

RANCANG BANGUN SISTEM INFORMASI MONITORING SANTRI BERBASIS WEB PADA *ISLAMIC BOARDING SCHOOL* ASH-SHIDDIIQI MENGGUNAKAN METODE *PROTOTYPE*

Dwi Abdi Putra¹, Rizqa Raaika Bintana², Daniel Arsa³

1,2,3) Sistem Informasi, Sains dan Teknologi, Universitas Jambi, Indonesia

Article Info	ABSTRACT
Article history: Received: 24 Mei 2026 Revised: 02 Juli 2026 Accepted: 28 Juli 2026	Abstrak Penilaian kemajuan belajar santri di lingkungan pesantren seringkali dilakukan melalui penerimaan rapor, yang biasanya merupakan satu-satunya kesempatan formal dalam satu semester untuk menyampaikan informasi mengenai pencapaian akademik mereka. Hal ini tidak selalu memberikan solusi yang memadai bagi orang tua yang ingin memantau perkembangan anak secara cepat dan real-time. Penelitian ini bertujuan untuk merancang dan membangun sistem informasi monitoring santri berbasis web di Islamic Boarding School Ash-Shiddiiqi menggunakan metode <i>Prototype</i>. Sistem ini dikembangkan menggunakan framework Laravel 9, bahasa pemrograman PHP, basis data MySQL, dan framework tampilan Bootstrap. Tahapan metode <i>Prototype</i> meliputi pengumpulan kebutuhan (<i>communication</i>), perancangan cepat (<i>quick design</i>), dan pembangunan <i>prototype</i> (<i>construction</i>). Hasil dari penelitian ini adalah sebuah sistem informasi yang menyediakan fitur dashboard, manajemen data santri, pencatatan hafalan Al-Qur'an, absensi, pelanggaran, pencapaian, serta pengajuan izin yang dapat diakses oleh admin, ustadz, dan wali santri. Pengujian sistem dilakukan menggunakan metode Black Box Testing, yang menunjukkan bahwa seluruh fungsionalitas sistem berjalan dengan baik dan layak digunakan. Kata Kunci: Sistem Informasi Monitoring, Islamic Boarding School, Prototype, Laravel, Black Box Testing Abstract <i>Assessment of student learning progress in Islamic boarding school environments is often conducted through report card distribution, which is typically the only formal opportunity in one semester to convey information regarding students' academic achievements. This does not always provide an adequate solution for parents who wish to monitor their children's development quickly and in real-time. This study aims to design and develop a web-based student monitoring information system at Islamic Boarding School Ash-Shiddiiqi using the Prototype method. The system was developed using the Laravel 9 framework, PHP programming language, MySQL database, and Bootstrap front-end framework. The stages of the Prototype method include requirements gathering (communication), rapid design (quick design), and prototype construction (construction). The result of this research is an information system that provides features for dashboard, student data management, Qur'an memorization tracking, attendance, disciplinary violations, achievements, and leave requests, accessible by administrators, teachers (ustadz), and student guardians (wali santri). System testing was conducted using the Black Box Testing method, demonstrating that all system functionalities operate correctly and are fit for use.</i> Keywords: Student Monitoring Information System, Islamic Boarding School, Prototype, Laravel, Black Box Testing

Djtechno: Jurnal Teknologi Informasi oleh Universitas Dharmawangsa Artikel ini bersifat open access yang didistribusikan di bawah syarat dan ketentuan dengan Lisensi



Corresponding Author:E-mail : rizqa.raaiqa.bintana@unja.ac.id

1. PENDAHULUAN

Perkembangan Teknologi Informasi dan Komunikasi (TIK) yang pesat mendorong berbagai sektor, termasuk pendidikan, untuk mengadopsi solusi berbasis digital guna meningkatkan efisiensi dan kualitas layanan. Salah satu penerapan TIK di bidang pendidikan adalah sistem informasi monitoring berbasis web yang memungkinkan pengelolaan dan penyampaian informasi secara terstruktur dan real-time. Pemanfaatan TIK dalam pendidikan turut menopang pengelolaan pembelajaran daring pada pendidikan tinggi serta aktivitas pembelajaran jarak jauh di bidang kajian keislaman [1], [2].

Islamic Boarding School (IBS) Ash-Shiddiqi merupakan pondok pesantren yang berlokasi di Kecamatan Pemayung, Kabupaten Batanghari, Provinsi Jambi. Pesantren ini memberikan pendidikan agama Islam sekaligus pendidikan formal kepada para santri yang tinggal di dalamnya. Proses pemantauan perkembangan santri selama ini mengandalkan penerimaan rapor semesteran sebagai satu-satunya sarana penyampaian informasi formal kepada orang tua, sehingga pengawasan berjalan secara terbatas dan tidak *real-time*. Aspek yang dipantau mencakup hafalan Al-Qur'an, absensi kegiatan, pelanggaran tata tertib, pencapaian, serta pengelolaan perizinan keluar pesantren. [3]

Keterlibatan dan pengawasan orang tua dinilai penting dalam membantu meningkatkan prestasi belajar anak, terutama jika informasi perkembangan dapat diakses secara cepat dan berkala [4]. Penelitian-penelitian sebelumnya menunjukkan bahwa sistem informasi monitoring berbasis web mampu meningkatkan efektivitas komunikasi antara pihak lembaga dan orang tua.

Penelitian Mursidin et al. [5] mengembangkan sistem monitoring perkembangan anak berbasis website yang memungkinkan orang tua mengakses informasi secara efektif dan efisien. Andhika dan Samsoni [6] merancang sistem monitoring kegiatan santri dengan fitur hafalan, pelanggaran, dan prestasi untuk pondok pesantren berbasis web.

