
SISTEM INFORMASI TERINTEGRASI BERBASIS WEB UNTUK PENGUMPULAN DAN VALIDASI TUGAS AKHIR MAHASISWA

Fachra Maulidiana Anggit¹, Faiz Rafdhi²

^{1,2} Universitas Saintek Muhammadiyah

Email: 1maulidianafachra@gmail.com, 2faizrafdhi@gmail.saintekmu.ac.id

Abstrak

Universitas Saintek Muhammadiyah merupakan perguruan tinggi yang bergerak pada bidang inovasi dan teknologi, namun kegiatan akademik masih berlangsung secara manual seperti pengumpulan tugas akhir mahasiswa, hal ini sangat beresiko mengingat metode manual berbasis kertas dapat menimbulkan kehilangan dokumen, duplikasi data, dokumen rusak, serta kesulitan dalam proses verifikasi. Pada penelitian ini dibahas perancangan sistem informasi terintegrasi berbasis web untuk mendukung proses pengumpulan dan validasi tugas akhir mahasiswa di Universitas Saintek Muhammadiyah.

Metode pengembangan perangkat lunak yang digunakan adalah *waterfall*, dengan tahapan analisis kebutuhan, desain sistem, pengkodean, dan pengujian. Sistem dibangun menggunakan *PHP Native* dengan basis data *MySQL*, serta menerapkan autentikasi berbasis role untuk mahasiswa, dosen, dan admin.

Hasil implementasi menunjukkan bahwa sistem mampu menyediakan fitur unggah dokumen, validasi dosen pembimbing, monitoring pengumpulan, serta laporan administrasi secara terintegrasi. Pengujian metode *black-box* membuktikan bahwa seluruh fungsi utama berjalan sesuai kebutuhan. Dengan adanya sistem ini, proses pengumpulan tugas akhir menjadi lebih efisien, terstruktur, dan terdokumentasi dengan baik, sehingga mendukung digitalisasi administrasi akademik di perguruan tinggi.

Kata Kunci: berbasis web, metode *waterfall*, sistem informasi, tugas akhir mahasiswa

Abstract

Saintek Muhammadiyah University is a higher education institution focused on innovation and technology. However, its academic activities such as the submission of student final projects are still conducted manually. This poses significant risks, as paper-based manual methods can lead to document loss, data duplication, damaged files, and difficulties in the verification process. This research discusses the design of a web-based integrated information system to support the submission and validation process of student final projects at Saintek Muhammadiyah University.

The software development method used is the *waterfall* model, with stages of requirements analysis, system design, coding, and testing. The system was built using Native PHP with a *MySQL* database, and implements *role-based authentication* for students, lecturers, and administrators.

The implementation results show that the system provides features for document *upload*, supervisor validation, submission monitoring, and integrated administrative reporting. *Black-box testing* proves that all main functions run according to requirements. With this system, the process of collecting final projects becomes more efficient, structured, and well-documented, thereby supporting the digitalization of academic administration in higher education.

Keywords: information system, student final project, web-based, *waterfall* method

1. PENDAHULUAN

Teknologi informasi saat ini merupakan kebutuhan penting bagi setiap individu, dan kelompok terutama di dunia pendidikan seperti Universitas. Pada era ini, memanfaatkan dan mengandalkan teknologi merupakan salah satu solusi untuk meningkatkan efektivitas dan efisiensi sebagai hasil yang lebih baik[1]. Dengan kemajuan teknologi informasi ini tentu membuat sistem pengelolaan dan layanan akademik berubah signifikan[2] akibatnya banyak orang perlu mencari solusi dan jalan alternatif sebagai

pemecah masalah mereka guna meningkatkan kualitas teknologi informasi terutama pendidikan[3]. Dalam dunia pendidikan, penilaian akademik merupakan hal yang krusial sehingga membutuhkan pengelolaan data yang baik, masih banyak institusi pendidikan yang masih menggunakan kertas atau *software* seperti *Microsoft Word* atau *Microsoft Excel*. [4]

Universitas Saintek Muhammadiyah merupakan institusi pendidikan tinggi yang berorientasi pada inovasi dan digitalisasi[5], namun pada proses administrasi masih banyak yang belum terintegrasi dengan baik bahkan masih sering menggunakan kertas untuk pengumpulan dokumen. Hal tersebut sangat berisiko terutama untuk dokumen penting seperti tugas akhir mahasiswa, resiko dokumen hilang, duplikasi data, sulitnya validasi dokumen, penumpukan berkas dan tentunya kurang efisien terhadap waktu.

