

Rancang Bangun Sistem Informasi Pengelolaan Kegiatan Kepanitiaan Organisasi Mahasiswa Berbasis Website (EVOMA)

PROYEK AKHIR

Disusun Oleh:

Niati

3312301057

Disusun untuk memenuhi salah satu syarat memperoleh gelar Ahli Madya
Teknik Informatika



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
POLITEKNIK NEGERI BATAM
2026**

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa, karena atas izin dan pertolongan-Nya penulis dapat menyelesaikan proyek akhir ini secara baik dan tepat waktu. Penyusunan laporan ini menjadi salah satu syarat dan pertanggungjawaban atas pelaksanaan dan hasil proyek yang telah dilakukan.

Selama proses penyusunan laporan proyek akhir, penulis diberikan banyak bantuan, dukungan dan bimbingan dari banyak pihak. Atas segala bantuan tersebut, penulis ingin mengucapkan terima kasih 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 yang telah memfasilitasi dan memberikan dukungan kepada penulis selama menempuh pendidikan hingga dapat menyelesaikan proyek ini.
2. Bapak Ahmad Hamim Thohari, S.S.T., M.T. selaku Kepala Jurusan Teknik Informatika Politeknik Negeri Batam yang telah berperan dalam memberikan dukungan selama perkuliahan.
3. Ibu Yeni Rokhayati, S.Si., [M.Sc](#) selaku Kepala Program Studi Teknik Informatika Politeknik Negeri Batam yang telah berperan dalam memberikan dukungan hingga di tahap menyelesaikan studi.
4. Ibu Cyntia Lasmi Andesti, S.Kom., M.Kom, selaku dosen pembimbing yang telah memberikan arahan, masukan serta bimbingan hingga selesainya pengerjaan proyek.
5. Kedua orang tua dan keluarga tercinta yang selalu memberikan doa, dukungan, motivasi, perhatian dan semangat kepada penulis sepanjang proses pengerjaan proyek ini.
6. Rekan-rekan satu tim lomba yang telah bekerja sama, memberikan dukungan dan kontribusi dalam pengembangan proyek sehingga dapat mencapai hasil yang membanggakan.
7. Rekan-rekan sejawat yang meluangkan waktu berbagi ilmu, pengalaman dan dukungan selama proses pengerjaan proyek.

Laporan ini disusun dengan tujuan untuk menjelaskan proses, hasil, dan implementasi dari proyek yang telah dikerjakan. Penulis berharap laporan ini memberikan manfaat bagi pembaca maupun pihak lain. Penulis menyadari bahwa laporan ini memiliki keterbatasan. Oleh sebab itu, segala kritik dan saran yang membangun sangat diharapkan sebagai perbaikan di masa yang akan datang.

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.....	13
1.3. Tujuan.....	13
1.4. Batasan Masalah.....	13
1.5. Manfaat.....	14
1.6. Kolaborasi dan Sinergi dalam Pelaksanaan Proyek.....	15
BAB II TINJAUAN PUSTAKA.....	16
2.1. Tinjauan Solusi dan Sistem Terkait.....	16
2.2. Landasan Konsep dan Teknologi.....	19
2.2.1 Sistem Informasi.....	20
2.2.2 Organisasi Mahasiswa (Ormawa).....	20
2.2.3 Manajemen Acara.....	20
2.2.4 Role-Based Access Control (RBAC).....	21
2.2.5 Quasar Framework.....	21
2.2.6 NestJs Framework.....	21
2.2.7 Relational Database Management System (RDBMS).....	22
2.3. Metode Pengembangan Produk.....	22
BAB III ANALISIS DAN PERANCANGAN.....	25
3.1. Analisis Kebutuhan.....	25
3.1.1 Proses Bisnis Berjalan.....	25
3.1.2 Gambaran Umum Sistem.....	26
3.2. Perancangan.....	27
3.2.1 Kebutuhan Fungsional.....	27
3.2.2 Kebutuhan Non-Fungsional.....	29
3.2.3 Diagram Use Case.....	30
3.2.4 Skenario Use Case.....	31
3.2.5 Activity diagram.....	55
3.2.6 ER Diagram.....	77
3.2.7 Perancangan Antarmuka (Wireframe).....	80

BAB IV IMPLEMENTASI DAN PENGUJIAN.....	91
4.1. Hasil Implementasi.....	91
4.1.1 Deskripsi dan Proses Implementasi.....	91
4.1.2 Fitur Aplikasi.....	92
4.1.3 Alur Penggunaan Aplikasi.....	106
4.2. Pengujian.....	109
4.2.1 Pengujian Fungsional.....	109
4.2.2 Pengujian User Acceptance Testing UAT.....	111
BAB V KESIMPULAN.....	113
5.1 Kesimpulan.....	113
5.2 Saran.....	113
DAFTAR PUSTAKA.....	114
DAFTAR LAMPIRAN.....	116

DAFTAR TABEL

Tabel 2.1 Tinjauan Aplikasi Sejenis.....	17
Tabel 3.1 Kebutuhan Fungsional.....	27
Tabel 3.2 Kebutuhan Non Fungsional.....	29
Tabel 3.3 Use Case Scenario Tambah Acara.....	31
Tabel 3.4 Use Case Scenario Edit Acara.....	32
Tabel 3.5 Use Case Scenario Hapus Acara.....	33
Tabel 3.6 Use Case Scenario Menyetujui Panitia.....	34
Tabel 3.7 Use Case Scenario Menolak Panitia.....	35
Tabel 3.8 Use Case Scenario Mengelola Hak Akses Pengguna.....	36
Tabel 3.9 Use Case Scenario Mengelola Hak Akses Pengguna.....	37
Tabel 3.10 Use Case Scenario Tambah Divisi.....	38
Tabel 3.11 Use Case Scenario Edit Divisi.....	39
Tabel 3.12 Use Case Scenario Hapus Divisi.....	40
Tabel 3.13 Use Case Scenario Mendaftarkan Akun.....	41
Tabel 3.14 Use Case Scenario Melihat Absensi.....	42
Tabel 3.15 Use Case Scenario Melihat Rapat dan Notulen.....	43
Tabel 3.16 Use Case Scenario Mendaftar Acara.....	44
Tabel 3.17 Use Case Scenario Mengelola Absensi.....	45
Tabel 3.18 Use Case Scenario Tambah Rapat.....	46
Tabel 3.19 Use Case Scenario Edit Rapat.....	47
Tabel 3.20 Use Case Scenario Hapus Rapat.....	48
Tabel 3.21 Use Case Scenario Notulen Rapat.....	48
Tabel 3.22 Use Case Scenario Mendapatkan Sertifikat.....	49
Tabel 3.23 Use Case Scenario Masuk ke Sistem.....	50
Tabel 3.24 Use Case Scenario Menerima Notifikasi.....	51
Tabel 3.25 Use Case Scenario Mengelola Profil.....	52
Tabel 3.26 Use Case Scenario Mengubah Kata Sandi.....	53
Tabel 4.1 Dependensi dan Tools yang Digunakan.....	92
Tabel 4.2 Pengujian Fungsional.....	110
Tabel 4.3 Pengujian UAT.....	111
Tabel 1 Lampiran Hasil Kuesioner Pengurus Organisasi.....	116
Tabel 2 Lampiran Hasil Kuesioner Pengurus Organisasi.....	117

DAFTAR GAMBAR

Gambar 2.1 Metode Pengembangan Produk.....	23
Gambar 3.1 Proses Bisnis Berjalan.....	25
Gambar 3.2 Gambaran Umum Sistem.....	27
Gambar 3.3 Use Case Diagram.....	31
Gambar 3.4 Activity Diagram Tambah Acara.....	55
Gambar 3.5 Activity Diagram Tambah Acara.....	56
Gambar 3.6 Activity Diagram Hapus Acara.....	57
Gambar 3.7 Activity Diagram Menyetujui Panitia.....	58
Gambar 3.8 Activity Diagram Menolak Panitia.....	59
Gambar 3.9 Activity Diagram Menolak Panitia.....	60
Gambar 3.10 Activity Diagram Mengelola Sertifikat.....	61
Gambar 3.11 Activity Diagram Tambah Divisi.....	62
Gambar 3.12 Activity Diagram Edit Divisi.....	63
Gambar 3.13 Activity Diagram Hapus Divisi.....	64
Gambar 3.14 Activity Diagram Mendaftarkan Akun.....	65
Gambar 3.15 Activity Diagram Melihat Absensi.....	66
Gambar 3.16 Activity Diagram Melihat Rapat dan Notulen.....	67
Gambar 3.17 Activity Diagram Mendaftar Acara.....	68
Gambar 3.18 Activity Diagram Mengelola Absensi.....	69
Gambar 3.19 Activity Diagram Tambah Rapat.....	70
Gambar 3.20 Activity Diagram Edit Rapat.....	71
Gambar 3.21 Activity Diagram Hapus Rapat.....	72
Gambar 3.22 Activity Diagram Kelola Notulen.....	73
Gambar 3.23 Activity Diagram Mendapatkan Sertifikat.....	74
Gambar 3.24 Activity Diagram Masuk ke Sistem.....	75
Gambar 3.25 Activity Diagram Menerima Notifikasi.....	75
Gambar 3.26 Activity Diagram Mengelola Profil.....	76
Gambar 3.27 Activity Diagram Mengubah Kata Sandi.....	76
Gambar 3.28 Entity Relationship Diagram.....	77
Gambar 3.29 Antarmuka Mengelola Acara.....	80
Gambar 3.30 Antarmuka Verifikasi Panitia Acara.....	81
Gambar 3.31 Antarmuka Mengelola Hak Akses Pengguna.....	81
Gambar 3.32 Antarmuka Mengelola Sertifikat.....	82
Gambar 3.33 Antarmuka Mengelola Divisi.....	82
Gambar 3.34 Antarmuka Mendaftarkan Akun.....	83

Gambar 3.35 Antarmuka Melihat Absensi.....	83
Gambar 3.36 Antarmuka Melihat Rapat dan Notulen.....	84
Gambar 3.37 Antarmuka Mendaftar Acara.....	85
Gambar 3.38 Antarmuka Mengelola Absensi.....	86
Gambar 3.39 Antarmuka Mengelola Rapat dan Notulen.....	87
Gambar 3.40 Antarmuka Mendapatkan Sertifikat.....	88
Gambar 3.41 Antarmuka Masuk ke Sistem.....	88
Gambar 3.42 Antarmuka Menerima Notifikasi.....	89
Gambar 3.43 Antarmuka Mengelola Profil.....	89
Gambar 3.44 Antarmuka Mengubah Kata Sandi.....	90
Gambar 4.1 Halaman Kelola Acara.....	93
Gambar 4.2 Halaman Tambah Edit Acara.....	93
Gambar 4.3 Halaman Detail Acara.....	94
Gambar 4.4 Halaman Verifikasi Panitia Acara.....	95
Gambar 4.5 Halaman Hak Akses Panitia.....	95
Gambar 4.6 Halaman Kelola Sertifikat.....	96
Gambar 4.7 Halaman Detail Kelola Sertifikat.....	96
Gambar 4.8 Halaman Kelola Divisi.....	97
Gambar 4.9 Halaman Tambah Edit Divisi.....	97
Gambar 4.10 Halaman Daftar Akun.....	98
Gambar 4.11 Halaman Melihat Absensi.....	99
Gambar 4.12 Halaman Melihat Rapat.....	99
Gambar 4.13 Halaman Melihat Notulen.....	100
Gambar 4.14 Halaman Daftar Acara.....	101
Gambar 4.15 Halaman Mendaftar Acara.....	101
Gambar 4.16 Halaman Mengelola Absensi.....	102
Gambar 4.17 Halaman Rapat dan Notulen Admin.....	102
Gambar 4.18 Halaman Rapat dan Notulen Koordinator.....	103
Gambar 4.19 Halaman Mendapatkan Sertifikat.....	104
Gambar 4.20 Halaman Masuk.....	104
Gambar 4.21 Halaman Notifikasi.....	105
Gambar 4.22 Halaman Profil.....	105
Gambar 4.23 Halaman Masuk.....	106
Gambar 4.24 Alur Penggunaan Aplikasi Admin.....	107
Gambar 4.25 Alur Penggunaan Aplikasi Koordinator.....	108
Gambar 4.26 Alur Penggunaan Aplikasi Panitia.....	109

DAFTAR LAMPIRAN

Lampiran A. Dokumen Proses Pengumpulan Requirement.....	116
Lampiran B. Link Produk.....	118
Lampiran C. Dokumen Pengujian.....	118

ABSTRAK

Salah satu yang memiliki peran penting dalam mendukung perkembangan kemampuan kepemimpinan, kerjasama dan kemampuan manajemen mahasiswa adalah organisasi kemahasiswaan. Namun, proses pengelolaan kepanitiaan acara organisasi mahasiswa masih dilakukan secara manual dan tidak terintegrasi dengan informasi yang tersebar di berbagai platform. Kondisi tersebut menyebabkan ketidak terpusatnya informasi dan dokumentasi yang sulit untuk ditelusuri. Maka, dikembangkan *Event Organization Management* (EVOMA) sebagai sistem informasi pengelolaan kepanitiaan organisasi mahasiswa berbasis website yang terpusat. Pengembangan sistem dilakukan menggunakan metode *Waterfall* yang terdiri atas tahapan analisis, desain, implementasi, pengujian dan pemeliharaan. Sistem dibangun menggunakan *framework* Quasar sebagai *frontend* dan NestJs sebagai *backend*. Hasil pengujian fungsional menggunakan metode *Black Box Testing* menunjukkan bahwa 277 test case berhasil dijalankan dengan hasil sesuai. Kemudian, hasil pengujian *User Acceptance Testing* (UAT) dengan *System Usability Scale* (SUS) yang melibatkan 30 responden perolehan skor rata-rata 86,75 yang termasuk dalam kategori *Excellent* dan *Acceptable*. Hasil tersebut menunjukkan bahwa EVOMA mampu memenuhi kebutuhan pengguna dalam mengelola kepanitiaan kegiatan organisasi mahasiswa secara terstruktur dan terpusat, sehingga mendukung digitalisasi pengelolaan kegiatan organisasi mahasiswa.

Kata Kunci: organisasi mahasiswa, kepanitiaan acara, sistem informasi, *Waterfall*

BAB I

PENDAHULUAN

1.1. Latar Belakang

Organisasi mahasiswa berperan dalam pengembangan karakter, kreativitas dan kemampuan kolaboratif mahasiswa. Dalam pelaksanaannya, organisasi mahasiswa akan secara terus-menerus menjalankan kegiatan operasional kepanitiaan (Wiguna et al., 2026), yang melibatkan anggota internal organisasi maupun mahasiswa kampus yang memenuhi kriteria partisipan.

Namun, dalam pelaksanaannya, proses pengelolaan dan penyebaran informasi kegiatan masih dilakukan secara manual. Penyebaran informasi akan dilakukan melalui media sosial seperti Instagram dan WhatsApp. Sementara itu, proses lanjutan hanya melalui media *chat* dan google Form dengan alur yang tidak terdokumentasi dengan baik.

Kondisi tersebut menyebabkan penyebaran informasi ke berbagai platform dan tidak terpusatnya informasi menjadi sistem yang terintegrasi. Hal ini menyebabkan sulitnya dalam memperoleh detail kegiatan, dokumentasi, pendaftaran, dan *timeline* pelaksanaan yang terstruktur. Cara manual tersebut berpotensi menyebabkan informasi bertumpuk dan sulit untuk ditelusuri kembali. Sehingga, sebuah sistem yang terintegrasi dan terpusat sangat diperlukan untuk mengurangi risiko tidak teraturnya penjadwalan agenda (Adimurdaka, 2022), kurang jelasnya informasi mengenai detail kegiatan, *track record* kegiatan sebelumnya, dan pemantauan jumlah pendaftar (Wiguna et al., 2026).

Dalam upaya memperoleh gambaran mengenai kondisi pengelolaan kepanitiaan dan kebutuhan pengguna, maka penulis melakukan penyebaran kuesioner terhadap anggota internal organisasi dan mahasiswa yang pernah mengikuti kegiatan kepanitiaan yang dilaksanakan oleh organisasi. Kuesioner tersebut memuat beberapa aspek, yang terdiri atas proses pendaftaran panitia, seleksi dan pembagian divisi, pengelolaan absensi, dokumentasi notulen rapat, penerbitan sertifikat, penyebaran informasi kegiatan, serta kebutuhan terhadap sistem yang dirancang.

Berdasarkan hasil kuesioner yang telah penulis laksanakan terhadap 6 anggota internal organisasi dan 6 mahasiswa yang pernah mengikuti kepanitiaan, didapat hasil bahwa mayoritas responden menyatakan bahwa proses pendaftaran, seleksi, pembagian divisi, absensi, hingga penerbitan sertifikat masih dilakukan secara manual. Pada kelompok mahasiswa yang pernah mengikuti kepanitiaan, seluruh responden menyatakan bahwa pendaftaran dan seleksi masih dilakukan secara manual, informasi pembukaan kepanitiaan juga masih disebarakan melalui media seperti WhatsApp, Instagram atau informasi langsung, tidak melalui satu sistem terpusat.

Sebagai solusi atas kendala dalam pengelolaan informasi, pelaksanaan kepanitiaan dan program kerja yang dilaksanakan oleh organisasi kemahasiswaan. Sebagai solusi atas permasalahan, dirancang sistem informasi acara organisasi kemahasiswaan berbasis website yang terintegrasi dan terpusat. Sistem yang berfungsi untuk menyebarkan informasi kepanitiaan, mengelola pendaftaran, pembagian divisi, absensi, notulen, serta penyebaran sertifikat secara terpusat.

Melalui kuesioner yang telah disebarakan didapat bahwa 5 responden yang merupakan anggota internal organisasi dan 5 responden yang merupakan mahasiswa yang pernah mengikuti kepanitiaan menyatakan bahwa sistem informasi pengelolaan kepanitiaan acara diperlukan untuk membantu dalam pengelolaan kegiatan agar lebih terstruktur dan terdokumentasi dengan baik. Data ini menunjukkan bahwa adanya urgensi nyata terhadap sistem pengelolaan acara organisasi dengan sistem yang terintegrasi.

Melalui penerapan sistem informasi ini, diharapkan proses penyebaran informasi dan pengelolaan kegiatan kepanitiaan organisasi mahasiswa di lingkungan kampus dapat dilaksanakan secara efisien, terstruktur dan terdokumentasi secara sistematis. Penelitian ini difokuskan pada perancangan dan pembangunan sistem informasi acara kepanitiaan organisasi mahasiswa berbasis website sebagai upaya dalam mendukung digitalisasi pengelolaan kegiatan mahasiswa menjadi sebuah sistem yang terintegrasi dan terpusat.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah dibahas, terdapat beberapa masalah yang dihadapi, yaitu sebagai berikut:

1. Bagaimana merancang dan membangun sistem informasi acara organisasi kemahasiswaan berbasis website yang terintegrasi dan terpusat?
2. Bagaimana sistem dapat memfasilitasi penyebaran informasi kegiatan secara efektif kepada mahasiswa?
3. Bagaimana sistem dapat membantu pengelolaan kepanitiaan, pendaftaran peserta, absensi, dan dokumentasi kegiatan secara terstruktur?

1.3. Tujuan

Berdasarkan rumusan masalah, Tujuan proyek ini adalah sebagai berikut:

1. Merancang dan membangun sistem informasi acara organisasi kemahasiswaan berbasis website yang terintegrasi dan terpusat.
2. Merancang dan membangun sistem yang dapat memfasilitasi penyebaran informasi kegiatan kepada mahasiswa.
3. Merancang dan membangun sistem yang membantu pengelolaan kepanitiaan, peserta, absensi dan dokumentasi kegiatan secara terstruktur.

1.4. Batasan Masalah

Batasan masalah pada penelitian ini digunakan sebagai acuan untuk menghindari keluasan pembahasan dan menjaga penelitian tetap pada jalur yang sesuai dengan tujuan. Maka batasan masalah pada penelitian ini adalah sebagai berikut:

1. Sistem yang dikembangkan berbasis website dan belum mencakup *mobile*.
2. Sistem digunakan untuk pengelolaan kegiatan pada organisasi mahasiswa di lingkungan kampus.
3. Sistem tidak mencakup pengelolaan keuangan kegiatan.
4. Sistem tidak mencakup proses pendaftaran dan pengelolaan anggota tetap organisasi mahasiswa.

5. Sistem hanya menyediakan hak akses terbatas berdasarkan peran (*role*) untuk kebutuhan pengelolaan acara kegiatan kepanitiaan.
6. Sistem tidak terintegrasi dengan sistem akademik kampus.
7. Permasalahan dan data penelitian dibatasi pada organisasi mahasiswa di Politeknik Negeri Batam, serta belum mendukung penerapan multi-kampus.

1.5. Manfaat

Adapun manfaat dari penelitian ini adalah sebagai berikut:

1. Manfaat Teoritis

- a. Diharapkan penelitian mampu dijadikan referensi dalam pengembangan sistem informasi terutama di lingkup organisasi kampus.
- b. Diharapkan penelitian dapat menambah pengetahuan dalam penerapan teknologi informasi yang mendukung pengelolaan dan penyebaran informasi secara terstruktur dan terintegrasi.

2. Manfaat Praktis

- a. Memberikan solusi digital berupa sistem informasi dan pengelolaan acara kepanitiaan organisasi kemahasiswaan.
- b. Menyediakan dokumentasi dan informasi yang terstruktur.
- c. Meningkatkan efisiensi dalam penyebaran informasi dan memperoleh informasi.
- d. Memudahkan proses pendaftaran kepanitiaan secara efektif dan efisien.

3. Kontribusi terhadap SDGs

Penelitian ini turut memberikan kontribusi terhadap *Sustainable Development Goals* (SDGs) pada poin 4 (*Quality Education*) dan poin 9 (*Industry, Innovation and Infrastructure*). Dimana pada poin 4 penelitian mendukung kegiatan organisasi mahasiswa dan penyebaran informasi yang lebih efektif di lingkungan kampus dengan memanfaatkan teknologi

informasi. Kemudian, juga mendukung poin 9 dengan memberikan inovasi teknologi digital dalam lingkungan pendidikan.

1.6. Kolaborasi dan Sinergi dalam Pelaksanaan Proyek

Proyek akhir ini dilaksanakan melalui kolaborasi dari berbagai pihak guna memastikan solusi yang dikembangkan sesuai dengan kebutuhan pengguna serta sudah memiliki kualitas sesuai standar. Bentuk kolaborasi yang dilakukan berupa diskusi mengenai alur dari aplikasi dan kolaborasi akademik.

Kolaborasi dengan dosen pembimbing dilakukan secara berkala untuk memperoleh arahan, masukan, serta evaluasi terkait sistem yang sedang dikembangkan dan penyusunan dokumen-dokumen terkait seperti laporan. Selain itu, bentuk kolaborasi lain juga dilakukan kepada rekan mahasiswa calon pengguna aplikasi. Khususnya, mahasiswa yang sudah pernah mengikuti kepanitiaan acara yang pernah dilaksanakan oleh organisasi dan anggota internal organisasi (pengurus acara).

Melalui kolaborasi, penulis memperoleh masukan terkait kebutuhan pengguna, alur proses kepanitiaan, serta kemudahan dalam pengembangan sistem yang sesuai dengan kebutuhan pengguna. Bentuk kolaborasi dan sinergi tersebut menjadi bagian penting dalam mendukung keberhasilan pengembangan sistem serta memastikan solusi yang dihasilkan dapat memberikan manfaat secara optimal.

BAB II

TINJAUAN PUSTAKA

2.1. Tinjauan Solusi dan Sistem Terkait

Solusi dan sistem terkait yang relevan dengan pengembangan aplikasi EVOMA digunakan sebagai referensi dalam proses perancangan sistem. Tinjauan ini dilakukan untuk mengetahui solusi, fitur utama, teknologi yang digunakan serta kelebihan dan keterbatasan dari sistem yang telah dikembangkan sebelumnya.

Penelitian yang dilakukan oleh Sihombing (2023), dengan judul “Perancangan Sistem Informasi Pengelolaan Kegiatan Himpunan Mahasiswa Menggunakan *System Development Life Cycle*” menghasilkan sebuah aplikasi yang dikembangkan dengan menggunakan metode *waterfall*. Aplikasi ini mendukung mekanisme *approval user* yang sudah mencakup kontrol terhadap akses. Namun, aplikasi yang dibangun hanya berfokus pada kegiatan himpunan secara umum dan tidak memiliki detail pembagian divisi dan absensi.

Penelitian selanjutnya dilakukan oleh Sihotang & Yutantom (2021), dengan judul “Tata Kelola Organisasi Mahasiswa melalui Pengembangan Sistem Informasi” menghasilkan sebuah aplikasi yang dikembangkan menggunakan metode *waterfall*. Sistem yang dibangun adalah sistem berbasis website yang bertujuan untuk membantu program kerja organisasi mahasiswa dan membantu dalam pengelolaan keuangan. Namun, penelitian ini hanya berfokus pada program kerja organisasi dan pengelolaan keuangan sehingga belum mencakup manajemen operasional.

Penelitian yang dilakukan oleh Mokodompit et al., (2024), dengan judul “Sistem Informasi Pengelolaan Administrasi Organisasi Kemahasiswaan Berbasis Web di Universitas Negeri Gorontalo” menghasilkan sebuah aplikasi yang dikembangkan dengan metode *waterfall* serta pengujian dilakukan menggunakan metode *blackbox* dan *whitebox*. Sistem yang dikembangkan membantu dalam manajemen administrasi organisasi, peminjaman fasilitas, pengelolaan perguruan,

dan monitoring secara langsung. Namun, sistem belum mendukung manajemen teknis kepanitiaan *event*.

Penelitian dilakukan oleh Wiguna et al., (2026), dengan judul “Rancang Bangun Sistem Informasi Organisasi Mahasiswa Berbasis Website dengan Metode *Extreme Programming* pada Universitas Primakara” menghasilkan sebuah aplikasi yang membantu dalam meningkatkan penyebaran informasi, mendukung pendaftaran dan evaluasi pelaksanaan kepanitiaan, pengarsipan dokumen serta publikasi informasi organisasi mahasiswa. Namun sistem belum menyediakan manajemen operasional *event* seperti pembagian tugas.

Penelitian oleh Riswanto & Arayaji (2024) dengan judul “Implementasi Sistem Manajemen *Event* Terintegrasi untuk Meningkatkan Efektivitas Kegiatan Organisasi Mahasiswa” menghasilkan sebuah sistem yang membantu dalam manajemen *event* dengan mengadakan fitur yang cukup lengkap secara administratif. Namun, sistem lebih berfokus pada administrasi *event* bukan operasional panitia, belum menampilkan manajemen struktur kepanitiaan secara detail.

