



8.37%

SIMILARITY OVERALL

SCANNED ON: 20 JAN 2025, 11:41 PM

Similarity report

Your text is highlighted according to the matched content in the results above.

● IDENTICAL
0.17%

● CHANGED TEXT
8.19%

Report #24485693

BAB I PENDAHULUAN 1.1 Latar Belakang Pendidikan merupakan elemen utama dalam perkembangan masyarakat dan negara. Sekolah adalah lembaga penting dalam penyelenggaraan pendidikan, dan administrasi sekolah yang efisien adalah kunci dalam memastikan proses pendidikan berjalan dengan baik. Namun, dalam era yang semakin terhubung dan canggih, manajemen administrasi sekolah juga harus ikut bertransformasi. Sekolah Jakarta Academics, yang telah berusia 8 tahun, mewakili salah satu contoh sekolah baru yang menghadapi sejumlah tantangan dalam mengelola administrasi mereka. 40 Dalam beberapa tahun terakhir, perkembangan dan kemajuan teknologi telah mengubah cara sekolah mengelola data administrasi. Sayangnya, Sekolah Jakarta Academics masih mengandalkan proses manual dalam pengelolaan data mereka. Data administrasi, seperti data terkait siswa, orang tua siswa, serta data akademik siswa, tersebar dalam berbagai file terpisah. Hal tersebut tidak hanya menghambat efisiensi operasional, namun juga memperbesar peluang terjadinya kesalahan manusiawi dalam pengelolaan data tersebut. Dalam dunia yang semakin terkoneksi dan kompleks, tuntutan akan manajemen administrasi yang terintegrasi semakin mendesak. Data administrasi yang terintegrasi memungkinkan sekolah untuk memiliki pandangan yang lebih holistik terhadap operasional mereka, mengurangi potensi kesalahan, dan menyediakan

informasi yang lebih tepat yang mendukung pengambilan keputusan. Selain itu, manajemen administrasi yang terintegrasi juga memberikan penghematan dalam hal waktu dan sumber daya yang dihabiskan untuk proses administrasi. Beberapa masalah yang dihadapi oleh Sekolah Jakarta Academics dalam manajemen administrasi mereka meliputi: Kesulitan dalam mengakses dan mengintegrasikan data, data yang terpisah-pisah dalam file-file terpisah membuat sulit untuk menggabungkan dan mengakses informasi yang relevan dengan cepat dan efisien. Potensi kesalahan manusiawi, proses manual rentan terhadap kesalahan manusiawi seperti kesalahan pengetikan atau ketidakselarasan data. Pemborosan waktu dalam proses manual, proses manual memerlukan waktu yang signifikan dan sumber daya manusia yang berharga yang dapat dialokasikan untuk tugas-tugas lain yang lebih produktif. Tidak adanya sistem pelaporan yang efisien dan terstruktur, ketersediaan informasi secara cepat dan efisien merupakan hal yang kritis dalam pengambilan keputusan. Untuk mengatasi permasalahan-permasalahan ini, pengembangan sistem administrasi sekolah yang terintegrasi menjadi suatu keharusan. Sistem administrasi sekolah yang terintegrasi memungkinkan data terkait siswa, data orang tua, dan data akademik siswa untuk diakses dan dikelola dalam satu platform terpadu. Ini bukan saja meningkatkan efisiensi operasional,

namun juga mendukung sekolah dalam menyediakan laporan yang lebih akurat dan terstruktur. Dalam konteks ini, pemanfaatan teknologi menjadi elemen penting dalam merancang dan mengembangkan sistem administrasi sekolah yang terintegrasi. Framework Laravel dan mySQL dipilih sebagai basis pengembangan aplikasi integrasi ini A-1 karena kemampuan, keamanan, dan fleksibilitas yang mereka tawarkan. Dalam penelitian ini, akan diteliti dan dirancang aplikasi sistem administrasi sekolah yang terintegrasi berbasis Framework Laravel dan didukung oleh mySQL. Dengan adanya sistem administrasi sekolah yang terintegrasi ini, diharapkan Sekolah Jakarta Academics akan dapat mengatasi tantangan dalam pengelolaan data administrasi mereka secara lebih efisien, meningkatkan akurasi data, serta mendukung pengambilan keputusan yang lebih optimal. Selain itu, sistem ini akan menyediakan fondasi yang kokoh untuk perkembangan dan pertumbuhan Sekolah Jakarta Academics yang berkelanjutan dalam jangka panjang. Penelitian ini akan berfokus pada pengembangan aplikasi sistem administrasi sekolah yang terintegrasi untuk Sekolah Jakarta Academics. Dengan harapan bahwa aplikasi ini akan membantu sekolah dalam mengatasi tantangan dalam mengelola data administrasi secara efisien dan terintegrasi.

A-2 1.2 Identifikasi dan Rumusan Masalah

1.2.1 Identifikasi masalah 1. Pengelolaan Data yang Tersebar: Salah

REPORT #24485693

satu tantangan utama yang dialami oleh Sekolah Jakarta Academics adalah pengelolaan data yang tersebar dalam file-file terpisah. Data siswa, orang tua, dan akademik siswa disimpan dalam berbagai dokumen terpisah. Hal ini menyulitkan akses, integrasi, dan pemeliharaan data secara efisien.

2. Potensi Kesalahan Manusia: Proses administrasi yang masih dilakukan secara manual meningkatkan risiko kesalahan manusiawi. Kesalahan pengetikan data, pencatatan yang tidak akurat, atau kehilangan data menjadi ancaman serius terhadap akurasi dan integritas informasi administrasi sekolah.

3. Waktu dan Sumber Daya yang Tersisa-sia: Proses manual memerlukan waktu yang signifikan dan memanfaatkan sumber daya manusia yang berharga. Tenaga yang bisa dialokasikan untuk tugas-tugas lebih produktif menjadi terbuang untuk proses administrasi yang bisa diotomatisasi.

4. Ketidakterediaan Sistem Pelaporan yang Efisien: Sekolah Jakarta Academics menghadapi kesulitan dalam menyajikan dan mengakses informasi secara cepat dan efisien. Ini menghambat kemampuan sekolah dalam mengambil keputusan berdasarkan data administrasi yang akurat.

5. Tantangan Pertumbuhan dan Skalabilitas: Dalam jangka panjang, Sekolah Jakarta Academics mungkin akan mengalami pertumbuhan. Proses manual dan sistem yang tidak terintegrasi dapat menjadi hambatan dalam mengatasi pertumbuhan ini dengan efisien.

6. Ketidaksetaraan

Akses Informasi: Beberapa anggota staf sekolah mungkin memiliki akses terbatas ke informasi yang mereka butuhkan. Ini dapat mengganggu kolaborasi antar staf dan menghambat aliran informasi yang diperlukan dalam operasi sehari-hari. 7. Keamanan Data yang Rendah: Data administrasi sekolah yang penting seringkali tidak memiliki tingkat keamanan yang memadai. Kerentanan keamanan data ini dapat membahayakan privasi dan integritas informasi siswa, orang tua, dan sekolah. 1.2 **44** 2 Rumusan Masalah Bagaimana merancang dan membangun aplikasi untuk sistem administrasi dan akademik sekolah Jakarta Academics?

1.3 Ruang Lingkup dan Batasan Masalah 1.3 **29** 1 Ruang Lingkup

Penelitian ini berkonsentrasi pada pengembangan aplikasi administrasi dan akademik sekolah berbasis web yang terintegrasi, dengan ruang lingkup yang mencakup aspek berikut: 1. Manajemen Data Siswa Aplikasi akan mencakup proses pendaftaran, pembaruan profil, dan pengelolaan data siswa. Data ini mencakup informasi pribadi, riwayat akademik, dan data administrasi lainnya. A-3 2. Penjadwalan Pelajaran Sistem menyediakan fitur untuk menyusun dan mengelola jadwal pelajaran secara otomatis, termasuk alokasi guru, mata pelajaran, dan ruang kelas. 3. Pengelolaan Nilai dan Laporan Akademik Fitur ini memungkinkan pengelolaan nilai siswa, penyusunan laporan akademik, serta akses siswa dan orang tua terhadap informasi akademik secara real-time. 4. Portal Interaktif untuk Pengguna o Siswa dan Orang Tua: Portal ini memberikan akses bagi siswa dan orang tua untuk mengakses informasi akademik, dan jadwal pelajaran. o Guru dan Administrator: Portal untuk pengelolaan nilai, jadwal pelajaran, dan laporan. 5. Keamanan dan Integrasi Data Penelitian ini mencakup integrasi berbagai sumber data (data siswa, guru, orang tua, dan akademik) dalam satu sistem yang aman dan terpadu. 6. Penggunaan Teknologi Framework Laravel dan MySQL Pengembangan aplikasi dilakukan menggunakan Framework Laravel 11 untuk backend dan MySQL sebagai sistem pengelolaan basis data, dengan fokus pada skalabilitas,

keamanan, dan kemudahan pengembangan. 10 26 7. Metodologi Pengembangan

Waterfall Pengembangan aplikasi menggunakan metodologi Waterfall, meliputi tahapan perencanaan, analisis, perancangan, implementasi, pengujian, serta pemeliharaan.

8. Lingkungan Implementasi Penelitian ini diterapkan untuk Sekolah Jakarta Academics sebagai studi kasus, dengan fokus pada proses administrasi dan akademik sekolah tersebut.

Ruang lingkup ini diharapkan dapat menjawab permasalahan administrasi dan akademik yang ada serta memberikan dasar untuk pengembangan

lebih lanjut di masa depan. 1.3.2 Batasan Masalah Dalam penelitian "Rancang Bangun Aplikasi Administrasi dan Akademik Sekolah Terintegrasi Berbasis Web", beberapa batasan masalah ditetapkan agar penelitian ini tetap fokus dan terarah. Batasan masalah yang ditentukan dirinci berikut ini:

1. Lingkup Institusi Penelitian ini terbatas pada penerapan di Sekolah Jakarta Academics sebagai studi kasus, dan hasil pengembangan aplikasi disesuaikan dengan kebutuhan spesifik institusi tersebut.
2. Fitur Aplikasi Aplikasi yang dikembangkan hanya mencakup fitur administrasi dan akademik utama, yaitu: A-4 o Manajemen data siswa. o Penjadwalan pelajaran. o Pengelolaan nilai dan laporan akademik. o Portal interaktif untuk siswa, guru, orang tua, dan administrator. Fitur tambahan seperti pengelolaan inventaris, perpustakaan, atau sistem lainnya tidak termasuk dalam ruang lingkup penelitian.
3. Platform Pengembangan Penelitian ini berkonsentrasi pada pengembangan aplikasi berbasis web memanfaatkan Framework Laravel dan system manajemen basis data MySQL. Aplikasi ini tidak mencakup pengembangan aplikasi mobile atau desktop.
4. Metodologi Pengembangan Pengembangan aplikasi dilakukan menggunakan metode Waterfall, sehingga penelitian ini tidak mencakup metodologi pengembangan perangkat lunak lain, seperti Agile atau Scrum.
5. Pengguna Aplikasi Pengguna aplikasi yang menjadi fokus adalah: o Administrator sekolah. o Guru. o Siswa. o Orang tua siswa.
6. Integrasi Sistem Penelitian ini hanya mencakup integrasi data administrasi dan akademik sekolah

dalam satu platform. Sistem pihak ketiga, seperti sistem perpustakaan, sistem penilaian eksternal, atau layanan cloud tambahan, tidak menjadi bagian dari penelitian ini. 7. Keamanan Data Penelitian berfokus pada pengamanan data internal sekolah melalui fitur otentikasi dan otorisasi pengguna. Pengamanan data terhadap ancaman tingkat lanjut seperti serangan cyber kompleks tidak menjadi bagian dari penelitian ini. 8. Durasi Penelitian Penelitian ini dikerjakan dalam jangka waktu yang terbatas, sehingga pengujian aplikasi hanya mencakup pengujian fungsional dasar (unit testing dan integrasi), tanpa mencakup pengujian skala besar atau uji coba langsung dengan semua pengguna potensial. 9. Sumber Daya Penelitian ini dilakukan dengan sumber daya yang tersedia, termasuk server dan hosting sederhana yang memadai untuk kebutuhan awal aplikasi, tanpa mencakup infrastruktur skala besar seperti cloud hosting atau server high-performance. Dengan batasan-batasan tersebut, penelitian ini difokuskan pada penyelesaian masalah administrasi dan akademik sekolah secara efisien dan A-5 terintegrasi, dengan tetap mempertimbangkan waktu, sumber daya, dan kebutuhan prioritas institusi. 69

1.4 Maksud dan Tujuan Penelitian 1.4 1 Maksud Penelitian Penelitian "Rancang Bangun Aplikasi Administrasi dan Akademik Sekolah Terintegrasi Berbasis Web bermaksud untuk menyediakan solusi teknologi yang mampu mengatasi permasalahan administrasi dan akademik sekolah secara efisien dan terintegrasi.

Penelitian ini bertujuan untuk mendukung Sekolah Jakarta Academics dalam: 1.

27 Meningkatkan Efisiensi Operasional Dengan mengotomatisasi proses administrasi dan akademik yang selama ini dilakukan secara manual, penelitian ini bertujuan untuk menghemat waktu dan sumber daya sekolah. 2.

Menyediakan Sistem Terpadu Penelitian ini berkonsentrasi pada pengembangan aplikasi berbasis web yang mampu mengintegrasikan berbagai data administrasi dan akademik, seperti data siswa, orang tua, dan akademik, dalam satu platform. 3. Mengurangi Potensi Kesalahan Manusiawi Dengan digitalisasi dan pengintegrasian data,

penelitian ini bertujuan untuk meminimalkan kesalahan manual yang sering terjadi dalam pengelolaan data. 4. Mempermudah Pengambilan Keputusan Penelitian ini dirancang untuk menyediakan sistem pelaporan yang terstruktur, yang dapat digunakan oleh pihak sekolah untuk mendukung pengambilan keputusan berbasis data. 5. Meningkatkan Keamanan Data Penelitian ini diarahkan untuk memberikan sistem dengan keamanan yang lebih baik dalam mengelola data sensitif, seperti data siswa, dan laporan akademik. 6. Mendukung Skalabilitas dan Pertumbuhan Sekolah Dengan pengembangan sistem yang skalabel, penelitian ini dimaksudkan untuk menyediakan solusi jangka panjang yang dapat mendukung pertumbuhan dan perkembangan Sekolah Jakarta Academics di masa depan. 7. Meningkatkan Kualitas Layanan Pendidikan Melalui portal interaktif untuk siswa, guru, dan orang tua, penelitian ini bermaksud untuk memperbaiki kualitas pengalaman pengguna, memudahkan akses informasi serta mendukung peningkatan kualitas pendidikan Secara keseluruhan, maksud penelitian ini adalah memberikan solusi teknologi berbasis web yang dapat menjadi model pengelolaan administrasi dan akademik sekolah yang efisien, modern, dan terintegrasi, sesuai dengan kebutuhan institusi pendidikan di era digital.

A-6 1.4.2 Tujuan Penelitian Penelitian "Rancang Bangun Aplikasi Administrasi dan Akademik Sekolah Terintegrasi Berbasis Web bertujuan sebagai berikut:

1. Merancang Sistem Administrasi Sekolah yang Terintegrasi Mengembangkan aplikasi berbasis web yang dapat mengintegrasikan data siswa, orang tua, dan akademik dalam satu sistem yang terstruktur dan terpadu.
2. Meningkatkan Efisiensi Operasional Sekolah Mengotomatisasi proses administrasi dan akademik sekolah yang sebelumnya dikerjakan secara manual, sehingga lebih menghemat waktu dan tenaga yang dibutuhkan untuk tugas-tugas administratif.
3. Mengurangi Potensi Kesalahan Manual Meminimalkan risiko kesalahan manusia dalam pengelolaan data melalui sistem yang terstandarisasi dan digital.
4. Menyediakan Sistem Pelaporan yang Efisien dan

Akurat Mengembangkan fitur pelaporan yang memudahkan pihak sekolah dalam mengakses data yang relevan secara cepat, membantu menghasilkan keputusan yang lebih tepat dan berbasis data. 5. Meningkatkan Keamanan Data Sekolah Menjamin keamanan data administrasi dan akademik dengan menerapkan fitur otentikasi, otorisasi, dan perlindungan terhadap akses yang tidak sah. 6. Memudahkan Akses Informasi untuk Pengguna Menyediakan portal interaktif yang memungkinkan siswa, guru, dan orang tua mengakses informasi akademik dan administrasi secara mudah dan real-time. 7. Mendukung Pertumbuhan dan Skalabilitas Sekolah Membangun aplikasi yang dapat disesuaikan dengan kebutuhan sekolah seiring pertumbuhan jumlah siswa, guru, dan data yang dikelola. 8. Meningkatkan Kualitas Layanan Pendidikan Dengan integrasi sistem, aplikasi ini diharapkan dapat mendukung proses pendidikan yang lebih optimal melalui akses informasi yang mudah dan manajemen yang lebih sistematis. Dengan tujuan-tujuan ini, penelitian diharapkan dapat memberikan solusi inovatif yang tidak hanya menjawab kebutuhan administrasi sekolah saat ini, tetapi juga memberikan fondasi bagi pengembangan dan pertumbuhan sekolah di masa depan. **67** . A-7 BAB II TINJAUAN PUSTAKA

2.1 Landasan Teori 2.1 1 Rancang bangun Menurut A. Ray dalam jurnalnya, Perancangan adalah kumpulan prosedur yang bertujuan untuk menerjemahkan hasil analisis sistem ke dalam kode pemrograman, yang secara rinci menggambarkan bagaimana setiap komponen sistem akan direalisasikan. Di sisi lain, pengembangan melibatkan aktivitas pembuatan sistem baru atau penggantian serta penyempurnaan sistem yang sudah ada, baik secara menyeluruh maupun sebagian. Istilah "rancang bangun" atau desain mengacu pada proses umum dalam menciptakan atau merancang suatu objek, mulai dari tahap awal hingga penyelesaiannya. (Ray, 2020) Rancang bangun merupakan proses perencanaan dan pelaksanaan sistematis yang bertujuan untuk menghasilkan solusi terhadap suatu permasalahan atau kebutuhan

tertentu. Dalam konteks teknologi dan rekayasa, rancang bangun melibatkan serangkaian langkah mulai dari analisis kebutuhan, perancangan konsep, pengembangan desain, hingga implementasi solusi. Proses ini bertujuan untuk menciptakan suatu produk, sistem, atau infrastruktur yang memenuhi kriteria tertentu, baik dari segi fungsi, efisiensi, maupun keberlanjutan. Pengertian Rancang Bangun Menurut berbagai literatur, rancang bangun dapat didefinisikan sebagai kegiatan yang melibatkan pengorganisasian ide, desain, dan sumber daya untuk menghasilkan suatu produk yang dapat memenuhi kebutuhan spesifik. Proses ini tidak hanya mencakup aspek teknis tetapi juga memperhatikan faktor estetika, ergonomi, dan keberlanjutan. Tahapan Rancang Bangun

- o Analisis Kebutuhan Tahap ini bertujuan untuk mengidentifikasi dan memahami permasalahan atau kebutuhan yang akan diselesaikan. Analisis kebutuhan biasanya melibatkan pengumpulan data, wawancara, dan observasi untuk memperoleh informasi yang relevan.
- o Perancangan Konsep Setelah kebutuhan diidentifikasi, langkah selanjutnya adalah merancang konsep awal. Konsep ini dapat berupa gambaran kasar atau prototipe awal yang menggambarkan solusi yang diusulkan.
- o Pengembangan Desain Tahap ini melibatkan pembuatan desain detail berdasarkan konsep awal. Desain yang dikembangkan mencakup spesifikasi teknis, model, dan dokumentasi yang diperlukan untuk implementasi.
- o Implementasi dan Pengujian Setelah desain selesai, solusi yang dirancang diterapkan dalam bentuk nyata. Setelah implementasi selesai, tahap pengujian dilakukan untuk memastikan bahwa sistem yang dibangun sudah memenuhi kebutuhan yang telah ditentukan sebelumnya.
- o Evaluasi dan Perbaikan Setelah pengujian, dilakukan evaluasi terhadap hasil yang telah dicapai. Jika ditemukan kekurangan, perbaikan dilakukan untuk meningkatkan kualitas produk atau sistem.

