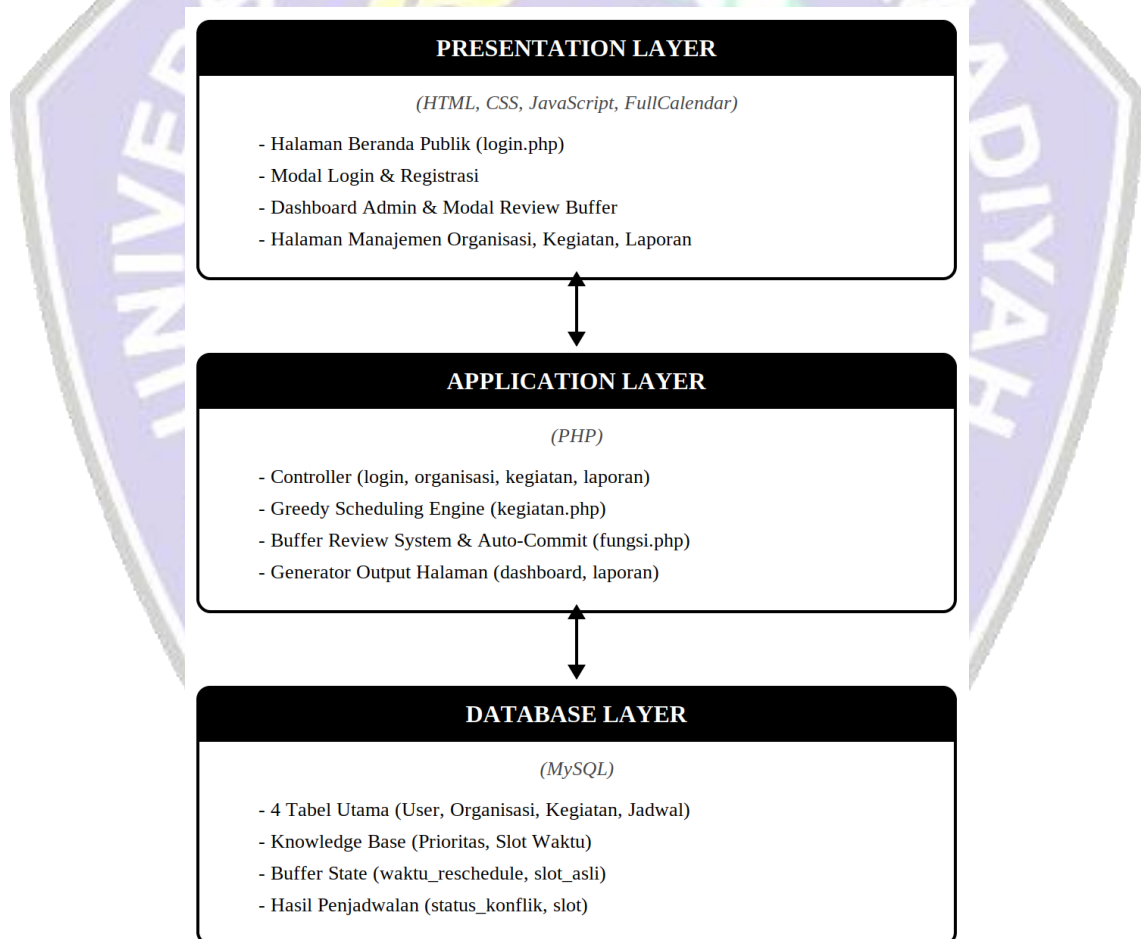


APLIKASI PENJADWALAN KEGIATAN ORGANISASI MENGUNAKAN ALGORITMA GREEDY

1. IMPLEMENTASI SISTEM

1.1 Arsitektur Sistem

Sistem ini menggunakan arsitektur *three-tier web* yang membagi sistem ke dalam tiga lapisan berbeda yaitu *presentation layer* (lapisan depan atau *frontend*), *application layer* (lapisan aplikasi atau *backend*), lalu *database layer* atau lapisan basis data. Keputusan ini bertujuan untuk mempermudah pemeliharaan program ketika ini melakukan pengembangan, pemeliharaan dan juga keamanan.[1]



Gambar 1.1 Diagram layer web three-tier

Setiap lapisan memiliki fungsinya masing-masing tetapi tetap terhubung satu sama lain melalui antarmuka yang sudah didefinisikan:

a. *Presentation Layer* (Lapisan Depan)

Lapisan depan atau *frontend* dikembangkan dengan tiga bahasa kunci web modern, HTML, CSS dan Javascript. Sedangkan untuk menampilkan kalender interaktifnya, kami menggunakan pustaka *FullCalendar* untuk mendapat fungsi penuh kalender di dalam program ini. Serta lapisan ini bertanggung jawab untuk tampilan dari setiap halaman di program ini.

