

Implementasi Dan Analisis Sistem Informasi Manajemen Data Mahasiswa Berbasis Web

¹Athaya Rima Hariyanto, ²Berliani Salsabilah

¹Sistem Informasi, Ilmu Komputer, Universitas Pamulang, Kota Tangerang Selatan, Indonesia

²Sistem Informasi, Ilmu Komputer, Universitas Pamulang, Kota Tangerang Selatan, Indonesia

¹berlianisalsabilah04@gmail.com, ²athaya.rima@gmail.com

Abstract

This research focuses on the development and implementation of a web-based Student Data Management Information System. The goal is to support more effective, organized, and accurate data management. The system was developed using the Waterfall method, which includes the stages of requirement analysis, design, implementation, testing, and maintenance. Black Box Testing was used during the testing phase to ensure that each system function operates as expected by the user, without examining the internal code. The test results show that all key features—such as login, data input, editing, deletion, search, and logout—function properly. It is expected that this system will speed up administrative services and minimize errors in managing student data.

Keywords: Information System, Waterfall, Black Box Testing, Student Data, Web-based

Abstrak

Penelitian ini berfokus pada pembuatan dan penerapan sebuah Sistem Informasi Manajemen Data Mahasiswa berbasis web. Tujuannya adalah untuk membantu pengelolaan data agar lebih efektif, teratur, dan akurat. Pengembangan sistem ini memakai metode Waterfall, meliputi tahapan analisis keperluan, perancangan, penerapan, pengujian, serta perawatan. Metode Black Box Testing digunakan saat pengujian untuk memastikan setiap fungsi sistem berjalan sesuai harapan pengguna, tanpa melihat kode internalnya. Hasil pengujian memperlihatkan bahwa seluruh fitur penting seperti login, input, pengeditan, penghapusan, pencarian, dan logout berfungsi dengan baik. Diharapkan sistem ini dapat mempercepat pelayanan administrasi dan meminimalisir kesalahan dalam pengelolaan data mahasiswa.

Kata Kunci: Sistem Informasi, Waterfall, Black Box Testing, Data Mahasiswa, Berbasis Web.

A. PENDAHULUAN

Perkembangan pesat teknologi informasi telah membawa dampak signifikan terhadap cara organisasi mengatur dan memproses data. Cara manual dalam menyampaikan informasi serta layanan kepada pengguna. Sejak awal pandemi COVID-19 hingga tahun 2025, pemanfaatan sistem digital di dunia pendidikan mengalami lonjakan yang cukup besar, menjadikannya bagian yang tak terpisahkan dari kegiatan operasional institusi pendidikan (Rizal dan Nuryasin, 2025).

Dalam ranah perguruan tinggi, pengelolaan data mahasiswa yang cepat dan akurat menjadi perhatian utama. Penggunaan sistem digital untuk mengatur informasi mahasiswa, seperti data pribadi, nilai akademik, status keaktifan, hingga riwayat pendidikan, telah menjadi praktik yang umum. Sistem ini berperan dalam meningkatkan kinerja staf administrasi, sekaligus memberikan akses informasi yang lebih cepat kepada dosen dan mahasiswa.

pencatatan yang dulu lazim digunakan, kini perlahan digantikan oleh sistem yang memanfaatkan komputer dan jaringan. Kesuksesan ini tak lepas dari peran sistem informasi berbasis web yang mampu menghadirkan efisiensi, efektivitas, dan akurasi dalam

Namun, tidak semua institusi pendidikan memiliki sistem informasi akademik yang mumpuni. Banyak sistem yang dibangun secara tergesa-gesa tanpa melalui proses pengujian yang komprehensif, sehingga seringkali menimbulkan masalah seperti kesalahan saat login, kegagalan penyimpanan data, tampilan antarmuka yang membingungkan, atau kinerja sistem yang lambat. Masalah-masalah ini dapat mengganggu kelancaran operasional dan mengurangi kepercayaan pengguna terhadap sistem (Wulandari et al., 2022).

