

BAB I

PENDAHULUAN

A. Latar Belakang

Perkembangan dibidang teknologi informasi khususnya teknologi internet mempermudah dan membantu berbagai bidang pekerjaan yang terkait dengan kemudahan akses, jarak, dan waktu, semakin mudahnya akses internet pada masa sekarang turut mendorong berbagai industri konstruksi maupun jasa konsultan baik dikalangan pemerintahan maupun swasta Teknologi informasi yang pesat saat ini sudah banyak dimanfaatkan oleh banyak pihak dengan hadirnya aplikasi bergerak (*mobile apps*). *Smartphone* dengan aplikasinya mampu menjembati kebutuhan antara pengguna dan pemberi kerja khususnya bagi pekerjaan bangunan.

Manajemen konstruksi bisa dikatakan suatu pekerjaan yang dilakukan dengan tujuan untuk mengatur, mengorganisir dan mengkoordinir semua pekerjaan yang dilaksanakan dan terlibat dalam pembangunan sebuah proyek konstruksi. Pekerjaan konstruksi dapat dikatakan baik apabila semua pelaksanaan pembangunan dapat berjalan sesuai rencana. Sistem pengelolaan pembangunan dalam pekerjaan konstruksi dapat dikatakan mempunyai pengaruh seperti yang diharapkan dengan melakukan penataan akan semua proses kegiatan pekerjaan konstruksi yang akan dilaksanakan sehingga mendapatkan hasil pekerjaan yang baik.

Konsultan Manajemen konstruksi memegang peranan yang sangat penting di dalam keberhasilan sebuah proyek. Tugas sebuah perusahaan konsultan adalah

mengawal klien pada tahap awal proyek (tahap perencanaan dan perancangan) untuk mempersiapkan tahap selanjutnya, serta pada masa konstruksi (pelaksanaan pembangunan fisik). *Job description* konsultan adalah menerjemahkan keinginan dan kebutuhan klien dengan mendampingi konsultan perencana dalam proses desain yang dituangkan ke dalam dokumen gambar, perhitungan dan dokumen pendukung lainnya. Kemudian melakukan pengawasan dan pendampingan kontraktor pada fase pelaksanaannya. Perencanaan di awal proyek yang matang akan menghasilkan sebuah produk pedoman pelaksanaan yang akurat.

Dalam suatu pelaksanaan proyek konstruksi terdiri dari serangkaian aktivitas-aktivitas yang saling berkaitan satu dengan yang lain. Untuk itu, diperlukan suatu manajemen konstruksi yang tepat dan dapat mengendalikan suatu proyek konstruksi mulai dari tahap perencanaan, tahap perancangan, tahap pelelangan, tahap pelaksanaan dan tahap sesudah pelaksanaan. Dalam mengendalikan tahap demi tahap tersebut, dibutuhkan konsultan manajemen konstruksi.

CV. Mutiara Alga Suppa adalah perusahaan yang bergerak di bidang jasa konsultan yang telah terdaftar sebagai anggota Ikatan Nasional Konsultan Indonesia (INKINDO) yang berada di Kab. Pinrang, Sulawesi Selatan. Dalam menjalankan kegiatannya masih menggunakan metode konvensional, telah terkomputerisasi namun masih dalam tahap operasional. Sehingga untuk pelaporan harian yang dilakukan oleh konsultan atau pengawas lapangan kepada *owner* masih memanfaatkan media sosial berupa *group chat whatsapp* untuk melaporkan kegiatan di lapangan dan akan mengisi *form* laporan harian yang berupa file excel

saat konsultan atau pengawas lapangan telah kembali ke kantor. Namun dalam pelaporan dengan menggunakan media sosial berupa *group chat whatsapp* membuat *owner* kebingungan untuk menyeleksi setiap laporan dari beberapa proyek yang ada, karena *history grup chat* yang tertumpuk membuat *owner* tidak bisa sepenuhnya mengetahui perkembangan dari setiap proyek perharinya dan dari sisi konsultan atau pengawas lapangan, sekitar 50% lupa akan kegiatan selama di lapangan untuk diinputkan kedalam file *excel* yang telah disediakan ketika kembali ke kantor, dikarenakan waktu berada di lapangan dapat dilakukan sehari-hari sebelum konsultan atau pengawas lapangan kembali ke kantor. Hal ini mengakibatkan tidak akuratnya data karena tidak selarasnya pelaporan yang dilaporkan pengawas lapangan sehari-hari kepada *owner* melalui media sosial dengan pelaporan yang diinputkan kedalam file *excel* oleh pengawas.

Selain itu CV. Mutiara Alga Suppa masih sering terjadi keterlambatan dan penyimpangan kualitas konstruksi pada tahap pelaksanaan proyek bukan hanya disebabkan oleh faktor alam, tetapi juga disebabkan oleh beberapa hal antara lain koordinasi, komunikasi, administrasi, pemberdayaan tenaga kerja sebagai sumber daya manusia yang kurang optimal.

Undang – Undang No. 2 Tahun 2017 Tentang Jasa Konstruksi, terlihat bahwa pekerjaan konsultan juga mempunyai resiko yang cukup besar dari berbagai segi, baik dari segi materiil maupun segi non materiil yang berhubungan dengan ketepatan waktu, kebutuhan tenaga, menjaga kepercayaan, serta kredibilitas perusahaan itu sendiri. Dalam hal keterlambatan pekerjaan, nama baik dari perusahaan konsultan sangat penting dan menjadi pertaruhan, karena akan

mempengaruhi kepercayaan dari pengguna barang/ jasa. Apabila kepercayaan dari pengguna barang/ jasa sudah tidak dipercayai, maka kesempatan konsultan untuk berkembang lebih baik kedepannya menjadi semakin sempit.

Berdasarkan permasalahan pada CV. Mutiara Alga Suppa khususnya pada bagian konsultan atau pengawas lapangan membutuhkan aplikasi yang dapat menampung laporan *progress* harian yang mana dapat langsung di pantau oleh *user/ klien*, maka penulis tertarik melakukan penelitian dengan judul “**Aplikasi e-Management Konstruksi Dalam Pencegahan Rework Pada Progress Bangunan Perumahan Mutiara Alga Suppa Berbasis Android**”.

B. Rumusan Masalah

Berdasarkan uraian latar belakang yang telah diuraikan, maka akan dibuat suatu penelitian pada masalah yang dihadapi oleh CV. Mutiara Alga Suppa. Adapun latar belakang masalah yang dapat diidentifikasi sebagai berikut:
Bagaimana cara membuat Aplikasi *e-Management Konstruksi Progress Bangunan Rumah Berbasis Android*?

C. Batasan Masalah

Adapun agar pembahasan menjadi lebih terarah dan tidak menyimpang dan juga sesuai dengan latar belakang yang sudah diuraikan sebelumnya, maka penulis membatasi permasalahan penelitian sebagai berikut :

1. Ruang lingkup penelitian ini difokuskan atau terpusat di CV. Mutiara Alga Suppa, dalam hal ini sumber data diperoleh dari perusahaan tersebut.

2. Jenis proyek konstruksi ini adalah proyek konstruksi pada bangunan rumah diperumahan Multi Alga Suppa berlokasi di Kabupaten Pinrang.
3. Aplikasi ini dibangun untuk membantu pihak konsultan manajemen konstruksi dan dapat menampung laporan *progress* harian sebagai bentuk pengawasan yang langsung dipantau oleh *user/ klien* secara berkala.

D. Tujuan Penelitian

Berdasarkan rumusan masalah yang telah didefinisikan sebelumnya tujuan dari penelitian dan perancangan sistem dalam penelitian ini adalah :

1. Untuk membangun suatu sistem Aplikasi *e-Management* Konstruksi *Progress* Bangunan Perumahan Mutiara Alga Suppa Berbasis *Android*.
2. Untuk menerapkan *e-Management* Kontruksi pada aplikasi *android* di CV. Mutiara Alga Suppa yang dapat meningkatkan efektifitas kesuksesan sebuah proyek dengan tepat sasaran.

E. Manfaat Penelitian

Adapun Salah satu fungsi dari penelitian adalah bisa memberikan manfaat dan kegunaan yang secara nyata akan diterima oleh berbagai pihak yang terkait dengan obyek penelitian seperti aspek akademis dan praktis. Dalam penelitian kali ini peneliti berharap bisa memberikan kegunaan, antara lain :

1. Aspek Akademis

- a. Untuk mengaplikasikan teori yang di dapat di bangku perkuliahan dengan kondisi real di lapangan, khususnya terkait dengan permasalahan yang dibahas terkait penelitian dan *tools* atau metode penyelesaian masalah.
- b. Diharapkan Sebagai bahan masukan bagi para pengkaji ilmu komunikasi di bidang *Public Relations* yang berminat untuk meneliti masalah yang sama.

2. Aspek Praktis

- a) Membangun sebuah sistem *e-management* kontruksi di CV. Mutiara Alga Suppa yang akan memberikan kemudahan dan pegelolaan data dan memberikan informasi bagi rekanan, *owner/klien* atau pejabat instansi yang terkait dengan konsultan manajemen konstruksi, dan memudahkan memahami proses kerja konsultan, agar terjadi hubungan komunikasi antar berbagai pihak terkait.
- b) Bagi peneliti diharapkan dapat memberikan pengalaman dan menerapkan ilmu pengetahuan yang diterima selama mengikuti perkuliahan.
- c) Sebagai tugas akhir untuk mendapatkan gelar sarjana strata 1 (S.1) pada Fakultas Teknik, Program Studi Informatika, Universitas Muhammadiyah Parepare.

F. Sistematika Penulisan

Adapun dalam menyusun sistematika penulisan, penulis menguraikan ke dalam enam bab yaitu :

BAB I : PENDAHULUAN

Pada bab ini akan dijelaskan tentang latar belakang penulisan, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

BAB II : TINJAUAN PUSTAKA

Bab ini berisi uraian tentang landasan teori-teori dari perancangan “Aplikasi *e-Management* Konstruksi *Progress* Bangunan Perumahan Mutiara Alga Suppa Berbasis *Android*”. Bab ini juga membahas mengenai berbagai teori yang mendasari dari analisis permasalahan yang berhubungan dengan topik yang dibahas.

BAB III : METODOLOGI PENELITIAN

Dalam bab ini membahas mengenai metode penelitian yang digunakan dalam merancang dan membangun sistem. Pada bab ini juga membahas tentang identifikasi masalah, analisis kebutuhan aplikasi yang meliputi kebutuhan masukan, proses dan keluaran aplikasi yang akan dibangun. Penjelasan mengenai proses perancangan dengan menggunakan rancangan dan UML.

BAB IV : HASIL DAN PEMBAHASAN

Pada bab ini membahas mengenai hasil dari tahap analisis dan perancangan sesuai dengan metode yang digunakan pada alat dan sistem yang dibuat serta hasil dari pengujian.

BAB V : PENUTUP

Pada bab akhir ini berisi tentang kesimpulan dari penelitian yang telah dilakukan dan saran-saran yang diberikan untuk pengembangan lebih lanjut agar tercapai hasil yang lebih maksimal.

DAFTAR PUSTAKA

Pada bagian ini akan dipaparkan tentang sumber-sumber literatur, tutorial, buku maupun situs-situs yang digunakan dalam pembuatan laporan tugas akhir ini.

LAMPIRAN

BAB II

TINJAUAN PUSTAKA

A. Penelitian Terdahulu

Berbagai penelitian sebelumnya merupakan hal yang sangat perlu dan dapat dijadikan sebagai data pendukung. Salah satu data pendukung yang menurut peneliti perlu dijadikan bagian tersendiri adalah penelitian terdahulu yang relevan dengan permasalahan yang sedang dibahas dalam penelitian ini. Oleh karena itu, peneliti melakukan kajian terhadap beberapa hasil penelitian berupa tugas akhir/skripsi atau jurnal melalui *internet*.

1. Harri Gunawan, (2020) dengan penelitian dalam sebuah jurnal yang berjudul **“Sistem Informasi Manajemen Perumahan Berbasis Web Pada PT Catur Graha Mandiri Palembang”**. Peneliti diatas membuat salah satu solusi dari permasalahan PT. Catur Graha Mandiri dalam memonitoring dan mengevaluasi

proyek yang sedang berjalan dan sudah direalisasikan. Tujuan dari penelitian ini adalah untuk memberikan kemudahan dalam pengolahan data proyek serta membantu perusahaan terkait dalam mengatasi permasalahan diantaranya RAB, *monitoring* persediaan, pembuatan penjadwalan, proses pembangunan. Dalam pengembangan sistem ini penulis menggunakan metodologi literasi, untuk menganalisis suatu masalah menggunakan analisis PIECES, mengumpulkan data melalui observasi, wawancara/ *interview*, dan studi pustaka. Peneliti membuat sistem informasi manajemen perumahan berbasis *website* ini untuk memecahkan masalah tersebut.

2. Luthfan Noor Irsyad, (2021) dengan penelitian dalam sebuah skripsi yang berjudul **“Analisis Dan Perancangan Desain Sistem Informasi Manajemen Proyek (Studi Kasus: Dinas Perumahan Kota Pasuruan)”**. Berdasarkan hasil penelitian tujuan utama dari penelitian tugas akhir ini adalah untuk merancang aplikasi berbasis *web* yang dapat digunakan untuk membantu aktivitas manajemen proyek di Dinas Perumahan Kota Pasuruan. Hasil analisis dan perancangan sistem akan dilakukan evaluasi dengan menggunakan metode *traceability*, *usability testing* dan pengujian *prototype*. Dari hasil analisis kebutuhan menghasilkan 24 fitur yang terdiri dari 21 persyaratan fungsional dan persyaratan non-fungsional, 3 aktor dan 8 *use case diagram*. Pada evaluasi menggunakan metode *traceability* dengan sebuah *matrix*.
3. Wita Dwijayanti, (2021) dengan penelitian dalam sebuah jurnal yang berjudul **“Sistem Informasi Manajemen Proyek Pembangunan Perumahan Griya Aldena Karanganyar”**. Sistem informasi ini bertujuan memudahkan

pekerjaan pembangunan perumahan sehingga terciptalah manajemen yang sistematis dan teratur. Hasil dari penelitian ini berdasarkan kasus dan hasil pengujian dengan menggunakan metode black box yang telah dibuat maka disimpulkan bahwa semua data dapat terupdate. Sehingga sistem dapat berjalan secara fungsional dan berjalan dengan baik.

Pada ketiga penelitian terkait diatas memanfaatkan suatu teknologi internet berbasis *online* dalam suatu manajemen perumahan. Jika dibandingkan dengan penelitian saat ini, tentunya lebih unggul aplikasi ini, karena sistem ini menerapkan teknologi perangkat bergerak *smartphone* untuk aplikasi manajemen kontruksi.

B. Tinjauan Teori

1. Aplikasi (*Application*)

Aplikasi berasal dari kata *application* yaitu bentuk benda dari kata kerja *to apply* yang dalam Bahasa Indonesia berarti pengolah. Secara istilah, aplikasi komputer adalah suatu *sub* kelas perangkat lunak komputer yang menggunakan komputer langsung untuk melakukan suatu tugas yang diinginkan pemakai. Menurut Kamus Besar Bahasa Indonesia (KBBI, 2015) Aplikasi 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 pengguna. Sementara menurut Hartono (1999) aplikasi adalah penggunaan dalam suatu komputer, instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses *input* menjadi *output*.

2. Defenisi Manajemen Kontruksi

Dalam suatu proyek konstruksi terdapat banyak rangkaian - rangkaian kegiatan yang nantinya akan dilaksanakan dan umumnya dibatasi oleh waktu yang telah ditentukan. Dengan kriteria bangunan konstruksi yang sangat berbeda-beda karakteristik dan kemungkinan banyak kendala dengan kondisi yang belum kita tahu Apabila pegawai atau pekerja tidak mencatatkan jam hadirnya pada kartu jam hadir maka tunjangan makan dan hingga transpotasinya yang diterima pegawai atau pekerja setiap bulannya akan berkurang dan juga akan mempengaruhi gaji bersih yang diterima pegawai atau pekerja tersebut sebelumnya menyebabkan kebutuhan akan manajemen konstruksi sangat penting dengan sistem-sistem yang dibuat untuk mendapat hasil seperti yang diharapkan. Berikut ini tentang beberapa definisi dari manajemen proyek, diantaranya sebagai berikut :

- a. Menurut Budi Santoso (2003), Manajemen proyek merupakan kegiatan merencanakan, mengorganisasikan, mengarahkan dan mengendalikan sumber daya organisasi perusahaan untuk mencapai tujuan dalam waktu tertentu dengan sumber daya tertentu.
- b. Menurut Harold Kerzner (2001), Manajemen proyek berarti suatu kegiatan yang berada dalam konstruksi dengan 3 (tiga) hal yang perlu diperhatikan yaitu waktu, biaya dan mutu.
- c. Menurut Wulfram I. Ervianto (2005), Manajemen proyek konstruksi ada dua pemahaman yang pada pelaksanaannya menjadi satu kesatuan dalam mencapai tujuan proyek yaitu :

- 1) Teknologi Konstruksi (*Construction Technology*) yaitu mempelajari metode atau teknik tahapan pelaksanaan pekerjaan dalam mewujudkan bangunan fisik di suatu lokasi proyek, sesuai dengan spesifikasi teknik yang disyaratkan.
- 2) Manajemen Konstruksi (*Construction Management*) adalah bagaimana sumber daya (*man, material, machine, money, method*) yang terlibat dalam pekerjaan dapat dikelola secara efektif dan efisien untuk mencapai tujuan proyek, sesuai dengan kebutuhan atau hukum yang berhubungan dengan konstruksi.

Manajemen konstruksi telah diakui sebagai salah satu cabang manajemen yang khusus, yang dikembangkan dengan tujuan untuk dapat melakukan koordinasi dan pengendalian atas beberapa kegiatan pelaksanaan proyek yang sifatnya kompleks. Dengan demikian, teknik atau manajemen yang dapat mengakomodasi kebutuhan sumber daya konstruksi selalu dilakukan peninjauan dan penyesuaian terus menerus, setiap saat dalam menyelesaikan pelaksanaan pekerjaan yang sedang berjalan.

3. Gambaran Umum Perusahaan

CV. Mutiara Alga Suppa adalah Perusahaan Jasa Konsultan Teknik Nasional yang independen dan tidak mempunyai hubungan dengan kontraktor, supplier atau perusahaan industri manapun. Perusahaan ini didirikan pada Tanggal 18 Maret 2019 Nomor : 31 dan pada tanggal 19 September 2019 dengan Nomor : 18 terjadi perubahan kepengurusan. CV. Mutiara Alga Suppa didirikan dengan

tujuan untuk dapat memberikan jasa layanan teknik dalam memenuhi tuntutan dan kebutuhan program Pembangunan Perumahan di seluruh wilayah.

CV. Mutiara Alga Suppa dalam kurun waktu tiga tahun sejak didirikan telah melaksanakan pekerjaan dengan baik. Jasa layanan teknik tersebut mencakup bidang-bidang pekerjaan sipil, tata lingkungan dan arsitektur dan sejak tahun 2019 terjadi penambahan bidang yaitu bidang layanan Jasa Inspeksi Teknik.

CV. Mutiara Alga Suppa dalam melaksanakan pekerjaan layanan teknik yang dipercayakan oleh pemberi tugas, sebagian besar kami laksanakan secara mandiri, namun demikian juga kami berpengalaman dalam bekerja secara konsorsium dengan konsultan lain.

CV. Mutiara Alga Suppa berkantor pusat di Jl. Kebangkitan No. 01, Dusun Polewali, Kec. Suppa, Kabupaten Pinrang, Sulawesi Selatan. dan pada saat ini memiliki tenaga ahli profesional dan tenaga pendukung teknik yang berpengalaman. Peningkatan kemampuan teknis tenaga ahli diselenggarakan dengan mengikuti program training, kursus singkat serta ahli teknologi dari tenaga ahli senior perusahaan.

CV. Multi Alga Suppa dalam mendukung seluruh kegiatan perusahaan menerapkan sistem manajemen yang efisien serta dilengkapi dengan fasilitas pendukung berupa peralatan *survey*, investigasi, studio serta fasilitas komputer berikut perangkat lunak.