b. *Application Layer* (Lapisan Belakang)

Lapisan ini biasa disebut *backend*. Dengan basis PHP, lapisan ini bertanggung jawab untuk mengurus logika dan algoritma di balik program ini. proses-proses seperti perhitungan algoritma *greedy*, penjadwalan otomatis, pengelolaan *buffer*, dan lain sejenisnya di balik layar.

c. *Database Layer* (Lapisan Basis Data)

Lapisan ini mengurus manajemen basis data yang sudah diinputkan oleh pengguna dari dalam program. Lapisan ini menyimpan semua data baik itu kegiatan, organisasi maupun penjadwalan yang sudah di tetapkan. Lapisan ini menggunakan basis data berbasis MySQL.

1.2 Struktur Database

Basis data yang digunakan terdiri dari 4 tabel utama yang saling berelasi. Keempatnya saling bekerja sama dalam menjalankan sistem dengan lapisan lain.

1. Tabel *tb_user*

Tabel ini menyimpan data pengguna atau *user*. Terdiri dari id, nama, email, password (menggunakan hash MD5) dan *role*.

2. Tabel *tb_organisasi*

Digunakan untuk menyimpan informasi organisasi mahasiswa yang memiliki kegiatan. Field utamanya adalah id, nama_organisasi, dan deskripsi.

3. Tabel *tb_kegiatan*

Tabel ini menyimpan kegiatan beserta dengan atribut-atributnya. *Field* yang tersedia yaitu id, nama_kegiatan, tanggal_mulai, tanggal_selesai, jam_mulai, jam_selesai, durasi_jam, prioritas, id_organisasi, dan status.

4. Tabel *tb_jadwal*

Tabel ini menyimpan hasil dari algoritma *greedy* dan juga menyimpan *buffer*. *Field* yang tersedia id, id_kegiatan, slot_mulai, slot_selesai, status_konflik, waktu_reschedule, slot_mulai_asli, dan slot_selesai_asli.[2]

1.3 Implementasi Algoritma *Greedy*

Program ini menggunakan algoritma *greedy* untuk menjadi otak sistem dalam menentukan penjadwalan secara otomatis setiap ada kegiatan baru yang ditambahkan. Secara umum algoritmya bekerja melalui beberapa tahapan:

1.3.1 Pengurutan Kegiatan Berdasarkan Prioritas

Pertama adalah mengurutkan data secara berurut dari yang besar berdasarkan prioritas-nya lalu tanggal mulai kemudian jam berakhir. Proses ini memastikan kegiatan yang ber-prioritas tinggi dapat dimasukkan lebih awal sbelum slot jadwal penuh.[3]

```
SELECT * FROM tb_kegiatan
ORDER BY prioritas ASC, tanggal_mulai ASC, jam_mulai ASC
```

1.3.2 Deteksi Konflik Slot Waktu

Setelah diurutkan, setiap kegiatan dicek untuk melihat apakah waktu penyelaksanaannya bertabrakan dengan jadwal lain. Pendeteksinya dengan melihat waktu awal dimulainya kegiatan, ketika sebuah kegiatan baru dimasukkan, jam awal yang diinputkan akan dicek ke jadwal yang sudah ada apakah tidak menabrak jam awal yang

sudah ada atau sebelum jam akhir jadwal yang ada, maka bentrokan tidak terjadi. Input maka langsung di terima.

```
if (selesai <= jadwal.mulai || mulai >= jadwal.selesai)
    // tidak bentrok, slot aman
else
    // konflik terdeteksi, perlu digeser
```

1.3.3 Penggeseran Otomatis ke Slot Kosong

Ketika bentrok terjadi, sistem akan mencari slot selanjutnya yang kosong untuk memenuhi kriteria input yang baru. Jika di hari yang sama sudah terpenuhi, maka akan digeser ke hari berikutnya pada jam yang sama yang diinputkan. Proses ini akan terus dicek sampai 336 kali atau kurang lebih 14 hari ke depan.

1.3.4 Sistem *Buffer Review* (Fitur Unggulan)

Kegiatan yang sudah diberikan slot baru setelah bentrok, akan memasuki masa review atau dianggap *buffer*. Kegiatan sudah terlihat di tempat yang dipilihkan sistem dari dashboard, tetapi masih terhitung masa tenggang atau *buffer* sebelum di review atau setelah 3 jam tanpa perubahan yang dilakukan pengguna.

Pengguna memiliki dua pilihan ketika membuka kegiatan yang masih dalam masa tenggang:

a. Diterima (*approve*)

Kegiatan langsung dikonfirmasi dan statusnya berubah menjadi aman dan sudah ditetapkan di jam yang sudah ditentukan otomatis sebelumnya.

b. Biarkan

Jika dalam 3 jam tidak ada tindakan, sistem secara otomatis mengubah status menjadi aman melalui mekanisme *auto-commit* yang berjalan setiap kali halaman dimuat.

