

**PENGEMBANGAN SISTEM ABSENSI BERBASIS QR
CODE UNTUK KEGIATAN RESES DPRD KOTA
BATAM**

PROYEK AKHIR

Disusun Oleh:

Farissa Elfirayanti

3312311094

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 ke hadirat Allah SWT karena atas rahmat dan karunia-Nya, penulis dapat menyelesaikan Tugas Karya Akhir yang berjudul “Pengembangan Sistem Absensi Berbasis QR Code untuk kegiatan Reses DPRD Kota Batam”. Tugas akhir ini disusun sebagai salah satu syarat untuk memperoleh gelar Ahli Madya pada Program Studi Diploma III Teknik Informatika, Jurusan Teknik Informatika, Politeknik Negeri Batam.

Proyek akhir ini bertujuan untuk mengembangkan sistem absensi berbasis QR Code yang dapat mendukung proses pencatatan kehadiran, pengelolaan data peserta, serta penyusunan laporan kegiatan reses DPRD Kota Batam Secara lebih terstruktur, terdokumentasi, dan mudah dikelola. Sistem ini diharapkan dapat menjadi solusi digital yang membantu Sekretariat DPRD Kota Batam dalam mengatasi kendala pencatatan kehadiran yang masih dilakukan secara manual.

Dalam perjalanan penyusunan Proyek Akhir ini, penulis menyadari bahwa keberhasilan ini tidak terlepas dari doa, dukungan, bimbingan, dan arahan dari berbagai pihak. Oleh karena itu, melalui kata pengantar ini, penulis ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. **Bapak Ir. Bambang Hendrawan, S.T., M.S.M., C.I.P.M.P., C.I.S.C.P.,** selaku Rektor Politeknik Negeri Batam, atas kebijakan, dukungan, serta fasilitas yang telah diberikan selama penulis menempuh pendidikan dan menyelesaikan Tugas Akhir ini.
2. **Bapak Ir. Ahmad Hamim Thohari, S.S.T., M.T.,** selaku Kepala Jurusan Teknik Informatika, atas arahan, dukungan, dan fasilitas akademik yang telah diberikan.
3. **Bapak Agus Fatulloh, S.T., M.T.,** selaku Dosen Pembimbing yang telah meluangkan waktu, tenaga, dan pikiran untuk memberikan bimbingan, arahan, motivasi, serta dukungan yang sangat berharga sejak awal hingga akhir penyusunan Proyek Akhir ini.

4. **Ibu Yeni Rokhayati, S.Si., M.Sc.**, selaku Ketua Program Studi Teknik Informatika, atas fasilitas dan dukungan akademik selama masa studi.
5. **Bapak Ir. Ahmad Hamim Thohari, S.S.T., M.T.** dan **Bapak Nur Cahyono Kushardianto, S.Si., M.T., M.Sc., Ph.D.**, selaku Dosen Penguji yang telah memberikan masukan, kritik, dan saran yang konstruktif untuk penyempurnaan Proyek Akhir ini.
6. **Sekretariat DPRD Kota Batam** beserta seluruh staf yang telah memberikan izin, data, serta Informasi yang diperlukan selama proses pengembangan sistem.
7. **Kedua orang tua dan keluarga tercinta**, atas segala doa yang tidak pernah putus, dukungan moral dan material, serta kasih sayang yang tulus tanpa syarat. Terima kasih atas setiap tetes keringat dan pengorbanan yang telah di berikan, yang menjadi landasan kekuatan dan motivasi terbesar bagi penulis dalam menyelesaikan pendidikan ini.
8. **Sahabat dan rekan-rekan terdekat, terutama teman-teman dari grup 5 Pendekar Pedang x 4 Ratu Pantai Selatan** — Siti Zahwa, Nailah Dipa, Puput N, Raja Ulka, Rivaldo A, Leonardo, Al Ayubi, dan Viodama yang senantiasa memberikan semangat, dukungan, dan kebersamaan selama masa perkuliahan. Terima kasih karena telah menjadi saksi bisu perjalanan penulis mulai dari hari-hari yang penuh tawa hingga malam-malam yang dihiasi kegelisahan. Kalian bukan sekadar teman, tetapi bagian dari cerita hidup yang tidak akan pernah pudar. Semoga persahabatan ini tetap terjaga, meski waktu dan jarak suatu hari nanti memisahkan kita.
9. **Meliana**, sahabat terdekat penulis, yang selalu hadir tanpa syarat. Terima kasih karena telah menjadi tempat berpulang saat dunia terasa asing, menjadi bahu untuk bersandar saat air mata tak terbendung, dan menjadi suara yang selalu berkata "kamu bisa" saat penulis sendiri mulai meragukan dirinya. Kehadiranmu adalah bukti bahwa Tuhan masih baik Dia mengirimkan malaikat dalam wujud sahabat seperti dirimu. Tanpamu, perjalanan ini mungkin terasa jauh lebih berat. Terima kasih telah menjadi rumah kedua bagi penulis.

10. **Karya dan inspirasi dari NCT 127 dan NCT Dream, terutama Mark Lee dan Lee Haechan.** yang telah memberikan energi positif dan motivasi bagi penulis selama proses pengerjaan Proyek Akhir ini.
11. **Semua pihak** yang tidak dapat disebutkan satu per satu, yang telah memberikan dukungan dalam bentuk apapun.
12. ***Last but not least, Penulis sendiri,*** yang telah berjuang untuk menyelesaikan apa yang telah dimulai bertahan hingga titik ini, dan tetap berusaha meski sering kali di hadapkan pada rasa putus asa. Terima kasih kepada diri sendiri atas ketekunan dan semangat pantang menyerah. Penulis bersyukur kepada Tuhan atas kekuatan yang diberikan sehingga dapat menjadi pribadi yang mandiri dan mencapai pencapaian ini.

Penulis menyadari sepenuhnya bahwa Proyek Akhir ini masih jauh dari kata sempurna dan masih memiliki banyak keterbatasan. Oleh karena itu, penulis dengan rendah hati dan tangan terbuka menerima segala bentuk kritik dan saran yang membangun dari semua pihak untuk perbaikan yang di masa yang akan datang.

Akhir kata, penulis berharap semoga Proyek Akhir ini dapat memberikan manfaat dan kontribusi positif, baik bagi pengembangan ilmu pengetahuan di bidang Teknologi Informasi, khususnya dalam penerapan sistem absensi digital berbasis *QR Code*, maupun bagi DPRD Kota Batam dan masyarakat luas pada umumnya. Penulis juga berdoa semoga langkah kecil ini selalu di perkuat dan menjadi awal dari perjalanan yang lebih baik di masa mendatang.

Batam, Juli 2026



Farissa Elfirayanti

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	iv
DAFTAR TABEL	vi
DAFTAR GAMBAR.....	vii
DAFTAR LAMPIRAN	ix
ABSTRAK	x
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2 Rumusan Masalah	2
1.3. Tujuan	3
1.4. Batasan Masalah.....	3
1.5. Manfaat	4
1.6. Kolaborasi dan Sinergi dalam Pelaksanaan Proyek	5
BAB II TINJAUAN PUSTAKA.....	6
2.1. Penelitian Terkait	6
2.2. Landasan Teori.....	8
2.2.1 Absensi Digital.....	8
2.2.2 <i>QR code</i>	9
2.2.3 Aplikasi Berbasis Web (<i>Web-Based Application</i>)	10
2.2.4 <i>Framework Laravel</i>	10
2.2.5 <i>Database MySQL</i>	10
2.2.6 <i>Bring Your Own Device (BYOD)</i>	11
2.2.7 Kegiatan <i>Reses</i> DPRD	11
2.2.8 Akuntabilitas Publik.....	12
2.3 Metode Pengembangan Produk.....	12
BAB III ANALISIS DAN PERANCANGAN	15
3.1. Analisis Kebutuhan	15
3.1.1 Proses Bisnis Berjalan.....	15

3.1.2 Gambaran Umum <i>Sistem</i>	16
3.2 Perancangan	17
3.2.1 Kebutuhan <i>Fungsional</i>	17
3.2.2 Kebutuhan Non-Fungsional	19
3.2.3 Diagram <i>Use Case</i>	19
3.2.4 Scenario <i>Use Case</i>	21
3.2.5 <i>Activity Diagram</i>	37
3.2.6 <i>ER Diagram</i>	51
3.2.7 Perancangan Antarmuka (<i>Wireframe</i>)	53
BAB IV IMPLEMENTASI DAN PENGUJIAN	59
4.1. Hasil Implementasi.....	59
4.1.1 Antarmuka Admin.....	59
4.1.2 Antarmuka Peserta Reses	65
4.1.3 Alur Penggunaan Sistem	67
4.2 Pengujian.....	67
4.2.1 Pengujian Fungsional (<i>Black-Box Testing</i>).....	68
4.2.2 Pengujian Kinerja Dasar (<i>Load Testing</i>).....	75
A. Spesifikasi Lingkungan Pengujian.....	75
B. Hasil Pengujian	76
C. Analisis Hasil Pengujian	80
4.2.3 Kesimpulan Pengujian	81
BAB V KESIMPULAN	83
5.1 Kesimpulan	83
5.2 Saran.....	84
DAFTAR PUSTAKA	85
DAFTAR LAMPIRAN	87

DAFTAR TABEL

Tabel 2.1 Perbandingan Penelitian Sebelumnya.....	7
Tabel 3.1 Kebutuhan <i>Fungsional</i>	17
Tabel 3.2 Kebutuhan <i>Non-Fungsional</i>	19
Tabel 3.3 Scenario <i>Use Case Login Admin</i>	21
Tabel 3.4 Scenario <i>Use Case Kelola Data Acara</i>	22
Tabel 3.5 Scenario <i>Use Case Generate & Refresh QR code</i>	23
Tabel 3.6 Scenario <i>Use Case Buka / Tutup Akses Absensi</i>	24
Tabel 3.7 Scenario <i>Use Case Scan QR code</i> oleh Peserta	25
Tabel 3.8 Scenario <i>Use Case Mengisi & Mengirim Form Absensi</i>	26
Tabel 3.9 Scenario <i>Use Case Pencatatan Waktu &</i>	27
Tabel 3.10 Scenario <i>Use Case Lihat Daftar Peserta & Statistik</i>	28
Tabel 3.11 Scenario <i>Use Case Unduh Laporan Kehadiran (PDF & CSV)</i>	29
Tabel 3.12 Scenario <i>Use Case Edit & Hapus Data Kehadiran</i>	30
Tabel 3.13 S Scenario <i>Use Case Pencatatan Activity Log</i>	31
Tabel 3.14 Scenario <i>Use Case Arsipkan Acara</i>	32
Tabel 3.15 Scenario <i>Use Cse Lihat & Restore Acara Terarsip</i>	33
Tabel 3.16 Scenario <i>Use Case Kelola Profil Admin</i>	35
Tabel 3.17 Scenario <i>Use Case Logout Admin</i>	36
Tabel 4.1 <i>Black-Box Testing</i>	68
Tabel 4.2 Rekapitulasi Hasil Pengujian Fungsional	74
Tabel 4.3 Spesifikasi Perangkat Keras.....	75
Tabel 4.4 Spesifikasi Perangkat Lunak.....	75
Tabel 4.5 Hasil Pengujian Administrator (50 User).....	76
Tabel 4.6 Hasil Pengujian Peserta (50 User)	77
Tabel 4.7 Hasil Pengujian Administrator (100 User)	78
Tabel 4.8 Hasil Pengujian Peserta (100 User)	79
Tabel 4.9 Anomali penurunan request	81

DAFTAR GAMBAR

Gambar 2.1 <i>QR code</i>	9
Gambar 2.2 Model <i>Waterfall</i>	13
Gambar 3.1 Gambaran Umum Sistem.....	17
Gambar 3.2 <i>Diagram Use Case</i>	20
Gambar 3.3 <i>Activity Diagram Login Admin</i>	37
Gambar 3.4 <i>Activity Diagram Kelola Data Acara</i>	38
Gambar 3.5 <i>Activity Diagram Generate & Refresh QR code</i>	39
Gambar 3.6 <i>Activity Diagram Buka / Tutup Akses Absensi</i>	40
Gambar 3.7 <i>Activity Diagram Scan QR code oleh Peserta</i>	41
Gambar 3.8 <i>Activity Diagram Mengisi & Mengirim Form Absensi</i>	42
Gambar 3.9 <i>Activity Diagram Pencatatan Waktu & Informasi Teknis</i>	43
Gambar 3.10 <i>Activity Diagram Lihat Rekap Kehadiran & Statistik</i>	44
Gambar 3.11 <i>Activity Diagram Unduh Laporan Kehadiran (PDF/CSV)</i>	45
Gambar 3.12 <i>Activity Diagram Hapus Peserta</i>	46
Gambar 3.13 <i>Activity Diagram Pencatatan Activity Log</i>	47
Gambar 3.14 <i>Activity Diagram Arsipkan Acara</i>	48
Gambar 3.15 <i>Activity Diagram Lihat & Restore Acara Terarsip</i>	49
Gambar 3.16 <i>Activity Diagram Kelola Profil Admin</i>	50
Gambar 3.17 <i>Activity Diagram Logout Admin</i>	50
Gambar 3.18 <i>ER Diagram</i>	51
Gambar 3.19 <i>Wireframe Halaman Beranda</i>	53
Gambar 3.20 <i>Wireframe Login Admin</i>	54
Gambar 3.21 <i>Wireframe Dashboard Admin</i>	54
Gambar 3.22 <i>Wireframe Kelola data acara</i>	55
Gambar 3.23 <i>Wireframe Tambah Acara Baru</i>	56
Gambar 3.24 <i>Wireframe QR code Absensi</i>	56
Gambar 3.25 <i>Wireframe Rekap kehadiran</i>	57
Gambar 3.26 <i>Wireframe Laporan akhir</i>	58
Gambar 3.27 <i>Wireframe Rekap Kehadiran Peserta (Laporan Cetak / PDF)</i>	58

Gambar 4.1 Halaman <i>Login</i>	59
Gambar 4.2 Halaman <i>Dashboard</i>	60
Gambar 4.3 Halaman Daftar Acara.....	61
Gambar 4.4 Halaman Buat Acara Baru.....	61
Gambar 4.5 Halaman <i>QR code</i> Acara	62
Gambar 4.6 Halaman Rekap Kehadiran.....	63
Gambar 4.7 Halaman Activity Log.....	63
Gambar 4.8 Halaman Arsip Acara	64
Gambar 4.9 Halaman Profil Admin	64
Gambar 4.10 Halaman Form Absensi Peserta	65
Gambar 4.11 Halaman Konfirmasi Absensi Berhasil.....	66
Gambar 4.12 Halaman <i>QR code</i> Tidak Valid / Acara Berakhir.....	66
Gambar 4.13 Alur Penggunaan Sistem Secara Keseluruhan	67
Gambar 4.14 Hasil <i>Load Testing</i> Administrator (50 User) pada Locus	77
Gambar 4.15 Hasil <i>Load Testing</i> Administrator (50 User) pada Locust	78
Gambar 4.16 Hasil <i>Load Testing</i> Administrator (100 User) pada Locust	79
Gambar 4.17 Hasil <i>Load Testing</i> Peserta (100 User) pada Locust	80

DAFTAR LAMPIRAN

Lampiran A. Dokumen Proses Pengumpulan Requirement	87
Lampiran B. Link Produk	87
Lampiran C. Hasil Pengujian	87

ABSTRAK

Penyelenggaraan sesi jeda selama rapat Dewan Kota Batam memerlukan pencatatan kehadiran peserta yang tepat dan terdokumentasi dengan baik. Namun, prosedur pencatatan kehadiran manual yang berlaku saat ini berisiko menimbulkan masalah seperti pencatatan yang tidak efisien, kesalahan entri data, catatan kehadiran ganda, dan kesulitan dalam membuat laporan. Tujuan tesis ini adalah merancang dan mengimplementasikan sistem pencatatan kehadiran berbasis kode QR untuk rapat Dewan Kota Batam guna mendukung pencatatan kehadiran secara digital. Model waterfall digunakan sebagai metodologi pengembangan, yang terdiri dari fase analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan. Sistem ini dikembangkan menggunakan kerangka kerja Laravel dan basis data MySQL. Fitur-fitur utamanya meliputi manajemen acara, pembuatan kode QR, pelacakan kehadiran peserta, validasi untuk mencegah entri kehadiran ganda, ringkasan data, dan pembuatan laporan. Hasil pengujian fungsional menggunakan metode pengujian black-box pada 52 kasus uji menunjukkan tingkat keberhasilan 100%. Selain itu, pengujian kinerja menggunakan Locust dengan skenario yang melibatkan masing-masing 50 dan 100 pengguna virtual menunjukkan bahwa semua permintaan diproses tanpa kesalahan (tingkat kegagalan 0%), meskipun waktu respons rata-rata sebesar 21–94 detik sangat buruk, yang berarti sistem ini belum cocok untuk penerapan di dunia nyata. Hasil proyek menunjukkan bahwa sistem yang dikembangkan mampu mengelola pelacakan kehadiran selama jeda sesi dengan cara yang lebih terstruktur, terdokumentasi, dan lebih sederhana dibandingkan dengan pelacakan manual.

Kata Kunci: *QR code, Sistem Absensi, Laravel, Reses DPRD, Load Testing*

BAB I PENDAHULUAN

1.1. Latar Belakang

Perkembangan teknologi informasi telah mendorong transformasi digital di berbagai sektor, termasuk dalam administrasi publik. Sistem pencatatan kehadiran yang sebelumnya dilakukan secara manual kini mulai memanfaatkan teknologi digital untuk mendukung proses pencatatan, penyimpanan, dan pengelolaan data secara lebih terstruktur. Salah satu teknologi yang banyak digunakan adalah *Quick Response Code (QR code)*, yang memungkinkan proses pencatatan kehadiran dilakukan secara cepat dan terdokumentasi melalui perangkat digital (Sitorus, Jamaluddin, & Harianja, 2023).

Penelitian ini berfokus pada kegiatan reses yang diselenggarakan oleh Dewan Perwakilan Rakyat Daerah (DPRD) Kota Batam. Reses merupakan kegiatan yang dilaksanakan oleh anggota DPRD di daerah pemilihannya untuk menyerap aspirasi masyarakat serta menjalin komunikasi dengan konstituen. Dalam pelaksanaannya, data kehadiran peserta menjadi salah satu dokumen pendukung administrasi dan penyusunan laporan kegiatan. Oleh karena itu, pencatatan kehadiran yang baik diperlukan agar data peserta dapat terdokumentasi dengan rapi dan mudah dikelola.

Berdasarkan hasil wawancara dengan salah satu staf Sekretariat DPRD Kota Batam yang terlibat dalam pengelolaan kegiatan reses, DPRD Kota Batam memiliki 50 anggota dewan yang terbagi ke dalam enam daerah pemilihan (dapil). Setiap anggota dewan melaksanakan kegiatan reses pada beberapa lokasi yang berbeda sesuai dengan jadwal yang telah ditentukan. Kondisi tersebut menyebabkan proses pencatatan dan pengelolaan data kehadiran menjadi lebih kompleks apabila masih dilakukan secara manual menggunakan daftar hadir.

Hasil wawancara juga menunjukkan bahwa pencatatan kehadiran peserta masih dilakukan secara manual menggunakan daftar hadir. Proses tersebut dinilai kurang praktis ketika jumlah peserta kegiatan cukup banyak karena data harus dicatat dan dikelola secara manual. Selain itu, tidak adanya mekanisme verifikasi

pada proses pencatatan berpotensi menyebabkan terjadinya duplikasi data. Dokumen kehadiran yang masih berbentuk fisik juga berisiko hilang atau rusak sehingga dapat menyulitkan proses pengelolaan dan penyusunan laporan kegiatan.

Kebutuhan akan pemanfaatan teknologi dalam proses pencatatan kehadiran sebenarnya telah disadari oleh pihak Sekretariat DPRD Kota Batam. Berdasarkan hasil wawancara, penggunaan teknologi *barcode* atau *QR code* pernah diusulkan sebagai alternatif untuk membantu proses pencatatan kehadiran peserta secara lebih terstruktur. Hal tersebut menunjukkan adanya kebutuhan terhadap sistem yang dapat mendukung proses administrasi kegiatan reses secara digital.

Sebagai solusi, penelitian ini mengembangkan sistem absensi berbasis *QR code* yang dirancang untuk membantu proses pencatatan kehadiran dan pengelolaan data peserta kegiatan reses. Sistem yang dikembangkan berbasis web dan menerapkan konsep *Bring Your Own Device* (BYOD), sehingga peserta dapat melakukan absensi menggunakan *smartphone* pribadi tanpa memerlukan instalasi aplikasi tambahan.

Melalui penelitian ini diharapkan dapat dihasilkan sistem absensi berbasis *QR code* yang mendukung proses pencatatan kehadiran, pengelolaan data peserta, serta penyusunan laporan kegiatan reses DPRD Kota Batam secara lebih terstruktur, terdokumentasi, dan mudah dikelola dibandingkan dengan proses pencatatan manual.

1.2 Rumusan Masalah

Berdasarkan identifikasi permasalahan pada latar belakang, rumusan masalah dalam penelitian ini adalah:

1. Bagaimana merancang dan membangun sistem absensi berbasis *QR code* yang dapat digunakan untuk mendukung proses pencatatan kehadiran pada kegiatan reses DPRD Kota Batam?
2. Bagaimana penerapan mekanisme validasi pada sistem absensi berbasis *QR code* dapat membantu mencegah terjadinya duplikasi data kehadiran peserta kegiatan reses DPRD Kota Batam?

3. Bagaimana sistem absensi berbasis *QR code* dapat mendukung penyelenggaraan kegiatan reses yang lebih tertib melalui pencatatan data kehadiran yang terstruktur dan penyajian laporan secara otomatis?

1.3. Tujuan

Berdasarkan rumusan masalah yang telah diidentifikasi, tujuan dari penelitian ini adalah sebagai berikut:

1. Untuk merancang dan mengimplementasikan sistem absensi berbasis *QR code* yang dapat digunakan dalam proses pencatatan kehadiran pada kegiatan reses DPRD Kota Batam.
2. Untuk menerapkan mekanisme validasi data kehadiran guna membantu mencegah terjadinya duplikasi data peserta pada kegiatan reses DPRD Kota Batam.
3. Untuk menyediakan fitur pelaporan kehadiran yang terstruktur dan dapat dihasilkan secara otomatis guna mendukung pengelolaan data kehadiran kegiatan reses DPRD Kota Batam.

1.4. Batasan Masalah

Agar penelitian lebih terarah dan fokus, proyek akhir ini dibatasi pada:

1. Sistem yang dikembangkan difokuskan pada pengelolaan absensi peserta kegiatan reses DPRD Kota Batam, meliputi pengelolaan acara, pencatatan kehadiran, rekapitulasi data, dan pembuatan laporan.
2. Sistem dikembangkan berbasis web menggunakan *Framework Laravel* dan basis data *MySQL*, serta dapat diakses melalui perangkat yang memiliki *Browser* dan koneksi internet.
3. Proses absensi dilakukan melalui pemindaian *QR code* yang telah dibuat untuk setiap kegiatan reses.
4. Validasi kehadiran peserta menggunakan nomor telepon dan tidak mencakup validasi identitas resmi seperti NIK, biometrik, maupun validasi lokasi berbasis *GPS*.

5. Pengujian sistem dibatasi pada *Black-Box Testing* dan *Load Testing* menggunakan Locust dengan simulasi pengguna sesuai kebutuhan kegiatan reses.

1.5. Manfaat

a. Manfaat Teoritis

1. Menambah referensi mengenai pengembangan sistem absensi berbasis *QR code* pada aplikasi berbasis web.
2. Memberikan gambaran penerapan teknologi *QR code* dalam mendukung pencatatan kehadiran dan pengelolaan data kegiatan.

b. Manfaat Praktis

1. Sistem ini diharapkan dapat membantu proses pencatatan kehadiran, pengelolaan data peserta, dan penyusunan laporan kegiatan secara digital.
2. Sistem memberikan kemudahan bagi peserta untuk melakukan absensi melalui pemindaian *QR code* menggunakan perangkat pribadi.
3. Sistem dapat mendukung penyimpanan data kehadiran secara lebih terstruktur dan terdokumentasi.

c. Kontribusi terhadap Tujuan Pembangunan Berkelanjutan (SDGs)

Proyek pengembangan sistem absensi berbasis *QR code* ini memberikan kontribusi terhadap pencapaian Sustainable Development Goals (SDGs) yang dicanangkan oleh PBB, khususnya pada dua tujuan berikut:

1. SDG 9: Industry, Innovation and Infrastructure

Sistem ini mendukung pemanfaatan teknologi informasi melalui pengembangan layanan digital berbasis web yang memanfaatkan teknologi *QR code* dan menerapkan konsep *Bring Your Own Device* (BYOD). Pendekatan ini memungkinkan peserta reses mengakses sistem menggunakan perangkat pribadi tanpa instalasi aplikasi, sehingga mendukung akses universal terhadap teknologi informasi.

