

SKRIPSI

**APLIKASI MONITORING STOK PERGUDANGAN PADA PT
MULTI KARYA SEJATI BERBASIS ANDROID**



ADELIA WIDYANINGSIH

1620221073

FAKULTAS TEKNIK

PROGRAM STUDI TEKNIK ELEKTRO

UNIVERSITAS FAJAR

2022

**Aplikasi *Monitoring* Stok Pergudangan Pada PT Multi
Karya Sejati Berbasis Android**

TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk mendapatkan gelar Sarjana
Teknik pada Program Studi Teknik Elektro Universitas Fajar**

OLEH:

ADELIA WIDYANINGSIH

1620221073



FAKULTAS TEKNIK

PROGRAM STUDI TEKNIK ELEKTRO

UNIVERSITAS FAJAR MAKASSAR

2022

HALAMAN PENGESAHAN

**Aplikasi Monitoring Stok Pergudangan Pada PT Multi Karya Sejati
Berbasis Android**

Disusun Oleh
ADELIA WIDYANINGSIH
1620221073

Telah diperiksa dan disetujui oleh Dosen
Pembimbing Makassar, 08 Februari 2023

Pembimbing I



Indah Purwitasari Ihsan, ST., MT.
NIDN.1221089001

Pembimbing II



Asmawati Azis, ST., MT.
NIDN.0905058504

Mengetahui,

Dekan Fakultas Teknik



Prof. Dr. Ir. Erniati, ST., MT.
NIDN.0906107701

Ketua Prodi Teknik Elektro



Safaruddin, S. St., MT.
NIDN.09091106901

PERNYATAAN ORISINALITAS

Yang bertanda tangan di bawah ini:

Nama : Adelia Widyaningsih

NIM : 1620221073

Program Studi : Teknik Elektro

Menyatakan dengan sebenarnya bahwa tugas akhir “ Aplikasi Monitoring Stok Pergudangan Pada PT Multi Karya Sejati Berbasis Android “ yang saya tulis ini benar-benar merupakan hasil karya saya sendiri dan setiap serta seluruh sumber acuan telah ditulis sesuai dengan panduan penulis ilmiah yang berlaku di Fakultas Teknik Universitas Fajar.

Makassar, 08 Februari 2023



Yang menyatakan

ABSTRAK

“Aplikasi Monitoring Stok Pergudangan Pada PT Multi Karya Sejati Berbasis Android (Studi Kasus pada PT Multi Karya Sejati)” Adelia Widyarningsih.

Pada perusahaan PT Multi Karya Sejati masih menerapkan sistem konvensional pada proses pencatatan barang yang dalam artian masih bersifat manual, mulai dari pengambilan barang, sistemnya masih mewajibkan karyawan atau pengguna untuk datang ke gudang untuk melakukan proses permintaan barang. Maka dibuatlah sebuah penelitian aplikasi *monitoring* stok pergudangan berbasis *android* menggunakan *AppInventor* dengan metode penelitian yang digunakan untuk perancangannya menggunakan metode *MDLC* sedangkan untuk metode pengujian yaitu menggunakan pengujian *black box* dan *white box*. Berdasarkan hasil pengujian fungsional menggunakan metode *black box* disimpulkan bahwa seluruh fungsional sistem telah berhasil berfungsi sesuai fungsinya sehingga layak untuk digunakan. Berdasarkan hasil pengujian *white box* tidak terdapat kesalahan logika, maka aplikasi dinyatakan layak karena telah bebas dari segala kesalahan logika.

Kata Kunci: Stok Pergudangan, *Android*, Aplikasi, *MDLC*

ABSTRACT

“Application for Monitoring Stock Warehousing at PT Multi Karya Sejati Based on Android (Case Study at PT Multi Karya Sejati)” Adelia Widyaningsih. *The PT Multi Karya Sejati company still applying the conventional system in the process of recording goods which in the sense that it is still manual, starting from picking up goods, the system still requires employees or users to come to the warehouse to process goods requests. So, a research on Android-based warehousing stock monitoring applications was made using AppInventor with the research method used for its design using the MDLC method while the testing method was using black box and white box testing. Based on functional testing using the black box method, the key is that the entire operating system has succeeded in accordance with its function so that it is feasible to use. Based on white box testing, there are no logical errors, then the application is declared feasible because it is free from all logic.*

Key Word: *Warehousing Stock, Android, Application, MDLC.*

KATA PENGANTAR

Assalamu Alaikum Warahmatullahi Wabarakatuh

Puji syukur kita panjatkan kehadirat Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya. Tak lupa salam dan taslim atas junjungan Nabiullah Muhammad SAW sehingga penulisan tugas akhir ini dapat diselesaikan. Tugas akhir ini ditulis sebagai salah satu syarat untuk menyelesaikan Program Studi Strata Satu Teknik Elektro di Universitas Fajar.

Penulis menyadari bahwa dalam penyusunan tugas akhir ini dapat diselesaikan karena adanya dukungan dan bantuan berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih setulus-tulusnya kepada:

1. Allah SWT yang telah memberikan kesehatan dan kemudahan serta ilmu dalam mengerjakan tugas akhir ini
2. Ayah dan ibu serta keluarga besar yang senantiasa selalu mendoakan, memberikan semangat dan dukungan dalam pengerjaan tugas akhir ini
3. Dekan Fakultas Teknik Universitas Fajar, Ibu PDr. Erniati, ST., MT
4. Ketua Prodi Teknik Elektro Universitas Fajar, Bapak Safaruddin, S.Si., MT
5. Dosen pembimbing 1 dan 2, Ibu Indah Purwitasari Ihsan, ST., MT dan Ibu Asmawaty Azis, ST., MT yang telah meluangkan waktu untuk membimbing dan memberikan petunjuk-petunjuk dalam penyusunan tugas akhir ini
6. Sekretaris Prodi Teknik Elektro, Kakanda Mimi Asmianti
7. Teman-temanku di Prodi Teknik Elektro, terkhusus kepada angkatan 2016

Tak lupa penulis ucapkan maaf sebesar-besarnya jikalau dalam penyusunan tugas akhir ini terdapat kata atau tingkah laku yang kurang berkenan dihati.

Penulis menyadari bahwa dalam penyusunan tugas akhir ini, masih banyak kekurangan dan kelemahan yang dimiliki penulis baik itu sistematika

penulisan maupun penggunaan bahasa. Untuk itu penulis mengharapkan saran dan kritik dari berbagai pihak yang bersifat membangun demi penyempurnaan tugas akhir ini. Semoga tugas akhir ini berguna bagi pembaca secara umum dan penulis secara khusus, akhir kata, penulis ucapkan terima kasih.

Makassar, 08 Februari 2023

Penulis

DAFTAR ISI

HALAMAN SAMPUL	i
HALAMAN PENGESAHAN	ii
PERNYATAAN ORISINALITAS	Error! Bookmark not defined.
ABSTRAK.....	iv
<i>ABSTRACT</i>	v
KATA PENGANTAR	vi
DAFTAR ISI.....	viii
DAFTAR GAMBAR.....	xi
DAFTAR TABEL.....	xiii
BAB I.....	1
PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian.....	2
1.4 Batasan Masalah.....	3
BAB II.....	4
TINJAUAN PUSTAKA	4
2.1. Kerangka Teori	4
2.1.1. <i>Monitoring</i>	4
2.1.2. Gudang	4
2.1.3. <i>Android</i>	5
2.1.4. PT Multi Karya Sejati	5
2.1.5. Aplikasi	5
2.1.6. <i>Multimedia Development Life Cycle (MDLC)</i>	5
2.1.7. <i>Unified Modeling Language (UML)</i>	6
2.1.8. <i>Flowchart</i>	10

2.1.9. <i>AppInventor</i>	10
2.1.10. Pengujian Sistem.....	11
2.1.11. <i>Database</i>	12
2.1.12. <i>TinyDB</i>	12
2.2. Penelitian Terdahulu (<i>State of The Art</i>).....	13
2.3. Kerangka Pikir.....	18
BAB III	19
METODE PENELITIAN.....	19
3.1 Tahapan Peneliian	19
3.2 Rancangan Penelitian	20
3.2.1. Sistem Berjalan	20
3.2.2. Sistem Yang Diusulkan.....	21
3.2.3. Perancangan Sistem	21
3.2.4. <i>Class Diagram</i>	32
3.3. Waktu dan Tempat Penelitian	34
3.4. Alat dan Bahan Penelitian	34
3.5. Metode Pengumpulan Data	35
3.5.1. Studi Pustaka.....	35
3.5.2. Studi lapangan (wawancara)	35
3.5.3. Studi Lapangan (<i>observasi</i>)	35
3.6. Metode Pengujian Sistem.....	35
BAB 4	37
HASIL DAN PEMBAHASAN.....	37
4.1. Hasil Penelitian	37
4.1.1. Tampilan Petugas Gudang	37
4.2. Pembahasan	45
4.2.1. Pengujian <i>White Box</i>	59
BAB V SIMPULAN DAN SARAN.....	75
5.1. Kesimpulan	75
5.2. Saran	75
DAFTAR PUSTAKA	76

LAMPIRAN.....	79
---------------	----

DAFTAR GAMBAR

Gambar 2. 1 Kerangka Pemikiran.....	18
Gambar 3. 1 Tahapan Penelitian	19
Gambar 3. 2 Sistem Berjalan	20
Gambar 3. 3 Sistem Yang Diusulkan.....	21
Gambar 3. 4 Use Case Diagram.....	22
Gambar 3. 5 <i>Activity Diagram</i> Registrasi	26
Gambar 3. 6 <i>Activity Diagram Login</i>	27
Gambar 3. 7 <i>Activity Diagram</i> Permintaan Barang	28
Gambar 3. 8 <i>Activity Diagram</i> Penginputan Data	29
Gambar 3. 9 <i>Sequence Diagram</i> Registrasi	30
Gambar 3. 10 <i>Sequence Diagram Login</i>	30
Gambar 3. 11 <i>Sequence Diagram</i> Permintaan Barang	31
Gambar 3. 12 <i>Sequence Diagram</i> Penginputan Data.....	32
Gambar 3. 13 <i>Class Diagram</i>	32
Gambar 4. 1 Tampilan Login.....	37
Gambar 4. 3 Tampilan Halaman Beranda.....	38
Gambar 4. 4 Tampilan Daftar Pengambilan Barang.....	41
Gambar 4. 5 Tampilan Add Barang	38
Gambar 4. 6 Tampilan List Barang.....	39
Gambar 4. 7 Tampilan List Akun	41
Gambar 4. 8 Halaman Login Pengguna	42
Gambar 4. 9 Halaman Pengambilan Barang.....	43
Gambar 4. 10 Tampilan List Barang.....	43
Gambar 4. 11 Tampilan History Pengambilan Barang	44
Gambar 4. 12 Flowgraph Daftar	59

Gambar 4. 13 Flowgraph Login.....	62
Gambar 4. 14 Flowgraph Add Barang.....	65
Gambar 4. 15 Flowgraph Add Category.....	69

DAFTAR TABEL

Tabel 2. 1 Simbol-Simbol Class Diagram	7
Tabel 2. 2 Use Case Diagram	8
Tabel 2. 3 Simbol-Simbol Sequence Diagram.....	9
Tabel 2. 4 Simbol-Simbol Activity Diagram.....	9
Tabel 3. 1 Skenario Use Case Daftar	22
Tabel 3. 2 Login	23
Tabel 3. 3 Skenario Use Case Pengambilan Barang.....	23
Tabel 3. 4 Skenario use case input data barang	24
Tabel 3. 5 Skenario use case update stok barang.....	25
Tabel 3. 6 <i>Class</i> Barang	32
Tabel 3. 7 <i>Class</i> Pengambilan Barang	33
Tabel 3. 8 <i>Class User</i>	33
Tabel 3. 9 Perangkat Keras	34
Tabel 3. 10 Perangkat Lunak	34
Tabel 4. 1 Pengujian Black Box	45
Tabel 4. 2 Rekapulasi Hasil Pengujian Blackbox.....	56
Tabel 4. 3 Pengujian Responden.....	57
Tabel 4. 4 Path Daftar	61
Tabel 4. 5 Jalur(path) Login.....	64
Tabel 4. 6 Jalur(path) Add Barang.....	67
Tabel 4. 7 Jalur(path) Pengambilan Barang.....	72
Tabel 4. 8 Tabel Hasil Pengujian White Box	73

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pada beberapa tahun belakangan ini semakin banyak perusahaan yang tumbuh dan berkembang. di Indonesia. Hal ini terlihat dengan adanya persaingan yang ketat dalam dunia usaha, baik perdagangan maupun perindustrian. Persaingan yang semakin ketat mengharuskan perusahaan untuk mengelola semua sumber daya yang dimiliki seoptimal mungkin. (Minda Mora Purba, Chaerul Rahmat)

Gudang memegang peranan yang sangat vital dalam satu siklus rantai pasok dimana gudang berperan dalam menjaga ketersediaan stok sehingga permintaan pelanggan dapat dipenuhi dalam berbagai kondisi.

Perusahaan PT Multi Karya Sejati MKS masih belum menggunakan sistem terkomputerisasi atau belum efektif dalam penggunaannya, seperti dalam *menginput* atau pencatatan ketersediaan bahan, *mengupdate* jumlah atau stok barang masuk serta stok barang keluar, yang mengakibatkan belum tercapainya efektivitas dan kemudahan dalam mengelola dengan baik dan tepat. Salah satu masalah yang sering timbul dalam sebuah persediaan barang masih secara manual mengakibatkan tidak diketahuinya jumlah dan keadaan sisa barang yang tersedia di gudang dengan pasti. Hal ini dapat mengakibatkan barang tersimpan di gudang terlalu lama, berlebihan, kekurangan, atau bahkan terjadi kehabisan barang serta kerusakan. Pencatatan secara manual dapat mengakibatkan laporan pemasukan yang kurang efisien seperti lalai untuk mencatat barang yang keluar dan semua kemungkinan lainnya dapat menyebabkan catatan persediaan berbeda dengan persediaan yang sebenarnya ada di gudang

Pada penelitian sebelumnya oleh (Aldi Lukman Al Hakim, Ilham Maulana, Itmamul Wafa, Yoga Kaswara, Yulianti, 2021) dengan judul Perancangan Aplikasi Inventaris Gudang Menggunakan Bahasa Program PHP dan

Database MYSQL Berbasis WEB. Ungkapnya sistem ini dibuat untuk memperbaiki sistem manual yang telah ada menjadi terkomputerisasi sehingga dapat memberikan informasi stok bahan yang ada sama dengan data yang dimasukkan. Penelitian lainnya oleh (Ismi Mariyah Syamsudin, 2022) dengan judul Rancang Bangun Monitring Pengendalian Intern Untuk Pendistribusian Barang di Gudang. Penelitian ini menggunakan sistem berbasis *web* untuk *memonitoring* persediaan barang di dalam gudang dan mendapatkan informasi persediaan barang terbaru.

Perbedaan penelitian yang akan saya teliti dengan kedua peneliti sebelumnya terletak pada metode pengembangan sistem. Kedua peneliti sebelumnya menggunakan metode pengembangan sistem *waterfall* sedangkan yang akan saya gunakan yaitu metode *MDLC*. Peneliti sebelumnya melakukan penelitian berbasis *web* sedangkan saya akan melakukan penelitian berbasis *android*.

Berdasarkan latar belakang di atas maka penulis melakukan penelitian berjudul “**Aplikasi Monitoring Stok Pergudangan Pada PT Multi Karya Sejati Berbasis Android**’ untuk memudahkan petugas dalam mencatat data barang, stok, dan barang yang keluar.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka dapat dirumuskan masalah sebagai berikut:

1. Bagaimana memudahkan karyawan dalam *memonitoring* stok pergudangan pada PT Multi Karya Sejati?
2. Bagaimana kelayakan fungsionalitas dari aplikasi *monitoring* stok pergudangan pada PT Multi Karya Sejati?

