

BAB I PENDAHULUAN

1.1 Latar Belakang

Lulusan Sekolah Menengah Kejuruan (SMK) merupakan salah satu sumber tenaga kerja yang dipersiapkan untuk langsung memasuki dunia industri. Namun pada kenyataannya, tingkat penyerapan tenaga kerja lulusan SMK masih menghadapi tantangan yang cukup besar. Data Badan Pusat Statistik (BPS) menunjukkan bahwa per Februari 2025 tingkat pengangguran terbuka (TPT) lulusan SMK mencapai 9,01%, yang secara persentase masih lebih tinggi dibanding lulusan SMA maupun perguruan tinggi (Badan Pusat Statistik Indonesia, 2025). Selain itu, survei pada Agustus 2024 memperlihatkan bahwa dari total lebih 7,4 juta orang menganggur, lulusan SMK mendominasi angka pengangguran terbuka dengan TPT sekitar 8,62% (Arifa, 2024). Kondisi ini menunjukkan bahwa masih terdapat ketimpangan antara kompetensi yang dihasilkan oleh pendidikan vokasi dengan kebutuhan aktual dunia industri.

Ketidaksesuaian tersebut tidak hanya berkaitan dengan keterampilan teknis, tetapi juga menyangkut minimnya sertifikasi, pelatihan tambahan, serta akses informasi kerja yang relevan. Dalam konteks ini, pemanfaatan sistem informasi yang terstruktur mampu meningkatkan efisiensi dan akurasi dalam proses penyediaan informasi, pengambilan keputusan, hingga layanan yang terkait dengan mobilitas tenaga kerja. Konsep *e-recruitment* menjadi salah satu bentuk implementasi sistem informasi yang memungkinkan integrasi lowongan, skrining kandidat, serta proses pencocokan (*matching*) antara kompetensi pelamar dan kebutuhan pekerjaan secara digital (Marler & Parry, 2016). Implementasi sistem informasi dalam rekrutmen tidak hanya mempercepat proses seleksi, tetapi juga memungkinkan adanya personalisasi rekomendasi berdasarkan kesesuaian kompetensi dan kualifikasi yang dimiliki kandidat.

Dalam lingkup industri besar, perusahaan seperti Telkomsel memiliki kebutuhan yang tinggi terhadap talenta digital yang memiliki keahlian teknis, pemahaman teknologi, serta sertifikasi yang sesuai dengan standar industri. Sebagai bentuk kontribusi terhadap peningkatan kualitas dan penyerapan tenaga kerja vokasi, Telkomsel berinisiatif menyediakan platform digital khusus bagi alumni SMK yang berfungsi sebagai pusat informasi lowongan pekerjaan, pelatihan, dan sertifikasi. Platform ini diharapkan tidak hanya menjadi sarana pencarian kerja, tetapi juga wadah ekosistem kolaboratif antara Telkomsel, mitra industri, dan lembaga pelatihan yang dapat menjembatani kesenjangan kompetensi dengan kebutuhan pasar kerja.

Untuk mewujudkan platform yang efektif, diperlukan mekanisme pencocokan yang akurat antara profil alumni SMK dengan lowongan kerja yang tersedia. Setiap lulusan SMK perlu memperoleh rekomendasi lowongan yang sesuai dengan profil kompetensi dan sertifikasi yang dimilikinya, sementara perusahaan dapat menjangkau calon tenaga kerja secara lebih cepat, efisien, dan terukur. Dalam konteks ini, sistem rekomendasi berbasis *content-based filtering* telah terbukti efektif dalam domain *e-recruitment* untuk meningkatkan kualitas pencocokan antara pelamar dan lowongan kerja. Penelitian oleh Rahmawati et al. (2025) menunjukkan bahwa penggunaan *TF-IDF* dan *cosine similarity* mampu melakukan pencocokan otomatis antara data perusahaan dan alumni SMK pada sistem rekomendasi Bursa Kerja Khusus (BKK) dengan tingkat

presisi 1.00. Hersianty et al. (2025) juga membuktikan bahwa penerapan algoritma *TF-IDF* dan *cosine similarity* dalam sistem rekomendasi lowongan pekerjaan mencapai akurasi 80% dalam menyandingkan pencari kerja dengan peluang karir yang sesuai dengan kompetensi mereka. Sementara itu, Habibi & Albanna (2022) menerapkan metode *content-based filtering* dengan *word embedding* dan *cosine similarity* untuk menghasilkan rekomendasi pekerjaan berdasarkan profil LinkedIn yang berhasil memberikan rekomendasi pekerjaan yang diurutkan berdasarkan tingkat kemiripan. Temuan-temuan tersebut memperkuat bahwa algoritma berbasis analisis teks dan *similarity* measure memiliki tingkat keandalan yang tinggi untuk menghasilkan rekomendasi pekerjaan yang relevan.

Oleh karena itu, dalam penelitian ini dikembangkan sebuah sistem informasi terintegrasi yang dilengkapi fitur rekomendasi lowongan kerja berbasis *content-based filtering* dengan metode *TF-IDF* dan *cosine similarity*. Sistem ini dirancang untuk membantu alumni SMK dalam menemukan peluang kerja yang relevan, meningkatkan kualitas kompetensi melalui program sertifikasi dan pelatihan yang terarah, serta mendukung Telkomsel dan mitra industri dalam memperoleh talenta yang sesuai dengan kebutuhan transformasi digital saat ini.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, maka muncul rumusan masalah penelitian:

1. Bagaimana merancang sistem informasi terpadu yang menggabungkan data lowongan kerja, sertifikasi, dan pelatihan yang relevan untuk lulusan SMK?
2. Bagaimana mengimplementasikan metode *content-based filtering* dengan *TF-IDF* dan *cosine similarity* untuk mencocokkan profil lulusan SMK dengan lowongan kerja yang disediakan oleh perusahaan?

1.3 Tujuan Penelitian

Tujuan penelitian ini adalah sebagai berikut:

1. Merancang dan membangun sistem informasi berbasis web yang memungkinkan alumni SMK untuk mencari pekerjaan, mengikuti pelatihan, dan memperoleh sertifikasi dalam satu platform digital.
2. Mengimplementasikan fitur sistem rekomendasi lowongan kerja menggunakan pendekatan *content-based filtering* dengan metode *TF-IDF* dan *cosine similarity* guna mencocokkan profil pengguna dengan deskripsi pekerjaan secara otomatis.

1.4 Batasan Masalah

Agar pembahasan dalam penelitian ini tidak melebar dan tetap fokus pada inti permasalahan, maka penelitian ini dibatasi oleh beberapa ruang lingkup sebagai berikut:

1. Cakupan sistem informasi yang dibangun hanya berfokus pada tiga fitur utama, yaitu pencarian lowongan kerja, sertifikasi, dan pelatihan bagi lulusan SMK, dengan tambahan fitur rekomendasi lowongan kerja berbasis profil pengguna.
2. Fitur rekomendasi lowongan kerja dibatasi pada penggunaan algoritma *content-based filtering* dengan metode *TF-IDF* dan *cosine similarity*, tanpa melibatkan pendekatan *Collaborative Filtering* atau *deep learning*.

3. Sistem yang dikembangkan berbasis web dan tidak mencakup implementasi dalam bentuk aplikasi *mobile native*.
4. Pengembangan sistem menggunakan arsitektur terpisah (*separated architecture*) dengan *backend* menggunakan FastAPI *framework* berbasis Python untuk sistem rekomendasi, dan *frontend* menggunakan Laravel *framework* berbasis PHP dengan pendekatan *Model-View-Controller (MVC)*.
5. Pengujian sistem menggunakan metode *black box testing* untuk menguji fungsionalitas dan kesesuaian dengan kebutuhan pengguna.

1.5 Landasan teori

1.5.1 Sistem Informasi

Sistem informasi merupakan suatu kesatuan yang terdiri atas berbagai komponen yang saling berinteraksi dan terintegrasi untuk mengumpulkan, mengelola, menyimpan, serta mendistribusikan data menjadi informasi yang bermanfaat bagi pengambilan keputusan dalam organisasi. Komponen utama sistem informasi meliputi manusia, perangkat keras (*hardware*), perangkat lunak (*software*), jaringan komunikasi, dan prosedur kerja yang berfungsi secara terpadu (Fatimah & Samsudin, 2019). Menurut Henny et al. (2018), sistem informasi adalah sistem di dalam organisasi yang dirancang untuk memenuhi kebutuhan pengolahan transaksi harian, mendukung kegiatan operasional dan manajerial, serta menyediakan laporan-laporan bagi pihak internal maupun eksternal. Dengan demikian, sistem informasi berperan penting dalam menghasilkan informasi yang akurat, relevan, dan tepat waktu guna mendukung proses pengambilan keputusan yang efektif.

Dalam konteks yang lebih luas, manusia hidup dalam lingkungan yang dipenuhi oleh berbagai sistem seperti sistem pendidikan, ekonomi, transportasi, dan bisnis yang semuanya berfungsi melalui keteraturan dan interaksi antar komponennya. Menurut Firman et al. (2016), sistem merupakan sekumpulan elemen yang saling terintegrasi untuk mencapai tujuan tertentu, sedangkan informasi adalah data yang telah diolah menjadi bentuk yang lebih bermakna dan berguna bagi penggunanya. Informasi menjadi elemen vital dalam suatu sistem karena tanpa informasi, sistem tidak dapat berfungsi secara optimal. Oleh karena itu, sistem informasi dibangun untuk mengelola data mentah menjadi informasi yang bernilai, yang selanjutnya digunakan untuk mendukung efisiensi dan efektivitas kegiatan organisasi.

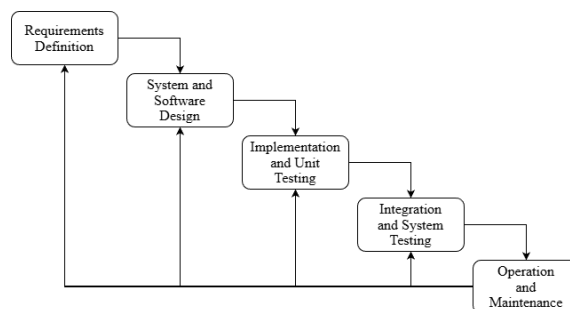
Lebih lanjut, Firman et al. (2016) menjelaskan bahwa sistem informasi merupakan kombinasi teratur dari manusia, perangkat keras, perangkat lunak, jaringan komunikasi, dan sumber daya data yang berfungsi untuk mengumpulkan, mengubah, serta menyebarkan informasi di dalam organisasi. Sistem informasi tidak hanya mendukung kegiatan operasional dan manajerial, tetapi juga menjadi sarana strategis dalam pengembangan organisasi di era digital. Dengan penerapan sistem informasi yang baik, organisasi dapat meningkatkan produktivitas, mempercepat proses pengambilan keputusan, serta menciptakan keunggulan kompetitif dalam menghadapi perkembangan teknologi dan dinamika lingkungan bisnis.

