BAB I

PENDAHULUAN

A. Latar Belakang

Desa Botto Mallangga, yang terletak di Kecamatan Maiwa, Kabupaten Enrekang, merupakan salah satu dari banyak desa di Indonesia yang menerima dana Bantuan Langsung Tunai (BLT) dari Dana Desa. BLT ini bertujuan untuk membantu masyarakat desa yang terdampak oleh berbagai krisis ekonomi. Namun, proses pencatatan dan pengelolaan bantuan BLT di Desa Botto Mallangga masih dilakukan secara manual. Hal ini menimbulkan berbagai permasalahan, seperti, kesalahan pencatatan, dan kurangnya transparansi serta akuntabilitas.

Sistem informasi keuangan sangat penting untuk mengelola aliran dana, memastikan transparansi, akuntabilitas, dan mengurangi risiko penyelewengan dana. Dengan adanya sistem ini, pemerintah desa dapat memantau dan melaporkan penggunaan dana secara real-time, memudahkan proses pencatatan, dan meningkatkan kepercayaan masyarakat terhadap pengelolaan dana desa. Penerapan sistem informasi keuangan dalam pengelolaan BLT Dana Desa di Desa Botto Mallangga diharapkan dapat memberikan solusi terhadap permasalahan yang ada dan meningkatkan efisiensi serta efektivitas penyaluran bantuan.

Metode pencatatan manual yang saat ini di gunakan di Desa Botto Mallangga memiliki beberapa kelemahan, antara lain: keterlambatan dan ketidakakuratan data, proses manual sering kali memakan waktu lebih lama dan rentang terhadap kesalahan manusia, yang dapat mengakibatkan data yang tidak

akurat kurangnya transparansi dan akuntabilitas, tanpa sistem yang terintegrasi, sulit bagi pihak terkait untuk memantau dan mengaudit aliran dana secara efektif, serta sulitnya akses informasi, masyarakat desa kesulitan untuk mendapatkan informasi yang akurat dan terkini mengenai penyaluran BLT.

Dengan adanya penelitian ini di harapkan terciptanya, sistem informasi keuangan akan menjadi alat yang berguna untuk pengelolaan dana BLT-DD dan akan meningkatkan efisiensi operasional, mengurangi risiko kesalahan dan pengawasan, meningkatkan transparansi dan akuntabilitas dalam manajemen dana bantuan.

B. Rumuasan Masalah

Berdasarkan uraian latar belakang, maka rumusan masalah penelitian ini adalah "Bagimana cara merancang sebuah sistem informasi berbasis android dalam pengolahan data untuk mengurangi resiko penyalahgunaan dan kecurangan dalam penyaluran Bantuan Langsung Tunai Dana Desa (BLT-DD)?"

C. Batasan Masalah

Adapun batasan masalah dari penelitian ini adalah:

1. Penelitian ini hanya berfokus pada penerapan sistem informasi penyaluran keuangan sebagai solusi untuk mengurangi resiko penyalahgunaan dan kehilangan data dalam penyaluran Bantuan Langsung Tunai Dana Desa (BLT-DD).
2. Penelitian hanya berfokus pada analisis dan pemecahan masalah yang di hadapi dalam proses pencatatan manual yang saat ini digunakan, seperti

keterlambatan, ketidakakuratan data kurangnya trasparansi dan kesulitan akses informasi.

D. Tujuan Penelitian

Tujuan dari penelitian ini adalah menghasilkan sebuah sistem informasi berbasis android untuk mengolah data secara transparan, yang efektif dan efisien dalam mengurangi risiko kecurangan dan penyalahgunaan penyaluran Bantuan Langsung Tunai Dana Desa (BLT-DD).

E. Manfaat Penelitian

Adapun manfaat yang di harapkan dalam pembuatan sistem ini yaitu:

1. Manfaat bagi penulis

   Penelitian ini menjadi momen bagi penulis untuk memperluas wawasan dan pengetahuan, serta mengimplementasikan teori-teori yang telah diperoleh selama proses perkuliahan. Selain itu, penelitian ini juga merupakan syarat yang harus dipenuhi oleh penulis untuk menyelesaikan pendidikan di bangku perkuliahan.

2. Manfaat bagi pembaca

   Penelitian ini diharapkan dapat menjadi sumbangan literatur ilmiah tentang penggunaan teknologi informasi dalam sistem informasi keuangan dalam penyaluran bantuan, selain itu sistem yang dikembangkan dapat menjadi inspirasi bagi penelitian atau pengembangan lebih lanjut dengan bidang penelitian yang sama.

3. Manfaat bagi masyarakat

   Dengan adanya penelitian ini diharapkan menghasilkan sebuah sistem

aplikasi yang dapat memudahkan masyarat dalam mendapatkan informasi terkait penyaluran bantuan, serta dapat mengawal dan mendukung transparansi proses penyaluran BLT-DD kepada masyarakat Desa Botto Mallangga.

BAB II

TINJAUAAN PUSTAKA

A. Kajian Hasil Penelitian Terdahulu

Penelitian mengacu pada beberapa penelitian sebelumnya yang menjadi acuan dalam penelitian ini, penelitian yang relevan tersebut antara lain:

1. Sukma, E., Sutrisno, M., Risdayani, N., & Rahmayani, M.T.I. (2023) "Perancangan Sistem Informasi Penyaluran Dana Bantuan Langsung Tunai (BLT) Pada Desa Pambang Pesisir" Tujuan penelitian ini adalah untuk memperoleh beberapa kemudahan dalam pendataan penyaluran dana BLT dan melakukan perancangan agar dapat mempermudah dalam pendataan, untuk mengetahui para penerima dan penyaluran yang dilakukan secara online.
2. Mangdalena, H., Aprilianiza, N., & Santoso, H. (2023) "Pengembangan Sistem Informasi Penyaluran dan Penatausahaan Bantuan Dana Desa Berbasis Web" Tujuan dari penelitian ini adalah untuk memudahkan pihak perangkat desa dalam mendata, mengelola, menyalurkan, sampai melaporkan pemanfaatan dana desa agar sesuai dengan ketentuan undang – undang dan peraturan yang berlaku.
3. Suherman, Waru, M.V., & Nurnaningsih (2022) "Perancangan Aplikasi Pendataan Bantuan Langsung Tunai (BLT) Berbasis web Pada Kantor Desa Maccile Kecamatan Lalabata Kabupaten Soppeng" Tujuan Penelitian ini adalah Untuk menyelesaikan permasalahan yang ada pada Kantor Desa Maccile diperlukan teknologi yang mendukung, yakni dengan

mengimplementasikan sistem informasi pendataan BLT dana desa berbasis web.

4. Setyaningrum, A.N., Subhi H.D., Ekojono (2024) "Sistem Penentu Penerimaan Bantuan Sosial (BLT-DD) Untuk Keluarga Kurang Mampu" Tujuan dari penelitian ini adalah Kriteria yang digunakan dalam menentukan penerima BLT-DD berdasarkan parameter seperti status tempat tinggal, jenis lantai bangunan, jenis dinding bangunan, sumber penerangan, usia, status perkawinan, penghasilan, jumlah tanggungan, jumlah ternak, penerima bantuan pemerintah, kehilangan pekerjaan, dan anggota keluarga sakit. Metode AHP diterapkan untuk menilai dan menentukan bobot dari setiap kriteria berdasarkan penilaian multi-expert, sedangkan TOPSIS digunakan untuk menentukan skor akhir dari penerima BLT-DD.

B. Sistem Informasi

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan tertentu untuk mencapai tujuan tertentu.

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna bagi penerimanya untuk mengambil keputusan saat ini atau saat mendata.

Sistem Informasi adalah sekumpulan prosedur organisasi pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan dan atau untuk mengendalikan organisasi. (Asmara, 2019)

C. BLT Dana Desa

Bantuan Langsung Tunai Dana Desa (BLT) adalah penyerahan sejumlah uang tunai kepada warga desa miskin yang beralamat di Desa untuk mengurangi dampak ekonomi. Sehingga BLT desa disalurkan untuk menopang daya tahan ekonomi masyarakat desa khususnya bagi masyarakat miskin atau kurang mampu (Magdalena, Aprilianiza, & Santoso, 2023).

BLT adalah program bantuan pemerintah dengan memberikan uang tunai atau bantuan lainnya dengan persyaratan ataupun tanpa persyaratan untuk masyarakat miskin. Berdasarkan amanah Undang-Undang Dasar (UUD) 1945, pada pasal 33 dan 34, Negara bertanggung jawab untuk mlindungi segenap bangsa Indonesia dan memajukan kesejahteraan umum dalam rangka mewujudkan keadilan sosial bagi seluruh rakyat Indonesia. Salah satu upaya pemerintah dalam memenuhi kesejahteraan sosial tersebut adalah dengan memberikan bantuan sosial (UU Nomor 11 Tahun 2009).

Desa adalah kesatuan masyarakat hukum dengan batas-batas wilayah yang di berdayakan untuk mengatur dan mengurus kepentingan masyarakat berdasarkan asal usul dan adat istiadat setempat yang diakui dan dihormati oleh sistem pemerintahan. NKRI (Undang-Undang Nomor 32 Tahun 2004 tentang pemerintahan daerah pasal 1 ayat 12). Pelaksanaan otonomi daerah desa memberdayakan pemerintah desa untuk mengelola, mengatur dan menata rumahnya sendiri, menambah beban tanggung jawab dan kewajiban desa. Tanggung jawab yang dipermasalahakan antara lain pengelolaan anggaran desa,

salah satunya pelaporan. laporan dana desa harus transparan dan mudah dipahami agar tidak terjadi penyimpangan. (Suherman, 2022)

Upaya pemerintah dalam penanggulangan kemiskinan di Indonesia adalah dengan menyalurkan dana BLT melalui desa-desa. Maka dari itu, peran aparat desa sangat penting dalam proses penyaluran BLT ini, proses penyaluran BLT di Desa masih berjalan sampai sekarang namun proses seleksi penerima BLT tidak objektif sehingga penyaluran dana tidak tepat sasaran, dan juga terjadi manipulasi data karena pendataan masih di-input secara manual dengan pencatatan di buku besar dan Microsoft Excel. Masyarakat yang seharusnya menerima BLT terabaikan dan tidak terdaftar pada daftar penerima BLT. Sedangkan masyarakat yang kondisi ekonominya stabil, masuk ke dalam daftar penerima BLT.

Kriteria dan syarat penerima BLT (Bantuan Langsung Tunai) desa menurut Peraturan Menteri Keuangan (PMK) nomor 190/PMK.07/2021 tentang Pengelolaan Dana Desa, pada tahun 2022 sebagai berikut:

a. Keluarga miskin atau tidak mampu yang berdomisili dan diprioritaskan untuk keluarga miskin yang termasuk dalam kategori kemiskinan ekstrem.
b. Masyarakat yang kehilangan mata pencaharian
c. Keluarga miskin penerima jaring pengaman sosial lainnya yang terhenti baik yang bersumber dari APBD dan atau APBN.
d. Rumah tangga dengan satu anggota rumah tangga lanjut usia.

(Sutianah, 2023)

Adapun syarat dan ketentun Operasional Penerimaan bantuan BLT-DD ditetapkan dalam permendesa No.13 tahun 2023 yang terdapat pada pasal 3 hingga 4 sebagai berikut

Pasal 3

1. Fokus penggunaan Dana Desa untuk penanganan kemiskinan ekstrem sebagaimana dimaksud dalam Pasal 2 ayat (1) huruf a berupa Bantuan Langsung Tunai Desa.
2. Bantuan Langsung Tunai Desa sebagaimana dimaksud pada ayat (1) diberikan kepada keluarga penerima manfaat diprioritaskan keluarga miskin ekstrem dan keluarga miskin yang berdomisili di Desa bersangkutan.
3. Keluarga penerima manfaat sebagaimana dimaksud pada ayat (2) ditetapkan berdasarkan kriteria:
   a. kehilangan mata pencaharian;
   b. mempunyai anggota keluarga yang rentan sakit menahun, sakit kronis, dan/atau penyandang disabilitas;
   c. tidak menerima bantuan sosial program keluarga harapan;
   d. rumah tangga dengan anggota rumah tangga tunggal lanjut usia; atau
   e. perempuan kepala keluarga dari keluarga miskin ekstrem.
4. Dalam menentukan keluarga penerima manfaat sebagaimana dimaksud pada ayat (3), Pemerintah Desa dapat menggunakan data pemerintah sebagai acuan untuk menetapkan keluarga penerima manfaat Bantuan Langsung Tunai Desa.

5. Daftar keluarga penerima manfaat sebagaimana dimaksud pada ayat (2), ayat (3), dan ayat (4) dibahas dan disepakati dalam Musyawarah Desa dan ditetapkan dengan keputusan kepala Desa.

Pasal 4

1. Bantuan Langsung Tunai Desa sebagaimana dimaksud dalam Pasal 3 ayat (1) dialokasikan paling tinggi 25% (dua puluh lima persen) dari pagu Dana Desa setiap Desa.
2. Besaran Bantuan Langsung Tunai Desa sebagaimana dimaksud pada ayat (1) ditetapkan sebesar Rp300.000,00 (tiga ratus ribu rupiah) setiap bulan.
3. Bantuan Langsung Tunai Desa sebagaimana dimaksud pada ayat (2) diberikan selama 12 (dua belas) bulan per keluarga penerima manfaat.
4. Pemberian Bantuan Langsung Tunai Desa sebagaimana dimaksud pada ayat (3) dilaksanakan mulai bulan Januari dan dapat dibayarkan paling banyak untuk 3 (tiga) bulan secara sekaligus.

Adapun Prosedur Penyaluran Bantuan Langsung Tunai Dana Desa (BLT-DD) Pada Desa Botto Mallangga Kec.Maiwa Kabupaten Enrekang sebagai berikut:

a. Mengelola Data Calon Penerima BLT-DD

Langkah pertama dalam prosedur penyaluran BLT-DD adalah mengelola data calon penerima. Data ini biasanya diperoleh dari hasil pendataan di desa yang melibatkan Kepala Desa, Kepala Dusun Dan Perangkat Desa Lainnya. Data calon penerima mencakup informasi seperti Nama, Alamat, Kondisi Ekonomi, Dan Status Sosial. Proses pengelolaan ini

bertujuan untuk memastikan bahwa hanya warga yang benar-benar memenuhi kriteria yang ditetapkan yang akan menerima bantuan.

b. Melakukan Survei (Peninjauan)

Setelah data calon penerima terkumpul, langkah selanjutnya adalah melakukan survei atau peninjauan lapangan. Survei ini dilakukan oleh tim yang ditunjuk oleh desa atau pemerintah daerah untuk memverifikasi kondisi calon penerima secara langsung. Survei ini melibatkan kunjungan ke rumah-rumah calon penerima untuk memastikan keabsahan data dan kondisi ekonomi mereka. Proses ini penting untuk menghindari adanya penerima fiktif atau tidak layak.

c. Menyerahkan Hasil Survei

Hasil dari survei kemudian diserahkan kepada pihak yang berwenang untuk ditinjau dan disahkan. Hasil survei ini mencakup laporan lengkap tentang kondisi setiap calon penerima, disertai rekomendasi apakah mereka layak menerima bantuan atau tidak.

d. Melakukan Penyaluran

Setelah daftar penerima bantuan disahkan, tahap berikutnya adalah melakukan penyaluran BLT-DD. Penyaluran ini dilakukan secara langsung dengan menyerahkan sembako dan uang tuani, Proses penyaluran ini harus dilakukan dengan transparan dan tertib untuk memastikan bantuan sampai kepada yang berhak sesuai jadwal yang telah ditetapkan.

e. Menerima Penyaluran Bantuan

Pada tahap ini, penerima bantuan akan menerima BLT-DD sesuai dengan ketentuan yang berlaku. Penerima bantuan harus menandatangani bukti penerimaan dan menunjukkan identitas diri untuk memastikan bahwa bantuan diterima oleh orang yang tepat. Proses ini diawasi oleh aparat desa untuk mencegah penyalahgunaan dan memastikan akuntabilitas.

f. Membuat Laporan Hasil Penyaluran Bantuan

Tahap terakhir dalam prosedur ini adalah membuat laporan hasil penyaluran bantuan. Laporan ini mencakup jumlah dana yang disalurkan, jumlah penerima bantuan, serta kendala dan temuan selama proses penyaluran. Laporan ini dibuat oleh tim pelaksana dan diserahkan kepada pihak yang berwenang sebagai bentuk pertanggung jawaban.

D. Aplikasi

Pengertian Aplikasi Menurut Kamus Besar Bahasa Indonesia (KBBI) adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu. Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari user (pengguna).

Aplikasi merupakan perangkat lunak proses data yang berpacu pada sebuah komputasi. Aplikasi berasal dari bahasa Inggris application yang berarti penerapan, lamaran ataupun penggunaan. Sedangkan secara istilah, pengertian aplikasi adalah suatu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju. (Parjito, 2022)

E. Android



Gambar 2. 1 Logo Android

Android merupakan sistem operasi yang banyak digunakan pada perangkat bergerak yang dewasa ini sangat terkenal dan popular digunakan pada ponsel cerdas. Android juga merupakan platform pemrograman yang dikembangkan oleh Google untuk ponsel cerds dan perangkat selular lainnya, misalnya tablet. Android bisa berjalan di beberapa macam perangkat yang dikembangkan oleh banyak vendor ponsel cerdas yang berbeda. Android menyertakan paket pengembang perangkat lunak untuk penulisan kode asli. (Nur Azis, 2020)

F. Android Studio



Gambar 2. 2 Logo Android Studio

Android Studio merupakan sebuah Integrated Development Environment (IDE) khusus untuk membangun aplikasi yang berjalan pada platform android. Android studio ini berbasis pada IntelliJ IDEA, sebuah IDE untuk Bahasa pemrograman Java. Bahasa pemrograman utama yang digunakan adalah Java,

sedangkan untuk membuat tampilan atau layout, digunakan bahasa XML. Android studio juga terintegrasi dengan Android Software Development Kit (SDK) untuk deploy ke perangkat android. Android Studio juga merupakan pengembangan dari eclipse, dikembangkan menjadi lebih kompleks dan professional yang telah tersedia didalamnya Android Studio IDE, Android SDK tools. Setiap proyek di Android Studio berisi satu atau beberapa modul dengan file kode sumber dan file sumber daya. Jenis-jenis modul mencakup:

a. Modul aplikasi Android

b. Modul Pustaka

c. Modul Google App Engine. (Sondang Sibuea, 2022)

G. Java



Gambar 2. 3 Logo Java

Java merupakan bahasa pemrograman tingkat tinggi yang berorientasi pada object dan program java tersusun dari bagian yang disebut dengan Class. Class terdiri dari metode-metode yang melakukan pekerjaan dan mengembalikan informasi setelah melakukan tugas-nya. Para programmer Java banyak mengambil keuntungan dari kumpulan class di pustaka class Java yang disebut dengan Java *Application Programming Cross Interface* (API). Class-class ini diorganisasikan menjadi sekelompok yang disebut dengan paket (package). Java API telah

menciptakan applet dan aplikasi canggih dengan menyediakan fungsionalitas yang memadai. (Hartiwi, 2022).

H. MySQL



Gambar 2. 4 Logo MySQL

MySQL adalah perangkat lunak sistem manajemen basis data *SQL* (*database management system*) atau *DBMS*, yang bersifat *multi-threaded* dan *multi-user. MySQL* adalah sistem manajemen basis data *relasional (RDBMS)* didistribusikan secara gratis di bawah *GPL* (Lisensi Publik Umum). Setiap orang bebas menggunakan *MySQL,* tatapi tidak dapat digunakan sebagai sumber tertutup atau peoduk turunan komersial. *MySqL* sebenarnya merupakan turunan dari *SQL (Structured Query Language),* salah satu konsep utama dalam *database* lama.

SQL merupakan konsep operasi basis data, khususnya konsep operasi basis data yang digunakan untuk memilih dan memasukkan data, sehingga operasi data dapat diselesaikan dengan mudah dan otomatis. Keunggulan sistem database (*DBMS)* dapat dilihat dari cara pengoptimal memproses perintah *SQL* yang dibuat oleh pengguna dan aplikasinya. Sebagai *server database, MySQL* dapat dikatakan lebih unggul dibandingkan *database server* lainnya dalam *query* data. Hal ini terbukti untuk *query* yang dilaukan oleh *single user,* kecepatan *MySQL* bisa sepuluh

kali lebih cepat dari *PostgreSQL* dan lima kali lebih cepat dibandingkan *Interbase.* (Herawan, 2021)

I. XAMPP



Gambar 2. 5 Logo *Xampp*

xampp sebuah layanan web server yang mudah digunakan serta dapat menyediakan tampilan dinamis untuk mengakses secara lokal menggunakan web server local (localhost) XAMPP adalah singkatan dari *X-platform, Apache, MySQL, PHP, dan Perl.* perangkat lunak server web sumber terbuka (gratis) yang bekerja dengan berbagai sistem operasi, termasuk *Windows, Linux, dan Mac OS. XAMPP*, sering dikenal sebagai localhost, digunakan sebagai server yang berdiri sendiri. Ini memfasilitasi proses pengeditan, perancangan, dan pengembangan aplikasi. Untuk dapat membuat tampilan software atau website dengan lebih cepat, mudah, dan sistematis maka harus digunakan XAMPP. Program ini terdiri dari tiga bagian utama*: PhpMyAdmin, Control Panel, dan htdocs* (Ramadhan & Nurhad, 2024)

J. Visual Studio Code



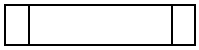
Gambar 2. 6 Logo *StudioCode*

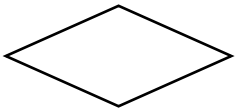
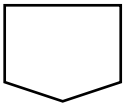
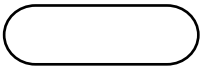
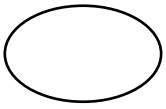
Microsoft menciptakan *Visual Studio Code*, editor kode sumber terbuka dan gratis. Untuk menghasilkan kode komputer, diperlukan aplikasi yang kompleks. *Visual Studio Code* dapat digunakan dalam situasi ini. Visual Studio Code adalah Software yang sangat ringan, namun kuat editor kode sumbernya yang berjalan dari desktop. Muncul dengan built-in dukungan untuk JavaScript, naskah dan Nodejs dan memiliki array beragam ekstensi yang tersedia untuk bahasa lain, termasuk C ++, C # , Python, dan PHP (Herwa, Hendrayudi, & Hartati, 2024)

K. Flowchart

Flowchart adalah sekumpulan gambar-gambar tertentu untuk menyatakan alur dari suatu program yang akan diterjemahkan ke salah satu bahasa pemrograman. Kegunaan *flowchart* sama seperti halnya *algoritma* yaitu untuk menuliskan alur program tetapi dalam bentuk gambar atau simbol (Husain, 2021). Symbol yang digunakan sebagai berikut:

Tabel 2. 1 *Symbol* dan Fungsi *Flowchart*

Simbol	Fungsi
	Permulaan sub program

Simbol	Fungsi
	Perbandingan,pernyataan, penyeleksian data yang memberikan pilihanuntuk langkahselanjutnya
	Penghubung bagian- bagian flowchart yang berada pada halaman berbeda
	Permulaan/akhir program
	Arah aliran program
	Proses inisialisasi/pemberian harga awal
	Proses penghitung/ proses pengolahan data
	Proses input/output data
	Penghubung bagian- bagian flowchart yang berada pada satu halaman.

L. Blackbox Testing

Blackbox Testing adalah sebuah metode yang dipakai untuk menguji sebuah software tanpa harus memperhatikan detail *software*. Proses *Blackbox Testing* dengan cara mencoba program yang telah dibuat dengan mencoba memasukkan data pada setiap formnya. Pengujian ini diperlukan untuk mengetahui program

tersebut berjalan sesuai dengan yang dibutuhkan oleh perusahaan. (Ni Made Dwi Febriyanti, 2021).

Pengujian dilakukan terhadap 30 (tiga puluh) fitur sistem. Hasil pengujian *blackbox* testing pada beberapa fitur ditunjukkan pada tabel Berdasarkan tabel *blackbox* testing menunjukkan hasil sukses pada semua fitur yang diuji.

Tabel 2. 2 Contoh hasil pengujian *Black box Testing*

NO	Fitur yang di uji	Skenario	Hasil yang diharapkan	Hasil pengujian
1	Login	Admin memasukka *username* dan *Password* yang salah	*Login* dan sistem tetap berada pada halaman *login*	Sukses
2	Melihat *list* berita	Admin klik menu berita, klik *list* berita	Admin berhasil melihat *list* bertita	Sukses
3	Mengedit dan *update* pengguna (admin)	Admin klik menu pengguna, klik *icon edit* pada pengguna yang akan di*edit*, isi from *edit*, klik *update*	Admin berhasil *menegdit* dan *update* pengguna	Sukses
4	Menambah foto	Admin klik menu galeri, klik *photos*, klik *add photo*, isi *form photo*, klik simpan	Admin berhasil menambahkan foto	Sukses
5	*Log out*	Admin klik *log out*	Admin berhasil *log out* dan kembali ke halaman *login*	Sukses
6	Melihat halam *profile*	Admin masuk ke halaman beranda *website*, klik menu tentang sekolah, klik *profile*	Admin berhasil masuk ke halaman *profile*	Sukses
7	Membalas komentar	Admin klik menu komentar, klik *icon reply* pada komentar yang ingin di bahas	Admin berhasil membalas komentar	Sukses

NO	Fitur yang di uji	Skenario	Hasil yang diharapkan	Hasil pengujian
8	Mengirim pesan	Admin masuk ke halaman beranda *website*, klik menu kontak, isi *form* tinggalkan pesan, klik *submit*	Admin dapat mengirim pesan	Sukses
9	Menghapus album	Admin klik menu galeri, klik album, klik *icon* sampah pada album yang ingin dihpus	Admin berhasil menghapus album	Sukses
10	Melihat *inbox*	Admin klik menu *inbox*	Admin berhasil melihat halaman *list inbox*	Sukses

M. Whitebox Testing

Whitebox Testing merupakan pengujian yang dikembangkan berdasarkan pada kode program. Penguji dalam *whitebox testing* harus memiliki pengetahuan tentang kode dan penulisan kasus uji dengan parameter yang sesuai. Hal ini terutama menyangkut dengan aliran kontrol dan aliran data suatu program, Pada pengujian *whitebox* terdapat beberapa teknik yang dapat di lakukan diantaranya: Data *Flow Testing, Control Flow Testing, Basic Path atau Path Testing, dan Loop Testing*. (Shiddiq, 2021)

Whitebox pengujian yang didasarkan pada detail prosedur dan alur logika kode program. Pada kegiatan *Whitebox testing, tester* melihat *souce code* program dan menemukan *bugs* dari kode program yang diuji. *Whitebox testing* adalah pengujian yang dilakukan sampai detail pengecekan kode program. (Nurhidayani S,2023).

N. UML (*Unified Modelling Languange*)

Unified Modelling Languange (UML) adalah sebuah “bahasa” yang telah

menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. *UML* menawarkan sebuah standar untuk merancang model sebuah sistem. *UML* adalah sekumpulan alat yang digunakan untuk melakukan abstraksi terhadap sebuah sistem atau perangkat lunak berbasis objek. *UML* merupakan singkatan dari *Unified Modeling Language.* *UML* juga menjadi salah satu cara untuk mempermudah penegmbangan aplikai yang berkelanjutan (Yunengsi, 2023).

