# BAB I

# PENDAHULUAN

## A. Latar Belakang

Manusia Pada umumnya mempunyai kebutuhan akan tempat tinggal yakni rumah. Disamping sebagai tempat untuk berlindung, rumah tangga juga sebagai tempat berkumpul dan berkomunikasinya anggota keluarga. Sebagai salah satu bank yang memperhatikan kebutuhan masyarakat, bank ini juga menyediakan layanan kredit kepemilikan rumah yang ditawarkan kepada para nasabahnya. Kredit tersebut juga bisa digunakan sebagai alat membangun bisnis dan memenuhi kebutuhan hidup lainnya. Pemberian kredit tidak dapat dilakukan secara sembarangan, demikian juga dengan pemberian kredit dibank. Sebelum memberikan dana melalui pembiayaan pada nasabah, pihak bank terlebih dahulu melakukan penilaian nasabah untuk mengetahui layak atau tidak nasabah tersebut menerima pembiayaan.

Untuk itu nasabah harus memenuhi syarat 5C, bagaimana karakter dari nasabah (*character*), kapasitas melunasi kredit tersebut (*capacity*), kemampuan modal yang dimiliki oleh nasabah (*capital*), jaminan atau asset yang dimiliki oleh nasabah untuk menanggung resiko kredit (*collateral*), dan kondisi ekonomi saat ini yang mempengaruhi usaha nasabah (*condition of economic*). Dalam keputusan layak atau tidaknya kepada calon debitur dibank masih menggunakan proses manual dan database yang digunakan masih dalam bentuk kertas sehingga menyulitkan dalam penyimpanan atau mencari data.

Dalam penelitian ini dibuatlah sebuah sistem informasi berbasis komputer atau biasa dikenal dengan sebutan Sistem Pendukung Keputusan sehingga dapat mempermudah dan mempercepat proses penentuan kredit. Penulis membuat aplikasi ini agar aplikasi ini dapat memudahkan seorang kredit analisis dalam menentukan layak atau tidaknya kepada calon nasabah yang lebih efesien. AHP adalah prosedur yang berbasis matematis yang bagus dan sesuai untuk kondisi evaluasi atribut-atribut kualitatif. Atribut tersebut kemudian secara matematik dikuantitatifkan ke dalam satu set matrik perbandingan berpasangan.

berdasarkan uraian diatas maka penulis ingin mengusulkan judul **"Sistem Penunjang Keputusan Kredit Rumah Menggunakan Metode *Analiytic Hierarchy Process*".**

## B. Rumusan Masalah

Rumusan masalah pada penelitian ini adalah bagaimana menentukan proses penilaian kelayakan pemberian kredit rumah melalui sistem penunjang keputusan, yang mempertimbangkan berbagai faktor yang menjadi acuan dalam pengambilan keputusan oleh bagian kredit.

## C. Batasan Masalah

Penulis hanya membatasi pada penelitian pada hal-hal berikut :

1. Sistem hanya melibatkan calon nasabah sebagai sumber data, Analisa kredit sebagai penilaian dan direksi sebagai pemberi keputusan.
2. Proses perhitungan dibatasi hanya sampai penilaian kriteria pemohon kredit yang dilakukan setelah pemerikasaan data pemohon.

3. Metode yang digunakan dalam pemberian kredit ini menggunakan metode *Analitycal Hierarchy Process* (AHP).

## D. Tujuan Penelitian

Adapun tujuan penelitian ini adalah :

1. Untuk membangun aplikasi yang membantu memudahkan mengambil keputusan dan menangani proses seleksi nasabah.
2. Untuk membangun aplikasi sistem penunjang keputusan yang membantu menentukan kelayakan nasabah mendapatkan kredit.

## E. Manfaat Penelitian

Adapun manfaat yang diperoleh dari tugas akhir ini adalah sebagai berikut:

1. Aplikasi ini dapat digunakan untuk membantu dan mempercepat proses Analisa kredit dalam pemilihan debitur yang layak diberikan kredit.
2. Dapat mengelolah data nasabah yang mengajukan pinjaman dengan terstruktur dan rapi.
3. Bagi penulis dapat menambah wawasan tentang system penunjang keputusan khususnya pengetahuan tentang metode *Analytic Hirarchy Process*.

# BAB II

# TINJAUAN PUSTAKA

## A. Kajian Pusataka

### 1. Sistem Penunjang Keputusan (SPK)

Sistem penunjang keputusan merupakan sistem informasi pada level manajemen dari suatu organisasi yang mengkombinasikan data dan model analisis canggih atau peralatan data analisis untuk mendukung pengambilan keputusan yang semi terstruktur dan tidak terstruktur.(Hanif Al Fatta,2007)

a Tahapan Pengambilan Keputusan

Saat melakukan pemodelan dalam pembangunan DSS dilakukan Langkah-langkah sebagai berikut :

1) Studi Kelayakan (*Intelegence*)

   Pada Langkah ini, sasaran ditentukan dan dilakukan pencarian prosedur, pengumpulan data, identifikasi masalah, identifikasi kepemilikan masalah, klarifikasi masalah, hingga terbentuk sebuah pernyataan masalah.

2) Perancangan (*Design*)

   Pada tahapan ini akan diformulasikan model yang akan digunakan dan kriteria-kriteria yang ditentukan. Setelah itu, dicari *alternative* model yang bisa menyelesaikan permasalahan tersebut. Langkah selanjutnya adalah memprediksi keluaran yang mungkin. Kemudian ditentukan variabel-variabel model.

3) Pemilihan (*Choice*)

   Setelah pada tahap perancangan ditentukan berbagai alternatif model beserta variabel-variabelnya. Pada tahapan ini akan dilakukan pemilihan model.

4) Membuat DSS

   Setelah menentukan modelnya, berikutnya adalah mengimplementasikannya dalam aplikasi DSS.

Definisi awal sistem penunjang keputusan(*Decision Support System*/DSS) adalah suatu system yang ditujukan untuk mendukung manajemen pengambilan keputusan. System berbasis model yang terdiri dari prosedur-prosedur dalam pemrosesan data dan pertimbangannya untuk membantu manajer dalam mengambil keputusan. DSS merupakan system informasi interaktif yang menyediakan informasi, pemodelan, dan pemanipulasian data.sistem itu digunakan untuk pengambilan keputusan dalam situasi yang semiterstruktur dan situasi yang tidak terstruktur,dimana tak seorang pun yang tahu secara pasti bagaimana keputusan seharusnya dibuat. Sistem pendukung keputusan merupakan suatu pendekatan untuk mendukung pengambilan keputusan. System pendukung keputusan menggunakan data, memberikan antar muka pengguna yang mudah dan dapat menggabungkan pemikiran pengambil keputusan.(Turban,2005).

## 2. Kredit

Kredit berasal dari kata Yunani Credere yang berarti kepercayaan oleh karena itu dasar dari kredit adalah kepercayaan. Menurut pasal 1 angka 11 undang- undang perbankan yang dimaksud kredit adalah penyediaan uang atau tagihan yang dapat

dipersamakan dengan itu, berdasarkan persetujuan atau kesepakatan pinjam meminjam antara bank dengan pihak lain yang mewajibkan pihak peminjam untuk melunasi hutangnya setelah jangka waktu tertentu setelah memberi bunga.

Adapun unsur-unsur kredit adalah :

a Adanya kedua belah pihak yaitu pemberi kredit dan yang menerima kredit

b Adanya persetujuan antara kedua belah pihak

c Adanya kepercayaan antara pemberi kredit kepada penerima kredit

d Adanya unsur waktu atau jangka waktu

e Adanya unsur resiko baik itu penerima kredit dan pemberi kredit(Yohanes Suhari, Muji sukur, Sri eniyati, 2009).

Dalam pemberian kredit harus memperhatikan tingkat kesehatannya. Nasabah yang memenuhi syarat kredit adalah nasabah yang potensial untuk diajak bekerja sama dan layak mendapatkan kredit. Syarat-syarat yang harus dipenuhi pihak calon nasabah untuk pengajuan kredit rumah adalah :

a Warga Negara Indonesia

b Memiliki E-KTP

c Tidak memiliki rumah

d Belum pernah memperoleh subsidi rumah dari pemerintah

e Memiliki NPWP

f Memiliki surat nikah/cerai g. Memiliki SPT tahunan

g Dan memiliki penghasilan dibawah 4jt

### 3. *Analytical Hierarchy Process* (AHP)

Melalui perbandingan berpasangan dapat diterapkan teori pengukuran dan penilaian denagn menggunakan skala penilaian yang sudah ditentukan dimana penilaian melibatkan elemen dan atribut yang ada, merupakan defines *Analytic Hierarchy Process*. Beberapa kelebihan *Analytical Hierarchy Process* diantaranya penyajian secara grafis sehingga memudahkan untuk dibaca dan dipahami, kemudain pada *Analytical Hierarchy Process* memiliki kelebihan dalam proses perbandingan berpasangan antar kriteria yang ada dan dalam penentuaj konsistensi dari kriteria yang digunakan. *Analytical Hierarchy Process* merupakan sebuah pandangan manusia yang dibentuk dalam hirarki untuk menggambarkan suatu keputusan atas permasalahan yang kompleks.

Metode AHP mampu menguraikan masalah multifactor terpecah menjadi sub-sub masalah dan digambarkan salam hirarki. AHP memiliki beberapa prinsip dasar yang harus dipahami untuk menyelesaikan masalah, antara lain :

a Menyusun *hierarki* dari permasalahan yang dihadapi

Persoalan yang rumit bisa dimengerti dengan diuraikan menjadi elemen-elemen pendukung, menyusun elemen secara hierarki dan menggabungkannya.

b Penilaian kriteria dan alternative

Kriteria dan alternative dapat dilakukan dengan cara membuat matriks perbandingan berpasangan. Menurut saaty, nilai dan defenisi pendapat kualitatif dari skala perbandingan bisa diukur dengan menggunakan table analisis seperti berikut :

**Tabel 2. 1** *Penilaian Kriteria dan Alternative*

| Intensitas Kepentingan | Keterangan |
|---|---|
| 1 | Kedua elemen sama pentingnya |
| 3 | Elemen yang satu sedikit lebih penting dari pada elemen lainnya |
| 5 | Elemen yang satu sedikit lebih penting dari pada elemen lainnya |
| 7 | Satu elemen yang jelas mutlak penting dari pada elemen lainnya |
| 9 | Satu elemen yang jelas mutlak penting dari pada elemen lainnya |
| 2,4,6,8 | Nilai-nalai antara dua nilai pertimbangan yang berdekatan |

c Menentukan prioritas

Setiap kriteria dan alternative perlu dilakukan perbandingan berpasangan. Nilai perbandingan dari kriteria dan alternative dilakukan sendiri untuk menghasilkan bobot dan prioritas dihitung dengan cara manipulasi matriks.(Resti Lia Andharsaputri, 2020).

### 4. PHP

PHP (PHP Hypertext Preprocessor) telah menjadi salah satu bahasa pemrograman yang sangat dominan dalam pengembangan website dinamis. Dengan kemampuan berjalan di sisi server, PHP memfasilitasi eksekusi kode di server sebelum hasilnya disampaikan kepada browser pengguna. Inilah yang membuatnya disebut sebagai bahasa Server Side Scripting yang vital dalam pengembangan web modern.

Salah satu fitur utama dari PHP adalah kemampuannya untuk menciptakan halaman web yang dinamis. Artinya, konten yang ditampilkan tidak statis, melainkan dapat berubah sesuai dengan berbagai faktor seperti input pengguna, data dari database, atau kondisi lainnya. Hal ini memberikan fleksibilitas yang besar dalam membangun aplikasi web yang responsif dan interaktif.

Selain dinamis, PHP juga memiliki integrasi yang kuat dengan HTML, yang memungkinkan pengembang untuk menulis kode PHP langsung di dalam markup HTML. Ini membuat proses pengembangan web menjadi lebih mudah dan efisien. Selain itu, PHP juga mendukung integrasi dengan berbagai jenis database, termasuk MySQL, PostgreSQL, dan MongoDB. Ini memungkinkan aplikasi web untuk menyimpan, mengambil, dan memanipulasi data dengan mudah.

Awalnya, PHP diciptakan oleh Rasmus Lerdorf pada tahun 1995 dengan nama FI (Form Interpreted), dimaksudkan untuk mengelola formulir web. Namun, seiring dengan perkembangan waktu, PHP berkembang menjadi sebuah bahasa pemrograman yang kuat dan beragam, mampu menangani berbagai macam tugas dalam pengembangan web.

Pada tahun 1997, PHP dirilis dalam versi 2.0, yang menandai integrasi PHP dengan bahasa pemrograman C dan dilengkapi dengan berbagai modul baru. Ini menghasilkan peningkatan kualitas dan kemampuan kerja yang signifikan bagi PHP, menjadikannya salah satu bahasa pemrograman yang paling populer dan banyak digunakan hingga saat ini.(Saputra, 2011)

### 5. Sublime Text

Sublime text adalah aplikasi editor untuk kode dan teks yang dapat berjalan diberbagai platform operating system dengan menggunakan teknologi Phyton API. Terciptanya aplikasi ini terinspirasi dari aplikasi vim , aplikasi ini sangatlah fleksibel dan powerfull. Fungsional dari aplikasi ini dapat dikembangkan dengan menggunakan sublime-packages. Sublime text bukanlah aplikasi open source dan juga aplikasi yang dapat digunakan dan didapatkan secara gratis, akan tetapi beberapa fitur pengembangan fungsionalitas(packages) dari aplikasi ini merupakan hasil dari temuan dan mendapat dukungan penuh dari komunitas serta memiliki lisensi aplikasi gratis. Sublime text mendukung berbagai Bahasa pemrograman yang didukung ataupun dikembangkan oleh komunitas seperti ; C, C++, C#, CSS, D, Dylan, Erlang, HTML, Groovy, Haskell, Java, Javascript, LaTeX, Lisp, Lua, Markdown, MATLAB, OCaml, Perl, PHP, Phyton, R, Ruby, SQL, TCL, Textile, and XML. Biasanya bagi Bahasa pemrograman yang didukung ataupun belum terdukung secara default dapat lebih dimaksimalkan atau didukung dengan menggunakan add-ons yang bisa didownload sesuai kebutuhan server.(Hasan Tarmizi, 2017).

### 6. Website

Website adalah kumpulan dari halaman-halaman web yang berhubungan dengan file-file lain yang terkait. Dalam sebuah website terdapat suatu halaman yang dikenal dengan sebutan home page. Home page adalah sebuah halaman yang pertama kali dilihat ketika sesorang mengunjungi website. Dari home page, pengunjung dapat mengklik hyperlink untuk pindah ke halaman yang lain yang terdapat dalam website tersebut. Website merupakan komponen atau kumpulan komponen yang terdiri dari teks, gambar, suara animasi sehingga lebih merupakan media informasi yang menarik untuk dikunjungi.

Sebuah situs web(sering pula disingkat menjadi situs saja; web site, site)Adalah sebutan bagi sekelompok halaman web (web page), yang umumnya merupakan bagian dari suatu nama domain (domain name) atau subdomain di World Wide Web (WWW) di internet. WWW terdiri dari seluruh situs web tersedia kepada public. Halaman-halaman sebuah situs web diakses dari sebuah URL yang menjadi “akar” (root), yang disebut homepage (halaman induk sering diterjemahkan menjadi “beranda”, “halaman muka”), dan biasanya disimpan dalam server yang sama. Tidak semua situs web dapat diakses dengan gratis. Beberapa situs web memerlukan pembayaran agar dapat menjadi pelanggan, misalnya situs-situs yang menampilkan pornografi, situs-situs berita, layanan surat elektronik (e- mail), dan lain-lain.

Secara termilogi, website adalah kumpulan dari halaman-halaman situs yang biasanya terangkum dalam sebuah domain atau subdomain, yang tempatnya berada didalam World Wide Web(WWW) di internet. Sebuah halaman adalah dokumen

yang ditulis dalam format HTML (Hyper Text Markup Language), yang hamper selalu bisa diakses melalui HTTP, yaitu protocol yang menyampaikan informasi dari server website untuk ditampilkan kepada para pemakai melalui web browser. Semua publikasi dari website-website tersebut dapat membentuk sebuah jaringan informasi yang sangat besar.