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah di atas maka tujuan dari penelitian ini adalah:

1. Membuat aplikasi *monitoring* sistem pergudangan pada PT Multi Karya Sejati Berbasis *Android*
2. Mengetahui kelayakan fungsionalitas dari aplikasi *monitoring* stok pergudangan pada PT Multi Karya Sejati Berbasis *Android*.

1.4 Batasan Masalah

Batasan masalah pada penelitian ini antara lain:

1. Studi kasus hanya pada PT Multi Karya Sejati
2. Aplikasi hanya dapat berjalan pada *smartphone* dengan sistem operasi *android* minimal versi 5
3. *Tools* pengembangan aplikasi yang dipakai adalah *AppInventor*
4. Fitur yang tersedia terbatas pada pencatatan data barang, penambahan stok, dan pencatatan pengambilan barang.
5. Pengguna sistem adalah petugas gudang dan karyawan yang mencatat pengambilan barang. Fitur-fitur yang dapat diakses oleh petugas gudang yaitu *login*, menu data pengambilan, menu *add* barang, menu *list* akun, dan menu *list* barang. Selain itu fitur fitur yang dapat diakses oleh karyawan yaitu tampilan *login*, menu pengambilan barang dan menu *history* pengambilan barang.

BAB II

TINJAUAN PUSTAKA

2.1. Kerangka Teori

2.1.1. *Monitoring*

Monitoring atau pengawasan adalah proses pengumpulan dan analisis informasi tentang sebuah proyek atau program yang dilakukan pada saat proyek atau program sedang berjalan (Hobson, Mayne, & Hamilton, 2014)

Monitoring didefinisikan sebagai sebuah fungsi berulang yang menggunakan pengumpulan data secara sistematis tentang indikator tertentu untuk memberikan pihak manajemen dan pemangku kepentingan utama dari sebuah proyek yang sedang berjalan dengan indikasi tingkat kemajuan dalam penggunaan dana yang dialokasikan (PSC, 2008)

Monitoring mengacu pada pengawasan kemajuan apakah *input* menghasilkan *output* yang sesuai dengan rencana. Dengan kata lain *monitoring* melihat apakah sebuah proyek tersebut sesuai dengan rancangan awal yang telah direncanakan (Sweeney & Pritchard, 2011)

2.1.2. Gudang

Gudang adalah rumah atau bangsal tempat menyimpan barang-barang, sedangkan pergudangan adalah hal simpan menyimpan barang di gudang. Jika dijabarkan lebih luas pergudangan adalah segala kegiatan yang melakukan upaya pengelolaan gudang yang merupakan sarana pendukung kegiatan penerimaan, kegiatan penyimpanan, kegiatan pemeliharaan, kegiatan *pendistribusian*, kegiatan pengendalian dan kegiatan pemusnahan, serta kegiatan pelaporan material dan peralatan agar kualitas dan kuantitas barang-barang yang terdapat dalam gudang terjamin (Andre Pratama, 2018)

2.1.3. Android

Android adalah sebuah sistem operasi untuk perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware* dan aplikasi. *Android* menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka (Ndaru Adi dan Catur Hermawan, 2016).

2.1.4. PT Multi Karya Sejati

CV. MULTI KARYA SEJATI , berdiri sejak tahun 2008 tepatnya pada tanggal 26 Juni 2008 sebagai sebuah perusahaan dengan fokus di bidang jasa kontruksi bangunan hunian, gudang , *industry* , *property* serta *interior* . Seiring perkembangannya pada tanggal 25 Juli 2019 CV. MULTI KARYA SEJATI berubah menjadi PT. MULTI KARYA SEJATI MKS.

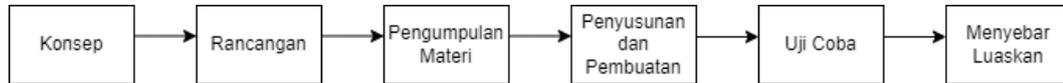
2.1.5. Aplikasi

Aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah-perintah dan 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 tehnik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputansi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan. Pengertian aplikasi secara umum adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya (Hasan Abdurahman dan Asep Ririh, 2014).

2.1.6. Multimedia Development Life Cycle (MDLC)

Prosedur penelitian dan pengembangan yang dipakai pada penelitian ini secara struktural menggunakan *Research and Development* sebagai acuan yang 2 merupakan prosedur yang dikemukakan oleh Luther-Sutopo atau yang lebih dikenal dengan metode *Multimedia Development Life Cycle*. Sebagaimana menurut Luther dalam Iwan Binanto (2010: 259), Terkait penelitian dan pengembangan yang menyangkut pendidikan, *Multimedia*

Development Life Cycle (MDLC) terdiri dari 6 tahap, yaitu konsep, rancangan, pengumpulan materi, Pembuatan, uji coba, dan distribusi.



Tahapan- tahapan dalam metode *MDLC* :

1. Konsep

Pada tahap ini menentukan tujuan dan siapa pengguna aplikasi

2. Rancangan

Tahap dimana pembuatan visualisasi yang dibuat dan dijabarkan apa yang akan dilakukan dan bagaimana tahapan rancangan menu serta data-data yang akan dibuat.

3. Pengumpulan Materi

Merupakan proses pengumpulan bahan yang sesuai dengan kebutuhan seperti kebutuhan *software* dan *hardware*.

4. Penyusunan dan Pembuatan

Proses penginputan data-data yang sudah ada kemudian diklasifikasikan sesuai dengan jenisnya

5. Uji Coba

Setelah hasil visualisasi jadi, maka dilakukan uji coba apakah hasil visualisasi yang telah dibuat sebelumnya memang tepat sebelum dapat diterapkan oleh pengguna

6. Menyebar Luskan

Tahap penggandaan dan penyebaran hasil kepada pengguna (Prasetyo and Bastian 2017; Subandi 2021)

2.1.7. Unified Modeling Language (UML)

UML (*Unified Modeling Language*) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma (berorientasi objek). Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan

permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami (Nugroho, 2010).

UML bisa juga berfungsi sebagai sebuah (*blue print*) cetak biru karena sangat lengkap dan detail. Dengan cetak biru ini maka akan bisa diketahui informasi secara detail tentang koding program atau bahkan membaca program dan menginterpretasikan kembali ke dalam bentuk diagram (*reverse engineering*)

Tujuan penggunaan UML adalah:

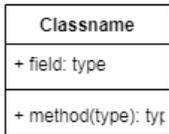
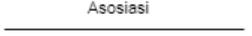
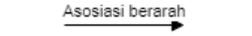
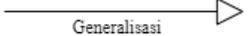
- a. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
- b. Menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan
- c. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.

Kesembilan jenis diagram dalam UML yang dapat digunakan yaitu:

a. *Class Diagram*

Diagram ini bersifat statis yang memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi.

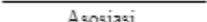
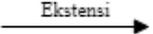
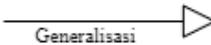
Tabel 2. 1 Simbol-Simbol *Class Diagram*

Simbol	Deskripsi
	Kelas pada struktur sistem
	Sama dengan konsep interface dalam pemrograman berorientasi objek
	kelas antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity
	kelas antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain
	kelas antar kelas dengan makna generalisasi-spesialisasi (umum-khusus)
	kelas antar kelas dengan makna kebergantungan antar kelas
	kelas antar kelas dengan makna semua bagian

b. *Use Case Diagram*

Diagram ini bersifat statis yang memperlihatkan himpunan *use case* dan aktor-aktor. Berikut ini merupakan gambar dari tiga komponen sistem dalam *use case diagram*

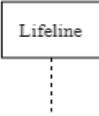
Tabel 2. 2 *Use Case Diagram*

Simbol	Deskripsi
	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i> .
	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah orang tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal <i>frase</i> nama aktor.
	Komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor
	Relasi atau use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu.
	Hubungan generealisasi dan spesialisasi (umum-khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
	relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini.

c. *Sequence Diagram*

Diagram ini bersifat dinamis. *Diagram* urutan merupakan diagram interaksi yang menekankan pada pengiriman pesan (*message*) dalam suatu waktu tertentu.

Tabel 2. 3 Simbol-Simbol *Sequence Diagram*

Simbol	Deskripsi
	Mengidentifikasi keberadaan sebuah objek dalam basis waktu. notasi untuk lifeline adalah garis putus-putus vertikal yang ditarik dari sebuah objek.
Activation 	dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah lifeline, mengidentifikasikan sebuah objek yang akan melakukan sebuah aksi
Message 	digambarkan dengan anak panah horizontal antara activation. message mengindikasikan komunikasi antara object-object
	Object merupakan instance dari sebuah class dan dituliskan tersusun secara horizontal. digambarkan sebagai sebuah class(kotak) dengan nama object didalamnya yang diawali dengan sebuah titik koma
	aktor juga dapat berkomunikasi

d. *Activity Diagram*

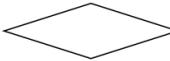
Diagram ini bersifat dinamis. *Diagram* ini adalah tipe khusus dari *diagram state* yang memperlihatkan aliran dari suatu aktifitas ke aktifitas lainnya dalam suatu sistem.

Tabel 2. 4 Simbol-Simbol *Activity Diagram*

Simbol	Deskripsi
	untuk menyatakan awal dari proses
	Untuk menyatakan akhir dari suatu proses
	Digunakan untuk menyatakan kondisi dari suatu proses
	Menyatakan aksi yang dilakukan dalam suatu arsitektur sistem

2.1.8. Flowchart

Flowchart merupakan bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. Bagan aliran adalah bagan yang menunjukkan aliran di dalam program atau prosedur sistem secara logika. Bagan aliran digunakan untuk menggambarkan prosedur sistem baik sistem berjalan ataupun sistem yang akan diusulkan (Jogiyanto, 2005).

Simbol	Fungsi
	Permulaan sub program
	Perbandingan, pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya

Gambar 2. 1 *Flowchart*
(Sumber: Jogiyanto,2005)

	Penghubung bagian-bagian flowchart yang berada pada satu halaman.
	Penghubung bagian-bagian flowchart yang berada pada halaman berbeda
	Permulaan/akhir program
	Arah aliran program
	Proses inialisasi/pemberian harga awal
	Proses penghitung/ proses pengolahan data
	Proses input/output data

Gambar 2. 2 *Flowchart*
(Sumber: Jogiyanto,2005)

2.1.9. AppInventor

App Inventor adalah sebuah aplikasi *web OpenSource* yang awal mulanya di kembangkan oleh *Google*, akan tetapi saat ini di kelola oleh *Massachusetts Institute of Technology (MIT)*, Universitas yang bergerakdi bidang teknologi

yang diakui oleh dunia. *App Inventor* ini memungkinkan pengguna baru untuk menciptakan aplikasi-aplikasi *system* Operasi *Android* .(Sari and Alvionita 2021)



Gambar 2. 3 *AppInventor*
(Sumber: Sari and Alvionita 2021)

2.1.10. Pengujian Sistem

Pengujian sistem adalah pengujian yang dilakukan pada sistem komputer (*computer-based system*) secara keseluruhan. Pengujian ini umumnya dilakukan oleh pengembang bersamaan dengan pengembang lain, karena pengujian yang dilakukan berhubungan dengan elemen lain perangkat lunak. Pengujian ini dilakukan untuk mengantisipasi masalah-masalah antarmuka dan perancangan jalur penanganan kesalahan antar sistem pada perangkat lunak (Waskitho Wibisono dan Fajar Baskoro, 2002).

2.1.10.1. *Black Box*

Black Box Testing berfokus pada spesifikasi fungsional dari perangkat lunak. *Tester* dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program. *Black Box Testing* bukanlah solusi alternatif dari *White Box Testing* tapi lebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh *White Box Testing* (Mustaqbal, 2015).

2.1.10.2. *White Box*

White Box Testing adalah salah satu cara untuk menguji suatu aplikasi atau *software* dengan cara melihat modul untuk dapat meneliti dan menganalisa kode dari program yang di buat ada yang salah atau tidak. Kalau

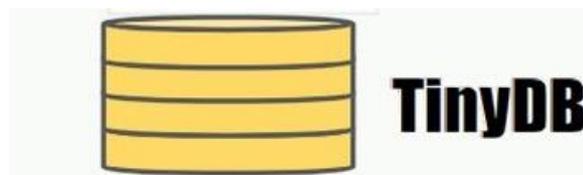
modul yang telah dan sudah di hasilkan berupa *output* yang tidak sesuai dengan yang di harapkan maka akan dikompilasi ulang dan di cek kembali kode-kode tersebut hingga sesuai dengan yang diharapkan (Nidhra and Dondetti, 2012).

2.1.11. Database

Database atau basis data adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Kegunaan utama sistem basis data adalah agar pemakai mampu menyusun suatu pandangan (*view*) abstraksi data. Hal ini bertujuan untuk menyederhanakan intraksi antara pengguna dengan sistemnya dan basis data dapat mempresentasikan pandangan yang berbeda kepada para pengguna, *programmer*, dan *administratornya* (Andaru, 2018).

2.1.12. TinyDB

TinyDB merupakan *database* untuk menyimpan data *input app inventor* agar dapat ditampilkan kembali saat *app* dibuka kembali adalah dengan memanfaatkan media *storage* seperti *TinyDB* untuk menyimpan data tersebut. Data-data yang disimpan dalam *database TinyDB* diberi *index/tag* tertentu. Untuk mengambil data dari *TinyDB* dibutuhkan *input/tag* yang sesuai. (Sari and Alvionita 2021)



Gambar 2. 4 Logo *TinyDB*
(Sumber: Sari and Alvionita 2021)

2.2. Penelitian Terdahulu (*State of The Art*)

Tabel 2. 5 State of The Art

No	Nama Penelitian	Judul Penelitian	Tahun Penelitian	Metode penelitian	Hasil Penelitian
1	Ismi Mariah Syamsudin (Syamsudin 2022)	Rancang Bangun Monitoring Pengendalian Intern Untuk Pendistribusian Barang di Gudang	2022	Metode pengembangan sistem waterfall	Mengetahui informasi mengenai persediaan barang yang tersedia di dalam gudang secara cepat dan mudah
2	Rachmat Kurniawan Putra, Ilhamsyah, Nurul Mutiah (Putra, Ilhamsyah, and Mutiah 2022)	Sistem informasi pergudangan dan pengendalian persediaan stok barang menggunakan metode economic order quantity	2022	Penelitian ini mengusulkan sebuah sistem pergudangan dan pengendalian stok barang dengan mengimplementasikan	Hasil penelitian ini berupa sebuah sistem yang dapat mengelola data barang gudang dan dapat mengontrol persediaan barang optimal

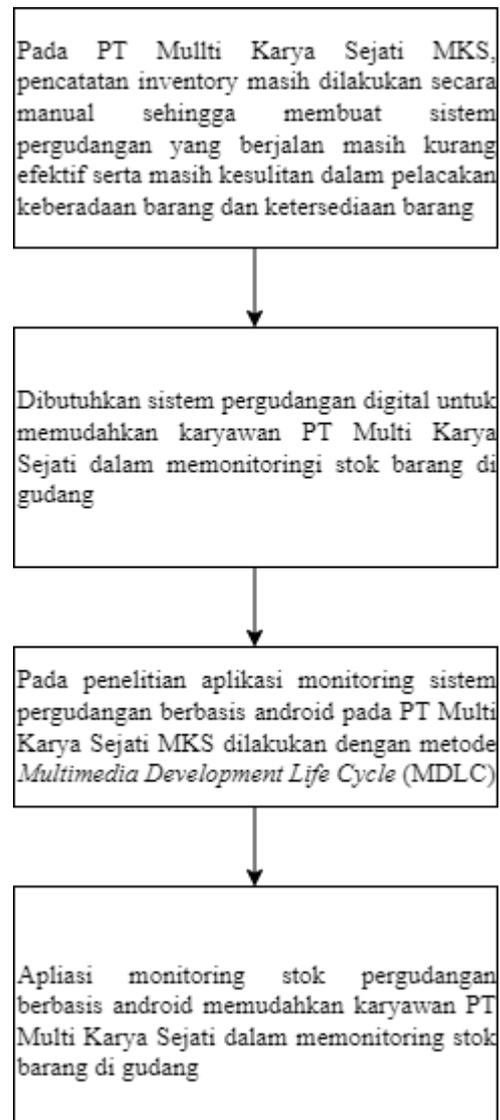
				<p>an metode Economic Order Quantity (EOQ) dimana sistem ini dapat mengontrol siklus barang masuk dan barang keluar, mengelola penjualan serta memberikan peramalan jumlah stok barang optimal beserta jumlah stok aman (safety stock) dan titik pemesanan</p>	<p>dimana hasil pengujian fungsionalitas sistem menyatakan sistem berjalan sesuai dengan yang diharapkan dengan nilai 87,76% berdasarkan skala likert.</p>
--	--	--	--	--	--

				n kembali (reorder point) untuk memenuhi ketidakpastian permintaan	
3	E Listiyan, E R Subhiyakt o(Listiyan and Subhiyakt o 2021)	Rancang Bangun Sistem Inventory Gudang Menggunakan Metode Waterfall di CV.Aqualu x Duspha Abadi Kudus Jawa tengah	2021	Metode Waterfall	Metode waterfall yang digunakan sangat berguna dalam pengembangan sistem inventory yang dibuat serta menghasilkan aplikasi yang berguna dan memiliki fungsionalitas sebesar 100%
4	Ricky Agustian,	Analisis dan	2021	Penelitian ini	Memudahkan mengelola

