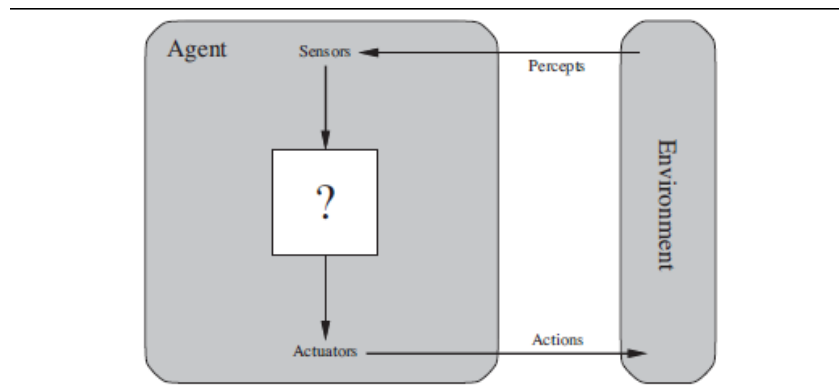


BAB II KERANGKA TEORITIS

A. Landasan Teori

1. Pengertian *Artificial Intelligence*

Artificial Intelligence dapat berupa agen dalam sistem yang berada di suatu lingkungan dan memiliki kemampuan untuk bertindak secara otomatis dalam situasi lingkungan tersebut. (Russel & Norvig, 2010, hlm. 34). Perilaku *agent* dideskripsikan oleh *agent function* di mana secara internal *agent function* yang berisi deskripsi matematika abstrak diimplementasikan oleh *agent program*. *Agent program* menerima persepsi saat ini sebagai input dari sensor dan menghasilkan output berupa tindakan kepada aktuator. Terdapat beberapa jenis program agen, yaitu agen refleks sederhana, agen refleks berbasis model, agen berbasis tujuan, dan agen berbasis utilitas. (Russel & Norvig, 2010, hlm. 46–53).

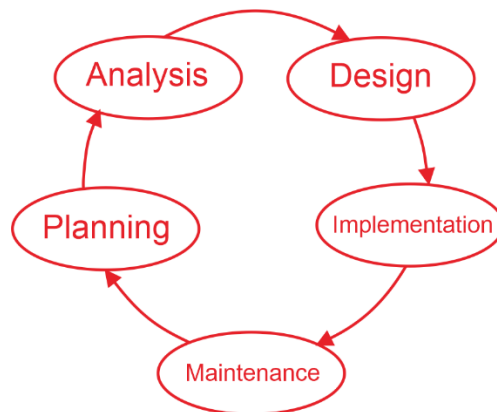


Gambar 2.1 Struktur *Agent*

(Sumber: Russel & Norvig, 2010, p. 35)

2. Pengertian SDLC

SDLC merupakan dasar bagi berbagai metodologi pengembangan perangkat lunak yang membentuk kerangka kerja untuk merencanakan dan mengendalikan pembuatan sistem informasi. SDLC (*Software Development Life Cycle*) mengubah suatu sistem perangkat lunak dengan menerapkan model dan metodologi yang digunakan untuk mengembangkan sistem - sistem perangkat lunak sebelumnya (S. & Shalahuddin, 2016, hlm. 26). SDLC memiliki tahap – tahap dalam mengembangkan sistem perangkat lunak di antaranya (Hasanah & Untari, 2020, hlm. 20):



Gambar 2.2 Model Siklus Pengembangan Sistem
(Sumber: (Hasanah & Untari, 2020, hlm. 20)

- (1) perencanaan;
- (2) analisis;
- (3) desain;
- (4) implementasi;
- (5) pengelolaan.

Model *prototyping* dikembangkan secara cepat untuk memeriksa kebutuhan persyaratan dari pengguna dan kelayakan desain yang memungkinkan pengguna untuk menguji coba lebih terlebih dahulu sistem awal sebelum disebarluaskan (Sommerville, 2016, hlm. 62). Model *prototyping* mendemonstrasikan konsep perangkat lunak, menguji coba desain, dan mengidentifikasi lebih lanjut permasalahan dan solusi yang memungkinkan. Paradigma model pengembangan *prototyping* diawali dengan langkah – langkah di bawah (Pressman & Maxim, 2010, hlm. 45):

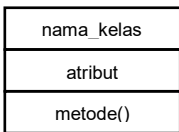



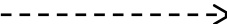
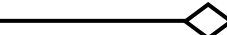
- (1) *communication*, membangun komunikasi antara pihak *developer* dengan pihak *stakeholder* untuk mendeskripsikan tujuan dari pengembangan perangkat lunak melalui identifikasi kebutuhan umum dan kebutuhan spesifik;
- (2) *quick plan and modelling quick design*, berfokus pada aspek – aspek perangkat lunak yang dapat diinteraksi langsung oleh pengguna seperti tata letak *human interface*;
- (3) *construction of prototype*, hasil dari pemodelan desain cepat selanjutnya dikembangkan ke dalam konstruksi *prototype*;
- (4) *deployment delivery & feedback*, *prototype* disebarkan kepada pihak *stakeholder*, lalu pihak *stakeholder* memberikan *feedback* sebagai evaluasi yang digunakan untuk perbaikan dan penyempurnaan *prototype* lebih lanjut.

