

**RANCANG BANGUN APLIKASI MYORGA
SEBAGAI PLATFORM MANAJEMEN DAN
DIGITALISASI ORGANISASI KAMPUS
BERBASIS *WEB* DAN *MOBILE***

PROYEK AKHIR

Disusun Oleh:

Nania Prima Citra Aulia

3312301056



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
POLITEKNIK NEGERI BATAM**

2026

KATA PENGANTAR

Puji dan syukur penulis panjatkan atas ke hadirat Allah SWT Tuhan Yang Maha Esa atas limpahan rahmat, karunia, serta hidayah-Nya, sehingga penyusunan laporan tugas akhir ini dapat diselesaikan oleh penulis dengan baik. Tanpa pertolongan dan penyertaan-Nya, tentu proses penyusunan proyek akhir ini tidak akan berjalan dengan lancar.

Adapun judul tugas akhir yang penulis ajukan adalah: “Rancang Bangun Aplikasi MyOrga Sebagai *Platform* Manajemen dan Digitalisasi Organisasi Kampus Berbasis *Web* dan *Mobile*”.

Penelitian ini berfokus pada pengembangan aplikasi *mobile* MyOrga sebagai platform manajemen organisasi kemahasiswaan. Aplikasi ini didesain untuk mempermudah proses manajemen organisasi, mulai dari pendaftaran anggota, pengelolaan kepengurusan, penyampaian informasi, hingga monitoring kegiatan organisasi secara lebih terstruktur, efisien, dan mudah diakses kapan saja.

Penyusunan proyek akhir ini tentunya tidak lepas dari bantuan, dukungan, serta bimbingan yang diberikan oleh berbagai pihak. Oleh karena itu, penulis ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Bapak Ir. Bambang Hendrawan, S.T., M.S.M., C.I.P.M.P., C.I.S.C.P. selaku Direktur Politeknik Negeri Batam atas dukungan terhadap penyelenggaraan pendidikan dan kegiatan akademik di Politeknik Negeri Batam.
2. Bapak Ahmad Hamim Thohari, S.S.T., M.T., selaku Ketua Jurusan Teknik Informatika Politeknik Negeri Batam atas segala kemudahan dan dukungan dalam proses akademik.
3. Ibu Yeni Rokhayati, S.Si., M.Sc., selaku Ketua Program Studi Teknik Informatika Politeknik Negeri Batam atas arahan dan fasilitas yang diberikan selama penulis menempuh masa perkuliahan.
4. Ibu Cyntia Lasmi Andesti, S.Kom., M.Kom selaku dosen pembimbing yang telah meluangkan waktu, tenaga, serta memberikan arahan dan bimbingan.
5. Bapak/Ibu dosen penguji atas kritik, saran, dan evaluasi yang membangun.
6. Keluarga tercinta atas doa, motivasi, serta dukungan yang tiada henti.

7. Rekan-rekan mahasiswa dan sahabat atas dukungan dan diskusinya.

Penulis berharap proyek akhir ini dapat memberikan manfaat, baik bagi penulis sendiri maupun bagi pembaca yang memiliki ketertarikan pada bidang yang sama. Penulis juga menyadari bahwa proyek akhir ini masih jauh dari sempurna, sehingga dengan rendah hati penulis membuka diri untuk menerima kritik dan saran demi perbaikan di masa mendatang.

Akhir kata, semoga laporan tugas akhir ini dapat memberikan manfaat serta menjadi kontribusi dalam pengembangan aplikasi *mobile* MyOrga sebagai platform manajemen organisasi kemahasiswaan yang lebih efektif dan juga terstruktur.

DAFTAR ISI

KATA PENGANTAR	2
DAFTAR ISI.....	4
DAFTAR TABEL.....	6
DAFTAR GAMBAR	7
DAFTAR LAMPIRAN.....	9
ABSTRAK	10
BAB I PENDAHULUAN	11
1.1. Latar Belakang	11
1.2. Rumusan Masalah	12
1.3. Tujuan	12
1.4. Batasan Masalah.....	13
1.5. Manfaat	13
1.6. Kolaborasi dan Sinergi dalam Pelaksanaan Proyek	14
BAB II TINJAUAN PUSTAKA.....	16
2.1. Tinjauan Solusi dan Sistem Terkait	16
2.2. Landasan Konsep dan Teknologi.....	20
2.2.1. Sistem Informasi untuk Manajemen Organisasi Kampus	20
2.2.2. <i>Global Positioning System</i> (GPS)	20
2.2.3. <i>Haversine Formula</i>	20
2.2.4. Firebase Cloud Messaging (FCM)	20
2.2.5. Notifikasi dan <i>Reminder</i>	21
2.2.6. Sistem Absensi Berbasis Lokasi	21
2.3. Metode Pengembangan Produk.....	21
BAB III ANALISIS DAN PERANCANGAN	25
3.1. Analisis Kebutuhan	25
3.2. Perancangan	31
BAB IV IMPLEMENTASI DAN PENGUJIAN.....	87
4.1. Hasil Implementasi.....	87
4.1.1 Deskripsi Proses Instalasi.....	87
4.1.2 Implementasi Antarmuka dan Fitur	88
4.1.3 Implementasi Basis Data.....	100

4.1.4 Contoh Alur Penggunaan Sistem	101
4.2. Pengujian.....	102
4.2.1 Pengujian Black Box.....	102
4.2.2 User Acceptance Testing (UAT).....	105
BAB V KESIMPULAN	107
5.1 Kesimpulan	107
5.2 Saran.....	107
DAFTAR PUSTAKA	108
DAFTAR LAMPIRAN	110

DAFTAR TABEL

Tabel 2.1 Tinjauan Aplikasi Sejenis (Pengembangan Perangkat Lunak)	18
Tabel 3.1 Kebutuhan Fungsional	31
Tabel 3.2 Kebutuhan Non-Fungsional	33
Tabel 3.3 Skenario <i>Use case Login Website</i>	35
Tabel 3.4 Skenario <i>Use case Login Mobile</i>	36
Tabel 3.5 Skenario <i>Use case Kelola Rekrutmen</i>	38
Tabel 3.6 Skenario <i>Use case Lihat Daftar Pendaftar</i>	39
Tabel 3.7 Skenario <i>Use case Lihat Daftar Anggota</i>	40
Tabel 3.8 Skenario <i>Use case Kelola Kegiatan</i>	41
Tabel 3.9 Skenario <i>Use case Kelola Absensi</i>	42
Tabel 3.10 Skenario <i>Use case Catat Notulensi</i>	43
Tabel 3.11 Skenario <i>Use case Lihat Dashboard Manajemen</i>	44
Tabel 3.12 Skenario <i>Use case Kelola Organisasi</i>	45
Tabel 3.13 Skenario <i>Use case Lihat Daftar Organisasi</i>	46
Tabel 3.14 Skenario <i>Use case Lihat Detail Organisasi</i>	47
Tabel 3.15 Skenario <i>Use case Daftar Calon Anggota</i>	48
Tabel 3.16 Skenario <i>Use case Lihat Daftar Kegiatan</i>	49
Tabel 3.17 Skenario <i>Use case Absensi</i>	50
Tabel 3.18 Skenario <i>Use case Terima Notifikasi Push</i>	51
Tabel 3.19 Skenario <i>Use case Logout</i>	52
Tabel 4.1 Tabel Pengujian <i>Dashboard Web</i>	103

DAFTAR GAMBAR

Gambar 2.1 Model <i>Waterfall</i>	22
Gambar 3.1 Diagram Alir Proses Rekrutmen Sebelum	25
Gambar 3.2 Diagram Alir Bisnis Rekrutmen Sesudah	26
Gambar 3.3 Diagram Alir Bisnis Kegiatan Sebelum	28
Gambar 3.4 Diagram Alir Bisnis Kegiatan Sesudah.....	29
Gambar 3.5. Gambaran Umum Sistem MyOrga.....	30
Gambar 3.6 Diagram <i>Use case</i> Aplikasi	34
Gambar 3.7 Diagram Aktivitas <i>Login Website</i>	54
Gambar 3.8 Diagram Aktivitas <i>Login Mobile</i>	55
Gambar 3.9 Diagram Aktivitas Kelola Rekrutmen.....	56
Gambar 3.10 Diagram Aktivitas Lihat Daftar Pendaftar	57
Gambar 3.11 Diagram Aktivitas Lihat Daftar Anggota.....	58
Gambar 3.12 Diagram Aktivitas Kelola Kegiatan	59
Gambar 3.13 Diagram Aktivitas Kelola Absensi.....	60
Gambar 3.14 Diagram Aktivitas Catat Notulensi	61
Gambar 3.15 Diagram Aktivitas Kelola Organisasi	62
Gambar 3.16 Diagram Aktivitas Lihat <i>Dashboard</i> Manajemen.....	63
Gambar 3.17 Diagram Aktivitas Lihat Daftar Organisasi	64
Gambar 3.18 Diagram Aktivitas Lihat Detail Organisasi.....	65
Gambar 3.19 Diagram Aktivitas Daftar Calon Anggota.....	66
Gambar 3.20 Diagram Aktivitas Lihat Daftar Kegiatan	67
Gambar 3.21 Diagram aktivitas Absensi	68
Gambar 3.22 Diagram Aktivitas Terima Notifikasi Push.....	69
Gambar 3.23 Diagram Aktivitas <i>Logout</i>	70
Gambar 3.24 ER Diagram Aplikasi	71
Gambar 3.25 Desain Antarmuka Halaman <i>Login</i>	74
Gambar 3.26 Desain Antarmuka Halaman <i>Dashboard</i> Manajemen	74
Gambar 3.27 Desain Antarmuka Halaman Profil Organisasi	75

Gambar 3.28 Desain Antarmuka Halaman <i>Dashboard</i> Pengurus	76
Gambar 3.29 Desain Antarmuka Halaman Anggota.....	76
Gambar 3.30 Desain Antarmuka Halaman Periode Rekrutmen	77
Gambar 3.31 Desain Antarmuka Halaman Data Pendaftar	77
Gambar 3.32 Desain Antarmuka Halaman Kegiatan.....	78
Gambar 3.33 Desain Antarmuka Halaman <i>Splashscreen</i>	79
Gambar 3.34 Desain Antarmuka Halaman <i>Login</i>	80
Gambar 3.35 Desain Antarmuka Halaman <i>Homepage</i>	81
Gambar 3.36 Desain Antarmuka Halaman Kegiatan.....	82
Gambar 3.37 Desain Antarmuka Halaman Absensi	83
Gambar 3.38 Desain Antarmuka Halaman Profil	84
Gambar 3.39 Desain Antarmuka Halaman <i>Landing page</i>	85
Gambar 3.40 Desain Antarmuka Halaman Detail Organisasi	86
Gambar 4.1 Implementasi Antarmuka <i>Login Website</i>	88
Gambar 4.2 Implementasi Antarmuka <i>Dashboard</i> Manajemen	88
Gambar 4.3 Implementasi Antarmuka Profil Organisasi	89
Gambar 4.4 Implementasi Antarmuka <i>Dashboard</i> Pengurus	90
Gambar 4.5 Implementasi Antarmuka Anggota	90
Gambar 4.6 Implementasi Antarmuka Periode Rekrutmen	91
Gambar 4.7 Implementasi Antarmuka Data Pendaftar	91
Gambar 4.8 Implementasi Antarmuka Kegiatan.....	92
Gambar 4.9 Implementasi Antarmuka <i>Splashscreen</i>	93
Gambar 4.10 Implementasi Antarmuka <i>Login Mobile</i>	94
Gambar 4.11 Implementasi Antarmuka <i>Homepage</i>	95
Gambar 4.12 Implementasi Antarmuka Kegiatan.....	96
Gambar 4.13 Implementasi Antarmuka Absensi	97
Gambar 4.14 Implementasi Antarmuka Profil	98
Gambar 4.15 Implementasi Antarmuka <i>Landing page</i>	99
Gambar 4.16 Implementasi Antarmuka Detail Organisasi	99
Gambar 4.17 Implementasi Basis Data	100

DAFTAR LAMPIRAN

Lampiran A. Dokumen Proses Pengumpulan Kebutuhan Pengguna.....	110
Lampiran B. Link Produk	110
Lampiran C. Dokumen Pengujian.....	110
Lampiran D. Link Desain Antarmuka.....	110

ABSTRAK

Pengelolaan organisasi kemahasiswaan masih banyak dilakukan secara *manual*, seperti penyampaian informasi, pencatatan absensi, serta proses rekrutmen anggota yang menyebabkan informasi sulit diakses secara cepat dan pengelolaan data menjadi kurang efisien. Oleh karena itu, proyek akhir ini bertujuan untuk mengembangkan aplikasi *mobile* dan *web* bernama MyOrga sebagai platform manajemen organisasi kemahasiswaan yang terintegrasi. Sistem ini dirancang untuk membantu proses pengelolaan organisasi, mulai dari pengelolaan data organisasi, rekrutmen anggota, absensi kegiatan, hingga penyampaian informasi organisasi secara *digital*. Metode pengembangan perangkat lunak yang digunakan dalam proyek ini adalah metode *Waterfall* yang meliputi tahap analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan. Aplikasi dikembangkan menggunakan *framework* Laravel pada sisi *web* dan Flutter pada sisi *mobile*. Hasil dari proyek akhir ini berupa aplikasi manajemen organisasi yang dapat digunakan oleh manajemen, pengurus, dan anggota organisasi sesuai dengan hak akses masing-masing. Sistem yang dikembangkan mampu membantu proses administrasi organisasi menjadi lebih terstruktur, efektif, dan mudah diakses kapan saja. Berdasarkan hasil pengujian menggunakan metode *System Usability Scale* (SUS) kepada 33 responden, aplikasi MyOrga memperoleh nilai rata-rata sebesar 92,58 yang termasuk dalam kategori *Excellent* dengan grade A. Hasil tersebut menunjukkan bahwa aplikasi memiliki tingkat usability yang sangat baik, mudah dipahami, mudah digunakan, serta mampu memberikan pengalaman penggunaan yang baik bagi pengguna. Dengan adanya aplikasi MyOrga, diharapkan proses pengelolaan organisasi kemahasiswaan dapat berjalan lebih efisien serta meningkatkan kemudahan akses informasi bagi seluruh pengguna..

Kata Kunci: Organisasi Kemahasiswaan, Aplikasi *Mobile*, Flutter, Laravel, Manajemen Organisasi

BAB I PENDAHULUAN

1.1. Latar Belakang

Organisasi mahasiswa merupakan wadah yang berperan penting dalam mengembangkan potensi, minat, dan kemampuan mahasiswa, baik dalam bidang kepemimpinan, komunikasi, manajemen, maupun kerja sama tim. Keaktifan mahasiswa dalam organisasi juga memberikan manfaat dalam melatih kemampuan *problem solving*, *public speaking*, serta memperluas wawasan dan relasi sosial mahasiswa (Yusuf Amir & Lestari Perdana, 2024). Selain sebagai sarana pengembangan *soft skill*, organisasi mahasiswa juga menjadi media pelaksanaan berbagai kegiatan akademik maupun non-akademik di lingkungan perguruan tinggi.

Dalam pelaksanaannya, pengelolaan organisasi mahasiswa melibatkan berbagai aktivitas administratif seperti penyebaran informasi kegiatan, rekrutmen anggota, pengelolaan program kerja, pencatatan notulensi rapat, hingga *monitoring* kehadiran anggota. Namun, pada banyak organisasi mahasiswa, proses pengelolaan tersebut masih dilakukan secara *manual* atau menggunakan media yang terpisah-pisah, sehingga informasi sering kali tidak terintegrasi dengan baik. Kondisi ini dapat menyebabkan keterlambatan penyampaian informasi, kesulitan pengelolaan data anggota, risiko kehilangan arsip kegiatan, serta kurang efektifnya monitoring aktivitas organisasi (Putri dkk., 2022). Selain itu, pengelolaan administrasi organisasi yang masih dilakukan secara *manual* juga berpotensi menimbulkan ketidakefisienan dalam pelaksanaan kegiatan dan pengarsipan data organisasi (Yusuf Amir & Lestari Perdana, 2024).

Seiring dengan perkembangan teknologi informasi, kebutuhan akan sistem manajemen organisasi berbasis *digital* menjadi semakin penting. Sistem informasi organisasi berbasis *web* dan *mobile* dapat membantu pengelolaan data organisasi secara lebih terstruktur, terintegrasi, dan efisien. Selain mempermudah penyampaian informasi dan pengelolaan kegiatan organisasi, sistem *digital* juga dapat meningkatkan efektivitas administrasi, monitoring aktivitas anggota, serta penyimpanan arsip organisasi secara terpusat (Repi dkk., 2021).

Pencatatan kehadiran anggota pada kegiatan maupun rapat organisasi merupakan salah satu aspek penting dalam pengelolaan organisasi. Dalam era *digital*, sistem absensi berbasis lokasi menjadi solusi yang banyak diterapkan karena mampu meningkatkan akurasi dan transparansi pencatatan kehadiran. Validasi lokasi menggunakan *Global Positioning System* (GPS) dan metode *Haversine Formula* memungkinkan sistem menghitung jarak pengguna dengan titik lokasi yang telah ditentukan secara real-time. Penerapan metode ini terbukti efektif dalam memverifikasi kehadiran pengguna berdasarkan posisi geografis dan mengurangi potensi manipulasi absensi (Anita & Rahmawati, 2025). Penelitian lain juga menunjukkan bahwa implementasi algoritma Haversine pada sistem absensi berbasis *mobile* mampu meningkatkan akurasi, efisiensi, serta transparansi dalam pengelolaan kehadiran pengguna (Fauzan dkk., 2025).

Berdasarkan permasalahan yang dipaparkan di atas, diperlukan pengembangan sistem manajemen organisasi kampus berbasis *web* dan *mobile* yang mampu mendukung digitalisasi pengelolaan organisasi secara terintegrasi. Sistem berbasis *web* difokuskan sebagai media informasi dan *dashboard* manajerial organisasi, sedangkan aplikasi *mobile* Android digunakan oleh anggota untuk mengakses informasi kegiatan, menerima pengingat, serta melakukan absensi berbasis lokasi menggunakan GPS dan *Haversine Formula*. Dengan adanya sistem ini, diharapkan proses pengelolaan organisasi mahasiswa dapat berjalan lebih efektif, efisien, dan terorganisasi.

1.2. Rumusan Masalah

Bagaimana mendesain dan membangun Sistem Informasi Manajemen Organisasi Kampus berbasis *website* dan *mobile* yang dapat mengelola data organisasi, pengurus, dan anggota secara terintegrasi?

1.3. Tujuan

Merancang dan membangun Sistem Informasi Manajemen Organisasi Kampus berbasis *web* dan *mobile* yang terintegrasi.

1.4. Batasan Masalah

Batasan masalah dari Sistem Informasi Manajemen Organisasi Kampus berbasis *web* dan *mobile* adalah sebagai berikut.

1. Pengembangan aplikasi *mobile* difokuskan pada *platform* Android.
2. Mahasiswa hanya dapat menjadi anggota dalam satu organisasi pada tahap pengembangan sistem.
3. Sistem diuji dalam lingkup lokal (*development environment*) dan belum mencakup implementasi skala produksi dengan beban pengguna yang besar.
4. Sistem belum terintegrasi dengan sistem akademik kampus atau database resmi institusi.

1.5. Manfaat

Manfaat yang diharapkan dari Sistem Informasi Manajemen Organisasi Kampus berbasis *web* dan *mobile* adalah sebagai berikut.

1. Menjadi referensi dalam pengembangan sistem informasi organisasi kemahasiswaan berbasis *mobile* dan *web* serta menambah pemahaman mengenai penerapan framework Laravel dan Flutter dalam pengembangan aplikasi terintegrasi.
2. Membantu pihak manajemen dalam mengelola data organisasi secara lebih terpusat, efektif, dan efisien.
3. Mempermudah pengurus organisasi dalam mengelola kegiatan, anggota, rekrutmen, serta absensi secara *digital*.
4. Membantu anggota organisasi dalam mengakses informasi, melakukan absensi, menerima notifikasi, serta memantau aktivitas organisasi melalui aplikasi *mobile*.
5. Mengurangi proses administrasi *manual* sehingga proses pengelolaan organisasi menjadi lebih terstruktur dan mudah diakses kapan saja.
6. Mendukung Sustainable Development Goals (SDGs) tujuan ke-4, yaitu Quality Education, dengan membantu menciptakan lingkungan organisasi kemahasiswaan yang lebih terorganisir, kolaboratif, dan mendukung pengembangan kemampuan mahasiswa.

7. Mendukung Sustainable Development Goals (SDGs) tujuan ke-9, yaitu Industry, Innovation, and Infrastructure, melalui pengembangan aplikasi berbasis teknologi *digital* untuk mendukung inovasi dalam pengelolaan organisasi kemahasiswaan.

1.6. Kolaborasi dan Sinergi dalam Pelaksanaan Proyek

Proyek Akhir ini dilaksanakan secara mandiri. Seluruh tahapan siklus pengembangan sistem, mulai dari analisis kebutuhan, perancangan arsitektur basis data dan antarmuka, implementasi kode program (pengembangan *web backend* dan aplikasi *mobile Android*), hingga tahap pengujian fungsional dan penyusunan dokumentasi teknis, sepenuhnya menjadi tanggung jawab pelaksana proyek.

Meskipun dilaksanakan secara individu dan tidak berada di bawah program magang industri, proyek ini tetap mengedepankan kolaborasi dan sinergi dengan berbagai pihak terkait untuk memastikan kualitas, kelayakan, dan relevansi solusi yang dihasilkan. Bentuk kolaborasi yang dilakukan selama pelaksanaan proyek antara lain:

1. Konsultasi dengan dosen pembimbing, Ibu Cyntia Lasmi Andesti, S.Kom., M.Kom. Hal ini bertujuan untuk memantau progres pengembangan sistem, mencari solusi atas kendala teknis, serta memastikan kesesuaian metodologi dan kelengkapan dokumentasi dengan standar akademik program studi.
2. Pengumpulan Umpan Balik dari Calon Pengguna. Melibatkan target pengguna akhir sistem, yaitu perwakilan mahasiswa yang berperan sebagai pengurus maupun anggota organisasi kampus. Kolaborasi ini direalisasikan pada tahap pengujian sistem (*User Acceptance Testing*) untuk mengumpulkan umpan balik terkait fungsionalitas dan tingkat kemudahan (*usability*) aplikasi MyOrga, guna memastikan sistem yang dibangun benar-benar menjawab kebutuhan operasional organisasi di lapangan.

Bentuk kolaborasi ini menunjukkan bahwa meskipun proyek dikerjakan secara mandiri, terdapat sinergi yang kuat dengan pemangku kepentingan utama. Keterlibatan aktif dari dosen pembimbing dan umpan balik langsung dari calon

pengguna menjadi landasan penting dalam menjamin keberhasilan dan kualitas akhir dari proyek ini.

BAB II TINJAUAN PUSTAKA

2.1. Tinjauan Solusi dan Sistem Terkait

Berikut merupakan tinjauan aplikasi terkait yang menjadi acuan dalam pengembangan sistem manajemen organisasi kampus berbasis *web* dan *mobile*. Penelitian terkait ini digunakan sebagai landasan akademis untuk memahami pendekatan, teknologi, serta fitur yang telah diterapkan pada penelitian sebelumnya, sekaligus menjadi pembanding dan pembeda terhadap sistem yang dikembangkan dalam penelitian ini.

Penelitian pertama dilakukan oleh Anita & Rahmawati (2025) dengan judul “Pengembangan Sistem Presensi Online Berbasis *Web* Dengan Validasi Lokasi Menggunakan GPS dan *Haversine Formula*.” Penelitian ini mengembangkan sistem presensi berbasis *web* dengan fitur validasi lokasi menggunakan GPS dan rumus Haversine untuk memastikan pengguna berada di dalam radius yang telah ditentukan. Sistem juga dilengkapi fitur rekap presensi dan pengajuan cuti secara *digital*. Hasil penelitian menunjukkan bahwa sistem dapat mengurangi manipulasi data kehadiran dan meningkatkan efisiensi administrasi presensi. Namun, sistem yang dikembangkan masih terbatas pada platform *web-browser* sehingga fleksibilitas penggunaan pada perangkat *mobile* belum optimal (Anita & Rahmawati, 2025).

Penelitian kedua dilakukan oleh Amir Alkodri dkk. (2025) dengan judul “*Optimization of Mobile Attendance System with Haversine Formula Method for Field Work Practice Students*.” Penelitian ini berfokus pada pengembangan sistem absensi *mobile* berbasis Android dengan validasi lokasi secara real-time menggunakan algoritma *Haversine*. Sistem mampu menghitung jarak koordinat pengguna dengan tingkat akurasi tinggi pada radius di bawah 50 meter sehingga dapat meminimalkan potensi kecurangan absensi. Hasil penelitian menunjukkan bahwa penggunaan GPS dan *Haversine Formula* mampu meningkatkan transparansi dan akurasi pencatatan kehadiran. Akan tetapi, penelitian ini lebih berfokus pada sistem absensi praktik kerja lapangan dan belum mencakup fitur

pengelolaan organisasi maupun manajemen kegiatan secara terintegrasi (Amir Alkodri dkk., 2025).

Penelitian ketiga dilakukan oleh Putri dkk. (2022) dengan judul “Perancangan Sistem Informasi Manajemen Organisasi Mahasiswa Universitas Islam Negeri Sultan Syarif Kasim Riau.” Penelitian ini mengembangkan sistem informasi organisasi mahasiswa berbasis *web* yang menyediakan fitur informasi profil organisasi, pendaftaran kegiatan, dan rekomendasi organisasi bagi mahasiswa. Hasil penelitian menunjukkan bahwa sistem efektif dalam menyediakan informasi organisasi secara menyeluruh dan meningkatkan efisiensi penyelenggaraan kegiatan organisasi mahasiswa. Namun, sistem yang dikembangkan belum memiliki fitur manajemen absensi kegiatan berbasis lokasi sehingga monitoring kehadiran anggota belum dapat dilakukan secara otomatis dan akurat (Putri dkk., 2022).

Penelitian keempat dilakukan oleh Repi dkk. (2021) dengan judul “Rancang Bangun Aplikasi Manajemen Organisasi Online Berbasis *Mobile*.” Penelitian ini merancang aplikasi *mobile* berbasis React Native dan Firebase Realtime Database untuk membantu pengelolaan keanggotaan organisasi, data organisasi, serta penjadwalan kegiatan program kerja. Hasil penelitian menunjukkan bahwa aplikasi dapat mempermudah pengolahan data organisasi secara lebih efisien melalui perangkat *mobile*. Akan tetapi, aplikasi yang dikembangkan belum mendukung validasi presensi berbasis lokasi menggunakan GPS sehingga belum dapat digunakan untuk memastikan kehadiran anggota secara akurat pada lokasi kegiatan tertentu (Repi dkk., 2021).

