

BAB I

PENDAHULUAN

A. Latar Belakang

Kemajuan Sistem Bantuan pemerintah merupakan bentuk tanggung jawab kepada masyarakat bantuan terbagi atas beberapa jenis dan dipermudah diolah otoritas Desa masing masing kondisi seperti itu tidak lepas dari upaya yang telah dilakukan oleh pemerintah untuk berusaha meningkatkan tingkat kesejahteraan masyarakat. Seperti masyarakat di Desa Maritengngae yang mayoritas penduduknya berprofesi sebagai nelayan, baik melalui pemberian bantuan peralatan tangkap, kemudahan akses permodalan, maupun melalui program pemberdayaan masyarakat pesisir. Dimana semua program tersebut bertujuan untuk meningkatkan kesejahteraan masyarakat pesisir khususnya di Maritengngae.

Desa adalah kesatuan masyarakat hukum yang memiliki batas batas wilayah yang berwenang untuk mengatur dan menadmins urusan pemerintahan kepentingan masyarakat setempat berdasarkan prakarsa masyarakat hak asal usul dan atau/ atau hak tradisional yang diakui dan dihormati dalam system pemerintahan Negara kesatuan republic Indonesia (Perda Nomor 6 Tahun 2015)

Dengan adanya aplikasi Sistem Informasi berbasis Web, diharapkan pengguna dapat mengakses secara mandiri, fleksibel dan efisien. Aplikasi ini akan menyediakan Informasi bantuan pemerintah yang masuk ke Desa. Selain itu,

Aplikasi ini juga akan dilengkapi dengan fitur profile Desa serta informasi mengenai bantuan serta kriteria yang berhak menerima bantuan tersebut.

Masalah yang timbul ini belum ada sistem yang membantu dalam pengelolaan bantuan desa dan masih melakukan dengan cara manual, sehingga tidak tersajinya informasi, pengarsipan dan menyajikan laporan bantuan kepada warga

Pendekatan berorientasi objek terhadap pengembangan sistem yang mencakup suatu metode pengembangan serta perangkat lunak, dengan tujuan mempersingkat waktu yang biasanya diperlukan dalam siklus hidup pengembangan tradisional antara perancangan dan penerapan suatu sistem informasi. Hasil pengujian blackbox adalah sistem informasi pengelolaan bantuan desa berbasis Web ini sudah berfungsi dengan baik karena di buktikannya dengan black-box testing yang menyatakan valid atau berhasil, maka dapat menunjukkan bahwa sistem tersebut sudah dapat diterima kelayakannya untuk di gunakan di desa Maritengngae.

Awal dari penulis ingin mengambil ini adalah karena melihat secara langsung permasalahan yang terjadi di desa Maritengngae kabupaten pinrang dimana di setiap pembagian bantuan yang terjadi yaitu selalu terjadinya permasalahan entah itu dari staff nya tersendiri ataupun masalah dari hubungan internal warga. Berdasarkan pemaparan diatas dan dengan dasar Undang-Undang yang ada yaitu:

Pasal 8 Berdasarkan hasil penghitungan alokasi Dana Desa setiap Desa sebagaimana dimaksud dalam Pasal 4 sampai dengan Pasal 7, Pemerintah menetapkan rincian Dana Desa setiap Desa tahun anggaran 2024. Pasal 9 Sumber data dalam pengalokasian Dana Desa sebagaimana dimaksud dalam Pasal 4 sampai dengan Pasal 7, sebagai berikut: a. data jumlah Desa, data nama dan kode Desa, dan data jumlah penduduk bersumber dari

B. Rumusan Masalah

Berdasarkan latar belakang diatas, Adapun rumusan masalah yang akan dibahas penulis yaitu:

1. Bagaimana cara merancang dan membuat website sitem Informasi bantuan pemerintah (BLT) di Desa Maritengngae?

C. Batasan Masalah

Berdasarkan rumusan masalah di atas, maka batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Aplikasi ini hanya dirancang pada website dengan menggunakan bahasa pemrograman PHP, MYSQL.
2. Sistem ini hanya akan mencakup data penerima bantuan BLT dan BNPT identitas penerima

D. Tujuan Penelitian

Tujuan dari penelitian ini adalah Sebagai system Informasi atau wadah dalam penyampain serta database untuk kantor Desa Maritengngae

E. Manfaat Penelitian

Adapun manfaat yang dapat diperoleh dari penelitian ini adalah:

1. Bagi penulis

Penulis dapat mengaplikasikan ilmu dan teori yang telah didapatkan di bangku perkuliahan,serta dapat menambah wawasan tentang aplikasi Sistem Informasi bantuan pemerintah

2. Bagi pembaca

Penelitian ini diharapkan dapat memberikan informasi secara tertulis maupun sebagai referensi mengenai teknologi system informasi bantuan pemerintah berbasis Web.

3. Bagi pengguna

Pengguna dapat mengakses informasi bantuan pemerintah berbasis Web di rumah masing masing.

BAB II

TINJAUAN PUSTAKA

A. Kajian Teori

1. Pengembangan Sistem Informasi

Pengembangan sistem informasi adalah proses pencarian solusi atau pemecahan dari suatu masalah baik secara terstruktur maupun berorientasi objek. Pengembangan secara terstruktur biasanya lebih menekankan pembuatan system berdasarkan proses kerja/prosedur yang telah ditetapkan. Sedangkan pengembangan sistem berorientasi objek lebih menekankan pembuatan sistem terhadap peranan objek yang terlibat dalam sistem tersebut. Pengembangan sistem informasi secara terstruktur terdiri dari beberapa kegiatan / tahapan (*phased*), yaitu tahap analisis sistem (*analyst*), konstruksi sistem (*construction*), pengkodean (*coding*), uji sistem (*testing*), dan tahap pemeliharaan system (*maintenance*). Sedangkan pengembangan sistem informasi berorientasi objek terdiri dari tahap analysis (*Inception*), design (*elaboration*), konstruksi (*construction*) dan penggantian sistem (*Transition*). (Arbani, 2022),

2. Sistem

Definisi sistem telah banyak dikemukakan oleh para ahli dari berbagai disiplin ilmu. Walaupun definisi sistem begitu banyak namun system dapat diartikan dengan mengacu pada pendapat para ahli.

(Susanti, 2020) sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Suatu sistem dibangun

bukan dinilai dari segi tampilan semata, namun dinilai dari pola aliran informasi dan implementasi. Faktor-faktor yang harus dipertimbangkan dalam pembangunan sistem, yaitu:

- 1) Efisiensi dan efektifitas.

Pola sistem yang dibangun harus sederhana, lengkap dan akurat.

- 2) Prosedur yang singkat. Sistem yang dirancang tidak menjemukan pada saat melakukan input data.

- 3) Sistem yang optimal. Sistem dapat bekerja secara optimal dalam pemanfaatan sumber daya yang dimiliki oleh perusahaan.

- 4) Tren di masa depan. Sistem yang dibangun mampu beradaptasi dengan perkembangan jaman.

- 5) Efisiensi dalam biaya. Sistem dibangun berdasarkan pada perencanaan dan perancangan secara matang guna menghemat biaya.

- 6) Integritas dan keamanan data. Sistem yang baru dibangun harus memenuhi standar integritas dan keamanan data.

- 7) Interaktif. Sistem yang dibangun harus mudah dipahami dengan cara mengidentifikasi, menyeleksi dan menyusun komponen-komponen yang sesuai dengan latar belakang pemakaiannya.

3. Informasi

Dalam hal ini penulis akan mendeskripsikan definisi dari Informasi menurut pendapat ahli yang memiliki gagasan yang sama yaitu :

Informasi adalah data yang telah diorganisasi dan telah memiliki kegunaan dan manfaat (Dewi Maharani, 2019).

Informasi adalah data yang telah dikelola dan diproses untuk memberikan arti dan memperbaiki proses pengambilan keputusan, sebagaimana perannya, pengguna membuat keputusan yang lebih baik sebagai kuantitas dan kualitas dari peningkatan informasi (Retno sari, 2020).

Dari beberapa pengertian informasi menurut pendapat ahli diatas dapat penulis simpulkan, bahwa informasi adalah data mentah yang telah diolah sedemikian rupa sehingga menghasilkan sesuatu yang bermakna bagi penggunanya dalam mengambil sebuah keputusan.

4. Sistem Informasi Bantuan Pemerintah

Kementerian Koordinator Bidang Pembangunan Manusia dan Kebudayaan (Kemenko PMK) sesuai dengan Perpres No. 9 Tahun 2015, tentang Kemenko PMK bertanggung jawab untuk melakukan koordinasi, sinkronisasi, dan pengendalian urusan pembangunan manusia dan kebudayaan. Urusan ini salah satunya menjangkau program kesejahteraan rakyat, melalui pemberian bantuan sosial pada masyarakat. Bantuan ini diberikan untuk memenuhi dan menjamin kebutuhan dasar serta meningkatkan taraf hidup penerima bansos.

Sistem Informasi merupakan suatu sistem yang tujuannya menghasilkan informasi. Informasi adalah data yang diolah menjadi bentuk yang berguna bagi para pemakainya. Data yang diolah saja belum cukup menjadi informasi. Untuk menjadi suatu informasi, data yang diolah tersebut harus berguna bagi penggunanya. Pendapat lain mengatakan bahwa informasi

merupakan proses lebih lanjut. Dari data yang sudah memiliki nilai tambah. Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi berguna bagi yang menerimanya (Sangga Rasefta, 2020).

Fungsi ini juga sejalan dengan amanat dalam Inpres Nomor 7 Tahun 2014 tentang Pelaksanaan Program Simpanan Keluarga Sejahtera, Program Indonesia Pintar, dan Program Indonesia Sehat untuk Membangun Keluarga Produktif. Menko PMK dalam hal ini dapat mengambil langkah-langkah yang diperlukan sesuai tugas, fungsi, dan kewenangannya untuk meningkatkan efektivitas dan efisiensi pelaksanaan program Simpanan

Keluarga Sejahtera, Program Indonesia Pintar, dan Program Indonesia Sehat bagi keluarga kurang mampu dengan melibatkan seluruh unsur masyarakat dan dunia usaha. Diantaranya dengan meningkatkan koordinasi pelaksanaan dan pengawasan, penanganan pengaduan masyarakat, dan meningkatkan koordinasi dan evaluasi pelaksanaan program.

BLT Dana Desa 2024 adalah program pemberian bantuan berupa dana tunai kepada keluarga penerima manfaat (KPM) yang tinggal di desa. Program ini menggunakan Dana Desa, yaitu bagian dari transfer ke daerah yang diperuntukkan bagi desa untuk mendukung penyelenggaraan pemerintahan, pembangunan, pemberdayaan masyarakat, dan kemasyarakatan. Penyaluran BLT DD kepada Keluarga Penerima Manfaat (KPM) bertujuan agar dana

desa dirasakan oleh masyarakat yang kurang mampu, khususnya untuk membantu meningkatkan ekonomi masyarakat. Diharapkan segala macam bentuk bantuan yang disampaikan kepada masyarakat betul-betul tepat sasaran dan tepat guna, benar-benar bisa meningkatkan kesejahteraan masyarakat penerima bantuan. Untuk besaran dana BLT DD sebesar Rp.300.000,00 per bulan dan pada tahap 1 ini total sebesar Rp. 1.800.000,00 (Januari - Juni). Besaran dana yang diberikan dan mekanisme yang dijalankan dalam program BLT berbeda-beda tergantung kebijakan pemerintah di negara tersebut. dan jumlah anggaran yang dikeluarkan untuk program ini adalah 3,8 triliun rupiah untuk 18,5 juta keluarga miskin, dengan uang tunai 100 ribu rupiah per bulannya. Tahapan pelaksanaan program bantuan langsung tunai di Indonesia umumnya dimulai dari sosialisasi, verifikasi data nama nominasi Rumah Tangga Sasaran (RTS) yang akan diberikan bantuan, pembagian kartu BLT, pencairan dana, dan terakhir pembuatan laporan dan evaluasi.

5. Web

Website merupakan fasilitas internet yang menghubungkan dokumen dalam lingkup lokal maupun jarak jauh. Dokumen pada website disebut dengan web page dan link dalam website memungkinkan pengguna bisa berpindah dari satu page ke page lain (*hypertext*), baik antara page yang disimpan dalam server yang sama maupun server di seluruh dunia. Halaman

dapat diakses dan dibaca melalui browser seperti *Google Chrome*, *Mozilla Firefox*, dan lainnya.

Web berasal dari kata Bahasa Inggris yang bila diterjemahkan dalam Bahasa Indonesia berarti “Jaring Laba-Laba”. Hampir sama dengan arti dari kata web itu sendiri, web telah membentang ke seluruh penjuru dunia. Tidak hanya terbatas pada lembaga-lembaga penelitian yang ingin memublikasikan hasil riset, tetapi juga telah banyak digunakan oleh perusahaan bisnis yang ingin mengiklankan produk atau untuk melakukan transaksi bisnisnya (Chandra Kesuma, 2020)

6. *HTML*



Gambar 2. 1 *Hypertext Markup Language*

Hypertext Markup Language atau *HTML*, adalah bahasa *markup* standar industri untuk membangun halaman *web*. Ini menawarkan kumpulan *tag* dan properti yang menentukan pengorganisasian halaman *web*, konten, dan format halaman *web*. Untuk merepresentasikan elemen pada halaman *web*, *HTML* menggunakan struktur hierarkis yang dikenal sebagai *Document Object Model (DOM)*. Teks, foto, tautan, multimedia, dan elemen lainnya

semuanya dapat dimuat dalam setiap elemen, yang dibatasi oleh *tag* pembuka dan penutup.