Mekanisme *auto-commit* ini dijalankan lewat query SQL berikut:

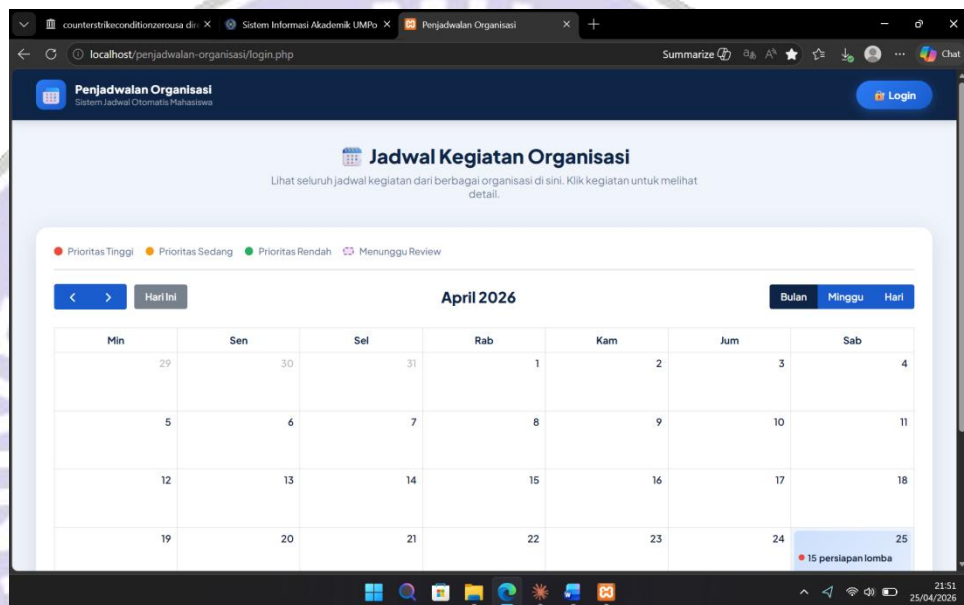
```
UPDATE tb_jadwal
SET status_konflik = 'aman'
```

```
WHERE status_konflik = 'reschedule'
AND TIMESTAMPDIFF(MINUTE, waktu_reschedule, NOW()) >= 180
```

Fitur ini dirancang agar proses penjadwalan tetap transparan. Pengguna dapat melihat dan memperbaiki input jika ada jadwal yang digeser secara tidak tepat, tapi sistem tetap bisa berjalan sendiri tanpa harus menunggu konfirmasi manual jika semuanya sudah sesuai.

2. IMPLEMENTASI INTERFACE

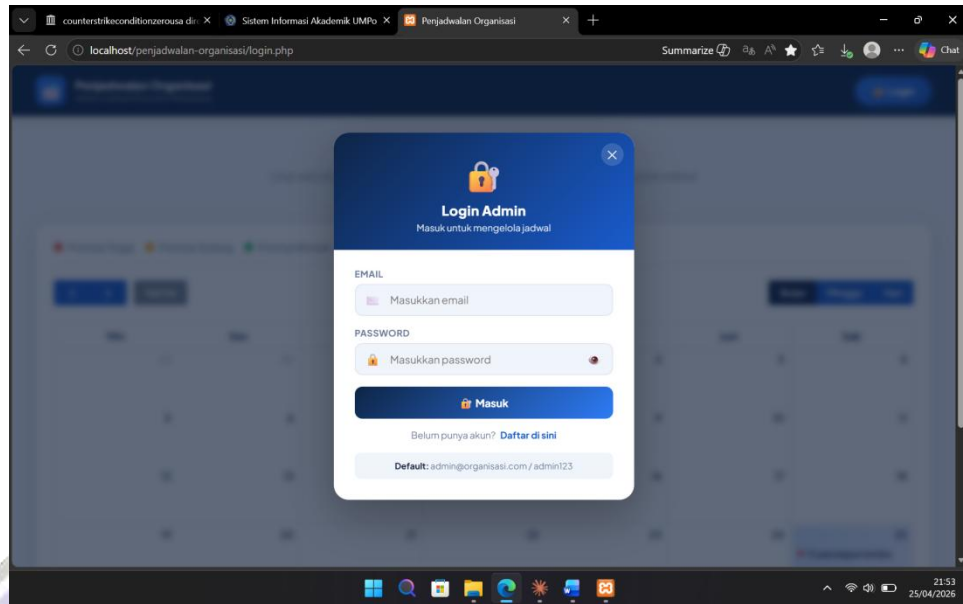
2.1 Halaman Beranda Publik (login.php)



