BAB I

PENDAHULUAN

A. Latar Belakang

Saat ini, penting untuk memahami bagaimana kemajuan teknologi terjadi. Setiap teknologi lahir dengan target. Contohnya, mesin pencari tercipta untuk menyortir beberapa besar data secara online. Pada setiap perkembangannya, sebuah teknologi baru mengkolaborasikan teknologi yang sudah ada untuk menghadirkan subjek yang lebih baik dari yang digunakan sebelumnya (Mangero & Abidin, 2022).

Kondisi alam Indonesia yang tidak sepenuhnya dapat diprediksi membuat pemerintah serta lembaga terkait tidak dapat berjalan sendiri dalam menangani setiap bencana yang terjadi, kini berbagai kalangan hadir tanpa pamrih untuk turut ikut serta membantu mengulurkan tangan sebagai partner pemerintah dan masyarakat berangkat dari keterpurukan. Mulai dari lembaga non pemerintah hingga lembaga organisasi keagamaan tidak tinggal diam menyaksikan duka yang melanda (Pangeran Bungsu, 2023).

Meskipun Badan Nasional Penanggulangan Bencana (BNPB) merupakan lembaga pemerintah yang bertugas menangani tentang kebencanaan, tidak menutup kemungkinan untuk gerakan sosial lain dapat bermitra untuk tindakan cepat pada kejadian-kejadian luar biasa yang bersifat kebencanaan. Pemerintah merupakan penanggung jawab inti dalam pelaksanaan penanggulangan bencana dan gerakan sosial kebencanaan merupakan kekuatan luar biasa dalam

pelaksanaan penanggulangan dan mitigasi bencana (Hawing, 2021). Salah satu organisasi kemanusiaan nasional di bawah naungan organisasi sosial yang sampai saat ini masih eksis serta memiliki andil dalam penanggulangan bencana yaitu MDMC (*Muhamadiyah Disaster Management Center*). *Muhammadiyah Disaster Management Center* (MDMC) mengambil bagian sebagai lembaga penanggulangan bencana untuk membantu masyarakat mulai dari pra bencana, tanggap darurat dan pasca bencana. Di dalam penanganan bencana terdapat beberapa aspek yaitu aspek mitigasi bencana (pencegahan), kegawatdaruratan saat terjadinya bencana, dan aspek rehabilitasi (Hawing, 2019).

MDMC (*Muhammadiyah Disaster Management Center*) atau Lembaga Penanggulangan Bencana Muhammadiyah merupakan sebuah lembaga sebagai unsur pembantu pimpinan Muhammadiyah dalam proses penanggulangan bencana serta revitalisasi paska bencana sebagai bentuk Dakwah Amar Ma‟ruf Nahi Munkar (Al-Haq, 2019).

Pengaruh teknologi dalam kehidupan manusia saat ini memberi pengaruh yang sangat besar dalam setiap aktivitasnya. Salah satu teknologi yang sering ramai dibicarakan adalah internet (*Interconnected Network*) (Kusniawan & Sardiarinto, 2016). Seiring perkembangan teknologi yang semakin canggih, diperlukan pengaksesan informasi yang cepat dan akurat. Komputer menjadikan segala pekerjaan diselesaikan dengan cepat baik yang sederhana sampai yang komplek sekalipun. Pengolahan informasi yang dulu dilakukan secara pencatatan kini mulai berkembang menuju sistem yang terkomputerisasi (Habil, n.d.).

Sebuah organisasi dapat meningkatkan efektivitas kerjanya dengan mengelola data secara komputerisasi dibandingkan dilakukan secara manual untuk menghasilkan informasi yang lebih akurat. Sistem pengolahan data yang efesien dan efektif dapat berpengaruh besar dalam mencapai tujuan bersama dalam sebuah organisasi (Fasirah, 2021). Dengan adanya aplikasi ini, penulis ingin meningkatkan kecepatan dan ketepatan dalam mengolah data, sehingga informasi yang dibutuhkan dapat segera diperoleh dan tepat pada waktunya tanpa batasan ruang dan waktu, informasi yang dapat di akses kapan saja dan dimana saja, baik dalam kehidupan sehari-hari.

Berdasarkan permasalahan yang telah dijabarkan diatas, maka penulis mengangkat judul ***"Aplikasi Elektronik Muhammadiyah Disaster Management Center (E-Mdmc)"*** dimana aplikasi ini memuat informasi-informasi tentang *Muhammadiyah Disaster Management Center* (MDMC) seperti profil, sejarah, struktur organisasi, program kegiatan, serta form yang dapat dimanfaatkan pengguna untuk melaporkan bencana yang terjadi. Aplikasi ini juga dapat memudahkan petugas MDMC (*Muhammadiyah Disaster Management Center)* untuk mendapatkan informasi bencana yang dilaporkan pengguna.

B. Rumusan Masalah

Berdasarkan latar belakang yang telah di uraikan di atas maka peneliti merumuskan bagaimana cara membuat sebuah Aplikasi Elektronik Muhammadiyah Disaster Management Center (E-MDMC) Berbasis Web?

C. Tujuan Penelitian

Berdasarkan rumusan masalah yang ada, maka rumusan masalah yang hendak dicapai dalam penulisan serta pelaksanaan ini adalah cara membuat sebuah Aplikasi Elektronik *Muhammadiyah Disaster Management Center* (E-MDMC) Berbasis Web.

D. Batasan Masalah

Pembatasan suatu masalah digunakan untuk menghindari adanya penyimpangan maupun pelebaran pokok masalah supaya penelitian tersebut lebih terarah dan memudahkan dalam pembahasan sehingga tujuan penelitian akan tercapai. Batasan masalah dalam penelitian ini adalah Aplikasi ini menampilkan Profile MDMC meliputi data anggota, program atau kegiatan, foto kegiatan, sejarah, Serta susunan kepengurusan organisasi.

E. Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan manfaat baik peneliti dan semua pihak-pihak yang terkait. Manfaat penelitian ini dintara lain sebagai berikut:

1. Manfaat Bagi Peneliti

Mampu mendewasakan pola pikir penulis dalam memecahkan masalah yang ada di lingkungan luar serta mampu menambah wawasan mengenai organisasi *Muhammadiyah Disaster Management Center*

2. Manfaat Bagi Universitas

Penelitian ini diharapkan dapat menjadi sumber referensi tambahan sebagai bahan penelitian lanjutan yang lebih mendalam pada masa yang datang khususnya yang berkaitan dengan webgis di Universitas.

3. Manfaat Bagi Masyarakat

Membantu masyarakat agar mengetahui orgasnisasis penanggulanagan bencana dibawah naungan Muhammadiyah yaitu Muhammadiyah Disaster Management Center.

F. Sistematika Penulisan

Sistematika yang digunakan dalam penyusunan proposal ini adalah sebagai berikut :

BAB I PENDAHULUAN

Dalam bab ini akan diuraikan mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, serta sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini berisikan beberapa teori-teori yang mendukung dalam pembahasan penyusunan tugas akhir ini serta bahasa pemrograman yang digunakan. sehingga memudahkan penulis dalam menyelesaikan masalah.

BAB III METODE PENELITIAN

Pada bab ini menjelaskan mengenai tahapan - tahapan yang dilalui dalam penyelesaian tugas akhir ini, yaitu tempat penelitian dan waktu penelitian, metode pengumpulan data, alat dan bahan penelitian, tahap penelitian,

metode pengujian serta gambaran desain sistem yang akan dirancang atau dibuat.

BAB IV HASIL PENELITIAN DAN PEMBAHASAN

Dalam bab ini menjelaskan mengenai rancangan sistem yang telah dibuat, metode dan teknik pengujian.

BAB V KESIMPULAN DAN SARAN

Dalam bab ini menjelaskan mengenai kesimpulan dan saran.

BAB II

TINJAUAN PUSTAKA

A. Kajian Teori

1. Muhammadiyah Disaster Management Center (MDMC)



Gambar 2 1. Logo MDMC

Muhammadiyah Disaster Management Center (MDMC) merupakan lembaga otonom di bawah Majelis Pembina Kesejahteraan Sosial (MPKS) Pimpinan Pusat Muhammadiyah yang berfokus pada kegiatan penanggulangan bencana. MDMC dibentuk sebagai respons terhadap semakin seringnya terjadi bencana alam di Indonesia serta kebutuhan akan partisipasi masyarakat sipil, khususnya ormas Islam, dalam penanganan bencana secara profesional dan sistematis.(MDMC, 2020).

MDMC atau (*Muhammadiyah Disaster Management Center*) sebagai salah satu lembaga organisasi keagamaan di bawah pimpinan pusat Muhammadiyah yang bergerak dalam menangani isu kebencanaan baik yang ada di dalam negeri maupun di luar negeri. Dalam hal ini kerjasama yang baik antar lembaga yang ada dalam perserikatan merupakan salah satu nilai penting yang harus diteladani dari MDMC, bahu-membahu melakukan kegiatan penanganan

bencana hingga MDMC menjadi lembaga organisasi keagamaan yang mumpuni dalam melakukan kegiatan kemanusiaan (Pangeran Bungsu, 2023).

a. **Susunan Personalia Anggota Pimpinan Lembaga Resiliensi Bencana Pimpinan Daerah Muhammadiyah Kota Makassar Periode 2022-2027**

Ketua : Haeril H Baharuddin, S.P

Wakil Ketua I : Agus Salim, Amk., S.Kar

Wakil Ketua II : Juma

Sekretaris : Muh Dahlan

Wakil Sekretaris : Nur Syamsir, S.Pd.I

Bendahara : Rohani, S.Pd

Bidang I : Mitigasi dan Kesiapsiagaan

Ketua : Muh.Syahril, SE

Anggota : Safaruddin

Anggota: Bakri

Anggata : Muh Ilham

Anggota : Amri Setiawan

Anggota : Faizal Halim

Anggata : Andi Qomaini

Anggota : Muh. Ridwan

Bidang II : Pendidikan dan Pelatihan

Ketua : Nursyam Hendarto

Anggota : Fadly

Anggota : Bayu

Anggata : Sultan

Anggota : Muh Yusran

Anggota : Arham Chezar Saputr

Bidang III : Tanggap Darurat dan Rehabilitasi Rekonstruksi

Ketua : Muh Yusuf Edy

Anggota : Danial

Anggota : Adiet

Anggata : Zildjian

Anggota : Ammar

Anggota : Muh. Dirham

Anggota : Supardi

Bidang IV : Jaringan dan Kerjasama

Ketua : Mulawarman

Anggota : Nurul Awalia. SH.i

Anggota : Nanda Amalia

Anggata : Muh. Nabil

Anggota : Muh Arjal

Anggota : M. Anwar

Bidang V : Emergency Medical Team

Ketua : Wahyuni Hafid, S.ST., M.Kes

:Anggota : Fauzan A.md

Anggota : Muh ikram

Anggata : Muh Yusran

Anggota : Amri Setiawan

Anggota : Muh. Ridwan Dg.Sutte

Bidang VI : Assessme

Ketua : Wahyudi

Anggota : Ayu Ashari

Anggota : Muh Fahri

Anggata : Ilham Mustafa

Anggota : Syahrul Rusli

Anggota : Andi Ichomeini

Bidang VII : Sarana Prasarana

Ketua : Ashar

Anggota : Iqbal Tawakkal

Anggota : Salman Sanjaya

Anggata : M Arfah

Anggota : Naswar

Anggota : Faisal Halim

b. Latar Belakang dan Visi Misi MDMC

Pembentukan MDMC berawal dari pengalaman Muhammadiyah dalam menangani dampak gempa bumi Yogyakarta tahun 2006. Muhammadiyah dinilai berhasil dalam membangun sistem respon bencana berbasis komunitas, sehingga kemudian pengalaman ini dikembangkan secara nasional melalui pembentukan MDMC.

Visi MDMC adalah “Terwujudnya masyarakat Indonesia yang tangguh menghadapi bencana melalui pendekatan Islam Berkemajuan.” Sedangkan misi MDMC antara lain:

1) Meningkatkan kapasitas organisasi dan masyarakat dalam pengurangan risiko bencana (PRB).
2) Menyelenggarakan tanggap darurat yang cepat, efektif, dan berkeadilan.
3) Mengembangkan sistem pemulihan dan rehabilitasi pascabencana yang berkelanjutan.
4) Membangun kemitraan strategis dengan berbagai pihak baik nasional maupun internasional.

c. Struktur Organisasi dan Jejaring

MDMC memiliki struktur organisasi dari tingkat pusat hingga ke tingkat wilayah, daerah, bahkan cabang dan ranting Muhammadiyah. Hal ini memungkinkan mobilisasi relawan dan distribusi bantuan secara cepat dan merata saat terjadi bencana.

MDMC juga memiliki jejaring relawan yang disebut dengan Relawan Muhammadiyah, yang terdiri dari individu-individu yang telah mendapatkan pelatihan penanggulangan bencana dan tanggap darurat. Para relawan ini bekerja secara sukarela namun profesional dengan mengedepankan prinsip kemanusiaan dan keikhlasan.(MDMC, 2020).

d. Pendekatan dalam Manajemen Bencana

MDMC mengadopsi pendekatan manajemen bencana berbasis komunitas (Community-Based Disaster Risk Reduction – CBDRR). Pendekatan ini bertujuan untuk meningkatkan ketahanan masyarakat lokal dengan melibatkan mereka dalam seluruh proses manajemen bencana, mulai dari pemetaan risiko, perencanaan mitigasi, pelatihan, hingga pelibatan aktif dalam fase tanggap darurat dan pemulihan.

Beberapa prinsip utama yang digunakan MDMC:

1) Humanity (Kemanusiaan): Menempatkan penyelamatan dan pemulihan manusia sebagai prioritas utama.
2) Neutrality (Netralitas): Tidak berpihak dan tidak terlibat dalam konflik politik.
3) Independence (Kemandirian): Bekerja secara independen berdasarkan nilai Islam dan prinsip Muhammadiyah.

e. Program Unggulan MDMC

1) Pelatihan dan Sertifikasi Relawan: MDMC secara berkala mengadakan pelatihan relawan bencana untuk meningkatkan kapasitas sumber daya manusia yang siap diterjunkan kapan pun diperlukan.
2) Pemetaan Risiko Bencana (*Risk Assessment*): Melakukan identifikasi daerah rawan bencana untuk menentukan prioritas program mitigasi dan kesiapsiagaan.

3) Respon Tanggap Darurat: Dalam situasi bencana, MDMC menggerakkan relawan dan logistik untuk memberikan pertolongan pertama, evakuasi, distribusi bantuan, dan layanan kesehatan.
4) Pemulihan dan Rehabilitasi: Setelah fase darurat, MDMC juga terlibat dalam pembangunan kembali infrastruktur, fasilitas pendidikan, rumah ibadah, dan penguatan ekonomi masyarakat terdampak.
5) Kolaborasi Nasional dan Internasional: MDMC telah menjalin kerja sama dengan lembaga-lembaga internasional seperti USAID, IFRC, dan lembaga lokal seperti BNPB, PMI, dan lainnya dalam pelaksanaan program penanggulangan bencana.

f. Pemanfaatan Teknologi Informasi

Seiring dengan perkembangan zaman, MDMC telah mulai mengadopsi teknologi informasi dalam pelaksanaan programnya, misalnya melalui pelaporan digital, sistem koordinasi online, serta database relawan dan logistik. Meski begitu, belum ada sistem terintegrasi khusus berbasis web yang dikhususkan untuk manajemen internal MDMC secara menyeluruh, sehingga pengembangan aplikasi E-MDMC menjadi sangat relevan dan dibutuhkan.

Pemanfaatan sistem informasi ini diharapkan dapat:

1) Mempercepat proses pelaporan dan dokumentasi kegiatan lapangan.
2) Memudahkan proses validasi data relawan dan kebutuhan logistik.
3) Menyediakan dashboard monitoring kegiatan MDMC secara real-time.

4) Menunjang pengambilan keputusan dalam kondisi darurat dengan data yang lebih akurat.(MDMC, 2020).

2. **Dasar Pemrograman**

Pemrograman merupakan proses menulis, menguji, dan memelihara instruksi (kode program) yang dapat dijalankan oleh komputer untuk menyelesaikan tugas tertentu. Dalam konteks pengembangan sistem informasi berbasis web seperti E-MDMC, pemrograman menjadi fondasi utama untuk membangun logika sistem, antarmuka pengguna, dan pengelolaan data.

Pemrograman adalah proses menulis instruksi yang dapat dijalankan oleh komputer untuk menyelesaikan tugas tertentu. Instruksi-instruksi ini ditulis menggunakan bahasa pemrograman yang memiliki aturan sintaks dan semantik tertentu. Tujuan utama pemrograman adalah untuk menciptakan perangkat lunak yang mampu menyelesaikan masalah atau memenuhi kebutuhan pengguna. Dalam pemrograman, data merupakan informasi yang dapat diolah oleh komputer. Data dapat berupa angka, teks, atau simbol lainnya. Pengelolaan data yang efektif sangat penting dalam pengembangan perangkat lunak yang efisien dan handal. (Ade Hastuty Hasyim, 2021).

a. Algoritma

Konsep dasar suatu program. Dalam pembuatan algoritmanya diperlukan daya nalar yang baik dan logis, karena algoritma merupakan susunan langkah-langkah logis penyelesaian masalah yang disusun secara sistematis dan logis, Dengan kata lain, algoritma adalah urutan langkah-langkah logis yang disusun secara sistematis untuk menyelesaikan suatu

masalah. Penyusunan algoritma memerlukan kemampuan bernalar yang baik dan logis, karena setiap langkah harus dapat ditentukan kebenarannya dan dapat dieksekusi secara berurutan.

Algoritma merupakan fondasi dalam pengembangan program komputer. Sebelum menulis kode program, seorang programmer harus merancang algoritma yang tepat untuk menyelesaikan masalah yang dihadapi. Dengan algoritma yang baik, program yang dihasilkan akan lebih efisien, mudah dipahami, dan mudah dipelihara.

Sebagai contoh, dalam kehidupan sehari-hari, resep memasak dapat dianggap sebagai algoritma. Resep tersebut berisi langkah-langkah yang harus diikuti secara berurutan untuk menghasilkan masakan yang diinginkan. Demikian pula dalam pemrograman, algoritma digunakan untuk menentukan langkah-langkah yang harus dilakukan oleh komputer untuk menyelesaikan tugas tertentu. (Ade Hastuty Hasyim, 2021).

b. Flowchart

Flowchart dalam Bahasa Indonsia diterjemahan sebagai Diagram Alir. Dari dua kata ini, maka dapat kita maka dapat kita bayangkan bahwa *flowchart* itu berbentuk diagram yang bentuknya dapat mengalirkan sesuatu. Hal ini memang benar, *flowchart* memang melukiskan suatu aliran kegiatan dari awal hingga akhir mengenai suatu langkah-langkah dalam penyelesaian suatu masalah. Masalah yang kita pelajari tentu saja masalah pemrograman dengan menggunakan komputer.

Flowchart adalah representasi grafilk dari langkah-langkah yang harus diikuti dalam menyelesaikan suatu permasalahan yang terdiri atas sekumpulan simbol, dimana masing-masing simbol mempresentasikan suatu kegiatan tertentu. *Flowchart* diawali dengan penerimaan *input,* pemrosesan input, dan diakhiri dengan penampilan *output* Dengan kata lain *flowchaert* merupakan suatu gambar yang menjelaskan urutan pembaca data, pemrosesan data, pengambilan keputusan terhadap data, penyajian hasil perosesan data

Ada dua macam *flowchart* yang menggambarkan proses dengan komputer, yaitu:

a. *Flowchart* sistem

Flowchart sistem yaitu bagan dengan simbol-simbol tertentu yang menggambarkan urutan prosedur dan proses suatu file dalam suatu media menjadi file didalam media lain, dalam suatu sistem pengolahan data.

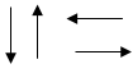
b. *Flowchart* program

Flowchart program yaitu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses dan hubungan antar proses secara mendetail didalam suatu program.

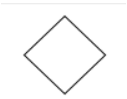
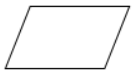
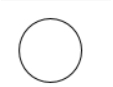
Dalam pembuatan *flowchart* program tidak ada rumus atau patokan yang bersifat mutlak. Karena *Flowchart* merupakan gambaran hasil pemikiran dalam menganalisis suatu masalah dengan komputer. Sehingga flowchart yang dihasilkan dapat bervariasi antara satu pemrograman dengan yang lainnya. Namum secara garis besar setiap pengolahan selalu terdiri atas tiga bagian utama, yaitu *Input,* Proses pengolahan dan *Output.* (Ade Hastuty Hasyim, 2021).

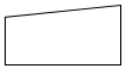
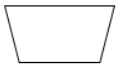
Berikut simbol-simbol beserta fungsi *flowchart*:

Tabel 2.1 Simbol Standar *Flowchart*

Simbol	Nama	Fungsi
	*Terminal Point Symbol*	*Terminal Menu*njukkan permulaan (start) atau akhir (stop) dari suatu proses.
	*Flow Direction Symbol*	Adalah simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain (*connecting line*). Simbol ini juga berfungsi untuk *Menu*njukkan garis alir dari proses.
	*Processing Symbol*	Digunakan untuk *Menu*njukkan kegiatan yang dilakukan oleh komputer. Pada bidang industri (proses produksi barang), simbol ini

Lanjutan Tabel 2.1 Simbol Standar *Flowchart*

		kan kegiatan inspeksi atau yang biasa dikenal dengan simbol inspeksi
	*Decision Symbol*	Merupakan simbol yang digunakan untuk memilih proses atau keputusan berdasarkan kondisi yang ada. Simbol ini biasanya ditemui pada *Flowchart* program.
	I*nput-Output*	Masuk menunjukkan proses *input-output* yang terjadi tanpa bergantung dari jenis peralatannya.
	*Predefined Process*	*Terdefinisi*merupakan simbol yang digunakan untuk *Menu*njukkan pelaksanaan suatu bagian prosedur (*sub*-proses).
	*Connector (On-page)*	Simbol ini fungsinya adalah untuk menyederhanakan hubungan antar simbol yang letaknya berjauhan atau rumit bila dihubungkan dengan garis dalam satu halaman

Simbol	Nama	Fungsi
	*Connector (Off-page)*	Sama seperti *on-page connector*, hanya saya simbol ini digunakan untuk menghubungkan simbol dalam halaman berbeda. label dari simbol ini dapat menggunakan huruf atau angka
	*Preparation Symbol*	Merupakan simbol yang digunakan untuk mempersiapkan penyimpanan di dalam storage.
	*Manual Input Symbol*	Digunakan untuk menunjukkan input data secara manual menggunakan online keyboard.
	*Manual Operation Symbol*	Yang digunakanuntuk menunjukkankegiatan/proses yang tidak dilakukan oleh komputer.
		Jika Anda menemukan simbol ini ıt berasal dari dokumen uk kertas, atau output yang perlu dicetak di atas kertas.
	*Multiple Documents*	Sama seperti *document symbol* hanya saja dokumen yang digunakan lebih dari satu dalam simbol ini
	*Display Symbol*	Adalah simbol yang menyatakan penggunaan peralatan output, seperti layar monitor, printer, plotter dan lain sebagainya
	*Delay Symbol*	Digunakan untuk *Menu*njukkan proses delay (*Menu*nggu) yang perlu dilakukan. Seperti *Menu*nggu surat untuk diarsipkan

Lanjutan Tabel 2.1 Simbol Standar *Flowchart*

3. **UML (Unified Modelling Language)**

Unified Modelling Language (UML) adalah sebuah "bahasa" yang telah menjadi standar dalam industri untuk *visualisasi,* merancang dan mendokumentasikan sistem piranti lunak. *UML* menawarkan sebuah standar untuk merancang model sebuah sistem. *UML* adalah sekumpulan alat yang digunakan untuk melakukan abstraksi terhadap seb uah sistem atau perangkat lunak berbasis objek. *UML* merupakan singkatan dari *Unified Modeling Language*. *UML* juga menjadi salah satu cara untuk mempermudah pengembangan aplikasi yang berkelanjutan.

