

BAB II

KERANGKA TEORITIS

A. Landasan Teori

1. Information Retrieval (IR)

Information retrieval (IR) adalah proses untuk menemukan informasi yang relevan yang dibutuhkan oleh pengguna dari kumpulan data yang sudah tersimpan. Salah satu tantangan dalam pencarian informasi adalah menentukan informasi mana yang paling sesuai dengan permintaan pengguna tertentu. Agar dapat memenuhi kebutuhan informasi tersebut, sistem IR menerapkan berbagai metode dan teknik untuk memprediksi apa yang diinginkan pengguna. Hasil pencarian menampilkan informasi yang relevan di bagian teratas, sedangkan informasi yang kurang relevan muncul di bawahnya. Dengan demikian, pengguna dapat dengan cepat menemukan informasi yang dibutuhkan. IR saat ini telah diterapkan di berbagai bidang kehidupan manusia. Perpustakaan, museum, manajemen dokumen, manajemen pengetahuan, manajemen koleksi, dan manajemen arsip telah ada sebelum penerapan mesin pencari web atau sering disebut *search engine*. Penerapan IR dalam mesin pencari adalah contoh yang tepat tentang bagaimana manusia sangat terbantu dalam memperoleh informasi yang mereka butuhkan dengan memberikan permintaan berupa kata kunci atau bahkan rangkaian kalimat tertentu (AHMAD NURAINI et al., 2021).

2. Metode *Crawling*

Menurut Lathifah et al., (2021) *crawling* adalah proses pengambilan atau pengumpulan data yang dapat dilakukan secara otomatis, data yang disesuaikan dengan kebutuhan analisis. Menurut (Kosasih et al., 2022) metode *crawling* adalah teknik pengumpulan data yang menggunakan informasi indeks pada halaman web melalui URL (Uniform Resource Locator). Iskandar et al. (2024) menjelaskan bahwa *crawling* data adalah proses otomatis yang dilakukan untuk mengambil atau menggali data dari suatu situs web. Salah satu aplikasi *crawling* yang paling dikenal adalah pada mesin pencari seperti Google, Yahoo, dan Bing. Mesin pencari ini menggunakan bot untuk secara terus-menerus menjelajahi situs web, mengindeks halaman-halaman baru, dan memperbarui informasi yang terdapat dalam indeks mereka. Bot tersebut memulai kunjungannya dari URL awal dan kemudian

mengikuti tautan yang ada di halaman untuk menemukan halaman-halaman baru.

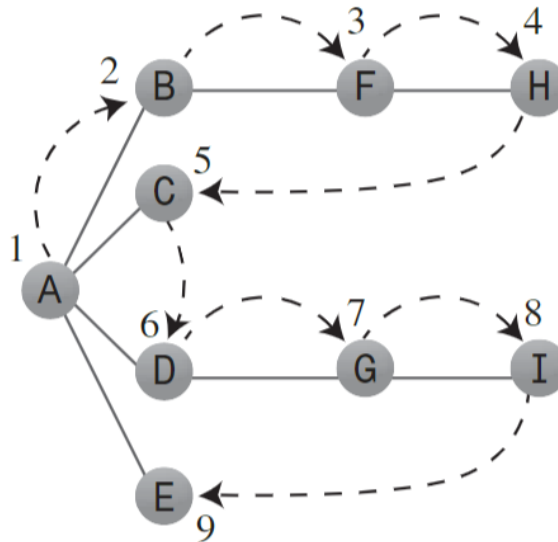
Menurut Mitchell (2018), *web crawler* beroperasi dengan cara mengekstraksi konten dari suatu URL dan memindai halaman tersebut secara rekursif untuk menemukan serta mengunjungi tautan baru secara berkelanjutan. Dalam skala besar, efektivitas sistem ini bergantung pada arsitektur kode yang fleksibel, yang diwujudkan melalui penggunaan *Class* Python untuk memisahkan logika data dan navigasi; di mana kelas *Website* berfungsi sebagai "peta" berisi instruksi tag HTML (seperti *titleTag*), sementara kelas *Crawler* bertindak sebagai mesin eksekutor yang memanfaatkan pustaka *BeautifulSoup*. Pendekatan ini memungkinkan skalabilitas melalui penggunaan *subclasses* (seperti *Product* atau *Article*) untuk normalisasi data yang berbeda dalam satu domain, serta didukung oleh implementasi pemrosesan paralel melalui modul *_thread* dan *multiprocessing* guna meningkatkan performa secara signifikan. Untuk menjaga integritas data dalam lingkungan paralel, penggunaan *Queue* yang bersifat *thread-safe* sangat krusial dalam mendistribusikan tugas dari *task delegator* ke proses pengunduhan sekaligus mencegah terjadinya *race condition* pada basis data seperti *MySQL*. Seluruh ekosistem ini, yang juga mengintegrasikan fungsi *time.sleep()* sebagai bentuk etika *polite crawling* untuk menghindari pemblokiran IP, membentuk sebuah model *web crawler* profesional yang tidak hanya cepat, tetapi juga adaptif dan mudah dirawat dalam menghadapi perubahan struktur situs web di dunia nyata.

3. Algoritma Depth-First Search (DFS)

Menurut Siswanto (2011), Depth-First Search (DFS) merupakan salah satu algoritma pencarian pada struktur data graf yang digunakan untuk menelusuri seluruh simpul (vertex) yang terhubung dari suatu titik awal. Metode ini bekerja dengan cara menjelajah graf sedalam mungkin terlebih dahulu sebelum kembali ke simpul sebelumnya untuk mengeksplorasi jalur lain. DFS umumnya diimplementasikan menggunakan struktur data *stack*, yang berfungsi untuk menyimpan jejak lintasan yang telah dilalui selama proses pencarian.

Berdasarkan Gambar 2.4, proses pencarian dimulai dari vertex A sebagai titik awal. Angka-angka yang tertera pada setiap vertex menunjukkan urutan kunjungan selama proses DFS berlangsung. Dari A, algoritma bergerak ke vertex yang berdekatan dan belum dikunjungi, yaitu B, kemudian dilanjutkan

ke F, dan selanjutnya ke H. Pada titik H, tidak terdapat lagi vertex yang dapat dikunjungi, sehingga kondisi ini disebut sebagai jalan buntu.