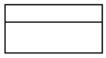
Adapun daftar simbol *UML* yaitu:

Tabel 2. 3 *Symbol UseCaseDiagram*

No	GAMBAR	NAMA	KETERANGAN
1		*Actor*	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan *Use Case*.
2		*Dependency*	Hubungan di mana perubahan yang terjadi pada suatu elemen mandiri (*Independent*) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
3		*Generalization*	Hubungan di mana objek anak *(Descendent*) berbagai perilaku dan struktur data dari objek yang ada di atasnya objek induk (*Ancestor*).
4		*Include*	Menypesifikasikan bahwa Use Case sumber secara Eksplisit.
5		*Extend*	Menypesifikasikan bahwa *use case* target memperluas perilaku dari *use case* sumber pada suatu titik yang diberikan.

No	GAMBAR	NAMA	KETERANGAN
6		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		*System*	Menypesifikasikan paket yang menampilkan sistem secara terbatas.
8		*Use Case*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu *actor*
9		*Collaboration*	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		*Note*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Tabel 2. 4 *Symbol Class Diagram*

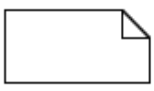
No.	GAMBAR	NAMA	KETERANGAN
1		*Generalization*	Hubungan di mana objek anak *(descendent)* berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
2		*Nary Association*	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		*Class*	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		*Collaboration*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu *actor*
5		*Realization*	Operasi yang benar-benar dilakukan oleh suatu objek.
6		*Dependency*	Hubungan di mana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempengaruhi

No.	GAMBAR	NAMA	KETERANGAN
			elemen yang bergantung padanya elemen yang tidak mandiri
7		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya

Tabel 2. 5 *Symbol Sequence* Diagram

No	GAMBAR	NAMA	KETERANGAN
1		*LifeLine*	Objek *entity*, antarmuka yang saling berinteraksi.
2		*Message*	Spesifikasi dari komunikasi antar objek yang memuat in*form*asi-in*form*asi tentang aktivitas yang terjadi
3		*Message*	Spesifikasi dari komunikasi antar objek yang memuat in*form*asi-in*form*asi tentang aktivitas yang terjadi

Tabel 2. 6 *Symbol State Chart Diagram*

No	GAMBAR	NAMA	KETERANGAN
1		*State*	Nilai atribut dan nilai Link pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		*Initial Pseudo State*	Bagaimana objek dibentuk atau diawali
3		*Final State*	Bagaimana objek dibentuk dan dihancurkan
4		*Transition*	Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya
5		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya.
6		*Node*	Elemnefisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Tabel 2. 7 *Symbol Activity Diagram*

No.	GAMBAR	NAMA	KETERANGAN
1		*Actifity*	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain

No.	GAMBAR	NAMA	KETERANGAN
2		*Action*	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		*Initial Node*	Bagaimana objek dibentuk atau diawali.
4		*Actifity Final Node*	Bagaimana objek dibentuk dan dihancurkan
5		*Fork Node*	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

O. Kerangka berfikir

Untuk Memahami alur peneltian diatas, diuraikan ke dalam kerangka berpikir yang akan disajikan dalam bentuk diagram ini:

Program BLT-DD bertujuan meningkatkan kesejahteraan masyarakat desa, namun menghadapi masalah pengelolaan dana, termasuk risiko penyalahgunaan dan keterbatasan akses informasi real-time.

↓

Identifikasi risiko penyalahgunaan dana, kehilangan, dan keterbatasan akses informasi dalam pengelolaan BLT-DD.

↓

Mengembangkan aplikasi sistem informasi keuangan berbasis Android untuk meningkatkan efisiensi pengelolaan dana, memastikan pengiriman tunai yang tepat waktu, dan memfasilitasi akses informasi real-time bagi masyarakat.

↓

pengumpulan data melalui wawancara, observasi, dan analisis dokumen. Perancangan sistem melibatkan analisis kebutuhan, perancangan arsitektur, dan pengembangan aplikasi. Pengujian dilakukan untuk mengevaluasi fungsionalitas dan usability aplikasi.

↓

Pengembangan aplikasi sistem informasi keuangan berbasis Android yang dapat meningkatkan efisiensi pengelolaan dana BLT-DD, memastikan pengiriman tunai tepat waktu, dan meningkatkan akses informasi real-time bagi masyarakat, sehingga mengurangi risiko penyalahgunaan dan kehilangan dana.

BAB III

METODE PENELITIAN

A. Jenis Penelitian

Jenis penelitian yang di gunakan adalah jenis penelitian kualitatif dimana metode penelitian ini digunakan untuk menganalisis kebutuhan pengguna, merancang dan mengembangkan aplikasi.

B. Waktu Dan Tempat Penelitian

Pelaksanaan waktu penelitian ini berlangsung selama (±) 2 bulan, Tempat pelaksanaan penelitian ini di desa Botto Mallangga Kecamatan Maiwa Kabupaten Enrekang.

Tabel 3. 1 Jadwal Pelaksaan Penelitian

No	Uraian penelitian	Tahun 2024		
		Oktober	Nonember	Desember
1.	Pengumpulan Data			
2.	Analisis Sistem			
3.	Desain sistem			
4.	Implementasi			
5.	Pengujin sistem			

C. Alat Dan Bahan

Adapun alat dan bahan yang akan dibutuhkan selama proses penelitian ini adalah sebagai berikut:

1. Alat yang digunakan selama proses penelitian

Laptop ACER-R5D17LVF dengan spesifikasi:

Tabel 3. 2 Spesifikasi Laptop

No.	Nama	Spesifikasi
1	Prosessor	AMD 3020e with Radeon Graphics
2	RAM	4,00 GB (3,44 GB usable)
3	Hardisk	500GB
4	Monitor	12 Inci

2. Perangkat Lunak (*Software)*

Tabel 3. 3 *Software* yang digunakan

No.	Nama	spesifikasi
1	Sistem Operasi	*Windows*11 Home Single Language
2	*Code Editor*	*Android Studio*
3	Bahasa Pemrograman	*Java*
4	*Data Base*	*MySQL*

D. Teknik Pengumpulan Data

Pengumpulan data dilakukan untuk mengumpulkan seluruh informasi yang terkait dan mendukung pelaksanaan penelitian ini:

1. Observasi

Metode pengumpulan data yang digunakan oleh peneliti dalam pembuatan skripsi ini adalah dengan cara melakukan pengamatan atau peninjauan secara langsung terhadap objek dengan tujuan untuk mendapatkan informasi tentang BLT-DD yang diperlukan oleh peneliti dalam penelitian ini.

2. Wawancara

   Untuk mendapat informasi yang lebih akurat dalam melakukan penelitian ini, peneliti melakukan wawancara atau tanya jawab secara langsung.

3. Studi kepustakaan

   Studi pustaka dilakukan dengan dengan mencari sumber-sumber pustaka yang diperoleh dari internet atau perpustakaan berupa buku jurnal atau karya ilmiah yang berkaitan dengan penelitian.

E. Tahapan Penelitian

Tahapan penelitian yang dimakasud dalam penelitian ini adalah berkenaan dengan proses pelaksanaan penelitian.

1. Pengumpulan data

   Mengumpulkan data-data dan informasi untuk dijadikan acuan dalam membangun aplikasi yang dirancang.

2. Analis sistem

   Menegindetifikasi dan mengenali masalah yang ada, kemudian mencari alternatif-alternatif pemecahnya.

3. Desain sistem

Setelah masalah ditentukan dan analisa data sudah dilakukan maka perlu dilakukan pembangunan atau mendesain sistem tentang masalah yang sudah ditentukan.

4. Implementasi

   Membangun sistem sesui dengan desain yang telah dibuat.

5. Pengujian sistem

   Menguji sistem untuk memastikan kinerjanya sesuai dengan spesifikasi yang ditetapkan.

F. Metode Pengujian

Dalam penelitian ini, digunakan 2 (dua) metode dalam pengujian datanya yaitu *blackbox testing* dan *whitebox testing*:

1. *Blackbox testing*

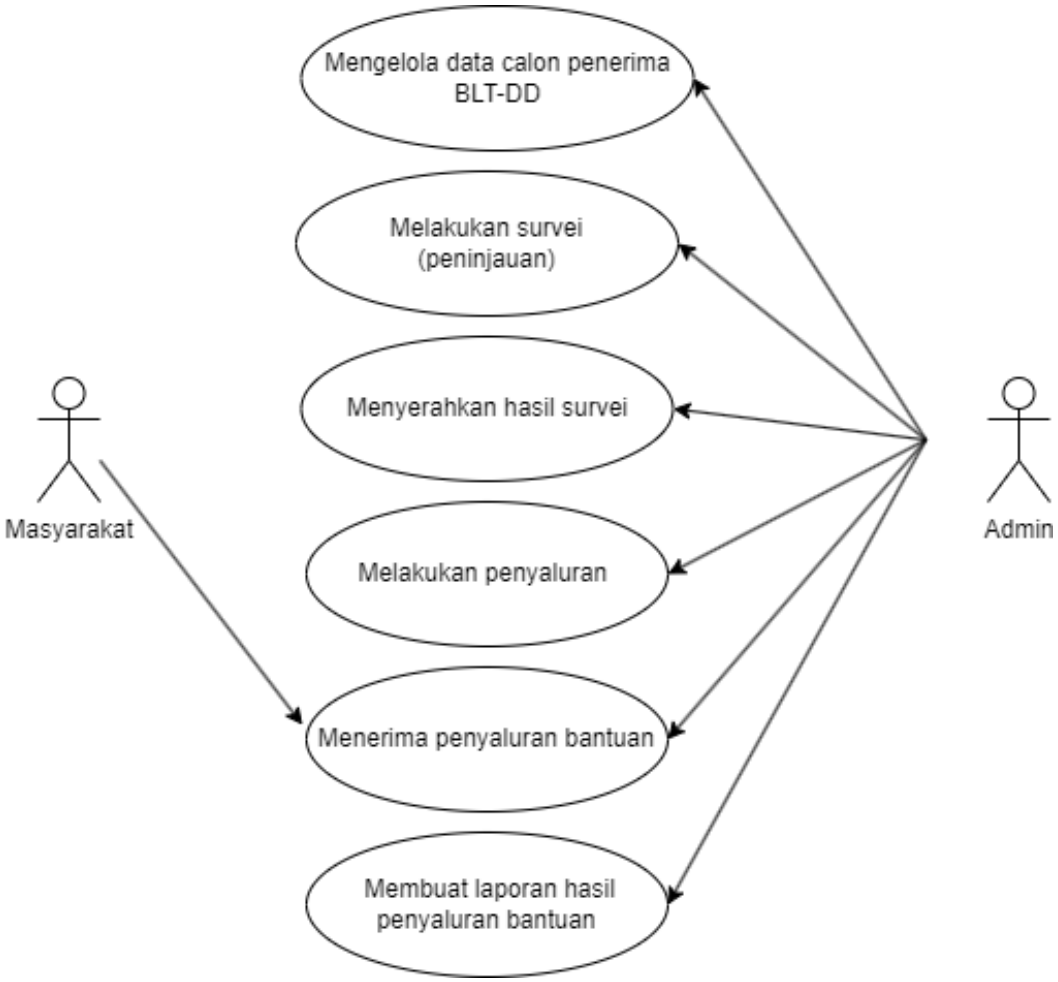
Blackbox testing digunakan untuk menguji program yang sedang berjalan apakah aplikasi ini sudah berhasil atau tidak.

2. *Whitebox testing*

Whitebox testing digunakan untuk mengetahui struktur dan alur pada aplikasi ini apakah sudah sesuai dengan ketentuan kemudian menguji aplikasi tersebut untuk memperoleh hasil.

G. Desain Sistem

1. Sistem yang berjalan

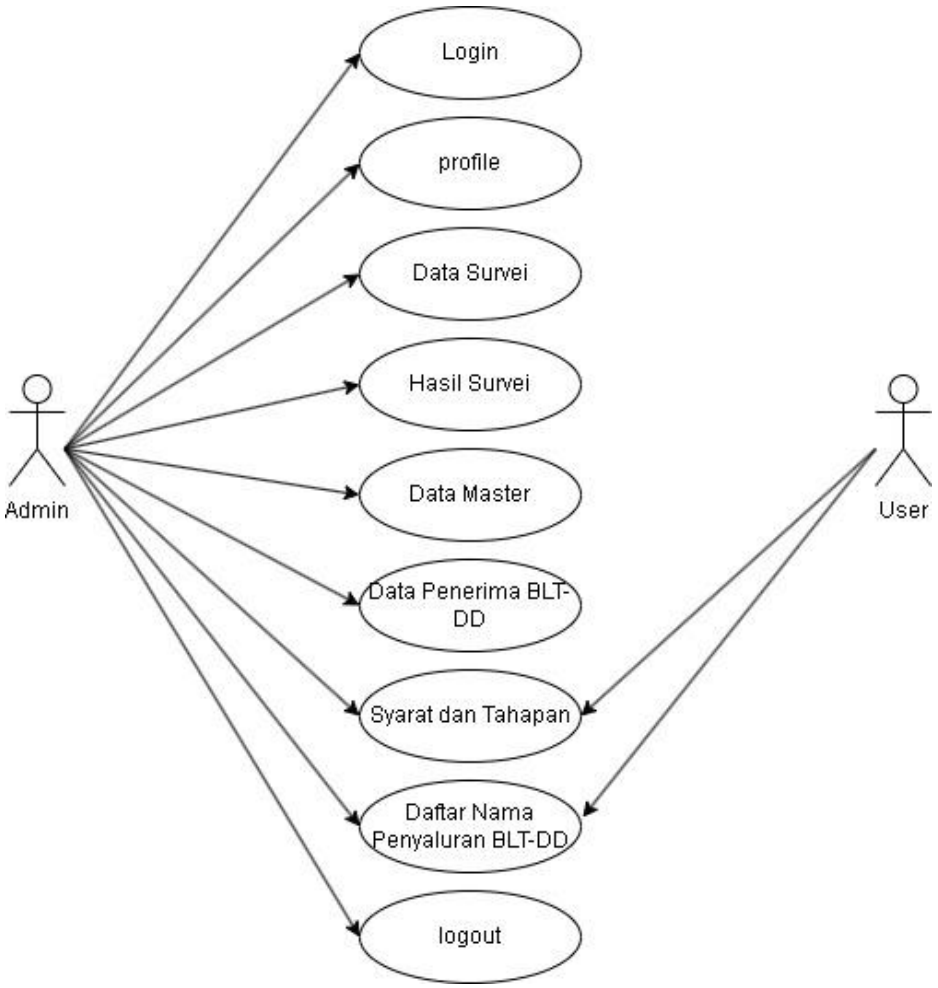


Gambar 3. 1 *Use Case* Sistem yang Sedang Berjalan

Prosdur penyaluran Bantuan Langsung Tunai Dana Desa (BLT-DD) dimulai dengan mengelola data calon penerima yang dikumpulkan dari pendataan desa. Data ini mencakup informasi seperti Nama, Alamat, Dan Kondisi Ekonomi. Selanjutnya, tim survei melakukan peninjauan lapangan untuk memverifikasi data dan kondisi calon penerima secara langsung. Hasil survei diserahkan dan disahkan oleh pihak berwenang melalui rapat desa.

Setelah daftar penerima bantuan disahkan, penyaluran dilakukan secara langsung. Kemudian Penerima bantuan harus menandatangani bukti penerimaan dan menunjukkan identitas. Tahap terakhir adalah membuat laporan hasil penyaluran yang mencakup jumlah dana, dan jumlah penerima.

2. Sistem yang Usulkan



Gambar 3. 2 *Use Case* Sistem yang diusulkan

Prosedur penyaluran BLT-DD yang diusulkan:

Tabel 3. 4 Penjelasan Desain Sistem yang Diusulkan untuk Admin

Nama Use Case	Deskripsi
Login	*Use Case* ini menjelaskan tentang *Admin* memasukkan kredensial (*username* dan *password*) untuk mengakses sistem. Jika berhasil, *Admin* diarahkan ke dashboard utama. Jika gagal, muncul pesan *username* tidak terdaftar.
Profile	*Use Case* ini menjelaskan tentang Setelah *login, Admin* dapat mengakses menu profile untuk melihat dan memperbarui informasi pribadinya, seperti Nama, *Username*, Password, dan *Role*.
Data Survei	*Use Case* ini menjelaskan tentang Admin dapat melakukan input dan pengelolaan data survei.data ini berisi informasi yang digunakan untuk menentukan penerima BLT-DD, seperti kondisi ekonomi, pekerjaan, dan jumlah tanggungan.
Hasil Survei	*Use Case* ini menjelaskan tentang Admin dapat melihat hasil survei yang telah dilakukan, Hasil survei ini digunakan untuk menganalisis dan menentukan kelayakan calon penerima BLT-DD.
Data Master	*Use Case* ini menjelaskan tentang Admin dapat mengelola data utama dalam sistem.
Data Penerima BLT	*Use Case* ini menjelaskan tentang Admin dapat mengakses daftar penerima BLT-DD yang telah lolos seleksi.
Syarat dan Tahapan	*Use Case* ini Menjelaskan tentang admin dapat mengelola informasi mengenai persyaratan dan tahapan penyaluran BLT-DD.
Daftar Nama Penyaluran BLT-DD	*Use case* ini menjelaskan tentang admin dapat melihat daftar penerima BLT-DD berdasarkan hasil verifikasi.
	Use Case ini menjelaskan tentang Setelah selesai menggunakan sistem,

Nama Use Case	Deskripsi
Logout	*Admin* dapat keluar dengan memilih opsi *Logout*.sistem akan mengakhiri sesi dan mengembalikan *Admin* ke halaman *login.*

Tabel 3. 5 Penjelasan Desain Sistem yang Diusulkan untuk User

Nama Use Case	**Deskripsi**
Syarat dan Tahapan	*Use case* ini menjelaskan tentang User dapat melihat informasi mengenai persyaratan serta langkah-langkah yang harus di penuhi oleh calon penerima BLT-DD
Daftar Nama Penerima BLT-DD	*Use case* ini menjelaskan tentang User dapat mengakses daftar nama penerima bantuan yang telah disetujui dan diproses

BAB IV

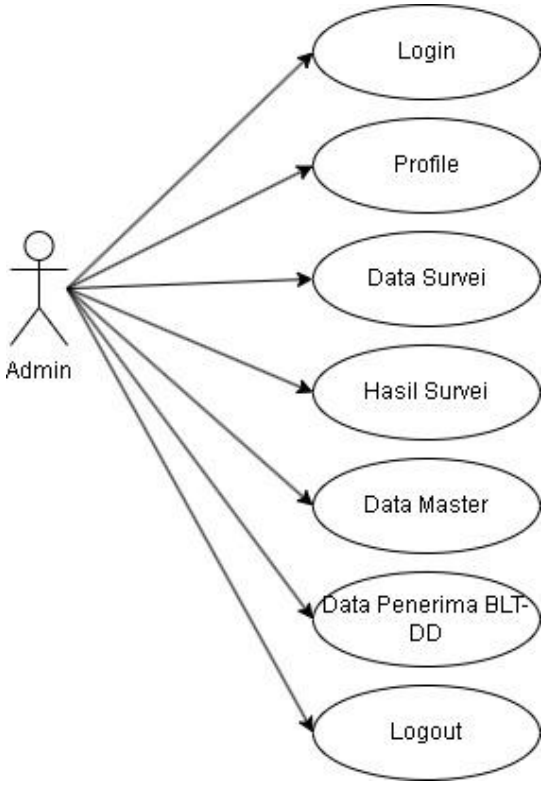
HASIL PENELITIAN DAN PEMBAHASAN

A. Analisis Aliran Data Diagram UML

Dengan tujuan memberikan informasi tahapan dan alur aplikasi, *Use Case Diagram, Activity Diagram,* dan *Sequance* Diagram digunakan untuk mengorientasikan objek sistem melalui analisis aliran data.

1. Use Case Diagram Admin

Use Case Diagram ditunjukan untuk menujukkan fungsi kerangka dari perspektif pengguna atau aktor yang berada di luar sistem.



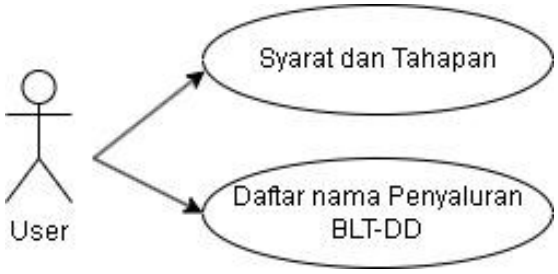
Gambar 4. 1 *Use case* Diagram Admin

Pada gambar 4.1 admin akan login, untuk mengelola profile, serta data survei, hasil survei, data master dan data penerima. Admin juga dapat menginput data survei, mengelola data master, dan melihat daftar penerima bantuan. Setelah selesai, admin dapat logout dari sistem, dan masyarat tidak usah login, cukup melihat syarat dan tahapan serta masyarakat yang lolos menerima BLT-DD.

Tabel 4. 1 Penjelasan Desain Sistem yang diusulkan untuk Admin

Nama *Use Case*	**Deskripsi**
Login	*Admin* memasukkan kredensial (*username* dan *password*) untuk mengakses sistem. Jika berhasil, *Admin* diarahkan ke dashboard utama. Jika gagal, muncul pesan *username* tidak terdaftar.
profile	Setelah *login*, *Admin* dapat mengakses menu profile untuk melihat dan memperbarui informasi pribadinya, seperti Nama, *Username*, Password, dan *Role*.
Data Survei	Admin dapat melakukan input dan pengelolaan data survei.data ini berisi informasi yang digunakan untuk menentukan penerima BLT-DD, seperti kondisi ekonomi, pekerjaan, dan jumlah tanggungan.
Hasil Survei	Admin dapat melihat hasil survei yang telah dilakukan, Hasil survei ini digunakan untuk menganalisis dan menentukan kelayakan calon penerima BLT-DD.
Data Master	Admin dapat mengelola data utama dalam sistem.
Data Penerima BLT-DD	Admin dapat mengakses daftar penerima BLT-DD yang telah lolos seleksi.
logout	Setelah selesai menggunakan sistem, *Admin* dapat keluar dengan memilih opsi *Logout*.sistem akan mengakhiri sesi dan mengembalikan *Admin* ke halaman *login.*

2. Use Case Diagram User



Gambar 4. 2 *Use case* Diagram User

Tabel 4. 2 Penjelasan Desain Sistem yang diusulkan untuk User

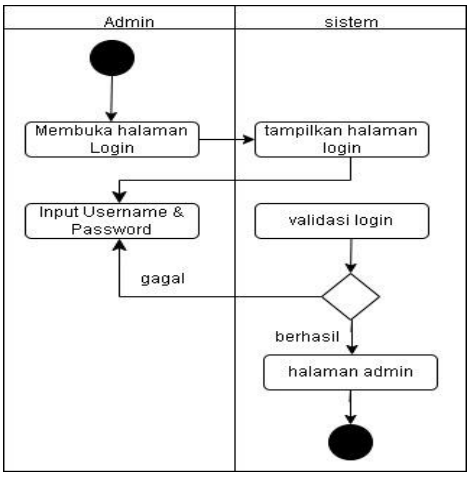
Nama Use Case	Deskripsi
Syarat dan Tahapan	Fitur ini menyediakan informasi mengenai persyaratan serta langkah-langkah yang harus dipenuhi oleh calon penerima untuk mendapatkan bantuan BLT-DD. Dengan adanya fitur ini, User dapat memahami prosedur pendaftaran.
Daftar Nama Penyaluran BLT-DD	User dapat mengakses daftar nama penerima bantuan yang telah disetujui dan diproses. Fitur ini bertujuan untuk memberikan transparansi dalam penyaluran bantuan, sehingga masyarakat dapat mengetahui apakah mereka lolos menerima bantuan.

3. Activity Diagram

Activity diagram digunakan untuk memodelkan aliran kerja atau aliran kontrol dari sebuah proses bisnis atau use case. Diagram ini menujukkan aktivitas-aktivitas yang terjadi dalam sistem dan urutan eksekusinya.

1. *Activity Diagram Admin*

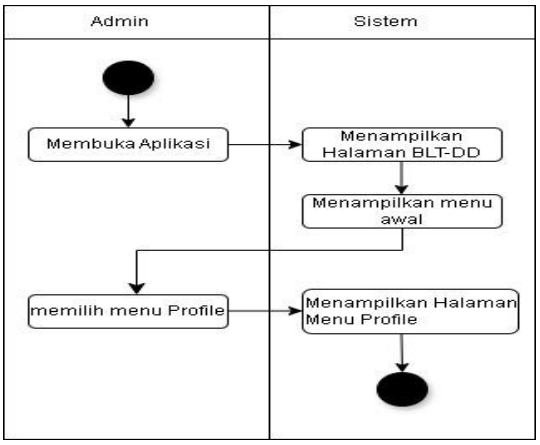
a. Activity diagram tampilan menu *Login*



Gambar 4. 3 *Activity* Diagram *Login*

Pada Gambar 4.3 menjelaskan cara masuk sebagai admin, admin harus terlebih dahulu mengakses tampilan *form login* dan di haruskan memasukkan kembali *username* dan *password* yang telah dibuat. Kemudian sistem akan mencecokkan jika cocok, akan berada di halaman *admin* apabila tidak, tetap di halaman *login*.

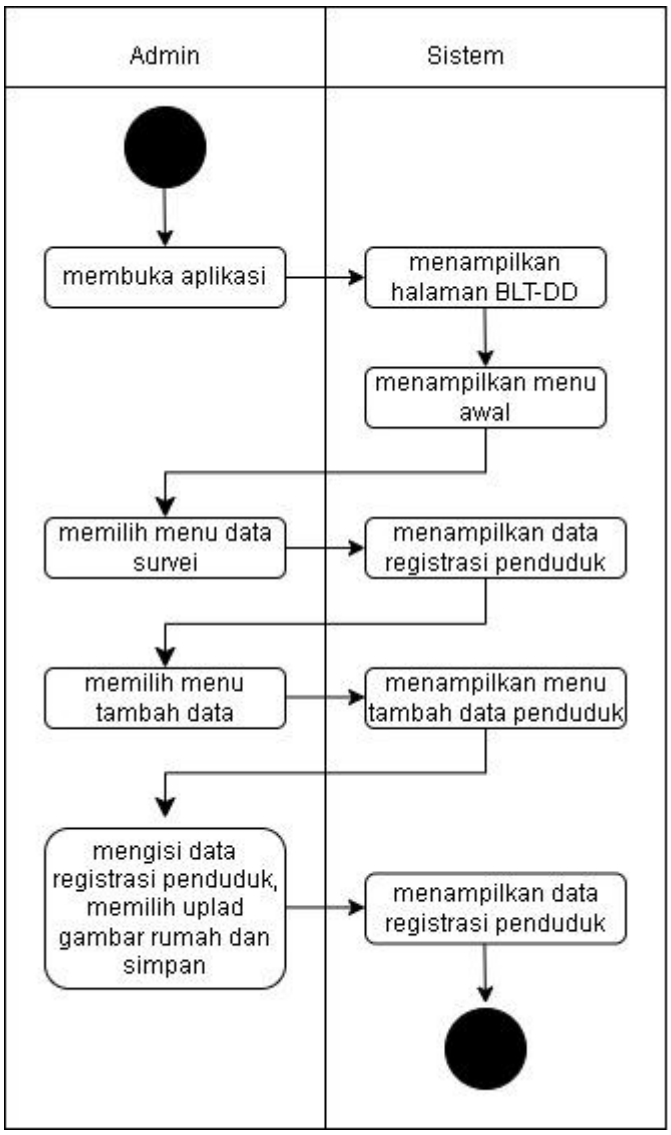
b. Activity diagram tampilan *profile*



Gambar 4. 4 *Activity* Diagram *Profile*

Pada Gambar 4.4 Menjelaskan tentang tampilan menu profile, dimana admin membuka aplikasi maka sistem akan menampilkan halaman BLT-DD dan menampilankan menu awal, setelah itu memilih menu profile kemudian sistem akan menampilkan menu profile.

