

## IMPLEMENTASI METODE WATERFALL DALAM PENGEMBANGAN SISTEM INFORMASI REKRUTMEN ASISTEN LABORATORIUM ILMU KOMPUTER BERBASIS WEB

Kaka Davi Dharmawan<sup>1</sup>, Dodyk Fahlome<sup>2</sup>, Said Arrahman<sup>3</sup>, Dea Alya<sup>4</sup>, Nazwa Aliya Muthmainnah Hasibuan<sup>5</sup>,  
Naina Nazwa Hasibuan<sup>6</sup>, Milawati<sup>7</sup>, Tiara Bela Harahap<sup>8</sup>, Salsabila Mahfuza<sup>9</sup>, Ilka Zufria<sup>10</sup>

<sup>1,2,3,4,5,6,7,8,9,10</sup> Ilmu Komputer, Fakultas Sains dan Teknologi, Universitas Islam Negeri Sumatera Utara

<sup>1</sup>kakadavidarmawan@gmail.com, <sup>2</sup>dodyk0701232093@uinsu.ac.id, <sup>3</sup>saidarrahan8@gmail.com, <sup>4</sup>deaalya675@gmail.com,  
<sup>5</sup>nazwahasibuan098@gmail.com, <sup>6</sup>nainanazwa212@gmail.com, <sup>7</sup>milawati2023new@gmail.com, <sup>8</sup>tiarabelahrp@gmail.com,  
<sup>9</sup>salsabilamahfuza@gmail.com, <sup>10</sup>ilkazufria@uinsu.ac.id

### ABSTRAK

Perkembangan teknologi informasi mendorong institusi pendidikan untuk mengadopsi sistem administrasi yang lebih efisien dan terintegrasi. Proses rekrutmen asisten laboratorium di Program Studi Ilmu Komputer sebelumnya dilakukan secara manual melalui Google Form dan via WhatsApp, yang mengakibatkan data yang tidak teratur, sulitnya mengetahui status seleksi, serta keterlambatan rekapitulasi laporan akhir. Penelitian ini bertujuan merancang dan membangun sistem informasi rekrutmen asisten laboratorium berbasis web menggunakan metode *Waterfall*. Hasil implementasi menunjukkan bahwa sistem mampu meningkatkan efisiensi pengelolaan rekrutmen dari sisi admin sebesar 66,67%, dengan waktu rekapitulasi nilai berkurang dari rata-rata tiga hari menjadi satu hari, serta mengurangi beban kerja administratif secara signifikan melalui otomatisasi proses. Dari sisi pendaftar, efisiensi proses pendaftaran dan pelacakan status meningkat sekitar 50% dibanding metode sebelumnya, berkat integrasi fitur pendaftaran akun, unggah dokumen, dan pemantauan status secara *real-time*. Sistem ini terbukti mempercepat alur seleksi, meminimalkan risiko duplikasi data, serta meningkatkan transparansi informasi bagi semua pihak yang terlibat.

**Kata Kunci—** Sistem Informasi, Rekrutmen, Metode Waterfall, Berbasis Web, Efisiensi Administrasi

### ABSTRACT

*Advances in information technology have prompted educational institutions to adopt more efficient and integrated administrative systems. The recruitment process for laboratory assistants in the Computer Science Study Program was previously conducted manually via Google Forms and WhatsApp, resulting in disorganized data, difficulty in determining selection status, and delays in compiling final reports. This study aims to design and develop a web-based laboratory assistant recruitment information system using the Waterfall method. The implementation results show that the system can improve recruitment management efficiency from the administrative side by 66.67%, with the time required for score compilation reduced from an average of three days to one day, and significantly reducing administrative workload through process automation. From the applicants' perspective, the efficiency of the registration and status tracking processes improved by approximately 50% compared to the previous method, thanks to the integration of account registration, document upload, and real-time status monitoring features. This system has proven to accelerate the selection process, minimize the risk of data duplication, and increase transparency of information for all parties involved.*

**Keywords—** Information System, Recruitment, Waterfall Methodology, Web-Based, Administrative Efficiency

## I. PENDAHULUAN

Perkembangan teknologi informasi yang pesat mendorong institusi pendidikan tinggi untuk mengadopsi sistem administrasi yang lebih efisien, terintegrasi, dan berbasis digital. Salah satu proses penting yang memerlukan transformasi digital adalah rekrutmen asisten laboratorium, terutama di laboratorium komputer yang mendukung kegiatan praktikum mahasiswa.

Selama ini, proses seleksi asisten laboratorium masih dilakukan secara manual dengan bantuan Google Form dan komunikasi informal seperti WhatsApp. Metode ini menimbulkan berbagai permasalahan, antara lain data yang tidak terintegrasi, proses seleksi yang lambat dan rawan kesalahan, serta kurangnya jaminan keamanan dan transparansi informasi dalam administrasi pada proses rekrutmen asisten laboratorium tersebut.

