

**RANCANG BANGUN APLIKASI MONITORING OPERASI P2TL
PENGUKURAN TIDAK LANGSUNG 3 PHASA DI PT. PLN (PERSERO)
UNIT PELAKSANA PELAYANAN PELANGGAN (UP3) METRO**

(Skripsi)

Oleh:

ARYA DILLAH

1715061008



PROGRAM STUDI S-1 TEKNIK INFORMATIKA

JURUSAN TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS LAMPUNG

2024

**RANCANG BANGUN APLIKASI MONITORING OPERASI P2TL
PENGUKURAN TIDAK LANGSUNG 3 PHASA DI PT. PLN (PERSERO)
UNIT PELAKSANA PELAYANAN PELANGGAN (UP3) METRO**

Oleh

ARYA DILLAH

Skripsi

Sebagai Salah Satu Syarat Untuk Mencapai Gelar

SARJANA TEKNIK

Pada

Program Studi Teknik Informatika

Fakultas Teknik Universitas Lampung



FAKULTAS TEKNIK

UNIVERSITAS LAMPUNG

BANDAR LAMPUNG

2024

ABSTRAK

RANCANG BANGUN APLIKASI MONITORING OPERASI P2TL PENGUKURAN TIDAK LANGSUNG 3 PHASA DI PT. PLN (PERSERO) UNIT PELAKSANA PELAYANAN PELANGGAN (UP3) METRO

Oleh

ARYA DILLAH

Dalam rangka meningkatkan efisiensi dan efektivitas dokumentasi kegiatan monitoring operasi P2TL pengukuran tidak langsung 3 phasa, penelitian ini mengembangkan sebuah aplikasi berbasis *mobile*. Aplikasi ini dirancang untuk memudahkan pekerja lapangan dalam pencatatan hasil monitoring, mengurangi penggunaan kertas berita acara, dan mempercepat proses pelaporan. Dengan menggunakan teknologi informasi khususnya aplikasi *mobile*, penelitian ini bertujuan untuk menciptakan solusi yang dapat membantu PT.PLN (Persero) UP3 Metro dalam mengelola data monitoring dengan lebih baik. Penelitian ini menggunakan metode *waterfall* dalam pengembangan aplikasi dan melibatkan berbagai alat seperti *android studio* dan *software* penunjang lainnya untuk mencapai tujuan tersebut. Proses pengujian aplikasi ini menggunakan metode *user acceptance test* dan *black box testing*. Hasil penelitian menunjukkan bahwa aplikasi monitoring ini telah berhasil dikembangkan dengan menggunakan metode *waterfall*. Pengujian menggunakan metode *user acceptance test* pada penelitian ini melibatkan 4 orang karyawan UP3 Metro bagian distribusi pelanggan dengan 4 *smartphone* berbeda yang menghasilkan bahwa aplikasi ini tidak terdapat *error* ataupun kendala sehingga aplikasi berjalan dengan baik.

Kata kunci : Aplikasi *Mobile*, Monitoring Operasi P2TL, Pengukuran Tidak Langsung, PT. PLN (Persero), *Waterfall*.

ABSTRACT

DEVELOPMENT APPLICATION OF INDIRECT 3 PHASES P2TL OPERATION MONITORING AT PT. PLN (PERSERO) METRO CUSTOMER SERVICE EXECUTION (UP3)

By

ARYA DILLAH

In order to improve the efficiency and effectiveness of documentation of indirect 3 phases P2TL operation monitoring activities, this research develops a mobile-based application. This application is designed to facilitate field workers in recording monitoring results, reduce the use of paper minutes, and expedite the reporting process. By utilizing information technology, especially mobile applications, this research aims to create a solution that can assist PT.PLN (Persero) UP3 Metro in managing monitoring data more effectively. This research uses the waterfall method in application development and involves various tools such as Android Studio and other supporting software to achieve these goals. The testing process of this application uses user acceptance test method and black box testing. The research results show that this monitoring application has been successfully developed using the waterfall method. Testing using the user acceptance test method in this research involves 4 UP3 Metro employees from the customer distribution department with 4 different smartphones, resulting in no errors or obstacles, thus the application runs smoothly.

Keywords: Mobile Application, P2TL Operation Monitoring, Indirect Measurement, PT. PLN (Persero), Waterfall.

Judul Skripsi : **RANCANG BANGUN APLIKASI
MONITORING OPERASI P2TL
PENGUKURAN TIDAK LANGSUNG 3
PHASA DI PT. PLN (PERSERO) UNIT
PELAKSANA PELAYANAN
PELANGGAN (UP3) METRO**

Nama Mahasiswa : **Arya Dillah**

Nomor Pokok Mahasiswa : **1715061008**

Program Studi : **Teknik Informatika**

Fakultas : **Teknik**

MENYETUJUI

1. Komisi Pembimbing

Pembimbing Utama

Pembimbing Pendamping

Ir. Gigh Forda Nama, S.T., M.T.I., I.P.M.

Deny Budiyanto, S.Kom., M.T.

NIP. 19830712 200812 1 003

NIP. 19911208 201903 1 011

1. Mengetahui

Ketua Jurusan

Ketua Program Studi

Teknik Elektro

Teknik Elektro

Herlinawati, S.T., M.T.

Yessi Mulyani, S.T., M.T.

NIP. 19710314 199903 2 001

NIP. 19731226 200012 2 001

MENGESAHKAN

1. **Tim Penguji**

Ketua : Ir. Gigih Forda Nama, S.T., M.T.I., I.P.M.

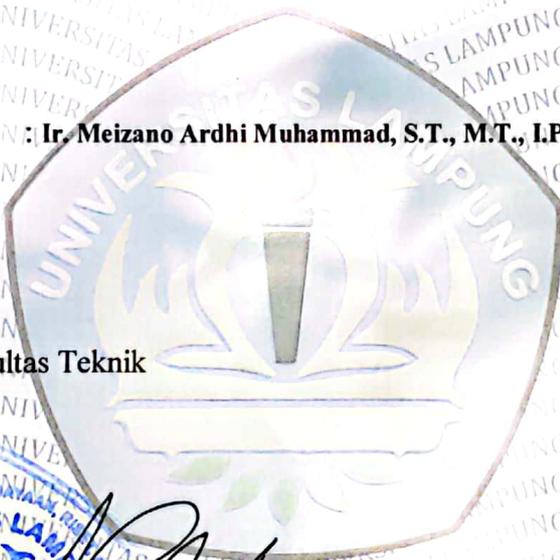
Sekretaris : Deny Budiyanto, S.Kom., M.T.

Penguji : Ir. Meizano Ardhi Muhammad, S.T., M.T., I.P.M.

2. **Dekan Fakultas Teknik**

Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc.

NIP. 19750928 200112 1 002



[Handwritten signatures of Ir. Gigih Forda Nama, Deny Budiyanto, and Ir. Meizano Ardhi Muhammad]

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi dengan judul “Rancang Bangun Aplikasi Monitoring Operasi P2TL Pengukuran Tidak Langsung 3 Phasa di PT. PLN (Persero) Unit Pelaksana Pelayanan Pelanggan (UP3) Metro” dibuat oleh saya sendiri. Semua hasil yang terdapat dalam skripsi ini telah mengikuti kaidah penulisan karya tulis ilmiah Universitas Lampung. Apabila pada kemudian hari terbukti bahwa skripsi ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi dengan ketentuan hukum atau akademik yang berlaku.

Bandar Lampung, 25 April 2024

Yang membuat pernyataan,



Arya Dillah

1715061008

RIWAYAT HIDUP



Penulis lahir di Pasar Baru Kedondong pada tanggal 4 Juni 1998 dari pasangan Bapak Muhammad Nasir dan Ibu Mariah. Penulis telah menyelesaikan pendidikan formal di SD Negeri 1 Pasar Baru pada tahun 2010, MTs Negeri 1 Pesawaran pada tahun 2013, dan MAN 1 Pesawaran pada tahun 2016. Pada tahun 2017 terdaftar sebagai mahasiswa Program Studi Teknik Informatika, Jurusan Teknik Elektro, Fakultas Teknik Universitas Lampung melalui jalur SBMPTN (Seleksi Bersama Masuk Perguruan Tinggi Negeri). Selama menjadi mahasiswa penulis melakukan beberapa kegiatan antara lain :

1. Mengikuti program *Electrical Goes To School* yaitu pengenalan jurusan Teknik Elektro dan Teknik Informatika kepada siswa Sekolah Menengah Atas pada setiap kabupaten pada Januari 2018.
2. Mengikuti kegiatan Basic Training Himpunan Mahasiswa Islam Cabang Bandar Lampung Komisariat Teknik Universitas Lampung pada Mei 2018
3. Mengikuti program Survei Opini Publik Provinsi Lampung yang dilakukan oleh *Cyrus Network* pada Mei 2018.
4. Mengikuti program Studi Elektrifikasi yang dilakukan di kabupaten Tulang Bawang bersama PT. PLN (Persero) Unit Induk Distribusi Lampung bekerjasama dengan LPEM-FEB Universitas Indonesia untuk mengetahui kelayakan subsidi konsmen pemakaian listrik pada tahun Juli 2018.
5. Mengikuti program Jejak Pendapat Ekonomi, Sosial dan Politik Masyarakat Lampung yang dilakukan oleh PT. Charta Politika Indonesia pada September 2018.

6. Mengikuti kegiatan Survei Dapil Lampung Pemilu serentak 2019 yang dilakukan oleh lembaga survey Pakar Indonesia pada Oktober 2018.
7. Mengikuti kegiatan *International Business Integrity Conference (IBIC)* Hari Anti Korupsi Sedunia yang dilakukan oleh Komisi Pemberantasan Korupsi pada Desember 2018.
8. Mengikuti kegiatan Survei Opini Publik Preferensi Politik yang dilakukan Indikator Politik Indonesia pada Desember 2018.
9. Menjadi *Junior Engineer* di PT. Pelayanan Listrik Bumi Siger Januari-Mei 2019.
10. Mengikuti kegiatan Survei Penyelenggaraan Pemerintah Daerah dan Pemilihan Kepala Daerah Kabupaten Lampung Tengah yang dilakukan oleh Citra Research Center pada Juli 2019.
11. Mengikuti kegiatan Survei Opini Publik Tentang Ekonomi, Sosial dan Politik yang dilakukan oleh Polmark Research Center pada Agustus 2019.
12. Mengikuti kegiatan Survei Masalah Sosial Kemasyarakatan Kabupaten Pesawaran yang dilakukan oleh *Indonesia Election Survey (IES)* September 2019.
13. Mengikuti kegiatan *Focus Group Discussion* Pengembangan Pemasaran Pariwisata Di Lampung yang dilakukan oleh Kementrian Pariwisata Dan Ekonomi Kreatif pada Oktober 2019.
14. Mengikuti kegiatan Survei Preferensi Masyarakat Daerah Lampung yang dilakukan oleh Cirus Surveyors Group pada Desember 2019.
15. Melakukan kegiatan Survei Permasalahan Ekonmoi, Sosial dan Politik Masyarakat Lampung yang dilakukan oleh Lembaga Riset Indonesia pada Desember 2019.
16. Melakukan kegiatan Survei *Centre For Strategic And International Studies* Kontribusi Ekonomi Grab Indonesia Survei Mitra Grab di 12 Kota di Indonesia pada Januari 2020.
17. Melakukan kegiatan Survei Masalah Sosial Kemasyarakatan Kabupaten Lampung Timur yang dilakukan oleh Indo Barometer pada Januari 2020.

18. Melakukan kegiatan survei Masalah Sosial Kemasyarakatan Kabupaten Lampung Selatan yang dilakukan oleh *Saifulmujani Research and Consulting* pada Februari 2020.
19. Mengikuti kegiatan *Intermediate Training* Tingkat Nasional Himpunan Mahasiswa Islam Cabang Palembang pada Februari 2020.
20. Melakukan kegiatan Survei Masalah Sosial Kemasyarakatan Di Kabupaten Pesawaran yang dilakukan oleh *Pandawa Research* pada Maret 2020.
21. Melakukan kegiatan Survei Masalah Sosial Kemasyarakatan Di Kabupaten Lampung Selatan yang dilakukan oleh Lembaga Survei Indonesia pada Maret 2020.
22. Melakukan kegiatan Survei Kondisi Sosial, Ekonomi dan Politik di Pulau Pahawang Kabupaten Pesawaran yang dilakukan oleh Sinergi Data Indonesia pada Juni 2020.
23. Melakukan kegiatan Survei Dampak Covid-19 Bagi Kegiatan Perekonomian Masyarakat Kabupaten Lampung Barat yang dilakukan oleh Universitas Prof.Dr.Moestopo Jakarta FIKOM MOESTOPO pada Juli 2020.
24. Melakukan kegiatan Sosialisasi dan Konsultasi Publik Studi Larap Modernisasi Daerah Irigasi Way Sekampung yang dilakukan oleh Kementrian Pekerjaan Umum Dan Perumahan Rakyat Direktorat Jendral Sumber Daya Air Direktorat Irigasi dan Rawa Satuan Kerja Direktorat Irigasi Dan Rawa pada Desember 2020.
25. Mengikuti kegiatan *Training of Trainer Regional* Badan Pengelola Latihan Himpunan Mahasiswa Islam Cabang Bandar Lampung pada Juni 2021.
26. Melaksanakan Kuliah Kerja Nyata pada bulan Februari – Maret 2021 di Desa Babakan Loa Kecamatan Kedondong Kabupaten Pesawaran.
27. Melaksanakan Kerja Praktek di PT. PLN (Persero) Unit Pelaksana Pelayanan Pelanggan Metro pada April-Mei 2021.
28. Mengikuti kegiatan Survei Sanitasi Rumah Tangga sebagai Mid-Term Survey Program Wash SDGS di Bandar Lampung pada Mei-Juni 2021.
29. Menjadi Pemateri dalam kegiatan LKMM-TD Mahasiswa Program Studi Arsitektur, Fakultas Teknik, Universitas Lampung pada November 2021.

30. Menjadi Pemateri dalam kegiatan LKMM-TD Mahasiswa Program Studi Teknik Sipil, Fakultas Teknik, Universitas Lampung pada November 2021.
31. Menjadi petugas monitoring P2TL PT. PLN (Persero) Unit Pelaksana Pelayanan Pelanggan Metro pada Desember 2021 – Mei 2022.
32. Menjadi Pembicara dalam Latihan Kepemimpinan 1 (LK1) Badan Eksekutif Mahasiswa (BEM) FMIPA-Farmasi Universitas Tulang Bawang Lampung Pada Januari 2022.
33. Mengikuti kegiatan Survei Preferensi Masyarakat dari Segi Ekonomi, Sosial dan Politik yang dilakukan oleh Indo Riset Survei pada Juni 2022.
34. Menjadi Pemateri dalam kegiatan Membangun Harmonisasi Kepengurusan Himpunan Mahasiswa Ilmu Komunikasi Universitas Tulang Bawang Lampung Yang Progresif dan Inovatif pada Juni 2022.
35. Mengikuti Survei Daerah Lampung Dalam Pemilihan Umum 2024 yang dilaksanakan oleh Politika Research and Consulting pada Oktober 2023.
36. Mengikuti Survei Moderasi Beragama yang dilakukan oleh Kementerian Agama Republik Indonesia di Wilayah Kerja Universitas Islam Negeri Lampung pada November 2023.
37. Menjadi Asisten Kordinator Indo Riset Survei Wilayah Lampung pada Desember 2023 – Februari 2024.

MOTTO

“Tetaplah menjadi teduh, tidak ada yang permanen semua kondisi akan berubah
kapanpun”

(Arya Dillah)

“Sesungguhnya sesudah kesulitan itu ada kemudahan”

(Al-Insyirah : 6)

SANWACANA

Puji dan syukur atas kehadiran Allah SWT, karena berkat rahmat-Nya yang tak pernah terputus kepada penulis, sehingga penulis dapat menyelesaikan skripsi ini dengan selesai. Salawat serta salam penulis sangjung agungkan kepada junjungan sang pembawa kebenaran Rasulullah Muhammad SAW yang penulis harapkan syafaatnya di hari akhir kelak.

Skripsi yang berjudul “Rancang Bangun Aplikasi Monitoring Operasi P2TL Pengukuran Tidak Langsung 3 Phasa di PT. PLN (Persero) Unit Pelaksana Pelayanan Pelanggan (UP3) Metro” yang disusun sebagai salah satu syarat memperoleh gelar sarjana teknik pada Program Studi Teknik Informatika, Universitas Lampung.

Dalam proses penelitian ini, sangat banyak orang-orang terlibat dalam pelaksanaannya. Oleh sebab itu, penulis mengucapkan terimakasih yang sebesar-besarnya kepada :

1. Allah SWT yang telah memberikan banyak nikmat, bantuan dan pertolongan dalam sepanjang hidup penulis, terutama dalam pembuatan skripsi ini.
2. Kedua orang tua dan keluarga yang telah memberikan support dan doa kepada penulis.
3. Bapak Dr. Eng. Helmy Fitriawan, S.T., M.Sc., selaku Dekan Fakultas Teknik Universitas Lampung.
4. Ibu Herlinawati, S.T., M.T., selaku Ketua Jurusan Teknik Elektro Universitas Lampung.
5. Ibu Yessi Mulyani, S.T., M.T. selaku Ketua Program Studi Teknik Informatika Universitas Lampung yang telah membantu proses kelancaran pengerjaan skripsi.
6. Bapak Ir. H, Gigih Forda Nama, S.T., M.T., I.P.M. selaku pembimbing utama yang telah membimbing dan banyak membantu dalam melaksanakan proses penelitian ini.

7. Bapak Deny Budiyanto, S.Kom., M.T. selaku pembimbing pendamping yang telah banyak memberikan masukan dalam proses penelitian ini.
8. Bapak Ir. Meizano Ardhi Muhammad, S.T., M.T., I.P.M. selaku penguji yang telah memberikan banyak pelajaran kepada penulis terkait dengan pelaksanaan penelitian ini.
9. Bapak Mona Arif Muda, S.T., M.T. selaku dosen pembimbing akademik yang selalu membantu dan memberikan dukunga dalam proses perkuliahan.
10. Seluruh dosen dan staff Program Studi Teknik Informatika Unila yang memberi masukan dan membantu mempermudah pembuatan skripsi serta dalam proses perkuliahan.
11. Kawan angkatan 2017 yang selalu ada dalam proses penyelesaian penelitian dan menemani semasa menjalani proses perkuliahan.
12. Kawan seperjuangan Komisariat Teknik Unila tempat bertukar cerita yang selalu ada dalam proses pertumbuhan semasa menjalani proses perkuliahan.
13. Pihak PT. PLN Persero Unit Pelaksana Pelayanan Pelanggan Metro yang telah membantu penulis selama melaksanakan penelitian dan selalu berbagi ilmu baru.
14. Serta semua pihak yang pernah membantu penulis dalam hal moral dan moril yang tidak bisa penulis sebutkan satu persatu.