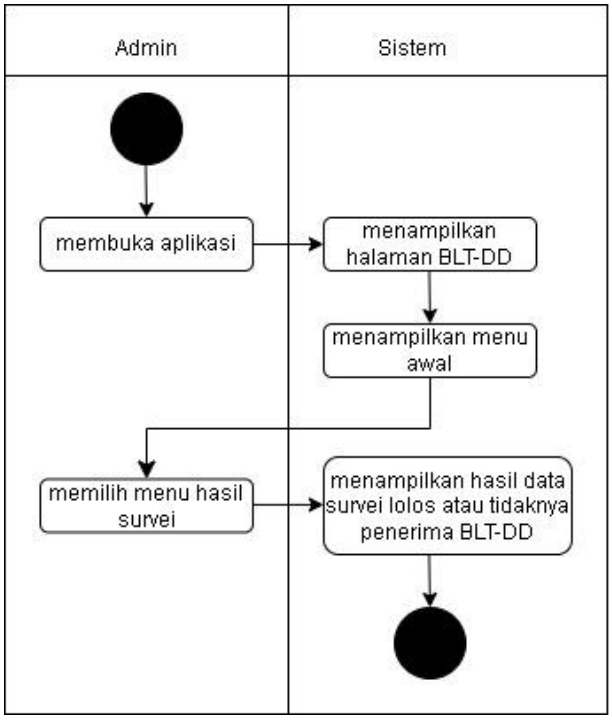
c. Activity diagram tampilan Data Survei



Gambar 4. 5 *Activity* Diagram Data Survei

Pada gambar 4.5 Menjelaskan tentang tanpilan menu data survei, admin dimulai dengan membuka aplikasi, yang kemudian menampilkan halaman awal BLT-DD serta menu utama. Setelah itu, admin memilih menu data survei, sehingga sistem menampilkan data registrasi penduduk yang telah tersedia. Jika admin ingin menambahkan data baru, ia memilih menu tambah data, dan sistem akan menampilkan menu tambah data untuk memasukkan data penduduk. Selanjutnya, admin mengisi data registrasi penduduk, termasuk mengunggah gambar rumah, lalu menyimpannya ke dalam sistem. Setelah data disimpan, sistem secara otomatis memperbarui dan menampilkan kembali data registrasi penduduk yang telah diperbarui.

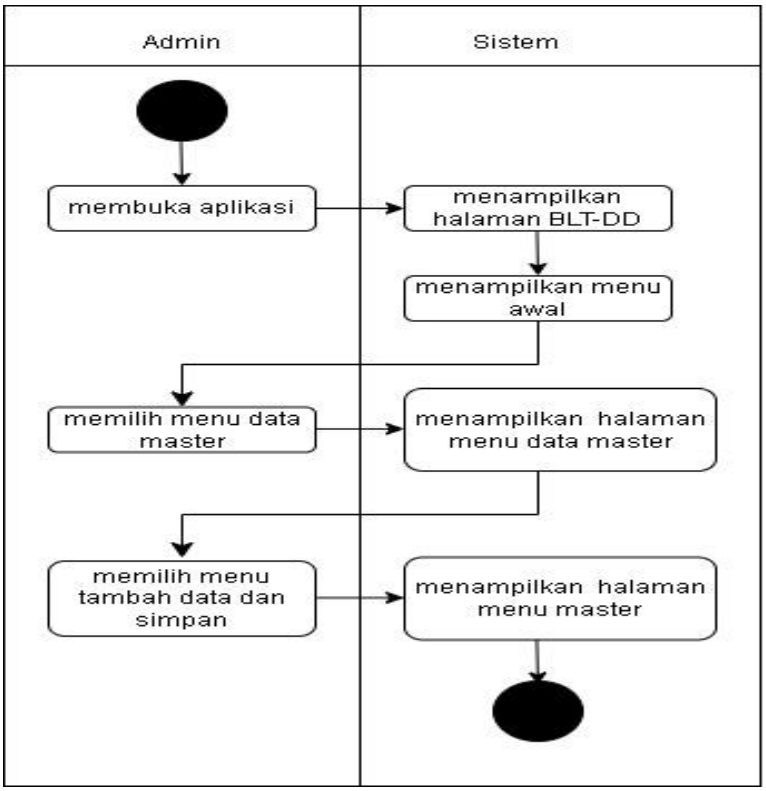
d. Activity diagram tampilan Hasil Survei



Gambar 4. 6 *Activity* Diagram Hasil Survei

Pada gambar 4.6 Menjelaskan tentang tanpilan menu data hasil survei admin membuka aplikasi, yang kemudian direspons oleh sistem dengan menampilkan halaman awal BLT-DD serta menu utama. Setelah itu, admin memilih menu hasil survei untuk melihat status penerima BLT-DD. Sistem kemudian menampilkan hasil data survei, yang menunjukkan apakah penduduk yang telah didaftarkan lolos atau tidak sebagai penerima bantuan.

e. *Activity Diagram* tampilan Data Master

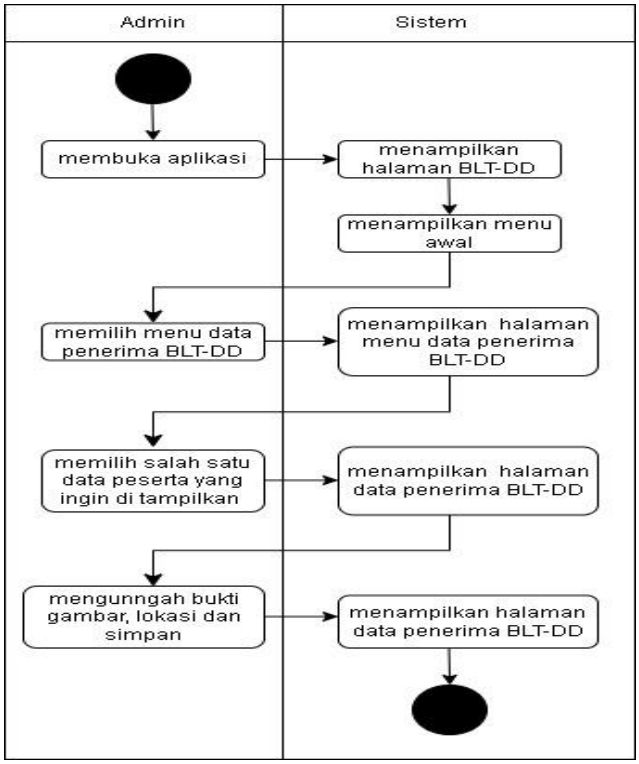


Gambar 4. 7 *Activity* Diagram Data Master

Pada gambar 4.7 Menjelaskan tentang tanpilan menu data master, Alur penggunaan aplikasi BLT-DD dalam diagram di atas dimulai dengan admin membuka aplikasi, yang kemudian

direspons oleh sistem dengan menampilkan halaman BLT-DD serta menu utama. Selanjutnya, admin memilih menu data master, sehingga sistem menampilkan halaman menu data master. Setelah itu, admin memilih menu tambah data dan menyimpannya, lalu sistem akan menampilkan kembali halaman menu master.

f. *Activity Diagram* tampilan Data Penerima BLT-DD



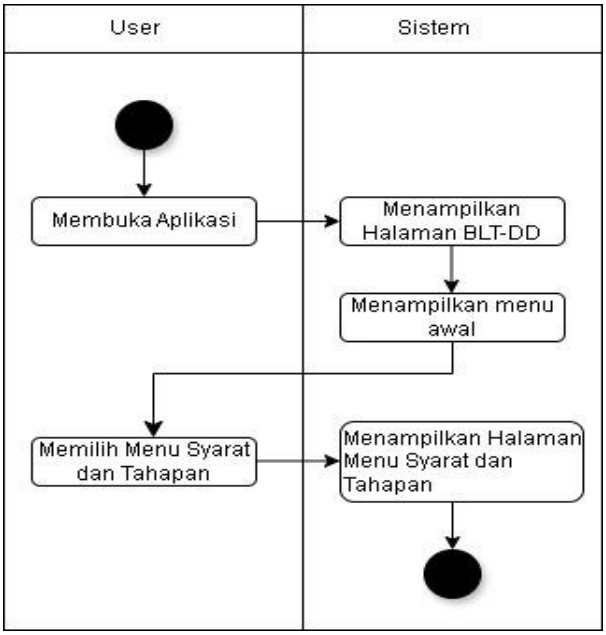
Gambar 4. 8 *Activity Diagram* Tampilan Data Penerima BLT-DD

Pada gambar 4.8 Menjelaskan tentang tampilan menu data penerima BLT-DD dalam diagram di atas dimulai dengan admin membuka aplikasi, yang kemudian direspons oleh sistem dengan menampilkan halaman BLT-DD serta menu utama. Selanjutnya, admin memilih menu data penerima BLT-DD, sehingga sistem

menampilkan halaman data penerima BLT-DD. Setelah itu, admin memilih salah satu data peserta yang ingin ditampilkan, dan sistem akan menampilkan detail informasi penerima BLT-DD tersebut. Kemudian, admin dapat mengunggah bukti gambar, lokasi, dan menyimpan data, yang selanjutnya akan diperbarui oleh sistem dan ditampilkan kembali dalam halaman data penerima BLT-DD. Proses ini memastikan bahwa data penerima BLT-DD terdokumentasi dengan bukti yang lengkap dan akurat.

2. Activity Diagram User

a. *Activity* diagram tampilan syarat dan tahapan

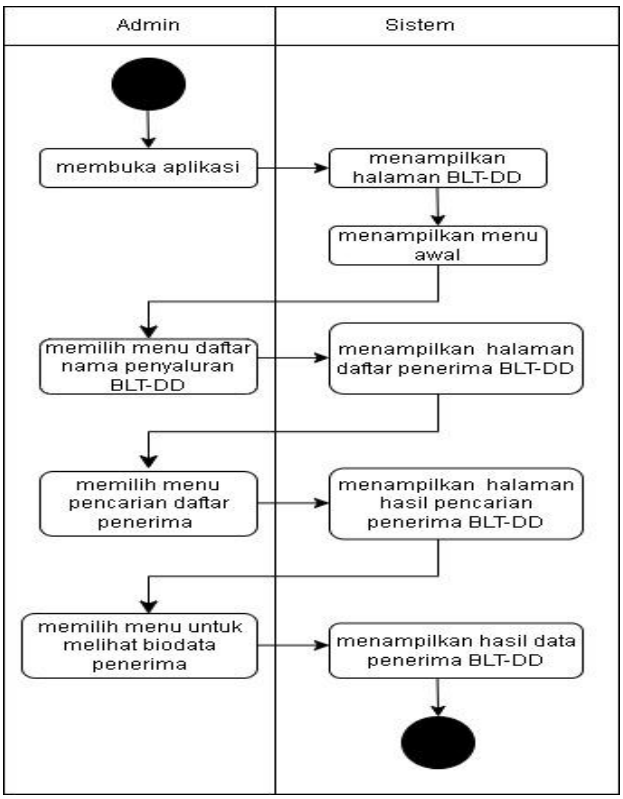


Gambar 4. 9 *Activity Diagram* Tampilan Syarat dan Tahapan

Pada gambar 4.9 Menjelaskan tentang tampilan syarat dan tahapan tampilan menu Data Survei, dimana admin membuka aplikasi maka sistem akan menampilkan halaman BLT-DD dan

menampilankan menu awal, setelah itu memilih menu Syarat dan Tahapan kemudian sistem akan menampilkan menu Syarat dan Tahapan.

b. Activity diagram tampilan Daftar nama penerima BLT-DD



Gambar 4. 10 *Activity Diagram* Daftar Nama Penerima

Pada gambar 4.10 Menjelaskan tentang tampilan daftar nama penerima Alur penggunaan aplikasi BLT-DD dalam diagram di atas dimulai dengan admin membuka aplikasi, yang kemudian direspons oleh sistem dengan menampilkan halaman BLT-DD serta menu utama. Selanjutnya, admin memilih menu daftar nama penyaluran BLT-DD, sehingga sistem menampilkan halaman daftar penerima BLT-DD. Setelah itu, admin memilih menu pencarian daftar penerima, dan sistem

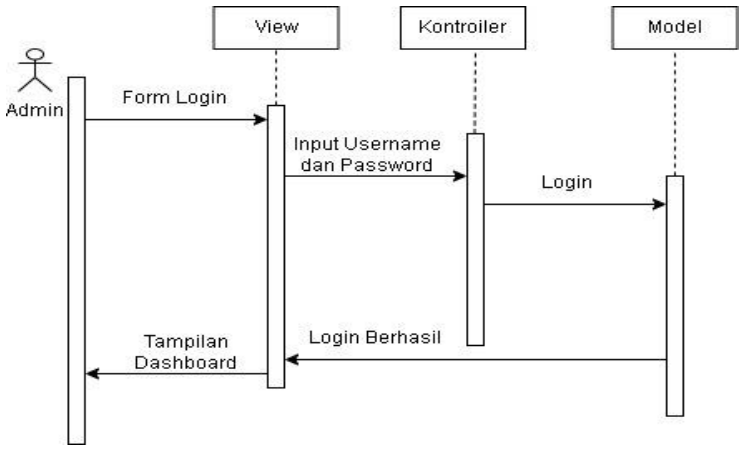
akan menampilkan halaman hasil pencarian penerima BLT-DD berdasarkan kriteria tertentu. Kemudian, admin memilih menu untuk melihat biodata penerima, dan sistem menampilkan hasil data penerima BLT-DD secara lengkap.

3. Sequence Diagram Admin

Diagram yang menvisualisasikan interaksi antara objek-objek secara berurutan dalam sistem seiring berjalannya waktu. Diagram ini memperlihatkan cara objek saling bertukar pesan atau data untuk menyelesaikan suatu tugas atau *use case* tertentu.

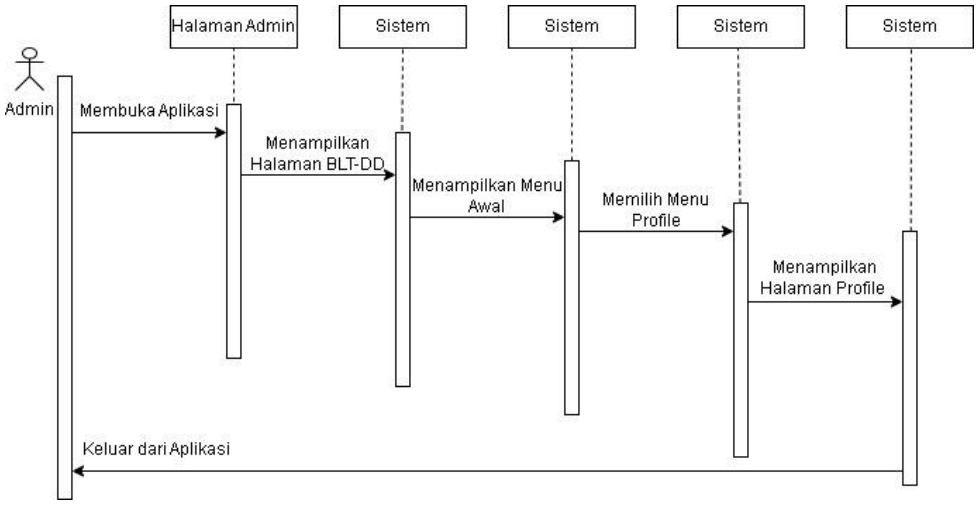
1. ***Sequence Diagram* Admin**

a. *Sequence Diagram* login



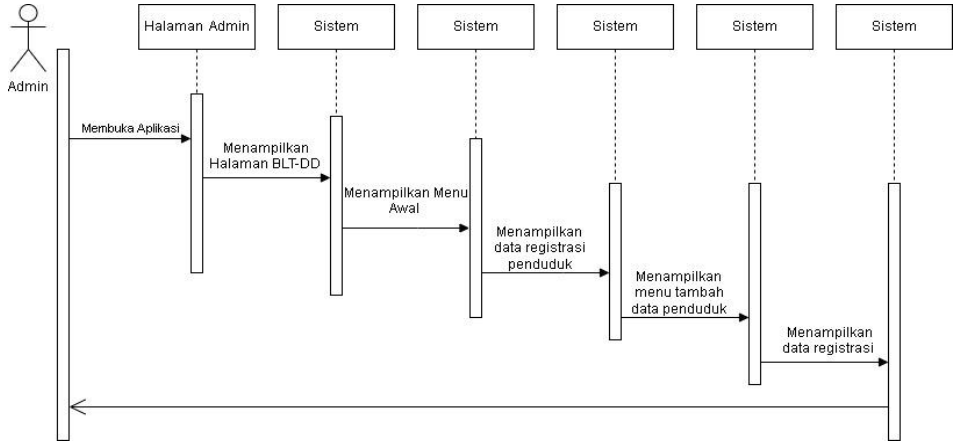
Gambar 4. 11 *Sequence Diagram* Login

b. *Sequence Diagram* profile



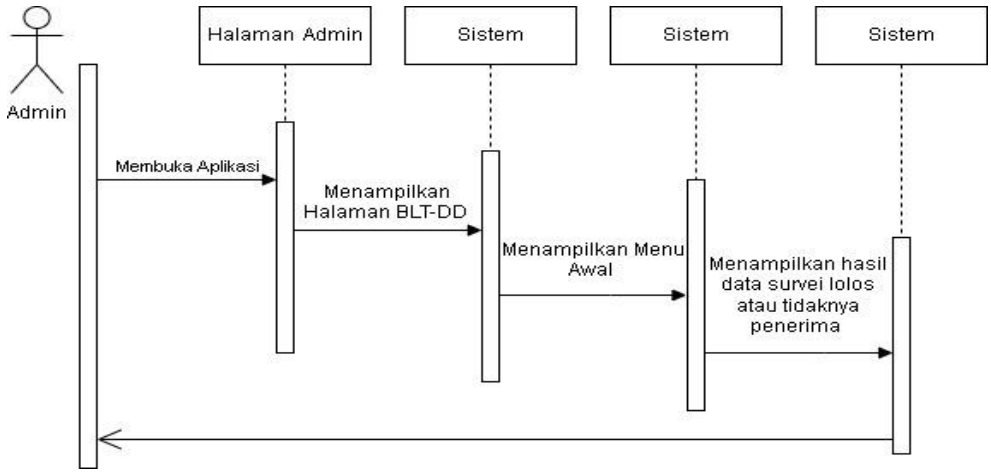
Gambar 4. 12 *Sequence Diagram* Profile

c. *Sequence Diagram* Data survei



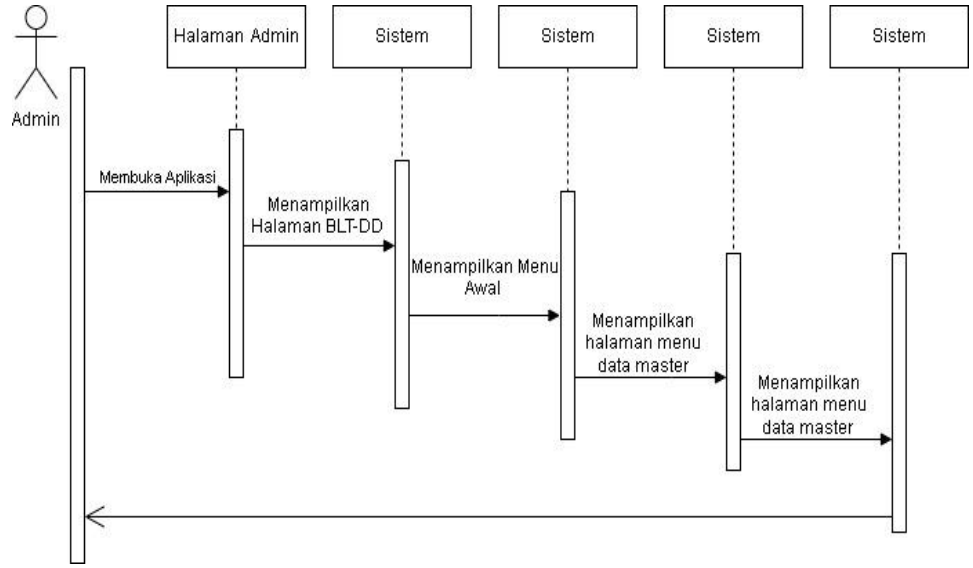
Gambar 4. 13 *Sequence Diagram* Survei

d. *Sequence Diagram* Hasil Survei



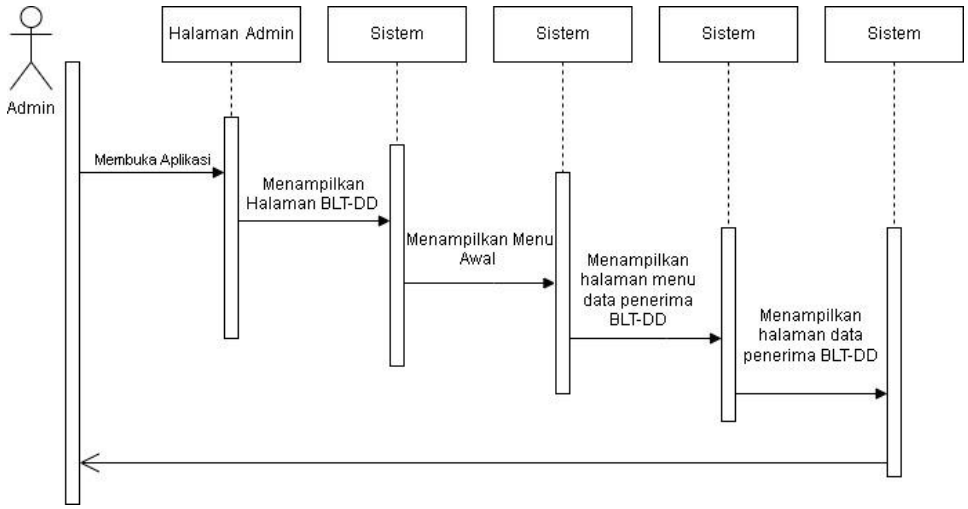
Gambar 4. 14 *Sequence Diagram* Hasil Survei

e. *Sequence Diagram* data master



Gambar 4. 15 *Sequence Diagram* Data Master

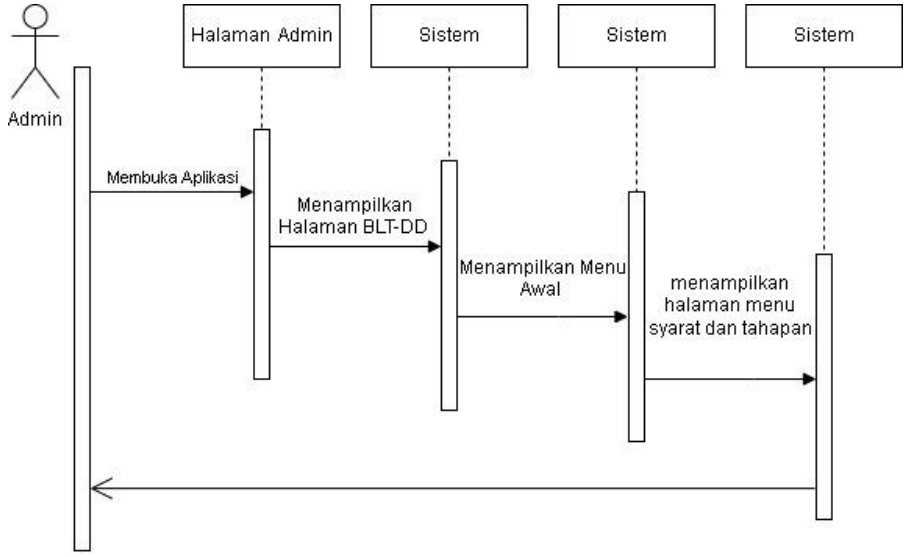
f. *Sequence Diagram* Data Penerima BLT-DD



Gambar 4. 16 *Sequence Diagram* Data Penerima BLT-DD

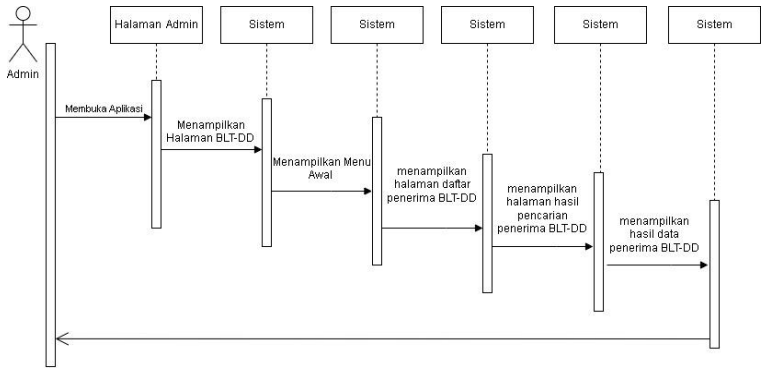
2. *Sequence Diagram* User

a. *Sequence Diagram* Syarat dan Tahapan



Gambar 4. 17 *Sequence Diagram* Syarat dan Tahapan

b. *Sequence diagram* daftar Nama Penerima BLT-DD



Gambar 4. 18 *Sequence Diagram* Daftar Nama Penerima BLT-DD

B. Rancangan Database

Berikut ini adalah tabel-tabel yang digunakan dalam aplikasi bantuan BLT-DD:

1. Tabel penduduk

Tabel 4. 3 Database Penduduk

No	Nama Item Data	Tipe	Lebar	Keterangan
1.	Id	Integer	11	AUTO INCREMENT
2.	Name	Varchar	255	
3.	Price	Varchar	255	
4.	No. KK	Varchar	255	No. kartu keluarga
5.	Nik	Varchar	255	No. induk kependudukan
6.	Nama	Varchar	255	Nama Penerima
7.	TTL	Varchar	255	Tempat Tanggal Lahir
8.	Alamat	Varchar	255	Alamat Penerima
9.	No. Hp	Varchar	255	
10.	Email	Varchar	255	
11.	Jml_Tanggungan	Varchar	255	Jumlah Tanggunggan Penerima
12.	Pdp_klg	Varchar	255	Pendapatan keluarga
13.	Validasi	Varchar	255	Proses penerima bantuan
14.	Date	Date		
15.	Image	Varchar	250	
16	Image1	Varchar	255	
17.	Image2	Varchar	255	
18.	Image3	Varchar	255	
19.	Image4	Varchar	255	
20.	Image5	Varchar	255	
21.	Image6	Varchar	255	

No	Nama Item Data	Tipe	Lebar	Keterangan
22.	Image7	Varchar	255	
23.	Latitude	Varchar	255	Garis Lintang
24.	Longituda	Varchar	255	Garis Bujur

2. Tabel user

Tabel 4. 4 Database User

No	Nama Item Data	Tipe	Lebar	Keterangan
1.	id	integer	11	AUTO INCREMENT
2.	Name	Varchar	32	Nama
3.	Username	Varchar	32	Nama Pengguna
4.	Password	Varchar	32	Kata Sandi
5.	Level	Varchar	32	tingkatan

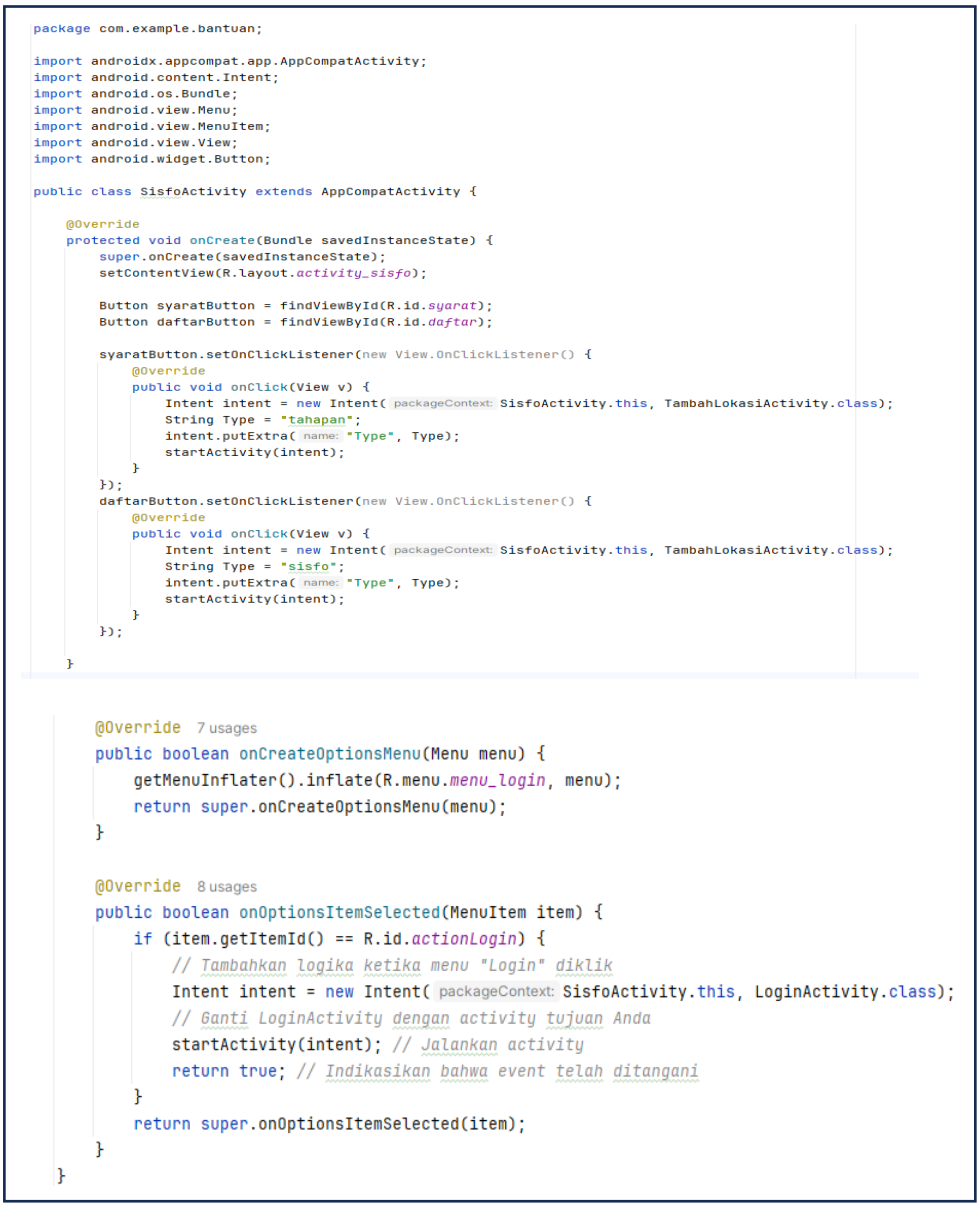
C. Detail Sistem

1. Tampilan Menu Awal

Gambar dibawah menujukkan tampilan pertama menu program yang berisi tombol syarat dan tahapan, daftar nama penyaluran BLT-DD beserta login.