CV. Multi Alga Suppa dengan tenaga ahli, pengalaman, manajemen beserta fasilitas pendukung yang ada yakin dapat menyelesaikan pekerjaan ini dengan baik.

a. Visi dan Misi Perusahaan

1) Visi Perusahaan

Memberikan layanan terpadu dan menyeluruh di bidang perencanaan, pengawasan bidang arsitektur, sipil dan tata lingkungan untuk mencapai hasil yang optimal. Dengan rentang layanan yang berorientasikan pada komitmen secara efektif dan efisien

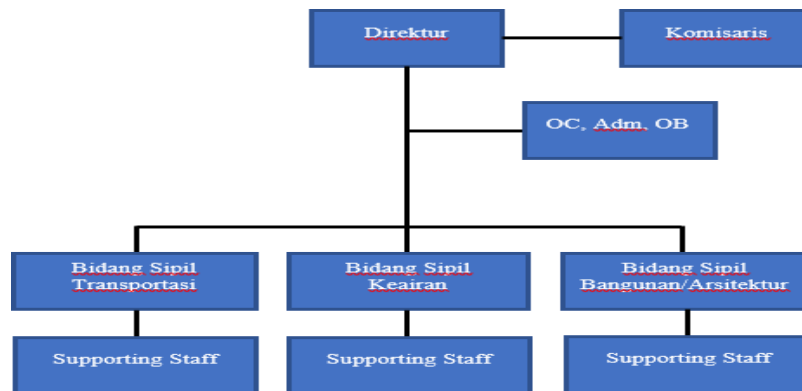
2) Misi Perusahaan

Misi CV. Mutiara Alga Suppa adalah sebagai berikut :

- a) SOLUSI. Menawarkan solusi hasil terbaik, optimal untuk kondisi saat ini dan perubahan yang terjadi di tingkat global.
- b) KEPUASAN KONSUMEN. Peka terhadap perubahan kebutuhan konsumen dan selalu mencari cara inovatif untuk memuaskannya.
- c) PENGEMBANGAN LAYANAN YANG AKSELERATIF. Tim riset pengembangan dan tim lapangan kami selalu berupaya merancang layanan yang cepat dan selaras dengan perkembangan mutakhir di bidang teknologi, arsitektur, sipil dan tata lingkungan.
- d) LAYANAN BERKUALITAS. Acuan standar kualitas layanan dilengkapi dengan pertimbangan efisiensi dan komitmen penuh terhadap hasil.
- e) PENGEMBANGAN SUMBER DAYA MANUSIA. Konsistensi dalam upaya untuk mendorong pengembangan staf untuk mencapai tingkat kompetensi unggulan.

b. Struktur Organisasi Perusahaan

Adapun stuktur organisasi perusahaan dapat dilihat pada gambar sebagai berikut.



Gambar 2.1 Struktur Organisasi CV. Mutiara Alga Suppa

C. Dasar Pemrograman

1. Pengertian Pemrograman

Pemrograman atau *programming* merupakan proses menulis dan merancang serangkaian instruksi untuk mengarahkan komputer agar dapat menjalankan tugas-tugas tertentu. Tugas atau instruksi-instruksi tersebut ditulis dalam bentuk kode-kode yang terdiri dari berbagai simbol dan sintaksis yang dimengerti oleh komputer. Singkatnya, pemrograman merupakan proses membuat program di komputer.

2. Pengertian Algoritma

Algoritma adalah urutan langkah-langkah logis penyelesaian masalah yang disusun secara sistematis dan logis". Kata logis merupakan kata kunci dalam algoritma. Langkah-langkah dalam algoritma harus logis dan harus dapat ditentukan bernilai salah atau benar. Dalam beberapa konteks, algoritma adalah spesifikasi urutan langkah untuk melakukan pekerjaan tertentu. Pertimbangan

dalam pemilihan algoritma adalah, pertama, algoritma haruslah benar. Artinya algoritma akan memberikan keluaran yang aksebagai apapun algoritma, kalau memberikan keluaran yang kanlah algoritma yanertimbangan kedua yang harus diperhatikan adalah kita harusmengetahui seberapa baik hasil yangdicapai oleh algoritma tersebut. Hal ini penting terutama pada algoritma untuk menyelesaikan masalah yang memerlukan aproksimasi hasil mampu memberikan hasil yang sedekat mungkin dengan nilai yang sebenarnya. Ketiga adalah efisiensi algoritma. Efisiensi algoritma dapat ditinjau dari 2 hal yaitu efisiensi waktu dan memori. Meskipun algoritma memberikan keluaran yang benar (palingmendekati), tetapi jika kita harus menunggu berjam-jam untuk mendapatkan keluarannya, algoritma tersebut biasanya tidakakan dipakai, setiap orang menginginkan keluaran yang cepat. Begitu juga dengan memori, semakin besar memori yang terpakaimaka semakin buruklah algoritma tersebut. Dalam kenyataannya, setiap orang bisa membuat algoritma yang berbeda untuk menyelesaikan suatu permasalahan, walaupun terjadi perbedaandalam menyusun algoritma, tentunya kita mengharapkan keluaran yang sama. Jika terjadi demikian, carilah algoritma yang paling efisien dan cepat.

a. Algoritma Teks

Teks algoritma berisi deskripsi langkah-langkah penyelesaian masalah. Deskripsi tersebut dapat ditulis dalam notasi apa-pun asalkan mudah dimengerti dan dipahami. Tidak ada notasi yang baku dalam penulisan teks algoritma seperti notasi bahasa pemrograman. Notasi yang digunakan dalam menulis algoritma disebut notasi algoritmik. Setiap orang dapat membuat aturan penulisan dan notasi

algoritmik sendiri. Hal ini dikarenakan teks algoritma tidak samadengan teks program. Namun, supaya notasi algoritmik mudah di translasikan ke dalam notasi bahasa pemrograman tertentu, maka sebaiknya notasi algoritmik tersebut berkorespondensi dengan notasi bahasa pemrograman secara umum.

b. Algoritma Flowchart

Seperti yang di tuangkan oleh Fadilah di dalam makalah Flowchart (2020:3) membuat algorit ma dalam bent uk flowchart ada beberapa petunjuk yang harus diperhatikan, seperti :

1. Flowchart digambarkan dari halaman atas ke bawah dan dari kiri ke kanan
2. Aktivitas yang digambarkan harus didefinisikan secara hati -hati dan definisi ini harus dapat dimengerti oleh pembacanya.
3. Kapan aktivitas dimulai dan berakhir harus ditentukan secara jelas
4. Setiap langkah dari aktivitas harus diuraikan dengan menggunakan deskripsi kata kerja, misalkan MENGHITUNG PAJAK PENJUALAN.
5. Setiap langkah dari aktivitas harus berada pada urutan yang benar.
6. Lingkup dan range dari aktifitas yang sedang digambarkan harus ditelusuri dengan hati -hati. Percabangan-percabangan yang memotong aktivitas yang sedang digambarkan tidak perlu digambarkan pada flowchart yang sama. Simbol konektor harus digunakan dan percabangannya diletakan pada halaman yang terpisah atau hilangkan seluruhnya bila percabangannya tidak berkaitan dengan sistem.
7. Gunakan simbol-simbol flowchart yang standar

3. Tinjauan Perangkat Lunak

1. *Android*

Android memungkinkan penggunanya untuk memasang aplikasi pihak ketiga, baik yang diperoleh dari toko aplikasi seperti *Google Play*, *Amazon Appstore*, ataupun dengan mengunduh dan memasang berkas APK dari situs pihak ketiga. Di *Google Play*, pengguna bisa menjelajah, mengunduh, dan memperbarui aplikasi yang diterbitkan oleh *Google* dan pengembang pihak ketiga, sesuai dengan persyaratan *kompatibilitas Google*. *Google Play* akan menyaring daftar aplikasi yang tersedia berdasarkan *kompatibilitasnya* dengan perangkat pengguna, dan pengembang dapat membatasi aplikasi ciptaan mereka bagi operator atau negara tertentu untuk alasan bisnis. Pembelian aplikasi yang tidak sesuai dengan keinginan pengguna dapat dikembalikan dalam waktu 15 menit setelah pengunduhan. Beberapa operator seluler juga menawarkan tagihan langsung untuk pembelian aplikasi di *Google Play* dengan cara menambahkan harga pembelian aplikasi pada tagihan bulanan pengguna. Pada bulan September 2012, ada lebih dari 675.000 aplikasi yang tersedia untuk *Android*, dan perkiraan jumlah aplikasi yang diunduh dari *Play Store* adalah 25 miliar.

Aplikasi *Android* dikembangkan dalam bahasa pemrograman *Java* dengan menggunakan kit pengembangan perangkat lunak *Android* (SDK). SDK ini terdiri dari seperangkat perkakas pengembangan, termasuk *debugger*, perpustakaan perangkat lunak, *emulator* handset yang berbasis QEMU, dokumentasi, kode sampel, dan *tutorial*. Didukung secara resmi oleh lingkungan pengembangan terpadu (IDE) *Eclipse*, yang menggunakan *plugin Android Development Tools*.

2. PHP (*Hypertext Preprocessor*)

Menurut Diar Puji Octavian (2010:31) “PHP (*PHP Hypertext Prosesor*) adalah akronim dari *Hypertext Preprocessor*, yaitu suatu bahasa pemograman berbasis kode-kode (*script*) yang di gunakan untuk mengolah suatu data dan mengirimkannya kembali ke *web browser* menjadi kode HTML. Kode PHP mempunyai ciri-ciri khusus, yaitu :

- a. Hanya dapat dijalankan menggunakan *web server* misalnya : *Apache*.
- b. Kode PHP dapat diletakan dan dijalankan di *web server*.
- c. Kode PHP dapat digunakan untuk mengakses database, seperti: *MY SQL*, *PostgreSQL*, *Oracle*, dan lain-lain.
- d. Merupakan *software* yang bersifat *open source*.
- e. Gratis untuk di *download* dan digunakan.
- f. Memiliki sistem *multiplatform*, artinya dapat dijalankan menggunakan sistem operasi apapun, seperti *Linux*, *Unix*, *Windows*, dan lain-lain.

PHP merupakan *software Open-Source* yang disebarakan dan dilisensikan secara gratis serta dapat di-*download* secara bebas dari situs resminya <http://www.php.net>. PHP ditulis menggunakan bahasa C. PHP memiliki banyak kelebihan yang tidak dimiliki oleh bahasa *script* sejenis. PHP difokuskan pada pembuatan *script server-side*, yang bisa melakukan apa saja yang dapat dilakukan oleh CGI, seperti mengumpulkan data *form*, menghasilkan isi halaman *web* dinamis, dan kemampuan mengirim serta menerima *cookies*, bahkan lebih daripada kemampuan CGI.

3. *JavaScript*

JavaScript adalah bahasa pemrograman tingkat tinggi dan dinamis. *JavaScript* populer di internet dan dapat bekerja di sebagian besar penjelajah web populer seperti *Google Chrome*, *Internet Explorer (IE)*, *Mozilla Firefox*, *Netscape* dan *Opera*. Kode *JavaScript* dapat disisipkan dalam halaman web menggunakan tag *SCRIPT*. *JavaScript* merupakan salah satu teknologi inti *World Wide Web* selain *HTML* dan *CSS*. *JavaScript* membantu membuat halaman web interaktif dan merupakan bagian aplikasi web yang esensial. Awalnya hanya diimplementasi sebagai *client-side* dalam penjelajah web, kini *engine JavaScript* disisipkan ke dalam perangkat lunak lain seperti dalam *server-side* dalam server web dan basis data, dalam program *non web* seperti perangkat lunak pengolah kata dan pembaca *PDF*, dan sebagai *runtime environment* yang memungkinkan penggunaan *JavaScript* untuk membuat aplikasi *desktop* maupun *mobile*.

4. *HTTP (Hypertext Transfer Protocol)*

HTTP (Hypertext Transfer Protocol) adalah *protocol* yang dipergunakan untuk mentransfer dokumen dalam *World Wide Web (WWW)*. *Protocol* ini adalah *protocol* ringan, tidak berstatus dan generik yang dapat dipergunakan berbagai macam *tipe* dokumen. Pengembangan *HTTP* dikoordinasikan oleh *Konsorsium World Wide Web (W3C)* dan grup bekerja *Internet Engineering Task Force (IETF)*, bekerja dalam publikasi satu seri *RFC*, yang paling terkenal *RFC* yang menjelaskan *HTTP/1.1*, versi *HTTP* yang digunakan umum sekarang ini. *HTTP* adalah sebuah *protocol* meminta/ menjawab antara *client* dan *server*.

Sebuah *client* HTTP seperti web *browser*, biasanya melalui permintaan dengan membuat hubungan *TPC/IP* ke port tertentu. Sebuah server HTTP yang mendengarkan di *port* tersebut menunggu *client* mengirim kode permintaan di ikuti dengan pesan *MIME* yang memiliki beberapa informasi data tertentu. (Deka M Wildan, 2014).

5. *Adobe Dreamweaver CS6*

Adobe Dreamweaver adalah sebuah HTML (*Hyper Text Markup Language*) *editor* profesional untuk mendesain web secara visual dan mengelola situs atau halaman web. Saat ini terdapat *software* dari kelompok *adobe* yang belakangan banyak digunakan untuk mendesain suatu situs web. Versi terbaru dari *adobe dreamweaver* saat ini adalah *Dreamweaver CS6*. Pada *Dreamweaver CS6*, terdapat beberapa kemampuan bukan hanya sebagai *software* untuk mendesain web saja tetapi juga untuk menyunting kode serta pembuatan aplikasi web dengan menggunakan berbagai bahasa pemrograman web, antara lain *HTML*, *PHP*, *CSS* dan sebagainya.

Dreamweaver merupakan *software* utama yang digunakan oleh web desainer maupun web pemrograman dalam mengembangkan suatu situs web. Hal ini disebabkan ruang kerja, fasilitas dan kemampuan *dreamweaver* yang mampu meningkatkan produktivitas dan efektivitas dalam desain maupun membangun suatu situs web. (Madcoms Madiun, 2012)

6. *PhpMyAdmin*

PhpMyAdmin adalah perangkat lunak yang ditulis dalam bahasa pemrograman PHP digunakan untuk menangani administrasi *MySQL* dengan menggunakan protokol WWW secara langsung dari *browser*. *PhpMyAdmin* mendukung berbagai operasi *MySQL*, diantaranya mengelola basis data, tabel-tabel, bidang *fields*, relasi *relations*, indeks, pengguna *users*, perijinan *permissions*, dan lain-lain. Dengan menggunakan *phpMyAdmin* diharapkan akan memudahkan dalam memanajemen database *MySQL*. (Alan Nur Aditya, 2011)

7. *XAMPP*

XAMPP adalah perangkat lunak besar, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program *Apache HTTP server*, *MySql* database dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*. Nama *XAMPP* merupakan singkatan dari X (tempat sistem operasi apapun), *Apache*, *MySql*, *PHP* dan *Perl*. Program ini tersedia dalam GNU *general public license* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis sehingga mampu menyajikan tampilan dengan baik dan dinamis. (Alan Nur Aditya, 2011)

8. *UML (Unified Modelling Language)*

UML (Unified Modelling Language) menurut Munawar dalam buku “Pemodelan Visual dengan UML” (2005 : 17) menyatakan UML adalah salah satu alat bantu yang sangat handal didunia pengembangan sistem yang berorientasi

obyek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain.

Unified Modelling Language (UML) adalah “suatu alat bantu yang sangat handal dalam bidang pengembangan yang berorientasi objek. Hal ini disebabkan karena uml menyediakan bahasa pemodelan visual yang memungkinkan pengembang sistem untuk membuat cetak cetak biru atas visi mereka dalam bentuk baku”. *Metode Unified Modelling Language* (UML) menggunakan tiga bangunan dasar untuk mendeskripsikan suatu sistem atau perangkat lunak yang akan dikembangkan, yaitu :

a. Sesuatu (*things*)

Ada empat *things* dalam *Unified Modelling Language* (UML) :

1. *Structural things*, bagian yang relative statis dapat berupa elemen-elemen yang bersifat fisik maupun konseptual.
2. *Behaviorial things*, bagian dinamis biasanya merupakan kata kerja dari model UML yang mencerminkan perilaku sepanjang ruang dan waktu.
3. *Grouping things*, bagian pengorganisasian dalam UML. Dalam penggambaran model UML yang rumit diperlukan penggambaran paket yang menyederhanakan model.

4. *An notational things*, merupakan bagian yang memperjelas model UML.

Dapat berisi komentar yang menjelaskan fungsi serta ciri-ciri tiap element dalam model UML.

b. Relasi (*relationship*)

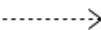
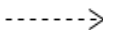
Untuk upaya tersebut UML menyediakan 9 jenis diagram yang dapat dikelompokkan berdasarkan sifatnya *statis* atau *dinamis*. Ke 9 jenis diagram dalam UML itu adalah :

1. *Diagram* kelas Bersifat statis *Diagram* ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. *Diagram* ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula *diagram* kelas memuat kelas-kelas aktif.
2. *Diagram* Objek Bersifat statis. *Diagram* ini memperlihatkan objek-objek serta relasi-relasi antar objek. *Diagram* objek memperlihatkan instalasi statis dari segala sesuatu yang dijumpai pada *diagram* kelas.
3. *Use-Case Diagram* Bersifat statis. *Diagram* ini memperlihatkan himpunan *use case* dan aktor-aktor (suatu jenis khusus dari kelas). *Diagram* ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan serta diharapkan pengguna.
4. *Sequence Diagram* Bersifat dinamis. *Diagram* urutan adalah *diagram* interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu.

5. *Collaboration Diagram* Bersifat dinamis. Diagram kolaborasi adalah diagram interaksi yang menekankan organisasi struktural dari objek-objek yang menerima serta mengirim pesan.
6. *Statechart Diagram* Bersifat dinamis. Diagram state ini memperlihatkan *state-state* pada sistem memuat *state*, *transisi*, *event*, serta aktifitas. Diagram ini terutama penting untuk memperlihatkan sifat dinamis dari antarmuka, kelas, kolaborasi dan terutama penting pada pemodelan sistem-sistem yang reaktif.
7. *Activity Diagram* Bersifat dinamis. *Diagram* aktivitas ini adalah tipe khusus dari *diagram state* yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem.
8. *Component Diagram* Bersifat statis. Dengan komponen ini memperlihatkan organisasi serta kebergantungan sistem atau perangkat lunak pada komponen-komponen yang telah ada sebelumnya.
9. *Deployment Diagram* Bersifat statis. *Diagram* ini memperlihatkan konfigurasi saat aplikasi dijalankan. *Diagram* ini membuat simpul-simpul beserta komponen-komponen yang ada di dalamnya. *Diagram* ini sangat berguna pada banyak mesin (Sholiq, 2006).

Tabel 2.1 Table Use Case Diagram

NO	GAMBAR	NAMA	KETERANGAN
----	--------	------	------------

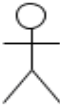
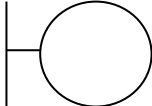
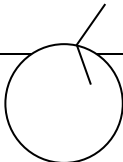
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek.
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan suatu paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan dalam sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (<i>sinergi</i>).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

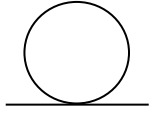
Tabel 2.2 *Symbol Activity Diagram*

NO	GAMBAR	NAMA	KETERANGAN
----	--------	------	------------

1		<i>Activity</i>	Memperlihatkan cara bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2		<i>Action</i>	State dari suatu sistem yang mencerminkan eksekusi dari suatu aksi.
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap- tahap tertentu dan berubah menjadi beberapa aliran

Tabel 2.3 *Symbol Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Objekct (Partisipan)</i>	<i>Object</i> atau biasa juga disebut partisipan merupakan <i>instance</i> dari sebuah <i>class</i> dan dituliskan tersusun secara <i>horizontal</i> .
2		<i>Actor</i>	<i>Actor</i> juga dapat berkomunikasi dengan <i>object</i> , maka <i>actor</i> juga dapat diurutkan sebagai kolom.
3		<i>Life line</i>	<i>Life line</i> mengidentifikasi keberadaan sebuah <i>object</i> dalam basis waktu. Notasi untuk <i>lifeline</i> .
4		<i>Collaborations</i>	<i>Activication</i> dinotasikan sebagai kotak segi empat yang digambar pada sebuah <i>lifeline</i> <i>action</i> mengidentifikasi sebuah <i>object</i> .
5		<i>Boundary</i>	<i>Boundary</i> terletak diantara <i>system</i> dengan dunia di sekelilingnya.
NO	GAMBAR	NAMA	KETERANGAN
6		<i>Control</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen

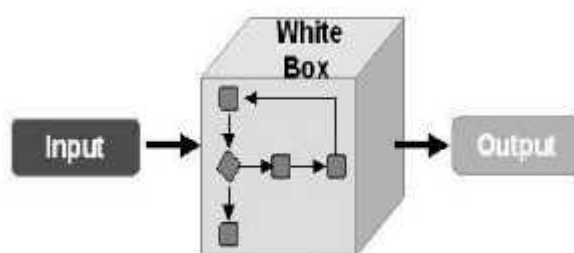
			mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7		<i>Entity</i>	Entity digunakan menengani informasi yang mungkin akan disimpan secara permanen. Entity biasa juga merupakan sebuah <i>table</i> .