Penelitian kelima dilakukan oleh Harits dkk. (2021) dengan judul “Pengembangan Sistem Informasi Rekrutmen Anggota Organisasi Berbasis *Web* dengan Pemberitahuan dan Konfirmasi Memanfaatkan SMTP, SMS, dan Whatsapp Gateway.” Penelitian ini mengembangkan sistem rekrutmen anggota organisasi berbasis *web* dengan fitur pendaftaran terpusat, seleksi anggota, serta notifikasi otomatis menggunakan SMTP, SMS, dan Whatsapp Gateway. Hasil penelitian menunjukkan bahwa sistem dapat mempermudah penyebaran informasi rekrutmen dan mempermudah pengelolaan data pendaftar organisasi. Namun, sistem yang

dikembangkan hanya berfokus pada proses rekrutmen anggota dan belum mencakup fitur manajemen kegiatan organisasi maupun absensi anggota setelah proses penerimaan selesai (Harits dkk., 2021).

Berdasarkan beberapa penelitian terdahulu tersebut, dapat disimpulkan bahwa sebagian penelitian berfokus pada pengembangan sistem absensi berbasis GPS dan *Haversine Formula*, sedangkan penelitian lainnya berfokus pada sistem informasi organisasi dan rekrutmen anggota. Namun, belum terdapat penelitian yang mengintegrasikan fitur manajemen organisasi, pengelolaan kegiatan, rekrutmen anggota, *reminder* kegiatan, serta absensi berbasis lokasi GPS dan *Haversine Formula* dalam satu sistem berbasis *web* dan *mobile*. Oleh karena itu, penelitian ini memiliki tujuan untuk mengembangkan sistem manajemen organisasi kampus terintegrasi yang dapat mendukung kebutuhan administrasi organisasi sekaligus meningkatkan efektivitas monitoring kehadiran anggota secara *digital*.

Tabel 2.1 Tinjauan Aplikasi Sejenis (Pengembangan Perangkat Lunak)

Pengembang/ Tahun	Nama Aplikasi	Fitur Utama	Teknologi	Kelebihan	Keterbatasan
(Anita & Rahmawati, 2025)	Pengembangan Sistem Presensi Online Berbasis Web Dengan Validasi Lokasi Menggunakan GPS dan <i>Haversine Formula</i>	Validasi lokasi GPS, rekap presensi, pengajuan cuti	Web, PHP, MySQL, Algoritma Haversine	Mengurangi manipulasi data kehadiran dan lebih efisien	Akses presensi terbatas pada platform <i>web-browser</i>
(Amir Alkodri dkk., 2025)	<i>Optimization of Mobile Attendance System with Haversine Formula Method for Field Work Practice Students</i>	Perekaman lokasi <i>real-time</i> , validasi otomatis koordinat GPS	Android, <i>Haversine Formula</i>	Jarak dihitung dengan akurasi tinggi di bawah radius 50m	Performa menurun
(Putri dkk., 2022)	Perancangan Sistem Informasi Manajemen Organisasi Mahasiswa Universitas Islam Negeri	Informasi profil organisasi, pendaftaran event, rekomendasi	Web, Django, Python	Menyediakan informasi menyeluruh bagi mahasiswa umum	Tidak memiliki fitur manajemen absensi kegiatan berbasis lokasi

Pengembang/ Tahun	Nama Aplikasi	Fitur Utama	Teknologi	Kelebihan	Keterbatasan
	Sultan Syarif Kasim Riau				
(Repi dkk., 2021)	Rancang Bangun Aplikasi Manajemen Organisasi <i>Online</i> Berbasis <i>Mobile</i>	Pengelolaan keanggotaan, data organisasi, dan penjadwalan kegiatan program	<i>Mobile</i> (React Native, Firebase Realtime Database)	Memfasilitasi pengolahan data organisasi secara efisien melalui genggaman aplikasi <i>mobile</i>	Belum terintegrasi dengan validasi presensi absensi berbasis lokasi (GPS).
(Harits dkk., 2021)	Pengembangan Sistem Informasi Rekrutmen Anggota Organisasi berbasis <i>Web</i> dengan Pemberitahuan dan Konfirmasi Memanfaatkan Simple Mail Transfer Protocol (SMTP), SMS, dan Whatsapp Gateway (Studi Kasus Universitas Brawijaya)	Pendaftaran terpusat, seleksi, notifikasi kelulusan	Berbasis <i>Web</i> , SMTP, SMS & Whatsapp Gateway	Penyebaran informasi rekrutmen terpusat dengan notifikasi real- time yang menjangkau banyak pendaftar	Hanya berfokus pada tahapan rekrutmen dan pendaftaran, belum mencakup manajemen kegiatan setelah anggota diterima.
Proyek ini (2026)	Rancang Bangun Aplikasi Myorga Sebagai Platform Manajemen Dan Digitalisasi Organisasi Kampus Berbasis <i>Web</i> Dan <i>Mobile</i>	Manajemen rekrutmen, presensi GPS (Haversine) & fallback swafoto, notifikasi FCM, tugas, notulensi	<i>Web</i> (Laravel, Reack), <i>Mobile</i> (Flutter, Android), SMTP, Firebase Cloud Messaging	Sistem hibrida yang mengintegrasikan administrasi (<i>Web</i>) dan kemudahan absensi akurat berlapis (<i>Mobile</i> & GPS	Implementasi sistem masih dibatasi pada lingkup satu keanggotaan mahasiswa per satu organisasi.

2.2. Landasan Konsep dan Teknologi

2.2.1. Sistem Informasi untuk Manajemen Organisasi Kampus

Sistem informasi untuk manajemen organisasi kampus merupakan platform *digital* yang digunakan untuk mendukung pengelolaan aktivitas organisasi mahasiswa secara terintegrasi, seperti pengelolaan anggota, kegiatan, divisi, rekrutmen, dan arsip organisasi. Penggunaan sistem informasi dapat membantu meningkatkan efisiensi administrasi, mempercepat penyampaian informasi, serta mempermudah monitoring kegiatan organisasi dibandingkan pengelolaan *manual* (Putri dkk., 2022).

2.2.2. Global Positioning System (GPS)

Global Positioning System (GPS) merupakan sistem navigasi berbasis satelit yang menggunakan koordinat lintang dan bujur untuk menentukan lokasi pengguna secara waktu nyata. Untuk meningkatkan akurasi dan transparansi data kehadiran, sistem absensi digital memanfaatkan GPS untuk memverifikasi lokasi pengguna sehingga proses absensi hanya dapat dilakukan pada lokasi yang telah ditentukan. (Anita & Rahmawati, 2025).

2.2.3. Haversine Formula

Rumus *Haversine* merupakan rumus matematika yang menggunakan koordinat lintang dan bujur untuk menentukan jarak terpendek antara dua titik koordinat di permukaan Bumi. Algoritma ini sering digunakan pada sistem absensi berbasis lokasi karena mampu menentukan secara akurat apakah pengguna berada dalam radius yang telah ditentukan untuk melakukan proses absensi (Amir Alkodri dkk., 2025).

2.2.4. Firebase Cloud Messaging (FCM)

Firebase Cloud Messaging (FCM) merupakan layanan pengiriman pesan lintas platform yang dikembangkan oleh Google dan berfungsi sebagai arsitektur perantara untuk mengirimkan pesan data maupun notifikasi dari server backend ke aplikasi klien secara andal dan efisien (Firebase, 2026). Pada sistem ini, FCM

digunakan untuk mengirimkan notifikasi atau pengingat mengenai informasi penting dan jadwal kegiatan secara langsung ke perangkat Android milik anggota.

2.2.5. Notifikasi dan *Reminder*

Notifikasi adalah pesan singkat yang dikirimkan oleh aplikasi kepada pengguna dengan tujuan memberikan informasi secara cepat dan *real-time* (Android Developers, 2025). Notifikasi dapat berupa pemberitahuan adanya informasi baru, peringatan sistem, maupun *update* aktivitas pengguna.

Reminder merupakan bentuk pesan pengingat yang berfungsi untuk membantu pengguna mengingat suatu aktivitas atau agenda yang telah ditentukan sebelumnya (Sukmaningsih, 2025). Dalam konteks penelitian ini, *reminder* difokuskan pada kegiatan berupa rapat atau kegiatan lainnya. *Reminder* dibuat melalui *dashboard* admin dengan menentukan jadwal kegiatan, menetapkan sifat jadwal (sekali atau berulang), serta memilih penerima *reminder*, baik kepada biro tertentu maupun pegawai secara individu. *Reminder* yang telah dijadwalkan kemudian akan dikirimkan dalam bentuk notifikasi kepada aplikasi *mobile* sehingga pengguna tidak melewatkan jadwal kegiatan yang telah ditentukan.

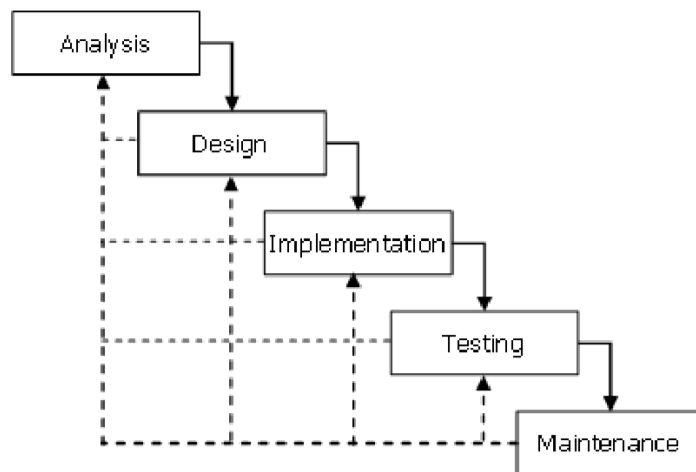
2.2.6. Sistem Absensi Berbasis Lokasi

Sistem absensi berbasis lokasi merupakan sistem presensi *digital* yang memanfaatkan teknologi GPS dan geolokasi untuk memvalidasi posisi pengguna saat melakukan absensi. Penerapan sistem ini dapat membantu mengurangi manipulasi kehadiran, meningkatkan akurasi data presensi, serta mendukung monitoring kehadiran secara real-time (Anita & Rahmawati, 2025).

2.3. Metode Pengembangan Produk

Metodologi pengembangan sistem yang digunakan dalam penelitian ini adalah System Development Life Cycle (SDLC) dengan model *Waterfall*. Model *Waterfall* merupakan pendekatan pengembangan perangkat lunak yang dilakukan secara sistematis dan berurutan, di mana setiap tahapan harus diselesaikan terlebih dahulu sebelum melanjutkan ke tahapan berikutnya (Firmana dkk., 2024).

Model *Waterfall* yang ditunjukkan pada Gambar 2.1 dipilih karena menyediakan tahapan pengembangan yang sistematis dan berurutan, sehingga memudahkan proses perencanaan, dokumentasi, dan pengembangan sistem secara bertahap. Selain itu, metode ini juga sesuai digunakan pada penelitian yang memiliki kebutuhan sistem yang telah dijabarkan dengan jelas sejak awal proses pengembangan.



Gambar 2.1 Model *Waterfall*

1. Analisis Kebutuhan

Tahapan ini memiliki tujuan untuk mengidentifikasi kebutuhan pengguna secara lengkap. Untuk memahami kebutuhan sistem yang akan dikembangkan, dilakukan observasi terhadap proses pengelolaan organisasi kemahasiswaan yang masih dilakukan secara *manual*, seperti pengelolaan anggota, absensi kegiatan, penyampaian informasi, dan proses rekrutmen anggota baru. Selain itu, dilakukan diskusi dengan calon pengguna sistem, yaitu pihak manajemen, pengurus organisasi, dan anggota organisasi, untuk mengetahui kendala yang dihadapi serta fitur-fitur yang dibutuhkan dalam aplikasi. Hasil dari tahapan ini berupa dokumen spesifikasi kebutuhan sistem yang menjadi kerangka dalam proses perancangan aplikasi MyOrga.

2. Perancangan

Langkah ini menerjemahkan kebutuhan sistem ke dalam bentuk rancangan yang lebih terstruktur. Perancangan basis data dilakukan menggunakan

Entity Relationship Diagram (ERD) yang mencakup entitas organisasi, anggota, kegiatan, absensi, rekrutmen, dan notifikasi. Selanjutnya, dirancang arsitektur sistem berbasis client-server menggunakan kerangka kerja (*framework*) Laravel untuk *website* dan Flutter untuk *mobile*. Selain itu, dibuat pula rancangan antarmuka pengguna yang menggambarkan halaman *login*, *dashboard*, pengelolaan organisasi, rekrutmen anggota, absensi kegiatan, dan notifikasi. Desain antarmuka dirancang agar mudah digunakan sesuai kebutuhan masing-masing pengguna berdasarkan hak aksesnya.

3. Implementasi

Tahap ini menerjemahkan hasil perancangan ke dalam bentuk kode program yang dapat dijalankan. Pengkodean aplikasi *web* dilakukan menggunakan *framework* Laravel, sedangkan aplikasi *mobile* dikembangkan menggunakan Flutter. Basis data MySQL dibangun sesuai dengan rancangan ERD yang telah dibuat sebelumnya. Fitur-fitur utama seperti pengelolaan organisasi, pengelolaan anggota, rekrutmen anggota, absensi kegiatan, notifikasi, dan manajemen hak akses pengguna dikembangkan secara bertahap. Selain itu, sistem juga mendukung upload file seperti logo organisasi dan dokumentasi kegiatan menggunakan media penyimpanan sistem.

4. Pengujian

Tahapan ini bertujuan untuk memvalidasi bahwa sistem telah berfungsi sesuai dengan kebutuhan dan fitur yang sudah direncanakan di awal. Pengujian dilakukan menggunakan pendekatan *Black Box Testing* dengan menyusun skenario pengujian berdasarkan kebutuhan fungsional sistem. Pengujian mencakup fungsi *login*, pengelolaan organisasi, rekrutmen anggota, absensi kegiatan, pengelolaan anggota, serta notifikasi kepada pengguna. Selain itu, pengujian juga dilakukan menggunakan metode *System Usability Scale* (SUS) untuk mengevaluasi tingkat kegunaan dan kepuasan pengguna terhadap aplikasi MyOrga. Pada pengujian SUS, sejumlah responden mencoba menggunakan sistem, kemudian memberikan

penilaian melalui kuesioner. Hasil pengujian tersebut digunakan sebagai bahan evaluasi untuk memastikan bahwa aplikasi dapat digunakan secara efektif, efisien, serta memenuhi kebutuhan pengguna.

5. Pemeliharaan

Setelah tahap pengembangan selesai, tahap ini dilakukan untuk memastikan bahwa sistem tetap berfungsi dengan baik. Tahap pemeliharaan meliputi perbaikan kesalahan yang ditemukan selama penggunaan aplikasi, peningkatan efisiensi sistem, dan modifikasi fungsionalitas sesuai dengan kebutuhan perusahaan. Untuk menjamin keamanan data pengguna, pencadangan data secara berkala juga dilakukan. Dengan adanya tahap pemeliharaan, diharapkan program MyOrga dapat terus digunakan secara optimal dan membantu administrasi organisasi kemahasiswaan dengan cara yang lebih efisien dan sistematis.

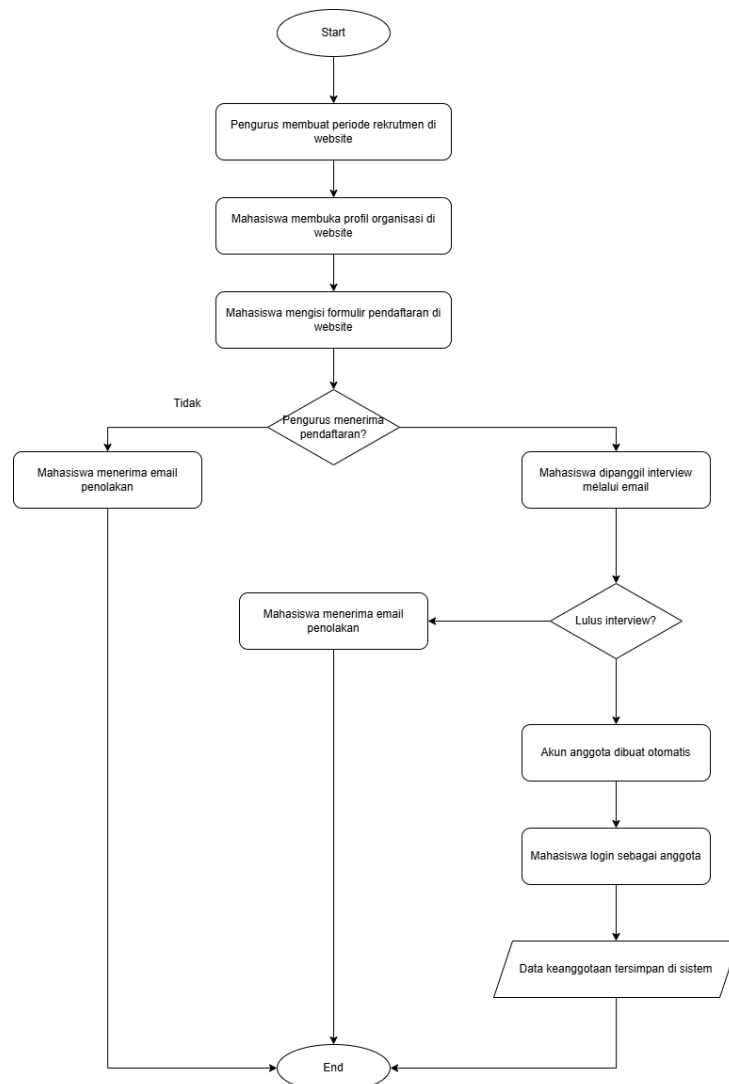
.

BAB III ANALISIS DAN PERANCANGAN

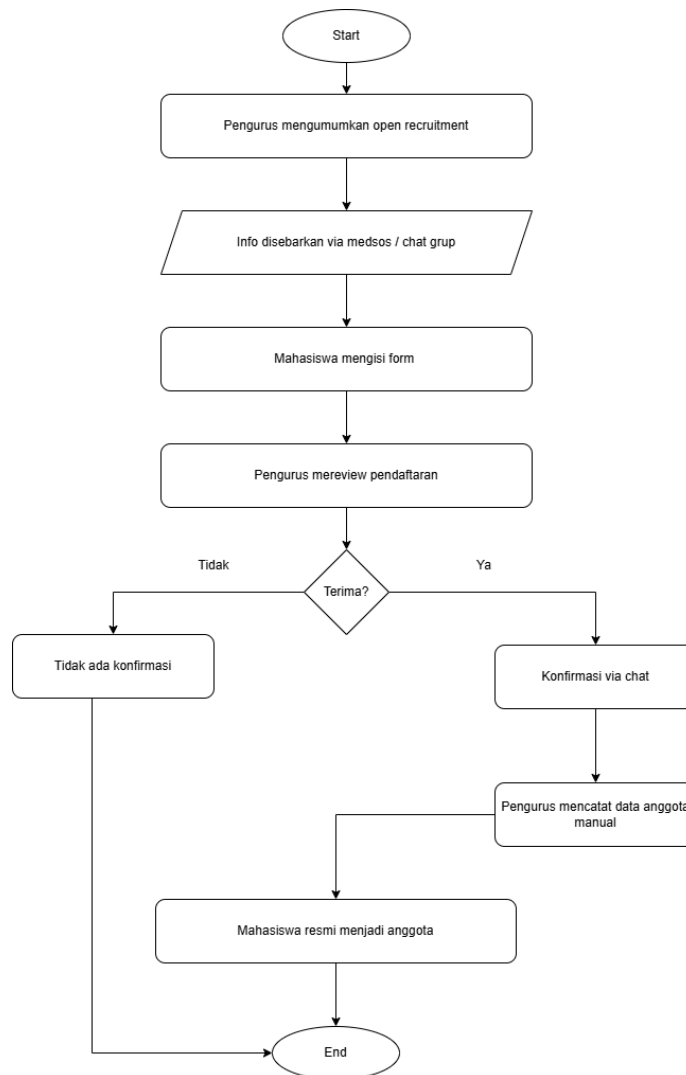
3.1. Analisis Kebutuhan

Kebutuhan pengguna dan proses bisnis yang sedang berjalan di dalam organisasi kemahasiswaan diidentifikasi pada tahap analisis kebutuhan sistem. Tahap ini bertujuan untuk memahami permasalahan yang ada serta mengidentifikasi kebutuhan sistem yang perlu ditetapkan sebagai dasar dalam merancang alur sistem, struktur data, dan antarmuka pengguna pada tahap selanjutnya.

3.1.1 Proses Bisnis Berjalan



Gambar 3.1 Diagram Alir Proses Rekrutmen Sebelum

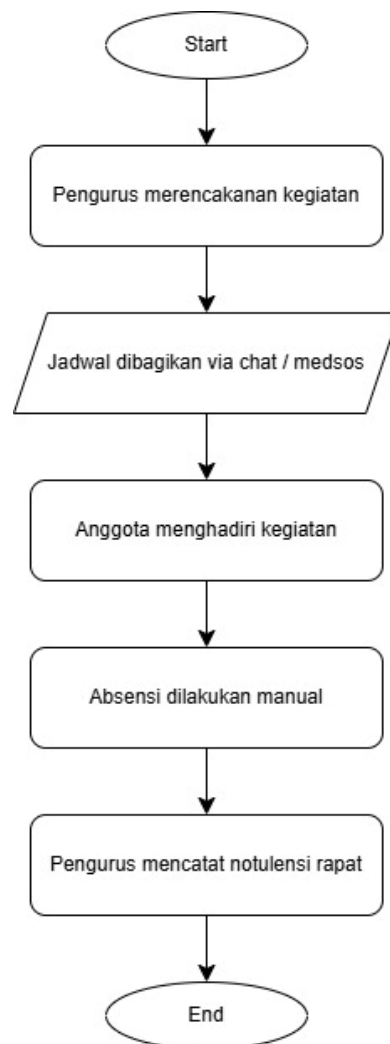


Gambar 3.2 Diagram Alir Bisnis Rekrutmen Sesudah

Flowchart proses rekrutmen sistem berjalan pada Gambar 3.1 menggambarkan proses penerimaan anggota organisasi yang masih dilakukan secara *manual*. Proses dimulai ketika pengurus mengumumkan open recruitment melalui media sosial atau grup percakapan. Informasi tersebut kemudian disebarikan kepada mahasiswa sehingga mahasiswa yang berminat dapat mengisi formulir pendaftaran. Setelah formulir dikumpulkan, pengurus melakukan review data pendaftaran secara *manual* untuk menentukan apakah pendaftar diterima atau tidak. Jika pendaftar tidak diterima, maka tidak terdapat konfirmasi lanjutan kepada mahasiswa. Namun apabila diterima, pengurus akan memberikan konfirmasi melalui chat pribadi atau grup percakapan. Selanjutnya pengurus mencatat data

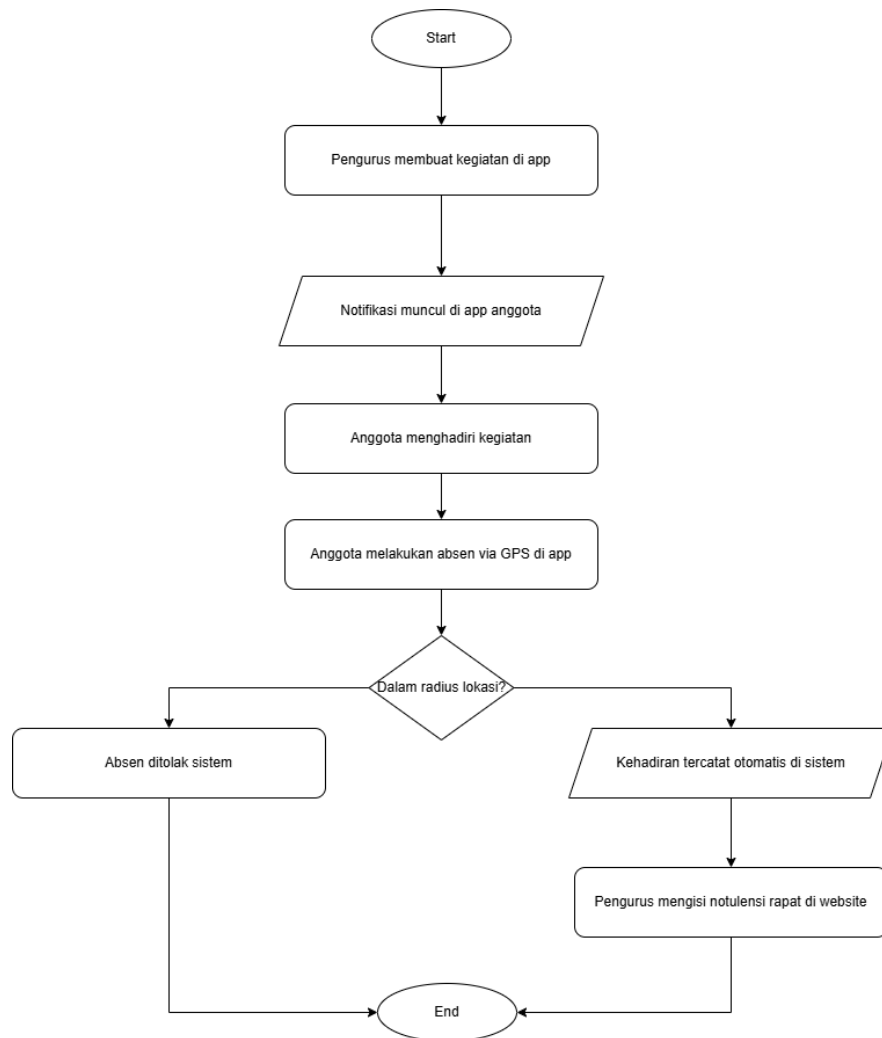
anggota baru secara *manual* sebelum mahasiswa resmi menjadi anggota organisasi. Pelaksanaan proses administrasi yang masih bersifat *manual* mengakibatkan pengelolaan menjadi kurang efektif dan berpotensi menyebabkan data tercecer maupun hilang.

Sementara itu, *flowchart* proses rekrutmen sistem usulan pada Gambar 3.2 menggambarkan proses penerimaan anggota yang telah terintegrasi dengan sistem berbasis *website* dan aplikasi. Proses dimulai ketika pengurus membuat periode rekrutmen pada *website* organisasi. Setelah periode rekrutmen dibuka, mahasiswa dapat mengakses profil organisasi dan mengisi formulir pendaftaran secara online melalui *website*. Data pendaftaran kemudian diterima dan ditinjau oleh pengurus melalui sistem. Apabila pendaftaran tidak diterima, sistem akan mengirimkan email penolakan kepada mahasiswa. Namun jika diterima, mahasiswa akan memperoleh email panggilan interview. Setelah tahap interview selesai dilakukan, pengurus menentukan hasil seleksi interview. Jika mahasiswa tidak lulus interview, sistem akan mengirimkan email penolakan. Sebaliknya, apabila mahasiswa dinyatakan lulus, akun anggota akan dibuat secara otomatis oleh sistem sehingga mahasiswa dapat *login* sebagai anggota organisasi. Seluruh data keanggotaan kemudian tersimpan secara otomatis di dalam sistem sehingga proses rekrutmen menjadi lebih terdokumentasi dan efisien.