Berdasarkan kondisi tersebut, penulis berinisiatif untuk merancang serta mengembangkan Sistem Informasi Manajemen Data Mahasiswa berbasis web. Sistem ini tidak hanya mengedepankan kemudahan penggunaan, tetapi juga menerapkan proses pengujian kualitas perangkat lunak sebelum sistem digunakan secara luas.

Pengujian dilakukan dengan metode *Black Box Testing*, yang memungkinkan evaluasi fungsi sistem dari sudut pandang pengguna tanpa melihat struktur internal perangkat lunak. Metode ini terbukti efektif dalam mendeteksi kesalahan fungsi dengan mengamati *input* dan *output* dari sistem (Fahrullah, 2021).

Proses pengujian merupakan tahapan penting dalam pengembangan sistem informasi, karena berfungsi untuk memastikan bahwa sistem dapat memenuhi kebutuhan pengguna dan tetap berfungsi dengan baik dalam berbagai kondisi. Metode *Black Box Testing* dipilih karena fokusnya pada fungsi keseluruhan sistem, sehingga cocok untuk mengevaluasi keandalan sistem dalam menangani berbagai skenario penggunaan (Hendartie et al. , 2023).

Sistem yang dibuat dalam penelitian ini memiliki beberapa fitur utama seperti *input* data mahasiswa, pencarian dan penyaringan data, pengelompokan berdasarkan program studi dan tahun angkatan, serta kemampuan untuk menghasilkan laporan dalam format *PDF* atau *Excel*. Dengan memanfaatkan teknologi *web*, sistem ini dirancang agar dapat diakses kapan saja dan di mana saja oleh pengguna, termasuk staf administrasi dan dosen.

Selain merancang sistem, penelitian ini juga bertujuan untuk menilai sejauh mana sistem tersebut dapat memenuhi kriteria *fungsionalitas*, kemudahan penggunaan, dan keandalan. Hasil dari pengujian yang diperoleh akan dianalisis untuk mengetahui fitur-fitur yang telah berfungsi dengan baik serta fitur yang memerlukan pengembangan lebih lanjut. Evaluasi ini menjadi landasan untuk memperbaiki sistem agar dapat digunakan secara berkelanjutan dan stabil.

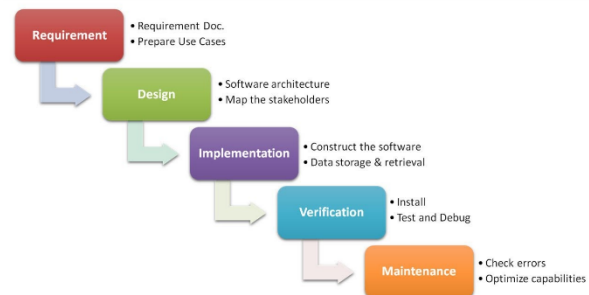
Dengan diterapkannya sistem informasi ini, diharapkan pengelolaan data mahasiswa di institusi pendidikan dapat berjalan lebih teratur, cepat, dan akurat. Selain itu, penelitian ini diharapkan dapat berkontribusi pada pengembangan sistem informasi pendidikan serta menjadi referensi dalam penerapan metode pengujian sistem yang sesuai dan efisien.

B. METODE

Model *Waterfall*, meski tergolong metode lawas dalam pengembangan perangkat lunak, tetap relevan, terutama untuk proyek dengan kebutuhan yang jelas sejak awal. Prosesnya terstruktur dan berurutan, mulai dari analisis kebutuhan, perancangan sistem, implementasi, pengujian, hingga pemeliharaan (Asrin dan Utami, 2023). Kelebihan utamanya adalah perencanaan dan pengendalian proyek yang lebih mudah, berkat dokumentasi yang rapi di setiap tahap (Aisyah dan Wahyudi, 2021).

Namun, *Waterfall* cenderung kurang fleksibel terhadap perubahan. Jika kebutuhan berubah di tengah jalan,

pengembang sering kali harus kembali ke tahap awal, yang bisa menunda proyek (Handayani dan Ningsih, 2022). Selain itu, pengguna jarang terlibat setelah fase awal, sehingga masukan baru biasanya muncul saat pengujian, yang bisa memicu masalah di akhir (Asrin dan Utami, 2023). Karena itu, metode ini sering dikombinasikan dengan model yang lebih adaptif seperti *Agile*.