2. SDG 16: Peace, Justice and Strong Institutions

Sistem ini mendukung terciptanya tata kelola pemerintahan yang transparan dan akuntabel melalui:

- Pencatatan kehadiran digital dengan *timestamp* otomatis sebagai bukti akuntabilitas
- Pencegahan duplikasi data melalui validasi nomor telepon
- Pencatatan *activity log* untuk mendukung proses audit
- Penyediaan data kehadiran yang terstruktur untuk mendukung pengambilan keputusan yang lebih tepat

1.6. Kolaborasi dan Sinergi dalam Pelaksanaan Proyek

Proyek Akhir ini dilaksanakan secara individu dengan melibatkan beberapa pihak dalam proses pengembangannya. Kolaborasi dilakukan dengan staf Sekretariat DPRD Kota Batam untuk memperoleh informasi mengenai proses absensi kegiatan reses dan kebutuhan sistem yang akan dikembangkan. Selain itu, konsultasi secara berkala dengan dosen pembimbing dilakukan untuk memperoleh arahan terkait metodologi, pengembangan sistem, pengujian, dan penyusunan laporan.

Melalui kolaborasi tersebut, sistem yang dikembangkan diharapkan dapat sesuai dengan kebutuhan pengguna serta tujuan yang telah ditetapkan dalam Proyek Akhir.

BAB II TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Tinjauan pustaka menunjukkan bahwa pengembangan sistem absensi berbasis *QR code* telah banyak dilakukan, namun mayoritas masih terbatas pada lingkungan tertutup dengan pengguna yang telah terdaftar sebelumnya (*pre-registered*). Alfikri dkk. (2024) berhasil mengembangkan sistem E-Absensi berbasis *web* di Kantor Desa Binamang dengan tingkat akurasi mencapai 98%. Namun, sistem tersebut hanya dapat diakses oleh pegawai desa yang datanya telah tersedia dalam basis data. Hal ini menimbulkan keterbatasan jika diterapkan pada acara publik yang pesertanya berasal dari masyarakat umum tanpa data awal. Demikian pula, penelitian Khaerudin dkk. (2022) di Dinas Pendidikan Kota Bekasi berhasil menggantikan sistem *Fingerprint* dengan *QR code* yang lebih efisien, namun lingkupnya tetap pada pegawai internal instansi.

Penelitian dengan pendekatan berbeda dilakukan oleh Djamarullah dkk. (2024) yang mengusulkan konsep *Bring Your Own Device* (BYOD). Meski menawarkan solusi ekonomis, penelitian ini masih bersifat konseptual dan belum diimplementasikan secara riil, khususnya pada skala acara pemerintahan. Di sisi lain, Eliza Natalia dkk. (2023) mengembangkan aplikasi *Android* untuk absensi siswa yang cepat (5-10 detik) dengan validasi *GPS*. Kendati efisien, sistem ini memerlukan instalasi aplikasi khusus dan hanya berlaku bagi siswa yang telah terdaftar sebelumnya.

Berdasarkan analisis tersebut, teridentifikasi dua pola keterbatasan penelitian terdahulu yang relevan dengan konteks acara publik seperti reses DPRD: (1) ketergantungan pada basis data pengguna yang telah ada, dan (2) persyaratan instalasi perangkat lunak khusus. Oleh karena itu, penelitian ini mengusulkan sistem alternatif yang dirancang khusus untuk mengatasi keterbatasan tersebut. Sistem yang dikembangkan tidak memerlukan data awal pengguna maupun instalasi aplikasi. Mekanisme kerjanya dirancang sederhana: peserta cukup

memindai *QR code* yang disediakan menggunakan kamera *smartphone* pribadi. Setelah pemindaian, akan muncul formulir konfirmasi kehadiran langsung pada *Browser*. Pengisian formulir ini sekaligus berfungsi sebagai proses registrasi dan validasi kehadiran secara *real-time*. Pendekatan ini dinilai sesuai untuk menangani partisipasi masyarakat yang bersifat spontan dan dinamis dalam kegiatan publik.

Tabel 2.1 Perbandingan Penelitian Sebelumnya

Peneliti/Tahun	Judul/Topik Penelitian	Metode/Teknologi	Kelebihan	Kekurangan/Celah
Alfikri, Gusman, & Azrialdi (2024)	E-Absensi Menggunakan <i>QR Code</i> Berbasis Web di Kantor Desa Binamang	<i>Website, Laravel, MySQL, Waterfall</i>	Berbasis web, akurat (98%), rekapitulasi cepat (15-20 menit)	Hanya untuk pegawai tetap, tidak untuk <i>Event</i> publik
Djamarullah, Nuryasin, & Wibowo (2024))	Perancangan Sistem Absensi <i>QR code</i> Menggunakan Konsep BYOD	<i>Website, BYOD, QR code</i>	Belum implementasi skala pemerintahan	Fokus pada pegawai, tidak mencakup kegiatan massal
Khaerudin, Sumantri, Supriatna, & Ritzkal (2022))	Pengembangan Sistem Informasi Absensi Pegawai Menggunakan <i>QR code</i> di Dinas Pendidikan Kota Bekasi	<i>Website, QR code, PHP</i>	Efisien, ganti <i>Fingerprint</i> , data terpusat	Fokus pegawai tetap, tidak untuk acara massal
Eliza Natalia et al. (2023))	Sistem Informasi Kehadiran Siswa Menggunakan <i>QR code</i> Berbasis Android	Android, QR Scanner, GPS, SQLite	Cepat (5-10 detik), validasi GPS, <i>real-time</i>	Android (perlu install), data siswa pre-registered

Farissa Elf- irayanti (2026)	Pengembangan Sistem Absensi Berbasis <i>QR code</i> untuk Kegiatan Reses DPRD Kota Batam	<i>Laravel</i> , <i>MySQL</i> , <i>QR</i> <i>code</i> , <i>Water-</i> <i>fall</i>	Mendukung pengelolaan acara, absensi pe- serta, rekapitulasi data, dan laporan berbasis web	Belum menggunakan validasi NIK, GPS, atau OTP
------------------------------------	---	--	--	---

2.2. Landasan Teori

Pada bab ini, teori dasar mengenai Absensi Digital, *QR code*, dibahas secara singkat.

2.2.1 Absensi Digital

Absensi digital merupakan metode pencatatan kehadiran yang memanfaatkan teknologi informasi sebagai pengganti sistem manual berbasis kertas. Sistem ini memungkinkan proses pencatatan, penyimpanan, dan pengelolaan data kehadiran dilakukan secara digital sehingga data dapat terdokumentasi dengan lebih terstruktur dan mudah dikelola (Alfikri, Gusman, & Azrialdi, 2024).

1. **Absensi *Fingerprint***, menggunakan sidik jari sebagai identitas unik pengguna.
2. **Absensi *Face Recognition***, memanfaatkan teknologi pengenalan wajah untuk melakukan verifikasi kehadiran.
3. **Absensi *RFID***, menggunakan kartu yang memanfaatkan teknologi *Radio Frequency Identification*.

Absensi berbasis *QR code* banyak digunakan karena dapat diakses menggunakan *smartphone* pribadi tanpa memerlukan perangkat keras khusus. Selain itu, metode ini sesuai untuk kegiatan yang melibatkan peserta yang dinamis dan tidak memerlukan proses instalasi aplikasi tambahan pada perangkat pengguna (Hamdani, Wibowo, & Heryono, 2021).

2.2.2 QR code



Gambar 2.1 *QR code*

QR code (*Quick Response Code*) merupakan representasi dari evolusi teknologi kode batang, yang mengalami transformasi dari format linier menjadi pola matriks dua dimensi, yang memungkinkan penyimpanan data dengan kompleksitas lebih tinggi dan kapasitas yang lebih besar. *QR code* dapat menyimpan beragam jenis data, termasuk teks, URL, nomor telepon, hingga koordinat lokasi, dengan kapasitas maksimum mencapai 7.089 karakter numerik atau 4.296 karakter alfanumerik (Herlina, L., & Hidayatulloh, M. K. (2016)). Penggunaan *QR code* dalam sistem absensi memberikan kemudahan dalam proses pencatatan kehadiran, karena karakteristiknya yang lebih cepat, praktis, dan dapat mengurangi risiko kecurangan dibandingkan dengan metode manual (Hamdani, M., Wibowo, A., & Heryono, H. (2021)). Proses pemindaian *QR code* hanya memerlukan kamera *smartphone*, yang saat ini umumnya dimiliki oleh sebagian besar masyarakat, sehingga tidak diperlukan investasi pada perangkat tambahan.

Spesifikasi Teknis *QR code*:

1. **Kapasitas Data** – Mampu menyimpan data dalam jumlah besar dengan ukuran fisik yang kecil
2. **Error Correction Level** – Memiliki 4 tingkat koreksi kesalahan (L=7%, M=15%, Q=25%, H=30%) yang memungkinkan *QR code* tetap terbaca meskipun sebagian rusak
3. **Kecepatan Scanning** – Dapat dipindai dengan cepat dari berbagai sudut dan jarak
4. **Keamanan** – Dapat dienkripsi untuk mencegah pemalsuan dan penyalahgunaan

Dalam penelitian ini, *QR code* digunakan sebagai metode utama untuk mengidentifikasi kegiatan *reses* tertentu. Setiap acara memiliki *QR code* unik yang di-generate oleh sistem dengan masa berlaku sesuai durasi acara.

2.2.3 Aplikasi Berbasis Web (*Web-Based Application*)

Aplikasi berbasis web adalah perangkat lunak yang dapat diakses melalui web *Browser* tanpa perlu instalasi khusus di perangkat pengguna (Wei, Manori, Devnath, Pasi, & Kumar, 2017). Keunggulan aplikasi berbasis web meliputi aksesibilitas tinggi dari berbagai perangkat (*desktop*, *laptop*, *smartphone*), tidak perlu instalasi, bersifat *cross-platform*, data tersimpan terpusat di *server*, pembaruan sistem mudah, dan biaya pengembangan lebih rendah. Aplikasi web umumnya menggunakan arsitektur *Client-Server* dengan pola *Model-View-Controller* (MVC) yang memisahkan logika bisnis, tampilan, dan kontrol aplikasi. Pemilihan aplikasi berbasis web untuk sistem absensi kegiatan *reses* DPRD sangat tepat karena peserta tidak perlu persiapan teknis dan dapat langsung mengakses sistem hanya dengan memindai *QR code*.

2.2.4 Framework Laravel

Laravel adalah *framework PHP* yang populer untuk pengembangan aplikasi web modern, dikembangkan oleh Taylor Otwell dengan filosofi kesederhanaan dan produktivitas tinggi (Laravel LLC, 2024). Fitur utama *Laravel* yang mendukung proyek ini meliputi Eloquent *ORM* untuk interaksi dengan *database*, Blade Template Engine untuk membangun antarmuka dinamis, routing yang fleksibel, middleware untuk autentikasi dan validasi, migration dan seeding untuk pengelolaan *database*, sistem validasi built-in, serta fitur keamanan terhadap SQL Injection, XSS, dan CSRF (Laravel LLC, 2024). *Laravel* dipilih karena memiliki dokumentasi lengkap, komunitas besar, ekosistem matang, performa baik, dan cocok untuk pengembangan sistem absensi dengan fitur CRUD dan reporting (Laravel LLC, 2024).

2.2.5 Database MySQL

MySQL adalah sistem manajemen basis data relasional (*RDBMS*) yang paling populer, bersifat *open-source*, dan dikembangkan oleh Oracle Corporation. *MySQL* memiliki karakteristik *open source*, performa tinggi, keamanan dengan enkripsi data, *scalability*, kompatibilitas dengan berbagai bahasa pemrograman, dan *ACID compliance* untuk menjamin konsistensi data. *MySQL* dipilih karena integrasi sempurna dengan *Laravel* melalui *Eloquent ORM*, performa optimal untuk transaksi banyak, mudah dalam *backup* dan *recovery* data, cocok untuk sistem yang memerlukan integritas data tinggi, dan banyak tersedia *Hosting* dengan biaya terjangkau. Dalam sistem ini, *Database* dirancang dengan prinsip normalisasi untuk menghindari redundansi dan memastikan integritas referensial antar tabel.

2.2.6 Bring Your Own Device (BYOD)

Bring Your Own Device (BYOD) merupakan suatu konsep di mana perangkat pribadi digunakan oleh pengguna untuk mengakses sistem atau aplikasi dalam konteks kegiatan tertentu. Keuntungan dari BYOD meliputi pengurangan biaya karena tidak diperlukan penyediaan perangkat khusus, fleksibilitas dalam akses, kemudahan implementasi tanpa perlu distribusi perangkat, serta kenyamanan bagi pengguna. Di sisi lain, tantangan yang dihadapi oleh BYOD mencakup masalah keamanan data, variasi perangkat yang harus kompatibel, dan ketergantungan pada koneksi internet yang dimiliki oleh pengguna. Sistem absensi ini dirancang dengan pendekatan berbasis web yang responsif, sehingga dapat diakses dari berbagai perangkat dengan ukuran layar yang berbeda. Peserta reses dapat menggunakan kamera *smartphone* pribadi mereka untuk memindai *QR code* tanpa perlu menginstal aplikasi, yang sejalan dengan karakteristik kegiatan yang melibatkan masyarakat umum dengan tingkat literasi teknologi yang bervariasi.

2.2.7 Kegiatan Reses DPRD

Reses merupakan kegiatan yang dilaksanakan oleh anggota DPRD di daerah pemilihannya untuk menyerap aspirasi masyarakat serta menjalin komunikasi dengan konstituen. Dalam pelaksanaannya, data kehadiran peserta menjadi salah

satu dokumen pendukung administrasi dan penyusunan laporan kegiatan. Berdasarkan hasil wawancara dengan staf Sekretariat DPRD Kota Batam, pencatatan kehadiran peserta masih dilakukan secara manual menggunakan daftar hadir. Proses tersebut memerlukan pengelolaan data secara manual sehingga dinilai kurang praktis ketika jumlah peserta kegiatan cukup banyak. Oleh karena itu, penelitian ini mengembangkan sistem absensi berbasis *QR code* yang diharapkan dapat mendukung proses pencatatan kehadiran, pengelolaan data peserta, dan penyusunan laporan kegiatan reses DPRD Kota Batam secara lebih terstruktur dan terdokumentasi.

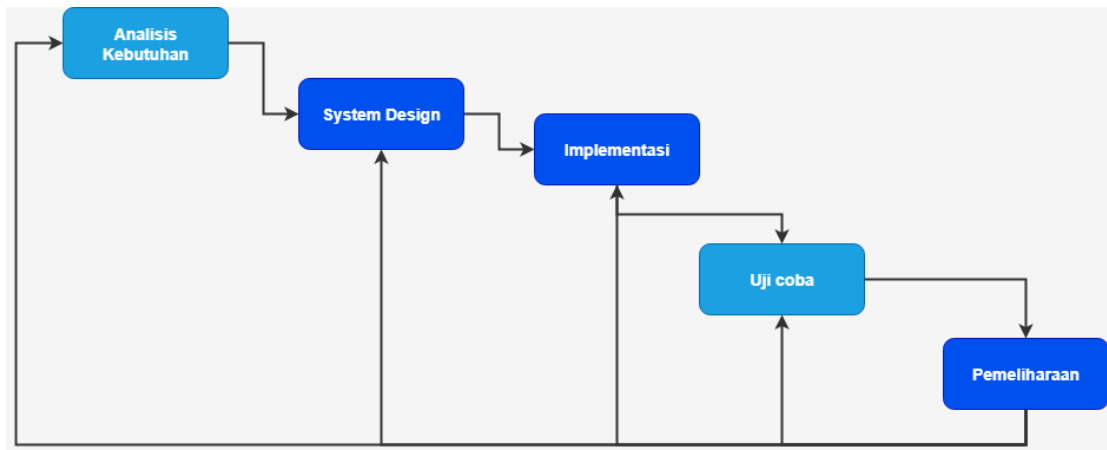
2.2.8 Akuntabilitas Publik

Akuntabilitas publik merupakan kewajiban institusi pemerintah untuk mempertanggungjawabkan pengelolaan sumber daya dan kebijakan publik kepada masyarakat (Dwiyanto, 2020). Dalam era digital, teknologi informasi berperan strategis dalam meningkatkan akuntabilitas melalui transparansi data dan proses yang dapat dilacak (Kompas, 2022). Pada konteks kegiatan reses DPRD, sistem absensi berbasis *QR code* mendukung prinsip akuntabilitas publik dengan menyediakan:

1. Bukti digital terverifikasi melalui pencatatan timestamp otomatis
2. Data *real-time* yang transparan dan dapat diakses pihak berwenang
3. Mekanisme audit trail untuk pertanggungjawaban penggunaan anggaran
4. Laporan terstruktur yang meminimalisasi risiko manipulasi data.

2.3 Metode Pengembangan Produk

Pada tahapan pengembangan *sistem* ini untuk penelitian system absensi menggunakan metode *QR code*, Metode penelitian yang digunakan dalam membangun system ini adalah metode *Waterfall*. Model ini terdiri dari beberapa tahapan utama, yaitu :



Gambar 2.2 Model *Waterfall*

Langkah-langkah dalam membangun sistem ini dengan metode *Waterfall* sebagai berikut:

1. **Analisis Kebutuhan**

Pada tahap ini, penelitian dimulai dengan upaya untuk memahami dan mengidentifikasi informasi yang diperlukan terkait perangkat lunak. Proses ini dapat dilakukan melalui berbagai metode, termasuk observasi, wawancara, dan studi pustaka.

2. **Desain Sistem**

Desain Sistem merupakan tahap krusial dalam pengembangan sistem. Pada fase ini, perancangan dilakukan pada antarmuka perangkat lunak atau arsitektur sistem dengan menggunakan diagram UML seperti *Use Case Diagram*, *Activity Diagram*, dan class diagram, yang bertujuan untuk membangun sistem absensi berbasis teknologi *Flowchart QR code*.

3. **Implementasi**

Tahapan implementasi mencakup proses pemrograman yang berfokus pada pembangunan sistem absensi berdasarkan data yang telah disusun. Setelah desain selesai, langkah selanjutnya adalah menerjemahkan desain tersebut ke dalam kode (coding). Bahasa pemrograman yang digunakan meliputi PHP, CSS, HTML, dan JavaScript, dengan *MySQL* sebagai basis data.

4. **Uji Coba**

Tahap uji coba dilakukan untuk memastikan bahwa sistem yang dikembangkan telah berjalan sesuai dengan kebutuhan yang ditetapkan. Pengujian dilakukan menggunakan metode *Black-Box Testing* untuk memverifikasi fungsi sistem serta *Load Testing* menggunakan Locust untuk mengukur kinerja sistem pada kondisi tertentu. Hasil pengujian digunakan sebagai dasar evaluasi dan perbaikan sistem sebelum diimplementasikan.

5. Pemeliharaan

Tahap pemeliharaan dilakukan setelah sistem digunakan oleh pengguna untuk memperbaiki kesalahan, melakukan penyesuaian kebutuhan, dan mengembangkan fitur tambahan. Pada penelitian ini, tahap pemeliharaan tidak menjadi bagian dari ruang lingkup penelitian sehingga pengembangan sistem dibatasi hingga tahap pengujian.

BAB III ANALISIS DAN PERANCANGAN

3.1. Analisis Kebutuhan

Bagian ini menjelaskan kebutuhan sistem secara menyeluruh, dimulai dari pemahaman terhadap kondisi saat ini hingga rancangan awal sistem yang akan diusulkan.

3.1.1 Proses Bisnis Berjalan

Saat ini, proses absensi pada kegiatan reses DPRD Kota Batam masih dilakukan secara manual. Setiap warga atau undangan yang hadir diminta mengisi daftar hadir yang telah disediakan oleh panitia. Setelah kegiatan selesai, data kehadiran tersebut dikumpulkan dan direkap oleh petugas administrasi untuk keperluan dokumentasi dan pelaporan kegiatan.

Alur proses yang berjalan saat ini dapat dijelaskan sebagai berikut:

- 1. Pembuatan Undangan**

Bagian administrasi menyiapkan dan mendistribusikan undangan kepada warga atau konstituen yang akan mengikuti kegiatan reses.

- 2. Kedatangan Peserta**

Peserta yang hadir datang ke lokasi kegiatan dan menuju meja registrasi yang telah disediakan oleh panitia.

- 3. Pencatatan Kehadiran**

Peserta mengisi data kehadiran pada daftar hadir yang tersedia sesuai dengan informasi yang diminta oleh panitia.

- 4. Pemeriksaan Data Kehadiran**

Panitia memeriksa kelengkapan data yang telah diisi oleh peserta pada daftar hadir.

- 5. Rekapitulasi Data**

Setelah kegiatan selesai, data kehadiran dikumpulkan dan direkap oleh petugas administrasi untuk penyusunan laporan kegiatan. Permasalahan atau Kendala yang Muncul

Permasalahan atau Kendala yang Muncul

Berdasarkan hasil wawancara dengan Bapak Jendral selaku staf Divisi Fasilitasi Penganggaran dan Pengawasan, Urusan Kerjasama dan Aspirasi, Sekretariat DPRD Kota Batam, proses pencatatan kehadiran yang masih dilakukan secara manual memerlukan pengelolaan data secara manual sehingga dinilai kurang praktis ketika jumlah peserta kegiatan cukup banyak. Selain itu, proses rekapitulasi data kehadiran harus dilakukan kembali setelah kegiatan selesai sehingga memerlukan waktu tambahan dalam penyusunan laporan.

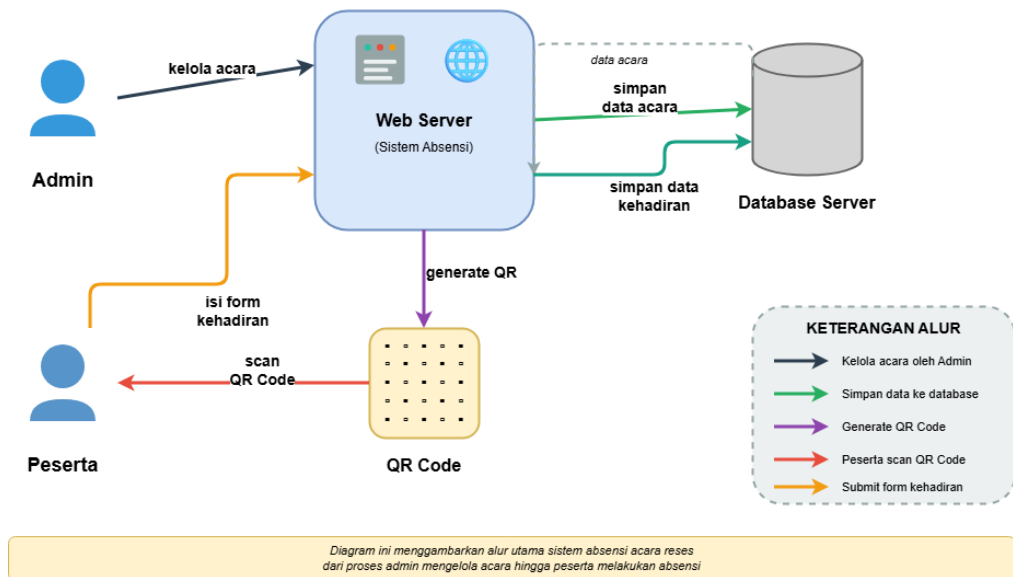
Narasumber juga menyampaikan bahwa penggunaan teknologi seperti *barcode* atau *QR code* pernah menjadi usulan untuk membantu proses pencatatan kehadiran peserta secara lebih terstruktur. Oleh karena itu, diperlukan suatu sistem absensi berbasis *QR code* yang dapat mendukung pencatatan kehadiran, pengelolaan data peserta, serta penyusunan laporan kegiatan reses DPRD Kota Batam secara lebih terstruktur dan terdokumentasi dengan baik.

3.1.2 Gambaran Umum Sistem

Bagian ini memberikan penjelasan umum mengenai sistem absensi yang telah dikembangkan sebagai solusi untuk proses pencatatan kehadiran pada kegiatan reses Anggota DPRD Kota Batam, yang sebelumnya dilakukan secara manual. Sistem ini berbentuk *Website Absensi Berbasis QR code* yang terdiri dari beberapa komponen utama, yaitu Admin (DPRD), Peserta (Masyarakat), *Web Server*, dan *Database Server*.

Tugas Admin meliputi pembuatan dan pengelolaan data acara melalui *Web Server*, setelah itu sistem akan menyimpan data tersebut ke dalam *Database* dan menghasilkan *QR code* unik untuk setiap acara. *QR code* tersebut akan dipajang di lokasi acara dan dipindai oleh Peserta menggunakan *smartphone* untuk melakukan absensi. Data kehadiran Peserta akan dikirim ke *Web Server* dan secara otomatis disimpan dalam *Database Server* secara *real-time*.

ALUR SISTEM ABSENSI ACARA RESES



Gambar 3.1 Gambaran Umum Sistem

3.2 Perancangan

Bagian ini memuat rancangan sistem yang akan dikembangkan. Perancangan dilakukan berdasarkan analisis kebutuhan yang telah dijelaskan pada bagian sebelumnya.

3.2.1 Kebutuhan *Fungsional*

Tabel 3.1 Kebutuhan *Fungsional*

ID	Kebutuhan <i>Fungsional</i>
FR-01	Sistem harus menyediakan halaman <i>Login</i> untuk Admin.
FR-02	Admin dapat mengelola data acara reses (tambah, lihat, ubah, hapus) dengan validasi <i>Field</i> wajib serta penolakan penghapusan acara yang sudah memiliki data kehadiran.
FR-03	Sistem harus menghasilkan <i>QR code</i> unik untuk setiap acara yang dibuat. Admin dapat melakukan <i>refresh QR code</i> (generate hash baru) dan mengunduh atau menyalin URL absensi.