Prinsip-Prinsip Rancang Bangun

- o Kesesuaian Fungsi Produk yang dirancang harus mampu memenuhi fungsi utama yang diharapkan.
- o Efisiensi Proses dan hasil rancang bangun harus memperhatikan

efisiensi sumber daya, termasuk waktu, biaya, dan material. o Keberlanjutan Desain harus mempertimbangkan dampaknya terhadap lingkungan dan memastikan bahwa solusi yang dihasilkan berkelanjutan. o Estetika dan Ergonomi Selain aspek teknis, desain juga perlu memperhatikan kenyamanan pengguna dan daya tarik visual. Rancang bangun banyak digunakan di berbagai bidang, seperti arsitektur, teknologi informasi, dan rekayasa industri. Dalam pengembangan perangkat lunak, misalnya, rancang bangun melibatkan pembuatan diagram alur, desain antarmuka, dan pengkodean. Sementara dalam bidang arsitektur, proses ini mencakup perancangan bangunan, pemilihan material, dan pengelolaan proyek konstruksi. 2.1 32 2 Aplikasi Aplikasi merupakan istilah yang merujuk pada perangkat lunak yang dibuat khusus untuk mendukung pengguna dalam menyelesaikan tugas-tugas tertentu. 11 Dalam konteks teknologi informasi, aplikasi mencakup berbagai jenis perangkat lunak yang dapat dijalankan di berbagai platform, seperti komputer, smartphone, tablet, atau perangkat lainnya. Aplikasi memiliki peran penting dalam kehidupan modern karena mampu meningkatkan produktivitas, efisiensi, dan aksesibilitas dalam berbagai aspek kehidupan. Pengertian Aplikasi Secara umum, aplikasi didefinisikan sebagai perangkat lunak yang dibuat khusus untuk memenuhi kebutuhan pengguna tertentu. Menurut Techopedia (2020), aplikasi merupakan perangkat lunak komputer yang dikembangkan untuk melakukan fungsi tertentu, baik itu untuk keperluan pribadi, bisnis, pendidikan, maupun hiburan. Dalam konteks ini, aplikasi berbeda dengan A-9 perangkat lunak sistem, yang berfungsi untuk mengelola operasi dasar perangkat keras. Jenis-Jenis Aplikasi o Aplikasi Desktop Aplikasi yang dirancang untuk dijalankan pada komputer atau laptop. Contohnya adalah Microsoft Word, Adobe Photoshop, dan AutoCAD. o Aplikasi Web Aplikasi web adalah aplikasi yang dapat diakses melalui browser internet. Contohnya adalah Google Docs, Gmail, dan e-commerce platform seperti Tokopedia. o Aplikasi Mobile

Aplikasi yang dibuat untuk perangkat mobile seperti smartphone dan tablet. 7

Contohnya adalah WhatsApp, Instagram, dan Gojek. o Aplikasi Enterprise Aplikasi yang dirancang untuk kebutuhan bisnis atau organisasi, seperti perangkat lunak Enterprise Resource Planning (ERP) dan Customer Relationship Management (CRM). o Aplikasi Hybrid Aplikasi yang memadukan fitur aplikasi native dan web, sehingga dapat dijalankan di berbagai platform dengan fleksibilitas tinggi. Tahapan Pengembangan Aplikasi o Analisis Kebutuhan Langkah awal ini melibatkan identifikasi kebutuhan pengguna dan tujuan aplikasi. Proses ini biasanya dilakukan melalui wawancara, survei, atau analisis pasar. o Perancangan Pada tahap ini, dilakukan pembuatan desain awal, termasuk diagram alur kerja, wireframe, dan mockup antarmuka pengguna (UI). o Pengembangan Proses pengembangan mencakup pembuatan kode program berdasarkan desain yang telah disepakati. Bahasa pemrograman yang digunakan tergantung pada jenis aplikasi yang akan dibuat. o Pengujian Setelah pengembangan aplikasi selesai, pengujian dilakukan untuk memastikan bahwa aplikasi berjalan sesuai harapan tanpa adanya bug atau masalah. o Peluncuran dan Pemeliharaan Aplikasi diluncurkan ke pasar atau pengguna akhir. Selanjutnya, dilakukan pemeliharaan untuk menjaga agar aplikasi tetap beroperasi dengan baik dan sesuai dengan kebutuhan pengguna. Prinsip-Prinsip Pengembangan Aplikasi o Kegunaan (Usability) A-10 Aplikasi harus mudah dioperasikan serta memberikan pengalaman yang intuitif bagi pengguna. o Fungsionalitas (Functionality) Aplikasi harus mampu menjalankan fungsi utama yang diharapkan dengan baik. o Keamanan (Security) Perlindungan terhadap data pengguna dan privasi menjadi aspek penting dalam pengembangan aplikasi. o Skalabilitas (Scalability) Aplikasi harus dirancang untuk dapat berkembang seiring dengan peningkatan jumlah pengguna atau kebutuhan fitur tambahan. Aplikasi memainkan peran signifikan dalam berbagai sektor kehidupan. Dalam pendidikan, aplikasi membantu proses belajar mengajar melalui

platform e-learning. Di bidang bisnis, aplikasi mendukung otomatisasi proses kerja dan pengelolaan data. Dalam kehidupan sehari-hari, aplikasi mobile menyediakan kemudahan untuk berkomunikasi, bertransaksi, hingga hiburan. 2.1.3 Administrasi Administrasi merupakan serangkaian kegiatan yang berhubungan dengan pengelolaan sumber daya, perencanaan, implementasi, serta pengawasan guna mencapai tujuan tertentu dengan cara efektif dan efisien. Dalam berbagai bidang, administrasi memiliki peran penting dalam memastikan kelancaran operasional, pengaturan sumber daya, dan pelaksanaan kebijakan yang telah ditetapkan. 24

Menurut para ahli, administrasi didefinisikan sebagai proses pengorganisasian dan pengelolaan yang mencakup perencanaan, koordinasi, pengendalian, dan evaluasi sumber daya untuk mencapai tujuan organisasi.

Administrasi tidak hanya terbatas pada aspek birokrasi, tetapi juga melibatkan pengelolaan data, dokumen, dan informasi secara sistematis. Fungsi Administrasi Administrasi memiliki beberapa fungsi utama yang saling berkaitan, yaitu:

- o Perencanaan (Planning) Menentukan tujuan serta strategi dan tahapan-tahapan yang perlu diambil untuk mencapainya.
- o Pengorganisasian (Organizing) Mengatur sumber daya, termasuk tenaga kerja, alat, dan waktu, agar dapat digunakan secara optimal.
- o Pengarahan (Directing) Memberikan panduan dan instruksi kepada anggota organisasi untuk menjalankan tugas sesuai dengan perencanaan.
- o Pengendalian (Controlling) Memastikan semua kegiatan berjalan sesuai dengan rencana serta melakukan perbaikan jika terjadi penyimpangan.
- o Evaluasi (Evaluating) Menilai hasil kegiatan untuk memastikan pencapaian tujuan yang telah ditetapkan.

A-11 Jenis-Jenis Administrasi

- o Administrasi Pemerintahan Mengelola kegiatan administratif yang berkaitan dengan pelayanan publik, pengelolaan kebijakan, dan pengawasan pemerintahan.
- o Administrasi Perusahaan Berfokus pada pengelolaan operasional perusahaan, seperti manajemen keuangan, sumber daya manusia, dan produksi.
- o Administrasi Pendidikan Menurut pandangan Lara Tastilia

dalam jurnalnya, akademik mencakup serangkaian aktivitas atau keseluruhan proses koordinasi kerja sama di antara sekelompok individu, yang dilakukan secara terstruktur dan sistematis untuk mencapai tujuan spesifik dalam suatu konteks, khususnya dalam lingkungan lembaga pendidikan formal. (Tastilia et al., 2022)

Administrasi mengatur kegiatan administratif di lingkungan pendidikan, seperti pengelolaan kurikulum, sumber daya pendidikan, dan sistem informasi akademik. o Administrasi Keuangan Berfokus pada pencatatan, pengelolaan, dan pelaporan keuangan dalam suatu organisasi atau institusi. Prinsip-Prinsip Administrasi o Efisiensi Administrasi harus dilakukan dengan meminimalkan pemborosan waktu, tenaga, dan biaya. o Efektivitas Kegiatan administrasi harus mewujudkan hasil yang selaras dengan tujuan yang telah direncanakan. o Transparansi Administrasi harus dilaksanakan secara terbuka agar dapat dipertanggungjawabkan. o Akuntabilitas Setiap kegiatan administratif harus dapat dipertanggungjawabkan kepada pihak yang berwenang. Koordinasi Administrasi harus melibatkan kerja sama antar individu atau antar bagian untuk mencapai hasil yang optimal. Administrasi berfungsi sebagai tulang punggung operasional organisasi. 33 Dengan administrasi yang baik, organisasi dapat mengelola sumber daya secara optimal, menyelesaikan tugas secara terstruktur, dan meningkatkan produktivitas.

Administrasi juga berperan dalam menciptakan sistem kerja yang efisien, mendukung pengambilan keputusan, dan menjamin keberlangsungan organisasi dalam jangka panjang. A-12 2.1.4 Web Menurut Mohamad Solahudin dalam jurnalnya, Website atau web merupakan sekumpulan halaman di internet yang menyajikan informasi tertentu sesuai dengan kebutuhan masing-masing. Web dirancang berdasarkan standar teknologi yang ditetapkan oleh World Wide Web Consortium (W3C). Standar ini bertujuan untuk memastikan bahwa web dapat digunakan secara lebih kompatibel di berbagai platform dan perangkat yang semakin beragam. (Informatika et al., 2021) Web, atau yang dikenal sebagai World

Wide Web (WWW), adalah salah satu teknologi utama yang memungkinkan akses dan pertukaran informasi melalui jaringan internet. Teknologi ini berperan sebagai platform untuk menyajikan konten dalam berbagai format, termasuk teks, gambar, dan video, dan lainnya, yang bisa diakses melalui perangkat dengan koneksi internet. Pengertian Web Secara umum, web adalah sistem informasi global yang memungkinkan pengguna untuk mengakses, berbagi, dan berinteraksi dengan data melalui browser internet. Tim Berners-Lee, pencipta web pada tahun 1989, mendefinisikan web sebagai ruang informasi tempat pengguna dapat mengakses dokumen dan data yang saling terhubung melalui hyperlink. Dalam konteks pengembangan teknologi informasi, web mengacu pada platform yang digunakan untuk membangun dan mengakses aplikasi berbasis jaringan. Komponen Utama Web

- o Browser Web Perangkat lunak yang dirancang untuk mengakses dan menampilkan konten web, seperti Google Chrome, Mozilla Firefox, atau Safari.
- o Server Web Komputer atau perangkat yang bertugas menyimpan dan menyajikan data atau halaman web kepada pengguna.
- o Protokol HTTP/HTTPS Protokol komunikasi yang berfungsi untuk mentransfer data antara browser web dan server web. HTTPS adalah versi yang lebih aman dengan enkripsi.
- o HTML, CSS, dan JavaScript

HTML (HyperText Markup Language): Bahasa pemrograman markup untuk membangun struktur konten web. **3 16 28**

CSS (Cascading Style Sheets): Bahasa pemrograman untuk mendesain tampilan dan layout halaman web. **3 20 28 58**

JavaScript: Bahasa pemrograman yang digunakan untuk meningkatkan interaktivitas pada halaman web.

Jenis-Jenis Web

- o Web Statis Halaman web yang kontennya tetap dan hanya dapat diubah dengan memodifikasi kode sumbernya secara manual.
- o Web Dinamis Halaman web yang kontennya dapat berubah secara otomatis berdasarkan interaksi pengguna atau data dari server.
- o Web Responsif Desain web yang dirancang untuk menyesuaikan tampilan dengan berbagai perangkat, seperti komputer, tablet, dan smartphone.
- o Web Aplikasi

Platform berbasis web yang menawarkan fungsi seperti aplikasi perangkat lunak, contohnya adalah Google Drive atau Trello. Fungsi dan Peran Web o Sebagai Media Informasi Web digunakan untuk menyebarkan informasi dalam berbagai format, mulai dari berita, artikel, hingga data publik. o Sebagai Alat Komunikasi Web memungkinkan komunikasi melalui media sosial, email, dan platform konferensi daring. o Sebagai Sarana Bisnis Web menjadi platform utama untuk e-commerce, pemasaran digital, dan layanan bisnis berbasis online. o Sebagai Alat Pendidikan Web menyediakan akses ke sumber belajar melalui platform e-learning, kursus online, dan materi edukasi. Prinsip-Prinsip Desain Web o Keterbacaan (Readability) Desain web harus memungkinkan pengguna membaca dan memahami konten dengan mudah. o Navigasi yang Mudah (Ease of Navigation) Struktur dan navigasi web harus intuitif agar pengguna dapat dengan mudah dan cepat memperoleh informasi yang mereka butuhkan. o Kecepatan Akses (Performance) Halaman web harus dimuat dengan cepat untuk memberikan pengalaman pengguna yang baik. o Keamanan (Security) Web harus dilengkapi dengan protokol keamanan untuk melindungi data pengguna dari ancaman seperti peretasan. o Kesesuaian (Compatibility) Web harus dapat berfungsi dengan baik di berbagai perangkat, browser, dan sistem operasi. Teknologi Terkait Web Web Development Frameworks: Seperti Laravel, React.js, dan Angular untuk mempercepat pengembangan aplikasi web. 20 42 Database: Teknologi seperti MySQL, MongoDB, dan PostgreSQL yang digunakan untuk menyimpan dan mengelola data dalam aplikasi web. A-14 API (Application Programming Interface): Digunakan untuk menghubungkan aplikasi web dengan layanan atau aplikasi lain. 2.1 21 5 Software Development Life Cycle (SDLC Software Development Life Cycle (SDLC) merupakan kerangka kerja terorganisir yang dimanfaatkan untuk merencanakan, mengembangkan, menguji, dan memelihara perangkat lunak. SDLC menyediakan pendekatan terstruktur untuk memastikan bahwa perangkat lunak yang dihasilkan memenuhi kebutuhan

pengguna dengan kualitas tinggi dan dalam waktu serta anggaran yang telah ditentukan. SDLC biasanya meliputi beberapa tahapan utama, yaitu:

- o Planning (Perencanaan) Tahap ini mencakup identifikasi kebutuhan proyek, penentuan ruang lingkup, tujuan, dan batasan sistem. Perencanaan juga melibatkan pengalokasian sumber daya dan penyusunan jadwal kegiatan. Hasil: Dokumen perencanaan proyek.
- o Requirement Analysis (Analisis Kebutuhan) Melibatkan identifikasi dan pengumpulan kebutuhan sistem dari pengguna atau stakeholder. Kebutuhan tersebut dapat berupa kebutuhan fungsional (apa yang dilakukan sistem) dan non-fungsional (keamanan, kinerja, dll.). Hasil: Dokumen yang menjelaskan spesifikasi kebutuhan perangkat lunak, dikenal sebagai Software Requirement Specification (SRS).
- o Design (Perancangan) Pada tahap ini, arsitektur sistem dan desain teknis dibuat berdasarkan spesifikasi kebutuhan. Desain mencakup:
 - ▣ High-Level Design (HLD): Gambaran arsitektur sistem secara umum.
 - ▣ Low-Level Design (LLD): Detail implementasi seperti struktur database, antarmuka pengguna, dan diagram alur sistem.Hasil: Dokumen desain sistem (System Design Document - SDD).
- o Implementation (Pengembangan) Tahap ini melibatkan pengkodean atau pengembangan perangkat lunak berdasarkan desain yang telah dirancang. Pemrogram menggunakan bahasa dan framework yang relevan. Penggunaan standar coding dan version control sangat penting untuk memastikan kualitas kode.
- o Testing (Pengujian) Sistem diuji untuk menjamin bahwa perangkat lunak bebas dari kesalahan serta sesuai dengan kebutuhan pengguna. Beberapa jenis pengujian meliputi:
 - ▣ Unit Testing: Pengujian modul atau fungsi secara individual.
 - ▣ Integration Testing: Pengujian interaksi antar modul.
 - ▣ System Testing: Pengujian keseluruhan sistem.
 - ▣ User Acceptance Testing (UAT): Pengujian oleh pengguna untuk memastikan sistem sesuai dengan kebutuhan.
- o Deployment (Penerapan) Setelah pengujian selesai dengan sukses, perangkat lunak diimplementasikan ke dalam lingkungan produksi. Proses

ini dapat melibatkan instalasi, migrasi data, dan pelatihan pengguna. o Maintenance (Pemeliharaan) Tahap ini meliputi perbaikan bug yang ditemukan setelah implementasi, pembaruan sistem, atau penambahan fitur baru sesuai kebutuhan pengguna.

2.1.6 Laravel

Menurut Purnama Sari di dalam jurnalnya , Laravel adalah kerangka kerja web berbasis PHP yang bersifat open-source dan gratis, dikembangkan oleh Taylor Otwell. **6 45** Framework ini dirancang untuk mendukung pembuatan dan pengelolaan aplikasi web dengan pola arsitektur MVC (Model-View-Controller). Pola MVC yang diterapkan Laravel memiliki perbedaan dari pola MVC konvensional, terutama pada penggunaan routing. Dalam Laravel, routing berfungsi sebagai penghubung antara permintaan pengguna (request) dan controller, sehingga controller tidak langsung menerima permintaan tersebut. (Purnama Sari & Wijanarko, 2020) Pengertian Laravel Laravel adalah kerangka kerja open-source berbasis PHP yang berfokus pada kesederhanaan dan efisiensi pengembangan aplikasi web. Laravel menyediakan berbagai fitur bawaan, seperti routing, authentication, session management, dan templating, yang membuat pengembangan aplikasi menjadi lebih cepat dan terorganisir. Menurut dokumentasi resminya, Laravel dirancang untuk "developer happiness" dengan menyediakan alat dan metode modern untuk menyelesaikan tugas-tugas kompleks secara elegan. Laravel mengikuti pola arsitektur MVC (Model-View-Controller):

1. Model Digunakan untuk berkomunikasi dengan database. **17** Model di Laravel menggunakan Eloquent ORM untuk mempermudah pengelolaan database secara object-oriented.
2. View Bagian antarmuka pengguna yang menggunakan blade templating engine Laravel untuk menghasilkan halaman dinamis. **54**
3. Controller Menghubungkan Model dan View dengan mengelola logika aplikasi dan menangani permintaan dari pengguna.

Fitur-Fitur Utama Laravel

1. Routing Laravel menawarkan sistem routing yang sederhana namun fleksibel untuk menangani URL dan permintaan HTTP.
2. Eloquent ORM (Object Relational Mapping) Laravel menyediakan ORM bawaan yang

mempermudah interaksi dengan database menggunakan sintaks PHP yang sederhana. A-16 3.