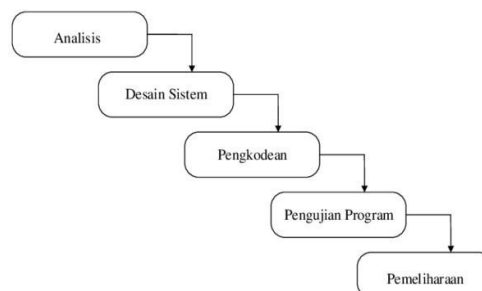
Maka dari itu Universitas Saintek Muhammadiyah membutuhkan sistem yang mampu mendukung proses pengumpulan tugas akhir secara lebih efektif dan efisien. Sistem informasi berbasis web dengan memanfaatkan *PHP Native* dan *MySQL* menjadi salah satu solusi yang dapat diterapkan untuk mengatasi permasalahan tersebut[6]. PHP sendiri dipilih karena sifatnya fleksibel dan mudah diimplementasikan serta memudahkan developer untuk mengembangkan web yang interaktif[7]. Dengan adanya sistem ini, mahasiswa dapat mengunggah dokumen tugas akhir secara daring, sementara dosen pembimbing dan pihak administrasi dapat melakukan verifikasi serta monitoring secara terintegrasi. Dengan adanya sistem informasi ini, kebutuhan administrasi dan proses pengumpulan tugas akhir lebih efisien.

Berdasarkan permasalahan yang timbul, maka terbentuk susunan rumusan masalah. Adapun rumusan masalah tersebut diantaranya:

1. Bagaimana merancang sistem informasi berbasis web yang dapat mendigitalisasi proses pengumpulan tugas akhir mahasiswa di Universitas Saintek Muhammadiyah?
2. Bagaimana membangun fitur monitoring dan pelaporan bagi admin supaya dapat memantau seluruh proses pengumpulan tugas akhir secara *real-time* dan terintegrasi dalam satu sistem?
3. Bagaimana penerapan metode *waterfall* dalam pengembangan sistem informasi pengumpulan tugas akhir di Universitas Saintek Muhammadiyah?

2. METODE

Metode pengembangan sistem yang digunakan untuk membangun sistem ini yaitu menggunakan metode *waterfall*. Metode *waterfall* merupakan metode pendekatan pengembangan perangkat lunak yang sistematis dan berurutan, *waterfall* diperkenalkan oleh Wiston Royce sekitar tahun 1970 dan dikenal juga sebagai linear sequential model karena tahapannya dilalui secara linier[8]. Metode ini dipilih karena fleksibel dan tahapannya terstruktur sehingga cocok untuk web akademik.



Gambar 1. Metode *Waterfall*

Berdasarkan gambar 1, tahapan metode *waterfall* adalah sebagai berikut:

a. Analisis

Tahap mengumpulkan dan menganalisis kebutuhan sistem, kebutuhan sistem dibagi menjadi dua yaitu kebutuhan Fungsional (Tabel 1) dan kebutuhan nonFungsional (tabel 2) :

Tabel 1 Kebutuhan Fungsional

No	Kebutuhan Fungsional	Deskripsi
1	Autentikasi pengguna	Sistem menyediakan <i>Login</i> untuk mahasiswa, dosen, dan admin sesuai hak akses.
2	Manajemen data mahasiswa	Admin dapat menambah, mengedit, dan menghapus data mahasiswa.
3	Pengumpulan tugas akhir	Mahasiswa dapat mengunggah <i>file</i> tugas akhir (PDF/DOCX/PPT) ke sistem
4	Validasi dosen pembimbing	Dosen dapat memverifikasi, memberi catatan, dan menentukan status(Diterima/Revisi/Ditolak)
5	Monitoring Pengumpulan	Admin dapat melihat daftar mahasiswa yang sudah/belum mengumpulkan
6	Laporan Pengumpulan	Sistem dapat menghasilkan laporan tugas akhir
7	Status	Mahasiswa mendapat status tugas akhir (diterima/revisi/ditolak)