9. Metode Pengujian Perangkat Lunak

Jika struktur kendali antar modul sudah terbukti bagus, maka pengujian yang tak kalah pentingnya adalah pengujian unit. Pengujian unit digunakan untuk menguji setiap modul untuk menjamin setiap modul menjalankan fungsinya dengan baik. Ada 2 metode untuk melakukan unit testing, yaitu:

1. *White Box Testing*

Whitebox testing adalah metode pengujian perangkat lunak yang memungkinkan penguji memiliki akses penuh ke struktur internal, algoritma, dan kode sumber dari sistem yang diuji. Dalam metode ini, penguji memahami bagaimana sistem bekerja secara internal dan mengujinya berdasarkan pengetahuan tersebut. Whitebox juga dikenal sebagai structural testing atau glass-box testing. Secara keseluruhan, kedua metode ini saling melengkapi dalam proses pengembangan perangkat lunak. Whitebox testing membantu dalam menemukan kesalahan internal pada kode,



sedangkan blackbox testing memastikan bahwa sistem bekerja sesuai harapan dari perspektif pengguna.

Gambar 2. 3 *White box testing*

Ciri-ciri Pengujian *White Box* Testing:

- Pengujian berbasis pengetahuan: Penguji mengetahui cara kerja internal sistem dan menggunakan informasi ini untuk membuat tes yang tepat.
- Fokus pada struktur: Metode ini menguji arsitektur, aliran data, jalur eksekusi, dan logika kode yang digunakan dalam program.
- Deteksi bug mendalam: Dapat menemukan bug yang tersembunyi dalam kode, seperti kesalahan logika atau cacat dalam aliran kontrol program.
- Alat dan teknik: Teknik yang umum digunakan dalam whitebox testing termasuk path testing, control flow testing, data flow testing, dan branch testing.

Berikut adalah langkah-langkah umum dalam pengujian *white box*:

- Analisis Kode Sumber: Pelajari kode sumber aplikasi untuk memahami struktur internal dan alur logika. Identifikasi fungsi, alur kontrol, dan alur data.
- Identifikasi Kasus Uji: Berdasarkan analisis kode, identifikasi berbagai kondisi, cabang, dan jalur dalam kode. Ini bisa termasuk semua jalur eksekusi, kondisi logika, dan pernyataan dalam kode.

- **Desain Kasus Uji:** Buat kasus uji yang mencakup semua jalur dan kondisi yang diidentifikasi. Pastikan kasus uji dirancang untuk menguji semua cabang dan jalur eksekusi yang mungkin terjadi.
- **Buat Skrip Uji:** Kembangkan skrip uji atau prosedur uji yang diperlukan untuk menjalankan kasus uji. Skrip ini akan menjalankan pengujian sesuai dengan desain yang telah dibuat.
- **Jalankan Kasus Uji:** Eksekusi kasus uji dengan menggunakan skrip uji yang telah dibuat. Pastikan untuk memeriksa hasil dan bandingkan dengan hasil yang diharapkan.
- **Tulis Laporan Hasil Uji:** Dokumentasikan hasil uji, termasuk hasil yang berhasil dan masalah yang ditemukan. Laporkan masalah yang memerlukan perhatian lebih lanjut.
- **Perbaiki dan Uji Ulang:** Jika ditemukan masalah, koordinasikan dengan pengembang untuk memperbaikinya. Setelah perbaikan dilakukan, jalankan uji ulang untuk memastikan bahwa masalah telah teratasi dan tidak ada masalah baru yang muncul.
- **Lakukan Pengujian Tambahan:** Jika perlu, lakukan pengujian tambahan untuk memastikan bahwa perubahan atau perbaikan tidak memengaruhi bagian lain dari aplikasi.
- **Evaluasi dan Penutupan:** Setelah semua pengujian selesai, evaluasi keseluruhan hasil pengujian dan pastikan semua kasus uji telah berhasil. Dokumentasikan proses pengujian secara menyeluruh dan tutup fase pengujian.

Contoh penggunaan *white box testing* sebagai berikut:

- Menguji bagaimana variabel diproses dalam suatu program.
- Memeriksa jalur logika dalam algoritma, seperti memastikan semua cabang if-else dieksekusi.
- Digunakan dalam pengujian unit (*unit testing*), di mana pengembang memeriksa bagian kecil dari kode, seperti fungsi atau modul.

2. **Black Box Testing**

Blackbox testing adalah metode pengujian di mana penguji tidak memiliki pengetahuan atau akses ke kode sumber atau struktur internal dari sistem yang diuji. Penguji hanya mengetahui apa yang seharusnya dilakukan oleh sistem berdasarkan spesifikasi dan input-output, tanpa mengetahui bagaimana cara kerja internalnya. Secara keseluruhan, kedua metode ini saling melengkapi dalam proses pengembangan perangkat lunak. Whitebox testing membantu dalam menemukan kesalahan internal pada kode, sedangkan blackbox testing memastikan bahwa sistem bekerja sesuai harapan dari perspektif pengguna.



Gambar 2. 4 Black box testing

Ciri-ciri pengujian Blackbox testing:

- Pendekatan berbasis fungsional: Pengujian dilakukan berdasarkan fungsionalitas eksternal sistem tanpa memperhatikan bagaimana sistem tersebut diprogram.
- Fokus pada hasil: Penguji memvalidasi bahwa sistem bekerja sesuai dengan yang diharapkan berdasarkan input yang diberikan dan keluaran yang dihasilkan.
- Pengujian pengguna akhir: Blackbox testing sering kali mirip dengan cara pengguna akhir akan menggunakan perangkat lunak.
- Alat dan teknik : Teknik umum termasuk equivalence partitioning, boundary value analisis dan decision table testing.

Berikut adalah langkah-langkah umum dalam pengujian black box:

1. Memahami Spesifikasi dan Persyaratan:

- Pelajari dokumentasi spesifikasi fungsional, persyaratan sistem, dan dokumen desain untuk memahami apa yang diharapkan dari sistem.
- Identifikasi fungsionalitas utama, alur kerja, dan kebutuhan pengguna.

2. Perencanaan Pengujian:

- Buat rencana pengujian yang mencakup tujuan pengujian, ruang lingkup, sumber daya yang diperlukan, dan jadwal.
- Tentukan jenis pengujian black box yang akan dilakukan, seperti pengujian fungsional, pengujian batas, pengujian komponen, dll.

3. Desain Kasus Uji:

- Buat kasus uji berdasarkan spesifikasi fungsional dan persyaratan sistem. Pastikan untuk mencakup berbagai skenario pengguna, input yang valid, dan input yang tidak valid.
- Rancang kasus uji untuk menguji fungsi-fungsi sistem, alur kerja, dan interaksi antar komponen.

4. Persiapkan Lingkungan Pengujian:

- Siapkan lingkungan pengujian yang mencakup perangkat keras, perangkat lunak, dan konfigurasi yang diperlukan untuk menjalankan pengujian.
- Pastikan semua data yang diperlukan dan konfigurasi sistem sesuai dengan spesifikasi pengujian.

5. Jalankan Kasus Uji:

- Eksekusi kasus uji yang telah dirancang. Masukkan data uji dan amati bagaimana sistem merespons.
- Bandingkan hasil yang diperoleh dengan hasil yang diharapkan untuk menentukan apakah fungsionalitas sistem berfungsi dengan baik.

6. Dokumentasikan Hasil Uji:

- Catat hasil pengujian, termasuk kasus uji yang berhasil dan yang gagal.
- Dokumentasikan setiap masalah yang ditemukan, lengkap dengan deskripsi masalah, langkah-langkah untuk mereproduksi, dan hasil yang diharapkan versus yang diperoleh.

7. Tindak Lanjut dan Perbaikan:

- Laporkan masalah kepada tim pengembangan untuk diperbaiki.
- Setelah perbaikan dilakukan, jalankan uji ulang untuk memastikan masalah telah teratasi dan tidak ada masalah baru.

8. Pengujian Regresi:

- Lakukan pengujian regresi untuk memastikan bahwa perbaikan atau perubahan tidak memengaruhi bagian lain dari sistem yang sudah ada.

9. Evaluasi dan Penutupan:

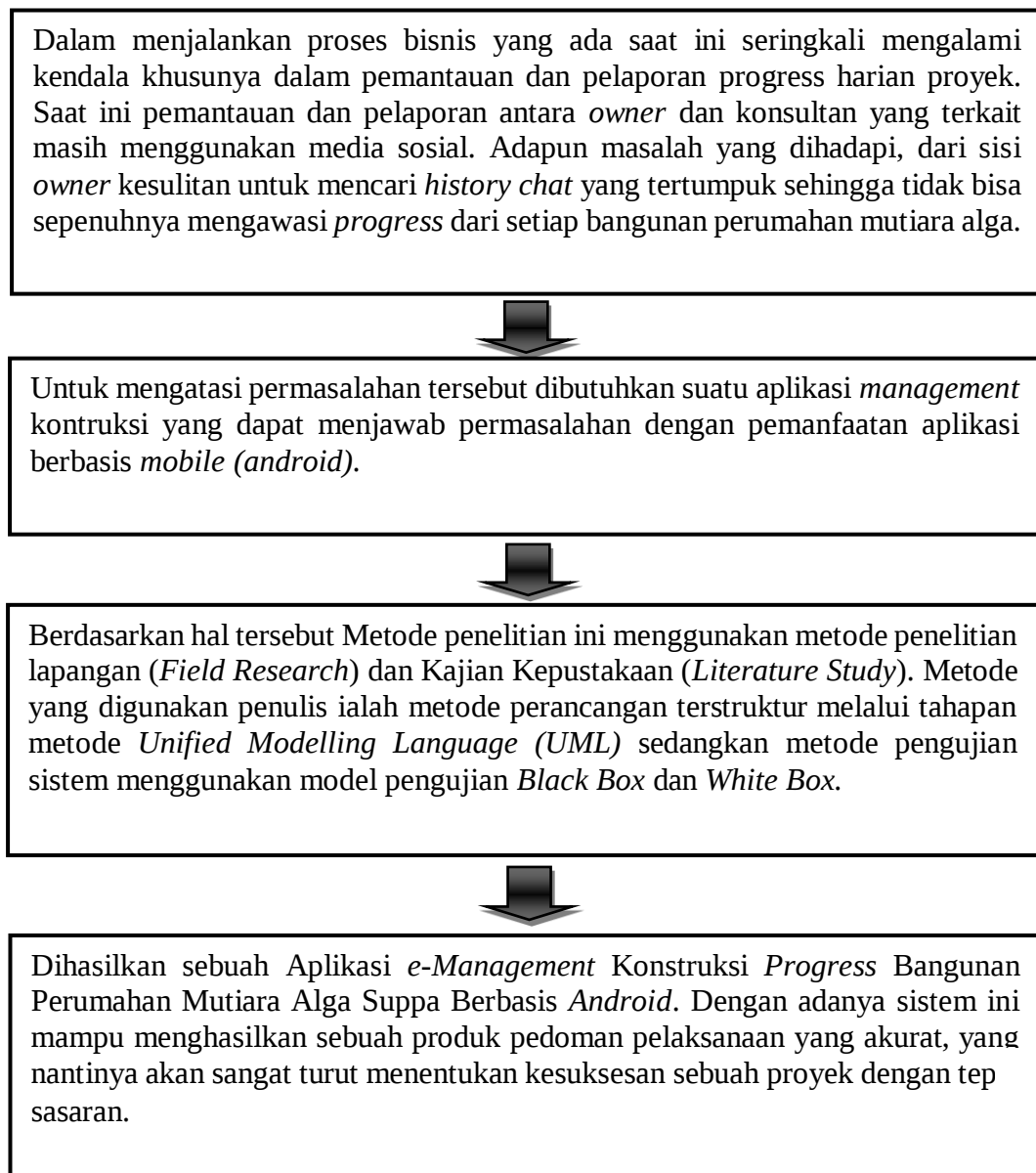
- Tinjau hasil pengujian secara keseluruhan dan pastikan semua kasus uji telah dieksekusi.
- Dokumentasikan proses pengujian dan hasilnya secara menyeluruh. Persiapkan laporan akhir yang mencakup ringkasan temuan dan rekomendasi.

Contoh penggunaan black box testing sebagai berikut:

- Menguji sebuah aplikasi berdasarkan input yang diberikan dan memeriksa apakah output sesuai dengan spesifikasi.
- Digunakan dalam pengujian sistem (system testing) atau pengujian penerimaan (acceptance testing), di mana penguji memvalidasi bahwa aplikasi berfungsi sesuai dengan kebutuhan pengguna.

10. Kerangka Pikir

Untuk lebih memperjelas suatu kerangka berpikir yang akan ditulis dalam penelitian akhir ini. Maka berikut ini digambarkan kerangka pikir tersebut pada gambar 2.4.



BAB III

METODOLOGI PENELITIAN

A. Tempat dan Waktu Penelitian

1. Penelitian yang dilakukan oleh penulis bertempat di CV. Mutiara Alga Suppa Jl. Kebangkitan No. 01, Dusun Polewali, Kec. Suppa, Kabupaten Pinrang, Sulawesi Selatan 91272.
2. Pelaksanaan penelitian ini dilakukan selama ($\pm$) 3 bulan yakni bulan Februari – April 2024.

B. Metode Penelitian

Agar dalam perancangan sistem lebih terarah dan terfokus di dalam mendapatkan data dan informasi yang akurat yang berhubungan dengan pokok pembahasan maka pengumpulan data dilakukan dengan tiga cara adalah metode *literature riview*, wawancara, observasi dan kajian pustaka.

1. Penelitian Kepustakaan (*Literature riview*)

Penulis melakukan studi kepustakaan dengan mencari informasi yang akurat sesuai dengan judul tugas akhir menggunakan beberapa teori, literatur, jurnal, buku dan kepustakaan. Penulis berharap agar program aplikasi yang dibuat nantinya tidak menyimpang dari defenisi-defenisi yang ada sehingga benar-benar memiliki acuan yang sesuai dan lengkap. Studi literatur merupakan salah satu metode pengumpulan data untuk mendapatkan data-data yang sifatnya teoritis yaitu dengan cara membaca literatur yang relevan dengan pengamatan yang penulis lakukan.

2. Wawancara (*Interview*)

Wawancara merupakan teknik pengumpulan data yang dilakukan melalui tatap muka dan tanya jawab langsung antara pengumpul data terhadap narasumber/

sumber data. Wawancara dilakukan untuk mendapatkan informasi yang akurat sesuai dengan data yang dibutuhkan dimana pengumpulan data dilakukan penelitian wawancara langsung dengan pihak CV. Mutiara Alga Suppa, yakni dengan cara mengadakan tanya jawab langsung kepada general direktur, *consultant* dan karyawan atau staff CV. Mutiara Alga Suppa, yang dapat memberikan penjelasan langsung ataupun data sebagai pelengkap penelitian ini.

3. Pengamatan (*Observation*)

Studi lapangan (*observasi*) merupakan teknik pengumpulan data dengan langsung terjun ke lapangan untuk mengamati permasalahan yang terjadi secara langsung di tempat kejadian secara sistematis kejadian-kejadian, perilaku, objek-objek yang dilihat dan hal-hal lain yang diperlukan dalam mendukung penelitian yang sedang berlangsung. Dalam penelitian ini, peneliti melakukan pengamatan langsung ke lokasi-lokasi yang dianggap perlu.

C. Metode Pengumpulan Data

Pada tahap ini peneliti melakukan wawancara dengan pihak yang terkait dengan penelitian, serta melakukan pencatatan dan pengamatan (*observation*) langsung di tempat penelitian. Adapun uraian dari tahapan tersebut adalah sebagai berikut :

1. Persiapan Penelitian

Pada tahapan ini peneliti melakukan persiapan penelitian. Persiapan penelitian yang dimaksud adalah menyiapkan buku-buku, artikel-artikel tentang topik penelitian serta *software* yang digunakan selama penelitian.

2. Studi *Literature*

Pada tahapan ini peneliti melakukan apa yang disebut dengan kajian pustaka, yaitu mempelajari buku-buku referensi dan hasil penelitian sejenis sebelumnya yang pernah dilakukan oleh orang lain.

3. Pengumpulan Data

Pada tahap ini peneliti melakukan wawancara dengan pihak yang terkait dengan penelitian, serta melakukan pencatatan dan pengamatan langsung di tempat penelitian.

4. Analisis

Pada tahap analisis, peneliti melakukan analisa terhadap sistem yang di terapkan sekarang berdasarkan kemudian merumuskan masalah yang menjadi pokok penelitian sehingga dapat dibuat alternatif pemecahan masalah.

5. Perancangan

Peneliti kemudian merancang aplikasi yang ingin dibuat berdasarkan alternatif pemecahan masalah.

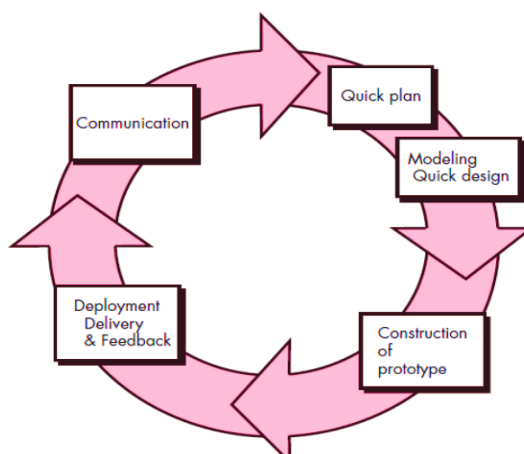
6. Pengujian

Setelah melakukan perancangan, peneliti kemudian menguji hasil perancangan yang telah dibuat. Jika hasil perancangan terdapat kekurangan atau kelemahan maka kembali ke tahap analisis.

D. Metode Pengembangan Sistem

Metode Pengembangan sistem merupakan suatu kerangka kerja mengenai bagaimana proses pelaksanaan Aplikasi *e-Management* Konstruksi Dalam Pencegahan *Rework* Pada *Progress* Bangunan Perumahan Mutiara Alga Suppa Berbasis *Android*, adapun metode pengembangan yang digunakan yaitu *Prototype* hal ini dikarenakan pada CV. Mutiara Alga Suppa belum sama sekali memiliki aplikasi yang berkaitan dengan manajemen konstruksi pada bangunan perumahan sehingga sangat cocok menggunakan metode *prototype*.

Menurut Pressman (2012), dalam melakukan perancangan sistem yang akan dikembangkan dapat menggunakan metode *prototype*. Metode ini cocok digunakan untuk mengembangkan perangkat-perangkat lunak yang akan dikembangkan kembali. Metode ini dimulai dengan pengumpulan kebutuhan pengguna, dalam hal ini pengguna dari perangkat yang dikembangkan adalah Aplikasi *e-Management* Konstruksi Dalam Pencegahan *Rework* Berbasis *Android*. Tahap berikutnya membuat rancangan kilat yang selanjutnya akan di evaluasi kembali sebelum diproduksi secara benar. Metode pengembangan sistem yang digunakan adalah model *prototype* dengan 5 tahapan yaitu *communication*, *quick plan*, *modeling* *quick design*, *construction of prototipe*, *deployment, delivery & feedback*.



Gambar 3.1 Model *Prototype*

Berikut adalah tahapan-tahapan dalam metode *prototype* :

1. *Communication*

Pada tahap ini, antara pelanggan dengan tim pengembang perangkat lunak berkomunikasi mengenai spesifikasi kebutuhan yang diinginkan. Tim pengembang melakukan pengumpulan data yang akan digunakan dalam pembuatan sistem yang dibutuhkan.

2. *Quick Plan*

Tim pengembang memberikan gambaran besar mengenai sistem yang akan dibuat kepada *general manager* dan spv. Dalam hal ini sistem yang akan dibuat adalah Aplikasi *e-Management* Konstruksi Dalam Pencegahan *Rework* Pada *Progress* Bangunan Perumahan Mutiara Alga Suppa Berbasis *Android*, kemudian pengembang merancang penggunaan spesifikasi *hardware* dan *software* yang akan digunakan dalam pembuatan sistem.

3. *Modeling Quick Design*

Pada tahap ini, pengembang membuat perancangan sistem menggunakan *use case diagram*, *activity diagram*, *sequence diagram* dan *user interface*.