Gambar 2. 1 Pencarian Depth-First (Siswanto, 2011)

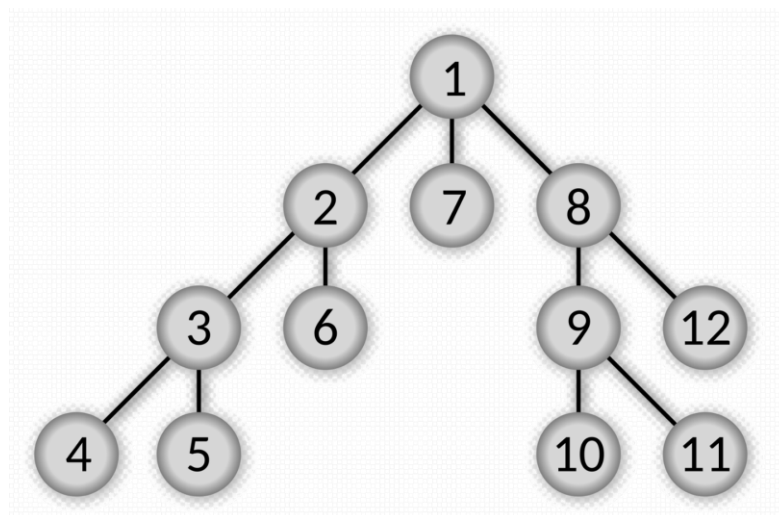
Ketika mencapai jalan buntu, algoritma akan kembali ke vertex sebelumnya (backtracking) dengan cara mengeluarkan elemen dari stack. Dari H kembali ke F, kemudian ke B, hingga akhirnya kembali ke A untuk melanjutkan pencarian ke vertex lain yang belum dikunjungi seperti C, D, G, I, dan E. Proses ini berlangsung hingga seluruh vertex yang terhubung telah dikunjungi.

Dalam pelaksanaannya, algoritma Depth-First Search mengikuti tiga aturan utama sebagai berikut:

- a. Rule 1: Jika memungkinkan, kunjungi vertex yang berdekatan dan belum dikunjungi, kemudian tandai dan masukkan ke dalam stack.
- b. Rule 2: Jika tidak ada vertex berdekatan yang dapat dikunjungi, maka lakukan *backtracking* dengan mengeluarkan (pop) vertex dari stack.
- c. Rule 3: Jika Rule 1 dan Rule 2 tidak dapat dilakukan, maka proses pencarian dinyatakan selesai.

Metode crawling berbasis DFS, yang dikenal juga sebagai *depth-first crawling*, telah digunakan dalam berbagai penelitian, salah satunya dalam ekstraksi informasi destinasi wisata populer Jawa Timur. Penelitian tersebut menunjukkan bahwa DFS dapat menelusuri tautan dari sebuah situs sedalam

mungkin sebelum kembali untuk menelusuri tautan lain, sehingga informasi yang relevan dapat diperoleh dari jalur yang lebih dalam pada struktur situs. Keunggulan pendekatan ini antara lain kemudahan implementasi, kemampuan menemukan konten yang tersembunyi di level dalam, serta penggunaan memori yang relatif lebih efisien dibandingkan metode Breadth-First Search (BFS). Namun, kelemahannya adalah potensi terjebak dalam jalur panjang atau siklus apabila tidak dilakukan penandaan terhadap simpul yang sudah dikunjungi (*visited*), serta kurang optimal ketika target informasi justru berada pada simpul yang lebih dangkal atau menyebar secara lebar (Kristanto & Hakim, 2021).



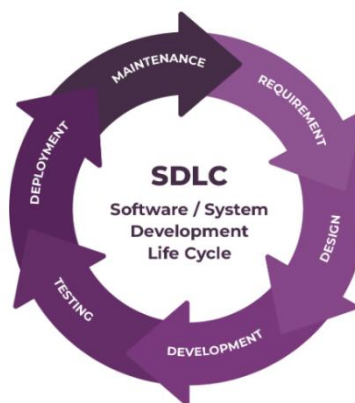
Gambar 2. 2 Depth-First Search (Teguh Yuwono & Fadlil, 2016)

Pada gambar 2.5, merupakan ilustrasi penerapan algoritma Depth First Search (DFS) pada sebuah pohon (tree). Pada algoritma ini, penjelajahan dilakukan dengan menelusuri simpul (node) sedalam mungkin sebelum kembali (backtracking) ke simpul sebelumnya. Penelusuran dimulai dari simpul akar bernomor 1, kemudian bergerak ke kiri menuju simpul 2, dilanjutkan ke simpul 3, lalu ke simpul 4. Karena simpul 4 tidak memiliki anak, penelusuran kembali ke simpul 3, kemudian ke 5, lalu mundur lagi untuk mengunjungi 6. Setelah seluruh cabang kiri selesai, penelusuran kembali ke simpul akar untuk melanjutkan ke cabang kanan melalui simpul 7, kemudian ke 8, terus ke 9, 10, dan 11, terakhir ke 12. Proses ini menggambarkan prinsip DFS yaitu menjelajah cabang terdalam terlebih dahulu sebelum beralih ke cabang lain.

Pada buku yang ditulis oleh Siswanto (2011), dijelaskan bahwa efisiensi algoritma graf sangat dipengaruhi oleh cara representasi graf, yaitu adjacency matrix dan adjacency list. Penggunaan adjacency matrix umumnya menghasilkan kompleksitas waktu sebesar $O(V^2)$ karena setiap vertex harus memeriksa seluruh kemungkinan hubungan dengan vertex lain, sehingga semua sel dalam matriks (sebanyak V^2) harus diuji. Pendekatan ini kurang efisien terutama pada graf yang jarang (sparse graph), yaitu graf yang memiliki jumlah edge relatif sedikit. Sebaliknya, dengan menggunakan adjacency list, efisiensi dapat ditingkatkan karena hanya edge yang benar-benar ada saja yang diperiksa. Oleh karena itu, untuk graf tidak berbobot, algoritma Depth-First Search (DFS) memiliki kompleksitas waktu $O(V+E)$, di mana V adalah jumlah vertex dan E adalah jumlah edge, yang menunjukkan bahwa setiap vertex dan edge hanya dikunjungi satu kali. Sementara itu, pada graf berbobot seperti pada algoritma minimum spanning tree dan shortest path, kompleksitasnya menjadi $O((V+E)\log V)$, sedangkan algoritma seperti Warshall dan Floyd memiliki kompleksitas yang lebih tinggi yaitu $O(V^3)$ akibat penggunaan perulangan bersarang.

4. Software Development Life Cycle (SDLC)

System Development Life Cycle (SDLC) adalah serangkaian langkah yang dikerjakan oleh analis sistem informasi untuk mengembangkan sistem (Alam et al., 2022).