Semoga seluruh kebaikan yang telah diberikan mendapatkan balasan kebaikan dari Tuhan Yang Maha Esa. Penulis menyadari bahwa penelitian ini masih banyak kekurangannya, baik dari segi penulisan maupun sistem yang dikembangkan. Oleh karena itu penulis memohon maaf dan menerima kritik terhadap aoa yang telah penulis tuangkan dalam karya ilmiah skripsi ini.

Bandar Lampung 25 April 2024

Arya Dillah

1715061008

DAFTAR ISI

| | Halaman |
|---|----------------|
| DAFTAR ISI | ix |
| DAFTAR GAMBAR | xii |
| DAFTAR TABEL | xv |
| DAFTAR NOTASI DAN ISTILAH..... | xvii |
| I. PENDAHULUAN..... | 1 |
| 1.1. Latar Belakang..... | 1 |
| 1.2. Tujuan Penelitian | 4 |
| 1.3. Manfaat Penelitian | 4 |
| 1.4. Perumusan Masalah | 4 |
| 1.5. Batasan Masalah | 4 |
| 1.6. Hipotesis..... | 5 |
| 1.7. Sistematika Penulisan | 5 |
| II. TINJAUAN PUSTAKA | 7 |
| 2.1. Penelitian Terkait..... | 7 |
| 2.2. Rancang Bangun..... | 12 |
| 2.3. Aplikasi | 12 |
| 2.4. Monitoring..... | 12 |
| 2.5. Penertiban Pemakaian Tenaga Listrik (P2TL) | 13 |
| 2.6. Pengukuran Energi Listrik | 14 |
| 2.7. Perancangan..... | 15 |

| | | |
|----------------------------------|---|----|
| 2.8. | <i>Android</i> | 16 |
| 2.9. | <i>Database</i> | 16 |
| 2.10. | <i>Android Studio</i> | 17 |
| 2.11. | <i>Visual Studio Code</i> | 18 |
| 2.12. | <i>Flutter</i> | 19 |
| 2.13. | <i>Firebase</i> | 20 |
| 2.14. | <i>Software Development Life Cycle (SDLC)</i> | 21 |
| 2.15. | <i>Unified Modeling Language (UML)</i> | 24 |
| 2.16. | <i>Black Box Testing</i> | 30 |
| 2.17. | <i>User Acceptance Testing (UAT)</i> | 30 |
| III. METODOLOGI PENELITIAN | | 32 |
| 3.1. | Waktu dan Tempat Penelitian | 32 |
| 3.2. | Perangkat Penelitian | 33 |
| 3.2.1. | Perangkat Keras (<i>Hardware</i>) | 33 |
| 3.2.2. | Perangkat Lunak (<i>Software</i>)..... | 33 |
| 3.3. | Tahapan Penelitian..... | 34 |
| 3.4. | Metode Penelitian | 34 |
| 3.5. | Prosedur Pengembangan..... | 36 |
| 3.5.1. | Analisis Kebutuhan (<i>Requirement Definition</i>)..... | 36 |
| 3.5.2. | Desain Sistem (<i>System and Software Design</i>) | 38 |
| 3.6. | Metode Pengujian | 69 |
| 3.7. | Penulisan Laporan | 70 |
| IV. HASIL DAN PEMBAHASAN | | 71 |
| 4.1 | Implementasi Antar Muka | 71 |
| 4.1.1. | Tampilan <i>Login</i> Aplikasi | 71 |
| 4.1.2. | Tampilan Registrasi Aplikasi..... | 75 |

| | |
|--|-----|
| 4.1.3. Tampilan Lupa <i>Password</i> | 76 |
| 4.1.4 Tampilan <i>Home Screen</i> | 79 |
| 4.1.5 Tampilan Input Data..... | 81 |
| 4.1.6 Tampilan Edit Data..... | 85 |
| 4.1.7. Tampilan Hapus Data | 90 |
| 4.1.8. Tampilan Hasil Data | 91 |
| 4.1.9. Tampilan <i>Logout</i> | 95 |
| 4.2 Pengujian Sistem Aplikasi..... | 96 |
| 4.2.1. Prosedur Pengujian | 97 |
| 4.2.2. Hasil Pengujian | 97 |
| V. PENUTUP | 110 |
| 5.1 Kesimpulan | 110 |
| 5.2 Saran..... | 110 |
| DAFTAR PUSTAKA | 111 |
| LAMPIRAN | 115 |

DAFTAR GAMBAR

| | |
|---|----|
| Gambar 2.1 <i>Emsyst Electricity</i> | 14 |
| Gambar 2.2 <i>Android Studio</i> | 17 |
| Gambar 2.3 <i>Visual Studio Code</i> | 18 |
| Gambar 3.1 Pengembangan Perangkat Lunak Berurutan/Linear | 33 |
| Gambar 3.2 Rancangan <i>Use Case Diagram</i> | 48 |
| Gambar 3.3 <i>Sequence Diagram Login</i> | 49 |
| Gambar 3.4 <i>Sequence Diagram Registrasi</i> | 49 |
| Gambar 3.5 <i>Sequence Diagram Lupa Password</i> | 50 |
| Gambar 3.6 <i>Sequence Diagram Home Screen</i> | 50 |
| Gambar 3.7 <i>Sequence Diagram Input Data</i> | 51 |
| Gambar 3.8 <i>Sequence Diagram Edit Data</i> | 51 |
| Gambar 3.9 <i>Sequence Diagram Hapus Data</i> | 52 |
| Gambar 3.10 <i>Sequence Diagram Tampil Data</i> | 52 |
| Gambar 3.11 <i>Sequence Diagram Logout</i> | 53 |
| Gambar 3.12 <i>Activity Diagram Login</i> | 54 |
| Gambar 3.13 <i>Activity Diagram Registrasi</i> | 54 |
| Gambar 3.14 <i>Activity Diagram Lupa Password</i> | 55 |
| Gambar 3.15 <i>Activity Diagram Home Screen</i> | 55 |
| Gambar 3.16 <i>Activity Diagram Input Data</i> | 56 |
| Gambar 3.17 <i>Activity Diagram Edit Data</i> | 56 |
| Gambar 3.18 <i>Activity Diagram Hapus Data</i> | 57 |
| Gambar 3.19 <i>Activity Diagram Tampil Data</i> | 57 |
| Gambar 3.20 <i>Activity Diagram Logout</i> | 58 |
| Gambar 3.21 Desain Tampilan Halaman <i>Login</i> Aplikasi | 59 |
| Gambar 3.22 Desain Tampilan Halaman <i>Registrasi</i> Aplikasi..... | 60 |

| | |
|--|----|
| Gambar 3.23 Desain Tampilan Halaman Lupa <i>Password</i> | 61 |
| Gambar 3.24 Desain Tampilan Halaman <i>Input Data</i> | 62 |
| Gambar 3.25 Desain Tampilan Halaman <i>Edit Data</i> | 63 |
| Gambar 3.26 Desain Tampilan Halaman Hapus Data | 64 |
| Gambar 3.27 Desain Tampilan Halaman Tampil Hasil Data..... | 65 |
| Gambar 3.28 Desain Tampilan Halaman <i>Logout</i> | 66 |
| Gambar 4.1 <i>Source Code Login</i> Aplikasi | 69 |
| Gambar 4.2 Tampilan <i>Login</i> Aplikasi | 71 |
| Gambar 4.3 <i>Login</i> Berhasil | 72 |
| Gambar 4.4 <i>Login</i> Gagal..... | 72 |
| Gambar 4.5 <i>Source Code</i> Registrasi | 73 |
| Gambar 4.6 Tampilan Registrasi Aplikasi | 74 |
| Gambar 4.7 <i>Source Code</i> Lupa <i>Password</i> | 75 |
| Gambar 4.8 Menu Lupa <i>Password</i> | 76 |
| Gambar 4.9 Form <i>Reset Password</i> | 76 |
| Gambar 4.10 <i>Source Code</i> Home Screen | 77 |
| Gambar 4.11 Tampilan Home Screen | 79 |
| Gambar 4.12 <i>Source Code</i> <i>Input Data</i> | 80 |
| Gambar 4.13 Tampilan <i>Input Data</i> | 81 |
| Gambar 4.14 Tampilan <i>Input Data</i> Lanjutan 1 | 81 |
| Gambar 4.15 Tampilan <i>Input Data</i> Lanjutan 2..... | 82 |
| Gambar 4.16 Tampilan <i>Input Data</i> Lanjutan 3..... | 82 |
| Gambar 4.17 Tampilan <i>Input Data</i> Lanjutan 4..... | 83 |
| Gambar 4.18 <i>Source Code</i> <i>Edit Data</i> | 84 |
| Gambar 4.19 Tampilan <i>Edit Data</i> | 85 |
| Gambar 4.20 Tampilan <i>Edit Data</i> Lanjutan 1 | 85 |
| Gambar 4.21 Tampilan <i>Edit Data</i> Lanjutan 2 | 86 |
| Gambar 4.22 Tampilan <i>Edit Data</i> Lanjutan 3 | 87 |
| Gambar 4.23 Tampilan <i>Edit Data</i> Lanjutan 4 | 87 |
| Gambar 4.24 <i>Source Code</i> Hapus Data | 88 |
| Gambar 4.25 Tampilan Hapus Data | 89 |
| Gambar 4.26 <i>Source Code</i> Tampilan Hasil Data | 90 |

| | |
|--|----|
| Gambar 4.27 Tampilan Hasil Data | 91 |
| Gambar 4.28 Tampilan Hasil Data Lanjutan 1 | 91 |
| Gambar 4.29 Tampilan Hasil Data Lanjutan 2 | 92 |
| Gambar 4.30 Tampilan Hasil Data Lanjutan 3 | 92 |
| Gambar 4.31 Tampilan Hasil Data Lanjutan 4 | 93 |
| Gambar 4.32 <i>Source Code Logout</i> | 94 |
| Gambar 4.33 Tampilan <i>Logout</i> | 94 |

DAFTAR TABEL

| | |
|--|-----|
| Tabel 2.1 <i>Use Case Diagram</i> | 23 |
| Table 2.2 <i>Squence Diagram</i> | 25 |
| Tabel 2.3 <i>Class Diagram</i> | 26 |
| Tabel 2.4 <i>Activity Diagram</i> | 27 |
| Tabel 3.1 Alur Penelitian | 30 |
| Tabel 3.2 Definisi Aktor <i>User</i> | 37 |
| Tabel 3.3 Definisi <i>Use Case</i> | 38 |
| Tabel 3.4 Skenario <i>Use Case Login</i> | 40 |
| Tabel 3.5 Skenario <i>Use Case Registrasi</i> | 41 |
| Tabel 3.6 Skenario <i>Use Case Lupa Password</i> | 42 |
| Tabel 3.7 Skenario <i>Use Case Home Screen</i> | 42 |
| Tabel 3.8 Skenario <i>Use Case Input Data</i> | 43 |
| Tabel 3.9 Skenario <i>Use Case Edit Data</i> | 44 |
| Tabel 3.10 Skenario <i>Use Case Hapus Data</i> | 46 |
| Tabel 3.11 Skenario <i>Use Case Tampil Data</i> | 46 |
| Tabel 3.12 Skenario <i>Use Case Logout</i> | 47 |
| Tabel 4.1 Pengujian menu <i>Login</i> | 96 |
| Tabel 4.2 Pengujian menu Registrasi..... | 96 |
| Tabel 4.3 Pengujian menu <i>Lupa Password</i> | 97 |
| Tabel 4.4 Pengujian menu <i>Home Screen</i> | 97 |
| Tabel 4.5 Pengujian menu <i>Input Data</i> | 98 |
| Tabel 4.6 Pengujian menu <i>Edit Data</i> | 98 |
| Tabel 4.7 Pengujian menu Hapus Data..... | 99 |
| Tabel 4.8 Pengujian menu Tampil Data..... | 100 |
| Tabel 4.9 Pengujian menu <i>Logout</i> | 100 |

| | |
|--|-----|
| Tabel 4.10 Kriteria skala <i>likert</i> | 101 |
| Tabel 4.11 Tabel skenario pengujian kuisisioner..... | 102 |
| Tabel 4.12 Tabel pengujian kuisisioner nomor 1..... | 103 |
| Tabel 4.13 Tabel pengujian kuisisioner nomor 2..... | 103 |
| Tabel 4.14 Tabel pengujian kuisisioner nomor 3..... | 104 |
| Tabel 4.15 Tabel pengujian kuisisioner nomor 4..... | 105 |
| Table 4.16 Tabel pengujian kuisisioner nomor 5..... | 105 |
| Tabel 4.17 Tabel pengujian kuisisioner nomor 6..... | 106 |
| Tabel 4.18 Tabel pengujian kuisisioner nomor 7..... | 107 |

DAFTAR NOTASI DAN ISTILAH

| | |
|------|---|
| PLN | = Perusahaan Listrik Negara |
| P2TL | = Penertiban Pemakaian Tenaga Listrik adalah rangkaian kegiatan yang meliputi perencanaan, pemeriksaan, tindakan dan penyelesaian yang dilakukan oleh PLN terhadap instalasi PLN dan/atau instalasi pemakai tenaga listrik dari PLN |
| Kwh | = <i>Kilowatt Hour</i> |
| VA | = <i>Volt Ampere</i> |
| CT | = <i>Current Transformer</i> |
| UP3 | = Unit Pelaksana Pelayanan Pelanggan |
| SDLC | = <i>System Development Life Cycle</i> |
| UML | = <i>Unified Modeling Language</i> |
| VT | = <i>Voltage Transformer</i> |
| IDE | = <i>Integrated Development Environment</i> |
| OS | = <i>Operation System</i> |
| SDK | = <i>Software Development Kit</i> |
| API | = <i>Application Programming Interface</i> |
| HTTP | = <i>Hypertext Transfer Protocol</i> |
| CND | = <i>Certified Network Defender</i> |
| SNL | = <i>Standardizes Nursing Language</i> |
| UAT | = <i>User Acceptance Testing</i> |
| UI | = <i>User Interface</i> |
| IR | = Arus (R) 3 Phasa |
| IS | = Arus (S) 3 Phasa |
| IT | = Arus (T) 3 Phasa |
| VR | = Tegangan (R) 3 Phasa |

| | |
|-----|--|
| VS | = Tegangan (S) 3 Phasa |
| VT | = Tegangan (T) 3 Phasa |
| CTR | = <i>Current Transformer</i> (R) 3 Phasa |
| CTS | = <i>Current Transformer</i> (S) 3 Phasa |
| CTT | = <i>Current Transformer</i> (T) 3 Phasa |

I. PENDAHULUAN

1.1. Latar Belakang

Kebutuhan energi listrik semakin meningkat pesat seiring perkembangan teknologi, sedangkan masyarakat sebagai konsumen energi listrik juga selalu bertambah jumlah penggunaannya dan menuntut mutu serta kualitas yang harus di penuhi terhadap konsumen energi listrik untuk permasalahan pelayanan yang lebih baik secara berkelanjutan.

Dokumentasi yang jelas sangat dibutuhkan dalam setiap kegiatan pekerjaan di lapangan. Salah satu bentuk dokumentasi penting dari kegiatan adalah monitoring operasi, karena monitoring operasi P2TL pengukuran tidak langsung 3 phasa sangat penting bagi perusahaan. Pada kwh 3 phasa harus selalu dilakukan pemeriksaan secara berkala. Monitoring operasi harus dilakukan dengan teliti dan minim kesalahan karena informasi dari monitoring operasi tersebut dapat menjadi penentu baiknya diagram fasor baik arus dan tegangan pada setiap kwh 3 phasa.

Teknologi sangat berperan penting dalam kehidupan manusia. Tidak dapat dipungkiri lagi, kehadiran teknologi telah membawa pengaruh cukup besar dalam kehidupan manusia. Teknologi juga dapat mempermudah manusia dalam melakukan pekerjaan sehari-hari dengan lebih cepat dan mudah. Teknologi informasi merupakan teknologi yang dapat berfungsi untuk mengolah atau memproses data, menghasilkan, menyimpan, dan mentransmisikan data untuk menghasilkan informasi yang berkualitas. Teknologi yang sering digunakan salah satunya adalah aplikasi *mobile* (*Mobile Application*), Aplikasi *Mobile* adalah sebuah aplikasi yang memungkinkan untuk melakukan mobilitas dengan menggunakan

perlengkapan seperti telpon genggam (*hanphone*) atau *smartphone*. Aplikasi *mobile* dapat mengakses dan menggunakan sebuah aplikasi web secara *nirkabel* dengan menggunakan perangkat *mobile*, dimana data yang diperoleh hanyalah berupa teks sehingga tidak perlu membutuhkan *bandwidth* yang terlalu besar. Data dalam bentuk teks tidak semerarik data dalam bentuk teks disertai dengan gambar. Namun penggunaan data teks adalah yang paling banyak digunakan saat ini. Menggunakan aplikasi *mobile* hanya memerlukan *handphone* yang sudah dilengkapi dengan fasilitas GPRS dan koneksinya [1].

Kegiatan monitoring Penertiban Pemakaian Listrik (P2TL) Kwh 3 Phasa di PT. PLN (Persero) UP3 Metro dilakukan secara rutin setiap 3 bulan, terdapat kurang lebih 250 titik kwh 3 phasa yang berada di wilayah metro maka monitoring dilakukan dalam sehari mencapai target 4 titik Kwh. Untuk kwh 3 phasa yang dilakukan monitoring yaitu kwh dengan spesifikasi diatas 6.600 VA atau untuk kegunaan industri.

Kegiatan monitoring meliputi, pengecekan segel kotak kwh dipastikan dalam keadaan rusak atau tidak, id palanggan, nama pelanggan, golongan tarif, daya, alamat pelanggan, tanggal periksa, merk kwh, merk CT, Rasio CT, Keterangan Box, arus premier, arus sekunder, tegangan uji, diagram phasor, eror CT, eror kwh, untuk menentukan bahwa penggunaan listrik yaitu pelanggan masih di tertib pemakaian dan aman.