Gambar 4. 19 Tampilan Menu Awal



```java
package com.example.bantuan;

import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;

public class SisfoActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_sisfo);

        Button syaratButton = findViewById(R.id.syarat);
        Button daftarButton = findViewById(R.id.daftar);

        syaratButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent( packageContext: SisfoActivity.this, TambahLokasiActivity.class);
                String Type = "tahapan";
                intent.putExtra( name: "Type", Type);
                startActivity(intent);
            }
        });
        daftarButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent( packageContext: SisfoActivity.this, TambahLokasiActivity.class);
                String Type = "sisfo";
                intent.putExtra( name: "Type", Type);
                startActivity(intent);
            }
        });

    }

    @Override  7 usages
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu_login, menu);
        return super.onCreateOptionsMenu(menu);
    }

    @Override  8 usages
    public boolean onOptionsItemSelected(MenuItem item) {
        if (item.getItemId() == R.id.actionLogin) {
            // Tambahkan logika ketika menu "Login" diklik
            Intent intent = new Intent( packageContext: SisfoActivity.this, LoginActivity.class);
            // Ganti LoginActivity dengan activity tujuan Anda
            startActivity(intent); // Jalankan activity
            return true; // Indikasikan bahwa event telah ditangani
        }
        return super.onOptionsItemSelected(item);
    }
}
```

Gambar 4. 20 Script Java Menu Awal

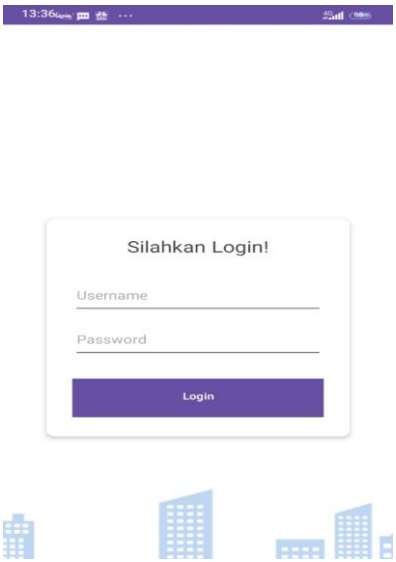
```
<TextView
    android:id="@+id/welcomeText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Selamat Datang Di Sistem Informasi Penyaluran
     Bantuan Langsung Tunai Kabupaten Enrekang"
    android:textSize="20sp"
    android:textStyle="bold"
    android:padding="16dp"
    android:gravity="center"
    android:lineSpacingExtra="8dp"
    android:letterSpacing="0.05"
    android:layout_marginTop="20dp"/>

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <Button
        android:id="@+id/syarat"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Syarat dan tahapan"
        android:layout_centerHorizontal="true"
        android:background="@drawable/rounded_button"
        android:textColor="@color/hitam"
        android:layout_marginTop="250dp"
        android:padding="10dp"/>
    <Button
        android:id="@+id/daftar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Daftar Nama Penyaluran BLT-DD"
        android:layout_centerHorizontal="true"
        android:layout_below="@id/syarat"
        android:background="@drawable/rounded_button"
        android:textColor="@color/hitam"
        android:layout_marginTop="20dp"
        android:padding="10dp"/>
```

Gambar 4. 21 Script Layout Menu Awal

2. Tampilan Login

Tampilan di bawah adalah halaman login aplikasi BLT-DD, di mana pengguna harus memasukkan username dan password lalu menekan tmbol Login untuk mengakses sistem.



Gambar 4. 22 Tampilan Login

```
package com.example.bantuan;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import com.example.bantuan.Adapter.SessionManager;
import com.example.bantuan.Api.ApiClient;
import com.example.bantuan.Api.ApiInterface;
import com.example.bantuan.Model.Login.Login;
import com.example.bantuan.Model.Login.LoginData;

import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

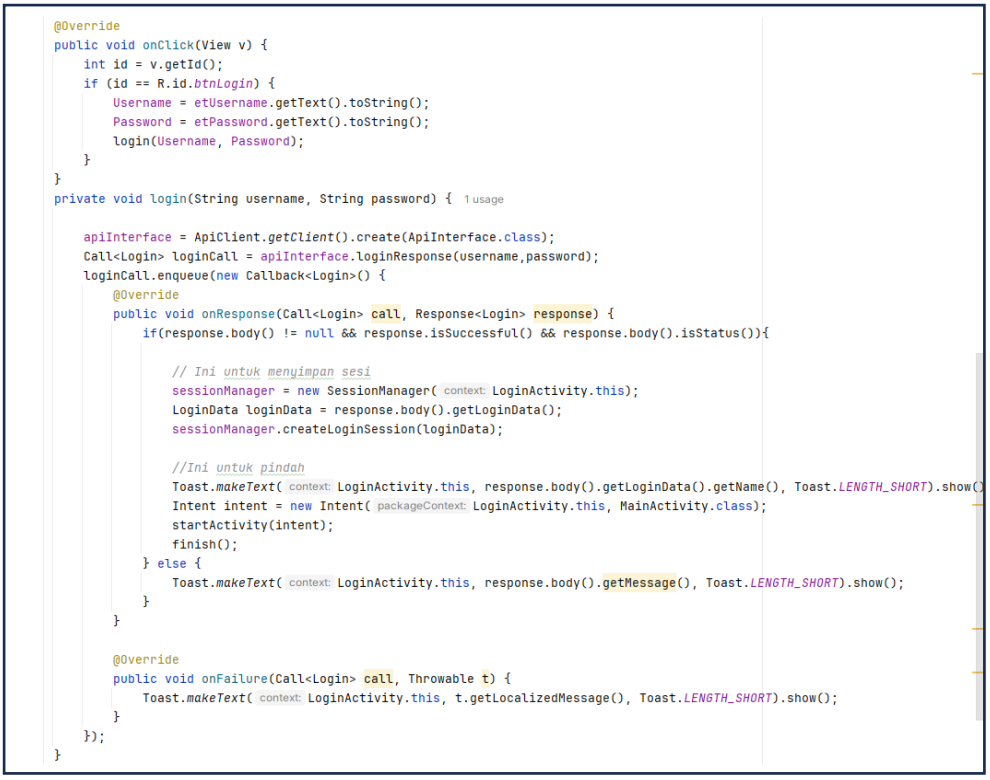
public class LoginActivity extends AppCompatActivity implements View.OnClickListener {

    EditText etUsername, etPassword;  2 usages
    Button btnLogin;  2 usages
    String Username, Password;  2 usages
    ApiInterface apiInterface;  2 usages
    SessionManager sessionManager;  2 usages


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);

        etUsername = findViewById(R.id.etUsername);
        etPassword = findViewById(R.id.etPassword);

        btnLogin = findViewById(R.id.btnLogin);
        btnLogin.setOnClickListener(this);
    }
```



```java
@Override
public void onClick(View v) {
    int id = v.getId();
    if (id == R.id.btnLogin) {
        Username = etUsername.getText().toString();
        Password = etPassword.getText().toString();
        login(Username, Password);
    }
}
private void login(String username, String password) {  1 usage

    apiInterface = ApiClient.getClient().create(ApiInterface.class);
    Call<Login> loginCall = apiInterface.loginResponse(username,password);
    loginCall.enqueue(new Callback<Login>() {
        @Override
        public void onResponse(Call<Login> call, Response<Login> response) {
            if(response.body() != null && response.isSuccessful() && response.body().isStatus()){

                // Ini untuk menyimpan sesi
                sessionManager = new SessionManager( context: LoginActivity.this);
                LoginData loginData = response.body().getLoginData();
                sessionManager.createLoginSession(loginData);

                //Ini untuk pindah
                Toast.makeText( context: LoginActivity.this, response.body().getLoginData().getName(), Toast.LENGTH_SHORT).show()
                Intent intent = new Intent( packageContext: LoginActivity.this, MainActivity.class);
                startActivity(intent);
                finish();
            } else {
                Toast.makeText( context: LoginActivity.this, response.body().getMessage(), Toast.LENGTH_SHORT).show();
            }
        }

        @Override
        public void onFailure(Call<Login> call, Throwable t) {
            Toast.makeText( context: LoginActivity.this, t.getLocalizedMessage(), Toast.LENGTH_SHORT).show();
        }
    });
}
```

Gambar 4. 23 Script Java Menu Login



```xml
<EditText
    android:id="@+id/etUsername"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Username"
    android:drawablePadding="16dp"
    android:layout_marginBottom="12dp"/>

<EditText
    android:id="@+id/etPassword"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Password"
    android:inputType="textPassword"
    android:drawablePadding="16dp"
    android:layout_marginBottom="24dp"/>

<Button
    android:id="@+id/btnLogin"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textColor="@android:color/white"
    android:background="@color/buttonColor"
    android:text="Login"/>
```

Gambar 4. 24 Layout Menu Login

3. Tampilan menu Utama Login

Gambar menunjukkan menu utama aplikasi BLT-DD setelah login, berisi tombol Profile, Data Survei, Data Master, Hasil Survei, dan Data Penerima BLT-DD untuk navigasi fitur. Tersedia juga tombol Logout untuk keluar.



Gambar 4. 25 Tampilan Menu Utama Login

```java
package com.example.bantuan;

import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import com.example.bantuan.Adapter.SessionManager;
import com.example.bantuan.Model.DataRepository;
import com.example.bantuan.Model.LoadDataTask;
import com.example.bantuan.Model.Provinsi;
import java.util.HashMap;
import java.util.List;

public class MainActivity extends AppCompatActivity {
    Button btnProfile, btnDataUser, btnDataPenduduk, btnLogout, btnDataSurvei, btnData;  3 usages
    SessionManager sessionManager;  4 usages

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        sessionManager = new SessionManager( context: MainActivity.this);

        new LoadDataTask( context: this, new LoadDataTask.DataLoadListener() {
            @Override  1 usage
            public void onDataLoaded(List<Provinsi> provinsiList) {
                if (provinsiList != null && !provinsiList.isEmpty()) {
                    DataRepository.getInstance().setProvinsiList(provinsiList);
                    android.util.Log.d( tag: "MainActivity",  msg: "Data berhasil dimuat: " + provinsiList.size() + " provinsi");
                } else {
                    android.util.Log.e( tag: "MainActivity",  msg: "Data gagal dimuat atau kosong.");
                }
            }
        }).execute();

        if (!sessionManager.isLoggedIn()) {
            moveToLogin();
        }

        btnProfile = findViewById(R.id.btnProfile);
        btnDataPenduduk = findViewById(R.id.btnDataPenduduk);
        btnDataUser = findViewById(R.id.btnDataUser);
        btnLogout = findViewById(R.id.btnLogout);
        btnDataSurvei = findViewById(R.id.btnDataSurvei);
        btnData = findViewById(R.id.btnData);
        // Mendapatkan detail user dari SessionManager
        HashMap<String, String> user = sessionManager.getUserDetail();
        String level = user.get(SessionManager.LEVEL); // Ambil level pengguna
        // Cek apakah user adalah admin
        if (!"Admin".equals(level)) {
            // Jika bukan admin, sembunyikan tombol-tombol berikut
            btnDataUser.setVisibility(View.GONE);
            btnProfile.setVisibility(View.GONE);
        }
        btnProfile.setOnClickListener(v -> {
            Intent intent = new Intent( packageContext: this, ProfileActivity.class);
            startActivity(intent);
        });
        btnDataUser.setOnClickListener(v -> {
            Intent intent;
            if ("Admin".equals(level)) {
                intent = new Intent( packageContext: this, UserDataActivity.class);
            } else {
                intent = new Intent( packageContext: this, TambahLokasiActivity.class);
                String Type = "sisfo";
                intent.putExtra( name: "Type", Type);
            }
            startActivity(intent);
        });
```

```
    btnDataPenduduk.setOnClickListener(v -> {
        Intent intent;
        if ("Admin".equals(level) || ("Aparat").equals(level)) {
            intent = new Intent( packageContext: this, PendudukDataActivity.class);
            String Validasi = "1"; // Contoh data
            intent.putExtra( name: "Valid", Validasi);
        } else {
            intent = new Intent( packageContext: this, TambahLokasiActivity.class);
            String Type = "tahapan";
            intent.putExtra( name: "Type", Type);
        }
        startActivity(intent);
    });
    btnDataSurvei.setOnClickListener(v -> {
        Intent intent = new Intent( packageContext: this, PendudukDataActivity.class);

        String Validasi = "2";
        intent.putExtra( name: "Valid", Validasi);

        startActivity(intent);
    });
    btnData.setOnClickListener(v -> {
        Intent intent = new Intent( packageContext: this, PendudukDataActivity.class);

        String Validasi = "3"; // Contoh data
        intent.putExtra( name: "Valid", Validasi);

        startActivity(intent);
    });
    btnLogout.setOnClickListener(v -> {
        sessionManager.logoutSession();
        moveToLogin();
    });
}
private void moveToLogin() {  2 usages
    Intent intent = new Intent( packageContext: MainActivity.this, SisfoActivity.class);
    intent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TASK | Intent.FLAG_ACTIVITY_NO_HISTORY);
    startActivity(intent);
    finish();
```

Gambar 4. 26 Script Java Menu Utama Login

```
<Button
    android:id="@+id/btnProfile"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="200dp"
    android:background="@drawable/rounded_button"
    android:text="Profile"
    android:textColor="@color/hitam"
    app:layout_constraintEnd_toStartOf="@+id/btnDataUser"
    app:layout_constraintHorizontal_bias="0.502"
    app:layout_constraintStart_toEndOf="@+id/btnDataPenduduk"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintWidth_percent="0.45" />

<Button
    android:id="@+id/btnDataUser"
    android:layout_width="0dp"
    android:layout_height="60dp"
    android:layout_marginTop="270dp"
    android:text="Data Master"
    android:background="@drawable/rounded_button"
    android:textColor="@color/hitam"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toEndOf="@+id/btnDataPenduduk"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintWidth_percent="0.45" />


        <Button
            android:id="@+id/btnDataPenduduk"
            android:layout_width="0dp"
            android:layout_height="60dp"
            android:layout_marginTop="270dp"
            android:background="@drawable/rounded_button"
            android:textColor="@color/hitam"
            android:text="Data Survei"
            app:layout_constraintEnd_toStartOf="@+id/btnDataUser"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintWidth_percent="0.45" />
        <Button
            android:id="@+id/btnDataSurvei"
            android:layout_width="0dp"
            android:layout_height="60dp"
            android:layout_marginTop="20dp"
            android:background="@drawable/rounded_button"
            android:text="Hasil Survei"
            android:textColor="@color/hitam"
            app:cornerRadius="20dp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.061"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@+id/btnDataPenduduk"
            app:layout_constraintVertical_bias="0.0"
            app:layout_constraintWidth_percent="0.45" />
```

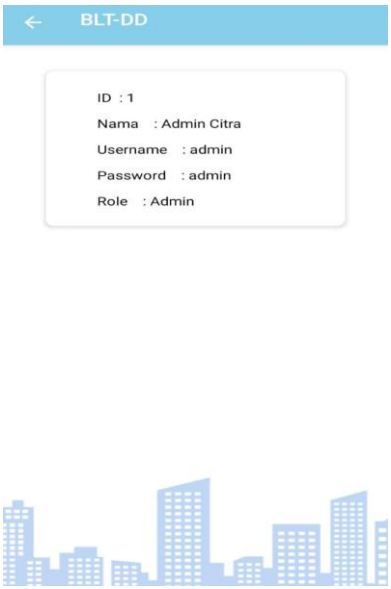
```
<Button
    android:id="@+id/btnData"
    android:layout_width="0dp"
    android:layout_height="60dp"
    android:layout_marginTop="20dp"
    android:background="@drawable/rounded_button"
    android:text="Data Penerima BLT-DD"
    android:textColor="@color/hitam"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.942"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/btnDataPenduduk"
    app:layout_constraintVertical_bias="0.0"
    app:layout_constraintWidth_percent="0.45" />

<Button
    android:id="@+id/btnLogout"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:text="Logout"
    android:background="@drawable/rounded_button"
    android:textColor="@color/hitam"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/btnData"
    app:layout_constraintVertical_bias="0.0" />
```

Gambar 4. 27 Script Loyout Menu Utama Login

4. Tampilan Profile

Tampilan di bawah menunjukkan halaman Profile dalam aplikasi BLT-DD. Di dalam kotak informasi, terdapat data pengguna yang mencakup ID**,** Nama (Admin Citra), Username**,** Password, dan Role (Admin).



Gambar 4. 28 Tampilan Profile

```java
package com.example.bantuan;

import androidx.appcompat.app.AppCompatActivity;
import androidx.cardview.widget.CardView;
import android.os.Bundle;
import android.view.View;
import android.webkit.WebSettings;
import android.webkit.WebView;
import android.webkit.WebViewClient;
import android.widget.TextView;
import com.example.bantuan.Adapter.SessionManager;

public class ProfileActivity extends AppCompatActivity {
    TextView etUsername, etName, etLevel, etId, etPassword;  2 usages
    SessionManager sessionManager;  6 usages
    String username, name, level, id, password;  2 usages
    private WebView webViewProfile;  6 usages
    private CardView cardProfile;  3 usages

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_profile);

        // Inisialisasi TextView
        etUsername = findViewById(R.id.etMainUsername);
        etName = findViewById(R.id.etMainName);
        etLevel = findViewById(R.id.etMainLevel);
        etId = findViewById(R.id.etMainId);
        etPassword = findViewById(R.id.etMainPassword);
        webViewProfile = findViewById(R.id.webViewProfile);
        cardProfile = findViewById(R.id.cardProfile);
        // Inisialisasi session manager
        sessionManager = new SessionManager( context: ProfileActivity.this);
        // Mengambil data dari session manager
        username = sessionManager.getUserDetail().get(SessionManager.USERNAME);
        name = sessionManager.getUserDetail().get(SessionManager.NAME);
        level = sessionManager.getUserDetail().get(SessionManager.LEVEL);
        id = sessionManager.getUserDetail().get(SessionManager.USER_ID);
        password = sessionManager.getUserDetail().get(SessionManager.PASSWORD);

        etId.setText("ID       : " + id);
        etName.setText("Nama     : " + name);
        etUsername.setText("Username    : " + username);
        etPassword.setText("Password    : " + password);
        etLevel.setText("Role     : " + level);
        // Aktifkan JavaScript jika dibutuhkan
        WebSettings webSettings = webViewProfile.getSettings();
        webSettings.setJavaScriptEnabled(true);

        // Pastikan WebView tetap di aplikasi (tanpa membuka browser eksternal)
        webViewProfile.setWebViewClient(new WebViewClient());
        // Muat URL yang ingin ditampilkan dengan ID dari session
        webViewProfile.loadUrl("https://bantuan.maiwa.id/profile.php?id=" + id);
        webViewProfile.setVisibility(View.VISIBLE);
        cardProfile.setVisibility(View.VISIBLE);
        // Cek level user
        if (level.equals("Admin")) {
            // Sembunyikan WebView jika level adalah admin
            webViewProfile.setVisibility(View.GONE);
            // Tampilkan CardView untuk admin
        } else {
            // Tampilkan WebView untuk user biasa
            // Sembunyikan CardView untuk user biasa
            cardProfile.setVisibility(View.GONE);

        }
    }
}
```

Gambarmbar 4. 29 Script Java Menu Profile

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="16dp">
    <!-- Bagian ID -->
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:gravity="center_vertical"
        android:layout_marginTop="8dp">

        <ImageView
            android:layout_width="24dp"
            android:layout_height="24dp"
            android:layout_marginEnd="8dp" />

        <TextView
            android:id="@+id/etMainId"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:textSize="18sp"
            android:textColor="@color/hitam"
            android:layout_weight="1" />
    </LinearLayout>
    <WebView
        android:id="@+id/webViewProfile"
        android:layout_width="match_parent"
        android:layout_height="500dp" />
    <!-- Menambahkan Guideline untuk pengaturan margin kiri dan kanan
    <androidx.constraintlayout.widget.Guideline
        android:id="@+id/guidelineStart"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        app:layout_constraintGuide_begin="24dp" />

    <androidx.constraintlayout.widget.Guideline
        android:id="@+id/guidelineEnd"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        app:layout_constraintGuide_end="24dp" />

    <!-- CardView untuk membuat informasi profil lebih menarik -->
    <androidx.cardview.widget.CardView
        android:id="@+id/cardProfile"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:layout_marginBottom="16dp"
        app:cardElevation="4dp"
        app:cardCornerRadius="8dp"
        app:layout_constraintEnd_toStartOf="@id/guidelineEnd"
        app:layout_constraintStart_toEndOf="@id/guidelineStart"
        app:layout_constraintTop_toTopOf="parent">

        etId.setText("ID  : " + id);
        etName.setText("Nama    : " + name);
        etUsername.setText("Username    : " + username);
        etPassword.setText("Password    : " + password);
        etLevel.setText("Role    : " + level);
        // Aktifkan JavaScript jika dibutuhkan
        WebSettings webSettings = webViewProfile.getSettings();
        webSettings.setJavaScriptEnabled(true);

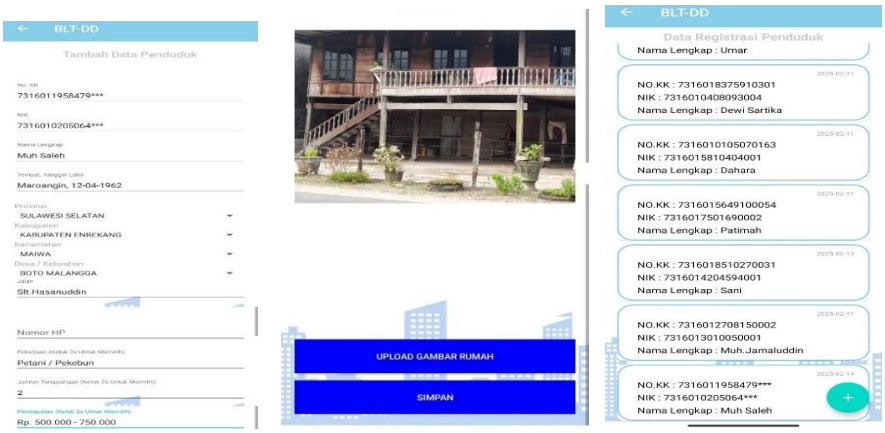
        // Pastikan WebView tetap di aplikasi (tanpa membuka browser eksternal)
        webViewProfile.setWebViewClient(new WebViewClient());
        // Muat URL yang ingin ditampilkan dengan ID dari session
        webViewProfile.loadUrl("https://bantuan.maiwa.id/profile.php?id=" + id);
        webViewProfile.setVisibility(View.VISIBLE);
        cardProfile.setVisibility(View.VISIBLE);
        // Cek level user
        if (level.equals("Admin")) {
            // Sembunyikan WebView jika level adalah admin
            webViewProfile.setVisibility(View.GONE);
            // Tampilkan CardView untuk admin
        } else {
            // Tampilkan WebView untuk user biasa
            // Sembunyikan CardView untuk user biasa
            cardProfile.setVisibility(View.GONE);

        }
    }
}
```

Gambar 4. 30 Script Layout Menu Profile

5. **Tampian Data Survei**

Gambar menampilkan aplikasi BLT-DD dengan menu utama, formulir tambah data penduduk, dan fitur unggah gambar rumah. Pengguna dapat memasukkan data pribadi, alamat, kontak, serta menyimpan informasi**.**



Gambar 4. 31 Tampilan Data Survei

```java
public class PendudukTambahActivity extends AppCompatActivity {
    private Spinner spinnerProvinsi, spinnerKabupaten, spinnerKecamatan, spinnerDesa;  4 usages
    private List<Provinsi> provinsiList;  4 usages
    private EditText edtJalan;  3 usages

    EditText edtName, edtPrice, edtNoKK, edtNama, edtNik, edtTtl, edtAlamat, edtNoHp, edtEmail, edtTanggung, edtPendapatan;
    Button btnGalery, btSubmit;  2 usages
    ImageView imgHolder;  2 usages

    private String mediaPath;  4 usages

    ApiInterface mApiInterface;  3 usages
    private static final int REQUEST_PICK_PHOTO = Config.REQUEST_PICK_PHOTO;  2 usages
    private static final int REQUEST_WRITE_PERMISSION = Config.REQUEST_WRITE_PERMISSION;  2 usages
    private static final String INSERT_FLAG = Config.INSERT_FLAG;  2 usages