Gambar 2. 3 Software Development Life Cycle (SDLC)

a) Tahap Identifikasi Kebutuhan

Pada tahap identifikasi kebutuhan atau tahapan mengumpulkan dan menganalisis kebutuhan pengguna. Hasilnya dituangkan dalam spesifikasi kebutuhan perangkat lunak dan dokumen kebutuhan bisnis,

yang memberikan gambaran tentang ruang lingkup proyek dan risiko yang ada.

b) Desain Sistem

Tahapan menjelaskan bagaimana menyusun dokumen desain yang menjadi panduan pengembangan. Mulai merancang antarmuka dan alur data, memastikan prioritas dan peta produk untuk keselarasan dengan tujuan proyek.

c) Tahap Pengembangan Kode

Tahap Pengembangan kode ini dilakukan sesudah desain sistem telah di rancang kemudian mengimplementasikan ke skrip kode di mana menulis kode dalam bahasa yang sesuai untuk membangun sistem.

d) Tahap Pengujian Kualitas

Pada tahap ini pengembang akan mempresentasikan sistem perangkat lunak yang telah memenuhi kriteria yang tercantum dalam dokumen kebutuhan fungsional.

e) Tahap Peluncuran Produk

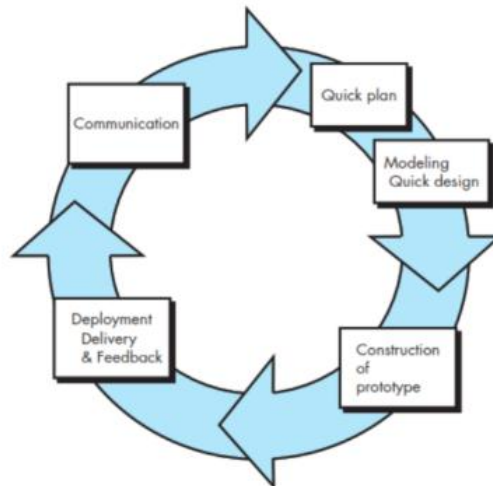
Setelah pengujian selesai dan bug atau kesalahan ditemukan dalam tahap pengujian teratasi, tahap ini melibatkan penempatan perangkat lunak ke dalam lingkungan kerja. Memungkinkan pengguna untuk mengakses.

f) Tahap Pemeliharaan Produk

Tahap ini menjelaskan kegiatan yang dilakukan untuk mengoperasikan dan memelihara sistem informasi di area produksi atau lingkungan pengguna.

5. Model Prototipe

Model prototipe merupakan salah satu pendekatan dalam rekayasa perangkat lunak yang memungkinkan pengembang untuk menunjukkan komponen fungsional perangkat lunak dan mengevaluasi kondisi awalnya. Proses ini dimulai dengan pengumpulan persyaratan dari pengembang dan klien, yang bertujuan untuk menetapkan tujuan keseluruhan perangkat lunak serta mendefinisikan semua persyaratan yang diperlukan. Model ini memberikan pengguna gambaran awal tentang program yang sedang dikembangkan. Selain itu, prototipe membantu mengidentifikasi dan memperbaiki kesalahpahaman antara pengembang sistem dan pengguna, serta sangat berguna untuk verifikasi manajemen proyek yang fleksibel, memastikan keberlanjutan proyek (Rahman Prehanto, D. 2020).



Gambar 2. 4 Pengembangan Sistem Model Prototyping

Menurut Widarti et al. (2024) pengembangan prototipe sistem menjadi fokus utama sebelum versi final dikembangkan. Dalam proses ini, klien berperan aktif dengan memberikan masukan yang berharga. Selain itu, penggunaan prototipe memungkinkan tim untuk mengidentifikasi dan mengatasi masalah sejak tahap awal, sehingga meningkatkan efektivitas dan efisiensi pengembangan sistem secara keseluruhan.

6. Unified Modeling Language (UML)

Unified Modeling Language (UML) merupakan himpunan aturan visual yang digunakan untuk menggambarkan, merancang, dan mendokumentasikan sistem perangkat lunak, khususnya yang berbasis objek. UML berfungsi sebagai bahasa standar yang menyediakan simbol, struktur, serta tata cara pemodelan agar gagasan pengembang dapat divisualisasikan secara konsisten. Dalam penggunaannya, UML membantu proses komunikasi antar-pengembang, memunculkan ide baru, menguji rancangan, serta memahami hubungan antar-komponen sistem. Pendekatan ini menempatkan UML sebagai metode pemodelan berorientasi objek yang menitikberatkan pada representasi struktur statis dan perilaku dinamis sistem, bukan pada pemodelan proses tradisional. UML juga menyediakan berbagai jenis diagram dalam tiga kelompok utama yaitu: structure, behavior, dan interaction yang bekerja layaknya cetak biru dalam proses pembangunan perangkat lunak (Henderi et al., 2021).

Sebagai bahasa pemodelan visual, UML dapat digunakan untuk merancang perangkat lunak berorientasi objek maupun prosedural, meskipun lebih optimal pada pendekatan objek. UML menyediakan aturan serta simbol

pemodelan yang membantu menggambarkan fungsi, alur, interaksi, dan struktur sistem. Dalam penelitian ini, pemodelan dilakukan menggunakan empat diagram yaitu use case diagram, activity diagram, sequence diagram, dan class diagram yang berfungsi merepresentasikan kebutuhan dan rancangan sistem dari berbagai sudut pandang. Dengan demikian, UML menjadi alat komunikasi yang efektif untuk menyamakan pemahaman antara pengembang dan pemangku kepentingan terkait rancangan sistem yang dibangun.

Pada buku Dasar-Dasar Rekayasa Perangkat Lunak (Sumirat et al., 2023), dijelaskan bahwa konsep use case, activity, sequence, dan class diagram merupakan bagian dari diagram UML yang digunakan untuk menggambarkan kebutuhan, alur proses, interaksi, serta struktur sistem, sebagai berikut :

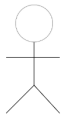
a. Use Case Diagram

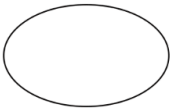
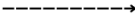
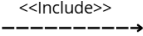
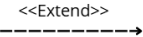
Use Case merupakan diagram yang menjelaskan hubungan antara aktor dan sistem dalam mencapai suatu tujuan tertentu. Diagram ini digunakan untuk menggambarkan fungsi-fungsi yang dapat dilakukan pengguna, alur interaksi, serta batasan peran setiap aktor dalam sistem. Representasi ini membantu memahami kebutuhan sistem dari sudut pandang pengguna dan memastikan setiap proses penting telah terstruktur dengan jelas.

Komponen utama dalam Use Case Diagram mencakup tiga elemen dasar yaitu:

- 1) Sistem, digambarkan sebagai pembatas ruang lingkup fungsi yang dimodelkan.
- 2) Aktor, merepresentasikan pihak luar yang berinteraksi dengan sistem.
- 3) Use Case, yaitu fungsi atau layanan yang disediakan oleh sistem untuk aktor.

Tabel 2. 1 Simbol Use Case Diagram

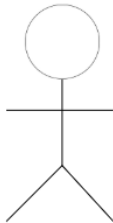
Simbol	Keterangan
	Aktor: merepresentasikan peran pengguna, sistem lain, atau perangkat yang berinteraksi dengan suatu use case.

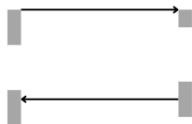
Simbol	Keterangan
	Use Case: menggambarkan bentuk abstraksi interaksi antara aktor dan sistem.
	Association: menunjukkan hubungan atau koneksi antara aktor dengan sebuah use case.
	Generalisasi: menggambarkan adanya aktor khusus (spesialisasi) yang dapat melakukan peran dari aktor yang lebih umum.
	Includes: menyatakan bahwa suatu use case sepenuhnya memanfaatkan fungsionalitas use case lain sebagai bagian dari prosesnya.
	Extends: menunjukkan bahwa sebuah use case dapat menambahkan perilaku atau fungsi tambahan pada use case lain ketika kondisi tertentu terpenuhi.

