

BAB IV

HASIL DAN PEMBAHASAN

4.1 System and Software Design (Perancangan Sistem dan Perangkat Lunak)

Berdasarkan analisis kebutuhan yang telah dijabarkan sebelumnya, tahapan selanjutnya adalah memodelkan arsitektur perangkat lunak sebelum memasuki fase pemrograman.

Pemodelan sistem ini dirancang menggunakan standar Unified Modeling Language (UML) untuk memetakan alur kerja di BKPSDM Kabupaten Grobogan secara visual.

1. Deskripsi Sistem

Sistem Informasi Manajemen Agenda ini merupakan aplikasi berbasis web yang dikembangkan untuk mendukung pengelolaan jadwal kegiatan kedinasan di lingkungan BKPSDM Kabupaten Grobogan. Sistem ini menggantikan proses pencatatan manual dan koordinasi lisan antar-bidang dengan alur kerja digital yang terpusat, sehingga seluruh kegiatan, mulai dari pendaftaran jadwal baru, perubahan, pembatalan, hingga penyusunan laporan rekapitulasi, dapat ditelusuri dan dipertanggungjawabkan.

Sistem ini menangani empat area fungsional utama, yaitu: (1) pengelolaan jadwal kegiatan beserta deteksi bentrok waktu secara otomatis; (2) pengelolaan data pendukung berupa akun pengguna dan ruangan; (3) pelaporan dan rekapitulasi kegiatan yang dapat difilter dan diekspor; serta (4) layanan informasi publik bagi pegawai untuk memantau jadwal dan mengunduh dokumen undangan tanpa perlu melakukan autentikasi.

a. Ruang Lingkup Sistem

Sistem mencakup 10 fungsionalitas utama (use case) yang dikelompokkan menjadi dua kategori aktor sebagai berikut.

- 1) Admin (8 use case): Login, Tambah Jadwal Kegiatan, Ubah Jadwal Kegiatan, Batal Jadwal Kegiatan, Kelola Pengguna Sistem, Kelola Data Ruangan, Rekapitulasi & Filter Laporan, serta Cetak & Ekspor Laporan.
- 2) Pegawai (2 use case): Lihat Jadwal Publik dan Unduh Dokumen Undangan.

2. Pengguna Sistem (Aktor)

Sistem ini melibatkan dua jenis pengguna dengan hak akses yang berbeda secara tegas: satu aktor dengan akses penuh melalui autentikasi, dan satu aktor dengan akses terbatas tanpa autentikasi.

a. Admin

Admin adalah pemegang kendali penuh atas operasional sistem. Setelah berhasil login, Admin berwenang mengelola seluruh siklus hidup jadwal kegiatan (menambah, mengubah, membatalkan), mengatur data ruangan dan akun pengguna lain, serta menyusun dan mengekspor laporan rekapitulasi untuk keperluan pengambilan keputusan maupun pertanggungjawaban unit kerja.

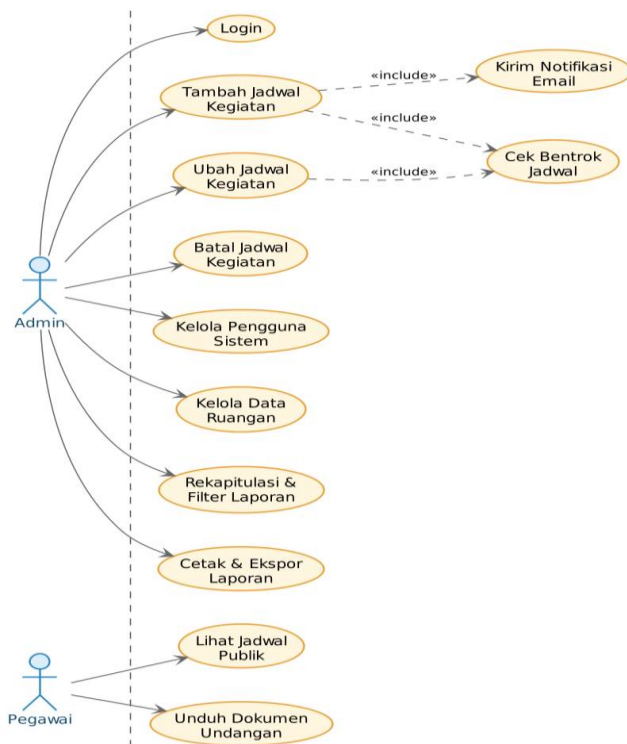
Pegawai

Pegawai adalah pengguna akhir yang diberi akses secara publik tanpa perlu memiliki akun maupun melakukan login. Melalui akses ini, pegawai dapat memantau daftar kegiatan kedinasan yang sedang berjalan secara real-time serta mengunduh berkas undangan yang terlampir pada suatu jadwal, sehingga distribusi informasi berjalan lebih transparan dan tidak bergantung pada komunikasi manual dengan Admin.

3. Use Case Diagram

Use Case Diagram berikut menggambarkan interaksi antara kedua aktor dengan 10 fungsionalitas yang disediakan sistem. Seluruh aktor ditempatkan di luar batas sistem, sedangkan seluruh use

case digambarkan di dalam kotak batas sistem.



Gambar 4. 1 Use Case Diagram

Diagram ini juga memuat dua relasi include pada use case Tambah Jadwal Kegiatan, yaitu terhadap Cek Bentrok Jadwal dan Kirim Notifikasi Email, serta satu relasi include pada use case Ubah Jadwal Kegiatan terhadap Cek Bentrok Jadwal. Relasi include menandakan bahwa perilaku tersebut senantiasa dijalankan sebagai bagian dari alur utama use case induknya, bukan sekadar opsi tambahan.