    // Akses Izin Ambil Gambar dari Storage
    @Override  14 usages
    public void onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults) {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults);
        if (requestCode == REQUEST_WRITE_PERMISSION && grantResults[0] == PackageManager.PERMISSION_GRANTED) {
            saveImageUpload();
        }
    }
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_penduduk_tambah);
        edtJalan = findViewById(R.id.edt_jalan);
        spinnerProvinsi = findViewById(R.id.spinner_provinsi);
        spinnerKabupaten = findViewById(R.id.spinner_kabupaten);
        spinnerKecamatan = findViewById(R.id.spinner_kecamatan);
        spinnerDesa = findViewById(R.id.spinner_desa);
        provinsiList = DataRepository.getInstance().getProvinsiList();
        if (provinsiList != null) {
            setupSpinners();
        } else {
            // Tampilkan pesan jika data belum siap
        }
    edtJalan.setOnFocusChangeListener((v, hasFocus) -> {
        if (!hasFocus) {
            updateAlamatLengkap();
        }
    });

    // Identifikasi Komponen Form
    edtName = findViewById(R.id.edt_name);
    edtPrice = findViewById(R.id.edt_price);
    edtNoKK = findViewById(R.id.edt_no_kk);
    edtNik = findViewById(R.id.edt_nik);
    edtNama = findViewById(R.id.edt_nama);
    edtTtl = findViewById(R.id.edt_ttl);
    edtAlamat = findViewById(R.id.edt_alamat);
    edtNoHp = findViewById(R.id.edt_no_hp);
    edtEmail = findViewById(R.id.edt_email);
    edtTanggung = findViewById(R.id.edt_tanggung);
    edtPendapatan = findViewById(R.id.edt_pendapatan);
    imgHolder = findViewById(R.id.imgHolder);
    btnGalery = findViewById(R.id.btn_galery);
    btSubmit = findViewById(R.id.btn_submit);

    edtEmail.setOnClickListener(v -> {
        String[] pekerjaanArray = getResources().getStringArray(R.array.pekerjaan_array);

        PopupMenu popupMenu = new PopupMenu( context: this, edtEmail);
        Menu menu = popupMenu.getMenu();

        for (String pekerjaan : pekerjaanArray) {
            menu.add(pekerjaan);
        }

        popupMenu.setOnMenuItemClickListener(item -> {
            edtEmail.setText(item.getTitle());  // Set pekerjaan yang dipilih ke EditText
            return true;
        });

        popupMenu.show();
    });
```

```
    edtPendapatan.setOnClickListener(v -> {
        String[] pendapatanArray = getResources().getStringArray(R.array.pendapatan_array);

        PopupMenu popupMenu = new PopupMenu( context: this, edtPendapatan);
        Menu menu = popupMenu.getMenu();

        for (String pendapatan : pendapatanArray) {
            menu.add(pendapatan);
        }

        popupMenu.setOnMenuItemClickListener(item -> {
            edtPendapatan.setText(item.getTitle());  // Set pekerjaan yang dipilih ke EditText
            return true;
        });

        popupMenu.show();
    });

    edtTanggung.setOnClickListener(v -> {
        String[] jumlahArray = getResources().getStringArray(R.array.jumlah_array);

        PopupMenu popupMenu = new PopupMenu( context: this, edtTanggung);
        Menu menu = popupMenu.getMenu();

        for (String jumlah : jumlahArray) {
            menu.add(jumlah);
        }

        popupMenu.setOnMenuItemClickListener(item -> {
            edtTanggung.setText(item.getTitle());  // Set pekerjaan yang dipilih ke EditText
            return true;
        });

        popupMenu.show();
    });
    edtName.setVisibility(View.GONE);
    edtPrice.setVisibility(View.GONE);
    edtAlamat.setVisibility(View.GONE);

    edtPendapatan.setInputType(InputType.TYPE_NULL);
    edtPendapatan.setKeyListener(null);

    edtEmail.setInputType(InputType.TYPE_NULL);
    edtEmail.setKeyListener(null);

    edtTanggung.setInputType(InputType.TYPE_NULL);
    edtTanggung.setKeyListener(null);


    // Definisi API
    mApiInterface = ApiClient.getClient().create(ApiInterface.class);

    // Fungsi Tombol Pilih Galery
    btnGalery.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent galleryIntent = new Intent(Intent.ACTION_PICK,
                    android.provider.MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
            startActivityForResult(galleryIntent, REQUEST_PICK_PHOTO);
        }
    });


    // Fungsi Tombol Simpan
    btSubmit.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) { requestPermission(); }
    });
}
```

```
        ArrayAdapter<String> kecamatanAdapter = new ArrayAdapter<>( context: PendudukTambahActivity.this,
                android.R.layout.simple_spinner_item, kecamatanNames);
        spinnerKecamatan.setAdapter(kecamatanAdapter);

        spinnerKecamatan.setOnItemSelectedListener(new android.widget.AdapterView.OnItemSelectedListener() {
            @Override  no usages
            public void onItemSelected(android.widget.AdapterView<?> parent, android.view.View view,
                                       int position, long id) {
                Kecamatan selectedKecamatan = selectedKabupaten.kecamatanList.get(position);

                // Isi Spinner Desa
                List<String> desaNames = new ArrayList<>();
                for (Desa desa : selectedKecamatan.desaList) {
                    desaNames.add(desa.namaDesa);
                }

                ArrayAdapter<String> desaAdapter = new ArrayAdapter<>( context: PendudukTambahActivity.this,
                        android.R.layout.simple_spinner_item, desaNames);
                spinnerDesa.setAdapter(desaAdapter);
            }

            @Override  no usages
            public void onNothingSelected(android.widget.AdapterView<?> parent) {}
        });
    }

    @Override  no usages
    public void onNothingSelected(android.widget.AdapterView<?> parent) {}
});
}
```

```
        private void setupSpinners() {  1 usage
            // Isi Spinner Provinsi
            List<String> provinsiNames = new ArrayList<>();
            for (Provinsi provinsi : provinsiList) {
                provinsiNames.add(provinsi.namaProvinsi);
            }

            ArrayAdapter<String> provinsiAdapter = new ArrayAdapter<>( context: this, android.R.layout.simple_spinner_item,
                    provinsiNames);
            spinnerProvinsi.setAdapter(provinsiAdapter);

            spinnerProvinsi.setOnItemSelectedListener(new android.widget.AdapterView.OnItemSelectedListener() {
                @Override  no usages
                public void onItemSelected(android.widget.AdapterView<?> parent, android.view.View view, int position,
                                           long id) {
                    Provinsi selectedProvinsi = provinsiList.get(position);

                    // Isi Spinner Kabupaten
                    List<String> kabupatenNames = new ArrayList<>();
                    for (Kabupaten kabupaten : selectedProvinsi.kabupatenList) {
                        kabupatenNames.add(kabupaten.namaKabupaten);
                    }

                    ArrayAdapter<String> kabupatenAdapter = new ArrayAdapter<>( context: PendudukTambahActivity.this,
                            android.R.layout.simple_spinner_item, kabupatenNames);
                    spinnerKabupaten.setAdapter(kabupatenAdapter);

                    spinnerKabupaten.setOnItemSelectedListener(new android.widget.AdapterView.OnItemSelectedListener() {
                        @Override  no usages
                        public void onItemSelected(android.widget.AdapterView<?> parent, android.view.View view,
                                                   int position, long id) {
                            Kabupaten selectedKabupaten = selectedProvinsi.kabupatenList.get(position);

                            // Isi Spinner Kecamatan
                            List<String> kecamatanNames = new ArrayList<>();
                            for (Kecamatan kecamatan : selectedKabupaten.kecamatanList) {
                                kecamatanNames.add(kecamatan.namaKecamatan);
                            }
                ArrayAdapter<String> kecamatanAdapter = new ArrayAdapter<>( context: PendudukTambahActivity.this,
                        android.R.layout.simple_spinner_item, kecamatanNames);
                spinnerKecamatan.setAdapter(kecamatanAdapter);

                spinnerKecamatan.setOnItemSelectedListener(new android.widget.AdapterView.OnItemSelectedListener() {
                    @Override  no usages
                    public void onItemSelected(android.widget.AdapterView<?> parent, android.view.View view,
                                               int position, long id) {
                        Kecamatan selectedKecamatan = selectedKabupaten.kecamatanList.get(position);

                        // Isi Spinner Desa
                        List<String> desaNames = new ArrayList<>();
                        for (Desa desa : selectedKecamatan.desaList) {
                            desaNames.add(desa.namaDesa);
                        }

                        ArrayAdapter<String> desaAdapter = new ArrayAdapter<>( context: PendudukTambahActivity.this,
                                android.R.layout.simple_spinner_item, desaNames);
                        spinnerDesa.setAdapter(desaAdapter);
                    }

                    @Override  no usages
                    public void onNothingSelected(android.widget.AdapterView<?> parent) {}
                });
            }

            @Override  no usages
            public void onNothingSelected(android.widget.AdapterView<?> parent) {}
        });
    }
```

```
    MultipartBody.Part partImage = null;
    if (mediaPath != null && !mediaPath.isEmpty()) {
        File imagefile = new File(mediaPath);
        RequestBody reqBody = RequestBody.create(MediaType.parse( string: "multipart/form-file"), imagefile);
        partImage = MultipartBody.Part.createFormData( name: "image", imagefile.getName(), reqBody);
    }

    // Panggil API dengan gambar atau tanpa gambar
    Call<PostPutDelPenduduk> postPendudukCall;
    if (partImage != null) {
        // Jika gambar ada, kirim bersama data
        postPendudukCall = mApiInterface.postPenduduk(
                partImage,
                name,
                price,
                RequestBody.create(MediaType.parse( string: "text/plain"), date),
                noKk,
                nik,
                nama,
                ttl,
                alamat,
                noHp,
                email,
                jml_tanggung,
                pdp_klg,
                RequestBody.create(MediaType.parse( string: "text/plain"), INSERT_FLAG)
        );
    } else {
        // Jika gambar kosong, kirim tanpa gambar
        postPendudukCall = mApiInterface.postPendudukWithoutImage(
                name,
                price,
                RequestBody.create(MediaType.parse( string: "text/plain"), date),
                noKk,
                nik,
                nama,
                ttl,
                alamat,
                noHp,
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    if (resultCode == RESULT_OK) {
        if (requestCode == REQUEST_PICK_PHOTO) {
            if (data != null) {
                // Ambil Image Dari Galeri dan Foto
                Uri selectedImage = data.getData();
                String[] filePat
                    Uri selectedImage = data.getData()
                    Bantuan.app.main
                Cursor cursor =                                  filePathColumn, selection: null, selectionArgs: null,
                        sortOrder: null);
                assert cursor != null;
                cursor.moveToFirst();

                int columnIndex = cursor.getColumnIndex(filePathColumn[0]);
                mediaPath = cursor.getString(columnIndex);
                imgHolder.setImageURI(data.getData());
                cursor.close();
            }
        }
    }
}
// Simpan Gambar
private void saveImageUpload() { 2 usages
    final String date = new SimpleDateFormat( pattern: "yyyy-MM-dd", Locale.getDefault()).format(new Date());

    // Mengambil input dari EditText
    RequestBody name = RequestBody.create(MediaType.parse( string: "text/plain"), edtName.getText().toString());
    RequestBody price = RequestBody.create(MediaType.parse( string: "text/plain"), edtPrice.getText().toString());
    RequestBody noKk = RequestBody.create(MediaType.parse( string: "text/plain"), edtNoKK.getText().toString());
    RequestBody nik = RequestBody.create(MediaType.parse( string: "text/plain"), edtNik.getText().toString());
    RequestBody nama = RequestBody.create(MediaType.parse( string: "text/plain"), edtNama.getText().toString());
    RequestBody ttl = RequestBody.create(MediaType.parse( string: "text/plain"), edtTtl.getText().toString());
    RequestBody alamat = RequestBody.create(MediaType.parse( string: "text/plain"), edtAlamat.getText().toString());
    RequestBody noHp = RequestBody.create(MediaType.parse( string: "text/plain"), edtNoHp.getText().toString());
    RequestBody email = RequestBody.create(MediaType.parse( string: "text/plain"), edtEmail.getText().toString());
    RequestBody jml_tanggung = RequestBody.create(MediaType.parse( string: "text/plain"), edtTanggung.getText().toString());
    RequestBody pdp_klg = RequestBody.create(MediaType.parse( string: "text/plain"), edtPendapatan.getText().toString());
```



```
                email,
                jml_tanggung,
                pdp_klg,
                RequestBody.create(MediaType.parse( string: "text/plain"), INSERT_FLAG)
        );
    }

    postPendudukCall.enqueue(new Callback<PostPutDelPenduduk>() {
        @Override
        public void onResponse(Call<PostPutDelPenduduk> call, Response<PostPutDelPenduduk> response) {
            if (response.isSuccessful() && response.body() != null) {
                PendudukDataActivity.ma.refresh();
                finish();
            } else {
                Toast.makeText(getApplicationContext(), text: "Gagal menyimpan data", Toast.LENGTH_LONG).show();
            }
        }

        @Override
        public void onFailure(Call<PostPutDelPenduduk> call, Throwable t) {
            Log.d( tag: "RETRO", msg: "ON FAILURE : " + t.getMessage());
            Toast.makeText(getApplicationContext(), text: "Error, data tidak tersimpan", Toast.LENGTH_LONG).show();
        }
    });
}
private void requestPermission() {  1 usage
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.Q) {
        requestPermissions(new String[]{Manifest.permission.READ_MEDIA_IMAGES, Manifest.permission.WRITE_EXTERNAL_STORAGE},
                REQUEST_WRITE_PERMISSION);
    } else {
        saveImageUpload();
    }
}
private void updateAlamatLengkap() {  1 usage
    String jalan = edtJalan.getText().toString();
    String alamatLengkap = spinnerProvinsi.getSelectedItem().toString() + ", " + spinnerKabupaten.getSelectedItem().
            toString() + ", " + spinnerKecamatan.getSelectedItem().toString() + ", " + spinnerDesa.getSelectedItem().
            toString() + ", " + jalan;
    edtAlamat.setText(alamatLengkap);
}
```

Gambar 4. 32 Script Menu Data Survei

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    xmlns:app="http://schemas.a-droid.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_marginHorizontal="16dp"
    android:layout_marginTop="5dp"
    android:orientation="vertical">

    <androidx.cardview.widget.CardView
        android:id="@+id/cv_item_Penduduk"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginVertical="1dp"
        app:cardCornerRadius="8dp"
        app:cardElevation="1dp">

        <androidx.constraintlayout.widget.ConstraintLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@drawable/rounded_button"
            android:padding="10dp">

            <ImageView
                android:id="@+id/img_item_photo"
                android:layout_width="0dp"
                android:layout_height="0dp"
                android:scaleType="centerCrop"
                app:layout_constraintDimensionRatio="1:1"
                app:layout_constraintStart_toStartOf="parent"
                app:layout_constraintTop_toTopOf="parent"
                app:layout_constraintBottom_toBottomOf="parent" />

    <TextView
        android:id="@+id/tv_item_date"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginEnd="10dp"
        android:textColor="@android:color/darker_gray"
        android:textSize="12sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        tools:text="tanggal" />

    <TextView
        android:id="@+id/tv_item_name"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="4dp"
        android:textColor="@android:color/black"
        android:textSize="20sp"
        android:textStyle="bold"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/tv_item_date"
        tools:text="name" />
```

```
<TextView
    android:id="@+id/tv_item_nik"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginTop="4dp"
    android:textColor="@android:color/black"
    android:textSize="18sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@id/tv_item_no_kk"
    tools:text="NIK" />

<TextView
    android:id="@+id/tv_item_nama"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginTop="4dp"
    android:textColor="@android:color/black"
    android:textSize="18sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@id/tv_item_nik"
    tools:text="Nama" />

 <TextView
     android:id="@+id/tv_item_price"
     android:layout_width="0dp"
     android:layout_height="wrap_content"
     android:layout_marginStart="16dp"
     android:layout_marginTop="4dp"
     android:textColor="@android:color/black"
     android:textSize="14sp"
     android:ellipsize="end"
     android:maxLines="1"
     app:layout_constraintEnd_toEndOf="parent"
     app:layout_constraintStart_toStartOf="parent"
     app:layout_constraintTop_toBottomOf="@id/tv_item_name"
     tools:text="harga" />

 <!-- Informasi Penduduk -->
 <TextView
     android:id="@+id/tv_item_no_kk"
     android:layout_width="0dp"
     android:layout_height="wrap_content"
     android:layout_marginStart="16dp"
     android:layout_marginTop="4dp"
     android:textColor="@android:color/black"
     android:textSize="18sp"
     app:layout_constraintEnd_toEndOf="parent"
     app:layout_constraintStart_toStartOf="parent"
     app:layout_constraintTop_toBottomOf="@id/tv_item_price"
     tools:text="No. KK" />
```

```
    <TextView
        android:id="@+id/tv_item_ttl"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="4dp"
        android:textColor="@android:color/black"
        android:textSize="14sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/tv_item_nama"
        tools:text="TTL" />
    <TextView
        android:id="@+id/tv_item_alamat"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="4dp"
        android:textColor="@android:color/black"
        android:textSize="14sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/tv_item_ttl"
        tools:text="Alamat" />
    <TextView
        android:id="@+id/tv_item_no_hp"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="4dp"
        android:textColor="@android:color/black"
        android:textSize="14sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/tv_item_alamat"
        tools:text="No. HP" />
```

```
        <TextView
            android:id="@+id/tv_item_email"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:layout_marginTop="4dp"
            android:textColor="@android:color/black"
            android:textSize="14sp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@id/tv_item_no_hp"
            tools:text="Email" />


        <TextView
            android:id="@+id/tv_item_validasi"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:layout_marginTop="4dp"
            android:textColor="@android:color/black"
            android:textSize="18sp"
            android:textStyle="bold"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@id/tv_item_email"
            tools:text="Validasi" />

    </androidx.constraintlayout.widget.ConstraintLayout>
</androidx.cardview.widget.CardView>


</LinearLayout>
```

```
edtJalan.setOnFocusChangeListener((v, hasFocus) -> {
    if (!hasFocus) {
        updateAlamatLengkap();
    }
});

// Identifikasi Komponen Form
edtName = findViewById(R.id.edt_name);
edtPrice = findViewById(R.id.edt_price);
edtNoKK = findViewById(R.id.edt_no_kk);
edtNik = findViewById(R.id.edt_nik);
edtNama = findViewById(R.id.edt_nama);
edtTtl = findViewById(R.id.edt_ttl);
edtAlamat = findViewById(R.id.edt_alamat);
edtNoHp = findViewById(R.id.edt_no_hp);
edtEmail = findViewById(R.id.edt_email);
edtTanggung = findViewById(R.id.edt_tanggung);
edtPendapatan = findViewById(R.id.edt_pendapatan);
imgHolder = findViewById(R.id.imgHolder);
btnGalery = findViewById(R.id.btn_galery);
btSubmit = findViewById(R.id.btn_submit);

edtEmail.setOnClickListener(v -> {
    String[] pekerjaanArray = getResources().getStringArray(R.array.pekerjaan_array);

    PopupMenu popupMenu = new PopupMenu( context: this, edtEmail);
    Menu menu = popupMenu.getMenu();

    for (String pekerjaan : pekerjaanArray) {
        menu.add(pekerjaan);
    }

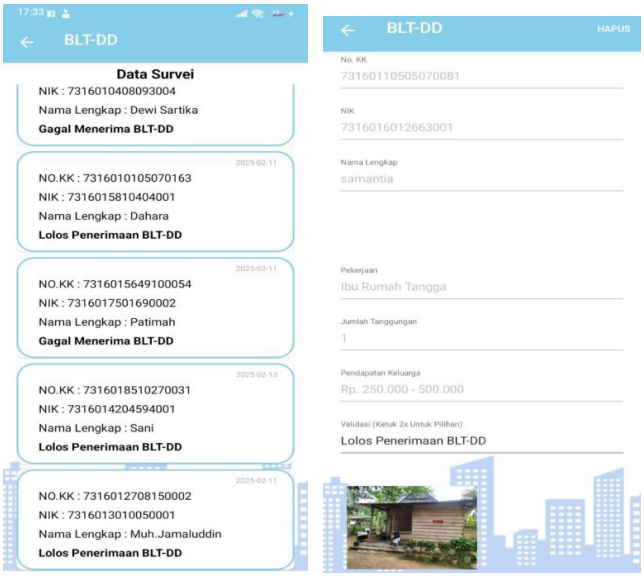
    popupMenu.setOnMenuItemClickListener(item -> {
        edtEmail.setText(item.getTitle());  // Set pekerjaan yang dipilih ke EditText
        return true;
    });

    popupMenu.show();
});
```

Gambar 4. 33 Script Layout data survei

6. Tampilan menu Hasil Survei

Gambar menunjukkan antarmuka aplikasi pendataan penerima BLT-DD. Setiap kartu menampilkan informasi warga, termasuk NO, KK, NIK, nama lengkap, dan status penerimaan BLT-DD (Lolos/Gagal).