6 Blade Templating Engine Blade adalah sistem templating yang memungkinkan penggunaan sintaks dinamis dalam file HTML, mendukung inheritance dan komponen. 4. Migration dan Seeder Laravel menyediakan alat untuk mengelola struktur database melalui migration dan populasi data awal dengan seeder. 5 39 5. Middleware Middleware memungkinkan pengembang untuk memfilter permintaan HTTP masuk sebelum mencapai aplikasi, seperti autentikasi atau logging. 6. Authentication dan Authorization Laravel memiliki sistem bawaan untuk mengelola autentikasi pengguna dan izin akses, sehingga keamanan aplikasi lebih mudah dikelola. 7. Task Scheduling dan Queue Laravel mendukung task scheduling untuk menjalankan tugas berkala dan queue untuk pemrosesan tugas di latar belakang. 8. Testing Laravel menyediakan alat bawaan untuk melakukan pengujian unit dan integrasi aplikasi. Keunggulan Laravel 1. Kemudahan Penggunaan Dokumentasi lengkap dan sintaks yang bersih membuat Laravel mudah dipelajari bahkan untuk pengembang pemula. 5 17 43 2. Komunitas yang Besar Laravel memiliki komunitas aktif yang menyediakan berbagai paket, tutorial, dan dukungan teknis. 3. Dukungan Modular Laravel mendukung pengembangan aplikasi berbasis modul untuk mempermudah pengelolaan proyek besar. 5 6 51 4. Keamanan Laravel dilengkapi dengan berbagai fitur keamanan, seperti perlindungan terhadap SQL Injection, XSS, dan CSRF. 5. Skalabilitas Laravel cocok untuk aplikasi kecil maupun besar, dengan dukungan yang baik untuk penskalaan horizontal. Workflow Pengembangan Laravel 1. Pembuatan Project Laravel menggunakan Composer sebagai manajer dependensi. Perintah `composer create-project` atau `laravel new` digunakan untuk memulai proyek baru. 2. Pengaturan Routing File `routes/web.php` digunakan untuk mendefinisikan jalur aplikasi. 3. Pembuatan Model, Controller, dan Views A-17 Model dibuat untuk berinteraksi dengan database, Controller untuk logika aplikasi, dan View untuk antarmuka pengguna. 4. Migrasi dan Seeder Struktur

database diatur menggunakan migrasi, dan data awal diisi menggunakan seeder. 5. Pengujian dan Debugging Laravel menyediakan alat seperti PHPUnit dan Laravel Telescope untuk pengujian dan debugging.

Prinsip-Prinsip dalam Laravel 1. DRY (Don't Repeat Yourself) Laravel mendukung pendekatan untuk menghindari pengulangan kode melalui fitur seperti helper functions dan traits. 2. Convention Over Configuration Laravel menggunakan konvensi standar untuk mempermudah pengembangan tanpa memerlukan konfigurasi berlebih. 3. Service Container Laravel menyediakan kontainer layanan untuk mengelola dependensi aplikasi. 4. RESTful API Design Laravel memfasilitasi pembuatan API yang sesuai dengan standar REST.

2.1.7 Database

Database merupakan sekumpulan data yang disusun dengan cara sistematis untuk memudahkan akses, pengelolaan, dan pembaruan. Connolly dan Begg (2015) mendefinisikan database sebagai sekumpulan data yang saling terkait, dirancang untuk memenuhi kebutuhan informasi dalam suatu organisasi. **36** Data disimpan secara digital dan dikelola menggunakan perangkat lunak bernama Sistem Manajemen Basis Data (Database Management System/DBMS). DBMS merupakan perangkat lunak yang dirancang untuk mendefinisikan, membuat, mengelola, serta mengatur akses. terhadap database. Beberapa fungsi utama DBMS meliputi:

- o Penyimpanan dan pengelolaan data secara efisien.
- o Pengamanan data dari akses yang tidak sah.
- o Menyediakan cara mudah untuk mengakses dan memanipulasi data.

4 15 Contoh DBMS yang umum digunakan meliputi MySQL, PostgreSQL, Oracle Database, dan Microsoft SQL Server.

Jenis-Jenis Database

1. Database Relasional (Relational Database)

Adalah jenis database yang menyimpan data dalam tabel-tabel yang saling terhubung. Setiap tabel terdiri dari kolom (atribut) dan baris (rekaman). Contoh: MySQL, PostgreSQL, SQLite.

2. Database NoSQL

A-18 Digunakan untuk menangani data yang tidak terstruktur atau semi- terstruktur. **4 19 57** NoSQL mencakup berbagai model seperti dokumen, kolom, graf, dan penyimpanan berbasis key-value. **4 19** Contoh:

MongoDB, Cassandra, Redis. 2.1 9 18 34 8 MySQL MySQL merupakan salah satu sistem manajemen basis data relasional (Relational Database Management System/RDBMS) yang sangat populer dan banyak dipakai di dunia. Dibangun oleh MySQL AB pada tahun 1995, MySQL kini dimiliki oleh Oracle Corporation setelah akuisisi pada tahun 2010. MySQL dikenal karena kinerjanya yang cepat, skalabilitasnya yang baik, dan sifatnya yang open-source, menjadikannya pilihan utama untuk berbagai aplikasi, termasuk pengembangan web. 4 9 15 30 Pengertian MySQL MySQL merupakan sistem manajemen basis data relasional yang memanfaatkan Structured Query Language (SQL) sebagai bahasa utama untuk mengelola data. MySQL berfungsi untuk menyimpan, mengakses, dan memanipulasi data yang tersimpan dalam tabel-tabel yang saling berhubungan. Dalam lingkungan aplikasi, MySQL sering digunakan sebagai penyimpanan data untuk sistem backend. Komponen Utama MySQL o Database Kumpulan data yang terstruktur dalam tabel-tabel yang saling berhubungan. o Tabel Struktur utama dalam database yang digunakan untuk menyimpan data dalam bentuk baris (rows) dan kolom (columns). o Query SQL Perintah yang digunakan untuk melakukan operasi pada database, seperti SELECT, INSERT, UPDATE, dan DELETE. o MySQL Server Komponen inti MySQL yang bertanggung jawab untuk menyimpan data, memproses query, dan mengelola koneksi. Kelebihan MySQL o Kecepatan dan Efisiensi MySQL dirancang untuk memberikan kinerja tinggi, terutama dalam lingkungan yang membutuhkan akses data secara cepat. o Open Source MySQL memiliki sifat open-source, sehingga dapat dipakai tanpa Batasan dan tanpa biaya lisensi. Namun, terdapat versi komersial dengan fitur tambahan. o Portabilitas MySQL dapat beroperasi di banyak sistem operasi, seperti Windows, Linux, dan macOS. A-19 o Dukungan untuk Bahasa Pemrograman MySQL mendukung integrasi dengan banyak bahasa pemrograman, seperti PHP, Python, Java, dan lainnya. o Keamanan MySQL menawarkan mekanisme keamanan termasuk autentikasi pengguna, pengaturan hak akses, dan enkripsi

koneksi. o Skalabilitas MySQL cocok untuk aplikasi kecil hingga aplikasi besar yang membutuhkan jutaan baris data. Fungsi MySQL o Penyimpanan Data MySQL digunakan untuk menyimpan data dalam struktur tabel yang terorganisir. o Pengelolaan Data Mendukung operasi pengelolaan data seperti menyisipkan, memperbarui, dan menghapus data. o Query Data MySQL memungkinkan pengguna untuk mengekstrak informasi spesifik dari database dengan menggunakan perintah SQL. o Integrasi dengan Aplikasi MySQL berfungsi sebagai backend untuk berbagai aplikasi, terutama dalam pengembangan aplikasi berbasis web.

Jenis-Jenis Tipe Data MySQL o Numeric Digunakan untuk data angka, seperti INT, FLOAT, DOUBLE. o String/Text Digunakan untuk data teks, seperti VARCHAR, CHAR, TEXT. o Date/Time Digunakan untuk data waktu, seperti DATE, DATETIME, TIMESTAMP. o Spatial Tipe data khusus untuk menyimpan data geografis, seperti POINT dan POLYGON.

Prinsip-Prinsip Desain Database MySQL o Normalisasi Proses pengorganisasian data untuk mengurangi redundansi dan meningkatkan integritas data. o Indeks (Indexing) Digunakan untuk mempercepat akses data dengan membuat indeks pada kolom tertentu. o Konsistensi Data Menjaga agar data tetap akurat dan konsisten melalui penggunaan relasi dan kunci utama (primary key). o Keamanan A-20 Membatasi akses ke database dengan menggunakan pengguna dan kata sandi, serta mengatur hak akses. MySQL sering digunakan bersama dengan bahasa pemrograman PHP untuk membangun aplikasi berbasis web. Kombinasi ini menjadi dasar untuk banyak Content Management Systems (CMS) populer, seperti WordPress, Joomla, dan Drupal. MySQL juga mendukung pengembangan aplikasi modern berbasis framework, seperti Laravel dan CodeIgniter.

2.1 **3 12** 9 JavaScript JavaScript adalah bahasa pemrograman paling populer dan luas digunakan di dunia untuk pembuatan aplikasi web. **3 12 16 31** Bahasa ini pertama kali dikembangkan pada tahun 1995 oleh Brendan Eich, JavaScript awalnya dirancang untuk menambah interaktivitas pada halaman web. Saat ini,

JavaScript telah berkembang menjadi bahasa pemrograman serbaguna (general-purpose language) yang dapat dijalankan baik dari sisi klien (client-side) maupun dari sisi server (server-side).

Pengertian JavaScript JavaScript adalah bahasa pemrograman tingkat tinggi dengan sifat dinamis, lemah dalam tipe (loosely typed), dan berbasis objek (object-based). JavaScript dirancang untuk memungkinkan pengembang menciptakan antarmuka web yang interaktif dan responsif. JavaScript berbeda dari Java, meskipun memiliki nama yang mirip, karena keduanya dirancang untuk tujuan dan lingkungan yang berbeda.

Fungsi Utama JavaScript Interaktivitas JavaScript digunakan untuk menambah elemen interaktif pada halaman website, termasuk validasi form, animasi, dan pengelolaan atau pengubahan DOM (Document Object Model). Pengendalian Peristiwa (Event Handling) JavaScript dapat mendeteksi dan merespons tindakan pengguna, seperti klik, hover, atau pengisian formulir. Manipulasi DOM JavaScript memungkinkan pengembang untuk menambahkan, mengubah, atau menghapus elemen HTML dan CSS secara dinamis. JavaScript digunakan untuk mengambil dan mengirim data ke server menggunakan teknik seperti AJAX dan Fetch API. Dengan teknologi seperti Node.js, JavaScript dapat digunakan untuk membangun aplikasi sisi server yang cepat dan efisien.

Karakteristik JavaScript

- o Interpreted Language JavaScript dieksekusi langsung oleh browser tanpa memerlukan proses kompilasi.
- o Dynamic Typing A-21 JavaScript tidak memerlukan deklarasi tipe variabel secara eksplisit, sehingga bersifat fleksibel.
- o Event-Driven JavaScript dirancang untuk merespons berbagai peristiwa yang terjadi pada halaman web.
- o Cross-Platform JavaScript bisa dioperasikan pada berbagai perangkat dan sistem operasi melalui browser modern. Penggunaan JavaScript dalam Pengembangan Web
- o Front-End Development Framework dan pustaka seperti React.js, Angular, dan Vue.js digunakan untuk membangun antarmuka pengguna.
- o Back-End Development Dengan Node.js, JavaScript dapat digunakan untuk membangun server web yang cepat dan efisien.

o Full-Stack Development Kombinasi framework front-end dan back-end memungkinkan pengembang menggunakan JavaScript di seluruh tumpukan pengembangan. o Pengembangan Aplikasi Mobile Framework seperti React Native dan Ionic memungkinkan pengembangan aplikasi mobile menggunakan JavaScript. Kelebihan JavaScript o Mudah Dipelajari Sintaks JavaScript relatif sederhana dan mudah dipahami oleh pemula. o Kompatibilitas Browser Didukung oleh hampir semua browser modern tanpa memerlukan instalasi tambahan. o Ekosistem yang Luas Ekosistem JavaScript yang kaya menyediakan pustaka dan framework untuk berbagai kebutuhan. o Kecepatan Eksekusi langsung di browser tanpa kompilasi membuat JavaScript sangat cepat untuk tugas-tugas tertentu. 8

Kekurangan JavaScript o Keamanan Karena dieksekusi di sisi klien, JavaScript rentan terhadap serangan seperti XSS (Cross-Site Scripting). o Keterbatasan Browser Perbedaan implementasi JavaScript di berbagai browser dapat menyebabkan inkonsistensi. o Debugging yang Rumit A-22 Pemrograman asinkron dapat membuat proses debugging lebih sulit. 2.1 46

10 Waterfall Metode Waterfall adalah model dalam pengembangan perangkat lunak yang memakai pendekatan sistematis dan berurutan. Model ini dikenal sebagai salah satu pendekatan tradisional dalam pengembangan perangkat lunak, yang berfokus pada pemisahan setiap tahap dalam proses pengembangan. 11 47

Nama Waterfall menggambarkan alur proses yang mengalir ke bawah seperti air terjun, dari tahap satu ke tahap berikutnya. Pengertian Waterfall Waterfall adalah metode pengembangan perangkat lunak yang mengikuti tahapan-tahapan yang telah ditentukan, di mana setiap tahap harus diselesaikan sepenuhnya sebelum melanjutkan ke tahap selanjutnya. 14 41

Model ini pertama kali diperkenalkan oleh Winston W. Royce pada tahun 1970 sebagai pendekatan yang terstruktur untuk pengembangan sistem. Pada model ini, perubahan atau revisi hanya dapat dilakukan di tahap tertentu, sehingga perencanaan awal menjadi sangat penting. 60

Model ini cocok digunakan untuk proyek dengan kebutuhan yang sudah jelas dan stabil. Tahapan Waterfall Metode Waterfall terdiri dari beberapa tahapan

utama, yaitu:

- o Requirement Analysis (Analisis Kebutuhan) Tahap ini bertujuan untuk mengumpulkan dan mencatat kebutuhan pengguna dan sistem secara rinci. Tahap ini biasanya menghasilkan dokumen spesifikasi kebutuhan perangkat lunak (Software Requirement Specification).
- o System Design (Perancangan Sistem) Pada tahap ini, arsitektur sistem dirancang sesuai dengan kebutuhan yang telah dianalisis. Desain ini meliputi struktur data, arsitektur perangkat lunak, serta antarmuka pengguna.
- o Implementation (Implementasi) Tahap ini melibatkan proses pengkodean perangkat lunak berdasarkan desain yang telah dibuat. Setiap modul sistem dikembangkan dan diuji secara terpisah.
- o Integration and Testing (Integrasi dan Pengujian) Setelah tahap implementasi selesai, seluruh modul digabungkan menjadi satu kesatuan sistem yang lengkap. Pengujian kemudian dikerjakan untuk menjamin perangkat lunak bekerja sesuai dengan kebutuhan yang telah ditetapkan.
- o Deployment (Penerapan) Perangkat lunak yang telah diuji diterapkan di lingkungan produksi dan siap digunakan oleh pengguna akhir.

A-23

- o Maintenance (Pemeliharaan) Tahap ini mencakup perbaikan bug, penyesuaian untuk perubahan kebutuhan, dan peningkatan sistem setelah perangkat lunak dirilis.

Karakteristik Model Waterfall

- o Linear dan Berurutan Setiap langkah dalam model ini dilakukan secara berurutan, dan tidak dapat melanjutkan ke tahap berikutnya sebelum tahap sebelumnya selesai.
- o Dokumentasi yang Lengkap Model ini menekankan dokumentasi yang mendetail di setiap tahap, yang menjadi acuan dalam proses pengembangan.
- o Kebutuhan yang Stabil Model ini sesuai untuk proyek di mana kebutuhan sudah jelas dan tidak berubah selama proses pengembangan.
- o Pengendalian yang Ketat Model Waterfall memungkinkan pengendalian proyek yang ketat melalui hasil yang terukur di setiap tahap.

Keunggulan Model Waterfall

- o Struktur yang Jelas Dengan alur yang terdefinisi, model ini memudahkan tim pengembang untuk mengikuti proses secara terstruktur.
- o Dokumentasi yang Terorganisir Dokumentasi

yang lengkap membantu dalam pemeliharaan sistem di masa depan dan memberikan acuan yang jelas untuk semua pihak. o Cocok untuk Proyek Kecil Model ini sangat efektif untuk proyek kecil dengan kebutuhan yang sudah terdefinisi dengan baik. o Mudah Dikontrol Setiap tahap memiliki hasil yang dapat diukur, sehingga memudahkan pengendalian proyek. Kekurangan Model Waterfall o Kurangnya Fleksibilitas Perubahan pada kebutuhan atau desain sulit dilakukan setelah tahap tertentu selesai. o Tidak Cocok untuk Proyek Kompleks Pada proyek yang kompleks atau dengan kebutuhan yang sering berubah, model ini kurang efektif. o Waktu Pengujian yang Terbatas Pengujian dilakukan di tahap akhir, sehingga potensi masalah mungkin baru terdeteksi setelah pengembangan selesai. o Resiko Kegagalan yang Tinggi Jika terjadi kesalahan di awal, dampaknya dapat dirasakan hingga tahap akhir. A-24 Penerapan Model Waterfall Metode Waterfall biasanya digunakan dalam situasi berikut: o CProyek kecil dengan kebutuhan yang telah jelas. o Proyek yang membutuhkan dokumentasi yang terstruktur dan lengkap. o Sistem dengan persyaratan yang tidak sering berubah, seperti sistem perbankan atau aplikasi yang sudah memiliki standar tertentu. 2.1 2 11 UML (Unified Modeling Language) Menurut Dede Wira Trise Putra dalam jurnalnya, UML merupakan standar bahasa yang umum dipakai di industri untuk mendefinisikan kebutuhan sistem, melakukan analisis dan desain, serta memvisualisasikan arsitektur dalam pengembangan perangkat lunak berbasis pemrograman berorientasi objek. (Putra & Andriani, 2019) Unified Modeling Language (UML) adalah suatu standar pemodelan visual yang dimanfaatkan untuk merancang, mendokumentasikan, dan memahami sistem perangkat lunak. UML dibangun oleh Object Management Group (OMG) pada tahun 1997 dan sejak saat itu menjadi alat penting dalam proses pembuatan perangkat lunak yang berorientasi objek (object-oriented). Pengertian UML UML adalah sebuah bahasa pemodelan yang menyediakan alat visualisasi untuk menggambarkan

struktur dan perilaku sistem perangkat lunak. UML bukanlah metode atau proses pengembangan, melainkan alat bantu yang dapat digunakan di berbagai metode pengembangan perangkat lunak seperti Agile, Waterfall, dan lain-lain. UML dipakai untuk memodelkan berbagai aspek sistem, mulai dari struktur statis hingga dinamika interaksi antar komponen dalam sistem tersebut. Tujuan UML Visualisasi Membantu menggambarkan desain sistem secara visual sehingga lebih mudah dipahami oleh pengembang, pemangku kepentingan, dan klien. Jenis Diagram UML UML memiliki 14 diagram yang terbagi menjadi dua kategori utama: Structural Diagram dan Behavioral Diagram. 25 1.