Penelitian ini juga didorong oleh perkembangan penelitian sejenis yang belum sepenuhnya menjawab kebutuhan spesifik pesantren modern. Maulana [7] menggunakan model *prototype* untuk membangun sistem monitoring berbasis website dan menekankan pentingnya interaksi antara pengembang dan pengguna dalam setiap iterasi. Fauzi et al. [12] mengembangkan *backend* aplikasi monitoring santri berbasis web dengan pendekatan *iterative incremental* yang menunjukkan efektivitas pendekatan bertahap dalam pengembangan sistem pesantren. Arisandi dan Fatmawati [13] mengembangkan aplikasi monitoring santri berbasis Android, namun belum mencakup manajemen perizinan dan sistem poin pelanggaran yang terotomatisasi. Kesenjangan-kesenjangan tersebut memperkuat urgensi penelitian ini dalam mengembangkan sistem yang lebih komprehensif dan relevan dengan konteks IBS Ash-Shiddiqi Jambi.

Secara lebih spesifik, hasil observasi dan wawancara awal di IBS Ash-Shiddiqi menunjukkan bahwa permasalahan yang dihadapi tidak terbatas pada aspek akademik semata. Pencatatan kehadiran dan setoran hafalan Al-Qur'an santri masih dilakukan secara manual pada buku catatan ustadz, sehingga rentan hilang dan sulit direkapitulasi secara cepat. Pesantren juga menerapkan sistem poin kedisiplinan yang memengaruhi nilai akhir ijazah, namun pencatatannya belum terstruktur sehingga menyulitkan perhitungan yang akurat dan transparan. Dokumentasi prestasi santri dan proses perizinan keluar pesantren pun masih dilakukan secara konvensional dan minim terdokumentasi. Kondisi ini berbeda dengan penelitian-penelitian sejenis [5], [6], [7], [12], [13] yang pada umumnya hanya berfokus pada satu atau dua aspek monitoring (misalnya hafalan atau perizinan saja) tanpa mengintegrasikan pengelolaan kehadiran, poin kedisiplinan berjenjang, pencapaian, dan perizinan dua arah dalam satu sistem yang disesuaikan dengan konteks operasional pesantren tertentu. Research Gap inilah yang mendasari kebutuhan sistem monitoring terintegrasi khusus bagi IBS Ash-Shiddiqi Jambi.

Berdasarkan latar belakang tersebut, penelitian ini bertujuan merancang dan membangun Sistem Informasi Monitoring Santri berbasis web pada IBS Ash-Shiddiqi Jambi menggunakan metode *prototype*. Sistem dikembangkan dengan enam modul utama (*dashboard*, hafalan Al-Qur'an, kehadiran santri, pelanggaran disiplin, pencapaian santri, dan izin santri) serta tiga peran pengguna (Admin, Ustadz, dan Wali

Santri). Metode *prototype* dipilih karena kebutuhan pengguna belum terdefinisi secara rinci dan memerlukan iterasi bersama pengguna sebelum sistem final dibangun [7], [8]. Pengujian dilakukan menggunakan *Black Box Testing* dengan melibatkan responden dari ketiga peran pengguna secara langsung.

2. METODE PENELITIAN

2.1 Model Pengembangan Sistem

Metode yang digunakan adalah model *prototype*, yaitu pendekatan pengembangan iteratif di mana versi awal sistem dibangun secara cepat, kemudian dievaluasi dan diperbaiki bersama pengguna hingga kebutuhan terpenuhi [7]. Model ini sesuai diterapkan ketika kebutuhan sistem belum terdefinisi dengan jelas dan memerlukan interaksi aktif antara pengembang dan pengguna [8]. Perbandingan beberapa model pengembangan sistem disajikan pada Tabel 1.

Pengumpulan data kebutuhan pengguna pada tahap Communication dilakukan melalui tiga teknik yang saling melengkapi. Pertama, observasi langsung terhadap proses pencatatan kehadiran, hafalan, pelanggaran, dan perizinan yang berjalan di IBS Ash-Shiddiqi Jambi. Kedua, wawancara terstruktur dengan kepala staf IT pesantren untuk menggali kebutuhan fungsional serta kendala pada proses monitoring yang berjalan. Ketiga, studi dokumen terhadap data riwayat santri, tata tertib pesantren, jadwal kegiatan, dan transkrip nilai yang menjadi acuan perancangan kebutuhan sistem. Ketiga teknik tersebut digunakan untuk memastikan kebutuhan pengguna terdefinisi secara menyeluruh sebelum memasuki tahap Quick Plan.

Tabel 1. Perbandingan Model Pengembangan Sistem

Metode	Kelebihan	Kekurangan	Kapan Digunakan
Waterfall	Struktur jelas dan terorganisir	Tidak fleksibel, tidak ada persetujuan klien hingga produk selesai	Proyek dengan kebutuhan tetap
Spiral	Menekankan mitigasi risiko	Membutuhkan banyak waktu perencanaan, tidak cocok untuk proyek kecil	Proyek risiko tinggi dan kompleks
Agile	Responsif terhadap perubahan kebutuhan	Kurang cocok untuk proyek besar dan kompleks	Proyek dengan kebutuhan berubah-ubah
RAD	Pengembangan cepat dan efisien	Memerlukan sumber daya manusia dan biaya yang besar	Proyek waktu singkat dengan budget tinggi
Prototype	Menemukan kebutuhan pengguna lebih awal melalui evaluasi iteratif bertahap	Kompleksitas sistem dapat meningkat seiring bertambahnya iterasi	Sistem custom dengan kebutuhan yang belum terdefinisi jelas

Model prototype dalam penelitian ini terdiri dari lima tahapan iteratif:

- (1) *Communication* — komunikasi dengan pengguna untuk mengidentifikasi kebutuhan;
- (2) *Quick Plan* — perencanaan cepat solusi teknis dan spesifikasi awal;
- (3) *Modeling Quick Design* — pemodelan UML (*Use Case Diagram*, *Activity Diagram*, *Class Diagram*) dan rancangan mockup antarmuka;
- (4) *Construction of Prototype* — penerjemahan desain ke dalam kode program dan basis data; serta
- (5) *Deployment, Delivery, and Feedback* — demonstrasi dan evaluasi bersama pengguna untuk memperoleh umpan balik perbaikan [7].

