PEMBANGUNAN APLIKASI PENJADWALAN RAPAT BERBASIS WEB MENGGUNAKAN *FRAMEWORK* LARAVEL (Studi Kasus: Diskominfotik Kabupaten Bandung Barat)

TUGAS AKHIR

Disusun sebagai salah satu syarat untuk kelulusan
Program Strata 1, Program Studi Teknik Informatika,
Universitas Pasundan Bandung

Oleh :

Salma Az Zahra
NRP : 24.3040.107

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PASUNDAN
BANDUNG
JULI 2026

LEMBAR PENGESAHAN
LAPORAN TUGAS AKHIR

Telah disetujui dan disahkan Laporan Tugas Akhir, dari :

Nama	:	Salma Az Zahra
NPM	:	24.3040.107

Dengan Judul :

"Pembangunan Aplikasi Penjadwalan Rapat Berbasis Web Menggunakan *Framework* Laravel (Studi Kasus: Diskominfotik Kabupaten Bandung Barat)"

Bandung, 28 Juli 2026

Menyetujui,
Pembimbing Utama,

FAKULTAS TEKNIK UNIVERSITAS SILIWANGI

(Ade Sukendar, S.T., M.T.)

LEMBAR PERNYATAAN KEASLIAN TUGAS AKHIR

Saya menyatakan dengan sesungguhnya bahwa:

1. Tugas Akhir ini adalah benar-benar asli dan belum pernah diajukan untuk mendapatkan gelar akademik, baik di Universitas Pasundan Bandung maupun di Perguruan Tinggi lainnya.
2. Tugas Akhir ini merupakan gagasan, rumusan, dan penelitian saya sendiri, tanpa bantuan pihak lain kecuali arahan dari tim Dosen Pembimbing.
3. Dalam Tugas Akhir ini tidak terdapat karya atau pendapat orang lain, kecuali bagian-bagian tertentu dalam penulisan laporan Tugas Akhir yang saya kutip dari hasil karya orang lain telah dituliskan sumbernya secara jelas sesuai dengan norma, kaidah, dan etika penulisan karya ilmiah, serta disebutkan dalam Daftar Pustaka pada Tugas Akhir ini.
4. Kakas, perangkat lunak, dan alat bantu kerja lainnya yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab saya, bukan tanggung jawab Universitas Pasundan Bandung.

Apabila di kemudian hari ditemukan seluruh atau sebagian laporan Tugas Akhir ini bukan hasil karya saya sendiri atau adanya plagiasi dalam bagian-bagian tertentu, saya bersedia menerima sanksi akademik, termasuk pencabutan gelar akademik yang saya sandang sesuai dengan norma yang berlaku di Universitas Pasundan, serta perundang-undangan lainnya.

Bandung, 28 Juli 2026

Yang membuat pernyataan,

(Salma Az Zahra)

NRP : 24.3040.107

ABSTRAK

Penjadwalan rapat di Dinas Komunikasi, Informatika, dan Statistik Kabupaten Bandung Barat masih menghadapi kendala dalam penyampaian informasi jadwal dan perubahan jadwal kepada operator yang belum terpusat, pencatatan peminjaman dan pengembalian peralatan yang masih dilakukan secara terpisah, serta belum adanya mekanisme notifikasi otomatis kepada operator. Kondisi tersebut menyebabkan proses koordinasi kegiatan rapat menjadi kurang efektif. Penelitian ini bertujuan membangun aplikasi penjadwalan rapat berbasis web yang mampu mendukung pengelolaan jadwal rapat, dokumentasi penggunaan peralatan, notifikasi jadwal kepada operator melalui WhatsApp secara otomatis serta pembuatan tautan Zoom secara otomatis.

Pembangunan aplikasi dilakukan menggunakan framework Laravel dengan metode pengembangan perangkat lunak Waterfall. Tahapan penelitian meliputi identifikasi masalah, pengumpulan data melalui wawancara dan studi literatur, analisis kebutuhan sistem, perancangan menggunakan Unified Modeling Language (UML), implementasi aplikasi, serta pengujian menggunakan metode Black Box Testing. Aplikasi dikembangkan dengan integrasi WhatsApp Gateway dan Zoom API.

Hasil penelitian berupa aplikasi penjadwalan rapat berbasis web yang mampu mendukung pengelolaan data pengguna, penjadwalan rapat, penugasan operator, peminjaman dan pengembalian peralatan, pengiriman notifikasi WhatsApp secara otomatis, pembuatan tautan Zoom secara otomatis, serta penyajian laporan kegiatan rapat. Berdasarkan hasil pengujian, seluruh fungsi utama aplikasi berjalan sesuai dengan kebutuhan yang telah ditetapkan sehingga aplikasi mampu meningkatkan efektivitas pengelolaan jadwal rapat, koordinasi operator, dan dokumentasi penggunaan peralatan.

Kata Kunci: penjadwalan rapat, Laravel, aplikasi berbasis web, WhatsApp Gateway, Zoom API, peminjaman peralatan.

ABSTRACT

The meeting scheduling process at the Department of Communication, Informatics, and Statistics of West Bandung Regency still faces challenges in delivering meeting schedules and schedule updates to operators through a centralized system, recording equipment borrowing and return activities separately, and the absence of an automatic notification mechanism for operators. These conditions reduce the effectiveness of meeting activity coordination. This study aims to develop a web-based meeting scheduling application that supports meeting schedule management, equipment usage documentation, automatic meeting notifications to operators via WhatsApp, and automatic Zoom meeting link generation.

The application was developed using the Laravel framework and the Waterfall software development method. The research stages consisted of problem identification, data collection through interviews and literature studies, system requirements analysis, system design using Unified Modeling Language (UML), application implementation, and application testing using the Black Box Testing method. The application integrates a WhatsApp Gateway and the Zoom API.

The result of this study is a web-based meeting scheduling application that supports user management, meeting scheduling, operator assignment, equipment borrowing and return management, automatic WhatsApp notifications, automatic Zoom meeting link generation, and meeting report generation. Based on the testing results, all major system functions operated according to the specified requirements. Therefore, the application can improve the effectiveness of meeting schedule management, operator coordination, and equipment usage documentation.

Keywords: meeting scheduling, Laravel, web-based application, WhatsApp Gateway, Zoom API, equipment borrowing.

KATA PENGANTAR

Puji dan syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul "Pembangunan Aplikasi Penjadwalan Rapat Berbasis Web Menggunakan *Framework* Laravel (Studi Kasus: Diskominfotik Kabupaten Bandung Barat)" sebagai salah satu syarat untuk memperoleh gelar Sarjana Informatika pada Program Studi Teknik Informatika Universitas Pasundan Bandung.

Selama penyusunan Tugas Akhir ini penulis tidak terlepas dari dukungan berbagai pihak. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan ucapan terima kasih kepada:

1. Allah SWT atas rahmat, kemudahan, dan kelancaran yang diberikan kepada penulis selama penyusunan Tugas Akhir.
2. Bapak Prof. Dr. H. Azhar Affandi, S.E., M.Sc., selaku Rektor Universitas Pasundan.
3. Bapak Prof. Dr. Ir. H. Yusman Taufik, M.P., selaku Dekan Fakultas Teknik Universitas Pasundan.
4. Bapak Anggoro Ari Nur Cahyo, S.T., M.Kom., selaku Ketua Program Studi Teknik Informatika Universitas Pasundan.
5. Bapak Ade Sukendar, S.T., M.T., selaku Dosen Pembimbing yang telah meluangkan waktu, memberikan bimbingan, arahan, serta masukan selama proses penyusunan Tugas Akhir ini.
6. Bapak Wanda Gusdya Purnama, S.T., M.T., dan Bapak Fajar Darmawan, S.T., M.Kom., selaku Dosen Penguji yang telah meluangkan waktu serta memberikan evaluasi, arahan, dan saran yang membangun untuk penyempurnaan Tugas Akhir ini.
7. Bapak dan Ibu Dosen serta seluruh Staf Pengajar Program Studi Teknik Informatika Universitas Pasundan Bandung yang telah memberikan ilmu pengetahuan, pengalaman, dan dukungan selama penulis menjalani perkuliahan.
8. Bapak Tedy Yulianto, S.Sos., M.M., beserta Bapak/Ibu pegawai di Dinas Komunikasi, Informatika, dan Statistik Kabupaten Bandung Barat yang telah memberikan kesempatan, informasi, serta dukungan kepada penulis selama proses penelitian berlangsung.
9. Orang tua tercinta, Bapak Yusup Sudrajat, Ibu Dewi Susanti, dan Ibu Rd. Dewi Herlianawati, yang senantiasa memberikan doa, kasih sayang, perhatian, dukungan, dan motivasi kepada penulis selama menempuh pendidikan hingga menyelesaikan Tugas Akhir ini.
10. Saudara-saudari penulis, Dea Adelia, Raja Zaidan Khan, Rafka Raihatur Rajani, dan Muhammad Abrisam Ramadhan, yang senantiasa memberikan dukungan, semangat, perhatian, serta keceriaan melalui canda dan tingkah laku yang selalu membuat penulis tertawa di tengah proses penyelesaian Tugas Akhir ini.

11. Keluarga besar penulis yang tidak dapat disebutkan satu per satu, yang senantiasa memberikan doa, dukungan, perhatian, dan bantuan kepada penulis dalam berbagai keadaan selama menjalani pendidikan hingga menyelesaikan Tugas Akhir ini.
12. Zeinrauf Hakim Husyeimi, S.Ikom., yang telah menjadi *support system* serta memberikan motivasi, perhatian, dan dukungan selama proses penyelesaian Tugas Akhir ini.
13. Mila, Elsa, dan Fitri, yang telah menjadi bagian dari perjalanan penulis selama menjalani bangku perkuliahan. Terima kasih atas kebersamaan, cerita, serta berbagai momen menyenangkan yang telah dilalui bersama. Canda dan tawa yang hadir di tengah kesibukan perkuliahan menjadi kenangan yang berharga bagi penulis. Terima kasih telah menemani berbagai cerita hingga penulis dapat menyelesaikan Tugas Akhir ini.
14. Egi, Ayu, Ai, dan Suci, yang telah menemani penulis sejak awal perkuliahan hingga saat ini. Terima kasih telah menjadi tempat berbagi cerita, mendengarkan keluh kesah, serta memberikan ruang bagi penulis untuk menjadi diri sendiri tanpa perlu menyembunyikan perasaan. Berbagai suka, duka, dan pengalaman yang telah dilalui bersama menjadi bagian yang berharga dalam perjalanan perkuliahan penulis.
15. Annisa, Ilman, Indri, Davina, Feby, Yuma, dan Nova, yang penulis kenal selama menjalani perkuliahan di Universitas Pasundan. Terima kasih atas kebersamaan, dukungan, dan bantuan dalam menyelesaikan berbagai tugas maupun Tugas Akhir, serta telah menjadi partner Tugas Akhir untuk saling berdiskusi, berbagi cerita, dan memberikan semangat selama proses penyelesaian.
16. Aufa dan Dayu, yang meskipun baru penulis kenal, telah menemani dan memberikan dukungan selama proses penyusunan Tugas Akhir. Terima kasih atas waktu, diskusi, canda, dan kebersamaan dalam melewati proses penyusunan Tugas Akhir, termasuk malam-malam panjang yang dilalui bersama.
17. Teman-teman seperjuangan lainnya yang tidak dapat disebutkan satu per satu, yang telah menemani perjalanan perkuliahan, saling memberikan dukungan, berbagi pengalaman, dan membantu penulis dalam menghadapi berbagai proses selama perkuliahan hingga penyusunan Tugas Akhir.
18. Member Discord Sweet Peace dan teman-teman di media sosial, khususnya TikTok, yang telah menjadi teman berbagi dan memberikan hiburan serta semangat selama proses penyelesaian Tugas Akhir ini.

Penulis menyadari bahwa Tugas Akhir ini masih jauh dari sempurna. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan demi perbaikan di masa yang akan datang.

Semoga Tugas Akhir ini dapat memberikan manfaat bagi penulis maupun pembaca.

Bandung, 28 Juli 2026

Salma Az Zahra

DAFTAR ISI

LEMBAR PENGESAHAN LAPORAN TUGAS AKHIR i

LEMBAR PERNYATAAN KEASLIAN TUGAS AKHIR ii

ABSTRAK iii

ABSTRACT iv

KATA PENGANTAR v

DAFTAR ISI viii

DAFTAR TABEL xii

DAFTAR GAMBAR xv

DAFTAR SIMBOL xvii

DAFTAR ISTILAH xx

BAB 1 PENDAHULUAN 1-1

1.1 Latar Belakang 1-1

1.2 Identifikasi Masalah 1-2

1.3 Tujuan Tugas Akhir 1-3

1.4 Lingkup Tugas Akhir 1-3

1.5 Metodologi Tugas Akhir 1-4

1.6 Sistematika Penulisan Tugas Akhir 1-5

BAB 2 LANDASAN TEORI 2-1

2.1 Penjadwalan 2-1

2.1.1 Penjadwalan Rapat 2-1

2.2 Inventaris 2-2

2.3 Aplikasi 2-2

2.4 Aplikasi Berbasis *Web* 2-2

2.5 *Framework* 2-3

2.6 Laravel 2-4

2.7 Basis Data 2-5

2.8 MySQL 2-6

2.9 Model View Controller (MVC) 2-7
2.10 UML 2-7
2.11 Application Programming Interface (API) 2-8
2.12 Bahasa Pemrograman 2-8
2.12.1 HTML 2-9
2.12.2 CSS 2-9
2.12.3 Tailwind CSS 2-9
2.12.4 PHP 2-10
2.12.5 JavaScript 2-11
2.12.6 SQL 2-11
2.13 Metode Waterfall 2-11
2.14 Pengujian Perangkat Lunak 2-13
2.15 Black Box Testing 2-14
2.16 Penelitian Terdahulu 2-14
BAB 3 SKEMA PENELITIAN 3-1
3.1 Alur Penyelesaian Tugas Akhir 3-1
3.2 Perumusan Masalah 3-2
3.2.1 Analisis Sebab Akibat 3-3
3.2.2 Analisis Solusi Masalah 3-4
3.3 Profil Penelitian 3-5
3.3.1 Bagan Struktur Organisasi Instansi 3-6
BAB 4 ANALISIS DAN PERANCANGAN 4-1
4.1 Sistem Saat Ini (Current System) 4-1
4.1.1 Diagram Use Case Bisnis 4-1
4.1.2 Diagram Aktivitas 4-2
4.2 Analisis Kebutuhan 4-6
4.2.1 Analisis Stakeholder 4-7
4.2.2 Analisis Fungsional 4-7
4.2.3 Analisis Pengguna 4-8

4.2.4 Pemodelan Berbasis Skenario 4-8
4.2.5 Deskripsi Use Case 4-10
4.2.6 Skenario Use Case 4-10
4.3 Pemodelan Data 4-17
4.3.1 Pemodelan Berbasis Perilaku 4-17
4.4 Pemodelan Berbasis Kelas 4-31
4.4.1 Identifikasi Kelas-kelas Analisis 4-31
4.4.2 Menentukan Atribut 4-32
4.4.3 Menentukan Perilaku 4-34
4.5 Perancangan Perangkat Lunak 4-35
4.5.1 Perancangan Data 4-36
4.5.2 Perancangan Kelas 4-37
4.5.3 Objek Data 4-39
4.5.4 Atribut Data 4-39
4.6 Perancangan Antarmuka Pengguna 4-41
4.7 Arsitektur Perangkat Lunak 4-49
BAB 5 IMPLEMENTASI DAN PENGUJIAN 5-1
5.1 Kontruksi Perangkat Lunak 5-1
5.1.1 Spesifikasi Perangkat Keras 5-1
5.1.2 Spesifikasi Perangkat Lunak dan kakas 5-1
5.1.3 Pustaka dan Bahasa yang digunakan 5-2
5.2 Pengkodean 5-2
5.2.1 Tahap Pengkodean Aplikasi 5-2
5.3 Implementasi Kode 5-6
5.4 Implementasi Tampilan Antarmuka Pengguna 5-13
5.5 Pengujian Perangkat Lunak 5-25
BAB 6 KESIMPULAN DAN SARAN 6-1
6.1 Kesimpulan 6-1
6.2 Saran 6-1

DAFTAR PUSTAKA xxi

LAMPIRAN L-1

DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu 2-15
Tabel 3.1 Langkah Penelitian 3-1
Tabel 3.2 Analisis Sebab Akibat 3-3
Tabel 3.3 Analisis Solusi Masalah 3-4
Tabel 4.1 Diagram Use Case Bisnis 4-1
Tabel 4.2 Diagram Aktivitas Perencanaan Kegiatan Rapat 4-2
Tabel 4.3 Diagram Aktivitas Perubahan Jadwal Rapat 4-3
Tabel 4.4 Diagram Aktivitas Persiapan Kegiatan Rapat 4-4
Tabel 4.5 Diagram Aktivitas Pelaksanaan Kegiatan Rapat 4-5
Tabel 4.6 Diagram Aktivitas Penyelesaian Kegiatan Rapat 4-6
Tabel 4.7 Analisis Stakeholder 4-7
Tabel 4.8 Analisis Fungsional 4-7
Tabel 4.9 Fitur 4-8
Tabel 4.10 Analisis Pengguna 4-8
Tabel 4.11 Use Case 4-9
Tabel 4.12 Deskripsi Use Case 4-10
Tabel 4.13 Skenario Use Case Melihat Jadwal Rapat 4-10
Tabel 4.14 Skenario Use Case Melakukan Tambah Jadwal Rapat 4-11
Tabel 4.15 Skenario Use Case Melakukan Ubah Jadwal Rapat 4-12
Tabel 4.16 Skenario Use Case Melakukan Batal Jadwal Rapat 4-13
Tabel 4.17 Skenario Use Case Melihat Peralatan 4-13
Tabel 4.18 Skenario Use Case Melihat Peminjaman Peralatan 4-14
Tabel 4.19 Skenario Use Case Melakukan Peminjaman Peralatan 4-14
Tabel 4.20 Skenario Use Case Melakukan Ubah Peminjaman Peralatan 4-15
Tabel 4.21 Skenario Use Case Melakukan Batal Peminjaman Peralatan 4-16
Tabel 4.22 Sequence Melihat Jadwal Rapat 4-18
Tabel 4.23 Sequence Melakukan Tambah Jadwal 4-19
Tabel 4.24 Sequence Melakukan Ubah Jadwal Rapat 4-21
Tabel 4.25 Sequence Melakukan Batal Jadwal Rapat 4-22
Tabel 4.26 Sequence Melihat Peralatan 4-24
Tabel 4.27 Sequence Melihat Peminjaman Peralatan 4-25
Tabel 4.28 Sequence Melakukan Peminjaman Peralatan 4-26
Tabel 4.29 Sequence Melakukan Ubah Peminjaman Peralatan 4-27
Tabel 4.30 Sequence Melakukan Peminjaman Peralatan 4-30
Tabel 4.31 Identifikasi Kelas 4-31

Tabel 4.32 Atribut 4-32
Tabel 4.33 Perilaku 4-34
Tabel 4.34 Perancangan Data 4-36
Tabel 4.35 Objek Data 4-39
Tabel 4.36 Atribut Data 4-39
Tabel 4.37 Tampilan Login 4-41
Tabel 4.38 Tampilan Halaman Dashboard Admin 4-42
Tabel 4.39 Tampilan Halaman Users 4-42
Tabel 4.40 Tampilan Halaman Tambah User 4-43
Tabel 4.41 Tampilan Halaman Jadwal 4-44
Tabel 4.42 Tampilan Halaman Tambah Jadwal 4-44
Tabel 4.43 Tampilan Halaman Laporan 4-45
Tabel 4.44 Tampilan Halaman Dashboard Operator 4-46
Tabel 4.45 Tampilan Halaman Jadwal Operator 4-46
Tabel 4.46 Tampilan Halaman Peralatan 4-47
Tabel 4.47 Tampilan Halaman Peminjaman 4-48
Tabel 4.48 Tampilan Halaman Ajukan Peminjaman 4-49
Tabel 5.1 Spesifikasi Perangkat Keras 5-1
Tabel 5.2 Spesifikasi Perangkat Lunak 5-1
Tabel 5.3 Pustaka dan Bahasa yang digunakan 5-2
Tabel 5.4 Struktur Folder 5-3
Tabel 5.5 Penjelasan Struktur Umum Proyek 5-4
Tabel 5.6 Model 5-5
Tabel 5.7 Controller 5-5
Tabel 5.8 View 5-6
Tabel 5.9 Kode Model Penjadwalan.php 5-6
Tabel 5.10 Kode Model Peminjaman.php 5-8
Tabel 5.11 Kode PenjadwalanController.php 5-10
Tabel 5.12 Kode PenjadwalanController.php 5-10
Tabel 5.13 Kode index.blade.php halaman jadwal 5-11
Tabel 5.14 Kode operator-index.blade.php halaman peminjaman 5-12
Tabel 5.15 Antarmuka Halaman Login 5-13
Tabel 5.16 Antarmuka Halaman Dashboard Admin 5-14
Tabel 5.17 Antarmuka Halaman Users 5-15
Tabel 5.18 Antarmuka Halaman Tambah User 5-15
Tabel 5.19 Antarmuka Halaman Ubah User 5-16
Tabel 5.20 Antarmuka Nonaktifkan User 5-16

Tabel 5.21 Antarmuka Halaman Jadwal Rapat 5-17
Tabel 5.22 Antarmuka Halaman Tambah Jadwal 5-18
Tabel 5.23 Antarmuka Halaman Ubah Jadwal 5-19
Tabel 5.24 Antarmuka Batalkan Jadwal 5-19
Tabel 5.25 Antarmuka Halaman Laporan 5-20
Tabel 5.26 Antarmuka Halaman Dashboard Operator 5-20
Tabel 5.27 Antarmuka Halaman Jadwal Rapat Saya 5-21
Tabel 5.28 Antarmuka Halaman Peralatan 5-21
Tabel 5.29 Antarmuka Halaman Peminjaman Peralatan 5-22
Tabel 5.30 Antarmuka Halaman Pengajuan Peminjaman Peralatan 5-23
Tabel 5.31 Antarmuka Halaman Ubah Pengajuan Peminjaman Peralatan 5-24
Tabel 5.32 Antarmuka Batalkan Pengajuan Peminjaman Peralatan 5-24
Tabel 5.33 Pengujian Perangkat Lunak 5-25
Tabel 5.34 Rekapitulasi Hasil Pengujian 5-34

DAFTAR GAMBAR

Gambar 1.1 Metodologi Dalam Tugas Akhir 1-4
Gambar 2.1 Model Waterfall 2-11
Gambar 2.2 Ilustrasi Black Box Testing 2-14
Gambar 3.1 Fishbone diagram analisis sebab akibat 3-3
Gambar 3.2 Fishbone diagram analisis solusi masalah 3-4
Gambar 3.3 Logo Diskominfotik Kabupaten Bandung Barat 3-5
Gambar 3.4 Bagan Struktur Organisasi Instansi 3-6
Gambar 4.1 Diagram Use Case Bisnis 4-1
Gambar 4.2 Diagram Aktivitas Perencanaan Rapat 4-2
Gambar 4.3 Diagram Aktivitas Perubahan Kegiatan Rapat 4-3
Gambar 4.4 Diagram Aktivitas Persiapan Kegiatan Rapat 4-4
Gambar 4.5 Diagram Aktivitas Pelaksanaan Kegiatan Rapat 4-5
Gambar 4.6 Diagram Aktivitas Penyelesaian Kegiatan Rapat 4-6
Gambar 4.7 Use Case 4-9
Gambar 4.8 Sequence Melihat Jadwal Rapat 4-18
Gambar 4.9 Sequence Melakukan Tambah Jadwal 4-19
Gambar 4.10 Sequence Melakukan Ubah Jadwal Rapat 4-21
Gambar 4.11 Sequence Melakukan Batal Jadwal Rapat 4-23
Gambar 4.12 Sequence Melihat Peralatan 4-24
Gambar 4.13 Sequence Melihat Peminjaman Peralatan 4-25
Gambar 4.14 Sequence Melakukan Peminjaman Peralatan 4-26
Gambar 4.15 Sequence Melakukan Ubah Peminjaman Peralatan 4-28
Gambar 4.16 Sequence Melakukan Batal Peminjaman Peralatan 4-30
Gambar 4.17 Perancangan Kelas 4-38
Gambar 4.18 Tampilan Login 4-41
Gambar 4.19 Tampilan Halaman Dashboard Admin 4-42
Gambar 4.20 Tampilan Halaman Users 4-43
Gambar 4.21 Tampilan Halaman Tambah User 4-43
Gambar 4.22 Tampilan Halaman Jadwal 4-44
Gambar 4.23 Tampilan Halaman Tambah Jadwal 4-45
Gambar 4.24 Tampilan Halaman Laporan 4-45
Gambar 4.25 Tampilan Halaman Dashboard Operator 4-46
Gambar 4.26 Tampilan Halaman Jadwal Operator 4-46
Gambar 4.27 Tampilan Halaman Peralatan 4-47
Gambar 4.28 Tampilan Halaman Peminjaman 4-48

Gambar 4.29 Tampilan Halaman Ajukan Peminjaman Peralatan 4-49
Gambar 4.30 Arsitektur Perangkat Lunak 4-50
Gambar 5.1 Arsitektur Pengkodean 5-3
Gambar 5.2 Struktur Folder 5-3
Gambar 5.3 Model Penjadwalan 5-8
Gambar 5.4 Model Peminjaman 5-9
Gambar 5.5 Controller Penjadwalan 5-10
Gambar 5.6 Controller Peminjaman 5-11
Gambar 5.7 Source Code Penjadwalan.blade.php 5-12
Gambar 5.8 Source Code operator-index.blade.php 5-12
Gambar 5.9 Antarmuka Halaman Login 5-13
Gambar 5.10 Antarmuka Halaman Dashboard Admin 5-14
Gambar 5.11 Antarmuka Halaman Users 5-15
Gambar 5.12 Antarmuka Halaman Tambah User 5-15
Gambar 5.13 Antarmuka Halaman Ubah User 5-16
Gambar 5.14 Antarmuka Nonaktifkan User 5-16
Gambar 5.15 Antarmuka Halaman Jadwal Rapat 5-17
Gambar 5.16 Antarmuka Halaman Tambah Jadwal 5-18
Gambar 5.17 Antarmuka Halaman Ubah Jadwal 5-19
Gambar 5.18 Antarmuka Batalkan Jadwal 5-19
Gambar 5.19 Antarmuka Halaman Laporan 5-20
Gambar 5.20 Antarmuka Halaman Dashboard Operator 5-20
Gambar 5.21 Antarmuka Halaman Jadwal Saya 5-21
Gambar 5.22 Antarmuka Halaman Peralatan 5-22
Gambar 5.23 Antarmuka Halaman Peminjaman Peralatan 5-22
Gambar 5.24 Antarmuka Halaman Ajukan Peminjaman Peralatan 5-23
Gambar 5.25 Antarmuka Halaman Ubah Pengajukan Peminjaman Peralatan 5-24
Gambar 5.26 Antarmuka Batalkan Pengajukan Peminjaman Peralatan 5-24
Gambar 6.1 Dokumentasi Wawancara dengan Sub Koordinator 6-15
Gambar 6.2 Dokumentasi Wawancara dengan Operator1 6-15
Gambar 6.3 Dokumentasi Wawancara dengan Inventaris 6-16
Gambar 6.4 Dokumentasi Wawancara dengan Operator 2 6-16

DAFTAR SIMBOL

SIMBOL DIAGRAM USECASE

No	Simbol	Nama	Deskripsi
1		Aktor	Orang atau sistem lain yang berinteraksi dengan sistem. Aktor merupakan sebuah peran, bukan pengguna tertentu, dan berfungsi untuk memberikan dan menerima informasi dari sistem
2		*Use Case*	Gambaran fungsionalitas sistem agar pengguna memahami kegunaannya. Menampilkan interaksi antara aktor dengan sistem dari sudut pandang pengguna, dengan fokus pada fungsi sistem, bukan urutan proses
3		*Association*	Menghubungkan aktor dengan use case, yang menunjukkan adanya komunikasi dua arah antar elemen
4		*Business Actor*	Mewakili peran yang dijalankan oleh orang atau sistem di luar bisnis yang dimodelkan dan berinteraksi dengan bisnis tersebut
5		*Use Case Business*	Proses yang terjadi dalam bisnis ketika use case diminta oleh Business Actor. Use case ini menggambarkan alur kerja lengkap atau proses bisnis yang menghasilkan produk yang dibutuhkan oleh Business Actor.
6	<<extend>>	*Extend*	menunjukkan adanya fungsi tambahan yang dapat memperluas perilaku suatu use case. Fungsi tersebut hanya dijalankan ketika kondisi tertentu terpenuhi.

SIMBOL DIAGRAM AKTIVITAS

No	Simbol	Nama	Deskripsi
1		*Initial Node*	Menunjukkan awal dari suatu alur kerja pada activity diagram dan digunakan untuk menandai dimulainya suatu proses
2		*Control Flow*	Digunakan untuk mewakili beberapa tindakan (*action*) dan menunjukkan urutan kegiatan selanjutnya

3		*Action/Activity*	Activity menggambarkan tugas dalam alur kerja. Action adalah perilaku sederhana yang tidak dapat diuraikan, sedangkan activity mewakili kumpulan tindakan yang diberi nama
4		*Decision Node*	Menunjukkan adanya pilihan kondisi, sehingga alur proses bisa bercabang sesuai keputusan
5		*Final Node*	Digunakan untuk menunjukkan akhir dari suatu aktivitas

SIMBOL DIAGRAM FLOWCHART

No	Simbol	Nama	Deskripsi
1		*Terminator*	Digunakan untuk menunjukkan bagian awal dan akhir dari suatu proses, alur program, atau tahapan kegiatan pada flowchart
2		Garis Alir	Digunakan untuk menunjukkan arah alur proses dan hubungan antar langkah pada flowchart
3		Proses	Digunakan untuk merepresentasikan suatu proses atau aktivitas yang terjadi pada flowchart
4		*Off Page Connector*	Simbol yang digunakan untuk menunjukkan bahwa alur proses berpindah atau dilanjutkan ke halaman lain

SIMBOL DIAGRAM SEQUENCE

No	Simbol	Nama	Deskripsi
1		*Actor*	Merupakan orang atau sistem yang berada di luar sistem dan berpartisipasi dalam sequence diagram dengan mengirim atau menerima pesan

2		*Boundary*	Menggambarkan bagian antarmuka sistem yang berhubungan langsung dengan pengguna, seperti tampilan atau input dan output pada program
3		*Control*	Mengatur jalannya proses dalam sistem dan menghubungkan antara boundary dan entity class
4		*Entity*	Menggambarkan data atau objek yang ada dalam sistem beserta atributnya
5	1 : message ()	*Message*	Menyampaikan informasi dari satu objek ke objek lain berupa pemanggilan operasi, yang ditunjukkan dengan panah dan diberi label sesuai pesan yang dikirim
6		*Focus of Control and Life Line*	Menunjukkan periode waktu ketika suatu objek sedang menjalankan proses atau menerima pesan dalam sequence diagram

SIMBOL DIAGRAM KELAS

No	Simbol	Nama	Deskripsi
1	Class atribut method	*Class*	Merepresentasikan objek yang dibutuhkan sistem untuk menyimpan informasi, dengan nama, atribut, dan operasi yang ditampilkan dalam satu kotak
2		*Association*	Association merupakan relasi yang mewakili hubungan antara dua kelas atau lebih, termasuk hubungan suatu kelas dengan dirinya sendiri. Relasi ini diberi label berupa kata kerja atau nama peran, dapat terjadi di antara satu kelas atau lebih, serta dilengkapi simbol multiplisitas yang menunjukkan jumlah minimum dan maksimum instance kelas yang dapat dikaitkan

DAFTAR ISTILAH

No	Nama Istilah	Deskripsi
1.	Diskominfotik	Dinas Komunikasi, Informatika, dan Statistik
2.	Pintek	Penjadwalan Infrastruktur dan Teknologi, nama aplikasi yang dibangun dalam penelitian ini
3.	*Front-End*	Bagian aplikasi yang berfungsi menampilkan antarmuka dan interaksi pengguna dengan sistem
4.	*Back-End*	Bagian aplikasi yang berfungsi mengelola logika program, proses data, dan database sistem
5.	WhatsApp	Media sosial percakapan
6.	Zoom	Platform konferensi video yang digunakan untuk pelaksanaan rapat secara daring
7.	Operator	Petugas yang bertanggung jawab sebagai fasilitator teknis dalam kegiatan rapat dengan menyiapkan dan mengoperasikan peralatan pendukung yang dibutuhkan
8.	OTP	*One-Time Password*, yaitu kode verifikasi yang hanya dapat digunakan satu kali untuk proses autentikasi pengguna
9.	*Hybrid*	Metode pelaksanaan rapat yang menggabungkan pertemuan secara luring (*offline*) dan daring (*online*) dalam satu kegiatan
10.	*Password*	Kata sandi yang digunakan sebagai kredensial keamanan untuk mengakses akun pengguna
11.	Platform	Media atau sarana pelaksanaan rapat yang dipilih, meliputi *Online, Offline,* atau *Hybrid*
12.	ORM	*Object Relational Mapping*, yaitu teknik yang digunakan untuk memetakan objek pada aplikasi ke tabel dalam basis data sehingga proses pengolahan data dapat dilakukan tanpa menuliskan perintah SQL secara langsung
13.	*Gateway*	Penghubung yang memungkinkan pertukaran data atau komunikasi antara dua sistem yang berbeda

BAB 1
PENDAHULUAN

1.1 Latar Belakang

Teknologi informasi mengubah aktivitas operasional instansi pemerintahan. Proses yang sebelumnya dilakukan secara konvensional, seperti pencatatan, penyimpanan, dan penyampaian informasi, kini banyak dialihkan ke bentuk digital agar dapat berjalan lebih cepat, akurat, dan mudah dipantau. Pengelolaan kegiatan rutin instansi juga dipengaruhi oleh pergeseran tersebut, termasuk kegiatan rapat yang menjadi bagian dari proses koordinasi antar unit kerja.

Pengelolaan jadwal rapat pada instansi pemerintahan umumnya mencakup penentuan waktu, lokasi, penugasan pihak yang bertanggung jawab atas kebutuhan teknis, hingga kesiapan peralatan pendukung, yang seluruhnya perlu disampaikan dengan jelas kepada pihak terkait. Situasi tersebut juga ditemukan di **Dinas Komunikasi, Informatika, dan Statistik Kabupaten Bandung Barat** (Diskominfotik Kabupaten Bandung Barat). Berdasarkan hasil wawancara yang dilakukan di instansi ini, diketahui bahwa pengelolaan kegiatan rapat belum diatur melalui prosedur operasional standar yang tertulis, serta belum didukung oleh pencatatan yang memuat keterangan operator yang bertugas maupun kebutuhan peralatan yang harus dipersiapkan. Kondisi tersebut menyebabkan penyampaian informasi jadwal, termasuk penugasan operator, masih dilakukan secara lisan, sedangkan proses peminjaman dan pengembalian peralatan pendukung rapat tetap tercatat terpisah dari data jadwal kegiatan. Akibatnya, operator masih perlu melakukan konfirmasi ulang untuk memastikan kebenaran informasi yang diterima, dan riwayat pemakaian peralatan juga belum terdokumentasi secara memadai sehingga proses penelusurannya menjadi kurang optimal. Pada kegiatan rapat yang dilaksanakan secara daring, tautan pertemuan Zoom juga masih dibuat secara manual oleh admin, termasuk pemilihan akun Zoom yang digunakan di antara beberapa akun yang tersedia.

Penelitian terdahulu menghasilkan aplikasi sistem informasi penjadwalan rapat berbasis web pada Kantor DPRD Provinsi Sumatera Utara, dilengkapi dengan fitur penjadwalan, pencatatan absensi, serta dokumentasi hasil rapat, dengan tujuan mengurangi konflik jadwal, meningkatkan transparansi informasi, dan meminimalkan kesalahan pencatatan kegiatan rapat [RAM25]. Melalui penelitian tersebut, diketahui bahwa penyajian informasi jadwal yang jelas berperan penting dalam mendukung pelaksanaan rapat, tetapi kebutuhan berupa penugasan operator dalam kegiatan rapat maupun pencatatan proses peminjaman dan pengembalian peralatan pendukung rapat belum dijangkau. Meskipun sistem tersebut telah menyediakan mekanisme notifikasi rapat kepada anggota, penyampaian informasi jadwal secara otomatis melalui media komunikasi seperti WhatsApp belum menjadi bagian dari penelitian tersebut.

Kesenjangan tersebut menunjukkan perlunya penyampaian informasi jadwal dan penugasan operator yang lebih jelas, serta pencatatan peminjaman dan pengembalian peralatan yang lebih

terdokumentasi, khususnya bagi instansi yang menangani penugasan operator lintas kegiatan dengan sumber daya peralatan yang terbatas seperti Diskominfotik Kabupaten Bandung Barat. Penelitian ini diharapkan memberi manfaat praktis berupa alat bantu yang dapat digunakan langsung oleh instansi terkait, serta manfaat akademis berupa penerapan penjadwalan rapat yang mengaitkan penugasan operator dan pencatatan peminjaman peralatan dalam satu aplikasi, sebagai pelengkap penelitian sebelumnya yang belum membahas kedua aspek tersebut secara terintegrasi.

Berdasarkan uraian tersebut, penelitian ini dilakukan untuk membangun aplikasi penjadwalan rapat berbasis web yang menyajikan informasi jadwal rapat, perubahan jadwal, dan penugasan operator secara lebih jelas kepada pihak terkait, dengan peran pengguna yang dibahas dalam penelitian ini terbatas pada Admin dan Operator. Pencatatan proses peminjaman dan pengembalian peralatan turut didukung, dengan pengajuan peminjaman yang dapat dikaitkan dengan jadwal rapat tertentu apabila peralatan yang dipinjam digunakan untuk mendukung pelaksanaan kegiatan tersebut, meskipun modul peminjaman tetap dipertahankan sebagai modul tersendiri sehingga tidak seluruh pengajuan peminjaman harus terhubung dengan jadwal rapat. Pembahasan mengenai inventaris dalam penelitian ini dibatasi pada proses peminjaman dan pengembalian peralatan sehingga tidak mencakup pengelolaan inventaris secara menyeluruh. Sebagai upaya mendukung penyampaian informasi secara lebih cepat, aplikasi ini juga dilengkapi dengan notifikasi WhatsApp otomatis untuk jadwal rapat baru, perubahan jadwal, dan pembatalan jadwal kepada operator, serta notifikasi pengajuan dan pembatalan peminjaman peralatan kepada bagian inventaris. Untuk kegiatan rapat yang menggunakan platform Online atau Hybrid, disediakan pula fitur pembuatan tautan pertemuan Zoom secara otomatis melalui API pada saat admin menyusun jadwal, dengan opsi bagi admin untuk tetap mengisi tautan secara manual apabila diperlukan. Selain itu, disediakan pula fitur lupa password menggunakan kode OTP yang dikirimkan melalui WhatsApp untuk mendukung pemulihan akses akun pengguna secara mandiri.

1.2 Identifikasi Masalah

Berdasarkan latar belakang yang telah diuraikan sebelumnya, beberapa permasalahan dapat diidentifikasi sebagai berikut:

1. Bagaimana penyampaian informasi jadwal rapat, perubahan jadwal, dan penugasan operator dapat dilakukan agar informasi yang diterima pihak terkait lebih jelas?
2. Bagaimana proses peminjaman dan pengembalian peralatan dapat dilakukan agar penggunaan peralatan dapat terdokumentasi dengan baik untuk mendukung kelancaran rapat?
3. Bagaimana penyampaian informasi jadwal kepada operator melalui WhatsApp secara otomatis dapat mendukung pelaksanaan kegiatan rapat?

1.3 Tujuan Tugas Akhir

Adapun tujuan dari tugas akhir ini adalah:

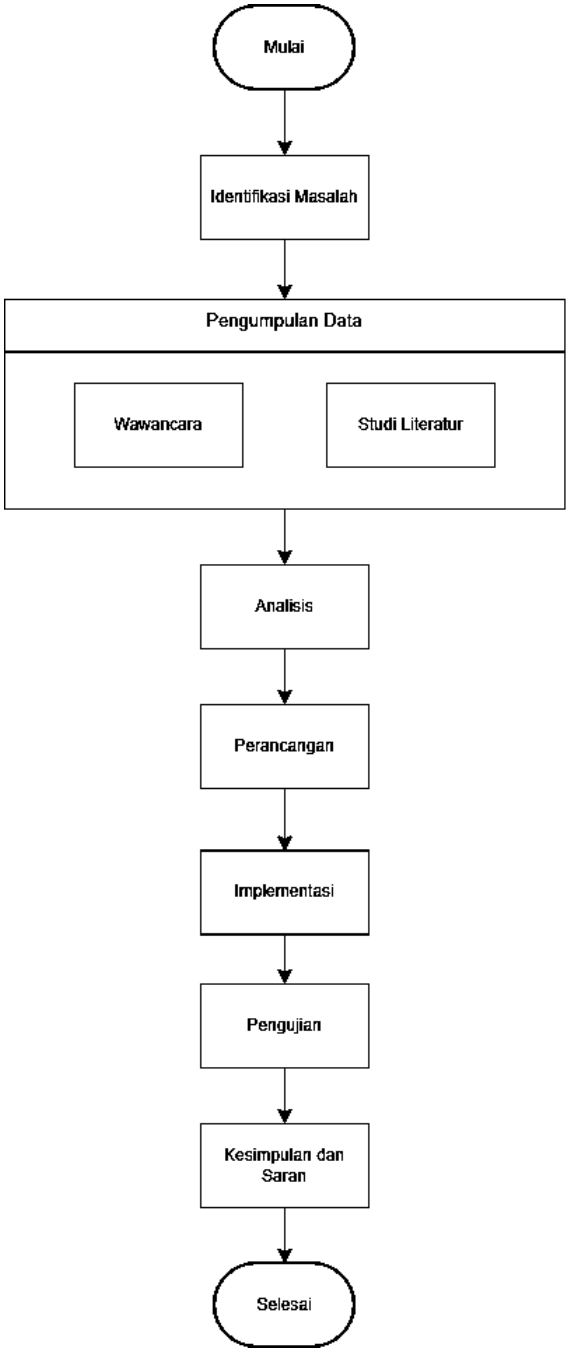
1. Membangun aplikasi penjadwalan rapat berbasis web yang dapat menyajikan informasi jadwal rapat, perubahan jadwal, dan penugasan operator kepada pihak terkait.
2. Mendukung proses peminjaman dan pengembalian peralatan agar penggunaan peralatan dapat terdokumentasi dengan baik dalam mendukung kelancaran rapat.
3. Menyediakan penyampaian informasi jadwal dan perubahan jadwal kepada operator melalui WhatsApp secara otomatis untuk mendukung pelaksanaan kegiatan rapat.

1.4 Lingkup Tugas Akhir

Adapun lingkup atau batasan pengerjaan Tugas Akhir ini adalah sebagai berikut:

1. Penelitian Tugas Akhir dilakukan pada Seksi Infrastruktur dan Teknologi (Jaringan) Dinas Komunikasi, Informatika, dan Statistik Kabupaten Bandung Barat dengan membangun aplikasi penjadwalan rapat berbasis web.
2. Pembuatan akun pengguna hanya dapat dilakukan oleh admin.
3. Pembahasan inventaris dibatasi pada proses peminjaman dan pengembalian peralatan.
4. Satu operator menangani satu kegiatan rapat dalam satu waktu, sedangkan satu kegiatan rapat dapat melibatkan lebih dari satu operator.
5. Pengelolaan data peserta rapat tidak menjadi fokus penelitian.
6. Notifikasi jadwal rapat, perubahan jadwal, pembatalan jadwal, serta pengajuan dan pembatalan peminjaman peralatan dikirimkan melalui WhatsApp secara otomatis.
7. Pemulihan akses akun tersedia melalui kode OTP yang dikirimkan melalui WhatsApp.
8. Bahasa pemrograman yang digunakan adalah PHP.
9. Framework yang digunakan adalah Laravel.
10. Metode pembangunan perangkat lunak yang digunakan adalah metode Waterfall yang dibatasi sampai pada tahap pengujian.

1.5 Metodologi Tugas Akhir



Gambar 1.1 Metodologi Dalam Tugas Akhir

Berikut merupakan rincian dari metodologi tugas akhir ini, diantaranya:

1. Identifikasi Masalah

   Tahap awal dilakukan dengan mengidentifikasi permasalahan pada proses penjadwalan rapat sebagai dasar dalam penentuan solusi yang akan dibangun.

2. Pengumpulan Data

   Data dikumpulkan melalui wawancara dan studi literatur yang relevan dengan topik penelitian.

3. Analisis

   Analisis kebutuhan aplikasi dilakukan berdasarkan kebutuhan pengguna dan fungsi yang diperlukan.
4. Perancangan

   Tahap perancangan dilakukan dengan menyusun alur aplikasi, perancangan data, serta desain antarmuka yang akan dibangun.
5. Implementasi

   Hasil perancangan kemudian diimplementasikan ke dalam bentuk aplikasi berbasis web.
6. Pengujian

   Pengujian dilakukan untuk memastikan bahwa aplikasi yang dibangun berjalan sesuai dengan kebutuhan dan fungsi yang telah ditentukan.
7. Kesimpulan dan Saran

   Tahap akhir dilakukan dengan penarikan kesimpulan dari hasil penelitian dan pemberian saran untuk penelitian selanjutnya.

1.6 Sistematika Penulisan Tugas Akhir

Buku Tugas Akhir ini disusun dengan sistematika penulisan sebagai berikut:

BAB 1 : PENDAHULUAN

Bab ini berisi gambaran umum penelitian yang meliputi latar belakang, identifikasi masalah, tujuan tugas akhir, lingkup tugas akhir, metodologi tugas akhir, serta sistematika penulisan.

BAB 2 : LANDASAN TEORI

Bab ini berisi teori, konsep, dan definisi yang berkaitan dengan topik penelitian sebagai dasar dalam pembangunan aplikasi, serta kajian penelitian terdahulu yang relevan.

BAB 3 : SKEMA PENELITIAN

Bab ini menjelaskan rancangan penelitian, kerangka pemikiran, serta analisis awal terhadap aplikasi yang akan dibangun pada tempat penelitian.

BAB 4 : ANALISIS DAN PERANCANGAN

Bab ini memuat analisis kebutuhan serta perancangan aplikasi yang akan dibangun berdasarkan hasil analisis.

BAB 5 : IMPLEMENTASI DAN PENGUJIAN

Bab ini menjelaskan proses implementasi aplikasi serta pengujian untuk memastikan aplikasi berjalan sesuai kebutuhan.

BAB 6 : KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dari hasil penelitian serta saran untuk penelitian selanjutnya.

BAB 2
LANDASAN TEORI

2.1 Penjadwalan

Berdasarkan **Kamus Besar Bahasa Indonesia** (KBBI), penjadwalan adalah proses, cara, atau perbuatan menjadwalkan. Jadwal merupakan pembagian waktu berdasarkan rencana pengaturan urutan kegiatan dengan waktu pelaksanaan yang telah ditentukan [KBBI].

Di lingkungan organisasi maupun instansi, penjadwalan berperan mengatur waktu pelaksanaan suatu aktivitas agar berjalan sesuai dengan rencana yang telah disusun. Penjadwalan turut membantu menyusun urutan pelaksanaan serta mencegah terjadinya benturan waktu antaraktivitas.

Di samping itu, penjadwalan juga berfungsi sebagai acuan dalam pelaksanaan suatu kegiatan, termasuk dalam menentukan kebutuhan yang diperlukan pada setiap pelaksanaannya.

Prinsip penjadwalan tersebut diterapkan sebagai dasar pengaturan waktu kegiatan rapat, penentuan kebutuhan peralatan, serta penugasan operator pada aplikasi yang dibangun dalam penelitian ini.