Sumber: (Sumirat et al., 2023)

b. Sequence diagram

Tabel 2. 2 Simbol Sequence Diagram

Simbol	Keterangan
	Aktor: merupakan elemen yang mewakili pengguna atau pihak luar yang berinteraksi dengan sistem. Dalam sequence diagram, aktor biasanya ditampilkan menggunakan simbol berbentuk stick figure sebagai penanda pihak yang memicu atau menerima interaksi.

Simbol	Keterangan
	Activation box: menunjukkan durasi suatu objek menjalankan operasi. Semakin lama prosesnya, semakin panjang bentuk persegi panjang yang digambarkan pada lifeline.
	Lifeline: komponen ini ditampilkan sebagai garis vertikal putus-putus. Pada bagian atasnya biasanya terdapat kotak berisi nama objek yang menunjukkan aktivitas atau keberadaan objek tersebut dalam interaksi.
	Objek: komponen ini ditampilkan menggunakan simbol dan nama yang mewakili objek tersebut. Fungsinya adalah mencatat atau merepresentasikan perilaku objek di dalam sistem. Contohnya meliputi Boundary, Entity, dan Aktor.
	Message: komponen ini berfungsi menampilkan proses komunikasi antara objek. Pesan yang dikirim ditunjukkan secara berurutan di sepanjang lifeline dan digambarkan dengan panah. Pada diagram urutan, inti interaksi terutama terlihat dari lifeline dan rangkaian pesan ini.

Sumber: (Sumirat et al., 2023)

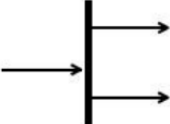
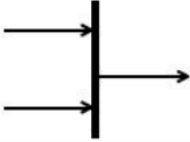
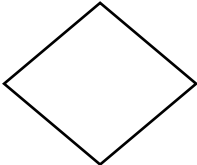
Sequence diagram digunakan untuk menggambarkan interaksi antar objek dalam sistem secara berurutan, termasuk pesan yang dikirim dan waktu terjadinya. Objek ditampilkan secara horizontal dengan lifeline vertikal, sedangkan pesan digambarkan sebagai panah antar lifeline. Diagram ini menunjukkan langkah-langkah yang terjadi sebagai respons suatu peristiwa dan biasanya berkaitan dengan satu use case. Tujuan

utamanya adalah memperjelas urutan kejadian, alur kerja, serta aliran data dalam proses sistem.

c. Activity diagram

Activity diagram atau diagram aktivitas merupakan salah satu pemodelan yang membantu menggambarkan proses bisnis sistem secara keseluruhan. Diagram ini menunjukkan rangkaian aktivitas dan alur proses yang saling berkaitan dalam sistem. Fokus utamanya adalah memodelkan aktivitas yang dilakukan oleh sistem, bukan aktivitas dari actor. Activity diagram lebih menekankan pada proses-proses serta jalur aktivitas pada level yang lebih umum.

Tabel 2. 3 Simbol Activity Diagram

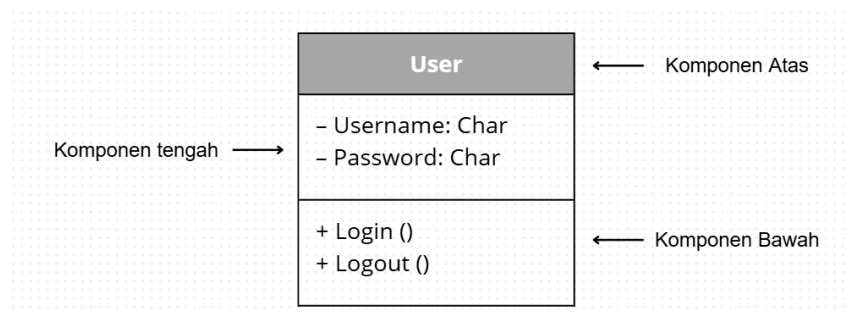
Simbol	Keterangan
	Initial Menandai titik mulai dari rangkaian aktivitas atau aksi.
	Final Menunjukkan akhir dari seluruh aliran kontrol atau objek dalam suatu aksi ataupun aktivitas.
	Activities Merepresentasikan satu set aktivitas atau tindakan yang dilakukan dalam proses.
	Fork (Percabangan) Digunakan untuk memisahkan satu alur menjadi beberapa tindakan atau aktivitas yang berjalan secara paralel.
	Join (Penggabungan) Berfungsi untuk menyatukan kembali beberapa aliran paralel menjadi satu jalur aktivitas.
	Decision Melambangkan titik keputusan yang digunakan untuk memeriksa kondisi sebelum menentukan arah alur selanjutnya.

Simbol	Keterangan
	Swimlane Merupakan kolom yang dipakai untuk mengelompokkan aktivitas berdasarkan actor atau pihak yang bertanggung jawab.

Sumber: (Andriyanto, 2022)

d. Class diagram

Class Diagram adalah diagram struktur UML yang digunakan untuk menunjukkan susunan kelas beserta atribut, metode, dan relasi antar kelas dalam sebuah sistem. Diagram ini bersifat statis karena berfokus pada struktur, bukan alur proses, sehingga sangat cocok untuk perancangan berbasis objek. Secara umum, class diagram membantu memvisualisasikan rancangan sistem, memahami hubungan antar komponen, serta mendukung proses analisis dan desain perangkat lunak.



Gambar 2. 5 Komponen Class Diagram (Sumirat et al., 2023)

Class diagram terdiri dari tiga komponen bisa dilihat pada gambar 2.3 yaitu bagian atas untuk nama kelas, bagian tengah untuk atribut, dan bagian bawah yang memuat operasi atau metode. Selain itu, hubungan antar kelas ditunjukkan melalui beberapa relasi utama, yaitu :

- 1) Asosiasi sebagai hubungan dasar antar kelas,
- 2) Agregasi yang menggambarkan hubungan “bagian dari” namun tetap dapat berdiri sendiri,
- 3) Serta generalisasi atau pewarisan yang menunjukkan bahwa sebuah subclass menerima atribut dan metode dari superclass-nya.

7. Rekomendasi Buku Referensi

Rekomendasi buku referensi adalah sebuah teknologi yang dirancang untuk membantu pengguna menemukan buku yang relevan dengan

kebutuhan dan minat pengguna. Sistem ini menjadi sangat penting di perpustakaan modern yang menyaring informasi dari koleksi buku yang sangat melimpah. Dengan adanya sistem ini, pengguna tidak lagi kewalahan dan dapat menemukan bacaan yang sesuai secara lebih cepat dan personal.

Dalam penelitian ini, rekomendasi buku referensi berfungsi sebagai solusi langsung untuk masalah yang dihadapi mahasiswa, yaitu keterbatasan koleksi dan tidak relevannya buku di perpustakaan induk. Melalui penerapan metode crawling, sistem ini dapat mengambil data dari perpustakaan lain, sehingga rekomendasi tidak terbatas pada koleksi internal. Ini secara langsung menjawab kebutuhan mahasiswa yang membutuhkan referensi untuk penelitian dan tugas akhir yang tidak tersedia di perpustakaan mereka. Dengan demikian, sistem rekomendasi ini tidak hanya meningkatkan efisiensi dan pengalaman pengguna, tetapi juga memperkuat peran perpustakaan sebagai pusat layanan informasi yang adaptif dan responsif.