Gambar 1. Tahapan dalam Metode Pengembangan Perangkat Lunak Waterfall

Metode *Waterfall* memiliki lima tahapan utama:

- 1. Penilaian Kebutuhan (*Requirement*)**
Tahap ini mengumpulkan data terkait kebutuhan sistem, baik fungsional maupun non-fungsional. Data ini kemudian dicatat dalam dokumen kebutuhan dan *use case* (Rahmawati et al. , 2022).
- 2. Penyusunan Sistem (*Desain*)**
Berdasarkan analisis, pengembang merancang struktur sistem, termasuk desain basis data, antarmuka pengguna, dan alur proses bisnis. Desain juga mempertimbangkan kebutuhan semua pihak terkait agar sistem dapat diterima (Purnamasari dan Siregar, 2023).
- 3. Pelaksanaan (*Implementation*)**
Tahap ini meliputi penulisan kode berdasarkan desain. Sistem dibuat menggunakan bahasa pemrograman *PHP* untuk *backend*, serta *HTML*, *CSS*, dan *JavaScript* dengan *framework Bootstrap* untuk *frontend*. Data disimpan di *MySQL*, yang dirancang dengan prinsip normalisasi agar lebih efisien (Asrin dan Utami, 2023).
- 4. Pengujian (*Verifikasi*)**
Setelah sistem selesai, dilakukan *verifikasi* dengan menguji semua fungsi. Pengujian menggunakan metode *Black Box Testing*, yang fokus pada *input* dan *output* tanpa melihat kode internal (Zhang et al. , 2023). Uji coba dilakukan pada fungsi utama seperti *login*, *input* dan *edit* data mahasiswa, pencarian data, serta pembuatan laporan dalam format *PDF* atau *Excel*.
- 5. Perawatan (*Maintenance*)**
Tahap ini memantau sistem secara berkala setelah implementasi untuk memastikan sistem berjalan lancar. Jika ada kesalahan atau permintaan fitur baru,

pengembang akan memperbaikinya atau menambahkan fitur tersebut (Purnamasari dan Siregar, 2023).

Selama pengembangan, sistem dibuat di lingkungan lokal dengan *XAMPP* sebagai server lokal dan *Sublime Text* sebagai editor kode. Pengujian dilakukan dengan *browser Google Chrome*, pada perangkat dengan *prosesor Intel Celeron N4020*, *RAM 8 GB*, *SSD 256 GB*, dan *Windows 11 Pro 64-bit*.

Pengujian sistem menggunakan metode *Black Box Testing*, di mana *input* diuji terhadap *output* yang diharapkan tanpa memeriksa kode program secara langsung. Pendekatan ini dinilai tepat karena mengevaluasi sistem dari sudut pandang pengguna (Corradini et al. , 2023). Penilaian pengujian mempertimbangkan tiga aspek: *fungsi* (apakah sistem bekerja sesuai tugasnya), *kegunaan* (kemudahan penggunaan), dan *keandalan* (stabilitas dan ketahanan sistem).

Metode *Waterfall* dan pengujian *Black Box* dianggap cocok untuk pengembangan sistem ini karena batasan proyek sudah jelas sejak awal. Kombinasi ini memberikan keseimbangan antara organisasi kerja yang teratur dan verifikasi sistem yang berfokus pada pengguna. Dengan demikian, sistem informasi yang dikembangkan diharapkan dapat mendukung pengelolaan data mahasiswa dengan lebih efektif, efisien, dan andal.