2.1.1 Penjadwalan Rapat

Rapat merupakan salah satu bentuk kegiatan komunikasi dalam organisasi atau instansi yang digunakan untuk membahas suatu permasalahan atau kegiatan tertentu. Berdasarkan Kamus Besar Bahasa Indonesia (KBBI), rapat adalah pertemuan atau kumpulan untuk membicarakan sesuatu, sidang, atau majelis [KBBI].

Rapat berfungsi sebagai sarana komunikasi untuk menyampaikan informasi, membahas permasalahan, dan mengambil keputusan bersama, sehingga pelaksanaannya memerlukan pengaturan waktu agar berjalan lancar dan sesuai rencana.

Penjadwalan rapat menjadi aspek penting karena berkaitan dengan penyampaian jadwal kepada operator serta kebutuhan peralatan yang digunakan dalam pelaksanaannya. Berdasarkan hasil wawancara, keterangan jadwal yang belum lengkap dan sering disampaikan secara terpisah menyebabkan operator perlu melakukan konfirmasi tambahan sebelum kegiatan berlangsung.

Selain itu, perubahan jadwal yang tidak tercatat dengan baik dapat menyebabkan keterlambatan dalam persiapan sebelum rapat berlangsung. Kondisi tersebut menunjukkan bahwa penjadwalan rapat perlu didukung penyajian informasi yang jelas agar dapat membantu kelancaran pelaksanaannya.

2.2 Inventaris

Inventaris berkaitan dengan pengelolaan, pengawasan, dan pengendalian barang atau aset dalam suatu organisasi atau instansi. Pengelolaan inventaris bertujuan untuk memastikan ketersediaan barang yang dibutuhkan dalam mendukung kegiatan operasional. Inventaris merupakan barang yang memiliki nilai ekonomi, nilai komersial, atau nilai tukar yang dimiliki oleh badan usaha, instansi, maupun individu [NUG25].

Inventaris pada suatu instansi berkaitan erat dengan ketersediaan sarana dan prasarana yang dimanfaatkan untuk mendukung pelaksanaan tugas. Setiap inventaris perlu dicatat dan diketahui pemakaiannya agar dapat difungsikan sesuai dengan kebutuhan.

Pada instansi pemerintahan, inventaris yang dipakai dalam kegiatan rapat umumnya berupa peralatan pendukung seperti perangkat presentasi, audio, dan fasilitas lainnya. Ketersediaan serta pemakaian peralatan tersebut perlu diperhatikan agar rapat dapat berlangsung sesuai dengan rencana.

Pencatatan proses peminjaman dan pengembalian peralatan yang dikaitkan dengan jadwal rapat pada aplikasi Pintek turut merujuk pada pemahaman inventaris tersebut, khususnya dalam menelusuri pemakaian serta mendukung penyusunan laporan penggunaan peralatan.

2.3 Aplikasi

Aplikasi merupakan perangkat lunak yang digunakan untuk membantu pengguna dalam menjalankan tugas tertentu pada komputer. Aplikasi dirancang untuk mempermudah pekerjaan pengguna dalam mengolah data serta menghasilkan informasi sesuai dengan kebutuhan [PAR22].

Aplikasi tidak memiliki bentuk fisik, tetapi dapat dioperasikan pengguna melalui perangkat komputer. Perangkat lunak ini bekerja dengan mengolah data masukan menjadi keluaran berupa informasi yang dibutuhkan.

Pada suatu instansi, aplikasi berperan membantu penyajian informasi serta pencatatan data secara terstruktur. Dalam pelaksanaan rapat, penyajian jadwal serta pencatatan peminjaman peralatan yang dipakai selama kegiatan berlangsung turut didukung oleh sistem yang dikembangkan.

2.4 Aplikasi Berbasis *Web*

Aplikasi berbasis web merupakan program komputer yang memanfaatkan web browser serta teknologi web untuk menjalankan berbagai fungsi melalui jaringan internet. Aplikasi ini menggunakan teknologi pada sisi server, seperti PHP, untuk mengelola penyimpanan dan pengolahan data, serta teknologi pada sisi pengguna, seperti HTML dan JavaScript, untuk menyampaikan informasi dan mendukung interaksi pengguna [SUR19].

Melalui aplikasi berbasis web, berbagai aktivitas seperti pengisian data, pengelolaan informasi, serta interaksi dapat dilakukan pengguna melalui fitur yang disediakan. Kegiatan berbagi

informasi dan kolaborasi turut didukung oleh aplikasi ini, sehingga data dapat diakses dan dikelola dari berbagai perangkat serta lokasi yang berbeda selama terhubung dengan jaringan internet.

Secara umum, proses kerja aplikasi berbasis web dimulai dari permintaan pengguna melalui web browser yang dikirimkan ke server. Permintaan tersebut kemudian diolah melalui proses pengambilan atau penyimpanan data, dan hasilnya dikembalikan kepada pengguna dalam bentuk informasi yang ditampilkan melalui browser yang sama.

Karakteristik aplikasi berbasis web yang dapat diakses dari berbagai perangkat dan lokasi tersebut menjadi pertimbangan dalam mendukung pengelolaan jadwal rapat serta peminjaman peralatan secara terpusat pada sistem yang dikembangkan.

2.5 *Framework*

Framework merupakan kumpulan kode program yang memiliki aturan tertentu dan digunakan untuk membantu proses pembangunan aplikasi agar lebih cepat. *Framework* dibuat karena adanya kebutuhan untuk menghindari penulisan kode yang sama secara berulang pada setiap proyek. Dengan adanya *framework*, tidak perlu lagi membuat fungsi dari awal karena sebagian besar sudah disediakan dalam bentuk kode dasar yang dapat digunakan kembali [SAB24].

Dalam pembuatan aplikasi, sering ditemukan fitur yang serupa, seperti pembuatan akun pengguna dan login. Jika fitur tersebut selalu dibuat dari awal, tentu akan memakan waktu. Oleh karena itu, *framework* digunakan sebagai solusi agar proses pembangunan aplikasi menjadi lebih efisien. Selain mempercepat pekerjaan, *framework* juga membantu menghasilkan kode yang lebih rapi.

Tujuan utama penggunaan *framework* adalah untuk mempermudah dan mempercepat pembangunan aplikasi karena telah tersedia berbagai fitur yang siap digunakan. Pengguna hanya perlu memanfaatkan fitur tersebut tanpa harus membuat semuanya dari awal. Selain itu, *framework* juga memiliki aturan penulisan tertentu sehingga kode program menjadi lebih mudah dipahami.

Beberapa manfaat penggunaan *framework* antara lain sebagai berikut:

1. Tersedia fitur siap pakai. *Framework* menyediakan berbagai komponen yang dapat langsung digunakan, seperti pembuatan form, validasi input, pengelolaan database, hingga pengaturan tampilan. Hal ini membuat proses pembangunan aplikasi menjadi lebih cepat dibandingkan membuat semuanya secara manual.
2. Mengikuti standar penulisan. *Framework* memiliki aturan penulisan yang telah ditentukan, sehingga membantu dalam menghasilkan kode yang lebih rapi. Salah satu konsep yang umum digunakan adalah ***Model View Controller*** (MVC), yang memisahkan bagian data, tampilan, dan logika sistem agar lebih mudah dikelola.
3. Memudahkan kerja tim. Dengan adanya aturan yang baku, setiap anggota tim dapat memahami alur program dengan lebih mudah. Pembagian tugas juga menjadi lebih jelas,

misalnya bagian tampilan dikerjakan oleh *front-end*, sedangkan logika sistem dikerjakan oleh *back-end*.

4. Memudahkan pembacaan dan pemeliharaan kode. Struktur kode yang jelas membuat program lebih mudah dipahami, terutama jika terjadi pergantian anggota tim. Programmer baru dapat lebih cepat menyesuaikan diri karena mengikuti pola yang sama.
5. Meningkatkan keamanan aplikasi. *Framework* umumnya telah menyediakan fitur keamanan, salah satunya perlindungan terhadap serangan CSRF (*Cross-Site Request Forgery*), sehingga aplikasi menjadi lebih aman tanpa perlu membuat sistem keamanan dari awal.

Dengan demikian, penggunaan framework dapat membantu dalam pembangunan aplikasi yang lebih cepat, rapi, serta lebih mudah untuk dikelola dan dipelihara. Prinsip kerja tersebut menjadi pertimbangan dalam penelitian ini, khususnya melalui pemanfaatan framework Laravel untuk membangun aplikasi penjadwalan rapat. Struktur MVC yang disediakan Laravel membantu memisahkan pengelolaan data, tampilan, serta logika penugasan operator dan peminjaman peralatan, sehingga proses pengembangan aplikasi dapat dilakukan secara lebih terorganisasi.

2.6 Laravel

Laravel merupakan *framework* web berbasis PHP yang bersifat open source dan dikembangkan oleh Taylor Otwell. *Framework* ini digunakan untuk pembangunan aplikasi berbasis web dan saat ini menjadi salah satu *framework* PHP yang banyak digunakan. Hal tersebut disebabkan karena kemudahan dalam penggunaannya, serta didukung oleh dokumentasi yang jelas dan berbagai library yang tersedia [SAB24].

Laravel dirancang untuk membantu dalam proses pembangunan aplikasi agar lebih cepat dan efisien. Berbagai fitur yang telah disediakan dapat langsung digunakan sehingga tidak perlu membuat semua fungsi dari awal. Selain itu, penggunaan Laravel juga membuat kode program lebih mudah dikelola dan dipahami.

Kepopuleran Laravel juga dapat dilihat dari kebutuhan industri, di mana banyak perusahaan yang menggunakan Laravel dalam pembangunan aplikasi web. Hal ini menunjukkan bahwa Laravel memiliki peran penting dalam dunia pengembangan perangkat lunak berbasis web.

Beberapa manfaat penggunaan Laravel antara lain sebagai berikut:

1. Mempercepat pembangunan aplikasi, karena banyak fitur yang telah tersedia dan dapat langsung digunakan.
2. Memudahkan dalam pengelolaan dan pemeliharaan sistem, sehingga aplikasi lebih mudah dikembangkan kembali.
3. Aplikasi lebih stabil dan andal, karena Laravel telah banyak digunakan dan teruji.

4. Memudahkan dalam membaca kode program, sehingga kesalahan dapat lebih cepat ditemukan.
5. Meningkatkan keamanan aplikasi, karena Laravel telah menyediakan fitur untuk mengantisipasi berbagai celah keamanan.
6. Memudahkan pembuatan dokumentasi aplikasi, sehingga sistem lebih mudah dipahami.
7. Didukung dokumentasi yang lengkap, sehingga memudahkan dalam proses pembelajaran dan penggunaan.

Laravel juga menerapkan konsep MVC dalam pengelolaan sistem, sehingga pembagian antara data, tampilan, dan logika aplikasi dapat dilakukan dengan lebih jelas. Fitur bawaan Laravel, seperti Eloquent ORM untuk pengelolaan relasi antar data serta sistem *routing* yang tersusun rapi, dimanfaatkan dalam penelitian ini untuk mengelola relasi antara jadwal rapat, penugasan operator, serta data peminjaman peralatan pada aplikasi yang dibangun.

2.7 Basis Data

Secara etimologis, basis data tersusun dari dua unsur kata, yaitu basis dan data. Kata basis merujuk pada tempat berkumpul, markas, atau gudang, sedangkan data merujuk pada representasi fakta dunia nyata yang mewakili suatu objek, seperti manusia, peristiwa, benda, maupun keadaan, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasi dari bentuk-bentuk tersebut [HAS20].

Istilah basis data memiliki makna implisit yang lebih khusus, yang tercermin dari beberapa karakteristik berikut:

a. Basis data dipahami sebagai kumpulan data yang secara logis memiliki keterkaitan makna, sehingga data yang terkumpul secara acak tanpa keterkaitan tersebut tidak dapat dikategorikan sebagai basis data.
b. Basis data dirancang agar dapat digunakan oleh banyak pemakai maupun oleh berbagai aplikasi sesuai dengan kepentingan masing-masing pemakai.
c. Basis data merupakan representasi dari suatu aspek pada dunia nyata (real world atau miniworld), sebagaimana tampak pada basis data perbankan, perpustakaan, pertanahan, maupun perpajakan.
d. Basis data perlu melalui tahap perancangan dan pembangunan, serta data di dalamnya dikumpulkan untuk mendukung tujuan tertentu.

Berdasarkan keempat karakteristik tersebut, dapat disimpulkan bahwa sistem basis data merupakan sistem terkomputerisasi yang bertujuan memelihara data yang telah diolah agar proses pengaksesannya dapat berlangsung lebih cepat ketika dibutuhkan. Basis data, dengan demikian, berfungsi sebagai media penyimpanan yang memungkinkan data diakses secara mudah dan cepat. Kebutuhan basis data dalam suatu sistem informasi umumnya mencakup proses memasukkan data,

menyimpan data, mengambil kembali data yang telah tersimpan, serta menyusun laporan berdasarkan data tersebut.

Konsep basis data tersebut menjadi salah satu landasan dalam penelitian ini, khususnya dalam merancang struktur penyimpanan data pada aplikasi penjadwalan rapat yang dibangun. Data jadwal kegiatan, penugasan operator, serta pencatatan peminjaman dan pengembalian peralatan perlu disimpan secara terstruktur dan saling berelasi agar dapat diakses dan diolah sesuai kebutuhan sistem, sebagaimana karakteristik basis data yang telah diuraikan sebelumnya.

2.8 MySQL

MySQL merupakan salah satu ***database management system*** (DBMS) yang cukup dikenal luas, menggunakan ***Structured Query Language*** (SQL) sebagai bahasa dasar dalam pengoperasiannya, serta bersifat multi pengguna dan multi thread [SRI21]. MySQL bersifat open source dengan lisensi GNU ***General Public License*** (GPL), sehingga dapat digunakan untuk keperluan pribadi maupun komersial tanpa dikenakan biaya. Sebagai salah satu jenis ***Relational Database Management System*** (RDBMS), MySQL menggunakan istilah baris, kolom, dan tabel dalam pengelolaan datanya, dengan SQL berperan sebagai bahasa penghubung antara aplikasi dan database server.

Beberapa kelebihan MySQL yang relevan dalam pengembangan perangkat lunak antara lain:

a. Mendukung integrasi dengan berbagai bahasa pemrograman serta penggunaan oleh banyak pengguna (*multi user*) secara bersamaan tanpa mengganggu kinerja sistem.
b. Tidak membutuhkan kapasitas RAM yang besar, sehingga dapat dipasang pada server dengan spesifikasi terbatas.
c. Memiliki struktur tabel yang fleksibel serta mendukung berbagai tipe data, seperti integer, *date, char,* dan *text*.
d. Dilengkapi fitur keamanan seperti pengaturan hak akses pengguna dan enkripsi *password,* serta didukung dokumentasi dan komunitas pengguna yang luas.

Di sisi lain, MySQL memiliki keterbatasan dalam pengelolaan database berskala besar serta kurang optimal digunakan pada aplikasi *game* atau *mobile*, karena dirancang agar tetap ringan pada perangkat dengan spesifikasi rendah.

Basis data relasional dibutuhkan dalam penelitian ini untuk menyimpan data pengguna, jadwal kegiatan, penugasan operator, serta pencatatan peminjaman dan pengembalian peralatan yang saling berelasi satu sama lain. MySQL dipilih sebagai *database server* pada aplikasi penjadwalan rapat berbasis Laravel yang dibangun, mengingat dukungannya terhadap konsep RDBMS, kemudahan integrasi dengan Laravel, serta sifatnya yang *open source*.

2.9 Model View Controller (MVC)

Model View Controller (MVC) merupakan suatu konsep arsitektur yang digunakan untuk membagi pemrograman ke dalam beberapa bagian yang lebih spesifik. Dengan adanya pembagian ini, setiap komponen dapat diubah tanpa memengaruhi komponen lainnya. Selain itu, perubahan fungsi dalam aplikasi tidak akan berdampak langsung pada bagian antarmuka [SAB24].

MVC membagi sistem ke dalam tiga bagian utama, yaitu Model, View, dan Controller. Model berfungsi untuk mengelola data dalam aplikasi, View digunakan untuk menampilkan antarmuka kepada pengguna, sedangkan Controller berperan dalam mengatur logika aplikasi serta menjadi penghubung antara Model dan View.

Dengan penerapan konsep MVC, pembagian tugas dalam sistem menjadi lebih jelas antara pengolahan data, tampilan, dan logika proses, sehingga memudahkan dalam pembangunan dan pemeliharaan aplikasi.

2.10 UML

Unified Modeling Language (UML) merupakan bahasa pemodelan visual yang digunakan untuk menspesifikasi, memvisualisasikan, membangun, dan mendokumentasikan sistem perangkat lunak [SWA24].

UML dikembangkan pada tahun 1990-an oleh Grady Booch, James Rumbaugh, dan Ivar Jacobson. Ketiganya menggabungkan pendekatan pemodelan berorientasi objek yang sebelumnya dikembangkan secara terpisah, seperti ***Object Oriented Design*** (OOD), ***Object Modeling Technique*** (OMT), dan ***Object Oriented Software Engineering*** (OOSE). Penggabungan tersebut menghasilkan metode terpadu yang kemudian berkembang menjadi *Unified Modeling Language* (UML) sebagai standar pemodelan sistem perangkat lunak.

Dalam perkembangannya, UML menjadi standar yang digunakan secara luas dalam industri perangkat lunak dan didukung oleh berbagai perusahaan teknologi. UML digunakan untuk menggambarkan sistem melalui berbagai jenis diagram yang terbagi ke dalam dua kelompok utama, yaitu *Structure Diagram* dan *Behavior Diagram*. Structure Diagram digunakan untuk menggambarkan struktur statis sistem, sedangkan *Behavior Diagram* digunakan untuk menggambarkan perilaku serta interaksi dalam sistem.

UML berperan sebagai alat bantu dalam perancangan perangkat lunak karena menyediakan notasi standar yang dapat digunakan untuk menjelaskan sistem secara terstruktur.

2.11 Application Programming Interface (API)

Application Programming Interface **(API)** dapat dipahami sebagai mekanisme yang memungkinkan dua aplikasi atau lebih, meskipun berjalan pada platform yang berbeda, untuk saling berkomunikasi satu sama lain [TIN24].

Proses komunikasi antar aplikasi tersebut berlangsung melalui pengiriman permintaan (request) dari satu sistem kepada sistem lain, yang kemudian direspons (response) oleh sistem penerima permintaan. Komunikasi ini umumnya dilakukan menggunakan protokol HTTP agar data dapat dikirimkan melalui jaringan internet, dengan format ***Javascript Object Notation*** (JSON) digunakan sebagai format pertukaran data karena sifatnya yang ringan serta mudah dipahami, baik oleh manusia maupun oleh mesin.

Dua jenis API yang umum digunakan saat ini adalah RESTful API dan GraphQL. RESTful API mengikuti prinsip ***Representational State Transfer*** (REST), yang merepresentasikan setiap permintaan sebagai operasi tertentu, seperti pengambilan data (GET), penambahan data (POST), pembaruan data (PUT), atau penghapusan data (DELETE). GraphQL, di sisi lain, memungkinkan data diambil secara lebih spesifik sesuai dengan kebutuhan pengguna.

Beberapa manfaat penggunaan API dalam pengembangan perangkat lunak antara lain:

a. Integrasi aplikasi, yaitu kemampuan untuk menghubungkan berbagai platform aplikasi sehingga data dapat dipertukarkan tanpa perlu membangun kembali fitur yang telah tersedia.
b. Efisiensi pengembangan, karena pengembang tidak perlu membangun setiap komponen dari awal dan dapat memanfaatkan komponen yang telah disediakan oleh API, sehingga proses pengembangan aplikasi menjadi lebih cepat.
c. Kolaborasi antar platform, karena API memungkinkan aplikasi diakses melalui web, perangkat seluler, maupun desktop secara bersamaan dengan informasi yang konsisten.

Kebutuhan integrasi semacam ini juga ditemukan pada sistem yang dibangun dalam penelitian ini. Aplikasi penjadwalan rapat yang dibangun memanfaatkan API pihak ketiga untuk dua kebutuhan utama, yaitu API Zoom untuk membuat tautan rapat secara otomatis ketika jadwal rapat dibuat, serta API Fonnte untuk mengirimkan permintaan HTTP ke layanan WhatsApp sehingga notifikasi terkait jadwal rapat, perubahan jadwal, pembatalan jadwal, maupun pengajuan dan pembatalan peminjaman peralatan dapat disampaikan kepada operator dan bagian inventaris. Pemanfaatan kedua API tersebut memungkinkan aplikasi menjalankan fungsi integrasi tanpa perlu membangun infrastruktur pembuatan tautan rapat maupun pengiriman pesan secara mandiri.

2.12 Bahasa Pemrograman

Bahasa pemrograman merupakan seperangkat aturan atau sintaks yang digunakan untuk menyusun instruksi agar komputer dapat menjalankan tugas tertentu, mulai dari pengolahan data

hingga pembuatan perangkat lunak. Bahasa ini berfungsi sebagai media komunikasi antara manusia dan komputer, sehingga logika dan algoritma yang dirancang manusia dapat dipahami dan dieksekusi oleh mesin [AND25].

Pada penelitian ini, bahasa pemrograman digunakan sebagai teknologi utama dalam pembangunan aplikasi penjadwalan rapat berbasis web. Bahasa pemrograman yang digunakan meliputi HTML, CSS, JavaScript, PHP, dan SQL yang masing-masing memiliki peran dalam pengembangan sistem:

2.12.1 HTML

HTML (***Hypertext Markup Language***) merupakan bahasa markup yang digunakan sebagai dasar dalam membangun struktur halaman web. HTML berfungsi untuk menyusun dan mengatur berbagai elemen yang ditampilkan pada halaman, seperti teks, gambar, tabel, formulir, serta komponen lainnya [AND25].

Melalui HTML, pengembang dapat menentukan bagaimana setiap elemen akan ditampilkan dan diorganisasikan sehingga dapat dilihat dengan baik oleh pengguna melalui browser. Dengan demikian, HTML menjadi fondasi utama dalam pengembangan website sebelum ditambahkan gaya (CSS) dan interaktivitas (JavaScript).

2.12.2 CSS

CSS (***Cascading Style Sheets***) merupakan bahasa yang digunakan untuk mengatur tampilan dan desain halaman web. Sebelum mempelajari CSS, penting untuk memahami dasar HTML karena CSS bekerja dengan memodifikasi elemen-elemen yang telah dibuat menggunakan HTML.

CSS berfungsi untuk mengatur berbagai komponen dalam halaman web agar tampil lebih menarik dan terstruktur sesuai kebutuhan. Dalam penggunaannya, CSS memanfaatkan selector seperti id dan class untuk menentukan elemen mana yang akan diberi gaya tertentu. Selain digunakan pada HTML dan XHTML, CSS juga telah berkembang dan dapat diterapkan dalam berbagai platform lain, termasuk dalam desain antarmuka aplikasi [SRI21].

2.12.3 Tailwind CSS

Tailwind CSS merupakan sebuah *framework* CSS bertipe *utility-first* yang digunakan untuk membangun tampilan situs web secara cepat tanpa perlu keluar dari berkas HTML [TWCSS].

Framework ini dilengkapi dengan sekumpulan kelas utilitas (*utility classes*) siap pakai, seperti kelas untuk pengaturan tata letak, jarak antar elemen, maupun perataan teks, yang dapat dikombinasikan langsung pada markah HTML untuk membentuk berbagai jenis

rancangan antarmuka. Pendekatan ini memungkinkan pengembang menyusun tampilan tanpa harus menulis kode CSS secara terpisah pada berkas tersendiri.

Pada proses pembangunan untuk lingkungan produksi, seluruh kelas CSS yang tidak digunakan akan dihapus secara otomatis, sehingga ukuran berkas CSS akhir yang dikirimkan ke klien menjadi jauh lebih kecil. Sejumlah fitur pengembangan antarmuka modern turut didukung oleh Tailwind CSS, antara lain rancangan responsif (*responsive design*) melalui titik jeda (*breakpoint*) ukuran layar, mode gelap (*dark mode*), serta penyesuaian tema melalui variabel CSS.

Kemudahan penyusunan antarmuka melalui kelas utilitas tersebut dimanfaatkan dalam penelitian ini untuk membangun tampilan aplikasi penjadwalan rapat secara lebih efisien, khususnya pada halaman dashboard, formulir jadwal, serta daftar peralatan dan peminjaman, tanpa perlu menulis berkas CSS kustom secara ekstensif.

2.12.4 PHP

PHP (***Hypertext Preprocessor***) merupakan bahasa pemrograman yang dirancang khusus untuk pengembangan aplikasi berbasis web. PHP termasuk dalam kategori *server-side scripting*, yaitu bahasa yang dijalankan di sisi server untuk menghasilkan konten web yang dinamis dan interaktif [AND25].

Beberapa alasan mengapa PHP menjadi pilihan populer di kalangan pengembang web:

1. Mudah Dipelajari, PHP memiliki sintaks yang relatif sederhana sehingga mudah dipahami, terutama bagi pemula yang baru mempelajari pemrograman web.
2. Bersifat Multiplatform, PHP dapat dijalankan pada berbagai sistem operasi seperti Windows, Linux, dan macOS, sehingga fleksibel dalam proses pengembangan maupun implementasi.
3. Mendukung Berbagai Database, PHP memiliki kompatibilitas yang baik dengan berbagai sistem manajemen basis data, seperti MySQL, PostgreSQL, dan SQLite, sehingga memudahkan dalam pengolahan data.
4. Komunitas yang Luas, dukungan komunitas yang besar membuat PHP memiliki banyak dokumentasi, tutorial, serta forum diskusi yang membantu dalam proses pembangunan.
5. Tersedia Beragam *Framework*, PHP didukung oleh berbagai *framework* populer seperti Laravel, CodeIgniter, dan Symfony yang dapat mempercepat proses pengembangan aplikasi serta meningkatkan kualitas kode.

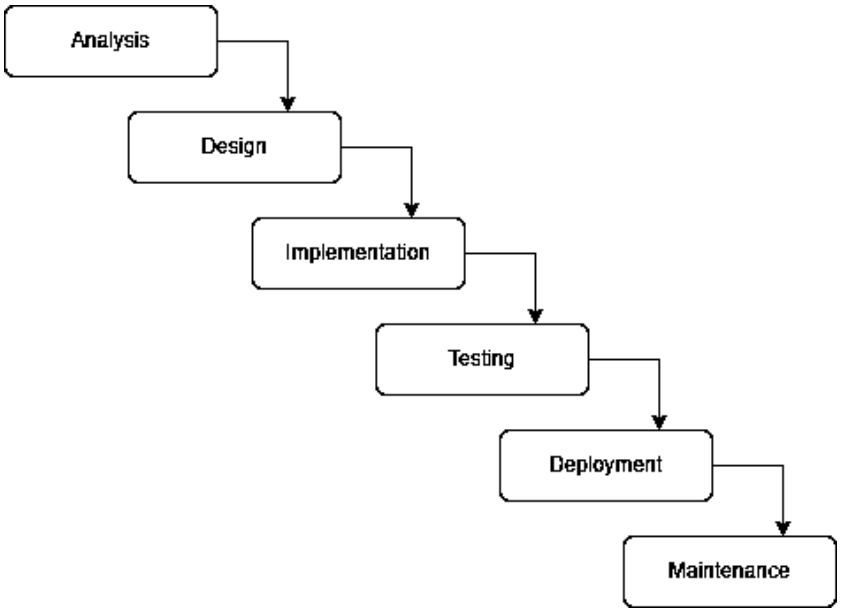
2.12.5 JavaScript

JavaScript merupakan bahasa pemrograman yang digunakan untuk menambahkan interaktivitas pada halaman web. JavaScript memungkinkan halaman web merespon tindakan pengguna seperti klik tombol, pengisian form, serta manipulasi elemen pada halaman web [SRI21].

2.12.6 SQL

SQL (*Structured Query Language*) merupakan bahasa yang digunakan untuk mengelola dan memanipulasi data pada sistem basis data. SQL digunakan untuk melakukan operasi seperti penyimpanan, pengambilan, pengubahan, dan penghapusan data dalam database [SRI21].

2.13 Metode Waterfall



Gambar 2.1 Model Waterfall

Metode Waterfall atau model air terjun merupakan salah satu pendekatan klasik dalam pengembangan perangkat lunak yang menggambarkan proses pengembangan secara linier dan berurutan. Pada model ini, setiap tahapan harus diselesaikan terlebih dahulu sebelum melanjutkan ke tahap berikutnya, sehingga hasil dari suatu tahap akan menjadi masukan bagi tahap selanjutnya. Model Waterfall termasuk dalam **Software Development Life Cycle** (SDLC) yang digunakan untuk menggambarkan proses pembangunan perangkat lunak secara sistematis dari awal hingga tahap pemeliharaan [HAS20].

Tahapan dalam metode Waterfall terdiri dari beberapa fase utama sebagai berikut:

1. *Requirement Gathering and Analysis*, tahap ini merupakan proses pengumpulan dan analisis kebutuhan sistem secara menyeluruh. Informasi diperoleh melalui wawancara dengan pihak

Diskominfotik Kabupaten Bandung Barat, serta studi literatur yang berkaitan dengan aplikasi penjadwalan. Hasil kegiatan tersebut digunakan untuk mengidentifikasi kebutuhan pengguna dan menentukan fungsi-fungsi yang harus tersedia pada aplikasi.

2. *Design*, berdasarkan kebutuhan yang telah teridentifikasi, selanjutnya disusun rancangan aplikasi sebagai acuan pengembangan. Perancangan meliputi pemodelan menggunakan Unified Modeling Language (UML) yang terdiri dari use case diagram, activity diagram, sequence diagram, dan class diagram. Selain itu dilakukan perancangan basis data serta antarmuka pengguna sebagai acuan dalam proses pembangunan aplikasi. Tahapan ini bertujuan menghasilkan gambaran sistem secara menyeluruh sebelum proses pengkodean dilakukan.
3. *Implementation*, tahap implementasi merupakan proses penerjemahan desain ke dalam bentuk kode program. Hasil dari tahap ini berupa modul-modul program yang selanjutnya akan diintegrasikan menjadi aplikasi yang utuh. Pada penelitian ini implementasi dilakukan menggunakan bahasa pemrograman PHP dengan *framework* Laravel serta basis data MySQL.
4. *Integration and Testing*, setelah seluruh modul selesai dikembangkan, proses dilanjutkan dengan penggabungan komponen sistem dan pengujian fungsionalitas. Pengujian dilakukan untuk memastikan bahwa setiap fitur dapat berjalan sesuai rancangan serta mampu menghasilkan keluaran yang diharapkan. Tahap ini juga digunakan untuk menemukan dan memperbaiki kesalahan yang masih muncul pada aplikasi.
5. Verifikasi, sistem yang telah melalui proses pengujian kemudian dievaluasi bersama pengguna. Kegiatan verifikasi bertujuan memastikan bahwa aplikasi yang dibangun telah memenuhi kebutuhan yang disepakati pada tahap analisis serta dapat mendukung proses penjadwalan rapat sesuai kondisi operasional yang berlaku.
6. *Operation and Maintenance*, tahap terakhir merupakan fase penggunaan dan pemeliharaan sistem. Aktivitas yang dilakukan mencakup perbaikan kesalahan yang ditemukan selama penggunaan, penyesuaian terhadap kebutuhan baru, serta pengembangan fitur apabila diperlukan guna menjaga kinerja aplikasi tetap optimal.

Metode Waterfall memiliki keunggulan berupa alur pengembangan yang terstruktur dan sistematis, sehingga memudahkan dalam pengelolaan proyek serta menghasilkan dokumentasi yang jelas pada setiap tahap. Selain itu, kemajuan proyek dapat dipantau dengan lebih mudah dan estimasi waktu maupun biaya dapat direncanakan secara lebih terukur apabila kebutuhan sistem telah ditentukan sejak awal.

Namun, metode ini juga memiliki beberapa keterbatasan, di antaranya kurang fleksibel terhadap perubahan kebutuhan serta membutuhkan waktu pengembangan yang relatif lebih lama karena setiap tahap harus diselesaikan secara berurutan.

Berdasarkan karakteristik tersebut, metode Waterfall dipilih dalam penelitian ini karena sistem yang dikembangkan memiliki kebutuhan yang relatif jelas dan tidak mengalami banyak perubahan selama proses pengembangan. Selain itu, model ini sesuai untuk pengembangan sistem dengan ruang lingkup yang terdefinisi dengan baik karena memiliki alur yang sederhana, terstruktur, dan mudah diterapkan. Dibandingkan dengan metode lain seperti Agile atau Scrum yang bersifat iteratif dan dinamis terhadap perubahan kebutuhan, Waterfall dinilai lebih tepat digunakan dalam penelitian ini karena memberikan kejelasan tahapan serta kemudahan dalam dokumentasi dan evaluasi proses pengembangan.

2.14 Pengujian Perangkat Lunak

Pengujian sistem dapat dipahami sebagai pengujian terhadap program perangkat lunak yang telah lengkap dan terintegrasi secara utuh, mengingat perangkat lunak pada dasarnya hanya merupakan salah satu elemen dari sistem berbasis komputer yang lebih besar dan umumnya saling terhubung dengan perangkat lunak maupun perangkat keras lainnya [HAS20].

Pengujian perangkat lunak merupakan suatu proses atau rangkaian proses yang dirancang untuk memastikan bahwa program komputer berjalan sesuai dengan fungsi yang seharusnya, sekaligus memastikan program tidak menjalankan hal-hal yang tidak diharapkan, mengingat sebuah perangkat lunak seharusnya bersifat dapat diprediksi dan konsisten. Tugas utama seorang penguji difokuskan pada penemuan bug atau error sebanyak mungkin, disertai pemahaman mengenai bagaimana bug atau error tersebut dapat dihasilkan. Dengan demikian, pengujian dapat dimaknai sebagai proses eksekusi program dengan tujuan menemukan error di dalamnya.

Pengujian merupakan proses penting dalam siklus pengembangan perangkat lunak untuk menjamin kualitas perangkat lunak yang dihasilkan. Proses pengujian terdiri atas beberapa aktivitas yang secara keseluruhan disebut sebagai siklus hidup pengujian, yang pada praktiknya dapat dilaksanakan secara formal maupun informal.

Kualitas suatu perangkat lunak perlu dijaga karena berkaitan erat dengan tingkat kepuasan pelanggan. Beberapa alasan pentingnya menjaga kualitas perangkat lunak antara lain:

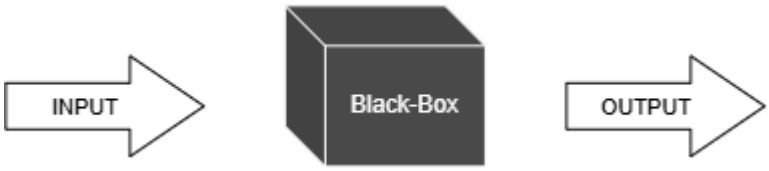
a. Agar perangkat lunak dapat bertahan (survive) di dunia bisnis perangkat lunak.
b. Agar perangkat lunak dapat bersaing dengan perangkat lunak lain.
c. Kualitas menjadi hal penting untuk kebutuhan pemasaran global (global marketing).
d. Menjaga efektivitas biaya agar tidak banyak perangkat lunak yang terbuang akibat kegagalan pemasaran atau kegagalan produksi.
e. Mempertahankan pelanggan serta meningkatkan keuntungan.

Pengujian perangkat lunak sering dikaitkan dengan dua elemen topik, yaitu verifikasi dan validasi. Verifikasi mengacu pada sekumpulan aktivitas yang menjamin bahwa perangkat lunak mengimplementasikan suatu fungsi tertentu secara benar, sedangkan validasi mengacu pada

sekumpulan aktivitas lain yang menjamin bahwa perangkat lunak yang dibangun dapat ditelusuri kesesuaiannya dengan kebutuhan pelanggan.

Tahap pengujian tersebut juga diterapkan pada penelitian ini sebagai bagian akhir dari metode Waterfall yang digunakan, guna memastikan aplikasi penjadwalan rapat yang dibangun berjalan sesuai fungsi yang telah dirancang serta memenuhi kebutuhan Diskominfotik Kabupaten Bandung Barat sebagaimana diuraikan pada tahap analisis kebutuhan.

2.15 Black Box Testing



Gambar 2.2 Ilustrasi Black Box Testing

Menurut [HAS20], Black Box Testing merupakan metode pengujian perangkat lunak yang berfokus pada pengujian fungsional untuk memastikan bahwa setiap fungsi masukan (*input*) dan keluaran (*output*) telah berjalan sesuai dengan spesifikasi yang telah ditetapkan. Pada metode ini, penguji hanya memeriksa apakah fungsi sistem memberikan hasil yang sesuai berdasarkan masukan yang diberikan tanpa mengetahui atau memperhatikan bagaimana proses tersebut diimplementasikan di dalam kode program.

Sebagai contoh, pada pengujian fitur login, penguji dapat membuat beberapa skenario, seperti menguji kondisi ketika pengguna memasukkan username dan password yang benar, serta menguji kondisi ketika salah satu atau kedua data yang dimasukkan tidak sesuai.

Metode Black Box Testing memiliki beberapa kelebihan, yaitu mampu menguji perangkat lunak secara efisien pada aplikasi yang berukuran besar, tidak memerlukan akses terhadap kode sumber (*source code*), serta memberikan sudut pandang yang mewakili pengalaman pengguna sehingga terpisah dari proses pengembangan sistem. Namun, metode ini juga memiliki beberapa keterbatasan, di antaranya cakupan pengujian yang bergantung pada skenario yang dirancang sehingga tidak seluruh kemungkinan kondisi dapat diuji, serta efektivitas pengujian sangat dipengaruhi oleh kemampuan penguji dalam menyusun kasus uji tanpa mengetahui struktur internal perangkat lunak.

Dengan demikian, metode Black Box Testing dinilai sesuai untuk mengevaluasi fungsionalitas aplikasi karena mampu memverifikasi bahwa setiap fitur telah memberikan keluaran sesuai dengan kebutuhan pengguna tanpa memperhatikan implementasi kode program.

2.16 Penelitian Terdahulu

Penelitian-penelitian terdahulu yang relevan menjadi dasar penting dalam tugas akhir.

Berikut adalah ringkasan beberapa penelitian relevan:

Tabel 2.1 Penelitian Terdahulu

No	Peneliti	Judul Penelitian	Hasil Penelitian
1	M Rizky Ramadhan, M Angga Septiawan (2025) [RAM25]	Analisis dan Perancangan Sistem Informasi Penjadwalan Rapat pada Kantor DPRD Provinsi Sumatera Utara dengan Metode Waterfall	Penelitian ini menghasilkan aplikasi sistem informasi penjadwalan rapat berbasis web yang dilengkapi dengan fitur penjadwalan, notifikasi rapat kepada anggota, pencatatan absensi, serta dokumentasi hasil rapat. Sistem yang dikembangkan ditujukan untuk mengurangi konflik jadwal, meningkatkan transparansi informasi, serta meminimalkan kesalahan dalam pencatatan kegiatan rapat. Penerapan fitur penjadwalan dalam aplikasi tersebut menunjukkan pentingnya penyajian informasi jadwal yang jelas dalam mendukung pelaksanaan kegiatan.
2	Sayed Fachrurrazi, Muhammad Ghaffar Shidqi, Arief Rahman (2022) [FAC22]	Rancang Bangun dan Aplikasi Sistem Informasi Penjadwalan Petugas Piket dan Layanan Informasi Berbasis Media Web	Sistem informasi penjadwalan petugas piket dikembangkan dalam bentuk aplikasi berbasis web untuk mengelola jadwal siaran, khususnya jadwal khutbah dan penceramah (imam dan khatib), yang sebelumnya dicatat secara manual menggunakan Microsoft Word. Pengelolaan data jadwal dilakukan melalui aplikasi sehingga data dapat ditambahkan, diperbarui, serta dicetak dalam bentuk Excel maupun PDF. Penggunaan aplikasi berbasis web pada sistem tersebut menunjukkan pemanfaatan teknologi dalam pengelolaan jadwal siaran informasi.
3	Mohammad Agil, Ahmad Homaidi, Iqbal Hamdani (2025) [AGI25]	Perancangan Sistem Penjadwalan Otomasi Petugas Vidiotron di Dinas Komunikasi dan Informatika Situbondo	Perancangan sistem penjadwalan petugas vidiotron dilakukan untuk mengatasi permasalahan penjadwalan yang sebelumnya dilakukan secara manual, seperti benturan jadwal, pembagian tugas yang tidak merata, serta keterlambatan dalam penyampaian informasi. Sistem dirancang berbasis web dengan fitur pengelolaan data petugas, penjadwalan kegiatan, penjadwalan otomatis, pemberitahuan penugasan, serta penyimpanan laporan yang dapat diunduh dalam bentuk dokumen. Pendekatan penjadwalan otomatis dalam sistem tersebut menunjukkan adanya upaya pengaturan jadwal kegiatan secara lebih terkoordinasi.
4	Muhammad Fauzan Rusby Kholiq, Noufal Maulana, Muhammad Dabit Habile Alzidane, Wasis Haryono (2025) [KHO25]	Pengembangan Sistem Informasi Terintegrasi untuk Inventori, Absensi, dan Penjadwalan Kerja di Casa Coffevibes dengan Model Agile	Pengelolaan inventori, absensi, dan penjadwalan kerja yang sebelumnya dilakukan secara manual dikembangkan ke dalam sistem berbasis web yang terintegrasi. Sistem tersebut mencakup pengelolaan stok, pencatatan kehadiran, serta pengaturan jadwal kerja dalam satu aplikasi. Proses pengembangan dilakukan secara bertahap menggunakan metode *Agile*. Integrasi pengelolaan data dalam satu sistem menunjukkan adanya penggabungan beberapa kebutuhan operasional dalam satu aplikasi.
5	Syaif Al Khalim, Agung Handayanto (2024) [KHA24]	Perancangan Aplikasi Penjadwalan Agenda Kegiatan Berbasis Website di BPS Provinsi Jawa Tengah	Aplikasi penjadwalan kegiatan dikembangkan dalam bentuk sistem informasi berbasis web yang digunakan untuk mengelola agenda kegiatan pada instansi. Sistem menyediakan pengelolaan data agenda, notulen, absensi, serta data pendukung lainnya dalam satu

			aplikasi, dengan pembagian akses antara pengguna dan admin. Melalui sistem informasi tersebut, proses penyusunan jadwal dan pemantauan kegiatan dapat dilakukan melalui satu media. Pengelolaan agenda kegiatan dalam sistem informasi tersebut memperlihatkan penggunaan aplikasi berbasis web dalam penyajian informasi jadwal kegiatan.

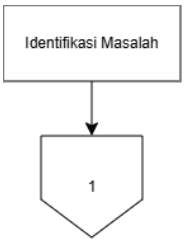
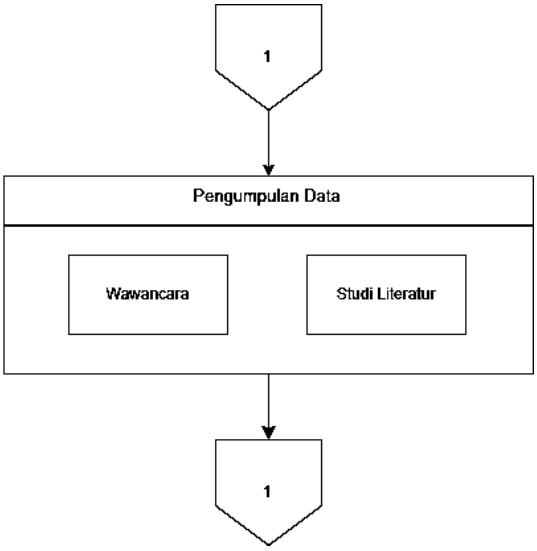
BAB 3
SKEMA PENELITIAN

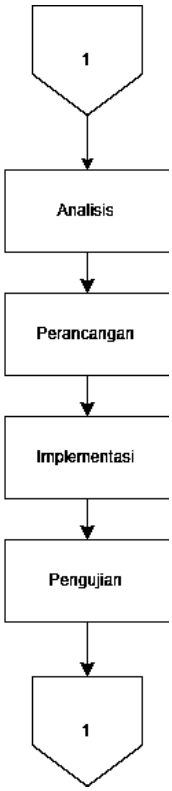
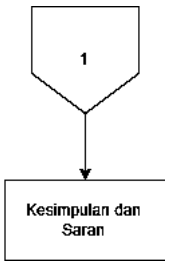
3.1 Alur Penyelesaian Tugas Akhir

Dalam menyelesaikan tugas akhir ini diperlukan suatu alur penelitian yang sistematis agar proses penelitian dapat berjalan secara terarah. Alur penyelesaian tugas akhir digunakan sebagai pedoman dalam melakukan tahapan penelitian mulai dari identifikasi masalah hingga penarikan kesimpulan.

Adapun tahapan penelitian yang dilakukan dalam pembangunan aplikasi penjadwalan rapat berbasis web menggunakan *framework* Laravel pada Diskominfotik Kabupaten Bandung Barat dapat dilihat pada langkah-langkah berikut.

Tabel 3.1 Langkah Penelitian

Tahap dan Hasil	Langkah Penelitian	Literatur dan Referensi
Tahap 1: Identifikasi Masalah Hasil: Masalah yang diteliti Kontribusi: Berguna untuk memberikan gambaran mengenai permasalahan yang terjadi serta menentukan arah penelitian yang akan dilakukan		Wawancara dengan pihak Diskominfotik Kabupaten Bandung Barat
Tahap 2: Pengumpulan Data Hasil: Memahami topik penelitian Kontribusi: Membantu dalam mengidentifikasi kebutuhan aplikasi yang akan digunakan sebagai dasar dalam membangun aplikasi		1. Wawancara dengan pihak Diskominfotik Kabupaten Bandung Barat 2. Konsep penjadwalan dan rapat [KBBI] 3. Konsep inventaris [NUG25] 4. Konsep aplikasi [PAR22] 5. Aplikasi berbasis web [SUR19]

		6. Bahasa pemrograman web [AND25] [SRI21]
Tahap 3: Proses rancang bangun dan *testing* aplikasi berdasarkan kebutuhan sistem yang telah diperoleh pada tahap sebelumnya Hasil: Rancang bangun aplikasi Kontribusi: Menghasilkan aplikasi yang dapat mendukung penjadwalan rapat serta peminjaman peralatan agar data dapat tercatat dengan baik		1. Hasil wawancara di Dinas Komunikasi, Statistik dan Informatika Kabupaten Bandung Barat 2. *Framework* [SAB24] 3. Basis Data [SRI21] 4. MySQL [SRI21] 5. MVC [SAB24] 6. Pemodelan sistem menggunakan UML [SWA24] 7. Laravel [SAB24] 8. Metode pengembangan perangkat lunak Waterfall [HAS20] 9. Testing [HAS20] 10. Black Box Testing [HAS20]
Tahap 4: Menentukan kesimpulan dan saran Hasil: Kesimpulan dan saran Dari sistem yang dibuat Kontribusi: Berguna untuk pengembangan aplikasi di masa mendatang.		