B. Tinjauan Pustaka

Tinjauan pustaka merupakan acuan penting yang diperlukan dalam sebuah penelitian. Tinjauan ini diambil berdasarkan permasalahan yang serupa atau mendekati, serta metode yang mirip dengan penelitian yang ada. Berikut penelitian yang memiliki kemiripan antara lain:

1. Implementasi Website Berita Online Menggunakan Metode Crawling Data Dengan Bahasa Pemrograman Python. (Algifari Rismawan et al., 2023).

Kesimpulan dari penelitian ini adalah implementasi website berita online dengan metode crawling data menjadi semakin krusial di era digital saat ini. Metode ini memungkinkan pengambilan data dari berbagai sumber secara efisien dan terstruktur, sehingga bermanfaat untuk keperluan seperti penelitian, analisis, dan pengembangan aplikasi. Dalam penelitian ini, akan dibangun sebuah website berita online menggunakan metode crawling data dengan bahasa pemrograman Python. Metode pengembangan yang digunakan adalah waterfall. Penelitian sebelumnya telah menunjukkan bahwa web crawling merupakan metode efektif untuk mengumpulkan dan mengambil informasi dari sebuah website dengan hasil yang akurat. Beberapa studi sebelumnya telah menerapkan metode ini untuk mengumpulkan informasi terkait tempat kuliner, harga suku cadang, serta kebutuhan masyarakat selama pandemi Covid-19 di media sosial. Web crawling adalah program yang beroperasi dengan cara tertentu dan secara otomatis mengumpulkan semua berita nasional dari suatu website. Metode crawling bekerja dengan menjelajahi setiap URL yang diberikan, mengunjungi URL tersebut satu per

satu. Setiap halaman URL akan diidentifikasi untuk menentukan apakah URL tersebut sudah ada dalam database. Jika ada URL yang belum terdaftar, maka akan ditambahkan ke dalam database. Penelitian ini diharapkan dapat memberikan kontribusi dan inspirasi bagi pengembangan website berita online menggunakan metode crawling data dengan bahasa pemrograman Python.

2. IMPLEMENTASI WEB CRAWLING PADA WEBSITE BPS JAWA TIMUR PADA DATA INFLASI DI JAWA TIMUR. (Danang et al., 2024).

Web crawling adalah program yang dirancang untuk mengumpulkan informasi dari internet dengan tujuan memperoleh data yang diperlukan, seperti teks, gambar, tautan, dan metadata. Tujuan utamanya mencakup penulisan, pengindeksan mesin pencari, pemantauan perubahan, analisis situs web, dan berbagai keperluan lainnya. Terdapat beberapa jenis web crawling, seperti news crawling, social media crawling, video crawling, image crawling, dan email crawling, masing-masing fokus pada pengumpulan informasi spesifik dari sumber yang berbeda.

Dalam konteks penulisan, penggunaan web crawling memudahkan akses data dari situs resmi, seperti BPS Jawa Timur, untuk topik-topik seperti inflasi. Data tersebut dapat dieksplorasi dengan langkah-langkah tertentu di dalam website, memungkinkan informasi dipindahkan ke Excel tanpa perlu mengunduhnya terlebih dahulu, sehingga meningkatkan efisiensi dalam penulisan. Selain itu, pelatihan dan penyuluhan mengenai cara memanfaatkan website BPS juga dilakukan untuk memberikan pemahaman lebih luas tentang fungsi dan akses data dari BPS. Dengan demikian, web crawling menjadi alat yang sangat bermanfaat untuk mengoptimalkan akses dan penggunaan informasi dalam konteks penulisan.

3. IMPLEMENTASI WEB CRAWLER UNTUK Mencari Harga Barang TERMURAH DARI BERBAGAI SITUS E-COMMERCE INDONESIA. (Gusniawan Pradana, 2022).

Dari penelitian ini, dapat disimpulkan beberapa poin penting. Pertama, teknik Web Crawler terbukti efektif untuk melakukan pencarian dan perbandingan harga produk di berbagai situs e-commerce. Kedua, hasil pencarian yang diperoleh melalui Web Crawler konsisten dengan informasi yang terdapat di situs asli. Selain itu, metode ini mampu menampilkan harga produk terendah, memberikan keuntungan bagi pengguna yang mencari penawaran terbaik. Namun, waktu respons Web Crawler juga dipengaruhi oleh beberapa faktor, seperti kestabilan koneksi internet, waktu respons

masing-masing server e-commerce, ukuran gambar, dan struktur HTML situs yang dikunjungi.

4. SISTEM CRAWLING DATA INSTRUMEN AKREDITASI BERBASIS SELENIUM DAN PANDAS. (Lathifah et al., 2021).

Implementasi sistem crawling data menunjukkan bahwa proses pengambilan data bervariasi tergantung pada tampilan antarmuka setiap laman web. Pada laman web Eduk Undip, pengambilan data dilakukan dengan mengunduh berkas dalam format JSON, tanpa perlu melakukan filtrasi data. Di sisi lain, pada laman web Prestasi Undip dan Foplap Ristekdikti, data diambil dengan memasukkan informasi ke dalam daftar. Sementara itu, laman web Sip3mu Undip menggunakan metode pengunduhan berkas dalam format Excel untuk pengambilan data.

5. Perancangan Aplikasi Data Crawling Untuk Pencarian Buku Pada Toko Buku Online. (Ronald Talisman Bryan et al., 2024)

Penelitian mengenai perancangan aplikasi data crawling untuk pencarian buku di toko buku online bertujuan untuk mengembangkan sebuah sistem yang dapat mengumpulkan informasi secara otomatis dari berbagai sumber di internet. Dengan menggunakan teknik crawling, aplikasi ini mengakses dan mengekstrak data penting seperti judul, harga, dan ketersediaan buku dari toko buku online seperti Grobmart dan Deepublishstore. Metode pengembangan yang digunakan adalah Rapid Application Development (RAD), yang memungkinkan proses pengembangan berjalan cepat dan efisien, serta penggunaan teknologi seperti Python dan MySQL untuk mendukung fungsionalitas aplikasi.

Hasil dari penelitian ini menunjukkan bahwa aplikasi yang dikembangkan dapat mengumpulkan total 31.785 data buku, memberikan pengguna akses yang lebih mudah untuk menemukan buku yang diinginkan. Aplikasi search engine "Find Your Book" memungkinkan pencarian berdasarkan kriteria tertentu, sehingga pengguna dapat membandingkan harga dan ketersediaan dari berbagai toko. Dengan demikian, penelitian ini tidak hanya memberikan solusi untuk masalah pencarian buku yang efisien, tetapi juga membuka peluang untuk pengembangan lebih lanjut di bidang data crawling untuk aplikasi lain, seperti pencarian tiket atau pemesanan layanan.