1.5.2 Waterfall

Metode *Waterfall* atau yang sering disebut sebagai *Classic Life Cycle Model* merupakan salah satu model pengembangan perangkat lunak tertua dan paling dikenal dalam bidang *Software Engineering*. Model ini pertama kali diperkenalkan oleh Winston W. Royce pada tahun 1970 dan hingga kini masih banyak digunakan karena pendekatannya yang sistematis, terstruktur, dan mudah dipahami (Wahid, 2020). Secara formal, metode ini dikenal dengan istilah *Linear Sequential Model* karena setiap tahapan dalam proses pengembangannya dilakukan secara berurutan, di mana satu tahap harus diselesaikan terlebih dahulu sebelum melanjutkan ke tahap berikutnya. Model ini mengalir layaknya air terjun (*Waterfall*), dimulai dari tahap awal perencanaan hingga tahap akhir pemeliharaan sistem, tanpa memungkinkan untuk kembali ke tahap sebelumnya setelah tahap tersebut diselesaikan (Wahid Abdul Aceng, 2020)

Menurut Fridayanthie et al., (2016), metode *Waterfall* terdiri dari lima tahapan utama yang mencerminkan alur logis pembangunan perangkat lunak, yaitu:

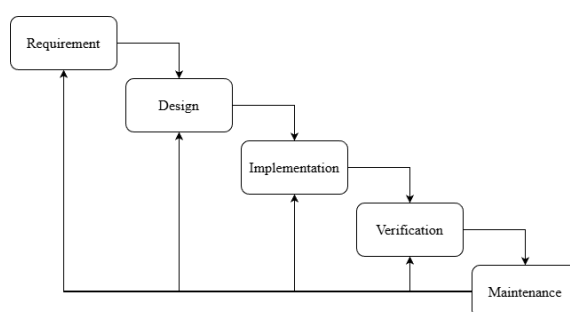
1. *Requirement Analysis and Definition*, yaitu tahap pengumpulan kebutuhan pengguna secara menyeluruh yang kemudian dianalisis untuk menentukan spesifikasi sistem. Proses ini menjadi dasar untuk menghasilkan rancangan perangkat lunak yang sesuai dengan kebutuhan pengguna.
2. *System and Software Design*, tahap perancangan dilakukan berdasarkan kebutuhan yang telah didefinisikan, mencakup perancangan arsitektur sistem, antarmuka, serta komponen perangkat lunak yang akan dibangun.
3. *Implementation and Unit Testing*, tahap ini menerjemahkan rancangan ke dalam kode program menggunakan bahasa pemrograman yang telah ditentukan. Setiap unit program kemudian diuji untuk memastikan bahwa fungsi dasarnya berjalan dengan benar.
4. *Integration and System Testing*, pada tahap ini seluruh unit program yang telah diuji disatukan menjadi satu kesatuan sistem, lalu dilakukan pengujian menyeluruh untuk memastikan bahwa sistem berfungsi sesuai dengan kebutuhan dan bebas dari kesalahan besar.
5. *Operation and Maintenance*, yaitu tahap penerapan sistem ke lingkungan operasional yang sesungguhnya dan dilanjutkan dengan pemeliharaan untuk memperbaiki kesalahan yang belum terdeteksi atau menyesuaikan sistem dengan kebutuhan baru.



Gambar 1. *Waterfall* Menurut Sommerville

Sementara itu, Wahid Abdul Aceng (2020) membagi tahapan *Waterfall* menjadi lima fase utama, yakni:

1. *Requirement*, tahap komunikasi dengan pengguna untuk memahami kebutuhan sistem serta batasan perangkat lunak. Data dikumpulkan melalui wawancara, observasi, atau survei, lalu dianalisis agar sesuai dengan kebutuhan pengguna.
2. *Design*, pengembang membuat desain sistem yang mencakup kebutuhan perangkat keras, arsitektur perangkat lunak, serta antarmuka sistem agar proses implementasi lebih terarah.
3. *Implementation*, sistem dikembangkan dalam bentuk modul-modul kecil (*unit*) yang kemudian diuji secara individu melalui *unit testing*.
4. *Verification*, dilakukan pengujian untuk memastikan sistem telah memenuhi seluruh kebutuhan yang telah ditetapkan, meliputi *unit testing*, *system testing*, dan *acceptance testing*.
5. *Maintenance*, merupakan tahap akhir yang berfokus pada perawatan sistem, termasuk perbaikan kesalahan dan penyesuaian terhadap perubahan kebutuhan pengguna di masa mendatang.



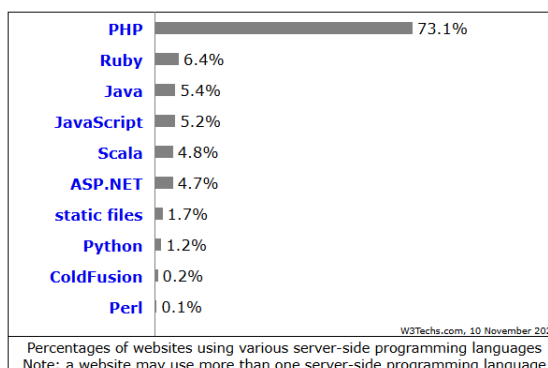
Gambar 2. *Waterfall* Menurut Pressman

Metode *Waterfall* memiliki beberapa keunggulan, di antaranya: (1) menghasilkan sistem dengan kualitas yang baik karena setiap tahap dilakukan secara bertahap dan terstruktur; (2) proses pengembangan dilakukan fase demi fase sehingga meminimalkan potensi kesalahan; serta (3) dokumentasi pengembangan sistem menjadi sangat terorganisir karena setiap tahapan harus diselesaikan secara lengkap sebelum melanjutkan ke tahap berikutnya (Wahid, 2020).

1.5.3 PHP

PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman berbasis *open source* yang dirancang khusus untuk pengembangan web dinamis dan dapat disisipkan secara langsung ke dalam skrip HTML. Bahasa ini termasuk dalam kategori *server-side scripting language*, di mana proses eksekusi program dilakukan di sisi server, kemudian hasilnya dikirim ke klien dalam bentuk halaman web yang telah diolah (Firman et al., 2016). PHP memiliki sintaks yang sederhana dan mudah dipahami, serta mengadopsi beberapa konsep dari bahasa pemrograman lain seperti C, Java, dan Perl. Menurut Firman et al. (2016), PHP merupakan skrip sisi server yang ditambahkan ke dalam HTML untuk

memproses data dan menghasilkan keluaran berupa dokumen web dinamis. Dengan kemampuan ini, PHP menjadi salah satu bahasa pemrograman yang banyak digunakan dalam pembuatan website interaktif dan aplikasi berbasis web.



Gambar 3. Data Penggunaan Bahasa Pemrograman Terbanyak 2025

Berdasarkan data dari w3techs (2025), PHP masih menjadi bahasa pemrograman *server-side* yang paling dominan di dunia dengan tingkat penggunaan mencapai 73,1% dari seluruh situs web yang diketahui menggunakan pemrograman sisi server. Bahasa lain yang juga banyak digunakan meliputi Ruby (6,4%), Java (5,4%), JavaScript (5,2%), dan ASP.NET (4,7%). Data ini menunjukkan bahwa PHP masih mempertahankan posisinya sebagai bahasa utama dalam pengembangan web dinamis, meskipun berbagai bahasa pemrograman modern terus berkembang. Sejak pertama kali diperkenalkan secara resmi pada November 1997, PHP terus mengalami peningkatan kemampuan signifikan dibandingkan versi awalnya yang hanya mampu membuat aplikasi web sederhana (Niarmman et al., 2023).

1.5.4 Laravel Framework

Framework Laravel, yang dikembangkan oleh *Taylor Otwell* pada tahun 2011, merupakan salah satu *framework* PHP paling populer yang menerapkan pola arsitektur *Model-View-Controller (MVC)* untuk memisahkan logika bisnis dari tampilan sehingga mempermudah pengelolaan kode dan meningkatkan skalabilitas aplikasi. Laravel menyediakan berbagai fitur modern seperti *routing*, *middleware*, *Eloquent ORM*, *Blade templating engine*, migrasi basis data, serta sistem autentikasi dan otorisasi yang kuat (Putra et al., 2025). *Framework* ini dirancang dengan prinsip *Don't Repeat Yourself (DRY)* yang menekankan efisiensi dan mengurangi redundansi dalam penulisan kode, serta mendukung pengembangan aplikasi yang aman, cepat, dan mudah dipelihara. Dengan sintaks yang ekspresif dan elegan, Laravel banyak digunakan oleh pengembang untuk membangun aplikasi web yang terstruktur, dinamis, dan responsif (Sinlae et al., 2024).

Selain itu, Laravel juga dikenal karena kemampuannya dalam meningkatkan produktivitas pengembang dengan menyediakan beragam fitur bawaan seperti sistem templating *Blade*, manajemen basis data menggunakan *Eloquent ORM*, serta dukungan autentikasi yang siap digunakan. Dibandingkan dengan *framework* lain seperti *Symfony*,

Laravel lebih menonjol dalam kemudahan dan kecepatan pengembangan, terutama untuk proyek berskala kecil hingga menengah. Sementara itu, *Symfony* lebih unggul dalam fleksibilitas dan ketepatan konfigurasi karena pendekatan berbasis komponen yang cocok untuk proyek berskala besar dan arsitektur kustom (Putra et al., 2025).