Menanggapi permasalahan tersebut, penelitian ini bertujuan untuk merancang dan mengembangkan sistem informasi rekrutmen asisten laboratorium berbasis *web*. Sistem ini diharapkan mampu mengelola seluruh tahapan seleksi secara digital mulai dari pendaftaran, unggah dokumen, jadwal wawancara, hingga pengumuman hasil akhir secara lebih efisien dan aman.

Metodologi pengembangan yang digunakan adalah model *Waterfall*, yang mencakup tahapan analisis kebutuhan, perancangan sistem, implementasi, pengujian, hingga pemeliharaan. Pemilihan metode ini didasarkan pada karakteristik alur rekrutmen yang bersifat tetap dan tidak banyak mengalami perubahan selama proses berlangsung.

Dengan hadirnya sistem ini, diharapkan proses rekrutmen menjadi lebih tertata, transparan, dan mendukung upaya digitalisasi administrasi akademik secara berkelanjutan.

## II. METODE PENELITIAN

Siklus Hidup Pengembangan Sistem (*Systems Development Life Cycle/SDLC*) merupakan sebuah pendekatan yang digunakan dalam rekayasa sistem dan perangkat lunak untuk merancang, membangun, serta memodifikasi sistem informasi. Konsep ini mengacu pada langkah-langkah sistematis yang dijalankan dalam proses pengembangan perangkat lunak, terutama dalam konteks sistem komputer atau sistem informasi. SDLC mencakup sejumlah tahapan utama, yakni perencanaan, analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan, yang dijalankan secara terstruktur guna memastikan sistem yang dihasilkan memenuhi tujuan dan fungsionalitas yang diharapkan.

Metode *Waterfall*, atau yang dikenal sebagai model sekuensial linear, merupakan salah satu pendekatan klasik dalam pengembangan perangkat lunak. Model ini menerapkan tahapan-tahapan yang dilakukan secara bertahap dan berurutan, dimulai dari identifikasi kebutuhan hingga tahap pemeliharaan sistem. Pendekatan ini mengilustrasikan proses yang sistematis, di mana setiap tahapan harus diselesaikan sebelum melanjutkan ke tahapan berikutnya.

Tahapan-tahapan utama dalam model ini meliputi:

### 1. Rekayasa Sistem dan Analisis

Pada tahap ini, sistem dianalisis secara menyeluruh, termasuk elemen-elemen pendukungnya seperti perangkat keras, pengguna, serta interaksinya dengan perangkat lunak. Tujuan utamanya adalah mengidentifikasi kebutuhan umum sistem sebelum difokuskan ke bagian perangkat lunak secara khusus.

### 2. Analisis Kebutuhan Perangkat Lunak

Fokus pada tahap ini adalah merinci kebutuhan perangkat lunak secara spesifik, mencakup informasi domain, fungsi sistem, performa yang diharapkan, serta antarmuka pengguna. Hasil analisis ini kemudian dituangkan dalam dokumentasi yang harus dikonfirmasi bersama pihak pengguna.

### 3. Perancangan (*Design*)

Tahapan ini mengubah hasil analisis menjadi bentuk rancangan teknis yang terdiri atas struktur data, arsitektur sistem, prosedur rinci, dan elemen antarmuka. Desain harus terdokumentasi dengan baik karena akan menjadi acuan utama dalam proses implementasi.

### 4. Implementasi (*Coding*)

Perancangan yang telah disusun diterjemahkan ke dalam bentuk kode menggunakan bahasa pemrograman tertentu. Proses ini menghasilkan perangkat lunak dalam bentuk yang dapat dijalankan oleh mesin.

### 5. Pengujian (*Testing*)

Setelah proses implementasi selesai, sistem diuji untuk memastikan fungsionalitas berjalan sesuai dengan kebutuhan. Pengujian dilakukan baik terhadap logika internal maupun terhadap hasil keluaran agar kesalahan dapat ditemukan dan diperbaiki sejak dini.

### 6. Pemeliharaan (*Maintenance*)

Tahapan ini dilakukan setelah sistem mulai digunakan secara operasional. Fokusnya adalah memperbaiki kesalahan yang mungkin muncul, meningkatkan performa, serta menyesuaikan sistem dengan kebutuhan baru yang berkembang di masa depan [1].

Adapun beberapa *tools* yang digunakan yaitu:

1) **Visual Studio Code**: Editor kode yang dibuat oleh Microsoft dan bisa digunakan di Windows, Linux, dan macOS. Aplikasi ini sangat membantu dalam menulis kode karena mendukung banyak bahasa pemrograman. Salah satu fitur menariknya adalah pewarnaan sintaks yang membuat struktur kode lebih mudah dibaca dan dipahami. Selain itu, VS Code juga mendukung penambahan ekstensi, jadi pengguna bisa menambahkan fitur-fitur baru sesuai kebutuhan. VS Code bersifat *open source*, artinya siapa pun bisa melihat, menggunakan, atau ikut mengembangkan kode sumbernya. Kode ini tersedia di GitHub dan didukung oleh komunitas yang aktif [2].

2) **XAMPP**: Merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan menginstall XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi *web server* Apache, PHP dan MySQL secara manual. XAMPP akan menginstallasi dan mengkonfigurasi secara otomatis atau auto konfigurasi [3].