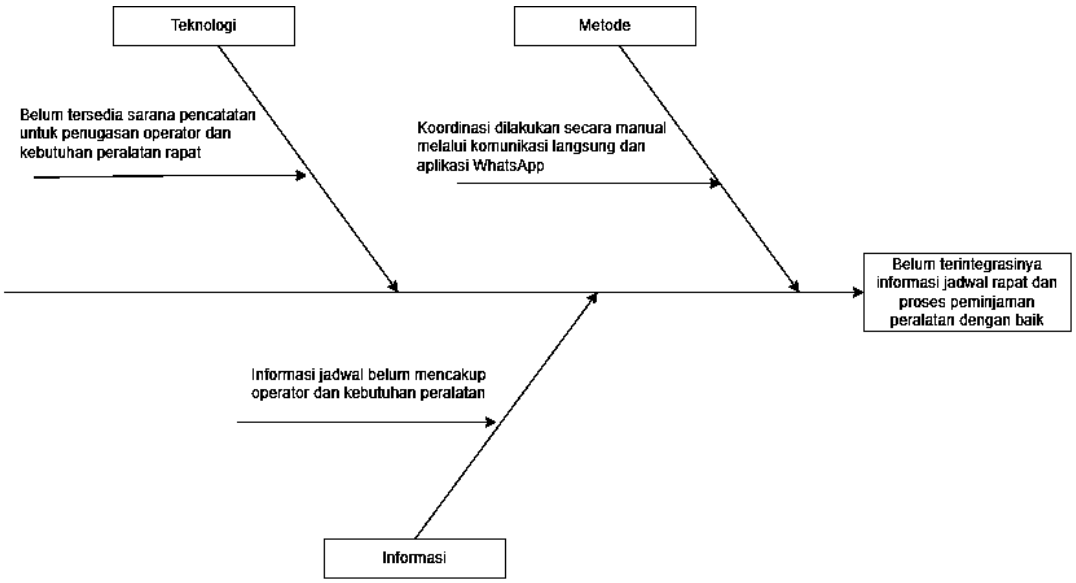
3.2 Perumusan Masalah

Perumusan masalah merupakan tahap untuk mengidentifikasi permasalahan yang terjadi berdasarkan latar belakang penelitian. Perumusan ini bertujuan untuk mengetahui kesesuaian antara permasalahan yang ada dengan solusi yang akan dikembangkan dalam penelitian tugas akhir.

3.2.1 Analisis Sebab Akibat

Analisis sebab akibat merupakan metode yang digunakan untuk mengidentifikasi faktor-faktor penyebab terjadinya suatu permasalahan. Analisis ini dilakukan berdasarkan hasil wawancara kemudian divisualisasikan menggunakan *fishbone* diagram.

Berikut ini merupakan analisis sebab akibat yang muncul pada penelitian tugas akhir:



Gambar 3.1 Fishbone diagram analisis sebab akibat

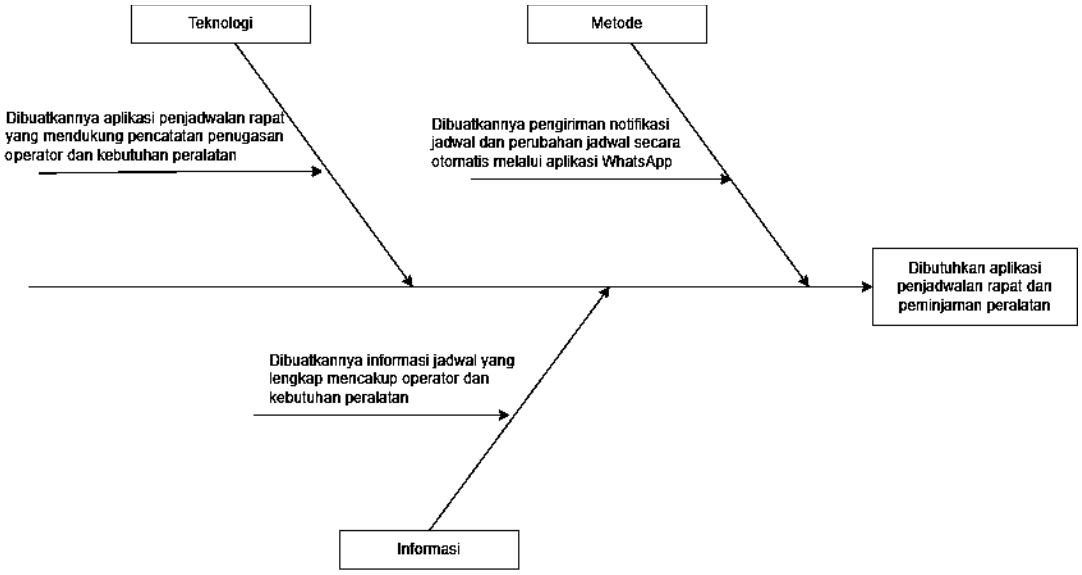
Berdasarkan dari hasil analisis sebab akibat yang sudah digambarkan pada gambar di atas maka dapat dijelaskan pada tabel di bawah ini:

Tabel 3.2 Analisis Sebab Akibat

No.	Masalah	Analisis Sebab akibat
1	Teknologi	Belum tersedia sarana pencatatan untuk penugasan operator dan kebutuhan peralatan rapat sehingga informasi mengenai operator yang bertugas dan peralatan yang harus dipersiapkan belum dapat didokumentasikan dalam satu media. Kondisi tersebut menyebabkan proses penyampaian informasi masih dilakukan secara terpisah
2	Informasi	Informasi jadwal belum mencakup penugasan operator dan kebutuhan peralatan sehingga informasi yang diterima operator belum lengkap. Akibatnya, operator perlu memastikan kembali tugas dan peralatan yang harus dipersiapkan sebelum pelaksanaan rapat
3	Metode	Koordinasi dilakukan secara manual melalui komunikasi langsung dan aplikasi WhatsApp sehingga proses penyampaian informasi jadwal dan perubahan jadwal belum terdokumentasi dengan baik. Kondisi tersebut berpotensi menyebabkan keterlambatan penyampaian informasi kepada pihak yang terkait

3.2.2 Analisis Solusi Masalah

Analisis solusi masalah merupakan bagian dari perumusan masalah yang disusun berdasarkan hasil wawancara dan digambarkan menggunakan *fishbone* diagram. Solusi yang diusulkan merupakan hasil dari permasalahan yang terjadi. Berikut merupakan analisis solusi masalah pada penelitian tugas akhir ini:



Gambar 3.2 Fishbone diagram analisis solusi masalah

Berdasarkan dari analisis solusi masalah yang sudah digambarkan pada gambar di atas maka dapat dijelaskan pada tabel di bawah ini:

Tabel 3.3 Analisis Solusi Masalah

No	Masalah	Analisis Solusi Masalah
1	Teknologi	Dibuatkannya aplikasi penjadwalan rapat yang mendukung pencatatan penugasan operator dan kebutuhan peralatan sehingga informasi yang dibutuhkan dalam pelaksanaan rapat dapat dikelola dalam satu aplikasi. Selain itu, aplikasi juga menyediakan pembuatan tautan Zoom secara otomatis untuk rapat daring sehingga proses pembuatan tautan tidak lagi dilakukan secara manual.
2	Informasi	Dibuatkannya informasi jadwal yang lengkap mencakup penugasan operator, kebutuhan peralatan, waktu, lokasi, serta perubahan jadwal sehingga informasi yang diterima oleh operator menjadi lebih jelas dan mengurangi kebutuhan konfirmasi ulang.
3	Metode	Dibuatkannya pengiriman notifikasi jadwal dan perubahan jadwal secara otomatis melalui aplikasi WhatsApp, serta pencatatan proses peminjaman dan pengembalian peralatan yang terhubung dengan data jadwal rapat sehingga penyampaian informasi menjadi lebih cepat dan proses dokumentasi kegiatan dapat dilakukan dengan lebih baik

3.3 Profil Penelitian

Berikut ini adalah tempat dan profil instansi tempat penelitian:

Nama Instansi	:	Dinas Komunikasi, Informatika, dan Statistik Kabupaten Bandung Barat
Alamat	:	Kompleks Perkantoran Pemerintah Kabupaten Bandung Barat, Jl. Raya Padalarang Cisarua Km. 2, Ngamprah, Kabupaten Bandung Barat, Jawa Barat 40552

Gambar 3.3 Logo Diskominfotik Kabupaten Bandung Barat

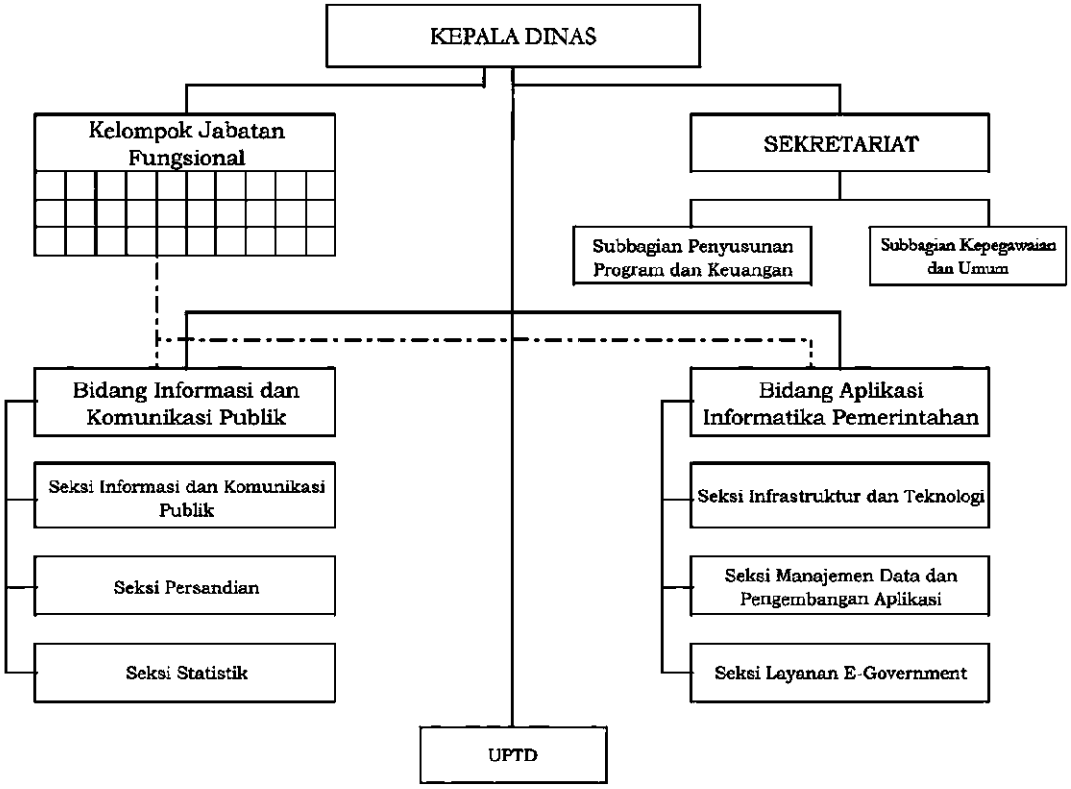
Dinas Komunikasi, Informatika, dan Statistik Kabupaten Bandung Barat (Diskominfotik Bandung Barat) merupakan perangkat daerah yang melaksanakan urusan pemerintahan di bidang komunikasi dan informatika serta statistik sektoral di lingkungan Pemerintah Kabupaten Bandung Barat. Keberadaan perangkat daerah ini berkaitan dengan terbentuknya Kabupaten Bandung Barat sebagai daerah otonom baru berdasarkan Undang-Undang Republik Indonesia Nomor 12 Tahun 2007 tentang Pembentukan Kabupaten Bandung Barat.

Dalam rangka mendukung penyelenggaraan urusan pemerintahan di bidang komunikasi dan informatika, Pemerintah Kabupaten Bandung Barat membentuk perangkat daerah yang menangani bidang tersebut. Landasan hukum organisasi Diskominfotik Bandung Barat diatur dalam Peraturan Bupati Kabupaten Bandung Barat Nomor 62 Tahun 2016 tentang Kedudukan, Susunan Organisasi, Tugas, dan Fungsi serta Tata Kerja Dinas Komunikasi, Informatika, dan Statistik Kabupaten Bandung Barat.

Pembentukan perangkat daerah di Kabupaten Bandung Barat juga mengacu pada Peraturan Daerah Kabupaten Bandung Barat Nomor 9 Tahun 2016 tentang Pembentukan dan Susunan Perangkat Daerah, yang mengatur struktur organisasi perangkat daerah di lingkungan Pemerintah Kabupaten Bandung Barat.

3.3.1 Bagan Struktur Organisasi Instansi

Struktur organisasi pada instansi pemerintah merupakan elemen penting yang memastikan setiap unit dapat menjalankan tugas dan fungsinya secara terarah, terkoordinasi, serta saling mendukung dalam mencapai tujuan organisasi. Dinas Komunikasi, Informatika, dan Statistik Kabupaten Bandung Barat memiliki struktur organisasi yang terdiri atas beberapa bidang dan subbagian yang berperan dalam penyelenggaraan layanan komunikasi publik, pengelolaan teknologi informasi, pengembangan infrastruktur, serta pengelolaan data dan statistik sektoral.



Gambar 3.4 Bagan Struktur Organisasi Instansi

Penelitian Tugas Akhir ini dilaksanakan pada Bidang Aplikasi Informatika Pemerintahan, khususnya di Seksi Infrastruktur dan Teknologi. Unit ini bertugas mengelola infrastruktur teknologi informasi, termasuk jaringan komputer dan perangkat pendukung yang digunakan untuk menunjang operasional layanan pemerintahan berbasis elektronik di lingkungan Pemerintah Kabupaten Bandung Barat. Dalam penelitian ini, kegiatan difokuskan pada pembangunan Aplikasi Penjadwalan Rapat Berbasis Web sebagai salah satu solusi untuk mendukung pengelolaan penjadwalan rapat, penugasan operator, serta dokumentasi peminjaman peralatan agar proses koordinasi pelaksanaan rapat menjadi lebih efektif.

BAB 4
ANALISIS DAN PERANCANGAN

4.1 Sistem Saat Ini (Current System)

Pada saat ini, proses penjadwalan kegiatan rapat masih dilakukan secara konvensional melalui surat dan komunikasi langsung maupun aplikasi WhatsApp. Informasi yang disampaikan hanya mencakup waktu dan lokasi kegiatan tanpa disertai detail penugasan. Operator sebagai pelaksana tidak terlibat secara langsung dalam proses awal sehingga diperlukan konfirmasi tambahan terkait penugasan dan kebutuhan peralatan. Proses peminjaman peralatan juga belum memiliki pencatatan yang terintegrasi. Kondisi tersebut menyebabkan proses koordinasi kurang efektif, berpotensi menimbulkan ketidaksesuaian informasi dalam pelaksanaan kegiatan.

4.1.1 Diagram Use Case Bisnis

Business Use Case Model merupakan pemodelan yang digunakan untuk menggambarkan proses bisnis dalam suatu organisasi beserta interaksi yang terjadi dengan pihak luar. Proses bisnis tersebut terdiri dari serangkaian aktivitas yang dilakukan untuk menghasilkan layanan atau keluaran sesuai kebutuhan pengguna.

Tabel 4.1 Diagram Use Case Bisnis

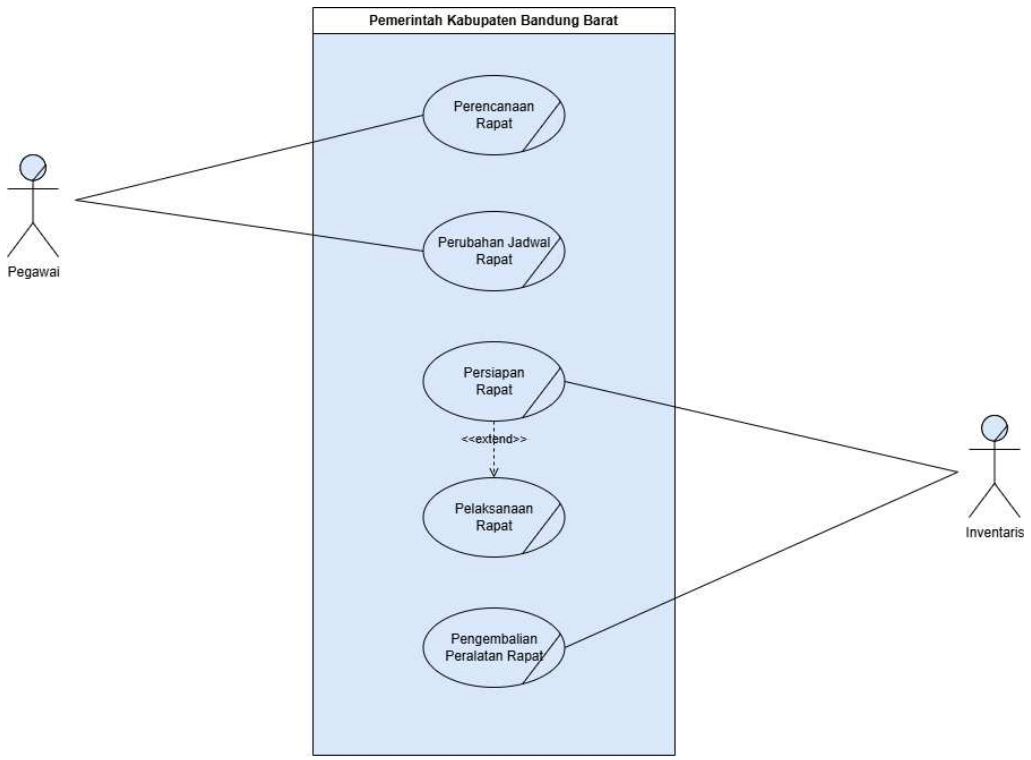
Use Case Bisnis
 *Gambar 4.1 Diagram Use Case Bisnis*
Deskripsi

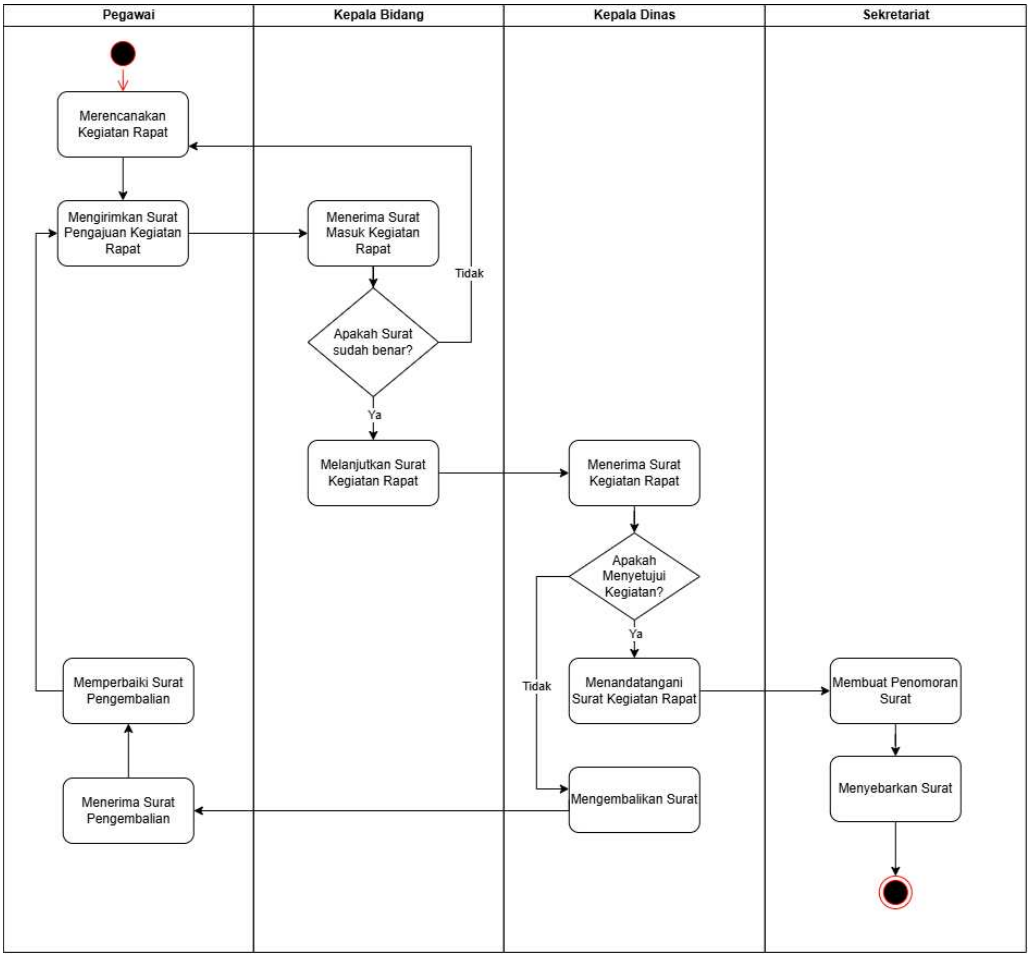
Diagram use case bisnis penjadwalan rapat menyajikan proses bisnis yang berkaitan dengan penyelenggaraan kegiatan rapat di Diskominfotik Kabupaten Bandung Barat. Proses bisnis tersebut terdiri atas perencanaan rapat, perubahan jadwal rapat, persiapan rapat, pelaksanaan rapat, serta pengembalian peralatan rapat. Diagram ini menunjukkan hubungan antara aktor yang terlibat dengan setiap proses bisnis yang dijalankan sesuai tugas dan tanggung jawab masing-masing dalam mendukung pelaksanaan kegiatan rapat

4.1.2 Diagram Aktivitas

Diagram aktivitas merupakan representasi alur proses yang menunjukkan urutan aktivitas, keputusan, dan perpindahan proses dalam suatu kegiatan. Diagram ini digunakan untuk memodelkan proses bisnis berdasarkan aktivitas yang dilakukan oleh pihak yang terlibat sehingga hubungan antar aktivitas dapat terlihat secara jelas, baik yang berjalan secara berurutan maupun secara paralel. Pada penelitian ini, diagram aktivitas digunakan untuk menunjukkan alur proses yang terdapat pada aplikasi penjadwalan rapat.

Tabel 4.2 Diagram Aktivitas Perencanaan Kegiatan Rapat

Diagram Aktivitas Perencanaan Rapat

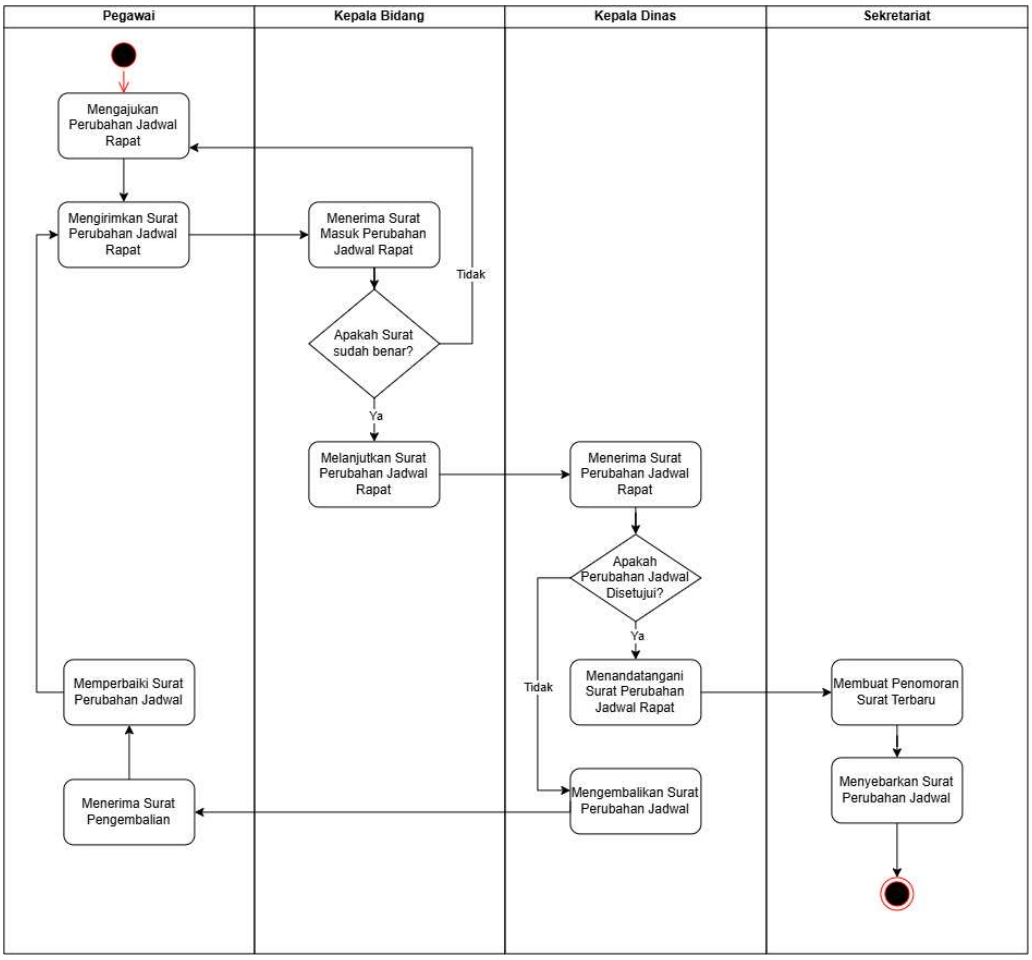


Gambar 4.2 Diagram Aktivitas Perencanaan Rapat

Deskripsi
Diagram aktivitas perencanaan rapat menggambarkan alur pengajuan kegiatan rapat yang dimulai dari pegawai pada bidang atau seksi penyelenggara hingga proses persetujuan oleh pimpinan. Alur tersebut mencakup pengiriman surat pengajuan, pemeriksaan surat kegiatan, persetujuan kegiatan rapat, pembuatan penomoran surat, kemudian dilanjutkan dengan penyebaran surat kepada pihak terkait

Tabel 4.3 Diagram Aktivitas Perubahan Jadwal Rapat

Diagram Aktivitas Perubahan Jadwal Rapat

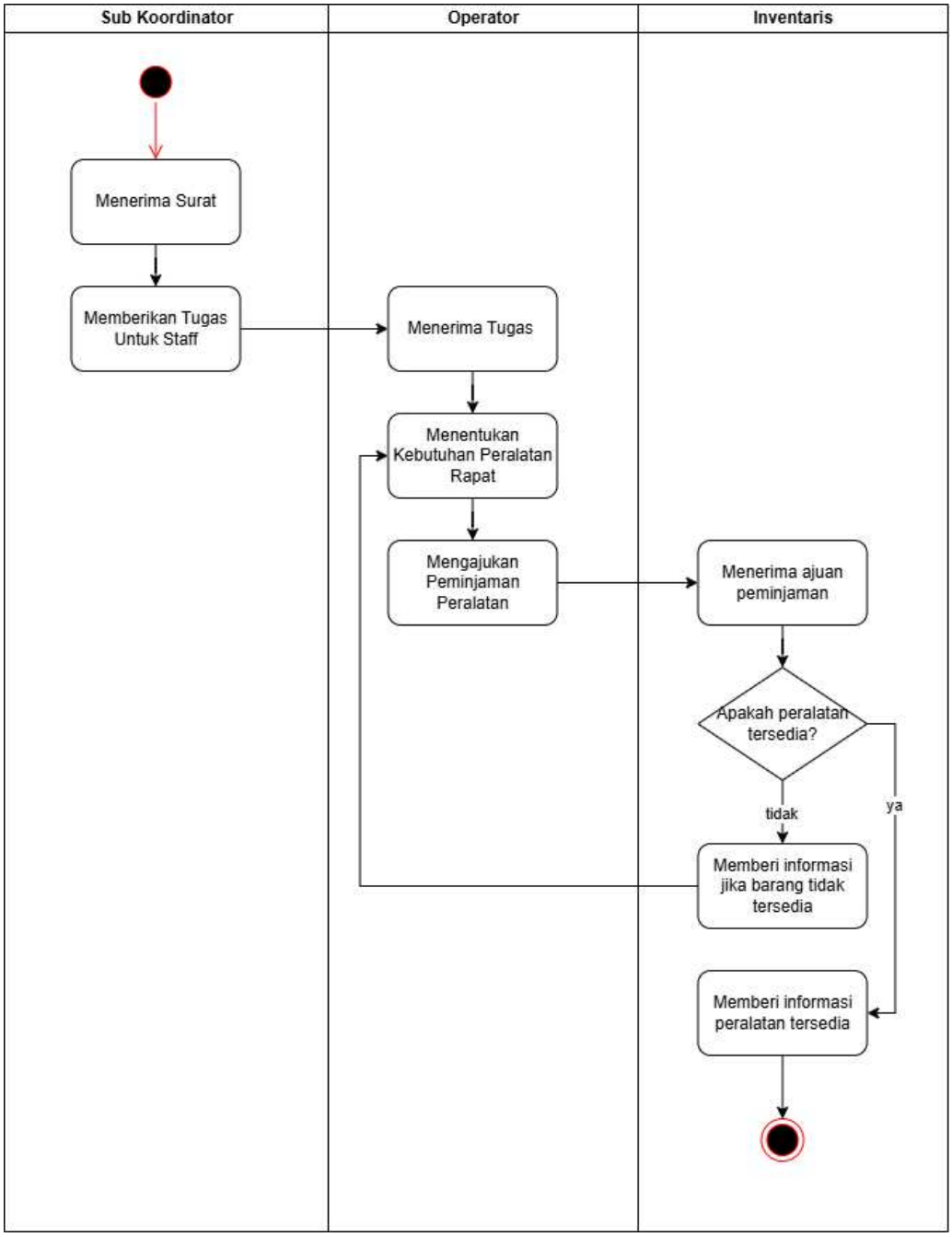


Gambar 4.3 Diagram Aktivitas Perubahan Kegiatan Rapat

Deskripsi
Diagram aktivitas perubahan jadwal rapat menggambarkan alur perubahan jadwal kegiatan rapat yang dimulai dari pegawai pada bidang atau seksi penyelenggara hingga proses persetujuan perubahan jadwal oleh pimpinan. Alur tersebut mencakup pengajuan surat perubahan jadwal rapat, pemeriksaan surat perubahan jadwal, persetujuan perubahan jadwal kegiatan, penandatanganan surat perubahan jadwal, pembuatan penomoran surat terbaru, kemudian dilanjutkan dengan penyebaran informasi perubahan jadwal kepada pihak terkait

Tabel 4.4 Diagram Aktivitas Persiapan Kegiatan Rapat

Diagram Aktivitas Persiapan Rapat



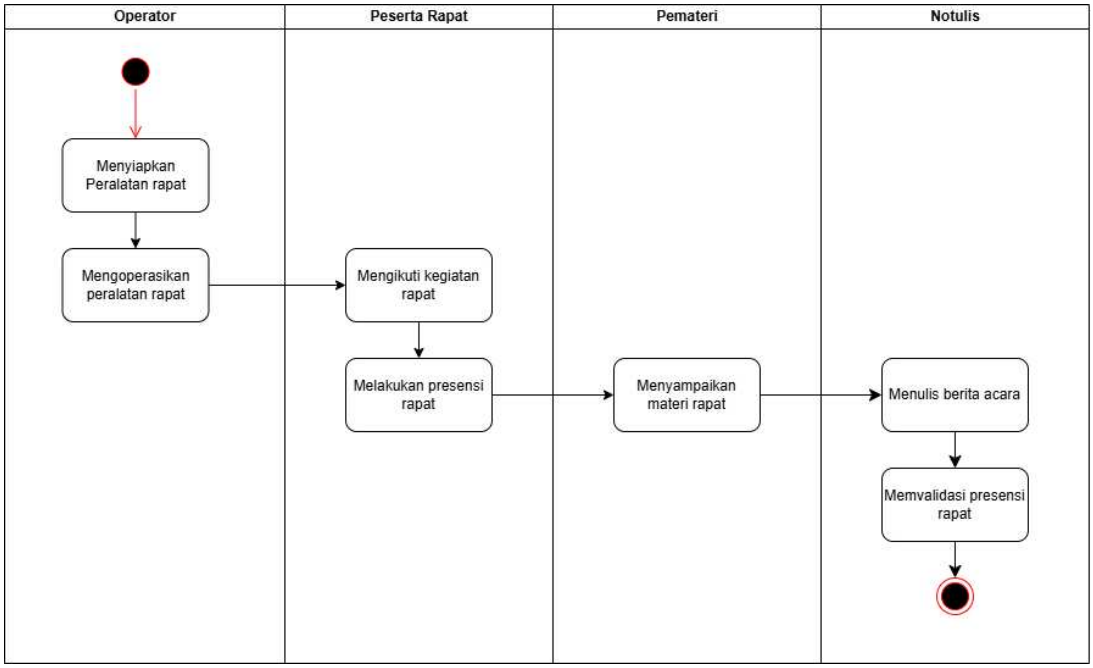
Gambar 4.4 Diagram Aktivitas Persiapan Kegiatan Rapat

Deskripsi

Diagram aktivitas persiapan rapat menggambarkan proses persiapan kegiatan setelah surat rapat diterima. Kegiatan yang dilakukan meliputi pembagian tugas kepada operator, penentuan kebutuhan peralatan, pengajuan peminjaman peralatan, dan pemeriksaan ketersediaan peralatan oleh bagian inventaris

Tabel 4.5 Diagram Aktivitas Pelaksanaan Kegiatan Rapat

Diagram Aktivitas Pelaksanaan Rapat

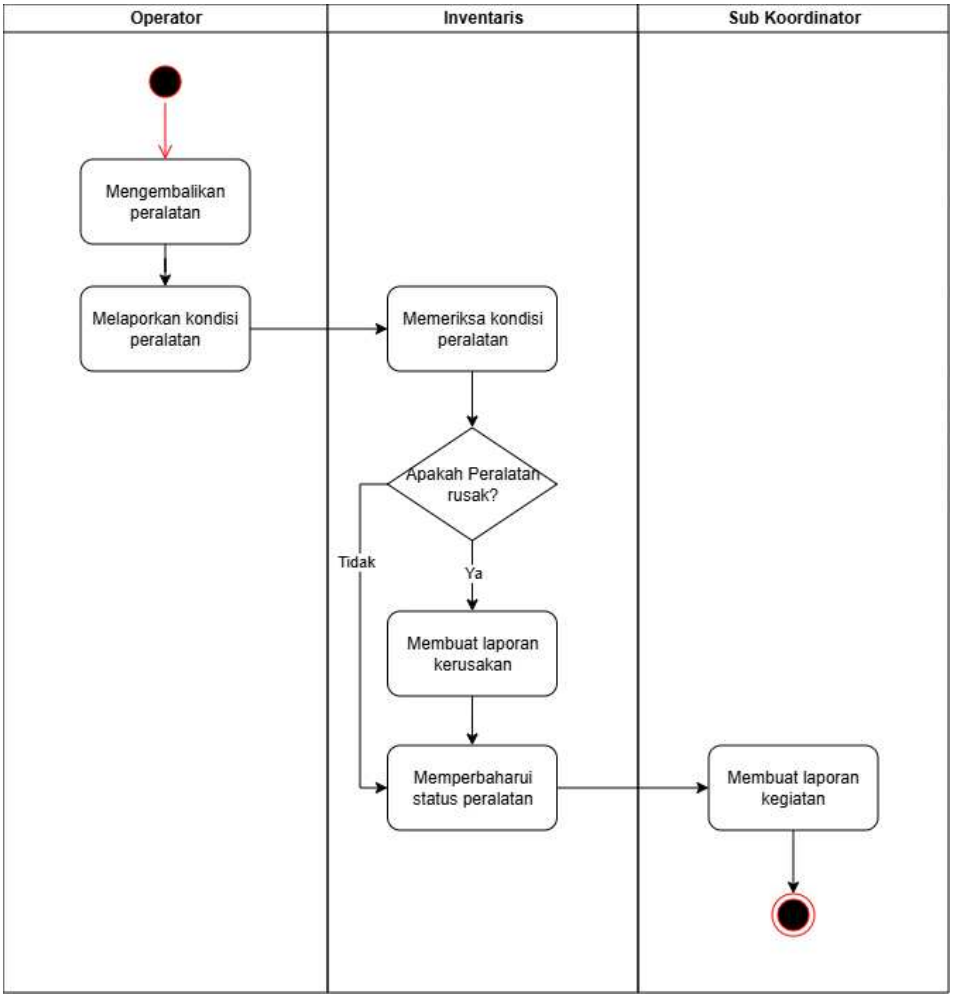


Gambar 4.5 Diagram Aktivitas Pelaksanaan Kegiatan Rapat

Deskripsi

Diagram aktivitas pelaksanaan rapat menggambarkan alur proses kegiatan rapat yang dimulai dari operator dalam menyiapkan serta mengoperasikan peralatan rapat yang digunakan selama kegiatan berlangsung. Selanjutnya peserta rapat mengikuti kegiatan rapat dan melakukan presensi rapat. Setelah proses presensi dilakukan, pemateri menyampaikan materi rapat kepada peserta rapat. Kemudian Notulis menulis berita acara rapat dan melakukan validasi presensi rapat hingga seluruh proses kegiatan rapat selesai

Tabel 4.6 Diagram Aktivitas Penyelesaian Kegiatan Rapat

Diagram Aktivitas Penyelesaian Rapat
 *Gambar 4.6 Diagram Aktivitas Penyelesaian Kegiatan Rapat*
Deskripsi
Diagram aktivitas penyelesaian rapat menggambarkan proses setelah kegiatan rapat selesai dilaksanakan. Tahapan yang dilakukan meliputi pengembalian peralatan, pemeriksaan kondisi peralatan, pembuatan laporan kerusakan apabila ditemukan kerusakan, pembaruan status peralatan, kemudian diakhiri dengan pembuatan laporan hasil kegiatan rapat

4.2 Analisis Kebutuhan

Pada tahap ini dilakukan analisis kebutuhan yang akan digunakan dalam pembangunan aplikasi penjadwalan rapat. Analisis ini dilakukan berdasarkan hasil analisis sistem yang sedang berjalan serta permasalahan yang telah diidentifikasi sebelumnya.

4.2.1 Analisis Stakeholder

Analisis stakeholder dilakukan untuk mengidentifikasi pihak-pihak yang terlibat dalam penggunaan aplikasi. Stakeholder dalam aplikasi ini terdiri dari beberapa peran yang memiliki fungsi dan tanggung jawab masing-masing dalam proses penjadwalan rapat.

Tabel 4.7 Analisis Stakeholder

No	Stakeholder	Deskripsi
1.	Sub Koordinator	Orang yang bertugas mengelola kegiatan penjadwalan rapat, menerima informasi kegiatan, dan memberikan penugasan kepada staff operator
2.	Staff Operator	Orang yang bertugas melaksanakan kegiatan di lapangan, melakukan pengecekan lokasi, serta menyiapkan peralatan dan jaringan
3.	Inventaris	Orang yang bertanggung jawab terhadap pencatatan, peminjaman, dan pengembalian peralatan yang digunakan dalam kegiatan rapat
4.	Pegawai	Orang yang mengajukan atau menyelenggarakan kegiatan rapat sesuai kebutuhan unit kerja serta berkoordinasi dalam pelaksanaan kegiatan rapat
5.	Peserta Rapat	Orang yang mengikuti kegiatan rapat untuk menerima informasi, memberikan masukan, atau terlibat dalam pengambilan keputusan
6.	Pemateri	Orang yang menyampaikan materi atau informasi dalam kegiatan rapat sesuai dengan agenda yang telah ditentukan
7.	Notulis	Orang yang mendukung pelaksanaan kegiatan rapat melalui pencatatan hasil rapat dan presensi kegiatan

4.2.2 Analisis Fungsional

Analisis kebutuhan fungsional dilakukan untuk mengetahui kebutuhan fungsional yang diperlukan dalam pembangunan perangkat lunak. Kebutuhan fungsional ini digunakan sebagai acuan dalam pengembangan perangkat lunak agar dapat berjalan sesuai dengan kebutuhan pengguna.

Tabel 4.8 Analisis Fungsional

No	Kode	Kebutuhan Fungsional	Deskripsi
1.	FR-01	Perangkat lunak harus dapat melihat jadwal rapat	Perangkat lunak memiliki halaman jadwal untuk melihat jadwal rapat
2.	FR-02	Perangkat lunak harus dapat menambah jadwal rapat	Perangkat lunak memiliki halaman tambah jadwal untuk menambahkan jadwal rapat
3.	FR-03	Perangkat lunak harus dapat mengubah jadwal rapat	Perangkat lunak memiliki halaman edit jadwal untuk mengubah jadwal rapat sebelum kegiatan dimulai
4.	FR-04	Perangkat lunak harus dapat membatalkan jadwal rapat	Perangkat lunak memiliki fungsi untuk membatalkan jadwal rapat sebelum kegiatan dimulai
5.	FR-05	Perangkat lunak harus dapat melihat peralatan	Perangkat lunak memiliki halaman peralatan untuk melihat informasi peralatan pendukung kegiatan rapat

6.	FR-06	Perangkat lunak harus dapat melihat peminjaman peralatan	Perangkat lunak memiliki halaman peminjaman untuk menampilkan informasi serta riwayat pengajuan peminjaman peralatan yang dilakukan oleh operator
7.	FR-07	Perangkat lunak harus dapat melakukan peminjaman peralatan	Perangkat lunak memiliki halaman ajukan peminjaman untuk operator dalam melakukan proses peminjaman peralatan yang digunakan dalam kegiatan rapat
8.	FR-08	Perangkat lunak harus dapat mengubah peminjaman peralatan	Perangkat lunak memiliki halaman edit peminjaman untuk mengubah pengajuan peminjaman peralatan sebelum proses peminjaman disetujui atau digunakan.
9.	FR-09	Perangkat lunak harus dapat membatalkan peminjaman peralatan	Perangkat lunak memiliki fungsi untuk membatalkan pengajuan peminjaman peralatan sebelum proses peminjaman disetujui atau digunakan

Berdasarkan kebutuhan fungsional pada tabel di atas, maka akan menjadi fitur untuk perangkat lunak yang dibangun. Adapun fitur yang akan disediakan pada perangkat lunak dapat dilihat pada tabel berikut.

Tabel 4.9 Fitur

No	Fitur	Kode Kebutuhan Fungsional
1	Halaman Dashboard	FR-01, FR-01, FR-06
2	Halaman Jadwal Rapat	FR-01, FR-02, FR-03, FR-04
3	Halaman Laporan	FR-01, FR-05
4	Halaman Jadwal Operator	FR-01
5	Halaman Peralatan	FR-05
6	Halaman Peminjaman Peralatan	FR-06, FR-07, FR-08, FR-09

4.2.3 Analisis Pengguna

Pada tahap ini merupakan analisis yang digunakan untuk menentukan pengguna yang akan menggunakan dan berinteraksi dengan perangkat lunak yang dibangun. Daftar pengguna dapat dilihat pada tabel berikut.

Tabel 4.10 Analisis Pengguna

No	Pengguna	Deskripsi
1	Sub Koordinator	Pengguna yang menggunakan aplikasi serta memiliki akses untuk mengelola penjadwalan rapat dan memberikan penugasan kepada operator
2	Staff Operator	Pengguna yang menggunakan aplikasi untuk melihat jadwal rapat serta mengajukan, mengubah, dan membatalkan peminjaman peralatan yang dibutuhkan

4.2.4 Pemodelan Berbasis Skenario

Pemodelan berbasis skenario digunakan untuk menggambarkan interaksi antara pengguna dengan sistem yang dibangun. Pemodelan ini dilakukan menggunakan UML dalam

bentuk use case diagram dan activity diagram. Use case diagram digunakan untuk menunjukkan hubungan antara aktor dengan sistem, sedangkan activity diagram digunakan untuk menggambarkan alur proses yang terjadi. Diagram use case dari sistem yang diusulkan dapat dilihat pada gambar berikut.

Tabel 4.11 Use Case

Use Case untuk aplikasi penjadwalan rapat dan peminjaman peralatan
 Gambar 4.7 Use Case
Deskripsi
Diagram use case aplikasi penjadwalan rapat Diskominfotik Kabupaten Bandung Barat digunakan untuk menggambarkan interaksi antara pengguna dengan aplikasi dalam pengelolaan penjadwalan rapat, peminjaman peralatan.

4.2.5 Deskripsi Use Case

Diagram use case digunakan untuk menggambarkan interaksi antara pengguna dengan aplikasi yang dibangun. Setiap use case menjelaskan fungsi yang dapat dilakukan oleh pengguna. Penjelasan masing-masing use case pada aplikasi penjadwalan rapat dijabarkan pada tabel berikut.

Tabel 4.12 Deskripsi Use Case

No	Kode	Use Case	Deskripsi
1.	UC-01	Melihat jadwal rapat	Use case melihat jadwal rapat digunakan untuk melihat jadwal kegiatan rapat
2.	UC-02	Melakukan tambah jadwal rapat	Use case melakukan tambah jadwal rapat digunakan untuk menambahkan jadwal kegiatan rapat
3.	UC-03	Melakukan ubah jadwal rapat	Use case melakukan ubah jadwal rapat digunakan untuk memperbarui jadwal sebelum kegiatan dimulai
4.	UC-04	Melakukan batal jadwal rapat	Use case melakukan batal jadwal rapat digunakan untuk membatalkan jadwal sebelum kegiatan dimulai
5.	UC-05	Melihat peralatan	Use case melihat peralatan digunakan untuk melihat informasi peralatan
6.	UC-06	Melihat peminjaman peralatan	Use case melihat peminjaman peralatan digunakan untuk melihat informasi serta riwayat pengajuan peminjaman peralatan yang dilakukan oleh operator
7.	UC-07	Melakukan peminjaman peralatan	Use case melakukan peminjaman peralatan digunakan untuk meminjam peralatan pendukung kegiatan rapat
8.	UC-08	Melakukan ubah peminjaman peralatan	Use case melakukan ubah peminjaman peralatan digunakan untuk memperbarui pengajuan peminjaman peralatan sebelum proses peminjaman dilakukan
9.	UC-09	Melakukan batal peminjaman peralatan	Use case melakukan batal peminjaman peralatan digunakan untuk membatalkan pengajuan peminjaman peralatan sebelum proses peminjaman dilakukan

4.2.6 Skenario Use Case

Skenario use case digunakan untuk menggambarkan alur proses yang dilakukan oleh pengguna dalam setiap use case. Skenario ini menjelaskan langkah-langkah interaksi pengguna dengan aplikasi berdasarkan fungsi yang tersedia. Skenario use case pada aplikasi penjadwalan rapat disajikan pada tabel berikut.