Aplikasi atau sistem yang tidak terdokumentasi biasanya dapat menghambat pengembangan karena developer harus melakukan penelusuran dan mempelajari kodeprogram *UML* juga dapat menjadi alat bantu untuk transfer ilmu tentang sistem atau aplikasi yang akan dikembangkan dari satu *developer* ke *developer* lainya. Tidak hanya antar *developer* terhadap orang bisnis dan siapapun dapat memahami sebuah sistem dengan adanya *UML*.

UML diciptakan oleh *Object Management Group* yang diawali dengan versi 1.0 pada Januari 1997. Dalam pengembangan berorientasi objek ada beberapa prinsip yang harus dikenal: *Object, Class, Abstraction, Encapsulation, Inheritance* dan *Polymorphism*(Kroenke. 2018).

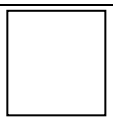
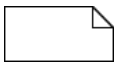
Dalam UML sendiri terdapat beberapa diagram yaitu :

a) *Use Case* Diagram

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan

bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-create sebuah daftar belanja, dan sebagainya. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. *Use case* adalah suatu urutan interaksi yang saling berkaitan antara sistem dan aktor. *Use case* dijalankan melalui cara menggambarkan tipe interaksi antara user suatu program (sistem) dengan sistemnya sendiri. (Gilang Ramadhan.2023).

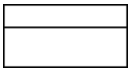
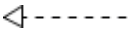
Tabel 2. 2. Use Case Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		*Actor*	Menspesifikasikan himpuan peran yang pengguna mainkan ketika berinteraksi dengan *use case*.
2		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (*independent*) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (*independent*).
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		*Sistem*	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 aktor.
9		*Collaboration*	Interaksi aturan-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

Class diagram adalah visual dari struktur sistem program pada jenis-jenis yang di bentuk. Class diagram merupakan alur jalannya database pada sebuah sistem. *Class* diagram merupakan penjelasan proses database dalam suatu program. Dalam sebuah laporan sistem maka class diagram ini wajib ada.

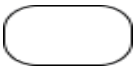
Tabel 2.3 Simbol Class Diagram

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-banar dilakukan oleh suatu *object.*
6		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya.

c) *Activity* diagram

Adalah sesuatu yang menjelaskan tentang alir kegiatan dalam program yang sedang dirancang, bagaimana proses alir berawal, keputusan yang mungkin terjadi, dan bagaimana sistem akan berakhir.

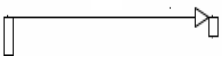
Tabel 2.4 Simbol *Activity* Diagram

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 berubah menjadi beberapa aliran.

d) *Sequence* Diagram

adalah suatu diagram yang menjelaskan interaksi objek dan menunjukkan (memberi tanda atau petunjuk) komunikasi diantara objek-objek tersebut.

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 tentang aktifitas yang terjadi

Lanjutan Tabel 2.5 Simbol *Sequence* Diagram

3		*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi tentang aktifitas yang terjadi

4. **HTML (Hypertext Markup Language)**

HTML(Hypertext Markup Language) merupakan kerangka dasar untuk tampilan pada browser, menggunakan bahasa markah sebagai dasar pengkodean. Hingga pada saat ini *HTML* menggunkan versi ke 5.

Endra & Aprilita (2018) mengemukakan bahwa "*HTML* atau *Hypertext Markup Languange* merupakan salah satu bahasa yang biasa digunakan oleh pengguna dalam membuat tampilan yang digunakan oleh *web application.*". HTML *(Hypertext Markup Language)* adalah bahasa standard yang digunakan untuk menampilkan halaman web (Hidayatullah dan Kawistara, 2019).

Sama seperti hal teknis lainnya dalam dunia web, bahasa markup ini juga punya kelebihan dan kekurangannya.

Berikut kelebihan HTML:

a. Digunakan secara luas dan memiliki banyak sumber serta komunitas yang besar.
b. Dijalankan secara alami di setiap web browser.
c. Mudah dipelajari.
d. *Open-source* dan sepenuhnya gratis.
e. Rapi dan konsisten.

f. Menjadi standar resmi web, dikelola oleh (W3C).

g. Integrasi mudah dengan bahasa *backend*, seperti PHP dan Node.js.

Kekurangan HTML adalah:

a. Umumnya digunakan untuk halaman web statis. Untuk fitur dinamis, Anda bisa menggunakan *JavaScript* atau bahasa backend, seperti PHP.

b. Tidak bisa menjalankan *logic*. Alhasil, semua halaman web harus dibuat terpisah meskipun menggunakan elemen yang sama, seperti *header* dan *footer*.

c. Fitur baru tidak bisa digunakan dengan cepat di sebagian browser.

d. Perilaku browser terkadang sulit diprediksi. Misalnya, *browser* lama tidak selalu bisa merender tag yang lebih baru.

a. PHP (Hypertext Preprocessor)

PHP pertama kali dibuat oleh Rasmus Lerdorf, yang diberi nama FI *(Form Interpreted)* dan digunakan untuk mengelola *structure* dari web. Pada perkembangannya, kode tersebut dirilis umum sehingga mulai banyak dikembangkan oleh *developer* di seluruh dunia (Anhar, 2010). PHP merupakan singkatan dari "*Hypertext Preprocessor*", PHP adalah bahasa pemrograman *open source* yang dieksekusi pada server dan bebas untuk di download dan digunakan. Bahasa pemrograman ini merupakan dasar pembuatan aplikasi blog terkenal yaitu *Wordpress* dan jejaring sosial terbesar yaitu *Facebook* serta mudah untuk programmer baru. File yang diperoleh dari pemrograman PHP memiliki extensi “php”. Di dalam file php dapat berisi text, HTML, CSS,

Javascript dan code PHP. Kode PHP ini dapat dieksekusi pada server dan hasilnya akan ditampilkan kembali ke browser sebagai HTML (Rusli et al, 2019).

Fungsi utama PHP adalah sebgai kumpulan script atau bahasa pemrograman untuk dapat dikumpulkan dan evaluasi hasil survey atau segala bentuk server basis data, dan menghasilkan efek tindak lanjut pada tahap selanjutnya. Efek stripe PHP ini merupakan bentuk yang akan berkomunikasi dengan database, mengumpulkan dan mengklasifikasikan informasi dan kemudian menampilkan operasi skrip lainnya ketika pengunjung situs membutuhkannya (menampilkan informasi berdasarkan permintaan pengguna) (Hidayatullah, 2019).

PHP adalah script pemrograman yang terletak dan dieksekusi di server untuk menerima, mengolah dan menampilkan information dari dan ke sebuah situs. Information akan diolah ke sebuah data set server untuk kemudian hasilnya akan di tampilkan di program sebuah situs (Marhamelda, 2019).

Ada beberapa keuntungan menggunakan bahasa pemrograman PHP adalah sebagai berikut :

1) Open source : hal ini dikembangkan dan dikelola oleh sekelompok besar pengembang PHP, ini akan membantu dalam menciptakan sebuah komunitas dukungan, perpustakaan ekstensi berlimpah.
2) Kecepatan : ini relatif cepat karena menggunakan banyak sumber daya sistem.

3) Mudah digunakan: menggunakan C seperti sintaks, sehingga bagi mereka yang terbiasa dengan C, itu sangat mudah bagi mereka untuk mengambil dan sangat mudah untuk membuat script website.
4) Stabil : karena dikelola oleh banyak pengembang, sehingga ketika bug yang ditemukan dapat dengan cepat diperbaiki.
5) Dukungan library yang kuat : anda dapat dengan mudah menggunakan moduk fungsional yang anda butuhkan seperti PDF, Grafik, dll.
6) *Modul Built-in* koneksi *database* : anda dapat terhubung ke database dengan mudah menggunakan PHP, karena banyak situs yang data atau konten didorong, jadi kita akan menggunakan database sering, ini akan sangat mengurangi waktu pengembangan aplikasi web.
7) Dapat dijalankan di banyak flatform, termasuk *Windows, Linux* dan *Mac*, mudah bagi pengguna untuk menemukan penyedia layanan *hosting* (Rusli et al., 2019).

b. XAMPP

Menurut Aryanto (2019) "*XAMPP* merupakan sebuah aplikasi perangkat lunak pemrograman dan database yang didalamnya terdapat berbagai macam aplikasi seperti; *Appache HTTP Server*, database MYSQL, bahasa pemrograman PHP seta Perl.". *XAMPP* berperan sebagai server app pada komputer anda. *XAMPP* juga dapat disebut sebuah *Cpanel server virtual*, yang dapat membuat anda melakukan *preview* sehingga dapat memodifikasi website tanpa harus online atau terakses dengan internet.

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost),* yang terdiri atas program *Apache HTTP* Server, *MySQL* database, dan penerjamah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), *Apache, MySQL, PHP* dan *Perl* (Aryanto. 2019).

c. MySQL

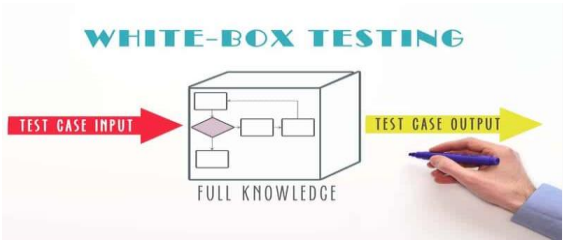
MySQL merupakan salah satu perangkat lunak sistem pengelola basis data DBMS (*Data Base Management System*). MySQL merupakan sebuah hubungan Data Base Management System (DBMS) yang membantu sebuah model data yang terdiri atas kumpulan hubungan nama (*named relation*) (M. Ramaddan Julianti et al., 2018). Menurut (Munawir, 2019) menyatakan bahwa "MySQL adalah software atau program database server". MySQL merupakan software yang tergolong database server dan bersifat opensource. Opensource menyatakan bahwa software ini dilengkapi dengan sourcecode (kode yang dipakai untuk membuat MySQL), selain tentu saja bentuk executable-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi, dan bisa diperoleh dengan cara mengunduh di internet secara gratis (Murdiono & Widodo, 2020).

MySQL adalah database open source terpopuler di dunia. Dengan kinerja, kehandalan dan kemudahan penggunaan yang terbukti, MySQL telah

menjadi pilihan database terdepan untuk aplikasi berbasis web, yang digunakan oleh properti web profil tinggi termasuk Facebook, Twitter, YouTube, Yahoo! dan banyak lagi. Kata “SQL” dari "MySQL" adalah singkatan dari "Structured Query Language". SQL adalah bahasa standar yang paling umum digunakan untuk mengakses database (Arfianto & Nugrahanti, 2018). Menurut (Hendini, 2016), *database* adalah sekumpulan table-tabel yang saling berelasi, relasi tersebut bisa ditunjukkan dengan kunci dari tiap tabel yang ada. Satu database menunjukkan satu lingkup perusahaan atau instansi. *Database* atau basis data merupakan Struktur penyimpanan data untuk menambah, mengakses dan memperoses data yang disimpan dalam sebuah database komputer, diperlukan sistem manajemen *database* seperti MYSQL Server (Robihaini et al., 2022).

5. **Metode Pengujian White Box**

White box testing adalah suatu pengujian oleh software atau aplikasi dengan melihat modul untuk menganalisis kode program ada yang salah atau tidak. Bila suatu modul yang diproduksi tidak memenuhi syarat, kode akan dikompilasi ulang dan periksa lagi hingga mencapai apa yang diharapkan dalam arti bahwa white box testing menguji dengan cara melihat pure code dari suatu perangkat lunak atau aplikasi yang diuji tanpa mempedulikan tampilan atau UI dari aplikasi tersebut.



Gambar 2. 2 *White Box Testing*

a. Teknik

Pada white box testing ada beberapa teknik yang digunakan. Berikut akan kami sajikan :

1) Basis path testing adalah metode yang memungkinkan perancang tast case untuk membuat pengukuran kompleksitas logika dari rancangan prosedural dan menggunakan pengukuran untuk mendefinisikan himpunan basis dari jalur eksekusi.

2) Flow graph merupakan notasi sederhana guna mempresentasi control flow.

3) Cyclomatic complexity yang digunakan guna mengetahui jumlah path yang perlu dicari. Di mana cyclomatic complexity merupakan metric perangkat lunak yang menyediakan ukuran kuantitatif dari kompleksitas logika program.

4) Branch coverage merupakan pengujian yang didesain agar masing-masing branch code yang paling tidak diuji satu kali.

5) Loop testing merupakan testing pada berbagai looping/perulangan yang terdapat pada program, seperti for, while, dan do-while. Dengan pengujian ini Anda bisa mengecek kondisi perulangan, sudahkah berjalan dengan benar ataupun tidak.

6) Condition coverage merupakan teknik pengujian yang bertujuan guna melakukan uji semua kode untuk menghasilkan value FALSE atau TRUE.

Tester bisa memastikan software dalam beroperasi dan menghasilkan output sesuai input user.

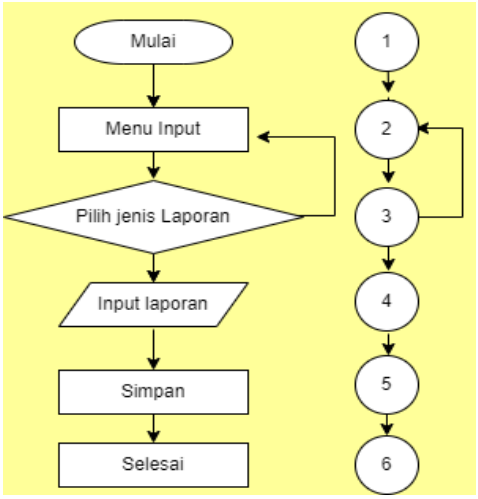
7) Multiple condition coverage merupakan teknik yang digunakan guna menguji semua kombinasi beragam kode yang digunakan dalam berbagi kondisi.

8) Statement coverage merupakan teknik yang minimal 1 kali dilakukan guna menguji masing-masing statement pada software. Anda bisa mengetahui beberapa code yang error dengan pengujian ini sehingga bisa segera memperbaikinya.

b. Trik Pengujian

White box testing merupakan pengujian dengan dasar pada pengecekan di dalam detail desain dari suatu rancangan pemrograman untuk membagi pengujian. Dalam membuat pengujian white box maka tester perlu memiliki pemahaman terkait pemrograman. Sebab untuk memahami kode sumber yang dipelajari dan harus memiliki pandangan global tentang fungsi aplikasi serta elemen yang dirancangnya. Metode white test ini dapat diterapkan untuk tes integrasi, tes unit dan tes sistem.(Hasan, 2023).

c. Contoh *White Box*



Gambar 2.3 Contoh *White box*

Hasan (2023) Pengujian white box adalah pengujian yang didasarkan pada pengecekan detail perancangan, menggunakan struktur kontrol dari desain program secara procedural untuk membagi pengujian ke dalam beberapa kasus pengujian. Pemetaan flowchart ke dalam flowgraph dan proses perhitungan V(G) terhadap perangkat lunak dapat dilihat pada penjelasan berikut :

Diketahui :

N = 6 E = 6 R = 2

Penyelasaian :

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

= (6 – 6) + 2

= 2

Independent Path :

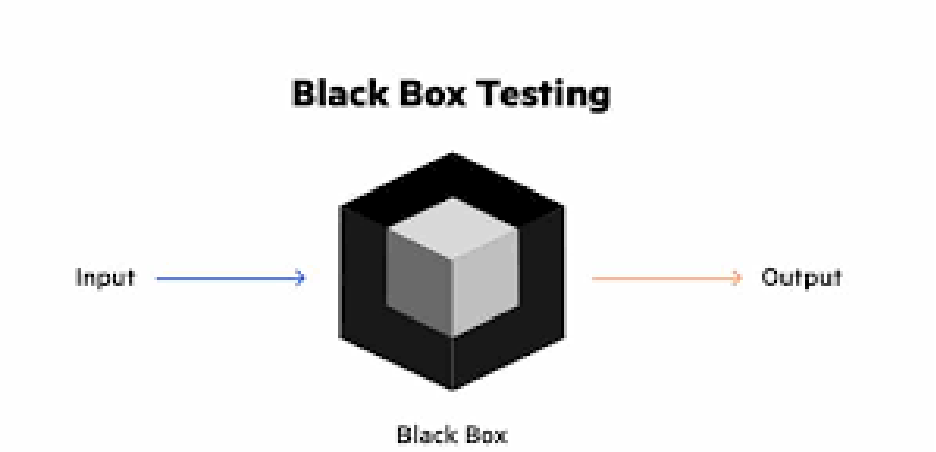
Path 1 = 1 – 2 – 3 – 4 – 5 – 6

Path 2 = 1 – 2 – 3 – 2 – 3 – 4 –5 – 6

Tabel 2.7. *White Box* Perhituangan

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

6. **Metode Pengujian Black Box**



Gambar 2. 3 *Black Box Testing*

Pengujian black box adalah pengujian yang dilakukan semata-mata dengan memantau hasil eksekusi melalui data pengujian dan memvalidasi fungsionalitas perangkat lunak. Jadi, mirip dengan melihat kotak hitam, kita hanya bisa melihat tampilan luarnya saja dan tidak tahu apa yang ada di balik bungkus hitamnya. Mirip dengan pengujian black box, hanya mengevaluasi tampilan luar (antarmuka) dan fungsionalitasnya. Tanpa mengetahui apa yang terjadi dalam proses detail (hanya mengetahui input dan output).

a. Jenis Pengujian

Pada suatu black box testing terdapat tiga jenis pengujian, berikut akan kami jelaskan :

1) Functional testing

Merupakan suatu proses pengujian pada fitur spesifik atau fungsi pada suatu software. Pengujian ini bisa dilakukan guna memastikan bahwa user tidak dapat masuk pada aplikasi yang diuji tanpa informasi sehingga program tersebut terjaga keamanannya.

2) Non functional testing

Memiliki fungsi yang tidak kalah dari functional testing, di mana Non functional testing merupakan suatu pengujian pada beberapa aspek tambahan non fungsional. Bila functional testing menguji software menjalankan suatu tugas atau peringkat.

3) Regression testing

Jenis ini merupakan pengujian yang dilakukan pada software untuk mengamati apakah terjadinya kemunduran atau regresi ketika software diperbaharui. Dalam regression test dilakukan pengecekan aspek non fungsional maupun fungsional.

b. Teknik Menguji Black Box Dengan Software

Tulisan di atas sudah menjelaskan apa itu black box testing dan contohnya. Namun black box testing memiliki beberapa teknik dalam menguji suatu software. Berikut akan kami jelaskan :

1) Equivalence partitioning merupakan cara kerja dengan melakukan pembagian atau partition menjadi beberapa partisi dari input data.

2) Fuzzing merupakan teknik untuk mencari gangguan atau bug dari software dengan cara injeksi data yang terbilang sesi semi otomatis atau cacat.

3) Boundary value analysis merupakan teknik yang lebih fokus pada boundary di mana jika ada error dari sisi atau luar baik minimum maupun maksimum pada pada software yang didapat.

4) Orthogonal array testing merupakan teknik yang digunakan jika input domain yang relatif ukurannya kecil tetapi cukup berat jika digunakan dalam skala besar.

5) Cause effect graph merupakan teknik yang digunakan di mana graphic sebagai pacuannya. Dalam teknik ini di mana graphic menggambarkan relasi antara penyebab dan efek dari error tersebut.

6) State transition merupakan testing yang digunakan untuk mengecek apakah kondisi dari navigasi dan mesin dalam bentuk grafik. (Sumber : https://itbox.id/blog/black-box-testing-adalah/).

c. Contoh Pengujian Black Box

Pengujian Black Box didasarkan pada detail aplikasi, fungsi – fungsi yang ada pada aplikasi dan kesesuaian alur fungsi dengan proses yang diinginkan oleh pengguna atau user, pengujian ini tidak melihat dan menguji Logika program (Hasan, 2023).

Tabel 2.6 Contoh Tabel Pengujian *Black Box*

Test Faktor	Hasil	Kesimpulan
Login	✔	Sukses, karena ketika *users* memasukkan *user* ID dan *Password*, akan masuk di halaman Utama
Screen Shot		
		

B. Kajian Hasil Penelitian Terdahulu

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

1. Penelitian yang dilakukan oleh: Sutte A.H.Z (2017) Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Parepare yang berjudul, "Aplikasi Pendaftaran dan Pengolahan Data Ko-Kurikuler Tapak Suci Berbasis Web pada LP2K Universitas Muhammadiyah Parepare" Penelitian ini mengembangkan aplikasi berbasis web untuk mendukung proses pendaftaran dan pengelolaan data kegiatan ko-kurikuler Tapak Suci di lingkungan LP2K Universitas Muhammadiyah Parepare. Tujuan utama dari sistem ini adalah meningkatkan efisiensi dalam manajemen data anggota, kehadiran, jadwal latihan, serta pencatatan hasil evaluasi kegiatan. Teknologi berbasis web dipilih agar sistem dapat diakses secara mudah oleh mahasiswa maupun pengelola secara daring.

2. Penelitian yang dilakukan oleh: Rahmah F.A (2022) Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Yogyakartayang berjudul, "Aplikasi Manajemen Data Relawan dan Logistik Muhammadiyah Disaster Management Center (MDMC) Berbasis Web" Penelitian ini secara spesifik mengembangkan aplikasi internal MDMC untuk mempermudah pendataan relawan, logistik, dan lokasi kejadian bencana. Teknologi yang digunakan adalah PHP dan MySQL dengan pendekatan metodologi waterfall.