Tabel 2.1 Tinjauan Aplikasi Sejenis

Pengembang/ Tahun	Nama Aplikasi	Fitur Utama	Teknologi	Kelebihan	Keterbatasan
Sihombing (2023)	Sistem Informasi Pengelolaan Kegiatan Himpunan Mahasiswa	Login, dashboard, manajemen pengguna, pengelolaan himpunan mahasiswa, pengajuan kegiatan, dokumentasi kegiatan, verifikasi kegiatan dan laporan kerja.	Sistem berbasis web, SDLC <i>Waterfall</i> , ERD, UML (<i>Use case & Activity Diagram</i>)	Mendukung pengelolaan kegiatan himpunan dengan terstruktur, mulai dari pengajuan sampai pelaporan kegiatan.	Belum menyediakan fitur manajemen divisi, absensi, sertifikat dan notifikasi,
Sihotang & Yutantom (2021)	Ormawasite	Pengelolaan anggota, agenda, absensi, proker, pengajuan dan pelaporan anggaran,	HTML, PHP, Codeigniter, Bootstrap, PostgreSQL	Sistem terintegrasi, mempermudah pengelolaan aktivitas dan keuangan ormawa.	Berfokus pada pengelolaan dokumen dan keuangan, belum memiliki fitur tanda tangan elektronik

		monitoring kegiatan, <i>e-banner</i> , video, dan integrasi dengan sistem kemahasiswaan			sehingga proses persetujuan masih terbatas dalam aspek legalitas
Mokodompit et al., (2024)	SiOrmawa (Sistem Informasi Pengelolaan Administrasi Organisasi Mahasiswa)	Pengelolaan administrasi organisasi, kepengurusan, monitoring kegiatan, pengelolaan dan peminjaman fasilitas, dokumentasi kegiatan, laporan organisasi, dan pengawasan <i>progress</i> Ormawa	Waterfall, MySQL, BlackBox dan WhiteBox Testing	Membantu administrasi organisasi, mengurangi kesalahan administrasi dan mempermudah pengelolaan fasilitas.	Belum menyediakan fitur pengelolaan kepanitiaan acara
Wiguna et al., (2026)	Sistem Informasi Organisasi Mahasiswa Universitas Primakara	Publikasi berita, pembukaan pendaftaran kepanitiaan, pendaftaran panitia, seleksi, evaluasi, rekam jejak kegiatan, arsip administrasi dan pengelolaan profil	PHP, Laravel, MySQL, UML, Extreme Programming (XP), dan <i>Black Box Testing</i>	Menyediakan pengelolaan informasi organisasi dan kepanitiaan secara terpusat.	Belum menyediakan fitur pengelolaan operasional kepanitiaan.
Riswanto & Arayaji (2024)	Sistem Manajemen Event Terintegrasi Organisasi Mahasiswa	Pengajuan event, pengelolaan peserta, <i>QR Code</i> absensi, pengelolaan laporan event, sponsorship dokumentasi kegiatan, pengiriman email otomatis, ekspor PDF dan excel,	PHP, Laravel, MySQL, HTML, CSS, JavaScript, UML, Rapid Application Development (RAD), BlackBox Testing	Mendukung pengelolaan <i>event</i> secara terintegrasi dan meningkatkan efektivitas pengelolaan <i>event</i> organisasi mahasiswa.	Belum menyediakan fitur khusus pengelolaan kepanitiaan

		serta <i>dashboard</i> monitoring			
Proyek ini	EVOMA(<i>Event Organization Management</i>)	Publikasi acara, pendaftaran panitia, seleksi panitia, pembagian divisi, absensi rapat, dan kegiatan, notulen rapat, manajemen anggota, sertifikat digital, dokumentasi kegiatan dan monitoring kepanitiaan	Quasar, NestJs, JavaScript, TypeScript, REST API, dan MySQL.	Mengelola seluruh siklus kepanitiaan acara dalam satu sistem terintegrasi mulai dari pendaftaran sampai penerbitan sertifikat.	-

2.2. Landasan Konsep dan Teknologi

Aplikasi EVOMA (*Event Organization Management*) merupakan sistem informasi pengelolaan kepanitiaan acara organisasi mahasiswa berbasis website yang dirancang untuk memfasilitasi proses penyebaran informasi dan pengelolaan kegiatan kepanitiaan organisasi mahasiswa di lingkungan kampus. Aplikasi bertujuan untuk menyatukan proses manual yang tersebar di berbagai platform seperti Whatsapp, Google Form dan Instagram menjadi satu sistem yang terintegrasi.

Sebelumnya proses penyebaran informasi dan pengelolaan acara kepanitiaan organisasi kemahasiswaan pada lingkungan penelitian dilakukan masih menggunakan cara manual dan tersebar. Kondisi tersebut menyebabkan kesulitan dalam memperoleh detail kegiatan, dokumentasi, dan pendaftaran yang terstruktur. Tersebar nya informasi di berbagai platform dan cara manual tersebut berpotensi menyebabkan informasi yang bertumpuk dan sulit untuk ditelusuri kembali. Dengan adanya EVOMA penyebaran informasi dan pengelolaan acara organisasi kemahasiswaan lebih terintegrasi dan terpusat sehingga dapat

mengurangi risiko tidak teraturnya penjadwalan agenda dan dokumentasi yang terstruktur.

2.2.1 Sistem Informasi

Noor et al., (2022) menyatakan bahwa sistem informasi memiliki keterkaitan dengan bidang lainnya yang berkolaborasi dalam mengumpulkan, memperoses, dan menyebarkan informasi yang berfungsi dalam mendukung penetapan keputusan, koordinasi, serta pengelolaan terhadap organisasi. Sistem informasi adalah bagian penting dalam penelitian mengenai rancang bangun aplikasi sistem informasi dan pengelolaan kepanitiaan organisasi kemahasiswaan dalam penyebaran dan pengumpulan untuk memudahkan terjadinya koordinasi terhadap berbagai pihak.

2.2.2 Organisasi Mahasiswa (Ormawa)

Handani & Prayoga (2022) menyatakan bahwa Organisasi mahasiswa (Ormawa) merupakan organisasi yang berada di perguruan tinggi menjadi sarana sekelompok orang atau mahasiswa untuk mengadakan kerjasama dalam mencapai tujuan bersama. Organisasi memiliki peran penting dalam menunjang perkembangan diri mahasiswa dalam mengembangkan potensi dan minat mahasiswa, sekaligus menjadi wadah pembentukan *soft skill* mahasiswa. Pengelolaan organisasi yang tepat memerlukan sebuah sistem yang terstruktur agar tujuan organisasi dapat tercapai secara optimal.

2.2.3 Manajemen Acara

Khairy et al (2026) menyatakan bahwa manajemen acara adalah proses perencanaan dan perorganisasian suatu acara secara sistematis dan terstruktur untuk mencapai tujuan tertentu. Pengelolaan acara yang baik diperlukan untuk melakukan perancangan dan pengorganisasian acara dengan efektif dan optimal.

2.2.4 Role-Based Access Control (RBAC)

Firmansyah et al (2025) menyatakan bahwa *Role-Based Access Control* (RBAC) merupakan sistem kontrol hak akses yang melakukan pengaturan hak akses berdasarkan peran pengguna, sehingga memungkinkan pengguna dalam melaksanakan tugas tertentu sesuai dengan hak akses, memungkinkan setiap informasi yang ada dapat dibatasi dan diatur sesuai dengan peran. Dengan Adanya RBAC, sistem dapat berjalan secara efektif karena memberikan pembatasan akses sesuai dengan tanggung jawab sehingga meningkatkan keamanan dan menjaga keteraturan proses kerja.

Pada konteks EVOMA, sistem manajemen organisasi mahasiswa memungkinkan pembagian peran dan hak akses yang dibagi menjadi Admin, Koordinator Divisi dan Panitia sehingga setiap pengguna hanya dapat mengakses berdasarkan fitur berdasarkan hak akses yang dimiliki.

2.2.5 Quasar Framework

Ardilla & Warsito (2025) menyatakan bahwa Quasar merupakan *framework* yang digunakan sebagai *frontend website framework*, dimana *framework* ini memberikan pengalaman *responsive* dan cukup menggunakan satu baris kode untuk melakukan pembuatan aplikasi untuk tiga platform berbeda. Quasar juga menyediakan berbagai komponen yang mempermudah proses pembuatan website bagi *developer*. Pada konteks pengembangan aplikasi EVOMA, Quasar digunakan sebagai *framework frontend* untuk mendukung pengembangan sistem secara lebih cepat dan efisien, sekaligus memastikan tampilan antarmuka dan pengalaman pengguna (*user experience*) tetap optimal.

2.2.6 NestJs Framework

Muhammad & Paputungan (2024) menyatakan bahwa NestJs merupakan *framework backend* yang efektif dan terstruktur, dimana mendukung Javascript dan Typescript untuk mengkombinasikan OOP, FP dan FRP. Pada konteks pengembangan aplikasi EVOMA, NestJs

digunakan untuk membuat *backend* menggunakan Typescript secara lebih cepat dan efisien.

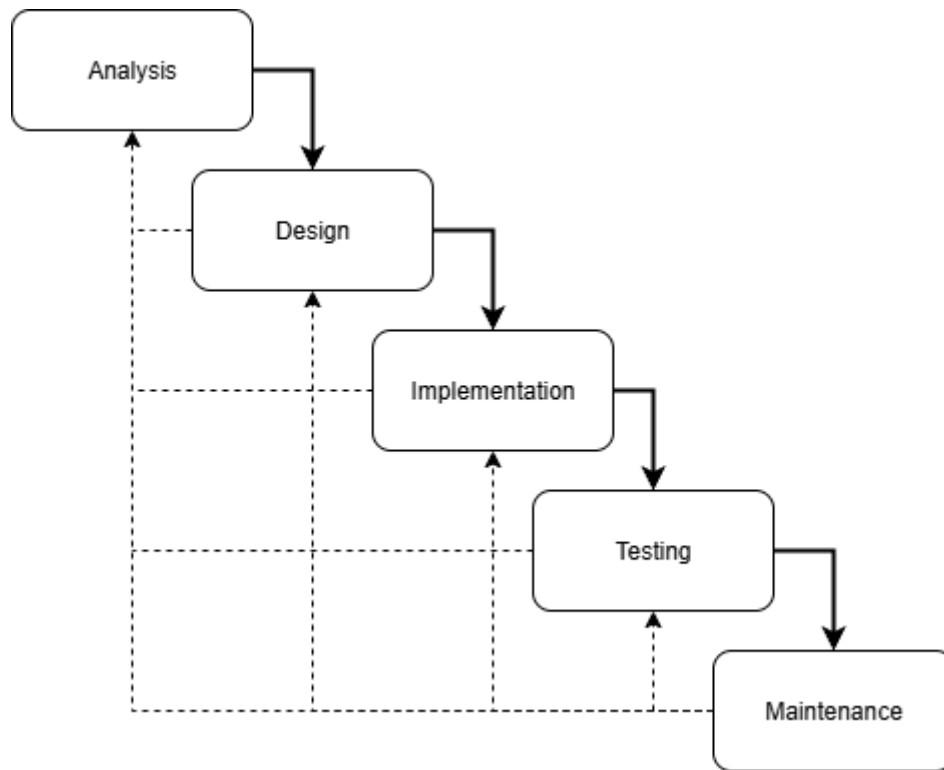
2.2.7 Relational Database Management System (RDBMS)

Menurut Adisa & Nasution (2023), *Relational Database Management System* (RDBMS) adalah sistem manajemen data yang menyimpan dan mengelola data dalam bentuk tabel-tabel yang saling terhubung. Hal ini memungkinkan untuk dilakukannya pengelolaan data yang mencakup penambahan, pembaruan, penghapusan serta pencarian data. Sebagian besar RDBMS menggunakan *Structured Query Language* (SQL) untuk mengelola dan mengakses basis data. Dalam konteks pembuatan aplikasi EVOMA, digunakan RDBMS dengan SQL untuk melakukan pengelolaan dan penyimpanan aplikasi, hal ini dipilih karena mampu menjaga konsistensi data melalui relasi antar tabel dan mendukung proses pengolahan data dengan lebih efisien.

2.3. Metode Pengembangan Produk

Metode yang diterapkan dalam proses pembuatan “Sistem Informasi Pengelolaan Kepanitiaan Acara Organisasi Mahasiswa Berbasis Website” adalah metode *waterfall*.

Bagian ini menjelaskan tahapan pengembangan perangkat lunak yang digunakan Menurut Bassil (2012) Waterfall salah satu pendekatan *Software Development Life Cycle* (SLDC), metode ini merupakan proses pengembangan perangkat lunak yang dilaksanakan secara berurutan dari atas ke bawah mirip air terjun, metode ini bersifat rekursif, dimana fase saling berurutan dan diselesaikan satu demi satu untuk berpindah ke fase selanjutnya harus menyelesaikan fase sebelumnya sepenuhnya, diulang tanpa henti hingga disempurnakan. Gambar model *Waterfall* disajikan pada Gambar 2.1 berikut.



Gambar 2.1 Metode Pengembangan Produk

Tahapan metode *Waterfall* terdiri atas tahapan analisis, tahapan desain, tahapan implementasi, tahapan pengujian, dan tahapan pemeliharaan. Berikut merupakan penjelasan metode waterfall dari masing-masing tahapan:

1. Tahapan Analisis (*Analysis*)

Tahapan analisis dikenal sebagai tahapan *Software Requirements Specification* (SRS) yaitu deskripsi mengenai perilaku sistem yang akan dikembangkan. Melaksanakan penyusunan kebutuhan fungsional dan kebutuhan non-fungsional yang menggambarkan interaksi antara pengguna dan perangkat lunak. Proses analisis dilakukan dengan observasi dan pengumpulan kuesioner kepada pihak terkait.

2. Tahapan Desain (*Design*)

Tahapan desain adalah tahapan perancangan dan perumusan solusi perangkat lunak. Pada tahap ini akan dilakukan perancangan sistem yang

meliputi pembuatan *Use Case Diagram*, *Entity Relationship Diagram*, dan *User Interface*.

3. Tahapan Implementasi (*Implementation*)

Tahapan implementasi merupakan tahapan untuk merealisasikan kebutuhan bisnis, spesifikasi desain menjadi sebuah program, basis data, situs web, atau komponen perangkat lunak yang dieksekusi melalui pemrograman.

4. Tahapan Pengujian (*Testing*)

Tahapan pengujian merupakan tahapan untuk memverifikasi dan memvalidasi perangkat lunak yang dikembangkan telah memenuhi spesifikasi. Pada tahap ini akan dilakukan pengujian menggunakan metode *Black Box Testing* dan *User Acceptance Testing* (UAT) untuk memeriksa fungsional dari aplikasi.

5. Tahapan Pemeliharaan (*Maintenance*)

Tahapan pemeliharaan juga merupakan tahapan memverifikasi dan memvalidasi bertujuan memeriksa perangkat lunak telah sesuai dengan kebutuhan yang telah ditetapkan sebelumnya.

BAB III

ANALISIS DAN PERANCANGAN

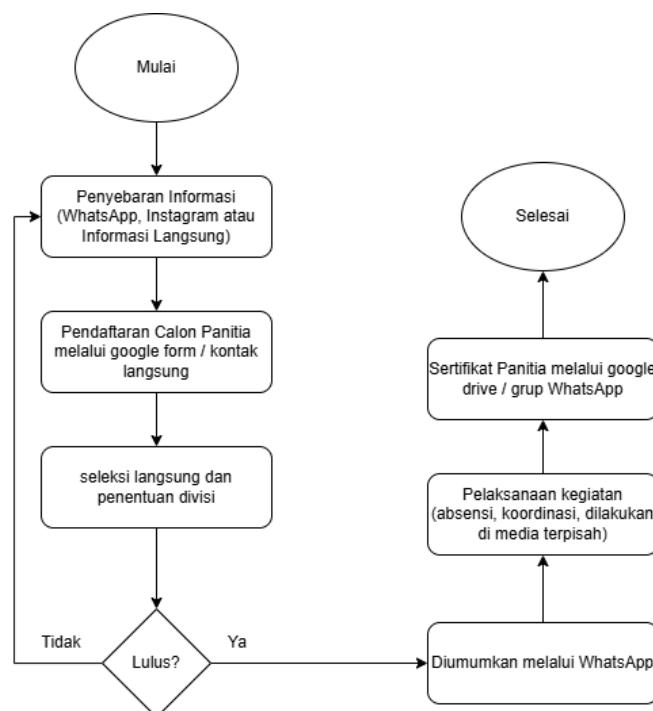
3.1. Analisis Kebutuhan

3.1.1 Proses Bisnis Berjalan

Mengacu pada hasil observasi dan kuesioner yang telah dilaksanakan, dapat dijabarkan bahwa proses bisnis berjalan dilakukan secara manual, tersebar di berbagai platform dan belum terintegrasi dalam satu sistem.

Proses penyebaran informasi dilakukan melalui media WhatsApp, Instagram dan informasi langsung. Kemudian calon panitia mendaftar melalui media terpisah seperti Google Form atau menghubungi kontak yang tertera secara langsung. Kemudian, pemilihan divisi, pengumuman kelulusan dilakukan diumumkan kembali melalui grup WhatsApp.

Sementara itu, dalam pelaksanaan kegiatan, absensi dan notulen masih dilakukan dengan media terpisah. Setelah kegiatan selesai, sertifikat panitia yang telah dibuat akan dibagikan melalui google Drive atau grup WhatsApp.



Gambar 3.1 Proses Bisnis Berjalan

3.1.2 Gambaran Umum Sistem

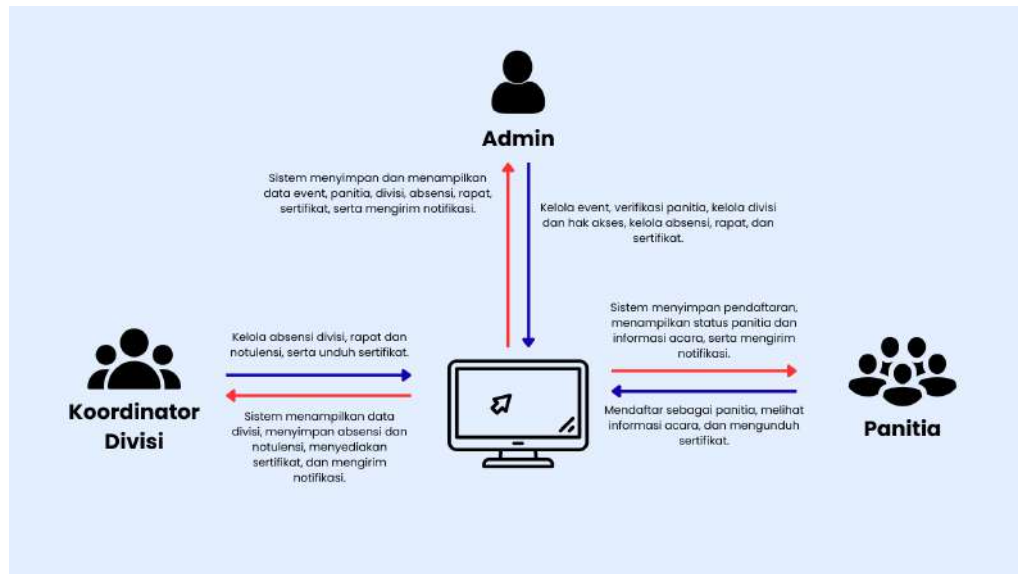
Sistem yang dirancang merupakan sistem informasi pengelolaan kepanitiaan organisasi mahasiswa berbasis website yang bertujuan untuk memfasilitasi proses pengelolaan kegiatan kepanitiaan organisasi mahasiswa menjadi satu sistem terpusat.

Sistem dirancang sebagai solusi dari permasalahan yang telah dibahas pada bagian latar belakang dan dijabarkan pada bagian proses bisnis berjalan. Secara umum, sistem akan memiliki 3 pengguna utama yang terdiri atas Panitia Acara, Koordinator Divisi dan Admin.

Panitia acara merupakan mahasiswa baik itu internal organisasi maupun mahasiswa lain yang telah memenuhi syarat yang mendaftarkan diri sebagai panitia di acara yang dilaksanakan. Koordinator Divisi adalah perangkat divisi (ketua, wakil ketua, bendahara dan sekretaris) yang bertugas dalam mengurus divisi acara. Sementara itu, Admin adalah *role* yang diperankan oleh pengurus organisasi untuk mewakili organisasi tersebut.

Panitia dapat melakukan pendaftaran acara, menerima notifikasi, menerima sertifikat melihat absensi, rapat dan notulen. Kemudian, koordinator divisi dapat melakukan pengelolaan terhadap absensi, rapat dan notulen. Sementara itu, admin dapat melakukan pengelolaan acara, pengelolaan divisi, pengelolaan sertifikat, pengelolaan hak akses, serta pengelolaan terhadap absensi, rapat dan notulen.

Melalui sistem ini, seluruh proses pengelolaan kepanitiaan diharapkan dapat berjalan lebih efektif, efisien, terdokumentasi dengan baik, dan terpusat pada satu sistem.



Gambar 3.2 Gambaran Umum Sistem

3.2. Perancangan

Tahapan perancangan dilaksanakan mengacu pada hasil analisis kebutuhan yang diperoleh pada tahap sebelumnya, bertujuan untuk memberikan gambaran mengenai konsep sistem sebelum implementasi dilakukan.

3.2.1 Kebutuhan Fungsional

Seluruh kebutuhan fungsional dari sistem EVOMA dapat mengacu pada Tabel 3.1, dijelaskan secara ringkas untuk memberikan gambaran mengenai fungsi yang tersedia pada sistem.

Tabel 3.1 Kebutuhan Fungsional

Admin	FR-01	Mengelola Acara : Admin Organisasi dapat melakukan pengelolaan data acara, meliputi pembuatan, perubahan, dan penghapusan acara.
	FR-02	Verifikasi Panitia Acara : Admin Organisasi dapat melakukan proses verifikasi terhadap mahasiswa yang mendaftar sebagai panitia acara untuk menentukan kelayakan keikutsertaan.

	FR-03	Mengelola Hak Akses Panitia: Admin Organisasi berwenang mengelola batasan akses pengguna dengan menetapkan panitia tertentu sebagai Koordinator Divisi sesuai kebutuhan acara.
	FR-04	Mengelola Sertifikat : Admin Organisasi dapat mengelola data sertifikat panitia acara.
	FR-05	Mengelola Divisi : Admin Organisasi dapat melakukan pengelolaan divisi pada setiap acara, meliputi pembuatan divisi dan penyesuaian struktur divisi sesuai dengan kebutuhan acara.
Panitia	FR-06	Mendaftarkan Akun : Panitia dapat melakukan pendaftaran akun untuk mengakses fitur lanjutan.
	FR-07	Melihat Absensi : Panitia dapat melihat absensi kehadiran dirinya sendiri pada acara yang sedang atau sudah diikuti.
	FR-08	Melihat Rapat dan Notulen : Panitia dapat melihat jadwal rapat yang sudah ditentukan oleh admin atau koordinator divisi, serta melihat notulen dari hasil rapat.
	FR-09	Mendaftar Acara: Panitia dapat melakukan pendaftaran acara yang tersedia di dalam aplikasi dan mengisi form pendaftaran.
Admin dan Koordinator Divisi	FR-10	Mengelola Absensi : Admin Organisasi dan Koordinator Divisi dapat melakukan pengelolaan absensi panitia, meliputi pencatatan dan pemantauan kehadiran panitia dalam kegiatan rapat.
	FR-11	Mengelola Rapat dan Notulen : Admin Organisasi dan Koordinator Divisi dapat mengelola data rapat, meliputi pembuatan, perubahan, dan penghapusan jadwal rapat, serta pencatatan hasil notulensi rapat.
Koordinator Divisi dan Panitia	FR-12	Mendapatkan Sertifikat : Panitia dan koordinator divisi dapat melakukan pengunduhan sertifikat jika sudah

		diterbitkan/sudah menyelesaikan kepanitiaan pada acara yang diikuti.
Admin, Koordinator Divisi, dan Panitia	FR-13	Masuk ke Sistem : Sistem menyediakan fitur autentikasi bagi pengguna yang telah terdaftar (Admin Organisasi, Koordinator Divisi, dan Panitia) untuk masuk ke dalam sistem.
	FR-14	Menerima Notifikasi : Admin, koordinator divisi dan panitia dapat menerima notifikasi dari sistem.
	FR-15	Mengelola Profil : Admin, koordinator divisi dan panitia dapat mengelola profil masing-masing.
	FR-16	Mengubah Kata Sandi : Admin, koordinator divisi dan panitia dapat mengubah kata sandi dengan memasukkan email yang valid.

3.2.2 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional disusun sebagai standar yang telah ditetapkan dalam pengembangan sistem. Daftar kebutuhan non-fungsional dari sistem yang dikembangkan dapat disajikan pada Tabel 3.2.

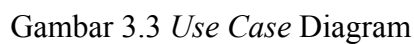
Tabel 3.2 Kebutuhan Non Fungsional

NFR-01	Kegunaan (<i>Usability</i>)	Sistem harus menampilkan pesan kesalahan yang informatif dan jelas sehingga tidak menimbulkan kebingungan terhadap pengguna.
NFR-02		Sistem menggunakan bahasa indonesia yang baik dan benar di seluruh antarmuka dan menjaga konsistensi penggunaan bahasa.
NFR-03	Keamanan (<i>Security</i>)	Sistem harus dapat menjaga keamanan data pengguna dan menyimpan data sensitif pengguna seperti kata sandi dalam bentuk yang terenkripsi.

NFR-04		Sistem mendukung autentikasi berbasis peran, sehingga pengguna dapat mengakses fungsi sesuai dengan perannya masing-masing.
NFR-05	Kompabilitas (<i>Compatibility</i>)	Sistem dapat diakses dan di jalankan di browser modern seperti Chrome, Firefox, dan Edge.

3.2.3 Diagram Use Case

Gambar 3.3 dibawah menggambarkan bahwa sistem memiliki tiga jenis aktor yang memiliki perannya masing-masing. Panitia dapat mendaftarkan akun, melihat absensi, mendaftar acara, melihat acara, melihat rapat dan notulen. Admin dapat mengelola acara, verifikasi panitia acara, mengelola hak akses pengguna, mengelola sertifikat dan mengelola divisi. Admin dan koordinator divisi dapat mengelola absensi, mengelola rapat dan notulen. Koordinator divisi dan panitia dapat mendapatkan sertifikat. Admin, koordinator divisi, dan panitia dapat masuk ke sistem, menerima notifikasi, mengelola profil dan mengubah kata sandi.



3.2.4.1 Use Case Scenario Mengelola Acara

Tabel 3.3 *Use Case Scenario* Tambah Acara

Nama Use Case	Mengelola Acara
Aktor	Admin
Deskripsi	Admin Organisasi dapat melakukan pembuatan acara baru.
Kondisi Awal	Admin mengakses sistem dengan

	kredensial valid.
Kondisi Akhir	Admin telah berhasil membuat acara baru.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin mengakses sistem. 2. Admin mengakses halaman kelola acara. 4. Admin menekan tombol tambah acara. 6. Admin mengisi formulir acara. 7. Admin memilih opsi <i>publish</i> . 9. Admin memilih opsi simpan sebagai draft.	3. Sistem menampilkan halaman kelola acara. 5. Sistem menampilkan formulir tambah acara. 8. Sistem melakukan validasi apakah inputan benar, menyimpan dan mempublish acara. 10. Sistem melakukan validasi dan menyimpan acara sebagai draft.
Alur kejadian Alternatif	
1. Admin menekan tombol <i>publish</i> namun inputan wajib belum dilengkapi, pesan kesalahan ditampilkan sistem. 2. Admin mengunggah gambar selain berformat jpg, png, atau jpeg, pesan kesalahan ditampilkan sistem. 3. Admin mengunggah gambar dengan ukuran lebih dari 2 MB, pesan kesalahan ditampilkan sistem.	

Scenario use case mengelola acara pada Tabel 3.4 menggambarkan proses dimana admin dapat mengubah acara yang telah dibuat.

Tabel 3.4 *Use Case Scenario* Edit Acara

Nama Use Case	Mengelola Acara
Aktor	Admin
Deskripsi	Admin Organisasi memiliki hak untuk mengubah informasi yang telah dibuat sebelumnya.

Kondisi Awal	Admin masuk ke sistem dengan kredensial valid.
Kondisi Akhir	Admin telah berhasil mengubah acara.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin masuk ke sistem. 2. Admin mengakses halaman kelola acara. 4. Admin menekan tombol edit acara. 6. Admin mengisi formulir acara. 7. Admin memilih opsi Simpan Perubahan.	3. Halaman kelola acara. 5. Formulir edit acara. 8. Sistem melakukan validasi apakah inputan benar, menyimpan dan menyimpan perubahan acara.
Alur kejadian Alternatif	
1. Admin menekan tombol <i>publish</i> namun inputan wajib dilengkapi, pesan kesalahan ditampilkan sistem. 2. Admin mengunggah gambar selain berformat jpg, png, atau jpeg, pesan kesalahan ditampilkan sistem. 3. Admin upload gambar dengan ukuran lebih dari 2 MB, pesan kesalahan ditampilkan sistem. 4. Admin ingin mengubah acara dengan status selesai, sistem akan menonaktifkan tombol edit.	

Scenario use case mengelola acara pada Tabel 3.5 menggambarkan proses dimana admin dapat menghapus acara yang telah di buat.

Tabel 3.5 *Use Case Scenario* Hapus Acara

Nama Use Case	Mengelola Acara
Aktor	Admin
Deskripsi	Admin Organisasi memiliki hak menghapus acara yang telah ia buat.

Kondisi Awal	Admin masuk ke sistem dengan kredensial valid.
Kondisi Akhir	Admin telah menghapus acara.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin masuk ke sistem. 2. Admin mengakses halaman kelola acara. 4. Admin menekan tombol hapus pada acara dengan status draft. 6. Admin menekan yakin.	3. Sistem menampilkan halaman kelola acara. 5. Sistem menampilkan validasi apakah admin yakin ingin menghapus acara. 7. Sistem menghapus acara.
Alur kejadian Alternatif	
Admin ingin menghapus acara dengan status selain draft, sistem akan menonaktifkan tombol hapus.	

3.2.4.2 Use Case Scenario Verifikasi Panitia Acara

Scenario use case verifikasi panitia acara pada Tabel 3.6 menggambarkan proses dimana admin dapat menyetujui panitia yang mengikuti acara setelah melakukan verifikasi data diri pendaftar.