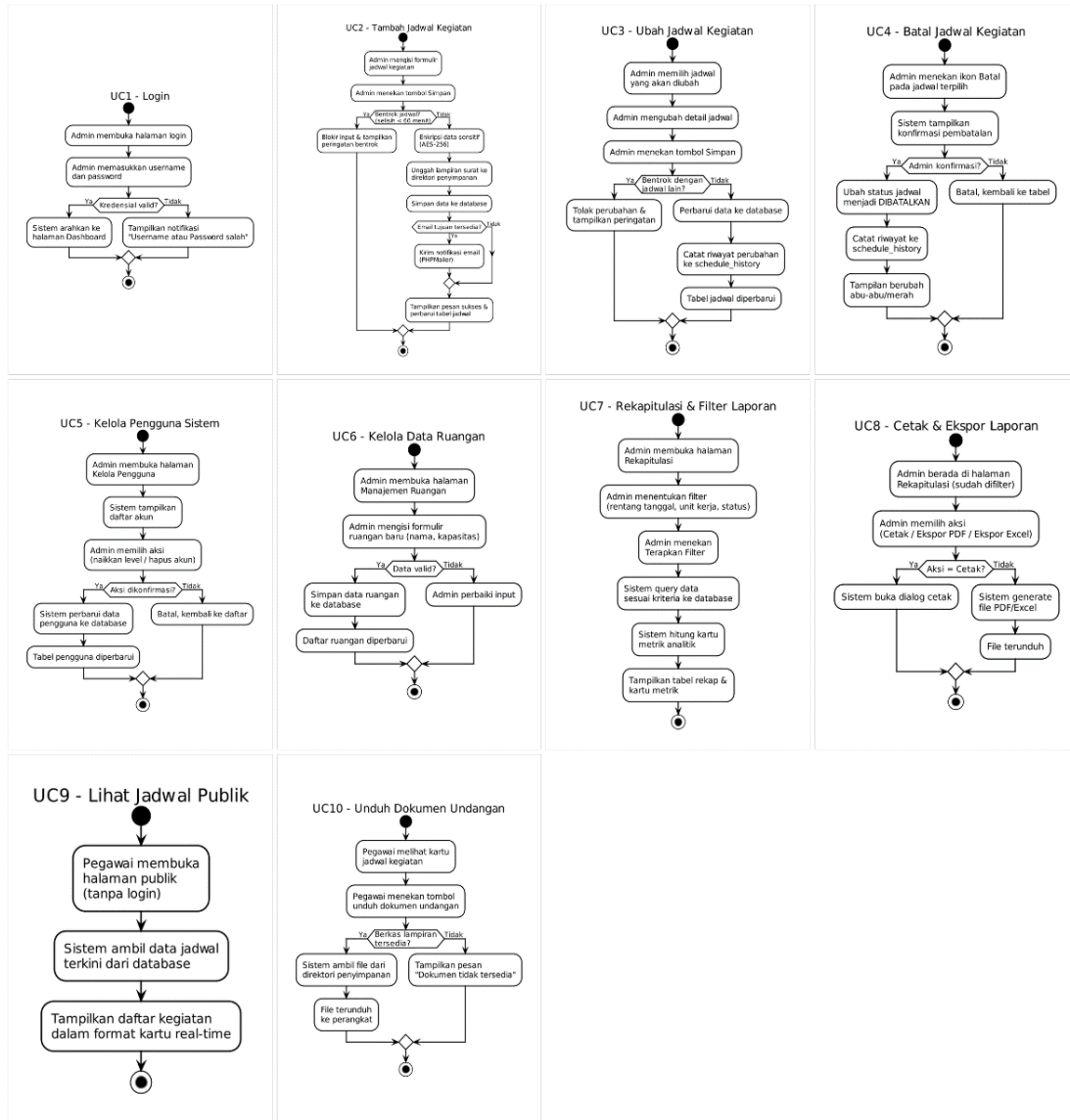
4. Penjelasan Setiap Use Case

Tabel 4.1 Tabel Penjelasan Use Case

Kode	Nama Use Case	Aktor	Deskripsi Singkat
UC1	Login	Admin	Admin memasukkan username dan password untuk masuk ke sistem; kredensial diverifikasi sebelum sesi Dashboard dibuka.

UC2	Tambah Jadwal Kegiatan	Admin	Admin menginput data kegiatan baru. Sistem menjalankan include Cek Bentrok Jadwal sebelum data disimpan, dan include Kirim Notifikasi Email setelah data tersimpan.
UC3	Ubah Jadwal Kegiatan	Admin	Admin mengubah rincian jadwal yang sudah ada. Sistem kembali menjalankan include Cek Bentrok Jadwal (dengan mengecualikan jadwal yang sedang diubah) sebelum pembaruan disimpan.
UC4	Batal Jadwal Kegiatan	Admin	Admin membatalkan sebuah kegiatan melalui mekanisme konfirmasi, sehingga status jadwal berubah menjadi DIBATALKAN tanpa menghapus datanya secara permanen.
UC5	Kelola Pengguna Sistem	Admin	Admin menambah, mengubah level akses, atau menonaktifkan akun pengguna lain di dalam sistem.
UC6	Kelola Data Ruangan	Admin	Admin mengelola data master ruangan (nama dan kapasitas) yang nantinya dipilih sebagai lokasi pada saat penjadwalan kegiatan.
UC7	Rekapitulasi & Filter Laporan	Admin	Admin menyaring data kegiatan berdasarkan rentang tanggal, unit kerja, atau status, kemudian sistem menghitung metrik rekap secara otomatis.
UC8	Cetak & Ekspor Laporan	Admin	Admin mencetak atau mengekspor hasil rekapitulasi yang sudah difilter ke dalam berkas PDF atau Excel.
UC9	Lihat Jadwal Publik	Pegawai	Pegawai mengakses halaman publik tanpa login untuk memantau daftar kegiatan kedinasan yang sedang berjalan secara real-time.
UC10	Unduh Dokumen Undangan	Pegawai	Pegawai mengunduh berkas lampiran undangan yang melekat pada sebuah jadwal kegiatan, apabila berkas tersebut tersedia.

5. Activity Diagram



Gambar 4.2 Overview seluruh 10 Activity Diagram

a. Penjelasan Alur Tiap Activity Diagram

UC1 — Login

Admin membuka halaman login → memasukkan username dan password → sistem memvalidasi kredensial → jika valid, admin diarahkan ke halaman Dashboard; jika tidak, sistem menampilkan notifikasi "Username atau Password salah".

UC2 — Tambah Jadwal Kegiatan

Admin mengisi formulir jadwal kegiatan dan menekan tombol Simpan → sistem mengecek potensi bentrok (selisih waktu < 60 menit) → jika bentrok, input diblokir dan peringatan ditampilkan; jika tidak, data sensitif dienkripsi (AES-256), lampiran surat diunggah ke direktori penyimpanan, lalu data disimpan ke database → sistem mengecek ketersediaan email tujuan → jika tersedia, notifikasi dikirim via PHPMailer → pesan sukses ditampilkan dan tabel jadwal diperbarui.

UC3 — Ubah Jadwal Kegiatan

Admin memilih jadwal yang akan diubah dan mengganti detailnya, lalu menekan Simpan → sistem mengecek ulang potensi bentrok dengan jadwal lain → jika bentrok, perubahan ditolak dan peringatan ditampilkan; jika tidak, data diperbarui ke database dan riwayat perubahan dicatat ke `schedule_history` → tabel jadwal diperbarui.

UC4 — Batal Jadwal Kegiatan

Admin menekan ikon Batal pada jadwal terpilih → sistem menampilkan konfirmasi pembatalan → jika dikonfirmasi, status jadwal diubah menjadi DIBATALKAN dan riwayatnya dicatat ke `schedule_history`, tampilan berubah abu-abu/merah; jika tidak, admin kembali ke tabel tanpa perubahan.

UC5 — Kelola Pengguna Sistem

Admin membuka halaman Kelola Pengguna → sistem menampilkan daftar akun → admin memilih aksi (menaikkan level akses atau menghapus akun) → jika dikonfirmasi, sistem

memperbarui data pengguna ke database dan tabel pengguna diperbarui; jika tidak, aksi dibatalkan.

UC6 — Kelola Data Ruangan

Admin membuka halaman Manajemen Ruangan dan mengisi formulir ruangan baru (nama, kapasitas) → sistem memvalidasi data → jika valid, data disimpan ke database dan daftar ruangan diperbarui; jika tidak, admin memperbaiki input.

UC7 — Rekapitulasi & Filter Laporan

Admin membuka halaman Rekapitulasi dan menentukan filter (rentang tanggal, unit kerja, status) → menekan Terapkan Filter → sistem melakukan query ke database sesuai kriteria dan menghitung kartu metrik analitik secara internal → tabel rekap dan kartu metrik ditampilkan ke Admin.

UC8 — Cetak & Ekspor Laporan

Admin, dari halaman Rekapitulasi yang sudah difilter, memilih aksi Cetak atau Ekspor PDF/Excel → jika Cetak, sistem membuka dialog cetak; jika Ekspor, sistem men-generate berkas PDF/Excel dan file otomatis terunduh.

UC9 — Lihat Jadwal Publik

Pegawai membuka halaman publik tanpa perlu login → sistem mengambil data jadwal terkini dari database secara real-time → daftar kegiatan ditampilkan dalam format kartu interaktif.

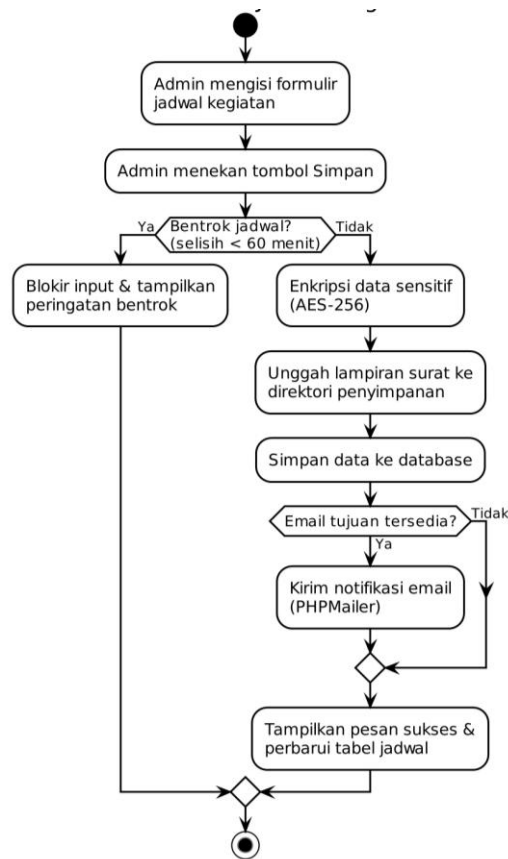
UC10 — Unduh Dokumen Undangan

Pegawai melihat kartu jadwal kegiatan dan menekan tombol unduh dokumen undangan → sistem mengecek ketersediaan berkas lampiran → jika tersedia, file diambil dari direktori

penyimpanan dan terunduh ke perangkat Pegawai; jika tidak, sistem menampilkan pesan "Dokumen tidak tersedia".

b. Contoh Detail: Activity Diagram UC2 (Tambah Jadwal Kegiatan)