4. *Construction of Prototype*

Setelah tahap pemodelan, maka pengembang mulai melakukan pengkodean program. Bahasa pemrograman yang digunakan adalah *PHP* dan *JavaScript* menggunakan database *MySQL* dengan tools *PhpMyAdmin*.

5. *Deployment, Delivery & Feedback*

Tahap pengkodean program dibarengi oleh tahapan implementasi dan pengujian sistem. Sistem diuji dengan metode *white box* dan *black box* testing dengan pengujian fungsional. Setelah sistem lulus uji coba, maka sistem yang baru diberikan kepada pengguna dan pengembang mengajari pengguna dalam penggunaan sistem yang baru. Jika ada kekurangan atau penambahan kebutuhan sistem, maka pihak terkait akan mengkomunikasikan kembali dengan tim pengembang.

E. Sumber Data Penelitian

Adapun jenis dan sumber data yang dibutuhkan dalam penelitian ini adalah sebagai berikut :

1. Data Primer yaitu data yang diperoleh melalui wawancara yang dilakukan langsung dengan responden yang dapat mewakili beberapa Sumber data. Dalam penelitian ini merupakan subjek penelitian utama yakni CV. Mutiara Alga.
2. Data sekunder yaitu merupakan data yang diperoleh melalui studi kepustakaan atau dari berbagai literatur dengan menelaah buku-buku dan tulisan-tulisan atau internet, jurnal hukum, serta peraturan perundang-undangan yang relevan dengan permasalahan yang diteliti.

F. Alat dan Bahan

Tahap desain dilakukan untuk menentukan ruang lingkup pembuatan aplikasi yang ditangani, yang selanjutnya dijabarkan dalam bentuk diagram alir. Dalam pembuatan program aplikasi tersebut penulis membutuhkan perangkat keras

(*Hardware*) dan perangkat lunak (*Software*). Adapun *Hardware* dan *Software* yang digunakan adalah :

1. Perangkat keras (*hardware*)

Perangkat keras yang digunakan untuk membuat aplikasi menggunakan PC (*Personal Komputer*) dengan spesifikasi yang dapat dilihat pada tabel 3.1 berikut :

Table 3.1 Spesifikasi Perangkat Keras

Jenis	Spesifikasi
Laptop	Asus
Processor	Intel(R) Core (TM) i3-7500 U CPU @ 4.0 Ghz
Ram	4 GB DDR 3
Hardisk	500 GB
Smartphone	Android versi 5.0

2. Perangkat Lunak (*software*)

Perangkat lunak yang digunakan untuk membangun aplikasi ini dapat dilihat pada tabel 3.2 berikut :

Table 3.2 Spesifikasi Perangkat Lunak

Jenis	Spesifikasi
Sistem Operasi	Windows 10 ultimate 64 bit
Bahasa Pemrograman	PHP, Javascript
Basis Data	MySQL
Tools	XAMPP, Visual Studio Code

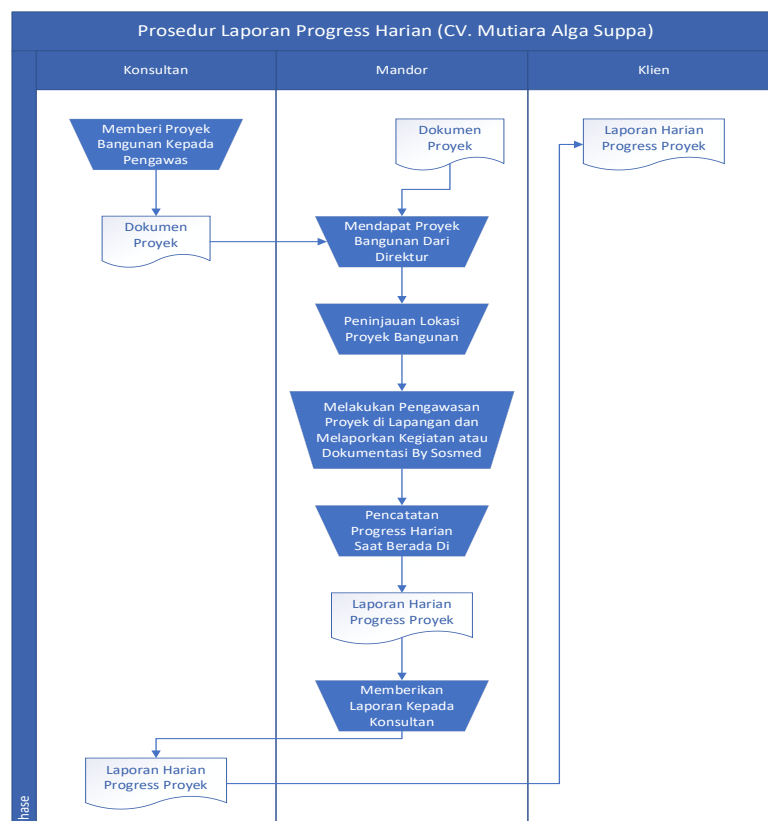
BAB IV

HASIL DAN PEMBAHASAN

A. Rancangan Sistem

1. Analisis Sistem Berjalan

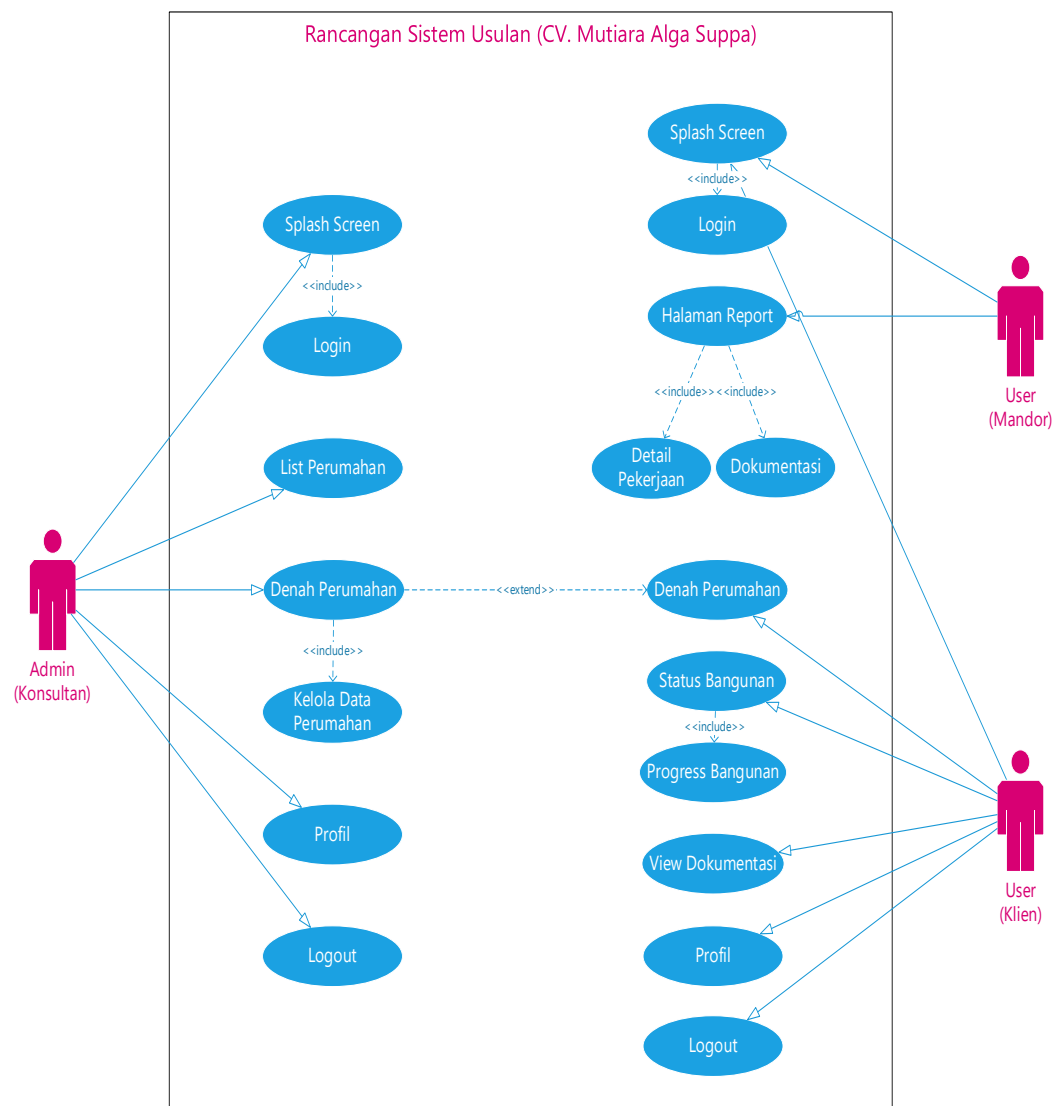
Analisis sistem yang sedang berjalan saat ini di CV. Mutiara Alga Suppa Dalam hal ini masih melakukan proses pencatatan laporan harian yang belum mengikuti perkembangan teknologi, dikarenakan oleh pengawas lapangan di perusahaan tersebut masih menggunakan cara konvensional. Proses pencatatan laporan harian tersebut dilakukan dalam beberapa tahap seperti yang dapat dilihat pada gambar berikut.



Gambar 4.1 Flowmap Diagram Sistem Berjalan

2. Analisis Sistem Diusulkan

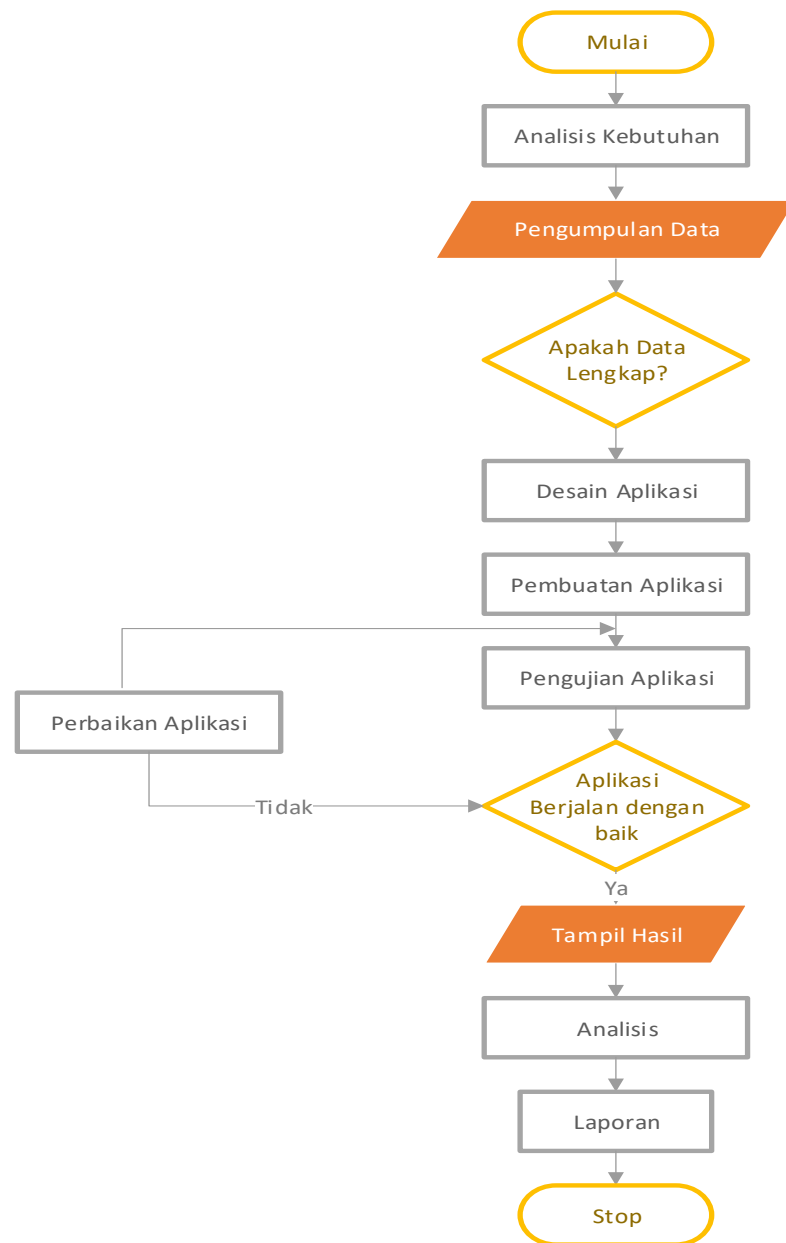
Analisis sistem diusulkan merupakan penguraian dari suatu sistem yang utuh kedalam bagian komponennya untuk mengidentifikasi dan mengevaluasi permasalahan. Bagian analisis terdiri dari analisis masalah dan analisis kebutuhan. Perancangan sistem yang akan digambarkan dengan *use case diagram*, perancangan database dan perancangan antarmuka aplikasi yang akan dibangun dengan berbasis *Android*.



Gambar 4.2 Use Case Diagram Sistem Diusulkan

3. Flowchart

Flowchart merupakan bagan yang menunjukkan alur kerja atau apa yang sedang dikerjakan di dalam sistem secara keseluruhan dan menjelaskan urutan dari prosedur-prosedur yang ada di dalam alur penelitian. Berikut gambaran *flowchart* alur penelitian dapat dilihat sebagai berikut.



Gambar 4.3 Flowchart Alur Penelitian

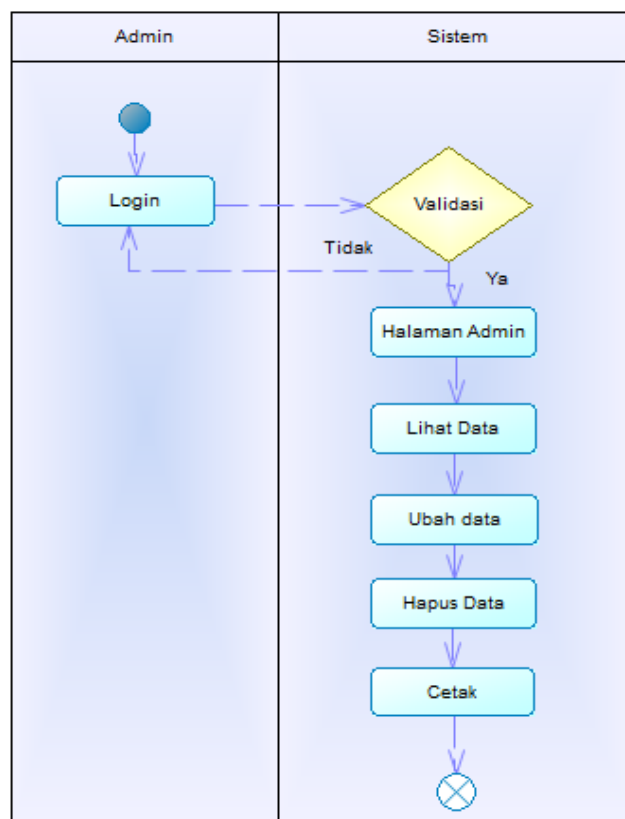
B. Analisis Aliran Data UML

Analisis aliran data yang diusulkan ini bertujuan mengetahui proses informasi, dalam analisis sistem penulis menggunakan pengembangan orientasi objek sehingga menggunakan *Use Case Diagram*, *Activity Diagram*, dan *sequence diagram*.

1. Activity Diagram

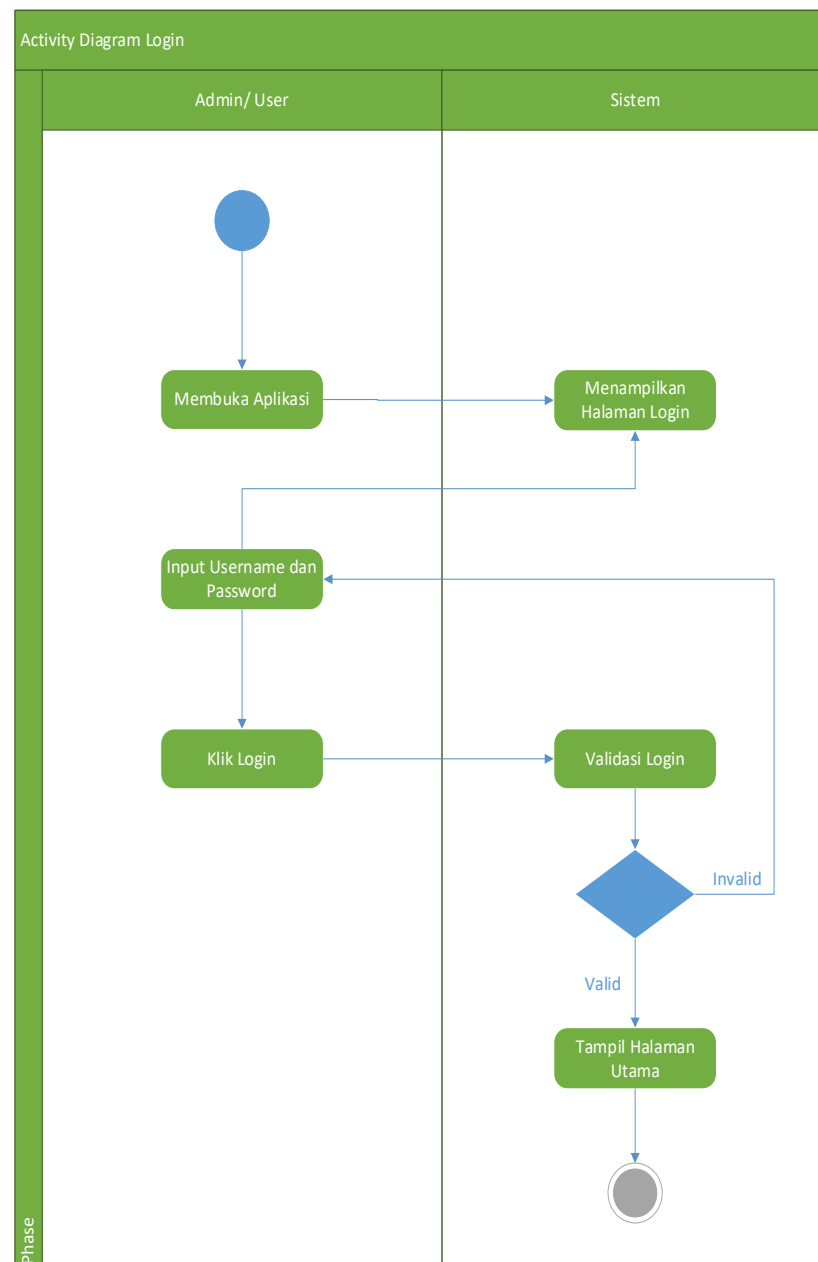
Activity diagram adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. *Activity diagram* ini menjelaskan tentang aktifitas-aktifitas yang terjadi dalam sebuah aliran proses pada sebuah *system*.

