

**PEMBANGUNAN WEBSITE ABSENSI SISWA DENGAN INTEGRASI
WHATSAPP *GATEWAY*
(STUDI KASUS: SDN CIBITUNGKULON 02)**

TUGAS AKHIR

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

Oleh :

Muhammad Azhar Utama

nrp. 22.304.0077



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PASUNDAN BANDUNG
SEPTEMBER 2026**

LEMBAR PENGESAHAN

LAPORAN TUGAS AKHIR

Telah diujikan dan dipertahankan dalam Sidang Sarjana Program Studi Teknik Informatika Universitas Pasundan Bandung, pada hari dan tanggal sidang sesuai berita acara sidang, Tugas Akhir dari:

Nama : Muhammad Azhar Utama

NPM : 223040077

Dengan Judul :

**" PEMBANGUNAN WEBSITE ABSENSI SISWA DENGAN INTEGRASI WHATSAPP
GATEWAY**

(STUDI KASUS: SDN CIBITUNGKULON 02) "

Bandung, September 2026

Menyetujui,

Pembimbing Utama



(Dr. Ir. Leony Lidya, MT.)

LEMBAR PERNYATAAN KEASLIAN TUGAS AKHIR

Saya menyatakan dengan sesungguhnya bahwa :

1. Tugas Akhir ini 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 dan telah dituliskan dalam 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, 23 September 2026

Y
aan,


Muhammad Azhar Utama

NRP. 22.304.0077

ABSTRAK

Kehadiran siswa merupakan salah satu indikator penting dalam mendukung kedisiplinan dan proses pembelajaran di sekolah dasar. Namun, proses absensi di SDN Cibitungkulon 02 masih menggunakan media konvensional berupa buku absensi dan rekap dilakukan secara manual menggunakan Microsoft Excel. Selain membutuhkan waktu yang cukup lama, proses tersebut berpotensi menimbulkan kesalahan pencatatan dan kesulitan dalam pengelolaan data absensi. Pengiriman informasi ketidakhadiran siswa kepada orang tua siswa juga masih dilakukan secara manual sehingga penyampaian informasi menjadi kurang efisien dan tidak real-time.

Penelitian ini bertujuan untuk membangun website absensi siswa yang mampu mendigitalisasi proses pencatatan kehadiran serta menyediakan fitur pesan otomatis kepada orang tua siswa melalui WhatsApp *Gateway*. Metode penelitian yang digunakan adalah rekayasa perangkat lunak dengan pendekatan pengembangan sistem Waterfall yang meliputi tahapan analisis kebutuhan, perancangan, implementasi, dan pengujian sistem. Sistem dikembangkan menggunakan framework Laravel dan database MySQL.

Hasil penelitian berupa website absensi yang memiliki fitur pencatatan kehadiran digital, rekap absensi otomatis, dashboard monitoring, serta pengiriman pesan WhatsApp kepada orang tua siswa setiap kali siswa tercatat dengan status Alpa, Izin, atau Sakit. Sistem yang dibangun diharapkan dapat meningkatkan efisiensi pengelolaan data absensi, mengurangi kesalahan pencatatan, serta mempermudah penyampaian informasi ketidakhadiran siswa kepada orang tua siswa secara otomatis dan real-time.

Kata kunci: absensi siswa, WhatsApp *Gateway*, pesan otomatis, Laravel, MySQL, sistem berbasis web, Waterfall, monitoring kehadiran, sekolah dasar

ABSTRACT

Student attendance is one of the important indicators in supporting discipline and the learning process in elementary schools. However, the attendance process at SDN Cibitungkulon 02 still relies on conventional methods using attendance books, with recapitulation done manually through Microsoft Excel. In addition to being time-consuming, this process is prone to recording errors and difficulties in managing attendance data. Notification of student absences to parents is also still carried out manually, making information delivery less efficient and non real time.

This study aims to develop a website-based student attendance system capable of digitalizing the attendance recording process and providing an automatic notification feature to parents via WhatsApp *Gateway*. The research method used is software engineering with a Waterfall system development approach, encompassing the stages of requirements analysis, design, implementation, and testing. The system was developed using the Laravel framework and a MySQL database.

The result of this research is a website-based attendance system featuring digital attendance recording, automatic attendance recapitulation, a monitoring dashboard, and WhatsApp notification delivery to parents each time a student is recorded as absent without notice (*alpa*), on leave (*izin*), or sick (*sakit*). The system is expected to improve the efficiency of attendance data management, reduce recording errors, and facilitate the automatic and real-time delivery of student attendance information to parents.

Keywords: student attendance, WhatsApp *Gateway*, automatic notification, Laravel, MySQL, web-based system, Waterfall, attendance monitoring, elementary school

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah Subhanahu Wa Ta'ala atas rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan laporan Tugas Akhir yang berjudul “Pembangunan Website Absensi Siswa Dengan Integrasi WhatsApp *Gateway* (Kasus: SDN Cibitungkulon 02)” sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Informatika, Fakultas Teknik, Universitas Pasundan Bandung.

Penulis menyadari laporan ini masih jauh dari sempurna, sehingga kritik dan saran yang membangun sangat penulis harapkan. Pada kesempatan ini, penulis mengucapkan terima kasih kepada:

1. Orang tua dan keluarga yang selalu memberikan motivasi serta do'a dalam penyusunan tugas akhir ini.
2. Ibu Dr. Ir. Leony Lidya, MT. selaku dosen pembimbing yang telah membimbing penulis.
3. Bapak Wanda Gusdya, ST., MT. selaku Koordinator Tugas Akhir.
4. Seluruh civitas akademik Teknik Informatika, Universitas Pasundan Bandung.
5. Teman-teman seperjuangan yang tidak bisa disebutkan satu per satu.
6. Pihak SDN Cibitungkulon 02 yang telah memberikan izin dan dukungan selama penelitian.

Semoga laporan ini bermanfaat bagi pengembangan sistem absensi berbasis website di lingkungan pendidikan serta menjadi referensi bagi penelitian selanjutnya.

Bandung, September 2026

Penulis

DAFTAR ISI

ABSTRAK.....	iv
ABSTRACT.....	v
KATA PENGANTAR	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	x
DAFTAR TABEL.....	xii
DAFTAR LAMPIRAN.....	xv
DAFTAR LAMBANG/SIMBOL	xvi
DAFTAR ISTILAH	xviii
BAB 1_PENDAHULUAN	1-1
1.1 Latar Belakang.....	1-1
1.2 Identifikasi Masalah	1-3
1.3 Tujuan.....	1-3
1.4 Lingkup Tugas Akhir.....	1-3
1.5 Metodologi Tugas Akhir	1-4
1.6 Standar dan Kakas	1-8
1.7 Sistematika Penulisan	1-8
BAB 2_LANDASAN TEORI.....	2-1
2.1 Teori Yang Digunakan	2-1
2.1.1 Pembangunan Sistem	2-1
2.1.2 Siswa	2-1
2.1.3 Absensi.....	2-1
2.1.4 Website	2-2
2.1.5 Metode Waterfall.....	2-2
2.1.6 PHP Hypertext Preprocessor (PHP).....	2-4
2.1.7 Framework	2-4
2.1.8 Laravel.....	2-5
2.1.9 Model-View-Controller (MVC)	2-5
2.1.10 Tailwind CSS	2-6
2.1.11 HyperText Markup Language (HTML)	2-6
2.1.12 Cascading Style Sheets (CSS).....	2-6

2.1.13 MySQL.....	2-6
2.1.14 Application Programming Interface (<i>API</i>).....	2-7
2.1.15 WhatsApp <i>Gateway</i>	2-7
2.1.16 Unified Modeling Language (UML).....	2-7
2.1.17 <i>Black box testing</i>	2-9
2.1.18 <i>Acceptance testing</i>	2-10
2.1.19 Wawancara	2-11
2.1.20 Skala Guttman	2-12
2.2 Penelitian Terdahulu.....	2-13
BAB 3 SKEMA PENELITIAN	3-1
3.1 Alur Penyelesaian Tugas	3-1
3.2 Perumusan Masalah	3-3
3.2.1 Analisis Masalah	3-3
3.2.2 Solusi Masalah.....	3-4
3.2.3 Skema Analisis	3-6
3.3 Profil Tempat Penelitian.....	3-7
3.3.1 Struktur Organisasi.....	3-7
3.3.2 Visi dan Misi	3-8
3.4 Teknik Pengumpulan Data	3-8
BAB 4 ANALISIS DAN PERANCANGAN	4-1
4.1 Analisis	4-1
4.1.1 Teknik Analisis	4-1
4.1.2 Analisis Sistem yang Sedang Berjalan.....	4-1
4.1.3 Analisis Proses Bisnis	4-2
4.1.4 Analisis Kebutuhan Sistem	4-5
4.1.4.1 <i>User story</i>	4-6
4.1.4.2 Analisis Aktor	4-6
4.1.4.3 Diagram <i>Use Case</i> Bisnis.....	4-7
4.1.4.4 Analisis Kebutuhan Pengguna.....	4-7
4.1.4.5 Analisis Kebutuhan Fungsional.....	4-7
4.1.4.6 Analisis Kebutuhan Non Fungsional.....	4-9
4.1.4.7 Identifikasi <i>Use Case</i>	4-10
4.2 Perancangan.....	4-11
4.2.1 <i>Use Case</i> Diagram.....	4-11
4.2.2 Skenario <i>Use Case</i>	4-13
4.2.3 Activity Diagram.....	4-29

4.2.4 Sequence Diagram.....	4-31
4.2.5 Entity Relationship Diagram (ERD)	4-37
4.2.6 Gambaran Umum Infrastruktur yang Dibangun.....	4-38
BAB 5 KONSTRUKSI DAN PENGUJIAN	5-1
5.1 Konstruksi Perangkat Lunak.....	5-1
5.1.1 Kakas dan Kebutuhan Perangkat Lunak	5-1
5.1.1.1 Spesifikasi Perangkat Keras	5-1
5.1.1.2 Perangkat Lunak yang Digunakan	5-1
5.1.1.3 Teknologi Pengembangan Sistem	5-2
5.1.2 Pengelolaan Konstruksi.....	5-3
5.2 Pengkodean.....	5-3
5.2.1 Struktur Folder	5-4
5.2.2 Model	5-6
5.2.3 View	5-7
5.2.4 Controller dan Service.....	5-8
5.2.5 Basis Data.....	5-9
5.2.6 Konfigurasi Routing Sistem	5-14
5.3 Antarmuka dan Kode.....	5-15
5.4 Proses <i>Deployment</i> Aplikasi.....	5-53
5.4.1 Konfigurasi Dockerfile.....	5-53
5.4.2 Konfigurasi Layanan Railway.....	5-54
5.4.3 Deployment Aplikasi	5-55
5.5 Pengujian Perangkat Lunak	5-55
5.5.1 Black Box Testing.....	5-55
5.5.2 Rekapitulasi Hasil Black Box Testing.....	5-65
5.5.3 Pengujian Penerimaan Pengguna (User Acceptance Testing).....	5-66
BAB 6 KESIMPULAN DAN SARAN	6-1
6.1 Kesimpulan.....	6-1
6.2 Saran.....	6-1
DAFTAR PUSTAKA	xxii
LAMPIRAN.....	xxiv

DAFTAR GAMBAR

Gambar 1.1 Langkah-langkah Penyelesaian Tugas Akhir.....	1-5
Gambar 2.1 Tahapan Metode Waterfall.....	2-3
Gambar 3.1 <i>Fishbone</i> Masalah	3-4
Gambar 3.2 <i>Fishbone</i> Solusi.....	3-5
Gambar 3.3 Skema Analisis.....	3-6
Gambar 3.4 Struktur Organisasi	3-8
Gambar 4.1 Activity Diagram Persiapan Master	4-3
Gambar 4.2 Activity Diagram Proses Absensi Harian & Rekap Laporan.....	4-4
Gambar 4.3 Diagram <i>Use Case</i> Bisnis	4-7
Gambar 4.4 <i>Use Case</i> Diagram Sistem Absensi SDN Cibitungkulon 02.....	4-12
Gambar 4.5 Activity Diagram Kelola guru, kelas, siswa dan periode	4-30
Gambar 4.6 Activity Diagram proses Absensi rekap, dan Kirim <i>Pesan</i> ke Orang tua siswa.....	4-31
Gambar 4.7 Sequence Diagram Kelola Data Guru	4-32
Gambar 4.8 Sequence Diagram Kelola Data Siswa.....	4-33
Gambar 4.9 Sequence Diagram Kelola Data Kelas	4-34
Gambar 4.10 Sequence Diagram Kelola Periode Tahun Ajaran.....	4-35
Gambar 4.11 Sequence Diagram Lihat Rekap Absensi	4-36
Gambar 4.12 Sequence Diagram Cetak Laporan Rekap.....	4-36
Gambar 4.13 Sequence Diagram Lihat Riwayat Pengiriman Pesan	4-37
Gambar 4.14 Sequence Diagram Kelola Absensi	4-37
Gambar 4.15 Entity Relationship Diagram (ERD)	4-38
Gambar 4.16 Gambaran Umum Infrastruktur.....	4-39
Gambar 5.1 Struktur folder proyek	5-4
Gambar 5.2 Tampilan halaman Login	5-16
Gambar 5.3 Tampilan halaman Dashboard.....	5-17
Gambar 5.4 Tampilan halaman Daftar Guru	5-18
Gambar 5.5 Tampilan halaman Tambah Guru.....	5-20
Gambar 5.6 Tampilan halaman Ubah Guru.....	5-22
Gambar 5.7 Tampilan halaman Daftar Siswa	5-25
Gambar 5.8 Tampilan halaman Tambah Siswa	5-27
Gambar 5.9 Tampilan halaman Ubah Siswa.....	5-28
Gambar 5.10 Tampilan halaman Daftar Kelas.....	5-30

Gambar 5.11 Tampilan halaman Tambah Kelas.....	5-31
Gambar 5.12 Tampilan halaman Edit Kelas	5-33
Gambar 5.13 Tampilan halaman Periode.....	5-35
Gambar 5.14 Tampilan halaman Rekap Absensi.....	5-37
Gambar 5.15 Tampilan halaman Hasil Ekspor Laporan Excel.....	5-41
Gambar 5.16 Tampilan halaman Riwayat Pesan	5-43
Gambar 5.17 Tampilan halaman Isi Absensi	5-45
Gambar 5.18 Tampilan halaman Koreksi Absensi	5-48
Gambar 5.19 Tampilan pesan WhatsApp	5-51

DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu	2-13
Tabel 3.1 Alur Penyelesaian Tugas Akhir	3-1
Tabel 3.2 Analisis Masalah.....	3-4
Tabel 3.3 Solusi Masalah.....	3-5
Tabel 3.4 Deskripsi Skema Analisis	3-7
Tabel 4.1 Analisis Persiapan Data Master	4-3
Tabel 4.2 Analisis Swimlane Proses Absensi Harian & Rekap Laporan.....	4-5
Tabel 4.3 <i>User story</i>	4-6
Tabel 4.4 Analisis Aktor	4-6
Tabel 4.5 Analisis Kebutuhan Pengguna	4-7
Tabel 4.6 Kebutuhan Fungsional Sistem	4-8
Tabel 4.7 Kebutuhan Non-Fungsional Sistem	4-9
Tabel 4.8 Identifikasi <i>Use Case</i>	4-10
Tabel 4.9 Definisi <i>Use Case</i>	4-12
Tabel 4.10 Skenario UC-01: Kelola Data Guru	4-13
Tabel 4.11 Skenario UC-02: Kelola Data Siswa.....	4-15
Tabel 4.12 Skenario UC-03: Kelola Data Kelas	4-18
Tabel 4.13 Skenario UC-04: Kelola Data Periode.....	4-20
Tabel 4.14 Skenario UC-05: Melihat Data Rekap	4-22
Tabel 4.15 Skenario UC-06: Cetak Laporan Rekap	4-23
Tabel 4.16 Skenario UC-07: Melihat Riwayat Pengiriman <i>Pesan</i>	4-24
Tabel 4.17 Skenario UC-08: Kelola Absensi.....	4-26
Tabel 5.1 Spesifikasi Perangkat Keras.....	5-1
Tabel 5.2 Perangkat Lunak yang Digunakan.....	5-1
Tabel 5.3 Teknologi Pengembangan Sistem.....	5-2
Tabel 5.4 Pembagian Urutan Implementasi.....	5-3
Tabel 5.5 Struktur folder.....	5-5
Tabel 5.6 Struktur Kelas Model.....	5-6
Tabel 5.7 Struktur Folder View	5-7
Tabel 5.8 Struktur Kelas Controller, Service, Job, dan Export.....	5-8
Tabel 5.9 Daftar Tabel Basis Data.....	5-9
Tabel 5.10 Struktur Tabel users	5-10
Tabel 5.11 Struktur Tabel periodes.....	5-11
Tabel 5.12 Struktur Tabel kelas	5-11

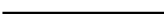
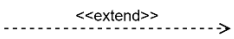
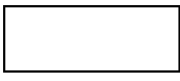
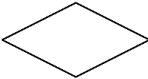
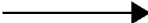
Tabel 5.13 Struktur Tabel siswas.....	5-11
Tabel 5.14 Struktur Tabel hari_liburs	5-12
Tabel 5.15 Struktur Tabel absensis	5-12
Tabel 5.16 Struktur Tabel whatsapp_notifications	5-13
Tabel 5.17 Struktur Tabel activity_logs.....	5-13
Tabel 5.18 Daftar Route Sistem.....	5-14
Tabel 5.19 Antarmuka dan Kode Halaman Login	5-16
Tabel 5.20 Antarmuka dan Kode Halaman Dashboard	5-17
Tabel 5.21 Antarmuka dan Kode Halaman Daftar Guru	5-18
Tabel 5.22 Antarmuka dan Kode Halaman Tambah Guru	5-20
Tabel 5.23 Antarmuka dan Kode Halaman Ubah Guru.....	5-22
Tabel 5.24 Antarmuka dan Kode Halaman Daftar Siswa	5-25
Tabel 5.25 Antarmuka dan Kode Halaman Tambah Siswa	5-27
Tabel 5.26 Antarmuka dan Kode Halaman Ubah Siswa	5-28
Tabel 5.27 Antarmuka dan Kode Halaman Daftar Kelas	5-30
Tabel 5.28 Antarmuka dan Kode Halaman Tambah Kelas.....	5-31
Tabel 5.29 Antarmuka dan Kode Halaman Edit Kelas	5-33
Tabel 5.30 Antarmuka dan Kode Halaman Periode	5-34
Tabel 5.32 Antarmuka dan Kode Halaman Hasil Ekspor Laporan Excel.....	5-41
Tabel 5.33 Antarmuka dan Kode Halaman Riwayat Pesan	5-43
Tabel 5.34 Antarmuka dan Kode Halaman Isi Absensi.....	5-45
Tabel 5.35 Antarmuka dan Kode Halaman Koreksi Absensi	5-48
Tabel 5.36 Antarmuka dan Kode Pesan WhatsApp Kepada Orang tua.....	5-51
Tabel 5.37 Black Box Testing Login Sistem	5-56
Tabel 5.38 Black Box Testing Kelola Data Guru	5-57
Tabel 5.39 Black Box Testing Kelola Data Siswa.....	5-59
Tabel 5.40 Black Box Testing Kelola Data Kelas	5-60
Tabel 5.41 Black Box Testing Kelola Data Periode	5-61
Tabel 5.42 Black Box Testing Lihat Data Rekap	5-62
Tabel 5.43 Black Box Testing Cetak Laporan Rekap.....	5-63
Tabel 5.44 Black Box Testing Melihat Riwayat Pengiriman Pesan	5-63
Tabel 5.45 Black Box Testing Kelola Absensi	5-64
Tabel 5.46 Rekapitulasi Hasil Black Box Testing	5-65
Tabel 5.47 UAT Login, Hak Akses & Lupa Password.....	5-68
Tabel 5.48 UAT Kelola Data Guru.....	5-68

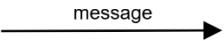
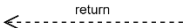
Tabel 5.49 UAT Kelola Data Siswa.....	5-68
Tabel 5.50 UAT Kelola Data Kelas	5-69
Tabel 5.51 UAT Kelola Periode Tahun Ajaran	5-69
Tabel 5.52 UAT Lihat Rekap Absensi.....	5-70
Tabel 5.53 UAT Cetak Laporan Rekap	5-70
Tabel 5.54 UAT Melihat Riwayat Pengiriman Pesan.....	5-70
Tabel 5.55 UAT Absensi, Koreksi & Pesan WhatsApp Otomatis.....	5-70
Tabel 5.56 Hasil Pengujian Penerimaan Pengguna untuk Operator (UAT)	5-71
Tabel 5.57 Hasil Pengujian Penerimaan Pengguna untuk Guru (UAT)	5-72
Tabel 5.58 Hasil Pengujian Penerimaan Pengguna untuk Kepala Sekolah (UAT)	5-73
Tabel 5.59 Rekapitulasi Hasil Pengujian Penerimaan Pengguna (UAT).....	5-73

DAFTAR LAMPIRAN

Lampiran 1 Buku absensi	xxiv
Lampiran 2 Wawancara dengan kepala sekolah operator dan guru	xxiv
Lampiran 3 Hasil wawancara guru.....	xxvi
Lampiran 4 Hasil wawancara operator.....	xxvii
Lampiran 5 Hasil wawancara Kepala sekolah.....	xxviii
Lampiran 6 pelaksanaan UAT.....	xxix

DAFTAR LAMBANG/SIMBOL

No	Simbol	Nama Simbol	Deskripsi
Notasi Diagram Use Case			
1.		Use Case	Merepresentasikan sebuah fungsionalitas atau layanan yang disediakan oleh sistem
2.		Actor	Merepresentasikan aktor yang akan berinteraksi dengan sistem, baik pengguna maupun sistem lain
3.		Association	Merepresentasikan interaksi atau hubungan antara Use Case dengan actor
4.		extend	Menunjukkan bahwa suatu Use Case secara opsional menyertakan Use Case lain
5		System Boundary	Kotak yang membatasi ruang lingkup sistem dari aktor eksternal
Notasi Diagram Activity			
1.		Initial Node	Menandai titik awal (start) dari alur aktivitas
2.		Activity Final Node	Menandai titik akhir (end) dari seluruh alur aktivitas
3.		Activity	Merepresentasikan langkah atau tindakan yang dilakukan dalam suatu proses
4.		Decision Node	Merepresentasikan percabangan kondisi (ya/tidak) dalam alur proses
5.		Control Flow	Menggambarkan aliran kontrol dari satu aktivitas ke aktivitas berikutnya

No	Simbol	Nama Simbol	Deskripsi
6.		Swimlane	Partisi yang mengelompokkan aktivitas berdasarkan aktor atau pihak yang bertanggung jawab
Notasi Sequence Diagram			
1		Aktor	Merepresentasikan pengguna atau entitas eksternal yang berinteraksi dengan sistem
2		Boundary	Merepresentasikan antarmuka atau halaman yang menjadi perantara interaksi antara aktor dengan sistem.
3		Control	Merepresentasikan objek yang mengatur alur proses dan logika antara boundary dengan objek lain dalam sistem.
4		Entity	Merepresentasikan objek atau data yang dikelola dan disimpan dalam sistem, seperti data guru, kelas, dan log aktivitas
5		Activation Bar	Menunjukkan periode waktu suatu objek aktif menjalankan operasi
6		Message	Panah yang menggambarkan pengiriman pesan atau pemanggilan method antar objek
7		Return Message	Panah putus-putus yang menggambarkan nilai balik dari suatu pemanggilan method

DAFTAR ISTILAH

Istilah	Keterangan
<i>API</i>	Antarmuka pemrograman yang memungkinkan dua aplikasi saling berkomunikasi dan bertukar data
<i>Backend</i>	Bagian sistem yang mengurus logika, basis data, dan <i>Server</i> agar aplikasi berjalan dengan baik
<i>Black-box Testing</i>	Metode pengujian perangkat lunak yang berfokus pada fungsionalitas sistem tanpa memeriksa struktur internal kode program
<i>Browser</i>	Perangkat lunak yang digunakan untuk mengakses dan menampilkan halaman <i>Website</i>
<i>Cache</i>	Penyimpanan sementara yang digunakan untuk mempercepat proses pengambilan data yang sering diakses
<i>Dashboard</i>	Halaman utama sistem yang menampilkan ringkasan informasi dan menu navigasi
<i>Database</i>	Basis data berupa kumpulan data terstruktur yang disimpan dan dikelola secara elektronik
<i>Deployment</i>	Proses penerapan atau peluncuran sistem/aplikasi agar dapat digunakan pada lingkungan produksi
<i>Domain</i>	Nama alamat unik yang digunakan untuk mengakses suatu <i>Website</i> di internet
<i>Username</i>	Surat elektronik yang digunakan untuk mengirim dan menerima pesan secara digital
<i>Error</i>	Kesalahan yang terjadi pada sistem akibat proses yang tidak berjalan sebagaimana mestinya

Istilah	Keterangan
<i>File</i>	Berkas data yang disimpan dan dikelola di dalam sistem komputer
<i>Fishbone</i>	Metode diagram berbentuk tulang ikan yang digunakan untuk mengidentifikasi akar penyebab masalah
<i>Folder</i>	Wadah yang digunakan untuk mengelompokkan dan menyimpan beberapa <i>File</i> dalam sistem komputer
<i>Foreign Key</i>	Kolom pada suatu tabel yang mengacu pada <i>Primary Key</i> tabel lain guna menjaga keterhubungan antar data
<i>Form</i>	Halaman atau elemen antarmuka yang digunakan pengguna untuk memasukkan data ke dalam sistem
<i>Framework</i>	Kerangka kerja perangkat lunak yang menyediakan struktur dasar untuk mempercepat pengembangan aplikasi
<i>Gateway</i>	Gerbang atau perantara yang menghubungkan dua sistem atau jaringan yang berbeda
<i>Input</i>	Data yang dimasukkan ke dalam sistem untuk diproses
<i>Interface</i>	Antarmuka yang menjadi penghubung interaksi antara pengguna dengan sistem
<i>Library</i>	Kumpulan kode program siap pakai yang digunakan untuk mempermudah proses pengembangan aplikasi
<i>Login</i>	Proses masuk ke dalam sistem dengan memasukkan kredensial berupa nama pengguna dan kata sandi
<i>Migration</i>	Berkas yang digunakan untuk mendefinisikan dan mengelola struktur tabel pada basis data secara terprogram
<i>Notification</i>	Pemberitahuan yang dikirimkan sistem kepada pengguna terkait suatu peristiwa atau informasi tertentu

Istilah	Keterangan
<i>Output</i>	Hasil akhir yang dihasilkan sistem setelah memproses data masukan
<i>Parameter</i>	Nilai atau variabel yang dikirimkan sebagai masukan dalam suatu proses atau fungsi
<i>Password</i>	Kata sandi rahasia yang digunakan pengguna untuk melakukan autentikasi ke dalam sistem
<i>Pesan</i>	Pemberitahuan otomatis yang dikirimkan sistem kepada pengguna terkait suatu informasi tertentu
<i>Primary Key</i>	Kolom pada suatu tabel yang berfungsi sebagai pengidentifikasi unik untuk setiap baris data
<i>Real-time</i>	Proses atau respons yang terjadi secara langsung dan terkini tanpa penundaan
<i>Request</i>	Permintaan yang dikirimkan pengguna atau sistem ke <i>Server</i> untuk mendapatkan respons tertentu
<i>Requirement</i>	Kebutuhan atau syarat yang harus dipenuhi dalam pengembangan suatu sistem
<i>Seeder</i>	Berkas yang digunakan untuk mengisi data awal ke dalam basis data secara otomatis
<i>Server</i>	Komputer atau perangkat lunak yang menyediakan layanan dan sumber daya bagi komputer lain dalam jaringan
<i>Session</i>	Rentang waktu interaksi pengguna dengan sistem sejak <i>Login</i> hingga keluar dari sistem
<i>Template</i>	Kerangka atau format baku yang digunakan sebagai acuan dalam pembuatan suatu tampilan atau dokumen
<i>Testing</i>	Proses pengujian yang dilakukan untuk memastikan sistem berjalan sesuai dengan yang diharapkan

Istilah	Keterangan
<i>Token</i>	Kode unik yang digunakan sebagai penanda otorisasi dalam suatu proses autentikasi
<i>Transaction</i>	Serangkaian operasi pada basis data yang diperlakukan sebagai satu kesatuan proses
<i>User</i>	Pengguna atau pihak yang menggunakan dan berinteraksi dengan sistem
<i>Username</i>	Nama pengguna yang digunakan sebagai identitas unik untuk mengakses sistem
<i>Website</i>	Sekumpulan halaman yang dapat diakses melalui jaringan internet menggunakan browser
<i>White-box Testing</i>	Metode pengujian perangkat lunak yang dilakukan dengan memeriksa struktur internal dan alur logika kode program

BAB 1

PENDAHULUAN

Bab ini memberikan pandangan umum mengenai pelaksanaan tugas akhir. Adapun penjelasan tersebut meliputi latar belakang, identifikasi masalah, tujuan tugas akhir, lingkup tugas akhir, metodologi tugas akhir, serta sistematika penulisan tugas akhir.

1.1 Latar Belakang

Perkembangan teknologi informasi telah mendorong berbagai sektor untuk bertransformasi menuju sistem berbasis digital, termasuk sektor pendidikan. Salah satu aspek operatoristatif yang memegang peranan penting dalam penyelenggaraan pendidikan adalah pencatatan kehadiran siswa. Di lingkungan sekolah dasar khususnya, sistem absensi memiliki peranan penting dalam membentuk dan mengembangkan perilaku disiplin serta giat belajar sejak usia dini sebagaimana dikemukakan oleh Nur Aizah dan Firman Santoso (Aizah & Santoso, 2024). Data kehadiran pun menjadi salah satu indikator evaluasi akademik dan pelaporan operatoristrasi yang tidak dapat diabaikan.

Berdasarkan hasil wawancara dan observasi awal yang dilakukan di SDN Cibitungkulon 02, proses pencatatan kehadiran siswa saat ini masih sepenuhnya bergantung pada media konvensional. Guru mencatat kehadiran secara manual di buku absensi, kemudian merekapnya ke dalam file Microsoft Excel dan menyerahkannya kepada operator sekolah setiap akhir bulan. Berdasarkan hasil wawancara dengan Operator Sekolah, proses pengumpulan dan penyatuan data absensi dari seluruh kelas memerlukan waktu sekitar 2–3 hari karena tidak semua guru menyerahkan laporan tepat waktu, dan format laporan yang diserahkan pun tidak seragam antar guru sebagian berupa file Excel yang dicetak, sebagian lain dikirim melalui WhatsApp sehingga operator harus menyusun ulang data secara manual sebelum dapat direkap. Kondisi ini bahkan pernah menyebabkan hilangnya data, seperti buku absensi yang rusak dan sulit dibaca sehingga sebagian catatan kehadiran tidak dapat dipulihkan. Temuan ini sejalan dengan pernyataan Moh. Baha'uddin, Adi Susanto, dan A. Hamdani bahwa pencatatan presensi yang masih dilakukan secara manual menggunakan dokumen fisik dan file Excel terbukti rawan terhadap kesalahan, keterlambatan, dan keterbatasan akses informasi secara real-time (Baha'uddin et al., 2025), serta pengamatan Paiman Nababan dkk. bahwa absensi yang dicatat secara manual dalam bentuk lembar kertas sering mengalami kehilangan, sehingga data absensi berpotensi menjadi tidak akurat, efektif, dan efisien (Nababan et al., 2022).

Permasalahan verifikasi juga ditemukan dari hasil wawancara dengan guru itu sendiri, yang mengakui bahwa kesalahan pencatatan memang pernah terjadi dalam proses absensi manual, dan beberapa kali menimbulkan komplain langsung dari orang tua siswa karena status kehadiran anaknya tercatat tidak sesuai kondisi sebenarnya. Guru tersebut juga menyatakan bahwa proses absensi saat ini

belum efisien, terutama pada tahap rekap dan pelaporan ke operator, karena koreksi baru bisa dilakukan setelah data sempat diserahkan dan harus dikirim ulang. Di sisi lain, Kepala Sekolah turut mengonfirmasi bahwa keterlambatan penerimaan rekap kehadiran dari operator maupun guru cukup sering terjadi, sebagai akibat langsung dari proses pengumpulan data yang masih dilakukan secara manual dan memakan banyak waktu, dan selama ini hanya menerima laporan rekapitulasi keseluruhan untuk keperluan evaluasi dan pelaporan ke dinas pendidikan tanpa akses langsung untuk memantau data secara real-time. Kondisi tersebut sejalan dengan temuan Rahi dkk. (2024) bahwa tingginya tingkat ketidakhadiran siswa yang sulit dipantau sekolah secara efektif dapat berdampak negatif pada proses pembelajaran, terutama karena orang tua tidak memiliki akses real-time terhadap informasi kehadiran anaknya sehingga tidak dapat segera mengambil tindakan (Rahi et al., 2024). Hal ini diperkuat oleh catatan Paiman Nababan dkk. bahwa orang tua siswa pada umumnya baru memahami laporan kehadiran anaknya pada saat pengambilan rapor (Nababan et al., 2022), sehingga kesalahan pencatatan yang terjadi berulang tidak dapat segera diketahui dan dikoreksi.

Permasalahan lain yang teridentifikasi dari wawancara adalah belum adanya mekanisme notifikasi otomatis kepada orang tua siswa. Guru menyampaikan bahwa pemberitahuan ketidakhadiran kepada orang tua saat ini baru dilakukan melalui WhatsApp secara manual, itu pun umumnya hanya ketika siswa tercatat alpa selama tiga hari berturut-turut, sehingga kasus ketidakhadiran atau kesalahan pencatatan yang terjadi sesekali cenderung luput dari perhatian. Akibatnya, ketika terjadi kesalahan pencatatan status kehadiran, orang tua baru mengetahuinya belakangan — bahkan pernah terjadi kasus orang tua siswa mengajukan keluhan langsung karena status kehadiran anaknya tercatat tidak sesuai dengan kondisi sebenarnya. Kejadian tersebut menegaskan perlunya mekanisme notifikasi otomatis yang dapat memberikan transparansi informasi kehadiran secara langsung kepada orang tua untuk setiap status kehadiran, sehingga kesalahan input serupa dapat lebih cepat diketahui dan ditindaklanjuti, sebagaimana ditunjukkan oleh Zahid dan Prapanca bahwa metode absensi konvensional masih kurang dapat diandalkan dalam menjamin kejujuran siswa terkait kehadiran mereka di sekolah, sehingga membuka peluang terjadinya manipulasi data kehadiran (Zahid & Prapanca, 2023).

Penelitian-penelitian sebelumnya telah menunjukkan bahwa penerapan sistem absensi berbasis website mampu memberikan solusi yang efektif dan efisien dalam pengelolaan absensi, meminimalkan kesalahan manual, dan meningkatkan akurasi data. Nur Aizah dan Firman Santoso menyimpulkan bahwa sistem absensi siswa berbasis web dengan WhatsApp *Gateway* dapat menjadi pendekatan yang memberikan kontribusi positif dalam meningkatkan efisiensi dan efektivitas proses operatorisasi sekolah (Aizah & Santoso, 2024). Ferlando Mbay Rahi, Erwianta Gustial Radjah, dan Alfrian Carmen Talakua juga melaporkan hasil serupa pada penerapan WhatsApp *Gateway* di SMA Negeri 1 Pandawai, yakni pengelolaan absensi menjadi lebih efisien dan akurat, sekaligus memberi orang tua kemudahan untuk mengetahui kehadiran anak mereka kapan saja secara real-time. (Rahi et al., 2024). Lebih jauh, Mas Muhammadun Zahid dan Aditya Prapanca membuktikan bahwa tingkat kehadiran siswa

menggunakan website absensi berbasis RFID dan WhatsApp mencapai 98%, meningkat signifikan dibandingkan metode konvensional yang hanya mencapai 88% (Zahid & Prapanca, 2023).

Berdasarkan permasalahan yang ditemukan di lapangan tersebut, penulis tertarik untuk melakukan penelitian dengan judul "Pembangunan Website Absensi Siswa dengan Integrasi WhatsApp *Gateway* (Studi Kasus: SDN Cibitungkulon 02)" yang bertujuan untuk menggantikan proses pencatatan di media konvensional, mengotomatisasi rekap kehadiran, serta mengintegrasikan pengiriman pesan WhatsApp secara otomatis kepada orang tua siswa setiap kali siswa tercatat dengan status Alpa, Izin, atau Sakit, sehingga potensi kesalahan input yang dilakukan guru dapat segera diketahui dan diverifikasi.

1.2 Identifikasi Masalah

Berdasarkan latar belakang yang telah diuraikan, maka permasalahan utama yang diidentifikasi adalah sebagai berikut:

1. Bagaimana membangun website absensi siswa yang dapat mengurangi penggunaan media konvensional berupa buku absensi di SDN Cibitungkulon 02?
2. Bagaimana mengotomatisasi proses rekap absensi siswa sehingga dapat mengurangi potensi kesalahan perhitungan yang terjadi pada proses rekap manual?
3. Bagaimana mengintegrasikan pengiriman secara otomatis kepada orang tua siswa melalui WhatsApp *Gateway* setiap kali siswa tercatat dengan status Alpa, Izin, atau Sakit, sehingga potensi kesalahan input yang dilakukan guru dapat segera diketahui oleh orang tua siswa tanpa memerlukan tindakan manual tambahan dari guru?

1.3 Tujuan

Berdasarkan identifikasi masalah tersebut, maka dapat diperoleh tujuan tugas akhir yaitu:

1. Membangun website absensi siswa yang mampu mendigitalisasi proses pencatatan kehadiran.
2. Merancang dan mengimplementasikan mekanisme rekap absensi siswa secara otomatis guna mengurangi proses rekap manual yang memerlukan waktu lama dan berpotensi menimbulkan kesalahan perhitungan.
3. Mengintegrasikan WhatsApp *Gateway* sebagai mekanisme pengiriman pesan otomatis kepada orang tua siswa setiap kali siswa tercatat dengan status Alpa, Izin, atau Sakit, sebagai bentuk verifikasi terhadap keakuratan data yang diinput oleh guru.

1.4 Lingkup Tugas Akhir

Lingkup tugas akhir ini bertujuan untuk membatasi pembahasan agar penelitian yang dilakukan lebih terarah dan tidak menyimpang dari tujuan yang telah ditetapkan. Adapun batasan dalam pembangunan website absensi siswa ini adalah sebagai berikut:

1. Sistem dibuat berupa website, digunakan oleh 3 peran pengguna yaitu Operator, Kepala Sekolah, dan Guru.
2. Sistem mencakup Kelola data master, absensi siswa, rekap kehadiran, dan pengiriman pesan WhatsApp (Fonnte) ke orang tua siswa dengan batas kuota gratis 1.000 pesan per bulan.
3. Sistem tidak terhubung dengan data nilai maupun buku induk siswa.
4. Sistem hanya untuk absensi siswa, tidak untuk absensi guru atau pegawai.
5. Pesan WhatsApp hanya dikirim dari sistem ke orang tua, orang tua tetap diharuskan mengirim pesan kepada wali kelas jika status kehadiran tidak sesuai.
6. Pengujian sistem dibatasi pada Black Box Testing dan User Acceptance Testing (UAT).
7. Penelitian dilakukan khusus di SDN Cibitungkulon 02, tidak untuk sekolah lain.

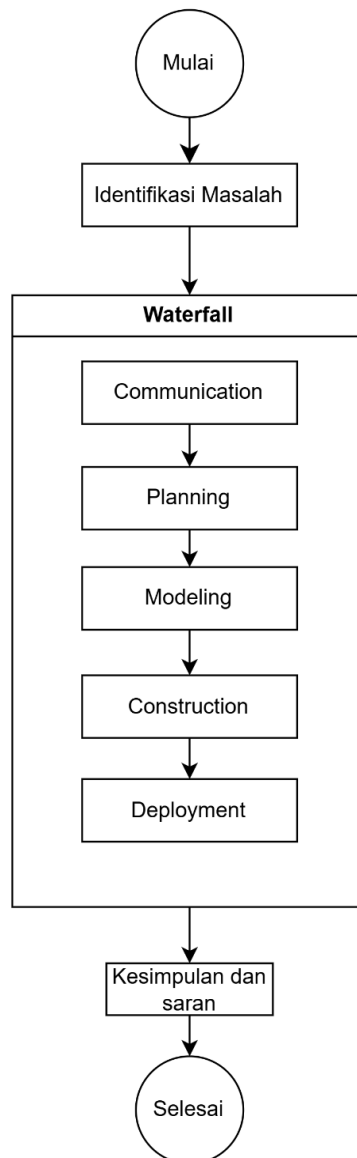
1.5 Metodologi Tugas Akhir

Metodologi tugas akhir merupakan tahapan yang digunakan sebagai pedoman dalam pelaksanaan penelitian dan pembangunan website absensi siswa dengan WhatsApp *Gateway* pada SDN Cibitungkulon 02. Penelitian ini menggunakan metode rekayasa perangkat lunak yang bertujuan untuk menghasilkan suatu produk berupa website absensi siswa

Dalam proses pembangunan sistem, digunakan metode Waterfall sebagai model pengembangan perangkat lunak. Metode Waterfall merupakan model pengembangan perangkat lunak yang dilakukan secara sistematis dan berurutan. Tahapan pengembangan perangkat lunak terdiri dari Communication, Planning, Modeling, Construction dan Deployment (Pressman, 2020).

Sebelum proses pengembangan menggunakan metode Waterfall dilakukan, penelitian diawali dengan identifikasi masalah untuk mengetahui permasalahan yang terdapat pada proses absensi siswa di SDN Cibitungkulon 02. Setelah seluruh proses pengembangan selesai, dilakukan kesimpulan dan saran sebagai tahap akhir penelitian.

Adapun tahapan metodologi tugas akhir yang dilakukan dapat dilihat pada Gambar 1.1.



Gambar 1.1 Langkah-langkah Penyelesaian Tugas Akhir

Gambar 1.1 menunjukkan alur metodologi penelitian yang digunakan dalam pembangunan website absensi siswa dengan integrasi WhatsApp *Gateway*. Diagram ini menggambarkan tahapan penelitian mulai dari identifikasi masalah, dilanjutkan dengan tahapan Waterfall yang terdiri atas Communication, Planning, Modeling, Construction, dan Deployment, hingga penarikan kesimpulan dan saran.

Berikut merupakan rincian dari metodologi penelitian yang digunakan, diantaranya:

1. Identifikasi Masalah

Tahap identifikasi masalah dilakukan untuk mengetahui permasalahan yang terjadi pada proses absensi siswa di SDN Cibitungkulon 02. Identifikasi dilakukan dengan memahami proses absensi yang

sedang berjalan, mulai dari pencatatan kehadiran siswa hingga proses penyampaian informasi ketidakhadiran siswa kepada orang tua siswa.

Hasil dari tahap ini digunakan sebagai dasar untuk menentukan permasalahan yang akan diselesaikan melalui pembangunan website absensi siswa dengan *WhatsApp Gateway*.

2. Communication

Tahap communication merupakan tahap awal dalam metode Waterfall yang dilakukan untuk memperoleh informasi mengenai permasalahan, tujuan, serta kebutuhan sistem yang akan dikembangkan (Pressman, 2020).

Pada tahap ini dilakukan project initiation dan requirements gathering. Project initiation dilakukan untuk menentukan tujuan dan ruang lingkup sistem yang akan dibangun. Selanjutnya, requirements gathering dilakukan untuk memperoleh kebutuhan sistem melalui observasi, wawancara, dan studi literatur.

Observasi dilakukan dengan mengamati proses absensi siswa yang sedang berjalan di SDN Cibitungkulon 02. Wawancara dilakukan dengan pihak yang berkaitan dengan proses absensi untuk memperoleh informasi mengenai kebutuhan dan permasalahan yang dihadapi. Studi literatur dilakukan dengan mempelajari sumber-sumber yang berkaitan dengan absensi siswa, website, *WhatsApp Gateway*, dan teknologi yang digunakan dalam pembangunan sistem.

Hasil dari tahap communication berupa informasi mengenai permasalahan, kebutuhan pengguna, kebutuhan sistem, serta ruang lingkup sistem yang akan dikembangkan.

3. Planning

Tahap planning dilakukan berdasarkan kebutuhan yang telah diperoleh pada tahap communication. Tahap ini mencakup kegiatan estimating, scheduling, dan tracking (Pressman, 2020).

Kegiatan estimating dilakukan untuk menentukan perkiraan sumber daya yang diperlukan dalam pembangunan sistem, baik perangkat keras, perangkat lunak, maupun sumber daya lainnya. Scheduling dilakukan untuk menentukan jadwal pelaksanaan setiap kegiatan dalam pembangunan sistem. Selanjutnya, tracking dilakukan untuk memantau perkembangan pembangunan sistem agar sesuai dengan rencana yang telah ditentukan.

Hasil dari tahap planning berupa rencana pembangunan sistem yang digunakan sebagai pedoman dalam proses pengembangan.

4. Modeling

Tahap modeling berfokus pada kegiatan analysis dan design untuk menerjemahkan kebutuhan sistem menjadi rancangan yang dapat digunakan dalam proses pembangunan perangkat lunak (Pressman, 2020).

Pada tahap analysis dilakukan analisis terhadap kebutuhan sistem berdasarkan hasil communication. Analisis mencakup kebutuhan pengguna, kebutuhan fungsional, kebutuhan non-fungsional, serta analisis proses sistem yang sedang berjalan.

Selanjutnya, pada tahap design dilakukan perancangan sistem berdasarkan hasil analisis. Perancangan meliputi Use Case Diagram, Activity Diagram, Sequence Diagram, serta Entity Relationship Diagram (ERD).

Hasil dari tahap modeling berupa spesifikasi kebutuhan dan rancangan sistem yang menjadi dasar dalam proses construction.

5. Construction

Tahap construction merupakan tahap pembangunan perangkat lunak berdasarkan hasil analisis dan perancangan yang telah dilakukan pada tahap modeling. Tahap ini terdiri dari kegiatan code dan test (Pressman, 2020).

Pada kegiatan code, rancangan sistem diimplementasikan menjadi perangkat lunak menggunakan teknologi yang telah ditentukan, yaitu Laravel, PHP, MySQL, dan WhatsApp *Gateway*. Implementasi dilakukan berdasarkan kebutuhan dan rancangan sistem yang telah dibuat.

Setelah proses coding selesai, dilakukan kegiatan test untuk memastikan setiap fungsi sistem dapat berjalan sesuai dengan kebutuhan yang telah ditentukan. Pengujian dilakukan menggunakan black-box testing dengan menguji fungsi sistem berdasarkan masukan dan keluaran yang dihasilkan dan User acceptance testing.

Hasil dari tahap construction berupa website absensi siswa yang telah dibangun dan diuji.

6. Deployment

Tahap deployment merupakan tahap penerapan perangkat lunak yang telah selesai dibangun dan diuji ke lingkungan produksi agar dapat digunakan oleh pengguna (Pressman, 2020).

Pada tahap ini, aplikasi dikemas ke dalam container menggunakan Dockerfile dan diterapkan pada platform Railway sebagai Platform as a Service (PaaS). Railway dipilih karena mendukung deployment otomatis melalui integrasi dengan repositori GitHub serta menyediakan basis data MySQL terkelola, sehingga proses build dan deploy dapat dilakukan tanpa konfigurasi server secara manual.

Hasil dari tahap deployment berupa website absensi siswa yang telah diterapkan dan dapat diakses oleh pengguna.

7. Kesimpulan dan Saran

Tahap kesimpulan dan saran merupakan tahap akhir dalam pelaksanaan tugas akhir. Pada tahap ini dilakukan penarikan kesimpulan berdasarkan hasil penelitian, pembangunan, dan pengujian sistem yang telah dilakukan.

Kesimpulan digunakan untuk mengetahui ketercapaian tujuan yang telah ditetapkan dalam penelitian. Selanjutnya, saran diberikan sebagai bahan pertimbangan untuk pengembangan dan penyempurnaan sistem pada penelitian selanjutnya.

1.6 Standar dan Kakas

Dalam pelaksanaan tugas akhir ini, lingkungan pengembangan yang digunakan untuk membangun sistem adalah sebagai berikut:

- Laptop pribadi, sebagai alat utama untuk mengembangkan website absensi berbasis website.
- Framework Laravel sebagai fondasi utama pembangunan sistem, didukung PHP sebagai bahasa pemrograman, Blade sebagai template engine, dan Laravel Breeze sebagai starter kit autentikasi.
- Tailwind CSS, HTML, dan CSS untuk membangun antarmuka pengguna yang responsif dan modern.
- WhatsApp *Gateway* (fonnte) yang digunakan untuk pengiriman Pesan kepada orang tua siswa.
- MySQL sebagai sistem manajemen basis data untuk menyimpan seluruh data absensi, siswa, kelas, dan Pesan.
- Visual Studio Code sebagai code editor dalam proses pengembangan sistem.

1.7 Sistematika Penulisan

Berikut di bawah ini merupakan sistematika penulisan yang akan digunakan dalam penyusunan tugas akhir ini:

BAB 1 PENDAHULUAN

Bab ini memberikan pandangan umum mengenai pelaksanaan tugas akhir. Adapun penjelasan tersebut meliputi latar belakang, identifikasi masalah, tujuan tugas akhir, lingkup tugas akhir, metodologi tugas akhir, dan sistematika penulisan tugas akhir.

BAB 2 LANDASAN TEORI

Bab ini menjelaskan tentang kerangka pikir penulisan tugas akhir meliputi definisi-definisi, teori-teori serta konsep-konsep yang diperlukan sebagai sumber dalam penelitian. Pada bab ini dikemukakan hasil-hasil penelitian yang tertulis di buku-buku teks ataupun makalah-makalah di jurnal terkait.

BAB 3 SKEMA PENELITIAN

Bab ini menjelaskan alur penyelesaian tugas akhir, perumusan masalah, kerangka pemikiran teoritis, dan profil tempat penelitian sebagai panduan awal bagi pembaca untuk memahami tujuan, ruang lingkup, dan landasan penelitian yang dilakukan.

BAB 4 ANALISIS & PERANCANGAN

Pada Bab ini berisi mengenai konsep aplikasi dan desain aplikasi berdasarkan kebutuhan yang telah dipaparkan. Di dalamnya berisi deskripsi mengenai analisis kebutuhan sistem, dan model-model diagram perancangan menggunakan UML.

BAB 5 KONSTRUKSI DAN PENGUJIAN

Bab ini membahas konstruksi dan pengujian absensi siswa berbasis website dengan Pesan WhatsApp *Gateway* menggunakan framework Laravel, serta deployment

BAB 6 KESIMPULAN & SARAN

Bab ini berisi mengenai hasil penelitian serta pernyataan yang didapat berdasarkan identifikasi masalah yang dikemukakan, serta keterkaitan dari semua tahap yang dilakukan dalam penelitian. Di dalamnya terdapat pula saran yang diusulkan untuk penelitian selanjutnya.

BAB 2

LANDASAN TEORI

Bab ini menjelaskan tentang kerangka pikir penulisan tugas akhir meliputi teori yang digunakan terkait konsep-konsep dasar, teknologi, dan metodologi yang menjadi landasan penelitian, serta penelitian terdahulu yang relevan.

2.1 Teori Yang Digunakan

2.1.1 Pembangunan Sistem

Pembangunan sistem (system building) merupakan proses menciptakan sistem yang lengkap dan dapat dieksekusi melalui kompilasi dan penautan (linking) komponen-komponen sistem, pustaka eksternal, berkas konfigurasi, serta informasi pendukung lainnya (Sommerville, 2015). Proses ini berkaitan erat dengan tahapan pengadaan (procurement) dan pengembangan (development) dalam rekayasa sistem, di mana desain konseptual dikembangkan lebih lanjut untuk menghasilkan informasi yang digunakan dalam pengambilan keputusan kontrak pengembangan, sementara tahap pengembangan sendiri mencakup definisi kebutuhan, perancangan sistem, rekayasa perangkat keras dan perangkat lunak, integrasi sistem, hingga pengujian (Sommerville, 2015).

2.1.2 Siswa

Siswa atau peserta didik merupakan anggota masyarakat yang berusaha mengembangkan potensi diri melalui proses pembelajaran yang tersedia pada jalur, jenjang, dan jenis pendidikan tertentu (Republik Indonesia, 2003). Dalam konteks pendidikan formal di sekolah dasar, siswa merupakan subjek utama dalam proses pembelajaran yang mengikuti kegiatan pendidikan sesuai dengan ketentuan yang berlaku.

Hak dan kewajiban peserta didik dalam penyelenggaraan pendidikan diatur dalam Undang-Undang Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional. Pasal 12 ayat (1) mengatur hak peserta didik untuk memperoleh pelayanan pendidikan sesuai dengan bakat, minat, dan kemampuannya. Selain itu, peserta didik memiliki kewajiban untuk menjaga norma-norma pendidikan serta mengikuti proses pembelajaran sesuai dengan ketentuan yang berlaku (Republik Indonesia, 2003).

Dalam pelaksanaan pendidikan di sekolah, kehadiran siswa menjadi salah satu informasi yang perlu dicatat dan dikelola untuk mendukung proses operatoristrasi serta pemantauan kegiatan pembelajaran. Oleh karena itu, data siswa dalam penelitian ini mencakup identitas siswa dan informasi kelas yang diperlukan dalam proses pencatatan dan pengelolaan absensi.

2.1.3 Absensi

Absensi dapat diartikan sebagai kondisi kehadiran atau ketidakhadiran seseorang dalam suatu kegiatan yang memerlukan pencatatan status kehadiran. Dalam lingkungan pendidikan, absensi digunakan untuk mencatat kehadiran siswa selama mengikuti kegiatan pembelajaran sehingga informasi kehadiran dapat dikelola dan digunakan sebagai bagian dari operatoristrasi sekolah (Al-Husain, 2017).

Pencatatan absensi siswa dapat dilakukan dengan mencatat status kehadiran pada setiap hari pembelajaran. Dalam penelitian ini, status kehadiran siswa terdiri atas Hadir, Izin, Sakit, dan Alpa. Data absensi yang tersimpan kemudian digunakan untuk menghasilkan rekapitulasi kehadiran berdasarkan siswa, kelas, dan periode tertentu.

Untuk mengetahui tingkat kehadiran siswa dalam suatu periode, sistem menghitung persentase kehadiran berdasarkan jumlah hari siswa hadir dibandingkan dengan total hari efektif. Rumus yang digunakan adalah sebagai berikut:

$$\text{Persentase Kehadiran} = (\text{Jumlah Kehadiran} / \text{Total Hari Efektif}) \times 100\%$$

Selain persentase kehadiran, tingkat ketidakhadiran siswa dapat dihitung dengan rumus:

$$\text{Persentase Ketidakhadiran} = (\text{Jumlah Ketidakhadiran} / \text{Total Hari Efektif}) \times 100\%$$

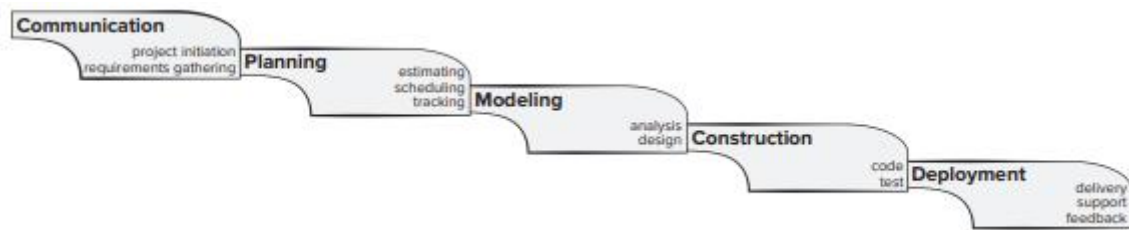
Kedua perhitungan tersebut digunakan dalam sistem untuk membantu menghasilkan rekapitulasi absensi secara otomatis sehingga pengguna dapat memperoleh informasi kehadiran dan ketidakhadiran siswa tanpa melakukan perhitungan secara manual.