Gambar 2.1 Tangkapan layar halaman beranda publik

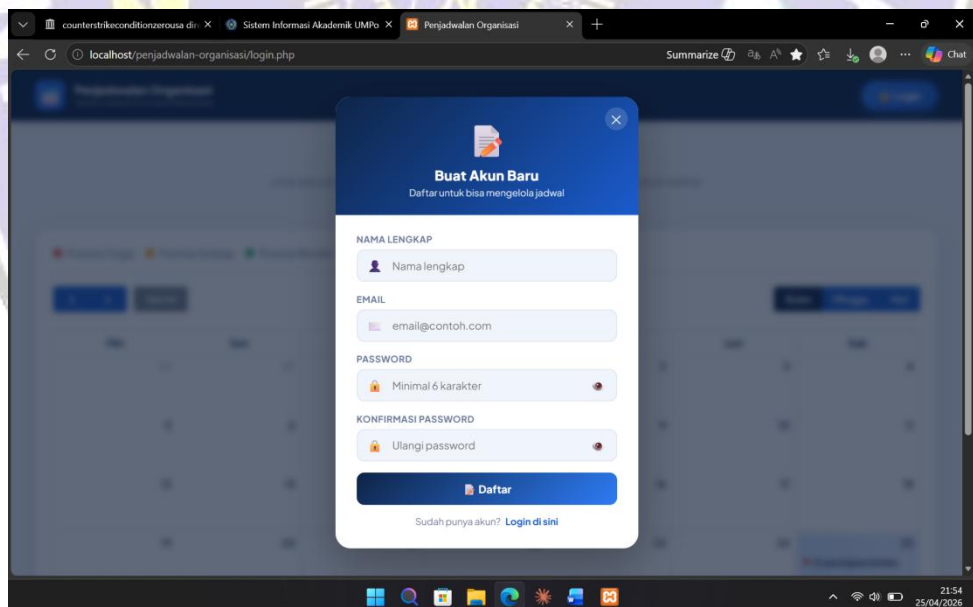
Pada Gambar 2.1, Tangkapan layar dari halaman beranda yang berisi kalender yang sudah ada jadwal yang ditetapkan juga tombol login serta logo aplikasi.

2.2 Modal Login dan Registrasi



Gambar 2.2 Tangkapan layar modal login

Pada Gambar 2.2, Tangkapan layar *popup* setelah menekan tombol login untuk masuk ke dalam dasbor, pengguna dapat memasukkan identitas akun yang diperlukan atau membuat akun baru dengan menekan tulisan kecil dibawah betuliskan “Daftar di sini”.

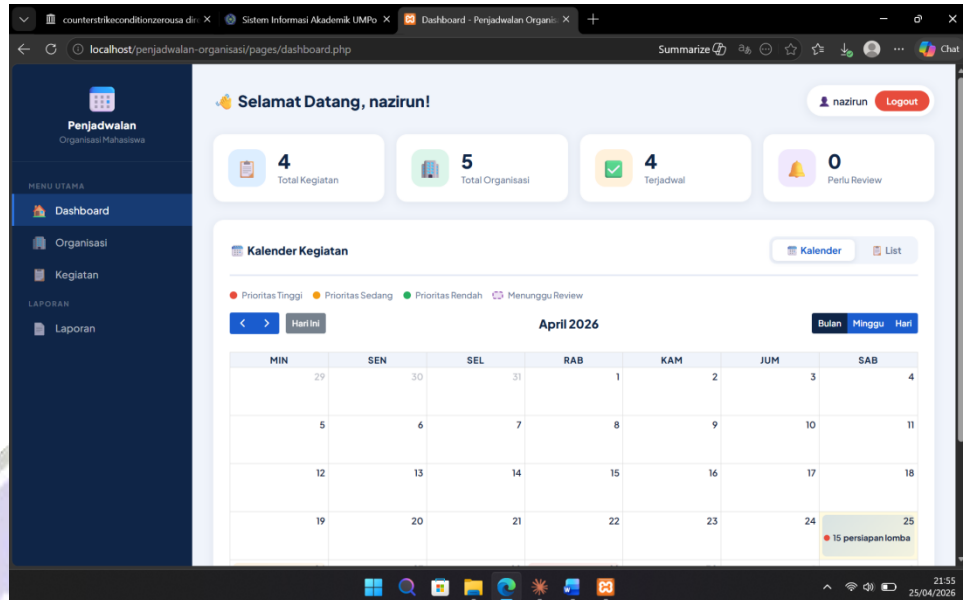


Gambar 2.3 Tangkapan layar modal registrasi akun

Pada Gambar 2.3, Setelah menekan tulisan “Daftar di sini”, sebuah *popup* akan muncul berisi kolom nama, email, password, dan konfirmasi password untuk membuat akun baru. Lalu tekan tombol daftar untuk

menyimpan. Jika sudah memiliki akun, klik tulisan “Login di sini” untuk kembali ke *popup* sebelumnya.