Pada kegiatan monitoring biasanya selalu dilakukan pembuatan data pelaporan hasil monitoring menggunakan kertas berita acara dan dalam pelaksanaan terkadang menambah pekerjaan selain harus membawa alat ukur pekerja lapangan juga harus membawa alat tulis untuk pencatatan hasil monitong dimana pencatatan ini bisa menambah pekerjaan. Maka dari itu penulis mempunyai gagasan untuk merancang suatu aplikasi monitong berbasis *mobile* dimana pekerja lapangan nantinya tidak lagi perlu memcatatat hasil monitong menggunakan kertas berita acara hanya perlu mengiput melalui aplikasi, selain tidak diperlukan lagi membawa alat tulis

ke lapangan dan juga dapat mempermudah pelaporan hasil monitoring serta bisa mengurangi penggunaan kertas berita acara.

Maka dari itu penulis memilih melaksanakan penelitian tugas akhir di PT.PLN (Persero) UP3 Metro yang berlokasi di Jalan Letjen Alamsyah Ratu Prawira Negara Metro, Kec. Metro Pusat, Kota Metro, Lampung untuk mengetahui dan mempelajari suatu pekerjaan yang ada di unit distribusi khususnya monitoring P2TL pengukuran tidak langsung 3 phasa. Seringkali mengalami permasalahan antara lain hasil monitoring yang masih menggunakan cara konvensional dengan menggunakan banyak kertas sebagai berita acara dan penginputan data hasil monitoring yang masing manual dalam penginputan data membuat pekerjaan yang masih kurang efektif dan efisien.

Berdasarkan permasalahan yang ada suatu perusahaan membutuhkan sistem yang terkomputerisasi dan memerlukan *database* untuk memudahkan penyimpanan data dan mengakses data jika diperlukan sewaktu – waktu guna menghasilkan informasi yang berguna bagi perusahaan. Sebuah aplikasi *mobile* yang bisa digunakan dimana saja, efektif, efisien dan memberikan data lengkap. Untuk itu penulis merancang suatu aplikasi yang berfungsi untuk monitoring P2TL pengukuran tidak langsung 3 phasa menggunakan aplikasi berbasis *mobile*.

Dengan aplikasi ini diharapkan dapat membantu mempermudah perusahaan dan karyawan dalam membuat laporan monitoring P2TL pengukuran tidak langsung 3 phasa agar menjadi tertata secara efektif dan efisien, berdasarkan uraian latar belakang diatas penulis tertarik untuk membuat penelitian dengan judul **“Rancang Bangun Aplikasi Monitoring Operasi P2TL Pengukuran Tidak Langsung 3 Phasa di PT. PLN (Persero) Unit Pelaksana Pelayanan Pelanggan (UP3) Metro.**

1.2. Tujuan Penelitian

Tujuan penelitian ini adalah menghasilkan rancang bangun aplikasi monitoring berbasis *mobile* yang dapat membantu pengambilan dokumentasi P2TL pengukuran tidak langsung 3 phasa di PT. PLN (Persero) UP3 Metro bagian distribusi pelanggan.

1.3. Manfaat Penelitian

Manfaat dari penelitian ini adalah membantu pengambilan dokumentasi operasi P2TL pengukuran tidak langsung 3 phasa di PT. PLN (Persero) unit pelaksana pelayanan pelanggan (UP3) Metro bagian distribusi pelanggan dengan aplikasi berbasis android dan tidak lagi menggunakan cara konvensional dengan kertas berita acara.

1.4. Perumusan Masalah

Berdasarkan pada masalah yang ada pada latar belakang maka perumusan masalah penelitian ini adalah merancang aplikasi yang mampu digunakan untuk pengambilan dokumentasi data hasil monitoring berbasis android.

1.5. Batasan Masalah

Penelitian tugas akhir ini terdapat batasan masalah yaitu aplikasi ini dirancang sesuai kebutuhan operasi P2TL pengukuran tidak langsung 3 phasa di PT. PLN (Persero) Unit Pelaksana Pelayanan Pelanggan bagian Distribusi Pelanggan untuk pengambilan dokumentasi hasil monitoring.

1.6. Hipotesis

Aplikasi ini dapat membantu input data hasil monitoring dengan tidak lagi menggunakan kertas berita acara tetapi sudah menggunakan aplikasi *mobile* untuk mempermudah pekerjaan dilapangan agar lebih efektif dan efisien.

1.7. Sistematika Penulisan

Adapun sistematika penulisan untuk memudahkan penulisan dan pemahaman laporan pada penelitian tugas akhir ini dibagi menjadi lima bab, yaitu:

BAB I PENDAHULUAN

Menjelaskan latar belakang, tujuan penelitian, manfaat penelitian, perumusan masalah, batasan masalah, hipotesis, dan sistematika penulisan.

BAB II: TINJAUAN PUSTAKA

Berisi tentang tinjauan pustaka secara teoritis mengenai landasan dalam penelitian dan berisi literatur dari penelitian terdahulu.

BAB III: METODOLOGI PENELITIAN

Menjelaskan rancangan sistem, waktu dan tempat penelitian, alat dan bahan yang akan digunakan, serta langkah-langkah pelaksanaan penelitian.

BAB IV: PEMBAHASAN

Menjelaskan hasil dari proses perancangan, analisis dari hasil pengujian, dan pengambilan data.

BAB V: KESIMPULAN DAN SARAN

Memuat kesimpulan penelitian yang diperoleh dari pembuatan serta pengujian alat, dan saran-saran mengenai perbaikan untuk pengembangan lebih lanjut.

DAFTAR PUSTAKA**LAMPIRAN**

II. TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Penelitian terkait mengambil dari beberapa contoh penelitian yang telah dilakukan sebagai pedoman atau contoh dalam penelitian yang sedang dilakukan.

2.1.1 Penerapan Metode *Waterfall* Dalam Perancangan Sistem Informasi Aplikasi Bantuan Sosial Berbasis Android

Jurnal yang ditulis oleh Dini Silvi Purnia, Achmad Rifai dan Syaifur Rahmatullah. Merupakan penelitian tentang pembuatan aplikasi mobile *m-government* yang dibuat sebagai alternatif yang potensial untuk memperluas hubungan komunikasi pemerintah dan masyarakat yang di khususkan untuk bantuan social. Peneliti yang menggunakan metode *waterfall* dalam merancang sebuah aplikasi Bantuan Sosial berbasis android untuk melakukan pengawasan transparansi dana yang disalurkan di Dinas Sosial dan juga memepromudah masyarakat untuk menyalurkan bantuan berupa dana atau barang [2].

2.1.2 Pembangunan Aplikasi *Mobile* Pariwisata “GoTrip” Menggunakan Metode *Waterfall*

Jurnal yang ditulis oleh Ika Nurmila, Yuli Adam Prasetyo, S.T., M.T. dan Pitrascha Adytia, S.T., M.T. merupakan penelitian tentang pembuatan aplikasi mobile bernama GoTrip aplikasi yang menjadi tempat berbagi tentang lokasi-lokasi wisata. Sistem perencanaan wisata pada aplikasi GoTrip

yang telah dibuat dapat memudahkan *user* untuk melakukan perencanaan wisata dan menjadi tempat berbagai *review* lokasi wisata yang terorganisir [3].

2.1.3 Pengembangan Aplikasi *Cosycalship* Berbasis Android Untuk Pengelolaan Beasiswa Menggunakan Metode *Waterfall*

Jurnal yang ditulis oleh Hendrawan Widiyanto, Annisa Putri Pratama dan Ayu Permana Laksmi. Merupakan penelitian tentang pembuatan aplikasi *COSYCALSHIP (Complete, Easy, Practical Scholarship)* adalah sebuah aplikasi berbasis android yang menyediakan informasi, tips dan trik mendapatkan beasiswa, klasifikasi pendaftaran pendaftaran beasiswa dan pengontrolan terhadap mahasiswa penerima beasiswa dan menyajikan pelayanan terbaik bagi penggunaanya serta untuk mengatasi permasalahan antara mahasiswa dan instansi terkait pemanajemenan beasiswa [4].

2.1.4 Analisa Dan Perancangan Aplikasi *Mobile Mobile Voteme* Menggunakan Metode *Waterfall*

Jurnal yang ditulis oleh Muhammad Thoha dan Fajar Masya. Merupakan penelitian tentang menentukan produk atau jasa yang akan digunakan oleh calon wirausaha maka dibuatlah media aplikasi berbasis android yang mudah digunakan untuk membantu pengambilan keputusan kepada calon wirausaha untuk dapat menentukan produk apa yang mereka ingin pasarkan. Metode yang digunakan *System Development Life Cycle (SDLC)* dan menggunakan metode *Waterfall* (air terjun) [5].

2.1.5 Perancangan Aplikasi Pemesanan Makanan Ringan Berbasis *Object Oriented* Dengan Metode *Waterfall*

Jurnal yang ditulis oleh Faris Rosyid Ridho dan R. Soelistijadi. Merupakan penelitian tentang pembuatan aplikasi pemesanan makanan dikarenakan di UD Enggal Jaya Semarang masih menggunakan sistem yang sederhana dalam memesan ataupun membeli produk. Untuk itu diperlukan pengembangan sistem informasi sebagai salah satu solusi untuk memecahkan masalah tersebut agar mempermudah konsumen dalam memperoleh informasi maupun memesan produk yang dijual [6].

2.1.6. *Waterfall Modeling Pada Sistem E-Restorant*

Jurnal yang ditulis oleh Agung Triayudi dan Achmad Syaifudin Rodhi. Merupakan penelitian tentang pembuatan aplikasi pesan antar berupa sistem informasi berbasis android yang bertujuan untuk mempermudah dan mengoptimalkan layanan pesan antar makanan pada restoran Krebo Jantan. Aplikasi pesan antar makanan ini dirancang menggunakan *UML (Unified Modeling Language)* dibangun menggunakan bahasa pemrograman *Java* dan *MySQL* sebagai databasenya. Dalam pengembangannya menggunakan metode *waterfall* dan pengujian menggunakan *black box testing* [7].

2.1.7. *Implementation Of The Waterfall Model On Android-Based Travel Ticket Booking Applications*

Jurnal yang ditulis oleh Ganda Yoga Swara, Indra Warman dan Dede Wira Trise Putra. Merupakan penelitian tentang merancang dan mengembangkan sistem informasi berbasis android untuk bisnis travel terutama dalam proses pemasaran, pemesanan, pembayaran, keberangkatan. Yang selama ini dilakukan secara manual menjadi lebih sistematis dan menyajikan informasi yang terstruktur secara *real-time* [8].

2.1.8. Development Of Android Based Try Out Application For National Exam Of Vocational Theory In Vocational School Accompanied With Learning Outcomes Recommendation

Jurnal yang ditulis oleh Muklis Eko Apriyanto, Rosihan Ari Yuana dan Mintasih Indriayu. Merupakan penelitian tentang bagaimana mengetahui cara mengembangkan aplikasi *try out* Ujian Nasional SMK berbasis Android. Aplikasi ini juga memberikan rekomendasi sesuai hasil *try out* dan juga bertujuan untuk mengetahui kelayakan aplikasi berdasarkan penilaian oleh ahli media dan mahasiswa jurusan TKJ [9].

2.1.9. Using Waterfall Method to Design Information System of SPMI STIMIK Sepuluh Nopember Jayapura

Jurnal yang ditulis oleh Elvis Pawam, Rosiyati Thamrin, Patmawati Hasan, Sariyati H.Y. Bei, dan Paulisen Matu. Merupakan penelitian tentang bagaimana mengetahui cara pengarsipan dokumen SPMII secara konvensional merupakan kendala yang dialami oleh STIMIK Sepuluh Nopember Jayapura, pengarsipan dicara ini dapat mengakibatkan dokumen hilang atau tercecer selain itu menimbulkan kesulitan dalam pencairan dokumen pada saat internal dan audit eksternal. Penelitian ini bertujuan untuk membantu institusi dalam memperbaiki sistem pemberkasannya, agar data atau dokumen SPMI dapat disimpan dengan baik dan mudah diperoleh pada saat dibutuhkan [10].

2.1.10. Mobile Augmented Reality Media Design with Waterfall Model for Learning Geometry in College

Jurnal ini ditulis oleh Achmad Buchori, Punaji Setyosari, I wayan Dasna dan Saida Ulfa. Merupakan penelitian tentang membuat media pembelajaran yang mampu menampilkan objek *3D* dan disesuaikan secara khusus dalam pembelajaran geometri, salah satunya media *augmented reality mobile*, yang

mampu mengeluarkan tiga dimensi yang bagus dan menarik objek. Tujuan dari penelitian ini adalah membuat aplikasi media *Augmented Reality* dengan model *waterfall* dalam mata pelajaran Geometri pada mata kuliah bagi mahasiswa Program Studi Pendidikan Matematika Universitas PGRI Semarang [11].

2.1.11. *Mobile Application Development Methodologies Adopted In Omani Market: A Comparative Study*

Jurnal ini ditulis oleh Seiren Al Ratrout, Omar Husain Tarawneh, Moath Husni Altarawneh dan Mejhem Yosef Altarawneh. Merupakan penelitian tentang pendekatan pengembangan aplikasi seluler yang diadopsi di pasar Oman dan memberikan perbandingan antara metode yang ada, hasil dari penelitian ini memberikan serangkaian rekomendasi untuk aplikasi seluler pengembang yang akan membantu dalam memilih metode yang paling tepat yang sesuai dengan target pasar [12].

2.1.12. *Development of a Mobile Application for Smart Clinical Trial Subject Data Collection and Management*

Jurnal ini ditulis oleh Hyeongju Ryu, Meihua Piao, Heejin Kim, Wooseok Yang dan Kyung Hwan Kim. Merupakan penelitian tentang pembuatan perangkat yang dapat dikenakan pada teknologi digital kesehatan ketika melakukan uji klinis. Aplikasi dikembangkan untuk meningkatkan fungsi desentralisasi uji klinis dan melakukan evaluasi heuristik untuk mencerminkan tuntutan pengguna uji klinis yang ada pada pekerja. Penelitian ini menggunakan metode *waterfall* dalam pengembangannya [13].

2.2. Rancang Bangun

Rancang Bangun (desain) adalah tahapan dari setelah analisis dari siklus pengembangan sistem yang merupakan pendefinisian dari kebutuhan-kebutuhan fungsional, serta menggambarkan bagaimana suatu sistem dibentuk yang dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam suatu kesatuan yang utuh dan berfungsi, termasuk menyangkut megkonfigurasi dari komponen-komponen perangkat lunak dari suatu sistem. Rancang bangun adalah menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada baik secara keseluruhan maupun sebagian [14].

2.3. Aplikasi

Aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah – perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi tersebut, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputasi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan [15].

2.4. Monitoring

Monitoring secara umum merupakan suatu proses penilaian yang bertujuan untuk memberikan informasi tentang apa yang sedang dikerjakan. Monitoring atau pengawasan adalah mendeterminasi apa yang telah dilaksanakan, maksudnya mengevaluasi prestasi kerja dan apabila perlu, menerapkan tindakan-tindakan korektif sehingga hasil pekerjaan sesuai dengan rencana yang telah ditetapkan.

Monitoring dan pengawasan adalah pengendalian yang dilakukan dengan melaksanakan pemeriksaan, penilaian kemampuan, meningkatkan dan menyempurnakan baik manajemen maupun bidang operasionalnya.

Monitoring bagian dari kegiatan pengawasan, dalam pengawasan ada aktivitas memantau (monitoring). Pemantauan umumnya dilakukan untuk tujuan tertentu, untuk memeriksa apakah program yang telah berjalan itu sesuai dengan sasaran atau sesuai dengan tujuan program [16].

Jadi monitoring adalah proses pengamatan, pemeriksaan dan pengendalian dari suatu pekerjaan.

2.5. Penertiban Pemakaian Tenaga Listrik (P2TL)

P2TL adalah singkatan dari penertiban pemakaian tenaga listrik, yang dimaksud P2TL adalah rangkaian kegiatan meliputi perencanaan, pemeriksaan, tindakan dan penyelesaian yang dilakukan PLN terhadap aset jaringan dan proteksi milik PLN terkait adanya pemakaian listrik yang tertib. Tujuan P2TL :

1. Memberi kepastian bahwa pelanggan PLN benar-benar telah menggunakan listrik sesuai prosedur dan dengan cara yang benar
2. Sebagai upaya untuk meningkatkan mutu dan kendala pasokan listrik

Sedangkan untuk pelaksanaan dan organisasi P2TL :

1. Setiap unit PLN secara rutin atau khusus melaksanakan P2TL dalam rangka menertibkan penyaluran tenaga listrik untuk menghindari bahaya listrik bagi masyarakat, meningkatkan pelayanan dan menekan susut
2. Pelaksanaan P2TL dilakukan pada Unit Organisasi PLN berupa :
 - a. P2TL khusus tingkat nasional
 - b. P2TL khusus tingkat unit pelaksanaan induk
 - c. P2TL rutin pada unit pelaksana jenjang ketiga dan/atau kedua oleh unit pelaksana induk
 - d. P2TL rutin pada dan oleh unit pelaksana jenjang kedua
 - e. P2TL rutin pada dan oleh unit pelaksana jenjang ketiga
 - f. P2TL rutin pada dan oleh unit dibawah unit pelaksana jenjang ketiga

3. Pelaksana P2TL sebagaimana dimaksud ayat (2) huruf a ditetapkan dengan keputusan Direksi sebagai pemberi tugas dan pelaksana P2TL pada ayat (2) huruf b sampai huruf f ditetapkan dengan keputusan General Manager/Manajer Unit yang bersangkutan sebagai pemberi tugas
4. Pelaksana P2TL sebagaimana dimaksud pada ayat (2) huruf a dan huruf b hanya dapat dilakukan oleh tim, sedang pada ayat (2) huruf c sampai dengan huruf f dapat dilakukan oleh struktural maupun oleh tim
5. Pelaksana P2TL bertanggung jawab kepada pemberi tugas
6. Pelaksana P2TL dapat mengikutsertakan Penyidik Pegawai Negeri Sipil (PPNS) atau Penyidik Kepolisian Republik Indonesia atau pihak terkait lainnya [16].

2.6. Pengukuran Energi Listrik

Pengukuran listrik mempunyai tujuan untuk mengetahui besarnya listrik yang digunakan. Alat yang digunakan sebagai pembanding/penunjuk disebut instrumen pengukur, instrumen ini berfungsi untuk mengukur nilai besaran listrik yang diukur, pada sistem pengukuran energi listrik AC tiga fasa diperlukan alat ukur energi listrik yaitu kWh (*KiloWatt Hour*) meter analog atau kWh meter digital tiga fasa. Ada dua cara dalam pengukuran energi listrik tiga fasa yaitu :

3.1 Sistem pengukuran langsung

Sistem pengukuran langsung digunakan untuk pengukuran daya yang kecil dan tegangan rendah dan dalam prakteknya kWh meter langsung dihubungkan ke jala-jala dan beban yang akan diukur energi listriknya.