	Prionggo Hendradi(Agustian and Hendradi 2021)	Perancang an Sistem Informasi Monitoring Inventory Barang Pada PT.Sumber Laris Abadi Berbasis Android Dengan Metode Fast (Framewor k For The Application OF System Thingking)		mengguna kan metode Fast (Framewo rk For The Applicatio n Of System Thinking)	data dalam penarikan laporan dan mengontrol stok barang
5	Andre Pratama, Ikhstison Mekongga	Aplikasi Pencatat Barang Masuk Barang Keluar Berbasis <i>Android</i> di PT. Bhanda	2018	Metode Waterfall	Aplikasi Pencatat Barang masuk dan keluar Berbasis <i>Android</i> di PT. Bhanda Ghara Reksa Palembang

		Ghara Reksa			telah berhasil dirancang dengan memanfaatka n <i>software</i> <i>Eclipse</i> sebagai pemrograma n pembuatan aplikasi <i>Android</i> . Dengan adanya aplikasi ini membantu karyawan dalam memantau barang masuk dan barang keluar.
--	--	----------------	--	--	--

2.3.Kerangka Pikir

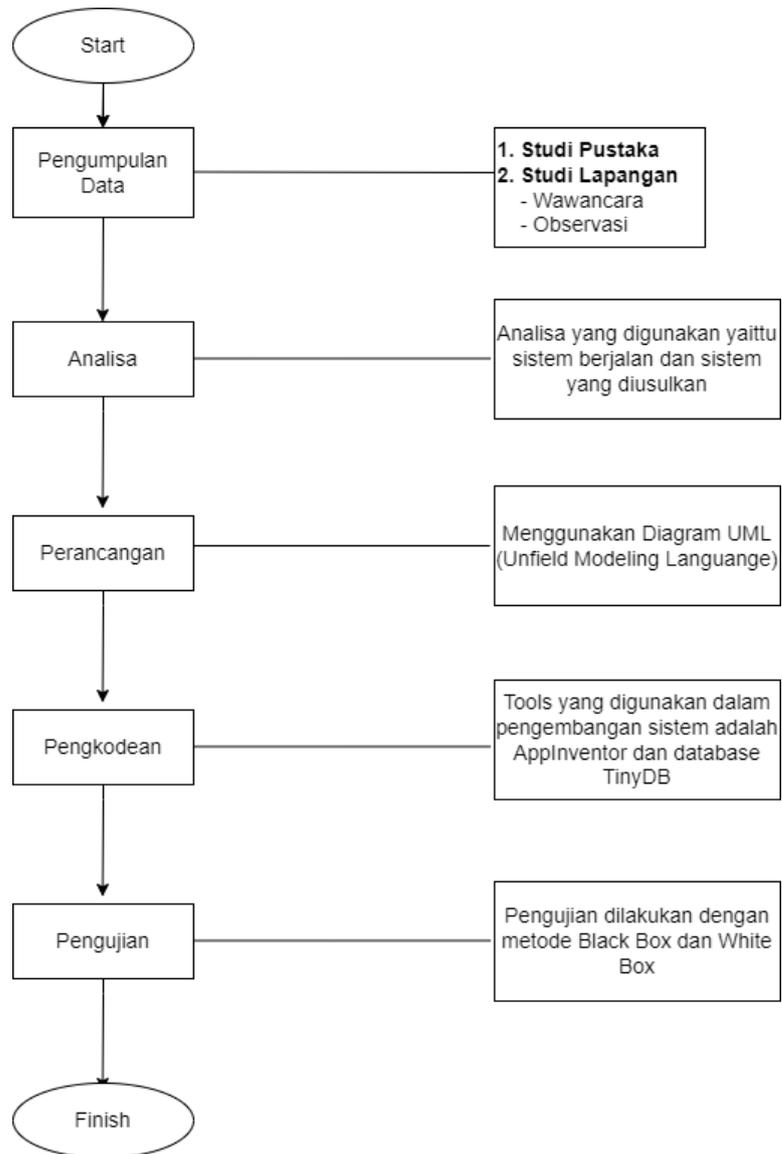


Gambar 2. 5 Kerangka Pemikiran
(Sumber: Pribadi)

BAB III

METODE PENELITIAN

3.1 Tahapan Peneliiian



Gambar 3. 1 Tahapan Penelitian

3.2 Rancangan Penelitian

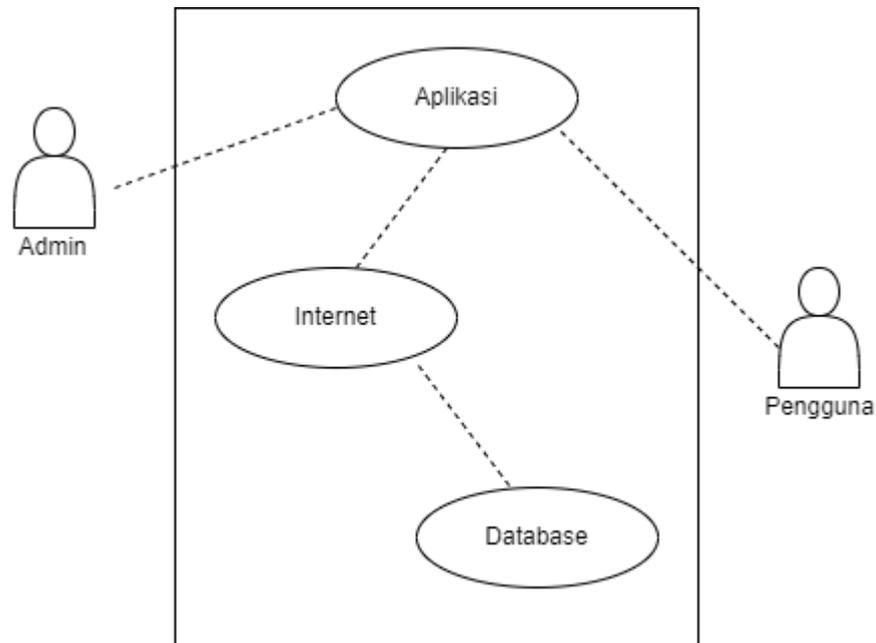
3.2.1. Sistem Berjalan



Gambar 3. 2 Sistem Berjalan

Sistem yang sedang berjalan adalah masih manual dimana pekerja harus datang ke kantor PT Multi Karya Sejati MKS untuk melampirkan *form* permintaan barang atau peralatan yang diinginkan, kemudian bagian kepala gudang atau logistik melihat data dan melacak ketersediaan barang kemudian menyerahkan barang atau peralatan tersebut ke pekerja kemudian mencatat barang atau peralatan yang keluar dari gudang.

3.2.2. Sistem Yang Diusulkan



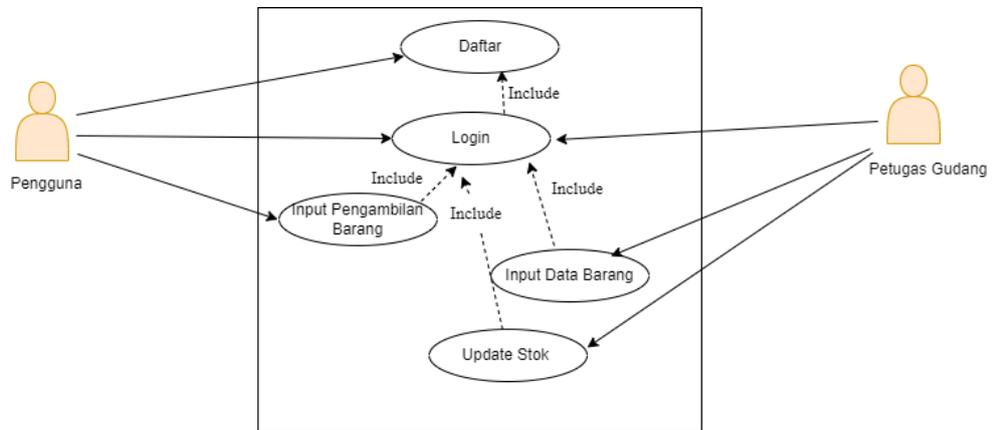
Gambar 3. 3 Sistem Yang Diusulkan

Sistem yang diusulkan yaitu menggunakan aplikasi berbasis *android* yang dimana admin dan pengguna bisa terhubung ke dalam aplikasi menggunakan internet sehingga admin tidak repot lagi mendatangi kedua gudang untuk mengecek atau mencari barang atau peralatan karena dengan aplikasi tersebut admin dan pengguna bisa mengakses kapan saja dan dimana saja. Kemudian data-data tersebut tersimpan dengan aman di *database*.

3.2.3. Perancangan Sistem

3.2.3.1. Use Case Diagram

Use case diagram adalah sebuah diagram yang saling berinteraksi antara aktor dan sistem dengan tujuan agar sistem tersebut bisa berjalan dengan baik



Gambar 3. 4 Use Case Diagram

Use case diagram pada aplikasi android ini memiliki 2 *actor* yaitu (petugas gudang dan pengguna) dimana petugas gudang dan pengguna sama-sama melakukan *login* tetapi bedanya pengguna harus melakukan pendaftaran terlebih dahulu agar bisa *login*.

A. Deskripsi skenario *use case*

1. Skenario *use case* daftar

Nama *use case*: Daftar

Aktor : Pengguna

Tujuan : Registrasi ke Aplikasi

Deskripsi : Pengguna melakukan proses pendaftaran untuk dapat mengakses sistem.

Tabel 3. 1 Skenario *Use Case* Daftar

Pengguna	Aplikasi
1. Pengguna membuka aplikasi	
	2. Aplikasi menampilkan <i>form</i> Daftar akun
3. Pengguna <i>menginputkan</i> data pendaftaran kemudian klik simpan	

	4. Proses menyimpan, kemudian menampilkan halaman <i>login</i>
--	--

2. Skenario *use case Login*

Nama *use case* : *Login*

Aktor : Admin dan Pengguna

Tujuan : *Login* ke Aplikasi

Deskripsi : Admin dan Pengguna sama-sama melakukan *login* ke dalam aplikasi dengan *username* dan *password* yang berbeda

Tabel 3. 2 *Login*

Admin dan Pengguna	Aplikasi
1.Admin dan Pengguna membuka <i>form login</i>	
	2.Aplikasi menampilkan <i>form login</i>
3.Admin dan Pengguna memasukkan <i>username</i> dan <i>password</i> yang berbeda ke dalam <i>form login</i>	
	4.Aplikasi menampilkan beranda

3. Skenario *use case* pengambilan barang

Nama *use case* : Pengambilan barang

Aktor : Pengguna

Tujuan : Mencatat pengambilan barang

Aplikasi Deskripsi : Pengguna melakukan pencatatan barang yang diambil dengan memilih barang yang dibutuhkan dan memasukkan jumlah barang yang akan diambil.

Tabel 3. 3 Skenario *Use Case* Pengambilan Barang

Pengguna	Aplikasi
Pengguna memilih menu pengambilan pada halaman beranda	
	Menampilkan <i>form</i> isian pengambilan barang
Klik Simpan	Menyimpan data, dan kembali kehalaman beranda

4. Skenario *use case* *Input* Data Barang

Nama *use case* : *Input* data barang

Aktor : Petugas gudang

Tujuan : *Penginputan* data-data barang ke aplikasi

Deskripsi : Admin melakukan *penginputan* Data barang ke dalam aplikasi

Tabel 3. 4 Skenario *use case* *input* data barang

Admin	Aplikasi
Admin menekan menu <i>add</i> barang	
	Aplikasi menampilkan <i>form</i> isian <i>add</i> barang
Admin klik tombol simpan	
	Proses menyimpan data, dan kembali ke halaman beranda

5. Skenario *use case update* stok barang

Nama <i>use case</i>	: <i>update</i> stok barang
Aktor	: Admin
Tujuan	: <i>Mengupdate</i> stok barang
Deskripsi	: Admin <i>menginput</i> jumlah stok barang

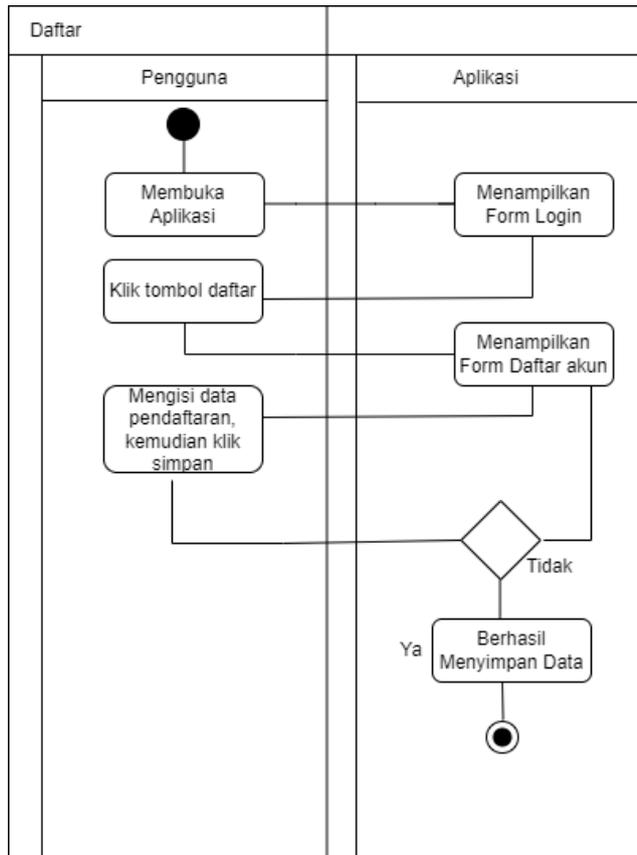
Tabel 3. 5 Skenario *use case update* stok barang

Admin	Aplikasi
Admin memilih menu <i>list</i> barang pada halaman beranda	
	Aplikasi menampilkan daftar data barang
Admin memilih data barang yang ingin ditambah stok barang	
	Tampil konfirmasi tambah stok
Pilih tombol tambah stok	
	Menampilkan <i>form</i> isian <i>add</i> stok
Mengisi jumlah stok	
Klik tombol tambah	
	Proses menyimpan, kemudian kembali ke halaman daftar data barang

3.2.3.2. Activity Diagram

Activity diagram adalah suatu proses yang menggambarkan aliran aktivitas yang terjadi dalam sistem.