3) **Figma**: Figma adalah salah satu *design tool* berbasis *cloud* gratis yang bisa dijalankan di browser (*web based*) atau aplikasi *desktop* di OS Windows dan MAC OS yang mirip dengan Sketch atau Adobe XD untuk fungsionalitas dan fiturnya, namun memiliki perbedaan besar yang membuat Figma lebih baik yaitu fitur untuk kolaborasi tim. Figma memberi pengguna semua alat yang dibutuhkan untuk tahap desain proyek, termasuk alat vektor yang mampu membuat ilustrasi sepenuhnya, serta kemampuan *prototyping*, dan pembuatan kode untuk *hand-off* [4].

4) **Draw.io**: Merupakan sebuah situs web yang dirancang untuk memfasilitasi pembuatan diagram UML secara daring. Platform ini memiliki antarmuka yang responsif dan mendukung integrasi dengan layanan penyimpanan Google Drive, sehingga memudahkan pengguna dalam menyimpan serta mengakses diagram yang telah dibuat. Dengan fitur-fitur tersebut, Draw.io menjadi salah satu alternatif yang efisien dalam pembuatan diagram UML [5].

5) **PHPMyAdmin**: Merupakan aplikasi berbasis web yang bersifat *open source* dan dikembangkan menggunakan bahasa pemrograman PHP. Aplikasi ini digunakan untuk mempermudah proses administrasi database MySQL, baik melalui jaringan lokal maupun internet. phpMyAdmin mendukung berbagai fungsi manajemen MySQL, seperti pengelolaan basis data, tabel, atribut (*fields*), relasi, indeks, pengguna, serta pengaturan hak akses (*permissions*) dan lainnya [6].

Analisis PIECES digunakan untuk melakukan identifikasi permasalahan dalam sistem lama dari enam aspek utama, yaitu *Performance*, *Information*, *Economics*, *Control*, *Efficiency*, dan *Service*. Hasil analisis ini menjadi dasar dalam merancang solusi sistem yang lebih baik dan efektif sehingga proses rekrutmen terintegrasi dengan baik.

Tabel 1. Analisis PIECES

Aspek	Masalah pada Sistem Lama	Solusi Sistem Baru
<i>Performance</i>	Waktu proses seleksi memerlukan waktu lama karena input data dilakukan secara manual.	Sistem baru mempercepat proses input dan seleksi dengan pengolahan data otomatis.
<i>Information</i>	Data pendaftar tersebar di file/formulir digital (Google Form).	Data disimpan dalam satu sistem terpusat dan tersusun rapi di database.
<i>Economic</i>	Pengelolaan administrasi membutuhkan banyak waktu dan tenaga admin.	Penghematan tenaga karena data ter-input otomatis dan sistem menyediakan rekapitulasi.
<i>Control</i>	Tidak ada kontrol validasi dokumen pendaftar dan data mudah tertukar.	Terdapat perbedaan hak akses yakni <i>login</i> admin untuk verifikasi serta upload dokumen langsung via sistem.
<i>Efficiency</i>	Komunikasi hasil seleksi dan jadwal tidak efisien karena lewat grup WhatsApp.	Semua informasi ditampilkan langsung melalui <i>dashboard</i> sistem.
<i>Service</i>	Mahasiswa tidak dapat memantau status secara mandiri.	Mahasiswa bisa melihat status lulus, tidak lulus, atau menunggu langsung di sistem.

### III. HASIL DAN PEMBAHASAN

#### A. Analisis PIECES

#### B. Analisis Kebutuhan Fungsional

Kebutuhan fungsional menggambarkan layanan utama yang harus disediakan oleh sistem, seperti

Tabel 2. Analisis kebutuhan fungsional

```
graph TD; A[Pendaftar/Calon Aslab] -- "1.1 Data akun pendaftar" --> B[Sistem Informasi Rekrutmen Asisten Lab Kampus]; B -- "1.2 Formulir pendaftaran/Data pendaftaran" --> A; B -- "1.3 Status seleksi & jadwal wawancara" --> A; B -- "1.4 Hasil wawancara" --> A; B -- "2.1 List data pendaftar" --> C[Admin/Laboran]; C -- "2.2 Data hasil seleksi" --> B; B -- "3.1 Laporan Hasil Rekrutmen" --> D[Kepala Laboratorium];
```

The flowchart illustrates the recruitment process for Aslab. It features a central box labeled "Sistem Informasi Rekrutmen Asisten Lab Kampus".