Tabel 3.6 *Use Case Scenario* Menyetujui Panitia

Nama Use Case	Verifikasi Panitia Acara
Aktor	Admin
Deskripsi	Admin Organisasi dapat melakukan proses verifikasi dan menyetujui terhadap mahasiswa yang mendaftar sebagai panitia acara untuk menentukan kelayakan keikutsertaan.
Kondisi Awal	Admin masuk ke sistem dengan kredensial valid.

Kondisi Akhir	Admin telah menyetujui panitia.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin masuk ke sistem. 2. Admin mengakses halaman kelola panitia. 4. Admin mengklik checkbox dan memilih lebih dari satu panitia, kemudian menekan tombol setuju. 6. Admin mengklik tombol detail. 8. Admin klik tombol setuju.	3. Sistem menampilkan halaman kelola panitia. 5. Sistem memvalidasi dan menyimpan perubahan status. 7. Sistem menampilkan halaman detail. 9. Sistem memvalidasi dan menyimpan perubahan status.
Alur kejadian Alternatif	
Panitia yang status pengajuannya telah disetujui atau ditolak tidak dapat dilakukan proses persetujuan maupun penolakan kembali.	

Scenario use case verifikasi panitia acara pada Tabel 3.7 menggambarkan proses dimana admin dapat menolak panitia yang mengikuti acara setelah melakukan verifikasi data diri pendaftar.

Tabel 3.7 *Use Case Scenario* Menolak Panitia

Nama Use Case	Verifikasi Panitia Acara
Aktor	Admin
Deskripsi	Admin Organisasi dapat melakukan proses verifikasi dan menolak terhadap mahasiswa yang mendaftar sebagai panitia acara untuk menentukan kelayakan keikutsertaan.
Kondisi Awal	Admin masuk ke sistem dengan kredensial valid.

Kondisi Akhir	Admin telah berhasil menolak panitia.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin masuk ke sistem. 2. Admin mengakses halaman kelola panitia. 4. Admin mengklik checkbox dan memilih lebih dari satu panitia, kemudian menekan tombol tolak. 6. Admin mengklik tombol detail. 8. Admin klik tombol tolak.	3. Sistem menampilkan halaman kelola panitia. 5. Sistem memvalidasi dan menyimpan perubahan status. 7. Sistem menampilkan halaman detail. 9. Sistem memvalidasi dan menyimpan perubahan status.
Alur kejadian Alternatif	
Panitia yang status pengajuannya telah disetujui atau ditolak tidak dapat dilakukan proses persetujuan maupun penolakan kembali.	

3.2.4.3 Use Case Scenario Mengelola Hak Akses Panitia

Scenario use case mengelola hak akses panitia pada Tabel 3.8 menggambarkan proses dimana admin dapat mengelola hak akses panitia yang mendaftarkan diri di acara.

Tabel 3.8 *Use Case Scenario* Mengelola Hak Akses Pengguna

Nama Use Case	Mengelola Hak Akses Pengguna
Aktor	Admin
Deskripsi	Admin organisasi dapat mengelola hak akses pengguna dengan menetapkan panitia tertentu sebagai Koordinator Divisi sesuai kebutuhan acara.
Kondisi Awal	Admin mengakses sistem dengan kredensial valid.

Kondisi Akhir	Sistem menyimpan perubahan hak akses.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin mengakses sistem. 2. Admin mengakses halaman kelola divisi. 4. Admin memilih opsi detail. 6. Admin memilih opsi aksi. 7. Admin menekan jadikan koordinator.	3. Halaman kelola divisi ditampilkan oleh sistem. 5. Halaman detail divisi ditampilkan oleh sistem. 8. Sistem menyimpan perubahan.
Alur kejadian Alternatif	
Jika terjadi kesalahan, sistem akan menampilkan pesan kesalahan.	

3.2.4.4 Use Case Scenario Mengelola Sertifikat

Scenario use case mengelola sertifikat pada Tabel 3.9 menggambarkan proses dimana admin dapat melakukan pengelolaan sertifikat panitia dan koordinator.

Tabel 3.9 *Use Case Scenario* Mengelola Hak Akses Pengguna

Nama Use Case	Mengelola Sertifikat
Aktor	Admin
Deskripsi	Admin Organisasi dapat mengelola data sertifikat panitia acara.
Kondisi Awal	Admin masuk ke sistem dengan kredensial valid
Kondisi Akhir	Admin telah mengelola sertifikat
Alur Kejadian Normal	
Aktor	Sistem
1. Admin mengakses sistem	

2. Admin mengakses menu kelola sertifikat dan menekan salah satu acara selesai. 4. Admin membuka kelola template. 6. Admin melakukan pengelolaan template. 7. Admin generate sertifikat untuk peserta dengan kehadiran 75%. 9. Admin menekan publish.	3. Sistem menampilkan daftar peserta. 5. Sistem menampilkan halaman kelola template. 8. Sistem generate sertifikat. 10. Sistem mempublish dan membagikan sertifikat pada panitia yang terdaftar pada acara.
Alur kejadian Alternatif	
1. Panitia atau koordinator divisi yang memiliki persentase kehadiran dibawah 75%, maka sistem tidak dapat melakukan generate sertifikat. 2. Panitia atau koordinator divisi yang memiliki persentase kehadiran dibawah 75% dapat diberikan sertifikat dengan upload manual.	

3.2.4.5 Use Case Scenario Mengelola Divisi

Scenario use case mengelola divisi pada Tabel 3.10 menggambarkan proses dimana admin dapat melakukan penambahan divisi pada acara yang dimiliki.

Tabel 3.10 *Use Case Scenario* Tambah Divisi

Nama Use Case	Mengelola Divisi
Aktor	Admin
Deskripsi	Admin Organisasi dapat menambahkan divisi di setiap acara berstatus tertentu.
Kondisi Awal	Admin masuk ke sistem dengan kredensial valid.
Kondisi Akhir	Admin telah menambahkan divisi
Alur Kejadian Normal	
Aktor	Sistem

1. Admin mengakses sistem. 2. Admin mengakses halaman kelola divisi. 4. Admin memilih acara yang diinginkan. 5. Admin menekan tombol tambah divisi. 7. Admin mengisi inputan dan memilih opsi simpan.	3. Halaman kelola divisi ditampilkan oleh sistem. 6. Dialog tambah divisi ditampilkan oleh sistem. 8. Sistem menyimpan divisi.
Alur kejadian Alternatif	
1. Penambahan divisi yang dilakukan saat acara berstatus selesai, sistem akan menampilkan pesan kesalahan.	

Scenario use case mengelola divisi pada Tabel 3.11 menggambarkan proses dimana admin dapat melakukan perubahan informasi divisi pada acara yang dimiliki.

Tabel 3.11 *Use Case Scenario* Edit Divisi

Nama Use Case	Mengelola Divisi
Aktor	Admin
Deskripsi	Admin Organisasi dapat melakukan perubahan informasi divisi pada setiap acara yang dengan status tertentu.
Kondisi Awal	Admin mengakses sistem dengan kredensial valid.
Kondisi Akhir	Admin telah berhasil mengubah informasi divisi
Alur Kejadian Normal	
Aktor	Sistem
1. Admin mengakses sistem. 2. Admin mengakses halaman kelola divisi.	

4. Admin memilih acara yang diinginkan. 5. Admin menekan tombol edit divisi. 7. Admin mengisi inputan dan menekan tombol simpan.	3. Halaman kelola divisi ditampilkan oleh sistem. 6. Dialog edit divisi ditampilkan oleh sistem. 8. Sistem menyimpan divisi.
Alur kejadian Alternatif	
1. Perubahan informasi divisi yang dilakukan saat acara berstatus selesai, sistem akan menampilkan pesan kesalahan.	

Scenario use case mengelola divisi pada Tabel 3.12 menggambarkan proses dimana admin dapat melakukan penghapusan divisi pada acara yang dimiliki.

Tabel 3.12 *Use Case Scenario* Hapus Divisi

Nama Use Case	Mengelola Divisi
Aktor	Admin
Deskripsi	Admin Organisasi dapat menghapus divisi di setiap acara dengan status tertentu.
Kondisi Awal	Admin masuk ke sistem dengan kredensial valid.
Kondisi Akhir	Admin telah menghapus divisi.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin masuk ke sistem. 2. Admin mengakses halaman kelola divisi. 4. Admin menentukan acara yang diinginkan. 5. Admin memilih opsi hapus divisi.	3. Sistem menampilkan halaman kelola divisi.

	6. Sistem menampilkan validasi apakah admin yakin. 7. Sistem menghapus divisi dari sistem.
Alur kejadian Alternatif	
1. Admin menghapus divisi yang sudah memiliki anggota, sistem akan menolak. 2. Admin menghapus divisi dengan status acara sedang berlangsung atau selesai, pesan kesalahan akan ditampilkan oleh sistem.	

3.2.4.6 Use Case Scenario Mendaftarkan Akun

Scenario use case mendaftarkan akun pada Tabel 3.13 menggambarkan proses dimana panitia dapat melakukan pendaftaran akun untuk mengakses fitur lanjutan.

Tabel 3.13 *Use Case Scenario* Mendaftarkan Akun

Nama Use Case	Mendaftarkan Akun
Aktor	Panitia
Deskripsi	Panitia bisa mendaftarkan akun untuk mengakses fitur lanjutan.
Kondisi Awal	Panitia mengakses Sistem
Kondisi Akhir	Panitia berhasil mendaftar.
Alur Kejadian Normal	
Aktor	Sistem
1. Panitia mengakses fitur registrasi pengguna. 3. Panitia mengisi inputan. 4. Panitia memilih opsi daftar.	2. Halaman registrasi pengguna ditampilkan oleh sistem. 5. Sistem memvalidasi inputan dan menyimpan akun pendaftar.
Alur kejadian Alternatif	
1. Pengguna mengosongkan salah satu inputan, pesan kesalahan ditampilkan. 2. Pengguna mengisi email dengan format salah, pesan kesalahan ditampilkan.	

3. Pengguna mengisi kata sandi dengan format dan jumlah karakter salah, pesan kesalahan ditampilkan.
4. Pengguna mengisi konfirmasi kata sandi tidak persis dengan kata sandi, pesan kesalahan akan ditampilkan sistem.

3.2.4.7 Use Case Scenario Melihat Absensi

Scenario use case melihat absensi pada Tabel 3.14 menggambarkan proses dimana panitia melihat absensi keikutsertaan rapat.

Tabel 3.14 *Use Case Scenario* Melihat Absensi

Nama Use Case	Melihat Absensi
Aktor	Panitia
Deskripsi	Panitia bisa melihat absensi kehadiran dirinya sendiri pada acara yang sedang atau sudah diikuti.
Kondisi Awal	Panitia masuk dengan kredensial valid.
Kondisi Akhir	Panitia berhasil melihat absensi.
Alur Kejadian Normal	
Aktor	Sistem
1. Panitia masuk ke sistem. 2. Panitia mengakses halaman acara saya. 4. Panitia mengakses detail acara. 5. Panitia membuka tab rapat. 6. Panitia membuka detail rapat.	3. Halaman acara ditampilkan sistem. 7. Halaman absensi panitia ditampilkan sistem.
Alur kejadian Alternatif	
Jika terdapat kesalahan, pesan kesalahan akan ditampilkan sistem.	

3.2.4.8 Use Case Scenario Melihat Rapat dan Notulen

Scenario use case melihat rapat dan notulen pada Tabel 3.15 menggambarkan proses dimana panitia melihat rapat dan notulen hasil rapat.

Tabel 3.15 *Use Case Scenario* Melihat Rapat dan Notulen

Nama Use Case	Melihat Rapat dan Notulen
Aktor	Panitia
Deskripsi	Panitia dapat mengakses rapat ditentukan oleh admin atau koordinator divisi, serta melihat notulen dari hasil rapat.
Kondisi Awal	Panitia masuk dengan kredensial valid.
Kondisi Akhir	Panitia berhasil melihat rapat dan notulen
Alur Kejadian Normal	
Aktor	Sistem
1. Panitia masuk ke sistem. 2. Panitia mengakses halaman acara saya. 4. Panitia mengakses detail acara. 5. Panitia membuka tab rapat. 6. Panitia membuka detail rapat.	3. Halaman acara saya ditampilkan sistem. 7. Halaman notulen dan detail rapat ditampilkan sistem.
Alur kejadian Alternatif	
Jika terjadi kesalahan, pesan kesalahan ditampilkan sistem.	

3.2.4.9 Use Case Scenario Mendaftar Acara

Scenario use case mendaftar acara pada Tabel 3.16 menggambarkan proses dimana panitia mendaftarkan diri untuk mengikuti acara kepanitiaan.

Tabel 3.16 *Use Case Scenario* Mendaftar Acara

Nama Use Case	Mendaftar Acara
Aktor	Panitia
Deskripsi	Panitia dapat melakukan pendaftaran acara yang tersedia di dalam aplikasi dan mengisi form pendaftaran.
Kondisi Awal	Panitia masuk ke sistem dengan kredensial valid.
Kondisi Akhir	Panitia berhasil mendaftarkan diri ke acara.
Alur Kejadian Normal	
Aktor	Sistem
1. Panitia masuk ke sistem. 2. Panitia mengakses halaman daftar acara. 4. Panitia mengakses halaman detail acara. 5. Panitia menekan tombol daftar acara. 7. Panitia mengisi formulir daftar acara dan menekan tombol kirim.	3. Sistem menampilkan halaman daftar acara. 6. Sistem menampilkan formulir pendaftaran acara. 8. Sistem memvalidasi apakah inputan sudah benar dan menyimpan data pendaftar.
Alur kejadian Alternatif	
1. Peserta yang telah melakukan pendaftaran pada suatu acara tidak diizinkan mendaftar di acara yang sama. Namun, peserta bisa mendaftar ulang apabila status pendaftaran sebelumnya ditolak.	

3.2.4.10 Use Case Scenario Mengelola Absensi

Scenario use case mengelola absensi pada Tabel 3.17 menggambarkan proses dimana admin dan koordinator dapat melakukan pengelolaan terhadap kehadiran panitia.

Tabel 3.17 *Use Case Scenario* Mengelola Absensi

Nama Use Case	Mengelola Absensi
Aktor	Admin dan Koordinator
Deskripsi	Admin Organisasi dan Koordinator Divisi dapat melakukan pengelolaan absensi panitia, meliputi pencatatan dan pemantauan kehadiran panitia dalam kegiatan acara maupun rapat.
Kondisi Awal	Admin dan koordinator masuk ke sistem dengan kredensial valid.
Kondisi Akhir	Admin dan koordinator berhasil mengelola absensi.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin dan koordinator memulai sesi sistem. 2. Admin dan koordinator mengakses halaman rapat. 3. Admin dan koordinator mengakses menu absensi. 5. Admin dan koordinator melakukan absensi dan memilih opsi simpan.	4. Halaman absensi ditampilkan oleh sistem. 6. Absensi disimpan oleh sistem.
Alur kejadian Alternatif	
1. Jika terjadi kesalahan, sistem akan menampilkan pesan kesalahan. 2. Absensi dikelola di acara yang telah selesai, sistem akan menampilkan pesan kesalahan.	

3.2.4.11 Use Case Scenario Mengelola Rapat dan Notulen

Scenario use case tambah rapat pada Tabel 3.18 menggambarkan proses dimana admin dan koordinator dapat melakukan penambahan rapat.

Tabel 3.18 Use Case Scenario Tambah Rapat

Nama Use Case	Mengelola Rapat dan Notulen
Aktor	Admin dan Koordinator
Deskripsi	Admin Organisasi dan Koordinator Divisi dapat melakukan penambahan rapat baru.
Kondisi Awal	Admin dan koordinator masuk ke sistem dengan kredensial valid.
Kondisi Akhir	Admin dan koordinator berhasil menambahkan rapat baru.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin dan Koordinator memulai sesi sistem. 2. Admin dan Koordinator mengakses halaman rapat. 4. Admin dan Koordinator memilih acara. 5. Admin dan Koordinator memilih opsi tambah rapat. 7. Admin dan Koordinator mengisi dan memilih opsi simpan.	3. Sistem menampilkan daftar rapat. 6. Dialog tambah rapat ditampilkan sistem. 8. Rapat divalidasi dan disimpan oleh sistem.
Alur kejadian Alternatif	
Mengisi inputan dengan salah, pesan kesalahan ditampilkan sistem.	

Scenario use case edit rapat pada Tabel 3.19 menggambarkan proses dimana admin dan koordinator dapat melakukan perubahan informasi rapat.

Tabel 3.19 *Use Case Scenario* Edit Rapat

Nama Use Case	Mengelola Rapat dan Notulen
Aktor	Admin dan Koordinator
Deskripsi	Admin Organisasi dan Koordinator Divisi dapat melakukan perubahan informasi rapat.
Kondisi Awal	Admin dan koordinator masuk ke sistem dengan kredensial valid.
Kondisi Akhir	Admin dan koordinator berhasil mengubah informasi rapat.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin dan Koordinator memulai sesi. 2. Admin dan Koordinator mengakses halaman rapat. 4. Admin dan Koordinator memilih acara. 5. Admin dan Koordinator menekan tombol edit rapat. 7. Admin dan Koordinator mengisi dan memilih opsi simpan.	3. Sistem menampilkan daftar rapat. 6. Sistem menampilkan dialog edit rapat. 8. Rapat divalidasi dan disimpan oleh sistem.
Alur kejadian Alternatif	
Mengisi inputan dengan salah, pesan kesalahan ditampilkan sistem.	

Scenario use case hapus rapat pada Tabel 3.20 menggambarkan proses dimana admin dan koordinator dapat melakukan penghapusan rapat.

Tabel 3.20 *Use Case Scenario* Hapus Rapat

Nama Use Case	Mengelola Rapat dan Notulen
Aktor	Admin dan Koordinator
Deskripsi	Admin Organisasi dan Koordinator Divisi dapat melakukan penghapusan rapat.
Kondisi Awal	Admin dan koordinator masuk ke sistem dengan kredensial valid.
Kondisi Akhir	Admin dan koordinator berhasil menghapus rapat.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin dan Koordinator memulai sesi sistem. 2. Admin dan Koordinator mengakses halaman rapat. 4. Admin dan Koordinator memilih acara. 5. Admin dan Koordinator menekan tombol hapus rapat.	3. Sistem menampilkan daftar rapat. 6. Sistem berhasil menghapus rapat.
Alur kejadian Alternatif	
Menghapus rapat yang telah selesai atau sedang dilaksanakan, sistem akan menampilkan pesan kesalahan.	

Scenario use case pengelolaan notulen pada Tabel 3.21 menggambarkan proses dimana admin dan koordinator dapat melakukan pengelolaan notulen.

Tabel 3.21 *Use Case Scenario* Notulen Rapat

Nama Use Case	Mengelola Rapat dan Notulen
Aktor	Admin dan Koordinator

Deskripsi	Admin Organisasi dan Koordinator Divisi dapat melakukan pengelolaan notulen hasil rapat.
Kondisi Awal	Admin dan koordinator masuk ke sistem dengan kredensial valid.
Kondisi Akhir	Admin dan koordinator berhasil mengelola notulen hasil rapat.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin dan Koordinator memulai sesi sistem. 2. Admin dan Koordinator mengakses halaman rapat. 4. Admin dan Koordinator memilih acara. 5. Admin dan Koordinator memilih opsi notulen. 7. Admin dan Koordinator mengisi dan memilih opsi simpan.	3. Sistem menampilkan daftar rapat. 6. Halaman notulen ditampilkan sistem. 8. Notulen divalidasi dan disimpan oleh sistem.
Alur kejadian Alternatif	
Mengisi notulen hasil rapat dengan status rapat selain selesai, pesan kesalahan ditampilkan sistem.	

3.2.4.12 Use Case Scenario Mendapatkan Sertifikat

Scenario use case mendapatkan sertifikat pada Tabel 3.22 menggambarkan proses dimana admin dan koordinator dapat mengunduh sertifikat acara yang telah mereka ikuti.

Tabel 3.22 *Use Case Scenario* Mendapatkan Sertifikat

Nama Use Case	Mendapatkan Sertifikat
Aktor	Panitia dan Koordinator
Deskripsi	Panitia dan koordinator divisi dapat

	melakukan pengunduhan sertifikat jika sudah diterbitkan/sudah menyelesaikan kepanitiaan pada acara yang diikuti.
Kondisi Awal	Panitia dan koordinator masuk ke sistem dengan kredensial valid.
Kondisi Akhir	Panitia dan koordinator berhasil mendapatkan sertifikat.
Alur Kejadian Normal	
Aktor	Sistem
1. Koordinator divisi dan panitia masuk ke sistem. 2. Koordinator divisi dan panitia mengakses halaman acara saya. 4. Koordinator divisi dan panitia mengakses halaman detail acara. 5. Koordinator divisi dan panitia membuka tab sertifikat.	3. Halaman acara saya ditampilkan sistem. 6. Sertifikat ditampilkan sistem.
Alur kejadian Alternatif	
Jika proses mengalami kegagalan, pesan kesalahan ditampilkan sistem.	

3.2.4.13 Use Case Scenario Masuk ke Sistem

Scenario use case masuk ke sistem pada Tabel 3.23 menggambarkan proses dimana admin, koordinator dan panitia dapat mengakses sistem sesuai dengan izin akses yang dimiliki.

Tabel 3.23 *Use Case Scenario* Masuk ke Sistem

Nama Use Case	Masuk ke Sistem
Aktor	Admin, Koordinator Divisi, dan Panitia
Deskripsi	Sistem menyediakan fitur autentikasi bagi pengguna yang telah terdaftar (Admin Organisasi, Koordinator Divisi, dan Panitia) untuk mengakses sistem sesuai dengan batasan akses yang dimiliki.

Kondisi Awal	Mengakses <i>page</i> masuk.
Kondisi Akhir	Berhasil memasuki sistem.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin, panitia dan koordinator divisi mengakses halaman masuk. 3. Admin, panitia dan koordinator divisi mengisi inputan. 4. Admin, panitia dan koordinator divisi memilih opsi masuk.	2. Halaman masuk ditampilkan oleh sistem. 5. Sistem memvalidasi inputan dan mengarahkan ke halaman beranda masing-masing.
Alur kejadian Alternatif	
1. Nama pengguna dan kata sandi salah, pesan kesalahan ditampilkan sistem. 2. Mengosongkan salah satu inputan, pesan kesalahan ditampilkan sistem.	

3.2.4.14 Use Case Scenario Menerima Notifikasi

Scenario use case menerima notifikasi pada Tabel 3.24 menggambarkan proses dimana admin, koordinator dan panitia dapat menerima notifikasi dari sistem.

Tabel 3.24 *Use Case Scenario* Menerima Notifikasi

Nama Use Case	Menerima Notifikasi
Aktor	Admin, Koordinator Divisi, dan Panitia
Deskripsi	Admin, koordinator divisi dan panitia dapat menerima notifikasi dari sistem.
Kondisi Awal	Pengguna memasuki sistem dengan kredensial valid.
Kondisi Akhir	Notifikasi dari sistem berhasil diterima oleh pengguna.
Alur Kejadian Normal	

Aktor	Sistem
1. Admin, Koordinator Divisi, dan Panitia masuk ke sistem 2. Admin, Koordinator Divisi, dan Panitia membuka halaman notifikasi.	3. Sistem menampilkan notifikasi.
Alur kejadian Alternatif	
Jika terjadi kesalahan, pesan kesalahan ditampilkan sistem.	

3.2.4.15 Use Case Scenario Mengelola Profil

Scenario use case mengelola profil pada Tabel 3.25 menggambarkan proses dimana admin, koordinator dan panitia mengelola profil masing-masing.

Tabel 3.25 *Use Case Scenario* Mengelola Profil

Nama Use Case	Mengelola Profil
Aktor	Admin, Koordinator Divisi, dan Panitia
Deskripsi	Admin, koordinator divisi dan panitia dapat mengelola profil masing-masing.
Kondisi Awal	Masuk ke sistem dengan kredensial valid.
Kondisi Akhir	Berhasil memperbarui profil.
Alur Kejadian Normal	
Aktor	Sistem
1. Pengguna masuk ke sistem. 2. Halaman profil diakses pengguna. 4. Pengguna memilih opsi edit. 5. Pengguna mengisi profil dan memilih opsi simpan.	3. Profil pengguna ditampilkan oleh sistem. 6. Informasi perubahan profil pengguna disimpan oleh sistem.

Alur kejadian Alternatif
Jika terjadi kesalahan, sistem akan menampilkan pesan kesalahan.

3.2.4.16 Use Case Scenario Mengubah Kata Sandi

Scenario use case mengubah kata sandi pada Tabel 3.26 menggambarkan proses dimana admin, koordinator dan panitia dapat mengubah kata sandi yang mereka miliki.

Tabel 3.26 Use Case Scenario Mengubah Kata Sandi

Nama Use Case	Mengubah kata sandi
Aktor	Admin, Koordinator Divisi, dan Panitia
Deskripsi	Admin, koordinator divisi dan panitia dapat mengubah kata sandi dengan menginputkan email yang valid.
Kondisi Awal	Admin, Koordinator Divisi, dan Panitia masuk ke sistem dengan kredensial valid.
Kondisi Akhir	Admin, Koordinator Divisi, dan Panitia berhasil mengubah kata sandi.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin, Koordinator Divisi, dan Panitia mengakses halaman lupa kata sandi. 3. Admin, Koordinator Divisi, dan Panitia menginputkan email valid. 5. Admin, Koordinator Divisi, dan Panitia mengakses email dan mengubah kata sandi.	2. Sistem menampilkan halaman lupa kata sandi. 4. Sistem mengirim verifikasi melalui email. 6. Sistem menyimpan perubahan kata sandi
Alur kejadian Alternatif	
1. Pengguna menginputkan email dengan format salah, sistem memberikan pesan	

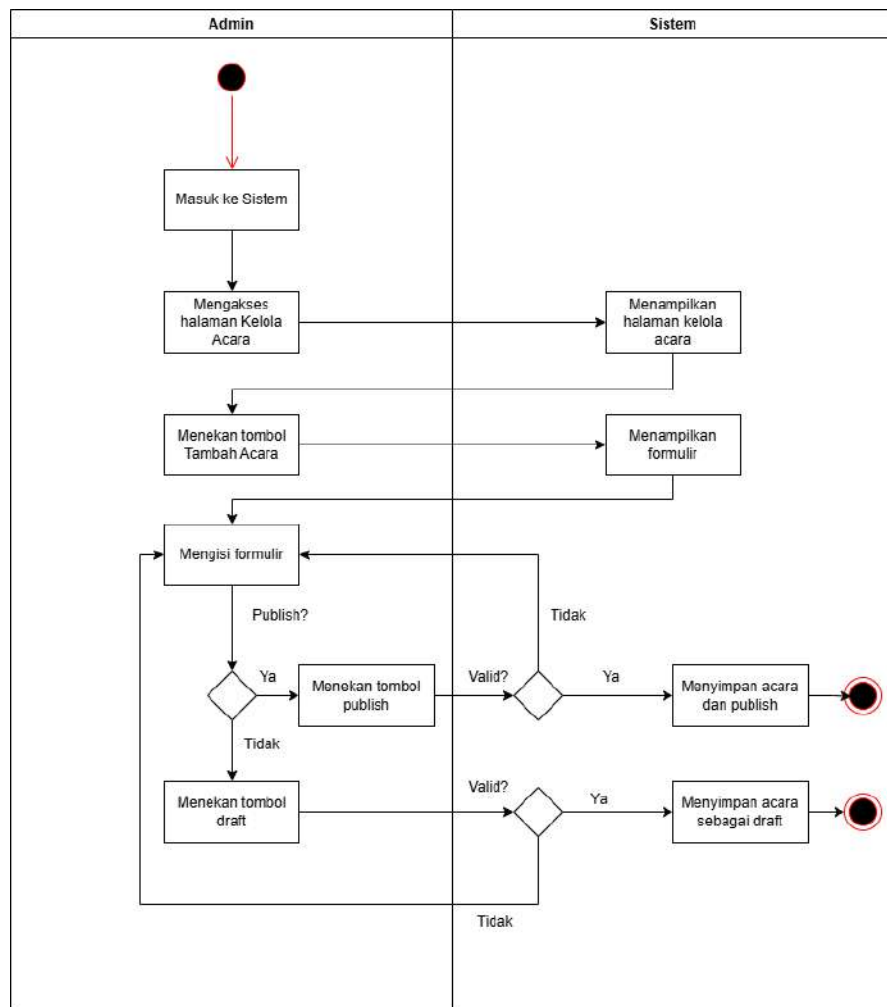
kesalahan.