C. HASIL DAN PEMBAHASAN

1. Ouput Teknis dari Perangkat Lunak/Hardware

a. Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan dalam menjalankan aplikasi program ini menggunakan laptop Lenovo dengan *Processor Intel(R) Celeron(R) N4020* dengan spesifikasi sebagai berikut:

Tabel 1. Spesifikasi Perangkat Keras

No	Jenis	Spesifikasi	Keterangan
1	Computer Server	<i>Processor</i>	AMD Ryzen 7 3700X 3.6GHz
		<i>Memory RAM</i>	DDR4 32 GB
		<i>VGA</i>	NVIDIA Quadro P620 4GB
		<i>System Type</i>	64-bit <i>Operating System</i>
		<i>Hardisk</i>	4 Terabyte + 512GB SSD
		<i>Display</i>	LED 21 inci Full HD 1920 x 1080 pixel

		OS	<i>Windows Server 2019</i>
2	Computer Client	<i>Processor</i>	Minimal Intel Pentium G4400
		<i>Memory RAM</i>	DDR3 4 GB
		<i>VGA</i>	Intel HD Graphics
		<i>System Type</i>	64-bit <i>Operating System</i>
		<i>Hardisk</i>	250 MegaByte
		<i>Display</i>	LED 19 inci HD 1366 x 768 pixel
		OS	<i>Windows 10 Home 64 Bit</i>

b. Spesifikasi Perangkat Lunak (*Software*)

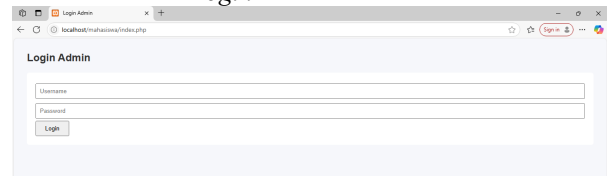
Perangkat lunak yang digunakan untuk mengimplementasikan sistem yaitu sebagai berikut :

Tabel 2. Tabel Spesifikasi Perangkat Lunak

No	Spesifikasi	Keterangan
1	<i>OS</i>	<i>Windows 11 Pro</i>
2	<i>Database</i>	PHP MyAdmin 5.2.1
3	<i>Xampp</i>	Versi 8.2.12
4	Program <i>WEB</i>	<i>HTML, Javascript, PHP, CSS, Apache, PHP</i>
5	<i>Software Editor</i>	<i>Vs Code</i>

2. Fungsi-Fungsi yang Berhasil Dijalankan

a. Halaman *Login Admin*

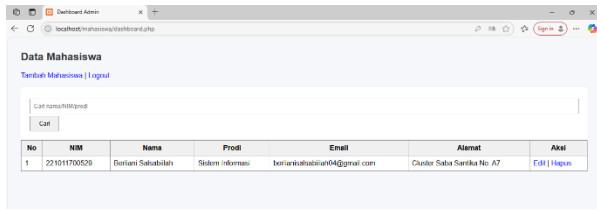


Gambar 1. Halaman Login

Deskripsi:

Sistem berhasil menampilkan halaman login sederhana sebagai antarmuka awal bagi pengguna. Fungsi validasi yang diterapkan akan memeriksa *kredensial* yang dimasukkan, dan apabila sesuai, sistem secara otomatis mengarahkan pengguna menuju halaman *dashboard*. Sebaliknya, jika pengguna memasukkan *username* atau *password* yang salah, sistem akan menampilkan pesan kesalahan berupa "Username atau password salah!". Selain itu, apabila pengguna tidak mengisi salah satu atau kedua kolom *input*, sistem akan menampilkan peringatan "Please fill out this field" untuk memastikan seluruh data telah dilengkapi sebelum diproses.

b. *Dashboard* – Lihat Daftar Mahasiswa



Gambar 2. Halaman Dashboard

Deskripsi:

Setelah berhasil *login*, admin akan langsung diarahkan menuju halaman *dashboard* sebagai pusat kendali sistem. Pada halaman ini, ditampilkan sebuah tabel yang memuat seluruh data mahasiswa yang telah tersimpan dalam basis data, yang diurutkan secara menurun berdasarkan *id mahasiswa*, sehingga data terbaru muncul di bagian atas. Tabel tersebut terdiri dari beberapa kolom informasi, yaitu: No, NIM, Nama, Program Studi (Prodi), Email, Alamat, serta kolom Aksi yang berisi tombol untuk melakukan proses edit atau hapus terhadap data yang ada.