a. Activity Diagram Admin



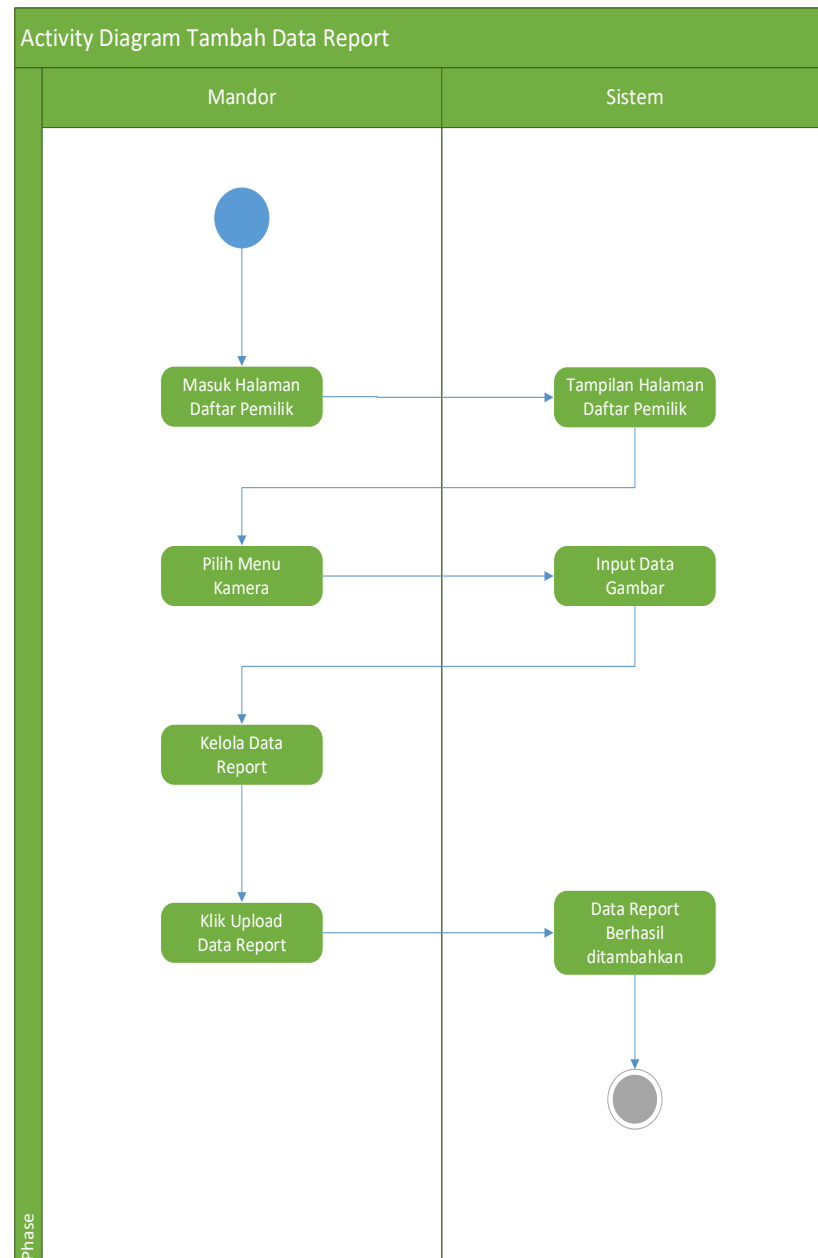
Gambar 4.4 Activity Diagram Admin

b. *Activity Diagram Login*



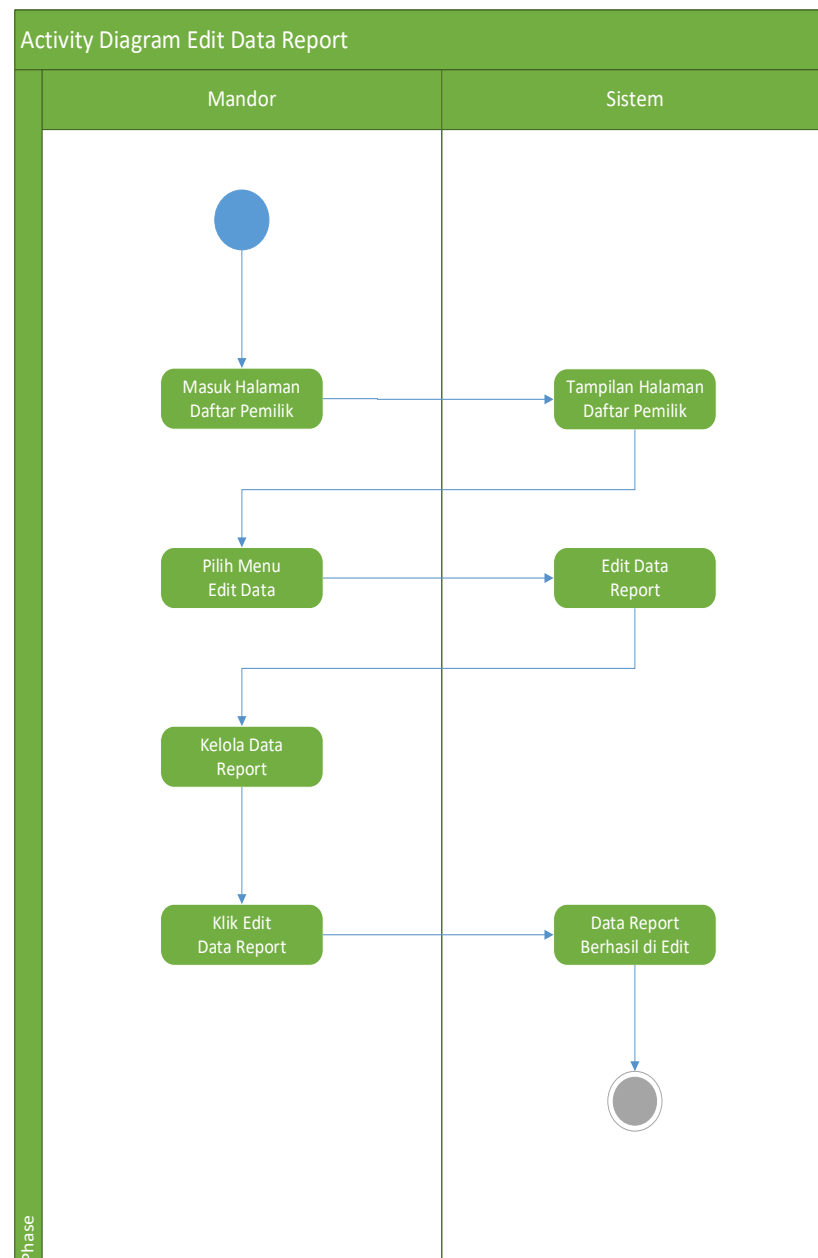
Gambar 4.5 Activity Diagram Login

c. *Activity Diagram Tambah Data Report*

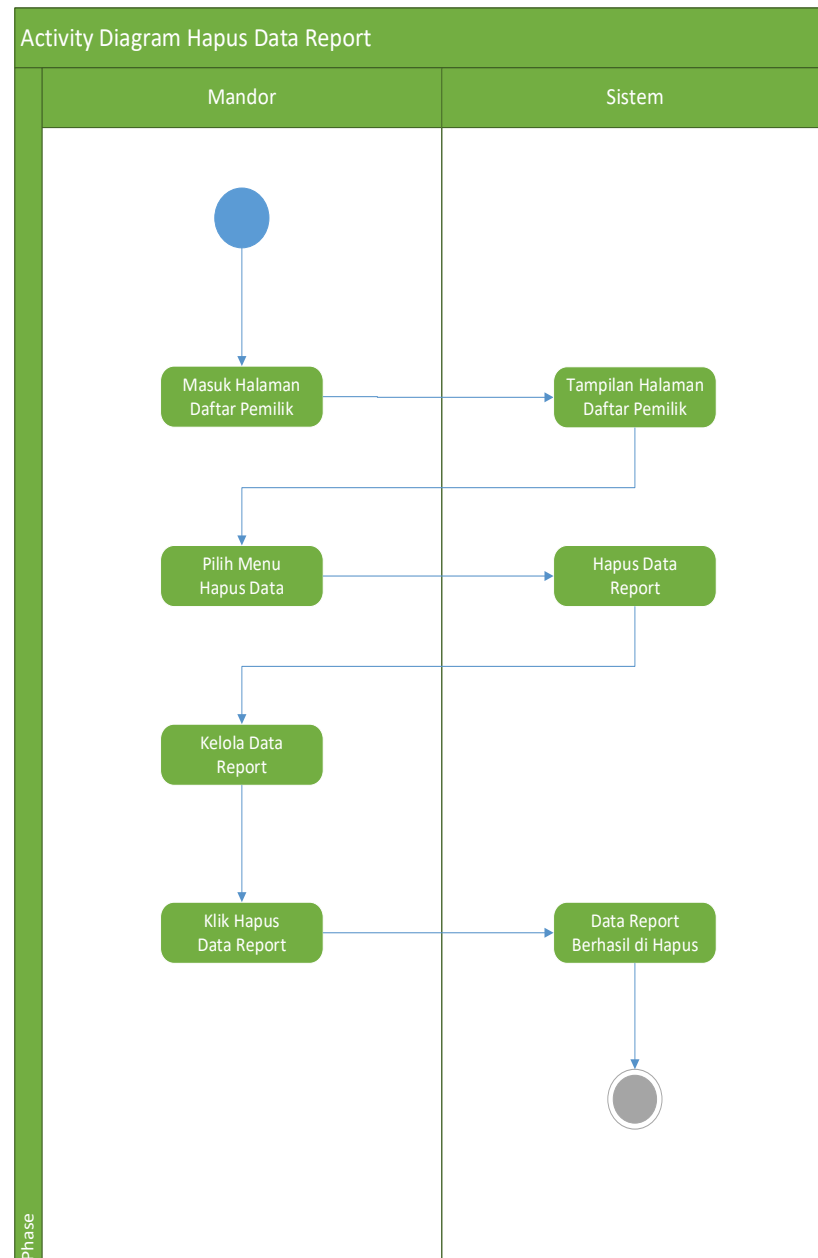


Gambar 4.6 *Activity Diagram Tambah Data Report*

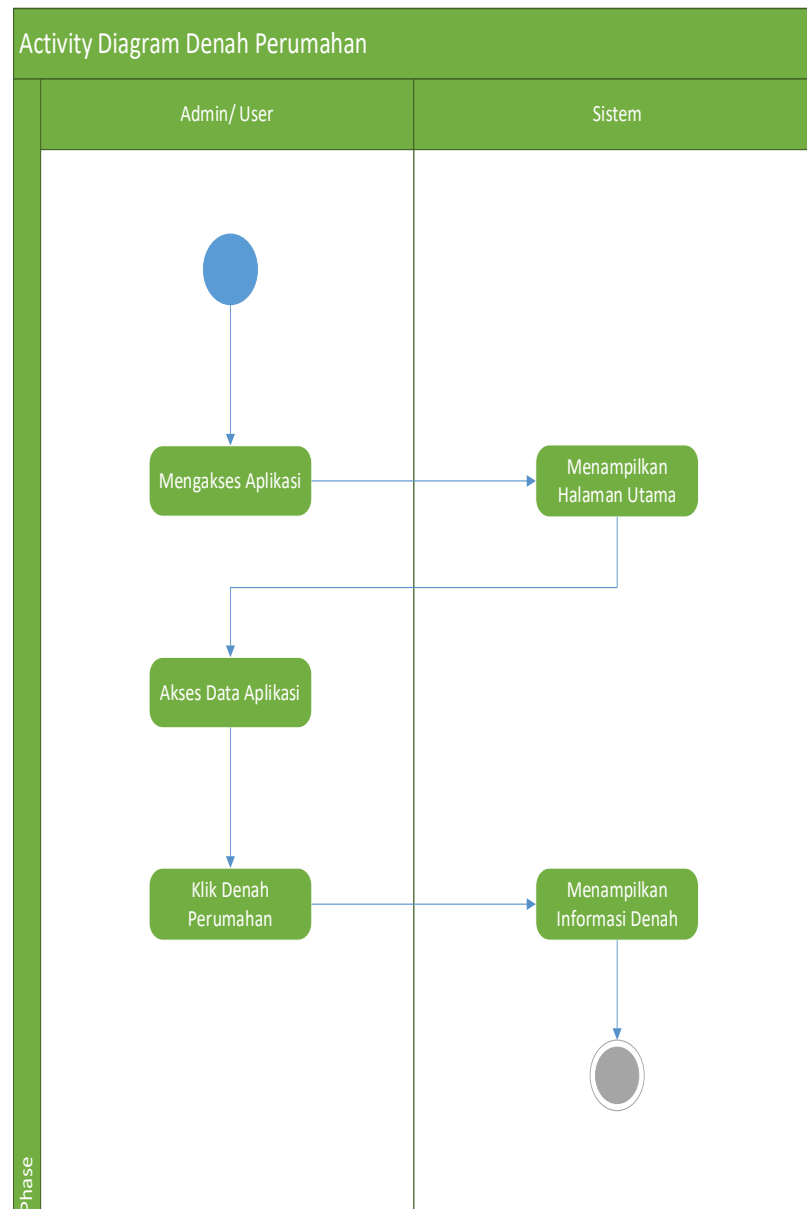
d. *Activity Diagram Edit Data Report*



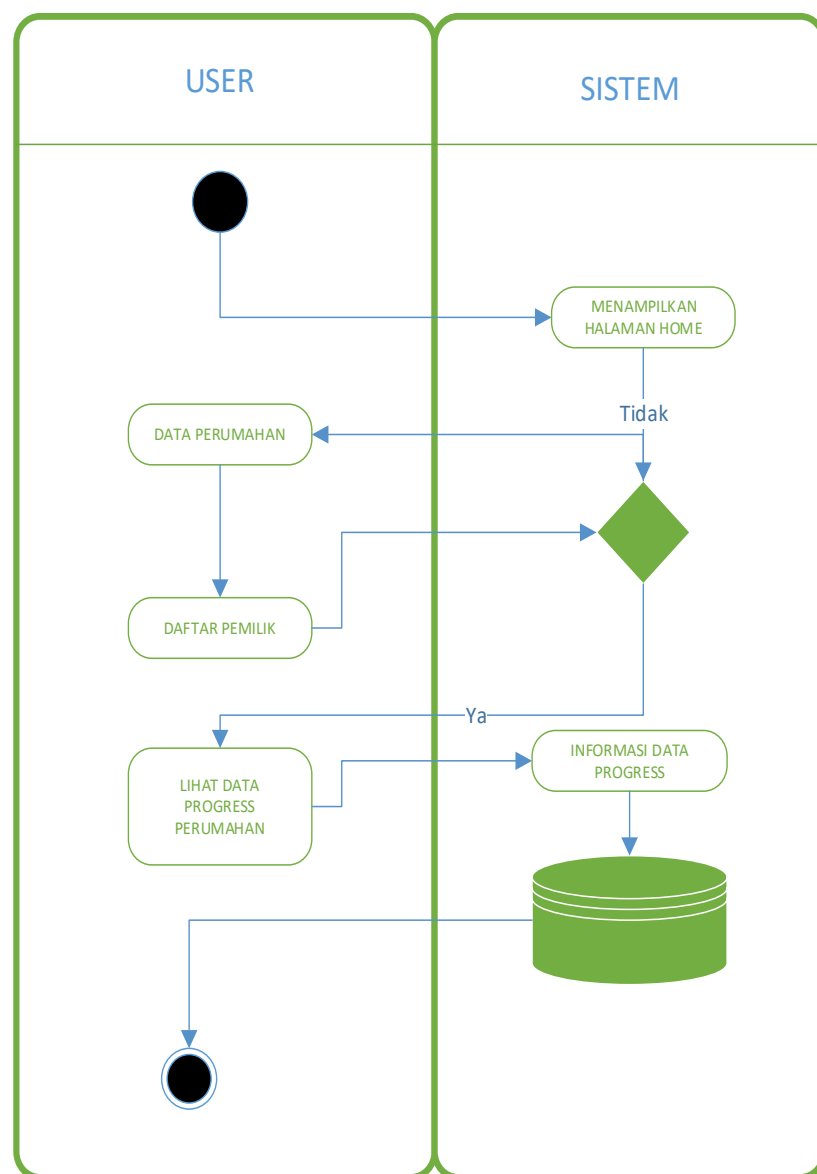
Gambar 4.7 *Activity Diagram Edit Data Report*

e. *Activity Diagram Hapus Data Report*Gambar 4.8 *Activity Diagram Hapus Data Report*

g. *Activity Diagram View Denah Perumahan*



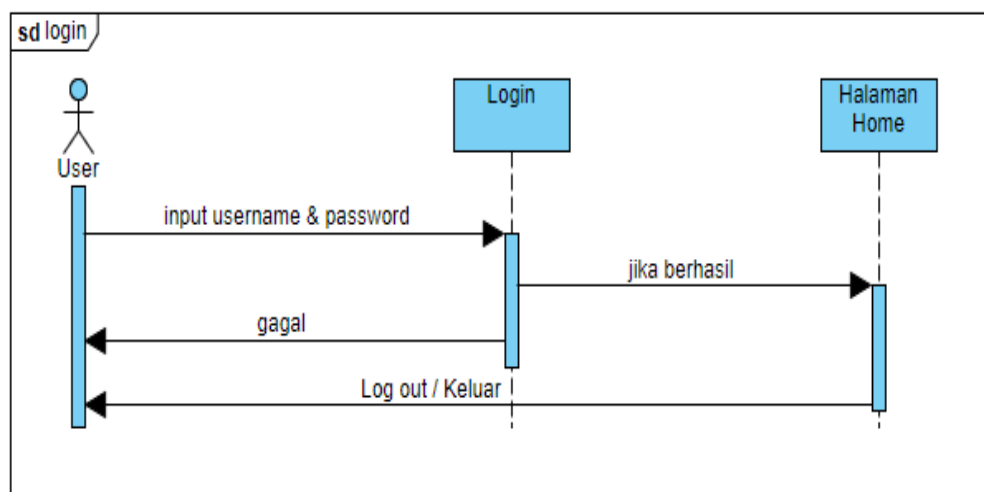
Gambar 4.10 *Activity Diagram View Denah Perumahan*

h. *Activity Diagram User*Gambar 4.11 *Activity Diagram User*

2. Sequence Diagram

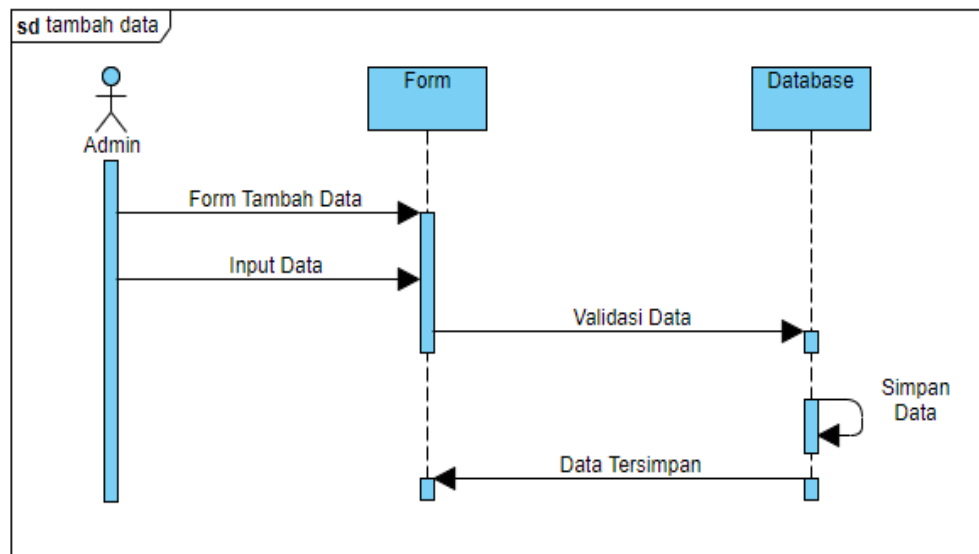
Sequence Diagram merupakan salah satu diagram *Interaction* yang menjelaskan bagaimana suatu operasi itu dilakukan *message* (pesan) apa yang dikirim dan kapan pelaksanaannya.