2.1.4 Website

Website merupakan kumpulan halaman yang berada dalam suatu domain di internet yang saling terhubung dan dapat diakses oleh pengguna melalui browser. Website dibuat untuk menyajikan informasi dalam berbagai bentuk serta dapat diakses secara luas oleh pengguna melalui halaman utama (homepage) (Lembaga Penjaminan Mutu Pendidikan [LPMP], 2020)

Selain itu, website juga dapat dimanfaatkan sebagai media pembelajaran alternatif yang memungkinkan guru menyampaikan materi secara daring, mempercepat penyampaian informasi, serta mendorong kreativitas dan kemandirian siswa dalam proses belajar (LPMP, 2020).

2.1.5 Metode Waterfall



Gambar 2.1 Tahapan Metode Waterfall

Metode Waterfall atau linear sequential model merupakan salah satu model pengembangan perangkat lunak yang menggunakan pendekatan sistematis dan berurutan. Model ini dimulai dengan melakukan spesifikasi kebutuhan perangkat lunak, kemudian proses pengembangan dilanjutkan melalui tahapan planning, modeling, construction, dan deployment. Setelah perangkat lunak diterapkan, proses dilanjutkan dengan kegiatan dukungan terhadap perangkat lunak yang telah selesai dikembangkan (Pressman, 2020).

Tahapan pada metode Waterfall terdiri atas beberapa aktivitas utama, yaitu:

1 Communication

Tahap communication merupakan tahap awal yang dilakukan untuk memahami kebutuhan perangkat lunak. Pada tahap ini dilakukan komunikasi dengan pengguna atau pihak yang berkepentingan untuk memperoleh informasi mengenai permasalahan, tujuan, serta kebutuhan sistem yang akan dikembangkan. Hasil dari tahap ini menjadi dasar untuk menentukan spesifikasi kebutuhan perangkat lunak (Pressman, 2020).

2. Planning

Tahap planning dilakukan berdasarkan kebutuhan yang telah diperoleh pada tahap sebelumnya. Aktivitas pada tahap ini meliputi perencanaan pekerjaan, estimasi sumber daya, penjadwalan pengembangan, serta kegiatan pelacakan terhadap proses pengembangan. Perencanaan tersebut digunakan sebagai pedoman agar proses pengembangan dapat dilakukan secara terarah (Pressman, 2020)

3 Modeling

Tahap modeling berfokus pada kegiatan analisis dan perancangan sistem. Kebutuhan perangkat lunak yang telah ditentukan kemudian diterjemahkan ke dalam rancangan yang menggambarkan bagaimana sistem akan dibangun. Aktivitas pada tahap ini mencakup analisis kebutuhan dan perancangan sistem sebagai dasar bagi proses konstruksi perangkat lunak (Pressman, 2020).

4 Construction

Tahap construction merupakan tahap pembangunan perangkat lunak berdasarkan hasil analisis dan rancangan yang telah dibuat. Kegiatan utama pada tahap ini meliputi proses pengkodean (coding) dan pengujian (testing) untuk memastikan perangkat lunak yang dibangun dapat berfungsi sesuai dengan kebutuhan yang telah ditentukan (Pressman, 2020).

5 Deployment

Tahap deployment merupakan proses penerapan perangkat lunak kepada pengguna. Setelah perangkat lunak selesai dibangun dan diuji, sistem diberikan kepada pengguna untuk digunakan pada lingkungan sebenarnya. Tahap ini juga mencakup kegiatan dukungan, umpan balik, dan pemeliharaan terhadap perangkat lunak yang telah diterapkan (Pressman, 2020).

Meskipun memiliki alur yang sistematis, metode Waterfall memiliki beberapa keterbatasan. Proyek perangkat lunak pada kondisi nyata tidak selalu mengikuti alur kerja yang benar-benar berurutan seperti yang digambarkan oleh model Waterfall. Selain itu, pengguna sering mengalami kesulitan dalam menyampaikan seluruh kebutuhan sistem secara lengkap sejak awal pengembangan. Pengguna juga harus menunggu hingga proses pengembangan mencapai tahap tertentu sebelum dapat memperoleh versi perangkat lunak yang dapat digunakan. Permasalahan lainnya adalah kesalahan atau kekeliruan yang terjadi pada tahap awal dapat baru diketahui ketika perangkat lunak telah selesai dibangun dan ditinjau oleh pengguna (Pressman, 2020).

Dengan demikian, metode Waterfall lebih sesuai digunakan pada pengembangan perangkat lunak yang kebutuhan dan ruang lingkupnya telah dapat didefinisikan dengan jelas sejak awal serta relatif stabil selama proses pengembangan. Sebaliknya, model ini menjadi kurang sesuai apabila sistem yang dikembangkan memiliki kebutuhan yang sering berubah atau membutuhkan umpan balik pengguna secara terus-menerus selama proses pengembangan (Pressman, 2020).

2.1.6 PHP Hypertext Preprocessor (PHP)

PHP: Hypertext Preprocessor merupakan bahasa pemrograman Server-side yang digunakan dalam pengembangan aplikasi web. PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1994 dan hingga saat ini banyak digunakan dalam pembuatan sistem berbasis web (Solichin, 2008).

2.1.7 Framework

Framework merupakan struktur umum yang digunakan sebagai kerangka kerja dalam pengembangan perangkat lunak yang dapat diperluas untuk membangun aplikasi atau subsistem yang lebih spesifik. Framework terdiri dari sekumpulan komponen perangkat lunak seperti kelas, objek, dan

modul yang saling berinteraksi untuk menyediakan arsitektur yang dapat digunakan kembali (reusable) dalam pengembangan aplikasi (Sommerville, 2015).

Framework juga menyediakan fitur-fitur umum yang sering digunakan dalam pengembangan aplikasi, seperti pengelolaan antarmuka pengguna, pengolahan data, serta integrasi dengan database. Dengan adanya framework, pengembang dapat lebih fokus pada logika aplikasi tanpa harus membangun sistem dari awal (Sommerville, 2015).

2.1.8 Laravel

Laravel merupakan framework PHP yang digunakan untuk membangun aplikasi web dengan cepat dan efisien. Laravel dirancang untuk mempermudah proses pengembangan dengan menyediakan berbagai komponen yang mendukung pembuatan aplikasi web, seperti pengelolaan database, autentikasi, pengiriman email, dan caching (Stauffer, 2017).

Laravel menyediakan struktur dan API yang konsisten sehingga memudahkan pengembang dalam membangun aplikasi serta memahami alur kerja sistem. Selain itu, Laravel juga mengusung konsep convention over configuration, yaitu penggunaan konfigurasi bawaan sehingga pengembang tidak perlu melakukan pengaturan secara berulang, yang membuat proses pengembangan menjadi lebih cepat dibandingkan framework lain (Stauffer, 2017).

Laravel juga didukung oleh ekosistem yang lengkap, seperti tools untuk pengembangan lokal, manajemen Server, serta berbagai paket tambahan yang dapat digunakan untuk menambahkan fitur seperti autentikasi, WebSocket, dan integrasi lainnya. Dengan demikian, Laravel mampu mengurangi pekerjaan berulang dalam pengembangan aplikasi sehingga pengembang dapat lebih fokus pada logika bisnis sistem (Stauffer, 2017).

2.1.9 Model-View-Controller (MVC)

Model-View-Controller (MVC) adalah pola arsitektur perangkat lunak yang memisahkan presentasi dan interaksi dari data sistem. Sistem distrukturkan ke dalam tiga komponen logis yang saling berinteraksi satu sama lain (Sommerville, 2015).

Model merupakan komponen yang mengelola data sistem beserta operasi-operasi yang terkait dengan data tersebut. Dalam framework Laravel, komponen Model diimplementasikan menggunakan Eloquent ORM yang memungkinkan pengembang berinteraksi dengan basis data secara langsung melalui objek PHP.

View merupakan komponen yang mendefinisikan dan mengelola cara data ditampilkan kepada pengguna. Dalam framework Laravel, komponen View dibangun menggunakan Blade Template

Engine yang memungkinkan pembuatan tampilan halaman website secara dinamis dan dapat digunakan ulang.

Controller merupakan komponen yang mengelola interaksi pengguna dan meneruskan interaksi tersebut ke komponen View dan Model. Dalam framework Laravel, Controller berperan mengatur alur permintaan dari pengguna, memanggil Model untuk mengambil atau menyimpan data, kemudian mengembalikan respons melalui View yang sesuai.

Dalam penelitian ini, website absensi siswa dibangun menggunakan framework Laravel yang mengimplementasikan pola arsitektur MVC. Penggunaan MVC memungkinkan perubahan pada data dilakukan secara independen dari tampilannya, sehingga sistem menjadi lebih terstruktur dan mudah dipelihara (Sommerville, 2015).

2.1.10 Tailwind CSS

Tailwind CSS merupakan framework CSS berbasis pendekatan utility-first yang menyediakan berbagai class siap pakai seperti flex, pt-4, text-center, dan lainnya yang dapat dikombinasikan untuk membangun tampilan apapun langsung di dalam markup HTML (Tailwind CSS, 2023). Dengan pendekatan ini, pengembang tidak perlu menulis CSS secara terpisah sehingga proses pembangunan antarmuka pengguna menjadi lebih cepat dan efisien.

2.1.11 HyperText Markup Language (HTML)

HTML (HyperText Markup Language) adalah bahasa markup yang digunakan untuk membuat halaman web, memungkinkan penyajian berbagai informasi seperti teks, gambar, dan tautan pada web browser. HTML bukan bahasa pemrograman melainkan sistem untuk mengidentifikasi dan mendeskripsikan berbagai komponen dokumen seperti judul, paragraf, dan daftar (Robbins, 2012).

2.1.12 Cascading Style Sheets (CSS)

CSS (Cascading Style Sheets) adalah bahasa yang digunakan untuk mendeskripsikan tampilan halaman web, mengatur warna, tata letak, font, dan elemen visual lainnya. CSS memisahkan presentasi dari struktur HTML sehingga memungkinkan pengaturan tampilan yang konsisten di seluruh halaman situs secara efisien (Robbins, 2012).

2.1.13 MySQL

MySQL merupakan perangkat lunak sistem manajemen basis data berbasis SQL (Structured Query Language) yang bersifat multi-User dan banyak digunakan dalam pengelolaan data pada aplikasi web. MySQL tersedia sebagai perangkat lunak bebas yang dapat digunakan secara gratis di bawah lisensi GNU General Public License (GPL), sehingga memungkinkan pengguna untuk menggunakannya baik untuk keperluan pribadi maupun pengembangan sistem (Solichin, 2008).

MySQL dikembangkan oleh perusahaan MySQL AB yang didirikan oleh David Axmark, Allan Larsson, dan Michael “Monty” Widenius. Dalam penggunaannya, MySQL dikenal sebagai database yang fleksibel dan mampu diintegrasikan dengan berbagai bahasa pemrograman, termasuk PHP, sehingga banyak digunakan dalam pengembangan aplikasi web (Solichin, 2008).

Selain itu, MySQL memiliki beberapa keunggulan seperti bersifat bebas (*free*), stabil, memiliki tingkat keamanan yang baik, didukung oleh komunitas yang luas, serta mampu mendukung transaksi dan perkembangan sistem yang cepat (Solichin, 2008).

2.1.14 Application Programming Interface (API)

Application Programming Interface (API) merupakan sekumpulan aturan dan mekanisme yang memungkinkan aplikasi untuk saling berkomunikasi dan bertukar data. API menyediakan akses terhadap fungsi dan data sehingga memungkinkan interaksi antara berbagai sistem atau aplikasi yang berbeda (Masse, 2011).

Dalam konteks aplikasi web, API berperan sebagai penghubung antara klien dan Server. API menerima permintaan dari klien dan kemudian memprosesnya untuk menghasilkan respons yang sesuai, sehingga memungkinkan pertukaran informasi secara efisien antar sistem (Masse, 2011).

2.1.15 WhatsApp Gateway

WhatsApp *Gateway* merupakan layanan yang digunakan oleh penggunanya untuk berbagi pesan baik untuk mengirim maupun menerima. Dengan adanya WhatsApp *Gateway*, pesan dapat disebarkan ke ratusan nomor secara otomatis dan cepat tanpa harus mengetik secara manual karena data nomor diambil langsung dari database sehingga dapat menghemat waktu (Khoeriyah et al., 2021).

2.1.16 Unified Modeling Language (UML)

Unified Modeling Language (UML) merupakan sekumpulan notasi yang terdiri atas 13 jenis diagram untuk memodelkan sistem perangkat lunak, berdasarkan survei terhadap pengguna, lima di antaranya, yaitu activity diagram, use case diagram, sequence diagram, class diagram, dan state diagram, dinilai sudah cukup untuk merepresentasikan aspek-aspek penting suatu sistem (Sommerville, 2015). Model UML pada dasarnya merupakan abstraksi yang menonjolkan karakteristik penting dari sistem agar lebih mudah dipahami oleh pengembang maupun pemangku kepentingan, dan dapat digunakan sebagai sarana diskusi antar pengembang, dokumentasi sistem yang sedang berjalan, maupun dasar implementasi sistem (Sommerville, 2015).

Dalam konteks penelitian ini, UML digunakan sebagai alat untuk mendokumentasikan dan merancang website absensi siswa. Diagram-diagram UML yang digunakan meliputi Use Case Diagram, Activity Diagram, dan Sequence Diagram.

a. Use Case Diagram

Use case modeling digunakan untuk memodelkan interaksi antara sistem dengan agen eksternal, baik pengguna manusia maupun sistem lain, di mana setiap use case merepresentasikan satu tugas diskrit, use case digambarkan sebagai elips dan aktor sebagai simbol stick figure yang tidak selalu berupa manusia (Sommerville, 2015). Karena gambaran yang dihasilkan bersifat umum, deskripsi interaksi yang lebih lengkap biasanya dilengkapi dengan deskripsi tekstual, tabel, atau sequence diagram, dan composite use case diagram dapat digunakan apabila jumlah use case dalam sistem cukup banyak (Sommerville, 2015).

Pada website absensi siswa yang dikembangkan dalam penelitian ini, Use Case Diagram menggambarkan interaksi antara aktor Operator, Guru, Kepala Sekolah, dan Orang tua siswa dengan berbagai fungsionalitas sistem seperti pengelolaan data siswa, kelola absensi, rekap absensi, dan pengiriman pesan WhatsApp.

b. Activity Diagram

Activity diagram menggambarkan aktivitas dalam suatu proses beserta alur kendalinya (flow of control), dengan lingkaran penuh menandai awal proses, lingkaran penuh di dalam lingkaran lain menandai akhir proses, dan persegi panjang bersudut lengkung merepresentasikan aktivitas (Sommerville, 2015). Tanda panah merepresentasikan aliran kerja antaraktivitas, synchronization bar menunjukkan koordinasi ketika beberapa alur bertemu atau bercabang secara paralel, dan guard condition dalam tanda kurung siku dapat ditambahkan sebagai anotasi untuk menunjukkan syarat suatu alur diikuti (Sommerville, 2015).

Dalam penelitian ini, Activity Diagram digunakan untuk menggambarkan alur proses pencatatan kehadiran siswa, rekap absensi, serta pengiriman pesan WhatsApp kepada orang tua siswa secara otomatis setiap kali siswa tercatat dengan status Alpa, Izin, atau Sakit.

c. Sequence Diagram

Sequence diagram digunakan untuk memodelkan interaksi antara aktor dengan objek maupun antarobjek dalam suatu sistem, dengan urutan interaksi dibaca dari atas ke bawah melalui garis putus-putus vertikal (lifeline) yang menggambarkan rentang waktu keterlibatan tiap objek, sementara anotasi pada tanda panah menunjukkan pemanggilan method, parameter, dan nilai kembalian (Sommerville,

2015). Notasi alt juga dapat digunakan untuk menggambarkan kondisi alternatif, dengan setiap opsi interaksi dipisahkan oleh garis putus-putus (Sommerville, 2015).

Sequence Diagram pada penelitian ini digunakan untuk menggambarkan alur komunikasi antar objek dalam proses kelola absensi, mulai dari interaksi guru dengan sistem hingga pengiriman pesan melalui WhatsApp *Gateway* ke orang tua siswa.

2.1.17 *Black box testing*

Black-box testing atau pengujian perilaku merupakan metode pengujian perangkat lunak yang berfokus pada kebutuhan fungsional sistem. Pengujian ini dilakukan dengan menentukan kondisi masukan (*input*) yang dapat digunakan untuk menguji seluruh kebutuhan fungsional suatu program tanpa memperhatikan struktur kendali atau struktur internal perangkat lunak. *Black-box testing* bukan merupakan pengganti *white-box testing*, melainkan pendekatan yang saling melengkapi karena masing-masing metode dapat menemukan jenis kesalahan yang berbeda (Pressman, 2020).

Menurut Pressman dan Maxim, *black-box testing* digunakan untuk menemukan beberapa kategori kesalahan, yaitu fungsi yang tidak tepat atau tidak tersedia, kesalahan pada antarmuka, kesalahan pada struktur data atau akses basis data eksternal, kesalahan perilaku atau kinerja sistem, serta kesalahan pada proses inisialisasi dan terminasi (Pressman, 2020).

Berbeda dengan *white-box testing* yang dilakukan dengan memperhatikan struktur internal program, *black-box testing* mengabaikan struktur kendali dan memusatkan pengujian pada domain informasi serta perilaku sistem terhadap berbagai masukan. Pengujian ini digunakan untuk mengetahui validitas fungsi, perilaku dan kinerja sistem, karakteristik masukan yang dapat menghasilkan kasus uji, sensitivitas sistem terhadap nilai masukan tertentu, batas suatu kelas data, kemampuan sistem dalam menangani tingkat dan volume data, serta pengaruh kombinasi data tertentu terhadap operasi sistem (Pressman, 2020).

Berdasarkan uraian tersebut, *black-box testing* pada penelitian ini digunakan untuk menguji fungsi-fungsi sistem berdasarkan masukan (*input*) dan keluaran (*output*) yang dihasilkan. Hasil aktual sistem dibandingkan dengan hasil yang diharapkan untuk mengetahui apakah setiap fungsi sistem telah berjalan sesuai dengan kebutuhan yang telah ditentukan (Pressman, 2020).

Salah satu teknik yang digunakan untuk merancang kasus uji pada *black-box testing* adalah *Equivalence Partitioning* atau partisi ekuivalensi. Teknik ini membagi domain masukan suatu program menjadi beberapa kelas data, yaitu kelas valid yang berisi masukan sesuai spesifikasi dan kelas tidak valid yang berisi masukan yang menyimpang dari spesifikasi, kemudian satu kasus uji representatif

dipilih dari masing-masing kelas untuk mewakili seluruh anggota kelas tersebut, sehingga jumlah kasus uji dapat diminimalkan tanpa mengurangi cakupan pengujian terhadap kemungkinan kesalahan pada domain masukan (Pressman, 2020).

Selain itu, digunakan pula teknik *Boundary Value Analysis* atau analisis nilai batas untuk merancang kasus uji yang melibatkan rentang nilai, seperti validasi tanggal terhadap batas awal dan akhir suatu periode. Teknik ini berfokus pada pengujian nilai-nilai yang berada tepat pada, di atas, dan di bawah batas domain data, karena kesalahan perangkat lunak lebih sering terjadi di sekitar batas domain masukan dibandingkan pada bagian tengahnya (Pressman, 2020).

Kedua teknik tersebut digunakan sebagai dasar perancangan kasus uji pada penelitian ini sebagaimana diuraikan pada BAB 5, dengan setiap kasus uji dikelompokkan ke dalam skenario bermasukan data valid (mewakili kelas data valid) dan skenario bermasukan data tidak valid atau di luar rentang yang ditentukan (mewakili kelas data tidak valid maupun nilai di sekitar batas domain).

2.1.18 Acceptance testing

Acceptance testing merupakan bagian yang melekat pada pengembangan sistem yang bersifat custom. Pelanggan menguji sistem menggunakan data mereka sendiri, kemudian memutuskan apakah sistem tersebut layak diterima dari pengembang sistem. (Sommerville, 2015).

Terdapat enam tahapan dalam proses acceptance testing, yaitu sebagai berikut.

1. Define acceptance criteria

Tahap ini idealnya dilakukan sejak awal, sebelum kontrak sistem ditandatangani. Kriteria penerimaan seharusnya menjadi bagian dari kontrak sistem dan disetujui bersama oleh pelanggan dan pengembang. Namun pada praktiknya, menetapkan kriteria sedini itu cukup sulit karena kebutuhan yang rinci belum tentu tersedia, dan kebutuhan tersebut hampir pasti akan berubah selama proses pengembangan.

2. Plan acceptance testing

Tahap ini melibatkan penentuan sumber daya, waktu, dan anggaran untuk pengujian penerimaan, serta penetapan jadwal pengujian. Rencana pengujian penerimaan juga harus membahas cakupan kebutuhan yang akan diuji, urutan pengujian fitur sistem, serta risiko dalam proses pengujian (misalnya sistem crash atau performa tidak memadai) beserta cara mitigasinya.

3. Derive acceptance tests

Setelah kriteria penerimaan ditetapkan, kasus uji dirancang untuk memeriksa apakah sistem dapat diterima. Pengujian penerimaan harus mencakup baik karakteristik fungsional maupun non-fungsional (misalnya performa) dari sistem, dan idealnya memberikan cakupan yang lengkap terhadap seluruh kebutuhan sistem. Sommerville menekankan bahwa dalam praktiknya sulit menetapkan kriteria yang sepenuhnya objektif sering kali masih ada ruang perdebatan apakah suatu pengujian benar-benar sudah memenuhi kriteria tertentu.

4. Run acceptance tests

Pengujian penerimaan yang telah disepakati dijalankan terhadap sistem. Idealnya dilakukan di lingkungan nyata tempat sistem akan digunakan, meskipun hal ini bisa mengganggu operasional sehari-hari sehingga terkadang perlu dibuat lingkungan pengujian tersendiri. Karena pengujian ini melibatkan interaksi antara pengguna akhir dan sistem, proses ini sulit diotomatisasi sepenuhnya, dan pengguna akhir mungkin memerlukan pelatihan terlebih dahulu.

5. Negotiate test results

Sangat jarang seluruh pengujian penerimaan lolos tanpa masalah. Jika ditemukan masalah, pengembang dan pelanggan harus bernegosiasi untuk memutuskan apakah sistem sudah cukup baik untuk digunakan, sekaligus menyepakati bagaimana masalah yang ditemukan akan diperbaiki.

6. Reject/accept system

Tahap ini merupakan pertemuan antara pengembang dan pelanggan untuk memutuskan apakah sistem diterima atau tidak. Bila sistem dinilai belum layak, diperlukan pengembangan lebih lanjut, dan tahap pengujian penerimaan akan diulang kembali setelah perbaikan selesai.

Sommerville menambahkan bahwa keputusan penerimaan tidak selalu bersifat hitam-putih secara kontraktual pelanggan kadang tetap ingin menggunakan sistem sesegera mungkin karena manfaat penerapannya, sehingga hasil negosiasi bisa berupa penerimaan bersyarat (*conditional acceptance*), yaitu sistem diterima untuk mulai digunakan sambil pengembang memperbaiki masalah mendesak di rilis berikutnya.

Dalam konteks penelitian ini, acceptance testing digunakan untuk menguji website Absensi Siswa dengan Integrasi WhatsApp *Gateway* di SDN Cibitungkulon 02 oleh seluruh pengguna sistem, yaitu Operator, Guru, dan Kepala Sekolah, guna memastikan sistem telah sesuai dengan kebutuhan dan layak digunakan dalam proses pencatatan kehadiran siswa sehari-hari.

2.1.19 Wawancara

Wawancara merupakan salah satu teknik yang paling umum digunakan dalam proses rekayasa kebutuhan (requirements engineering). Melalui wawancara, tim rekayasa kebutuhan mengajukan pertanyaan kepada pemangku kepentingan (stakeholder) mengenai sistem yang sedang berjalan maupun sistem yang akan dikembangkan, sehingga kebutuhan sistem dapat diturunkan dari jawaban-jawaban yang diperoleh (Sommerville, 2015).

Menurut Sommerville (2015), wawancara dalam konteks rekayasa kebutuhan dapat dibedakan menjadi dua jenis, yaitu:

1. Wawancara tertutup (closed interview), yaitu wawancara yang menggunakan sekumpulan pertanyaan yang telah ditentukan sebelumnya.
2. Wawancara terbuka (open interview), yaitu wawancara tanpa agenda yang baku, di mana tim rekayasa kebutuhan mengeksplorasi berbagai isu bersama stakeholder untuk memperoleh pemahaman yang lebih mendalam mengenai kebutuhan mereka.

Dalam praktiknya, kedua jenis wawancara ini biasanya digunakan secara kombinasi. Pertanyaan terstruktur digunakan sebagai titik awal untuk menjaga arah wawancara tetap fokus pada sistem yang akan dibangun, namun tetap memberi ruang bagi diskusi yang lebih bebas ketika muncul isu-isu baru yang relevan.

Wawancara efektif digunakan untuk memperoleh gambaran umum mengenai pekerjaan stakeholder, cara mereka berinteraksi dengan sistem yang baru, serta kendala yang dihadapi pada sistem yang sedang berjalan. Namun demikian, tanpa adanya prototipe sistem sebagai alat bantu, stakeholder umumnya sulit untuk menyampaikan kebutuhan secara spesifik dan rinci, sehingga tim rekayasa kebutuhan perlu menganalisis lebih lanjut informasi yang terkumpul untuk merumuskan kebutuhan yang sesungguhnya (Sommerville, 2015).

2.1.20 Skala Guttman

Skala Guttman merupakan salah satu jenis skala sikap yang digunakan untuk memperoleh jawaban tegas dan pasti dari responden, seperti "ya" atau "tidak", "benar" atau "salah", maupun "pernah" atau "tidak pernah" (Septiani et al., 2025). Berbeda dengan skala Likert yang menyediakan beberapa tingkatan jawaban, skala Guttman hanya menyediakan dua alternatif jawaban sehingga data yang dihasilkan bersifat dikotomis (Septiani et al., 2025). Skala ini bersifat kumulatif atau hierarkis, di mana pernyataan disusun berjenjang dari yang paling ringan hingga paling berat untuk disetujui, sehingga apabila responden menyetujui pernyataan pada tingkat tertentu, secara logis ia juga menyetujui pernyataan pada tingkat di bawahnya, jumlah item pada skala ini juga relatif ringkas, umumnya tiga sampai empat pernyataan per unit sikap, sehingga lebih mudah dianalisis dan diinterpretasikan dibandingkan skala dengan banyak tingkatan (Septiani et al., 2025).

Dalam penelitian ini, item skala Guttman disusun dalam bentuk pernyataan mengenai pengalaman penggunaan sistem yang direspons dengan pilihan "Ya" (diberi skor 1) atau "Tidak" (diberi skor 0).

2.2 Penelitian Terdahulu

Berikut ini adalah penelitian terdahulu yang penulis jadikan sebagai referensi untuk tugas akhir atau penelitian yang hampir sama dengan topik tugas akhir:

Tabel 2.1 Penelitian Terdahulu

No	Peneliti	Judul	Hasil Penelitian	Alasan Digunakan
1	Nur Aizah & Firman Santoso (2024)	Implementasi Sistem Absensi Siswa Berbasis Web Menggunakan WhatsApp Gateway di SDN 2 Seletreng	Sistem absensi berbasis web yang mengintegrasikan WhatsApp Gateway berhasil meningkatkan efisiensi operatoristrasi sekolah. Sistem memungkinkan pencatatan kehadiran siswa secara real-time, meminimalkan kesalahan manual, serta memfasilitasi pengiriman pesan otomatis kepada orang tua siswa melalui pesan WhatsApp ketika siswa tidak hadir.	Digunakan karena memiliki kesamaan konteks penelitian yaitu sistem absensi berbasis web dengan WhatsApp Gateway pada sekolah dasar. Penelitian ini juga menggunakan metode Waterfall yang sesuai
2	Moh. Baha'uddin, Adi Susanto & A. Hamdani (2025)	Implementasi RFID pada Sistem Informasi Presensi Siswa di MAS Al-Huda Kangayan dengan Pesan WhatsApp Gateway	Sistem presensi berbasis RFID yang terintegrasi dengan WhatsApp Gateway berhasil mengurangi waktu pencatatan dari rata-rata 3,5 menit menjadi ± 35 detik untuk 10 siswa per kelas, serta meningkatkan efisiensi waktu sebesar $\pm 83\%$. Pesan kehadiran dikirimkan secara otomatis dan real-time kepada orang tua siswa melalui WhatsApp. Sistem dikembangkan menggunakan metode Waterfall.	Digunakan karena memberikan referensi integrasi WhatsApp Gateway pada sistem presensi siswa. Penelitian ini menggunakan metode Waterfall yang relevan dengan penelitian yang dibangun. Perbedaannya terletak pada penggunaan RFID.
3	Paiman Nababan, Resianta Perangin-angin & Elvianti N. Purba (2022)	Sistem Informasi Absensi Siswa Pada SMK Negeri 1 Pantai Labu Berbasis Web dengan WhatsApp Gateway	Sistem informasi absensi berbasis web dengan WhatsApp Gateway mampu mengatasi permasalahan pencatatan manual yang sering hilang dan tidak akurat di SMK Negeri 1 Pantai Labu. Guru dapat mengabsensi siswa langsung melalui website, dan sistem menyediakan fitur pengiriman pesan WhatsApp kepada orang tua terkait informasi kehadiran anaknya. Sistem dibangun menggunakan PHP, HTML, CSS, dan JavaScript dengan basis data MySQL.	Digunakan sebagai referensi pembangunan sistem absensi berbasis website dengan WhatsApp Gateway. Penelitian ini juga menggunakan metode Waterfall sehingga sesuai dengan penelitian yang diterapkan.
4	Ferlando Mbay Rahi, Erwianta Gustial Radjah & Alfrian Carmen Talakua (2024)	Integrasi WhatsApp Gateway dalam Sistem Informasi Absensi Berbasis Website di SMA Negeri 1 Pandawai	Integrasi WhatsApp Gateway berbasis website berhasil meningkatkan efisiensi pengelolaan absensi dan memperkuat keterlibatan orang tua siswa. Sistem memungkinkan guru mencatat kehadiran secara digital dan mengirimkan pesan ketidakhadiran ke orang tua siswa dalam waktu 3–5 detik. Laporan absensi tersedia dalam format PDF. Pengujian menggunakan metode black box dengan seluruh fungsionalitas berjalan berhasil.	Digunakan karena memberikan referensi penerapan WhatsApp Gateway pada sistem absensi berbasis website. Penelitian ini menggunakan metode Waterfall yang sejalan dengan penelitian yang dibangun.

No	Peneliti	Judul	Hasil Penelitian	Alasan Digunakan
5	Mas Muhammadun Zahid & Aditya Prapanca (2023)	Rancang Bangun <i>Website</i> Absensi Menggunakan RFID dan WhatsApp untuk Meningkatkan Kedisiplinan dan Nilai Rapor Siswa	<i>Website</i> absensi berbasis RFID dan WhatsApp yang dikembangkan dengan metode Agile Development terbukti meningkatkan kedisiplinan siswa. Persentase kehadiran naik dari 88% (konvensional) menjadi 98% (sistem berbasis website). Hasil validasi media mencapai 87,22% dengan kategori "Sangat Valid". Orang tua siswa dapat memantau kehadiran anak secara real-time melalui pesan WhatsApp. Sistem dibangun menggunakan Angular CLI (frontend) dan NestJS (backend).	Digunakan sebagai referensi dampak sistem absensi digital terhadap kedisiplinan siswa. Perbedaannya terletak pada metode pengembangan yang digunakan, yaitu Agile Development, sedangkan penelitian yang dibangun menggunakan metode Waterfall.

Tabel 2.1 menyajikan beberapa penelitian terdahulu yang berkaitan dengan sistem absensi berbasis website, dan integrasi WhatsApp *Gateway*. Tabel ini digunakan sebagai referensi untuk mengetahui perbandingan penelitian sebelumnya dengan penelitian yang dilakukan.

BAB 3

SKEMA PENELITIAN

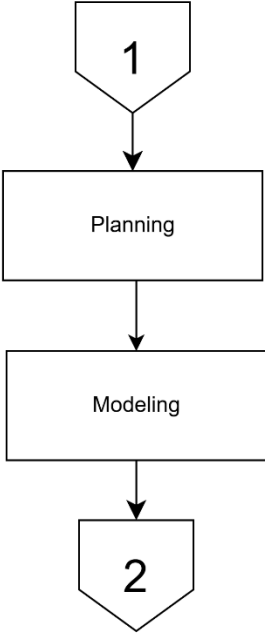
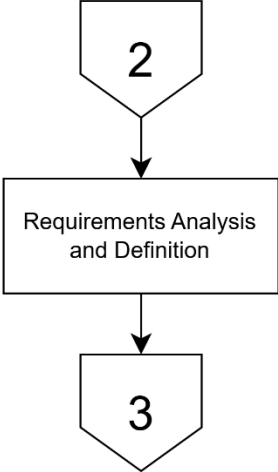
Bab ini menjelaskan alur penyelesaian tugas akhir, perumusan masalah, kerangka pemikiran teoritis, dan profil tempat penelitian sebagai panduan awal bagi pembaca untuk memahami tujuan, ruang lingkup, dan landasan penelitian yang dilakukan.

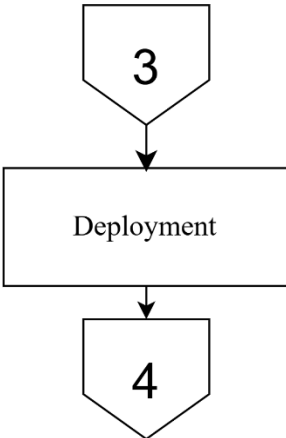
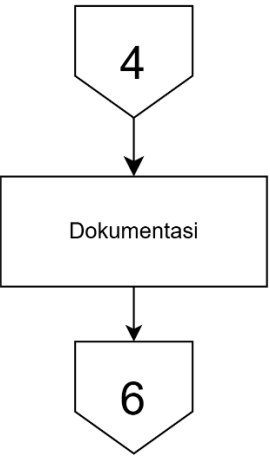
3.1 Alur Penyelesaian Tugas

Alur penyelesaian tugas akhir ini dilakukan berdasarkan diagram alur penyelesaian. Berikut ini merupakan alur penyelesaian tugas akhir:

Tabel 3.1 Alur Penyelesaian Tugas Akhir

Tahap dan Hasil	Langkah Penelitian	Literatur dan Referensi
Tahap 1 : Identifikasi Masalah dan Communication Tahap ini mengidentifikasi permasalahan pada proses absensi siswa di SDN Cibitungkulon 02 yang masih dilakukan secara manual, lalu dilanjutkan dengan communication untuk menggali kebutuhan sistem melalui observasi, wawancara, dan studi literatur, guna menentukan ruang lingkup pengembangan website absensi siswa berbasis WhatsApp Gateway (Pressman, 2020). Hasil : • Identifikasi proses absensi yang masih dilakukan secara manual • Identifikasi kendala pada proses rekap absensi • Identifikasi proses pengiriman <i>Pesan</i> yang masih dilakukan secara manual • Rumusan tujuan pengembangan sistem • Batasan dan ruang lingkup penelitian Kontribusi : Memberikan dasar yang jelas untuk menentukan tujuan dan ruang lingkup pengembangan sistem sejak awal.	<pre>graph TD; A[Identifikasi Masalah] --> B[Communication]; B --> C{{1}};</pre>	

Tahap 2 : Planning dan Modeling Tahap planning mencakup estimasi sumber daya, penjadwalan, dan tracking perkembangan pembangunan sistem, sedangkan tahap modeling mencakup analisis kebutuhan sistem dan design perancangan sistem meliputi Use Case Diagram, Activity Diagram, Sequence Diagram, dan ERD (Pressman, 2020). Hasil : • Kebutuhan fungsional sistem • Kebutuhan non-fungsional sistem • Identifikasi aktor sistem • Use Case Diagram • Skenario diagram • Activity Diagram • Sequence diagram • ERD Kontribusi : Menjadi pedoman terarah bagi proses pengembangan serta menghasilkan spesifikasi kebutuhan dan rancangan sistem sebagai dasar tahap construction.	 <pre> graph TD 1{{1}} --> Planning[Planning] Planning --> Modeling[Modeling] Modeling --> 2{{2}} </pre>	(Pressman, 2020).
Tahap 3 : Construction Tahap <i>construction</i> membangun kode program menggunakan <i>framework</i> Laravel dan MySQL, dilanjutkan dengan test menggunakan black-box testing untuk memastikan setiap fungsi berjalan sesuai kebutuhan (Pressman, 2020). Hasil : • Implementasi sistem absensi siswa berbasis web • Integrasi WhatsApp Gateway • Implementasi fitur sistem • Hasil Blackbox Testing dan User Acceptance Testing Kontribusi : Menghasilkan <i>website</i> absensi siswa yang membantu proses absensi dan pengiriman <i>Pesan</i> otomatis.	 <pre> graph TD 2{{2}} --> RA[Requirements Analysis and Definition] RA --> 3{{3}} </pre>	(Pressman, 2020).

Tahap 4 : Deployment Tahap <i>deployment</i> menerapkan perangkat lunak yang telah dibangun dan diuji ke lingkungan produksi menggunakan platform Railway (Platform as a Service), dengan aplikasi dikemas dalam container Dockerfile (Pressman, 2020), dilanjutkan kegiatan <i>operation and maintenance</i> berupa dukungan, pemeliharaan, dan perbaikan sistem setelah diterapkan. Hasil : • Aplikasi berhasil diterapkan pada platform <i>Railway</i> • Aplikasi dapat diakses melalui domain *.up.railway.app • Sistem siap digunakan • Pemeliharaan dan perbaikan sistem Kontribusi : Memastikan sistem berjalan sesuai kebutuhan dan dapat digunakan secara berkelanjutan.	 <pre> graph TD 3{{3}} --> Deployment[Deployment] Deployment --> 4{{4}} </pre>	(Pressman, 2020).
Tahap 5 : Dokumentasi Tahap akhir ini menyusun laporan tugas akhir secara menyeluruh berdasarkan seluruh tahapan penelitian sebelumnya, beserta dokumentasi teknis sistem dan penarikan kesimpulan serta saran pengembangan selanjutnya. Hasil : • Laporan tugas akhir yang telah disusun secara lengkap • Dokumentasi teknis sistem • Kesimpulan hasil penelitian dan saran pengembangan sistem selanjutnya Kontribusi : Memberikan gambaran menyeluruh mengenai proses penelitian serta menjadi bahan evaluasi dan rekomendasi pengembangan sistem di masa mendatang.	 <pre> graph TD 4{{4}} --> Dokumentasi[Dokumentasi] Dokumentasi --> 6{{6}} </pre>	(Pressman, 2020).

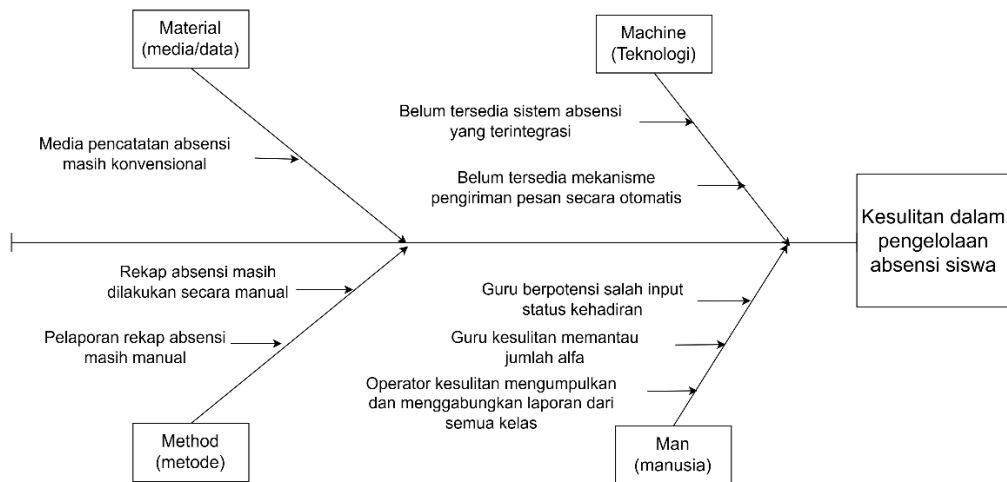
Tabel 3.1 merangkum lima tahapan penyelesaian tugas akhir, mulai dari identifikasi masalah hingga dokumentasi.

3.2 Perumusan Masalah

Analisis masalah yang dilakukan untuk mengidentifikasi penyebab dari permasalahan dan untuk mengetahui hubungan sebab akibat permasalahan. Hubungan dari analisis masalah dengan solusi tugas akhir untuk mengetahui ketepatan solusi dalam menyelesaikan permasalahan tugas akhir.

3.2.1 Analisis Masalah

Analisis faktor-faktor permasalahan yang terjadi dalam sistem menjadi fokus analisis. Dibawah ini merupakan deskripsi dari penyebab masalah yang terjadi



Gambar 3.1 Fishbone Masalah

Gambar 3.1 merupakan diagram fishbone yang digunakan untuk mengidentifikasi faktor-faktor penyebab permasalahan dalam pengelolaan absensi siswa. Diagram ini membantu menggambarkan hubungan sebab-akibat dari berbagai aspek seperti man (manusia), material (media), machine (teknologi), method (metode) yang mempengaruhi sistem yang berjalan.

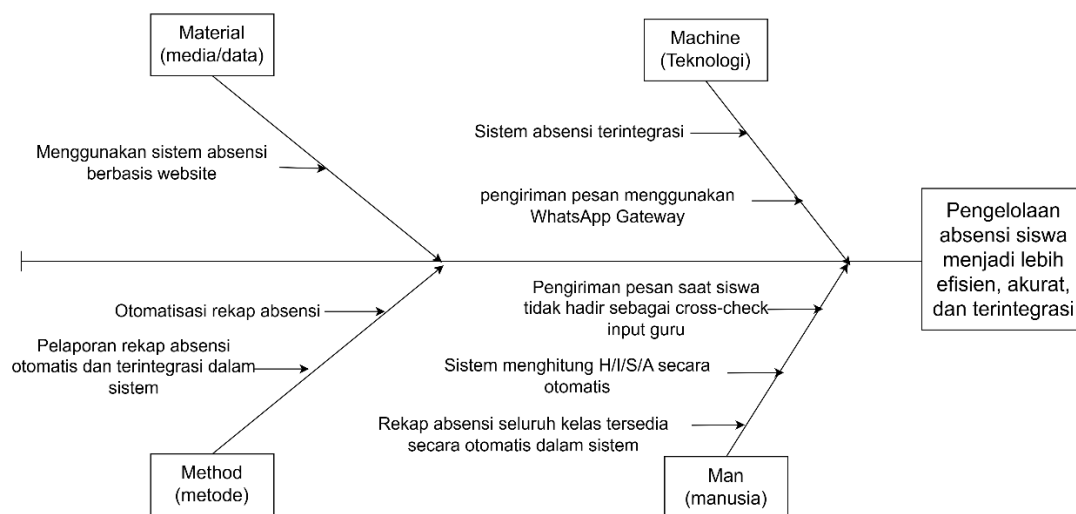
Tabel 3.2 Analisis Masalah

Aspek	Penyebab	Deskripsi Masalah
Man (Manusia)	Guru kesulitan memantau jumlah alfa	Guru harus memeriksa data absensi secara manual untuk mengetahui jumlah alfa siswa sehingga proses monitoring menjadi kurang efisien.
	Operator kesulitan mengumpulkan dan menggabungkan laporan dari semua kelas	Operator membutuhkan waktu lebih lama dalam mengelola laporan absensi karena data dikumpulkan dan digabungkan secara manual dari setiap kelas.
	Guru berpotensi salah input status kehadiran	Kesalahan input status kehadiran oleh guru tidak dapat langsung terdeteksi karena belum ada pihak lain yang memverifikasi data sebelum tersimpan sebagai catatan resmi.
Machine (Teknologi)	Belum tersedia sistem absensi yang terintegrasi	Data absensi belum tersimpan secara terpusat sehingga proses pengelolaan dan pencarian data menjadi lebih sulit.
	Belum tersedia mekanisme pengiriman pesan secara otomatis	Penyampaian informasi absensi kepada orang tua siswa belum dapat dilakukan secara otomatis dan real-time.
Method (Metode)	Rekap absensi masih dilakukan secara manual	Proses rekap absensi membutuhkan waktu lebih lama dan berpotensi terjadi kesalahan dalam pengolahan data absensi.
	Pelaporan rekap absensi masih manual	Penyampaian rekap absensi dari guru ke operator menjadi tidak efisien dan belum terintegrasi dalam satu sistem.
Material (Media/Data)	Media pencatatan absensi masih konvensional	Proses pencatatan absensi masih menggunakan media konvensional sehingga penggunaan data menjadi kurang efisien dan belum terdigitalisasi.

Tabel 3.2 merangkum hasil analisis masalah pada sistem absensi siswa yang sedang berjalan beserta penyebab dan dampaknya.

3.2.2 Solusi Masalah

Permasalahan yang ada harus memiliki sebuah solusi yang dapat menyelesaikan masalahnya, hal ini saling terhubung untuk dapat bisa menjawab setiap masalah dengan solusi yang baik, berikut adalah solusinya:



Gambar 3.2 *Fishbone* Solusi

Gambar 3.2 merupakan diagram fishbone solusi yang digunakan untuk menggambarkan pendekatan penyelesaian dari permasalahan yang telah diidentifikasi sebelumnya. Diagram ini menunjukkan solusi yang dirancang berdasarkan aspek man (manusia), material (media), machine (teknologi), method (metode) untuk meningkatkan efisiensi pengelolaan absensi siswa.

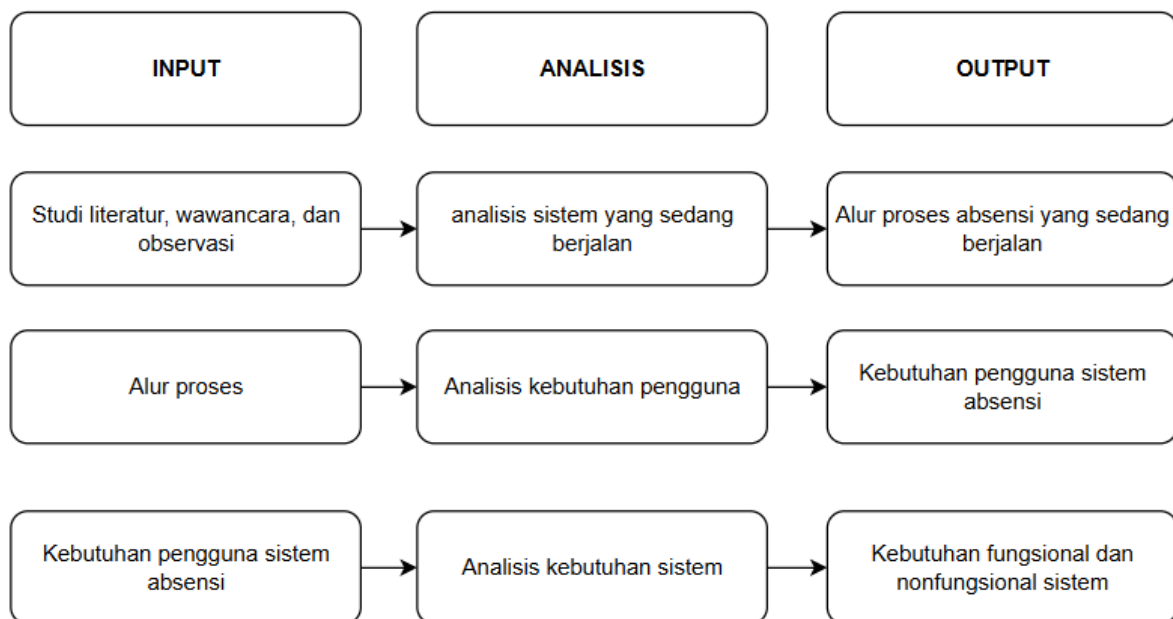
Tabel 3.3 Solusi Masalah

Aspek	Solusi	Deskripsi Solusi
Man (Manusia)	Pengiriman pesan WhatsApp otomatis kepada orang tua sebagai mekanisme cross-check terhadap input guru	Setiap kali guru mencatat status Alpa, Izin, atau Sakit, sistem otomatis mengirim pesan verifikasi ke orang tua sehingga kesalahan input guru dapat segera diketahui dan dikonfirmasi
	Sistem menghitung jumlah Hadir/Izin/Sakit/Alpa (H/I/S/A) setiap siswa secara otomatis	Guru dan operator tidak perlu lagi menghitung secara manual, cukup melihat rekap yang sudah tersedia pada dashboard
	Rekap absensi seluruh kelas tersedia otomatis dalam sistem	Operator tidak perlu lagi mengumpulkan dan menggabungkan laporan dari setiap guru secara manual
Machine (Teknologi)	Sistem absensi terintegrasi berbasis website	Sistem menyediakan platform terpusat untuk mencatat, menyimpan, dan mengelola seluruh data absensi secara digital
	Pengiriman pesan menggunakan WhatsApp Gateway (Fonnte)	Sistem terhubung dengan <i>API Fonnte</i> untuk mendukung pengiriman pesan WhatsApp otomatis kepada orang tua siswa tanpa intervensi manual

Aspek	Solusi	Deskripsi Solusi
Method (Metode)	Otomatisasi rekap absensi	Sistem menghasilkan rekap harian, bulanan, dan semester secara otomatis berdasarkan data absensi yang tersimpan, sehingga mengurangi risiko salah hitung pada perhitungan manual
	Pelaporan rekap absensi otomatis dan terintegrasi dalam sistem	Data absensi dapat langsung diakses oleh operator dan kepala sekolah tanpa perlu proses penyerahan laporan secara manual
Material (Media/Data)	Menggunakan sistem absensi berbasis website	Media pencatatan yang semula konvensional (buku fisik) beralih menjadi input digital melalui website yang langsung tersimpan di basis data terpusat

Tabel 3.3 merangkum solusi yang diusulkan, dikelompokkan ke dalam 4 kategori 4M seperti pada Tabel 3.2. Kolom "Menjawab" menunjukkan rumusan masalah pada Subbab 1.2 yang dijawab tiap solusi.

3.2.3 Skema Analisis



Gambar 3.3 Skema Analisis

Gambar 3.3 merupakan skema analisis yang digunakan untuk menggambarkan alur pengolahan data dalam penelitian, mulai dari Input, proses analisis, hingga menghasilkan Output. Skema ini digunakan sebagai dasar dalam memahami kebutuhan sistem dan menjadi acuan dalam proses perancangan website absensi siswa.

Tabel 3.4 Deskripsi Skema Analisis

Langkah Analisis	Objek Analisis	Hasil Analisis	Maksud Analisis
Analisis sistem yang sedang berjalan	Studi literatur, wawancara, dan observasi	Alur proses absensi yang sedang berjalan	Mengidentifikasi proses absensi yang berjalan saat ini serta permasalahan yang terjadi pada sistem absensi di sekolah
Analisis kebutuhan pengguna	Alur proses	Kebutuhan pengguna sistem absensi	Mengetahui kebutuhan pengguna terhadap sistem absensi yang akan dibangun
Analisis kebutuhan sistem	Kebutuhan pengguna	Kebutuhan fungsional dan nonfungsional sistem	Untuk menentukan kebutuhan sistem yang akan dikembangkan sesuai kebutuhan pengguna

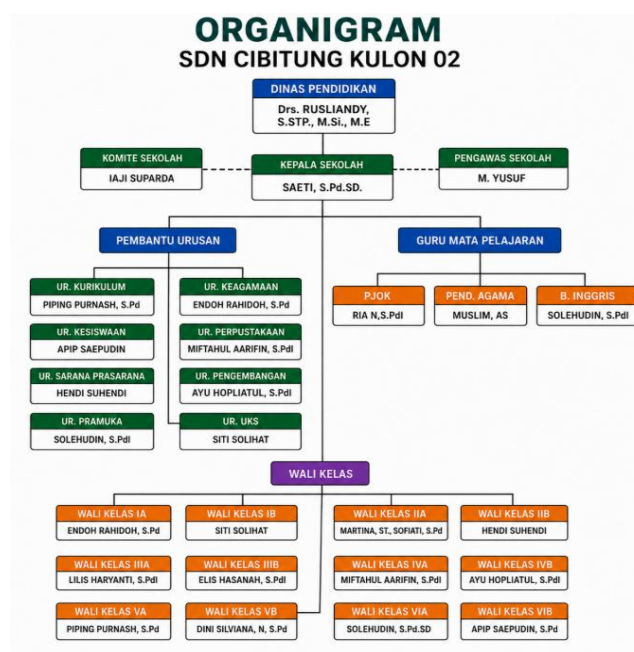
Tabel 3.4 merinci tiap tahapan skema analisis, meliputi langkah, objek, hasil, dan maksud analisis yang dilakukan.

3.3 Profil Tempat Penelitian

Penelitian ini dilaksanakan di SDN Cibitungkulon 02, sebuah sekolah dasar negeri yang menyelenggarakan pendidikan pada jenjang sekolah dasar. SDN Cibitungkulon 02 merupakan salah satu institusi pendidikan yang berperan dalam membentuk pengetahuan, keterampilan, serta karakter siswa.

Lokasinya berada di Cibitungkulon, Kecamatan Pamijahan, Kabupaten Bogor, Jawa Barat 16630. Sekolah ini menjadi lokasi penelitian karena masih menggunakan sistem absensi konvensional sehingga relevan untuk pengembangan website absensi berbasis web.

3.3.1 Struktur Organisasi



Gambar 3.4 Struktur Organisasi

Gambar 3.4 menggambarkan struktur organisasi SDN Cibitungkulon 02 beserta pembagian tugas tiap bagian.

3.3.2 Visi dan Misi

a. Visi

“Terwujudnya peserta didik yang berakhlak mulia, berprestasi, serta peduli lingkungan.”

b. Misi

1. Membangun lingkungan sekolah yang membentuk peserta didik memiliki akhlak mulia melalui rutinitas kegiatan keagamaan dan menerapkan ajaran agama melalui cara berinteraksi di sekolah.
2. Merancang pembelajaran yang menarik dan menyenangkan yang mampu memotivasi peserta didik untuk selalu belajar dan menemukan pembelajaran.
3. Mengembangkan kemandirian, bernalar kritis dan kreativitas yang memfasilitasi keragaman minat dan bakat peserta didik.
4. Menumbuhkembangkan semangat berkompetisi dan bermental juara secara intensif kepada peserta didik melalui kegiatan-kegiatan lomba baik dalam bidang akademik maupun non akademik.
5. Mengembangkan program sekolah yang membentuk ide dan gagasan cepat tanggap terhadap perubahan yang terjadi untuk merancang inovasi.
6. Mengembangkan dan memfasilitasi peningkatan prestasi peserta didik sesuai minat dan bakatnya melalui proses pendampingan dan kerja sama dengan orang tua siswa.
7. Mengembangkan sikap peduli lingkungan kepada seluruh warga sekolah agar berpola hidup bersih dan sehat melalui pembiasaan 10K (kedisiplinan, ketertiban, kerukunan, kebersihan, kerindangan, kekeluargaan, keindahan, keamanan, kesehatan, dan keteladanan) dan lomba kelas bersih.
8. Membudayakan karakter peduli lingkungan dengan menerapkan operasi semut setelah upacara hari senin.