2.3 Halaman Dashboard Admin (dashboard.php)

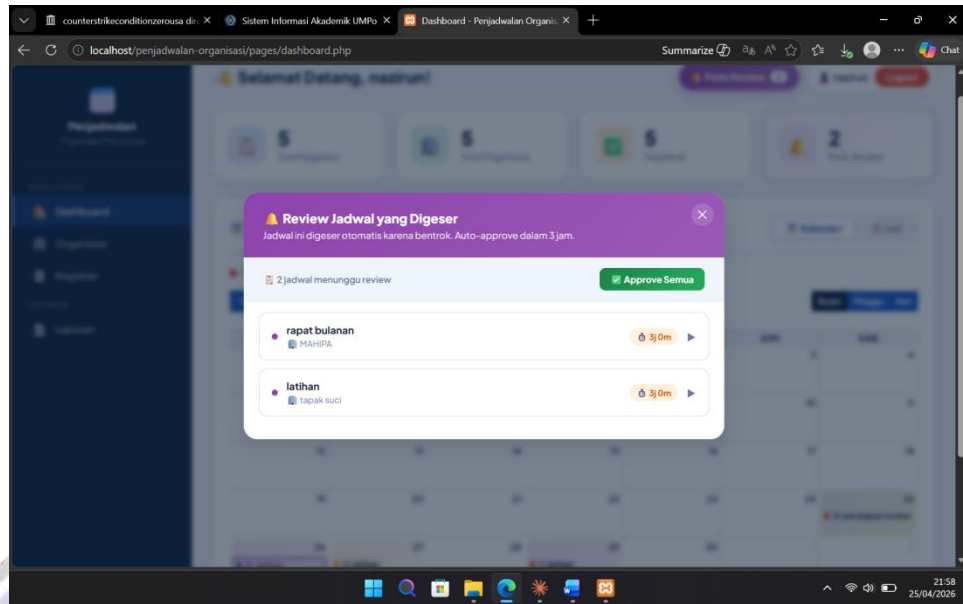


Gambar 2.4 Tangkapan layar halaman dashboard admin

Pada Gambar 2.4, Setelah login, pengguna akan masuk langsung ke dasbor utama. Di sisi kiri terdapat *sidebar* berisi tab untuk ke halaman kegiatan untuk mengorganisir kegiatan-kegiatan yang sudah di inputkan lalu organisasi untuk mengorganisir organisasi apa saja yang tersimpan dan terakhir adalah laporan untuk ekspor menjadi PDF atau CSV.

Di dasbor, ada kalender yang menampilkan semua jadwal yang sudah diinputkan dengan beberapa *stat card* d atasnya menampilkan total kegiatan, total organisasi, total terjadwal dan yang masih di masa review atau tenggang.