### 7. Basis Data (Database)

Basis data terdiri dari dua kata yakni basis dan data. Basis kurang lebih dapat diartikan pula sebagai Gudang atau markas, tempat bersarang atau berkumpul. Sedangkan data merupakan representasi fakta dunia nyata yang mewakili suatu objek seperti manusia(siswa, pegawai, dan pelanggan), hewan, barang, peristiwa, keadaan, konsep, dan sebagainya yang terekam dalam bentuk huruf, symbol, teks, angka, gambar, bunyi, atau kombinasi lainnya. Database atau basis data adalah kumpulan data yang disimpan secara sistematis didalam computer yang dapat diolah atau dimanipulasi menggunakan perangkat lunak(program aplikasi) untuk menghasilkan informasi.

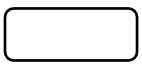
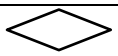
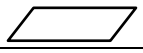
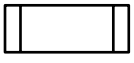
Pendenifisian basis data meliputi spesifikasi berupa tipe data, struktur data dan juga Batasan-batasan data yang akan disimpan. Basis data merupakan aspek yang sangat penting dalam system informasi dimana basis data merupakan Gudang penyimpanan data yang akan diolah lebih lanjut. Basis data menjadi penting karena dapat mengorganisasi data, menghindari duplikasi data, hubungan antara data yang tidak jelas dan juga update yang rumit.(Nugroho, 2012).

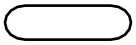
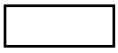
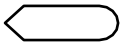
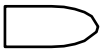
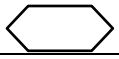
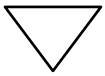
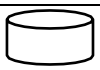
### a. Flowchart

Flowchart merupakan penggambaran secara fisik dari langkah-langkah dan urutan prosedur program yang biasanya mempermudah penyelesaian masalah. Flowchart atau diagram alir merupakan sebuah diagram dengan simbol-simbol garis yang menyatakan aliran algoritma atau proses yang menampilkan Langkah-langkah yang disimbolkan dalam bentuk kotak, beserta urutannya dengan menghubungkan masing-masing Langkah tersebut menggunakan tanda panah.

Flowchart adalah cara untuk menjelaskan tahap-tahap pemecahan masalah dengan mempersentasikan simbol-simbol tertentu yang mudah dipahami, mudah digunakan dan standar. Tujuan penggunaan flowchart adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai dan rapi dengan menggunakan symbol-simbol yang standar yang dapat dimengerti oleh programmer. Tahapan penyelesaian masalah disajikan harus tepat, sederhana, dan jelas.(Nugroho, 2012).

**Tabel 2. 2** *Simbol Diagram Flowchart*

| No. | Simbol | Nama Simbol | Keterangan |
|---|---|---|---|
| 1. |  | *Alternate Process* | Menyatakan segala jenis operasi yang diproses dengan menggunakan mesin yang memiliki keyboard. |
| 2. |  | *Decision* | Suatu penyelesaian kondisi dalam program. |
| 3. |  | *Data* | Mewakilik data *input* atau *output*. |
| 4. |  | *Predefined Process* | Suatu operasi yang rinciannya di tunjukkan di tempat lain |

| No. | Simbol | Nama Simbol | Keterangan |
|---|---|---|---|
| 5. | | *Document* | Document *input* dan *output* baik untuk proses manual, mekanik atau computer |
| 6. |  | *Terminator* | Untuk menunjukkan awal dan akhir dari suatu proses |
| 7. |  | *Process* | Kegiatan proses dari operasi program computer |
| 8. | | *Manual Input* | *Input* yang menggunakan *online keyboard* |
| 9. | | *Conector* | Penghubung ke halaman yang masih sama |
| 10. | | *Off -Page Connector* | Penghubung ke halaman lain |
| 11. |  | *Display* | *Output* yang ditampilkan di monitor |
| 12. |  | *Delay* | Menunjukkan penundaan |
| 13. |  | *Preparation* | Memberi nilai awal suatu besaran |
| 14. | | *Manual Operation* | Pekerjaan manual |
| 15. | | *Card* | *Input* atau *output* yang menggunakan kartu |
| 16. | | *Punch Tape* | *Input* atau *output* menggunakan pita kertas berlubang |
| 17. |  | *Merge* | Penggabungan atau penyimpanan beberapa proses atau informasi sebagai salah satu |
| 18. | | *Dirrect Access Storage* | *Input* atau *output* menggunakan drum magnetic |
| 19. |  | *Magnetic Disk* | *Input* atau *output* menggunakan *hard disk* |

| No. | Simbol | Nama Simbol | Keterangan |
|---|---|---|---|
| 20. | | *Sequential Access Storage* | *Input* atau *output* menggunakan pita magnetik. |
| 21. | | *Sort* | Proses pengurutan data di luar computer |
| 22. | | *Stored Data* | *Input* atau *output* menggunakan *diskette* |
| 23. | | *Extract* | Proses dalam jalur parallel. |
| 24. |  | *Arrow* | Menyatakan jalan atau arus suatu proses |
| 25. | | *Summing Junction* | Untuk berkumpul beberapa cabang sebagai proses tunggal |
| 26. | | *Or* | Proses menyimpang dalam dua proses |

### 8. UML (Unified Modelling Language)

*Unified Modeling Language* (UML) adalah bahasa pemodelan visual yang digunakan untuk menspeksifikasikan, memvisualisasiaka, membangun, dan mendokumentasikan rancanga dari suatu sistem perangkat lunak. Pemodelan memberikan gambaran yang jelas mengenai sistem yang akan dibangun baik dari sisi struktural ataupun fungsional. UML dapat diterapkan pada semua model pengembangan, tingkatan siklus sistem, dan berbagai macam domain aplikasi. Dalam UML terdapat konsep semantik, notasi dan panduan masing-masing

diagram. UML juga memiliki bagian statis, dinamis, ruang lingkup, dan organisasional. UML bertujuan menyatukan Teknik-teknik pemodelan berorientasi objek menjadi terstandarisasi.

UML diciptakan oleh *Object Management Group* yang diawali dengan versi 1.0 pada Januari 1997. Dalam pengembangan berorientasi objek ada beberapa prinsip yansg harus dikenal : *Object, Class, Abstraction, Encapsulation, Inheritance* dan *Polymorphism*.

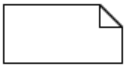
Dalam UML sendiri terdapat beberapa diagram yaitu :

a *Use case* diagram

*Use case* diagram adalah satu dari berbagai jenis diagram UML (*Unified Modelling Language*) yang menggambarkan hubungan interaksi antara sistem dan aktor. *Use Case* dapat mendeskripsikan tipe interaksi antara si pengguna sistem dengan sistemnya. Tentunya, *use case* diagram merupakan sesuatu yang mudah dipelajari. Langkah awal untuk melakukan pemodelan, tentu perlunya suatu diagram yang mampu menjabarkan aksi aktor dengan aksi sistem itu sendiri, seperti Berguna memperlihatkan proses aktivitas secara urut dalam sistem.

**Tabel 2. 3** *Simbol – Simbol Use Case Diagram*

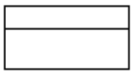
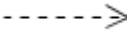
| NO | GAMBAR | NAMA | KETERANGAN |
|---|---|---|---|
| 1 | | *Actor* | Menspesifikasikan himpuan peran yang pengguna mainkan ketika berinteraksi dengan *use case*. |

| NO | GAMBAR | NAMA | KETERANGAN |
|---|---|---|---|
| 2 |  | *Dependency* | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri |
| 3 |  | *Generalization* | Hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*). |
| 4 |  | *Include* | Menspesifikasikan bahwa *use case* sumber secara *eksplisit*. |
| 5 |  | *Extend* | Menspesifikasikan bahwa *use case* target memperluas perilaku dari *use case* sumber pada suatu titik yang diberikan. |
| 6 |  | *Association* | Apa yang menghubungkan antara objek satu dengan objek lainnya. |
| 7 | | *System* | Menspesifikasikan 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 dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi). |
| 10 |  | *Note* | Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi. |

b Class Diagram

Adapun simbol-simbol *Class* Diagram antara lain :

**Tabel 2. 4** *Simbol Class Diagram*

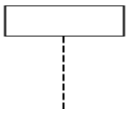
| NO | GAMBAR | NAMA | KETERANGAN |
|---|---|---|---|
| 1 |  | *Generalization* | Hubungan dimana objek anak *(descendent)* berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*). |
| 2 | | *Nary Association* | Upaya untuk menghindari asosiasi dengan lebih dari 2 objek. |
| 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 dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri. |

| NO | GAMBAR | NAMA | KETERANGAN |
|---|---|---|---|
| 7 | ——— | *Association* | Apa yang menghubungkan antara objek satu dengan objek lainnya. |

c Sequence Diagram

Adapun simbol-simbol *Sequence* Diagram antara lain :

**Tabel 2. 5** *Simbol Sequence Diagram*

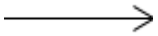
| NO | GAMBAR | NAMA | KETERANGAN |
|---|---|---|---|
| 1 |  | *LifeLine* | Objek *entity*, antarmuka yang saling berinteraksi. |
| 2 | | *Message* | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi. |
| 3 | S | *Message* | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi |

d StateChart Diagram

Adapun simbol-simbol StateChart Diagram antara lain :

**Tabel 2. 6** *Simbol StateChart 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* | Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi. |

e Activity Diagram

Adapun simbol-simbol Activity Diagram antara lain :

**Tabel 2. 7** *Simbol Activity Diagram*

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

| | | | |
|---|---|---|---|
| 1 | | *Actifity* | Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain |
| 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 |

### 9. *Blackbox Testing*

*Blackbox testing* adalah tahap yang digunakan untuk menguji kelancaran program yang telah dibuat. Pengujian ini penting dilakukan agar tidak terjadi kesalahan alur program yang telah dibuat (Edora, S., 2019)

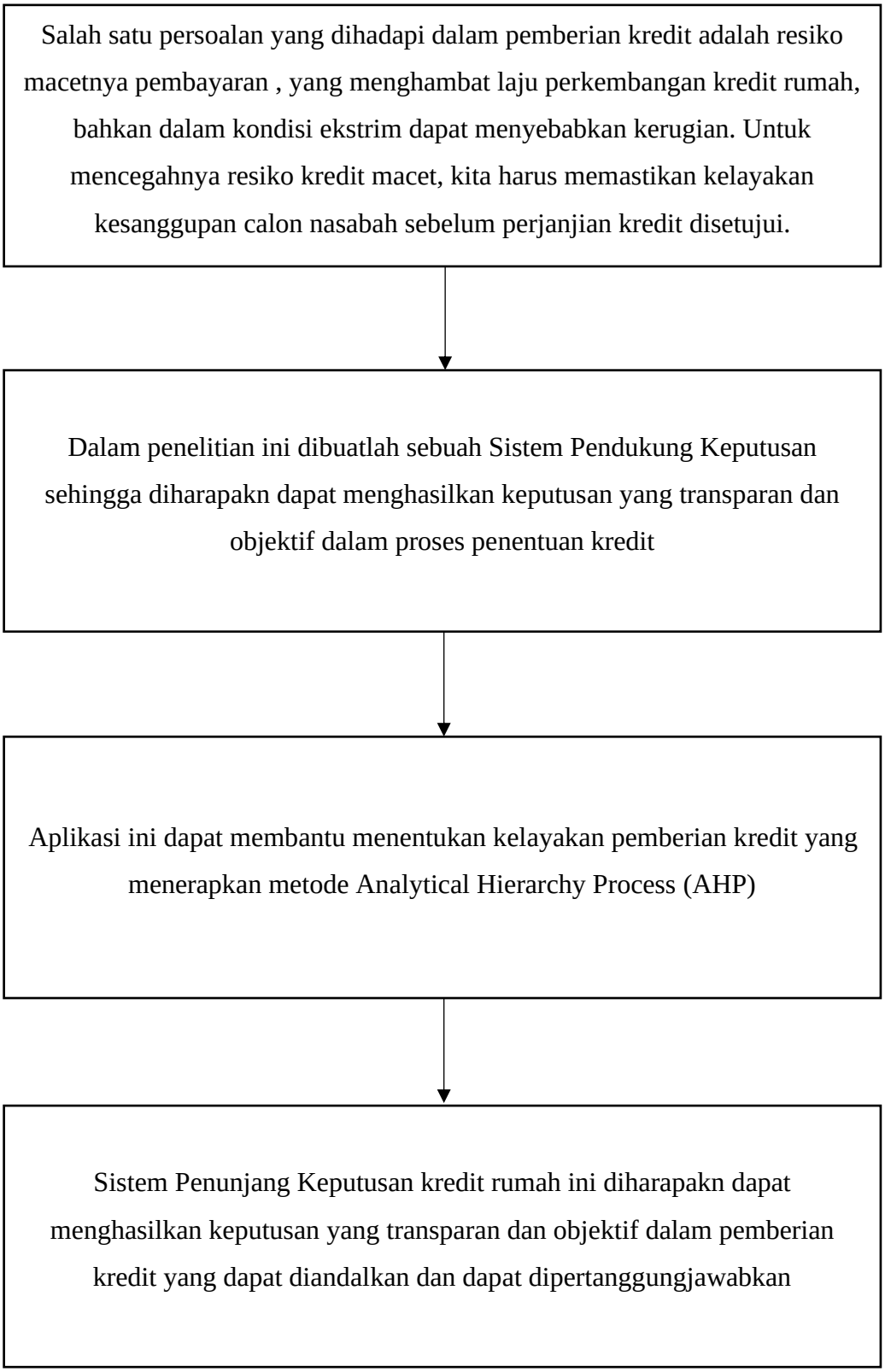
*Black Box Testing* yaitu pengujian desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan (Nugroho dkk., 2018).

### 10. *White Box Testing*

*White Box Testing* adalah test bagian dalam dari sebuah *software development*, pengecekan terhadap detail perancangan, menggunakan struktur kontrol dari desain

program secara *procedural* untuk membagi pengujian ke dalam beberapa kasus pengujian. Secara sekilas dapat diambil kesimpulan *white box testing* merupakan petunjuk untuk mendapatkan program yang benar secara 100%.

**11. Kerangka Pikir**



## B. Kajian Hasil Penelitian Terdahulu

Sebagai bahan pertimbangan dalam penelitian ini akan di cantumkan beberapa penelitian terdahulu yang dilakukan oleh peneliti lain :

1. Penelitian yang dilakukan Agus Jumadi, Zainal Arifin, Dyna Marisa Khairina (2014) tentang "Sistem Pendukung Keputusan pemberian kredit rumah sejahtera pada nasabah bank pembangunan daerah Kalimantan timur dengan metode TOPSIS" Dalam penelitian ini , Bank Pembangunan Daerah (BPD) Kalimantan Timur menjadi Bank yang di ambil penulis sebagai referensi memperoleh pinjaman kredit rumah sejahtera. Selama ini, pengambilan keputusan dilakukan oleh analasis kredit (account officer), selalu memiliki banyak kendala, seperti proses pengambilan keputusan yang tidak efektif (pengguna waktu yang cukup lama, tenaga dan biaya yang terbuang), human eror, dan adanya kemungkinan kolusi antara nasabah dan petugas bank. Sistem ini dibuat berbasis web sehingga pengguna dengan mudah dapat mengakses system ini kapanpun dan dimanapun. (Agus Jumadi, Zainal Arifin, Dyna Marisa Khairina (2014), metode TOPSIS).
2. Penelitian yang dilakukan Resti Lia Andharsaputri (2020) tentang "Sistem Pendukung Keputusan kredit kepemilikan rumah di koperasi " penelitian ini bertujuan untuk membantu pihak koperasi mengambil keputusan menggunakan metode AHP yang merupakan metode pendukung yang dapat memberikan alternatif solusi bagi permasalahan yang bersifat multikriteria. Dari hasil pengolahan data dan implementasi dengan Export Choice, maka hasil dari penelitian ini adalah penentuan kriteria yang menjadi pertimbangan

utama koperasi adalah penghasilan tetap, anggota yang bernama Rizal Subrata adalah anggota yang diprioritaskan untuk menerima pinjaman KPR, serta melalui metode AHP ini dapat diketahui jika perhitungan indeks konsistensi <0,1 maka perhitungan akan menghasilkan solusi optimal.( Resti Lia Andharsaputri (2020), metode AHP)

Tinjauan Hasil Pemelitian Sebelumnya Tentang SPK Kredit Rumah

**Tabel 2. 8** Tinjauan Hasil Penelitian Sebelumnya

| NO | JUDUL | NAMA/ TAHUN | METODE |
|---|---|---|---|
| 1. | SPK pemberian kredit rumah sejahtera pada nasabah bank pembangunan daerah Kalimantan timur dengan metode TOPSIS | Agus Jumadi, Zainal Arifin, Dyna Marisa Khairina/2014 | Penerapan metode TOPSIS digunakan untuk melakukan perhitungan terhadap |
| 2. | SPK Pemberian Kredit Pemilikan Rumah Di Koperasi | Resti Lia Andharsaputri /2020 | Penerapan menggunakan metode AHP yang merupakan metode pendukung yang dapat memberikan alternative solusi |

# BAB III

# METODOLOGI PENELITIAN

## A. Jenis Penelitian

Dalam melakukan penelitian ini, jenis penelitian yang digunakan adalah jenis penelitian empiris. Artinya bahwa penelitian ini dilakukan berdasarkan fakta atau fenomena yang terjadi dimasyarakat, yang akan diulas secara mendalam dan terperinci dari objek yang diteliti.

