



Sistem Informasi Pemantauan Penyakit Menular di Lingkungan Sekolah

Rifan Andriansyah¹, Djalaludin Machfud², Ghema Nusa Persada³, Bintang Septiansyah⁴

¹⁻⁴ Universitas Pamulang, Tangerang Selatan, Banten, Indonesia

Email: ¹Rifan4042@gmail.com, ³dosen02682@unpam.ac.id

Abstrak—Penelitian ini bertujuan merancang dan mengembangkan Sistem Informasi Pemantauan Penyakit Menular di Lingkungan Sekolah guna mengatasi proses pemantauan kesehatan siswa yang masih dilakukan secara manual, melalui pencatatan di buku atau laporan guru UKS, dan kurang terintegrasi. Proses manual ini menimbulkan kendala seperti sulitnya memperoleh data secara cepat, risiko kehilangan data, serta keterlambatan informasi yang berpotensi menyebarkan penyakit luas. Sistem yang diusulkan dirancang berbasis Web dan Mobile (*cross-platform*) dengan menggunakan arsitektur *client-server*. Komunikasi antar komponen sistem dilakukan melalui RESTful API dengan keamanan data berlapis, sementara penyimpanan data menggunakan database MySQL/PostgreSQL. Metodologi penelitian yang digunakan meliputi empat cara Pengumpulan Data utama: Observasi Lapangan untuk memahami alur kerja manual, Wawancara Mendalam dengan staf dan orang tua, Studi Dokumentasi untuk data sekunder, serta Survei dan Kuesioner untuk data primer. Setelah analisis kebutuhan, dilakukan Perancangan Sistem yang dimodelkan menggunakan Diagram UML, termasuk *Use Case Diagram* dan *Class Diagram*, untuk memvisualisasikan struktur dan dinamika interaksi aktor utama (Siswa, Petugas Kesehatan, Admin Sekolah, dan Dinas Kesehatan) dengan sistem. Fitur utama sistem mencakup kemampuan untuk mencatat dan menyimpan data kasus penyakit secara akurat, fitur pelaporan dan *monitoring* perkembangan penyakit secara real-time, serta notifikasi peringatan dini kepada pihak terkait ketika ditemukan peningkatan kasus. Sistem ini juga memiliki antarmuka pengguna yang dirancang sederhana, intuitif, dan responsif. Implementasi dilakukan dengan pengembangan kode program, dan Evaluasi sistem melibatkan serangkaian pengujian, termasuk *User Acceptance Testing* (UAT), untuk memastikan sistem bekerja dengan baik dan memenuhi kebutuhan pengguna. Hasilnya, sistem ini mampu menyediakan informasi kesehatan siswa secara cepat, akurat, dan dapat diandalkan. Sistem ini secara efektif mendukung percepatan respons dan penerapan tindakan pencegahan, serta mampu meningkatkan kecepatan dan akurasi pelaporan kasus. Dengan demikian, sistem ini efektif dalam upaya deteksi dini serta pengendalian penyakit menular, memberikan dasar yang kuat bagi pengambilan keputusan berbasis data guna pengendalian penyakit di lingkungan pendidikan.

Kata Kunci: Sistem Informasi, Penyakit Menular, Sekolah, Pemantauan Kesehatan, Real-time.

Abstract—This study aims to design and develop an Infectious Disease Monitoring Information System in the school environment to address student health monitoring processes that are still conducted manually through record-keeping in logbooks or reports from school health unit (UKS) teachers and are not yet well integrated. These manual processes create obstacles such as difficulties in obtaining data quickly, risks of data loss, and delays in information that may potentially contribute to the widespread transmission of diseases. The proposed system is designed as a web- and mobile-based (*cross-platform*) application using a *client-server* architecture. Communication between system components is carried out through a RESTful API with multi-layer data security, while data storage utilizes a MySQL/PostgreSQL database. The research methodology involves four main data collection methods: field observation to understand manual workflows, in-depth interviews with staff and parents, documentation studies for secondary data, and surveys and questionnaires for primary data. Following requirements analysis, system design is conducted using UML diagrams, including use case diagrams and class diagrams, to visualize the structure and interaction dynamics of the main actors (students, health officers, school administrators, and the health authority) with the system. The main features of the system include the ability to accurately record and store disease case data, reporting and real-time monitoring of disease progression, and early warning notifications to relevant parties when an increase in cases is detected. The system also features a user interface designed to be simple, intuitive, and responsive. Implementation is carried out through software development, and system evaluation involves a series of tests, including User Acceptance Testing (UAT), to ensure the system functions properly and meets user needs. The results indicate that the system is able to provide student health information quickly, accurately, and reliably. It effectively supports faster response and the implementation of preventive measures, while improving the speed and accuracy of case reporting. Therefore, this system is effective in early detection and control of infectious diseases, providing a strong foundation for data-driven decision-making in disease control within educational environments.

Keywords: Information System, Infectious Diseases, School, Health Monitoring, Real-time.



1. PENDAHULUAN

Kesehatan siswa adalah komponen yang sangat penting untuk keberhasilan pembelajaran di sekolah. Siswa yang dalam kondisi fisik prima cenderung memiliki konsentrasi, semangat, dan produktivitas belajar yang lebih baik, dibandingkan dengan siswa yang sering mengalami gangguan kesehatan. Meskipun demikian, lingkungan sekolah memiliki banyak interaksi sosial dan kepadatan siswa, guru, dan karyawan, ini membuat sekolah sangat rentan terhadap penyebaran berbagai penyakit menular. Di lingkungan sekolah, penyakit musiman atau endemis seperti flu, Demam Berdarah Dengue (DBD), Infeksi Saluran Pernapasan Akut (ISPA), dan penyakit kulit sering muncul berulang kali dan menjadi masalah kesehatan di lingkungan sekolah.

Tidak adanya sistem pemantauan kesehatan yang terintegrasi dan terorganisir di sekolah adalah masalah utama yang sering terjadi. Pemantauan kesehatan siswa saat ini biasanya dilakukan secara manual, melalui pencatatan di buku kesehatan, laporan insidental dari guru UKS, atau pemberitahuan langsung dari orang tua siswa. Cara manual ini memiliki banyak keterbatasan penting. Beberapa di antaranya adalah kesulitan mendapatkan data secara cepat, risiko kehilangan atau ketidak teraturan, dan kurangnya integrasi data antara sekolah, puskesmas, dan dinas kesehatan. Akibat dari keterbatasan dan keterlambatan dalam mendapatkan informasi tentang kondisi kesehatan siswa ini, ada kemungkinan penyakit akan menyebar ke siswa lain, meningkatkan ketidakhadiran sekolah, penurunan kualitas pembelajaran, dan bahkan membuat orang tua khawatir. Identifikasi, pencegahan, dan penanganan penyakit menjadi kurang efektif jika tidak ada informasi yang cukup.