Gambar 3.3 Diagram Alir Bisnis Kegiatan Sebelum

Flowchart pengelolaan kegiatan sistem berjalan pada Gambar 3.3 menggambarkan proses kegiatan organisasi yang masih dilakukan secara *manual*. Proses dimulai ketika pengurus merencanakan kegiatan organisasi dan membagikan jadwal kegiatan melalui media sosial atau grup percakapan. Setelah menerima informasi tersebut, anggota menghadiri kegiatan yang telah dijadwalkan. Pada saat kegiatan berlangsung, absensi dilakukan secara *manual* oleh pengurus atau anggota. Setelah kegiatan selesai, pengurus mencatat notulensi rapat secara *manual* sebagai dokumentasi kegiatan. Proses *manual* tersebut menyebabkan pencatatan kehadiran dan dokumentasi kegiatan menjadi kurang efektif serta berisiko mengalami kehilangan data atau kesalahan pencatatan.

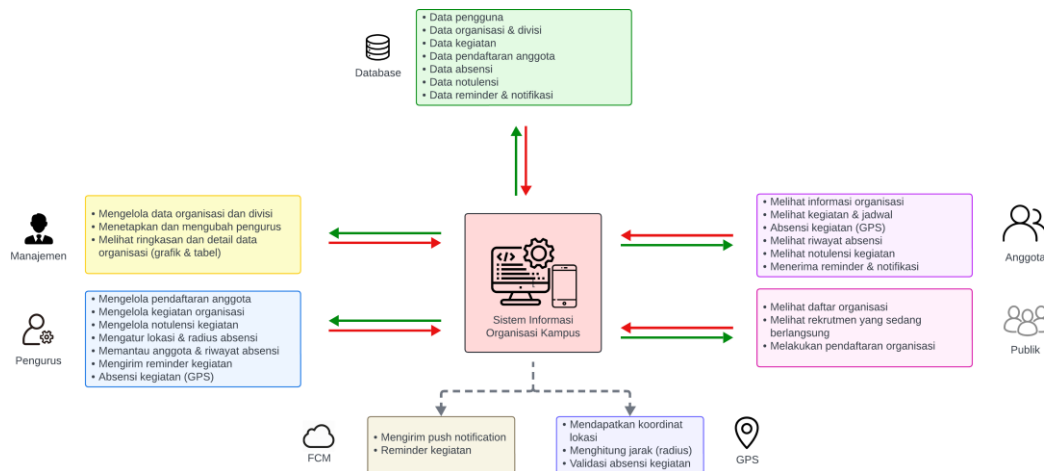


Gambar 3.4 Diagram Alir Bisnis Kegiatan Sesudah

Flowchart pengelolaan kegiatan sistem usulan pada Gambar 3.4 menggambarkan proses pengelolaan kegiatan yang telah terintegrasi dengan aplikasi dan *website*. Proses dimulai ketika pengurus membuat kegiatan pada aplikasi. Setelah kegiatan dibuat, sistem akan menampilkan notifikasi kegiatan pada aplikasi anggota sehingga informasi kegiatan dapat diterima secara langsung. Selanjutnya anggota menghadiri kegiatan dan melakukan absensi melalui fitur GPS pada aplikasi. Sistem kemudian akan memeriksa lokasi anggota saat ini berada. Jika lokasi anggota berada di luar radius, maka absensi akan ditolak oleh sistem. Namun apabila lokasi anggota berada dalam radius yang sesuai, kehadiran anggota akan tercatat secara otomatis di dalam sistem. Setelah data kehadiran tersimpan, pengurus dapat mengisi notulensi rapat melalui *website* sehingga seluruh

dokumentasi kegiatan tersimpan secara terpusat. Dengan adanya sistem MyOrga ini, proses manajemen kegiatan dapat menjadi lebih terintegrasi serta meminimalkan kesalahan pencatatan data.

3.1.2 Gambaran Umum Sistem



Gambar 3.5. Gambaran Umum Sistem MyOrga

Berdasarkan Gambar 3.5 di atas, terdapat 4 (empat) pengguna utama yang dapat menggunakan sistem informasi organisasi kampus, yaitu manajemen, pengurus, anggota, dan publik. Selain itu, sistem juga terintegrasi dengan *database*, Firebase Cloud Messaging (FCM), serta teknologi GPS dan *Haversine Formula* sebagai komponen pendukung sistem. Berikut adalah penjelasan mengenai peran dan interaksi masing-masing pengguna terhadap sistem.

1. Manajemen

Manajemen memiliki peran dalam mengelola data organisasi secara keseluruhan. Melalui sistem, manajemen dapat mengelola data organisasi dan divisi, menetapkan maupun mengubah pengurus organisasi, serta melihat ringkasan dan detail data organisasi dalam bentuk grafik dan tabel untuk membantu proses monitoring dan evaluasi organisasi.

2. Pengurus

Pengurus bertugas mengelola operasional organisasi melalui *dashboard* berbasis *web*. Pengurus dapat mengelola pendaftaran anggota, mengelola

kegiatan organisasi, mengelola notulensi kegiatan, mengatur lokasi dan radius absensi, memantau anggota dan riwayat absensi, serta mengirim *reminder* kegiatan kepada anggota organisasi. Selain itu, pengurus juga dapat melakukan absensi kegiatan melalui sistem.

3. Anggota

Anggota menggunakan aplikasi *mobile* Android untuk mengakses layanan organisasi kampus. Melalui aplikasi, anggota dapat melihat informasi organisasi, melihat jadwal kegiatan, melakukan pendaftaran organisasi, melakukan absensi kegiatan berbasis GPS, melihat riwayat absensi, melihat notulensi kegiatan, serta menerima *reminder* dan notifikasi kegiatan dari pengurus.

4. Publik

Publik merupakan pengguna umum yang dapat mengakses *landing page* sistem untuk melihat daftar organisasi dan informasi rekrutmen yang sedang berlangsung. Selain itu, publik juga dapat melakukan pendaftaran organisasi melalui sistem sebelum menjadi anggota resmi organisasi.

Selain melibatkan empat pengguna utama, sistem juga didukung oleh beberapa teknologi pendukung. FCM atau Firebase Cloud Messaging digunakan untuk mengirim notifikasi dan *reminder* kegiatan secara *real-time* kepada anggota organisasi. Sementara itu, GPS dan *Haversine Formula* digunakan dalam proses validasi absensi berbasis lokasi dengan menghitung jarak antara posisi pengguna dan lokasi kegiatan berdasarkan radius yang telah ditentukan.

3.2. Perancangan

3.2.1 Kebutuhan Fungsional

Tabel 3.1 berikut adalah kebutuhan fungsional dari aplikasi yang disajikan dalam bentuk tabel.

Tabel 3.1 Kebutuhan Fungsional

Kode	Kebutuhan Fungsional	Pengguna
FR-001	Dapat <i>login</i> ke sistem	Manajemen, Pengurus, Anggota

FR-002	Manajemen dapat mengelola data organisasi (tambah, ubah, hapus)	Manajemen, Pengurus, Anggota
FR-003	Manajemen dapat menetapkan dan mengubah Pengurus organisasi	Manajemen
FR-004	Manajemen dapat melihat ringkasan dan detail data organisasi dalam bentuk grafik dan tabel	Manajemen
FR-005	Pengurus dapat mengelola pendaftaran anggota (melihat daftar, melihat detail & CV, menerima/menolak)	Pengurus
FR-006	Pengurus dapat mengelola kegiatan organisasi (tambah, ubah, hapus)	Pengurus
FR-007	Pengurus dapat mengelola notulensi kegiatan	Pengurus
FR-008	Pengurus dapat mengatur lokasi dan radius absensi kegiatan	Pengurus
FR-009	Pengurus dapat memantau daftar anggota aktif dan riwayat absensi anggota	Pengurus
FR-010	Pengurus dapat menambahkan dan mengirim <i>reminder</i> kegiatan	Pengurus
FR-011	Pengurus dan Anggota dapat melakukan absensi kegiatan menggunakan GPS	Pengurus, Anggota
FR-012	Anggota dapat melihat riwayat absensi pribadi	Anggota
FR-013	Anggota dapat melihat notulensi kegiatan	Anggota
FR-014	Anggota dapat menerima <i>reminder</i> dari Pengurus	Anggota
FR-015	Sistem memvalidasi absensi berdasarkan perhitungan jarak menggunakan	Sistem

	Haversine Formula dan radius yang ditentukan	
FR-016	Sistem mengirimkan email hasil seleksi pendaftaran	Sistem
FR-017	Publik dapat melihat daftar organisasi pada <i>landing page</i>	Publik
FR-018	Publik dapat melakukan pendaftaran organisasi	Publik

3.2.2 Kebutuhan Non-Fungsional

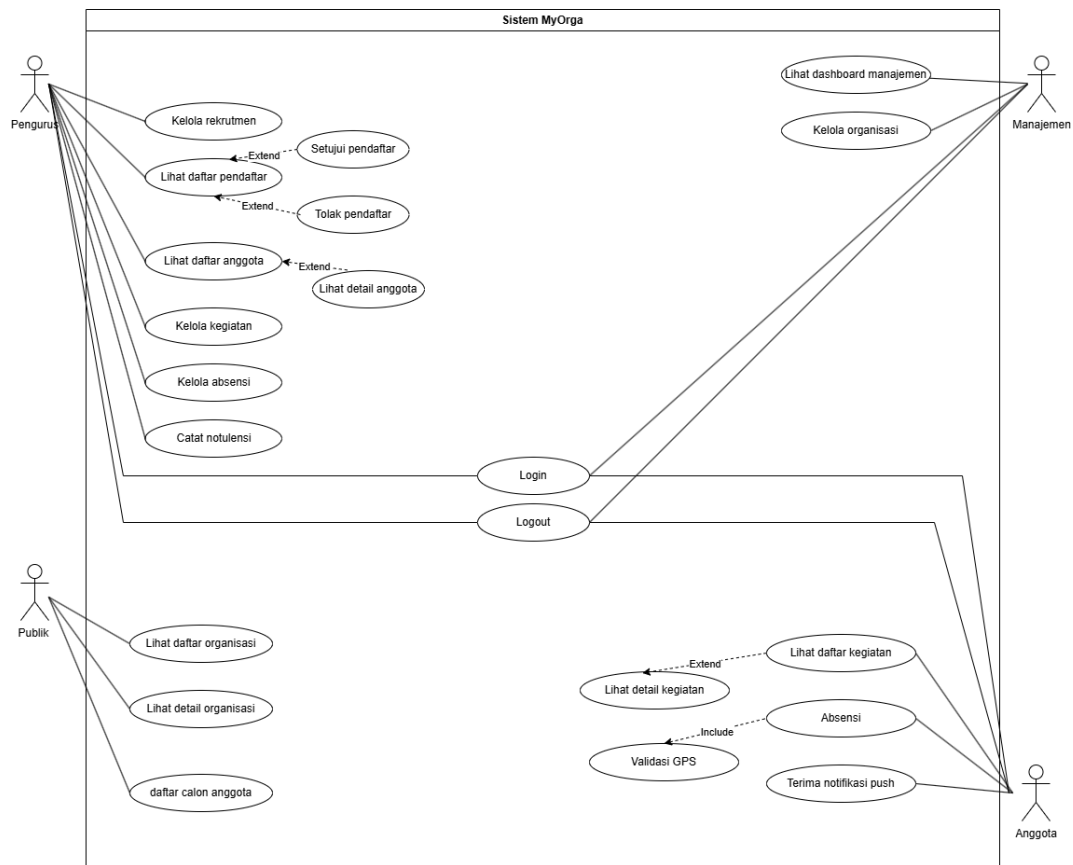
Tabel 3.2 berikut adalah kebutuhan fungsional dari aplikasi yang disajikan dalam bentuk tabel.

Tabel 3.2 Kebutuhan Non-Fungsional

Kode	Kebutuhan Non-Fungsional	Jenis
NFR-001	Sistem dapat menampilkan data organisasi dengan respon <5 detik pada <i>environment</i> lokal	<i>Performance</i>
NFR-002	Autentikasi <i>login</i> manajemen, pengurus, dan anggota harus berfungsi dengan validasi dasar (<i>email</i> dan <i>password</i>)	<i>Security</i>
NFR-003	Aplikasi menggunakan Bahasa Indonesia yang baik dan benar	<i>Language</i>

3.2.3 Diagram Use case

Berikut adalah diagram *use case* dari Sistem Manajemen Organisasi Kampus MyOrga.



Gambar 3.6 Diagram *Use case* Aplikasi

Gambar 3.6 di atas menjelaskan interaksi empat aktor utama, yaitu Publik, Anggota, Pengurus, dan Manajemen, dengan sistem aplikasi. Publik dapat melihat daftar organisasi, melihat detail organisasi, melihat rekrutmen aktif, serta melakukan pendaftaran sebagai calon anggota organisasi.

Anggota dapat melakukan *login* dan *logout* pada aplikasi *mobile*, melihat daftar dan detail kegiatan, melakukan absensi kegiatan, serta menerima notifikasi push.

Pengurus memiliki akses untuk melakukan *login* dan *logout*, mengelola periode rekrutmen, melihat daftar pendaftar, menyetujui atau menolak pendaftar, melihat daftar anggota dan detail anggota, mengelola kegiatan, mengelola absensi, serta mencatat notulensi kegiatan organisasi.

Sementara itu, Manajemen memiliki akses untuk melakukan *login* dan *logout*, melihat *dashboard* manajemen, serta mengelola data organisasi yang terdaftar pada sistem. Diagram ini menunjukkan bahwa sistem MyOrga dirancang

untuk memudahkan proses pengelolaan organisasi kemahasiswaan secara terintegrasi, mulai dari proses rekrutmen anggota hingga pengelolaan kegiatan dan administrasi organisasi.

3.2.4 Skenario *Use case*

1. Skenario *Use case Login Website*

Tabel 3.3 Skenario *Use case Login Website*

Kode <i>Use case</i>	UC-001
Nama <i>Use case</i>	<i>Login Website</i>
Aktor Utama	Manajemen dan Pengurus
Deskripsi	Pada skenario ini Manajemen dan Pengurus akan mencoba untuk <i>login</i> ke dalam aplikasi
Kondisi Awal	Aktor sedang berada di halaman <i>login</i>
Kondisi Akhir	Memperlihatkan halaman <i>dashboard</i> yang memiliki arti bahwa aktor berhasil <i>login</i> ke dalam aplikasi
Alur Kejadian Normal	
Aktor	Sistem
1. Aktor memasukkan email ke dalam <i>input field</i> yang disediakan	
	2. Sistem menampilkan email yang di- <i>input</i> ke layar
3. Admin memasukkan <i>password</i> ke dalam <i>input field</i> yang disediakan	
	4. Sistem menampilkan karakter dalam bentuk <i>masking</i>
5. Aktor menekan “Masuk”	
	6. Sistem memvalidasi kredensial yang dimasukkan.

	7. Sistem mengarahkan ke halaman <i>dashboard</i>
Alur Kejadian Alternatif	
Aktor	Sistem
[1a] Admin tidak memasukkan email	[1a] Sistem memberikan pesan " <i>The email field is required</i> "
[5a] Admin memasukkan kredensial yang salah	[5a] Sistem memberikan pesan " <i>Email atau password salah</i> "

2. Skenario *Use case Login Mobile*

Tabel 3.4 Skenario *Use case Login Mobile*

Kode <i>Use case</i>	UC-002
Nama <i>Use case</i>	<i>Login Mobile</i>
Aktor Utama	Anggota dan Pengurus
Deskripsi	Pada skenario ini, Aktor akan mencoba untuk masuk ke dalam aplikasi
Kondisi Awal	Aktor sedang di halaman <i>login</i>
Kondisi Akhir	Memuat halaman <i>homepage</i> yang menandakan aktor berhasil <i>login</i> ke dalam aplikasi
Alur Kejadian Normal	
Aktor	Sistem
1. Aktor memasukkan email ke dalam <i>input field</i> yang disediakan	
	2. Sistem menampilkan email yang di- <i>input</i> ke layar
3. Aktor memasukkan <i>password</i> ke dalam <i>input field</i> yang disediakan	

	4. Sistem menampilkan karakter dalam bentuk <i>masking</i>
5. Aktor menekan “Masuk”	
	6. Sistem memvalidasi kredensial yang dimasukkan.
	7. Sistem mengarahkan ke halaman <i>homepage</i>
Alur Kejadian Alternatif	
Aktor	Sistem
[1a] Aktor tidak memasukkan email	[1a] Sistem memberikan pesan “ <i>Email</i> dan <i>password</i> tidak boleh kosong”
[5a] Aktor memasukkan <i>username/password</i> yang salah	[5a] Sistem memberikan pesan “ <i>Email</i> atau <i>password</i> salah”

3. Skenario *Use case* Kelola Rekrutmen

Tabel 3.5 Skenario *Use case* Kelola Rekrutmen

Kode <i>Use case</i>	UC-003
Nama <i>Use case</i>	Kelola Rekrutmen
Aktor Utama	Pengurus
Deskripsi	Pada skenario ini, Aktor Pengurus akan mencoba untuk mengelola periode rekrutmen
Kondisi Awal	Pengurus telah masuk ke akunnya dan sedang di halaman Periode Rekrutmen
Kondisi Akhir	Periode rekrutmen berhasil dikelola oleh Pengurus
Alur Kejadian Normal	
Aktor	Sistem
	1. Sistem menyajikan daftar rekrutmen yang pernah dibuat
2. Pengurus memilih aksi "Tambah Periode" atau "Edit" periode rekrutmen	
	3. Sistem menampilkan form kelola rekrutmen (judul rekrutmen, deskripsi, tanggal mulai dan berakhir, status pendaftaran, dan poster pendaftaran)
4. Pengurus mengisi/memperbarui input <i>field</i> sesuai kebutuhan	
	5. Sistem menerima input
6. Pengurus menekan tombol "Simpan	
	7. Sistem memvalidasi input

	8. Sistem memperbarui halaman kelola rekrutmen
Alur Kejadian Alternatif	
Aktor	Sistem
[2a] Pengurus memilih aksi "Hapus" periode rekrutmen	[1a] Sistem memperlihatkan dialog konfirmasi penghapusan. Jika pengurus setuju, sistem akan menghapus data periode rekrutmen dan memperbarui daftar rekrutmen
[4a] Pengurus tidak mengisi <i>field</i> wajib (judul rekrutmen, deskripsi, tanggal mulai dan berakhir, serta status pendaftaran)	[4a] Sistem memberikan pesan "Harap isi bidang ini"

4. Skenario *Use case* Lihat Daftar Pendaftar

Tabel 3.6 Skenario *Use case* Lihat Daftar Pendaftar

Kode <i>Use case</i>	UC-004
Nama <i>Use case</i>	Lihat Daftar Pendaftar
Aktor Utama	Pengurus
Deskripsi	Pada skenario ini, Aktor Pengurus <i>me-monitoring</i> daftar pendaftar yang mendaftar pada rekrutmen yang sedang dibuka
Kondisi Awal	Pengurus saat ini sedang di halaman <i>dashboard</i> pengurus
Kondisi Akhir	Sistem berhasil menyajikan daftar pendaftar dalam bentuk tabel
Alur Kejadian Normal	
Aktor	Sistem
1. Admin menavigasi ke "Data Pendaftar"	

	2. Sistem memperlihatkan daftar pendaftar yang tersimpan dalam bentuk tabel
Alur Kejadian Alternatif	
Aktor	Sistem
	[2a] Jika tabel data pendaftar kosong, Sistem akan memberikan pesan "Tidak ada pendaftar yang sesuai dengan filter"

5. Skenario *Use case* Lihat Daftar Anggota

Tabel 3.7 Skenario *Use case* Lihat Daftar Anggota

Kode <i>Use case</i>	UC-005
Nama <i>Use case</i>	Lihat Daftar Anggota
Aktor Utama	Pengurus
Deskripsi	Pada skenario ini, Aktor Pengurus <i>me-monitoring</i> daftar anggota organisasinya
Kondisi Awal	Pengurus saat ini sedang di halaman <i>dashboard</i> pengurus
Kondisi Akhir	Daftar anggota yang disajikan dalam bentuk tabel berhasil dilihat oleh Pengurus
Alur Kejadian Normal	
Aktor	Sistem
1. Admin menavigasi ke "Anggota"	
	2. Sistem memperlihatkan daftar anggota yang tersimpan dalam bentuk tabel
Alur Kejadian Alternatif	

Aktor	Sistem
	[2a] Semisal anggota kosong, Sistem memberikan pesan "Tidak ada anggota yang ditemukan"

6. Skenario *Use case* Kelola Kegiatan

Tabel 3.8 Skenario *Use case* Kelola Kegiatan

Kode <i>Use case</i>	UC-006
Nama <i>Use case</i>	Kelola Kegiatan
Aktor Utama	Pengurus
Deskripsi	Pada skenario ini, Pengurus akan mencoba mengelola kegiatan organisasi
Kondisi Awal	Pengurus sekarang sedang berada di halaman Manajemen Kegiatan
Kondisi Akhir	Pengurus berhasil mengelola kegiatan organisasi
Alur Kejadian Normal	
Aktor	Sistem
	1. Sistem menyajikan daftar kegiatan yang pernah dibuat
2. Pengurus memilih aksi "Tambah Kegiatan" atau "Edit" kegiatan	
	3. Sistem menampilkan form kegiatan (agenda kegiatan, tanggal, waktu, nama lokasi, dan tandai di peta)
4. Pengurus mengisi input <i>field</i> sesuai kebutuhan	
	5. Sistem menerima input

6. Pengurus menekan tombol “Simpan”	
	7. Sistem memvalidasi input
	8. Sistem memperbarui daftar kegiatan
Alur Kejadian Alternatif	
Aktor	Sistem
[2a] Pengurus memilih aksi “Hapus” kegiatan	[2a] Sistem menampilkan konfirmasi hapus. Jika pengurus setuju, sistem menghapus data kegiatan dan memperbarui daftar.
[4a] Pengurus tidak mengisi <i>field</i> wajib (agenda kegiatan, tanggal, waktu, nama lokasi)	[4a] Sistem menampilkan pesan ”Harap isi bidang ini”

7. Skenario *Use case* Kelola Absensi

Tabel 3.9 Skenario *Use case* Kelola Absensi

Kode <i>Use case</i>	UC-007
Nama <i>Use case</i>	Kelola Absensi
Aktor Utama	Pengurus
Deskripsi	Pada skenario ini, Pengurus mencoba mengelola absensi dari anggota
Kondisi Awal	Pengurus saat ini berada di halaman ”Manajemen Kegiatan”
Kondisi Akhir	Pengurus berhasil mengelola absensi salah satu kegiatan
Alur Kejadian Normal	
Aktor	Sistem
1. Pengurus menekan ”Detail” pada salah satu kegiatan	

	2. Sistem mengeluarkan slide out form
3. Pengurus memilih "Daftar Hadir"	
	4. Sistem menampilkan daftar absensi anggota
5. Pengurus <i>manual</i> mengubah kehadiran anggota	
6. Pengurus menekan tombol "Simpan Daftar Hadir"	
	7. Perubahan tersimpan
Alur Kejadian Alternatif	
Aktor	Sistem
[6a] Pengurus menekan tombol "Batal"	[6a] Perubahan yang dilakukan oleh pengurus pada absensi tidak tersimpan

8. Skenario *Use case* Catat Notulensi

Tabel 3.10 Skenario *Use case* Catat Notulensi

Kode <i>Use case</i>	UC-008
Nama <i>Use case</i>	Catat Notulensi
Aktor Utama	Pengurus
Deskripsi	Pada skenario ini, Aktor Pengurus menambahkan catatan pada salah satu kegiatan yang ada
Kondisi Awal	Pengurus sekarang sedang berada di halaman "Manajemen Kegiatan"
Kondisi Akhir	Pengurus berhasil menambahkan notulensi
Alur Kejadian Normal	
Aktor	Sistem

	1. Sistem menampilkan tabel berisi daftar kegiatan yang pernah ditambahkan
2. Pengurus menekan "Detail"	
	3. Sistem mengeluarkan slide out form
4. Pengurus mengetikkan notulensi pada tempat yang disediakan	
	5. Sistem menyajikan inputan catatan ke layar
6. Pengurus menekan "Simpan Notulensi"	
	7. Sistem menyimpan notulensi yang diketikkan pengurus
Alur Kejadian Alternatif	
Aktor	Sistem
[6a] Pengurus menekan tombol "Batal"	[6a] Sistem tidak menyimpan notulensi

9. Skenario *Use case* Lihat *Dashboard* Manajemen

Tabel 3.11 Skenario *Use case* Lihat *Dashboard* Manajemen

Kode <i>Use case</i>	UC-009
Nama <i>Use case</i>	Lihat <i>Dashboard</i> Manajemen
Aktor Utama	Manajemen
Deskripsi	Pada skenario ini, manajemen akan mencoba untuk masuk dan melihat <i>dashboard</i> manajemen
Kondisi Awal	Manajemen sedang berada di halaman <i>login</i>

Kondisi Akhir	Manajemen berhasil melihat <i>dashboard</i> manajemen
Alur Kejadian Normal	
Aktor	Sistem
1. Manajemen mengisi kredensial	
	2. Sistem memvalidasi kredensial
	3. Sistem memperlihatkan halaman <i>dashboard</i> manajemen
Alur Kejadian Alternatif	
Aktor	Sistem
[2a] Manajemen memasukkan kredensial yang salah	[2a] Sistem memberikan pesan error "Email atau <i>password</i> salah"

10. Skenario *Use case* Kelola Organisasi

Tabel 3.12 Skenario *Use case* Kelola Organisasi

Kode <i>Use case</i>	UC-010
Nama <i>Use case</i>	Kelola Organisasi
Aktor Utama	Manajemen
Deskripsi	Pada skenario ini, Manajemen akan mencoba untuk mengelola organisasi
Kondisi Awal	Manajemen sedang berada di halaman Profil Organisasi
Kondisi Akhir	Manajemen berhasil mengelola organisasi
Alur Kejadian Normal	
Aktor	Sistem
	1. Sistem menyajikan daftar organisasi yang pernah ditambahkan