1.5.5 Sistem Rekomendasi

Sistem rekomendasi merupakan teknologi yang dirancang untuk memberikan saran atau prediksi informasi yang relevan berdasarkan preferensi, perilaku, atau kebutuhan pengguna. Sistem ini bekerja dengan mengumpulkan, menganalisis, serta memodelkan data pengguna untuk menghasilkan rekomendasi yang sesuai, sehingga mampu membantu pengguna menemukan item, layanan, atau informasi yang paling relevan di tengah banyaknya pilihan yang tersedia. Menurut Aulia et al. (2025), sistem rekomendasi berfungsi sebagai alat bantu dalam proses pengambilan keputusan ketika pengguna tidak memiliki cukup pengetahuan mengenai alternatif yang tersedia. Pendekatan umum dalam pengembangan sistem rekomendasi meliputi *Collaborative Filtering*, *Content-based filtering*, *Demographic Filtering*, serta *Hybrid Filtering*, yang masing-masing menyediakan mekanisme analisis data pengguna dan item dengan logika yang berbeda. Dalam konteks penelitian modern, Aulia et al. (2025) menegaskan bahwa sistem rekomendasi mampu meningkatkan kualitas layanan digital dengan menyajikan konten yang lebih personal serta mengurangi beban pencarian informasi bagi pengguna.

Sistem rekomendasi juga menjadi solusi atas masalah *information overload* yang sering dialami pengguna ketika berhadapan dengan jumlah data atau pilihan yang sangat besar. Sinkaye et al. (2015) menyebutkan bahwa sistem rekomendasi berperan dalam memanfaatkan pengalaman dan preferensi pengguna lain untuk menghasilkan saran yang lebih akurat melalui mekanisme otomatis. Sejalan dengan itu, beberapa metode lanjutan seperti *Artificial Immune Network*, *profitability-aware recommender*, hingga prediksi pergerakan pengguna berbasis *web usage mining* telah dikembangkan untuk meningkatkan kualitas rekomendasi. Penelitian Oyadila et al. (2025) menunjukkan bahwa penggunaan pendekatan berbasis konten dan kolaboratif mampu meningkatkan relevansi rekomendasi dalam sistem informasi modern. Selain itu, studi oleh Isinkaye et al., (2015) menekankan bahwa personalisasi konten merupakan elemen penting dalam efektivitas sistem rekomendasi, karena mampu memberikan pengalaman yang lebih eksklusif dan adaptif sesuai kebutuhan masing-masing pengguna. Dengan demikian, sistem rekomendasi tidak hanya meningkatkan pengalaman pengguna, tetapi juga menjadi komponen penting dalam pengembangan berbagai aplikasi digital seperti *e-commerce*, pendidikan, media, dan layanan berbasis web lainnya.

1.5.6 TF-IDF

TF-IDF merupakan teknik statistik yang digunakan dalam analisis teks untuk menilai tingkat kepentingan suatu kata dalam sebuah dokumen dibandingkan dengan keseluruhan dokumen dalam korpus. Metode ini bekerja dengan menggabungkan dua komponen utama, yaitu *Term Frequency (TF)* yang mengukur frekuensi kemunculan kata dalam dokumen, serta *Inverse Document Frequency (IDF)* yang memberikan penalti pada kata-kata yang sering muncul di banyak dokumen. Pendekatan ini memungkinkan

sistem untuk mengidentifikasi kata-kata yang memiliki nilai informasi tinggi, sekaligus mengurangi pengaruh kata umum yang kurang relevan. Menurut Samosir & Ginting (2024), *TF-IDF* menghasilkan representasi numerik dalam bentuk matriks, di mana setiap baris merepresentasikan dokumen dan setiap kolom menunjukkan bobot suatu kata. Teknik ini banyak digunakan dalam sistem temu kembali informasi, peringkasan teks, hingga sistem rekomendasi konten karena kemampuannya menonjolkan kata-kata unik yang menjadi ciri khas suatu dokumen. Penelitian Fitria et al. (2024) juga menegaskan bahwa *TF-IDF* merupakan metode pembobotan yang efisien untuk mengoptimalkan proses ekstraksi fitur dalam aplikasi berbasis dokumen.

Dalam pengembangan sistem rekomendasi berbasis konten, *TF-IDF* menjadi metode penting untuk merepresentasikan karakteristik dokumen dan mengukur kemiripan antar item. Aulia et al. (2025) menjelaskan bahwa bobot *TF-IDF* yang tinggi menunjukkan bahwa kata tersebut sangat relevan dengan dokumen tertentu. Ketika *TF-IDF* dikombinasikan dengan algoritma *Cosine similarity*, sistem dapat menghitung tingkat kesamaan antar dokumen secara akurat sehingga mampu menghasilkan rekomendasi yang relevan. Penggunaan *TF-IDF* juga didukung dalam penelitian Timur (dalam Oyadila et al., 2025) serta Septiani & Isabela (dalam Oyadila et al., 2025), yang menyatakan bahwa metode ini efektif untuk pemilihan fitur pada berbagai aplikasi *NLP*. (Herimanto et al., 2024) menambahkan bahwa *TF-IDF* memiliki keunggulan pada pengolahan data teks berskala besar karena mampu menurunkan kompleksitas analisis. Selain itu, penelitian Aulia et al. (2025) serta Oyadila et al. (2025) menunjukkan bahwa *TF-IDF* memberikan performa yang stabil dalam sistem rekomendasi modern, terutama ketika digunakan untuk memetakan preferensi pengguna berdasarkan konten yang mereka akses. Dengan fleksibilitas dan ketepatannya, *TF-IDF* tetap menjadi teknik pembobotan teks yang paling banyak digunakan dalam berbagai penelitian dan aplikasi berbasis informasi digital.

1.5.7 *Cosine similarity*

Cosine similarity merupakan metode yang banyak digunakan dalam sistem rekomendasi untuk menghitung tingkat kemiripan antar item atau dokumen dengan membandingkan arah vektor fitur pada ruang vektor. Pendekatan ini bekerja dengan mengukur sudut kosinus antara dua vektor, di mana semakin kecil sudutnya, semakin tinggi tingkat kemiripannya. Nilai kemiripan yang dihasilkan berada pada rentang -1 hingga 1 , dengan nilai mendekati 1 menunjukkan kesamaan yang sangat tinggi. Teknik ini dianggap efektif terutama dalam sistem rekomendasi berbasis konten karena dapat mengukur kemiripan antar dokumen tanpa terpengaruh oleh perbedaan skala panjang dokumen. Menurut Oyadila et al. (2025), metode ini sangat relevan ketika digunakan bersama *TF-IDF* untuk memetakan hubungan antar dokumen berdasarkan kata-kata yang ada di dalamnya. Penelitian Oyadila et al. (2025) juga menunjukkan bahwa *Cosine similarity* mampu meningkatkan ketepatan rekomendasi melalui proses perbandingan vektor yang terstandarisasi.

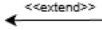
Dalam konteks pengolahan teks, *Cosine similarity* digunakan untuk menentukan kedekatan antar dokumen berdasarkan bobot kata yang telah dihitung menggunakan *TF-IDF*. Metode ini mengabaikan besarnya nilai vektor dan hanya berfokus pada kesamaan

arah, sehingga sangat cocok diterapkan pada dokumen dengan jumlah fitur Samosir & Ginting, (2024) menjelaskan bahwa nilai kemiripan mendekati 1 menunjukkan adanya hubungan kuat antara dua dokumen, sedangkan nilai mendekati 0 menandakan tidak adanya kesamaan. Integrasi antara *TF-IDF* dan *Cosine similarity* memungkinkan sistem melakukan proses pencocokan konten secara lebih akurat dalam aplikasi seperti rekomendasi film, resep, maupun berita. Studi Fitria et al. (2024) menegaskan bahwa metode ini memberikan representasi kemiripan yang stabil pada dataset teks berskala besar. Di sisi lain, penelitian Aulia et al. (2025) serta Herimanto et al. (2024) mendukung pandangan bahwa *Cosine similarity* tetap menjadi teknik pengukuran kemiripan paling efisien dalam berbagai model rekomendasi modern karena ketepatannya dalam membedakan dokumen yang relevan dan tidak relevan.

1.5.8 *Use case Diagram*

Use case Diagram merupakan salah satu model pemodelan utama dalam UML yang digunakan untuk memetakan interaksi antara pengguna dan sistem secara konseptual. Diagram ini membantu menggambarkan bagaimana sistem merespons kebutuhan aktor melalui sejumlah skenario fungsional yang terstruktur (Rospricilia et al., 2024). Dalam konteks rekayasa perangkat lunak, *use case* berfungsi sebagai fondasi dalam menetapkan kebutuhan fungsional karena menyediakan gambaran menyeluruh mengenai layanan yang harus disediakan oleh sistem. Selain itu, *use case* menjadi referensi penting selama proses analisis dan desain, membantu pengembang memahami ruang lingkup sistem, serta memastikan bahwa setiap fungsi yang dibangun benar-benar sesuai dengan kebutuhan pengguna.

Secara struktural, *Use case Diagram* melibatkan dua komponen utama, yaitu aktor dan *use case*. Aktor mewakili entitas eksternal yang berinteraksi dengan sistem baik manusia, perangkat, maupun sistem lainnya sedangkan *use case* mendeskripsikan layanan atau perilaku sistem yang dapat diakses oleh aktor. Relasi antara keduanya menunjukkan aliran interaksi yang mencerminkan bagaimana suatu proses dijalankan.

Simbol	Keterangan
	Aktor: Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan <i>use case</i>
	Generalisasi: Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i>
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya
	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi

Gambar 4. Simbol *Use case Diagram*

1.5.9 Activity Diagram

Activity Diagram merupakan salah satu komponen penting dalam UML yang berfungsi untuk menggambarkan alur aktivitas atau proses yang terjadi di dalam suatu sistem secara runtut dan terstruktur. Diagram ini menampilkan bagaimana sebuah aktivitas dimulai, dijalankan, bercabang, berlangsung secara paralel, hingga mencapai kondisi akhir, sehingga memberikan pemahaman yang komprehensif mengenai perilaku sistem (Gedam & Meshram, 2023). Melalui representasi visual yang jelas, *Activity Diagram* mempermudah pengembang dalam menelusuri logika proses, memahami hubungan antar langkah, serta mengidentifikasi area proses yang memerlukan optimalisasi.

Tujuan utama dari *Activity Diagram* adalah memberikan gambaran yang utuh mengenai alur proses yang berlangsung di dalam sistem sehingga pengembang dan analis dapat memahami setiap langkah secara lebih mendalam. Melalui diagram ini, alur kerja yang berasal dari *use case* dapat divisualisasikan menjadi rangkaian aktivitas yang lebih detail, sehingga memudahkan dalam mengidentifikasi peran pengguna, interaksi yang terjadi, serta urutan tindakan yang harus dilakukan.