FR-04	Admin dapat membuka atau menutup akses absensi suatu acara secara <i>real-time</i> , yang menentukan apakah peserta dapat mengisi form kehadiran.
FR-05	Peserta dapat memindai <i>QR code</i> ; sistem memvalidasi hash, status aktif acara, dan tanggal, lalu menampilkan form absensi jika valid atau halaman expired jika tidak valid.
FR-06	Peserta dapat mengisi dan mengirim form absensi (Nama, Alamat, No. Telepon, konfirmasi). Sistem menormalisasi input, memvalidasi format, dan mengecek duplikasi nomor telepon dalam acara yang sama.
FR-07	Sistem secara otomatis mencatat waktu kehadiran, alamat IP, dan user agent setiap peserta saat form absensi berhasil disubmit.
FR-08	Admin dapat melihat rekap kehadiran semua acara dengan statistik (total, hari ini, peserta unik) serta filter berdasarkan acara, tanggal, dan kata kunci (50 data per halaman).
FR-09	Admin dapat mengunduh laporan kehadiran dalam format PDF atau CSV berdasarkan filter aktif, dengan penamaan file timestamp; jika tidak ada data, sistem menampilkan peringatan.
FR-10	Admin dapat menghapus data peserta. Sistem menolak penghapusan jika peserta sudah memiliki riwayat kehadiran (total kehadiran ≥ 1) dan menampilkan pesan penolakan.
FR-11	Sistem harus mencatat seluruh aktivitas penting (<i>Login, Logout, create, edit, delete, arsip, restore, dll.</i>) ke dalam activity log. Admin dapat melihat, memfilter, dan membersihkan log lama.
FR-12	Admin dapat mengarsipkan acara yang sudah memiliki data kehadiran. Sistem mengubah status menjadi archived dan <i>is_active</i> false, serta mencatat log. Jika acara belum memiliki data, sistem menampilkan peringatan tetapi tetap mengizinkan arsip.
FR-13	Admin dapat melihat daftar acara yang diarsipkan dalam mode read-only (20 data per halaman, dengan filter pencarian dan rentang tanggal), serta memulihkan (restore) acara ke status aktif.

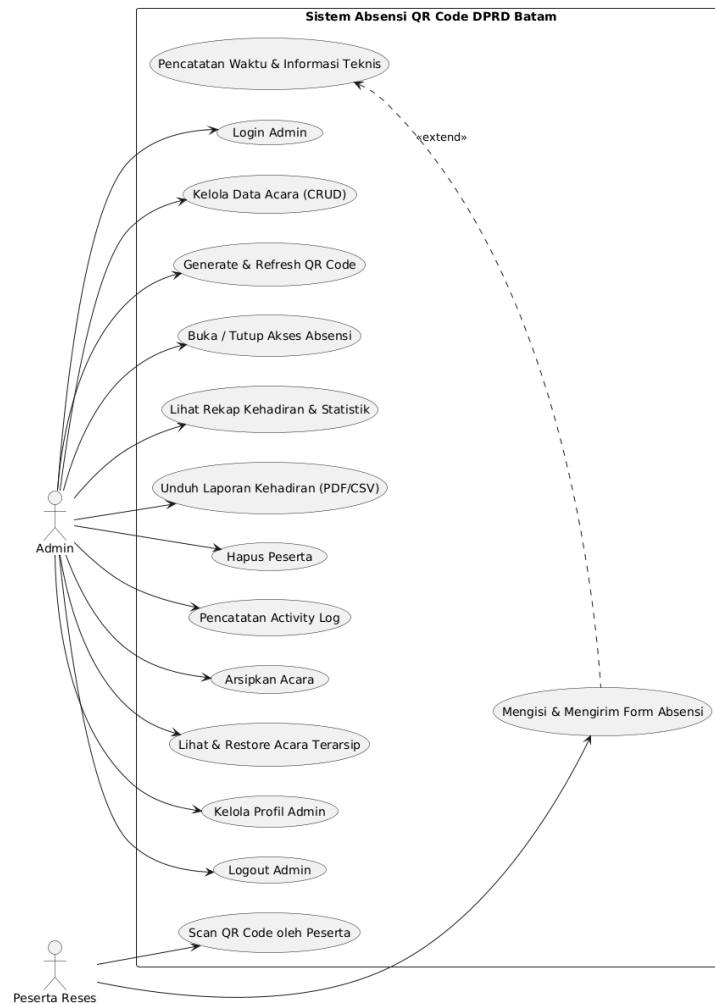
FR-14	Admin dapat mengelola profil pribadi: mengubah data profil (nama, email), mengganti <i>password</i> dengan verifikasi <i>password</i> lama, mengupload foto profil (validasi ukuran $\leq 2\text{MB}$ dan format JPG/PNG), serta menghapus foto profil.
FR-15	Sistem harus menyediakan fitur <i>Logout</i> untuk Admin yang mengakhiri sesi, menghapus data sesi, meregenerasi CSRF token, dan mengarahkan ke halaman <i>Login</i> .

3.2.2 Kebutuhan Non-Fungsional

Tabel 3.2 Kebutuhan *Non-Fungsional*

ID	Kebutuhan non-Fungsional
NFR-01	Antarmuka <i>website</i> harus <i>responsif</i> dan mudah digunakan (<i>user-friendly</i>) baik di <i>desktop</i> maupun perangkat <i>mobile</i> .
NFR-02	Sistem mampu menampilkan halaman dan memproses permintaan pengguna sesuai dengan kondisi pengujian yang dilakukan.
NFR-03	Data kehadiran harus disimpan dengan integritas yang tinggi dan dapat diambil kapan pun oleh admin.
NFR-04	<i>Kode QR</i> yang di-generate harus unik dan memiliki masa berlaku hingga acara <i>reses</i> selesai.
NFR-05	Keamanan (<i>HTTPS</i> , enkripsi <i>password</i> , proteksi <i>SQL injection/XSS</i>).
NFR-06	Backup dan <i>recovery</i> data.
NFR-07	Kompatibilitas <i>Browser</i> .
NFR-08	Skalabilitas (berapa banyak peserta yang bisa ditangani simultan?).

3.2.3 Diagram *Use Case*



Gambar 3.2 Diagram Use Case

Use Case menggambarkan fungsionalitas sistem absensi digital berbasis *QR code* untuk kegiatan reses DPRD Kota Batam yang melibatkan dua aktor utama: Admin dan Peserta Reses. Admin memiliki kewenangan penuh mengelola acara, *QR code*, akses absensi, rekap kehadiran, laporan, arsip, serta log aktivitas, sementara Peserta Reses hanya dapat memindai *QR code* dan mengirim form absensi. Sistem menerapkan validasi *real-time* berbasis nomor telepon sebagai identitas unik untuk mencegah duplikasi kehadiran, serta mencatat secara otomatis waktu, IP, dan user agent setiap transaksi. Desain ini memastikan akurasi data, transparansi, dan akuntabilitas pelaksanaan reses DPRD Kota Batam.

3.2.4 Scenario Use Case

1. Scenario Use Case Login Admin

Tabel 3.3 Scenario Use Case Login Admin

Kode Use Case	UC-01
Nama Use Case	<i>Login</i> Admin
Aktor	Admin
Deskripsi	Admin masuk ke sistem menggunakan <i>username</i> dan <i>password</i> melalui halaman <i>/Login</i> . Setiap keberhasilan maupun kegagalan <i>Login</i> dicatat ke activity log secara otomatis.
Kondisi Awal	Admin belum <i>Login</i> . Halaman GET <i>/Login</i> dapat diakses.
Kondisi Akhir	Admin berhasil masuk dan diarahkan ke <i>/Dashboard</i> , atau menerima pesan error.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin membuka halaman <i>/Login</i> .	2. Sistem menampilkan form <i>Login</i> (<i>username</i> , <i>password</i>).
3. Admin mengisi <i>username</i> dan <i>password</i> , lalu menekan tombol Masuk.	4. Sistem memvalidasi kredensial, membuat sesi baru, meregenerasi session ID, dan memperbarui <i>last_Login_at</i> serta <i>last_Login_ip</i> .
	5. Sistem mencatat activity log: " <i>Login</i> berhasil".
	6. Sistem mengarahkan Admin ke <i>/Dashboard</i> .
Alur Kejadian Alternatif	
Kondisi	Aksi Sistem
<i>Username</i> tidak ditemukan	Sistem menampilkan: " <i>Username</i> tidak ditemukan."
<i>Password</i> salah	Sistem menampilkan: " <i>Password</i> salah."

<i>Field</i> kosong	Sistem menampilkan pesan wajib diisi pada <i>Field</i> yang kosong.
---------------------	---

2. Scenario *Use Case* Kelola Data Acara (CRUD)

Tabel 3.4 Scenario *Use Case* Kelola Data Acara

Kode <i>Use Case</i>	UC-02
Nama <i>Use Case</i>	Kelola Data Acara (CRUD)
Aktor	Admin
Deskripsi	Admin dapat membuat, melihat, mengedit, dan menghapus data acara reses melalui menu <i>/Events</i> . Penghapusan acara yang sudah memiliki data kehadiran ditolak; sistem menyarankan fitur Arsipkan.
Kondisi Awal	Admin sudah <i>Login</i> dan berada di halaman GET <i>/Events</i> .
Kondisi Akhir	Data acara berhasil dibuat/diperbarui/dihapus, atau Admin menerima pesan error.
Alur Kejadian Normal — Membuat Acara Baru	
Aktor	Sistem
1. Admin mengakses GET <i>/Events/create</i> dan mengisi form.	2. Sistem memvalidasi seluruh <i>Field</i> wajib (nama min 3 karakter, lokasi, tanggal $\geq$ hari ini).
	3. Sistem menyimpan acara via POST <i>/Events</i> , membuat qr_code_hash (<i>UUID</i>), dan membuat qr_code_url via QuickChart.io.
	4. Sistem mencatat activity log dan mengarahkan ke GET <i>/Events/{id}/detail</i> dengan pesan sukses.
Alur Kejadian Alternatif	
Kondisi	Aksi Sistem

<i>Field</i> wajib kosong atau tidak valid	Sistem menampilkan pesan validasi pada <i>Field</i> yang bermasalah dan mengembalikan form.
Hapus acara (DELETE <i>/Events/{id}</i>) yang sudah memiliki peserta	Sistem menolak: "Tidak dapat menghapus acara yang sudah memiliki data absensi. Gunakan fitur Arsipkan."
Edit (GET <i>/Events/{id}/edit</i>) acara yang sudah diarsipkan	Sistem menolak perubahan dan mengarahkan ke halaman arsip.

3. Scenario *Use Case* Generate & Refresh QR code

Tabel 3.5 Scenario *Use Case* Generate & Refresh QR code

Kode Use Case	UC-03
Nama Use Case	Generate & Refresh QR code
Aktor	Admin
Deskripsi	QR code unik dibuat otomatis saat acara dibuat. Admin dapat merefresh QR code via AJAX POST <i>/Events/{id}/qr/refresh</i> untuk menghasilkan hash baru, mengunduh PNG, atau menyalin URL absensi dari halaman GET <i>/Events/{id}/qr</i> .
Kondisi Awal	Admin sudah Login dan membuka halaman GET <i>/Events/{id}/qr</i> .
Kondisi Akhir	QR code baru berhasil dibuat dan ditampilkan di halaman tanpa reload.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin menekan tombol Refresh QR.	2. Sistem mengirim AJAX POST ke <i>/Events/{id}/qr/refresh</i> .
	3. Sistem membuat qr_code_hash baru (UUID), memperbarui qr_code_hash, qr_code_generated_at, dan qr_code_url di Database.

	4. Sistem mencatat activity log: " <i>Refresh QR code</i> untuk acara: [nama acara]".
	5. Sistem memperbarui gambar <i>QR code</i> di halaman tanpa reload dan menampilkan notifikasi sukses.
Alur Kejadian Alternatif	
Kondisi	Aksi Sistem
<i>Browser</i> tidak mendukung <i>Clipboard API</i> (saat salin URL)	Sistem menggunakan fallback <code>execCommand('copy')</code> . Jika gagal, menampilkan: "Gagal menyalin URL".

4. Scenario *Use Case* Buka / Tutup Akses Absensi

Tabel 3.6 Scenario *Use Case* Buka / Tutup Akses Absensi

Kode <i>Use Case</i>	UC-04
Nama <i>Use Case</i>	Buka / Tutup Akses Absensi
Aktor	Admin
Deskripsi	Admin mengubah status <code>is_active</code> acara secara <i>real-time</i> melalui <i>AJAX POST /Events/{id}/toggle-status</i> dari halaman <i>QR code</i> . Status menentukan apakah peserta dapat mengisi form absensi.
Kondisi Awal	Admin sudah <i>Login</i> dan berada di halaman <code>GET /Events/{id}/qr</code> .
Kondisi Akhir	Status <code>is_active</code> acara berhasil diubah dan tampilan badge diperbarui.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin menekan tombol Buka/Tutup Akses Absensi dan mengonfirmasi dialog.	2. Sistem mengirim <i>AJAX POST</i> ke <code>/Events/{id}/toggle-status</code> .

	3. Sistem mengubah <code>is_active</code> (true/false) dan mencatat activity log.
	4. Sistem memperbarui badge ("Bisa Absen" / "Tidak Bisa Absen") dan teks tombol tanpa reload.
Alur Kejadian Alternatif	
Kondisi	Aksi Sistem
Admin membatalkan dialog konfirmasi	Sistem membatalkan proses; tidak ada perubahan status.

5. Scenario Use Case Scan QR code oleh Peserta

Tabel 3.7 Scenario Use Case Scan QR code oleh Peserta

Kode Use Case	UC-05
Nama Use Case	Scan QR code oleh Peserta
Aktor	Peserta Reses
Deskripsi	Peserta memindai <i>QR code</i> yang mengarah ke <code>GET /absensi/{qr_hash}</code> . Sistem memvalidasi hash, status aktif, dan tanggal acara sebelum menampilkan form absensi.
Kondisi Awal	Peserta memiliki <i>QR code</i> acara yang valid dan <i>Event</i> masih aktif.
Kondisi Akhir	Peserta melihat halaman attendance-form atau halaman expired sesuai hasil validasi.
Alur Kejadian Normal	
Aktor	Sistem
1. Peserta memindai <i>QR code</i> menggunakan kamera HP.	3. Sistem mencari <i>Event</i> berdasarkan <code>qr_code_hash</code> . Jika tidak ditemukan, tampilkan <code>view attendance.expired</code> .
2. <i>Browser</i> membuka URL <code>GET /absensi/{qr_hash}</code> secara otomatis.	4. Sistem memanggil <code>validateEventAccess()</code> : validasi <code>is_active</code> dan tanggal acara.

	5. Sistem mencatat activity log akses form dan menampilkan view attendance.attendance-form.
Alur Kejadian Alternatif	
Kondisi	Aksi Sistem
qr_code_hash tidak ditemukan di <i>Database</i>	Sistem menampilkan view expired: " <i>QR code</i> tidak valid atau sudah kadaluarsa."
Acara tidak aktif (is_active = false)	Sistem menampilkan view expired: " <i>Event</i> sudah tidak aktif. Tidak dapat melakukan absensi."
Tanggal acara sudah lewat	Sistem menampilkan view expired: "Acara Sudah Berakhir [tanggal]."

6. Scenario *Use Case* Mengisi & Mengirim Form Absensi

Tabel 3.8 Scenario *Use Case* Mengisi & Mengirim Form Absensi

Kode <i>Use Case</i>	UC-06
Nama <i>Use Case</i>	Mengisi & Mengirim Form Absensi
Aktor	Peserta Reses
Deskripsi	Peserta mengisi form absensi (Nama Lengkap, Alamat, Nomor Telepon, checkbox konfirmasi) lalu mengirimkan via POST /absensi/{qr_hash}. Sistem menormalisasi input, memvalidasi, mengecek duplikasi, dan menyimpan data kehadiran.
Kondisi Awal	Halaman attendance-form berhasil ditampilkan (UC-05 selesai).
Kondisi Akhir	Peserta diarahkan ke halaman /attendance/success/{Event}/{attendance} atau menerima pesan error.
Alur Kejadian Normal	

Aktor	Sistem
1. Peserta mengisi Nama Lengkap, Alamat Lengkap, dan Nomor Telepon.	3. Sistem menormalisasi <i>input</i> (kapitalisasi nama, hapus karakter non-digit dari telepon).
2. Peserta mencentang checkbox konfirmasi dan menekan Simpan Kehadiran.	4. Sistem memvalidasi seluruh <i>Field</i> melalui <code>validateAttendanceInput()</code> .
	5. Sistem memeriksa duplikasi via <code>checkDuplicateAttendance(Event_id, phone)</code> .
	6. Sistem menyimpan data peserta dan kehadiran ke <i>Database</i> , mencatat waktu, IP, dan user agent.
	7. Sistem commit transaksi DB dan redirect ke <code>/attendance/success/{Event}/{attendance}</code> .
Alur Kejadian Alternatif	
Kondisi	Aksi Sistem
<i>Field</i> wajib belum terisi atau format tidak valid	Sistem mengembalikan <code>redirect ()->back ()</code> dengan <code>withErrors()</code> dan <code>withInput()</code> .
Nomor telepon tidak 10–15 digit	Sistem menampilkan: "Nomor telepon harus 10–15-digit angka".
Peserta sudah absen di acara yang sama (duplikasi)	Sistem menampilkan: "ANDA SUDAH ABSEN SEBELUMNYA!" beserta waktu absen sebelumnya.
Acara tidak lagi aktif saat submit	Sistem menampilkan pesan <code>error_message</code> : acara tidak aktif dan mengembalikan form.

7. Scenario *Use Case* Pencatatan Waktu & Informasi Teknis Kehadiran

Tabel 3.9 Scenario *Use Case* Pencatatan Waktu & Informasi Teknis Kehadiran

Kode Use Case	UC-07
Nama Use Case	Pencatatan Waktu & Informasi Teknis
Aktor	Sistem (otomatis saat form disubmit)
Deskripsi	Saat form absensi berhasil disubmit dan lolos validasi, sistem secara otomatis mencatat waktu kehadiran, alamat IP, dan user agent perangkat peserta ke tabel attendances.
Kondisi Awal	Data form tervalidasi (UC-06) dan tidak ada duplikasi.
Kondisi Akhir	Data teknis tersimpan bersama data kehadiran di tabel attendances.
Alur Kejadian Normal	
Aktor	Sistem
	1. Sistem mengambil waktu saat ini (attendance_time = now()).
	2. Sistem mengambil IP pengirim (\$request->ip()) dan menyimpan ke kolom ip_address.
	3. Sistem mengambil user agent (\$request->userAgent()) dan menyimpan ke kolom user_agent.
	4. Sistem menyimpan ketiganya bersama data peserta dalam satu transaksi DB ke tabel attendances.

8. Lihat Daftar Peserta & Statistik

Tabel 3.10 Scenario *Use Case* Lihat Daftar Peserta & Statistik

Kode Use Case	UC-08
Nama Use Case	Lihat Rekap Kehadiran & Statistik
Aktor	Admin

Deskripsi	Admin melihat seluruh data kehadiran dari semua acara di halaman GET /attendances dengan statistik (total, hari ini, peserta unik) dan filter berdasarkan acara, rentang tanggal, dan kata kunci (50 data per halaman).
Kondisi Awal	Admin sudah <i>Login</i> .
Kondisi Akhir	Admin melihat daftar kehadiran dan statistik sesuai filter yang diterapkan.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin membuka halaman GET /attendances.	2. Sistem mengambil data kehadiran (50/halaman) dan menghitung statistik via <code>getStatistics()</code> .
	3. Sistem menampilkan kartu statistik dan tabel data kehadiran di <code>view attendances.index</code> .
Alur Kejadian Alternatif	
Kondisi	Aksi Sistem
Admin menerapkan filter (acara/kata kunci/tanggal)	Sistem memperbarui statistik dan tabel sesuai parameter filter yang dikirim.
Tidak ada data kehadiran	Sistem menampilkan: "Belum ada data kehadiran".

9. Scenario *Use Case* Unduh Laporan Kehadiran (PDF & CSV)

Tabel 3.11 Scenario *Use Case* Unduh Laporan Kehadiran (PDF & CSV)

Kode <i>Use Case</i>	UC-09
Nama <i>Use Case</i>	Unduh Laporan Kehadiran (PDF & CSV)
Aktor	Admin
Deskripsi	Admin mengekspor data kehadiran dalam format PDF (GET /attendances/export/PDF via

	DomPDF) atau CSV (GET /attendances/export/CSV). File diberi nama dengan timestamp dan dapat difilter sesuai parameter aktif.
Kondisi Awal	Admin sudah <i>Login</i> dan berada di halaman GET /attendances.
Kondisi Akhir	File PDF atau CSV laporan kehadiran berhasil diunduh oleh <i>Browser</i> .
Alur Kejadian Normal	
Aktor	Sistem
1. Admin menekan tombol Export PDF atau Export CSV.	2. Sistem mengambil data kehadiran sesuai filter aktif.
	3. Sistem membuat file (DomPDF untuk PDF / <i>StreamedResponse</i> untuk CSV) dengan nama: rekap-kehadiran-[acara]-[timestamp].PDF/.CSV.
	4. Sistem mencatat activity log dan mengirim file ke <i>Browser</i> untuk diunduh.
Alur Kejadian Alternatif	
Kondisi	Aksi Sistem
Tidak ada data yang sesuai filter	Sistem menampilkan pesan peringatan tanpa menghasilkan file.

10. Scenario *Use Case* Edit & Hapus Data Kehadiran

Tabel 3.12 Scenario *Use Case* Edit & Hapus Data Kehadiran

Kode <i>Use Case</i>	UC-10
Nama <i>Use Case</i>	Edit & Hapus Data Kehadiran
Aktor	Admin
Deskripsi	Admin dapat mengedit data kehadiran via GET /attendances/{id}/edit & PUT /attendances/{id}, menghapus satu data via DELETE

	/attendances/{id}, atau menghapus massal via POST /attendances/bulk-delete.
Kondisi Awal	Admin sudah <i>Login</i> dan berada di halaman GET /attendances.
Kondisi Akhir	Data kehadiran berhasil diperbarui atau dihapus; Admin diarahkan kembali ke /attendances.
Alur Kejadian Normal — Edit Data Kehadiran	
Aktor	Sistem
1. Admin menekan tombol Hapus pada baris peserta yang dipilih.	2. Sistem menampilkan dialog konfirmasi: "Apakah Anda yakin ingin menghapus data peserta ini?"
3. Admin menekan tombol Konfirmasi.	4. Sistem memeriksa riwayat kehadiran peserta.
	5. Sistem mengeksekusi DELETE /attendances/{id}, menghapus data peserta, dan mencatat activity log: "Menghapus peserta: [nama] - ID: [id]".
	6. Sistem menampilkan pesan sukses: "Data peserta berhasil dihapus."
Alur Kejadian Alternatif	
Kondisi	Aksi Sistem
Peserta sudah memiliki riwayat kehadiran (total kehadiran ≥ 1)	Sistem menolak penghapusan dan menampilkan pesan: "Peserta tidak dapat dihapus karena sudah memiliki riwayat kehadiran."
Admin membatalkan dialog konfirmasi	Sistem menutup dialog dan tidak melakukan perubahan apapun.

11. Scenario *Use Case* Pencatatan Activity Log

Tabel 3.13 S Scenario *Use Case* Pencatatan Activity Log

Kode <i>Use Case</i>	UC-11
-----------------------------	-------

Nama Use Case	Pencatatan Activity Log
Aktor	Admin
Deskripsi	Admin dapat melihat seluruh riwayat aktivitas sistem di halaman GET /activity-logs yang dicatat otomatis, melakukan filter berdasarkan kata kunci, dan membersihkan log lama via POST /activity-logs/clear-old.
Kondisi Awal	Admin sudah <i>Login</i> .
Kondisi Akhir	Admin melihat daftar activity log, atau log lama berhasil dibersihkan.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin membuka halaman GET /activity-logs.	2. Sistem mengambil seluruh data log dari <i>Database</i> (waktu, pengguna, aktivitas, IP address) dari terbaru.
	3. Sistem menampilkan daftar log di view activity-logs.index.
Alur Kejadian Alternatif	
Kondisi	Aksi Sistem
Admin menekan Bersihkan Log Lama (POST /activity-logs/clear-old)	Sistem menghapus log di luar batas waktu yang ditentukan dan menampilkan pesan sukses dengan jumlah log yang dihapus.

12. Scenario Use Case Arsipkan Acara

Tabel 3.14 Scenario Use Case Arsipkan Acara

Kode Use Case	UC-12
Nama Use Case	Arsipkan Acara
Aktor	Admin
Deskripsi	Admin mengarsipkan acara yang telah selesai via POST /Events/{id}/archive. Status acara

	berubah menjadi 'archived' dan is_active menjadi false; data tersimpan secara permanen dalam mode read-only.
Kondisi Awal	Admin sudah <i>Login</i> . Acara berstatus aktif dan memiliki data kehadiran.
Kondisi Akhir	Acara berstatus archived dan muncul di halaman GET <i>/Events/archived</i> .
Alur Kejadian Normal	
Aktor	Sistem
1. Admin menekan tombol Arsipkan pada halaman GET <i>/Events/{id}/detail</i> .	2. Sistem memverifikasi status acara.
	3. Sistem mengubah status menjadi 'archived' dan is_active menjadi false via POST <i>/Events/{id}/archive</i> .
	4. Sistem mencatat activity log: "Mengarsipkan acara: [nama acara]".
	5. Sistem redirect ke <i>/Events</i> dengan pesan sukses.
Alur Kejadian Alternatif	
Kondisi	Aksi Sistem
Acara belum memiliki data kehadiran peserta	Sistem menampilkan peringatan: "Acara tidak memiliki data absensi." namun tetap mengizinkan proses pengarsipan berlanjut.
Admin membatalkan sebelum konfirmasi	Sistem tidak melakukan perubahan apapun dan tetap berada di halaman detail acara.