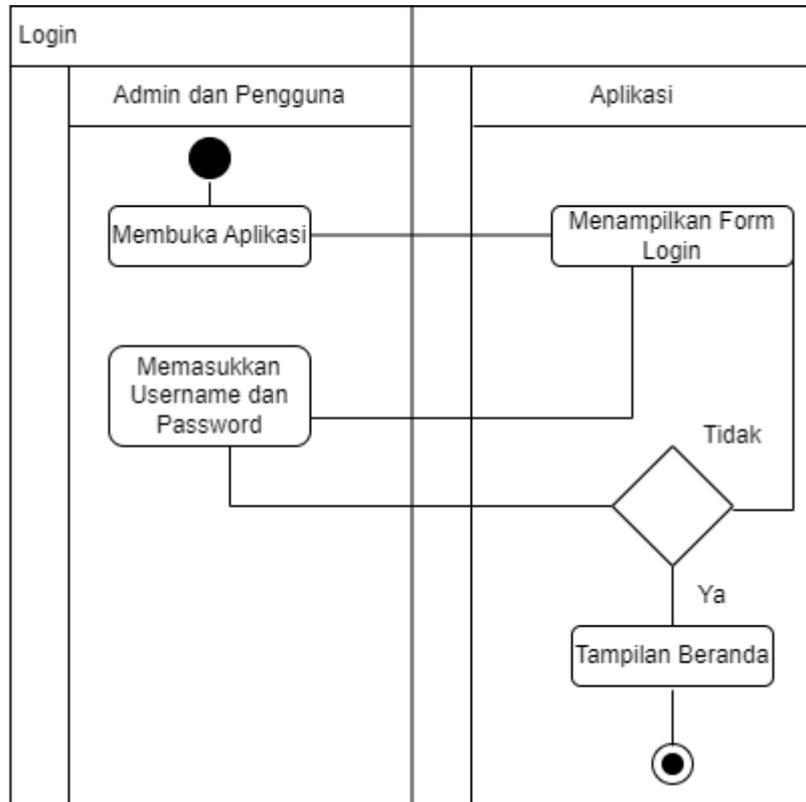
a. *Activity Diagram Registrasi*



Gambar 3. 5 *Activity Diagram* Daftar

Untuk dapat *login* ke aplikasi maka pengguna harus melakukan pendaftaran terlebih dahulu dengan *meninputkan* data akun.

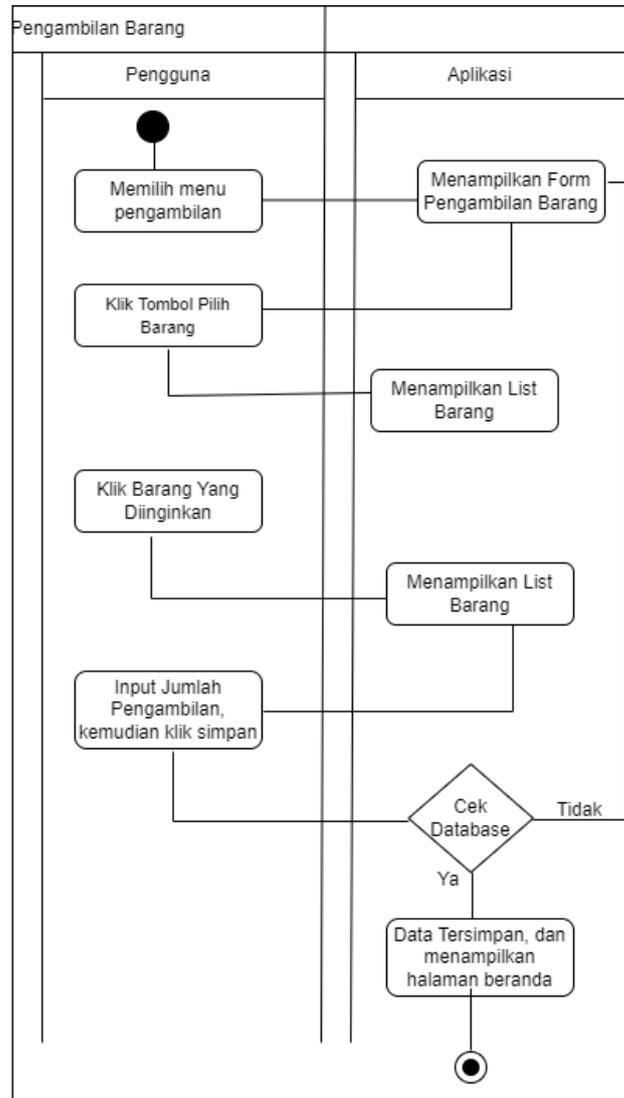
b. Activity Diagram Login



Gambar 3. 6 Activity Diagram Login

Admin dan pengguna membuka aplikasi pergudangan digital dengan melakukan *login* kemudian aplikasi menampilkan *form login*, memasukkan *username* dan *password* yang berbeda maka aplikasi mengirim perintah ke halaman beranda.

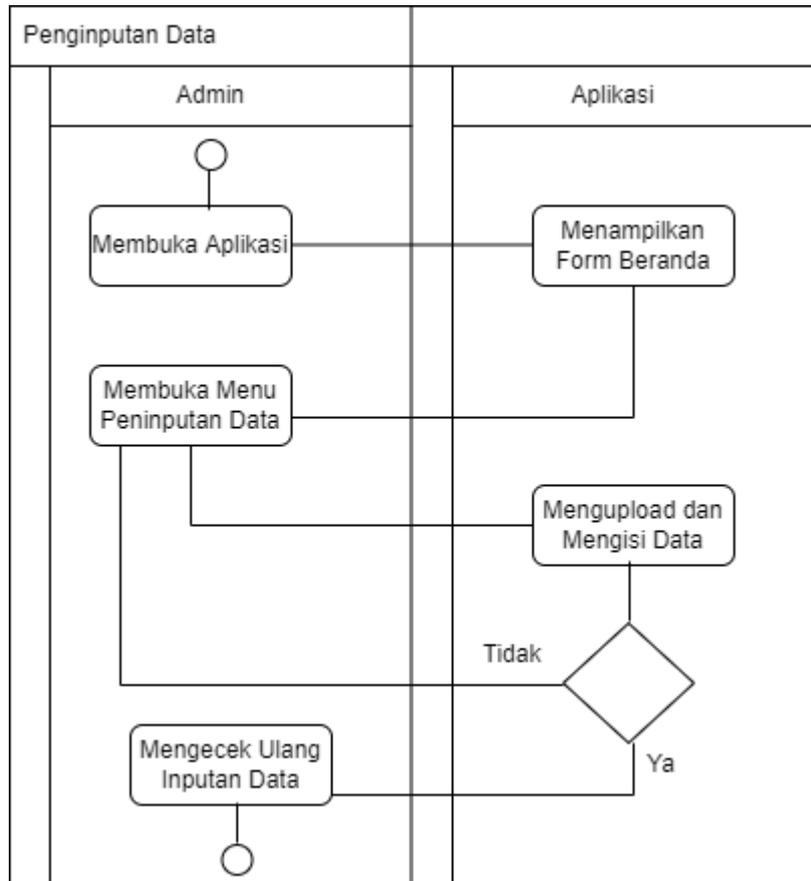
c. *Activity Diagram Input Pengambilan Barang*



Gambar 3. 7 *Activity Diagram Input Pengambilan Barang*

Pada *activity diagram input* pengambilan barang pengguna memilih barang yang ingin diminta, kemudian sistem pada aplikasi menampilkan *list* barang yang diinginkan.

d. Activity Diagram Input Data Barang



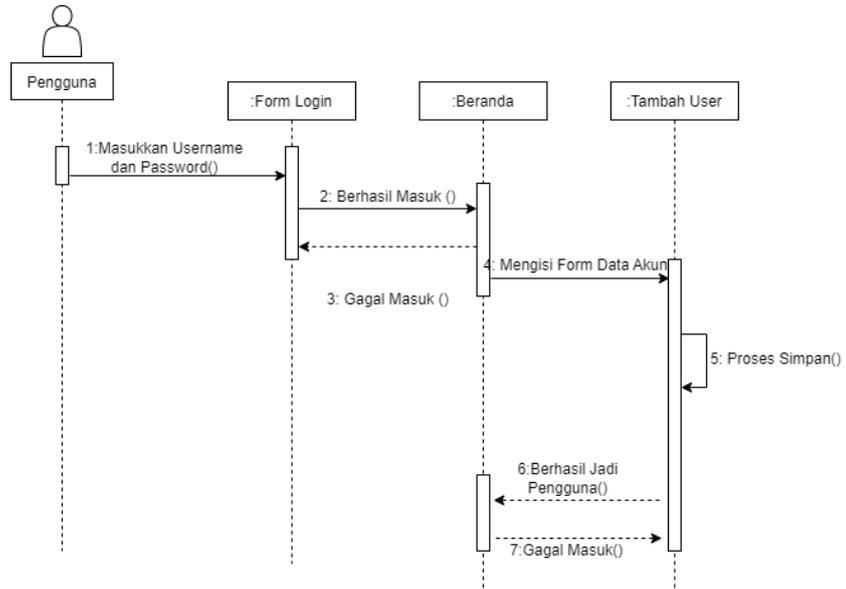
Gambar 3. 8 Activity Diagram Penginputan Data

Admin melakukan *penginputan* data dengan membuka menu *inputan* data pada aplikasi pergudangan digital kemudian sistem akan *mengupload* data dan akan menampilkan data yang *diinput* ke halaman beranda.

3.2.3.3. Sequence Diagram

Sequence diagram adalah suatu diagram dimana menggambarkan urutan interaksi antara objek yang dimana proses operasinya diurutkan dari kiri ke kanan.

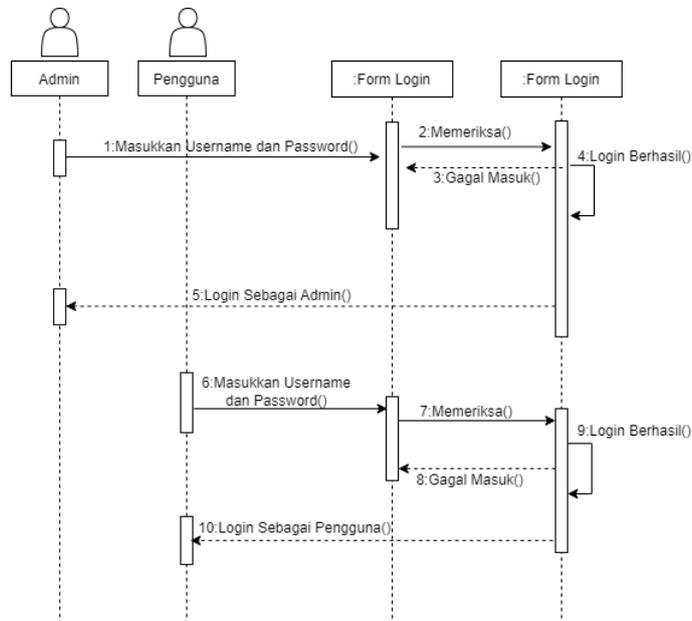
a. Sequence Diagram Daftar



Gambar 3. 9 Sequence Diagram Daftar

Admin mengisi *form* daftar untuk mendapatkan akun agar bisa *login* ke halaman beranda pergudangan digital kemudian sistem melihat registrasi jika gagal maka kembali ke halaman *form* registrasi

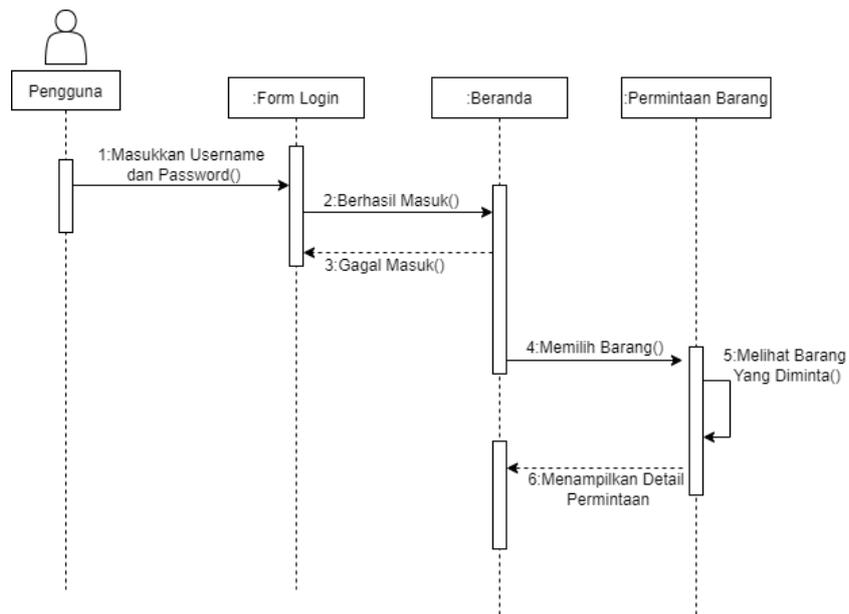
b. Sequence Diagram Login



Gambar 3. 10 Sequence Diagram Login

Pengguna dan admin memasukkan *username* dan *password* berbeda saat *login* ketika berhasil masuk maka akan mengarahkan ke halaman beranda sedangkan gagal maka tetap ke halaman *form login*.

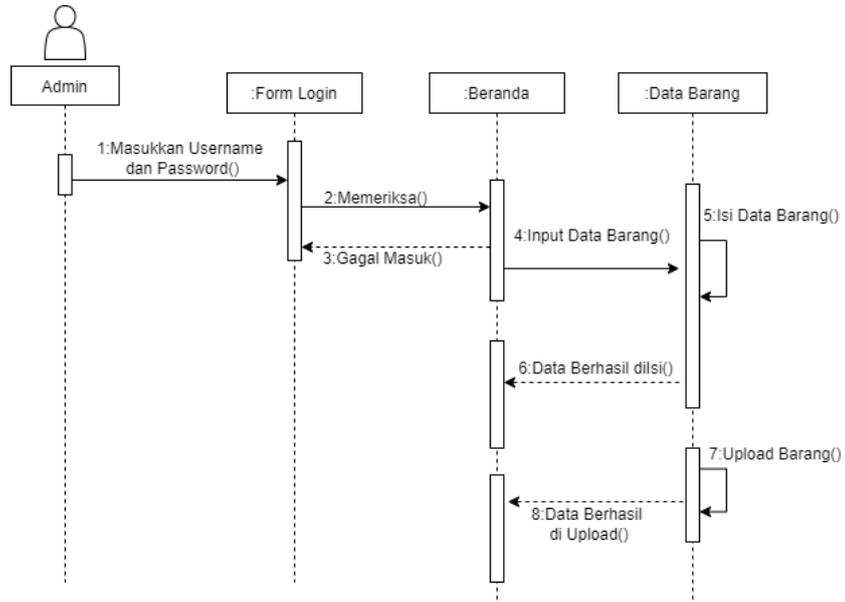
c. Sequence Diagram Permintaan Barang



Gambar 3. 11 Sequence Diagram Permintaan Barang

Untuk permintaan barang pertama harus memilih jenis barang yang diinginkan kemudian melihat barang yang dipilih apakah sesuai yang diinginkan atau tidak, didalam sistem akan menampilkan detail permintaan barang.

d. Sequence Diagram Penginputan Data

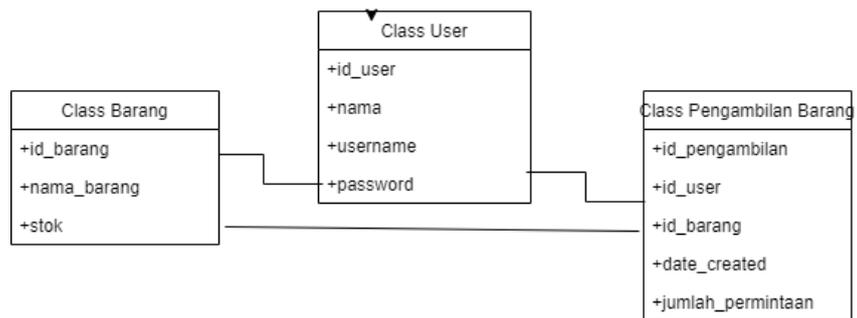


Gambar 3. 12 Sequence Diagram Penginputan Data

Untuk *sequence diagram* penginputan data, admin harus masuk ke halaman beranda untuk *menginput* barang dan *upload* barang jika sudah *menginput* barang dan *upload* barang maka data berhasil disimpan dan akan ditampilkan di halaman beranda.