2. Manajemen memilih aksi ”Tambah Organisasi” atau ”Edit” organisasi	
	3. Sistem menampilkan form organisasi (nama organisasi, deskripsi organisasi, logo organisasi, galeri foto, struktur divisi, dan pengurus inti)
4. Manajemen mengisi <i>field</i> sesuai kebutuhan	
	5. Sistem menerima input
6. Manajemen menekan tombol ”Simpan	
	7. Sistem memperbarui daftar organisasi
Alur Kejadian Alternatif	
Aktor	Sistem
[2a] Pengurus memilih aksi “Hapus” organisasi	[2a] Memberikan pesan konfirmasi penghapusan. Jika pengurus setuju, sistem akan membuang data organisasi tersebut dan memperbarui daftar
[4a] Pengurus tidak mengisi <i>field</i> wajib	[4a] Sistem memberikan pesan ”Harap isi bidang ini”

11. Skenario *Use case* Lihat Daftar Organisasi

Tabel 3.13 Skenario *Use case* Lihat Daftar Organisasi

Kode <i>Use case</i>	UC-011
Nama <i>Use case</i>	Lihat Daftar Organisasi
Aktor Utama	Publik

Deskripsi	Pada skenario ini, Publik akan mencoba untuk melihat daftar organisasi yang ada
Kondisi Awal	Publik membuka <i>website</i> MyOrga
Kondisi Akhir	Publik berhasil melihat daftar organisasi yang ada
Alur Kejadian Normal	
Aktor	Sistem
	1. Sistem menampilkan <i>landing page</i> aplikasi MyOrga
2. Publik menekan tombol "Eksplor Organisasi"	
	3. Sistem menampilkan card organisasi yang menampilkan nama, logo, dan deskripsi organisasi
Alur Kejadian Alternatif	
Aktor	Sistem
[3a] Publik melakukan refresh halaman	[3a] Sistem menampilkan halaman daftar organisasi tanpa data organisasi

12. Skenario *Use case* Lihat Detail Organisasi

Tabel 3.14 Skenario *Use case* Lihat Detail Organisasi

Kode <i>Use case</i>	UC-012
Nama <i>Use case</i>	Lihat Detail Organisasi
Aktor Utama	Publik
Deskripsi	Pada skenario ini, Publik akan mencoba untuk melihat detail salah satu organisasi

Kondisi Awal	Publik berada di <i>landing page</i> aplikasi MyOrga
Kondisi Akhir	Publik berhasil melihat detail salah satu organisasi
Alur Kejadian Normal	
Aktor	Sistem
	1. Sistem menampilkan <i>landing page</i> aplikasi MyOrga
2. Publik menekan tombol "Eksplor Organisasi"	
	3. Sistem menampilkan card organisasi yang menampilkan nama, logo, dan deskripsi organisasi
4. Publik menekan salah satu card organisasi	
	5. Sistem menyajikan halaman detail organisasi
Alur Kejadian Alternatif	
Aktor	Sistem
[5a] Publik melakukan refresh halaman	[5a] Sistem gagal memuat detail organisasi

13. Skenario *Use case* Daftar Calon Anggota

Tabel 3.15 Skenario *Use case* Daftar Calon Anggota

Kode <i>Use case</i>	UC-013
Nama <i>Use case</i>	Daftar Calon Anggota
Aktor Utama	Publik

Deskripsi	Pada skenario ini, Publik akan mencoba daftar di salah satu organisasi
Kondisi Awal	Publik sedang berada di halaman detail salah satu organisasi
Kondisi Akhir	Publik berhasil mendaftar
Alur Kejadian Normal	
Aktor	Sistem
	1. Sistem memuat "Detail Organisasi"
2. Publik menekan tombol "Daftar Sekarang" pada halaman detail organisasi	
	3. Sistem menyajikan form pendaftaran
4. Publik mengisi semua input <i>field</i> yang ada	
	5. Sistem menerima inputan
6. Publik menekan "Kirim"	
	7. Sistem merekam pendaftaran ke basis data
Alur Kejadian Alternatif	
Aktor	Sistem
[4a] Aktor tidak mengisi semua input <i>field</i>	[4a] Sistem tidak memperbolehkan Publik untuk lanjut ke tahap selanjutnya

14. Skenario *Use case* Lihat Daftar Kegiatan

Tabel 3.16 Skenario *Use case* Lihat Daftar Kegiatan

Kode <i>Use case</i>	UC-014
----------------------	--------

Nama <i>Use case</i>	Lihat Daftar Kegiatan
Aktor Utama	Anggota
Deskripsi	Pada skenario ini, Anggota akan melihat daftar kegiatan yang ditambahkan oleh Pengurus
Kondisi Awal	Anggota sedang di halaman <i>homepage</i>
Kondisi Akhir	Daftar kegiatan berhasil disajikan sistem
Alur Kejadian Normal	
Aktor	Sistem
1. Admin menavigasi ke "Kegiatan"	
	2. Sistem menyajikan halaman kegiatan yang memuat daftar kegiatan
Alur Kejadian Alternatif	
Aktor	Sistem
	[2a] Jika tidak ada kegiatan apapun, sistem akan menampilkan pesan "Tidak ada kegiatan"

15. Skenario *Use case* Absensi

Tabel 3.17 Skenario *Use case* Absensi

Kode <i>Use case</i>	UC-015
Nama <i>Use case</i>	Absensi
Aktor Utama	Anggota
Deskripsi	Pada skenario ini, Anggota akan mencoba melakukan absen di salah satu kegiatan

Kondisi Awal	Anggota sedang berada di halaman Absensi
Kondisi Akhir	Anggota berhasil melakukan absensi
Alur Kejadian Normal	
Aktor	Sistem
	1. Sistem memuat "Absensi"
2. Anggota mengkil tombol "Absen Sekarang"	
	3. Sistem menampilkan dialog sukses jika radius tidak melebihi batasan yang ditetapkan
Alur Kejadian Alternatif	
Aktor	Sistem
	[3a] Jika radius melebihi batasan, sistem akan menolak absen Anggota tersebut

16. Skenario *Use case* Terima Notifikasi Push

Tabel 3.18 Skenario *Use case* Terima Notifikasi Push

Kode <i>Use case</i>	UC-016
Nama <i>Use case</i>	Terima Notifikasi Push
Aktor Utama	Anggota
Deskripsi	Pada skenario ini, Anggota akan mencoba untuk menerima notifikasi push berisi <i>reminder</i> kegiatan dari aplikasi
Kondisi Awal	Anggota sedang di luar aplikasi
Kondisi Akhir	Anggota berhasil menerima notifikasi
Alur Kejadian Normal	

Aktor	Sistem
	1. Sistem mengirim notifikasi kepada Anggota
	2. Sistem menampilkan notifikasi pada perangkat <i>mobile</i> anggota
Alur Kejadian Alternatif	
Aktor	Sistem
	[2a] Jika koneksi perangkat tidak tersedia, sistem menunda pengiriman notifikasi hingga koneksi tersedia

17. Skenario *Use case Logout*

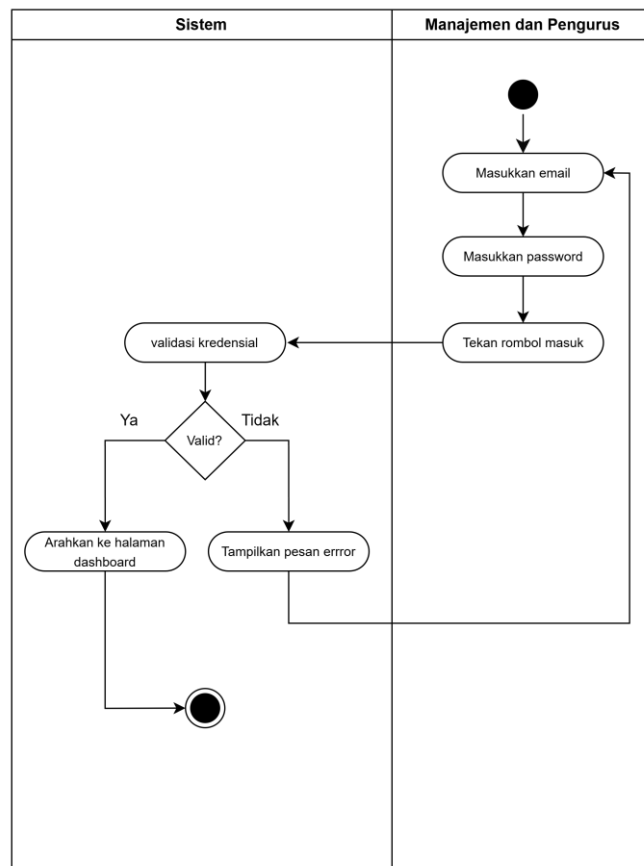
Tabel 3.19 Skenario *Use case Logout*

Kode <i>Use case</i>	UC-017
Nama <i>Use case</i>	<i>Logout</i>
Aktor Utama	Manajemen, Pengurus, dan Anggota
Deskripsi	Pada skenario ini, Aktor akan mencoba <i>logout</i> dari sistem
Kondisi Awal	Aktor sedang mengakses aplikasi
Kondisi Akhir	Aktor berhasil keluar dari aplikasi
Alur Kejadian Normal	
Aktor	Sistem
1. Aktor mengklik "Keluar"	
	2. Sistem menyajikan dialog konfirmasi <i>logout</i>
3. Aktor memilih "Ya"	
	4. Sistem menghapus sesi dan mengarahkan Aktor kembali ke halaman <i>login</i>

Alur Kejadian Alternatif	
Aktor	Sistem
[3a] Aktor memilih tombol “Batal”	[3a] Sistem membatalkan proses <i>logout</i> dan menampilkan halaman sebelumnya

3.2.5 Diagram Aktivitas

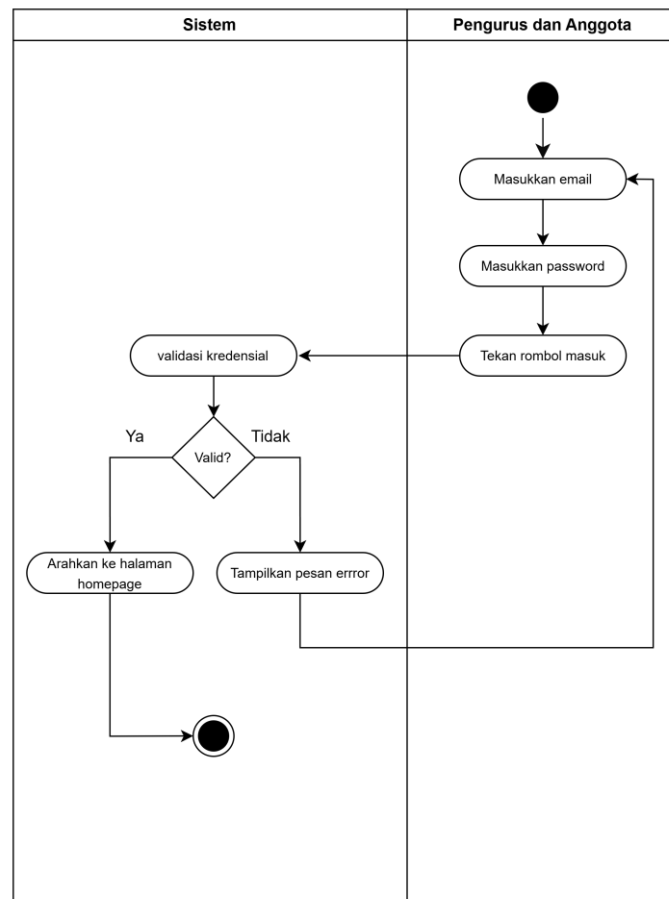
1. Diagram Aktivitas *Login Website*



Gambar 3.7 Diagram Aktivitas *Login Website*

Diagram aktivitas *login website* pada Gambar 3.7 menggambarkan alur proses autentikasi pengguna pada *platform web* MyOrga. Proses dimulai ketika pengguna mengakses halaman login dan memasukkan alamat email serta kata sandi. Selanjutnya, sistem memverifikasi kredensial. Jika kredensial valid, sistem akan membuat sesi pengguna dan mengarahkan pengguna ke halaman *dashboard* sesuai dengan perannya, yaitu sebagai pengurus atau manajemen.

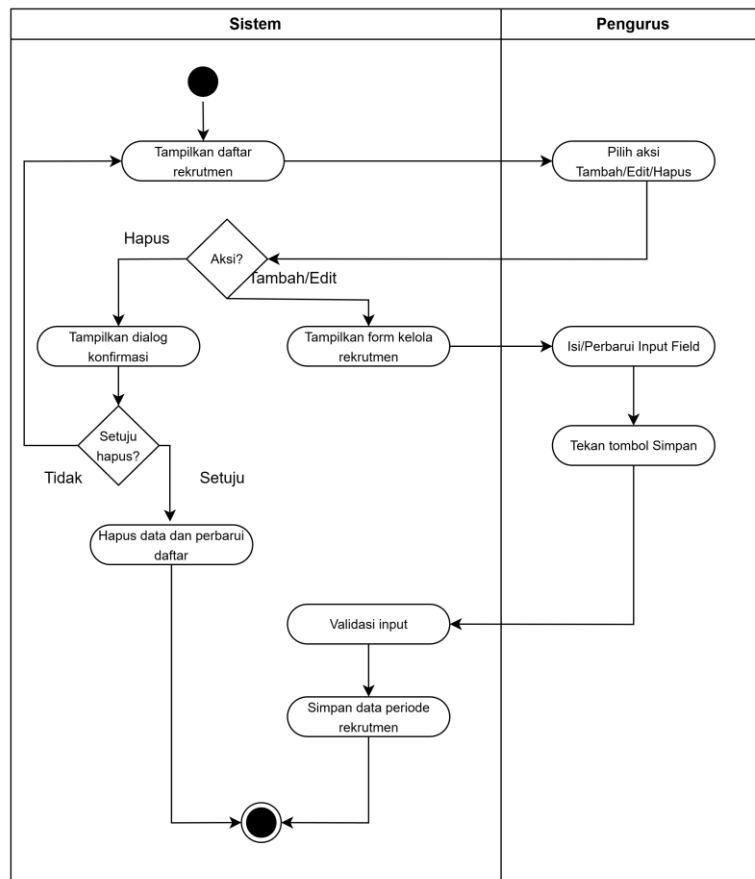
2. Diagram Aktivitas *Login Mobile*



Gambar 3.8 Diagram Aktivitas *Login Mobile*

Diagram aktivitas *login mobile* pada Gambar 3.8 menggambarkan alur proses autentikasi pengguna pada aplikasi *mobile* MyOrga. Pengguna diminta memasukkan kredensial pada form *login*. Sistem mengirimkan permintaan autentikasi ke server melalui endpoint API dan server memvalidasi kredensial menggunakan Laravel Sanctum. Apabila kredensial tidak valid, aplikasi menampilkan pesan kesalahan. Apabila valid, server mengembalikan Bearer token yang disimpan di penyimpanan lokal perangkat menggunakan SharedPreferences, kemudian pengguna diarahkan ke halaman utama sesuai *role* yang dimiliki, yaitu pengurus atau anggota.

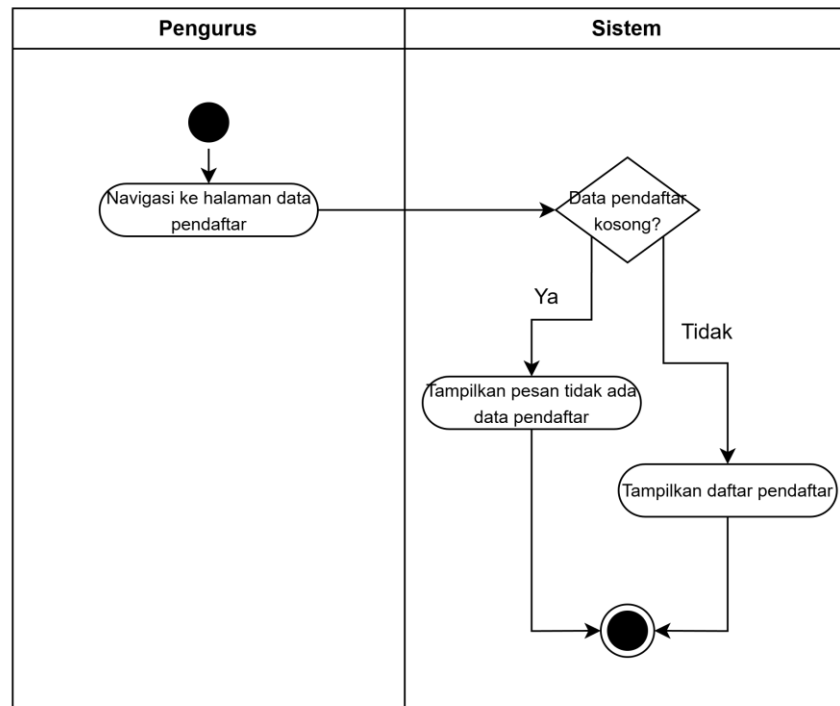
3. Diagram Aktivitas Kelola Rekrutmen



Gambar 3.9 Diagram Aktivitas Kelola Rekrutmen

Diagram aktivitas kelola rekrutmen pada Gambar 3.9 menggambarkan tahapan proses manajemen periode rekrutmen yang dilakukan oleh pengurus organisasi. Proses dimulai ketika pengurus mengakses halaman periode rekrutmen dan memilih aksi yang ingin dilakukan, yaitu menambah, mengedit, atau menghapus periode rekrutmen. Pada proses penambahan, pengurus mengisi judul, deskripsi, tanggal mulai, tanggal akhir, dan status periode rekrutmen. Sistem kemudian menyimpan data ke basis data dan menampilkan konfirmasi keberhasilan. Pada proses penghapusan, sistem menampilkan dialog konfirmasi sebelum menghapus data dari basis data.

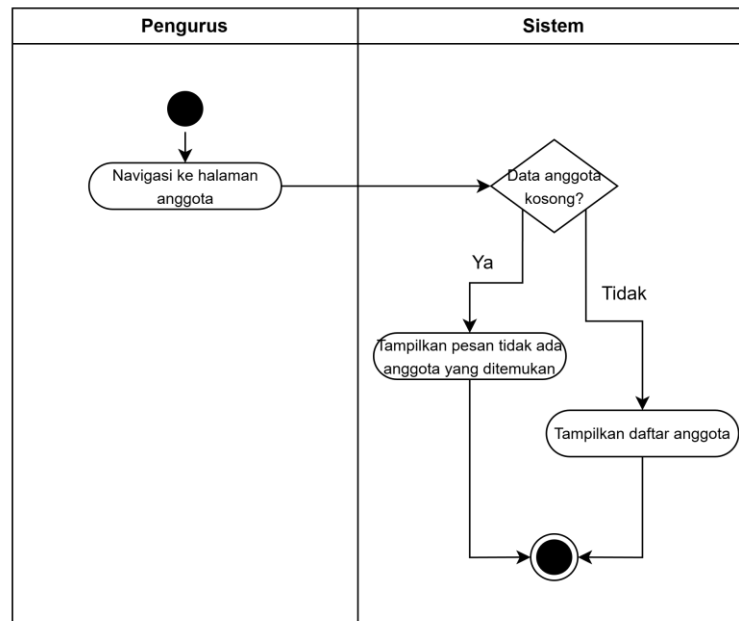
4. Diagram Aktivitas Lihat Daftar Pendaftar



Gambar 3.10 Diagram Aktivitas Lihat Daftar Pendaftar

Diagram aktivitas lihat daftar pendaftar pada Gambar 3.10 menggambarkan alur pengurus mengelolapendaftaran yang masuk. Alur dimulai saat pengurus membuka halaman data pendaftar dan sistem memuat seluruh aplikasi yang tersedia. Pengurus dapat memfilter tampilan berdasarkan periode rekrutmen dan status aplikasi. Pengurus kemudian dapat membuka detail aplikasi untuk melihat informasi lengkap pendaftar seperti motivasi, kontribusi, dan tautan CV. Setelah melakukan penilaian, pengurus dapat mengambil keputusan untuk menerima pendaftar dengan memilih divisi yang akan di-assign, atau menolak pendaftar, dan sistem akan memperbarui status aplikasi serta mengirimkan notifikasi kepada pendaftar yang bersangkutan.

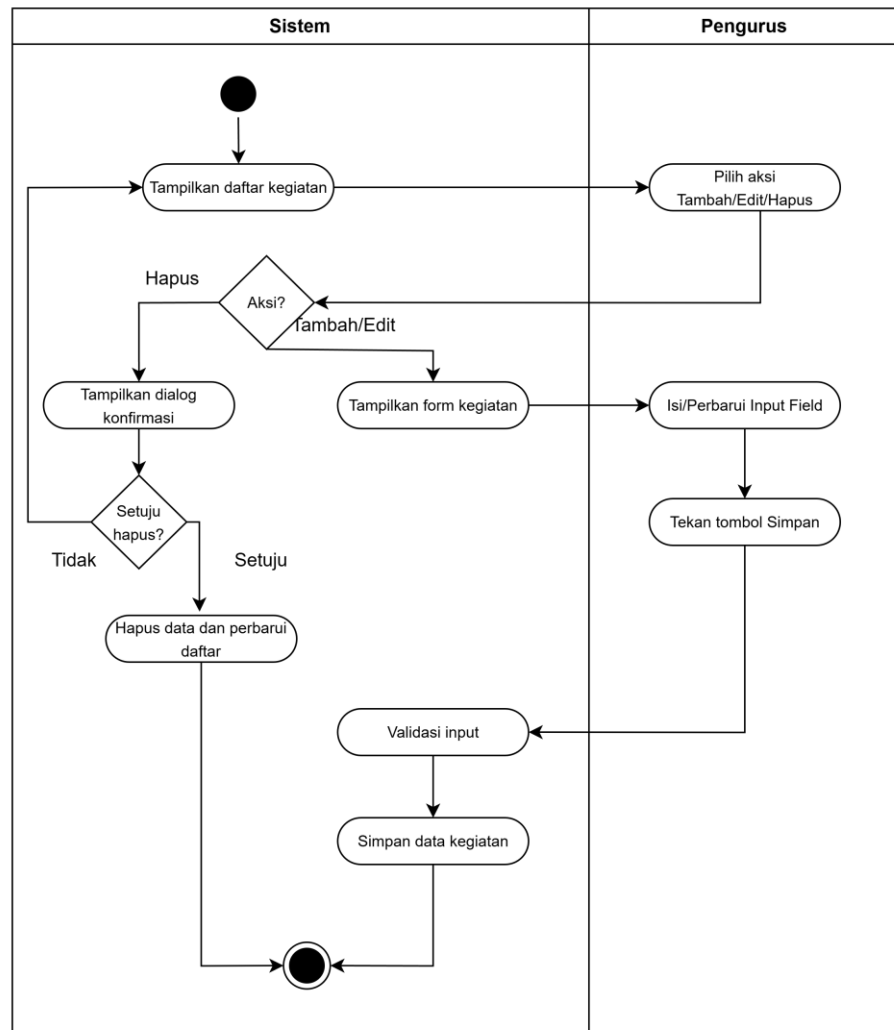
5. Diagram Aktivitas Lihat Daftar Anggota



Gambar 3.11 Diagram Aktivitas Lihat Daftar Anggota

Diagram aktivitas lihat daftar anggota menggambarkan alur proses pengurus dalam melihat data keanggotaan organisasi. Proses dimulai ketika pengurus berada di halaman anggota dan sistem menampilkan seluruh anggota aktif beserta informasi nama, NIM, divisi, dan jabatan. Pengurus dapat melakukan pencarian berdasarkan nama, NIM, atau divisi untuk mempermudah pencarian data anggota tertentu. Pengurus kemudian dapat membuka detail profil anggota untuk melihat informasi lengkap termasuk statistik kehadiran dan riwayat absensi anggota pada setiap kegiatan yang telah diselenggarakan.

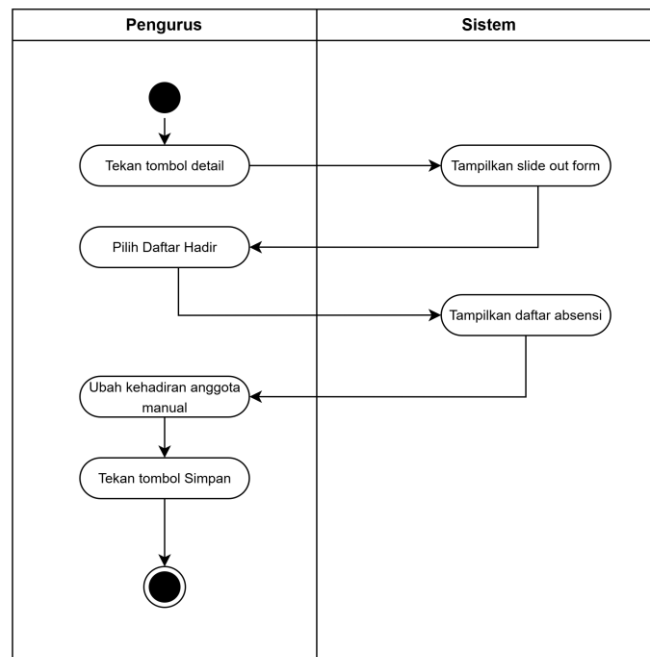
6. Diagram Aktivitas Kelola Kegiatan



Gambar 3.12 Diagram Aktivitas Kelola Kegiatan

Diagram aktivitas kelola kegiatan dimulai ketika pengurus mengakses halaman kegiatan dan memilih aksi yang ingin dilakukan, yaitu menambah, mengedit, atau menghapus kegiatan. Pada proses penambahan, pengurus mengisi informasi kegiatan meliputi judul, deskripsi, tanggal, waktu, lokasi, koordinat GPS berupa latitude dan longitude, serta radius kehadiran dalam satuan meter. Setelah data disimpan, sistem secara otomatis membuat record absensi untuk seluruh anggota aktif organisasi yang terhubung dengan kegiatan tersebut.