13. Scenario Use Cse Lihat & Restore Acara Terarsip

Tabel 3.15 Scenario Use Cse Lihat & Restore Acara Terarsip

Kode Use Case	UC-13
----------------------	-------

Nama Use Case	Lihat & Restore Acara Terarsip
Aktor	Admin
Deskripsi	Admin melihat daftar acara berstatus archived di GET /Events/archived (20/halaman, mode read-only) dan dapat memulihkan acara ke status aktif via POST /Events/{id}/restore.
Kondisi Awal	Admin sudah <i>Login</i> .
Kondisi Akhir	Admin melihat daftar arsip atau acara berhasil dipulihkan ke daftar aktif.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin membuka halaman GET /Events/archived.	2. Sistem mengambil acara berstatus archived (20/halaman) dengan filter pencarian dan rentang tanggal.
	3. Sistem menampilkan daftar arsip beserta jumlah peserta dan tanggal pengarsipan (read-only).
Alur Kejadian Normal	
Aktor	Sistem
1. Admin menekan tombol Restore pada acara yang dipilih.	2. Sistem menampilkan dialog konfirmasi: "Apakah Anda yakin ingin restore acara ini? Acara akan dikembalikan ke daftar aktif dengan status nonaktif."
3. Admin menekan tombol Konfirmasi.	4. Sistem mengeksekusi POST /Events/{id}/restore, mengubah status menjadi 'active' dengan is_active tetap false.
	5. Sistem mencatat activity log: "Memulihkan acara: [nama acara]".
	6. Sistem redirect ke /Events/{id}/detail dengan pesan sukses.

Alur Kejadian Alternatif	
Kondisi	Aksi Sistem
Admin membatalkan dialog konfirmasi restore	Sistem menutup dialog dan tidak melakukan perubahan apapun.
Tidak ada acara yang diarsipkan	Sistem menampilkan pesan: "Belum ada acara yang diarsipkan."

14. Scenario *Use Case* Kelola Profil Admin

Tabel 3.16 Scenario *Use Case* Kelola Profil Admin

Kode <i>Use Case</i>	UC-14
Nama <i>Use Case</i>	Kelola Profil Admin
Aktor	Admin
Deskripsi	Admin memperbarui data profil via PUT /profile, mengganti <i>password</i> via PUT /profile/ <i>password</i> , dan mengupload/menghapus foto profil via PUT /profile/avatar atau DELETE /profile/avatar.
Kondisi Awal	Admin sudah <i>Login</i> dan membuka halaman GET /profile.
Kondisi Akhir	Profil, <i>password</i> , atau foto berhasil diperbarui, atau Admin menerima pesan error.
Alur Kejadian Normal — Perbarui Profil & <i>Password</i>	
Aktor	Sistem
1. Admin mengubah data profil lalu menekan Simpan Perubahan.	2. Sistem memvalidasi keunikan email/ <i>username</i> , menyimpan via PUT /profile, mencatat log: "Memperbarui profil".
3. Admin mengisi <i>Password</i> Saat Ini, <i>Password</i> Baru, Konfirmasi, lalu menekan Ubah <i>Password</i> .	4. Sistem memverifikasi <i>password</i> lama (Hash: check ()), meng-hash <i>password</i> baru, menyimpan via PUT /profile/ <i>password</i> , mencatat log: "Mengubah <i>password</i> ".

Alur Kejadian Alternatif	
Kondisi	Aksi Sistem
Email atau <i>username</i> sudah digunakan	Sistem menampilkan pesan validasi unique.
<i>Password</i> saat ini salah	Sistem menampilkan: " <i>Password</i> saat ini tidak sesuai."
Konfirmasi <i>password</i> tidak cocok	Sistem menampilkan pesan validasi confirmed.
File foto > 2MB atau bukan JPG/PNG	Sistem menampilkan pesan validasi format/ukuran.
Hapus foto — tidak ada foto tersimpan	Sistem menampilkan: " <i>Tidak ada foto profil untuk dihapus.</i> "

15. Scenario Use Case Logout Admin

Tabel 3.17 Scenario Use Case Logout Admin

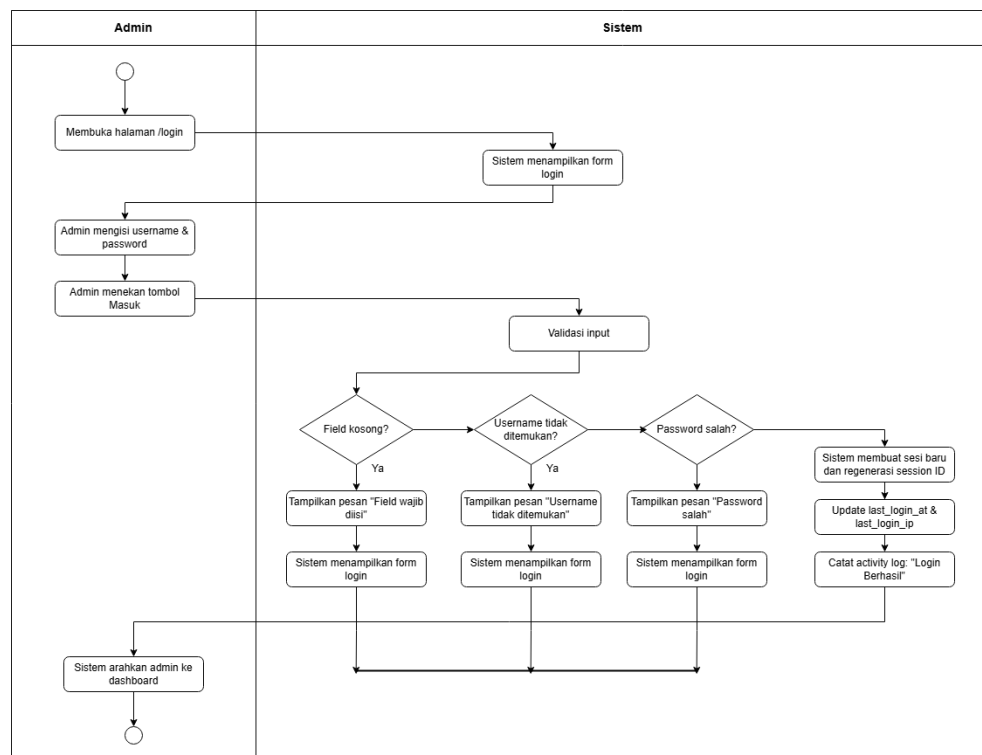
Kode Use Case	UC-15
Nama Use Case	Logout Admin
Aktor	Admin
Deskripsi	Admin keluar dari sistem melalui POST /Logout. Sistem mengakhiri sesi aktif secara aman, meregenerasi CSRF token, dan mengarahkan ke halaman Login.
Kondisi Awal	Admin sudah Login dan berada di halaman manapun dalam sistem.
Kondisi Akhir	Sesi Admin berakhir dan Admin diarahkan ke GET /Login.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin menekan tombol Logout.	2. Sistem mengirim POST /Logout dan mencatat activity log: "Logout".

	3. Sistem menghapus sesi (Auth: <i>Logout</i> (), session ()->invalidate ()) dan meregenerasi CSRF token.
	4. Sistem redirect ke <i>/Login</i> dengan pesan: "Anda telah berhasil <i>Logout</i> ."

3.2.5 Activity Diagram

3.2.5.1 Activity Diagram Login Admin

Admin mengisi *username* dan *password*, sistem memvalidasi; jika gagal (*Field* kosong, *username* tidak ditemukan, *password* salah) kembali ke form *Login*, jika berhasil buat sesi, catat log, redirect ke *Dashboard* (stop).

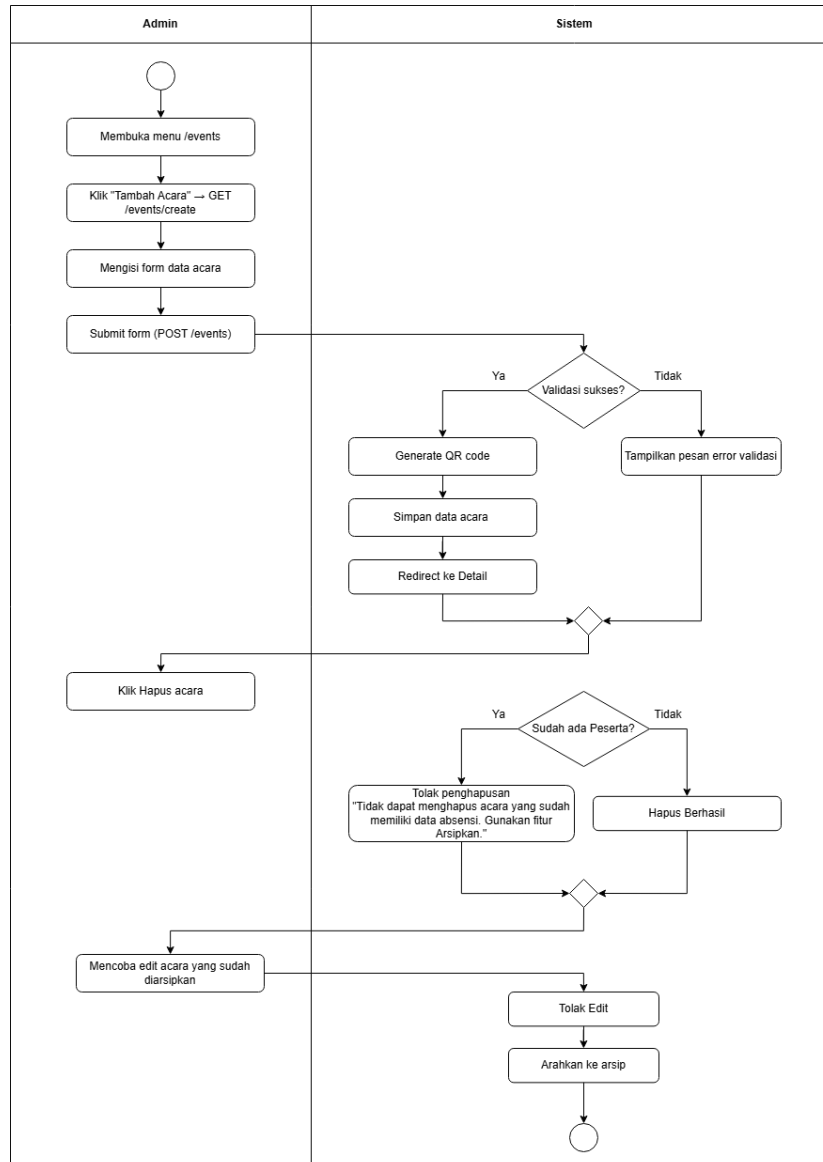


Gambar 3.3 Activity Diagram Login Admin

3.2.5.2 Activity Diagram Kelola Data Acara

Admin memilih aksi (buat/edit/hapus); saat buat: isi form, validasi *Field*, simpan acara & generate *QR*, catat log, redirect ke detail; saat edit: cek arsip, tampil form, validasi, simpan; saat hapus: cek peserta,

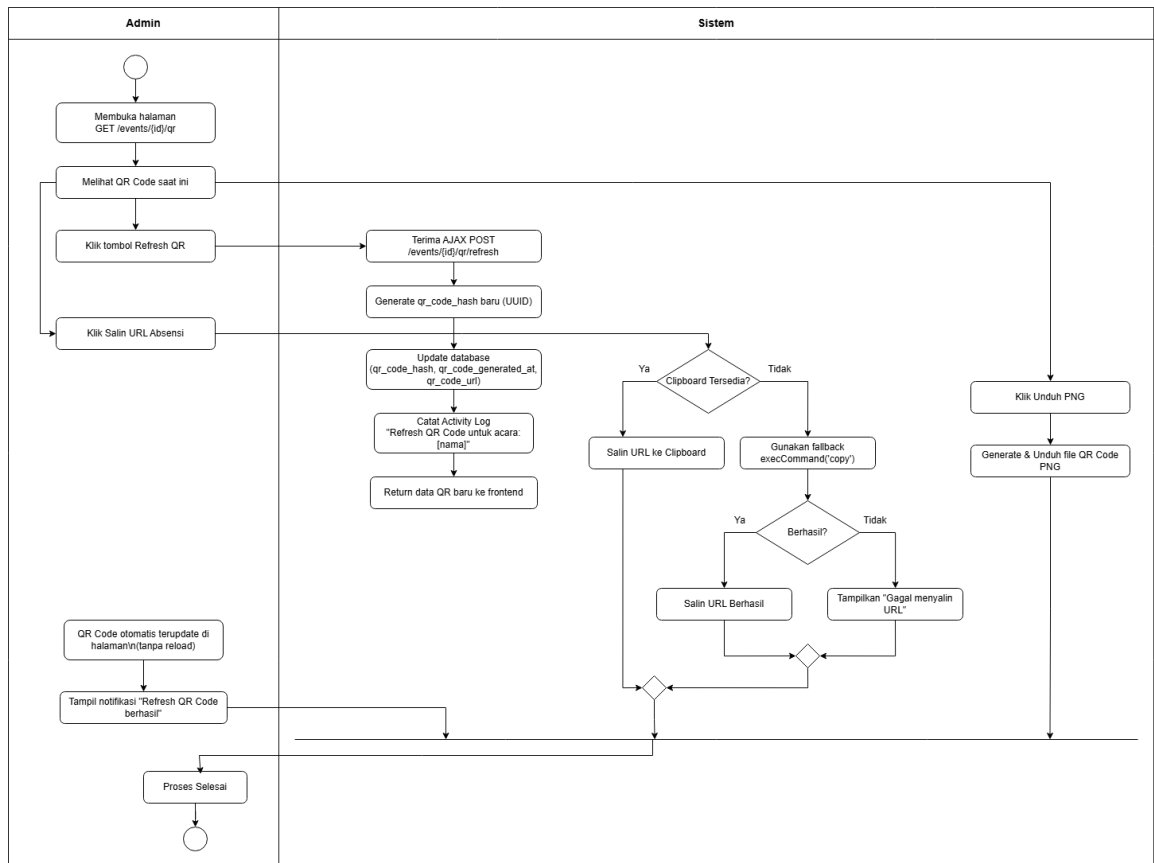
jika ada peserta tolak hapus (saran arsip), jika tidak hapus; semua sukses kembali ke daftar (stop).



Gambar 3.4 Activity Diagram Kelola Data Acara

3.2.5.3 Activity Diagram Generate & Refresh QR code

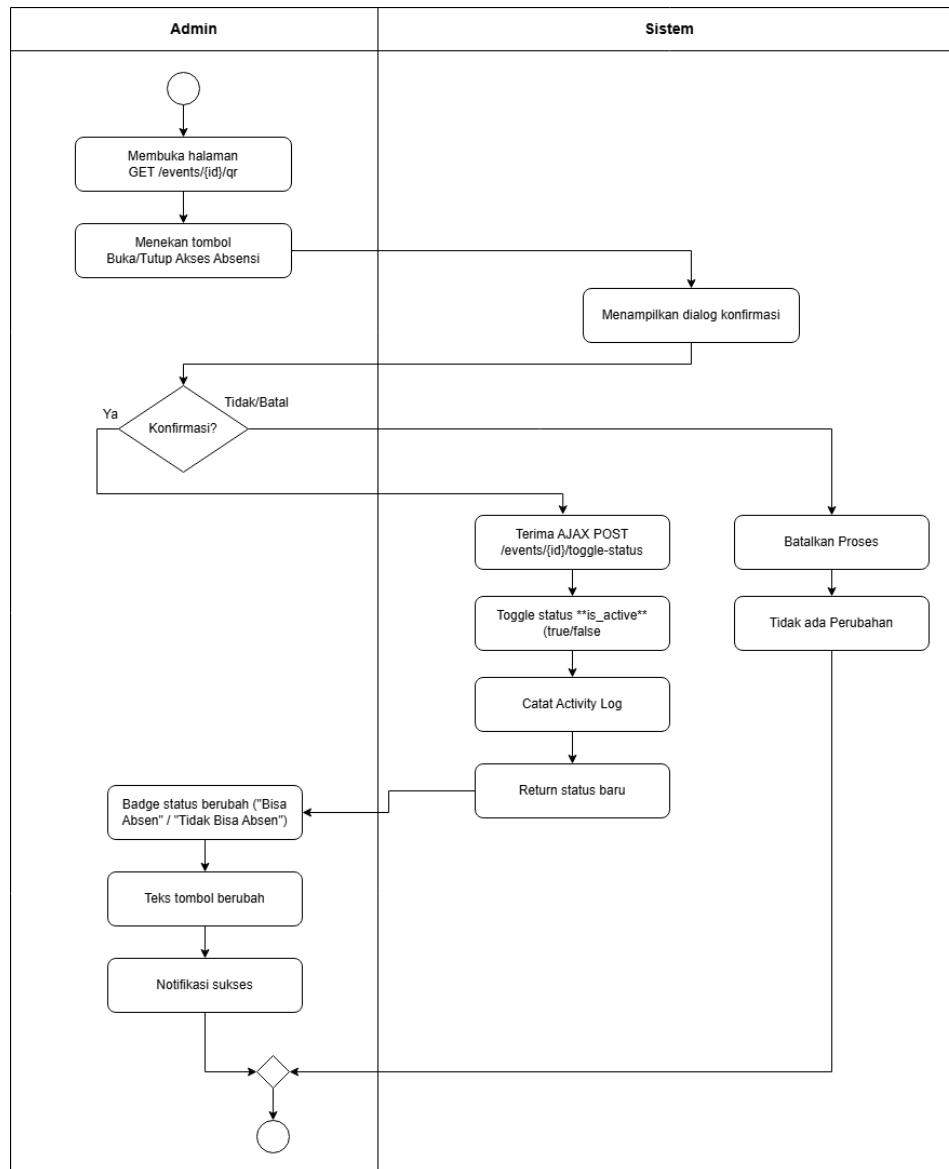
Admin di halaman detail, tekan *refresh* QR; sistem buat hash baru, update *Database*, catat log, perbarui gambar QR tanpa reload, tampilkan notifikasi.



Gambar 3.5 Activity Diagram Generate & Refresh QR code

3.2.5.4 Activity Diagram Buka / Tutup Akses Absensi

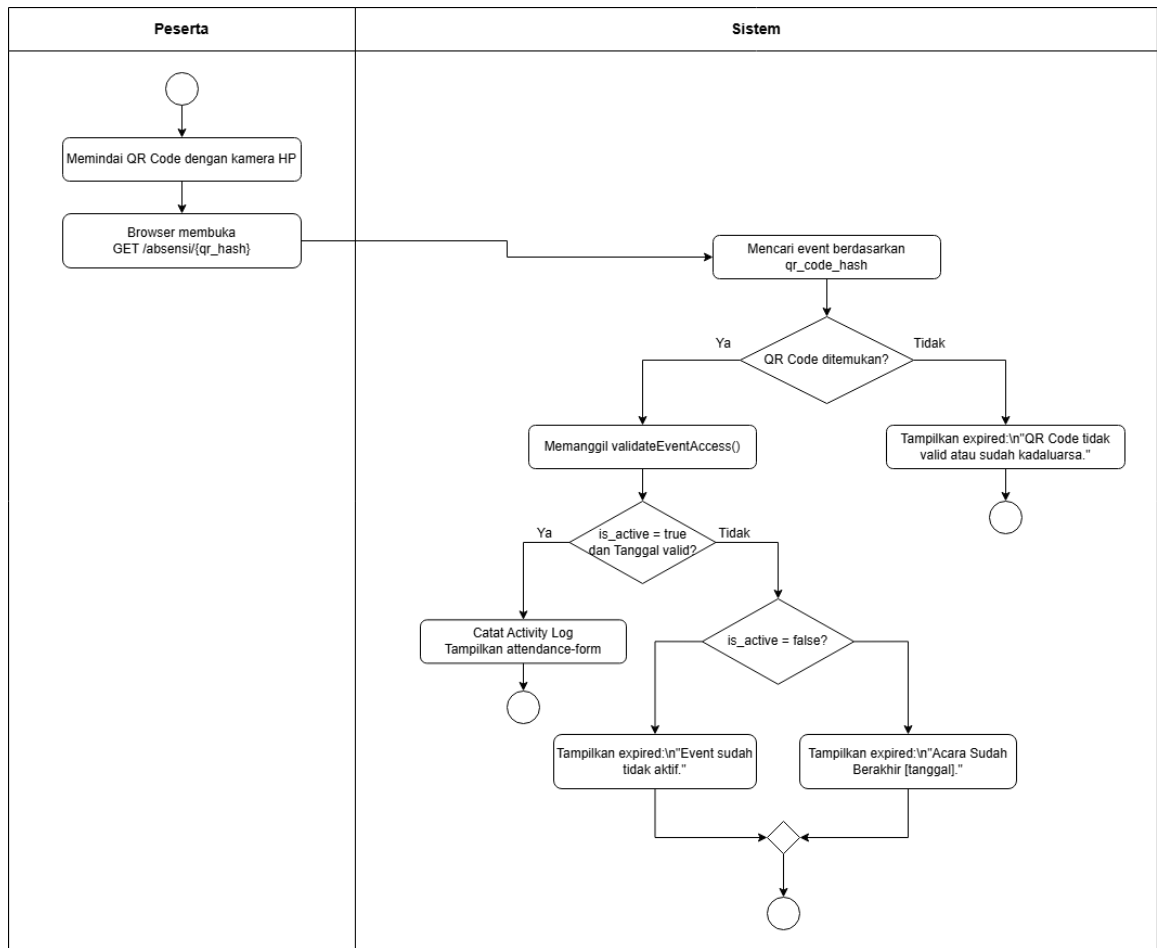
Admin tekan tombol toggle, sistem tampilkan konfirmasi; jika batal tutup dialog, jika ya ubah status is_active, catat log, perbarui badge dan teks tombol secara *real-time*.



Gambar 3.6 Activity Diagram Buka / Tutup Akses Absensi

3.2.5.5 Activity Diagram Scan QR code oleh Peserta

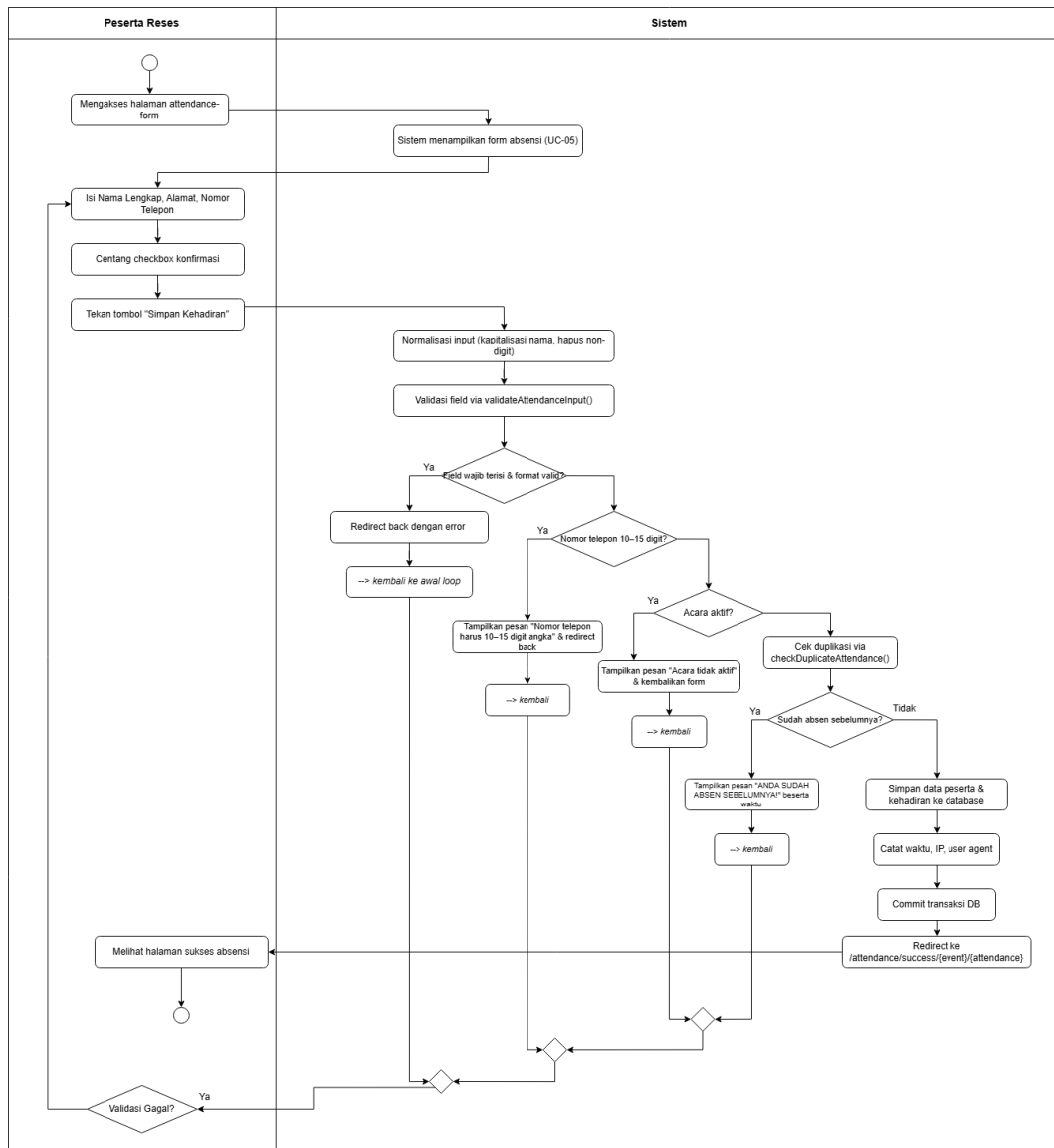
Peserta scan QR, *Browser* buka URL; sistem cari *Event* berdasarkan hash; jika hash tidak ditemukan atau *Event* tidak aktif/tanggal lewat, tampilkan halaman expired; jika valid, catat log akses dan tampilkan form absensi.