## B. Waktu dan Tempat Penelitian

Penelitian ini dilakukan di perumahan PT. Taruna Graha Mandiri Kota Parepare untuk memperoleh data sesuai dengan objek yang diteliti dan waktu yang dipergunakan untuk pelaksanaan penelitian ini berlangsung selama ± 2 bulan.

## C. Metode Pengumpulan Data

Untuk memperoleh data-data yang dibutuhkan dan cukup tepat, maka penulis mengumpulkan data-data melalui beberapa cara:

Untuk memperoleh data-data yang dibutuhkan dan cukup tepat, maka penulis mengumpulkan data-data melalui beberapa cara:

1. Metode Pengamatan (Observasi)

   Melalui metode ini, penulis melakukan pengamatan secara langsung proses pemberian kredit yang berlangsung, sehingga dapat diperoleh data, diantaranya data pada bagian pemberian kredit.

2. Metode Wawancara

Dalam metode ini, cara pengumpulan data melalui wawancara langsung atau tanya jawab dengan pihak yang berhubungan dengan penerimaan kredit. Data yang penulis peroleh adalah tentang proses pengambilan keputusan dan data profil.

3. Metode Studi Pustaka

Data-data yang diperoleh dalam metode ini adalah dari buku-buku bacaan, jurnal ilmiah yang ada kaitannya dengan penelitian.

## D. Alat dan Bahan Penelitian

Dalam penelitian ini penulis membutuhkan alat dan bahan untung mendukung kelancaran kegitan penelitian ini seperti perangkat keras (Hardware) dan perangkat lunak (software):

1. Perangkat Keras
    1. Processor : AMD Ryzen5 3550H with Radeon Vega Mobile Gfx (8CPUs) @2.1GHz
    2. RAM : 8 GB
    3. Storage : 1000GB HDD
    4. Monitor : 14”
2. Perangkat lunak
    1. Windows 10
    2. Sublime text

## E. Teknik Pengumpulan Data

Metode Pengumpulan data dilakukan melalui studi pustaka, terutama yang berhubungan data-data sekunder. Sementara data primer dilakukan melalui studi lapangan yaitu berupa:

1. *Wawancara,* yaitu teknik menganalisis data yang dilakukan dengan mengajukan pertanyaan secara langsung kepada responden atau narasumber.
2. *Observasi*, yaitu mengumpulkan data dengan cara mengetahui informasi dan data awal tentang keadaan objek penelitian.
3. *Studi Pustaka,* adalah bagian penting dari setiap proyek peneletian yang baik, dan jika tidak dilakukan dengan benar, proyek dapat berakhir sia-sia dengan mereplikasi penelitian sebelumnya.
4. *Dokumentasi*, yaitu mengumpulkan data dalam bentuk dokumen atau catatan tertulis.

## F. Teknik Analisis Data

Jenis data yang dikumpulkan dalam penelitian ini adalah data primer dan data sekunder yang relevan dengan masalah yang akan dibahas.

1. Data Primer

Data Primer adalah data yang berasal atau data yang diperoleh langsung dari sumber-sumber yang berada di lapangan yang berasal dari wawancara dengan pihak-pihak terkait dengan penelitian ini.

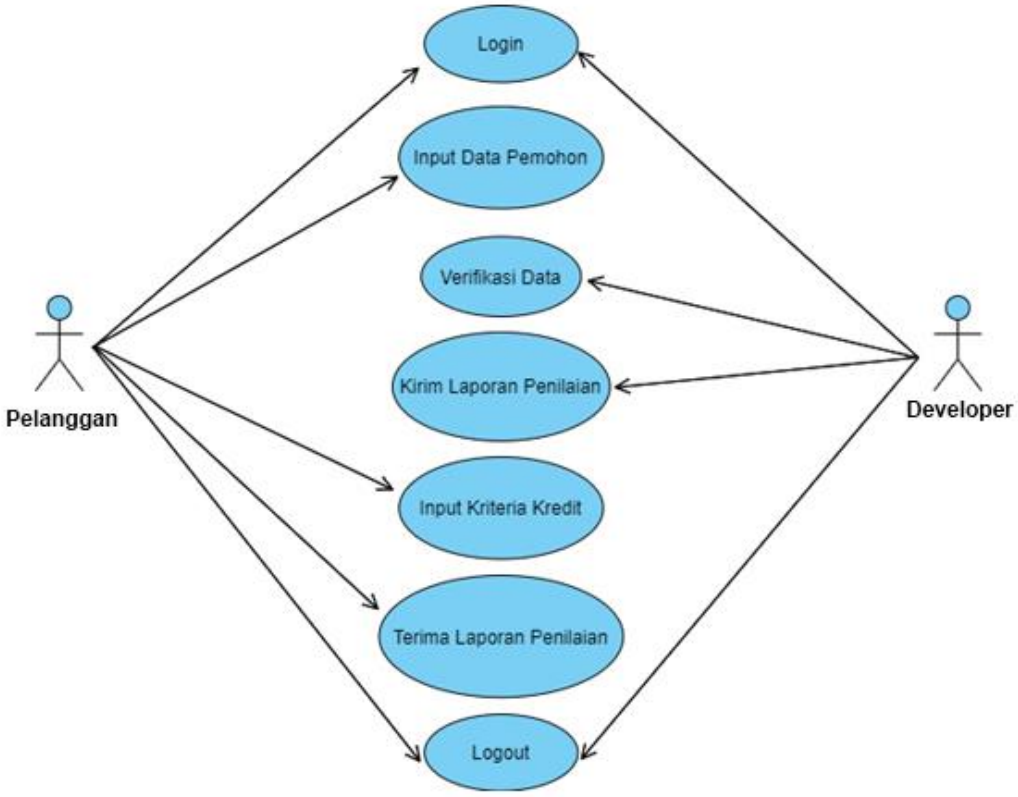
2. Data Sekunder

Data Sekunder adalah data yang diperoleh tidak secara langsung dari objek penelitian. Penelitian mendapatkan data yang sudah jadi yang dikumpulkan oleh pihak lain dengan berbagai cara atau metode baik secara komersial maupun non komersial.

## G. Rancangan Sistem

Rancangan Sistem Penunjang Keputusan Kredit Rumah disajikan menggunakan Unified Modelling Leanguage (UML). Use Case Diagram ini digunakan untuk menggambarkan hubungan sejumlah external actor dengan Use Case yang terdapat dalam sistem. Use Case ini hanya menggambarkan keadaan lingkungan system yang dapat dilihat dari luar faktor.

1. Use Case Diagram




**Gambar 3. 1** *Use Case Diagram*

2. Sistem yang berjalan

**Gambar 3. 2** *Sistem yang Berjalan*

3. Sistem yang diusulkan

**Gambar 3. 3** *Sistem yang Diusulkan*

# BAB IV

# HASIL DAN PEMBAHASAN

## A. Analisis Aliran Data UML

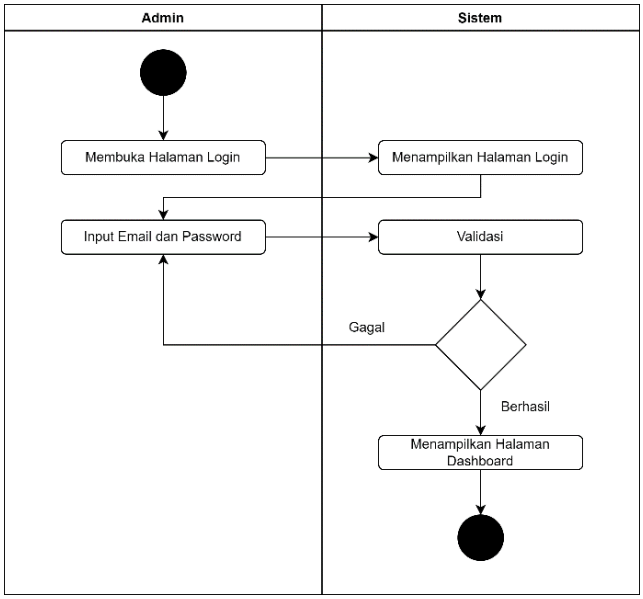
Adapun tujuan dalam perencanaan sistem ini yakni lebih mendalami sebuah proses penentuan kredit terutama mengenai penetuan kredit rumah yang lebih efisien. Pada kesempatan kali ini, peneliti merancang pengembangan yang berorientasi objek sehingga menggunakan use case diagram, activity diagram dan sequence diagram.