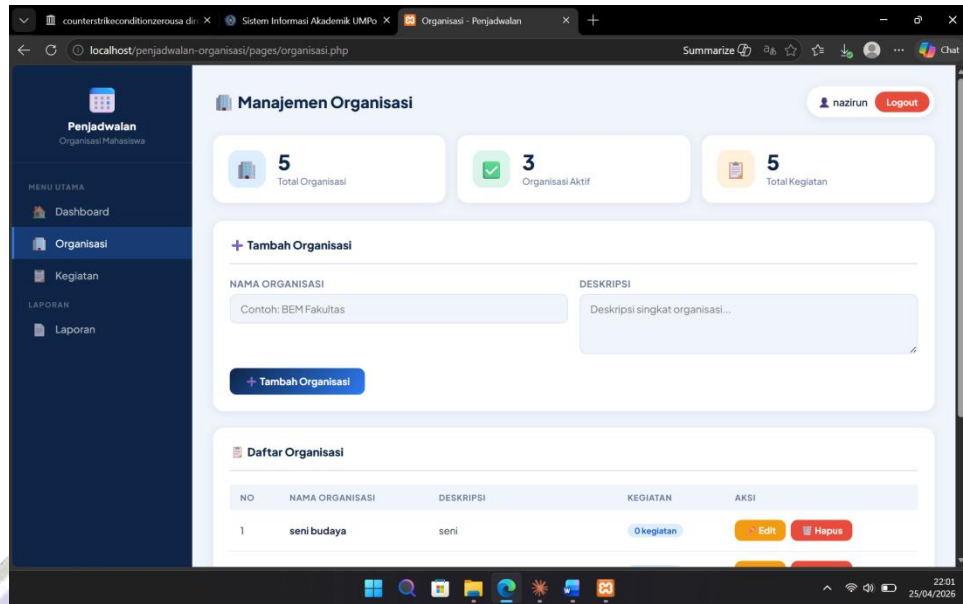
2.4 Modal Review Jadwal Buffer



Gambar 2.5 Tangkapan layar modal review jadwal

Pada Gambar 2.5, Tangkapan layar *popup* dari tombol “Perlu Review” yang ada di kanan atas layar berwarna ungu menampilkan beberapa kegiatan yang bentrok lalu digeser secara otomatis. Disini pengguna dapat menentukan apakah akan diubah secara manual atau disetujui dengan yang sudah diberikan oleh sistem secara otomatis. Selain itu juga pengguna dapat melihat informasi lebih jauh soal atribut per kegiatan untuk mengidentifikasinya.

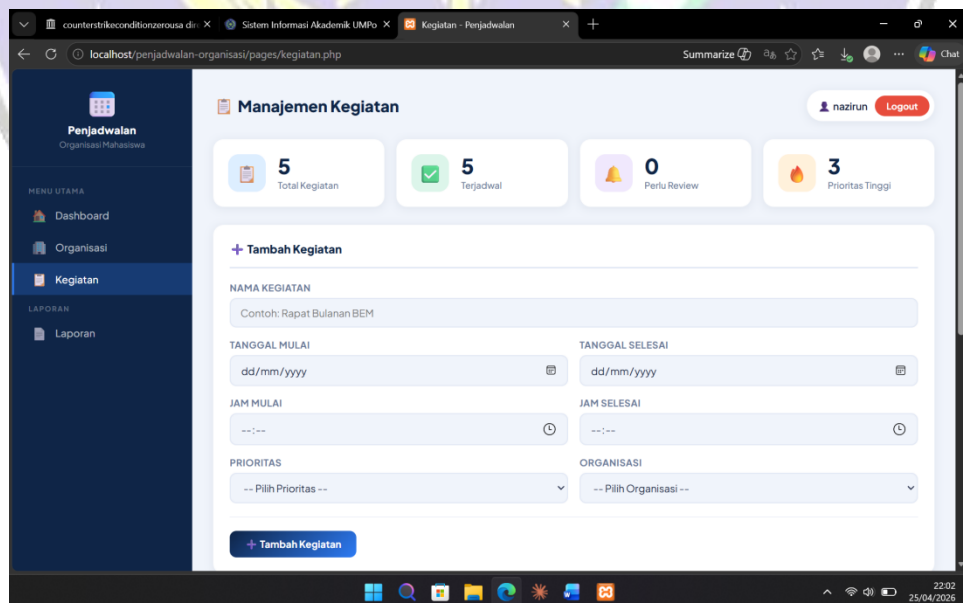
2.5 Halaman Manajemen Organisasi (organisasi.php)



Gambar 2.6 Tangkapan layar halaman manajemen organisasi

Pada Gambar 2.6, Tangkapan layar *page* Organisasi. Disini pengguna dapat mengelola seluruh organisasi yang sudah diinputkan maupun yang akan dimasukkan. Sistem CRUD sudah ditanam sejak awal untuk memudahkan pengguna dalam mengorganisir semua organisasi yang menggunakan program ini. Terdapat juga 3 *stat card* yaitu total organisasi, Organisasi Aktif, dan Total Kegiatan.

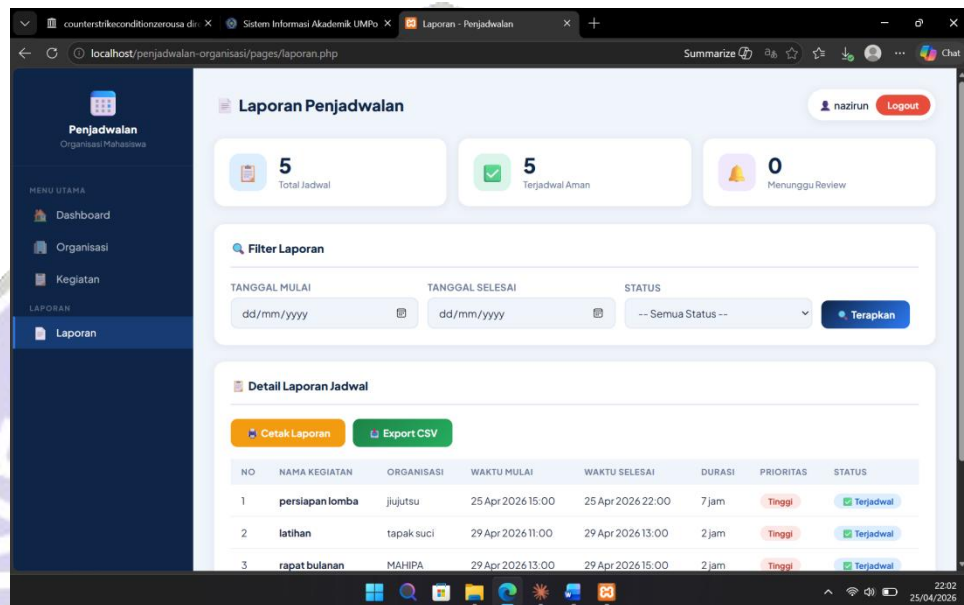
2.6 Halaman Manajemen Kegiatan (kegiatan.php)