Gambar 3.7 Activity Diagram Scan QR code oleh Peserta

3.2.5.6 Activity Diagram Mengisi & Mengirim Form Absensi

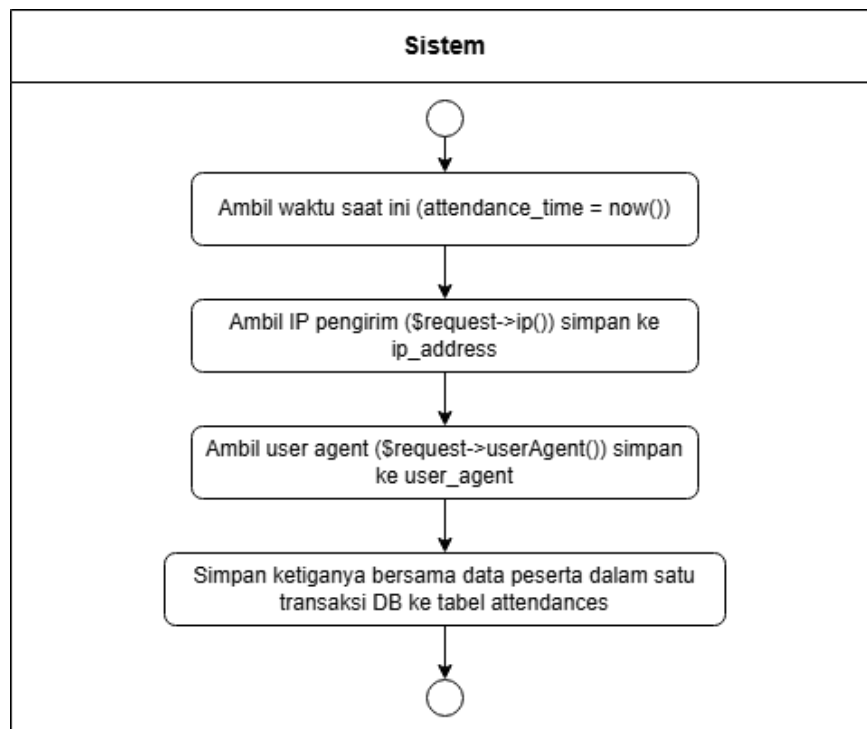
Peserta isi form, tekan simpan; sistem normalisasi input, validasi *Field*, cek nomor telepon (10-15 digit), cek status acara aktif, cek duplikasi; jika gagal (*Field* invalid, telepon salah, acara tidak aktif, duplikasi) kembali ke form; jika sukses simpan data & kehadiran, catat waktu, IP, user agent, commit DB, redirect ke halaman sukses



Gambar 3.8 Activity Diagram Mengisi & Mengirim Form Absensi

3.2.5.7 Activity Diagram Pencatatan Waktu & Informasi Teknis

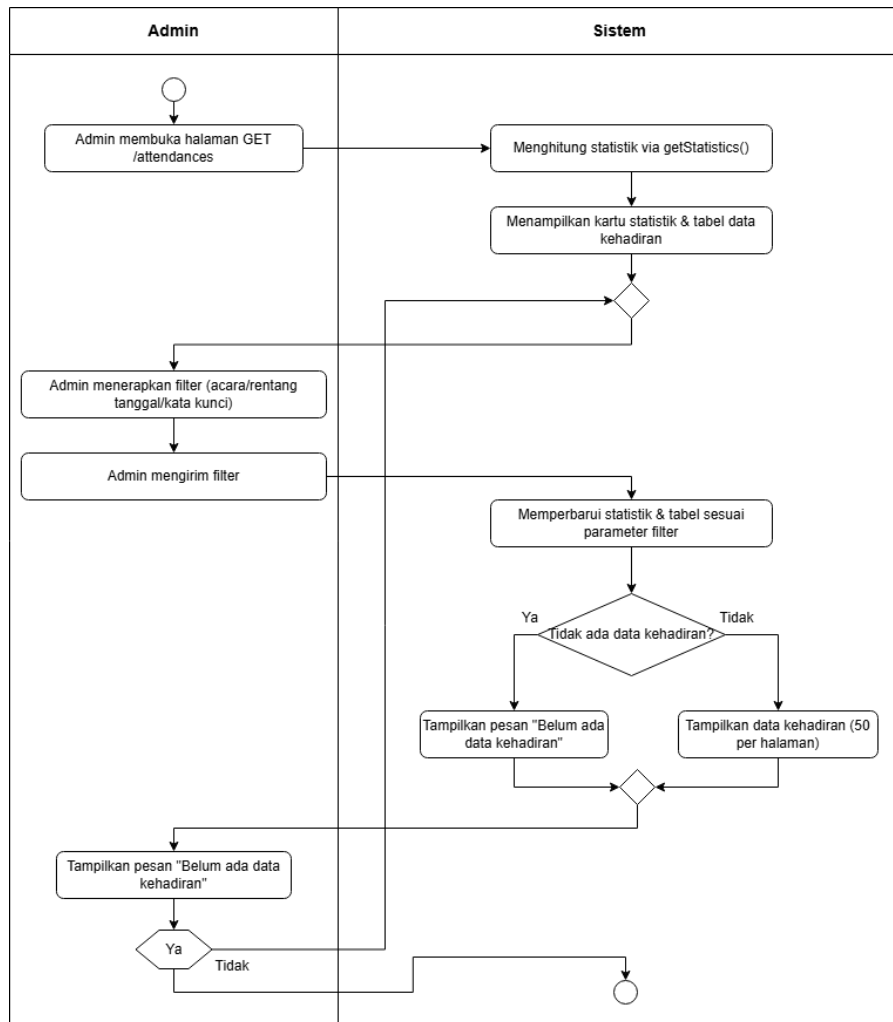
(otomatis terjadi saat UC-06 sukses) Sistem ambil waktu now (), IP (\$request->ip()), user agent, simpan ke tabel attendances bersama data peserta dalam satu transaksi



Gambar 3.9 *Activity Diagram* Pencatatan Waktu & Informasi Teknis

3.2.5.8 *Activity Diagram* Lihat Rekap Kehadiran & Statistik

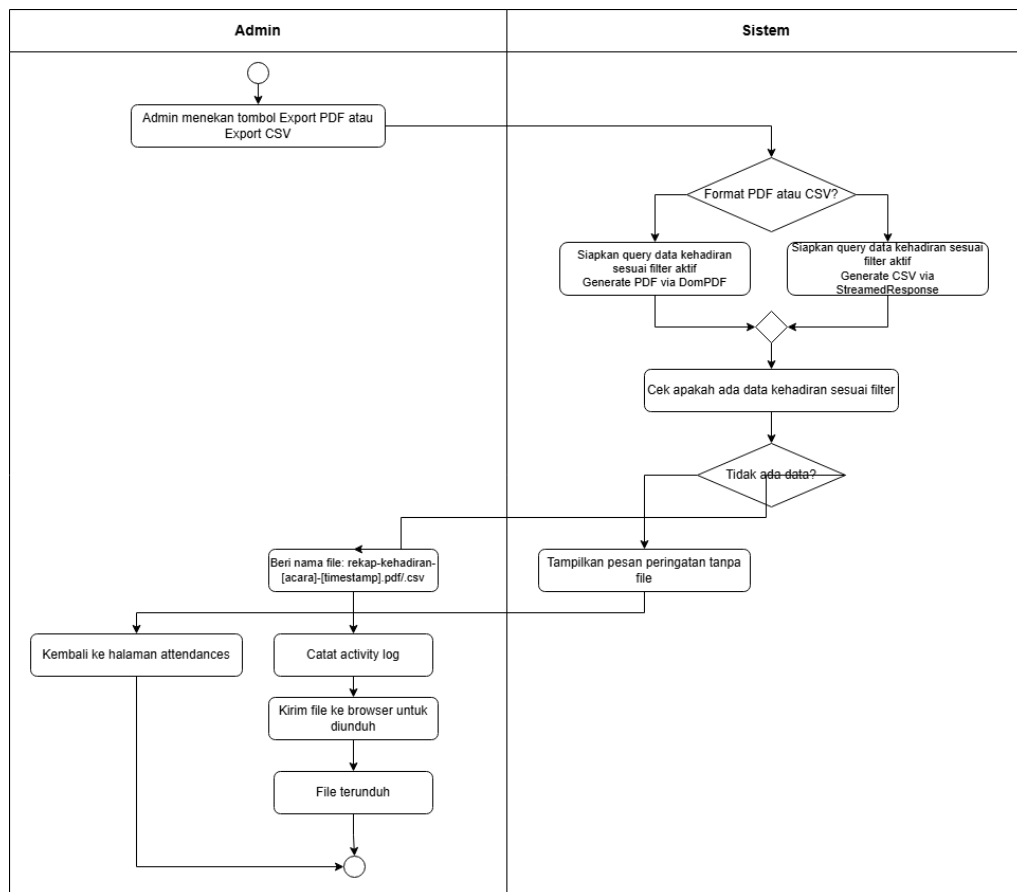
Admin buka halaman, sistem ambil data 50 per halaman dengan statistik; admin bisa filter (acara, tanggal, kata kunci), sistem perbarui tampilan sesuai filter; jika tidak ada data tampilkan "Belum ada data kehadiran".



Gambar 3.10 *Activity Diagram* Lihat Rekap Kehadiran & Statistik

3.2.5.9 *Activity Diagram* Unduh Laporan Kehadiran (PDF/CSV)

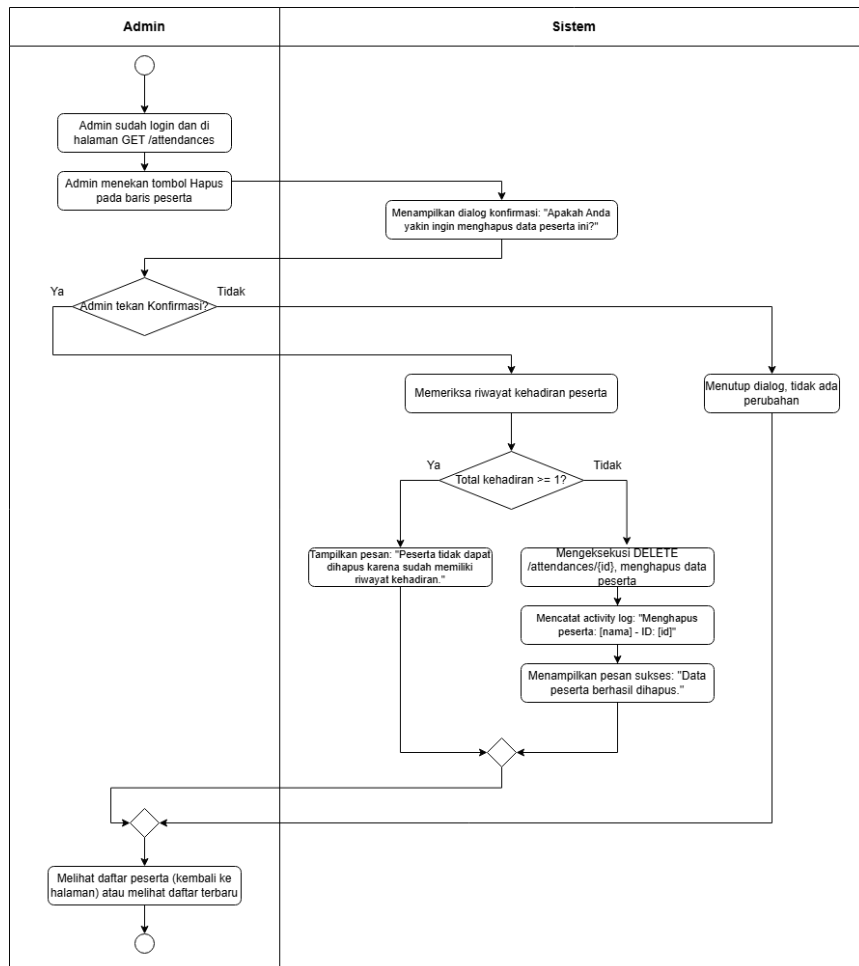
Admin pilih format (PDF/CSV), sistem ambil data sesuai filter aktif; jika tidak ada data tampilkan peringatan tanpa file; jika ada, generate file dengan nama timestamp, catat log, kirim ke *Browser*.



Gambar 3.11 *Activity Diagram* Unduh Laporan Kehadiran (PDF/CSV)

3.2.5.10 *Activity Diagram* Hapus Peserta

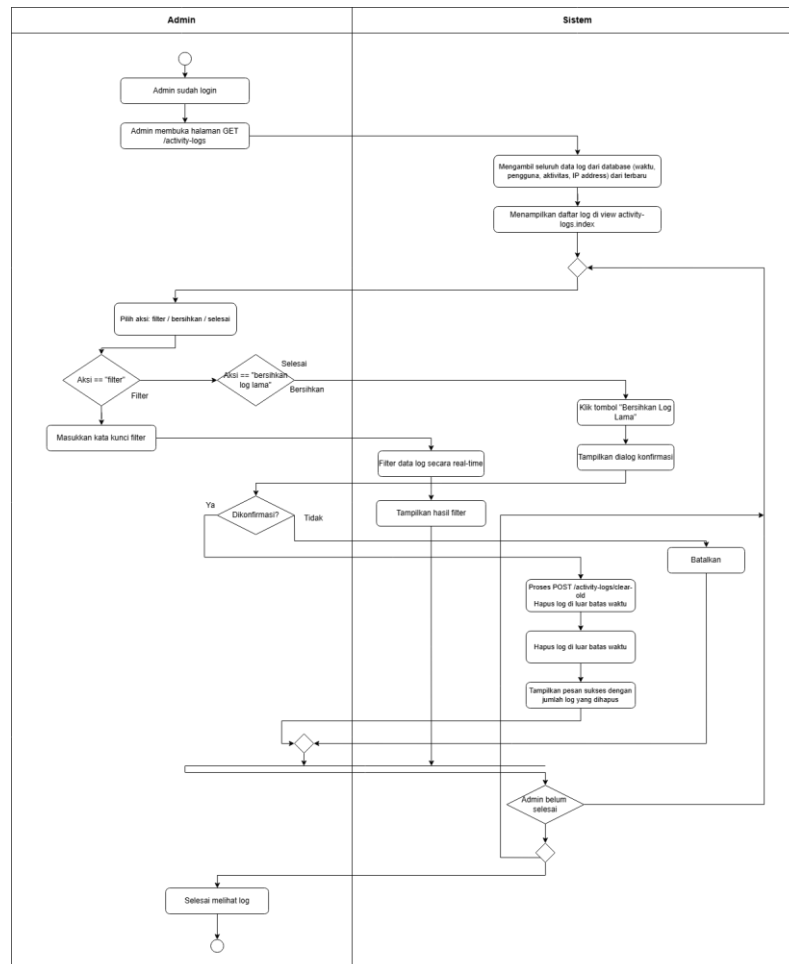
Admin pilih peserta, tekan hapus, konfirmasi; sistem cek total kehadiran; jika ≥ 1 tolak hapus dan tampilkan pesan "Peserta tidak dapat dihapus karena sudah memiliki riwayat"; jika 0, hapus data, catat log, tampilkan sukses, redirect ke daftar (stop). Jika batal konfirmasi, tutup dialog.



Gambar 3.12 *Activity Diagram* Hapus Peserta

3.2.5.11 *Activity Diagram* Pencatatan Activity Log

Admin buka halaman, sistem tampilkan semua log; admin bisa filter kata kunci atau tekan "Bersihkan Log Lama"; jika bersihkan, tampilkan konfirmasi, hapus log di luar batas waktu, tampilkan jumlah terhapus, kembali ke tampilan log



Gambar 3.13 *Activity Diagram* Pencatatan Activity Log

3.2.5.12 *Activity Diagram* Arsipkan Acara

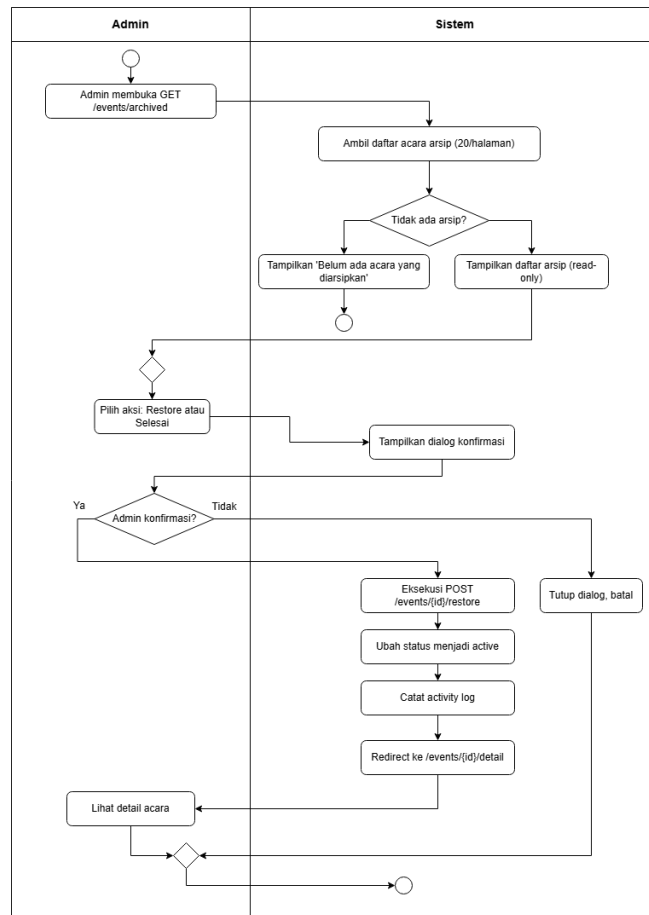
Admin di halaman detail, tekan arsipkan, konfirmasi; sistem verifikasi aktif, cek data kehadiran (tampilkan peringatan jika belum ada data tapi tetap lanjut), ubah status archived & is_active=false, catat log, redirect ke /Events (stop). Jika batal, tutup dialog.



Gambar 3.14 *Activity Diagram* Arsipkan Acara

3.2.5.13 *Activity Diagram* Lihat & Restore Acara Terarsip

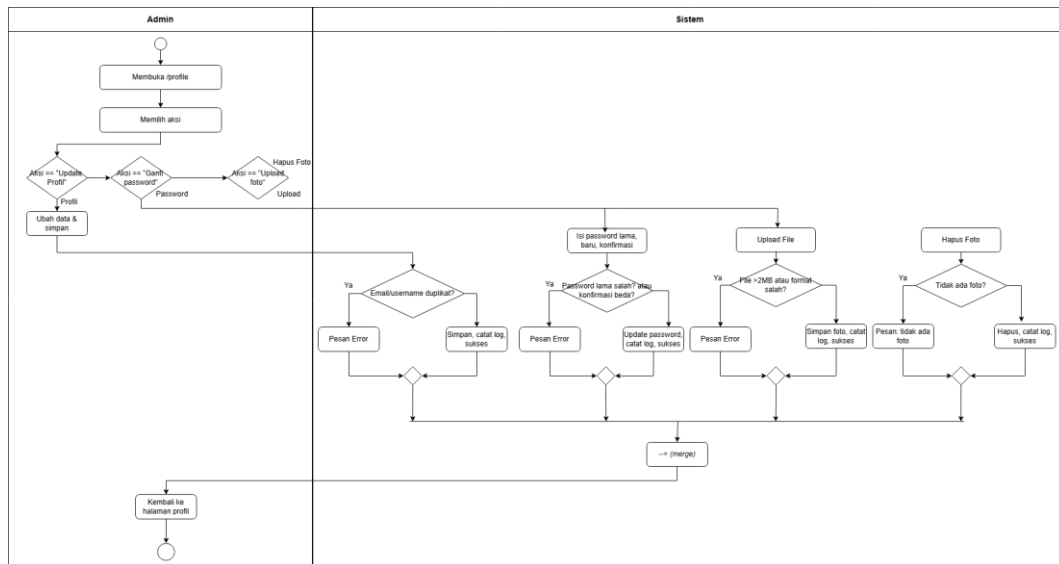
Admin buka halaman arsip; jika kosong tampilkan "Belum ada acara diarsipkan"; jika ada, tampilkan daftar (read-only). Admin pilih restore, konfirmasi; sistem ubah status active, catat log, redirect ke detail acara; jika batal, tutup dialog (stop).



Gambar 3.15 *Activity Diagram* Lihat & Restore Acara Tersaris

3.2.5.14 *Activity Diagram* Kelola Profil Admin

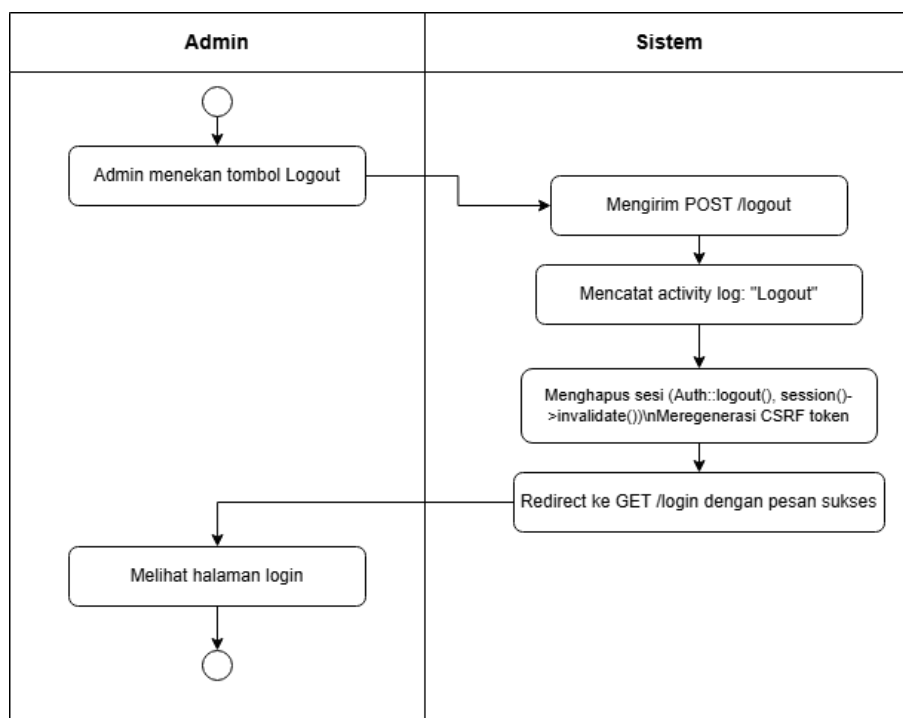
Admin pilih aksi (update profil, ganti *password*, *upload* foto, hapus foto). Untuk profil: ubah data, simpan, validasi unik email/*username*; jika duplikat tampilkan error, jika sukses simpan & catat log. Untuk *password*: isi lama/baru/konfirmasi, validasi (lama cocok, konfirmasi sama), update & catat log. Untuk upload: validasi file ($\leq 2\text{MB}$, JPG/PNG), simpan & log. Untuk hapus: cek ada foto, jika ada hapus & log, jika tidak ada tampilkan pesan. Semua cabang berakhir di halaman profil.