Gambar 4. 34 Tampilan Menu Hasil Survei

```java
public class PendudukTambahActivity extends AppCompatActivity {
    private Spinner spinnerProvinsi, spinnerKabupaten, spinnerKecamatan, spinnerDesa;  4 usages
    private List<Provinsi> provinsiList;  4 usages
    private EditText edtJalan;  3 usages

    EditText edtName, edtPrice, edtNoKK, edtNama, edtNik, edtTtl, edtAlamat, edtNoHp, edtEmail, edtTanggung, edtPendapatan;
    Button btnGalery, btSubmit;  2 usages
    ImageView imgHolder;  2 usages

    private String mediaPath;  4 usages

    ApiInterface mApiInterface;  3 usages
    private static final int REQUEST_PICK_PHOTO = Config.REQUEST_PICK_PHOTO;  2 usages
    private static final int REQUEST_WRITE_PERMISSION = Config.REQUEST_WRITE_PERMISSION;  2 usages
    private static final String INSERT_FLAG = Config.INSERT_FLAG;  2 usages

    // Akses Izin Ambil Gambar dari Storage
    @Override  14 usages
    public void onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults) {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults);
        if (requestCode == REQUEST_WRITE_PERMISSION && grantResults[0] == PackageManager.PERMISSION_GRANTED) {
            saveImageUpload();
        }
    }
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_penduduk_tambah);
        edtJalan = findViewById(R.id.edt_jalan);
        spinnerProvinsi = findViewById(R.id.spinner_provinsi);
        spinnerKabupaten = findViewById(R.id.spinner_kabupaten);
        spinnerKecamatan = findViewById(R.id.spinner_kecamatan);
        spinnerDesa = findViewById(R.id.spinner_desa);
        provinsiList = DataRepository.getInstance().getProvinsiList();
        if (provinsiList != null) {
            setupSpinners();
        } else {
            // Tampilkan pesan jika data belum siap
        }
```

```java
    edtJalan.setOnFocusChangeListener((v, hasFocus) -> {
        if (!hasFocus) {
            updateAlamatLengkap();
        }
    });

    // Identifikasi Komponen Form
    edtName = findViewById(R.id.edt_name);
    edtPrice = findViewById(R.id.edt_price);
    edtNoKK = findViewById(R.id.edt_no_kk);
    edtNik = findViewById(R.id.edt_nik);
    edtNama = findViewById(R.id.edt_nama);
    edtTtl = findViewById(R.id.edt_ttl);
    edtAlamat = findViewById(R.id.edt_alamat);
    edtNoHp = findViewById(R.id.edt_no_hp);
    edtEmail = findViewById(R.id.edt_email);
    edtTanggung = findViewById(R.id.edt_tanggung);
    edtPendapatan = findViewById(R.id.edt_pendapatan);
    imgHolder = findViewById(R.id.imgHolder);
    btnGalery = findViewById(R.id.btn_galery);
    btSubmit = findViewById(R.id.btn_submit);

    edtEmail.setOnClickListener(v -> {
        String[] pekerjaanArray = getResources().getStringArray(R.array.pekerjaan_array);

        PopupMenu popupMenu = new PopupMenu( context: this, edtEmail);
        Menu menu = popupMenu.getMenu();

        for (String pekerjaan : pekerjaanArray) {
            menu.add(pekerjaan);
        }

        popupMenu.setOnMenuItemClickListener(item -> {
            edtEmail.setText(item.getTitle());  // Set pekerjaan yang dipilih ke EditText
            return true;
        });

        popupMenu.show();
    });
```

```
edtPendapatan.setOnClickListener(v -> {
    String[] pendapatanArray = getResources().getStringArray(R.array.pendapatan_array);

    PopupMenu popupMenu = new PopupMenu( context: this, edtPendapatan);
    Menu menu = popupMenu.getMenu();

    for (String pendapatan : pendapatanArray) {
        menu.add(pendapatan);
    }

    popupMenu.setOnMenuItemClickListener(item -> {
        edtPendapatan.setText(item.getTitle());  // Set pekerjaan yang dipilih ke EditText
        return true;
    });

    popupMenu.show();
});

edtTanggung.setOnClickListener(v -> {
    String[] jumlahArray = getResources().getStringArray(R.array.jumlah_array);

    PopupMenu popupMenu = new PopupMenu( context: this, edtTanggung);
    Menu menu = popupMenu.getMenu();

    for (String jumlah : jumlahArray) {
        menu.add(jumlah);
    }

    popupMenu.setOnMenuItemClickListener(item -> {
        edtTanggung.setText(item.getTitle());  // Set pekerjaan yang dipilih ke EditText
        return true;
    });

    popupMenu.show();
});
```

```
private void setupSpinners() {  1 usage
    // Isi Spinner Provinsi
    List<String> provinsiNames = new ArrayList<>();
    for (Provinsi provinsi : provinsiList) {
        provinsiNames.add(provinsi.namaProvinsi);
    }

    ArrayAdapter<String> provinsiAdapter = new ArrayAdapter<>( context: this, android.R.layout.simple_spinner_item,
            provinsiNames);
    spinnerProvinsi.setAdapter(provinsiAdapter);

    spinnerProvinsi.setOnItemSelectedListener(new android.widget.AdapterView.OnItemSelectedListener() {
        @Override  no usages
        public void onItemSelected(android.widget.AdapterView<?> parent, android.view.View view, int position,
                                   long id) {
            Provinsi selectedProvinsi = provinsiList.get(position);

            // Isi Spinner Kabupaten
            List<String> kabupatenNames = new ArrayList<>();
            for (Kabupaten kabupaten : selectedProvinsi.kabupatenList) {
                kabupatenNames.add(kabupaten.namaKabupaten);
            }

            ArrayAdapter<String> kabupatenAdapter = new ArrayAdapter<>( context: PendudukTambahActivity.this,
                    android.R.layout.simple_spinner_item, kabupatenNames);
            spinnerKabupaten.setAdapter(kabupatenAdapter);

            spinnerKabupaten.setOnItemSelectedListener(new android.widget.AdapterView.OnItemSelectedListener() {
                @Override  no usages
                public void onItemSelected(android.widget.AdapterView<?> parent, android.view.View view,
                                           int position, long id) {
                    Kabupaten selectedKabupaten = selectedProvinsi.kabupatenList.get(position);

                    // Isi Spinner Kecamatan
                    List<String> kecamatanNames = new ArrayList<>();
                    for (Kecamatan kecamatan : selectedKabupaten.kecamatanList) {
                        kecamatanNames.add(kecamatan.namaKecamatan);
                    }
```

```
        ArrayAdapter<String> kecamatanAdapter = new ArrayAdapter<>( context: PendudukTambahActivity.this,
                android.R.layout.simple_spinner_item, kecamatanNames);
        spinnerKecamatan.setAdapter(kecamatanAdapter);

        spinnerKecamatan.setOnItemSelectedListener(new android.widget.AdapterView.OnItemSelectedListener() {
            @Override  no usages
            public void onItemSelected(android.widget.AdapterView<?> parent, android.view.View view,
                                       int position, long id) {
                Kecamatan selectedKecamatan = selectedKabupaten.kecamatanList.get(position);

                // Isi Spinner Desa
                List<String> desaNames = new ArrayList<>();
                for (Desa desa : selectedKecamatan.desaList) {
                    desaNames.add(desa.namaDesa);
                }

                ArrayAdapter<String> desaAdapter = new ArrayAdapter<>( context: PendudukTambahActivity.this,
                        android.R.layout.simple_spinner_item, desaNames);
                spinnerDesa.setAdapter(desaAdapter);
            }

            @Override  no usages
            public void onNothingSelected(android.widget.AdapterView<?> parent) {}
        });
    }

    @Override  no usages
    public void onNothingSelected(android.widget.AdapterView<?> parent) {}
});
}
```

```
    edtName.setVisibility(View.GONE);
    edtPrice.setVisibility(View.GONE);
    edtAlamat.setVisibility(View.GONE);

    edtPendapatan.setInputType(InputType.TYPE_NULL);
    edtPendapatan.setKeyListener(null);

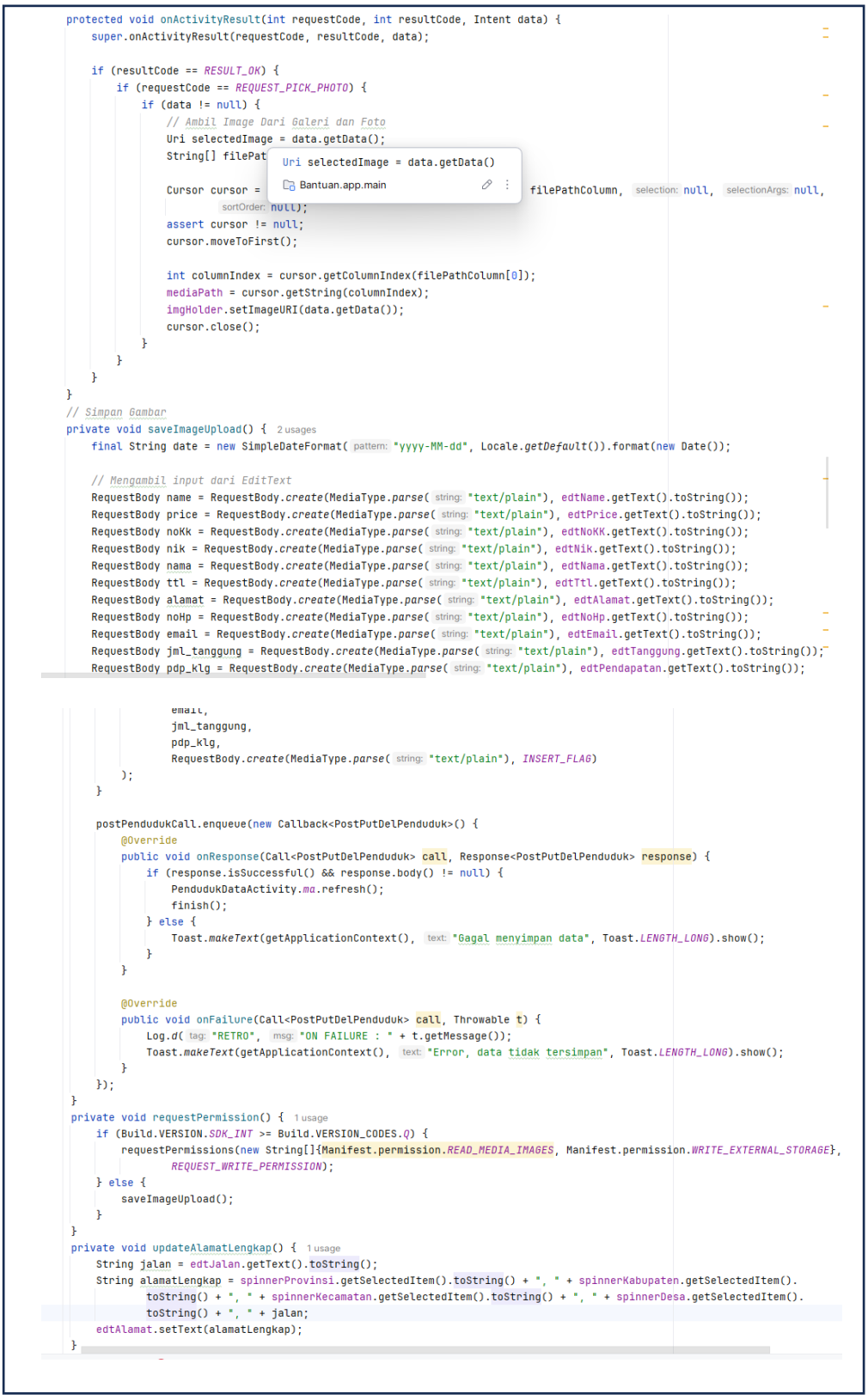
    edtEmail.setInputType(InputType.TYPE_NULL);
    edtEmail.setKeyListener(null);

    edtTanggung.setInputType(InputType.TYPE_NULL);
    edtTanggung.setKeyListener(null);


    // Definisi API
    mApiInterface = ApiClient.getClient().create(ApiInterface.class);

    // Fungsi Tombol Pilih Galery
    btnGalery.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent galleryIntent = new Intent(Intent.ACTION_PICK,
                    android.provider.MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
            startActivityForResult(galleryIntent, REQUEST_PICK_PHOTO);
        }
    });


    // Fungsi Tombol Simpan
    btSubmit.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) { requestPermission(); }
    });
}
```



```
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    if (resultCode == RESULT_OK) {
        if (requestCode == REQUEST_PICK_PHOTO) {
            if (data != null) {
                // Ambil Image Dari Galeri dan Foto
                Uri selectedImage = data.getData();
                String[] filePat
                Uri selectedImage = data.getData()
                Bantuan.app.main
                Cursor cursor =                                  filePathColumn,  selection: null,  selectionArgs: null,
                        sortOrder: null);
                assert cursor != null;
                cursor.moveToFirst();

                int columnIndex = cursor.getColumnIndex(filePathColumn[0]);
                mediaPath = cursor.getString(columnIndex);
                imgHolder.setImageURI(data.getData());
                cursor.close();
            }
        }
    }
}
// Simpan Gambar
private void saveImageUpload() {  2 usages
    final String date = new SimpleDateFormat( pattern: "yyyy-MM-dd", Locale.getDefault()).format(new Date());

    // Mengambil input dari EditText
    RequestBody name = RequestBody.create(MediaType.parse( string: "text/plain"), edtName.getText().toString());
    RequestBody price = RequestBody.create(MediaType.parse( string: "text/plain"), edtPrice.getText().toString());
    RequestBody noKk = RequestBody.create(MediaType.parse( string: "text/plain"), edtNoKK.getText().toString());
    RequestBody nik = RequestBody.create(MediaType.parse( string: "text/plain"), edtNik.getText().toString());
    RequestBody nama = RequestBody.create(MediaType.parse( string: "text/plain"), edtNama.getText().toString());
    RequestBody ttl = RequestBody.create(MediaType.parse( string: "text/plain"), edtTtl.getText().toString());
    RequestBody alamat = RequestBody.create(MediaType.parse( string: "text/plain"), edtAlamat.getText().toString());
    RequestBody noHp = RequestBody.create(MediaType.parse( string: "text/plain"), edtNoHp.getText().toString());
    RequestBody email = RequestBody.create(MediaType.parse( string: "text/plain"), edtEmail.getText().toString());
    RequestBody jml_tanggung = RequestBody.create(MediaType.parse( string: "text/plain"), edtTanggung.getText().toString());
    RequestBody pdp_klg = RequestBody.create(MediaType.parse( string: "text/plain"), edtPendapatan.getText().toString());

                email,
                jml_tanggung,
                pdp_klg,
                RequestBody.create(MediaType.parse( string: "text/plain"), INSERT_FLAG)
        );
    }

    postPendudukCall.enqueue(new Callback<PostPutDelPenduduk>() {
        @Override
        public void onResponse(Call<PostPutDelPenduduk> call, Response<PostPutDelPenduduk> response) {
            if (response.isSuccessful() && response.body() != null) {
                PendudukDataActivity.ma.refresh();
                finish();
            } else {
                Toast.makeText(getApplicationContext(),  text: "Gagal menyimpan data", Toast.LENGTH_LONG).show();
            }
        }

        @Override
        public void onFailure(Call<PostPutDelPenduduk> call, Throwable t) {
            Log.d( tag: "RETRO",  msg: "ON FAILURE : " + t.getMessage());
            Toast.makeText(getApplicationContext(),  text: "Error, data tidak tersimpan", Toast.LENGTH_LONG).show();
        }
    });
}
private void requestPermission() {  1 usage
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.Q) {
        requestPermissions(new String[]{Manifest.permission.READ_MEDIA_IMAGES, Manifest.permission.WRITE_EXTERNAL_STORAGE},
                REQUEST_WRITE_PERMISSION);
    } else {
        saveImageUpload();
    }
}
private void updateAlamatLengkap() {  1 usage
    String jalan = edtJalan.getText().toString();
    String alamatLengkap = spinnerProvinsi.getSelectedItem().toString() + ", " + spinnerKabupaten.getSelectedItem().
            toString() + ", " + spinnerKecamatan.getSelectedItem().toString() + ", " + spinnerDesa.getSelectedItem().
            toString() + ", " + jalan;
    edtAlamat.setText(alamatLengkap);
}
```

Gambar 4. 35 Script Tampilan Hasil Survei

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:orientation="vertical"
    android:padding="16dp"
    tools:context=".PendudukTambahActivity">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_centerInParent="true"
            android:text="Tambah Data Penduduk"
            android:textSize="22sp"
            android:textStyle="bold"/>
    </RelativeLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp">

        <EditText
            android:id="@+id/edt_name"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Username"
            android:inputType="textCapSentences"
            android:maxLines="1" />
    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp">

        <EditText
            android:id="@+id/edt_price"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Password"
            android:inputType="text"
            android:maxLines="1" />
    </com.google.android.material.textfield.TextInputLayout>
```

```
        <com.google.android.material.textfield.TextInputLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginBottom="16dp">

            <ImageView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:scaleType="centerInside"
                android:id="@+id/imgHolder"/>
        </com.google.android.material.textfield.TextInputLayout>

        <Button
            android:id="@+id/btn_galery"
            style="@style/Base.Widget.AppCompat.Button.Colored"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Upload Gambar Rumah"
            android:textColor="@android:color/white"
            android:background="@color/buttonColor"/>

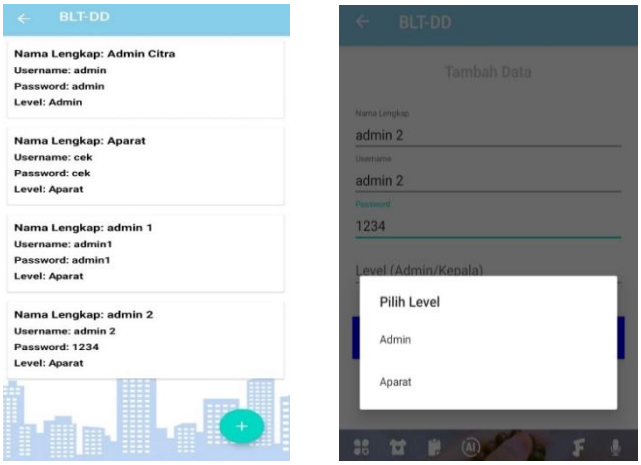
        <Button
            android:id="@+id/btn_submit"
            style="@style/Base.Widget.AppCompat.Button.Colored"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Simpan"
            android:textColor="@android:color/white"
            android:background="@color/buttonColor"
            android:layout_marginTop="10dp"/>

    </LinearLayout>
</ScrollView>
```

Gambar 4. 36 Script Layout Menu Hasil Survei

7. Tampilan menu Data master

Pada gambar di bawah merupakan Gambar menampilkan daftar akun dan form tambah akun di aplikasi BLT-DD, berisi nama, username, password, dan level akses (Admin/Aparat).



Gambar 4. 37 Tampilan Menu Data master

```java
package com.example.bantuan;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import com.example.bantuan.Api.ApiInterface;
import com.example.bantuan.Api.ApiClient;
import com.example.bantuan.Model.User.ResponseModel;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class UserTambahActivity extends AppCompatActivity {
    private EditText etName, etUsername, etLevel, etPassword;  3 usages
    private Button btnSimpan;  2 usages
    private String name, username, level, password;  3 usages

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_user_tambah);

        etName = findViewById(R.id.et_name);
        etUsername = findViewById(R.id.et_username);
        etLevel = findViewById(R.id.et_level);
        etPassword = findViewById(R.id.et_password);
        btnSimpan = findViewById(R.id.btn_simpan);

        // Klik pada etLevel untuk menampilkan pilihan
        etLevel.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) { showLevelOptions(); }
        });

        btnSimpan.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                name = etName.getText().toString();
                username = etUsername.getText().toString();
                level = etLevel.getText().toString();
                password = etPassword.getText().toString();

                if(name.trim().equals("")){
                    etName.setError("Name Harus Diisi");
                }
                else if(username.trim().equals("")){
                    etUsername.setError("Username Harus Diisi");
                }
                else if(level.trim().equals("")){
                    etLevel.setError("Level Harus Diisi");
                }
                else if(password.trim().equals("")){
                    etPassword.setError("Password Harus Diisi");
                }
                else{
                    createData();
                }
            }
        });
    }

    // Metode untuk menampilkan dialog pilihan level
    private void showLevelOptions() {  1 usage
        final String[] levelOptions = {"Admin", "Aparat"};

        android.app.AlertDialog.Builder builder = new android.app.AlertDialog.Builder( context: this);
        builder.setTitle("Pilih Level");
        builder.setItems(levelOptions, (dialog, which) -> {
            etLevel.setText(levelOptions[which]); // Set teks pada EditText
        });
        builder.show();
    }
```

```
    private void createData() {  1 usage
        ApiInterface ardData = ApiClient.getClient().create(ApiInterface.class);
        Call<ResponseModel> simpanData = ardData.ardCreateData(name, username, level, password);

        simpanData.enqueue(new Callback<ResponseModel>() {
            @Override
            public void onResponse(Call<ResponseModel> call, Response<ResponseModel> response) {
                int kode = response.body().getKode();
                String pesan = response.body().getPesan();

                Toast.makeText( context: UserTambahActivity.this,  text: "Pesan : " + pesan, Toast.LENGTH_SHORT).show();
                finish();
            }

            @Override
            public void onFailure(Call<ResponseModel> call, Throwable t) {
                Toast.makeText( context: UserTambahActivity.this,  text: "Gagal Menghubungi Server | " + t.getMessage(),
                        Toast.LENGTH_SHORT).show();
            }
        });
    }
}
```

Gambar 4. 38 Script Java Menu Data Master

```
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Tambah Data"
        android:textSize="22sp"
        android:textStyle="bold"
        android:layout_gravity="center"
        android:layout_marginBottom="30dp"
        />

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <EditText
            android:id="@+id/et_name"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Nama Lengkap"
            android:textSize="20sp"
            android:inputType="text"
            />

    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        >

        <EditText
            android:id="@+id/et_username"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Username"
            android:maxLines="2"
            android:textSize="20sp"
            android:inputType="text"
            />
```

```
<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/et_password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Password"
        android:textSize="20sp"
        android:inputType="text"
        />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    >

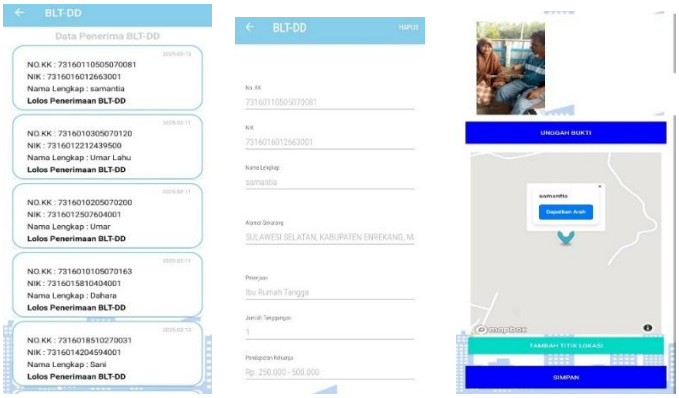
    <EditText
        android:id="@+id/et_level"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Level (Admin/Kepala)"
        android:textSize="20sp"
        android:inputType="text"
        android:focusable="false"
        />

</com.google.android.material.textfield.TextInputLayout>
```

Gambar 4. 39 Script Layout Menu Data Master

8. Data penerima BLT-DD

Pada gambar dibawah Gambar menunjukkan data penerima BLT-DD dalam aplikasi, mencakup daftar penerima, detail pribadi, dan lokasi penerima. Aplikasi ini digunakan untuk mendata, memverifikasi, dan memetakan penerima bantuan secara digital.



Gambar 4. 40 Tampilan Menu Data Penerima BLT-DD

```java
public class AdapterDataPenduduk extends RecyclerView.Adapter<AdapterDataPenduduk.MyViewHolder> {  3 usages
    List<Penduduk> mPendudukList;  36 usages

    public AdapterDataPenduduk(List<Penduduk> PendudukList) { mPendudukList = PendudukList; }

    @Override
    public MyViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
        View mView = LayoutInflater.from(parent.getContext()).inflate(R.layout.card_item_penduduk, parent,  attachToRoot: false);
        MyViewHolder mViewHolder = new MyViewHolder(mView);
        return mViewHolder;
    }

    @Override
    public void onBindViewHolder(MyViewHolder holder, @SuppressLint("RecyclerView") final int position) {

        SpannableString spannableNoKK = new SpannableString( source: "NO.KK : " + mPendudukList.get(position).getNo_kk());
        holder.mTextViewNoKK.setText(spannableNoKK);
        SpannableString spannableNik = new SpannableString( source: "NIK : " + mPendudukList.get(position).getNik());
        holder.mTextViewNik.setText(spannableNik);
        SpannableString spannableNama = new SpannableString( source: "Nama Lengkap : " + mPendudukList.get(position).getNama());
        holder.mTextViewNama.setText(spannableNama);

        // Set TextViews dengan data Penduduk
        holder.mTextViewName.setText(mPendudukList.get(position).getName());
        holder.mTextViewDate.setText(mPendudukList.get(position).getDate());
        holder.mTextViewTtl.setText(mPendudukList.get(position).getTtl());
        holder.mTextViewAlamat.setText(mPendudukList.get(position).getAlamat());
        holder.mTextViewNoHp.setText(mPendudukList.get(position).getNo_hp());
        holder.mTextViewEmail.setText(mPendudukList.get(position).getEmail());
        holder.mTextViewValidasi.setText(mPendudukList.get(position).getValidasi());

        holder.mTextViewPrice.setText(mPendudukList.get(position).getPrice());

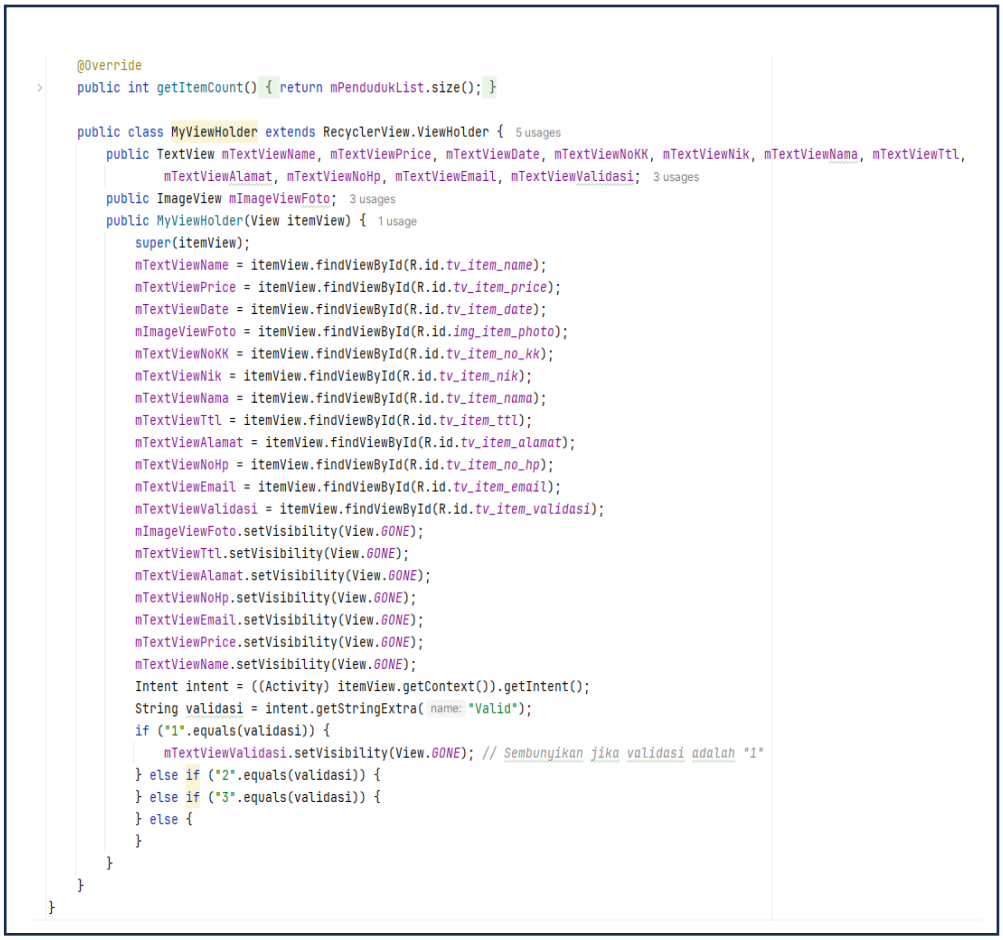
        // Load image menggunakan Glide
        Glide.with(holder.itemView.getContext()) RequestManager
                .load( string: Config.IMAGES_URL + mPendudukList.get(position).getImage()) RequestBuilder<Drawable>
                .apply(new RequestOptions().override( width: 350,  height: 350))
                .into(holder.mImageViewFoto);
```

```
holder.itemView.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent mIntent = new Intent(view.getContext(), PendudukUbahActivity.class);

        // Kirim data Penduduk ke Activity berikutnya
        mIntent.putExtra( name: "Id", mPendudukList.get(position).getId());
        mIntent.putExtra( name: "Name", mPendudukList.get(position).getName());
        mIntent.putExtra( name: "Price", mPendudukList.get(position).getPrice());
        mIntent.putExtra( name: "Date", mPendudukList.get(position).getDate());
        mIntent.putExtra( name: "Image", mPendudukList.get(position).getImage());
        mIntent.putExtra( name: "Image1", mPendudukList.get(position).getImage1());
        mIntent.putExtra( name: "Image2", mPendudukList.get(position).getImage2());
        mIntent.putExtra( name: "Image3", mPendudukList.get(position).getImage3());
        mIntent.putExtra( name: "Image4", mPendudukList.get(position).getImage4());
        mIntent.putExtra( name: "Image5", mPendudukList.get(position).getImage5());
        mIntent.putExtra( name: "Image6", mPendudukList.get(position).getImage6());
        mIntent.putExtra( name: "Image7", mPendudukList.get(position).getImage7());
        mIntent.putExtra( name: "NoKK", mPendudukList.get(position).getNo_kk());
        mIntent.putExtra( name: "Nik", mPendudukList.get(position).getNik());
        mIntent.putExtra( name: "Nama", mPendudukList.get(position).getNama());
        mIntent.putExtra( name: "Ttl", mPendudukList.get(position).getTtl());
        mIntent.putExtra( name: "Alamat", mPendudukList.get(position).getAlamat());
        mIntent.putExtra( name: "NoHp", mPendudukList.get(position).getNo_hp());
        mIntent.putExtra( name: "Email", mPendudukList.get(position).getEmail());
        mIntent.putExtra( name: "JmlTanggung", mPendudukList.get(position).getJml_tanggung());
        mIntent.putExtra( name: "PdpKlg", mPendudukList.get(position).getPdp_klg());
        mIntent.putExtra( name: "Validasi", mPendudukList.get(position).getValidasi());


        // Kirim nilai Validasi dari intent
        Intent currentIntent = ((PendudukDataActivity) view.getContext()).getIntent();
        String Validasi = currentIntent.getStringExtra( name: "Valid");
        mIntent.putExtra( name: "Valid", Validasi);

        view.getContext().startActivity(mIntent);
    }
});
```



```
    @Override
    public int getItemCount() { return mPendudukList.size(); }

    public class MyViewHolder extends RecyclerView.ViewHolder {  5 usages
        public TextView mTextViewName, mTextViewPrice, mTextViewDate, mTextViewNoKK, mTextViewNik, mTextViewNama, mTextViewTtl,
                mTextViewAlamat, mTextViewNoHp, mTextViewEmail, mTextViewValidasi;  3 usages
        public ImageView mImageViewFoto;  3 usages
        public MyViewHolder(View itemView) {  1 usage
            super(itemView);
            mTextViewName = itemView.findViewById(R.id.tv_item_name);
            mTextViewPrice = itemView.findViewById(R.id.tv_item_price);
            mTextViewDate = itemView.findViewById(R.id.tv_item_date);
            mImageViewFoto = itemView.findViewById(R.id.img_item_photo);
            mTextViewNoKK = itemView.findViewById(R.id.tv_item_no_kk);
            mTextViewNik = itemView.findViewById(R.id.tv_item_nik);
            mTextViewNama = itemView.findViewById(R.id.tv_item_nama);
            mTextViewTtl = itemView.findViewById(R.id.tv_item_ttl);
            mTextViewAlamat = itemView.findViewById(R.id.tv_item_alamat);
            mTextViewNoHp = itemView.findViewById(R.id.tv_item_no_hp);
            mTextViewEmail = itemView.findViewById(R.id.tv_item_email);
            mTextViewValidasi = itemView.findViewById(R.id.tv_item_validasi);
            mImageViewFoto.setVisibility(View.GONE);
            mTextViewTtl.setVisibility(View.GONE);
            mTextViewAlamat.setVisibility(View.GONE);
            mTextViewNoHp.setVisibility(View.GONE);
            mTextViewEmail.setVisibility(View.GONE);
            mTextViewPrice.setVisibility(View.GONE);
            mTextViewName.setVisibility(View.GONE);
            Intent intent = ((Activity) itemView.getContext()).getIntent();
            String validasi = intent.getStringExtra( name: "Valid");
            if ("1".equals(validasi)) {
                mTextViewValidasi.setVisibility(View.GONE); // Sembunyikan jika validasi adalah "1"
            } else if ("2".equals(validasi)) {
            } else if ("3".equals(validasi)) {
            } else {
            }
        }
    }
}
```

Gambar 4. 41 Script Java menu penerima

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    xmlns:app="http://schemas.a-droid.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_marginHorizontal="16dp"
    android:layout_marginTop="5dp"
    android:orientation="vertical">

    <androidx.cardview.widget.CardView
        android:id="@+id/cv_item_Penduduk"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginVertical="1dp"
        app:cardCornerRadius="8dp"
        app:cardElevation="1dp">

        <androidx.constraintlayout.widget.ConstraintLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@drawable/rounded_button"
            android:padding="10dp">

            <ImageView
                android:id="@+id/img_item_photo"
                android:layout_width="0dp"
                android:layout_height="0dp"
                android:scaleType="centerCrop"
                app:layout_constraintDimensionRatio="1:1"
                app:layout_constraintStart_toStartOf="parent"
                app:layout_constraintTop_toTopOf="parent"
                app:layout_constraintBottom_toBottomOf="parent" />


        <TextView
            android:id="@+id/tv_item_date"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginEnd="10dp"
            android:textColor="@android:color/darker_gray"
            android:textSize="12sp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            tools:text="tanggal" />

        <TextView
            android:id="@+id/tv_item_name"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:layout_marginTop="4dp"
            android:textColor="@android:color/black"
            android:textSize="20sp"
            android:textStyle="bold"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@id/tv_item_date"
            tools:text="name" />
```

```
    <TextView
        android:id="@+id/tv_item_price"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="4dp"
        android:textColor="@android:color/black"
        android:textSize="14sp"
        android:ellipsize="end"
        android:maxLines="1"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/tv_item_name"
        tools:text="harga" />

    <!-- Informasi Penduduk -->
    <TextView
        android:id="@+id/tv_item_no_kk"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="4dp"
        android:textColor="@android:color/black"
        android:textSize="18sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/tv_item_price"
        tools:text="No. KK" />


    <TextView
        android:id="@+id/tv_item_nik"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="4dp"
        android:textColor="@android:color/black"
        android:textSize="18sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/tv_item_no_kk"
        tools:text="NIK" />

    <TextView
        android:id="@+id/tv_item_nama"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="4dp"
        android:textColor="@android:color/black"
        android:textSize="18sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/tv_item_nik"
        tools:text="Nama" />
```

```
        <TextView
            android:id="@+id/tv_item_ttl"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:layout_marginTop="4dp"
            android:textColor="@android:color/black"
            android:textSize="14sp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@id/tv_item_nama"
            tools:text="TTL" />
        <TextView
            android:id="@+id/tv_item_alamat"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:layout_marginTop="4dp"
            android:textColor="@android:color/black"
            android:textSize="14sp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@id/tv_item_ttl"
            tools:text="Alamat" />
        <TextView
            android:id="@+id/tv_item_no_hp"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:layout_marginTop="4dp"
            android:textColor="@android:color/black"
            android:textSize="14sp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@id/tv_item_alamat"
            tools:text="No. HP" />

        <TextView
            android:id="@+id/tv_item_email"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:layout_marginTop="4dp"
            android:textColor="@android:color/black"
            android:textSize="14sp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@id/tv_item_no_hp"
            tools:text="Email" />


        <TextView
            android:id="@+id/tv_item_validasi"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:layout_marginTop="4dp"
            android:textColor="@android:color/black"
            android:textSize="18sp"
            android:textStyle="bold"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@id/tv_item_email"
            tools:text="Validasi" />

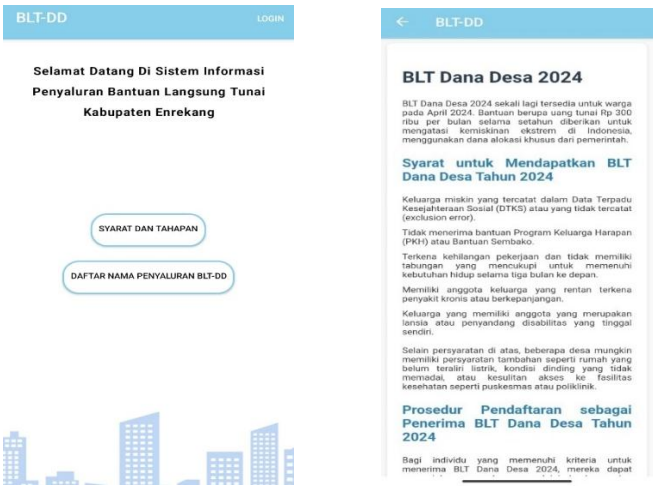
    </androidx.constraintlayout.widget.ConstraintLayout>
</androidx.cardview.widget.CardView>


</LinearLayout>
```