Istilah "*HyperText*" mengacu pada kemampuan *HTML* untuk membuat tautan yang memungkinkan pengguna dengan mudah dan cepat berpindah dari satu situs *web* ke situs *web* lainnya (Duckett & Wiley, 2019). *HyperText Markup Language*, juga dikenal sebagai *HTML*, adalah bahasa global *World Wide Web*. Seperti yang tersirat dari namanya *HyperText Markup Language*, *HTML* adalah bahasa *markup* bukan bahasa pemrograman (Hartl & Donahoe, 2020).

Dokumen yang dibuat dengan menggunakan bahasa *markup* seperti *HTML* memiliki bagian yang dirender pada hasil akhir serta bagian yang menginstruksikan program rendering cara menginterpretasikan teks yang tersisa. Proses mengubah dokumen teks yang berisi teks *HTML* menjadi, misalnya, representasi visualnya di layar disebut sebagai "*rendering*" (Nematian, 2020).

Perkembangan *HTML* sangat pesat. Dari permulaan *HTML* 1.0 yang sederhana, yang mencakup elemen dasar seperti judul dan paragraf, hingga penambahan tabel dan gambar pada *HTML* 2.0, dan akhirnya ke *frame* dan dukungan *form* yang disempurnakan pada *HTML* 3.2. Dengan *CSS*, konten dan presentasi sekarang dapat dipisahkan dalam *HTML* 4.01, yang juga meningkatkan komponennya. Dengan elemen semantiknya, dukungan multimedia, bentuk yang lebih baik, fitur aksesibilitas, dan kemampuan *mobile-friendly*, *HTML5* menandai periode transformasi.

Versi *HTML* terbaru, yang dikenal sebagai *HTML5*, digunakan untuk membuat dan mengatur dokumen di internet. Ini mencakup elemen dan fitur baru yang memungkinkan pembuatan halaman *web* yang lebih dinamis dan menarik, termasuk dukungan untuk audio dan video, kanvas untuk grafik dan animasi, dan peningkatan kompatibilitas dengan perangkat seluler (McLaughlin, 2019).

Fungsionalitas baru yang diperkenalkan oleh *HTML5* memungkinkan pembuatan halaman *web* yang lebih dinamis dan interaktif, yaitu:

- a. Dukungan audio dan video.
- b. Kanvas visual dan animasi.
- c. Dukungan yang ditingkatkan untuk *mobile*..
- d. *Semantic Element* baru termasuk *nav*, *article*, *sections* dan *headers*.
- e. Kontrol untuk form, yaitu tanggal, waktu, email, pencarian, dll.
- f. Penyimpanan *web* untuk menyimpan informasi secara lokal.
- g. Fitur *drag* and *drop*.

7. *PHP*



Gambar 2.2 Logo PHP

PHP adalah bahasa *server-side* yang secara khusus digunakan untuk aplikasi *web*. Bahasa *PHP* berjalan di sisi *server*, artinya ia berjalan pada *server* dan bisa dimasukkan di antara bahasa *HTML* (*HyperText Markup*

Language). Oleh karena itu, kode *PHP* tidak akan terlihat lagi dan hasil akhir dalam bentuk *HTML* akan dikirimkan ke *browser* (Sutarman, 2020). Karena *PHP* dijalankan di komputer *server*, itu dikenal sebagai bahasa pemrograman sisi *server*. Ini berbeda dengan bahasa pemrograman sisi klien, seperti JavaScript, yang beroperasi pada klien, yaitu *web browser*.

Pada awalnya, *PHP* adalah singkatan dari *Personal Home Page*. *PHP* digunakan untuk mengembangkan situs *web* pribadi, sesuai dengan namanya. Seiring berjalannya waktu, *PHP* berkembang menjadi bahasa pemrograman *web* yang sangat kuat, tidak hanya digunakan untuk membuat halaman *web* dasar tetapi juga untuk situs-situs populer seperti *Wikipedia*, *WordPress*, *Joomla*, dan lainnya yang dikunjungi oleh jutaan orang.

8. CSS

CSS adalah bahasa yang terus berkembang untuk menggambarkan presentasi konten *web* di layar, printer, sintesis ucapan, pembaca layar, dan jendela obrolan. CSS digunakan oleh semua *browser* pada semua ukuran layar di semua jenis perangkat *IoT*, termasuk ponsel, komputer, video game, televisi, jam tangan, dan konsol otomatis (Meyer & Weyl, 2020). Sedangkan menurut (Duckett, 2022) CSS adalah bahasa pemrograman *web* yang digunakan untuk mengendalikan tampilan halaman *web*. (Castro, 2022) menjelaskan bahwa CSS memiliki berbagai properti yang dapat digunakan untuk mengatur tampilan halaman *web*. CSS terdiri dari dua bagian utama, yaitu deklarasi dan selektor. Deklarasi adalah bagian yang menentukan properti dari suatu halaman *web*. Selektor adalah bagian yang menentukan

elemen *web* mana yang akan diatur oleh deklarasi. CSS juga memiliki berbagai properti yang dapat digunakan untuk mengatur tampilan halaman *web*. Beberapa properti CSS yang umum digunakan antara lain:

- a. Ukuran: properti ini digunakan untuk mengatur ukuran halaman *web*, seperti ukuran font, ukuran gambar, dan ukuran jarak antar elemen.
- b. Warna: properti ini digunakan untuk mengatur warna halaman *web*, seperti warna font, warna latar belakang, dan warna *border*.
- c. Font: properti ini digunakan untuk mengatur font elemen *web*, seperti jenis font, ukuran font, dan gaya font.
- d. Tata letak: properti ini digunakan untuk mengatur tata letak elemen *web*, seperti posisi elemen, ukuran elemen, dan jarak antar elemen.

9. *Unified Modelling Language*

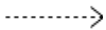
UML adalah serangkaian alat yang digunakan untuk menyederhanakan suatu sistem atau perangkat lunak berbasis objek. Singkatan dari *Unified Modelling Language*, *UML* juga berperan dalam mempermudah pengembangan aplikasi secara berkelanjutan. Sistem atau aplikasi yang tidak didokumentasikan dengan baik seringkali menghambat proses pengembangan karena memaksa para pengembang untuk melakukan penyelidikan dan memahami kode program yang kompleks. *UML* juga berfungsi sebagai alat untuk men-*transfer* pengetahuan tentang suatu sistem atau aplikasi dari satu pengembang ke pengembang lainnya. Keberadaan

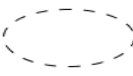
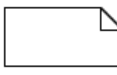
UML tidak hanya mempermudah komunikasi antara para pengembang, tetapi juga memungkinkan orang di luar lingkaran pengembangan, seperti pihak bisnis atau siapapun, untuk memahami suatu sistem dengan lebih baik.

UML (*Unified Modelling Language*), sebagai standar industri, digunakan untuk menggambarkan, merencanakan, dan mendokumentasikan sistem perangkat lunak. *UML* menyediakan format standar untuk merancang model sistem. Namun, karena *UML* memanfaatkan kelas dan operasi sebagai bagian dari konsep intinya, lebih sesuai digunakan untuk pengembangan perangkat lunak dengan bahasa berbasis objek seperti C++, Java, C#, atau VB.NET. Meskipun begitu, *UML* masih bisa digunakan untuk memodelkan aplikasi prosedural dalam bahasa seperti VB atau C.

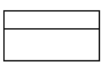
UML mulai dikembangkan oleh *Object Management Group* sejak versi 1.0 dirilis pada bulan Januari 1997. Dalam pengembangan berbasis objek, terdapat beberapa prinsip kunci yang harus dipahami, seperti Objek, Kelas, Abstraksi, Enkapsulasi, Pewarisan, dan Polimorfisme. Adapun daftar simbol *UML* yaitu:

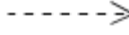
Tabel 2. 1 *Symbol Use Case Diagram*

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

NO.	GAMBAR	NAMA	KETERANGAN
			yang tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
5		<i>Extend</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
6		<i>Association</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
7		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
8		<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>).
9		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Tabel 2. 2 *Symbol Class Diagram*

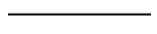
NO.	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang

NO.	GAMBAR	NAMA	KETERANGAN
			menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i>
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<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
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

Tabel 2. 3 *Symbol Sequence Diagram*

No	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi

Tabel 2. 4 *Symbol State Chart Diagram*

No	GAMBAR	NAMA	KETERANGAN
1		<i>State</i>	Nilai atribut dan nilai <i>Link</i> pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		<i>Initial Pseudo State</i>	Bagaimana objek dibentuk atau diawali
3		<i>Final State</i>	Bagaimana objek dibentuk dan dihancurkan
4		<i>Transition</i>	Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya
5		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

No	GAMBAR	NAMA	KETERANGAN
6		<i>Node</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Tabel 2. 5 *Symbol Activity Diagram*

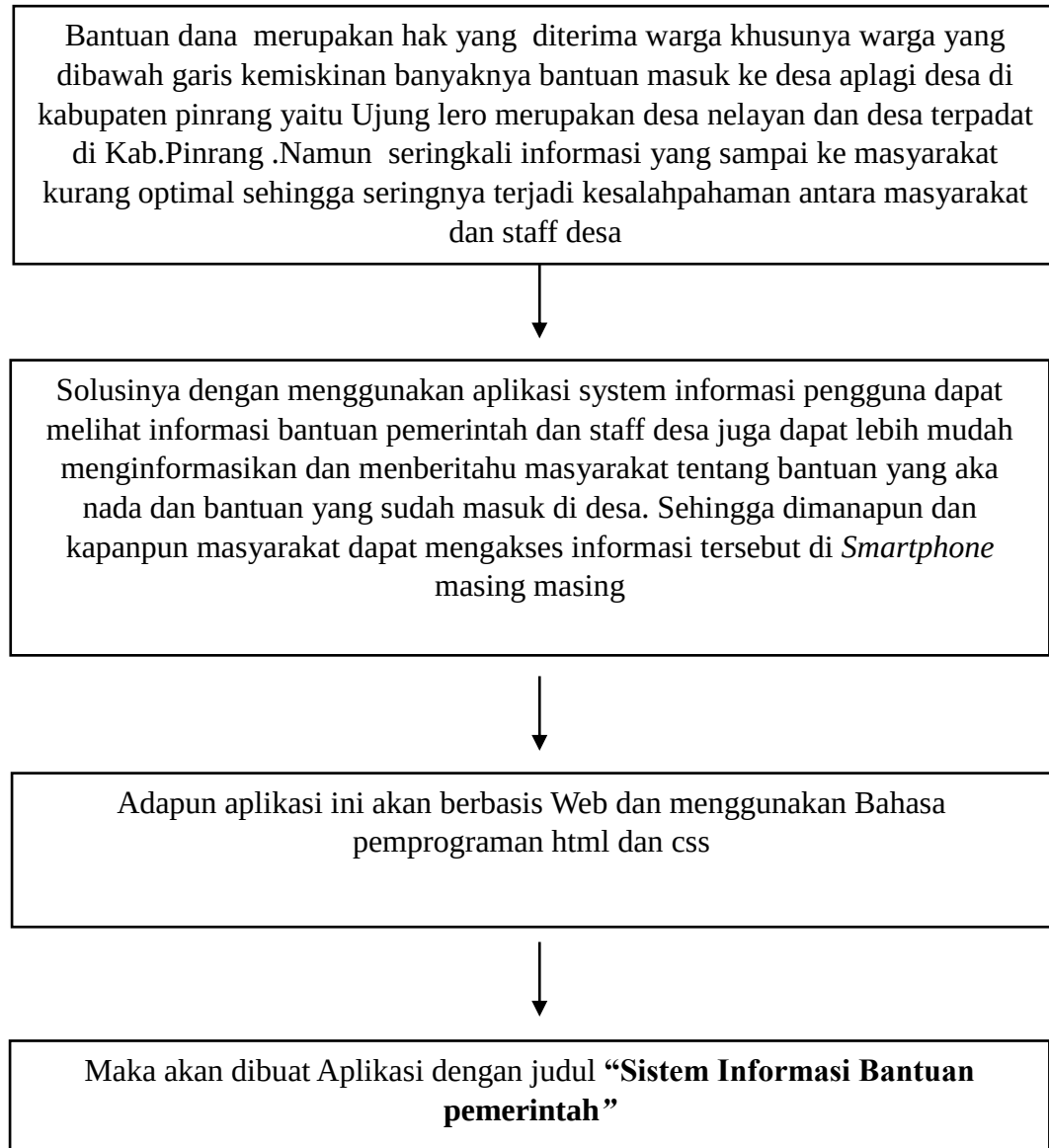
No.	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

B. Kajian Hasil Penelitian Terdahulu

Dalam penelitian ini, mengacu pada penelitian terdahulu yang relevan dengan penelitian yang akan dilakukan. Berikut ini beberapa hasil penelitian yang berhubungan dengan penelitian yang dijadikan acuan bagi peneliti yang akan dilakukan antara lain:

(Solihin, Jatnika, and Yunita n.d.) jurnal EFEKTIFITAS BANTUAN SOSIAL PROGRAM PRAKERJA DALAM MENBANTU PEREKONOMIAN 2019 penelitian ini menghasilkan rincian dan perhitungan tentang bagaimana keefektifan dari bantuan pemerintah tersebut namun dalam referensi ini spesifik karena membahas bantuan saat covid

C. Kerangka Pikir



BAB III

METODE PENELITIAN

A. Jenis Penelitian

Jenis penelitian yang digunakan adalah penelitian deskriptif dimana memberikan gambaran mengenai fenomena yang sesungguhnya terjadi dan menggunakan pendekatan kualitatif. Dalam pendekatan kualitatif perlu menekankan pada pentingnya kedekatan dengan orang-orang dan situasi penelitian, agar peneliti memperoleh pemahaman jelas tentang realitas dan kondisi kehidupan nyata.