- Top Interaction:** "Pendaftar/Calon Aslab" interacts with the system via:
  - 1.1 Data akun pendaftar (downward arrow)
  - 1.2 Formulir pendaftaran/Data pendaftaran (upward arrow)
  - 1.3 Status seleksi & jadwal wawancara (upward arrow)
  - 1.4 Hasil wawancara (upward arrow)
- Bottom Interaction:** "Admin/Laboran" interacts with the system via:
  - 2.1 List data pendaftar (downward arrow)
  - 2.2 Data hasil seleksi (upward arrow)
- Right Interaction:** "Kepala Laboratorium" receives data from the system via:
  - 3.1 Laporan Hasil Rekrutmen (leftward arrow)

Tabel 3. Analisis kebutuhan non-fungsional

```

    usecaseDiagram
        actor CalonAsaat as Calon Asaat
        actor AdminLaboran as Admin/Laboran
        actor KepalaLaboratorium as Kepala Laboratorium

        usecase Registrasi
        usecase LoginKeSistem as Login ke Sistem
        usecase MengisiData as Mengisi data diri & upload berkas
        usecase MelihatStatusKeputusan as Melihat status keputusan berkas
        usecase MelihatJadwalWawancara as Melihat jadwal wawancara
        usecase MelihatStatusSeluruh as Melihat status seluruh

        usecase LoginSebagaiAdmin as Login sebagai admin
        usecase MelihatDataPendftar as Melihat data pendftar
        usecase MenilaiTiapPendftar as Menilai tiap pendftar
        usecase MengaturJadwalWawancara as Mengatur jadwal wawancara
        usecase MengunduhLaporan as Mengunduh laporan hasil rekrutmen
        usecase MenerimaLaporan as Menerima laporan hasil rekrutmen

        CalonAsaat --> Registrasi
        CalonAsaat --> LoginKeSistem
        CalonAsaat --> MengisiData
        CalonAsaat --> MelihatStatusKeputusan
        CalonAsaat --> MelihatJadwalWawancara
        CalonAsaat --> MelihatStatusSeluruh

        AdminLaboran --> LoginSebagaiAdmin
        AdminLaboran --> MelihatDataPendftar
        AdminLaboran --> MenilaiTiapPendftar
        AdminLaboran --> MengaturJadwalWawancara
        AdminLaboran --> MengunduhLaporan
        AdminLaboran --> MenerimaLaporan

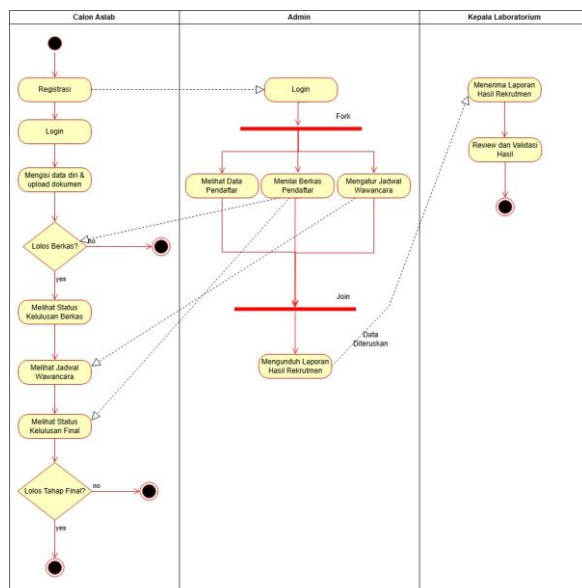
        KepalaLaboratorium --> MenerimaLaporan

        Registrasi --> LoginKeSistem : <<include>>
        LoginKeSistem --> MengisiData : <<include>>
        MengisiData --> MelihatStatusKeputusan : <<include>>
        MelihatStatusKeputusan --> MelihatJadwalWawancara : <<include>>
        MelihatJadwalWawancara --> MengaturJadwalWawancara : <<include>>
        MengaturJadwalWawancara --> MengunduhLaporan : <<include>>
        MengunduhLaporan --> MenerimaLaporan : <<include>>
        MenerimaLaporan --> MengaturJadwalWawancara : <<extend>>
  
```

165

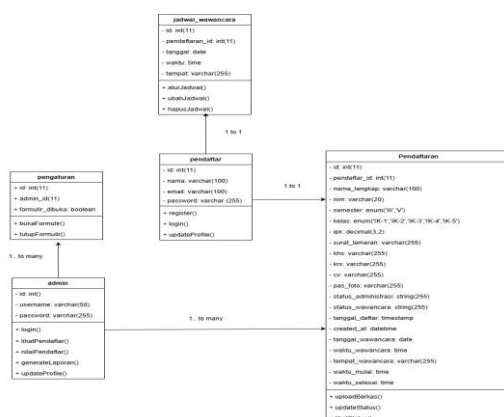


3) **Activity Diagram:** Digunakan memodelkan alur atau proses aktivitas sistem dari awal hingga akhir. Diagram ini menggambarkan tahapan secara menyeluruh yang dilakukan oleh mahasiswa mulai dari registrasi akun, proses login, pengisian formulir pendaftaran dan unggah dokumen, jadwal wawancara hingga menerima hasil seleksi. Di sisi lain, admin memverifikasi data, menjadwalkan wawancara, dan menetapkan kelulusan.