a. Sequence Diagram Login

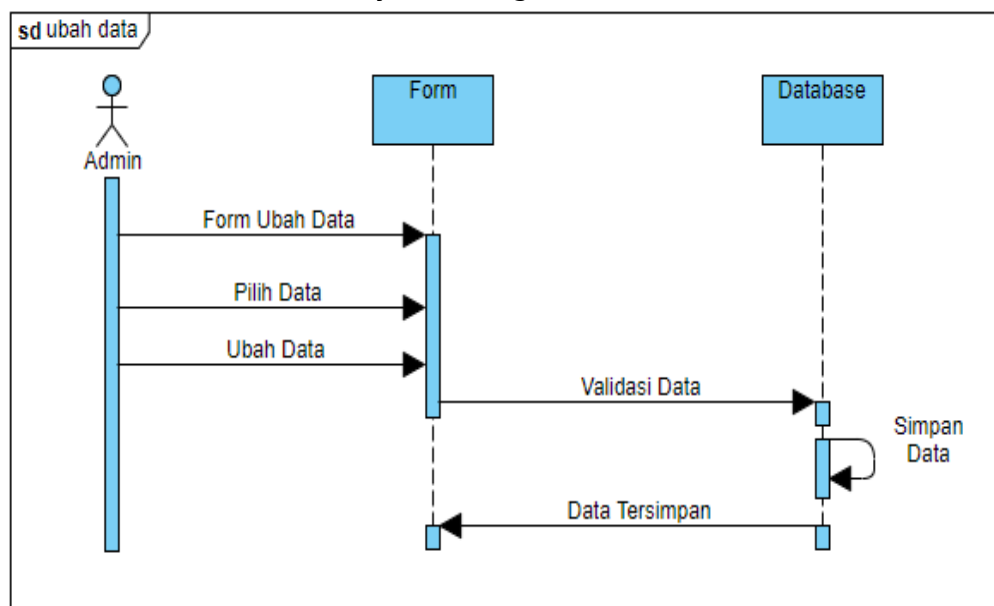


Gambar 4.12 Sequence Diagram Login

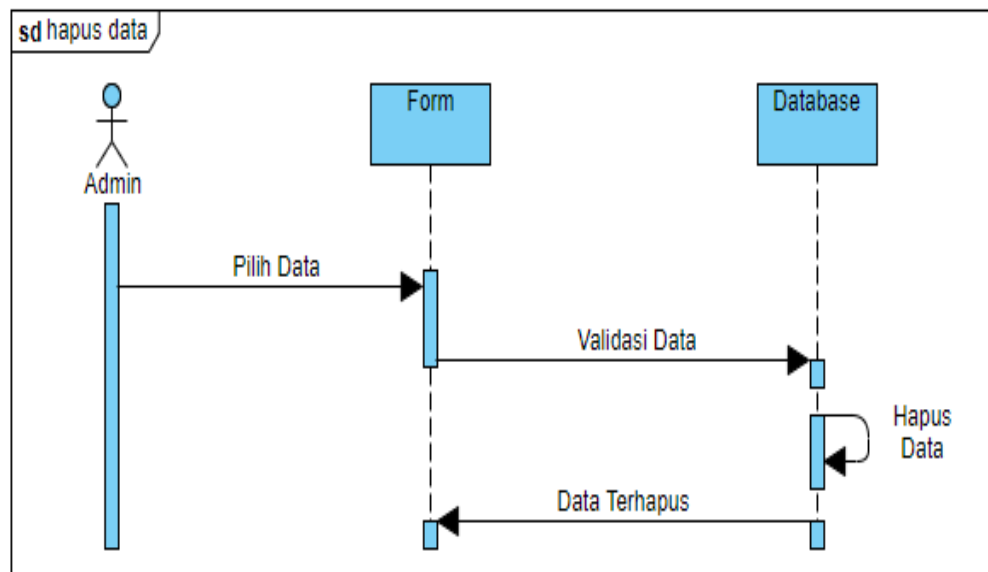
b. Sequence Diagram Admin



Gambar 4.13 *Sequence Diagram Tambah Data Admin*

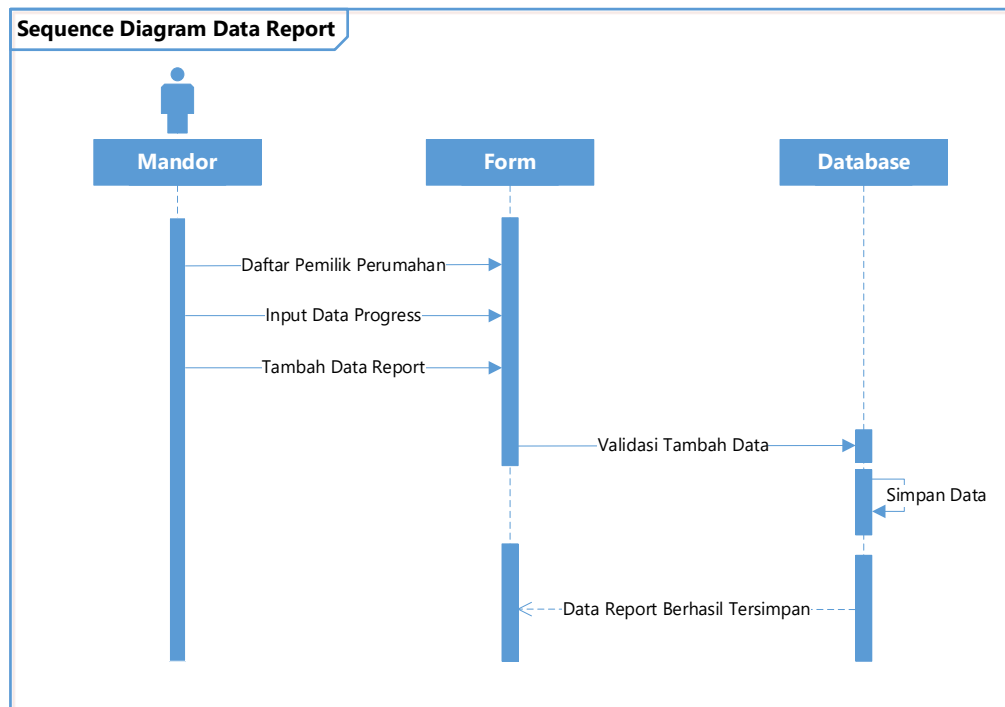


Gambar 4.14 *Sequence Diagram Ubah Data Admin*



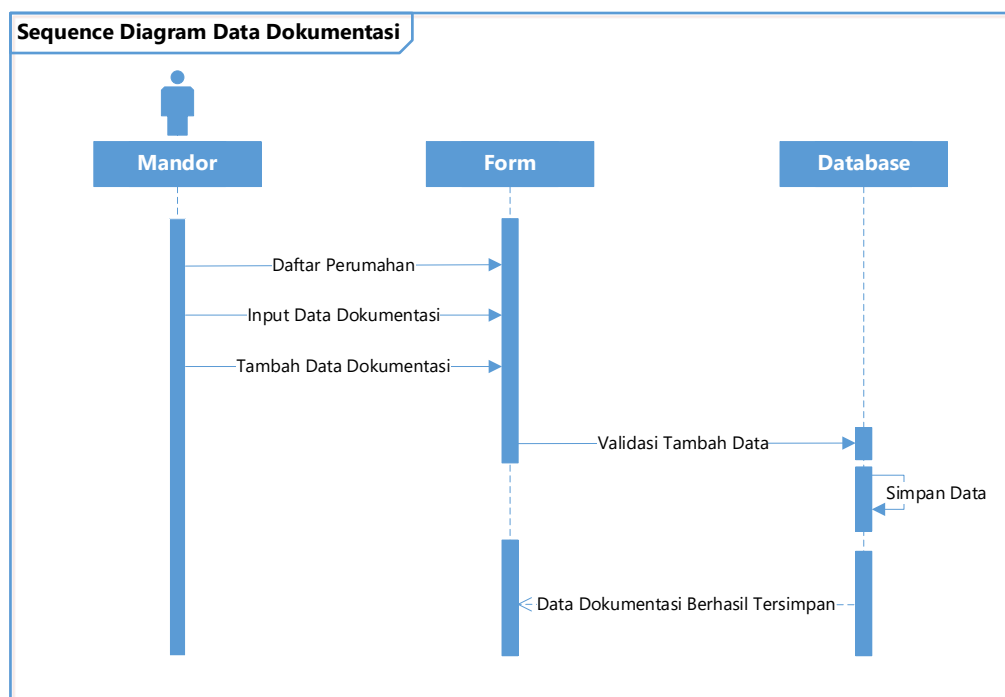
Gambar 4.15 *Sequence Diagram* Hapus Data Admin

c. *Sequence Diagram* Data Report



Gambar 4.16 *Sequence Diagram Data Report*

d. *Sequence Diagram Data Dokumentasi*



Gambar 4.17 *Sequence Diagram Data Dokumentasi*

C. Detail Aplikasi

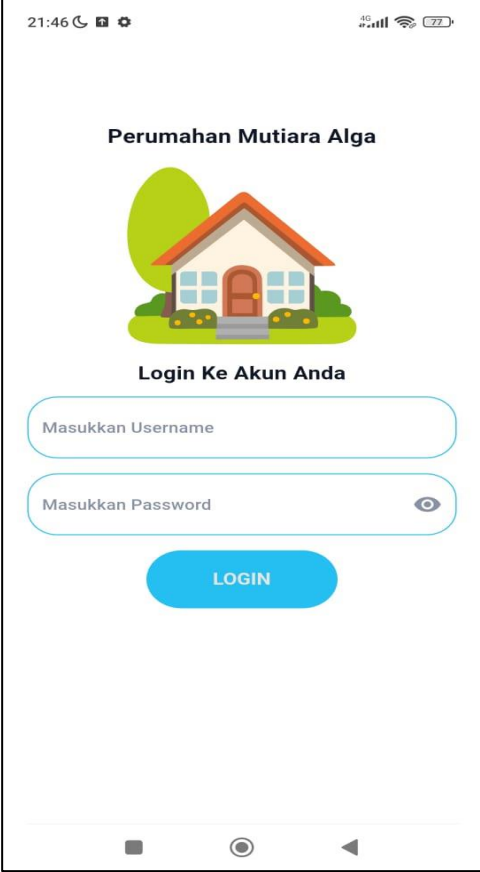
1. Halaman *Splash Screen*



Gambar 4.18 Halaman *Splash Screen*

Pada menu halaman *Splash Screen* merupakan halaman yang berada pada halaman awal saat pertama kali membuka sebuah aplikasi. Tampilan ini dibuat agar aplikasi terlihat lebih terkesan *flat design*. adapun gambar yang menjadi logo merupakan logo dari Perumahan yang menyimbolkan Perumahan Mutiara Alga.

2. Halaman *Login*

The image is a screenshot of a mobile application's login screen. At the top, the status bar shows the time 21:46, signal strength, and battery level. The app's title 'Perumahan Mutiara Alga' is centered at the top. Below the title is a colorful illustration of a small house with a red roof, a brown door, and green bushes. Underneath the illustration, the text 'Login Ke Akun Anda' is displayed. There are two input fields: the first is labeled 'Masukkan Username' and the second is labeled 'Masukkan Password' with an eye icon for toggling visibility. A blue 'LOGIN' button is positioned below the password field. At the bottom of the screen, there are three navigation icons: a square, a circle, and a triangle.

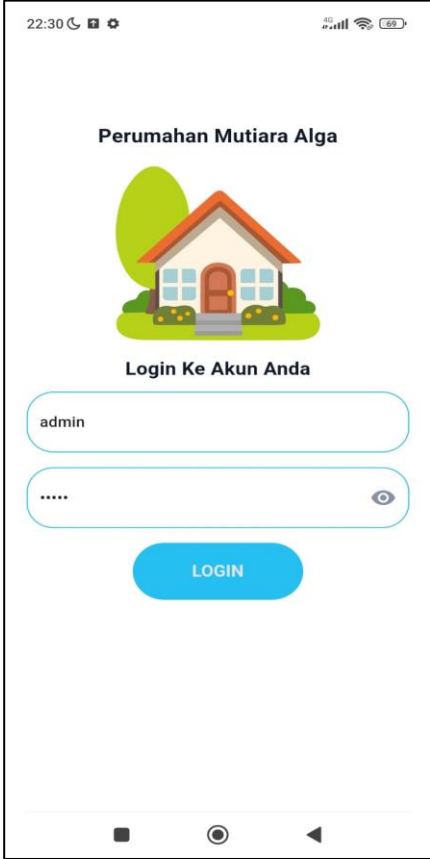
Gambar 4.19 Halaman *Login*

Pada menu halaman *form login* terdapat pengaksesan untuk pengguna. Pada halaman ini merupakan menu *form* yang berada di halaman awal. *Login* merupakan tampilan awal yang ada pada sistem, yang digunakan pengguna untuk masuk ke halaman selanjutnya. *Form Login* digunakan untuk melindungi sistem dari hal-hal yang dapat merusak data ataupun sebagai *security* sistem.

3. Admin

a. Halaman *Login Admin*

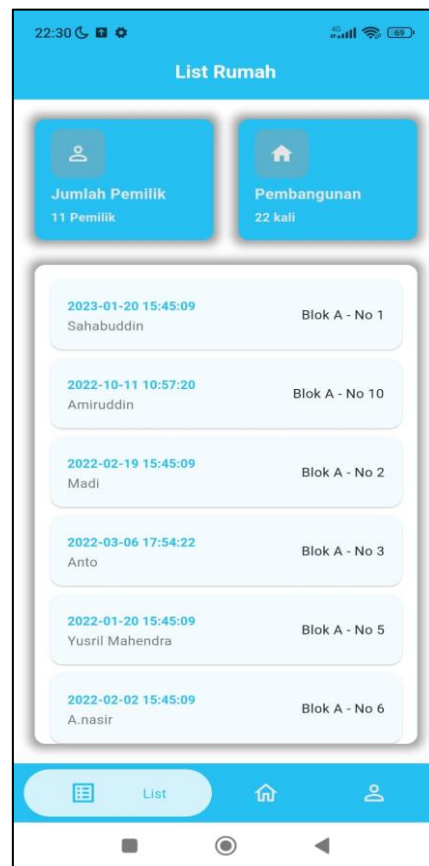
Halaman menu *login* merupakan suatu halaman *login admin*. *Login* merupakan tampilan awal yang ada pada sistem, yang digunakan *admin* untuk masuk ke halaman selanjutnya.

The image is a screenshot of a mobile application's login screen. At the top, the status bar shows the time 22:30, a moon icon, a gear icon, and signal/battery indicators. The app's title 'Perumahan Mutiara Alga' is centered at the top. Below the title is a cartoon illustration of a house with a red roof, a brown door, and green bushes. Under the illustration, the text 'Login Ke Akun Anda' is centered. There are two input fields: the first contains the text 'admin', and the second contains five dots with an eye icon to its right. Below these fields is a blue rounded rectangular button with the text 'LOGIN' in white. At the bottom of the screen, there are three navigation icons: a square, a circle, and a triangle.

Gambar 4.20 Halaman *Form Login Admin*

b. Halaman *Form* List Perumahan

Halaman list perumahan merupakan halaman menu *admin*, halaman ini digunakan *admin* untuk mengelola data dan informasi akun *user* klien. Halaman ini menampilkan berupa informasi Jumlah Pemilik, Pembangunan dan Nama Pemilik Perumahan.



Gambar 4.21 Halaman *Form* List Perumahan

c. Halaman *Form* Denah

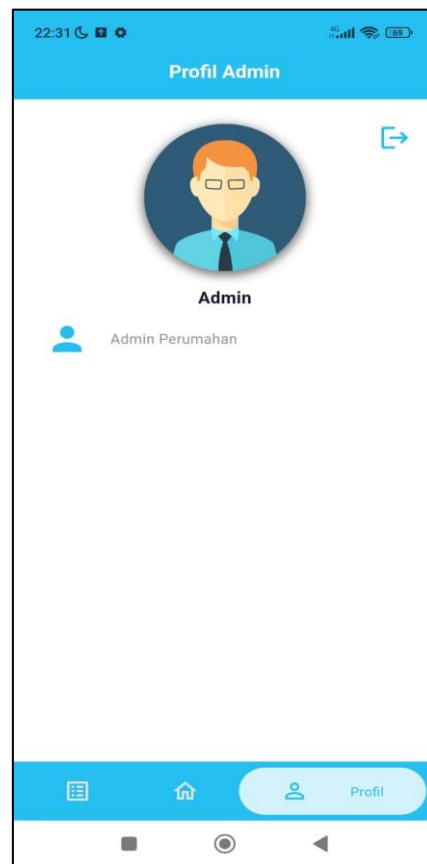
Halaman *form* denah merupakan halaman menu *admin*, halaman ini digunakan *admin* menampilkan gambaran denah perumahan. Halaman ini menampilkan berupa informasi jenis blok perumahan, nomor rumah dan nama pemilik perumahan dalam bentuk denah.



Gambar 4.22 Halaman *Form* Denah

d. Halaman *Form* Profil

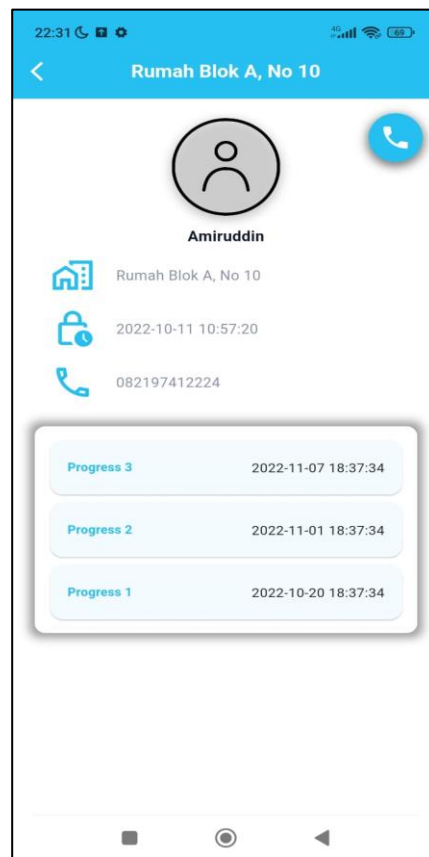
Halaman *form* profil merupakan halaman menu *admin*. Halaman ini menampilkan informasi data akun pengguna yakni *admin*. Pada halaman ini juga sebagai manajemen data profil *admin* serta terdapat fitur simpan data profil pengguna dan *logout*.



Gambar 4.23 Halaman *Form* Profil

e. Halaman Menu Lihat Informasi Pemilik

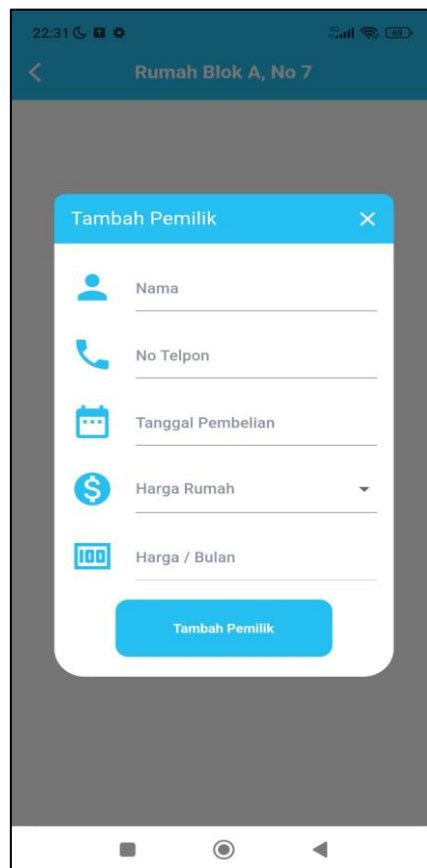
Pada menu lihat informasi pemilik rumah terdapat pengaksesan untuk *admin*. Halaman ini menampilkan detail informasi pemilik rumah berdasarkan nama, status rumah, tanggal dan tahun pembelian, nomor telepon dan progress perumahan.



Gambar 4.24 Halaman Menu Lihat Informasi Pemilik

f. Halaman Menu Tambah Pemilik

Pada menu tambah data pemilik rumah terdapat pengaksesan untuk *admin*. Pada halaman ini juga sebagai manajemen data pemilik rumah berdasarkan nama, nomor telepon, tanggal pembelian, harga rumah dan cicilan perbulan serta terdapat fitur simpan data.



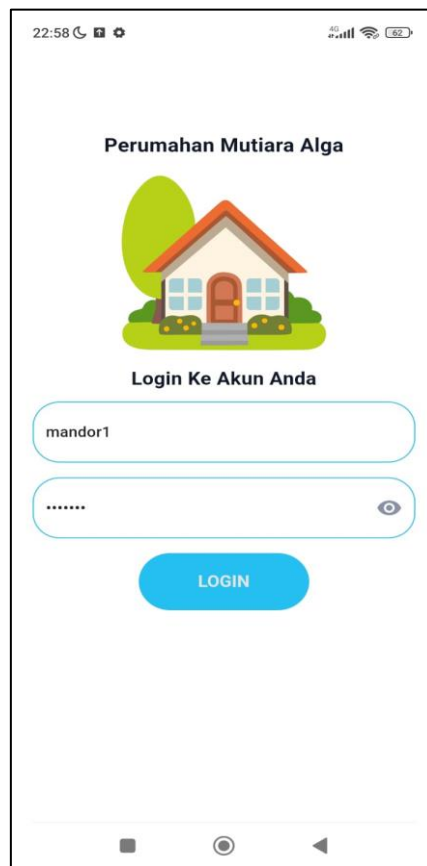
The screenshot shows a mobile application interface. At the top, the status bar displays the time 22:31 and various icons. Below the status bar, a dark blue header bar contains a back arrow and the text 'Rumah Blok A, No 7'. The main content area is a light gray background. Overlaid on this is a white modal form titled 'Tambah Pemilik' with a close button (X) in the top right corner. The form contains five input fields, each with an icon to its left: a person icon for 'Nama', a telephone icon for 'No Telp', a calendar icon for 'Tanggal Pembelian', a dollar sign icon for 'Harga Rumah', and a 100 icon for 'Harga / Bulan'. The 'Harga Rumah' field has a small downward arrow indicating a dropdown menu. At the bottom of the form is a blue button with the text 'Tambah Pemilik'. The bottom of the screen shows the standard Android navigation bar with three icons.