3.2.4. Class Diagram

Class diagram adalah sebuah struktur yang menggambarkan kelas-kelas yang dibuat untuk sebuah sistem tersebut.



Gambar 3. 13 Class Diagram

Untuk *class diagram* yang terdapat pada aplikasi pergudangan digital berbasis android meliputi 3 bagian class yaitu *Class* Barang, *Class* User, *Class* Pengambilan Barang

1. Struktur tabel

A. Tabel Barang

Nama Tabel: Barang

Fungsi: Menyimpan data barang

Primery key: id

Foreign key: -

Tabel 3. 6 *Class* Barang

No	Nama	Jenis	Ket
1	id_barang	<i>bigint</i>	
2	nama_barang	<i>text</i>	
3	stok	<i>text</i>	

B. Tabel Pengambilan Barang

Nama Tabel: Pengambilan barang

Fungsi: Menyimpan data pengambilan barang

Primery key: id

Foreign key: -

Tabel 3. 7 *Class* Pengambilan Barang

No	Nama	Jenis	Ket
1	id_pengambilan	<i>bigint</i>	
2	id_user	<i>bigint</i>	
3	id_barang	<i>bigint</i>	
4	<i>date_created</i>	<i>timestamp</i>	
5	jumlah_permintaan	<i>text</i>	

D. Tabel *User*

Nama Tabel: *User*

Fungsi: Menyimpan data *user*

Primary key: *id*

Foreign key: -

Tabel 3. 8 *Class User*

No	Nama	Jenis	Ket
1	<i>id_user</i>	<i>bigint</i>	
2	Nama	<i>varchar</i>	
6	<i>Username</i>	<i>varchar</i>	
7	<i>Password</i>	<i>varchar</i>	

3.3. Waktu dan Tempat Penelitian

Waktu pelaksanaan penelitian dimulai dari bulan Maret 2022 sampai dengan bulan Oktober 2022. Penelitian ini dilakukan di kantor PT Multi Karya Sejati MKS Jl. Abd.Dg.Sirua 2 No.14B, Tamamaung, Kec. Panakukkang, Kota Makassar, Sulawesi Selatan 90231

3.4. Alat dan Bahan Penelitian

Tabel 3. 9 Perangkat Keras

No	Nama perangkat keras yang digunakan	Spesifikasi
1	<i>Handphone</i>	Ram : 1GB <i>Storage</i> : 32GB Layar: IPS-LCD 4,7 inci (750x1334 piksel)
2	Printer (HP 2060)	Metode cetak : inkjet <i>Print Resolution</i> : 4800 x 1200 dpi
3	Laptop Dell	<i>Processor</i> : Intel®Core™ i7-4600U CPU @

		2.10GHz 2.70GHz Ram: 8GB
--	--	-----------------------------

Tabel 3. 10 Perangkat Lunak

No	Nama perangkat lunak yang digunakan	Deskripsi
1	<i>Windows 10</i>	Sistem operasi
2	<i>AppInventor</i>	Pengembangan Sistem
3	<i>TinyDB</i>	<i>Database</i>
4	<i>Google chrome dan microsoft edge</i>	<i>Browser</i>

3.5. Metode Pengumpulan Data

Dalam menyusun sebuah penelitian maka dibutuhkan sebuah metode pengumpulan data agar mempermudah sebuah penelitian:

3.5.1. Studi Pustaka

Pengumpulan data pada studi pustaka dilakukan dengan cara mencari dan mengumpulkan sumber informasi melalui jurnal penelitian, buku dan internet sebagai referensi untuk penelitian.

3.5.2. Studi lapangan (wawancara)

Pada sebuah wawancara didasarkan pada pertanyaan-pertanyaan penelitian mengenai sistem pergudangan agar mendapatkan sebuah informasi yang akan dibutuhkan pada saat penelitian.

3.5.3. Studi Lapangan (*observasi*)

Metode pengumpulan data dengan melakukan sebuah pengamatan langsung di lokasi penelitian kantor PT Multi Karya Sejati MKS.

3.6. Metode Pengujian Sistem

Pada penelitian perancangan aplikasi monitoring sistem pergudangan pada PT Multi Karya Sejati MKS Berbasis *Android*, pengujian sistem dilakukan menggunakan *black box* dan *white box* yang dimana pengujian dalam sistem *black box* ini mengamati hasil alur proses dan memastikan data uji sistem apakah sudah berfungsi dengan benar. Jika sudah benar maka sistem pengujian tersebut sudah dipastikan dapat berkerja dengan baik dan bisa digunakan. Sedangkan untuk *white box* mengamati *source code* pada program yang akan diuji apakah *source code* tersebut sudah benar atau tidak.

BAB 4

HASIL DAN PEMBAHASAN

4.1. Hasil Penelitian

Pada bab ini akan dijelaskan keseluruhan hasil dari penelitian yang telah dilakukan. Terdapat dua hak akses *user* pada aplikasi yaitu sebagai pengguna dan petugas gudang.

4.1.1. Tampilan Petugas Gudang

Tampilan aplikasi untuk petugas gudang adalah tampilan beranda atau *home*, kemudian tampilan menu data pengambilan, *add* barang, *list* akun, dan *list* barang.

1.1.1.1. Tampilan *Login*

Tampilan *login* pada petugas gudang.



14:03

Aplikasi Gudang
PT. Multi Karya Sejati

Username
Masukkan username...

Password
Masukkan password...

MASUK

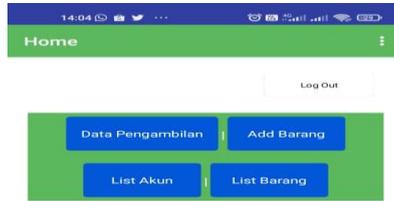
Tidak mempunyai akun? DAFTAR

Gambar 4. 1 Tampilan *Login*

Pada tampilan halaman *login* petugas memasukkan *username* dan *password* agar bisa masuk ke halaman beranda.

1.1.1.2. Tampilan Halaman Beranda

Berikut adalah tampilan menu halaman beranda pada petugas gudang



Gambar 4. 2 Tampilan Halaman Beranda

Pada halaman beranda terdapat tombol data pengambilan, *add* barang, *list* akun, dan *list* barang untuk memudahkan petugas *memonitoring* sistem.

1.1.1.3. Tampilan Add Barang

Tampilan halaman *penginputan* barang yang akan diisi oleh petugas gudang.



Gambar 4. 3 Tampilan Add Barang

Pada tampilan *add* barang, petugas gudang dapat menambah nama barang serta jumlah barang yang akan disimpan.

1.1.1.4. Tampilan Data Barang

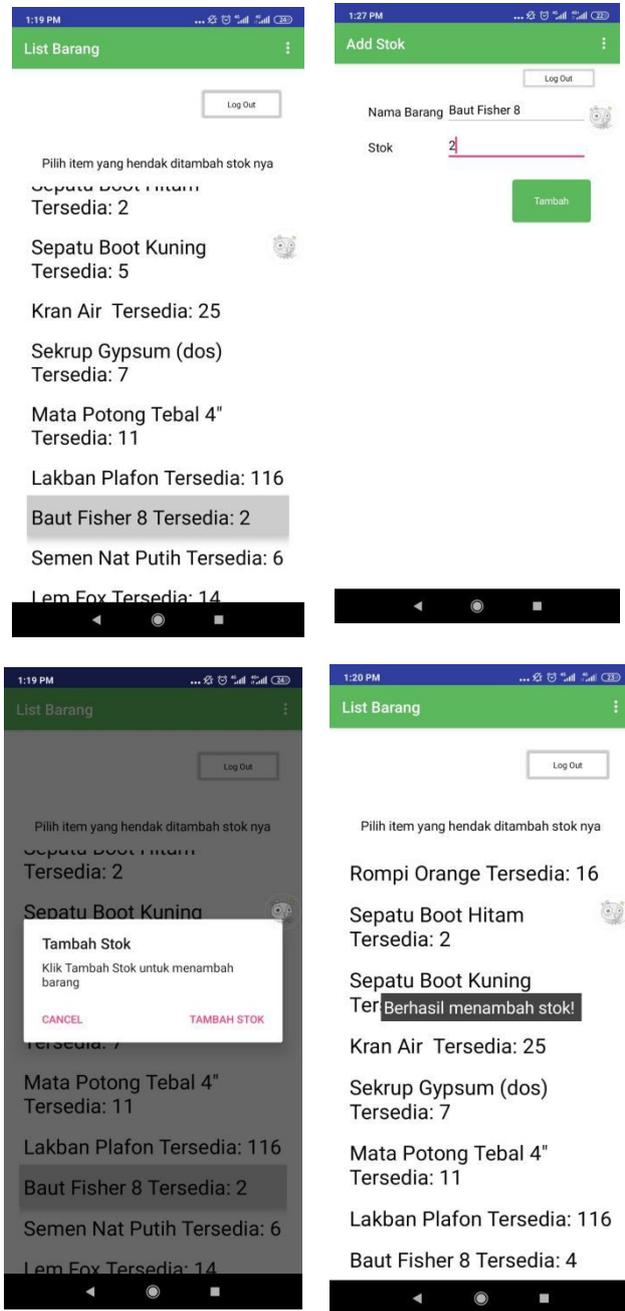
Tampilan data barang yang berhasil disimpan oleh petugas gudang pada *form* pengisian *add* barang



Gambar 4. 4 Tampilan *List* Barang

Pada tampilan halaman data barang petugas dapat melihat data-data barang yang berhasil disimpan serta dapat meng*update* jumlah stok barang dengan menekan salah satu nama barang yang ada.

1.1.1.5. Tampilan Halaman Tambah Stok Barang



Gambar 4. 5 Tampilan Halaman Tambah Stok Barang

Pada tampilan halaman tambah stok barang, petugas memilih barang pada menu *list* data barang yang ingin ditambah jumlahnya, lalu menginput jumlah stok, kemudian data stok berhasil disimpan.

1.1.1.6. Tampilan Halaman Data Pengambilan Barang



Gambar 4. 6 Tampilan Daftar Pengambilan Barang

Pada tampilan halaman pengambilan barang, pengguna yaitu petugas gudang dapat *memonitoring* nama karyawan yang mengambil barang.

1.1.1.7. Tampilan List Akun

Tampilan *list* akun pada petugas gudang



Gambar 4. 7 Tampilan List Akun

Pada tampilan halaman *list* akun, petugas dapat melihat *user- user* yang mengakses sistem stok pergudangan

1.1.2. Tampilan Pengguna

Tampilan aplikasi untuk pengguna adalah tampilan halaman *login*, tampilan halaman pengambilan barang, tampilan halaman *list* barang, dan tampilan halaman *history* pengambilan barang

1.1.2.1. Tampilan Halaman *Login*

Tampilan halaman *login* pada *user* atau karyawan



Gambar 4. 8 Halaman *Login* Pengguna

Pada tampilan halaman *login*, *user* harus memasukan *username* dan *password* agar bisa masuk ke halaman beranda.

1.1.2.2. Tampilan Halaman Pengambilan Barang

Tampilan halaman pengambilan barang pada petugas gudang



Gambar 4. 9 Halaman Pengambilan Barang

Pada halaman pengambilan barang, *user* dapat memilih barang yang akan diminta dengan menekan tombol pilih barang serta *menginput* jumlah barang yang diminta.

1.1.2.3. Tampilan Halaman *List* Barang

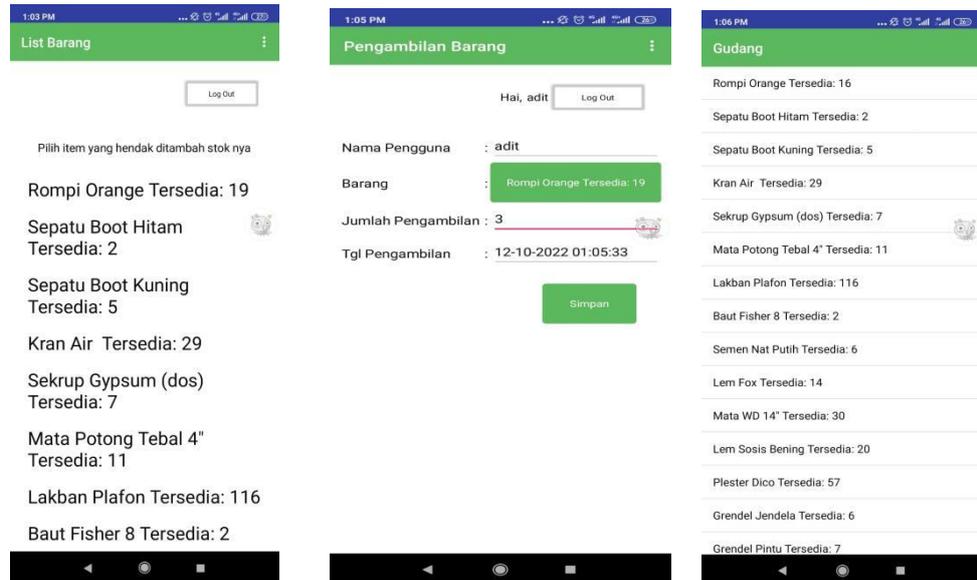
Tampilan halaman *list* barang pada petugas gudang



Gambar 4. 10 Tampilan *List* Barang

Pada tampilan halaman *list* barang memudahkan *user* untuk mencari barang yang diinginkan.

1.1.2.4. Tampilan Halaman *Monitoring* Stok Pergudangan



Gambar 4. 11 Tampilan *Monitoring* Stok Pergudangan

Pada tampilan *monitoring* stok pergudangan, karyawan atau *user* memilih barang yang akan di ambil, kemudian mengisi jumlah pengambilan. Setelah berhasil disimpan, maka jumlah stok pada barang yang dipilih akan berkurang.

1.1.2.5. Tampilan Halaman *History* Pengambilan Barang



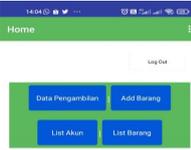
Gambar 4. 12 Tampilan *History* Pengambilan Barang

Pada halaman *history* pengambilan barang, *user* dapat melihat data barang-barang yang telah diambil.

4.2. Pembahasan

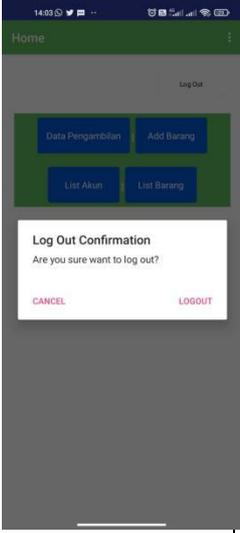
Pengujian dilakukan untuk memastikan bahwa aplikasi yang telah dibuat bebas dari kesalahan dan layak digunakan.