2.2 Arsitektur Sistem

Sistem mengadopsi pola *Model-View-Controller (MVC)* yang disediakan oleh *framework* Laravel [10]. Alur kerja sistem: Pengguna → Peramban → *Laravel Routes* → *Controller* → Model → *MySQL* → *View* → Peramban. Autentikasi dikelola melalui *middleware* bawaan Laravel dengan mekanisme *hashing* kata sandi. Sistem mendukung tiga peran pengguna (Admin, Ustadz, Wali Santri) masing-masing dengan hak akses terisolasi. Antarmuka dibangun menggunakan *Blade template engine* dan komponen *Bootstrap 5* agar responsif pada berbagai ukuran layar [14].

Tabel 2. Aktor dan Hak Akses Sistem

No	Aktor	Hak Akses
1	Admin (Pengurus / Pengasuh)	Login/logout, mengelola master data (kelas, kamar, sesi kehadiran, kategori pelanggaran), mengelola data santri (CRUD + import Excel + soft delete), mengelola akun pengguna (CRUD + toggle status aktif/non-aktif), menyetujui/menolak pengajuan izin, mengakses laporan dan ekspor Excel, melihat trigger surat panggilan otomatis, mengakses dashboard ringkasan global
2	Ustadz	Login/logout, mencatat setoran hafalan Al-Qur'an per santri, input absensi massal per kelas dan sesi, mencatat pelanggaran disiplin beserta kategori poin, mencatat pencapaian/prestasi santri, melihat daftar izin (read-only), mengakses dashboard rekap kelas yang diampun
3	Wali Santri	Login/logout, melihat dashboard terpersonalisasi anak yang terhubung, memantau riwayat hafalan, kehadiran, pelanggaran, dan pencapaian anak secara read-only, mengajukan izin keluar pesantren, memantau status pengajuan izin secara real-time

Sumber: Data penelitian, 2026

Isolasi akses antar peran dijamin oleh *middleware* autentikasi Laravel yang memvalidasi role setiap kali pengguna mengakses rute tertentu. Upaya akses lintas peran (misalnya Ustadz mengakses URL Admin) akan langsung diblokir dan diarahkan

kembali tanpa memberikan respons data apa pun. Mekanisme ini divalidasi dalam skenario pengujian 1 pada Tabel 5 (pengujian Ustadz).

2.3 Pengujian Sistem

Pengujian dilakukan menggunakan metode Black Box Testing yang memvalidasi spesifikasi fungsional dari perspektif pengguna tanpa memperhatikan struktur kode internal [9]. Responden pengujian terdiri dari tiga kelompok sesuai peran: Admin (11 skenario), Ustadz (5 skenario), dan Wali Santri (5 skenario), sehingga total 21 skenario uji. Persentase keberhasilan dihitung menggunakan rumus berikut:

$$\frac{\sum x}{n} \times 100\% \dots (1)$$

$$\frac{\sum y}{n} \times 100\% \dots (2)$$

Keterangan:

n : Jumlah semua fungsi yang ada

$\sum x$: Jumlah fungsi yang berjalan dengan baik

$\sum y$: Jumlah fungsi yang tidak berjalan baik

Sistem dinyatakan layak apabila nilai persentase $\sum x/n \times 100\%$ (fungsi berjalan baik) lebih besar daripada nilai $\sum y/n \times 100\%$ (fungsi tidak berjalan baik) [9].

Analisis hasil pengujian dilakukan secara deskriptif kuantitatif, yaitu dengan membandingkan nilai persentase fungsi yang berjalan baik ($\sum x/n \times 100\%$) terhadap nilai persentase fungsi yang tidak berjalan ($\sum y/n \times 100\%$) pada masing-masing kelompok responden maupun secara keseluruhan, sehingga dapat disimpulkan tingkat kelayakan sistem berdasarkan kriteria pada Persamaan (1) dan (2).

3. HASIL DAN PEMBAHASAN

3.1 Gambaran Umum Sistem

Sistem Informasi Monitoring Santri dikembangkan untuk IBS Ash-Shiddiqi Jambi sebagai sarana digital bagi pengurus pesantren dalam mencatat dan memantau perkembangan santri jenjang SMA secara terintegrasi. Sistem menyediakan enam modul utama yang menggantikan pencatatan manual yang sebelumnya tersebar dan tidak terstruktur, sehingga informasi dapat diakses kapan saja melalui antarmuka web. Gambaran spesifikasi lingkungan implementasi disajikan pada Tabel 3.

Tabel 3. Spesifikasi Lingkungan Implementasi

No	Komponen	Spesifikasi / Versi
1	Sistem Operasi	MacOS Tahoe 26.1
2	Perangkat Keras	MacBook Air M1 2020, RAM 8 GB, CPU 8-core
3	Web Server (lokal)	XAMPP (Apache + MySQL/MariaDB)
4	Bahasa Pemrograman	PHP 8.x
5	Framework Backend	Laravel 9.x (MVC Pattern)
6	Basis Data	MySQL
7	Framework Frontend	Bootstrap 5
8	Template Engine	Blade (Laravel)
9	Editor Kode	Visual Studio Code
10	Peramban Pengujian	Google Chrome (versi terbaru)

Sumber: Data penelitian, 2026

3.2 Pemodelan Sistem

Pemodelan sistem dilakukan menggunakan *Unified Modeling Language (UML)* yang mencakup *Use Case Diagram*, *Activity Diagram*, dan *Class Diagram*. *Use Case Diagram* menggambarkan interaksi tiga aktor (Admin, Ustadz, Wali Santri) dengan modul-modul sistem. Admin memiliki akses penuh pada seluruh modul, termasuk manajemen master data, pengelolaan pengguna, laporan, dan ekspor. Ustadz berperan mencatat hafalan, kehadiran, pelanggaran, dan pencapaian, sedangkan Wali Santri berperan sebagai pemantau perkembangan anak dan pengaju izin.