Oleh karena itu, untuk mengelola masalah ini dengan lebih efisien, diperlukan sistem berbasis komputer yang mampu memanfaatkan inovasi teknologi informasi. Diharapkan sistem informasi ini dapat mencatat, menyimpan, mengolah, dan menampilkan informasi kesehatan siswa secara terintegrasi. Sekolah dapat dengan mudah melaporkan kasus penyakit, dan puskesmas atau dinas kesehatan dapat melihat perkembangan kesehatan di sekolah secara langsung dan mengambil tindakan pencegahan. Tujuan penelitian ini adalah untuk menyelesaikan masalah di atas dengan berkonsentrasi pada desain dan pengembangan solusi digital, yang dirumuskan dengan tujuan berikut:

- Untuk memantau penyebaran penyakit menular di sekolah, sistem informasi yang handal dan terintegrasi harus dibangun untuk mencatat, menyimpan, dan menampilkan informasi kesehatan siswa secara terstruktur dan *real-time*.
- Membantu mengumpulkan, mengelola, dan menganalisis data tentang insiden penyakit menular pada siswa dan guru secara sistematis
- Mendukung respons dan tindakan pencegahan yang efektif oleh sekolah, petugas UKS, dan lembaga kesehatan terkait, termasuk kemampuan untuk mengidentifikasi penyebaran penyakit menular.
- Menyajikan laporan dan data statistik penyakit secara cepat dan akurat untuk mendukung proses pengambilan keputusan berbasis data untuk mengendalikan dan mencegah penyakit menular di lingkungan sekolah.

2. METODE PENELITIAN

2.1 Metodologi Penelitian

Metodologi penelitian yang digunakan dalam proyek pengembangan Sistem Informasi Pemantauan Penyakit Menular di Lingkungan Sekolah ini bertujuan untuk memastikan proses pengembangan sistem dilakukan secara sistematis, terukur, dan berbasis pengguna. Metode yang digunakan adalah model yang berorientasi pada siklus hidup pengembangan sistem (*System Development Life Cycle - SDLC*), atau siklus hidup pengembangan sistem, yang dimulai dengan analisis kebutuhan yang mendalam dan berakhir pada tahap implementasi dan evaluasi. Ini dilakukan untuk memastikan bahwa solusi yang dibuat tidak hanya dapat digunakan secara teknis tetapi juga relevan dan menyenangkan bagi pengguna akhir.



2.2 Pengumpulan Data

Tahap pengumpulan data merupakan langkah krusial untuk mendapatkan gambaran mendalam tentang kondisi saat ini, menemukan masalah dengan proses manual saat ini, dan menentukan kebutuhan sistem yang akan dikembangkan, tahap pengumpulan data sangat penting. Empat metode utama digunakan untuk mengumpulkan data:

- Observasi Lapangan (Observation): Pengamatan langsung akan dilakukan oleh peneliti tentang bagaimana kasus penyakit menular dilaporkan dan ditangani di lingkungan sekolah, khususnya di unit kesehatan sekolah (UKS). Tujuan observasi ini adalah untuk mendapatkan pemahaman tentang alur kerja manual yang ada, menemukan titik hambatan (*bottlenecks*), dan menemukan masalah praktis yang dihadapi oleh petugas UKS atau tenaga pendidik saat mencatat dan menanggulangi kasus penyakit menular.
- Wawancara Mendalam (In-Depth Interview): Wawancara terstruktur dan semi-terstruktur dengan berbagai stakeholder penting. Tenaga pendidik, petugas kesehatan sekolah (UKS), perwakilan siswa, orang tua, dan, jika memungkinkan, perwakilan dari Puskesmas atau Dinas Kesehatan setempat juga diwawancarai. Tujuan dari wawancara ini adalah untuk mengetahui ekspektasi mereka, pengalaman pribadi mereka dengan penyakit menular di sekolah, dan apa yang mereka anggap penting untuk sistem untuk menyelesaikan masalah mereka.
- Studi Dokumentasi (Documentation Study): Pengumpulan data sekunder melalui peninjauan dokumen resmi yang relevan, seperti laporan kesehatan tahunan sekolah, rekapitulasi epidemiologi data penyakit, dan literatur ilmiah dan peraturan perundang-undangan. (seperti Undang- Undang Republik Indonesia No. 36 Tahun 2009 tentang Kesehatan) yang berkaitan dengan kesehatan sekolah dan sistem informasi kesehatan.
- Survei dan Kuesioner (Survey and Questionnaire): Untuk mengumpulkan data primer kuantitatif dalam skala yang lebih besar, penyebaran kuesioner akan digunakan. Sebelum proses perancangan sistem dimulai, survei ini dimaksudkan untuk mengumpulkan data tentang frekuensi dan tingkat keparahan penyakit menular yang paling umum, serta umpan balik calon pengguna tentang kebutuhan dan preferensi fitur sistem. Data ini sangat penting sebagai validasi awal.

2.3 Pengembangan Sistem

Tahapan pengembangan sistem informasi ini, untuk memastikan bahwa sistem yang dibuat memenuhi kebutuhan fungsional dengan kinerja optimal, skalabilitas, dan arsitektur yang kokoh, tahapan pengembangan sistem informasi ini menggunakan pendekatan iteratif dan terstruktur. Pendekatan ini merupakan bagian penting dari Siklus Hidup Pengembangan Sistem (System Development Life Cycle - SDLC). Metode terorganisir ini menyediakan kerangka kerja yang sistematis untuk mengelola kompleksitas proyek, mulai dari konsep hingga implementasi dan perawatan. Semua langkah dilakukan secara berurutan melalui *checkpoint* dan *validasi* yang ketat. Ini membantu mengurangi risiko sejak dini dan memastikan bahwa produk akhir sesuai dengan harapan *stakeholder*.

- Analisis Kebutuhan (*Requirement Analysis*): Fase ini berfungsi untuk menentukan tujuan dan jangkauan proyek. Tujuan utamanya adalah untuk menyelaraskan keinginan pengguna dan fungsi sistem secara teknis. Daftar terperinci dari kebutuhan fungsional dan non-fungsional dibuat dari data awal yang dikumpulkan melalui *observasi*, wawancara, dan *survei*.

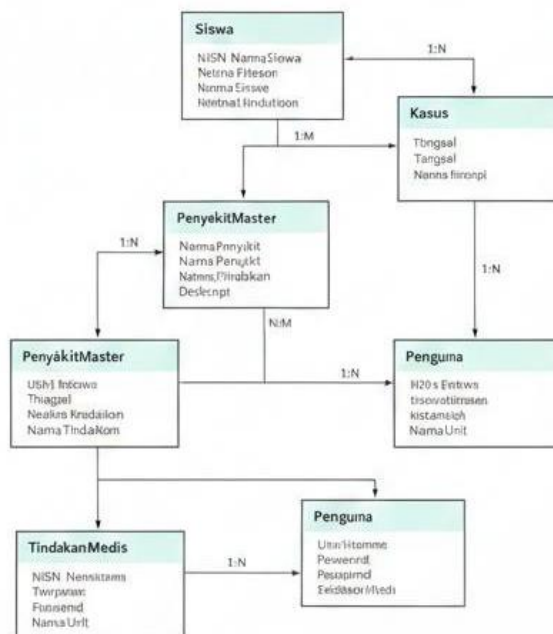
Kebutuhan fungsional yang sangat spesifik termasuk kemampuan sistem untuk mencatat insiden penyakit per siswa setiap hari dan memungkinkan sistem untuk memantau kasus dalam waktu nyata melalui *dashboard* interaktif yang menampilkan data visual. Selain itu, diperlukan modul pelaporan data statistik yang dapat menghasilkan laporan secara teratur dan *sporadis*. Selain itu, diperlukan mekanisme otomatis untuk memberi tahu pihak terkait (kepala sekolah, lembaga kesehatan) tentang peringatan dini jika lonjakan kasus melebihi ambang *epidemiologi* yang telah ditetapkan. Selain itu, sistem harus dapat melacak riwayat kesehatan pribadi setiap siswa, yang mencakup hasil pemeriksaan dan tindakan pencegahan yang telah dilakukan.