Tabel 4.13 Skenario Use Case Melihat Jadwal Rapat

Identifikasi	
No	UC-01
Nama	Melihat jadwal rapat
Tujuan	Melihat informasi jadwal kegiatan rapat

Deskripsi	Use case melihat jadwal rapat digunakan untuk melihat jadwal kegiatan rapat
Aktor	Sub Koordinator, Staff Operator
Skenario Utama	
Kondisi Awal	Halaman Dashboard
Aksi Aktor	Reaksi Sistem
1. Aktor berada di halaman dashboard	
	2. Sistem menampilkan halaman dashboard
3. Aktor menekan menu “Jadwal”	
	4. Sistem mengambil data jadwal dari database
	5. Sistem menampilkan halaman jadwal yang berisi data jadwal
Kondisi Akhir	Halaman Jadwal

Tabel 4.14 Skenario Use Case Melakukan Tambah Jadwal Rapat

Identifikasi	
No	UC-02
Nama	Melakukan tambah jadwal rapat
Tujuan	Menambahkan data jadwal rapat baru
Deskripsi	Use case melakukan tambah jadwal rapat digunakan untuk menambahkan jadwal kegiatan rapat
Aktor	Sub Koordinator
Skenario Utama	
Kondisi Awal	Halaman Dashboard
Aksi Aktor	Reaksi Sistem
1. Aktor berada di halaman dashboard	
	2. Sistem menampilkan halaman dashboard
3. Aktor menekan menu “Jadwal Rapat”	
	4. Sistem menampilkan halaman jadwal
5. Aktor menekan tombol “Tambah Jadwal”	
	6. Sistem menampilkan halaman tambah jadwal
7. Aktor mengisi form jadwal baru	
	8. Sistem melakukan validasi data jadwal dengan memeriksa bentrok penugasan operator berdasarkan tanggal dan waktu kegiatan
	9. Sistem memeriksa data peralatan yang dipilih
10. Aktor menekan tombol “Simpan”	
	11. Sistem membuat link meeting Zoom secara otomatis melalui Zoom API (jika opsi "Buat Link Zoom Otomatis" dicentang dan platform Online atau Hybrid), termasuk memilih akun Zoom yang tidak bentrok dengan jadwal lain pada waktu yang sama.
	12. Sistem menyimpan data jadwal ke dalam *database*

	13. Sistem mengirim pesan pemberitahuan jadwal melalui WhatsApp kepada operator terkait
	14. Sistem menampilkan notifikasi "Jadwal berhasil ditambahkan dan notifikasi WA telah dikirim."
	15. Sistem kembali ke halaman jadwal
Kondisi Akhir	Halaman Jadwal

Tabel 4.15 Skenario Use Case Melakukan Ubah Jadwal Rapat

Identifikasi	
No	UC-03
Nama	Melakukan ubah jadwal rapat
Tujuan	Mengubah data jadwal rapat sebelum dimulai
Deskripsi	Use case melakukan ubah jadwal rapat digunakan untuk memperbarui jadwal sebelum kegiatan dimulai
Aktor	Sub Koordinator
Skenario Utama	
Kondisi Awal	Halaman Dashboard
Aksi Aktor	Reaksi Sistem
1. Aktor berada di halaman dashboard	
	2. Sistem menampilkan halaman dashboard
3. Aktor menekan menu "Jadwal Rapat"	
	4. Sistem menampilkan halaman jadwal
5. Aktor menekan tombol "Edit" pada jadwal yang belum dimulai	
	6. Sistem menampilkan halaman edit jadwal
7. Aktor mengubah *form* jadwal rapat	
	8. Sistem melakukan validasi data jadwal dan pengecekan bentrok jadwal operator berdasarkan tanggal, waktu mulai, dan waktu selesai kegiatan
	9. Sistem memeriksa data peralatan yang dipilih
10. Aktor menekan tombol "Update"	
	11. Sistem menyelaraskan link meeting Zoom melalui Zoom API sesuai perubahan yang dilakukan (menghapus meeting lama jika opsi "Buat Link Zoom Otomatis" dimatikan atau platform bukan lagi Online/Hybrid, memperbarui judul/waktu meeting jika hanya itu yang berubah, atau membuat meeting baru jika akun Zoom diganti)
	12. Sistem memperbarui data jadwal ke dalam database
	13. Sistem mengirim pesan pemberitahuan perubahan jadwal melalui WhatsApp kepada operator terkait
	14. Sistem menampilkan notifikasi "Jadwal berhasil diperbarui."

	15. Sistem kembali ke halaman Jadwal
Kondisi Akhir	Halaman Jadwal

Tabel 4.16 Skenario Use Case Melakukan Batal Jadwal Rapat

Identifikasi	
No	UC-04
Nama	Melakukan batal jadwal rapat
Tujuan	Membatalkan jadwal rapat sebelum dimulai
Deskripsi	Use case melakukan batal jadwal rapat digunakan untuk membatalkan jadwal sebelum kegiatan dimulai
Aktor	Sub Koordinator
Skenario Utama	
Kondisi Awal	Halaman Dashboard
Aksi Aktor	Reaksi Sistem
1. Aktor berada di halaman dashboard	
	2. Sistem menampilkan halaman dashboard
3. Aktor menekan menu “Jadwal Rapat”	
	4. Sistem menampilkan halaman jadwal
5. Aktor memilih jadwal rapat yang akan dibatalkan	
6. Aktor menekan tombol “Batalkan Jadwal” pada jadwal yang belum dimulai	
	7. Sistem menampilkan konfirmasi pembatalan jadwal
8. Aktor memasukan alasan pembatalan	
9. Aktor menekan tombol “Ya”	
	10. Sistem menghapus meeting Zoom melalui Zoom API, jika jadwal ini memiliki link Zoom otomatis
	11. Sistem memperbarui status jadwal menjadi "Dibatalkan"
	12. Sistem mengirim pesan pemberitahuan pembatalan jadwal melalui WhatsApp kepada operator terkait
	13. Sistem menampilkan notifikasi “Jadwal berhasil dibatalkan dan notifikasi WA telah dikirim ke operator.”
	14. Sistem kembali ke halaman Jadwal
Kondisi Akhir	Halaman Jadwal

Tabel 4.17 Skenario Use Case Melihat Peralatan

Identifikasi	
No	UC-05
Nama	Melihat peralatan
Tujuan	Menampilkan informasi peralatan
Deskripsi	Use case melihat peralatan digunakan untuk melihat informasi peralatan
Aktor	Staff Operator

Skenario Utama	
Kondisi Awal	Halaman Dashboard
Aksi Aktor	Reaksi Sistem
1. Aktor berada di halaman dashboard	
	2. Sistem menampilkan halaman dashboard
3. Aktor menekan menu “Peralatan”	
	4. Sistem menampilkan halaman Peralatan yang berisi informasi peralatan, lokasi penyimpanan, jumlah stok, dan status ketersediaan
Kondisi Akhir	Halaman Peralatan

Tabel 4.18 Skenario Use Case Melihat Peminjaman Peralatan

Identifikasi	
No	UC-06
Nama	Melihat peminjaman peralatan
Tujuan	Menampilkan informasi peminjaman peralatan
Deskripsi	Use case melihat peminjaman peralatan digunakan untuk melihat informasi serta riwayat pengajuan peminjaman peralatan yang dilakukan oleh operator
Aktor	Staff Operator
Skenario Utama	
Kondisi Awal	Halaman Dashboard
Aksi Aktor	Reaksi Sistem
1. Aktor berada di halaman dashboard	
	2. Sistem menampilkan halaman dashboard
3. Aktor menekan menu “Peminjaman Peralatan”	
	4. Sistem menampilkan informasi setiap pengajuan peminjaman, seperti keperluan, tanggal peminjaman, rencana pengembalian, tanggal pengembalian, status pengajuan, serta aksi yang tersedia sesuai ketentuan
Kondisi Akhir	Halaman Peminjaman Peralatan

Tabel 4.19 Skenario Use Case Melakukan Peminjaman Peralatan

Identifikasi	
No	UC-07
Nama	Melakukan peminjaman peralatan
Tujuan	Mengajukan peminjaman peralatan untuk mendukung kegiatan rapat
Deskripsi	Use case melakukan peminjaman peralatan digunakan untuk meminjam peralatan pendukung kegiatan rapat
Aktor	Staff Operator
Skenario Utama	

Kondisi Awal	Halaman Dashboard
Aksi Aktor	Reaksi Sistem
1. Aktor berada di halaman dashboard	
	2. Sistem menampilkan halaman dashboard
3. Aktor menekan menu “Peminjaman Peralatan”	
	4. Sistem menampilkan daftar riwayat pengajuan peminjaman
5. Aktor menekan tombol "Ajukan Peminjaman"	
	6. Sistem menampilkan form pengajuan peminjaman
7. Aktor mengisi *form* pengajuan peminjaman peralatan	
8. Aktor menekan tombol "Kirim Pengajuan"	
	9. Sistem memeriksa (dari data jadwal yang sudah dimuat) apakah alat yang dipilih sudah diajukan operator lain untuk jadwal yang sama, lalu menampilkan konfirmasi duplikasi jika ditemukan
10. Aktor memilih "Tetap Ajukan" atau "Batalkan" (hanya jika konfirmasi duplikasi muncul)	
	11. Sistem memeriksa (lewat permintaan ke server) apakah operator ini sendiri sudah berkali-kali mengajukan alat yang sama untuk tanggal pinjam yang sama, lalu menampilkan peringatan pengajuan berulang jika ditemukan
12. Aktor memilih "Ya, Tetap Ajukan" atau "Batal" (hanya jika peringatan pengajuan berulang muncul)	
	13. Sistem melakukan validasi data pengajuan
	14. Sistem memeriksa dan mengurangi (mereservasi) stok peralatan yang dipilih
	15. Sistem menyimpan data pengajuan peminjaman ke dalam database dengan status "Menunggu"
	16. Sistem mengirimkan pesan pengajuan peminjaman melalui WhatsApp kepada petugas inventaris
	17. Sistem menampilkan notifikasi "Pengajuan berhasil dikirim. Notifikasi telah dikirim ke petugas inventaris."
	18. Sistem menampilkan halaman peminjaman
Kondisi Akhir	Halaman Peminjaman Peralatan

Tabel 4.20 Skenario Use Case Melakukan Ubah Peminjaman Peralatan

Identifikasi	
No	UC-08
Nama	Melakukan ubah peminjaman peralatan
Tujuan	Memperbarui data pengajuan peminjaman peralatan
Deskripsi	Use case melakukan ubah peminjaman peralatan digunakan untuk memperbarui pengajuan peminjaman peralatan sebelum proses peminjaman dilakukan

Aktor	Staff Operator
Skenario Utama	
Kondisi Awal	Halaman Dashboard
Aksi Aktor	Reaksi Sistem
1. Aktor berada di halaman dashboard	
	2. Sistem menampilkan halaman dashboard
3. Aktor menekan menu “Peminjaman Peralatan”	
	4. Sistem menampilkan halaman peminjaman
5. Aktor menekan tombol “Edit” pada peralatan yang memiliki status “Menunggu”	
	6. Sistem menampilkan form ubah pengajuan peminjaman beserta data yang telah tersimpan
7. Aktor mengubah form peminjaman peralatan	
8. Aktor menekan tombol "Simpan Perubahan"	
	9. Sistem memeriksa (dari data jadwal yang sudah dimuat) apakah alat yang dipilih sudah diajukan operator lain untuk jadwal yang sama, lalu menampilkan konfirmasi duplikasi jika ditemukan
10. Aktor memilih "Tetap Ajukan" atau "Batalkan" (hanya jika konfirmasi duplikasi muncul)	
	11. Sistem memeriksa (lewat permintaan ke server) apakah operator ini sendiri sudah berkali-kali mengajukan alat yang sama untuk tanggal pinjam yang sama, lalu menampilkan peringatan pengajuan berulang jika ditemukan
12. Aktor memilih "Ya, Tetap Ajukan" atau "Batal" (hanya jika peringatan pengajuan berulang muncul)	
	13. Sistem melakukan validasi data pengajuan
	14. Sistem mengembalikan reservasi stok dari alat lama, lalu memeriksa dan mereservasi stok untuk alat yang baru dipilih
	15. Sistem memperbarui data pengajuan peminjaman pada basis data
	16. Sistem mengirim pesan pemberitahuan perubahan peminjaman melalui WhatsApp kepada petugas inventaris terkait
	17. Sistem menampilkan notifikasi "Pengajuan berhasil diperbarui. Notifikasi telah dikirim ke petugas inventaris."
	18. Sistem menampilkan halaman Peminjaman
Kondisi Akhir	Halaman Peminjaman Peralatan

Tabel 4.21 Skenario Use Case Melakukan Batal Peminjaman Peralatan

Identifikasi	
No	UC-09

Nama	Melakukan Batal Peminjaman Peralatan
Tujuan	Membatalkan pengajuan peminjaman peralatan yang telah diajukan
Deskripsi	Use case melakukan batal peminjaman peralatan digunakan untuk membatalkan pengajuan peminjaman peralatan sebelum proses peminjaman dilakukan
Aktor	Staff Operator
Skenario Utama	
Kondisi Awal	Halaman Dashboard
Aksi Aktor	Reaksi Sistem
1. Aktor berada di halaman dashboard	
	2. Sistem menampilkan halaman dashboard
3. Aktor menekan menu “Peminjaman Peralatan”	
	4. Sistem menampilkan halaman peminjaman
5. Aktor menekan tombol “Batal” pada salah satu data pengajuan peminjaman	
	6. Sistem menampilkan konfirmasi pembatalan pengajuan
7. Aktor mengisi alasan pembatalan pengajuan peralatan	
8. Aktor menekan tombol “Konfirmasi Batalkan”	
	9. Sistem memvalidasi alasan pembatalan
	10. Sistem mengembalikan stok peralatan yang sebelumnya direservasi untuk pengajuan ini
	11. Sistem memperbarui status pengajuan menjadi "Dibatalkan" beserta alasan pembatalannya pada basis data
	12. Sistem mengirimkan pesan pembatalan pengajuan melalui WhatsApp kepada petugas inventaris
	13. Sistem menampilkan notifikasi "Pengajuan berhasil dibatalkan dan notifikasi WA telah dikirim ke inventaris."
	14. Sistem menampilkan halaman Peminjaman
Kondisi Akhir	Halaman Peminjaman Peralatan

4.3 Pemodelan Data

Pemodelan data digunakan untuk menggambarkan struktur data yang akan digunakan dalam suatu sistem atau aplikasi. Pemodelan ini bertujuan untuk merepresentasikan hubungan antar data sehingga proses pengolahan data dapat dilakukan secara terstruktur dan mudah dipahami.

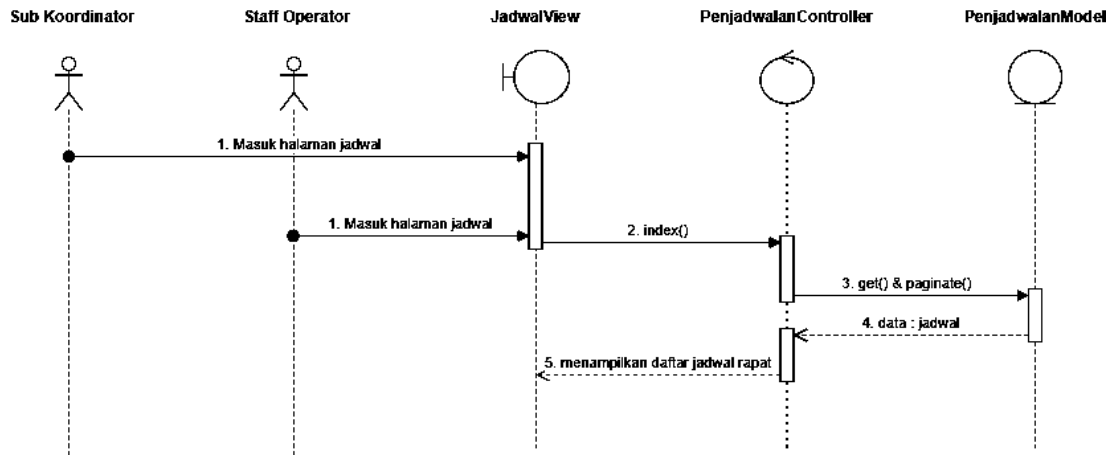
4.3.1 Pemodelan Berbasis Perilaku

Pemodelan berbasis perilaku digunakan untuk menunjukkan interaksi yang terjadi antara pengguna dan aplikasi selama proses berlangsung. Pemodelan ini berfokus pada urutan aktivitas, respons aplikasi, serta hubungan antar komponen yang terlibat dalam suatu proses.

Salah satu bentuk pemodelan berbasis perilaku adalah sequence diagram. Sequence diagram merupakan representasi interaksi antar objek yang terlibat dalam suatu use case melalui pertukaran pesan yang terjadi secara berurutan berdasarkan waktu. Melalui diagram ini, alur komunikasi dan proses yang berlangsung pada aplikasi dapat dipahami dengan lebih jelas.

Tabel 4.22 Sequence Melihat Jadwal Rapat

Melihat Jadwal Rapat



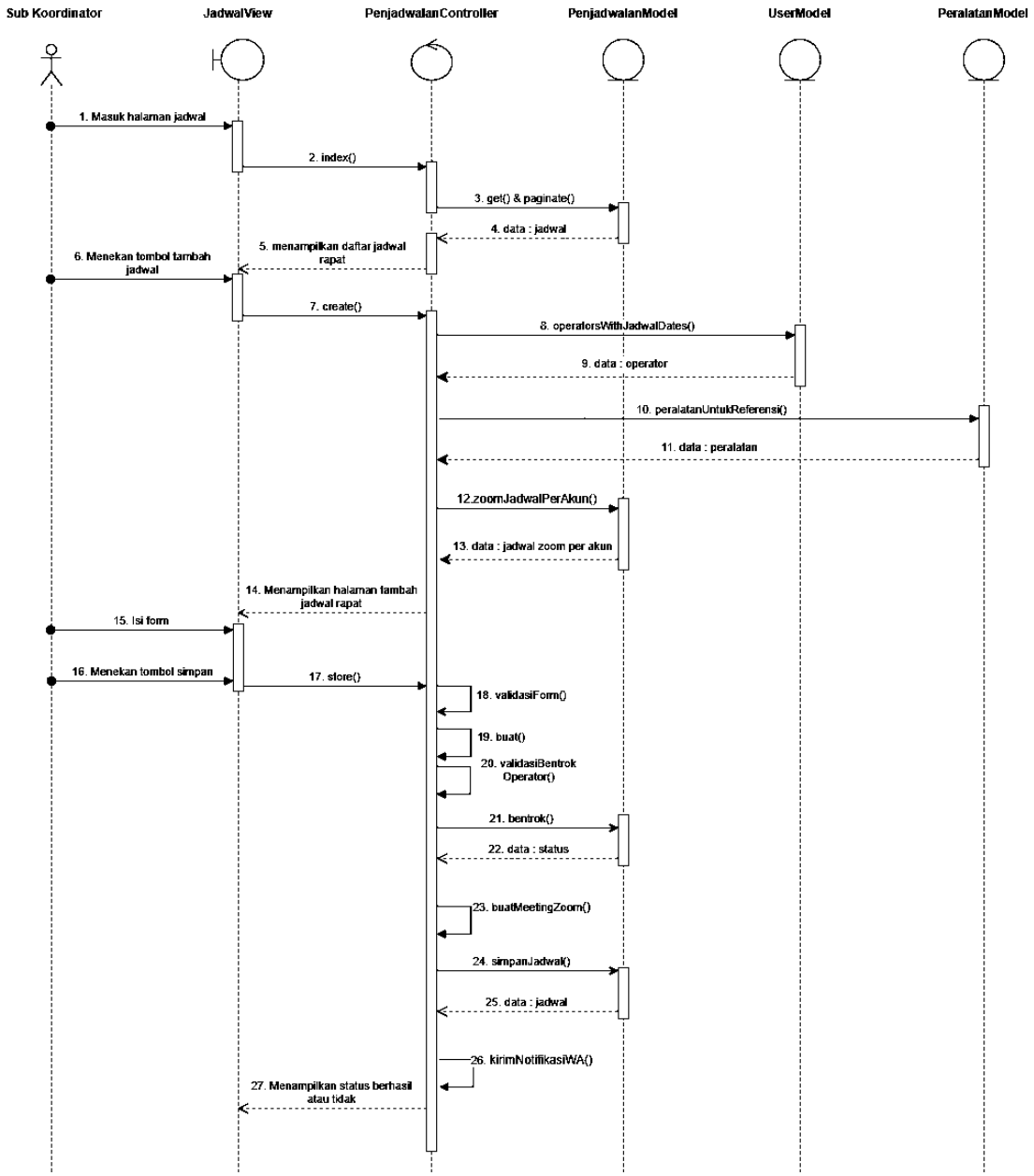
Gambar 4.8 Sequence Melihat Jadwal Rapat

Deskripsi

Sequence diagram di atas menggambarkan proses saat Sub Koordinator dan Staff Operator membuka halaman jadwal dan sistem menampilkan daftar jadwal rapat

1. Sub Koordinator dan Staff Operator masuk ke halaman jadwal rapat
2. JadwalView mengirim permintaan index() ke PenjadwalanController
3. PenjadwalanController mengambil data jadwal menggunakan fungsi get() dan paginate() pada PenjadwalanModel
4. PenjadwalanModel mengirimkan data jadwal kepada PenjadwalanController
5. PenjadwalanController menampilkan daftar jadwal rapat pada halaman jadwal

Tabel 4.23 Sequence Melakukan Tambah Jadwal

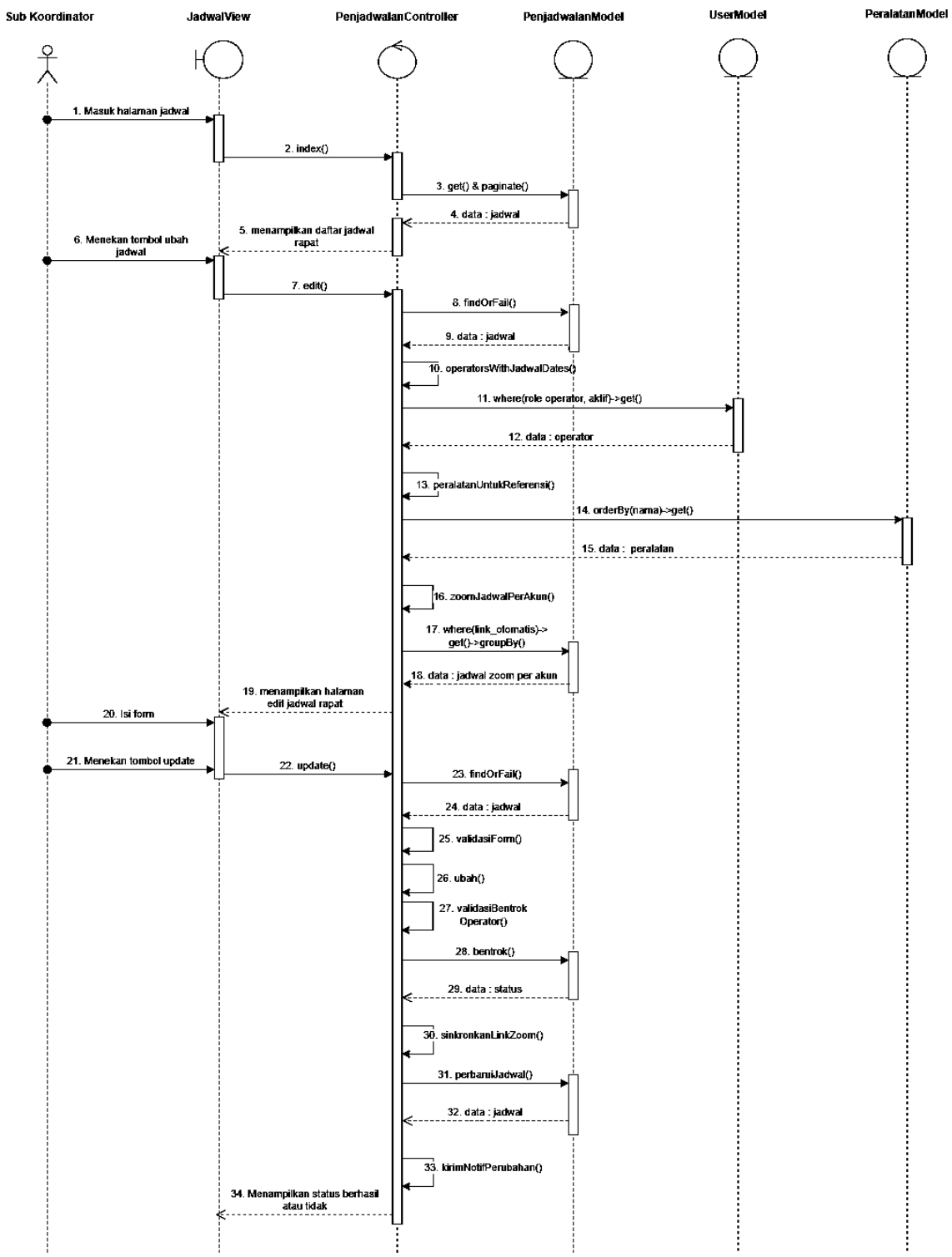
Melakukan Tambah Jadwal

Deskripsi

Gambar 4.9 Sequence Melakukan Tambah Jadwal

Sequence diagram di atas menggambarkan proses saat Sub Koordinator menambahkan jadwal rapat baru hingga data disimpan oleh sistem

1. Sub Koordinator masuk ke halaman jadwal rapat.
2. JadwalView mengirim permintaan index() kepada PenjadwalanController
3. PenjadwalanController mengambil data jadwal menggunakan fungsi paginate() pada PenjadwalanModel
4. PenjadwalanModel mengirimkan data jadwal kepada PenjadwalanController
5. PenjadwalanController meneruskan data tersebut kepada JadwalView untuk ditampilkan sebagai daftar jadwal rapat
6. Sub Koordinator menekan tombol Tambah Jadwal
7. JadwalView mengirim permintaan create() kepada PenjadwalanController
8. PenjadwalanController menjalankan fungsi internal operatorsWithJadwalDates(), yang mengambil daftar operator aktif dengan melakukan query pada UserModel
9. UserModel mengirimkan data operator kepada PenjadwalanController
10. PenjadwalanController menjalankan fungsi internal peralatanUntukReferensi(), yang mengambil daftar peralatan referensi dengan melakukan query pada PeralatanModel
11. PeralatanModel mengirimkan data peralatan kepada PenjadwalanController
12. PenjadwalanController menjalankan fungsi internal zoomJadwalPerAkun(), yang mengambil jadwal Zoom otomatis yang sudah ada per akun dengan melakukan query pada PenjadwalanModel
13. PenjadwalanModel mengirimkan data jadwal Zoom per akun kepada PenjadwalanController
14. PenjadwalanController meneruskan seluruh data tersebut kepada JadwalView untuk ditampilkan sebagai halaman tambah jadwal rapat
15. Sub Koordinator mengisi form jadwal rapat
16. Sub Koordinator menekan tombol Simpan
17. JadwalView mengirim permintaan store() kepada PenjadwalanController
18. PenjadwalanController menjalankan fungsi validasiForm() secara internal untuk memvalidasi data yang diinputkan
19. PenjadwalanController menjalankan fungsi buat() secara internal untuk memproses pembuatan jadwal baru
20. PenjadwalanController menjalankan fungsi validasiBentrokOperator() secara internal untuk memeriksa apakah operator yang dipilih memiliki jadwal yang bertabrakan
21. PenjadwalanController memeriksa bentrok jadwal menggunakan fungsi bentrok() pada PenjadwalanModel
22. PenjadwalanModel mengirimkan status hasil pemeriksaan bentrok kepada PenjadwalanController
23. PenjadwalanController menjalankan proses buatMeetingZoom() secara internal untuk membuat meeting Zoom secara otomatis apabila diperlukan
24. PenjadwalanController menyimpan data jadwal menggunakan fungsi simpanJadwal() pada PenjadwalanModel, kemudian menyinkronkan data operator dan peralatan referensi terkait
25. PenjadwalanModel mengirimkan data jadwal yang telah berhasil disimpan kepada PenjadwalanController
26. PenjadwalanController menjalankan fungsi kirimNotifikasiWA() secara internal untuk mengirimkan notifikasi WhatsApp kepada operator yang terlibat
27. PenjadwalanController meneruskan status berhasil atau gagal kepada JadwalView untuk ditampilkan kepada Sub Koordinator

Tabel 4.24 Sequence Melakukan Ubah Jadwal Rapat

Melakukan Ubah Jadwal Rapat

Deskripsi

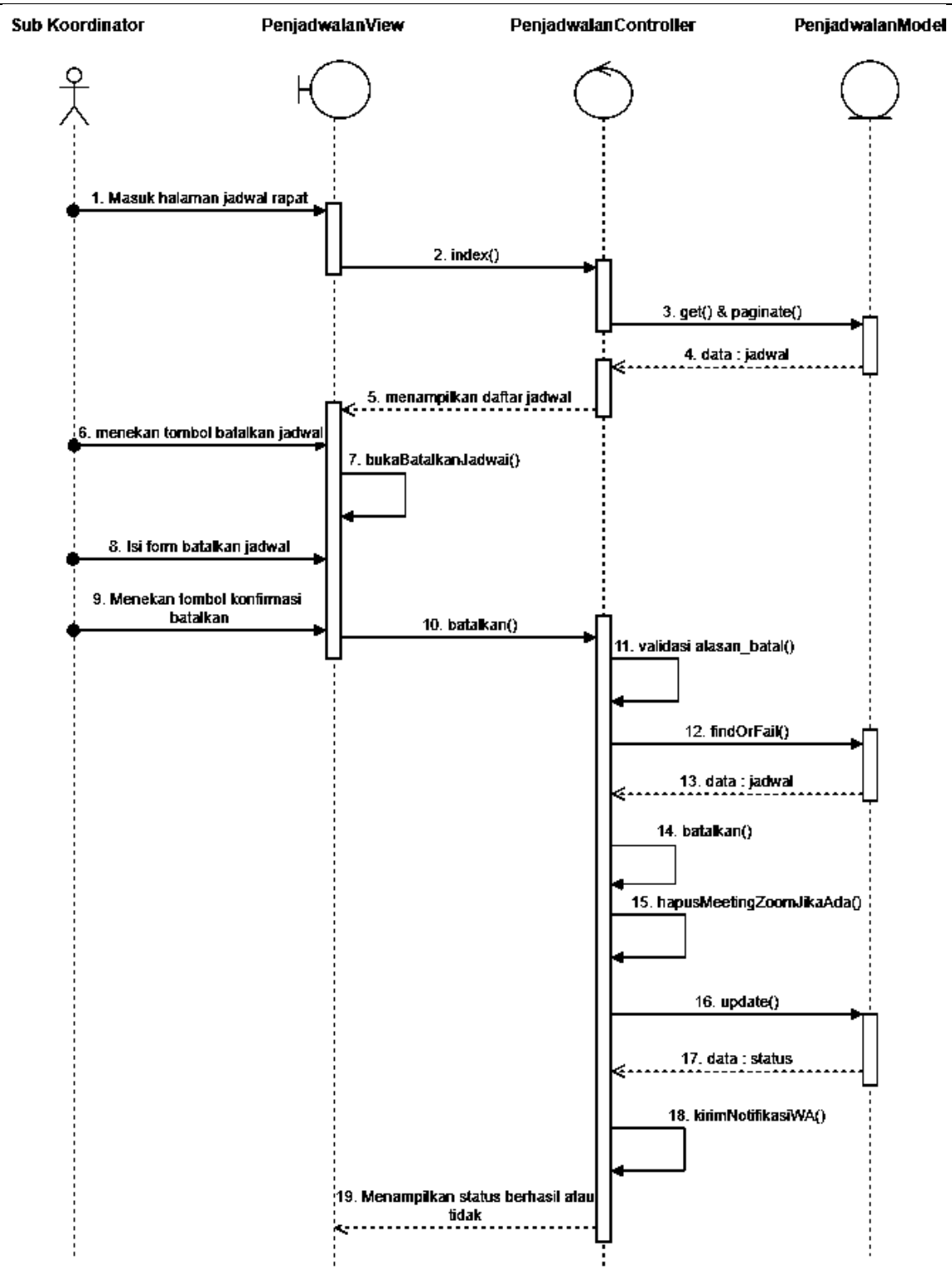
Gambar 4.10 Sequence Melakukan Ubah Jadwal Rapat

Sequence diagram di atas menggambarkan proses ketika Sub Koordinator melakukan perubahan data jadwal rapat hingga perubahan data berhasil disimpan oleh sistem.

1. Sub Koordinator masuk ke halaman jadwal rapat
2. JadwalView mengirim permintaan index() kepada PenjadwalanController
3. PenjadwalanController mengambil data jadwal menggunakan fungsi paginate() pada PenjadwalanModel
4. PenjadwalanModel mengirimkan data jadwal kepada PenjadwalanController
5. PenjadwalanController meneruskan data tersebut kepada JadwalView untuk ditampilkan sebagai daftar jadwal rapat
6. Sub Koordinator menekan tombol ubah jadwal
7. JadwalView mengirim permintaan edit() kepada PenjadwalanController
8. PenjadwalanController mengambil data jadwal menggunakan fungsi findOrFail() pada PenjadwalanModel
9. PenjadwalanModel mengirimkan data jadwal kepada PenjadwalanController
10. PenjadwalanController menjalankan fungsi internal operatorsWithJadwalDates(), yang mengambil daftar operator aktif dengan melakukan query pada UserModel
11. UserModel mengirimkan data operator kepada PenjadwalanController
12. PenjadwalanController menjalankan fungsi internal peralatanUntukReferensi(), yang mengambil daftar peralatan referensi dengan melakukan query pada PeralatanModel
13. PeralatanModel mengirimkan data peralatan kepada PenjadwalanController
14. PenjadwalanController menjalankan fungsi internal zoomJadwalPerAkun(), yang mengambil jadwal Zoom otomatis yang sudah ada per akun dengan melakukan query pada PenjadwalanModel
15. PenjadwalanModel mengirimkan data jadwal Zoom per akun kepada PenjadwalanController
16. PenjadwalanController meneruskan seluruh data tersebut kepada JadwalView untuk ditampilkan sebagai halaman ubah jadwal rapat
17. Sub Koordinator mengisi form jadwal
18. Sub Koordinator menekan tombol update
19. JadwalView mengirim permintaan update() kepada PenjadwalanController
20. PenjadwalanController mengambil ulang data jadwal menggunakan fungsi findOrFail() pada PenjadwalanModel
21. PenjadwalanModel mengirimkan data jadwal kepada PenjadwalanController
22. PenjadwalanController menjalankan fungsi validasiForm() secara internal untuk memvalidasi data yang diisi
23. PenjadwalanController menjalankan fungsi ubah() secara internal untuk memproses perubahan jadwal
24. PenjadwalanController menjalankan fungsi validasiBentrokOperator() secara internal untuk memeriksa jadwal operator yang dipilih
25. PenjadwalanController memeriksa jadwal bentrok menggunakan fungsi bentrok() pada PenjadwalanModel
26. PenjadwalanModel mengirimkan status bentrok kepada PenjadwalanController
27. PenjadwalanController menjalankan fungsi sinkronkanLinkZoom() secara internal untuk menyesuaikan meeting Zoom (menghapus, memperbarui, atau membuat meeting baru) sesuai perubahan yang dilakukan
28. PenjadwalanController memperbarui data jadwal menggunakan fungsi update() pada PenjadwalanModel
29. PenjadwalanModel mengirimkan data jadwal yang telah diperbarui kepada PenjadwalanController
30. PenjadwalanController menjalankan fungsi kirimNotifPerubahan() secara internal untuk mengirim notifikasi WhatsApp kepada operator terkait
31. PenjadwalanController meneruskan status keberhasilan kepada JadwalView untuk ditampilkan kepada Sub Koordinator

Tabel 4.25 Sequence Melakukan Batal Jadwal Rapat

Melakukan Batal Jadwal Rapat



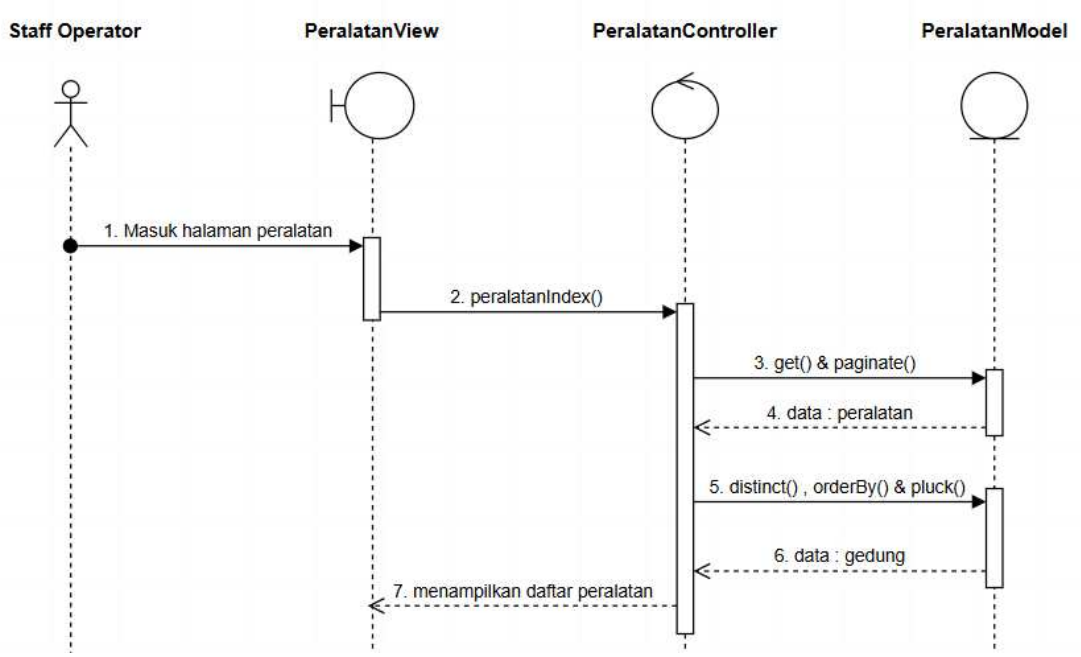
Gambar 4.11 Sequence Melakukan Batal Jadwal Rapat

Deskripsi
Sequence diagram di atas menggambarkan proses saat Sub Koordinator membatalkan jadwal rapat yang belum dimulai dan sistem memproses pembatalan data tersebut 1. Sub Koordinator masuk ke halaman jadwal rapat 2. PenjadwalanView mengirim permintaan index() kepada PenjadwalanController

3. PenjadwalanController mengambil data jadwal menggunakan fungsi paginate() pada PenjadwalanModel
4. PenjadwalanModel mengirimkan data jadwal kepada PenjadwalanController
5. PenjadwalanController meneruskan data tersebut kepada PenjadwalanView untuk ditampilkan sebagai daftar jadwal rapat
6. Sub Koordinator menekan tombol batalkan jadwal
7. PenjadwalanView menjalankan fungsi bukaBatalkanJadwal() secara internal untuk menyiapkan form konfirmasi pembatalan
8. Sub Koordinator mengisi form batalkan jadwal
9. Sub Koordinator menekan tombol konfirmasi batalkan
10. PenjadwalanView mengirim permintaan batalkan() kepada PenjadwalanController
11. PenjadwalanController menjalankan validasi alasan pembatalan secara internal untuk memastikan alasan sudah diisi
12. PenjadwalanController mengambil data jadwal menggunakan fungsi findOrFail() pada PenjadwalanModel
13. PenjadwalanModel mengirimkan data jadwal kepada PenjadwalanController
14. PenjadwalanController menjalankan fungsi batalkan() secara internal untuk memproses pembatalan jadwal
15. PenjadwalanController menjalankan fungsi hapusMeetingZoomJikaAda() secara internal untuk menghapus meeting Zoom melalui Zoom API, jika jadwal ini memiliki link Zoom otomatis
16. PenjadwalanController memperbarui status jadwal menjadi "Dibatalkan" menggunakan fungsi update() pada PenjadwalanModel
17. PenjadwalanModel mengirimkan status data kepada PenjadwalanController
18. PenjadwalanController menjalankan fungsi kirimNotifikasiWA() secara internal untuk mengirimkan notifikasi pembatalan melalui WhatsApp kepada operator terkait
19. PenjadwalanController meneruskan status tersebut kepada PenjadwalanView untuk ditampilkan kepada Sub Koordinator

Tabel 4.26 Sequence Melihat Peralatan

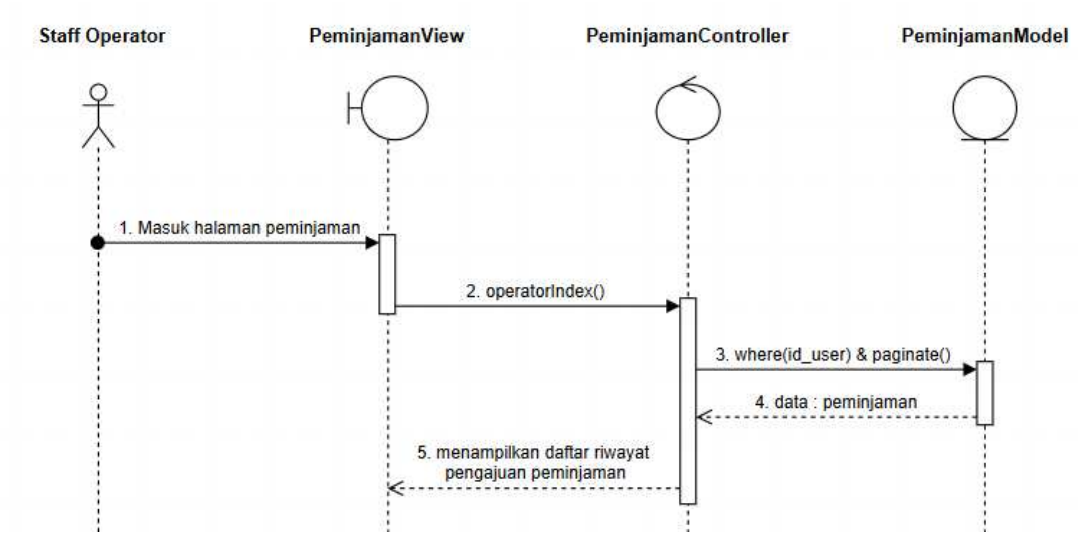
Melihat Peralatan



Gambar 4.12 Sequence Melihat Peralatan

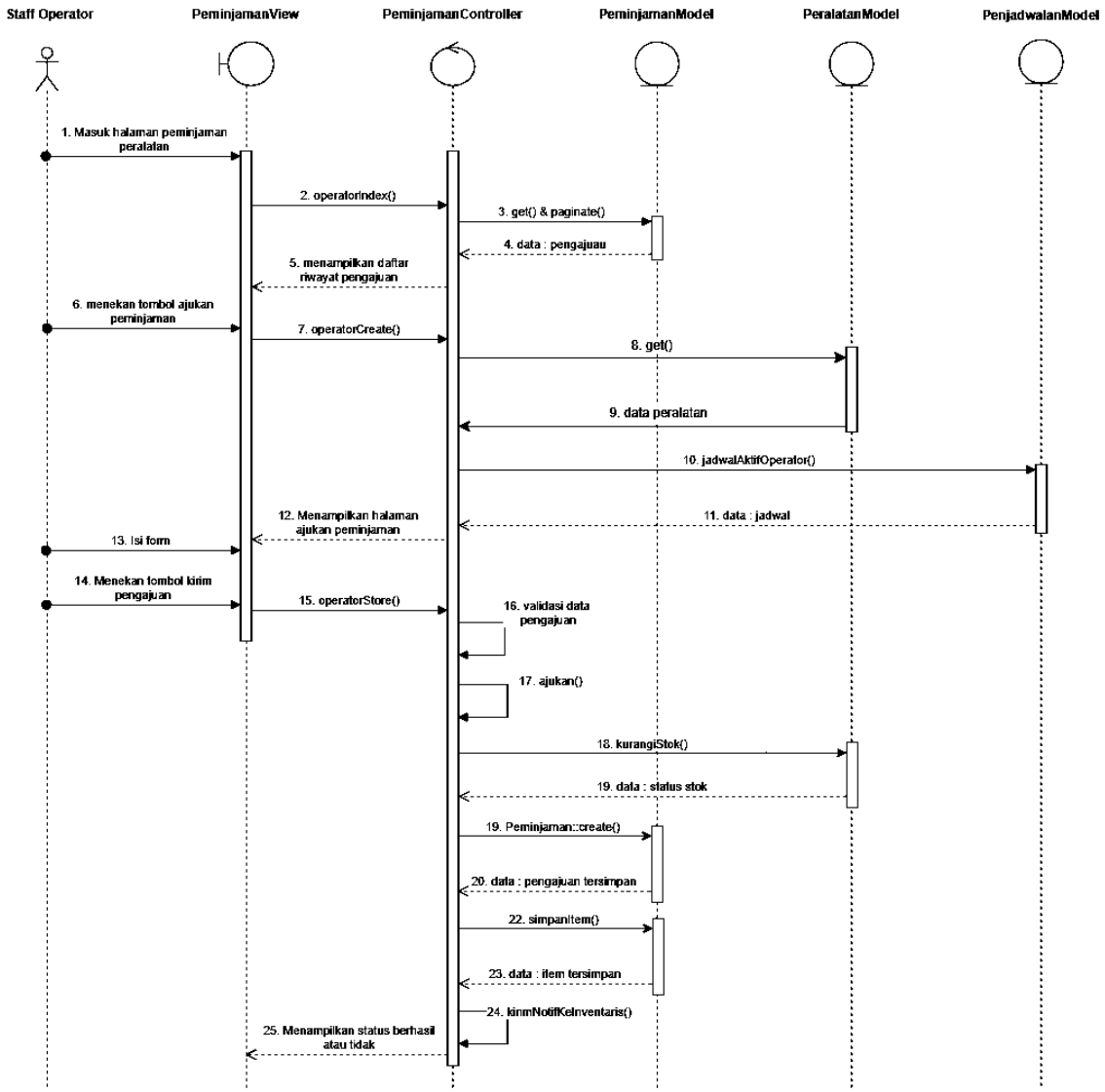
Deskripsi
Sequence diagram di atas menggambarkan proses saat Staff Operator membuka halaman peralatan dan sistem menampilkan daftar informasi peralatan 1. Staff Operator masuk ke halaman peralatan 2. PeralatanView mengirim permintaan index() kepada PeralatanController 3. PeralatanController mengambil daftar peralatan menggunakan fungsi get() & paginate() pada PeralatanModel, dengan filter pencarian, gedung, status, dan kondisi apabila diisi 4. PeralatanModel mengirimkan data peralatan kepada PeralatanController 5. PeralatanController mengambil daftar nama gedung yang tersedia menggunakan fungsi distinct(), orderBy(), dan pluck() pada PeralatanModel 6. PeralatanModel mengirimkan daftar gedung kepada PeralatanController 7. PeralatanController meneruskan seluruh data tersebut kepada PeralatanView untuk ditampilkan sebagai daftar peralatan

Tabel 4.27 Sequence Melihat Peminjaman Peralatan

Melihat Peminjaman Peralatan
 *Gambar 4.13 Sequence Melihat Peminjaman Peralatan*
Deskripsi
Sequence diagram di atas menggambarkan proses saat Staff Operator membuka halaman peminjaman peralatan dan sistem menampilkan daftar informasi peminjaman peralatan 1. Staff Operator masuk ke halaman peminjaman peralatan 2. PeminjamanView mengirim permintaan operatorIndex() kepada PeminjamanController 3. PeminjamanController mengambil data pengajuan peminjaman milik operator yang login menggunakan fungsi where(id_user) dan paginate() pada PeminjamanModel 4. PeminjamanModel mengirimkan data pengajuan peminjaman kepada PeminjamanController 5. PeminjamanController meneruskan data tersebut kepada PeminjamanView untuk ditampilkan sebagai daftar riwayat pengajuan peminjaman