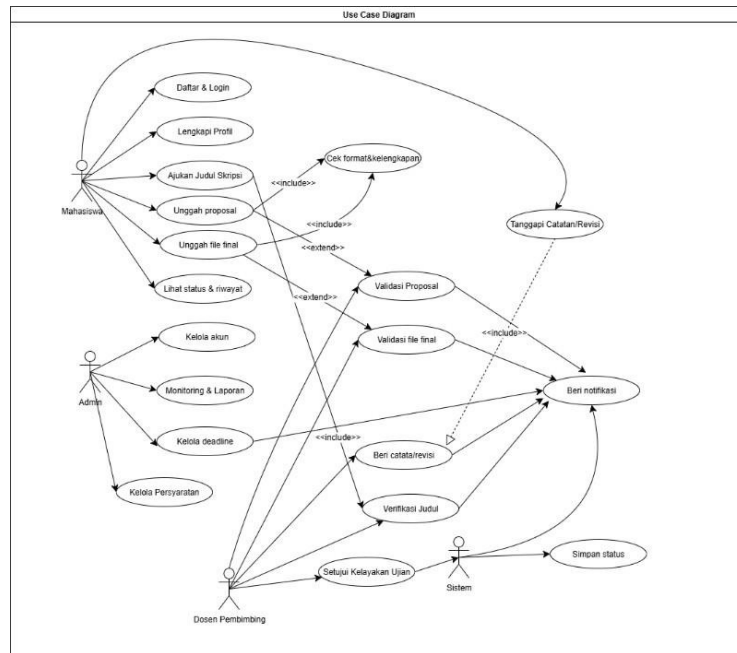
Tabel 2. Kebutuhan Non-Fungsional

No	Kebutuhan Non-Fungsional	Deskripsi
1	Keamanan	Sistem menggunakan autentikasi dan otorisasi berbasis <i>role</i> .
2	Ketersediaan	Sistem dapat diakses secara daring 24/7.
3	Kinerja	Proses unggah dan unduh dokumen berjalan cepat dan stabil
4	<i>Usability</i>	<i>Interface</i> sederhana dan mudah digunakan oleh <i>user</i>
5	Portabilitas	Sistem dapat diakses melalui berbagai <i>browser</i> (<i>Chrome, Firefox, Edge</i>)

b. Desain Sistem

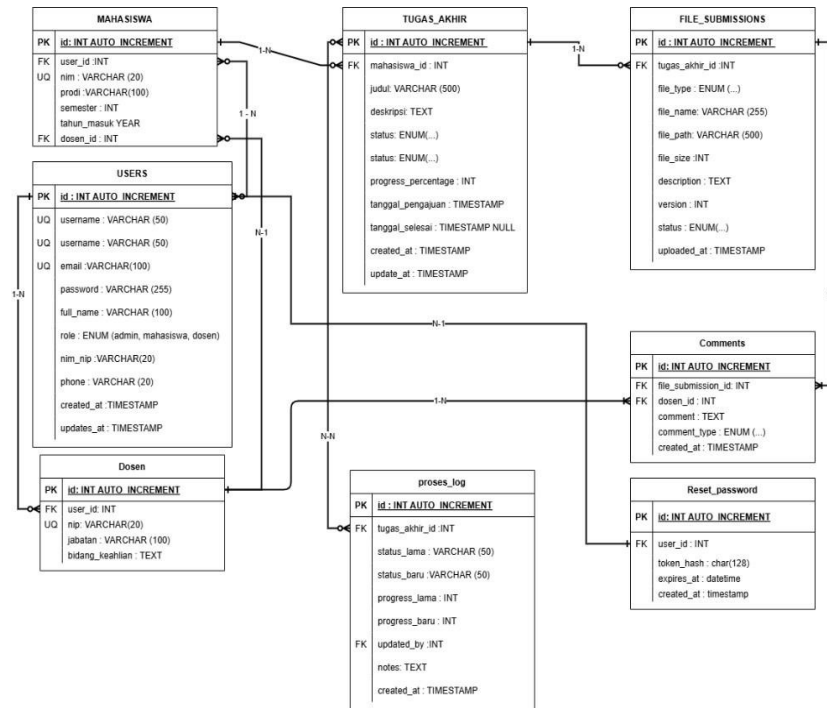
Tahap desain sistem dilakukan setelah proses analisis sistem selesai dilakukan, tujuannya untuk memberikan gambaran menyeluruh terkait arsitektur sistem, alur pengguna, serta struktur basis data yang digunakan[9]. Pada penelitian ini digunakan *Use Case Diagram*, *Entity Relationship Diagram(ERD)*, dan rancangan antarmuka aplikasi.

Use Case Diagram(Gambar 2) digunakan untuk menggambarkan interaksi antara aktor dengan sistem. Terdapat tiga aktor utama, yaitu mahasiswa, dosen pembimbing, dan admin. Mahasiswa berperan dalam proses unggah dokumen tugas akhir, melihat status verifikasi, serta menerima notifikasi. Dosen pembimbing berperan dalam melakukan validasi, memberikan catatan revisi, dan menentukan status dokumen. Admin bertugas mengelola data mahasiswa, memantau proses pengumpulan, serta menghasilkan laporan. Diagram ini menunjukkan batasan sistem dan fungsi-fungsi yang dapat diakses oleh masing-masing aktor.