Gambar 4. 42 Script Loyout Data Penerima

9. Tampilan Menu Syarat dan tahapan

Pada gambar dibawa dibawa menjelaskan syarat dan tahapan BLT Dana Desa 2024



Gambar 4. 43 Tampilan Menu Syarat dan Tahapan

```
package com.example.bantuan;

import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;

public class SisfoActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_sisfo);

        Button syaratButton = findViewById(R.id.syarat);
        Button daftarButton = findViewById(R.id.daftar);

        syaratButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent( packageContext: SisfoActivity.this, TambahLokasiActivity.class);
                String Type = "tahapan";
                intent.putExtra( name: "Type", Type);
                startActivity(intent);
            }
        });
        daftarButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent( packageContext: SisfoActivity.this, TambahLokasiActivity.class);
                String Type = "sisfo";
                intent.putExtra( name: "Type", Type);
                startActivity(intent);
            }
        });

    }
```

```
    @Override  7 usages
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu_login, menu);
        return super.onCreateOptionsMenu(menu);
    }

    @Override  8 usages
    public boolean onOptionsItemSelected(MenuItem item) {
        if (item.getItemId() == R.id.actionLogin) {
            // Tambahkan logika ketika menu "Login" diklik
            Intent intent = new Intent( packageContext: SisfoActivity.this, LoginActivity.class);
            // Ganti LoginActivity dengan activity tujuan Anda
            startActivity(intent); // Jalankan activity
            return true; // Indikasikan bahwa event telah ditangani
        }
        return super.onOptionsItemSelected(item);
    }
}
```

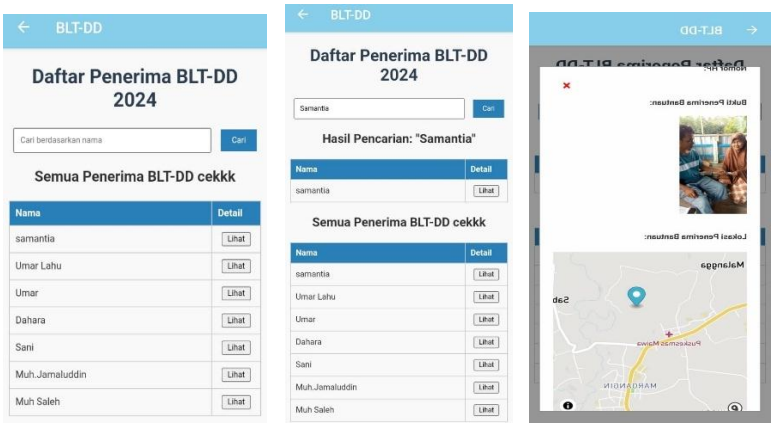
```
<TextView
    android:id="@+id/welcomeText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Selamat Datang Di Sistem Informasi Penyaluran
     Bantuan Langsung Tunai Kabupaten Enrekang"
    android:textSize="20sp"
    android:textStyle="bold"
    android:padding="16dp"
    android:gravity="center"
    android:lineSpacingExtra="8dp"
    android:letterSpacing="0.05"
    android:layout_marginTop="20dp"/>

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <Button
        android:id="@+id/syarat"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Syarat dan tahapan"
        android:layout_centerHorizontal="true"
        android:background="@drawable/rounded_button"
        android:textColor="@color/hitam"
        android:layout_marginTop="250dp"
        android:padding="10dp"/>
    <Button
        android:id="@+id/daftar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Daftar Nama Penyaluran BLT-DD"
        android:layout_centerHorizontal="true"
        android:layout_below="@id/syarat"
        android:background="@drawable/rounded_button"
        android:textColor="@color/hitam"
        android:layout_marginTop="20dp"
        android:padding="10dp"/>
```

Gambar 4. 44 Script Menu syarat Dan Tahapan

10. Tampilan Menu Daftar nama penyaluran BLT-DD

Pada gambar dibawah menjelaskan tentang daftar penerima BLT



Gambar 4. 45 Tampilan Menu Daftar Nama Penyaluran BLT-DD

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_tambah_lokasi);
    // Inisialisasi WebView
    webView = findViewById(R.id.webView);
    // Aktifkan JavaScript jika dibutuhkan
    WebSettings webSettings = webView.getSettings();
    webSettings.setJavaScriptEnabled(true);
    // Pastikan WebView tetap di aplikasi (tanpa membuka browser ek
    webView.setWebViewClient(new WebViewClient());
    // Muat URL yang ingin ditampilkan dengan ID dari intent
    String id = getIntent().getStringExtra( name: "Id");
    String Type = getIntent().getStringExtra( name: "Type"); // Mena
    // Cek kondisi untuk mengubah URL
    String url;
    if ("sisfo".equals(Type)) {
        url = "https://bantuan.maiwa.id/sisfo.php"; // Kondisi pert
        // Mengaktifkan tombol kembali
        if (getSupportActionBar() != null) {
            getSupportActionBar().setDisplayHomeAsUpEnabled(true);
            getSupportActionBar().setDisplayShowHomeEnabled(true);
        }
    } else if ("tahapan".equals(Type)) {
        url = "https://bantuan.maiwa.id/tahapan.php"; // Kondisi ke
```



```
            // Mengaktifkan tombol kembali
            if (getSupportActionBar() != null) {
                getSupportActionBar().setDisplayHomeAsUpEnabled(true);
                getSupportActionBar().setDisplayShowHomeEnabled(true);
            }
        } else {
            if (getSupportActionBar() != null) {
                getSupportActionBar().setDisplayHomeAsUpEnabled(true);
                getSupportActionBar().setDisplayShowHomeEnabled(true);
            }
            url = "https://bantuan.maiwa.id/tambahlokasi.php?id=" + id;
        }
        webView.loadUrl(url);
    }