2. Pengguna menginputkan email salah, sistem memberikan pesan kesalahan.
3. Pengguna mengubah kata sandi dengan format dan jumlah karakter salah, pesan kesalahan ditampilkan sistem.
4. Pengguna mengisi konfirmasi kata sandi tidak persis dengan kata sandi, pesan kesalahan ditampilkan sistem.

3.2.5 Activity diagram

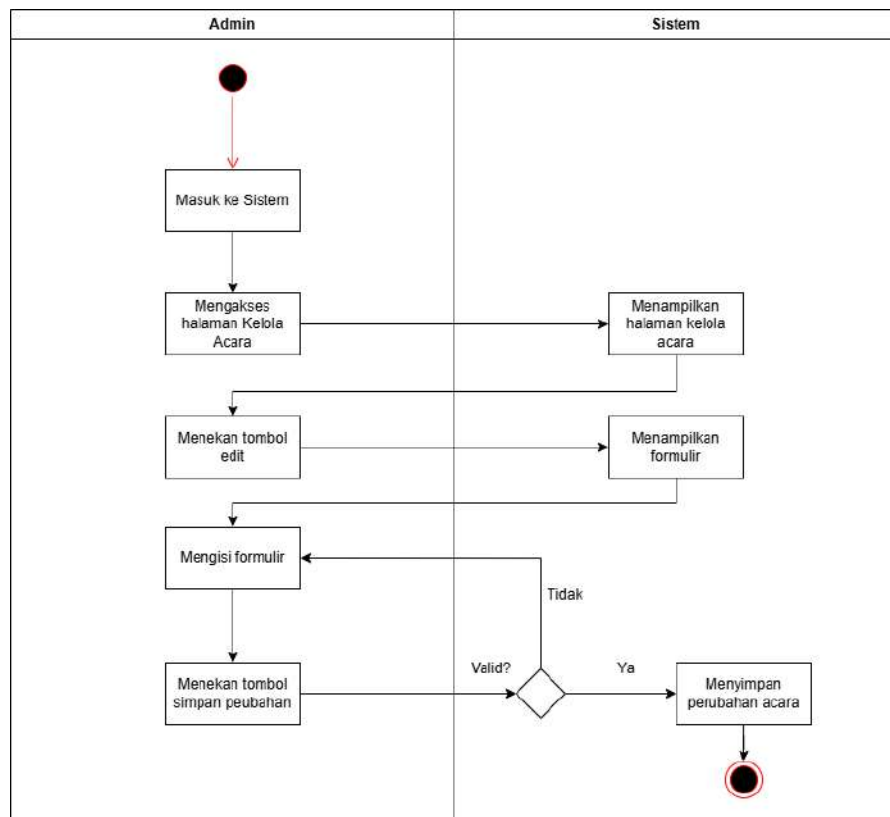
3.2.5.1 Activity Diagram Mengelola Acara

Pada Gambar 3.4 dapat dilihat *activity diagram* yang menggambarkan alur proses pembuatan acara yang ditangani admin organisasi. Proses berawal dari admin masuk ke sistem dengan kredensial valid. Sampai sistem menyimpan acara sebagai *draft* atau *publish*.



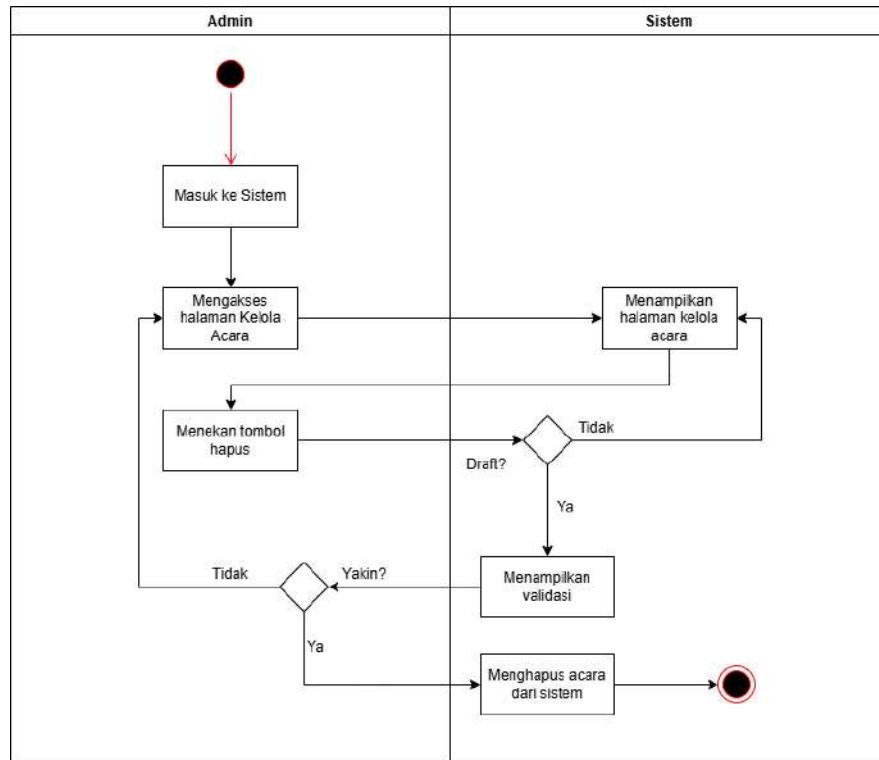
Gambar 3.4 Activity Diagram Tambah Acara

Pada Gambar 3.5 dapat dilihat *activity diagram* yang menggambarkan alur proses pengubahan informasi acara yang ditangani admin organisasi. Proses berawal dari admin masuk ke sistem dengan kredensial valid sampai sistem berhasil menyimpan perubahan informasi.



Gambar 3.5 *Activity Diagram* Tambah Acara

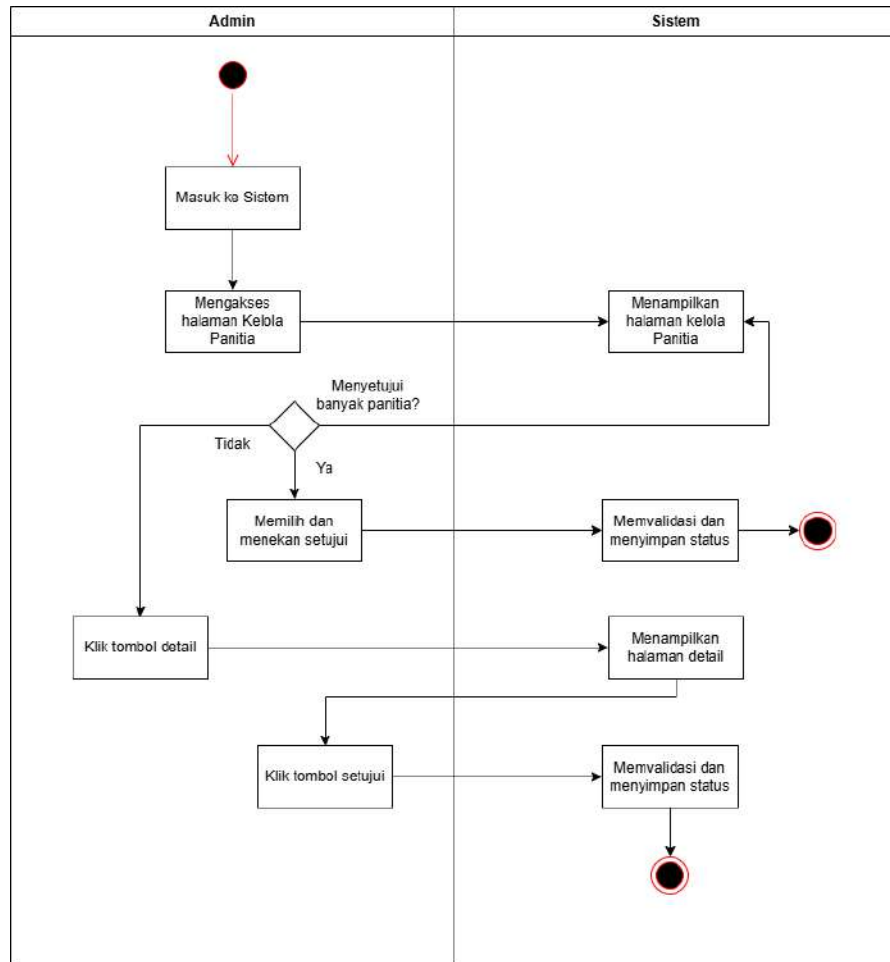
Pada Gambar 3.6 dapat dilihat *activity diagram* yang menggambarkan alur proses pembuatan acara yang ditangani admin organisasi. Proses berawal dari admin masuk ke sistem dengan kredensial valid sampai sistem berhasil menghapus acara.



Gambar 3.6 *Activity Diagram* Hapus Acara

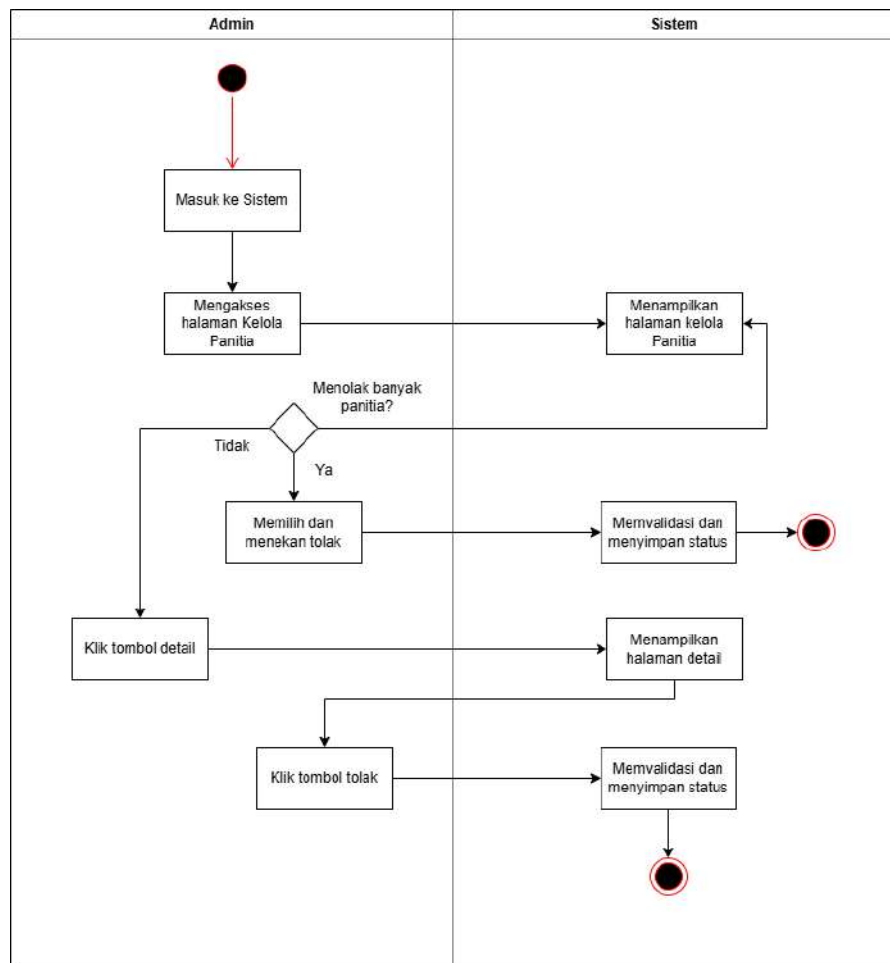
3.2.5.2 *Activity Diagram* Verifikasi Panitia Acara

Pada Gambar 3.7 dapat dilihat *activity diagram* yang menjelaskan alur proses verifikasi persetujuan yang dilakukan admin. Proses diawali dengan admin masuk ke sistem dengan kredensial valid sampai sistem berhasil menyimpan informasi persetujuan peserta.



Gambar 3.7 Activity Diagram Menyetujui Panitia

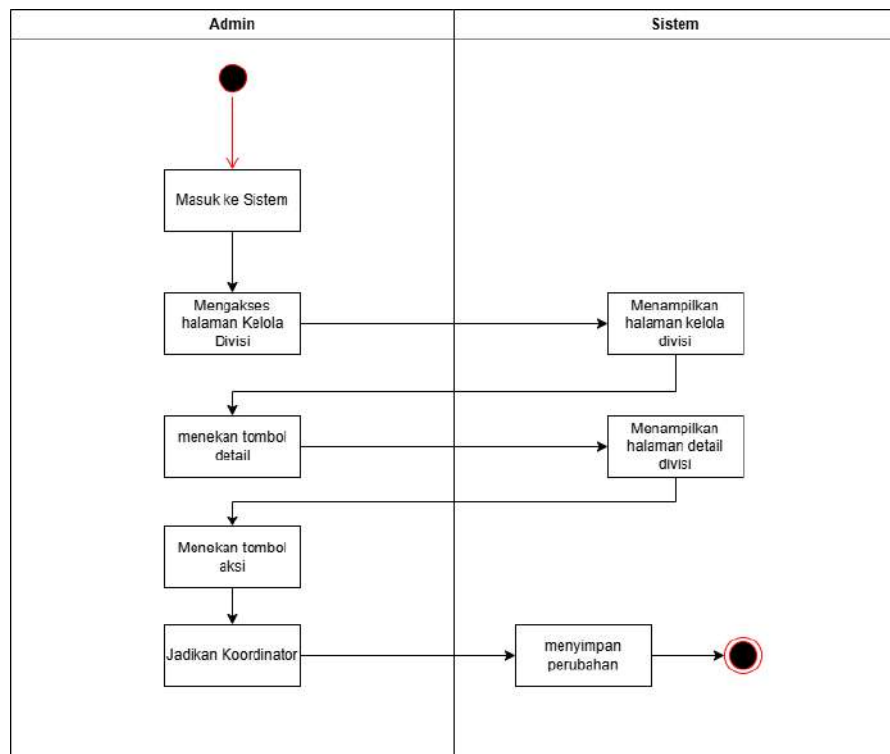
Pada Gambar 3.8 dapat dilihat *activity diagram* yang menggambarkan alur proses verifikasi penolakan ditangani admin. Proses berawal dari admin masuk ke sistem dengan kredensial valid sampai sistem berhasil menyimpan informasi penolakan peserta.



Gambar 3.8 *Activity Diagram* Menolak Panitia

3.2.5.3 *Activity Diagram* Mengelola Hak Akses Panitia

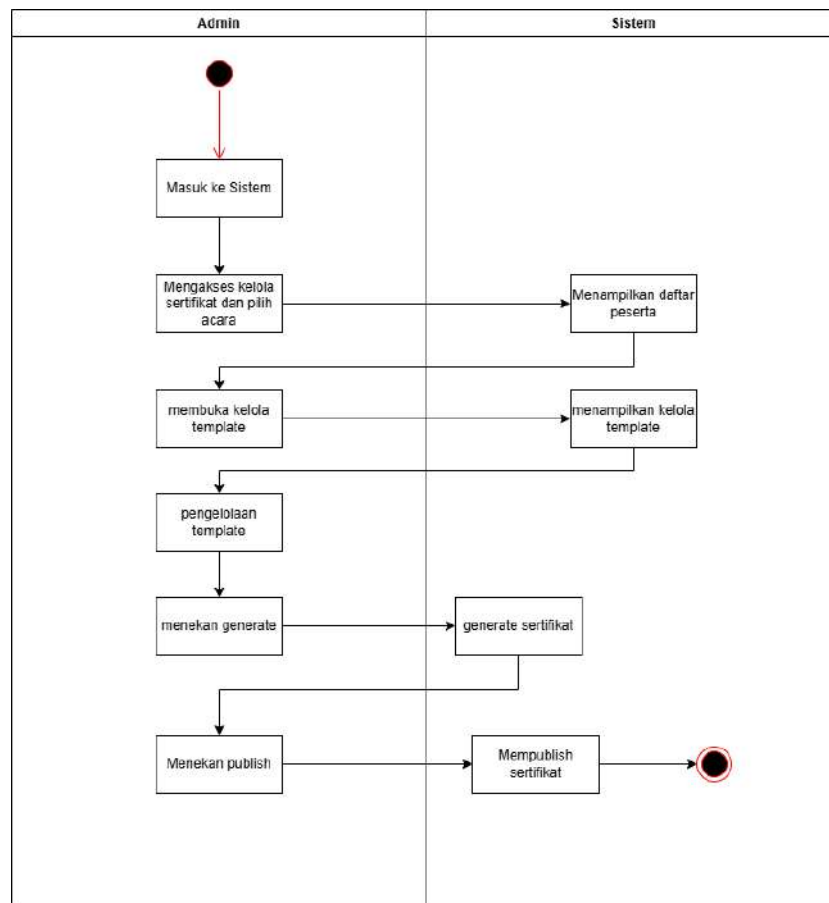
Pada Gambar 3.9 dapat dilihat *activity diagram* yang menggambarkan alur proses mengelola hak akses panitia dimulai dari admin masuk ke sistem dengan kredensial valid sampai admin berhasil mengelola hak akses panitia.



Gambar 3.9 *Activity Diagram* Menolak Panitia

3.2.5.4 *Activity Diagram* Mengelola Sertifikat

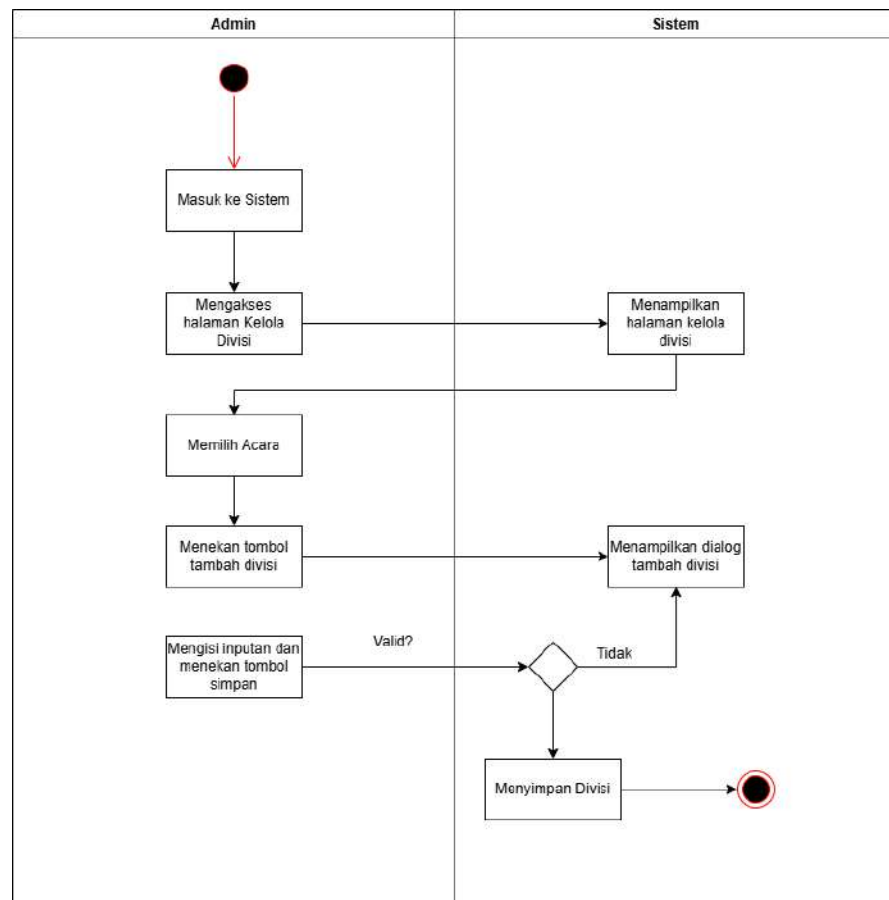
Pada Gambar 3.10 dapat dilihat *activity diagram* yang menggambarkan alur proses mengelola sertifikat dimulai dari admin masuk ke sistem dengan kredensial valid sampai admin berhasil mengelola sertifikat.



Gambar 3.10 *Activity Diagram* Mengelola Sertifikat

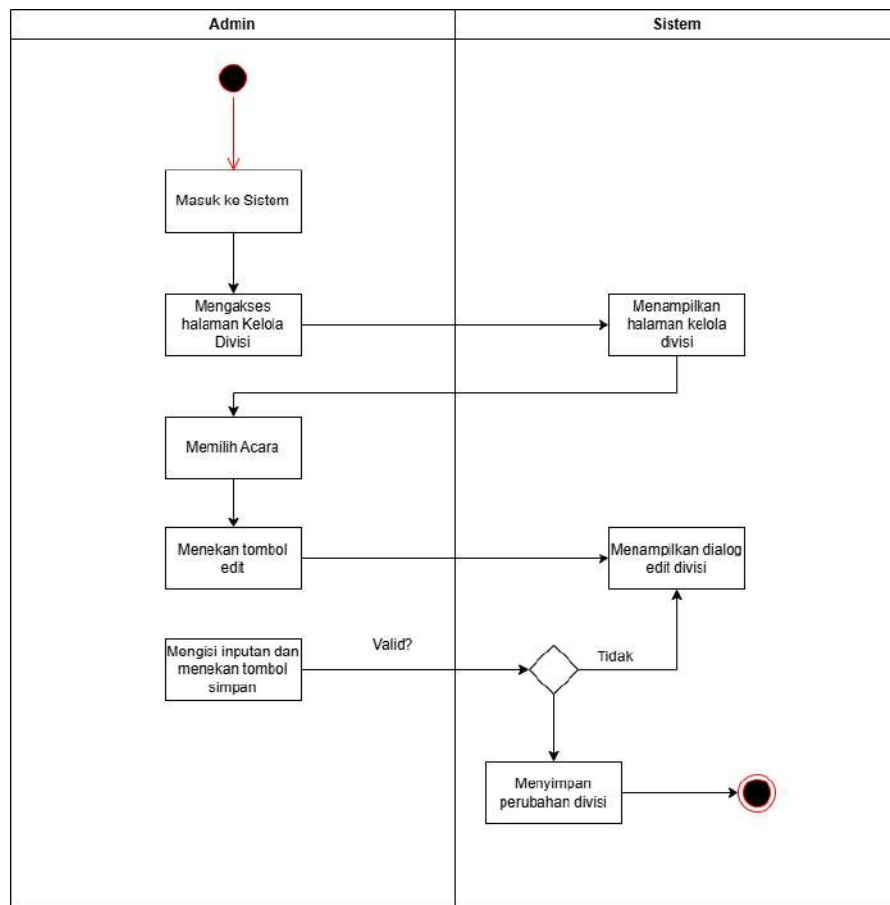
3.2.5.5 *Activity Diagram* Mengelola Divisi

Pada Gambar 3.11 dapat dilihat *activity diagram* yang menggambarkan alur proses tambah divisi pada acara yang dimiliki, dimulai dari admin masuk dengan kredensial valid sampai admin berhasil menambah divisi.



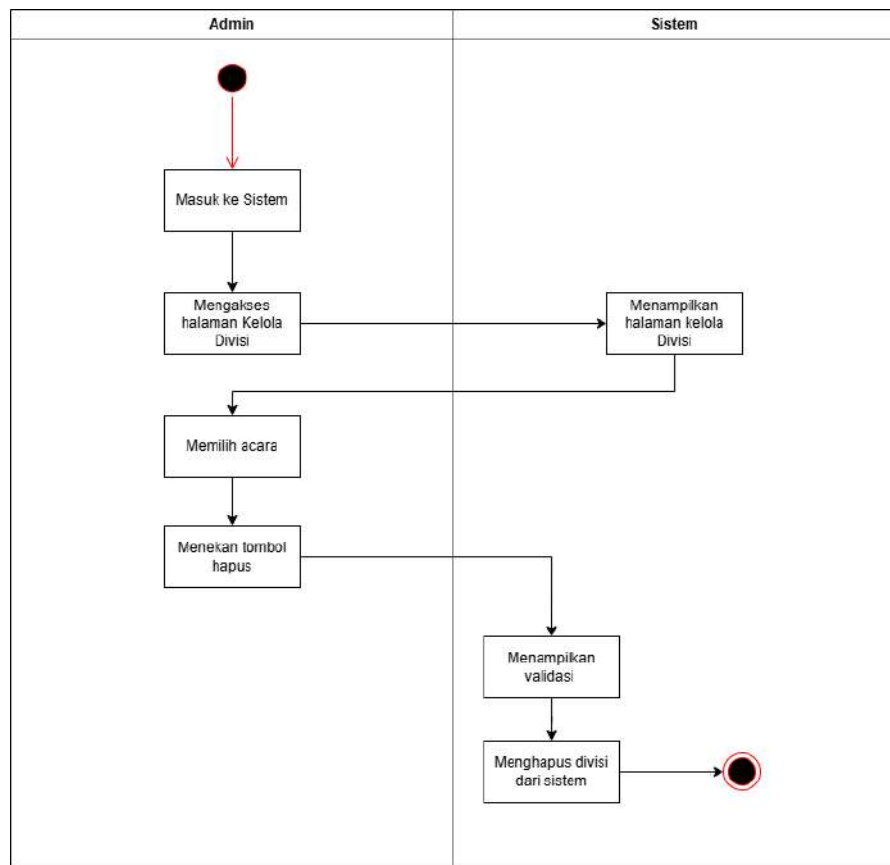
Gambar 3.11 *Activity Diagram* Tambah Divisi

Pada Gambar 3.12 dapat dilihat *activity diagram* yang menggambarkan alur proses edit divisi pada acara yang dimiliki, dimulai dari admin masuk dengan kredensial valid sampai admin berhasil melakukan perubahan divisi.



Gambar 3.12 *Activity Diagram* Edit Divisi

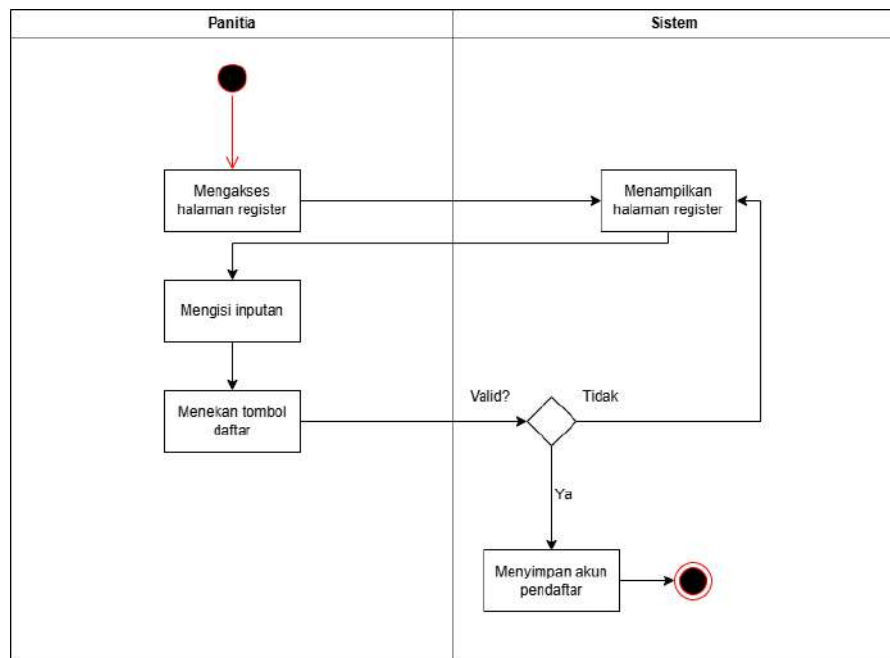
Pada Gambar 3.13 dapat dilihat *activity diagram* yang menggambarkan alur proses hapus divisi pada acara yang dimiliki, dimulai admin masuk dengan kredensial valid sampai admin berhasil menghapus divisi.