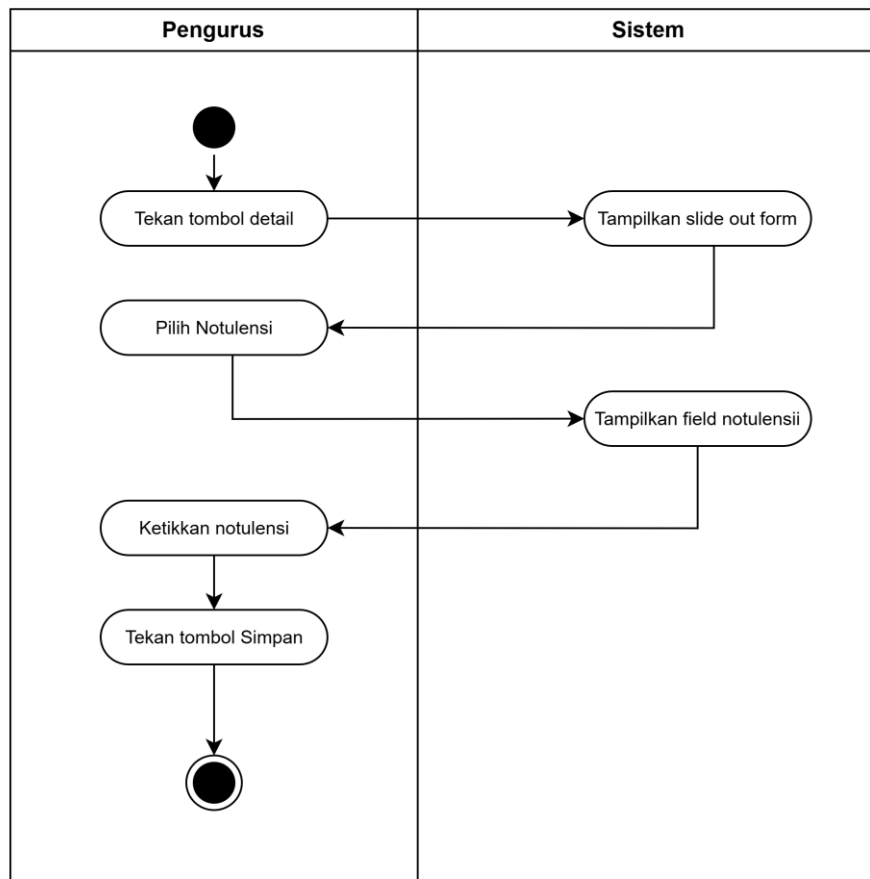
7. Diagram Aktivitas Kelola Absensi



Gambar 3.13 Diagram Aktivitas Kelola Absensi

Diagram aktivitas kelola absensi menggambarkan alur proses pengurus dalam mengelola data kehadiran anggota pada suatu kegiatan secara *manual*. Proses dimulai ketika pengurus membuka detail kegiatan dan mengakses tab daftar hadir. Sistem menampilkan seluruh anggota beserta status kehadiran masing-masing. Pengurus dapat mengubah status kehadiran setiap anggota melalui dropdown dengan pilihan Hadir, Izin, Sakit, atau Alpa. Sistem kemudian menyimpan perubahan status kehadiran ke basis data dengan menandai bahwa pembaruan dilakukan secara *manual* oleh pengurus.

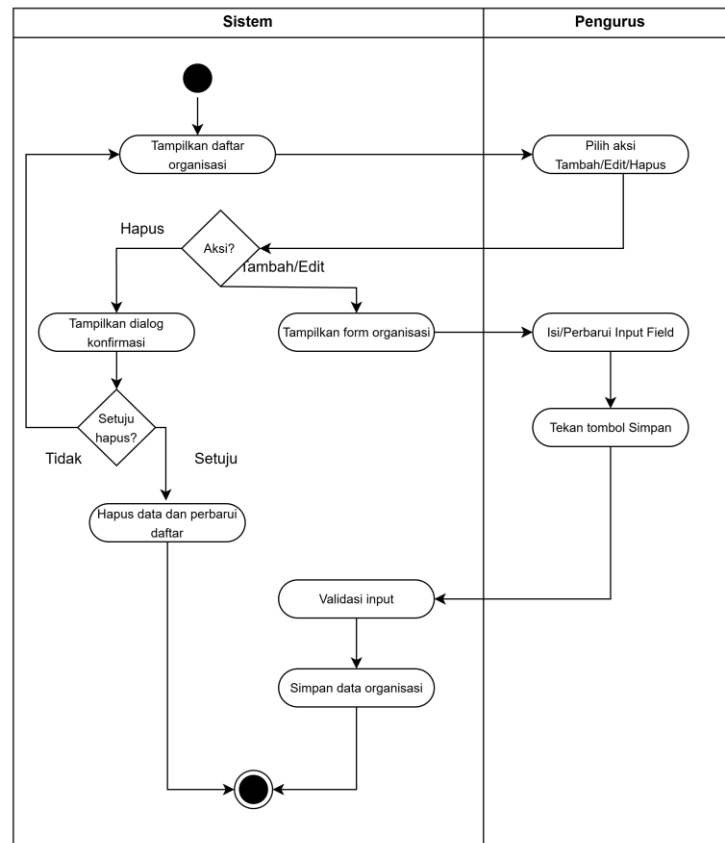
8. Diagram Aktivitas Catat Notulensi



Gambar 3.14 Diagram Aktivitas Catat Notulensi

Diagram aktivitas catat notulensi menggambarkan alur proses pengurus dalam mencatat hasil rapat atau kegiatan yang telah diselenggarakan. Proses dimulai ketika pengurus membuka detail kegiatan dan mengakses tab notulensi. Sistem menampilkan area teks beserta notulensi yang sudah ada apabila sebelumnya telah dicatat. Pengurus mengetikkan catatan hasil rapat mencakup poin-poin penting, hasil diskusi, dan keputusan yang diambil. Setelah selesai kemudian pengurus menyimpan notulensi tersebut.

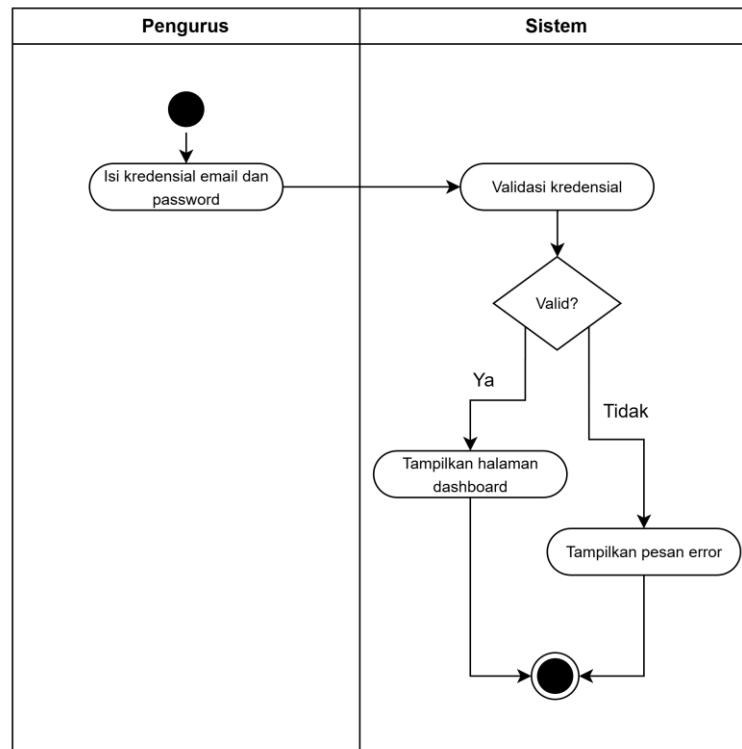
9. Diagram Aktivitas Kelola Organisasi



Gambar 3.15 Diagram Aktivitas Kelola Organisasi

Diagram aktivitas kelola organisasi menggambarkan tahapan proses manajemen dalam mengelola data organisasi mahasiswa pada sistem. Tahapan dimulai ketika manajemen mengakses halaman profil organisasi dan memilih aksi yang ingin dilakukan, yaitu menambah, mengedit, atau menghapus organisasi. Pada proses penambahan dan pengeditan, manajemen mengisi informasi organisasi meliputi nama, deskripsi, logo, galeri foto, dan struktur divisi. Khusus pada mode edit, manajemen juga dapat menentukan susunan pengurus inti dengan memilih pengguna yang terdaftar melalui fitur pencarian. Sistem kemudian menyimpan seluruh perubahan ke basis data dan menampilkan konfirmasi keberhasilan kepada pengguna.

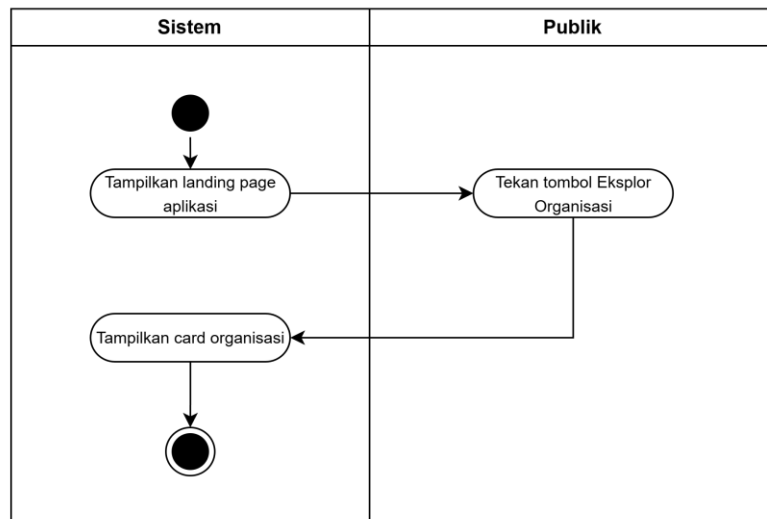
10. Diagram Aktivitas Lihat *Dashboard* Manajemen



Gambar 3.16 Diagram Aktivitas Lihat *Dashboard* Manajemen

Diagram aktivitas lihat *dashboard* manajemen menggambarkan alur proses manajemen dalam mengakses ringkasan informasi sistem secara keseluruhan. Proses dimulai ketika manajemen berhasil *login* dan sistem secara otomatis menampilkan halaman *dashboard*. Sistem mengambil data statistik dari basis data meliputi total organisasi yang terdaftar, total anggota aktif di seluruh organisasi, dan data pertumbuhan jumlah anggota dalam enam bulan terakhir. Data tersebut kemudian ditampilkan dalam bentuk kartu statistik dan grafik area untuk memudahkan manajemen dalam memantau perkembangan organisasi mahasiswa secara keseluruhan.

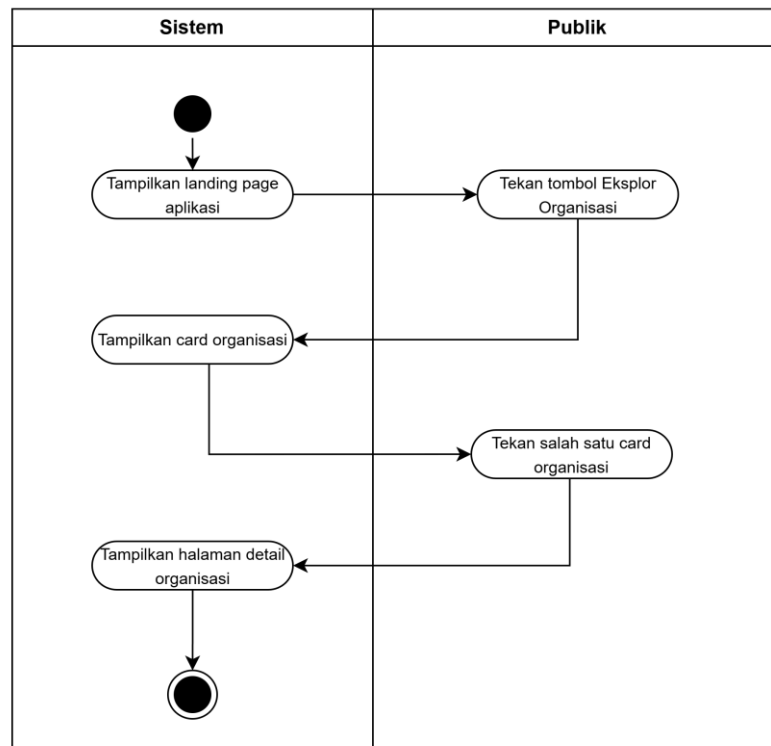
11. Diagram Aktivitas Lihat Daftar Organisasi



Gambar 3.17 Diagram Aktivitas Lihat Daftar Organisasi

Diagram aktivitas lihat daftar organisasi menggambarkan alur proses pengguna publik dalam mengeksplorasi organisasi yang tersedia di *landing page* sistem MyOrga. Proses dimulai ketika pengguna mengakses halaman utama dan sistem menampilkan kartu-kartu organisasi yang telah terdaftar beserta nama, logo, dan deskripsi singkat masing-masing. Pengguna dapat melihat seluruh organisasi dengan mengklik tombol lihat semua apabila jumlah organisasi melebihi tampilan awal. Pengguna kemudian dapat memilih organisasi yang diminati untuk melihat informasi lebih lengkap pada halaman detail organisasi.

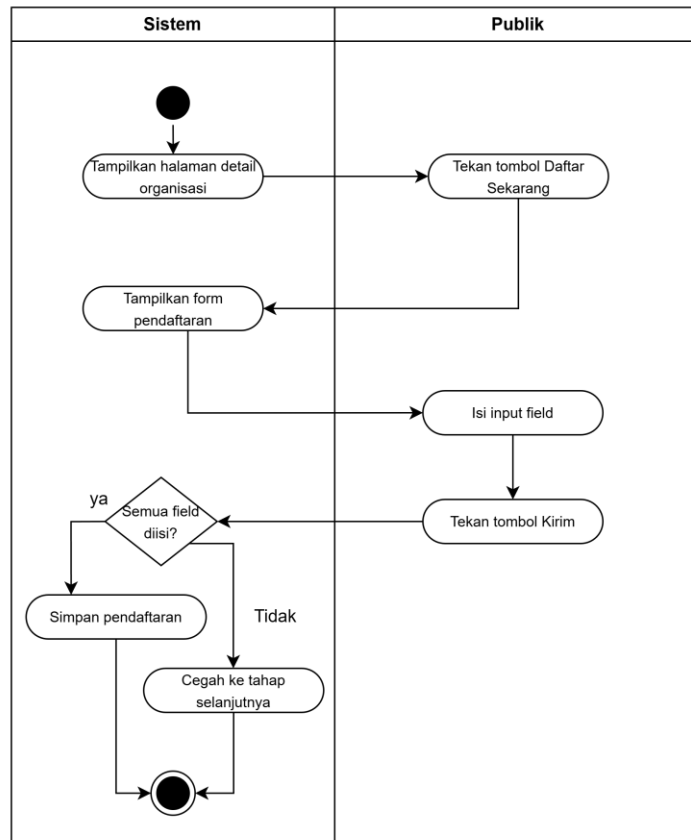
12. Diagram Aktivitas Lihat Detail Organisasi



Gambar 3.18 Diagram Aktivitas Lihat Detail Organisasi

Diagram aktivitas lihat detail organisasi menggambarkan alur proses pengguna publik dalam melihat informasi lengkap suatu organisasi. Proses dimulai ketika pengguna memilih organisasi dari daftar organisasi dan sistem menampilkan halaman detail. Sistem kemudian mengambil data organisasi dari *database* meliputi deskripsi lengkap, susunan pengurus inti beserta foto profil, struktur divisi, serta galeri foto kegiatan organisasi. Pengguna dapat menggunakan halaman ini sebagai referensi sebelum memutuskan untuk mendaftar sebagai calon anggota apabila terdapat periode rekrutmen yang sedang aktif.

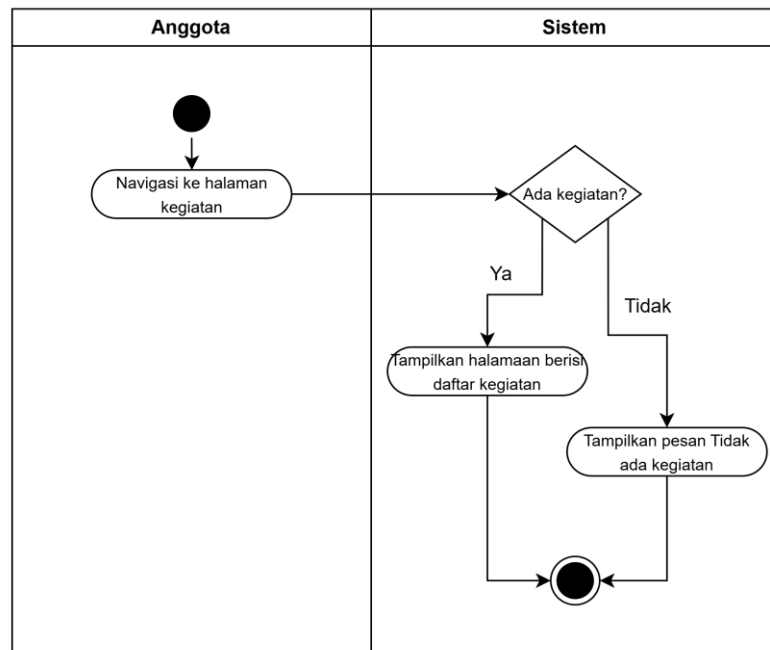
13. Diagram Aktivitas Daftar Calon Anggota



Gambar 3.19 Diagram Aktivitas Daftar Calon Anggota

Diagram aktivitas daftar calon anggota menggambarkan alur proses pendaftaran yang dilakukan oleh pengguna publik yang ingin bergabung dengan suatu organisasi. Proses dimulai ketika pengguna mengklik tombol daftar pada periode rekrutmen yang sedang aktif. Pengguna kemudian mengisi formulir pendaftaran yang mencakup motivasi, tujuan, kontribusi, pilihan divisi utama, pilihan divisi kedua, serta mengunggah berkas CV. Sistem memvalidasi data yang dimasukkan dan menyimpan aplikasi pendaftaran ke basis data dengan status pending. Aplikasi tersebut selanjutnya dapat ditinjau oleh pengurus organisasi untuk proses seleksi lebih lanjut.

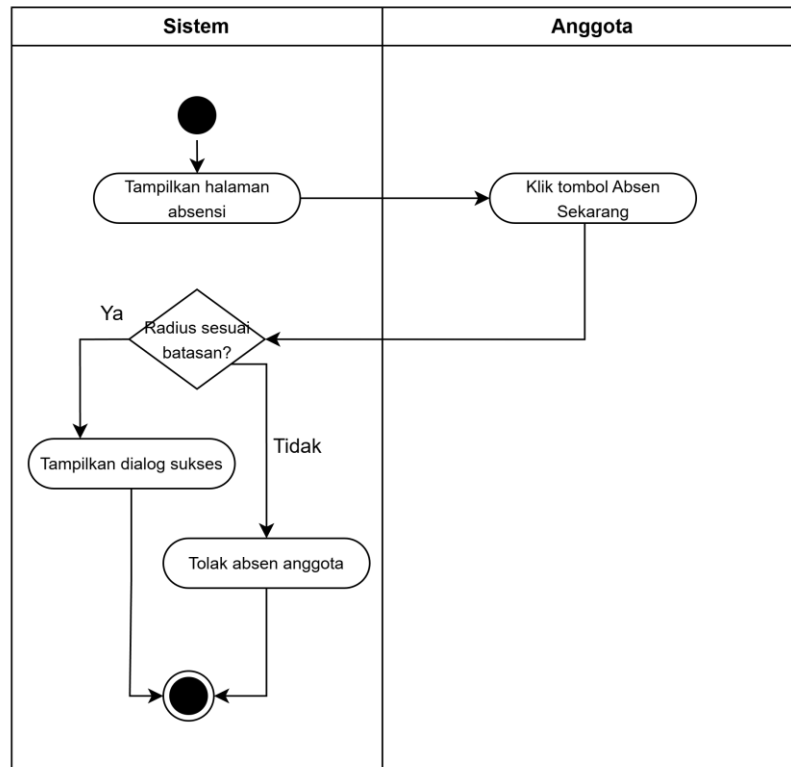
14. Diagram Aktivitas Lihat Daftar Kegiatan



Gambar 3.20 Diagram Aktivitas Lihat Daftar Kegiatan

Diagram aktivitas lihat daftar kegiatan menggambarkan alur proses anggota dalam melihat kegiatan yang diselenggarakan oleh organisasi dengan menggunakan aplikasi *mobile*. Tahapan dimulai ketika anggota berada di halaman kegiatan pada aplikasi dan sistem mengambil data kegiatan dari server melalui API. Anggota dapat memfilter tampilan kegiatan berdasarkan kategori mendatang atau selesai untuk memudahkan pencarian informasi yang relevan. Sistem menampilkan setiap kegiatan beserta tanggal, waktu, lokasi, dan status kehadiran anggota pada kegiatan tersebut sehingga anggota dapat memantau riwayat keikutsertaannya secara mandiri.

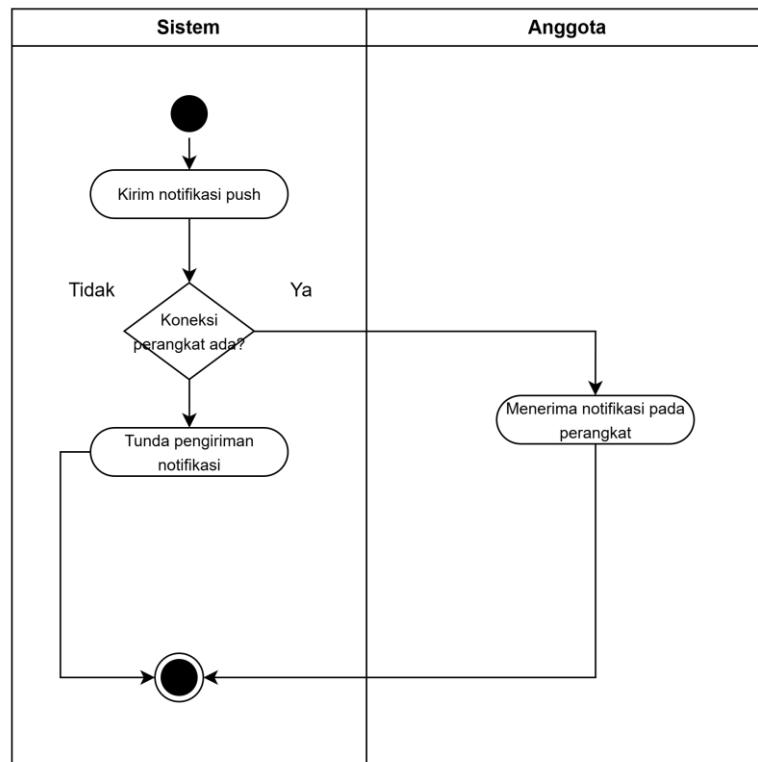
15. Diagram Aktivitas Absensi



Gambar 3.21 Diagram aktivitas Absensi

Diagram aktivitas absensi menggambarkan proses pencatatan kehadiran anggota pada kegiatan yang sedang berlangsung dengan menggunakan aplikasi *mobile*. Proses dimulai ketika anggota mengakses halaman absensi dan sistem menampilkan daftar kegiatan yang berlangsung pada hari tersebut. Selanjutnya, anggota menekan tombol absensi, kemudian sistem memperoleh koordinat GPS, lalu sistem mengirimkan data koordinat ke server untuk menghitung jarak antara posisi anggota dan lokasi kegiatan menggunakan Rumus Haversine. Apabila jarak melebihi radius yang ditentukan, sistem akan menampilkan notifikasi kegagalan, beserta informasi mengenai jarak aktual dan batas radius yang berlaku.

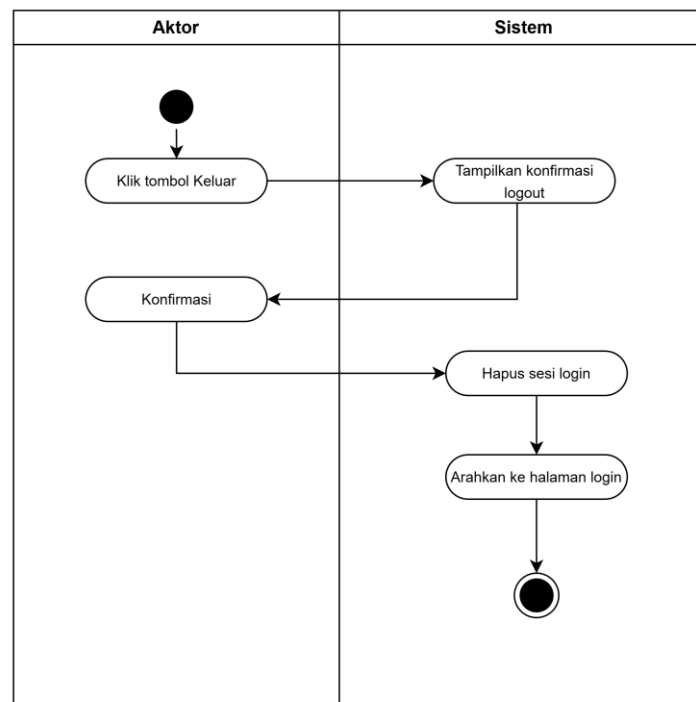
16. Diagram Aktivitas Terima Notifikasi Push



Gambar 3.22 Diagram Aktivitas Terima Notifikasi Push

Diagram aktivitas terima notifikasi *push* menggambarkan alur proses penerimaan notifikasi *real-time* oleh anggota melalui aplikasi *mobile*. Proses dimulai ketika sistem *backend* mengirimkan notifikasi melalui layanan Firebase Cloud Messaging menggunakan token perangkat yang telah terdaftar saat anggota *login*. Perangkat anggota menerima push notification dan menampilkannya di notification center perangkat. Anggota dapat membuka aplikasi dan mengakses halaman notifikasi untuk melihat daftar seluruh notifikasi beserta statusnya. Ketika anggota membuka salah satu notifikasi, sistem memperbarui status *is_read* menjadi sudah dibaca dan mengurangi jumlah badge notifikasi yang belum terbaca.

