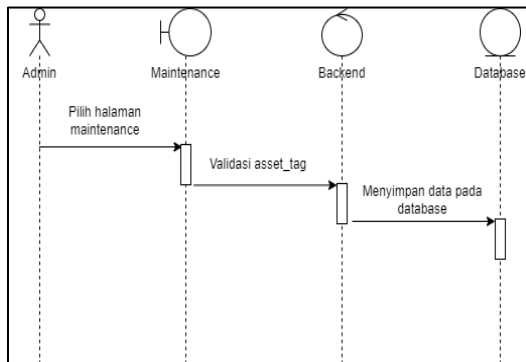


Gambar 2.13

Activity Diagram Inventory Admin

D. Sequence Diagram Maintenance Admin

Pada Diagram ini memberikan gambaran yang jelas mengenai alur aktivitas yang terjadi mulai dari pengguna memilih halaman pemeliharaan hingga data yang valid berhasil disimpan dalam *database*. Setiap langkah dijelaskan secara rinci untuk memberikan pemahaman yang lengkap tentang proses yang terjadi di dalam sistem :

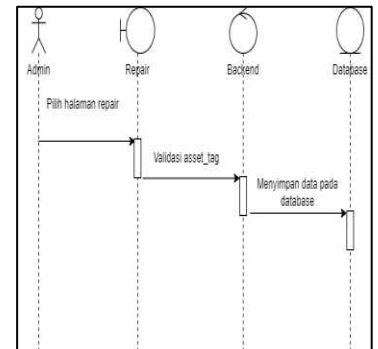


Gambar 2.14

Sequence Diagram Maintenance Admin

E. Sequence Diagram Repair Admin

Pada diagram ini memberikan gambaran yang jelas mengenai alur aktivitas yang terjadi mulai dari pengguna memilih halaman perbaikan hingga data yang valid berhasil disimpan dalam *database*. Setiap langkah dijelaskan secara rinci untuk memberikan pemahaman yang lengkap tentang proses yang terjadi di dalam sistem :

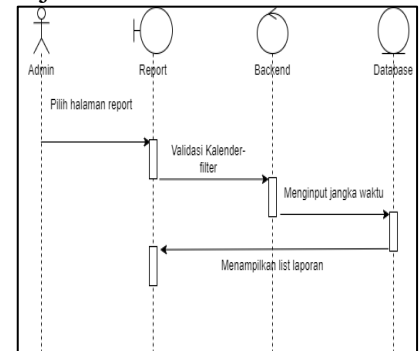


Gambar 2.15

Sequence Diagram Repair Admin

F. Sequence Diagram Report Admin

Pada diagram ini memberikan gambaran yang jelas mengenai alur aktivitas yang terjadi mulai dari pengguna memilih halaman laporan hingga laporan yang sesuai ditampilkan. Setiap langkah dijelaskan secara terperinci untuk memberikan pemahaman yang lengkap tentang proses yang terjadi di dalam sistem :

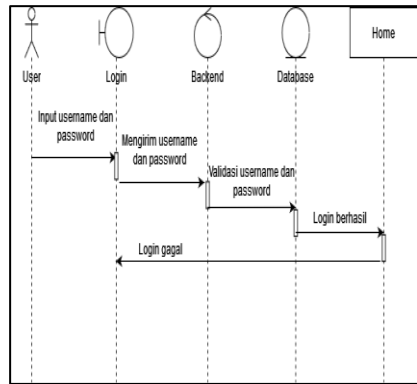


Gambar 3.16

Sequence Diagram Report Admin

G. Sequence Diagram Login User

Pada Diagram sequence ini membantu dalam memahami alur proses *login* dengan lebih jelas dan memastikan bahwa setiap komponen dalam sistem berinteraksi dengan benar untuk mencapai tujuan akhir yaitu autentikasi pengguna dan pengalihan ke halaman beranda jika *login* berhasil.