Structural Diagram Diagram ini memvisualisasikan struktur statis dari sistem, termasuk komponen-komponen sistem dan relasinya. **Class Diagram** Menggambarkan kelas-kelas dalam sistem dan hubungan antar kelas. Contoh:

Atribut dan metode suatu kelas. **Object Diagram** Menunjukkan objek tertentu dan relasi antar objek pada suatu waktu tertentu. **Component Diagram** Menggambarkan komponen perangkat lunak dan hubungan antar komponen. **Deployment Diagram** Memvisualisasikan penyebaran perangkat keras dan perangkat lunak dalam sistem. **Package Diagram** Menggambarkan pengelompokan elemen-elemen sistem ke dalam paket-paket. **Composite Structure Diagram** Mengilustrasikan struktur internal dari kelas atau komponen. **2. Behavioral Diagram** Diagram ini menggambarkan dinamika dan perilaku sistem. **Use Case Diagram** Menampilkan hubungan antara aktor (pengguna atau sistem lain) dan sistem. Contoh: Aktor "User" yang melakukan aksi "Login". **Activity Diagram** Memodelkan proses kerja atau aktivitas yang berlangsung dalam sistem. **Sequence Diagram** Menunjukkan urutan hubungan antara objek dalam sistem. **Komponen Dasar UML** **o Kelas (Class)** Merepresentasikan entitas dalam sistem yang memiliki atribut dan operasi. **o Objek (Object)** Instansi dari kelas yang berisi nilai spesifik untuk atributnya. **o Relasi (Relationships)** Hubungan antar elemen, seperti asosiasi, agregasi, komposisi, dan pewarisan.

o Aktor (Actor) Entitas eksternal yang berinteraksi dengan sistem.

o Pesan (Messages) Komunikasi antar elemen dalam sistem. Kelebihan UML 1. **55**

Standar Internasional UML digunakan secara luas dan diakui sebagai standar dalam pemodelan perangkat lunak. 2. Fleksibilitas Dapat digunakan pada berbagai jenis proyek dan metode pengembangan. 3. Visualisasi yang Jelas Memberikan gambaran visual yang memudahkan pemahaman sistem. A-26 4. Interoperabilitas Mendukung integrasi dengan alat bantu pemodelan lain dan dokumentasi proyek. 5. Mendukung Pendekatan Berorientasi Objek UML dirancang khusus untuk sistem berorientasi objek, tetapi juga dapat digunakan untuk pendekatan lain. Kekurangan UML 1. Kompleksitas Untuk proyek kecil, UML dapat terasa terlalu kompleks dan membutuhkan upaya tambahan. 2. Kurva Pembelajaran yang Curam Pemahaman UML membutuhkan waktu, terutama bagi pengembang pemula. 3. Dokumentasi Berlebih Jika tidak diatur dengan baik, UML dapat menghasilkan dokumentasi yang terlalu banyak dan sulit diatur. Penerapan UML UML digunakan dalam berbagai tahap pengembangan perangkat lunak, seperti:

- o Analisis Kebutuhan: Untuk memahami kebutuhan pengguna dan sistem (Use Case Diagram).
- o Perancangan: Untuk merancang struktur sistem (Class Diagram, Component Diagram).
- o Implementasi: Untuk memandu pengembangan kode.
- o Pengujian: Untuk memastikan setiap elemen sistem sesuai dengan spesifikasi.

2.1.12 Blackbox Testing Menurut Lara Tastilia dalam jurnalnya, Pendekatan pengujian Black-Box merupakan metode pengujian yang menggunakan data tes berdasarkan persyaratan fungsional yang telah ditetapkan, tanpa mempertimbangkan struktur internal dari program yang diuji. **49** Fokus utama dari pengujian ini adalah pada fungsi perangkat lunak, sehingga sering disebut juga sebagai pengujian fungsional. Metode ini menekankan pada pengujian fungsi dengan memeriksa masukan dan keluaran data. (Tastilia et al., 2022) Blackbox Testing merupakan metode pengujian perangkat lunak yang berkonsentrasi pada pengujian fungsionalitas sistem tanpa memperhatikan struktur internal maupun

detail implementasi kode. **13** Dalam pendekatan ini, penguji hanya mengetahui masukan (input) dan keluaran (output) yang diharapkan, tetapi tidak memahami bagaimana proses internal terjadi. Blackbox Testing adalah metode pengujian perangkat lunak yang bertujuan untuk memastikan kebenaran atau keakuratan apakah sistem A-27 berfungsi sesuai dengan spesifikasi yang telah ditentukan. **52** Metode ini sering disebut sebagai behavioral testing karena hanya mengevaluasi perilaku sistem berdasarkan spesifikasi fungsional. Dalam pengujian ini, penguji bertindak seperti pengguna akhir dengan memberikan input tertentu dan memeriksa output yang dihasilkan oleh sistem, tanpa perlu memahami logika program atau struktur kode. Tujuan Blackbox Testing

- o Validasi Fungsionalitas Memastikan bahwa seluruh fitur dan fungsi sistem berfungsi sesuai dengan kebutuhan atau spesifikasi yang telah ditetapkan.
- o Mendeteksi Kesalahan Mengidentifikasi bug atau cacat dalam fungsionalitas perangkat lunak, seperti kesalahan dalam pemrosesan data, validasi input, atau antarmuka pengguna.
- o Memastikan Kesesuaian Menjamin bahwa perangkat lunak memenuhi harapan pengguna akhir berdasarkan kebutuhan yang telah ditentukan.

Karakteristik Blackbox Testing

- o Berbasis Fungsionalitas Pengujian hanya difokuskan pada fungsionalitas sistem, tanpa mempertimbangkan struktur internalnya.
- o Tidak Bergantung pada Kode Penguji tidak perlu mengetahui bahasa pemrograman atau implementasi kode yang digunakan.
- o Berbasis Spesifikasi Pengujian dilakukan berdasarkan spesifikasi atau dokumen kebutuhan perangkat lunak.
- o Dilakukan dari Perspektif Pengguna Pengujian dilakukan seperti pengguna akhir yang hanya melihat hasil tanpa memahami proses internal.

Teknik Blackbox Testing

Beberapa teknik yang digunakan dalam Blackbox Testing meliputi:

- o Equivalence Partitioning Membagi data masukan ke dalam beberapa kelas yang ekuivalen, di mana pengujian dilakukan hanya pada satu data dari setiap kelas. Contoh: Untuk masukan bilangan 1-100, penguji dapat memilih 50 sebagai representasi kelas data

valid. o Boundary Value Analysis Memfokuskan pengujian pada nilai-nilai batas, seperti nilai minimum, maksimum, dan nilai di sekitar batas. Contoh: Jika rentang masukan adalah 1-100, maka nilai 0, 1, 100, dan 101 diuji. o Decision Table Testing A-28 Menggunakan tabel keputusan untuk mengidentifikasi kombinasi masukan dan keluaran yang diharapkan. o State Transition Testing Memvalidasi transisi antara berbagai status sistem berdasarkan masukan tertentu. o Error Guessing Mengandalkan pengalaman penguji untuk menebak kemungkinan kesalahan dalam sistem. Keunggulan Blackbox Testing 1. Pendekatan Pengguna Akhir Pengujian ini meniru cara pengguna berinteraksi dengan sistem, sehingga membantu mengidentifikasi masalah yang relevan bagi pengguna. 2. Tidak Membutuhkan Pengetahuan Teknis Penguji tidak perlu memahami struktur kode atau bahasa pemrograman. 3. Cakupan yang Luas Blackbox Testing dapat digunakan untuk menguji berbagai jenis aplikasi, dari aplikasi desktop hingga berbasis web. 4. Efektif untuk Pengujian Fungsional Sangat berguna untuk memastikan bahwa fungsi sistem bekerja sesuai spesifikasi. Kekurangan Blackbox Testing 1. Terbatas pada Fungsionalitas Tidak dapat mengidentifikasi masalah pada struktur internal atau implementasi kode. 2. Kesulitan dalam Menentukan Kasus Uji Tanpa pemahaman tentang sistem internal, penguji mungkin melewatkan beberapa skenario penting. 3. Tidak Efisien untuk Sistem Kompleks Pengujian sistem besar dengan pendekatan ini dapat memakan waktu lebih lama tanpa pendekatan yang sistematis. 4. Tidak Menjamin Semua Cacat Terdeteksi Beberapa bug mungkin terlewat, terutama yang terkait dengan logika internal atau integrasi modul. Penerapan Blackbox Testing Blackbox Testing biasanya dilakukan dalam tahap-tahap berikut: 1. Unit Testing Untuk menguji komponen individu perangkat lunak. 2. Integration Testing Untuk memeriksa integrasi antar modul perangkat lunak. 3. System Testing Untuk memverifikasi sistem secara keseluruhan. 4. Acceptance Testing A-29 Untuk memastikan bahwa

perangkat lunak sesuai dengan kebutuhan pengguna akhir. Contoh Implementasi 1. Validasi Formulir Menguji apakah sistem menerima masukan data yang valid dan menolak masukan yang salah, seperti memeriksa email atau nomor telepon yang salah format. 2. Pengujian Aplikasi Web Memastikan halaman web memuat dengan benar berdasarkan masukan URL tertentu. 2.2 Tinjauan Studi 1. Jurnal dengan judul “Rancang Bangun Sistem Informasi Akademik Sekolah (SIAS) Berbasis Web (Informatika et al., 2021) Vol.4, No.2 Journal of Computer and Information Technology ditulis oleh Muhamad Solahudin pada Februari 2021 dari Teknik Informatika, Fakultas Teknologi Informasi & Komputer, Universitas Nusa Putra, Sukabumi. Penelitian ini bertujuan untuk merancang dan membangun Sistem Informasi Akademik (SIKAD) berbasis web untuk mengatasi masalah pengelolaan data akademik di MTs Ar-Rahmah Sukaraja, Sukabumi. Sebelum adanya sistem ini, pengelolaan data dilakukan secara manual menggunakan arsip dan selebaran, yang dianggap tidak efektif dan sulit dalam pencarian data. 22 53 Penelitian menggunakan metode Waterfall, yang dikenal dengan pendekatan sistematis dan berurutan. 10 22 Tahapan mencakup analisis kebutuhan, desain sistem, implementasi, pengujian, penerapan, serta pemeliharaan. Penelitian ini memanfaatkan alat bantu seperti Use Case Diagram, Activity Diagram, dan Class Diagram untuk menggambarkan desain system. Sistem informasi akademik ini dikembangkan menggunakan: o PHP: Digunakan sebagai bahasa pemrograman sisi server. o MySQL: Digunakan sebagai sistem manajemen basis data. o CSS: Untuk desain dan tata letak antarmuka. o Sublime Text: Sebagai editor teks dalam proses pengembangan Penelitian ini relevan sebagai referensi bagi pengembangan sistem informasi akademik di institusi pendidikan yang ingin beralih dari metode pengelolaan manual ke digital. Pendekatan berbasis web memberikan fleksibilitas dalam aksesibilitas dan pengelolaan data. 2. Jurnal dengan judul “Sistem Informasi Akademik Sekolah Menengah Kejuruan (SMK) Terpadu Ujung Batu Berbasis Web (Yuda et al.,

2024) Vol.10, No.2 Riau Journal of Computer Science ditulis oleh Fitra Yuda, Khairul Sabri, Dona, Retchi Puspita, dan M. Rasyid pada Juli 2024 dari Institut Teknologi Rokan Hilir Universitas Pasir Pengaraian. Penelitian ini bertujuan untuk mengatasi tantangan dalam pengelolaan data akademik di SMK Terpadu Ujung Batu yang sebelumnya dilakukan secara manual atau dengan aplikasi spreadsheet yang tidak terintegrasi. Hal ini menyebabkan data sulit dikelola dan sering terjadi kerangkapan, sehingga efisiensi dan akurasi pengelolaan data terganggu. Penelitian ini bertujuan untuk merancang A-30 dan mengembangkan Sistem Informasi Akademik Sekolah berbasis web menggunakan metode Waterfall. Sistem ini diharapkan dapat:

- o Mempermudah pengelolaan data siswa, nilai, absensi, dan laporan.
- o Meningkatkan efisiensi administrasi sekolah.
- o Memfasilitasi komunikasi antara siswa, guru, dan pihak sekolah

Metode Waterfall digunakan sebagai pendekatan pengembangan sistem, dengan tahapan meliputi:

- o Analisis kebutuhan.
- o Perancangan sistem.
- o Implementasi.
- o Pengujian.
- o Penerapan dan pemeliharaan.

Teknologi yang digunakan mencakup PHP untuk pemrograman sisi server, MySQL sebagai sistem basis data, serta integrasi fitur e-learning untuk mendukung pembelajaran Sistem Informasi Akademik berbasis web di SMK Terpadu Ujung Batu terbukti meningkatkan efisiensi pengelolaan data akademik dan administrasi. Dengan implementasi yang sukses, sistem ini memberikan kemudahan akses data dan mendukung transparansi serta komunikasi yang lebih baik antara pihak sekolah dan siswa. Namun, pengembangan berkelanjutan diperlukan untuk memastikan sistem tetap relevan dengan kebutuhan pendidikan yang berkembang

3. Jurnal dengan judul “Sistem Informasi Administrasi Sekolah Berbasis Web (Studi Kasus: SMK Negeri 16 Jakarta) (Fitriati, n.d.) Vol.1, No.2 JURNAL KONSTELASI: Konvergensi Teknologi dan Sistem Informasi ditulis oleh Nadiah, D Fitriati pada Desember 2021 dari Teknik Informatika, Universitas Pancasila, Jakarta. Penelitian ini dilakukan untuk mengatasi kendala

dalam pengelolaan administrasi di SMK Negeri 16 Jakarta, yang masih menggunakan metode konvensional. Beberapa masalah yang diidentifikasi antara lain: o Perekapan data kehadiran siswa dan guru yang tidak terpusat. o Pembuatan jadwal pelajaran secara manual menggunakan Excel, yang rawan kesalahan. o Kesulitan dalam memantau siswa berprestasi maupun bermasalah karena pencatatan data yang tidak terorganisir. Penelitian ini memiliki tujuan untuk mendesain sistem informasi administrasi berbasis web yang dapat: o Mempermudah pengelolaan data administrasi seperti kehadiran, jadwal, dan kelas siswa. o Memberikan akses mudah kepada admin, guru, dan siswa kapanpun dan dimanapun. o Mengintegrasikan data administrasi sekolah ke dalam satu sistem yang terpusat. Metode yang digunakan adalah SDLC Model Waterfall, dengan tahapan: o Analisa Kebutuhan: Mengumpulkan kebutuhan sistem melalui wawancara. o Pemodelan: Membuat pemodelan perangkat lunak berdasarkan kebutuhan. o Implementasi: Pengkodean sistem sesuai desain. A-31 o Pengujian: Menggunakan teknik Black Box Testing untuk memvalidasi fungsi sistem tanpa memeriksa kode program. o Pemeliharaan: Memastikan sistem tetap relevan dengan kebutuhan pengguna dan memperbaiki kesalahan yang mungkin terjadi. Sistem Informasi Administrasi Sekolah berbasis web yang dirancang untuk SMK Negeri 16 Jakarta berhasil meningkatkan efisiensi dan transparansi dalam pengelolaan administrasi. Sistem ini memberikan solusi terintegrasi yang relevan dengan kebutuhan pendidikan modern, terutama dalam masa pembelajaran jarak jauh. Namun, pengembangan berkelanjutan diperlukan untuk menjaga relevansi dan efektivitas sistem.

4. Jurnal dengan judul “PERANCANGAN SISTEM INFORMASI AKADEMIK SEKOLAH BERBASIS WEBSITE DENGAN LARAVEL 5 PADA SMK NEGERI 1 CILELES (Ruhawati et al., 2019) Vol.3, No.1 Jurnal Ilmiah Sains dan Teknologi ditulis oleh Irma Yunita Ruhawati, Siti Sopiah pada Februari 2019 dari Fakultas Ilmu Komputer Universitas Banten Jaya, Serang Banten. Penelitian ini dilakukan

REPORT #24485693

untuk mengatasi kendala dalam pengelolaan administrasi akademik di SMK Negeri 1 Cileles, yang masih dilakukan secara manual. Beberapa masalah utama yang ditemukan meliputi:

- o Pengelolaan absensi guru dan siswa yang masih menggunakan kertas, sehingga sulit melakukan pencarian data historis.
- o Administrasi sekolah, seperti persiapan UTS, UAS, dan pengumpulan data soal yang masih dikelola manual.
- o Ketidakterediaan fasilitas berbasis teknologi untuk mendukung kegiatan belajar mengajar sesuai Kurikulum 2013

Penelitian ini ditujukan untuk mengembangkan Sistem Informasi Akademik berbasis web dengan menggunakan Laravel 5. Sistem ini dirancang untuk:

- o Mengotomatisasi pengelolaan absensi siswa dan guru.
- o Meningkatkan efisiensi dalam administrasi sekolah.
- o Mendukung pembelajaran berbasis teknologi yang selaras dengan Kurikulum 2013.

Penelitian ini memanfaatkan metode Waterfall, yang terdiri dari tahapan:

- o Analisis Kebutuhan: Mengidentifikasi kebutuhan pengguna untuk pengelolaan akademik.
- o Perancangan: Membuat desain sistem, termasuk prototipe dan modul tampilan.
- o Implementasi: Mengembangkan sistem berdasarkan desain menggunakan Laravel 5.
- o Pengujian: Melakukan pengujian fungsionalitas dengan Black Box Testing.
- o Pemeliharaan: Mengantisipasi kebutuhan penyesuaian atau penambahan fitur di masa depan

Sistem Informasi Akademik berbasis Laravel 5 di SMK Negeri 1 Cileles memberikan solusi komprehensif untuk mengatasi kendala administrasi akademik. Dengan pendekatan sistematis melalui metode Waterfall, sistem ini berhasil meningkatkan efisiensi dan kualitas pengelolaan akademik di sekolah tersebut. Namun, pengembangan berkelanjutan diperlukan untuk memperluas fitur dan manfaat sistem.

A-32 5. Jurnal dengan judul “SISTEM INFORMASI ADMINISTRASI AKADEMIK UNTUK MENINGKATKAN PELAYANAN TERHADAP SISWA (STUDY KASUS: SMA PGRI KATIBUNG) (Tastilia et al., 2022) Vol.3, No.2 Jurnal Teknologi dan Sistem Informasi ditulis oleh Lara Tastilia, Dyah Ayu Megawaty, Ari Sulistiyawati pada Juni 2022 dari Sistem Informasi Fakultas

Teknik dan Ilmu Komputer Universitas Teknokrat Indonesia. Penelitian ini dilakukan untuk mengatasi masalah administrasi akademik di SMA PGRI Katibung, yang mencakup pengelolaan Rencana Kegiatan dan Anggaran Sekolah (RKAS) serta pembayaran administrasi siswa seperti uang pendaftaran dan SPP. Sebelumnya, pengelolaan data dilakukan secara manual menggunakan Microsoft Excel, yang menyebabkan:

- o Penumpukan file yang memperlambat pencarian data.
- o Kesulitan mencetak laporan secara berkala.
- o Risiko kehilangan data serta potensi kesalahan dalam penyusunan laporan

Penelitian ini memiliki tujuan untuk mendesain sistem informasi berbasis web yang dapat:

- o Mengelola administrasi pembayaran siswa secara terpusat dan aman.
- o Mempercepat pelayanan kepada siswa dan memudahkan pelaporan.
- o Meningkatkan efisiensi pengolahan data RKAS

Metode pengembangan yang digunakan adalah Extreme Programming (XP), yang memungkinkan pengembangan perangkat lunak secara cepat dan iteratif. Pengujian sistem dilakukan memakai Black Box Testing untuk memverifikasi bahwa sistem bekerja sesuai kebutuhan tanpa memeriksa struktur internal kode. Tahapan yang dilaksanakan meliputi:

- o Pengumpulan Data: Wawancara dan dokumentasi.
- o Perancangan Sistem: Menggunakan diagram UML, seperti Use Case, Activity, Class, dan ERD.
- o Implementasi: Menggunakan PHP dan MySQL sebagai teknologi utama.
- o Pengujian: Validasi fungsionalitas sistem melalui pengujian Black Box.