Tabel 4.28 Sequence Melakukan Peminjaman Peralatan

Melakukan Peminjaman Peralatan



Gambar 4.14 Sequence Melakukan Peminjaman Peralatan

Deskripsi

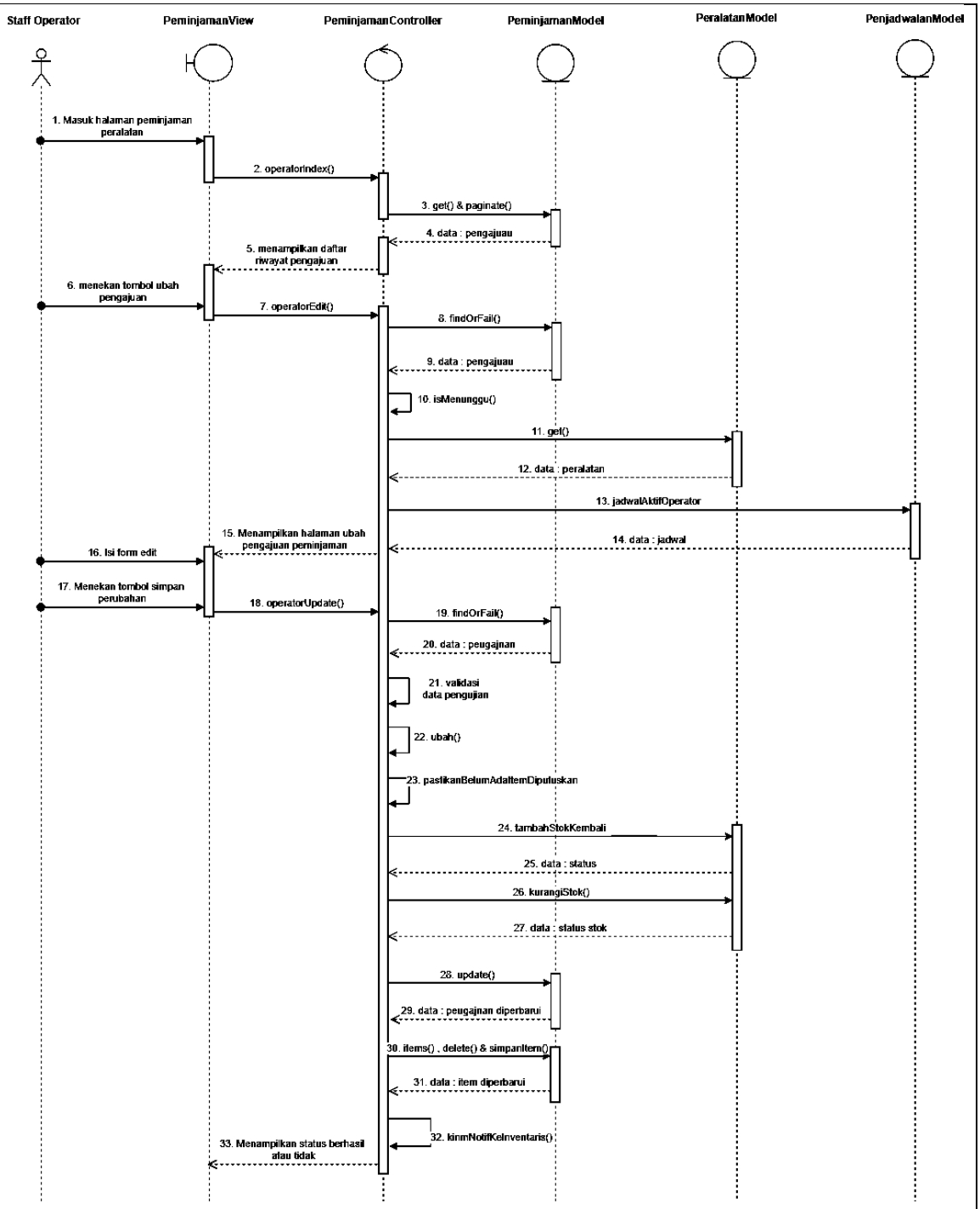
Sequence diagram di atas menggambarkan proses pengajuan peminjaman peralatan yang dilakukan oleh Staff Operator hingga data pengajuan berhasil disimpan oleh sistem

1. Staff Operator masuk ke halaman peminjaman peralatan
2. PeminjamanView mengirim permintaan operatorIndex() kepada PeminjamanController
3. PeminjamanController mengambil daftar pengajuan milik operator yang login menggunakan fungsi where(id_user) dan paginate() pada PeminjamanModel
4. PeminjamanModel mengirimkan data pengajuan kepada PeminjamanController
5. PeminjamanController meneruskan data tersebut kepada PeminjamanView untuk ditampilkan sebagai daftar riwayat pengajuan

6. Staff Operator menekan tombol ajukan peminjaman 7. PeminjamanView mengirim permintaan operatorCreate() kepada PeminjamanController 8. PeminjamanController mengambil daftar peralatan yang tersedia menggunakan fungsi get() pada PeralatanModel 9. PeralatanModel mengirimkan data peralatan kepada PeminjamanController 10. PeminjamanController menjalankan fungsi internal jadwalAktifOperator(), yang mengambil daftar jadwal aktif milik operator dengan melakukan query pada PenjadwalanModel 11. PenjadwalanModel mengirimkan data jadwal kepada PeminjamanController 12. PeminjamanController meneruskan seluruh data tersebut kepada PeminjamanView untuk ditampilkan sebagai halaman ajukan peminjaman 13. Staff Operator mengisi form pengajuan peminjaman 14. Staff Operator menekan tombol kirim pengajuan 15. PeminjamanView mengirim permintaan operatorStore() kepada PeminjamanController 16. PeminjamanController memvalidasi data pengajuan yang diinputkan secara internal 17. PeminjamanController menjalankan fungsi ajukan() secara internal untuk memproses pengajuan peminjaman 18. PeminjamanController mengurangi stok peralatan yang diajukan menggunakan fungsi kurangiStok() pada PeralatanModel 19. PeralatanModel mengirimkan status stok kepada PeminjamanController 20. PeminjamanController menyimpan data pengajuan menggunakan fungsi Peminjaman::create() pada PeminjamanModel 21. PeminjamanModel mengirimkan data pengajuan yang telah tersimpan kepada PeminjamanController 22. PeminjamanController menyimpan daftar peralatan yang diajukan menggunakan fungsi simpanItem() pada PeminjamanModel 23. PeminjamanModel mengirimkan data item yang telah tersimpan kepada PeminjamanController 24. PeminjamanController menjalankan fungsi kirimNotifKeInventaris() secara internal untuk mengirimkan notifikasi WhatsApp kepada petugas inventaris 25. PeminjamanController meneruskan status keberhasilan kepada PeminjamanView untuk ditampilkan kepada Staff Operator

Tabel 4.29 Sequence Melakukan Ubah Peminjaman Peralatan

Melakukan Ubah Peminjaman Peralatan



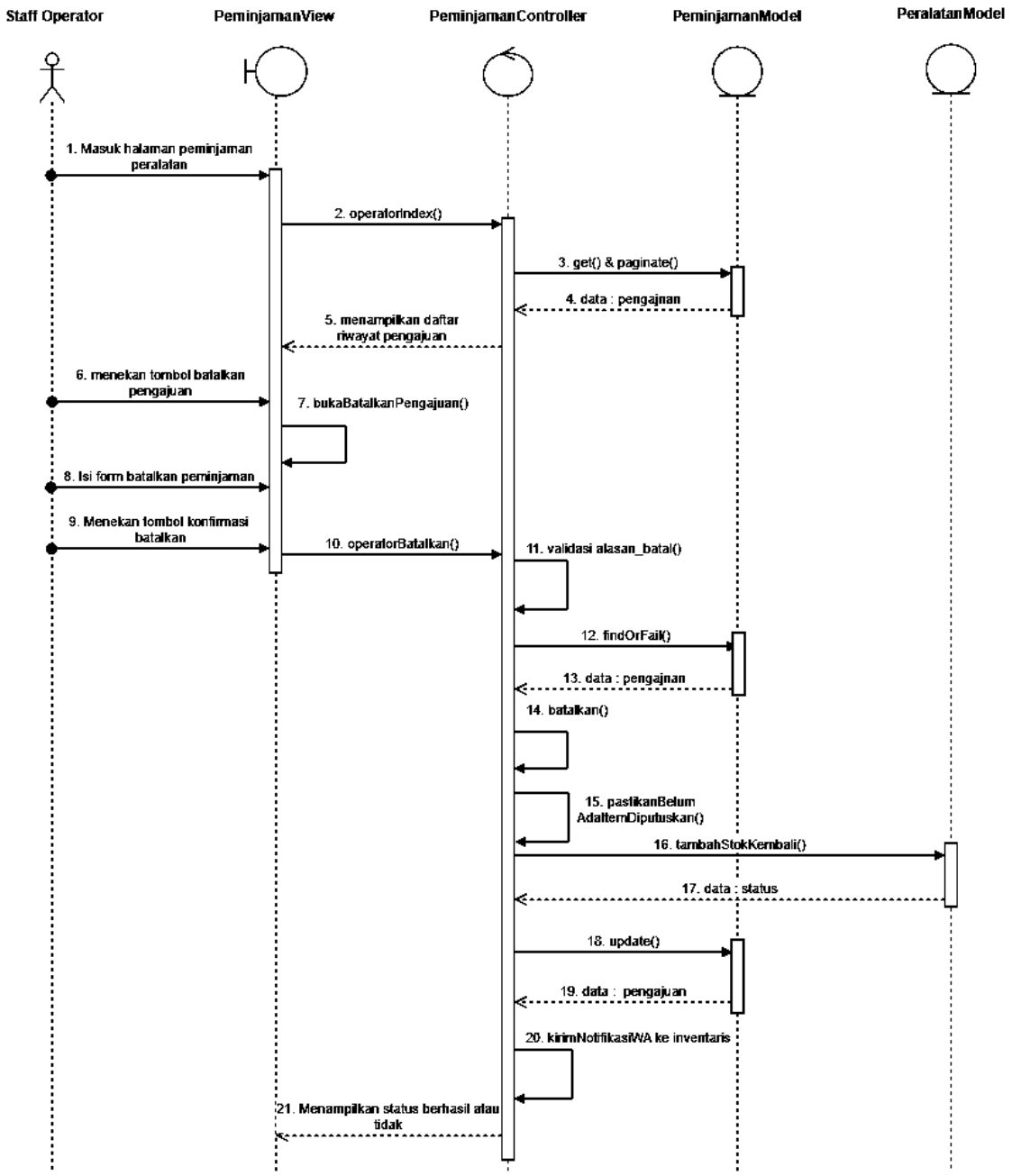
Gambar 4.15 Sequence Melakukan Ubah Peminjaman Peralatan

Deskripsi
Sequence diagram di atas menggambarkan proses perubahan pengajuan peminjaman peralatan yang dilakukan oleh Staff Operator hingga data pengajuan berhasil disimpan oleh sistem

1. Staff Operator masuk ke halaman peminjaman peralatan
2. PeminjamanView mengirim permintaan operatorIndex() kepada PeminjamanController
3. PeminjamanController mengambil daftar pengajuan milik operator yang login menggunakan fungsi where(id_user) dan paginate() pada PeminjamanModel
4. PeminjamanModel mengirimkan data pengajuan kepada PeminjamanController
5. PeminjamanController meneruskan data tersebut kepada PeminjamanView untuk ditampilkan sebagai daftar riwayat pengajuan
6. Staff Operator menekan tombol ubah pengajuan
7. PeminjamanView mengirim permintaan operatorEdit() kepada PeminjamanController
8. PeminjamanController mengambil data pengajuan menggunakan fungsi findOrFail() pada PeminjamanModel
9. PeminjamanModel mengirimkan data pengajuan kepada PeminjamanController
10. PeminjamanController memeriksa status pengajuan menggunakan fungsi isMenunggu() secara internal hanya pengajuan berstatus "menunggu" yang dapat diubah
11. PeminjamanController mengambil daftar peralatan yang tersedia menggunakan fungsi get() pada PeralatanModel
12. PeralatanModel mengirimkan data peralatan kepada PeminjamanController
13. PeminjamanController menjalankan fungsi internal jadwalAktifOperator(), yang mengambil daftar jadwal aktif milik operator dengan melakukan query pada PenjadwalanModel
14. PenjadwalanModel mengirimkan data jadwal kepada PeminjamanController
15. PeminjamanController meneruskan seluruh data tersebut kepada PeminjamanView untuk ditampilkan sebagai halaman ubah pengajuan peminjaman
16. Staff Operator mengubah isi form
17. Staff Operator menekan tombol simpan perubahan
18. PeminjamanView mengirim permintaan operatorUpdate() kepada PeminjamanController
19. PeminjamanController mengambil ulang data pengajuan menggunakan fungsi findOrFail() pada PeminjamanModel
20. PeminjamanModel mengirimkan data pengajuan kepada PeminjamanController
21. PeminjamanController memvalidasi data perubahan yang diinputkan secara internal
22. PeminjamanController menjalankan fungsi ubah() secara internal untuk memproses perubahan pengajuan
23. PeminjamanController memeriksa syarat perubahan menggunakan fungsi pastikanBelumAdaItemDiputuskan() secara internal pengajuan tidak bisa diubah kalau sudah ada alat yang diputuskan inventaris
24. PeminjamanController mengembalikan stok peralatan dari daftar lama menggunakan fungsi tambahStokKembali() pada PeralatanModel
25. PeralatanModel mengirimkan status kepada PeminjamanController
26. PeminjamanController mengurangi stok peralatan yang baru dipilih menggunakan fungsi kurangiStok() pada PeralatanModel
27. PeralatanModel mengirimkan status stok kepada PeminjamanController
28. PeminjamanController memperbarui data pengajuan menggunakan fungsi update() pada PeminjamanModel
29. PeminjamanModel mengirimkan data pengajuan yang telah diperbarui kepada PeminjamanController
30. PeminjamanController menghapus daftar peralatan lama lalu menyimpan daftar peralatan baru menggunakan fungsi items()->delete() dan simpanItem() pada PeminjamanModel
31. PeminjamanModel mengirimkan data item yang telah diperbarui kepada PeminjamanController
32. PeminjamanController menjalankan fungsi kirimNotifKeInventaris() secara internal untuk mengirimkan notifikasi WhatsApp kepada petugas inventaris
33. PeminjamanController meneruskan status keberhasilan kepada PeminjamanView untuk ditampilkan kepada Staff Operator

Tabel 4.30 Sequence Melakukan Peminjaman Peralatan

Melakukan Peminjaman Peralatan



Gambar 4.16 Sequence Melakukan Batal Peminjaman Peralatan

Deskripsi

Sequence diagram di atas menggambarkan proses pembatalan pengajuan peminjaman peralatan yang dilakukan oleh Staff Operator hingga status pengajuan berhasil diperbarui oleh sistem

1. Staff Operator masuk ke halaman peminjaman peralatan
2. PeminjamanView mengirim permintaan operatorIndex() kepada PeminjamanController
3. PeminjamanController mengambil daftar pengajuan milik operator yang login menggunakan fungsi where(id_user) dan paginate() pada PeminjamanModel

4. PeminjamanModel mengirimkan data pengajuan kepada PeminjamanController
5. PeminjamanController meneruskan data tersebut kepada PeminjamanView untuk ditampilkan sebagai daftar riwayat pengajuan
6. Staff Operator menekan tombol batalkan pengajuan
7. PeminjamanView menjalankan fungsi bukaBatalkanPengajuan() secara internal untuk menyiapkan form konfirmasi pembatalan
8. Staff Operator mengisi form batalkan peminjaman
9. Staff Operator menekan tombol konfirmasi batalkan
10. PeminjamanView mengirim permintaan operatorBatalkan() kepada PeminjamanController
11. PeminjamanController menjalankan validasi alasan pembatalan secara internal untuk memastikan alasan sudah diisi
12. PeminjamanController mengambil data pengajuan menggunakan fungsi findOrFail() pada PeminjamanModel
13. PeminjamanModel mengirimkan data pengajuan kepada PeminjamanController
14. PeminjamanController menjalankan fungsi batalkan() secara internal untuk memproses pembatalan pengajuan
15. PeminjamanController memeriksa syarat pembatalan menggunakan fungsi pastikanBelumAdaItemDiputuskan() secara internal pengajuan tidak bisa dibatalkan kalau sudah ada alat yang diputuskan inventaris
16. PeminjamanController mengembalikan stok peralatan yang direservasi menggunakan fungsi tambahStokKembali() pada PeralatanModel
17. PeralatanModel mengirimkan status kepada PeminjamanController
18. PeminjamanController memperbarui status pengajuan menjadi "Dibatalkan" menggunakan fungsi update() pada PeminjamanModel
19. PeminjamanModel mengirimkan data pengajuan kepada PeminjamanController
20. PeminjamanController menjalankan fungsi kirimNotifikasiWA() secara internal untuk mengirimkan notifikasi pembatalan melalui WhatsApp kepada petugas inventaris
21. PeminjamanController meneruskan status keberhasilan kepada PeminjamanView untuk ditampilkan kepada Staff Operator

4.4 Pemodelan Berbasis Kelas

Pemodelan berbasis kelas digunakan untuk menggambarkan struktur objek dalam sistem yang dirancang, meliputi atribut, operasi, serta hubungan antar kelas yang saling berinteraksi sehingga dapat memberikan gambaran mengenai struktur sistem secara keseluruhan.

4.4.1 Identifikasi Kelas-kelas Analisis

Pada tahap ini dilakukan identifikasi kelas-kelas yang digunakan dalam sistem yang dirancang. Identifikasi dilakukan berdasarkan kebutuhan sistem serta proses yang terjadi, sehingga diperoleh kelas-kelas yang merepresentasikan objek dalam sistem. Kelas-kelas tersebut kemudian digunakan sebagai dasar dalam pembuatan class diagram untuk menggambarkan struktur sistem.

Tabel 4.31 Identifikasi Kelas

No	Nama Kelas	Jenis	Deskripsi
1.	User	Model	Kelas ini digunakan untuk berinteraksi dengan tabel users pada database

2.	Penjadwalan	Model	Kelas ini digunakan untuk berinteraksi dengan tabel penjadwalan pada database
3.	Peralatan	Model	Kelas ini digunakan untuk berinteraksi dengan tabel peralatan pada database
4.	Peminjaman	Model	Kelas ini digunakan untuk berinteraksi dengan tabel peminjaman pada database
5.	PeminjamanItem	Model	Kelas ini digunakan untuk berinteraksi dengan tabel peminjaman_item pada database
6.	AdminController	Controller	Kelas ini digunakan untuk menampilkan dashboard admin dan laporan
7.	UserController	Controller	Kelas ini digunakan untuk mengelola proses data pengguna
8.	PenjadwalanController	Controller	Kelas ini digunakan untuk mengelola proses jadwal rapat
9.	OperatorController	Controller	Kelas ini digunakan untuk menampilkan dashboard operator
10.	PeralatanController	Controller	Kelas ini digunakan untuk menampilkan peralatan
11.	PeminjamanController	Controller	Kelas ini digunakan untuk mengelola proses pengajuan, perubahan, dan pembatalan peminjaman peralatan
12.	DashboardAdmin	View	Kelas ini digunakan untuk menampilkan halaman dashboard admin
13.	DashboardOperator	View	Kelas ini digunakan untuk menampilkan halaman dashboard operator
14.	Users	View	Kelas ini digunakan untuk menampilkan halaman pengguna
15.	JadwalRapat	View	Kelas ini digunakan untuk menampilkan halaman jadwal rapat secara keseluruhan
16.	Peralatan	View	Kelas ini digunakan untuk menampilkan halaman peralatan
17.	PeminjamanOperator	View	Kelas ini digunakan untuk menampilkan halaman peminjaman peralatan
18.	Laporan	View	Kelas ini digunakan untuk menampilkan halaman laporan

4.4.2 Menentukan Atribut

Atribut merupakan karakteristik yang dimiliki oleh setiap kelas yang digunakan dalam sistem. Atribut digunakan untuk menyimpan data yang berkaitan dengan kelas tersebut.

Pada aplikasi yang dirancang, setiap kelas memiliki atribut yang berbeda sesuai dengan kebutuhan dan fungsi masing-masing kelas. Atribut-atribut ini digunakan untuk mendukung proses pengolahan data dalam sistem.

Tabel 4.32 Atribut

No	Nama Kelas	Nama Atribut
1.	User	id_user
		nama_user
		jenis_kelamin
		alamat
		nohp
		email
		password
		role
		status
		remember_token
2.	Penjadwalan	id_penjadwalan

		judul_kegiatan
		tanggal
		waktu_mulai
		waktu_selesai
		platform
		keterangan
		lokasi_fisik
		zoom_meeting_id
		zoom_password
		zoom_account
		link_otomatis
		status
		alasan_batal
		dibatalkan_at
3.	Peralatan	id_peralatan
		kode_barang
		nama_peralatan
		gedung
		lokasi_detail
		stok
		rusak
		keterangan
		foto
4.	Peminjaman	id_peminjaman
		id_user
		id_penjadwalan
		tanggal_pinjam
		tanggal_kembali_rencana
		tanggal_kembali_aktual
		keperluan
		status
		catatan_inventaris
		alasan_batal
		dibatalkan_at
5.	PeminjamanItem	id_item
		id_peminjaman
		id_peralatan
		jumlah
		status
6.	AdminController	User
		Penjadwalan
		Peminjaman
		PeminjamanItem

7.	UserController	User
		Role
8.	PenjadwalanController	Penjadwalan
		User
		Peralatan
9.	OperatorController	Peminjaman
		Penjadwalan
		PeminjamanItem
10.	PeralatanController	Peralatan
11.	PeminjamanController	Peminjaman
		Penjadwalan
		Peralatan
		PeminjamanItem

4.4.3 Menentukan Perilaku

Perilaku kelas merupakan fungsi atau operasi yang dimiliki oleh setiap kelas untuk menjalankan proses sesuai dengan tanggung jawabnya dalam sistem. Perilaku kelas digunakan untuk menggambarkan aktivitas atau operasi yang dapat dilakukan oleh objek pada proses pengolahan data dan pengelolaan fitur sistem. Berikut merupakan perilaku kelas yang digunakan pada aplikasi penjadwalan rapat:

Tabel 4.33 Perilaku

No	Nama Kelas	Deskripsi	Perilaku
1.	User	Mengambil relasi jadwal yang ditugaskan kepada operator	jadwalDitugaskan()
		Mengambil relasi data peminjaman pengguna	peminjaman()
		Memeriksa status aktif pengguna	isActive()
		Menghasilkan nomor WhatsApp pengguna	getNomorWaAttribute()
2.	Penjadwalan	Mengambil relasi operator yang bertugas	operators()
		Mengambil relasi data peminjaman	peminjaman()
		Mengambil relasi data peralatan referensi	peralatanReferensi()
		Menghitung jumlah tiap peralatan yang sudah diajukan untuk jadwal ini	peralatanSudahDiajukan()
		Memeriksa status pembatalan jadwal	isDibatalkan()
		Menghasilkan tanggal dan waktu mulai kegiatan	getStartDateTimeAttribute()
		Menghasilkan tanggal dan waktu selesai kegiatan	getEndDateTimeAttribute()
		Memeriksa bentrok jadwal operator	scopeBentrok()
3.	Peralatan	Menghitung stok peralatan yang tersedia	getStokTersediaAttribute()
		Menentukan label status peralatan	getStatusLabelAttribute()
		Menentukan kelas badge status peralatan	getStatusBadgeClassAttribute()
		Menghasilkan URL foto peralatan	getFotoUrlAttribute()
4.	Peminjaman	Mengambil relasi data pengguna	user()

		Mengambil relasi data jadwal rapat	penjadwalan()
		Mengambil relasi item peminjaman	items()
		Memeriksa status menunggu	isMenunggu()
		Memeriksa status disetujui	isDisetujui()
		Memeriksa status dibatalkan	isDibatalkan()
		Menghasilkan badge status peminjaman	getBadgeAttribute()
5.	PeminjamanItem	Mengambil relasi data peminjaman	peminjaman()
		Mengambil relasi data peralatan	peralatan()
		Memeriksa status menunggu item	isMenunggu()
		Menghasilkan badge status item peminjaman	getBadgeAttribute()
6.	AdminController	Menampilkan halaman Dashboard Admin	dashboard()
		Menampilkan halaman laporan	laporanIndex()
		Mengekspor laporan ke format PDF	laporanExportPdf()
		Mengekspor laporan ke format Excel	laporanExportExcel()
7.	UserController	Menampilkan data pengguna	index()
		Menampilkan halaman tambah pengguna	create()
		Menyimpan data pengguna	store()
		Menampilkan halaman ubah pengguna	edit()
		Memperbarui data pengguna	update()
		Mengubah status pengguna	destroy()
8.	PenjadwalanController	Menampilkan data jadwal rapat	index()
		Menampilkan halaman tambah jadwal rapat	create()
		Menyimpan data jadwal rapat	store()
		Menampilkan detail jadwal rapat	show()
		Menampilkan halaman ubah jadwal rapat	edit()
		Memperbarui data jadwal rapat	update()
		Membatalkan jadwal rapat	batalkan()
9.	OperatorController	Menampilkan halaman Dashboard Operator	dashboard()
10.	PeminjamanController	Menampilkan data peminjaman operator	operatorIndex()
		Menampilkan halaman pengajuan peminjaman	operatorCreate()
		Menyimpan pengajuan peminjaman	operatorStore()
		Memeriksa duplikasi pengajuan peralatan	operatorCekSpam()
		Menampilkan halaman ubah pengajuan peminjaman	operatorEdit()
		Memperbarui pengajuan peminjaman	operatorUpdate()
		Membatalkan pengajuan peminjaman	operatorBatalkan()
11.	PeralatanController	Menampilkan data peralatan	index()

4.5 Perancangan Perangkat Lunak

Perancangan perangkat lunak merupakan tahapan dalam proses pengembangan sistem yang bertujuan untuk menerapkan konsep, metode, serta prinsip perancangan ke dalam bentuk rancangan yang lebih terstruktur. Tahapan ini berfokus pada penerjemahan kebutuhan sistem yang telah dianalisis sebelumnya ke dalam bentuk desain yang dapat diimplementasikan.

Pada penelitian ini, perancangan perangkat lunak meliputi perancangan data dan perancangan antarmuka yang saling berkaitan untuk mendukung pembangunan sistem secara menyeluruh.

4.5.1 Perancangan Data

Perancangan data dilakukan untuk memodelkan struktur data yang digunakan dalam sistem, meliputi objek data, atribut, dan relasi antar objek. Hasil perancangan disajikan dalam bentuk ERD (***Entity Relationship Diagram***) dan tabel perancangan data.

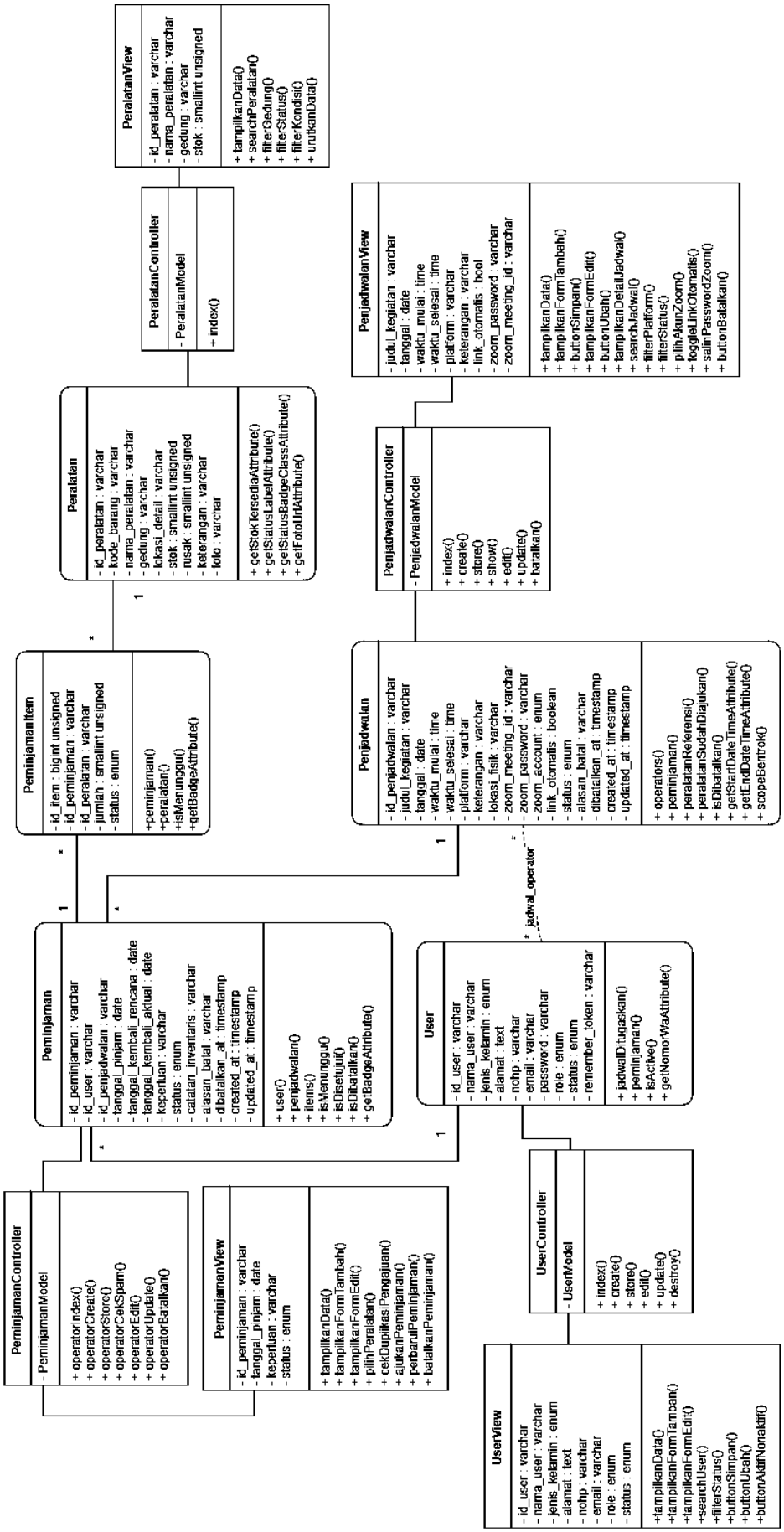
Tabel 4.34 Perancangan Data

No	Nama Objek	Atribut	Tipe Data
1.	User	id_user	Varchar
		nama_user	Varchar
		jenis_kelamin	Enum
		alamat	Text
		nohp	Varchar
		email	Varchar
		password	Varchar
		role	Varchar
		status	Enum
		remember_token	Varchar
2.	Penjadwalan	id_penjadwalan	Varchar
		judul_kegiatan	Varchar
		tanggal	Date
		waktu_mulai	Time
		waktu_selesai	Time
		platform	Varchar
		keterangan	Varchar
		lokasi_fisik	Varchar
		zoom_meeting_id	Varchar
		zoom_password	Varchar
		zoom_account	Enum
		link_otomatis	Tinyint
		status	Enum
		alasan_batal	Varchar
		dibatalkan_at	Timestamp
		created_at	Timestamp
		updated_at	Timestamp
3.	Peralatan	id_peralatan	Varchar
		kode_barang	Varchar
		nama_peralatan	Varchar

		gedung	Varchar
		lokasi_detail	Varchar
		stok	Smallint Unsigned
		rusak	Smallint Unsigned
		keterangan	Varchar
		foto	Varchar
4.	Peminjaman	id_peminjaman	Varchar
		id_user	Varchar
		id_penjadwalan	Varchar
		tanggal_pinjam	Date
		tanggal_kembali_rencana	Date
		tanggal_kembali_aktual	Date
		keperluan	Varchar
		status	Enum
		catatan_inventaris	Varchar
		alasan_batal	Varchar
		dibatalkan_at	Timestamp
		created_at	Timestamp
		updated_at	Timestamp
5.	PeminjamanItem	id_item	Bigint Unsigned
		id_peminjaman	Varchar
		id_peralatan	Varchar
		jumlah	Smallint Unsigned
		status	Enum

4.5.2 Perancangan Kelas

Perancangan kelas merupakan tahap untuk menyusun diagram kelas berdasarkan kelas-kelas yang telah diidentifikasi sebelumnya, termasuk atribut dan operasinya. Diagram kelas yang dihasilkan ditampilkan pada gambar berikut.



Gambar 4.17 Perancangan Kelas

4.5.3 Objek Data

Objek data merupakan kumpulan elemen data yang saling berkaitan dan digunakan untuk merepresentasikan informasi tertentu dalam sebuah aplikasi. Objek data digunakan sebagai media penyimpanan dan pengelolaan data yang dibutuhkan oleh sistem sesuai dengan proses bisnis yang berjalan.

Tabel 4.35 Objek Data

No	Nama Objek	Deskripsi
1.	User	Objek ini digunakan untuk mengelola data pengguna yang mengakses aplikasi sesuai dengan hak akses yang dimiliki
2.	Penjadwalan	Objek ini digunakan untuk mengelola data jadwal kegiatan rapat, termasuk informasi pelaksanaan rapat dan pembuatan tautan Zoom secara otomatis apabila diperlukan
3.	Peralatan	Objek ini digunakan untuk melihat data peralatan yang digunakan sebagai pendukung kegiatan rapat maupun di luar rapat
4.	Peminjaman	Objek ini digunakan untuk mengelola data pengajuan peminjaman peralatan yang digunakan dalam kegiatan rapat maupun di luar rapat
5.	PeminjamanItem	Objek ini digunakan untuk mengelola rincian peralatan yang diajukan pada setiap transaksi peminjaman

4.5.4 Atribut Data

Atribut data merupakan karakteristik atau informasi yang dimiliki oleh suatu objek data. Atribut digunakan untuk menjelaskan detail data yang disimpan pada setiap objek dalam sistem. Daftar atribut dari masing-masing objek data yang telah dirancang dapat dilihat pada tabel berikut.

Tabel 4.36 Atribut Data

No	Nama Objek	Atribut	Deskripsi
1.	User	id_user	Atribut ini digunakan untuk menyimpan identitas pengguna pada sistem
		nama_user	Atribut ini digunakan untuk menyimpan nama pengguna
		jenis_kelamin	Atribut ini digunakan untuk menyimpan jenis kelamin pengguna
		alamat	Atribut ini digunakan untuk menyimpan alamat pengguna
		nohp	Atribut ini digunakan untuk menyimpan nomor telepon pengguna
		email	Atribut ini digunakan untuk menyimpan alamat email pengguna
		password	Atribut ini digunakan untuk menyimpan kata sandi pengguna
		role	Atribut ini digunakan untuk menyimpan hak akses pengguna
		status	Atribut ini digunakan untuk menyimpan status pengguna
		remember_token	Atribut ini digunakan untuk menyimpan token autentikasi pengguna
2.	Penjadwalan	id_penjadwalan	Atribut ini digunakan untuk menyimpan identitas jadwal kegiatan
		judul_kegiatan	Atribut ini digunakan untuk menyimpan judul kegiatan rapat
		tanggal	Atribut ini digunakan untuk menyimpan tanggal kegiatan rapat
		waktu_mulai	Atribut ini digunakan untuk menyimpan waktu mulai kegiatan rapat

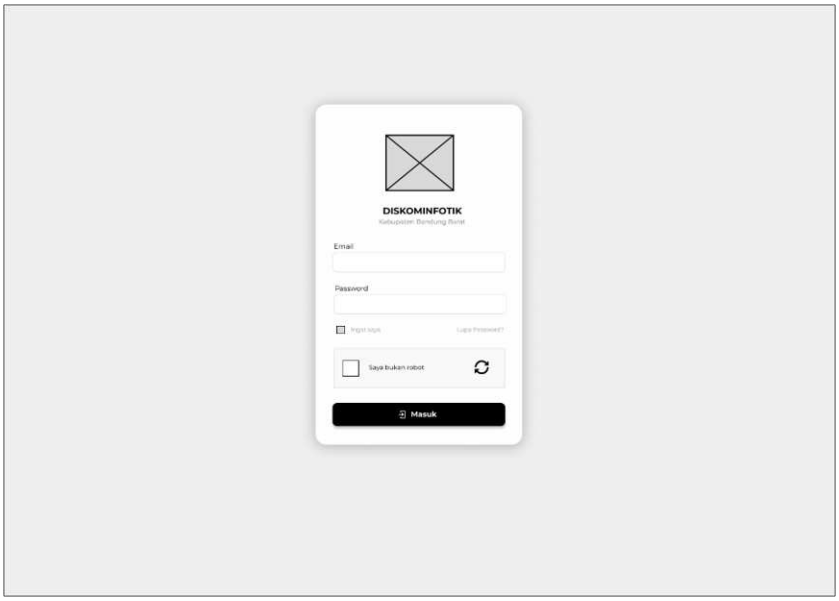
		waktu_selesai	Atribut ini digunakan untuk menyimpan waktu selesai kegiatan rapat
		platform	Atribut ini digunakan untuk menyimpan platform pelaksanaan rapat
		keterangan	Atribut ini digunakan untuk menyimpan informasi lokasi atau keterangan kegiatan
		lokasi_fisik	Atribut ini digunakan untuk menyimpan lokasi fisik pelaksanaan rapat, khusus untuk platform Hybrid saat kolom keterangan sudah dipakai untuk link meeting
		zoom_meeting_id	Atribut ini digunakan untuk menyimpan identitas meeting Zoom yang dibuat secara otomatis
		zoom_password	Atribut ini digunakan untuk menyimpan kata sandi meeting Zoom
		zoom_account	Atribut ini digunakan untuk menyimpan akun Zoom yang digunakan untuk membuat meeting
		link_otomatis	Atribut ini digunakan untuk menandai apakah link meeting rapat dibuat secara otomatis melalui integrasi Zoom atau diisi secara manual oleh admin
		status	Atribut ini digunakan untuk menyimpan status kegiatan rapat
		alasan_batal	Atribut ini digunakan untuk menyimpan alasan pembatalan kegiatan rapat
		dibatalkan_at	Atribut ini digunakan untuk menyimpan waktu pembatalan kegiatan rapat
		created_at	Atribut ini digunakan untuk menyimpan waktu data jadwal rapat pertama kali dibuat
		updated_at	Atribut ini digunakan untuk menyimpan waktu data jadwal rapat terakhir diperbarui
3.	Peralatan	id_peralatan	Atribut ini digunakan untuk menyimpan identitas peralatan
		kode_barang	Atribut ini digunakan untuk menyimpan kode identitas peralatan
		nama_peralatan	Atribut ini digunakan untuk menyimpan nama peralatan
		gedung	Atribut ini digunakan untuk menyimpan lokasi gedung penyimpanan peralatan
		lokasi_detail	Atribut ini digunakan untuk menyimpan lokasi detail penyimpanan peralatan
		stok	Atribut ini digunakan untuk menyimpan jumlah stok peralatan
		rusak	Atribut ini digunakan untuk menyimpan jumlah peralatan yang rusak
		keterangan	Atribut ini digunakan untuk menyimpan informasi tambahan mengenai peralatan
		foto	Atribut ini digunakan untuk menyimpan foto peralatan
4.	Peminjaman	id_peminjaman	Atribut ini digunakan untuk menyimpan identitas peminjaman
		id_user	Atribut ini digunakan untuk menyimpan identitas pengguna yang mengajukan peminjaman
		id_penjadwalan	Atribut ini digunakan untuk menyimpan identitas jadwal rapat yang berkaitan dengan peminjaman
		tanggal_pinjam	Atribut ini digunakan untuk menyimpan tanggal peminjaman peralatan
		tanggal_kembali_rencana	Atribut ini digunakan untuk menyimpan rencana tanggal pengembalian peralatan
		tanggal_kembali_aktual	Atribut ini digunakan untuk menyimpan tanggal peralatan benar-benar dikembalikan, diisi oleh petugas inventaris saat konfirmasi pengembalian
		keperluan	Atribut ini digunakan untuk menyimpan tujuan peminjaman peralatan
		status	Atribut ini digunakan untuk menyimpan status peminjaman

		catatan_inventaris	Atribut ini digunakan untuk menyimpan catatan dari petugas inventaris terhadap pengajuan peminjaman
		alasan_batal	Atribut ini digunakan untuk menyimpan alasan pembatalan peminjaman
		dibatalkan_at	Atribut ini digunakan untuk menyimpan waktu pembatalan peminjaman
		created_at	Atribut ini digunakan untuk menyimpan waktu data peminjaman pertama kali diajukan
		updated_at	Atribut ini digunakan untuk menyimpan waktu data peminjaman terakhir diperbarui
5.	PeminjamanItem	id_item	Atribut ini digunakan untuk menyimpan identitas item peminjaman
		id_peminjaman	Atribut ini digunakan untuk menyimpan identitas peminjaman
		id_peralatan	Atribut ini digunakan untuk menyimpan identitas peralatan yang dipinjam
		jumlah	Atribut ini digunakan untuk menyimpan jumlah peralatan yang dipinjam
		status	Atribut ini digunakan untuk menyimpan status persetujuan tiap alat dalam satu pengajuan peminjaman

4.6 Perancangan Antarmuka Pengguna

Perancangan antarmuka pengguna merupakan proses penyusunan tampilan aplikasi yang digunakan sebagai media interaksi antara pengguna dan aplikasi. Perancangan ini mencakup penentuan struktur halaman, navigasi, tata letak, serta komponen antarmuka yang akan digunakan. Hasil perancangan disusun dalam bentuk wireframe dan prototype sebagai acuan dalam proses pembangunan aplikasi penjadwalan rapat.

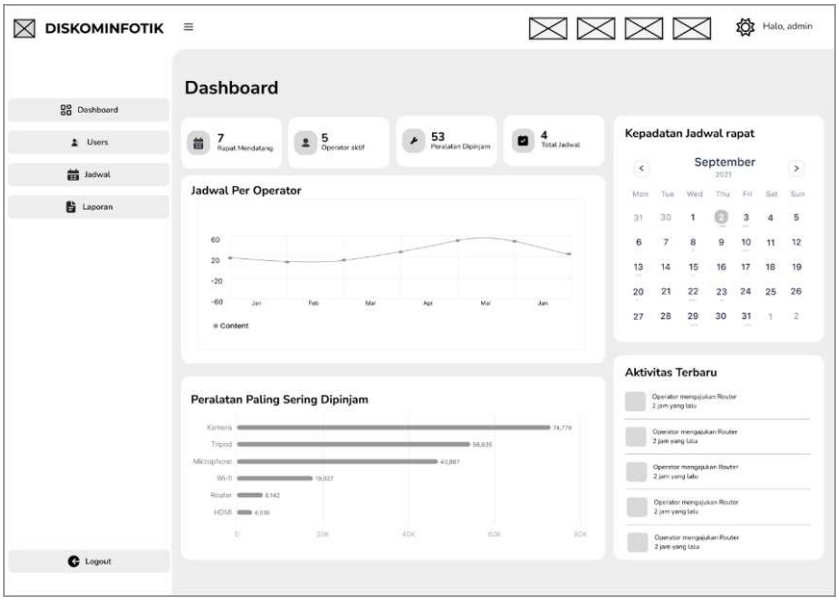
Tabel 4.37 Tampilan Login

Halaman Login


Gambar 4.18 Tampilan Login

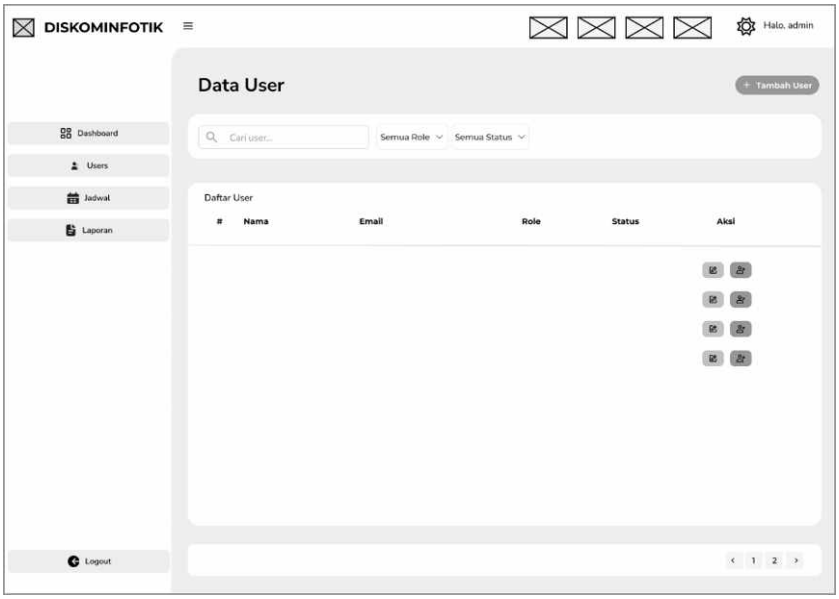
Deskripsi
Ini merupakan rancangan tampilan login yang digunakan sebagai halaman autentikasi pengguna untuk mengakses aplikasi. Pengguna dapat masuk ke dalam aplikasi dengan memasukkan alamat email, kata sandi, ingat password, lupa password dan melakukan verifikasi CAPTCHA menggunakan akun yang telah terdaftar

Tabel 4.38 Tampilan Halaman Dashboard Admin

Halaman Dashboard
 *Gambar 4.19 Tampilan Halaman Dashboard Admin*
Deskripsi
Ini merupakan rancangan tampilan dashboard admin yang digunakan untuk menampilkan informasi jumlah rapat mendatang, jumlah operator aktif, jumlah peralatan yang dipinjam, jumlah rapat selesai, statistik jadwal per operator, kepadatan jadwal rapat, peralatan yang sering dipinjam dan aktivitas terbaru

Tabel 4.39 Tampilan Halaman Users

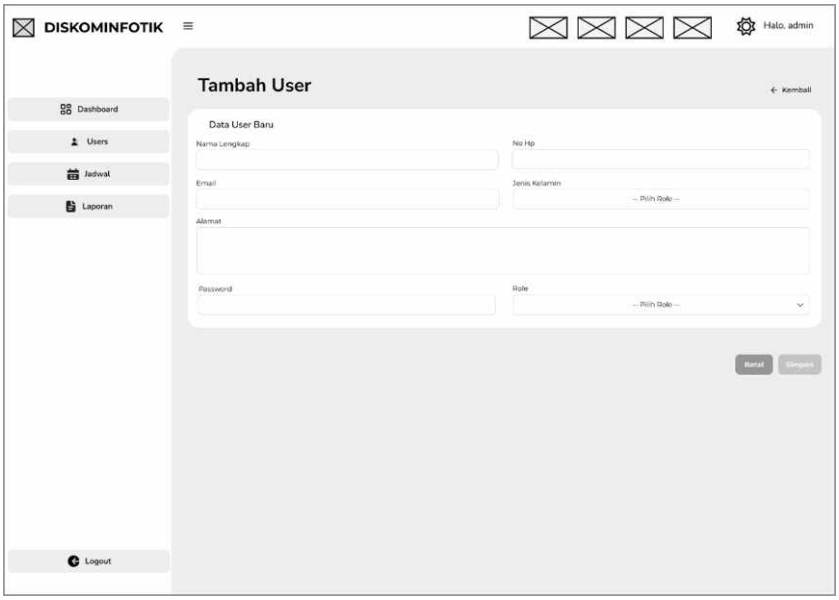
Halaman Users



Gambar 4.20 Tampilan Halaman Users

Deskripsi
Ini merupakan rancangan tampilan data user yang digunakan untuk menampilkan informasi pengguna, melakukan pencarian data pengguna, menambah user, mengubah data user, serta mengaktifkan dan menonaktifkan akun pengguna

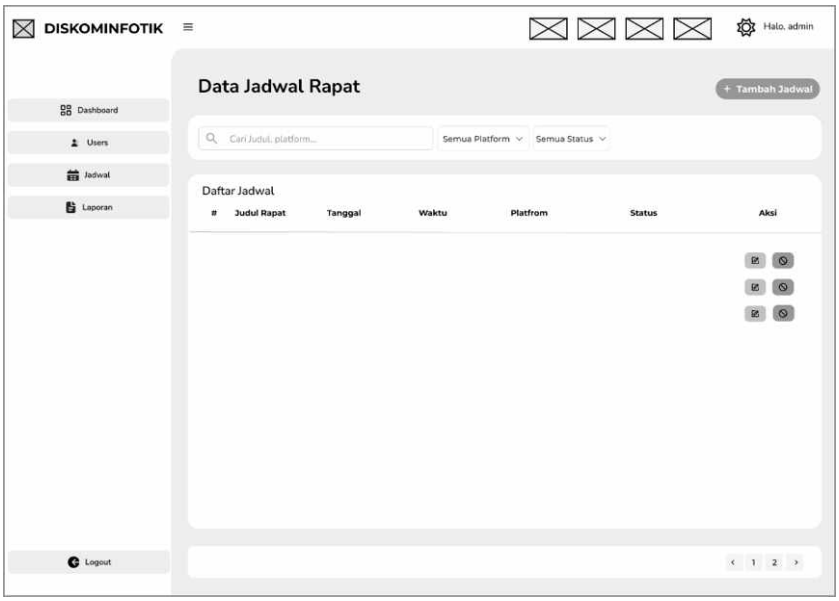
Tabel 4.40 Tampilan Halaman Tambah User

Halaman Tambah User


Gambar 4.21 Tampilan Halaman Tambah User

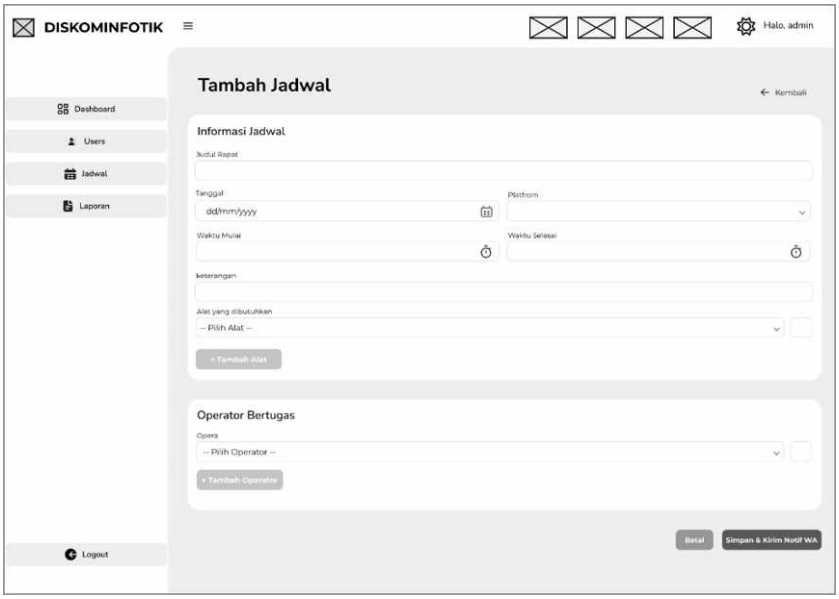
Deskripsi
Ini merupakan rancangan tampilan tambah data user yang digunakan admin untuk menambahkan data pengguna baru dengan mengisi informasi pengguna seperti nama user, nomor telepon, email, jenis kelamin, alamat, password, dan role pengguna