Activity Diagram dibuat untuk menjelaskan alur kerja pada proses utama, meliputi login dengan validasi *role-based access control (RBAC)*, pencatatan hafalan dengan kalkulasi halaman dan konversi grade otomatis, input absensi massal per sesi kegiatan, pencatatan pelanggaran dengan akumulasi poin dan *trigger* surat panggilan otomatis, serta alur pengajuan dan persetujuan izin. *Class Diagram* merekam sepuluh entitas utama (*User*, Santri, Kelas, Kamar, Hafalan, Kehadiran, Kategori Pelanggaran, Pelanggaran, Pencapaian, Izin) beserta atribut, *method*, dan relasi antar-entitas sesuai implementasi ORM Laravel.

Rancangan antarmuka menggunakan palet warna utama berbasis identitas IBS Ash-Shiddiiqi Jambi (hijau dominan dipadukan kuning keemasan) dan palet warna pendukung standar sistem (biru untuk aksi/informasi, merah untuk peringatan/batal, hitam untuk teks). *Mockup* dikerjakan menggunakan *Figma* sebelum diimplementasikan ke dalam kode.

Kebutuhan Fungsional dan Non-Fungsional

Berdasarkan hasil tahap *Communication* pada *Prototype* I, kebutuhan sistem diklasifikasikan menjadi kebutuhan fungsional (Tabel 4) dan non-fungsional (Tabel 5). Penetapan kebutuhan ini menjadi acuan validasi pada tahap pengujian.

Tabel 4. Kebutuhan Fungsional Sistem

No	Kode	Deskripsi Kebutuhan Fungsional
1	FR-01	Sistem menyediakan halaman login dengan otentikasi berbasis akun pengguna dan pengarahannya dashboard sesuai role
2	FR-02	Admin dapat menambah, mengubah, melihat, menghapus (soft delete), dan mengimpor data santri dari Excel
3	FR-03	Admin dapat mengelola master data (kelas, kamar, sesi kehadiran, kategori pelanggaran) yang mempengaruhi fitur lain
4	FR-04	Ustadz dapat mencatat setoran hafalan Al-Qur'an per santri dengan empat jenis hafalan beserta kalkulasi halaman dan konversi grade otomatis
5	FR-05	Ustadz dapat mencatat kehadiran santri secara massal (bulk input) per kelas dan sesi kegiatan dengan mekanisme idempoten
6	FR-06	Ustadz dapat mencatat pelanggaran disiplin santri beserta poin otomatis berdasarkan kategori, sistem memicu surat panggilan otomatis saat akumulasi poin mencapai batas berat
7	FR-07	Ustadz dapat mencatat pencapaian/prestasi santri beserta tingkat dan keterangan
8	FR-08	Wali Santri dapat mengajukan izin keluar pesantren (tanggal mulai, tanggal berakhir, alasan) dan memantau status secara real-time
9	FR-09	Admin dapat menyetujui atau menolak pengajuan izin santri, status diperbarui secara real-time pada akun Wali Santri
10	FR-10	Sistem menyediakan laporan dan ekspor Excel untuk data hafalan, kehadiran, dan pelanggaran dengan filter periode

Sumber: Data penelitian, 2026

Tabel 5. Kebutuhan Non-Fungsional Sistem

No	Aspek	Deskripsi Kebutuhan Non-Fungsional
1	Usability	Antarmuka dibangun dengan Bootstrap 5 agar tampilan konsisten, responsif, dan mudah dipelajari oleh seluruh peran pengguna
2	Security	Akses halaman dibatasi untuk pengguna yang telah login, isolasi data antar peran dijamin oleh middleware Laravel, kata sandi disimpan menggunakan mekanisme hashing bcrypt
3	Performance	Sistem dirancang untuk operasional harian pada skala santri jenjang SMA di satu pesantren dengan waktu respons halaman yang wajar pada basis data relasional
4	Compatibility	Sistem dapat diakses melalui peramban modern (Google Chrome, Mozilla Firefox, Microsoft Edge) pada perangkat desktop atau laptop
5	Maintainability	Kode terstruktur dalam pola MVC Laravel, migrations basis data memudahkan pengelolaan versi skema, setiap modul berada dalam direktori controller dan view yang terpisah

Sumber: Data penelitian, 2026

3.3 Penerapan Metode Prototype

Pengembangan sistem dilaksanakan dalam dua iterasi utama. *Prototype I* dibangun berdasarkan hasil *communication* dan *quick design* yang menghasilkan enam modul inti beserta struktur MVC (Model, View, Controller) sesuai pemodelan UML. *Prototype I* difokuskan pada kelengkapan fungsi daripada kesempurnaan tampilan untuk mempercepat iterasi. Setelah demonstrasi kepada pengurus pesantren, diperoleh enam poin umpan balik yang menjadi dasar perbaikan pada *Prototype II*, sebagaimana disajikan pada Tabel 6.

Tabel 6. Hasil Evaluasi Prototype I dan Tindak Lanjut Prototype II

No	Modul	Umpan Balik Pengguna (Prototype I)	Tindak Lanjut (Prototype II)
1	Hafalan	Jenis hafalan perlu diperluas menjadi 4 jenis: setoran harian, wisuda, zaidah, dan ujian	Menambahkan empat jenis hafalan dan fitur pencarian/filter berdasarkan jenis
2	Kehadiran	Sesi kehadiran harus sesuai jadwal pesantren (± 20 kegiatan/hari); akses presensi tidak boleh terbatas satu kelas saja	Fitur kelola sesi kegiatan oleh Admin, akses presensi dibuka untuk semua Ustadz tanpa batasan kelas
3	Pelanggaran	Akumulasi poin tertentu harus memunculkan keterangan surat peringatan, jenis pelanggaran belum lengkap	Menambahkan fitur akumulasi poin, trigger otomatis surat panggilan, dan kelola kategori pelanggaran
4	Izin	Belum tersedia fitur rekap izin santri secara keseluruhan	Menambahkan halaman rekap data izin santri dengan filter periode
5	Data Santri	Diperlukan fitur import data massal dari file Excel (migrasi dari sistem lama)	Fitur import data santri format Excel (.xlsx)
6	Pencapaian	Diperlukan fitur rekap data pencapaian dengan filter periode tertentu	Menambahkan halaman rekap data pencapaian santri

Sumber: Data penelitian, 2026

Seluruh umpan balik pada iterasi I diterapkan pada *Prototype II*. *Prototype II* mencakup keenam modul yang telah berfungsi penuh beserta penyempurnaan berdasarkan evaluasi. Setelah demonstrasi *Prototype II*, pengurus pesantren menyatakan sistem telah memenuhi kebutuhan dan siap untuk diimplementasikan sebagai sistem akhir.