3.2 Sistem pengukuran tidak langsung

Sistem pengukuran tidak langsung digunakan untuk pengukuran daya besar atau tegangan tinggi dan dalam prakteknya kWh meter dihubungkan melalui alat bantu yaitu transformator tegangan (PT) atau transformator arus (CT) ke jala-jala dan beban yang akan diukur energi listriknya.

Pembacaan kWh meter pada pengukuran tidak langsung terdiri dari :

- a. Pencatatan primer, penunjukannya langsung sama dengan pada kWh meter sambungan langsung, karena perbandingan transformator sudah diperhitungkan dalam perbandingan gigi penggerak.
- b. Pencatat semi primer, hasil pembacaan harus dikalikan dengan perbandingan transformator arus untuk mendapatkan harga sebenarnya.
- c. Pencatat sekunder, hasil pembacaan harus dikalikan dengan perbandingan transformator arus dan transformator tegangan [17].

Pada UP3 Metro pengukuran tidak langsung 3 Phasa menggunakan alat ukur Emsyst Electricity, berikut gambar alat :



Gambar 2.1 *Emsyst Electricity*

2.7. Perancangan

Menurut Verdi yasin dalam buku rekayasa perangkat lunak berbasis objek, perancangan didefinisikan sebagai proses untuk mendefinisikan suatu model atau rancangan perangkat lunak dengan menggunakan teknik dan prinsip

tertentu sedemikian sehingga model atau rancangan tersebut dapat diwujudkan menjadi perangkat lunak [18].

2.8. *Android*

Menurut Andry menjelaskan bahwa *Android* adalah sebuah sistem operasi untuk perangkat *mobile* yang menyertakan *middleware* (*virtual machine*) dan sejumlah aplikasi utama. *Android* merupakan modifikasi dari *kernel Linux* pada awalnya sistem operasi ini dikembangkan oleh sebuah perusahaan bernama *Android Inc.* dari sinilah awal mula nama *Android* muncul. *Android Inc.* adalah sebuah perusahaan *start-up* kecil yang berlokasi di Palo Alto, California, Amerika Serikat yang didirikan oleh Andrew E. Rubin bersama Rich Miner, Nick Sears, dan Chris White. Pada bulan juli 2005, perusahaan tersebut diakuisisi oleh *Google* dan para pendirinya bergabung ke *Google*. Andrew E. Rubin sendiri kemudian diangkat menjadi Wakil Presiden divisi *Mobile* dari *Google*.

Tujuan pembuatan sistem operasi ini adalah untuk menyediakan *platform* yang terbuka, yang memudahkan orang mengakses internet menggunakan telepon seluler. *Android* juga dirancang untuk pengembangan menjadi lebih berkembang [19].

2.9. *Database*

Database adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis, sehingga dapat digunakan oleh suatu program komputer untuk memperoleh informasi dari basis data tersebut. Basis data terdiri dua kata yaitu Basis berarti markas atau gudang, tempat bersarang/berkumpul. Sedangkan Data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia, barang, hewan, peristiwa, konsep, keadaan dan sebagainya.

Tujuan perancangan basis data atau *database* adalah untuk memenuhi kebutuhan akan informasi dari pengguna dan aplikasi yang digunakan. *Database* adalah salah satu koleksi terorganisasi dari data terstruktur, yang disimpan dengan duplikasi item data yang minimum guna memberikan *pool* (kelompok) data yang konsisten dan terkontrol. Data ini umum bagi semua sistem, namun independen terhadap program yang menggunakan data itu. *Database* disimpan didalam tabel, dan tabel mengandung data yang berhubungan atau *entity*, seperti misalnya orang, produk, pesanan, dan sebagainya. Tujuannya adalah menjaga tabel tetap kecil dan dapat dikelola, serta *entity-entity* yang terpisah disimpan dalam tabel-tabel tersendiri. Tentu *entity* tidak dapat satu-satu independen satu sama lain. Di dalam sebuah *database* setiap tabel memiliki sebuah *field* yang memiliki nilai unik untuk setiap baris [19].

2.10. Android Studio

Android studio adalah *Integrated Development Environment (IDE)* resmi untuk pengembangan aplikasi android. Berbasis editor kode dan alat developer yang andal dari *Intellij IDEA*, android studio menawarkan lebih banyak fitur yang mampu meningkatkan produktivitas saat membangun aplikasi android, seperti :

1. Sistem *build* berbasis Gradle yang fleksibel
2. Emulator yang cepat dan kata fitur
3. Lingkungan terpadu untuk bisa mengembangkan aplikasi untuk semua perangkat android
4. *Edit live* untuk mengupdate *composable* di emulator dan perangkat fisik secara *real time*.
5. Template kode dan integrasi Github untuk membantu membuat fitur aplikasi umum dan mengimpor kode sampel.
6. *Framework* dan alat pengujian yang lengkap

7. Alat untuk merekam performa, kegunaan, kompatibilitas versi, dan masalah lainnya.

8. Dukungan C++ dan NDK

9. Dukungan bawaan untuk *Google Cloud Platform*, yang memudahkan integrasi *Google Cloud Messaging* dan *App Engine* [20].



Gambar 2.2 *Android Studio*

2.11. *Visual Studio Code*

Visual Studio Code adalah aplikasi *editor* teks yang dikembangkan oleh *Microsoft* yang dapat digunakan disemua bahasa pemrograman yang ada tanpa perlu berganti aplikasi *editor*, serta dapat dijalankan diberbagai *platform Operation System (OS)* seperti *windows*, *linux*, dan *mac OS*.

Visual Studio Code adalah *Integrated Development Enviroment (IDE)* yang dapat melakukan *debugging*, *refactoring*, *compiling* yang biasanya digunakan untuk pengembangan aplikasi *NET*, *desktop*, *cloud*, maupun *mobile*. *Visual Studio Code* hanya dapat digunakan pada bahasa pemrograman yang sudah terdapat *compiler* yang telah disediakan secara terpisah, dan *Visual Studio Code* tidak mendukung *cross platform*. Jadi *Visual Studio Code* hanya bisa berjalan di *windows* dan *mac OS* saja [21].



Gambar 2.3 *Visual Studio Code*

2.12. *Flutter*

Flutter adalah *software development kit* (SDK) buatan *Google* yang berfungsi untuk membuat aplikasi *mobile* menggunakan bahasa pemrograman *Dart*, baik untuk *Android* maupun *iOS*. Dengan *Flutter*, Aplikasi *Android* dan *iOS* dapat dibuat menggunakan basis kode dan bahasa pemrograman yang sama yaitu *Dart*, bahasa pemrograman yang juga diproduksi oleh *Google* tahun 2011.

Sebelumnya, aplikasi murni (*native*) untuk *Android* perlu dibuat menggunakan bahasa pemrograman *Java* atau *Kotlin*, sedangkan aplikasi *iOS* perlu dibuat menggunakan bahasa pemrograman *Objective-C* atau *Swift*. *Flutter* ditujukan untuk mempermudah dan mempercepat proses pengembangan aplikasi *mobile* yang dapat berjalan di atas *Android* dan *iOS*, tanpa harus mempelajari dua bahasa pemrograman secara terpisah.

Flutter dapat dikatakan sebagai produk *Google* yang masih relative baru. Rilis perdana *Flutter*, versi *Alpha* (v.0.0.6), dipublikasikan pada bulan Mei 2017. Selain itu *Flutter* juga menyertakan kerangka *reactive-functional*, mesin render 2D, *widget* siap pakai, dan *tools* untuk pengembangan [22].

2.13. *Firestore*

Firestore adalah penyedia layanan *cloud* dengan *back-end* sebagai servis yang berbasis di San Fransisco, California, *Firestore* membuat sejumlah produk untuk pengembang aplikasi *Mobile* ataupun *web*. *Firestore* didirikan oleh Andrew Lee dan James Tamplin pada tahun 2011 dan diluncurkan dengan *cloud database* secara *realtime* ditahun 2012.

Produk utama dari *Firestore* yakni suatu *database* yang menyediakan *API* untuk memungkinkan pengembang menyimpan dan mensinkronasi data lewat *multiple client*. Perusahaan ini diakuisisi oleh *Google* pada Oktober 2014.

Firestore adalah penyedia layanan *realtime database* dan *backend* sebagai layanan. Suatu aplikasi yang memungkinkan pengembang membuat *API* untuk mensinkronasikan untuk *client* yang berbeda-beda dan disimpan pada *cloud Firestore*.

Firestore memiliki banyak *library* yang memungkinkan untuk mengintegrasikan layanan ini dengan *Android*, *IOS*, *Javascript*, *Java*, *Objective-C* dan *NodeJS*. *Database Firestore* juga bersifat bisa diakses lewat *Rest API*.

Rest API tersebut menggunakan protocol *Server-Sent Event* dengan membuat koneksi *HTTP* untuk menerima *push notification* dari *server*. Pengembang menggunakan *Rest API* untuk *post* data yang selanjutnya *Firestore client library* yang sudah diterapkan pada aplikasi yang dibangun yang akan mengambil data secara *realtime*.

Pengembang juga dapat menggunakan *database* ini untuk mengamankan data menggunakan *server Firestore* dengan *rules* yang ada. Untuk *hosting* file *firebase* menyediakan *hosting* untuk *static file* dengan fasilitas *CND* dan *SNL* [23].

2.14. *Software Development Life Cycle (SDLC)*

Pengembangan *software* atau aplikasi tentunya memerlukan berbagai tahapan dari awal pembuatan sampai *software* atau aplikasi tersebut jadi. Dalam pengembangan *software* berbagai tahapan itu disebut dengan *Software Development Life Cycle (SDLC)*. Definisi dari SDLC adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya.

Berikut beberapa model *software development life cycle*:

1. Model *Waterfall* (Air terjun)

Model *waterfall* sering disebut *the classic life cycle* atau alur hidup klasik yang melakukan pendekatan alur secara sekuensial atau terurut dari setiap tahapan pengembang. Tahapan dari model *waterfall* dimulai dari analisis kebutuhan, desain, pengodean, dan pengujian. Setiap tahapan dari model *waterfall* harus diselesaikan terlebih dahulu sebelum melanjutkan ke tahap pengembang yang berikutnya.

Model *waterfall* memiliki beberapa kelemahan dan kelebihan, namun model *waterfall* ini menjadi dasar dari model pengembangan yang lainnya. Model *waterfall* merupakan model SLDC yang paling sederhana. Model ini cocok digunakan untuk mengembangkan perangkat lunak dengan spesifikasi yang tidak berubah-ubah.

Model ini bersifat linier, prosesnya mengalir secara sekuensial dimulai dari awal hingga akhir. Pada model ini, tiap tahap harus diselesaikan dianggap tuntas agar bisa berlanjut ke tahap berikutnya. Berikut merupakan kerangka kerja model *waterfall* secara umum :

a. *Requirements Definition*

Berisi studi kebutuhan pengguna, studi kelayakan baik secara teknis maupun secara teknologi serta penjadwalan pengembangan perangkat lunak.

b. *System and Software Design*

Dalam tahap ini kita berusaha mengenali seluruh permasalahan yang dihadapi *user*, mengenali bagian-bagian sistem, objek-objek, hubungan antar objek, dan sebagainya.

c. *Implementation and Unit Testing*

Dalam tahap ini berisi proses pemilihan piranti keras, penyusunan perangkat lunak aplikasi (coding), dengan pengujian (Testing) apakah sistem sesuai dengan kebutuhan. Jika belum sesuai, dilakukan proses iterative yaitu kembali ke tahap-tahap sebelumnya.

d. *Integration and System Testing*

Tujuan *testing* adalah menemukan kesalahan-kesalahan terhadap sistem tersebut dan kemudian bisa diperbaiki. Tahapan ini bisa dikatakan final dalam pembuatan sebuah sistem.

e. *Operation and Maintenance*

Tahapan melakukan pengoperasian sistem dan melakukan perbaikan-perbaikan kecil jika diperlukan. Jika masa penggunaan sistem telah habis, maka kembali ke tahap pertama, yaitu perencanaan.

2. Model *Prototype*

Model *prototype* merupakan SDLC yang dimulai dari tahap komunikasi dengan pengguna untuk mengumpulkan kebutuhan dari pengguna tersebut, kemudian pihak pengembang membuat model purwarupa yang akan ditunjukkan kepada pengguna. Kemudian purwarupa tersebut dievaluasi oleh pengguna sampai ditemukan spesifikasi yang sesuai keinginan dari pengguna (*user*).

Model *prototype* cocok digunakan untuk membantu pengembang dalam mengetahui kebutuhan pengguna secara detail tetapi memiliki resiko besar terhadap biaya pengembangan dan waktu pengerjaannya. Model *prototype* kurang cocok digunakan untuk mengembangkan aplikasi dengan skala besar karena akan memakan waktu dan tenaga.

3. *Rapid Application Development (RAD)*

Model *Rapid Application Development (RAD)* merupakan model proses pengembangan perangkat lunak yang bersifat *incremental* terutama untuk waktu pengerjaan yang pendek. Model RAD membagi tim menjadi beberapa komponen sehingga pengerjaan perangkat lunak dapat dilakukan secara paralel.

Model RAD cocok diterapkan apabila memenuhi beberapa kriteria sebagai berikut :

- a. Anggota tim sudah memiliki pengalaman dalam mengembangkan perangkat lunak sejenis.
- b. Pengembang sudah memiliki komponen-komponen sistem yang bisa digunakan kembali dalam proyek tersebut.

4. Model *Iteratif*

Model *Iteratif* menggabungkan antara model *waterfall* dan *prototype*. Model iteratif dibuat untuk mengatasi kelemahan dalam model *waterfall* yang tidak mengakomodasi iterasi dan model *prototype* yang tidak selalu menghasilkan produk dalam proses iteratifnya. Model ini sangat cocok digunakan pengembang dengan pergantian staff yang tinggi.

5. Model *Spiral*

Model *Spiral* merupakan model pengembangan yang memsangkan *iterative* pada model *prototype* dengan control dan sistematik dari model *waterfall*. Model *Spiral* menyediakan pengembangan dengan cara cepat dan pada setiap versi perangkat lunak memiliki tambahan fungsi.

Model *spiral* cocok digunakan untuk mengembangkan aplikasi skala besar tetapi dengan target waktu dan biaya yang tidak mengikat. Dari beberapa SDLC model *Spiral* adalah model yang dapat memberikan jaminan kualitas yang paling baik untuk aplikasi skala besar [24].

Berdasarkan beberapa penjelasan tentang model *Software Development Life Cycle* diatas, model yang cocok digunakan untuk penelitian ini adalah model *waterfall* dikarenakan model tersebut paling sederhana, cocok

digunakan untuk mengembangkan aplikasi yang spesifikasinya tidak berubah-ubah dan keterbatasan waktu serta biaya yang dimiliki oleh peneliti.

2.15. *Unified Modeling Language (UML)*

Tahapan analisa kebutuhan sistem merupakan tahapan dalam menentukan spesifikasi kebutuhan sistem dan objektif yang ingin dicapai. Kualitas dari perangkat lunak yang akan dihasilkan bergantung pada proses pengidentifikasian tahap ini.

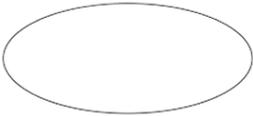
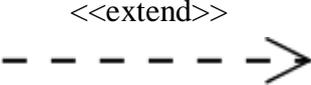
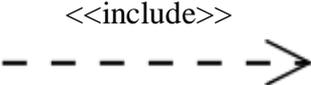
Pola seperti ini dari realita yang sederhana dan dituangkan kedalam bentuk pemetaan dengan aturan tertentu. UML (*Unified Modeling Language*) merupakan suatu alat bantu untuk pemodelan, pengertian dari UML itu sendiri merupakan bahasa visual (dapat dilihat dengan indra penglihatan mata) dalam pola dan komunikasi dari sebuah sistem dalam bentuk diagram dan naskah-naskah pendukung [25].

Berikut UML dapat dilihat pada penjelasan berikut :

1. *Use Case Diagram*

Use case diagram mendeskripsikan apa yang sistem dapat dikerjakan melalui pemodelan *tools* sistem yang terlatih dan berguna bagi actor. *Use case* tidak menerangkan cara alur kerja sistem atau bagaimana pengimplementasian sistem. Kegunaan *use case* yaitu untuk menjelaskan sistem, wilayahnya, dan sangat paut antara *sistem* dan wilayahnya. Sehingga perilaku sistem dapat dijelaskan melalui *use case* [26]. Untuk lebih jelas perhatikan Tabel 2.1.

Tabel 2.1 . Use case diagram

| No | Simbol | Fungsi | Deskripsi |
|----|---|-----------------|---|
| 1 |  | Aktor | Aktor menggambarkan semua yang berinteraksi dengan <i>system</i> . Aktor bisa berupa orang, mesin, atau sistem lain. |
| 2 |  | <i>Use Case</i> | <i>Use case</i> merupakan deretan transaksi yang dilakukan oleh sistem, menghasilkan hasil yang terukur untuk actor. |
| 3 |  | Asosiasi | Mengilustrasikan relasi antara actor dan <i>use case</i> dengan menggunakan cara menyampaikan stimulant antara satu dengan lainnya. |
| 4 |  | <i>Extend</i> | Menspesifikasikan <i>use case</i> bahwa tujuan memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan |
| 5 |  | <i>Include</i> | Merinci bahwa <i>use case</i> sumber secara eksplisit (terus terang). |

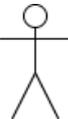
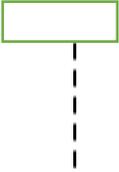
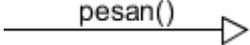
2. Sequence Diagram

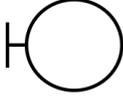
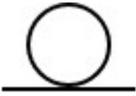
Sequence Diagram menerangkan alur eksekusi pada setiap alur yang ada pada *use case*. Sehingga memungkinkan deskripsi tekstual dari perilaku yang ada di dalam *use case* diterjemahkan menjadi operasi pada *class diagram*.

Scenario merupakan instans dari *use case*, berisi uraian kejadian yang terjadi selama proses eksekusi sistem. Scenario yang didapatkan dari setiap *use case* beragam sesuai kebutuhan. Scenario ini dimodelkan dengan *sequence diagram*. Objek dari *sequence diagram* dapat dideteksi dengan melihat kata benda yang terdapat di dalam *use case* [26].