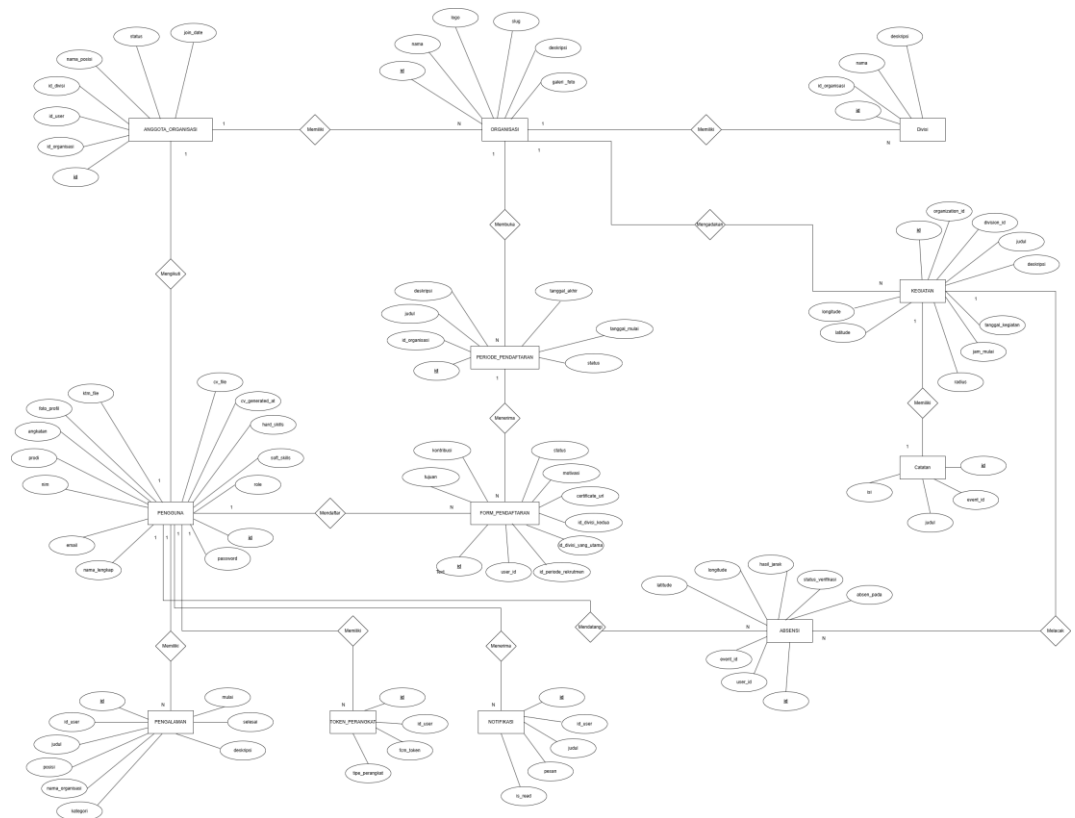
17. Diagram Aktivitas *Logout*



Gambar 3.23 Diagram Aktivitas *Logout*

Diagram aktivitas *logout* menggambarkan tahapan proses pengakhiran sesi pengguna pada sistem MyOrga baik di platform *web* maupun *mobile*. Pada platform *web*, proses dimulai ketika pengguna mengklik tombol keluar pada menu dropdown profil. Sistem menghapus sesi pengguna yang tersimpan di server dan mengarahkan pengguna kembali ke halaman *login*. Pada platform *mobile*, proses dimulai ketika pengguna mengklik tombol keluar pada halaman profil. Aplikasi mengirimkan permintaan *logout* ke server untuk mencabut Bearer token yang aktif, kemudian menghapus token yang tersimpan di *SharedPreferences* perangkat dan mengarahkan pengguna kembali ke halaman *login*.

3.2.6 ER Diagram



Gambar 3.24 ER Diagram Aplikasi

Gambar di atas merupakan diagram relasi dari sistem MyOrga yang menggambarkan struktur *database* beserta relasi antar entitas yang digunakan dalam sistem. ERD tersebut terdiri dari 12 entitas utama yang saling berelasi satu sama lain untuk mendukung seluruh proses bisnis sistem, mulai dari manajemen organisasi, rekrutmen anggota, pengelolaan kegiatan, hingga pencatatan absensi berbasis lokasi. Setiap entitas memiliki atribut yang merepresentasikan data yang disimpan, serta kardinalitas yang menggambarkan jenis hubungan antar entitas tersebut. Berikut adalah penjelasan masing-masing entitas beserta atribut dan kardinalitasnya.

1. Pengguna

Entitas Pengguna menyimpan seluruh data akun pengguna sistem. Atribut yang dimiliki meliputi id sebagai kunci utama, nama_lengkap, email, password, nim, prodi, angkatan, foto_profil, cv_file, ktm_file, hard_skills, soft_skills, role, dan cv_generated_id. Entitas ini berelasi dengan Anggota_Organisasi (1:N),

Form_Pendaftaran (1:N), Pengalaman (1:N), Token_Perangkat (1:N), Notifikasi (1:N), dan Absensi (1:N).

2. Organisasi

Entitas Organisasi menyimpan profil setiap organisasi mahasiswa. Atribut yang dimiliki meliputi id sebagai kunci utama, nama, slug, deskripsi, logo, dan galeri_foto. Entitas ini berelasi dengan Anggota_Organisasi (1:N), Divisi (1:N), Periode_Pendaftaran (1:N), dan Kegiatan (1:N).

3. Divisi

Entitas Divisi menyimpan data pembagian divisi dalam suatu organisasi. Atribut yang dimiliki meliputi id sebagai kunci utama, nama, deskripsi, dan id_organisasi sebagai kunci tamu yang mengacu pada entitas Organisasi. Entitas ini berelasi dengan Organisasi (N:1) dan Anggota_Organisasi (1:N).

4. Anggota_Organisasi

Entitas Anggota_Organisasi merupakan entitas penghubung yang merepresentasikan keanggotaan seorang pengguna dalam suatu organisasi. Atribut yang dimiliki meliputi id sebagai kunci utama, id_user, id_organisasi, id_divisi, nama_posisi, status, dan join_date. Entitas ini berelasi dengan Pengguna (N:1), Organisasi (N:1), dan Divisi (N:1).

5. Periode_Pendaftaran

Entitas Periode_Pendaftaran menyimpan data jadwal rekrutmen yang dibuka oleh organisasi. Atribut yang dimiliki meliputi id sebagai kunci utama, judul, deskripsi, id_organisasi, tanggal_mulai, tanggal_akhir, dan status. Entitas ini berelasi dengan Organisasi (N:1) dan Form_Pendaftaran (1:N).

6. Form_Pendaftaran

Entitas Form_Pendaftaran menyimpan data aplikasi yang dikirimkan oleh calon anggota pada suatu periode rekrutmen. Atribut yang dimiliki meliputi id sebagai kunci utama, user_id, id_periode_rekrutmen, id_divisi_yang_utama, id_divisi_kedua, motivasi, tujuan, kontribusi, status, certificate_url, dan *field*. Entitas ini berelasi dengan Pengguna (N:1) dan Periode_Pendaftaran (N:1).

7. Kegiatan

Entitas Kegiatan menyimpan data kegiatan yang diselenggarakan oleh organisasi. Atribut yang dimiliki meliputi id sebagai kunci utama, judul, deskripsi, `organization_id`, `division_id`, `tanggal_kegiatan`, `jam_mulai`, longitude, latitude, dan radius. Atribut longitude, latitude, dan radius digunakan sebagai dasar validasi absensi berbasis lokasi menggunakan metode Haversine. Entitas ini berelasi dengan Organisasi (N:1), Divisi (N:1), Catatan (1:1), dan Absensi (1:N).

8. Catatan

Entitas Catatan menyimpan notulensi dari setiap kegiatan yang diselenggarakan. Atribut yang dimiliki meliputi id sebagai kunci utama, `event_id`, judul, dan isi. Entitas ini berelasi dengan Kegiatan (N:1).

9. Absensi

Entitas Absensi menyimpan data kehadiran anggota pada setiap kegiatan. Atribut yang dimiliki meliputi id sebagai kunci utama, `event_id`, `user_id`, latitude, longitude, `hasil_jarak`, `status_verifikasi`, dan `absen_pada`. Atribut `hasil_jarak` menyimpan hasil perhitungan jarak antara koordinat anggota dengan koordinat kegiatan menggunakan formula Haversine. Entitas ini berelasi dengan Kegiatan (N:1) dan Pengguna (N:1).

10. Pengalaman

Entitas Pengalaman menyimpan riwayat keikutsertaan pengguna dalam organisasi atau kepanitiaan sebelumnya. Atribut yang dimiliki meliputi id sebagai kunci utama, `id_user`, judul, posisi, `nama_organisasi`, kategori, mulai, selesai, dan deskripsi. Entitas ini berelasi dengan Pengguna (N:1).

11. Token_Perangkat

Entitas Token_Perangkat menyimpan token FCM dari perangkat *mobile* milik pengguna untuk keperluan pengiriman push notification. Atribut yang dimiliki meliputi id sebagai kunci utama, `id_user`, `fcm_token`, dan `tipe_perangkat`. Entitas ini berelasi dengan Pengguna (N:1).

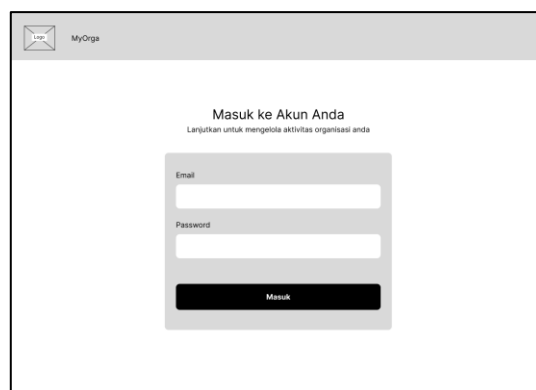
12. Notifikasi

Entitas Notifikasi menyimpan data notifikasi yang diterima oleh pengguna dari sistem. Atribut yang dimiliki meliputi id sebagai kunci utama, id_user, judul, pesan, dan is_read. Entitas ini berelasi dengan Pengguna (N:1).

3.2.7 Perancangan Antarmuka (Wireframe)

A. Rancangan Antarmuka Manajemen

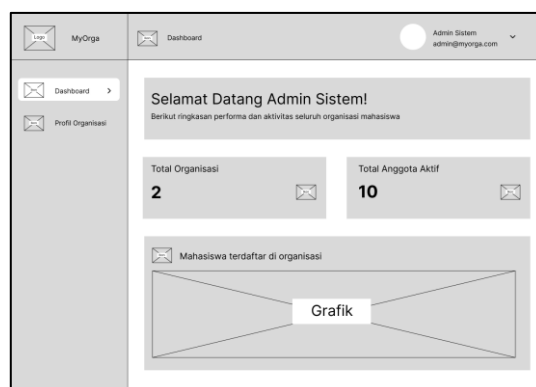
1. Halaman *Login*



Gambar 3.25 Desain Antarmuka Halaman *Login*

Gambar 3.25 merupakan rancangan halaman login manajemen yang berfungsi sebagai antarmuka pertama yang diakses oleh pengguna dengan *role* manajemen dan pengurus sebelum masuk ke sistem. Halaman ini menampilkan form autentikasi yang terdiri dari *field* email dan *password* beserta tombol masuk.

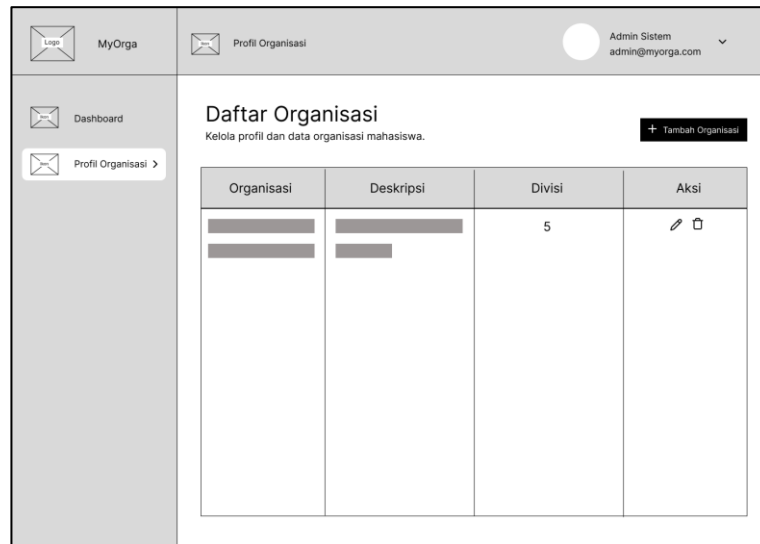
2. Halaman *Dashboard* Manajemen



Gambar 3.26 Desain Antarmuka Halaman *Dashboard* Manajemen

Halaman *dashboard* manajemen pada Gambar 3.26 menampilkan ringkasan informasi sistem secara keseluruhan. Halaman ini terdiri dari kartu statistik yang memuat total organisasi dan total anggota aktif, serta grafik area yang menggambarkan pertumbuhan jumlah anggota dalam enam bulan terakhir.

3. Halaman Profil Organisasi

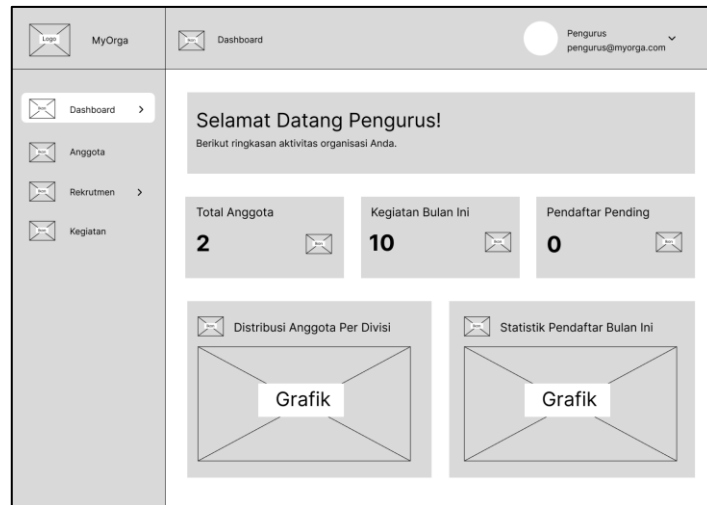


Gambar 3.27 Desain Antarmuka Halaman Profil Organisasi

Gambar 3.27 berfungsi sebagai antarmuka utama manajemen dalam mengelola data organisasi mahasiswa. Halaman ini menampilkan tabel daftar organisasi beserta informasi nama, logo, deskripsi, dan jumlah divisi, dilengkapi tombol tambah organisasi serta aksi edit dan hapus.

B. Antarmuka Pengurus

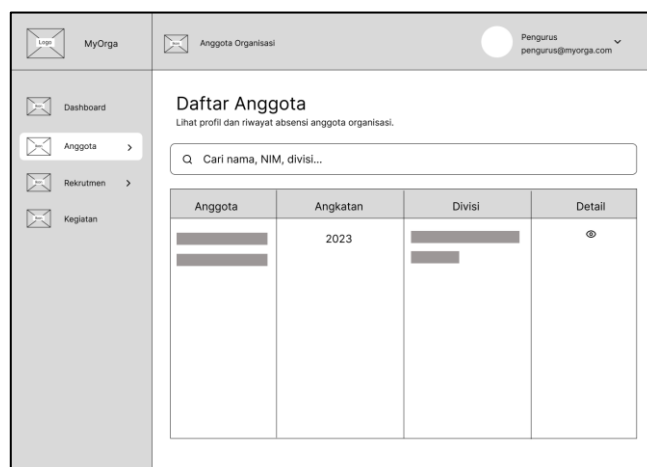
1. Halaman *Dashboard* Pengurus



Gambar 3.28 Desain Antarmuka Halaman *Dashboard* Pengurus

Halaman *dashboard* pengurus pada Gambar 3.28 menampilkan ringkasan operasional organisasi yang relevan bagi pengurus. Halaman ini memuat kartu statistik anggota, divisi, dan kegiatan bulan ini, card kegiatan terdekat, informasi pendaftar yang menunggu proses, grafik distribusi anggota per divisi, serta grafik statistik pendaftar.

2. Halaman Anggota

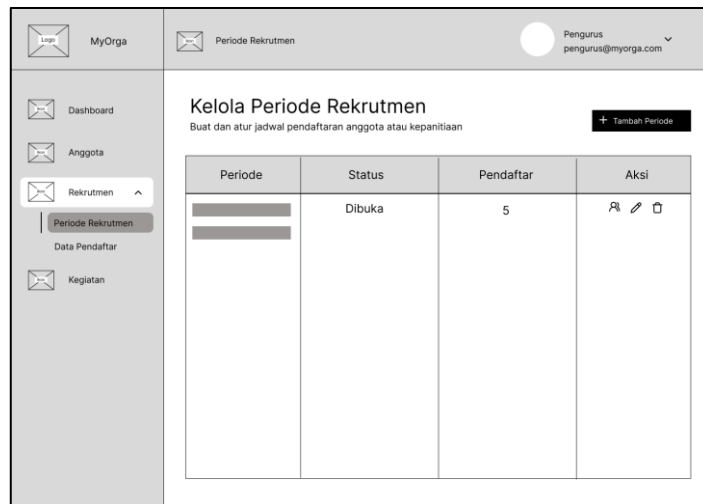


Gambar 3.29 Desain Antarmuka Halaman Anggota

Halaman anggota pada Gambar 3.29 menampilkan daftar seluruh anggota aktif dalam organisasi. Halaman ini dilengkapi fitur pencarian berdasarkan

nama, NIM, atau divisi, serta tabel yang memuat informasi nama, NIM, angkatan, divisi, dan jabatan anggota, dengan tombol untuk membuka detail profil dan riwayat absensi masing-masing anggota.

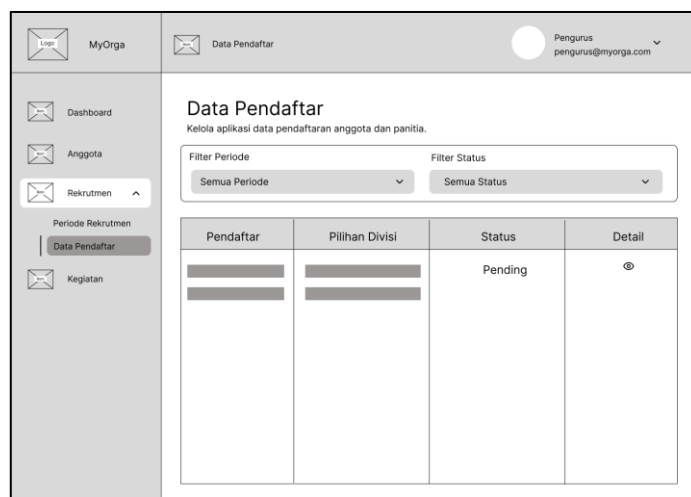
3. Halaman Periode Rekrutmen



Gambar 3.30 Desain Antarmuka Halaman Periode Rekrutmen

Halaman periode rekrutmen pada Gambar 3.30 merupakan antarmuka pengurus dalam mengelola jadwal rekrutmen organisasi. Halaman ini menampilkan tabel daftar periode beserta judul, tanggal, status, dan jumlah pendaftar, dilengkapi tombol tambah periode serta aksi lihat pendaftar, edit, dan hapus pada setiap periode.

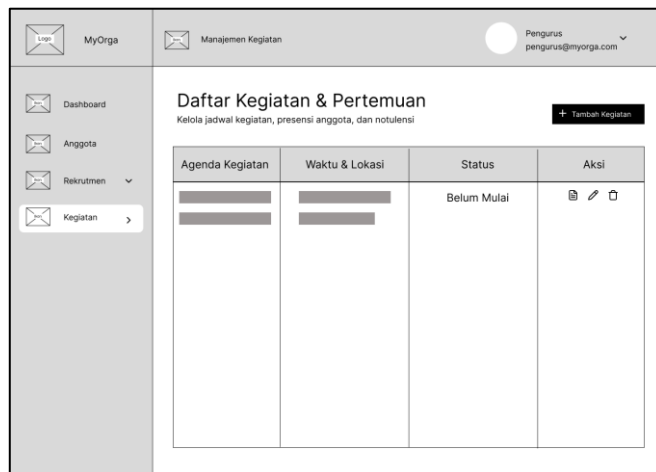
4. Halaman Data Pendaftar



Gambar 3.31 Desain Antarmuka Halaman Data Pendaftar

Halaman data pendaftar pada Gambar 3.31 menampilkan seluruh aplikasi pendaftaran yang masuk pada suatu periode rekrutmen. Halaman ini dilengkapi filter berdasarkan periode dan status aplikasi, serta tabel yang memuat nama pendaftar, pilihan divisi, dan status, dengan tombol untuk membuka detail aplikasi guna melakukan proses penerimaan atau penolakan.

5. Halaman Kegiatan

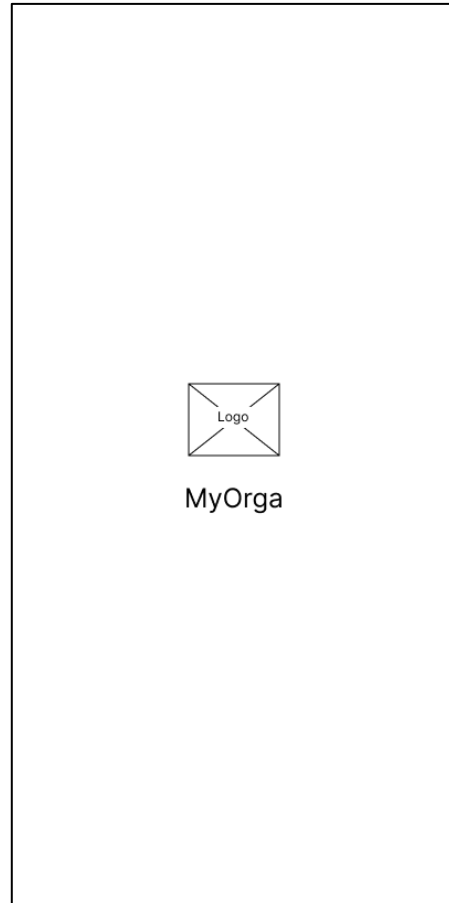


Gambar 3.32 Desain Antarmuka Halaman Kegiatan

Halaman kegiatan merupakan antarmuka pengurus dalam mengelola seluruh kegiatan organisasi. Halaman ini menampilkan tabel daftar kegiatan beserta judul, waktu, lokasi, dan status, dilengkapi tombol tambah kegiatan serta aksi untuk membuka detail kegiatan yang di dalamnya terdapat tab pengelolaan daftar hadir dan pencatatan notulensi.

C. Antarmuka Anggota

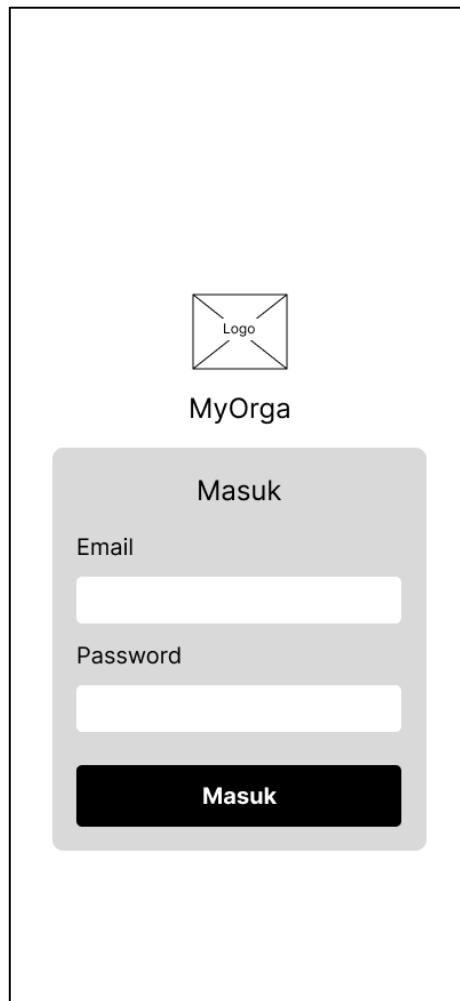
1. Halaman *Splashscreen*



Gambar 3.33 Desain Antarmuka Halaman *Splashscreen*

Halaman *splashscreen* pada Gambar 3.33 adalah layar pembuka yang ditampilkan saat aplikasi *mobile* pertama kali dijalankan. Halaman ini menyajikan logo dan nama MyOrga di tengah layar dengan latar belakang gelap, sembari sistem memeriksa status autentikasi pengguna sebelum mengarahkan ke halaman yang sesuai.

2. Halaman *Login*

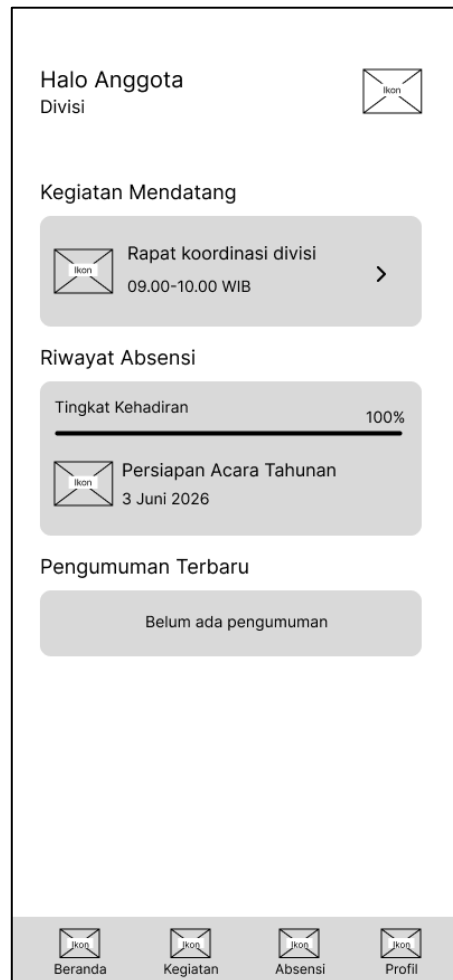


The image shows a mobile login screen design. At the top center is a square logo placeholder with the word "Logo" inside. Below the logo is the text "MyOrga". Underneath is a light gray rounded rectangle containing the login form. Inside this rectangle, the word "Masuk" is centered at the top. Below it are two labels, "Email" and "Password", each followed by a white input field. At the bottom of the rounded rectangle is a black button with the word "Masuk" in white text.

Gambar 3.34 Desain Antarmuka Halaman *Login*

Halaman *login mobile* pada Gambar 3.34 merupakan desain antarmuka autentikasi pengguna pada aplikasi *mobile* MyOrga. Halaman ini menampilkan logo di bagian atas, diikuti card form yang berisi *field* email dan *password* beserta tombol masuk untuk memulai proses autentikasi ke server.

3. Halaman *Homepage*



Gambar 3.35 Desain Antarmuka Halaman *Homepage*

Halaman *homepage* pada Gambar 3.35 menampilkan sapaan personalisasi, card kegiatan mendatang terdekat, card ringkasan statistik absensi pribadi beserta riwayat singkat, dan daftar notifikasi terbaru dari organisasi.

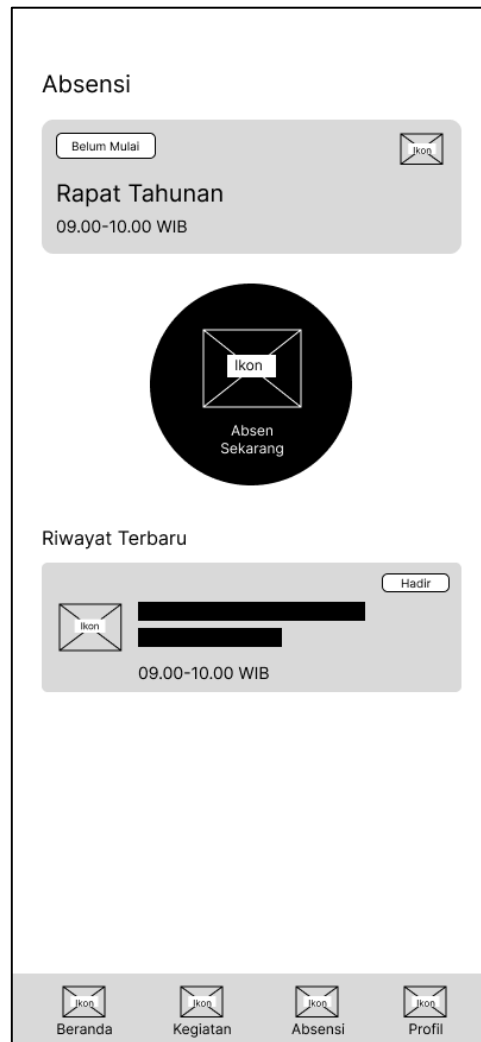
4. Halaman Kegiatan



Gambar 3.36 Desain Antarmuka Halaman Kegiatan

Halaman kegiatan pada Gambar 3.36 menampilkan daftar kegiatan organisasi yang dapat difilter berdasarkan kategori mendatang atau selesai. Setiap item pada daftar menampilkan nama kegiatan, tanggal, lokasi, dan status kehadiran anggota pada kegiatan tersebut.