Secara tegas didefinisikan, kebutuhan non-fungsional termasuk keamanan data (otentikasi

dan otorisasi berbasis peran untuk membatasi akses ke data sensitif), kemampuan sistem untuk menangani banyak pengguna bersama dengan waktu *respons* yang cepat, dan kemudahan penggunaan (kemudahan penggunaan) antarmuka. Skalabilitas juga dipertimbangkan untuk memenuhi perkembangan pengguna dan data di masa depan. Dokumen Spesifikasi Kebutuhan Pengguna (SKP), yang disahkan oleh *stakeholder* dan berfungsi sebagai acuan resmi selama seluruh proses pengembangan, merupakan hasil akhir dari tahap ini.

- b. Perancangan Sistem (*System Design*): Pada tahap ini, rancangan konseptual diubah menjadi cetak biru teknis yang rinci. Untuk meminimalkan redundansi dan menjamin integritas data, perancangan dimulai dengan membuat Model Basis Data yang kuat dan terstruktur. *Entity Relationship Diagram (ERD)* menunjukkan entitas utama (seperti siswa, penyakit, kasus, petugas kesehatan, dan notifikasi) dan relasi kardinalitas antar entitas. Arsitektur sistem dibangun berdasarkan model tiga tingkat atau *client-server* yang terpisah. *Frontend* (presentasi), yang berfungsi sebagai presentasi, sepenuhnya terpisah dari backend, yang berfungsi sebagai logika bisnis.

Diagram UML digunakan untuk memodelkan perilaku sistem secara menyeluruh. *Use Case Diagram* menggambarkan semua fungsi yang dapat diakses oleh aktor; Diagram Kelas menunjukkan struktur kode program dan fitur data; dan *Sequence Diagram* secara khusus menunjukkan alur proses penting, seperti bagaimana petugas kesehatan memasukkan data kasus baru, divalidasi oleh sistem, dan disimpan ke database. Setelah struktur logis selesai, desain *wireframe* dan *mockup UI/UX* dibuat untuk memastikan alur kerja pengguna logis dan mengurangi tingkat kesalahan interaksi. Ini adalah contoh gambar diagram UMLnya.



Gambar 1. Diagram Hubungan Entitas (ERD)

- c. Implementasi (*Implementation*): Dalam fase implementasi, kode program harus ditulis secara aktual sesuai dengan spesifikasi desain dan arsitektur yang telah dibuat. Untuk melindungi data operasional, pengembangan ini dilakukan dalam lingkungan pengembangan terpisah, atau *staging environment*. Digunakan *framework kontemporer* yang mendukung *cross-platform* seperti *Flutter* atau *React* untuk membuat antarmuka pengguna (klien) yang dapat diakses di berbagai perangkat, seperti Web dan Mobile. Ini memastikan tampilan yang konsisten dan kinerja yang cepat. Logika bisnis dan *API backend* dibuat dengan menggunakan bahasa pemrograman *enterprise-level* seperti *Java*, *Python*, atau *PHP*. *API RESTful* berfungsi sebagai jembatan aman untuk pertukaran data

antara klien dan server. Sistem manajemen basis data relasional yang dipilih, *MySQL* atau *PostgreSQL*, menyimpan semua data transaksi dan master; keduanya dikenal karena keandalan dan kemampuan *query* yang cepat. dan kapasitas penyimpanan data yang tinggi. Fase ini memprioritaskan aspek keamanan, seperti penggunaan protokol enkripsi HTTPS untuk transmisi data dan skema otentikasi berbasis *JSON Web Token (JWT)* untuk melindungi akses data sensitif pengguna.

- d. Pengujian Sistem (*System Testing*): Proses validasi terakhir untuk memastikan kualitas perangkat lunak adalah pengujian. Pengujian dilakukan dengan langkah demi langkah dan secara menyeluruh. Pengembang melakukan pengujian unit untuk memastikan bahwa fungsi-fungsi unit tertentu (unit) akurat, ini dapat mencakup validasi format tanggal atau kode penyakit. Selain itu, pengujian integrasi dilakukan untuk memastikan bahwa modul-modul yang berbeda, seperti modul input kasus yang dapat memicu modul notifikasi, bekerja sama dan terintegrasi dengan database dengan lancar. Akhir dari pengujian, *User Acceptance Testing (UAT)*, adalah ketika pengguna akhir yang sebenarnya, petugas UKS dan admin sekolah menggunakan sistem dalam situasi operasi nyata di sekolah. Tujuan UAT adalah untuk memastikan bahwa sistem memenuhi semua kebutuhan SKP, mudah digunakan, dan benar-benar menyelesaikan masalah pemantauan penyakit yang ditemukan pada awalnya. Pengujian ini akan menghasilkan laporan bug dan umpan balik untuk proses *debugging* dan iterasi akhir sistem sebelum dikirimkan ke lingkungan produksi untuk diterapkan.

Berikut gambar mockup UX/UI Halaman Dashboard:



Gambar 2. Mockup UX/UI Halaman Dashboard

2.4 Batasan dan Spesifikasi Sistem

Proyek pengembangan sistem informasi ini memiliki Batasan Masalah yang jelas untuk menjamin fokus proyek yang jelas, memastikan alokasi sumber daya yang efisien, dan mengurangi risiko *scope creep*.

Batasan masalah sebagian besar berpusat pada pengawasan epidemiologis dan manajemen penyakit menular yang sering terjadi di lingkungan sekolah dan memiliki tingkat insiden yang signifikan. Influenza, Demam Berdarah Dengue (DBD), Infeksi Saluran Pernapasan Akut (ISPA), dan penyakit kulit yang mudah menyebar melalui kontak fisik atau lingkungan adalah beberapa jenis penyakit yang secara khusus dicakup dalam sistem ini. Pencatatan data kasus harian, penyimpanan data historis yang aman dan terpusat, pelaporan statistik dan analisis tren penyebaran, dan visualisasi data yang mudah dipahami melalui dashboard interaktif adalah semua fungsi inti sistem. Sistem ini tidak dapat melakukan tugas klinis seperti diagnosis medis, rekomendasi pengobatan, atau rekam medis elektronik (RME) yang rinci untuk setiap pasien. Ini dilakukan untuk memastikan sistem berfungsi dengan baik



sebagai alat pemantauan kesehatan masyarakat sekolah daripada berfungsi sebagai pengganti layanan medis profesional atau SIMRS yang rumit dengan regulasi ketat. Data yang dikumpulkan terbatas pada identitas siswa, jenis penyakit, tanggal mulai dan berakhir penyakit, dan status penanganan awal.