Sistem informasi administrasi berbasis web di SMA PGRI Katibung memberikan solusi komprehensif untuk mengatasi kendala administrasi akademik yang dihadapi sekolah. Dengan metode XP, sistem ini mampu meningkatkan efisiensi dan kualitas layanan administrasi. Pengembangan lanjutan diperlukan untuk memastikan sistem dapat terus relevan dan memenuhi kebutuhan pendidikan yang berkembang.

3.1 Objek Penelitian 3.1 1 Sejarah Perusahaan Jakarta Academics

didirikan oleh Mr. Sam Fulzele, seorang warga negara India, pada tahun 2013. Awalnya, Mr. Sam mengajar di sebuah lembaga pendidikan

di Jakarta untuk mempersiapkan siswa menghadapi ujian masuk perguruan tinggi di Inggris. Ia memiliki gagasan bahwa metode pembelajaran akan lebih efektif jika difokuskan pada individu atau kelompok kecil, dibandingkan dengan pendekatan kelas besar yang umum di sekolah tradisional. Konsep ini dikenal sebagai *personalized learning* dan menjadi landasan utama bagi Jakarta Academics. Setelah mengembangkan usahanya dari tahun 2013 melalui promosi dari mulut ke mulut, pada tahun 2017 Jakarta Academics resmi beroperasi di bawah PT Prima Bangsa Akademik, dengan kantor pusat di Plaza Semanggi, Jakarta Selatan. Sekolah ini menjadi pelopor *mikro school* di Indonesia, yang mengutamakan kelas kecil dengan maksimal delapan siswa untuk menciptakan pembelajaran yang lebih intensif dan personal.

3.1.2 Struktur Organisasi

Struktur organisasi Jakarta Academics dirancang secara hierarkis untuk mendukung operasional yang efisien. Posisi tertinggi dipegang oleh CEO, diikuti oleh direktur, eksekutif pemasaran, kepala sekolah, dan staf HRD sebagai jajaran manajerial utama. Selanjutnya, terdapat *head of subject* atau kepala mata pelajaran dan kepala divisi yang bertanggung jawab sebagai pengawas untuk masing-masing bidang studi atau departemen. Struktur ini dirancang untuk memastikan setiap divisi dapat menjalankan tugasnya secara terorganisir dan terarah.

3.1.3 Kegiatan Umum Perusahaan

Jakarta Academics merupakan lembaga pendidikan internasional yang melaksanakan berbagai kegiatan pembelajaran, mulai dari level dasar hingga tingkat atas. **37** Kegiatan belajar mengajar meliputi penyampaian materi, tugas harian, pekerjaan rumah, ujian bulanan, ujian tengah semester, hingga ujian akhir semester. Selain itu, Jakarta Academics juga berfungsi sebagai pusat ujian resmi untuk sertifikasi Pearson Edexcel, yang diadakan tiga kali setahun. Sertifikasi ini tidak hanya diikuti oleh siswa Jakarta Academics tetapi juga terbuka untuk umum, dengan nilai yang diakui secara internasional. Dengan berbagai aktivitas ini, Jakarta Academics

mendukung proses pendidikan yang berkualitas dan berstandar global

3.2 Metode penelitian Penulis merancang system administrasi pada Jakarta Acaemics ini memanfaatkan metode waterfall yang dimulai dari Planning, Requirement Analysis (Analisis Kebutuhan), System Design (Perancangan Sistem), Implementation (Implementasi). A-34 3.2.1 Planning

Pada tahap planning (perencanaan), penulis menentukan tujuan utama dari sistem administrasi yang akan dirancang untuk Jakarta Academics. Langkah ini melibatkan identifikasi masalah yang dihadapi dalam sistem administrasi saat ini serta kebutuhan utama yang harus dipenuhi oleh sistem baru. Selain itu, dilakukan analisis awal mengenai ruang lingkup sistem, termasuk penentuan batasan dan cakupan proyek.

14 Perencanaan ini mencakup penjadwalan waktu, alokasi sumber daya, serta estimasi biaya yang dibutuhkan untuk menyelesaikan proyek secara efisien.

Tahap ini bertujuan untuk menyediakan panduan yang jelas agar proyek berjalan terstruktur dan mencapai hasil sesuai dengan yang diharapkan. 3.2.2 Requirement Analysis (Analisis Kebutuhan) Pada tahap requirement analysis (analisis kebutuhan), penulis mengidentifikasi kebutuhan sistem yang akan dirancang, baik dari sisi pengguna maupun dari perspektif teknis. Proses ini dimulai dengan mengumpulkan informasi terkait masalah utama dalam sistem administrasi yang saat ini berjalan di Jakarta Academics. Data yang diperoleh digunakan untuk menentukan fitur-fitur utama yang diperlukan dalam sistem baru, seperti pengelolaan data siswa, jadwal pelajaran, dan laporan akademik. Kebutuhan diklasifikasikan ke dalam dua kategori utama: kebutuhan fungsional dan non-fungsional. Kebutuhan fungsional mencakup kemampuan sistem dalam memproses data administrasi dan akademik, seperti pendaftaran siswa, pengaturan jadwal, dan pelaporan. Sedangkan kebutuhan non-fungsional meliputi aspek keamanan data, kecepatan akses, dan kemudahan penggunaan oleh pengguna, seperti administrator, guru, siswa, dan orang tua. Analisis kebutuhan ini juga melibatkan pembuatan dokumen spesifikasi yang mendetail, yang

akan menjadi acuan dalam proses perancangan sistem berikutnya.

Langkah ini penting guna memastikan bahwa semua kebutuhan pengguna telah terakomodasi dan bahwa sistem yang dirancang mampu menyelesaikan permasalahan yang dihadapi oleh institusi secara efektif.

3.2.3 System Design (Perancangan Sistem) Pada tahap desain sistem, penulis mulai menerjemahkan hasil analisis kebutuhan yang telah dilakukan ke dalam bentuk rancangan teknis.

13 50 Desain sistem mencakup perencanaan arsitektur sistem, database, antarmuka pengguna, serta alur kerja sistem.

48 Langkah ini bertujuan untuk memberikan gambaran visual dan teknis tentang bagaimana sistem akan diimplementasikan secara menyeluruh.

1. Desain Arsitektur Sistem Sistem dirancang menggunakan arsitektur berbasis web dengan framework Laravel 11 sebagai backend dan MySQL sebagai sistem manajemen basis data.

Arsitektur ini mendukung pengelolaan data secara terintegrasi, memungkinkan pengguna untuk mengakses sistem melalui antarmuka berbasis web.

2. Desain Database Struktur database dirancang untuk menyimpan informasi penting seperti data siswa, data orang tua, jadwal pelajaran, dan laporan akademik. Setiap tabel diatur untuk memiliki hubungan yang jelas dan efisien guna mengurangi redundansi data. Penggunaan MySQL memberikan kemampuan untuk menangani transaksi database secara aman dan cepat.

3. Desain Antarmuka Pengguna (User Interface) Desain antarmuka dirancang dengan mempertimbangkan kemudahan penggunaan (user-friendly) untuk berbagai jenis pengguna, termasuk administrator, guru, siswa, dan orang tua.

Elemen-elemen seperti navigasi, form input, dan tampilan laporan dibuat agar intuitif dan mudah diakses.

4. Desain Alur Kerja Sistem

Setiap proses dalam sistem, seperti pendaftaran siswa, pembuatan jadwal, pengelolaan nilai, dan pelaporan, dijelaskan melalui diagram alir (flowchart) untuk memastikan alur kerja yang jelas dan terstruktur.

Ini membantu tim pengembang memahami setiap langkah yang diperlukan dalam implementasi.

5. Keamanan Sistem Desain sistem

juga mencakup aspek keamanan, seperti otentikasi dan otorisasi pengguna, serta perlindungan terhadap akses tidak sah. Hal ini bertujuan untuk menjaga kerahasiaan dan integritas data yang dikelola oleh sistem. Desain sistem yang detail ini menjadi panduan utama dalam tahap implementasi berikutnya, memastikan bahwa sistem yang dikembangkan sesuai dengan kebutuhan pengguna dan dapat mengatasi permasalahan yang telah diidentifikasi sebelumnya. 3.2.4 Implementation (Implementasi). Tahap implementasi merupakan proses penerapan desain sistem yang telah dirancang ke dalam bentuk kode program.

23 Pada tahap ini, sistem administrasi Jakarta Academics mulai dibangun dengan menggunakan framework Laravel 11 sebagai platform utama untuk pengembangan backend dan MySQL sebagai sistem manajemen basis data.

Proses implementasi dilakukan secara bertahap untuk memastikan setiap komponen sistem berfungsi dengan baik. 1. Pengembangan Backend Pada bagian backend, seluruh logika bisnis diterapkan menggunakan fitur-fitur Laravel, seperti routing, controller, dan model. Komponen ini bertanggung jawab untuk mengelola permintaan dari pengguna, memproses data, dan mengintegrasikan berbagai fungsi utama, seperti manajemen data siswa, jadwal pelajaran, dan pengelolaan laporan akademik. 2. Pengembangan Database A-36 Database MySQL dirancang dan diimplementasikan berdasarkan skema yang telah dibuat pada tahap desain. Tabel-tabel seperti siswa, orang tua, guru, dan nilai akademik dibuat dengan relasi yang terstruktur, untuk memastikan efisiensi dalam penyimpanan dan pengambilan data. Proses migrasi database juga dilakukan untuk mempermudah pengelolaan dan dokumentasi. 3. Pengembangan Frontend Antarmuka pengguna (frontend) dibangun menggunakan Blade templates dari Laravel, yang memanfaatkan kerangka kerja Tailwind CSS untuk menciptakan desain yang responsif dan modern. Setiap halaman dirancang agar mudah digunakan oleh berbagai jenis pengguna, seperti administrator, guru, siswa, dan orang tua. Fitur-fitur interaktif, seperti formulir

pendaftaran dan tampilan laporan, diimplementasikan dengan memperhatikan kemudahan akses. 4. Integrasi Sistem Komponen-komponen yang dikembangkan, seperti backend, frontend, dan database, diintegrasikan untuk membentuk sistem yang utuh. Proses integrasi dilakukan dengan menguji komunikasi antar modul untuk memastikan bahwa setiap fungsi berjalan sesuai dengan spesifikasi. 5. Pengujian Awal Sebelum sistem diluncurkan, dilakukan pengujian awal untuk memastikan bahwa setiap fitur yang diimplementasikan berfungsi dengan baik. Pengujian ini mencakup pengujian unit (unit testing) untuk memeriksa fungsi-fungsi individu, serta pengujian integrasi untuk memastikan bahwa modul-modul dapat bekerja secara bersamaan. Tahap implementasi ini bertujuan untuk mewujudkan desain sistem ke dalam bentuk nyata, memastikan bahwa semua fitur beroperasi dengan baik, dan memenuhi kebutuhan pengguna seperti yang telah dirumuskan pada tahap sebelumnya.

3.2.5 Maintenance

Tahapan maintenance atau pemeliharaan merupakan tahap krusial dalam siklus pengembangan perangkat lunak. Untuk memastikan bahwa sistem tetap berfungsi sesuai kebutuhan pengguna dan dapat beradaptasi dengan perubahan yang terjadi. Pada tahap ini, berbagai aktivitas dilakukan untuk menjaga, memperbaiki, dan meningkatkan performa sistem. Aktivitas dalam Maintenance:

- o Bug Fixing Mengidentifikasi dan memperbaiki kesalahan (bug) yang mungkin ditemukan setelah implementasi sistem dilakukan. Kesalahan ini dapat berupa kesalahan fungsional, kesalahan desain, atau kesalahan integrasi dengan sistem lain.
- o Adapting to Changes A-37 Penyesuaian sistem terhadap perubahan kebutuhan pengguna atau lingkungan, seperti perubahan dalam kebijakan administrasi sekolah, kurikulum, atau teknologi yang digunakan.
- o Optimization Melakukan optimasi untuk meningkatkan kinerja sistem, seperti mempercepat waktu respons halaman, memperbaiki struktur basis data, atau meningkatkan keamanan sistem.
- o Preventive Maintenance Menerapkan langkah-langkah pencegahan untuk menghindari potensi masalah di masa depan, seperti

memperbarui teknologi framework, melakukan backup data secara rutin, dan memonitor penggunaan sistem. o Enhancement Penambahan fitur baru atau peningkatan fitur yang ada berdasarkan kebutuhan pengguna yang muncul setelah sistem digunakan. Prosedur Maintenance: o Pelaporan Masalah: Pengguna melaporkan masalah atau kebutuhan melalui portal dukungan. o Analisis: Tim pengembang menganalisis laporan untuk menentukan prioritas dan langkah perbaikan. o Perbaikan atau Penyesuaian: Tim melakukan pengkodean ulang atau perubahan konfigurasi yang dibutuhkan. o Pengujian Ulang: Sistem diuji kembali untuk memastikan bahwa perbaikan tidak menyebabkan kerusakan baru. o Penerapan: Perubahan diterapkan ke sistem produksi dengan minimal gangguan terhadap pengguna. Dokumentasi Maintenance: Setiap aktivitas pemeliharaan dicatat dalam dokumentasi untuk memberikan riwayat perubahan dan sebagai referensi untuk pengembangan di masa mendatang. Tujuan Maintenance: o Memastikan sistem tetap relevan dengan kebutuhan sekolah. o Menjaga keandalan dan efisiensi operasional aplikasi. o Memberikan pengalaman pengguna yang lebih baik. o Mendukung pertumbuhan dan skalabilitas Sekolah Jakarta Academics. 3.3 Teknik pengumpulan data Dalam penelitian ini, data dikumpulkan melalui wawancara terstruktur dengan beberapa pihak kunci di Jakarta Academics, yaitu kepala sekolah, kepala divisi pemasaran (Head of Marketing), dan kepala divisi administrasi (Head of Administration). Wawancara ini bertujuan untuk memperoleh informasi mendalam terkait kebutuhan sistem, tantangan yang dihadapi dalam proses administrasi, serta harapan terhadap sistem administrasi yang akan dirancang. 1. Wawancara dengan Kepala Sekolah Kepala sekolah diwawancarai untuk memahami kebutuhan utama sistem dari perspektif operasional sekolah secara keseluruhan. Fokus wawancara meliputi A-38 pengelolaan data siswa, pengawasan kegiatan akademik, serta kebutuhan laporan yang tepat dan real-time guna untuk mendukung pengambilan keputusan. Tabel 3. 1 Sampel Pertanyaan untuk

Kepala Sekolah Apa yang harus dilakukan seorang calon siswa sebelum masuk dalam proses KBM? Ada berapa level yang ada di sekolah? Ada berapa mata pelajarannya yang diambil untuk masing-masing level? Bagaimana sertifikasi kelulusan untuk setiap level? 2. Wawancara dengan Head of Marketing Divisi pemasaran memberikan pandangan terkait kebutuhan sistem yang dapat membantu dalam mempromosikan sekolah dan menarik calon siswa baru. Informasi yang dikumpulkan mencakup pengelolaan data calon siswa, dan pelacakan data administrasi selama proses pendaftaran. Tabel 3. 2 Sampel Pertanyaan untuk Kepala Marketing Bagaimana cara untuk masuk / bergabung dengan sekolah ini? Bagaimana proses pendaftarannya? 35 Data apa saja yang harus diberikan kepada sekolah saat pendaftaran? Apa saja yang harus dilakukan oleh calon siswa? Dokumen apa saja yang harus disiapkan? Bagaimana proses pembayaran yang dilakukan oleh orang tua? 3. Wawancara dengan Head of Administration Kepala divisi administrasi diwawancarai untuk menggali informasi mengenai tantangan yang dihadapi dalam mengelola data siswa, dan orang tua. Fokus wawancara ini adalah untuk memastikan bahwa sistem yang dirancang mampu mengintegrasikan semua data administrasi dalam satu platform yang efisien. Tabel 3. 3 Sampel Pertanyaan untuk Kepala Administrasi Bagaimana proses KBM di sekolah ini? Bagaimana Guru melakukan presensi terhadap siswa? Bagaimana Guru memberikan materi, tugas, dan ujian ke pada siswa? Bagaimana Guru memberikan feedback terhadap tugas atau ujian kepada siswa? Bagaimana Guru melihat perkembangan hasil pembelajaran? Bagaimana siswa dan orang tua mengetahui perkembangan pembelajarannya? Hasil wawancara ini digunakan sebagai dasar untuk menyusun analisis kebutuhan sistem dan memastikan bahwa solusi yang dirancang mampu memenuhi kebutuhan spesifik dari setiap divisi yang terlibat. Teknik wawancara ini dipilih karena memberikan informasi langsung dan mendetail, yang sangat penting untuk keberhasilan desain dan implementasi sistem.

A-39 3.4 Analisis sistem berjalan Tahap analisis sistem berjalan dilakukan untuk memahami bagaimana sistem administrasi di Jakarta Academics beroperasi saat ini. Analisis ini mencakup evaluasi terhadap proses manual dan identifikasi kelemahan yang menjadi hambatan dalam pengelolaan data serta pelaksanaan tugas administrasi dan akademik.

3.4.1 Analisis dokumen

Pada analisis dokumen, penulis mengidentifikasi dan mengevaluasi dokumen-dokumen yang digunakan dalam sistem administrasi Jakarta Academics. Dokumen-dokumen ini meliputi:

1. Data Pendaftaran Siswa Dokumen ini mencakup formulir pendaftaran siswa baru, yang berisi informasi seperti data pribadi, data orang tua, dan dokumen pendukung lainnya. Proses pengisian dan pengarsipan masih dilakukan secara semi-manual dengan dokumen excel terpisah, sehingga berpotensi terjadi kesalahan pencatatan atau kehilangan data.
2. Laporan Akademik Siswa Laporan ini berisi hasil nilai siswa yang dicatat oleh guru pada setiap semester. Saat ini, laporan tersebut disimpan dalam format semi-manual atau dokumen Gdoc, Gsheet, ms. Office dan excel terpisah, yang menyulitkan pengelolaan dan penggabungan data untuk analisis lebih lanjut.
3. Jadwal Pelajaran Jadwal pelajaran disusun secara manual dan disampaikan dalam bentuk cetak atau file statis. Perubahan jadwal memerlukan waktu tambahan karena harus dilakukan secara manual dan didistribusikan ulang.

Analisis dokumen ini menunjukkan bahwa sistem administrasi yang berjalan saat ini memiliki banyak proses manual dan tidak terintegrasi. Hal ini mengakibatkan potensi kesalahan manusia, kesulitan dalam pengarsipan, serta efisiensi waktu yang rendah. Temuan ini menjadi dasar untuk merancang sistem baru yang lebih terintegrasi dan otomatis untuk mengatasi masalah-masalah tersebut. Langkah berikutnya adalah melakukan analisis terhadap proses berjalan untuk menggambarkan alur kerja sistem saat ini.

3.4.2 Analisis proses berjalan (flow diagram)

Pada tahap ini, analisis dilakukan untuk menggambarkan alur kerja sistem administrasi yang

sedang berjalan di Jakarta Academics menggunakan diagram alur (flow diagram). Diagram ini bertujuan untuk memberikan visualisasi yang jelas tentang proses-proses manual yang dilakukan saat ini, sehingga kelemahan dan hambatan dalam sistem dapat diidentifikasi dengan lebih mudah.