### 1. Activity Diagram

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

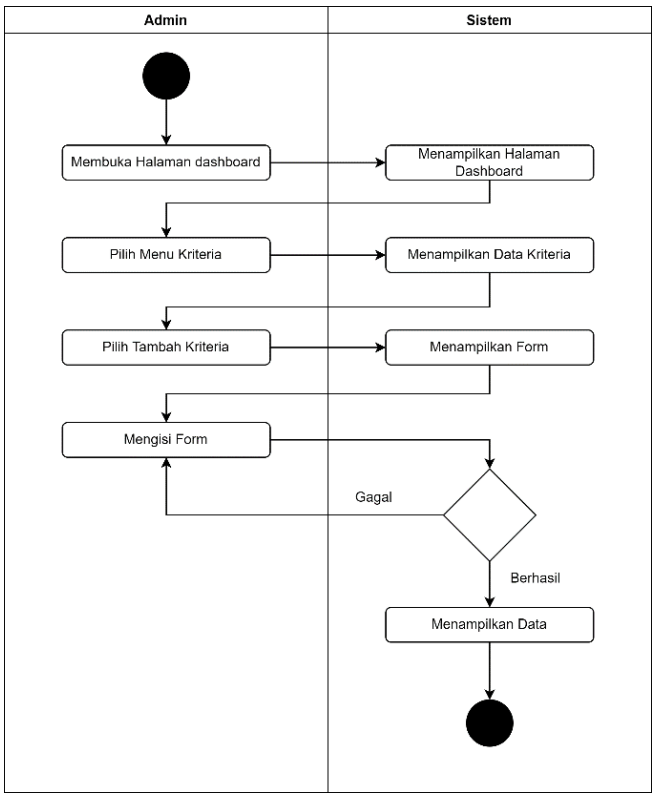
a. *Activity* Diagram Admin

1) Activity diagram Login



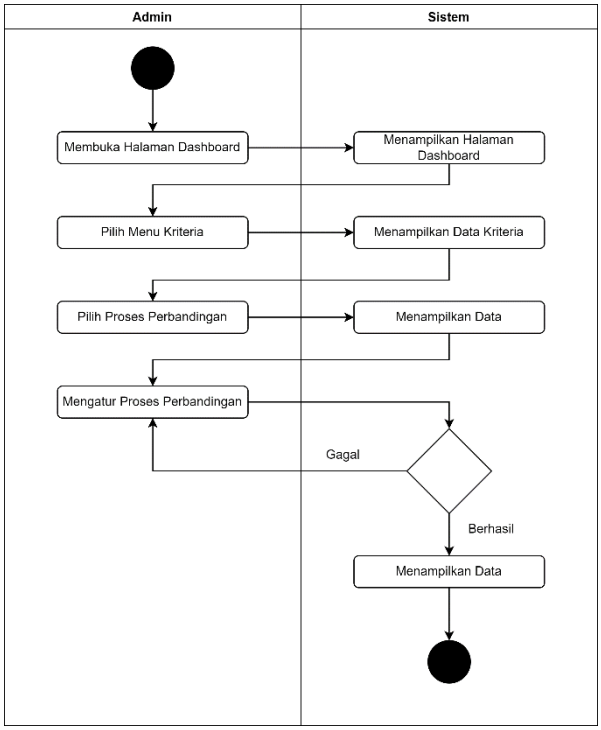

**Gambar 4. 1** *Activity Diagram Login*

2) Activity Diagram Tambah Kriteria



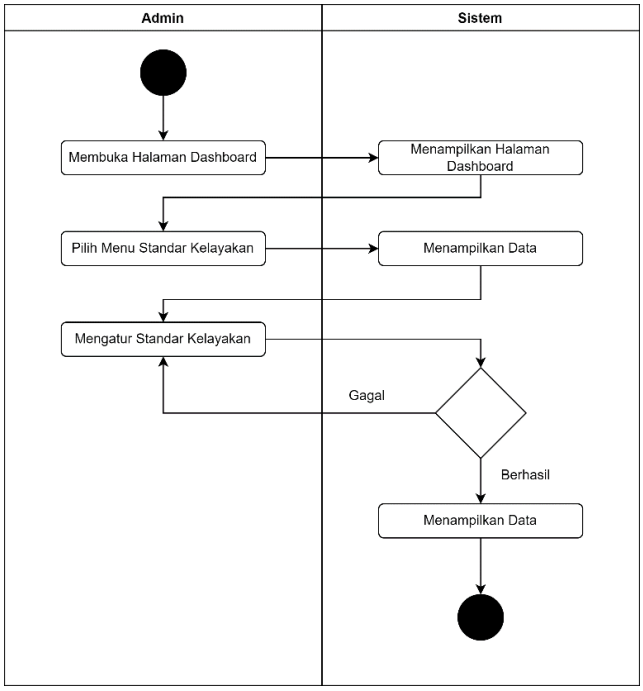

**Gambar 4. 2** *Activity Diagram Tambah Kriteria*

3) Activity Diagram Proses Perbandingan Kriteria



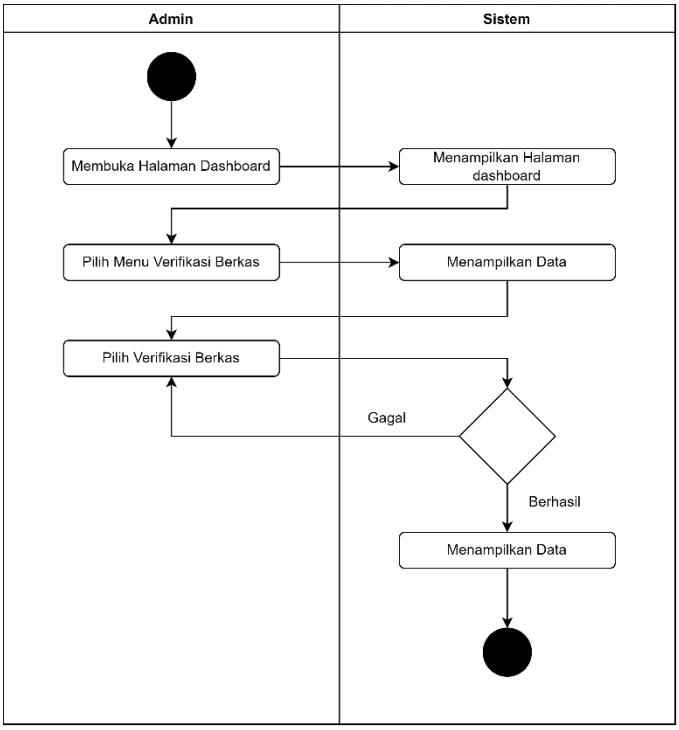

**Gambar 4. 3** *Activity Diagram Proses Perbandingan Kriteria*

4) Activity Diagram Standar Kelayakan




**Gambar 4. 4** *Activity Diagram Standar Kelayakan*

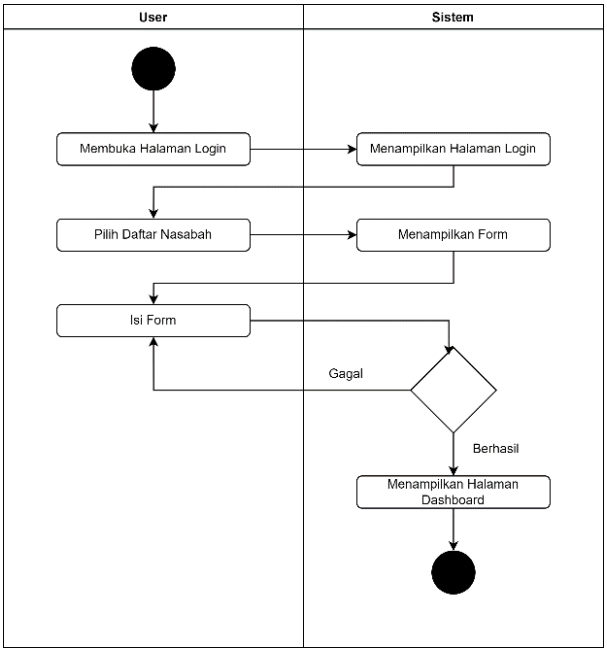
5) Activity Diagram Verifikasi Berkas




**Gambar 4. 5** *Activity Diagram Verifikasi Berkas*

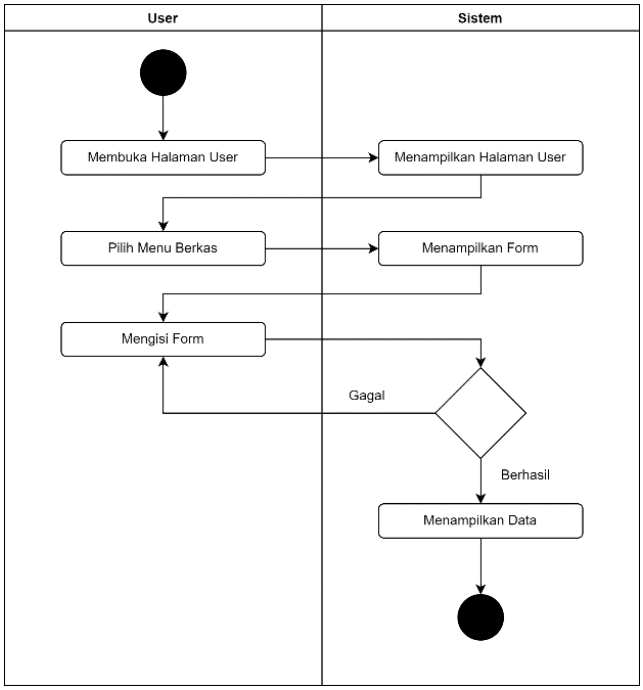
b. Acitivity Diagram User

1) Activity diagram Daftar



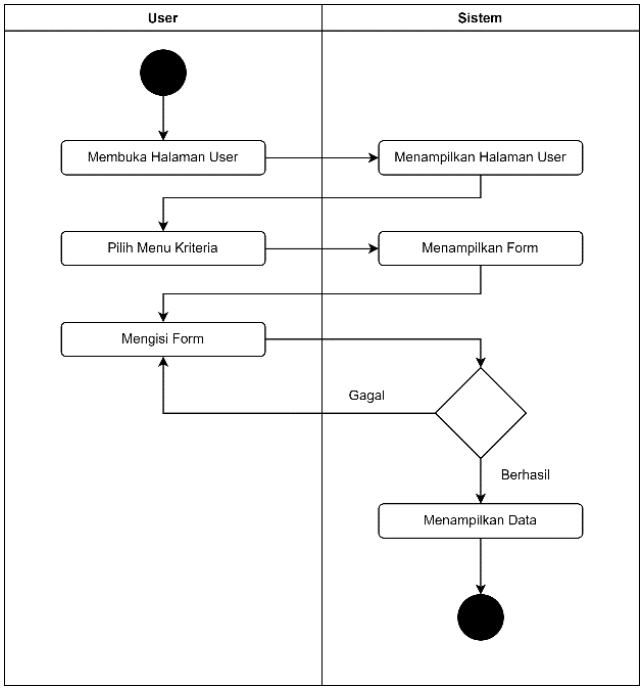

**Gambar 4. 6** *Activity Diagram Daftar*

2) Activity Diagram Upload Berkas




**Gambar 4. 7** *Activity Diagram Upload Berkas*

3) Acitivity Diagram Input Kriteria



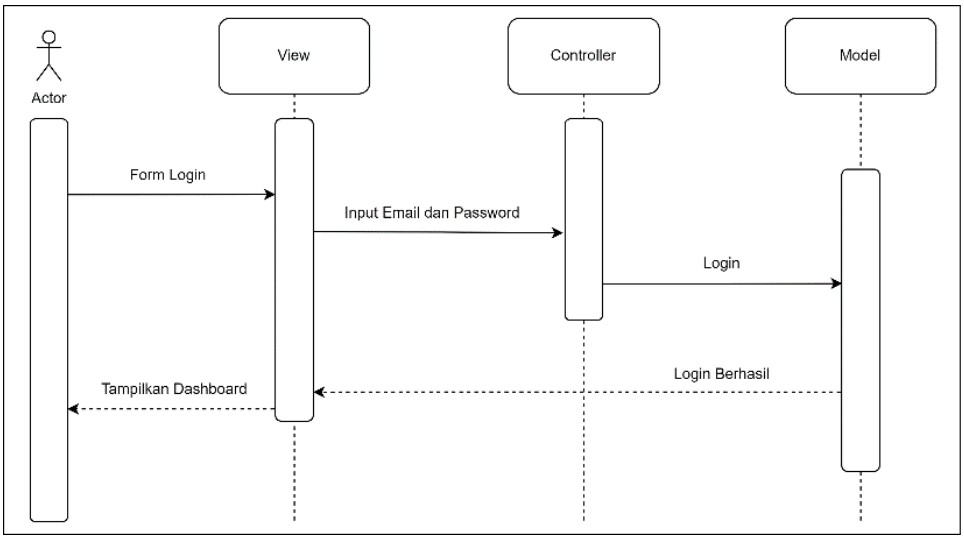

**Gambar 4. 8** *Activity Diagram Input Kriteria*

## 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.

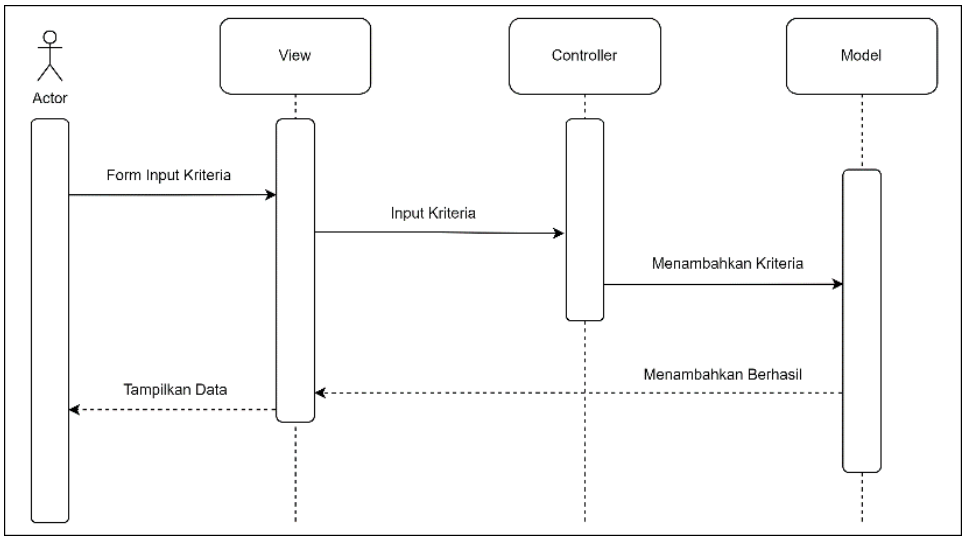
a. Sequence Diagram Admin

1) Sequence Diagram Login



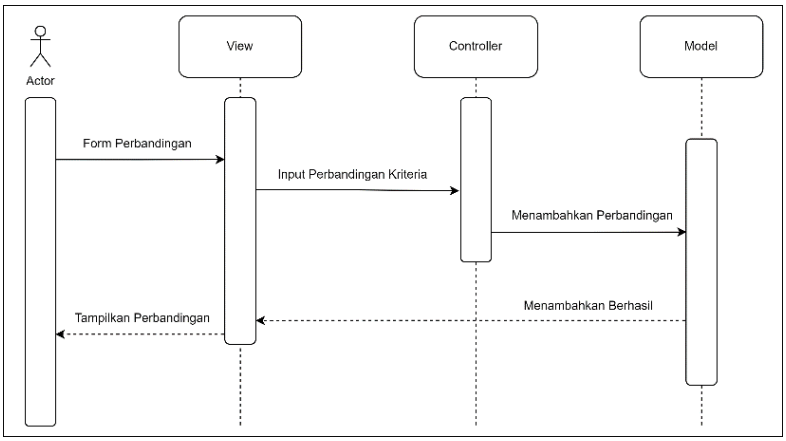

**Gambar 4. 9** *Sequence Diagram Login*

2) Sequence Diagram Tambah Kriteria



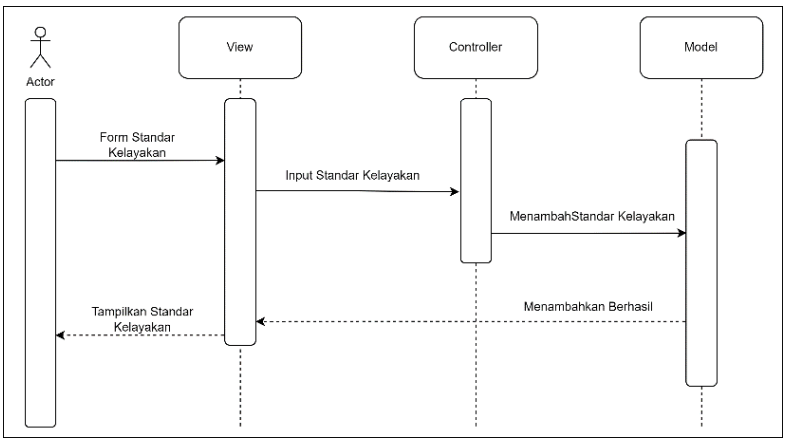

**Gambar 4. 10** *Sequence Diagram Tambah Kriteria*

3) Sequence Diagram Proses Perbandingan Kriteria



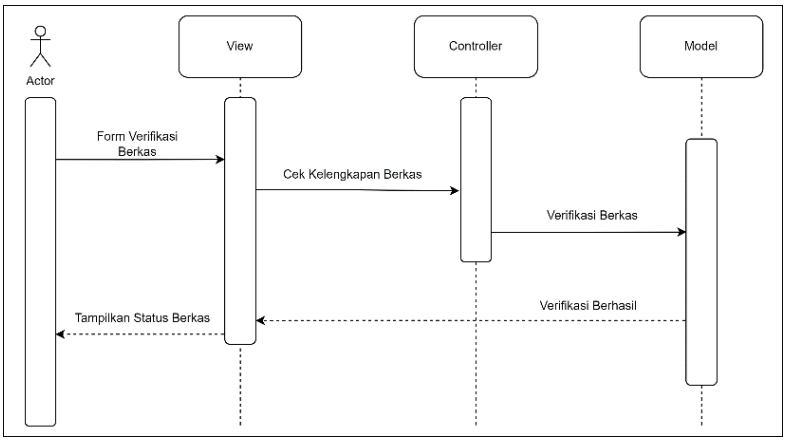

**Gambar 4. 11** *Sequence Diagram Proses Perbandingan Kriteria*

4) Sequence Diagram Standar Kelayakan




**Gambar 4. 12** *Sequence Diagram Standar Kelayakan*

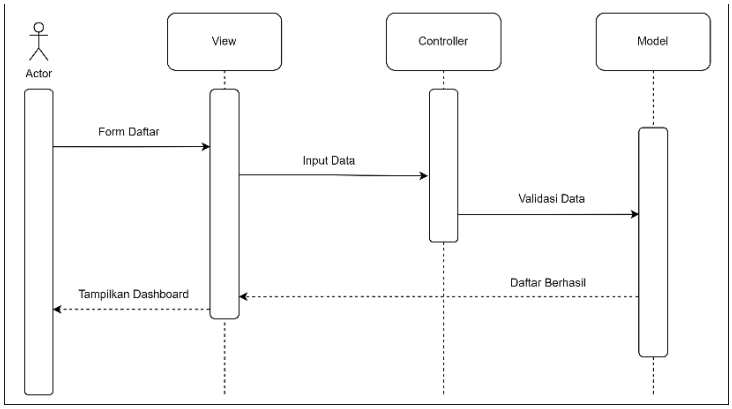
5) Sequence Diagram Verifikasi Berkas




**Gambar 4. 13** *Sequence Diagram Verifikasi Berkas*

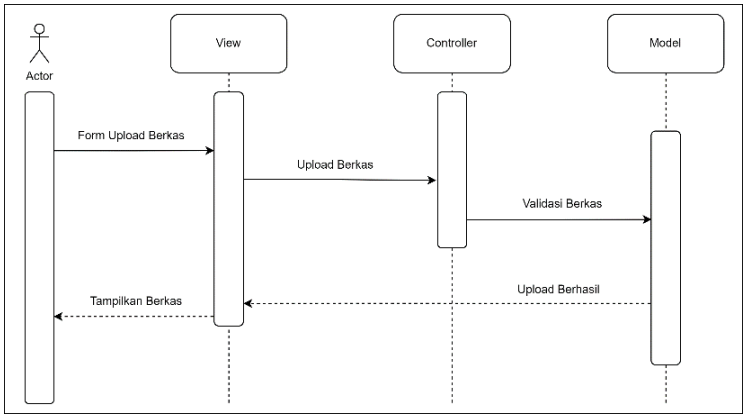
b. Sequence Diagram User

1) Sequence diagram Daftar



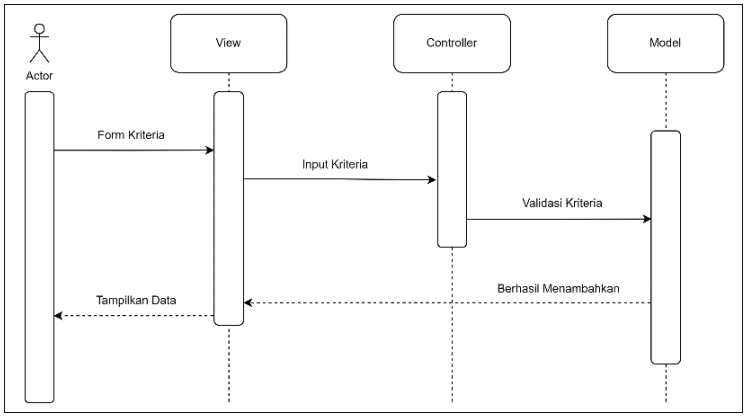

**Gambar 4. 14** *Sequence Diagram Daftar*

2) Sequence Diagram Upload Berkas




**Gambar 4. 15** *Sequence Diagram Upload Berkas*

3) Sequence Diagram Input Kriteria




**Gambar 4. 16** *Sequence Diagram Input Kriteria*

## B. Perancangan Database

Racangan database untuk sistem aplikasi market place hasil industri rumahan daerah enrekang berbasis web dan juga tabel-tabelnya.