Gambar 2. Use Case Diagram

Selanjutnya, *Entity Relationship Diagram (ERD)* digunakan untuk merancang struktur basis data. ERD tidak seperti diagram alir yang menjelaskan alur program [10] namun ERD (gambar 2) menggambarkan hubungan antar entitas, seperti entitas mahasiswa, dosen, tugas akhir, dan admin. mahasiswa memiliki atribut NIM, nama, program studi, dan tahun angkatan. Entitas tugas akhir memiliki atribut judul, file, status, serta tanggal unggah. Relasi antara mahasiswa dan tugas akhir bersifat satu ke banyak, karena setiap mahasiswa dapat mengunggah lebih dari satu dokumen revisi. Relasi antara dosen dan tugas akhir juga bersifat satu ke banyak, karena seorang dosen dapat membimbing beberapa mahasiswa.



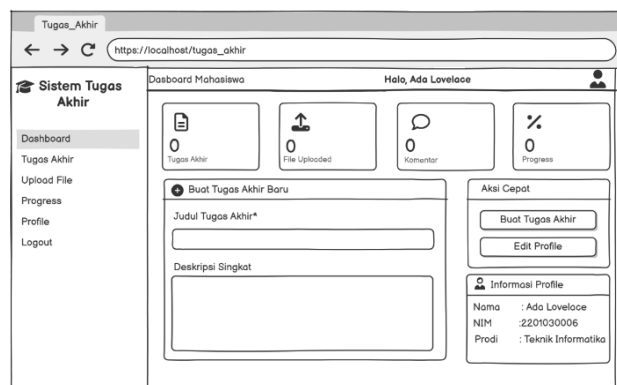
Gambar 3. Entity Relationship Diagram

Selain itu, dilakukan perancangan antarmuka (*User Interface*) untuk mempermudah interaksi pengguna dengan sistem. Gambar 4 merupakan *wireframe* dari halaman *Log-In* dan setiap role memiliki hak akses masing-masing, sehingga saat *Log-In* dengan memasukkan *e-mail* dan *password* sistem akan melakukan validasi *user*, jika berhasil maka akan masuk ke halaman utama masing-masing role.



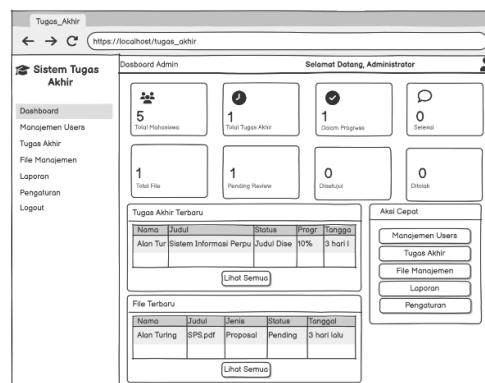
Gambar 4. Halaman *Login*

Kemudian jika mahasiswa *Login* akan diarahkan ke halaman utama mahasiswa (Gambar 5), fitur mahasiswa diantaranya Membuat judul tugas akhir, unggah *file-file* tugas akhir, melihat komentar dan progress pengajuan tugas akhir.



Gambar 5 Halaman Mahasiswa

Gambar 6 merupakan halaman admin, admin memiliki hak untuk memantau seluruh proses pengajuan tugas akhir mahasiswa, aksi yang dilakukan mahasiswa dan dosen, serta manajemen *user*.



Gambar 6 Halaman Admin

c. Pengkodean

Tahap pengkodean merupakan proses implementasi dari desain sistem yang telah dibuat sebelumnya. Pengembangan sistem informasi pengumpulan tugas akhir mahasiswa dilakukan menggunakan bahasa pemrograman *PHP Native*. Struktur program dibangun dengan pendekatan modular, di mana setiap fungsi utama sistem dipisahkan ke dalam *file* yang berbeda, seperti *file* untuk autentikasi pengguna, pengelolaan data mahasiswa, pengumpulan tugas akhir, validasi dosen, dan pembuatan laporan. Basis data yang digunakan adalah *MySQL*, dengan koneksi langsung melalui ekstensi *mysqli* atau *PDO*.