Gambar 3.13 *Activity Diagram* Hapus Divisi

3.2.5.6 *Activity Diagram* Mendaftarkan Akun

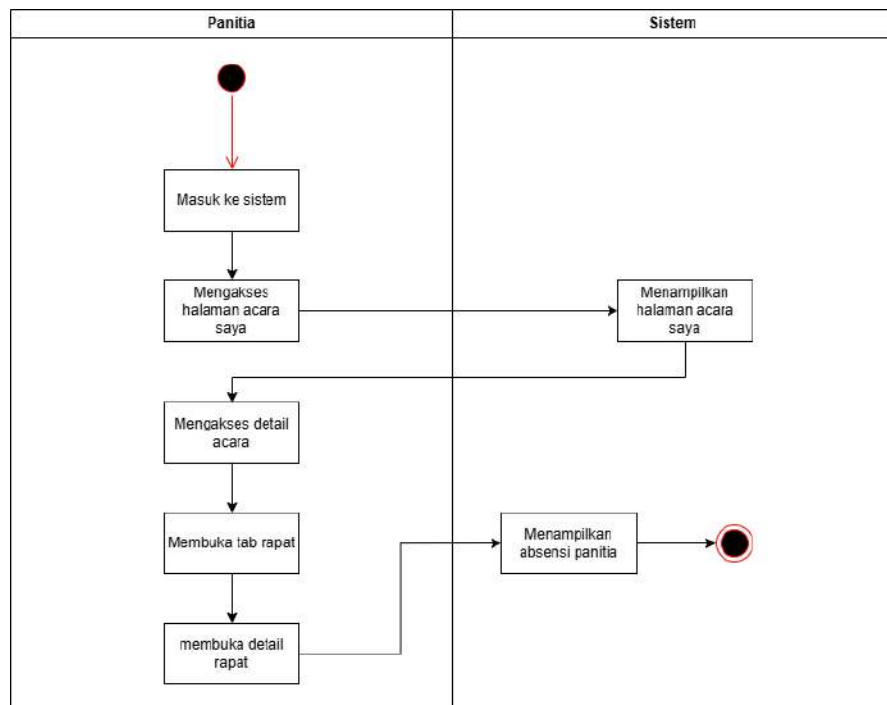
Pada Gambar 3.14 dapat dilihat *activity diagram* yang menjelaskan alur proses pendaftaran akun. Dimulai dari panitia mengakses halaman pendaftaran sampai sistem berhasil menyimpan informasi panitia.



Gambar 3.14 *Activity Diagram* Mendaftarkan Akun

3.2.5.7 *Activity Diagram* Melihat Absensi

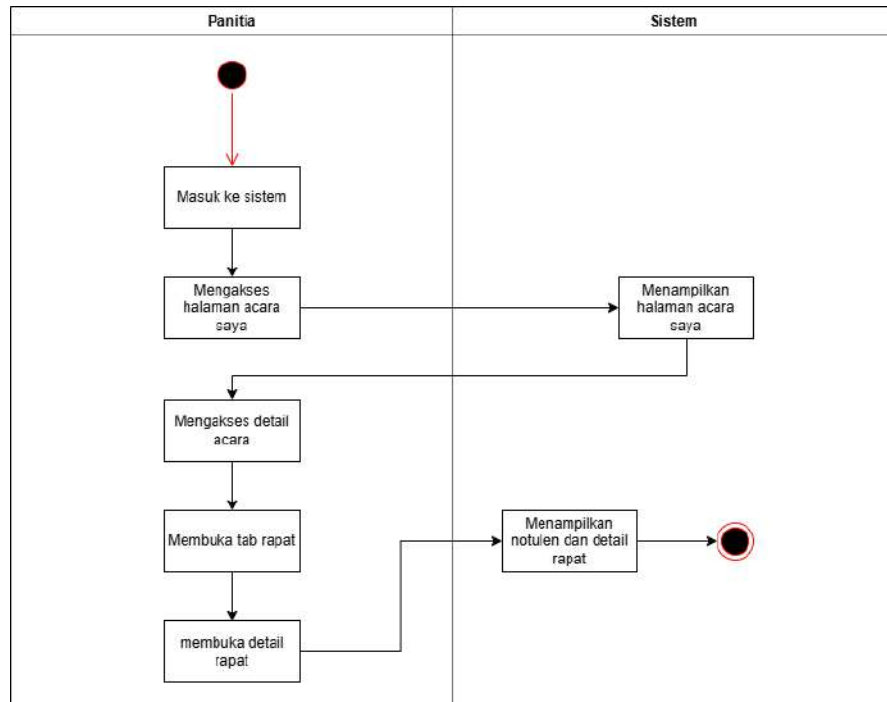
Pada Gambar 3.15 dapat dilihat *activity diagram* yang menggambarkan alur proses melihat absensi. Panitia mengakses halaman saya, mengakses detail acara, membuka tab rapat dan membuka detail rapat. Sistem menampilkan absensi panitia.



Gambar 3.15 *Activity Diagram* Melihat Absensi

3.2.5.8 *Activity Diagram* Melihat Rapat dan Notulen

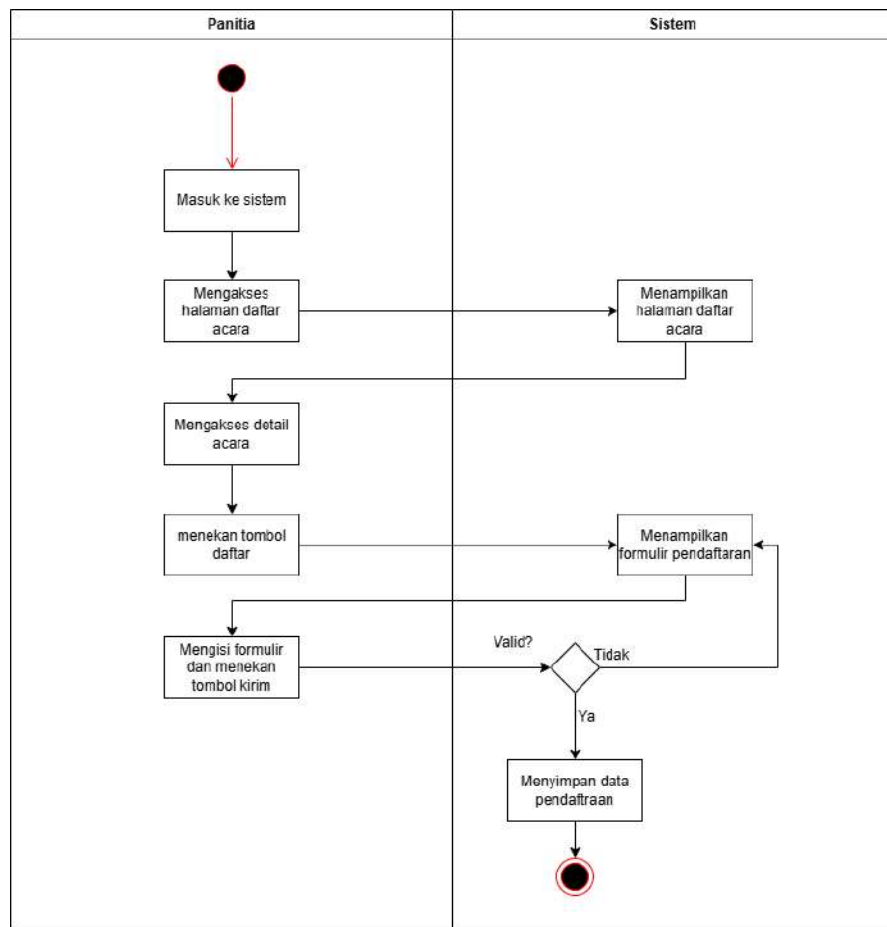
Pada Gambar 3.16 dapat dilihat *activity diagram* yang menggambarkan alur proses melihat absensi. Panitia mengakses halaman saya, mengakses detail acara, membuka tab rapat dan membuka detail rapat. Sistem akan menampilkan notulen dan detail rapat.



Gambar 3.16 *Activity Diagram* Melihat Rapat dan Notulen

3.2.5.9 *Activity Diagram* Mendaftar Acara

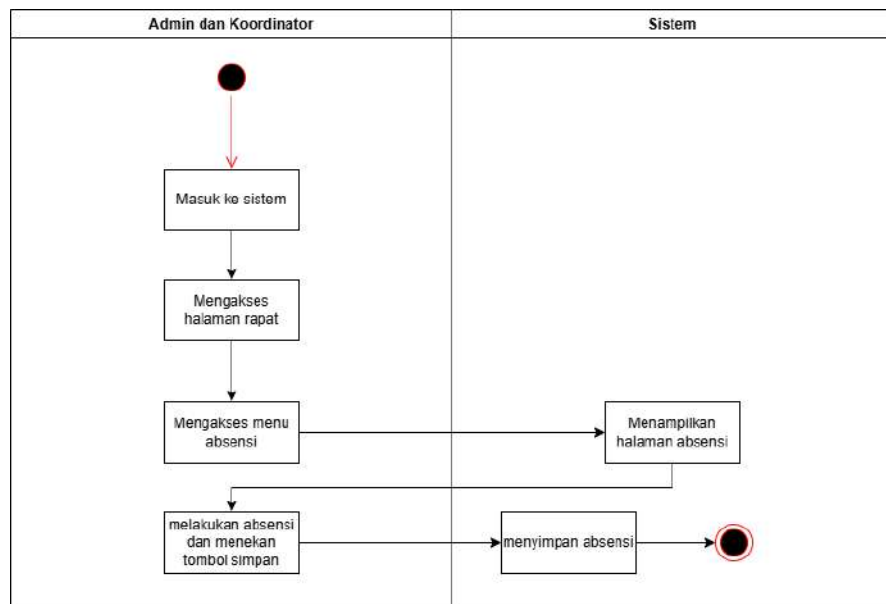
Pada Gambar 3.17 dapat dilihat *activity diagram* yang menggambarkan alur proses mendaftar acara. Dimulai dari panitia masuk ke sistem dengan kredensial valid sampai panitia berhasil mendaftarkan diri ke acara.



Gambar 3.17 *Activity Diagram* Mendaftar Acara

3.2.5.10 *Activity Diagram* Mengelola Absensi

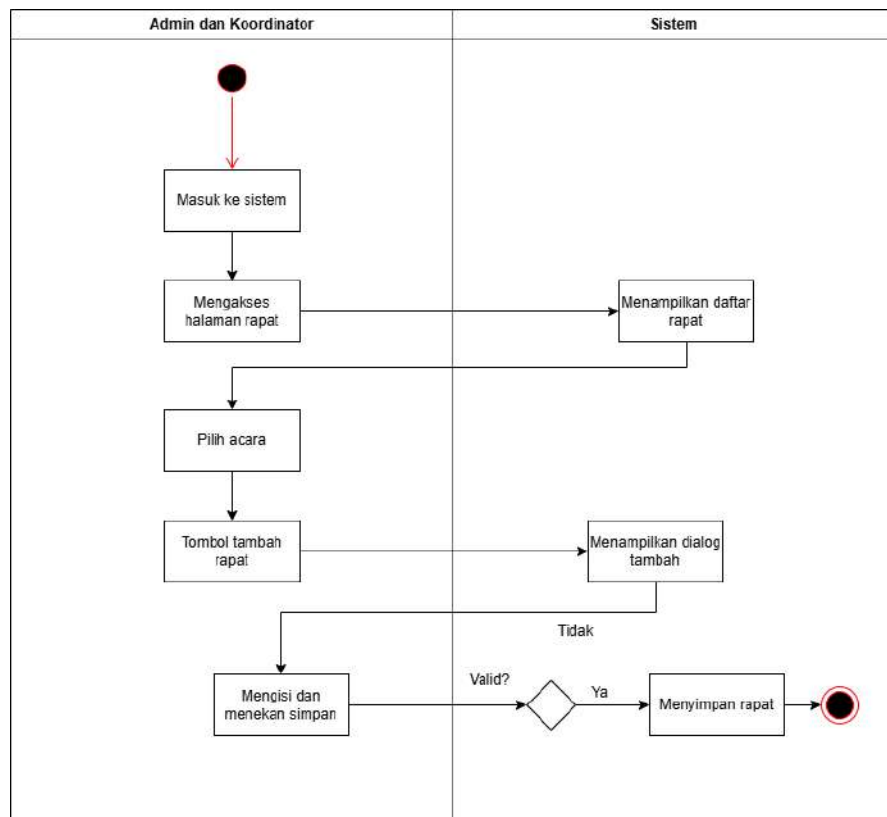
Pada Gambar 3.18 dapat dilihat *activity diagram* yang menggambarkan alur proses mengelola absensi. Dimulai dari admin dan koordinator divisi masuk ke sistem sampai berhasil mengelola absensi.



Gambar 3.18 *Activity Diagram* Mengelola Absensi

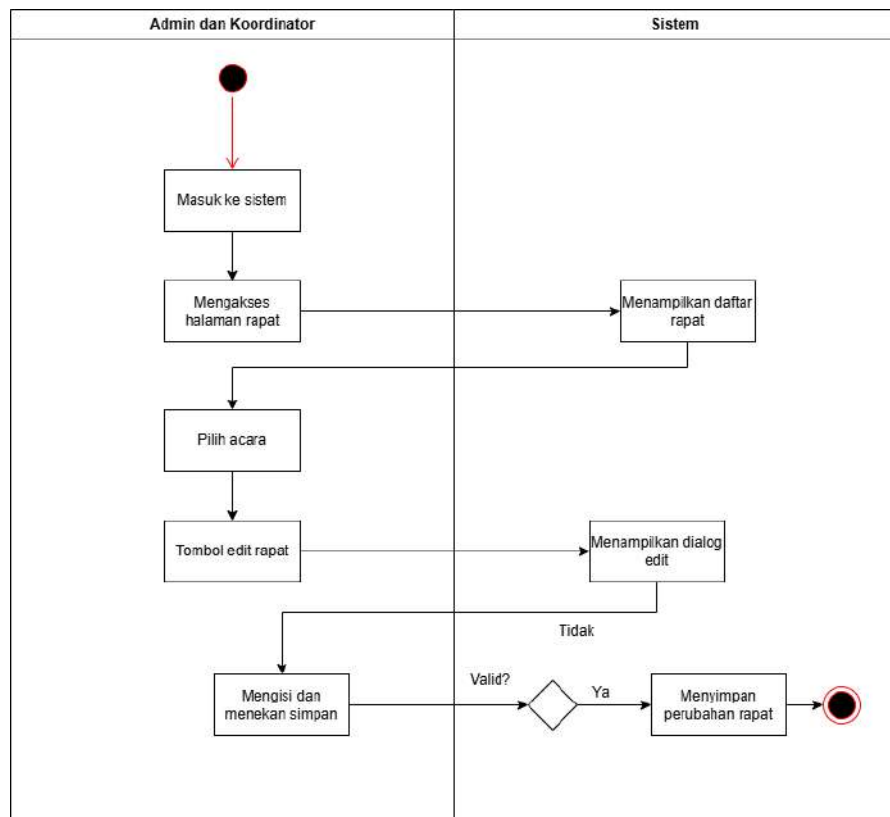
3.2.5.11 *Activity Diagram* Mengelola Rapat dan Notulen

Pada Gambar 3.19 dapat dilihat *activity diagram* yang menggambarkan alur proses tambah rapat. Dimulai dari admin dan koordinator masuk ke sistem sampai berhasil menambah rapat.



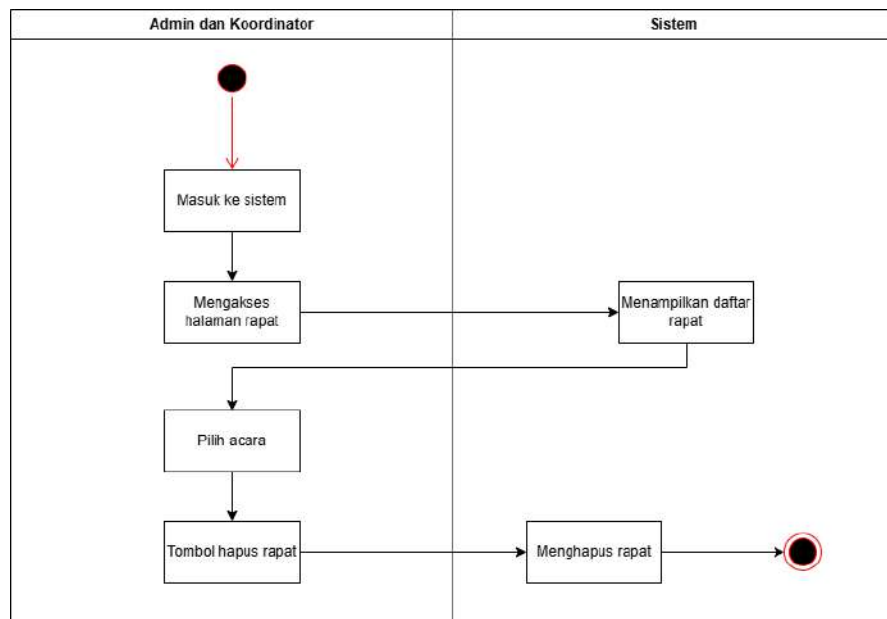
Gambar 3.19 *Activity Diagram* Tambah Rapat

Pada Gambar 3.20 dapat dilihat *activity diagram* yang menggambarkan alur proses edit rapat. Dimulai dari admin dan koordinator masuk ke sistem sampai berhasil mengubah rapat.



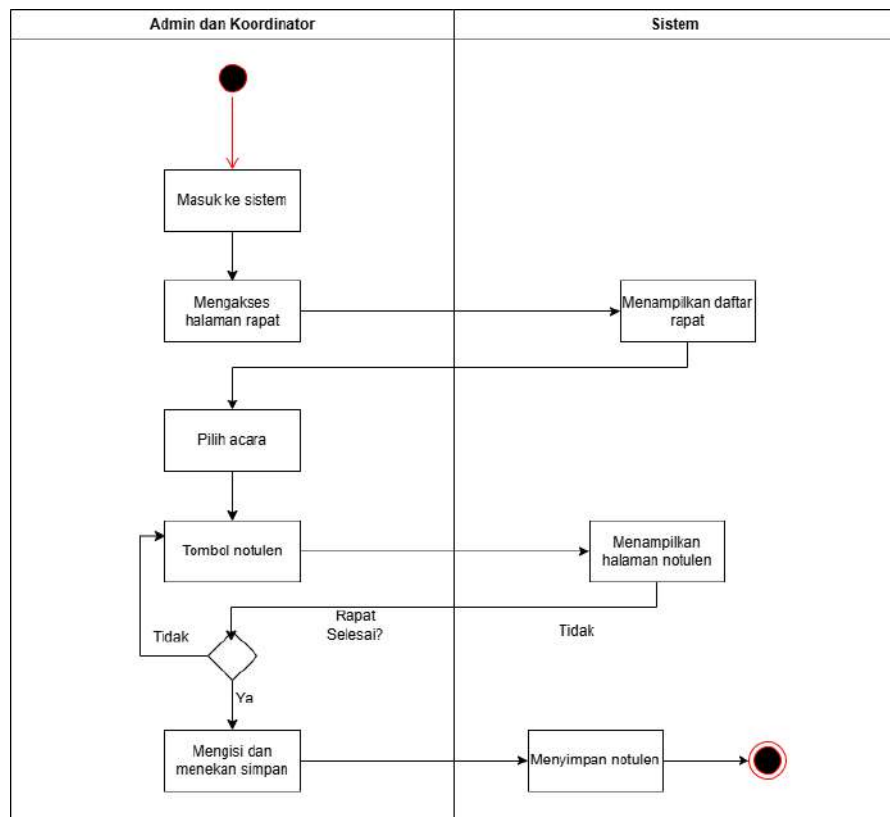
Gambar 3.20 *Activity Diagram* Edit Rapat

Pada Gambar 3.21 dapat dilihat *activity diagram* yang menggambarkan alur proses hapus rapat. Dimulai dari admin dan koordinator masuk ke sistem sampai berhasil menghapus rapat.



Gambar 3.21 *Activity Diagram* Hapus Rapat

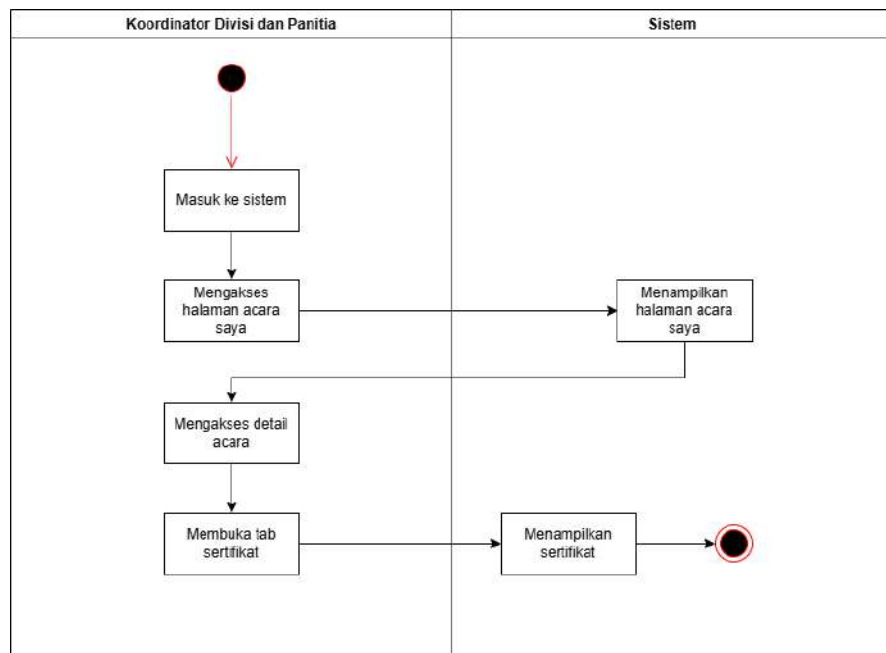
Pada Gambar 3.22 dapat dilihat *activity diagram* yang menggambarkan alur proses pengelolaan notulen. Dimulai dari admin dan koordinator masuk ke sistem sampai berhasil mengelola notulen.



Gambar 3.22 *Activity Diagram* Kelola Notulen

3.2.5.12 *Activity Diagram* Mendapatkan Sertifikat

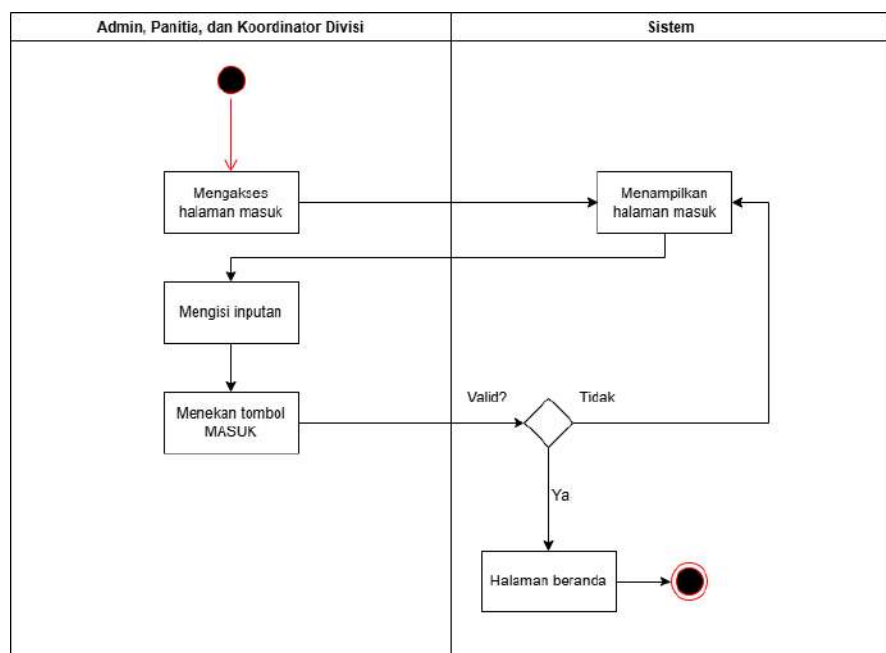
Pada Gambar 3.23 dapat dilihat *activity diagram* yang menggambarkan alur proses mendapatkan sertifikat. Dimulai dari admin dan koordinator masuk ke sistem sampai berhasil mengunduh sertifikat.



Gambar 3.23 *Activity Diagram* Mendapatkan Sertifikat

3.2.5.13 *Activity Diagram* Masuk ke Sistem

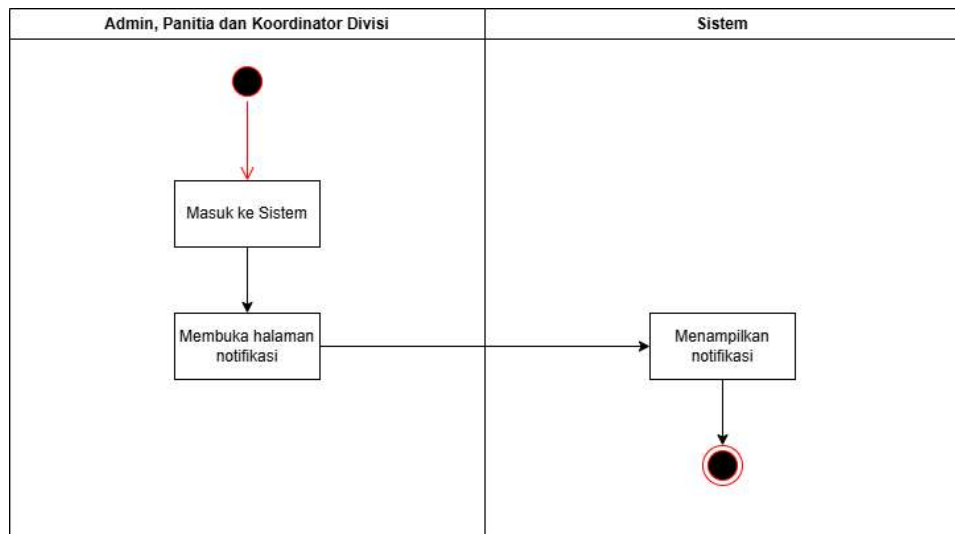
Pada Gambar 3.24 dapat dilihat *activity diagram* yang menggambarkan alur proses masuk ke sistem. Dimulai dari pengguna (Admin, panitia dan koordinator) mengakses halaman masuk sampai berhasil memasuki sistem.



Gambar 3.24 *Activity Diagram* Masuk ke Sistem

3.2.5.14 *Activity Diagram* Menerima Notifikasi

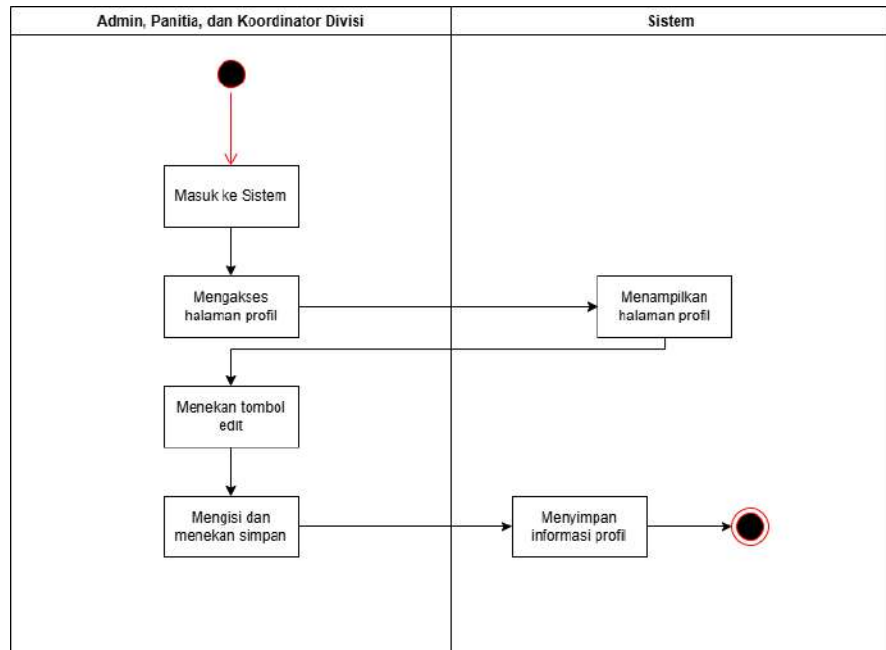
Pada Gambar 3.25 dapat dilihat *activity diagram* yang menggambarkan alur proses menerima notifikasi. Dimulai dari pengguna (Admin, panitia dan koordinator) masuk ke sistem sampai berhasil melihat notifikasi.