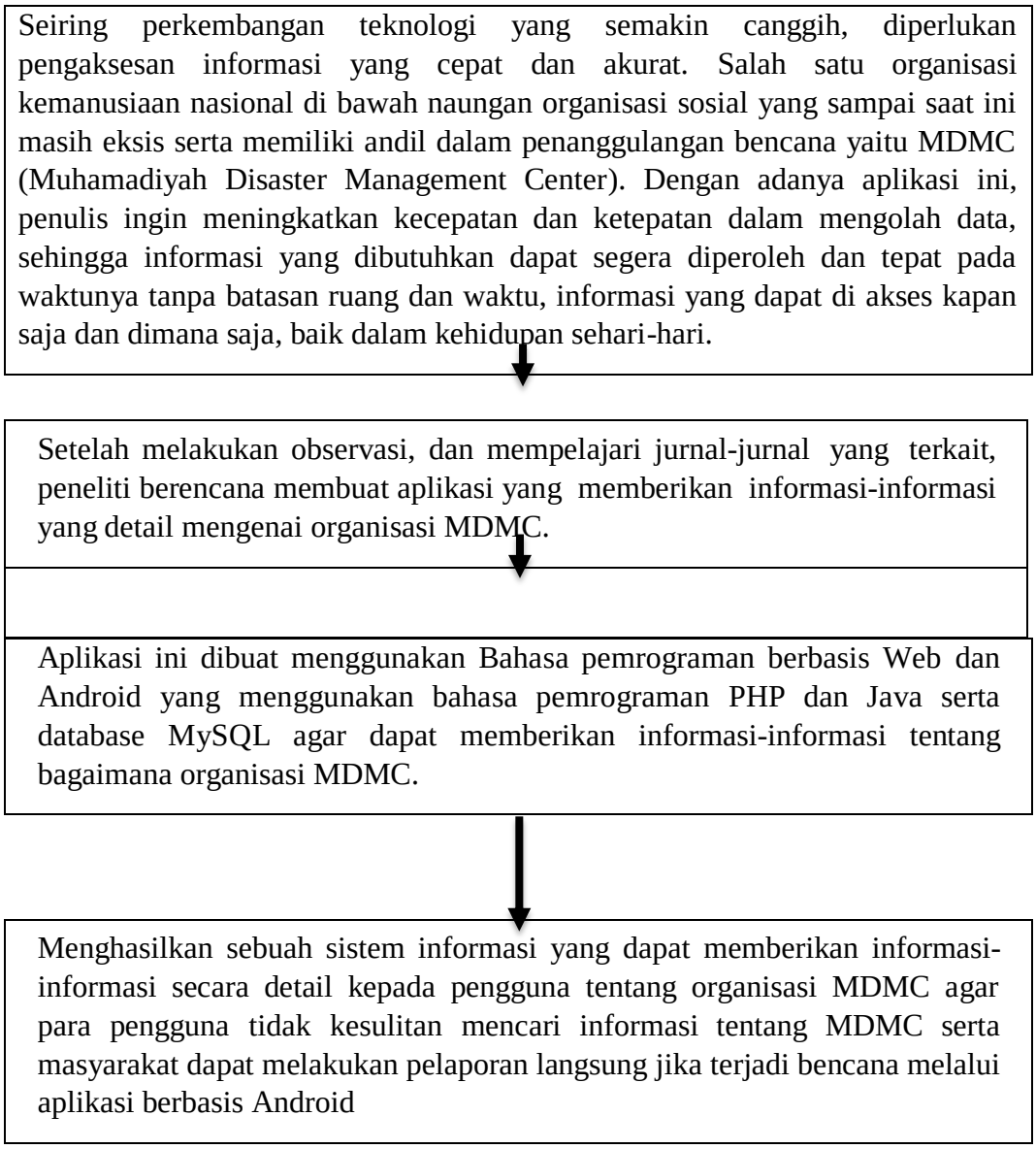
Perbandingan dari penelitian sebelumnya adalah Penelitian yang dilakukan oleh Sutte A.H.Z (2017) di Universitas Muhammadiyah Parepare mengembangkan aplikasi pendaftaran dan pengolahan data kegiatan ko-kurikuler Tapak Suci berbasis web. Fokus utama dari penelitian tersebut adalah mempermudah proses administrasi kegiatan mahasiswa, seperti pendaftaran anggota, pencatatan kehadiran, dan pengelolaan jadwal. Sementara itu, penelitian oleh Rahmah F.A (2022) di Universitas Muhammadiyah Yogyakarta mengembangkan sistem manajemen data relawan dan logistik untuk Muhammadiyah Disaster Management Center (MDMC), dengan tujuan mempercepat pendataan dan penyaluran logistik saat terjadi bencana. Penelitian tersebut menggunakan pendekatan waterfall dan teknologi PHP serta MySQL. Berbeda dengan kedua penelitian tersebut, penelitian yang akan dilakukan bertujuan untuk mengembangkan sistem informasi E-MDMC berbasis web yang lebih komprehensif. Aplikasi ini tidak hanya mencakup pendataan relawan dan logistik, tetapi juga sistem pelaporan kejadian bencana, manajemen dokumentasi, serta penyebaran informasi ke masyarakat secara daring. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi yang lebih luas dalam pengelolaan informasi kebencanaan di lingkungan Muhammadiyah.

Kesimpulan dari penelitian terdahulu dan penelitian yang akan dibuat berbeda dari lokasi penelitian dan efisiensi dalam pengelolaan data baik untuk kegiatan ko-kurikuler maupun aktivitas kebencanaan seperti yang dilakukan oleh LP2K dan MDMC. Masing-masing penelitian berfokus pada sistem sederhana dengan ruang lingkup terbatas, seperti pendaftaran anggota atau manajemen

logistik internal. Sementara itu, penelitian yang akan dilakukan bertujuan untuk membangun aplikasi E-MDMC yang lebih komprehensif dan terintegrasi, mencakup manajemen data relawan, logistik, pelaporan bencana, serta publikasi informasi ke masyarakat. Hal ini menunjukkan adanya perkembangan signifikan dalam penerapan teknologi informasi untuk mendukung respon kebencanaan yang lebih terstruktur dan menyeluruh di lingkungan Muhammadiyah.

C. Kerangka Pikir

Untuk memahami kerangka pikir penelitian ini dapat dilihat dibawah ini.



Gambar 2. 5 Kerangka Pikir

BAB III

METODE PENELITIAN

A. Jenis Penelitian

Berdasarkan jenis penelitian di atas maka peneliti memilih pendekatan penelitian yaitu metode kualitatif. Menurut (Hermawansyah & Nur, 2019) berpendapat penelitian kualitatif dilaksanakan untuk membangun pengetahuan melalui pemahaman dan penemuan, menggunakan observasi terstruktur dan tidak terstruktur serta interaksi komunikasi sebagai alat mengumpulkan data terutama wawancara mendalam.

B. Lokasi dan Waktu

Lokasi penelitian dilaksanakan di Kantor MDMC (*Muhammadiyah Disaster Management Center*), Jalan Muhammadiyah nomor 8 Kota Parepare dan waktu penelitian dilakukan selama kurang lebih 2 (dua) bulan.

C. Alat dan Bahan

Dalam pelaksanaan penelitian ini dibutuhkan beberapa bahan dan komponen untuk merealisasikan rancangan sistem yang terbagi dalam dua bagian yaitu perangkat keras (Hardware) dan perangkat lunak (Software). Adapun alat yang dibutuhkan dapat dilihat pada Tabel berikut.

Tabel 3 1. Spesifikasi Perangkat Keras *(Hardware)*

No	Jenis (Hardware)	Spesifikasi
1	Laptop	Asus 14 Inc
2	Memori	4 GB

3	Harddisk	1 TB
4	I/O	Monitor,Keyboard,Mouse

Tabel 3 2. Spesifikasi Perangkat Lunak *(Software)*

No	Jenis (Software)	Spesifikasi
1	OS	Windows 10
2	Aplikasi Pemrograman	*Visual Studio Code,* XAMPP,*Chrome,* Android Studio
3	Bahasa Pemrograman	PHP, HTML, JavaScript
4	Database	MySQL

Adapun bahan yang dibutuhkan pada penelitian ini adalah sebagai berikut.

1. Data profil dan sejarah MDMC Kota Parepare
2. Data anggota dan struktur organisasi MDMC Kota Parepare
3. Data-data program MDMC Kota Parepare serta dokumentasi kegiatan

D. Metode Pengumpulan Data

Untuk memperoleh data-data yang dibutuhkan dalam rangka melakukan penelitian, maka penulis mengumpulkan data melalui beberapa cara yaitu :

1. Analisis Data

   Menganalisa data-data yang sebelumnya telah dikumpulkan.

2. Perancangan Program

   Sebagai pedoman dalam penulisan program atau kode-kode agar berjalan sesuai rencana.

3. Uji Coba Program

   Pengujian program dilakukan untuk memastikan bahwa program yang dibuat dapat berjalan dengan baik.

4. Evaluasi

   Sistem yang telah selesai dibangun perlu adanya evaluasi untuk menemukan kelemahan yang terdapat pada program yang telah dibangun tadi, yang nantinya bisa digunakan sebagai acuan untuk memperbaiki program sehingga lebih sempurna.

E. Jenis Data

Data yang digunakan dalam penelitian ini berupa data-data yang telah dikumpulkan melalui Penelitian Pustaka (*Library Research*). Adapun jenis data primer dan data sekunder yang relevan dengan masalah yang akan dibahas.

a. Data primer

   Data Primer adalah data yang berasal atau data yang diperoleh langsung dari sumber data dan pengetahuan.

b. Data Sekunder

   Data sekunder adalah data yang diperoleh tidak secara langsung dari objek penelitian. Peneliti mendapatkan data yang sudah jadi dari internet, website dan jurnal.

F. Tahapan Peneletian

Tahapan penelitian yang dimaksud dalam penelitian ini ada beberapa tahapan yaitu persiapan penelitian, pengumpulan data, analisis perancangan, pengujian dan implementasi. Adapun Uraian dari tahapan tersebut adalah sebagai berikut :

1. Persiapan Penelitian

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

2. Pengumpulan Data

Pada tahap ini peneliti melakukan observasi dengan peninjauan, pencatatan dan pengamatan langsung di tempat penelitian.

3. Analisis

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

4. Perancangan

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

5. Pengujian

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

6. Implementasi

Setelah pada perancangan tidak terdapat kekurangan maka aplikasi siap untuk di gunakan oleh *User*.

G. Metode Pengujian Aplikasi

Beberapa *test-case* harus dilaksanakan dengan beberapa perbedaan strategi, query, atau jalur navigasi yang mewakili penggunaan sistem yang *tipical*, kritis atau abnormal. Isu kunci pada pengembangan sistem adalah pemilihan sekelompok *test-case* yang cocok, sekecil dan secepat mungkin, untuk meyakinkan perilaku sistem secara detail. Pengujian harus mencakup *unit testing*, yang mengecek validasi dari prosedur dan fungsi-fungsi secara independen dari

komponen sistem yang lain. Kemudian modul testing harus menyusul dilakukan untuk mengetahui apakah penggabungan beberapa unit dalam satu modul sudah berjalan dengan baik, termasuk eksekusi dari beberapa modul yang saling berelasi, apakah sudah berjalan sesuai karakteristik sistem yang diinginkan.

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

Ada 2 metode untuk melakukan unit testing, yaitu :

1. *White Box Testing*

   Uji coba *White Box Testing* merupakan metode perancangan *testcase* yang mengunakan struktural untuk mendapatkan *testcase*, test ini digunakan untuk meramal cara kerja perangkat lunak secara rinci kepada *logic path* (jalur logika), perangkat lunak di tes dengan kondisi dan perulangan secara fisik.

   Contoh pengujian *White Box Testing* ini merupakan peringatan ketika *User* menginputkan password *User* yang salah, untuk kesalahan semacam ini akan memberikan suatu in*Form*asi kepada *User* mengenai kesalahan yang di lakukan.

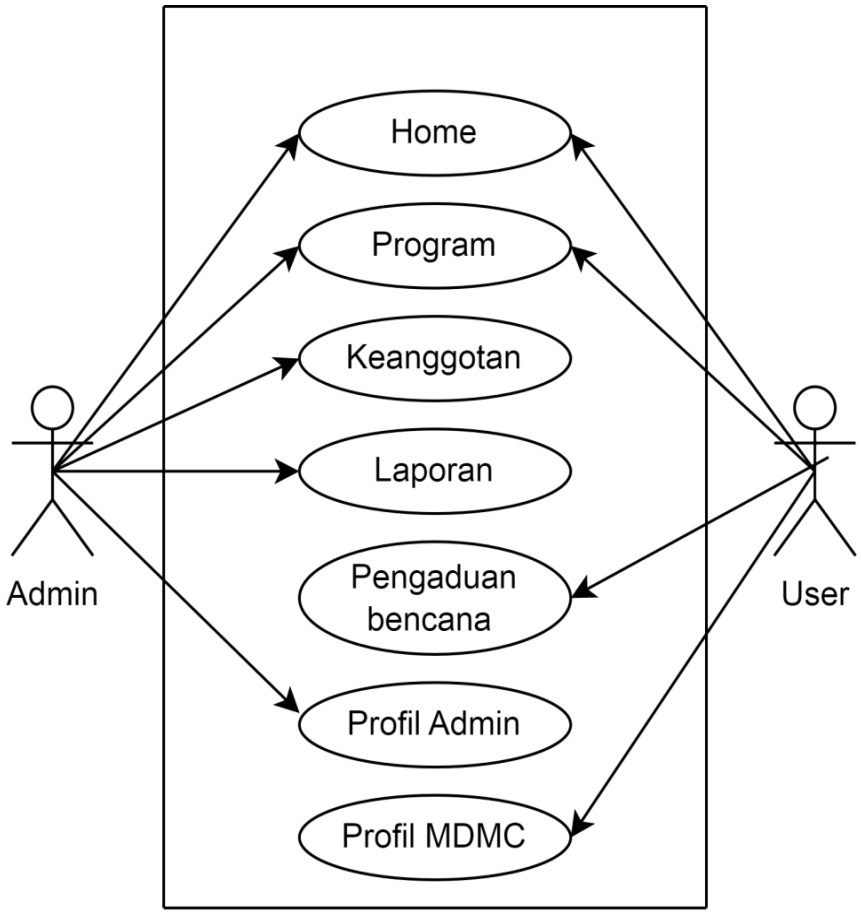
2. *Black Box*

   Berdasarkan hasil uji coba yang dilakukan, seluruh navigasi dan tombol fasilitas program lainnya serta proses yang di jalankan tidak terjadi kesalahan, tetapi aplikasi mempunyai aturan-aturan yang sudah di tetapkan dan harus di ikuti karena apabila di hiraukan maka sistem akan menolak perintah yang tidak sesuai seperti kesalahan ketika *User* belum menginput data yang harusnya di input sesuai ketentuan sistem yang di jalankan dan sistem memberikan in*Form*asi kepada *User* karena data yang

ingin diproses belum lengkap atau tidak memenuhi ketentuan untuk proses selanjutnya.

H. Rancangan Sistem

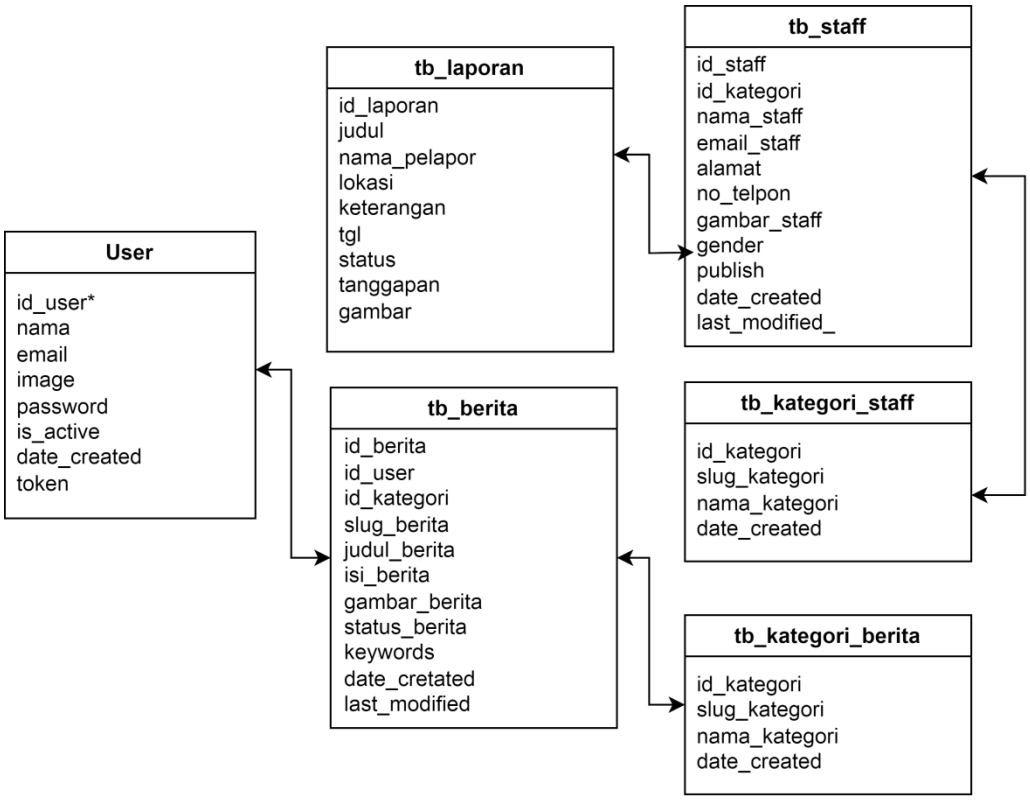
1. Sistem yang diusulkan



Gambar 3 1. Sistem yang diusulkan

Pada gambar 3.1 Sistem yang diusulkan terdapat dua aktor admin dan user, Admin dapat mengakses menu program, keanggotan, laporan serta profil admin sedangkan user dapat mengakses menu program, pengaduan bencana dan profil MDMC.

2. Relasi Antar Tabel



Gambar 3.2 Relasi Antar Tabel

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

A. Hasil Penelitian

Selama melakukan penelitian di Kantor MDMC (*Muhammadiyah Disaster Management Center*), Jalan Muhammadiyah nomor 8 Kecamatan Soreang Kota Parepare, Sulawesi Selatan, dalam membangun aplikasi elektronik MDMC, didapatkanlah hasil penelitian antara lain. Data Program, Data Keanggotaan, Data Laporan, serta Profil MDMC.

B. Pembahasan

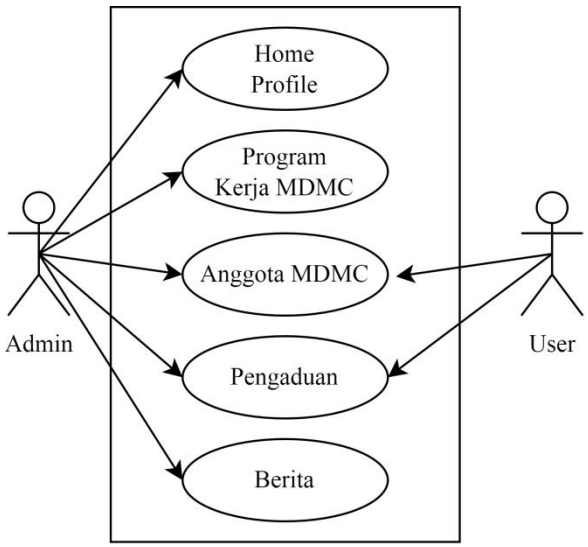
1. Analisis Aliran Data UML

a. ***Use Case Diagram***

Use Case Diagram atau diagram *use case* merupakan pemodelan untuk memodelkan kelakuan (*behavior*) sistem informasi yang dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi. Use case merupakan suatu bentuk diagram yang menggambarkan fungsionalitas yang diharapkan dari sebuah sistem dilihat dari perspektif pengguna di luar sistem.

Use case juga dapat digunakan untuk merepresentasikan interaksi yang terjadi antara aktor dengan proses sistem yang dibuat. Pada perancangan *use case* juga terdapat skenario, yaitu langkah yang menerangkan urutan kejadian

antara pengguna dengan sistem. Diagram *use case* pada aplikasi elektronik MDMC pada penelitian ini digambarkan seperti berikut ini:



Gambar 4 1. Use Case Diagram.

b. Tabel *Use Case*

Penjabaran use case diagram di atas didefinisikan atas definisi aktor dan definisi use case. Definisi aktor sebagai berikut :

Tabel 4 1. Definisi Aktor dalam Sistem.

No	Aktor	Keterangan
1	*User*	*User* merupakan aktor yang berperan sebagai pengguna aplikasi elektronik MDMC.
2	Admin	Admin merupakan aktor yang memiliki hak akses lebih tinggi dibanding *User* karena Admin dapat mengelola program, keanggotaan, serta laporan.

Sedangkan untuk penjelasan definisi setiap use case sebagai berikut:

Tabel 4 2. Definisi Use Case Sistem

No	*Use Case*	Keterangan
1	Home Profil	Menampilkan halaman isi home profiladmin
2	Program Kerja MDMC	Menampilkan halaman program kerja MDMC
3	Anggota MDMC	Menampilkan halaman anggota MDMC
5	Pengaduan	Manampilkan halaman pengaduan
7	Berita	Menampilkan halaman berita

Dalam use case yang telah didefinisikan di atas, terdapat beberapa use case yang memiliki alur proses yang hampir sama, sehingga dalam penjabaran selanjutnya akan didefinisikan secara ringkas untuk memudahkan pemahaman proses tiap use case. Berikut adalah skenario use case yang telah didefinisikan sebagai berikut :

1) Use Case Home.

Aktor : Admin

Deskripsi : Sebuah kegiatan untuk menampilkan menu-menu aplikasi.

Pre-condition : Admin dan user berada di halaman utama.

Post-condition : - .

Tabel 4 3. Skenario Use Case Home.

Aksi Aktor	Reaksi Sistem
1. Masuk kehalaman utama aplikasi	
	2. Menampilkan menu-menu aplikasi.
3. Memilih salah satu menu.	
	4. Jika menu yang dipilih valid, maka menu tersebut akan ditampilkan di form baru.
Jika menu yang diakses tidak valid, maka menu tidak dapat diakses.	

3) Use Case Program

Aktor : Admin

Deskripsi : Sebuah kegiatan untuk menampilkan program.

Pre-condition : Admin dan user berada di halaman program..

Post-condition : -

Tabel 4 4. Skenario Use Case Program Kerja MDMC

Aksi Aktor	Reaksi Sistem
1. Memilih menu program	
	2. Menampilkan form program

4) Skenario Use Case Anggota

Aktor : Admin dan User

Deskripsi : Sebuah kegiatan untuk melihat keanggotaan.

Pre-condition : Admin berada di halaman keanggotaan Post-condition : -

Tabel 4 5. Skenario *Use Case* Anggota

Aksi Aktor	Reaksi Sistem
1. Menekan menu k Anggota	
	2. Menampilkan data Anggota

5) Use Case Pengaduan

Aktor : Admin dan User

Deskripsi : Sebuah kegiatan untuk melihat pengaduan.

Pre-condition : Karyawan berada di halamanpengaduan.

Post-condition : Data dalam sistem akan diperbarui.

Tabel 4 6. Skenario *Use Case* Pengaduan

Aksi Aktor	Reaksi Sistem
1. Menekan menu pengaduan	
	2. Menampilkan menu pengaduan

6) Use Case Berita

Aktor : Admin

Deskripsi : Sebuah kegiatan untuk melihat berita.

Pre-condition : berada di halaman pengaduanberita.

Post-condition : Data dalam sistem akan diperbarui.

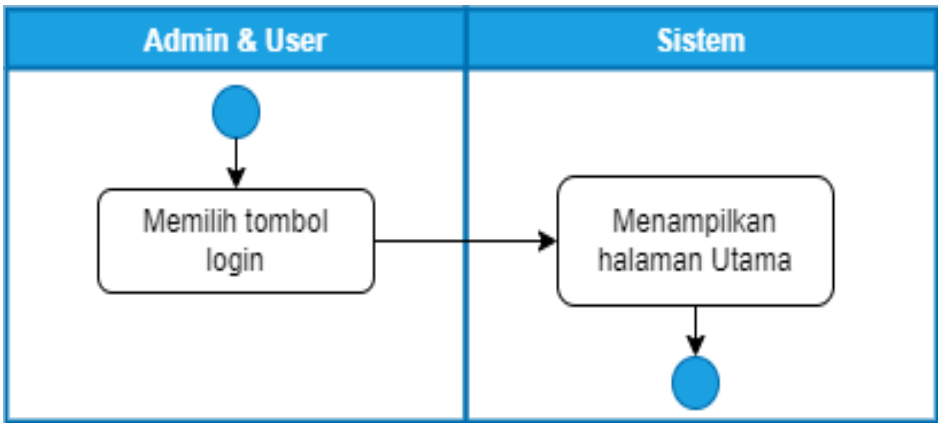
Tabel 4 7.Skenario Use Case Berita

Aksi Aktor	Reaksi Sistem
1.Menekan menu berita	
	1. Menampilkan menu berita

2. Activity Diagram

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

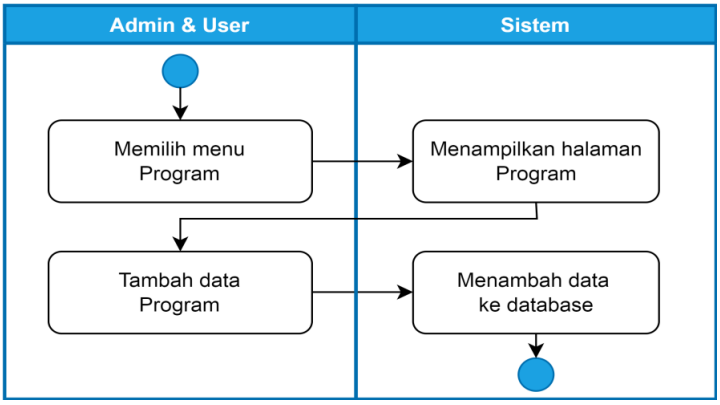
a. *Activity Diagram Home*



Gambar 4 2. Activity Diagram Home

Diagram aktivitas di atas menerangkan alur proses untuk aktivitas Home. Proses dimulai dengan admin atau karyawan memilih tombol login, kemudian system berhasil menampilkan halaman utama aplikasi.