Penjelasan *sequence diagram* dapat dilihat pada Tabel 2.2.

Tabel 2.2 . Sequence Diagram

| No | Simbol | Fungsi | Deskripsi |
|----|--|-----------------|--|
| 1 |  Actor | Aktor | Aktor merepresentasikan semua yang berinteraksi dengan sistem. Aktor bisa berupa orang, mesin, atau sistem lain. |
| 2 |  | <i>Lifeline</i> | Objek <i>entity</i> , antarmuka yang saling berinteraksi. |
| 3 |  | <i>Message</i> | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang |

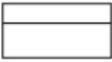
| | | | |
|---|--|-----------------|--|
| | | | aktifitas yang terjadi. |
| 4 |  | <i>Boundary</i> | Menggambarkan sebuah penggambaran dari form. |
| 5 |  | <i>Control</i> | Menggambarkan penghubung antar <i>boundary</i> dengan tabel. |
| 6 |  | <i>Entity</i> | Menggambarkan hubungan kegiatan yang akan dilakukan. |

3. Class Diagram

Class diagram mendeskripsikan susunan sistem dari segi pengertian *class-class* yang dapat dibuat untuk membangun sistem. Pada kelas terdapat atribut dan metode. Pengertian atribut adalah variable-variabel yang dimiliki oleh suatu kelas, sementara metode merupakan fungsi-fungsi atau metode yang dimiliki kelas tersebut [26].

Penjelasan *class diagram* dapat dilihat pada Tabel 2.3.

Tabel 2.3 . *Class diagram*

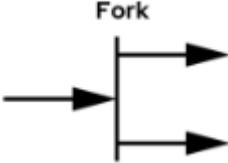
| No | Simbol | Fungsi | Keterangan |
|----|---|-------------------------|--|
| 1 |  | <i>Class</i> | Kelompok dari berbagai objek-objek yang berbagi atribut serta operasi yang sama. |
| 2 |  | <i>Generalisasi</i> | Hubungan antara objek anak (<i>descendent</i>) berbagi perilaku dan susunan data pada objek yang ada di atasnya objek induk (<i>ancestor</i>). |
| 3 |  | <i>Nary Association</i> | Usaha dalam menghindari asosiasi dengan lebih dari 2 objek |

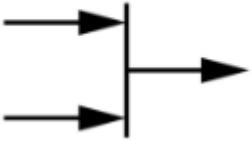
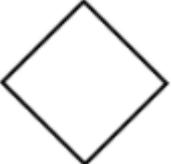
4. *Activity Diagram*

Activity diagram yaitu sebuah diagram yang dimanfaatkan untuk menggambarkan alur kerja pada *use case* proses, logika, proses bisnis dan hubungan antara actor dengan alur-alur kerja *use case* [26].

Adapun penjelasan mengenai simbol dan kegunaan *activity diagram* dijelaskan pada Tabel 2.4.

Tabel 2.4. *Activity diagram*

| No | Simbol | Fungsi | Keterangan |
|----|---|-------------------------|--|
| 1 |  | <i>Start State</i> | <i>Start state</i> merupakan keadaan awal dari sebuah <i>object</i> terjadinya perubahan keadaan. <i>Start state</i> digambarkan dengan sebuah lingkaran solid. |
| 2 |  | <i>End State</i> | <i>End state</i> adalah gambaran ketika objek berhenti memberi respon terhadap sebuah <i>event</i> . <i>End state</i> digambarkan dengan lingkaran solid di dalam sebuah lingkaran kosong. |
| 3 |  | <i>State/Activities</i> | <i>State</i> atau <i>Activities</i> merepresentasikan kondisi dari sebuah entitas. |
| 4 |  | <i>Fork/Percabangan</i> | <i>Fork</i> atau percabangan adalah pemisah beberapa aliran konkuren dari sebuah aliran tunggal. |

| | | | |
|--|---|--------------------------|---|
| | <p style="text-align: center;">Merge</p>  | <i>Join/Penggabungan</i> | <i>Join</i> atau penggabungan merupakan penggabungan beberapa aliran konkuren dalam suatu aliran tunggal. |
| | <p style="text-align: center;">Decision node</p>  | <i>Decision</i> | <i>Decision</i> adalah suatu logika aliran konkuren yang mempunyai dua cabang aliran konkuren. |

2.16. *Black Box Testing*

Black Box Testing merupakan metode pengujian kualitas sistem/komponen perangkat lunak yang berfokus pada fungsionalitas perangkat lunak yang telah didefinisikan dalam spesifikasi kebutuhan. Pengujian *Black Box Testing* tidak bergantung pada *platform*, basis data, dan sistem untuk memastikan bahwa sistem berfungsi sesuai dengan persyaratan kebutuhan spesifikasi yang telah ditentukan. *Pengujian Black Box Testing* bertujuan untuk menemukan fungsi yang tidak benar, kesalahan pada antarmuka/*interface*, kesalahan pada struktur data, kesalahan performansi, kesalahan inisialisasi dan terminasi [27].

2.17. *User Acceptance Testing (UAT)*

UAT dapat juga dikatakan pengujian beta (*beta testing*), pengujian aplikasi (*application testing*), pengujian akhir pengguna akhir (*end user testing*) merupakan tingkatan pengembangan perangkat lunak ketika perangkat lunak diuji pada dunia nyata yang dimaksudkan oleh pengguna. UAT dapat dilakukan dengan *in-house testing* dengan membayar relawan atau subjek pengujian menggunakan perangkat lunak atau, biasanya mendistribusikan perangkat lunak secara luas dengan menggunakan pengujian versi yang

tersedia gratis untuk diunduh melalui *website*. Pengalaman awal pengguna akan diteruskan kembali kepada para pengembang yang membuat perubahan sebelum akhirnya melepaskan perangkat lunak komersial [26]

III. METODOLOGI PENELITIAN

3.1. Waktu dan Tempat Penelitian

Penelitian ini dimulai pada semester genap tahun ajaran 2023 pada bulan Mei 2023 hingga pada bulan Oktober 2023. Penelitian ini dilaksanakan di PT. PLN (Persero) Unit Pelaksana Pelayanan Pelanggan (UP3) Metro. Berikut ini pada tabel 3.1 merupakan alur penelitian.

Tabel 3.1. Alur Penelitian

| Kegiatan | Pelaksanaan | | | | | | |
|--|-------------|------|------|---------|-----------|---------|----------|
| | Mei | Juni | Juli | Agustus | September | Oktober | November |
| <i>Requirements Definition</i> | ■ | ■ | | | | | |
| <i>System and Software Design</i> | | | ■ | | | | |
| <i>Implementation and Unit Testing</i> | | | | ■ | ■ | | |
| <i>Integration and System Testing</i> | | | | | | ■ | |
| <i>Operation and Maintenance</i> | | | | | | | ■ |

3.2. Perangkat Penelitian

Adapun perangkat penelitian yang digunakan dalam penelitian ini yaitu :

3.2.1. Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan dalam penelitian ini adalah laptop dengan spesifikasi sebagai berikut :

1. Laptop

- a. *System Manufacture* : *Lenovo Inc.*
- b. *System Model* : *Thinkpad T440p*
- c. *Processor* : *Intel(R) Core(TM) i5-4300M CPU @ 2.60GHz 2.59 GHz*
- d. *Installed RAM* : *8 GB*
- e. *Operating System* : *Windows 10 Pro 64-bit (10.0, Build 19044)*

2. Handphone

- a. *System Manufacture* : *Xiaomi*
- b. *System Model* : *Xiami Redmi Note 5*
- c. *Processor* : *Octa-core Max 1.80GHz*
- d. *Installed RAM* : *6 GB*

3.2.2. Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan dalam penelitian ini adalah sebagai berikut :

1. Sistem Operasi Windows 10 Pro

2. Android Studio (Version: 17.0.6 amd64 VM: OpenJDK 64-Bit Server VM by JetBrains s.r.o.)
3. Visual Studio Code (Version 1.81.1, Node.js 16.17.1)
4. Java (Version 19.0.2)
5. Flutter (Version 3.10.5)
6. Dart (Version 3.0.5)
7. DevTools (Version2.23.1)

3.3. Tahapan Penelitian

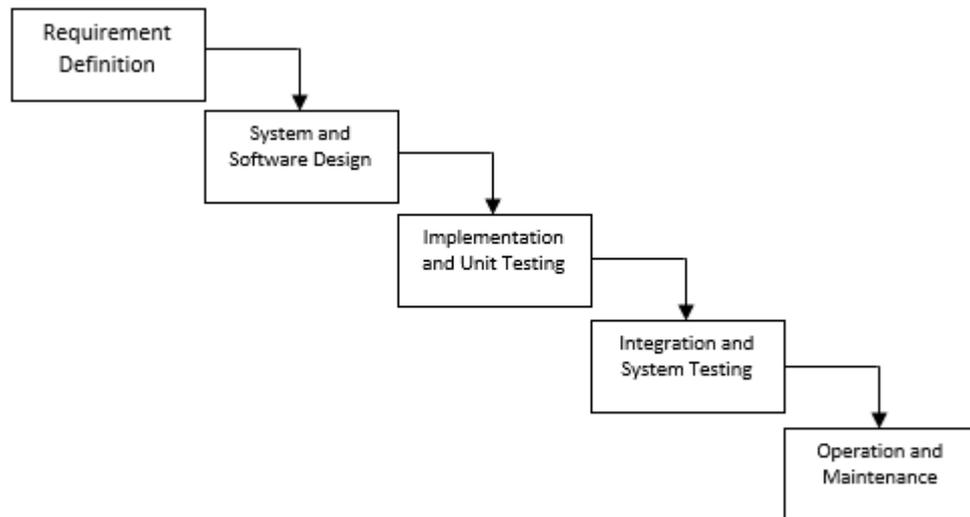
Tahapan penelitian yang dilakukan Pertama, dimulai dari pembacaan literatur-literatur yang berhubungan dengan penelitian yang dilakukan dan mencari kebutuhan pengguna. Kedua, melakukan pembuatan design sesuai dengan kebutuhan pengguna, kemudian dilakukannya perancangan dan pembuatan aplikasi pada penelitian ini. Setelah pembuatan aplikasi dilakukan pemeriksaan apakah aplikasi yang dirancang sesuai dengan yang diinginkan. Jika sudah sesuai, maka masuk dalam pembuatan aplikasi. Setelah aplikasi dibuat, maka dilakukan simulasi kegunaan aplikasi.

Jika aplikasi belum berfungsi dengan yang diinginkan, maka dilakukan pembuatan aplikasi ulang. Jika aplikasi sudah sesuai dengan yang diinginkan maka dapat berlanjut untuk pembuatan laporan. Jika laporan sudah selesai dan benar, maka penelitian ini dinyatakan berhasil dan selesai.

3.4. Metode Penelitian

Pada penelitian ini, metode perancangan aplikasi yang digunakan adalah *waterfall*. Metode *waterfall* adalah suatu proses pengembangan perangkat lunak berurutan, dimana kemajuan dipandang sebagai terus mengalir ke bawah (seperti air terjun) melewati fase-fase perencanaan, pemodelan,

implementasi (kontruksi), dan pengujian. Berikut Gambar 3.2 pengembangan perangkat lunak berurutan/linear.



Gambar 3.1. Pengembangan Perangkat Lunak Berurutan/Linear.

Dalam pengembangannya, metode waterfall memiliki beberapa tahapan yang runtut yaitu :

1. *Requirement Definition* (analisis kebutuhan)

Tahapan ini akan menghasilkan dokumen *user requirement* atau bisa dikatakan sebagai data yang terhubung dengan keinginan *user* dalam pembuatan sistem. Dokumen ini lah yang akan menjadi acuan sistem analisis untuk menerjemahkan ke dalam bahasa pemrograman.

2. *System and Software Design* (desain sistem)

Proses desain akan menerjemahkan syarat kebutuhan tersebut perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat coding. Tahapan ini akan menghasilkan dokumen yang disebut *Software Requirement*. Dokumen inilah yang akan digunakan *programmer* untuk melakukan aktivitas pembuatan sistem.

3. *Implentation and Unit Testing* (Implementasi sistem)

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut *unit*, yang terintegrasi dalam tahap selanjutnya. Setiap *unit* dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai *unit testing*.

4. *Integration and Sytem Testing* (Penerapan/Pengujian Program)

Tujuan *testing* adalah menemukan kesalahan-kesalahan terhadap sistem tersebut dan kemudian bisa diperbaiki. Tahpan ini bisa dikatakan final dalam pembuatan sebuah sistem.

5. *Operation and Maintenance* (Pemeliharaan)

Perangkat lunak yang sudah disampaikan kepada pelanggan pasti akan mengalami perubahan. Perubahan tersebut bisa karena mengalami kesalahan karena perangkat lunak harus menyesuaikan dengan lingkungan baru, atau karena pelanggan membutuhkan perkembangan fungsional.

3.5. Prosedur Pengembangan

Prosedur pengembangan perangkat lunak yang dilakukan peneliti berdasarkan *System Development Life Cycle Model Waterfall*. Prosedur tersebut terdiri dari analisis kebutuhan, desain, implementasi, pengujian dan pemeliharaan.

3.5.1. Analisis Kebutuhan (*Requirement Definition*)

Langkah pertama yang dilakukan dalam penelitian dan pengembangan ini adalah menetapkan ruang lingkup materi yang akan dikembangkan. Ruang lingkup materi yang akan dikembangkan yaitu penginputan data hasil monitoring P2TL yang meliputi sesuai dengan komponen berita acara yaitu, Id Pelanggan, Nama Pengguna, Tarif Pemakaian, Daya Pengguna, Alamat Pengguna, Tanggal Periksa, Merk Kwh, Merk CT, Keterangan Box, Arus Premier 3 Phasa (R, S, T), Arus Sekunder 3 Phasa (R, S, T), Tegangan Uji 3 Phasa (R, S, T), Error Total, Error Kwh, CTR, CTS, CTT, dan Error CT, pada bagian transaksi energi di PT.PLN

UP3 Metro. Langkah selanjutnya peneliti melakukan beberapa analisis dan kajian terkait dengan perangkat lunak yang akan dikembangkan. Proses analisis kebutuhan dilaksanakan dengan mencari dan mengkaji informasi mengenai perangkat yang akan dikembangkan baik secara langsung maupun media cetak dan elektronik.

Analisis kebutuhan perangkat lunak meliputi observasi dan kajian literatur pendukung perancangan sistem perangkat lunak.

a. Observasi

Observasi dilakukan untuk mengetahui apa saja fitur yang diharapkan ada pada perangkat lunak yang akan dikembangkan. Fitur tersebut terkait dengan pengembangan aplikasi *android* Monitoring P2TL. Observasi dilakukan terhadap beberapa karyawan dan petugas di PT. PLN Unit Pelaksana Pelayanan Pelanggan (UP3) Metro bagian transaksi energi.

Berdasarkan hasil observasi dan wawancara diperoleh beberapa fungsi dalam konsep media yang akan dikembangkan. Beberapa fungsi yang dibutuhkan antara lain :

1. Sistem mampu membuat fitur autentikasi pengguna sebagai *user*.
2. Sistem mampu menginput hasil inputan monitoring P2TL sesuai dengan komponen data berita acara yang terlampir
3. Sistem mampu mengedit data hasil inputan monitoring P2TL sesuai dengan komponen data berita acara yang terlampir.
4. Sistem mampu menampilkan data hasil inputan monitoring P2TL sesuai dengan komponen data berita acara yang terlampir.
5. Sistem mampu menghapus data hasil inputan monitoring P2TL sesuai dengan komponen data berita acara yang terlampir.

b. Studi Literatur

Studi literatur digunakan untuk mencari data mengenai konsep-konsep dan teori yang sudah ada untuk mendukung proses pengembangan perangkat lunak. Studi literature dilakukan dengan melakukan analisis terhadap penelitian-penelitian yang sudah dilakukan sebelumnya maupun terhadap aplikasi sejenis yang sudah ada.

Studi literatur yang telah dilakukan mendapatkan konsep teori mengenai beberapa hal, yaitu :

1. Aplikasi monotirong dapat membantu pekerjaan ketika dilapangan tidak lagi membawa dan menggunakan alat tulis kantor
2. *Software* yang digunakan untuk media *cloud* dan *realtime database* menggunakan *firebase*.
3. *IDE Visual Studio Code* dengan menggunakan *framework Flutter* dan memakai bahasa *Dart* dapat digunakan untuk mengembangkan aplikasi *android*.

3.5.2. Desain Sistem (*System and Software Design*)

Adapun desain dalam pengembangan aplikasi ini dibagi dalam beberapa bagian diantaranya :

a. Perancangan *Unified Modeling Language (UML)*

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program. Fokus desain pada tahap ini termasuk didalamnya dan prosedur pengkodean [28].

Desain pengembangan perangkat lunak yang dibuat nantinya akan menjadi panduan pengembang dalam proses pengkodean. Desain

sistem perangkat lunak yang dibuat harus mempermudah pengembang dalam mengembangkan perangkat lunak. Desain model yang digunakan nantinya harus sesuai dengan karakteristik perangkat lunak yang dikembangkan dan memperjelas pengembang.

Penelitian ini menggunakan desain sistem *Unified Modeling Language (UML)*. Peneliti memilih model ini dikarenakan model tersebut sesuai dengan pengembangan sistem yang dilakukan yaitu berorientasi objek. UML yang dibuat yaitu *use case diagram* dan *sequence diagram*.

1. *Use Case Diagram*

Use case merupakan pemodelan untuk memodelkan kelakuan sistem yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem. *Use case* merupakan bentuk diagram yang menggambarkan fungsionalitas yang diharapkan dari sebuah sistem dilihat dari perspektif pengguna diluar sistem.

Use case juga dapat digunakan untuk merepresentasikan interaksi yang terjadi antara aktor dengan proses sistem yang dibuat. Skenario pada perancangan *use case* merupakan langkah-langkah yang menerangkan urutan kejadian antara pengguna dan sistem.

Tahap-tahap perancangan *use case* yang dilakukan peneliti yaitu :

a). Menentukan Definisi Aktor dan *Use Case* Aplikasi Monitoring P2TL

Berdasarkan hasil dari tahap analisis kebutuhan dapat diketahui tentang aktor dan apa yang diberikan sistem kepada aktor dalam hal ini *use case*. Penjelasan tentang aktor dan *use case* ditunjukkan pada tabel 3.2 dan tabel 3.3 berikut ini.