Adapun Spesifikasi teknis sistem ini dibuat sebagai pedoman teknis yang tepat untuk fase implementasi, yang memastikan sistem dapat beroperasi sesuai harapan dan interoperabilitas tinggi:

- a. Platform dan Aksesibilitas: Sistem ini harus dikembangkan sebagai aplikasi *client-server* yang menawarkan aksesibilitas ganda melalui web dan mobile. Keputusan ini dibuat untuk mendukung mobilitas tinggi karena aplikasi web ideal untuk petugas administrasi dan dinas kesehatan yang membutuhkan visualisasi data statistik yang luas, sementara aplikasi mobile sangat penting bagi petugas kesehatan sekolah di lapangan yang perlu memasukkan data kasus secara *real-time* dan cepat.
- b. Arsitektur Sistem: Arsitektur Tiga Tingkat (*Three-Tier Architecture*), akan digunakan untuk sistem. Arsitektur ini secara eksplisit membedakan lapisan presentasi (*frontend*), lapisan logika bisnis (*backend*), dan lapisan data (*database*). Pemisahan ini sangat penting untuk skalabilitas sistem karena mempermudah pemeliharaan dan memungkinkan pengoperasian yang independen pada setiap lapisan, yang meningkatkan kinerja sistem secara keseluruhan.
- c. Bahasa Pemrograman *Backend*: Bahasa pemrograman *server-side* yang teruji akan digunakan untuk mengelola logika bisnis yang kompleks, memproses data, dan menangani otentikasi dan otorisasi. *Java*, *Python*, atau *PHP* dengan (*framework* modern seperti *Spring Boot*, *Django*, atau *Laravel*), adalah pilihan yang tersedia. Bahasa-bahasa ini dipilih karena memiliki banyak library, dukungan komunitas yang luas, dan kemampuan yang luar biasa untuk membuat *RESTful API* yang aman dan efektif untuk pertukaran data.
- d. Bahasa Pemrograman *Frontend*: Fokus pengembangan antarmuka pengguna adalah untuk menghasilkan pengalaman pengguna yang mudah dipahami dan responsif. Akan digunakan *framework* canggih seperti *Flutter* atau *React*. *Flutter* disarankan karena kecepatan dan konsistensi *cross-platform mobile-first*, sedangkan *react* disarankan untuk aplikasi web yang sangat dinamis. Kedua opsi ini memastikan antarmuka pengguna tampilan antarmuka (UI) kontemporer dan *user experience* (UX) yang luar biasa di berbagai perangkat keras.
- e. Database Management System (DBMS): Sistem Manajemen Basis Data *Relasional* (RDBMS) yang kuat, seperti *MySQL* atau *PostgreSQL*, digunakan untuk menyimpan data kasus historis yang penting dan sensitif. Kedua DBMS ini menawarkan kepercayaan transaksi, integritas data melalui pembatasan relasional, kecepatan eksekusi query kompleks, dan ketahanan data yang tinggi, yang sangat penting untuk kebutuhan pelaporan epidemiologis jangka panjang.
- f. Protokol Keamanan dan Otentikasi: Keamanan sistem adalah aspek non-fungsional yang paling penting. Protokol enkripsi HTTPS (*Secure Hypertext Transfer Protocol*) harus digunakan untuk setiap komunikasi data antara *client* dan *server*. Ini dilakukan untuk menghindari penyadapan data selama pengiriman. Selain itu, sistem akan menggunakan mekanisme otentikasi dan otorisasi berbasis *JSON Web Token* (JWT). Mekanisme ini memberikan kontrol akses yang tidak statis, aman, dan efektif yang memastikan hanya pengguna yang diidentifikasi dengan peran tertentu, seperti admin, petugas UKS, atau lembaga kesehatan, yang dapat mengakses atau memanipulasi data tertentu.

3. ANALISA DAN PEMBAHASAN

3.1 Analisis Sistem dan Kebutuhan Fungsional

Analisis sistem adalah langkah diagnostik penting yang dilakukan untuk memahami kondisi operasional saat ini. Tujuan analisis ini adalah untuk menemukan dan menghubungkan kelemahan utama dalam metode pemantauan kesehatan siswa yang masih manual dan

tradisional. Solusi digital yang terorganisir, efisien, efektif, dan sepenuhnya berfokus pada pengguna kemudian menjadi celah penting yang harus dipenuhi dan diatasi.

3.1.1 Kelemahan dan Keterbatasan Sistem Manual yang Diidentifikasi:

1. Keterlambatan dan Ketiadaan *Real-time Reporting*: Pelaporan kasus penyakit sangat bergantung pada catatan fisik di buku catatan UKS, yang seringkali baru direkapitulasi secara berkala, seperti mingguan atau bulanan. Jika data ini tidak disebarluaskan dengan cepat, terutama penyakit menular, manajemen sekolah dan lembaga kesehatan tidak dapat mengantisipasi wabah secara dini.
2. Rendahnya Akurasi, Integritas, dan Kehilangan Data: Salah interpretasi tulisan, duplikat entri, dan kesalahan manusia dapat mengganggu data yang dicatat secara manual. Selain itu, catatan fisik sangat rentan terhadap kerusakan, kehilangan, atau tidak terarsip dengan baik. Hal ini berdampak negatif pada kualitas analisis *epidemiologis* yang dilakukan di masa lalu dan perencanaan pencegahan di masa depan.
3. Fragmentasi Informasi dan Kurangnya Integrasi: Tidak adanya pusat informasi kesehatan menyebabkan data tersebar dan terfragmentasi. Orang tua, Puskesmas, Sekolah, dan Dinas Kesehatan berbagi data. Misalnya, intervensi yang terkoordinasi terhambat karena Dinas Kesehatan tidak memiliki gambaran keseluruhan tren penyebaran penyakit di semua sekolah di daerah tersebut.
4. Keterbatasan Analisis dan Visualisasi Data: Sistem manual hanya dapat menghasilkan data mentah secara pasif, dan sekolah tidak dapat analisis tren, prediksi wabah kecil yang mungkin, atau visualisasi data seperti peta panas penyebaran adalah semua komponen yang sangat penting untuk mendukung pengambilan keputusan yang cepat dan berbasis bukti (*data-driven decision making*).