### 1. Tabel User

**Tabel 4. 1** *Tabel user*

| No | Nama | Jenis | Ukuran | Ket |
|---|---|---|---|---|
| 1 | id | Bigint | 20 | *AUTO_INCREMENT* |
| 2 | id user | Varchar | 50 | |
| 3 | name | Varchar | 255 | |
| 4 | username | Varchar | 100 | |
| 5 | wa | Varchar | 16 | |
| 6 | email | Varchar | 255 | |
| 7 | foto | Varchar | 500 | |
| 8 | level | Enum | | ‘User’ , ‘Admin’ |
| 9 | password | Varchar | 255 | |

### 2. Tabel Berkas

**Tabel 4. 2** *Tabel Berkas*

| No | Nama | Jenis | Ukuran | Ket |
|---|---|---|---|---|
| 1 | id | Int | 11 | *AUTO_INCREMENT* |
| 2 | id user | Varchar | 50 | |
| 3 | status | Enum | | *'Tunggu Verifikasi','Terverifikasi','Ditolak'* |
| 4 | ktp | Varchar | 500 | |
| 5 | kk | Varchar | 500 | |
| 6 | surat_keterangan_kerja | Varchar | 500 | |
| 7 | surat_keterangan_belum_nikah | Varchar | 500 | |
| 8 | buku_rekening | Varchar | 500 | |
| 9 | npwp | Varchar | 500 | |

### 3. Tabel Kriteria

**Tabel 4. 3** *Tabel Kriteria*

| No | Nama | Jenis | Ukuran | Ket |
|---|---|---|---|---|
| 1 | id | Int | 11 | *AUTO_INCREMENT* |
| 2 | nama_kriteria | Varchar | 50 | |

### 4. Tabel Hasil Test

**Tabel 4. 4** *Tabel Hasil Test*

| No | Nama | Jenis | Ukuran | Ket |
|---|---|---|---|---|
| 1 | id | Int | 11 | *AUTO_INCREMENT* |
| 2 | id_user | Varchar | 50 | |
| 3 | input_kriteria | Longtext | | |
| 4 | status_kelayakan | Enum | | *'Layak menerima kredit','tidak layak menerima kredit'* |

### 5. Tabel Perbandingan Kriteria

**Tabel 4. 5** *Tabel Perbandingan Kriteria*

| No | Nama | Jenis | Ukuran | Ket |
|---|---|---|---|---|
| 1 | id | Int | 11 | |
| 2 | matrix_inputan | Text | | |
| 3 | matrix_perbadingan | Text | | |
| 4 | jumlah_per_baris | Text | | |
| 5 | eigen | Text | | |
| 6 | jml_eigen | Text | | |
| 7 | rata2_eigen | Text | | |
| 8 | total_perhitungan | Varchar | 50 | |
| 9 | lamda_max | Varchar | 50 | |
| 10 | ci | Varchar | 50 | |
| 11 | ir | Varchar | 50 | |
| 12 | cr | Varchar | 50 | |

### 6. Tabel Standar Kelayakan

**Tabel 4. 6** *Tabel Standar Kelayakan*

| No | Nama | Jenis | Ukuran | Ket |
|---|---|---|---|---|
| 1 | id | Int | 11 | |
| 2 | nilai_max | Varchar | 50 | |
| 3 | nilai_min | Varchar | 50 | |

### 7. Tabel Sub Kriteria

**Tabel 4. 7** *Tabel Sub Kriteria*

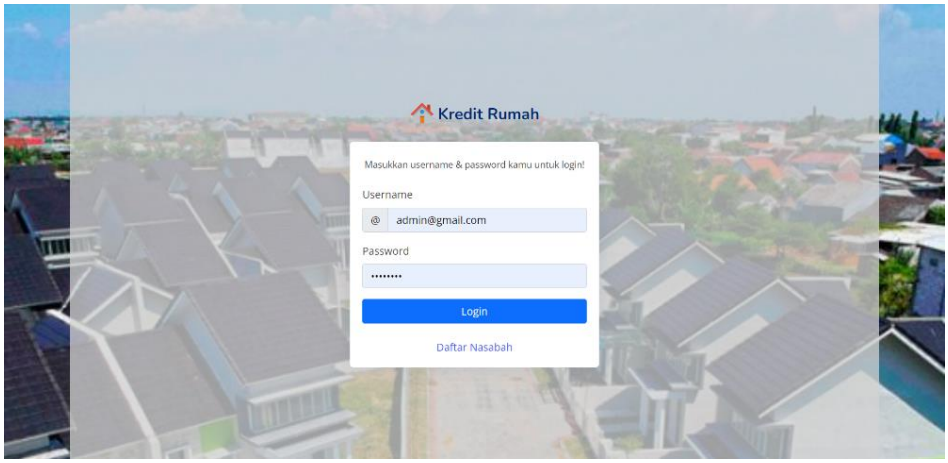
| No | Nama | Jenis | Ukuran | Ket |
|---|---|---|---|---|
| 1 | id | Int | 11 | *AUTO_INCREMENT* |
| 2 | id_kriteria | Int | 11 | |
| 3 | nama_sub_kriteria | Varchar | 50 | |
| 4 | nilai_bobot | Int | 11 | |

## C. Detail Sistem

### 1. Admin

a. Halaman Login

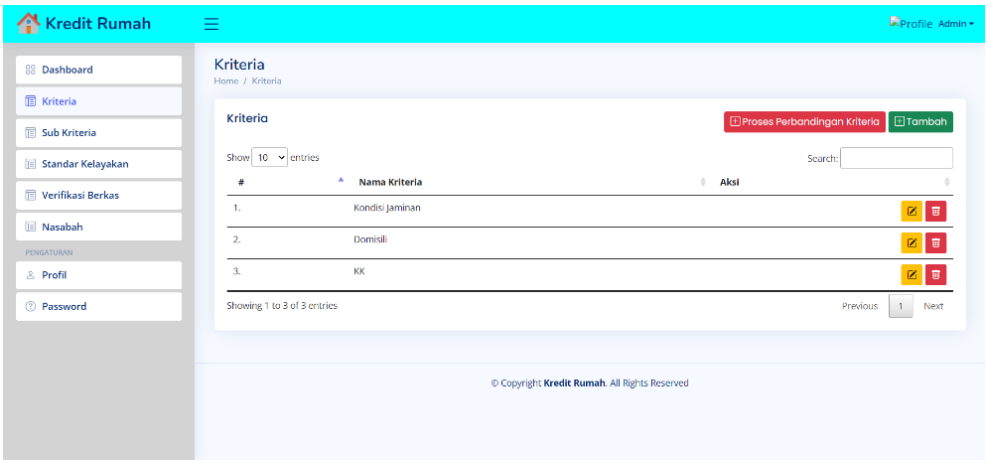
Halaman Login merupakan titik awal yang digunakan oleh admin untuk memperoleh akses masuk ke beragam fitur yang tersedia di dalam halaman menu admin. Ini adalah pintu gerbang yang memungkinkan admin untuk masuk ke dalam sistem dengan mengautentikasi identitas mereka.




**Gambar 4. 17** *Halaman Login*

b. Halaman Tambah Kriteria

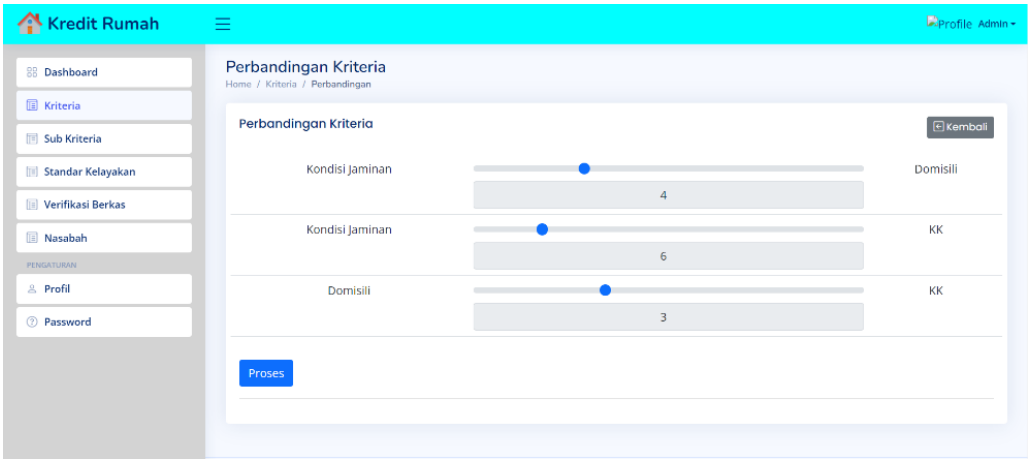
Halaman Tambah Kriteria adalah bagian dari sistem yang dirancang khusus untuk memungkinkan pengguna, terutama admin, menambahkan kriteria yang diperlukan bagi nasabah yang ingin mengajukan kredit rumah.




**Gambar 4. 18** *Halaman Tambah Kriteria*

c. Halaman Proses Perbandingan Kriteria

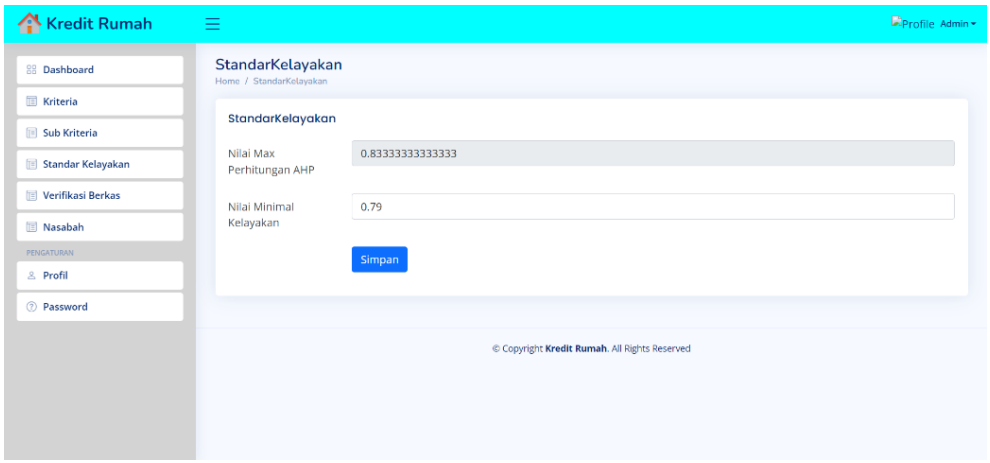
Halaman Proses Perbandingan Kriteria merupakan halaman untuk melakukan perbandingan antara satu kriteria dengan kriteria lainnya.Pada halaman ini, admin dapat menentukan bobot atau tingkat kepentingan relatif dari setiap kriteria dalam pengambilan keputusan.




**Gambar 4. 19** *Halaman Proses Perbandingan Kriteria*

d. Halaman Standar Kelayakan

Halaman Standar Kelayakan merupakan halaman untuk menetapkan nilai kelayakan standar yang harus dipenuhi oleh calon nasabah. Nilai kelayakan ini berhubungan secara langsung dengan nilai maksimum perhitungan metode Analytic Hierarchy Process (AHP).

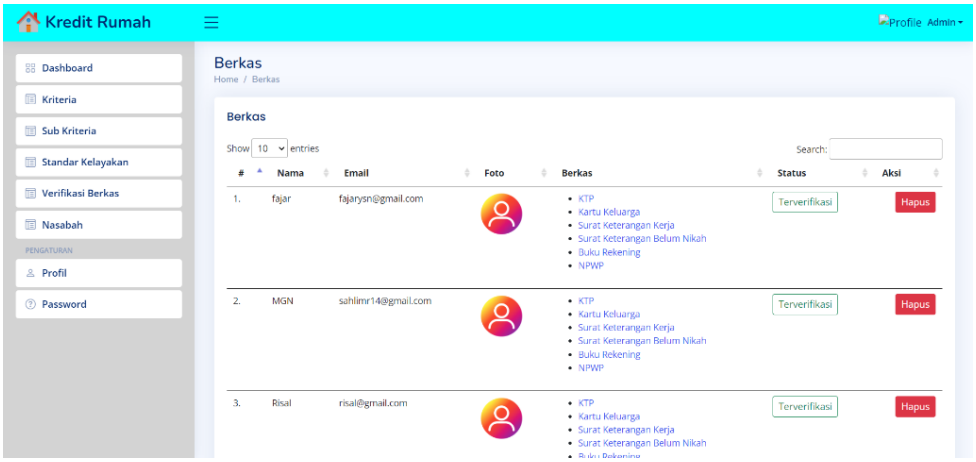



**Gambar 4. 20** *Halaman Standar kelayakan*

e. Halaman Verifikasi Berkas

Halaman Verifikasi Berkas merupakan halaman untuk melakukan pengecekan terhadap kelengkapan berkas yang diajukan oleh nasabah.

Pada halaman ini, admin memiliki tanggung jawab untuk memastikan bahwa semua dokumen yang diperlukan telah tersedia dan lengkap sesuai dengan persyaratan yang telah ditetapkan.




**Gambar 4. 21** *Halaman Verifikasi Berkas*

f. Max Kriteria

| MAX KRITERIA | | | |
|---|---|---|---|
| Kriteria | Kondisi Jaminan | Domisili | KK |
| Kondisi Jaminan | 1 | 4 | 6 |
| Domisili | 0,25 | 1 | 3 |
| KK | 0,166666667 | 0,333333333 | 1 |
| Jumlah | 1,416666667 | 5,333333333 | 10 |
| NILAI EIGEN | | | |
| Kriteria | Kondisi Jaminan | Domisili | KK |
| Kondisi Jaminan | 0,705882353 | 0,75 | 0,6 |
| Domisili | 0,176470588 | 0,1875 | 0,3 |
| KK | 0,117647059 | 0,0625 | 0,1 |

**Gambar 4. 22** *Max Kriteria Admin*

g. Admin Rata-rata Eigen

JUMLAH RATA-RATA EIGEN

| Kriteria | Jumlah | Rata-Rata |
|---|---|---|
| Kondisi Jaminan | 2,055882353 | 0,685294118 |
| Domisili | 0,663970588 | 0,221323529 |
| KK | 0,280147059 | 0,093382353 |
| Rata-rata | | 1 |

Consistency Ratio

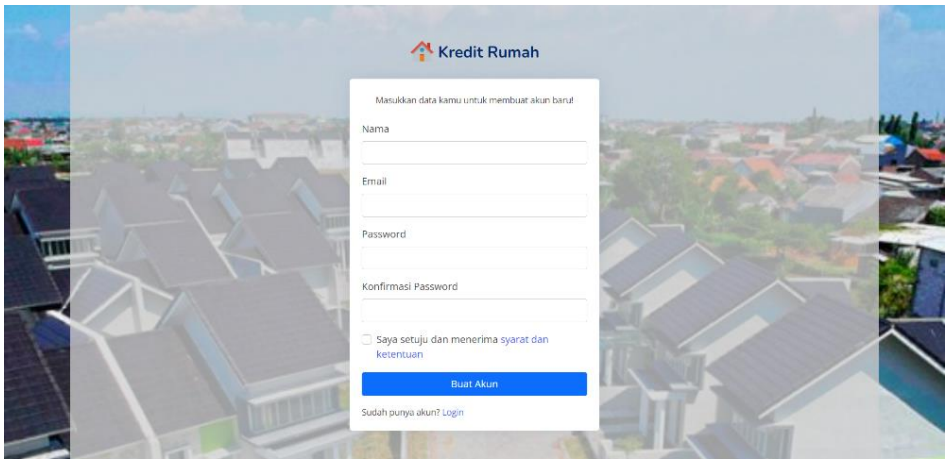
| | |
|---|---|
| Lamda Max | 3,08504902 |
| Indeks Konsistensi | 0,04252451 |
| Indeks Random | 0,58 |
| Consistency Ratio | 0,07331812 |
| Status | |

**Gambar 4. 23** *Admin Rata-rata Eigen*

## 2. User

a. Halaman Daftar

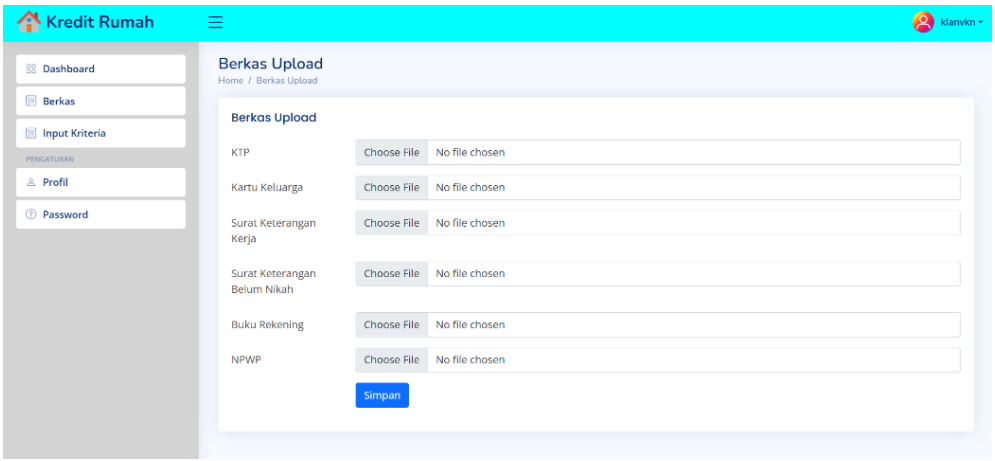
Halaman Daftar adalah tempat yang disediakan bagi calon nasabah yang ingin mengambil kredit rumah sebagai langkah awal dalam proses pendaftaran. Pada halaman ini, calon nasabah dapat mengisi informasi pribadi dan detail yang diperlukan oleh sistem untuk memulai proses pengajuan kredit rumah.