Tabel 3.2. Definisi Aktor *user*

| Aktor | Deskripsi |
|--------------|---|
| User | User merupakan aktor penuh dari perangkat lunak Aplikasi Monitoring P2TL pada <i>android</i> yang dapat melakukan Login, Registrasi, Lupa Password, Menginput Data, Mengedit Data, Melihat Data, Mengapus Data, dan Logout. |

Tabel 3.3. Definisi *Use Case*

| Use Case | Deskripsi |
|----------------------|---|
| <i>Login</i> | Tampilan <i>login</i> adalah tampilan awal aplikasi, Agar dapat mengakses aplikasi <i>user</i> harus login terlebih dahulu menggunakan <i>email</i> dan <i>password</i> yang telah terdaftar. |
| Registrasi | Agar dapat menggunakan fitur <i>login</i> maka <i>email</i> dan <i>password</i> harus terlebih dahulu didaftarkan menggunakan fitur Registrasi |
| <i>Lupa password</i> | Pada fitur ini apabila <i>password</i> yang didaftarkan lupa maka bisa menggunakan |

| | |
|--------------------|---|
| | fitur ini yang akan terhubung dan mengirim melalui <i>email</i> . |
| <i>Home Screen</i> | Pada fitur ini <i>user</i> dapat melihat tampilan <i>card</i> data yang sudah berhasil diinput. |
| <i>Input Data</i> | Pada fitur ini <i>user</i> dapat menginput data hasil pemeriksaan. |
| <i>Edit Data</i> | Pada fitur ini <i>user</i> dapat merubah data hasil pemeriksaan apabila terjadi kesalahan. |
| Hapus Data | Pada fitur ini <i>user</i> dapat menghapus data hasil pemeriksaan yang sudah ada. |
| Tampil Data | Pada fitur ini <i>user</i> dapat menampilkan data hasil pemeriksaan yang sudah ada. |
| <i>Logout</i> | Pada fitur ini <i>user</i> dapat digunakan untuk keluar dari aplikasi dan kembali pada tampilan awal <i>login</i> . |

b). Skenario *Use Case* Aplikasi Monitoring P2TL

Berdasarkan *use case* tersebut maka dibuatlah skenario aksi aktor dan reaksi sistem. Skenario yang dibuat peneliti berdasarkan *use case* yaitu :

Nama *Use Case* : *Login*

Aktor : *User*

Deskripsi : Proses ini adalah kegiatan awal pada tampilan awal ketika ingin menjalankan aplikasi

Pre-Condition : -

Post-Condition : Menampilkan halaman utama aplikasi

Tabel 3.4. Skenario *Use Case Login*

| Aksi Aktor | Reaksi Sistem |
|--|---|
| Alur Dasar | |
| 1. <i>Login</i> dengan memasukan <i>email</i> dan <i>password</i> yang sudah terdaftar | |
| | 2. Memanggil <i>file main.dart</i> untuk menampilkan menu utama aplikasi. |
| | 3. Melakukan inisiasi hubungan ke <i>database</i> . |

Nama *Use Case* : Registrasi

Aktor : User

Deskripsi : Proses ini adalah kegiatan untuk mendaftarkan alamat *email* dan

password agar terkoneksi ke database untuk dapat mengakses *login*

Pre-Condition : *User* harus sudah berada di menu utama *login* dan mengklik belum punya akun ketika ingin registrasi

Post-Condition : Form registrasi ditampilkan untuk melakukan proses registrasi akun.

Tabel 3.5. Skenario *Use Case* Registrasi

| Aksi Aktor | Reaksi Sistem |
|------------------------------------|---|
| Alur Dasar | |
| 1. Menekan tombol Belum Punya Akun | |
| | 2. Memanggil <i>file</i> Registrasi.dart |
| | 3. Menampilkan Halaman <i>form</i> Registrasi |

Nama *Use Case* : Lupa *Password*

Aktor : *User*

Deskripsi : Proses ini adalah kegiatan apabila akun yang sudah terdaftar dan lupa *password* dan ingin melakukan *reset*.

Pre-Condition : *User* harus udah berada di menu utama *login* dan mengklik tombol lupa *password*.

Post-Condition : Mengirimkan *link* untuk mereset *password* ke *email* yang terdaftar.

Tabel 3.6. Skenario *Use Case* Lupa *Password*

| Aksi Aktor | Reaksi Sistem |
|---------------------------------|--|
| Alur Dasar | |
| 1. Menekan tombol Lupa Password | |
| | 2. Mengirimkan <i>link</i> untuk <i>reset password</i> ke <i>email</i> . |

Nama *Use Case* : *Home Screen*

Aktor : *User*

Deksripsi : Proses ini adalah kegiatan tampilan awal setelah login menampilkan menu *Home Screen*

Pre-Condition : *User* harus sudah melakukan *login* untuk bisa berada pada proses ini

Post-Condition : Data yang telah diinput ditampilkan di halaman.

Tabel 3.7. Skenario *Use Case* *Home Screen*

| Aksi Aktor | Reaksi Sistem |
|--|---------------|
| Alur Dasar | |
| 1. Memasukan <i>email</i> dan <i>password</i> yang sudah | |

| | |
|------------------------------------|--|
| terdaftar dan menekan tombol login | |
| | 2. Memanggil <i>file</i> HomeUserScreen.dart untuk menampilkan menu <i>Home Screen</i> . |

Nama *Use Case* : *Input Data*

Aktor : *User*

Deskripsi : Proses ini adalah kegiatan untuk menginput data hasil pemeriksaan

Pre-Condition : *User* harus sudah berada di tampilan menu *Home Screen*

Post-Condition : Tampilan *Form* untuk input data ditampilkan.

Tabel 3.8. Skenario *Use Case Input Data*

| Aksi Aktor | Reaksi Sitem |
|------------------------------|---|
| Alur Dasar | |
| 1. Menekan Tombol Input Data | |
| | 2. Memanggil <i>file</i> InputDataScreen.dart dan menampilkan menu <i>form input data</i> . |

| | |
|---|--|
| 3. Memasukkan seluruh data yang dibutuhkan dan menekan tombol <i>Save</i> | |
| | 4. Menyimpan seluruh data yang sudah terinput ke database dan juga menampilkan data pada menu <i>Home Screen</i> . |
| 5. Apabila ketika menekan tombol <i>save</i> terdapat <i>form</i> yang belum terisi | |
| | 6. Maka akan memunculkan dialog data <i>form</i> tidak boleh kosong |

Nama *Use Case* : *Edit Data*

Aktor : *User*

Deskripsi : Proses ini adalah kegiatan untuk mengedit data hasil pemeriksaan

Pre-Condition : *User* harus sudah berada di tampilan menu *Home Screen*

Post-Condition : Tampilan *Form* untuk *edit* data ditampilkan.

Tabel 3.9. Skenario *Use Case Edit Data*

| Aksi Aktor | Reaksi Sitem |
|--|--|
| Alur Dasar | |
| 1. Menekan Tombol Edit Data | |
| | 2. Memanggil <i>file</i> EditDataScreen.dart dan menampilkan menu <i>form edit</i> data. |
| 3. Mengganti data yang dibutuhkan dan menekan tombol <i>Save</i> | |
| | 4. Menyimpan seluruh data yang sudah terinput ke database dan juga menampilkan data pada menu <i>Home Screen</i> . |

Nama *Use Case* : Hapus Data

Aktor : *User*

Deskripsi : Proses ini adalah kegiatan untuk menghapus data hasil pemeriksaan

Pre-Condition : *User* harus sudah berada di tampilan menu *Home Screen*

Post-Condition : Data yang telah terinput akan terhapus

Tabel 3.10. Skenario *Use Case* Hapus Data

| Aksi Aktor | Reaksi Sitem |
|------------------------------|---|
| Alur Dasar | |
| 1. Menekan Tombol Hapus Data | |
| | 2. Data yang sudah terinput akan terhapus di <i>database</i> dan <i>Home Screen</i> . |

Nama *Use Case* : Tampil Data

Aktor : *User*

Deskripsi : Proses ini adalah kegiatan untuk menampilkan data hasil pemeriksaan

Pre-Condition : *User* harus sudah berada di tampilan menu *Home Screen*

Post-Condition : Data yang telah terinput akan ditampilkan

Tabel 3.11. Skenario *Use Case* Tampil Data

| Aksi Aktor | Reaksi Sitem |
|------------|--------------|
| Alur Dasar | |

| | |
|---|--|
| 1. Menekan <i>Card Data</i> pada <i>Home Screen</i> | |
| | 2. Memanggil <i>file</i> <i>TampilDataScreen.dart</i> |
| | 3.. Data yang sudah terinput di <i>database</i> akan ditampilkan |

Nama *Use Case* : *Logout*

Aktor : *User*

Deskripsi : Proses ini adalah kegiatan dimana ingin keluar dari aplikasi

Pre-Condition : *User* harus sudah berada di tampilan menu *Home Screen*

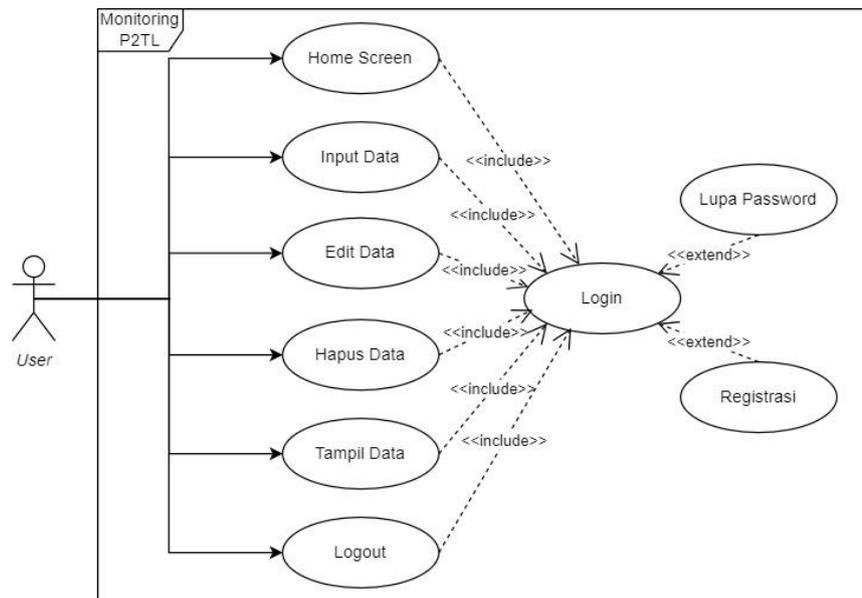
Post-Condition : Menampilkan halaman *Login*

Tabel 3.12. Skenario *Use Case Logout*

| Aksi Aktor | Reaksi Sistem |
|---------------------------------|---|
| Alur Dasar | |
| 1. Menekan Tombol <i>Logout</i> | |
| | 2. Memanggil <i>file</i> <i>main.dart</i> untuk kembali ke menu utama aplikasi. |

c). Rancangan *Use Case*

Berdasarkan hasil deskripsi actor dan *use case* serta hasil *scenario use case* maka rancangan *use case* aplikasi yang akan dikembangkan adalah sebagai berikut :

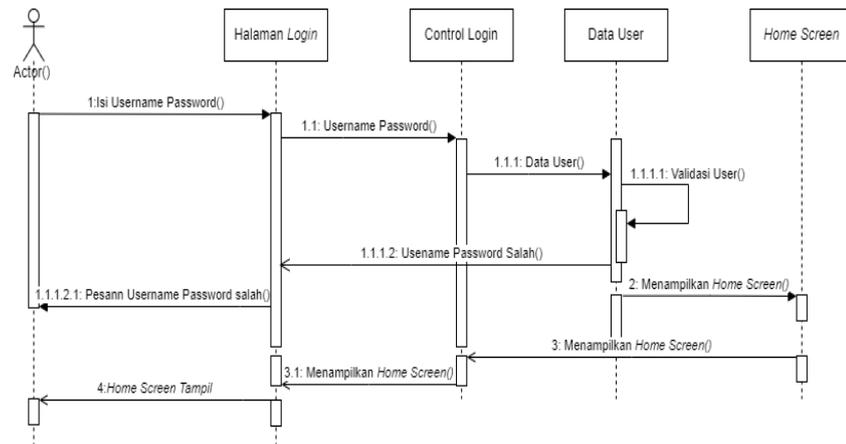


Gambar 3.2. Rancangan *Use Case Diagram*

2. *Sequence Diagram*

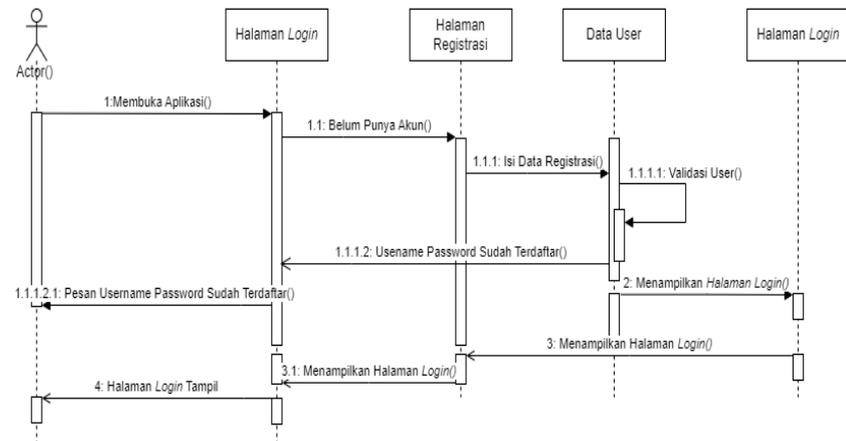
Sequence Diagram menggambarkan perilaku objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirim dan diterima antar objek. Jumlah *sequence diagram* yang dibuat sesuai dengan jumlah *use case* yang didefinisikan. *Sequence diagram* yang dibuat dalam tahap desain antara lain :

a). *Diagram Sequence : Login*



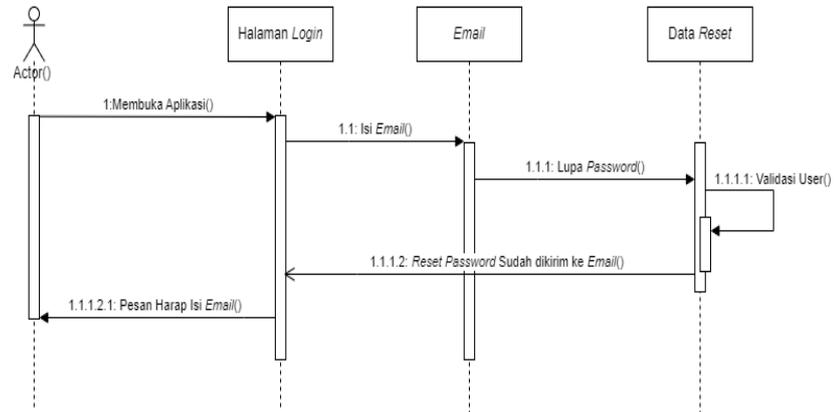
Gambar 3.3. *Sequence Diagram Login*

b). *Diagram Sequence : Registrasi*



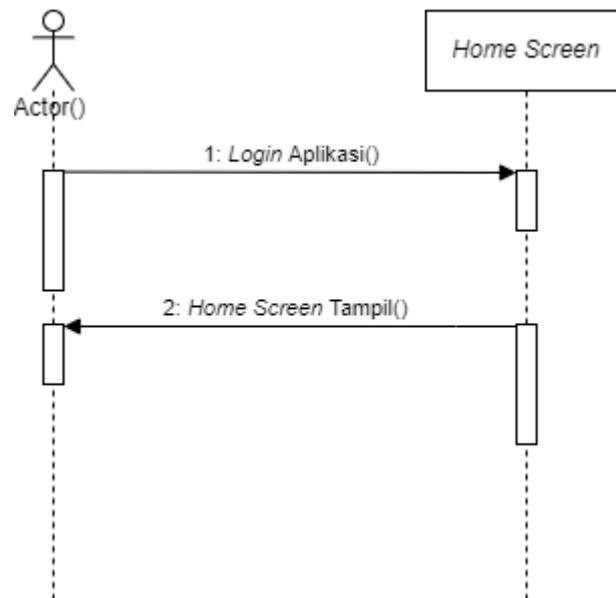
Gambar 3.4. *Sequence Diagram Registrasi*

c). *Diagram Sequence : Lupa Password*



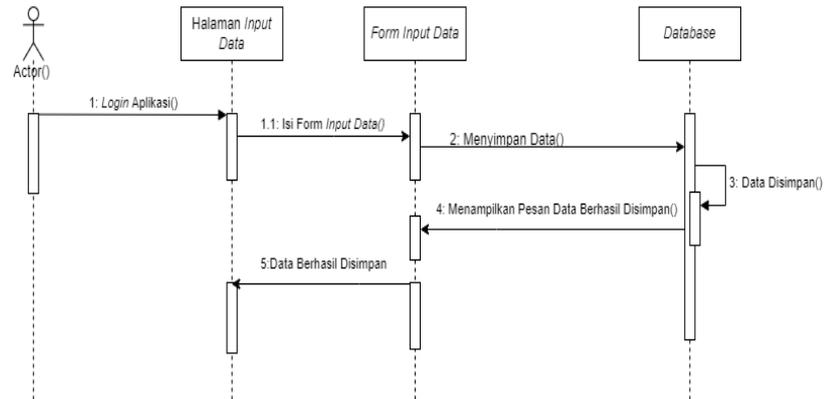
Gambar 3.5. *Sequence Diagram Lupa Password*

d). *Diagram Sequence : Home Screen*



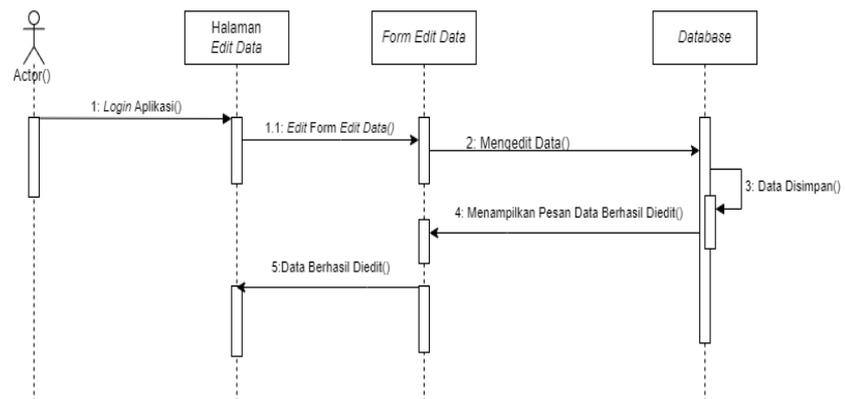
Gambar 3.6. *Sequence Diagram Home Screen*