Tabel 4.41 Tampilan Halaman Jadwal

Halaman Jadwal
 *Gambar 4.22 Tampilan Halaman Jadwal*
Deskripsi
Ini merupakan rancangan tampilan data jadwal rapat yang digunakan untuk menampilkan informasi jadwal kegiatan rapat, melakukan pencarian data, menambah jadwal, mengubah jadwal, membatalkan jadwal, dan melihat detail jadwal rapat

Tabel 4.42 Tampilan Halaman Tambah Jadwal

Halaman Tambah Jadwal

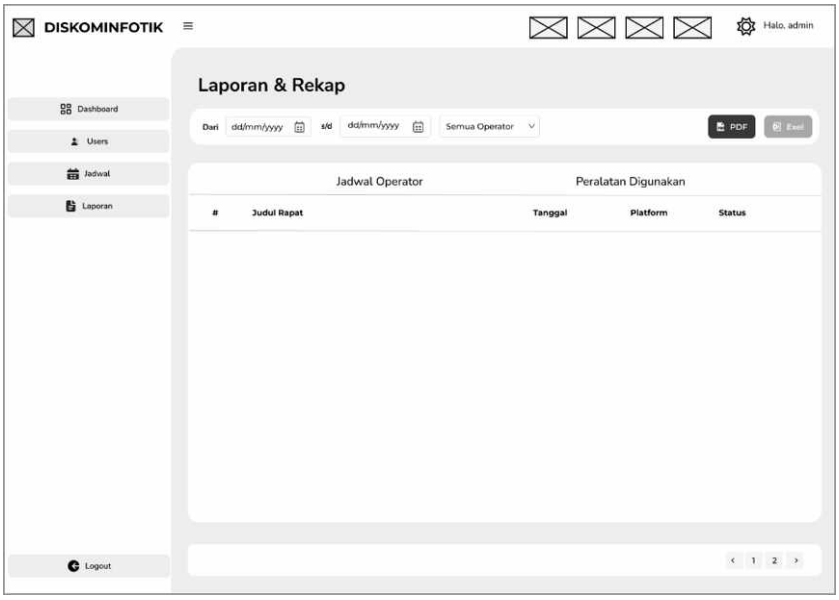


Gambar 4.23 Tampilan Halaman Tambah Jadwal

Deskripsi
Ini merupakan rancangan tampilan tambah jadwal yang digunakan admin untuk menambahkan jadwal kegiatan rapat dengan mengisi informasi kegiatan, waktu pelaksanaan, platform rapat, keterangan, dan operator yang bertugas

Tabel 4.43 Tampilan Halaman Laporan

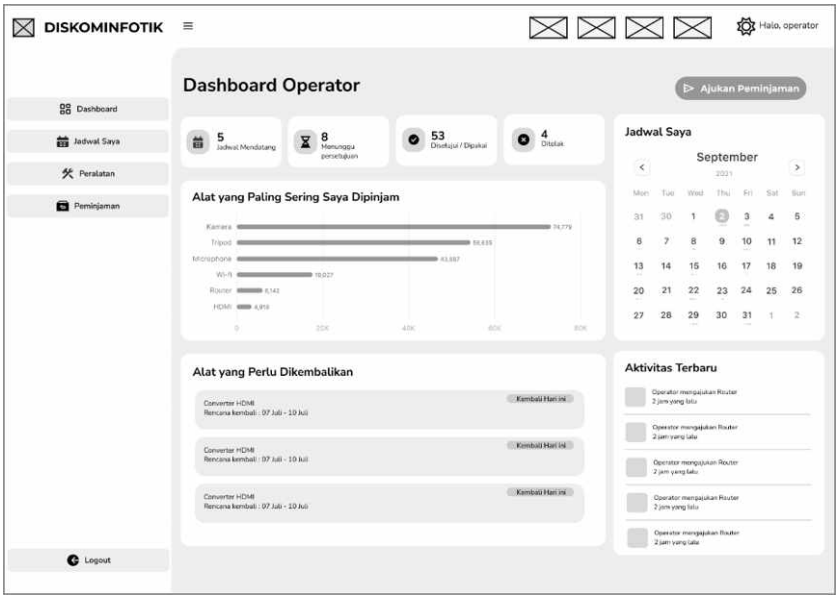
Halaman Laporan



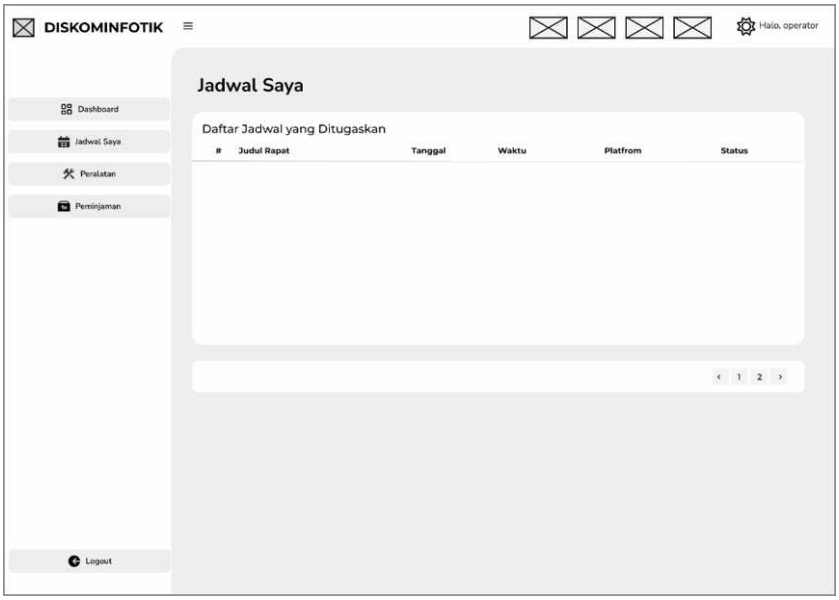
Gambar 4.24 Tampilan Halaman Laporan

Deskripsi
Ini merupakan rancangan tampilan laporan penjadwalan yang digunakan untuk menampilkan laporan kegiatan rapat berdasarkan rentang tanggal dan operator, serta melakukan export laporan ke dalam format PDF dan Excel

Tabel 4.44 Tampilan Halaman Dashboard Operator

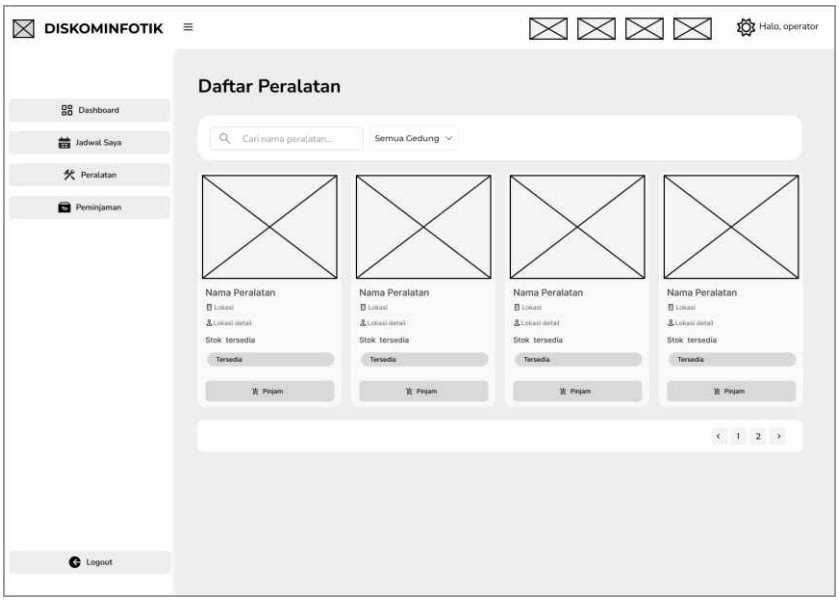
Halaman Dashboard Operator
 *Gambar 4.25 Tampilan Halaman Dashboard Operator*
Deskripsi
Ini merupakan rancangan tampilan dashboard operator yang digunakan untuk menampilkan jumlah jadwal mendatang, jumlah pengajuan peminjaman berdasarkan status (menunggu, disetujui, ditolak), daftar peminjaman yang perlu dikembalikan, peralatan yang paling sering dipinjam, aktivitas terbaru, serta kalender jadwal bulanan

Tabel 4.45 Tampilan Halaman Jadwal Operator

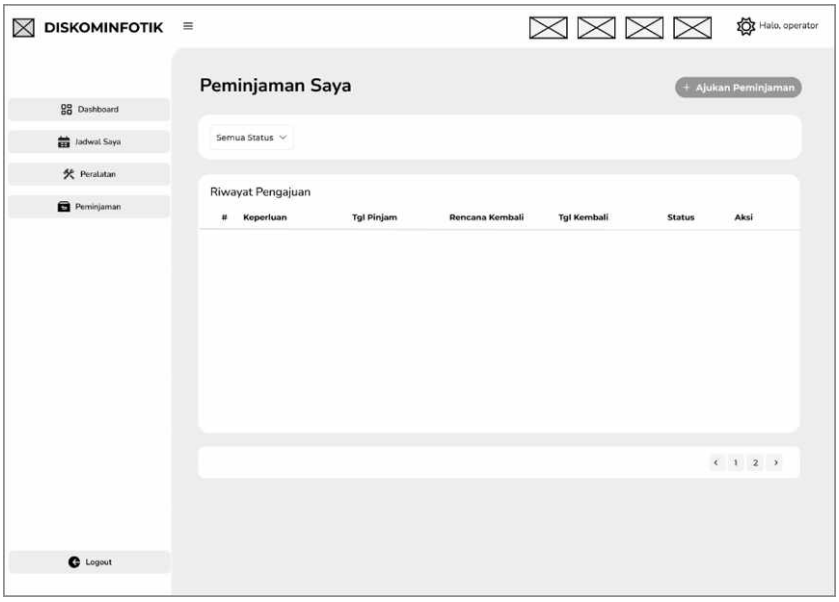
Halaman Jadwal Operator
 *Gambar 4.26 Tampilan Halaman Jadwal Operator*

Deskripsi
Ini merupakan rancangan tampilan jadwal operator yang digunakan untuk menampilkan daftar jadwal kegiatan yang sedang berlangsung dan riwayat jadwal rapat yang telah dilaksanakan oleh operator

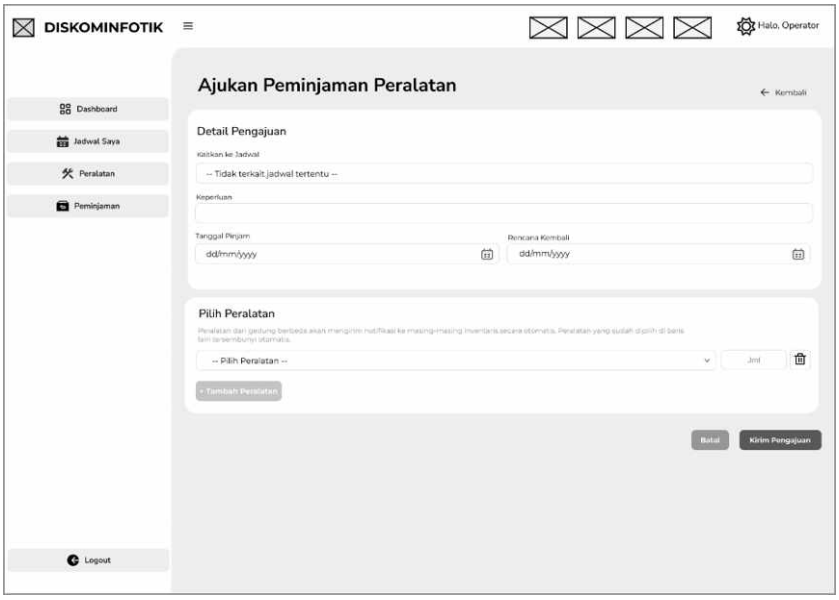
Tabel 4.46 Tampilan Halaman Peralatan

Halaman Peralatan
 *Gambar 4.27 Tampilan Halaman Peralatan*
Deskripsi
Ini merupakan rancangan tampilan peralatan yang digunakan operator untuk melihat daftar peralatan yang tersedia serta mengajukan peminjaman peralatan sesuai kebutuhan kegiatan rapat

Tabel 4.47 Tampilan Halaman Peminjaman

Halaman Peminjaman
 *Gambar 4.28 Tampilan Halaman Peminjaman*
Deskripsi
Ini merupakan rancangan tampilan riwayat peminjaman peralatan yang digunakan operator untuk melihat daftar pengajuan peminjaman yang telah diajukan beserta status persetujuannya, serta mengubah atau membatalkan pengajuan yang masih berstatus menunggu

Tabel 4.48 Tampilan Halaman Ajukan Peminjaman

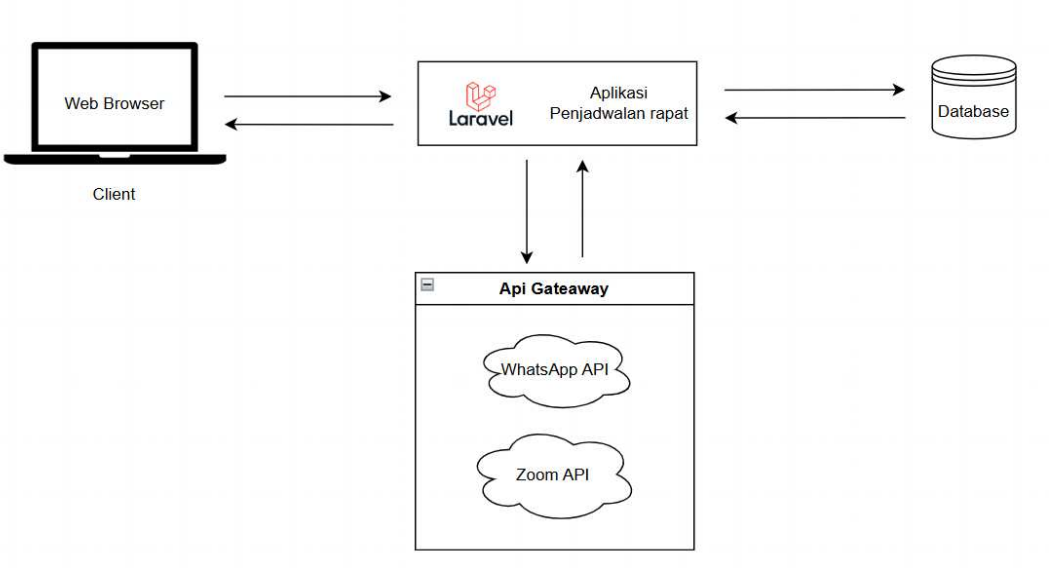
Halaman Ajukan Peminjaman
 *Gambar 4.29 Tampilan Halaman Ajukan Peminjaman Peralatan*
Deskripsi
Ini merupakan rancangan tampilan pengajuan peminjaman peralatan yang digunakan operator untuk mengajukan peminjaman dengan memilih peralatan dan jumlah yang dibutuhkan, mengaitkan pengajuan ke jadwal rapat (opsional), serta mengisi keperluan dan tanggal peminjaman

4.7 Arsitektur Perangkat Lunak

Arsitektur perangkat lunak merupakan struktur dasar yang menggambarkan susunan komponen utama pada suatu sistem beserta hubungan dan mekanisme komunikasi antarkomponen dalam menjalankan fungsi aplikasi secara keseluruhan. Arsitektur ini berfungsi sebagai acuan teknis yang menjelaskan proses pengolahan data, interaksi antar lapisan aplikasi, serta keterhubungan sistem dengan layanan eksternal yang digunakan.

Arsitektur perangkat lunak pada aplikasi ini menggambarkan alur kerja sistem yang diakses oleh pengguna melalui web browser. Permintaan (*request*) yang dikirimkan pengguna diteruskan melalui protokol HTTP menuju aplikasi yang dibangun menggunakan framework Laravel. Selanjutnya, *request* diproses secara bertahap melalui beberapa lapisan, yaitu Routes yang berfungsi memetakan URL ke proses yang sesuai, Middleware yang melakukan validasi hak akses pengguna berdasarkan role dan menu yang diizinkan, Controller yang menangani *request* serta menghasilkan *response*, dan *Business Logic Layer (Service)* yang menjalankan logika bisnis aplikasi, seperti validasi bentrok jadwal, pengelolaan stok peminjaman, serta integrasi dengan layanan pihak ketiga.

Data yang dibutuhkan sistem dikelola melalui Model yang berinteraksi dengan basis data MySQL menggunakan Eloquent ORM. Selain itu, aplikasi terintegrasi dengan layanan eksternal, yaitu Fonnte API untuk mengirimkan notifikasi WhatsApp secara otomatis, meliputi pemberitahuan jadwal baru, perubahan jadwal, dan pengajuan peminjaman. Aplikasi juga memanfaatkan Zoom API untuk membuat, memperbarui, dan menghapus meeting secara otomatis pada penjadwalan rapat dengan *platform Online* maupun *Hybrid*. Seluruh komponen tersebut saling berinteraksi untuk mendukung proses penjadwalan rapat dan peminjaman peralatan secara terstruktur dengan pemisahan tanggung jawab yang jelas pada setiap lapisan arsitektur.



Gambar 4.30 Arsitektur Perangkat Lunak

BAB 5
IMPLEMENTASI DAN PENGUJIAN

5.1 Kontruksi Perangkat Lunak

Tahap konstruksi adalah tahap pengkodean atau tahap implementasi model dari rancangan analisis dan perancangan dari tahap sebelumnya. Berikut ini merupakan spesifikasi perangkat keras dan perangkat lunak yang digunakan dalam proses pembangunan aplikasi.

5.1.1 Spesifikasi Perangkat Keras

Spesifikasi minimal perangkat keras yang digunakan untuk membangun dan menjalankan aplikasi yang dirancang pada bab sebelumnya dapat dilihat pada Tabel 5.1

Tabel 5.1 Spesifikasi Perangkat Keras

No.	Nama Perangkat	Spesifikasi
1.	*Processor*	Dual Core 2.0 GHz
2.	Memori (RAM)	4 GB direkomendasiikan 8 GB
3.	Ruang Penyimpanan	50 GB
4.	Sistem Operasi	Windows 10/11 (64-bit)

5.1.2 Spesifikasi Perangkat Lunak dan kakas

Spesifikasi perangkat lunak yang digunakan untuk membangun aplikasi yang telah dirancang pada bab sebelumnya dapat dilihat pada Tabel 5.2

Tabel 5.2 Spesifikasi Perangkat Lunak

No	Nama Perangkat Lunak	Sumber	Deskripsi
1.	Draw.io	http://app.diagrams.net	Alat bantu untuk menggambar diagram-diagram pada tahap analisis dan perancangan sistem.
2.	Figma	http://figma.com	Digunakan untuk merancang tampilan antarmuka (UI) sebelum diimplementasikan ke dalam kode.
3.	GitHub	http://github.com	Layanan hosting repositori berbasis git yang dipakai untuk menyimpan riwayat perubahan kode selama pengembangan.
4.	Visual Studio Code	http://code.visualstudio.com	Aplikasi editor kode yang dipakai penulis untuk mengembangkan seluruh berkas program.
5.	Laragon	http://laragon.org	Menyediakan lingkungan server lokal (local development environment) yang dipakai selama proses pembangunan aplikasi.
6.	MySQL	http://mysql.com	Sebagai database management system (DBMS) untuk penyimpanan data aplikasi.
7.	Google Chrome	http://google.com/chrome	Web browser yang digunakan untuk menjalankan dan menguji aplikasi selama pengembangan.

5.1.3 Pustaka dan Bahasa yang digunakan

Pustaka atau *package* merupakan sekumpulan kode program yang berisi fungsi, *library*, atau kelas yang dapat digunakan kembali untuk mendukung dan mempermudah proses pembangunan aplikasi.

Tabel 5.3 Pustaka dan Bahasa yang digunakan

No	Nama Perangkat	Deskripsi
1.	PHP	Bahasa pemrograman utama untuk backend, digunakan bersama framework Laravel.
2.	Laravel	Framework PHP untuk pengembangan aplikasi web, menyediakan struktur MVC dan berbagai fitur bawaan.
3.	JavaScript	Bahasa pemrograman untuk interaktivitas sisi frontend, ditulis native tanpa framework tambahan.
4.	Tailwind CSS	Framework CSS utility-first untuk styling responsif, dikompilasi melalui Vite.
5.	Composer	Dependency manager untuk PHP, mengelola paket dan autoloading.
6.	Git	Version control system yang berfungsi mencatat dan melacak perubahan kode program.
7.	Laravel Excel (Maatwebsite)	Package untuk menghasilkan laporan dalam format Excel dengan styling.
8.	jsPDF & jsPDF-AutoTable	Library JavaScript sisi client untuk menghasilkan dan mengunduh laporan dalam format PDF asli.
9.	Guzzle HTTP (Laravel HTTP Client)	Digunakan untuk melakukan request ke REST API pihak ketiga, seperti WhatsApp (Fonnte) dan Zoom.
10.	No CAPTCHA (Google reCAPTCHA)	Package untuk proteksi form login dari bot/spam menggunakan reCAPTCHA.

5.2 Pengkodean

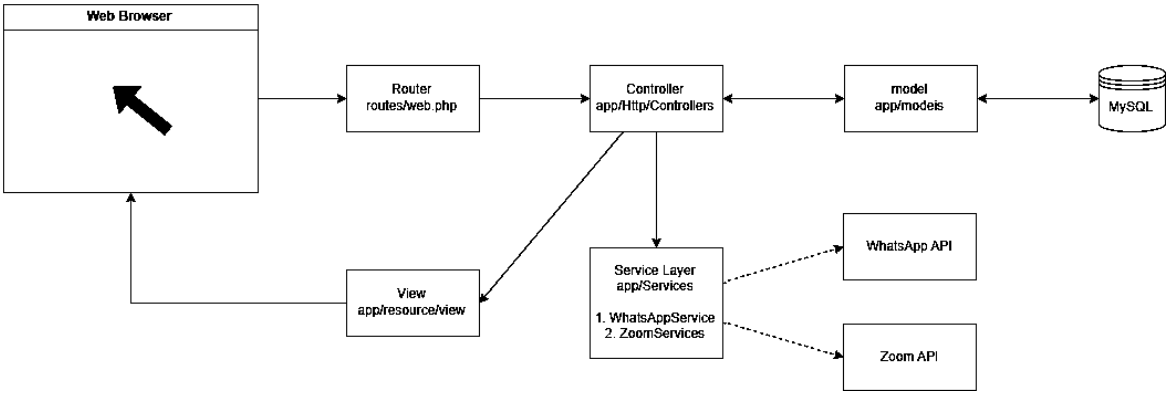
Pengkodean merupakan tahap implementasi dari hasil perancangan sistem yang telah dilakukan sebelumnya ke dalam bentuk kode program sehingga dapat dijalankan sebagai sebuah aplikasi.

5.2.1 Tahap Pengkodean Aplikasi

Tahap Pengkodean dilakukan implementasi aplikasi berbasis web menggunakan bahasa pemrograman PHP dengan *framework* Laravel. Sistem juga menggunakan MySQL sebagai *Database Management System* untuk mengelola dan menyimpan data aplikasi.

5.2.1.1 Arsitektur Pengkodean Aplikasi

Arsitektur aplikasi digunakan sebagai struktur yang mengatur hubungan dan interaksi antar komponen dalam aplikasi sehingga setiap komponen dapat menjalankan fungsinya sesuai dengan kebutuhan sistem. Arsitektur aplikasi yang digunakan pada pembangunan sistem ini dapat dilihat pada gambar berikut.

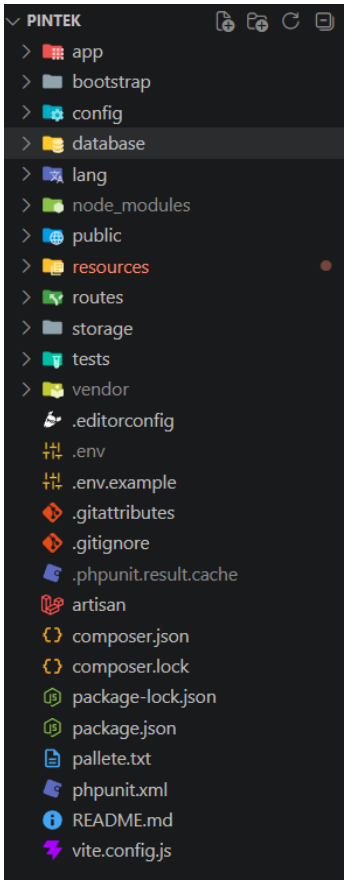


Gambar 5.1 Arsitektur Pengkodean

5.2.1.2 Struktur Folder Website

Struktur direktori dalam Laravel mengikuti pola standar yang bertujuan untuk menjaga keteraturan dan kemudahan dalam pemeliharaan kode. Tabel di bawah ini menunjukkan struktur folder utama proyek aplikasi Pintek.

Tabel 5.4 Struktur Folder

Struktur Folder
 *Gambar 5.2 Struktur Folder*
Deskripsi
Gambar di atas merupakan struktur folder utama yang berada di aplikasi penjadwalan rapat

Tabel dibawah ini menunjukkan deskripsi penjelasan struktur folder utama proyek aplikasi penjadwalan rapat

Tabel 5.5 Penjelasan Struktur Umum Proyek

No	Nama Folder	Deskripsi
1.	app	Berisi komponen inti aplikasi (kode utama)
2.	app/Exports	Berisi kelas untuk ekspor data ke Excel
3.	app/Helpers	Berisi fungsi bantu custom, seperti pembuat ID unik
4.	app/Http	Mengelola request HTTP masuk dan response
5.	app/Http/Controllers	Mendefinisikan logika aplikasi dan menangani request serta menghasilkan response
6.	app/Http/Middleware	Menyaring request sebelum diproses, mis. pembatas akses berdasarkan role
7.	app/Models	Merepresentasikan model data atau objek yang berhubungan dengan basis data
8.	app/Providers	Mendefinisikan penyedia layanan yang digunakan aplikasi
9.	app/Services	Berisi logika bisnis utama yang dipisah dari Controller
10.	app/View	Berisi kelas komponen tampilan custom
11.	bootstrap	Merupakan direktori bawaan Laravel yang berisi proses inisialisasi aplikasi
12.	config	Berisi berkas-berkas konfigurasi aplikasi (database, autentikasi, cache, mail, dsb)
13.	database	Berisi folder factories (cetakan data dummy), migrations (struktur tabel), dan seeders (isi awal tabel)
14.	lang	Berisi berkas terjemahan, khususnya pesan validasi dan autentikasi berbahasa Indonesia
15.	node_modules	Untuk menyimpan modul dependensi JavaScript dan Node.js
16.	public	Sebagai penyedia berkas dan sumber daya yang dapat diakses publik
17.	resources	Menyimpan berkas sumber tampilan, yaitu views (template Blade) dan css (sumber Tailwind)
18.	routes	Berisi file endpoint dan routing
19.	storage	Menyimpan berbagai jenis data berkas atau media yang tidak disimpan di bawah direktori "public"
20.	tests	Tempat menulis, menjalankan, dan mengelola berbagai jenis pengujian untuk aplikasi
21.	vendor	Tempat semua dependensi pihak ketiga disimpan

5.2.1.3 Struktur Aplikasi

Aplikasi yang dibangun menerapkan pola arsitektur *Model View Controller* (MVC), yaitu pola yang memisahkan pengelolaan data (*Model*), tampilan antarmuka yang disajikan kepada pengguna (*View*), serta logika penghubung antara keduanya (*Controller*). Penerapan arsitektur MVC bertujuan untuk memisahkan tanggung jawab setiap komponen sehingga proses pengembangan dan pemeliharaan aplikasi dapat dilakukan dengan lebih mudah. Struktur aplikasi yang dibangun dijelaskan pada bagian berikut.

5.2.1.4 Model

Model berperan mengatur, menyiapkan, memanipulasi, serta mengorganisasikan data pada database sesuai dengan instruksi yang diberikan oleh Controller.

Tabel 5.6 Model

No	Nama Model	Deskripsi
1.	User.php	Kelas ini digunakan untuk berinteraksi dengan tabel users pada basis data
2.	Penjadwalan.php	Kelas ini digunakan untuk berinteraksi dengan tabel penjadwalan pada basis data
3.	Peralatan.php	Kelas ini digunakan untuk berinteraksi dengan tabel peralatan pada basis data
4.	Peminjaman.php	Kelas ini digunakan untuk berinteraksi dengan tabel peminjaman pada basis data
5.	PeminjamanItem.php	Kelas ini digunakan untuk berinteraksi dengan tabel peminjaman_item pada basis data

5.2.1.5 Controller

Controller bertugas menerima permintaan dari pengguna, melakukan validasi data masukan, menjalankan logika aplikasi, dan berinteraksi dengan Model untuk mengelola data pada basis data. Hasil pengolahan tersebut kemudian diteruskan ke View agar dapat ditampilkan kepada pengguna. Controller yang digunakan pada aplikasi penjadwalan rapat ditunjukkan pada Tabel 5.5.

Tabel 5.7 Controller

No	Nama Controller	Deskripsi
1	AdminController.php	Kelas ini digunakan untuk menampilkan halaman dashboard admin, mengelola laporan, dan menyediakan data yang dibutuhkan pada halaman dashboard beserta laporan
2	UserController.php	Kelas ini digunakan untuk mengelola data pengguna, meliputi penambahan, perubahan, pencarian, dan pengaktifan maupun penonaktifan pengguna
3	PenjadwalanController.php	Kelas ini digunakan untuk mengelola data penjadwalan rapat, meliputi penambahan, perubahan, pembatalan, serta menampilkan detail jadwal rapat
4	OperatorController.php	Kelas ini digunakan untuk menampilkan halaman dashboard operator beserta ringkasan aktivitas jadwal dan peminjaman peralatan miliknya
5	PeminjamanController.php	Kelas ini digunakan untuk mengelola proses pengajuan peminjaman peralatan, perubahan pengajuan, pembatalan pengajuan, beserta riwayat peminjaman peralatan
6	PeralatanController.php	Kelas ini menyediakan data dari model Peralatan untuk ditampilkan pada halaman peralatan

5.2.1.6 View

View menjadi lapisan yang menampilkan antarmuka aplikasi kepada pengguna sekaligus menerima data yang dikirimkan oleh *Controller*. Melalui View ini, pengguna dapat berinteraksi dengan aplikasi, misalnya mengisi formulir atau menekan tombol, yang selanjutnya data tersebut diteruskan kembali kepada *Controller* untuk diproses lebih lanjut.

Interaksi pengguna dengan aplikasi menjadi lebih mudah berkat keberadaan *View*, sementara *Controller* di sisi lain menangani logika bisnis dan mengambil data dari Model untuk kemudian disajikan. Pembagian peran semacam ini memisahkan tampilan dari logika bisnis, sehingga aplikasi dapat dikembangkan secara lebih tertata dan fleksibel terhadap perubahan kebutuhan di kemudian hari.

Tabel 5.8 View

No	Nama View	Folder	Deskripsi
1.	login.blade.php	auth	Kelas ini digunakan untuk menampilkan halaman login.
2.	admin.blade.php	dashboard/beranda	Kelas ini digunakan untuk menampilkan halaman dashboard Admin.
3.	index.blade.php	dashboard/jadwal	Kelas ini digunakan untuk menampilkan halaman daftar jadwal rapat.
4.	create.blade.php	dashboard/jadwal	Kelas ini digunakan untuk menampilkan halaman tambah jadwal rapat.
5.	edit.blade.php	dashboard/jadwal	Kelas ini digunakan untuk menampilkan halaman ubah jadwal rapat.
6.	show.blade.php	dashboard/jadwal	Kelas ini digunakan untuk menampilkan halaman detail jadwal rapat.
7.	admin-index.blade.php	dashboard/laporan	Kelas ini digunakan untuk menampilkan halaman laporan.
8.	print_jadwal.blade.php	dashboard/laporan	Kelas ini digunakan untuk menampilkan halaman cetak laporan jadwal rapat dalam format PDF.
9.	print_peralatan.blade.php	dashboard/laporan	Kelas ini digunakan untuk menampilkan halaman cetak laporan peralatan dalam format PDF.
10.	index.blade.php	dashboard/users	Kelas ini digunakan untuk menampilkan halaman daftar pengguna.
11.	create.blade.php	dashboard/users	Kelas ini digunakan untuk menampilkan halaman tambah pengguna.
12.	edit.blade.php	dashboard/users	Kelas ini digunakan untuk menampilkan halaman ubah pengguna.
13.	operator.blade.php	dashboard/beranda	Kelas ini digunakan untuk menampilkan halaman dashboard Operator.
14.	index.blade.php	dashboard/peralatan	Kelas ini digunakan untuk menampilkan halaman daftar peralatan.
15.	operator-index.blade.php	dashboard/peminjaman	Kelas ini digunakan untuk menampilkan halaman daftar pengajuan peminjaman peralatan.
16.	create.blade.php	dashboard/peminjaman	Kelas ini digunakan untuk menampilkan halaman pengajuan peminjaman peralatan.
17.	edit.blade.php	dashboard/peminjaman	Kelas ini digunakan untuk menampilkan halaman ubah pengajuan peminjaman peralatan.

5.3 Implementasi Kode

Tahap ini menyajikan hasil implementasi kode program yang dibangun berdasarkan hasil analisis dan perancangan pada bab sebelumnya, mencakup penerapan struktur Model, Controller, dan View sesuai dengan arsitektur aplikasi yang telah dirancang. Implementasi ini merupakan wujud nyata dari rancangan kelas dan basis data ke dalam baris kode program yang dapat dijalankan.

1. Model

Tabel 5.9 Kode Model Penjadwalan.php

Model Penjadwalan.php

app > Models > Penjadwalan.php > { } App\Models > Penjadwalan

```php
<?php
namespace App\Models;
use Carbon\Carbon;
use Illuminate\Database\Eloquent\Model;

60 references | 0 implementations
class Penjadwalan extends Model
{
    0 references
    protected $table        = 'penjadwalan';
    0 references
    protected $primaryKey   = 'id_penjadwalan';
    0 references
    public    $incrementing = false;
    0 references
    protected $keyType      = 'string';
    0 references
    protected $fillable = [
    'id_penjadwalan',
    'judul_kegiatan',
    'tanggal',
    'waktu_mulai',
    'waktu_selesai',
    'platform',
    'keterangan',
    'lokasi_fisik',
    'status',
    'alasan_batal',
    'zoom_meeting_id',
    'zoom_password',
    'zoom_account',
    'link_otomatis',
];

5 references | 0 overrides
public function operators()
{
    return $this->belongsToMany(User::class, 'jadwal_operator', 'id_penjadwalan', 'id_user');
}
0 references
protected $casts = [
    'tanggal'       => 'date:Y-m-d',
    'link_otomatis' => 'boolean',
];

1 reference | 0 overrides
public function peminjaman()
{
    return $this->hasMany(Peminjaman::class, 'id_penjadwalan', 'id_penjadwalan');
}

2 references | 0 overrides
public function peralatanSudahDiajukan(?string $kecualiIdPeminjaman = null): array
{
    return $this->peminjaman()
        ->whereIn('status', ['diajukan', 'disetujui', 'dikembalikan'])
        ->when($kecualiIdPeminjaman, fn ($q, $id) => $q->where('id_peminjaman', '!=', $id))
        ->with('items.peralatan')
        ->get()
        ->flatMap(fn ($p) => $p->items)
        ->filter(fn ($item) => $item->peralatan)
        ->groupBy('peralatan.nama_peralatan')
        ->map(fn ($items) => $items->sum('jumlah'))
        ->all();
}
```

```php
    2 references | 0 overrides
    public function peralatanReferensi()
    {
        return $this->belongsToMany(Peralatan::class, 'jadwal_peralatan', 'id_penjadwalan', 'id_peralatan')
            ->withPivot('jumlah');
    }

    1 reference | 0 overrides
    public function isDibatalkan(): bool
    {
        return $this->status === 'dibatalkan' || !empty($this->alasan_batal);
    }

    0 references | 0 overrides
    public function getStartDateTimeAttribute(): ?Carbon
    {
        if (!$this->tanggal || !$this->waktu_mulai) return null;

        return Carbon::parse(
            $this->tanggal->format('Y-m-d') . ' ' . $this->waktu_mulai,
            'Asia/Jakarta'
        );
    }

    0 references | 0 overrides
    public function getEndDateTimeAttribute(): ?Carbon
    {
        if (!$this->tanggal || !$this->waktu_selesai) return null;

        return Carbon::parse(
            $this->tanggal->format('Y-m-d') . ' ' . $this->waktu_selesai,
            'Asia/Jakarta'
        );
    }

    0 references | 0 overrides
    public function scopeBentrok($query, string $tanggal, string $mulai, string $selesai, ?string $excludeId = null)
    {
        return $query
            ->whereDate('tanggal', $tanggal)
            ->when($excludeId, fn($q) => $q->where('id_penjadwalan', '!=', $excludeId))
            ->where(function ($q) use ($mulai, $selesai) {
                $q->whereBetween('waktu_mulai',    [$mulai, $selesai])
                  ->orWhereBetween('waktu_selesai', [$mulai, $selesai])
                  ->orWhere(function ($qq) use ($mulai, $selesai) {
                      $qq->where('waktu_mulai',   '<=', $mulai)
                         ->where('waktu_selesai', '>=', $selesai);
                  });
            });
    }
}
```

Gambar 5.3 Model Penjadwalan

Tabel 5.10 Kode Model Peminjaman.php

Model Peminjaman.php

app > Models > Peminjaman.php > { } App\Models > Peminjaman

```php
<?php
namespace App\Models;
use Illuminate\Database\Eloquent\Model;

61 references | 0 implementations
class Peminjaman extends Model
{
    0 references
    protected $table      = 'peminjaman';
    0 references
    protected $primaryKey = 'id_peminjaman';
    0 references
    public    $incrementing = false;
    0 references
    protected $keyType      = 'string';
    0 references
    protected $fillable = [
        'id_peminjaman',
        'id_user',
        'id_penjadwalan',
        'tanggal_pinjam',
        'tanggal_kembali_rencana',
        'tanggal_kembali_aktual',
        'keperluan',
        'status',
        'catatan_inventaris',
        'alasan_batal',
    ];
    0 references
    protected $casts = [
        'tanggal_pinjam'          => 'date',
        'tanggal_kembali_rencana' => 'date',
        'tanggal_kembali_aktual'  => 'date',
    ];

    0 references | 0 overrides
    public function user()
    {
        return $this->belongsTo(User::class, 'id_user', 'id_user');
    }

    0 references | 0 overrides
    public function penjadwalan()
    {
        return $this->belongsTo(Penjadwalan::class, 'id_penjadwalan', 'id_penjadwalan');
    }

    11 references | 0 overrides
    public function items()
    {
        // chaperone() menautkan balik relasi peminjaman pada tiap item yang di-load
        // lewat induknya - dipakai PeminjamanItem::getBadgeAttribute() untuk membaca
        // status induk tanpa query tambahan (N+1).
        return $this->hasMany(PeminjamanItem::class, 'id_peminjaman', 'id_peminjaman')->chaperone();
    }

    9 references | 0 overrides
    public function isMenunggu(): bool   { return $this->status === 'diajukan'; }
    1 reference | 0 overrides
    public function isDisetujui(): bool  { return $this->status === 'disetujui'; }
    1 reference | 0 overrides
    public function isDibatalkan(): bool { return $this->status === 'dibatalkan'; }

    0 references | 0 overrides
    public function getBadgeAttribute(): array
    {
        return match ($this->status) {
            'diajukan','menunggu'     => ['class' => 'badge-warning',  'label' => 'Menunggu'],
            'disetujui'               => ['class' => 'badge-active',   'label' => 'Disetujui'],
            'ditolak'                 => ['class' => 'badge-danger',   'label' => 'Ditolak'],
            'dikembalikan'            => ['class' => 'badge-info',     'label' => 'Dikembalikan'],
            'dibatalkan'              => ['class' => 'badge-danger',   'label' => 'Dibatalkan'],
            default                   => ['class' => '',               'label' => $this->status],
        };
    }

}
```

Gambar 5.4 Model Peminjaman

2. Controller

Tabel 5.11 Kode PenjadwalanController.php

PenjadwalanController.php

app > Http > Controllers > PenjadwalanController.php > { } App\Http\Controllers > PenjadwalanController > batalkan()

```php
<?php
namespace App\Http\Controllers;
use App\Models\Penjadwalan;
use App\Models\Peralatan;
use App\Models\User;
use App\Services\PenjadwalanService;
use Illuminate\Http\Request;
use Illuminate\Validation\ValidationException;
use Carbon\Carbon;

10 references | 0 implementations
class PenjadwalanController extends Controller
{
    0 references | 0 overrides
    public function __construct(private PenjadwalanService $service) {}

    1 reference | 0 overrides
    public function index(Request $request)
    {
        // Operator cuma lihat jadwal yang ditugaskan ke dirinya; role lain lihat semua.
        $jadwal = Penjadwalan::with(['operators', 'peralatanReferensi'])
            ->when(auth()->user()->role === 'operator', fn($q) =>
                $q->whereHas('operators', fn($qq) => $qq->where('users.id_user', auth()->user()->id_user))
            )
            // where() dibungkus closure supaya orWhere tidak bocor ke scope operator di atas.
            ->when($request->search, fn($q, $s) =>
                $q->where(fn($qq) => $qq->where('judul_kegiatan', 'like', "%{$s}%")
                                        ->orWhere('platform', 'like', "%{$s}%"))
            )
            ->when($request->platform, fn($q, $p) =>
                $q->where('platform', 'like', "%{$p}%")
            )
            // Status Aktif/Selesai diturunkan dari tanggal+waktu, bukan kolom status literal.
            ->when($request->status, fn($q, $s) => match ($s) {
                'aktif'      => $q->where('status', '!=', 'dibatalkan')->whereRaw('TIMESTAMP(tanggal, waktu_selesai) >= NOW()'),
                'selesai'    => $q->where('status', '!=', 'dibatalkan')->whereRaw('TIMESTAMP(tanggal, waktu_selesai) < NOW()'),
                'dibatalkan' => $q->where('status', 'dibatalkan'),
                default      => $q,
            })
            // Aktif tampil dulu (tanggal terdekat); Selesai+Dibatalkan digabung satu riwayat di bawah (paling baru dulu).
            ->orderByRaw("(status = 'dibatalkan' OR TIMESTAMP(tanggal, waktu_selesai) < NOW()) ASC")
            ->orderByRaw("CASE WHEN status = 'dibatalkan' OR TIMESTAMP(tanggal, waktu_selesai) < NOW()
                          THEN -DATEDIFF(tanggal, CURDATE()) ELSE DATEDIFF(tanggal, CURDATE()) END ASC")
            ->orderBy('waktu_mulai')
            ->paginate(10)
            ->withQueryString();
        $bisaTambah = auth()->user()->punyaAkses('jadwal', 'tambah');
        $bisaUbah   = auth()->user()->punyaAkses('jadwal', 'ubah');
        return view('dashboard.jadwal.index', compact('jadwal', 'bisaTambah', 'bisaUbah'));
    }
```

Gambar 5.5 Controller Penjadwalan

Tabel 5.12 Kode PenjadwalanController.php

PeminjamanController.php

app > Http > Controllers > PeminjamanController.php > { } App\Http\Controllers > PeminjamanController > jadwalAktifOperator()

```php
<?php
namespace App\Http\Controllers;
use App\Models\Peminjaman;
use App\Models\PeminjamanItem;
use App\Models\Penjadwalan;
use App\Models\Peralatan;
use App\Services\PeminjamanService;
use Illuminate\Http\Request;
use Illuminate\Validation\Rule;

21 references | 0 implementations
class PeminjamanController extends Controller
{
    0 references | 0 overrides
    public function __construct(private PeminjamanService $service) {}
    2 references | 0 overrides
    public function operatorIndex(Request $request)
    {
        $peminjaman = Peminjaman::with(['items.peralatan', 'penjadwalan'])
            ->where('id_user', auth()->user()->id_user)
            // Search mencakup keperluan DAN nama alat di dalam pengajuan - dibungkus
            // where() supaya orWhereHas tidak bocor keluar dari filter kepemilikan.
            ->when($request->search, fn($q, $s) =>
                $q->where(fn($qq) => $qq->where('keperluan', 'like', "%{$s}%")
                    ->orWhereHas('items.peralatan', fn($qa) => $qa->where('nama_peralatan', 'like', "%{$s}%")))
            )
            ->when($request->status, fn($q, $v) => $q->where('status', $v))
            ->when($request->start, fn($q, $v) => $q->whereDate('tanggal_pinjam', '>=', $v))
            ->when($request->end,   fn($q, $v) => $q->whereDate('tanggal_pinjam', '<=', $v))
            ->orderByDesc('created_at')
            ->paginate(10)
            ->withQueryString();
        return view('dashboard.peminjaman.operator-index', compact('peminjaman'));
    }
    2 references | 0 overrides
    public function operatorCreate(Request $request)
    {
        abort_if(!auth()->user()->punyaAkses('peminjaman', 'tambah'), 403, 'Anda tidak memiliki akses untuk mengajukan peminjaman.');
        $peralatan = Peralatan::whereRaw('(stok - COALESCE(rusak,0)) > 0')
            ->orderBy('gedung')
            ->orderBy('nama_peralatan')
            ->get()
            ->groupBy('gedung');

        $selectedPeralatanId = $request->query('id_peralatan');
        $jadwalAktif         = $this->jadwalAktifOperator();
        return view('dashboard.peminjaman.create', compact('peralatan', 'selectedPeralatanId', 'jadwalAktif'));
    }
    2 references | 0 overrides
    public function operatorStore(Request $request)
    {
        abort_if(!auth()->user()->punyaAkses('peminjaman', 'tambah'), 403, 'Anda tidak memiliki akses untuk mengajukan peminjaman.');
        $idJadwalAktif = $this->jadwalAktifOperator()->pluck('id_penjadwalan');
        $request->validate($this->aturanValidasiPengajuan($idJadwalAktif), $this->pesanValidasiPengajuan());

        try {
            $this->service->ajukan(
                header: [
                    'id_user'                => auth()->user()->id_user,
                    'id_penjadwalan'         => $request->id_penjadwalan ?: null,
                    'tanggal_pinjam'         => $request->tanggal_pinjam,
                    'tanggal_kembali_rencana' => $request->tanggal_kembali_rencana,
                    'keperluan'              => $request->keperluan,
                    'status'                 => 'diajukan',
                ],
                peralatanIds: $request->peralatan_ids,
                jumlahArr:    $request->peralatan_jumlah,
            );
```