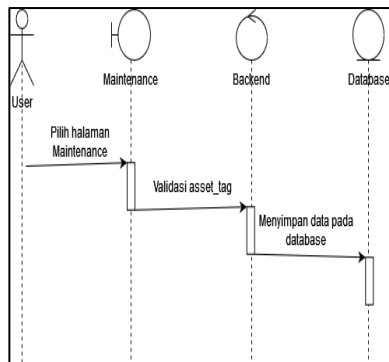


Gambar 2. 17

Sequence Diagram Login User

H. Sequence Diagram Maintenance User

Pada Diagram ini memberikan gambaran yang jelas mengenai alur aktivitas yang terjadi mulai dari pengguna memilih halaman pemeliharaan hingga data yang valid berhasil disimpan dalam *database*. Setiap langkah dijelaskan secara rinci untuk memberikan pemahaman yang lengkap tentang proses yang terjadi di dalam sistem:

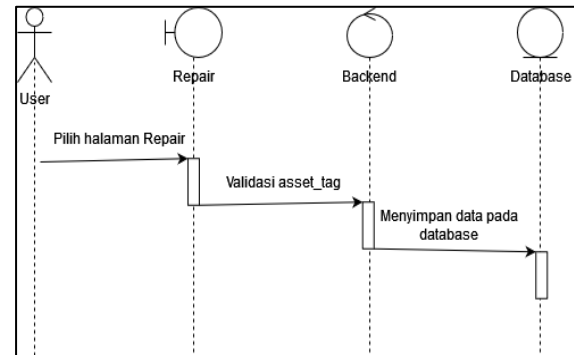


Gambar 2.18

Sequence Diagram Maintenance User

I. Sequence Diagram Repair Admin

Pada diagram ini memberikan gambaran yang jelas mengenai alur aktivitas yang terjadi mulai dari pengguna memilih halaman perbaikan hingga data yang valid berhasil disimpan dalam *database*. Setiap langkah dijelaskan secara rinci untuk memberikan pemahaman yang lengkap tentang proses yang terjadi di dalam sistem:

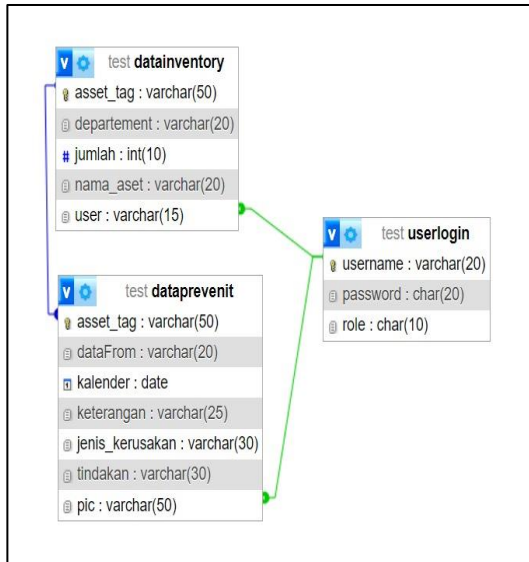


Gambar23. 19

Sequence Diagram Repair User

2.1.3 Perancangan Class Diagram

Class Diagram akan menggambarkan atau menampilkan struktur dari sebuah sistem. Sistem tersebut menampilkan sistem kelas, atribut dan hubungan antar kelas ketika suatu sistem telah selesai membuat diagram. Untuk lebih jelasnya dapat dilihat pada gambar berikut di halaman berikutnya :



Gambar 3. 20

Class Diagram

2.1.4 Perancangan Database

Database yang digunakan pada pembangunan aplikasi ini merupakan satu file database yang akan berfungsi untuk menyimpan informasi structural table yang dibutuhkan untuk menunjang pengoperasian program. Table yang dibutuhkan sebagai berikut :

Tabel 2.2 Data Inventory

No	Nama Field	Type Field	Length Field
1	Asset_tag	Varchar	50
2	Department	Varchar	20
3	jumlah	Int	10
4	Nama_aset	Varchar	20
5	user	Varchar	15

Table Data PrevenIT

Pada tabel Produk ini, penulis merancang tabel Data PrevenIT.

Tabel 2. 3 Data PrevnIT

No	Nama Field	Type Field	Length Field
1	Asset_tag	Varchar	50
2	dataFrom	Varchar	20

No	Nama Field	Type Field	Length Field
3	Kalender	Date	-
4	Keterangan	Varchar	25
5	Jenis_kerusakan	Varchar	30
6	Tindakan	Varchar	30
7	pic	Varchar	50

Table Data User

Pada tabel Produk ini, penulis merancang tabel Data User.