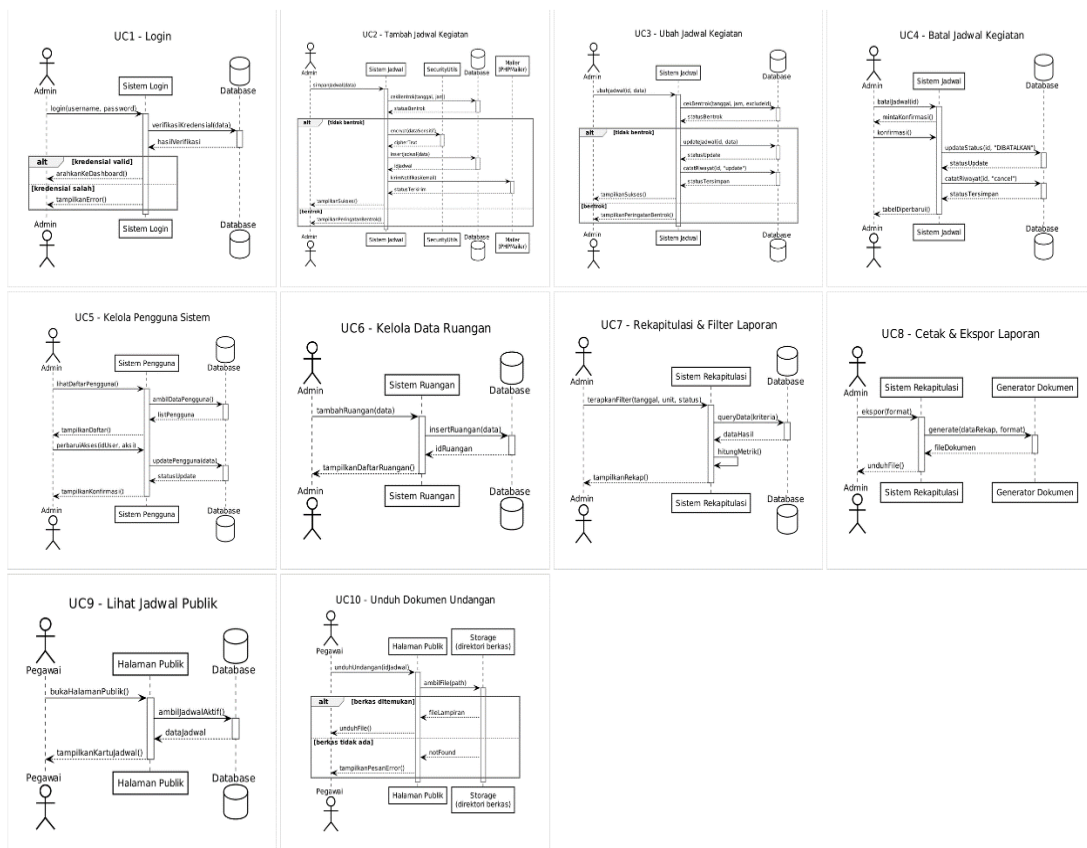
Sebagai ilustrasi visual, berikut Activity Diagram lengkap untuk UC2, yang menampilkan dua titik decision sekaligus, yaitu pemeriksaan bentrok jadwal dan pemeriksaan ketersediaan alamat email tujuan sebelum notifikasi dikirim.



Gambar 4.3 Activity Diagram UC2: Tambah Jadwal Kegiatan

6. Sequence Diagram

Sequence Diagram menggambarkan interaksi antar objek dari waktu ke waktu, siapa memanggil siapa dan pesan apa yang dikirim. Jika Activity Diagram menjelaskan langkah-langkah proses bisnis, Sequence Diagram melengkapinya dengan menjelaskan komunikasi teknis di balik setiap use case.



Gambar 4.4 Overview seluruh 10 Sequence Diagram

a. Penjelasan Alur Tiap Sequence Diagram

UC1 — Login

Partisipan: Admin, Sistem Login, Database

Admin mengirim username dan password ke Sistem Login → sistem meneruskan permintaan verifikasi kredensial ke Database → Database mengembalikan hasil verifikasi → jika valid,

sistem mengarahkan Admin ke Dashboard; jika tidak, sistem mengembalikan pesan error ke Admin.

UC2 — Tambah Jadwal Kegiatan

Partisipan: Admin, Sistem Jadwal, SecurityUtils, Database, Mailer (PHPMailer)

Diagram paling kompleks (5 partisipan). Admin mengirim data jadwal ke Sistem Jadwal → sistem meminta pengecekan bentrok ke Database → jika tidak bentrok, sistem meminta SecurityUtils mengenkripsi data sensitif → data terenkripsi disimpan ke Database → sistem mengirim permintaan notifikasi ke Mailer (PHPMailer) → status terkirim dikembalikan → konfirmasi sukses ditampilkan ke Admin; jika bentrok, sistem langsung mengembalikan peringatan tanpa melibatkan SecurityUtils maupun Mailer.

UC3 — Ubah Jadwal Kegiatan

Partisipan: Admin, Sistem Jadwal, Database

Admin mengirim data perubahan jadwal ke Sistem Jadwal → sistem meminta pengecekan bentrok (mengecualikan jadwal itu sendiri) ke Database → jika tidak bentrok, sistem mengirim perintah update ke Database, dilanjutkan pencatatan riwayat perubahan → konfirmasi sukses dikembalikan ke Admin; jika bentrok, sistem mengembalikan peringatan.

UC4 — Batal Jadwal Kegiatan

Partisipan: Admin, Sistem Jadwal, Database

Admin mengirim permintaan pembatalan ke Sistem Jadwal → sistem meminta konfirmasi ke Admin → setelah dikonfirmasi, sistem mengirim perintah update status (DIBATALKAN) ke Database, dilanjutkan pencatatan riwayat → konfirmasi tabel diperbarui dikembalikan ke Admin.

UC5 — Kelola Pengguna Sistem

Partisipan: Admin, Sistem Pengguna, Database

Admin meminta daftar pengguna ke Sistem Pengguna → sistem mengambil data ke Database → daftar dikembalikan dan ditampilkan → Admin mengirim perintah perbarui akses → sistem meneruskan update ke Database → konfirmasi dikembalikan ke Admin.

UC6 — Kelola Data Ruangan

Partisipan: Admin, Sistem Ruangan, Database

Admin mengirim data ruangan baru ke Sistem Ruangan → sistem menyimpan data ke Database → ID ruangan dikembalikan → daftar ruangan terbaru ditampilkan ke Admin.

UC7 — Rekapitulasi & Filter Laporan

Partisipan: Admin, Sistem Rekapitulasi, Database

Admin mengirim kriteria filter ke Sistem Rekapitulasi → sistem meminta data ke Database sesuai kriteria → data hasil dikembalikan → sistem menghitung metrik secara internal (self-message) → tabel rekap dan kartu metrik ditampilkan ke Admin.

UC8 — Cetak & Ekspor Laporan

Partisipan: Admin, Sistem Rekapitulasi, Generator Dokumen

Admin mengirim permintaan ekspor (format) ke Sistem Rekapitulasi → sistem meneruskan data rekap ke Generator Dokumen → berkas dikembalikan → sistem memicu unduhan file ke Admin.

UC9 — Lihat Jadwal Publik

Partisipan: Pegawai, Halaman Publik, Database

Pegawai membuka Halaman Publik → sistem meminta data jadwal aktif ke Database → data dikembalikan → sistem menampilkan kartu jadwal real-time ke Pegawai.