Gambar 5.6 Controller Peminjaman

3. Views

Tabel 5.13 Kode index.blade.php halaman jadwal

Index.blade.php

resources > views > dashboard > jadwal > index.blade.php

```
1  @extends('layouts.app')
2  @section('title', $bisaUbah ? 'Data Jadwal' : 'Jadwal Saya')
3  @section('sidebar-menu') <x-sidebar /> @endsection
4
5  @section('content')
6      @php $roleAktif = auth()->user()->role; @endphp
7      <main class="w-full pt-9 px-6 pb-9 font-sans max-h-[calc(100vh-56px)] overflow-y-auto overflow-x-hidden">
8          <div class="flex items-center justify-between gap-4 flex-wrap mb-5">
9              <div>
10                 <h1 class="text-4xl font-semibold mb-2.5 text-text dark:text-text-dark">
11                     {{ $bisaUbah ? 'Data Jadwal Rapat' : 'Jadwal Saya' }}</h1>
12             </div>
13             @if($bisaTambah)
14                 <a href="{{ route($roleAktif . '.jadwal.create') }}"
15                     class="h-9 px-4 rounded-full bg-primary text-surface dark:text-surface-dark flex justify-center items-center gap-2.5 font-medium">
16                     <i class="bx bx-plus"></i><span class="text">Tambah Jadwal</span>
17                 </a>
18             @endif
19         </div>
20
21         {{-- Toolbar filter tampil untuk semua role - operator (baca saja) juga bisa
22              search/filter jadwal miliknya, query-nya ditangani PenjadwalanController::index(). --}}
23         <div
24             class="bg-surface dark:bg-surface-dark rounded-[10px] py-3.5 px-4 mb-4 flex items-center gap-2.5 flex-wrap shadow-[0_2px_8px_rgba(0,0,0,0.04)]
               ">
25             <form method="GET" action="{{ route($roleAktif . '.jadwal.index') }}" class="contents">
26                 <div class="relative flex-1 min-w-[180px] max-w-[300px]">
27                     <i
28                         class="bx bx-search absolute left-2.5 top-1/2 -translate-y-1/2 text-text-muted text-base pointer-events-none"></i>
29                     <input type="text" name="search" data-live-search="#hasil-jadwal" autocomplete="off"
30                         placeholder="Cari judul, platform..." value="{{ request('search') }}"
31                         class="w-full h-9 pl-[34px] pr-3 border border-page-bg dark:border-page-bg-dark rounded-lg bg-page-bg dark:bg-page-bg-dark
                           text-text dark:text-text-dark text-[13px] font-sans transition-colors duration-200 focus:border-primary focus:outline-none
                           focus:bg-surface dark:focus:bg-surface-dark">
32                 </div>
33                 <select name="platform" onchange="this.form.submit()"
34                     class="h-9 px-2.5 border border-page-bg dark:border-page-bg-dark rounded-lg bg-page-bg dark:bg-page-bg-dark text-text
                       dark:text-text-dark text-[13px] font-sans cursor-pointer">
35                     <option value="">Semua Platform</option>
36                     <option value="Online" {{ request('platform') == 'Online' ? 'selected' : '' }}>Online</option>
37                     <option value="Offline" {{ request('platform') == 'Offline' ? 'selected' : '' }}>Offline</option>
38                 </select>
39                 <select name="status" onchange="this.form.submit()"
40                     class="h-9 px-2.5 border border-page-bg dark:border-page-bg-dark rounded-lg bg-page-bg dark:bg-page-bg-dark text-text
```

Gambar 5.7 Source Code Penjadwalan.blade.php

Tabel 5.14 Kode operator-index.blade.php halaman peminjaman

operator-index.blade.php

resources > views > dashboard > peminjaman > operator-index.blade.php

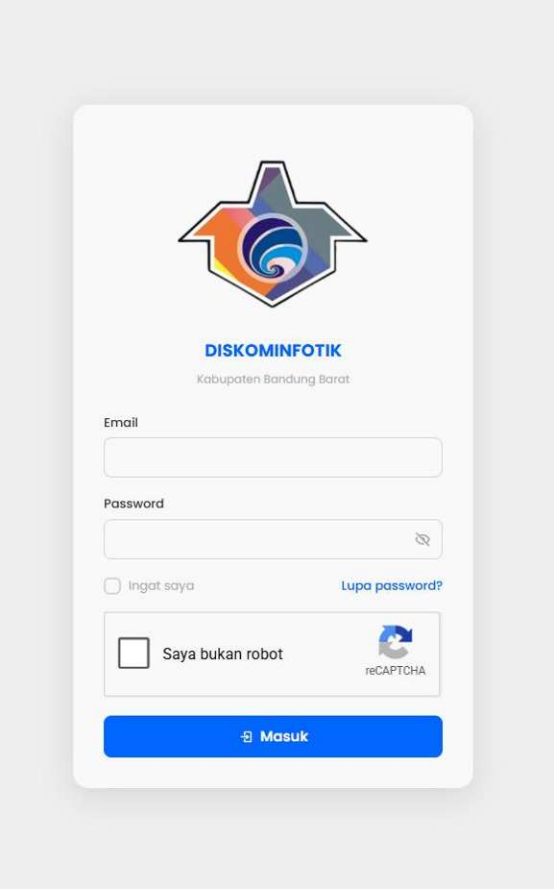
```
1  @extends('layouts.app')
2  @section('title', 'Peminjaman Saya')
3  @section('sidebar-menu') <x-sidebar /> @endsection
4
5  @section('content')
6      @php
7          $roleAktif = auth()->user()->role;
8          $bisaUbah  = auth()->user()->punyaAkses('peminjaman', 'ubah');
9      @endphp
10     <main class="w-full pt-9 px-6 pb-9 font-sans max-h-[calc(100vh-56px)] overflow-y-auto overflow-x-hidden">
11         <div class="flex items-center justify-between gap-4 flex-wrap mb-5">
12             <div>
13                 <h1 class="text-4xl font-semibold mb-2.5 text-text dark:text-text-dark">Peminjaman Saya</h1>
14             </div>
15             @if(auth()->user()->punyaAkses('peminjaman', 'tambah'))
16                 <a href="{{ route($roleAktif . '.peminjaman.create') }}"
17                     class="h-9 px-4 rounded-full bg-primary text-surface dark:text-surface-dark flex justify-center items-center gap-2.5 font-medium">
18                     <i class="bx bx-plus"></i><span class="text">Ajukan Peminjaman</span>
19                 </a>
20             @endif
21         </div>
22
23         <div class="bg-surface dark:bg-surface-dark rounded-[10px] py-3.5 px-4 mb-4 flex items-center gap-2.5 flex-wrap shadow-[0_2px_8px_rgba(0,0,0,0.
           04)]">
24             <form method="GET" action="{{ route($roleAktif . '.peminjaman.index') }}" class="contents">
25                 <div class="relative flex-1 min-w-[180px] max-w-[300px]">
26                     <i class="bx bx-search absolute left-2.5 top-1/2 -translate-y-1/2 text-text-muted text-base pointer-events-none"></i>
27                     <input type="text" name="search" data-live-search="#hasil-peminjaman" autocomplete="off"
28                         placeholder="Cari keperluan, nama alat..." value="{{ request('search') }}"
29                         class="w-full h-9 pl-[34px] pr-3 border border-page-bg dark:border-page-bg-dark rounded-lg bg-page-bg dark:bg-page-bg-dark
                           text-text dark:text-text-dark text-[13px] font-sans transition-colors duration-200 focus:border-primary focus:outline-none
                           focus:bg-surface dark:focus:bg-surface-dark">
30                 </div>
31                 <select name="status" onchange="this.form.submit()"
32                     class="h-9 px-2.5 border border-page-bg dark:border-page-bg-dark rounded-lg bg-page-bg dark:bg-page-bg-dark text-text
                       dark:text-text-dark text-[13px] font-sans cursor-pointer">
33                     <option value="">Semua Status</option>
34                     <option value="diajukan" {{ request('status') == 'diajukan' ? 'selected' : '' }}>Menunggu</option>
35                     <option value="disetujui" {{ request('status') == 'disetujui' ? 'selected' : '' }}>Disetujui</option>
36                     <option value="ditolak" {{ request('status') == 'ditolak' ? 'selected' : '' }}>Ditolak</option>
37                     <option value="dikembalikan" {{ request('status') == 'dikembalikan' ? 'selected' : '' }}>Dikembalikan
38                     </option>
39                     <option value="dibatalkan" {{ request('status') == 'dibatalkan' ? 'selected' : '' }}>Dibatalkan</option>
40                 </select>
```

Gambar 5.8 Source Code operator-index.blade.php

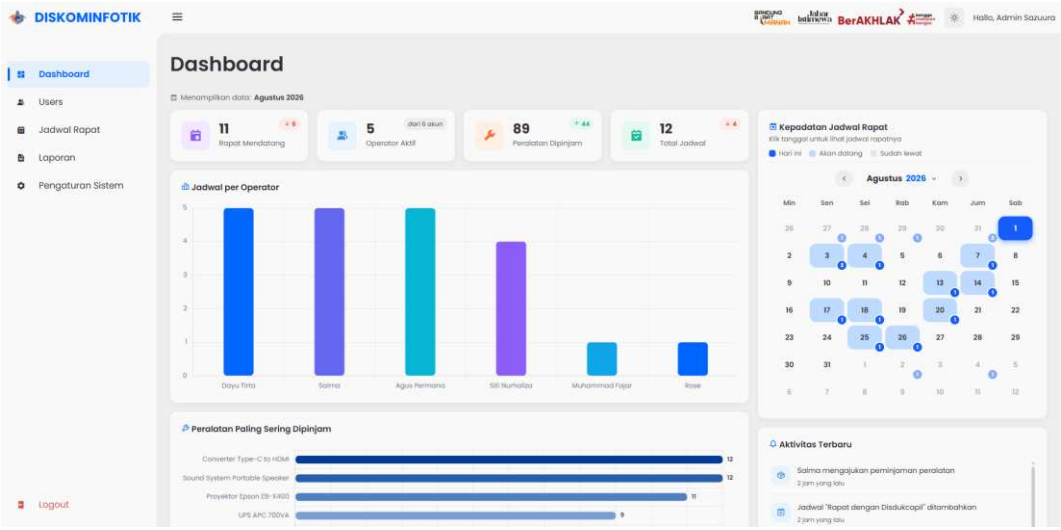
5.4 Implementasi Tampilan Antarmuka Pengguna

Tahap ini menyajikan hasil implementasi antarmuka pengguna yang dibangun berdasarkan rancangan yang telah disusun sebelumnya. Antarmuka tersebut merupakan bagian aplikasi berbasis web yang berinteraksi langsung dengan pengguna

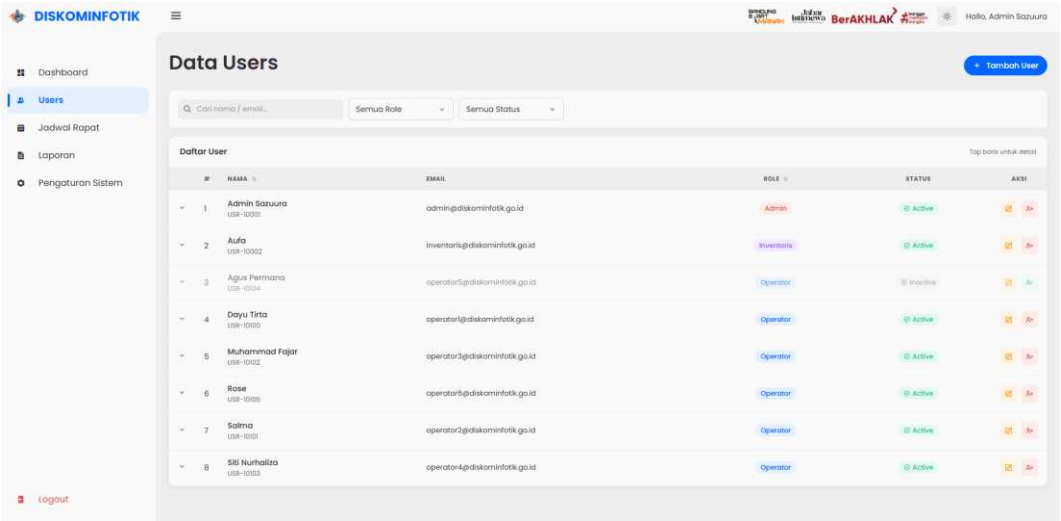
Tabel 5.15 Antarmuka Halaman Login

Antarmuka
 *Gambar 5.9 Antarmuka Halaman Login*
Deskripsi
Pada gambar di atas ditampilkan antarmuka halaman login yang digunakan sebagai akses awal ke dalam sistem. Pengguna diminta memasukkan alamat email dan kata sandi yang telah terdaftar untuk melakukan proses autentikasi sebelum dapat mengakses aplikasi

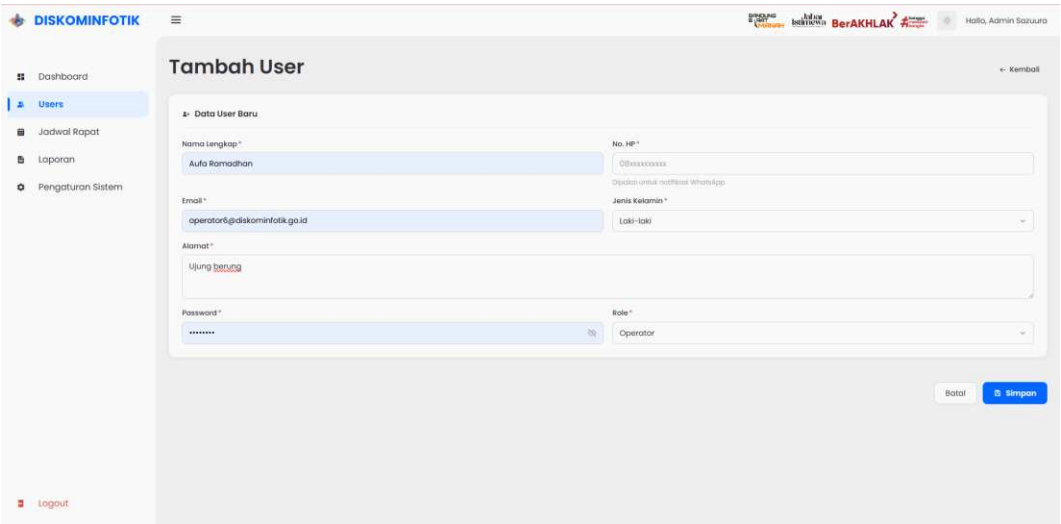
Tabel 5.16 Antarmuka Halaman Dashboard Admin

Antarmuka
 *Gambar 5.10 Antarmuka Halaman Dashboard Admin*
Deskripsi
Gambar 5.10 menunjukkan antarmuka halaman Dashboard Admin yang berfungsi sebagai halaman utama setelah proses login berhasil dilakukan. Halaman ini menyajikan ringkasan informasi berupa jumlah rapat mendatang, operator aktif, peralatan yang sedang dipinjam, dan total jadwal rapat. Selain itu, ditampilkan grafik jadwal per operator, grafik peralatan yang paling sering dipinjam, kalender kegiatan rapat, serta aktivitas terbaru pengguna. Informasi tersebut digunakan untuk membantu admin memantau kondisi sistem dan mendukung pengambilan keputusan dalam pengelolaan jadwal rapat dan peralatan

Tabel 5.17 Antarmuka Halaman Users

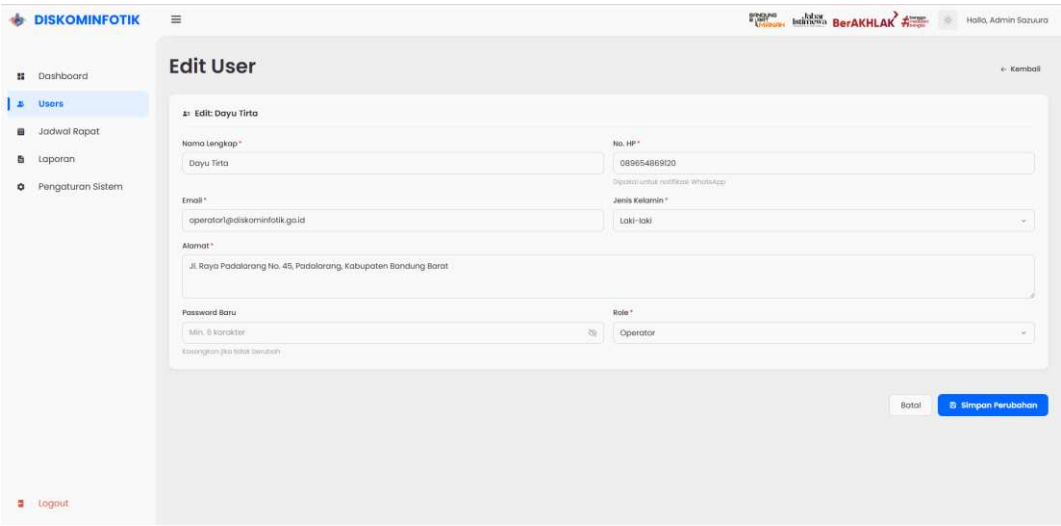
Antarmuka
 *Gambar 5.11 Antarmuka Halaman Users*
Deskripsi
Halaman Data Users digunakan oleh admin untuk mengelola data pengguna aplikasi. Halaman ini menyediakan daftar pengguna beserta fitur pencarian, penyaringan berdasarkan peran dan status, penambahan pengguna, beserta pengelolaan data pengguna

Tabel 5.18 Antarmuka Halaman Tambah User

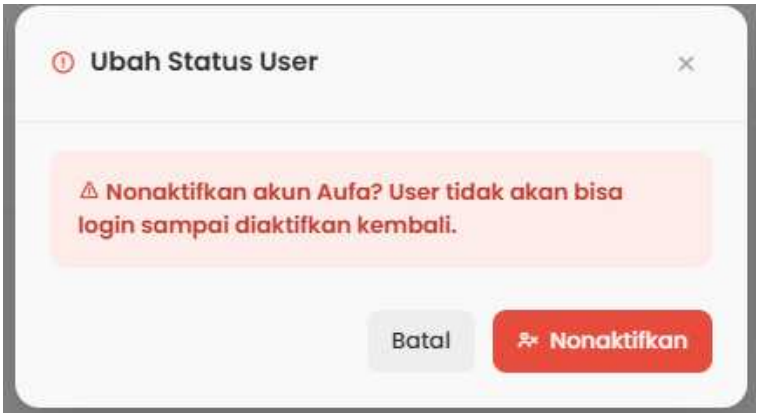
Antarmuka
 *Gambar 5.12 Antarmuka Halaman Tambah User*
Deskripsi

Halaman Tambah User digunakan untuk menambahkan data pengguna baru ke dalam sistem. Pada halaman ini, admin mengisi informasi pengguna berupa nama lengkap, alamat email, nomor telepon, jenis kelamin, alamat, kata sandi, dan peran pengguna sebelum data disimpan

Tabel 5.19 Antarmuka Halaman Ubah User

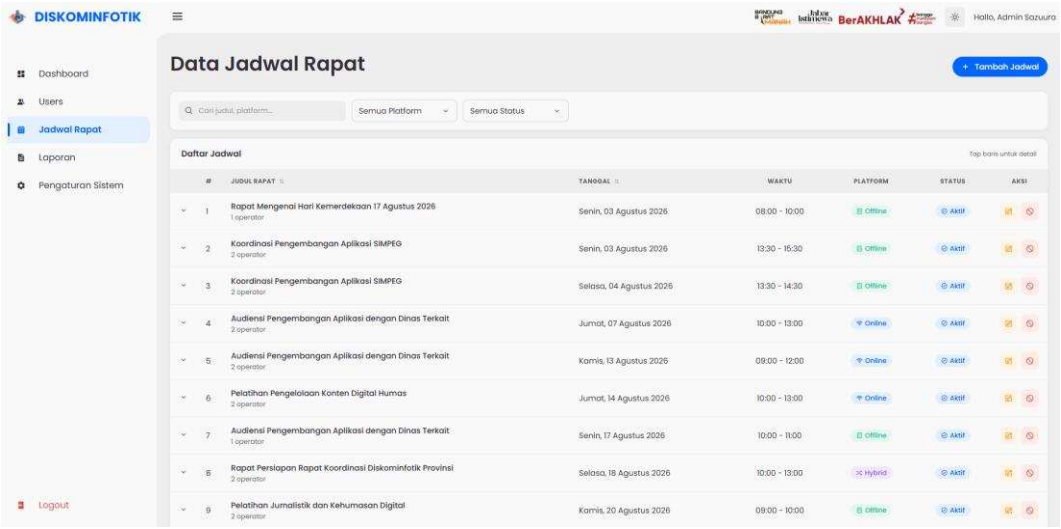
Antarmuka
 *Gambar 5.13 Antarmuka Halaman Ubah User*
Deskripsi
Halaman Ubah User digunakan untuk mengubah data pengguna yang telah terdaftar di dalam sistem. Pada halaman ini, admin dapat memperbarui informasi pengguna berupa nama lengkap, alamat email, nomor telepon, jenis kelamin, alamat, kata sandi, dan peran pengguna sebelum perubahan data disimpan

Tabel 5.20 Antarmuka Nonaktifkan User

Antarmuka
 *Gambar 5.14 Antarmuka Nonaktifkan User*
Deskripsi

Modal Nonaktifkan User digunakan untuk melakukan konfirmasi penonaktifan akun pengguna. Melalui modal ini, admin dapat mengonfirmasi tindakan penonaktifan sehingga status pengguna berubah menjadi tidak aktif dan pengguna tidak lagi dapat mengakses sistem

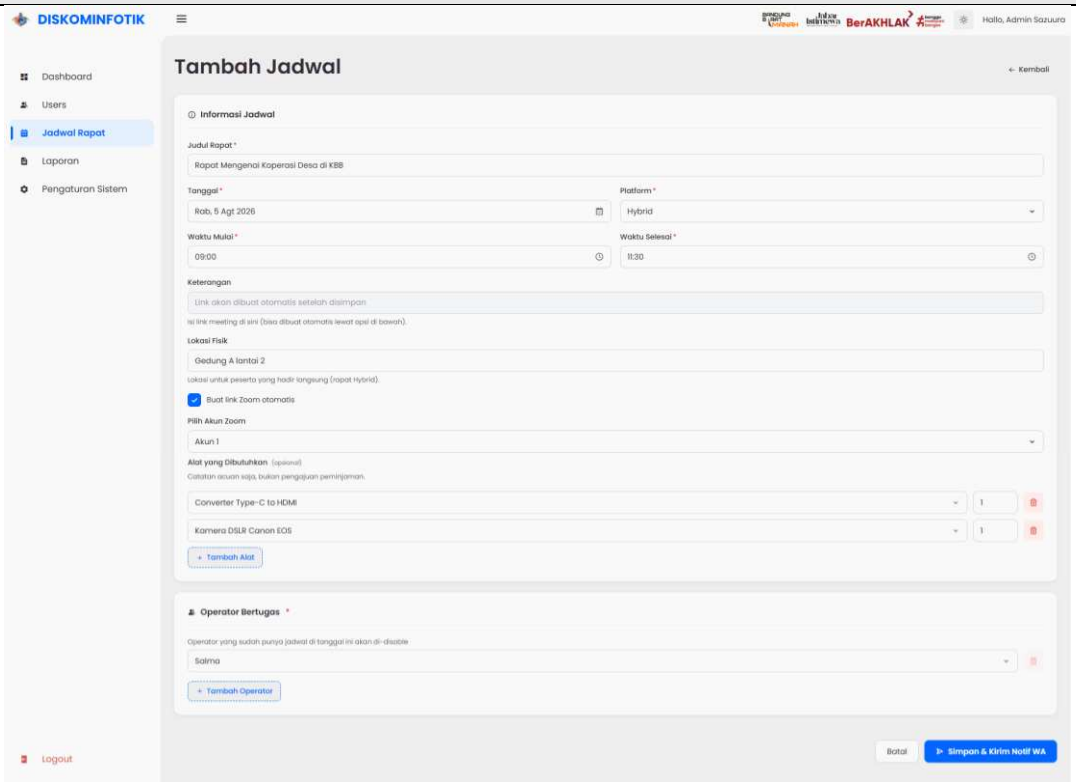
Tabel 5.21 Antarmuka Halaman Jadwal Rapat

Antarmuka


Gambar 5.15 Antarmuka Halaman Jadwal Rapat

Deskripsi
Halaman Data Jadwal Rapat digunakan untuk mengelola data jadwal rapat pada sistem. Halaman ini menampilkan daftar jadwal rapat, menyediakan fitur pencarian dan penyaringan berdasarkan platform maupun status, serta dilengkapi tombol untuk menambah, mengubah, melihat detail, dan membatalkan jadwal rapat

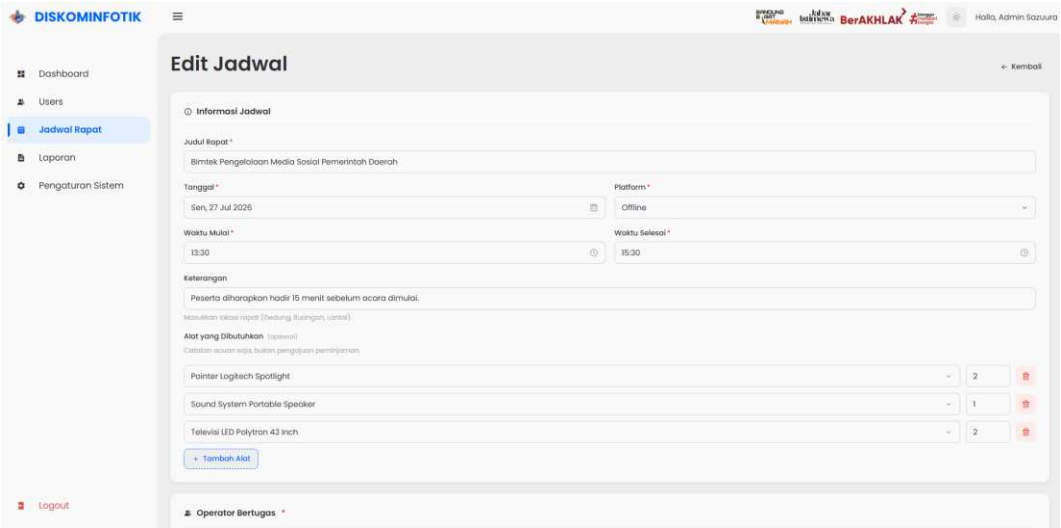
Tabel 5.22 Antarmuka Halaman Tambah Jadwal

Antarmuka


Gambar 5.16 Antarmuka Halaman Tambah Jadwal

Deskripsi
Halaman Tambah Jadwal digunakan untuk menambahkan data jadwal rapat baru ke dalam sistem. Pada halaman ini, admin mengisi informasi jadwal rapat, menentukan platform pelaksanaan, memilih akun Zoom, menambahkan peralatan yang dibutuhkan, menetapkan operator yang bertugas, kemudian menyimpan data jadwal rapat

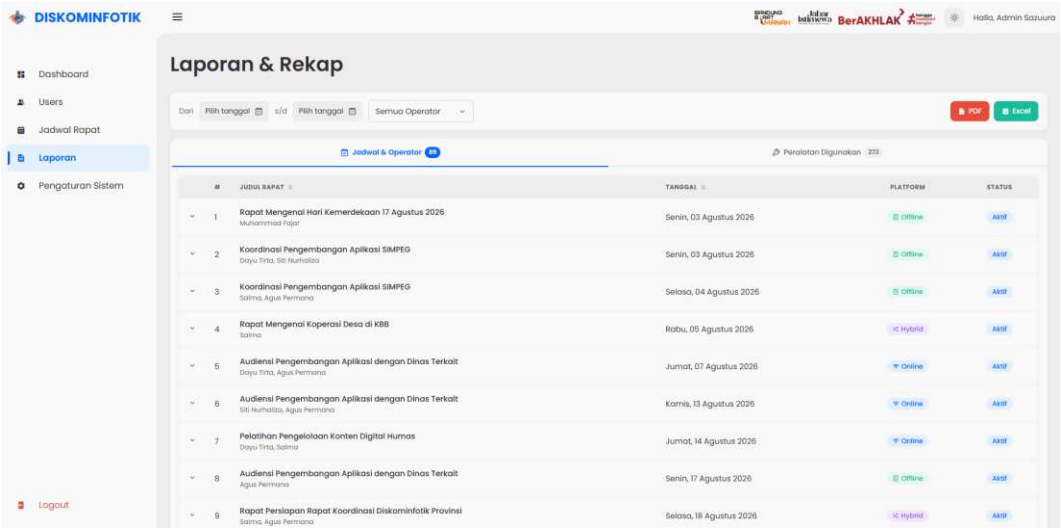
Tabel 5.23 Antarmuka Halaman Ubah Jadwal

Antarmuka
 *Gambar 5.17 Antarmuka Halaman Ubah Jadwal*
Deskripsi
Halaman Ubah Jadwal digunakan untuk mengubah data jadwal rapat yang telah tersimpan di dalam sistem. Pada halaman ini, admin dapat memperbarui informasi jadwal rapat, mengubah platform pelaksanaan, memilih akun Zoom, menyesuaikan peralatan yang dibutuhkan, menetapkan operator yang bertugas, kemudian menyimpan perubahan data jadwal rapat

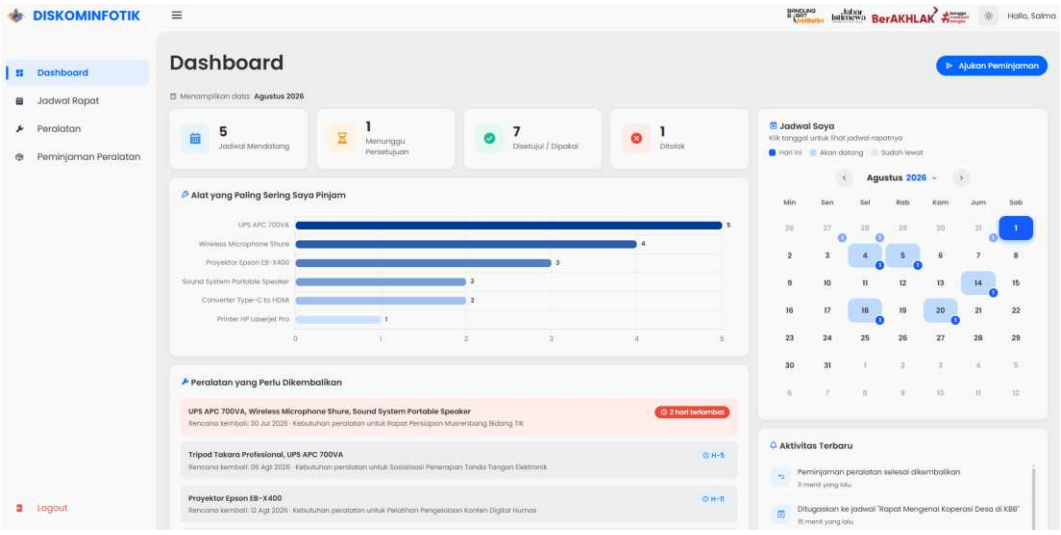
Tabel 5.24 Antarmuka Batalkan Jadwal

Antarmuka
 Gambar 5.18 Antarmuka Batalkan Jadwal
Deskripsi
Modal Batalkan Jadwal digunakan untuk melakukan pembatalan jadwal rapat yang telah tersimpan di dalam sistem. Pada modal ini, admin mengisi alasan pembatalan, kemudian mengonfirmasi proses pembatalan sehingga status jadwal diperbarui menjadi batal dan notifikasi WhatsApp dikirim kepada seluruh operator yang bertugas

Tabel 5.25 Antarmuka Halaman Laporan

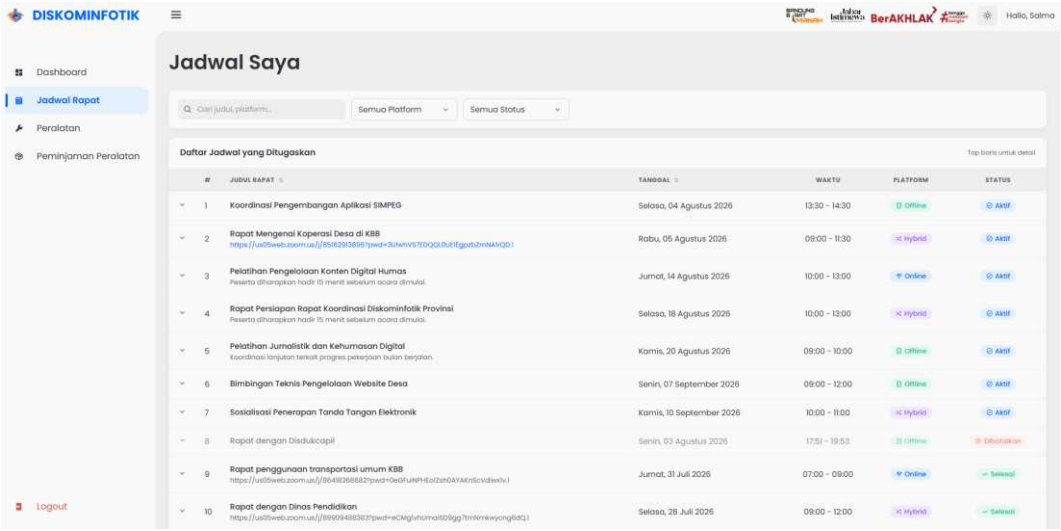
Antarmuka
 *Gambar 5.19 Antarmuka Halaman Laporan*
Deskripsi
Halaman Laporan & Rekap digunakan untuk menampilkan data laporan jadwal rapat dan penggunaan peralatan. Pada halaman ini tersedia fitur penyaringan data berdasarkan rentang tanggal dan operator, serta fasilitas untuk mengekspor laporan ke dalam format PDF dan Excel

Tabel 5.26 Antarmuka Halaman Dashboard Operator

Antarmuka
 *Gambar 5.20 Antarmuka Halaman Dashboard Operator*
Deskripsi

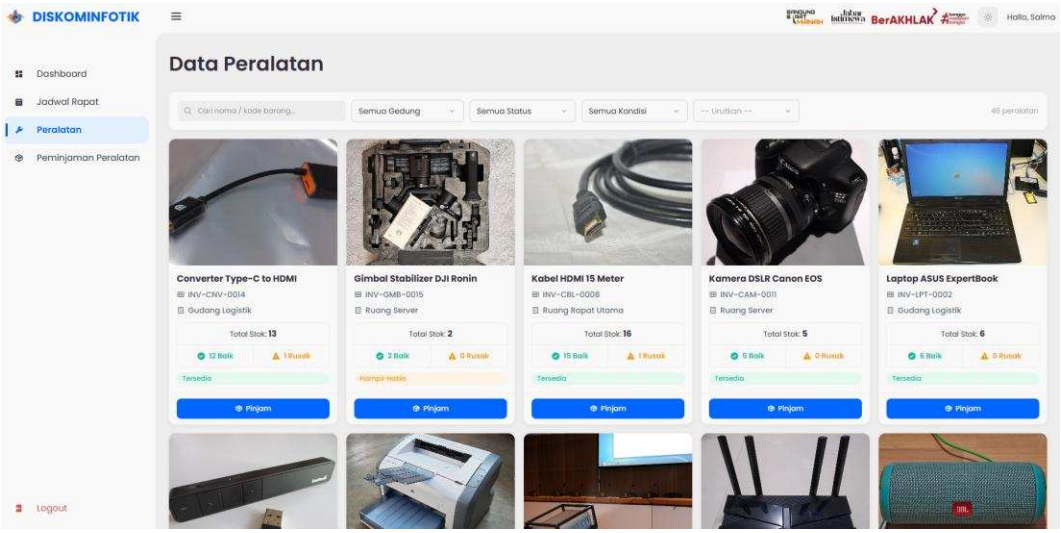
Halaman Dashboard Operator digunakan untuk menampilkan ringkasan informasi yang berkaitan dengan aktivitas operator. Pada halaman ini ditampilkan jumlah jadwal rapat mendatang, status pengajuan peminjaman peralatan, grafik peralatan yang paling sering dipinjam, daftar peralatan yang perlu dikembalikan, kalender jadwal rapat, dan aktivitas terbaru

Tabel 5.27 Antarmuka Halaman Jadwal Rapat Saya

Antarmuka
 *Gambar 5.21 Antarmuka Halaman Jadwal Saya*
Deskripsi
Halaman Jadwal Saya digunakan untuk menampilkan daftar jadwal rapat yang ditugaskan kepada operator. Pada halaman ini ditampilkan informasi judul rapat, tanggal, waktu pelaksanaan, platform, dan status jadwal sehingga operator dapat mengetahui jadwal yang menjadi tanggung jawabnya

Tabel 5.28 Antarmuka Halaman Peralatan

Antarmuka

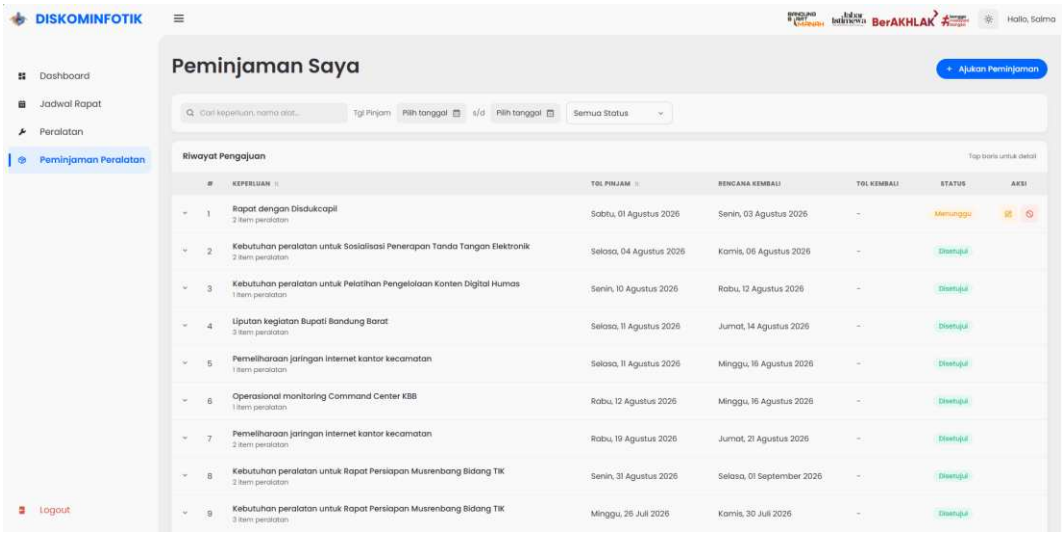


Gambar 5.22 Antarmuka Halaman Peralatan

Deskripsi
Halaman Daftar Peralatan digunakan untuk menampilkan informasi peralatan yang tersedia untuk diajukan peminjamannya oleh operator. Halaman ini menyediakan fitur pencarian, penyaringan berdasarkan gedung, status, dan kondisi peralatan, serta tombol peminjaman pada setiap peralatan yang tersedia

Tabel 5.29 Antarmuka Halaman Peminjaman Peralatan

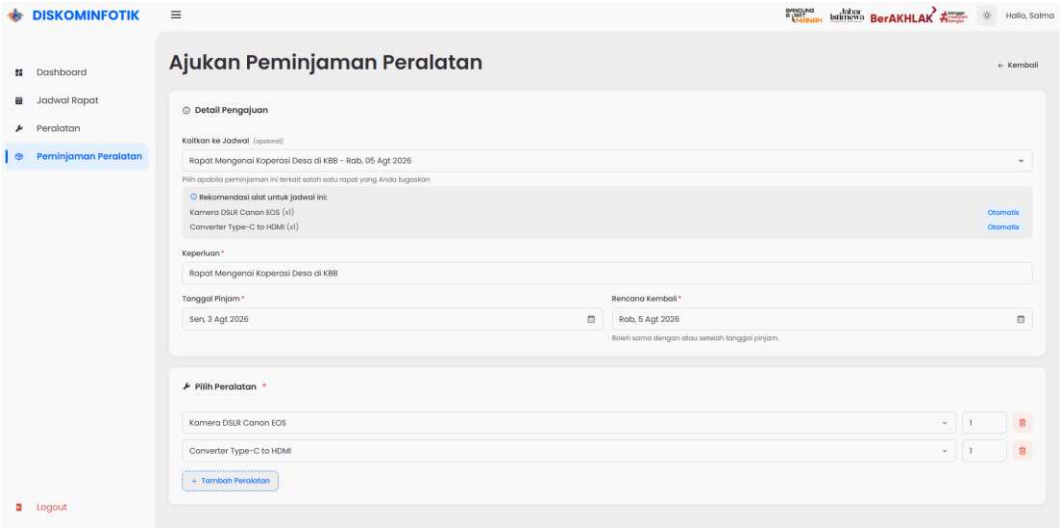
Antarmuka



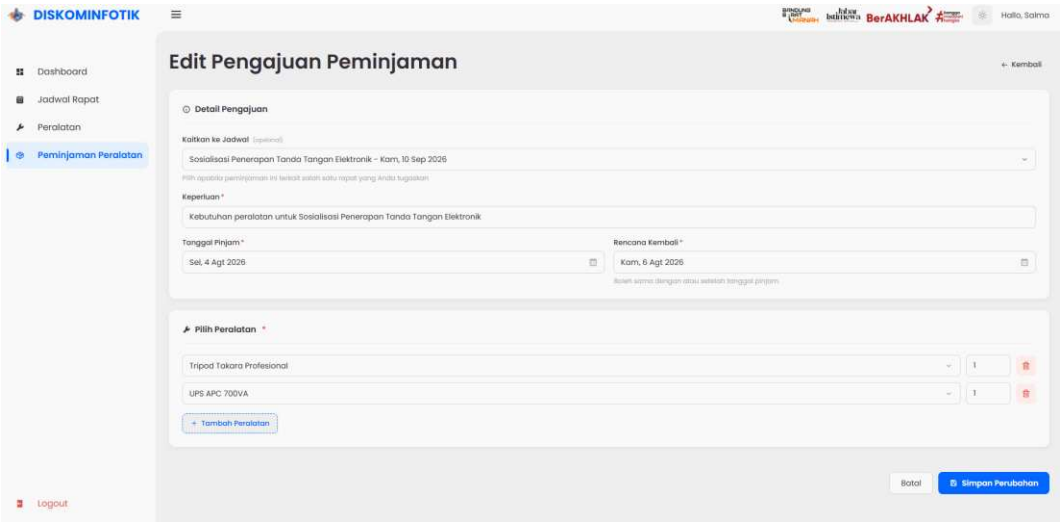
Gambar 5.23 Antarmuka Halaman Peminjaman Peralatan

Deskripsi
Halaman Peminjaman Saya digunakan untuk menampilkan riwayat pengajuan peminjaman peralatan oleh operator. Halaman ini menyediakan informasi keperluan peminjaman, tanggal peminjaman, rencana pengembalian, tanggal pengembalian, status pengajuan, serta aksi untuk mengubah atau membatalkan pengajuan sesuai ketentuan yang berlaku

Tabel 5.30 Antarmuka Halaman Pengajuan Peminjaman Peralatan

Antarmuka
 *Gambar 5.24 Antarmuka Halaman Ajukan Peminjaman Peralatan*
Deskripsi
Halaman Ajukan Peminjaman Peralatan digunakan untuk mengajukan peminjaman peralatan yang dibutuhkan dalam pelaksanaan rapat maupun diluar rapat. Pada halaman ini, operator memilih jadwal rapat, mengisi informasi pengajuan, menentukan tanggal peminjaman dan rencana pengembalian, serta memilih peralatan yang akan dipinjam sebelum pengajuan disimpan

Tabel 5.31 Antarmuka Halaman Ubah Pengajuan Peminjaman Peralatan

Antarmuka
 *Gambar 5.25 Antarmuka Halaman Ubah Pengajukan Peminjaman Peralatan*
Deskripsi
Halaman Ubah Pengajuan Peminjaman Peralatan digunakan untuk mengubah data pengajuan peminjaman peralatan yang telah tersimpan di dalam sistem. Pada halaman ini, operator dapat memperbarui informasi pengajuan, mengubah tanggal peminjaman dan rencana pengembalian, serta menyesuaikan peralatan yang akan dipinjam sebelum perubahan pengajuan disimpan

Tabel 5.32 Antarmuka Batalkan Pengajuan Peminjaman Peralatan

Antarmuka
Gambar 5.26 Antarmuka Batalkan Pengajukan Peminjaman Peralatan
Deskripsi
Modal Batalkan Pengajuan Peminjaman Peralatan digunakan untuk melakukan pembatalan pengajuan peminjaman peralatan. Pada modal ini, operator mengisi alasan pembatalan, kemudian mengonfirmasi proses pembatalan sehingga status pengajuan diperbarui menjadi batal dan notifikasi WhatsApp dikirim kepada pihak inventaris

5.5 Pengujian Perangkat Lunak

Pengujian terhadap aplikasi berbasis web ini dilakukan menggunakan metode Black-Box Testing, yaitu pengujian yang dilaksanakan dengan hanya mengamati hasil eksekusi berdasarkan data uji serta memeriksa aspek fungsional dari perangkat lunak. Hasil pengujian perangkat lunak yang telah dilakukan disajikan pada tabel berikut.