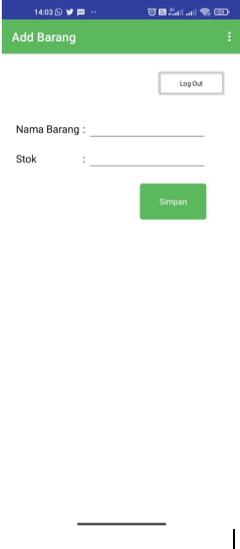
Tabel 4. 1 Pengujian *Black Box*

No	Deskripsi Uji Coba	Sampel Data	Hasil yang diinginkan	Bukti	Hasil yang sebenarnya
1. Login					
1.1	Pengguna menekan tombol “Masuk” dengan menginput <i>username</i> dan <i>password</i> yang sudah diverifikasi dan benar.	<i>Username</i> : petugas <i>Password</i> : petugas	Mengarahkan pengguna ke tampilan beranda atau home setelah berhasil <i>login</i> .		Sesuai dengan yang diharapkan
1.2	Pengguna menekan tombol “Masuk” dengan menginput <i>username</i> dan <i>password</i> yang tidak <i>valid</i> .	<i>Username</i> : asdfghjkl <i>Password</i> : asdfghjkl	Menampilkan pesan kesalahan “ <i>Username</i> atau <i>Password</i> salah”		Sesuai dengan yang diharapkan

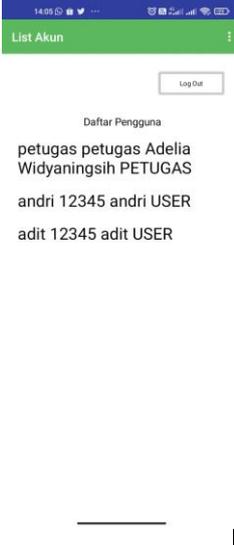
1.3	<p>Pengguna menekan tombol “Masuk” dengan <i>menginput username</i> tanpa <i>menginputkan password</i>.</p>	<p><i>Username</i>: petugas <i>Password</i>: :</p>	<p>Menampilkan pesan kesalahan “Masukkan <i>Password</i>”</p>	 <p>Sesuai dengan yang diharapkan</p>
1.4	<p>Pengguna menekan tombol “Masuk” dengan <i>menginput username</i> yang belum terdaftar.</p>	<p><i>Username</i>: andin <i>Password</i>: : 1234</p>	<p>Menampilkan pesan kesalahan “<i>Username</i> atau <i>password</i> salah”</p>	 <p>Sesuai dengan yang diharapkan</p>
<p>2. Daftar</p>				

2.1	Pegguna menekan tombol “Daftar”	-	Menampilkan <i>form</i> data akun pada tampilan daftar akun	 <p>Sesuai dengan yang diharapkan</p>
2.2	Pegguna menekan tombol “Simpan” dengan isian nama, <i>username</i> , <i>password</i> , dan ulangi <i>password</i>	<p>nama : adit</p> <p><i>Username</i> : adit</p> <p><i>Password</i> : 12345</p> <p><i>Confirm password</i> : 12345</p>	Data tersimpan dan kembali ke halaman <i>login</i>	 <p>Sesuai dengan yang diharapkan</p>
2.3	Pegguna menekan tombol “Simpan” dengan isian nama, <i>username</i> , tanpa mengisi <i>password</i> , dan ulangi	<p>nama : dita</p> <p><i>Username</i> : dita</p> <p><i>Password</i> :</p> <p><i>Confirm password</i> :</p>	Menampilkan pesan kesalahan “ <i>Password</i> harus diisi”	 <p>Sesuai dengan yang diharapkan</p>

	<i>password</i>				
3. Logout					
3.1	Pengguna menekan tombol <i>logout</i>	--	Menampilkan pesan “ <i>Log Out Confirmation</i> ”		Sesuai dengan yang diharapkan
3.2	Pengguna menekan tombol <i>logout</i> pada pesan “ <i>Log Out Confirmation</i> ”	-	Menampilkan halaman <i>login</i>		Sesuai dengan yang diharapkan
4. Add Barang					

4.1	Pegguna menekan menu <i>add</i> barang pada halaman beranda petugas gudang	--	Menampilkan <i>form add</i> barang	 <p>Sesuai dengan yang diharapkan</p>
4.2	Pegguna klik tombol simpan setelah mengisi nama barang dan stok pada <i>form add</i> barang	-	Kembali menampilkan halaman beranda petugas gudang	 <p>Sesuai dengan yang diharapkan</p>
5. Update Stok				

5.1	Pegguna menekan tombol <i>list</i> barang	--	Menampilka n <i>list</i> barang	 <p>14:04 List Barang Log Out Pilih item yang hendak ditambah stok nya Rompi Orange Tersedia: 17 Sepatu Boot hitam Tersedia: 2 Sepatu Boot Kuning Tersedia: 5 Kran Air Tersedia: 29 Sekrup Gypsum Tersedia: 7 Mata Potong Tebal 4" Tersedia: 11 Lakban Plafon Tersedia: 116 Baut Fisher 8 Tersedia: 2 Semen Nat Putih Tersedia: 6</p>	Sesuai dengan yang diharapkan
5.2	Pegguna memilih daftar barang yang akan ditambah stoknya, kemudian menekan tombol tambah stok	-	Menampilka n <i>form add</i> stok	 <p>14:04 Add Stok Log Out Nama Barang Rompi Orange Stok Tambah</p>	Sesuai dengan yang diharapkan
6. List Akun					

6.1	Pegguna menekan tombol <i>list</i> akun	--	Menampilk n halaman daftar pengguna yang telah terdaftar	 <p>Sesuai dengan yang diharapkan</p>
7. Data Pengambilan				
7.1	Pegguna menekan tombol data pengambilan	--	Menampilk n daftar pengambilan barang	 <p>Sesuai dengan yang diharapkan</p>
8. Pengambilan				

8.1	Pengguna menekan tombol pengambilan	--	Menampilkan <i>form</i> isian pengambilan barang	 <p>Sesuai dengan yang diharapkan</p>
8.2	Pengguna menekan tombol pilih barang	--	Menampilkan daftar barang beserta stoknya	 <p>Sesuai dengan yang diharapkan</p>
8.3	Pengguna tidak memilih barang apapun, kemudian klik simpan	-	Menampilkan pesan kesalahan “Pilih Barang Terlebih Dahulu”	 <p>Sesuai dengan yang diharapkan</p>

8.4	Pegguna klik simpan setelah memilih data barang, namun tidak mengisikan jumlah pengambilan		Menampilk n Pesan Kesalahan “ Jumlah Pinjam Tidak Boleh Kosong”	 <p>Sesuai yang diharapkan</p>
8.5	Pegguna klik simpan setelah memilih data barang namun menginputkan jumlah pengambilan melebihi stok		Menampilan Pesan Kesalahan” Stok tidak mencukupi! Kurangi Jumlah Pinjam”	 <p>Sesuai yang diharapkan</p>
8.6	Pegguna klick simpan setelah memilih barang dan mengisi jumlah pengambilan yang sesuai stok		Menampilka n halaman beranda pengguna	

9. History Pengambilan

9.1	Pengguna menekan tombol <i>history</i> pengambilan	--	Menampilkan <i>history</i> pengambilan barang	 <p>Sesuai dengan yang diharapkan</p>
-----	--	----	---	---

Tabel 4. 2 Rekapitulasi Hasil Pengujian *Blackbox*

No	Modular	Berhasil	Tidak berhasil
1.	Pengujian <i>login</i>	✓	-
2.	Pengujian daftar	✓	-
3.	Pengujian <i>logout</i>	✓	-
4.	Pengujian <i>add barang</i>	✓	-
5.	Pengujian <i>update</i> stok	✓	-
6.	Pengujian <i>list</i> akun	✓	-
7.	Pengujian data pengambilan	✓	-
8.	Pengujian pengambilan	✓	-
9.	Pengujian <i>history</i> pengambilan	✓	-

Berdasarkan rekapitulasi hasil pengujian *black box* dapat dinyatakan bahwa seluruh fungsional sistem telah berhasil berfungsi sesuai yang diharapkan, sehingga aplikasi tersebut layak untuk digunakan.

Berdasarkan hasil pengujian pada aplikasi monitoring stok pergudangan pada PT Multi Karya Sejati berbasis *android* yang dilakukan 10 responden, maka didapatkan data sebagai berikut:

Tabel 4. 3 Pengujian Responden

No Des kripsi Uji Coba	Hasil Pengujian Ke-																			
	1		2		3		4		5		6		7		8		9		10	
	B	T	B	T	B	T	B	T	B	T	B	T	B	T	B	T	B	T	B	T
1.1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
1.2	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
1.3	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
1.4	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
2.1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
2.2	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
2.3	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
3.1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
3.2	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
4.1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
4.2	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
5.1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
5.2	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
6.1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
7.1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
8.1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
8.2	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
8.3	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
8.4	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
8.5	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
8.6	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
9.1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0

Tot al Nilai	2 2	0																		
Pres enta se	1 0 0 %	0																		
Keterangan: B = Berhasil T = Tidak Berhasil																				

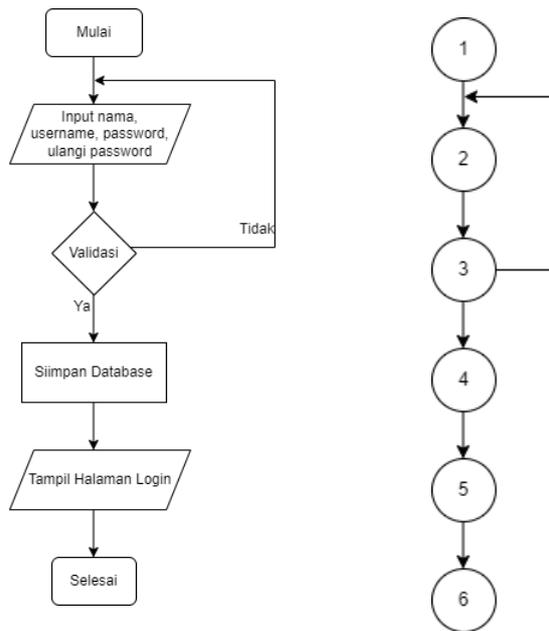
Berdasarkan hasil pengujian sistem untuk *black box*, terdiri dari 10 responden dan 22 pertanyaan mengenai pengujian sistem *black box* pada aplikasi *monitoring* stok pergudangan berbasis *android*. Dimana masing-masing responden memberikan nilai 1 dalam artian (berhasil) pada 22 pertanyaan mengenai pengujian sistem.

4.2.1. Pengujian *White Box*

Pengujian *white box* dilakukan untuk menguji kebenaran logika program yang telah dibuat.

a. Pengujian Daftar

Menu daftar digunakan untuk *user* yang berfungsi untuk mendaftarkan identitas pengguna sebelum memasuki aplikasi tersebut.



Gambar 4. 13 *Flowgraph* Daftar

Dari *flowgraph* daftar diatas dapat diketahui terdapat 6 *edge* dan 6 *node*, berikut ini adalah potongan-potongan *script* dari *node* diatas yaitu:

Script Node 1

```
initialize global tampung_akun to “ ”
```

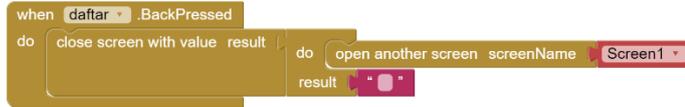
Script Node 2 dan 3

```
when btn_simpan .Click
do
  call daftar .HideKeyboard
  if length nama . Text = 0
  then
    call Notifier1 .ShowAlert
    notice " Nama harus diisi! "
    call nama .RequestFocus
  else if length username . Text = 0
  then
    call Notifier1 .ShowAlert
    notice " Username harus diisi! "
    call username .RequestFocus
  else if length password . Text = 0
  then
    call Notifier1 .ShowAlert
    notice " Password harus diisi! "
    call password .RequestFocus
  else if tf_konfir . Text ≠ password . Text
  then
    call Notifier1 .ShowAlert
    notice " Password tidak sama! "
    call tf_konfir .RequestFocus
  else
    call main
```

Script Node 4

```
to simpan_ke_database
do
  set global tampung_akun to join
  call TinyDB1 .GetValue
  tag " akun "
  valueIfTagNotThere " "
  username . Text
  " | "
  password . Text
  " | "
  nama . Text
  " | "
  " USER "
  " , "
  call TinyDB1 .StoreValue
  tag " akun "
  valueToStore get global tampung_akun
```

Script Node 5 dan 6



Bagan alur registrasi standar untuk kompleksitas siklomatik dihitung dalam tiga cara:

1. Grafik alur memiliki 2 area(*region*)
2. $V(G) = 6 \text{ tepian}(\textit{edge}) - 6 \text{ simpul}(\textit{node}) + 2 = 2$
3. $V(G) = 1 \text{ simpul yang diperkirakan} + 1 = 2$

Oleh karena itu, diagram alur yang ditunjukkan pada gambar memiliki kompleksitas siklomatik 2. Jalur independen adalah :

Jalur(*path*) 1: 1-2-3-4-5-6

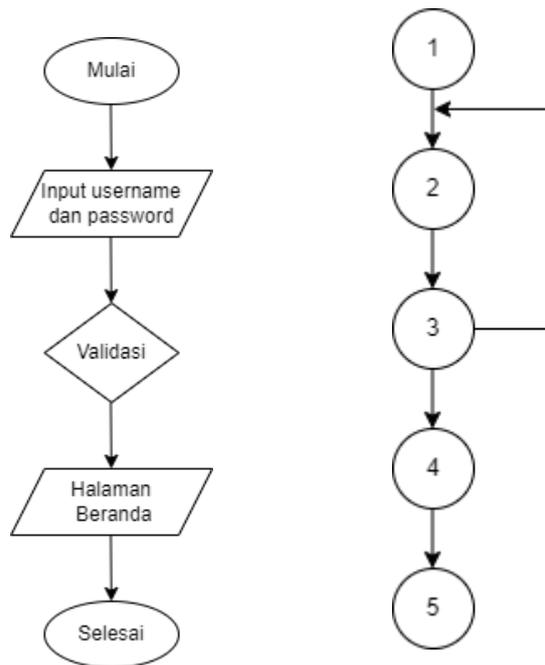
Jalur(*path*) 2: 1-2-3-2-3-4-5-6

Tabel 4. 4 *Path* Daftar

Jalur(<i>path</i>)	1
Jalur(<i>path</i>)	1-2-3-4-5-6
Skenario	Mulai Isi <i>form</i> Daftar, Validasi Menyimpan data Tampil Halaman <i>Login</i> Selesai
Hasil Pengujian	Berhasil
Jalur(<i>path</i>)	2
Jalur(<i>path</i>)	1-2-3-2-3-4-5-6
Skenario	Mulai Isi <i>form</i> Daftar Validasi (ada) Isi <i>form</i> Daftar Validasi (tidak ada)

	Menyimpan data Tampil Halaman <i>Login</i> Selesai
Hasil Pengujian	Berhasil

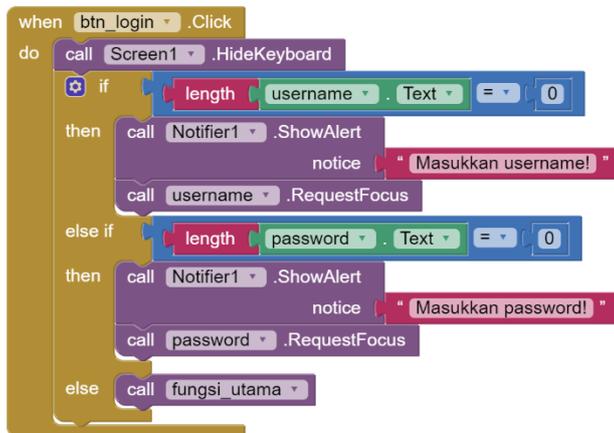
b. Pengujian *Login*



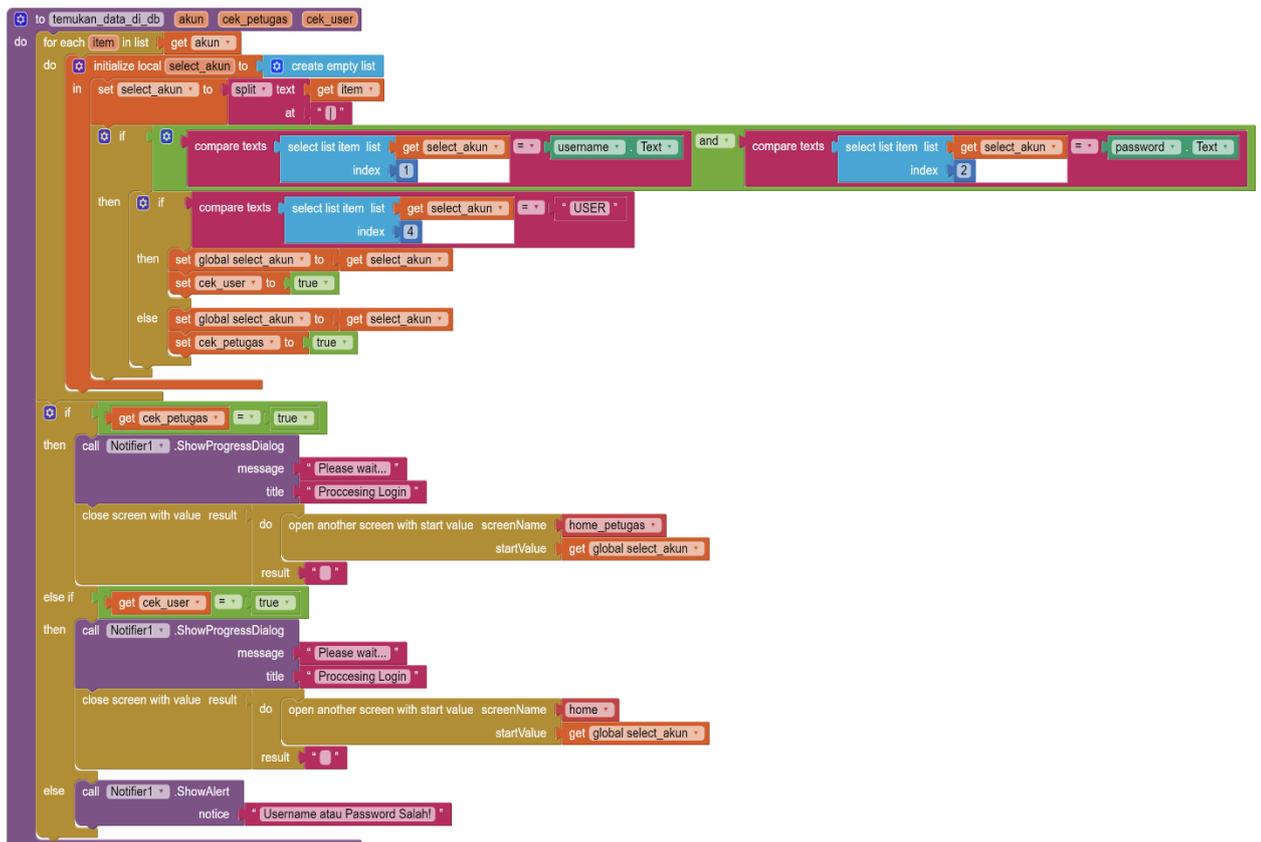
Gambar 4. 14 *Flowgraph Login*

Dari *flowgraph login* diatas dapat diketahui terdapat 5 *edge* dan 5 *node*, berikut ini adalah potongan-potongan *script* dari *node* diatas yaitu:

Script Node 1 dan 2



Script Node 3



Script Node 4 dan 5



Bagan alur pendaftaran standar untuk kompleksitas siklomatik dihitung dalam tiga cara:

1. Grafik alir memiliki dua area(*region*)
2. $V(G) = 5 \text{ tepian}(\textit{edge}) - 5 \text{ simpul}(\textit{node}) + 2 = 2$
3. $V(G) = 1 \text{ simpul yang diperkirakan} + 1 = 2$

Oleh karena itu, diagram alur yang ditunjukkan pada gambar memiliki kompleksitas siklomatik 2. Jalur independen adalah:

Jalur(*path*) 1: 1-2-3-4-5

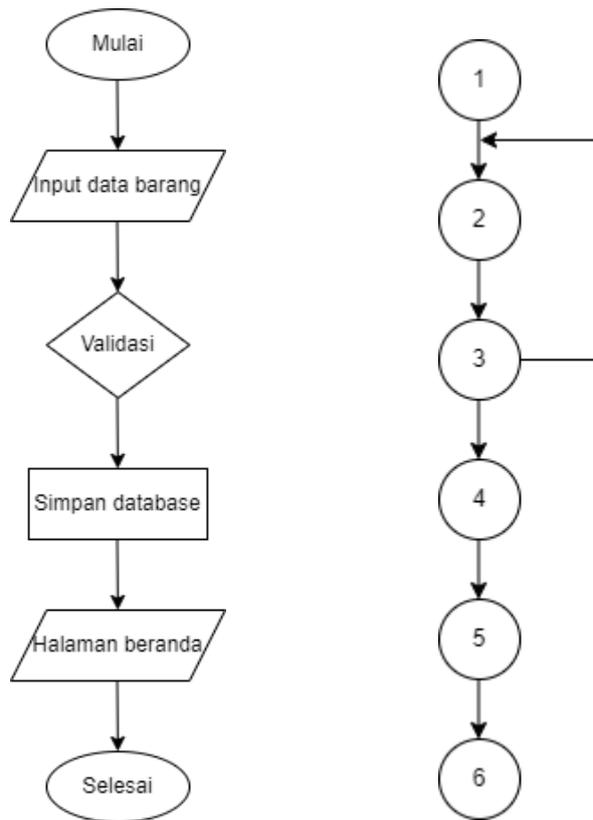
Jalur(*path*) 2: 1-2-3-2-3-4-5

Tabel 4. 5 Jalur(*path*) Login

Jalur(<i>path</i>)	1
Jalur(<i>path</i>)	1-2-3-4-5
Skenario	Mulai Isi <i>form Login</i> Cek Validasi (Jika ada) Menampilkan halaman utama Selesai
Hasil Pengujian	Sukses
Jalur(<i>path</i>)	2
Jalur(<i>path</i>)	1-2-3-2-3-4-5-6
Skenario	Mulai Isi <i>form Login</i> Cek <i>user</i> (Jika tidak ada) Isi <i>form Login</i> Cek Validasi (Jika ada)

	Menampilkan halaman utama Selesai
Hasil Pengujian	Sukses

c. Pengujian *Add Barang*



Gambar 4. 15 *Flowgraph Add Barang*

Dari *flowgraph add* barang diatas dapat diketahui terdapat 6 *edge* dan 6 *node*, berikut ini adalah potongan-potongan *script* dari *node* diatas yaitu:

Script Node 1 Mulai

```
when add_barang ▾ .Initialize
do
  set Label2 ▾ . Text ▾ to select list item list ▾ get start value
  index ▾ 3
  set global session ▾ to get start value
```

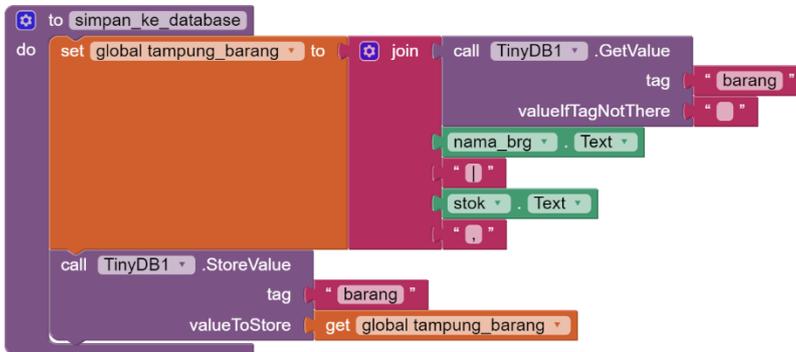
Script Node 2 Input Data Barang

```
set global tampung_barang ▾ to join
call TinyDB1 ▾ .GetValue
tag "barang"
valueIfTagNotThere " "
nama_brg ▾ . Text ▾
" | "
stok ▾ . Text ▾
" | "
```

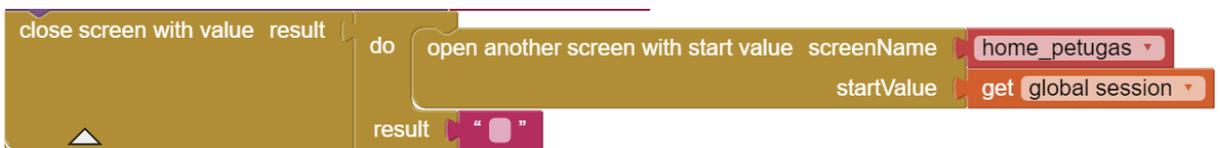
Script 3 Validasi

```
when btn_simpan ▾ .Click
do
  call add_barang ▾ .HideKeyboard
  initialize local data_barang to create empty list
  in set data_barang to split text call TinyDB1 ▾ .GetValue
  tag "barang"
  valueIfTagNotThere " "
  at " | "
  initialize local cek to false
  in for each (item) in list get data_barang
  do
    initialize local select_data to create empty list
    in set select_data to split text get item
    at " | "
    if compare texts select list item list ▾ get (select_data) = nama_brg ▾ . Text ▾
    index 1
    then set cek to true
  end
  if get cek = true
  then call Notifier1 ▾ .ShowAlert
  notice "Barang already exist!"
  else call simpan_ke_database ▾
  call Notifier1 ▾ .ShowProgressDialog
  message "Please wait..."
  title "Saving Data"
  close screen with value result
  do open another screen with start value screenName home_petugas ▾
  startValue get global session ▾
  result
```

Script Node 4 Simpan Database



Script Node 5 dan 6 Tampil Halaman Beranda dan Selesai



Bagan alur pendaftaran standar untuk kompleksitas siklomatik dihitung dalam tiga cara:

1. Grafik alir memiliki dua area(*region*)
2. $V(G) = 6 \text{ tepian(edge)} - 6 \text{ simpul(node)} + 2 = 2$
3. $V(G) = 1 \text{ simpul yang diperkirakan} + 1 = 2$

Oleh karena itu, diagram alur yang ditunjukkan pada gambar memiliki kompleksitas siklomatik 2. Jalur independen adalah:

Jalur(*path*) 1: 1-2-3-4-5-6

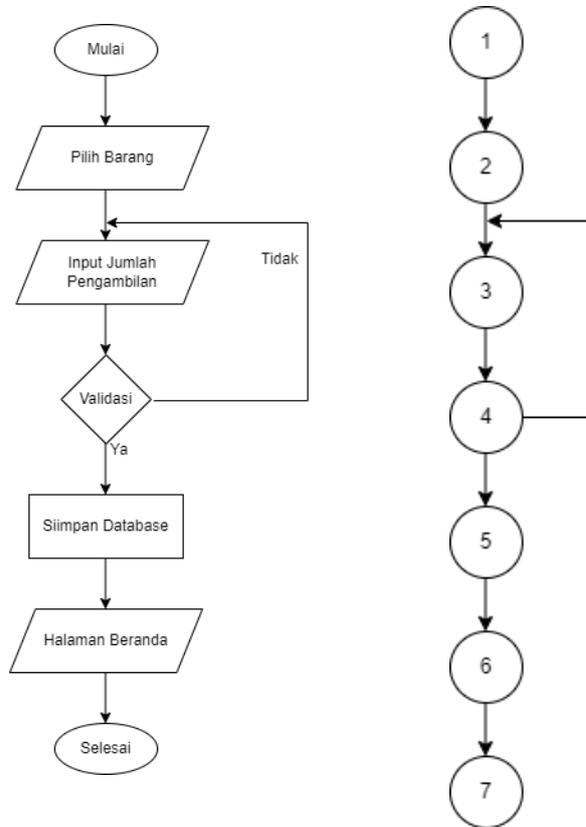
Jalur(*path*) 2: 1-2-3-2-3-4-5-6

Tabel 4. 6 Jalur(*path*) Add Barang

Jalur(<i>path</i>)	1
Jalur(<i>path</i>)	1-2-3-4-5-6
Skenario	Mulai Isi form <i>add barang</i> Cek validasi (Jika tidak ada) Menyimpan <i>database</i>

	Menampilkan beranda Selesai
Hasil Pengujian	Sukses
Jalur(<i>path</i>)	2
Jalur(<i>path</i>)	1-2-3-2-3-4-5-6
Skenario	Mulai Isi <i>form add barang</i> Cek validasi (Jika ada) Isi <i>form add barang</i> Cek validasi (Jika tidak ada) Menyimpan <i>database</i> Menampilkan beranda Selesai
Hasil Pengujian	Sukses

d. Pengujian Pengambilan Barang



Gambar 4. 16 *Flowgraph Add Category*

Dari *flowgraph add category* diatas dapat diketahui terdapat 7 *edge* dan 7 *node*, berikut ini adalah potongan-potongan *script* dari *node* diatas yaitu:

Script Node 1 Mulai

```

when add_pinjam .Initialize
do
  set Label2 . Text to select list item list get start value
                        index 3
  set nama . Text to select list item list get start value
                     index 3
  set tgl . Text to call Clock1 .FormatDateTime
                   instant call Clock1 .Now
                   pattern " dd-MM-yyyy hh:mm:ss "

```

Script Node 2 Pilih Barang



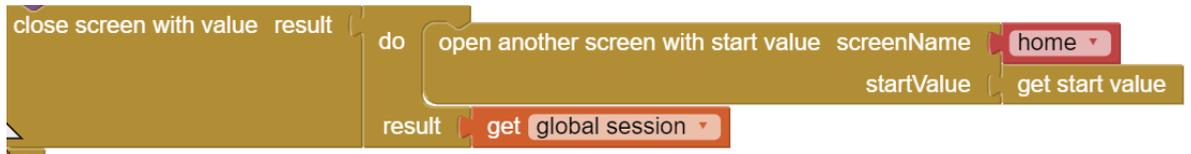
Script Node 3 *Input* Jumlah Barang



Skript Node 3 Simpan ke Database

```
when btn_simpan .Click
do
  call add_pinjam .HideKeyboard
  if length pilih_barang .Selection == 0
  then
    call Notifier1 .ShowAlert
    notice "Pilih barang terlebih dahulu!"
  else if length jmlh .Text == 0
  then
    call Notifier1 .ShowAlert
    notice "Jumlah pinjam tidak boleh kosong!"
  else if jmlh .Text <= 0
  then
    call Notifier1 .ShowAlert
    notice "Jumlah pinjam kurang dari atau sama dengan 0!"
  else
    initialize local barang to create empty list
    in
      set barang to split text pilih_barang .Selection
      at "Tersedia:"
      if jmlh .Text > select list item list get barang
      index 2
      then
        call Notifier1 .ShowAlert
        notice "Stok tidak mencukupi! Kurangi jumlah pinjam."
      else
        set global tampung_barang to replace all text
        segment
          call TinyDB1 .GetValue
          tag "barang"
          valueIfTagNotThere ""
          join
            select list item list get barang
            index 1
            ""
            select list item list get barang
            index 2
          replacement
            join
              select list item list get barang
              index 1
              ""
              select list item list get barang
              index 2
              - jmlh .Text
        call TinyDB1 .StoreValue
        tag "barang"
        valueToStore get global tampung_barang
        set global tampung_peminjaman to join
        call TinyDB1 .GetValue
        tag "peminjaman"
        valueIfTagNotThere ""
        nama .Text
        ""
        select list item list get barang
        index 1
        ""
        jmlh .Text
        ""
        tgl .Text
        ""
        call TinyDB1 .StoreValue
        tag "peminjaman"
        valueToStore get global tampung_peminjaman
        call Notifier1 .ShowAlert
        notice "Berhasil Melakukan Peminjaman!"
```

Script Node 4 dan 5 Tampil Halaman Beranda dan Selesai



Bagan alur pendaftaran standar untuk kompleksitas siklomatik dihitung dalam tiga cara:

1. Grafik alir memiliki dua area(*region*)
2. $V(G) = 7 \text{ tepian}(edge) - 7 \text{ simpul}(node) + 2 = 2$
3. $V(G) = 1 \text{ simpul yang diperkirakan} + 1 = 2$

Oleh karena itu, diagram alur yang ditunjukkan pada gambar memiliki kompleksitas siklomatik 2. Jalur independen adalah:

Jalur(*path*) 1: 1-2-3-4-5-6-7

Jalur(*path*) 2: 1-2-3-4-3-4-5-6-7

Tabel 4. 7 Jalur(*path*) Pengambilan Barang

Jalur(<i>path</i>)	1
Jalur(<i>path</i>)	1-2-3-4-5-6-7
Skenario	Mulai Isi <i>form</i> Pilih barang Isi jumlah pengambilan barang Cek validasi Simpan database Menampilkan halaman beranda Selesai
Hasil Pengujian	Sukses
Jalur (<i>path</i>)	2
Jalur(<i>path</i>)	1-2-3-4-3-4-5-6-7
Skenario	Mulai Isi <i>form</i> pilih barang

	Isi jumlah pengambilan barang Cek validasi(jika tidak sesuai) Isi jumlah pengambilan barang Cek validasi(jika sesuai) Simpan <i>database</i> Menampilkan halaman beranda Selesai
--	--

Tabel 4. 8 Tabel Hasil Pengujian *White Box*

Rekapitulasi Hasil Pengujian <i>White Box</i>			
Halaman yang diuji	<i>Skenario Test Case</i>	Pengamatan	Hasil
Pengujian <i>Daftar</i>	Jalur 1: 1-2-3-4-5-6 Jalur 2: 1-2-3-2-3-4-5	Pengujian yang dilakukan berdasarkan <i>scenario test case</i> aplikasi dapat berjalan dengan sesuai	Berhasil
Pengujian <i>Login</i>	Jalur 1: 1-2-3-4-5 Jalur 2: 1-2-3-2-3-4-5	Pengujian yang dilakukan berdasarkan <i>scenario test case</i> aplikasi dapat berjalan dengan sesuai	Berhasil
Pengujian <i>Add Barang</i>	Jalur 1: 1-2-3-4-5-6 Jalur 2: 1-2-3-4-2-3-4-5-6	Pengujian yang dilakukan berdasarkan <i>scenario test case</i> aplikasi dapat berjalan dengan sesuai	Berhasil

<p>Pengujian Pengambilan Barang</p>	<p>Jalur 1: 1-2-3-4-5-6-7 Jalur 2 : 1-2-3-4-3-4-5-6-7</p>	<p>Pengujian yang dilakukan berdasarkan <i>scenario test case</i> aplikasi dapat berjalan dengan sesuai</p>	<p>Berhasil</p>
-------------------------------------	--	---	-----------------

Berdasarkan rekapitulasi hasil pengujian *white box* terlihat bahwa seluruh logika sistem telah benar dan aplikasi berhasil berjalan sesuai dengan logika atau alur kerja aplikasi.