3.4 Teknik Pengumpulan Data

Teknik pengumpulan data merupakan metode yang digunakan untuk memperoleh informasi yang dibutuhkan dalam penelitian, sekaligus sebagai alat untuk memahami proses absensi yang sedang berjalan, mengidentifikasi permasalahan yang terjadi, serta mengetahui kebutuhan pengguna terhadap website absensi siswa yang akan dikembangkan.

a. Wawancara

Wawancara dilakukan dengan empat narasumber di SDN Cibitungkulon 02, yaitu satu Kepala Sekolah, satu Operator, dan dua Guru, yang dipilih secara purposive berdasarkan peran masing-masing terhadap proses absensi. Kepala Sekolah, yaitu Ibu Saeti, dipilih karena perannya yang tunggal di sekolah sebagai pemegang kebijakan sekaligus pihak yang menerima rekap kehadiran lintas kelas dan laporan absensi secara keseluruhan. Operator, yaitu Bapak Miftahul Arifin, dipilih karena merangkap peran sebagai operator sekaligus guru, sehingga memahami proses absensi baik dari sisi pengelolaan data sekolah secara menyeluruh (data siswa, guru, kelas, dan rekap absensi) maupun dari sisi pencatatan kehadiran harian di kelas. Dua guru, yaitu Bapak Solehudin dan Ibu Endoh Raudhoh, dipilih untuk mewakili variasi kondisi kelas sekaligus memastikan proses absensi manual yang diamati konsisten di berbagai kelas hingga mencapai titik jenuh (data saturation), sehingga diperoleh gambaran permasalahan yang representatif dari sudut pandang guru. Secara khusus, Bapak Solehudin dipilih karena merangkap peran sebagai guru sekaligus guru mata pelajaran, sehingga memahami kondisi proses absensi dari kedua peran tersebut.

b. Observasi

Observasi dilakukan dengan mengamati secara langsung proses pencatatan kehadiran siswa, proses rekap data absensi, serta proses penyampaian informasi kepada orang tua siswa di SDN Cibitungkulon 02. Hasil observasi digunakan untuk memahami alur kerja sistem yang sedang berjalan serta mengidentifikasi permasalahan yang terjadi dalam proses pengelolaan absensi siswa.

c. Studi Literatur

Studi literatur dilakukan dengan mengumpulkan referensi dari buku, jurnal, dan penelitian terdahulu yang berkaitan dengan absensi siswa, website, serta teknologi WhatsApp *Gateway*. Studi ini bertujuan untuk memperoleh landasan teori yang mendukung penelitian.

Hasil dari wawancara, observasi, studi literatur, dan dokumentasi digunakan sebagai dasar dalam analisis kebutuhan sistem dan perancangan website absensi siswa dengan integrasi WhatsApp *Gateway*.

BAB 4

ANALISIS DAN PERANCANGAN

Bab ini berisi mengenai konsep aplikasi dan desain aplikasi berdasarkan kebutuhan yang telah dipaparkan. Di dalamnya berisi deskripsi mengenai analisis sistem yang sedang berjalan, analisis kebutuhan sistem, perancangan sistem menggunakan diagram UML, serta rancangan basis data Website Absensi Siswa dengan Pesan WhatsApp *Gateway* pada SDN Cibatungkulon 02.

4.1 Analisis

Pada tahap ini dilakukan analisis terhadap proses dan kebutuhan sistem absensi siswa yang sedang berjalan di SDN Cibatungkulon 02. Analisis dilakukan untuk memahami permasalahan, alur aktivitas, serta kebutuhan pengguna sebagai dasar dalam pengembangan sistem.

4.1.1 Teknik Analisis

Penelitian ini menggunakan teknik analisis berdasarkan hasil wawancara yang dilakukan secara langsung kepada pihak-pihak terkait di SDN Cibatungkulon 02. Wawancara dilakukan kepada empat narasumber, yaitu Ibu Saeti selaku Kepala Sekolah, Bapak Miftahul Arifin selaku Operator, serta Bapak Solehudin dan Ibu Endoh Raudhoh selaku Guru.

Wawancara dengan Kepala Sekolah mencakup kebijakan sekolah terkait pengelolaan kehadiran siswa, kebutuhan terhadap rekap dan laporan kehadiran lintas kelas, serta harapan terhadap sistem baru yang akan dibangun. Wawancara dengan Operator mencakup pembahasan mengenai pengelolaan data absensi, pengelolaan data siswa, guru, dan kelas, serta kesiapan infrastruktur sekolah. Wawancara dengan Guru mencakup kondisi sistem absensi yang berjalan, proses rekap, mekanisme pengiriman informasi ketidakhadiran kepada orang tua siswa, serta kebutuhan dan harapan terhadap sistem baru yang akan dibangun.

Hasil wawancara kemudian dianalisis untuk menjadi dasar dalam menentukan kebutuhan fungsional sistem, fitur-fitur yang diperlukan, serta alur kerja sistem yang akan dirancang.

4.1.2 Analisis Sistem yang Sedang Berjalan

Analisis sistem yang sedang berjalan dilakukan untuk mengetahui proses absensi siswa yang saat ini diterapkan di SDN Cibatungkulon 02. Berdasarkan hasil observasi dan wawancara yang telah dilakukan, proses absensi siswa masih dilakukan secara manual menggunakan buku absensi kelas dan belum terintegrasi dalam sistem berbasis digital.

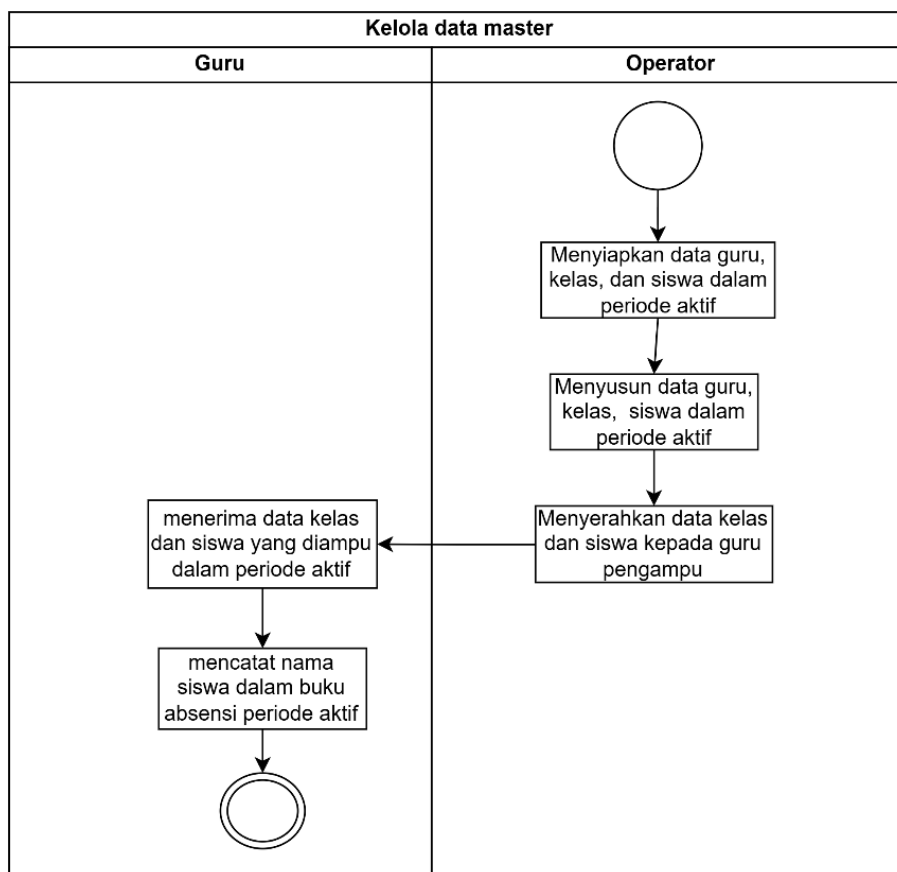
Pada proses absensi, guru melakukan pencatatan kehadiran siswa secara manual di dalam buku absensi pada setiap kegiatan pembelajaran berlangsung. Setelah proses absensi selesai, data kehadiran

siswa direkap kembali secara manual menggunakan Microsoft Excel sebelum diserahkan kepada operator sekolah sebagai laporan absensi. Proses rekap yang dilakukan secara manual menyebabkan pengelolaan data membutuhkan waktu yang cukup lama serta berpotensi menimbulkan kesalahan pencatatan. Kesalahan tersebut bukan hanya potensi, melainkan pernah benar-benar terjadi berdasarkan hasil wawancara, tercatat kasus di mana orang tua siswa mengeluh kepada guru karena status kehadiran anaknya tidak sesuai dengan kondisi sebenarnya. Selain itu, data absensi belum tersimpan dalam basis data terpusat sehingga proses pencarian dan pengelolaan data menjadi kurang efisien.

Dalam proses penyampaian informasi ketidakhadiran siswa kepada orang tua siswa, guru masih melakukan pengiriman pesan WhatsApp secara manual satu per satu. Proses ini dinilai kurang efisien dan berpotensi menyebabkan keterlambatan penyampaian informasi, bahkan guru dapat lupa atau terlambat mengirimkan Pesan kepada orang tua siswa.

4.1.3 Analisis Proses Bisnis

Analisis yang difokuskan dalam penelitian ini adalah proses bisnis absensi siswa yang sedang berjalan, yang meliputi proses pencatatan kehadiran siswa, rekap absensi, serta penyampaian informasi ketidakhadiran kepada orang tua siswa secara manual melalui pesan WhatsApp apabila siswa tercatat Absen sebanyak tiga kali atau lebih. Selain proses utama tersebut, analisis juga mencakup proses pengelolaan data guru, siswa, dan kelas periode yang masih dilakukan secara manual, sehingga proses absensi belum berjalan secara terstruktur dan terintegrasi dalam satu platform.



Gambar 4.1 Activity Diagram Persiapan Data Master

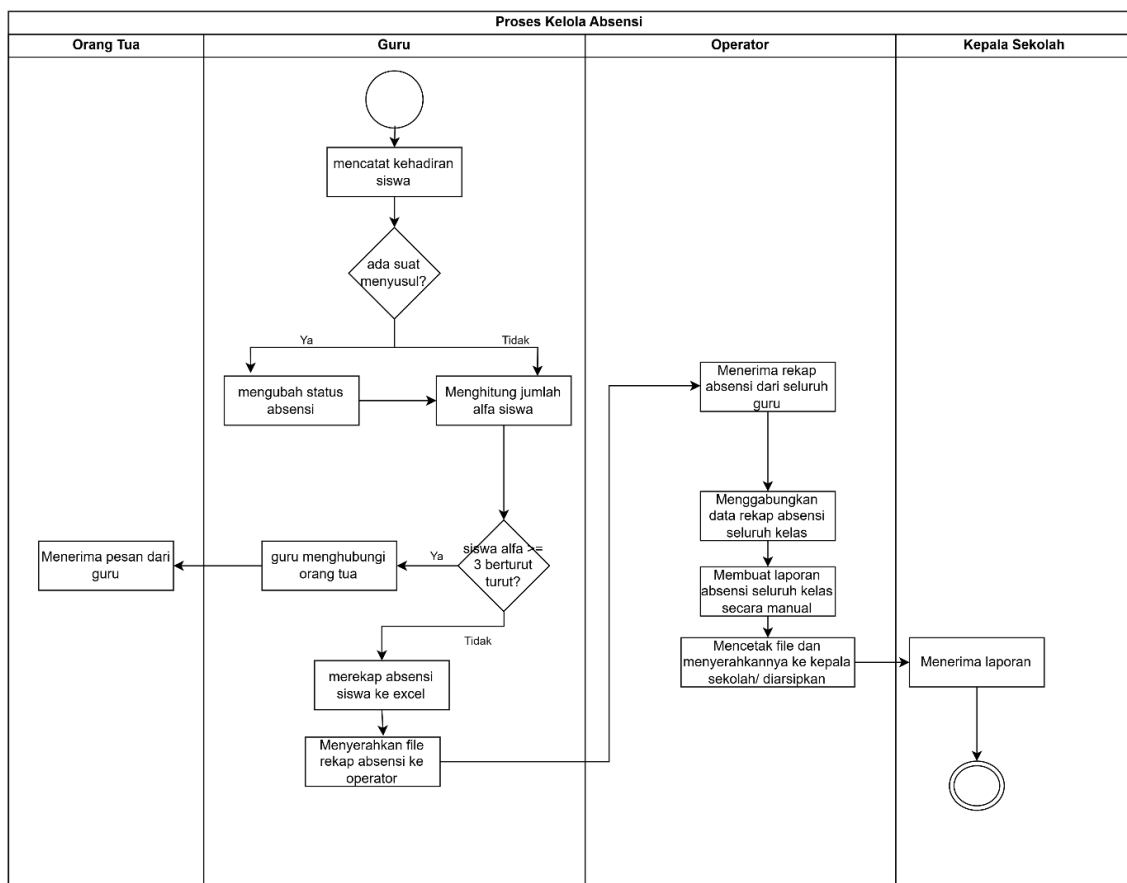
Gambar 4.1 menggambarkan alur persiapan data guru dan siswa oleh Operator dan Guru sebelum sistem digunakan, Operator menyiapkan dan menyerahkan data, lalu Guru mencatatnya secara manual ke buku absensi.

Tabel 4.1 Analisis Persiapan Data Master

No	Aktor	Aktivitas	Data/Informasi	Permasalahan	Solusi
1	Operator	Menyiapkan data guru, kelas, dan siswa dalam periode aktif	file data siswa, guru, data kelas dalam periode aktif	Data tersimpan pada banyak dokumen sehingga sulit dicari dan diperbarui ketika diperlukan	Sistem menyediakan fitur pengelolaan data guru, siswa, dan kelas secara terpusat
2	Operator	Menyusun data guru, kelas, siswa dalam periode aktif	file kelas dan siswa yang diampu guru dalam periode aktif	Risiko kesalahan penempatan siswa atau guru pengampu saat proses penyusunan data	Sistem menyusun dan menampilkan data kelas serta siswa sesuai guru pengampu

No	Aktor	Aktivitas	Data/Informasi	Permasalahan	Solusi
3	Operator	Menyerahkan data kelas dan siswa kepada guru pengampu	fike kelas dan siswa yang diampu guru periode aktif	Penyampaian data kepada guru membutuhkan waktu ketika terjadi perubahan atau pembaruan data	Sistem menampilkan data kelas dan siswa langsung pada akun guru pengampu
4	Guru	menerima data kelas dan siswa yang diampu dalam periode aktif	fike kelas dan siswa yang diampu guru periode aktif	Guru kesulitan memperoleh informasi kelas dan siswa secara cepat ketika dibutuhkan	guru dapat langsung melihat data siswa melalui sistem
5	Guru	mencatat nama siswa dalam buku absensi periode aktif	Data siswa dalam periode aktif	Pencarian dan pencocokan data siswa memerlukan waktu lebih lama saat proses absensi berlangsung	Sistem menyediakan daftar siswa digital sehingga guru tidak perlu mencatat nama siswa secara berulang

Tabel 4.1 merangkum analisis aktivitas kelola data guru dan siswa (aktor, aktivitas, data, informasi, masalah, solusi) untuk menentukan solusi pada sistem yang dikembangkan



Gambar 4.2 Activity Diagram Proses Absensi Harian & Rekap Laporan

Gambar 4.2 menggambarkan alur absensi harian dan rekap laporan yang sedang berjalan, dari pencatatan manual hingga penggabungan laporan oleh Operator, sebagai acuan penentuan solusi sistem.

Tabel 4.2 Analisis Swimlane Proses Absensi Harian & Rekap Laporan

No	Aktor	Aktivitas	Permasalahan	Solusi
1	Guru	Mencatat kehadiran siswa	Pernah terjadi kesalahan pencatatan status kehadiran siswa.	pencatatan absensi melalui sistem dan notifikasi WhatsApp dikirim otomatis.
2	Guru	Mengubah status absensi	Perubahan status dilakukan ketika terdapat surat izin atau surat sakit yang menyusul.	Sistem menyediakan fitur koreksi absensi.
3	Guru	Menghitung jumlah Alfa siswa	Pemantauan ketidakhadiran masih dilakukan secara manual.	Sistem menghitung rekap kehadiran secara otomatis.
4	Guru	Menghubungi orang tua jika Alfa ≥ 3	Pengiriman WhatsApp masih dilakukan manual dan dapat terlambat atau terlupa.	Sistem mengirim WhatsApp otomatis.
5	Orang Tua	Menerima pesan dari guru	Informasi ketidakhadiran belum dikirim secara otomatis.	Sistem mengirim WhatsApp otomatis.
6	Guru	Merekap absensi siswa ke Excel	Rekap dilakukan manual sehingga membutuhkan waktu dan berisiko salah input.	Sistem membuat rekap absensi secara otomatis.
7	Guru	Menyerahkan file rekap absensi kepada Operator	Laporan masih diserahkan secara manual kepada Operator.	Data rekap tersimpan secara terpusat dalam sistem.
8	Operator	Menerima rekap absensi seluruh guru	Pengumpulan laporan membutuhkan waktu karena keterlambatan guru.	Sistem menyediakan rekap absensi secara terpusat.
9	Operator	Menggabungkan data rekap absensi seluruh kelas	Operator masih menggabungkan data secara manual.	Sistem menyediakan rekap seluruh kelas.
10	Operator	Membuat laporan absensi seluruh kelas	Penyusunan laporan masih dilakukan secara manual.	Sistem menghasilkan laporan rekap absensi.
11	Operator	Mencetak dan menyerahkan laporan	Kepala Sekolah harus menunggu laporan dari Operator.	Laporan dapat diakses dan dicetak melalui sistem.
12	Kepala Sekolah	Menerima laporan	Keterlambatan penerimaan laporan	Kepala Sekolah dapat melihat rekap absensi melalui sistem.

Tabel 4.2 merangkum analisis aktivitas sebagai dasar solusi pada sistem yang dikembangkan.

4.1.4 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan untuk mengidentifikasi kebutuhan yang diperlukan dalam pembangunan sistem absensi siswa. Kebutuhan sistem dalam penelitian ini dibagi menjadi dua bagian utama, yaitu kebutuhan pengguna (User requirement) dan kebutuhan sistem (functional requirement dan non-functional requirement).

Kebutuhan pengguna menggambarkan kebutuhan dari sudut pandang pengguna sistem, sedangkan kebutuhan sistem merupakan turunan dari kebutuhan pengguna yang telah diterjemahkan ke dalam bentuk teknis yang dapat diimplementasikan.

4.1.4.1 User story

Kebutuhan pihak orang tua siswa pada Tabel 4.3 diturunkan secara tidak langsung dari hasil wawancara dengan operator dan guru serta observasi proses berjalan, karena orang tua siswa bukan responden penelitian ini.

Tabel 4.3 User story

Kode	User Story
US-01	Sebagai Operator, saya ingin mengelola data guru agar data guru tersimpan dengan baik.
US-02	Sebagai Operator dan guru, saya ingin mengelola data siswa agar data siswa terorganisir dengan baik.
US-03	Sebagai Operator, saya ingin mengelola data kelas agar data kelas terorganisir dengan baik.
US-04	Sebagai Operator, saya ingin mengelola data periode tahun ajaran agar sistem dapat menentukan batas waktu absensi, mengatur hari libur.
US-05	Sebagai Operator, Kepala sekolah dan Guru, saya ingin melihat data rekap absensi berdasarkan filter tanggal agar dapat memantau kehadiran siswa secara spesifik.
US-06	Sebagai Operator, kepala sekolah, dan Guru, saya ingin mengekspor laporan rekap absensi berdasarkan filter tanggal agar laporan dapat disimpan dan dicetak sesuai kebutuhan.
US-07	Sebagai Operator dan Guru, saya ingin melihat riwayat pengiriman pesan agar dapat mengetahui status pengiriman pesan.
US-08	Sebagai Guru, saya ingin mencatat dan mengoreksi data absensi siswa agar data kehadiran akurat dan sistem dapat mengirim pesan otomatis ke orang tua saat status Alpa, Izin, atau Sakit.

Tabel 4.3 memuat User Story hasil analisis proses bisnis, menjadi dasar perancangan fitur sistem.

4.1.4.2 Analisis Aktor

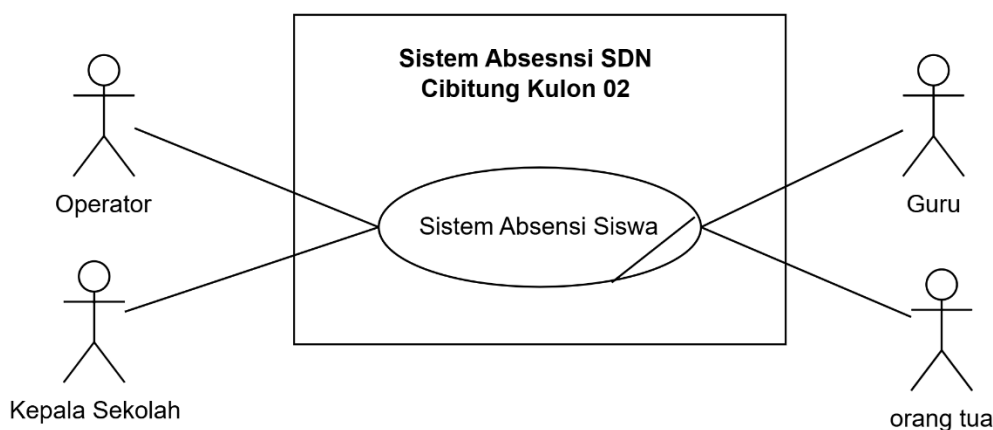
Tabel 4.4 Analisis Aktor

ID	Aktor	Deskripsi
AC-01	Operator	Operator adalah pengguna yang bertanggung jawab terhadap pengelolaan data master sistem, yaitu data guru, data siswa, data kelas dan data periode. Operator dapat memantau rekap absensi seluruh kelas, mengekspor laporan rekap, serta melihat riwayat pengiriman <i>Pesan WhatsApp</i> kepada orang tua siswa .
AC-02	Guru	Guru adalah pengguna yang bertugas mencatat kehadiran siswa pada kelas yang diampu. Setiap kali absensi disimpan dan terdapat siswa dengan status Alpa, Izin, atau Sakit, sistem secara otomatis mengirimkan <i>Pesan WhatsApp</i> kepada orang tua siswa, sehingga apabila terjadi kesalahan input status kehadiran oleh guru, orang tua siswa dapat segera mengetahui dan mengonfirmasikannya. Guru juga dapat memantau rekap absensi kelas yang diampu, mengekspor laporan rekap, serta melihat riwayat pengiriman <i>Pesan</i> .
AC-03	Orang tua siswa	Aktor eksternal yang menerima pesan <i>WhatsApp</i> otomatis dari sistem ketika siswa tercatat atau dikoreksi menjadi Alpa, Izin, atau Sakit, maupun ketika status tersebut dikoreksi kembali menjadi Hadir. Orang tua siswa tidak memiliki akses login ke sistem, namun berperan sebagai pihak yang memverifikasi keakuratan data kehadiran secara real-time, sehingga kesalahan input yang dilakukan guru dapat segera diketahui dan dikonfirmasi ke pihak sekolah.

ID	Aktor	Deskripsi
AC-04	Kepala sekolah	Kepala sekolah adalah pengguna yang bisa mengakses rekap semua kelas dan melakukan cetak data rekap berdasarkan filter yang dipilih

Tabel 4.4 memuat aktor sistem beserta peran dan hak aksesnya.

4.1.4.3 Diagram Use Case Bisnis



Gambar 4.3 Diagram Use Case Bisnis

Gambar 4.3 menggambarkan interaksi Operator, Kepala Sekolah, Orang Tua Siswa, dan Guru dengan sistem pada level bisnis.

4.1.4.4 Analisis Kebutuhan Pengguna

Tabel 4.5 Analisis Kebutuhan Pengguna

Kode	User Requirement
UR-01	User dapat login ke sistem sesuai dengan role yang dimiliki.
UR-02	User dapat mengelola data guru.
UR-03	User dapat mengelola data siswa.
UR-04	User dapat mengelola data kelas.
UR-05	User dapat mengelola data periode tahun ajaran.
UR-06	User dapat melihat data rekap absensi berdasarkan filter tanggal.
UR-07	User dapat mengekspor laporan rekap absensi berdasarkan filter tanggal.
UR-08	User dapat melihat riwayat pengiriman pesan WhatsApp.
UR-09	User dapat menyimpan dan mengubah absensi siswa, dan sistem dapat mengirim pesan otomatis ketika status tersebut dicatat atau dikoreksi menjadi Alpa, Izin, atau Sakit, maupun dikoreksi kembali menjadi Hadir, sehingga kesalahan input guru dapat diverifikasi oleh orang tua siswa

Tabel 4.5 memuat kebutuhan pengguna (User Requirement) sebagai dasar pengembangan sistem.

4.1.4.5 Analisis Kebutuhan Fungsional

Tabel 4.6 Kebutuhan Fungsional Sistem

Kode UR	User Requirement	Kode FR	Functional Requirement
UR-01	User dapat login ke sistem sesuai dengan role yang dimiliki.	FR-01	Sistem menyediakan fitur login menggunakan username/NIP dan password.
		FR-02	Sistem membedakan hak akses berdasarkan role pengguna (Operator, Guru, dan Kepala Sekolah).
		FR-03	Sistem mengarahkan pengguna ke dashboard sesuai role setelah login berhasil.
		FR-04	Sistem menampilkan pesan kesalahan apabila login gagal.
		FR-05	Sistem menyediakan fitur lupa password yang mengirim link perubahan password melalui WhatsApp.
UR-02	User dapat mengelola data guru.	FR-06	Sistem menyediakan fitur tambah data guru.
		FR-07	Sistem menyediakan fitur ubah data guru.
		FR-08	Sistem menyediakan fitur hapus data guru.
		FR-09	Sistem menyediakan fitur melihat data guru.
		FR-10	Sistem mencatat aktivitas user ke dalam log.
UR-03	User dapat mengelola data siswa.	FR-11	Sistem menyediakan fitur tambah data siswa.
		FR-12	Sistem menyediakan fitur ubah data siswa.
		FR-13	Sistem menyediakan fitur hapus data siswa.
		FR-14	Sistem menyediakan fitur ubah kelas untuk seluruh siswa.
		FR-15	Sistem menyediakan fitur melihat data siswa.
		FR-16	Sistem mencatat aktivitas user ke dalam log.
UR-04	User dapat mengelola data kelas.	FR-17	Sistem menyediakan fitur tambah data kelas.
		FR-18	Sistem menyediakan fitur ubah data kelas.
		FR-19	Sistem menyediakan fitur hapus data kelas.
		FR-20	Sistem menyediakan fitur melihat data kelas.
UR-05	User dapat mengelola data periode tahun ajaran.	FR-21	Sistem menyediakan fitur menentukan periode tahun ajaran aktif.
		FR-22	Sistem menyediakan fitur ubah periode tahun ajaran.
		FR-23	Sistem menyediakan fitur reset periode untuk mengatur ulang hari aktif absensi dan rekap absensi.
		FR-24	Sistem menyediakan fitur pengelolaan hari libur sekolah.
		FR-25	Sistem membatasi pencatatan absensi hanya pada tanggal mulai hingga tanggal berakhir periode aktif.
		FR-26	Sistem mencegah pencatatan absensi pada hari libur.
		FR-27	Sistem mencatat aktivitas user ke dalam log.
UR-06	User dapat melihat data rekap absensi berdasarkan filter tanggal.	FR-28	Sistem menampilkan data rekap absensi siswa.
		FR-29	Sistem menyediakan filter rekap per hari, per bulan, dan per semester serta input tanggal manual untuk rekap absensi.
		FR-30	Sistem menyediakan filter kelas pada rekap absensi.
		FR-31	Sistem menampilkan jumlah hadir, izin, sakit, dan alpa beserta persentase total absensi.
UR-07	User dapat mengeksport laporan rekap absensi berdasarkan filter tanggal.	FR-32	Sistem menyediakan fitur cetak laporan rekap absensi.
		FR-33	Sistem menghasilkan laporan sesuai filter yang dipilih.
		FR-34	Sistem menghasilkan laporan dalam format Excel.
UR-08	User dapat melihat riwayat pengiriman pesan WhatsApp.	FR-35	Sistem menyimpan riwayat pengiriman pesan WhatsApp.
		FR-36	Sistem menampilkan tanggal dan waktu pengiriman pesan.
		FR-37	Sistem menampilkan status pengiriman pesan.
		FR-38	Sistem menyediakan pencarian riwayat pesan.

Kode UR	User Requirement	Kode FR	Functional Requirement
UR-09	User dapat menyimpan dan mengubah absensi siswa, dan sistem dapat mengirim pesan otomatis ketika status tersebut dicatat atau dikoreksi menjadi Alpa, Izin, atau Sakit, maupun dikoreksi kembali menjadi Hadir, sehingga kesalahan input guru dapat diverifikasi oleh orang tua siswa.	FR-39	Sistem menyediakan fitur pencatatan absensi siswa.
		FR-40	Sistem menyediakan pilihan status Hadir, Izin, Sakit, dan Alpa.
		FR-41	Sistem mengirim pesan WhatsApp secara otomatis kepada orang tua siswa setiap kali status kehadiran dicatat atau dikoreksi menjadi Alpa, Izin, atau Sakit, maupun ketika status tersebut dikoreksi kembali menjadi Hadir, sebagai mekanisme verifikasi terhadap kemungkinan kesalahan input yang dilakukan guru.
		FR-42	Sistem menampilkan informasi nama siswa, kelas, tanggal, dan status kehadiran pada pesan WhatsApp.
		FR-43	Sistem mencatat status pengiriman pesan WhatsApp.
		FR-44	Sistem menyediakan fitur koreksi data absensi siswa.
		FR-45	Sistem membatasi pengisian absensi hanya pada hari aktif dan tidak dapat dilakukan untuk hari berikutnya.

Tabel 4.6 memuat kebutuhan fungsional hasil analisis kebutuhan pengguna, sebagai dasar penentuan fitur utama sistem.

4.1.4.6 Analisis Kebutuhan Non Fungsional

Tabel 4.7 Kebutuhan Non-Fungsional Sistem

Kode	Non-Functional Requirement	Deskripsi
NFR-01	Keamanan Sistem	Sistem harus memiliki login dan pembatasan hak akses berbasis role (Operator/Guru /Kepala sekolah). Guru hanya dapat mengakses data kelas yang diampu. Sistem dilindungi dari serangan CSRF, XSS, dan SQL Injection menggunakan mekanisme bawaan Laravel.
NFR-02	Kemudahan Penggunaan	Antarmuka harus sederhana dan mudah digunakan.
NFR-03	Responsif	Sistem dapat diakses melalui smartphone, laptop, maupun desktop menggunakan browser tanpa instalasi aplikasi tambahan.
NFR-04	Kinerja	Sistem mampu memproses penyimpanan data absensi dan trigger Pesan WhatsApp secara cepat setelah guru menekan tombol simpan absensi.
NFR-05	Integrasi WhatsApp Gateway	Sistem terhubung dengan layanan API Fonnte untuk mendukung pengiriman Pesan WhatsApp otomatis kepada orang tua siswa tanpa intervensi manual, dengan mempertimbangkan batas kuota pengiriman pesan pada tier gratis (1.000 pesan/bulan).

Kode	Non-Functional Requirement	Deskripsi
NFR-06	Integritas Data	Sistem menyimpan data absensi secara terstruktur dan konsisten pada basis data MySQL, sehingga risiko kehilangan atau ketidaksesuaian data akibat lembar absensi fisik yang tercecer maupun rusaknya file Excel, sebagaimana terjadi pada proses pencatatan manual, dapat diminimalkan. Data yang ditampilkan sistem kepada pengguna, baik berupa rekap, riwayat pengiriman pesan, maupun status kehadiran, harus selalu akurat dan konsisten dengan data yang diinput, sehingga integritas data kehadiran siswa tetap terjaga dalam jangka panjang.

Tabel 4.7 memuat kebutuhan non-fungsional terkait keamanan, kinerja, responsivitas, dan kemudahan penggunaan sistem.

4.1.4.7 Identifikasi Use Case

Tabel 4.8 Identifikasi Use Case

Kode FR	Functional Requirement	Kode UC	Nama Use Case	Aktor
FR-01	Sistem menyediakan fitur login menggunakan username/NIP dan password.	-	Login Sistem	Operator, Guru, Kepala Sekolah
FR-02	Sistem membedakan hak akses berdasarkan role pengguna (Operator, Guru, dan Kepala Sekolah).	-	Login Sistem	Operator, Guru, Kepala Sekolah
FR-03	Sistem mengarahkan pengguna ke dashboard sesuai role setelah login berhasil.	-	Login Sistem	Operator, Guru, Kepala Sekolah
FR-04	Sistem menampilkan pesan kesalahan apabila login gagal.	-	Login Sistem	Operator, Guru, Kepala Sekolah
FR-05	Sistem menyediakan fitur lupa password yang mengirim link perubahan password melalui WhatsApp.	-	Login Sistem	Operator, Guru, Kepala Sekolah
FR-06	Sistem menyediakan fitur tambah data guru.	UC-01	Kelola Data Guru	Operator
FR-07	Sistem menyediakan fitur ubah data guru.	UC-01	Kelola Data Guru	Operator
FR-08	Sistem menyediakan fitur hapus data guru.	UC-01	Kelola Data Guru	Operator
FR-09	Sistem menyediakan fitur melihat data guru.	UC-01	Kelola Data Guru	Operator
FR-10	Sistem mencatat aktivitas user ke dalam log.	UC-01	Kelola Data Guru	Operator
FR-11	Sistem menyediakan fitur tambah data siswa.	UC-02	Kelola Data Siswa	Operator, Guru
FR-12	Sistem menyediakan fitur ubah data siswa.	UC-02	Kelola Data Siswa	Operator, Guru
FR-13	Sistem menyediakan fitur hapus data siswa.	UC-02	Kelola Data Siswa	Operator, Guru
FR-14	Sistem menyediakan fitur ubah kelas untuk seluruh siswa.	UC-02	Kelola Data Siswa	Operator
FR-15	Sistem menyediakan fitur melihat data siswa.	UC-02	Kelola Data Siswa	Operator, Guru
FR-16	Sistem mencatat aktivitas user ke dalam log.	UC-02	Kelola Data Siswa	Operator
FR-17	Sistem menyediakan fitur tambah data kelas.	UC-03	Kelola Data Kelas	Operator
FR-18	Sistem menyediakan fitur ubah data kelas.	UC-03	Kelola Data Kelas	Operator
FR-19	Sistem menyediakan fitur hapus data kelas.	UC-03	Kelola Data Kelas	Operator
FR-20	Sistem menyediakan fitur melihat data kelas.	UC-03	Kelola Data Kelas	Operator
FR-21	Sistem menyediakan fitur menentukan periode tahun ajaran aktif.	UC-04	Kelola Periode	Operator
FR-22	Sistem menyediakan fitur ubah periode tahun ajaran.	UC-04	Kelola Periode	Operator
FR-23	Sistem menyediakan fitur reset periode untuk mengatur ulang hari aktif absensi dan rekap absensi.	UC-04	Kelola Periode	Operator
FR-24	Sistem menyediakan fitur pengelolaan hari libur sekolah.	UC-04	Kelola Periode	Operator
FR-25	Sistem membatasi pencatatan absensi hanya pada tanggal mulai hingga tanggal berakhir periode aktif.	UC-04	Kelola Periode	Operator

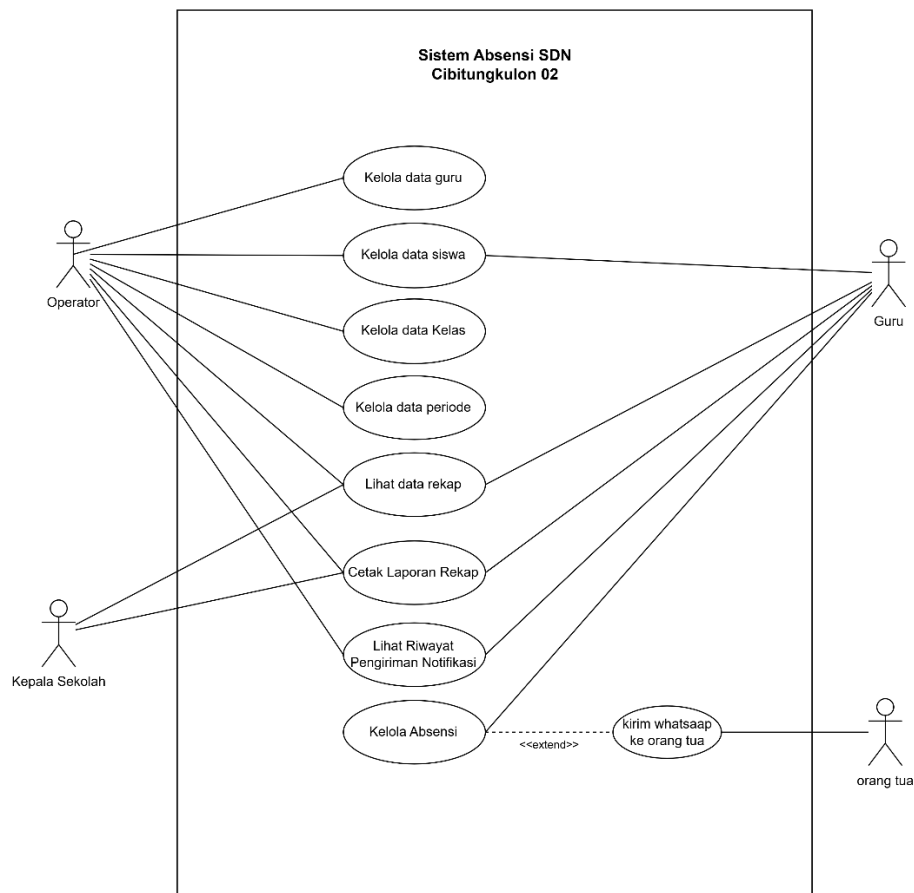
Kode FR	Functional Requirement	Kode UC	Nama Use Case	Aktor
FR-26	Sistem mencegah pencatatan absensi pada hari libur.	UC-04	Kelola Periode	Operator
FR-27	Sistem mencatat aktivitas user ke dalam log.	UC-04	Kelola Periode	Operator
FR-28	Sistem menampilkan data rekap absensi siswa.	UC-05	Lihat Data Rekap	Operator, Guru, Kepala Sekolah
FR-29	Sistem menyediakan filter rekap per hari, per bulan, dan per semester serta input tanggal manual untuk rekap absensi.	UC-05	Lihat Data Rekap	Operator, Guru, Kepala Sekolah
FR-30	Sistem menyediakan filter kelas pada rekap absensi.	UC-05	Lihat Data Rekap	Operator, Guru, Kepala Sekolah
FR-31	Sistem menampilkan jumlah hadir, izin, sakit, dan alpa beserta persentase total absensi.	UC-05	Lihat Data Rekap	Operator, Guru, Kepala Sekolah
FR-32	Sistem menyediakan fitur cetak laporan rekap absensi.	UC-06	Cetak Laporan Rekap	Operator, Guru, Kepala Sekolah
FR-33	Sistem menghasilkan laporan sesuai filter yang dipilih.	UC-06	Cetak Laporan Rekap	Operator, Guru, Kepala Sekolah
FR-34	Sistem menghasilkan laporan dalam format Excel.	UC-06	Cetak Laporan Rekap	Operator, Guru, Kepala Sekolah
FR-35	Sistem menyimpan riwayat pengiriman pesan WhatsApp.	UC-07	Lihat Riwayat Pengiriman Pesan	Operator, Guru
FR-36	Sistem menampilkan tanggal dan waktu pengiriman pesan.	UC-07	Lihat Riwayat Pengiriman Pesan	Operator, Guru
FR-37	Sistem menampilkan status pengiriman pesan.	UC-07	Lihat Riwayat Pengiriman Pesan	Operator, Guru
FR-38	Sistem menyediakan pencarian riwayat pesan.	UC-07	Lihat Riwayat Pengiriman Pesan	Operator, Guru
FR-39	Sistem menyediakan fitur pencatatan absensi siswa.	UC-08	Kelola Absensi	Guru
FR-40	Sistem menyediakan pilihan status Hadir, Izin, Sakit, dan Alpa.	UC-08	Kelola Absensi	Guru
FR-41	Sistem mengirim pesan WhatsApp secara otomatis kepada orang tua siswa setiap kali status kehadiran dicatat atau dikoreksi menjadi Alpa, Izin, atau Sakit, maupun ketika status tersebut dikoreksi kembali menjadi Hadir, sebagai mekanisme verifikasi terhadap kemungkinan kesalahan input yang dilakukan guru.	UC-08	Kelola Absensi	Guru, Orang tua
FR-42	Sistem menampilkan informasi nama siswa, kelas, tanggal, dan status kehadiran pada pesan WhatsApp.	UC-08	Kelola Absensi	Guru
FR-43	Sistem mencatat status pengiriman pesan WhatsApp.	UC-08	Kelola Absensi	Guru
FR-44	Sistem menyediakan fitur koreksi data absensi siswa.	UC-08	Kelola Absensi	Guru
FR-45	Sistem membatasi pengisian absensi hanya pada hari aktif dan tidak dapat dilakukan untuk hari berikutnya.	UC-08	Kelola Absensi	Guru

Tabel 4.8 memetakan software requirement terhadap Use Case yang merepresentasikan fungsi sistem.

4.2 Perancangan

Pada tahap ini dilakukan proses perancangan terhadap sistem usulan berdasarkan hasil analisis kebutuhan yang telah dilakukan sebelumnya. Perancangan sistem ini bertujuan untuk memberikan gambaran mengenai alur kerja sistem serta interaksi pengguna dengan sistem dalam pembangunan Website Absensi Siswa dengan WhatsApp Gateway pada SDN Cibitungkulon 02.

4.2.1 Use Case Diagram



Gambar 4.4 Use Case Diagram Sistem Absensi SDN Cibitungkulon 02

Gambar 4.4 menggambarkan Use Case diagram sistem, mencakup fungsi teknis pengelolaan data guru, siswa, kelas, periode, rekap absensi, cetak laporan, riwayat Pesan, dan pengelolaan absensi.

Tabel 4.9 Definisi Use Case

KODE UC	Nama Use Case	Deskripsi
UC-01	Kelola Data Guru	Digunakan oleh Operator untuk menambah, mengubah, menghapus, dan melihat data guru yang terdaftar pada sistem.
UC-02	Kelola Data Siswa	Digunakan oleh Operator untuk menambah, mengubah, menghapus, dan melihat data siswa beserta data orang tua siswa dan nomor WhatsApp yang digunakan untuk pengiriman pesan.
UC-03	Kelola Data Kelas	Digunakan oleh Operator untuk menambah, mengubah, menghapus, dan melihat data kelas yang digunakan dalam proses pengelolaan absensi siswa.
UC-04	Kelola Periode Tahun Ajaran	Digunakan oleh Operator untuk mengelola periode tahun ajaran, termasuk menambah, mengubah, mereset periode, dan mengelola hari libur.
UC-05	Lihat Data Rekap	Digunakan oleh Operator, Kepala Sekolah, dan Guru untuk melihat detail data rekap absensi siswa yang tersimpan pada sistem.
UC-06	Cetak Laporan Rekap	Digunakan oleh Operator, Kepala Sekolah, dan Guru untuk mengeksport laporan rekap absensi siswa ke dalam format Excel berdasarkan periode yang dipilih.
UC-07	Lihat Riwayat Pengiriman Pesan	Digunakan oleh Operator dan Guru untuk melihat riwayat pengiriman pesan WhatsApp beserta status keberhasilan atau kegagalan pengiriman.

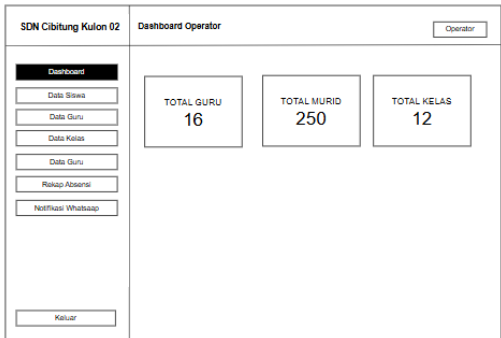
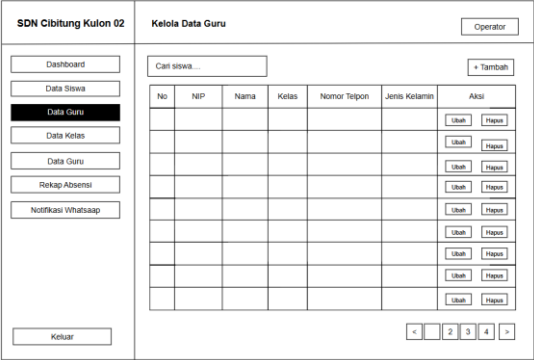
KODE UC	Nama Use Case	Deskripsi
UC-08	Kelola Absensi	Digunakan oleh Guru untuk mencatat kehadiran siswa dengan memilih status Hadir, Izin, Sakit, atau Alfa, kemudian menyimpan data absensi ke dalam sistem. Use case ini memiliki hubungan <<extend>> dengan Kirim WhatsApp ke Orang Tua, yaitu sistem mengirim notifikasi WhatsApp apabila status tercatat atau dikoreksi menjadi Izin, Sakit, atau Alfa, sedangkan status Hadir tidak mengirimkan notifikasi. Guru juga dapat mengoreksi data absensi yang telah tersimpan apabila terjadi kesalahan pencatatan sesuai batasan yang ditetapkan.

Tabel 4.9 menjelaskan fungsi dan tujuan setiap Use Case pada sistem absensi siswa.

4.2.2 Skenario Use Case

Skenario Use Case menjelaskan secara rinci alur interaksi antara aktor dan sistem untuk setiap Use Case yang telah diidentifikasi. Setiap skenario mencakup kondisi awal (Kondisi Awal), kondisi akhir (Kondisi Akhir), alur normal, serta alur alternatif yang menangani kondisi tidak biasa atau kesalahan.

Tabel 4.10 Skenario UC-01: Kelola Data Guru

Identifikasi	
Use Case ID	UC-01
Nama Use Case	Kelola Data Guru
Aktor	Operator
Tujuan	Memungkinkan Operator mengelola data
Kondisi Awal	Operator berhasil login ke sistem
 Masuk halaman dashboard  Menekan Tombol Masuk Data Guru	
Kondisi Akhir	Data guru berhasil ditambahkan, diubah, atau dihapus sesuai operasi yang dilakukan

Identifikasi	
 Masuk halaman tambah data ketika <i>Operator</i> menekan tombol tambah	 Masuk halaman ubah data ketika <i>Operator</i> menekan tombol Ubah
 Menekan tombol hapus untuk menghapus data guru	
Skenario Utama	
Aksi Aktor	Reaksi Sistem
1. Operator membuka menu Kelola Data Guru.	2. Sistem menampilkan daftar guru dalam tabel beserta tombol Tambah Guru, Ubah, dan Hapus
3. Operator klik tombol 'Tambah Guru'.	4. Sistem menampilkan formulir: Nama Lengkap, NIP, No. Telepon, Jenis Kelamin, Kelas yang Diampu (<i>checkbox</i> multi-pilih), Username, password, dan role.
5. Operator mengisi seluruh field dan klik 'Simpan'.	6. Sistem memvalidasi <i>Input</i>
	7. Sistem menyimpan data guru dan log ke database. Menampilkan pesan sukses dan mengembalikan <i>Operator</i> ke halaman kelola data guru
8. Operator klik tombol 'Ubah'.	9. Sistem menampilkan formulir yang sudah di isi: Nama Lengkap, NIP, No. Telepon, Jenis Kelamin, Kelas yang Diampu (<i>checkbox</i> multi-pilih), Username, password, dan role.
10. Operator mengisi data yang ingin di ubah dan menekan tombol 'Simpan'.	11. Sistem memvalidasi <i>Input</i>
	12. Sistem mengubah data guru dan log di database. Menampilkan pesan sukses dan mengembalikan <i>Operator</i> ke halaman kelola data guru

Identifikasi	
13. Operator memilih data yang ingin dihapus dan mengklik tombol 'Hapus'.	14. Sistem menampilkan <i>pop up</i> konfirmasi penghapusan data
15. Operator menekan tombol 'Hapus' pada <i>pop up</i>	16. Sistem menghapus data dari database
Skenario Alternatif	
Kondisi	Reaksi Sistem
Nama/ <i>Username</i> /NIP/No telpon guru sudah terdaftar.	Sistem menampilkan error: ' Nama/ <i>Username</i> /NIP/No telpon ini sudah digunakan.' Data tidak disimpan.
Field dengan tanda (*) tidak boleh dikosongkan.	Sistem menampilkan pesan validasi di bawah field yang kosong.
Operator input value yang berbeda pada kolom password dan konfirmasi password	Sistem menampilkan pesan validasi di bawah password dan konfirmasi password harus memiliki value yang sama
Operator mencoba menghapus guru yang memiliki riwayat absensi.	Sistem menampilkan pesan: 'Pengguna tidak dapat dihapus karena memiliki riwayat absensi.' Data tidak dihapus.

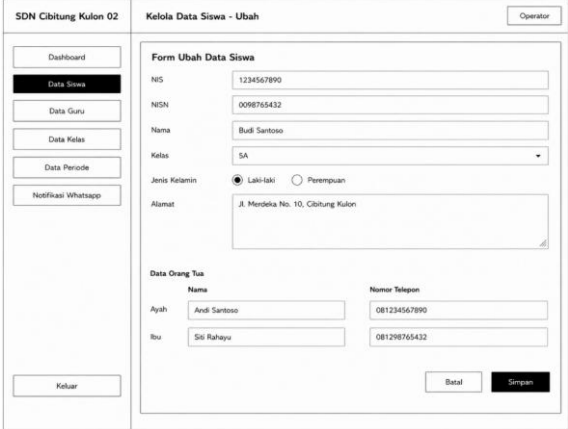
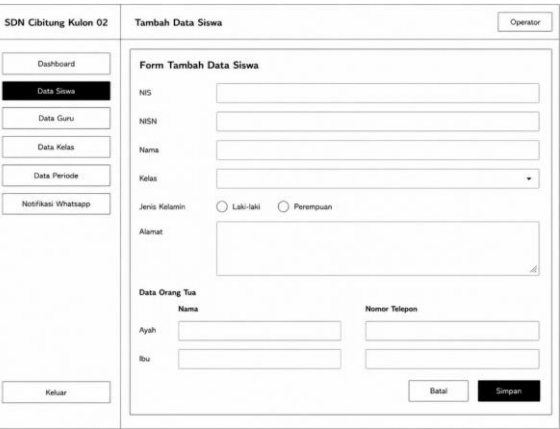
Tabel 4.10 menjelaskan skenario Use Case kelola data guru, yaitu alur interaksi Operator dengan sistem.

Tabel 4.11 Skenario UC-02: Kelola Data Siswa

Identifikasi	
Use Case ID	UC-02
Nama Use Case	Kelola Data Siswa
Aktor	Operator dan Guru
Tujuan	Memungkinkan Operator dan guru mengelola data siswa
Kondisi Awal	Operator/guru berhasil login ke sistem

Masuk halaman dashboard

Menekan tombol data siswa lalu akan diarahkan ke halaman data siswa

Identifikasi	
Kondisi Akhir	Data siswa berhasil ditambahkan, diubah, atau dihapus sesuai operasi yang dilakukan
 	
Menekan tombol ubah dan diarahkan ke halaman ubah	Menekan tombol tambah dan diarahkan ke halaman tambah
	
Menekan tombol hapus untuk menghapus data siswa	
Skenario Utama	
Aksi Aktor	Reaksi Sistem
1. Operator atau guru membuka menu Kelola Data Siswa.	2. Sistem menampilkan daftar siswa dalam tabel beserta tombol Tambah Siswa, Ubah, dan Hapus.
3. Operator atau guru klik tombol 'Tambah Siswa'.	4. Sistem menampilkan formulir: periode, Nama Lengkap, NISN, Kelas, jenis kelamin, alamat, nama ayah, nama ibu, nomor telepon ayah, nomor ibu, nama wali dan no telepon wali
5. Operator atau guru mengisi seluruh field dan klik 'Simpan'.	6. Sistem memvalidasi <i>Input</i>
	7. Sistem menyimpan data ke tabel siswa dan tabel log, menampilkan pesan sukses, dan mengarahkan Operator atau Guru ke halaman kelola data siswa
8. Operator atau guru klik tombol 'ubah'.	9. Sistem menampilkan formulir yang sudah terisi: periode, Nama Lengkap, NISN, Kelas, jenis kelamin, alamat, nama ayah, nama ibu, nomor telepon ayah, nomor ibu, nama wali dan no telepon wali

Identifikasi	
10. Operator atau guru mengisi data yang ingin di ubah dan menekan tombol 'Simpan perubahan'.	11. Sistem memvalidasi <i>Input</i>
	12. Sistem mengubah data siswa dan menambah data log di database lalu Menampilkan pesan sukses dan mengembalikan <i>Operator atau Guru</i> ke halaman kelola data siswa
13. Operator atau guru memilih data yang ingin dihapus dan mengklik tombol 'Hapus'.	14. Sistem menampilkan <i>pop up</i> konfirmasi penghapusan data
15. Operator atau guru menekan tombol 'Hapus' pada <i>pop up</i>	16. Sistem menghapus data dari database
17. Operator menekan tombol 'Pindah Kelas'.	18. Sistem menampilkan form: kelas awal dan kelas tujuan
19. Operator mengisi form kelas awal dan kelas tujuan, lalu menekan tombol 'Simpan'.	20. Sistem memvalidasi input.
	21. Sistem menyimpan data kelas ke tabel siswa dan tabel log, menampilkan pesan sukses, dan mengarahkan Operator ke halaman kelola data siswa
Skenario Alternatif	
Kondisi	Reaksi Sistem
Operator atau guru input NISN/NIS>Nama sudah terdaftar.	Sistem menampilkan error: 'NISN/NIS>Nama ini sudah digunakan.' Data tidak disimpan.
Field dengan tanda (*) tidak boleh dikosongkan.	Sistem menampilkan pesan validasi di bawah field yang kosong.
Guru ingin mengakses kelas yang bukan kelasnya.	Sistem membatasi data kelas yang muncul pada filter hanya menampilkan kelasnya saja
Operator dan guru mengubah data kelas siswa yang memiliki data absensi	Sistem menampilkan pesan 'Tidak dapat mengubah data kelas siswa karena masih memiliki data absensi.'
Operator atau guru mencoba menghapus siswa yang memiliki data absensi.	Sistem menampilkan konfirmasi. Jika dikonfirmasi, data absensi terkait ikut dihapus.
Operator atau guru tidak mengisi satupun data nomor telpn orang tua secara lengkap	Sistem menampilkan pesan 'Minimal harus mengisi 1 nomor telepon orang tua.'
Operator pindah kelas dari kelas yang tidak ada siswa	Sistem menampilkan pesan 'Tidak ada siswa yang berada di kelas tersebut.'
Operator pindah kelas(seluruh siswa) yang sudah ada data absensi	Sistem menampilkan pesan 'Tidak dapat memindahkan siswa dari kelas A ke kelas B karena memiliki data absensi.'
Operator tambah siswa ke kelas yang sudah punya data absensi.	Sistem menyimpan data siswa tapi data rekap tetap 0 dan perlu input manual oleh gurunya di menu absensi

Tabel 4.11 menjelaskan skenario Use Case kelola data siswa, mencakup pengelolaan data siswa dan nomor WhatsApp orang tua.