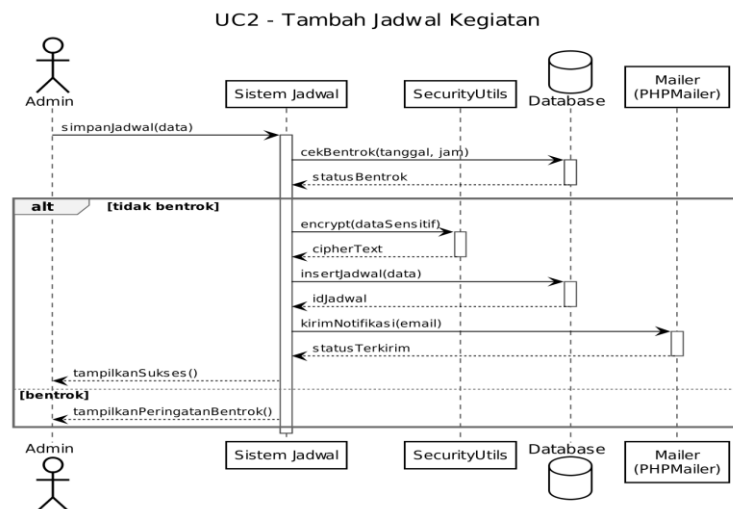
UC10 — Unduh Dokumen Undangan

Partisipan: Pegawai, Halaman Publik, Storage (direktori berkas)

Pegawai mengirim permintaan unduh undangan ke Halaman Publik → sistem meminta berkas ke Storage → jika berkas ditemukan, file lampiran dikembalikan dan diunduh ke perangkat Pegawai; jika tidak, sistem mengembalikan status "tidak ditemukan" dan menampilkan pesan error.

b. Contoh Detail: Sequence Diagram UC2 (Tambah Jadwal Kegiatan)

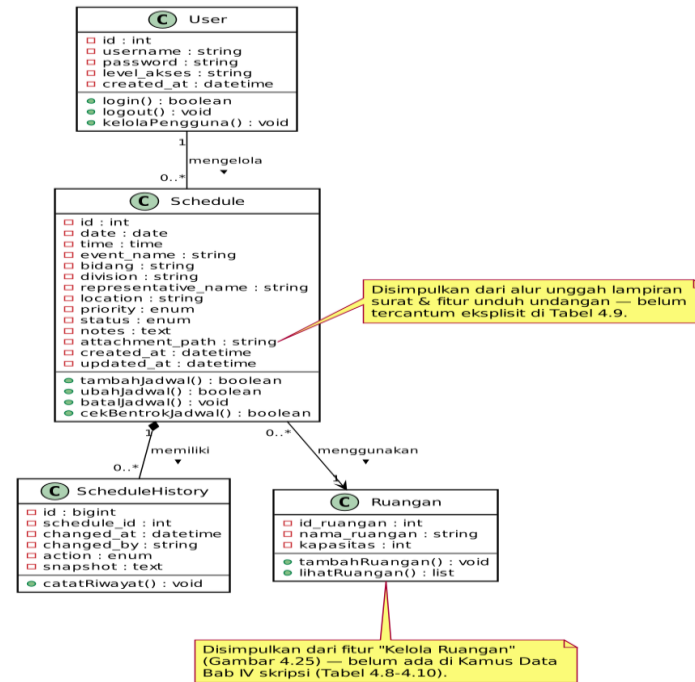
Diagram berikut menunjukkan notasi standar UML: aktor (stick figure), objek (kotak persegi), lifeline (garis putus-putus vertikal), activation bar, pesan sinkron, serta blok alt untuk memisahkan dua kemungkinan hasil pemeriksaan bentrok jadwal.



Gambar 4.5 Sequence Diagram UC2: Tambah Jadwal Kegiatan

7. Class Diagram

Class Diagram menggambarkan struktur data statis sistem, yaitu kelas-kelas yang menjadi cetak biru rancangan basis data. Sistem ini dimodelkan ke dalam empat kelas yang saling berelasi.



Gambar 4. 6 Class Diagram

a. Penjelasan Tiap Kelas

User — menyimpan kredensial akses (username, password, level_akses) serta menyediakan operasi login(), logout(), dan kelolaPegguna() untuk mengatur akun lain.

Schedule — kelas inti yang menampung seluruh rincian kegiatan: tanggal, waktu, nama kegiatan, bidang/divisi, lokasi, prioritas, status, catatan, hingga lampiran berkas. Kelas ini membawa operasi tambahJadwal(), ubahJadwal(), batalJadwal(), dan cekBentrokJadwal().

ScheduleHistory — mencatat jejak audit setiap perubahan pada sebuah Schedule (waktu perubahan, pelaku perubahan, jenis aksi, dan cuplikan datanya) melalui operasi catatRiwayat().

Ruangan — menyimpan data master ruangan (nama dan kapasitas) yang digunakan sebagai lokasi kegiatan, dengan operasi tambahRuangan() dan lihatRuangan().

b. Relasi Antar Kelas

User 1 — 0..* Schedule (“mengelola”): satu akun Admin dapat mengelola banyak jadwal kegiatan.

Schedule 1 — 0. ScheduleHistory (“memiliki”, composition): setiap baris riwayat hanya bermakna sebagai bagian dari satu jadwal induknya.

Schedule 0..* — 1 Ruangan (“menggunakan”): banyak jadwal kegiatan dapat memakai satu ruangan yang sama pada waktu yang berbeda.

8. Rancangan Tabel dan Relasi Antar Tabel

Rancangan tabel berikut diturunkan dari Class Diagram pada Bab 4 dan diselaraskan dengan Kamus Data yang sudah dituangkan pada BAB IV skripsi (Tabel 4.8–4.10, mengikuti Entity Relationship Diagram Gambar 4.18 dan Logical Record Structure Gambar 4.19). Tiga tabel pertama (users, schedules, schedule_history) merupakan struktur yang sudah baku di skripsi; satu tabel tambahan (ruangan) disusun sebagai turunan dari Class Diagram karena entitas ini belum tercantum eksplisit di Kamus Data BAB IV, sehingga masih memerlukan konfirmasi sebelum ditetapkan sebagai bagian resmi rancangan basis data.

a. Detail Struktur Tabel

Tabel 4.2 Struktur Tabel Users

Nama Kolom	Tipe Data	Panjang	Keterangan
id	INT	11	Primary Key, Auto Increment
username	VARCHAR	50	Nama pengguna untuk akses login
password	VARCHAR	255	Kata sandi pengguna (terenkripsi/hash)
created_at	TIMESTAMP	-	Waktu dan tanggal akun dibuat

Tabel 4.3 Struktur Tabel Schedules

Nama Kolom	Tipe Data	Panjang	Keterangan
id	INT	11	Primary Key, Auto Increment
date	DATE	-	Tanggal pelaksanaan kegiatan
time	TIME	-	Jam pelaksanaan
event_name	VARCHAR	150	Nama/judul kegiatan
bidang	VARCHAR	100	Unit kerja penyelenggara (opsional)
division	VARCHAR	100	Divisi yang ditugaskan
representative_name	VARCHAR	150	Nama delegasi/perwakilan
location	VARCHAR	150	Lokasi kegiatan (teks bebas)
priority	ENUM	-	Urgensi acara: low, normal, urgent
status	ENUM	-	Status acara: scheduled, completed, canceled
notes	TEXT	-	Catatan tambahan
created_at	TIMESTAMP	-	Waktu rekaman dibuat
updated_at	TIMESTAMP	-	Waktu rekaman terakhir diubah

Tabel 4.4 Struktur Tabel schedule_history

Nama Kolom	Tipe Data	Panjang	Keterangan
id	BIGINT	20	Primary Key, Auto Increment
schedule_id	INT	11	Foreign Key → schedules
changed_at	TIMESTAMP	-	Waktu riwayat dicatat
changed_by	VARCHAR	100	Nama admin pelaku perubahan
action	ENUM	-	Jenis aksi: create, update, delete
snapshot	TEXT	-	Cuplikan data (JSON) jadwal terkait

Tabel 4.5 Struktur Tabel ruangan

Nama Kolom	Tipe Data	Panjang	Keterangan
id_ruangan	INT	11	Primary Key, Auto Increment
nama_ruangan	VARCHAR	100	Nama ruangan
kapasitas	INT	11	Kapasitas tampung ruangan

c. Ringkasan Relasi Antar Tabel

Tabel Induk (1)	Tabel Anak (N)	Melalui Kolom FK	Kardinalitas
users	schedules	(belum ada kolom FK eksplisit di Tabel 4.9)	1 : N — konseptual
schedules	schedule_history	schedule_id	1 : N
ruangan	schedules	id_ruangan (usulan)	1 : N — usulan

d. Catatan Rancangan

- 1) Relasi users → schedules dijelaskan secara naratif pada LRS BAB IV (satu Admin mengelola banyak jadwal), namun Tabel 4.9 (schedules) belum memuat kolom Foreign Key eksplisit (misalnya user_id atau created_by) yang mengikat baris jadwal ke akun pembuatnya. Perlu dikonfirmasi apakah kolom ini memang sudah ada pada implementasi basis data aktual namun belum tercantum di Kamus Data, atau perlu ditambahkan agar Class Diagram (relasi “mengelola”) konsisten dengan struktur fisik tabel.
- 2) Kolom location pada tabel schedules saat ini bertipe VARCHAR (teks bebas), sedangkan Class Diagram memodelkan Ruangan sebagai entitas tersendiri dengan relasi “menggunakan”. Jika use case Kelola Data Ruangan sudah diimplementasikan sebagai tabel

terpisah, disarankan menambahkan kolom Foreign Key `id_ruangan` pada `schedules` dan memperbarui Kamus Data BAB IV; jika belum, kelas `Ruangan` pada Class Diagram sebaiknya diberi catatan sebagai rancangan pengembangan lanjutan, bukan struktur yang sudah berjalan.

