

## PENGEMBANGAN SISTEM INFORMASI AKADEMIK BERBASIS WEB DENGAN INTEGRASI WEB ENGINEERING DAN CONTINUOUS INTEGRATION

Adetya Maharani Marpaung<sup>1)</sup>, Suendri<sup>2)</sup>

<sup>12</sup>Sains dan Teknologi, Universitas Islam Negeri Sumatera Utara, Jl. Lap. Golf No.120, Kp. Tengah, Kec.  
Pancur Batu, Kabupaten Deli Serdang, Sumatera Utara 20353  
email: <sup>1</sup> [adetyamaharanimarpaung2002@gmail.com](mailto:adetyamaharanimarpaung2002@gmail.com) <sup>2</sup> [suendri@gmail.com](mailto:suendri@gmail.com)

### Abstract

*With the development of information technology, educational institutions must be more efficient in managing academic data. Management of academic data such as grade processing and attendance is still not optimal because the previous system has not been integrated. The Web Engineering method and the Continuous Integration (CI) method are used in this study to create a web-based academic information system. Web Engineering provides a structured framework for system design and development, and CI ensures that system integration and updates can be done automatically and stably. Needs analysis, system design, development, testing, and implementation are research methods. This system was developed using Codeigniter, with pipeline automation through GitHub Actions and containerization using Docker. The implementation results show that this system can improve the efficiency of academic administration, reduce input errors, and accelerate user access to information. Internal trial results can save time that previously took  $\pm 10$  minutes, now it only takes 2 minutes with an estimated efficiency of 30%. The system can remain stable and ready to be updated without downtime thanks to the use of CI. This research contributes to the development of educational information systems using a contemporary approach. This research provides a practical approach in developing a CI-based academic information system.*

**Keywords:** Academic Information Systems, Web Engineering, Continuous Integration, MTs Budi Agung Medan.

### Abstrak

Dengan perkembangan teknologi informasi, lembaga pendidikan harus lebih efisien dalam mengelola data akademik. Pengelolaan data akademik seperti pengolahan nilai, dan presensi masih belum optimal karena sistem sebelumnya belum terintegrasi. Metode Web Engineering dan metode Continuous Integration (CI) digunakan dalam penelitian ini untuk menciptakan sistem informasi akademik berbasis web. Web Engineering memberikan kerangka kerja yang terstruktur untuk desain dan pengembangan sistem, dan CI memastikan bahwa integrasi dan pembaruan sistem dapat dilakukan secara otomatis dan stabil. Analisis kebutuhan, perancangan sistem, pengembangan, pengujian, dan implementasi adalah metode penelitian. Sistem ini dikembangkan menggunakan Codeigniter, dengan otomatisasi pipeline melalui GitHub Actions dan containerisasi menggunakan Docker. Hasil implementasi menunjukkan bahwa sistem ini dapat meningkatkan efisiensi administrasi akademik, mengurangi kesalahan input, dan mempercepat akses pengguna ke informasi. Hasil coba internal dapat menghemat waktu yang sebelumnya memakan waktu  $\pm 10$  menit, kini hanya membutuhkan waktu 2 menit dengan estimasi efisiensi sebesar 30%. Sistem dapat tetap stabil dan siap di-update tanpa downtime berkat penggunaan CI. Penelitian ini berkontribusi pada pengembangan sistem informasi pendidikan dengan menggunakan pendekatan kontemporer. Penelitian ini memberikan pendekatan praktis dalam pengembangan sistem informasi akademik berbasis CI.

**Kata Kunci:** Sistem Informasi Akademik, Web Engineering, Continuous Integration, MTs Budi Agung Medan.

## 1. PENDAHULUAN

Dengan perkembangan pesat teknologi informasi, semakin banyak orang yang menggunakannya. Akibatnya, internet menjadi bagian penting dari teknologi informasi dan komunikasi dalam segala bidang termasuk pendidikan (Fauzi and Samsudin 2022). Salah satu contoh penerapan teknologi dalam bidang pendidikan yaitu sistem informasi akademik.

Sistem informasi akademik adalah sistem yang menyediakan layanan informasi berupa data yang berkaitan dengan pendidikan, seperti penyimpanan data siswa baru, penentuan kelas, penentuan jadwal pelajaran, pembagian wali kelas, dan proses penilaian. Sistem informasi akademik mengolah data dan melakukan kegiatan akademik seperti administrasi sekolah, penilaian, siswa, guru, dan atribut lainnya (Bela Damanik, et al 2024).

MTs (Madrasah Tsanawiyah) Budi Agung merupakan sekolah setingkat SMP yang berbasis islam dengan menggabungkan pendidikan agama dan akademik umum. MTs ini berlokasi di Jl. Platina Raya, Rengas Pulau, Medan Marelان. Sekolah ini berkomitmen untuk memberikan pendidikan berkualitas tinggi dengan menggabungkan nilai-nilai agama islam dengan kemajuan ilmu pengetahuan (Ulfa et al. 2024).

Meskipun MTs Budi Agung telah memiliki sistem informasi akademik, sistem informasi eksisting belum terintegrasi secara menyeluruh. Beberapa proses penting seperti pengolahan nilai, manajemen data dan pengelolaan keuangan masih berjalan secara terpisah, sehingga menyulitkan pengelolaan data secara efisien.

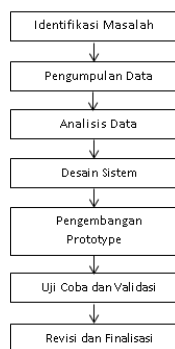
Selain itu, kendala pada jaringan internet yang tidak stabil sering mengganggu akses ke server EMIS, ditambah dengan format data yang berubah-ubah, semakin memperumit proses administrasi. Pencatatan pembayaran SPP juga masih dilakukan secara manual, sehingga rawan terjadi kesalahan. Akses terhadap informasi akademik pun masih terbatas, membuat siswa, guru, dan orang tua kesulitan memperoleh informasi secara real-time, kondisi ini menghambat komunikasi antar pihak terkait.