B. Lokasi dan Waktu Penelitian

Penelitian dilakukan di Kantor Desa Maritengngae Kac Suppa Kab Pinrang dan waktu penelitian dilakukan selama ± 2 (dua) bulan di tahun 2024.

C. Alat dan Bahan Penelitian

1. Perangkat keras

- a. Laptop HP Strix G512LI-I75TB6T dengan spesifikasi hardware:
 - *Processor* : Intel® Core™ i3-8130U CPU @ 2.20GHz
 - *Installed RAM* : 4GB
 - *SSD* : 1 TB

2. Perangkat lunak

- a. XAMPP
- b. *Visual Studio Code*
- c. *PHP*

- d. *XAMPP*
- e. *Browser*

D. Teknik Pengumpulan Data

1. Wawancara

Penulis melakukan tanya jawab dan wawancara dengan pihak yang dibutuhkan dalam penelitian ini seperti staff kantor Desa Maritengngae

2. Observasi

Observasi merupakan teknik pengumpulan data dengan cara mengamati secara langsung maupun tidak langsung. Observais tidak langsung dilakukan dengan cara mengamati objek penelitian melalui media tertentu, seperti video atau foto.

3. Dokumentasi

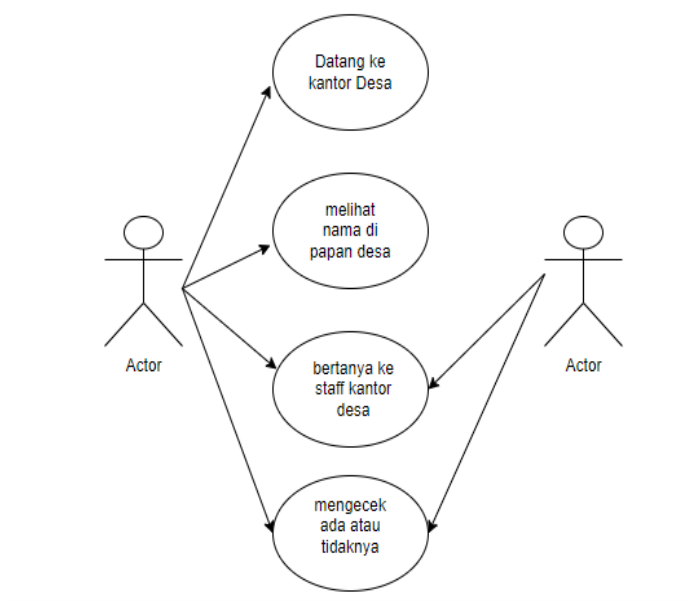
Pada tahap ini dilakukan mengumpulkan data yang berkaitan dalam bentuk dokumen. Dokumen-dokumen tersebut dapat berupa dokumen tertulis dan dokumen elektronik.

BAB IV

HASIL DAN PEMBAHASAN

A. Rancangan Sistem

1. Sistem yang berjalan

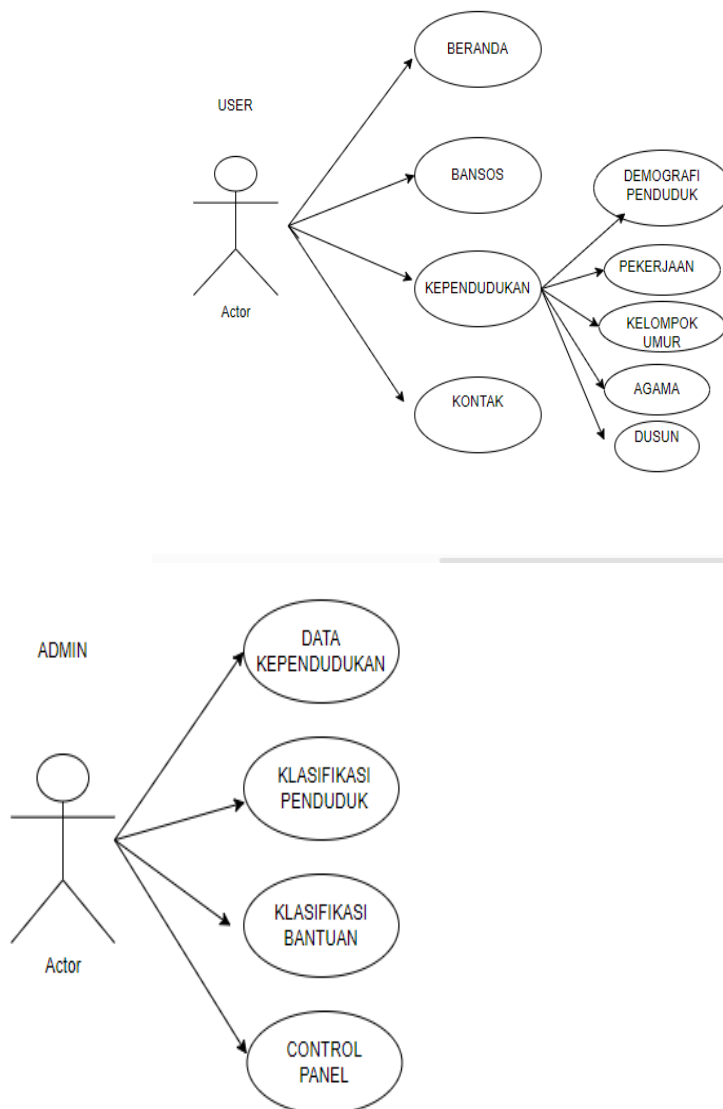


Gambar 4. 1 Sistem yang berjalan

Diagram *use case* tersebut menunjukkan interaksi antara dua aktor, yaitu user dan Admin, dengan sistem. Admin memiliki akses untuk melihat ditanyai, melihat apakah menerima atau tidak. User juga memiliki akses untuk melihat daftar penerima, melihat nama mereka sendiri. Garis panah dalam diagram menunjukkan akses atau interaksi yang dilakukan oleh masing-masing aktor terhadap use case tersebut. Diagram ini

membantu memperjelas fungsi-fungsi yang dapat diakses oleh Admin dan User dalam sistem, sehingga memudahkan perancangan dan implementasi sistem yang lebih efisien. Setelah melakukan pra penelitian penulis juga sadar bahwa informasi akurat bahwa warga terdaftar sebagai penerima kadang hanya disebarluaskan oleh mulut kemulut dimana kurang efektif karena pernah nya terjadi cekcok antara warga perihal apakah identitas keaslian yang tidak dapat di cek kebenarannya

2. Sistem yang diusulkan

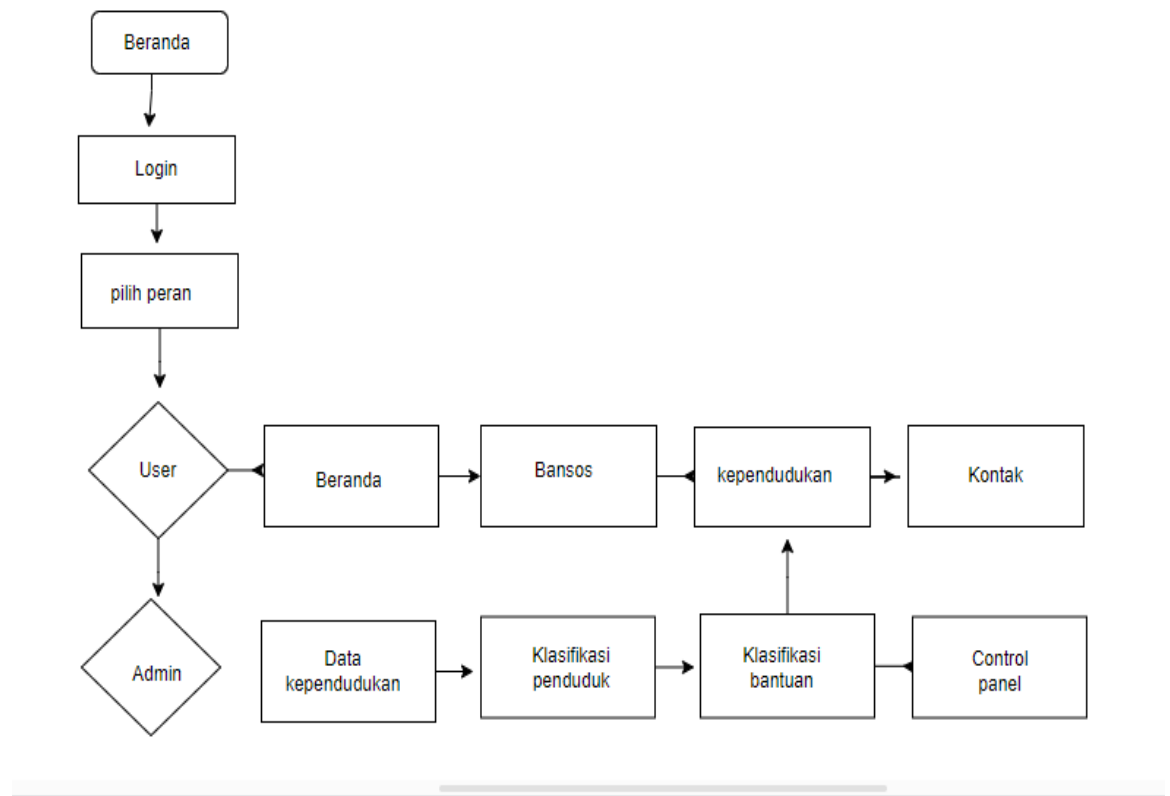


Gambar 4. 2 Sistem yang diusulkan

Diagram *use case* ini menggambarkan pengembangan sistem informasi Bantuan berbasis web dengan dua aktor utama: User, dan Admin. Admin memiliki akses untuk login, menambah data, menambah, mengakses database bantuan penerima. User memiliki akses untuk akses, melihat Informasi desa Maritengngae serta mengecek langsung nik dari user.

apakah terdaftar sebagai penerima bantuan BLT dan BPNT, user dapat mengecek tentang data umum penerima bantuan serta informasi umum desa Maritengngae seperti kontak ,email dan maps dari kantor desa maritengngae yang diharap dapat diakses oleh semua kalangan,Admin memiliki halaman yang dapat menampilkan informasi yang akan di tampilkan di halaman kependudukan dalam sistem. Admi memiliki banyak hal yanh dapat dilakukan seperti klasifikasi bantuan,klasifikasi dusun,klasifikasi bantuan penerima.yang data nya akan tampil di beranda dan dapat dilihta di menu kependudukan yang dapat diakses oleh User yang dimana menjadi Halaman beranda yang dapat di akses oleh user yang database nya di masukkan oleh admin sehingga menampilkan informasi yang ingin diperroleh oleh user. Sehingga dapat terwujudnya sistem informasi dua arah antara User dan Admin

B. Diagram Alir



Gambar 4. 3 Diagram Alir

C. Analisis Aliran Data UML

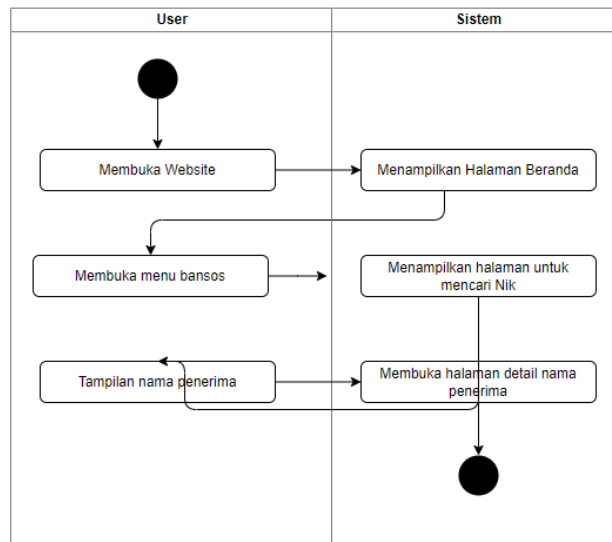
Dibahas dua diagram UML yang menggambarkan aliran data dan interaksi dalam Pengembangan Sistem Informasi Akademik Berbasis Web (Studi Kasus Kantor Desa Maritengngae), yaitu *Activity diagram* dan *Sequence diagram*.

1. Activity diagram

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

a. Activity diagram admin aplikasi

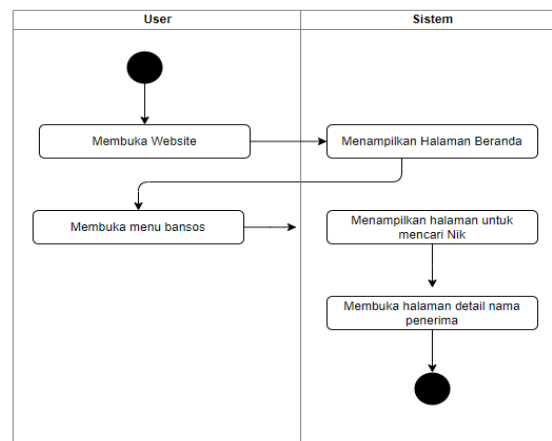
1) *Activity diagram Beranda*



Gambar 4. 4 *Activity diagram beranda*

Pada gambar 4. 2 menjelaskan cara masuk sebagai admin. Admin harus terlebih dahulu mengakses situs web, setelah itu sistem akan menampilkan form *login* dan admin harus memasukkan nama pengguna dan kata sandi. Sistem kemudian akan melakukan validasi; maka akan kembali ke halaman *login*.