Tabel 2. 4 Data User

No	Nama Field	Type Field	Length field
1	Username	Varchar	20
2	Password	Char	20
3	Role	Char	20

3. HASIL DAN PEMBAHASAN

Implementasi antar muka pengguna merupakan sebuah tahapan dimana sistem siap dioperasikan pada tahap yang sebenarnya, sehingga diketahui apakah sistem telah dibuat sesuai dengan yang direncanakan atau tidak. Pada implementasi antar muka pengguna ini dijelaskan bagaimana sistem ini bekerja dengan memberikan tampilan aplikasi yang dibuat. Berikut ini adalah antar muka pengguna yang sudah di bangun berdasarkan layout pada tahap perancangan program.

3.1.Tampilan Halaman Utama

Pada Halaman ini menampilkan halaman utama aplikasi PrevenIT yang menunjukkan login atau register akun user.



Gambar 3. 1

Tampilan Halaman Utama

3.2 Tampilan Halaman Login

Pada halaman ini menampilkan halaman login aplikasi PreventIT yang menunjukkan form login untuk mengakses ke dalam aplikasi.

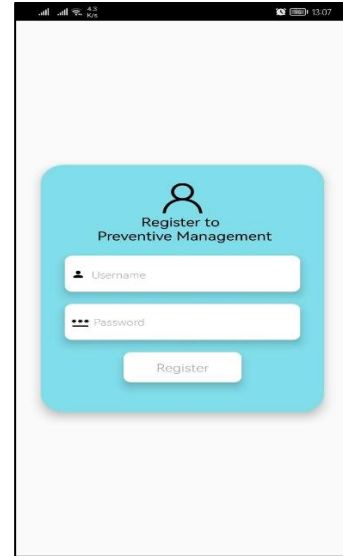


Gambar 3.2

Tampilan Halaman Login

3.3 Tampilan Halaman Registrasi User

Pada halaman ini menampilkan halaman Registrasi pada aplikasi PreventIT yang menunjukkan form registrasi untuk mengakses role user.



Gambar 3. 2

Tampilan Halaman Registrasi User

3.4 Tampilan Halaman Registrasi Admin

Pada halaman ini menampilkan halaman register untuk akun admin (Tambah Account). Dengan syarat, harus login sebagai admin terlebih dahulu untuk membuat akun admin.



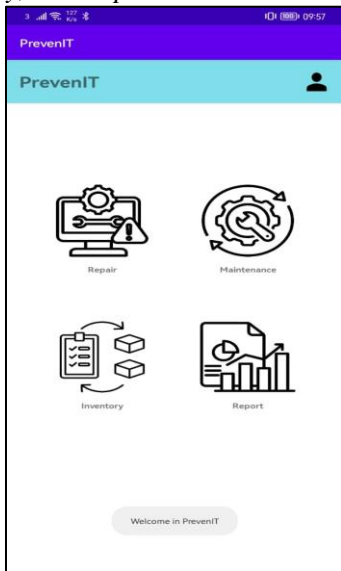


Gambar 3. 3

Tampilan Halaman Registrasi Admin

3.5 Tampilan Halaman Home Admin

Pada halaman ini menampilkan Halaman *Home* dan terdapat 4 Menu yaitu *Repair*, *Maintenance*, *Inventory*, dan *Report*.

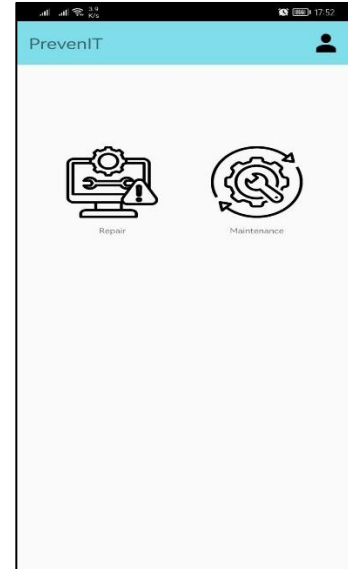


Gambar 3. 4

Tampilan Halaman Home

3.6 Tampilan Halaman Home User

Pada halaman ini menampilkan Halaman *Home* dan terdapat 2 Menu yaitu *Repair* dan *Maintenance*.