3.1.2 Usulan Solusi Digital dan Kebutuhan Fungsional (Fungsional Requirements):

Berdasarkan identifikasi kelemahan utama ditemukan di atas, sistem informasi baru harus dibuat untuk memenuhi beberapa kebutuhan fungsional yang kuat dan khusus:

- Modul Pencatatan dan Penyimpanan Data Kasus Penyakit yang Akurat dan Terstruktur: Untuk meminimalkan error input, sistem harus menyediakan formulir input digital yang mudah digunakan dan memiliki fitur validasi data. Identitas siswa, jenis penyakit jika memungkinkan), tanggal awal, tanggal dilaporkan, status penanganan awal, dan detail kontak darurat adalah semua informasi yang harus dicatat. Data ini harus disimpan secara terpusat dalam database relasional, yang akan membuat basis data kasus historis yang dapat diandalkan dan berfungsi sebagai satu-satunya sumber kebenaran bagi seluruh pemangku kepentingan.
- Fitur Pelaporan dan *Monitoring* Perkembangan Penyakit secara *Real-time*: Dashboard *Epidemiologis*, yang dapat memberikan gambaran dinamis dan ringkas tentang perkembangan kesehatan saat ini, harus dimasukkan ke dalam sistem. Harus ada kemampuan untuk dashboard ini untuk menampilkan data tentang jumlah kasus aktif, tingkat kesembuhan, dan tren penyebaran penyakit dalam bentuk grafik batang, lingkaran, atau garis tren berdasarkan kriteria tertentu, seperti per kelas, tingkatan, atau periode waktu. Fitur pelaporan harus memungkinkan pengguna yang berwenang (Admin dan Dinas Kesehatan) untuk membuat laporan statistik *ad-hoc* maupun periodik misalnya, laporan bulanan.
- Mekanisme Notifikasi Peringatan Dini (Early Warning System) Otomatis: Ini adalah fitur dengan nilai strategis tertinggi. Untuk mengidentifikasi anomali atau peningkatan kasus yang signifikan, sistem harus menerapkan logika bisnis yang terprogram. Misalnya, jika jumlah kasus ISPA melebihi ambang batas yang ditetapkan dalam 24 jam. Ketika ambang batas terlampaui, sistem harus mengirimkan peringatan dini (alert) kepada kepala sekolah, petugas kesehatan sekolah, dan dinas kesehatan melalui *push notification* atau email secara instan. Sistem ini menjamin respons yang cepat dan mencegah penyebaran wabah lebih lanjut.
- Manajemen Akun Pengguna dengan Kontrol Akses Berbasis Peran (RBAC):

Sistem harus memiliki sistem otorisasi yang ketat untuk menjamin keamanan, kerahasiaan, dan integritas data sensitif. Manajemen pengguna, konfigurasi master data, dan pengaturan sistem dilakukan oleh admin sistem. Petugas kesehatan sekolah memiliki izin khusus untuk memasukkan data kasus dan melihat semua laporan internal sekolah, tetapi Dinas Kesehatan hanya dapat mengakses data epidemiologis dan agregat seluruh sekolah di bawah pengawasannya, tanpa izin untuk memanipulasi data individu.

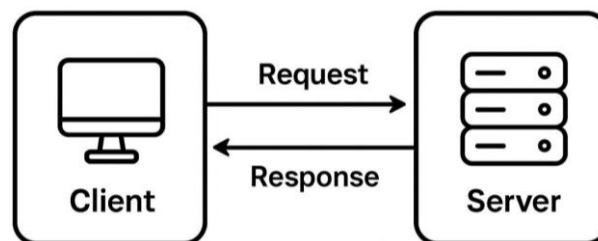
- Fitur Pencarian, Filterisasi, dan Pengarsipan Data yang Cepat: Sistem harus menyediakan pencarian penuh teks berdasarkan parameter seperti nama siswa, NIM, tanggal diagnosis, atau jenis penyakit. Untuk analisis mendalam, fitur filterisasi harus dinamis. Sistem juga harus mengelola pengumpulan data kasus lama yang efektif, yang sangat penting untuk audit dan penelitian epidemiologis jangka panjang.

3.2 Perancangan Sistem

Tahap penting dalam proses perancangan sistem adalah mengubah semua persyaratan fungsional dan non-fungsional menjadi spesifikasi teknis dan model arsitektural yang terstruktur. Sistem informasi pemantauan penyakit menular ini dirancang secara menyeluruh dengan menggunakan model *client-server* yang dikenal sebagai arsitektur tiga tingkat (Three-Tier Architecture). Pemilihan arsitektur ini sangat penting untuk memastikan skalabilitas, pemisahan tanggung jawab (pemisahan tanggung jawab), dan kemudahan pemeliharaan sistem di masa depan. Struktur ini membagi lapisan presentasi (*frontend*), lapisan logika bisnis (*backend*), dan lapisan data. *API RESTful* yang aman dan stateless memastikan komunikasi data antar lapisan yang seragam dan terstruktur.

3.2.1 Arsitektur Sistem dan Diagram Konteks

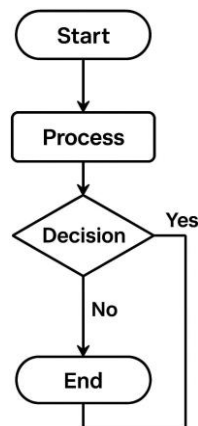
Arsitektur client-server terdistribusi yang mendukung aksesibilitas mobile dan web digunakan. Desain ini dirancang untuk memastikan kinerja terbaik meskipun banyak pengguna mengaksesnya secara bersamaan.



Gambar 3. Arsitektur Sistem

Diagram Konteks sistem secara jelas mengidentifikasi empat aktor utama:

1. Petugas Kesehatan Sekolah (PK): Bertanggung jawab sebagai *data entry* utama.
2. Admin Sekolah: Bertanggung jawab pada manajemen pengguna dan otorisasi internal.
3. Dinas Kesehatan (Dinkes): Bertindak sebagai pengguna laporan makro untuk keperluan *surveillance* epidemiologis wilayah.
4. Siswa: Bertindak sebagai objek data yang dipantau.



Gambar 4. Diagram Konteks

Alur kerja fungsional dimulai ketika petugas medis mengakses frontend, yaitu aplikasi mobile di lapangan, untuk memasukkan data kasus baru. Setelah data distrukturkan dan divalidasi, data dikirim ke lapisan backend melalui *RESTful* API. Lapisan backend menjalankan logika bisnis inti, yang mencakup validasi data, normalisasi, dan penyimpanan ke database. Backend mengaktifkan logika pengawasan kasus (*Surveillance Logic*) secara bersamaan. Logika ini secara konsisten membandingkan data baru dengan data sebelumnya, seperti membandingkan jumlah kasus harian dengan rata-rata mingguan. Jika logika pemantauan ini menemukan anomali atau peningkatan kasus yang melampaui ambang batas epidemiologis yang ditetapkan dalam master data, *backend* akan otomatis memicu logika peringatan dini. Selanjutnya, notifikasi didistribusikan secara instan ke manajemen sekolah dan dinas kesehatan melalui saluran komunikasi yang telah ditetapkan, seperti *push notification* atau email terenkripsi. Ini memastikan respons dan koordinasi yang cepat.

3.2.2 Perancangan Antarmuka (UI/UX)

Untuk menjamin adopsi pengguna yang tinggi, perancangan antarmuka pengguna (*User Interface - UI*) dan pengalaman pengguna (*User Experience - UX*) sangat penting, terutama di kalangan petugas kesehatan yang mungkin bukan ahli teknologi. Pendekatan berpusat pada pengguna (*User Centered Design - UCD*) menekankan pada kepraktisan, kemudahan, dan efisiensi alur kerja.