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    if (item.getItemId() == android.R.id.home) {
        String type = getIntent().getStringExtra( name: "Type"); // Ambil kembali nilai Type
        if ("sisfo".equals(type)) {
            // Arahkan ke halaman Sisfo
            Intent intent = new Intent( packageContext: this, MainActivity.class);
            startActivity(intent);
        } else if ("tahapan".equals(type)) {
            // Arahkan ke halaman Tahapan
            Intent intent = new Intent( packageContext: this, MainActivity.class);
            startActivity(intent);
        } else {
            finish(); // Tutup aktivitas saat ini dan kembali ke aktivitas sebelumnya
        }
        finish(); // Tutup aktivitas saat ini
        return true;
    }
    return super.onOptionsItemSelected(item);
}
```

Gambar 4. 46 Script Menu Daftar Nama Penerima

D. Pengujian Sistem

Penelitian ini menggunakan dua metode pengujian sistem, yaitu pengujian *Black Box* dan *White Box,*. Berikut ini adalah hasil pengujian yang diperoleh melalui pendekatan-pendekatan tersebut.

1. *Black Box Testing*

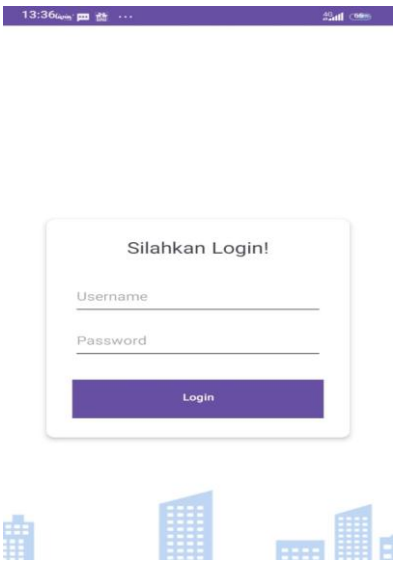
a. *Black Box Testing* Halaman Menu Awal

Tabel 4. 5 *Black Box Testing* untuk Halaman Menu Awal

Tes Faktor	Hasil	Keterangan
Admin pertama kali menjalankan aplikasi	✓	Berhasil tampil halaman splash screen menu awal
Screenshot		
 		

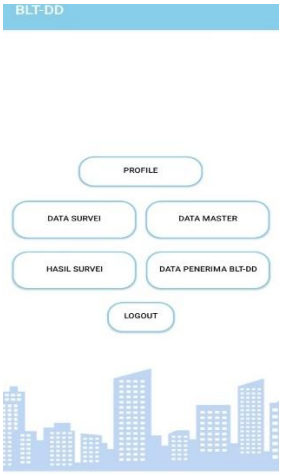
b. *Black Box Testing* Halaman Login

Tabel 4. 6 *Black Box Testing* Untuk Halaman Tampilan Login

Tes Faktor	Hasil	Keterangan
Admin memasukkan username dan password	✓	Berhasil tampil halaman login
Screenshot		
		

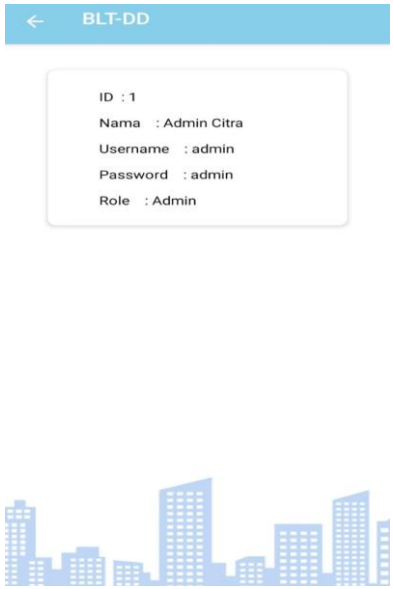
c. *Black Box Testing* Halaman Menu Awal

Tabel 4. 7 *Black Box Testing* Untuk Halaman Menu Utama Login

Tes Faktor	Hasil	Keterangan
Admin memilih menu utama login	✓	Berhasil tampil menu utama login
Screenshot		
		

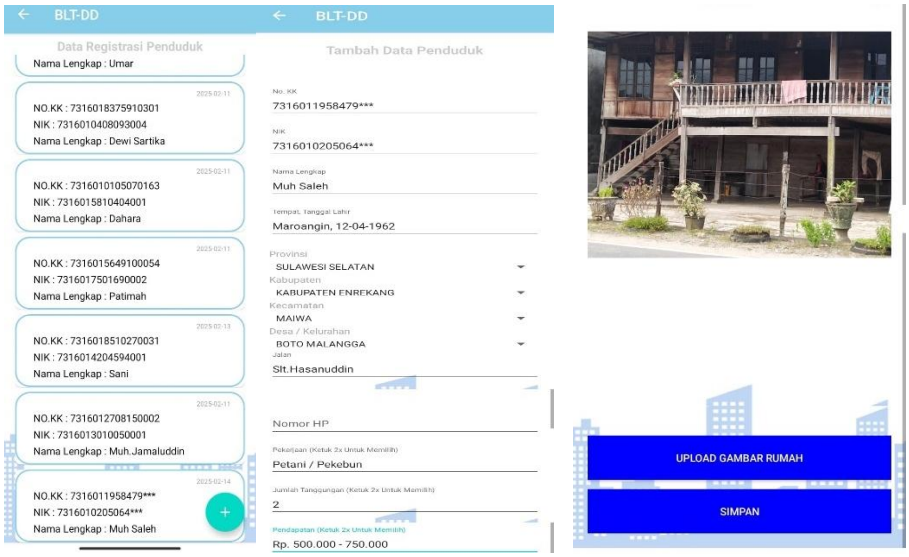
d. *Black Box Testing* Halaman Menu Profile

Tabel 4. 8 *Black Box Testing* Untuk Halaman Menu Profile

Tes Faktor	Hasil	Keterangan
Admin memilih menu profile	✓	Berhasil tampil menu profile
Screenshot		
		

e. *Black Box Testing* Halaman Menu Data Survei

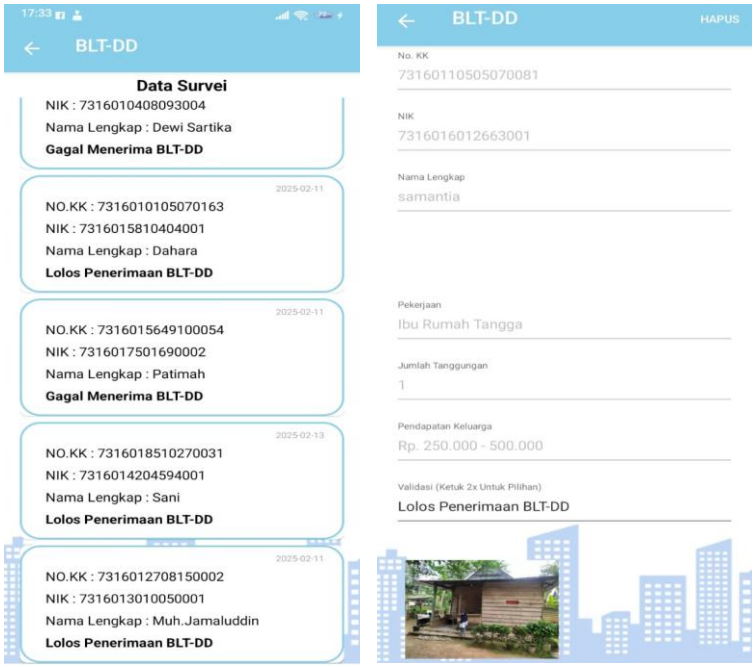
Tabel 4. 9 *Black Box Testing* Untuk Halaman Tampilan Data Survei

Tes Faktor	Hasil	Keterangan
Admin memilih menu data survei dan menambahkan data penduduk serta menyimpan data	✓	Berhasil tampil menu data survei dan menambahkan data registrasi penduduk
Screenshot		
		

f. *Black Box Testing* Halaman Menu Hasil Survei

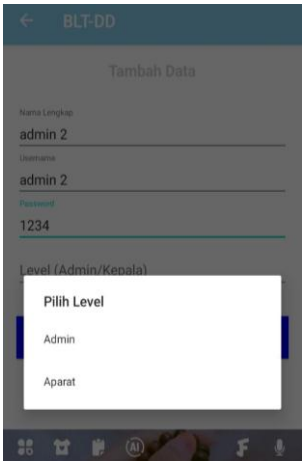
Tabel 4. 10 *Black Box Testing* Untuk Halaman Menu Hasil Survei

Tes Faktor	Hasil	Keterangan
Admin memilih menu hasil survei	✓	Berhasil tampil menu hasil survei yang di input
Screenshot		



g. *Black Box Testing* Halaman Menu Data Master

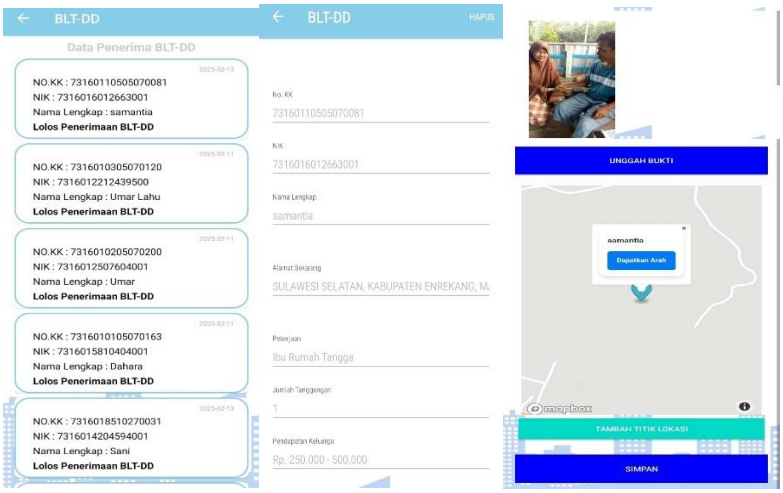
Tabel 4. 11 *Black Box Testing* untuk Halaman Menu Data Master

Tes Faktor	Hasil	Keterangan
Admin memilih menu data master dan menambahkan admin selanjutnya	✓	Berhasil tampil menu data master yang telah di tambahkan
Screenshot		
 		

h. *Black Box Testing* Halaman Menu data Penerima BLTDD

Tabel 4. 12 *Black Box Testing* untuk Halaman Menudata Penerima BLT-DD

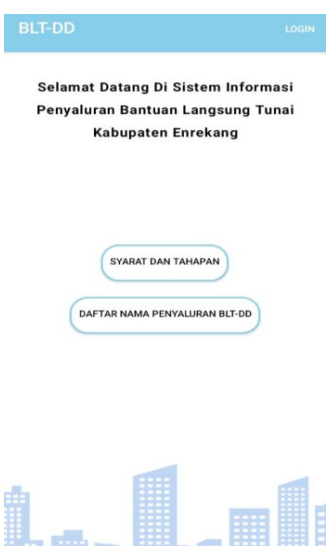
Tes Faktor	Hasil	Keterangan
Admin memilih menu data penerima BLT-DD	✓	Berhasil tampil menu data penerima BLT-DD dan mengunggah bukti penyaluran BLT-DD
Screenshot		



i. *Black Box Testing* Halaman Menu Syarat dan Tahapan

Tabel 4. 13 *Black Box Testing* untuk Halaman Menu Syarat dan Tahapan

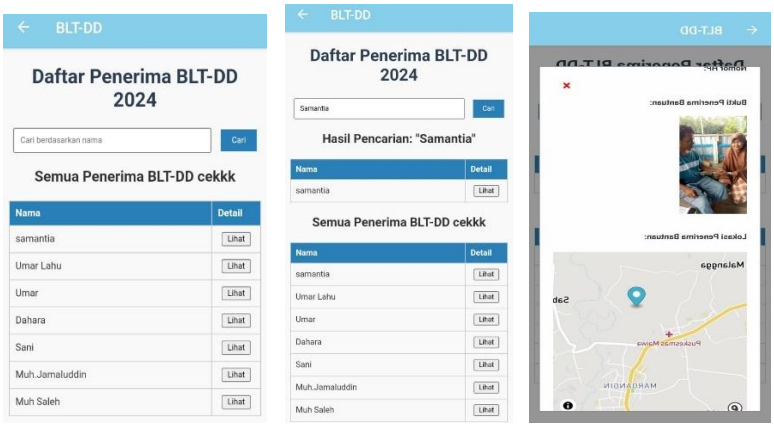
Tes Faktor	Hasil	Keterangan
User memilih menu syarat dan tahapan	✓	Berhasil menampilkan syarat dan tahapan
Screenshot		





j. *Black Box Testing* Halaman Menu daftar nama penyaluran BLT-DD

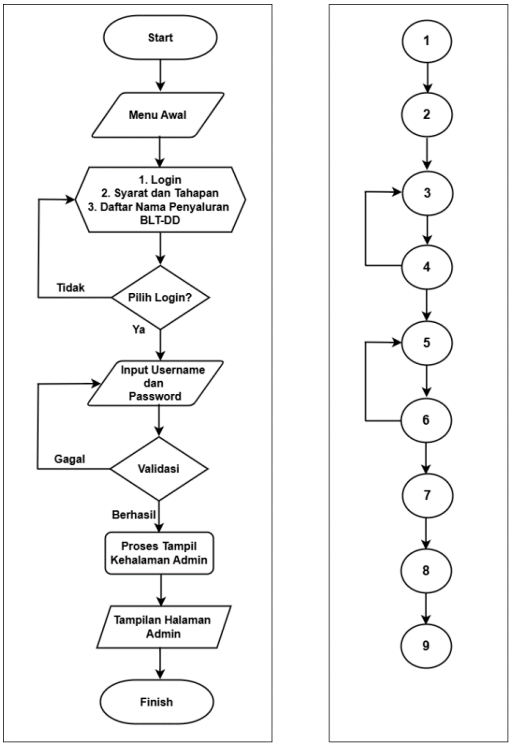
Tabel 4. 14 *Black Box Testing* untuk Halaman Menu Daftar Nama Penyaluran BLT-DD

Tes Faktor	Hasil	Keterangan
User memilih menu daftar nama penyaluran BLT-DD dan pencarian nama penerima	✓	Berhasil tampil menu daftar semua nama penerima BLT-DD dan menampilkan hasil data diri yang telah menerima bantuan
Screenshot		
		

2. *White Box Testing*

Dalam pengujian ini akan menampilkan *flowchart* dan *flowgraph* dari sistem yang telah dibuat. Berikut ini adalah hasil dari pengujian menggunakan *white box testing* secara keseluruhan.

a. *Flowchart* dan *Flowgraph* Menu Login



Gambar 4. 47 *Flowchart* dan *Flowgraph* Utama Login

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau *Flowchart* dan *Flowgraph* di atas:

1). Menghitung kompleksitas *cyclomatic* V(G) pada *egde* dan *node*

Pada rumus: V(G) = E – N + 2

E (*edge*) = 10

N (*node*) = 9

P (Predikat *node*) = 2

Penyelesaian:

V(G) = E – N + 2

= 10 – 9 + 2

= 3

Predikat (P) = P + 1

= 2 + 1

= 3

2). Berdasarkan perhitungan kompleksitas *cyclomatic* dari *flowgraph* di atas, memiliki = 3

3). Jalan independen pada *flowgraph* ini adalah:

Path 1 = 1 – 2 – 3 – 4 – 3

Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 5

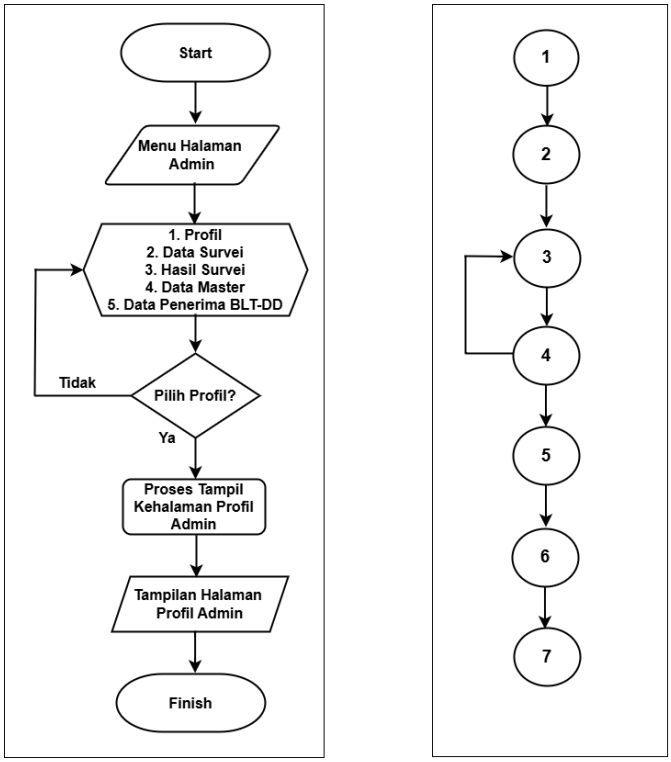
Path 3 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9

4). Grafik matriks menu login

Tabel 4. 15 Grafik matriks menu login

	1	2	3	4	5	6	7	8	9	E - 1
1		1								1 - 1 = 0
2			1							1 - 1 = 0
3				1						1 - 1 = 0
4			1		1					2 - 1 = 1
5						1				1 - 1 = 0
6					1		1			2 - 1 = 1
7								1		1 - 1 = 0
8									1	1 - 1 = 0
9										1 - 1 = 0
	SUM (E + 1)									2 + 1 = 3

b. *Flowchart* dan *Flowgraph* Menu Profile



Gambar 4. 48 *Flowchart* dan *Flowgraph* Utama Profile

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau *Flowchar*t dan *Flowgraph* di atas:

1). Menghitung kompleksitas *cyclomatic* V(G) pada *egde* dan *node*

Pada rumus : V(G) = E – N + 2

E (*edge*) = 7

N (*node*) = 7

P (Predikat *node*) = 1

Penyelesaian :

V(G) = E – N + 2

= 7 – 7 + 2

= 2

Predikat (P) = P + 1

= 1 + 1

= 2

2). Berdasarkan perhitungan kompleksitas *cyclomatic* dari *flowgraph* di atas, memiliki = 2

3). Jalan independen pada *flowgraph* ini adalah:

Path 1 = 1 – 2 – 3 – 4 – 3

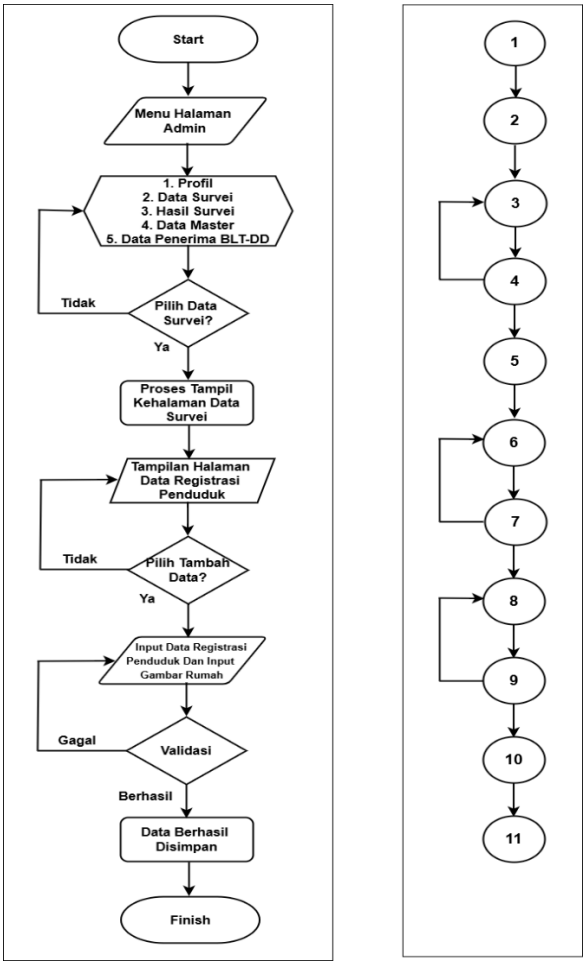
Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 7

4).Grafik matriks menu menu profile

Tabel 4. 16 Grafik matriks menu Profile

	1	2	3	4	5	6	7	E - 1
1		1						1 - 1 = 0
2			1					1 - 1 = 0
3				1				1 - 1 = 0
4			1		1			2 - 1 = 1
5						1		1 - 1 = 0
6							1	1 - 1 = 0
7								
	SUM (E + 1)							1 + 1 = 2

c. *Flowchart* dan *Flowgraph* Menu Data Survei



Gambar 4. 49 *Flowchart* dan *Flowgraph* Menu Data Survei

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau *Flowchart* dan *Flowgraph* di atas:

1). Menghitung kompleksitas *cyclomatic* V(G) pada *egde* dan *node*

Pada rumus : V(G) = E – N + 2

E (*edge*) = 13

N (*node*) = 11

P (Predikat *node*) = 3

Penyelesaian :

V(G) = E – N + 2

= 13 – 11 + 2

= 4

Predikat (P) = P + 1

= 3 + 1

= 4

2). Berdasarkan perhitungan kompleksitas *cyclomatic* dari *flowgraph* di atas, memiliki = 4

3). Jalan independen pada *flowgraph* ini adalah:

Path 1 = 1 – 2 – 3 – 4 – 3

Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 6

Path 3 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 8

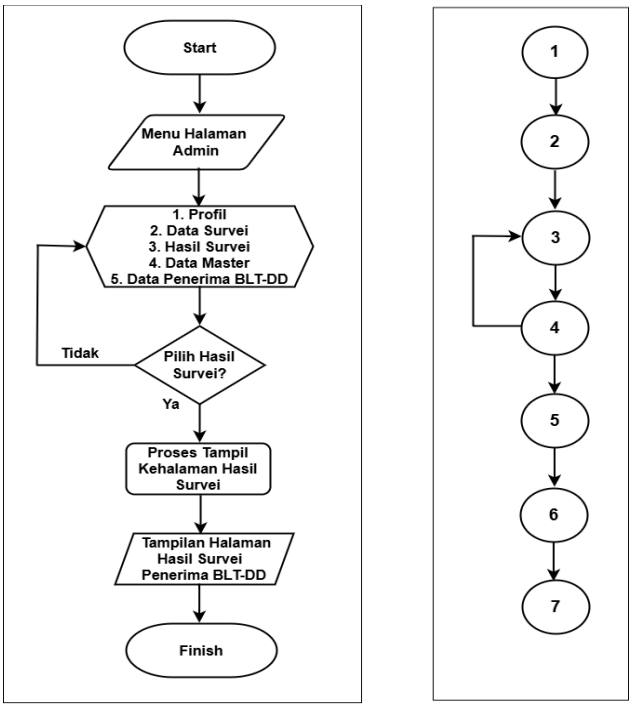
Path 3 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11

4). Grafik matriks menu data survei

Tabel 4. 17 Grafik matriks menu data survei

	1	2	3	4	5	6	7	8	9	10	11	E - 1
1		1										1 - 1 = 0
2			1									1 - 1 = 0
3		1		1								2 - 1 = 1
4					1							1 - 1 = 0
5						1						1 - 1 = 0
6					1		1					2 - 1 = 1
7								1				1 - 1 = 0
8									1			1 - 1 = 0
9								1		1		2 - 1 = 1
10											1	1 - 1 = 0
11												
	SUM (E + 1)											3 + 1 = 4

d. *Flowchart* dan *Flowgraph* Menu Hasil Survei



Gambar 4. 50 *Flowchart* dan *Flowgraph* Menu Hasil Survei

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau *Flowchar*t dan *Flowgraph* di atas:

1). Menghitung kompleksitas *cyclomatic* V(G) pada *egde* dan *node*

Pada rumus : V(G) = E – N + 2

E (*edge*) = 7

N (*node*) = 7

P (Predikat *node*) = 1

Penyelesaian :

V(G) = E – N + 2

= 7 – 7 + 2

= 2

Predikat (P) = P + 1

= 1 + 1

= 2

2). Berdasarkan perhitungan kompleksitas *cyclomatic* dari *flowgraph* di atas, memiliki = 2

3). Jalan independen pada *flowgraph* ini adalah:

Path 1 = 1 – 2 – 3 – 4 – 3

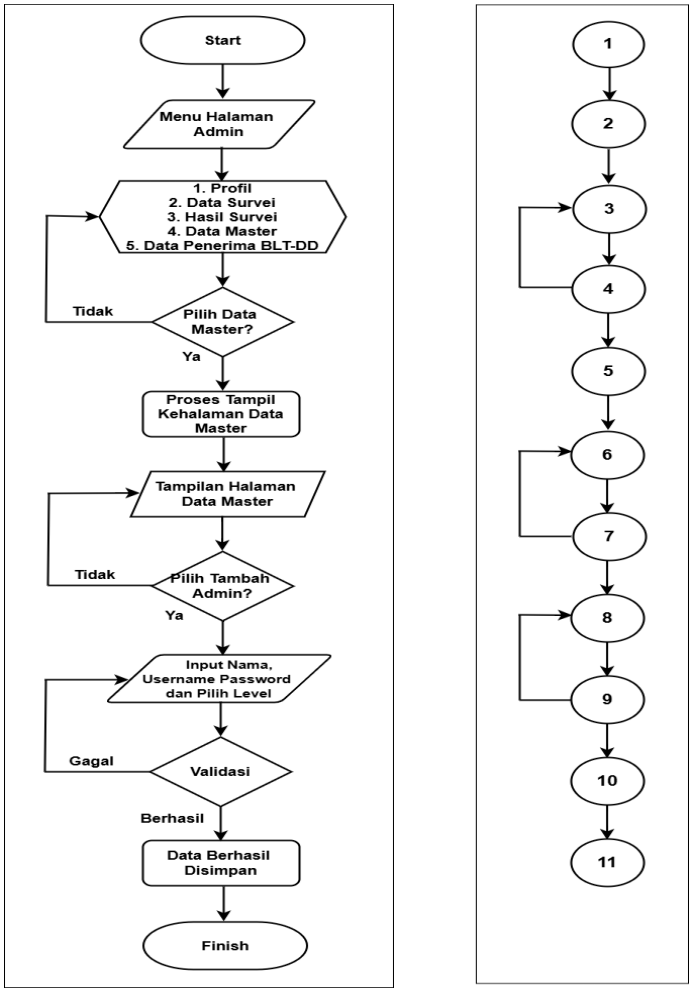
Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 7

4).Grafik matriks menu hasil survei

Tabel 4. 18 Grafik matriks menu hasil survei

	1		2	3	4	5	6	7	E - 1
1			1						1 - 1 = 0
2				1					1 - 1 = 0
3					1				1 - 1 = 0
4				1		1			2 - 1 = 1
5							1		1 - 1 = 0
6								1	1 - 1 = 0
7									0
		SUM (E + 1)							1 + 1 = 2

e. *Flowchart* dan *Flowgraph* Menu Data Master



Gambar 4. 51 *Flowchart* dan *Flowgraph* Menu Data Master

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau *Flowchart* dan *Flowgraph* di atas:

1). Menghitung kompleksitas *cyclomatic* V(G) pada *egde* dan *node*

Pada rumus : V(G) = E – N + 2

E (*edge*) = 13

N (*node*) = 11

P (Predikat *node*) = 3

Penyelesaian :

V(G) = E – N + 2

= 13 – 11 + 2

= 4

Predikat (P) = P + 1

= 3 + 1

= 4

2). Berdasarkan perhitungan kompleksitas *cyclomatic* dari *flowgraph* di atas, memiliki = 4

3). Jalan independen pada *flowgraph* ini adalah:

Path 1 = 1 – 2 – 3 – 4 – 3

Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 6

Path 3 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 8

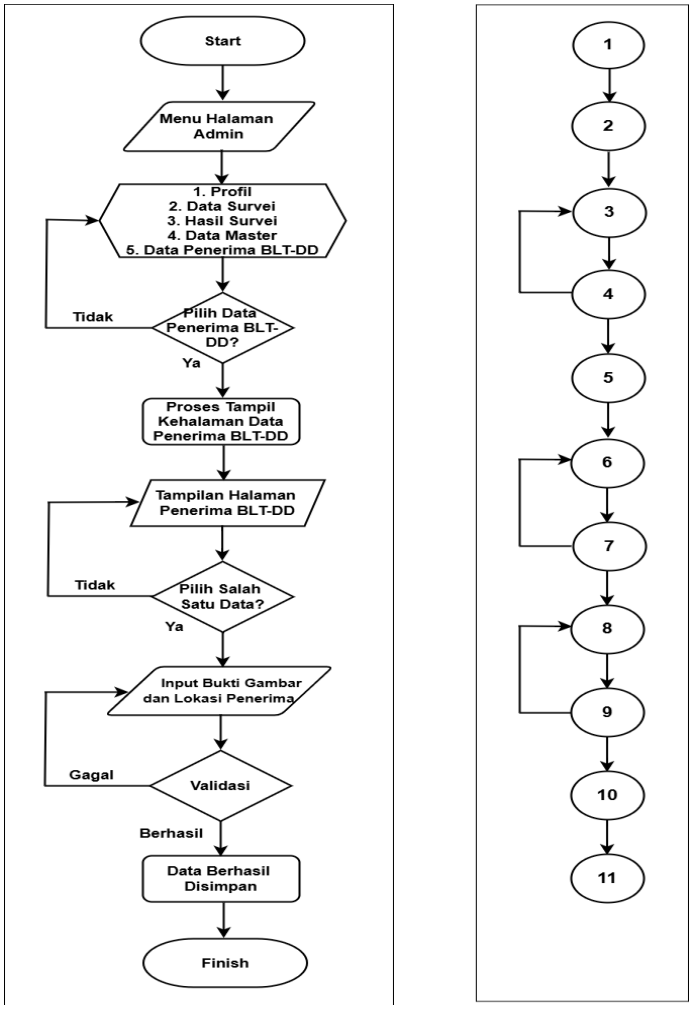
Path 3 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11

4).Grafik matriks menu data master

Tabel 4. 19 Grafik matriks menu data master

	1	2	3	4	5	6	7	8	9	10	11	E - 1
1		1										1 - 1 = 0
2			1									1 - 1 = 0
3		1		1								2 - 1 = 1
4					1							1 - 1 = 0
5						1						1 - 1 = 0
6					1		1					2 - 1 = 1
7								1				1 - 1 = 0
8									1			1 - 1 = 0
9								1		1		2 - 1 = 1
10											1	1 - 1 = 0
11												
	SUM (E + 1)											3 + 1 = 4

e. *Flowchart* dan *Flowgraph* Menu Data Penerima BLT-DD



Gambar 4. 52 *Flowchart* dan *Flowgraph* Menu Data Penerima BLT-DD

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau *Flowchart* dan *Flowgraph* di atas:

1). Menghitung kompleksitas *cyclomatic* V(G) pada *egde* dan *node*

Pada rumus : V(G) = E – N + 2

E (*edge*) = 13

N (*node*) = 11

P (Predikat *node*) = 3

Penyelesaian :

V(G) = E – N + 2

= 13 – 11 + 2

= 4

Predikat (P) = P + 1

= 3 + 1

= 4

2). Berdasarkan perhitungan kompleksitas *cyclomatic* dari *flowgraph* di atas, memiliki = 4

3). Jalan independen pada *flowgraph* ini adalah:

Path 1 = 1 – 2 – 3 – 4 – 3

Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 6

Path 3 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 8

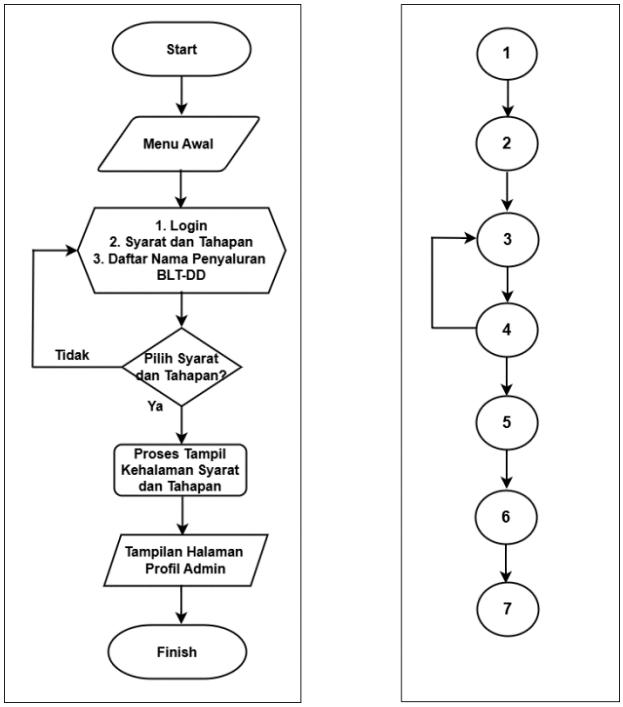
Path 3 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11

4).Grafik matriks menu data penerima BLT-DD

Tabel 4. 20 Grafik matriks menu data penerima BLT-DD

	1	2	3	4	5	6	7	8	9	10	11	E - 1
1		1										1 - 1 = 0
2			1									1 - 1 = 0
3		1		1								2 - 1 = 1
4					1							1 - 1 = 0
5						1						1 - 1 = 0
6					1		1					2 - 1 = 1
7								1				1 - 1 = 0
8									1			1 - 1 = 0
9								1		1		2 - 1 = 1
10											1	1 - 1 = 0
11												
	SUM (E + 1)											3 + 1 = 4

e. *Flowchart* dan *Flowgraph* Menu Syarat dan Tahapan



Gambar 4. 53 *Flowchart* dan *Flowgraph* Menu Syarat dan Tahapan

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau *Flowchar*t dan *Flowgraph* di atas:

1). Menghitung kompleksitas *cyclomatic* V(G) pada *egde* dan *node*

Pada rumus : V(G) = E – N + 2

E (*edge*) = 7

N (*node*) = 7

P (Predikat *node*) = 1

Penyelesaian :

V(G) = E – N + 2

= 7 – 7 + 2

= 2

Predikat (P) = P + 1

= 1 + 1

= 2

2). Berdasarkan perhitungan kompleksitas *cyclomatic* dari *flowgraph* di atas, memiliki = 2

3). Jalan independen pada *flowgraph* ini adalah:

Path 1 = 1 – 2 – 3 – 4 – 3

Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 7

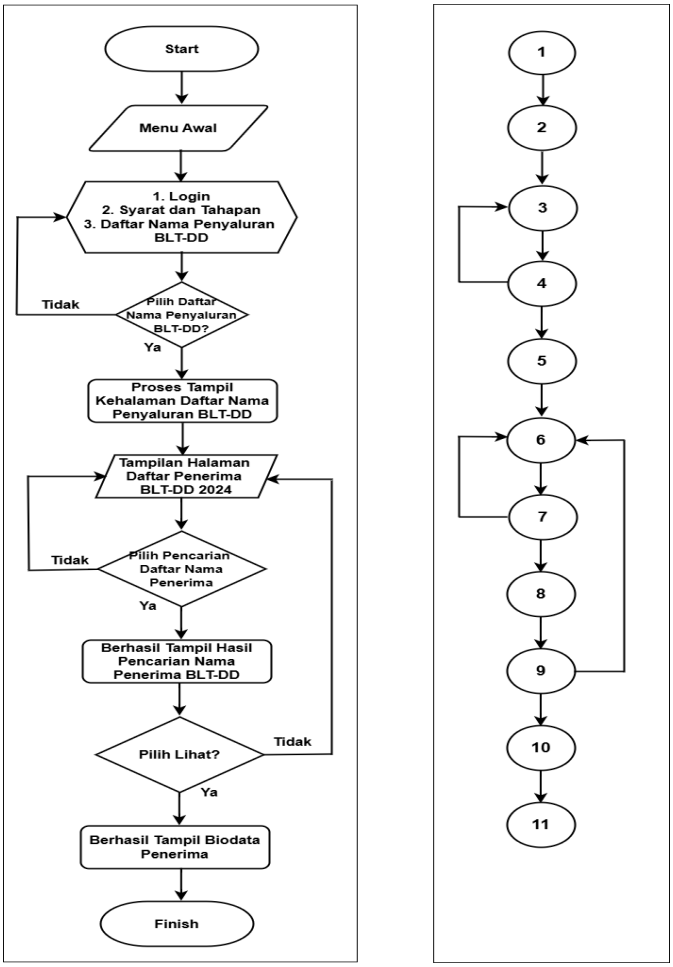
4).Grafik matriks menu syarat dan tahapan

Tabel 4. 21 Grafik matriks menu syarat dan tahapan

	1	2	3	4	5	6	7	E - 1
1		1						1 - 1 = 0

2			1					1 - 1 = 0
3				1				1 - 1 = 0
4			1		1			2 - 1 = 1
5						1		1 - 1 = 0
6							1	1 - 1 = 0
7								0
	SUM (E + 1)							1 + 1 = 2

e. *Flowchart* dan *Flowgraph* Menu Daftar Nama Penyaluran BLT-DD



Gambar 4. 54 *Flowchart* dan *Flowgraph* Menu Daftar Nama Penyaluran BLT-DD

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau *Flowchart* dan *Flowgraph* di atas:

1). Menghitung kompleksitas *cyclomatic* V(G) pada *egde* dan *node*

Pada rumus : V(G) = E – N + 2

E (*edge*) = 13

N (*node*) = 11

P (Predikat *node*) = 3

Penyelesaian :

V(G) = E – N + 2

= 13 – 11 + 2

= 4

Predikat (P) = P + 1

= 3 + 1

= 4

2). Berdasarkan perhitungan kompleksitas *cyclomatic* dari *flowgraph* di atas, memiliki = 4

3). Jalan independen pada *flowgraph* ini adalah:

Path 1 = 1 – 2 – 3 – 4 – 3

Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 6

Path 3 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 6

Path 3 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11

4).Grafik matriks menu daftar penyaluran BLT-DD

Tabel 4. 22 Grafik matriks daftar nama penyaluran BLT-D

	1	2	3	4	5	6	7	8	9	10	11	E - 1
1		1										1 - 1 = 0
2			1									1 - 1 = 0
3		1		1								2 - 1 = 1
4					1							1 - 1 = 0
5						1						1 - 1 = 0
6					1		1					2 - 1 = 1
7								1				1 - 1 = 0
8									1			1 - 1 = 0
9						1				1		2 - 1 = 1
10											1	1 - 1 = 0
11												
	SUM (E + 1)											3 + 1 = 4

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan penelitian, penulis dapat mengambil kesimpulan sebagai berikut :

1. Pengembangan aplikasi Android untuk sistem informasi penyaluran BLT-DD untuk meningkatkan transparansi, efisiensi, dan akuntabilitas distribusi bantuan. Fitur pemetaan memungkinkan verifikasi lokasi penerima secara real-time, mengurangi kesalahan administrasi, mencegah potensi penyimpangan, dan mempercepat proses distribusi dibandingkan metode manual.
2. Selain itu, aplikasi ini mempermudah pemerintah desa dalam mengelola data penerima secara lebih terstruktur, memastikan bantuan tepat sasaran, serta mendukung pelaporan digital yang sistematis dan akurat. Dengan akses data yang lebih mudah bagi pihak berwenang, pengawasan menjadi lebih efektif, sehingga meningkatkan kepercayaan masyarakat terhadap transparansi penyaluran bantuan.

B. Saran

Penulis penelitian ini mengakui bahwa masih banyak permasalahan yang memerlukan pengembangan dan perbaikan dalam penelitian lebih lanjut. Oleh karena itu, penulis menawarkan rekomendasi berikut untuk pengembangan studi lebih lanjut sebagai berikut:

1. Pertama, perlu dilakukan pengujian pengguna dengan melibatkan perangkat desa dan penerima BLT-DD guna memastikan aplikasi mudah digunakan serta mampu mencatat data dengan akurat. Selain itu, keamanan data harus diperhatikan dengan menerapkan mekanisme perlindungan seperti enkripsi data dan autentikasi pengguna agar informasi yang tersimpan tetap aman. Selanjutnya, integrasi dengan sistem informasi desa atau pemerintah dapat menjadi langkah strategis untuk meningkatkan efisiensi dalam pengelolaan dan distribusi dana BLT-DD.
2. Pengembangan lebih lanjut juga dapat mencakup fitur tambahan, seperti pelaporan otomatis dan notifikasi kepada penerima bantuan, sehingga proses distribusi menjadi lebih transparan dan efisien. Terakhir, evaluasi efektivitas sistem setelah implementasi sangat diperlukan untuk mengukur dampak aplikasi terhadap peningkatan transparansi, akurasi, serta kecepatan dalam penyaluran BLT-DD. Dengan berbagai perbaikan ini, diharapkan aplikasi dapat memberikan manfaat optimal bagi pemerintah desa maupun masyarakat penerima bantuan.

DAFTAR PUSTAKA

Asmara, J. (2019). Rancang Banagun Sistem Informasi Desa Berbasis Website(Studi Kasus Desa Netpala). *Jurnal Pendidikan Teknologi Informasi Jukanti Volume (2) NO (1) STIKOM Uyelindo Kupang, Nusa Tenggara Timur*.

Hartiwi, E. N. (2022). Aplikasi Inventori Barang Menggunakan Java dengan PHP Myadmin. *Cross-borderVol. 5No. 1 Fakultas Ilmu Komputer, Universitas Gunadarma, Indonesia*.

Herawan. (2021). sistem Informasi Penyedia Jasa Penjualan Pakaian Berbasis Android. *Universitas Muhammadiyah Parepare*.

Herwa, Hendrayudi, & Hartati, S. (2024). Membangun Sistem Informasi Inventaris Barang Pada Pt. Buana Eltra Menggunakan Php Dan Mysql . *Jurnal Sistem Informasi Mahakarya (JSIM), Vol. 7, No. 1, Juni 2024, Universitas Mahakarya Asia 2,3JL. Jend. A. Yani No. 267A Tanjung Baru, Baturaja, OKU, Sumatera Selatan* .

Husain, M. Z. (2021). Implementasi Library Tensorflow untuk Pendeteksi Wajah Berbasis Android. *Universitas Muhammadiyah Parepare*.

Lestanti, B. E. (2024). Analisis Sentimen Berdasarkan Hasil Review Lokasi Google Map Menggunakan Natural Language Toolkit TextBlob dan Naïve Bayes. *Program Studi Teknik Informatika, Universitas Islam Balitar, Blitar, Jawa Timur - Indonesia*.

Magdalena, H., Aprilianiza, N., & Santoso, H. (2023). Pengembangan Sistem Informasi Penyaluran Dan penatausahaan Bantuan Dana Desa Berbasis Web. *Matrik Jurnal Volue,7 Teknik Informatika, Universitas Mercubuana, Indonesia*.

Ni Made Dwi Febriyanti, A. K. (2021). Implementasi Black Box Testing pada Sistem Informasi Manajemen Dosen. *Jurnal Ilmiah Teknologi dan Komputer Vol. 2, No. 3 Program Studi Teknologi Informasi, Fakultas Teknik, Universitas Udayana Bukit Jimbaran, Bali, Indonesia*.

Nur Azis, G. P. (2020). Analisa dan Perancangan Aplikasi Pembelajaran Bahasa Inggris Dasar Berbasis Android. *Jurnal IKRA-ITH Informatika Vol 4 No 3*.

Nurhidayani S. (2023). Aplikasi Tes Uji Kecerdasan Berbasis Android. *Universitas Muhammadiyah Parepare*.

Parjito, R. U. (2022). Rancang Bangun Aplikasi E-Agribisnis Untuk Meningkatkan penjualan Hasil Tanaman Hortikultura. *Jurnal Informatika dan Rekayasa*

Perangkat Lunak (JATIKA) Fakultas Teknik dan Ilmu Komputer Universitas Teknokrat Indonesia .

Ramadhan, A., & Nurhad, A. (2024). Sistem Informasi Pengolahan Data PPKS Pada Sentra Terpadu "Pangudi Luhur" Bekasi. *Jurnal Informatics and Computer Engineering Journal Volume 4 No. 1 Februari 2024, Universitas Bina Sarana Informatika*.

Setyaningrum, S. E. (2024). Sistem Penentu Penerimaan Bantuan Sosial (BLT-DD) untuk Keluarga Kurang Mampu. *Program Studi Sistem Informasi Bisnis, Jurusan Teknologi Informasi, Politeknik Negeri Malang*.

Shiddiq, M. I. (2021). IMPLEMENTASI WHITE BOX TESTING BERBASIS PATH PADA FORM LOGIN APLIKASI BERBASIS WEB. *Jurnal Siliwangi Vol. 8.No.1 Program Studi Informatika Fakultas Teknik Universitas Siliwangi Tasikmalaya*.

Sondang Sibuea, M. I. (2022). Aplikasi Mobile Collection Berbasis Android pada PT. Suzuki Finance Indonesia. *JURNAL JITEK Vol 2 No. 1 Universitas Mohammad Husni Thamrin* .

Suherman, W. N. (2022). Perancangan Aplikasi Pendataan Bantuan Langsung Tunai(BLT) Berbasis Web Pada Kantor Desa Maccile Kecamatan Lalabat Kabupaten Soppeng. *Jurnal Ilmiah Sistem Informasi dan Teknik Informatika (JISTI), Volume 5 Nomor2 Teknik Informatika, Universitas Lamappapoleonro*.

Sukma, S. R. (2023). Perancangan Sistem Informasi Penyaluran Dana Bantuan Langsung Tunai (BLT) pada Desa Pambang Pesisir. *Ilmu Komputer, Manajemen Dakwah, Sekolah Tinggi Agama Islam (STAIN) Bengkalis* .

Sutianah, S. K. (2023). Sistem Klasifikasi Penentuan Penerima Bantuan Langsung Tunai Menggunakan Metode K-Nearest Neighbor. *Fakultas Teknologi Informasi, Program Studi Teknik Informatika Universitas Catur Insan Cendekia Cirebon, Indonesia.*

Wahid, A. A. (2020). Analisis Metode Waterfall Untuk Pengembangan Sistem informasi. *Jurnal Ilmu-ilmu Informatika dan Manajemen STMIK Sumedang* .

Yunengsi. (2023). Sistem Informasi Penyewaan Lapangan Futsal dan Badminton Sekota Parepare. *Universitas MuhammadiyahParepare*.