3.4 Implementasi Sistem

Implementasi dilaksanakan menggunakan framework Laravel 9 dengan pola MVC. Basis data dibangun di *MySQL* melalui fitur *migrations* Laravel yang memudahkan pengelolaan versi struktur basis data. Sistem mendukung tiga peran pengguna dengan *dashboard* yang berbeda, yaitu admin menampilkan statistik global (total santri aktif, rekap pelanggaran bulan ini, grafik absensi, dan daftar izin terbaru) beserta fitur manajemen master data. Ustadz menampilkan rekap kegiatan harian dan

metrik kelas yang diampu. Wali Santri menampilkan ringkasan perkembangan anak yang terhubung.

Modul hafalan mendukung pencatatan setoran dengan empat jenis (setoran harian, wisuda, zaidah, ujian), kalkulasi halaman otomatis di sisi *JavaScript*, dan konversi nilai numerik ke *grade* huruf secara otomatis di sisi *backend*. Modul kehadiran mendukung input absensi massal (*bulk input*) per kelas dan sesi, memastikan tidak ada duplikasi data meskipun form disubmit ulang. Modul pelanggaran dilengkapi dengan mekanisme tarik poin dinamis berdasarkan kategori dan *trigger* otomatis pembuatan dokumen Surat Panggilan (SP) saat akumulasi poin santri mencapai batas tingkat berat. Modul izin mendukung alur pengajuan oleh Wali Santri, verifikasi oleh Admin (setujui/tolak), serta pembaruan status *real-time* yang dapat dipantau oleh Wali Santri. Seluruh data antar pengguna diisolasi di *backend* sehingga satu wali tidak dapat mengakses data anak wali lain.

3.5 Pengujian Black Box Testing

Pengujian *Black Box Testing* dilakukan oleh tiga kelompok responden sesuai peran masing-masing, mencakup total 21 skenario pengujian. Seluruh hasil pengujian menunjukkan status valid (YA). Tabel 7, 8, dan 9 menyajikan hasil lengkap pengujian untuk masing-masing peran.

Tabel 7. Hasil Pengujian Black Box Testing — Responden Admin (11 Skenario)

No	Fitur Sistem	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
1	Autentikasi Admin	(a) Input kredensial valid; (b) Input password salah; (c) Akses dashboard tanpa login	(a) Diarahkan ke Dashboard Admin; (b) Gagal login dengan notifikasi 'Kredensial tidak valid'; (c) Akses ditolak, dikembalikan ke login	[✓] (a) Berhasil masuk Dashboard; (b) Peringatan muncul; (c) Proteksi middleware berjalan sempurna
2	Dashboard Admin	Mengakses halaman muka (Dashboard) setelah login	Menampilkan metrik total santri aktif, total pelanggaran bulan ini, grafik absensi, dan daftar izin terbaru	[✓] Berhasil merender metrik dan grafik secara akurat dari basis data
3	Master Data: Kelas	Tambah, edit, dan hapus data Kelas beserta tingkatnya	Data tersimpan di database dan langsung tercermin pada dropdown kelas di fitur lainnya	[✓] Berhasil CRUD data Kelas dengan format valid
4	Master Data: Kamar	Tambah, edit, dan hapus nama Kamar dan kapasitasnya	Peringatan kapasitas terisi muncul saat ada santri di kamar tersebut sebelum dihapus	[✓] Berhasil mengelola data ruangan kamar santri
5	Master Data: Sesi Kehadiran	Mengisi rentang waktu sesi (misal: Subuh 04:30–05:30) dan menetapkan urutannya	Validasi form menolak jam tumpang tindih; data berhasil disimpan	[✓] Berhasil membuat master sesi kehadiran untuk modul absensi

6	Master Kategori Pelanggaran	Memasukkan jenis sanksi, menetapkan default poin, dan indikator (Ringan/Sedang/Berat)	Poin default terikat ke kategori dan otomatis terpanggil di sisi Ustadz	[✓] Berhasil menyimpan data parameter pelanggaran
7	Manajemen Hak Akses User	Mengubah status is_active pengguna (Wali Santri/Ustadz) menjadi non-aktif	Akun terkait dipaksa keluar (force logout) dari sistem	[✓] Berhasil mengunci akun dan mencegah login ilegal
8	Manajemen Data Santri (CRUD)	(a) Tambah data manual dengan detail Santri & Wali; (b) Soft Delete pada santri	(a) Profil santri terhubung dengan data master dan akun Wali; (b) Data hilang dari daftar aktif, riwayat historis tetap aman	[✓] (a) Relasi Foreign Key tervalidasi; (b) Soft delete tanpa merusak relasi historis
9	Kelola Izin Santri	(a) Pencarian spesifik nama anak; (b) Ubah status 'Menunggu' menjadi 'Disetujui' atau 'Ditolak'	(a) Tabel tersaring sesuai nama/tanggal; (b) Status berubah dari kuning ke hijau (Disetujui)	[✓] (a) Filter data izin berjalan lancar; (b) Otorisasi persetujuan berhasil disimpan
10	Laporan & Ekspor	Akses laporan Hafalan, Kehadiran, dan Pelanggaran, set filter bulan/tahun, tekan Export Excel	Tabel hanya memunculkan baris di rentang waktu tersebut; sistem menghasilkan file .xlsx tanpa galat	[✓] Berhasil filtering tingkat lanjut dan ekstrak laporan Excel
11	Surat Panggilan (Otomasi)	Simulasikan poin pelanggaran santri mencapai batas maksimal tingkat Berat	Sistem otomatis meng-generate dokumen Surat Panggilan (SP) yang siap diunduh Admin	[✓] Logika trigger otomatis SP sukses memicu pembuatan surat peringatan