e). *Diagram Sequence : Input Data*

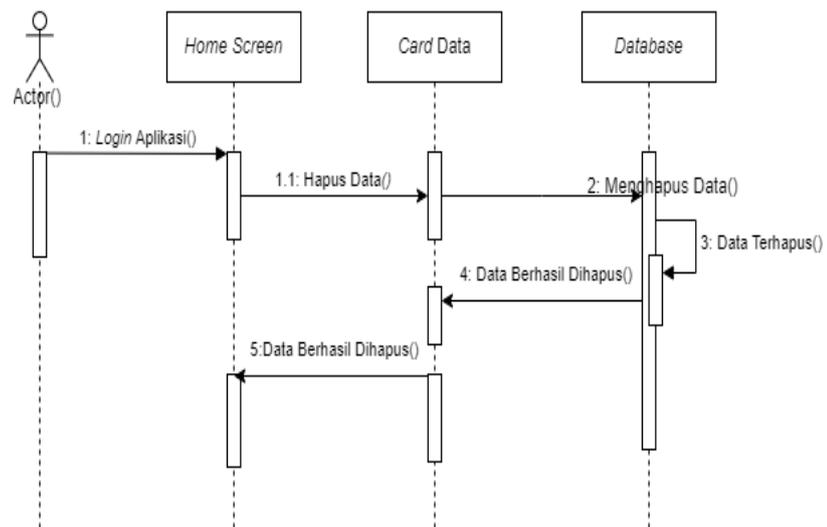
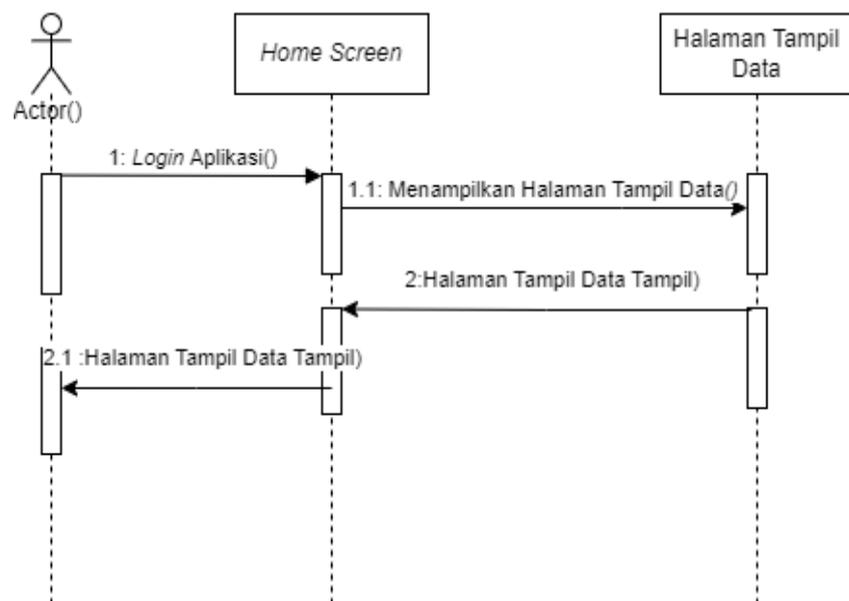


Gambar 3.7. *Sequence Diagram Input Data*

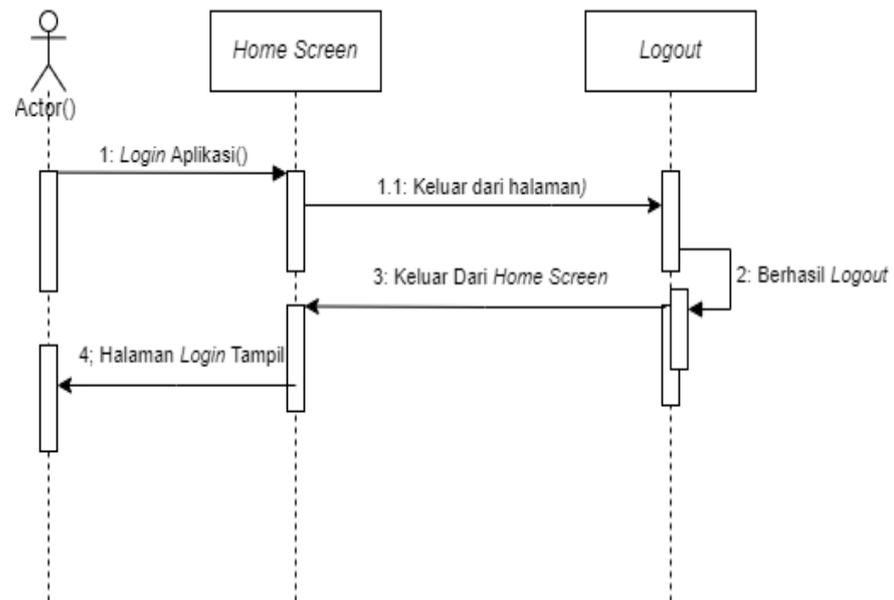
f). *Diagram Sequence : Edit Data*



Gambar 3.8. *Sequence Diagram Edit Data*

g). *Diagram Sequence* : Hapus DataGambar 3.9. *Sequence Diagram* Hapus Datah). *Diagram Sequence* : Tampil DataGambar 3.10. *Sequence Diagram* Tampil Data

i). *Diagram Sequence : Logout*

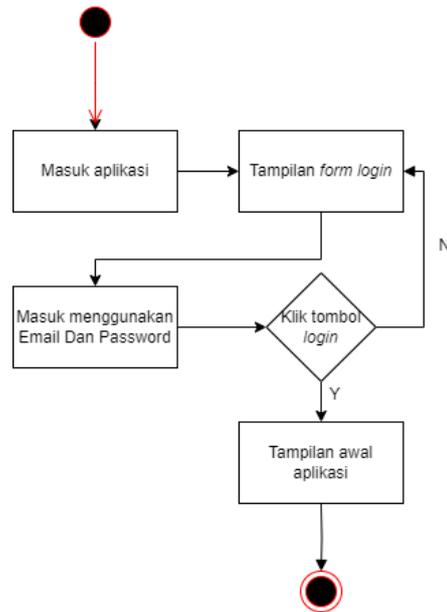
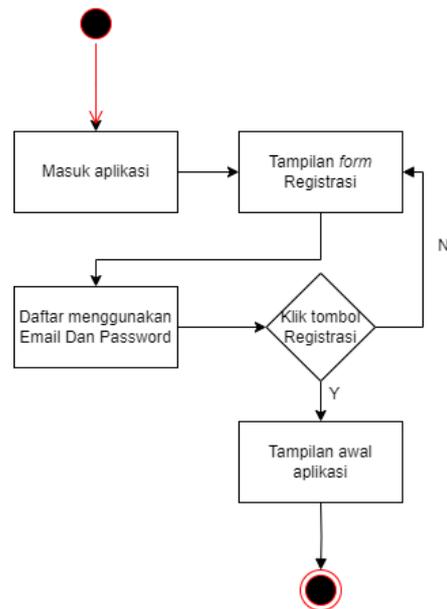


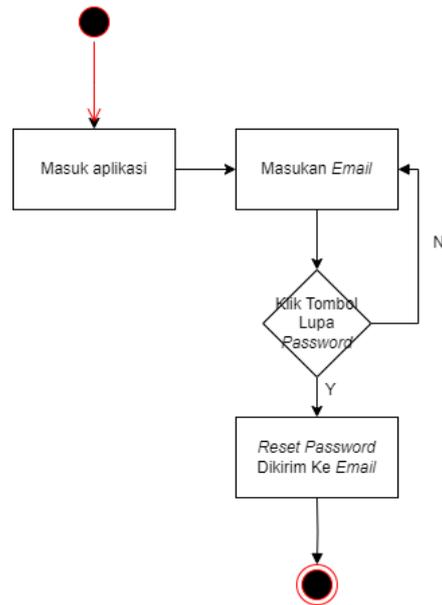
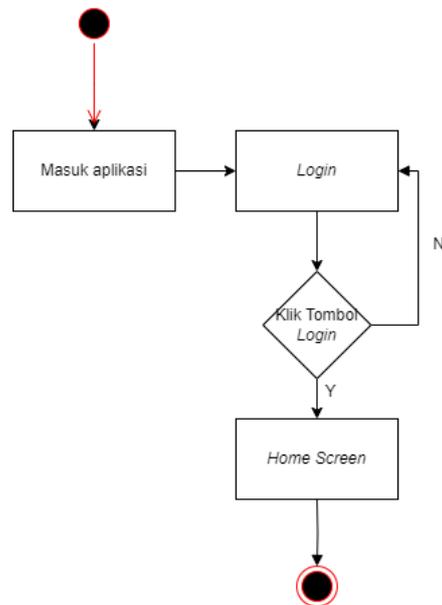
Gambar 3.11. *Sequence Diagram Logout*

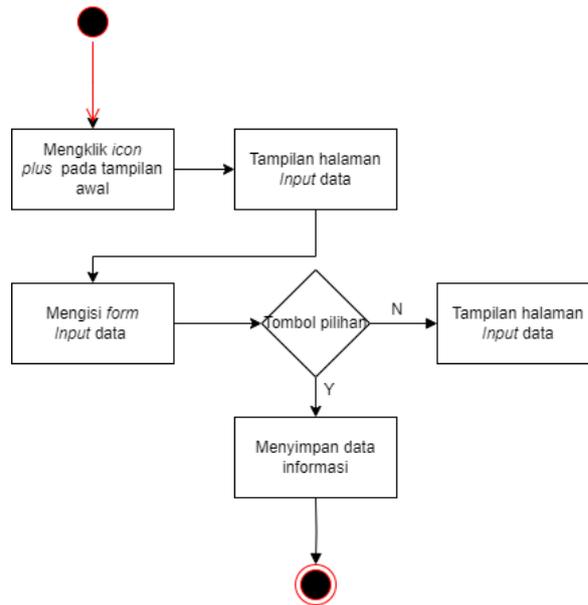
3. *Activity Diagram*

Activity Diagram adalah diagram yang berperan untuk mendeskripsikan proses bisnis sebuah sistem. Satu *Activity Diagram* difungsikan untuk menunjukkan satu alur kegiatan secara berurutan.

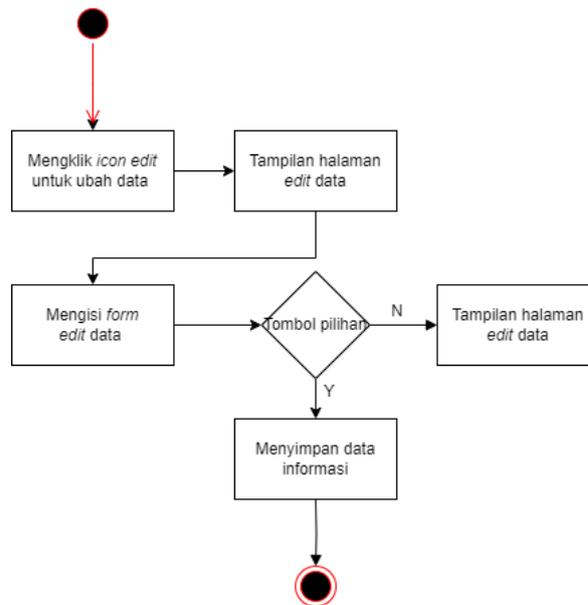
Activity diagram yang dibuat pada tahap ini adalah sebagai berikut :

a). *Activity Diagram Login***Gambar 3.12.** *Activity Diagram Login*b). *Activity Diagram Registrasi***Gambar 3.13.** *Activity Diagram Registrasi*

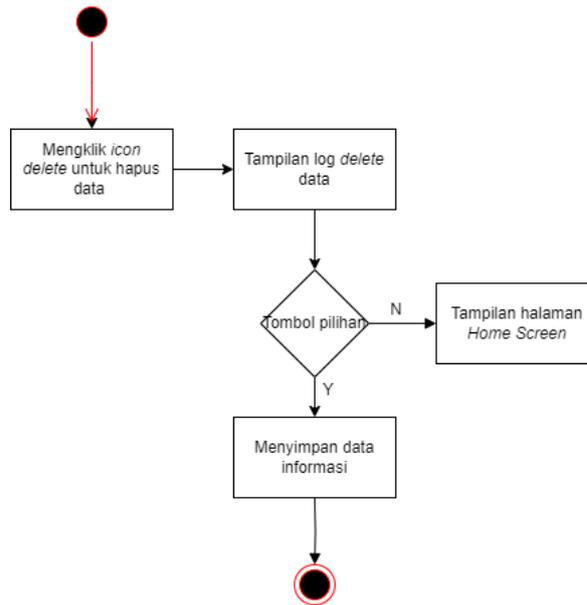
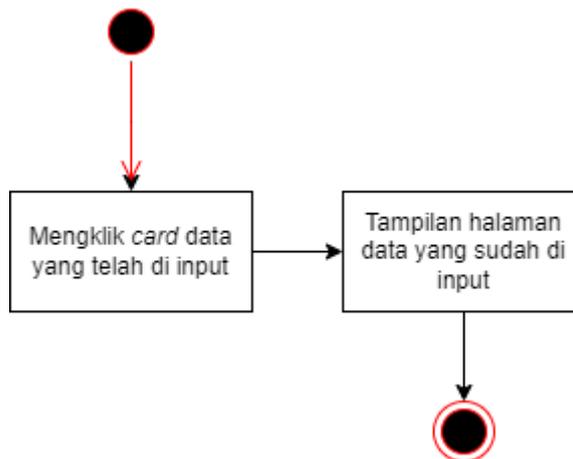
c). *Activity Diagram Lupa Password***Gambar 3.14.** *Activity Diagram Lupa Password*d). *Activity Diagram Home Screen***Gambar 3.15.** *Activity Diagram Home Screen*

e). *Activity Diagram Input Data*

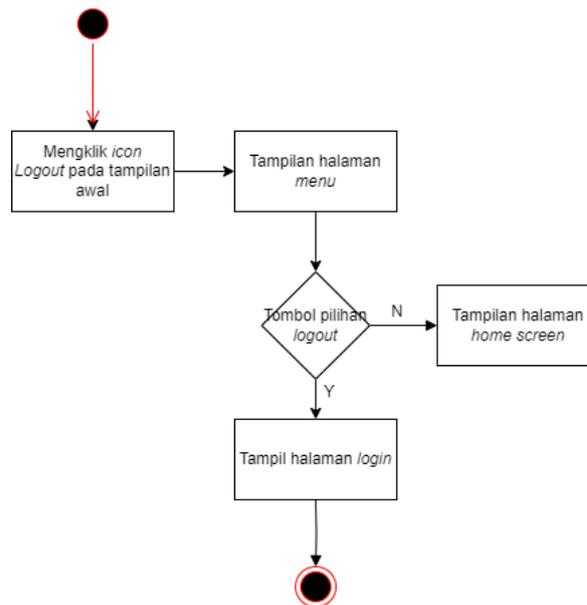
Gambar 3.16. *Activity Diagram Input Data*

f). *Activity Diagram Edit Data*

Gambar 3.17. *Activity Diagram Edit Data*

g). *Activity Diagram Hapus Data***Gambar 3.18.** *Activity Diagram Hapus Data*h). *Activity Diagram Tampil Data***Gambar 3.19.** *Activity Diagram Tampil Data*

i). *Activity Diagram Logout*



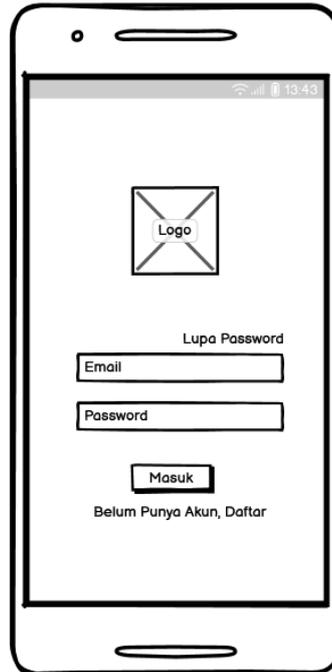
Gambar 3.20. *Activity Diagram Logout*

b. Desain User Interface (UI)

Desain *user interface* (UI) merupakan sebuah rancangan tampilan berbentuk *mockup*/desain dari aplikasi yang mencerminkan secara garis besar bentuk aplikasi yang sebenarnya. Berikut desain *user interface* (UI) yang akan digunakan untuk aplikasi ini :

a). Halaman *Login* Aplikasi

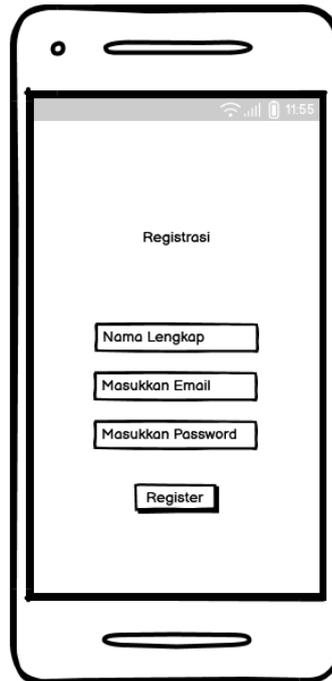
Halaman saat aplikasi berada pada halaman login dimana terdiri dari logo aplikasi, form untuk memasukkan *email*, *password*, tombol lupa *password*, tombol masuk, dan tombol untuk registrasi. Desain tampilan terdapat pada Gambar 3.22.



Gambar 3.21. Desain Tampilan Halaman *Login* Aplikasi

b). Halaman Registrasi Aplikasi

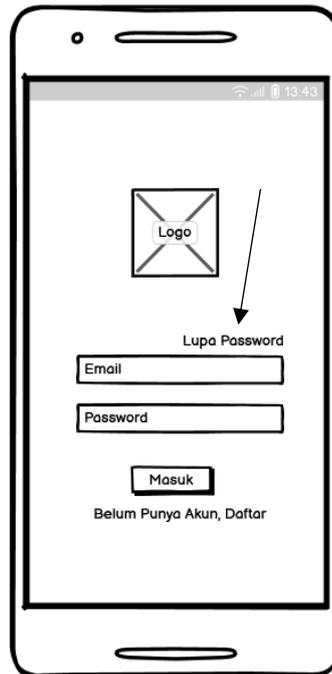
Halaman saat aplikasi berada pada halaman registrasi yang berisikan data-data diri yang harus diisi untuk mendaftar ke sistem. Halaman ini berisi form-form yang harus diisi yaitu nama lengkap, *email*, *password*. Desain tampilan terdapat pada Gambar 3.23.