6. RANCANG BANGUN SISTEM REKOMENDASI BUKU BERBASIS ITEM-BASED COLLABORATIVE FILTERING MENGGUNAKAN ALGORITMA K-NEAREST NEIGHBORS. (Saputra et al., 2025)

Penelitian ini menegaskan bahwa keberadaan sistem rekomendasi buku di perpustakaan merupakan kebutuhan yang semakin mendesak di era digital. Dengan semakin melimpahnya koleksi buku, baik dalam bentuk cetak maupun digital, pengguna kerap mengalami kesulitan menemukan bacaan yang sesuai dengan minat dan kebutuhannya. Sistem rekomendasi hadir sebagai solusi strategis untuk mengatasi masalah tersebut dengan memberikan daftar bacaan yang relevan, personal, dan adaptif. Dalam konteks penelitian ini, penerapan sistem rekomendasi mampu menyaring informasi berlebih, mempercepat proses pencarian, serta meningkatkan pengalaman pengguna dalam berinteraksi dengan perpustakaan. Hal ini menunjukkan bahwa sistem rekomendasi bukan sekadar fitur tambahan, melainkan komponen inti yang memperkuat peran perpustakaan sebagai pusat layanan informasi yang modern dan responsif terhadap kebutuhan pengguna.

Penelitian ini juga memperlihatkan peran signifikan sistem rekomendasi dalam mendukung budaya literasi dan meningkatkan minat baca. Dengan menyajikan pilihan buku yang sesuai dengan preferensi pembaca, sistem rekomendasi membantu memperluas wawasan pengguna, mendorong eksplorasi bacaan baru, serta memperkuat motivasi dalam mengakses koleksi perpustakaan. Keberhasilan sistem dalam memberikan rekomendasi yang tepat sasaran juga berdampak pada meningkatnya pemanfaatan koleksi yang sebelumnya kurang dikenal atau jarang dipinjam. Dengan demikian, sistem rekomendasi tidak hanya meningkatkan kepuasan pengguna secara individual, tetapi juga memberi kontribusi terhadap optimalisasi fungsi perpustakaan dalam menyediakan akses informasi yang inklusif, efektif, dan berkelanjutan.

7. Sistem Rekomendasi Penelusuran Buku Berbasis Content-Based Filtering dengan Pembobotan TF-RF. (Rahmadhani et al., 2024)

Penelitian ini berhasil menunjukkan bahwa penerapan metode Content-Based Filtering yang dipadukan dengan algoritma Term Frequency – Relevance Frequency (TF-RF) dapat secara signifikan meningkatkan performa sistem rekomendasi buku di E-Library Politeknik Negeri Banyuwangi. Dengan dataset sebanyak 6.148 judul buku berbahasa Indonesia, sistem diuji menggunakan metrik presisi. Hasil evaluasi memperlihatkan bahwa sistem rekomendasi yang dikembangkan mampu mencapai presisi rata-rata 86%, jauh lebih tinggi dibandingkan sistem pencarian sebelumnya yang hanya memiliki akurasi 51%. Hal ini membuktikan bahwa pendekatan yang diusulkan mampu memperbaiki akurasi dan relevansi rekomendasi buku, sekaligus

mengatasi masalah lemahnya pencarian yang kerap ditemui pada sistem lama, misalnya kasus kata kunci panjang yang gagal menemukan data padahal buku tersedia di database.

Penelitian ini memberikan kontribusi penting dalam konteks perpustakaan digital, karena mampu menghadirkan solusi nyata untuk meningkatkan kepuasan pengguna. Dengan memanfaatkan tahapan pengolahan teks seperti pre-processing (case folding, tokenizing, filtering, stemming) serta perhitungan kemiripan menggunakan cosine similarity, sistem dapat menyajikan daftar buku yang lebih relevan sesuai kebutuhan pengguna. Implikasi dari penelitian ini adalah perpustakaan kini dapat memberikan layanan yang lebih adaptif, efisien, dan mendukung pemustaka dalam menemukan referensi secara cepat. Walaupun demikian, penelitian ini juga menyoroti keterbatasan sistem, terutama ketika pengguna menggunakan kata kunci yang tidak lengkap atau multi suku kata. Maka, pengembangan lebih lanjut dapat diarahkan pada integrasi metode lain seperti hybrid recommendation atau teknik NLP yang lebih canggih untuk semakin meningkatkan kualitas rekomendasi.

8. Prediksi Rekomendasi Buku Berdasarkan Riwayat Peminjaman Menggunakan Collaborative Filtering Pada Perpustakaan Unjaya.(Aprilia Damayanti et al., 2025)

Penelitian ini menekankan pentingnya sistem rekomendasi berbasis item-based collaborative filtering dalam meningkatkan layanan perpustakaan digital di lingkungan perguruan tinggi. Dengan menggunakan adjusted cosine similarity, sistem mampu menganalisis riwayat peminjaman pengguna untuk menghasilkan rekomendasi buku yang relevan. Evaluasi menggunakan Mean Absolute Error (MAE) dengan hasil 0,0024 dan 0,0047 menunjukkan tingkat kesalahan prediksi yang sangat rendah, sehingga akurasi sistem dapat dikategorikan sangat baik. Hasil ini mengindikasikan bahwa metode collaborative filtering mampu menjawab kebutuhan pengguna yang semakin kompleks dalam pencarian informasi di era digital.

Penelitian ini memperlihatkan bahwa tantangan utama perpustakaan perguruan tinggi saat ini bukan sekadar menyediakan koleksi digital, melainkan bagaimana menghadirkan layanan yang adaptif dan personal sesuai preferensi pengguna. Data internal menunjukkan bahwa pemanfaatan e-book dan e-jurnal masih rendah karena mahasiswa lebih sering mencari sumber di luar sistem perpustakaan. Hal ini menimbulkan kesenjangan antara layanan yang disediakan dengan perilaku informasi mahasiswa. Sistem

rekomendasi hadir sebagai solusi strategis, karena mampu menyajikan koleksi yang sesuai kebutuhan secara otomatis, sehingga meningkatkan keterlibatan pengguna serta relevansi layanan perpustakaan.

Kontribusi penelitian ini bukan hanya pada tataran teknis, tetapi juga pada transformasi peran perpustakaan. Dengan sistem rekomendasi, perpustakaan tidak lagi dipandang sekadar tempat penyimpanan koleksi, melainkan sebagai pusat distribusi pengetahuan yang cerdas dan responsif. Penerapan metode ini mendukung percepatan akses informasi, meningkatkan pengalaman pengguna, serta memperkuat posisi perpustakaan sebagai bagian integral dari ekosistem akademik. Penelitian ini juga membuka peluang pengembangan sistem rekomendasi lanjutan, misalnya dengan mengombinasikan pendekatan collaborative filtering dengan metode lain (hybrid system), agar kualitas prediksi semakin akurat dan sesuai dengan kebutuhan mahasiswa di era pembelajaran digital.