BAB V

SIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan penelitian yang telah dilakukan maka dapat diambil kesimpulan sebagai berikut:

1. Aplikasi *monitoring* stok pergudangan pada PT Multi Karya Sejati Berbasis *Android* telah dibangun menggunakan *AppInventor* dan *database TinyDB*, aplikasi tersebut dapat diakses oleh petugas gudang dan pengguna. Fitur-fitur yang ada pada aplikasi yang dapat diakses oleh petugas gudang adalah *penginputan* data barang, *update* stok barang, *melihat list user*, *melihat* data pengambilan barang, petugas dapat *memonitoring* stok barang pada gudang melalui fitur *list* barang. Sedangkan fitur yang dapat diakses oleh pengguna adalah *pencatatan* pengambilan barang, dan *melihat* daftar pengambilan barang. Aplikasi yang telah dibangun memudahkan dalam proses *pencatatan* dan *monitoring* stok barang di gudang.
2. Berdasarkan hasil pengujian fungsional menggunakan metode *black box* disimpulkan bahwa seluruh fungsional sistem telah berhasil berfungsi sesuai fungsinya sehingga layak untuk digunakan. Berdasarkan hasil pengujian *white box* tidak terdapat kesalahan logika, maka aplikasi dinyatakan layak karena telah bebas dari segala kesalahan logika.

5.2. Saran

Adapun beberapa saran dalam penelitian tersebut agar bisa dikembangkan lebih baik lagi dari sebelumnya, yaitu:

1. Diharapkan untuk dapat dikembangkan dengan menambahkan fitur-fitur *edit* dan *delete*.
2. Diharapkan untuk memperbaiki *user interfact*.
3. Diharapkan dapat dikembangkan ke sistem operasi *mobile* lainnya.

DAFTAR PUSTAKA

- Abdurahman, Hasan, and Asep Ririh Riswaya. (2014). "Aplikasi Pinjaman Pembayaran Aplikasi Pinjaman Pembayaran Secara Kredit Pada Bank Yudha Bhakti STMIK Mardira Indonesia, Bandung." *Jurnal Computech & Bisnis* 8(2):61–69.
- Agustian, Ricky, and Prionggo Hendradi. (2021). "Analisis Perancangan Sistem Informasi Monitoring Inventory Barang Pada PT. Sumber Laris Abadi Berbasis Android Dengan Metode Fast." 8:122–27.
- Al Hakim, Aldie Lukman, Ilham Maulana, Itmamul Wafa, Yoga Koswara, and Yulianti Yulianti. (2021). "Perancangan Aplikasi Inventaris Gudang Menggunakan Bahasa Program PHP Dan Database MySQL Berbasis WEB." *Jurnal Teknologi Sistem Informasi Dan Aplikasi* 4(1):7.
- Andaru, A. N. D. R. Y. (2018). Pengertian database secara umum. OSF Preprints, 2.
- Andre Pratama, Ikhtison Mekongga (2018). Aplikasi Pencatat Barang Masuk Barang Keluar Berbasis Android di PT. Bhandha Ghara Reksa
- Desa, Kepong, and Bukit Lawang. (2009). "Perancangan Sistem Informasi Desa, K., & Lawang, B. (2009). Perancangan Sistem Informasi Stok Barang Berbasis Web Di PT. Mahesa Cipta. *Gender & Behaviour*, 17(2), 2019, 13007-13015, 17(1), 1–19. <https://doi.org/10.1101/2020.11.10.376129>Stok Barang Berbasis." *Gender & Behaviour*, 17(2), 2019, 13007-13015 17(1):1–19.
- Hasan Abdurrahma, Asep Ririh Riswaya (2014). APLIKASI PEMBELAJARAN TES POTENSI AKADEMIK BERBASIS ANDROID
- Hobson, K, Mayne, R, & Hamilton, J. (2014). A step by step guide to Monitoring and Evaluation. Retrieved December 05, 2016, from Geog.ox.ac.uk:<http://www.geog.ox.ac.uk/research/technologies/projects/mesc/guide-to-monitoring-and-evaluation-v1-march2014.pdf>
- Ismi Mariah Syamsudin (2022). Rancang Bangun Monitoring Pengendalian Intern Untuk Pendistribusian Barang di Gudang
- Jaya, Safitri, and Sindy M. Arif Sembiring. (2019). "Aplikasi Monitoring Inventory Obat Di Gudang PT Kimia Farma Berbasis Android." *Teknologi Informatika Dan Komputer* 11(1):7–12.

- Jogiyanto. (2005). Analisis & Desain Sistem Informasi: pendekatan terstruktur teori dan praktek aplikasi bisnis. Yogyakarta: Andi.
- Listiyan, Edy, and Egia Rosi Subhiyakto. (2021). "Rancang Bangun Sistem Inventory Gudang Menggunakan Metode Waterfall Studi Kasus Di Cv. Aqualux Duspha Abadi Kudus Jawa Tengah." *KONSTELASI: Konvergensi Teknologi Dan Sistem Informasi* 1(1):74–82.
- Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. (2015). Pengujian aplikasi menggunakan black box testing boundary value analysis (studi kasus: Aplikasi prediksi kelulusan smnptn). *Jurnal Ilmiah Teknologi Infomasi Terapan*, 1(3).
- Ndaru Adi Pratama, Catur Hermawan (2016). APLIKASI PEMBELAJARAN TES POTENSI AKADEMIK BERBASIS ANDROID
- Nidhra, Srinivas, and Dondeti, Jagruthi, (2012), Blackbox and Whitebox Testing Techniques – A Literature Review, *International Journal of Embedded Systems and Applications (IJESA)* Vol.2, No.2, June 2012
- Nugroho, A. (2010). Rekayasa perangkat lunak berorientasi objek dengan metode USDP. Penerbit Andi.
- Prasetyo, Tri Ferga, and Ade Bastian. (2017). "VISUALISASI EDUKATIF PENYIARAN TELEVISI SATELIT DAN TELEVISI ANTENA MENGGUNAKAN METODE MULTIMEDIA DEVELOPMENT LIFE CYCLE (MDLC)." (Mdlc):184–90.
- Pratama, Andre, Ikhtison Mekongga, Jurusan Teknik Komputer, Politeknik Negeri Sriwijaya, Jalan Srijaya, and Negara Bukit. (1978). "Aplikasi Pencatat Barang Masuk Barang Keluar Berbasis Android Di PT . Bhandha Ghara Reksa." 12(x):47–56.
- PSC- Public Service Commision Republic of South Africa (2008). Basic Concepts in Monitoring And Evaluation. Retrieved December 05, 2016, from Public Service Commission Republic of South Africa;:http://www.psc.gov.za/documents/docs/guidelines/PSC%206%20in%20one.pdf
- Putra, Rachmat Kurniawan, Ilhamsyah, and Nurul Mutiah. (2022). "Sistem Informasi Pergudangan Dan Pengendalian Persediaan Stok Barang Menggunakan Metode Economic Order Quantity." *Komputer Dan Aplikasi* 10(01):23–33.
- Sari, and Alvionita. (2021). "Desain Fisik Aplikasi Nomor Antrean Pendaftaran Pasien Rawat Jalan Berbasis Android Di Puskesmas Nganjuk." *Jurnal Rekam Medis Dan Informasi Kesehatan Indonesia (Jurmiki)* 01(Pelayanan Kesehatan):39–46.

- Setiadi, B. (2019). “APLIKASI MONITORING MATERIAL PERGUDANGAN PADA PT. PLN (Persero) AREA BANJARMASIN.” *Technologia: Jurnal Ilmiah* 10(4):179–93.
- Subandi, Husein. (2021). “Augmented Reality Dalam Mendeteksi Produk Rotan Menggunakan Metode Multimedia Development Life Cycle (MDLC).” 6(2):135–41.
- Syamsudin, Ismi Mariah. (2022). “Rancang Bangun Monitoring Pengendalian Intern Untuk Pendistribusian Barang Di Gudang.” 2(5):1–11.
- Wibisono, W., & Baskoro, F. (2002). Pengujian perangkat lunak dengan menggunakan model behaviour UML. *Jurnal Ilmiah Teknologi Informasi*, 1(1), 43-50.

LAMPIRAN

Script Pengujian Daftar

```
initialize global tampung_akun to ""

to simpan_ke_database
do
  set global tampung_akun to
  join
    call TinyDB1 .GetValue
    tag "akun"
    valueIfTagNotThere ""
    username . Text
    ""
    password . Text
    ""
    nama . Text
    ""
    "USER"
    ""
  call TinyDB1 .StoreValue
  tag "akun"
  valueToStore get global tampung_akun

when btn_simpan .Click
do
  call daftar .HideKeyboard
  if length nama . Text = 0
  then
    call Notifier1 .ShowAlert
    notice "Nama harus diisi!"
    call nama .RequestFocus
  else if length username . Text = 0
  then
    call Notifier1 .ShowAlert
    notice "Username harus diisi!"
    call username .RequestFocus
  else if length password . Text = 0
  then
    call Notifier1 .ShowAlert
    notice "Password harus diisi!"
    call password .RequestFocus
  else if tf_konfir . Text != password . Text
  then
    call Notifier1 .ShowAlert
    notice "Password tidak sama!"
    call tf_konfir .RequestFocus
  else
    call main

when daftar .BackPressed
do
  close screen with value result
  do
    open another screen screenName Screen1
    result ""
```

Script Pengujian Login

```
when btn_login .Click
do
  call Screen1 .HideKeyboard
  if length username .Text = 0
  then
    call Notifier1 .ShowAlert
    notice "Masukkan username!"
    call username .RequestFocus
  else if length password .Text = 0
  then
    call Notifier1 .ShowAlert
    notice "Masukkan password!"
    call password .RequestFocus
  else
    call fungsi_utama

to temukan_data_di_db akun cek_petugas cek_user
do
  for each (item) in list get_akun
  do
    initialize local select_akun to create empty list
    in set select_akun to split text get item
    at ' '
    if compare texts select list item list get select_akun = username .Text and compare texts select list item list get select_akun = password .Text
    then
      if compare texts select list item list get select_akun = USER
      then
        set global select_akun to get select_akun
        set cek_user to true
      else
        set global select_akun to get select_akun
        set cek_petugas to true
    if get cek_petugas = true
    then
      call Notifier1 .ShowProgressDialog
      message "Please wait..."
      title "Processing Login"
      close screen with value result do
        open another screen with start value screenName home_petugas
        startValue get global select_akun
      result
    else if get cek_user = true
    then
      call Notifier1 .ShowProgressDialog
      message "Please wait..."
      title "Processing Login"
      close screen with value result do
        open another screen with start value screenName home
        startValue get global select_akun
      result
    else
      call Notifier1 .ShowAlert
      notice "Username atau Password Salah!"

then
  call Notifier1 .ShowProgressDialog
  message "Please wait..."
  title "Processing Login"
  close screen with value result do
    open another screen with start value screenName home
    startValue get global select_akun
  result
```

Script Add Barang

```

when add_barang .Initialize
do
  set Label2 . Text to select list item list index 3 get start value
  set global session to get start value

set global tampung_barang to join
  call TinyDB1 .GetValue tag "barang" valueIfTagNotThere ""
  nama_brg . Text " "
  stok . Text " "

when btn_simpan .Click
do
  call add_barang .HideKeyboard
  initialize local data_barang to create empty list
  in set data_barang to split text call TinyDB1 .GetValue tag "barang" valueIfTagNotThere "" at " "
  initialize local cek to false
  in for each (item) in list get data_barang
  do
    initialize local select_data to create empty list
    in set select_data to split text get item at " "
    if compare texts select list item list index 1 nama_brg . Text
    then set cek to true
  if get cek = true
  then call Notifier1 .ShowAlert notice "Barang already exist!"
  else call simpan_ke_database
  call Notifier1 .ShowProgressDialog message "Please wait..." title "Saving Data"
  do close screen with value result open another screen with start value screenName home_petugas startValue get global session result ""

to simpan_ke_database
do
  set global tampung_barang to join
  call TinyDB1 .GetValue tag "barang" valueIfTagNotThere ""
  nama_brg . Text " "
  stok . Text " "
  call TinyDB1 .StoreValue tag "barang" valueToStore get global tampung_barang
  
```

```
close screen with value result
do
  open another screen with start value screenName home_petugas
  startValue get global session
  result ""
```

Script Pengambilan Barang

```
when add_pinjam .Initialize
do
  set Label2 . Text to select list item list get start value
  index 3
  set nama . Text to select list item list get start value
  index 3
  set tgl . Text to call Clock1 .FormatDateTime
  instant call Clock1 .Now
  pattern "dd-MM-yyyy hh:mm:ss"

when pilih_barang .AfterPicking
do
  set pilih_barang . Text to pilih_barang . Selection

length jmlh . Text = 0
```

Script Database

```
when btn_simpan .Click
do
  call add_pinjam .HideKeyboard
  if length pilih_barang Selection = 0
  then call Notifier1 .ShowAlert
       notice "Pilih barang terlebih dahulu!"
  else if length jmlh Text = 0
  then call Notifier1 .ShowAlert
       notice "Jumlah pinjam tidak boleh kosong!"
  else if jmlh Text ≤ 0
  then call Notifier1 .ShowAlert
       notice "Jumlah pinjam kurang dari atau sama dengan 0!"
  else
    initialize local barang to create empty list
    in set barang to split text pilih_barang Selection
       at "Tersedia:"
    if jmlh Text > select list item list get barang
       index 2
    then call Notifier1 .ShowAlert
         notice "Stok tidak mencukupi! Kurangi jumlah pinjam."
    else
      set global tampung_barang to replace all text
      call TinyDB1 .GetValue
         tag barang
         valueIfTagNotThere ""
      segment join select list item list get barang
         index 1
         ""
      replacement join select list item list get barang
         index 2
         ""
      select list item list get barang
         index 2
         jmlh Text
      call TinyDB1 .StoreValue
         tag barang
         valueToStore get global tampung_barang
      set global tampung_peminjaman to join
         call TinyDB1 .GetValue
            tag peminjaman
            valueIfTagNotThere ""
            nama Text
            select list item list get barang
               index 1
            jmlh Text
            tgl Text
            ""
      call TinyDB1 .StoreValue
         tag peminjaman
         valueToStore get global tampung_peminjaman
      call Notifier1 .ShowAlert
         notice "Berhasil Melakukan Peminjaman!"
    end
  end
end
```