Proses Pendaftaran Siswa

1. Penerimaan Formulir Siswa baru mengisi formulir pendaftaran secara manual atau melalui file yang diunduh.
2. Verifikasi Dokumen A-40 Administrasi memeriksa kelengkapan dokumen yang diserahkan, seperti identitas siswa, data orang tua, dan dokumen pendukung lainnya.
3. Pencatatan Data Data siswa dicatat dalam dokumen spreadsheet atau buku catatan manual, tanpa integrasi ke sistem yang terpusat.
4. Pengarsipan Dokumen fisik disimpan dalam lemari arsip, sementara data digital disimpan dalam file terpisah.

Proses Pengelolaan Jadwal Pelajaran

1. Penyusunan Jadwal Kepala sekolah atau kepala bagian akademik menyusun jadwal pelajaran menggunakan aplikasi sederhana (seperti Microsoft Excel) atau secara manual.
2. Distribusi Jadwal Jadwal dicetak dan disebarakan kepada guru dan siswa melalui media fisik atau grup pesan.
3. Perubahan Jadwal Jika terjadi perubahan jadwal, proses revisi dilakukan secara manual, yang memakan waktu lama untuk disebarakan ulang.

Proses Laporan Akademik

1. Pengumpulan Nilai Guru menyerahkan data nilai siswa dalam bentuk dokumen manual atau file spreadsheet.
2. Pembuatan Laporan Administrasi mengumpulkan data dari berbagai guru untuk disusun menjadi laporan akhir semester.
3. Distribusi Laporan Laporan dicetak dan diberikan kepada siswa dan orang tua, tanpa ada akses digital yang mempermudah proses.

Flow Diagram Proses Berjalan Visualisasi proses di atas dapat digambarkan dengan flow diagram, yang memetakan tahapan-tahapan dalam setiap proses. Diagram ini menunjukkan bagaimana data bergerak melalui sistem manual saat ini, dan di mana potensi masalah muncul, seperti:

- o Duplikasi data akibat pencatatan manual.
- o Kesulitan akses informasi karena data tersimpan secara

terpisah. o Proses tidak efisien karena banyak pekerjaan dilakukan secara manual. Berikut alur proses pendaftaran siswa baru, mulai dari pengisian formulir hingga siswa resmi masuk kelas:

1. Calon Siswa Mendaftar Melalui Form di Website Calon siswa mengisi formulir pendaftaran yang tersedia di website sekolah sebagai langkah awal untuk mendaftar.
2. Jadwal Meeting A-41 Calon siswa dihubungi melalui WhatsApp atau email untuk mengatur jadwal meeting. Langkah ini dilakukan untuk mengonfirmasi dan melanjutkan proses pendaftaran.
3. Konfirmasi Meeting o Jika calon siswa tidak hadir pada jadwal meeting, maka:
 - Akan dihubungi ulang untuk menjadwalkan ulang (reschedule).
 - Jika sudah dihubungi ulang dan tidak ada respon, proses aplikasi dihentikan (application discontinue).o Jika meeting terlaksana, maka setelah semua informasi mengenai sekolah disampaikan ke pihak orang tua / wali siswa, maka dilanjutkan proses penjadwalan calon siswa untuk mengikuti placement test.
4. Placement Test o Calon siswa mengikuti tes penempatan. Hasil tes diberikan dalam waktu maksimal 3 hari kerja.
5. Keputusan Lanjutan o Jika proses tidak dilanjutkan (misalnya siswa gagal memenuhi kriteria, atau orang tua mengundurkan diri), pendaftaran dihentikan. o Jika proses dilanjutkan, siswa diarahkan untuk melakukan pembayaran.
6. Dokumen dan Pembayaran Setelah pembayaran selesai, dokumen-dokumen yang diperlukan diserahkan oleh calon siswa atau orang tua.
7. Masuk Kelas dan Mulai Pembelajaran Setelah semua proses selesai, siswa resmi masuk ke kelas dan memulai kegiatan pembelajaran. Berikut alur proses kegiatan belajar mengajar (KBM) yang dimulai dari pemberian tugas hingga pembagian laporan nilai siswa kepada orang tua atau wali siswa.

1. Guru Memberikan Materi/Tugas/Ujian Guru mengunggah materi pembelajaran, tugas, atau ujian ke platform Canvas (<https://k12.instructure.com/>) agar dapat diakses oleh siswa.
2. Siswa Mengerjakan dan Mengunggah Tugas ke Canvas Siswa mengerjakan tugas

yang diberikan dan mengunggah hasilnya kembali ke Canvas untuk dinilai. 3. Guru Memberikan Feedback dan Nilai Melalui Canvas Guru mengevaluasi tugas siswa, memberikan feedback, serta menilai pekerjaan siswa langsung melalui platform Canvas. 4. Guru Memindahkan Nilai ke GSheet di Google Drive Setelah selesai menilai, guru memindahkan nilai siswa ke dalam Google Sheet yang tersimpan di Google Drive untuk manajemen data lebih lanjut. 5. Guru Memberikan Komentar dan Progress di GDoc Guru membuat dokumentasi terkait komentar, nilai keaktifan siswa, dan progress pembelajaran di Google Doc yang juga tersimpan di Google Drive. 6. Homeroom/Wali Kelas Menggabungkan Data A-42 Homeroom atau wali kelas menggabungkan semua data siswa, termasuk nilai dan laporan komentar dari Google Sheet dan Google Doc, untuk menghasilkan file laporan (rapor). 7. Proses Validasi Hasil Rapor o Jika hasil disetujui oleh bagian administrasi, wakil kepala sekolah, atau kepala sekolah: Rapor dibagikan kepada orang tua/wali siswa. o Jika hasil belum disetujui: Dilakukan pengecekan ulang oleh Homeroom/Wali Kelas. Flow diagram analisis proses berjalan ini membantu mengidentifikasi area yang perlu ditingkatkan, seperti integrasi data, otomasi proses, dan pengurangan kesalahan manusia. Diagram ini juga menjadi acuan dalam mendesain sistem baru yang lebih terstruktur, terintegrasi, dan efisien.

3.5 Analisis kebutuhan

Setelah melakukan wawancara pada beberapa pihak terkait di sekolah Jakarta Academics, maka penulis melakukan analisis kebutuhan terhadap proses administrasi sekolah sebagai berikut.

Kebutuhan	Fungsional	Fungsional No
1	Harus ada halaman login untuk masing-masing role	2
2	Harus ada dashboard untuk masing-masing role	3
3	User marketing dapat melihat/mengubah status aplikasi pendaftaran	4
4	User akademik staff akademik dapat melihat siswa yang baru terdaftar	5
5	User akademik staff dapat mengelola tahun ajaran, semester, dan term	6
6	User akademik dapat mengelola mata Pelajaran, guru, dan ruang	

kelas 7 User guru dapat mengelola pemberian tugas, presensi dan pemberian feedback atau nilai kepada siswa 8 User guru dapat melihat jadwal kelas yang menjadi tanggung jawabnya 9 User siswa dapat melihat jadwal kelas dari mata Pelajaran yang diambilnya 10 User siswa dapat mengumpulkan tugas yang diberikan oleh guru 11 User orang tua dapat melihat data perkembangan akademik anaknya

Fungsional No Keterangan 1 Mudah untuk digunakan (user-friendly)
2 Memiliki antarmuka yang responsif A-43 A-44 BAB IV HASIL DAN ANALISIS PENELITIAN 4.1 Analisa Perancangan Sistem Pada tahap ini, dilakukan analisis untuk menghasilkan perancangan sistem yang mampu menjawab kebutuhan administrasi dan akademik Sekolah Jakarta Academics.

Hasil analisis ini menjadi dasar dalam pengembangan sistem yang terintegrasi berbasis web.

4.1.1 Hasil Analisis Data Hasil analisis data menunjukkan beberapa permasalahan utama yang perlu diatasi melalui sistem yang dirancang, di antaranya: o Pengelolaan Data Terpisah: Data siswa, guru, dan akademik yang sebelumnya dikelola secara manual dalam file terpisah menyulitkan akses dan integrasi informasi. o Efisiensi Operasional: Proses administrasi manual mengakibatkan pemborosan waktu dan sumber daya manusia yang dapat dialokasikan untuk kegiatan produktif lainnya. o Keakuratan Informasi: Tingginya potensi kesalahan dalam pengelolaan data manual menyebabkan kurangnya kepercayaan terhadap informasi yang digunakan dalam pengambilan keputusan. o Minimnya Sistem Pelaporan: Tidak adanya sistem pelaporan yang efisien menghambat proses pengambilan keputusan berbasis data.

4.1.2 Desain Sistem Berdasarkan hasil analisis, sistem yang dirancang memiliki karakteristik berikut: o Terintegrasi: Semua data administrasi sekolah dapat diakses dalam satu platform terpadu. Sistem mendukung integrasi data siswa, guru, orang tua. o Kemudahan Penggunaan: Antarmuka dirancang responsif dengan navigasi yang sederhana untuk memudahkan pengguna dengan tingkat kemampuan teknologi yang bervariasi. Portal khusus disediakan untuk

administrator, guru, siswa, dan orang tua. 4.1.3 Komponen Utama Sistem Manajemen Data Siswa: Fitur untuk mengelola data siswa, termasuk pendaftaran, profil, dan riwayat akademik. 1. Jadwal Pelajaran: Sistem otomatisasi untuk membuat dan mengatur jadwal pelajaran. 2. Pengelolaan Nilai: Fitur untuk memasukkan, mengelola, dan mempublikasikan nilai siswa. 3. Laporan Akademik: Sistem menyediakan laporan akademik yang dapat diakses oleh siswa dan orang tua secara real-time. 19 4. Keamanan Data: Fitur otentikasi dan otorisasi guna memastikan bahwa data hanya dapat diakses oleh pihak yang berwenang.

4.1.4 Teknologi Pendukung Dalam perancangan sistem ini, teknologi yang digunakan meliputi: A-45 o Framework Laravel 11: Untuk pengembangan aplikasi berbasis web yang handal. o MySQL: Untuk pengelolaan basis data relasional. o Tailwind CSS: Untuk tampilan yang responsif dan modern. o Blade Template: Untuk mempermudah pengelolaan antarmuka pengguna. 4.1.5 Diagram Perancangan Sistem ini dirancang menggunakan diagram sebagai berikut: o Use Case Diagram: Menampilkan hubungan antara pengguna dan sistem. o Activity Diagram: Menggambarkan alur proses utama dalam sistem. o Class Diagram: menunjukkan struktur kelas dalam sistem dan relasinya. o ERD (Entity Relationship Diagram): Memvisualisasikan relasi antar entitas dalam basis data. 1 68 4.2 Perancangan Diagram Sistem Usulan 4.2 1 Use Case Diagram Pihak yang terlibat terlibat dalam system web ini adalah calon siswa atau calon wali murid, staff marketing, staff akademik, guru, orang tua dan siswa. Berikut adalah Use Case dari bagian marketing: Berikut adalah Use Case dari bagian akademik: Berikut adalah Use Case dari bagian guru: Berikut adalah Use Case dari bagian orang tua: Berikut adalah Use Case dari bagian siswa: Dari gambar 4.1 hingga gambar 4.5 memperlihatkan masing-masing aktor yang akan menggunakan aplikasi web ini. 1 Berikut ini adalah penjelasan tugas yang akan dilakukan setiap aktor: 1. Calon siswa atau wali siswa dapat melakukan pendaftaran melalui web dengan

mengisi form pendaftaran. 2. Staff marketing dapat melihat aplikasi pendaftaran yang sudah masuk ke dalam system, lalu dapat melakukan pengelolaan seperti mengubah status aplikasi dimulai dari new, followed-up, placement-test, payment, Doc-submission, enrolled, discontinue, rejected . Dan juga dapat mengubah dan menghapus aplikasi. 3. Staf akademik dapat mengelola aplikasi yang sudah berstatus terdaftar, membuat akun orang tua dan siswa. Staff akademik juga dapat melakukan pengelolaan untuk tahun akademik, semester, dan term, serta akun orang tua, siswa, dan juga ruang kelas. Pembuatan jadwal, serta pengaturan guru mengajar kelas yang mana juga menjadi tanggung jawabnya. 4. Guru bertanggung jawab untuk mengajar kelas yang telah ditentukan oleh staff akademik, guru dapat melihat jadwal, mengambil presensi , A-46 memberikan tugas, memberikan feedback dan nilai pada siswa, serta mencatat aktivitas siswa di dalam kelas. 5. Orang tua dapat memantau perkembangan akademik anaknya, melihat presensi dan hasil nilai akademiknya, serta dapat berkomunikasi dengan pihak staff akademik untuk memberikan feedback atau complain kepada pihak sekolah. 6. Siswa adalah peserta didik yang menjalani proses pembelajaran di sekolah Jakarta Academics. Siswa dapat melihat jadwal kelasnya, mengumpulkan tugas yang diberikan guru, melihat hasil dari tugas yang dikerjakan, serta melihat presensinya. Berikut ini adalah deskripsi masing-masing pengguna yang akan menggunakan system ini:

Tabel 4. 1 Deskripsi Aktor dalam Use Case No. Pengguna Deskripsi

- 1 Calon siswa/Wali siswa Orang yang ingin bergabung ke Jakarta Academics
 - 2 Marketing Pihak yang mengelola pendaftaran siswa baru
 - 3 Staff Akademik Aktor yang bertugas mengelola administrasi sekolah
 - 4 Guru Guru mata Pelajaran di Jakarta Akademik yang mengajar kelas
 - 5 Orang tua / Wali murid Pihak yang bertanggung jawab terhadap pengawasan dan kebutuhan siswa
 - 6 Siswa Pihak yang terdaftar untuk menerima proses pembelajaran dari sekolah
- 4.2.2

Spesifikasi Use Case Berikut ini adalah spesifikasi dari use case yang ada pada bagian sebelumnya. Tabel 4. 2 Spesifikasi use case membuat aplikasi pendaftaran Nama Membuat Aplikasi Pendaftaran Aktor Calon siswa / Wali siswa. Trigger Calon siswa / Wali siswa ingin mendaftarkan siswa ke sekolah Jakarta Academics Preconditions Calon siswa / Wali siswa membuka form pendaftaran dalam website Jakarta Academics Post Conditions Data pendaftaran masuk dan disimpan ke dalam sistem web. Normal Flow User Action System Reaction 1. Aktor membuka form pendaftaran di web Jakarta 1. Sistem menampilkan form pendaftaran yang harus A-47 Academics. 2. Aktor klik tombol daftar, dan menunggu dihubungi oleh pihak marketing. diisi oleh pengguna. **66** 2. Sistem menyimpan data aplikasi ke dalam database. Alternative Flow – – Tabel 4. 3 Spesifikasi use case Login Nama Login Aktor Staff marketing, Staff akademik, Guru, Orang tua, Siswa. Trigger Pengguna ingin mengakses sistem informasi dalam aplikasi sesuai role nya. Preconditions Pengguna harus memiliki akun sesuai role yang dimilikinya. **1** Post Conditions Pengguna dapat mengakses ke sistem informasi sesuai dengan role yang dimilikinya. Normal Flow User Action System Reaction 3. Aktor memasukkan username / email / telepon dan password kemudian klik Login 4. Masuk ke halaman dashboard sesuai role yang dimiliki atau menerima pesan error dari input 3. Sistem melakukan cek kredensial pengguna 4. Jika kredensial benar maka akan diarahkan ke dashboard sesuai role dari akun tersebut, atau mengembalikan pesan error jika ada kesalahan input Alternative Flow 1. Jika lupa password maka pengguna dapat mereset password dengan klik Forgot Password / Lupa Kata Sandi 2. **18** Pengguna memasukkan email terdaftar 3. User akan menerima reset-link jika email terdaftar atau akan menerima pesan error jika email tidak ada. 1. Sistem akan mengarahkan ke halaman untuk input email 2. Sistem akan mengirim link- reset ke email yang terdaftar jika email ditemukan, dan mengembalikan pesan error jika

email tidak terdaftar. Tabel 4. 4 Spesifikasi use case logout

Nama Logout Aktor Staff marketing, Staff akademik, Guru, Orang tua, Siswa. A-48 Trigger Pengguna ingin keluar dari sistem informasi dalam aplikasi. Preconditions Pengguna sedang login di dalam sistem aplikasi Post Conditions Pengguna keluar dari sistem.

Scenario Normal Flow User Action System Reaction 1. Aktor mengklik tombol logout 1. Sistem menghapus semua data login pada database dan session

Alternative Flow – – Tabel 4. 5 Spesifikasi use c

ase reset password Nama Reset Password Aktor Staff marketing, Staff akademik, Guru, Orang tua, Siswa. Trigger Pengguna ingin gagal login dan mau mereset passwordnya Preconditions Pengguna mereset password melalui link-reset yang dikirimkan ke email Post Conditions Pengguna dapat login dengan password yang baru. Normal Flow User Action System Reaction 1. Aktor mengklik tombol Forgot

Password / Lupa Kata Sandi 2. **18** Aktor memasukkan email yang terdaftar 3.

Aktor akan mendapatkan email berisi link-reset atau pesan error jika email salah. 4. Aktor bisa mencoba login kembali jika reset password berhasil, atau menghubungi sekolah jika proses reset gagal.

1. Sistem mengarahkan ke halaman untuk memasukkan email terdaftar.

2. Sistem akan mengirim reset-link ke email yang terdaftar atau pesan error jika email tidak ditemukan. 3. Sistem akan mereset password melalui reset-link yang hanya berlaku selama 60 menit.

Sistem akan mengirim email kembali berisi info password reset.

Alternative Flow – – Tabel 4. 6 Spesifikasi use case kelola aplik

asi pendaftaran Nama Kelola Aplikasi Pendaftaran Aktor Staff marketing. Trigger Pengguna ingin mengelola aplikasi pendaftaran yang masuk Preconditions Pengguna harus sudah login ke dalam sistem web

Post Conditions Pengguna dapat mengelola data aplikasi, mengubah status, mengubah data, dan menghapus aplikasi. Normal Flow User

Action System Reaction 1. Aktor dapat melakukan 1. Sistem

menampilkan table A-49 pencarian aplikasi berdasarkan filter nama,

email, nomor telepon, atau tanggal pembuatan aplikasi. 2. Aktor mengklik tombol details sesuai dengan status aplikasi yang ditampilkan di dashboard 3. Aktor dapat melakukan klik tombol ekspor, edit, atau delete dari data yang ditampilkan 4. Aktor dapat mendownload file excel berisi semua data aplikasi. 63 5. Aktor dapat mengubah status aplikasi dan data aplikasi. 6. Aktor dapat menghapus data aplikasi yang ditampilkan. data sesuai dengan status aplikasi yang dipilih pengguna, atau melalui filter pencarian . Dilengkapi tombol ekspor, edit, dan delete. 2. Sistem akan melakukan sesuai tombol yang dipilih pengguna: o Menampilkan halaman aplikasi lengkap jika tombol edit dipilih o Mendownload file excel jika tombol ekspor dipilih. o Menghapus data setelah popup konfirmasi jika tombol delete dipilih Alternative Flow – – Tabel 4. 7 Spesifikasi use case melihat dan mendaftarkan aplikasi Nama Melihat dan Mendaftarkan Aplikasi Aktor Staff akademik. Trigger Pengguna ingin mengelola aplikasi pendaftaran yang masuk Preconditions Pengguna harus sudah login ke dalam sistem web Post Conditions Pengguna dapat mendaftarkan data aplikasi sebagai akun orang tua dan siswa. Normal Flow User Action System Reaction 1. Aktor melihat aplikasi yang telah selesai dan resmi terdaftar dengan klik tombol details pada dashboard. 2. Aktor mendaftarkan aplikasi ke dalam akun orang tua dan siswa. 1. Sistem menampilkan seluruh data aplikasi yang telah berstatus terdaftar di dashboard. 2. Sistem membuat akun pengguna dengan role orang tua dan siswa. Alternative Flow – – Ta bel 4. 8 Spesifikasi use case kelola akun orang tua dan siswa Nama Kelola Akun Orang tua dan Siswa A-50 Aktor Staff akademik. Trigger Pengguna ingin mengelola aplikasi akun orang tua dan siswa Preconditions Pengguna harus sudah login ke dalam sistem web Post Conditions Pengguna dapat mengubah data dan menghapus akun orang tua dan siswa Normal Flow User Action System Reaction 1. Aktor memilih dan klik menu daftar orang tua atau siswa di dashboard.