Untuk menjawab permasalahan tersebut, penelitian ini mengusulkan pengembangan sistem informasi akademik berbasis web dengan pendekatan Web Engineering dan Continuous Integration (CI). Web Engineering memberikan pendekatan sistematis dan analisis desain, dan implementasi sistem berbasis web yang efisien dan terstruktur (Oktaviani, et al 2022). Sementara itu, CI memungkinkan proses integrasi kode secara otomatis, mendukung pembaruan sistem tanpa downtime, serta menjamin stabilitas dan kontinuitas dalam pengembangan perangkat lunak (Danur Wijayanto, et al 2021).

Dengan menerapkan pendekatan ini, diharapkan sistem informasi akademik berbasis web yang dikembangkan dapat membantu MTs Budi Agung dalam memperlancar dan memperbaiki proses kegiatan akademik di MTs Budi Agung.

## 2. METODE PENELITIAN

Penelitian ini menggunakan jenis metode kualitatif, yaitu bertujuan agar dapat mendiskripsikan dengan jelas tentang “Pengembangan Sistem Informasi Akademik Dengan Integrasi *Web engineering* Dan Metode *Continuous integration* Di Mts Budi Agung Medan”.



Gambar 1. Tahapan Alur Penelitian

1. Identifikasi Masalah

Tahap pertama dalam penelitian ini adalah identifikasi masalah dengan tujuan untuk mengetahui dan memahami masalah yang akan diteliti (M. Audi Ikhsan and Samsudin 2024). Tahap ini dilakukan dengan melihat proses akademik di MTs Budi Agung Medan. Hasil observasi menunjukkan bahwa, meskipun sekolah memiliki sistem informasi akademik, sistem tersebut belum digunakan secara optimal. Hal ini menyebabkan pengelolaan data yang kurang efektif, pelaporan akademik yang tertunda sering, dan orang tua, guru, dan siswa belum memiliki akses ke informasi secara real-time. Hasil ini digunakan sebagai dasar untuk mengembangkan solusi

2. Pengumpulan Data

Pengumpulan data dikumpulkan melalui wawancara dengan operator sekolah untuk mengidentifikasi kendala yang ada, serta melalui studi literatur terkait penerapan web engineering dan continuous integration dalam sistem informasi akademik (Samsudin and Pramesti 2023). Wawancara dilakukan secara semi-terstruktur kepada satu orang operator sekolah di MTs Budi Agung Medan pada tanggal 15 Maret 2025, karena yang bersangkutan merupakan pengguna utama sistem akademik. Pertanyaan berjumlah 16 item terbuka yang disusun berdasarkan aspek kebutuhan fungsional sistem, kendala teknis, serta kesiapan infrastruktur sekolah terhadap penerapan metode Continuous Integration (CI). Proses wawancara dicatat secara manual tanpa rekaman suara, dengan pencatatan tertulis langsung saat sesi berlangsung. Pendekatan analisis data dilakukan secara tematik, yaitu dengan mengelompokkan temuan berdasarkan tema-tema umum seperti integrasi sistem, akses informasi, dan efisiensi kerja. Validasi dilakukan melalui klarifikasi langsung selama wawancara serta konfirmasi interpretasi setelah analisis untuk menjaga akurasi data (Samsudin 2023).

3. Analisis Data

Data yang dikumpulkan dievaluasi dan dirumuskan melalui analisis data (Syahrani et al. 2023). Analisis ini mencakup pengenalan kebutuhan pengguna, perbandingan dengan sistem akademik berbasis web yang telah ada, dan pemetaan fitur yang harus ada pada sistem informasi akademik yang akan dibangun. Faktor keamanan, skalabilitas, dan kemudahan penggunaan juga dipertimbangkan dalam analisis ini.

4. Desain Sistem

Model basis data, desain antarmuka pengguna, dan alur kerja sistem termasuk dalam desain arsitektur sistem. Untuk memastikan bahwa sistem memiliki struktur yang kuat, dapat disesuaikan, dan mudah dikembangkan, prinsip web engineering diterapkan dalam perancangan ini. Untuk memastikan pembaruan berkelanjutan, desain sistem juga mempertimbangkan integrasi dengan metode integrasi berkelanjutan (Siregar, et al 2023).

5. Pengembangan Prototype

Setelah itu, sistem mulai diimplementasikan dalam bentuk prototipe, yang mencakup fitur dasar yang telah dirancang sebelumnya untuk diuji di universitas. Pengembangan dilakukan secara bertahap dengan mempertimbangkan fitur fungsionalitas, performa, dan keamanan sistem.

6. Uji Coba dan Validasi

Setelah itu, sistem mulai diimplementasikan dalam bentuk prototipe, yang mencakup fitur dasar yang telah dirancang sebelumnya untuk diuji di universitas. Pengembangan dilakukan secara bertahap dengan mempertimbangkan fitur fungsionalitas, performa, dan keamanan sistem.

7. Revisi dan Finalisasi

Setelah uji coba dan validasi, tahap selanjutnya adalah Revisi dan Finalisasi. Perbaikan dilakukan berdasarkan masukan pengguna dan hasil pengujian, mencakup desain antarmuka, optimasi basis data, dan penyesuaian fitur. Tahap ini bertujuan agar sistem dapat berfungsi optimal dan memberi manfaat maksimal bagi pengguna.

Metode yang digunakan dalam pengembangan sistem adalah pendekatan Web Engineering dan penerapan Continuous Integration (CI). Web Engineering digunakan untuk merancang dan mengimplementasikan sistem secara terstruktur melalui tahapan analisis, desain, implementasi, dan evaluasi. Continuous Integration digunakan untuk memastikan proses integrasi kode yang dilakukan secara otomatis, cepat, dan stabil (Enda et al. 2022).