Tabel 4.12 Skenario UC-03: Kelola Data Kelas

Identifikasi	
Use Case ID	UC-03
Nama Use Case	Kelola Data Kelas
Aktor	Operator
Tujuan	Memungkinkan Operator mengelola data kelas
Kondisi Awal	Operator berhasil login ke sistem

SDN Cibitung Kulon 02

Dashboard Operator

Operator

Dashboard

Data Siswa

Data Guru

Data Kelas

Data Guru

Rekap Absensi

Notifikasi Whatsapp

Keluar

TOTAL GURU

16

TOTAL MURID

250

TOTAL KELAS

12

SDN Cibitung Kulon 02

Kelola Data Kelas

Operator

Dashboard

Data Siswa

Data Guru

Data Kelas

Data Periode

Rekap Absensi

Notifikasi Whatsapp

Keluar

Data Kelas

Tambah

No	Nama Kelas	Wali Kelas	Aksi	
1	1A	Ahmad Ricky Pratama, S.Pd	Ubah	Hapus
2	1B	Siti Nurhaliza, S.Pd	Ubah	Hapus
3	2A	Dewi Lestari, S.Pd	Ubah	Hapus
4	2B	Budi Santoso, S.Pd	Ubah	Hapus
5	3A	Rina Agustina, S.Pd	Ubah	Hapus
6	3B	Fajar Nugroho, S.Pd	Ubah	Hapus
7	4A	Maya Sultri, S.Pd	Ubah	Hapus
8	5A	Andi Setiawan, S.Pd	Ubah	Hapus

<

1

2

3

4

>

Masuk halaman dashboard

Menekan tombol Data kelas dan diarahkan ke halaman data kelas

Kondisi Akhir	Data kelas berhasil ditambahkan, diubah, atau dihapus sesuai operasi yang dilakukan
---------------	---

SDN Cibitung Kulon 02

Kelola Data Kelas - Tambah

Operator

Dashboard

Data Siswa

Data Guru

Data Kelas

Data Periode

Rekap Absensi

Notifikasi Whatsapp

Keluar

Form Tambah Data Kelas

Nama Kelas

Wali Kelas

Batal

Simpan

SDN Cibitung Kulon 02

Kelola Data Kelas - Ubah

Operator

Dashboard

Data Siswa

Data Guru

Data Kelas

Data Periode

Rekap Absensi

Notifikasi Whatsapp

Keluar

Form Ubah Data Kelas

Nama Kelas

Wali Kelas

Batal

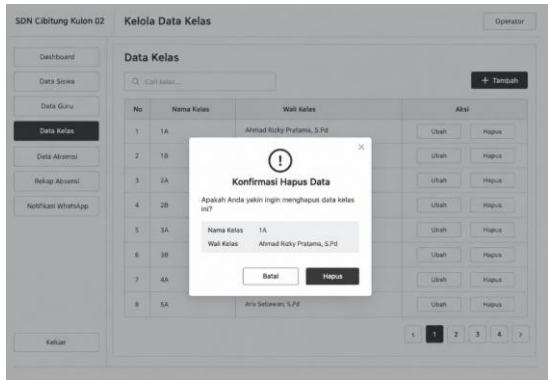
Simpan

Menekan tombol tambah dan diarahkan ke halaman tambah

Menekan tombol ubah dan diarahkan ke halaman

Masuk halaman dashboard

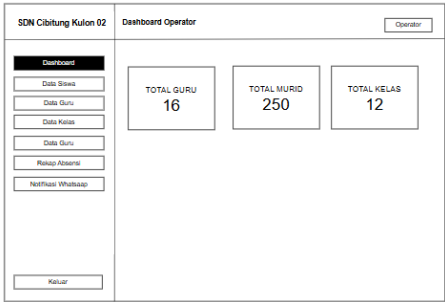
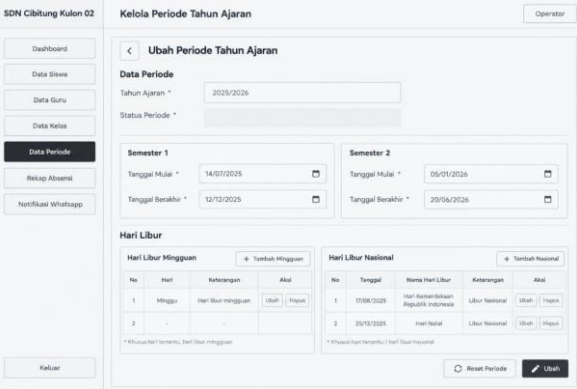
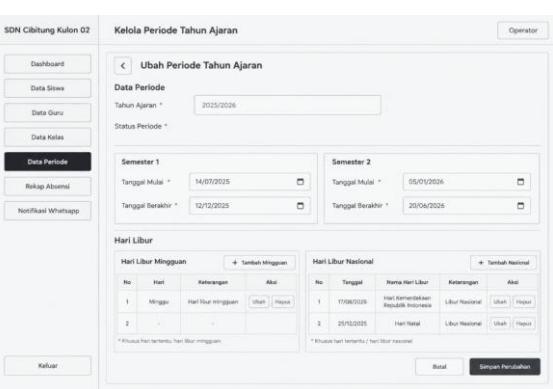
Menekan tombol Data kelas dan diarahkan ke halaman data kelas

Identifikasi	
 Menekan tombol hapus untuk menghapus data kelas	
Skenario Utama	
Aksi Aktor	Reaksi Sistem
1. Operator membuka menu Kelola Kelas	2. Sistem menampilkan daftar kelas dalam tabel beserta tombol Tambah Kelas, Ubah, dan Hapus.
3. Operator klik tombol Tambah Kelas.	4. Sistem menampilkan formulir: Nama Kelas dan wali kelas.
5. Operator mengisi seluruh field dan klik Simpan.	6. Sistem memvalidasi input.
	7. Sistem menyimpan data kelas dan log ke database. Menampilkan pesan sukses dan mengembalikan Operator ke halaman kelola data kelas.
8. Operator klik tombol Ubah pada baris kelas.	9. Sistem menampilkan formulir yang sudah terisi dengan data kelas yang dipilih.
10. Operator mengubah data yang diperlukan dan klik Simpan.	11. Sistem memvalidasi input dan menyimpan perubahan. Menampilkan pesan sukses.
	12. Sistem mengubah data kelas dan log di database. Menampilkan pesan sukses dan mengembalikan Operator ke halaman kelola data kelas
13. Operator memilih data yang ingin dihapus dan mengklik tombol 'Hapus'.	14. Sistem menampilkan <i>pop up</i> konfirmasi penghapusan data
15. Operator menekan tombol 'Hapus' pada <i>pop up</i>	16. Sistem menghapus data dari database
Skenario Alternatif	
Kondisi	Reaksi Sistem
Nama kelas sudah terdaftar.	Sistem menampilkan error: 'Nama kelas sudah terdaftar.' Data tidak disimpan.
Field dengan tanda (*) tidak boleh dikosongkan.	Sistem menampilkan pesan validasi di bawah field yang kosong.
Operator mencoba menghapus kelas yang memiliki data absensi aktif.	Sistem menampilkan pesan: 'Kelas ini memiliki data absensi aktif dan tidak dapat dihapus.'

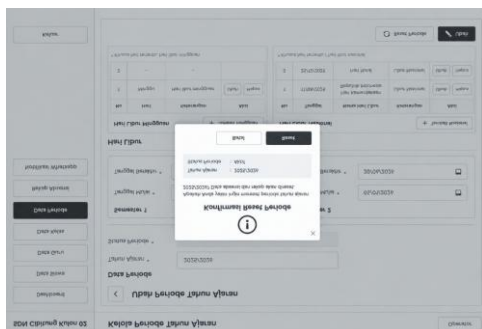
Identifikasi	
Operator mencoba menghapus kelas yang memiliki data siswa.	Sistem menampilkan pesan: 'Kelas ini memiliki data siswa dan tidak dapat dihapus.'
Operator membuka dropdown pilihan wali kelas pada formulir Tambah/Ubah Kelas.	Sistem hanya menampilkan guru yang saat ini belum menjadi wali kelas manapun, guru yang sudah menjadi wali kelas kelas lain tidak muncul dalam pilihan.

Tabel 4.12 menjelaskan skenario Use Case kelola data kelas pada sistem.

Tabel 4.13 Skenario UC-04: Kelola Data Periode

Identifikasi	
Use Case ID	UC-04
Nama Use Case	Kelola Data Periode
Aktor	Operator
Tujuan	Memungkinkan Operator mengelola periode tahun ajaran untuk mendukung proses absensi, pengaturan hari libur
Kondisi Awal	Operator berhasil login ke sistem
 Masuk halaman dashboard	
Kondisi Akhir	Data periode tahun ajaran berhasil diubah atau direset
 Halaman periode  Menekan tombol ubah dan diarahkan ke halaman ubah	

Identifikasi



Menekan tombol reset periode dan muncul pop up reset periode

Skenario Utama

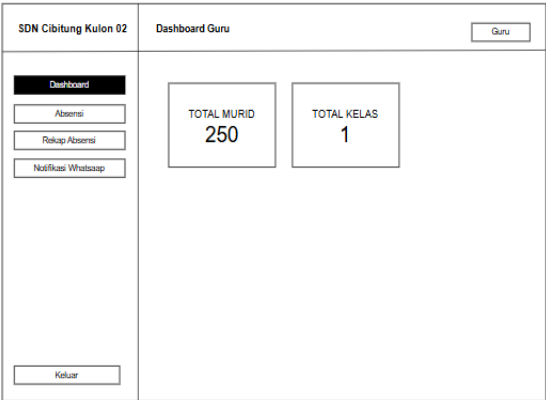
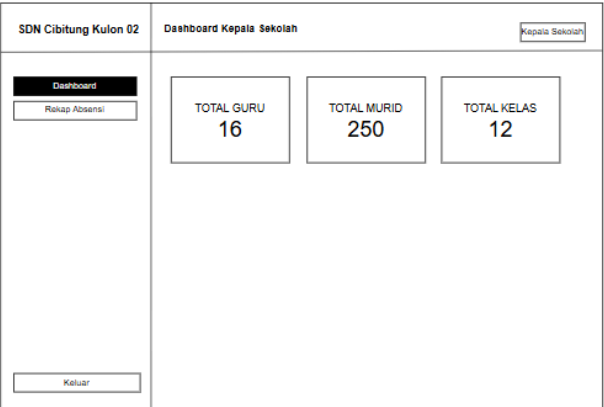
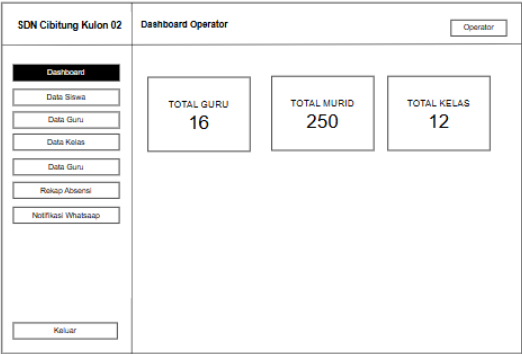
Aksi Aktor	Reaksi Sistem
1. Operator membuka menu Kelola Periode Tahun Ajaran.	2. Sistem menampilkan periode (Semester 1 & 2) beserta detail: tahun ajaran, tanggal mulai/selesai per semester, daftar hari libur mingguan & nasional.
3. Operator klik tombol 'Ubah' untuk masuk mode edit formulir.	4. Sistem menampilkan formulir periode dengan input enabled (tambah baru atau edit existing).
5. Operator mengisi tahun ajaran, tanggal mulai, tanggal selesai, hari libur mingguan dan nasional	6. Sistem melakukan validasi data input.
7. Operator klik tombol 'Simpan'.	8. Sistem menyimpan data periode tahun ajaran dan log ke database dan menampilkan pesan berhasil.
9. Operator klik tombol 'Reset Periode' lalu mengonfirmasi 'Ya, Reset'.	10. Sistem menghapus semua data tabel periodes dan absensis dalam satu transaksi, tampil pesan berhasil.

Skenario Alternatif

Kondisi	Reaksi Sistem
Tanggal mulai lebih besar dari tanggal selesai.	Sistem menampilkan pesan validasi dan meminta Operator memperbaiki data
Field dengan tanda (*) tidak boleh dikosongkan.	Sistem menampilkan pesan validasi di bawah field yang kosong.
Tanggal mulai semester 2 tidak boleh kurang dari sama dengan tanggal berakhir semester 1	Sistem menampilkan pesan validasi di bawah field tanggal.
Operator mengubah tanggal periode sehingga sebagian data absensi yang sudah ada berada di luar rentang tanggal baru	Sistem langsung menyimpan perubahan tanpa menampilkan konfirmasi, data absensi yang tanggalnya berada di luar rentang periode baru otomatis terhapus, sedangkan data absensi yang masih berada dalam rentang baru tetap tersimpan.
Operator mereset periode dengan tanggal yang belum diisi di semester 1 dan 2	Sistem tetap mereset periode

Tabel 4.13 menjelaskan skenario Use Case kelola data periode, meliputi pengelolaan periode tahun ajaran dan hari libur.

Tabel 4.14 Skenario UC-05: Melihat Data Rekap

Identifikasi	
Use Case ID	UC-05
Nama Use Case	Lihat Data Rekap
Aktor	Operator, Guru, Kepala Sekolah
Tujuan	Memungkinkan Operator , Kepala sekolah dan Guru melihat rekap data absensi
Kondisi Awal	Aktor berhasil login ke sistem
 Masuk halaman dashboard Guru  Masuk halaman dashboard Kepala sekolah  Masuk halaman dashboard	
Kondisi Akhir	Data rekap absensi ditampilkan sesuai filter dan lingkup akses aktor;

Identifikasi

Rekap Absensi

Operator

Filter Periode

Hari ini

Minggu

Bulan

Semester 1

Semester 2

Tanggal Mulai

Tanggal Akhir

Filter Manual

01/09/2006

01/09/2006

Pilih Filter

Cetak

No	Nama	Kelas	Status Kehadiran				Total Kehadiran	Total Ketidakhadiran	Persentase
			Hadir	Sakit	Izin	Alfa			
24	Pupur Zamira Kuswardani S.IP	1-A	2	0	0	0	2	0	33.3%
25	Qori Astuti S.Kom	1-A	2	0	0	0	2	0	33.3%
26	Queen Yulianti	1-A	2	0	0	0	2	0	33.3%
27	Suci Nurdyanti	1-A	2	0	0	0	2	0	33.3%
28	Tari Fujati	1-A	2	0	0	0	2	0	33.3%
29	Virman Wicana	1-A	2	0	0	0	2	0	33.3%
30	Wani Nadia Pertiwi S.T.	1-A	2	0	0	0	2	0	33.3%
TOTAL			53	2	3	2	53	2	-
PERSENTASE (%)			29.4%	1.1%	1.7%	1.1%	-	-	-

<

1

2

3

4

>

Menampilkan rekap setelah menekan tombol lihat rekap

Skenario Utama

Aksi Aktor	Reaksi Sistem
1. Aktor membuka menu Rekap Absensi.	2. Untuk Guru: sistem menampilkan filter kelas yang terdaftar pada guru. Untuk Operator: sistem menampilkan filter semua kelas. Untuk Kepala sekolah: sistem menampilkan filter semua kelas.
3. Aktor memilih filter kelas dan periode	4. Sistem menampilkan tabel rekap default (bulan berjalan) dengan kolom: Nama Siswa, Kelas, Hadir, Izin, Sakit, Alfa, Total Hari, % Kehadiran.

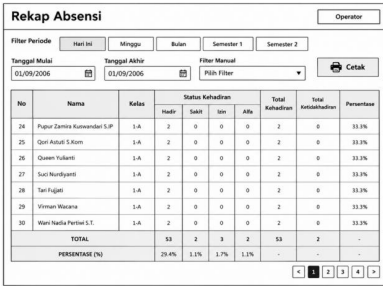
Skenario Alternatif

Kondisi	Reaksi Sistem
Tidak ada data absensi sesuai filter yang dipilih.	Sistem menampilkan tabel kosong dengan pesan: 'Tidak ada data absensi pada periode yang dipilih.'

Tabel 4.14 menjelaskan skenario Use Case melihat rekap absensi siswa.

Tabel 4.15 Skenario UC-06: Cetak Laporan Rekap

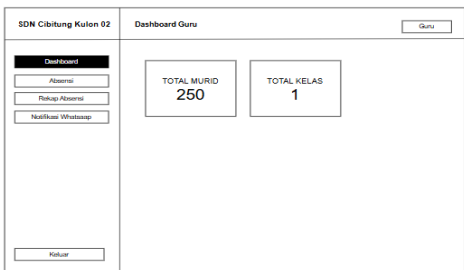
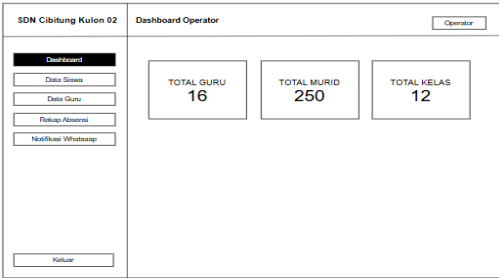
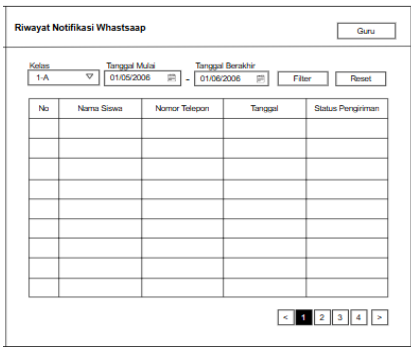
Identifikasi	
Use Case ID	UC-06
Nama Use Case	Cetak Laporan Rekap
Aktor	Operator, Guru, Kepala Sekolah
Tujuan	Menghasilkan laporan sesuai filter
Kondisi Awal	Data rekap sudah ditampilkan

Identifikasi	
 Masuk halaman Lihat rekap	
Kondisi Akhir	File berhasil diunduh
 Menekan tombol cetak rekap dan memunculkan pop up konfirmasi cetak berdasarkan filter yang dimasukkan	
Skenario Utama	
Aksi Aktor	Reaksi Sistem
1. Operator, Guru, Kepala Sekolah mengklik tombol cetak.	2. Sistem mengambil data sesuai filter
	3. Sistem menghasilkan file Excel dengan isi sesuai filter yang dipilih
	4. Sistem mengunduh file sesuai filter yang dipilih
Skenario Alternatif	
Kondisi	Reaksi Sistem
Data kosong	Sistem tetap cetak tapi isinya sesuai dengan data yang ada

Tabel 4.15 menjelaskan skenario Use Case cetak laporan rekap absensi.

Tabel 4.16 Skenario UC-07: Melihat Riwayat Pengiriman *Pesan*

Identifikasi	
Use Case ID	UC-07
Nama Use Case	Melihat Riwayat Pengiriman <i>Pesan</i>

Identifikasi	
Aktor	Operator, Guru
Tujuan	Memungkinkan Operator dan Guru memantau status pengiriman <i>Pesan</i> WhatsApp kepada orang tua siswa yang telah dikirimkan oleh sistem
Kondisi Awal	Aktor berhasil login ke sistem
 Masuk halaman dashboard guru  Masuk halaman dashboard	
Kondisi Akhir	Daftar riwayat pengiriman <i>Pesan</i> WhatsApp ditampilkan sesuai lingkup akses aktor
 Menekan tombol lihat Riwayat dan ditampilkan halaman Riwayat pengiriman pesan	
Skenario Utama	
Aksi Aktor	Reaksi Sistem
1. Operator, Guru membuka menu Riwayat <i>Pesan</i> .	2. Sistem menampilkan tabel riwayat dengan kolom: No, Nama Siswa, Kelas, Nomor WhatsApp Orang tua siswa, Pesan, Waktu Kirim, dan Status (Terkirim/Gagal).
3. Operator, Guru memilih filter kelas dan/atau tanggal, lalu mengklik tombol 'Filter'.	4. Sistem memuat ulang tabel riwayat sesuai parameter filter yang dipilih.
Skenario Alternatif	
Kondisi	Reaksi Sistem
Belum ada <i>Pesan</i> yang pernah dikirim.	Sistem menampilkan tabel kosong dengan pesan: 'Belum ada riwayat pengiriman <i>Pesan</i> .'
Operator, Guru memilih filter yang tidak menghasilkan data.	Sistem menampilkan tabel kosong dengan pesan: 'Tidak ada data pada filter yang dipilih.'

Tabel 4.16 menjelaskan skenario Use Case melihat riwayat pengiriman Pesan WhatsApp kepada orang tua siswa.

Tabel 4.17 Skenario UC-08: Kelola Absensi

Identifikasi	
Use Case ID	UC-08
Nama Use Case	Kelola Absensi
Aktor	Guru
Tujuan	Mencatat kehadiran siswa secara digital per kelas dan tanggal. Setiap penyimpanan absensi yang mengandung siswa dengan status Alpa, Izin, atau Sakit wajib disertai pengiriman Pesan WhatsApp kepada orang tua siswa. Apabila status tersebut kemudian dikoreksi oleh guru, baik berubah menjadi status bermasalah lainnya maupun dikoreksi kembali menjadi Hadir, sistem juga wajib mengirimkan Pesan WhatsApp susulan agar orang tua siswa selalu menerima informasi kehadiran anaknya sesuai dengan data terbaru
Kondisi Awal	Guru berhasil login dan terdaftar pada minimal satu kelas di sistem

SDN Cibitung Kulon 02

Dashboard Guru

Guru

Dashboard

Absensi

Rekap Absensi

Notifikasi Whatsapp

TOTAL MURID

250

TOTAL KELAS

1

Keluar

Masuk halaman dashboard guru

Kondisi Akhir	Data absensi tersimpan di basis data, sistem secara otomatis menjalankan proses kirim Pesan WhatsApp kepada orang tua siswa yang berstatus Alpa, Izin, atau Sakit, hasil pengiriman dicatat di tabel riwayat Pesan
---------------	--

SDN Cibitung Kulon 02

Pencatatan Absensi

Guru

Dashboard

Absensi

Rekap Absensi

Notifikasi Whatsapp

Kelas

1-A

Tanggal

01/05/2008

Filter

Reset

Total Siswa

23

Hadir

20

Izin

1

Sakit

1

Alfa

1

No	Nama Siswa	Status Kehadiran			
		H	I	S	A
		H	I	S	A
		H	I	S	A
		H	I	S	A
		H	I	S	A

Keluar

Simpan

SDN Cibitung Kulon 02

Pencatatan Absensi

Guru

Dashboard

Absensi

Rekap Absensi

Notifikasi Whatsapp

Kelas

1-A

Tanggal

01/05/2024

Filter

Reset

Total Siswa

23

Hadir

20

Izin

1

Sakit

1

Alfa

1

No	Nama Siswa	Status Kehadiran			
1		H	I	S	A
2		H	I	S	A
3		H	I	S	A
4		H	I	S	A
5		H	I	S	A

Keluar

Simpan

Menekan tombol Absensi dan masuk ke halaman absensi

Mencatat kehadiran siswa sesuai status kehadiran

Identifikasi

SDN Cibitung Kulon 02

Dashboard

Absensi

Rekap Absensi

Notifikasi WhatsApp

Keluar

Pencatatan Absensi

Guru

Kelas

1-A

Tanggal

01/06/2024

Filter

Reset

Total Siswa

23

Hadir

20

Izin

1

Sakit

1

Absen

1

No	Nama Siswa	Status Kehadiran
1		H I S A
2		H I S A
3		H I S A
4		H I S A
5		H I S A

Konfirmasi Simpan

Apakah Anda yakin ingin menyimpan absensi untuk tanggal 01/06/2024 kelas 1-A?

Ya, Simpan

Batal

Simpan

Menekan tombol simpan lalu akan ditampilkan pop up konfirmasi simpan absensi

SDN Cibitung Kulon 02

Akun resmi sekolah

Hari ini

⚠ Pesan dan panggilan terestensi secara end-to-end. Hanya orang di chat ini yang bisa membaca, mendengarkan, atau mengirimkannya. Pelajari selengkapnya.

NOTIFIKASI ABSENSI SISWA

Yth. Bapak/Ibu Orang Tua/Wali,
Berikut informasi kehadiran putra/putri Anda.

Nama Siswa

: Ahmad Fauzi

Kelas

: 1-A

Tanggal

: 01 Mei 2024

Jam

: 07:00 WIB

Status Kehadiran

: ALPA

Keterangan

: Tidak hadir tanpa keterangan.

Mohon untuk dapat memperhatikan kehadiran putra/putri Anda.

Terima kasih atas perhatian dan kerja samanya.

Hormat kami,
SDN Cibitung Kulon 02

Pesan ini dikirim otomatis, mohon tidak membalas.

10:30

Kirim pesan

Jika ada siswa Alpa, Izin, atau Sakit maka akan dikirim pesan ke orang tua siswa

SDN Cibitung Kulon 02

Dashboard

Absensi

Rekap Absensi

Notifikasi WhatsApp

Keluar

Pencatatan Absensi

Guru

Kelas

1-A

Tanggal

01/06/2024

Filter

Reset

Total Siswa

23

Hadir

20

Izin

1

Sakit

1

Absen

1

No	Nama Siswa	Status Kehadiran
1	Andi Pratama	Hadir
2	Bunga Lestari	Izin
3	Cahya Putri	Sakit
4	Dimas Saputra	Hadir
5	Eka Ramadhani	Alpa

Edit Absensi

Memasukan tanggal yang sudah diisi absensi lalu menekan tombol edit lalu diarahkan ke halaman koreksi absensi

SDN Cibitung Kulon 02

Dashboard

Absensi

Rekap Absensi

Notifikasi WhatsApp

Keluar

Pencatatan Absensi

Guru

Kelas

1-A

Tanggal

01/06/2024

Filter

Reset

Total Siswa

23

Hadir

20

Izin

1

Sakit

1

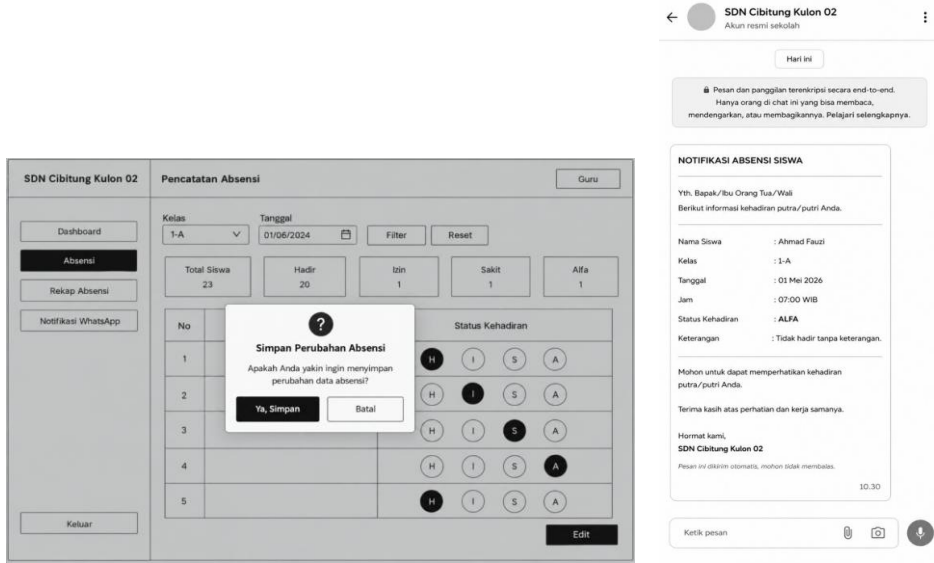
Absen

1

No	Nama Siswa	Status Kehadiran
1		H I S A
2		H I S A
3		H I S A
4		H I S A
5		H I S A

Edit

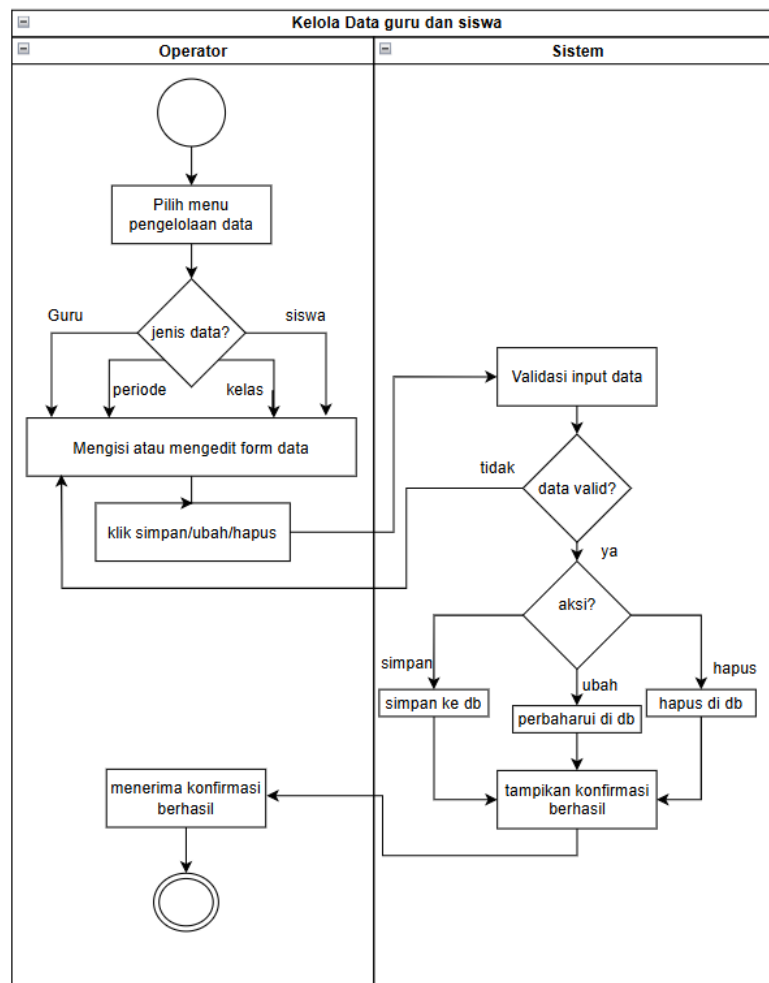
Mengganti status absensi siswa

Identifikasi	
 Menekan tombol simpan lalu diberikan pop up konfirmasi untuk mengoreksi absensi Jika tercatat Alpa, Izin, atau Sakit atau diubah dari tidak hadir ke hadir maka akan dikirim pesan ke orang tua siswa	
Skenario Utama	
Aksi Aktor	Reaksi Sistem
1. Guru membuka menu Absensi	2. Sistem menampilkan halaman absensi
3. Guru memilih tanggal yang belum diisi absensinya.	4. Sistem menampilkan daftar siswa beserta status kehadiran yang dapat diisi, dengan nilai default "Hadir (H)"
5. Guru mengisi status kehadiran setiap siswa (H/I/S/A), lalu mengklik tombol 'Simpan Absensi'.	6. Sistem memvalidasi: seluruh siswa terisi statusnya dan tidak ada duplikasi absensi untuk kelas dan tanggal yang sama.
	7. Sistem menyimpan seluruh data absensi ke tabel absensi.
	8. Sistem mengidentifikasi setiap siswa yang berstatus Alpa, Izin, atau Sakit pada absensi yang baru disimpan.
	9. Untuk setiap siswa yang tercatat Alpa, Izin, atau Sakit, Sistem memanggil <i>Notification Service</i> secara otomatis.
	10. <i>Notification Service</i> mengambil nomor WhatsApp orang tua siswa (prioritas ibu, namun jika nomor ibu tidak valid berganti ke nomor ayah) dari tabel siswa, menyusun pesan berisi nama siswa, kelas, dan tanggal absensi, lalu mengirim melalui <i>API Fonnte</i> .
	11. Sistem mencatat hasil pengiriman ke tabel <i>Pesan logs</i> (status: Terkirim atau Gagal).
	12. Sistem menampilkan pesan sukses: 'Data absensi berhasil disimpan.' beserta ringkasan: Hadir, Izin, Sakit, Alpa.

Identifikasi	
13. Guru membuka menu Absensi dan memilih tanggal yang ingin dikoreksi.	14. Sistem menampilkan daftar siswa beserta status kehadiran yang sudah terisi.
15. Guru mengklik tombol 'Edit'.	16. Sistem menampilkan pilihan status (Hadir/Izin/Sakit/Alpa) dengan nilai saat ini yang sudah terpilih.
17. Guru memilih status baru dan mengklik tombol 'Simpan'.	18. Sistem memvalidasi input.
	19. Sistem memperbarui data absensi di database.
	20. Jika status baru berbeda dengan status sebelumnya dan salah satu di antara status lama atau status baru adalah Alpa, Izin, atau Sakit, Sistem mengirimkan Pesan WhatsApp kepada orang tua siswa (prioritas ibu, namun jika nomor ibu tidak valid berganti ke nomor ayah) melalui API Fonnte. Isi pesan menyesuaikan arah koreksi: jika status baru adalah Alpa, Izin, atau Sakit, pesan berisi informasi ketidakhadiran siswa, jika status baru adalah Hadir (mengoreksi status sebelumnya yang Alpa, Izin, atau Sakit), pesan berisi klarifikasi bahwa status kehadiran siswa telah dikoreksi menjadi Hadir. Sistem kemudian mencatat hasil pengiriman di tabel pesan logs.
	21. Sistem menampilkan pesan sukses: 'Data absensi berhasil diperbarui.'
Skenario Alternatif	
Kondisi	Reaksi Sistem
Nomor WhatsApp orang tua siswa tidak valid atau pengiriman API Fonnte gagal.	Sistem tetap menyimpan data absensi. Status pengiriman dicatat sebagai Gagal di tabel <i>Pesan logs</i> .
Tidak ada siswa yang berstatus Alpa, Izin, atau Sakit (seluruh siswa Hadir).	Proses Kirim <i>Pesan</i> tidak dijalankan. Data absensi tetap tersimpan
Guru isi absensi di tanggal setelah hari ini	Sistem menonaktifkan input tanggal setelah hari ini di filter
Guru isi absensi selain di hari aktif	Sistem menampilkan pesan 'Tidak bisa mengisi absensi karena hari ini bukan hari aktif.'
Guru ingin isi absensi di hari libur	Sistem menampilkan pesan 'Tidak bisa mengisi absensi karena hari ini merupakan hari libur.'
Guru belum terdaftar pada kelas maupun mencoba membuka menu Absensi	Sistem menampilkan pesan bahwa Guru belum terdaftar pada kelas

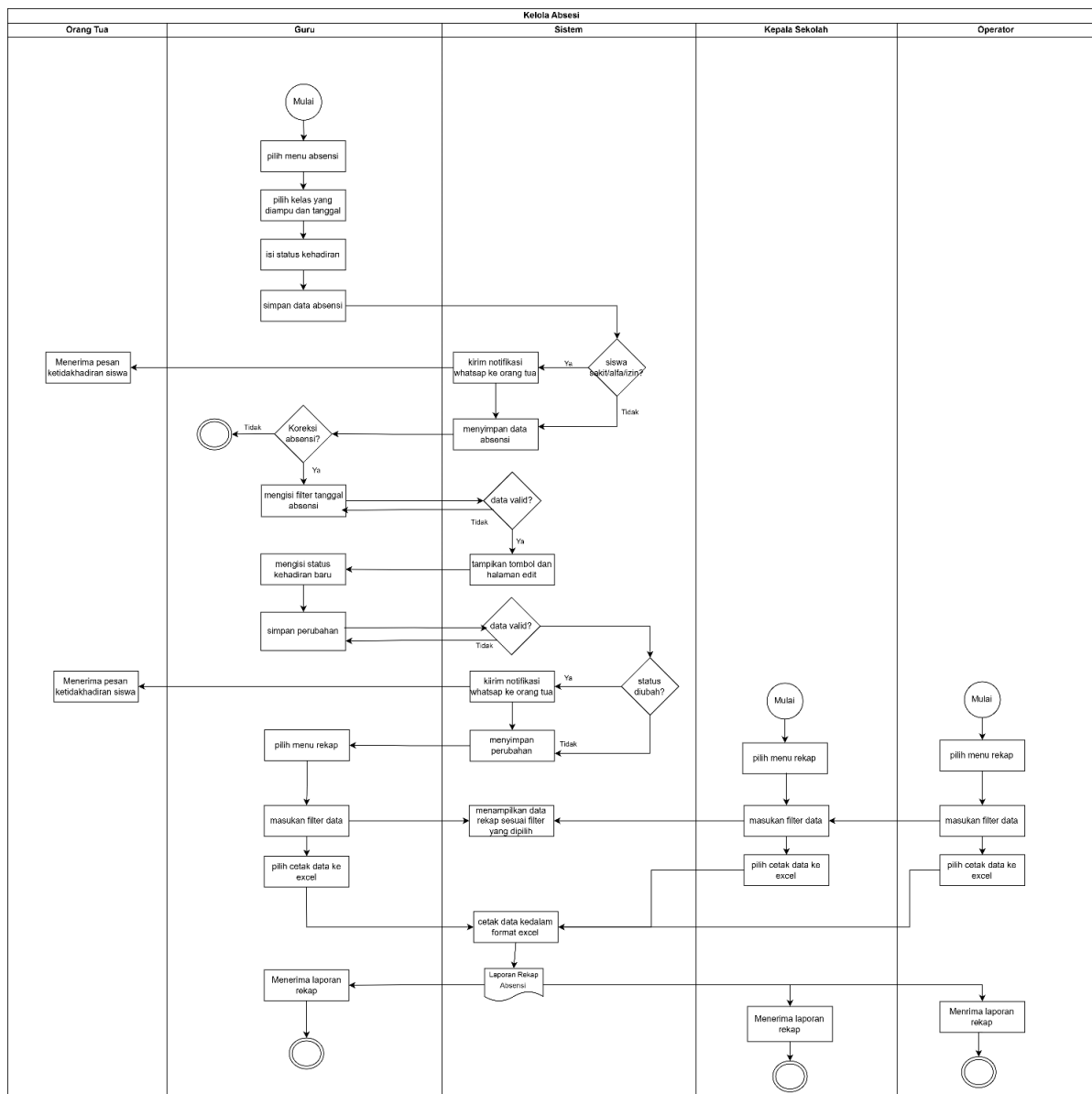
Tabel 4.17 menjelaskan skenario Use Case kelola absensi, mencakup pencatatan kehadiran dan pengiriman Pesan WhatsApp otomatis saat status berubah ke Alpa, Izin, atau Sakit, maupun kembali ke Hadir.

4.2.3 Activity Diagram



Gambar 4.5 Activity Diagram Kelola guru, kelas, siswa dan periode

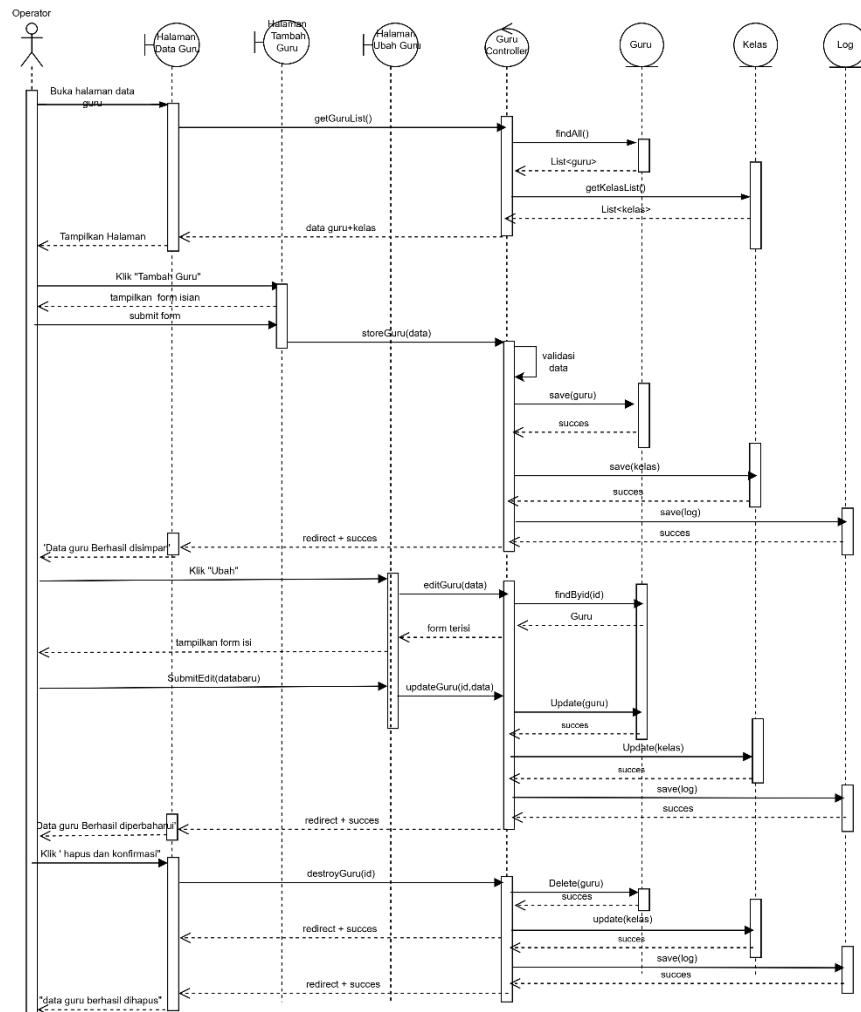
Gambar 4.5 menggambarkan activity diagram pengelolaan data master oleh Operator, mencakup data guru, siswa, kelas, hingga periode.



Gambar 4.6 Activity Diagram proses Absensi rekap, dan Kirim *Pesan* ke Orang tua siswa

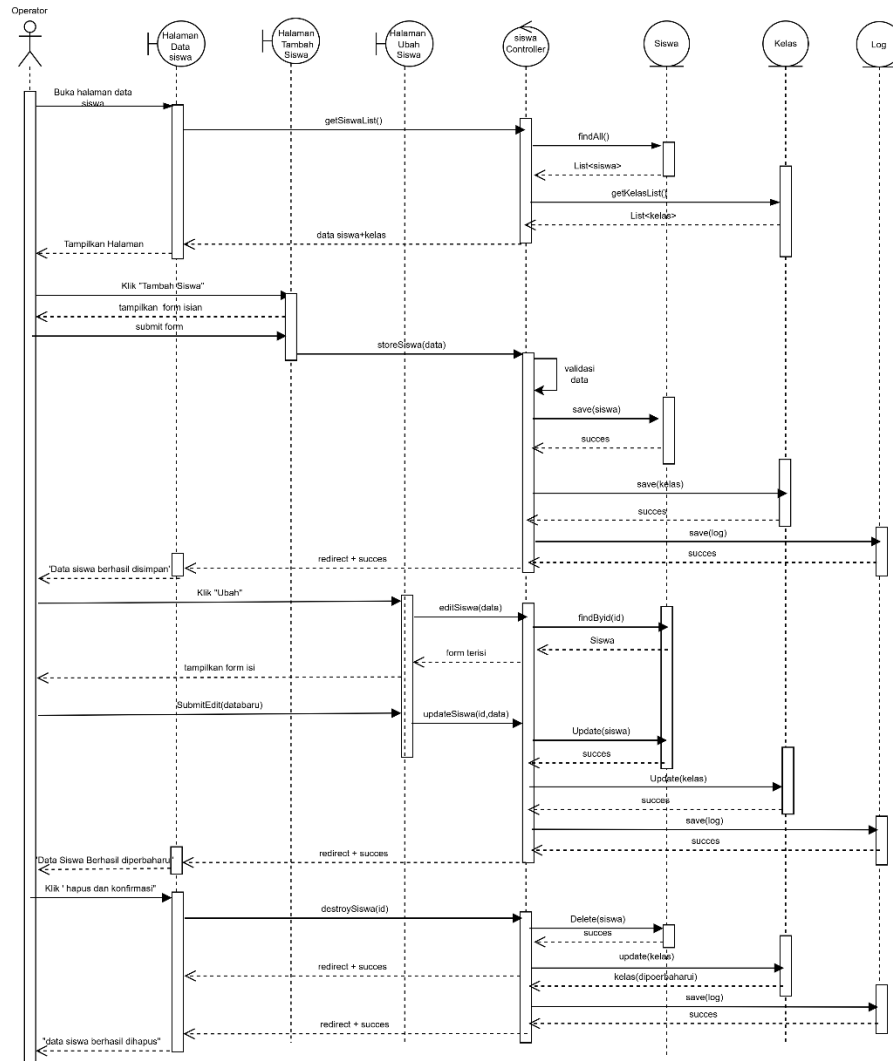
Gambar 4.6 menggambarkan activity diagram absensi, rekap, dan pengiriman pesan WhatsApp otomatis, dari pemilihan kelas hingga pengiriman melalui API Fonnte.

4.2.4 Sequence Diagram



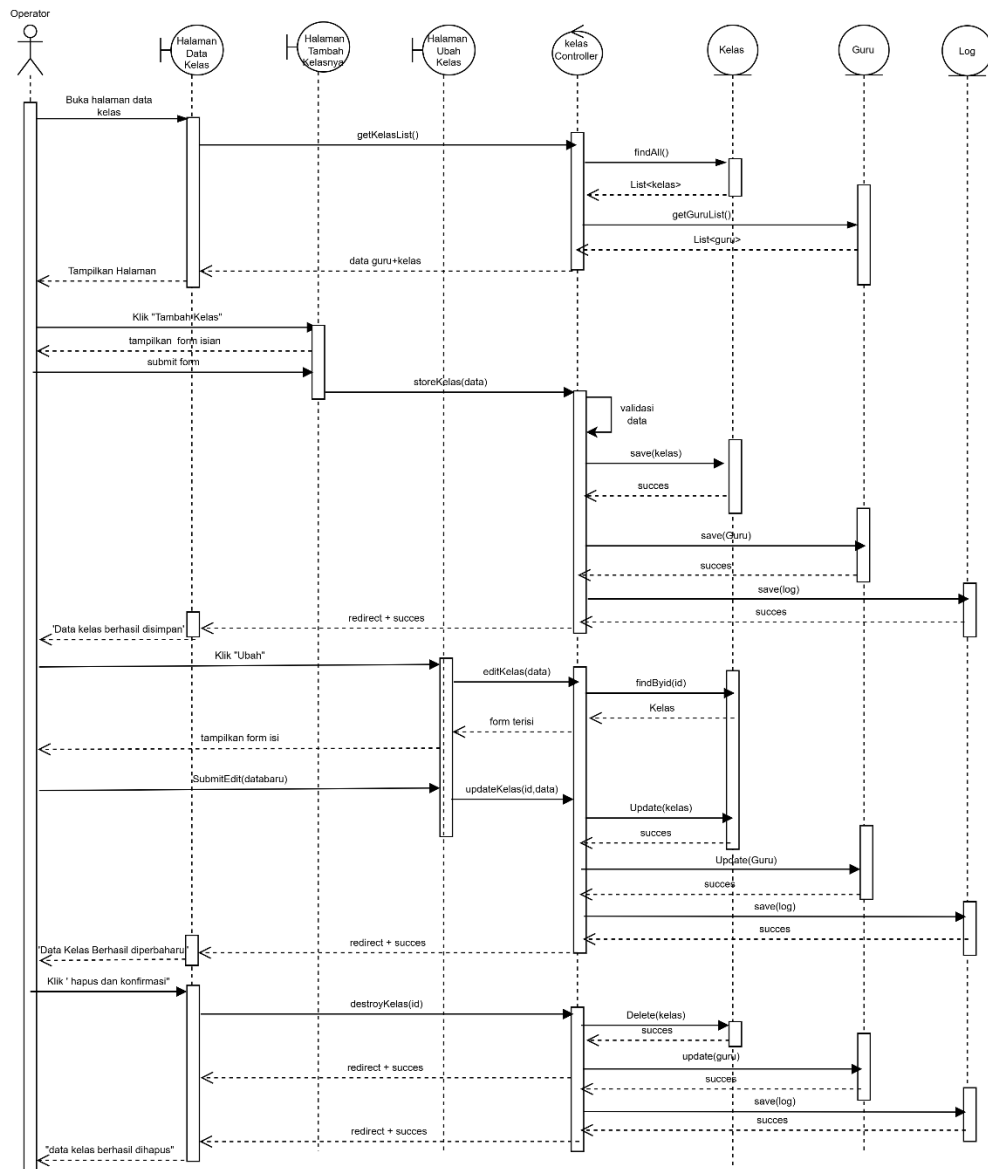
Gambar 4.7 Sequence Diagram Kelola Data Guru

Gambar 4.7 menggambarkan sequence diagram kelola data guru oleh Operator, mencakup interaksi dengan controller dan basis data serta validasi sebelum penyimpanan.



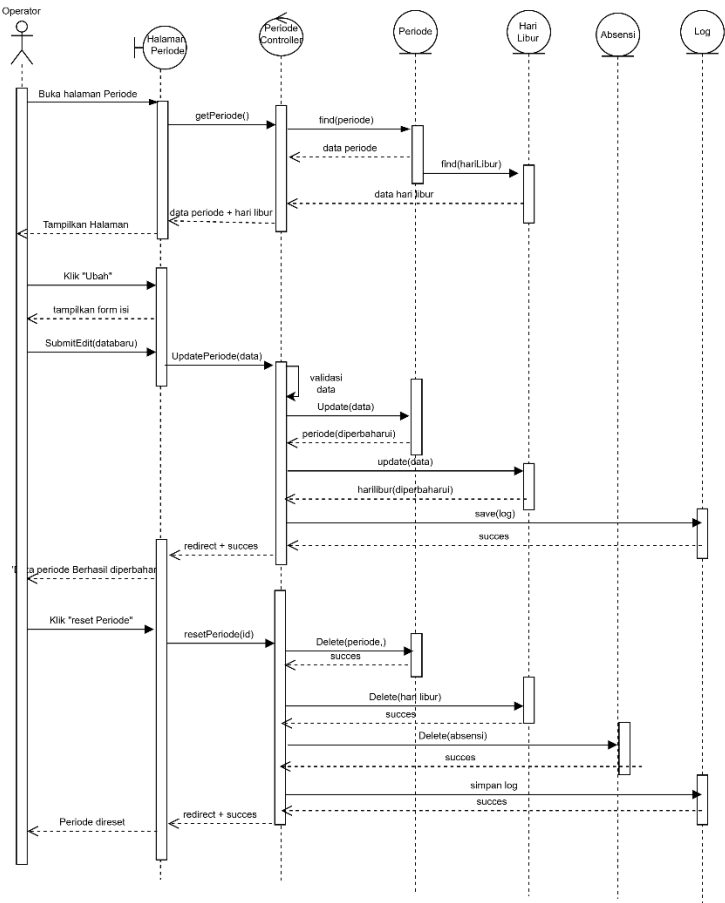
Gambar 4.8 Sequence Diagram Kelola Data Siswa

Gambar 4.8 menggambarkan sequence diagram kelola data siswa oleh Operator, termasuk validasi data siswa dan orang tua sebelum disimpan.



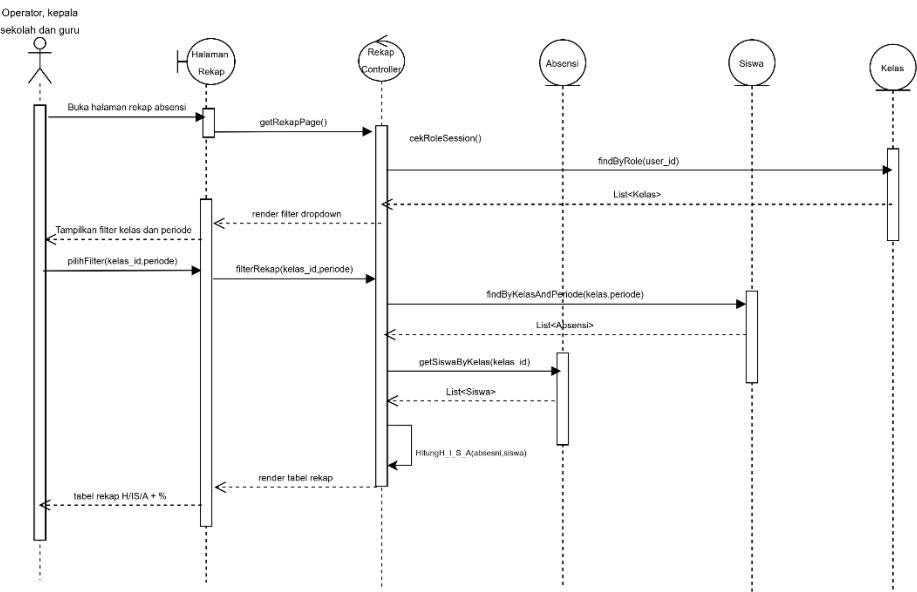
Gambar 4.9 Sequence Diagram Kelola Data Kelas

Gambar 4.9 menggambarkan sequence diagram kelola data kelas oleh Operator dengan validasi sebelum data disimpan.



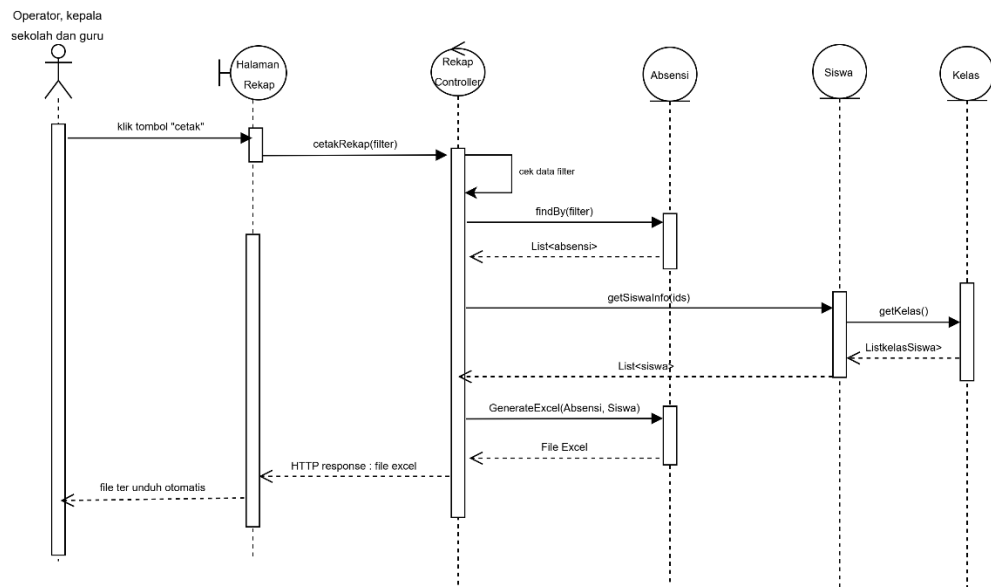
Gambar 4.10 Sequence Diagram Kelola Periode Tahun Ajaran

Gambar 4.10 menggambarkan sequence diagram pengelolaan periode tahun ajaran oleh Operator, mencakup penambahan, aktivasi, hari libur, hingga pengarsipan siswa lulus.



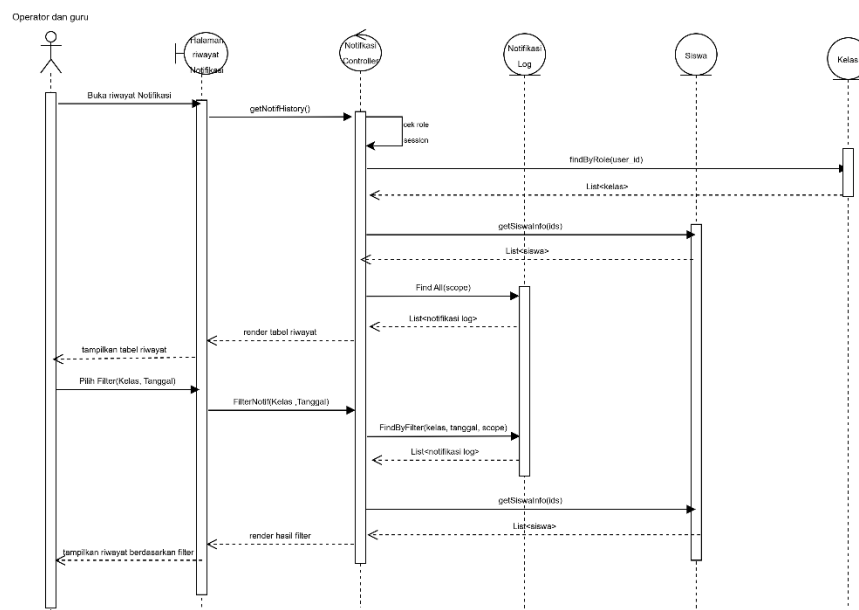
Gambar 4.11 Sequence Diagram Lihat Rekap Absensi

Gambar 4.11 menggambarkan sequence diagram melihat rekap absensi oleh Operator, Guru, dan Kepala Sekolah sesuai filter yang dipilih.