Tabel 5.33 Pengujian Perangkat Lunak

Test Scenario	*Test Case*	Test Step	Test Dasar	Expected Result	Actual Result	Pass / Fail
TS-01 Login	TC-01 Login berhasil (kredensial valid)	1. Buka halaman login 2. Isi email & password terdaftar 3. Isi captcha 4. Klik tombol login	Email: admin@diskominfotik.go.id Password: password captcha	Pengguna berhasil login ke halaman dashboard sesuai role (admin atau operator), muncul pesan sukses login	Pengguna berhasil login ke halaman dashboard sesuai role (admin atau operator), muncul pesan sukses login	Pass
	TC-02 Login gagal (password salah)	1. Buka halaman login 2. Isi email terdaftar, password salah 3. Isi captcha 4. Klik tombol login	Email: admin@diskominfotik.go.id Password: password captcha	Pengguna tetap berada halaman login, muncul pesan email/password tidak sesuai	Pengguna tetap berada halaman login, muncul pesan email/password tidak sesuai	Pass
	TC-03 Login dengan akun berstatus nonaktif	1. Login pakai akun yang statusnya inactive 2. Klik tombol login	Email: operator2@diskominfotik.go.id Password: password captcha	Pengguna tidak bisa login, muncul pesan "Akun Anda telah dinonaktifkan. Silakan hubungi administrator."	Pengguna tidak bisa login, muncul pesan "Akun Anda telah dinonaktifkan. Silakan hubungi administrator."	Pass
	TC-04 Login tanpa mengisi captcha	1. Isi email & password valid 2. Kosongkan captcha 3. Klik tombol login	Email: admin@diskominfotik.go.id Password: password	Muncul pesan error "Captcha wajib diisi."	Muncul pesan error "Captcha wajib diisi."	Pass
	TC-05 Login dengan	1. Buka halaman login	Email: admin@diskominfotik.go.id Password: password Ingat saya	Pengguna berhasil login ke halaman dashboard sesuai	Pengguna berhasil login ke halaman dashboard sesuai	Pass

	mengisi “ingat saya”	2. Isi email & password terdaftar 3. Isi ingat saya 4. Isi captcha 5. Klik tombol login	captcha	role (admin atau operator), dan Ketika menutup browser lalu Kembali lagi, tidak harus login	role (admin atau operator), dan Ketika menutup browser lalu Kembali lagi, tidak harus login	
TS-02 Lupa Password OTP WhatsApp	TC-01 Kirim OTP ke email terdaftar	1. Buka halaman lupa password 2. Isi email yang terdaftar & punya nomor WA 3. Klik kirim kode	Email: admin@diskominfotik.go.id	Kode OTP 6 digit terkirim ke WhatsApp terdaftar, muncul pesan "Kode verifikasi sudah dikirim ke WhatsApp terdaftar."	Kode OTP 6 digit terkirim ke WhatsApp terdaftar, muncul pesan "Kode verifikasi sudah dikirim ke WhatsApp terdaftar."	Pass
	TC-02 Kirim OTP ke email tidak terdaftar	1. Isi email yang tidak ada di sistem 2. Klik kirim kode	Email: salma250803@gmail.com	Ditolak, muncul pesan "Email tidak terdaftar."	Ditolak, muncul pesan "Email tidak terdaftar."	Pass
	TC-03 Kirim OTP berulang lebih dari 3x dalam 15 menit	1. Kirim OTP ke email yang sama sebanyak 4x berturut-turut	Kirim ulang kode OTP	Pada percobaan ke-4 ditolak, muncul pesan "Terlalu banyak percobaan. Coba lagi dalam 900 detik." Sekitar 15 menit	Pada percobaan ke-4 ditolak, muncul pesan "Terlalu banyak percobaan. Coba lagi dalam 900 detik." Sekitar 15 menit	Pass
	TC-04 Verifikasi dengan kode OTP salah	1. Minta kode OTP 2. Masukkan kode OTP asal (bukan yang dikirim) 3. Klik verifikasi	Kode OTP asli: 654320 Kode OTP palsu: 659231	Ditolak, muncul pesan "Kode verifikasi salah."	Ditolak, muncul pesan "Kode verifikasi salah."	Pass
	TC-05 Verifikasi OTP yang sudah kedaluwarsa (>10 menit)	1. Minta kode OTP 2. Tunggu lebih dari 10 menit 3. Masukkan kode OTP, klik verifikasi	Kode OTP asli: 654320 Percobaan lebih dari 10 menit	Ditolak, muncul pesan "Kode verifikasi sudah kedaluwarsa. Silakan minta kode baru."	Ditolak, muncul pesan "Kode verifikasi sudah kedaluwarsa. Silakan minta kode baru."	Pass

	TC-06 Reset password dengan OTP valid	1. Masukkan Email yang benar 2. Masukkan kode OTP yang benar 3. Isi password baru (>=8 karakter) & konfirmasi 4. Klik simpan	Email: admin@diskominfotik.go.id Kode OTP asli: 160033 Password baru: password1	Password berhasil diperbarui, token OTP terhapus, muncul pesan "Kata sandi berhasil diperbarui."	Password berhasil diperbarui, token OTP terhapus, muncul pesan "Kata sandi berhasil diperbarui."	Pass
TS-03 Kelola Pengguna (Admin)	TC-01 Tambah pengguna baru	1. Masuk halaman tambah pengguna 2. Isi nama, jenis kelamin, alamat, no HP unik, email unik, password >=6 karakter, role 3. Klik simpan	Nama Lengkap: Salma Az Zahra No. HP: 089663370967 Email: adminzura@diskominfotik.go.id Jenis Kelamin: Perempuaan Alamat: Setia Budi, Kota Bandung Password: password Role: admin	Data pengguna berhasil tersimpan dengan status default "active", tampil di daftar pengguna, dan muncul pesan “User berhasil ditambahkan”	Data pengguna berhasil tersimpan dengan status default "active", tampil di daftar pengguna, dan muncul pesan “User berhasil ditambahkan”	Pass
	TC-02 Tambah pengguna dengan email/no HP yang sudah dipakai	1. Isi form tambah pengguna dengan email atau no HP yang sudah terdaftar 2. Klik simpan	Nama Lengkap: Salma Az Zahra No. HP: 089663370967 Email: adminzura@diskominfotik.go.id Jenis Kelamin: Perempuaan Alamat: Setia Budi, Kota Bandung Password: password Role: admin	Ditolak, muncul pesan error validasi data sudah digunakan	Ditolak, muncul pesan error validasi data sudah digunakan	Pass
	TC-03 Ubah data pengguna tanpa mengganti password	1. Masuk halaman edit pengguna 2. Ubah nama/role 3. Kosongkan field password 4. Klik simpan	Nama Lengkap: Salma Az Zahra No. HP: 089663370967 Email: adminzura@diskominfotik.go.id Jenis Kelamin: Perempuaan Alamat: Setia Budi, Kota Bandung	Data pengguna berhasil diperbarui, password lama tetap berlaku. Muncul pesan “User berhasil diperbarui”	Data pengguna berhasil diperbarui, password lama tetap berlaku. Muncul pesan “User berhasil diperbarui”	Pass

			Password: Role: admin			
	TC-04 Menonaktif kan akun milik sendiri	1. Login sebagai admin 2. Klik nonaktifkan pada akun admin yang sedang login	Menekan tombol aksi nonaktifkan akun	Ditolak, muncul pesan "Tidak dapat menonaktifkan akun Anda sendiri."	Ditolak, muncul pesan "Tidak dapat menonaktifkan akun Anda sendiri."	Pass
	TC-05 Mengaktifk an kembali pengguna nonaktif	1. Cari pengguna berstatus inactive 2. Klik tombol aktifkan	Menekan tombol aksi aktifkan akun	Status pengguna berubah menjadi "active", muncul pesan "User berhasil diaktifkan."	Status pengguna berubah menjadi "active", muncul pesan "User berhasil diaktifkan."	Pass
TS-04 Kelola Jadwal Rapat (Admin)	TC-01 Tambah jadwal tanpa link Zoom otomatis	1. Masuk halaman tambah jadwal 2. Isi judul, tanggal (>= hari ini), jam mulai/selesai, platform 3. Pilih alat yang dibutuhkan 4. Pilih minimal 1 operator 5. Klik simpan	Judul Rapat: Rapat koordinasi Kab Bandung Tanggal: 07/13/2026 Platform: Offline Waktu mulai: 10:00 AM Waktu selesai: 01:00 PM Keterangan: Tempat ruang rapat gedung A, Untuk mempersiapkan manajemen kelola media Kab Bandung Alat yang dibutuhkan: Converter Type-C to HDMI Jumlah: 1 Operator: Aufa Ramadhan (Operator)	Data penjadwalan berhasil ditambahkan, notifikasi WA terkirim ke tiap operator yang ditugaskan. Muncul pesan “Jadwal berhasil ditambahkan dan notifikasi WA telah dikirim.”	Data penjadwalan berhasil ditambahkan, notifikasi WA terkirim ke tiap operator yang ditugaskan. Muncul pesan “Jadwal berhasil ditambahkan dan notifikasi WA telah dikirim.”	Pass
	TC-02 Tambah jadwal dengan jam selesai lebih awal dari jam mulai	1. Isi waktu_selesai lebih awal dari waktu_mulai 2. Klik simpan	Judul Rapat: Rapat bulanan koordinasi Kab Bandung Tanggal: 07/24/2026 Platform: Offline Waktu mulai: 02:00 PM Waktu selesai: 01:00 PM Keterangan: Ruang rapat gedung B LT 1 Alat yang dibutuhkan: Converter Type-C to HDMI Jumlah: 1 Operator: Aufa Ramadhan (Operator)	Ditolak, muncul error validasi jam selesai harus setelah jam mulai. Muncul pesan “waktu selesai harus tanggal setelah waktu mulai”	Ditolak, muncul error validasi jam selesai harus setelah jam mulai. Muncul pesan “waktu selesai harus tanggal setelah waktu mulai”	Pass

	TC-03 Tambah jadwal dengan operator yang bentrok jadwal	1. Pilih operator yang sudah punya jadwal lain di tanggal & jam yang sama	Judul Rapat: Rapat koordinasi Kab Bandung Tanggal: 07/13/2026 Platform: Offline Waktu mulai: 09:00 AM Waktu selesai: 12:00 PM Keterangan: Tempat ruang rapat gedung A, Untuk mempersiapkan manajemen kelola media Kab Bandung Alat yang dibutuhkan: Converter Type-C to HDMI Jumlah: 1 Operator: Aufa Ramadhan (Operator) (*disabled*)	Ditolak, muncul pesan error operator tersebut bentrok jadwal	Ditolak, muncul pesan error operator tersebut bentrok jadwal	Pass
	TC-04 Tambah jadwal dengan link Zoom otomatis (akun tersedia)	1. Centang "Buat link Zoom otomatis" 2. Pilih akun otomatis/manual 3. Klik simpan	Judul Rapat: Rapat online tahunan Diskominfotik Tanggal: 07/31/2026 Platform: Online (Zoom) Waktu mulai: 10:00 AM Waktu selesai: 12:00 PM Keterangan: Otomatis (Link Zoom) Ceklis buat link Zoom otomatis Akun Zoom: akun 1 Alat yang dibutuhkan: Converter Type-C to HDMI Jumlah: 1 Operator: Kanda Wahyuni (Operator)	Data penjadwalan berhasil ditambahkan besertan dengan link Zoom meeting, notifikasi WA terkirim ke tiap operator yang ditugaskan. Muncul pesan “Jadwal berhasil ditambahkan dan notifikasi WA telah dikirim.”	Data penjadwalan berhasil ditambahkan besertan dengan link Zoom meeting, notifikasi WA terkirim ke tiap operator yang ditugaskan. Muncul pesan “Jadwal berhasil ditambahkan dan notifikasi WA telah dikirim.”	Pass
	TC-05 Memilih akun Zoom yang sedang bentrok jadwal Zoom lain	1. Buka form tambah/ubah jadwal 2. Perhatikan dropdown pilihan akun Zoom pada tanggal & jam yang bentrok	Judul Rapat: Rapat online divisi jaringan Tanggal: 07/31/2026 Platform: Online (Zoom) Waktu mulai: 10:00 AM Waktu selesai: 12:00 PM Keterangan: Otomatis (Link Zoom) Ceklis buat link Zoom otomatis Akun Zoom: akun 1 (disable)	Pilihan akun Zoom yang bentrok tetap tampil di dropdown, tapi menjadi disable (tidak bisa dipilih) dan diberi label "(bentrok jadwal)" pada opsinya	Pilihan akun Zoom yang bentrok tetap tampil di dropdown, tapi menjadi disable (tidak bisa dipilih) dan diberi label "(bentrok jadwal)" pada opsinya	Pass

			Alat yang dibutuhkan: Converter Type-C to HDMI Jumlah: 1 Operator: Samsul Haryanto (Operator)			
	TC-06 Ubah jadwal tanpa mengubah judul/waktu /platform	1. Masuk halaman edit jadwal 2. Ubah hanya daftar peralatan referensi 3. Klik simpan	Judul Rapat: Rapat online divisi jaringan Tanggal: 07/31/2026 Platform: Online (Zoom) Waktu mulai: 10:00 AM Waktu selesai: 12:00 PM Keterangan: Otomatis (Link Zoom) Ceklis buat link Zoom otomatis Akun Zoom: akun 1 (disable) Alat yang dibutuhkan: Gimbal Stabilizer DJI Ronin Jumlah: 1 Operator: Samsul Haryanto (Operator)	Data penjadwalan berhasil diperubarui, TIDAK mengirim notifikasi WA "jadwal diubah" ke operator. Muncul pesan “Jadwal berhasil diperbarui”	Data penjadwalan berhasil diperubarui, TIDAK mengirim notifikasi WA "jadwal diubah" ke operator. Muncul pesan “Jadwal berhasil diperbarui”	Pass
	TC-07 Ubah jadwal dengan mengubah jam rapat	1. Masuk halaman edit jadwal 2. Ubah jam mulai/selesai 3. Klik simpan	Judul Rapat: Rapat online divisi jaringan Tanggal: 07/31/2026 Platform: Online (Zoom) Waktu mulai: 07:00 AM Waktu selesai: 09:00 AM Keterangan: Otomatis (Link Zoom) Ceklis buat link Zoom otomatis Akun Zoom: akun 1 (disable) Alat yang dibutuhkan: Gimbal Stabilizer DJI Ronin Jumlah: 1 Operator: Samsul Haryanto (Operator)	Data penjadwalan berhasil diperubarui, menerima notifikasi WA "jadwal diubah" ke operator. Muncul pesan “Jadwal berhasil diperbarui”	Data penjadwalan berhasil diperubarui, menerima notifikasi WA "jadwal diubah" ke operator. Muncul pesan “Jadwal berhasil diperbarui”	Pass
	TC-08 Batalkan jadwal tanpa	1. Klik tombol batalkan jadwal	Alasan:	Ditolak, muncul pesan "Alasan pembatalan wajib diisi."	Ditolak, muncul pesan "Alasan pembatalan wajib diisi."	Pass

	mengisi alasan	2. Kosongkan field alasan pembatalan 3. Klik yakin				
	TC-09 Batalkan jadwal dengan mengisi alasan	1. Klik tombol batalkan jadwal 2. Isi alasan pembatalan 3. Klik yakin	Alasan: Rapat dibatalkan karena pimpinan tidak bisa menghadiri rapat	Status data penjadwala berubah menjadi "dibatalkan", meeting Zoom (jika ada) ikut dihapus, notifikasi WA terkirim ke operator	Status data penjadwala berubah menjadi "dibatalkan", meeting Zoom (jika ada) ikut dihapus, notifikasi WA terkirim ke operator	Pass
TS-05 Pengajuan Peminjaman Alat (Operator)	TC-01 Ajukan peminjaman dengan data valid	1. Masuk halaman ajukan peminjaman 2. Pilih tanggal pinjam >= hari ini & tanggal kembali >= tanggal pinjam 3. Pilih alat & jumlah, isi keperluan 4. Klik ajukan	Jadwal: Rapat Koordinasi Kab Bandung - Sen, 13 Jul 2026 Keperluan: Rapat Koordinasi Kab Bandung Tanggal Pinjam: 07/13/2026 Rencana Kembali: 07/14/2026 Alat yang dibuthkan: Converter Type-C to HDMI Jumlah: 1	Data pengajuan peminjaman berhasil tersimpan dengan status "diajukan", notifikasi WA terkirim ke petugas inventaris. Muncul pesan “Pengajuan berhasil dikirim. Notifikasi telah dikirim ke petugas inventaris.”	Data pengajuan peminjaman berhasil tersimpan dengan status "diajukan", notifikasi WA terkirim ke petugas inventaris. Muncul pesan “Pengajuan berhasil dikirim. Notifikasi telah dikirim ke petugas inventaris.”	Pass
	TC-02 Ajukan peminjaman tanpa memilih peralatan	1. Isi tanggal & keperluan 2. Tidak memilih peralatan apapun 3. Klik ajukan	Jadwal: Rapat Koordinasi Kab Bandung - Sen, 13 Jul 2026 Keperluan: Rapat Koordinasi Kab Bandung Tanggal Pinjam: 07/13/2026 Rencana Kembali: 07/14/2026 Alat yang dibutuhkan: Jumlah:	Ditolak, muncul pesan "Pilih minimal 1 peralatan."	Ditolak, muncul pesan "Pilih minimal 1 peralatan."	Pass
	TC-03 Tanggal kembali rencana lebih awal dari tanggal pinjam	1. Isi tanggal_kembali_rencana sebelum tanggal_pinjam 2. Klik ajukan	Jadwal: Rapat Koordinasi Kab Bandung - Sen, 13 Jul 2026 Keperluan: Rapat Koordinasi Kab Bandung Tanggal Pinjam: 07/14/2026	Ditolak, muncul pesan "Tanggal kembali harus sama dengan atau setelah tanggal pinjam."	Ditolak, muncul pesan "Tanggal kembali harus sama dengan atau setelah tanggal pinjam."	Pass

			Rencana Kembali: 07/13/2026 Alat yang dibuthkan: Converter Type-C to HDMI Jumlah: 1			
	TC-04 Ajukan peralatan yang stoknya sudah habis	1. Buka form ajukan peminjaman 2. Cari peralatan dengan stok tersedia 0	Jadwal: Rapat Koordinasi Kab Bandung - Sen, 13 Jul 2026 Keperluan: Rapat Koordinasi Kab Bandung Tanggal Pinjam: 07/13/2026 Rencana Kembali: 07/14/2026 Alat yang dibuthkan: Converter Type-C to HDMI (stok 0 *disable/hidden*) Jumlah: 0	Data peralatan dengan stok habis (tidak tersedia) tidak muncul di daftar pilihan form	Data peralatan dengan stok habis (tidak tersedia) tidak muncul di daftar pilihan form	Pass
	TC-05 Indikasi pengajuan berulang (alat sama >= 2x, tanggal sama)	1. Ajukan alat yang sama untuk tanggal pinjam yang sama, padahal sudah diajukan 2x sebelumnya	Jadwal: Rapat Koordinasi Kab Bandung - Sen, 13 Jul 2026 Keperluan: Rapat Koordinasi Kab Bandung Tanggal Pinjam: 07/13/2026 Rencana Kembali: 07/14/2026 Alat yang dibuthkan: Gimbal Stabilizer DJI Ronin Jumlah: 1	Muncul modal peringatan indikasi pengajuan berulang, operator tetap bisa lanjut submit setelah konfirmasi	Muncul modal peringatan indikasi pengajuan berulang, operator tetap bisa lanjut submit setelah konfirmasi	Pass
	TC-06 Ubah pengajuan yang masih menunggu (tambah alat)	1. Login sbg operator 2. Masuk halaman edit pengajuan berstatus "diajukan" milik sendiri 3. Tambah/ubah alat & jumlah 4. Klik simpan	Jadwal: Rapat Koordinasi Kab Bandung - Sen, 13 Jul 2026 Keperluan: Rapat Koordinasi Kab Bandung Tanggal Pinjam: 07/13/2026 Rencana Kembali: 07/14/2026 Alat yang dibuthkan: -Gimbal Stabilizer DJI Ronin Jumlah: 1 -Kabel HDMI 15 Meter	Data pengajuan peminjaman berhasil diperbarui dengan alat/jumlah baru, notifikasi WA terkirim ulang ke petugas inventaris	Data pengajuan peminjaman berhasil diperbarui dengan alat/jumlah baru, notifikasi WA terkirim ulang ke petugas inventaris	Pass

			Jumlah: 1			
	TC-07 Batalkan pengajuan tanpa mengisi alasan	1. Klik batalkan pengajuan 2. Kosongkan field alasan pembatalan 3. Klik yakin	Alasan:	Ditolak, muncul pesan "Alasan pembatalan wajib diisi."	Ditolak, muncul pesan "Alasan pembatalan wajib diisi."	Pass
	TC-08 Mengakses /mengubah pengajuan milik operator lain	1. Ubah URL id_peminjaman menjadi milik operator lain 2. Akses halaman edit	http://127.0.0.1:8000/operator/peminjaman/PMJ-2026-0001/edit	Response 403 Forbidden, Pesan muncul “Anda tidak memiliki izin untuk mengakses halaman ini.”	Response 403 Forbidden, Pesan muncul “Anda tidak memiliki izin untuk mengakses halaman ini.”	Pass
TS-06 Laporan	TC-01 Melihat halaman laporan	1. Login sebagai admin 2. Masuk menu laporan	Menekan menu tombol laporan	Menampilkan rekap data laporan	Menampilkan rekap data laporan	Pass
	TC-02 Export laporan ke PDF	1. Buka halaman laporan 2. Klik "Export PDF"	Menekan aksi tombol export laporan PDF	File PDF berhasil terunduh, berisi data laporan sesuai filter	File PDF berhasil terunduh, berisi data laporan sesuai filter	Pass
	TC-03 Export laporan ke Excel	1. Buka halaman laporan 2. Klik "Export Excel"	Menekan aksi tombol export laporan Excel	File .xlsx berhasil terunduh, berisi data laporan sesuai filter	File .xlsx berhasil terunduh, berisi data laporan sesuai filter	Pass
TS-07 Dashboard & Kontrol Akses Role	TC-01 Redirect dashboard sesuai role	1. Login sebagai admin atau operator 2. Akses halaman /dashboard	User admin: admin@diskominfotik.go.id password User operator: operator1@diskominfotik.go.id password	Diarahkan ke halaman dashboard sesuai role masing-masing	Diarahkan ke halaman dashboard sesuai role masing-masing	Pass
	TC-02 Mengakses route milik role lain	1. Login sebagai operator 2. Akses langsung URL halaman admin	http://127.0.0.1:8000/admin/dashboard	Response 403, muncul pesan "Anda tidak memiliki akses ke halaman ini."	Response 403, muncul pesan "Anda tidak memiliki akses ke halaman ini."	Pass
	TC-03 Mengakses halaman tanpa login	1. Logout / hapus session 2. Akses langsung URL	http://127.0.0.1:8000/admin/dashboard	Diarahkan ke halaman login / response 403	Diarahkan ke halaman login / response 403	Pass

		halaman yang butuh login				

Berdasarkan pengujian yang telah dilakukan terhadap aplikasi ini menggunakan metode *Black-Box Testing*, hasil rekapitulasi pengujian disajikan pada tabel berikut

Tabel 5.34 Rekapitulasi Hasil Pengujian

No	Test Scenario	Jumlah *Test Case*	Presentase Lulus Ujian
1.	TS-01 Login	4	100%
2.	TS-02 Lupa Password OTP WhatsApp	6	100%
3.	TS-03 Kelola Pengguna (Admin)	5	100%
4.	TS-04 Kelola Jadwal Rapat (Admin)	9	100%
5.	TS-05 Pengajuan Peminjaman Alat (Operator)	8	100%
6.	TS-06 Laporan (Admin)	3	100%
7.	TS-07 Dashboard & Kontrol Akses Role	3	100%

Berdasarkan tabel rekapitulasi tersebut, terdapat tujuh *Test Scenario* dengan total 38 *Test Case* pada pengujian *Black-Box Testing* aplikasi ini. Seluruh *Test Case* dinyatakan lulus dengan presentase keberhasilan sebesar 100%, sehingga dapat disimpulkan bahwa fitur-fitur yang terdapat pada aplikasi telah berfungsi sesuai dengan yang diharapkan.

BAB 6
KESIMPULAN DAN SARAN

6.1 Kesimpulan

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

1. Aplikasi yang dibangun telah mampu menyajikan informasi jadwal rapat, perubahan jadwal, serta penugasan operator secara lebih jelas kepada pihak terkait. Informasi jadwal yang sebelumnya belum memuat penugasan operator dan kebutuhan peralatan kini dapat dikelola dalam satu aplikasi sehingga mendukung koordinasi pelaksanaan rapat.
2. Aplikasi yang dibangun telah mendukung proses peminjaman dan pengembalian peralatan yang terhubung dengan kegiatan rapat apabila diperlukan. Dengan demikian, penggunaan peralatan dapat terdokumentasi dengan lebih baik sehingga memudahkan proses pengelolaan dan penelusuran data peminjaman.
3. Aplikasi yang dibangun telah menyediakan penyampaian informasi jadwal rapat dan perubahan jadwal kepada operator melalui notifikasi WhatsApp secara otomatis. Selain itu, untuk rapat yang menggunakan platform Online atau Hybrid, aplikasi mendukung pembuatan tautan Zoom secara otomatis melalui API sehingga penyampaian informasi kepada operator menjadi lebih cepat dan proses penyusunan jadwal menjadi lebih efisien.

6.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, beberapa saran untuk penelitian selanjutnya adalah sebagai berikut.

1. Penelitian selanjutnya disarankan untuk mempertimbangkan penggunaan WhatsApp Business API resmi dari Meta sebagai pengganti WhatsApp Gateway pihak ketiga (Fonnte). Penggunaan API resmi diharapkan dapat meningkatkan keandalan layanan serta mengurangi risiko pemblokiran nomor WhatsApp yang digunakan. Penerapan layanan tersebut perlu disesuaikan dengan kesiapan anggaran dan proses verifikasi bisnis yang berlaku.
2. Penelitian selanjutnya disarankan untuk menambahkan media notifikasi selain WhatsApp, seperti *email*, sebagai alternatif penyampaian informasi apabila terjadi kendala pada layanan WhatsApp. Pengiriman surat elektronik dapat diimplementasikan menggunakan layanan SMTP, seperti Gmail SMTP, sehingga diharapkan dapat meningkatkan keandalan penyampaian informasi kepada pengguna.
3. Penelitian selanjutnya disarankan untuk melakukan *usability testing* guna mengevaluasi tingkat kemudahan penggunaan aplikasi dari perspektif pengguna. Pengujian tersebut

diharapkan dapat memberikan penilaian yang lebih komprehensif terhadap antarmuka aplikasi, sehingga dapat melengkapi hasil *black box testing* yang telah dilakukan.

DAFTAR PUSTAKA

[AGI25] Agil, M., Homaidi, A., Hamdani, I., "Perancangan Sistem Penjadwalan Otomasi Petugas Vidiotron di Dinas Komunikasi dan Informatika Situbondo", *Jurnal Ilmu Komputer dan Sistem Informasi*, Vol. 4, No. 3, 2025.

[ALK24] Al Khalim, S., Handayanto, A., "Perancangan Aplikasi Penjadwalan Agenda Kegiatan Berbasis Website di BPS Provinsi Jawa Tengah", 2024.

[AND25] Andaria, A. C., Ardhana, V. Y. P., Nugroho, A. Y., Prastyo, D., Bakhtiar, M. Y., Martawireja, A. R. H., Arifin, M., Triyadi, Zaenuddin, I., Saputra, M. H. K., Wijaksono, B. A., Suwandono, Rauf, B. W., Rahman, A., Fatich, E. V. L. N., Nurhadi, *PENGENALAN BAHASA PEMROGRAMAN UNTUK PEMULA*, PT. Hadla Media Informasi, Bekasi, 2025.

[FAC22] Fachrurrazi, S., Shidqi, M. G., Rahman, A., "Rancang Bangun dan Aplikasi Sistem Informasi Penjadwalan Petugas Piket dan Layanan Informasi Berbasis Media Web", Jurnal SISFO, Vol. 6, No. 1, 2022.

[HAS20] Hasanah, F. N., Untari, R. S., *Rekayasa Perangkat Lunak*, UMSIDA Press, Sidoarjo, 2020.

[KBBI] Pusat Bahasa Departemen Pendidikan Nasional, "Kamus Besar Bahasa Indonesia", tersedia: https://kbbi.kemdikbud.go.id.

[KHO25] Kholiq, M. F. R., Maulana, N., Alzidane, M. D. H., Haryono, W., "Pengembangan Sistem Informasi Terintegrasi untuk Inventori, Absensi, dan Penjadwalan Kerja", Jurnal Global, Vol. 3, No. 4, 2025.

[NUG25] Nugroho, A. Y., *Sistem dan Teknik Inventory*, Gemilang Press Indonesia, Padang, 2025.

[PAR22] Pardomuan, G. N., Limbong, A., *Pengantar Aplikasi Komputer*, Cipta Media Nusantara, Surabaya, 2022.

[RAM25] Ramadhan, M. R., Septiawan, M. A., "Analisis dan Perancangan Sistem Informasi Penjadwalan Rapat pada Kantor DPRD Provinsi Sumatera Utara dengan Metode Waterfall", Jurnal Merkurius, Vol. 3, No. 1, 2025.

[SAB24] Sabaruddin, R., Murni, S., Lisnawanty, Nugraha, W., *BELAJAR MUDAH LARAVEL 9 LEVEL PEMULA*, PT. Insan Cendekia Mandiri Group, Sijunjung, 2024.

[SRI21] Sriadhi, T. L., *Pemrograman Web Dasar*, Yayasan Kita Menulis, Medan, 2021.

[SUR19] Suryawinata, M., *Pengembangan Aplikasi Berbasis Web*, UMSIDA Press, Sidoarjo, 2019

[SWA24] Swastika, Rahayu, *Buku Ajar Pemodelan Sistem Berbasis Objek Beserta Studi Kasus*, Universitas Bina Sarana Informatika, Jakarta, 2024.

[TIN24] Tinambunan, M. H., Siregar, A. H., Ginting, S., *Fullstack Programming: Membangun Application Programming Interface (API) Dengan Laravel*, Tahta Media Group, Medan, 2024.

[TWCSS] Tailwind Labs Inc., "Tailwind CSS", tersedia: https://tailwindcss.com/.

LAMPIRAN

LAMPIRAN A

SURAT KETERANGAN PENELITIAN TUGAS AKHIR

UNIVERSITAS PASUNDAN
Fakultas Teknik

Teknik Industri 022 - 2019335
Teknologi Pangan 022 - 2019339
Teknik Mesin 022 - 2019352
Teknik Informatika 022 - 2019371
Teknik Lingkungan 022 - 2009574
Perencanaan Wilayah dan Kota 022 - 2006466

Bandung 9 Maret 2026

No : 239/Unpas.FT.IF/Q/ II /2026
Hal : Penelitian Tugas Akhir
Lamp : -

Kepada Yth.
Bapak/Ibu Kepala
Badan Kesatuan Bangsa dan Politik Kabupaten Bandung Barat
Jl. Raya Padalarang Cisarua Km. 2, Desa Mekarsari, Kecamatan Ngamprah,
Kabupaten Bandung Barat, Jawa Barat 40552.

Dengan hormat,
Bersama ini kami perkenalkan 1 (satu) orang mahasiswa Program Studi Teknik Informatika Fakultas Teknik Universitas Pasundan.

Nama : **Salma Az Zahra**
Nrp : **243040107**
Program Studi : **Teknik Informatika.**

Untuk menyelesaikan studinya, mahasiswa tersebut di atas bermaksud untuk penelitian Tugas Akhir, untuk mengumpulkan data di Instansi/Perusahaan yang Bapak / Ibu Pimpin.

Sehubungan dengan itu kami mohon dengan hormat agar mahasiswa tersebut di atas diberi izin mengumpulkkan data yang diperlukan untuk penelitian tersebut.

Atas perhatian, bantuan dan kerjasama yang baik, kami ucapkan terima kasih.

Program Studi Teknik Informatika
Universitas Pasundan.
Sekretaris,

(Erik., S.T., M. Kom)
NIDN : 0423088302

Tembusan Arsip
1. Arsip

Jl. Setiabudhi No. 193 Bandung - 40153 DEKANAT ☎ 022 - 2019435, (HUNTING) 2019433, 2019407 Fax. 022 - 2019329

LAMPIRAN B

FORM WAWANCARA

UNIVERSITAS PASUNDAN
Fakultas Teknik
Program Studi Teknik Informatika
Jl. Dr. Setiabudi No.193, Gegerkalong, Kec. Sukasari, Kota Bandung, Jawa Barat 40153

Nama	:	Tedy Yulianto, S.Sos., M.M.
NIP	:	19790717200912100
Tanggal	:	31 Maret 2026
Jabatan	:	Pranata Komputer Ahli Muda

Isi Wawancara :

1. Siapa yang bertanggung jawab dalam mengatur jadwal rapat?

   Saya sendiri

2. Apakah pernah terjadi kesalahan atau konflik jadwal dalam pelaksanaan rapat?

   Untuk masalah lebih ke pemohon kadang-kadang ada perubahan, misalnya hari Senin ternyata berubah menjadi hari Rabu atau jamnya yang berubah. Kalau di internal kita tidak ada karena sifatnya hanya memberikan bantuan, tergantung *user* saja permohonan mereka.

3. Apakah seorang operator dapat menangani lebih dari satu kegiatan rapat pada waktu yang sama, atau sebaiknya hanya menangani satu kegiatan rapat dalam satu waktu?

   Hanya bisa menangani satu kegiatan dalam satu waktu

4. Dalam satu kegiatan rapat, apakah biasanya melibatkan lebih dari satu operator yang bertugas menyiapkan peralatan rapat?

   Bisa iya bisa tidak tergantung sifat dari rapat tersebut. Karena untuk rapat kita punya empat perangkat. Satu kegiatan rapat bisa untuk satu operator bisa, kecuali untuk acara-acara yang besar kita butuh sampai tiga orang. Acara besar itu misalnya event-event yang ada Bupati atau Kapolres kita biasanya lebih dari satu orang bahkan sampai empat orang. Tapi untuk rapat-rapat kecil kita cukup satu orang.

5. Media apa yang paling efektif digunakan untuk memberikan informasi jadwal rapat kepada operator, misalnya melalui aplikasi WhatsApp atau media lainnya?

Bisa menggunakan WhatsApp untuk menegaskan lagi

6. Data apa saja yang perlu dikelola oleh admin dalam sistem penjadwalan rapat?

Biasanya jika kita ada rapat tertentu materi dari mereka, nanti paparan kita yang menyiapkan. Untuk formatnya nanti ada *rundown,* kita hanya memaparkan materi yang diberikan sesuai *rundown*

7. Siapa yang bertanggung jawab dalam mengelola akun pengguna pada sistem penjadwalan rapat di instansi?

Saya, admin

8. Apakah operator atau pengguna lain diperbolehkan membuat atau mengubah akun mereka sendiri dalam sistem?

Tidak

9. Apakah sistem penjadwalan rapat perlu terhubung langsung dengan aplikasi Zoom atau cukup digunakan untuk mengatur jadwal kegiatan rapat saja?

Tidak, kalau masalah penjadwalan karena salah satu contohnya begini, ada rapat yang dari pusat yang sudah ada *link* Zoom-nya otomatis dari mereka tapi kalau internal biasanya kita yang membuat karena kita punya akun Zoom tiga. Kita membuat sendiri akun sesuai dengan *request* dari mereka. Temanya apa? Jam berapa? Hari apa?

10. Apakah data peserta rapat perlu dicatat dalam sistem penjadwalan rapat?

Untuk aplikasi penjadwalan rapat tidak perlu. Tetapi di dalam rapat yang berlangsung tergantung permohonan dari mereka, OPD (Organisasi Perangkat Daerah) misalnya *request* tolong buatkan absen tapi kalau tidak maka tidak perlu.

11. Mengingat adanya perubahan nomenklatur kementerian di tingkat pusat menjadi Komunikasi dan Digital (Komdigi), apakah terdapat rencana atau kebijakan di tingkat daerah, khususnya pada Dinas Komunikasi, Informatika, dan Statistik Kabupaten Bandung Barat, untuk menyesuaikan nomenklatur instansi menjadi Komdigi?

Sejauh ini tidak ada imbauan untuk mengubah nama, bisa diperhatikan semua Kabupaten, Kota sampai Provinsi semua namanya masih tetap diskominfotik karena itu kebijakannya kembali lagi ke daerah masing-masing. Dengan mengubah nomenklatur itu tidak semudah itu harus merubah pergubnya, contoh kecil merubah webnya, banyak yang mesti dirubah.

Untuk sementara ini kita tidak melakukan perubahan karena tidak ada anjuran wajib untuk merubah nama menjadi Komdigi. Itu tidak ada.

12. Apakah terdapat batasan tertentu dalam penentuan operator untuk suatu kegiatan rapat?

Tergantung sifat rapatnya, jika event besar yang di outdoor itu bisa lebih dari empat orang, karena kita harus menyiapkan starlink, pointing dulu, displaynya, audio tapi untuk rapat disini paling satu sampai dua

13. Apa arti pintek pada aplikasi yang akan dibangun?

Singkatan dari Penjadwalan Infrastruktur dan Teknologi

14. Apakah penggunaan layanan WhatsApp Gateway melalui Fonnte diperbolehkan untuk implementasi sistem?

Diperlukan, karena memudahkan user

15. Saat ini penelitian menggunakan layanan Fonnte versi gratis yang memiliki batas pengiriman pesan. Apakah solusi tersebut sudah cukup untuk kebutuhan operasional yang ada?

Cukup karena dalam satu hari tidak mungkin lebih dari seribu rapat, karena untuk internal bukan untuk masyarakat

Mengetahui,

Peneliti	Narasumber
Salma Az Zahra	Tedy Yulianto, S.Sos., M.M.

UNIVERSITAS PASUNDAN
Fakultas Teknik
Program Studi Teknik Informatika
Jl. Dr. Setiabudi No.193, Gegerkalong, Kec. Sukasari, Kota Bandung, Jawa Barat 40153

Nama	:	Melky
NIP	:	199908152025211040
Tanggal	:	31 Maret 2026
Jabatan	:	Operator Layanan Operasional

Isi Wawancara :

1. Bagaimana proses penjadwalan rapat yang berjalan selama ini di Diskominfotik Bandung Barat, mulai dari pengajuan sampai pelaksanaan. Biasanya menggunakan apa dan siapa yang terlibat?

   Untuk SOP rapat tidak ada yang tertulis, tetapi kita biasanya dari:
   1) Permintaan dinas lain masuk ke Kepala Dinas atau Kepala bidang
   2) Menginformasikan ke Pak Tedy sebagai Sub Koordinator di Seksi Infrastruktur dan Jaringan
   3) Lalu menginfokan ke stafnya (operator)
   4) Lalu menyiapkan perangkat kebutuhannya apa
   5) Selanjutnya survei ke lokasi untuk mengecek alat karena perangkatnya terbatas
   6) Menyiapkan internet atau starlink kalau di desa yang jaringan internetnya terbatas
   7) Terakhir, langsung eksekusi ke lokasi

2. Bagaimana operator mengetahui jadwal rapat yang harus dipersiapkan?

   Kalau jadwal langsung, kadang bersifat mendadak sehingga tidak menentu dan masih disampaikan secara lisan

3. Informasi apa saja yang perlu diketahui operator ketika menerima jadwal rapat?

   Biasanya untuk ke lokasi sama ketersediaan alatnya sudah ada apa saja di sana, jadi kita mengisi kekurangannya

4. Apakah operator perlu mendapatkan notifikasi ketika terdapat jadwal rapat baru?

   Iya, lebih bagus jika ada aplikasinya

5. Media apa yang paling efektif untuk menerima notifikasi jadwal rapat?

WhatsApp, agar lebih cepat

6. Apa saja tugas operator dalam mempersiapkan kegiatan rapat?

Kita cek dulu lokasi dan kebutuhannya apa, kalau sudah tahu kebutuhannya kita menyiapkan dan memasangkan perangkatnya. Kebutuhannya misalnya membutuhkan berapa kamera sama butuh apa saja seperti mikrofon. Untuk rapat besar butuh *microphone wireless* yang menjangkau jarak jauh kalau jarak dekat pakai bawaan kamera sudah cukup

7. Apakah operator perlu mengetahui daftar peralatan yang harus disiapkan untuk setiap kegiatan rapat?

Iya, agar tidak lupa

8. Apakah operator perlu melaporkan setelah meminjam dan mengembalikan peralatan rapat?

Perlu juga, agar keberadaan perangkat dapat dipantau dan mengurangi risiko kehilangan

Mengetahui,

Peneliti	Narasumber
Salma Az Zahra	Melky

PEMERINTAH KABUPATEN BANDUNG BARAT
DINAS KOMUNIKASI, INFORMATIKA DAN STATISTIK

UNIVERSITAS PASUNDAN
Fakultas Teknik
Program Studi Teknik Informatika
Jl. Dr. Setiabudi No.193, Gegerkalong, Kec. Sukasari, Kota Bandung, Jawa Barat 40153

Nama	:	Danie Ramadhani
NIP	:	198107242024211003
Tanggal	:	31 Maret 2026
Jabatan	:	Pranata Komputer Terampil

Isi Wawancara :

1. Peralatan apa saja yang biasanya digunakan dalam kegiatan rapat?

   Peralatan yang biasanya digunakan kamera, laptop, speaker itu saja untuk rapat online. Untuk rapat offlinenya biasanya monitor, microphone dan laptop.

2. Bagaimana proses peminjaman peralatan rapat yang dilakukan saat ini?

   Untuk proses peminjaman biasanya dari instansi lain mereka bersurat melalui E-Office nanti dilakukan penjadwalan oleh admin nanti baru dilaksanakan rapat

3. Data apa saja yang perlu dicatat ketika peralatan dipinjam untuk kegiatan rapat?

   Biasanya nomor seri, alatnya apa dan dokumentasi

4. Siapa yang bertanggung jawab dalam proses peminjaman serta pencatatan penggunaan peralatan rapat?

   Untuk peminjaman biasanya ada adminnya, ada operatornya, dan ada inventaris

5. Apakah dalam satu kegiatan rapat dapat menggunakan lebih dari satu peralatan?

   Iya lebih dari satu peralatan

6. Apakah diperlukan laporan penggunaan peralatan setelah kegiatan rapat selesai?

   Diperlukan

7. Seberapa sering rapat dilaksanakan di dinas dalam jangka waktu satu minggu?

   Fasilitasi rapat biasanya dalam satu minggu itu ada satu sampai dua kali

8. Apakah di setiap gedung Kabupaten Bandung Barat terdapat petugas inventaris atau hanya Bapak yang mengelolanya?

Pengelolaan inventaris umum dilakukan oleh bendahara barang. Kami hanya mengelola barang-barang infrastruktur jaringan saja

9. Apakah selalu dilakukan survei lokasi sebelum pemasangan peralatan? Jika tidak, dalam kondisi seperti apa survei dilakukan?

Untuk rapat internal sebenarnya kita survei juga namun jika sudah terbayang titiknya seperti apa kadang tidak perlu, kalau untuk fasilitasi rapat di luar kita pasti setidaknya survei. Surveinya bisa lewat google maps, atau koordinasi dengan dinas yang mau difasilitasi, kondisinya seperti apa? Mau ditempatkan di mana? Operatornya ada di mana? Kita ada satu kebutuhan barang juga yang sering digunakan yaitu fasilitasi internetnya misalnya starlink atau modem telkom, ada jaringan atau tidaknya di sana. Jika di sana ada jaringan kita membawa kabel

10. Bagaimana proses koordinasi antara pihak penyelenggara rapat dan petugas inventaris terkait kebutuhan peralatan?

Harus ada. Karena kita di sana mengenai tempat, jalurnya ke mana lalu kita di sana apa yang harus ditampilkan misalnya zoom atau streaming youtube dari mereka, linknya mereka yang menyediakan atau kami? Misalnya kita menyediakan 1 atau 2 kamera? Atau apa yang harus ditampilkan?

11. Bagaimana proses peminjaman peralatan yang saat ini berjalan?

Untuk peminjaman internal mungkin ada surat dari pemohon kita biasanya pakai E-Office lalu diteruskan ke kita apakah kita yang survei ke lokasi maka apa saja yang harus dipersiapkan? misalnya mereka tidak memiliki PC maka kita sediakan laptop, jika mereka tidak ada jaringan internet kita carikan jaringannya bisa pakai modem atau jaringan satelit atau kabel. Setelah itu kita koordinasi dengan PIC dari sana apa yang harus ditampilkan? Pembukaannya seperti apa? Akan ada sesi tanya jawab atau tidak? Untuk *rundown*nya pasti kita selalu minta

12. Bagaimana proses pengembalian peralatan setelah selesai digunakan?

Setelah kegiatan berjalan lancar lalu barang dikembalikan, untuk internal karena dekat bisa dihari yang sama dikembalikan, jika rapat di luar bisa hari yang sama atau esok harinya atau jika hari libur bisa dikembalikan hari Senin

13. Apakah peralatan sering dipinjam untuk kegiatan selain rapat? Jika iya, kegiatan apa saja yang biasanya menggunakan peralatan tersebut?

Misalnya saat bencana alam kemarin di Cisarua kita ada permintaan atau tidak kita sudah memprediksi harus ke sana untuk memfasilitasi barangkali ada jaringan internet atau tidak. Karena saat longsor semua jaringan rusak semua jadi kita sudah koordinasi internal memang apa yang dibutuhkan di sana. Lalu beberapa hari lalu kita diminta untuk fasilitasi dinas lingkungan hidup di Batu Jajar mereka tidak rapat hanya *streaming* saja, siapkan jaringan internet juga. Ada juga di Pertanian ada panen raya seluruh Indonesia yang dibuka oleh presiden kita di sana mengurus jaringan internet juga bisa menggunakan modem, satelit atau minta bantuan pihak ketiga bagaimana situasi di lapangan

14. Bagaimana cara memeriksa kondisi peralatan sebelum dan sesudah digunakan?

Pertama apakah peralatan itu tersedia atau tidak? misalnya tersedia, bisa digunakan atau tidak? diperiksa dahulu. Misalnya bisa digunakan maka digunakan, yang tidak bisa digunakan setelah acara atau hari-H saat pengembalian memeriksa gudang, kondisinya baik atau tidak, kotor atau tidaknya maka kita lakukan pemeliharaan

15. Bagaimana penanganan jika ditemukan peralatan rusak atau sedang dalam proses perbaikan?

Kalau untuk yang rusak itu kita coba perbaiki internal kalau tidak kita cari jasa servis, jika barangnya masih bisa diperbaiki maka kita perbaiki jika tidak kita akan memberi alat baru

Mengetahui,

Peneliti	Narasumber
Salma Az Zahra	Danie Ramadhani

UNIVERSITAS PASUNDAN
Fakultas Teknik
Program Studi Teknik Informatika
Jl. Dr. Setiabudi No.193, Gegerkalong, Kec. Sukasari, Kota Bandung, Jawa Barat 40153

Nama	:	Adi Rahmat Prasetio
NIP	:	199303202025211075
Tanggal	:	26 Juni 2026
Jabatan	:	Penata Layanan Operasional

Isi Wawancara :

1. Apa saja tugas dan tanggung jawab operator dalam pelaksanaan rapat?

   Tugasnya sebagai fasilitator teknis, seperti penyediaan kamera laptop, jika rapatnya via zoom

2. Istilah "operator" di lingkungan Diskominfotik mengacu pada siapa? Apakah terdapat penugasan khusus sebagai operator?

   Tugas operator ditunjukkan bagi pegawai yang memiliki jabatan operator layanan operasional, dan juga penunjukan langsung

3. Saat ini terdapat berapa operator yang aktif dan bagaimana pembagian tugasnya?

   Ada 5 pegawai, pembagian tugas berdasarkan arahan langsung dari sub koordinator

4. Apakah terdapat perbedaan tugas antara operator dan petugas rapat? Jika ada, apa perbedaannya?

   Perbedaannya dari segi teknis, seperti penyediaan fasilitas laptop dan kabel HDMI. Sedangkan petugas rapat biasanya hanya mengikuti kegiatannya

5. Seberapa sering operator ditugaskan dalam kegiatan rapat setiap minggunya?

   Tidak tentu, tetapi biasanya minimal satu kali dalam satu minggu

6. Hari atau waktu apa yang biasanya paling sering digunakan untuk kegiatan rapat?

   Biasanya hari Kamis, tapi tidak menentu juga. Menyesuaikan surat undangan rapat

7. Bagaimana operator mengetahui adanya perubahan jadwal rapat atau penugasan operator?

Perubahan atau pembatalan jadwal rapat biasanya diinformasikan langsung oleh Sub Koordinator

8. Bagaimana proses penentuan peralatan yang akan digunakan dalam suatu kegiatan rapat?

Peralatan ditentukan berdasarkan surat undangan rapat, biasanya di surat tersebut dituliskan apa saja kebutuhannya

9. Berapa lama waktu yang biasanya dibutuhkan untuk menyiapkan peralatan sebelum rapat berlangsung?

Biasanya operator menyiapkan dalam waktu 10 sampai 15 menit

10. Bagaimana proses pengembalian peralatan setelah selesai digunakan?

Operator melaporkan kepada Sub Koordinator, lalu mengembalikan peralatan ke tempat semula

11. Bagaimana prosedur apabila terjadi bentrok jadwal operator?

Operator menunggu arahan dari Sub Koordinator, dan biasanya 5 operator dibagi tugas kembali oleh Sub Koordinator

Mengetahui,

Peneliti	Narasumber
Salma Az Zahra	Ari Rahmat Prasetio

PEMERINTAH KABUPATEN BANDUNG BARAT
DINAS KOMUNIKASI, INFORMATIKA DAN STATISTIK

LAMPIRAN C
DOKUMENTASI WAWANCARA

Gambar 6.1 Dokumentasi Wawancara dengan Sub Koordinator

Gambar 6.2 Dokumentasi Wawancara dengan Operator1

Gambar 6.3 Dokumentasi Wawancara dengan Inventaris

Gambar 6.4 Dokumentasi Wawancara dengan Operator 2