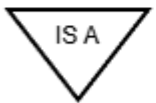
Simbol	Nama	Keterangan
	Status awal	Sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan/ <i>Decision</i>	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
	Penggabungan/ <i>Join</i>	Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu.
	Status akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
	<i>Swimlane</i>	<i>Swimlane</i> memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

Gambar 5. Simbol Activity Diagram

1.5.10 Entity Relational Database (ERD)

Entity Relational Database (ERD) merupakan model konseptual yang berfungsi untuk menggambarkan struktur data serta hubungan antar entitas dalam suatu basis data secara terstruktur dan logis. *ERD* memuat tiga komponen utama entitas, atribut, dan relasi yang direpresentasikan melalui notasi grafis untuk memudahkan proses analisis kebutuhan data dan penyusunan skema konseptual basis data. Dalam konteks perancangan sistem, keberadaan *ERD* membantu pengembang mengidentifikasi objek-objek data yang relevan, menghindari redundansi, serta memastikan kesesuaian antara kebutuhan informasi dengan desain basis data yang dihasilkan (Akbar & Haryanti, 2021). Dengan demikian, *ERD* tidak hanya berfungsi sebagai alat representasi visual, tetapi juga menjadi sarana pengendali kualitas desain data pada tahap awal rekayasa perangkat lunak.

Selain itu, *ERD* berperan penting dalam memetakan jenis relasi antardata seperti *one-to-one*, *one-to-many*, dan *many-to-many* yang menggambarkan tingkat kedekatan serta arah keterhubungan antara entitas. Pemodelan tersebut memungkinkan analisis sistem menilai integritas referensial, mendeteksi ketergantungan data, serta mengoptimalkan struktur basis data sebelum masuk ke tahap implementasi. Penggunaan *ERD* juga mendukung pembentukan database relasional yang konsisten dengan prinsip-prinsip normalisasi sehingga data dapat dikelola secara efisien dan mudah dipelihara (Afifah et al., 2022). Oleh sebab itu, *ERD* menjadi instrumen fundamental dalam proses desain konseptual dan dokumentasi sistem informasi, sekaligus landasan bagi pengembangan basis data yang terukur dan terstandarisasi.

Simbol	Nama	Keterangan
	Entitas	Sebuah diagram aktivitas memiliki sebuah status awal.
	Relasi	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Atribut	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
	Spesialisasi	Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu.
	Connection	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

Gambar 6. Simbol *Entity Relational Database*

1.5.11 *Black box testing*

Black box testing merupakan metode pengujian perangkat lunak yang berfokus pada evaluasi fungsi sistem berdasarkan spesifikasi yang telah ditetapkan tanpa memerlukan pemahaman terhadap struktur internal atau kode program yang digunakan. Pendekatan ini memungkinkan penguji untuk mengamati perilaku sistem dari perspektif pengguna akhir, sekaligus menilai apakah setiap fitur beroperasi sesuai dengan kebutuhan fungsional yang telah dirancang (Zen et al., 2024). Dalam praktiknya, teknik ini sering digunakan untuk menguji validitas input dan output, proses bisnis, serta alur kerja sistem secara keseluruhan. Karena tidak bergantung pada detail teknis, metode ini sangat relevan dalam tahap verifikasi fungsional pada siklus pengembangan perangkat lunak.

Selain memeriksa kesesuaian fungsi, *Black box testing* juga membantu mengidentifikasi berbagai kategori kesalahan seperti fungsi yang tidak berjalan atau tidak tersedia, kesalahan antarmuka, masalah struktur data dan akses basis data, hingga gangguan performa maupun kesalahan inisialisasi dan terminasi sistem. Pengujian ini memberikan manfaat penting dalam menemukan ketidaksesuaian terhadap spesifikasi kebutuhan, namun tetap memiliki keterbatasan karena penguji tidak memiliki akses terhadap kode program sehingga pengujian tidak dapat dilakukan secara menyeluruh pada level teknis (Fediando et al., 2023). Walaupun demikian, teknik ini dinilai efektif untuk memastikan pengalaman pengguna yang konsisten serta mendeteksi cacat fungsional yang berdampak langsung pada penggunaan sistem.

Dalam pengembangannya, *Black box testing* banyak diterapkan karena sifatnya yang objektif dan tidak terpengaruh oleh kompleksitas implementasi internal. Pengujian dilakukan dengan mendefinisikan berbagai skenario input berdasarkan spesifikasi sistem dan kemudian memeriksa apakah keluaran yang dihasilkan sudah sesuai dengan yang diharapkan. Dengan demikian, metode ini berperan sebagai alat verifikasi penting untuk mengevaluasi kualitas fungsi sistem sebelum dirilis kepada pengguna.

1.6 Penelitian Terkait

Penelitian mengenai sistem rekomendasi lowongan kerja telah menjadi topik yang banyak dikaji dalam literatur ilmiah. Berbagai studi terdahulu telah membahas pengembangan sistem rekomendasi lowongan kerja dengan pendekatan yang beragam, mulai dari metode berbasis konten, *Collaborative Filtering*, hingga pendekatan *hybrid* yang mengombinasikan beberapa teknik. Ringkasan mengenai penelitian terdahulu yang relevan dengan topik penelitian ini dapat dilihat pada Tabel 1.

Tabel 1. Ringkasan Penelitian Terkait

Tahun	Penulis	Judul	Identifikasi Masalah	Teknik / Metode	Hasil
2025	Hersianty et al.	Penerapan Algoritma <i>TF-IDF</i> dan <i>Cosine similarity</i> Dalam Sistem Rekomendasi Lowongan Pekerjaan	Pencari kerja kesulitan menemukan lowongan yang sesuai karena banyaknya informasi dan variasi format deskripsi pekerjaan.	<i>TF-IDF</i> untuk pembobotan kata kunci dan <i>Cosine similarity</i>	Sistem mencapai tingkat akurasi 80% dalam menyandingkan pencari kerja dengan peluang karir yang sesuai dengan kompetensi mereka.
2025	Rahmawati et al.	Sistem Rekomendasi Bursa Kerja Khusus (BKK) Menggunakan Metode <i>Content Based Filtering</i> pada SMK Tunas Bangsa	Penyaluran alumni ke dunia kerja masih manual menggunakan <i>Google Form</i> , menghasilkan rekomendasi tidak sesuai keahlian dan menyulitkan pemantauan data.	<i>Content-based filtering</i> dengan algoritma <i>TF-IDF</i> untuk pembobotan skill dan <i>Cosine similarity</i>	Sistem mencapai <i>Precision</i> 1.00, <i>Recall</i> 0.60, dan <i>F1-Measure</i> 0.75, mempermudah pencocokan alumni dengan perusahaan secara otomatis dan terstruktur.

2024	Fitria et al.	Sistem Rekomendasi Lowongan Pekerjaan Menggunakan <i>Content-based filtering</i>	Platform pencarian kerja online menawarkan banyak pilihan namun membuat pengguna kewalahan, serta kesenjangan antara kualifikasi pencari kerja dan kebutuhan pasar.	<i>Content-based filtering</i> dengan <i>TF-IDF</i> untuk pembobotan kata kunci dan <i>Cosine similarity</i>	Sistem memberikan rekomendasi relevan dengan rata-rata nilai presisi 0.53, menggunakan 437 data lowongan dari JobStreet dan 100 data profil dari LinkedIn.
2024	Darmawan et al.	Aplikasi Lowongan Kerja Online Menggunakan Metode <i>Hybrid-Based Recommendation</i>	Sulitnya mendapatkan pekerjaan sesuai latar belakang pendidikan dan keterbatasan informasi lowongan kerja menjadi hambatan.	<i>Hybrid-Based Recommendation</i> yang menggabungkan <i>Collaborative Filtering</i> (<i>Cosine similarity</i>) dan <i>Content-Based</i> (<i>Alternating Least Squares</i>).	Sistem berhasil memberikan rekomendasi pekerjaan sesuai latar belakang pendidikan pelamar dengan nilai UAT 86,75% (kategori berhasil).
2023	Koloman et al.	Sistem Rekomendasi Pekerjaan di bidang IT Menggunakan Algoritma <i>Content-based filtering</i>	Pencari kerja di bidang IT menghadapi kesulitan menemukan pekerjaan yang sesuai dengan latar belakang dan keterampilan mereka.	<i>Content-based filtering</i> dengan <i>TF-IDF</i> dan <i>Cosine similarity</i>	Sistem berhasil mencocokkan latar belakang keterampilan pengguna dengan lowongan yang sesuai, dan dapat diidentifikasi area keterampilan

					yang perlu ditingkatkan.
2022	Habibi & Albanna	Analisis Sistem Rekomendasi Pada <i>Job Recommendation</i> Berdasarkan Profil LinkedIn Menggunakan <i>Cosine similarity</i>	Proses <i>filtering</i> calon karyawan memakan waktu lama dan tingkat kecocokan data calon karyawan dengan pekerjaan tidak dapat diukur.	<i>Content-based filtering</i> dengan <i>word embedding</i> menggunakan GloVe dan <i>Cosine similarity</i>	Sistem berhasil memberikan rekomendasi pekerjaan yang diurutkan dari <i>similarity</i> terbesar hingga terkecil, mempermudah proses pencarian kandidat.
2022	Raharjo et al.	Sistem Rekomendasi <i>Content Based Filtering</i> Pekerjaan dan Tenaga Kerja Potensial menggunakan <i>Cosine similarity</i>	Pandemi COVID-19 menyebabkan PHK massal dan proses pencarian kerja secara tradisional membuang banyak waktu.	<i>Content-based filtering</i> dengan algoritma <i>TF-IDF</i> dan <i>Word2Vec-CBOW</i> , serta <i>Cosine similarity</i>	<i>TF-IDF</i> menghasilkan <i>Mean Reciprocal Rank</i> 0,857 dan <i>Mean Average Precision</i> 0,833, lebih baik dibanding <i>Word2Vec</i> (MRR 0,6078, MAP 0,5822).
2020	Wibowo et al.	Sistem Pendukung Keputusan Berbasis Web Rekomendasi Pekerjaan Bagi Lulusan JTI Polinema Dengan Metode SAW	Fresh graduate kesulitan mencari pekerjaan sesuai kemampuan karena keterbatasan pengalaman kerja dan banyaknya pilihan lowongan.	<i>Simple Additive Weighting</i> (SAW) dengan kriteria pendidikan terakhir, IPK, usia, akreditasi, dan kemampuan berbahasa Inggris.	Sistem berhasil memberikan 3 rekomendasi pekerjaan yang sesuai dengan kriteria pengguna berdasarkan bobot yang diinputkan.
2020	Alim et al.	Rekomendasi Keterampilan Teknologi Informasi	Pencari kerja bingung memilih keterampilan	<i>User-Based Collaborative Filtering</i> dengan <i>Log-</i>	Sistem dapat memberikan rekomendasi keterampilan