- 3) Nilai ENUM pada kolom `status` (`scheduled`, `completed`, `canceled`) sudah selaras dengan alur UC4 — Batal Jadwal Kegiatan, sehingga tidak diperlukan penambahan nilai baru untuk mendukung use case tersebut.
- 4) Kolom `attachment_path` yang muncul pada Class Diagram (lampiran surat/dokumen undangan) juga belum tercantum eksplisit pada Tabel 4.9; jika fitur unggah/unduh lampiran (UC2 dan UC10) sudah berjalan pada aplikasi, kolom ini perlu ditambahkan ke Kamus Data agar dokumentasi tetap sinkron dengan implementasi.

4.2 Implementation and Unit Testing (Implementasi dan Pengujian Unit)

Tahap implementasi adalah proses menerjemahkan desain perangkat lunak ke dalam bahasa pemrograman.

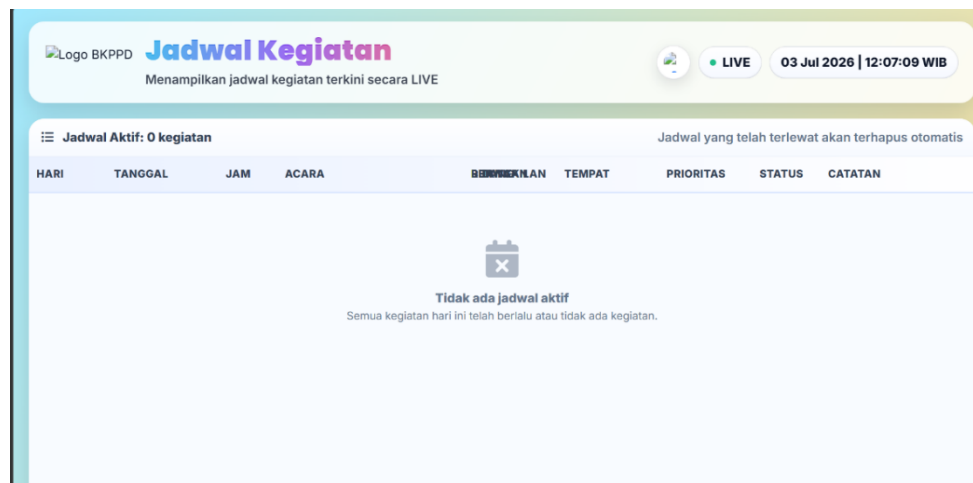
1. Lingkungan Implementasi

Sistem ini dibangun menggunakan arsitektur `ClientServer`. Perangkat lunak yang digunakan dalam membangun sistem ini meliputi bahasa pemrograman PHP (`Hypertext Preprocessor`) sebagai bahasa `server-side`, `HTML` dan `CSS` untuk merancang antarmuka visual pengguna, `JavaScript` untuk interaktivitas di sisi klien (`AJAX`), dan `MySQL` sebagai sistem manajemen basis data (`DBMS`). Proses pengembangan dan pengujian lokal menggunakan web server `Laragon`.

2. Implementasi Antarmuka Pengguna (User Interface)

Tahap implementasi antarmuka merupakan wujud nyata dari perancangan sistem, di mana logika di balik layar diterjemahkan ke dalam bentuk visual yang dapat berinteraksi langsung dengan pengguna. Berikut adalah hasil penjabaran antarmuka pada Sistem Informasi Manajemen Agenda di lingkungan BKPSDM:

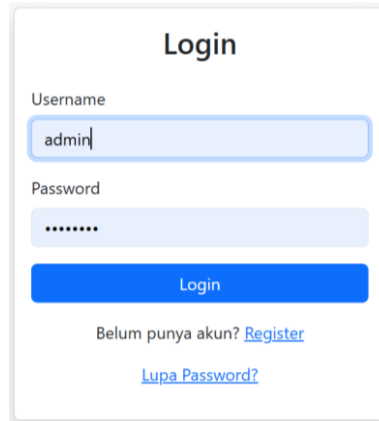
a. Halaman Publik (Tampilan Utama Jadwal)



Gambar 4. 7 Tampilan Halaman Publik (Mode Display) Antarmuka Web

Antarmuka halaman publik dirancang sebagai layar informasi utama yang akan langsung menyambut pengunjung saat aplikasi diakses. Layar ini menampilkan daftar kegiatan secara real-time tanpa mengharuskan pegawai melakukan proses login terlebih dahulu. Informasi jadwal disusun ke dalam format kartu interaktif yang merangkum berbagai data krusial, mulai dari penunjuk jam digital, nama acara, bidang penanggung jawab, hingga lokasi kegiatan. Di samping itu, tersemat pula tombol pintasan untuk melihat dan mengunduh dokumen undangan, sebuah fitur yang secara signifikan memangkas birokrasi penyampaian informasi antarpegawai.

b. Halaman Login Administrator

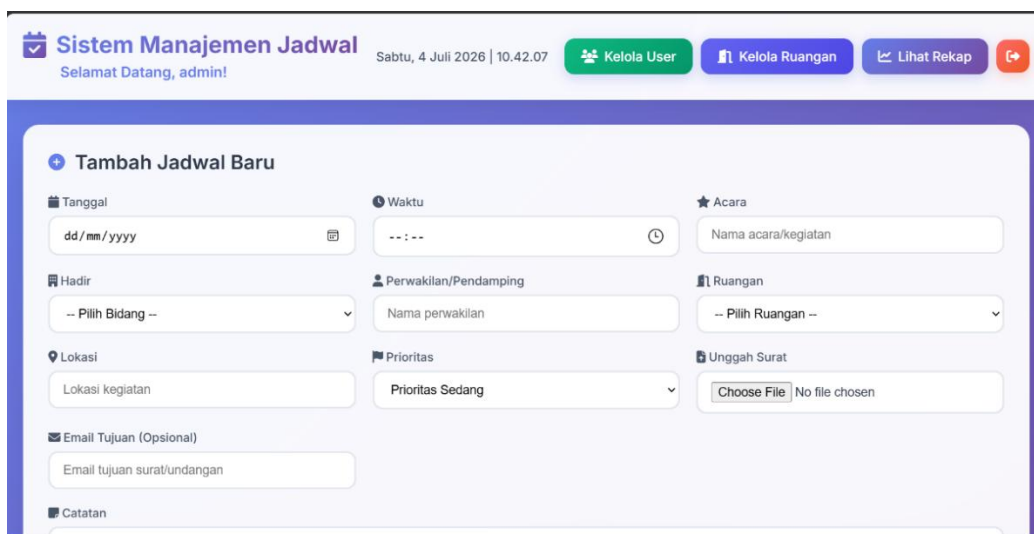


The login form is titled "Login" and is enclosed in a light gray border. It features two input fields: "Username" with the text "admin" and "Password" with masked characters "*****". Below these fields is a blue "Login" button. At the bottom, there are two links: "Belum punya akun? [Register](#)" and "[Lupa Password?](#)".