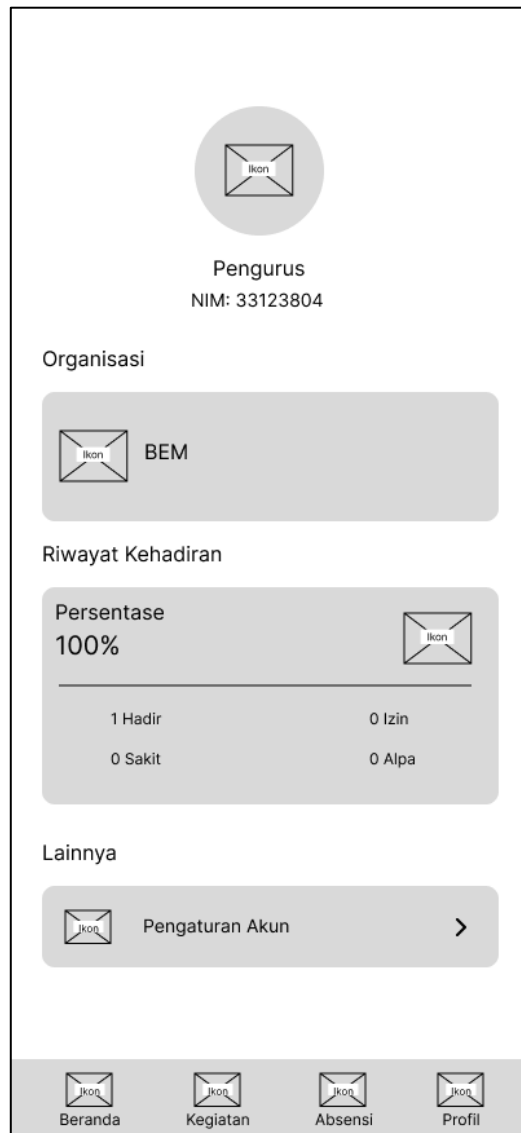
5. Halaman Absensi



Gambar 3.37 Desain Antarmuka Halaman Absensi

Halaman absensi pada Gambar 3.37 menampilkan informasi kegiatan aktif hari ini beserta tombol absen sekarang yang akan memicu proses pengambilan lokasi GPS dan validasi jarak menggunakan metode Haversine, dilengkapi riwayat absensi terbaru di bagian bawah.

6. Halaman Profil

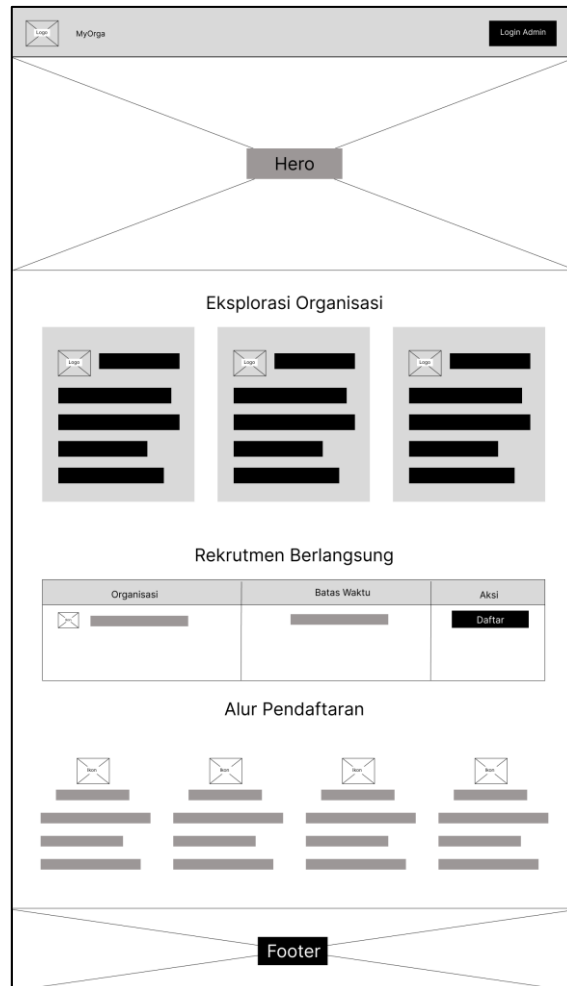


Gambar 3.38 Desain Antarmuka Halaman Profil

Halaman profil pada Gambar 3.38 menampilkan informasi akun anggota yang sedang *login* meliputi nama, NIM, angkatan, organisasi, dan divisi. Halaman ini juga memuat rekap statistik kehadiran lengkap beserta riwayat absensi per kegiatan, serta tombol pengaturan akun dan keluar di bagian bawah.

D. Antarmuka Publik

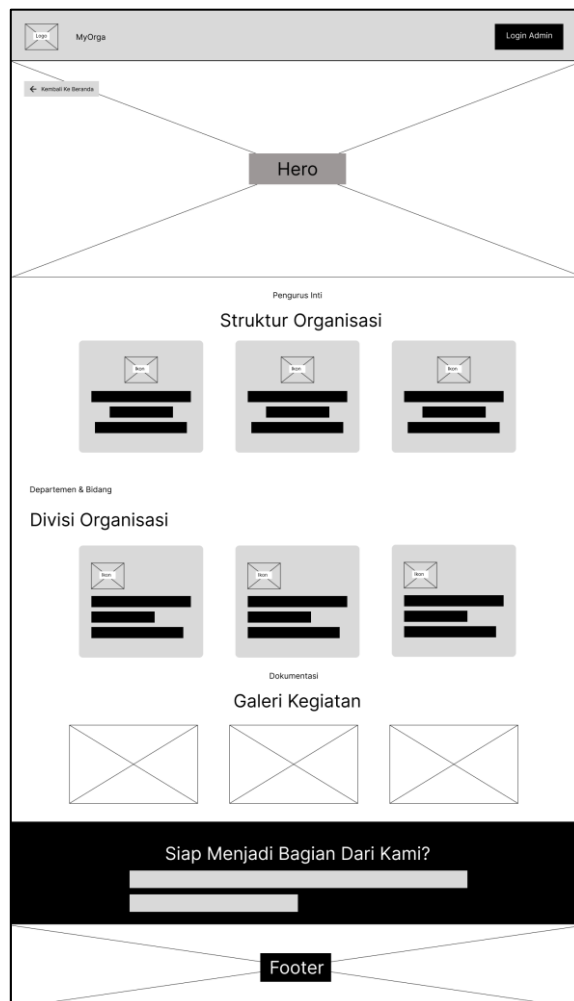
1. Halaman *Landing page*



Gambar 3.39 Desain Antarmuka Halaman *Landing page*

Gambar 3.39 merupakan antarmuka publik utama yang dapat diakses tanpa autentikasi. Halaman ini terdiri dari beberapa *section* yaitu hero *section* dengan tagline dan tombol navigasi, *section* eksplorasi organisasi yang menampilkan kartu-kartu organisasi terdaftar, *section* rekrutmen yang sedang aktif, dan *section* alur pendaftaran yang menjelaskan tahapan proses rekrutmen.

2. Halaman Detail Organisasi



Gambar 3.40 Desain Antarmuka Halaman Detail Organisasi

Gambar 3.40 menampilkan informasi lengkap suatu organisasi yang dipilih oleh pengguna publik. Halaman ini memuat deskripsi lengkap organisasi, susunan pengurus inti beserta foto profil, struktur divisi, dan galeri foto kegiatan, sehingga calon pendaftar dapat memperoleh gambaran menyeluruh sebelum memutuskan untuk mendaftar.

BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1. Hasil Implementasi

4.1.1 Deskripsi Proses Instalasi

Proses implementasi mencakup instalasi sistem, konfigurasi, dan tahap penyebaran aplikasi (*deployment*). Sistem ini dibangun menggunakan arsitektur *client-server*, dengan platform *Dashboard* berbasis *Web* dan aplikasi pengguna berbasis *Mobile*.

1. Instalasi dan Konfigurasi Backend (*Web Dashboard*)

Sistem backend dan *dashboard web* dikembangkan menggunakan *framework* Laravel (PHP). Implementasi dimulai dengan melakukan *clone* dari *repository* proyek, kemudian melakukan instalasi *dependency* menggunakan *Composer* untuk pustaka PHP dan Node Package Manager (NPM) untuk assets frontend. Selanjutnya, dilakukan konfigurasi *environment* pada file *.env* untuk menghubungkan sistem dengan database MySQL. Struktur *database* dan data awal dibentuk dengan menjalankan perintah migrasi dan *seeder* (php artisan migrate --seed). Di tahap pengembangan, server dijalankan menggunakan php artisan serve.

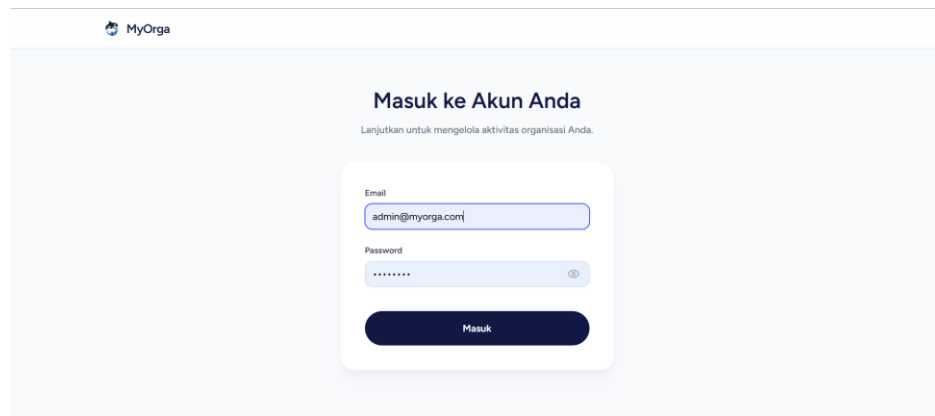
2. Instalasi dan Konfigurasi Aplikasi *Mobile*

Kerangka kerja *flutter* dengan bahasa pemrograman dart digunakan untuk mengembangkan Aplikasi *mobile* MyOrga. Proses implementasinya meliputi instalasi *dependency packages* dengan perintah flutter pub get. Selanjutnya, file konfigurasi API diatur agar aplikasi dapat berkomunikasi langsung dengan endpoint REST API yang disediakan oleh backend Laravel. Konfigurasi tambahan seperti integrasi Firebase Cloud Messaging (FCM) juga disematkan untuk menangani *push notification*.

4.1.2 Implementasi Antarmuka dan Fitur

A. Implementasi Antarmuka Manajemen

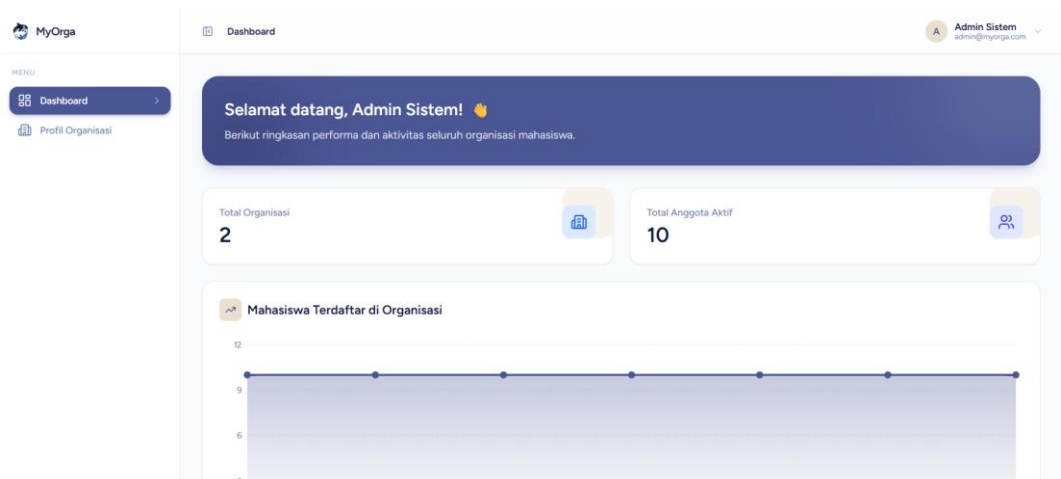
1. Halaman *Login Website*



Gambar 4.1 Implementasi Antarmuka *Login Website*

Gambar 4.1 menampilkan implementasi antarmuka *login website* dilakukan sebagai antarmuka autentikasi bagi pengguna dengan *role* manajemen dan pengurus sebelum dapat mengakses sistem. Pada halaman ini diterapkan form autentikasi serta tombol masuk untuk memproses validasi akun pengguna.

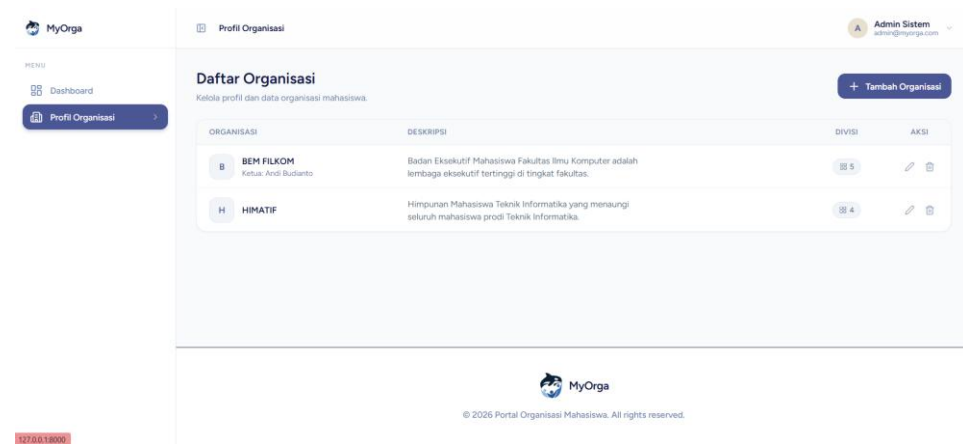
2. Halaman *Dashboard* Manajemen



Gambar 4.2 Implementasi Antarmuka *Dashboard* Manajemen

Halaman *dashboard* manajemen pada Gambar 4.2 dilakukan untuk menampilkan ringkasan informasi sistem secara keseluruhan. Halaman ini menerapkan kartu statistik yang memuat total organisasi dan total anggota aktif, serta grafik area untuk menampilkan pertumbuhan jumlah anggota dalam enam bulan terakhir.

3. Halaman Profil Organisasi

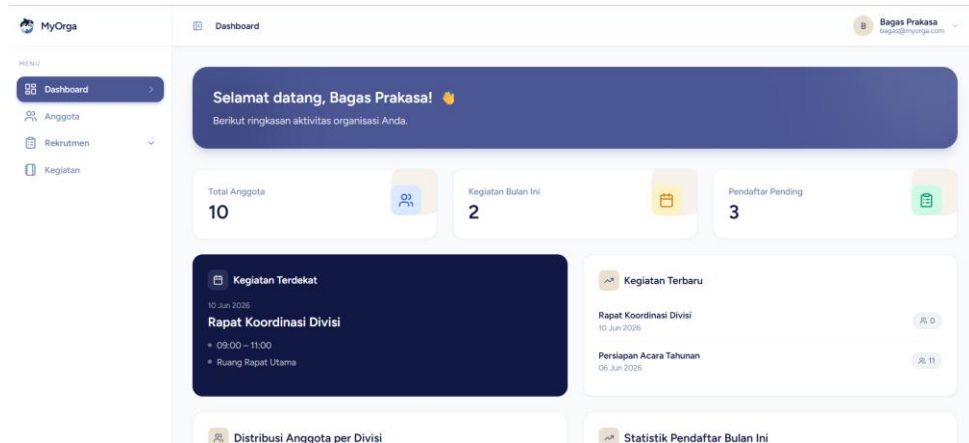


Gambar 4.3 Implementasi Antarmuka Profil Organisasi

Implementasi halaman profil organisasi pada Gambar 4.3 dilakukan sebagai antarmuka utama manajemen dalam mengelola data organisasi mahasiswa. Halaman ini menerapkan tabel daftar organisasi yang memuat informasi nama, logo, deskripsi, dan jumlah divisi, serta dilengkapi tombol tambah organisasi dan aksi edit maupun hapus pada setiap data organisasi.

B. Implementasi Antarmuka Pengurus

1. Implementasi Halaman *Dashboard* Pengurus



Gambar 4.4 Implementasi Antarmuka *Dashboard* Pengurus

Implementasi halaman *dashboard* pengurus pada Gambar 4.4 dilakukan untuk menampilkan informasi operasional organisasi yang relevan bagi pengurus. Halaman ini menerapkan kartu statistik anggota, divisi, dan kegiatan bulan ini, card kegiatan terdekat, informasi pendaftar yang menunggu proses, grafik distribusi anggota per divisi, serta grafik statistik pendaftar.

2. Implementasi Halaman Anggota

The screenshot shows the 'Anggota Organisasi' page in the MyOrga system. It features a search bar for finding members by name, NIM, or division. Below the search bar is a table listing active members with their details.

ANGGOTA	ANGKATAN	DIVISI	DETAIL
A Andi Budianto 20515020111001	2020	Pengurus Inti Ketua	Detail
B Bagas Prakasa 20515020111002	2020	Pengurus Inti Wakil Ketua	Detail
S Siti Nurhaliza 21515030011001	2021	Kementerian Dalam Negeri	Detail
P Budi Santoso 21515030011002	2021	Kementerian Dalam Negeri	Detail
C Citra Kirana 22515040111001	2022	Kementerian Luar Negeri	Detail
D Dina Mariana 22515040111002	2022	Kementerian Keuangan	Detail

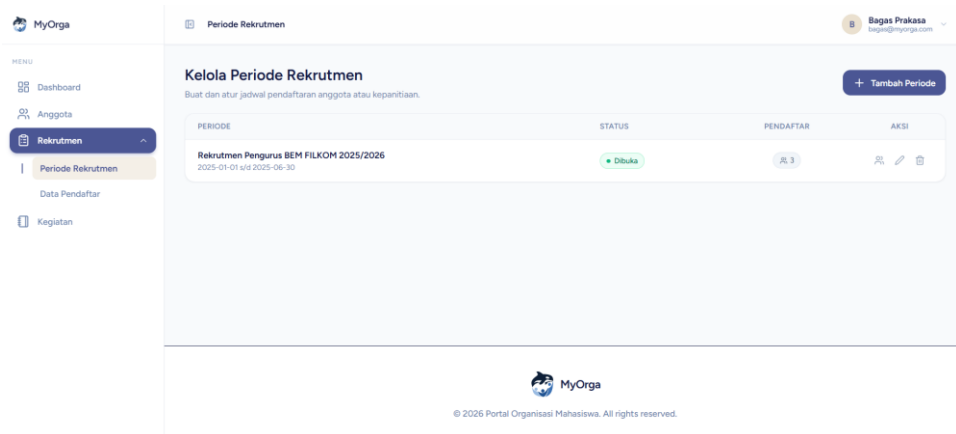
127.00.18000/members

Gambar 4.5 Implementasi Antarmuka Anggota

Implementasi halaman anggota pada Gambar 4.5 dilakukan untuk menampilkan data seluruh anggota aktif organisasi. Halaman ini menerapkan fitur pencarian berdasarkan nama, NIM, atau divisi, serta tabel

data anggota yang memuat informasi nama, NIM, angkatan, divisi, dan jabatan anggota, lengkap dengan tombol detail profil dan riwayat absensi anggota.

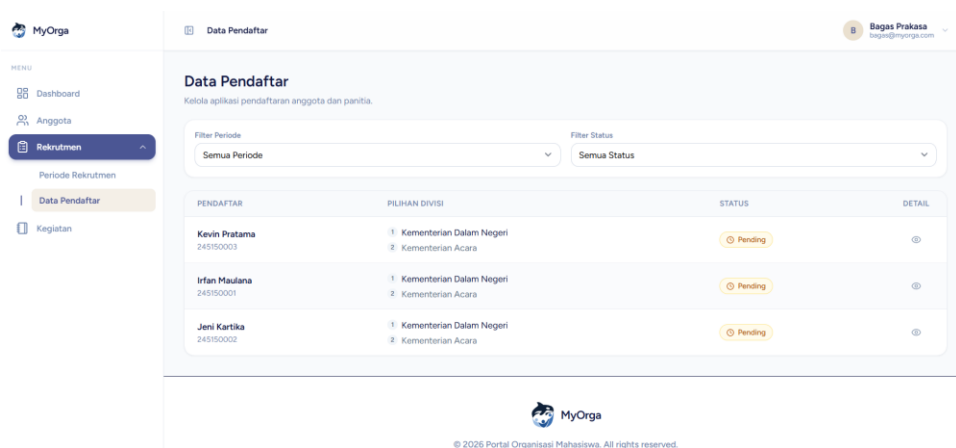
3. Implementasi Halaman Periode Rekrutmen



Gambar 4.6 Implementasi Antarmuka Periode Rekrutmen

Implementasi halaman periode rekrutmen pada Gambar 4.6 dilakukan sebagai antarmuka pengurus dalam mengelola jadwal rekrutmen organisasi. Halaman ini menerapkan tabel daftar periode yang memuat judul, tanggal, status, dan jumlah pendaftar, serta dilengkapi tombol tambah periode dan aksi lihat pendaftar, edit, serta hapus data periode.

4. Implementasi Halaman Data Pendaftar

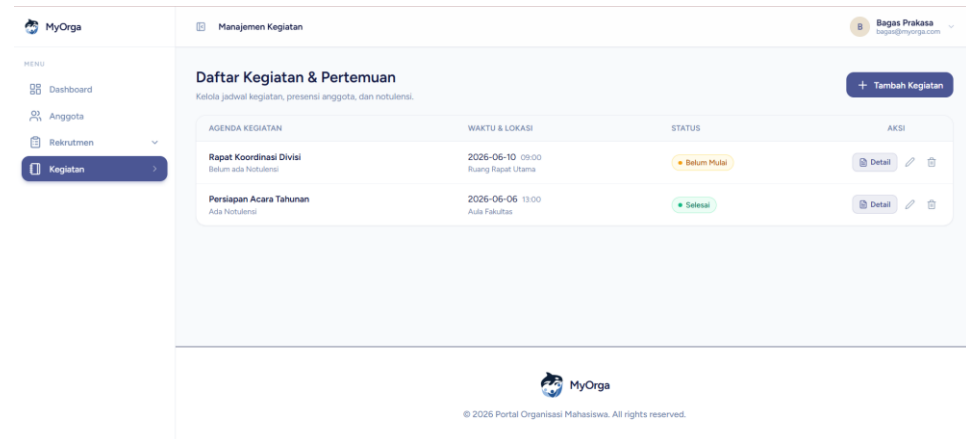


Gambar 4.7 Implementasi Antarmuka Data Pendaftar

Implementasi antarmuka data pendaftar pada Gambar 4.7 dilakukan untuk menampilkan seluruh data aplikasi pendaftaran yang masuk pada setiap

periode rekrutmen. Halaman ini menerapkan fitur filter berdasarkan periode dan status aplikasi, serta tabel data pendaftar yang memuat nama pendaftar, pilihan divisi, dan status aplikasi, dengan tombol detail aplikasi untuk proses penerimaan maupun penolakan.

5. Implementasi Halaman Kegiatan



Gambar 4.8 Implementasi Antarmuka Kegiatan

Implementasi antarmuka kegiatan pada Gambar 4.8 dilakukan sebagai antarmuka pengurus dalam mengelola kegiatan organisasi. Halaman ini menerapkan tabel daftar kegiatan yang memuat judul, waktu, lokasi, dan status kegiatan, serta dilengkapi tombol tambah kegiatan dan fitur detail kegiatan yang berisi pengelolaan daftar hadir serta pencatatan notulensi.

C. Implementasi Antarmuka Anggota

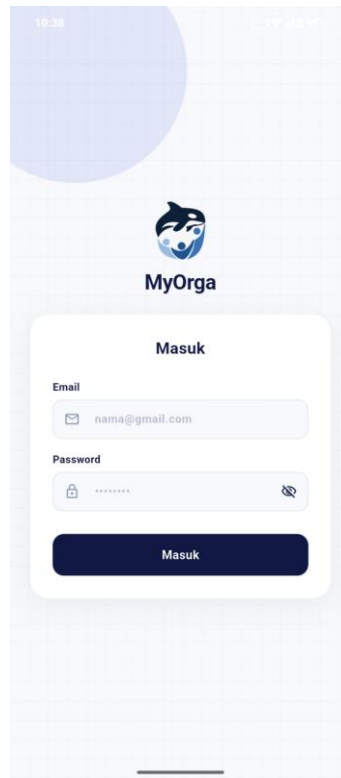
1. Implementasi Halaman *Splashscreen*



Gambar 4.9 Implementasi Antarmuka *Splashscreen*

Implementasi pada Gambar 4.9 *splashscreen* dilakukan sebagai tampilan awal saat aplikasi *mobile* dijalankan. Halaman ini menerapkan tampilan logo dan nama aplikasi MyOrga di bagian tengah layar dengan latar belakang gelap, sekaligus melakukan pengecekan status autentikasi pengguna sebelum diarahkan ke halaman berikutnya.

2. Implementasi Halaman *Login*



Gambar 4.10 Implementasi Antarmuka *Login Mobile*

Implementasi *login mobile* pada Gambar 4.10 dilakukan sebagai antarmuka autentikasi pengguna pada aplikasi MyOrga. Halaman ini menerapkan tampilan logo aplikasi pada bagian atas dan form serta tombol masuk untuk memproses autentikasi pengguna ke server.