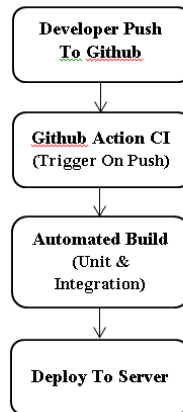
## 2.1 WEB ENGINEERING

Dalam pengembangan Sistem Informasi Akademik di MTs Budi Agung Medan, lima tahapan Web engineering diterapkan agar sistem dapat dikembangkan dengan baik dan sesuai kebutuhan (Yasin and Hamidy 2023). Berikut adalah penjelasannya:

1. **Communication**  
Tahap pertama adalah Communication, di mana dilakukan komunikasi intensif dengan pihak sekolah, guru, siswa, dan staf administrasi untuk memahami kendala yang ada dalam pengelolaan akademik. Proses ini mencakup observasi langsung, wawancara, serta analisis sistem yang sudah berjalan. Dari sini diperoleh gambaran kebutuhan sistem, seperti fitur manajemen siswa, jadwal, absensi, dan nilai yang harus disediakan dalam sistem baru.
2. **Planning**  
Setelah kebutuhan sistem dipahami, masuk ke tahap Planning. Pada tahap ini, disusun perencanaan teknis, termasuk pemilihan teknologi seperti Laravel untuk backend dan MySQL untuk database. Penggunaan Continuous Integration (CI) mulai diterapkan agar setiap perubahan kode dapat diuji otomatis sebelum diintegrasikan ke dalam sistem utama. Selain itu, perencanaan juga mencakup penyusunan timeline proyek, pembagian tugas tim, serta strategi pengujian dan deployment.
3. **Modelling**  
Selanjutnya adalah Modeling, yaitu tahap perancangan sistem secara lebih teknis. Diagram Use Case dibuat untuk menggambarkan bagaimana pengguna berinteraksi dengan sistem, sementara diagram Aktivitas menjelaskan alur kerja utama, seperti proses guru menginput nilai atau siswa melihat hasil akademik. Pada tahap ini juga dibuat desain database dan tampilan antarmuka (UI/UX) agar sistem lebih user-friendly dan sesuai kebutuhan pengguna.
4. **Construction**  
Tahap berikutnya adalah Construction, di mana sistem mulai dikembangkan berdasarkan desain yang telah dibuat. Backend dan frontend dibangun secara bertahap, serta database dihubungkan agar data akademik dapat dikelola dengan baik. Proses ini dilakukan dengan Continuous integration, sehingga setiap perubahan kode langsung diuji untuk memastikan stabilitas sistem. Beberapa fitur utama yang dikembangkan termasuk pendaftaran siswa, pengelolaan jadwal, absensi, serta input dan pemrosesan nilai oleh guru.
5. **Deployment**  
Terakhir, tahap Deployment, di mana sistem yang telah diuji mulai diterapkan di lingkungan sekolah. Setelah implementasi awal, dilakukan evaluasi berdasarkan umpan balik dari guru, siswa, dan staf administrasi. Jika ada perbaikan yang dibutuhkan, sistem terus disempurnakan agar lebih optimal. Dengan adanya sistem informasi akademik berbasis web ini, diharapkan pengelolaan akademik di MTs Budi Agung menjadi lebih efisien, transparan, dan mudah diakses oleh semua pihak yang berkepentingan.

## 2.2 CONTINUOUS INTEGRATION

Penerapan CI pada pengembangan ini dilakukan secara individu. Alur kerja CI pada penelitian ini menggunakan layanan GitHub Actions sebagai otomatisasi pipeline, serta Docker untuk proses build dan containerisasi aplikasi. Berikut adalah workflow CI yang diterapkan:



Gambar 2. Workflow CI

#### 1. Version Control

Tahap awal Continuous Integration adalah Version Control, di mana seluruh kode sistem informasi akademik dikelola dengan Git. Setiap pengembang rutin melakukan commit ke repositori untuk memudahkan pelacakan, pemulihan, dan kolaborasi. Dengan ini, fitur seperti manajemen siswa, absensi, jadwal, dan nilai dikelola lebih sistematis dan aman dari perubahan tak terkontrol.

#### 2. Automated Build

Tahap selanjutnya adalah Automated Build, yakni proses otomatis yang dijalankan setiap ada perubahan kode (Zulhakim and Kurniawan 2024). Pada Sistem Informasi Akademik MTs Budi Agung, build memastikan modul seperti pendaftaran, input nilai, dan laporan bebas error sebelum pengujian. Metode ini membantu mendeteksi kesalahan lebih cepat tanpa pemeriksaan manual.

#### 3. Automated Testing

Tahap berikutnya adalah Automated Testing, di mana setiap fitur diuji otomatis untuk memastikan bebas bug sebelum masuk ke sistem utama. Unit testing diterapkan pada validasi data dan absensi, sementara integration testing memastikan modul seperti input nilai dan laporan akademik berjalan tanpa konflik. Pengujian ini membuat sistem lebih stabil dan andal saat digunakan.

#### 4. Code Integration

Setelah pengujian berhasil, tahap selanjutnya adalah Code Integration, yaitu menggabungkan perubahan kode ke repositori utama. Pada proyek Sistem Informasi Akademik MTs Budi Agung, kode yang lolos pengujian otomatis langsung diintegrasikan tanpa mengganggu fungsionalitas yang ada. Ini memungkinkan pengembangan dilakukan secara bertahap tanpa menunggu seluruh fitur selesai.

#### 5. Deployment Automation

Terakhir adalah Deployment Automation, yaitu proses otomatis untuk menerapkan sistem ke server tanpa intervensi manual. Dengan metode ini, pembaruan seperti perbaikan nilai atau penambahan modul laporan dapat dilakukan cepat dan tanpa downtime. Continuous Integration memastikan sistem tetap stabil dan dapat diperbarui rutin tanpa risiko besar yang mengganggu pengelolaan data akademik di MTs Budi Agung Medan.

### 3. HASIL DAN PEMBAHASAN

#### 3.1 PENERAPAN WEB ENGINEERING

##### a. Tahap Komunikasi

Pada tahap komunikasi, dilakukan identifikasi kebutuhan pengguna sistem melalui interaksi langsung dengan pihak-pihak terkait, yaitu admin sekolah, guru, siswa, dan orang tua. Hasil

komunikasi menunjukkan bahwa admin sekolah membutuhkan sistem untuk mengelola data guru, agenda, serta berita atau pengumuman sekolah. Untuk menjawab kebutuhan tersebut, dirancang modul khusus yang memungkinkan admin melakukan tambah, edit, dan hapus data guru, agenda, serta pengumuman melalui dashboard yang mudah diakses.

Di sisi lain, guru membutuhkan fitur untuk mengatur jadwal pelajaran dan mengunggah materi pembelajaran dalam format digital. Oleh karena itu, sistem ini menyediakan fitur “Data Mapel” dan “Download” yang memudahkan guru dalam mengunggah jadwal serta materi pembelajaran secara mandiri.