Gambar 3.16 Activity Diagram Kelola Profil Admin

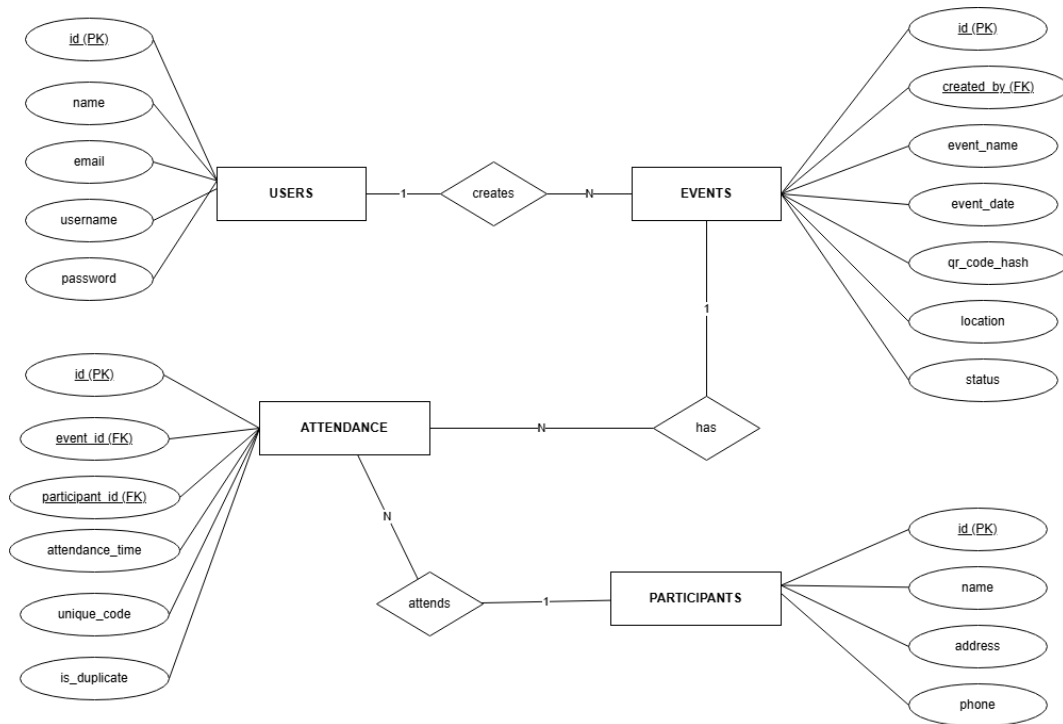
3.2.5.15 Activity Diagram Logout Admin

Admin tekan tombol *Logout*; sistem kirim POST */Logout*, hapus sesi, invalidate session, regenerasi CSRF token, catat log, redirect ke halaman *Login*.



Gambar 3.17 Activity Diagram Logout Admin

3.2.6 ER Diagram



Gambar 3.18 ER Diagram

Berikut merupakan penjelasan singkat dari *ER Diagram* di atas:

1. *Users*

Entitas ***Users*** digunakan untuk menyimpan data administrator yang memiliki hak akses untuk mengelola sistem absensi berbasis *QR code*. Data yang disimpan meliputi nama, email, *username*, dan *password* yang digunakan untuk proses autentikasi. Administrator bertanggung jawab dalam mengelola kegiatan reses, menghasilkan *QR code*, memantau data kehadiran peserta, serta mengelola laporan absensi yang dihasilkan oleh sistem.

2. *Events*

Entitas ***Events*** digunakan untuk menyimpan informasi mengenai kegiatan reses DPRD Kota Batam. Data yang tersimpan meliputi nama kegiatan, tanggal pelaksanaan, lokasi kegiatan, status kegiatan, serta *QR code* yang digunakan sebagai media absensi. Setiap kegiatan memiliki *QR*

code yang berbeda sehingga dapat digunakan untuk membedakan data kehadiran antar kegiatan reses.

3. *Participants*

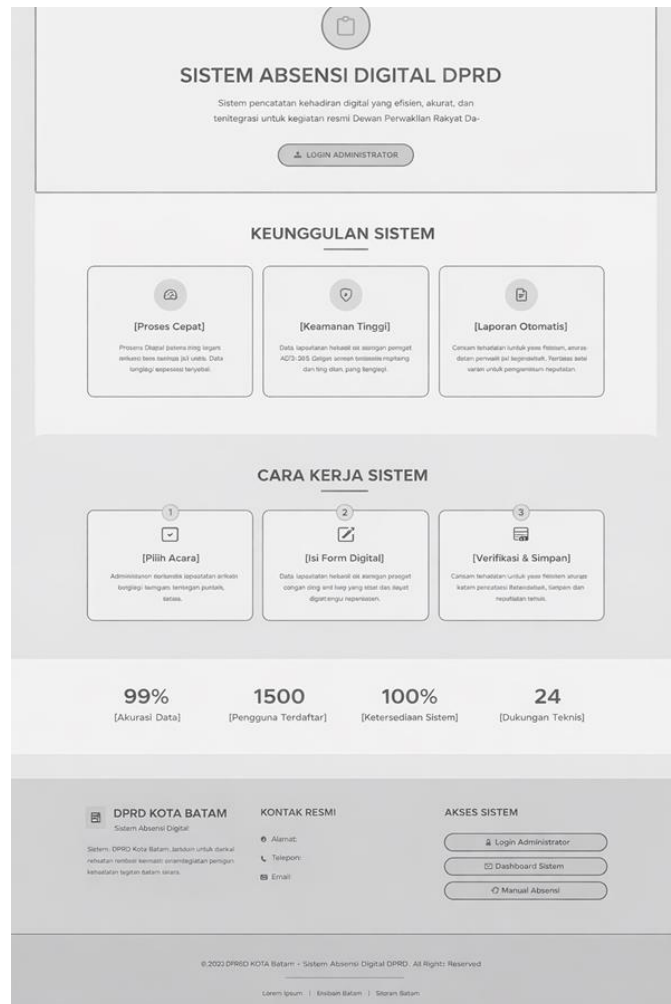
Entitas *Participants* digunakan untuk menyimpan data peserta yang mengikuti kegiatan reses DPRD Kota Batam. Informasi yang disimpan meliputi nama peserta, alamat, dan nomor telepon. Data peserta digunakan sebagai identitas dalam proses absensi dan dapat digunakan kembali apabila peserta mengikuti kegiatan reses lainnya tanpa perlu melakukan pencatatan data dari awal.

4. *Attendance*

Entitas *Attendance* menyimpan *record* kehadiran peserta dalam suatu acara tertentu, mencakup *timestamp* absensi, kode unik verifikasi, *flag* duplikasi, serta *metadata* teknis seperti *IP address* dan *user agent*. *Attendance* berfungsi sebagai *bridge table* yang menghubungkan entitas *Events* dan *Participants* dalam relasi *many-to-many*, sekaligus mengimplementasikan mekanisme anti-duplikasi untuk mencegah kecurangan dalam *presensi*. Setiap *record* kehadiran merepresentasikan satu kali interaksi absensi antara peserta dan acara.

3.2.7 Perancangan Antarmuka (*Wireframe*)

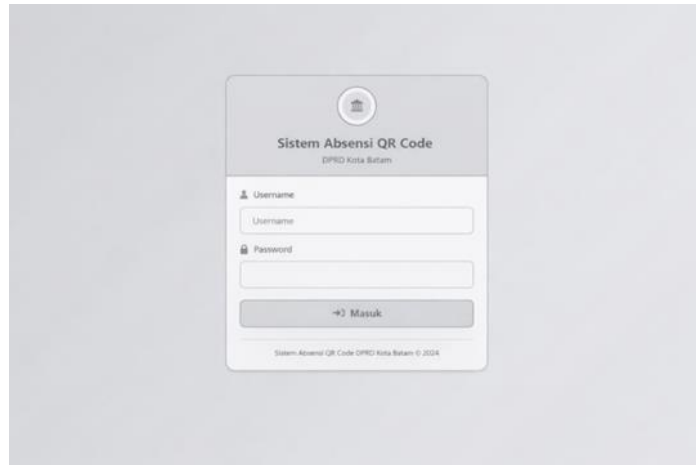
1. *Halaman Beranda*



Gambar 3.19 *Wireframe* Halaman Beranda

Wireframe Halaman Beranda Sistem Absensi *QR code* DPRD Kota Batam. Halaman ini berfungsi sebagai halaman utama yang memberikan gambaran umum mengenai sistem, tujuan penggunaan, serta fitur utama yang tersedia. *Wireframe* menampilkan judul sistem, deskripsi singkat, tombol *Login* administrator, informasi keunggulan sistem, alur cara kerja sistem, serta bagian footer yang berisi identitas instansi dan informasi pendukung.

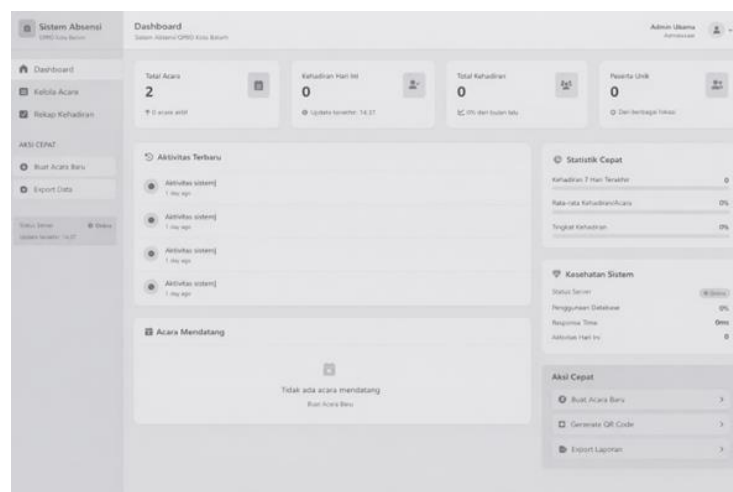
2. Wireframe Login Admin



Gambar 3.20 Wireframe Login Admin

Wireframe Halaman Login Sistem Absensi QR code DPRD Kota Batam. Halaman ini digunakan sebagai proses autentikasi administrator dengan memasukkan *username* dan *password* sebelum mengakses sistem. Wireframe menampilkan *form Login* sederhana yang berfokus pada kemudahan dan keamanan akses pengguna.

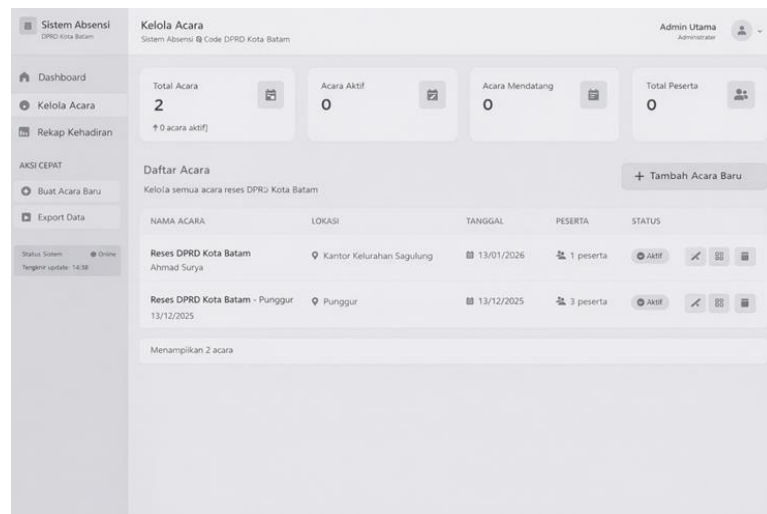
3. Wireframe Dashboard Admin



Gambar 3.21 Wireframe Dashboard Admin

Wireframe Halaman *Dashboard* Administrator. Halaman ini berfungsi sebagai pusat informasi sistem yang menampilkan ringkasan data kehadiran, total acara, aktivitas terbaru, statistik singkat, serta status kesehatan sistem. *Dashboard* dirancang untuk memberikan gambaran kondisi sistem secara cepat dan informatif

4. *Wireframe* Kelola data acara



Gambar 3.22 *Wireframe* Kelola data acara

Wireframe Halaman Kelola Acara. Halaman ini digunakan oleh administrator untuk mengelola data acara, termasuk melihat daftar acara, lokasi, tanggal, jumlah peserta, serta status acara. *Wireframe* juga menyediakan tombol aksi untuk menambah, mengedit, dan menghapus data acara.

5. Wireframe Tambah Acara Baru

The wireframe shows a web application interface for adding a new event. On the left is a sidebar menu with options like Dashboard, Kelola Acara, Rekap Kehadiran, and Aksi Cepat. The main content area is titled 'Tambah Acara Baru' and contains a form with the following fields: 'Nama Acara' (text input), 'Lokasi' (text input with a location pin icon), 'Tanggal Acara' (date picker set to 01/14/2026), and 'Deskripsi Acara' (text area). Below these is a 'Status Acara' section with radio buttons for 'Aktif' and 'Tidak Aktif'. At the bottom of the form are 'Kembali' and 'Simpan Acara' buttons. A 'Tips Membuat Acara' section at the very bottom provides instructions on how to use the system.

Gambar 3.23 Wireframe Tambah Acara Baru

Wireframe Halaman Tambah Acara Baru. Halaman ini berfungsi untuk memasukkan data acara baru ke dalam sistem. *Wireframe* menampilkan *form input* yang terdiri dari nama acara, lokasi, tanggal acara, deskripsi, dan status acara, serta tombol simpan untuk menyimpan data ke dalam sistem.

6. Wireframe QR code Absensi

The wireframe shows the 'QR Code Absensi' page. It features a large QR code in the center for scanning. Below the QR code is a 'URL Absensi' section with a text input field and a 'Salin URL' button. To the right of the QR code is a 'Status Akses Absensi' section showing 'Aktif' status, 'Tanggal Event' (13/01/2026), 'Lokasi' (Kantor Kelurahan Sagulung), and 'Dibuat' (10/01/2026). Below this is an 'Akses Absensi Dibuka' section. The bottom of the page contains a 'Petunjuk Penggunaan' section with instructions on how to use the QR code and a 'Tips Kontrol Akses' section.

Gambar 3.24 Wireframe QR code Absensi

Wireframe Halaman QR code Absensi. Halaman ini digunakan untuk menampilkan QR code yang dapat dipindai oleh peserta sebagai akses absensi.

Wireframe menampilkan *QR code* acara, informasi status akses absensi, *URL* absensi, statistik kehadiran, serta tombol kontrol seperti unduh dan cetak *QR code*.

7. *Wireframe* Rekap kehadiran

FORM KEHADIRAN
Reses DPRD Kota Batam
coba

Periode: 15/01/2026

NAMA LENGKAP *

MASUKKAN NAMA LENGKAP ANDA

ALAMAT LENGKAP *

MASUKKAN ALAMAT LENGKAP ANDA

NOMOR TELEPON/HP *

CONTOH: 081234567890

Format: angka 10-15 digit, tanpa +, -, spasi, atau karakter khusus

☐ KONFIRMASI KEHADIRAN
Data yang Anda isi akan tersimpan secara digital dan digunakan untuk kepentingan dokumentasi serta laporan kehadiran rapat.

SCAN QR CODE
UNTUK AKSES
CEPAT

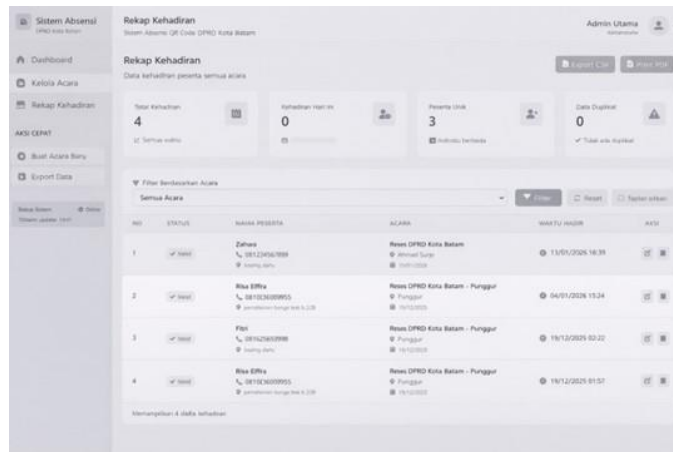
(0778) 123456 | email@dprd-kotabatam.go.id

← Kembali ke Halaman Utama

Gambar 3.25 *Wireframe* Rekap kehadiran

Wireframe Halaman *Form* Kehadiran Peserta. Halaman ini digunakan oleh peserta untuk mengisi data kehadiran setelah memindai *QR code*. *Wireframe* menampilkan *form input* nama lengkap, alamat, nomor telepon, konfirmasi kehadiran, serta tombol simpan kehadiran.

8. Wireframe Laporan akhir



Gambar 3.26 Wireframe Laporan akhir

Wireframe Halaman Rekap Kehadiran. Halaman ini berfungsi untuk menampilkan data kehadiran seluruh peserta dalam bentuk tabel. Wireframe menampilkan ringkasan data kehadiran, fitur *filter* berdasarkan acara, serta tombol *export* data untuk kebutuhan laporan.

9. Wifera Rekap Kehadiran Peserta (Laporan Cetak / PDF)



Gambar 3.27 Wireframe Rekap Kehadiran Peserta (Laporan Cetak / PDF)

Wireframe Halaman Rekap Kehadiran Peserta dalam format laporan cetak. Halaman ini menyajikan data kehadiran peserta secara terstruktur dan formal, meliputi identitas peserta, acara, waktu kehadiran, dan status kehadiran, yang digunakan sebagai dokumentasi resmi sistem.

BAB IV IMPLEMENTASI DAN PENGUJIAN

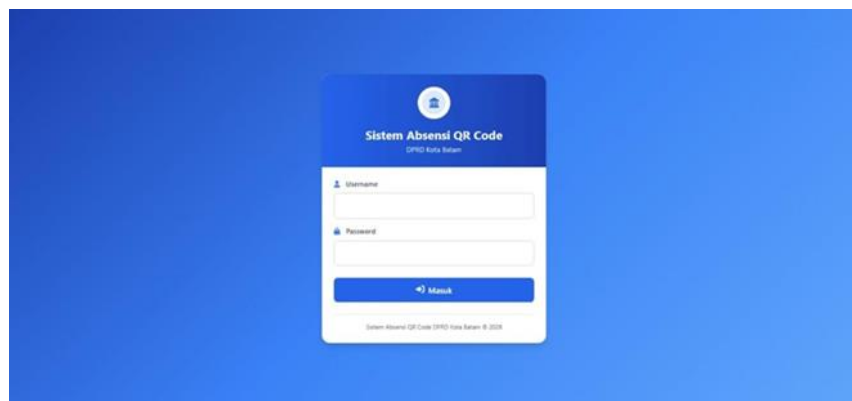
4.1. Hasil Implementasi

Sistem Absensi Berbasis *QR code* untuk DPRD Kota Batam diimplementasikan sebagai aplikasi berbasis web menggunakan *Framework Laravel* 10 dengan basis data *MySQL*. Proses implementasi dilakukan pada lingkungan pengembangan lokal menggunakan *XAMPP* sebagai web *server*. Pengelolaan dependensi pustaka menggunakan *Composer*, konfigurasi environment dikelola melalui file *.env*, pembuatan *QR code* memanfaatkan layanan *QuickChart.io*, ekspor PDF menggunakan pustaka *DomPDF*, dan ekspor CSV menggunakan *Laravel StreamedResponse*.

4.1.1 Antarmuka Admin

a. Halaman *Login*

Halaman *Login* dapat diakses melalui alamat */Login* dan merupakan pintu masuk utama sistem. Admin mengisi *username* dan *password*, lalu menekan tombol *Masuk* untuk mengakses sistem. Sistem memvalidasi kredensial dan mengarahkan Admin ke halaman *Dashboard* apabila berhasil, atau menampilkan pesan kesalahan apabila gagal.

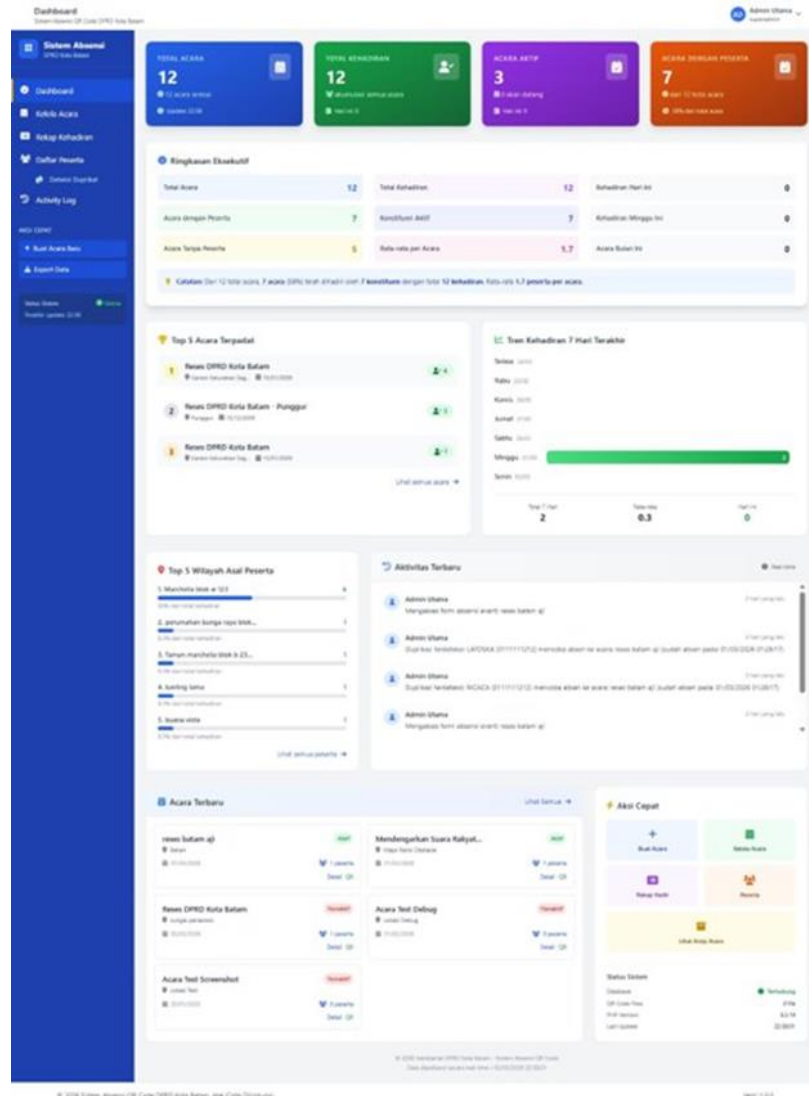


Gambar 4.1 Halaman *Login*

b. Halaman *Dashboard*

Halaman *Dashboard* menampilkan ringkasan statistik sistem, meliputi jumlah total acara, total kehadiran, dan jumlah acara yang

sedang aktif. Halaman ini hanya dapat diakses oleh pengguna yang telah *Login*; akses tanpa autentikasi akan dialihkan secara otomatis ke halaman *Login* oleh *middleware Laravel*.

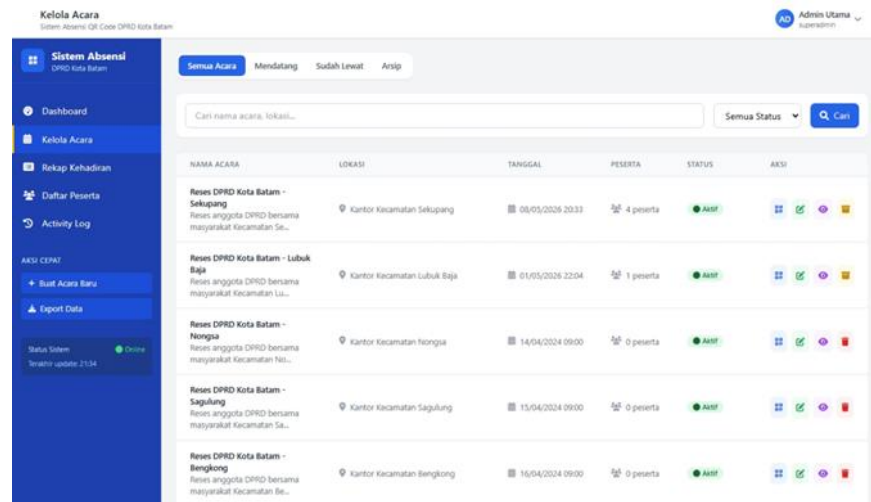


Gambar 4.2 Halaman *Dashboard*

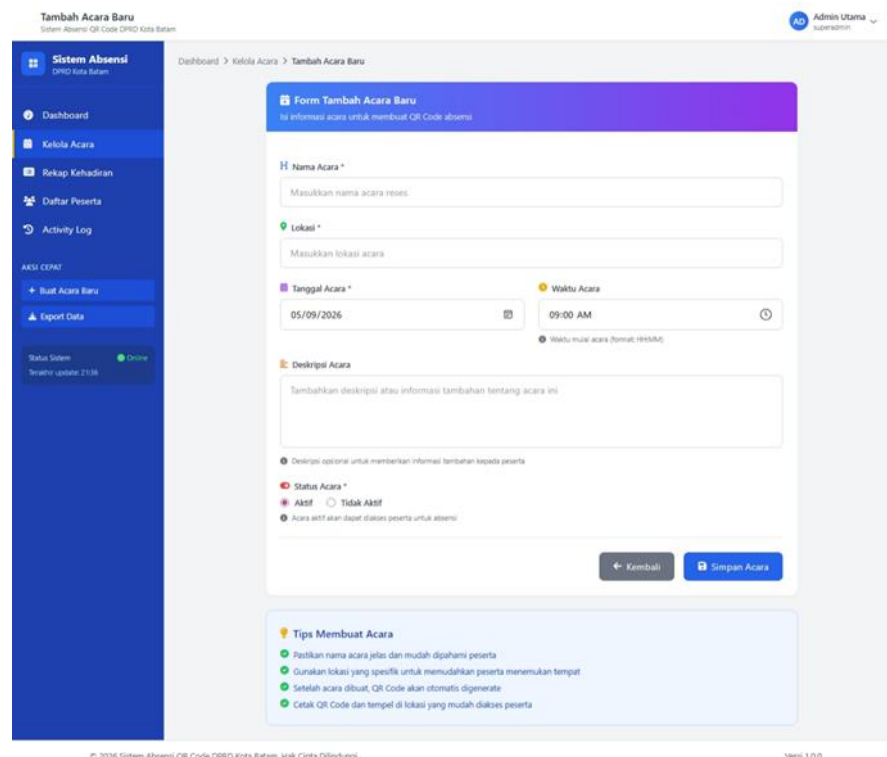
c. Halaman Kelola Acara

Halaman ini dapat diakses melalui */Events* dan menampilkan daftar seluruh acara reses yang telah dibuat beserta statusnya. Admin dapat membuat acara baru, mengedit acara yang belum diarsipkan, serta menghapus acara yang belum memiliki data kehadiran. Setiap acara

baru yang berhasil disimpan akan secara otomatis menghasilkan *QR code* unik berbasis *UUID*.