2. Aktor dapat melakukan ekspor, hapus, atau edit dari data table yang ditampilkan. 1. Sistem menampilkan daftar orang tua atau daftar siswa sesuai pilihan pengguna. 2. Sistem akan melakukan sesuai pilihan dari pengguna: o Menampilkan halaman berisi data lengkap akun yang dipilih oleh pengguna jika klik tombol edit. o Mendownload file excel berisi data akun yang dipilih pengguna jika klik tombol ekspor. o Menghapus data setelah popup konfirmasi jika tombol delete dipilih

Alternative Flow – – Tabel 4 . 9 Spesifikasi use case kelola tahun ajaran, semester, dan term

Nama Kelola Akun Tahun ajaran, semester, dan term Aktor Staff akademik. Trigger Pengguna ingin mengelola tahun ajaran, semester, dan term Preconditions Pengguna harus sudah login ke dalam sistem web Post Conditions Pengguna dapat membuat tahun ajaran baru, mengubah data tanggal mulainya tahun ajaran, semester, dan term, atau menghapus tahun ajaran baru. Normal Flow User Action System Reaction 1. Aktor memilih dan klik menu daftar orang tua atau siswa di dashboard. 2. Aktor dapat melakukan ekspor, hapus, atau edit dari data table yang ditampilkan. 1. Sistem menampilkan daftar orang tua atau daftar siswa sesuai pilihan pengguna. 2. Sistem akan melakukan sesuai pilihan dari pengguna: o Menampilkan halaman A-51 berisi data lengkap akun yang dipilih oleh pengguna jika klik tombol edit. o Mendownload file excel berisi data akun yang dipilih pengguna jika klik tombol ekspor. o Menghapus data setelah popup konfirmasi jika tombol delete dipilih

Alternative Flow – – Tabel 4. 10 Spesifikasi use case kelola guru, mata Pelajaran, dan ruang kelas

Nama Kelola Guru, Mata Pelajaran, dan Ruang Kelas Aktor Staff akademik. Trigger Pengguna ingin mengelola Guru, Mata Pelajaran, dan Ruang Kelas Preconditions Pengguna harus sudah login ke dalam sistem web Post Conditions Pengguna dapat membuat akun guru baru, mata Pelajaran dan ruang kelas. Dapat mengubah data guru, mata pelajaran, dan ruang kelas, dan juga dapat

menghapusnya. Normal Flow User Action System Reaction 1. Aktor memilih menu tahun ajaran, dan memilih data guru, mata Pelajaran, atau ruang kelas yang ditampilkan. 59 2. Aktor dapat mengubah data guru, mata pelajaran, dan ruang kelas atau menghapusnya. 1. Sistem menampilkan table daftar guru, mata Pelajaran, dan ruang kelas 2. Sistem akan melakukan sesuai pilihan dari pengguna: o Menampilkan data yang dapat diubah jika klik tombol edit. o Menghapus data setelah popup konfirmasi jika tombol delete dipilih Alternative Flow – – Tabel 4. 11 Spesifikasi use case kelola jadwal KBM Nama Kelola Jadwal KBM Aktor Staff akademik. Trigger Pengguna ingin mengelola jadwal kegiatan belajar mengajar (KBM) A-52 Preconditions Pengguna harus sudah login ke dalam sistem web Post Conditions Pengguna membuat jadwal KBM untuk 1 minggu di term yang dipilih Normal Flow User Action System Reaction 1. Aktor memilih menu jadwal yang ada di dashboard, kemudian pilih term yang akan dibuat jadwalnya. 2. Aktor mengecek ketersediaan jam guru atau ruang dengan klik tombol cek-ketersediaan/. 3. Aktor membuat ketersediaan jam guru atau ruang jika belum tersedia dengan klik tombol buat- ketersediaan. 4. Aktor dapat melihat ketersediaan spesifik guru atau ruang kelas dengan klik tombol lihat. 5. Aktor mengubah ketersediaan jam guru atau ruang dengan klik tombol simpan perubahan. 6. Aktor membuat jadwal sementara untuk term yang dipilih dengan klik tombol buat jadwal. 7. Aktor mengubah jadwal sementara yang dibuat sistem atau menyimpannya ke dalam sistem. 8. Aktor melihat atau mengubah jadwal yang telah ada di dalam database sistem. 9. Aktor dapat mengubah jadwal yang telah ada sebelumnya. 1. Sistem menampilkan halaman pengaturan jadwal. 2. Sistem akan melakukan pengecekan ketersediaan waktu untuk jam guru atau ruang kelas jika tombol cek diklik. 3. Sistem akan membuat ketersediaan waktu untuk jam guru dan ruang kelas jika tombol buat ketersediaan diklik. 4. Sistem akan menampilkan table berisi

waktu jam ketersediaan guru atau ruang kelas dalam 1 minggu. 5. Sistem akan menyimpan perubahan ketersediaan waktu untuk guru atau ruang kelas jika tombol simpan perubahan diklik. 6. Sistem akan menampilkan jadwal yang dibuat secara acak dan menampilkannya dalam bentuk table. 7. Sistem akan mengecek potensial bentrok antara kelas, guru, dan ruang kelas jika terjadi bentrok, dan mencegah penyimpanan data sebelum bentrok diubah. 8. Sistem akan menampilkan jadwal yang telah berhasil disimpan pengguna. 9. Sistem akan melakukan pengecekan bentrok untuk setiap perubahan yang dilakukan pengguna, dan mencegah penyimpanan A-53 sebelum bentrok diatasi.

Alternative Flow – – Tabel 4. 12 Spesifikasi use case melihat jadwal kelas Nama Melihat Jadwal kelas Aktor Guru Trigger Pengguna ingin melihat jadwal mengajar mingguan Preconditions Pengguna harus sudah login ke dalam sistem web Post Conditions Pengguna dapat melihat jadwal mengajar lengkap per minggu. Normal Flow User Action System Reaction 1. Aktor menekan link complete Schedule yang ada di dashboard. 1. Sistem akan mendownload file excel yang berisi jadwal lengkap satu minggu. Alternative Flow – – Tabel 4. 13 Spesifikasi use case kelola presensi siswa Nama Kelola Presensi Siswa Aktor Guru Trigger Pengguna ingin mengelola presensi dari kelas yang diajarkannya. Preconditions Pengguna harus sudah login ke dalam sistem web Post Conditions Pengguna dapat melihat /mengubah/menghapus data presensi siswa untuk setiap sesi yang telah dijalani pada term berjalan. Normal Flow User Action System Reaction 1. Aktor memilih kelas yang diajarnya di dashboard. 2. Aktor klik menu presensi pada halaman mata pelajaran. 3. Aktor mengubah/menghapus data presensi siswa dan klik tombol simpan. 1. Sistem akan mengarahkan ke halaman mata Pelajaran yang dipilih. 2. Sistem akan menampilkan data table berisi detail presensi siswa setiap sesi yang sudah berjalan pada term berjalan. 3. Sistem akan menyimpan perubahan data atau menghapusnya sesuai pilihan data

oleh aktor. Alternative Flow – – Tabel 4. 14 Spesifikasi use case kelola aktivitas siswa Nama Kelola Aktivitas Siswa Aktor Guru Trigger Pengguna ingin mengelola aktivitas siswa dari kelas yang A-54 diajarkannya. Preconditions Pengguna harus sudah login ke dalam sistem web Post Conditions Pengguna dapat melihat/mengubah/menghapus data aktivitas siswa untuk setiap sesi yang telah dijalani pada term berjalan. Normal Flow User Action System Reaction 1. Aktor memilih kelas yang diajarnya di dashboard. 2. Aktor klik menu aktivitas pada halaman mata pelajaran. 3. Aktor mengubah/menghapus data presensi siswa dan klik tombol simpan. 1. Sistem akan mengarahkan ke halaman mata Pelajaran yang dipilih. 2. Sistem akan menampilkan data table berisi detail aktivitas siswa setiap sesi yang sudah berjalan pada term berjalan. 3. Sistem akan menyimpan perubahan data atau menghapusnya sesuai pilihan data oleh aktor.

Alternative Flow – – Tabel 4. 15 Spesifikasi lihat detail akun anak dari orang tua Nama Lihat Detail Akun Anak/Siswa Aktor Orang tua Trigger Pengguna ingin melihat detail akun anak yang didaftarkan di sekolah. Preconditions Pengguna harus sudah login ke dalam sistem web Post Conditions Pengguna dapat melihat/mengubah data akun anaknya yang terdaftar. Normal Flow User Action System Reaction 1. Aktor memilih anak yang tampilkan di dashboard dengan klik tombol detail. 2. Aktor dapat mengubah data yang ditampilkan dan klik simpan perubahan. 1. Sistem akan mengarahkan ke halaman anak yang dipilih. 2. Sistem akan menyimpan perubahan yang dilakukan oleh aktor.

Alternative Flow – – Tabel 4. 16 Spesifikasi use case lihat data akademik anak dari orang tua Nama Lihat Data Akademik Anak/Siswa Aktor Orang tua Trigger Pengguna ingin melihat detail perkembangan data akademik dari anak yang didaftarkan di sekolah. Preconditions Pengguna harus sudah login ke dalam sistem web A-55 Post Conditions Pengguna dapat melihat data nilai dan perkembangan anaknya yang terdaftar. Normal Flow User Action

System Reaction 1. Aktor memilih anak yang ditampilkan di dashboard dengan klik tombol detail. 2. Aktor dapat melihat data akademik dengan klik tombol akademik. 1. Sistem akan mengarahkan ke halaman anak yang dipilih. 2. Sistem akan mengarahkan ke halaman akademik siswa yang dipilih oleh aktor. 3. Halaman yang ditampilkan juga sudah termasuk data presensi untuk setiap mapel yang diambil oleh siswa yang bersangkutan. Alternative Flow – – Tabel 4. 17 Spesifikasi use case pemberian feedback/komplain dari orang tua Nama Pemberian Feedback atau Komplain dari Orang Tua Aktor Orang tua Trigger Pengguna ingin mengajukan masukan atau complain pada pihak sekolah. Preconditions Pengguna harus sudah login ke dalam sistem web Post Conditions Pengguna dapat memberikan masukan atau complain pada pihak sekolah. Normal Flow User Action System Reaction 1. Aktor memilih menu Feedback/Complain yang ada di dashboard. 2. Aktor memasukkan ke dalam kolom masukan dan kemudian klik tombol kirim. 1. Sistem akan mengarahkan ke halaman untuk memasukkan data masukan/complain dari orang tua. 2. Sistem akan menyimpan data masukan ke dalam database. Alternative Flow – – Tabel 4. 18 Spesifikasi use case melihat data akademik oleh siswa Nama Melihat Data Akademik oleh Siswa Aktor Siswa Trigger Pengguna ingin melihat perkembangan dan hasil akademik untuk mata Pelajaran yang diambilnya. Preconditions Pengguna harus sudah login ke dalam sistem web Post Conditions Pengguna dapat melihat perkembangan akademiknya per mata A-56 pelajaran atau secara keseluruhan pada term yang berjalan. Normal Flow User Action System Reaction 1. Aktor memilih mata Pelajaran yang ada di dashboard. 2. Aktor memilih menu akademik, presensi, atau aktivitas kelas. 1. Sistem akan mengarahkan ke halaman detail mata pelajaran. 2. Sistem menampilkan table berisi data yang dipilih oleh actor, data akademik, presensi, atau aktivitas kelas sesuai mapel yang dipilih. Alternative Flow – – Tabel 4. 19 Spesifikasi use case chat sisw

a ke guru mapel Nama Chat Siswa ke Guru Mapel Aktor Siswa
Trigger Pengguna ingin bertanya atau diskusi pada guru mapel.
Preconditions Pengguna harus sudah login ke dalam sistem web Post
Conditions Pengguna dapat berdiskusi dengan guru mapel Normal Flow
User Action System Reaction 1. Aktor memilih mata Pelajaran yang
ada di dashboard. 2. Aktor memilih menu chat/diskusi pada
halaman yang ditampilkan. 3. Aktor memasukkan isi pertanyaan / diskus
i pada kolom input, atau menambahkan file yang dilampirkan. 1.
Sistem akan mengarahkan ke halaman detail mata pelajaran. 2.
Sistem menampilkan halaman untuk memasukkan data chat pada guru
mapel. 3. Sistem akan mengecek validitas file, dan menyimpan ke
database jika semua valid atau mengembalikan pesan error jika ada
kesalahan. Alternative Flow Aktor gagal memberikan file untuk
dilampirkan, dan mencobanya Kembali. Sistem akan melakukan pengecekan

ulang setiap ada file yang dilampirkan oleh aktor. A-57 4.2  3 Activity
Diagram Diagram aktivitas memvisualisasikan bagaimana proses sistem
berlangsung. Alur kontrol merupakan bagian dari sistem yang
menggambarkan berbagai pilihan keputusan yang dapat terjadi selama
aktivitas sistem berlangsung. Alur ini menunjukkan tahapan yang akan
dijalankan dalam sebuah use case, mulai dari titik awal hingga
titik akhirnya. Berikut ini adalah diagram aktivitas untuk desain aplikasi ini. 4.2

4 Sequence Diagram Diagram urutan umumnya digunakan untuk
menggambarkan hubungan antara berbagai objek dalam suatu sistem,
yang disusun berdasarkan alur waktu sesuai dengan peristiwa yang terjadi.
Diagram ini menunjukkan serangkaian kejadian yang dipicu oleh
aktor selama pengoperasian sistem. Biasanya, banyak diagram urutan
dibuat berdasarkan aktivitas yang terdapat dalam use case yang digunakan.  64

Berikut adalah sequence diagram dalam sistem aplikasi yang dibangun. 4.2

 5 Class Diagram Diagram kelas menggambarkan entitas serta
hubungan di antara mereka dalam sistem yang dirancang, memberikan
penjelasan terstruktur mengenai cara sistem tersebut dirancang. 4.3

Perancangan Antar Muka Pengguna Gambar di atas merupakan halaman login untuk akun staff marketing, staff akademik, guru, orang tua, dan siswa A-58 4.3.1 Rancangan Layout Marketing Halaman ini berisi aplikasi pendaftaran yang sudah disortir berdasarkan status aplikasi yang masuk. Jika di klik detail, maka akan menampilkan data table yang berisi status yang sama. Berikut ini adalah tampilan saat pengguna klik details dari salah satu status yang ada.

Sistem akan menampilkan aplikasi yang memiliki status sama. **56** Berikut ini

adalah tampilan konfirmasi untuk penghapusan data untuk mencegah penghapusan yang tidak disengaja. Berikut ini adalah tampilan saat

pengguna menekan tombol edit, system akan mengarahkan ke halaman yang menampilkan data lebih lengkap. Berikut ini adalah tampilan

untuk profil pengguna. 4.3.2 Rancangan Layout Staff Akademik Halaman

ini berisi pengaturan bidang akademik yang digunakan oleh staff

akademik sekolah. Gambar 4. 1 Halaman dashboard staff akademik

Berikut untuk tampilan pengaturan tahun akademik, semester, term,

guru, kelas, dan mata Pelajaran. Berikut untuk pengaturan mata

pelajaran kelas dan siswa yang mengambilnya. Berikut adalah halaman

pengaturan jadwal dan ketersediaan waktu jam untuk guru dan

pengunaan ruang kelas Berikut adalah tampilan jadwal sementara yang

dibuat oleh system berdasarkan kelas, mata pelajaran, dan

ketersediaan waktu untuk guru dan ruang kelas. A-59 Berikut adalah

tampilan halaman untuk data daftar orang tua, pengguna dapat mengeksport data atau menghapusnya dengan klik tombol yang sesuai.

Berikut adalah tampilan data orang tua yang lebih detail, disini

pengguna dapat mengubah, mengeksport, atau menghapus data. Berikut

adalah tampilan halaman untuk data daftar siswa, pengguna dapat

mengeksport data atau menghapusnya dengan klik tombol yang sesuai.

Berikut adalah tampilan data orang tua yang lebih detail, disini

pengguna dapat mengubah, mengeksport, atau menghapus data. 4.3.3

Rancangan Layout LMS Halaman ini berisi pengaturan untuk menunjang

proses kegiatan belajar mengajar yang digunakan oleh guru mata pelajaran, siswa, dan orang tua siswa. 4.4 Perancangan Implementasi Tahap implementasi merupakan langkah akhir dalam proses perancangan sistem. Pada tahap ini, seluruh rancangan yang telah dibuat sebelumnya disatukan dan digabungkan menjadi sebuah sistem informasi yang terpusat untuk mendukung administrasi sekolah secara menyeluruh. Penyederhanaan sistem terjadi pada bagian proses pelaporan hasil KBM, yang memotong proses manual menjadi otomatisasi oleh sistem.

4.5 Pengujian Untuk memastikan bahwa sistem yang diusulkan sesuai dengan kebutuhan dan harapan, dilakukan pengujian untuk mengevaluasi fungsionalitasnya, termasuk pengujian terhadap data input dan output yang diolah oleh sistem.