**Gambar 4. 24** *Halaman Daftar Nasabah*

b. Halaman Upload Berkas

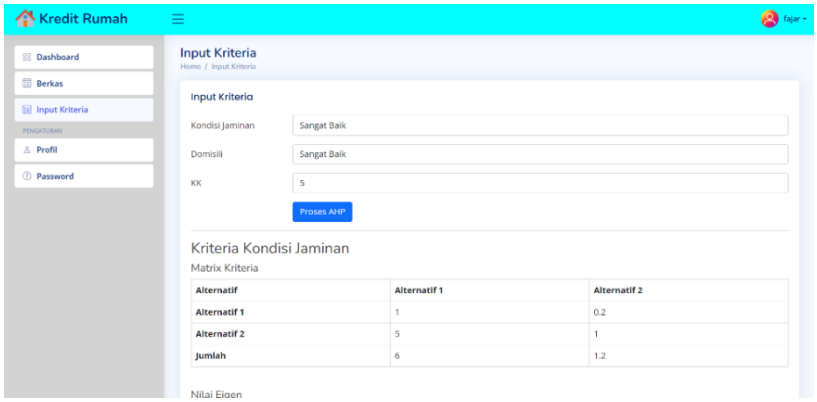
Halaman Upload Berkas merupakan bagian dari proses pengajuan kredit rumah yang memungkinkan nasabah untuk mengunggah berkas-berkas yang diperlukan sesuai dengan persyaratan yang tercantum dalam formulir upload berkas.




**Gambar 4. 25** *Halaman Upload Berkas*

c. Halaman Input Kriteria

Halaman Input Kriteria adalah bagian dari proses pengajuan kredit rumah yang memungkinkan nasabah untuk memberikan informasi tentang kriteria dan jaminan yang dimiliki.




**Gambar 4. 26** *Halaman Input Kriteria*

d. Layak 1 (Kriteria Domisili)

Kriteria Domisli

Matrix Kriteria

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 1 | 0,25 |
| Alternatif 2 | 4 | 1 |
| Jumlah | 5 | 1,25 |

Nilai Eigen

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 0,2 | 0,2 |
| Alternatif 2 | 0,8 | 0,8 |

Jumlah Rata-Rata Eigen

| Kriteria | Jumlah | Rata-rata |
|---|---|---|
| Alternatif 1 | 0,4 | 0,2 |
| Alternatif 2 | 1,6 | 0,8 |
| Total Rata-rata | | 1 |

**Gambar 4. 27** *Layak 1 (Kriteria Domisili)*

e. Layak 2 (Kriteria Kartu Keluarga)

Kriteria KK

Matrix Kriteria

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 1 | 0,25 |
| Alternatif 2 | 4 | 1 |
| Jumlah | 5 | 1,25 |

Nilai Eigen

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 0,2 | 0,2 |
| Alternatif 2 | 0,8 | 0,8 |

Jumlah Rata-Rata Eigen

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 0,4 | 0,2 |
| Alternatif 2 | 1,6 | 0,8 |
| Total Rata-rata | | 1 |

**Gambar 4. 28** *Layak 2 (Kriteria Kartu Keluarga)*

f. Layak 3 (Perangkingan)

Perangkingan

| | |
|---|---|
| Alternatif 1 | 0,2 |
| Alternatif 2 | 0,8 |
| Total | 1 |
| Nilai Rangking | 0,8 |
| Standar Kelayakan | |
| Status Kelayakan | |

**Gambar 4. 29** *Layak 3 (Perangkingan)*

g. Layak 4 (Kriteria Kondisi Jaminan)

Kriteria Kondisi Jaminan

Matrix Kriteria

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 1 | 0,25 |
| Alternatif 2 | 4 | 1 |
| Jumlah | 5 | 1,25 |

Nilai Eigen

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 0,2 | 0,2 |
| Alternatif 2 | 0,8 | 0,8 |

Jumlah Rata-Rata Eigen

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 0,4 | 0,2 |
| Alternatif 2 | 1,6 | 0,8 |
| Total Rata-rata | | 1 |

**Gambar 4. 30** *Layak 4 (Kriteria Kondisi Jaminan)*

h. Tidak Layak 1 (Kriteria Domisili)

Kriteria Domisli

Matrix Kriteria

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 1 | 1 |
| Alternatif 2 | 1 | 1 |
| Jumlah | 2 | 2 |

Nilai Eigen

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 0,5 | 0,5 |
| Alternatif 2 | 0,5 | 0,5 |

Jumlah Rata-Rata Eigen

| Kriteria | Jumlah | Rata-rata |
|---|---|---|
| Alternatif 1 | 1 | 0,5 |
| Alternatif 2 | 1 | 0,5 |
| Total Rata-rata | | 1 |

**Gambar 4. 31** *Tidak Layak 1 (Kriteria Domisili)*

i. Tidak Layak 2 (Kriteria Kartu Keluarga)

Kriteria KK

Matrix Kriteria

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 1 | 1 |
| Alternatif 2 | 1 | 1 |
| Jumlah | 2 | 2 |

Nilai Eigen

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 0,5 | 0,5 |
| Alternatif 2 | 0,5 | 0,5 |

Jumlah Rata-Rata Eigen

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 1 | 0,5 |
| Alternatif 2 | 1 | 0,5 |
| Total Rata-rata | | 1 |

**Gambar 4. 32** *Tidak Layak 2 (Kriteria Kartu Keluarga)*

j. Tidak Layak 3 (Perangkingan)

Perangkingan

| | |
|---|---|
| Alternatif 1 | 0,5 |
| Alternatif 2 | 0,5 |
| Total | 1 |
| Nilai Rangking | 0,5 |
| Standar Kelayakan | |
| Status Kelayakan | |

**Gambar 4. 33** Tidak Layak 3 (Perangkingan)

k. Tidak Layak 4 (Kriteria Kondisi Jaminan)

Kriteria Kondisi Jaminan

Matrix Kriteria

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 1 | 1 |
| Alternatif 2 | 1 | 1 |
| Jumlah | 2 | 2 |

Nilai Eigen

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 0,5 | 0,5 |
| Alternatif 2 | 0,5 | 0,5 |

Jumlah Rata-Rata Eigen

| Kriteria | Alternatif 1 | Alternaitf 2 |
|---|---|---|
| Alternatif 1 | 1 | 0,5 |
| Alternatif 2 | 1 | 0,5 |
| Total Rata-rata | | 1 |

**Gambar 4. 34** *Tidak Layak 4 (Kriteria Kondisi Jaminan)*

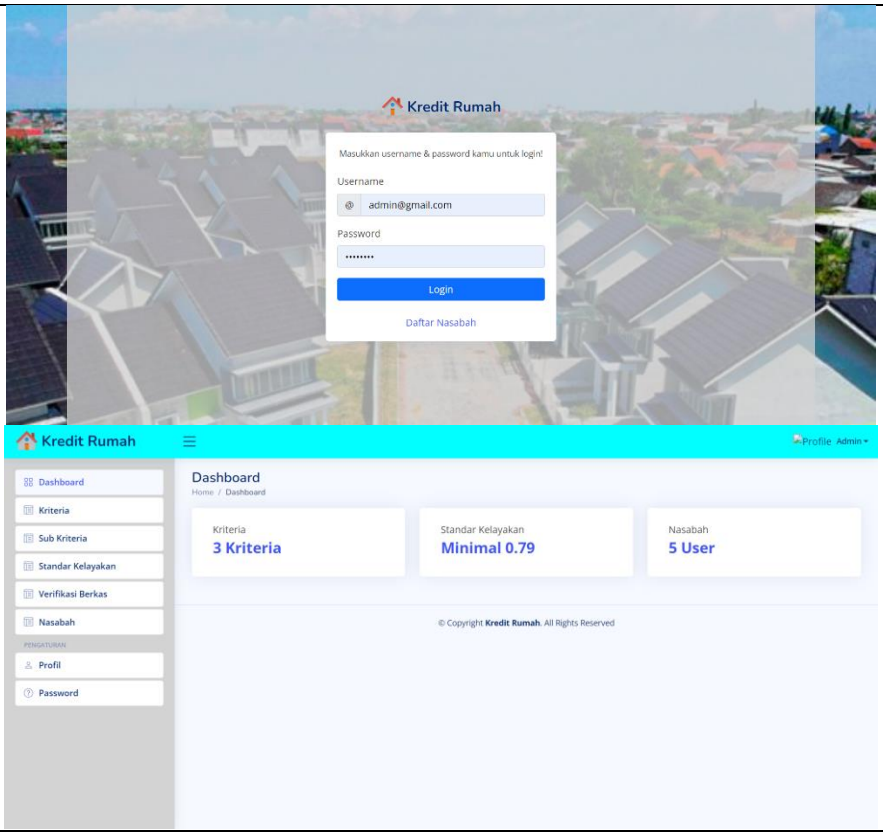
## D. Pengujian Sistem

Pengujian sistem pada penelitian ini menggunakan metode *Black Box Testing* dan juga *White Box Testing*. Berikut di bawah ini hasil pengujian menggunakan metode tersebut.

### 1. *Black Box*

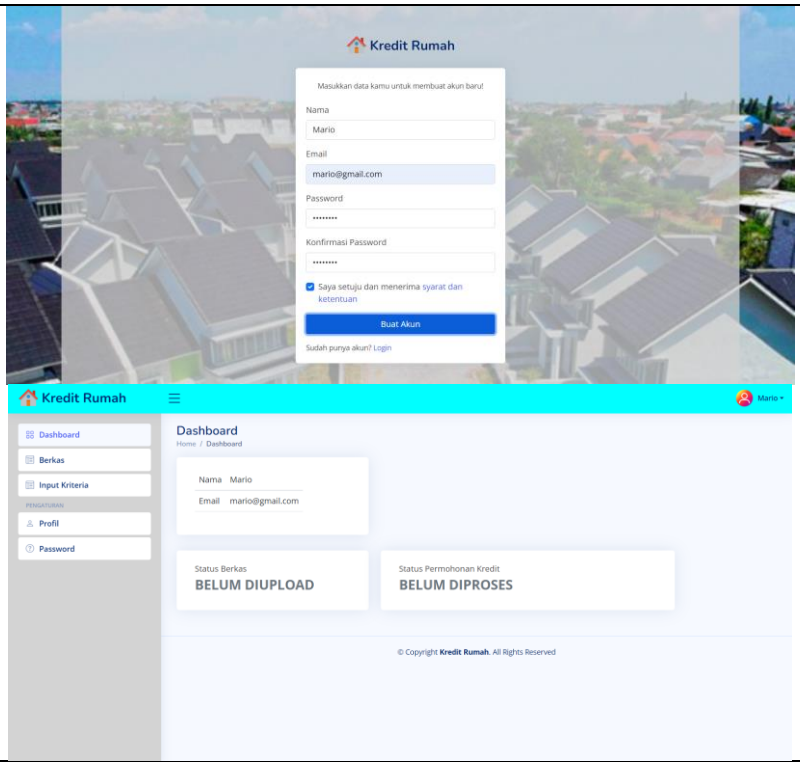
a. *Black Box Testing* Login Admin

**Tabel 4. 8** *Black Box Testing Login Admin*

| Tes Faktor | Hasil | Keterangan |
|---|---|---|
| Memasukkan *email* atau *password* yang tidak sesuai | ✓ | Berhasil, ketika *email* atau *password* sesuai maka tampil Halaman Dashboard |
| ***Screenshot*** | | |
|  | | |

b. Black Box Testing Daftar Nasabah

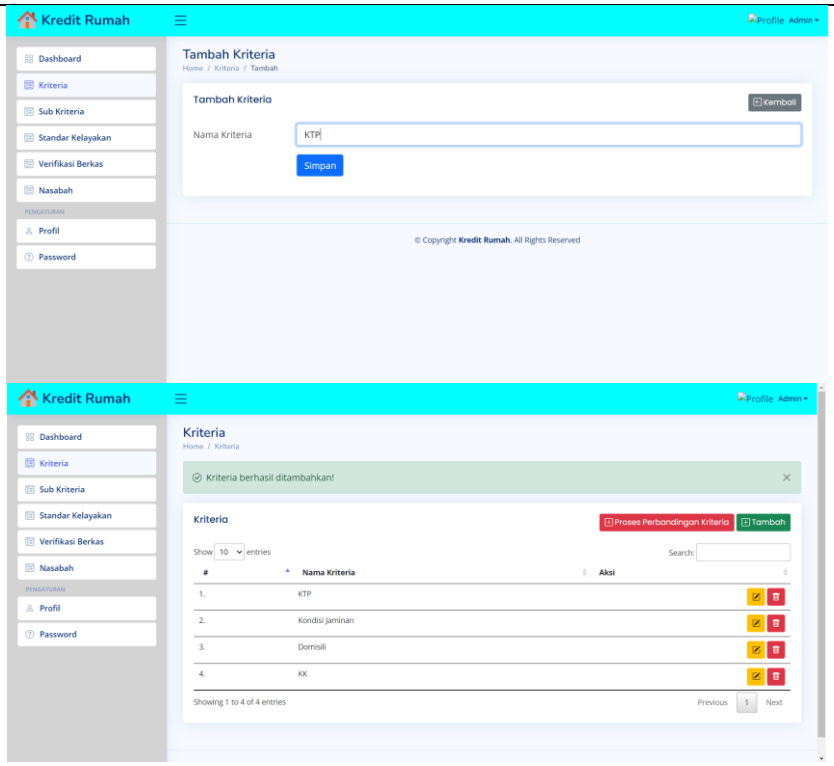
**Tabel 4. 9** *Black Box Testing Daftar Nasabah*

| Tes Faktor | Hasil | Keterangan |
|---|---|---|
| Memasukkan Data sesuai yang tercantum di forma daftar | ✓ | Berhasil, ketika data yang di masukkan sesuai, dan melewati proses validasi dan akan langsung masuk ke halaman dashboard |
| ***Screenshot*** | | |
|  | | |

c. *Black Box Testing* Tambah Kriteria

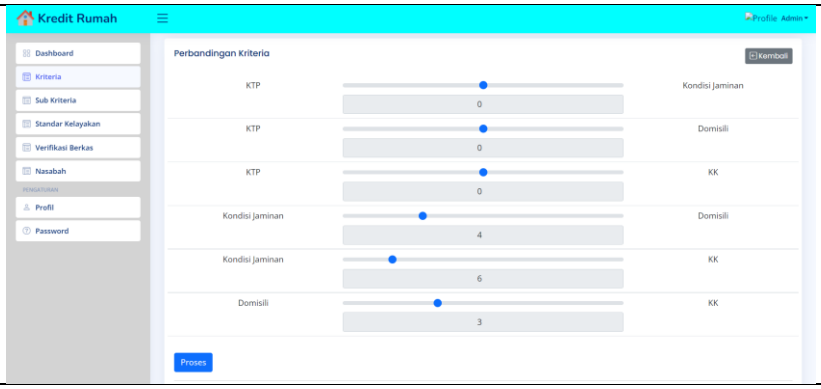
**Tabel 4. 10** *Black Box Testing Tambah Kriteria*