Antarmuka sistem dikembangkan menggunakan kombinasi *HTML*, *CSS*, dan *JavaScript* untuk memberikan tampilan yang sederhana dan mudah digunakan. Proses unggah dokumen tugas akhir diimplementasikan melalui fitur unggah berkas pada *PHP*, kemudian dokumen disimpan pada direktori *server* dan informasinya dicatat pada tabel basis data. Setiap pengguna, yaitu mahasiswa, dosen, dan admin, memiliki hak akses berbeda yang diatur melalui mekanisme *session* dan *role-based access control* sederhana. Pada proses *Login*, sistem melakukan validasi akun pengguna, menyimpan informasi role pada *session*, lalu mengarahkan pengguna ke halaman sesuai hak aksesnya.

d. Pengujian Program

Tahap pengujian dilakukan untuk memastikan bahwa sistem informasi pengumpulan tugas akhir mahasiswa berjalan sesuai dengan kebutuhan yang telah ditentukan. Pengujian dilakukan menggunakan metode *black-box testing*, yaitu dengan memeriksa fungsi-fungsi utama sistem tanpa melihat kode program secara langsung. Fokus pengujian adalah pada *input* yang diberikan pengguna dan *output* yang dihasilkan sistem[11].

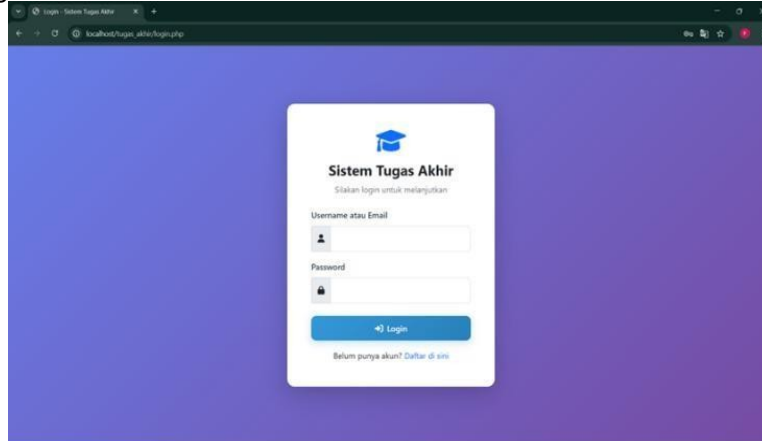
Tabel 3. *Black-Box Testing*

No	Fitur / Fungsi	Skenario Pengujian	Data Uji (<i>Input</i>)	Hasil yang Diharapkan	Hasil Pengujian	Status
1	<i>Login</i>	Mengosongkan semua kolom	<i>Username:</i> (kosong), <i>Password:</i> (kosong)	Sistem menampilkan pesan " <i>Username/Password</i> tidak boleh kosong"	Sesuai Harapan	Valid
2	<i>Login</i>	<i>Login</i> dengan data yang salah	<i>Username:</i> salah, <i>Password:</i> salah	Sistem menampilkan pesan " <i>Username</i> atau <i>Password</i> salah"	Sesuai Harapan	Valid
3	Upload <i>File</i>	Mengunggah <i>file</i> dengan format yang salah	<i>File:</i> .exe atau .mp3	Sistem menolak <i>file</i> dan menampilkan pesan " <i>Format file</i> harus .pdf"	Sesuai Harapan	Valid
4	Upload <i>File</i>	Mengunggah <i>file</i> melebihi batas ukuran (misal >10MB)	<i>File</i> PDF ukuran 15MB	Sistem menampilkan pesan " <i>Ukuran file</i> terlalu besar"	Sesuai Harapan	Valid
5	Upload <i>File</i>	Mengunggah dokumen tugas akhir dengan data lengkap	<i>File:</i> TA_Nama_NIM.pdf	Sistem berhasil menyimpan <i>file</i> dan muncul di daftar pengiriman	Sesuai Harapan	Valid
6	Verifikasi	Admin mengubah status tugas akhir	Klik tombol " <i>Setujui</i> "	Status tugas akhir mahasiswa berubah menjadi " <i>Tervalidasi</i> "	Sesuai Harapan	Valid
7	Pencarian	Mencari data mahasiswa berdasarkan NIM	<i>Input:</i> NIM Mahasiswa	Sistem menampilkan data mahasiswa yang sesuai dengan NIM tersebut	Sesuai Harapan	Valid
8	<i>Logout</i>	Keluar dari sistem	Klik tombol <i>Logout</i>	Sistem menghapus <i>session</i> dan kembali ke halaman <i>Login</i>	Sesuai Harapan	Valid

Hasil pengujian (Tabel 3) menunjukkan bahwa seluruh fungsi utama sistem dapat berjalan dengan baik sesuai rancangan.