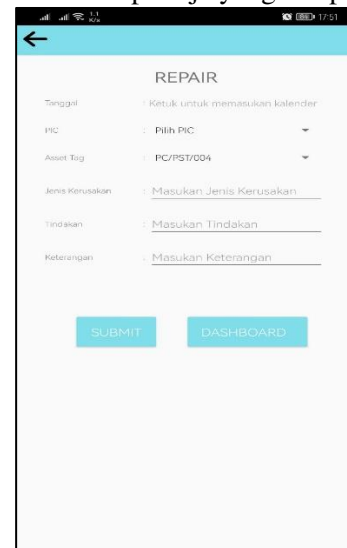


Gambar 3. 5

Tampilan Halaman Home User

3.7 Tampilan Halaman Repair

Pada Halaman ini menampilkan Halaman *Repair* yang berisi form apa saja yang direpair oleh User.

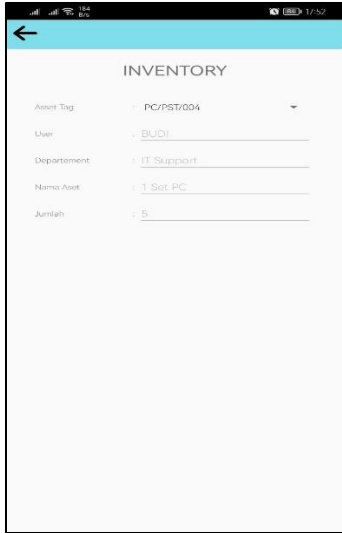


Gambar 43 6

Tampilan Halaman Repair

3.8 Tampilan Halaman Inventory

Pada halaman ini menampilkan halaman *inventory* yang berisi form *inventory*.

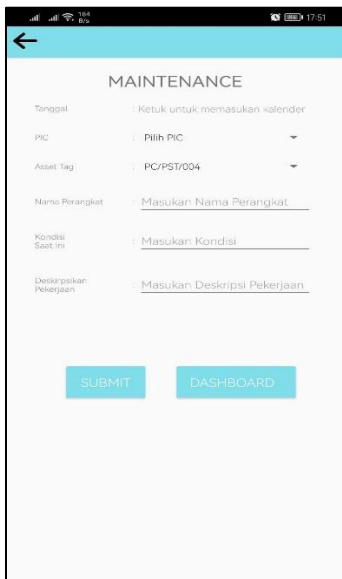


Gambar 3. 7

Tampilan Halaman Inventory

3.9 Tampilan Halaman Maintenance

Pada halaman ini menampilkan Halaman Maintenance yang berisi form apa saja yang akan di maintenance.



Gambar 3. 8

Tampilan Halaman Maintenance

4.0 Tampilan Halaman Report

Pada halaman ini menampilkan halaman report yang merekap semua hasil pencatatan dokumen pemeliharaan inventaris.



Gambar 4. 0

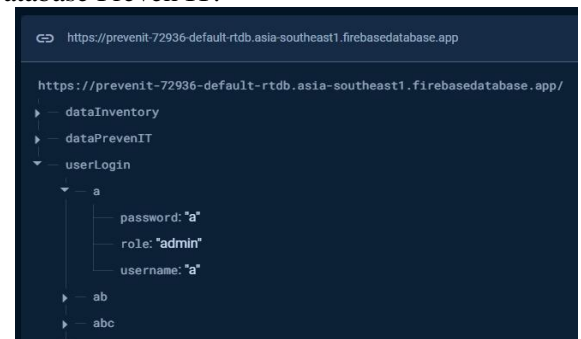
Tampilan Halaman Report

4.1. Implementasi Database

Implementasi database merupakan proses krusial dalam pengelolaan data yang melibatkan beberapa tahapan penting untuk memastikan keefektifan dan efisiensi sistem. Tahapan ini mencakup perancangan skema database yang mendetail, penentuan struktur tabel, serta penetapan relasi antar tabel untuk mengoptimalkan integritas dan konsistensi data.

4.2 tampilan Database UserLogin

Pada halaman ini menampilkan Collection userlogin pada Database Preven IT.