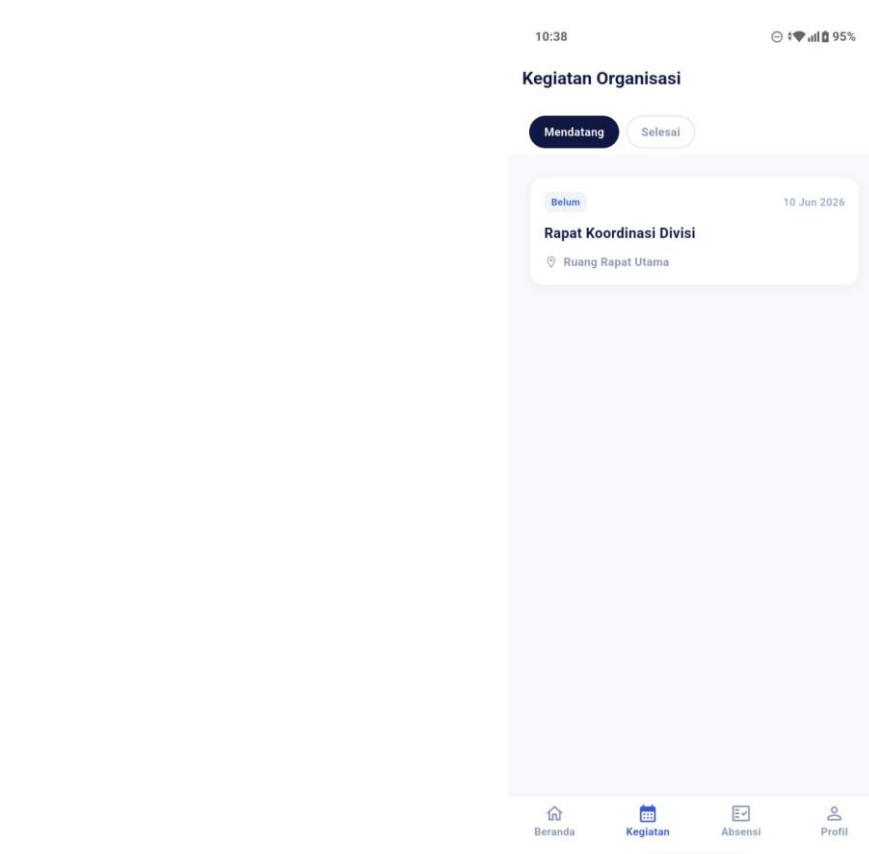
3. Implementasi Halaman *Homepage*



Gambar 4.11 Implementasi Antarmuka *Homepage*

Implementasi *homepage* pada Gambar 4.11 dilakukan sebagai tampilan utama setelah anggota berhasil masuk ke aplikasi. Halaman ini menerapkan tampilan sapaan personalisasi pengguna, card kegiatan mendatang terdekat, ringkasan statistik absensi pribadi beserta riwayat singkat, serta daftar notifikasi terbaru dari organisasi.

4. Implementasi Halaman Kegiatan



Gambar 4.12 Implementasi Antarmuka Kegiatan

Implementasi kegiatan *mobile* pada Gambar 4.12 dilakukan untuk menampilkan daftar kegiatan organisasi kepada anggota. Halaman ini menerapkan fitur filter kegiatan berdasarkan kategori mendatang atau selesai, dengan setiap item kegiatan menampilkan nama kegiatan, tanggal, lokasi, dan status kehadiran anggota.

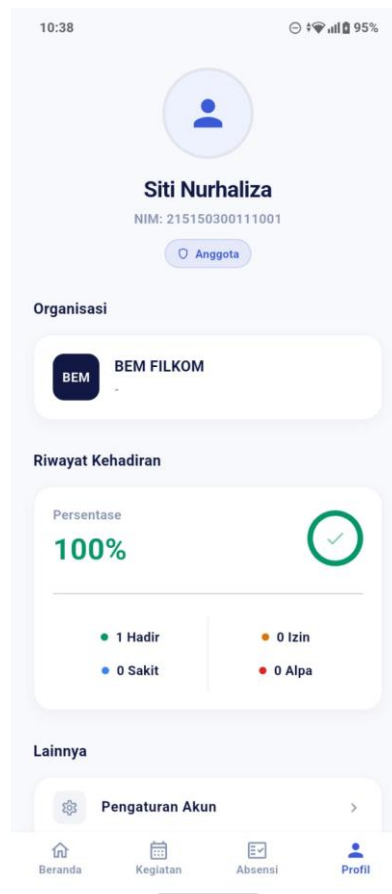
5. Implementasi Halaman Absensi



Gambar 4.13 Implementasi Antarmuka Absensi

Implementasi absensi pada Gambar 4.13 dilakukan sebagai tampilan utama anggota dalam melakukan pencatatan kehadiran kegiatan. Halaman ini menerapkan informasi kegiatan aktif hari ini beserta tombol absen sekarang yang terintegrasi dengan pengambilan lokasi GPS dan validasi jarak menggunakan metode Haversine, serta menampilkan riwayat absensi terbaru pengguna.

6. Implementasi Halaman Profil



Gambar 4.14 Implementasi Antarmuka Profil

Implementasi tampilan profil pada Gambar 4.14 dilakukan untuk menampilkan informasi akun anggota yang sedang *login*. Halaman ini menerapkan tampilan data nama, NIM, angkatan, organisasi, dan divisi, serta rekap statistik kehadiran lengkap dengan riwayat absensi per kegiatan. Selain itu, halaman ini juga dilengkapi tombol pengaturan akun dan keluar dari aplikasi.

D. Implementasi Antarmuka Publik

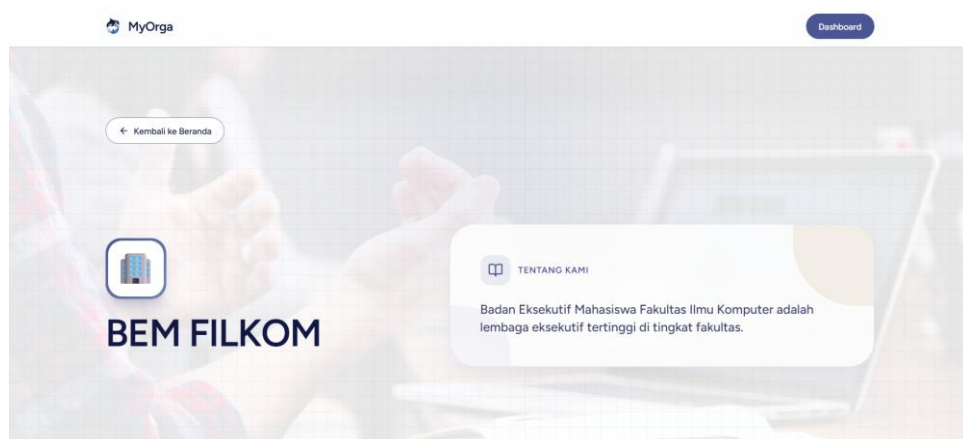
1. Implementasi Halaman *Landing page*



Gambar 4.15 Implementasi Antarmuka *Landing page*

Implementasi antarmuka *landing page* pada Gambar 4.15 dilakukan sebagai antarmuka publik utama yang dapat diakses tanpa proses autentikasi. Halaman ini menerapkan beberapa *section*, yaitu *hero section* yang memuat tagline dan tombol navigasi, *section* eksplorasi organisasi yang menampilkan kartu organisasi terdaftar, *section* rekrutmen aktif, serta *section* alur pendaftaran yang menjelaskan tahapan proses rekrutmen organisasi.

2. Implementasi Halaman Detail Organisasi

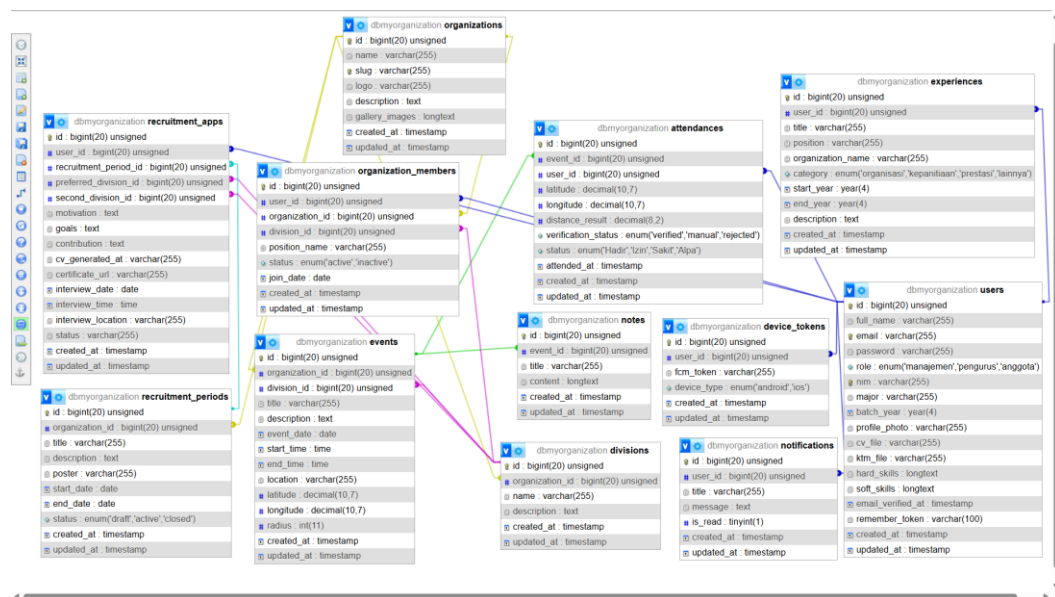


Gambar 4.16 Implementasi Antarmuka Detail Organisasi

Implementasi halaman detail organisasi pada Gambar 4.16 dilakukan untuk menampilkan informasi lengkap organisasi kepada pengguna publik.

Halaman ini menerapkan tampilan deskripsi organisasi, susunan pengurus inti beserta foto profil, struktur divisi, dan galeri foto kegiatan, sehingga calon pendaftar dapat memperoleh gambaran organisasi sebelum melakukan pendaftaran.

4.1.3 Implementasi Basis Data



Gambar 4.17 Implementasi Basis Data

Gambar 4.17 menunjukkan realisasi *database* pada sistem MyOrga yang dirancang menggunakan model relasional. Basis data terdiri atas beberapa tabel utama, yaitu *users*, *organizations*, *divisions*, *organization_members*, *events*, *attendances*, *recruitment_periods*, *recruitment_apps*, *experiences*, *notifications*, *notes*, dan *device_tokens*. Tabel *users* digunakan untuk merekam data akun pengguna, sedangkan tabel *organizations* menyimpan informasi mengenai organisasi. Hubungan antara pengguna dan organisasi direpresentasikan melalui tabel *organization_members*, yang juga memuat informasi divisi, jabatan, status keanggotaan, dan tanggal bergabung anggota dalam organisasi.

Pengelolaan kegiatan organisasi dilakukan melalui tabel *events*, sementara data kehadiran anggota pada setiap kegiatan disimpan pada tabel *attendances* yang mencakup informasi lokasi, hasil perhitungan jarak, status verifikasi, dan status kehadiran. Proses rekrutmen organisasi dikelola melalui tabel *recruitment_periods*

untuk menyimpan periode rekrutmen dan tabel *recruitment_apps* untuk menyimpan data pendaftaran calon anggota. Riwayat pengalaman organisasi maupun kepanitiaan pengguna disimpan pada tabel *experiences*. Selain itu, sistem juga dilengkapi dengan tabel *notes* untuk menyimpan catatan kegiatan, tabel *notifications* untuk mengelola notifikasi kepada pengguna, serta tabel *device_tokens* yang menyimpan token perangkat sebagai pendukung layanan *push notification*.

4.1.4 Contoh Alur Penggunaan Sistem

Sistem MyOrga diimplementasikan dengan menghubungkan empajenis pengguna, yaitu publik, anggota, pengurus, dan manajemen, sehingga seluruh proses organisasi dapat berjalan secara terintegrasi melalui platform *web* dan *mobile*. Implementasi sistem mencakup proses rekrutmen anggota baru, pengelolaan kegiatan organisasi, absensi anggota, hingga distribusi informasi organisasi secara *digital*.

1. Alur Pendaftaran Anggota Baru

Pada implementasi proses pendaftaran anggota baru, pengguna publik dapat mengakses *landing page* untuk melihat informasi organisasi yang sedang membuka rekrutmen. Pengguna kemudian dapat melakukan pendaftaran dengan mengisi formulir registrasi yang telah disediakan oleh sistem. Formulir tersebut memuat beberapa data yang diperlukan organisasi, seperti identitas diri, program studi, pengalaman organisasi, kemampuan, dan data pendukung lainnya. Setelah seluruh data diisi, sistem akan secara otomatis menghasilkan curriculum vitae (CV) berdasarkan data yang telah dimasukkan oleh pengguna itu sendiri.

Data pendaftaran beserta CV yang telah dihasilkan kemudian tersimpan ke dalam database dan dapat diakses oleh pengurus melalui *dashboard web* pada menu pengelolaan rekrutmen. Selanjutnya, pengurus melakukan proses seleksi dengan meninjau data dan CV pelamar melalui sistem. Pengurus dapat menerima atau menolak pelamar sesuai hasil seleksi yang dilakukan. Ketika pelamar dinyatakan diterima, sistem secara otomatis menambahkan data

pengguna sebagai anggota baru organisasi dan mengirimkan informasi akun agar pengguna dapat melakukan *login* ke aplikasi *mobile*.

2. Alur Pelaksanaan Kegiatan dan Absensi

Pada implementasi proses kegiatan organisasi, pengurus dapat membuat data kegiatan baru melalui *dashboard web* dengan menentukan nama kegiatan, jadwal pelaksanaan, serta titik koordinat lokasi kegiatan. Setelah kegiatan berhasil dibuat, sistem backend akan mengirimkan push notification kepada seluruh anggota untuk memberikan informasi terkait kegiatan yang akan dilaksanakan.

Saat kegiatan berlangsung, anggota dapat membuka aplikasi *mobile* dan mengakses detail kegiatan untuk melakukan absensi kehadiran. Sistem akan mengambil lokasi GPS pengguna dan melakukan validasi jarak menggunakan metode Haversine untuk memastikan anggota berada dalam radius lokasi yang telah ditentukan sebelum absensi dicatat ke database. Setelah kegiatan selesai, pengurus dapat memantau status kehadiran anggota, melakukan perubahan status absensi secara *manual* apabila diperlukan, serta menambahkan notifikasi kegiatan yang nantinya dapat diakses oleh seluruh anggota melalui aplikasi *mobile*.

4.2. Pengujian

4.2.1 Pengujian Black Box

Berikut merupakan tabel hasil pengujian *black box* pada aplikasi MyOrga yang dilakukan untuk memastikan seluruh fungsi dan fitur sistem telah berjalan sesuai dengan kebutuhan dan tujuan perancangan. Melalui pengujian ini, dapat diketahui apakah fungsi-fungsi yang terdapat pada aplikasi telah berjalan sesuai skenario yang dirancang serta dapat memberikan hasil yang telah sesuai dengan kebutuhan pengguna.

1. Dashboard Web

Tabel 4.1 Tabel Pengujian Dashboard Web

Modul/Fitur	Skenario Pengujian	Data Uji	Hasil yang Diharapkan	Status
Autentikasi	<i>Login</i> dengan data valid	Email: admin@myorg.com Password: password123	Masuk ke halaman <i>Dashboard</i> sesuai <i>role</i>	Berhasil
	<i>Login</i> dengan data tidak valid	Email: user@salah.com Password: 12345salah	Tampil pesan error kredensial tidak valid, tetap di halaman <i>Login</i>	Berhasil
	<i>Logout</i> dari sistem	Klik tombol <i>Logout</i>	Sesi berakhir, dialihkan ke halaman <i>Login</i>	Berhasil
Otorisasi <i>Role</i>	Akses rute Manajemen oleh Pengurus	Akun Pengurus mengakses URL /organization/profile	Ditolak (403 Forbidden) atau dialihkan ke halaman utama	Berhasil
<i>Landing page & Publik</i>	Melihat detail organisasi	Mengakses /organization/himatif	Menampilkan informasi publik organisasi HIMATIF	Berhasil
	Submit form pendaftaran	Input <i>field</i> diisi semua	Data tersimpan, muncul notifikasi sukses mendaftar	Berhasil
	Submit form pendaftaran (Kosong)	Salah satu input <i>field</i> kosong	Menampilkan pesan validasi error pada <i>field</i> yang wajib (misal: "Nama wajib diisi")	Berhasil
Profil Organisasi	Update profil organisasi	Input <i>field</i> deskripsi diperbarui	Data organisasi diperbarui, tampil pesan sukses	Berhasil
Manajemen Anggota	Melihat daftar anggota	Klik menu Anggota	Menampilkan tabel/list data seluruh anggota	Berhasil
	Melihat detail anggota	Klik baris data salah satu anggota	Menampilkan profil lengkap anggota tersebut	Berhasil
Rekrutmen	Membuat periode rekrutmen baru	Menambahkan periode rekrutmen baru	Periode baru tersimpan dan muncul di daftar	Berhasil
	Mengubah status pelamar	Klik tombol Terima/Tolak pada salah satu pendaftar	Status aplikasi berubah menjadi Diterima/Ditolak, email terkirim	Berhasil
Manajemen Kegiatan	Membuat kegiatan baru	Semua input <i>field</i> diisi	Kegiatan baru tersimpan dan tampil di daftar kegiatan	Berhasil

Modul/Fitur	Skenario Pengujian	Data Uji	Hasil yang Diharapkan	Status
	Mengubah detail kegiatan	Mengedit jam kegiatan	Perubahan tersimpan	Berhasil
	Kelola Absensi (<i>Manual</i>)	Mengubah status hadir anggota	Status absensi anggota tersimpan sebagai hadir	Berhasil
	Kelola catatan	Menambahkan catatan	Catatan tersimpan di detail kegiatan	Berhasil

2. Mobile App

Modul / Fitur	Skenario Pengujian	Data Uji (Input)	Hasil yang Diharapkan (Expected Result)	Status
Autentikasi	<i>Login</i> dengan kredensial valid	Email: andi@gmail.com Password: password123	Masuk ke halaman Beranda (<i>Home</i>)	Berhasil
	<i>Login</i> dengan kredensial salah	Email: sdhslkadh@gmail.com Password: jghkshf	Loading berhenti, muncul popup/snackbar error "Email atau Password salah"	Berhasil
	<i>Logout</i> aplikasi	Tekan tombol <i>Logout</i> di menu Profil	Sesi dihapus, kembali ke layar <i>Login</i>	Berhasil
Beranda & Profil	Melihat data beranda	Buka aplikasi setelah <i>login</i> menggunakan akun andi@myorga.com	Menampilkan ringkasan informasi dan daftar event terdekat	Berhasil
	Update profil pengguna	Mengubah Program Studi menjadi Teknik Mesin	Data profil terupdate di server dan UI aplikasi	Berhasil
Event & Absensi	Melihat daftar event	Buka menu Event	Menampilkan seluruh event yang dijadwalkan, misal Rapat Kerja	Berhasil
	Melakukan absensi (Berhasil)	Klik tombol "Hadir" pada event Rapat Kerja pada pukul 14:10	Status absensi berubah menjadi "Hadir"	Berhasil
	Melakukan absensi (Gagal)	Mencoba absen pada event Rapat Kerja di luar tanggal kegiatan berlangsung	Muncul pesan error "Absensi hanya bisa	Berhasil

Modul / Fitur	Skenario Pengujian	Data Uji (Input)	Hasil yang Diharapkan (Expected Result)	Status
			dilakukan pada hari kegiatan berlangsung", status tidak berubah	
Manajemen Pengurus (<i>Hanya Role Pengurus</i>)	Melihat daftar anggota	Login sebagai Pengurus, Buka menu Anggota	Menampilkan list seluruh anggota organisasi	Berhasil
	Update kehadiran anggota lain	Pilih anggota Budi Santoso -> Ubah status jadi Izin	Data absensi anggota tersebut terupdate di sistem	Berhasil
	Tambah/Edit Notulensi Event	Ketik catatan: Andi bertugas sebagai notulen hari ini.	Catatan/Notulensi tersimpan dan dapat dilihat oleh peserta lain	Berhasil
Notifikasi	Menerima Notifikasi Baru	Sistem backend mengirimkan FCM Broadcast "Event Rapat Kerja akan dimulai"	Muncul push notification di HP pengguna "Event Rapat Kerja akan dimulai"	Berhasil

4.2.2 User Acceptance Testing (UAT)

Pengujian pada aplikasi MyOrga dilakukan melalui User Acceptance Testing (UAT) dengan menggunakan metode SUS atau System Usability Scale. Pengujian ini memiliki tujuan untuk menguji tingkat kebergunaan, efisiensi, serta kemudahan penggunaan antarmuka aplikasi berdasarkan pengalaman dan penilaian langsung dari pengguna akhir.

Berdasarkan hasil pengujian yang melibatkan 33 responden, aplikasi MyOrga memperoleh skor rata-rata SUS sebesar 92,58. Nilai tersebut diperoleh dari perhitungan terhadap 10 butir pertanyaan SUS, di mana setiap jawaban responden dikonversi sesuai dengan ketentuan penilaian metode SUS, kemudian dijumlahkan dan dikalikan dengan faktor 2,5 untuk menghasilkan skor akhir. Skor rata-rata yang diperoleh menempatkan aplikasi MyOrga pada kategori *Excellent*, dengan tingkat *Acceptability* yang sangat baik serta termasuk dalam Grade A. Hasil

tersebut menunjukkan bahwa aplikasi MyOrga memiliki tingkat kebergunaan yang sangat tinggi, mudah dipahami dan dioperasikan oleh pengguna, serta mampu menciptakan kemudahan dan kenyamanan pengguna, serta mendukung pengelolaan organisasi kemahasiswaan.

BAB V KESIMPULAN

5.1 Kesimpulan

Mengacu pada keseluruhan proses pengembangan aplikasi yang dilakukan, aplikasi MyOrga berhasil dikembangkan sebagai sistem informasi organisasi berbasis *web* dan *mobile* yang mewadahi pengelolaan organisasi kemahasiswaan secara lebih efektif dan terstruktur. Hasil pengujian menunjukkan bahwa seluruh fitur sistem telah berfungsi sesuai dengan kebutuhan yang telah ditetapkan. Pengujian aspek *usability* dilakukan menggunakan metode *System Usability Scale* (SUS) dengan melibatkan 33 responden. Hasil pengujian menunjukkan nilai rata-rata sebesar 92,58 yang berada pada kategori *Excellent* dengan Grade A. Capaian tersebut mengindikasikan bahwa aplikasi MyOrga mudah digunakan oleh pengguna serta mampu mendukung proses pengelolaan organisasi kemahasiswaan agar berlangsung secara lebih efektif dan terstruktur..

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, beberapa saran untuk pengembangan aplikasi MyOrga di masa mendatang adalah untuk menambahkan menambahkan fitur yang dirasa perlu dan meningkatkan keamanan data pengguna. Pengembangan tampilan antarmuka yang lebih interaktif dan optimalisasi performa aplikasi juga dapat dilakukan untuk meningkatkan kenyamanan pengguna pada penggunaan jangka panjang.

DAFTAR PUSTAKA

- Amir Alkodri, A., Fitriyani, Isnanto, B., Suci Maya Sari, M., & Andrika, Y. (2025). Optimization of *Mobile Attendance System with Haversine Formula Method for Field Work Practice Students*. *International Journal of Information System & Technology Akreditasi*, 8(158), 312–318.
- Anita, D., & Rahmawati. (2025). Pengembangan Sistem Presensi Online Berbasis Web Dengan Validasi Lokasi Menggunakan GPS dan Haversine Formula. *OKTAL : Jurnal Ilmu Komputer Dan Science*, 828–839.
- Fauzan, R., Nasution, K., & Haramaini, T. (2025). *IMPLEMENTASI ALGORITMA HAVERSINE PADA ABSENSI PERKULIAHAN BERBASIS ONLINE DI FAKULTAS TEKNIK UNIVERSITAS ISLAM SUMATERA UTARA* (Vol. 2). <https://doi.org/10.71282/jurmie.v2i7.728>
- Firebase. (2026, June 8). *Firebase Cloud Messaging*. <https://firebase.google.com/docs/cloud-messaging>
- Firmana, I. A., Wijayanto, D., & Batubara, H. (2024). ANALISIS PERANCANGAN SISTEM INFORMASI ORGANISASI KEMAHASISWAAN UNIVERSITAS TANJUNGPURA (SOMATA). In *INTEGRATE: Industrial Engineering and Management System* (Vol. 8, Number 2). <https://jurnal.untan.ac.id/index.php/jtinUNTAN/editor/issueToc/2388-148->
- Harits, A., Rahayudi, B., & Hanggara, B. T. (2021). *Pengembangan Sistem Informasi Rekrutmen Anggota Organisasi berbasis Web dengan Pemberitahuan dan Konfirmasi Memanfaatkan Simple Mail Transfer Protocol (SMTP), SMS, dan Whatsapp Gateway (Studi Kasus Universitas Brawijaya)* (Vol. 5, Number 2). <http://j-ptiik.ub.ac.id>
- Putri, A., Sulaiman Syah, N., Haikel Oksama, M., Frilla Safiesza, Q. F., & Luthfi Hamzah, M. (2022). *Perancangan Sistem Informasi Manajemen Organisasi Mahasiswa Universitas Islam Negeri Sultan Syarif Kasim Riau*.

- Repi, A. P., Najoan, X. B. N., & Rindengan, Y. D. (2021). Rancang Bangun Aplikasi Manajemen Organisasi Online Berbasis *Mobile*. *Jurnal Teknik Elektro Dan Komputer*.
- Syauqi, M., Huzaeni, & Mahdi. (2023). *Rancang Bangun Aplikasi Agenda Berbasis Android Dengan Fitur Push Notification dan Reminder (Studi Kasus : Dewan Perwakilan Rakyat Aceh)*.
- Yusuf Amir, M., & Lestari Perdana, A. (2024). *RANCANG BANGUN SISTEM INFORMASI ORGANISASI BERBASIS WEBSITE (STUDI KASUS PENGURUS BESAR HIMPUNAN PELAJAR TURATEA KABUPATEN JENEPONTO)*. <http://jtek.ft-uim.ac.id/index.php/jtek>

DAFTAR LAMPIRAN

Lampiran A. Dokumen Proses Pengumpulan Kebutuhan Pengguna

https://drive.google.com/file/d/1MmYYw4U_7GTLsz88EkvX2TkUCHBX-6oT/view?usp=drive_link

Lampiran B. Link Produk

1. *Website*

<https://github.com/cabbageflower/DashboardMyOrga>

2. *Mobile*

<https://github.com/cabbageflower/MobileMyOrga>

Lampiran C. Dokumen Pengujian

https://drive.google.com/file/d/1KzNOj_wDV6ffPbBU5NMKnYIMNI7rVCr2/view?usp=sharing

Lampiran D. Link Desain Antarmuka

<https://www.figma.com/design/0EYZXGERciMzpw7XB6ktzj/MyOrga?node-id=3-444&t=zkivLAowWq7nhwa6-1>