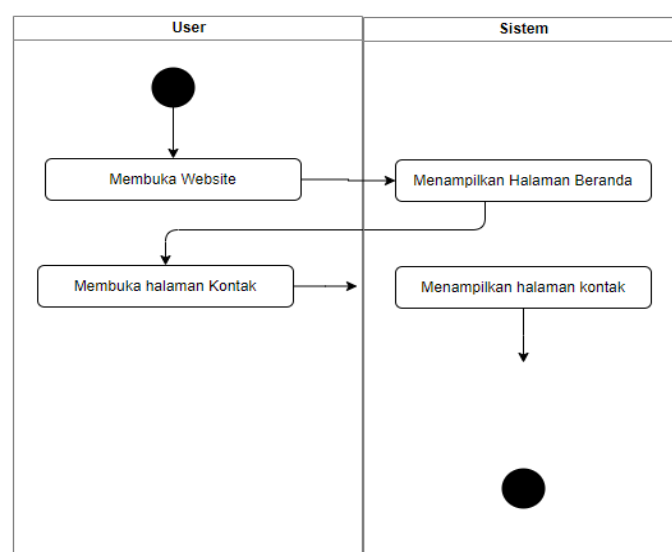
2) *Activity diagram Bansos*



Gambar 4. 5 Activity diagram Bansos

Pada gambar 4.3 Activity diagram ini menggambarkan interaksi antara Admin dan Sistem. Admin memulai dengan membuka halaman admin, yang kemudian ditampilkan oleh Sistem. Selanjutnya, Admin membuka menu kependudukan, dan Sistem menampilkan data kependudukan yang diminta. Proses berakhir setelah data kependudukan ditampilkan, menunjukkan selesainya alur kerja.

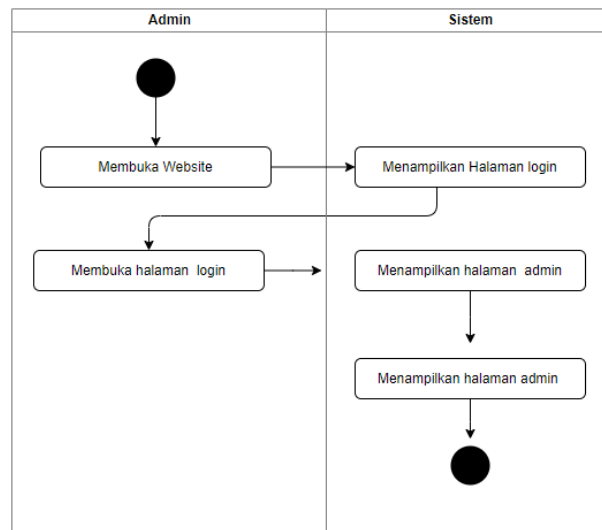
3) Activity diagram Kependudukan



Gambar 4. 6 Activity diagram kependudukan

Pada gambar 4.4 *Activity diagram* ini menunjukkan interaksi antara Admin dan Sistem. Admin membuka halaman admin, yang kemudian ditampilkan oleh Sistem. Selanjutnya, Admin membuka menu kependudukan, dan Sistem menampilkan data kependudukan . Proses berakhir setelah data kependudukan ditampilkan.

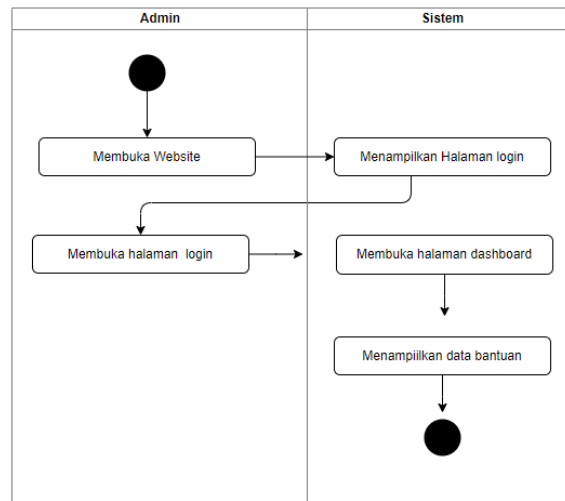
4) *Activity diagram Login*



Gambar 4. 7 Activity diagram login

Pada gambar 4.5 *Activity diagram* ini menunjukkan Admin membuka halaman admin, yang kemudian ditampilkan oleh Sistem. Admin lalu membuka menu, dan Sistem menampilkan data . Proses berakhir setelah data ditampilkan.

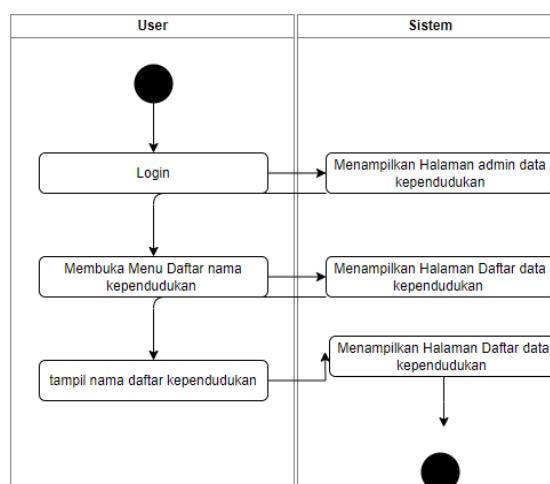
5) *Activity diagram Dashboard*



Gambar 4. 8 *Activity diagram Dashboard*

Pada gambar 4.6 *Activity diagram* ini menunjukkan Admin membuka halaman dashboard, yang kemudian ditampilkan oleh Sistem. Admin lalu membuka menu dashboard, dan Sistem menampilkan data bantuan. Proses berakhir setelah data ditampilkan.

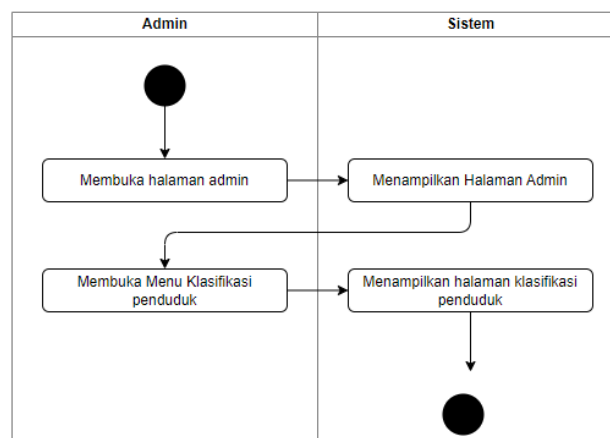
6) *Activity diagram Data Kependudukan*



Gambar 4. 9 *Activity diagram Diagram kependudukan*

Pada gambar 4.7 *Activity diagram* ini menunjukkan Admin membuka halaman admin, yang kemudian ditampilkan oleh Sistem. Admin lalu membuka menu data kependudukan , dan Sistem menampilkan data tersebut. Proses berakhir setelah data ditampilkan.

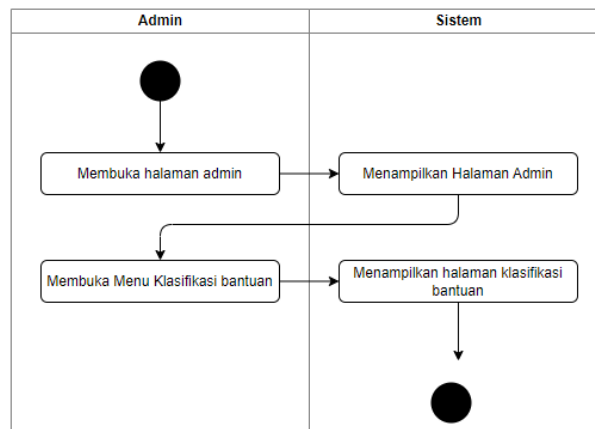
7) *Activity diagram* Klasifikasi penduduk



Gambar 4. 10 *Activity diagram* klasifikasi penduduk

Pada gambar 4.8 *Activity diagram* ini menunjukkan Admin membuka halaman admin, yang kemudian ditampilkan oleh Sistem. Admin lalu membuka menu user, dan Sistem menampilkan data penduduk. Proses berakhir setelah data ditampilkan.

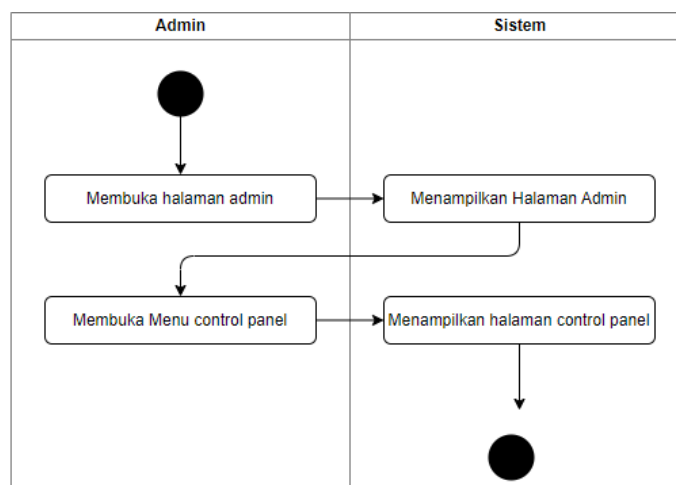
8) *Activity diagram* Klasifikasi bantuan



Gambar 4. 11 *Activity diagram* klasifikasi bantuan

Pada gambar 4.9 *Activity diagram* ini menunjukkan Admin membuka halaman admin, yang kemudian ditampilkan oleh Sistem. Admin lalu membuka menu jadwal, dan Sistem menampilkan data bantuan. Proses berakhir setelah data ditampilkan.

9) *Activity diagram* Control panel



Gambar 4. 12 *Activity diagram* control panel

Pada gambar 4.10 *Activity diagram* ini menunjukkan Admin membuka halaman admin, yang kemudian ditampilkan oleh Sistem. Admin lalu

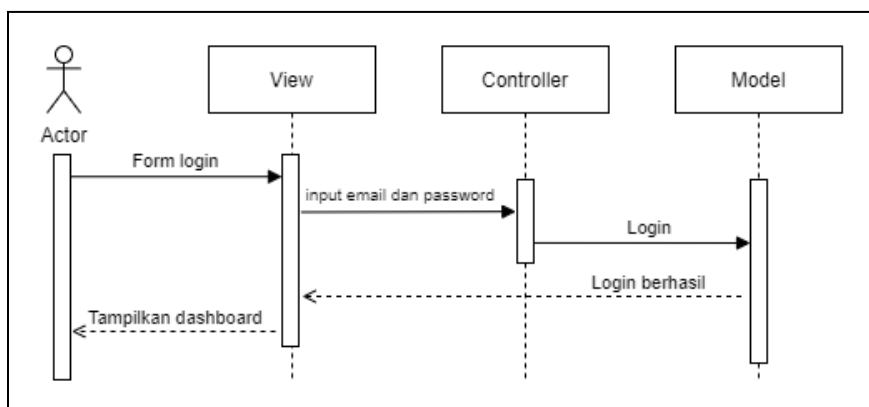
membuka control panel dan Sistem menampilkan data. Proses berakhir setelah data ditampilkan.

2. Sequence diagram

Sequence diagram merupakan diagram yang menggambarkan interaksi antara objek-objek dalam sistem secara berurutan berdasarkan waktu. Diagram ini menunjukkan bagaimana objek-objek saling bertukar pesan atau data untuk menyelesaikan suatu tugas atau *use case*.