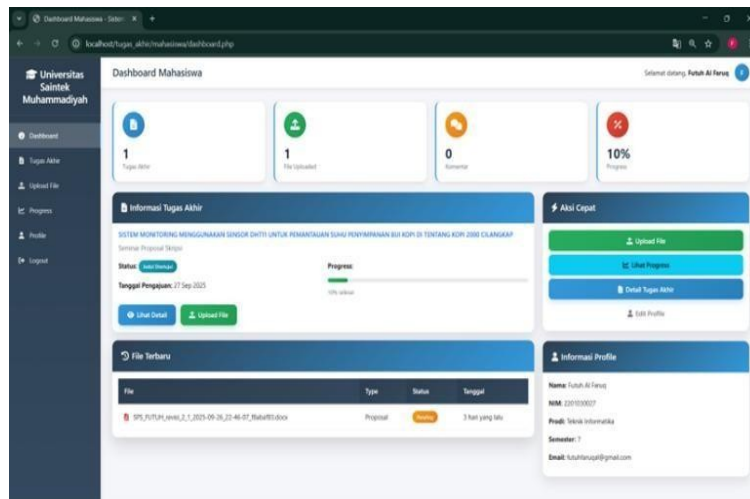
3. HASIL DAN PEMBAHASAN

Halaman *Login* pada Gambar 7 menunjukkan penerapan autentikasi pengguna sebagai pintu masuk utama sistem. Fitur ini menjadi dasar pembagian hak akses antara mahasiswa, dosen, dan admin, sehingga setiap pengguna hanya dapat mengakses fitur sesuai perannya masing-masing.



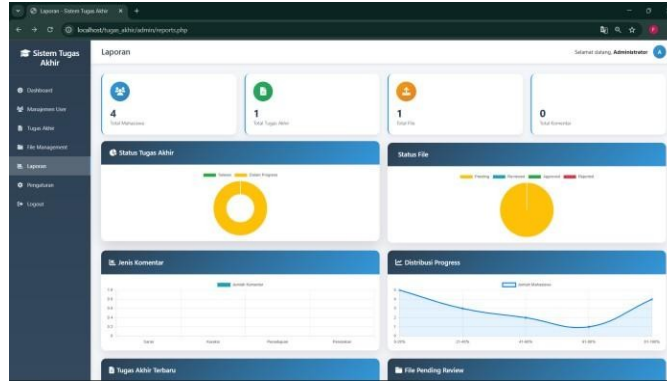
Gambar 7 Login

Halaman mahasiswa pada Gambar 8 menunjukkan bahwa proses pengumpulan tugas akhir yang sebelumnya dilakukan secara manual dapat dialihkan ke sistem berbasis web. Mahasiswa dapat mengunggah dokumen, melihat progres, serta menerima komentar dosen secara terpusat, sehingga riwayat pengumpulan dan bimbingan lebih mudah ditelusuri.



Gambar 8. Halaman Mahasiswa

Halaman monitoring admin pada Gambar 9 berfungsi untuk membantu pihak administrasi memantau status pengumpulan tugas akhir mahasiswa. Dengan adanya informasi jumlah dokumen, status progres, dan data mahasiswa dalam satu dashboard, proses pengawasan menjadi lebih terstruktur dibandingkan pengecekan manual.



Gambar 9. Monitoring Admin

Hasil implementasi sistem ini menunjukkan bahwa proses pengumpulan dan validasi tugas akhir dapat dilakukan secara lebih terpusat dibandingkan proses manual. Jika dibandingkan dengan penelitian sistem informasi akademik berbasis web yang menekankan peningkatan efektivitas dan efisiensi pengelolaan akademik [2], sistem yang dikembangkan pada penelitian ini memiliki fokus yang lebih spesifik, yaitu pengumpulan dokumen tugas akhir, validasi dosen pembimbing, monitoring status, dan laporan administrasi. Sistem ini juga memiliki kesamaan dengan penelitian sistem akademik berbasis web yang memanfaatkan teknologi web untuk mengurangi ketergantungan terhadap pencatatan manual [3], namun ruang lingkupnya diarahkan pada pengelolaan dokumen tugas akhir mahasiswa.

Dari sisi teknologi, penggunaan *PHP Native* dan *MySQL* memberikan kelebihan berupa pengembangan yang sederhana, ringan, dan mudah diterapkan pada kebutuhan sistem berskala kecil hingga menengah. Hal ini sejalan dengan pemanfaatan *PHP Native* dan *MySQL* pada penelitian perancangan sistem berbasis web [6]. Namun, jika dibandingkan dengan pengembangan menggunakan *framework* modern seperti *Laravel* atau *CodeIgniter*, pendekatan *PHP Native* memiliki keterbatasan pada struktur kode, keamanan bawaan, pengelolaan *routing*, serta kemudahan pemeliharaan sistem. Oleh karena itu, sistem ini sudah cukup sesuai untuk kebutuhan digitalisasi awal, tetapi masih dapat dikembangkan lebih lanjut menggunakan *framework* agar struktur aplikasi lebih modular dan mudah dikembangkan.