Menggunakan Metode <i>User-Based Collaborative Filtering</i> dan <i>Log-Likelihood Similarity</i>	yang harus dikuasai karena banyaknya pilihan dan perkembangan teknologi yang cepat.	<i>Likelihood Similarity</i>	dengan baik, dengan nilai pengujian ISO-9126 sebesar 83,81% (kategori baik).
---	---	------------------------------	--

Berdasarkan tinjauan terhadap penelitian-penelitian terdahulu yang dijelaskan pada Tabel 1, metode *Content-based filtering* dengan algoritma *TF-IDF* dan *Cosine similarity* menjadi pendekatan yang paling dominan digunakan dalam sistem rekomendasi lowongan kerja. Dari sembilan penelitian yang dikaji, enam penelitian menggunakan kombinasi *TF-IDF* dan *Cosine similarity*, menunjukkan bahwa metode ini telah teruji. Keunggulan metode ini dibuktikan oleh beberapa penelitian dengan hasil yang konsisten tinggi, seperti Hersianty et al. (2025) yang mencapai akurasi 80%, Rahmawati et al. (2025) dengan presisi sempurna 1.00 dan *F1-Measure* 0.75, serta Raharjo et al. (2022) yang menunjukkan *TF-IDF* menghasilkan *Mean Reciprocal Rank* 0,857 dan *Mean Average Precision* 0,833, lebih unggul dibandingkan metode *Word2Vec-CBOW*. *Track record* keberhasilan yang konsisten ini menjadi landasan kuat untuk mengadopsi metode yang sama dalam penelitian ini.

Pemilihan *Content-based filtering* dengan *TF-IDF* dan *Cosine similarity* pada penelitian ini didasarkan pada beberapa pertimbangan teknis dan praktis. Pertama, metode ini efektif dalam mengekstrak dan membobot kata kunci dari data tekstual seperti keterampilan, kualifikasi, dan pengalaman yang terdapat dalam profil pengguna maupun deskripsi lowongan kerja. Kedua, *Cosine similarity* terbukti efisien dalam mengukur tingkat kemiripan antara vektor fitur tekstual, sebagaimana ditunjukkan oleh Koloman et al. (2023) yang berhasil mencocokkan latar belakang keterampilan pengguna dengan lowongan sesuai dan mengidentifikasi area keterampilan yang perlu ditingkatkan. Ketiga, metode ini mampu mengatasi *cold start problem* dengan hanya mengandalkan data profil pengguna tanpa memerlukan data interaksi historis yang kompleks, sehingga cocok untuk fresh graduate atau pengguna baru. Terakhir, efisiensi komputasi yang relatif ringan membuat metode ini cocok untuk implementasi sistem yang responsif dan dapat diakses secara *real-time*. Dengan demikian, penelitian ini mengadopsi pendekatan *Content-based filtering* dengan algoritma *TF-IDF* untuk pembobotan fitur dan *Cosine similarity* untuk perhitungan kemiripan, mengikuti *best practice* yang telah terbukti efektif pada penelitian-penelitian sebelumnya, sambil menyesuaikannya dengan konteks dan kebutuhan spesifik sistem rekomendasi lowongan kerja yang dikembangkan.

BAB II METODE PENELITIAN

2.1 Waktu dan Tempat Penelitian

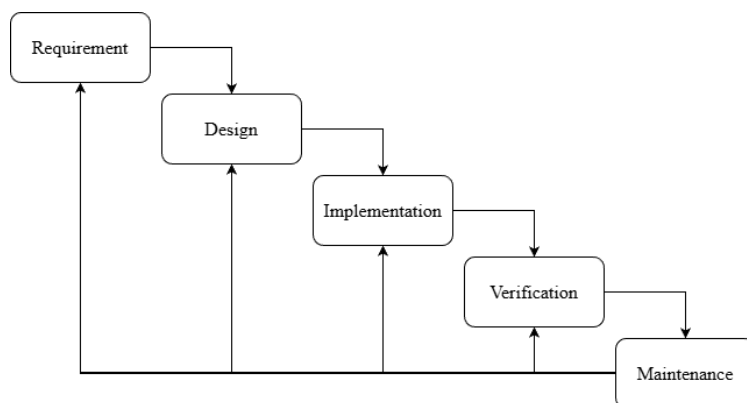
Penelitian ini berlangsung mulai Juni 2025 hingga Oktober 2025. Penelitian ini bertempat di Baruga Telkomsel dan Laboratorium Program Studi Sistem Informasi yaitu Laboratorium Rekayasa dan Perangkat Lunak (Lab RPL) Universitas Hasanuddin Kota Makassar, Indonesia. Berikut Adalah rinci dari timeline penelitian:

No	Tahapan Penelitian	Juni				Juli				Agustus				September				Oktober			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Pengumpulan Data																				
2	Studi Literatur																				
3	Requirement																				
4	Desain Sistem																				
5	Implementasi																				
6	Testing																				
7	Maintenance																				

Gambar 7. Timeline Penelitian

2.2 Metode Pengembangan Sistem

Penelitian ini menggunakan metode *System Development Life Cycle* (SDLC) model *Waterfall* seperti pada Gambar 8, yaitu pendekatan pengembangan sistem yang berlangsung secara berurutan dan terstruktur. Setiap tahapan harus diselesaikan terlebih dahulu sebelum berlanjut ke tahap berikutnya. Model ini dipilih karena mampu memberikan alur kerja yang jelas, terdokumentasi, dan sesuai untuk pengembangan sistem yang kebutuhan awalnya telah ditentukan.



Gambar 8. Metode *Waterfall*

2.3 Metode Pengumpulan Data

Untuk menunjang pengembangan sistem, dibutuhkan informasi dan data yang memadai. Pada penelitian ini, proses pengumpulan data dilakukan melalui beberapa metode tertentu guna memastikan sistem yang dibangun sesuai dengan kebutuhan dan tujuan penelitian.

1. *Study Literature*

Studi literatur merupakan proses menelaah berbagai sumber referensi dan hasil penelitian terdahulu yang berkaitan dengan topik, guna memperoleh dasar teori yang mendukung permasalahan yang diteliti Sarwono (dalam Munib & Wulandari, 2021). Dalam penelitian ini, penulis melakukan studi literatur dengan menghimpun referensi dari jurnal, buku, hingga situs-situs web yang relevan.

2. *Focus Discussion Group (FGD)*

Focus Discussion Group merupakan Teknik pengumpulan data yang di mana sesi diskusi kelompok fokus melibatkan beberapa partisipan yang dikumpulkan dalam sebuah diskusi terencana untuk mengeksplorasi topik tertentu Krueger (dalam Belanger & Tech, 2012). Dalam penelitian ini, penulis melaksanakan FGD di Telkomsel bersama manajer terkait serta perwakilan dari Kementerian Pekerjaan Umum, guna memperoleh pemahaman yang lebih komprehensif mengenai kebutuhan dalam pembangunan sistem yang diinginkan.

2.4 Instrumen Penelitian

Instrumen penelitian dalam pengembangan sebuah sistem memanfaatkan penggunaan perangkat keras dan perangkat lunak sebagai komponen utama. Perangkat keras yang digunakan pada penelitian ini adalah Lenovo Ideapad Slim 1 Ryzen 3 7320 dengan spesifikasi sebagai berikut.

Tabel 2. *Hardware yang digunakan*

Nama Perangkat	Spesifikasi
Operating System (OS)	Windows 11
Processor	AMD Ryzen 3 7320U with Radeon Graphics 2.40 GHz
RAM	8,00 GB (5,74 GB usable)
GPU	AMD Radeon 610M Graphics

Penelitian ini turut memanfaatkan sejumlah perangkat lunak yang berfungsi mendukung dan memfasilitasi proses perancangan sistem. Setiap perangkat lunak tersebut memiliki peran khusus dalam meninjau serta menunjang pengembangan sistem yang akan dibangun, yaitu sebagai berikut.

Tabel 3. *Software yang digunakan*

Nama Perangkat	Kegunaan
Visual Studio Code	Digunakan untuk menulis dan mengelola kode pemrograman
Figma	Digunakan untuk merancang desain <i>User Interface</i> sistem yang akan dibuat, seperti layout, ikon yang digunakan dan lain lain.
Draw.io	Digunakan untuk merancang flowchart, <i>ERD</i> , <i>Use case Diagram</i> dan <i>Activity Diagram</i>

Nama Perangkat	Kegunaan
Postman	Digunakan untuk melakukan permintaan HTTP serta berinteraksi dengan API guna memperoleh data yang selanjutnya disajikan pada antarmuka pengguna.
Xampp	Digunakan sebagai lingkungan server lokal untuk menjalankan aplikasi web, mengelola basis data, serta melakukan pengujian sistem sebelum dipublikasikan ke server produksi.
<i>Web Browser</i>	<i>Web browser</i> seperti Chrome digunakan sebagai media untuk mengakses sekaligus melakukan pengujian terhadap sistem yang sedang dikembangkan.

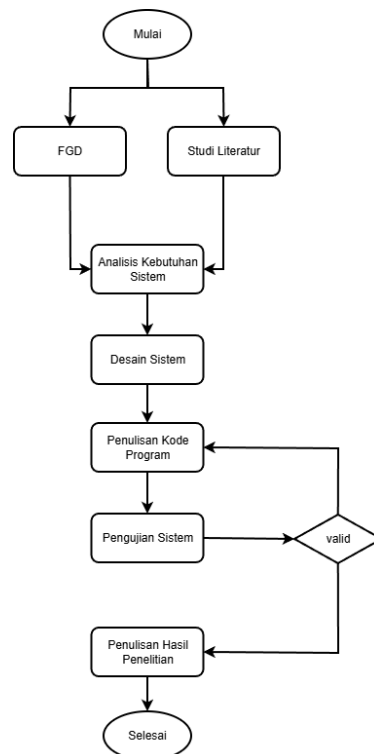
2.5 Tahapan Penelitian

Penelitian ini dilaksanakan melalui enam tahapan yang saling berkaitan, dimulai dari indentifikasi kebutuhan hingga dokumentasi hasil akhir.