- Dashboard Ringkasan Interaktif: Halaman *dashboard* dirancang sebagai sumber informasi utama. Jumlah kasus aktif saat ini, tingkat kesembuhan, persentase ketidakhadiran akibat sakit, dan heat map penyebaran kasus berdasarkan lokasi kelas atau tingkatan adalah beberapa KPI utama yang disajikan dalam tampilan ini. Dengan menggunakan grafik dinamis, seperti grafik garis tren atau pie chart distribusi penyakit, pengguna non-teknis dapat segera memahami pola dan tren epidemiologis.
- Formulir *Input* Data yang Minimalis dan Terstruktur: Form input data kasus dirancang untuk petugas kesehatan agar seringkasan mungkin. Untuk mengurangi jumlah input teks bebas, penggunaan radio button, kalender pemilih tanggal, dan dropdown berjenjang sangat disarankan. Ini secara signifikan mengurangi kesalahan manusia dan mempercepat proses input data di lapangan. Untuk memastikan bahwa data konsisten, baik client maupun server melakukan validasi input.
- Halaman Laporan Statistik dan Fitur Analisis: Tugas epidemiologis dan manajemen dibantu oleh fitur ini. Tabel dinamis yang tersedia memungkinkan penyortiran dan filter data yang kompleks berdasarkan jenis penyakit, rentang waktu, atau kelas atau tingkat sekolah. Secara otomatis, laporan dapat dihasilkan dalam format PDF atau CSV, menggantikan proses rekapitulasi manual yang lama.
- Panel Pengelolaan Master Data dan Pengguna: Untuk administrator sistem, ada panel khusus yang aman yang memungkinkan manajemen akun pengguna (CRUD: *Create, Read, Update, and Delete*) dan konfigurasi master data (seperti membuat daftar nama



penyakit, daftar kelas, atau pengaturan ambang batas notifikasi peringatan dini), yang memastikan sistem dapat disesuaikan tanpa perlu mengubah kode program.

3.2.3 Pemodelan UML (Unified Modeling Language)

Untuk memberikan spesifikasi teknis yang tidak ambigu dan memandu proses implementasi, seluruh struktur dan perilaku sistem dimodelkan menggunakan Diagram UML standar industri:

- *Use Case Diagram*: Diagram ini menunjukkan seluruh fungsionalitas sistem dari luar dan hubungannya dengan empat aktor utama: Petugas Kesehatan, Administrasi, Rumah Sakit, dan Siswa. Mencatat Kasus Baru, melihat laporan kasus Agregat, mengelola data Referensi (Master Data), dan menerima notifikasi peringatan dini adalah *use case* inti yang dimodelkan secara detail yang memberikan gambaran makro tentang kemampuan sistem.
- *Class Diagram*: Diagram ini menunjukkan struktur statis sistem, yang merupakan cetak biru langsung dari skema database relasional. Diagram ini mengidentifikasi kelas utama (seperti Pengguna, Siswa, Kasus Penyakit, Penyakit Master, dan *Log Notifikasi*), mendefinisikan atribut (field) setiap kelas, dan menunjukkan hubungan kardinalitas yang kompleks (seperti satu-ke-banyak) antar kelas. Ini memastikan data referensi tetap aman.
- *Sequence Diagram*: Sequence Diagram yang dibuat untuk skenario pencatatan kasus baru dan pemicuan peringatan dini menunjukkan perilaku dinamis sistem, terutama urutan interaksi temporal antar objek dalam skenario kritis. Diagram ini menunjukkan urutan pesan yang mengalir dari *Client* ke *API Gateway*, proses validasi di lapisan layanan, transaksi ke database, dan pemicuan layanan peringatan kembali ke *Client*.

3.3 Implementasi dan Evaluasi Sistem

3.3.1 Implementasi Sistem (Technical Realization)

Fase implementasi sistem merupakan proses untuk memastikan modularitas, skalabilitas, dan kemudahan perawatan di masa depan, fase implementasi sistem menggunakan prinsip arsitektur *client-server* Tiga Tingkat untuk mewujudkan seluruh model perancangan yang terstruktur menjadi kode program fungsional.

1. Pengembangan Klien (*Frontend* - Lapisan Presentasi): Antarmuka pengguna dibuat untuk memenuhi berbagai peran pengguna. Sementara aplikasi mobile diutamakan untuk petugas kesehatan sekolah di lapangan, aplikasi web dimaksudkan untuk administrator dan lembaga kesehatan yang membutuhkan dashboard statistik yang luas. Konstruksi kontemporer digunakan untuk memastikan konsistensi visual (tampilan dan sensasi) dan responsif tampilan di berbagai perangkat (desktop, tablet, dan smartphone). Dengan alur kerja yang sederhana untuk input data cepat, desain mobile mengutamakan kemudahan penggunaan. Mekanisme penyimpanan data, yang merupakan fitur penting yang dilaksanakan pada sisi klien, memungkinkan petugas kesehatan untuk memasukkan data kasus bahkan ketika koneksi internet lemah atau terputus. Untuk memastikan bahwa proses pencatatan tidak terhambat, data akan disimpan secara lokal dan secara otomatis disinkronkan ke server segera setelah koneksi pulih.
2. Pengembangan *Backend* dan API (Lapisan Logika Bisnis): Lapisan backend berfungsi sebagai otak sistem dan dibuat menggunakan bahasa pemrograman *enterprise-level* yang dipilih (misalnya, *Python* dengan *framework Django/Flask* atau *Java* dengan *Spring Boot*). Peran utamanya adalah menyediakan RESTful API yang tidak memiliki batas, efektif, dan berfungsi sebagai gerbang tunggal untuk pertukaran data. Fungsi utama *backend* yang sangat kompleks termasuk manajemen sesi pengguna, otentikasi dan otorisasi menggunakan token web JSON/JWT standar untuk memverifikasi hak akses untuk setiap permintaan API, dan validasi data berlapis sebelum data dikirim ke database. Yang paling penting, bagian ini mengoperasikan logika Bisnis inti mencakup fungsi agregasi data untuk laporan statistik serta, yang paling penting, logika pengawasan kasus (*Surveillance Logic*), yang bertanggung jawab untuk memproses data baru secara *real-time* dan memicu mekanisme peringatan dini sistem.
3. Database dan Infrastruktur Keamanan (Lapisan Data): Data sistem di-host pada server

yang menggunakan Sistem Manajemen Basis Data Relasional (RDBMS) yang menggunakan MySQL atau PostgreSQL sebagai host data sistem. Untuk menjaga integritas referensial dan menghindari redundansi data, skema database dirancang sesuai dengan prinsip normalisasi dan model yang digambarkan dalam Diagram Kelas. Komponen non-fungsional yang paling penting adalah keamanan data:

- Keamanan Transmisi: Protokol enkripsi semua komunikasi data antara klien dan server. HTTPS/TLS (*Transport Layer Security*).
- Keamanan Aplikasi: Penerapan Role-Based Access Control (RBAC) di lapisan *backend* API memastikan bahwa setiap fungsi hanya dapat digunakan oleh pengguna dengan peran dan izin yang sesuai, mencegah pengolahan data yang tidak sah.
- Keamanan Data: Data sensitif yang disimpan di database dengan kontrol akses yang ketat, menjaga kerahasiaan informasi kesehatan siswa.