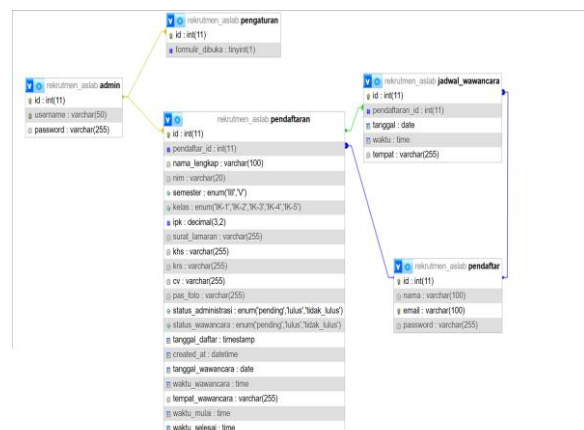
Gambar 3. Activity diagram

4) **Class Diagram:** Menyajikan struktur yang statis sistem berupa objek yang digunakan dalam sistem beserta atribut, *method* dan relasi antar kelas. Beberapa kelas utama yang diidentifikasi antara lain: admin, pendaftar, pendaftaran, pengaturan, dan jadwal wawancara. Setiap kelas memiliki atribut dan *method* spesifik, serta terhubung dengan relasi satu ke satu (*one-to-one*) maupun satu ke banyak (*one-to-many*).



Gambar 4. Class diagram

5) **Entity Relationship Diagram:** Menggambarkan model konseptual basis data yang digunakan dalam sistem. Entitas-entitas seperti pendaftaran, admin, pendaftar, pengaturan sampai jadwal wawancara ditampilkan lengkap dengan atributnya serta relasi antar entitas.



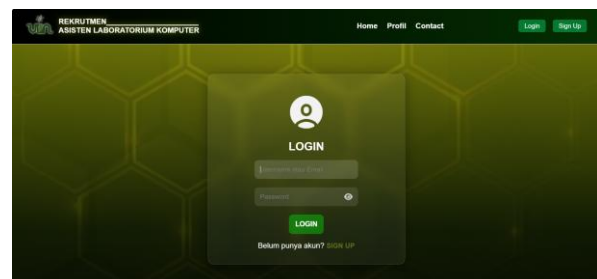
Gambar 5. Entity relationship diagram (ERD)

Setelah selesai membuat analisisnya, lalu dirancang tampilan hasil dari aplikasi yang telah dibuat, rancang bangun aplikasi rekrutmen asisten laboratorium komputer berbasis web. Sehingga hasil implementasinya dapat dilihat sesuai dengan hasil program yang telah dibuat. Dibawah ini akan dijelaskan tiap-tiap tampilan yang ada pada program.

## E. Penerapan Sistem Baru

### 1) Tampilan Menu Login

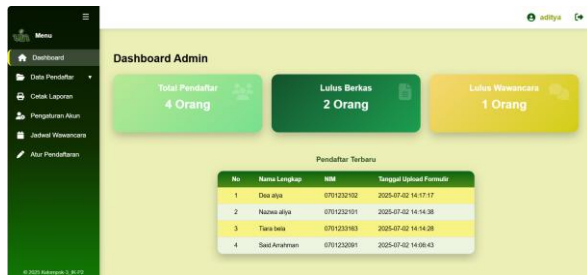
Tampilan Login merupakan tampilan yang muncul setelah meng-klik *button* login dibagian index. Berfungsi sebagai form input username dan password admin program. Gambar tampilan login dapat ditunjukkan pada gambar 1 sebagai berikut :



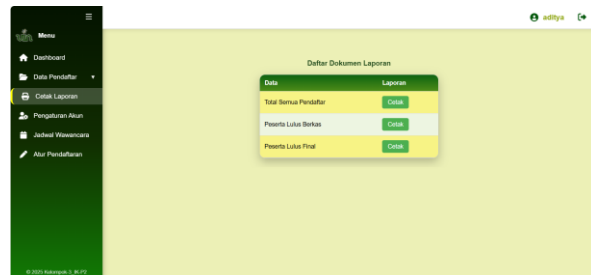
Gambar 6. Tampilan Form Login

### 2) Tampilan Form Dashboard Admin

Form ini menampilkan dashboard untuk admin. Form ini berisi total pendaftar, total lulus berkas, total lulus wawancara, nama pendaftar terbaru, dan sidebar. Gambar tampilan form dashboard admin dapat dilihat pada gambar 2 sebagai berikut :



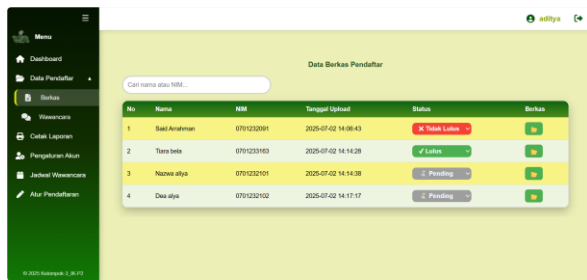
Gambar 7. Tampilan Form Dashboard Admin



Gambar 10. Tampilan Form Cetak Laporan

### 3) Tampilan Form Data Berkas Pendaftar

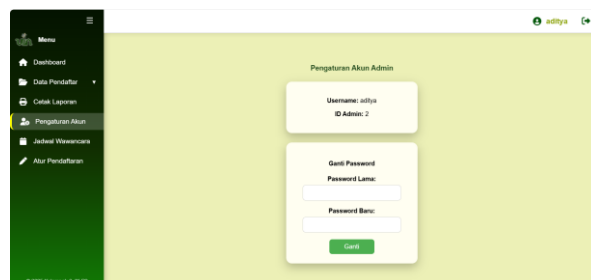
Form ini menampilkan semua data para pendaftar, admin dapat melihat data para pendaftar dan mengubah status antara lulus dan tidak lulus untuk pendaftar. Gambar tampilan form data berkas pendaftar dapat dilihat pada gambar 3 sebagai berikut :