Gambar 2.7 Tangkapan layar halaman manajemen kegiatan

Pada Gambar 2.7, Tangkapan layar *page* Kegiatan. Pengguna disini juga mendapat sistem CRUD untuk mengorganisir seluruh kegiatan yang sudah diinputkan maupun yang akan dimasukkan. Juga terdapat 4 *stat card* yaitu Total Kegiatan, Total Terjadwal, Perlu Review, dan Total Prioritas Tinggi.

2.7 Halaman Laporan (laporan.php)



Gambar 2.8 Tangkapan layar halaman laporan

Pada Gambar 2.8, Tangkapan layar *page* Laporan dimana berisi review seluruh kegiatan data yang nanti akan dapat diekspor menjadi PDF ataupun CSV. Selain itu juga ada fitur untuk filtering data yang akan dieskpor. Terdapat 3 *stat card* yaitu Total Jadwal, Terjadwal Aman, dan Menunggu Review.

3. PENGUJIAN HASIL

3.1 Metode Pengujian

Metode pengujiannya menggunakan dua metode berbeda:

1. *Blackbox Testing*

Blackbox Testing adalah pengujian seluruh fitur dalam sistem dari berbagai skenario yang telah ditentukan untuk memastikan semua fungsi dalam sistem berjalan dengan baik.

2. *User Acceptance Testing* (UAT)

Pengujian ini berfokus dalam reaksi atau pengalaman langsung yang diberikan oleh target proyek ini yaitu pengurus organisasi mahasiswa di kampus. Pengujiannya berfokus di kemudahan penggunaan, seberapa baik output dan juga seberapa baik apa program menyelesaikan masalah yang telah diangkat terhadap pengguna.

3.2 Hasil *Blackbox Testing*

Tabel 3.1 Hasil *Blackbox Testing*

No	Fitur	Skenario Input	Expected Output	Actual Output	Status
1	Login Admin	Email & password benar	Redirect ke dashboard	Berhasil masuk dashboard	Berhasil
2	Login Admin	Email atau password salah	Pesan error muncul	Pesan error tampil	Berhasil
3	Registrasi	Password & konfirmasi tidak sama	Pesan error muncul	Pesan error tampil	Berhasil
4	Tambah Organisasi	Nama: BEM Fakultas	Data tersimpan & muncul di tabel	Data tersimpan	Berhasil
5	Tambah Kegiatan	Kegiatan baru tanpa konflik	Status: aman, jadwal langsung dipakai	Status aman	Berhasil

No	Fitur	Skenario Input	Expected Output	Actual Output	Status
6	Algoritma Greedy	Tambah kegiatan bentrok dengan jadwal lain	Jadwal digeser, status: reschedule	Jadwal digeser otomatis	Berhasil
7	Buffer Review	Klik Approve pada modal review	Status berubah jadi aman	Status update real-time	Berhasil
8	Auto-Commit	Tunggu 3 jam tanpa aksi user	Status reschedule jadi aman	Auto-commit berhasil	Berhasil
9	Toggle View	Switch kalender ke list	List tampil dengan data sama	View berubah dengan benar	Berhasil
10	Filter & Search	Filter prioritas Tinggi	Tampil hanya kegiatan prioritas tinggi	Filter bekerja	Berhasil
11	Export Laporan	Klik Export CSV	File CSV terunduh	File terunduh	Berhasil
12	Cetak Laporan	Klik Cetak Laporan	Dialog print browser muncul	Dialog print muncul	Berhasil

Setelah beberapa skenario telah diuji, hasil keseluruhannya menunjukkan hasil yang memuaskan. Sehingga program dinyatakan berjalan dengan baik sesuai dengan yang diharapkan.

3.3 Hasil *User Acceptance Testing* (UAT)

Dalam pengujian ini, kami langsung meminta ke beberapa pengurus organisasi dari UKM kampus untuk langsung mencoba menggunakan program ini dengan penjelasan seluruh fungsi dan fitur agar terpapar seluruhnya. Mereka diminta menilai dalam skala 5 sampai 1, 5 untuk sangat baik dan 1 untuk sangat buruk.