Gambar 3.22. Desain Tampilan Halaman Registrasi Aplikasi

c). Halaman Lupa *Password*

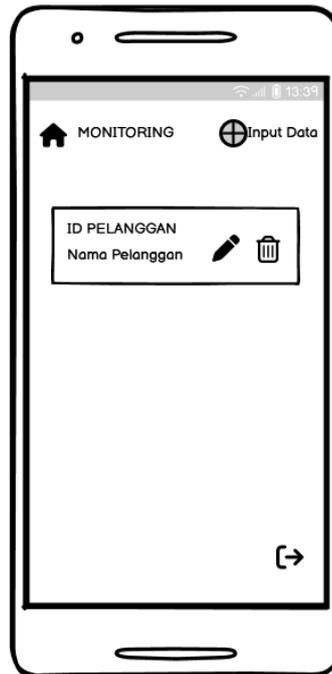
Halaman saat aplikasi berada di tampilan *login* dan pengguna lupa dengan *password*. *Reset password* akan dikirim melalui *email* yang terdaftar. Desain tampilan lupa *password* terdapat pada Gambar 3.24.



Gambar 3.23. Desain Tampilan Halaman Lupa *Password* Aplikasi

d). Tampilan *Home Screen* Aplikasi

Halaman saat aplikasi berada pada tampilan *home screen* setelah berhasil *login*, halaman ini berisikan tombol untuk masuk ke input data, *card* data yang sudah berhasil disimpan dan tombol *logout*. Tampilan halaman *home screen* terdapat pada Gambar 3.25.

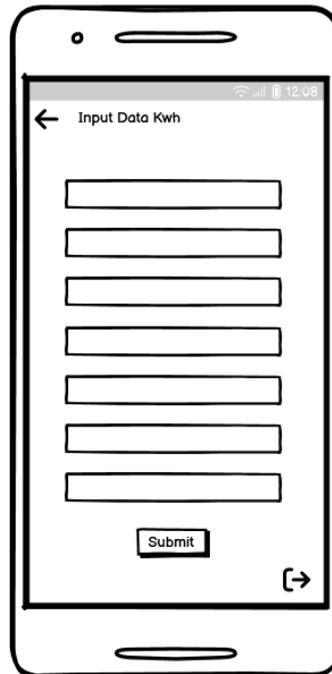


Gambar 3.24. Desain Tampilan Halaman *Home Screen*

e). Tampilan Input Data

Halaman saat aplikasi berada pada menu input data aplikasi.

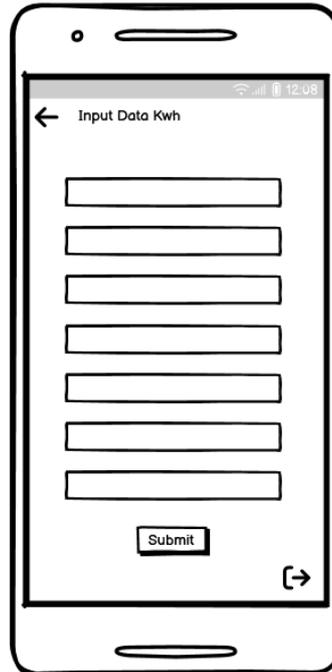
Halaman ini berisikan form-form antara lain, id pelanggan, nama pelanggan, tariff, daya, alamat, tanggal periksa, merk kwh, merk CT, rasio CT, keterangan box, ir premier, is premier, it premier, ir sekunder, is sekunder, it sekunder, vr, vs, vt, foto pashor, error total, foto error total, error kwh, foto error kwh, ctr, cts, ctt, foto error CT, dan tombol submit. Desain tampilan halaman input data terdapat pada gambar 3.26.



Gambar 3.25. Desain Tampilan Halaman Input Data

f). Tampilan Edit Data

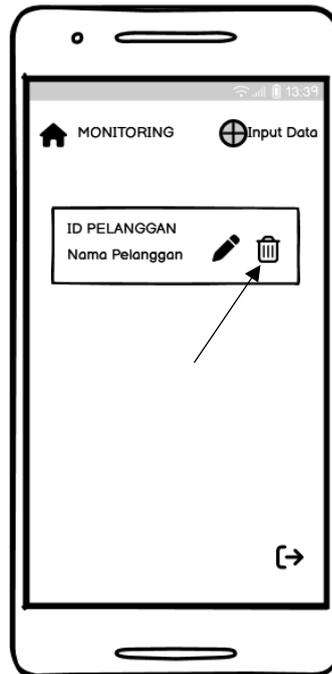
Halaman saat aplikasi berada pada menu edit data yang sebelumnya telah mengklik tombol edit pada *card* halaman awal, halaman ini berisikan sama dengan halaman input data yang sudah terisi dan akan diganti. Tampilan halaman edit data terdapat pada Gambar 3.27.



Gambar 3.26. Desain Tampilan Halaman Edit Data

g). Tampilan Hapus Data

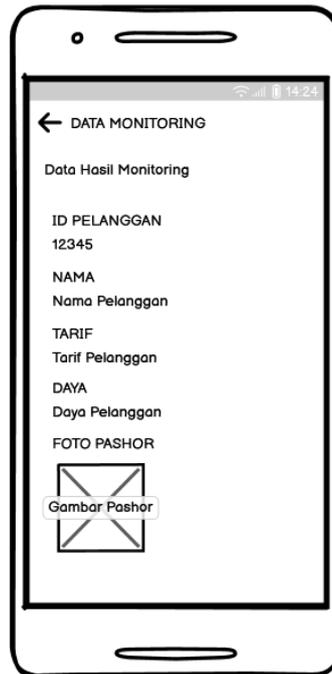
Halaman saat aplikasi berada pada menu hapus data yang berada pada halaman awal ketika setelah login. Tampilan halaman hapus data terdapat pada Gambar 3.28.



Gambar 3.27. Desain Tampilan Halaman Hapus Data

h). Tampilan Tampil Hasil Data

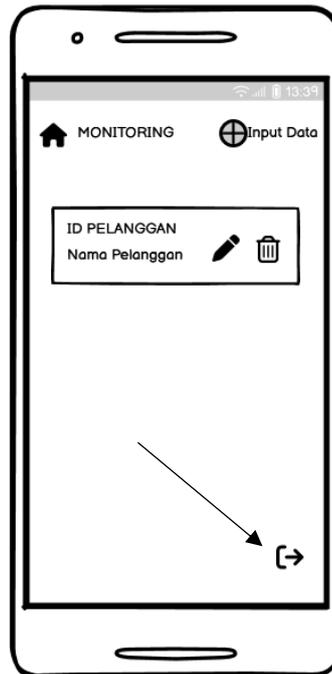
Halaman saat aplikasi sudah mengklik *card* hasil input untuk melihat hasil input data. Halaman ini berisikan data yang telah terinput. Tampilan tampil hasil data terdapat pada gambar 3.29.



Gambar 3.28. Desain Tampilan Halaman Hasil Data

i). Tampilan *Logout*

Halaman saat aplikasi berada di menu utama aplikasi. Pada halaman ini terdapat tombol *logout* untuk keluar dari aplikasi. Tampilan halaman *logout* terdapat pada gambar 3.30.



Gambar 3.29. Desain Tampilan Halaman Logout

3.6. Metode Pengujian

Pengujian adalah proses untuk menemukan *Error* pada perangkat lunak sebelum dikirim kepada pengguna. Pengujian *Software* adalah kegiatan yang ditunjukkan untuk mengevaluasi atribut atau kemampuan program dan memastikan bahwa itu memenuhi hasil yang dicari, atau suatu investigasi yang dilakukan untuk mendapatkan informasi mengenai kualitas dari produk atau layanan yang sedang diuji (*under test*). Pengujian perangkat lunak juga memberikan pandangan mengenai perangkat lunak secara objektif dan independen, yang bermanfaat dalam operasionalnya bisnis untuk memahami resiko pada implementasinya. Teknik pengujian sistem yang digunakan pada penelitian ini adalah *black box testing*.

Black box testing adalah metode pengujian perangkat lunak yang menguji fungsionalitas aplikasi yang bertentangan dengan struktur internal atau kerja. Pengetahuan khusus dari kode aplikasi/struktur internal dan pengetahuan pemrograman pada umumnya tidak diperlukan. Uji kasus dibangun disekitar

spesifikasi dan persyaratan, yakni, aplikasi apa yang seharusnya dilakukan. Mengguakan deskripsi eksternal perangkat lunak, termasuk spesifikasi, persyaratan, dan desain untuk menurunkan uji kasus. Tes ini dapat menjadi fungsional atau non-fungsional, meskipun biasanya fungsional. Perancangan uji memilih *input* yang valid dan tidak valid dan menentukan *output* yang benar. Tidak ada pengetahuan tentang struktur internal benda uji itu.

3.7. Penulisan Laporan

Tahapan dari penelitian ini yaitu penulisan laporan, tujuan dari penulisan laporan yaitu untuk memberikan dokumentasi terkait penelitian yang dilakukan, sehingga dapat dimanfaatkan oleh pembaca.

V. PENUTUP

Pada bab ini berisikan kesimpulan dari penelitian yang telah dilakukan serta saran-saran yang bermanfaat bagi pengembang aplikasi selanjutnya.

5.1 Kesimpulan

Dari pembahasan yang sudah diuraikan, dapat disimpulkan bahwa :

1. Pembuatan Aplikasi Mobile Monitoring P2TL di PT. PLN (Persero) Unit Pelaksana Pelayanan Pelanggan (UP3) Metro telah berhasil dilakukan sesuai dengan kebutuhan pengguna.
2. Aplikasi ini berhasil diujikan pada 4 orang karyawan UP3 Metro bagian distribusi pelanggan dengan 4 *smartphone* berbeda dan menghasilkan bahwa aplikasi ini tidak terdapat *error* ataupun kendala sehingga aplikasi berjalan baik dan lancar.

5.2 Saran

Sistem yang dibangun masih memiliki beberapa kekurangan, oleh karena itu ada beberapa hal yang perlu dikembangkan agar menjadi lebih baik, antara lain :

1. Aplikasi dapat dikembangkan lebih lanjut untuk versi IOS sehingga dapat dijalankan pada perangkat iPhone.
2. Aplikasi dapat dikembangkan lebih lanjut dengan menambahkan fitur untuk mendownload file hasil pekerjaan dalam bentuk PDF agar dapat mencetak hasil pekerjaan berupa data dan gambar yang lengkap.

DAFTAR PUSTAKA

- [1] F. A. Yulianto and D. P. Wigandi, "Implementasi Mobile-D Dalam Pengembangan Aplikasi Mobile Berbasis Android," *Seminar Nasional Inovasi dan Tren (SNIT)*, vol. 1, no. 1, pp. 69-77, 2018.
- [2] D. S. Purnia, A. Rifai and S. Rahmatullah, "Penerapan Metode Waterfall Dalam Perancangan Sistem Informasi Aplikasi Bantuan Sosial Berbasis Android," *Jurnal Universitas Muhammadiyah Jakarta*, vol. 10, no. 1, pp. 1-7, 2019.
- [3] I. Nurmila, Y. A. Prasetyo and P. Adytia, "Pembangunan Aplikasi Mobile Pariwisata "GoTrip" Menggunakan Metode Waterfall," *Jurnal Universitas Telkom*, pp. 1-7, 2015.
- [4] H. Widiyanto, A. P. Pratama and A. P. Laksmi, "Pengembang Aplikasi Cosycalship Berbasis Android Untuk Pengelolaan Beasiswa Menggunakan Metode Waterfall," *Journal of Advances in Information and Industrial Technology (JAIIT)*, vol. 2, no. 2, pp. 32-44, 2020.
- [5] M. Thoha and F. Masya, "Analisa Dan Perancangan Aplikasi Mobile Voteme Menggunakan Metode Waterfall," *Jurnal Ilmiah Fakultas Ilmu Komputer Universitas Mercu Buana*, vol. 8, no. 2, pp. 48-58, 2019.
- [6] F. R. Ridho and R. Soelistijadi, "Perancangan Aplikasi Pemesanan Makanan Ringan Berbasis Object Oriented Dengan Metode Waterfall," *Prosiding SENDI_U 2019*, pp. 277-284, 24 Juli 2019.
- [7] A. Triayudi and A. S. Rodhi, "Waterfall Modeling Pada Sistem E-Restorant," *Jurnal ProTekinfo Universitas Nasional*, vol. 5, pp. 17-22, 2018.
- [8] G. Y. Swara, I. Warman and D. W. T. Putra, "Implementation Of The Waterfall Model On Android-Based Travel Ticket Booking Applications,"

Journal of Information System, Informatics and Computing, vol. 6, no. 1, pp. 235-245, 2022.

- [9] M. E. Apriyanto, R. A. Yuana and M. Indriayu, "Development of Android Based Try Out Application For National Exam Of Vocational Theory In Vocational School Accompanied With Learning Outcomes Recommendation," *Indonesian Journal of Informatics Education*, vol. 3, no. 2, pp. 19-28, 2019.
- [10] E. Pawan, R. H. H. Thamrin, P. Hasan, S. H. Y. Bei and P. Matu, "Using Waterfall Method to Design Information System of SPMI STIMIK Sepuluh Nopember Jayapura," *International Journal of Computer and Information System (IJCIS)*, vol. 2, no. 2, pp. 34-39, 2021.
- [11] A. Buchori, P. Setyosari, I. W. Dasna and S. Ulfa, "Mobile Augmented Reality Media Design with Waterfall Model for Learning Geometry in College," *International Journal of Applied Engineering Research ISSN*, vol. 12, no. 13, pp. 3774-3780, 2017.
- [12] S. Al-Ratrout, O. H. Tarawneh, M. H. Al-Tarawneh and M. Y. Altarawneh, "Mobile Application Development Methodologies Adopted In Omani Market: A Comparative Study," *International Journal of Software Engineering and Applicaton (IJSEA)*, vol. 10, no. 2, pp. 13-23, 2019.
- [13] H. Ryu, M. Piao, H. Kim, W. Yang and K. H. Kim, "Development of a Mobile Application for Smart Clinical Trial Subject Data Collection and Management," *Appl SCI*, vol. 12, no. 3343, pp. 1-12, 2022.
- [14] Y. Mulyanto, F. Hamdani and H. , "Rancang Bangun Sistem Informasi Penjualan Pada Toko Omg Berbasis Web Di Kecamatan Empang Kabupaten Sumbawa," *Jurnal JINTEKS*, vol. 2, no. 1, pp. 69-77, 2020.
- [15] S. Fauzia, F. M. Agustin, U. Syaripudin and Y. Ichsani, "Perancangan Prototype Tampilan Antarmuka Pengguna Aplikasi Web Kamardagang.com

- Dengan Teknik Flat Design Pada PT. Selaras Utama Internasional," *Jurnal Teknik Informatika*, vol. 9, no. 2, pp. 148-157, 2016.
- [16] A. Hardiansyah, "Sistem Informasi Monitoring Penertiban Pemakaian Tenaga Listrik Dengan Metode PDCA (Plan, Do, Check, Action) Studi Kasus: PT. Mahera Jaya Perkasa," Digital Repository Universitas Jember, Jember, 2016.
- [17] D. Asmoro and S. , "Pengukuran Energi Listrik Tidak Langsung Menggunakan Kwh Meter Dan Kvarh Meter," *TEDC*, vol. 8, no. 3, pp. 198-204, 2014.
- [18] V. Yasin, *Rekayasa perangkat lunak berorientasi objek : pemodelan, arsitektur dan perancangan (modeling, architecture and design)*, Jakarta: Mitra Wacana Media, 2012.
- [19] A. Mardian, T. Budiman, R. Haroen and V. Yasin, "Perancangan Aplikasi Pemantauan Kinerja Karyawan Berbasis Android Di PT. Salestrade Corp.Indonesia," *Jurnal Manajemen Informatika Jayakarta*, vol. 1, no. 3, pp. 169-185, 2021.
- [20] Android Studio, "Developers," Android, 07 08 2023. [Online]. Available: <https://developer.android.com/studio/intro?hl=id>. [Accessed 28 08 2023].
- [21] Universitas Stekom, "Universitas Stekom," D3 Teknik Komputer, 04 Februari 2022. [Online]. Available: <https://teknik-komputer-d3.stekom.ac.id/informasi/baca/Mengenal-Code-Editor-Visual-Studio-Code/4bd1bb6f7ca0b022850747d950b7f73feab9ed17>. [Accessed 28 Agustus 2023].
- [22] D. Azzahra and S. Ramadhani, "Pengembangan Aplikasi Online Public Access Catalog (OPAC) Perpustakaan Bebas Mobile Pada STAI Auliaurrasyiddin," *Jurnal Intra Tech*, vol. 4, no. 2, pp. 11-25, 2020.

- [23] A. Sonita and R. F. Fardianitama, "Aplikasi E-Order Menggunakan Firebase dan Algoritme Knuth Morris Pratt Berbasis Android," *Jurnal Pseudocode*, vol. 5, no. 2, pp. 38-45, 2018.
- [24] B. Kusuma, "Pengembangan Aplikasi Aluradmi Sebagai Informasi Alur Administrasi Mahasiswa Berbasis Android Di Fakultas Teknik Universitas Negeri Yogyakarta," eprints Universitas Negeri Yogyakarta, Yogyakarta, 2017.
- [25] M. Toha, "Aplikasi Smart Living Cost Untuk Biaya Konsumsi Makanan Berbasis Android," Repository Uin Suska, Pekanbaru, 2019.
- [26] D. Iriadi, "Aplikasi Objek Wisata Berbasis Android Di Kota Pekanbaru," State Islamic University of Sultan Syarif Kasim Riau, Pekanbaru, 2019.
- [27] G. L. M, *Software Testing: Principles, techniques, and Tools*, New Delhi: Tata McGraw Hill Education Private Limited, 2009.
- [28] A. Hutomo, "Pengembangan Aplikasi Android Kamus Command Line (FYComm) Sebagai Media Bantu Belajar Siswa SMK Negeri 1 Bantul Kompetensi Keahlian Teknik Komputer Dan Jaringan," eprints UNY, Yogyakarta, 2014.