Gambar 4.25 Halaman Menu Tambah Pemilik

4. Mandor (*User*)

a. Halaman *Login* Mandor

Halaman menu *login* merupakan halaman *login user*. *Login* merupakan tampilan awal yang ada pada sistem, yang digunakan *user* untuk masuk ke halaman selanjutnya.



Gambar 4.26 Halaman *Login* Mandor

b. Halaman *Form* List Perumahan Mandor

Halaman list perumahan merupakan halaman menu *mandor*, halaman ini digunakan *mandor* untuk mengelola data dan informasi data perumahan. Halaman ini menampilkan berupa informasi Jumlah Pemilik, Pembangunan dan Nama Pemilik Perumahan.



Gambar 4.27 Halaman *Form* List Perumahan Mandor

c. Halaman *Form* Denah Mandor

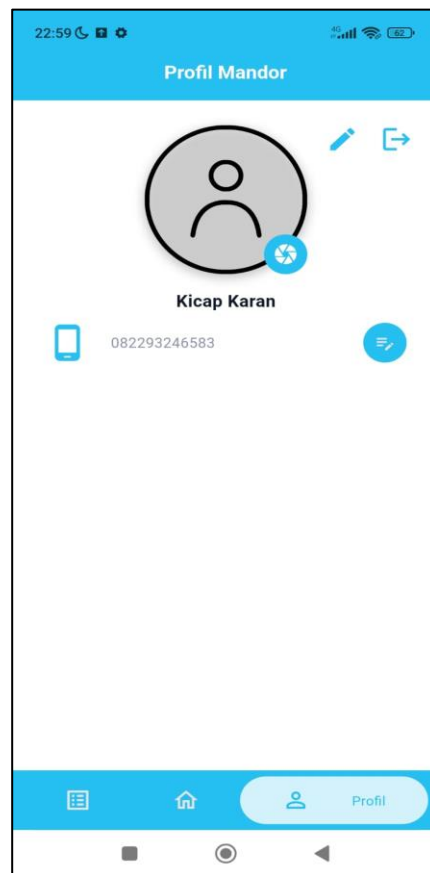
Halaman *form* denah merupakan halaman menu *mandor*, halaman ini digunakan *mandor* menampilkan gambaran denah perumahan. Halaman ini menampilkan berupa informasi jenis blok perumahan, nomor rumah dan nama pemilik perumahan dalam bentuk denah.



Gambar 4.28 Halaman *Form* Denah Mandor

d. Halaman *Form* Profil Mandor

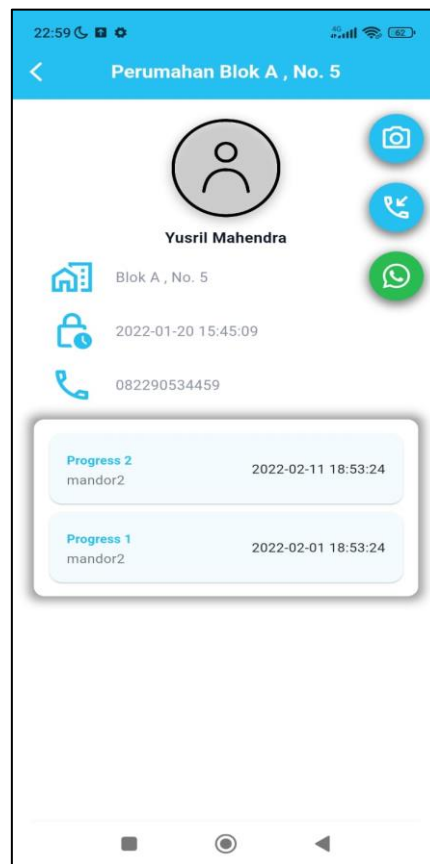
Halaman *form* profil merupakan halaman menu *mandor*. Halaman ini menampilkan informasi data akun pengguna yakni *mandor*. Pada halaman ini juga sebagai manajemen data profil *mandor* serta terdapat fitur simpan data profil pengguna dan *logout*.



Gambar 4.29 Halaman *Form* Profil Mandor

e. Halaman Menu Lihat Informasi Pemilik Mandor

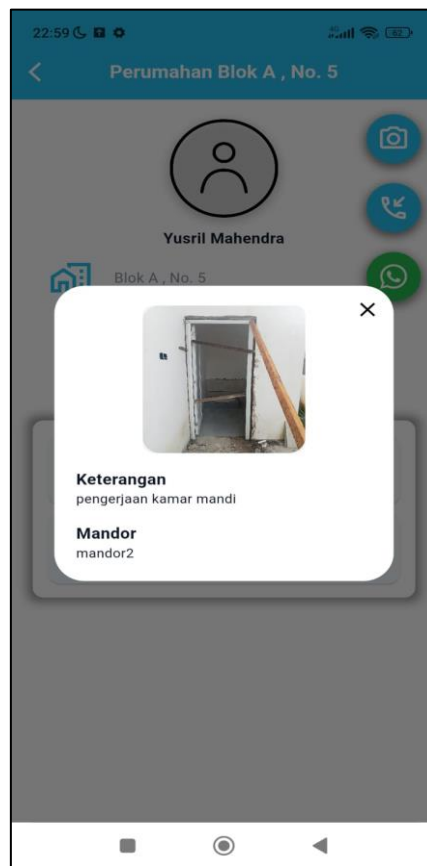
Pada menu lihat informasi pemilik rumah terdapat pengaksesan untuk *mandor*. Halaman ini menampilkan suatu detail informasi pemilik rumah berdasarkan nama, status rumah, tanggal dan tahun pembelian, nomor telepon dan *progress* perumahan.



Gambar 4.30 Halaman Menu Lihat Informasi Pemilik Mandor

f. Halaman Menu Lihat Dokumentasi Mandor

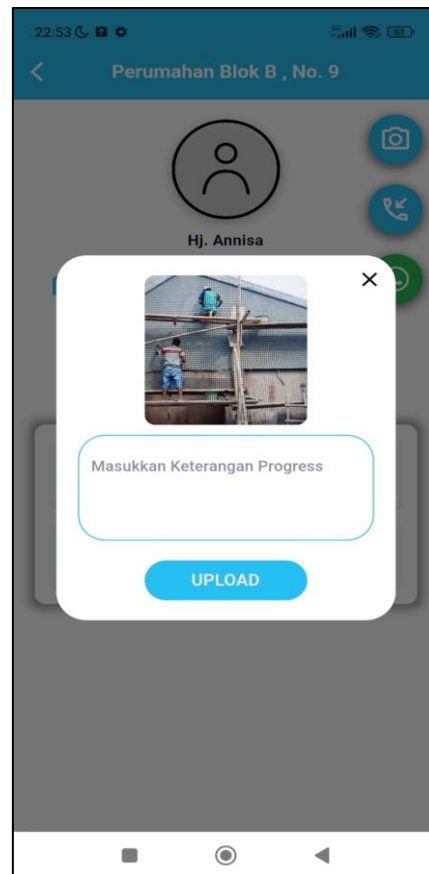
Pada menu lihat dokumentasi terdapat pengaksesan untuk *mandor*. Halaman ini menampilkan suatu dokumentasi *progress* pengerjaan bangunan berdasarkan keterangan dan nama *mandor*. Pada halaman ini juga terdapat fitur *on camera*, nomor telepon dan *whatsapp gateway*.



Gambar 4.31 Halaman Menu Lihat Dokumentasi Mandor

g. Halaman Menu Data *Progress* Mandor

Pada menu tambah data progress terdapat pengaksesan untuk *mandor*. Pada halaman ini juga sebagai manajemen data *progress* rumah berdasarkan data gambar, deskripsi pengerjaan serta terdapat fitur simpan data.

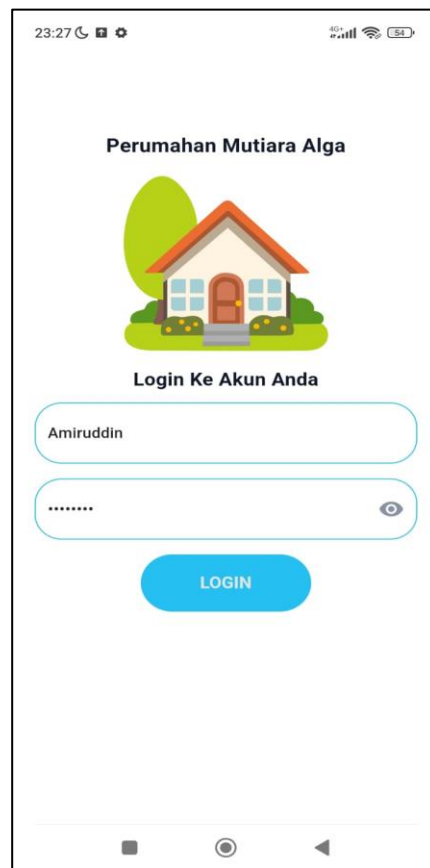


Gambar 4.32 Halaman Menu Data *Progress* Mandor

5. Klien (*User*)

a. Halaman *Login* Klien

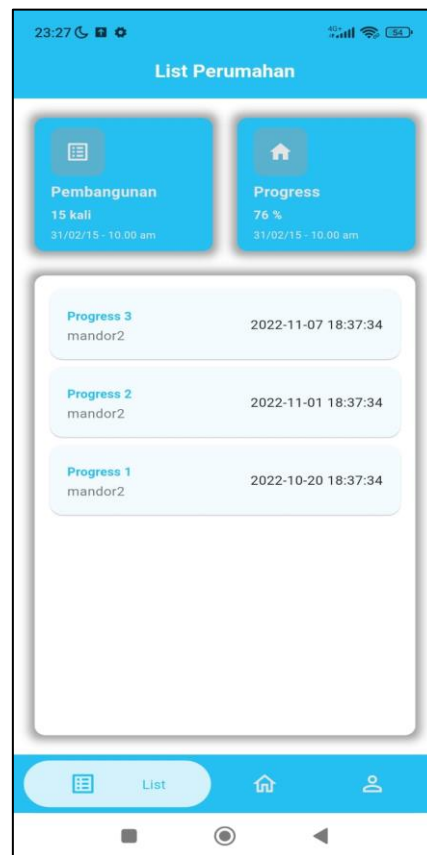
Halaman menu *login* merupakan halaman *login user*. *Login* merupakan tampilan awal yang ada pada sistem, yang digunakan *user* untuk masuk ke halaman selanjutnya.



Gambar 4.33 Halaman *Login* Klien

b. Halaman *Form* List Perumahan Klien

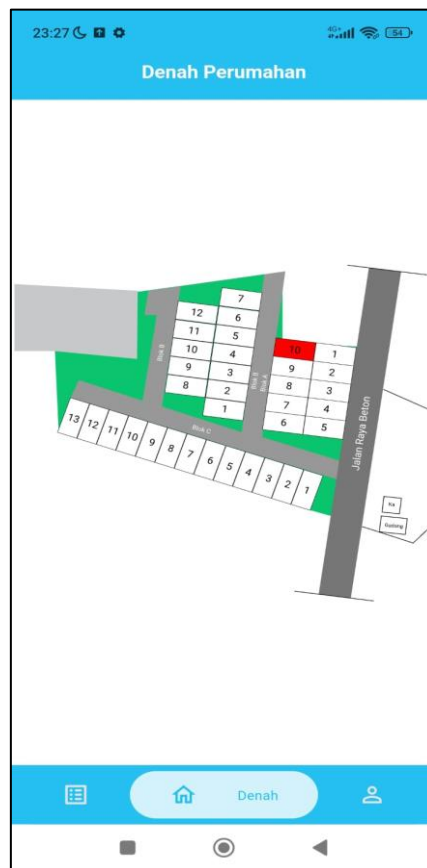
Halaman list perumahan merupakan halaman menu *klien*, halaman ini digunakan *klien* untuk menampilkan data dan informasi data perumahan. Halaman ini menampilkan berupa data dan informasi *Progress* Bangunan, Pembangunan dan Nama Mandor.



Gambar 4.34 Halaman *Form* List Perumahan Klien

c. Halaman *Form* Denah Klien

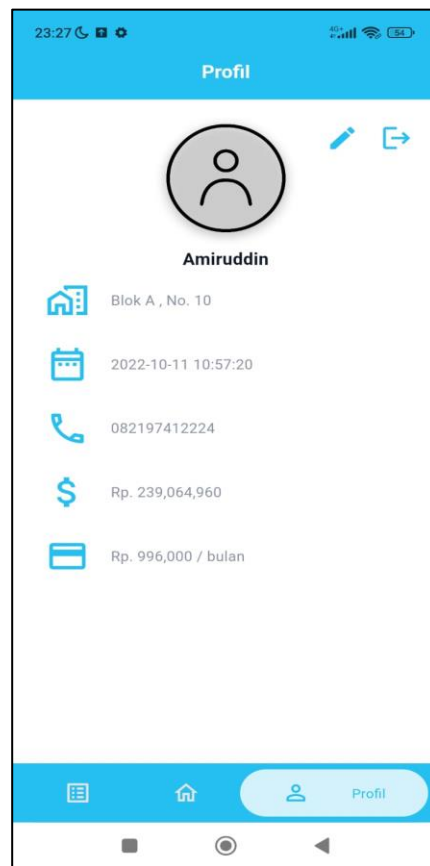
Halaman *form* denah merupakan halaman menu *klien*, halaman ini digunakan *klien* menampilkan gambaran denah perumahan. Halaman ini menampilkan berupa informasi jenis blok perumahan, nomor rumah dan nama pemilik perumahan dalam bentuk denah.



Gambar 4.35 Halaman *Form* Denah Klien

d. Halaman *Form* Profil Klien

Halaman *form* profil merupakan halaman menu *klien*. Halaman ini menampilkan informasi data akun pengguna yakni *klien*. Pada halaman ini juga sebagai manajemen data profil *klien* serta terdapat fitur simpan data profil pengguna dan *logout*.



Gambar 4.36 Halaman *Form* Profil Klien

e. Halaman Menu Lihat Informasi *Progress* Klien

Pada menu lihat informasi progress bangunan terdapat pengaksesan untuk *klien*. Halaman ini menampilkan suatu informasi *progress* bangunan berdasarkan dokumentasi pengerjaan rumah, deksripsi, nama mandor dan nomor telepon mandor serta terdapat fitur *whatsapp gateway*.



Gambar 4.37 Halaman Menu Lihat Informasi *Progress* Klien

D. Implementasi

Adapun kebutuhan yang digunakan untuk Membangun sebuah sistem dengan judul “Aplikasi e-Manajemen Kontruksi *Progress* Bangunan Perumahan Mutiara Alga Suppa Berbasis *Android*”.

1. Kebutuhan *Hardware*

Spesifikasi *hardware* pengguna aplikasi ini adalah sebagai berikut :

Tabel 4.1 Spesifikasi *Hardware* Komputer

Jenis	Spesifikasi
<i>Notebook/Komputer</i>	Asus
<i>Processor</i>	Intel(R) Core (TM) i3-7500 U CPU @ 4.0 Ghz
<i>Memory</i>	4 GB DDR 3
<i>Hardisk</i>	500 GB

Tabel 4.2 Spesifikasi *Smartphone* Mandor

Jenis	Spesifikasi
Jenis <i>Smartphone</i>	Vivo Y69
Layar	5 inch'
Ram	3 GB

Tabel 4.3 Spesifikasi *Smartphone* Klien

Jenis	Spesifikasi
Jenis <i>Smartphone</i>	<i>Xiaomi note 8 pro</i>
Layar	6 inch'
Ram	4 GB

2. Kebutuhan *Software*

Spesifikasi *software* pengguna aplikasi ini sebagai berikut :

Tabel 4.4 Spesifikasi *Software* Komputer

Jenis	Spesifikasi
Sistem Operasi	<i>Windows 10 Ultimate</i>

Tabel 4.5 Spesifikasi *Software Smartphone* Mandor

Jenis	Spesifikasi
Versi <i>Android</i>	5.0 Lolipop
Versi perangkat lunak	PD1705F_EX_A_1.18.0

Tabel 4.6 Spesifikasi *Software Smartphone* Klien

Jenis	Spesifikasi
Versi <i>Android</i>	<i>Android 9 Pie</i>
Versi <i>MIUI</i>	<i>MIUI 10 Global 9.6.27 Beta</i>

E. Metode pengujian

Pengujian sistem merupakan proses eksekusi suatu program atau sistem secara keseluruhan. Pengujian menunjukkan bahwa fungsi dari perangkat lunak bekerja sesuai dengan spesifikasi dan bahwa persyaratan kinerja telah dipenuhi. Metode pengujian yang digunakan ada *black-box*. Pengujian blackbox perangkat lunak dilakukan untuk memperlihatkan bahwa masing-masing fungsi telah beroperasi sepenuhnya dan secara bersamaan pula mencari kesalahan pada setiap fungsi.

F. Teknik Pengujian

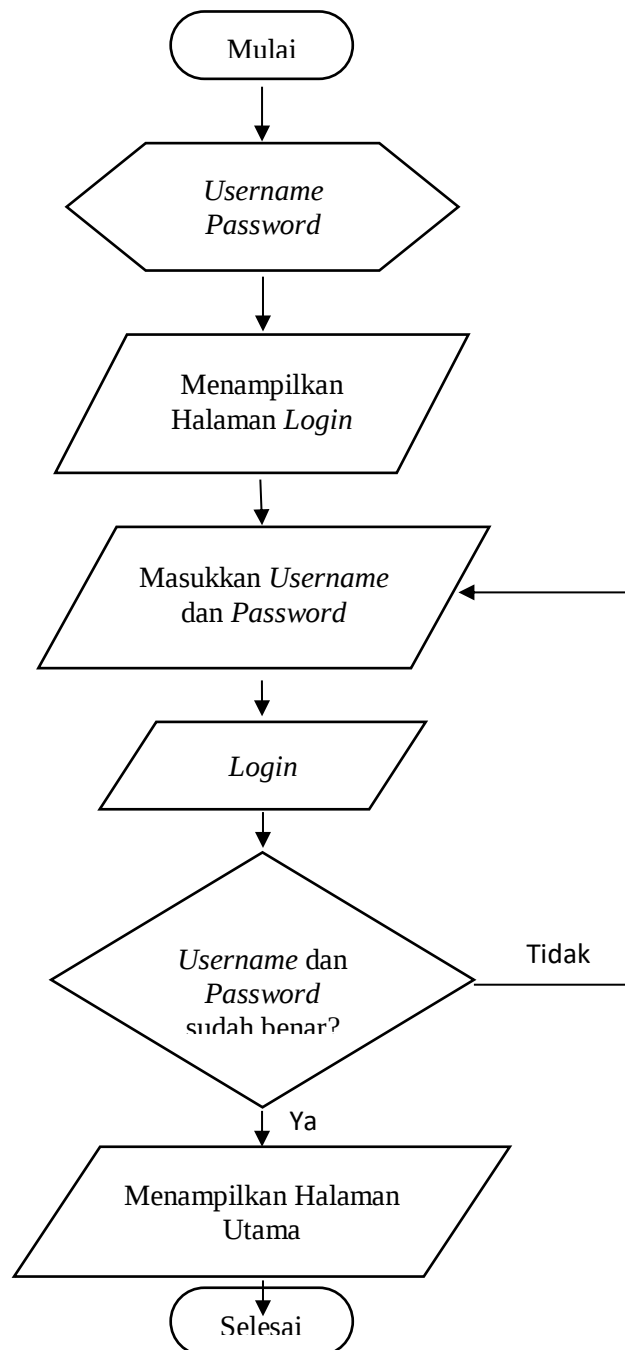
Pengujian sistem dilakukan untuk memeriksa suatu kekompatan antara komponen sistem yang akan diimplementasikan. Pengetesan sistem merupakan pengetesan program secara keseluruhan, adapun tahap yang dilakukan sebagai berikut :

1. Mekanisme pengujian :
 - a. Pengetesan program aplikasi dengan cara menjalankan aplikasi.
 - b. Pengetesan pemasukan data, perubahan data, dan penghapusan data.
 - c. Pengetesan terhadap fungsi tombol yang terdapat pada masing-masing halaman apakah berfungsi dengan baik.
2. Hasil yang diperoleh :

Aplikasi e-Manajemen Kontruksi *Progress* Bangunan Perumahan Mutiara Alga Suppa Berbasis *Android* sesuai usulan dapat berjalan dengan lancar sesuai dengan yang diharapkan.