c. Fungsi Tambah Data Mahasiswa

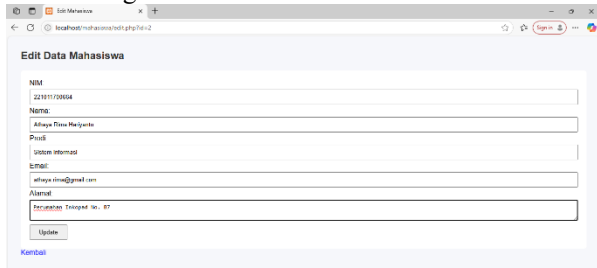


Gambar 3. Fungsi Tambah Data Mahasiswa

Deskripsi:

Admin dapat mengakses halaman khusus untuk menambahkan data mahasiswa baru melalui fitur "Tambah Mahasiswa". Pada halaman ini, tersedia *form input* yang dilengkapi dengan validasi, termasuk validasi format *email* dan pengecekan untuk memastikan tidak ada duplikasi NIM yang telah terdaftar sebelumnya. Setelah admin mengisi *form* dengan benar dan menekan tombol "Simpan", data mahasiswa akan langsung disimpan ke dalam *database* dan secara otomatis ditampilkan pada tabel di halaman *dashboard*.

d. Fungsi Edit Data Mahasiswa

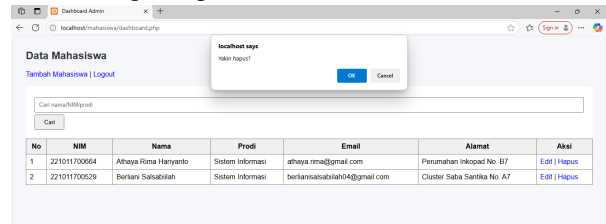


Gambar 4. Fungsi Edit Data Mahasiswa

Deskripsi:

Ketika admin mengklik tombol "Edit" pada salah satu baris data mahasiswa, sistem akan menampilkan *form* dengan data lama yang sudah terisi secara otomatis sesuai dengan entri yang dipilih. Setelah admin melakukan perubahan pada data yang diperlukan dan menekan tombol "Update", sistem akan memperbarui informasi tersebut di dalam *database*. Perubahan yang dilakukan akan langsung ditampilkan secara real-time pada tabel data di halaman *dashboard*.

e. Fungsi Hapus Data Mahasiswa

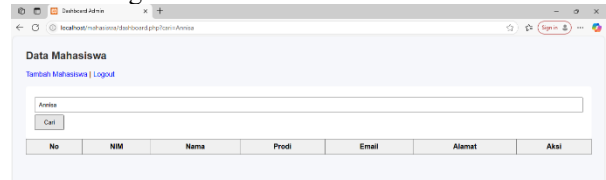


Gambar 5. Fungsi Hapus Data Mahasiswa

Deskripsi:

Saat admin mengklik tombol "Hapus" pada salah satu entri data mahasiswa, sistem akan menampilkan *popup konfirmasi* untuk memastikan tindakan penghapusan. Jika admin mengonfirmasi tindakan tersebut, data yang dipilih akan langsung dihapus dari *database*. Setelah proses penghapusan berhasil, tabel pada halaman *dashboard* akan diperbarui secara otomatis, dan baris data yang telah dihapus tidak lagi ditampilkan.

f. Fungsi Pencarian Data

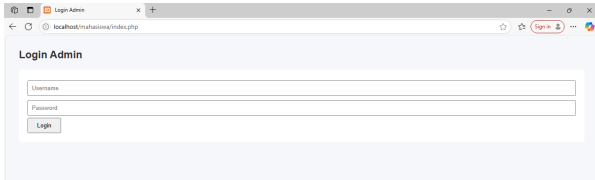


Gambar 6. Fungsi Pencarian Data

Deskripsi:

Admin dapat melakukan pencarian data mahasiswa dengan mengetikkan kata kunci tertentu pada kotak *input* "Cari nama/NIM/prodi" yang tersedia di halaman *dashboard*. Setelah tombol "Cari" di klik, sistem akan menjalankan proses pencarian menggunakan *query LIKE* untuk memfilter data berdasarkan nama, NIM, atau program studi yang sesuai dengan kata kunci yang dimasukkan. Hanya data yang memenuhi kriteria pencarian yang akan ditampilkan, sehingga pencarian menjadi lebih cepat dan efisien.