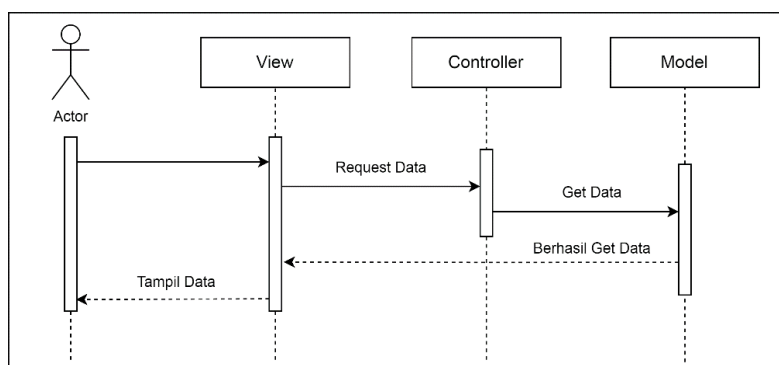
a. Sequence diagram Beranda

1) Sequence diagram Beranda



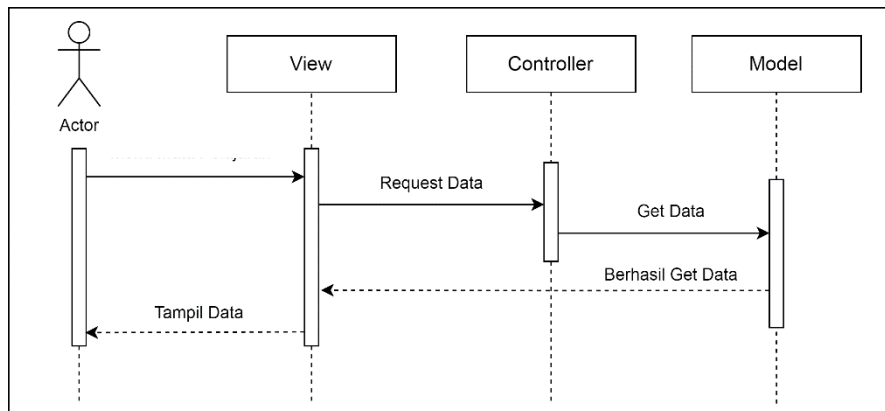
Gambar 4. 13Sequence diagram Login Beranda

2) Sequence diagram Bansos



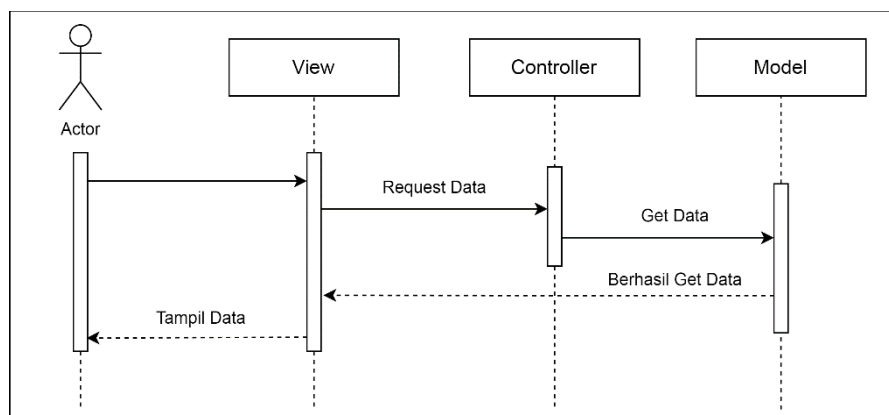
Gambar 4. 14Sequence diagram Bansos

3) Sequence diagram Kependudukan



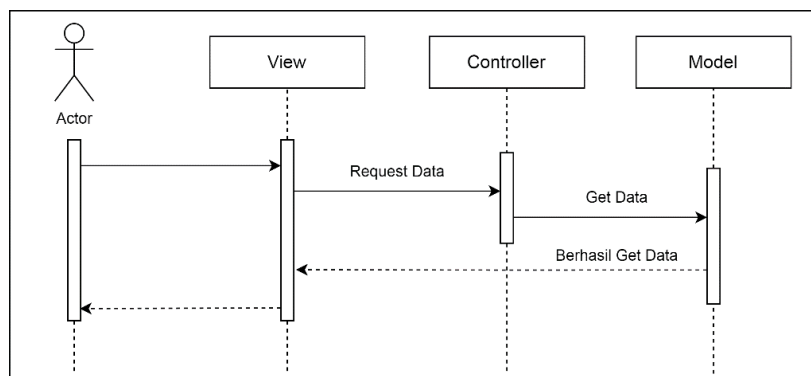
Gambar 4. 15Sequence diagram Kependudukan

4) Sequence diagram login



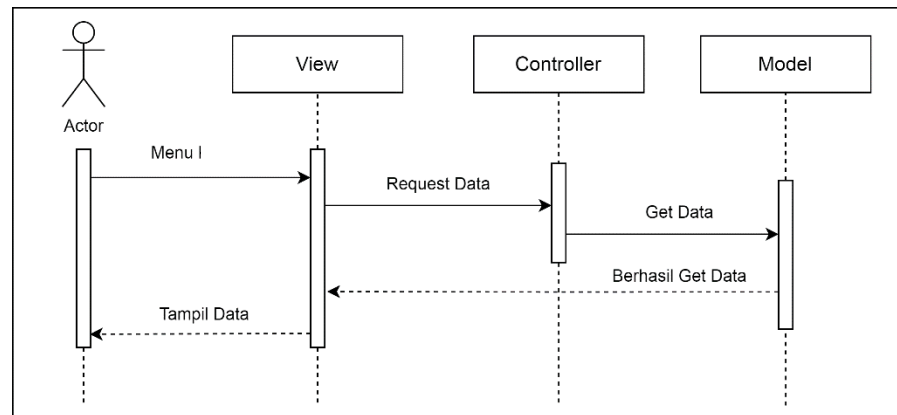
Gambar 4. 16Sequence diagram login

5) Sequence diagram Kontak



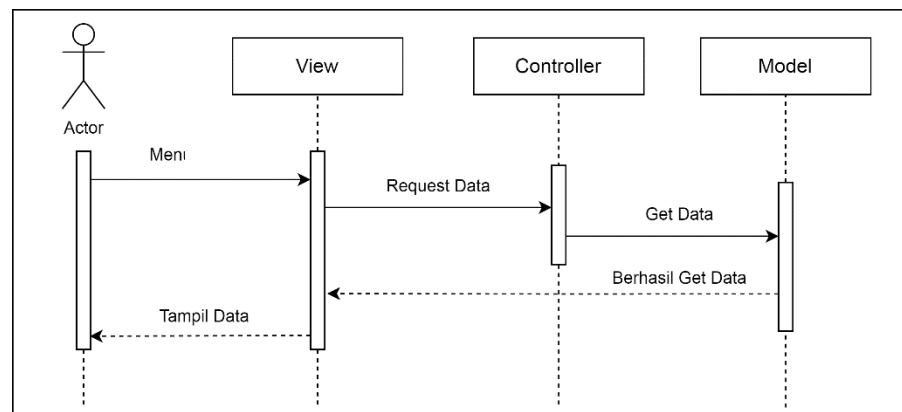
Gambar 4. 17Sequence diagram kontak

6) *Sequence diagram Dashboard*



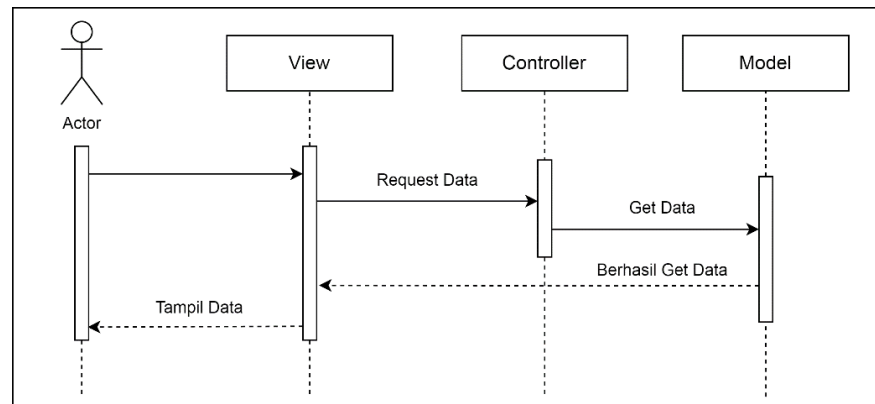
Gambar 4. 18 *Sequence diagram Dashboard*

7) *Sequence diagram Data kependudukan*



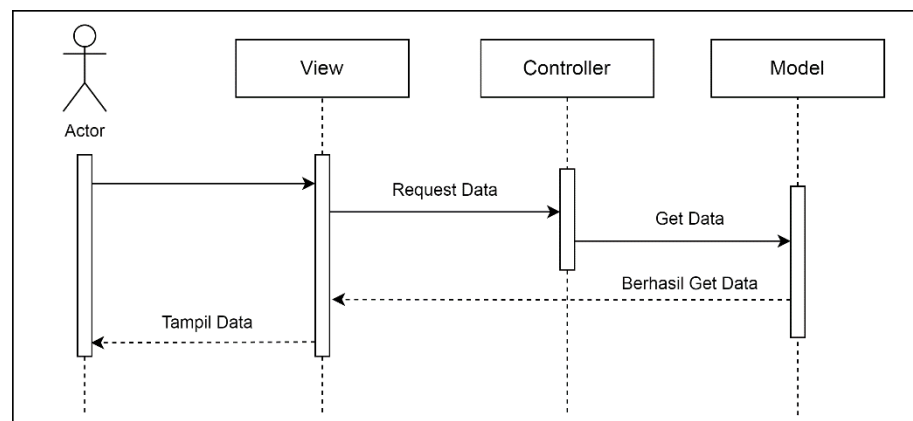
Gambar 4. 19 *Sequence diagram Data Kependudukan*

8) *Sequence diagram* Klasifikasi penduduk



Gambar 4. 20 *Sequence diagram* Klasifikasi penduduk

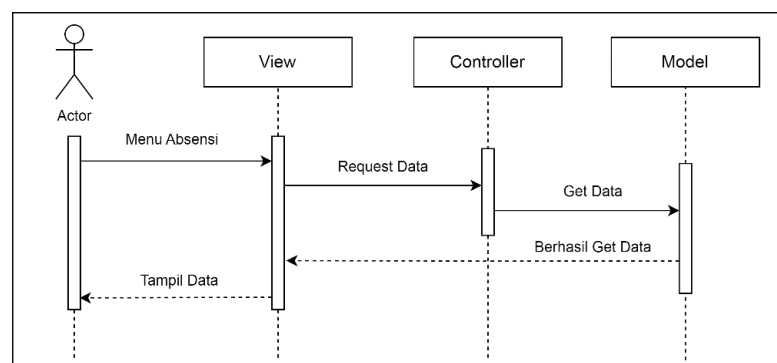
9) *Sequence diagram* Klasifikasi bantuan



Gambar 4. 21 *Sequence diagram* Klasifikasi bantuan

10). *Sequence diagram* Control panel

1) *Sequence diagram*



Gambar 4. 22 *Sequence diagram* Control panel

D. Rancangan Database

Berikut ini adalah tabel-tabel yang digunakan pada Pengembangan Sistem Informasi Akademik Berbasis Web:

1. Tabel Control

Tabel 4. 1 Tabel control

Column	Type	Attributes	Null	Default	Extra
Id	int (255)	UNSIGNED	No	NUL	Auto increment
nama_desa	varchar(255)		No	NUL	
Logo_desa	Varchar		No	NUL	
Alamat	Text		No	NUL	
Maps	Text		No	NUL	
Email	varchar		No	NUL	

2. Tabel Dusun

Tabel 4. 2 Dusun

Column	Type	Attributes	Null	Default	Extra
Id	int(11)		No		Auto increment
Dusun	Varchar(100)		No		

3. Tabel Kependudukan

Tabel 4. 3 Tabel Kependudukan

Column	Type	Attributes	Null	Default	Extra
Nik	char(16)		No		
nama_lgkp	Varchar(50)		No		
HBLKEL	Varchar(2)		No		
Jk	Tinyint(1)		No		
Tmpt_lhr	Varchar(30)		No		
Tgl_lhr	Date		No		
Tahun	Varchar(3)		No		
Bulan	Varchar(2)		No		
Hari	Varchar(2)		No		

Kecamatan	Varchar(30)		No		
Kelurahan	Varchar(30)		No		
Dsn	Int(11)		No		
Agama	Varchar(10)		No		
Bantuan	Varchar(2)		No		
Jenis_bantuan	Varchar(100)		No		
Ibu_hamil	Varchar(2)		No		
Disabilitas	Varchar(2)		No		

4. Tabel konsumsi

Tabel 4. 4 Tabel konsumsi

Column	Type	Attributes	Null	Default	Extra
Nik	Varchar(16)	UNSIGNED	No		Auto increment
Nama	Varchar(50)	UNSIGNED	No		
Bahan_makanan	Varchar(10)	UNSIGNED	No		
Frekuensi_per_minggu	Varchar(2)	UNSIGNED	No		
Pakaian_per_tahun	Varchar(2)		No		
Makan_per_hari	Varchar(2)		No		
Biaya pengobatan	Varchar(2)		Yes		
Periode	Timestamp		Yes		

5. Tabel pekerjaan

4.5 Tabel pekerjaan

Column	Type	Attributes	Null	Default	Extra
Nik	Varchar(16)		No		
Nama	Varchar(50)		No		
Pekerjaan	Varchar(30)		No		
Penghasilan_per_bulan	Int(12)		No		

6. Tabel Pendidikan

Tabel 4.6 Tabel Pendidikan

Column	Type	Attributes	Null	Default	Extra
Nik	Varchar(16)		No	NULL	
Nama	Varchar(50)		No	NULL	
Pendidikan_terakhir	Varchar(30)		No	NULL	

7. Tabel Rumah

Tabel 4. 7 Tabel Rumah

Column	Type	Attributes	Null	Default	Extra
Nik	Varchar(16)	UNSIGNED	No	NULL	Auto increment
Luas_lantai	Int(2)		No	NULL	
Jenis_lantai	Varchar(20)		No	NULL	
Jenis_dinding	Varchar(30)		No	NULL	
Fasilitas_bab	Tinyint(1)		No	NULL	
Sumber_penerangan	Tinyint(1)		No	NULL	
Sumber_air_ninum	Varchar(30)		No	NULL	
Bahan_bakar_memasak	Varchar(30)		No	NULL	

8. Tabel Tabungan

Tabel 4. 8 Tabel Tabungan

Column	Type	Attributes	Null	Default	Extra
Nik	Varchar(16)		NULL		Auto increment
Nama	Varchar(50)		NULL		
Kepemilikan_tabungan	Varchar(1)		NULL		
Jenis_tabungan	Varchar(4)		NULL		
Harga	Varchar(9)		NULL		

9. Tabel user

Tabel 4. 3 Tabel User

Column	Type	Attributes	Null	Default	Extra
Id	Int(11)		No		Auto increment
Nama	Varchar(100)		No		
Username	Varchar(10)		No		
Pass	Varchar(80)		No		

E. Sistem

1. Halaman *beranda* aplikasi

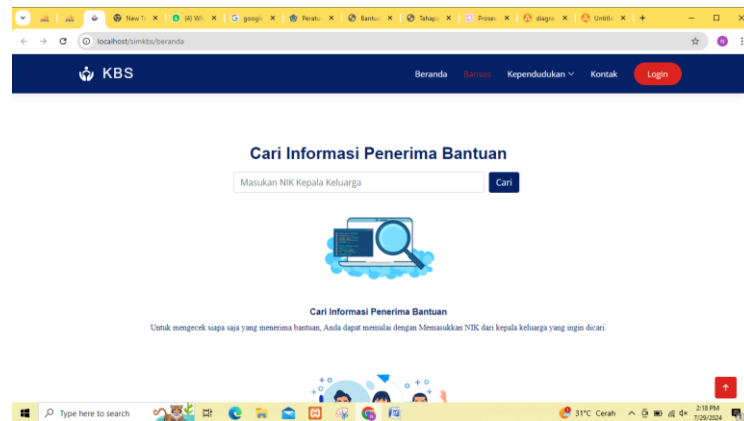
Halaman *login* aplikasi merupakan halaman utama yang digunakan oleh admin untuk mendapatkan akses ke halaman admin.