Gambar 8. Tampilan Form Data Berkas Pendaftar

### 6) Tampilan Form Pengaturan Akun

Form ini menampilkan pengaturan untuk mengubah password admin dan di bagian ini juga admin dapat mengubah password pendaftar jika pendaftar lupa password. Gambar tampilan form pengaturan akun dapat dilihat pada gambar 6 dan 7 sebagai berikut :



Gambar 11. Tampilan Form Pengaturan Akun (Admin)

### 4) Tampilan Form Data Wawancara Pendaftar

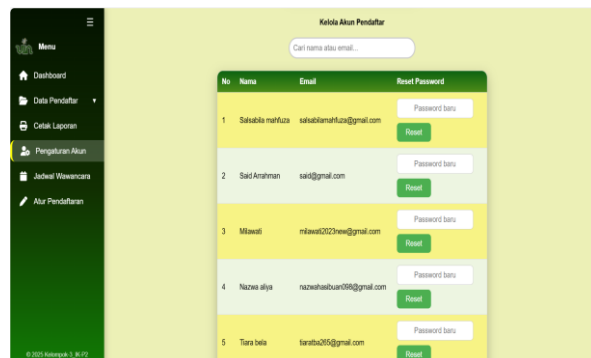
Form ini menampilkan data pendaftar yang telah lulus berkas dan akan lanjut ke fase wawancara, admin dapat mengubah status pendaftar antara lulus dan tidak lulus. Gambar tampilan form data wawancara pendaftar dapat dilihat pada gambar 4 sebagai berikut :



Gambar 9. Tampilan Form Data Wawancara Pendaftar

### 5) Tampilan Form Cetak Laporan

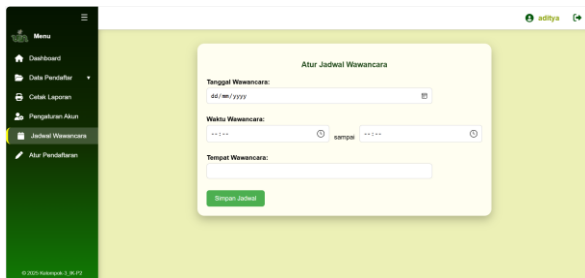
Form ini menampilkan pilihan cetak data untuk total semua pendaftar, total pendaftar lulus berkas, dan total pendaftar lulus wawancara (final). Gambar tampilan form cetak laporan dapat dilihat pada gambar 5 sebagai berikut :



Gambar 12. Tampilan Form Pengaturan Akun (Pendaftar) oleh admin

### 7) Tampilan Form Atur Jadwal Wawancara

Form ini berfungsi untuk admin mengatur jadwal wawancara pendaftar yang telah lulus berkas, admin dapat mengisi tanggal, waktu dan tempat wawancara di laksanakan. Gambar tampilan form atur jadwal wawancara dapat dilihat pada gambar 8 sebagai berikut :



Gambar 13. Tampilan Form Atur Jadwal Wawancara

### 8) Tampilan Form Atur Pendaftaran

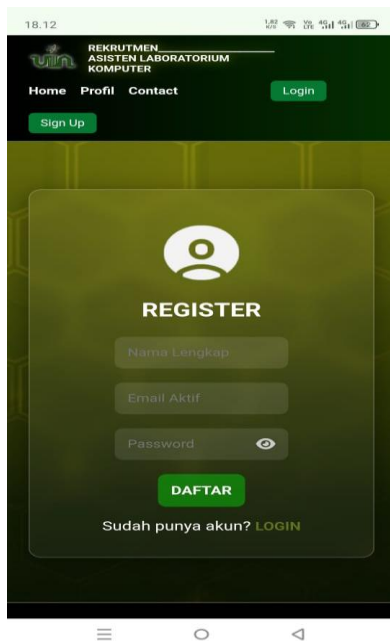
Form ini berfungsi agar admin dapat membuka maupun menutup form pendaftaran (isi formulir). Gambar tampilan form atur pendaftaran dapat dilihat pada gambar 9 sebagai berikut :