Sumber: Data pengujian, 2026

Tabel 8. Hasil Pengujian Black Box Testing — Responden Ustadz (5 Skenario)

No	Fitur Sistem	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
1	Autentikasi & Limitasi Akses	Login sebagai Ustadz dan mencoba mengakses URL milik Admin (/admin/santri)	Dashboard Ustadz terbuka normal; upaya akses URL Admin langsung dihadap (Akses Ditolak)	[✓] Berhasil login dan gatekeeping antar peran terjamin
2	Kehadiran Massal (Bulk Input)	(a) Pilih Kelas, Tanggal, dan Sesi; (b) Tekan 'Semua Hadir', tandai 1 santri 'Sakit', lalu simpan	(a) 30 nama santri termuat dalam satu layar; (b) 30 record status kehadiran tersimpan ke database dengan satu submit	[✓] (a) Daftar absensi berhasil difilter; (b) Bulk data kehadiran diproses tanpa galat
3	Pencatatan Hafalan (Form)	Input hafalan: Al-Mulk, halaman 562-564, jenis Ziyadah, nilai 92	JS menghitung total 3 halaman; backend mengkonversi nilai 92 menjadi grade 'A'	[✓] Data hafalan, kalkulasi halaman, dan konversi grade akurat
4	Pencatatan Pelanggaran	(a) Pilih santri dan kategori 'Terlambat'; (b) Kosongkan field 'Rincian Pelanggaran' lalu simpan	(a) Poin muncul otomatis saat kategori dipilih tanpa input manual; (b) Sistem menolak penyimpanan karena field wajib kosong	[✓] (a) Tarik poin dinamis JS berjalan; (b) Validasi null/empty field berfungsi
5	Pencatatan Pencapaian (Prestasi)	Input santri pemenang lomba dengan definisi Tingkat Juara dan Keterangan Lomba	Pencapaian langsung terkreditkan di portofolio anak yang bersangkutan	[✓] Berhasil menyimpan data apresiasi/prestasi santri

Sumber: Data pengujian, 2026

Tabel 9. Hasil Pengujian Black Box Testing — Responden Wali Santri (5 Skenario)

No	Fitur Sistem	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
1	Autentikasi & Dashboard Khusus	Login menggunakan kredensial email Wali Santri	Tampilan UI menyesuaikan menu fungsional khusus Wali Santri dan menampilkan data spesifik anak	[✓] Berhasil login dan menampilkan Dashboard terpersonalisasi yang terisolasi
2	Isolasi Data Dashboard Wali	Melihat rangkuman grafik kehadiran dan riwayat pelanggaran anak	Grafik dan metrik 100% hanya memunculkan data anak yang dikaitkan dengan ID wali tersebut (isolasi data antar orang tua)	[✓] Data berhasil di filter di backend; Wali A tidak dapat melihat data anak Wali B
3	Modul Monitoring Read-Only	Mengklik tab riwayat Hafalan, Kehadiran, Pelanggaran, dan Pencapaian anak	Rekap data historis tampil tanpa tombol Edit/Hapus (Wali tidak memiliki wewenang ubah data)	[✓] Seluruh riwayat tampil sempurna dan aman dari potensi manipulasi eksternal
4	Pengajuan Izin Santri Baru	Mengisi form tanggal izin (mulai dan berakhir) beserta alasan cuti/pulang kampung	Data langsung masuk ke notifikasi Admin dengan indikator warna status 'Menunggu' (kuning)	[✓] Berhasil mengirimkan permohonan izin cuti via sistem web
5	Cek Status Evaluasi Izin	Melihat daftar historis izin setelah di-approve oleh Admin	Label status yang awalnya kuning 'Menunggu' berubah otomatis menjadi 'Disetujui' (hijau) atau 'Ditolak' (merah)	[✓] Pembaruan state berhasil ditangkap dan ditampilkan tepat waktu di beranda wali

Sumber: Data pengujian, 2026

3.6 Perhitungan Persentase Keberhasilan

Berdasarkan hasil pengujian *Black Box Testing* terhadap 21 skenario yang mencakup tiga kelompok responden (Admin: 11 skenario, Ustadz: 5 skenario, Wali Santri: 5 skenario), seluruh fungsi yang diuji berjalan sesuai spesifikasi yang ditetapkan. Rekapitulasi perhitungan persentase keberhasilan disajikan pada Tabel 10.

Tabel 10. Rekapitulasi Hasil Pengujian Black Box Testing

No	Responden	Total Skenario (n)	Fungsi Berhasil (x)	Fungsi Gagal (y)	Persentase (X)
1	Admin	11	11	0	100%
2	Ustadz	5	5	0	100%
3	Wali Santri	5	5	0	100%
	Total	21	21	0	100%

Sumber: Data pengujian, 2026

Berdasarkan Tabel 10, perhitungan menggunakan Persamaan (1) dan (2) adalah sebagai berikut:

$$\Sigma x/n \times 100\% = 21/21 \times 100\% = 100\%$$

$$\Sigma y/n \times 100\% = 0/21 \times 100\% = 0\%$$

Karena $\Sigma x/n \times 100\%$ (100%) > $\Sigma y/n \times 100\%$ (0%), maka sistem dinyatakan layak digunakan dalam lingkungan pesantren [9]. Tidak terdapat satu pun skenario yang menunjukkan kegagalan fungsi, baik dari sisi autentikasi, isolasi data antar peran, operasi CRUD, kalkulasi otomatis, maupun alur persetujuan izin.