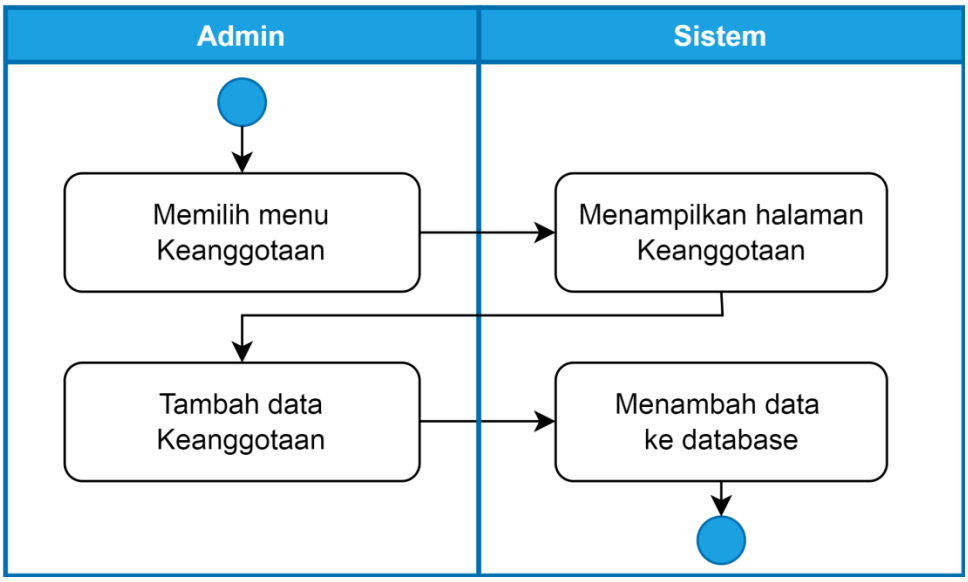
b. *Activity* Diagram Program Kerja MDMC



Gambar 4 3. Activity Diagram Program

Diagram aktivitas berikut menerangkan alur proses untuk aktivitas program Proses dimulai dengan admin maupun user memilih tombol program, kemudian sistem menampilkan halaman program. Jika ingin tambah data, maka dapat memilih tambah data sistem akan menambahkan data tersebut ke database.

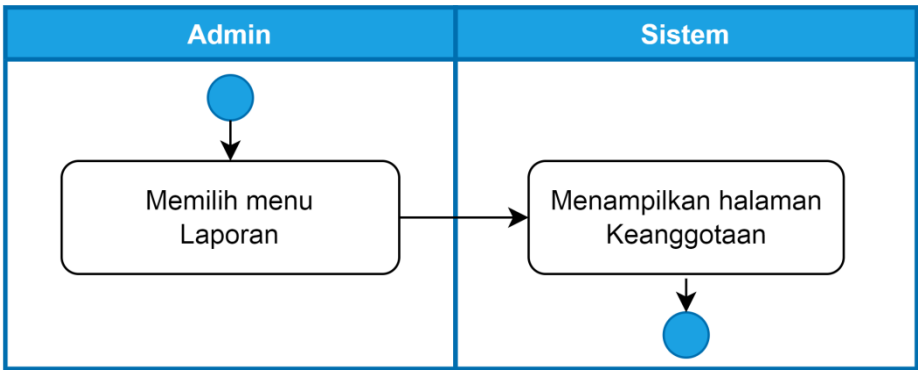
c. *Activity* Diagram Keanggotaan



Gambar 4 4. Activity Diagram Keanggotaan

Diagram aktivitas di atas menerangkan alur proses untuk aktivitas keanggotaan. Proses dimulai dengan admin memilih menu keanggotaan, kemudian sistem menampilkan halaman keanggotaan. Jika ingin tambah data keanggotaan, maka dapat memilih tambah data keanggotaan sistem akan menambahkan data tersebut ke database.

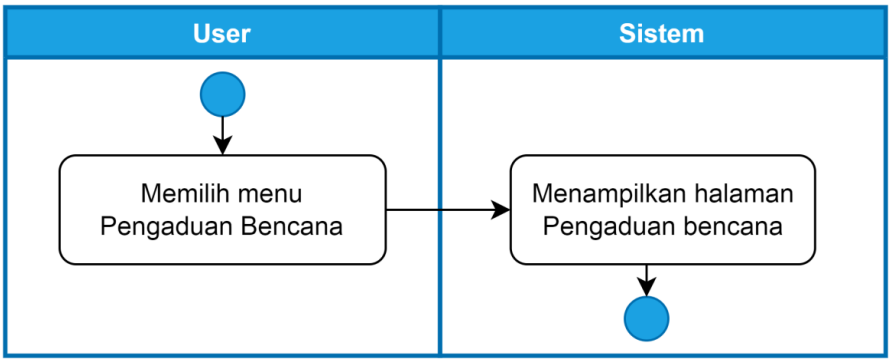
d. *Activity* Diagram Laporan



Gambar 4 5. Activity Diagram Laporan

Diagram aktivitas di atas menerangkan alur proses untuk aktivitas laporan. Proses dimulai dengan admin memilih menu laporan, kemudian sistem menampilkan halaman laporan.

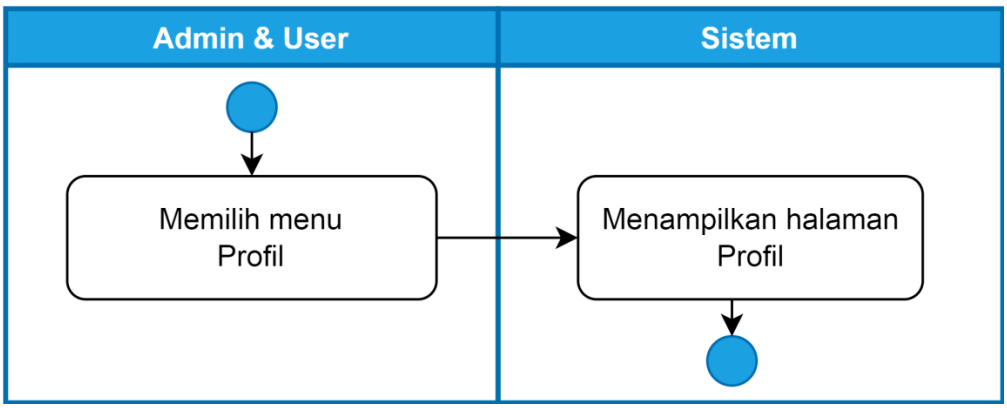
e. *Activity* Diagram Pengaduan Bencana



Gambar 4 6. Activity Diagram Pengaduan Bencana

Diagram aktivitas di atas menerangkan alur proses untuk aktivitas Pengaduan Bencana. Proses dimulai dengan user memilih menu Pengaduan Bencana, kemudian sistem menampilkan halaman tersebut.

f. *Activity* Diagram Profil



Gambar 4 7. Activity Diagram Profil

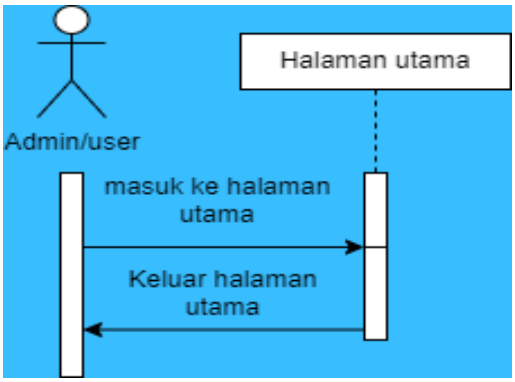
Diagram aktivitas di atas menerangkan alur proses untuk aktivitas profil. Proses dimulai dengan admin dan user memilih menu profil, kemudian sistem menampilkan halaman tersebut.

3. Sequence Diagram

Sequence diagram atau diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripisikan waktu hidup objek dan message yang dikirim dan diterima antar objek. Proses menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah use case beserta method yang dimiliki kelas yang diinisialisasi menjadi objek yang sudah tergambar dalam class diagram.

Berdasarkan desain use case, terdapat beberapa use case yang prosesnya hampir sama satu sama lain. Untuk mempermudah pembaha-san proses dalam pembuatan diagram sekuen, berikut ini ringkasan diagram sekuen pada aplikasi elektronik (E-MDMC):

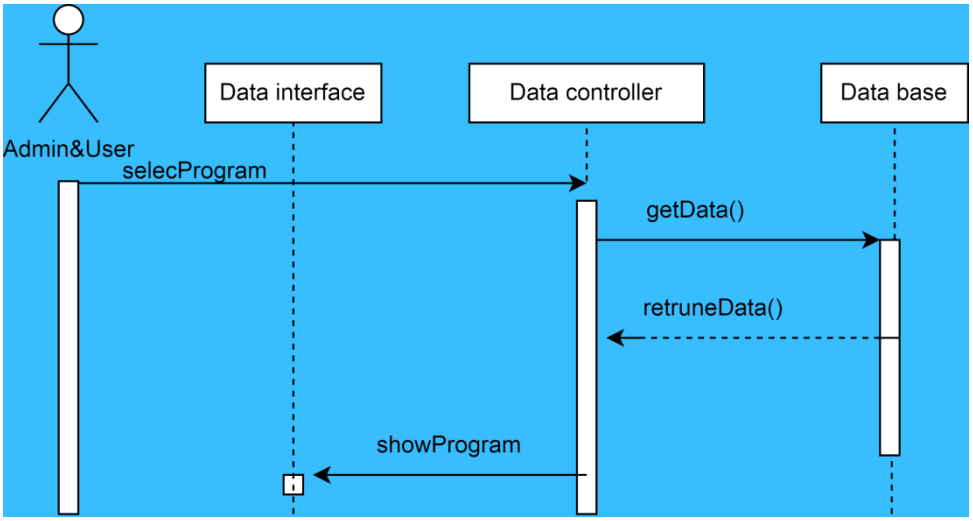
a. *Sequence Diagram*: Home



Gambar 4 8. Sequence Diagram Home

Diagram Sequence diagram home Admin dan user merupakan penggambaran proses masuk ke halaman utama. Bila proses login benar admin dan user bisa masuk ke dalam halaman utama dan menampilkan menu-menu yang ada pada aplikasi.

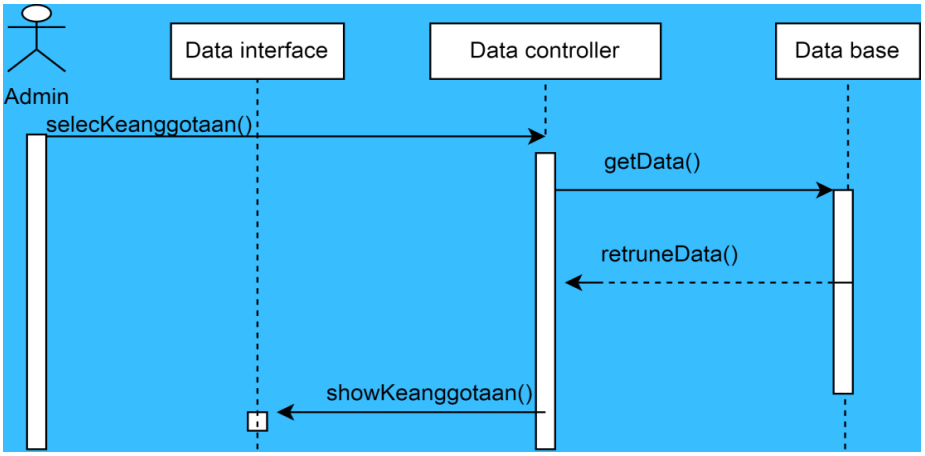
b. *Sequence Diagram*: Program



Gambar 4 9. Sequence Diagram Program

Diagram sekuen di atas menerangkan alur proses untuk aktivitas program. Admin dan user memilih data yang akan ditampilkan pada halaman program, dengan method selecProgram(), kemudian method getData() digunakan untuk mengambil data yang akan ditampilkan grafiknya ke database. Database akan mengembalikan data, kemudian method showProgram() akan menampilkan program.

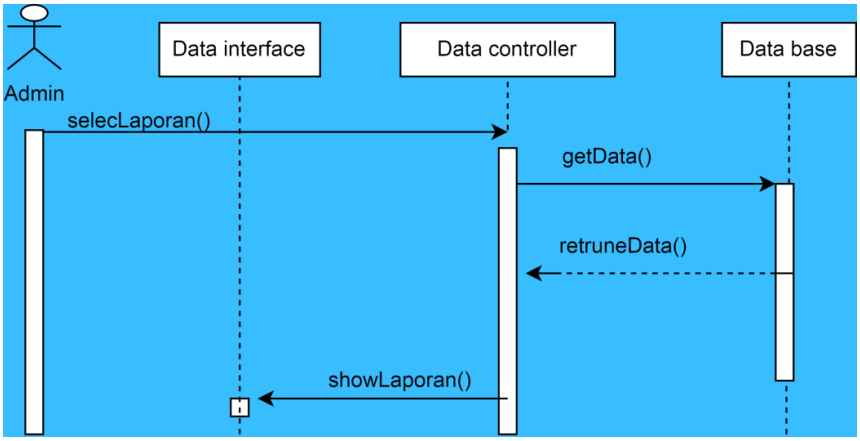
c. *Sequence Diagram*: Keanggotaan



Gambar 4 10. Sequence Diagram Keanggotaan

Diagram sekuen di atas menerangkan alur proses untuk aktivitas keanggotaan. Admin memilih data yang akan ditampilkan pada halaman keanggotaan, dengan method selecKeanggotaan(), kemudian method getData() digunakan untuk mengambil data yang akan ditampilkan grafiknya ke database. Database akan mengembalikan data, kemudian method showKeanggotaan() akan menampilkan keanggotaan.

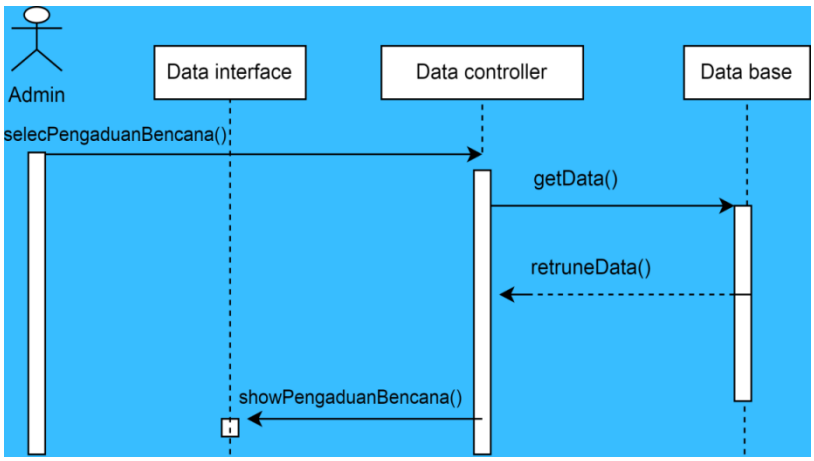
d. *Sequence Diagram*: Laporan



Gambar 4 11. Sequence Diagram Laporan

Diagram sekuen di atas menerangkan alur proses untuk aktivitas laporan. Admin memilih data yang akan ditampilkan pada halaman laporan, dengan method selecLaporan(), kemudian method getData() digunakan untuk mengambil data yang akan ditampilkan grafiknya ke database. Database akan mengembalikan data, kemudian method showLaporan() akan menampilkan laporan.

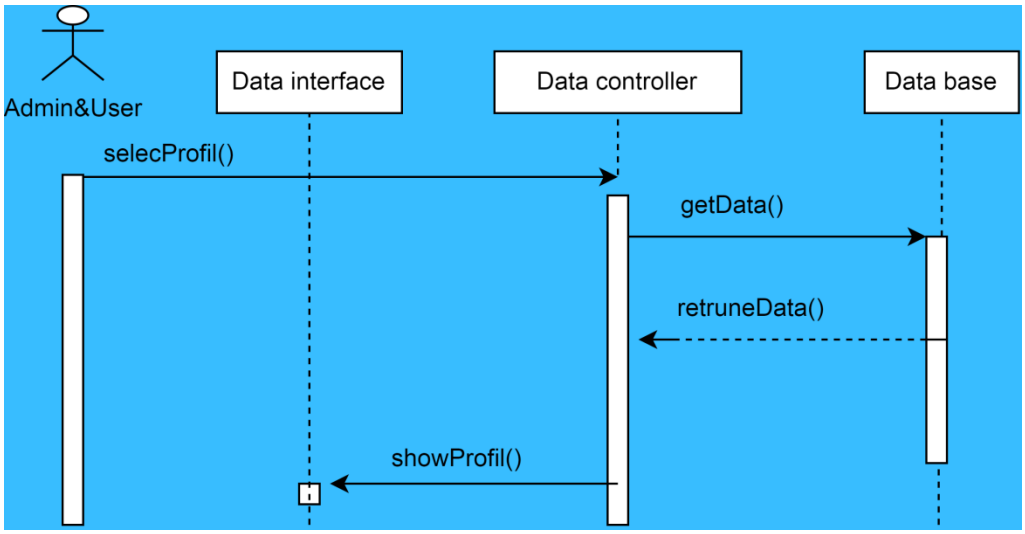
e. *Sequence Diagram*: Pengaduan Bencana



Gambar 4 12. Sequence Diagram Pengaduan Bencana

Diagram sekuen di atas menerangkan alur proses untuk aktivitas Pengaduan Bencana. User memilih data yang akan ditampilkan pada halaman Pengaduan Bencana, dengan method selecPengaduanBencana(), kemudian method getData() digunakan untuk mengambil data yang akan ditampilkan grafiknya ke database. Database akan mengembalikan data, kemudian method showPengaduanBencana() akan menampilkan Pengaduan Bencana.

f. *Sequence Diagram*: Pofil



Gambar 4 13. Sequence Diagram Profil

Diagram sekuen di atas menerangkan alur proses untuk aktivitas Profil. Admin dan User memilih data yang akan ditampilkan pada halaman Profil, dengan method selecProfil(), kemudian method getData() digunakan untuk mengambil data yang akan ditampilkan grafiknya ke database. Database akan mengembalikan data, kemudian method showProfil() akan menampilkan profil.

4. Tabel *Database*

Adapun rancangan *database* dalam pembangunan sistem ini adalah terdiri dari beberapa tabel diantaranya yaitu :

a. Nama *table*: setting**,** Adapun struktur *table* setting sebagai berikut.

Tabel 4 8. Tabel setting

No	Nama Field	Tipe Data	Ukuran
1	id_setting	int	11
2	nama_perusahaan	Varchar	255
3	alamat	Varchar	255
4	no_telpon	Varchar	20
5	email	Varchar	255
6	profil	text	-
7	visi	text	-
8	misi	text	-
9	program	text	-
10	sejarah	text	-
11	image	varchar	255
12	struktur	varchar	255

b. Tabel user. Nama table*: user*, Adapun struktur table *use*r sebagai berikut.

Tabel 4 9. Tabel user

No	Nama Field	Tipe Data	Ukuran
1	id_user	int	3
2	nama	Varchar	255
3	email	Varchar	255
4	image	Varchar	255

5	password	Varchar	255
6	is_active	int	1
7	date_created	int	11
8	token	Varchar	255

c. Nama table: data_informasi, Adapun struktur table data_informasi sebagai berikut.

Tabel 4 10. Tabel data_informasi

No	Nama Field	Tipe Data	Ukuran
1	id_informasi	int	11
2	judul_informasi	Varchar	100
3	tgl	datetime	-
4	keterangan	text	-
5	gambar_informasi	Varchar	100

d. Tabel laporan. Nama table: laporan, Adapun struktur table laporan sebagai berikut.

Tabel 4 11. Tabel laporan

No	Nama Field	Tipe Data	Ukuran
1	id_laporan	char	25
2	judul	Varchar	255
3	nama_pelapor	Varchar	255
4	lokasi	Varchar	255
5	keterangan	text	-
6	tgl	datetime	-
7	status	int	11
8	tanggapan	text	-
9	gambar	Varchar	255

e. Tabel berita. Nama table: berita, Adapun struktur table berita sebagai berikut.

Tabel 4 12. Tabel Pesanan

No	Nama Field	Tipe Data	Ukuran
1	id_berita	int	11
2	id_user	int	11

3	id_tketerangan	int	11
4	slug_berita	Varchar	255
5	judul_berita	Varchar	255
6	isi_berita	text	-
7	gambar_berita	Varchar	255
8	status_berita	Varchar	50
9	jenis_berita	Varchar	50
10	keywords	text	-
11	date_created	int	11
12	last_modified	date	-

f. Tabel staff. Nama table: staff, Adapun struktur table staff sebagai berikut.

Tabel 4 13. Tabel staf

No	Nama Field	Tipe Data	Ukuran
1	id_staff	int	11
2	id_kategori	int	11
3	nama_staff	Varchar	255
4	email_staff	Varchar	255
5	alamat	Varchar	255
6	no_telpon	Varchar	20
7	gambar_staff	Varchar	255
8	Gender	Varchar	20
9	Publish	Varchar	20
10	date_created	timestamp	-
11	last_modified	date	-

g. Tabel kategori_staff. Nama table: kategori_staff, Adapun struktur table kategori_ staff sebagai berikut.

Tabel 4 14. Tabel kategori_ staff

No	Nama Field	Tipe Data	Ukuran
1	id_kategori	int	11
2	slug_kategori	Varchar	255
3	nama_kategori	Varchar	255
4	date_created	timestamp	-

h. Tabel kategori_berita. Nama table: kategori_berita, Adapun struktur table kategori_ berita sebagai berikut.

Tabel 4 15. Tabel kategori_berita

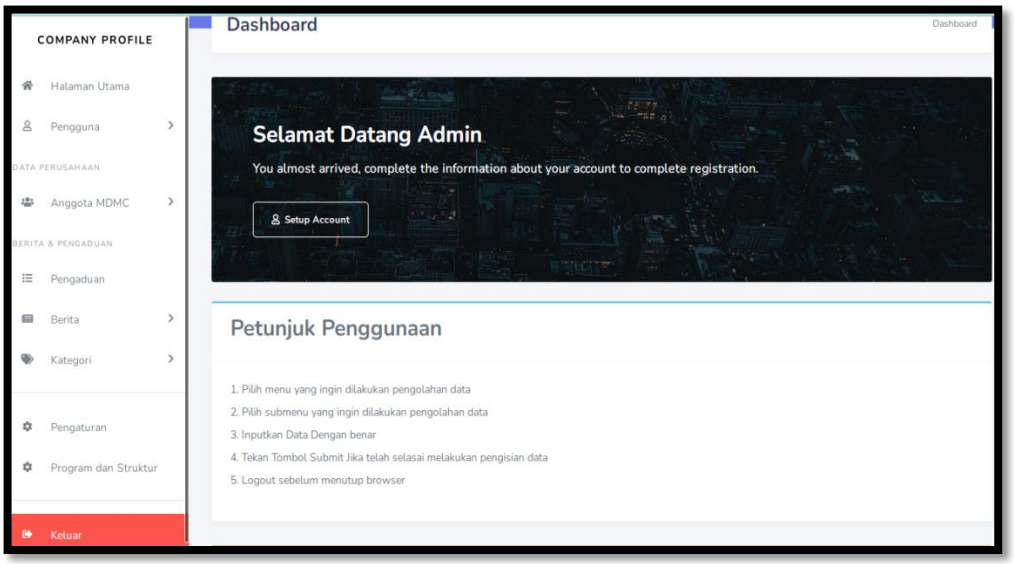
No	Nama Field	Tipe Data	Ukuran
1	id_kategori	int	11
2	slug_kategori	Varchar	255
3	nama_kategori	Varchar	255
4	date_created	timestamp	-

5. Perancangan Sistem

a. Halaman Admin

1) Halaman Home

Berikut adalah tampilan home pada Aplikasi Elektronik (E-MDMC). Halaman home dalam aplikasi ini berfungsi sebagai pusat informasi utama dan titik navigasi bagi pengguna.