Gambar 14. Tampilan Form Atur Pendaftaran

### 9) Tampilan Form Registrasi

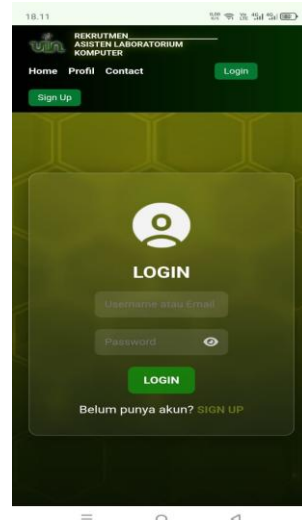
Tampilan ini merupakan tampilan form registrasi untuk pendaftar membuat akun. Gambar tampilan form registrasi ditunjukkan pada gambar 10 sebagai berikut :



Gambar 15. Tampilan Form Registrasi

### 10) Tampilan Form Login

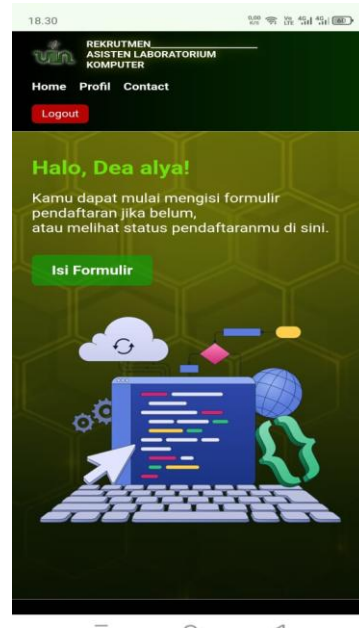
Tampilan Login merupakan tampilan yang muncul setelah meng-klik *button* login dibagian index. Berfungsi sebagai form input email dan password pendaftar. Gambar tampilan form login dapat ditunjukkan pada gambar 11 sebagai berikut :



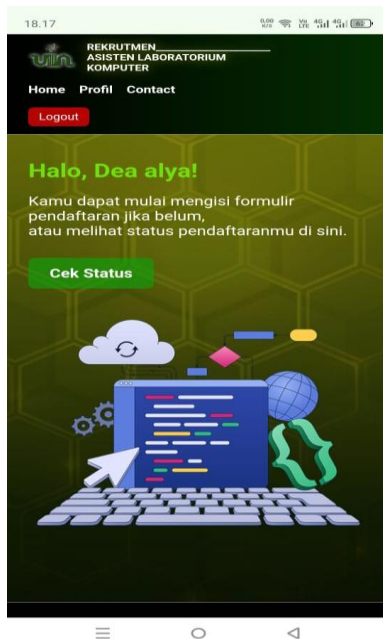
Gambar 16. Tampilan Form Login

### 11) Tampilan Form Dashboard Pendaftar

Form ini menampilkan dashboard untuk pendaftar. Form ini memiliki button untuk pendaftar mengisi formulir pendaftaran, jika pendaftar telah mengisi formulir pendaftaran maka buttonnya akan berubah menjadi cek status. Gambar tampilan form dashboard pendaftar dapat di lihat pada gambar 12 dan 13 sebagai berikut :



Gambar 17. Tampilan Dashboard pendaftar sebelum mengisi formulir pendaftaran



Gambar 18. Tampilan Dashboard pendaftar setelah mengisi formulir pendaftaran

### 12) Tampilan Form Isi Formulir

Form ini digunakan oleh pendaftar untuk menginput data formulir pendaftaran. Data yang di input diantaranya yaitu nama,nim,ipk,surat lamaran,dll dalam bentuk file pdf. Pendaftar tidak dapat menginput data lagi jika admin telah menutup pengisian formulir pendaftaran. Gambar tampilan form isi formulir dapat di lihat pada gambar 14 dan 15 sebagai berikut :



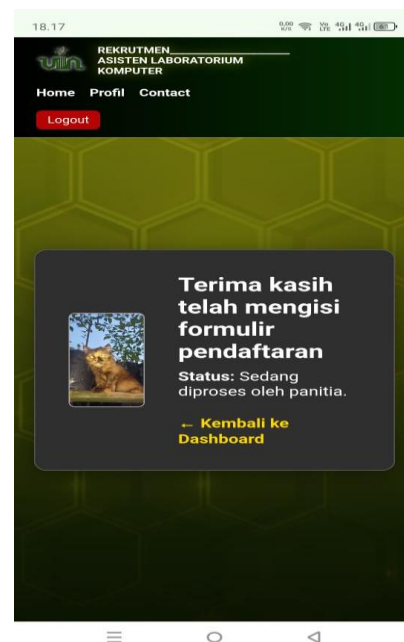
Gambar 19. Tampilan Form Isi Formulir



Gambar 20. Tampilan Form Isi Formulir (Ditutup)

### 13) Tampilan Form Status Administrasi

Form ini menampilkan status administrasi para pendaftar. Dibagian ini ada 3 status yaitu pending,lulus,dan tidak lulus.Di awal pendaftar akan berstatus pending dan pendaftar harus menunggu admin untuk mengecek data/berkas pendaftaran lalu admin akan menentukan lulus atau tidaknya pendaftar. Gambar tampilan form status administrasi dapat di lihat pada gambar 16,17 dan 18 sebagai berikut :