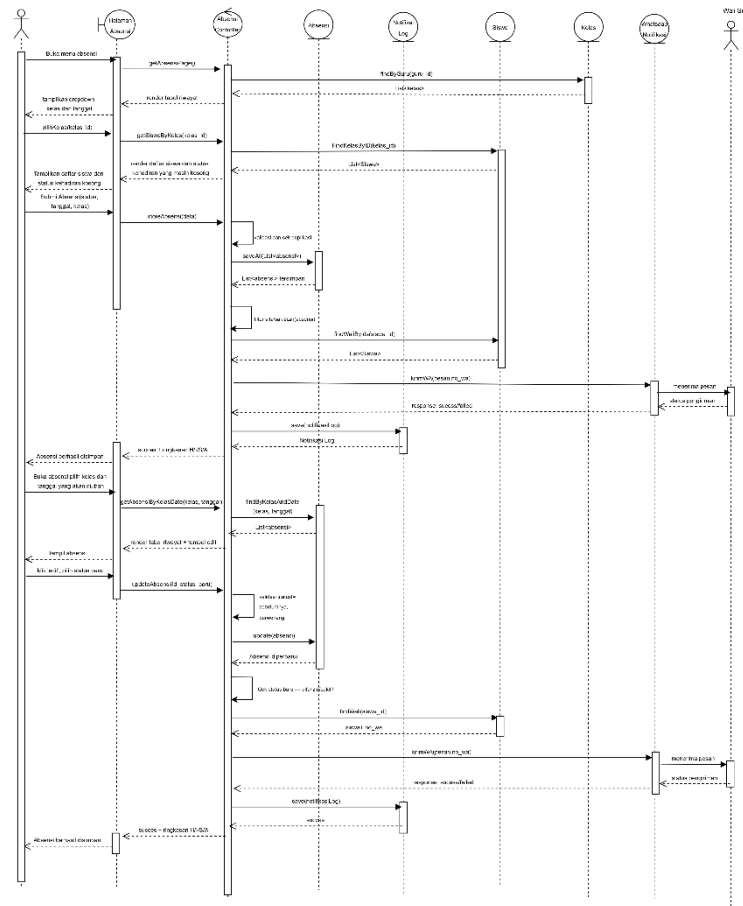
Gambar 4.12 Sequence Diagram Cetak Laporan Rekap

Gambar 4.12 menggambarkan sequence diagram cetak laporan rekap absensi hingga menghasilkan dokumen PDF yang dapat diunduh.



Gambar 4.13 Sequence Diagram Lihat Riwayat Pengiriman Pesan

Gambar 4.13 menggambarkan sequence diagram melihat riwayat pengiriman pesan WhatsApp oleh Operator dan Guru, lengkap dengan status dan waktu pengiriman.

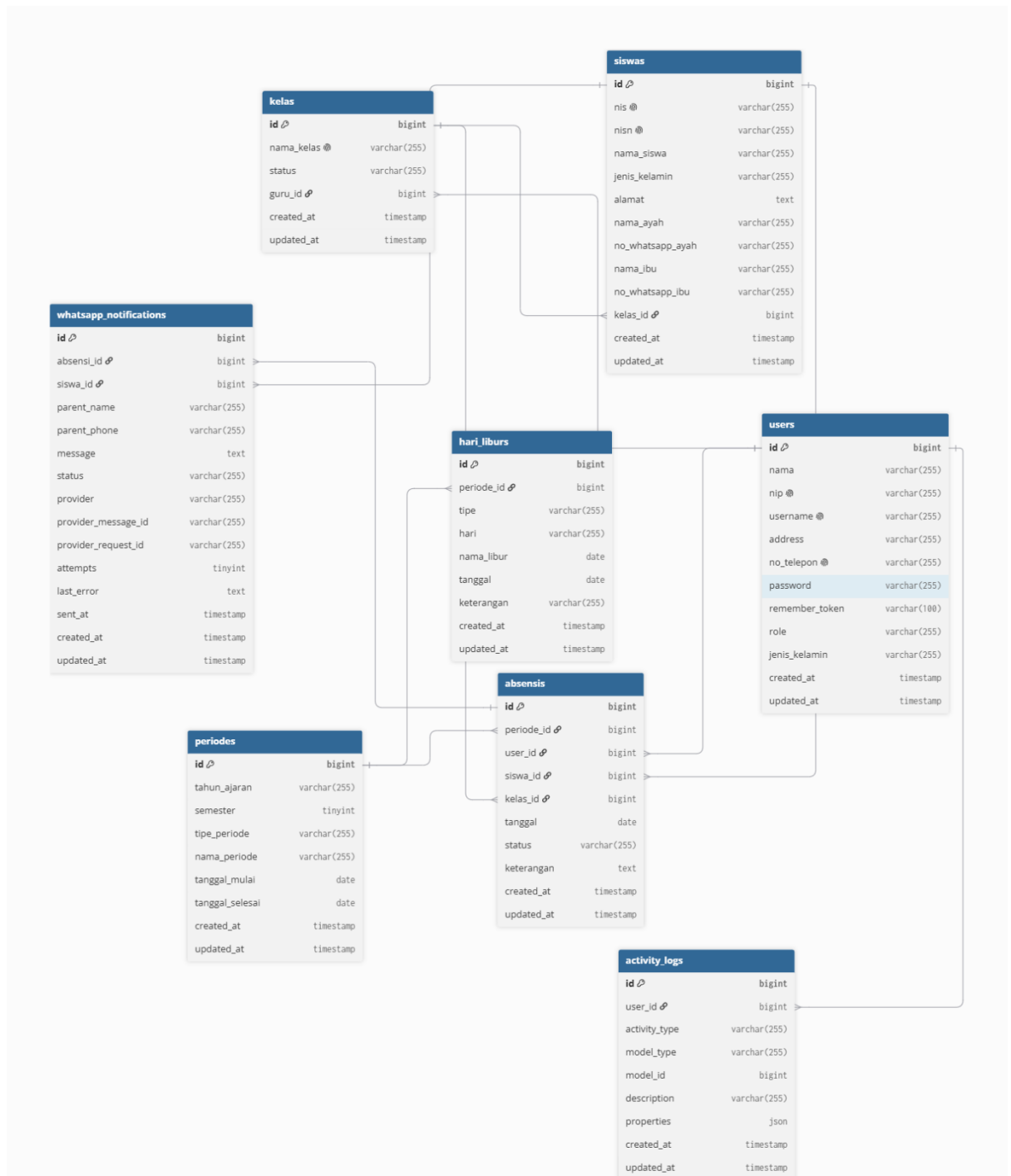


Gambar 4.14 Sequence Diagram Kelola Absensi

Gambar 4.14 menggambarkan sequence diagram kelola absensi oleh Guru: memilih kelas dan tanggal, mengisi status kehadiran, lalu sistem menyimpan data dan otomatis mengirim WhatsApp ke orang tua jika status Alpa, Izin, atau Sakit.

4.2.5 Entity Relationship Diagram (ERD)

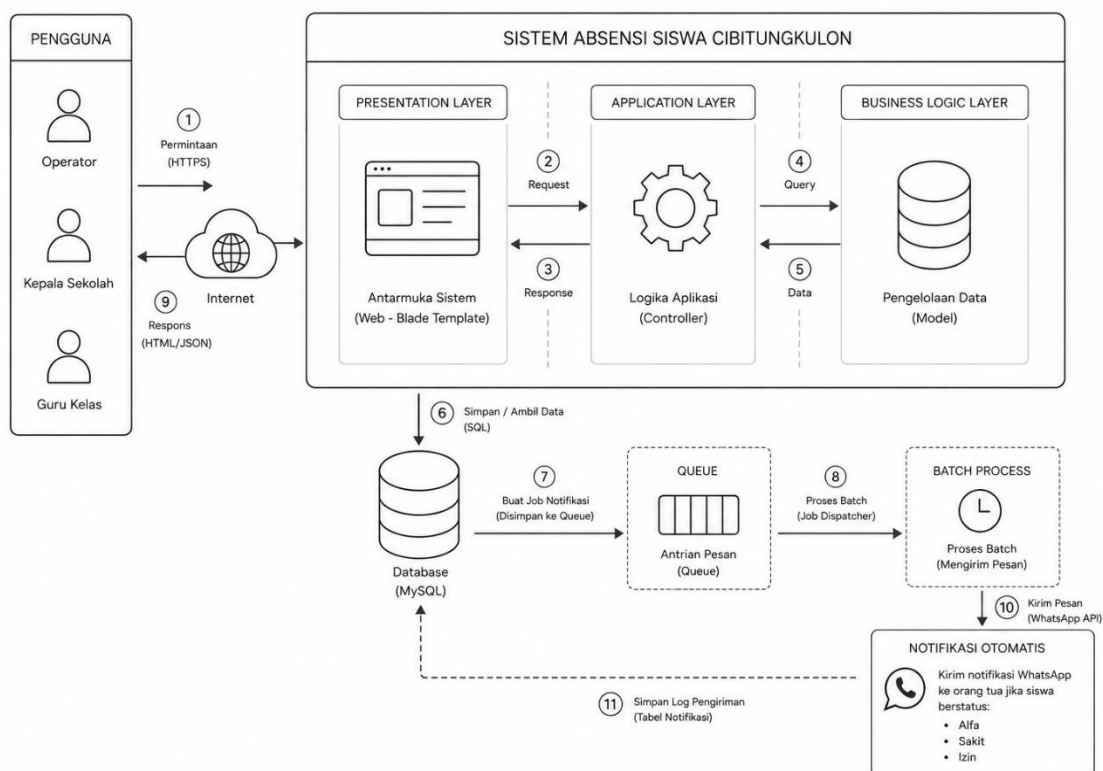
Rancangan database merupakan hasil implementasi dari Entity Relationship Diagram (ERD) yang digunakan untuk menyimpan seluruh data pada Sistem Informasi Absensi Siswa Berbasis Website dengan WhatsApp *Gateway*. Database dirancang menggunakan MySQL dengan tujuan menjaga konsistensi, integritas, dan kemudahan pengelolaan data.



Gambar 4.15 Entity Relationship Diagram (ERD)

Gambar 4.15 Entity Relationship Diagram (ERD) Absensi Siswa Berbasis Website dengan WhatsApp Gateway

4.2.6 Gambaran Umum Infrastruktur yang Dibangun



Gambar 4.16 Gambaran Umum Infrastruktur

Gambar 4.16 menunjukkan gambaran umum infrastruktur sistem absensi siswa berbasis website yang dibangun menggunakan konsep arsitektur Model–View–Controller (MVC). Sistem ini digunakan oleh guru dan operator untuk mengelola data absensi siswa melalui jaringan internet menggunakan browser.

Secara garis besar, alur kerja infrastruktur pada Gambar 4.16 dapat dibagi menjadi dua bagian, yaitu alur permintaan-tanggapan (*request-response*) yang berjalan secara sinkron ketika pengguna berinteraksi dengan sistem, dan alur notifikasi otomatis yang berjalan secara asinkron di latar belakang menggunakan mekanisme antrian (*queue*) dan proses *batch*. Penjelasan tiap tahapan mengikuti nomor pada Gambar 4.16 diuraikan sebagai berikut:

1. Permintaan (HTTPS), Pengguna (Operator, Kepala Sekolah, Guru) mengakses sistem melalui jaringan internet menggunakan protokol HTTPS, misalnya membuka halaman absensi atau menginput data kehadiran siswa.
2. Request, Permintaan diteruskan oleh internet menuju *Presentation Layer* (antarmuka sistem berbasis *Web-Blade Template*), lalu diteruskan ke *Application Layer* untuk diproses.
3. Response, Setelah diproses oleh *Application Layer*, hasilnya dikirim kembali ke *Presentation Layer* untuk ditampilkan kepada pengguna.

4. Query, *Application Layer* mengirim permintaan data ke *Business Logic Layer* (bagian Model).
5. Data, *Business Logic Layer* mengembalikan data hasil pengolahan ke *Application Layer* untuk diproses lebih lanjut.
6. Simpan/ambil data (MYSQL), *Business Logic Layer* menggunakan Eloquent ORM Laravel untuk berkomunikasi langsung dengan Database MySQL, baik menyimpan data baru maupun mengambil data tersimpan. Mendukung proses CRUD atas data guru, siswa, kelas, absensi, rekapitulasi, riwayat absensi, dan pesan WhatsApp.
7. Buat job notifikasi (disimpan ke *queue*), Jika status absensi yang diinput adalah Alpa, Sakit, atau Izin, atau koreksi ke hadir sistem otomatis membuat job notifikasi dan menyimpannya ke *Queue*, bukan mengirim langsung agar proses pengiriman tidak membebani respons utama sistem.
8. Proses *batch* (*job dispatcher*), Job-job dalam antrian diproses secara *batch* oleh *Job Dispatcher* yang berjalan di *background* untuk mengeksekusi antrian satu per satu.
9. Respons (HTML/JSON), Bersamaan dengan proses notifikasi di latar belakang, hasil akhir permintaan pengguna dikembalikan dalam bentuk HTML (tampilan halaman) atau JSON (data), sehingga pengguna tak perlu menunggu proses notifikasi selesai.
10. Kirim pesan (WhatsApp API), *Batch Process* yang dieksekusi memicu pengiriman pesan melalui WhatsApp API menuju modul Notifikasi Otomatis, mengirim pesan ke orang tua jika status kehadiran tercatat Alpa, Sakit, atau Izin.
11. Simpan log pengiriman (tabel notifikasi), Setelah pesan berhasil/gagal dikirim, sistem mencatat log ke tabel notifikasi pada database, untuk melacak riwayat, waktu, dan tujuan notifikasi.

Dengan pemisahan antara alur sinkron dan alur asinkron tersebut, sistem tidak perlu menunggu proses pengiriman pesan WhatsApp selesai untuk dapat memberikan respons kepada pengguna.

BAB 5

KONSTRUKSI DAN PENGUJIAN

5.1 Konstruksi Perangkat Lunak

Konstruksi perangkat lunak adalah proses pembuatan perangkat lunak melalui kombinasi penulisan kode, verifikasi, dan pengujian integrasi. Pada penelitian ini, sistem dibangun menggunakan Laravel dan MySQL dengan arsitektur MVC, digunakan oleh tiga peran pengguna (Operator, Guru, Kepala Sekolah), serta terintegrasi dengan WhatsApp *Gateway* (Fonnte) melalui mekanisme antrean (queue) di latar belakang.

5.1.1 Kakas dan Kebutuhan Perangkat Lunak

Sebelum melakukan pengkodean, perlu dipersiapkan kakas (tools) dan kebutuhan yang mendukung proses pembangunan sistem, mencakup spesifikasi perangkat keras yang digunakan selama pengembangan, perangkat lunak pendukung, serta teknologi utama yang menjadi fondasi pembangunan sistem.

5.1.1.1 Spesifikasi Perangkat Keras

Tabel 5.1 Spesifikasi Perangkat Keras

No	Komponen	Spesifikasi
1	Processor	Intel Core i7 generasi ke-11
2	RAM	8 GB
3	Penyimpanan	500 GB SSD
4	Sistem Operasi	Windows1 64-bit

Spesifikasi perangkat keras yang digunakan selama proses pengembangan sistem ditunjukkan pada tabel berikut.

5.1.1.2 Perangkat Lunak yang Digunakan

Tabel 5.2 Perangkat Lunak yang Digunakan

No	Perangkat Lunak	Fungsi
1	Visual Studio Code	Editor kode utama untuk penulisan seluruh berkas PHP, Blade, dan JavaScript
2	Laragon	Menyediakan server lokal Apache/Nginx dan MySQL untuk lingkungan pengembangan
3	Composer	Manajer dependensi PHP untuk instalasi Laravel dan paket pendukung (maatwebsite/excel)

No	Perangkat Lunak	Fungsi
4	Node.js dan NPM	Mengelola dependensi front-end dan menjalankan proses build Tailwind CSS serta Alpine.js melalui Vite
5	Git dan GitHub	Version control dan penyimpanan kode sumber sistem
6	Google Chrome	Browser untuk pengujian tampilan antarmuka dan proses debugging
7	Docker	Menjalankan lingkungan kontainer sesuai berkas Dockerfile pada repositori untuk kebutuhan deployment

Tabel 5.2 menjelaskan sejumlah perangkat lunak digunakan untuk mendukung proses pengembangan, mulai dari penulisan kode hingga pengujian API. Daftar perangkat lunak yang digunakan ditunjukkan pada tabel berikut.

5.1.1.3 Teknologi Pengembangan Sistem

Tabel 5.3 Teknologi Pengembangan Sistem

No	Teknologi / Kakas	Kategori	Fungsi dalam Sistem
1	Laravel 13	<i>Framework</i>	Kerangka kerja PHP (PHP ^8.3) sebagai fondasi utama pembangunan sistem dengan arsitektur MVC, routing, middleware, Eloquent ORM, queue, dan job scheduling
2	PHP	Bahasa Pemrograman	Bahasa pemrograman sisi server utama untuk seluruh logika sistem
3	Blade	<i>Template Engine</i>	Mesin templat Laravel untuk merender tampilan antarmuka pengguna pada lapisan View
4	Laravel Breeze	Starter Kit Autentikasi	Menyediakan fondasi autentikasi (login, reset password, manajemen sesi) melalui controller pada namespace App\Http\Controllers\Auth, dikembangkan lebih lanjut dengan middleware CheckRole
5	Tailwind CSS, Alpine.js	Antarmuka Pengguna	Membangun antarmuka pengguna yang responsif dan interaktif tanpa memuat ulang halaman secara penuh, dapat diakses dari browser desktop maupun smartphone
6	MySQL	Basis Data	Sistem manajemen basis data relasional untuk menyimpan seluruh data pengguna, siswa, kelas, absensi, dan pesan
7	WhatsApp Gateway (Fonnte)	Layanan Eksternal	API pihak ketiga yang diakses melalui kelas FonnteService untuk mengirim pesan WhatsApp secara paralel (HTTP pool) kepada orang tua siswa
8	Laravel Queue (database driver)	Antrean Latar Belakang	Menjalankan SendWhatsappBatchJob secara asinkron dengan mekanisme retry dan backoff bertingkat agar pengiriman pesan tidak memblokir proses penyimpanan absensi

No	Teknologi / Kakas	Kategori	Fungsi dalam Sistem
9	maatwebsite/excel (Laravel Excel)	Pelaporan	Menghasilkan berkas Excel (.xlsx) pada fitur ekspor rekap absensi melalui kelas RekapAbsensiExport

Pembangunan sistem membutuhkan sejumlah teknologi dan kakas pengembangan. Daftar berikut disusun berdasarkan lingkungan pengembangan yang digunakan dalam penelitian ini.

5.1.2 Pengelolaan Konstruksi

Tabel 5.4 Pembagian Urutan Implementasi

Fase	Modul	Fungsi yang Diimplementasikan	Berkas Utama
1	Basis Data & Model	Pembuatan 10 berkas migrasi dan 8 kelas Model (<i>User</i> , <i>Periode</i> , <i>Kelas</i> , <i>Siswa</i> , <i>HariLibur</i> , <i>Absensi</i> , <i>WhatsappNotification</i> , <i>ActivityLog</i>) beserta relasinya	app/Models/*.php, database/migrations/
2	Autentikasi & Middleware Role	<i>Login</i> berbasis Laravel Breeze, reset password, dan middleware <i>CheckRole</i> yang membatasi akses menu sesuai role (<i>Operator/Guru/Kepala Sekolah</i>)	app/Http/Controllers/Auth/*.php, app/Http/Middleware/CheckRole.php
3	Manajemen Pengguna & Data Master	CRUD data pengguna (<i>Operator/Guru/Kepala Sekolah</i>), kelas, siswa (termasuk pindah kelas), dan periode tahun ajaran beserta pengelolaan hari libur mingguan/nasional	GuruController.php, KelasController.php, SiswaController.php, PeriodeController.php
4	Pencatatan Absensi & Pesan Otomatis	Pencatatan kehadiran siswa per kelas dan tanggal, koreksi data absensi pada hari berjalan, serta pembuatan dan pengiriman pesan WhatsApp otomatis ke orang tua siswa saat status <i>Alpa</i> , <i>Izin</i> , atau <i>Sakit</i> melalui antrean	AbsensiController.php, app/Services/FonnteService.php, app/Jobs/SendWhatsappBatchJob.php
5	Rekapitulasi & Ekspor Laporan	Perhitungan rekap kehadiran (<i>Hadir/Izin/Sakit/Alpa</i>) berdasarkan filter kelas dan rentang tanggal, serta ekspor laporan ke format Excel	RekapController.php, app/Exports/RekapAbsensiExport.php
6	Riwayat Pesan	Penyimpanan dan penampilan riwayat pengiriman pesan WhatsApp beserta status pengiriman	NotifikasiController.php
7	Log Aktivitas Pengguna	Pencatatan dan penampilan riwayat aktivitas <i>create/update/delete</i> yang dilakukan pengguna pada data guru, siswa, kelas, dan periode	ActivityLogController.php, app/Http/Controllers/Concerns/LogsActivity.php

Pengelolaan konstruksi menggambarkan bagaimana proses pembangunan sistem diatur secara terstruktur. Pembangunan sistem dibagi menjadi 7 fase implementasi berdasarkan urutan ketergantungan antar modul, dimulai dari basis data hingga fitur log aktivitas.

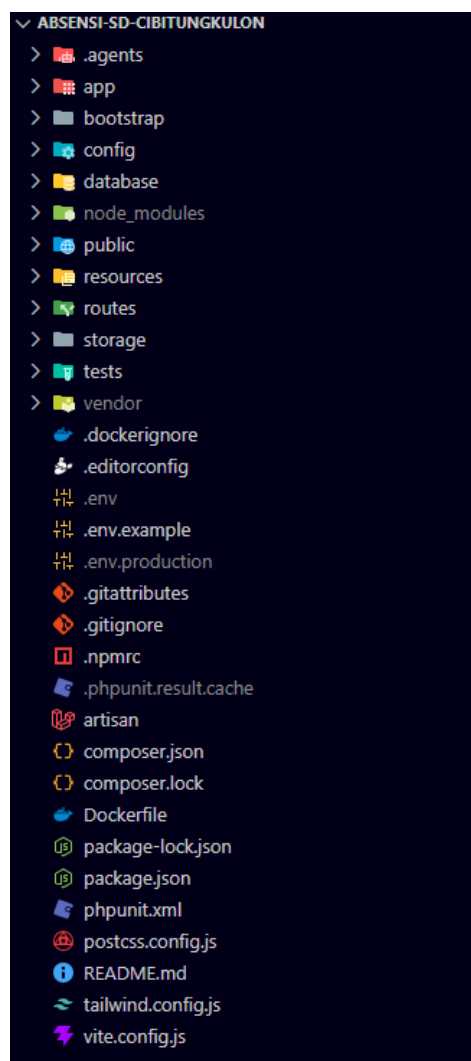
5.2 Pengkodean

Pengkodean merupakan tahap penerjemahan hasil perancangan pada BAB 4 ke dalam kode program menggunakan kerangka kerja Laravel dengan pola arsitektur Model-View-Controller (MVC).

Bagian ini menjelaskan struktur folder proyek, implementasi kelas Model, View, dan Controller beserta Service dan Job pendukung, serta penerapan integrasi layanan eksternal (WhatsApp *Gateway* Fonnte).

5.2.1 Struktur Folder

Struktur folder proyek mengikuti default Laravel: `app/Models` untuk kelas Model, `app/Http/Controllers` untuk kelas Controller (dengan sub-folder `Auth` untuk autentikasi Laravel Breeze dan `Concerns` untuk trait bersama), `app/Services` untuk integrasi WhatsApp *Gateway*, `app/Jobs` untuk antrean pengiriman pesan, `app/Exports` untuk ekspor Excel, `resources/views` untuk Blade, `routes/web.php` dan `routes/auth.php` untuk routing, serta `database/migrations` untuk skema basis data, sebagaimana ditunjukkan pada gambar berikut.



Gambar 5.1 Struktur folder proyek

Gambar 5.1 menyajikan struktur folder proyek yang mengikuti default Laravel.

Tabel 5.5 Struktur folder

No	Nama Folder	Deskripsi
1.	app	Berisi komponen inti aplikasi, termasuk Model, Controller, Middleware, Service Provider, dan logika bisnis lainnya.
2.	bootstrap	Mengelola proses inisialisasi awal aplikasi dan memuat file cache konfigurasi serta autoload yang diperlukan untuk menjalankan sistem.
3.	config	Berisi seluruh berkas konfigurasi aplikasi seperti database, mail, cache, queue, session, dan filesystem.
4.	database	Menyimpan migration, seeder, dan factory yang digunakan dalam pengaturan struktur dan data awal basis data.
5.	node_modules	Berisi seluruh dependensi JavaScript/Node.js yang diinstal melalui npm, digunakan dalam proses build aset front-end (Vite dan Tailwind CSS).
6.	public	Direktori yang dapat diakses langsung oleh pengguna melalui web server, berisi file index.php, aset hasil kompilasi (CSS/JS), dan gambar publik.
7.	resources	Menyimpan file tampilan Blade (views), serta berkas CSS dan JavaScript mentah sebelum melalui proses kompilasi.
8.	routes	Berisi definisi rute aplikasi seperti web.php dan auth.php yang menentukan alur akses fitur.
9.	storage	Digunakan untuk menyimpan cache, log, file upload, serta hasil kompilasi Blade dan data sementara aplikasi.
10.	tests	Menyimpan skrip pengujian aplikasi menggunakan PHPUnit untuk memastikan fungsi berjalan dengan benar.
11.	vendor	Folder yang berisi seluruh dependensi PHP/library yang diinstal melalui Composer.
12.	.dockerignore	Menentukan file/folder yang diabaikan saat proses build image Docker.
13.	.editorconfig	Mengatur konsistensi gaya penulisan kode (indentasi, encoding) antar editor/IDE yang berbeda.
14.	.env	Menyimpan konfigurasi environment aplikasi, seperti koneksi database, mail, dan kunci aplikasi untuk lingkungan development.
15.	.env.example	Contoh/template variabel environment yang dibutuhkan aplikasi, digunakan sebagai acuan pembuatan file .env.
16.	.env.production	Menyimpan konfigurasi environment khusus untuk lingkungan production/deployment.
17.	.gitattributes	Mengatur atribut penanganan file oleh Git, seperti normalisasi baris baru dan file yang dikecualikan saat export.
18.	.gitignore	Menentukan file/folder yang tidak dilacak oleh Git version control.
19.	.npmrc	Berkas konfigurasi npm, misalnya pengaturan registry atau opsi instalasi package.
20.	.phpunit.result.cache	Berkas cache hasil pengujian PHPUnit untuk mempercepat proses testing berikutnya.

No	Nama Folder	Deskripsi
21.	artisan	Skrip command-line interface (CLI) bawaan Laravel untuk menjalankan berbagai perintah, seperti migration, membuat controller, dan menjalankan server.
22.	composer.json	Mendefinisikan daftar dependensi PHP beserta metadata proyek yang dikelola oleh Composer.
23.	composer.lock	Mencatat versi pasti dependensi Composer yang terinstal agar konsisten di setiap lingkungan.
24.	Dockerfile	Berisi instruksi untuk membangun image Docker aplikasi.
25.	package-lock.json	Mencatat versi pasti dependensi npm yang terinstal agar konsisten di setiap lingkungan.
26.	package.json	Mendefinisikan daftar dependensi JavaScript beserta script build/development proyek.
27.	phpunit.xml	Berkas konfigurasi untuk menjalankan pengujian PHPUnit.
28.	postcss.config.js	Berkas konfigurasi PostCSS yang digunakan dalam proses kompilasi Tailwind CSS.
29.	README.md	Berisi dokumentasi singkat mengenai proyek, seperti cara instalasi dan menjalankan aplikasi.
30.	tailwind.config.js	Berkas konfigurasi Tailwind CSS untuk kustomisasi tema dan utilitas styling.
31.	vite.config.js	Berkas konfigurasi Vite, digunakan untuk proses bundling dan kompilasi aset front-end.

Tabel 5.5 menjelaskan struktur folder dan berkas utama pada root proyek yang mengikuti konvensi standar Laravel.

5.2.2 Model

Tabel 5.6 Struktur Kelas Model

No	Nama Kelas	Deskripsi
1	User.php	Merepresentasikan seluruh pengguna sistem (Operator, Guru, Kepala Sekolah) dengan pembeda kolom role, menyimpan NIP, nama, alamat, username, nomor telepon, jenis kelamin, dan relasi satu-ke-satu ke kelas yang diampu (khusus Guru)
2	Periode.php	Merepresentasikan periode tahun ajaran dan semester (Ganjil/Genap) beserta rentang tanggal mulai/selesai, memiliki relasi ke Kelas, Siswa, HariLibur, dan Absensi
3	Kelas.php	Merepresentasikan data kelas beserta status aktif dan wali kelas (guru_id), memiliki scope accessibleBy() untuk membatasi akses data sesuai role pengguna yang sedang login
4	Siswa.php	Merepresentasikan data identitas siswa (NIS, NISN) termasuk data ayah/ibu dan nomor WhatsApp masing-masing sebagai tujuan pesan, memiliki relasi ke Kelas dan

No	Nama Kelas	Deskripsi
		Absensi, serta method hasAbsensi() untuk memeriksa riwayat presensi sebelum siswa dipindahkan ke kelas lain
5	HariLibur.php	Merepresentasikan hari libur pada suatu periode, terdiri dari tipe mingguan (mis. Minggu/Sabtu) atau nasional (tanggal spesifik) yang diinput manual oleh Operator
6	Absensi.php	Merepresentasikan pencatatan kehadiran siswa per kelas dan tanggal dengan status hadir/izin/sakit/alpa, terhubung dengan guru pencatat (user_id) dan periode aktif
7	WhatsappNotification.php	Merepresentasikan antrean dan riwayat pengiriman pesan WhatsApp ke orang tua siswa, termasuk status (pending/processing/sent/failed/cancelled), jumlah percobaan (attempts), dan pesan error terakhir, menerapkan MassPrunable untuk pembersihan data lama
8	ActivityLog.php	Merepresentasikan log aktivitas create/update/delete yang dilakukan pengguna terhadap data Guru, Siswa, Kelas, atau Periode, termasuk data lama dan baru dalam kolom properties (JSON)

Tabel 5.6 menjelaskan kedelapan kelas model yang merepresentasikan entitas pada basis data sesuai hasil rancangan Entity Relationship Diagram (ERD).

5.2.3 View

Tabel 5.7 Struktur Folder View

No	Folder View	Deskripsi
1	auth/	Halaman login dan reset password untuk seluruh peran (disediakan Laravel Breeze)
2	guru/	Halaman kelola pengguna sistem (daftar, tambah, ubah) untuk akun Operator, Guru, dan Kepala Sekolah
3	siswa/	Halaman CRUD data siswa beserta halaman pindah kelas (ubah-kelas.blade.php)
4	kelas/	Halaman CRUD data kelas
5	periode/	Halaman kelola periode tahun ajaran beserta input hari libur mingguan dan nasional
6	absensi/	Halaman pencatatan (create) dan koreksi (edit) absensi harian oleh Guru
7	rekap/	Halaman rekapitulasi kehadiran siswa beserta tombol unduh laporan Excel
8	notifikasi/	Halaman riwayat pengiriman pesan WhatsApp
9	activity-log/	Halaman riwayat log aktivitas pengguna
10	profile/	Halaman pengaturan profil akun pengguna yang sedang login

No	Folder View	Deskripsi
11	layouts/ dan components/	Kerangka tampilan (navbar, sidebar) dan komponen Blade yang dibedakan sesuai peran pengguna aktif
12	dashboard.blade.php	Berkas tampilan halaman Dashboard, berada langsung di root resources/views/ (bukan di dalam folder terpisah)
13	vendor/pagination/	Override tampilan pagination bawaan Laravel bergaya Tailwind, dipublish dari package framework

Tabel 5.7 menjelaskan lapisan View dibangun menggunakan Blade, mesin templat bawaan Laravel, dengan struktur folder resources/views yang dikelompokkan berdasarkan modul dan peran pengguna. Setiap halaman memanfaatkan Tailwind CSS untuk tampilan responsif dan Alpine.js untuk interaksi sisi klien tanpa memuat ulang halaman secara penuh. Pengelompokan folder View ditunjukkan pada tabel berikut.

5.2.4 Controller dan Service

Tabel 5.8 Struktur Kelas Controller, Service, Job, dan Export

No	Kategori	Nama Kelas	Deskripsi
1	Controller (Auth)	AuthenticatedSessionController.php, PasswordResetLinkController.php, NewPasswordController.php, PasswordController.php, ConfirmablePasswordController.php	Menangani proses login, logout, dan reset password bawaan Laravel Breeze
2	Controller	DashboardController.php	Menampilkan ringkasan jumlah kelas, siswa, dan guru sesuai lingkup akses role pengguna
3	Controller	GuruController.php	Menangani CRUD data pengguna sistem (Operator, Guru, Kepala Sekolah) beserta penugasan wali kelas
4	Controller	KelasController.php	Menangani CRUD data kelas beserta penugasan/pelepasan wali kelas
5	Controller	SiswaController.php	Menangani CRUD data siswa beserta fitur pindah kelas
6	Controller	PeriodeController.php	Menangani pengelolaan periode tahun ajaran (Semester 1 dan 2 sekaligus) beserta hari libur mingguan/nasional
7	Controller	AbsensiController.php	Menangani pencatatan dan koreksi absensi harian, serta pemicu pembuatan pesan WhatsApp untuk siswa berstatus Alpa, Izin, atau Sakit
8	Controller	RekapController.php	Menangani tampilan rekapitulasi kehadiran dan ekspor laporan ke Excel
9	Controller	NotifikasiController.php	Menampilkan riwayat pengiriman pesan WhatsApp

No	Kategori	Nama Kelas	Deskripsi
10	Controller	ActivityLogController.php	Menampilkan riwayat log aktivitas pengguna
11	Controller	ProfileController.php	Menangani pengaturan profil akun pengguna yang sedang login
12	Service	FonnteService.php	Mengirim pesan WhatsApp ke <i>API Fonnte</i> secara tunggal maupun paralel (HTTP connection pool) dan menangani status HTTP yang dapat dicoba ulang
13	Job	SendWhatsappBatchJob.php	Job antrean yang memproses batch pesan WhatsApp berstatus Alpa, Izin, atau Sakit, dengan 5 kali percobaan dan jeda backoff bertingkat (60/300/900/1800/3600 detik), termasuk mekanisme kontak cadangan (fallback)
14	Export	RekapAbsensiExport.php	Menghasilkan berkas Excel (.xlsx) dari data rekap absensi yang telah difilter
15	Console Command	RetryFailedWhatsappNotifications.php	Artisan command (whatsapp:retry-failed) yang menjadwalkan ulang notifikasi WhatsApp berstatus failed (maksimal 20 percobaan) melalui SendWhatsappBatchJob

Tabel 5.8 menjelaskan sebelas kelas Controller (termasuk lima controller autentikasi bawaan Laravel Breeze) menangani logika bisnis sistem sesuai hasil identifikasi use case pada BAB 4, dibantu satu kelas Service (FonnteService) yang menangani komunikasi HTTP ke WhatsApp *Gateway*, satu kelas Job (SendWhatsappBatchJob) yang menjalankan pengiriman pesan secara asinkron melalui antrean, satu kelas Export (RekapAbsensiExport) untuk menghasilkan berkas Excel, dan satu Console Command (RetryFailedWhatsappNotifications) yang dapat dijalankan terjadwal untuk menjadwalkan ulang notifikasi WhatsApp yang gagal terkirim. Struktur kelas Controller, Service, Job, Export, dan Command ditunjukkan pada tabel berikut.

5.2.5 Basis Data

Tabel 5.9 Daftar Tabel Basis Data

No	Nama Tabel	Deskripsi
1	users	Menyimpan data akun pengguna sistem (Operator, Guru, Kepala Sekolah) beserta kredensial login dan role akses
2	perodes	Menyimpan data periode tahun ajaran dan semester (Ganjil/Genap) beserta rentang tanggal mulai/selesai
3	kelas	Menyimpan data kelas beserta status aktif dan wali kelas (guru_id, merujuk ke tabel users)
4	siswas	Menyimpan data identitas siswa beserta data orang tua/wali dan nomor WhatsApp ayah/ibu untuk pengiriman pesan

No	Nama Tabel	Deskripsi
5	hari_liburs	Menyimpan data hari libur mingguan atau nasional yang diinput manual oleh Operator per periode
6	absensis	Mencatat kehadiran siswa harian (hadir/izin/sakit/alpa) beserta guru yang mencatat dan periode terkait
7	whatsapp_notifications	Mencatat antrean dan riwayat pengiriman pesan WhatsApp ke orang tua siswa beserta status pengiriman (pending/processing/sent/failed/cancelled), jumlah percobaan, dan pesan error
8	activity_logs	Mencatat riwayat aktivitas create/update/delete yang dilakukan pengguna terhadap data guru, siswa, kelas, dan periode

Tabel 5.9 menjelaskan basis data sistem dibuat menggunakan MySQL berdasarkan hasil Entity Relationship Diagram (ERD) pada BAB 4, terdiri dari 8 tabel utama: users, periodes, kelas, siswas, hari_liburs, absensis, whatsapp_notifications, dan activity_logs. Daftar tabel beserta fungsinya disusun pada tabel berikut.

Tabel 5.10 Struktur Tabel users

No	Nama Field	Tipe Data	Keterangan
1	id	bigint, PK	Auto increment
2	nama	string	Nama pengguna
3	nip	string, unique per role, nullable	NIP (kosong untuk sebagian role), unik dalam lingkup role yang sama (constraint users_role_nip_unique), bukan unik global
4	username	string, unique	Digunakan untuk login
5	alamat	string, nullable	Alamat pengguna
6	no_telepon	string, unique	Nomor telepon
7	password	string	Hash password
8	role	enum: operator, guru, kepala_sekolah	Peran akses pengguna
9	jenis_kelamin	string, nullable	Jenis kelamin (L/P)
10	created_at, updated_at	timestamp	Dikelola otomatis oleh Laravel

Tabel 5.10 memuat struktur field tabel users, tempat data akun pengguna (Operator, Guru, Kepala Sekolah) disimpan.

Tabel 5.11 Struktur Tabel periodes

No	Nama Field	Tipe Data	Keterangan
1	id	bigint, PK	Auto increment
2	tahun_ajaran	string	Contoh: 2025/2026
3	semester	tinyint, nullable	1 = Ganjil, 2 = Genap
4	tipe_periode	enum: semester, tahunan (default: semester)	Jenis periode
5	nama_periode	string	Nama tampilan periode
6	tanggal_mulai	date	Tanggal awal periode
7	tanggal_selesai	date	Tanggal akhir periode
8	created_at, updated_at	timestamp	Dikelola otomatis oleh Laravel

Tabel 5.11 memuat struktur field tabel periodes, tempat data periode tahun ajaran dan semester disimpan.

Tabel 5.12 Struktur Tabel kelas

No	Nama Field	Tipe Data	Keterangan
1	id	bigint, PK	Auto increment
2	nama_kelas	string, unique	Nama kelas
3	status	string (default: aktif)	Status kelas
4	guru_id	bigint, FK ke users.id, nullable	Wali kelas, onDelete: set null
5	created_at, updated_at	timestamp	Dikelola otomatis oleh Laravel

Tabel 5.12 memuat struktur field tabel kelas, tempat data kelas beserta status dan wali kelasnya disimpan.

Tabel 5.13 Struktur Tabel siswas

No	Nama Field	Tipe Data	Keterangan
1	id	bigint, PK	Auto increment
2	nis	string, unique, nullable	Nomor Induk Siswa
3	nisn	string, unique, nullable	Nomor Induk Siswa Nasional
4	nama_siswa	string	Nama siswa

No	Nama Field	Tipe Data	Keterangan
5	jenis_kelamin	enum: laki-laki, perempuan	Jenis kelamin siswa
6	alamat	text, nullable	Alamat siswa
7	nama_ayah	string	Nama ayah
8	no_whatsapp_ayah	string, nullable	Untuk notifikasi WhatsApp
9	nama_ibu	string	Nama ibu
10	no_whatsapp_ibu	string, nullable	Untuk notifikasi WhatsApp
11	kelas_id	bigint, FK ke kelas.id	Kelas siswa saat ini
12	created_at, updated_at	timestamp	Dikelola otomatis oleh Laravel

Tabel 5.13 memuat struktur field tabel siswas, tempat identitas siswa dan data orang tua/wali disimpan.

Tabel 5.14 Struktur Tabel hari_liburs

No	Nama Field	Tipe Data	Keterangan
1	id	bigint, PK	Auto increment
2	periode_id	bigint, FK ke periodes.id	onDelete: cascade
3	tipe	enum: mingguan, nasional	Jenis hari libur
4	hari	string, nullable	Diisi jika tipe mingguan, mis. "Minggu"
5	tanggal	date, nullable	Diisi jika tipe nasional
6	nama_libur	string, nullable	Nama/label hari libur (opsional)
7	keterangan	string	Keterangan hari libur
8	created_at, updated_at	timestamp	Dikelola otomatis oleh Laravel

Tabel 5.14 memuat struktur field tabel hari_liburs, tempat data hari libur mingguan atau nasional per periode disimpan.

Tabel 5.15 Struktur Tabel absensis

No	Nama Field	Tipe Data	Keterangan
1	id	bigint, PK	Auto increment
2	siswa_id	bigint, FK ke siswas.id	onDelete: cascade
3	kelas_id	bigint, FK ke kelas.id	Kelas saat absen dicatat
4	user_id	bigint, FK ke users.id	Guru yang mencatat

No	Nama Field	Tipe Data	Keterangan
5	periode_id	bigint, FK ke periodes.id	Periode terkait
6	tanggal	date	Tanggal absen
7	status	enum: hadir, izin, sakit, alpa	Status kehadiran
8	keterangan	text, nullable	Catatan tambahan
9	created_at, updated_at	timestamp	Dikelola otomatis oleh Laravel

Tabel 5.15 memuat struktur field tabel absensis untuk mencatat kehadiran siswa harian.

Tabel 5.16 Struktur Tabel whatsapp_notifications

No	Nama Field	Tipe Data	Keterangan
1	id	bigint, PK	Auto increment
2	absensi_id	bigint, FK ke absensis.id	cascadeOnDelete
3	siswa_id	bigint, FK ke siswas.id	cascadeOnDelete
4	parent_name	string, nullable	Nama orang tua penerima
5	parent_phone	string, nullable	Nomor WhatsApp orang tua
6	message	text	Isi pesan
7	status	string (default: pending)	pending/processing/sent/failed/cancelled
8	provider	string (default: fonnte)	Penyedia layanan WhatsApp
9	provider_message_id	string, nullable	ID pesan dari provider
10	provider_request_id	string, nullable	ID request dari provider
11	attempts	tinyint unsigned (default: 0)	Jumlah percobaan kirim
12	last_error	text, nullable	Pesan error terakhir
13	sent_at	timestamp, nullable	Waktu pesan terkirim
14	created_at, updated_at	timestamp	Dikelola otomatis oleh Laravel

Tabel 5.16 memuat struktur field tabel whatsapp_notifications untuk mencatat antrean dan riwayat pengiriman pesan WhatsApp ke orang tua siswa.

Tabel 5.17 Struktur Tabel activity_logs

No	Nama Field	Tipe Data	Keterangan
1	id	bigint, PK	Auto increment

No	Nama Field	Tipe Data	Keterangan
2	user_id	bigint, FK ke users.id	cascadeOnDelete
3	activity_type	string	create/update/delete
4	model_type	string	Siswa/Guru/Kelas/Periode
5	model_id	bigint unsigned, nullable	ID data yang diubah
6	description	string	Deskripsi aktivitas
7	properties	json, nullable	Data lama dan baru
8	created_at, updated_at	timestamp	Dikelola otomatis oleh Laravel

Tabel 5.17 memuat struktur field tabel activity_logs untuk mencatat riwayat aktivitas create/update/delete pengguna.

5.2.6 Konfigurasi Routing Sistem

Tabel 5.18 Daftar Route Sistem

No	Metode	URI	Controller/Fungsi	Middleware / Akses
1	GET	/	Closure (redirect ke dashboard/login)	-
2	GET	/login	AuthenticatedSessionController/create	guest
3	POST	/login	AuthenticatedSessionController/store	guest
4	GET	/forgot-password	PasswordResetLinkController/create	guest
5	POST	/forgot-password	PasswordResetLinkController/store	guest
6	GET	/reset-password/{token}	NewPasswordController/create	guest
7	POST	/reset-password	NewPasswordController/store	guest
8	GET	/confirm-password	ConfirmablePasswordController/show	auth
9	POST	/confirm-password	ConfirmablePasswordController/store	auth
10	PUT	/password	PasswordController/update	auth
11	POST	/logout	AuthenticatedSessionController/destroy	auth
12	GET	/dashboard	DashboardController/index	auth
13	GET	/activity-logs	ActivityLogController/index	auth
14	GET	/rekap-absensi	RekapController/index	auth, role: operator, guru, kepala_sekolah
15	GET	/rekap-absensi/export	RekapController/export	auth, role: operator, guru, kepala_sekolah
16	GET	/notifikasi	NotifikasiController/index	auth, role: operator, guru
17	GET	/profile	ProfileController/edit	auth
18	PATCH	/profile	ProfileController/update	auth
19	GET	/guru	GuruController/index	auth, role: operator
20	GET	/guru/create	GuruController/create	auth, role: operator
21	POST	/guru	GuruController/store	auth, role: operator
22	GET	/guru/{guru}/edit	GuruController/edit	auth, role: operator
23	PUT	/guru/{guru}	GuruController/update	auth, role: operator
24	DELETE	/guru/{guru}	GuruController/destroy	auth, role: operator
25	GET	/kelas	KelasController/index	auth, role: operator
26	GET	/kelas/create	KelasController/create	auth, role: operator
27	POST	/kelas	KelasController/store	auth, role: operator
28	GET	/kelas/{kelas}/edit	KelasController/edit	auth, role: operator
29	PUT	/kelas/{kelas}	KelasController/update	auth, role: operator

No	Metode	URI	Controller/Fungsi	Middleware / Akses
30	DELETE	/kelas/{kelas}	KelasController/destroy	auth, role: operator
31	GET	/periode	PeriodeController/index	auth, role: operator
32	POST	/periode	PeriodeController/store	auth, role: operator
33	PUT	/periode/{periode}	PeriodeController/update	auth, role: operator
34	POST	/periode/reset	PeriodeController/reset	auth, role: operator
35	GET	/siswa/ubah-kelas	SiswaController/ubahKelasForm	auth, role: operator
36	POST	/siswa/ubah-kelas	SiswaController/ubahKelas	auth, role: operator
37	GET	/siswa	SiswaController/index	auth, role: operator, guru
38	GET	/siswa/create	SiswaController/create	auth, role: operator, guru
39	POST	/siswa	SiswaController/store	auth, role: operator, guru
40	GET	/siswa/{siswa}/edit	SiswaController/edit	auth, role: operator, guru
41	PUT	/siswa/{siswa}	SiswaController/update	auth, role: operator, guru
42	DELETE	/siswa/{siswa}	SiswaController/destroy	auth, role: operator, guru
43	GET	/absensi	AbsensiController/create	auth, role: guru
44	GET	/absensi/create	AbsensiController/create	auth, role: guru
45	POST	/absensi/store	AbsensiController/store	auth, role: guru
46	GET	/absensi/edit	AbsensiController/edit	auth, role: guru
47	PUT	/absensi/update	AbsensiController/update	auth, role: guru

Seluruh route sistem kecuali halaman login dan reset password mewajibkan pengguna untuk masuk sistem terlebih dahulu (middleware auth) dan sebagian dibatasi middleware role sesuai peran (Operator, Guru, atau kombinasi beberapa peran).

5.3 Antarmuka dan Kode

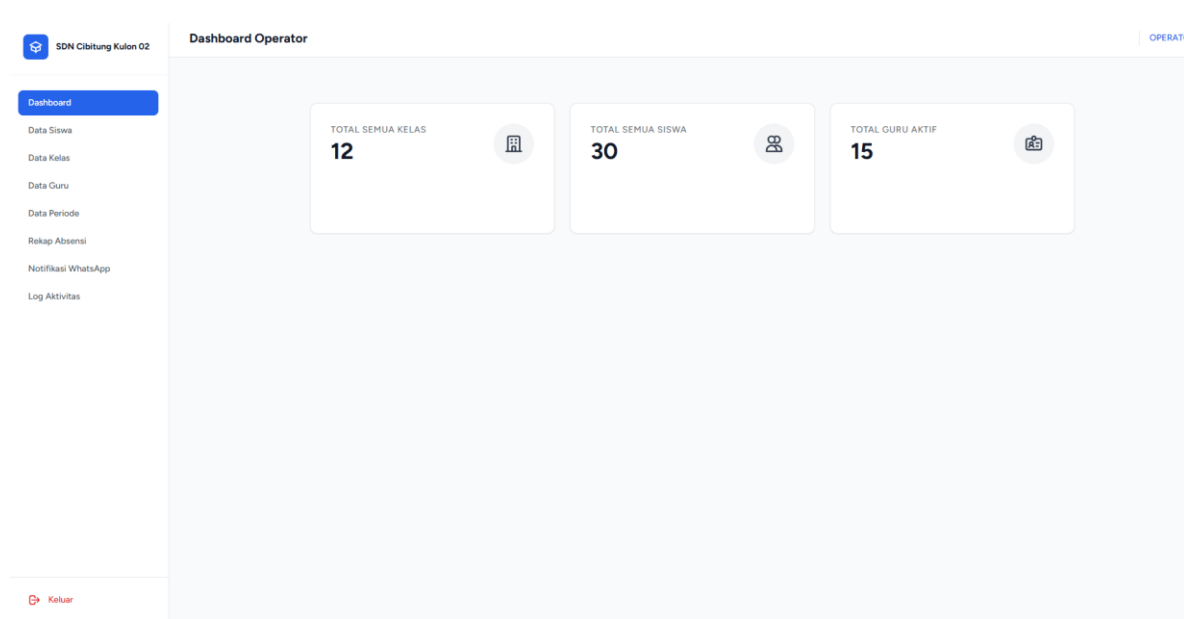
Bagian ini menampilkan implementasi antarmuka (halaman) sistem yang disusun berdasarkan use case pada BAB 4. Setiap halaman didokumentasikan dalam tiga bagian: tangkapan layar tampilan halaman, cuplikan kode program yang membangun halaman tersebut, dan deskripsi fungsi halaman. Tangkapan layar dan cuplikan kode disisipkan berdasarkan implementasi aktual pada repositori sistem.

Tabel 5.19 Antarmuka dan Kode Halaman Login

Halaman Login
SDN Cibitung Kulon 02 Masuk Silakan masuk ke akun Anda untuk melanjutkan. Username atau NIP Masukkan username atau NIP Kata Sandi Masukkan password ☐ Ingat saya di perangkat ini Masuk → Lupa kata sandi? Belum punya akun? Hubungi operator sekolah.

Halaman Login
<pre> 'password' => trans('auth.password'),]; } RateLimiter::clear(\$this->throttleKey()); } </pre>
Deskripsi
Halaman ini menampilkan form autentikasi pengguna sebelum masuk dalam sistem

Tabel 5.20 Antarmuka dan Kode Halaman Dashboard

Halaman Dashboard

Kode
<pre> class DashboardController extends Controller { public function index(Request \$request): View { \$user = \$request->user(); \$totalKelas = Kelas::query()->accessibleBy(\$user)->count(); \$totalSiswa = Siswa::query() ->whereHas('kelas', fn (\$query) => \$query->accessibleBy(\$user)) ->count(); \$totalGuru = in_array(\$user->role, ['operator', 'kepala_sekolah'], true) </pre>

Gambar 5.3 Tampilan halaman Dashboard

Halaman Dashboard
<pre>? User::query()->where('role', 'guru')->count() : 0; return view('dashboard', compact('totalKelas', 'totalSiswa', 'totalGuru')); } }</pre>
Deskripsi
Halaman ini menampilkan daftar ringkasan sistem seperti total siswa, guru dan kelas yang dapat diakses oleh guru, kepala sekolah dan Operator

Tabel 5.21 Antarmuka dan Kode Halaman Daftar Guru

Halaman Daftar Guru

The screenshot displays the 'Daftar Pengguna / Guru' (Teacher List) page within the 'Dashboard Operator' interface. The page features a sidebar on the left with navigation links: 'Dashboard', 'Data Siswa', 'Data Kelas', 'Data Guru' (highlighted), 'Data Periode', 'Rekap Absensi', 'Notifikasi WhatsApp', and 'Log Aktivitas'. The top bar shows the school name 'SDN Cibitung Kulon 02' and the user role 'OPERATOR'. The main content area contains a table of teachers with the following data:

NO	NIP	USERNAME	NAMA LENGKAP	ALAMAT	NO. TELEPON	ROLE AKSES	KELAS DIAJPU	AKSI
1	1975800013	guru13	Baijan Pangestu	Kl. Abang No. 209, Bitar 95377, Maluku	081210000013	GURU	-	Ec
2	1975800007	guru7	Banawi Indra Hardiansyah	Dk. Babakan No. 126, Kumpang 36142, Kaltel	081210000007	GURU	4-A	Ec
3	1975800004	guru4	Citra Riyanti S.Farm	Jr. Baranang No. 201, Padang 72076, Kaltang	081210000004	GURU	2-B	Ec
4	1975800011	guru11	Ega Kuswoyo	Per. Bass No. 281, Tebing Tinggi 94689, Kaltara	081210000011	GURU	6-A	Ec
5	1975800012	guru12	Elma Ulya Safitri	Ds. Suprpto No. 573, Lubuklinggau 48837, Kaltara	081210000012	GURU	6-B	Ec

At the bottom of the table, the text 'Ds. Antasani' is visible. The page also includes a search bar at the top right of the table area with the placeholder 'Cari nama atau NIP guru...' and buttons for 'CARI' and '+ TAMBAH GURU'. A 'Keluar' (Logout) button is located in the bottom left corner of the sidebar area.

Gambar 5.4 Tampilan halaman Daftar Guru

Kode

Halaman Daftar Guru
<pre> public function index(Request \$request): View { \$filters = \$request->validate(['search' => ['nullable', 'string', 'max:100'],]); \$search = trim(\$filters['search'] ?? ''); \$query = User::query() ->select(['id', 'nip', 'username', 'nama', 'no_telepon', 'role', 'alamat']) ->with('kelas:id,guru_id,nama_kelas'); // Cari berdasarkan nama, no telepon, NIP, atau username if (\$search !== '') { \$query->where(function (\$query) use (\$search): void { \$query->where('nama', 'like', "%{\$search}%") ->orWhere('no_telepon', 'like', "%{\$search}%") ->orWhere('nip', 'like', "%{\$search}%") ->orWhere('username', 'like', "%{\$search}%"); }); } \$gurus = \$query ->orderBy('nama') ->paginate(15) ->withQueryString(); return view('guru.index', compact('gurus')); } </pre>
Deskripsi
Halaman ini menampilkan daftar seluruh pengguna dengan peran Guru yang terdaftar dalam sistem, meliputi kolom NIP, nama lengkap, alamat, username, nomor telepon, role akses, dan kelas yang diampu. Operator dapat mencari data guru berdasarkan nama atau NIP melalui kolom pencarian, serta menambahkan guru baru melalui tombol "Tambah Guru" yang akan mengarahkan ke halaman form tambah data.

Tabel 5.22 Antarmuka dan Kode Halaman Tambah Guru

Halaman Tambah Guru	
The screenshot shows a web application interface for adding a new user. On the left is a sidebar menu with options like Dashboard, Data Siswa, Data Kelas, Data Guru (highlighted), Data Periode, Rangkap Absensi, Notifikasi WhatsApp, and Log Aktivitas. The main area is titled 'Dashboard Operator' and contains a form titled 'Tambah Pengguna / Guru Baru'. The form has the following fields: NIP (with a sample value 197801012005011001), Nama Lengkap (with a sample value Ahmad Fauzi), Alamat Email (with a sample value guru@sekolah.sch.id), No. Telepon / WhatsApp (with a sample value 081234567890), Jenis Kelamin (a dropdown menu), Hak Akses (Role) (a dropdown menu), Alamat Tempat Tinggal (with a sample value Jl. Merdeka No. 10, Cibitung), Password Log In (with a note 'Minimal 8 karakter'), and Konfirmasi Password (with a note 'Ulangi password'). A blue button labeled 'SIMPAN DATA USER' is at the bottom right of the form. A 'Kembali' link is at the top right of the form.	