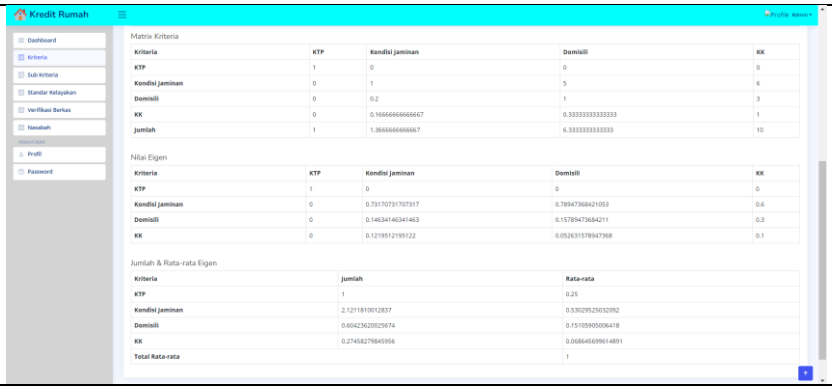
| Tes Faktor | Hasil | Keterangan |
|---|---|---|
| Memasukkan Kriteria kedalam form | ✓ | Berhasil, ketika data yang di masukkan sesuai, dan melewati proses validasi dan akan muncul notifikasi bahwa data berhasil ditambahkan |
| | | |

| ***Screenshot*** |
| --- |
|  |

d. Black Box Testing Proses Perbandingan Kriteria

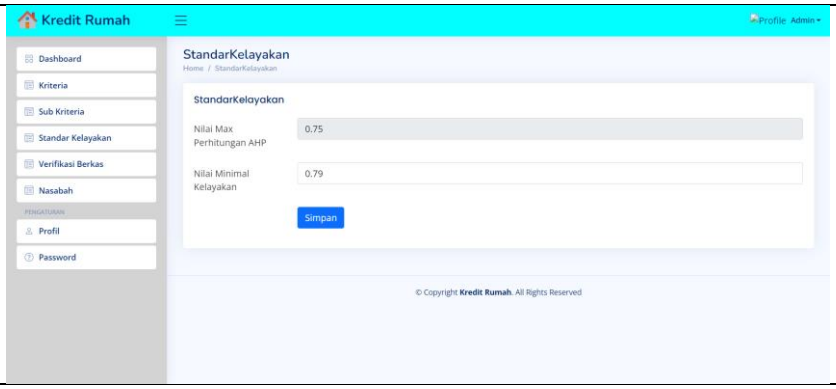
**Tabel 4. 11** *Black Box Testing Proses Perbandingan Kriteria*

| Tes Faktor | Hasil | Keterangan |
| --- | --- | --- |
| Mengatur nilai dari tiap kriteria | ✓ | Berhasil, akan tampil hasil dari perbandingan setiap kriteria. |
| ***Screenshot*** | | |
|  | | |




e. Black Box Testing Standar Kelayakan

**Tabel 4. 12** *Black Box Testing Standar Kelayakan*

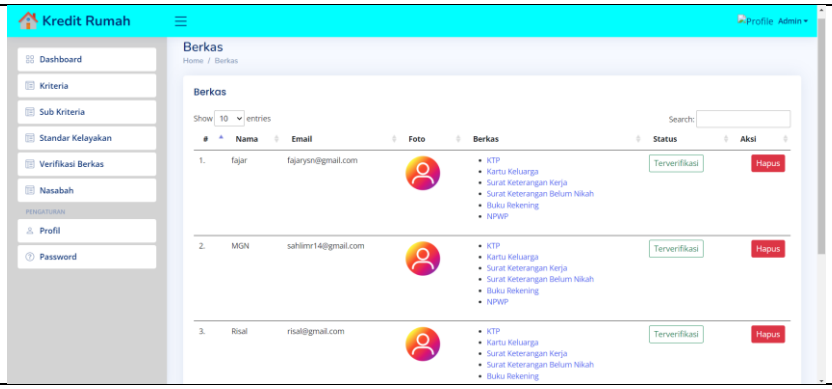
| Tes Faktor | Hasil | Keterangan |
|---|---|---|
| Memasukkan nilai minimal kelayakan | ✓ | Berhasil, nilai tersebut akan di simpan kedalam sistem dan akan menjadi standar kelayakan. |
| ***Screenshot*** | | |
|  | | |



f. Black Box Testing Verifikasi Berkas

**Tabel 4. 13** *Black Box Testing Verifikasi Berkas*

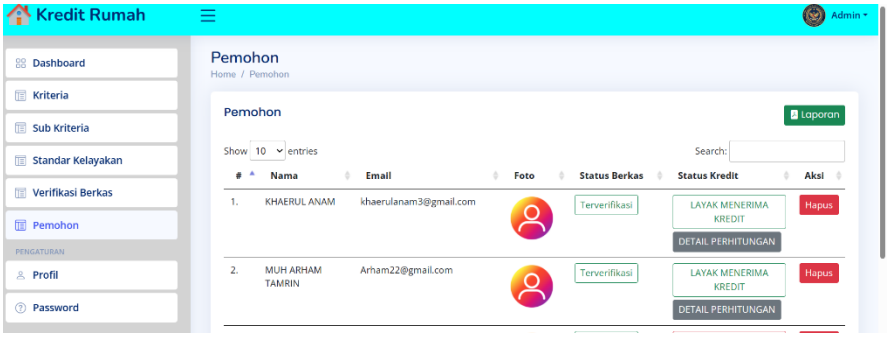
| Tes Faktor | Hasil | Keterangan |
|---|---|---|
| Memverifikasi berkas dari nasabah yang sudah melengkapi berkas | ✓ | Berhasil, ketika admin telah mengecek kelengkapan berkas nasabah |
| | | |

| ***Screenshot*** |
| --- |
|  |



g. Black Box Testing Berkas Diterima

**Tabel 4. 14** *Black Box Testing Berkas Diterima*

| Tes Faktor | Hasil | Keterangan |
| --- | --- | --- |
| Verifikasi berkas pemohon apakah layak menenerima Kredit Rumah | ✓ | Berhasil, Terverifikasi sebagai penerima Kredit Rumah |
| ***Screenshot*** | | |
|  | | |



h. Black Box Testing Berkas Ditolak

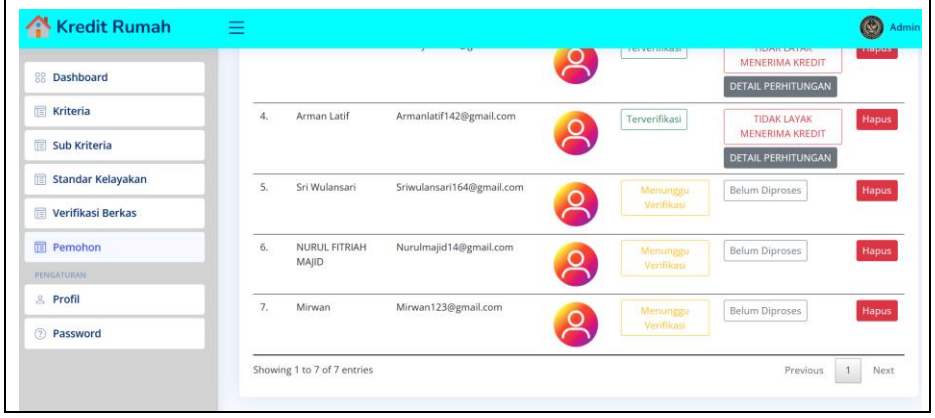
**Tabel 4. 15** *Black Box Testing Berkas Ditolak*

| Tes Faktor | Hasil | Keterangan |
| --- | --- | --- |
| Verifikasi berkas pemohon apakah layak menenerima Kredit Rumah | ✓ | Berhasil, Ketika terverifikasi tidak layak menerima Kredit Rumah |
| ***Screenshot*** | | |

i. Black Box Testing Berkas Pending

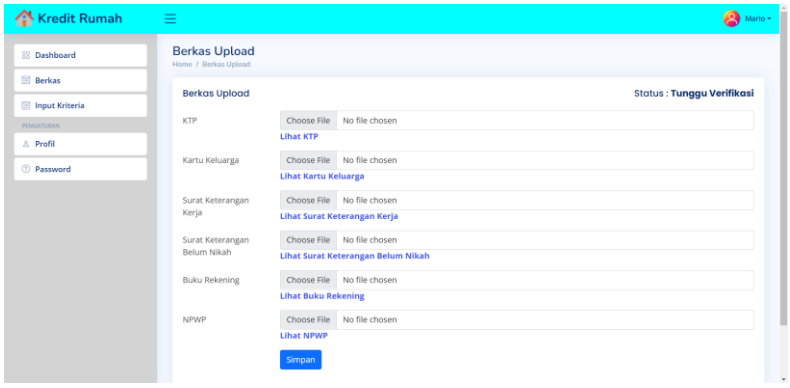
**Tabel 4. 16** *Black Box Testing Berkas Pending*

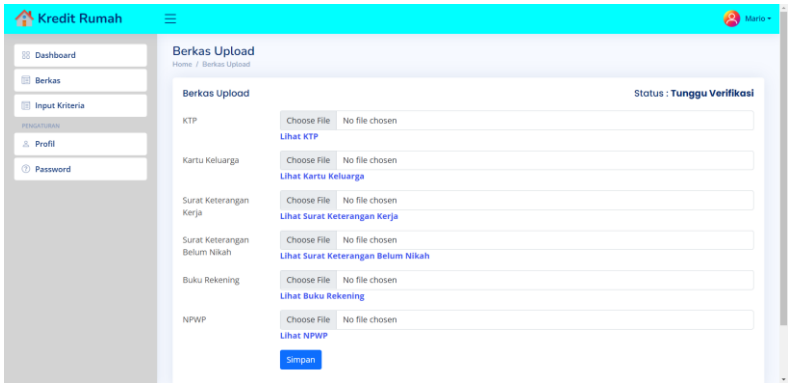
| Tes Faktor | Hasil | Keterangan |
|---|---|---|
| Verifikasi berkas pemohon apakah layak menenerima Kredit Rumah | ✓ | Berhasil, Ketika masih dalam proses menunggu verifikasi |
| ***Screenshot*** | | |



j. Black Box Testing Upload Berkas

**Tabel 4. 17** *Black Box Testing Upload Berkas*

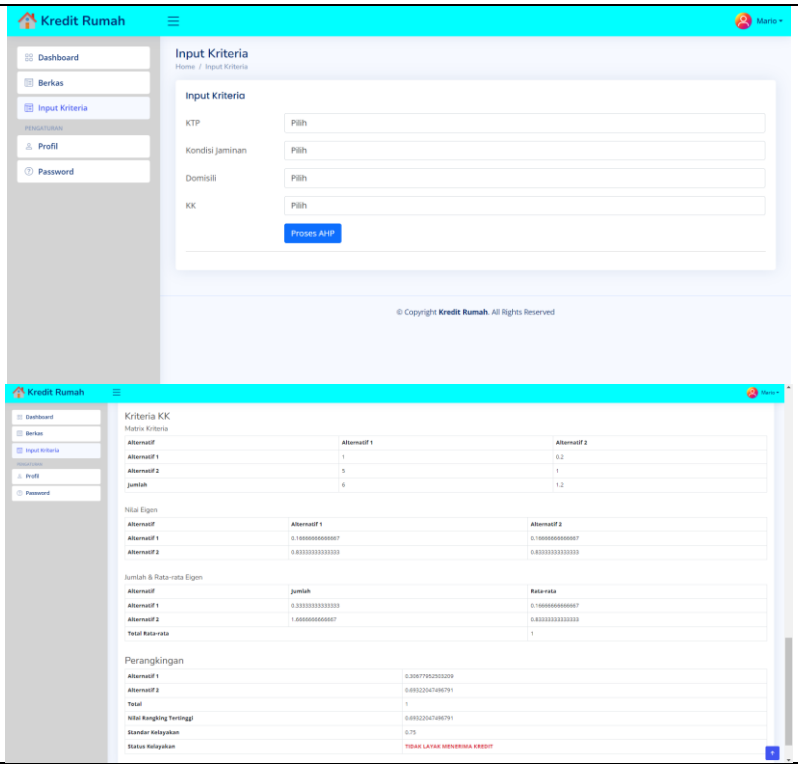
| Tes Faktor | Hasil | Keterangan |
|---|---|---|
| Mengupload berkas kelengkapan sesuai yang ada di form | ✓ | Berhasil, ketika berkas yang diupload berhasil melalui proses validasi |
| ***Screenshot*** | | |
|  | | |

| Tes Faktor | Hasil | Keterangan |
|---|---|---|
| Mengupload berkas kelengkapan sesuai yang ada di form | ✓ | Berhasil, ketika berkas yang diupload berhasil melalui proses validasi |
| ***Screenshot*** | | |
|  | | |

k. Black Box Testing Input Kriteria

**Tabel 4. 18** *Black Box Testing Input Kriteria*

| Tes Faktor | Hasil | Keterangan |
|---|---|---|
| Memasukkan kriteria sesuai yang ada di form | ✓ | Berhasil, akan tampil semua informasi tentang kelayakan dalam mengambil kredit |
| ***Screenshot*** | | |

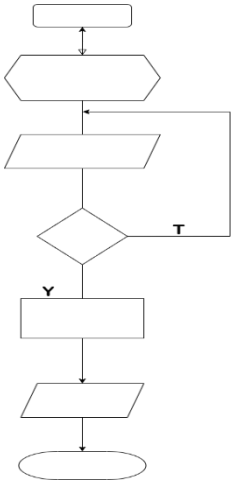



## 2. White Box

Pada metode pengujian ini akan ditampilkan flowchart dan flowgraph dari sistem yang telah dibuat. Berikut dibawah ini merupakan hasil pengujian menggunakan metode white box testing.

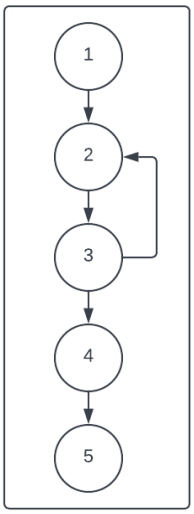
a. White Box Testing Login Admin

1) Flowchart




**Gambar 4. 35** *Flowchart Login Admin*

2) Flowgraph




**Gambar 4. 36** *Flowgraph Login Admin*

Berdasarkan gambar 4. 26 diatas dilakukan perhitungan sebagai berikut :

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

Pada rumus : V(G) = E – N + 2

E (*edge*) = 5

N (*node*) = 5

P (Predikat *node*) = 1

Penyelesaian :

V(G) = E – N + 2

= 5 – 5 + 2

= 2

Predikat (P) = P + 1

= 1 + 1

= 2

(2) Berdasarkan perhitungan *Cyclomatic Complexcity* dari *flowgraph* di atas memiliki *Region* = 2

(3) *Independent path* pada *flowgraph* tersebut yakni:

Path 1 = 1 – 2 – 3 – 2

Path 2 = 1 – 2 – 3 – 4 – 5

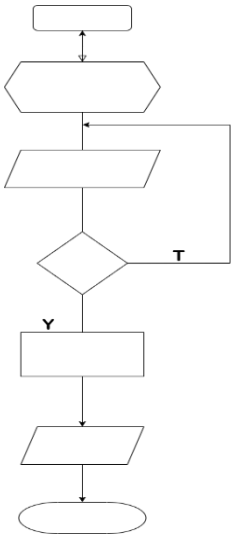
(4) Grafik matriks Login Admin

**Tabel 4. 19** *Grafik Matriks Login Admin*

| | 1 | 2 | 3 | 4 | 5 | E-1 |
|---|---|---|---|---|---|---|
| 1 | | 1 | | | | 1 - 1 = 0 |
| 2 | | | 1 | | | 1 – 1 = 0 |
| 3 | | 1 | | 1 | | 2 – 1 = 1 |
| 4 | | | | | 1 | 1 – 1 = 0 |
| 5 | | | | | | 0 |
| | SUM (E + 1) | | | | | 1 + 1 = 2 |