1. FGD dan Studi Pustaka
FGD dan studi pustaka dilakukan untuk mengidentifikasi kebutuhan serta permasalahan yang relevan, sehingga dapat dirumuskan kebutuhan sistem yang akan dikembangkan. Tahapan ini menjadi dasar dalam menentukan ruang lingkup dan spesifikasi kebutuhan secara menyeluruh.
2. Analisis Kebutuhan Sistem
Pada tahap ini dilakukan pendalaman terhadap seluruh kebutuhan pengguna dengan tujuan memastikan sistem mampu memberikan solusi terhadap permasalahan yang telah diidentifikasi. Semua kebutuhan fungsional maupun non-fungsional didokumentasikan sebagai acuan pada tahap desain.
3. Desain Sistem
Tahap ini mencakup penyusunan rancangan arsitektur sistem, desain antarmuka pengguna, serta perancangan alur proses kerja sistem. Hasil desain menjadi referensi utama bagi pengembang untuk memastikan sistem yang dibangun sesuai dengan kebutuhan yang telah ditetapkan.
4. Penulisan Kode Program
Berdasarkan desain yang telah disusun, tahap implementasi dilakukan dengan menulis kode program yang merealisasikan fungsi-fungsi sistem. Pada tahap ini, pengembang mengimplementasikan logika bisnis, struktur basis data, serta integrasi dengan API atau komponen lain sesuai desain.
5. Pengujian Sistem
Setelah implementasi selesai, dilakukan pengujian untuk memastikan seluruh bagian sistem berjalan sesuai spesifikasi. Pengujian dapat meliputi pengujian fungsional, performa, serta aspek keamanan untuk memverifikasi bahwa sistem bekerja dengan benar dan konsisten.

6. Penulisan Hasil Penelitian

Tahap terakhir adalah mendokumentasikan seluruh hasil pengembangan sistem. Laporan penelitian mencakup keseluruhan proses, mulai dari analisis, perancangan, implementasi, hingga pengujian sistem.

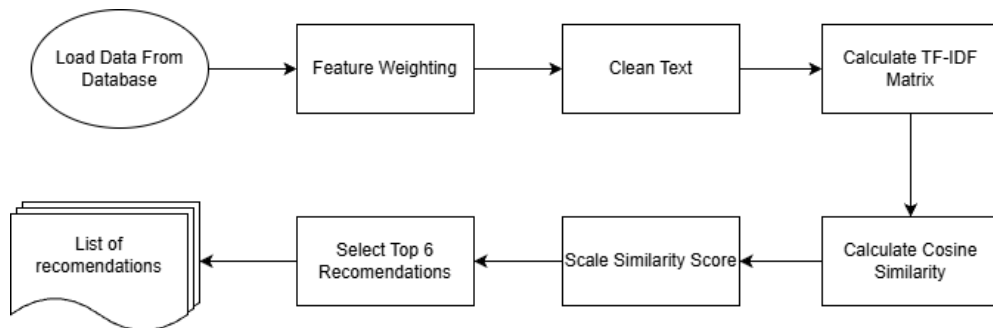


Gambar 9. Tahapan Penelitian

2.6 Rancangan Sistem Rekomendasi

Sistem rekomendasi lowongan kerja pada penelitian ini dirancang menggunakan pendekatan *content-based filtering* dengan metode *TF-IDF* (Term Frequency-Inverse Document Frequency) dan *cosine similarity*. Pemilihan metode ini sejalan dengan penelitian yang dilakukan oleh Fitria et al. (2024) yang menggunakan algoritma text filtering seperti *TF-IDF* dan *Cosine similarity* untuk memastikan peran pekerjaan yang diinginkan dan keterampilan pelamar dipertimbangkan sebelum merekomendasikan pekerjaan. Pendekatan *content-based filtering* dipilih karena kemampuannya untuk memberikan rekomendasi yang personal tanpa memerlukan data historis interaksi pengguna dalam jumlah besar, sehingga cocok diterapkan pada sistem yang baru diimplementasikan dan dapat mengatasi *cold start problem* untuk pengguna baru.

Berbeda dengan pendekatan standar yang memberikan bobot sama pada semua fitur, sistem ini menerapkan skema pembobotan hierarkis yang memberikan prioritas berbeda pada setiap atribut pengguna dan lowongan kerja. Pendekatan ini mengadopsi strategi yang diusulkan oleh Alsaif et al. (2022) yang menyatakan bahwa "*assigned higher weights to job skills*" untuk meningkatkan relevansi rekomendasi.



Gambar 10. Flowchart *content-based filtering*

Alur kerja sistem rekomendasi ini terdiri dari tujuh tahap utama yang saling terintegrasi. Proses dimulai dengan *Load Data From Database* yang mengambil profil pengguna dari tabel *alumni_siswa_profiles* berdasarkan *user_id* dan seluruh data lowongan dengan status 'buka' dari tabel *lokers*. Kemudian masuk ke tahap *Feature Weighting* dimana fitur-fitur pengguna dan lowongan kerja diberi bobot berbeda sesuai prioritas skills diberi bobot 5x (prioritas tertinggi), bidang_pekerjaan 3x, sedangkan jurusan dan sertifikasi 1x dengan mengulang kata-kata tersebut sesuai bobotnya untuk menekankan kepentingannya dalam perhitungan *TF-IDF*. Data teks hasil pembobotan tersebut kemudian melalui proses *Clean Text* menggunakan fungsi pembersihan teks yang mengubah teks menjadi *lowercase*, menghapus karakter khusus dan angka menggunakan *regular expression*, serta menormalisasi spasi untuk memastikan konsistensi data. Setelah teks bersih, sistem menghitung *TF-IDF Matrix* menggunakan *TfidfVectorizer* dengan parameter *ngram_range*=(1,2) untuk menangkap unigram dan bigram serta *min_df*=1, yang mengubah teks pengguna dan semua lowongan menjadi vektor numerik yang merepresentasikan kepentingan setiap term. Selanjutnya, sistem menghitung *Cosine similarity* antara vektor *TF-IDF* pengguna dengan setiap vektor lowongan untuk mengukur tingkat kecocokan konten mereka. Skor *similarity* kemudian melalui *Scale Similarity Score* dengan mengalikan nilai *cosine* (0-1) dengan 100 untuk mengkonversinya menjadi persentase yang lebih mudah dipahami. Pada tahap *Select Top 6 Recommendations*, sistem menggunakan metode *nlargest*(6) untuk memilih 6 lowongan dengan skor *similarity_persen* tertinggi. Akhirnya, hasil berupa *List of Recommendations* dikembalikan dalam format JSON lengkap dengan detail perhitungan *TF-IDF*, *matching terms*, formula *cosine similarity*, statistik *summary*, dan informasi lengkap setiap lowongan yang direkomendasikan kepada pengguna melalui endpoint API */rekomendasi/{user_id}*.

Formula *TF-IDF* yang digunakan dalam sistem ini adalah:

$$TF - IDF(t, D) = \frac{freq(t, D)}{max_others(t, D)} \times \log \frac{N}{df(t)} \quad (1)$$

Keterangan:

t	= <i>Term</i> (kata/token yang dihitung)
D	= <i>Document</i> (dokumen yang dianalisis)
freq	= Frekuensi kemunculan
max_others	= Frekuensi maksimal
N	= Total jumlah dokumen dalam koleksi
df(t)	= Jumlah dokumen yang mengandung term t

Formula *Cosine similarity* yang digunakan dalam sistem ini adalah:

$$\text{cosine_similarity}(A, B) = \frac{A \cdot B}{||A|| ||B||} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \times \sqrt{\sum_{i=1}^n B_i^2}} \quad (2)$$

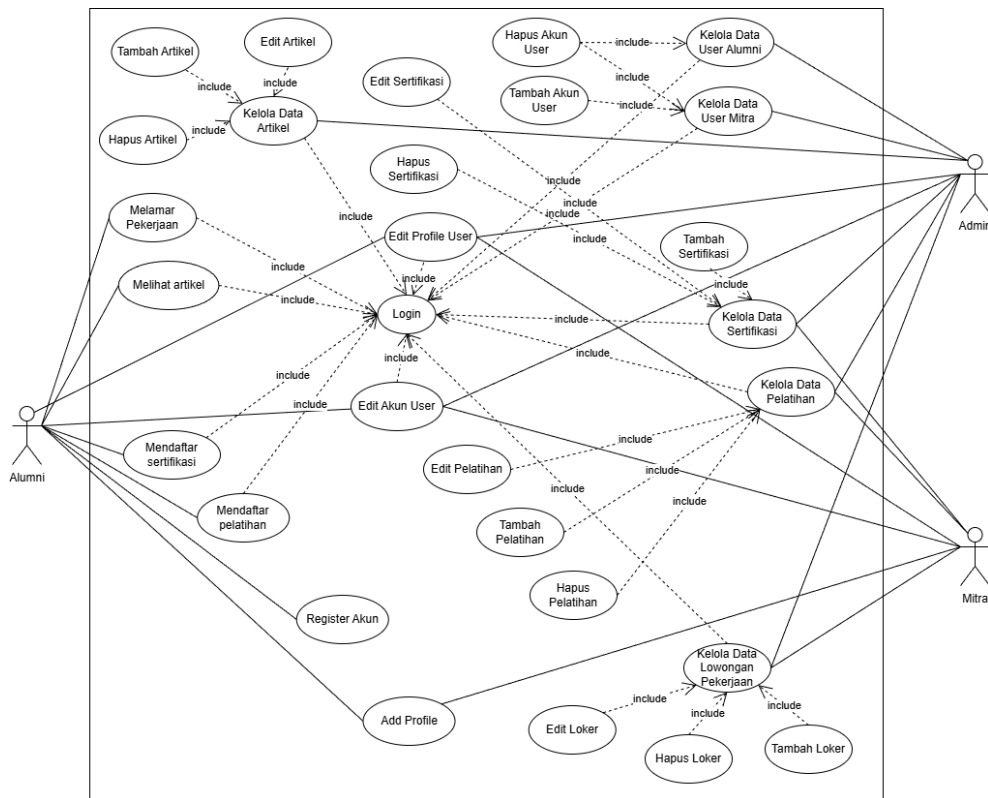
Keterangan:

A	= Vektor <i>TF-IDF</i> dari profil pengguna
B	= Vektor <i>TF-IDF</i> dari lowongan pekerjaan
A	= <i>Magnitude</i> (panjang/norm) dari vektor A
B	= <i>Magnitude</i> (panjang/norm) dari vektor B
A _i	= Nilai elemen ke-i dari vektor A
B _i	= Nilai elemen ke-i dari vektor B
n	= Jumlah total dimensi/fitur dalam vektor (jumlah <i>term</i> dalam <i>vocabulary</i>)

2.7 Rancangan Interaksi Sistem

Rancangan interaksi sistem menggambarkan hubungan antara aktor dengan berbagai fungsi yang tersedia dalam sistem. Terdapat tiga aktor utama yang berinteraksi dengan sistem, yaitu Alumni, Admin, dan Mitra. Setiap aktor memiliki hak akses dan fungsi yang berbeda sesuai dengan peran dan kebutuhannya dalam sistem. Diagram *use case* pada Gambar 11 menunjukkan secara komprehensif seluruh interaksi yang dapat dilakukan oleh masing-masing aktor, mulai dari proses registrasi, pengelolaan data, hingga manajemen pelatihan dan sertifikasi.