Gambar 4. 23 Halaman Beranda Aplikasi

2. Halaman Bansos

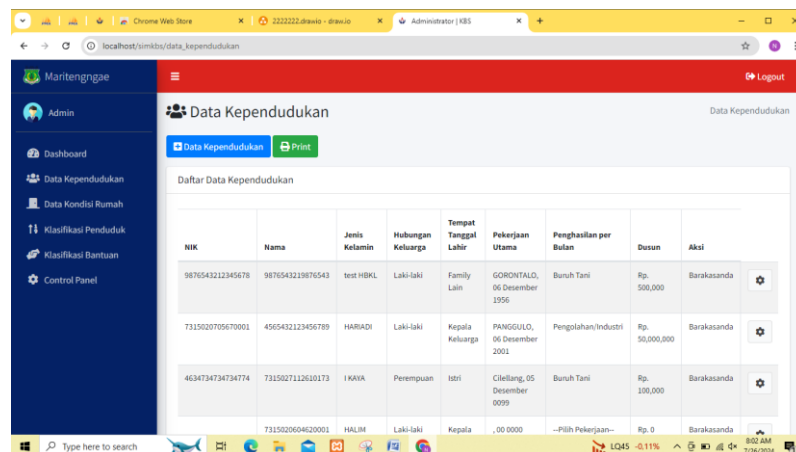
Laman dashboard admin merupakan tampilan yang menampilkan informasi mengenai apakah penerima bantuan atau tidak



Gambar 4. 24 Halaman Bansos

3. Halaman data Kependudukan

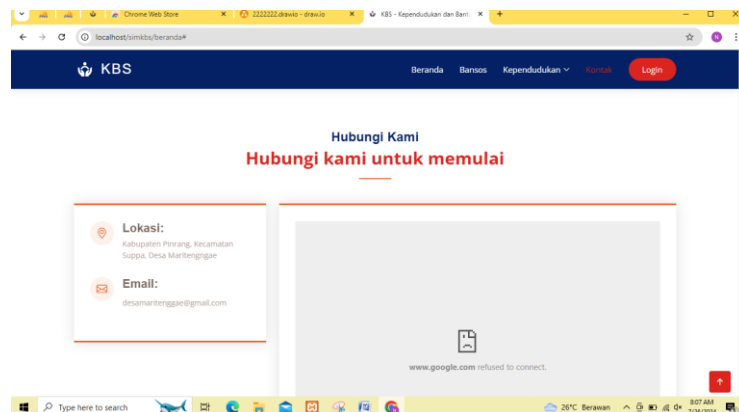
Halaman data kependudukan menampilkan data kependudukan yang ada di Kantor Desa Maritengngae



Gambar 4. 25 Halaman Data Kependudukan

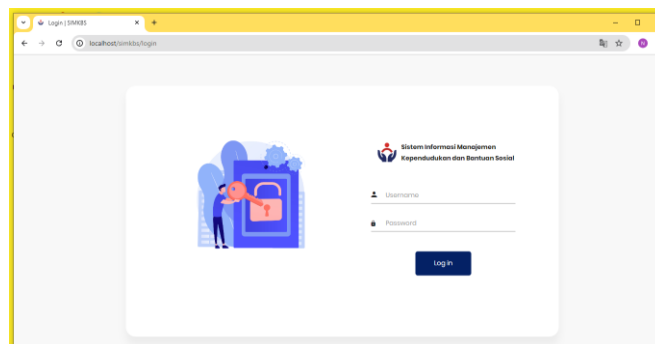
4. Halaman Kontak

Halaman kontak menampilkan halaman informasi tentang kontak personal Desa Maritengngae.



Gambar 4.26 *Halaman Kontak*

5. Halaman Login

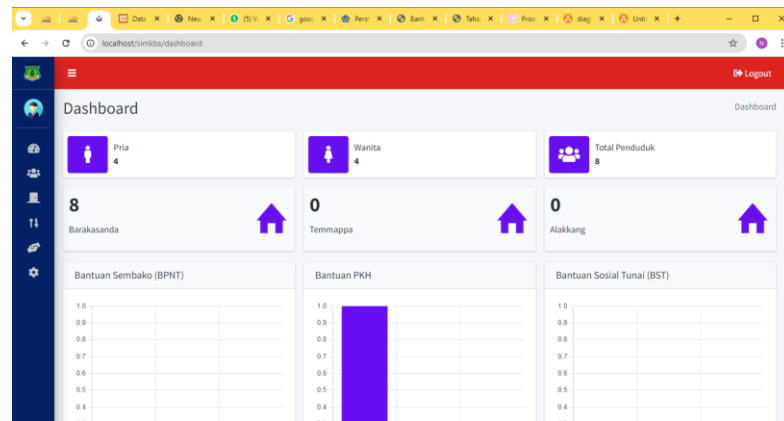


Gambar 4. 27 *Halaman Sistem informasi*

Halaman login yang dapat diakses oleh admin

6. Halaman Dashboard

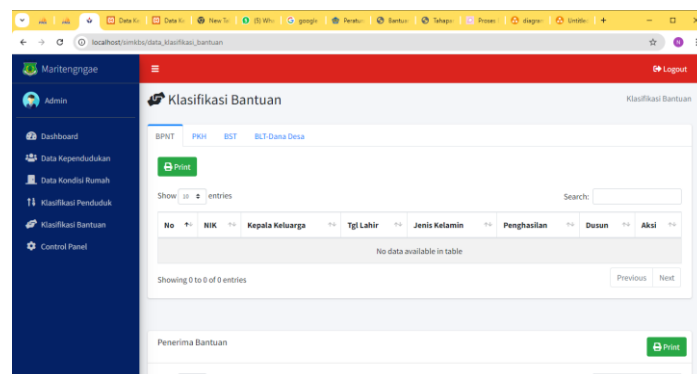
Halaman admin menampilkan data Desa Maritengngae



Gambar 4. 28 Halaman Dashboard

7. Halaman Klasifikasi bantuan

Klasifikasi bantuan menampilkan berbagai Informasi bantuan di Kantor Desa Maritengngae



Gambar 4. 29 Halaman Klasifikasi bantuan

8. Halaman Klasifikasi penduduk

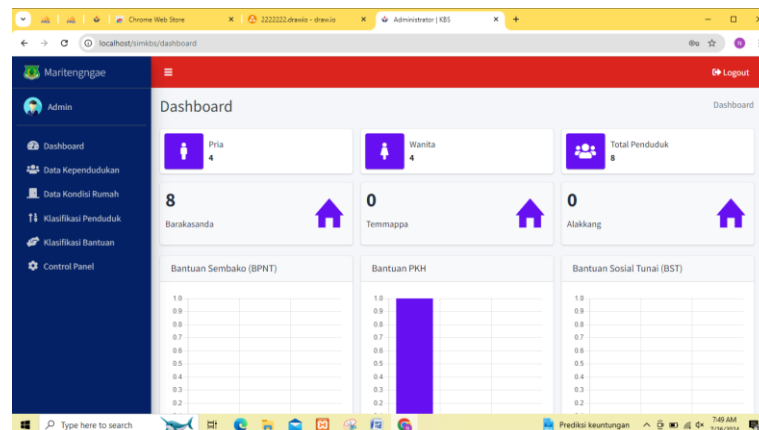
Halaman user menampilkan data data User yang terdaftar atau yang mendapatkan bantuan

IDK	Nama	Jenis Kelamin	Hubungan Keluarga	Tempat Tanggal Lahir	Pekerjaan Utama	Penghasilan per Bulan	Dusun	Akut
5876543212345678	9876543219876543	test HBKL	Laki-laki	Family Lain	GORONTALO, 06 Desember 2004	Buruh Tani	Rp. 500,000	Barakasanda
7315020709670901	4565432123456789	HARIADI	Laki-laki	Kepala Keluarga	PANGGULUD, 06 Desember 2004	Penghasilan Industri	Rp. 50,000,000	Barakasanda
4634734734734774	7315027112610173	I KAYA	Perempuan	Idiri	Cileiang, 05 Desember 0099	Buruh Tani	Rp. 100,000	Barakasanda

Gambar 4. 30 Halaman klasifikasi penduduk

9. Halaman Control Panel

Halaman menampilkan bantuan apa saja yang di data oleh website ini.



Gambar 4. 31 Halaman Control panel

F. Pengujian Sistem

Pengujian sistem bertujuan untuk mengidentifikasi kesalahan, kesenjangan, atau kekurangan dalam sistem sebelum digunakan oleh pengguna akhir. Ada dua pendekatan utama dalam pengujian sistem ini, yaitu pengujian *Black box* dan pengujian *White box*.

1. *Black box*

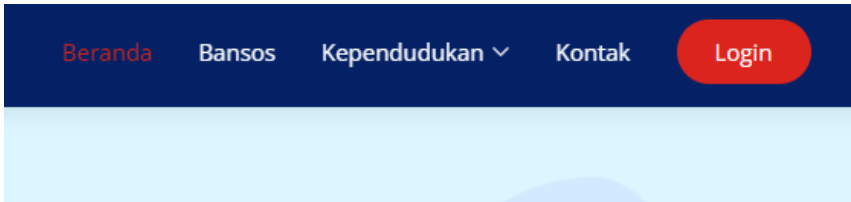
a. **Black box testing** kesalahan email dan password

Tabel 4. 11 Black box testing kesalahan email dan password

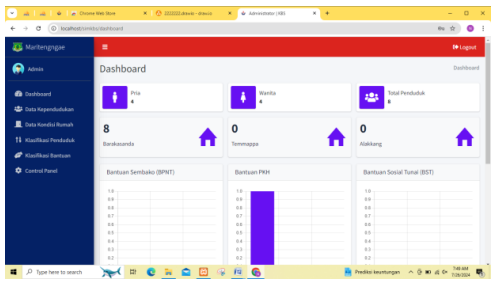
Tes Faktor	Hasil	Keterangan
Menu beranda	✓	Aplikasi sukses terbuka
		

a. Pengujian *Black Box*

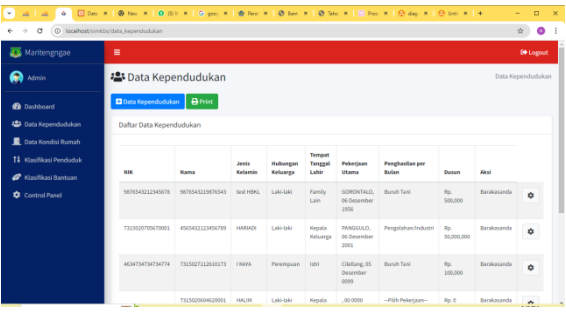
Tabel 4.12 Pengujian

Tes Faktor	Hasil	Keterangan
Tombol login	✓	Tombol login
		

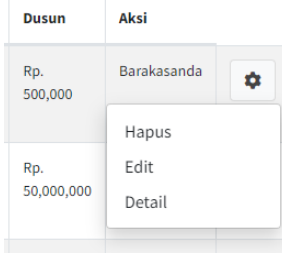
Tabel 4.13 Pengujian *Black Box*

Tes Faktor	Hasil	Keterangan
Admin	✓	Halaman akan dibuka jika password benar
		

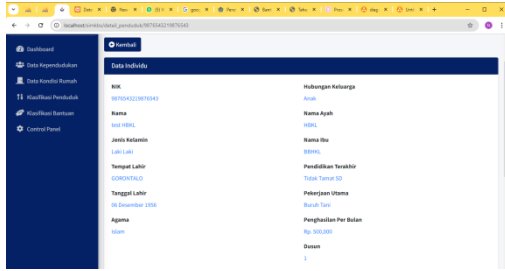
b. Pengujian *Black Box***Tabel 4.14** Pengujian *Black Box*