Gambar 4.3 Halaman Daftar Acara

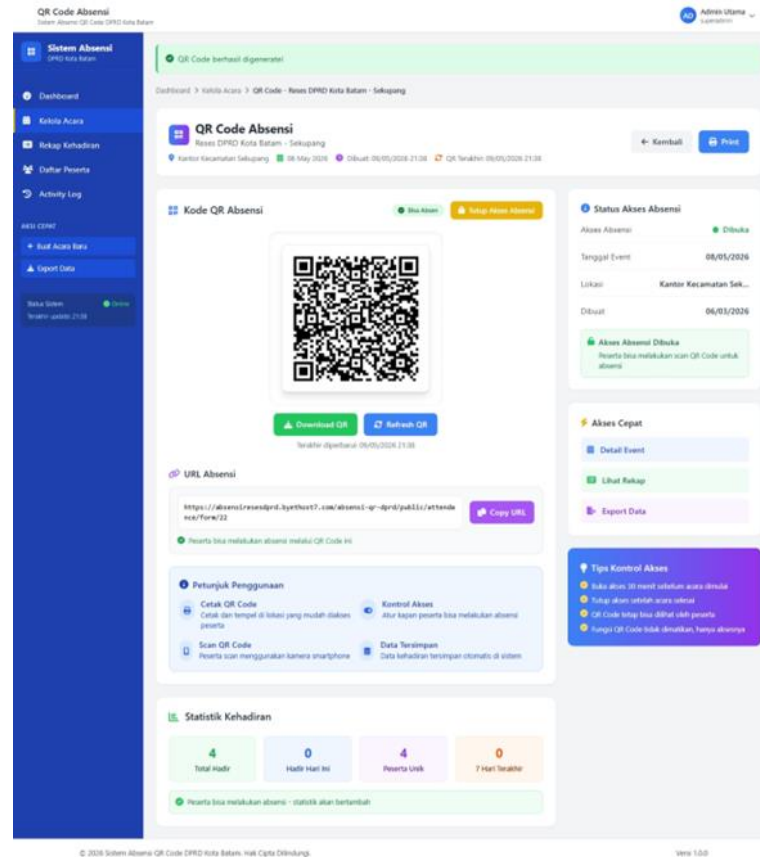


Gambar 4.4 Halaman Buat Acara Baru

d. Halaman *QR code* Acara

Halaman ini dapat diakses melalui `/Events/{id}/qr` dan menampilkan *QR code* beserta URL absensi acara yang bersangkutan. Admin dapat melakukan *refresh QR code* untuk menghasilkan *hash* baru,

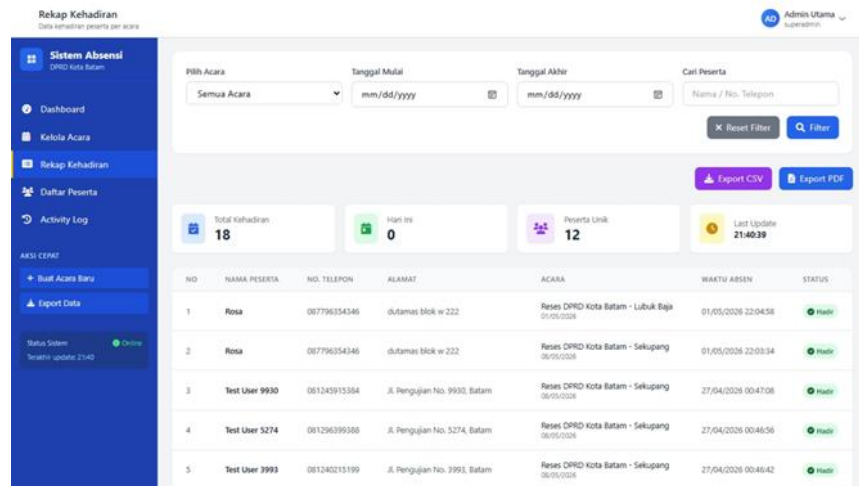
mengunduh *QR code* dalam format PNG, menyalin URL absensi ke *Clipboard*, serta membuka atau menutup akses absensi peserta secara *real-time* tanpa memuat ulang halaman.



Gambar 4.5 Halaman *QR code* Acara

e. Halaman Rekap Kehadiran

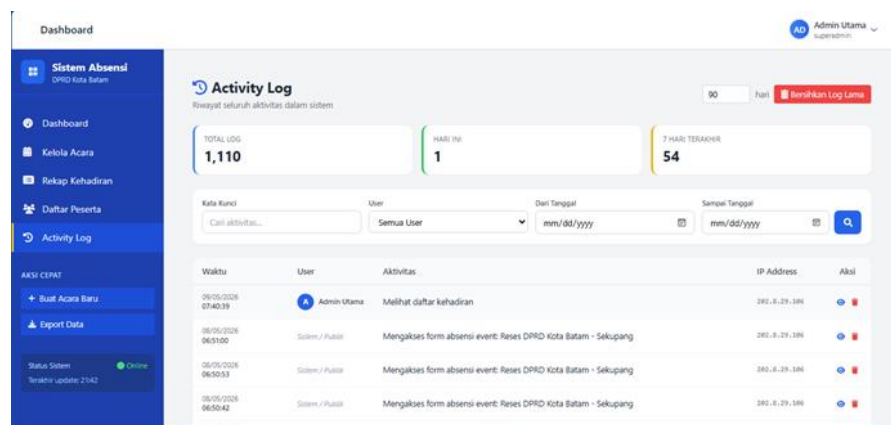
Halaman rekap kehadiran dapat diakses melalui `/attendances` dan menampilkan seluruh data kehadiran dari semua acara dalam satu tabel terpadu dengan 50 data per halaman. Terdapat kartu statistik di bagian atas yang menampilkan total kehadiran, jumlah kehadiran hari ini, dan jumlah peserta unik. Data dapat difilter berdasarkan acara, rentang tanggal, atau kata kunci, serta diekspor dalam format PDF maupun CSV.



Gambar 4.6 Halaman Rekap Kehadiran

f. Halaman Activity Log

Halaman *activity log* dapat diakses melalui `/activity-logs` dan mencatat seluruh riwayat aktivitas yang terjadi dalam sistem secara otomatis, mencakup waktu aktivitas, nama pengguna, deskripsi aktivitas, dan alamat IP. Admin dapat melakukan pencarian berdasarkan kata kunci serta membersihkan log lama melalui fitur *clear old logs*.

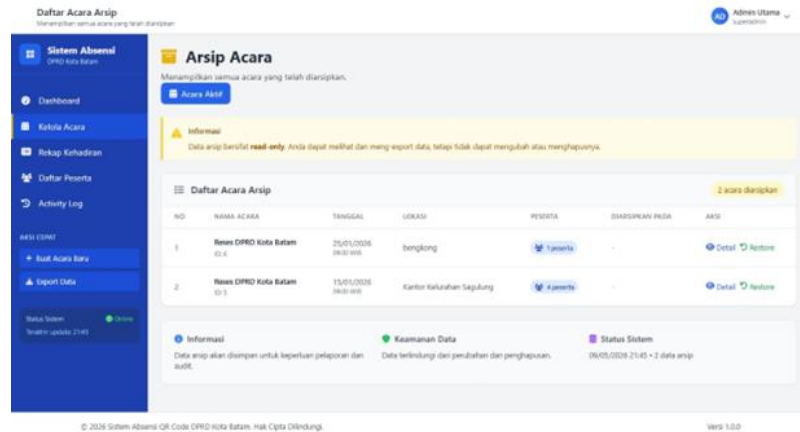


Gambar 4.7 Halaman Activity Log

g. Halaman Arsip Acara

Halaman arsip dapat diakses melalui `/Events/archived` dan menampilkan daftar acara yang telah diarsipkan dalam mode *read-only* beserta jumlah peserta dan tanggal pengarsipan. Admin dapat

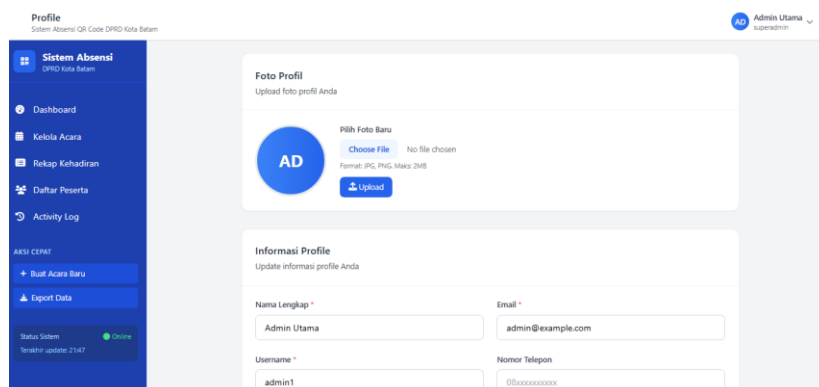
memulihkan acara dari arsip melalui tombol Restore, namun acara yang dipulihkan memiliki `is_active = false` sehingga akses absensi tetap perlu dibuka secara manual.



Gambar 4.8 Halaman Arsip Acara

h. Halaman Profil Admin

Halaman profil dapat diakses melalui `/profile` dan memungkinkan Admin memperbarui data diri seperti nama, email, dan *username*. Terdapat pula fitur penggantian *password* dengan verifikasi *password* lama, serta pengelolaan foto profil yang mendukung unggah dan penghapusan file berformat JPG atau PNG dengan ukuran maksimal 2 MB.



Gambar 4.9 Halaman Profil Admin

4.1.2 Antarmuka Peserta Reses

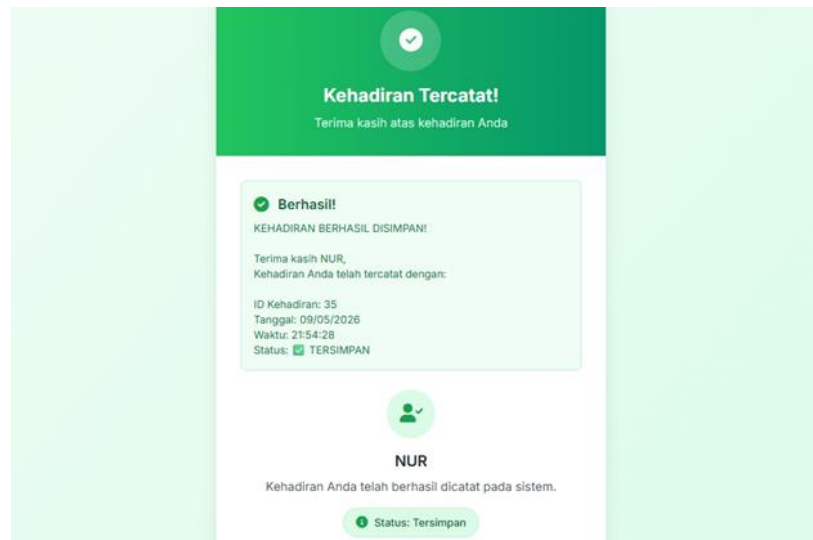
a. Halaman Form Absensi

Peserta mengakses halaman form absensi melalui `/absensi/{qr_hash}` dengan cara memindai *QR code* acara menggunakan kamera ponsel. Sistem memvalidasi keabsahan *hash*, status aktif acara, dan tanggal acara sebelum menampilkan formulir. Peserta mengisi nama lengkap, alamat, dan nomor telepon, lalu menekan tombol Simpan Kehadiran untuk menyelesaikan absensi.

Gambar 4.10 Halaman Form Absensi Peserta

b. Halaman Konfirmasi Absensi Berhasil

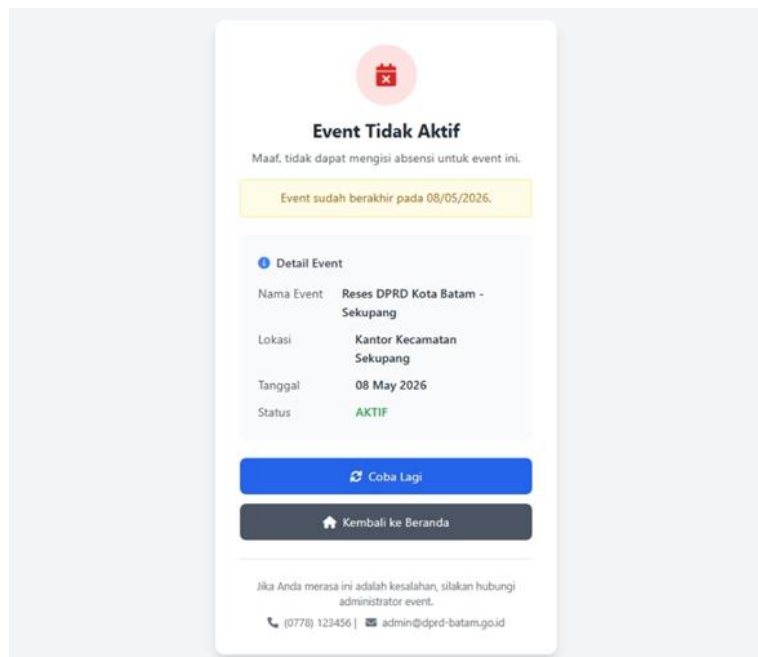
Setelah data kehadiran berhasil disimpan, peserta diarahkan ke halaman konfirmasi melalui `/attendance/success/{Event}/{attendance}` yang menampilkan ringkasan data absensi beserta waktu pencatatan. Sistem secara otomatis mencatat waktu kehadiran, alamat IP, dan informasi perangkat pada saat formulir disubmit.



Gambar 4.11 Halaman Konfirmasi Absensi Berhasil

c. **Halaman *QR code* Tidak Valid / Acara Berakhir**

Apabila peserta mengakses URL absensi dengan *hash* yang tidak valid, acara yang tidak aktif, atau acara yang telah melewati tanggalnya, sistem menampilkan halaman *expired* dengan pesan yang sesuai kondisi masing-masing. Halaman ini mencegah peserta mengisi formulir absensi di luar kondisi yang diizinkan.

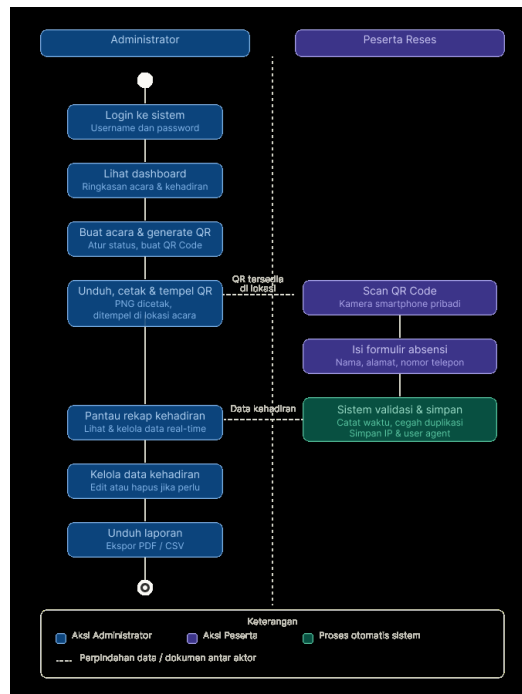


Gambar 4.12 Halaman *QR code* Tidak Valid / Acara Berakhir

4.1.3 Alur Penggunaan Sistem

Alur penggunaan sistem dapat dijelaskan secara berurutan:

1. Admin *Login* ke sistem, membuat acara reses baru, dan sistem otomatis menghasilkan *QR code* unik.
2. Admin membuka akses absensi, lalu mengunduh gambar *QR code* (PNG) dan mencetaknya (satu lembar per acara).
3. Panitia menempelkan cetakan *QR code* di lokasi strategis (meja registrasi, pintu masuk, dll.).
4. Peserta memindai *QR code* menggunakan kamera *smartphone*, mengisi formulir data diri, lalu menyimpan kehadiran.
5. Data kehadiran tersimpan langsung dan Admin dapat memantau secara *real-time* di halaman Rekap Kehadiran.
6. Setelah acara selesai, Admin mengarsipkan acara dan mengunduh laporan (PDF/CSV) sebagai dokumen resmi.



Gambar 4.13 Alur Penggunaan Sistem Secara Keseluruhan

4.2 Pengujian

Pengujian sistem dilakukan untuk memverifikasi bahwa seluruh fungsi yang diimplementasikan bekerja sesuai dengan rancangan dan

spesifikasi yang telah ditetapkan. Berdasarkan karakteristik proyek yang berupa pengembangan perangkat lunak berbasis web, terdapat dua jenis pengujian yang dilakukan, yaitu Pengujian Fungsional (*Black-Box Testing*) dan Pengujian Kinerja Dasar (*Load Testing*) menggunakan Locust.

4.2.1 Pengujian Fungsional (*Black-Box Testing*)

Pengujian fungsional dilakukan dengan metode *Black-Box Testing*, yaitu pengujian yang berfokus pada perilaku keluaran sistem terhadap masukan tertentu tanpa memperhatikan implementasi internal kode. Pengujian ini bertujuan untuk memastikan seluruh fungsi sistem berjalan sesuai dengan skenario yang telah dirancang pada *Use Case*. Pengujian dilakukan oleh peneliti secara langsung dengan menjalankan setiap skenario uji dan mencatat hasilnya pada tabel pengujian.

Pengujian fungsional mencakup 15 modul yang terbagi ke dalam 52 skenario uji (*test case*). Setiap skenario dievaluasi berdasarkan kesesuaian antara hasil yang diharapkan (*expected result*) dengan hasil aktual yang diperoleh, dengan status **B** (Berhasil) apabila sesuai dan **G** (Gagal) apabila tidak sesuai.

Tabel 4.1 *Black-Box Testing*

Test Case ID	Skenario Pengujian	Luaran yang Diharapkan	Data Uji	Luaran yang Dihasilkan	B	G
Modul 01: Autentikasi & Login Admin						
TC01	Login dengan <i>username</i> & <i>password</i> valid	Isi <i>username</i> dan <i>password</i> , klik <i>Login</i>	<i>username</i> =admin1, <i>password</i> =12345678	Redirect ke halaman <i>Dashboard</i>	✓	
TC02	Login dengan <i>password</i> salah	Isi <i>username</i> benar, <i>password</i> salah	<i>username</i> =admin1, <i>password</i> =salah	Muncul pesan " <i>Password salah</i> "	✓	

TC03	Login dengan <i>username</i> tidak terdaftar	Isi <i>username</i> tidak ada	<i>username</i> =tidakada, <i>password</i> =123	Muncul pesan " <i>Username</i> tidak terdaftar."	✓	
TC04	<i>Field username</i> kosong	Sistem Biarkan <i>username</i> kosong pesan error autentikasi, admin tidak dapat masuk	<i>username</i> = , <i>password</i> =12345678	Validasi " <i>Please fill out this Field</i> "	✓	
TC05	<i>Field password</i> kosong	Biarkan <i>password</i> kosong	<i>username</i> =admin1, <i>password</i> =	Validasi muncul	✓	
TC06	Kedua <i>Field</i> kosong	Kosongkan kedua <i>Field</i>	kosong	Validasi muncul untuk kedua <i>Field</i>	✓	
Modul 02: Kelola Data Acara Reses (CRUD)						
TC07	Membuat acara baru dengan data valid	Klik Buat Acara, isi data, simpan	Nama="Rapat Paripurna", tanggal=besok	Acara tersimpan, <i>QR code</i> tergenerate	✓	
TC08	Membuat acara dengan nama kurang dari 3 karakter	Isi nama hanya 2 karakter	Nama="Ra"	Validasi "Nama minimal 3 karakter"	✓	
TC09	Membuat acara dengan tanggal lampau	Pilih tanggal kemarin	Tanggal=kemarin	Validasi "Tanggal harus $\geq$ hari ini"	✓	
TC10	Mengedit acara yang belum diarsipkan	Edit acara aktif	Ubah nama	Data berhasil diperbarui	✓	
TC11	Menghapus acara tanpa	Hapus acara kosong	Acara tanpa peserta	Acara terhapus	✓	

	data ke- hadiran					
TC12	Menghapus acara yang sudah ada kehadiran	Hapus acara dengan peserta	Acara dengan kehadiran	Tolak hapus, saran arsipkan	✓	
Modul 03: Generate & Refresh QR code						
TC13	<i>Refresh QR code</i>	Klik tombol "Refresh QR"	Acara aktif	QR code baru tergenerate dengan hash berbeda	✓	
TC14	Unduh QR code sebagai PNG	Klik tombol "Unduh QR"	-	File PNG berhasil diunduh	✓	
TC15	Salin URL absensi	Klik tombol "Salin URL"	-	URL berhasil disalin ke <i>Clipboard</i>	✓	
Modul 04: FR-04 Buka / Tutup Akses Absensi						
TC16	Membuka akses absensi acara	Klik tombol "Buka Akses", konfirmasi	Acara aktif	Status is_active=true, badge "Bisa Absen"	✓	
TC17	Menutup akses absensi acara	Klik tombol "Tutup Akses", konfirmasi	Acara aktif	Status is_active=false, badge "Tidak Bisa Absen"	✓	
TC18	Batal konfirmasi toggle status	Klik toggle lalu pilih Batal	Acara aktif	Tidak ada perubahan status	✓	
Modul 05: Scan QR code oleh Peserta						
TC19	Scan QR acara yang aktif	Scan QR code dengan kamera HP	Acara aktif	Redirect ke form absensi	✓	
TC20	Scan QR acara yang tidak aktif	Scan QR yang sudah ditutup	Acara non-aktif	Halaman "Event sudah tidak aktif"	✓	
TC21	Scan QR acara yang sudah lewat tanggal	Scan QR expired	Acara tanggal lewat	Halaman "Acara Sudah Berakhir"	✓	

Modul 06: Mengisi & Mengirim Form Absensi						
TC22	Submit form absensi valid	Isi semua field, centang konfirmasi, klik Simpan	Nama valid, telepon 10-15 digit	Redirect ke halaman sukses, data tersimpan	✓	
TC23	Submit dengan nomor telepon duplikat	Masukkan telepon yang sudah absen di acara sama	Telepon terdaftar sebelumnya	Muncul pesan "ANDA SUDAH ABSEN SEBELUMNYA!"	✓	
TC24	Submit dengan field kosong	Biarkan nama kosong	Nama=""	Validasi "Please fill out this field"	✓	
TC25	Submit dengan format telepon salah	Masukkan huruf pada field telepon	Telepon="abc123"	Validasi "Nomor telepon harus 10-15-digit angka"	✓	
TC26	Submit saat acara tidak aktif	Akses form lalu submit setelah acara ditutup	Acara is_active=false	Pesan "Acara tidak aktif"	✓	
Modul 07: Pencatatan Waktu & Informasi Teknis Kehadiran						
TC27	Cek pencatatan waktu & informasi teknis setelah absensi	Lakukan absensi sukses lalu cek <i>Database</i>	Absensi valid	Kolom attendance_time, ip_address, user_agent terisi otomatis	✓	
Modul 08: Lihat Rekap Kehadiran & Statistik						
TC28	Buka halaman rekap tanpa filter	Akses /attendances	-	Tampil statistik + tabel data	✓	
TC29	Filter rekap berdasarkan acara	Pilih acara dari dropdown	Pilih salah satu acara	Hanya data acara tersebut yang muncul	✓	
TC30	Filter berdasarkan rentang tanggal	Isi tanggal mulai dan selesai	date_from=2025-01-01, date_to=2025-12-31	Data sesuai range tanggal	✓	

TC31	Filter berdasarkan kata kunci	Isi kata kunci di pencarian	Kata kunci="08123"	Tampil data yang mengandung kata kunci	✓	
TC32	Tidak ada data kehadiran	Buka rekap acara kosong	Acara tanpa peserta	Pesan "Belum ada data kehadiran"	✓	
Modul 09: Unduh Laporan Kehadiran (PDF & CSV)						
TC33	Export PDF saat ada data	Klik "Export PDF"	Filter dengan data	File PDF terunduh dengan nama sesuai	✓	
TC34	Export CSV saat ada data	Klik "Export CSV"	Filter dengan data	File CSV terunduh (UTF-8)	✓	
TC35	Export saat tidak ada data	Klik export pada filter kosong	Filter tanpa data	Peringatan "Tidak ada data yang sesuai filter"	✓	
Modul 10: Edit & Hapus Data Kehadiran						
TC36	Hapus peserta dengan total kehadiran 0	Pilih peserta baru, klik Hapus	Peserta tanpa riwayat	Peserta terhapus	✓	
TC37	Hapus peserta yang sudah memiliki riwayat kehadiran	Pilih peserta yang pernah absen	Peserta dengan kehadiran ≥ 1	Tolak hapus + pesan peringatan	✓	
TC38	Batalkan konfirmasi hapus	Klik Hapus lalu pilih Batal	-	Tidak ada perubahan	✓	
Modul 11: Pencatatan Activity Log						
TC39	Lihat halaman activity log	Akses /activity-logs	Ada log aktivitas	Tampil daftar log terbaru	✓	
TC40	Bersihkan log lama	Klik "Bersihkan Log Lama", konfirmasi	Log di luar batas waktu	Log terhapus, tampil jumlah terhapus	✓	

MODUL 12: Arsipkan Acara						
TC41	Arsipkan acara yang memiliki data kehadiran	Klik tombol Arsipkan, konfirmasi	Acara dengan peserta	Status archived, is_active=false, muncul di daftar arsip	✓	
TC42	Arsipkan acara tanpa data kehadiran	Klik tombol Arsipkan	Acara tanpa peserta	Tetap bisa diarsipkan dengan peringatan	✓	
Modul 13: Lihat & Restore Acara Terarsip						
TC43	Buka halaman arsip (ada data)	Akses /Events/archived	Ada acara arsip	Tampil daftar arsip read-only	✓	
TC44	Restore acara dari arsip	Klik Restore pada acara arsip, konfirmasi	Acara arsip	Acara aktif kembali, is_active=false, perlu buka akses manual	✓	
Modul 14: Kelola Profil Admin						
TC45	Ubah nama dan email unik	Ubah nama & email baru	Email baru unik	Profil diperbarui	✓	
TC46	Ubah email yang sudah digunakan	Masukkan email milik admin lain	Email duplikat	Validasi "Email sudah digunakan"	✓	
TC47	Ganti password dengan benar	Isi password lama benar + password baru	Password lama benar	Password berhasil diubah	✓	
TC48	Ganti password dengan password lama salah	Password lama salah	Password lama salah	Pesan "Password saat ini tidak sesuai"	✓	
TC49	Upload foto profil valid	Upload file JPG/PNG ≤2MB	Foto valid	Foto tersimpan dan tampil	✓	
TC50	Hapus foto profil	Klik "Hapus Foto"	Ada foto	Foto terhapus	✓	

Modul 15: <i>Logout</i>						
TC51	Klik tombol <i>Logout</i>	Klik <i>Logout</i> di menu	-	Sesi dihapus, redirect ke halaman <i>Login</i>	✓	
TC52	Akses halaman setelah <i>Logout</i> (keamanan)	Coba akses <i>/Dashboard</i> setelah <i>Logout</i>	-	Redirect ke halaman <i>Login</i>	✓	

Tabel 4.2 Rekapitulasi Hasil Pengujian Fungsional

No	Modul	Jumlah TC	Berhasil	Gagal
1.	FR-01 Autentikasi & <i>Login</i> Admin	6	6	0
2.	FR-02 Kelola Data Acara Reses (CRUD)	6	6	0
3.	FR-03 Generate & <i>Refresh QR code</i>	3	3	0
4.	FR-04 Buka / Tutup Akses Absensi	3	3	
5.	FR-05 <i>Scan QR code</i> oleh Peserta	3	3	0
6.	FR-06 Mengisi & Mengirim Form Absensi	5	5	0
7.	FR-07 Pencatatan Waktu & Informasi Teknis Kehadiran	1	1	0
8.	FR-08 Lihat Rekap Kehadiran & Statistik	5	5	0
9.	FR-09 Unduh Laporan Kehadiran (PDF & CSV)	3	3	0
10.	FR-10 Edit & Hapus Data Kehadiran	3	3	0
11.	FR-11 Pencatatan Activity Log	2	2	0
12.	FR-12 Arsipkan Acara	2	2	0
13.	FR-13 Lihat & Restore Acara Tersip	2	2	0
14.	FR-14 Kelola Profil Admin	6	6	0
15.	FR-15 <i>Logout</i> Admin	2	2	0

Total	52	52	0
--------------	-----------	-----------	----------

4.2.2 Pengujian Kinerja Dasar (*Load Testing*)

Pengujian kinerja dilakukan menggunakan aplikasi Locust untuk mengetahui kemampuan sistem dalam menangani akses pengguna secara bersamaan. Pengujian dilakukan pada lingkungan localhost dengan mensimulasikan aktivitas administrator dan peserta absensi yang mengakses sistem secara simultan.