Adapun siswa dan orang tua mengharapkan kemudahan akses terhadap informasi penting seperti jadwal, pengumuman, agenda, dan status pembayaran uang sekolah secara real-time tanpa harus datang langsung ke sekolah. Menanggapi hal ini, sistem dirancang memiliki halaman utama yang menyajikan informasi berupa berita, agenda, file pembelajaran, serta status pembayaran. Dashboard yang intuitif memastikan pengguna dapat mengakses semua informasi tersebut dengan cepat dan mudah, sehingga mendukung transparansi dan efektivitas komunikasi antara sekolah dan seluruh pemangku kepentingan.

### b. Tahap Perencanaan (Planning)

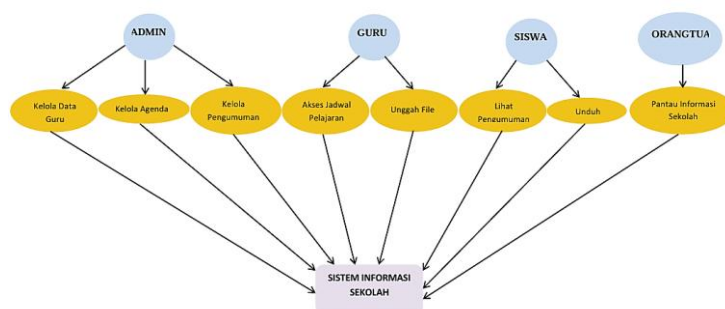
Tahap ini mencakup penyusunan jadwal dan alokasi tugas bagi para pengembang dalam pengembangan sistem informasi MTs Darul Ulum Budi Agung. Rencana pengembangan dimulai dengan pembuatan modul utama, yaitu Modul Data Guru yang dikembangkan oleh Backend Developer selama satu minggu dengan fungsi utama untuk mengelola data guru. Selanjutnya, Modul Agenda dan Pengumuman dikembangkan oleh tim Backend dan UI/UX selama satu minggu untuk mengelola jadwal kegiatan serta informasi pengumuman sekolah. Modul Mapel dan Galeri dikembangkan oleh Fullstack Developer dalam waktu dua minggu untuk menyajikan informasi terkait mata pelajaran serta galeri kegiatan sekolah.

Setelah pengembangan modul utama, dilakukan integrasi dashboard statistik oleh Frontend Developer selama satu minggu. Dashboard ini berfungsi untuk menampilkan data pengunjung dalam bentuk grafik yang informatif. Tahap akhir adalah pengujian sistem selama dua minggu oleh QA Tester, yang mencakup uji fungsionalitas, responsivitas antarmuka, serta aspek keamanan sistem.

Secara keseluruhan, proyek ini direncanakan berlangsung selama enam minggu dan dibagi ke dalam tiga sprint. Sprint pertama mencakup pengembangan backend untuk Modul Guru, Agenda, dan Pengumuman. Sprint kedua difokuskan pada pengembangan dashboard serta pengujian fungsional. Adapun Sprint ketiga mencakup pengujian responsivitas, keamanan sistem, serta penyusunan dokumentasi teknis.

### c. Pemodelan (Modelling)

Use case diagram adalah rangkaian atau uraian sekelompok yang saling terkait dan saling membentuk sistem secara teratur yang dilakukan atau diwakili oleh sebuah actor (Samsudin, et al 2021).

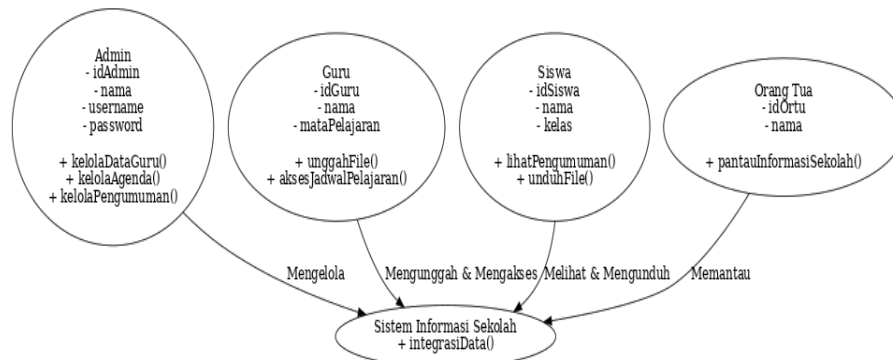


Gambar 3. Use case Diagram

Use Case Diagram sistem informasi sekolah menggambarkan interaksi antara Admin, Guru, Siswa, dan Orang Tua dengan sistem.



- Admin mengelola data guru, agenda, dan pengumuman, serta memastikan informasi selalu diperbarui.
  - 1. Guru mengakses jadwal pelajaran dan mengunggah materi pembelajaran.
  - 2. Siswa melihat pengumuman dan mengunduh file dari guru.
  - 3. Orang Tua memantau informasi sekolah dan kegiatan siswa.
- Sistem ini menjadi pusat interaksi antaraktor untuk mendukung komunikasi dan distribusi informasi yang terstruktur.

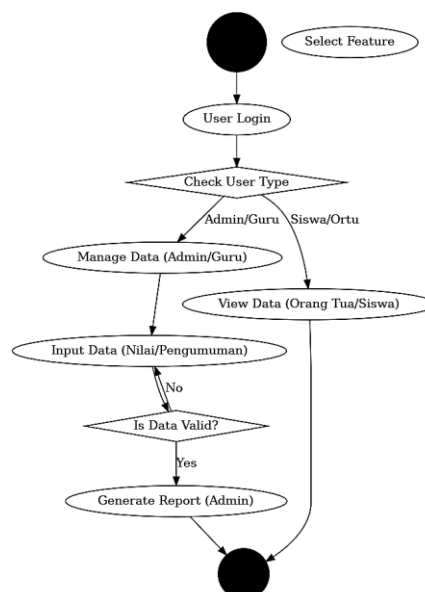


Gambar 4. Diagram Class

Diagram Class menggambarkan struktur dan deskripsi class, package dan object beserta hubungan satu sama lain dalam Sistem Informasi Sekolah(Aryati, et al 2022).