Tes Faktor	Hasil	Keterangan
Olah data	✓	Halaman data yang dapat di edit
		

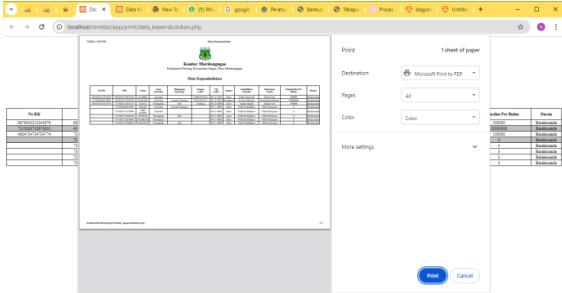
Tabel 4.15 Pengujian

Tes Faktor	Hasil	Keterangan
Tombol aksi	✓	Tombol aksi dapat dapat berjalan
		

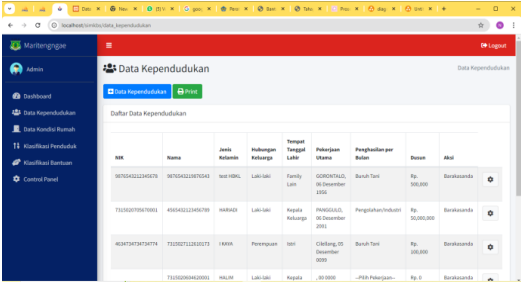
c. Pengujian *Black Box***Tabel 4.15** Pengujian *Black Box*

Tes Faktor	Hasil	Keterangan
Button detail	✓	Button aksi hapus,edit dan tambah dapat membuka detail
		

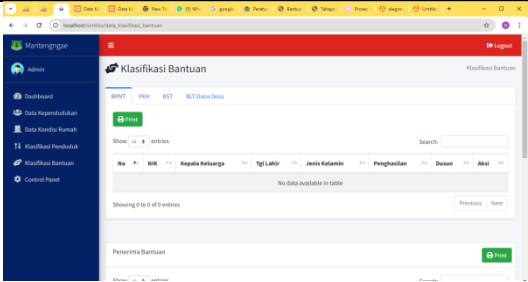
Tabel 4.16 Pengujian *Black box button*

Tes Faktor	Hasil	Keterangan
Print	✓	Tombol print dapat berjalan
		

d. Pengujian *Black Box* halaman**Tabel 4.17** Pengujian *Halaman* klasifikasi penduduk

Tes Faktor	Hasil	Keterangan
Klasifikasi penduduk	✓	Halaman klasifikasi penduduk Desa Maritengngae
		

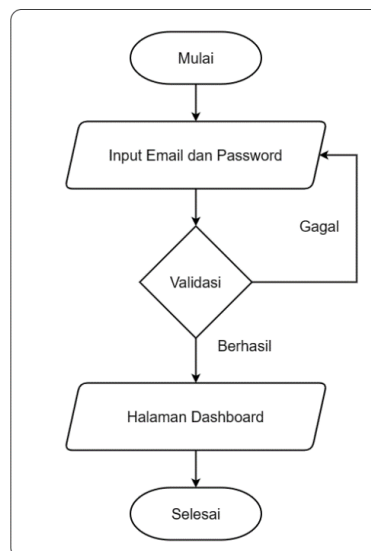
Tabel 4.18 Pengujian *Black Box*

Tes Faktor	Hasil	Keterangan
Halaman	✓	Aplikasi berhasil memindah ke halaman yang
		

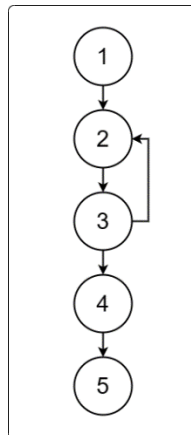
2. *White box*

a. *White box testing kesalahan email dan password*

(1) *Flowchart*

**Gambar 4. 32** Flowchart Kesalahan Email Dan Password

(2) *Flowgraph*



Gambar 4. 33 *Flowgraph* Kesalahan Email Dan Password

Berdasarkan gambar 4.73 diatas, dilakukan perhitungan sebagai berikut:

(1) Menghitung *cyclomatic complexity* $V(G)$ pada *edge* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 5$$

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

Penyelesaian :

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

$$= 5 - 5 + 2$$

$$= 2$$

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

$$= 1 + 1$$

$$= 2$$

(2) Berdasarkan perhitungan *cyclomatic complexity* dari *Flowgraph*

diatas memiliki *region* = 2

(3) *Independent path* pada *Flowgraph* yaitu:

Path 1 = 1 – 2 – 3 – 2

Path 2 = 1 – 2 – 3 – 4 – 5

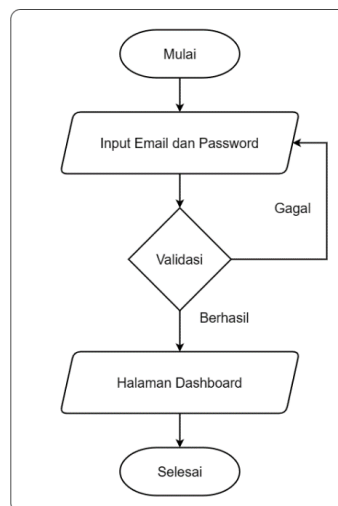
(4) Grafik matriks kesalahan *email* dan *password*

Tabel 4. 19 Grafik matriks kesalahan email dan password

	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)					$1 + 1 = 2$

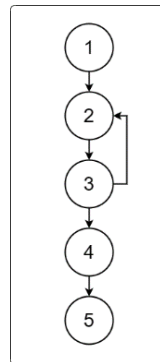
b. *White box testing login berhasil*

1) *Flowchart*



Gambar 4. 34 Flowchart login berhasil

2) *Flowgraph*



Gambar 4. 35 *Flowgraph login berhasil*

Berdasarkan gambar 4.75 diatas, dilakukan perhitungan sebagai berikut:

(1) Menghitung *cyclomatic complexity* $V(G)$ pada *edge* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 5$$

$$P \text{ (predikat node)} = 1$$

Penyelesaian :

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

$$= 5 - 5 + 2$$

$$= 2$$

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

$$= 1 + 1$$

$$= 2$$

(2) Berdasarkan perhitungan *cyclomatic complexity* dari *Flowgraph* diatas memiliki *region* = 2

(3) *Independent path* pada *Flowgraph* yaitu:

Path 1 = 1 – 2 – 3 – 2

Path 2 = 1 – 2 – 3 – 4 – 5

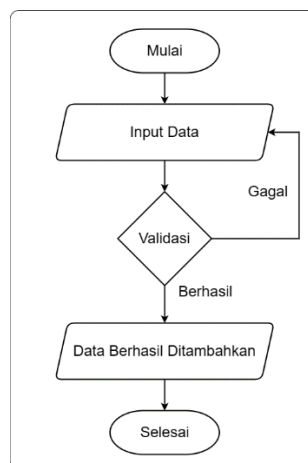
(4) Grafik matriks *login* berhasil

Tabel 4. 20 Grafik matriks login berhasil

	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)					$1 + 1 = 2$

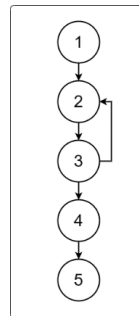
c. *White box testing* tambah data

1) *Flowchart*



Gambar 4. 36 *Flowchart* Tambah Data

2) Flowgraph



Gambar 4. 37 Flowgraph Tambah Data

Berdasarkan gambar 4.77 diatas, dilakukan perhitungan sebagai berikut:

(1) Menghitung *cyclomatic complexity* $V(G)$ pada *edge* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 5$$

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

Penyelesaian :

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

$$= 5 - 5 + 2$$

$$= 2$$

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

$$= 1 + 1$$

$$= 2$$

(2) Berdasarkan perhitungan *cyclomatic complexity* dari Flowgraph

didasar memiliki *region* = 2

(3) *Independent path* pada Flowgraph yaitu:

Path 1 = 1 – 2 – 3 – 2

Path 2 = 1 – 2 – 3 – 4 – 5

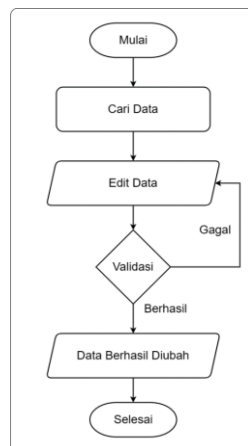
(4) Grafik matriks tambah data

Tabel 4. 21. Grafik matriks tambah data

	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)					$1 + 1 = 2$

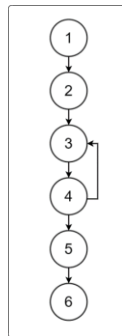
d. White box testing ubah data

1) Flowchart



Gambar 4. 38 Flowchart Ubah Data

2) Flowgraph



Gambar 4. 39 Flowchart Ubah Data

Berdasarkan gambar 4.79 diatas, dilakukan perhitungan sebagai berikut:

(1) Menghitung *cyclomatic complexity* $V(G)$ pada *edge* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 6$$

$$N \text{ (node)} = 6$$

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

Penyelesaian :

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

$$= 6 - 6 + 2$$

$$= 2$$

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

$$= 1 + 1$$

$$= 2$$

(2) Berdasarkan perhitungan *cyclomatic complexity* dari *Flowgraph* diatas memiliki *region* = 2

(3) *Independent path* pada *Flowgraph* yaitu:

$$\text{Path 1} = 1 - 2 - 3 - 4 - 3$$

Path 2 = 1 – 2 – 3 – 4 – 5 – 6

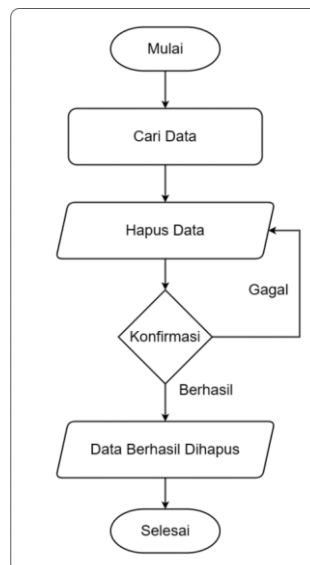
(4) Grafik matriks ubah data

Tabel 4. 22 Grafik matriks ubah data

	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)						$1 + 1 = 2$

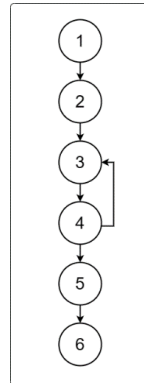
e. White box testing hapus data

1) Flowchart



Gambar 4. 40 flowchart hapus data

2) Flowgraph



Gambar 4. 41 Flowgraph Hapus Data

Berdasarkan gambar 4.81 diatas, dilakukan perhitungan sebagai berikut:

(1) Menghitung *cyclomatic complexity* $V(G)$ pada *edge* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 6$$

$$N \text{ (node)} = 6$$

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

Penyelesaian :

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

$$= 6 - 6 + 2$$

$$= 2$$

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

$$= 1 + 1$$

$$= 2$$

(2) Berdasarkan perhitungan *cyclomatic complexity* dari *Flowgraph*

diatas memiliki *region* = 2

(3) *Independent path* pada *Flowgraph* yaitu:

Path 1 = 1 – 2 – 3 – 4 – 3

Path 2 = 1 – 2 – 3 – 4 – 5 – 6

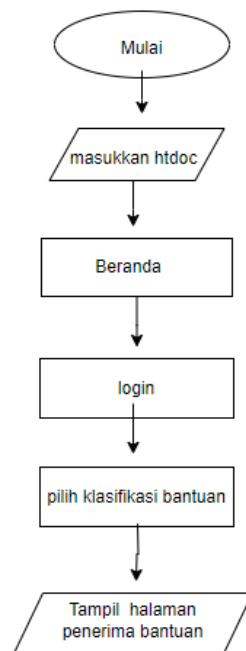
(4) Grafik matriks hapus data

Tabel 4. 23 Grafik matriks hapus data

	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)						$1 + 1 = 2$

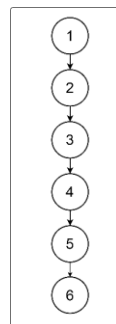
g. White box testing halaman Bantuan

1) Flowchart



Gambar 4.42 Flowchart Halaman Kependudukan

2) Flowgraph



Gambar 4.43 flowgraph Halaman Kependudukan

Berdasarkan gambar 4.83 diatas, dilakukan perhitungan sebagai berikut:

(1) Menghitung *cyclomatic complexity* $V(G)$ pada *edge* dan *node*.

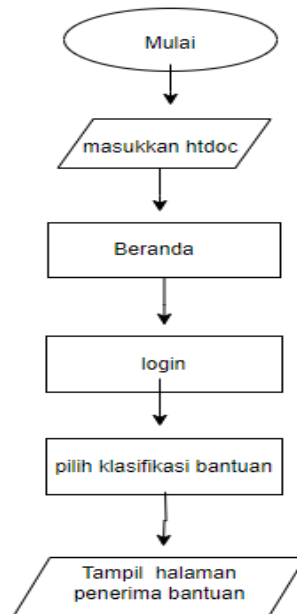
$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 6$$

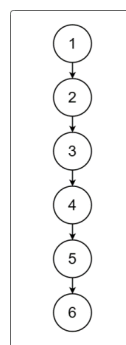
h. White box testing Halaman Bantuan

1) Flowchart



Gambar 4. 43 Flowchart Halaman Bantuan

2) Flowgraph



Gambar 4. 44 Flowgraph Halaman Bantuan

Berdasarkan gambar 4.85 diatas, dilakukan perhitungan sebagai berikut:

(1) Menghitung *cyclomatic complexity* $V(G)$ pada *edge* dan *node*.