Tabel 4. Perbandingan Proses Bisnis *As-Is* dan *To-Be*

Proses	Sebelum Sistem (<i>As-Is</i>)	Setelah Sistem (<i>To-Be</i>)	Dampak
Pengumpulan tugas akhir	Mahasiswa menyerahkan dokumen secara langsung atau melalui media terpisah.	Mahasiswa mengunggah dokumen melalui sistem berbasis web.	Pengumpulan lebih terpusat dan terdokumentasi.
Validasi dosen pembimbing	Validasi dilakukan secara manual sehingga status dokumen sulit dipantau.	Dosen memberikan komentar dan status validasi melalui sistem.	Proses validasi lebih jelas, cepat, dan mudah dilacak.
Penyimpanan dokumen	Dokumen berisiko hilang, rusak, atau tercampur dengan berkas lain.	Dokumen tersimpan pada direktori server dan datanya tercatat dalam basis data.	Risiko kehilangan dan duplikasi data berkurang.
Monitoring admin	Admin harus mengecek data mahasiswa dan dokumen satu per satu.	Admin dapat melihat status pengumpulan melalui halaman monitoring.	Rekapitulasi menjadi lebih efisien dan terintegrasi.

Berdasarkan Tabel 4, efisiensi sistem terlihat pada perubahan alur kerja dari proses manual menjadi proses digital yang terintegrasi. Sistem tidak hanya mempersingkat proses pengumpulan, tetapi juga mengurangi risiko kehilangan dokumen, kerusakan berkas, serta duplikasi data sebagaimana permasalahan yang telah dijelaskan pada bagian pendahuluan. Selain itu, fitur monitoring membantu admin melihat perkembangan pengumpulan tugas akhir tanpa harus memeriksa dokumen secara terpisah.

Penerapan metode *Waterfall* pada pengembangan sistem ini dinilai sesuai karena kebutuhan sistem telah dapat diidentifikasi sejak awal, yaitu kebutuhan mahasiswa untuk mengunggah dokumen, kebutuhan dosen untuk melakukan validasi, dan kebutuhan admin untuk melakukan monitoring serta pembuatan laporan. Tahap analisis digunakan untuk menentukan kebutuhan fungsional dan nonfungsional, tahap desain diwujudkan melalui use case diagram, ERD, dan *wireframe*, tahap pengkodean dilakukan menggunakan *PHP Native* dan *MySQL*, sedangkan tahap pengujian dilakukan dengan *black-box testing* untuk memastikan fungsi utama berjalan sesuai rancangan.

Model *Waterfall* juga membantu proses pengembangan menjadi lebih terarah karena setiap tahap dikerjakan secara berurutan. Namun, tantangan dari metode ini adalah kurang fleksibel apabila terdapat perubahan kebutuhan setelah proses desain atau pengkodean berjalan. Perubahan pada fitur, struktur basis data, atau alur validasi dapat memerlukan penyesuaian ulang pada tahap sebelumnya. Meskipun demikian, untuk sistem dengan kebutuhan yang relatif jelas seperti sistem pengumpulan tugas akhir, model *Waterfall* masih dapat digunakan secara efektif.

Keandalan validasi sistem didukung oleh pembagian hak akses berbasis role. Mahasiswa hanya dapat mengunggah dokumen, melihat status, dan membaca komentar dari dosen pembimbing. Dosen memiliki hak untuk memeriksa dokumen, memberikan catatan, serta menentukan status validasi seperti diterima, revisi, atau ditolak. Admin berperan dalam mengelola data mahasiswa, memantau proses pengumpulan, dan menghasilkan laporan. Dengan mekanisme ini, mahasiswa tidak dapat memanipulasi status validasi karena perubahan status hanya dapat dilakukan oleh pengguna yang memiliki hak akses sebagai dosen atau admin.

Dari sisi keamanan dan pengelolaan data, integrasi basis data *MySQL* membantu menghubungkan data pengguna, data mahasiswa, data tugas akhir, komentar, dan status validasi secara terstruktur. Dokumen yang diunggah disimpan pada direktori *server*, sedangkan informasi *file* seperti nama dokumen, pemilik dokumen, tanggal unggah, dan status validasi dicatat pada basis data. Mekanisme ini membuat dokumen lebih mudah ditelusuri, meminimalkan risiko duplikasi, serta mendukung proses administrasi yang lebih rapi dan terdokumentasi.

4. KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan hasil implementasi, sistem informasi pengumpulan tugas akhir mahasiswa mampu mendigitalisasi proses pengajuan judul, unggah dokumen, validasi dosen pembimbing, monitoring admin, dan pelaporan dalam satu sistem terintegrasi. Penerapan metode *waterfall* membantu proses pengembangan berjalan secara terstruktur mulai dari analisis kebutuhan, desain sistem, pengkodean, hingga pengujian *black-box*. Sistem ini juga mendukung pembagian hak akses berdasarkan role pengguna sehingga proses validasi lebih terkontrol, terdokumentasi, dan mengurangi risiko kehilangan maupun duplikasi dokumen.

Saran

Pengembangan sistem informasi ini selanjutnya diharapkan dapat ditambahkan notifikasi untuk mahasiswa atau dosen ke email, sehingga lebih mudah dan cepat untuk cek progress terbaru.

DAFTAR PUSTAKA

- [1] S. Sandfreni, M. B. Ulum, and A. H. Azizah, "ANALISIS PERANCANGAN SISTEM INFORMASI PUSAT STUDI PADA FAKULTAS ILMU KOMPUTER UNIVERSITAS ESA UNGGUL," *Sebatik*, vol. 25, no. 2, pp. 345–356, Dec. 2021, doi: 10.46984/sebatik.v25i2.1587.
- [2] A. A. Zulfa and O. Arifudin, "PERAN SISTEM INFORMASI AKADEMIK BERBASIS WEB DALAM UPAYA MENINGKATKAN EFEKTIVITAS DAN EFISIENSI PENGELOLAAN AKADEMIK DI PERGURUAN TINGGI," 2025.
- [3] Z. Hakim and P. Meilina, "SISTEM INFORMASI AKADEMIK BERBASIS WEBISTE (STUDI KASUS : SMPIT AVICENNA)," 2022. [Online]. Available: <https://jurnal.umj.ac.id/index.php/just-it/index>
- [4] T. I. Widyawan *et al.*, "PENGEMBANGAN SISTEM INFORMASI AKADEMIK BERBASIS WEB UNTUK EFISIENSI PENILAIAN SEKOLAH DEVELOPMENT OF A WEB-BASED ACADEMIC INFORMATION SYSTEM FOR EFFICIENT SCHOOL ASSESSMENT," vol. 9, no. 1, pp. 134–142.
- [5] Universitas Saintek Muhammadiyah, "Visi, Misi, dan Tujuan SaintekMu." [Online]. Available: <https://saintekmu.ac.id/page/profil/visi-misi-dan-tujuan-saintekmu>
- [6] Y. H. Maulana, N. Nugroho, and G. U. Pratama, "Perancangan Sistem Ujian Online Berbasis Web Menggunakan PHP Native dan Mysql," *RIGGS J. Artif. Intell. Digit. Bus.*, vol. 4, no. 3, pp. 53–61, 2025, doi: 10.31004/riggs.v4i3.1970.
- [7] B. J. Bagiono and R. C. W. Rizaq, "Sistem Pendukung Keputusan Dengan Metode Activity Based Costing (Abc) Untuk Penetapan Harga Produk Pada Pt. Sisma Duta Pertiwi Di Bekasi Utara," *J. Sibernetika*, vol. 10, no. 1, pp. 1–12, 2025, doi: 10.56459/sibernetika.v10i1.186.
- [8] T. S. Widodo, Y. Prihati, and J. Gondohanindijo, "Implementasi Metode Waterfall Pada Sistem Informasi Inventaris Barang Berbasis Web Di Hotel Grand Edge Semarang," *INTECOMS J. Inf. Technol. Comput. Sci.*, vol. 7, no. 5, pp. 1520–1528, 2024, doi: 10.31539/intecom.v7i5.11842.
- [9] A. Manajemen and I. Dan, "AKADEMI MANAJEMEN INFORMATIKA DAN KOMPUTER JI. Kelud Raya No. 19 Telp. 024 - 8310002 Semarang," vol. XX, no. 19, 2024.
- [10] A. M. Raya, "Penerapan Model Waterfall Pada Sistem Informasi Beasiswa Berbasis Web," *Jursima*, vol. 9, no. 1, pp. 82–88, 2021, doi: 10.47024/jrs.v9i1.245.
- [11] N. R. D. Zahra *et al.*, "Pengujian pada Website Smartpetscare untuk Layanan Grooming Hewan Menggunakan Metode Black Box Testing," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 9, no. 1, pp. 378–383, 2025.