Alumni memiliki akses untuk melakukan registrasi akun, mengelola profil pribadi, mendaftar sertifikasi dan pelatihan, serta melihat artikel yang tersedia dalam sistem. Admin berperan sebagai pengelola sistem dengan kemampuan untuk menambah, mengubah, dan menghapus data artikel, akun pengguna, mitra, pelatihan, sertifikasi, serta lowongan kerja yang ditawarkan. Sementara itu, Mitra memiliki fungsi untuk mendaftarkan sertifikasi dan pelatihan yang akan ditawarkan kepada alumni, serta mengelola informasi lowongan kerja yang tersedia. Pembagian peran ini dirancang untuk memastikan sistem dapat berjalan secara efektif dan efisien sesuai dengan kebutuhan masing-masing pengguna.



Gambar 11. Use case Diagram

2.8 Desain Sistem

Bagian ini menguraikan rancangan sistem yang dikembangkan sebagai landasan dalam membangun solusi yang sesuai dengan kebutuhan fungsional maupun non-fungsional. Desain sistem disusun secara terstruktur untuk memastikan setiap komponen dapat saling berintegrasi, mendukung alur proses yang telah dianalisis, serta memberikan gambaran menyeluruh mengenai arsitektur dan mekanisme kerja sistem sebelum tahap implementasi.

2.8.1 Login

Halaman *Login* menampilkan susunan elemen antarmuka yang dirancang sederhana dan terfokus untuk memudahkan pengguna dalam mengakses sistem. Pada bagian kiri terdapat area ilustrasi atau gambar pendukung yang biasanya digunakan untuk memperkuat identitas visual aplikasi. Sementara itu, bagian kanan berisi komponen utama berupa field *username/email* dan *password*, tombol untuk mengeksekusi proses masuk, serta elemen tambahan seperti pesan validasi.



Gambar 12. Wireframe *Login*

2.8.2 *Register Alumni*

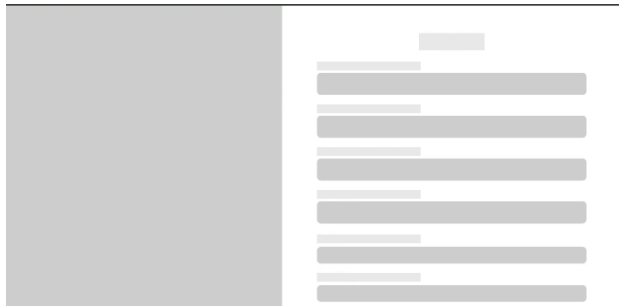
Halaman registrasi alumni memiliki tata letak yang mirip dengan halaman *Login*, namun dengan jumlah elemen formulir yang lebih banyak. Bagian kiri kembali berfungsi sebagai area ilustrasi, sedangkan sisi kanan menampilkan beberapa input yang harus diisi calon alumni, seperti nomor hp, password dan konfirmasi password. Selain itu terdapat tombol untuk menyelesaikan proses pendaftaran.



Gambar 13. Wireframe *Register Alumni*

2.8.3 *Add Profile*

Halaman penambahan profil menunjukkan tampilan yang lebih luas karena memuat formulir data yang lebih kompleks. Area kiri digunakan sebagai panel ilustrasi, sementara bagian kanan memuat rangkaian field untuk mengisi informasi profil alumni dan mitra.



Gambar 14. Wireframe *Add Profile* Alumni dan Mitra

2.8.4 Mitra

Pada bagian Mitra menggambarkan rancangan antarmuka yang disesuaikan dengan kebutuhan pengguna mitra dalam mengelola berbagai layanan seperti sertifikasi, lowongan kerja, pelatihan, maupun pengaturan profil. Setiap halaman memiliki struktur yang konsisten dengan penggunaan *sidebar* navigasi di bagian kiri dan area konten utama di sebelah kanan, sehingga memudahkan pengguna dalam berpindah antar fitur secara efisien.

Dashboard menampilkan ringkasan informasi penting yang berhubungan dengan aktivitas mitra pada platform. Bagian atas halaman berisi banner atau area informasi utama yang berfungsi sebagai pengingat atau pemberitahuan. Di bagian bawah terdapat beberapa *card*.



Gambar 15. Wireframe Dashboard Mitra

Halaman sertifikasi menampilkan daftar sertifikasi yang telah dibuat oleh mitra. Terdapat tabel utama yang memuat informasi seperti judul sertifikasi, tanggal pelaksanaan, status, dan tindakan seperti edit atau hapus. Elemen tombol aksi ditempatkan di bagian kanan setiap baris untuk memudahkan pengelolaan data.



Gambar 16. Wireframe Sertifikasi Mitra

Halaman loker memperlihatkan daftar lowongan kerja yang dipublikasikan oleh mitra. Serupa dengan halaman sertifikasi, bagian konten berisi tabel dengan kolom seperti posisi pekerjaan, kualifikasi, status lowongan, serta tombol aksi. Pengguna dapat melihat detail pelamar, memperbarui status lowongan, atau menutup lowongan yang tidak lagi aktif.



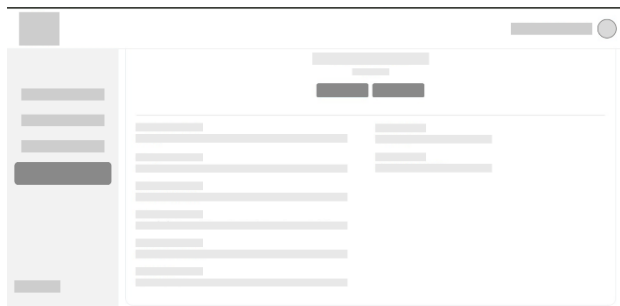
Gambar 17. Wireframe Lowongan Kerja Mitra

Halaman pelatihan menyajikan daftar program pelatihan yang diselenggarakan mitra. Kontennya mencakup informasi dasar seperti nama pelatihan, tempat, jadwal pelaksanaan, serta status kegiatan. Tombol aksi memungkinkan mitra melakukan pembaruan data maupun memeriksa jumlah peserta terdaftar.



Gambar 18. Wireframe Pelatihan Mitra

Halaman profil mitra menampilkan informasi identitas mitra yang dapat dikelola langsung oleh pengguna. Area konten utama berisi form yang memuat data seperti nama perusahaan/instansi, alamat, kontak, serta informasi administratif lainnya.



Gambar 19. Wireframe Profile Mitra

2.8.5 Alumni

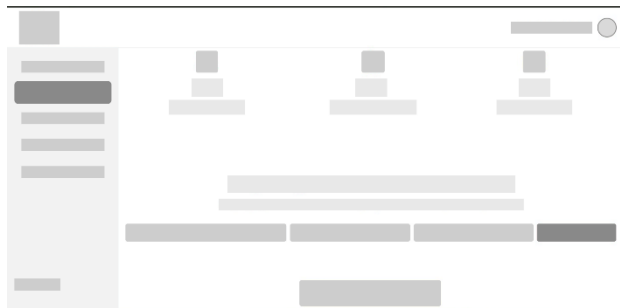
Pada bagian Alumni menggambarkan rancangan antarmuka yang ditujukan bagi pengguna alumni dalam melakukan pemantauan, pendaftaran kegiatan, serta pengelolaan informasi pribadi. Struktur halaman menggunakan pola konsisten, yaitu *sidebar* navigasi di sisi kiri dan area konten utama di sebelah kanan.

Dashboard alumni menampilkan rangkuman aktivitas dan informasi penting yang relevan bagi pengguna alumni. Pada bagian atas terdapat keterangan mengenai website dan pada bawah ada *card* yang serupa dengan navigasi.



Gambar 20. Wireframe Dashboard Alumni

Halaman sertifikasi menampilkan daftar sertifikasi yang dapat diikuti maupun yang sedang diikuti oleh alumni. Di area konten utama terdapat deretan kartu sertifikasi yang berisi informasi ringkas seperti nama kegiatan, jadwal pelaksanaan, deskripsi singkat, serta tombol untuk mendaftar.



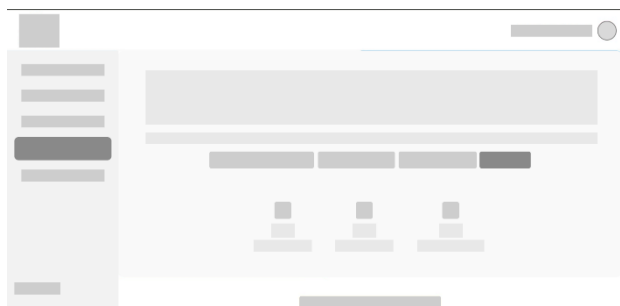
Gambar 21. Wireframe Sertifikasi Alumni

Halaman lowongan kerja menampilkan berbagai informasi mengenai kesempatan kerja yang ditawarkan oleh mitra. Bagian atas memuat tab angka atau kategori yang memfilter lowongan berdasarkan status, seperti “Semua”, “Kota”, atau “Tipe waktu pekerjaan”. Setiap kartu lowongan menampilkan posisi pekerjaan, nama perusahaan, lokasi, serta tombol aksi seperti simpan atau lamar.