Gambar 4 14. Halaman Home

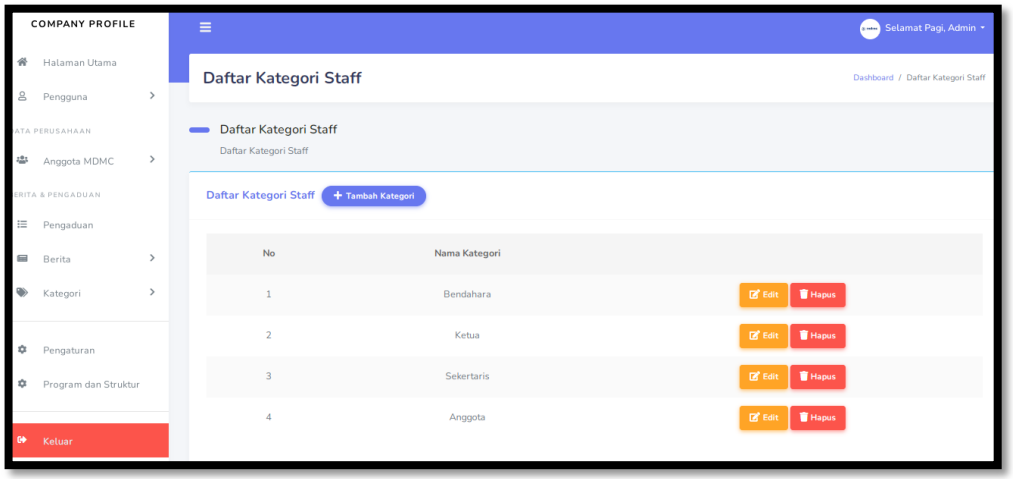
Berikut adalah coding tampilan halaman home

```
<?php namespace App\Controllers;
use App\Controllers\BaseController;
class Home extends BaseController
{
   public function index()
   {
      $data = [
         'title' => 'Home - MDMC',
```

```
        ];
        return view('home', $data);
    }
}
```

2) Halaman Anggota MDMC

Berikut adalah tampilan anggota MDMC pada Aplikasi Elektronik (E-MDMC). Halaman anggota MDMC dalam aplikasi ini berfungsi untuk menampilkan daftar anggota MDMC, memberikan akses ke fitur-fitur tertentu, serta menyediakan informasi terkait anggota MDMC.



Gambar 4 15. Halaman Anggota MDMC

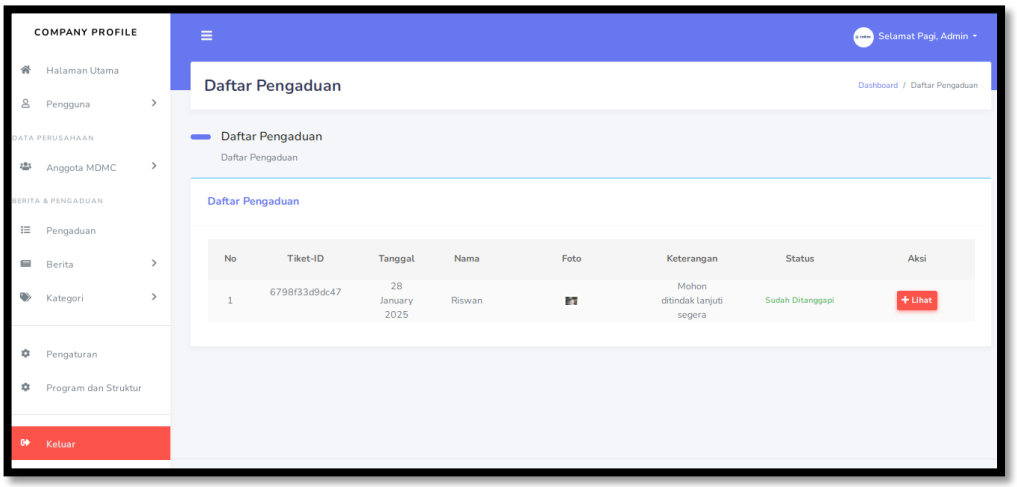
Berikut tampilan coding halaman anggota MDMC

```
<?php
// application/views/welcome_message.php
?><!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>MDMC Application</title>
```

```
    <link rel="stylesheet" href="<?php echo base_url('assets/css/style.css'); ?>">
</head>
<body>
    <header>
        <h1>Welcome to MDMC Application</h1>
    </header>
    <nav>
        <ul>
            <li><a href="<?php echo site_url('home'); ?>">Home</a></li>
            <li><a href="<?php echo site_url('about'); ?>">About</a></li>
            <li><a href="<?php echo site_url('contact'); ?>">Contact</a></li>
        </ul>
    </nav>
    <main>
        <h2>Dashboard</h2>
        <p>This is the main dashboard of the MDMC application.</p>
    </main>
    <footer>
        <p>&copy; 2025 MDMC Application</p>
    </footer>
</body>
</html>
```

3) Halaman Pengaduan

Berikut adalah tampilan pengaduan pada Aplikasi Elektronik (E-MDMC). Halaman pengaduan dalam aplikasi ini berfungsi untuk menampilkan daftar pengaduan.



Gambar 4 16. Halaman pengaduan

Berikut tampilan coding halaman pengaduan

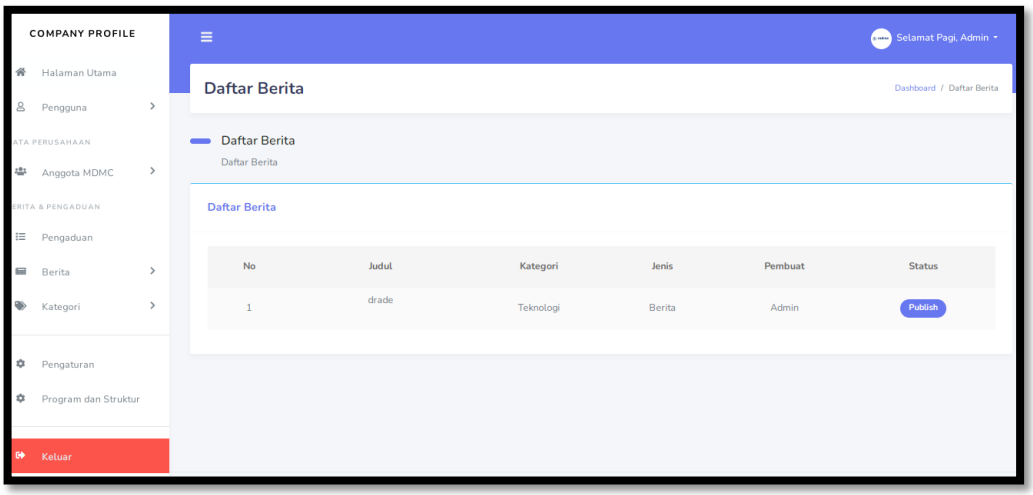
```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');
class Keanggotaan extends CI_Controller {
    public function __construct() {
        parent::__construct();
        $this->load->model('Keanggotaan_model');
    }
    public function index() {
        $data['title'] = 'Menu Keanggotaan';
        $data['members'] = $this->Keanggotaan_model->get_all_members();

        $this->load->view('templates/header', $data);
        $this->load->view('keanggotaan/index', $data);
        $this->load->view('templates/footer');
    }
    public function tambah() {
        $data['title'] = 'Tambah Anggota';
```

```
        $this->load->view('templates/header', $data);
        $this->load->view('keanggotaan/tambah', $data);
        $this->load->view('templates/footer');
    }
    public function simpan() {
        $this->Keanggotaan_model->insert_member();
        redirect('keanggotaan');
    }
    public function hapus($id) {
        $this->Keanggotaan_model->delete_member($id);
        redirect('keanggotaan');
    }
}
```

4) Halaman Berita

Berikut adalah tampilan berita pada Aplikasi Elektronik (E-MDMC). Halaman berita dalam aplikasi ini berfungsi untuk menampilkan berita.



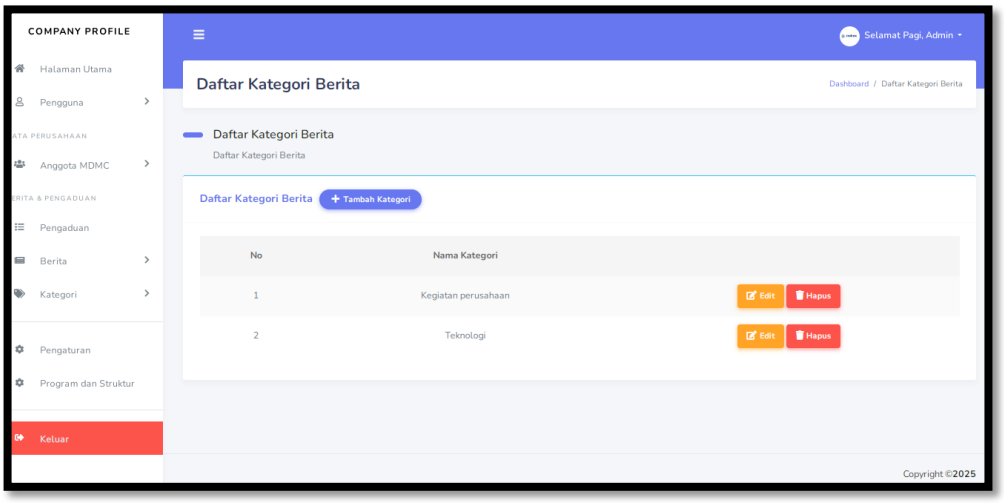
Gambar 4 17. Halaman Berita

Berikut tampilan halaman berita

```
<?php
namespace App\Controllers;
use App\Models\LaporanModel;
use CodeIgniter\Controller;
class Laporan extends Controller
{
   public function index()
   {
      $model = new LaporanModel();
      $data['laporan'] = $model->findAll();
      echo view('templates/header');
      echo view('laporan/index', $data);
      echo view('templates/footer');
   }
}
```

5) Halaman Kategori Berita

Berikut adalah tampilan kategori berita pada Aplikasi Elektronik (E-MDMC). Halaman kategori berita dalam aplikasi ini berfungsi untuk menampilkan data kategori berita .



Gambar 4 18. Halaman Kategori Berita

Berikut tampilan coding halaman profil

```
// app/Models/KategoriModel.php
namespace App\Models;
use CodeIgniter\Model;
class KategoriModel extends Model
{
    protected $table = 'kategori';
    protected $primaryKey = 'id';
    protected $allowedFields = ['nama_kategori', 'deskripsi'];
}
// app/Controllers/KategoriController.php
namespace App\Controllers;
use App\Models\KategoriModel;
use CodeIgniter\RESTful\ResourceController;
class KategoriController extends ResourceController
{
    protected $modelName = 'App\\Models\\KategoriModel';
    protected $format    = 'json';
```

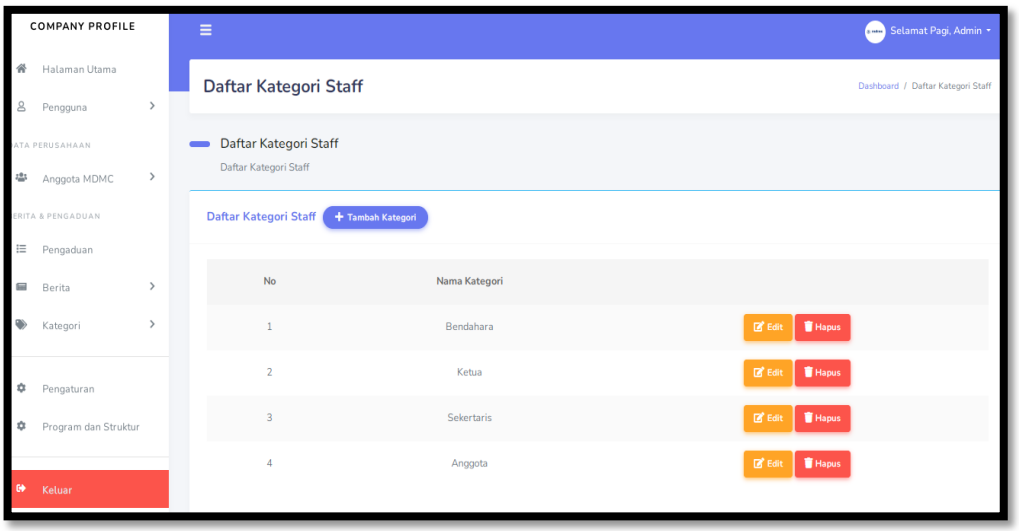
```
public function index()
{
    return $this->respond($this->model->findAll());
}
public function create()
{
    $data = [
        'nama_kategori' => $this->request->getPost('nama_kategori'),
        'deskripsi' => $this->request->getPost('deskripsi')
    ];
    $this->model->insert($data);
    return $this->respondCreated($data);
}
public function edit($id)
{
    return $this->respond($this->model->find($id));
}
public function update($id)
{
    $data = [
        'nama_kategori' => $this->request->getVar('nama_kategori'),
        'deskripsi' => $this->request->getVar('deskripsi')
    ];
    $this->model->update($id, $data);
    return $this->respond($data);
}
public function delete($id)
{
    $this->model->delete($id);
```

```
        return $this->respondDeleted(['id' => $id]);
    }
}
// app/Views/kategori/index.php
<!DOCTYPE html>
<html>
<head>
    <title>Daftar Kategori</title>
</head>
<body>
    <h1>Daftar Kategori</h1>
    <table border="1">
        <tr>
            <th>ID</th>
            <th>Nama Kategori</th>
            <th>Deskripsi</th>
        </tr>
        <?php foreach ($kategori as $k) : ?>
        <tr>
            <td><?= $k['id']; ?></td>
            <td><?= $k['nama_kategori']; ?></td>
            <td><?= $k['deskripsi']; ?></td>
        </tr>
        <?php endforeach; ?>
    </table>
</body>
</html>
// Routes (app/Config/Routes.php)
$routes->get('/kategori', 'KategoriController::index');
```

```
$routes->post('/kategori', 'KategoriController::create');
$routes->get('/kategori/(:num)', 'KategoriController::edit/$1');
$routes->put('/kategori/(:num)', 'KategoriController::update/$1');
$routes->delete('/kategori/(:num)', 'KategoriController::delete/$1');
```

6) Halaman Kategori Anggota MDMC

Berikut adalah tampilan kategori anggota MDMC pada Aplikasi Elektronik (E-MDMC). Halaman kategori anggota MDMC dalam aplikasi ini berfungsi untuk menampilkan kategori anggota MDMC.



Gambar 4 19. Halaman Kategori Anggota MDMC

Berikut tampilan coding halaman kategori anggota MDMC

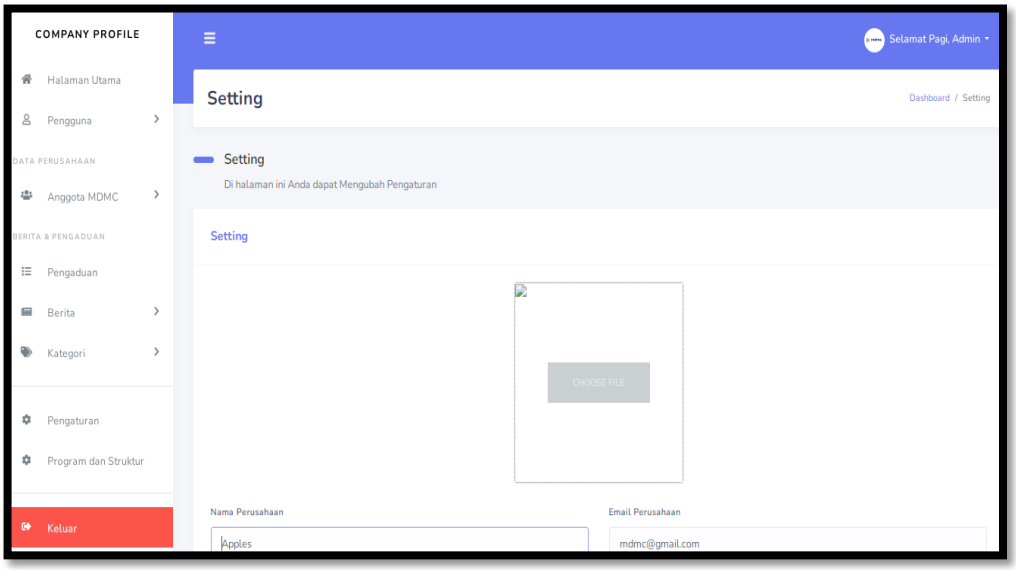
```
<?php
namespace App\Controllers;
use App\Models\KategoriAnggotaModel;
use CodeIgniter\Controller;
class KategoriAnggota extends Controller
{
   protected $model;
```

```
public function __construct()
{
    $this->model = new KategoriAnggotaModel();
}
public function index()
{
    $data['kategori'] = $this->model->findAll();
    return view('kategori_anggota/index', $data);
}
public function create()
{
    return view('kategori_anggota/form');
}
public function store()
{
    $this->model->save([
        'nama_kategori' => $this->request->getPost('nama_kategori')
    ]);
    return redirect()->to('/kategorianggota');
}
public function edit($id)
{
    $data['kategori'] = $this->model->find($id);
    return view('kategori_anggota/form', $data);
}
public function update($id)
{
    $this->model->update($id, [
        'nama_kategori' => $this->request->getPost('nama_kategori')
```

```
        ]);
        return redirect()->to('/kategorianggota');
    }
    public function delete($id)
    {
        $this->model->delete($id);
        return redirect()->to('/kategorianggota');
    }
}
```

7) Halaman Pengaturan

Berikut adalah tampilan pengaturan pada Aplikasi Elektronik (E-MDMC). Halaman pengaturan dalam aplikasi ini berfungsi untuk menampilkan pengaturan.



Gambar 4 20. Halaman Pengaturan

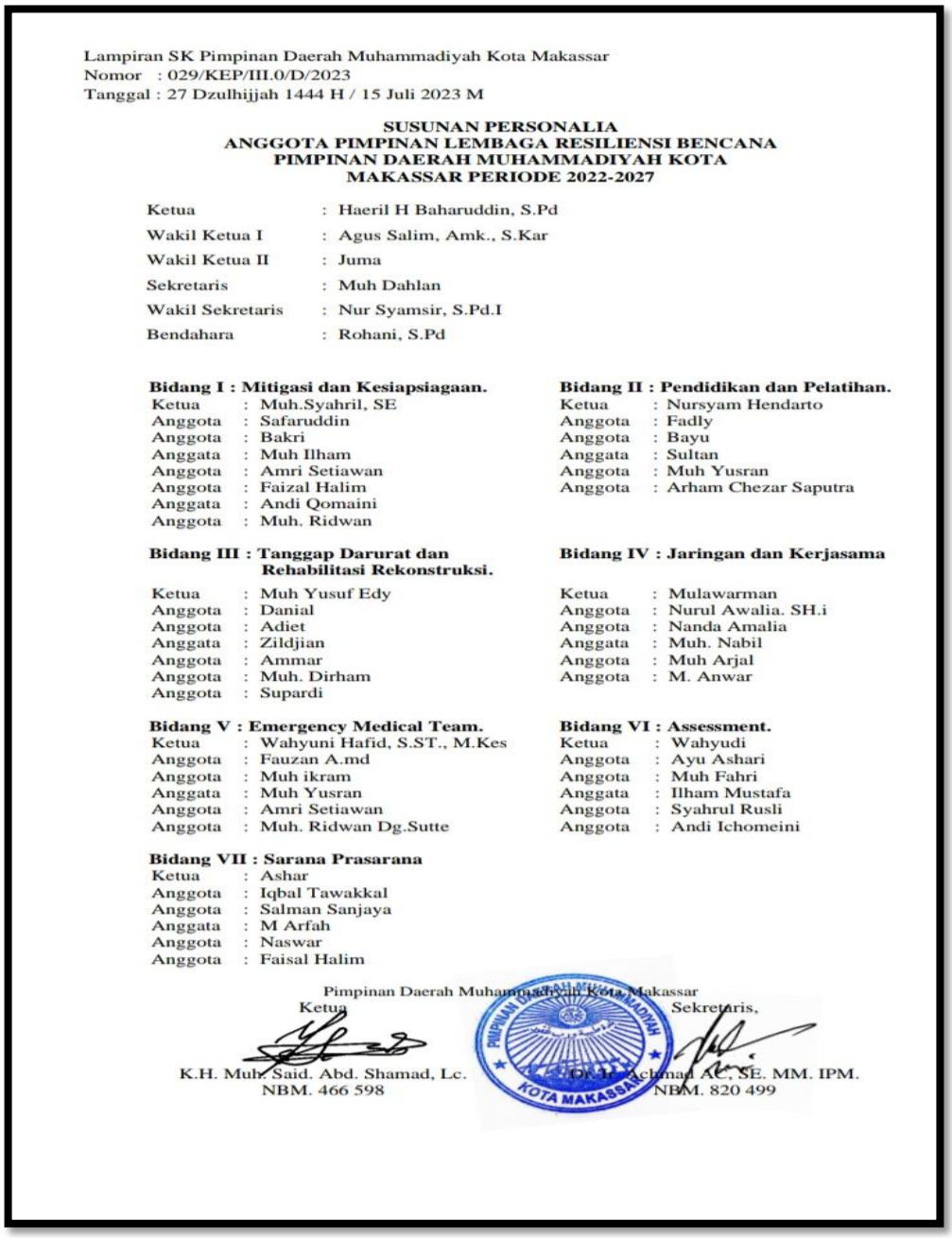
Berikut tampilan coding halaman kategori anggota MDMC

```
<?php
namespace App\Models;
use CodeIgniter\Model;
class SettingsModel extends Model
{
    protected $table = 'settings';
    protected $primaryKey = 'id';
    protected $allowedFields = ['key', 'value'];

    public function getSetting($key)
    {
        return $this->where('key', $key)->first();
    }
    public function updateSetting($key, $value)
    {
        $setting = $this->where('key', $key)->first();
        if ($setting) {
            $this->update($setting['id'], ['value' => $value]);
        } else {
            $this->insert(['key' => $key, 'value' => $value]);
        }
    }
}
```

8) Halaman Program dan Struktur

Berikut adalah tampilan struktur organisasi pada Aplikasi Elektronik (E-MDMC). Halaman program dan struktur dalam aplikasi ini berfungsi untuk menampilkan program dan struktur .