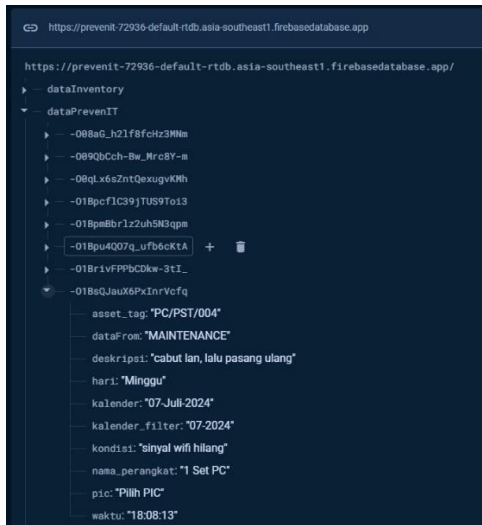


Gambar 4.2

Tampilan Database UserLogin

4.3 Tampilan Database DataPrevenIT

Pada halaman ini menampilkan Collection DataPrevenIT pada Database Preven IT.

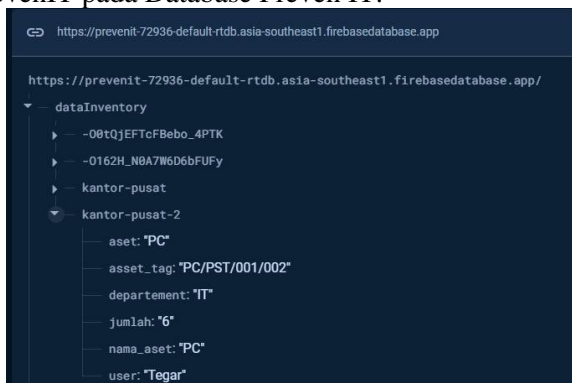


Gambar 4. 3

Tampilan Database DataPrevenIt

4.4 Tampilan Database DataInventory

Pada halaman ini menampilkan Collection DataPrevenIT pada Database Preven IT.



Gambar 4. 4

Tampilan Database DataInventory

4. KESIMPULAN DAN SARAN

Berdasarkan data yang diperoleh dari hasil penelitian maka dapat ditarik kesimpulan dan saran sebagai berikut:

1. Aplikasi android sudah berjalan dengan baik tested device Huawei Mate 20 Pro.
2. Sistem ini membantu management untuk membuat laporan pemeliharaan perangkat IT.
3. Implementasi *Rapid Application Development* pada sistem pencatatan pemeliharaan untuk divisi IT yang dinilai untuk mempermudah urusan management.
4. Sistem Pencatatan ini masih jauh dari kesempurnaan sehingga dapat dikembangkan

menjadi sistem yang lebih baik dan sesuai perkembangan teknologi yang ada.

5. Tampilan aplikasi sistem manajemen prevenIT yang masih sederhana kedepannya dapat dikembangkan untuk mendapatkan aplikasi yang lebih interaktif.
6. Aplikasi ini dapat dikembangkan untuk perusahaan yang mempunyai kebutuhan sejenis.

DAFTAR PUSTAKA

Afandi Nur Aziz Thohari, S. (2015). Pengantar Basis Data Multimedia. In *Lentera Ilmu Cendekia* (pp. 1–12).

Ahmad Ansori. (2022). *Pengertian Class Diagram : Fungsi, Simbol, dan Contohnya*. Ansoriweb.Com.

Aurellia, A. (2022). *Apa Itu Implementasi? Pengertian, Tujuan, dan Contoh Penerapannya*. Detik.Com.

Dicoding. (2021). *Apa itu Firebase? Pengertian, Jenis-Jenis, dan Fungsi Kegunaannya*. Dicoding.

Hutabri, E. (2019). Penerapan Metode Rapid Application Development (RAD) Dalam Perancangan Media Pembelajaran Multimedia. *Innovation in Research of Informatics (INNOVATICS)*, 1(2). <https://doi.org/10.37058/innovatics.v1i2.932>

Putra. (2019). *PENGERTIAN ANDROID: Sejarah, Kelebihan & Versi Sistem Operasi Android OS*. Salamadian.Com.

Sia, V. (2023). *Pentingnya Mengelola Inventaris untuk Kelangsungan Bisnis*. Mekari Jurnal.

Ulum, M. B., Hari Rarindo, H. W., & Dani, ono dan A. (2023). *Pengertian Preventive Maintenance*. *Jurnal Teknologi, Vol 17 No 2 (2023): Nopember 2023*.