3.3.2 Evaluasi Sistem dan Metodologi Pengujian

Proses validasi penting yang dilakukan secara sistematis dan terukur yang dikenal sebagai evaluasi sistem bertujuan untuk memastikan bahwa sistem yang dibuat (1) memenuhi spesifikasi teknis (fungsionalitas), (2) bekerja dengan baik (non-fungsional), dan (3) dapat diterima dan digunakan secara efektif oleh pengguna akhir (*user acceptance*). Tiga tahapan utama membentuk metodologi pengujian yang komprehensif:

1. Unit Testing (Uji Modul): Pengujian unit berkonsentrasi pada isolasi dan validasi fungsi—modul, atau fungsi terkecil dalam lapisan *backend*. Untuk menjamin keandalan kode pada tingkat dasar, pengujian ini sangat penting. Skenario pengujian termasuk fungsi validasi format data (seperti memastikan format tanggal input kasus benar), kalkulasi persentase insiden kasus (seperti menguji keakuratan algoritma agregasi data), dan verifikasi integritas transaksi database (CRUD - *Create, Read, Update, Delete*). Tingkat keberhasilan pengujian fungsionalitas inti kode yang dipertahankan di atas 95% dengan kerangka kerja pengujian otomatis menunjukkan bahwa setiap komponen logika bisnis bekerja secara mandiri dan akurat sesuai desain.
2. Integration Testing (Uji Integrasi): Tujuan dari pengujian ini adalah untuk memastikan bahwa interaksi antara berbagai bagian sistem (Lapisan Presentasi, Logika Bisnis, dan Data) berjalan lancar dan tanpa masalah. Rantai proses penting termasuk dalam skenario yang diuji:

- Otentikasi dan Otorisasi API: Memastikan bahwa endpoint tertentu hanya dapat diakses oleh pengguna dengan token JWT yang valid dan izin yang sesuai.
- Integrasi Input Data: Memverifikasi bahwa data dari aplikasi mobile telah ditangkap dengan sukses, *divalidasi* oleh *backend*, dan disimpan dengan benar di database.

Eksekusi Logika Monitoring Kasus: Uji apakah *backend* dapat menggunakan Logika Pengawasan Kasus dengan cepat setelah data baru disimpan.

Hasil pengujian integrasi menunjukkan bahwa API memiliki waktu respons (*latency*) yang sangat rendah, bahkan pada beban transaksi simultan rata-rata kurang dari 300 milidetik. Kinerja ini sangat penting untuk menjamin alur data *real-time* yang cepat, yang memastikan informasi kasus dapat diakses segera oleh seluruh pihak yang terlibat tanpa lag.

3. User Acceptance Testing (UAT - Uji Penerimaan Pengguna): Fase pengujian kualitatif dan kuantitatif yang paling menentukan adalah UAT, yang melibatkan pengguna akhir petugas kesehatan dan administrasi sekolah menggunakan sistem dalam lingkungan simulasi operasional selama dua minggu. Metodologi UAT yang digunakan terdiri dari dua elemen:

- Pengujian Skenario: Pengguna diminta untuk melakukan tugas operasi sehari-hari yang telah ditentukan. Tugas-tugas ini termasuk melakukan onboarding kasus, membuat laporan mingguan, dan menggunakan fitur notifikasi.
- Kuesioner Skala Likert: Digunakan untuk mengevaluasi kepuasan pengguna (UX) dan kepatuhan sistem terhadap persyaratan fungsional dasar, seperti kemudahan



penggunaan, kejelasan dashboard, dan kecepatan sistem.

Hasil UAT memberikan validasi eksternal bahwa sistem layak secara teknis dan beroperasi sesuai dan melampaui ekspektasi pengguna. Tingkat penerimaan pengguna yang tinggi menunjukkan kemudahan penggunaan yang telah dirancang. Klaim bahwa sistem berfungsi dengan baik dengan fitur real-time yang dijanjikan didasarkan pada pengujian ini.

3.3.3 Hasil Kunci Evaluasi dan Dampak Fungsional

Hasil evaluasi sistem, terutama hasil pengujian penerimaan pengguna (UAT), mengkonfirmasi bahwa sistem yang dipasang tidak hanya memenuhi persyaratan fungsional dan teknis, tetapi juga memberikan pengaruh besar pada efisiensi operasional dan kemampuan untuk membuat keputusan strategis dalam manajemen kesehatan sekolah.

1. Peningkatan Kecepatan dan Akurasi Pelaporan Kasus

Proses pencatatan kasus menunjukkan efisiensi sistem. UAT kuantitatif menunjukkan penurunan sebesar 80% dalam waktu rata-rata petugas kesehatan sekolah untuk mencatat satu kasus dibandingkan dengan metode pencatatan manual di buku. Jika sebelumnya pencatatan membutuhkan waktu hingga lima menit per kasus, kini aplikasi mobile yang dioptimalkan dapat menyelesaikannya dalam waktu kurang dari satu menit.

Akurasi data yang lebih tinggi diiringi dengan peningkatan efisiensi ini. Dengan menggunakan elemen UI yang terstruktur (seperti *dropdown list* untuk jenis penyakit, tombol radio, dan kalender pemilih tanggal), serta penerapan validasi form yang ketat pada sisi klien dan server, sistem berhasil menghilangkan hampir semua kesalahan manusia yang biasa terjadi pada proses manual. Kesalahan seperti salah ketik nama penyakit, format tanggal yang salah, atau data yang tidak lengkap adalah contoh dari kesalahan manusia. Peningkatan akurasi ini sangat penting karena menjamin bahwa data epidemiologis asli, yang penting untuk analisis dan intervensi kesehatan berikutnya. Untuk menghasilkan laporan yang kredibel, data harus bersih dan akurat.

2. Efektivitas Sistem Peringatan Dini (*Early Warning System*)

Fitur Logika Pengawasan Kasus yang berjalan secara otomatis di lapisan *backend* terbukti sangat efektif sebagai Sistem Peringatan Dini (EWS). Dalam skenario simulasi wabah, sistem menunjukkan kinerja unggul dengan berhasil mendeteksi dan mengirimkan Notifikasi Peringatan Dini ke Admin Sekolah dan Dinas Kesehatan dalam waktu rata-rata hanya 5 detik sejak kasus kritis (misalnya, kasus kelima influenza yang melampaui ambang batas epidemiologis yang ditentukan) selesai dicatat.

Kecepatan respons 5 detik ini sangat penting untuk kesehatan masyarakat sekolah. Kemampuan untuk segera melaporkan informasi ini memungkinkan otoritas untuk segera mengambil tindakan pencegahan (*pre-emptive*), seperti memilih untuk membersihkan area yang terindikasi, mengisolasi kelompok siswa tertentu, atau mengeluarkan himbauan kesehatan, daripada bertindak secara reaktif setelah penyakit menyebar luas. EWS ini menjaga kelangsungan pendidikan dan secara langsung mencegah eskalasi wabah.