Gambar 5.5 Tampilan halaman Tambah Guru

Kode

```

public function update(Request $request, $id): RedirectResponse
{
    $user = User::findOrFail($id); //Cari user berdasarkan ID.
    $uniqueSuffix = ",{$user->id}";

    $data = $request->validate([
        'nip' => [
            'required',
            'numeric',
            Rule::unique('users', 'nip')
                ->where(fn($query) => $query->where('role', $request-
>input('role'))
                ->ignore($user), //membuat aturan unik NIP berdasarkan role.
        ],
        'username' =>
"required|string|alpha_dash|max:50|unique:users,username{$uniqueSuffix}",
        'nama' => 'required|string|max:255',
        'no_telepon' => [
            'required',
            'numeric',
            Rule::unique('users', 'no_telepon')->ignore($user), //Nomor
telepon harus unik lintas role.
        ],
        'max_digits:20',
    ],
        'alamat' => 'required|string|max:255',
        'role' => 'required|string|in:operator,guru,kepala_sekolah',
        'jenis_kelamin' => 'required|string|in:laki-laki,perempuan',
        'password' => $user ? 'nullable|string|min:8|confirmed' :
'required|string|min:8|confirmed',
        'kelas_id' => 'nullable|exists:kelas,id',
    ]);

    // Password kosong berarti password lama tetap dipertahankan.

```

Halaman Tambah Guru

```

        if (filled($data['password'] ?? null)) {
            $data['password'] = Hash::make($data['password']);
        } else {
            unset($data['password']);
        }

        // Kunci data operator dan user agar aturan akun terakhir aman dari
        request bersamaan.
        DB::transaction(function () use ($request, $user, $data): void {
            $operatorIds = User::query()
                ->where('role', 'operator')
                ->orderBy('id')
                ->lockForUpdate()
                ->pluck('id');
            $lockedUser = User::query()->whereKey($user->id)->lockForUpdate()-
            >firstOrFail();

            // Operator tunggal tetap boleh memperbarui data akunnya sendiri.
            // Hanya perubahan role yang menghilangkan operator terakhir yang
            ditolak.
            if ($lockedUser->role === 'operator' && $data['role'] !== 'operator'
            && $operatorIds->count() <= 1) {
                throw ValidationException::withMessages([
                    'role' => 'Minimal satu akun operator harus tetap
tersedia.',
                ]);
            }

            $oldData = $lockedUser->makeHidden('password')->toArray();
            $lockedUser->update($data);

            // Sinkronkan penugasan kelas setelah perubahan role atau kelas
            pengguna.
            if ($lockedUser->role === 'guru') {
                $currentKelasId = $lockedUser->kelas->id;
                $newKelasId = $request->filled('kelas_id') ? $request-
                >integer('kelas_id') : null;

                if ($currentKelasId && $currentKelasId !== $newKelasId) {
                    // Lepas kelas lama
                    Kelas::where('id', $currentKelasId)->update(['guru_id' =>
                    null]);
                }

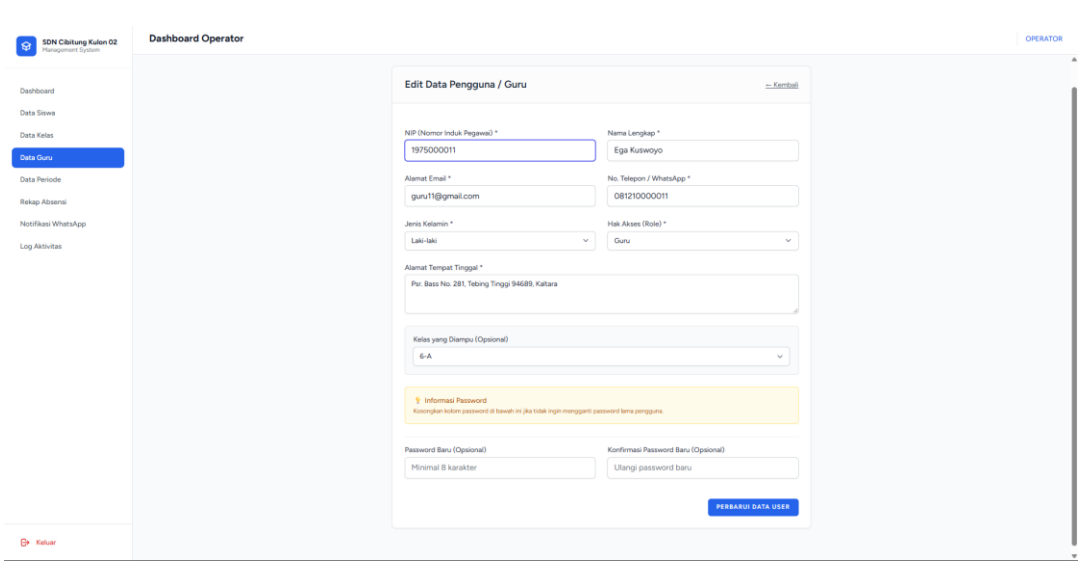
                if ($newKelasId && $currentKelasId !== $newKelasId) {
                    // Tugaskan kelas baru
                    $kelas = Kelas::where('id', $newKelasId)
                        ->where(function ($query) use ($lockedUser) {
                            $query->whereNull('guru_id')
                                ->orWhere('guru_id', $lockedUser->id);
                        })
                        ->lockForUpdate()
                        ->firstOrFail();
                    $kelas->update(['guru_id' => $lockedUser->id]);
                }
            } else {
                // Lepas kelas jika bukan guru
                Kelas::query()->where('guru_id', $lockedUser->id)-
                >update(['guru_id' => null]);
            }

            // Log aktivitas menggunakan trait
            $this->logUpdate(

```

Halaman Tambah Guru
<pre> 'Guru', \$lockedUser, ['old' => \$OldData, 'new' => \$lockedUser->fresh()- >makeHidden('password')->toArray()], "Memperbarui data pengguna: {\$lockedUser->nama}")); return redirect()->route('guru.index')->with('success', 'Data User berhasil diperbarui.');</pre>
Deskripsi
Halaman ini menyediakan form bagi operator untuk mendaftarkan akun guru baru, dengan input berupa NIP, nama lengkap, username, nomor telepon/WhatsApp, jenis kelamin, hak akses (role), alamat tempat tinggal, serta password login dan konfirmasi password. Seluruh field bertanda bintang (*) wajib diisi, dan sistem melakukan validasi kesesuaian antara password dan konfirmasi password sebelum data guru baru disimpan ke database.

Tabel 5.23 Antarmuka dan Kode Halaman Ubah Guru

Halaman Ubah Guru

Kode

Gambar 5.6 Tampilan halaman Ubah Guru

Halaman Ubah Guru

```

public function update(Request $request, $id): RedirectResponse
{
    $user = User::findOrFail($id); //Cari user berdasarkan ID.
    $uniqueSuffix = "{ $user->id}";

    $data = $request->validate([
        'nip' => [
            'required',
            'numeric',
            Rule::unique('users', 'nip')
                ->where(fn($query) => $query->where('role', $request-
>input('role'))
                ->ignore($user), //membuat aturan unik NIP berdasarkan role.
        ],
        'username' =>
"required|string|alpha_dash|max:50|unique:users,username{$uniqueSuffix}",
        'nama' => 'required|string|max:255',
        'no_telepon' => [
            'required',
            'numeric',
            Rule::unique('users', 'no_telepon')->ignore($user), //Nomor
telepon harus unik lintas role.
        ],
        'max_digits:20',
    ],
    'alamat' => 'required|string|max:255',
    'role' => 'required|string|in:operator,guru,kepala_sekolah',
    'jenis_kelamin' => 'required|string|in:laki-laki,perempuan',
    'password' => $user ? 'nullable|string|min:8|confirmed' :
'required|string|min:8|confirmed',
    'kelas_id' => 'nullable|exists:kelas,id',
    ]);

    // Password kosong berarti password lama tetap dipertahankan.
    if (filled($data['password']) ?? null) {
        $data['password'] = Hash::make($data['password']);
    } else {
        unset($data['password']);
    }

    // Kunci data operator dan user agar aturan akun terakhir aman dari
request bersamaan.
    DB::transaction(function () use ($request, $user, $data): void {
        $operatorIds = User::query()
            ->where('role', 'operator')
            ->orderBy('id')
            ->lockForUpdate()
            ->pluck('id');
        $lockedUser = User::query()->whereKey($user->id)->lockForUpdate()-
>firstOrFail();

        // Operator tunggal tetap boleh memperbarui data akunnya sendiri.
        // Hanya perubahan role yang menghilangkan operator terakhir yang
ditolak.
        if ($lockedUser->role === 'operator' && $data['role'] !== 'operator'
&& $operatorIds->count() <= 1) {
            throw ValidationException::withMessages([
                'role' => 'Minimal satu akun operator harus tetap
tersedia.',
            ]);
        }

        $oldData = $lockedUser->makeHidden('password')->toArray();
        $lockedUser->update($data);
    });
}

```

Halaman Ubah Guru
<pre> // Sinkronkan penugasan kelas setelah perubahan role atau kelas pengguna. if (\$lockedUser->role === 'guru') { \$currentKelasId = \$lockedUser->kelas->id; \$newKelasId = \$request->filled('kelas_id') ? \$request- >integer('kelas_id') : null; if (\$currentKelasId && \$currentKelasId !== \$newKelasId) { // Lepas kelas lama Kelas::where('id', \$currentKelasId)->update(['guru_id' => null]); } if (\$newKelasId && \$currentKelasId !== \$newKelasId) { // Tugaskan kelas baru \$kelas = Kelas::where('id', \$newKelasId) ->where(function (\$query) use (\$lockedUser) { \$query->whereNull('guru_id') ->orWhere('guru_id', \$lockedUser->id); }) ->lockForUpdate() ->firstOrFail(); \$kelas->update(['guru_id' => \$lockedUser->id]); } } else { // Lepas kelas jika bukan guru Kelas::query()->where('guru_id', \$lockedUser->id)- >update(['guru_id' => null]); } // Log aktivitas menggunakan trait \$this->logUpdate('Guru', \$lockedUser, ['old' => \$oldData, 'new' => \$lockedUser->fresh()- >makeHidden('password')->toArray()], "Memperbarui data pengguna: {\$lockedUser->nama}"); }); return redirect()->route('guru.index')->with('success', 'Data User berhasil diperbarui.');</pre>
Deskripsi
Halaman ini menampilkan form edit data guru yang sudah terisi dengan data guru yang dipilih sebelumnya (NIP, nama, username, no. telepon, jenis kelamin, role akses, alamat, dan kelas yang diampu). Operator dapat memperbarui data tersebut, sedangkan kolom password baru bersifat opsional jika dikosongkan, sistem akan tetap menggunakan password lama sesuai keterangan pada halaman.

Tabel 5.24 Antarmuka dan Kode Halaman Daftar Siswa

Halaman Daftar Siswa

Daftar Data Siswa

Search: Cari NIS, NISN, atau nama siswa [CARI] [PINDAH KELAS] [+ TAMBAH SISWA]

Kelas: Semua kelas [RESET]

NO	NIS / NISN	NAMA LENGKAP	KELAS	JENIS KELAMIN	ALAMAT	NAMA AYAH	NO. WA AYAH
1	20260007 / 0099000007	Alambana Danuja Irawan	1-A	Laki-Laki	Kpg. Sam Ratulangi No. 106, Semarang 69305, Kaltim	Prabu Umay Saragih	0813984319
2	20260023 / 0099000023	Bakda Makuta Zulkarnain S.H.	1-A	Laki-Laki	Par. Baik No. 383, Ternate 44966, Kalbar	Slamet Prasasta	0813984319
3	20260013 / 0099000013	Bakiman Saiful Siregar	1-A	Laki-Laki	Jln. Caji No. 629, Durnal 69035, Aceh	Prasetyo Kuswoyo	0813984319
4	20260017 / 0099000017	Cakrajiye Nasab Adriansyah	1-A	Laki-Laki	Kpg. Sukajadi No. 578, Sawahlunto 65337, Kalbar	Ajono lan Widodo S.Kom	0813984319
5	20260030 / 0099000030	Calista Winarsh	1-A	Perempuan	Gg. Padang No. 131, Lhokseumawe 24016, Sumut	Yosef Pratama S.Ked	0813984319

Dk. Sudirman

Gambar 5.7 Tampilan halaman Daftar Siswa

Kode

```

public function index(Request $request): View
{
    $filters = $request->validate([
        'search' => ['nullable', 'string', 'max:100'],
        'kelas_id' => ['nullable', 'integer', 'exists:kelas,id'],
    ]);

    $siswas = Siswa::query()
        ->select([
            'id',
            'nis',
            'nisn',
            'nama_siswa',
            'jenis_kelamin',
            'alamat',
            'nama_ayah',
            'no_whatsapp_ayah',
            'nama_ibu',
            'no_whatsapp_ibu',
            'kelas_id',
        ])
        ->with([
            'kelas:id,nama_kelas',
        ])
        ->whereHas('kelas', fn($query) => $query->accessibleBy($request->user())) //membatasi akses siswa hanya untuk kelas yang dapat diakses oleh user (guru/operator)
        ->when($filters['search'] ?? null, function ($query, string $search): void {
            //jalankan filter pencarian jika ada parameter search
            $query->where(function ($query) use ($search): void {
                $query->where('nama_siswa', 'like', "%{$search}%")
            });
        });
}

```

Halaman Daftar Siswa
<pre> ->orWhere('nis', 'like', "%{\$search}%") ->orWhere('nisn', 'like', "%{\$search}%"); }); }) ->when(\$filters['kelas_id'] ?? null, fn(\$query, \$kelasId) => \$query- >where('kelas_id', \$kelasId)) //filter berdasarkan kelas jika ada parameter kelas_id ->orderBy('nama_siswa') ->paginate(15) ->withQueryString(); return view('siswa.index', ['siswas' => \$siswas, ...\$this->formOptions(\$request->user()),]); } </pre>
Deskripsi
Halaman ini menampilkan daftar seluruh data siswa yang terdaftar, meliputi NIS/NISN, nama lengkap, kelas, jenis kelamin, alamat, nama ayah, dan nomor WhatsApp orang tua. Operator dapat memfilter data berdasarkan kelas, mencari siswa melalui NIS/NISN/nama, mengubah kelas siswa secara massal melalui tombol "Ubah Kelas", maupun menambahkan siswa baru melalui tombol "Tambah Siswa".

Tabel 5.25 Antarmuka dan Kode Halaman Tambah Siswa

Halaman Tambah Siswa	
SDN Cibitung Kulon 02 Management System Dashboard Operator Dashboard Data Siswa Data Kelas Data Guru Data Periode Rekap Absensi Notifikasi WhatsApp Log Aktivitas Keluar	Tambah Data Siswa Baru -- Kembali A. DATA PRIBADI SISWA NIS * Contoh: 20260001 NISN * Contoh: 0099123456 Nama Lengkap Siswa * Contoh: Ahmad Fauzi Kelas * -- Pilih Kelas -- Jenis Kelamin * -- Pilih Jenis Kelamin -- Alamat Rumah * Contoh: Jl. Merdeka No. 10, Cibitung B. DATA ORANG TUA Nama Ayah * Contoh: Budi Santoso No. WhatsApp Ayah (Optional) Contoh: 08123456789 Nama Ibu * Contoh: Siti Aminah No. WhatsApp Ibu (Optional) Contoh: 08123456789 Minimal salah satu nomor WhatsApp ayah atau ibu wajib diisi. SIMPAN DATA SISWA

Gambar 5.8 Tampilan halaman Tambah Siswa

Kode

```

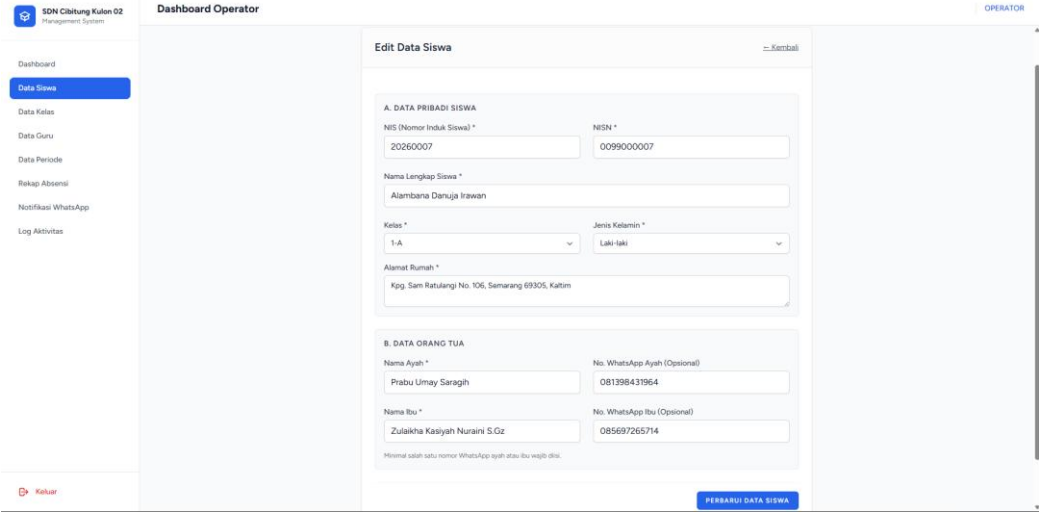
public function store(Request $request): RedirectResponse
{
    $data = $request->validate([
        'nis' => [
            'required',
            'numeric',
            'max_digits:50',
            Rule::unique('siswas', 'nis'),
        ],
        'nisn' => [
            'required',
            'numeric',
            'max_digits:50',
            Rule::unique('siswas', 'nisn'),
        ],
        'nama_siswa' => ['required', 'string', 'max:255'],
        'jenis_kelamin' => ['required', Rule::in(['laki-laki',
'perempuan'])],
        'nama_ayah' => ['required', 'string', 'max:255'],
        'no_whatsapp_ayah' => ['nullable', 'numeric', 'max_digits:20'],
        'nama_ibu' => ['required', 'string', 'max:255'],
        'no_whatsapp_ibu' => ['nullable', 'numeric', 'max_digits:20'],
        'kelas_id' => ['required', 'integer', 'exists:kelas,id'],
        'alamat' => ['required', 'string'],
    ]);

    // Minimal salah satu nomor WA orang tua wajib diisi
    if (blank($data['no_whatsapp_ayah']) && blank($data['no_whatsapp_ibu']))
    {
        throw ValidationException::withMessages([

```

Halaman Tambah Siswa
<pre> 'no_whatsapp_ayah' => 'Minimal salah satu nomor WhatsApp orang tua (ayah atau ibu) wajib diisi.',]); } \$data['kelas_id'] = \$this->findKelas((int) \$data['kelas_id'], \$request- >user())->id; \$siswa = Siswa::create(\$data); // Log aktivitas menggunakan trait \$this->logCreate('Siswa', \$siswa, "Menambahkan siswa baru: {\$siswa- >nama_siswa}", ['siswa' => \$siswa->toArray()]); return to_route('siswa.index') ->with('success', 'Data siswa berhasil ditambahkan.');</pre>
Deskripsi
Halaman ini menyediakan form pendaftaran data siswa baru yang terbagi menjadi dua bagian: data pribadi siswa (NIS, NISN, nama lengkap, kelas, jenis kelamin, dan alamat rumah) serta data orang tua (nama ayah, nama ibu, dan nomor WhatsApp masing-masing). Sistem mewajibkan minimal salah satu nomor WhatsApp orang tua (ayah atau ibu) untuk diisi, karena nomor tersebut digunakan sebagai tujuan pengiriman pesan absensi.

Tabel 5.26 Antarmuka dan Kode Halaman Ubah Siswa

Halaman Ubah Siswa

Kode

Halaman Ubah Siswa

```

public function update(Request $request, Siswa $siswa): RedirectResponse
{
    $this->authorizeSiswaAccess($siswa, $request->user());

    $data = $request->validate([
        'nis' => [
            'required',
            'numeric',
            'max_digits:50',
            Rule::unique('siswas', 'nis')->ignore($siswa),
        ],
        'nisn' => [
            'required',
            'numeric',
            'max_digits:50',
            Rule::unique('siswas', 'nisn')->ignore($siswa),
        ],
        'nama_siswa' => ['required', 'string', 'max:255'],
        'jenis_kelamin' => ['required', Rule::in(['laki-laki',
'perempuan'])],
        'nama_ayah' => ['required', 'string', 'max:255'],
        'no_whatsapp_ayah' => ['nullable', 'numeric', 'max_digits:20'],
        'nama_ibu' => ['required', 'string', 'max:255'],
        'no_whatsapp_ibu' => ['nullable', 'numeric', 'max_digits:20'],
        'kelas_id' => ['required', 'integer', 'exists:kelas,id'],
        'alamat' => ['required', 'string'],
    ]);

    // Minimal salah satu nomor WA orang tua wajib diisi
    if (blank($data['no_whatsapp_ayah']) &&
blank($data['no_whatsapp_ibu'])) {
        throw ValidationException::withMessages([
            'no_whatsapp_ayah' => 'Minimal salah satu nomor WhatsApp orang
tua (ayah atau ibu) wajib diisi.',
        ]);
    }

    $kelasSelected = $this->findKelas((int) $data['kelas_id'], $request-
>user()); //kelas mana yang dipilih dan apakah user berhak mengaksesnya.

    // Siswa tidak bisa pindah kelas jika sudah punya data absensi
    if ($siswa->kelas_id !== $kelasSelected->id && $siswa->hasAbsensi()) {
        return redirect()->back()
            ->withInput() //Mempertahankan input yang tadi dikirim.
            ->withErrors(['kelas_id' => 'Siswa tidak bisa dipindahkan
kelas karena sudah memiliki data absensi.']);
    }

    $data['kelas_id'] = $kelasSelected->id;

    $oldData = $siswa->toArray();
    $siswa->update($data);

    // Log aktivitas menggunakan trait
    $this->logUpdate('Siswa', $siswa, ['old' => $oldData, 'new' => $siswa-
>fresh()->toArray()], "Memperbarui data siswa: {$siswa->nama_siswa}");

    return to_route('siswa.index')
        ->with('success', 'Data siswa berhasil diperbarui.');
```

Deskripsi

Halaman Ubah Siswa

Halaman ini menampilkan form edit data siswa yang telah terisi dengan data eksisting (NIS, NISN, nama, kelas, jenis kelamin, alamat rumah, serta data orang tua). Operator dapat memperbarui setiap field yang diperlukan, dengan validasi yang sama seperti pada halaman tambah siswa, yaitu minimal salah satu nomor WhatsApp orang tua wajib terisi.

Tabel 5.27 Antarmuka dan Kode Halaman Daftar Kelas

Halaman Daftar Kelas

SDN Cibitung Kulon 02

- Dashboard
- Data Siswa
- Data Kelas**
- Data Guru
- Data Periode
- Rekap Absensi
- Notifikasi WhatsApp
- Log Aktivitas

Keluar

Dashboard Operator OPERATOR

Daftar Data Kelas

CARI
+ TAMBAH KELAS

NO	NAMA KELAS	WALI KELAS	AKSI
1	1-A	Muhammad Azhar Utama	Edit Hapus
2	1-B	Lintang Tri Nasyidah S.Pai	Edit Hapus
3	2-A	Gilang Hardiansyah	Edit Hapus
4	2-B	Citra Riyanti S.Farm	Edit Hapus
5	3-A	Hartika Pradana	Edit Hapus
6	3-B	Yessi Novitasari S.H.	Edit Hapus
7	4-A	Benawi Indra Hardiansyah	Edit Hapus
8	4-B	Hilda Hasanah	Edit Hapus
9	5-A	Teddy Marpaung	Edit Hapus
10	5-B	Uri Ida Uyainah	Edit Hapus

Gambar 5.10 Tampilan halaman Daftar Kelas

Kode

```
public function index(Request $request)
{
    $filters = $request->validate([
        'search' => ['nullable', 'string', 'max:100'],
    ]);
    $search = trim($filters['search'] ?? '');

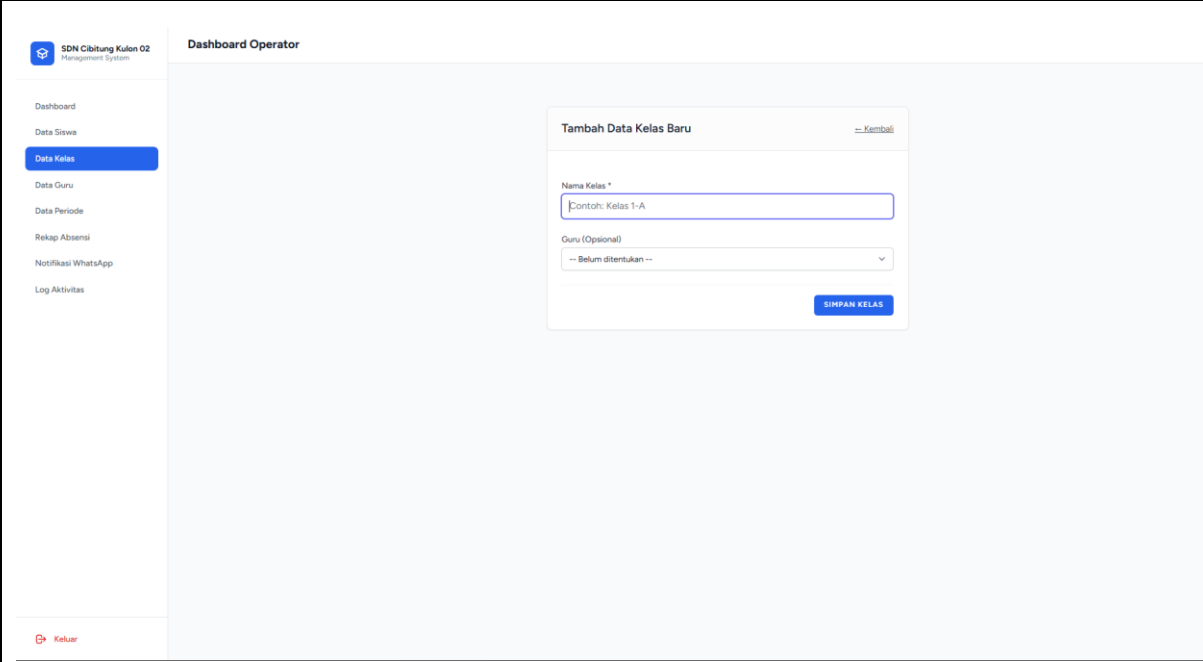
    $query = Kelas::query()
        ->select(['id', 'nama_kelas', 'status', 'guru_id'])
        ->with('guru:id,nama,nip');

    if ($search !== '') {
        $query->where('nama_kelas', 'like', "%{$search}%");
    }

    $kelas = $query
        ->orderBy('nama_kelas')
        ->paginate(15)
```

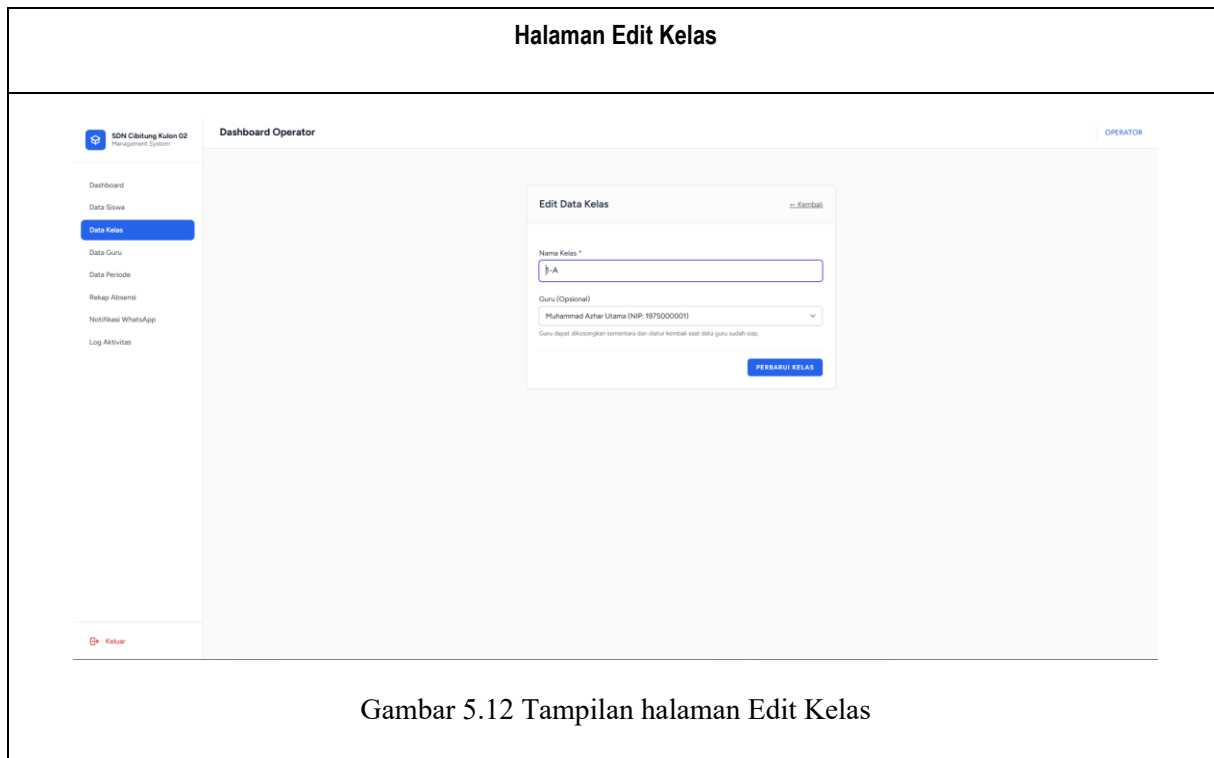
Halaman Daftar Kelas
<pre>->withQueryString(); return view('kelas.index', compact('kelas')); }</pre>
Deskripsi
Halaman ini menampilkan daftar seluruh kelas yang terdaftar dalam sistem beserta nama wali kelas yang mengampu masing-masing kelas. Operator dapat mencari kelas melalui kolom pencarian, menambah kelas baru melalui tombol "Tambah Kelas", serta mengedit atau menghapus data kelas melalui tombol aksi "Edit" dan "Hapus" pada setiap baris.

Tabel 5.28 Antarmuka dan Kode Halaman Tambah Kelas

Halaman Tambah Kelas

Gambar 5.11 Tampilan halaman Tambah Kelas
Kode

Halaman Tambah Kelas
<pre> public function store(Request \$request) { \$request->validate(['nama_kelas' => ['required', 'string', 'max:50', Rule::unique('kelas', 'nama_kelas'),], 'guru_id' => 'nullable exists:users,id',]); if (\$request->filled('guru_id') && ! \$this->availableGuruQuery()- >where('id', \$request->guru_id)->exists()) { return redirect()->back() ->withInput() ->withErrors(['guru_id' => 'Guru yang dipilih tidak tersedia (sudah memiliki kelas).']); } // Kunci guru dan validasi ulang di dalam transaksi untuk mencegah penugasan ganda. \$kelas = DB::transaction(function () use (\$request): Kelas { if (\$request->filled('guru_id')) { User::query()->whereKey(\$request->integer('guru_id'))- >lockForUpdate()->firstOrFail(); if (! \$this->availableGuruQuery()->whereKey(\$request- >integer('guru_id'))->exists()) { throw ValidationException::withMessages(['guru_id' => 'Guru yang dipilih sudah memiliki kelas.',]); } } \$kelas = Kelas::create(['nama_kelas' => \$request->string('nama_kelas')->trim()- >toString(), 'status' => 'aktif', 'guru_id' => \$request->filled('guru_id') ? \$request- >integer('guru_id') : null,]); return \$kelas; }); \$this->logCreate('Kelas', \$kelas, "Menambahkan kelas baru: {\$kelas- >nama_kelas}"); return redirect()->route('kelas.index')->with('success', 'Data Kelas berhasil ditambahkan.');</pre>
Deskripsi
Halaman ini menyediakan form sederhana untuk menambahkan data kelas baru, terdiri dari input nama kelas (wajib diisi) dan pemilihan guru wali kelas yang bersifat opsional. Data kelas akan tersimpan setelah operator menekan tombol "Simpan Kelas".

Tabel 5.29 Antarmuka dan Kode Halaman Edit Kelas



Gambar 5.12 Tampilan halaman Edit Kelas

Kode

```

public function update(Request $request, $id)
{
    $kelas = Kelas::findOrFail($id);

    $request->validate([
        'nama_kelas' => [
            'required',
            'string',
            'max:50',
            Rule::unique('kelas', 'nama_kelas')->ignore($kelas->id),
        ],
        'guru_id' => 'nullable|exists:users,id',
    ]);

    $currentGuruId = $kelas->guru_id;
    $oldData = $kelas->toArray();

    if (
        $request->filled('guru_id')
        && ! $this->guruOptionsQuery($currentGuruId)->where('id', $request-
>guru_id)->exists() //Tujuannya memastikan guru yang dipilih memang boleh
ditugaskan.
    ) {
        return redirect()->back()
            ->withInput()
            ->withErrors(['guru_id' => 'Guru yang dipilih tidak tersedia
(sudah memiliki kelas).']);
    }

    // Penguncian menjaga penugasan guru tetap konsisten saat ada request
bersamaan.
    DB::transaction(function () use ($currentGuruId, $kelas, $request): void {

```

Halaman Edit Kelas
<pre> \$lockedKelas = Kelas::query()->whereKey(\$kelas->id)->lockForUpdate()- >firstOrFail(); //Kelas yang sedang diedit dikunci sementara. if (\$request->filled('guru_id')) { User::query()->whereKey(\$request->integer('guru_id'))- >lockForUpdate()->firstOrFail(); if (! \$this->guruOptionsQuery(\$currentGuruId)->whereKey(\$request- >integer('guru_id'))->exists()) { throw ValidationException::withMessages(['guru_id' => 'Guru yang dipilih sudah memiliki kelas.',]); } } //Update data kelas dengan data baru yang diterima dari request. \$lockedKelas->update(['nama_kelas' => \$request->string('nama_kelas')->trim()- >toString(), 'guru_id' => \$request->filled('guru_id') ? \$request- >integer('guru_id') : null,]); \$kelas->refresh(); //Ini meminta Laravel mengambil ulang data terbaru dari database. \$this->logUpdate('Kelas', \$kelas, ['old' => \$oldData, 'new' => \$kelas->toArray()], "Memperbarui data kelas: {\$kelas->nama_kelas}"); return redirect()->route('kelas.index')->with('success', 'Data Kelas berhasil diperbarui.');</pre>
Deskripsi
Halaman ini menampilkan form edit data kelas yang telah terisi dengan nama kelas dan guru wali kelas yang sedang menjabat saat ini. Operator dapat mengubah nama kelas maupun mengganti wali kelas, termasuk mengosongkannya sementara sesuai keterangan pada halaman, apabila data guru pengganti belum siap.

Tabel 5.30 Antarmuka dan Kode Halaman Periode

Halaman Periode

Gambar 5.13 Tampilan halaman Periode

Kode

```

public function index(Request $request)
{
    // Ambil semester beserta relasi hari libur dalam satu query.
    $periodes = Periode::query()
        ->with('hariLiburs')
        ->get();

    // Pisahkan data semester agar masing-masing tanggal dapat ditampilkan di
    form.
    $semester1 = $periodes->firstWhere('semester', 1);
    $semester2 = $periodes->firstWhere('semester', 2);

    // Siapkan nilai ISO untuk input HTML date dan pertahankan nilai lama saat
    validasi gagal.
    $periodeData = [
        'tahun_ajaran' => $semester1?->tahun_ajaran ?? $semester2?->
        tahun_ajaran ?? old('tahun_ajaran', ''), //?-> adalah null safe operator, jika
        $semester1 null maka akan mengecek $semester2, jika keduanya null maka akan
        menggunakan old('tahun_ajaran', '').
        'semester_1_tanggal_mulai' => $semester1?->tanggal_mulai?->format('Y-
        m-d') ?? old('semester_1_tanggal_mulai', ''),
        'semester_1_tanggal_selesai' => $semester1?->tanggal_selesai?->
        format('Y-m-d') ?? old('semester_1_tanggal_selesai', ''),
        'semester_2_tanggal_mulai' => $semester2?->tanggal_mulai?->format('Y-
        m-d') ?? old('semester_2_tanggal_mulai', ''),
        'semester_2_tanggal_selesai' => $semester2?->tanggal_selesai?->
        format('Y-m-d') ?? old('semester_2_tanggal_selesai', ''),
    ];

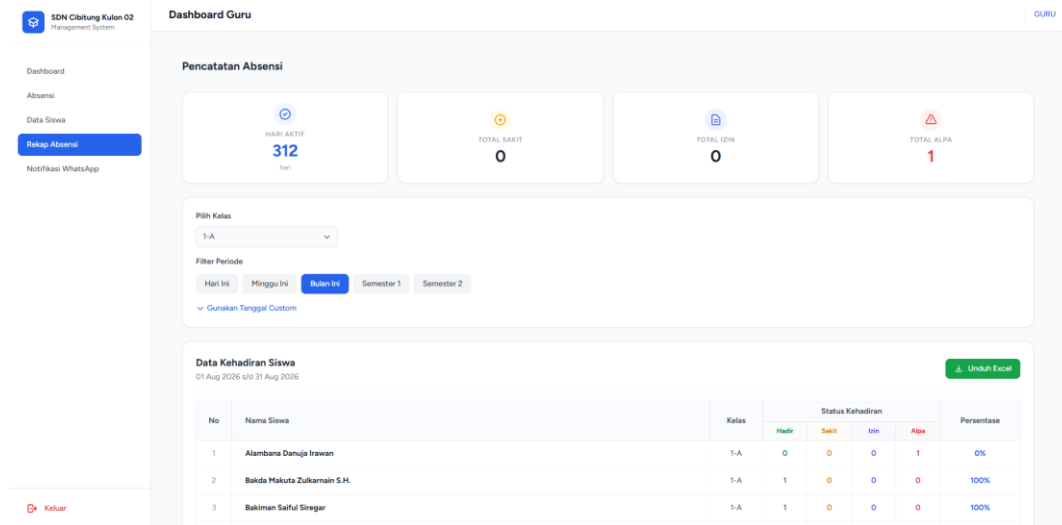
    // Siapkan format tanggal Indonesia untuk teks tampilan yang mudah dibaca
    pengguna.
    $periodeDataDisplay = [
        'semester_1_tanggal_mulai' => $semester1?->tanggal_mulai?->
        format('d/m/Y'),

```

Halaman Periode
<pre> 'semester_1_tanggal_selesai' => \$semester1?->tanggal_selesai?- >format('d/m/Y'), 'semester_2_tanggal_mulai' => \$semester2?->tanggal_mulai?- >format('d/m/Y'), 'semester_2_tanggal_selesai' => \$semester2?->tanggal_selesai?- >format('d/m/Y'),]; \$liburMingguan = collect(); \$liburNasional = collect(); // Gunakan semester 1 sebagai sumber tampilan awal; jika belum ada, gunakan semester 2. \$periode = \$semester1 ?? \$semester2; //Gunakan Semester 1 jika tersedia. Kalau tidak, gunakan Semester 2. if (\$periode) { // Pisahkan hari libur mingguan dan nasional agar sesuai dengan dua bagian form. \$liburMingguan = \$periode->hariLiburs //Mengambil libur mingguan ->where('tipe', 'mingguan') ->map(fn(\$item) => ['hari' => \$item->hari, 'keterangan' => \$item->keterangan,]); \$liburNasional = \$periode->hariLiburs //Mengambil libur nasional ->where('tipe', 'nasional') ->map(fn(\$item) => ['tanggal' => \$item->tanggal?->format('Y-m-d') ?? '', // Format Y-m-d untuk input HTML5 date 'nama_libur' => \$item->nama_libur ? : \$item->keterangan, 'keterangan' => \$item->nama_libur ? \$item->keterangan : '',]); } // Kirim konfigurasi periode dan hari libur ke halaman pengaturan akademik. return view('periode.index', compact('periode', 'periodeData', 'periodeDataDisplay', 'liburMingguan', 'liburNasional')); } </pre>
Deskripsi
Halaman ini digunakan operator untuk mengatur periode tahun ajaran aktif, meliputi tanggal mulai dan selesai untuk Semester 1 (Ganjil) dan Semester 2 (Genap), serta pengaturan hari libur mingguan dan hari libur nasional yang akan memengaruhi perhitungan hari aktif absensi. Tersedia pula fitur "Reset Periode" yang akan menghapus seluruh data periode, absensi, dan rekap terkait, sehingga penggunaannya memerlukan kehati-hatian.

Tabel 5.31 Antarmuka dan Kode Halaman Rekap Absensi

Halaman Rekap Absensi



Gambar 5.14 Tampilan halaman Rekap Absensi

Kode

```
private function rekapData(Request $request, ?array $filters = null, ?Collection $kelas = null): array
{
    // Export dapat memanggil method ini tanpa membawa hasil validasi dari index(),
    // sehingga request tetap divalidasi di sini sebagai pengaman.
    $filters ??= $request->validate([
        'kelas_id' => ['nullable', 'integer', 'exists:kelas,id'],
        'preset' => ['nullable', 'string',
            'in:today,this_week,this_month,semester_1,semester_2,custom'],
        'tanggal_mulai' => ['nullable', 'date'],
        'tanggal_berakhir' => ['nullable', 'date',
            'after_or_equal:tanggal_mulai'],
    ]);

    // Gunakan daftar kelas yang sudah tersedia atau ambil ulang berdasarkan akses user.
    $kelas ??= $this->accessibleKelas($request);

    // Gunakan bulan berjalan sebagai preset default jika pengguna belum memilih preset.
    $preset = $filters['preset'] ?? 'this_month';

    // Trait memilih kelas otomatis ketika user hanya memiliki satu kelas yang accessible.
    $kelasId = $this->getKelasIdWithAutoSelect($filters['kelas_id'] ?? null, $kelas);

    // Tentukan rentang laporan dari tanggal custom atau preset bawaan aplikasi.
    if ($preset === 'custom') {
        // Preset custom memakai tanggal manual, dengan fallback ke bulan berjalan.
        $tanggalMulaiInput = $filters['tanggal_mulai'] ?? today()->startOfMonth()->format('d/m/Y');
```

Halaman Rekap Absensi

```

        $tanggalBerakhirInput = $filters['tanggal_berakhir'] ?? today()-
>format('d/m/Y');

        // Normalisasi tanggal ke format database Y-m-d sebelum query.
        $tanggalMulai = $this->parseDate($tanggalMulaiInput)->format('Y-m-d');
        $tanggalBerakhir = $this->parseDate($tanggalBerakhirInput)->format('Y-
m-d');
    } else {
        // Preset otomatis menentukan tanggal berdasarkan hari, minggu, bulan,
atau semester.
        [$tanggalMulai, $tanggalBerakhir] = $this-
>getPresetDateRange($preset);
    }

    // Siapkan wadah hasil rekap dan state tampilan sebelum kelas diproses.
    $rekapSiswa = []; // Array rekap per siswa
    $namaKelas = null; // Nama kelas yang dipilih
    $totalHariAktif = 0; // Total hari aktif dalam 1 tahun ajaran (semester 1
+ 2)
    $totalHariAbsensi = 0; // Total hari yang ada data absensi (dalam range
filter)
    $hideRekapTabel = false; // Flag apakah tabel disembunyikan (jika tidak
ada data)

    // Statistik dimulai dari nol agar view tetap memiliki struktur data yang
konsisten.
    $stats = [
        'rata_hadir' => 0, // Rata-rata persentase kehadiran kelas
        'total_sakit' => 0,
        'total_izin' => 0,
        'total_alpa' => 0,
    ];

    if ($kelasId) {
        // Pastikan kelas yang diminta memang termasuk daftar kelas yang boleh
diakses user.
        $selectedKelas = $kelas->firstWhere('id', (int) $kelasId);

        abort_if($selectedKelas === null, 404);

        // Ambil tanggal sekolah aktif setelah mengeluarkan hari libur
nasional dan mingguan.
        $activeDates = $this->getActiveDatesForRange($tanggalMulai,
$tanggalBerakhir);

        // Ambil siswa kelas, termasuk siswa yang memiliki riwayat absensi di
rentang laporan.
        $siswas = Siswa::query()
            ->select(['id', 'nama_siswa'])
            ->where(function ($query) use ($kelasId, $tanggalBerakhir,
$tanggalMulai): void {
                $query->where('kelas_id', $kelasId)
                    ->orWhereHas('absensis', function ($query) use ($kelasId,
$tanggalBerakhir, $tanggalMulai): void {
                        $query->where('kelas_id', $kelasId)
                            ->whereBetween('tanggal', [$tanggalMulai,
$tanggalBerakhir]);
                    });
            });
        ->orderBy('nama_siswa')
        ->get();
    }

```

Halaman Rekap Absensi

```

        // Hitung jumlah setiap status langsung di database agar tidak
memproses seluruh
        // baris absensi satu per satu di PHP.
$absensiTotals = Absensi::query()
    ->select('siswa_id')
    ->selectRaw("SUM(CASE WHEN status = 'hadir' THEN 1 ELSE 0 END) AS
hadir")
    ->selectRaw("SUM(CASE WHEN status = 'sakit' THEN 1 ELSE 0 END) AS
sakit")
    ->selectRaw("SUM(CASE WHEN status = 'izin' THEN 1 ELSE 0 END) AS
izin")
    ->selectRaw("SUM(CASE WHEN status = 'alpa' THEN 1 ELSE 0 END) AS
alpa")
    ->whereBetween('tanggal', [$tanggalMulai, $tanggalBerakhir])
    ->whereIn('tanggal', $activeDates)
    ->whereIn('siswa_id', $siswas->pluck('id'))
    ->groupBy('siswa_id')
    ->get()
    ->keyBy('siswa_id');

siswa.
    // Jumlah hari aktif pada filter menjadi penyebut persentase setiap
$totalHariAktifFilter = count($activeDates);

    // Hitung juga total hari aktif seluruh periode untuk kebutuhan
ringkasan laporan.
$totalHariAktif = $this->hitungHariAktifPeriode();

$totalHariAbsensi = Absensi::query()
    ->where('kelas_id', $kelasId)
    ->whereBetween('tanggal', [$tanggalMulai, $tanggalBerakhir])
    ->whereIn('tanggal', $activeDates)
    ->distinct() //Mencegah tanggal yang sama dihitung berkali-kali.
    ->count('tanggal');

    // Siapkan akumulator statistik seluruh siswa dalam kelas.
$totalPersentaseSemuaSiswa = 0;
$totalHadir = 0;
$totalSakit = 0;
$totalIzin = 0;
$totalAlpa = 0;
$totalTidakMasuk = 0;
$namaKelas = $selectedKelas->nama_kelas;
$jumlahSiswa = $siswas->count();

foreach ($siswas as $siswa) {
    // Siswa tanpa absensi tetap dimasukkan ke laporan dengan nilai
status nol.
    $totals = $absensiTotals->get($siswa->id);
    $hadir = $totals->hadir ?? 0; //Jika tidak ditemukan data,
nilainya:0
    $sakit = $totals->sakit ?? 0; //?? adalah null coalescing
operator.Artinya:kalau nilai kiri tidak ada/null→ gunakan nilai kanan
    $izin = $totals->izin ?? 0;
    $alpa = $totals->alpa ?? 0;

    $tidakMasuk = $sakit + $izin + $alpa; //Menghitung tidak masuk
    $persentase = $totalHariAktifFilter > 0 ? round(($hadir /
$totalHariAktifFilter) * 100, 1) : 0; //Menghitung persentase siswa
    $persentase = min($persentase, 100);

    // Bentuk satu baris rekap yang dipakai oleh view dan export
Excel.

```

Halaman Rekap Absensi

```

$rekapSiswa[] = [
    'nama_siswa' => $siswa->nama_siswa,
    'nama_kelas' => $namaKelas,
    'hadir' => $hadir,
    'sakit' => $sakit,
    'izin' => $izin,
    'alpa' => $alpa,
    'total_tidak_masuk' => $tidakMasuk,
    'persentase' => $persentase,
];

$totalPersentaseSemuaSiswa += $persentase;

// Tambahkan hasil siswa ke statistik keseluruhan kelas.
$totalHadir += $hadir; // $totalHadir = $totalHadir + $hadir;
$totalSakit += $sakit;
$totalIzin += $izin;
$totalAlpa += $alpa;
$totalTidakMasuk += $tidakMasuk;
}

// Total kesempatan hadir kelas = jumlah hari aktif dikali jumlah
siswa.
$totalHariKerjaKelas = $totalHariAktifFilter * $jumlahSiswa;

// Rata-rata kelas dihitung dari persentase masing-masing siswa.
if ($jumlahSiswa > 0) {
    $stats['rata_hadir'] = round($totalPersentaseSemuaSiswa /
$jumlahSiswa, 1);
}

// Persentase keseluruhan memakai total kesempatan hadir seluruh kelas
sebagai penyebut.
$stats['total_hadir'] = $totalHadir;
$stats['total_sakit'] = $totalSakit;
$stats['total_izin'] = $totalIzin;
$stats['total_alpa'] = $totalAlpa;
$stats['total_tidak_masuk'] = $totalTidakMasuk;
$stats['persentase_hadir'] = $totalHariKerjaKelas > 0 ?
min(round(($totalHadir / $totalHariKerjaKelas) * 100, 1), 100) : 0;
$stats['persentase_sakit'] = $totalHariKerjaKelas > 0 ?
min(round(($totalSakit / $totalHariKerjaKelas) * 100, 1), 100) : 0;
$stats['persentase_izin'] = $totalHariKerjaKelas > 0 ?
min(round(($totalIzin / $totalHariKerjaKelas) * 100, 1), 100) : 0;
$stats['persentase_alpa'] = $totalHariKerjaKelas > 0 ?
min(round(($totalAlpa / $totalHariKerjaKelas) * 100, 1), 100) : 0;
$stats['persentase_tidak_masuk'] = $totalHariKerjaKelas > 0
? min(round(($totalTidakMasuk / $totalHariKerjaKelas) * 100, 1),
100)
: 0;
}

// Sembunyikan tabel jika kelas dipilih tetapi belum ada absensi dalam
rentang tersebut.
if ($kelasId && $totalHariAbsensi === 0) {
    $hideRekapTabel = true;
}

// Pertahankan format ISO untuk query/export dan format Indonesia untuk
tampilan.
$tanggalMulaiDisplay = Carbon::parse($tanggalMulai)->format('d/m/Y');
$tanggalBerakhirDisplay = Carbon::parse($tanggalBerakhir)-
>format('d/m/Y');

```

Halaman Rekap Absensi
<pre>// Kembalikan seluruh data yang dibutuhkan view rekap dan proses export. return compact('kelas', 'rekapSiswa', 'totalHariAktif', 'totalHariAbsensi', 'kelasId', 'tanggalMulai', 'tanggalBerakhir', 'tanggalMulaiDisplay', 'tanggalBerakhirDisplay', 'stats', 'namaKelas', 'preset', 'hideRekapTabel');</pre>
Deskripsi
Halaman ini menampilkan ringkasan rekap kehadiran siswa berupa kartu statistik (hari aktif, total sakit, total izin, total alpa) serta tabel rincian kehadiran per siswa (hadir, sakit, izin, alpa, dan persentase kehadiran). Guru dapat memfilter data berdasarkan kelas dan rentang periode (hari ini, minggu ini, bulan ini, semester, atau tanggal custom), serta mengunduh laporan dalam format Excel melalui tombol "Unduh Excel".

Tabel 5.32 Antarmuka dan Kode Halaman Hasil Ekspor Laporan Excel

Halaman Hasil Ekspor Laporan Excel

The screenshot displays the 'Dashboard Guru' (Teacher Dashboard) for 'SDN Cibitung Kulon 02'. The interface includes a sidebar with navigation links: Dashboard, Absensi, Data Siswa, Rekap Absensi (highlighted), and Notifikasi WhatsApp. The main content area is titled 'Pencatatan Absensi' and features four summary cards: 'HARI AKTIF' (312 hari), 'TOTAL SAKIT' (0), 'TOTAL IZIN' (0), and 'TOTAL ALPA' (1). Below these are filters for 'Pilih Kelas' (set to 1-A) and 'Filter Periode' (options: Hari Ini, Minggu Ini, Bulan Ini, Semester 1, Semester 2, and a link to 'Gunakan Tanggal Custom'). A table titled 'Data Kehadiran Siswa' for the period '01 Aug 2025 s/d 31 Aug 2025' shows attendance for three students. A green 'Unduh Excel' button is located at the top right of the table.