1. **Admin** memiliki atribut seperti idAdmin, nama, username, dan password, serta berfungsi mengelola data guru, agenda, dan pengumuman.
2. **Guru** dengan atribut idGuru, nama, dan mataPelajaran, dapat unggah materi dan akses jadwal.
3. **Siswa** memiliki idSiswa, nama, dan kelas, dengan fungsi lihat pengumuman dan unduh file.
4. **Orang Tua** dengan idOrtu dan nama, berfungsi memantau informasi sekolah.

Semua entitas terhubung ke kelas **Sistem Informasi Sekolah** melalui fungsi integrasiData(), memastikan sinkronisasi dan kelancaran komunikasi digital antar pengguna.



Gambar 5. Flowchart

Flowchart menggambarkan alur sistem informasi sekolah yang dimulai dari pemilihan fitur dan proses login. Setelah login, sistem memverifikasi jenis pengguna:

1. **Admin/Guru** dapat mengelola data (nilai, pengumuman), lalu data divalidasi. Jika valid, sistem menghasilkan laporan; jika tidak, diminta perbaikan.
  2. **Siswa/Orang Tua** hanya melihat data seperti nilai dan pengumuman.
- Proses berakhir dengan laporan yang dapat diakses admin. Alur ini memastikan setiap peran mendapat akses sesuai dan pengelolaan data berjalan efisien.

#### d. Kontruksi (construction)

Pada tahap konstruksi, sistem dikembangkan secara modular menggunakan CodeIgniter dan MySQL, dengan fitur utama yang sesuai alur proses pada diagram. Tahap konstruksi ini memastikan bahwa setiap modul dibuat dengan standar kode yang baik dan mengikuti alur proses yang efisien seperti yang ditunjukkan dalam diagram flow (Gustantia and Samsudin 2024). Ini juga mendukung sistem yang terintegrasi yang memudahkan pengelolaan data sekolah.

Tabel 2. Rincian Modul, Fitur, dan Fungsionalitas Sistem

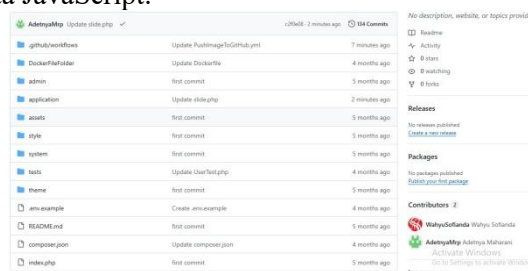
No	Modul/Fitur	Deskripsi
1	Manajemen Data Guru, Agenda, dan Pengumuman	
1.1	Manajemen Guru	Admin dapat melakukan CRUD (Create, Read, Update, Delete) data guru, termasuk informasi personal, mata pelajaran, dan jadwal kegiatan.
	Manajemen Agenda	Sistem memungkinkan admin atau guru untuk membuat dan mengelola agenda sekolah yang dapat dilihat oleh siswa dan orang tua.
1.2	Pengumuman	Fitur yang memudahkan pihak sekolah dalam menyampaikan informasi penting kepada seluruh pengguna sistem.
2	Dashboard Statistik Pengunjung	
2.1	Statistik Aktivitas Pengguna	Dashboard memberikan informasi visual tentang aktivitas pengguna yang login ke sistem (siswa, guru, orang tua).
2.2	Grafik Penggunaan Fitur	Informasi tambahan berupa grafik tren penggunaan fitur sistem, statistik input nilai, dan frekuensi akses pengumuman.
3	Sistem Galeri Dokumentasi Kegiatan Sekolah	
3.1	Pengelolaan Galeri	Admin atau guru dapat mengunggah foto dan video dokumentasi kegiatan sekolah.
3.2	Akses Galeri	Siswa dan orang tua dapat melihat dokumentasi visual dari berbagai acara dan kegiatan sekolah.
4	Alur Sistem Berdasarkan Diagram	
4.1	Login	Pengguna memulai dengan masuk ke sistem menggunakan akun masing-masing.
4.2	Verifikasi Jenis Pengguna	Sistem memeriksa apakah pengguna adalah admin/guru atau siswa/orang tua.
4.3	Admin/Guru	Admin atau guru dapat melakukan manajemen data guru, agenda, pengumuman, serta mengelola galeri dokumentasi.
4.4	Siswa/Orang Tua	Akses terbatas pada fitur melihat data, termasuk nilai, agenda, pengumuman, dan galeri kegiatan.



4.5	Validasi Data	Sistem memeriksa keabsahan data yang dimasukkan sebelum diterima. Jika valid, laporan akan dibuat; jika tidak, sistem meminta perbaikan data.
4.6	Laporan	Admin dapat melihat hasil laporan terkait nilai, agenda, dan aktivitas dalam sistem.

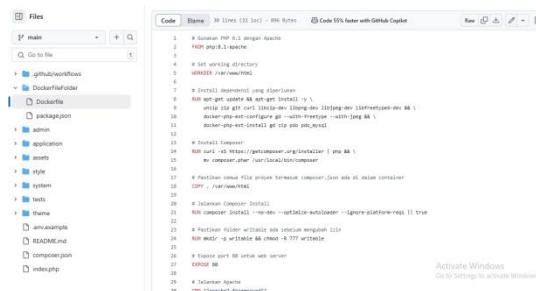
### 3.2 PENERAPAN CONTINUOUS INTEGRATION

Menyiapkan proyek CodeIgniter di GitHub dan menambahkan Dockerfile. Repositori ini memiliki sejumlah direktori penting seperti application, system, dan assets, yang menunjukkan bahwa proyek ini kemungkinan menggunakan framework CodeIgniter. File index.php, composer.json, dan README.md juga turut disertakan, memberikan gambaran awal tentang konfigurasi dan dokumentasi proyek. Aktivitas terakhir dilakukan beberapa menit yang lalu untuk memperbarui file PushImageToGithub.yml, menandakan adanya proses otomatisasi atau CI/CD. Kontributor utama dalam proyek ini adalah Adrestya Wahyuda Mitrya Sufrianda, dan proyek ini mayoritas menggunakan bahasa pemrograman PHP serta JavaScript.