Lampiran SK Pimpinan Daerah Muhammadiyah Kota Makassar
Nomor : 029/KEP/III.0/D/2023
Tanggal : 27 Dzulhijjah 1444 H / 15 Juli 2023 M

SUSUNAN PERSONALIA
ANGGOTA PIMPINAN LEMBAGA RESILIENSI BENCANA
PIMPINAN DAERAH MUHAMMADIYAH KOTA
MAKASSAR PERIODE 2022-2027

Ketua : Haeril H Baharuddin, S.Pd
Wakil Ketua I : Agus Salim, Amk., S.Kar
Wakil Ketua II : Juma
Sekretaris : Muh Dahlan
Wakil Sekretaris : Nur Syamsir, S.Pd.I
Bendahara : Rohani, S.Pd

Bidang I : Mitigasi dan Kesiapsiagaan.
Ketua : Muh.Syahril, SE
Anggota : Safaruddin
Anggota : Bakri
Anggata : Muh Ilham
Anggota : Amri Setiawan
Anggota : Faizal Halim
Anggata : Andi Qomaini
Anggota : Muh. Ridwan

Bidang II : Pendidikan dan Pelatihan.
Ketua : Nursyam Hendarto
Anggota : Fadly
Anggota : Bayu
Anggata : Sultan
Anggota : Muh Yusran
Anggota : Arham Chezar Saputra

Bidang III : Tanggap Darurat dan Rehabilitasi Rekonstruksi.
Ketua : Muh Yusuf Edy
Anggota : Danial
Anggota : Adiet
Anggata : Zildjian
Anggota : Ammar
Anggota : Muh. Dirham
Anggota : Supardi

Bidang IV : Jaringan dan Kerjasama
Ketua : Mulawarman
Anggota : Nurul Awalia. SH.i
Anggota : Nanda Amalia
Anggata : Muh. Nabil
Anggota : Muh Arjal
Anggota : M. Anwar

Bidang V : Emergency Medical Team.
Ketua : Wahyuni Hafid, S.ST., M.Kes
Anggota : Fauzan A.md
Anggota : Muh ikram
Anggata : Muh Yusran
Anggota : Amri Setiawan
Anggota : Muh. Ridwan Dg.Sutte

Bidang VI : Assessment.
Ketua : Wahyudi
Anggota : Ayu Ashari
Anggota : Muh Fahri
Anggata : Ilham Mustafa
Anggota : Syahrul Rusli
Anggota : Andi Ichomeini

Bidang VII : Sarana Prasarana
Ketua : Ashar
Anggota : Iqbal Tawakkal
Anggota : Salman Sanjaya
Anggata : M Arfah
Anggota : Naswar
Anggota : Faisal Halim

Pimpinan Daerah Muhammadiyah Kota Makassar
Ketua, Sekretaris,

K.H. Muh. Said. Abd. Shamad, Lc. Dr. Ir. Achmad AC, SE. MM. IPM.
NBM. 466 598 NBM. 820 499

Gambar 4 21. Halaman Struktur Organisasi

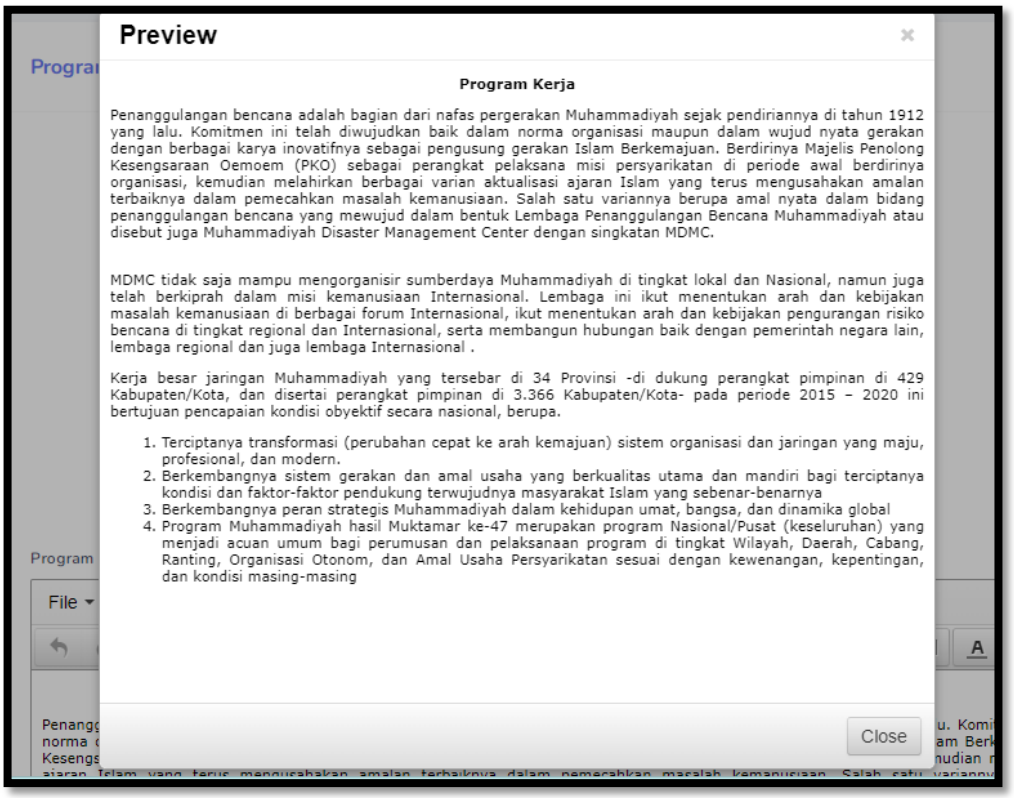
Berikut tampilan coding halaman Struktur Organisasi

```
<!DOCTYPE html>
<html>
<head>
    <title>Data Mahasiswa</title>
</head>
<body>
    <h2>Data Mahasiswa</h2>
    <a href="/mahasiswa/create">Tambah Mahasiswa</a>
    <table border="1">
        <tr>
            <th>ID</th>
            <th>Nama</th>
            <th>Email</th>
            <th>Jurusan</th>
            <th>Aksi</th>
        </tr>
        <?php foreach ($mahasiswa as $mhs): ?>
        <tr>
            <td><?= $mhs['id'] ?></td>
            <td><?= $mhs['nama'] ?></td>
            <td><?= $mhs['email'] ?></td>
            <td><?= $mhs['jurusan'] ?></td>
            <td>
                <a href="/mahasiswa/edit/<?= $mhs['id'] ?>">Edit</a>
                <a href="/mahasiswa/delete/<?= $mhs['id'] ?>">Hapus</a>
            </td>
        </tr>
        <?php endforeach; ?>
```

```
    </table>
</body>
</html> ?>
```

9) Halaman Program dan Struktur

Berikut adalah tampilan Program Kerja pada Aplikasi Elektronik (E-MDMC). Halaman program dan struktur dalam aplikasi ini berfungsi untuk menampilkan program dan struktur .



Gambar 4 22. Halaman Program Kerja

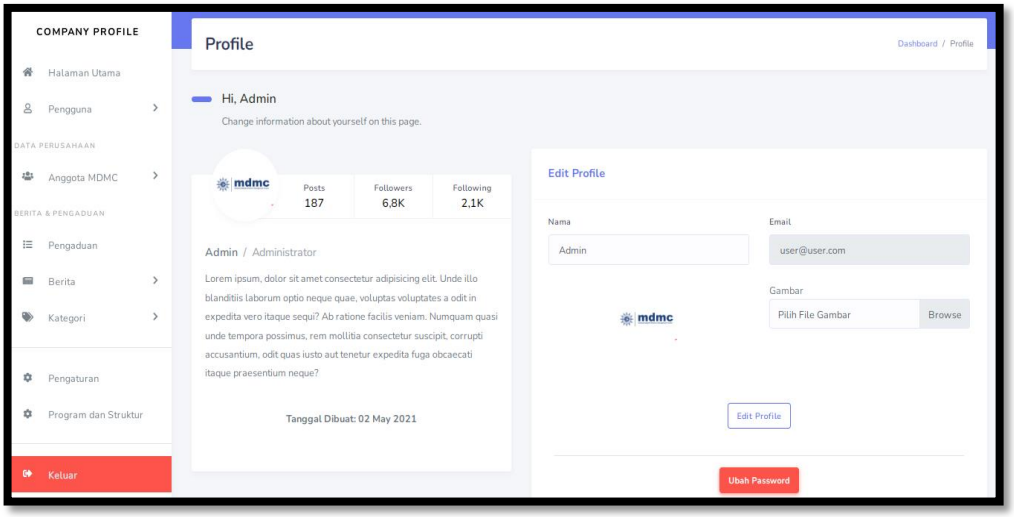
Berikut tampilan coding halaman Program Kerja

```
<!DOCTYPE html>
<html>
<head>
```

```
    <title>Data Mahasiswa</title>
</head>
<body>
    <h2>Data Mahasiswa</h2>
    <a href="/mahasiswa/create">Tambah Mahasiswa</a>
    <table border="1">
        <tr>
            <th>ID</th>
            <th>Nama</th>
            <th>Email</th>
            <th>Jurusan</th>
            <th>Aksi</th>
        </tr>
        <?php foreach ($mahasiswa as $mhs): ?>
        <tr>
            <td><?= $mhs['id'] ?></td>
            <td><?= $mhs['nama'] ?></td>
            <td><?= $mhs['email'] ?></td>
            <td><?= $mhs['jurusan'] ?></td>
            <td>
                <a href="/mahasiswa/edit/<?= $mhs['id'] ?>">Edit</a>
                <a href="/mahasiswa/delete/<?= $mhs['id'] ?>">Hapus</a>
            </td>
        </tr>
        <?php endforeach; ?>
    </table>
</body>
</html> ?>
```

10) Halaman Profil

Berikut adalah tampilan profil pada Aplikasi Elektronik (E-MDMC). Halaman profil aplikasi ini berfungsi untuk menampilkan profil.



Gambar 4 23. Halaman Profil

Berikut tampilan coding halaman profil

```
<?php
require_once('koneksi.php');
if($con){
    $id=$_POST['email'];
    $sql=mysqli_query($con, "SELECT * FROM `tb_user` WHERE email='$id'") or die (mysqli_error());
    if(mysqli_num_rows($sql)>0){
        $respon["Profil"]=array();
        while ($row=mysqli_fetch_array($sql)) {
            $hasil=array();
            $hasil["nama"] = $row['nama'];
            $hasil["email"] = $row['email'];
            $hasil["gambar"] = $lg.$row['image'];
```

```
                array_push($respon["Profil"], $hasil);
                $respon["hasil"]=1;
            }
            echo json_encode($respon);
        }else{
            $respon["hasil"]=0;
            $respon["pesan"]="Hasil Tidak Ditemukan";
            echo json_encode($respon);
        }
}else{
        echo json_encode(array('respon'=> 'koneksi gagal'));
}
mysqli_close($con);
 ?>
```

b. Halaman User

1) Halaman Home

Berikut adalah tampilan home user pada Aplikasi Elektronik (E-MDMC). Halaman home dalam aplikasi ini berfungsi sebagai pusat informasi utama dan titik navigasi bagi pengguna.



Gambar 4 24. Halaman Home

Berikut tampilan halaman home

```
<?php namespace App\Controllers;
use App\Controllers\BaseController;
class Home extends BaseController
{
    public function index()
    {
        $data = [
            'title' => 'Home - MDMC',
        ];
        return view('home', $data);
    }
}
```

2) Halaman Program

Berikut adalah tampilan program pada Aplikasi Elektronik (E-MDMC). Halaman program dalam aplikasi ini berfungsi untuk menampilkan daftar program, memberikan akses ke fitur-fitur tertentu, serta menyediakan informasi detail terkait setiap program.



Gambar 4 20. Halaman Program

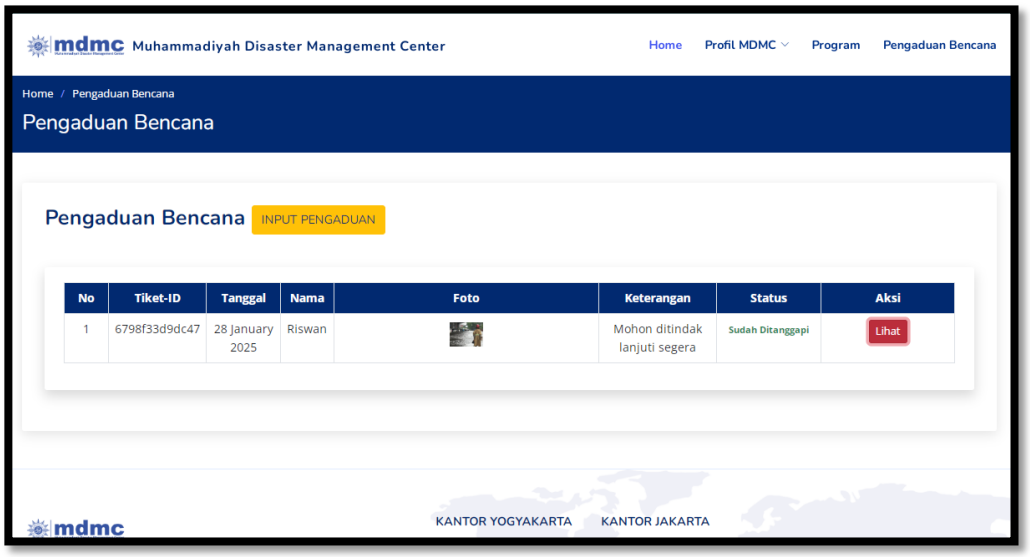
Berikut tampilan coding halaman program

```
<?php
// application/views/welcome_message.php
?><!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>MDMC Application</title>
    <link rel="stylesheet" href="<?php echo base_url('assets/css/style.css'); ?>">
</head>
```

```
<body>
    <header>
        <h1>Welcome to MDMC Application</h1>
    </header>
    <nav>
        <ul>
            <li><a href="<?php echo site_url('home'); ?>">Home</a></li>
            <li><a href="<?php echo site_url('about'); ?>">About</a></li>
            <li><a href="<?php echo site_url('contact'); ?>">Contact</a></li>
        </ul>
    </nav>
    <main>
        <h2>Dashboard</h2>
        <p>This is the main dashboard of the MDMC application.</p>
    </main>
    <footer>
        <p>&copy; 2025 MDMC Application</p>
    </footer>
</body>
</html>
```

3) Halaman Pengaduan Bencana

Berikut adalah tampilan pengaduan bencana pada Aplikasi Elektronik (E-MDMC). Halaman pengaduan bencana dalam aplikasi ini berfungsi untuk menampilkan daftar pengaduan bencana, memberikan akses ke fitur-fitur tertentu, serta menyediakan informasi tentang pengaduan bencana.



Gambar 4 21. Halaman Pengaduan bencana

```
// application/Controllers/Pengaduan.php
namespace App\Controllers;
use App\Models\PengaduanModel;
use CodeIgniter\Controller;
class Pengaduan extends Controller
{
    public function index()
    {
        $model = new PengaduanModel();
        $data['pengaduan'] = $model->findAll();
        echo view('templates/header');
        echo view('pengaduan/index', $data);
        echo view('templates/footer');
    }
    public function tambah()
    {
        echo view('templates/header');
        echo view('pengaduan/tambah');
```

```
        echo view('templates/footer');
    }
    public function simpan()
    {
        $model = new PengaduanModel();
        $model->insert([
            'nama' => $this->request->getPost('nama'),
            'lokasi' => $this->request->getPost('lokasi'),
            'deskripsi' => $this->request->getPost('deskripsi'),
        ]);
        return redirect()->to('/pengaduan');
    }
}
```

4) Halaman Profil MDMC

Berikut adalah tampilan Profil pada Aplikasi Elektronik (E-MDMC). Halaman Profil dalam aplikasi ini berfungsi untuk menampilkan data profil.



Gambar 4 22. Halaman Profil MDMC

Berikut tampilan coding halaman profil MDMC

```
<?php
require_once('koneksi.php');
if($con){
	$id=$_POST['email'];
	$sql=mysqli_query($con, "SELECT * FROM `tb_user` WHERE email='$id'") or die (mysqli_error());
	if(mysqli_num_rows($sql)>0){
		$respon["Profil"]=array();
		while ($row=mysqli_fetch_array($sql)) {
			$hasil=array();
			$hasil["nama"] = $row['nama'];
			$hasil["email"] = $row['email'];
			$hasil["gambar"] = $lg.$row['image'];
			array_push($respon["Profil"], $hasil);
			$respon["hasil"]=1;
		}
		echo json_encode($respon);
	}else{
		$respon["hasil"]=0;
		$respon["pesan"]="Hasil Tidak Ditemukan";
		echo json_encode($respon);
	}
}else{
	echo json_encode(array('respon'=> 'koneksi gagal'));
}
mysqli_close($con);
 ?>
```

5) Halaman Visi dan Misi

Berikut adalah tampilan visi dan misi pada Aplikasi Elektronik (E-MDMC). Halaman visi dan misi dalam aplikasi ini berfungsi untuk menampilkan data visi dan misi .



Gambar 4 23. Halaman Visi dan Misi

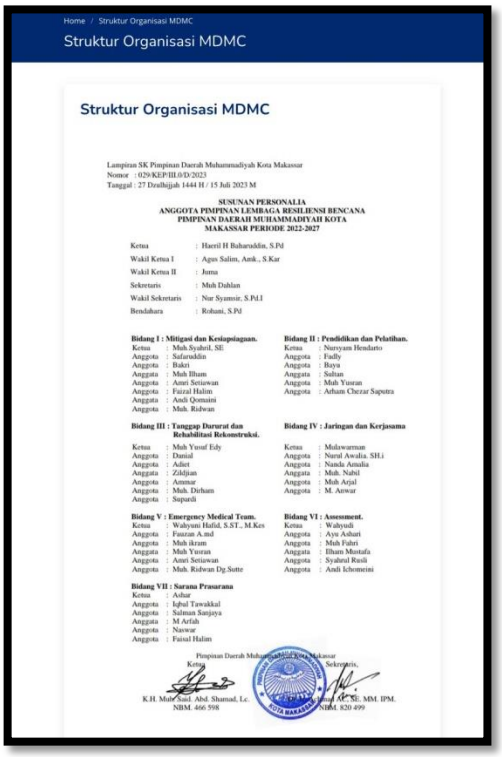
Berikut tampilan coding halaman visi dan misi MDMC

```
<?php
require_once('koneksi.php');
if($con){
    $id=$_POST['email'];
    $sql=mysqli_query($con, "SELECT * FROM `tb_user` WHERE email='$id'") or die (mysqli_error());
    if(mysqli_num_rows($sql)>0){
        $respon["Profil"]=array();
        while ($row=mysqli_fetch_array($sql)) {
            $hasil=array();
            $hasil["nama"] = $row['nama'];
            $hasil["email"] = $row['email'];
```

```
                $hasil["gambar"] = $lg.$row['image'];
                array_push($respon["Profil"], $hasil);
                $respon["hasil"]=1;
            }
            echo json_encode($respon);
        }else{
            $respon["hasil"]=0;
            $respon["pesan"]="Hasil Tidak Ditemukan";
            echo json_encode($respon);
        }
}else{
        echo json_encode(array('respon'=> 'koneksi gagal'));
}
mysqli_close($con);
 ?>
```

6) Halaman Struktur Organisasi

Berikut adalah tampilan Struktur Organisasi pada Aplikasi Elektronik (E-MDMC). Halaman Struktur Organisasi dalam aplikasi ini berfungsi untuk menampilkan data Struktur Organisasi.



Gambar 4 24. Halaman Struktur Organisasi

Berikut tampilan coding halaman Struktur Organisasi

```
<?php
require_once('koneksi.php');
if($con){
    $id=$_POST['email'];
    $sql=mysqli_query($con, "SELECT * FROM `tb_user` WHERE email='$id'") or die (mysqli_error());
    if(mysqli_num_rows($sql)>0){
        $respon["Profil"]=array();
        while ($row=mysqli_fetch_array($sql)) {
            $hasil=array();
            $hasil["nama"] = $row['nama'];
            $hasil["email"] = $row['email'];
            $hasil["gambar"] = $lg.$row['image'];
```

```
                array_push($respon["Profil"], $hasil);
                $respon["hasil"]=1;
            }
            echo json_encode($respon);
        }else{
            $respon["hasil"]=0;
            $respon["pesan"]="Hasil Tidak Ditemukan";
            echo json_encode($respon);
        }
}else{
    echo json_encode(array('respon'=> 'koneksi gagal'));
}
mysqli_close($con);
 ?>
```

7) Halaman Anggota MDMC

Berikut adalah tampilan Anggota MDMC pada Aplikasi Elektronik (E-MDMC). Halaman Anggota MDMC dalam aplikasi ini berfungsi untuk menampilkan data Anggota MDMC.



Gambar 4 25. Halaman Anggota MDMC

Berikut tampilan coding halaman Struktur Organisasi

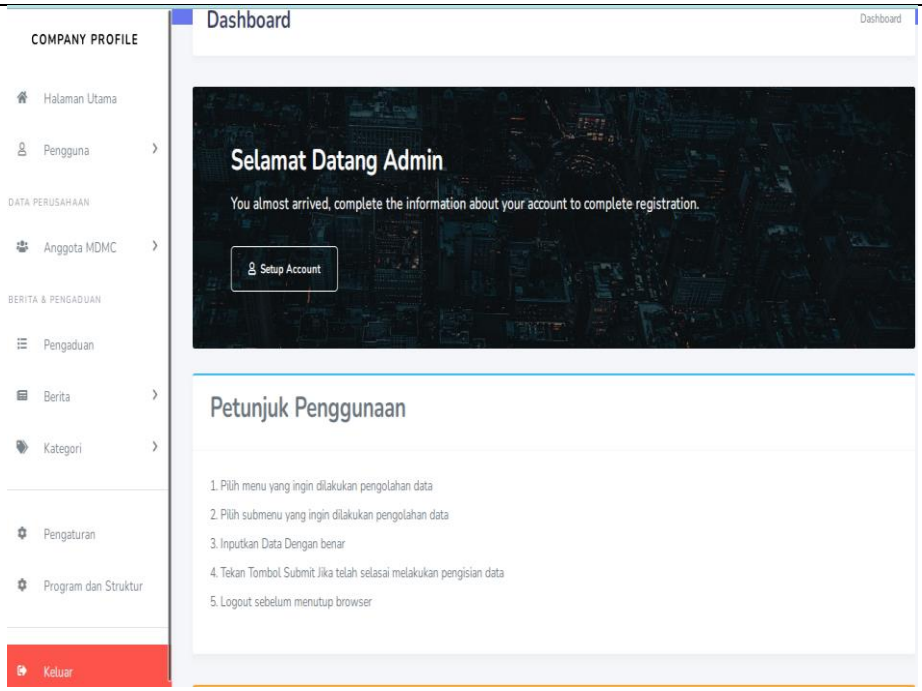
```
<?php
require_once('koneksi.php');
if($con){
	$id=$_POST['email'];
	$sql=mysqli_query($con, "SELECT * FROM `tb_user` WHERE email='$id'") or die (mysqli_error());
	if(mysqli_num_rows($sql)>0){
		$respon["Profil"]=array();
		while ($row=mysqli_fetch_array($sql)) {
			$hasil=array();
			$hasil["nama"] = $row['nama'];
			$hasil["email"] = $row['email'];
			$hasil["gambar"] = $lg.$row['image'];
			array_push($respon["Profil"], $hasil);
			$respon["hasil"]=1;
		}
		echo json_encode($respon);
	}else{
		$respon["hasil"]=0;
		$respon["pesan"]="Hasil Tidak Ditemukan";
		echo json_encode($respon);
	}
}else{
	echo json_encode(array('respon'=> 'koneksi gagal'));
}
mysqli_close($con);
 ?>
```

6. Pengujian Sistem

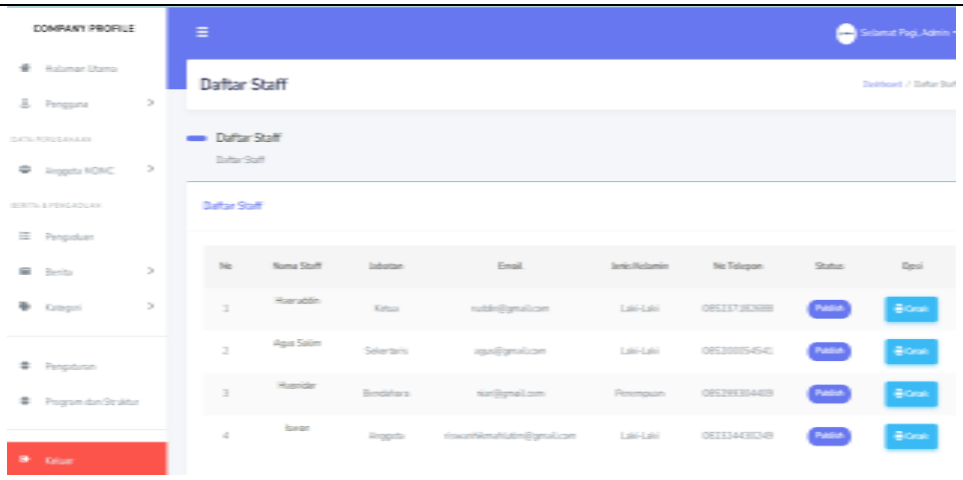
a. *Black box Admin*

Pengujian *BlackBox* digunakan untuk menguji sistem. *BlackBox* adalah metode pengujian perangkat lunak yang memeriksa fungsionalitas aplikasi tanpa mempelajari struktur atau cara kerja internalnya. Pendekatan pengujian ini berlaku untuk hampir semua tingkat pengujian perangkat lunak, termasuk unit, integrasi, sistem, dan penerimaan.