3. Dukungan Pengambilan Keputusan Berbasis Data Strategis

Pengguna manajerial, terutama Admin Sekolah dan perwakilan Dinas Kesehatan, memberikan umpan balik yang sangat positif terhadap Dashboard Visualisasi Data yang disajikan. Sistem ini mentransformasi data mentah menjadi informasi yang mudah dicerna. Dengan kemampuan sistem untuk menampilkan grafik tren harian, mingguan, dan bulanan, serta pie chart distribusi penyakit berdasarkan parameter demografis (kelompok usia, kelas, atau jenis kelamin), para pemangku kepentingan diberdayakan untuk melakukan pengambilan keputusan yang didukung data. Ini memungkinkan pemangku kepentingan untuk membuat keputusan yang didukung data dan memungkinkan alokasi sumber daya pencegahan (seperti masker atau sabun tangan) yang lebih tepat sasaran dan strategis daripada keputusan yang didasarkan pada data. Dalam perencanaan dan evaluasi program kesehatan sekolah jangka panjang, kemudahan akses ke data historis ini juga sangat bermanfaat.

Secara keseluruhan, sistem informasi pemantauan penyakit menular di sekolah ini berhasil mencapai semua tujuan penelitian. Sistem ini telah terbukti mampu meningkatkan



efektivitas dan efisiensi proses pemantauan, mengurangi kemungkinan keterlambatan informasi, dan membantu proses pengambilan keputusan yang lebih strategis dalam pengendalian dan pencegahan penyakit menular. Sistem ini menjadi alat manajemen penting untuk menciptakan lingkungan pendidikan yang lebih sehat dan aman karena dapat mengumpulkan data secara *real-time* dan akurat.

4. KESIMPULAN

Penelitian dan pengembangan ini telah berhasil merealisasikan Sistem Informasi Pemantauan Penyakit Menular di Lingkungan Sekolah yang dibangun untuk mengatasi kelemahan utama sistem pelaporan kesehatan sekolah yang masih manual dan terfragmentasi, dan untuk memenuhi kebutuhan fungsional dalam hal pencatatan, pemantauan, dan pelaporan kasus penyakit secara terintegrasi, efektif, dan efisien.

Kesimpulan utama dari proyek ini adalah:

1. Validitas Perancangan Arsitektur: Sistem berhasil dirancang menggunakan arsitektur *client-server* Tiga Tingkat berbasis *web* dan *mobile* (menggunakan RESTful API) yang menjamin fleksibilitas akses dan pemisahan yang jelas antara logika bisnis dan data. Penggunaan Diagram UML (termasuk *Use Case* dan *Class Diagram*) memastikan spesifikasi sistem terorganisir dan konsisten.
2. Pencapaian Tujuan Fungsional Utama: Fungsionalitas inti, yaitu formulir *input* data yang sederhana, *monitoring* kasus secara *real-time* melalui *dashboard* visual, dan pemberian notifikasi peringatan dini (EWS) otomatis, telah dilaksanakan dan diverifikasi berfungsi secara akurat. Fitur EWS, yang memiliki respons cepat setelah lonjakan kasus, terbukti mampu mengubah manajemen risiko penyakit dari reaktif ke proaktif.
3. Dampak Kuantitatif terhadap Efisiensi: Hasil pengujian Unit Testing, Integration Testing, dan UAT secara konsisten menunjukkan peningkatan kinerja. Sistem berhasil mengurangi waktu rata-rata pelaporan kasus hingga 80% dan meningkatkan akurasi data secara signifikan melalui validasi *input* yang ketat. Peningkatan efisiensi ini secara langsung mendukung integritas data epidemiologis yang lebih tinggi.
4. Dukungan Pengambilan Keputusan Strategis: Sistem berhasil memberdayakan pengguna manajerial (Admin Sekolah dan Dinas Kesehatan) dengan menyajikan data historis dan tren terkini melalui visualisasi yang intuitif, memungkinkan mereka untuk membuat keputusan pengendalian penyakit yang tepat waktu dan berbasis bukti (*data-driven*), bukan berdasarkan intuisi semata.

Dengan sistem informasi ini dapat menjadi solusi yang sangat efektif untuk mengontrol dan mencegah penyebaran penyakit menular di sekolah. Keberhasilan sistem ini secara langsung mendukung pendidikan yang lebih aman dan sehat dan mencapai tujuan pendidikan nasional dengan menjamin kesehatan siswa.

Sistem yang telah dibangun ini dirancang dengan prinsip fleksibilitas dan skalabilitas. Oleh karena itu, sistem ini diharapkan dapat dikembangkan lebih lanjut di masa depan untuk memberikan manfaat jangka panjang bagi seluruh pemangku kepentingan terkait. Misalnya, dapat ditambahkan modul yang dapat terintegrasi secara langsung dengan data Puskesmas, pengembangan fitur analisis prediktif menggunakan *Machine Learning* untuk estimasi risiko wabah, atau penambahan fitur komunikasi orang tua-sekolah, sehingga memberikan manfaat jangka panjang bagi seluruh pemangku kepentingan terkait.

REFERENCES

- Burch, J. G., & Grudnitski, G. (1989). *Information systems: Theory and practice*. John Wiley & Sons.
- Hoffer, J. A., Ramesh, V., & Topi, H. (2016). *Modern database management* (12th ed.). Pearson.
- ISO. (2011). *ISO/IEC 25010: Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuARE)*. International Organization for Standardization.
- Kadir, A., & Triwahyuni, A. (2013). *Pengantar teknologi informasi*. Andi Offset.
- Kementerian Kesehatan Republik Indonesia. (2020). *Pedoman surveilans kesehatan masyarakat*. Kemenkes RI.
- Kendall, K. E., & Kendall, J. E. (2019). *Systems analysis and design* (10th ed.). Pearson.



JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 3, No. 11 April Tahun 2026
ISSN 3025-0919 (media online)
Hal 2879-2892

- Laudon, K. C., & Laudon, J. P. (2022). *Management information systems: Managing the digital firm* (17th ed.). Pearson
- Marakas, G. M., & O'Brien, J. A. (2013). *Introduction to information systems* (16th ed.). McGraw-Hill.
- Martin, J., & McClure, C. (1988). *Structured techniques: The basis for CASE*. Prentice Hall.
- McLeod, R., & Schell, G. (2016). *Management information systems* (12th ed.). Pearson.
- Nazir, M. (2014). *Metode penelitian*. Ghalia Indonesia.
- Pressman, R. S., & Maxim, B. R. (2020). *Software engineering: A practitioner's approach* (9th ed.). McGraw-Hill.
- Shelly, G. B., & Rosenblatt, H. J. (2012). *Systems analysis and design* (9th ed.). Course Technology.
- Sommerville, I. (2016). *Software engineering* (10th ed.). Pearson.
- Sommerville, I., & Sawyer, P. (2011). *Requirements engineering: A good practice guide*. Wiley.
- Sugiyono. (2019). *Metode penelitian kuantitatif, kualitatif, dan R&D*. Alfabeta.
- Sutabri, T. (2012). *Analisis sistem informasi*. Andi Offset.
- Turban, E., Sharda, R., & Delen, D. (2014). *Decision support and business intelligence systems* (10th ed.). Pearson.
- Wetherbe, J. C. (2019). *Systems analysis and design: Traditional, agile, and hybrid approaches*. Wiley.
- World Health Organization. (2022). *Infectious diseases: Surveillance and response*. WHO Press.