9. PENGEMBANGAN SISTEM REKOMENDASI BUKU UNTUK MENINGKATKAN MINAT BACA DENGAN PENDEKATAN HYBRID FILTERING. (Verdyansyah Pratama, 2025)

Penelitian ini mengembangkan sistem rekomendasi buku digital dengan pendekatan hybrid filtering, yaitu menggabungkan collaborative filtering dan content-based filtering. Uji coba dilakukan pada data perpustakaan universitas dengan ribuan koleksi buku dan ratusan pengguna aktif. Hasil pengujian menunjukkan bahwa pendekatan hibrida ini mampu meningkatkan akurasi rekomendasi sekaligus mengurangi kelemahan yang biasa muncul pada metode tunggal, seperti masalah cold-start.

Kesimpulan penelitian menekankan bahwa sistem rekomendasi berbasis hybrid filtering sangat efektif untuk meningkatkan kualitas layanan perpustakaan digital. Dengan menggabungkan keunggulan dua metode, sistem dapat memberikan rekomendasi yang relevan bagi pengguna baru maupun pengguna lama. Hal ini memberikan kontribusi nyata dalam mendukung transformasi perpustakaan ke arah layanan yang lebih cerdas, personal, dan user-friendly.

10. Sistem Rekomendasi Pencarian Buku Perpustakaan Dengan Algoritma Content Based Filtering. (Hendrayana et al., 2024)

Penelitian ini menegaskan pentingnya sistem rekomendasi buku di perpustakaan sebagai solusi untuk mengatasi keterbatasan pencarian manual yang memakan waktu lama dan sering kali menyulitkan pengguna. Dengan adanya sistem rekomendasi, pengguna tidak lagi harus memeriksa satu per

satu koleksi di rak, melainkan dapat memperoleh daftar bacaan yang sesuai dengan kebutuhan mereka secara lebih cepat dan akurat. Hal ini menunjukkan bahwa penerapan sistem rekomendasi tidak hanya meningkatkan efisiensi pencarian, tetapi juga memperbaiki pengalaman pengguna dalam berinteraksi dengan perpustakaan. Pengguna dapat terbantu untuk menemukan buku relevan yang sesuai dengan kata kunci dan kategori yang mereka butuhkan, sehingga sistem berfungsi sebagai asisten cerdas yang menghubungkan pemustaka dengan koleksi yang paling tepat.

Sistem rekomendasi ini juga memiliki dampak signifikan terhadap fungsi perpustakaan sebagai pusat pembelajaran. Dengan mempermudah akses ke koleksi bacaan, perpustakaan mampu memperkuat perannya dalam mendukung proses belajar mengajar, menumbuhkan minat baca, serta meningkatkan kualitas layanan informasi. Hasil pengujian yang menampilkan tingkat kesamaan tertinggi pada buku "Panduan Lengkap Agama Islam Secara Kafah" menjadi bukti bahwa sistem ini dapat memberikan hasil yang konkret, jelas, dan bermanfaat. Secara keseluruhan, penelitian ini menunjukkan bahwa keberadaan sistem rekomendasi buku bukan hanya sekadar fitur tambahan, melainkan sebuah kebutuhan mendesak bagi perpustakaan modern untuk menghadirkan layanan yang lebih adaptif, personal, dan berorientasi pada kenyamanan pengguna.

Tabel 2. 4 Tinjauan Pustaka

No	Peneliti	Judul	Sumber	Kontribusi
1.	Sendi Algifari Rismawan ,Yuda Syahidin	Implementasi Website Berita Online Menggunakan Metode Crawling Data Dengan Bahasa Pemrograman Python.	https://jurnal.mdp.ac.id/index.php/jatisi/article/view/4902	Penelitian ini mengembangkan website berita online dengan metode crawling data menggunakan Python, memungkinkan pengambilan informasi secara efisien dan akurat. Hasilnya dapat diterapkan dalam aplikasi lain untuk memenuhi kebutuhan informasi digital.

No	Peneliti	Judul	Sumber	Kontribusi
2.	Raden Danang Raihan Hendrarto, Ignatia Martha Hendrati, Wiryana Wardaya	IMPLEMENTASI SI WEB CRAWLING PADA WEBSITE BPS JAWA TIMUR PADA DATA INFLASI DI JAWA TIMUR.	https://journal.alshobar.org/index.php/umat/article/view/202	Penelitian ini menunjukkan efisiensi web crawling untuk mengakses data inflasi BPS, meningkatkan pemahaman masyarakat tentang statistik resmi.
3.	Egi Gusniawan Pradana	IMPLEMENTASI SI WEB CRAWLER UNTUK MENCARI HARGA BARANG TERMURAH DARI BERBAGAI SITUS E-COMMERCE INDONESIA.	http://teknologiipintar.org/index.php/teknologiipintar/article/view/257	Penelitian ini berkontribusi pada pengembangan teknik web crawler untuk membandingkan harga barang di e-commerce secara efisien.
4.	Laila Lathifah, Eko Handoyo, dan Yosua Alvin Adi Soetrisn	Strategi Pengembangan Kerjasama Perpustakaan Uinsu dengan Perpustakaan Umsu Dalam Upaya Meningkatkan Layanan Perpustakaan.	https://pdfs.semanticscholar.org/40dc/6963a8a8c48f4ae4eb0e00101244a7c16c7e.pdf	Penelitian ini berkontribusi pada pengembangan sistem crawling data yang efisien untuk akreditasi kampus, memanfaatkan Selenium dan Pandas. Sistem ini mempercepat pengumpulan data akreditasi dan meningkatkan analisis data secara daring.