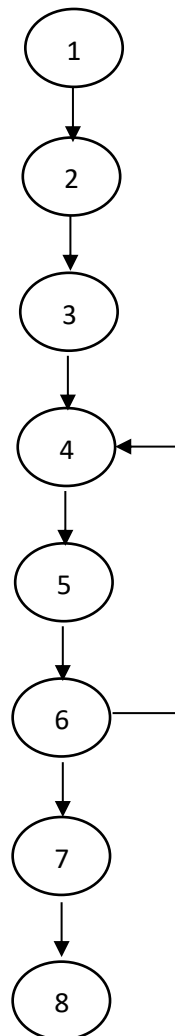
a. Pengujian *White Box*

1) *Flowchart dan Flowgraph Login Admin*



Gambar 4.7 *Flowchart Login Admin*

Dari *flowchart* yang digunakan untuk pengujian perangkat lunak, maka ditentukan *flowgraph* sebagai berikut:



Gambar 4.8 *Flowgraph Login Admin*

Dari *flowgraph menu login admin* diatas dapat dilakukan proses perhitungan sebagai berikut:

- 1) Menghitung *Cyclomatic Complexity* $V(G)$ dari *Edge* dan *Node*:

Dengan rumus : $V(G) = E - N + 2$

E (*edge*) = 8

N (*Node*) = 8

$$P(\text{Predikat Node}) = 1$$

Penyelesaian : $V(G) = E - N + 2$

$$= 8 - 8 + 2$$
$$= 2$$

$$\begin{aligned} \text{Predikat (P)} &= P + 1 \\ &= 1 + 1 \\ &= 2 \end{aligned}$$

2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *flowgraph* diatas memiliki *Region* = 2

3) *Independent path* pada *flowgraph* di atas adalah:

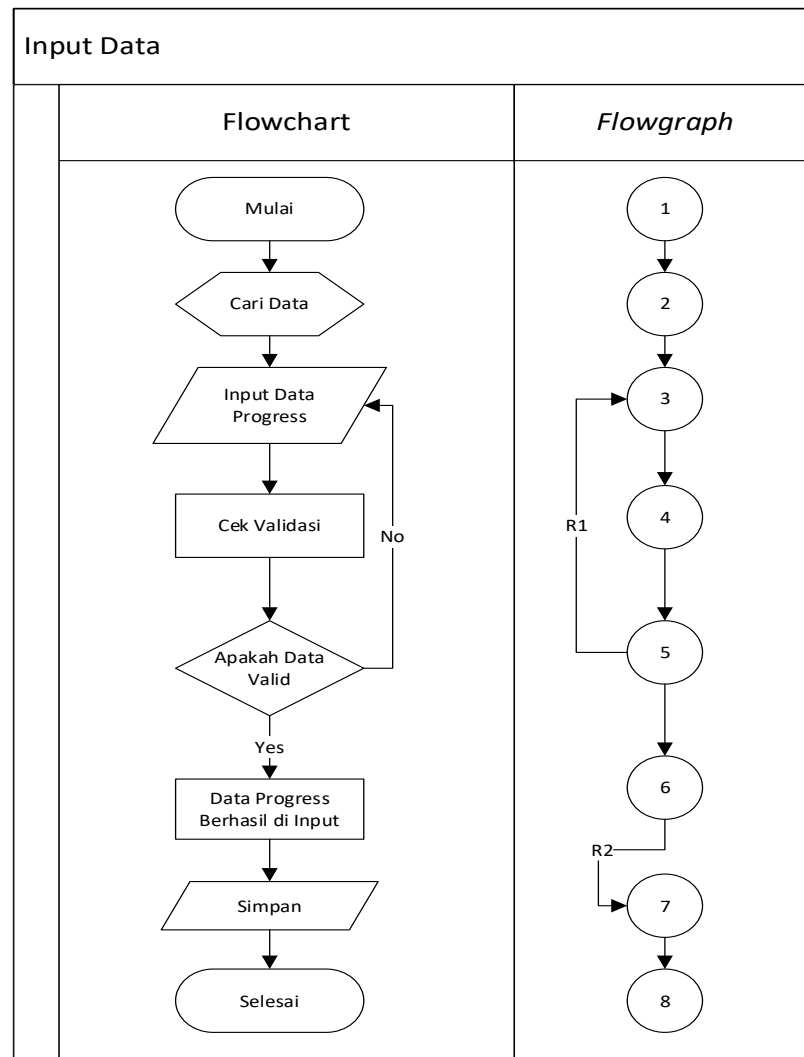
Path 1 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8

$$Path\ 2 = 1 - 2 - 3 - 4 - 5 - 6 - 4 - 5 - 6 - 7 - 8$$

4) Grafik Matriks *Login Admin*

Tabel 4.9 Grafik Matriks pada *Login Admin*

[illegible]

2) *Input Data*Tabel 4.10 Pengujian *Input Data*

Dari gambar *flowgraph Input Data* diatas dapat dilakukan proses perhitungan sebagai berikut :

a) Menghitung *Cyclomatic Complexity* $V(G)$ dari Egde dan *Node* :

Dengan Rumus : $V(G) = E - N + 2$

Dengan E (edge) = 7

N (node) = 7

Predikat *Node* (P) = 1

Penyelesaian : $V(G) = E - N + 2$
 $= 7 - 7 + 2 = 2$

Predikat (P) = P + 1

= 1 + 1 = 2

b) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* diatas memiliki *Region* = 2

c) Independent *Path* pada *flowgraph* diatas adalah :

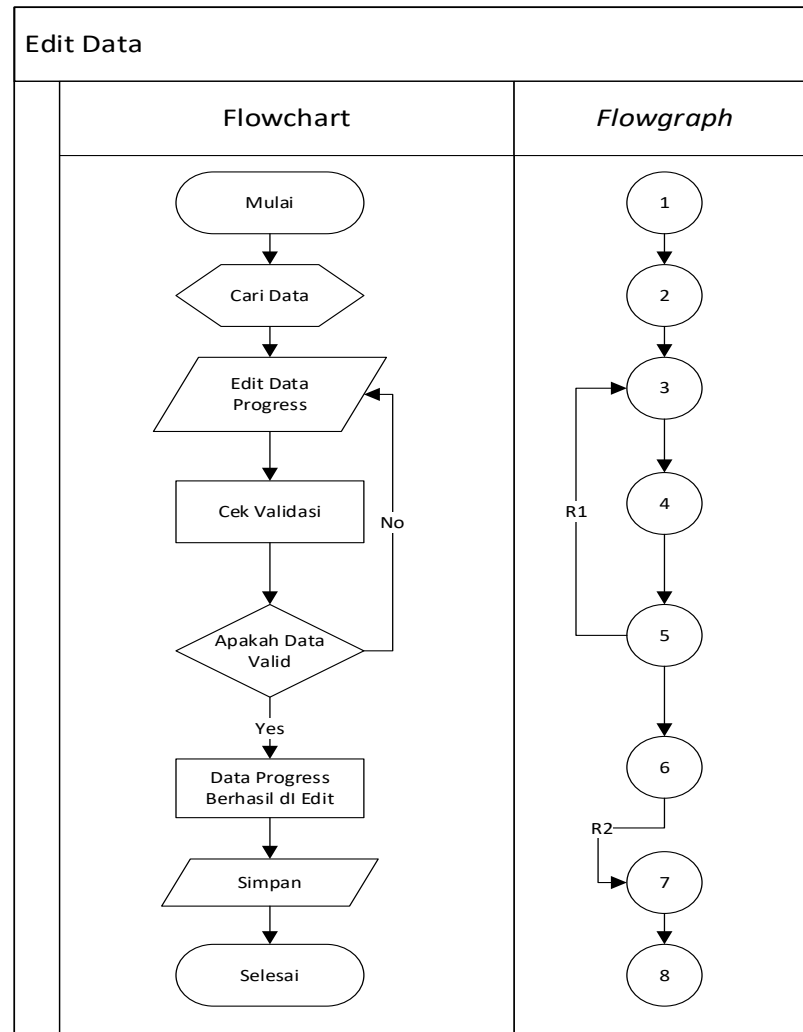
Path 1 = 1 – 2 – 3 – 4 – 2

Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 7

d) Grafik *Matriks*

Tabel 4.11 Grafik *Matriks* pada *Flowgraph* Input Data

[illegible]

3) *Edit Data*Tabel 4.12 Pengujian *Edit Data*

Dari gambar *flowgraph Edit Data* diatas dapat dilakukan proses perhitungan sebagai berikut:

a) Menghitung *Cyclomatic Complexity* $V(G)$ dari *Egde* dan *Node* :

Dengan Rumus : $V(G) = E - N + 2$

Dengan E (*edge*) = 8

N (*node*) = 8

Predikat *Node* (P) = 1

$$\begin{aligned}\text{Penyelesaian : } V(G) &= E - N + 2 \\ &= 8 - 8 + 2 = 2\end{aligned}$$

$$\begin{aligned} \text{Predikat (P)} &= P + 1 \\ &= 1 + 1 = 2 \end{aligned}$$

b) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* diatas memiliki *Region* = 2

c) Independent *Path* pada *flowgraph* diatas adalah :

Path 1 = 1 – 2 – 3 – 4 – 5 – 3

$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8$$

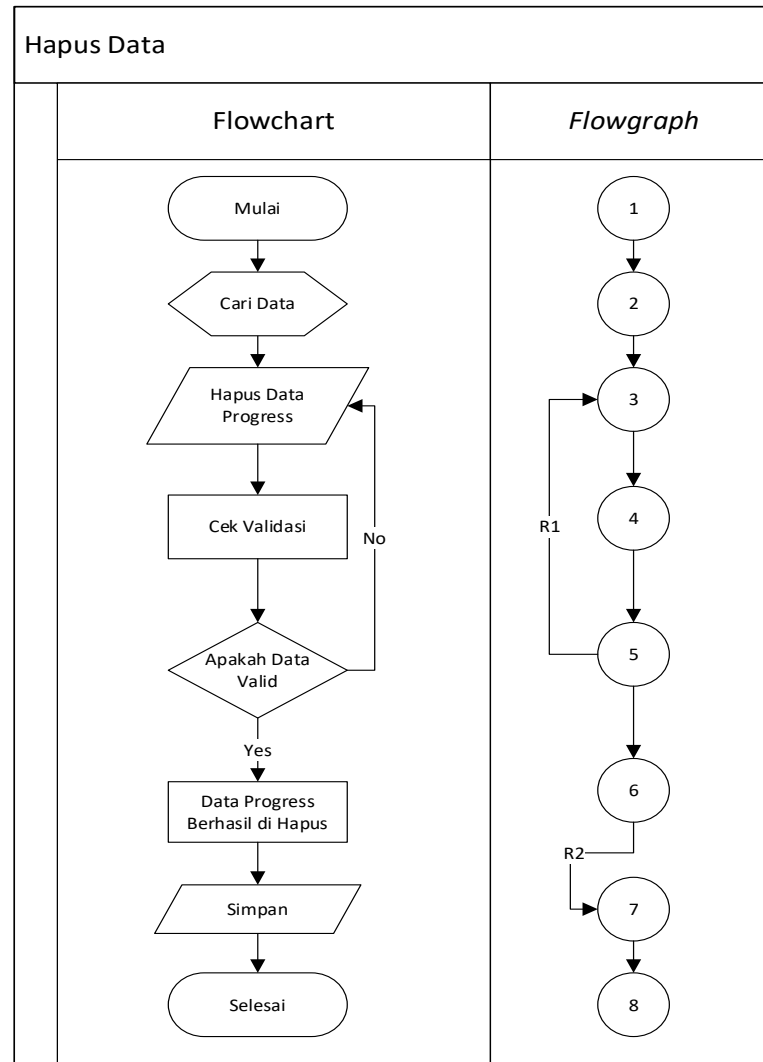
d) Grafik *Matriks*

Tabel 4.13 Grafik *Matriks* pada *Flowgraph Edit Data*

	1	2	3	4	5	6	7	8	$E - 1 = 0$
1		1							$1 - 1 = 0$
2			1						$1 - 1 = 0$
3				1					$1 - 1 = 0$
4					1				$1 - 1 = 0$
5			1			1			$2 - 1 = 1$
6							1		$1 - 1 = 0$
7								1	$1 - 1 = 0$
8									0
Sum (E) + 1									$1 + 1 = 2$

4) Hapus Data

Tabel 4.14 Pengujian Hapus Data



Dari gambar *flowgraph* Hapus Data diatas dapat dilakukan proses perhitungan sebagai berikut :

a) Menghitung *Cyclomatic Complexity* $V(G)$ dari Edge dan Node :

Dengan Rumus : $V(G) = E - N + 2$

Dengan E (edge) = 8

N (node) = 8

Predikat *Node* (P) = 1

$$\begin{aligned}\text{Penyelesaian : } V(G) &= E - N + 2 \\ &= 8 - 8 + 2 = 2\end{aligned}$$

Predikat (P) = P + 1
= 1 + 1 = 2

2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* diatas memiliki *Region* = 2

3) Independent *Path* pada *flowgraph* diatas adalah :

Path 1 = 1 – 2 – 3 – 4 – 5 – 3

$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8$$

4) Grafik Matriks

Tabel 4.15 Grafik *Matriks* pada *Flowgraph* Hapus Data

	1	2	3	4	5	6	7	8	$E - 1 = 0$
1		1							$1 - 1 = 0$
2			1						$1 - 1 = 0$
3				1					$1 - 1 = 0$
4					1				$1 - 1 = 0$
5			1			1			$2 - 1 = 1$
6							1		$1 - 1 = 0$
7								1	$1 - 1 = 0$
8									0
Sum (E) + 1									$1 + 1 = 2$

b. Pengujian *Black Box*

Pengujian sistem dengan pengujian *Blackbox*, pengujian ini bersifat menguji waktu perintah sesuai dengan fungsinya yang tanpa ada kesalahan atau *error*.

Tabel 4.16 Pengujian *BlackBox*

Rancangan Proses	Hasil yang Diharapkan	Hasil yang Sebenarnya	Hasil
Membuka Aplikasi Perumahan	Menampilkan halaman <i>login</i>	Menampilkan halaman <i>login</i>	✓
<i>Input username dan password</i>	Masuk ke halaman <i>Dashboard</i>	Masuk ke halaman Utama	✓
Klik tombol “simpan” dengan mengisi semua data	Berhasil menyimpan data yang sudah diinput	Berhasil menyimpan data yang sudah diinput	✓
Klik dan pilih data Pemilik Rumah	Klik tombol tambah data	Tampilkan <i>form</i> tambah data pemilik	✓
Klik dan pilih data <i>Progress</i> Bangunan	Klik tombol <i>list</i> pemilik rumah	Tampilkan <i>form</i> data <i>progress</i>	✓
Klik dan pilih data Dokumentasi	Klik tombol <i>list</i> pemilik	Tampilkan <i>form</i> Data Gambar	✓

Klik dan pilih Manajemen <i>User</i>	Klik tombol atau pilih Denah	Tampilkan <i>form</i> tambah data akun <i>user</i>	✓
--------------------------------------	------------------------------	--	---

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang telah diuraikan pada bab-bab sebelumnya, Dihasilkannya sebuah Aplikasi e-Manajemen Kontruksi Progress Bangunan Perumahan Mutiara Alga Suppa Berbasis *Android*. Aplikasi yang dibuat sudah bebas kesalahan karena diuji dengan menggunakan dua teknik pengujian *testing* yaitu pengujian *Black Box* dan pengujian *White Box*. Maka dapat ditarik kesimpulan sebagai berikut : Penelitian ini berhasil membangun suatu Aplikasi e-Manajemen konstruksi dalam memenuhi kebutuhan pengguna dalam memudahkan proses manajemen proyek dan memonitoring *progress* bangunan pada perumahan Mutiara Alga Suppa. Aplikasi ini dilengkapi dengan fitur-fitur monitoring *progress* bangunan yang memungkinkan pengguna untuk memantau perkembangan pekerjaan, status bangunan, serta penyimpanan dokumentasi *progress* bangunan. Dengan menggunakan aplikasi e-Manajemen konstruksi, pengguna dapat mempercepat proses manajemen proyek dan memonitoring progress bangunan dengan lebih efektif dan efisien. Selain itu, pengguna dapat meminimalkan kesalahan dan meningkatkan kualitas pelayanan pekerjaan di lapangan.

B. Saran-saran

Dalam Aplikasi e-Manajemen Kontruksi Progress Bangunan Perumahan Mutiara Alga Suppa Berbasis *Android*. Berdasarkan hasil penelitian maka untuk pengembangan aplikasi sebaiknya dibuat fitur untuk menncetak laporan berperiodik dan dapat diperbaharui setiap saat sesuai kondisi pogres pembangunan yang dilakukan. Juga menambahkan fitur untuk konsultasi langsung antara user dan pengembang perumahan ataupun pemilik perumahan.

DAFTAR PUSTAKA

- Ade Hastuty Hasyim. 2021. *Dasar Pemrograman* (Cetakan ke-2). Makassar: CV. Bangun Bumitama.
- Yudhanto, Yudha, and Ardhi Wijayanto. *Mudah membuat dan berbisnis aplikasi android dengan android studio*. Elex Media Komputindo, 2018.
- Enterprise, Jubilee. *HTML, PHP, dan MySQL untuk Pemula*. Elex Media Komputindo, 2018.
- Hadiprakoso, Raden Budiarto. *Rekayasa Perangkat Lunak*. Rbh, 2020.
- Nurzaman, Fahrul. "PENGEMBANGAN SISTEM OTOMATISASI TAGIHAN MENGGUNAKAN METODE AGILE S OFTWARE 1 DEVELOPMENT." *IKRA-ITH Informatika: Jurnal Komputer dan Informatika* 4.1 (2020): 46-57.
- Saputra, Y. (2021). SistemInformasi Penjurusan Siswa Berbasis Algoritma Naïve Bayes Pada MA Sunanulhuda.
- Harri Gunawan, (2020). "*Sistem Informasi Manajemen Perumahan Berbasis Web Pada PT Catur Graha Mandiri Palembang*".
- Huda, Baenil, and Bayu Priyatna. "Penggunaan Aplikasi Content Management System (CMS) Untuk Pengembangan Bisnis Berbasis E-commerce." *Systematics* 1.2 (2019): 81-88.
- Rahmawati, Nurul Alifah, and Arif Cahyo Bachtiar. "Analisis dan perancangan sistem informasi perpustakaan sekolah berdasarkan kebutuhan sistem." *Berkala Ilmu Perpustakaan dan Informasi* 14.1 (2018): 76-86.
- Suryadi, Andri, et al. "Pengembangan aplikasi sistem database Virtual Community Digital Learning Nusantara (VCDLN) menggunakan model waterfall dan pemrograman terstruktur." *Jurnal Petik* 8.1 (2022): 48-56.
- Luthfan Noor Irsyad (2021). "*Analisis Dan Perancangan Desain Sistem Informasi Manajemen Proyek (Studi Kasus: Dinas Perumahan Kota Pasuruan)*".
- Winanjar, J., & Susanti, D. (2021). Rancang Bangun Sistem Informasi Administrasi desa Berbasis Web Menggunakan PHP Dan MySQL. *PROSIDING SNAST*, 97-105.

- Julianti, M. R., Dzulhaq, M. I., & Subroto, A. (2019). Sistem Informasi Pendataan Alat Tulis Kantor Berbasis Web pada PT Astari Niagara Internasional. *Jurnal Sisfotek Global*, 9(2).
- Azis, Nur, Gali Pribadi, and Manda Savitrie Nurcahya. "Analisa dan Perancangan Aplikasi Pembelajaran Bahasa Inggris Dasar Berbasis Android." *IKRA-ITH INFORMATIKA: Jurnal Komputer Dan Informatika* 4.3 (2020): 1-5.
- Aripin, I. (2018). Konsep dan aplikasi mobile learning dalam pembelajaran biologi. *Jurnal Bio Educatio*, 3(1), 1-9.
- Sendiang, Maksy, and Mareyke Kondo. "Rekayasa Perangkat Lunak." (2018).
- Ardayan, Elia, et al. *Metode Penelitian Kualitatif dan Kuantitatif: Pendekatan Metode Kualitatif dan Kuantitatif di Berbagai Bidang*. PT. Sonpedia Publishing Indonesia, 2023.
- Universitas Muhammadiyah Parepare. (2021). *Panduan Akademik 2020-2021*, Parepare : UMPAR Press.
- Wita Dwijayanti, (2021). "Sistem Informasi Manajemen Proyek Pembangunan Perumahan Griya Aldena Karanganyar".
- Putra, Dede Wira Trise, and Rahmi Andriani. "Unified modelling language (uml) dalam perancangan sistem informasi permohonan pembayaran restitusi sppd." *Jurnal Teknoif Teknik Informatika Institut Teknologi Padang* 7.1 (2019): 32-39.