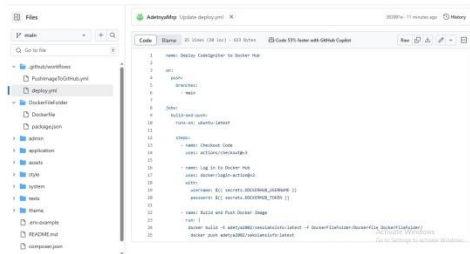
Gambar 6. Tampilan Repositori Proyek Aplikasi Web di GitHub

Konfigurasi Docker untuk menjalankan CodeIgniter. File ini mengatur direktori kerja, mengunduh dan menginstal Composer untuk manajemen dependensi PHP, serta memastikan file composer.json tersedia dalam container. Perintah RUN digunakan untuk instalasi Composer dan dependensi, termasuk ext-fileinfo, serta pengaturan environment variable. Setelah itu, file mengatur port 80 untuk server Apache dan menentukan perintah default untuk menjalankan server web. Dockerfile ini dirancang untuk mempermudah proses deployment dan pengujian aplikasi PHP dalam lingkungan



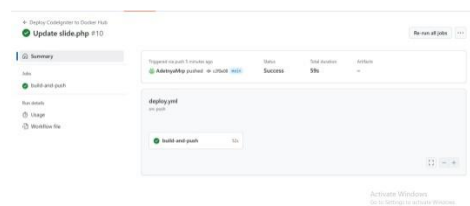
Gambar 7. Isi File Dockerfile

Menyusun workflow untuk build dan push image ke Docker Hub. File ini berfungsi untuk mengotomatiskan proses deploy container ke Docker Hub setiap kali ada perubahan pada branch main. Workflow ini menggunakan beberapa langkah penting, termasuk Checkout Kode, Login ke Docker Hub menggunakan kredensial rahasia (DOCKER\_USERNAME dan DOCKER\_PASSWORD), dan Build serta Push Docker Image ke repository Docker Hub yang telah ditentukan. File ini menunjukkan bahwa proyek telah mengimplementasikan praktik DevOps dengan memanfaatkan CI/CD (Continuous Integration and Deployment) secara efisien untuk menjaga konsistensi deployment dan mempercepat proses pengembangan aplikasi.



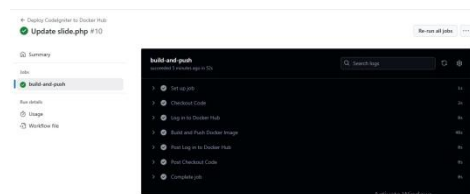
Gambar 8. Hasil Workflow GitHub Actions untuk Deploy Aplikasi CodeIgniter ke Docker Hub

Workflow ini dipicu oleh aksi push ke branch main, khususnya saat file slide.php diperbarui. Status eksekusi menunjukkan "Success" dengan durasi sekitar 58 detik, menandakan bahwa proses build dan push Docker image ke Docker Hub berjalan lancar tanpa kendala. Hal ini mencerminkan bahwa pipeline CI/CD telah dikonfigurasi dan berjalan dengan baik, mendukung otomatisasi deployment yang efisien dan andal untuk proyek berbasis CodeIgniter tersebut.



Gambar 9. Proses Otomatisasi Deploy Aplikasi Melalui GitHub Actions

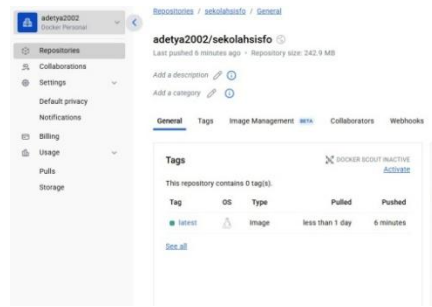
Workflow ini dijalankan secara otomatis saat terjadi push ke branch main, yang menunjukkan bahwa konfigurasi di dalam file deploy.yml menggunakan event on: push. Nama job yang dijalankan adalah build-and-push dan berhasil diselesaikan dalam waktu 1 menit 2 detik dengan status Success, yang berarti tidak ada error selama proses build dan push.



Gambar 10. Detail Proses Job build-and-push pada Workflow GitHub Actions

Pada gambar tersebut terlihat tahapan dalam workflow build-and-push yang berjalan di GitHub Actions untuk melakukan build dan push image Docker ke Docker Hub. Proses dimulai dengan Set up job yang berlangsung selama 1 detik, di mana GitHub Actions menyiapkan environment untuk menjalankan job, termasuk provisioning runner (virtual machine) dan memuat konfigurasi yang diperlukan. Setelah itu, dilanjutkan dengan Checkout Code yang memakan waktu 3 detik. Pada tahap ini, source code diambil dari repository menggunakan aksi actions/checkout@v2 sehingga kode terbaru dapat dibangun menjadi image Docker. Kemudian, dilakukan Log in to Docker Hub yang terjadi sangat cepat, yaitu 0 detik, karena autentikasi ke Docker Hub menggunakan username dan token yang disimpan sebagai secrets di GitHub. Langkah ini penting untuk memberikan izin pada proses push image Docker.

Tahap berikutnya adalah Build and Push Docker Image yang memakan waktu paling lama, yaitu 55 detik. Pada tahap ini, Dockerfile digunakan untuk membangun image Docker, dan setelah selesai, image tersebut didorong (push) ke Docker Hub dengan tag yang telah ditentukan, seperti latest. Setelah proses utama selesai, dilakukan Post Log in to Docker Hub yang bertujuan untuk membersihkan kredensial dari runner demi menjaga keamanan. Selanjutnya, ada Post Checkout Code yang membersihkan sisa-sisa file dari tahap checkout agar environment tetap bersih. Terakhir, job ditutup dengan Complete job yang menyelesaikan keseluruhan workflow dan melepaskan resource yang digunakan oleh runner. Secara keseluruhan, pipeline ini berjalan dengan sukses dalam waktu 1 menit 2 detik, menunjukkan bahwa proses build dan push telah selesai tanpa adanya error.



Gambar 11. tampilan dari Docker Hub Repository

Pada gambar tersebut terlihat halaman Docker Hub yang menunjukkan hasil dari proses build and push yang sebelumnya dilakukan melalui GitHub Actions. Pada tab General, terlihat bahwa repository Docker ini memiliki satu tag yaitu latest. Tag ini menunjukkan bahwa image yang di-push adalah versi terbaru dari aplikasi yang dibangun. Secara keseluruhan, gambar ini menegaskan bahwa proses push image Docker ke Docker Hub berhasil dilakukan dengan baik dan image dengan tag **latest** sekarang tersedia secara publik atau privat sesuai pengaturan repository Docker Hub yang digunakan.