No	Nama Siswa	Kelas	Status Kehadiran				Persentase
			Hadir	Sakit	Izin	Alpa	
1	Alambana Danuja Irawan	1-A	0	0	0	1	0%
2	Bakda Makuta Zulkarnain S.H.	1-A	1	0	0	0	100%
3	Bakiman Saiful Sinegar	1-A	1	0	0	0	100%

Gambar 5.15 Tampilan halaman Hasil Ekspor Laporan Excel

Kode

Halaman Hasil Ekspor Laporan Excel

```

public function export(Request $request): BinaryFileResponse
{
    // Gunakan pipeline data yang sama dengan halaman rekap agar hasil export
    konsisten.
    $data = $this->rekapData($request);

    // File tidak dapat dibuat jika pengguna belum memilih kelas.
    abort_unless($data['kelasId'], 422, 'Pilih kelas terlebih dahulu sebelum
    mengunduh rekap.');
```

// Bentuk nama file yang aman menggunakan slug kelas dan rentang tanggal
 laporan.
 \$filename = sprintf(
 'rekap-absensi-%s-%s-sampai-%s.xlsx',
 Str::slug(\$data['namaKelas']), // Slug untuk safe filename (spasi jadi
 -)
 \$data['tanggalMulai'], // Format Y-m-d
 \$data['tanggalBerakhir'],
);

 // Delegasikan pembuatan workbook ke class export khusus.
 return Excel::download(
 new RekapAbsensiExport(
 \$data['rekapSiswa'], // Data per siswa
 \$data['namaKelas'], // Nama kelas untuk header
 \$data['tanggalMulai'], // Range tanggal untuk header
 \$data['tanggalBerakhir'],
 count(\$this->getActiveDatesForRange(\$data['tanggalMulai'],
 \$data['tanggalBerakhir'])), // Total hari aktif
),
 \$filename,
);
}
// app/Exports/RekapAbsensiExport.php
class RekapAbsensiExport implements FromArray, WithColumnWidths, WithStyles,
WithTitle
{
 public function array(): array
 {
 \$rows = [
 ['REKAP ABSENSI SISWA'],
 ['Kelas', \$this->namaKelas],
 ['Periode', \$this->periodeLabel()],
 [],
 ['No', 'Nama Siswa', 'Kelas', 'Hadir', 'Sakit', 'Izin', 'Alpa',
 'Persentase'],
];

 foreach (\$this->rekapSiswa as \$index => \$rekap) {
 \$rows[] = [
 \$index + 1, \$rekap['nama_siswa'], \$rekap['nama_kelas'],
 \$rekap['hadir'], \$rekap['sakit'], \$rekap['izin'], \$rekap['alpa'],
 \$rekap['persentase'].'%',
];
 }

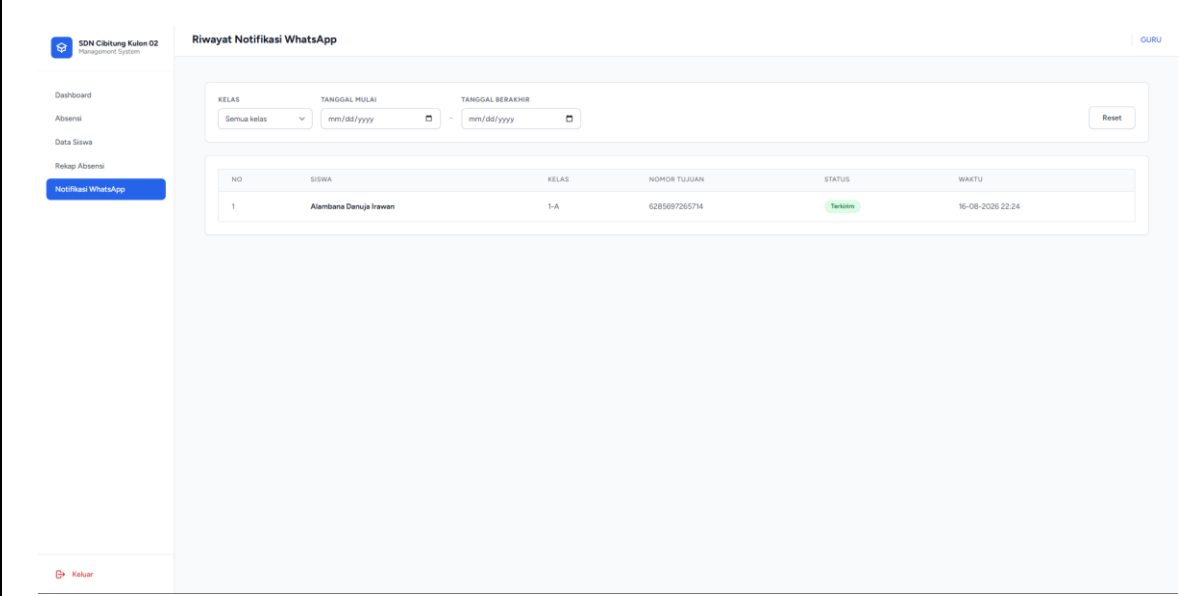
 return \$rows;
 }
}

Deskripsi

Halaman Hasil Ekspor Laporan Excel

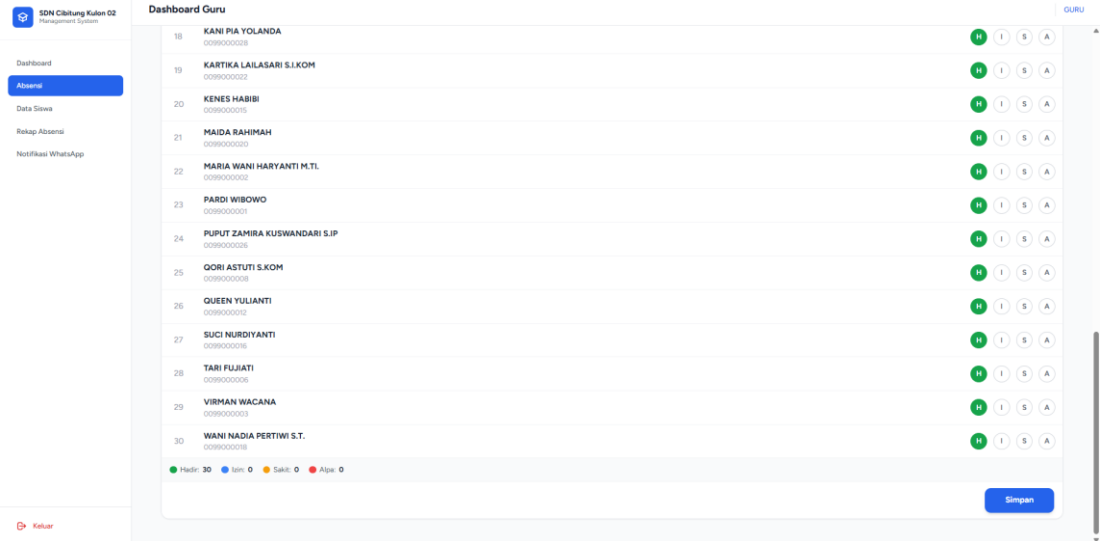
Halaman ini menampilkan tampilan rekap absensi sebagai dasar sebelum data diekspor ke dalam file Excel melalui tombol "Unduh Excel". File hasil ekspor memuat data kehadiran siswa sesuai kelas dan rentang periode yang telah difilter oleh guru, sehingga dapat digunakan sebagai laporan cetak maupun arsip kehadiran.

Tabel 5.33 Antarmuka dan Kode Halaman Riwayat Pesan

Halaman Riwayat Pesan	
	Gambar 5.16 Tampilan halaman Riwayat Pesan
Kode	
<pre> public function index(Request \$request): View { \$user = \$request->user(); \$filters = \$request->validate(['kelas_id' => ['nullable', 'integer', 'exists:kelas,id'], 'tanggal_mulai' => ['nullable', 'date'], 'tanggal_berakhir' => ['nullable', 'date', 'after_or_equal:tanggal_mulai'],]); \$tanggalMulai = (\$filters['tanggal_mulai'] ?? null) ? \$this-> >parseDate(\$filters['tanggal_mulai'])->startOfDay() : null; \$tanggalBerakhir = (\$filters['tanggal_berakhir'] ?? null) ? \$this-> >parseDate(\$filters['tanggal_berakhir'])->endOfDay() : null; \$kelas = Kelas::query() ->accessibleBy(\$user) ->select(['id', 'nama_kelas']) ->orderBy('nama_kelas') ->get(); </pre>	

Halaman Riwayat Pesan
<pre> // Batasi notifikasi berdasarkan kelas yang dapat diakses pengguna. \$kelasIds = \$kelas->pluck('id'); //Ambil semua id dari data kelas \$kelasId = isset(\$filters['kelas_id']) ? (int) \$filters['kelas_id'] : null; //Kalau kelas_id ada di \$filters, masukkan nilainya ke \$kelasId. Kalau tidak ada, \$kelasId diisi null if (\$kelasId === null && \$kelas->count() === 1) { \$kelasId = \$kelas->first()->id; } //Jika user belum memilih kelas DAN user hanya memiliki satu kelas, maka sistem otomatis memilih kelas tersebut. abort_if(\$kelasId !== null && ! \$kelasIds->contains(\$kelasId), 404); //Jika user meminta kelas tertentu, tetapi kelas tersebut tidak termasuk kelas yang boleh dia akses, hentikan request dengan HTTP 404. //Mulai mengambil data notifikasi \$notifikasi = WhatsappNotification::query() ->select(['id', 'absensi_id', 'siswa_id', 'parent_phone', 'status', 'last_error', 'sent_at', 'updated_at', 'created_at']) ->with(['siswa:id,nama_siswa,kelas_id', 'absensi:id,kelas_id', 'absensi.kelas:id,nama_kelas',]) ->when(\$user->role === 'guru' \$kelasId !== null, function (Builder \$query) use (\$user, \$kelasIds, \$kelasId): void { //user adalah guru, atau user memilih kelas. \$query->whereHas('absensi', function (Builder \$query) use (\$user, \$kelasIds, \$kelasId): void { //Jalankan fungsi ini, berikan objek query ke \$query, dan izinkan fungsi menggunakan variabel \$user, \$kelasIds, dan \$kelasId dari luar. if (\$user->role === 'guru') { \$query->whereIn('kelas_id', \$kelasIds); } if (\$kelasId !== null) { \$query->where('kelas_id', \$kelasId); } }); }) ->when(\$tanggalMulai, function (Builder \$query) use (\$tanggalMulai): void { \$query->where('created_at', '>=', \$tanggalMulai); }) ->when(\$tanggalBerakhir, function (Builder \$query) use (\$tanggalBerakhir): void { \$query->where('created_at', '<=', \$tanggalBerakhir); }) ->latest() ->paginate(10) ->withQueryString(); return view('notifikasi.index', compact('kelas', 'notifikasi')); } </pre>
Deskripsi
Halaman ini menampilkan riwayat pengiriman pesan WhatsApp terkait ketidakhadiran siswa kepada orang tua, meliputi nama siswa, kelas, nomor tujuan, status pengiriman (misalnya "Ter kirim"), dan waktu pengiriman. Guru dapat memfilter riwayat pesan berdasarkan kelas dan rentang tanggal melalui form filter di bagian atas halaman.

Tabel 5.34 Antarmuka dan Kode Halaman Isi Absensi

Halaman Isi Absensi	
	
Kode	
<pre> public function store(Request \$request): RedirectResponse { // Ambil data user yang sedang login \$user = \$request->user(); // Validasi: Hanya guru yang boleh menyimpan absensi abort_unless(\$user->role === 'guru', 403, 'Akses absensi hanya tersedia untuk guru.');</pre> <pre> // Ambil kelas yang diampu guru \$userKelas = \$user->kelas; // Validasi: Guru harus sudah ditugaskan ke kelas if (!\$userKelas) { return redirect()->route('absensi.create') ->with('error', 'Anda belum ditugaskan ke kelas manapun.');</pre> <pre> } \$kelasId = \$userKelas->id; // Validasi input dari form // absensi.* artinya: untuk setiap siswa_id sebagai key \$data = \$request->validate(['tanggal' => ['required', 'date', 'before_or_equal:today'], 'absensi' => ['required', 'array'], 'absensi.*' => ['required', 'in:hadir,izin,sakit,alpa'],]); // Parse tanggal ke format Y-m-d \$tanggal = \$this->parseDate(\$data['tanggal'])->format('Y-m-d');</pre> <pre> \$userId = \$user->getKey();</pre>	

Halaman Isi Absensi

```

// Cari periode aktif untuk tanggal yang dipilih
// Jika tidak ada periode, return 404 error
$activePeriode = $this->activePeriodeOrFail($tanggal);

// Pastikan guru punya akses ke kelas ini (hanya bisa input kelas yang
diampu)
$this->ensureKelasAccessibleTo($user, $kelasId);

// Validasi: tanggal harus dalam range periode akademik
// (tidak boleh di luar tanggal_mulai dan tanggal_selesai periode)
if ($dateError = $this->attendanceDateError($activePeriode, $tanggal)) {
    return redirect()->route('absensi.create', ['kelas_id' => $kelasId,
'tanggal' => $tanggal])
        ->with('error', $dateError);
}

// Validasi: tanggal tidak boleh jatuh pada hari libur
// (baik libur nasional maupun libur mingguan seperti Sabtu/Minggu)
$holiday = $this->findHariLibur($activePeriode->id, $tanggal);
if ($holiday) {
    return redirect()->route('absensi.create', ['kelas_id' => $kelasId,
'tanggal' => $tanggal])
        ->with('error', $this->formatHariLiburMessage($holiday,
$tanggal));
}

// Ambil semua ID siswa aktif di kelas ini
$siswaIds = Siswa::query()
    ->where('kelas_id', $kelasId)
    ->orderBy('id')
    ->pluck('id')
    ->all();

// Validasi: Pastikan payload absensi lengkap untuk semua siswa
// (tidak boleh ada siswa yang terlewat atau ID siswa yang tidak valid)
$this->ensureCompleteAttendancePayload($data['absensi'], $siswaIds);

// Cek apakah absensi untuk kelas & tanggal ini sudah pernah diinput
$sudahAbsen = Absensi::query()
    ->where('kelas_id', $kelasId)
    ->where('tanggal', $tanggal)
    ->exists();

// Jika sudah ada, redirect ke halaman edit (tidak boleh input ulang)
if ($sudahAbsen) {
    return redirect()->route('absensi.create', ['kelas_id' => $kelasId,
'tanggal' => $tanggal])
        ->with('error', 'Data absensi kelas ini pada tanggal tersebut
sudah terisi. Gunakan menu Edit Absensi untuk melakukan perubahan.');
```

}

```

$now = now();
$rows = []; // Data untuk batch insert
$notifSiswaIds = []; // ID siswa yang perlu dikirim notifikasi WA

// Siapkan data untuk insert batch (lebih efisien daripada insert satu per
satu)
foreach ($siswaIds as $siswaId) {
    $status = $data['absensi'][$siswaId];

    // Siapkan row untuk insert
    $rows[] = [
        'siswa_id' => $siswaId,
```

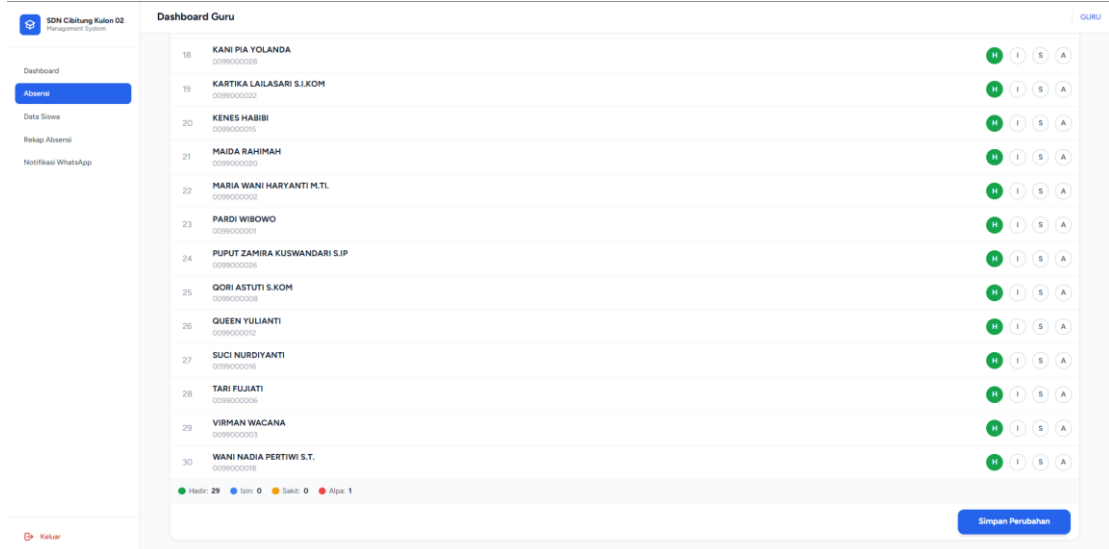
Halaman Isi Absensi
<pre> 'kelas_id' => \$kelasId, 'user_id' => \$userId, // ID guru yang input 'periode_id' => \$activePeriode->id, 'tanggal' => \$tanggal, 'status' => \$status, 'created_at' => \$now, 'updated_at' => \$now,]; // Kumpulkan ID siswa yang statusnya sakit/izin/alpa // (hanya siswa dengan status ini yang perlu dikirim notifikasi WA) if (in_array(\$status, ['alpa', 'sakit', 'izin'])) { \$notifSiswaIds[] = \$siswaId; } } // Validasi: Pastikan ada data untuk disimpan if (\$rows === []) { return redirect()->route('absensi.create', ['kelas_id' => \$kelasId, 'tanggal' => \$tanggal]) ->with('error', 'Tidak ada data siswa yang valid untuk disimpan.');</pre> <pre> } // Simpan data absensi dalam transaction untuk menjaga konsistensi data // Jika ada error, semua data akan di-rollback try { DB::transaction(function () use (\$rows): void { Absensi::insert(\$rows); // Batch insert untuk performa optimal }); } catch (QueryException \$exception) { // Handle race condition: duplicate entry (jika ada proses lain yang insert bersamaan) // Error code 23000 = integrity constraint violation if ((\$exception->errorInfo[0] ?? null) !== '23000') { throw \$exception; // Jika error lain, lempar exception } // Jika duplicate, beri tahu user untuk reload halaman return redirect()->route('absensi.create', ['kelas_id' => \$kelasId, 'tanggal' => \$tanggal]) ->with('error', 'Absensi pada tanggal tersebut sudah disimpan oleh proses lain. Muat ulang halaman.');</pre> <pre> } // Antrekan notifikasi WhatsApp untuk siswa yang sakit/izin/alpa // Notifikasi akan dikirim via queue/background job \$this->queueAbsensiNotificationsFor(\$notifSiswaIds, \$tanggal); // Redirect kembali ke halaman create dengan pesan sukses return redirect()->route('absensi.create', ['kelas_id' => \$kelasId, 'tanggal' => \$tanggal]) ->with('success', 'Data absensi baru berhasil disimpan.');</pre> <pre> } }</pre>
Deskripsi
Halaman ini digunakan guru untuk mengisi status kehadiran harian setiap siswa dalam satu kelas, dengan empat pilihan status: Hadir (H), Izin (I), Sakit (S), atau Alpa (A) yang dipilih melalui tombol radio berwarna di sisi kanan setiap baris siswa.

Halaman Isi Absensi

Sistem menampilkan ringkasan jumlah siswa pada masing-masing status di bagian bawah, dan data akan tersimpan setelah guru menekan tombol "Simpan".

Tabel 5.35 Antarmuka dan Kode Halaman Koreksi Absensi

Halaman Koreksi Absensi



Gambar 5.18 Tampilan halaman Koreksi Absensi

Kode

```
public function update(Request $request): RedirectResponse
{
    // Ambil data user yang sedang login
    $user = $request->user();

    // Validasi: Hanya guru yang boleh update absensi
    abort_unless($user->role === 'guru', 403, 'Akses absensi hanya tersedia
    untuk guru.');
```

```
    // Ambil kelas yang diampu guru
    $userKelas = $user->kelas;
    if (! $userKelas) {
        return redirect()->route('absensi.edit')
            ->with('error', 'Anda belum ditugaskan ke kelas manapun.');
```

```
    }

    $kelasId = $userKelas->id;

    // Validasi input dari form
    $data = $request->validate([
        'tanggal' => ['required', 'date', 'before_or_equal:today'],
        'absensi' => ['required', 'array'],
```

Halaman Koreksi Absensi

```

        'absensi.*' => ['required', 'in:hadir,izin,sakit,alpa'],
    ]);

    // Parse tanggal
    $tanggal = $this->parseDate($data['tanggal'])->format('Y-m-d');

    $userId = $user->getKey();

    // Cari periode aktif untuk tanggal yang dipilih
    $activePeriode = $this->activePeriodeOrFail($tanggal);

    // Pastikan guru punya akses ke kelas ini
    $this->ensureKelasAccessibleTo($user, $kelasId);

    // Validasi: tanggal harus dalam range periode
    if ($dateError = $this->attendanceDateError($activePeriode, $tanggal)) {
        return redirect()->route('absensi.edit', ['kelas_id' => $kelasId,
'tanggal' => $tanggal])
            ->with('error', $dateError);
    }

    // Validasi: tanggal tidak boleh jatuh pada hari libur
    $holiday = $this->findHariLibur($activePeriode->id, $tanggal);
    if ($holiday) {
        return redirect()->route('absensi.edit', ['kelas_id' => $kelasId,
'tanggal' => $tanggal])
            ->with('error', $this->formatHariLiburMessage($holiday,
$tanggal));
    }

    // Ambil semua ID siswa aktif di kelas
    $siswaIds = Siswa::query()
        ->where('kelas_id', $kelasId)
        ->orderBy('id')
        ->pluck('id')
        ->all();

    // Validasi: Pastikan payload lengkap untuk semua siswa
    $this->ensureCompleteAttendancePayload($data['absensi'], $siswaIds);

    // Ambil data absensi existing untuk perbandingan (detect changes)
    // Gunakan keyBy untuk mapping siswa_id => Absensi model
    $existingAbsensis = Absensi::query()
        ->where('kelas_id', $kelasId)
        ->where('tanggal', $tanggal)
        ->whereIn('siswa_id', $siswaIds)
        ->get()
        ->keyBy('siswa_id');

    $now = now();
    $upsertRows = []; // Data untuk upsert (insert or update)
    $changedSiswaIds = []; // ID siswa yang statusnya berubah (untuk
notifikasi WA)
    $hasChanges = false; // Flag apakah ada perubahan data

    // Bandingkan data lama vs baru, siapkan untuk upsert
    foreach ($siswaIds as $siswaId) {
        $status = $data['absensi'][$siswaId]; // Status baru dari form
        $existing = $existingAbsensis->get($siswaId); // Data lama dari
database
        $oldStatus = $existing->status; // Status lama (null jika belum ada
data)
    }

```

Halaman Koreksi Absensi
<pre> // Deteksi perubahan: status, user_id, atau kelas_id berbeda \$hasChanges = \$hasChanges ! \$existing // Belum ada data sebelumnya \$oldStatus !== \$status // Status berubah (int) \$existing->user_id !== (int) \$userId // User yang input berbeda (int) \$existing->kelas_id !== \$kelasId; // Kelas berbeda // Siapkan row untuk upsert \$upsertRows[] = ['siswa_id' => \$siswaId, 'kelas_id' => \$kelasId, 'user_id' => \$userId, 'periode_id' => \$activePeriode->id, 'tanggal' => \$tanggal, 'status' => \$status, 'created_at' => \$existing->created_at ?? \$now, // Pertahankan created_at lama jika ada 'updated_at' => \$now,]; // Kirim notifikasi WA untuk SEMUA siswa yang STATUS-nya berubah // Tidak peduli dari status apa ke status apa (hadir/sakit/izin/alpa) if (\$existing && \$oldStatus !== \$status) { \$changedSiswaIds[] = \$siswaId; } // Jika tidak ada perubahan, tidak perlu update database if (! \$hasChanges) { return redirect()->route('absensi.create', ['kelas_id' => \$kelasId, 'tanggal' => \$tanggal]) ->with('success', 'Tidak ada perubahan data absensi.');</pre> <pre> // Simpan perubahan menggunakan upsert (insert or update) // Upsert lebih efisien daripada cek exist -> update/insert DB::transaction(function () use (\$upsertRows): void { Absensi::upsert(\$upsertRows, ['siswa_id', 'tanggal'], // Unique key untuk menentukan update atau insert ['kelas_id', 'user_id', 'periode_id', 'status', 'updated_at'] // Kolom yang diupdate); }); // Antrekan notifikasi WhatsApp untuk semua siswa yang statusnya berubah // Notifikasi dikirim via queue/background job \$this->queueEditAbsensiNotificationsFor(\$changedSiswaIds, \$tanggal); // Redirect kembali dengan pesan sukses return redirect()->route('absensi.create', ['kelas_id' => \$kelasId, 'tanggal' => \$tanggal]) ->with('success', 'Data riwayat absensi berhasil diperbarui.');</pre>
Deskripsi

Halaman Koreksi Absensi

Halaman ini memiliki tampilan serupa dengan halaman Isi Absensi, namun digunakan untuk mengoreksi data absensi yang sudah pernah diinput sebelumnya. Guru dapat mengubah status kehadiran siswa (Hadir, Izin, Sakit, Alpa) yang sudah tersimpan, dan perubahan akan diperbarui ke database setelah menekan tombol "Simpan Perubahan".

Tabel 5.36 Antarmuka dan Kode Pesan WhatsApp Kepada Orang tua

Pesan WhatsApp Kepada Orang tua



Gambar 5.19 Tampilan pesan WhatsApp

Kode

```
private function buildAbsensiWhatsappMessage(Absensi $absensi, ?string
$parentName): string
{
    $siswa = $absensi->siswa;

    // Format tanggal: 02/09/2026
    $tanggal = Carbon::parse($absensi->tanggal)->format('d/m/Y');
```

Pesan WhatsApp Kepada Orang tua

```

        // Nama hari: Rabu
        $hari = $this->namaHariIndonesia(Carbon::parse($absensi->tanggal)-
>dayOfWeek);

        // Sapaan personal jika ada nama orang tua, generic jika tidak ada
        $sapaan = $parentName ? "Yth. Bapak/Ibu {$parentName}" : 'Yth. Bapak/Ibu
Orang Tua/Wali';

        // Nama kelas: 1-A, 2-B, dst
        $kelas = $absensi->kelas?->nama_kelas ?? '-';

        // Status dalam huruf kapital: SAKIT, IZIN, ALPA
        $status = strtoupper($absensi->status);

        $namaSekolah = 'SDN CIBITUNGKULON 02';

        // Keterangan status (penjelasan user-friendly)
        $keterangan = match (strtolower($absensi->status)) {
            'sakit' => 'Siswa tidak hadir karena sakit.',
            'izin' => 'Siswa tidak hadir karena izin.',
            'alpa' => 'Siswa tidak hadir tanpa keterangan.',
            default => 'Informasi kehadiran siswa.',
        };

        // Build pesan line by line
        $pesan = "NOTIFIKASI KEHADIRAN SISWA\n";
        $pesan .= "{$namaSekolah}\n\n";
        $pesan .= "{$sapaan},\n\n";
        $pesan .= "Berikut informasi kehadiran putra/putri Anda:\n\n";
        $pesan .= "Nama      : {$siswa->nama_siswa}\n";
        $pesan .= "Kelas   : {$kelas}\n";
        $pesan .= "Hari     : {$hari}, {$tanggal}\n";
        $pesan .= "Status   : {$status}\n\n";
        $pesan .= "{$keterangan}\n\n";
        $pesan .= "Untuk informasi lebih lanjut, silakan hubungi wali
kelas.\n\n";
        $pesan .= "Terima kasih.\n\n";
        $pesan .= $this->formatWaliKelasFooter($absensi); // Footer dengan
contact wali kelas

        return $pesan;
    }

    private function buildHadirWhatsappMessage(Absensi $absensi, ?string
$parentName): string
    {
        $siswa = $absensi->siswa;

        // Format tanggal dan hari
        $tanggal = Carbon::parse($absensi->tanggal)->format('d/m/Y');
        $hari = $this->namaHariIndonesia(Carbon::parse($absensi->tanggal)-
>dayOfWeek);

        // Sapaan
        $sapaan = $parentName ? "Yth. Bapak/Ibu {$parentName}" : 'Yth. Bapak/Ibu
Orang Tua/Wali';

        // Nama kelas
        $kelas = $absensi->kelas?->nama_kelas ?? '-';

        $namaSekolah = 'SDN CIBITUNGKULON 02';

        // Build pesan khusus untuk perubahan ke hadir
        $pesan = "PEMBARUAN KEHADIRAN SISWA\n";

```

Pesan WhatsApp Kepada Orang tua
<pre> \$pesan .= "{\$namaSekolah}\n\n"; \$pesan .= "{\$sapaan},\n\n"; \$pesan .= "Berikut pembaruan kehadiran putra/putri Anda:\n\n"; \$pesan .= "Nama : {\$siswa->nama_siswa}\n"; \$pesan .= "Kelas : {\$kelas}\n"; \$pesan .= "Hari : {\$hari}, {\$tanggal}\n"; \$pesan .= "Status : HADIR (Diperbarui)\n\n"; \$pesan .= "Data kehadiran telah dikoreksi oleh wali kelas.\n"; \$pesan .= "Siswa tercatat hadir pada tanggal tersebut.\n\n"; \$pesan .= "Untuk informasi lebih lanjut, silakan hubungi wali kelas.\n\n"; \$pesan .= "Terima kasih.\n\n"; \$pesan .= \$this->formatWaliKelasFooter(\$absensi); return \$pesan; </pre>
Deskripsi
Fitur ini secara otomatis mengirimkan notifikasi WhatsApp kepada orang tua setiap kali status kehadiran diinput atau dikoreksi oleh guru. Untuk siswa dengan status Sakit, Izin, atau Alpa, sistem mengirimkan pesan "NOTIFIKASI KEHADIRAN SISWA" yang memuat nama siswa, kelas, hari/tanggal, status kehadiran, serta kontak wali kelas. Jika status siswa dikoreksi kembali menjadi Hadir, sistem mengirimkan pesan "PEMBARUAN KEHADIRAN SISWA" sebagai pemberitahuan bahwa data telah diperbarui. Nomor tujuan diprioritaskan pada nomor WhatsApp ibu, dengan nomor ayah sebagai nomor cadangan (fallback) apabila pengiriman ke nomor utama gagal. Pesan dikirim melalui layanan WhatsApp Gateway Fonnte secara asynchronous menggunakan queue job agar tidak menghambat proses penyimpanan absensi.

5.4 Proses *Deployment* Aplikasi

Deployment sistem merupakan tahap akhir dalam pengembangan aplikasi, yaitu proses penerapan aplikasi yang telah selesai dibangun ke lingkungan produksi agar dapat diakses oleh pengguna secara luas. Pada penelitian ini, proses deployment dilakukan menggunakan Railway sebagai platform Platform as a Service (PaaS), dengan aplikasi dikemas ke dalam sebuah container menggunakan Dockerfile.

Railway dipilih karena mendukung deployment berbasis container secara otomatis melalui integrasi dengan repositori GitHub, sehingga setiap push ke branch yang ditentukan langsung dibangun ulang dan diterapkan tanpa konfigurasi server manual. Railway juga menyediakan basis data MySQL terkelola.

5.4.1 Konfigurasi Dockerfile

Aplikasi dikemas menggunakan Dockerfile agar dapat dibangun dan dijalankan secara konsisten, baik pada lingkungan Railway maupun pada lingkungan pengembangan lokal. Adapun tahapan build yang didefinisikan dalam Dockerfile meliputi:

1. Base image PHP beserta ekstensi yang dibutuhkan oleh framework Laravel, seperti OpenSSL, PDO, dan Mbstring, digunakan sebagai dasar image aplikasi.
2. Composer digunakan untuk menginstal seluruh dependency PHP (folder vendor) yang dibutuhkan aplikasi.
3. Node.js dan npm digunakan untuk menjalankan proses build aset front-end (Tailwind CSS dan Alpine.js) sebelum aplikasi dijalankan.
4. Seluruh source code aplikasi disalin ke dalam image, kemudian dilakukan pembersihan cache konfigurasi Laravel (config:clear, route:clear, view:clear) agar aplikasi tidak menggunakan cache konfigurasi/route/view dari proses build sebelumnya yang bisa saja sudah kedaluwarsa, mengingat variabel lingkungan (.env) baru disediakan oleh Railway saat container dijalankan, bukan saat image dibangun.
5. Port aplikasi dikonfigurasi agar mengikuti variabel PORT yang disediakan secara dinamis oleh Railway saat container dijalankan.
6. Container dikonfigurasi untuk menjalankan dua proses sekaligus saat startup: php artisan queue:work untuk memproses antrian pengiriman pesan WhatsApp secara asinkron (lihat Job pada Tabel 5.8), dan php artisan serve untuk melayani permintaan HTTP pada port yang ditentukan variabel PORT. Kedua proses dijalankan dalam satu container yang sama tanpa process manager terpisah (mis. Supervisor), sehingga jika proses queue worker berhenti karena error, proses tersebut tidak akan otomatis dijalankan ulang selama container masih berjalan.

5.4.2 Konfigurasi Layanan Railway

Konfigurasi layanan pada Railway dilakukan untuk memastikan aplikasi dapat berjalan dengan optimal dan terhubung dengan basis data. Adapun konfigurasi yang dilakukan meliputi:

1. Layanan Web Service dibuat berdasarkan Dockerfile yang terdapat pada repositori aplikasi, sehingga proses build dan deploy dilakukan secara otomatis oleh Railway.
2. Layanan basis data MySQL ditambahkan melalui plugin bawaan Railway, sehingga kredensial dan variabel koneksi basis data (host, port, nama database, user, dan password) disediakan secara otomatis oleh Railway.
3. Variabel lingkungan (.env) aplikasi, seperti APP_KEY, kredensial basis data hasil referensi dari plugin MySQL, serta token API Fonnte untuk WhatsApp *Gateway*, dikonfigurasi melalui tab Variables pada dashboard Railway.
4. Railway dikonfigurasi agar melakukan auto-deploy, yaitu membangun ulang dan menerapkan aplikasi secara otomatis setiap kali terdapat perubahan kode yang diunggah ke branch yang telah dihubungkan pada repositori GitHub.
5. Aplikasi dapat diakses melalui domain bawaan yang disediakan secara otomatis oleh Railway dengan format *.up.railway.app, tanpa memerlukan konfigurasi domain atau DNS tambahan.

5.4.3 Deployment Aplikasi

Proses deployment aplikasi ke Railway dilakukan melalui beberapa tahapan, antara lain:

1. Source code aplikasi diunggah (push) ke repositori GitHub pada branch yang telah dihubungkan dengan proyek Railway.
2. Railway secara otomatis mendeteksi Dockerfile pada repositori dan menjalankan proses build image container aplikasi.
3. Setelah image berhasil dibangun, Railway menerapkan (menjalankan) container tersebut sebagai Web Service dan menghubungkannya dengan layanan MySQL plugin melalui variabel lingkungan yang telah dikonfigurasi.
4. Migrasi database dijalankan secara manual melalui Railway CLI/shell setelah layanan berhasil di-deploy, untuk membuat struktur tabel sesuai dengan desain sistem yang telah dirancang sebelumnya. Proses ini tidak termasuk dalam langkah otomatis yang didefinisikan pada Dockerfile.
5. Setelah proses deployment selesai, aplikasi dapat langsung diakses oleh pengguna melalui domain bawaan Railway (*.up.railway.app), dan setiap perubahan kode berikutnya akan otomatis diterapkan mengikuti mekanisme auto-deploy yang telah dikonfigurasi.

5.5 Pengujian Perangkat Lunak

Pengujian perangkat lunak adalah proses eksekusi program untuk menemukan kesalahan serta memverifikasi bahwa sistem memenuhi kebutuhan yang ditetapkan. Pada penelitian ini, pengujian dilakukan menggunakan Black Box Testing untuk memverifikasi kesesuaian fungsional terhadap 8 use case pada BAB 4 beserta fitur Login & Lupa Password, dan User Acceptance Testing untuk memvalidasi kelayakan sistem dari perspektif Operator, Guru, dan Kepala Sekolah.

5.5.1 Black Box Testing

Black Box Testing merupakan metode pengujian yang berfokus pada fungsionalitas sistem dari sudut pandang pengguna tanpa memperhatikan struktur internal kode. Pengujian dilakukan terhadap setiap use case berdasarkan skenario utama (valid) dan skenario alternatif (tidak valid) yang telah dirumuskan pada BAB 4.

Kasus uji (test case) disusun berdasarkan kebutuhan fungsional dan skenario use case UC-01 sampai UC-08 pada BAB 4, dengan kolom Test Scenario (pengelompokan fungsi yang diuji), Test Case (kondisi pengujian spesifik), Test Step (langkah pengujian), Expected (hasil yang diharapkan), Actual (hasil aktual), dan Pass/Fail (status keberhasilan, Pass jika aktual sesuai harapan, Fail jika tidak).

Perancangan test case mengacu pada teknik Equivalence Partitioning , yang membagi data masukan ke dalam kelas valid dan kelas tidak valid. Kelas valid menguji masukan yang memenuhi ketentuan sistem, seperti kredensial benar atau formulir lengkap, sedangkan kelas tidak valid menguji masukan yang tidak memenuhi ketentuan, seperti kredensial salah, data duplikat, atau field wajib yang kosong.

Selain Equivalence Partitioning, teknik Boundary Value Analysis diterapkan pada pengujian batas domain data, yaitu pada UC-04 untuk memvalidasi batas tanggal periode tahun ajaran (tanggal mulai-selesai dan batas antar semester), dan pada UC-08 untuk memvalidasi batas tanggal absensi terhadap hari libur dan rentang periode aktif.

Tabel 5.37 Black Box Testing Login Sistem

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
TS-00 - <i>Login</i> Sistem Berdasarkan Peran	TC-00.1 <i>Login</i> dengan akun Operator yang valid	1. Buka halaman login.2. Masukkan username/NIP dan password Operator.3. Klik <i>Login</i> .	Sistem memvalidasi kredensial, login berhasil, dan mengarahkan ke <i>Dashboard</i> Operator sesuai role.	<i>Login</i> berhasil dan diarahkan ke <i>Dashboard</i> Operator.	Pass
	TC-00.2 <i>Login</i> dengan akun Guru yang valid	1. Buka halaman login.2. Masukkan username/NIP dan password Guru.3. Klik <i>Login</i> .	Sistem memvalidasi kredensial, login berhasil, dan mengarahkan ke <i>Dashboard</i> Guru sesuai role.	<i>Login</i> berhasil dan diarahkan ke <i>Dashboard</i> Guru.	Pass
	TC-00.3 <i>Login</i> dengan akun Kepala Sekolah yang valid	1. Buka halaman login.2. Masukkan username/NIP dan password Kepala Sekolah.3. Klik <i>Login</i> .	Sistem memvalidasi kredensial, login berhasil, dan mengarahkan ke <i>Dashboard</i> Kepala Sekolah sesuai role.	<i>Login</i> berhasil dan diarahkan ke <i>Dashboard</i> Kepala Sekolah.	Pass
	TC-00.4 <i>Login</i> dengan username/NIP atau password yang salah	1. Buka halaman login.2. Masukkan username/NIP dan/atau password yang salah.3. Klik <i>Login</i> .	Sistem menolak proses login dan menampilkan pesan kesalahan.	Sistem menolak login dan menampilkan pesan kesalahan.	Pass
	TC-00.5 Lupa password dengan nomor WhatsApp terdaftar	1. Buka login.2. Klik Lupa <i>Password</i> .3. Masukkan nomor WhatsApp terdaftar.4. Klik Kirim.	Sistem memvalidasi nomor, mengirim link perubahan password melalui WhatsApp, dan menampilkan pesan berhasil.	Link perubahan password berhasil dikirim.	Pass
	TC-00.6 Lupa password dengan nomor WhatsApp tidak terdaftar	1. Klik Lupa <i>Password</i> .2. Masukkan nomor yang tidak terdaftar.3. Klik Kirim.	Sistem menolak permintaan dan menampilkan pesan nomor WhatsApp tidak terdaftar.	Sistem menolak permintaan dan menampilkan pesan tersebut.	Pass
	TC-00.7 Lupa password tanpa nomor WhatsApp	1. Klik Lupa <i>Password</i> .2. Biarkan nomor WhatsApp kosong.3. Klik Kirim.	Sistem menampilkan validasi bahwa	Sistem menampilkan validasi dan tidak mengirim link.	Pass

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
			nomor WhatsApp wajib diisi.		
	TC-00.8 Lupa password dengan format nomor tidak valid	1. Klik Lupa Password.2. Masukkan nomor dengan format tidak valid.3. Klik Kirim.	Sistem menampilkan pesan bahwa format nomor WhatsApp tidak valid.	Sistem menampilkan pesan format nomor tidak valid.	Pass
	TC-00.9 Membuka link perubahan password	1. Melakukan proses lupa password.2. Buka WhatsApp.3. Buka link perubahan password.	Sistem mengarahkan pengguna ke halaman ubah password.	Sistem menampilkan halaman ubah password.	Pass
	TC-00.10 Mengubah password melalui link valid	1. Buka link perubahan password.2. Masukkan password baru.3. Masukkan konfirmasi.4. Klik Simpan/Ubah Password.	Sistem menyimpan password baru dan menampilkan pesan berhasil.	Password berhasil diubah.	Pass
	TC-00.11 Konfirmasi password berbeda	1. Buka link perubahan password.2. Masukkan password baru.3. Masukkan konfirmasi berbeda.4. Klik Simpan.	Sistem menolak perubahan dan menampilkan pesan konfirmasi tidak sesuai.	Sistem menolak perubahan password.	Pass
	TC-00.12 Menggunakan link perubahan password yang tidak berlaku	1. Buka link yang kedaluwarsa/tidak valid.	Sistem menolak akses dan menampilkan pesan link sudah tidak berlaku.	Sistem menolak akses perubahan password.	Pass
	TC-00.13 Login menggunakan password baru	1. Lakukan reset password.2. Kembali ke login.3. Masukkan password baru.4. Klik Login.	Sistem menerima password baru dan mengarahkan pengguna ke Dashboard sesuai role.	Login berhasil menggunakan password baru.	Pass
	TC-00.14 Login menggunakan password lama setelah reset	1. Lakukan reset password.2. Masukkan password lama.3. Klik Login.	Sistem menolak login karena password lama tidak berlaku.	Sistem menolak login.	Pass

Seluruh 14 kasus uji Login Sistem Berdasarkan Peran (Tabel 5.37) sesuai hasil yang diharapkan dan dinyatakan valid. Pengujian ini dilakukan oleh Aktor Operator, Guru, dan Kepala Sekolah, dengan Requirement terkait FR-01 s.d. FR-05.

Tabel 5.38 Black Box Testing Kelola Data Guru

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
TS-01 Kelola Data Guru	TC-01.1 Menampilkan daftar data guru	1. Login sebagai Operator.2. Buka Kelola Data Guru.	Sistem menampilkan daftar guru beserta tombol Tambah, Ubah, dan Hapus.	Daftar guru berhasil ditampilkan.	Pass
	TC-01.2 Menambah data guru valid	1. Klik Tambah Guru.2. Isi seluruh data.3. Klik Simpan.	Data guru tersimpan dan sistem menampilkan pesan sukses.	Data guru berhasil disimpan.	Pass

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
	TC-01.3 Menambah guru dengan data yang sudah terdaftar	1. Klik Tambah Guru.2. Masukkan Nama/Username/NIP/No. Telepon yang sudah digunakan.3. Klik Simpan.	Sistem menolak penyimpanan dan menampilkan pesan bahwa data sudah digunakan.	Sistem menolak data duplikat.	Pass
	TC-01.4 Field dengan tanda (*) wajib diisi	1. Klik Tambah Guru.2. Kosongkan field wajib.3. Klik Simpan.	Sistem menampilkan validasi dan tidak menyimpan data.	Validasi ditampilkan dan data tidak disimpan.	Pass
	TC-01.5 Password dan konfirmasi password berbeda	1. Klik Tambah/Ubah Guru.2. Isi password dan konfirmasi password dengan nilai berbeda.3. Klik Simpan.	Sistem menampilkan validasi bahwa password dan konfirmasi password harus memiliki value yang sama.	Validasi ditampilkan dan data tidak disimpan.	Pass
	TC-01.6 Mengubah data guru	1. Klik Ubah.2. Ubah data.3. Klik Simpan.	Data guru diperbarui dan pesan sukses ditampilkan.	Data berhasil diperbarui.	Pass
	TC-01.7 Menghapus guru tanpa relasi kelas aktif	1. Klik Hapus.2. Konfirmasi penghapusan.	Data guru dihapus dari database.	Data guru berhasil dihapus.	Pass
	TC-01.8 Menghapus guru yang menjadi wali kelas (tanpa riwayat absensi)	1. Klik Hapus pada guru yang menjadi wali kelas suatu kelas.2. Konfirmasi pada pop up konfirmasi (modal yang sama seperti penghapusan data lain).	Sistem menampilkan pop up konfirmasi umum (bukan konfirmasi khusus). Setelah dikonfirmasi, data guru dihapus dan kelas yang sebelumnya diampu otomatis dilepas (wali kelas menjadi kosong), bukan ikut terhapus.	Data guru terhapus, kelas yang diampu menjadi tanpa wali kelas (tidak ikut terhapus).	Pass
	TC-01.9 Menghapus guru yang memiliki riwayat absensi	1. Klik Hapus pada guru yang memiliki riwayat absensi (pernah mengisi/mengoreksi absensi).2. Konfirmasi pada pop up konfirmasi.	Sistem menolak penghapusan dan menampilkan pesan 'Pengguna tidak dapat dihapus karena memiliki riwayat absensi.' Data guru tidak terhapus.	Penghapusan ditolak, pesan 'Pengguna tidak dapat dihapus karena memiliki riwayat absensi.' ditampilkan.	Pass
	TC-01.10 Mencatat aktivitas pengelolaan guru ke log	1. Tambah data guru.2. Ubah data.3. Hapus data.4. Buka Log Aktivitas.	Sistem mencatat aktivitas tambah, ubah, hapus beserta user dan waktu.	Aktivitas berhasil tercatat pada log.	Pass

Seluruh 10 kasus uji UC-01: Kelola Data Guru (Tabel 5.38) sesuai hasil yang diharapkan dan dinyatakan valid.

Pengujian ini dilakukan oleh Aktor Operator, dengan Requirement terkait FR-06 s.d. FR-10.

Tabel 5.39 Black Box Testing Kelola Data Siswa

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
TS-02 Kelola Data Siswa	TC-02.1Menampilkan daftar siswa	1. Login sebagai Operator.2. Buka Kelola Data Siswa.	Sistem menampilkan daftar siswa beserta tombol Tambah, Ubah, dan Hapus.	Daftar siswa berhasil ditampilkan.	Pass
	TC-02.2Menambah siswa dengan data valid	1. Klik Tambah Siswa.2. Isi data siswa dan wali.3. Klik Simpan.	Sistem menyimpan data siswa dan menampilkan pesan sukses.	Data siswa berhasil disimpan.	Pass
	TC-02.3Menambah siswa dengan NISN/NIS>Nama duplikat	1. Klik Tambah Siswa.2. Masukkan NISN/NIS>Nama yang sudah terdaftar.3. Klik Simpan.	Sistem menolak penyimpanan dan menampilkan pesan data sudah digunakan.	Sistem menolak data duplikat.	Pass
	TC-02.4Field dengan tanda (*) wajib diisi	1. Klik Tambah Siswa.2. Kosongkan field wajib.3. Klik Simpan.	Sistem menampilkan validasi dan tidak menyimpan data.	Validasi ditampilkan.	Pass
	TC-02.5Tidak mengisi satu pun nomor telepon orang tua/wali	1. Klik Tambah Siswa.2. Kosongkan seluruh nomor telepon ayah, ibu, dan wali.3. Klik Simpan.	Sistem menampilkan pesan minimal harus mengisi 1 nomor telepon orang tua dan tidak menyimpan data.	Validasi ditampilkan dan data tidak disimpan.	Pass
	TC-02.6Mengubah data siswa	1. Klik Ubah.2. Ubah data.3. Klik Simpan Perubahan.	Data siswa diperbarui dan pesan sukses ditampilkan.	Data siswa berhasil diperbarui.	Pass
	TC-02.7Guru mencoba mengakses kelas yang bukan miliknya	1. Login sebagai Guru.2. Buka Kelola Data Siswa.	Sistem membatasi data siswa yang muncul hanya kelas yang diampu Guru.	Hanya siswa di kelas milik Guru yang ditampilkan.	Pass
	TC-02.8Mengubah data kelas siswa yang memiliki data absensi	1. Klik Ubah pada siswa yang sudah memiliki data absensi.2. Ubah field kelas.3. Klik Simpan.	Sistem disable field kelas	Sistem disable field kelas	Pass
	TC-02.9Menghapus siswa dengan riwayat absensi	1. Klik Hapus.2. Konfirmasi penghapusan.	Sistem menampilkan konfirmasi dan menghapus siswa beserta data absensi siswa yang dihapus lalu menampilkan pesan	Data siswa dan data terkait berhasil dihapus.	Pass
	TC-02.10Memindahkan kelas dari kelas yang tidak memiliki siswa	1. Buka Pindah kelas.2. Pilih kelas asal yang tidak memiliki siswa.3. Klik Simpan.	Sistem menampilkan pesan tidak ada siswa yang berada di kelas tersebut.	Sistem menampilkan pesan sesuai kondisi.	Pass
	TC-02.11Memindahkan kelas massal untuk siswa yang sudah memiliki data absensi	1. Buka Pindah kelas.2. Pilih kelas asal dan tujuan yang siswanya sudah memiliki data absensi.3. Klik Simpan.	Sistem menolak pemindahan dan menampilkan pesan tidak dapat memindahkan siswa karena memiliki data absensi.	Sistem menolak pemindahan kelas.	Pass

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
	TC-02.12Mengubah kelas siswa secara massal (kondisi valid)	1. Buka Pindah kelas.2. Pilih kelas asal dan tujuan.3. Pilih siswa.4. Klik Simpan.	Sistem memperbarui kelas siswa, mencatat riwayat kenaikan kelas, dan menampilkan pesan berhasil.	Kelas siswa berhasil diperbarui secara massal.	Pass
	TC-02.13Menambahkan siswa baru ke kelas yang sudah memiliki data absensi	1. Klik Tambah Siswa.2. Pilih kelas yang sudah memiliki data absensi.3. Isi data dan klik Simpan.	Sistem tetap menyimpan data siswa namun data rekap bernilai 0 dan perlu diisi manual oleh Guru di menu Absensi.	Data siswa tersimpan dengan rekap awal 0.	Pass
	TC-02.14Mencatat aktivitas siswa ke log	1. Tambah data siswa.2. Ubah data.3. Hapus data.4. Buka Log Aktivitas.	Sistem mencatat aktivitas tambah, ubah, hapus beserta user dan waktu.	Aktivitas berhasil tercatat.	Pass

Seluruh 14 kasus uji UC-02: Kelola Data Siswa (Tabel 5.39) sesuai hasil yang diharapkan dan dinyatakan valid. Pengujian ini dilakukan oleh Aktor Operator, dengan Requirement terkait FR-11 s.d. FR-16.

Tabel 5.40 Black Box Testing Kelola Data Kelas

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
TS-03 Kelola Data Kelas	TC-03.1Menampilkan daftar kelas	1. Login sebagai Operator.2. Buka Kelola Kelas.	Sistem menampilkan daftar kelas beserta tombol Tambah, Ubah, dan Hapus.	Daftar kelas berhasil ditampilkan.	Pass
	TC-03.2Menambah kelas dengan data valid	1. Klik Tambah Kelas.2. Isi nama kelas dan wali kelas.3. Klik Simpan.	Sistem menyimpan data kelas dan menampilkan pesan sukses.	Data kelas berhasil disimpan.	Pass
	TC-03.3Menambah kelas dengan nama yang sudah terdaftar	1. Klik Tambah Kelas.2. Masukkan nama kelas yang sudah terdaftar.3. Klik Simpan.	Sistem menolak data dan menampilkan pesan nama kelas sudah terdaftar.	Sistem menolak data duplikat.	Pass
	TC-03.4Field dengan tanda (*) wajib diisi	1. Klik Tambah Kelas.2. Kosongkan field wajib.3. Klik Simpan.	Sistem menampilkan pesan validasi di bawah field yang kosong.	Validasi ditampilkan dan data tidak disimpan.	Pass
	TC-03.5Mengubah data kelas	1. Klik Ubah.2. Ubah data.3. Klik Simpan.	Sistem menyimpan perubahan dan menampilkan pesan sukses.	Data kelas berhasil diperbarui.	Pass
	TC-03.6Menghapus kelas dengan absensi aktif	1. Klik Hapus pada kelas yang memiliki absensi aktif.	Sistem menolak penghapusan dan menampilkan pesan	Sistem menolak penghapusan.	Pass
	TC-03.7Menghapus kelas yang memiliki data siswa	1. Klik Hapus pada kelas yang masih memiliki data siswa.	Sistem menolak penghapusan dan menampilkan pesan	Sistem menolak penghapusan.	Pass
	TC-03.8Guru yang sudah menjadi wali kelas tidak muncul di dropdown	1. Buka form Tambah/Ubah Kelas.2. Amati daftar pilihan (dropdown) wali kelas.	Guru yang saat ini sudah menjadi wali kelas pada kelas lain tidak ditampilkan	Guru tersebut tidak muncul di dropdown.	Pass

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
			dalam pilihan dropdown wali kelas.		
	TC-03.9Mencatat aktivitas kelas ke log	1. Tambah kelas.2. Ubah kelas.3. Hapus kelas.4. Buka Log Aktivitas.	Sistem mencatat aktivitas beserta user dan waktu.	Aktivitas berhasil tercatat.	Pass
	TC-03.10Menghapus kelas tanpa data siswa dan tanpa absensi aktif	1. Buka Kelola Kelas.2. Pilih kelas yang tidak memiliki data siswa maupun absensi aktif.3. Klik Hapus.4. Konfirmasi penghapusan pada pop up.	Sistem menghapus data kelas dari database, kelas tidak lagi muncul pada daftar kelas.	Kelas berhasil dihapus dan tidak lagi tampil di daftar kelas.	Pass

Seluruh 10 kasus uji UC-03: Kelola Data Kelas (Tabel 5.40) sesuai hasil yang diharapkan dan dinyatakan valid, termasuk TC-03.10 yang menguji jalur penghapusan kelas yang berhasil pada kelas tanpa data siswa maupun absensi aktif. Pengujian ini dilakukan oleh Aktor Operator, dengan Requirement terkait FR-17 s.d. FR-20.

Tabel 5.41 Black Box Testing Kelola Data Periode

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
TS-04 Kelola Data Periode	TC-04.1Menampilkan periode tahun ajaran	1. Login sebagai Operator.2. Buka Kelola Periode Tahun Ajaran.	Sistem menampilkan periode, semester, tahun ajaran, tanggal mulai/selesai, dan hari libur.	Data periode berhasil ditampilkan.	Pass
	TC-04.2Mengubah periode tahun ajaran	1. Klik Ubah.2. Ubah tahun ajaran, tanggal, dan hari libur.3. Klik Simpan.	Sistem memperbarui data periode dan menampilkan pesan berhasil.	Data periode berhasil diperbarui.	Pass
	TC-04.3Tanggal mulai lebih besar dari tanggal selesai	1. Klik Ubah.2. Isi tanggal mulai lebih besar dari tanggal selesai.3. Klik Simpan.	Sistem menampilkan pesan validasi	Validasi ditampilkan dan data tidak disimpan.	Pass
	TC-04.4Field dengan tanda (*) wajib diisi	1. Klik Ubah.2. Kosongkan field wajib.3. Klik Simpan.	Sistem menampilkan pesan validasi di bawah field yang kosong.	Validasi ditampilkan dan data tidak disimpan.	Pass
	TC-04.5Tanggal mulai semester 2 <= tanggal berakhir semester 1	1. Klik Ubah.2. Isi tanggal mulai semester 2 kurang dari atau sama dengan tanggal berakhir semester 1.3. Klik Simpan.	Sistem menampilkan pesan validasi	Validasi ditampilkan dan data tidak disimpan.	Pass
	TC-04.6Mengubah tanggal periode yang sudah memiliki data absensi	1. Ubah tanggal periode (mulai/selesai) sehingga sebagian data absensi yang	Sistem langsung memperbarui tanggal periode dan hari libur tanpa menampilkan konfirmasi, data absensi	Tanggal periode diperbarui tanpa konfirmasi, absensi di luar rentang baru	Pass

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
		sudah ada berada di luar rentang tanggal baru.2. Klik Simpan.	yang tanggalnya berada di luar rentang periode baru otomatis terhapus, data absensi yang masih dalam rentang baru tetap tersimpan.	terhapus otomatis, absensi dalam rentang baru tetap tersimpan.	
	TC-04.7Mereset periode aktif	1. Pilih periode aktif.2. Klik Reset Periode.3. Konfirmasi.	Sistem menghapus data periode, absensi, dan rekap terkait dalam satu transaksi serta menampilkan pesan berhasil.	Data periode terkait berhasil direset.	Pass
	TC-04.8Mereset periode dengan tanggal semester belum diisi	1. Kosongkan tanggal semester 1 dan/atau 2.2. Klik Reset Periode.3. Konfirmasi.	Sistem tetap mereset periode meskipun tanggal semester belum diisi.	Periode berhasil direset.	Pass
	TC-04.9Mencatat aktivitas ubah periode ke log	1. Login sebagai Operator.2. Ubah data periode (tanggal atau hari libur).3. Buka Log Aktivitas.	Sistem mencatat aktivitas perubahan periode beserta user dan waktu.	Aktivitas ubah periode tercatat di Log Aktivitas.	Pass

Seluruh 9 kasus uji UC-04: Kelola Data Periode (Tabel 5.41) sesuai harapan dan dinyatakan valid lalu kasus nomor 3 dan 5 menerapkan Boundary Value Analysis pada batas tanggal mulai-selesai periode dan batas antar semester. Pengujian ini dilakukan oleh Aktor Operator, dengan Requirement terkait FR-21 s.d. FR-27.