Gambar 3.25 *Activity Diagram* Menerima Notifikasi

3.2.5.15 *Activity Diagram* Mengelola Profil

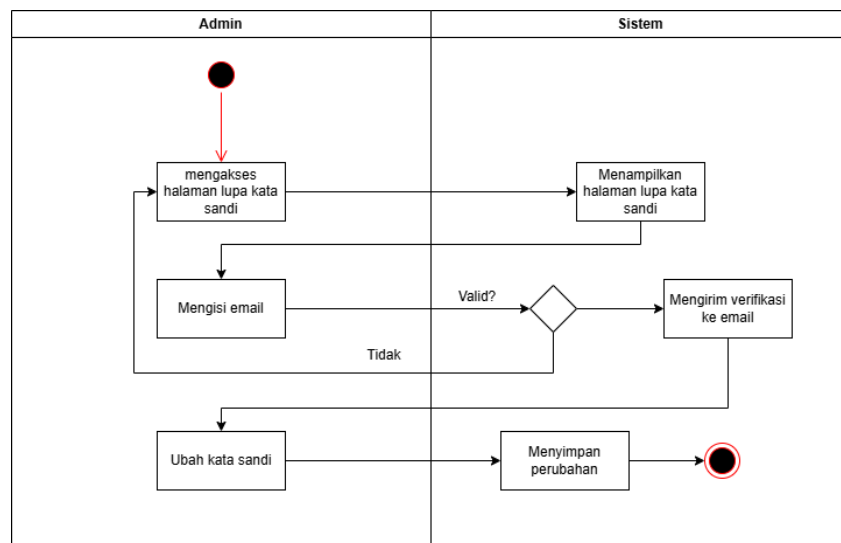
Pada Gambar 3.26 dapat dilihat *activity diagram* yang menggambarkan alur proses mengelola profil. Dimulai dari pengguna (Admin, panitia dan koordinator) masuk ke sistem sampai berhasil mengelola profil.



Gambar 3.26 *Activity Diagram* Mengelola Profil

3.2.5.16 *Activity Diagram* Mengubah Kata Sandi

Pada Gambar 3.27 dapat dilihat *activity diagram* yang menggambarkan alur proses mengubah kata sandi. Dimulai dari pengguna (Admin, panitia dan koordinator) masuk ke sistem sampai berhasil memperbarui kata sandi.

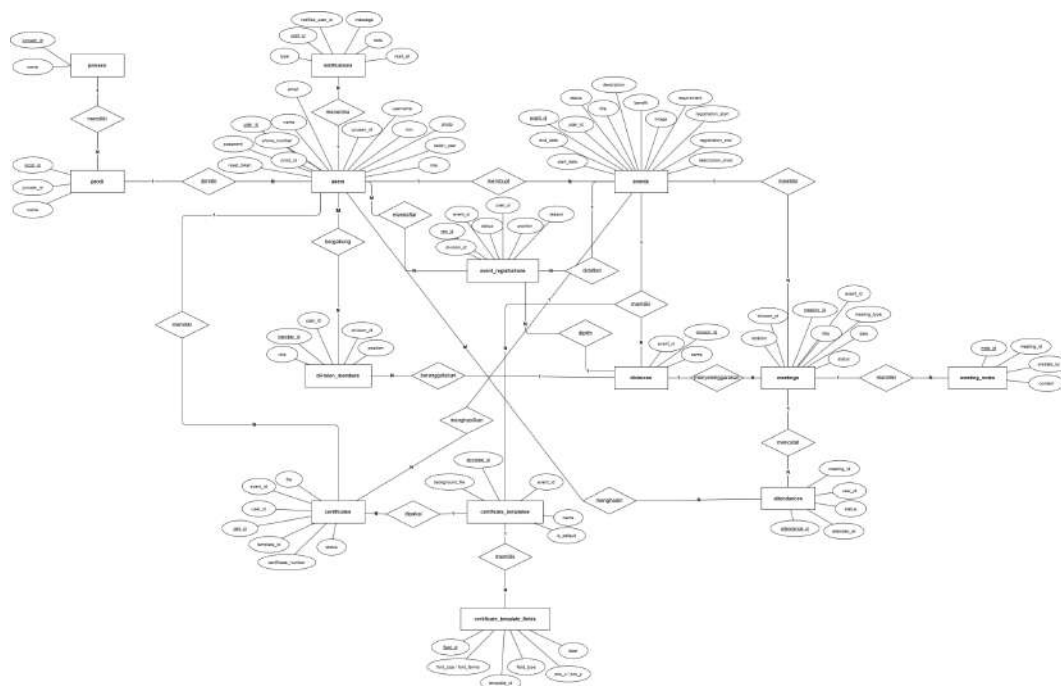


Gambar 3.27 *Activity Diagram* Mengubah Kata Sandi

3.2.6 ER Diagram

Entity Relationship Diagram (ERD) digunakan untuk menjelaskan struktur basis data dan relasi antara entitas. Dalam EVOMA, ERD dirancang untuk mendukung proses pengelolaan kepanitiaan acara organisasi mahasiswa, mulai dari pendaftaran panitia, pembagian divisi, pengelolaan rapat, absensi dan notulen, sampai pada penerbitan sertifikat.

Pada sistem EVOMA terdapat 14 entitas yang terdiri atas Jurusan, Program Studi, Users, Notifications, Events, Event Registrations, Divisions, Division Members, Meetings, Meeting Note, Attendances, Certificate Templates, Certificate Template Field, dan Certificates. Masing-masing entitas terdiri atas atribut nya masing-masing dan saling terhubung dalam bentuk relasi.



Gambar 3.28 *Entity Relationship Diagram*

Relasi antar entitas pada sistem EVOMA menggambarkan relasi antar entitas satu dengan entitas lainnya untuk mendukung proses bisnis sistem. Adapun relasi yang terbentuk dapat dijelaskan sebagai berikut:

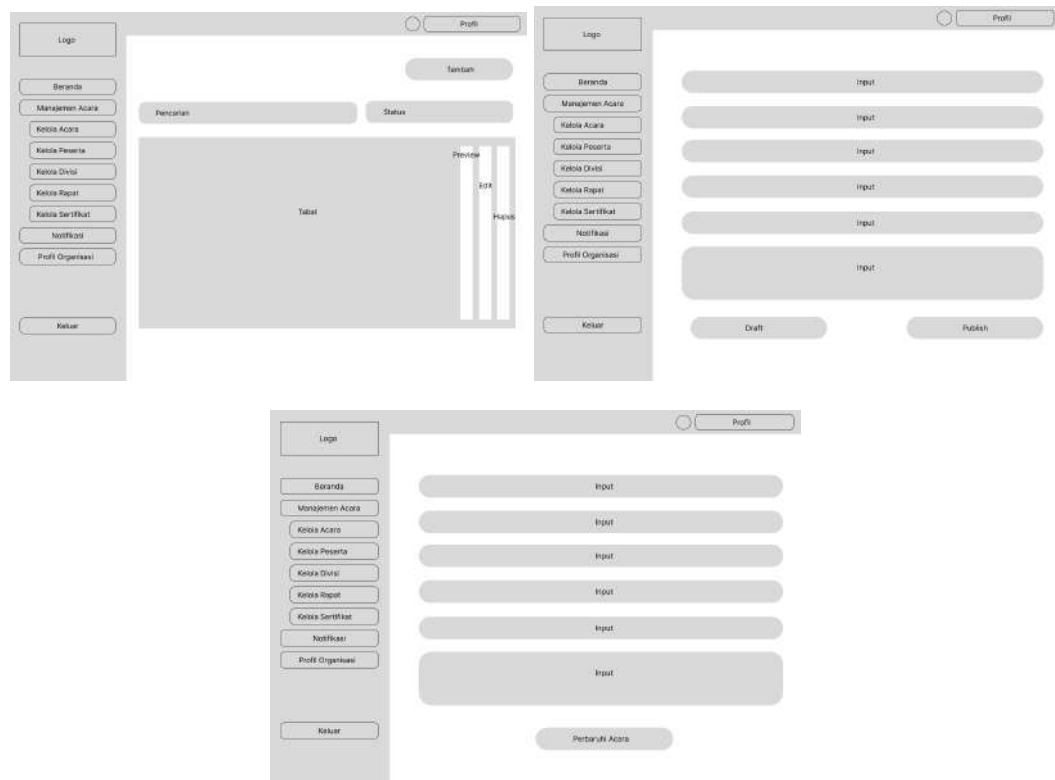
1. Entitas jurusan mempunyai hubungan (relasi) *One-to-Many* dengan entitas prodi, dimana sebuah jurusan bisa mempunyai banyak prodi dan satu prodi hanya memiliki satu jurusan.
2. Entitas prodi mempunyai hubungan (relasi) *One-to-Many* dengan entitas *users*, dimana sebuah prodi bisa mempunyai banyak pengguna dan satu pengguna hanya berasal dari satu prodi.
3. Entitas *users* mempunyai hubungan (relasi) *One-to-Many* dengan entitas *notifications*, dimana seorang pengguna bisa mempunyai banyak notifikasi dan satu notifikasi hanya ditunjukkan kepada satu pengguna.
4. Entitas *users* mempunyai hubungan (relasi) *One-to-Many* dengan entitas *events*, dimana seorang pengguna bisa membuat banyak acara dan satu acara hanya dibuat oleh satu pengguna.
5. Entitas *events* mempunyai hubungan (relasi) *One-to-Many* dengan entitas *divisions*, dimana sebuah acara bisa mempunyai banyak divisi dan satu divisi hanya dimiliki oleh satu acara.
6. Entitas *users* memiliki relasi *One-to-Many* dengan entitas *events_registrations*, dimana satu pengguna dapat melakukan pendaftaran pada beberapa acara dan satu data pendaftaran hanya dimiliki oleh satu pengguna.
7. Entitas *events* memiliki relasi *One-to-Many* dengan entitas *events_registrations*, dimana satu acara dapat memiliki banyak pendaftar dan satu data pendaftaran hanya terkait dengan satu acara.
8. Entitas *divisions* memiliki relasi *One-to-Many* dengan entitas *events_registrations*, dimana satu divisi dapat dipilih oleh banyak pendaftar dan satu pendaftar hanya dapat memilih satu divisi pada satu proses pendaftaran.
9. Entitas *users* memiliki relasi *One-to-Many* dengan entitas *divison_members*, dimana satu pengguna dapat menjadi anggota beberapa divisi pada acara yang berbeda dan satu data anggota divisi hanya dimiliki oleh satu pengguna.

10. Entitas *divisions* memiliki relasi *One-to-Many* dengan entitas *division_members*, dimana satu divisi dapat memiliki banyak anggota dan satu data anggota hanya berada pada satu divisi..
11. Entitas *division* mempunyai hubungan (relasi) *One-to-Many* dengan entitas *meetings*, dimana sebuah divisi bisa memiliki banyak rapat dan satu rapat hanya diselenggarakan di satu divisi.
12. Entitas *events* mempunyai hubungan (relasi) *One-to-Many* dengan entitas *meetings*, dimana sebuah acara bisa mempunyai banyak rapat dan satu rapat hanya terkait dengan satu acara.
13. Entitas *meeting* memiliki relasi *One-to-Many* dengan entitas *meeting_notes*, dimana satu rapat dapat memiliki beberapa notulen dan satu notulen hanya terkait dengan satu rapat.
14. Entitas *meetings* mempunyai hubungan (relasi) *One-to-Many* dengan entitas *attendances*, dimana sebuah rapat mempunyai banyak kehadiran dan satu data kehadiran hanya terkait pada satu rapat.
15. Entitas *users* mempunyai hubungan (relasi) *One-to-Many* dengan entitas *attendances*, dimana seorang pengguna bisa mempunyai banyak riwayat kehadiran dan satu data kehadiran hanya dimiliki satu pengguna.
16. Entitas *certificate_templates* memiliki relasi *One-to-Many* dengan entitas *certificate_template_field*, dimana satu template memiliki banyak *field* dan satu *field* hanya dimiliki satu template.
17. Entitas *certificate_templates* mempunyai hubungan (relasi) *One-to-Many* dengan entitas *certificates*, dimana sebuah template bisa digunakan oleh banyak entitas dan satu sertifikat hanya menggunakan satu template.
18. Entitas *users* mempunyai hubungan (relasi) *One-to-Many* dengan entitas *certificates*, dimana satu pengguna hanya bisa menerima sertifikat dari acara yang berbeda dan satu sertifikat hanya diberikan kepada satu pengguna.
19. Entitas *events* mempunyai hubungan (relasi) *One-to-Many* dengan entitas *certificates*, dimana sebuah acara bisa menghasilkan banyak sertifikat dan satu sertifikat hanya terkait dengan satu acara.

3.2.7 Perancangan Antarmuka (*Wireframe*)

3.2.7.1 Antarmuka Mengelola Acara

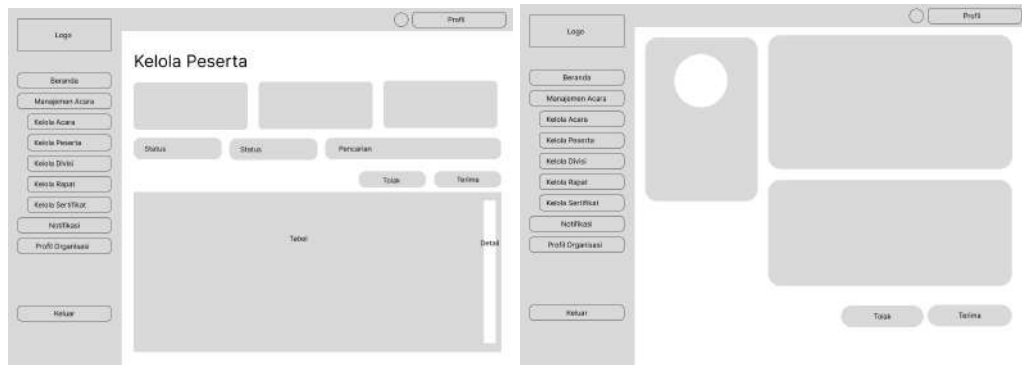
Wireframe mengelola acara disajikan pada Gambar 3.29. Antarmuka ini dirancang agar memudahkan admin organisasi dalam mengelola seluruh data acara yang terdapat pada sistem.



Gambar 3.29 Antarmuka Mengelola Acara

3.2.7.2 Antarmuka Verifikasi Panitia Acara

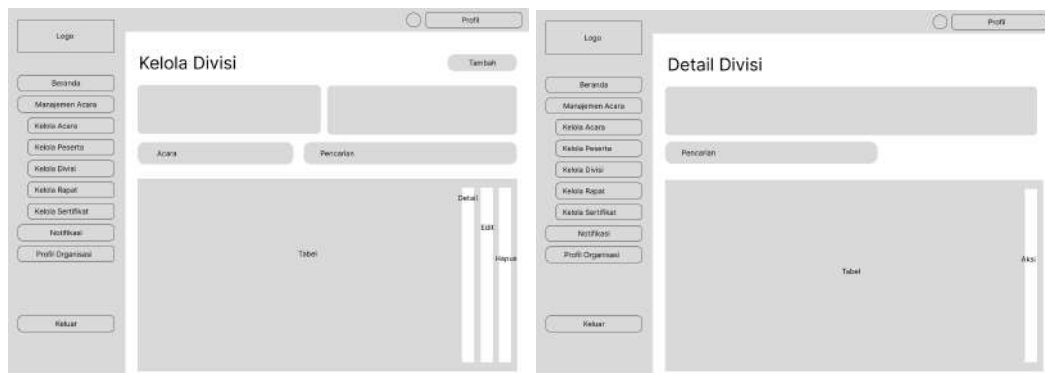
Wireframe verifikasi panitia acara disajikan pada Gambar 3.30. Antarmuka ini dirancang untuk memudahkan admin organisasi dalam memverifikasi panitia acara.



Gambar 3.30 Antarmuka Verifikasi Panitia Acara

3.2.7.3 Antarmuka Mengelola Hak Akses Panitia

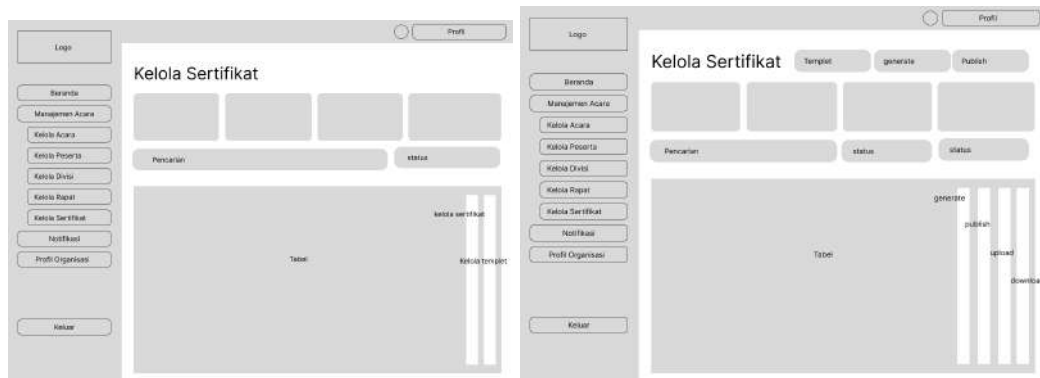
Wireframe mengelola hak akses disajikan pada Gambar 3.31. Antarmuka ini dirancang untuk memudahkan admin organisasi dalam mengelola hak akses panitia acara.



Gambar 3.31 Antarmuka Mengelola Hak Akses Pengguna

3.2.7.4 Antarmuka Mengelola Sertifikat

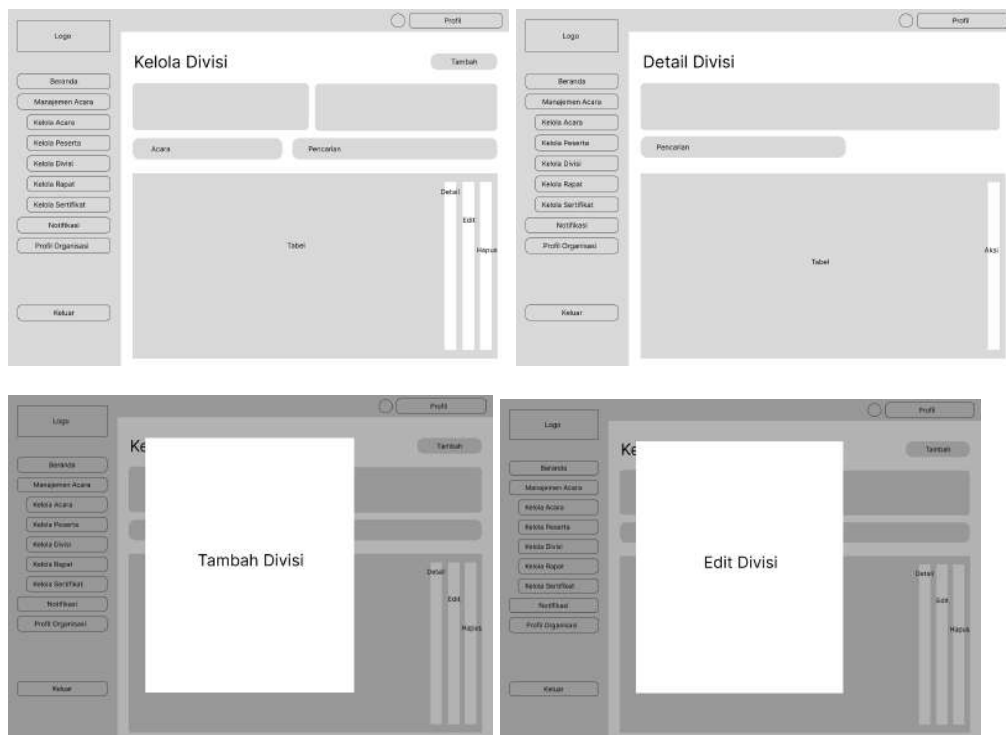
Wireframe mengelola sertifikat disajikan Gambar 3.32, dirancang untuk memudahkan admin organisasi dalam mengelola sertifikat peserta dalam acara.



Gambar 3.32 Antarmuka Mengelola Sertifikat

3.2.7.5 Antarmuka Mengelola Divisi

Wireframe mengelola divisi disajikan Gambar 3.33, dirancang untuk memudahkan admin organisasi dalam mengelola divisi pada acara.



Gambar 3.33 Antarmuka Mengelola Divisi

3.2.7.6 Antarmuka Mendaftarkan Akun

Wireframe mendaftar akun disajikan pada Gambar 3.34. Antarmuka ini dirancang untuk membantu panitia atau koordinator divisi untuk melakukan pendaftaran akun.



Gambar 3.34 Antarmuka Mendaftarkan Akun

3.2.7.7 Antarmuka Melihat Absensi

Wireframe melihat absensi disajikan pada Gambar 3.35. Antarmuka ini dirancang untuk membantu panitia untuk melihat absensi yang mereka miliki.



Gambar 3.35 Antarmuka Melihat Absensi

3.2.7.8 Antarmuka Melihat Rapat dan Notulen

Wireframe melihat rapat dan notulen disajikan pada Gambar 3.36. Antarmuka ini dirancang untuk membantu panitia untuk melihat rapat dan notulen yang mereka miliki.



Gambar 3.36 Antarmuka Melihat Rapat dan Notulen

3.2.7.9 Antarmuka Mendaftar Acara

Wireframe mendaftar acara disajikan pada Gambar 3.37. Antarmuka ini dirancang untuk membantu panitia untuk melakukan pendaftaran acara yang mereka inginkan.



Gambar 3.37 Antarmuka Mendaftar Acara

3.2.7.10 Antarmuka Mengelola Absensi

Wireframe mengelola absensi disajikan pada Gambar 3.38. Antarmuka ini dirancang untuk membantu admin organisasi dan koordinator divisi untuk mengelola absensi rapat.



Gambar 3.38 Antarmuka Mengelola Absensi

3.2.7.11 Antarmuka Mengelola Rapat dan Notulen

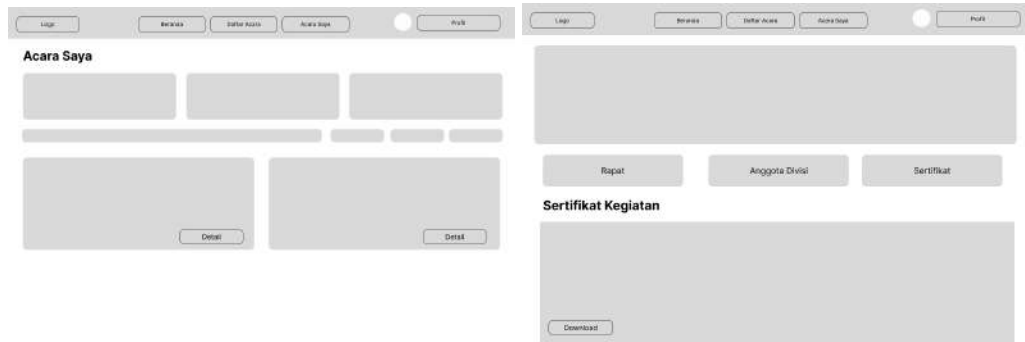
Wireframe mengelola rapat dan notulen disajikan pada Gambar 3.39. Antarmuka ini dirancang untuk membantu admin organisasi dan koordinator divisi untuk mengelola rapat dan notulen.



Gambar 3.39 Antarmuka Mengelola Rapat dan Notulen

3.2.7.12 Antarmuka Mendapatkan Sertifikat

Wireframe mendapatkan sertifikat disajikan Gambar 3.40, dirancang untuk membantu panitia dan koordinator divisi untuk mengunduh sertifikat.



Gambar 3.40 Antarmuka Mendapatkan Sertifikat

3.2.7.13 Antarmuka Masuk ke Sistem

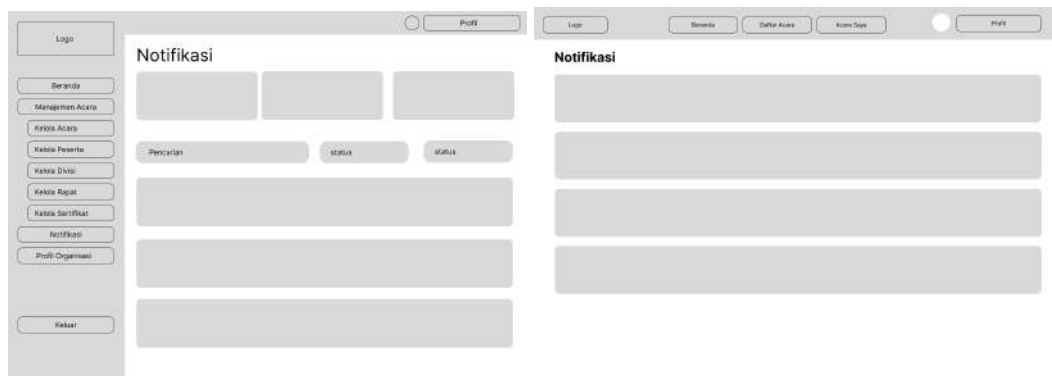
Wireframe masuk ke sistem disajikan pada Gambar 3.41. Antarmuka ini dirancang untuk membantu admin organisasi, panitia dan koordinator divisi untuk memasuki sistem.



Gambar 3.41 Antarmuka Masuk ke Sistem

3.2.7.14 Antarmuka Menerima Notifikasi

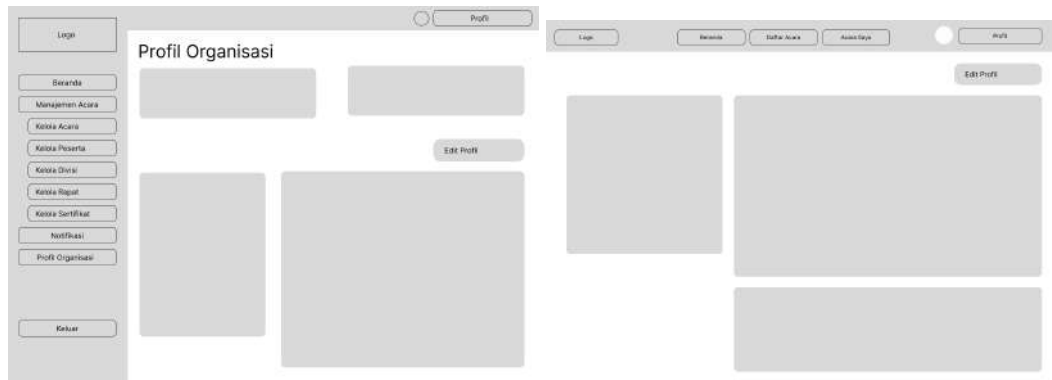
Wireframe menerima notifikasi disajikan pada Gambar 3.42. Antarmuka ini dirancang untuk membantu admin organisasi, panitia dan koordinator divisi untuk menerima notifikasi.



Gambar 3.42 Antarmuka Menerima Notifikasi

3.2.7.15 Antarmuka Mengelola Profil

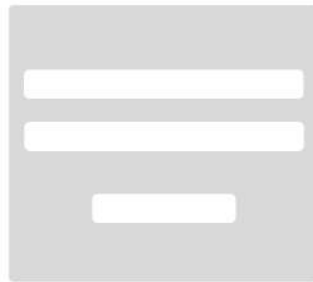
Wireframe mengelola profil disajikan pada Gambar 3.43. Antarmuka ini dirancang untuk membantu admin organisasi, panitia dan koordinator divisi untuk mengelola informasi profil yang dimiliki.



Gambar 3.43 Antarmuka Mengelola Profil

3.2.7.16 Antarmuka Mengubah Kata Sandi

Wireframe mengubah kata sandi disajikan pada Gambar 3.44. Antarmuka ini dirancang untuk membantu admin organisasi, panitia dan koordinator divisi untuk mengubah kata sandi.



Gambar 3.44 Antarmuka Mengubah Kata Sandi

BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1. Hasil Implementasi

4.1.1 Deskripsi dan Proses Implementasi

EVOMA merupakan aplikasi pengelolaan kegiatan kepanitiaan organisasi mahasiswa berbasis web. Sistem dirancang untuk membantu proses pengelolaan acara, rapat, absensi, sertifikat digital dan notifikasi secara terintegrasi. Dalam implementasinya, EVOMA dikembangkan melalui dua komponen utama, yaitu *Frontend* yang berfungsi sebagai antarmuka dan *Backend* yang berfungsi sebagai pengelolaan logika bisnis, pengolahan data dan penyedia layanan API.

1. *Frontend* dikembangkan menggunakan *framework* Quasar yang membantu dalam membangun antarmuka responsif dan mudah digunakan.
2. *Backend* dikembangkan menggunakan NestJs dengan bahasa pemrograman Typescript. *Backend* terhubung dengan *database* untuk menyimpan dan mengelola data sistem.
3. Sistem menggunakan layanan AWS S3 sebagai media penyimpanan file.
4. Sistem juga menggunakan Gmail SMTP sebagai layanan pengiriman email yang dimanfaatkan dalam fitur mengubah kata sandi.

Proses implementasi sistem dilakukan melalui beberapa tahapan, yaitu mempersiapkan lingkungan pengembangan, melakukan konfigurasi basis data, instalasi dependensi aplikasi, migrasi *database*, dan menjalankan layanan *frontend* dan *backend*. Berikut merupakan alur dalam melakukan implementasi sistem.