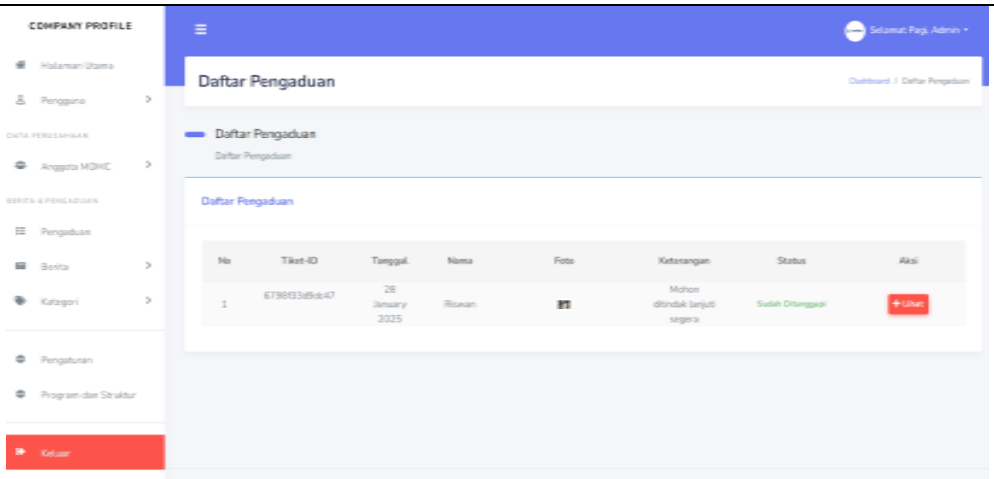
Tabel 4 16. *BlackBox* Halaman Utama Admin

Test Faktor	Hasil	Kesimpulan
Jika memasukkan username dan password dengan benar.	✓	Informasi, tampil halaman utama *admin*.
Screen Shot		
		

Tabel 4 17. *BlackBox Form* Anggota MDMC

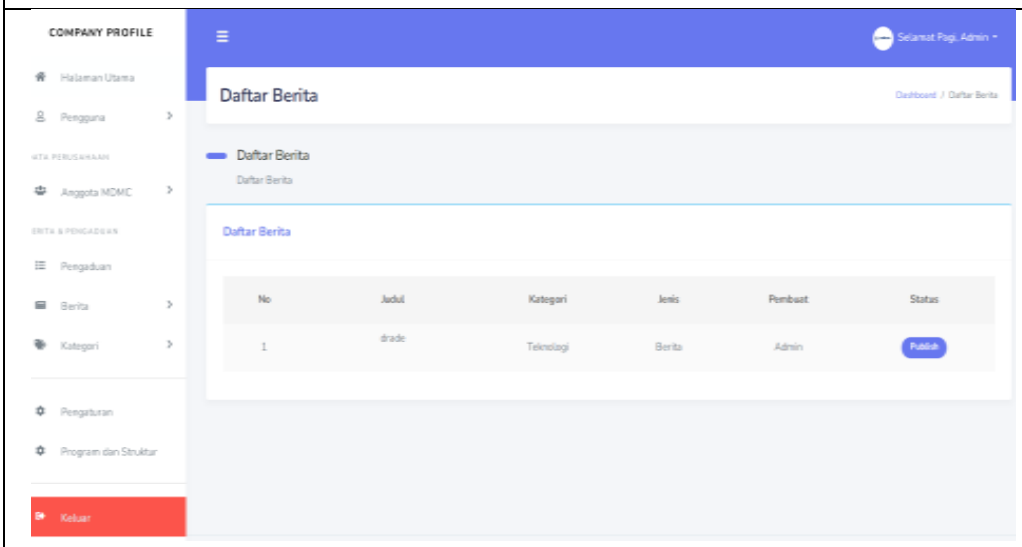
Test Faktor	Hasil	Kesimpulan
Jika memilih menu anggota MDMC pada halaman utama.	✓	Informasi, tampil form anggota MDMC.
Screen Shot		
		

Tabel 4 18. *BlackBox Form* Pengaduan

Test Faktor	Hasil	Kesimpulan
Jika memilih menu pengaduan pada halaman utama.	✓	Informasi, tampil form pengaduan.
Screen Shot		
		

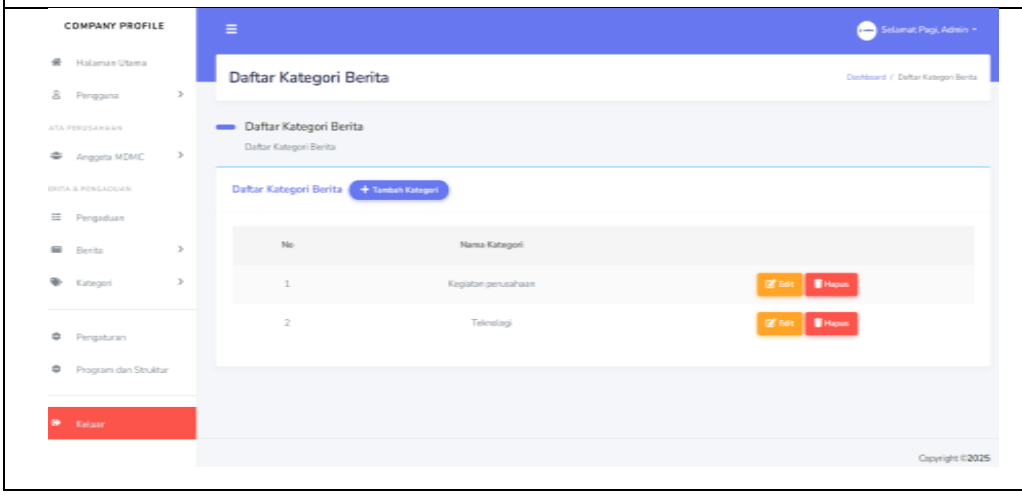
Tabel 4 19. *BlackBox Form* Berita

Test Faktor	Hasil	Kesimpulan
Jika memilih menu berita pada halaman utama.	✓	Informasi, tampil form berita.
Screen Shot		



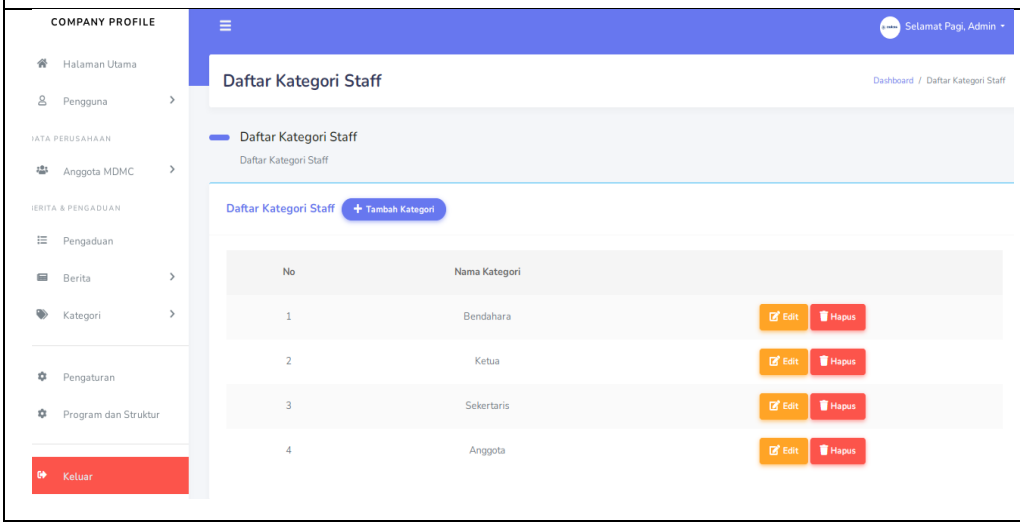
Tabel 4 20. *BlackBox Form* Daftar Kategori Berita

Test Faktor	Hasil	Kesimpulan
Jika memilih menu daftar kategori berita pada halaman utama.	✓	Informasi, tampil form daftar kategori berita
Screen Shot		



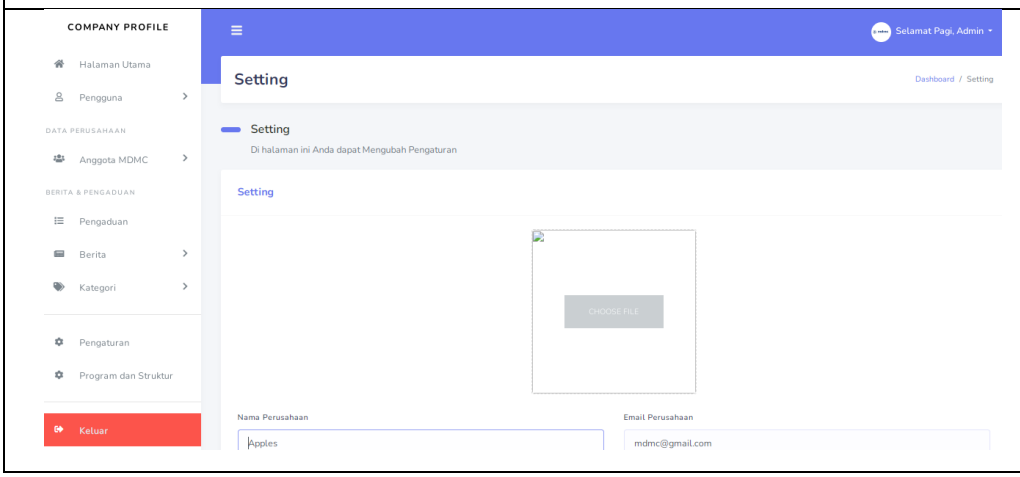
Tabel 4 21. *BlackBox Form* Daftar Kategori Staff

Test Faktor	Hasil	Kesimpulan
Jika memilih menu daftar kategori staff pada halaman utama.	✓	Informasi, tampil form daftar kategori staff.
Screen Shot		



Tabel 4 22. *BlackBox Form* Pengaturan

Test Faktor	Hasil	Kesimpulan
Jika memilih menu pengaturan pada halaman utama.	✓	Informasi, tampil form pengaturan
Screen Shot		



Tabel 4 23. *BlackBox Form* Program dan Struktur

Test Faktor	Hasil	Kesimpulan
Jika memilih menu struktur organisasi pada halaman utama.	✓	Informasi, tampil form struktur organisasi
Screen Shot		

Lampiran SK Pimpinan Daerah Muhammadiyah Kota Makassar
Nomor : 029/KEP/III.0/D/2023
Tanggal : 27 Dzulhijjah 1444 H / 15 Juli 2023 M

SUSUNAN PERSONALIA
ANGGOTA PIMPINAN LEMBAGA RESILIENSI BENCANA
PIMPINAN DAERAH MUHAMMADIYAH KOTA
MAKASSAR PERIODE 2022-2027

Ketua : Haeril H Baharuddin, S.Pd
Wakil Ketua I : Agus Salim, Amk., S.Kar
Wakil Ketua II : Juma
Sekretaris : Muh Dahlan
Wakil Sekretaris : Nur Syamsir, S.Pd.I
Bendahara : Rohani, S.Pd

Bidang I : Mitigasi dan Kesiapsiagaan.
Ketua : Muh.Syahril, SE
Anggota : Safaruddin
Anggota : Bakri
Anggata : Muh Ilham
Anggota : Amri Setiawan
Anggota : Faizal Halim
Anggata : Andi Qomaini
Anggota : Muh. Ridwan

Bidang II : Pendidikan dan Pelatihan.
Ketua : Nursyam Hendarto
Anggota : Fadly
Anggota : Bayu
Anggata : Sultan
Anggota : Muh Yusran
Anggota : Arham Chezar Saputra

Bidang III : Tanggap Darurat dan Rehabilitasi Rekonstruksi.
Ketua : Muh Yusuf Edy
Anggota : Danial
Anggota : Adiet
Anggata : Zildjian
Anggota : Ammar
Anggota : Muh. Dirham
Anggota : Supardi

Bidang IV : Jaringan dan Kerjasama
Ketua : Mulawarman
Anggota : Nurul Awalia. SH.i
Anggota : Nanda Amalia
Anggata : Muh. Nabil
Anggota : Muh Arjal
Anggota : M. Anwar

Bidang V : Emergency Medical Team.
Ketua : Wahyuni Hafid, S.ST., M.Kes
Anggota : Fauzan A.md
Anggota : Muh ikram
Anggata : Muh Yusran
Anggota : Amri Setiawan
Anggota : Muh. Ridwan Dg.Sutte

Bidang VI : Assessment.
Ketua : Wahyudi
Anggota : Ayu Ashari
Anggota : Muh Fahri
Anggata : Ilham Mustafa
Anggota : Syahrul Rusli
Anggota : Andi Ichomeini

Bidang VII : Sarana Prasarana
Ketua : Ashar
Anggota : Iqbal Tawakkal
Anggota : Salman Sanjaya
Anggata : M Arfah
Anggota : Naswar
Anggota : Faisal Halim

Pimpinan Daerah Muhammadiyah Kota Makassar

Ketua,

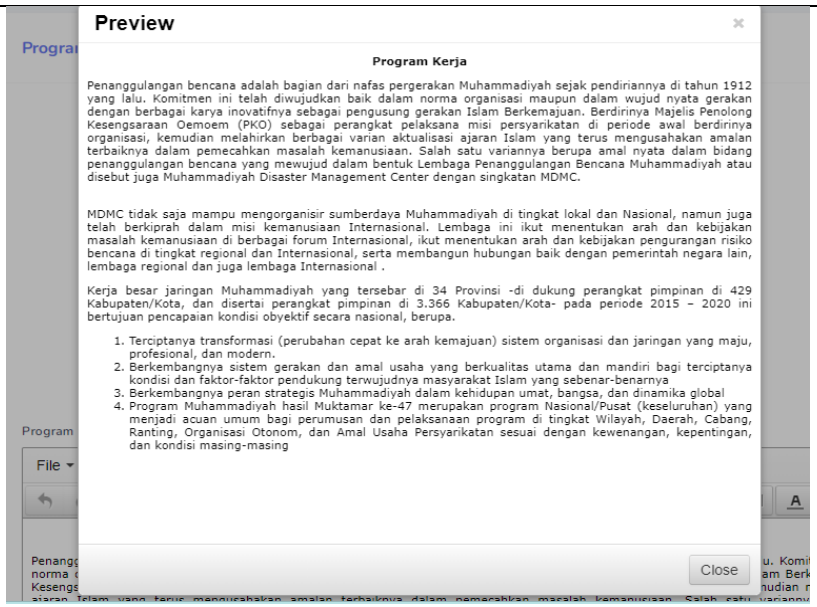
K.H. Muh. Said. Abd. Shamad, Lc.
NBM. 466 598

Sekretaris,

Dr. H. Aclunan AC, SE. MM. IPM.
NBM. 820 499

Tabel 4 24. *BlackBox Form* Program dan Struktur

Test Faktor	Hasil	Kesimpulan
Jika memilih menu program kerja pada halaman utama.	✓	Informasi, tampil form program kerja
Screen Shot		



b. *BlackBox User*

Tabel 4 25. *BlackBox* Halaman Utama *User*

Test Faktor	Hasil	Kesimpulan
Jika memasukkan username dan password dengan benar.	✓	Informasi, tampil halaman utama *user*.
Screen Shot		



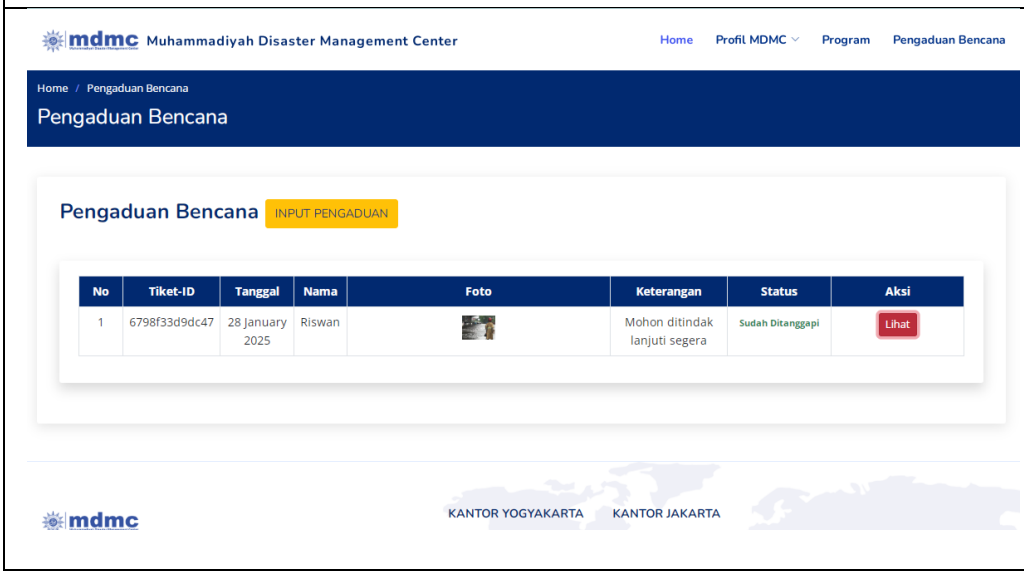
Tabel 4 26. *BlackBox Form* Program

Test Faktor	Hasil	Kesimpulan
Jika memilih menu Program pada halaman utama.	✓	Informasi, tampil form program.
Screen Shot		



Tabel 4 27. *BlackBox* Form Pengaduan Bencana

Test Faktor	Hasil	Kesimpulan
Jika memilih menu pengaduan bencana pada halaman utama.	✓	Informasi, tampil form pengaduan bencana.
Screen Shot		



Tabel 4 28. *BlackBox* Form Profil MDMC

Test Faktor	Hasil	Kesimpulan
Jika memilih menu Profil MDMC pada halaman utama.	✓	Informasi, tampil form profil MDMC

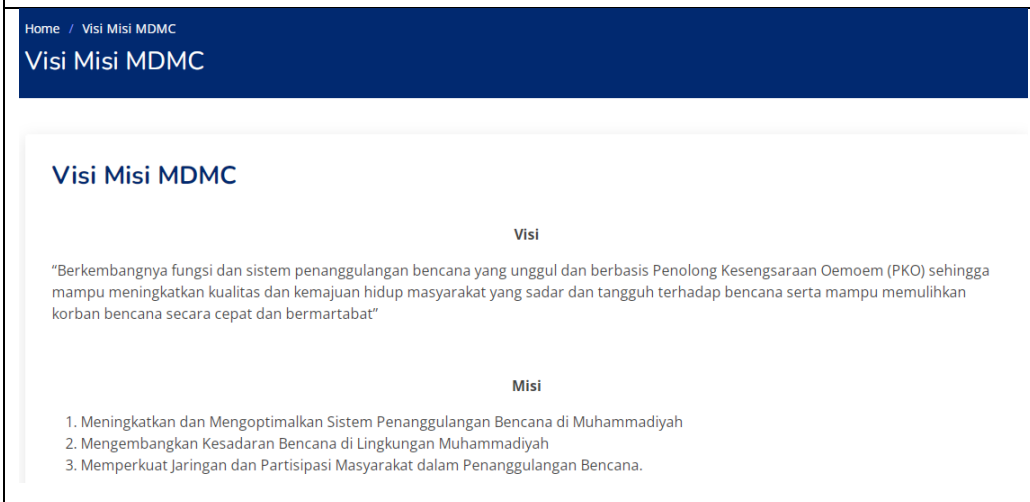
Screen Shot



Tabel 4 29. *BlackBox* Form Visi dan Misi

Test Faktor	Hasil	Kesimpulan
Jika memilih menu Visi dan Misi pada halaman utama.	✓	Informasi, tampil form Visi dan Misi

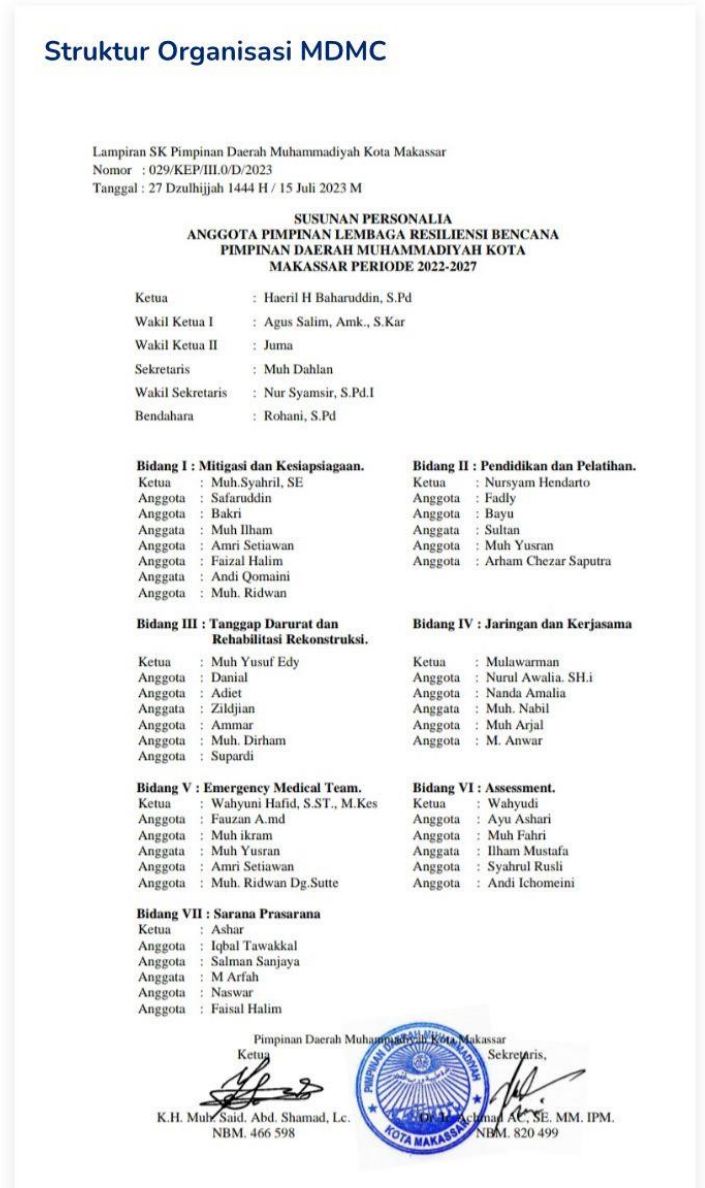
Screen Shot



Tabel 4 30. *BlackBox* Form Struktur Organisasi

Test Faktor	Hasil	Kesimpulan
Jika memilih menu Struktur Organisasi pada halaman utama.	✓	Informasi, tampil form Struktur Organisasi
Screen Shot		





Struktur Organisasi MDMC

Lampiran SK Pimpinan Daerah Muhammadiyah Kota Makassar
Nomor : 029/KEP/III.0/D/2023
Tanggal : 27 Dzulhijjah 1444 H / 15 Juli 2023 M

SUSUNAN PERSONALIA
ANGGOTA PIMPINAN LEMBAGA RESILIENSI BENCANA
PIMPINAN DAERAH MUHAMMADIYAH KOTA
MAKASSAR PERIODE 2022-2027

Ketua : Haeril H Baharuddin, S.Pd
Wakil Ketua I : Agus Salim, Amk., S.Kar
Wakil Ketua II : Juma
Sekretaris : Muh Dahlan
Wakil Sekretaris : Nur Syamsir, S.Pd.I
Bendahara : Rohani, S.Pd

Bidang I : Mitigasi dan Kesiapsiagaan.
Ketua : Muh.Syahril, SE
Anggota : Safaruddin
Anggota : Bakri
Anggata : Muh Ilham
Anggota : Amri Setiawan
Anggota : Faizal Halim
Anggata : Andi Qomaini
Anggota : Muh. Ridwan