Gambar 4. 8 Tampilan login

Antarmuka login dirancang sebagai lapisan pengamanan pertama untuk melindungi sistem dari intervensi pihak yang tidak sah. Halaman ini difokuskan pada desain visual yang bersih dan minimalis, menyajikan dua bidang isian utama, yakni username dan kata sandi. Kehadiran halaman ini merupakan mekanisme krusial yang memastikan bahwa hanya pengguna dengan otorisasi administratif yang sah yang diizinkan untuk mengakses panel kendali jadwal, sehingga kerahasiaan dan integritas data instansi tetap terjaga dengan baik.

c. Halaman Dashboard dan Formulir Tambah Jadwal

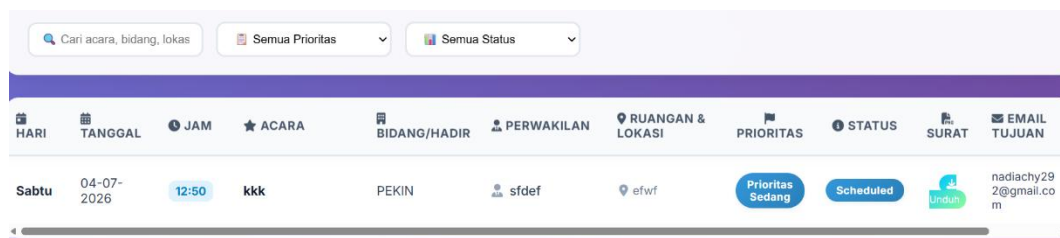


The dashboard header includes the title "Sistem Manajemen Jadwal", a welcome message "Selamat Datang, admin!", the date and time "Sabtu, 4 Juli 2026 | 10.42.07", and three buttons: "Kelola User", "Kelola Ruangan", and "Lihat Rekap". The main content area is titled "Tambah Jadwal Baru" and contains several form fields: "Tanggal" (dd/mm/yyyy), "Waktu" (--:--), "Acara" (Nama acara/kegiatan), "Hadir" (Pilih Bidang --), "Perwakilan/Pendamping" (Nama perwakilan), "Ruangan" (Pilih Ruangan --), "Lokasi" (Lokasi kegiatan), "Prioritas" (Prioritas Sedang), "Unggah Surat" (Choose File, No file chosen), "Email Tujuan (Opsional)" (Email tujuan surat/undangan), and "Catatan".

Gambar 4.9 Antarmuka Dashboard dan Formulir Input Data

Setelah berhasil melewati proses autentikasi, sistem akan mengarahkan pengguna ke halaman dashboard yang bertindak sebagai pusat komando utama. Pada bagian atas layar, tersedia panel navigasi cepat untuk berpindah ke fitur kelola pengguna, ruangan, dan rekapitulasi. Tepat di bawahnya, sistem membentangkan antarmuka formulir interaktif yang dialokasikan untuk merekam agenda baru. Tata letak formulir ini disusun secara sistematis agar pengelola dapat dengan efisien menginputkan detail waktu, lokasi, nama delegasi, hingga mengunggah dokumen surat pendukung.

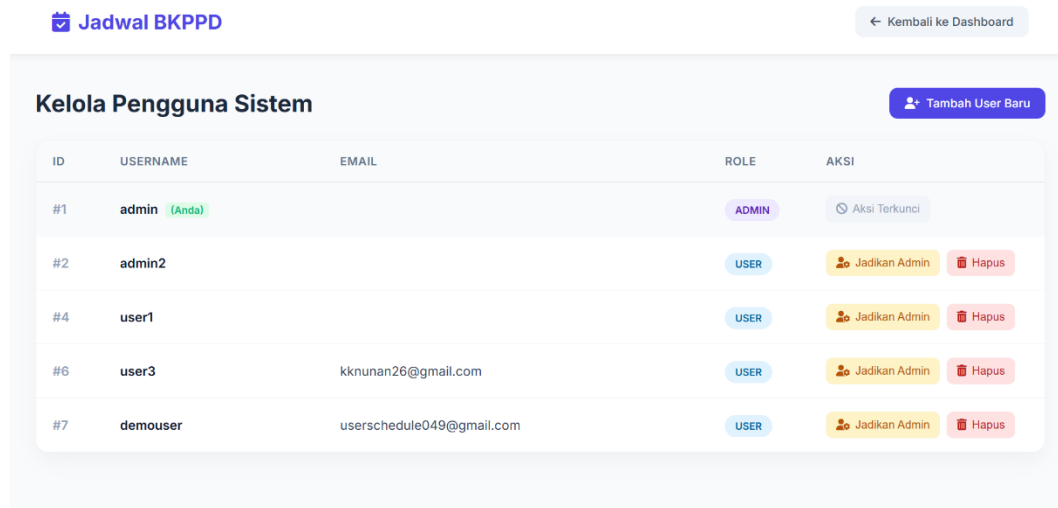
d. Antarmuka Daftar Jadwal dan Fitur Pencarian



Gambar 4.10 Antarmuka Tabel Pemantauan Jadwal

Guna mempermudah pemantauan agenda yang telah diinputkan, sistem menyajikan antarmuka tabel daftar jadwal yang sangat informatif. Tabel ini merangkum elemen-elemen data secara terstruktur, memfasilitasi pembacaan ringkas terkait status pelaksanaan hingga tingkat urgensi acara. Guna meningkatkan fleksibilitas navigasi, antarmuka ini turut dibekali dengan fitur bilah pencarian cerdas dan dropdown filter. Komponen ini sangat membantu administrator dalam melakukan pelacakan spesifik di tengah tumpukan data jadwal kegiatan yang padat.

e. Halaman Kelola Pengguna Sistem



Gambar 4.11 Antarmuka Manajemen Pengguna Sistem

Dalam rangka memelihara tata kelola hak akses yang sehat, aplikasi mengimplementasikan halaman khusus untuk manajemen pengguna. Melalui antarmuka ini, administrator utama (Super Admin) dapat mengawasi seluruh akun yang memiliki ikatan dengan sistem. Halaman ini menyajikan matriks data pengguna yang didampingi oleh tombol aksi dinamis, yang memberikan keleluasaan bagi pengelola untuk menaikkan level akses seseorang menjadi Admin, ataupun menghapus akun yang dinilai sudah tidak relevan dengan kebutuhan operasional.

f. Halaman Manajemen Ruangan

 **Kelola Ruangan**

[← Kembali ke Dashboard](#)

+ Tambah Ruangan Baru

Nama Ruangan / Tempat *

Contoh: Ruang Rapat A

Kapasitas (Orang)

Contoh: 50

Keterangan Tambahan

Fasilitas atau catatan khusus...