Tabel 5.42 Black Box Testing Lihat Data Rekap

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
TS-05 Lihat Data Rekap	TC-05.1Menampilkan rekap berdasarkan filter kelas dan periode	1. Operator login.2. Buka Rekap Absensi.3. Pilih kelas/periode.4. Klik Filter.	Sistem menampilkan tabel rekap sesuai filter dengan data kehadiran siswa.	Rekap sesuai filter berhasil ditampilkan.	Pass
	TC-05.2Guru hanya melihat rekap kelas yang diampu	1. Guru login.2. Buka Rekap Absensi.3. Amati pilihan kelas.	Sistem hanya menampilkan kelas yang diampu Guru.	Hanya kelas yang diampu ditampilkan.	Pass
	TC-05.3Filter tidak menghasilkan data	1. Pilih filter tanpa data absensi.2. Klik Filter.	Sistem menampilkan tabel kosong dengan pesan tidak ada data absensi.	Pesan tidak ada data ditampilkan.	Pass

Seluruh 3 kasus uji UC-05: Lihat Data Rekap (Tabel 5.42) sesuai hasil yang diharapkan dan dinyatakan valid. Pengujian ini dilakukan oleh Aktor Operator, Guru, dan Kepala Sekolah, dengan Requirement terkait FR-28 s.d. FR-31.

Tabel 5.43 Black Box Testing Cetak Laporan Rekap

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
TS-06 Cetak Laporan Rekap	TC-06.1Mencetak laporan rekap dalam format Excel	1. Tampilkan rekap sesuai filter.2. Klik Cetak.3. Tunggu proses generate file.	Sistem menghasilkan laporan sesuai filter dalam format Excel (.xlsx) dan file berhasil diunduh.	File Excel berhasil dibuat dan diunduh.	Pass
	TC-06.2Mencetak laporan saat data rekap kosong	1. Tampilkan rekap dengan filter yang tidak memiliki data.2. Klik Cetak.	Sistem tetap mencetak file namun isinya kosong sesuai data yang ada.	File berhasil diunduh dengan isi kosong.	Pass

Seluruh 3 kasus uji UC-06: Cetak Laporan Rekap (Tabel 5.43) sesuai hasil yang diharapkan dan dinyatakan valid. Pengujian ini dilakukan oleh Aktor Operator, Guru, dan Kepala Sekolah, dengan Requirement terkait FR-32 s.d. FR-34.

Tabel 5.44 Black Box Testing Melihat Riwayat Pengiriman Pesan

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
TS-07 - Melihat Riwayat Pengiriman Pesan	TC-07.1 - Menampilkan riwayat pengiriman WhatsApp	1. Login sebagai Operator/Guru.2. Buka Riwayat Pesan.	Sistem menampilkan tabel riwayat yang berisi nama siswa, kelas, nomor WhatsApp wali, pesan, waktu kirim, dan status.	Riwayat pesan berhasil ditampilkan.	Pass
	TC-07.2 Memfilter riwayat berdasarkan kelas/tanggal	1. Pilih filter kelas/tanggal.2. Klik Filter.	Sistem memuat ulang tabel berdasarkan parameter filter.	Data berhasil difilter sesuai parameter.	Pass
	TC-07.3Guru tidak dapat melihat riwayat kelas lain	1. Login sebagai Guru.2. Buka Riwayat Pesan.3. Amati data.	Sistem tidak menampilkan pesan dari kelas yang bukan diampu Guru.	Data kelas lain tidak ditampilkan.	Pass
	TC-07.4Belum ada pesan yang pernah dikirim	1. Login pada akun/periode yang belum memiliki riwayat	Sistem menampilkan tabel kosong dengan pesan belum ada riwayat pengiriman pesan.	Pesan kosong ditampilkan sesuai kondisi.	Pass
	TC-07.5Filter tidak menghasilkan data	1. Pilih filter kelas/tanggal yang tidak memiliki riwayat.2. Klik Filter.	Sistem menampilkan tabel kosong dengan pesan tidak ada data pada filter yang dipilih.	Pesan tidak ada data ditampilkan.	Pass

Seluruh 5 kasus uji UC-07: Melihat Riwayat Pengiriman Pesan (Tabel 5.44) sesuai hasil yang diharapkan dan dinyatakan valid. Pengujian ini dilakukan oleh Aktor Operator dan Guru, dengan Requirement terkait FR-35 s.d. FR-38.

Tabel 5.45 Black Box Testing Kelola Absensi

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
TS-08Kelola Absensi (Pencatatan & Koreksi)	TC-08.1Mencatat absensi dengan seluruh status terisi	1. Buka menu Absensi dan pilih kelas.2. Pilih tanggal.3. Isi status H/I/S/A seluruh siswa.4. Klik Simpan Absensi.	Sistem memvalidasi kelengkapan, menyimpan absensi, dan menampilkan ringkasan Hadir/Izin/Sakit/Alpa.	Data absensi berhasil disimpan dan ringkasan ditampilkan.	Pass
	TC-08.2Mengirim pesan saat siswa Alpa/Izin/Sakit	1. Isi siswa dengan status Alpa, Izin, dan Sakit secara bergantian.2. Simpan absensi.3. Amati pengiriman pesan untuk masing-masing status.	Sistem mengirim pesan WhatsApp melalui Fonnte dan mencatat status pengiriman.	Pesan dikirim dan status tercatat.	Pass
	TC-08.3Mencegah duplikasi absensi	1. Buka menu Absensi, pilih kelas dan tanggal yang sudah pernah dicatat.	Sistem menampilkan form absensi yang sudah terisi beserta tombol Edit, bukan form kosong untuk pencatatan baru, sehingga mencegah duplikasi data.	Form terisi dan tombol Edit muncul, tidak ada opsi input baru.	Pass
	TC-08.4Default status otomatis terisi	1. Buka form absensi.2. Amati status seluruh siswa tanpa mengubah apa pun.	Seluruh siswa sudah memiliki status default (Hadir) secara otomatis, tidak ada siswa yang berstatus kosong.	Seluruh siswa menampilkan status default Hadir.	Pass
	TC-08.5Mengisi absensi pada tanggal setelah hari ini	1. Buka menu Absensi.2. Coba pilih tanggal setelah hari ini pada filter.	Sistem menonaktifkan input tanggal setelah hari ini pada filter.	Tanggal setelah hari ini tidak dapat dipilih.	Pass
	TC-08.6Mengisi absensi selain di hari aktif periode	1. Guru mencoba mencatat absensi di luar rentang periode aktif.	Sistem menampilkan pesan tidak bisa mengisi absensi karena hari ini bukan hari aktif.	Sistem menolak pencatatan.	Pass
	TC-08.7Mengisi absensi pada hari libur	1. Guru membuka Absensi pada tanggal yang termasuk hari libur.2. Mencoba mencatat absensi.	Sistem menampilkan pesan tidak bisa mengisi absensi karena hari ini merupakan hari libur.	Sistem menolak pencatatan.	Pass
	TC-08.8Mengoreksi absensi	1. Buka Absensi kelas .2. Klik Edit.3. Pilih status baru.4. Klik Simpan.	Sistem memperbarui absensi dan menampilkan pesan berhasil.	Data absensi berhasil diperbarui.	Pass
	TC-08.9Pesan saat koreksi menjadi Alpa/Izin/Sakit/hadir	1. Ubah status siswa menjadi Alpa (atau	Sistem mengirim WhatsApp kepada orang	Pesan dikirim dan hasil tercatat.	Pass

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
		Izin/Sakit/hadir).2. Klik Simpan.	tua siswa dan mencatat hasil pengiriman.		
	TC-08.10 Status baru sama dengan status lama	1. Buka Edit.2. Pilih status yang sama.3. Klik Simpan.	Sistem tetap menyimpan perubahan	Sistem tetap menyimpan perubahan absensi dan menampilkan pesan	Pass
	TC-08.11 Nomor WhatsApp tidak valid/pengiriman gagal	1. Simpan absensi dengan siswa berstatus Alpa/Izin/Sakit yang nomor wali tidak valid.	Data absensi tetap disimpan dan status pesan dicatat sebagai Gagal.	Absensi tersimpan dan status pesan Gagal tercatat.	Pass
	TC-08.12 Tidak ada siswa berstatus Alpa/Izin/Sakit (seluruh Hadir)	1. Simpan absensi tanpa siswa berstatus Alpa/Izin/Sakit (seluruh siswa Hadir).	Sistem tidak menjalankan pengiriman pesan dan absensi tetap tersimpan.	Absensi tersimpan tanpa proses pesan.	Pass
	TC-08.13 Guru tanpa relasi kelas mengakses Absensi	1. Login dengan Guru tanpa relasi kelas.2. Buka menu Absensi.	Sistem menampilkan pesan bahwa Guru belum terdaftar pada kelas.	Pesan tersebut berhasil ditampilkan.	Pass

Seluruh 13 kasus uji UC-08: Kelola Absensi (Pencatatan & Koreksi) sesuai harapan dan dinyatakan valid. Kasus nomor 5-7 menerapkan Boundary Value Analysis pada batas tanggal setelah hari ini, di luar periode aktif, dan hari libur, sedangkan TC-08.9 menguji koreksi status bermasalah kembali menjadi Hadir dan memastikan orang tua tetap menerima Pesan klarifikasi. Pengujian ini dilakukan oleh Aktor Guru, dengan Requirement terkait FR-39 s.d. FR-45.

5.5.2 Rekapitulasi Hasil Black Box Testing

Tabel 5.46 Rekapitulasi Hasil Black Box Testing

No	Kode	Skenario Pengujian	Jumlah Test Case	Pass	Fail	Persentase
1	TS-00	Pengujian <i>Login</i> & Lupa <i>Password</i>	14	14	0	100%
2	TS-01	Pengujian Kelola Data Guru (UC-01)	10	10	0	100%
3	TS-02	Pengujian Kelola Data Siswa (UC-02)	14	14	0	100%
4	TS-03	Pengujian Kelola Data Kelas (UC-03)	10	10	0	100%
5	TS-04	Pengujian Kelola Data Periode (UC-04)	9	9	0	100%
6	TS-05	Pengujian Lihat Data Rekap (UC-05)	3	3	0	100%
7	TS-06	Pengujian Cetak Laporan Rekap (UC-06)	2	2	0	100%
8	TS-07	Pengujian Lihat Riwayat Pengiriman Pesan (UC-07)	5	5	0	100%
9	TS-08	Pengujian Kelola Absensi – Pencatatan & Koreksi (UC-08)	13	13	0	100%
		TOTAL	80	80	0	100%

Berdasarkan hasil pengujian black box, seluruh 80 kasus uji yang tercakup dalam 9 skenario pengujian (TS-00 sampai TS-08) meliputi pengujian fitur Login & Lupa Password serta 8 use case (UC-01 sampai UC-08) pada Website Absensi Siswa dengan WhatsApp *Gateway* menghasilkan keluaran sesuai spesifikasi kebutuhan fungsional, dengan persentase keberhasilan 100%, termasuk autentikasi berbasis peran dan integrasi pengiriman pesan WhatsApp melalui API Fonnte.

5.5.3 Pengujian Penerimaan Pengguna (User Acceptance Testing)

Selain Black Box Testing oleh developer, sistem juga diuji menggunakan User Acceptance Testing (UAT) untuk memvalidasi kelayakan sistem dari sudut pandang pengguna akhir (Operator, Guru, Kepala Sekolah). Jika Black Box Testing berfokus pada kesesuaian fungsional teknis, UAT menilai apakah sistem benar-benar dapat digunakan untuk pekerjaan sehari-hari, sesuai posisi UAT sebagai user testing menurut Sommerville.

Pelaksanaan UAT pada penelitian ini mengikuti enam tahap acceptance testing menurut Sommerville, sebagai berikut:

1) Define Acceptance Criteria

Pada tahap ini ditentukan aktor pengujian (Operator, Guru, Kepala Sekolah) serta kriteria kelayakan sistem, meliputi seluruh functional requirement (FR-01 s.d. FR-45) berjalan sesuai skenario, serta nonfunctional requirement.

Kode	Aktor	Peran dalam UAT
AC-01	Operator	Menguji seluruh menu data master (guru, siswa, kelas, periode) dan laporan
AC-02	Guru	Menguji pencatatan & koreksi absensi, dan rekap kelas yang diampu, dan pesan kepada orang tua
AC-04	Kepala Sekolah	Menguji akses rekap & cetak laporan lintas kelas

Kriteria penerimaan umum :

1. Semua functional requirement (FR-01–FR-45) berjalan sesuai hasil yang diharapkan pada skenario pengujian.
2. Fitur Lupa Password (FR-05) mengirim link perubahan password yang valid melalui WhatsApp, dan link tersebut hanya berlaku sekali atau sebelum kedaluwarsa.
3. Fitur Reset Periode tidak pernah terpicu tanpa konfirmasi eksplisit dari Operator, mengingat dampaknya menghapus data secara permanen.
4. Setiap role hanya dapat mengakses data sesuai hak aksesnya (NFR-01): Guru tidak bisa melihat kelas lain, tidak memiliki menu Kelola Data Siswa, dan hanya bisa mengelola absensi kelas yang diampu sendiri seperti menu data guru, data siswa, data kelas, dan periode tahun ajaran hanya muncul untuk role Operator.
5. Antarmuka sistem sederhana dan mudah digunakan (NFR-02) tanpa memerlukan pelatihan khusus bagi Operator, Guru, maupun Kepala Sekolah.

6. Sistem dapat diakses dari smartphone, laptop, maupun komputer/desktop tanpa gangguan tampilan (NFR-03).
7. Waktu respons simpan absensi beserta trigger pesan tergolong cepat, tidak membuat guru menunggu lama (NFR-04).
8. Pesan WhatsApp ke orang tua siswa terkirim otomatis saat status Alpa, Izin, atau Sakit tercatat baik pada pencatatan awal maupun saat koreksi absensi melalui API Fonnte (NFR-05), dengan status pengiriman (Terkirim/Gagal) selalu tercatat di riwayat.
9. Data absensi, rekap, dan riwayat pengiriman pesan yang ditampilkan sistem konsisten dengan data pada basis data (NFR-06), tanpa selisih atau duplikasi.
10. Tidak ditemukan error kritis (500/aplikasi crash) selama pengujian.

2) Plan Acceptance Testing

Tahap ini menentukan lingkup pengujian (Login & Lupa Password, serta UC-01 s.d. UC-08), jumlah dan komposisi peserta uji, lingkungan pengujian, data uji yang digunakan, risiko yang diantisipasi beserta bentuk antisipasinya, serta jadwal dan estimasi durasi pengujian.

Item	Detail
Lingkup	<i>Login & Lupa Password</i> (FR-01–FR-05) + UC-01 s.d. UC-08 (FR-06–FR-45)
Peserta uji	1 Operator, minimal 2 Guru (kelas berbeda), 1 Kepala Sekolah, 1 nomor WA aktif sebagai "orang tua siswa" uji coba, dan 1 nomor WA aktif milik salah satu akun uji (Operator/Guru/Kepala Sekolah) untuk menguji fitur <i>Lupa Password</i>
Lingkungan	di sekolah menggunakan laptop operator, HP guru sesuai anjuran Sommerville agar user testing merefleksikan kondisi kerja nyata
Data uji	Data dummy yang menyerupai data asli (nama guru/siswa fiktif, no. WA yang benar-benar aktif untuk mengetes pengiriman pesan pesan maupun link lupa password)
Risiko yang diantisipasi	Guru belum familiar dengan antarmuka sistem yang baru, karena proses absensi sebelumnya masih manual. Antisipasi: briefing singkat sebelum sesi pengujian dimulai.
Jadwal & Durasi	Dilaksanakan pada 24 Agustus 2026 di SDN Cibirungkulon 02, dengan rincian: Sesi 1 (08.00–08.45 WIB) Operator, Sesi 2 (09.00–09.45 WIB) Guru, Sesi 3 (10.00–10.45 WIB) Kepala Sekolah, dan Sesi 4/gabungan (10.45–11.30 WIB) untuk alur lintas-role (guru input Alpa/Izin/Sakit → wali terima WA → operator lihat riwayat).

3) Derive Acceptance Tests

Langkah pengujian pada setiap skenario UAT diturunkan dari skenario Use Case pada Bab 4 (Tabel 4.10–4.18) dan disederhanakan dari test step Black Box Testing developer pada Bab 5.5.1 (Tabel 5.37–5.45), sehingga alur pengujian yang dijalankan pengguna tetap konsisten dan tertelusur terhadap kebutuhan fungsional (FR) yang telah diverifikasi sebelumnya.

Tabel 5.47 UAT Login, Hak Akses & Lupa Password

Aktor	Operator, Guru, Kepala Sekolah
Prasyarat	Akun masing-masing role sudah terdaftar dengan nomor WhatsApp aktif
Langkah	1) <i>Login</i> dengan username/NIP & password valid. 2) Ulangi dengan password salah. 3) Klik "Lupa Password", masukkan nomor WA terdaftar, klik "Kirim". 4) Ulangi dengan nomor tidak terdaftar dan nomor kosong. 5) Buka link reset dari WhatsApp, isi password baru & konfirmasi yang cocok, klik "Simpan". 6) Ulangi dengan konfirmasi yang tidak cocok. 7) Coba buka link yang sudah kedaluwarsa/terpakai. 8) <i>Login</i> dengan password baru, lalu coba login dengan password lama.
Data Uji	<i>Username/NIP</i> & password valid tiap role, password salah, nomor WA terdaftar & tidak terdaftar.
Hasil Diharapkan	<i>Login</i> valid: masuk dashboard sesuai role. <i>Login</i> gagal: muncul pesan error. Nomor WA terdaftar: link reset terkirim. Nomor tidak terdaftar/kosong: pesan error sesuai kondisi. Password baru & konfirmasi cocok: berhasil diubah. Tidak cocok: ditolak, diminta perbaikan. Link kedaluwarsa/terpakai: ditolak. Password baru berhasil login, password lama ditolak.
Kriteria Sukses	Setiap role hanya melihat menu sesuai haknya (Operator: data master; Guru: absensi kelas sendiri, Kepala Sekolah: rekap semua kelas). Link reset password valid dan hanya berlaku sekali/sebelum kedaluwarsa: kritis karena FR-05 adalah fitur baru.

Tabel 5.48 UAT Kelola Data Guru

Prasyarat	Operator sudah login
Langkah	1) Tambah guru baru (isi semua field, termasuk kelas yang diampu), simpan. 2) Ubah salah satu data guru. 3) Coba tambah guru dengan username yang sudah terdaftar. 4) Hapus guru yang menjadi wali kelas suatu kelas (tidak punya riwayat absensi), lalu konfirmasi. 5) Cek Log Aktivitas mencatat seluruh aksi. 6) Coba hapus guru yang memiliki riwayat absensi (pernah mengisi/mengoreksi absensi), pastikan sistem menolak dan menampilkan pesan yang jelas.
Data Uji	1 data guru baru lengkap, 1 data untuk diedit, 1 username terduplikasi, 1 guru wali kelas (tanpa riwayat absensi) untuk uji hapus, 1 guru dengan riwayat absensi untuk uji penolakan hapus.
Hasil Diharapkan	Tambah/ubah tersimpan di daftar. <i>Username</i> duplikat ditolak dengan pesan error. Hapus guru wali kelas (tanpa riwayat absensi) memunculkan pop up konfirmasi umum, setelah dikonfirmasi, data guru terhapus dan kelas yang diampunya otomatis menjadi tanpa wali kelas (bukan ikut terhapus). Hapus guru yang memiliki riwayat absensi ditolak dengan pesan 'Pengguna tidak dapat dihapus karena memiliki riwayat absensi.' Log Aktivitas mencatat waktu, aktor, dan jenis aksi.
Kriteria Sukses	Tidak ada data gagal tersimpan tanpa pesan error yang jelas, field kosong dan username duplikat ditolak dengan pesan informatif.

Tabel 5.49 UAT Kelola Data Siswa

Aktor	Operator
Langkah	1) Tambah data siswa lengkap, termasuk data orang tua/wali dengan nomor WA yang valid & aktif. 2) Ubah data siswa. 3) Coba simpan dengan NISN yang sudah terdaftar. 4) Coba simpan dengan Field dengan tanda (*) wajib diisi. 5) Hapus siswa yang punya data absensi, lalu konfirmasi.
Data Uji	Nomor WA aktif milik tester (berperan sebagai "orang tua siswa").

Aktor	Operator
Hasil Diharapkan	Data tersimpan. NISN duplikat & field kosong ditolak dengan pesan validasi. Hapus siswa ber-data absensi memunculkan konfirmasi, setelah dikonfirmasi, data absensi terkait ikut terhapus.
Kriteria Sukses	Nomor WA yang disimpan di tahap ini nantinya benar-benar menerima pesan pada UAT-08.

Tabel 5.50 UAT Kelola Data Kelas

Langkah	1) Tambah kelas baru (nama kelas, wali kelas). 2) Coba tambah kelas dengan nama yang sudah terdaftar (nama kelas bersifat unik secara keseluruhan, karena data kelas tidak dikaitkan dengan periode, sistem hanya memiliki satu periode aktif yang diubah, bukan ditambah per tahun ajaran). 3) Ubah data kelas (mis. ganti wali kelas). 4) Amati dropdown wali kelas, pastikan guru yang saat ini sudah menjadi wali kelas kelas lain tidak muncul sebagai pilihan. 5) Coba hapus kelas yang punya absensi aktif.
Data Uji	1 data kelas baru, 1 nama kelas yang sengaja diduplikasi dari kelas yang sudah ada, 1 guru yang saat ini sudah menjadi wali kelas kelas lain.
Hasil Diharapkan	Duplikasi nama kelas ditolak dengan pesan jelas, karena nama kelas bersifat unik secara keseluruhan dan tidak dikaitkan dengan periode/tahun ajaran tertentu. Guru yang saat ini sudah menjadi wali kelas kelas lain tidak muncul di dropdown pilihan wali kelas. Hapus kelas ber-absensi aktif ditolak dengan pesan agar kelas diarsipkan atau data absensinya dihapus dahulu.
Kriteria Sukses	Pesan error informatif, bukan pesan generik/teknis.

Tabel 5.51 UAT Kelola Periode Tahun Ajaran

Langkah	1) Tambah periode dengan tanggal mulai > tanggal selesai (harus ditolak). 2) Tambah periode valid untuk Semester 1 & 2. 3) Coba isi tanggal mulai Semester 2 lebih awal/sama dengan akhir Semester 1 (harus ditolak). 4) Tambah periode baru yang tanggalnya mencakup hari ini, cek sistem otomatis mengenalinya sebagai periode aktif. 5) Tambahkan hari libur pada periode aktif. 6) Coba catat absensi di luar rentang periode aktif atau di hari libur. 7) Klik "Reset Periode" & konfirmasi, cek seluruh data terkait terhapus. 8) Buka menu Log Aktivitas dan periksa aksi mana saja yang tercatat.
Data Uji	1 periode dengan tanggal tidak valid, 1 periode valid Semester 1 & 2, tanggal di luar periode aktif, tanggal hari libur.
Hasil Diharapkan	Tanggal tidak valid ditolak dengan pesan validasi. Periode aktif ditentukan otomatis berdasarkan tanggal berjalan, tanpa perlu diaktifkan manual. Pencatatan absensi di luar periode aktif atau hari libur ditolak. Reset Periode memunculkan konfirmasi, setelah dikonfirmasi, seluruh data periode/absensi/rekap terkait terhapus. Log Aktivitas mencatat perubahan (ubah) periode, pembuatan periode awal dan Reset Periode saat ini belum tercatat di Log Aktivitas.
Kriteria Sukses	Guru tidak bisa menyimpan absensi di luar periode aktif atau di hari libur: kritis untuk integritas data absensi. Reset Periode hanya bisa dijalankan sengaja oleh Operator (ada konfirmasi eksplisit): kesalahan klik berdampak kehilangan data permanen.

Tabel 5.52 UAT Lihat Rekap Absensi

Langkah	1) Sebagai Guru, cek hanya kelas sendiri yang muncul di filter Rekap. 2) Sebagai Kepala Sekolah/Operator, cek semua kelas muncul. 3) Coba filter per hari, bulan, dan semester. 4) Coba input tanggal manual. 5) Pilih filter yang tidak menghasilkan data.
Data Uji	Data absensi hasil input UAT-08, kombinasi filter hari/bulan/semester/tanggal manual, termasuk filter tanpa hasil.
Hasil Diharapkan	Rekap menampilkan Nama Siswa, Kelas, Hadir, Izin, Sakit, Alpa, Total Hari, dan % Kehadiran sesuai data UAT-08. Setiap mode filter menghasilkan data yang sesuai. Filter tanpa hasil menampilkan pesan "Tidak ada data absensi pada periode yang dipilih".
Kriteria Sukses	Angka rekap cocok 100% dengan data absensi yang diinput guru.

Tabel 5.53 UAT Cetak Laporan Rekap

Langkah	1) Dari halaman rekap, klik cetak/export. 2) Buka file Excel hasil unduhan. 3) Coba cetak saat data rekap kosong.
Data Uji	Data rekap hasil UAT-05 dengan hasil ada dan filter tanpa hasil (kosong).
Hasil Diharapkan	File Excel tetap terunduh meski data kosong (tidak ditolak), isi laporan sesuai data yang ada (baris data kosong, header & judul tetap tampil).
Kriteria Sukses	File bisa dibuka tanpa corrupt dan datanya akurat.

Tabel 5.54 UAT Melihat Riwayat Pengiriman Pesan

Langkah	1) Buka menu Riwayat Pesan setelah UAT-08 dijalankan. 2) Cek status Terkirim/Gagal tampil dengan benar. 3) Filter berdasarkan kelas dan/atau tanggal. 4) Pilih filter yang tidak menghasilkan data. 5) Sebagai Guru, coba akses riwayat pesan kelas lain.
Data Uji	Data pesan hasil UAT-08, filter kelas/tanggal dengan dan tanpa hasil.
Hasil Diharapkan	Riwayat menampilkan Nama Siswa, Kelas, Nomor WA Wali, Pesan, Waktu Kirim, dan Status sesuai kejadian di HP wali. Filter tanpa hasil menampilkan pesan "Tidak ada data pada filter yang dipilih". Guru tidak bisa melihat data pesan kelas lain.
Kriteria Sukses	Status "Terkirim" di sistem benar-benar sesuai dengan pesan yang diterima di WhatsApp tester.

Tabel 5.55 UAT Absensi, Koreksi & Pesan WhatsApp Otomatis

Prasyarat	Guru login, terdaftar di minimal 1 kelas, siswa di kelas itu punya nomor WA wali yang valid (dari UAT-02)
Langkah	1) Pilih kelas & tanggal hari ini di menu Absensi. 2) Amati status default seluruh siswa sebelum diubah. 3) Isi status semua siswa: minimal 1 Alpa/Izin/Sakit dan 1 siswa dengan nomor WA sengaja tidak valid, lalu simpan. 4) Cek HP orang tua siswa menerima pesan WA berisi nama, kelas, tanggal. 5) Cek Riwayat Pesan mencatat status "Gagal" untuk nomor tidak valid, namun absensi tetap tersimpan. 6) Buka kembali form absensi kelas & tanggal yang sama, amati tampilan form. 7) Coba pilih tanggal mendatang lalu simpan (harus ditolak sesuai FR-45). 8) Klik Edit pada absensi, ubah status Hadir menjadi Alpa/Izin/Sakit, simpan. 9) Coba ubah ke status yang sama dengan sebelumnya (data tidak boleh berubah)

Prasyarat	Guru login, terdaftar di minimal 1 kelas, siswa di kelas itu punya nomor WA wali yang valid (dari UAT-02)
	10) Kembali ke absensi hari ini, klik Edit pada siswa yang tadi dikoreksi menjadi Alpa/Izin/Sakit (langkah 8), ubah kembali statusnya menjadi Hadir, lalu simpan. Cek HP orang tua siswa menerima Pesan klarifikasi.
Data Uji	Minimal 1 nomor WA aktif di HP tester, 1 nomor WA sengaja tidak valid untuk menguji status Gagal.
Hasil Diharapkan	Seluruh siswa memiliki status default (Hadir) secara otomatis. Absensi tersimpan, pesan WA otomatis terkirim untuk status Alpa/Izin/Sakit, kegagalan kirim tidak menggagalkan simpan, hanya tercatat "Gagal". Membuka kembali kelas & tanggal yang sama menampilkan form yang sudah terisi beserta tombol Edit (bukan form kosong), tanggal mendatang ditolak. Koreksi ke Alpa/Izin/Sakit memicu WA baru & tercatat log, koreksi kembali ke Hadir memicu WA klarifikasi & tercatat log, koreksi status sama tetap menampilkan pesan.
Kriteria Sukses	Skenario end-to-end paling menentukan kelulusan UAT: pesan WA yang gagal atau salah kirim membuat sistem tidak boleh dinyatakan lolos meski fitur lain sempurna. Pembatasan pengisian absensi hanya pada hari aktif (FR-45) bersifat kritis untuk integritas data: absensi "masa depan" tidak boleh bisa tersimpan.

4) Run Acceptance Tests

Tahap ini merupakan pelaksanaan skenario pengujian yang telah diturunkan pada tahap sebelumnya (Tabel 5.47–5.55) secara langsung oleh pengguna nyata di lingkungan SDN Cibitungkulon 02, yaitu Operator, Guru, dan Kepala Sekolah, sesuai jadwal, lingkungan, dan durasi yang telah ditentukan pada tahap *Plan Acceptance Testing* (Tahap 2). Setiap responden menjalankan seluruh langkah pengujian pada perangkatnya masing-masing (laptop untuk Operator, HP untuk Guru dan Kepala Sekolah), kemudian mengisi kuesioner penerimaan berupa pernyataan dengan pilihan jawaban Ya/Tidak beserta kolom Keterangan, berdasarkan pengalaman aktual saat menjalankan sistem. Hasil pengisian kuesioner untuk masing-masing role disajikan pada Tabel 5.56 (Operator), Tabel 5.57 (Guru), dan Tabel 5.58 (Kepala Sekolah), dengan rekapitulasi keseluruhan hasil pada Tabel 5.59.

Tabel 5.56 Hasil Pengujian Penerimaan Pengguna untuk Operator (UAT)

No	Pernyataan	Ya	Tidak	Keterangan
1	Saya dapat login dan masuk ke sistem dengan mudah	1		
2	Saya dapat mengelola data guru (tambah/ubah/hapus) dengan mudah dan sesuai kebutuhan	1		
3	Saya dapat mengelola data siswa (tambah/ubah/hapus) dengan mudah dan sesuai kebutuhan	1		
4	Saya dapat mengelola data kelas (tambah/ubah/hapus) dengan mudah dan sesuai kebutuhan	1		
5	Saya dapat mengelola periode dan hari libur sekolah dengan mudah dan sesuai kebutuhan	1		
6	Saya dapat melihat rekap absensi seluruh kelas dengan jelas dan datanya sesuai	1		
7	Saya dapat mencetak/mengunduh laporan rekap dalam format Excel dengan mudah dan sesuai kebutuhan	1		

No	Pernyataan	Ya	Tidak	Keterangan
8	Saya dapat melihat dan mencari riwayat pengiriman notifikasi WhatsApp dengan jelas	1		
9	Tampilan sistem mudah dipahami tanpa perlu penjelasan panjang	1		
10	Sistem nyaman diakses baik melalui HP/smartphone maupun laptop/komputer	1		
11	Sistem merespons dengan cepat saat saya menyimpan atau mengelola data	1		
12	Saya hanya dapat mengakses menu dan data yang sesuai dengan peran saya sebagai Operator	1		
13	Data yang ditampilkan sistem akurat dan dapat sesuai kebutuhan	1		
14	Secara keseluruhan, sistem ini mudah digunakan dan sesuai dengan kebutuhan pekerjaan saya sebagai Operator	1		

Tabel 5.56 berikut menyajikan hasil pengujian penerimaan yang diisi oleh responden Operator, berdasarkan 14 pernyataan yang mengacu pada kriteria penerimaan yang telah ditetapkan pada tahap Define Acceptance Criteria.

Tabel 5.57 Hasil Pengujian Penerimaan Pengguna untuk Guru (UAT)

No	Pernyataan	Ya	Tidak	Keterangan
1	Saya dapat login dan masuk ke sistem dengan mudah	10		
2	Saya dapat mengelola data siswa (tambah/ubah/hapus) di kelas yang saya ampu dengan mudah dan sesuai kebutuhan	10		
3	Saya dapat mencatat dan mengoreksi absensi kelas dengan mudah, cepat, dan sesuai kebutuhan.	10		
4	Pesan WhatsApp yang diterima orang tua sudah sesuai	10		
5	Saya dapat melihat dan mencari riwayat pengiriman pesan WhatsApp untuk kelas saya dengan jelas	10		
6	Saya dapat melihat rekap absensi untuk kelas saya dengan jelas	10		
7	Saya dapat mencetak/mengunduh laporan rekap kelas saya dalam format Excel dengan mudah dan sesuai kebutuhan	10		
8	Tampilan sistem mudah dipahami tanpa perlu penjelasan panjang	10		
9	Sistem nyaman diakses baik melalui HP/smartphone maupun laptop/komputer	10		
10	Sistem merespons dengan cepat saat saya menyimpan absensi	10		
11	Saya hanya dapat mengakses menu dan data yang sesuai dengan peran saya sebagai Guru	10		
12	Data yang ditampilkan sistem akurat dan sudah sesuai dengan kebutuhan	10		
13	Secara keseluruhan, sistem ini mudah digunakan dan sesuai dengan kebutuhan pekerjaan saya sebagai Guru	10		

Tabel 5.57 berikut menyajikan hasil pengujian penerimaan yang diisi oleh 10 responden Guru, berdasarkan 13 pernyataan yang mengacu pada kriteria penerimaan yang sama.

Tabel 5.58 Hasil Pengujian Penerimaan Pengguna untuk Kepala Sekolah (UAT)

No	Pernyataan	Ya	Tidak	Keterangan
1	Saya dapat login dan masuk ke sistem dengan mudah	1		
2	Saya dapat melihat rekap absensi seluruh kelas di sekolah dengan mudah dan sesuai kebutuhan	1		
3	Saya dapat mencetak/mengunduh laporan rekap dalam format Excel dengan mudah dan sesuai kebutuhan	1		
4	Tampilan sistem mudah dipahami tanpa perlu penjelasan panjang	1		
5	Sistem nyaman diakses baik melalui HP/smartphone maupun laptop/komputer	1		
6	Sistem merespons dengan cepat saat saya membuka rekap absensi atau laporan	1		
7	Saya hanya dapat mengakses menu dan data yang sesuai dengan peran saya sebagai Kepala Sekolah	1		
8	Data yang ditampilkan sistem akurat dan sesuai kebutuhan	1		
9	Secara keseluruhan, sistem ini mudah digunakan dan sesuai dengan kebutuhan pekerjaan saya sebagai Kepala Sekolah	1		

Tabel 5.58 berikut menyajikan hasil pengujian penerimaan yang diisi oleh responden Kepala Sekolah, berdasarkan 9 pernyataan yang mengacu pada kriteria penerimaan yang sama.

Tabel 5.59 Rekapitulasi Hasil Pengujian Penerimaan Pengguna (UAT)

No	Kelompok Penguji	Jumlah Responden	Jumlah Pernyataan	Ya	Tidak	Persentase
1	Operator	1	14	14	0	100%
2	Guru	10	13	130	0	100%
3	Kepala Sekolah	1	9	9	0	100%
	Total Keseluruhan	12	36	153	0	100%

5) Negotiate Test Results

Dari 36 pernyataan pada Tabel 5.56 s.d. 5.59 yang diujikan kepada 12 responden (1 Operator, 10 Guru, 1 Kepala Sekolah), seluruh 153 jawaban (100%) adalah "Ya" tanpa jawaban "Tidak", sehingga tidak ada masalah yang perlu dinegosiasikan.

6) Reject/Accept System

Karena seluruh kriteria penerimaan umum yang ditetapkan pada tahap Define Acceptance Criteria (Tahap 1) telah terpenuhi sepenuhnya (Tabel 5.59), Sistem Website Absensi Siswa dengan WhatsApp *Gateway* dinyatakan diterima (accepted) secara penuh (full acceptance) oleh SDN Cibitungkulon 02.

BAB 6

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan hasil pembangunan serta pengujian *black box* dan *User Acceptance Testing* (UAT) terhadap Website Absensi Siswa dengan WhatsApp Gateway di SDN Cibitungkulon 02, diperoleh kesimpulan sebagai berikut:

1. Website absensi siswa berhasil dibangun dan mendigitalisasi proses pencatatan kehadiran, menggantikan pencatatan manual menggunakan buku absensi dan Microsoft Excel di SDN Cibitungkulon 02. Seluruh pengujian *black box* berjalan valid, dan hasil UAT menunjukkan tingkat penerimaan 100% dari pengguna.
2. Mekanisme rekap kehadiran siswa secara otomatis berhasil dirancang dan diimplementasikan pada sistem, dengan tujuan mengurangi proses rekapitulasi yang sebelumnya dilakukan secara manual. Pengujian menunjukkan fitur rekap dan pencetakan laporan absensi harian, bulanan, dan semester berfungsi dengan baik, sehingga berpotensi mengurangi perhitungan dan penyusunan laporan manual apabila sistem diimplementasikan secara langsung di sekolah.
3. WhatsApp Gateway berhasil diintegrasikan untuk mengirim pesan otomatis kepada orang tua siswa setiap kali status kehadiran tercatat atau dikoreksi menjadi Alpa, Izin, atau Sakit, termasuk saat dikoreksi kembali menjadi Hadir. Pesan terbukti terkirim dan tercatat dengan baik, sehingga fitur ini berfungsi sebagai verifikasi keakuratan data tanpa memerlukan tindakan manual tambahan dari guru.

Dengan tercapainya ketiga tujuan tersebut, serta hasil User Acceptance Testing yang menyatakan penerimaan penuh (full acceptance) oleh SDN Cibitungkulon 02, Website Absensi Siswa dengan Integrasi WhatsApp Gateway ini dapat disimpulkan sudah siap dan layak digunakan untuk mendukung pencatatan dan pelaporan kehadiran siswa.

6.2 Saran

Beberapa saran untuk pengembangan sistem selanjutnya adalah sebagai berikut:

1. Menambahkan pencatatan kehadiran otomatis menggunakan RFID, sidik jari (*fingerprint scanner*), atau pengenalan wajah (*face recognition*), mengingat pencatatan saat ini masih dilakukan manual satu per satu oleh guru, guna meningkatkan efisiensi dan akurasi.
2. Meningkatkan ke tier WhatsApp Gateway (Fonnte) berbayar atau menambahkan penyedia cadangan (*multi-provider*), karena tier gratis saat ini memiliki kuota terbatas 1.000 pesan/bulan sehingga notifikasi status Hadir belum dapat dikiri

DAFTAR PUSTAKA

- Aizah, N., & Santoso, F. (2024). Implementasi sistem absensi siswa berbasis web menggunakan WhatsApp *gateway* di SDN 2 Seletreng. Seminar Nasional Riset dan Inovasi Teknologi (SEMNAS RISTEK).
- Al-Husain, A. A. (2017). Perancangan sistem absensi online menggunakan Android guna mempercepat proses kehadiran karyawan pada PT. Sintech Berkah Abadi. *Technomedia Journal (TMJ)*, 2(1).
- Baha'uddin, M., Susanto, A., & Hamdani, A. (2025). Implementasi RFID pada sistem informasi presensi siswa di MAS Al-Huda Kangayan dengan pesan WhatsApp *gateway*. *RABIT: Jurnal Teknologi dan Sistem Informasi Univrab*, 10(2).
- Khoeriyah, Y. S., Indah, R. N., & Ruqayah, F. (2021). Pemanfaatan layanan WhatsApp *gateway* sebagai sistem pesan pinjaman (SINOPI) di Dinas Kearsipan dan Perpustakaan Kota Pekalongan. *Pustabiblia: Journal of Library and Information Science*, 5(1), 104.
- Lembaga Penjaminan Mutu Pendidikan. (2020). Membuat website pribadi (guru) sebagai ruang kelas pembelajaran daring. Kementerian Pendidikan dan Kebudayaan.
- Masse, M. (2011). *REST API design rulebook*. O'Reilly Media.
- Nababan, P., Jamaluddin, Perangin-angin, R., & Purba, E. N. (2022). Sistem informasi absensi siswa pada SMK Negeri 1 Pantai Labu berbasis web dengan WhatsApp *gateway*. *TAMIKA: Jurnal Tugas Akhir Manajemen Informatika & Komputerisasi Akuntansi*, 2(2).
- Pressman, R. S. (2020). *Software engineering: A practitioner's approach* (9th ed.). McGraw-Hill.
- Rahi, F. M., Radjah, E. G., & Talakua, A. C. (2024). Integrasi WhatsApp *gateway* dalam website absensi berbasis website di SMA Negeri 1 Pandawai. 3rd National Seminar on Sustainable Agricultural Technology Innovation (SATI).
- Republik Indonesia. (2003). Undang-Undang Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional.
- Robbins, J. N. (2012). *Learning web design: A beginner's guide to HTML, CSS, JavaScript, and web graphics* (4th ed.). O'Reilly Media.
- Septiani, S., Reza, R., Akil, H., & Aziz, A. (2025). Jenis-jenis skala dan teknik skoring dalam penilaian psikomotorik pada mata pelajaran Pendidikan Agama Islam. *Jurnal Manajemen dan Pendidikan Agama Islam*, 3(4), 21–36.
- Solichin, A. (2008). *Pemrograman web dengan PHP dan MySQL*.

Sommerville, I. (2015). Software engineering (10th ed.). Addison-Wesley.

Stauffer, M. (2017). Laravel: Up and running. O'Reilly Media.

Tailwind CSS. (2023). Tailwind CSS documentation. Retrieved May 13, 2026, from <https://tailwindcss.com>

Zahid, M. M., & Prapanca, A. (2023). Rancang bangun website absensi menggunakan RFID dan WhatsApp untuk meningkatkan kedisiplinan dan nilai rapor siswa. Jurnal IT-EDU, 8(2).

LAMPIRAN

The first image shows the cover of a book titled "ADMINISTRASI KESISWAAN" with the subtitle "DAFTAR ABSEN PESERTA DIDIK". It includes fields for "Wali Kelas", "Kelas", "Nama Sekolah", and "Tahun Pelajaran".

The second image shows a "DAFTAR NAMA SISWA" table with columns for "No", "No Induk", "NISN", "Nama Siswa", "Jenis", "Alamat Rumah", and "Jarak ke Sekolah". It lists 30 students with their respective details.

The third image shows a "DAFTAR HADIR MURID DALAM BULAN" table for July 2025. It has columns for dates (1-31) and a "JUMLAH" column. The table is signed by the Head of School (Kepala Sekolah) and the Class Teacher (Guru Kelas).

Lampiran 1 Buku absensi



Lampiran 2 Wawancara dengan kepala sekolah operator dan guru

FORM WAWANCARA PENELITIAN TUGAS AKHIR
PEMBANGUNAN WEBSITE ABSENSI SISWA DENGAN WHATSAPP GATEWAY
 SDN Cibitungkulon 02
 Program Studi Teknik Informatika – Universitas Pasundan Bandung

A. IDENTITAS NARASUMBER

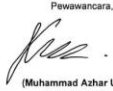
Nama Narasumber	Solihin
Jabatan / Peran	Guru
Hari / Tanggal Wawancara	23 April 2026
Tempat Wawancara	SDN Cibitung Kulon 02
Pewawancara	Muhammad Azhar Utama

1. Kondisi Sistem Absensi yang Sedang Berjalan

No.	Pertanyaan	Jawaban Narasumber
1.	Bagaimana proses pencatatan kehadiran siswa yang saat ini Bapak/Ibu lakukan setiap hari?	Dilakukan saat awal pelajaran dengan menggunakan sistem satu persatu lalu dicatat di buku absensi
2.	Berapa lama waktu yang dibutuhkan untuk mencatat kehadiran satu kelas secara manual?	± 5 - 10 menit
3.	Apakah pernah terjadi kesalahan (human error) dalam proses pencatatan?	Pernah terjadi kesalahan pencatatan karena lupa dan beberapa kali ada kendala dari orang tua
4.	Bagaimana cara Bapak/Ibu menyimpan data absensi setelah dicatat?	disimpan di kelas dan disimpan ke Excel
5.	Apakah Bapak/Ibu merasa proses absensi saat ini sudah efisien?	Belum, perlu waktu dan tenaga dan prosesnya cukup lama
6.	Bagaimana cara Orang tua siswa memberitahu guru jika siswa tidak bisa hadir?	Surat izin dan bisa pakai WhatsApp

3. Kebutuhan dan Harapan terhadap Sistem Baru

No.	Pertanyaan	Jawaban Narasumber
17.	Fitur apa saja yang Bapak/Ibu anggap paling penting dalam sistem yang akan dibangun?	- RPPKOP otomatis - Pesan ke orang tua
18.	Bagaimana preferensi tampilan antarmuka untuk guru?	Simple dan tidak rumit
19.	Apakah Bapak/Ibu memiliki perangkat seperti smartphone atau laptop yang biasa digunakan untuk keperluan sehari-hari? apa ada kendala dalam penggunaannya?	Ya, tidak ada kendala karena guru sudah menggunakan laptop dan smartphone
20.	Laporan absensi seperti apa yang paling sering dibutuhkan?	Yang paling penting dan terdapat penghitungan

Pewawancara,	Narasumber,
 (Muhammad Azhar Utama) NRP. 22.304.0077	 (Solihin S.H) NIP. 1955115022211003

Program Studi Teknik Informatika – Universitas Pasundan Bandung | NRP. 22.304.0077 | 2026

No.	Pertanyaan	Jawaban Narasumber
7.	Bagaimana biasanya Bapak/Ibu memantau pola kehadiran siswa dari waktu ke waktu, baik yang izin, sakit, maupun tanpa keterangan?	melihat catatan absensi / RPPKOP EXCEL
8.	Bagaimana cara Bapak/Ibu menginformasikan ketidakhadiran siswa kepada orang tua saat ini?	biasanya melalui WhatsApp dan dikirim atau dihubungi ketika siswa akan sakit atau berhalangan
9.	Apakah ada aturan atau tindakan khusus terkait pencatatan kehadiran siswa, baik untuk status izin, sakit, maupun tanpa keterangan?	Siswa yang izin sakit harus disertai Surat Keterangan dari orang tua
10.	Selain buku absensi kelas, apakah ada guru lain seperti guru mata pelajaran yang juga memiliki buku absensi sendiri? Bagaimana perbedaannya?	ada, buku absensi mata pelajaran, tapi absensi mata pelajaran tidak diinput ke komputer dan tidak ada absensi dari orang tua yang diinput ke komputer absensi dari orang tua
11.	Apakah Bapak/Ibu pernah lupa mengisi absensi di hari tertentu? Bagaimana penanganannya?	Pernah tapi tidak sering, dan biasanya diisi ulang berdasarkan catatan atau surat yang ada di depan kelas / di WhatsApp.

2. Proses Rekapitulasi Data Kehadiran

No.	Pertanyaan	Jawaban Narasumber
12.	Bagaimana alur rekapitulasi data absensi dari catatan harian hingga laporan ke admin sekolah?	dari buku absensi ke Excel (buku absensi) lalu diinput ke komputer
13.	Apakah Bapak/Ibu saat ini menggunakan Microsoft Excel? Seberapa lama prosesnya?	Ya, Excel, sekitar 1 jam.
14.	Seberapa sering Bapak/Ibu membuat rekap absensi?	Setiap bulan dan setiap akhir semester
15.	Apa kesulitan yang paling sering dihadapi saat membuat rekap secara manual?	kesulitan dalam jumlah kehadiran dan ketidakhadiran karena siswa tidak masuk, dan prosesnya cukup lama
16.	Bagaimana proses koreksi apabila status kehadiran yang tercatat ternyata tidak sesuai dengan kondisi siswa sebenarnya tapi kemudian membawa surat izin menyusul?	prosesnya diinput ke buku absensi dan Excel, lalu jika sudah diinput ke komputer ke komputer maka diinput ulang.

FORM WAWANCARA PENELITIAN TUGAS AKHIR
PEMBANGUNAN WEBSITE ABSENSI SISWA DENGAN WHATSAPP GATEWAY
 SDN Cibitungkulon 02
 Program Studi Teknik Informatika – Universitas Pasundan Bandung

A. IDENTITAS NARASUMBER

Nama Narasumber	Solihin
Jabatan / Peran	Guru
Hari / Tanggal Wawancara	23 April 2026
Tempat Wawancara	SDN Cibitung Kulon 02
Pewawancara	Muhammad Azhar Utama

1. Kondisi Sistem Absensi yang Sedang Berjalan

No.	Pertanyaan	Jawaban Narasumber
1.	Bagaimana proses pencatatan kehadiran siswa yang saat ini Bapak/Ibu lakukan setiap hari?	Dilakukan saat awal pelajaran dengan menggunakan sistem satu persatu lalu dicatat di buku absensi
2.	Berapa lama waktu yang dibutuhkan untuk mencatat kehadiran satu kelas secara manual?	± 5 - 10 menit
3.	Apakah pernah terjadi kesalahan (human error) dalam proses pencatatan?	Pernah terjadi kesalahan pencatatan karena lupa dan beberapa kali ada kendala dari orang tua
4.	Bagaimana cara Bapak/Ibu menyimpan data absensi setelah dicatat?	disimpan di kelas dan disimpan ke Excel
5.	Apakah Bapak/Ibu merasa proses absensi saat ini sudah efisien?	Belum, perlu waktu dan tenaga dan prosesnya cukup lama
6.	Bagaimana cara Orang tua siswa memberitahu guru jika siswa tidak bisa hadir?	Surat izin dan bisa pakai WhatsApp

No.	Pertanyaan	Jawaban Narasumber
11.	Berapa jumlah kelas dan siswa yang aktif di SDN Cibitung Kulon 02 saat ini?	12 KELAS DAN 1000 SISWA. SPHAP KELAS 25-30 SISWA.
12.	Siapa saja pengguna (user) yang perlu memiliki akses ke sistem sistem? Apa perbedaan hak aksesnya?	Jika OPERATOR DAN GURU. OPERATOR MANAJEMEN SPHAP DAN SISWA DAN GURU DAN KASIR. GURU MANAJEMEN SPHAP, DAN KASIR LISA MANAJEMEN SPHAP.
13.	Bagaimana proses pembaruan data siswa (tambah, ubah, hapus) dilakukan saat ini? Seberapa sering terjadi perubahan?	BIASAANYA MELALUI TANDA, SAKEL AG SISWA LAIN, PINDAAN ATAU KEMER PINDAAN DI KIP EXCEL.
14.	Apakah data nomor WhatsApp orang tua siswa sudah tersedia? Jika belum, bagaimana rencana pengumpulannya?	SUDAH TERSEDIA GURU DAN KASIR. SAKEL PINDAAN ATAU KEMER.
15.	Apakah ada pertimbangan khusus terkait keamanan data (login, hak akses, privasi data siswa) dalam sistem yang akan dibangun?	PINDAAN.

4. Infrastruktur dan Kesiapan Teknis Sekolah

No.	Pertanyaan	Jawaban Narasumber
16.	Apakah sekolah sudah memiliki koneksi internet yang stabil untuk mendukung penggunaan sistem berbasis web?	SUDAH ADA DAN LEBIH STABIL, MANAJEMEN WIFIS SPHAP.
17.	Perangkat apa saja yang tersedia di sekolah untuk mengakses sistem (komputer, laptop, tablet, smartphone)?	KOMPUTER DAN LAPTOP DI TU, GURU DAN MANAJEMEN SPHAP DAN KASIR. DAN PINDAAN GURU DAN MANAJEMEN HP.
18.	Apakah guru dan admin sudah terbiasa menggunakan sistem berbasis web dalam pekerjaan sehari-hari?	SUDAH TERBIA, TAPI GURU SAKEL AG KASIR KEMER DAN MANAJEMEN SPHAP.
19.	Apakah sekolah memiliki anggaran atau dukungan untuk pengembangan sistem informasi seperti ini?	ADA, TAPI MELALUI MANAJEMEN SPHAP.
20.	Adakah kendala lain yang perlu diketahui terkait implementasi sistem baru di sekolah?	MUNGKIN PERLU PELATIHAN DAN SAKEL AG.

Pewawancara, (Muhammad Azhar Utama) NRP. 22.304.0077	Narasumber, (Muhammad Aarifin S.pd) NRP. 198710122023211005
--	---

Program Studi Teknik Informatika – Universitas Pasundan Bandung | NRP. 22.304.0077 | 2026

Lampiran 4 Hasil wawancara operator

FORM WAWANCARA PENELITIAN TUGAS AKHIR Pembangunan Sistem Informasi Absensi Siswa dengan WhatsApp Gateway SDN Cibitung Kulon 02 Program Studi Teknik Informatika – Universitas Pasundan Bandung

A. IDENTITAS NARASUMBER

Nama Narasumber	Saeti, S.Pd.SD.
Jabatan / Peran	Kepala Sekolah
Hari / Tanggal Wawancara	23 APRIL 2026
Tempat Wawancara	SDN Cibitung Kulon 02
Pewawancara	Muhammad Azhar Utama

1. Kondisi Penerimaan Rekap Kehadiran Siswa Saat Ini

No.	Pertanyaan	Jawaban Narasumber
1.	Selama ini, bagaimana Bapak/Ibu menerima rekap kehadiran siswa dari operator atau guru kelas?	RECAP DIKIRIM DARI OPERATOR SPHAP DAN DIKIRIM DARI SPHAP KEMER.
2.	Dalam bentuk apa rekap tersebut biasanya diterima (misalnya dokumen cetak, file Excel, atau laporan lisan)?	DOKUMEN Cetak EXCEL.
3.	Seberapa sering Bapak/Ibu menerima rekap kehadiran siswa (harian, bulanan, atau hanya saat diminta)?	BIASAANYA BULANAN ATAU PAS GURU SPHAP DAN KASIR MANAJEMEN SPHAP DAN KASIR.
4.	Apakah rekap yang diterima selama ini sudah mencakup informasi yang dibutuhkan (misalnya jumlah hadir, izin, sakit, alfa)?	YA SUDAH.

No.	Pertanyaan	Jawaban Narasumber
5.	Apakah pernah terjadi keterlambatan atau kesulitan dalam menerima rekap kehadiran dari operator/guru?	YA, CUKUP SEDIKIT GARA-GARA PROSES PINDAAN DAN MANAJEMEN SPHAP DAN KASIR.

2. Kebutuhan terhadap Fitur Rekap dan Laporan pada Sistem

No.	Pertanyaan	Jawaban Narasumber
6.	Informasi apa saja yang Bapak/Ibu butuhkan dari rekap kehadiran siswa (misalnya jumlah hadir, izin, sakit, alfa, atau persentase kehadiran)?	Jumlah Kehadiran, Jumlah Alfa Kehadiran, dan PERSENTASAH (TERPINDAAN).
7.	Apakah Bapak/Ibu memerlukan rekap kehadiran yang dapat dilihat per hari, per bulan, maupun per semester?	YA.
8.	Apakah Bapak/Ibu perlu melihat rekap kehadiran untuk kelas tertentu saja, atau untuk seluruh kelas sekaligus?	KEDUA DARI PINDAAN DAN SPHAP KEMER.
9.	Apakah Bapak/Ibu memerlukan opsi untuk memilih rentang tanggal tertentu secara manual saat melihat rekap kehadiran?	PINDAAN, UNTUK TERPINDAAN DAN KASIR DAN KEMER.
10.	Apakah Bapak/Ibu perlu mengunduh atau mencetak laporan rekap tersebut, atau cukup melihatnya langsung di sistem?	PINDAAN, UNTUK TERPINDAAN DAN KASIR DAN KEMER.
11.	Jika laporan perlu diunduh, format apa yang paling memudahkan Bapak/Ibu (misalnya Excel)?	YA, EXCEL.

No.	Pertanyaan	Jawaban Narasumber
12	Apakah Bapak/Ibu menggunakan akun login tersendiri untuk dapat melihat rekam kehadiran tersebut secara mandiri melalui sistem?	Dit. IU

Pewawancara:  (Muhammad Athar Utama) NRP: 22.304.0077	Narasumber:  (Dwi S. Pd SD) NRP: 198102220014003001
---	---

Program Studi Teknik Informatika – Universitas Paseriden Bandung | NRP 22.304.0077 | 2028

Hasil Wawancara

Lampiran 5 Hasil wawancara Kepala sekolah





Lampiran 6 pelaksanaan UAT