1. Melakukan kloning (*clone*) repository proyek.
2. Melakukan konfigurasi file environment untuk koneksi *database*, layanan penyimpanan file dan layanan email.
3. Membuat database MySQL yang akan dikembangkan oleh sistem.
4. Menginstal seluruh dependensi *Backend* dan *Frontend*.
5. Menjalankan migrasi *database*.
6. Menjalankan server *Backend*.

7. Menjalankan server *Frontend*.
8. Melakukan pengujian akses aplikasi melalui *browser* untuk memverifikasi bahwa seluruh fitur telah berjalan sesuai standar.

Tabel 4.1 Dependensi dan *Tools* yang Digunakan

No	Software / Depedensi	Versi	Fungsi
1	Node.js	v18	Menjalankan <i>package</i> manager npm dan proses <i>build</i> aplikasi
2	Bun	v1.x	<i>Runtime backend</i>
3	MySQL	v8.0	Sistem manajemen basis data
4	npm	v6.13.4	Mengelola <i>package</i> dan <i>dependency frontend</i>
5	Quasar CLI	v2.x	Menjalankan <i>frontend</i>
6	Git	Terbaru	<i>Version control</i> dan <i>clone repository</i>
7	NestJs	v10.x	Framework <i>backend</i>
8	Sequelize	v6.x	ORM yang menghubungkan aplikasi dengan database MySQL
9	Vue.js	v3.x	<i>Framework frontend</i>
10	Quasar Framework	v2.x	<i>Framework</i> UI berbasis Vue.js
11	AWS SDK	Sesuai <i>package</i>	Integrasi penyimpanan file
12	Nodemailer	Sesuai <i>Package</i>	Pengiriman email melalui Gmail SMTP

4.1.2 Fitur Aplikasi

4.1.1.1 Mengelola Acara

Fitur mengelola acara merupakan fitur yang digunakan oleh admin organisasi untuk melakukan pengelolaan acara baik membuat, mengubah, sampai menghapus. Acara yang berhasil dikelola oleh admin organisasi

dengan status pendaftaran dibuka akan ditampilkan pada halaman daftar acara dan dapat diakses oleh panitia untuk melihat informasi maupun melakukan pendaftaran.

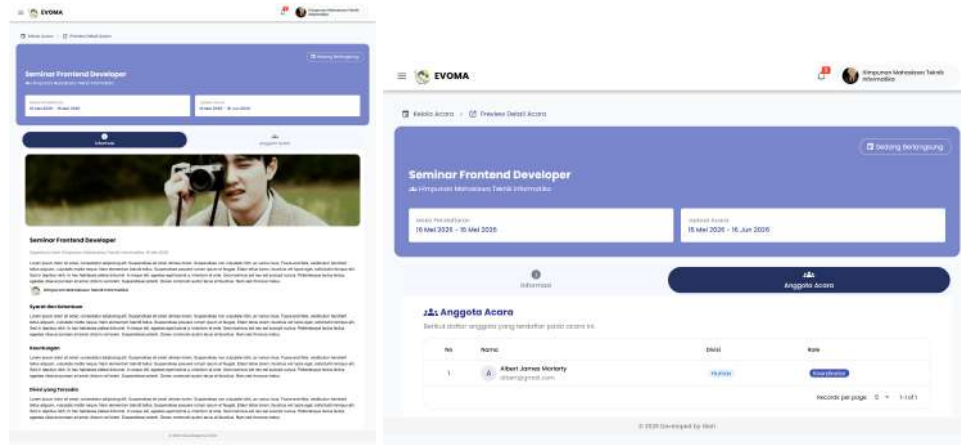
Acara	Pendaftaran	Pelaksanaan	Status	Publish	Aksi
Acara Akan Datang Test Himpunan Mahasiswa Teknik Informatika	01 Jun 2026 s/d 30 Jun 2026	30 Jun 2026 s/d 31 Jul 2026	Mendukung Dibuka	Sudah Publish	[Edit] [Delete] [Publish]
Seminar Frontend Developer Himpunan Mahasiswa Teknik Informatika	18 Mei 2026 s/d 18 Mei 2026	16 Mei 2026 s/d 16 Jun 2026	Sedang Berlangsung	Sudah Publish	[Edit] [Delete] [Publish]
Seminar AI Himpunan Mahasiswa Teknik Informatika	19 Mei 2026 s/d 31 Mei 2026	31 Mei 2026 s/d 30 Jun 2026	Pendaftaran Dibuka	Sudah Publish	[Edit] [Delete] [Publish]
HMTI Fair 2026 Himpunan Mahasiswa Teknik Informatika	01 Mei 2026 s/d 31 Mei 2026	31 Mei 2026 s/d 15 Jul 2026	Pendaftaran Dibuka	Sudah Publish	[Edit] [Delete] [Publish]
HMTI Fair Himpunan Mahasiswa Teknik Informatika	- s/d -	- s/d -	Tutup	Sudah Publish	[Edit] [Delete] [Publish]

Gambar 4.1 Halaman Kelola Acara

Gambar 4.1 adalah gambar daftar acara yang menampilkan seluruh data acara yang telah dibuat oleh admin organisasi. Admin organisasi dapat melihat, melakukan pencarian, memfilter status, melihat detail, mengubah data, menambah acara, menghapus acara, sampai mempublish acara melalui menu-menu yang tersedia pada halaman.

Gambar 4.2 Halaman Tambah Edit Acara

Gambar 4.2 adalah gambar tambah acara dan edit acara. Admin organisasi dapat mengisi atau mengubah informasi dari acara pada halaman ini.

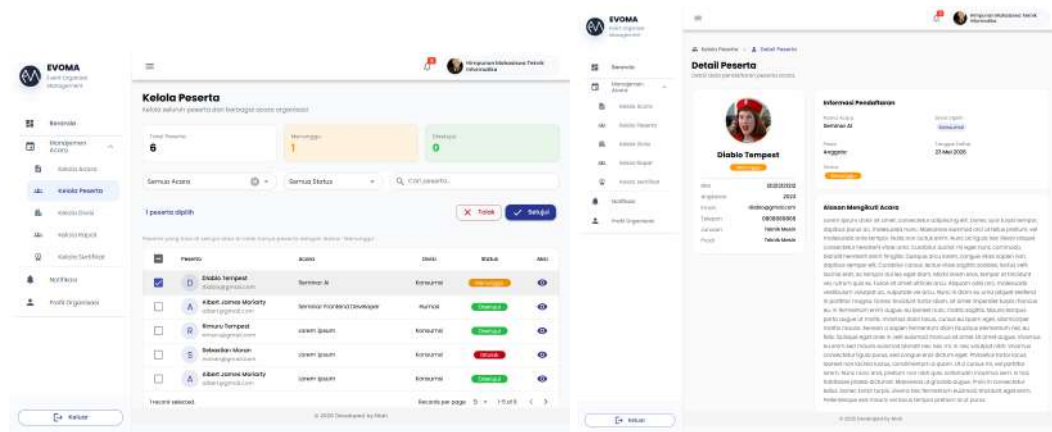


Gambar 4.3 Halaman Detail Acara

Gambar 4.3 adalah gambar detail acara. Admin organisasi dapat melihat detail acara yang telah dibuat beserta dengan anggota-anggota yang telah terdaftar di dalam acara.

4.1.1.2 Verifikasi Panitia Acara

Fitur verifikasi panitia acara merupakan fitur yang digunakan oleh admin organisasi untuk menyetujui atau menolak panitia yang mendaftar pada acara. Panitia yang disetujui secara default akan di setting sebagai anggota (panitia) acara.

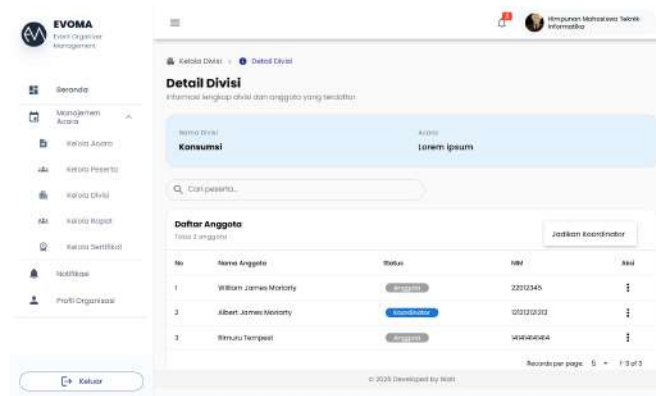


Gambar 4.4 Halaman Verifikasi Panitia Acara

Gambar 4.4 adalah gambar verifikasi panitia acara yang dapat dilakukan secara bersamaan dan satu persatu dengan mengakses halaman detail. Panitia yang dapat disetujui atau ditolak merupakan panitia dengan status menunggu, dan mendaftarkan diri di acara yang statusnya masih pendaftaran di buka.

4.1.1.3 Mengelola Hak Akses Panitia

Fitur mengelola hak akses panitia merupakan fitur yang digunakan oleh admin organisasi untuk mengubah tipe dari panitia berubah menjadi koordinator divisi, sehingga mendapatkan hak akses untuk melakukan pengelolaan terhadap rapat, notulen dan absensi divisi.

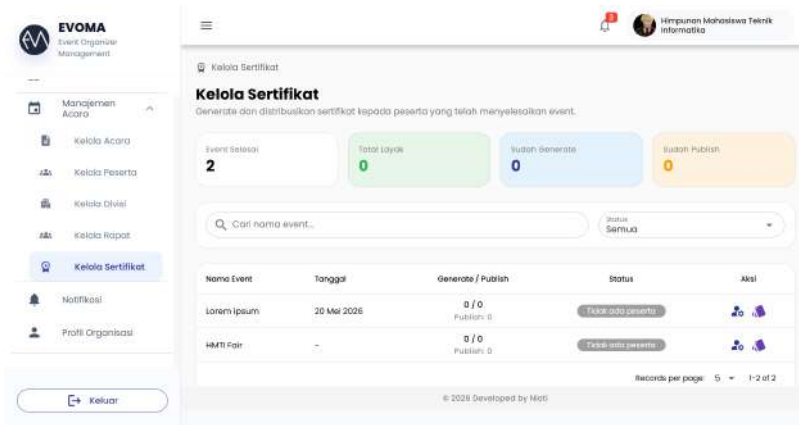


Gambar 4.5 Halaman Hak Akses Panitia

Gambar 4.5 adalah gambar detail divisi untuk mengubah hak akses panitia. Admin dapat menekan tombol aksi pada panitia yang diinginkan, kemudian menekan jadikan koordinator untuk mengubah hak akses panitia tersebut terhadap divisi pada acara.

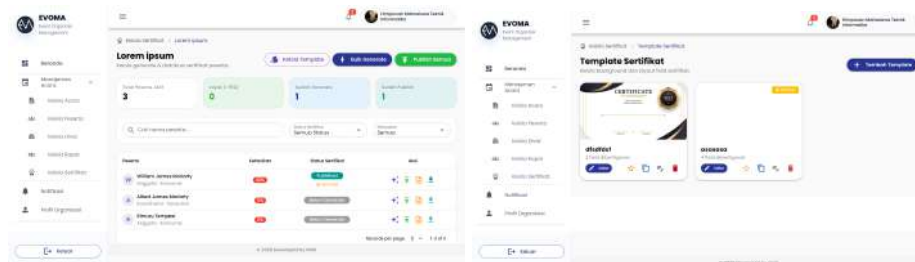
4.1.1.4 Mengelola Sertifikat

Fitur mengelola hak akses panitia merupakan fitur yang digunakan oleh admin organisasi untuk mengubah tipe dari panitia berubah menjadi koordinator divisi, sehingga mendapatkan hak akses untuk melakukan pengelolaan terhadap rapat, notulen dan absensi divisi.



Gambar 4.6 Halaman Kelola Sertifikat

Gambar 4.6 adalah gambar kelola sertifikat. Acara yang berstatus selesai akan ditampilkan pada halaman kelola sertifikat ini untuk melakukan pengelolaan terhadap sertifikat panitia yang telah berpartisipasi.

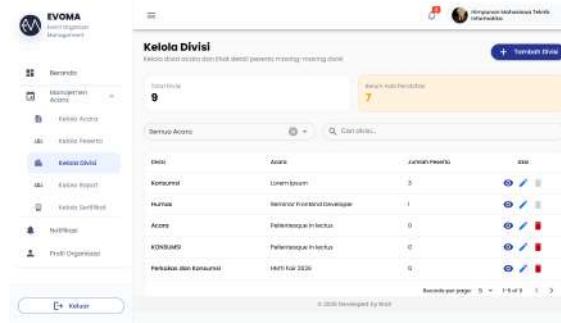


Gambar 4.7 Halaman Detail Kelola Sertifikat

Gambar 4.7 adalah gambar detail kelola sertifikat. Admin dapat melakukan pengelolaan terhadap template sertifikat, melakukan generate masal dengan syarat kehadiran minimal 75% dan melakukan upload manual.

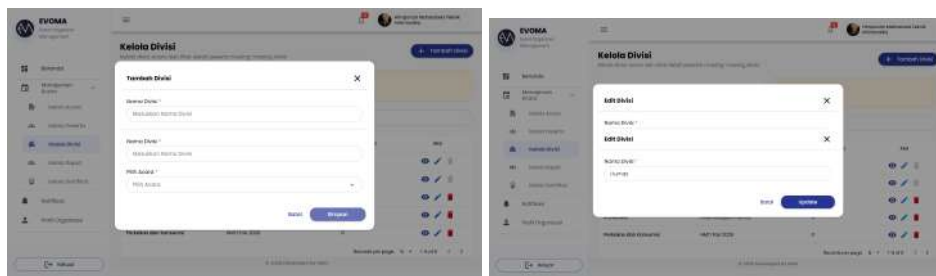
4.1.1.5 Mengelola Divisi

Fitur mengelola hak akses panitia merupakan fitur yang digunakan oleh admin organisasi untuk mengubah tipe dari panitia berubah menjadi koordinator divisi, sehingga mendapatkan hak akses untuk melakukan pengelolaan terhadap rapat, notulen dan absensi divisi.



Gambar 4.8 Halaman Kelola Divisi

Gambar 4.8 adalah gambar kelola divisi. Admin dapat melihat divisi acara yang telah dibuat sebelumnya, admin juga dapat melakukan perubahan informasi dan penghapusan divisi selama divisi belum ada anggota dan acara belum selesai.

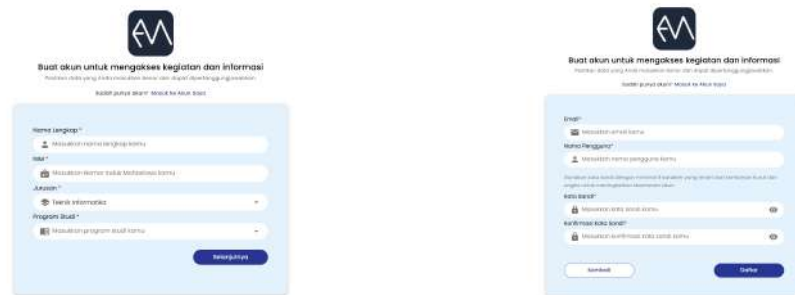


Gambar 4.9 Halaman Tambah Edit Divisi

Gambar 4.9 adalah gambar tambah dan edit divisi pada acara. Admin dapat melakukan penambahan dan perubahan informasi dengan mengisi dialog seperti pada gambar.

4.1.1.6 Mendaftarkan Akun

Fitur mengelola mendaftarkan akun merupakan fitur yang digunakan oleh panitia acara untuk mengakses aplikasi lebih lanjut lagi, sehingga panitia dapat mendaftarkan diri mereka di acara yang ingin mereka ikuti.

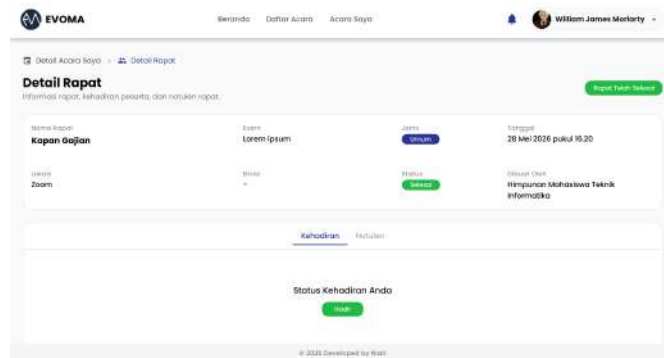


Gambar 4.10 Halaman Daftar Akun

Gambar 4.10 adalah gambar daftar. Panitia dapat mengisi seluruh inputan dengan valid, maka akun panitia akan terdaftar ke dalam sistem.

4.1.1.7 Melihat Absensi

Fitur melihat absensi merupakan fitur yang digunakan oleh panitia untuk melihat status kehadiran mereka pada rapat divisi maupun umum yang dilaksanakan oleh divisi maupun acara.

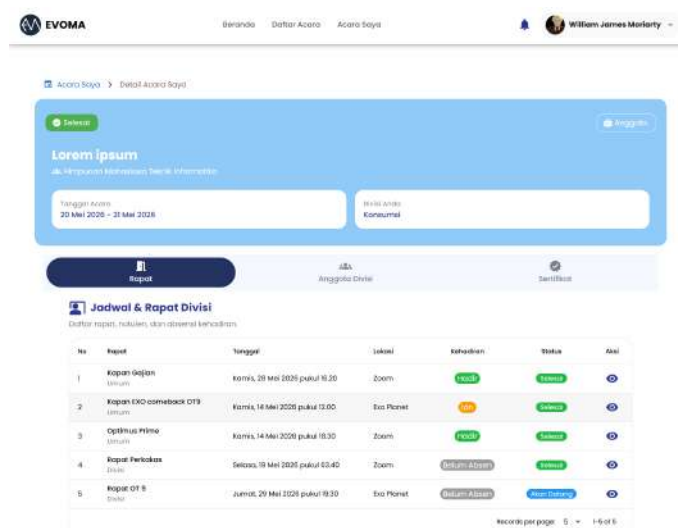


Gambar 4.11 Halaman Melihat Absensi

Gambar 4.11 adalah gambar detail rapat. Panitia dapat mengakses halaman ini dengan menekan detail dari rapat yang telah dilaksanakan oleh divisi atau acara.

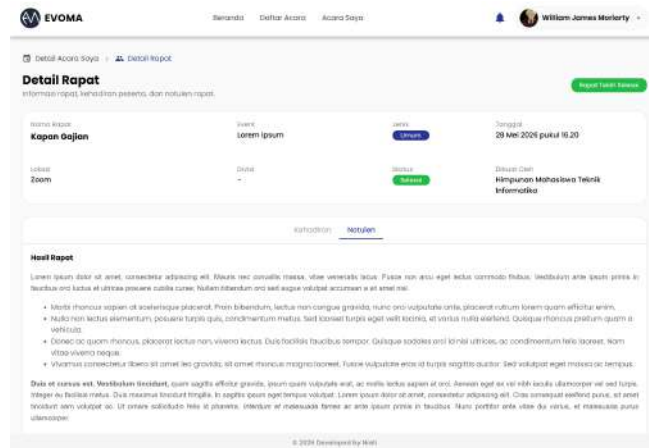
4.1.1.8 Melihat Rapat dan Notulen

Fitur melihat rapat dan notulen merupakan fitur yang digunakan oleh panitia untuk melihat rapat divisi atau rapat acara yang sedang, akan dan telah berlangsung. Kemudian, panitia dapat melihat notulen hasil rapat berstatus selesai.



Gambar 4.12 Halaman Melihat Rapat

Gambar 4.12 adalah gambar rapat panitia. Panitia dapat melihat daftar rapat divisi dan rapat acara. Halaman ini dapat diakses pada menu detail halaman saya.

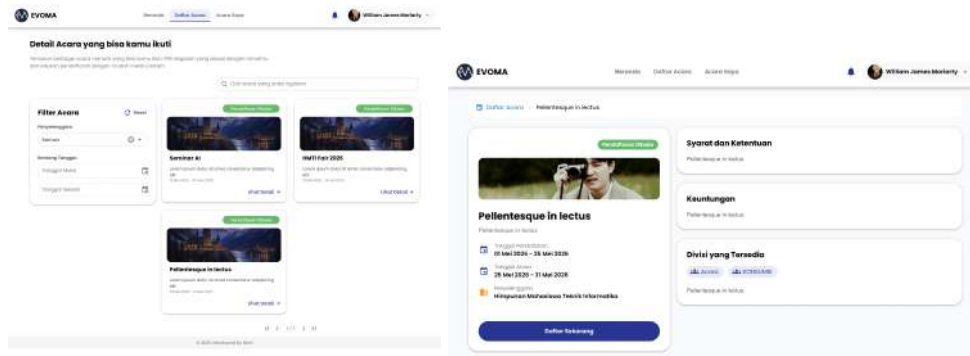


Gambar 4.13 Halaman Melihat Notulen

Gambar 4.13 adalah gambar notulen rapat. Notulen rapat akan tersedia pada rapat yang berstatus selesai, menu ini tersedia pada halaman detail rapat.

4.1.1.9 Mendaftar Acara

Fitur mendaftar acara adalah fitur yang digunakan oleh panitia untuk mendaftarkan diri mereka ke acara yang diinginkan. Pada Gambar 4.14 menunjukkan halaman daftar acara dan detail acara yang tersedia pada aplikasi EVOMA. Acara yang ditampilkan pada halaman daftar acara adalah acara dengan status pendaftaran di buka.



Gambar 4.14 Halaman Daftar Acara

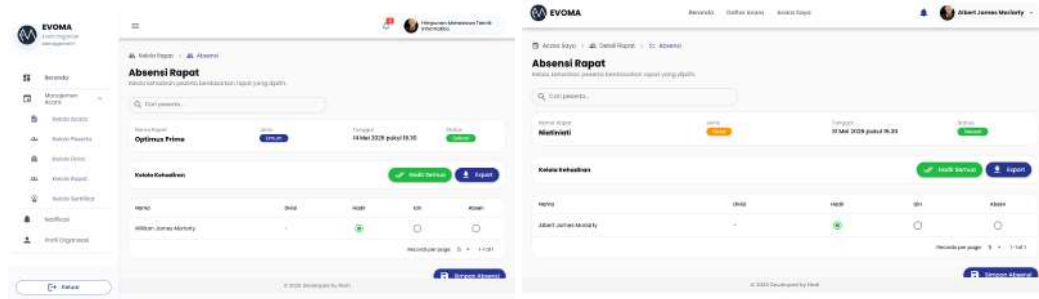
Gambar 4.15 adalah tampilan pendaftaran acara. Panitia dapat melengkapi data diri di profil sebelum mengisi formulir pendaftaran, kemudian panitia dapat mengisi inputan-inputan wajib yang belum diisi.

Gambar 4.15 Halaman Mendaftar Acara

4.1.1.10 Mengelola Absensi

Fitur mengelola absensi adalah fitur yang digunakan oleh admin organisasi dan koordinator divisi untuk mencatat kehadiran dari peserta rapat. Admin organisasi dapat melakukan pengelolaan absensi rapat umum dan koordinator divisi dapat melakukan pengelolaan absensi rapat divisi.

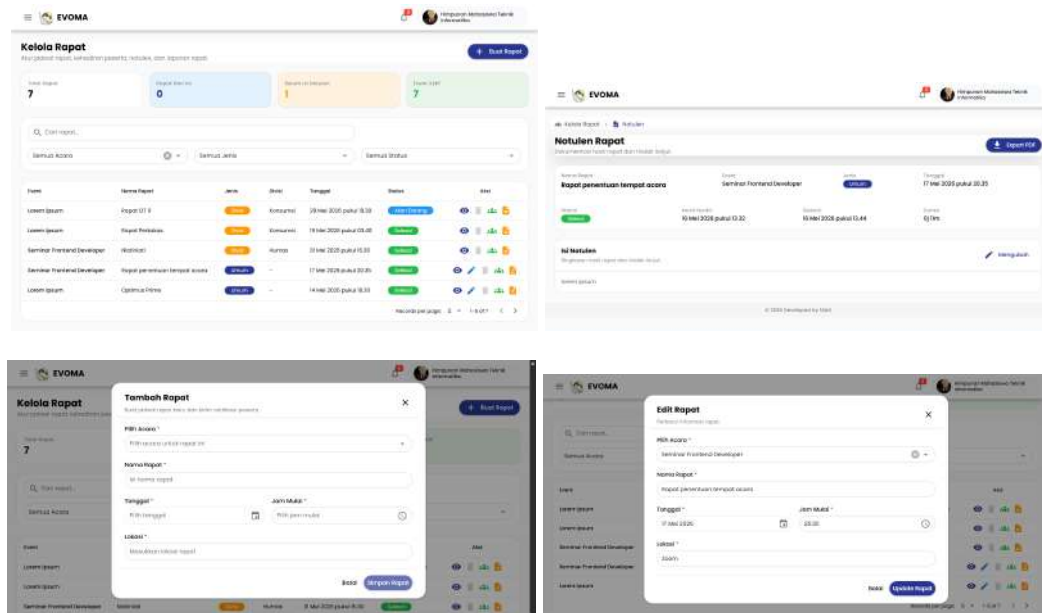
Halaman mengelola absensi tampilan disajikan pada Gambar 4.16 dibawah ini.



Gambar 4.16 Halaman Mengelola Absensi

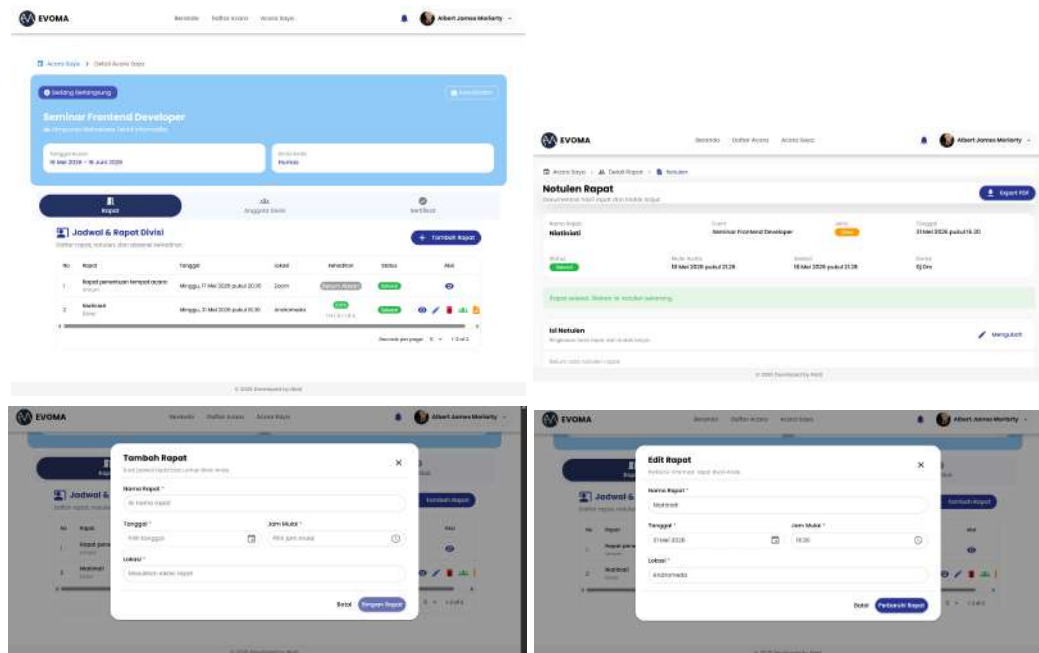
4.1.1.11 Mengelola Rapat dan Notulen

Fitur mengelola rapat dan notulen adalah fitur yang digunakan oleh admin organisasi dan koordinator divisi untuk melakukan pengelolaan terhadap rapat dan notulen hasil rapat. Admin organisasi dapat menambah, mengubah, atau menghapus rapat serta mengisi notulen hasil rapat setelah status rapat adalah selesai. Halaman mengelola rapat dan notulen pada admin disajikan Gambar 4.17.



Gambar 4.17 Halaman Rapat dan Notulen Admin

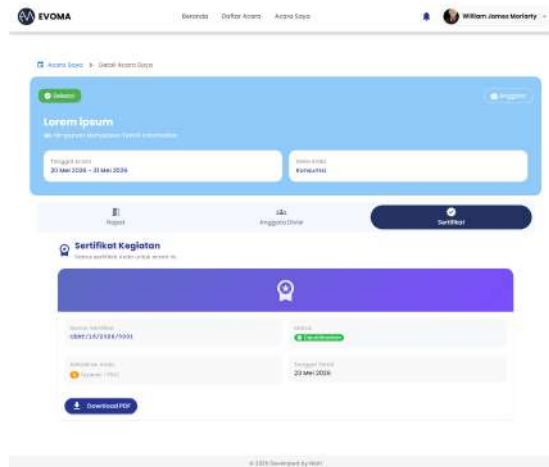
Gambar 4.18 menunjukkan halaman mengelola rapat dan notulen pada koordinator divisi. Sama dengan admin organisasi, koordinator divisi juga dapat mengelola rapat dan notulen, namun hanya sebatas rapat dan notulen pada divisi koordinator menjabat.