A. Spesifikasi Lingkungan Pengujian

Tabel 4.3 Spesifikasi Perangkat Keras

Komponen	Spesifikasi
Perangkat	ASUS A416MA / X415MA
Processor	Intel(R) Celeron(R) N4020 CPU @ 1.10 GHz
Jumlah Core	2 Core
Jumlah Thread	2 Threads
Base Speed	1.10 GHz
Max Turbo Speed	2.80 GHz
RAM	8.00 GB DDR4
Storage	256 GB SSD (NVMe)
Sistem Operasi	Windows 11 64-bit
Virtualization	Enabled

Tabel 4.4 Spesifikasi Perangkat Lunak

Perangkat Lunak	Versi	Fungsi
XAMPP	3.3.2	<i>Server</i> lokal (Apache, <i>MySQL</i> , PHP)
PHP	8.2.4	Bahasa pemrograman backend
<i>MySQL</i> / MariaDB	10.4.28	<i>Database</i> management system
<i>Laravel</i>	12.38.1	<i>Framework</i> PHP
Locust	2.37.14	<i>Load Testing</i> tool

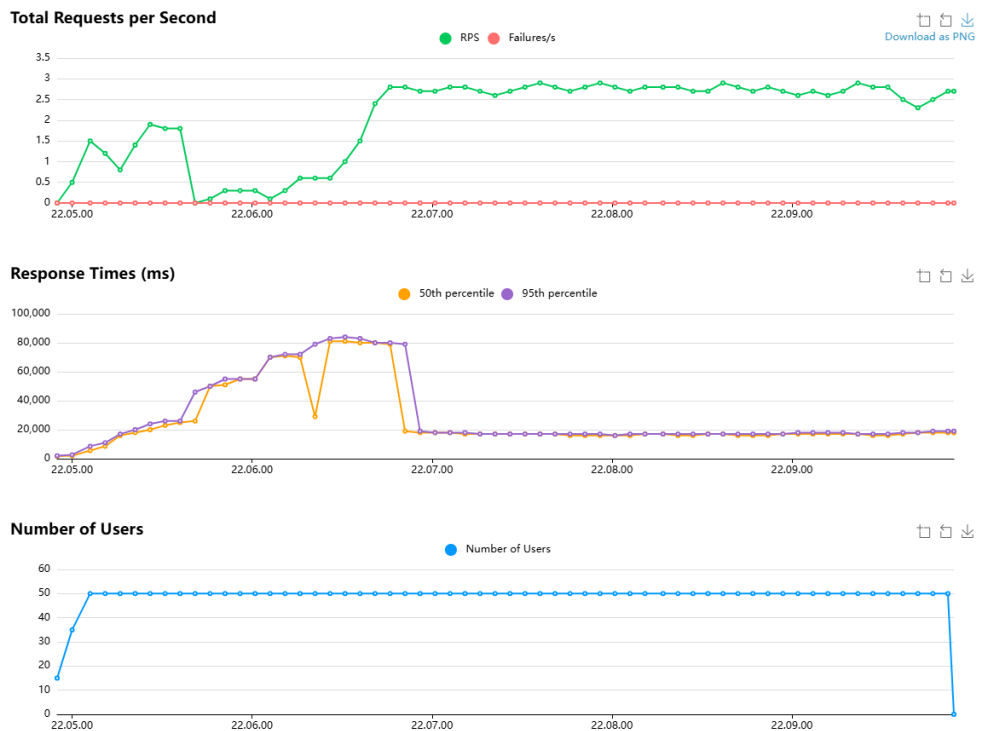
B. Hasil Pengujian

1. Pengujian Administrator (50 Pengguna Virtual)

Pengujian pertama dilakukan menggunakan 50 pengguna virtual yang mengakses sistem sebagai administrator secara bersamaan selama 5 menit.

Tabel 4.5 Hasil Pengujian Administrator (50 User)

Parameter	Hasil	Keterangan
Jumlah Pengguna Virtual	50 User	-
Total Request	630	-
Total Failure	0	-
Success Rate	100%	Semua <i>request</i> berhasil diproses
Failure Rate	0%	Tidak ada kegagalan
Rata-rata <i>Response Time</i>	21.552,48 ms	TIDAK LAYAK (jauh di atas batas toleransi 2-5 detik)



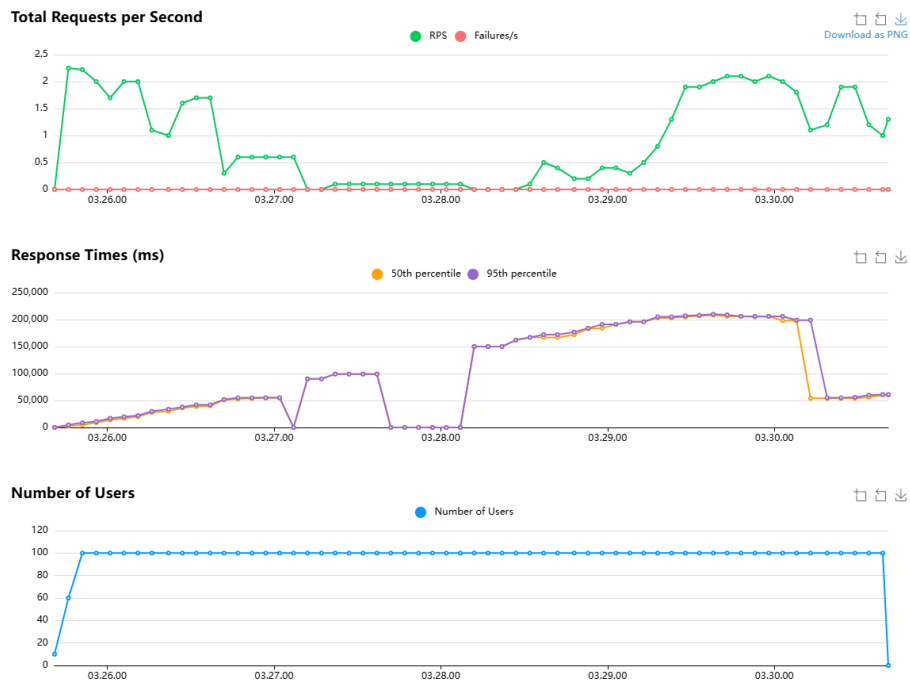
Gambar 4.14 Hasil *Load Testing* Administrator (50 User) pada Locus

2. Pengujian Peserta Absensi (50 Pengguna Virtual)

Pengujian dilakukan dengan mensimulasikan peserta yang mengakses formulir absensi dan melakukan proses absensi secara bersamaan.

Tabel 4.6 Hasil Pengujian Peserta (50 User)

Parameter	Hasil	Keterangan
Jumlah Pengguna Virtual	50 User	-
Total Request	654	-
Total Failure	0	-
Success Rate	100%	Semua <i>request</i> berhasil di-proses
Failure Rate	0%	Tidak ada kegagalan
Rata-rata <i>Response Time</i>	25.013,69 ms	TIDAK LAYAK (jauh di atas batas toleransi 2-5 detik)



Gambar 4.15 Hasil *Load Testing* Administrator (50 User) pada Locust

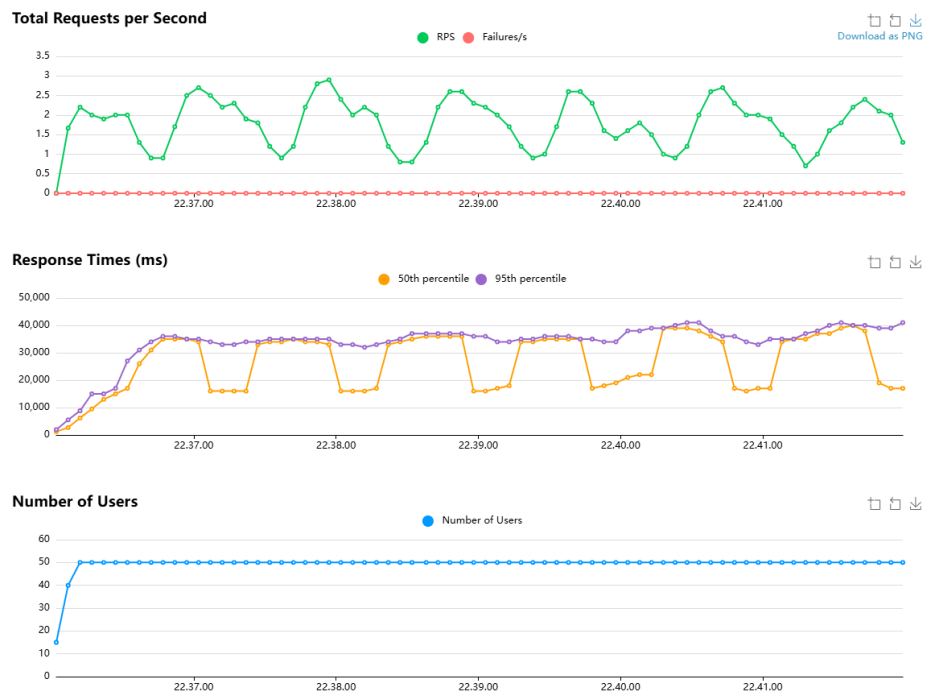
3. Pengujian Administrator (100 Pengguna Virtual)

Pengujian dilakukan dengan meningkatkan jumlah pengguna virtual menjadi 100 administrator yang mengakses sistem secara bersamaan.

Tabel 4.7 Hasil Pengujian Administrator (100 User)

Parameter	Hasil	Keterangan
Jumlah Pengguna Virtual	100 User	-
Total Request	279	Menurun 55,7% dari skenario 50 user (630 → 279)
Total Failure	0	-
Success Rate	100%	Semua <i>request</i> berhasil di-proses
Failure Rate	0%	Tidak ada kegagalan

Rata-rata <i>Response Time</i>	94.424,53 ms	TIDAK LAYAK (sangat buruk, > 90 detik)
--------------------------------	--------------	--



Gambar 4.16 Hasil *Load Testing* Administrator (100 User) pada Locust

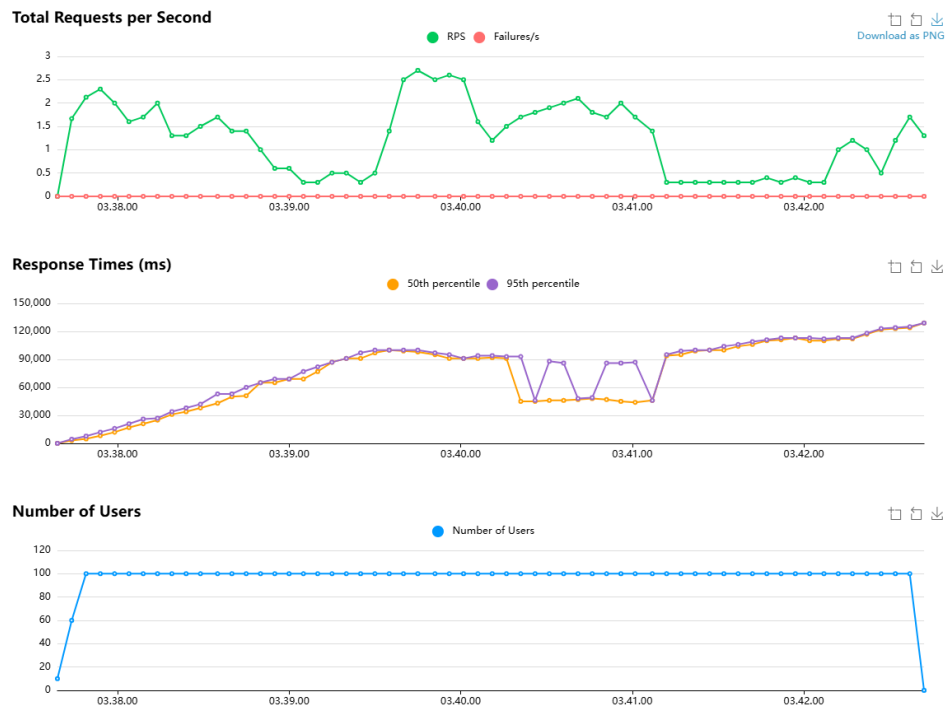
4. Pengujian Peserta Absensi (100 Pengguna Virtual)

Pengujian dilakukan dengan mensimulasikan 100 peserta yang mengakses formulir absensi dan mengirimkan data absensi secara bersamaan.

Tabel 4.8 Hasil Pengujian Peserta (100 User)

Parameter	Hasil	Keterangan
Jumlah Pengguna Virtual	100 User	-
Total Request	365	Menurun 44,2% dari skenario 50 user (654 → 365)
Total Failure	0	-
Success Rate	100%	Semua <i>request</i> berhasil diproses

Failure Rate	0%	Tidak ada kegagalan
Rata-rata <i>Response</i> Time	62.541,68 ms	TIDAK LAYAK (sangat buruk, > 60 detik, jauh di atas batas toleransi 2-5 detik)



Gambar 4.17 Hasil *Load Testing* Peserta (100 User) pada Locust

C. Analisis Hasil Pengujian

1. Waktu Respons

Waktu respons rata-rata pada seluruh skenario (21-94 detik) jauh di atas batas toleransi pengguna (2-5 detik). Hal ini mengindikasikan adanya *bottleneck* kinerja yang signifikan pada sistem.

2. Anomali Penurunan Total Request

Ketika jumlah pengguna dinaikkan dari 50 ke 100, total *request* justru menurun drastis:

Tabel 4.9 Anomali penurunan request

Skenario	50 User	100 User	Penurunan
Administrator	630	279	-55,7%
Peserta	654	365	-44,2%

Penurunan ini mengindikasikan bahwa *server* tidak mampu menangani 100 pengguna virtual secara bersamaan, kemungkinan disebabkan oleh *timeout*, *thread pool* penuh, atau kehabisan sumber daya CPU/RAM.

3. Faktor Penyebab

Berdasarkan spesifikasi perangkat, faktor utama penyebab buruknya kinerja adalah:

- Prosesor Celeron 2 Core — harus menjalankan Apache, MySQL, Laravel, dan Locust secara bersamaan
- RAM 8 GB — terbatas untuk menjalankan seluruh komponen secara simultan
- Lingkungan Localhost — terjadi kompetisi resource antara *server* dan alat uji

Dengan demikian, hasil pengujian ini lebih mencerminkan keterbatasan lingkungan pengembangan daripada kapasitas maksimal sistem.

4.2.3 Kesimpulan Pengujian

Pengujian fungsional terhadap **52 skenario uji** menunjukkan tingkat keberhasilan **100%**. Seluruh fitur sistem telah berfungsi sesuai dengan rancangan dan kebutuhan.

Namun, dari sisi kinerja, sistem masih memiliki kelemahan signifikan. Meskipun seluruh *request* berhasil diproses tanpa kegagalan (*failure rate* 0%), waktu respons rata-rata yang dihasilkan sangat buruk, yaitu **21–**

94 detik—jauh di atas batas toleransi pengguna yang hanya 2–5 detik. Selain itu, terjadi anomali penurunan total *request* hingga **44–56%** saat jumlah pengguna dinaikkan dari 50 ke 100, yang menandakan *server* tidak mampu menangani beban tinggi.

Hasil ini dipengaruhi oleh keterbatasan lingkungan pengujian (*localhost* dengan prosesor Celeron 2 Core dan RAM 8 GB). Dengan demikian, sistem **belum layak digunakan pada skala nyata** dan memerlukan optimasi kinerja sebelum diimplementasikan di lingkungan produksi.

BAB V KESIMPULAN

5.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian yang telah dilakukan, dapat disimpulkan sebagai berikut:

1. Sistem Berhasil Dikembangkan Secara Fungsional

Sistem Absensi Berbasis *QR code* untuk Kegiatan Reses DPRD Kota Batam berhasil dibangun dengan seluruh fitur yang direncanakan, meliputi pengelolaan acara, generate *QR code*, absensi peserta, rekapitulasi data, dan pembuatan laporan. Sistem mampu memfasilitasi proses absensi peserta secara digital melalui pemindaian *QR code* serta mendukung pengelolaan data kegiatan reses oleh administrator.

2. Validasi Anti-Duplikasi Berjalan dengan Baik

Sistem yang dibangun telah menerapkan mekanisme validasi untuk mencegah terjadinya absensi ganda pada acara yang sama berdasarkan data peserta yang tersimpan dalam sistem. Selain itu, sistem juga mampu mencatat data kehadiran secara otomatis dan menghasilkan rekapitulasi serta laporan kehadiran yang dapat digunakan untuk mendukung kegiatan administrasi reses.

3. Kinerja Sistem Masih Memerlukan Optimasi

Berdasarkan hasil pengujian fungsional menggunakan metode *Black-Box Testing*, seluruh 52 skenario uji berhasil dijalankan dengan tingkat keberhasilan 100%. Namun dari sisi kinerja, pengujian menggunakan Locust menunjukkan bahwa meskipun seluruh *request* berhasil diproses tanpa kegagalan (*failure rate* 0%), waktu respons rata-rata yang dihasilkan sangat buruk, yaitu 21–94 detik—jauh di atas batas toleransi pengguna (2-5 detik). Selain itu, terjadi anomali penurunan total *request* hingga 44–56% saat jumlah pengguna dinaikkan dari 50 ke 100, yang menandakan *server* tidak mampu menangani beban tinggi. Hasil ini dipengaruhi oleh keterbatasan lingkungan pengujian (*localhost*), sehingga sistem belum layak digunakan pada skala nyata dan memerlukan optimasi kinerja sebelum diimplementasikan di lingkungan produksi.

5.2 Saran

Berdasarkan keterbatasan dan hasil pengujian yang telah dilakukan, terdapat beberapa saran untuk pengembangan dan implementasi ke depan:

1. **Optimasi Kinerja Sistem**

Melakukan optimasi pada sisi aplikasi (*profiling query, caching*), basis data (penambahan indeks), dan *server* (spesifikasi perangkat yang memadai) untuk meningkatkan waktu respons sistem.

2. **Pengujian Ulang pada Lingkungan Produksi**

Setelah optimasi dilakukan, pengujian kinerja perlu diulang pada lingkungan yang mendekati kondisi produksi untuk memperoleh gambaran performa yang akurat.

3. **Pengembangan Fitur Tambahan**

Menambahkan validasi identitas peserta yang lebih kuat (NIK, OTP via WhatsApp), notifikasi otomatis, serta integrasi dengan sistem informasi DPRD.

4. **Dukungan Implementasi**

Menyediakan infrastruktur jaringan yang memadai dan pelatihan singkat bagi administrator agar seluruh fitur sistem dapat dimanfaatkan secara optimal.

DAFTAR PUSTAKA

- Alfikri, A., Gusman, A. P., & Azrialdi, A. (2024). E-Absensi menggunakan QR-Code berbasis web di Kantor Desa Binamang. *Jurnal Ilmu Teknologi dan Informatika*, 4(1), 45–56.
- Djamarullah, A. R., Nuryasin, I., & Wibowo, H. (2024). Designing a *QR code Attendance* system using BYOD. *Ultimatics: Jurnal Teknik Informatika*, 16(1), 32–37.
- Eliza Natalia, Sitorus, E. N., Jamaluddin, & Harianja, E. J. G. (2023). Sistem informasi kehadiran siswa menggunakan *QR code* berbasis Android (Studi kasus SD Negeri 105270). *TAMIKA: Jurnal Tugas Akhir Manajemen Informatika & Komputerisasi Akuntansi*, 5(2), 78–85.
- Hamdani, M., Wibowo, A., & Heryono, H. (2021). Pemanfaatan *QR code* dalam sistem absensi digital untuk pencatatan kehadiran. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 8(2), 345–352.
- Herlina, L., & Hidayatulloh, M. K. (2016). *QR code: Teknologi dan penerapannya*. Deepublish.
- Khaerudin, M., Sumantri, A., Supriatna, E., & Ritzkal. (2022). Development of employee *Attendance* information system using Quick Response Code in Bekasi City Education Office. *Jurnal Inovatif: Inovasi Teknologi Informasi dan Informatika*, 10(3), 112–125.
- Rafai, R., Solikhun, & Safii, M. (2024). Perancangan absensi *QR code* mahasiswa berbasis *website* pada STIKOM Tunas Bangsa Pematang Siantar menggunakan metode Agile. *Jurnal Manajemen Informatika Jayakarta*, 12(1), 67–78.
- Wei, K. L. J., Manori, M., Devnath, M. K., Pasi, G. F., & Kumar, S. (2017). Web-based application development: Trends and challenges. *International Journal of Advanced Computer Science and Applications*, 8(4), 156–165.

- Dwiyanto, A. (2020). **Mewujudkan Tata Pemerintahan yang Baik: Reformasi Birokrasi dan Governance di Indonesia**. Yogyakarta: Gadjah Mada University Press.
- Kompas. (2022, October 17). Litbang "Kompas" Ungkap Mayoritas Publik Anggap DPR Tidak Transparan Kelola Anggaran. **Kompas.com**. Retrieved from <https://nasional.kompas.com/read/2022/10/17/12460061/litbang-kompas-ungkap-mayoritas-publik-anggap-dpr-tidak-transparan-kelola>.
- Laravel LLC. (2024). *Laravel Documentation*. <https://laravel.com/docs>

DAFTAR LAMPIRAN

Lampiran A. Dokumen Proses Pengumpulan Requirement

https://drive.google.com/drive/folders/1QIsByVatTt-GmeLs2sKn7WTBWugUZ46Z5?usp=drive_link

Lampiran B. Link Produk

<https://github.com/farissaeflirayanti88/absensi-qr-code.git>

Lampiran C. Hasil Pengujian

https://drive.google.com/drive/folders/1ICH39mHMMK5Q0ihqv-R9P4pYtXgFGRWD?usp=drive_link