g. Fungsi Logout



Gambar 7. Fungsi Logout

Deksripsi:

Setelah pengguna mengklik tombol "Logout" pada halaman *dashboard*, sistem akan menjalankan perintah *session_destroy()* untuk menghapus seluruh data sesi yang tersimpan di server, termasuk variabel seperti *\$_SESSION['username']*. Setelah sesi dihapus, pengguna secara otomatis akan diarahkan kembali ke halaman *login (index.php)*. Apabila pengguna mencoba mengakses halaman *dashboard.php* tanpa melalui proses *login* terlebih dahulu, sistem akan mendeteksi ketiadaan sesi aktif dan segera mengarahkan kembali ke halaman *login*. Mekanisme ini memastikan bahwa hanya pengguna yang telah berhasil *login* yang dapat mengakses area admin, sehingga menjaga keamanan aplikasi dari akses yang tidak sah.

3. Tabel Format Analisis

Format Analisis berikut adalah format pengujian fungsional lengkap yang mencakup berbagai skenario, baik positif maupun negatif, untuk setiap fitur sistem:

Tabel 3. Tabel Format Analisis

N O	Fitur yang Dianalisis	langkah Pengujian	Output Aktual	Status
1	Login sukses	1. Buka localhost/mahasiswa/index.php 2. Masukkan username & password 3. Klik Login	Sesuai	Pass

N O	Fitur yang Dianalisis	langkah Pengujian	Output Aktual	Status
2	Login Gagal - Password Salah	1. Buka localhost/mahasiswa/index.php 2. Masukkan username & password 3. Klik Login	Sesuai	Pass
3	Login Gagal - Username Salah	1. Buka localhost/mahasiswa/index.php 2. Masukkan username & password 3. Klik Login	Sesuai	Pass
4	Login Gagal - Field Kosong	1. Buka localhost/mahasiswa/index.php 2. Masukkan username & password 3. Klik Login	Sesuai	Pass

N O	Fitur yang Dianalisis	langkah Pengujian	Output Aktual	Status
5	Login Gagal - Username Salah & Password Salah	1. Buka localhost/mahasiswa/index.php 2. Masukkan username & password 3. Klik Login	Sesuai	Pass
6	Tambah Mahasiswa - Valid	1. Klik Tambah Mahasiswa 2. Isi form 3. Klik Simpan	Sesuai	Pass
7	Tambah Mahasiswa - Email Tidak Valid	1. Klik Tambah Mahasiswa 2. Isi form 3. Klik Simpan	Sesuai	Pass
8	Tambah Mahasiswa - Duplikat NIM	1. Klik Tambah Mahasiswa 2. Isi form 3. Klik Simpan	Sesuai	Pass
9	Tambah Mahasiswa	1. Klik Tambah	Sesuai	Pass

N O	Fitur yang Dianalisis	langkah Pengujian	Output Aktual	Status
	swa - Field Kosong	Mahasiswa 2. Isi form 3. Klik Simpan		
10	Edit Mahasiswa - Valid	1. Klik Edit 2. Isi form 3. Klik Update	Sesuai	Pass
11	Edit Mahasiswa - Email Tidak Valid	1. Klik Edit 2. Isi form 3. Klik Update	Sesuai	Pass
12	Edit Mahasiswa - Duplikasi NIM	1. Klik Edit 2. Isi form 3. Klik Update	Sesuai	Pass
13	Hapus Mahasiswa - Konfirmasi OK	1. Klik Hapus 2. Klik OK pada popup	Sesuai	Pass
14	Hapus Mahasiswa - Konfirmasi Cancel	1. Klik Hapus 2. Klik Cancel pada popup	Sesuai	Pass

N O	Fitur yang Dianalisis	langkah Pengujian	Output Aktual	Status
15	Pencarian Mahasiswa - Ada Hasil	1. Ketik "Berli" di kolom cari 2. klik Cari	Sesuai	Pass
16	Pencarian Mahasiswa - Tidak Ada	1. Ketik "Annisa" di kolom cari 2. klik Cari	Sesuai	Pass
17	Logout	1. Klik Logout	Sesuai	Pass

4. PENUTUP

1. Kesimpulan

Melalui penelitian berjudul "Perancangan dan Implementasi Sistem Informasi Manajemen Data Mahasiswa Berbasis Web", dapat disimpulkan bahwa sistem yang dikembangkan telah berhasil memenuhi kebutuhan dasar pengelolaan data mahasiswa di lingkungan institusi pendidikan. Sistem ini mampu menjalankan fungsi-fungsi utama seperti *login*, penambahan data mahasiswa, pengeditan, penghapusan, pencarian data, serta *logout*, secara efektif dan efisien.