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 6$$

$$P \text{ (Predikat node)} = 0$$

Penyelesaian :

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

$$= 5 - 6 + 2$$

$$= 1$$

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

$$= 0 + 1$$

$$= 1$$

(2) Berdasarkan perhitungan *cyclomatic complexity* dari *Flowgraph*

diatas memiliki *region* = 1

(3) *Independent path* pada *Flowgraph* yaitu:

Path 1 = 1 – 2 – 3 – 4 – 5 – 6

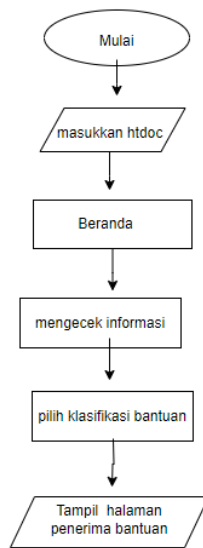
(4) Grafik matriks halaman bantuan

Tabel 4. 25 Grafik matriks halaman bantuan

	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 - 1 = 0
5						1	1 - 1 = 0
6							0
	SUM (E + 1)						0 + 1 = 1

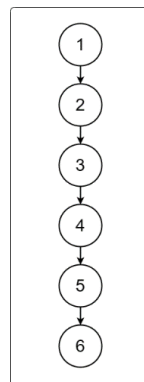
i. *White box testing* Halaman Sistem informasi

1) *Flowchart*



Gambar 4. 45 *Flowchart* Halaman Sistem informasi

2) *Flowgraph*



Gambar 4. 46 *Flowgraph* Halaman Sistem informasi

Berdasarkan gambar 4.87 diatas, dilakukan perhitungan sebagai berikut:

(5) Menghitung *cyclomatic complexity* $V(G)$ pada *edge* dan *node*.

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 6$$

$$P \text{ (Predikat node)} = 0$$

Penyelesaian :

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

$$= 5 - 6 + 2$$

$$= 1$$

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

$$= 0 + 1$$

$$= 1$$

(6) Berdasarkan perhitungan *cyclomatic complexity* dari *Flowgraph*

diatas memiliki *region* = 1

(7) *Independent path* pada *Flowgraph* yaitu:

Path 1 = 1 – 2 – 3 – 4 – 5 – 6

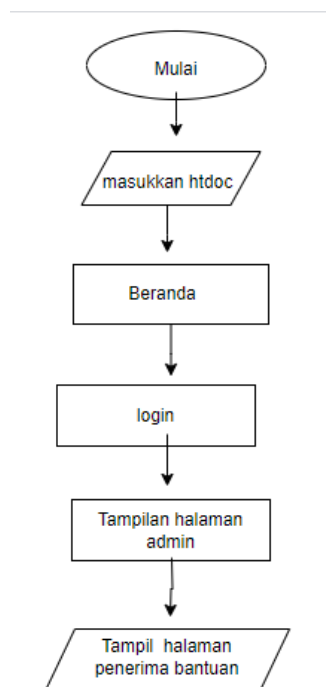
(8) Grafik matriks halaman sistem informasi

Tabel 4. 26 Grafik matriks halaman sistem informasi

	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 - 1 = 0
5						1	1 - 1 = 0
6							0
	SUM (E + 1)						0 + 1 = 1

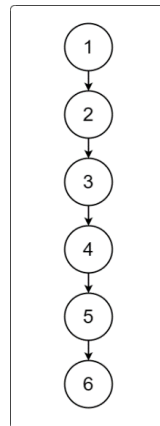
j. White box testing Halaman Admin

1) Flowchart



Gambar 4. 47 Flowchart halaman admin

2) Flowgraph



Gambar 4. 48 Flowgraph halaman admin

Berdasarkan gambar 4. 89 diatas, dilakukan perhitungan sebagai berikut:

- (1) Menghitung *cyclomatic complexity* $V(G)$ pada *edge* dan *node*.

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 6$$

$$P \text{ (Predikat node)} = 0$$

Penyelesaian :

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

$$= 5 - 6 + 2$$

$$= 1$$

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

$$= 0 + 1$$

$$= 1$$

- (2) Berdasarkan perhitungan *cyclomatic complexity* dari *Flowgraph*

didasar memiliki *region* = 1

(3) *Independent path* pada *Flowgraph* yaitu:

Path 1 = 1 – 2 – 3 – 4 – 5 – 6

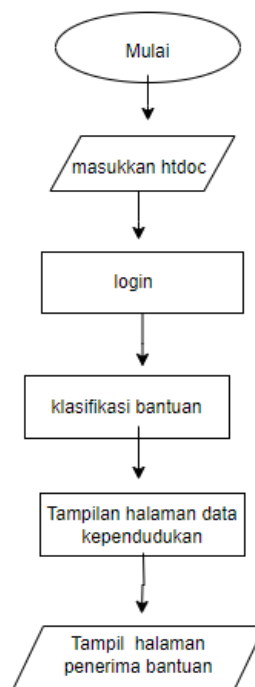
(4) Grafik matriks halaman admin

Tabel 4. 27 Grafik matriks halaman admin

	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		$1 - 1 = 0$
5						1	$1 - 1 = 0$
6							0
	SUM (E + 1)						$0 + 1 = 1$

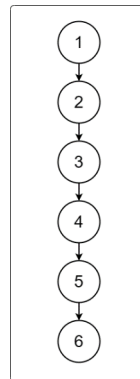
k . *White box testing* Klasifikasi bantuan

1) *Flowchart*



Gambar 4. 49 *Flowchart* Klasifikasi bantuan

2) *Flowgraph*



Gambar 4. 50 *Flowgraph* Klasifikasi bantuan

Berdasarkan gambar 4.91 diatas, dilakukan perhitungan sebagai berikut:

(1) Menghitung *cyclomatic complexity* $V(G)$ pada *edge* dan *node*.

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 6$$

$$P \text{ (Predikat node)} = 0$$

Penyelesaian :

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

$$= 5 - 6 + 2$$

$$= 1$$

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

$$= 0 + 1$$

$$= 1$$

(2) Berdasarkan perhitungan *cyclomatic complexity* dari *Flowgraph* diatas memiliki *region* = 1

(3) *Independent path* pada *Flowgraph* yaitu:

Path 1 = 1 – 2 – 3 – 4 – 5 – 6

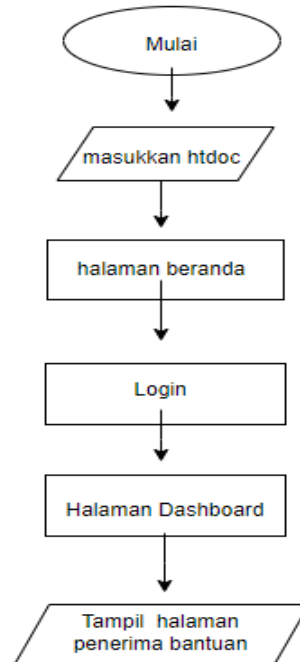
(4) Grafik matriks klasifikasi bantuan

Tabel 4. 28 Grafik matriks klasifikasi bantuan

	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		$1 - 1 = 0$
5						1	$1 - 1 = 0$
6							0
	SUM (E + 1)						$0 + 1 = 1$

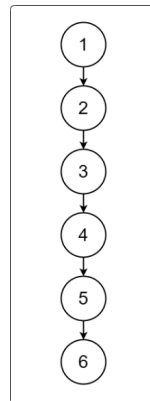
1. *White box testing* Halaman Informasi bantuan

1) *Flowchart*



Gambar 4. 51 *Flowchart* Halaman Informasi bantuan

2) *Flowgraph*



Gambar 4. 52 *Flowgraph* Halaman Informasi bantuan

Berdasarkan gambar 4.93 diatas, dilakukan perhitungan sebagai berikut:

(1) Menghitung *cyclomatic complexity* $V(G)$ pada *edge* dan *node*.

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 6$$

$$P \text{ (Predikat node)} = 0$$

Penyelesaian :

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

$$= 5 - 6 + 2$$

$$= 1$$

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

$$= 0 + 1$$

$$= 1$$

(2) Berdasarkan perhitungan *cyclomatic complexity* dari *Flowgraph* diatas memiliki *region* = 1

(3) *Independent path* pada *Flowgraph* yaitu:

Path 1 = 1 – 2 – 3 – 4 – 5 – 6

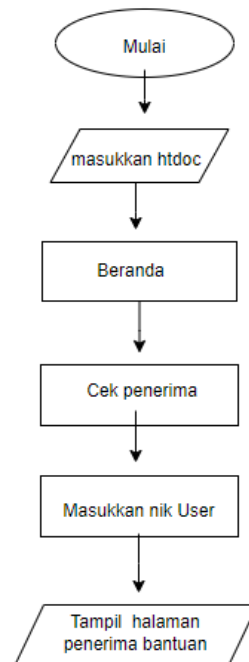
(4) Grafik matriks halaman informasi bantuan

Tabel 4. 29 Grafik matriks halaman informasi bantuan

	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		$1 - 1 = 0$
5						1	$1 - 1 = 0$
6							0
	SUM (E + 1)						$0 + 1 = 1$

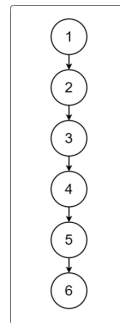
m. *White box testing* Halaman User

1) *Flowchart*



Gambar 4. 52 *Flowchart* Halaman User

2) *Flowgraph*



Gambar 4. 53 *Flowgraph* Halaman User

Berdasarkan gambar 4.95 diatas, dilakukan perhitungan sebagai berikut:

(5) Menghitung *cyclomatic complexity* $V(G)$ pada *edge* dan *node*.

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 6$$

$$P \text{ (Predikat node)} = 0$$

Penyelesaian :

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

$$= 5 - 6 + 2$$

$$= 1$$

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

$$= 0 + 1$$

$$= 1$$

(6) Berdasarkan perhitungan *cyclomatic complexity* dari *Flowgraph* diatas memiliki *region* = 1

(7) *Independent path* pada *Flowgraph* yaitu:

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

BAB V

PENUTUP

A. Kesimpulan

Merujuk pada hasil kajian yang telah rampung, berikut ini adalah kesimpulan yang dapat dirumuskan:

1. Dalam penelitian yang telah dilakukan, penulis berhasil mengembangkan sebuah sistem informasi bantuan pemerintah di Kantor Desa Maritengngae.
2. Sistem informasi ini dirancang agar dapat digunakan oleh user bantuan, staff, dan user, sehingga mempermudah mereka dalam mengelola dan mengakses informasi akademik yang relevan.
3. Sistem informasi ini dibangun menggunakan Laravel sebagai framework fullstack. Visual Studio Code digunakan sebagai text editor dan MySQL sebagai sistem manajemen basis data.

B. Saran

Dalam proses studi ini, peneliti menyadari adanya sejumlah keterbatasan yang memerlukan perbaikan dan pengembangan lebih lanjut. Oleh karena itu, peneliti memberikan beberapa saran untuk penelitian di masa mendatang, antara lain:

1. Untuk meningkatkan fungsionalitas dan kenyamanan pengguna yaitu staff dan admin dimana Website ini akan dikelola oleh admin staff desa oleh Kantor Desa Maritengngae.

2. Pengembangan informasi bantuan lain seperti PKH, dan bantuan untuk lansia yang dapat diperoleh datanya di Kantor Desa Maritengngae.
3. Ada 4 dusun di kecamatan Maritengngae diharapkan adanya tambahan vitur setiap dusus untuk lebih mempermudah aksesnya informasi kependudukan
4. Informasi data diri penerima bantuan agar lebih di jaga dengan rahasia kafena merupakan data diri sensiti bagi penerima bantuan..

DAFTAR PUSTAKA

- Abdul Wahab Podungge, dkk. 2023. “EFEKTIVITAS BANTUAN LANGSUNG TUNAI BAGI Universitas Bina Taruna Gorontalo Abdul Wahab Podungge , Cs : Efektivitas Bantuan Langsung Tunai” 10: 181–95.
- Darajat, Z, R Kambau, and ... 2022. “Sistem Informasi Monitoring Dana Desa Berbasis Web Menggunakan Framework Laravel Studi Kasus: Desa Malewong” *Jurnal INSYPRO* ...: 1–5. <https://journal3.uin-alauddin.ac.id/index.php/insypro/article/download/29990/15347>.
- Dwi Ratna Sari, Fitriani, and Irfan Hidayat. 2018. “Perancangan Sistem Informasi Pendataan Dana Bantuan Desa Menggunakan Borland Delphi 7.0 Pada Kantor Kecamatan Gombong.” *Jurnal E-Komtek (Elektro-Komputer-Teknik)* 2(2): 117–28.
- Gunawan, Agus, Eliana Wulandari, and Eti Suminartika. 2022. “Efektivitas Pelaksanaan Program Bantuan Benih Jagung Hibrida Di Kecamatan Nagreg k ... Bandung.” *Jurnal Agrinika* 6(2): 161–75.
- Husna, Muhammad Aliy ... ul, and Perani Rosyani. 2021. “Implementasi Sistem Monitoring Jaringan Dan Server Menggunakan Zabbix Yang Terintegrasi Dengan Grafana Dan Telegram.” *JURIKOM (Jurnal Riset Komputer)* 8(6): 247.
- Lawa, Hironimus Wangge, Yoseph D Da, Yen Khwuta, and Elvira Esperanza Sala. 2023. “Bantuan Dana dedsa hanglade.” 8(1): 103–9.
- Mustikawati, Titis Ulfa. 2022. “Sistem Informasi Pengelolaan Bantuan Desa Banjardowo.” *Skripsi Universitas MUhammadiyah Surakarta*.
- Nawassyarif, M. Julkarnain, and Kiki Rizki Ananda. 2020. “Sistem Informasi Pengolahan Data Ternak Unit Pelaksana Teknis Produksi Dan Kesehatan Hewan Berbasis Web.” *Jurnal Informatika, Teknologi dan Sains* 2(1): 32–39.
- Oleh, Menurun et al. 2022. “Efektivitas Program Bantuan Langsung Tunai Dana Desa Tahun