Bidang II : Pendidikan dan Pelatihan.
Ketua : Nursyam Hendarto
Anggota : Fadly
Anggota : Bayu
Anggata : Sultan
Anggota : Muh Yusran
Anggota : Arham Chezar Saputra

Bidang III : Tanggap Darurat dan Rehabilitasi Rekonstruksi.
Ketua : Muh Yusuf Edy
Anggota : Danial
Anggota : Adiet
Anggata : Zildjian
Anggota : Ammar
Anggota : Muh. Dirham
Anggota : Supardi

Bidang IV : Jaringan dan Kerjasama
Ketua : Mulawarman
Anggota : Nurul Awalia. SH.i
Anggota : Nanda Amalia
Anggata : Muh. Nabil
Anggota : Muh Arjal
Anggota : M. Anwar

Bidang V : Emergency Medical Team.
Ketua : Wahyuni Hafid, S.ST., M.Kes
Anggota : Fauzan A.md
Anggota : Muh ikram
Anggata : Muh Yusran
Anggota : Amri Setiawan
Anggota : Muh. Ridwan Dg.Sutte

Bidang VI : Assessment.
Ketua : Wahyudi
Anggota : Ayu Ashari
Anggota : Muh Fahri
Anggata : Ilham Mustafa
Anggota : Syahrul Rusli
Anggota : Andi Ichomeini

Bidang VII : Sarana Prasarana
Ketua : Ashar
Anggota : Iqbal Tawakkal
Anggota : Salman Sanjaya
Anggata : M Arfah
Anggota : Naswar
Anggota : Faisal Halim

Pimpinan Daerah Muhammadiyah Kota Makassar
Ketua, Sekretaris,

K.H. Muh. Said. Abd. Shamad, Lc. Dr. M. Rahman AC, SE. MM. IPM.
NBM. 466 598 NBM. 820 499

Tabel 4 31. *BlackBox* Form Anggota MDMC

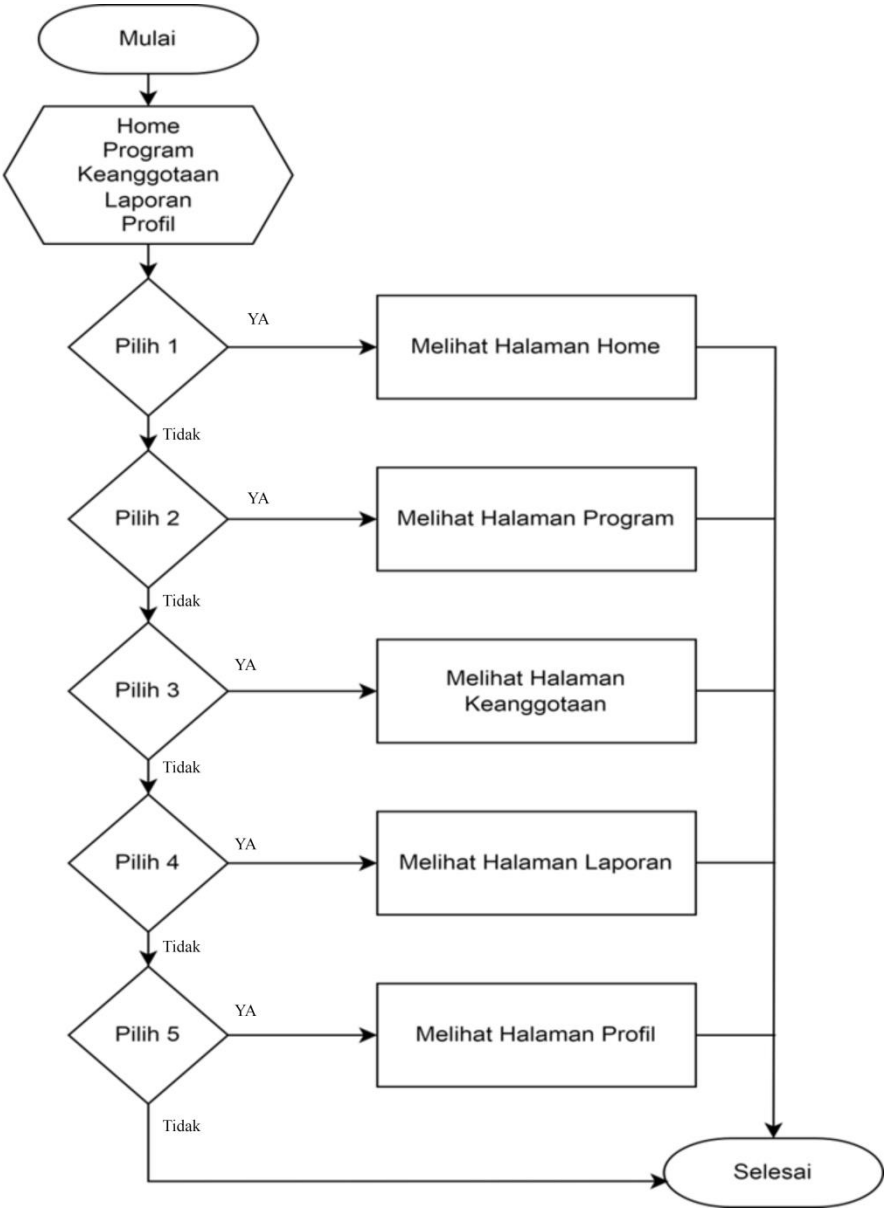
Test Faktor	Hasil	Kesimpulan
Jika memilih menu Anggota MDMC pada halaman utama.	✓	Informasi, tampil form Anggota MDMC
Screen Shot		



c. *WhiteBox*

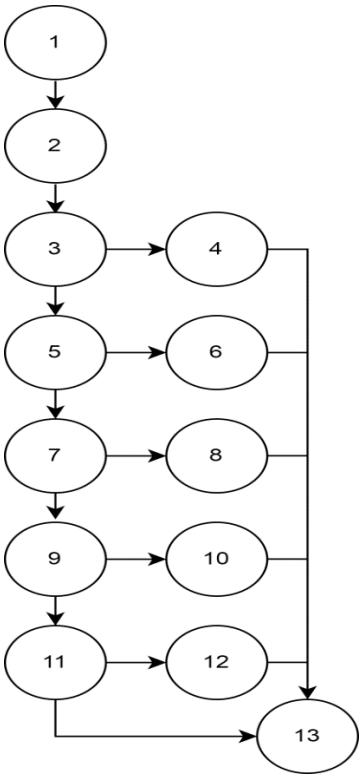
Pengujian aplikasi dilakukan dengan cara pengujian *WhiteBox*. *WhiteBox* adalah metode pengujian perangkat lunak yang menguji struktur internal atau cara kerja aplikasi, yang bertentangan dengan fungsinya. Dalam pengujian kotak putih, perspektif internal sistem digunakan untuk merancang kasus uji.

1) Flowchart dan *Flowgraph* Aktivitas *Admin*



Gambar 4 26. Flowchart Aktivitas Admin

Dari *Flowchart* yang digunakan *untuk* pengujian perangkat lunak, maka ditentukan *Flowgraph* sebagai berikut:



Gambar 4 27. Flowgraph Aktivitas Admin

Dari *Flowgraph* aktivitas *admin* diatas dapat dilakukan proses perhitungan sebagai berikut:

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

Diketahui :

E = 17 N = 13 R = 2

Penyelasaian :

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

= (17 – 13) + 2

= 6

Independent Path :

Path 1 = 1 – 2 – 3 – 4 – 13

Path 2 = 1 – 2 – 3 – 5 – 6 – 13

Path 3 = 1 – 2 – 3 – 5 – 7 – 8 –13

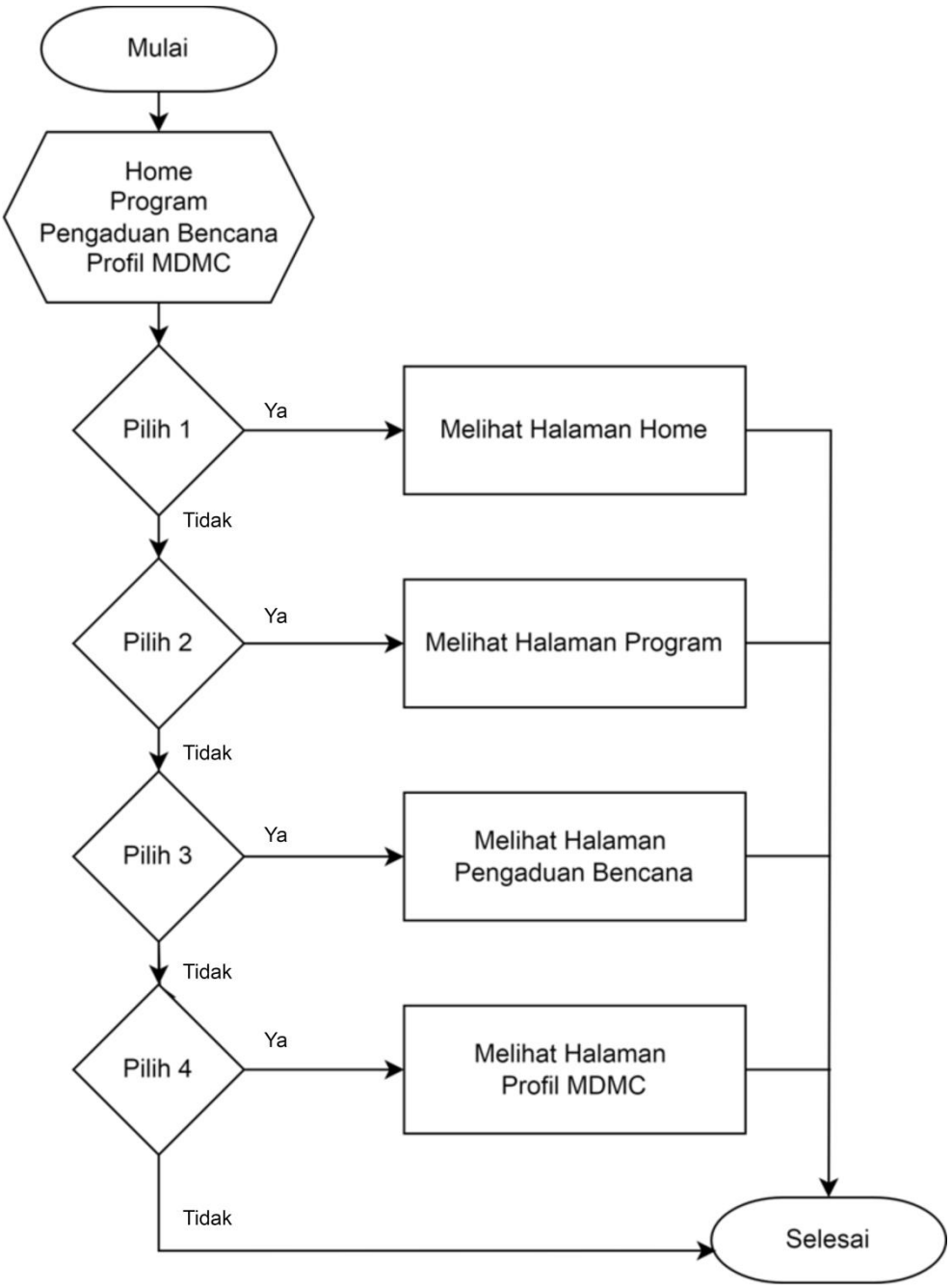
Path 4 = 1 – 2 – 3 – 5 – 7 – 9 – 10 – 13

Path 5 = 1 – 2 – 3 – 5 – 7 – 9 – 11 – 12 – 13

Tabel 4 30. *White Box* Perhitungan menu admin

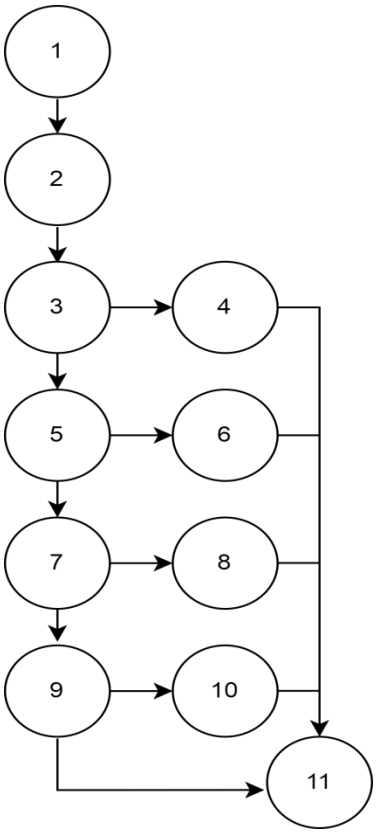
	1	2	3	4	5	6	7	8	9	10	11	12	13	E-1
1		1												1-1=0
2			1											1-1=0
3				1	1									2-1=1
4													1	1-1=0
5						1	1							2-1=1
6													1	1-1=0
7								1	1					2-1=1
8													1	1-1=0
9										1	1			2-1=1
10													1	1-1=0
11												1		1-1=0
12													1	1-1=0
13														0
SUM(E+1)														4+1=5

2) *Flowchart* dan *Flowgraph* Aktivitas *User*



Gambar 4 28. Flowchart Aktivitas *User*

Dari *Flowchart* yang digunakan *untuk* pengujian perangkat lunak, maka ditentukan *Flowgraph* sebagai berikut:



Gambar 4 29. Flowgraph Aktivitas *User*

Dari *Flowgraph* aktivitas *user* diatas dapat dilakukan proses perhitungan sebagai berikut:

b) Menghitung *Cyclomatic Complexcity* V(G) dari *Egde* dan *Node*:

Diketahui :

E = 11 V = 9 R = 2

Penyelasaian :

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

= (11 – 9) + 2

= 4

Independent Path :

Path 1 = 1 – 2 – 3 – 4 – 11

Path 2 = 1 – 2 – 3 – 5 – 6 – 11

Path 3 = 1 – 2 – 3 – 5 – 7 – 8 –11

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

Tabel 4 31. *White Box* Perhitungan menu *user*

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

7. Hasil Pengujian

Tabel 4.1. Hasil Pengujian

No	***Flowchart***	***Independent Path***	***Region***	***Cyclomatic Complexity***
1	Aktivitas *Admin*	5	5	5
2	Aktivitas *User*	4	4	4

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil penelitian dan pengembangan aplikasi Elektronik Muhammadiyah Disaster Management Center (E-MDMC) berbasis web, dapat disimpulkan bahwa aplikasi ini telah berhasil dibuat dengan menyediakan fitur yang mendukung pengelolaan data dan informasi terkait kegiatan MDMC. Aplikasi ini bertujuan untuk membantu admin MDMC dalam mengelola berbagai aspek organisasi serta memberikan kemudahan bagi masyarakat dalam mengakses informasi dan melaporkan kejadian bencana. Fitur utama yang tersedia dalam aplikasi ini mencakup. Bagi Admin MDMC dapat Mengelola data anggota MDMC sehingga pendataan lebih sistematis, Mengelola pengaduan bencana yang masuk agar dapat ditindaklanjuti dengan cepat, Mengelola berita terkait kebencanaan dan kegiatan MDMC untuk penyebaran informasi yang lebih luas, Mengelola kategori informasi untuk kemudahan klasifikasi, Mengatur berbagai pengaturan sistem dan struktur organisasi MDMC, Mengelola program dan kegiatan yang diselenggarakan oleh MDMC. Bagi User/Pengguna dapat Mengakses informasi terkait program dan kegiatan yang dilakukan oleh MDMC, Melakukan pengaduan bencana secara langsung melalui sistem dan Mengakses informasi mengenai profil MDMC, termasuk visi, misi, serta struktur organisasi. Dengan adanya aplikasi berbasis web ini, proses pengelolaan data dan komunikasi

antara MDMC dan masyarakat menjadi lebih efisien. Aplikasi ini diharapkan dapat meningkatkan respons dan koordinasi dalam penanganan bencana, serta memperkuat keterlibatan masyarakat dalam upaya mitigasi dan penanggulangan bencana.

B. Saran

Untuk pengembangan lebih lanjut, beberapa saran yang dapat diberikan adalah sebagai berikut:

1. Pengembangan Fitur Tambahan, Integrasi dengan sistem peringatan dini bencana dan Pengembangan fitur mobile-friendly agar lebih mudah diakses melalui perangkat seluler.
2. Keamanan dan Skalabilitas, Peningkatan keamanan aplikasi dengan menerapkan sistem enkripsi dan autentikasi dua faktor.
3. Kolaborasi dan Sosialisasi, Sosialisasi kepada anggota MDMC dan masyarakat luas agar aplikasi dapat dimanfaatkan secara optimal.

DAFTAR PUSTAKA

Al-Haq, Y. A. (2019). Peran MDMC (Muhammadiyah Disaster Management Center) Dalam Penanganan Pengungsi Rohingya. *Journal of Chemical Information and Modeling*, *53*(9), 1689–1699.

Arfianto, F. R., & Nugrahanti, F. (2018). Rancang Bangun Aplikasi Penjualan Perumahan Berbasis Web Pada Cv. Grand Permata Residence Magetan. *Seminar Nasional Teknologi Informasi Dan Komunikasi Literasi Digital Pada Era Revolusi Industri 4.0 SENATIK|2018|*, *2017*, 174–179.

Baghaskara, G., Putra, P. B. A. A., & Pranatawijaya, V. H. (2022). Sistem Informasi Geografis Tempat Kost Di Kota Palangka Raya Berbasis Website. *Journal of Information Technology and Computer Science*, *2*(4), 280–289. https://doi.org/10.47111/jointecoms.v2i4.8880

Faris, M., & Wisaksono, A. (2021). Pengembangan Aplikasi E-Commerce Untuk Pemasaran Biji dan Bubuk Kopi Berbasis Web. *Jurnal Janitra Informatika Dan Sistem Informasi*, *1*(1), 61–72. https://doi.org/10.25008/janitra.v1i1.116

Fasirah, N. U. R. P. (2021). PERANCANGAN SISTEM INFORMASI ORGANISASI OTONOM MUHAMMADIYAH IMM KOTA PAREPARE. *JURNAL SINTAKS LOGIKA*, *1*(1), 1–9.

Habil, N. I. (n.d.). *APLIKASI PENDAFTARAN DAN PENGOLAHAN DATA KO-KURIKULER TAPAK SUCI BERBASIS WEB PADA LP2K UNIVERSITAS MUHAMMADIYAH PAREPARE*.

Hawing, H. (2019). Muhammadiyah Disaster Management Center (MDMC) Dalam Penanggulangan Bencana Banjir Di Kota Makassar. *POPULIKA*, *7*(2), 121–128.

Hawing, H. (2021). Muhammadiyah Disaster Management Center (MDMC) dalam Penanggulangan Bencana Banjir di Kota Makassar. *Journal of Governance and Local Politics (JGLP)*, *3*(1), 63–74. https://doi.org/10.47650/jglp.v3i1.185

Hendini, A. (2016). Pemodelan Uml Sistem Informasi Monitoring Penjualan Dan Stok Barang (Studi Kasus: Distro Zhezha Pontianak). *JURNAL KHATULISTIWA INFORMATIKA*, *IV*(2), 107–116. https://doi.org/10.1145/358315.358387

Hermawansyah, A., & Nur, R. (2019). Perancangan Sistem Informasi E-Voting Pemilihan Ketua RT Dengan Verifikasi Nomor Induk Keluarga Berbasis Web. *Jurnal Sistem Informasi Volume 2 Nomor 1 April 2019*, *2*(1), 37–43.

Jayusman, Y., Apriadi, D., & Silviyani, Y. (2020). Sistem informasi geografis sebaran perumahan di kabupaten cianjur berbasis web. *Jurnal Teknologi*

Informasi Dan Komunikasi, *9*(2), 18–27.

Kusniawan, A., & Sardiarinto. (2016). Perancangan Website Jasa Desain Interior Sebagai Media Pemasaran Studi Kasus: CV. Focalpoint Interior. *Evolusi*, *4*(2), 1–10.

Lavarino, D., & Yustanti, W. (2016). Rancang Bangun E – Voting Berbasis Website Di Universitas Negeri Surabaya. *Jurnal Manajemen Informatika*, *6*(1), 72–81.

M. Ramaddan Julianti, Agus Budiman, & Agil Patriosa. (2018). Perancangan SIG Pemetaan Lokasi Apotek di Wilayah Kota Bogor Berbasis Web. *JURNAL SISFOTEK GLOBAL*, *8*(1), 13–19.

Mangero, R. D., & Abidin, G. (2022). *Rancang Bangun Sistem Informasi Desain Interior Dan Eksterior Pada Keyword : September*, 3401–3412.

MDMC. (2020). Laporan Tahunan MDMC. Muhammadiyah Disaster Management Center.

Munawir. (2019). *Implementasi Sistem Informasi Geografis Daerah Pariwisata Taman Wisata Puncak Bila Kabupaten Sidenreng Rappang Berbasis Android Dengan Global Positioning System (Gps)*.

Murdiono, I. A., & Widodo, T. (2020). *Perancangan Aplikasi Mobile Location Based Service (Lbs) Untuk Pencarian Lokasi Rumah Kos Di Kota Sleman Berbasis Android*.

Neyfa, B. C., & Tamara, D. (2016). PERANCANGAN APLIKASI E-CANTEEN BERBASIS ANDROID DENGAN MENGGUNAKAN METODE OBJECT ORIENTED ANALYSIS & DESIGN (OOAD). *Jurnal Penelitian Komunikasi Dan Opini Publik*, *20*(1), 83–91. https://doi.org/10.1136/bmj.1.6001.107

Pangeran Bungsu, A. (2023). Muhammadiyah Disaster Management Center (MDMC) dan Pemberdayaan Umat dalam Mewujudkan Solidaritas. *Journal of Da'wah*, *1*(2), 282–305. https://doi.org/10.32939/jd.v1i2.2465

Rachman. (2018). Sistem Informasi Wisata Di Ampera Waterpark. *Jurnal Siliwangi*,*4*(2),87 92.http://jurnal.unsil.ac.id/index.php/jssainstek/article/download/570/369

Rahmah, F. A. (2022). Aplikasi manajemen data relawan dan logistik Muhammadiyah Disaster Management Center (MDMC) berbasis web (Skripsi, Universitas Muhammadiyah Yogyakarta). Universitas Muhammadiyah Yogyakarta Repository.

Ressa. (2019). *Sistem Electronic Voting (E-Voting) Berbasis Web Pada Pemilihan Ketua Osis*.

Ridlo, I. A. (2017). Pedoman Pembuatan Flowchart. *Academia.Edu*, 27. academia.edu/34767055/Pedoman_Pembuatan_Flowchart

Rini. (2020). *Sistem Informasi Geografis Pemetaan Persebaran Sd Dan Smp Di Kota Palopo Berbasis Webgis*. *21*(1), 1–9.

Robihaini, Rosyadi, I., & Kusumawardani, H. H. (2022). Sistem informasi fasilitas penanganan covid-19 di kecamatan paninggaran berbasis web. *Surya Informatika*, *12*(1), 1–11.

Suwaryo, P. A. W., Sarwono, S., & Yuwono, P. (2020). Peran Muhammadiyah Disaster Management Center Dalam Mitigasi Bencana. *Jurnal Ilmiah Permas: Jurnal Ilmiah STIKES Kendal*, *10*(1), 33–40.

Sutte, A. H. Z. (2016). Aplikasi pendaftaran dan pengolahan data ko-kurikuler Tapak Suci berbasis web pada LP2K Universitas Muhammadiyah Parepare (Skripsi, Universitas Muhammadiyah Parepare). Digital Library Universitas Muhammadiyah Parepare.

Yuliani, S. T., Sudarsono, B., & Wijaya, A. P. (2016). Aplikasi Sistem Informasi Geografis (Sig) Untuk Pemetaan Pasar Tradisional Di Kota Semarang Berbasis Web. *Jurnal Geodesi Undip*, *5*(2), 208–216.