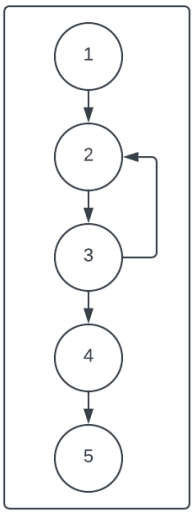
b. White Box Testing Daftar Nasabah

1) Flowchart




**Gambar 4. 37** *Flowchart Daftar Nasabah*

2) Flowgraph




**Gambar 4. 38** *Flowgraph Daftar Nasabah*

Berdasarkan gambar 4. 28 diatas dilakukan perhitungan sebagai berikut :

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

Pada rumus : V(G) = E – N + 2

E (*edge*) = 5

N (*node*) = 5

P (Predikat *node*) = 1

Penyelesaian :

V(G) = E – N + 2

= 5 – 5 + 2

= 2

Predikat (P) = P + 1

= 1 + 1

= 2

(2) Berdasarkan perhitungan *Cyclomatic Complexcity* dari *flowgraph* di atas memiliki *Region* = 2

(3) *Independent path* pada *flowgraph* tersebut yakni:

Path 1 = 1 – 2 – 3 – 2

Path 2 = 1 – 2 – 3 – 4 – 5

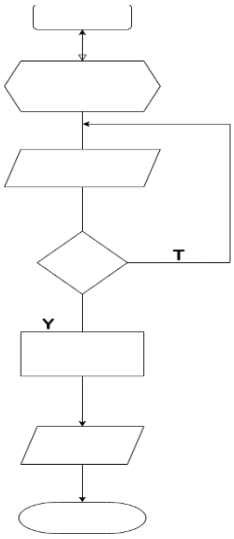
(4) Grafik matriks Daftar Nasabah

**Tabel 4. 20** *Grafik Matriks Daftar Nasabah*

| | 1 | 2 | 3 | 4 | 5 | E-1 |
|---|---|---|---|---|---|---|
| 1 | | 1 | | | | 1 - 1 = 0 |
| 2 | | | 1 | | | 1 – 1 = 0 |
| 3 | | 1 | | 1 | | 2 – 1 = 1 |
| 4 | | | | | 1 | 1 – 1 = 0 |
| 5 | | | | | | 0 |
| | SUM (E + 1) | | | | | 1 + 1 = 2 |

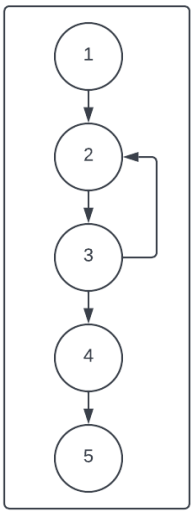
c. White Box Testing Tambah Kriteria

1) Flowchart




**Gambar 4. 39** *Flowchart Tambah Kriteria*

2) Flowgraph




**Gambar 4. 40** *Flowgraph Tambah Kriteria*

Berdasarkan gambar 4. 30 diatas dilakukan perhitungan sebagai berikut :

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

Pada rumus : V(G) = E – N + 2

E (*edge*) = 5

N (*node*) = 5

P (Predikat *node*) = 1

Penyelesaian :

V(G) = E – N + 2

= 5 – 5 + 2

= 2

Predikat (P) = P + 1

= 1 + 1

= 2

(2) Berdasarkan perhitungan *Cyclomatic Complexcity* dari *flowgraph* di atas memiliki *Region* = 2

(3) *Independent path* pada *flowgraph* tersebut yakni:

Path 1 = 1 – 2 – 3 – 2

Path 2 = 1 – 2 – 3 – 4 – 5

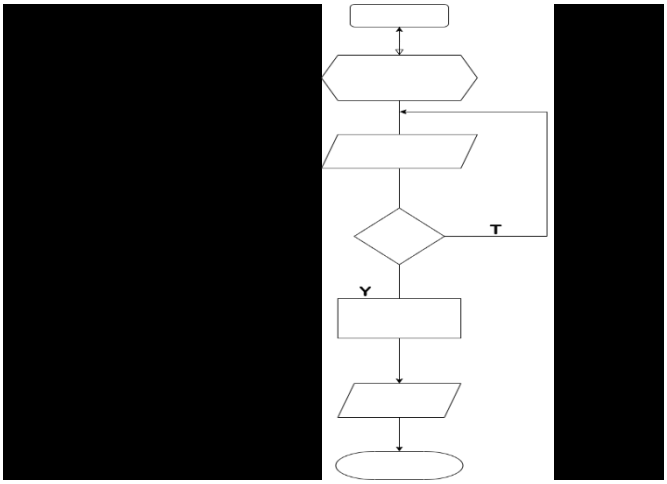
(4) Grafik matriks Tambah Kriteria

**Tabel 4. 21** *Grafik Matriks Tambah Kriteria*

| | 1 | 2 | 3 | 4 | 5 | E-1 |
|---|---|---|---|---|---|---|
| 1 | | 1 | | | | 1 - 1 = 0 |
| 2 | | | 1 | | | 1 – 1 = 0 |
| 3 | | 1 | | 1 | | 2 – 1 = 1 |
| 4 | | | | | 1 | 1 – 1 = 0 |
| 5 | | | | | | 0 |
| | SUM (E + 1) | | | | | 1 + 1 = 2 |

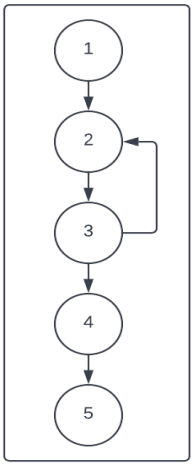
d. White Box Testing Standar Kelayakan

1) Flowchart




**Gambar 4. 41** *Flowchart Standar Kelayakan*

2) Flowgraph




**Gambar 4. 42** *Flowgraph Standar Kelayakan*

Berdasarkan gambar 4. 32 diatas dilakukan perhitungan sebagai berikut :

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

Pada rumus : V(G) = E – N + 2

E (*edge*) = 5

N (*node*) = 5

P (Predikat *node*) = 1

Penyelesaian :

V(G) = E – N + 2

= 5 – 5 + 2

= 2

Predikat (P) = P + 1

= 1 + 1

= 2

(2) Berdasarkan perhitungan *Cyclomatic Complexcity* dari *flowgraph* di atas memiliki *Region* = 2

(3) *Independent path* pada *flowgraph* tersebut yakni:

Path 1 = 1 – 2 – 3 – 2

Path 2 = 1 – 2 – 3 – 4 – 5

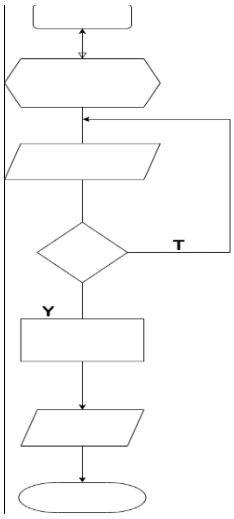
(4) Grafik matriks Standar Kelayakan

**Tabel 4. 22** *Grafik Matriks Standar Kelayakan*

| | 1 | 2 | 3 | 4 | 5 | E-1 |
|---|---|---|---|---|---|---|
| 1 | | 1 | | | | 1 - 1 = 0 |
| 2 | | | 1 | | | 1 – 1 = 0 |
| 3 | | 1 | | 1 | | 2 – 1 = 1 |
| 4 | | | | | 1 | 1 – 1 = 0 |
| 5 | | | | | | 0 |
| | SUM (E + 1) | | | | | 1 + 1 = 2 |

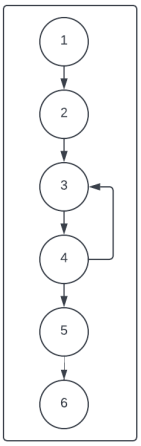
e. White Box TestingVerifikasi Berkas

1) Flowchart




**Gambar 4. 43** *Flowchart Verifikasi Berkas*

2) Flowgraph




**Gambar 4. 44** *Flowgraph Verifikasi Berkas*

Berdasarkan gambar 4. 34 diatas dilakukan perhitungan sebagai berikut :

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

Pada rumus : V(G) = E – N + 2

E (*edge*) = 6

N (*node*) = 6

P (Predikat *node*) = 1

Penyelesaian :

V(G) = E – N + 2

= 6 – 6 + 2

= 2

Predikat (P) = P + 1

= 1 + 1

= 2

(2) Berdasarkan perhitungan *Cyclomatic Complexcity* dari *flowgraph* di atas memiliki *Region* = 2

(3) *Independent path* pada *flowgraph* tersebut yakni:

Path 1 = 1 – 2 – 3 – 4 – 3

Path 2 = 1 – 2 – 3 – 4 – 5 – 6

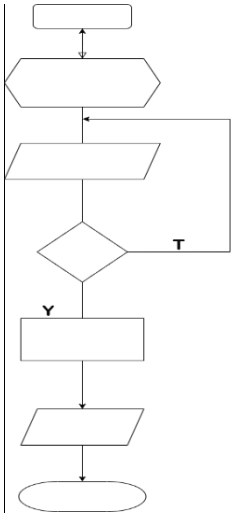
(4) Grafik matriks Verifikasi Berkas

**Tabel 4. 23** *Grafik Matriks Verifikasi Berkas*

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

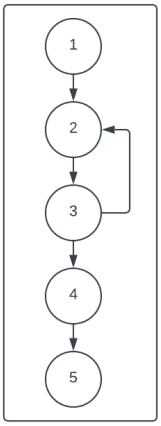
f. White Box Testing Upload Berkas

1) Flowchart




**Gambar 4. 45** *Flowchart Upload Berkas*

2) Flowgraph




**Gambar 4. 46** *Flowgraph Upload Berkas*

Berdasarkan gambar 4. 36 diatas dilakukan perhitungan sebagai berikut :

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

Pada rumus : V(G) = E – N + 2

E (*edge*) = 5

N (*node*) = 5

P (Predikat *node*) = 1

Penyelesaian :

V(G) = E – N + 2

= 5 – 5 + 2

= 2

Predikat (P) = P + 1

= 1 + 1

= 2

(2) Berdasarkan perhitungan *Cyclomatic Complexcity* dari *flowgraph* di atas memiliki *Region* = 2

(3) *Independent path* pada *flowgraph* tersebut yakni:

Path 1 = 1 – 2 – 3 – 2

Path 2 = 1 – 2 – 3 – 4 – 5

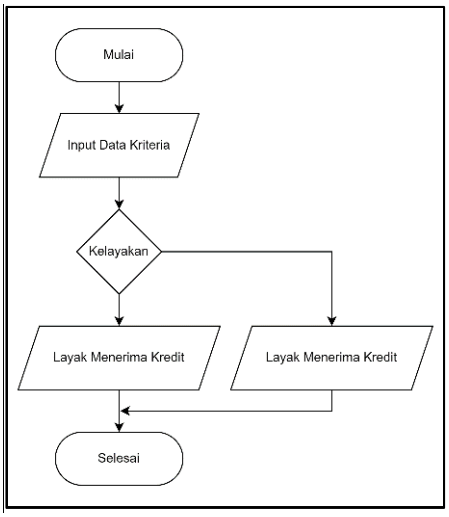
(4) Grafik matriks Upload Berkas

**Tabel 4. 24** *Grafik Matriks Upload Berkas*

| | 1 | 2 | 3 | 4 | 5 | E-1 |
|---|---|---|---|---|---|---|
| 1 | | 1 | | | | 1 - 1 = 0 |
| 2 | | | 1 | | | 1 – 1 = 0 |
| 3 | | 1 | | 1 | | 2 – 1 = 1 |
| 4 | | | | | 1 | 1 – 1 = 0 |
| 5 | | | | | | 0 |
| | SUM (E + 1) | | | | | 1 + 1 = 2 |

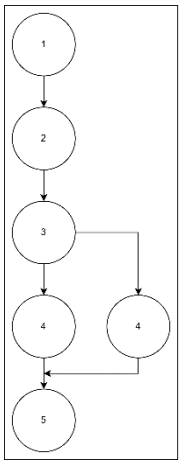
g. White Box Testing Input Kriteria

1) Flowchart




**Gambar 4. 47** *Flowchart Input Kriteria*

2) Flowgraph




**Gambar 4. 48** *Flowgraph Input Kriteria*

Berdasarkan gambar 4. 38 diatas dilakukan perhitungan sebagai berikut :

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

Pada rumus : V(G) = E – N + 2

E (*edge*) = 6

N (*node*) = 6

P (Predikat *node*) = 2

Penyelesaian :

V(G) = E – N + 2

= 6 – 6 + 2

= 2

Predikat (P) = P + 1

= 2 + 1

= 3

(2) Berdasarkan perhitungan *Cyclomatic Complexcity* dari *flowgraph* di atas memiliki *Region* = 2

(3) *Independent path* pada *flowgraph* tersebut yakni:

Path 1 = 1 – 2 – 3 – 4 – 5

Path 2 = 1 – 2 – 3 – 4 – 5

(4) Grafik matriks Input Kriteria

**Tabel 4. 25** *Grafik Matriks Input Kriteria*

| | 1 | 2 | 3 | 4 | 5 | E-1 |
|---|---|---|---|---|---|---|
| 1 | | 1 | | | | 1 - 1 = 0 |
| 2 | | | 1 | | | 1 – 1 = 0 |
| 3 | | | | 1 | | 1 – 1 = 0 |
| 4 | | | | | 1 | 1 – 1 = 0 |
| 5 | | | | | | 0 |
| | SUM (E + 1) | | | | | 0 + 1 = 1 |

# BAB V

# PENUTUP

## A. Kesimpulan

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

1. Proses penilaian kelayakan pemberian kredit rumah melalui system penunjang Keputusan menggunakan metode *Analytical Hierarchy Process* (AHP) yang dapat memperbaiki proses internal developer perumahan dan memberikan media alternatif yang dapat memudahkan dalam pengambilan kredit rumah..

## B. Saran

Saran untuk pengembangan sistem lebih lanjut sebagai bahan masukan agar perancangan aplikasi ini dapat berkembang sesuai dengan perkembangan teknologi, dengan menambah fitur – fitur yang dapat membuat sistem lebih baik sehingga lebih memudahkan pengguna, memperbaiki tampilan user interface-nya dan membuat sistem android agar compitable digunakan dimana saja.

# DAFTAR PUSTAKA


Aisyah, S. (2019). *Apliaksi Sistem Pendukung Keputusan Analisis Kelayakan Pemberian Kredit Menggunakan Metode SAW Pada Perusahaan Leasing. Jurnal Teknovasi.*

Andharsaputri, R. L. (2020). *Sistem Pendukung Keputusan Pemberian Kredit Rumah Pemilikan Rumah Di Koperasi. Jurnal Teknologi Informasi Mura.*

Budi Fachrizal, I. F. (2013). *Sistem Pendukung Keputusan Untuk Kredit Kepemilikan Rumah Bank UOB Menggunakan Metode Simple Additive Weighthing. Jurnal Informatika Mulawarman.*

Nofri Wandi Al-Hafiz, M. (2017). *Sistem Pendukung Keputusan Penentukan Kredit Kepemilikan Rumah Menerapkan Multi-Objective Optimization On The Basis Of Ratio Analysis(MOORA). KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer).*

Nugroho, A. (2012). *Pengertian Flowchart dan Jenis-Jenisnya*.

Sapta Adi Permana, B. W. (n.d.). *Sistem Pendukung Keputusan Berbasis Fuzzy Analytical Hierarchy Process Untuk Kelayakan Kredit Rumah* .

Saputra, H. d. (2014). *Pemograman Web.*

Tarmizi, H. (2017). *"Pengertian Sublime Text". Retrieved from https://hasantarmizi.blogspot.com/2017/04/pengertiansublimetext.html:*

Turban, E. A. (2005). *Decision Support System and Intelligent System (Sistem Pendukung keputusan dan Sistem Cerdas).*

Yohanes Suhari, M. S. (2009). *Sistem Pendukung Keputusan Pemberian Kredit Rumah Pada PT. BPR Artamanunggal Abadi Mragen. DINAMIKA INFORMATIKA.*

Al Fatta, H. (2007). Andi Offset. *Analisis Dan Perancangan Sistem Informasi. Hibatullah Fauzan Takalamingan, D. P. (2018). Analisi Penerapan Sistem Dan Prosedur Pemberian Kredit Kepemilikan Rumah (KPR) Subsidi Pada Bank Tabungan Negara Cabang Manado. Jurnal Riset Akuntansi Going Concern.*