 **Simpan Ruangan**

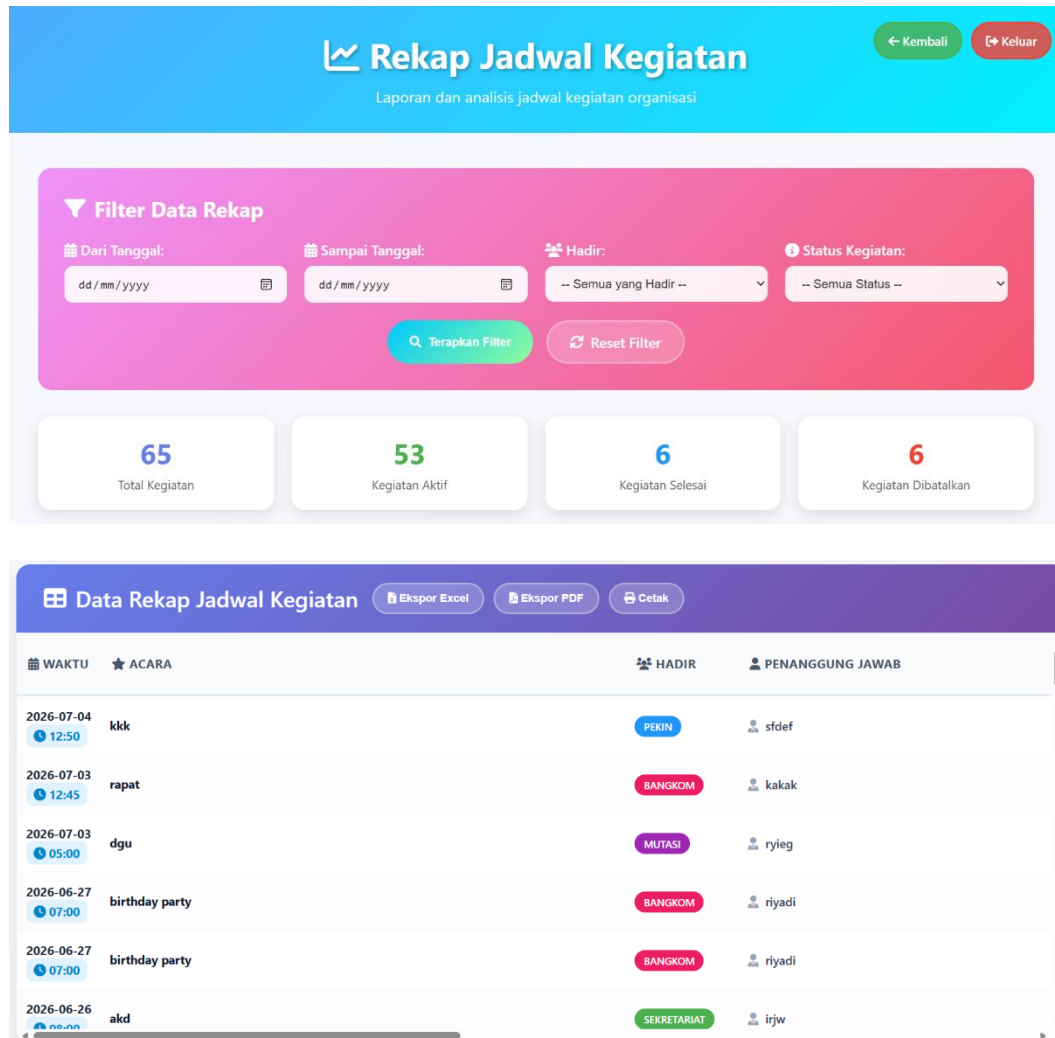
 **Daftar Ruangan**

No	Nama Ruangan	Kapasitas	Keterangan	Aksi
1	rapat	20 Orang	ac	

Gambar 4.12 Anatarmuka Manajemen Ruangan

Guna menunjang keakuratan fungsi deteksi bentrok fasilitas, sistem menyediakan antarmuka mandiri untuk mengelola basis data ruangan. Halaman ini dipecah menjadi dua wilayah operasional: area formulir untuk mendaftarkan fasilitas ruang baru beserta detail kapasitasnya, dan area tabel pemantauan untuk melihat seluruh lokasi yang tersedia. Terintegrasinya antarmuka ini memastikan bahwa setiap kali Admin membuat jadwal baru, pilihan lokasi yang tersedia selalu merujuk pada basis master ruangan yang sah dan mutakhir.

g. Halaman Rekapitulasi dan Pelaporan Jadwal



Gambar 4.13 Antarmuka Halaman Rekapitulasi dan Analisis Jadwal

Guna mendukung kebutuhan rekam arsip dan evaluasi manajerial instansi, sistem dibekali dengan halaman antarmuka khusus rekapitulasi. Halaman ini dirancang secara komprehensif dengan membagi antarmuka menjadi dua wilayah fungsional. Pada area atas, tersedia panel Filter Data Rekap dinamis yang memberikan keleluasaan bagi Admin untuk mengisolasi data berdasarkan rentang waktu tertentu (date range), unit kerja yang hadir, hingga memilah status kegiatannya. Sebagai nilai tambah, area ini juga menyajikan deretan kartu metrik analitik yang secara real-time menghitung kalkulasi total kegiatan, baik yang sedang aktif, telah selesai, maupun

yang dibatalkan.

Sementara itu, pada area bawah layar, seluruh data hasil penyaringan dihimpun ke dalam matriks tabel rekap yang rapi. Fungsionalitas antarmuka ini semakin disempurnakan dengan hadirnya fitur pelaporan terintegrasi. Tersedia deretan tombol aksi cepat yang memungkinkan pengelola untuk langsung mencetak (print) tabel tersebut atau mengekspor datanya ke dalam wujud dokumen baku (berformat Excel maupun PDF), sehingga penyusunan laporan kegiatan untuk pimpinan dapat dieksekusi dengan jauh lebih cepat dan efisien.

3. Implementasi Fitur Utama (Berdasarkan Kodingan Aktual)

Selain implementasi visual, sistem ini dilengkapi dengan fungsifungsi backend tingkat lanjut untuk menunjang operasional BKPSDM, antara lain:

- a. Pencegahan Bentrok Jadwal (Conflict Checking) Untuk mencegah terjadinya tumpang tindih (overlapping) kegiatan di ruangan atau waktu yang sama, sistem menggunakan teknologi AJAX yang berjalan di latar belakang saat form diisi. Sistem mengevaluasi selisih waktu (\$diff_minutes) dari jadwal yang ada di basis data dengan jadwal yang sedang diinput. Jika kurang dari 60 menit, sistem menolak inputan tersebut.
- b. Notifikasi Email Otomatis (Integrasi SMTP) Sistem memanfaatkan pustaka PHPMailer untuk mengirimkan notifikasi email undangan atau pemberitahuan acara secara otomatis ketika suatu agenda baru berhasil disimpan ke basis data. Fitur ini mempercepat alur informasi yang sebelumnya hanya mengandalkan grup pesan singkat.
- c. Enkripsi Keamanan (Data Security) Menyadari pentingnya privasi data pemerintahan, sistem menerapkan Advanced Encryption Standard (AES256) menggunakan algoritma kriptografi PHP (openssl_encrypt). Informasi vital seperti Nama Acara (event_name), Nama Perwakilan (representative_name), dan Lokasi (location) disimpan dalam bentuk teks acak (chiphertext) di

dalam basis data.

4. Pengujian Unit (Unit Testing)

Sejalan dengan tahapan model pengembangan Waterfall, setelah modul-modul program selesai ditulis, langkah krusial selanjutnya adalah melakukan Unit Testing (Pengujian Unit). Pengujian ini merupakan langkah validasi mandiri pada tingkat kode (white-box) yang dieksekusi oleh pengembang untuk memastikan setiap blok fungsi atau script dapat beroperasi sesuai logika matematika tanpa memicu error, jauh sebelum dirangkai menjadi sistem antarmuka yang utuh.

Dalam pengembangan Sistem Informasi Agenda ini, pengujian unit secara spesifik diterapkan pada komponen-komponen fungsional krusial di sisi backend. Sebagai contoh, pengujian dieksekusi pada berkas keamanan data (SecurityUtils.php). Pengembang menguji fungsi openssl_encrypt dengan cara menyuntikkan dummy data (data palsu) untuk memastikan algoritma kriptografi AES-256 mampu mengacak dan mengembalikan wujud teks secara akurat. Selain itu, pengujian unit juga difokuskan pada script logika pencegahan bentrok jadwal. Algoritma kalkulasi waktu (\$diff_minutes) pada PHP diuji coba menggunakan berbagai variabel jam fiktif secara lokal. Tujuannya adalah untuk memvalidasi apakah perhitungan matematika sistem sudah benar dalam mendeteksi dan memblokir selisih jadwal di bawah batas 60 menit. Setelah seluruh unit program penyusun ini diuji dan dinyatakan stabil tanpa cacat logika, barulah komponen-komponen tersebut diintegrasikan ke dalam antarmuka sistem secara menyeluruh.

4.3 Integration and System Testing (Integrasi dan Pengujian Sistem)

Setelah seluruh rancangan modul sistem diimplementasikan ke dalam bahasa pemrograman, tahapan selanjutnya adalah melakukan pengujian sistem secara menyeluruh. Proses ini bertujuan untuk memastikan seluruh fungsi dapat berjalan dan terintegrasi dengan baik tanpa adanya

kesalahan program (error/bug). Dalam penelitian ini, pengujian sistem dilakukan menggunakan dua metode pendekatan, yakni pengujian fungsionalitas melalui Black Box Testing dan evaluasi tingkat penerimaan pengguna melalui uji User Acceptance Test (UAT) dengan instrumen pengukuran Skala Likert.

1. Pengujian Integrasi (Integration Testing)

Sebelum pengujian sistem menyeluruh dilakukan, sistem telah melalui tahap pengujian integrasi untuk memastikan aliran data antar modul terhubung dengan baik. Berikut adalah hasil pengujian integrasinya:

- a. Integrasi Modul Frontend dan Backend: Memastikan form input jadwal berhasil mengirimkan data ke database MySQL. (Status: Berhasil)
- b. Integrasi Modul Aplikasi dengan API PHPMailer: Memastikan ketika jadwal baru disimpan, trigger email dari PHPMailer bekerja dan masuk ke inbox tujuan. (Status: Berhasil)

2. Pengujian Black Box

Pengujian Black Box dilakukan untuk mengevaluasi fungsionalitas sistem pada aspek masukan (input) dan keluaran (output) tanpa meninjau alur kode program yang ada di baliknya. Pengujian ini dilakukan langsung oleh pengembang/peneliti dengan cara memasukkan berbagai skenario data untuk memastikan apakah fitur utama aplikasi berjalan sesuai dengan spesifikasi kebutuhan yang telah dirancang. Pengujian berfokus pada fungsi autentikasi (login), pengelolaan data (CRUD), fungsionalitas pencarian/filter, hingga fitur ekspor dokumen.

Sebelum pengujian sistem menyeluruh (Black Box) dilakukan, sistem telah melalui tahap pengujian integrasi (Integration Testing) untuk memastikan aliran data antar modul seperti modul input antarmuka pengguna (frontend) terhubung dengan baik ke basis data MySQL, serta integrasi

modul jadwal dengan layanan PHPMailer berjalan secara sinkron.

Tabel 4. 6 Hasil Pengujian Black Box Testing.