Gambar 4.18 Halaman Rapat dan Notulen Koordinator

4.1.1.12 Mendapatkan Sertifikat

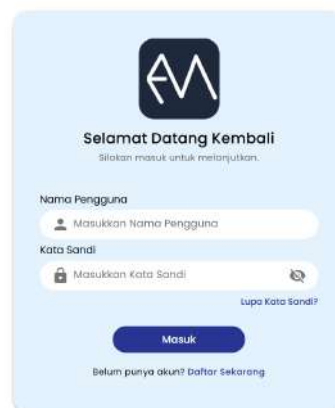
Fitur mendapatkan sertifikat adalah fitur yang digunakan oleh panitia dan koordinator divisi untuk mengunduh sertifikat dari acara yang telah mereka ikuti. Tampilan halaman mendapatkan sertifikat disajikan pada Gambar 4.19 dibawah ini.



Gambar 4.19 Halaman Mendapatkan Sertifikat

4.1.1.13 Masuk ke Sistem

Fitur masuk ke sistem adalah fitur yang digunakan oleh admin organisasi, panitia dan koordinator divisi untuk masuk ke sistem. Pengguna dapat mengisi nama pengguna dan kata sandi, kemudian akan diarahkan ke halaman beranda masing-masing sesuai dengan peran masing-masing. Tampilan halaman masuk ke sistem disajikan Gambar 4.20.

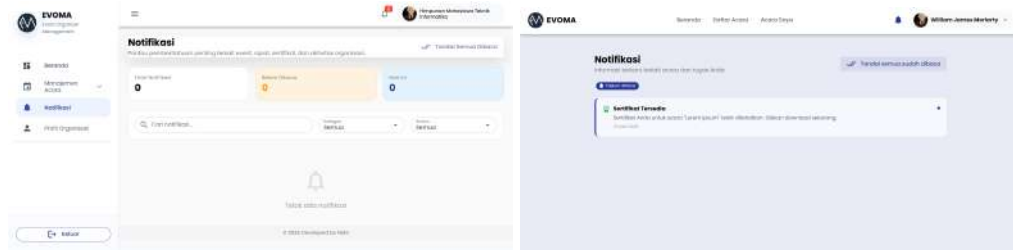


Gambar 4.20 Halaman Masuk

4.1.1.14 Menerima Notifikasi

Fitur menerima notifikasi merupakan fitur yang digunakan oleh admin organisasi, panitia dan koordinator divisi untuk mendapatkan

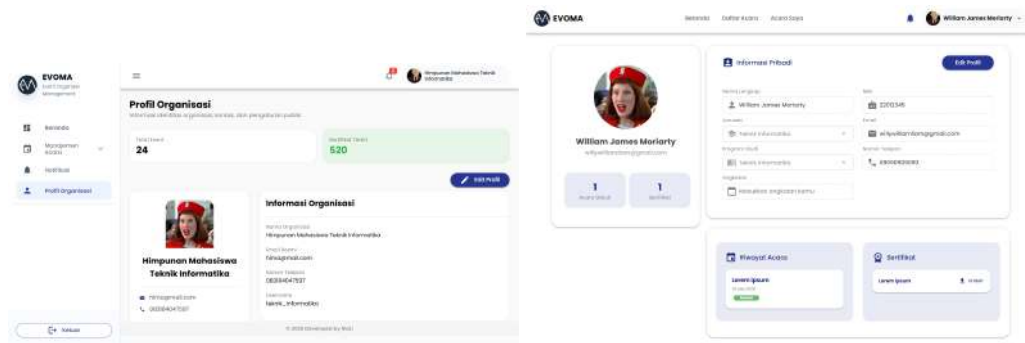
informasi dari pembaruan acara, rapat dan informasi penting lainnya pada aplikasi. Tampilan halaman notifikasi pada admin organisasi, panitia, dan koordinator divisi disajikan Gambar 4.21.



Gambar 4.21 Halaman Notifikasi

4.1.1.15 Mengelola Profil

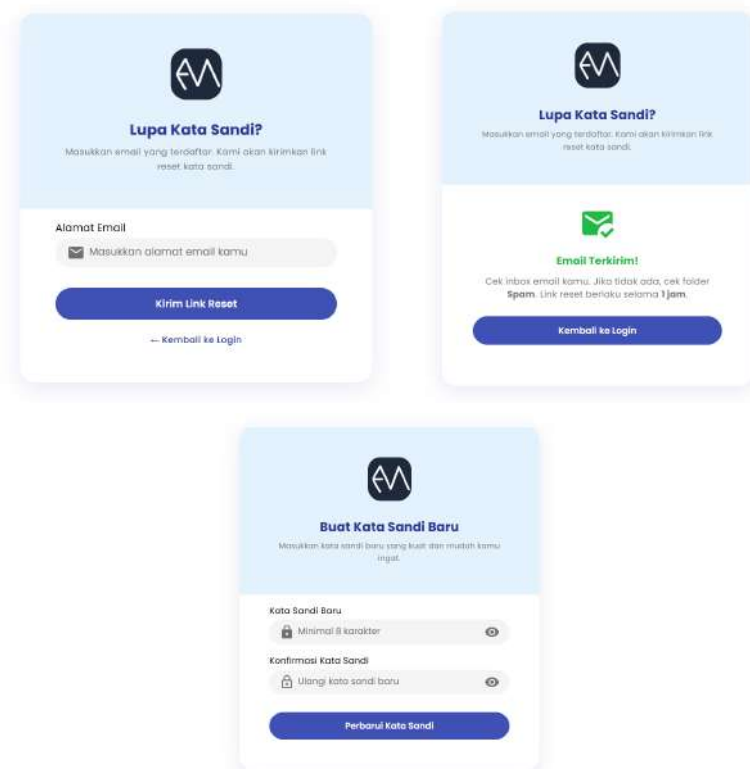
Fitur mengelola profil merupakan fitur yang digunakan oleh admin organisasi, panitia dan koordinator divisi untuk melakukan pengelolaan data diri. Tampilan halaman mengelola profil dapat dilihat Gambar 4.22.



Gambar 4.22 Halaman Profil

4.1.1.16 Mengubah Kata Sandi

Fitur mengubah kata sandi adalah fitur yang digunakan oleh admin organisasi, panitia dan koordinator divisi untuk melakukan perubahan kata sandi. Fitur ini dapat digunakan dengan mengisi email valid, kemudian akan terkirim validasi ke email yang telah diinputkan, pengguna dapat menekan verifikasi dan mengisi kata sandi baru sesuai dengan aturan sistem. Tampilan halaman mengubah kata sandi disajikan Gambar 4.23.



Gambar 4.23 Halaman Masuk

4.1.3 Alur Penggunaan Aplikasi

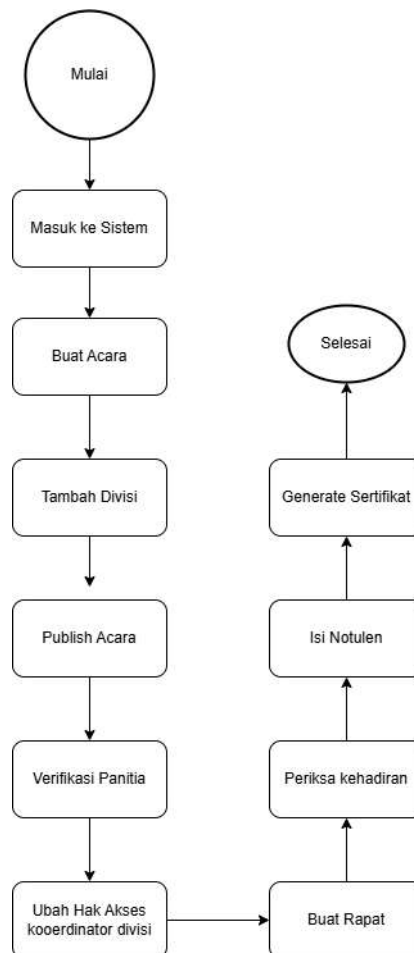
4.1.3.1 Alur Penggunaan Aplikasi Admin Organisasi

Gambar 4.24 menunjukkan alur penggunaan aplikasi EVOMA oleh admin organisasi. Proses dimulai dengan admin organisasi masuk ke sistem dengan kredensial valid, kemudian admin dapat membuat acara baru dan menambahkan divisi. Setelah acara baru berhasil di buat, admin dapat melakukan publish acara agar dapat dilihat oleh calon pendaftar.

Selanjutnya, setelah acara di buat dan terdapat pendaftar, admin dapat melakukan verifikasi peserta disetujui atau ditolak. Setelah peserta diverifikasi, admin dapat melakukan perubahan hak akses ke koordinator divisi pada peserta yang diinginkan.

Kemudian, admin dapat membuat rapat untuk melaksanakan rapat umum, admin dapat mengisi absensi saat rapat berlangsung atau selesai.

Setelah status rapat berubah menjadi selesai, admin dapat mengisi notulen hasil rapat. setelah rangkaian acara selesai, admin dapat melakukan generate sertifikat untuk dibagikan kepada panitia dan koordinator divisi yang sudah berpartisipasi.

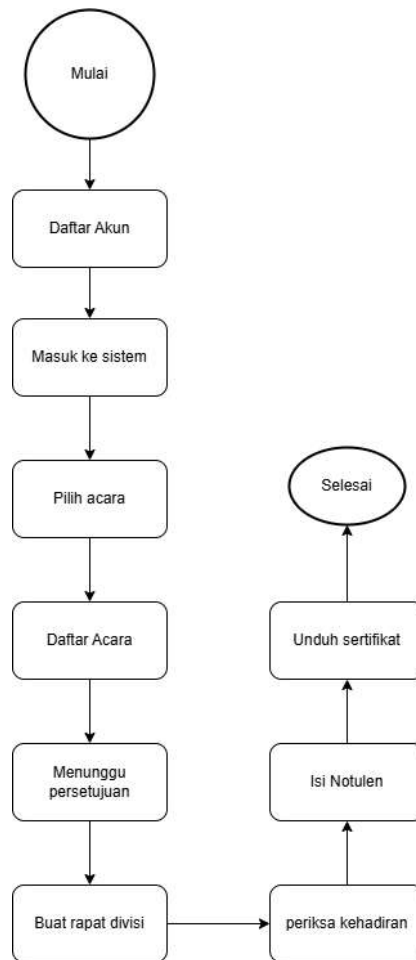


Gambar 4.24 Alur Penggunaan Aplikasi Admin

4.1.3.2 Alur Penggunaan Aplikasi Koordinator Divisi

Gambar 4.25 menunjukkan alur penggunaan aplikasi EVOMA oleh koordinator divisi. Proses dimulai saat koordinator divisi mendaftarkan akun, kemudian koordinator dapat masuk ke sistem dengan kredensial valid. Koordinator dapat memilih acara yang ingin diikuti dan mendaftarkan diri pada acara tersebut, pastikan sebelum melakukan pendaftaran koordinator sudah melengkapi profil. Setelah diverifikasi dan di ubah hak akses menjadi koordinator divisi, koordinator dapat membuat

rapat baru, melakukan absensi dan membuat notulen jika rapat sudah berstatus selesai. Setelah seluruh rangkaian acara selesai, koordinator dapat mengunduh sertifikat yang telah tersedia.

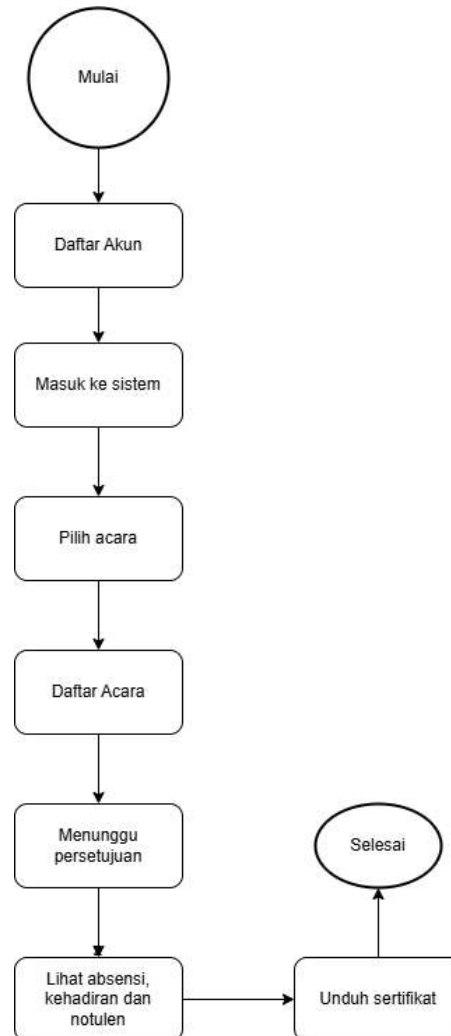


Gambar 4.25 Alur Penggunaan Aplikasi Koordinator

4.1.3.3 Alur Penggunaan Aplikasi Panitia

Gambar 4.26 menunjukkan alur penggunaan aplikasi EVOMA oleh koordinator divisi. Proses dimulai saat koordinator divisi mendaftarkan akun, kemudian koordinator dapat masuk ke sistem dengan kredensial valid. Koordinator dapat memilih acara yang ingin diikuti dan mendaftarkan diri pada acara tersebut, pastikan sebelum melakukan pendaftaran panitia sudah melengkapi profil. Setelah diverifikasi panitia dapat melihat rapat, absensi dirinya sendiri dan notulensi hasil rapat.

Setelah seluruh rangkaian acara selesai, koordinator dapat mengunduh sertifikat yang tersedia.



Gambar 4.26 Alur Penggunaan Aplikasi Panitia

4.2. Pengujian

4.2.1 Pengujian Fungsional

Pengujian fungsional berfungsi untuk memastikan seluruh fungsional sudah berjalan sesuai dengan standar yang telah ditetapkan, pengujian dilakukan menggunakan *BlackBox Testing* dengan membuat skenario *test case* dan menyesuaikan dengan kebutuhan fungsional. Hasil pengujian fungsional disajikan pada Tabel 4.2 dibawah ini.

Tabel 4.2 Pengujian Fungsional

Fungsional	Total Test Case	Total Berhasil	Total Gagal	Persentase Berhasil
Mengelola Acara	55	55	0	100%
Verifikasi Panitia Acara	23	23	0	100%
Mengelola Hak Akses Panitia	4	4	0	100%
Mengelola Sertifikat	30	30	0	100%
Mengelola Divisi	20	20	0	100%
Mendaftarkan Akun	17	17	0	100%
Melihat Absensi	6	6	0	100%
Melihat Rapat dan Notulen	16	16	0	100%
Mendaftar Acara	7	7	0	100%
Mengelola Absensi	16	16	0	100%
Mengelola Rapat dan Notulen	27	27	0	100%
Mendapatkan Sertifikat	6	6	0	100%
Masuk ke Sistem	8	8	0	100%
Menerima Notifikasi	9	9	0	100%
Mengelola Profil	25	25	0	100%
Mengubah Kata Sandi	8	8	0	100%

Berdasarkan pengujian yang telah dilaksanakan, didapat bahwa setiap fungsi telah berjalan sesuai dengan kebutuhan fungsional. Secara keseluruhan, pengujian fungsional yang dirancang dan siap digunakan untuk mendukung pengelolaan kegiatan organisasi mahasiswa secara lebih efektif.

4.2.2 Pengujian *User Acceptance Testing* UAT

Berdasarkan hasil pengujian menggunakan *User Acceptance Testing* (UAT) menggunakan metode *System Usability Scale* (SUS) yang melibatkan 30 responden, terdiri atas 7 Admin Organisasi, 9 Koordinator Divisi dan 14 Anggota Panitia. Maka diperoleh skor rata-rata penilaian adalah 86,75, dengan kategori *Excellent*, grade A dan termasuk dalam *Acceptability Range Acceptable*. Nilai tersebut berada di atas standar, sistem EVOMA memiliki tingkat *usability* yang baik dan dapat diterima oleh pengguna.

Tabel 4.3 Pengujian UAT

Pertanyaan	SS	S	N	TS	STS
Saya berpikir akan menggunakan sistem ini kembali di masa mendatang.	19	8	2	1	0
Saya merasa sistem ini terlalu rumit untuk digunakan.	1	0	3	9	17
Saya merasa sistem ini mudah digunakan.	19	9	2	0	0
Saya membutuhkan bantuan dari orang lain atau teknisi untuk dapat menggunakan sistem ini.	0	4	4	9	13
Saya merasa fitur-fitur dalam sistem ini terintegrasi dengan baik.	25	4	1	0	0
Saya merasa terdapat banyak ketidakkonsistenan dalam sistem ini.	0	0	3	7	20
Saya merasa sebagian besar orang dapat mempelajari penggunaan sistem ini dengan cepat.	21	7	2	0	0
Saya merasa sistem ini membingungkan untuk digunakan.	1	1	1	5	22
Saya merasa percaya diri saat menggunakan sistem ini.	23	5	2	0	0
Saya perlu mempelajari banyak hal sebelum dapat menggunakan sistem ini.	2	3	5	3	17

Berdasarkan hasil 30 responden tidak mengalami kendala berarti dalam menggunakan EVOMA. Beberapa pengguna memerlukan waktu untuk menyesuaikan diri dalam memakai sistem, namun keseluruhan sistem tetap mudah di pahami. Adapun terdapat beberapa saran dari pengguna mencakup pengembangan versi *mobile*, memberikan notifikasi email atau WhatsApp, serta fitur pengingat otomatis.

Secara keseluruhan, Evoma telah memenuhi ekspektasi pengguna dari sisi kemudahan penggunaan dan integrasi fitur, serta berpeluang untuk dikembangkan lebih lanjut.

BAB V

KESIMPULAN

5.1 Kesimpulan

Dari hasil perancangan dan pembangunan sistem, maka disimpulkan bahwa tujuan proyek akhir untuk membangun sistem informasi pengelolaan kegiatan kepanitiaan organisasi mahasiswa berbasis website dengan nama *Event Organization Management (EVOMA)* telah tercapai. Sistem yang dikembangkan mampu mendukung proses pengelolaan kepanitiaan acara, publikasi acara terpusat, pendaftaran panitia, verifikasi, pengelolaan divisi, mengelola rapat, notulen, absensi, sertifikat, notifikasi hingga profil pengguna.

Hasil evaluasi fungsional yang dilakukan menggunakan *Black Box Testing* menampilkan sebanyak 277 *test case* yang telah dirancang berhasil dijalankan tanpa ditemukan kegagalan. Hasil tersebut memperlihatkan seluruh fungsi dari sistem telah sesuai dengan standar yang telah ditetapkan.

Selain itu, hasil pengujian menggunakan *User Acceptance Testing (UAT)* dengan metode *System Usability Scale (SUS)* yang melibatkan 30 responden memperoleh skor rata-rata sebesar 86,75. Nilai tersebut berada di kategori *Excellent* dan *Acceptable*, hal tersebut menunjukkan bahwa EVOMA telah berhasil memenuhi kebutuhan pengguna.

5.2 Saran

Dari hasil pengembangan dan pengujian, terdapat beberapa saran yang dapat dipertimbangkan untuk pengembangan selanjutnya.

1. Mengembangkan sistem versi *mobile* agar pengguna dapat mengakses melalui perangkat *smartphone*.
2. Menambahkan fitur notifikasi email atau WhatsApp untuk memberikan informasi secara *real-time*.
3. Menambahkan fitur pengingat otomatis untuk membantu panitia mengingat jadwal dan agenda kegiatan.

DAFTAR PUSTAKA

- Adimurdaka, D. D., & Jati, H. (2022). PENGEMBANGAN SISTEM INFORMASI MANAJEMEN ORGANISASI MAHASISWA (ORMAWA) SEBAGAI SARANA PENDATAAN AGENDA KEGIATAN ORMAWA BERBASIS ANDROID DI HIMPUNAN MAHASISWA TEKNIK ELEKTRONIKA DAN INFORMATIKA. *E-JPTI (Jurnal Elektronik Pendidikan Teknik Informatika) PENGEMBANGAN*, 10(2).
<https://doi.org/https://doi.org/10.21831/e-jpti.v10i2.19109>
- Adisa, Y., & Nasution, M. I. P. (2023). Konsep Dan Peran Sistem Manajemen Basis Data Relasional Pada Sistem Informasi Manajemen. *MASIP: Jurnal Manajemen Administrasi Bisnis Dan Publik Terapan*, 1(3), 76–83.
<https://doi.org/https://doi.org/10.59061/masip.v1i3.314>
- Ardilla, V. Y., & Warsito, A. B. (2025). Rancang bangun aplikasi pemantauan tumbuh kembang dan imunisasi anak dengan framework quasar. *JIKA (Jurnal Informatika)*, 9(1), 1–8. <https://doi.org/https://doi.org/10.31000/jika.v9i1.12423>
- Bassil, Y. (2012). A Simulation Model for the Waterfall Software Development Life Cycle. *International Journal of Engineering & Technology (IJET)*, 2(5).
- Firmansyah, M., Mustika, & Pujiyanto. (2025). IMPLEMENTASI ROLE-BASED ACCESS CONTROL PADA SISTEM INFORMASI LAYANAN ADMINISTRASI PADA LEMBAGA BAHASA UM METRO. *JMIK (JURNAL MAHASISWA ILMU KOMPUTER)*, 6(2), 113–122.
<https://doi.org/https://doi.org/10.24127/ilmukomputer.v6i2.9875>
- Handani, S. S., & Prayoga, R. M. S. (2022). PERAN ORGANISASI KEMAHASISWAAN (ORMAWA) UNTUK MENINGKATKAN SOFT SKILL MAHASISWA FAKULTAS KEGURUAN DAN ILMU PENDIDIKAN UNIVERSITAS BALE BANDUNG. *Jurnal Pendidikan IPS UNIBBA*, 2(2), 23–31.
- Khairy, M. Z., Putra, R. A., Alaudah, N. F., & Tyas, A. A. W. P. (2026). Rancang Bangun Sistem Informasi Manajemen Event Berbasis Mobile dengan Arsitektur Multi-Peran Studi Kasus: Lomba Burung Kicau. *Jurnal Teknik Informatika*. <https://doi.org/Peran Organisasi, Soft Skill>.
- Mokodompit, N. S., Amali, L. N., & Dwinanto, A. (2024). SISTEM INFORMASI PENGELOLAAN ADMINISTRASI ORGANISASI KEMAHASISWAAN BERBASIS WEB DI UNIVERSITAS NEGERI GORONTALO. *Jurnal of*

System and Information Technology, 4(2), 113–122.
<https://doi.org/https://doi.org/10.37031/diffusion.v4i2.26069>

Muhammad, I. R. D., & Paputungan, I. V. (2024). Pengembangan Backend Server Berbasis Arsitektur REST API pada Sistem Transfer Dompot Digital. *Jurnal Sains, Nalar, Dan Aplikasi Teknologi Informasi*, 3(2), 79–87.
<https://doi.org/10.20885/snati.v3.i2.35>

Noor, M. M. F. M., Maksudi, & Gunawan, H. (2022). SISTEM INFORMASI ADMINISTRASI KEGIATAN ORGANISASI MAHASISWA (STUDI KASUS: UNIVERSITAS MUHAMMADIYAH CIREBON). *JURNAL INTI TALIFA*, 14(02), 22–28. <https://doi.org/https://doi.org/10.32534/int.v14i2.3766>

Riswanto, E., & Arayaji, R. G. (2024). IMPLEMENTASI SISTEM MANAJEMEN EVENT TERINTEGRASI UNTUK MENINGKATKAN EFEKTIVITAS KEGIATAN ORGANISASI MAHASISWA. *Prosiding Seminar Nasional Aplikasi Sains & Teknologi (SNAST)*, November, 166–175.

Sihombing, D. O. (2023). PERANCANGAN SISTEM INFORMASI PENGELOLAAN KEGIATAN HIMPUNAN MAHASISWA MENGGUNAKAN SYSTEM DEVELOPMENT LIFE CYCLE. *Technology, Business and Entrepreneurship*, 1(1), 30–41.
<https://doi.org/https://doi.org/10.61245/techbus.v1i1.5>

Sihotang, E. T., & Yutanto, H. (2021). Tata Kelola Organisasi Mahasiswa Melalui Pengembangan Sistem Informasi. *Matrik: Jurnal Manajemen, Teknik Informatika, Dan Rekayasa Komputer*, 21(1), 99–110.
<https://doi.org/10.30812/matrik.v21i1.1391>

Wiguna, K. B. P., Wijaya, I. N. Y. A., & Suyasa, I. P. B. (2026). Rancang Bangun Sistem Informasi Organisasi Mahasiswa Berbasis Website dengan Metode Extreme Programming pada Universitas Primakara. *Remik: Riset Dan E-Jurnal Manajemen Informatika Komputer*, 10(1), 342–350.
<https://doi.org/http://doi.org/10.33395/remik.v10i1.15844>

DAFTAR LAMPIRAN

Lampiran A. Dokumen Proses Pengumpulan *Requirement*

1. Kuesioner Pengurus Organisasi

Kuesioner ini disebarakan kepada 6 orang pengurus organisasi yang pernah terlibat dalam pengelolaan acara. Pengumpulan data dilakukan secara daring menggunakan Google Form pada tanggal 21 Februari 2026.

Skala penilaian yang digunakan adalah Skala Likert 5 poin dengan keterangan sebagai berikut:

SS = Sangat Setuju

S = Setuju

N = Netral

TS = Tidak Setuju

STS = Sangat Tidak Setuju

Tabel 1 Lampiran Hasil Kuesioner Pengurus Organisasi

No	Pertanyaan	SS	S	N	TS	STS
1.	Pendaftaran panitia event di organisasi saya masih dilakukan secara manual (chat, formulir kertas, atau Google Form terpisah)	2	2	1	1	
2.	Proses seleksi dan pembagian divisi panitia event masih dilakukan secara manual	3	1	1	1	
3.	Proses seleksi dan pembagian divisi panitia sering tidak terdokumentasi dengan baik		2	4		
4.	Absensi panitia dalam rapat atau kegiatan event masih dilakukan secara manual.	3	3			
5.	Data kehadiran panitia sulit di rekap dan ditelusuri kembali.	1	2	3		
6.	Notulensi rapat panitia sering tercecer atau tidak terdokumentasi secara rapi	1	1	4		

7.	Sertifikat panitia event masih dibuat dan dibagikan secara manual	3		1	2	
8.	Panitia sering mengalami keterlambatan atau kesalahan dalam penerbitan sertifikat.	1	2	3		
9.	Apakah menurut Anda organisasi membutuhkan sistem informasi khusus untuk mengelola kepanitiaan acara?	5		1		

2. Kuesioner Panitia

Kuesioner ini disebarakan kepada 6 orang panitia yang pernah mendaftarkan diri sebagai panitia acara yang dilaksanakan oleh organisasi. Pengumpulan data dilakukan secara daring menggunakan Google Form pada tanggal 21 Februari 2026.

Skala penilaian yang digunakan adalah Skala Likert 5 poin dengan keterangan sebagai berikut:

SS = Sangat Setuju

S = Setuju

N = Netral

TS = Tidak Setuju

STS = Sangat Tidak Setuju

Tabel 2 Lampiran Hasil Kuesioner Pengurus Organisasi

No	Pertanyaan	SS	S	N	TS	STS
1.	Pendaftaran panitia event di organisasi saya masih dilakukan secara manual (chat, formulir kertas, atau Google Form terpisah)	4	2			
2.	Proses seleksi dan pembagian divisi panitia event masih dilakukan secara manual	4	2			
3.	Sertifikat panitia event masih dibuat dan dibagikan secara	4	2			

	manual (melalui drive atau grup WhatsApp)					
4.	Selama menjadi panitia, apakah Anda pernah mengalami kesulitan dalam mendapatkan informasi atau pembagian tugas?	2	1	3		
5.	Apakah menurut Anda sistem informasi kepanitiaan diperlukan untuk membantu pekerjaan panitia?	3	2	1		

Lampiran B. Link Produk

Repository GitHub : <https://github.com/Niatii/Tugas-Akhir-Niati.git>

Lampiran C. Dokumen Pengujian

Lampiran Dokumen Pengujian Kebutuhan Fungsional : [Dokumentasi Pengujian Fungsional.pdf](#)

Lampiran Dokumen Pengujian UAT :
[Dokumentasi Pengujian UAT.pdf](#)