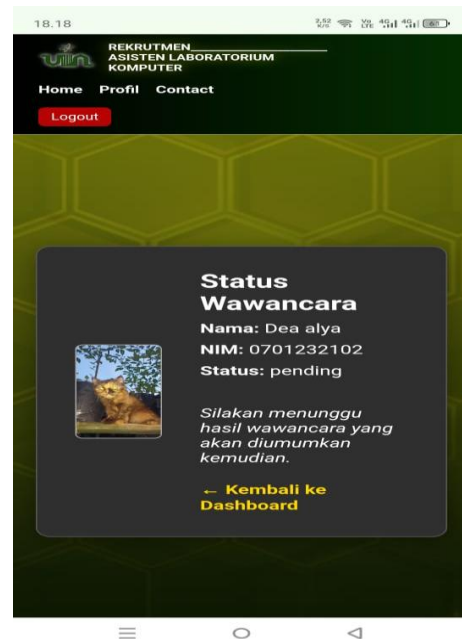


Gambar 21. Tampilan Form Status Administrasi (Pending)





Gambar 22. Tampilan Form Status Administrasi (Lulus)



Gambar 24. Tampilan Form Status Wawancara (Pending)



Gambar 23. Tampilan Form Status Administrasi (Tidak Lulus)



Gambar 25. Tampilan Form Status Wawancara (Lulus)

#### 14) Tampilan Form Status Wawancara

Form ini menampilkan status wawancara para pendaftar. Dibagian ini hanya pendaftar yang lulus administrasi saja yang dapat melihat status wawancara. Sama seperti status administrasi ,di bagian status wawancara juga ada 3 status yaitu pending,lulus,dan tidak lulus. Gambar tampilan form status wawancara dapat di lihat pada gambar 19,20 dan 21 sebagai berikut :



Gambar 26. Tampilan Form Status Wawancara (Tidak Lulus)

## REFERENSI

- [1] Adiya, A. N., Anggraeni, D. L., & Albana, I. (2024). Analisa Perbandingan Penggunaan Metodologi Pengembangan Perangkat Lunak (Waterfall, Prototype, Iterative, Spiral, Rapid Application Development (RAD)). *Merkurius: Jurnal Riset Sistem Informasi dan Teknik Informatika*, 2(4), 122-134.
- [2] Wilyanto, N., Firmando, J., Franko, B., Tanzil, S. P., Tan, H. C., & Hartati, E. (2023). Pembuatan Website Menggunakan Visual Studio Code di SMA Xaverius 3 Palembang. *Fordicate*, 3(1), 1-8.
- [3] Lestanti, S., & Susana, A. D. (2016). Sistem pengarsipan dokumen guru dan pegawai menggunakan Metode Mixture Modelling berbasis Web. *Antivirus: Jurnal Ilmiah Teknik Informatika*, 10(2).
- [4] FIGMA, K. M. M. A. (2022). JSIM: Jurnal Sistem Informasi Mahakarya. *Jurnal Sistem Informasi Mahakarya (JSIM)*, 5(1).
- [5] Marthiawati, N., Kurniawansyah, K., Nugraha, H., & Khairunnisa, F. (2024). Pelatihan Pembuatan UML (Unified Modelling Language) Menggunakan Aplikasi Draw. io Pada Prodi Sistem Informasi Universitas Muhammadiyah Jambi. *Transformasi Masyarakat: Jurnal Inovasi Sosial dan Pengabdian*, 1(2), 25-33.
- [6] Hartiwati, E. N. (2022). Aplikasi Inventori Barang Menggunakan Java Dengan Phpmyadmin. *Cross-border*, 5(1), 601-610.

## KESIMPULAN

Penelitian ini berhasil merancang dan mengimplementasikan sistem informasi rekrutmen asisten laboratorium berbasis web dengan menggunakan metode pengembangan perangkat lunak Waterfall. Proses rekrutmen yang sebelumnya dilakukan secara manual melalui Google Form dan WhatsApp kini telah ditransformasikan menjadi sistem digital yang lebih efisien, terstruktur, dan terintegrasi.

Penerapan metode Waterfall yang terdiri dari tahapan analisis kebutuhan, perancangan, implementasi, pengujian, dan pemeliharaan terbukti efektif dalam mengembangkan sistem yang sesuai dengan kebutuhan pengguna. Sistem ini mendukung fitur registrasi akun, pengisian formulir pendaftaran, unggah dokumen, penjadwalan wawancara, hingga pengumuman hasil seleksi yang dapat diakses secara real-time oleh mahasiswa dan admin.

Hasil implementasi menunjukkan bahwa sistem mampu:

- Meningkatkan efisiensi dan kecepatan proses rekrutmen,
- Memperkuat kontrol dan validasi data,
- Menyediakan transparansi informasi kepada pendaftar,
- Mengurangi beban kerja admin melalui otomatisasi proses administrasi.

Dengan demikian, sistem informasi ini menjadi solusi alternatif yang relevan dan mendukung digitalisasi proses administrasi akademik di lingkungan kampus secara berkelanjutan.