3.7 Pembahasan

Sistem yang dibangun berhasil menjawab rumusan masalah penelitian, yaitu merancang dan membangun sistem informasi monitoring santri berbasis web menggunakan metode *prototype* pada IBS Ash-Shiddiqi Jambi. Enam modul yang dikembangkan secara langsung menggantikan sistem pencatatan manual yang sebelumnya bergantung pada laporan semesteran dan catatan tertulis yang tersebar. Pengujian yang melibatkan ketiga kelompok pengguna nyata (Admin, Ustadz, Wali Santri) memvalidasi bahwa sistem tidak hanya berfungsi secara teknis tetapi juga memenuhi kebutuhan operasional dari perspektif setiap peran.

Fitur isolasi data *backend* yang tervalidasi pada pengujian Wali Santri (skenario 2, Tabel 9) merupakan aspek keamanan kritis yang memastikan privasi data antar keluarga santri. Fitur *trigger* otomatis Surat Panggilan yang tervalidasi pada pengujian Admin (skenario 11, Tabel 7) merupakan kontribusi baru dibandingkan dengan sistem serupa yang dikembangkan sebelumnya. Fitur ini mengotomatisasi proses administratif yang sebelumnya dilakukan secara manual oleh pengasuh. Mekanisme input absensi massal (*bulk input*) yang tervalidasi pada pengujian Ustadz (skenario 2, Tabel 8) menjawab kebutuhan spesifik pesantren yang memiliki hingga 20 sesi kegiatan per hari.

Hasil evaluasi kualitatif yang diperoleh melalui wawancara dan diskusi langsung dengan pengurus IBS Ash-Shiddiqi pada setiap demonstrasi *prototype* (Tabel 6) memperkuat temuan pengujian Black Box Testing. Pengurus menyatakan bahwa fitur dalam aplikasi ini sangat membantu karena menggantikan proses manual yang sebelumnya memakan waktu, sedangkan fitur impor data santri dari Excel dinilai signifikan mempercepat migrasi dari sistem pencatatan lama. Wali santri yang menjadi responden pengujian juga menyampaikan bahwa akses *read-only* terhadap riwayat hafalan, kehadiran, dan pelanggaran anak memberikan rasa keterlibatan yang tidak diperoleh dari sistem rapor semesteran sebelumnya. Di samping kelebihan tersebut, sistem ini juga memiliki keterbatasan. Sistem masih berbasis web sepenuhnya

sehingga aksesibilitas wali santri bergantung pada ketersediaan koneksi internet dan belum didukung aplikasi mobile native (Android/iOS). Pengujian juga masih dilakukan pada skala prototype akhir dalam periode terbatas, sehingga evaluasi kinerja sistem dalam jangka panjang (misalnya saat volume data santri dan aktivitas pencatatan bertambah signifikan) belum tercakup dalam penelitian ini dan menjadi peluang bagi penelitian selanjutnya.

Tabel 11. Perbandingan Sistem dengan Penelitian Terdahulu

Penulis (Tahun)	Objek / Judul Sistem	Metode Pengembangan	Fitur Utama	Perbedaan dengan Penelitian Ini
Mursidin et al. (2024) [5]	Sistem monitoring perkembangan anak berbasis web (Rumah Pintar Indonesia)	Prototype	Monitoring perkembangan anak, akses real-time untuk orang tua	Penelitian ini menambahkan sistem poin pelanggaran berjenjang, trigger surat panggilan otomatis, manajemen perizinan dua arah, dan ekspor laporan Excel
Andhika & Samsoni (2023) [6]	Sistem monitoring kegiatan santri Pondok Pesantren As-Shofa berbasis web	Tidak disebutkan	Hafalan, pelanggaran, prestasi santri	Penelitian ini melengkapi dengan modul izin terstruktur (pengajuan-persetujuan-penolakan), notifikasi status real-time, dan import data massal dari Excel
Arisandi & Fatmawati (2023) [13]	Aplikasi monitoring santri pesantren Tahfidhul Qur'an berbasis Android	Tidak disebutkan	Info pesantren, kegiatan santri	Penelitian ini berbasis web (bukan Android), mencakup modul izin, sistem poin pelanggaran, dan trigger surat panggilan otomatis
Maulana (2025) [7]	Sistem monitoring berbasis website (Pondok Pesantren HPAIC Merapi)	Prototype	Prototype awal sistem monitoring pesantren	Penelitian ini mengimplementasikan sistem monitoring ke tahap penuh dengan enam modul, tiga peran pengguna terisolasi, serta pengujian langsung oleh responden nyata
Fauzi et al. (2025) [12]	Backend monitoring santri berbasis web PPM Al-Azhar Muncar	Iterative Incremental	Backend monitoring santri, REST API	Penelitian ini membangun sistem full-stack (frontend + backend) dengan antarmuka pengguna lengkap dan pengujian Black Box Testing oleh pengguna akhir

Sumber: Kajian literatur, 2026

Dibandingkan dengan penelitian Mursidin et al. [5] yang membangun sistem monitoring perkembangan anak berbasis website, penelitian ini memperluas cakupan dengan penambahan sistem poin pelanggaran berjenjang, trigger surat panggilan otomatis, dan manajemen perizinan dua arah. Dibandingkan dengan Andhika dan Samsoni [6] yang mengembangkan sistem dengan fitur hafalan, pelanggaran, dan prestasi, penelitian ini melengkapinya dengan modul izin terstruktur beserta alur pengajuan, persetujuan/penolakan, dan notifikasi status *real-time* kepada orang tua. Selain itu, fitur *import* data massal dari Excel memfasilitasi migrasi dari sistem lama yang merupakan kebutuhan praktis yang tidak ditemukan pada penelitian-penelitian sejenis.

4. SIMPULAN

Sistem Informasi Monitoring Santri berbasis web pada IBS Ash-Shiddiqi Jambi berhasil dirancang dan dibangun menggunakan *framework* Laravel 9 dan metode *prototype* melalui dua iterasi pengembangan. Sistem menyediakan enam modul utama (*dashboard*, hafalan Al-Qur'an, kehadiran santri, pelanggaran disiplin, pencapaian santri, dan izin santri) dengan tiga peran pengguna (Admin, Ustadz, Wali Santri) yang terisolasi secara akses.