No	Nama Fitur	Proses	Output yang Diharapkan	Status Uji
1	Login (kredensial benar)	Admin memasukkan username dan password yang benar, lalu menekan tombol Login.	Sistem berhasil mengautentikasi dan mengarahkan Admin ke halaman Dashboard Jadwal.	Valid
2	Login (kredensial salah)	Admin memasukkan username atau password yang salah/kosong.	Sistem menolak akses dan menampilkan notifikasi “Username atau Password salah”.	Valid
3	Tambah Jadwal Kegiatan	Admin mengisi seluruh form jadwal secara lengkap lalu menekan tombol “Simpan Jadwal”.	Sistem menyimpan data ke basis data dan jadwal baru muncul otomatis di tabel jadwal.	Valid
4	Ubah Jadwal Kegiatan	Admin mengubah jam dan ruang acara pada jadwal yang sudah ada, lalu menekan “Simpan”.	Perubahan tersimpan dan tabel memperbarui detail jadwal yang bersangkutan.	Valid
5	Batal Jadwal Kegiatan	Admin menekan ikon Batal (coret) pada salah satu jadwal.	Sistem mengubah status jadwal menjadi “DIBATALKAN” dan tampil berwarna abu-abu/merah.	Valid
6	Filter Rekap	Admin memasukkan rentang “Tanggal Awal” s.d. “Tanggal Akhir” lalu menekan Terapkan Filter.	Tabel Rekap hanya menampilkan kegiatan pada rentang tanggal tersebut.	Valid
7	Ekspor PDF/Excel	Admin menekan tombol “Ekspor PDF” atau “Ekspor Excel” di halaman Rekap Jadwal.	Sistem mengunduh dokumen dengan ekstensi sesuai yang berisi daftar jadwal.	Valid

a. Persentase Keberhasilan Pengujian

Untuk mengetahui tingkat keberhasilan pengujian sistem, dilakukan perhitungan persentase

keberhasilan dengan rumus berikut.

$$\text{Persentase Keberhasilan} = (\text{Jumlah uji coba berhasil} / \text{Jumlah seluruh uji coba}) \times 100\%$$

Berdasarkan pengujian yang telah dilakukan, seluruh 7 skenario pengujian pada halaman Admin dinyatakan valid, sehingga:

$$\text{Persentase Keberhasilan} = (7 / 7) \times 100\% = 100\%$$

Berdasarkan pengujian sistem menggunakan metode black box testing yang telah ditampilkan pada Tabel 1, seluruh fitur yang diuji berjalan sesuai fungsi yang dirancang tanpa ditemukan kendala, sehingga sistem dapat dinyatakan bebas dari kesalahan logika (bug) pada fungsi-fungsi yang diuji.

3. Evaluasi Pengguna (User Acceptance Test)

Evaluasi pengguna dilakukan untuk mengetahui apakah sistem yang dibangun sudah sesuai dengan kebutuhan penggunanya. Evaluasi ini dilakukan melalui kuesioner yang disebarakan kepada 7 (tujuh) responden yang terdiri dari 1 orang Admin dan 6 orang Pegawai di lingkungan BKPSDM Kabupaten Grobogan, menggunakan instrumen Skala Likert.

a. Kriteria Penilaian

Kriteria penilaian yang digunakan dalam evaluasi pengguna sistem ini disajikan pada tabel berikut.

Tabel 4. 7 Keterangan Bobot Skala Likert

Jawaban	Bobot Nilai
Sangat Setuju (SS)	5
Setuju (S)	4
Netral (N)	3
Tidak Setuju (TS)	2
Sangat Tidak Setuju (STS)	1

b. Kriteria Penilaian Kelayakan

Kriteria penilaian kelayakan yang digunakan untuk menginterpretasikan hasil persentase skor disajikan pada tabel berikut.

Tabel 4.8 Kriteria Penilaian Kelayakan

Persentase	Kategori
81% – 100%	Sangat Layak
61% – 80%	Layak
41% – 60%	Cukup Layak
21% – 40%	Kurang Layak
0% – 20%	Sangat Tidak Layak

c. Rekap Kuesioner

Kuesioner memuat 4 indikator pernyataan, yaitu Desain, Hak Akses, Informasi, dan Efisiensi.

Rekapitulasi jawaban dari 7 responden (1 Admin dan 6 Pegawai) disajikan pada tabel berikut.

Tabel 4.9 Rekapitulasi Hasil Kuesioner (Skala Likert)

No	Nama Responden	P1 (Desain)	P2 (Hak Akses)	P3 (Informasi)	P4 (Efisiensi)	Total Skor
1	Ngatmin	5	5	5	5	20
2	Rheimanda Devin	5	5	5	5	20
3	Akhie Nadjan Atifa	5	5	5	5	20
4	Kuncoro	4	4	4	5	17
5	Muzammil	5	4	5	5	19
6	Netty Puspitowati	5	5	5	5	20
7	Safira Imas	5	5	5	5	20

	Total Skor per Indikator	34	33	34	35	136
--	--------------------------	----	----	----	----	-----

Ke-28 jawaban (7 responden $\times$ 4 pernyataan) tersebut apabila dikelompokkan berdasarkan kategori Skala Likert menghasilkan distribusi sebagai berikut.

Tabel 4.10 Distribusi Jawaban Berdasarkan Kategori Skala Likert

Kategori Jawaban	Jumlah	Bobot	Skor
Sangat Setuju (SS)	24	5	120
Setuju (S)	4	4	16
Netral (N)	0	3	0
Tidak Setuju (TS)	0	2	0
Sangat Tidak Setuju (STS)	0	1	0
Jumlah	28	-	136

d. Perhitungan Persentase Kelayakan

$$1) \text{ Skor Hasil Survei} = (24 \times 5) + (4 \times 4) = 120 + 16 = 136$$

$$2) \text{ Skor Maksimal} = 5 \text{ (skor tertinggi)} \times 7 \text{ (jumlah responden)} \times 4 \text{ (jumlah pernyataan)} = 140$$

$$3) \text{ Persentase Skor} = (\text{Skor Hasil Survei} / \text{Skor Maksimal}) \times 100\% = (136 / 140) \times 100\% = 97,14\%$$

Dengan total skor 136, skor maksimum 140, serta persentase skor 97,14%, maka merujuk pada kriteria penilaian kelayakan pada Tabel 3, sistem ini termasuk dalam kategori “Sangat Layak” untuk diimplementasikan.

Penilaian kuantitatif tersebut juga sejalan dengan respons kualitatif responden pada kolom masukan kuesioner, yang menyebutkan bahwa perangkat lunak yang dibangun sudah bagus,

memberikan kemudahan bagi pengguna, lebih praktis, serta diharapkan dapat terus dikembangkan secara berkelanjutan.

4.4 Operation and Maintenance (Operasi dan Pemeliharaan)

1. Fase Operasi

Setelah melalui tahap implementasi dan pengujian, pengembangan sistem informasi ini memasuki tahap terakhir dalam siklus pengembangan, yaitu tahap operasi dan pemeliharaan. Pada tahap operasi, Sistem Informasi Jadwal Kegiatan telah diimplementasikan secara resmi dan digunakan pada lingkungan kerja Badan Kepegawaian, Pendidikan, dan Pelatihan Daerah (BKPPD). Pengguna sistem, yang terdiri atas staf dan admin tata usaha, memanfaatkan aplikasi berbasis web tersebut dalam kegiatan operasional harian, khususnya untuk mencatat serta memantau ketersediaan ruang rapat dan kelengkapan penyelenggaraan acara kedinasan. Berdasarkan hasil pemantauan selama periode penggunaan, sistem menunjukkan kinerja yang stabil dan responsif, baik pada saat diakses melalui jaringan internal maupun ketika ditampilkan pada layar monitor publik.

2. Fase Pemeliharaan

Hingga penyusunan laporan ini dilakukan, sistem masih dinyatakan relevan dengan fungsi kedinasan yang menjadi tujuan pengembangannya. Belum terdapat masukan dari pihak instansi maupun pengguna terkait kebutuhan revisi fungsional atau penambahan fitur baru. Selain itu, tidak ditemukan adanya laporan kegagalan teknis yang signifikan selama masa operasional (zero major bugs/downtime). Aktivitas pemeliharaan yang dilaksanakan pada tahap ini bersifat preventif, meliputi pencadangan basis data (backup) serta pengawasan

performa server secara berkala, dengan tujuan menjaga proses pencatatan jadwal kegiatan harian agar tetap berjalan secara optimal.