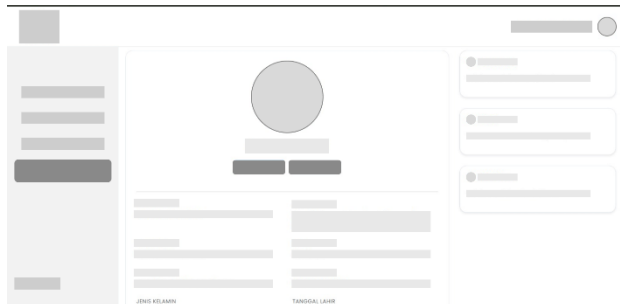
Gambar 22. Wireframe Lowongan Kerja Alumni

Halaman pelatihan memiliki struktur serupa dengan halaman sertifikasi, namun difokuskan pada kegiatan peningkatan keterampilan. Bagian atas halaman berisi navigasi search pelatihan seperti “Status”, “kota”, dan “Nama Pelatihan”. Di bawahnya terdapat daftar kartu pelatihan yang menampilkan informasi utama, termasuk jadwal, lokasi, penyelenggara, dan deskripsi ringkas.



Gambar 23. Wireframe Pelatihan Alumni

Halaman profil alumni menampilkan informasi personal pengguna secara menyeluruh. Pada bagian atas terdapat foto profil serta informasi identitas dasar seperti nama, program studi, dan tahun lulus. Bagian kanan berisi panel aktivitas atau notifikasi terkait lamaran kerja, sertifikasi, maupun pelatihan yang sedang diikuti.

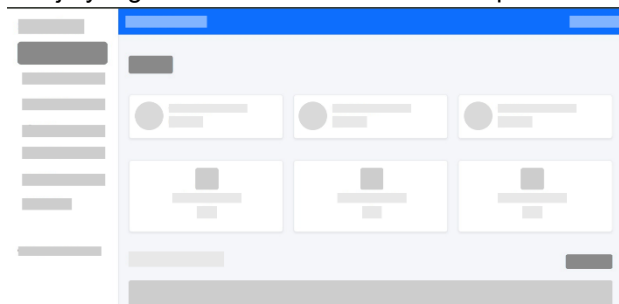


Gambar 24. Wireframe Profile Alumni

2.8.6 Admin

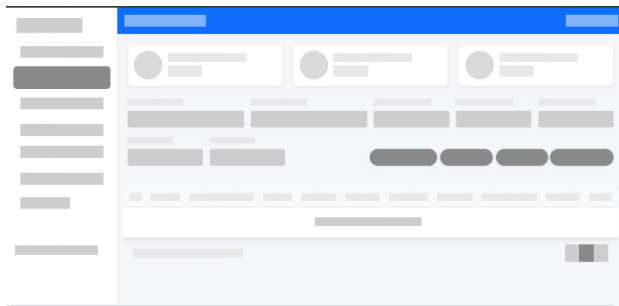
Pada bagian Admin merepresentasikan rancangan antarmuka untuk pengguna dengan hak akses pengelolaan sistem secara menyeluruh. Struktur navigasi menggunakan *sidebar* vertikal yang memuat seluruh menu administratif seperti dashboard, pengelolaan sertifikasi, pelatihan, lowongan kerja, artikel, dan manajemen user.

Dashboard admin menampilkan ringkasan informasi penting dalam bentuk statistik atau angka rekap, seperti jumlah alumni aktif, jumlah mitra, total sertifikasi, jumlah loker, dan kegiatan pelatihan. Tampilan awal halaman berisi beberapa kartu (*cards*) indikator kinerja yang membantu admin memantau perkembangan platform.



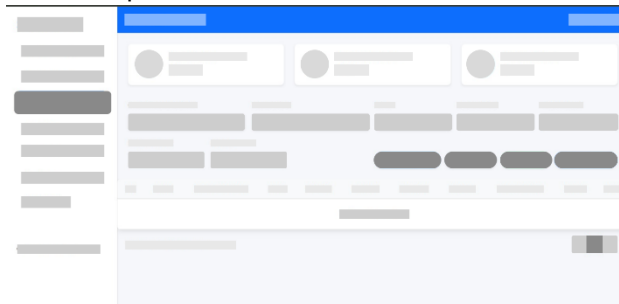
Gambar 25. Wireframe Dashboard Admin

Halaman sertifikasi admin menampilkan daftar seluruh program sertifikasi yang tersedia, lengkap dengan detail seperti nama sertifikasi, penyelenggara, tanggal pelaksanaan, serta jumlah peserta. Di bagian konten utama, setiap baris sertifikasi dilengkapi dengan tombol aksi seperti edit, lihat peserta, atau hapus. Halaman ini memungkinkan admin menambah sertifikasi baru dan memperbarui informasi.



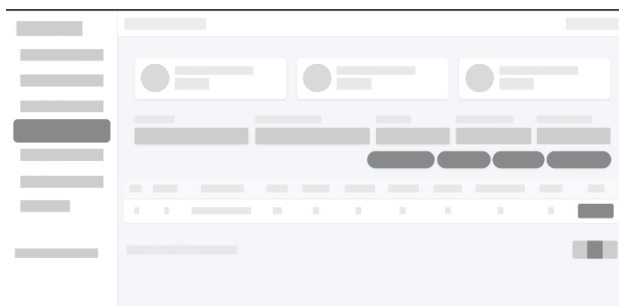
Gambar 26. Wireframe Sertifikasi Admin

Halaman pelatihan untuk admin memiliki struktur yang serupa dengan sertifikasi, namun difokuskan pada kegiatan pengembangan keterampilan. Admin dapat melihat daftar pelatihan beserta statusnya, mengatur jadwal, memperbarui deskripsi, dan memantau jumlah peserta. Terdapat beberapa tab untuk memfilter pelatihan berdasarkan status tertentu. Tabel utama memuat informasi ringkas serta tombol tindakan seperti edit dan hapus.



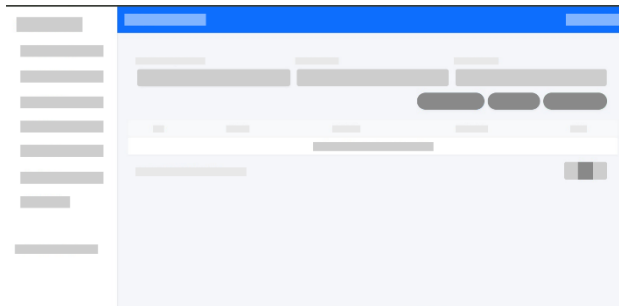
Gambar 27. Wireframe Pelatihan Admin

Halaman lowongan kerja menampilkan daftar loker yang dikirimkan oleh mitra ataupun yang dibuat langsung oleh admin. Kolom-kolom data menunjukkan posisi pekerjaan, perusahaan, lokasi, dan status publikasi. Pada bagian atas juga terdapat tab atau filter untuk mengelompokkan loker berdasarkan kondisi tertentu.



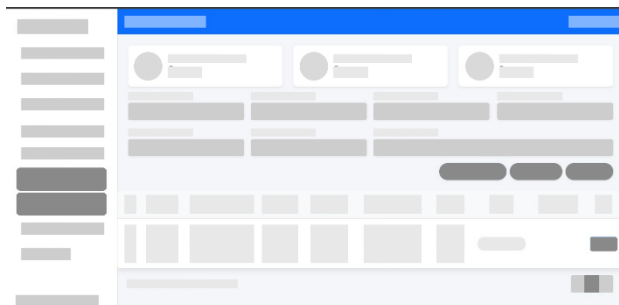
Gambar 28. Wireframe Lowongan Kerja Admin

Halaman artikel admin menggambarkan halaman pengelolaan konten informasi seperti berita, pengumuman, atau artikel edukatif. Pada bagian atas terdapat tombol untuk menambahkan artikel baru. Di bawahnya terdapat daftar artikel yang memuat judul, tanggal publikasi dan deskripsi singkat. Setiap artikel dilengkapi dengan tombol aksi seperti edit, lihat, atau hapus.



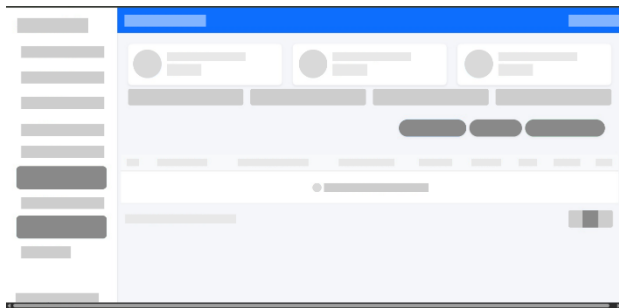
Gambar 29. Wireframe Artikel Admin

Halaman manajemen user mitra menggambarkan tampilan yang dirancang untuk mengelola seluruh akun mitra yang terdaftar pada sistem. *Sidebar* tetap berada di sisi kiri sebagai navigasi utama, sementara area konten menampilkan tabel daftar mitra.



Gambar 30. Wireframe Manajemen User Mitra

Halaman manajemen user alumni menampilkan halaman yang digunakan admin untuk mengatur seluruh akun alumni. Pada bagian atas tersedia kolom pencarian dan fitur penyaringan berdasarkan angkatan, asal sekolah dan status, sehingga memudahkan admin menelusuri data alumni dalam jumlah besar.



Gambar 31. Wireframe Manajemen User Alumni

2.9 Metode Pengujian Sistem

2.9.1 *Black box testing*

Metode *Black box testing* digunakan dalam penelitian ini untuk mengevaluasi fungsionalitas sistem berdasarkan spesifikasi yang telah ditetapkan, tanpa memerlukan pengetahuan mengenai struktur internal maupun kode program yang digunakan. Pendekatan ini menempatkan pengujian dari sudut pandang pengguna akhir sehingga setiap fitur dapat dinilai apakah telah bekerja sesuai kebutuhan fungsional yang dirancang. Melalui teknik ini, berbagai skenario input diuji untuk memastikan keluaran yang dihasilkan sesuai dengan perilaku yang diharapkan, mencakup validasi proses bisnis, alur kerja sistem, serta kesesuaian antar komponen fungsional. Selain mendeteksi fungsi yang tidak berjalan, pengujian ini juga membantu menemukan kesalahan antarmuka, masalah akses data, hingga potensi gangguan performa yang mempengaruhi pengalaman pengguna. Meskipun tidak mampu mengidentifikasi kesalahan pada level internal karena keterbatasan akses terhadap kode, metode ini tetap menjadi alat verifikasi yang efektif dalam menilai kualitas fungsional sistem sebelum diterapkan secara luas.