Pengujian *Black Box Testing* yang dilakukan oleh tiga kelompok responden mencakup 21 skenario uji terhadap seluruh modul fungsional. Seluruh fungsi yang diuji berjalan sesuai spesifikasi dengan persentase keberhasilan $\Sigma x/n \times 100\% = 21/21 \times 100\% = 100\%$ dan $\Sigma y/n \times 100\% = 0/21 \times 100\% = 0\%$, sehingga sistem dinyatakan layak digunakan. Sistem ini diharapkan meningkatkan transparansi, efisiensi pengelolaan data, dan kualitas komunikasi antara pihak pesantren dan orang tua santri, menggantikan ketergantungan pada laporan semesteran sebagai satu-satunya sarana informasi formal.

REFERENCES

- [1] I. Makruf, A. A. Rifa'i, and Y. Triana, "Moodle-based online learning management in higher education," *Int. J. Instruction*, vol. 15, no. 1, pp. 135–152, 2022, doi: 10.29333/iji.2022.1518a.
- [2] A. Yudiawan, B. Sunarso, and S. Suharmoko, "Exploring interaction patterns in distance learning of Islamic studies: A case in West Papua," *Int. J. Instruction*, vol. 14, no. 4, pp. 243–258, 2021.
- [3] D. Arwen, "Pentingnya peran orang tua dalam meningkatkan prestasi belajar siswa," *J. Education and Instruction (JOEAI)*, vol. 4, no. 2, pp. 564–576, 2021, doi: 10.31539/joeai.v4i2.3084.
- [4] E. M. Sipayung and W. S. Susilo, "Analisis usability portal akademik berbasis web menggunakan USE questionnaire," *J. Telematika*, vol. 16, no. 2, pp. 91–95, 2021.
- [5] Mursidin, Sukirman, H. Tehuayo, and I. Ali, "Rancang bangun sistem monitoring aktivitas santri berbasis web pada Pesantren Rahmatan Lil'alam," *JTEK: J. Teknik Informatika, Fakultas Teknik UIM*, 2024.
- [6] A. Andhika and Samsoni, "Perancangan sistem informasi monitoring kegiatan santri di Pondok Pesantren Technopreneur As-Shofa berbasis web," *Biner: J. Ilmu Komputer, Teknik dan Multimedia*, vol. 1, no. 4, pp. 947–956, 2023.
- [7] M. N. Maulana, "Implementasi metode *prototype* dalam pengembangan sistem monitoring berbasis website di Pondok Pesantren HPAIC Merapi," *J. Sosial dan Sains (SOSAINS)*, vol. 5, no. 7, pp. 1970–1986, 2025.
- [8] T. Pricillia and Zulfachmi, "Perbandingan metode pengembangan perangkat lunak (*waterfall*, *prototype*, *RAD*)," *J. Bangkit Indonesia*, vol. 10, no. 1, pp. 6–12, 2021, doi: 10.52771/bangkitindonesia.v10i1.153.
- [9] Uminingsih, M. Nur Ichsanudin, M. Yusuf, and S. Suraya, "Pengujian fungsional perangkat lunak sistem informasi perpustakaan dengan metode *black box testing* bagi pemula," *STORAGE: J. Ilmiah Teknik dan Ilmu Komputer*, vol. 1, no. 2, pp. 1–8, 2022, doi: 10.55123/storage.v1i2.270.
- [10] T. Otwell, *Laravel 9.x Documentation*. Laravel LLC, 2022. [Online]. Available: <https://laravel.com/docs/9.x>
- [11] Z. Subecz, "Web-development with Laravel framework," *Gradus*, vol. 8, no. 1, pp. 211–218, 2021, doi: 10.47833/2021.1.csc.006.

-
- [12] D. R. Fauzi et al., "Pengembangan backend aplikasi monitoring santri berbasis web PPM Al-Azhar Muncar dengan metode iterative incremental," JIKOM: J. Informatika dan Komputer, vol. 15, no. 2, pp. 48–61, 2025.
 - [13] A. Arisandi and K. Fatmawati, "Rancangan aplikasi monitoring santri pesantren tahfidhul Qur'an berbasis Android," STAINS: Seminar Nasional Teknologi dan Sains, vol. 2, no. 1, pp. 273–280, 2023.
 - [14] Bootstrap Team, Bootstrap 5.3 Documentation. 2023. [Online]. Available: <https://getbootstrap.com/docs/5.3/>
 - [15] Oracle Corporation, MySQL 8.0 Reference Manual. Oracle Corp., 2023. [Online]. Available: <https://dev.mysql.com/doc/refman/8.0/en/>
 - [16] The PHP Group, PHP Manual (PHP 8.3). 2024. [Online]. Available: <https://www.php.net/manual/en/>
 - [17] Microsoft, Visual Studio Code Documentation. Microsoft Corp., 2024. [Online]. Available: <https://code.visualstudio.com/docs>
 - [18] Apache Friends, XAMPP Documentation. 2024. [Online]. Available: <https://www.apachefriends.org/docs/>
 - [19] Sukirman, F. El Fazza, U. Pangerang, and Salmia, "Rancang bangun aplikasi absensi siswa berbasis web pada SMPN 1 Maros," JTEK: J. Teknologi dan Komputer, vol. 2, no. 01, pp. 118–124, 2022, doi: 10.56923/jtek.v2i01.62.
 - [20] I. P. Sari et al., "Perancangan sistem absensi pegawai kantoran secara online pada website berbasis HTML dan CSS," Blend Sains J. Teknik, vol. 1, no. 1, pp. 8–15, 2022, doi: 10.56211/blendsains.v1i1.66.
 - [21] Hasriadi, "Model pembelajaran jarak jauh pendidikan agama Islam berbasis teknologi informasi dan komunikasi," J. Konsepsi, vol. 11, no. 1, pp. 1–10, 2022.
 - [22] R. Yuliansyah, "Unit testing pada aplikasi web online," CHIPSET: J. Ilmu Komputer, Teknik, dan Multimedia, vol. 2, no. 3, 2024.