Gambar 12. Tampilan Antarmuka

Pada Gambar Halaman Depan web MTs Darul Ulum Budi Agung. Di bagian atas, terdapat logo madrasah dengan nama "MTs Swasta Darul Ulum Budi Agung Medan." Menu navigasi utama mencakup beberapa opsi seperti "Home," "Profile," "Akademik," "Download File," dan "Berita."



Gambar 13. Tampilan Dashboard

Gambar ini menampilkan halaman dashboard dari **Sistem Informasi Sekolah MTs Budi Agung**. Pada panel navigasi di sebelah kiri, terdapat berbagai menu seperti *Dashboard*, *Data Guru*, *Data Agenda*, *Data Pengumuman*, *Data Download*, *Data Mapel*, *Data User*, *Berita*, *Album Galeri*, dan *Logout*. Menu ini memungkinkan administrator untuk mengelola berbagai informasi terkait sekolah. Di bagian utama halaman, terdapat grafik berjudul "**Statistik Pengunjung**" yang menunjukkan jumlah pengunjung dalam kurun waktu tertentu. Grafik ini tampaknya menampilkan data kunjungan yang stabil di angka 1 selama periode yang ditampilkan. Pada bagian bawah halaman, terdapat keterangan hak cipta dengan tulisan "**Copyright © Sistem Informasi Sekolah MTs Budi Agung**." Hal ini menunjukkan bahwa sistem ini merupakan platform internal sekolah yang digunakan untuk pengelolaan data dan informasi akademik secara digital.

### 3.3 PENGUJIAN SISTEM (UAT & PERFORMA)

Untuk memastikan bahwa sistem berjalan sesuai kebutuhan pengguna, dilakukan pengujian User Acceptance Test (UAT) terhadap beberapa fitur utama, seperti login, input nilai, absensi, dan cetak laporan. UAT dilakukan oleh operator sekolah dan hasilnya ditampilkan pada Tabel 3 berikut :

Tabel 3. Hasil UAT Sistem Informasi Akademik

No	Fitur yang diuji	Status	Tingkat kepuasan rata-rata
1	Login & Dashboard	Berhasil	4.8
2	Input Nilai	Berhasil	4.6
3	Absensi Siswa	Berhasil	4.5
4	Cetak Laporan Nilai	Berhasil	4.9
5	Jadwal Pelajaran	Berhasil	4.7
6	Manajemen Data Siswa	Berhasil	4.6
7	Pembayaran SPP/Tagihan	Berhasil	4.9

Setelah UAT, dilakukan simulasi pengujian performa untuk menilai seberapa cepat sistem merespons permintaan pengguna dalam kondisi akses bersamaan. Pengujian ini berfokus pada kecepatan respon sistem terhadap beberapa fitur inti. Hasilnya ditampilkan dalam Tabel 4

Tabel 4. Simulasi Pengujian Performa

No	Jumlah Akses Bersamaan	Fitur Diuji	Waktu Respon	Eror (Y/T)	Catatan
1	5	Login	1.2 Detik	Tidak	Stabil
	5	Input Nilai	1.5 Detik	Tidak	Responsif
3	10	Cetak Laporan	2.8 Detik	Tidak	Sedikit Lambat

Sebagai tolak ukur keberhasilan sistem, dilakukan perbandingan durasi proses akademik sebelum dan sesudah penerapan sistem baru. Hasil perbandingan ini menunjukkan efisiensi waktu yang signifikan pada berbagai aktivitas. Detailnya dapat dilihat pada Tabel 5.

Tabel 5. Perbandingan Waktu Proses Sebelum dan Sesudah Implementasi Sistem

Proses	Sebelum sistem (Kurang Terintegrasi)	Sesudah Sistem (Setelah Terintegrasi)
Input Nilai Siswa	7 menit	3 menit
Rekapitulasi Nilai	15 menit	3 menit
Cek Pembayaran SPP Siswa	6 menit	1 menit
Akses Data Guru	5 menit	2 menit

Hasil pengujian menunjukkan bahwa sistem mampu meningkatkan efisiensi kerja hingga 70 – 80 % dibandingkan proses sebelumnya. Sistem diuji selama 1 minggu dan tidak menunjukkan gangguan. Uji coba dilakukan oleh operator sekolah.

#### 4. KESIMPULAN

Penelitian ini menghasilkan sistem informasi akademik berbasis web yang berhasil menyelesaikan masalah administrasi dan manajemen data akademik di MTs Budi Agung Medan. Metode ini menggunakan pendekatan Web Engineering dan Continuous Integration (CI).

Kebaruan (novelty) dari penelitian ini terletak pada penerapan konsep Continuous Integration (CI) dalam pengembangan sistem informasi pendidikan di tingkat sekolah menengah. Penerapan CI dalam konteks ini tergolong langka, karena umumnya hanya diimplementasikan pada proyek-proyek berskala besar di industri teknologi. Dalam penelitian ini, CI digunakan secara mandiri dengan GitHub

Actions dan Docker untuk mengotomatisasi proses commit, build, testing, dan deployment. Pendekatan ini membuktikan bahwa CI tetap dapat diadaptasi dan memberikan manfaat nyata pada proyek pendidikan berskala kecil-menengah, seperti meningkatkan kestabilan sistem, mempercepat iterasi pengembangan, serta mengurangi risiko kesalahan saat pembaruan dilakukan.

Hasil pengujian menunjukkan bahwa sistem dapat meningkatkan efisiensi kerja secara signifikan. Jika dibandingkan dengan proses sebelumnya, efisiensi waktu rata-rata antara 70 dan 80 persen. Melalui uji coba terbatas, seperti simulasi beban sistem dan UAT, sistem menunjukkan kinerja yang stabil dan mendapatkan respons pengguna yang positif.