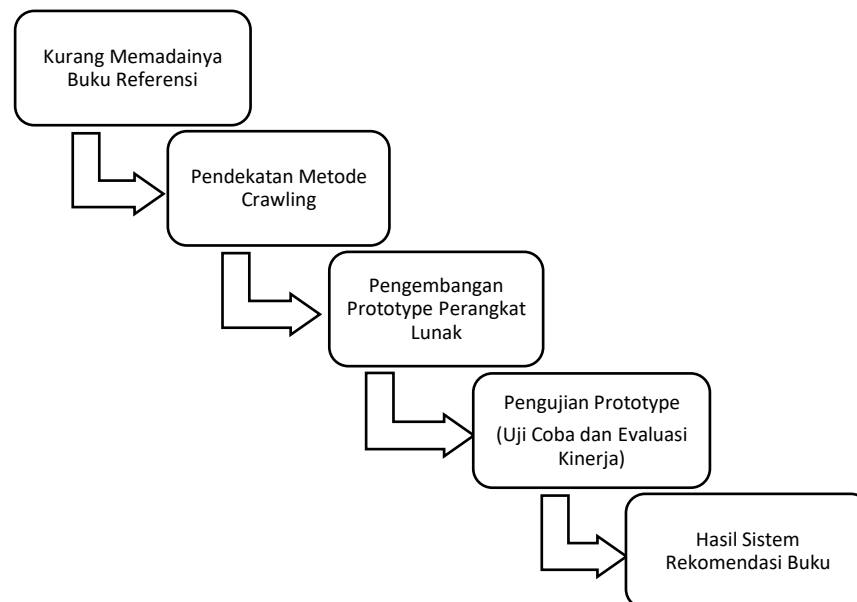
No	Peneliti	Judul	Sumber	Kontribusi
5.	Bryan Ronald Talisman, Irfan Darmaw n, Oktariani Nurul Pratiwi	Perancangan Aplikasi Data Crawling Untuk Pencarian Buku Pada Toko Buku Online.	https://reposit ori.telkomuni versity.ac.id/p ustaka/21230 8/	Penelitian ini berkontribusi dalam mengembangkan aplikasi data crawling yang efisien untuk pencarian buku online, meningkatkan aksesibilitas dan kemudahan bagi pengguna dalam menemukan informasi buku.
6.	Vernanda Sam Saputra, Achmad Ridwan, Taftazani Ghazi Pratama	Rancang Bangun Sistem Rekomendasi Buku Berbasis Item-Based Collaborative Filtering Menggunakan Algoritma K- Nearest Neighbors	https://journal .eng.unila.ac. id/index.php/ji tet/article/vie w/5995/2387	Memberikan solusi dalam menyaring informasi berlebih melalui sistem rekomendasi berbasis <i>item-based collaborative filtering</i> yang mampu meningkatkan pengalaman pengguna dalam menemukan bacaan relevan serta mendorong minat baca.
7.	Rahmadh ani, Sadesty Hakim, Lutfi Hendra Wibowo, Galih Purnama Kristanto, Sepyan	Sistem Rekomendasi Penelusuran Buku Berbasis Content-Based Filtering dengan Pembobotan TF-RF	https://jurnal. polinema.ac.i d/index.php/ji p/article/view/ 5565/3939	Mengusulkan metode content-based filtering dengan pembobotan TF- RF, menghasilkan tingkat presisi rekomendasi tinggi (86%), sehingga meningkatkan relevansi hasil pencarian buku dan kepuasan pengguna.

No	Peneliti	Judul	Sumber	Kontribusi
	Mistiko Rini, Eka			
8.	Aprilia Damayanti, Mita Siskasari, Dini Hidayatullah, Syarif Ali Fernandi, Rois Ali Fernandi, Rois	Prediksi Rekomendasi Buku Berdasarkan Riwayat Peminjaman Menggunakan Collaborative Filtering Pada Perpustakaan Unjaya	https://ejournal.unjaya.ac.id/index.php/jds/article/view/1570/925	Mengembangkan sistem rekomendasi berbasis <i>collaborative filtering</i> dengan <i>adjusted cosine similarity</i> yang terbukti akurat (MAE rendah), membantu perpustakaan menghadirkan layanan adaptif dan personal sesuai kebutuhan mahasiswa.
9.	Verdyansyah Pratama, Rafael	Pengembangan Sistem Rekomendasi Buku untuk Meningkatkan Minat Baca dengan Pendekatan Hybrid Filtering	https://jig.rivierapublishing.id/index.php/rv/article/view/255/438	Menawarkan pendekatan hybrid filtering yang menggabungkan collaborative dan content-based filtering, mampu mengatasi masalah cold-start serta meningkatkan akurasi rekomendasi pada perpustakaan digital.
10.	Hendrayana, Ndaru Sasongko Wibowo, Jati Tri Lomba Juang Semarang, Jalan	Sistem Rekomendasi Pencarian Buku Perpustakaan dengan Algoritma Content-Based Filtering	https://journal.stekom.ac.id/index.php/elektron/article/view/1927/1482	Menyediakan solusi pencarian buku yang lebih cepat dan akurat dengan content-based filtering, sekaligus memperbaiki pengalaman pengguna serta mendukung peran perpustakaan sebagai pusat pembelajaran modern.

Penelitian ini memperkenalkan metode *crawling* untuk menyempurnakan pendekatan yang telah ada, mengatasi keterbatasan informasi buku di situs internal perpustakaan, serta memperbaiki kelemahan yang muncul pada penerapan teknik rekomendasi sebelumnya. Selain memberikan rekomendasi buku, pendekatan metode *crawling* ini juga memperluas sumber informasi dengan mengumpulkan data dari berbagai katalog perpustakaan eksternal, sehingga akses informasi menjadi lebih luas dan adaptif terhadap kebutuhan pengguna.

C. Kerangka Pemikiran

Kerangka pemikiran adalah landasan konseptual yang menjabarkan hubungan sebab-akibat, mekanisme kerja, serta asumsi dasar yang mendasari seluruh rangkaian penelitian. Kerangka pemikiran ini mewakili tahapan-tahapan yang akan diambil untuk solusi masalah yang akan dibahas. Pemecahan masalah penelitian merupakan aspek mendasar dari kerangka ini, yang mencakup objek penelitian dan metode penelitian. Dalam kerang tersebut, metode penelitian yang diterapkan tidak lain adalah pendekatan metode *crawling*, berikut kerangka pemikiran sebagai konsep pemecahan masalah penelitian :



Gambar 2. 6 Kerangka Pemikiran

Pada Gambar 2. 6 tentang kerangka pemikiran dapat di jelaskan sebagai berikut :

1. Mengidentifikasi kendala dan masalah yang mendalam pada objek penelitian yaitu perpustakaan Universitas Binaniaga Indonesia Fakultas Informatika dan Komputer.

2. Pendekatan metode *crawling* untuk pengambilan data buku pada katalog online sebagai informasi rekomendasi buku.
3. Pengembangan *prototype* perangkat lunak untuk rekomendasi buku menggunakan metode *crawling*.
4. Melakukan uji coba dan mengukur efektivitas dari penerapan metode *crawling* pada sistem untuk memastikan bahwa semua fitur berfungsi dengan baik, evaluasi kinerja dilakukan untuk mengidentifikasi area yang perlu diperbaiki.
5. Hasil sistem rekomendasi buku berbasis *crawling*.

D. Hipotesis Penelitian

Penerapan metode *crawling* untuk rekomendasi buku referensi akan efektif meningkatkan aksesibilitas informasi buku bagi mahasiswa di perpustakaan universitas. Rekomendasi buku berbasis *crawling* mampu mengumpulkan data informasi buku berskala besar dari berbagai sumber secara efisien dan terstruktur, dalam penelitian (Ronald Talisman Bryan et al., 2024) berhasil mengumpulkan 31.785 data hasil *crawling*. Dalam konteks perpustakaan universitas, dengan permasalahan informasi buku yang terbatas dan kurang relevan maka penerapan metode *crawling* untuk informasi rekomendasi buku akan memberikan solusi yang efektif. Dengan pemahaman tersebut maka hipotesis dalam penelitian ini adalah penerapan metode *crawling* diduga akan mampu memecahkan permasalahan pencarian informasi buku di perpustakaan dan mengatasi keterbatasan koleksi yang di hadapi perpustakaan dengan memberikan rekomendasi informasi buku yang tersedia di perpustakaan lain di wilayah Kota Bogor.