Tabel 3.2 Feedback Pengguna (Pengurus Organisasi Mahasiswa)

Aspek	Skor (1-5)	Keterangan
Kemudahan Penggunaan	4.7	Antarmuka intuitif, sidebar navigasi mudah dipahami
Kecepatan Proses	4.9	Algoritma greedy bekerja instan saat kegiatan ditambahkan
Kejelasan Output	4.8	Color-coded calendar dan badge status sangat membantu
Fitur Buffer Review	4.9	Memberikan rasa kontrol terhadap hasil otomatis
Tampilan Interface	4.8	Desain modern, responsif, dan rapi
Manfaat Keseluruhan	4.8	Sangat membantu dalam koordinasi antar organisasi

Secara keseluruhan, kita bisa melihat bahwa para pengguna sangat menikmati dan juga senang menggunakan program ini dengan rata-rata skor per fungsi mendapat sekitar 4,8.

Sehingga dapat disimpulkan juga di pengujian kedua ini, program berjalan sesuai dengan ekspektasi dan juga dapat memenuhi kebutuhan target subjek proyek ini dengan baik.

4. PENUTUP

4.1 Kesimpulan

Aplikasi Penjadwalan Kegiatan Organisasi berbasis Algoritma Greedy telah berhasil dibangun dan diuji secara menyeluruh. Sistem ini mampu menyusun jadwal kegiatan secara otomatis dengan mempertimbangkan prioritas dan rentang waktu yang dimasukkan oleh pengguna. Ketika dua atau lebih kegiatan saling bertabrakan, sistem akan secara mandiri menggeser kegiatan dengan prioritas lebih rendah ke slot waktu lain yang masih kosong, tanpa perlu campur tangan manual dari admin.

Salah satu fitur yang menjadi keunggulan aplikasi ini adalah sistem buffer review. Alih-alih langsung memfinalisasi hasil pergeseran, sistem memberikan jeda 3 jam agar admin dapat meninjau kembali sebelum jadwal benar-benar dianggap final. Jika dalam waktu tersebut tidak ada tindakan, mekanisme auto-commit akan bekerja secara otomatis sehingga operasional sistem tetap berjalan lancar. Fitur ini hadir sebagai jawaban atas salah satu kelemahan umum sistem otomatis, yaitu kurangnya ruang bagi pengguna untuk memverifikasi hasil yang dihasilkan mesin.

Hasil *User Acceptance Testing* menunjukkan rata-rata skor 4.8 dari 5.0, yang mencerminkan penerimaan yang sangat baik dari para pengurus organisasi mahasiswa sebagai calon pengguna. Secara keseluruhan, sistem ini memberikan solusi nyata untuk mencegah benturan jadwal antar organisasi, sekaligus menyediakan tampilan kalender yang bisa diakses publik sebagai bentuk transparansi informasi.

Untuk pengembangan selanjutnya, disarankan beberapa hal:

a. Notifikasi aktif untuk buffer review

Saat ini admin hanya bisa mengetahui adanya jadwal dalam masa review jika membuka dashboard. Ke depan, sistem sebaiknya dilengkapi notifikasi via email atau push notification agar admin bisa segera merespons tanpa harus terus memantau halaman.

b. Durasi buffer yang bisa dikonfigurasi

Masa buffer saat ini dikunci pada 3 jam dan tidak bisa diubah. Akan lebih fleksibel jika admin dapat menyesuaikan sendiri durasi buffer sesuai kebutuhan organisasinya.

c. Peningkatan keamanan password

Sistem saat ini masih menggunakan MD5 untuk menyimpan password, yang sudah diketahui memiliki kelemahan keamanan. Sebaiknya diganti dengan algoritma hashing modern seperti bcrypt yang jauh lebih aman.

d. Sistem multi-role pengguna

Untuk skenario penggunaan yang lebih kompleks lintas organisasi, perlu ditambahkan pembagian peran yang lebih detail, misalnya admin per organisasi dan super admin yang memiliki akses penuh ke seluruh data.

e. Export laporan ke format PDF

Saat ini laporan hanya bisa diekspor dalam format CSV. Penambahan opsi ekspor ke PDF akan sangat membantu ketika laporan perlu didistribusikan secara resmi atau dicetak dengan format yang lebih rapi.

DAFTAR PUSTAKA

1. M. Fowler. (2021). *Patterns of Enterprise Application Architecture*. Boston: Addison-Wesley.
2. A. Solichin. (2022). *Pemrograman Web dengan PHP dan MySQL*, 2nd ed. Jakarta: Budi Luhur.
3. T. H. Cormen, C. E. Leiserson, R. L. Rivest, dan C. Stein. (2022). *Introduction to Algorithms*, 4th ed. Cambridge, MA: MIT Press.