Proses pengembangan yang dilakukan menggunakan metode *Waterfall* telah memberikan alur kerja yang terstruktur dan sistematis, memungkinkan dokumentasi serta evaluasi yang jelas di setiap tahap. Pengujian sistem menggunakan metode *Black Box Testing* juga telah menunjukkan bahwa sistem berfungsi sesuai dengan spesifikasi, dengan seluruh skenario uji menghasilkan *output* yang sesuai dan berstatus *pass*. Sistem ini juga mendukung pengelolaan data yang *real-time* dan dapat diakses kapan saja melalui *browser*, sehingga memperkuat aspek efisiensi kerja dan kemudahan akses bagi admin maupun pengguna lain yang berkepentingan.

Dengan adanya sistem informasi ini, pengelolaan data mahasiswa menjadi lebih terintegrasi, cepat, dan minim kesalahan. Sistem ini juga dapat menjadi model awal bagi institusi lain yang ingin mengembangkan sistem manajemen data akademik berbasis *web* secara mandiri.

2. Saran

Agar sistem dapat berkembang lebih optimal di masa mendatang, terdapat beberapa saran yang dapat dipertimbangkan untuk pengembangan lanjutan. Pertama, perlu ditambahkan fitur *otentikasi* tingkat lanjut serta manajemen hak akses berdasarkan peran pengguna (*role-based access control*) agar sistem lebih aman dan fleksibel untuk berbagai jenis pengguna, seperti admin, dosen, dan mahasiswa. Kedua, pengembangan *dashboard* analitik yang mampu menampilkan statistik data mahasiswa secara visual juga menjadi nilai tambah dalam pengambilan keputusan. Selain itu, sistem sebaiknya dilengkapi dengan fitur *ekspor* data dalam format yang lebih beragam, seperti *CSV* atau *XML*, serta kemampuan *import* data massal guna mempercepat proses *input* data dalam jumlah besar. Terakhir, integrasi dengan platform eksternal seperti sistem akademik kampus atau email mahasiswa juga dapat dilakukan untuk memperluas cakupan fungsi sistem dan meningkatkan efisiensi kerja. Penelitian ini diharapkan mampu memberikan kontribusi nyata dalam pengembangan sistem informasi pendidikan, serta menjadi referensi dalam penerapan metode pengujian *Black Box* sebagai pendekatan validasi sistem berbasis *web* yang efektif.

3. Ucapan Terima Kasih

Penulis, Athaya Rima Hariyanto dan Berliani Salsabiilah, menyampaikan rasa terima kasih yang sebesar-besarnya kepada dosen pembimbing dan seluruh staf pengajar di Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Pamulang atas bimbingan, ilmu, dan arahan yang telah diberikan selama penyusunan makalah ini. Terima kasih juga ditujukan kepada teman-teman mahasiswa serta semua pihak yang telah memberikan dukungan, motivasi, dan masukan yang berharga dalam proses penyelesaian tugas ini. Tanpa bantuan mereka, makalah ini tidak akan tersusun dengan baik seperti sekarang.

E. DAFTAR PUSTAKA

- Aisyah, N., & Wahyudi, R. (2021). Penerapan model *Waterfall* dalam pengembangan sistem informasi akademik. *Jurnal Teknologi Informasi dan Pendidikan*, 14(2), 121–128.
- Anwar, C. (2022). Application of academic information system with extreme programming method (Case study: Jakarta International Polytechnic).
- Anwar, C. (2024). Rekomendasi teknis untuk pengolahan data berbasis web. *Jurnal Informatika Utama*, 2(1), 50–54.
- Anwar, C., & Riyanto, J. (2019). Perancangan sistem informasi Human Resources Development pada PT. Semacom Integrated. *International Journal of Education, Science, Technology, and Engineering (IJESTE)*, 2(1), 19–38.