62 Berikut ini adalah hasil pengujian sistem administrasi sekolah. 62 Tabel 4. 20 Tabel Testing Bagian Marketing

No Nama Tes Langkah Hasil yang Hasil A-60 diharapkan 1

Pendaftaran 1) Buka form pendaftaran 2) Mengisi data 3) Klik tombol submit Sistem berhasil menyimpan aplikasi pendaftaran Sukses

2 Login marketing 1) Buka halaman login 2) Mengisi username / email / telepon dari akun marketing 3) Mengisi password 4) K

lik tombol sign in Berhasil login dan masuk ke dashboard

marketing Sukses 3 Ubah status atau data aplikasi 1) Login dan masuk dashboard 2) Klik tombol detail salah satu status yang ada

3) Pilih salah satu aplikasi 4) Ubah status aplikasi di menu

dropdown atau ubah data yang ditampilkan 5) Klik tombol simpan perubahan Status aplikasi atau pengubahan data berhasil diubah

Sukses 4 Transfer aplikasi terdaftar ke bagian akademik 1) Login dan masuk dashboard 2) Pilih data aplikasi dengan status enrolled

, klik tombol edit 3) Klik tombol Transfer to Administrations

Aplikasi berhasil dialihkan ke bagian Akademik Sukses Berikut ini

adalah hasil pengujian sistem bagian akademik: Tabel 4. 21 Tabel

Testing Bagian Akademik No Nama Tes Langkah Hasil yang diharapkan

Hasil 1 Login Staff akademik 1) Buka halaman login 2) Mengisi

username / email / telepon dari akun marketing 3) Mengisi password

4) Klik tombol sign in Berhasil login dan masuk ke dashboard

staff akademik Sukses 2 Daftarkan aplikasi 1) Login dan masuk dashboard 2) Pilih aplikasi atau cari dari menu New Applications 3) Pilih aplikasi yang ditampilkan 4) Klik Transfer Data Sistem akan membuat akun parents dan student Sukses 3 Membuat tahun ajaran baru 1) Login dan masuk dashboard 2) Pilih menu Academic year 3) Klik Add Academic Year 4) Isi start date dan end date 5) Klik submit Sistem membuat tahun ajaran , semester, dan term baru Sukses A-61 4 Edit tanggal tahun ajaran, semester, atau term 1) Login dan masuk dashboard 2) Pilih menu Academic year 3) Pilih tabel yang akan diubah tanggalnya, tahun ajaran, semester, atau term 4) Klik edit 5) Ubah tanggal mulai dan tanggal akhir 6) Klik submit Sistem akan menyimpan perubahan tanggal mulai atau tanggal akhir dari tabel tahun ajaran, semester, atau term Sukses 5 Hapus tahun ajaran 1) Login dan masuk dashboard 2) Pilih menu Academic year 3) Pilih tabel tahun ajaran. 4) Klik hapus 5) Masukkan password 6) Klik Yes Sistem akan menghapus tahun ajaran beserta semester dan term didalamnya Sukses 6 Tambah Guru 1) Login dan masuk dashboard 2) Pilih menu Academic year 3) Pilih tabel Teachers 4) Klik tombol Add Teacher 5) Isi formdata yang diperlukan 6) Klik save Sistem akan menambah data guru baru Sukses 7 Tambah Ruang Kelas 1) Login dan masuk dashboard 2) Pilih menu Academic year 3) Pilih tabel Classrooms 4) Klik tombol Add Classroom 5) Isi form data yang diperlukan 6) Klik save Sistem akan menambah data ruang kelas baru Sukses 8 Tambah Mata pelajaran 1) Login dan masuk dashboard 2) Pilih menu Academic year 3) Pilih tabel Subjects 4) Klik tombol Add Subject 5) Isi form data yang diperlukan 6) Klik save Sistem akan menambahkan data mata pelajaran baru Sukses 9 Tambah Levels/tingkat 1) Login dan masuk dashboard 2) Pilih menu

REPORT #24485693

Classes 3) Pilih tahun ajaran, semester, dan term 4) Tambah default level Sistem akan menambah level / Tingkat secara otomatis dari primary 1 hingga IAL Y2 Sukses 10 Masukkan siswa ke dalam data kelas 1) Login dan masuk dashboard 2) Pilih menu Classes 3) Pilih tahun ajaran, semester, dan term 4) Pilih mata pelajaran 5) Klik tombol add student Sistem akan menyimpan nama siswa sebagai murid kelas mata pelajaran yang Sukses A-62 6) Centrang murid yang akan dimasukkan ke dalam kelas 7) Klik assign student dipilih 11 Menghapus siswa dari kelas 1) Login dan masuk dashboard 2) Pilih menu Classes 3) Pilih tahun ajaran, semester, dan term 4) Pilih mata pelajaran 5) Klik tombol delete Sistem akan menghapus data siswa dalam maple tertentu sukses Tabel 4. 22 Tabel Testing Bagian Guru No Nama Tes Langkah Hasil yang diharapkan Hasil 1 Login guru 1) Buka halaman login 2) Mengisi username / email / telepon dari akun guru 3) Mengisi password 4) Klik tombol sign in Berhasil login dan masuk ke dashboard guru Sukses 2 Download jadwal lengkap guru 1) Login dan masuk dashboard 2) Klik tombol complete schedule Sistem akan mendownload file excel berupa jadwal lengkap guru yang login Sukses 3 Setting / ubah mata pelajaran 1) Login dan masuk dashboard 2) Pilih mata pelajaran di dashboard 3) Lakukan perubahan atau setting Sistem menyimpan perubahan yang dilakukan pengguna Sukses Tabel 4. 23 Tabel Testing Bagian siswa No Nama Tes Langkah Hasil yang diharapkan Hasil 1 Login siswa 1) Buka halaman login 2) Mengisi username / email / telepon dari akun siswa 3) Mengisi password 4) Klik tombol sign in Berhasil login dan masuk ke dashboard siswa Sukses 2 Download jadwal lengkap siswa 1) Login dan masuk dashboard 2) Klik tombol complete schedule Sistem akan mendownload file excel berupa jadwal lengkap siswa yang login Sukses A-63 A-64 BAB V KESIMPULAN DAN SARAN 5.1 Kesimpulan Penelitian ini telah berhasil merancang dan mengembangkan aplikasi

administrasi dan akademik sekolah berbasis web yang terintegrasi untuk Sekolah Jakarta Academics. Aplikasi ini dirancang untuk mengatasi berbagai tantangan yang dihadapi, seperti pengelolaan data yang tersebar, potensi kesalahan manusia, efisiensi waktu, serta keterbatasan sistem pelaporan. **38** Sistem yang dikembangkan menggunakan framework Laravel 11 dan MySQL sebagai teknologi utama, serta mengikuti metode pengembangan Waterfall. Aplikasi ini mencakup fitur-fitur utama seperti: 1. Manajemen Data Siswa: Mempermudah proses pendaftaran, pembaruan, dan pengelolaan data siswa. 2. Penjadwalan Pelajaran: Membantu guru dan administrator menyusun jadwal pelajaran secara otomatis dan efisien. 3. Pengelolaan Nilai dan Laporan Akademik: Memberikan akses real-time kepada siswa dan orang tua untuk memantau perkembangan akademik. 4. Portal Interaktif: Memungkinkan siswa, orang tua, guru, dan administrator untuk berinteraksi dalam satu platform. Penggunaan aplikasi ini diharapkan mampu meningkatkan efisiensi operasional, mengurangi kesalahan manusia, meningkatkan keamanan data, dan menyediakan sistem yang dapat mendukung pertumbuhan Sekolah Jakarta Academics di masa depan. 5.2

Saran Meskipun aplikasi ini telah memenuhi kebutuhan dasar administrasi dan akademik sekolah, ada beberapa hal yang perlu diperhatikan untuk pengembangan berikutnya. Saran berikut bertujuan untuk meningkatkan kualitas dan fungsionalitas aplikasi agar lebih optimal dalam mendukung kebutuhan sekolah: 1. Pengembangan Fitur Tambahan Aplikasi ini dapat ditingkatkan dengan melengkapi fitur-fitur tambahan yang relevan, seperti: o Modul E-Learning: Untuk mendukung pembelajaran jarak jauh dan kebutuhan pembelajaran digital yang semakin meningkat. o Sistem Perpustakaan Digital: Mempermudah akses dan pengelolaan koleksi buku serta bahan ajar. Penambahan fitur ini akan meningkatkan daya guna aplikasi dan memberikan pengalaman yang lebih lengkap bagi pengguna. 2. Uji Coba di Berbagai Lingkungan Sekolah Aplikasi yang telah dikembangkan perlu diuji coba

di berbagai jenis sekolah dengan kebutuhan berbeda untuk memastikan fleksibilitas dan skalabilitas sistem. Hal ini juga dapat memberikan wawasan baru dalam pengembangan fitur yang lebih universal. 3. Pemeliharaan dan Pembaruan Sistem A-65 Untuk menjaga kinerja aplikasi, diperlukan pemeliharaan rutin yang mencakup pembaruan fitur, perbaikan bug, serta peningkatan keamanan sistem. Hal ini penting untuk memastikan aplikasi tetap relevan dan dapat menghadapi tantangan teknologi yang terus berkembang. 4. Pelatihan Pengguna Agar aplikasi dapat dimanfaatkan secara maksimal, disarankan untuk memberikan pelatihan kepada seluruh pengguna, termasuk administrator, guru, dan staf lainnya. Dengan pelatihan ini, pengguna dapat memahami fungsi aplikasi secara menyeluruh dan memanfaatkan seluruh fiturnya dengan optimal. 5. Integrasi dengan Sistem Pihak Ketiga Aplikasi ini dapat dikembangkan lebih jauh dengan menggabungkan fitur pihak ketiga, seperti sistem pembayaran online atau layanan cloud. Dengan mengikuti saran-saran tersebut, diharapkan aplikasi ini tidak hanya menjadi solusi administrasi yang efektif tetapi juga mampu berkembang menjadi sistem yang inovatif dan relevan dengan kebutuhan pendidikan masa kini dan masa depan. A-66



REPORT #24485693

Results

Sources that matched your submitted document.

● IDENTICAL ● CHANGED TEXT

INTERNET SOURCE		
1.	0.63% eprints.upj.ac.id https://eprints.upj.ac.id/id/eprint/9194/11/11.%20BAB%20IV.pdf	● ●
INTERNET SOURCE		
2.	0.4% eprints.unipdu.ac.id http://eprints.unipdu.ac.id/3337/3/Bab%202.pdf	●
INTERNET SOURCE		
3.	0.37% raharja.ac.id https://raharja.ac.id/2023/11/13/21761/	●
INTERNET SOURCE		
4.	0.31% binus.ac.id https://binus.ac.id/bekasi/2024/11/perbedaan-sql-dan-nosql-serta-penggunaan...	●
INTERNET SOURCE		
5.	0.31% fmkelana.com https://fmkelana.com/laravel	●
INTERNET SOURCE		
6.	0.31% blog.myskill.id https://blog.myskill.id/istilah-dan-tutorial/mengenal-laravel-fitur-kelebihan-dan..	●
INTERNET SOURCE		
7.	0.29% mootaco https://mootaco.com/mengenal-perangkat-lunak-aplikasi/	●
INTERNET SOURCE		
8.	0.26% azuralabs.id https://azuralabs.id/blog-programming/4-rekomendasi-bahasa-pemrograman-u..	●
INTERNET SOURCE		
9.	0.26% blog.hoster.co.id http://blog.hoster.co.id/mengenal-pengertian-mysql-dan-fungsinya/	●



REPORT #24485693

INTERNET SOURCE		
10. 0.24%	ejurnal.itats.ac.id https://ejurnal.itats.ac.id/semantik/article/download/6232/4113	●
INTERNET SOURCE		
11. 0.23%	repository.upbatam.ac.id http://repository.upbatam.ac.id/6222/5/Bab%20II.pdf	●
INTERNET SOURCE		
12. 0.22%	www.toprankindonesia.com https://www.toprankindonesia.com/javascript-dan-fungsinya/	●
INTERNET SOURCE		
13. 0.22%	ejurnal.teknokrat.ac.id https://ejurnal.teknokrat.ac.id/index.php/teknoinfo/article/download/3386/1384	●
INTERNET SOURCE		
14. 0.21%	accounting.binus.ac.id https://accounting.binus.ac.id/2020/05/19/memahami-system-development-life...	●
INTERNET SOURCE		
15. 0.21%	nevacloud.com https://nevacloud.com/blog/dbms-adalah/	●
INTERNET SOURCE		
16. 0.2%	digi-lib.stekom.ac.id https://digi-lib.stekom.ac.id/assets/dokumen/ebook/feb_BcOFNdrpWQArml-eyU..	●
INTERNET SOURCE		
17. 0.19%	blog.myskill.id https://blog.myskill.id/istilah-dan-tutorial/panduan-lengkap-belajar-laravel-unt...	●
INTERNET SOURCE		
18. 0.19%	digilib.unila.ac.id http://digilib.unila.ac.id/81052/3/SKRIPSI%20FULL%20TANPA%20BAB%20PEMB..	● ●
INTERNET SOURCE		
19. 0.18%	www.syukra.org https://www.syukra.org/id/understanding-databases-and-their-various-importa...	●
INTERNET SOURCE		
20. 0.18%	citrahost.com https://citrahost.com/blog/frontend-dan-backend-penjelasan-serta-perbedaann..	●



REPORT #24485693

INTERNET SOURCE		
21.	0.17% www.academia.edu	●
	https://www.academia.edu/106202479/Analisis_Metode_Pengembangan_Sistem	
INTERNET SOURCE		
22.	0.17% ejournal.itn.ac.id	●
	https://ejournal.itn.ac.id/index.php/jati/article/download/9160/5405/	
INTERNET SOURCE		
23.	0.17% eprint.unipma.ac.id	●
	http://eprint.unipma.ac.id/2425/4/BAB%20II.pdf	
INTERNET SOURCE		
24.	0.16% www.liputan6.com	●
	https://www.liputan6.com/feeds/read/5869339/memahami-apa-itu-administras...	
INTERNET SOURCE		
25.	0.16% repository.uinjkt.ac.id	●
	https://repository.uinjkt.ac.id/dspace/bitstream/123456789/27386/1/YUSDIARDI...	
INTERNET SOURCE		
26.	0.15% it.telkomuniversity.ac.id	●
	https://it.telkomuniversity.ac.id/metode-waterfall-dalam-pengembangan-peran...	
INTERNET SOURCE		
27.	0.15% afila.co.id	●
	https://afila.co.id/blog/detail/Mempercepat-Pertumbuhan-Bisnis-dengan-Sistem..	
INTERNET SOURCE		
28.	0.15% toffeedev.com	●
	https://toffeedev.com/blog/website/front-end-adalah/	
INTERNET SOURCE		
29.	0.14% repository.unisbablitar.ac.id	●
	https://repository.unisbablitar.ac.id/id/eprint/952/4/BAB%20I%20PENDAHULUA...	
INTERNET SOURCE		
30.	0.13% codingstudio.id	●
	https://codingstudio.id/blog/basis-data-cek-pengertian-jenis-manfaat/	
INTERNET SOURCE		
31.	0.13% www.kompasiana.com	●
	https://www.kompasiana.com/zhenxxx1/6442419b08a8b52e4f2a4882/javascrip...	



REPORT #24485693

INTERNET SOURCE		
32.	0.13% www.merdeka.com	●
	https://www.merdeka.com/teknologi/mengenal-software-jenis-fungsi-dan-pera...	
INTERNET SOURCE		
33.	0.12% www.sipas.id	●
	https://www.sipas.id/blog/administrasi-adalah/	
INTERNET SOURCE		
34.	0.12% eskripsi.usm.ac.id	●
	https://eskripsi.usm.ac.id/files/skripsi/G21A/2018/G.211.18.0083/G.211.18.0083-...	
INTERNET SOURCE		
35.	0.11% www.gramedia.com	●
	https://www.gramedia.com/literasi/pengertian-administrasi/?srsltid=AfmBOopR..	
INTERNET SOURCE		
36.	0.11% hariono.site.unwaha.ac.id	●
	https://hariono.site.unwaha.ac.id/pengantar-basis-data/	
INTERNET SOURCE		
37.	0.11% eprints.upj.ac.id	●
	https://eprints.upj.ac.id/id/eprint/8578/26/BAB%20II.pdf	
INTERNET SOURCE		
38.	0.11% jurnal.mdp.ac.id	●
	https://jurnal.mdp.ac.id/index.php/jatisi/article/download/973/364/	
INTERNET SOURCE		
39.	0.11% www.domainesia.com	●
	https://www.domainesia.com/tips/cara-install-laravel-di-macos/	
INTERNET SOURCE		
40.	0.1% www.ferdichocs.my.id	●
	https://www.ferdichocs.my.id/2023/05/bagaimana-teknologi-memengaruhi-car...	
INTERNET SOURCE		
41.	0.1% osc.medcom.id	●
	https://osc.medcom.id/community/analisis-model-waterfall-pengertian-tahapa...	
INTERNET SOURCE		
42.	0.1% www.ranktracker.com	●
	https://www.ranktracker.com/id/blog/demystifying-web-app-development-a-co..	



REPORT #24485693

INTERNET SOURCE		
43.	0.1% www.exabytes.co.id	●
	https://www.exabytes.co.id/blog/apa-itu-laravel/	
INTERNET SOURCE		
44.	0.1% eskripsi.usm.ac.id	●
	https://eskripsi.usm.ac.id/files/skripsi/G21A/2018/G.211.18.0083/G.211.18.0083-...	
INTERNET SOURCE		
45.	0.1% jurnal.stkipppgritulungagung.ac.id	●
	https://jurnal.stkipppgritulungagung.ac.id/index.php/jipi/article/download/2870/...	
INTERNET SOURCE		
46.	0.09% www.sekawanmedia.co.id	●
	https://www.sekawanmedia.co.id/blog/metode-waterfall/	
INTERNET SOURCE		
47.	0.09% aws.amazon.com	●
	https://aws.amazon.com/id/what-is/sdlc/	
INTERNET SOURCE		
48.	0.09% www.lawencon.com	●
	https://www.lawencon.com/data-flow-diagram/	
INTERNET SOURCE		
49.	0.09% journal.aira.or.id	●
	https://journal.aira.or.id/index.php/cosmic/article/download/760/259/3034	
INTERNET SOURCE		
50.	0.09% journal.universitaspahlawan.ac.id	●
	https://journal.universitaspahlawan.ac.id/index.php/jrpp/article/download/1931..	
INTERNET SOURCE		
51.	0.09% www.liputan6.com	●
	https://www.liputan6.com/feeds/read/5789286/cara-membuat-formulir-pendaf...	
INTERNET SOURCE		
52.	0.09% ejurnal.teknokrat.ac.id	●
	https://ejurnal.teknokrat.ac.id/index.php/ictee/article/download/1012/578	
INTERNET SOURCE		
53.	0.08% majapahit.id	●
	https://majapahit.id/blog/2022/09/10/metode-waterfall/	



REPORT #24485693

INTERNET SOURCE		
54. 0.08%	blog.myskill.id <i>https://blog.myskill.id/istilah-dan-tutorial/memahami-middleware-dan-mvc-mo..</i>	●
INTERNET SOURCE		
55. 0.08%	www.kompasiana.com <i>https://www.kompasiana.com/atha02074/664f8642c925c40e9f525802/diagram-...</i>	●
INTERNET SOURCE		
56. 0.08%	cloud.google.com <i>https://cloud.google.com/sql/docs/mysql/delete-instance?hl=id</i>	●
INTERNET SOURCE		
57. 0.07%	maukode.com <i>https://maukode.com/blog/apa-itu-nosql-database</i>	●
INTERNET SOURCE		
58. 0.07%	www.kmtech.id <i>https://www.kmtech.id/post/mengenal-bahasa-pemrograman-html-css-dan-jav...</i>	●
INTERNET SOURCE		
59. 0.07%	journal.peradaban.ac.id <i>https://journal.peradaban.ac.id/index.php/jsitp/article/download/1172/780/2643</i>	●
INTERNET SOURCE		
60. 0.06%	id.scribd.com <i>https://id.scribd.com/document/617800185/Model-Waterfall</i>	●
INTERNET SOURCE		
61. 0.06%	journal.sinov.id <i>https://journal.sinov.id/index.php/juisik/article/download/517/461/1356</i>	●
INTERNET SOURCE		
62. 0.05%	media.neliti.com <i>https://media.neliti.com/media/publications/300974-rancang-bangun-sistem-in...</i>	●
INTERNET SOURCE		
63. 0.05%	e-journal.uajy.ac.id <i>https://e-journal.uajy.ac.id/12380/8/TF073216.pdf</i>	●
INTERNET SOURCE		
64. 0.05%	eprints.upj.ac.id <i>https://eprints.upj.ac.id/id/eprint/2314/11/11.%20BAB%20IV.pdf</i>	●



REPORT #24485693

INTERNET SOURCE		
65.	0.04% repository.unbara.ac.id https://repository.unbara.ac.id/2106/5/BAB%203.pdf	●
INTERNET SOURCE		
66.	0.04% epub.imandiri.id http://epub.imandiri.id/repository/docs/TaSkripsi/351743004%20Final.pdf	●
INTERNET SOURCE		
67.	0.03% repository.unhas.ac.id http://repository.unhas.ac.id/27566/2/H13115305_skripsi_24-05-2023%20bab%...	●
INTERNET SOURCE		
68.	0.03% eprints.upj.ac.id https://eprints.upj.ac.id/id/eprint/4255/10/10.%20BAB%20IV.pdf	●
INTERNET SOURCE		
69.	0.02% repository.nusamandiri.ac.id https://repository.nusamandiri.ac.id/repo/files/246257/download/11220551---A...	●