Pengujian performa sistem seperti load testing atau stress testing tidak dilakukan secara terpisah dalam penelitian ini. Fokus pengujian difokuskan pada validasi fungsional melalui metode User Acceptance Test (UAT) untuk memastikan fitur utama berjalan sesuai kebutuhan pengguna. Pengujian kestabilan dan performa sistem dalam kondisi beban tinggi menjadi batasan dalam studi ini dan direkomendasikan sebagai arah penelitian lanjutan guna memastikan kesiapan sistem untuk penggunaan skala besar atau multi-user secara simultan.

Penelitian ini tidak hanya memberikan kontribusi praktis berupa sistem yang dapat digunakan oleh sekolah, tetapi juga memberikan kontribusi metodologis dengan memperkenalkan model pengembangan sistem informasi akademik berbasis CI yang sederhana namun efektif, dan berpotensi untuk direplikasi pada lembaga pendidikan lainnya.

Sebagai rekomendasi untuk penelitian di masa depan, sistem sebaiknya diintegrasikan langsung dengan platform nasional seperti EMIS atau Dapodik, serta diuji lebih lanjut dalam kondisi beban tinggi secara simultan. Uji UX yang komprehensif dan pengembangan versi mobile juga disarankan guna meningkatkan kenyamanan dan fleksibilitas akses pengguna. Model pengembangan sistem berbasis CI ini terbukti efektif dan dapat direplikasi pada lembaga pendidikan lainnya.

## 5. DAFTAR PUSTAKA

- Aryati, Aryati, Samsudin Samsudin, and M Fakhriza. 2022. "Sistem Seleksi Penerimaan Tenaga Kerja Outsourcing Menggunakan Algoritma C5.0 Berbasis Android (Studi Kasus : Pt. Sinergi Indo Prima Medan)." *Rabit : Jurnal Teknologi dan Sistem Informasi Univrab* 7(1): 52–63.
- Bela Damanik, Raissa Amanda Putri, and Aninda Muliani Harahap. 2024. "Implementasi Metode Webqual 4.0 Dalam Mengevaluasi Sistem Informasi Akademik Uin Sumatera Utara." *JTIK (Jurnal Teknik Informatika Kaputama)* 8(1): 15–23.
- Danur Wijayanto, Arizona Firdonsyah, and Faisal Dharma Adhinata. 2021. "Implementasi Continuous Integration/Continuous Delivery Menggunakan Process Manager 2 (Studi Kasus: SIAKAD Akademi Keperawatan Bina Insan)." *Teknika* 10(3): 181–88.
- Enda, Depandi et al. 2022. "Penerapan Continuous Integration (CI) Pada Aplikasi Web Profil Karang Taruna (Studi Kasus : Karang Taruna Kabupaten Bengkalis) Depandi." *Seminar Nasional Industri dan Teknologi (SNIT), Politeknik Negeri Bengkalis* (November): 56–63.
- Fauzi, Muhammad Syafri, and Samsudin Samsudin. 2022. "Smart School Berbasis Web Interaktif Di SD Swasta Amaliyah Sunggal Dengan Algoritma K-Means Cluster." *Jurnal Sisfokom (Sistem Informasi dan Komputer)* 11(3): 332–41.
- Gustantia, Dwi, and Samsudin Samsudin. 2024. "Sistem Monitoring Perkembangan Siswa Di Sekolah Dasar Menggunakan Metode Peer Comparison Berbasis Website." *INOVTEK Polbeng - Seri Informatika* 9(1): 286–98.
- M. Audi Ikhsan, and Samsudin Samsudin. 2024. "Sistem Informasi Iklim Dan Kualitas Udara Pada Laboratorium Kalibrasi BMKG Berbasis Website." *SATESI: Jurnal Sains Teknologi dan Sistem Informasi* 4(2): 183–92.
- Oktaviani, Nia, Febriyanti Panjaitan, and Yomi Rosadi. 2022. "Sistem Informasi Akademik Pada Sekolah Menengah Atas Negeri 1 Tanjung Lubuk Menggunakan Metode Web Engineering." *Journal of Software Engineering Ampera* 3(1): 38–46.
- Samsudin, Samsudin, Indrawan Indrawan, and Sri Mulyati. 2021. "Perancangan Sistem Informasi Pembelajaran Algoritma Dan Pemrograman Berbasis Web Pada Program Studi Teknik Informatika STMIK ERESHA." *Jurnal Informatika Universitas Pamulang* 5(4): 521.
- Samsudin, Samsudin, and Adinda Ayu Mega Pramesti. 2023. "Rancang Bangun Sistem Informasi

- Manajemen Aset Pada BPJS Ketenagakerjaan Tanjung Morawa.” *Jutisi : Jurnal Ilmiah Teknik Informatika dan Sistem Informasi* 12(2): 456.
- Samsudin, Samsudin, and Yulisa Nanda Pratiwi. 2023. “Perancangan Sistem Informasi Pendaftaran Kerja Praktik Di PTTelkom Akses Medan Berbasis Mobile.” *Da'watuna: Journal of Communication and Islamic Broadcasting* 3(3): 1077–92.
- Siregar, Miftah R. Saadah, Samsudin, and Raissa Amanda Putri. 2023. “Sistem Informasi Geografis Dalam Monitoring Daerah Prioritas Penanganan Stunting Pada Anak Di Kota Medan.” *Journal of Science and Social Research* 6(3): 643–48.
- Syahrani, and Samsudin. 2023. “Pesantren Kabupaten Langkat Dan Binjai.” (April).
- Ulfa, Syarifah Widya et al. 2024. “Implementation Of Think Pair Share Learning Method In Class Viii Creed And Morals Lessons At Private Mts Darul Ulum Budi Agung Medan.” 4(02): 553–61.
- Yasin, Ikbil, and Fikri Hamidy. 2023. “Implementasi Sistem Informasi Data Kas Kecil Menggunakan Metode Web Engineering.” *CHAIN: Journal of Computer Technology, Computer Engineering, and Informatics* 1(1): 7–13.
- Zulhakim, Zaidan, and Ari Kurniawan. 2024. “Implementasi Continuous Integration Dan Continuous Deployment Pada Pengembangan Aplikasi Website Menggunakan Docker Dan Github Actions.” *Jurnal Manajemen Informatika* 16(1): 1–11.  
<https://ejournal.unesa.ac.id/index.php/jurnal-manajemen-informatika/article/view/58062>.