- Anwar, C., Jagat, L. S., Yanti, I., Anjarsari, E., & Sholihah, N. A. (2023). Pengembangan media pembelajaran berbasis teknologi untuk meningkatkan kemampuan anak. *Caruban: Jurnal Ilmiah Ilmu Pendidikan Dasar*, 6(2), 154–163.
- Anwar, C., Kom, S., Kom, M., Santiari, C. N. P. L., & Sitorus, Z. (2023). Buku referensi sistem informasi berbasis kearifan lokal.
- Anwar, C., Nurhasanah, M., Aflaha, D. S. I., & Handayani, S. (2023). Development of information technology-based learning media for educators in elementary schools. *Jurnal Konseling Pendidikan Islam*, 4(2), 345–353.
- Anwar, C., et al. (2023). The application of Mobile Security Framework (MOBSF) and Mobile Application Security Testing Guide to ensure the security in mobile commerce applications. *Jurnal Sistim Informasi dan Teknologi*, 11(1), 97–102.
- Asrin, F., & Utami, G. (2023). Implementing website-based school information systems using the Waterfall model. *Journal of Information Systems and Informatics*, 5(2), 590–614.
- Corradini, D., Pasqua, M., & Ceccato, M. (2023). Automated black-box testing of mass assignment vulnerabilities in RESTful APIs. *arXiv preprint arXiv:2302.09550*.
- Fahrullah. (2021). Implementasi pengujian Black Box pada sistem informasi monitoring akademik dengan Equivalence Partitions. *Jurnal Teknologi Informasi*, 13(2), 55–62.
- Handayani, M., & Ningsih, E. (2022). Analisis kelemahan metode Waterfall dalam pengembangan perangkat lunak. *Jurnal Rekayasa Sistem*, 10(1), 33–40.
- Handayani, T., Silalahi, L. M., Nugroho, S. S. P., Anwar, C., Mursyidin, I. H., Sumantri, A., ... & Yulianti, B. (2025). Pengantar sistem informasi: Konsep, teknologi, dan implementasi. CV Rey Media Grafika.
- Hendartie, D., Nugroho, A. P., & Lestari, M. (2023). Pengujian sistem informasi PMB menggunakan Black Box Testing pada Universitas X. *Jurnal Rekayasa Sistem Informasi*, 15(1), 44–50.
- Indra, S., Anwar, C., Kom, S., Asparizal, S., Kom, M., Nur, R. A., ... & Hafrida, L. (2024). Komputer dan masyarakat. CV Rey Media Grafika.
- Purnamasari, D., & Siregar, H. (2023). Perancangan sistem informasi pengolahan data mahasiswa berbasis web menggunakan Waterfall model. *Jurnal Teknologi dan Sistem Informasi*, 11(1), 45–52.
- Rahmawati, S., Prasetyo, T., & Widodo, B. (2022). Analisis kebutuhan dalam pengembangan sistem informasi akademik menggunakan model Waterfall. *Jurnal Informatika dan Komputer*, 17(3), 203–210.
- Rizal, M., & Nuryasin, A. (2025). Implementasi Blackbox Testing pada sistem informasi sirkulasi perpustakaan berbasis website dengan Equivalence Partitioning. *Jurnal Informatika*, 19(1), 18–26.
- Samsumar, L. D., Nasiroh, S., Farizy, S., Anwar, C., Mursyidin, I. H., Rosdiyanto, R., ... & Prastyo, D. (2025). Keamanan sistem informasi: Perlindungan data dan privasi di era digital. CV Rey Media Grafika.
- Wijayanti, R. R., S. ST, M. M. S. I., Anwar, C., Kom, S., Indra, S., Kom, M., ... & Kom, M. (2023). *Arsitektur dan organisasi komputer*. CV Rey Media Grafika.
- Zhang, T., Huang, H., Lu, Y., et al. (2023). State-sensitive black-box web application scanning. *Applied Sciences*, 13(16), 9212.