3. Pengertian *Unified Modelling Language*

Unified Modelling Language atau UML mendefinisikan kebutuhan, pemodelan dan desain suatu sistem, dan menggambarkan arsitektur dalam pemrograman berorientasi objek (S. & Shalahuddin, 2016, hlm. 133). UML digunakan untuk memenuhi kebutuhan pemodelan visual dengan menggambarkan, dan mendokumentasikan dari sistem perangkat lunak yang dapat dimengerti oleh orang. Secara kategori, UML dikelompokkan menjadi tiga macam, di antaranya: *structure diagram*, *behavior diagram*, dan *interaction diagram* (S. & Shalahuddin, 2016, hlm. 140–141).

a. *Class Diagram*

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas – kelas yang akan dibuat (S. & Shalahuddin, 2016, hlm. 141). Kelas memiliki atribut berupa variabel yang dimiliki suatu kelas dan metode berupa fungsi yang dimiliki suatu kelas. Simbol – simbol yang terdapat pada *class diagram* terdiri dari (S. & Shalahuddin, 2016, hlm. 146–147):

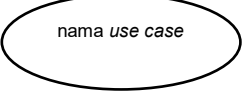
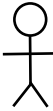



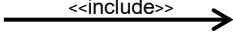
Simbol	Deskripsi
<p><i>Class</i></p> 	kelas pada struktur sistem
<p><i>Association</i></p> 	relasi antarkelas dengan makna umum.
<p><i>Directed Association</i></p> 	relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain.
<p><i>Generalization</i></p> 	relasi antarkelas dengan makna generalisasi - generalisasi
<p><i>Dependency</i></p> 	relasi antarkelas dengan makna kebergantungan antarkelas
<p><i>Aggregation</i></p> 	relasi antarkelas dengan makna semua-bagian

(Sumber: S. & Shalahuddin, 2016, pp. 146–147)

b. *Use Case Diagram*

Use case diagram mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi melalui fungsi – fungsi yang ada (S. &


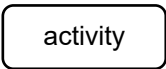
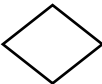


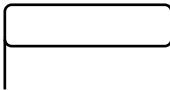
Shalahuddin, 2016, hlm. 155). Komponen *use case* terdiri dari aktor yang berinteraksi dengan sistem informasi dan *use case* yang merupakan fungsionalitas yang disediakan sistem sebagai unit yang saling bertukar pesan antar unit atau aktor. Simbol – simbol yang ada pada *use case* diagram terdiri dari (S. & Shalahuddin, 2016, hlm. 156–158):

Simbol	Deskripsi
<p><i>Use case</i></p> 	Fungsionalitas yang disediakan sistem sebagai unit – unit yang saling bertukar antar unit atau aktor,
<p><i>Actor</i></p> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri,
<p><i>Association</i></p> 	Komunikasi antara <i>actor</i> dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> yang memiliki interaksi dengan <i>actor</i>
<p><i>Extend</i></p> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu
<p><i>Generalization</i></p> 	Hubungan generalisasi dan spesialisasi antara dua buah <i>use case</i> di mana fungsi yang satu adalah fungsi yang lebih umum dari lainnya
<p><i>Include</i></p> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> utama untuk menjalankan fungsinya

(Sumber: (S. & Shalahuddin, 2016, hlm. 156–158))

c. *Activity Diagram*


Activity diagram menggambarkan *workflow* dari sebuah sistem atau proses bisnis yang ada pada perangkat lunak (S. & Shalahuddin, 2016, hlm. 161). *Activity diagram* digunakan untuk mendefinisikan rancangan urutan aktivitas proses bisnis. Simbol – simbol yang ada pada *activity diagram* di antaranya (S. & Shalahuddin, 2016, hlm. 162–163):


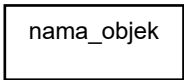

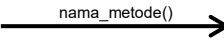
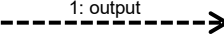
Simbol	Deskripsi
<i>Initial state</i> 	Status awal aktivitas sistem yang mengawali urutan aktivitas proses bisnis
<i>Activity</i> 	Aktivitas yang dilakukan sistem, aktivitas diawali dengan kata kerja
<i>Decision</i> 	Asosiasi percabangan apabila ada pilihan lebih dari satu aktivitas
<i>Join</i> 	Asosiasi penggabungan di mana lebih dari satu aktivitas digabungkan menjadi satu
<i>Final State</i> 	Status akhir yang dilakukan sistem sebagai akhir dari urutan aktivitas proses bisnis
<i>Swimlane</i> 	Simbol yang memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

(Sumber: (S. & Shalahuddin, 2016, hlm. 162–163))

d. *Sequence Diagram*

Sequence diagram menggambarkan *behaviour* objek pada *use case* dengan mendeskripsikan *lifecycle* objek dan *message* yang dikirimkan dan diterima antar objek (S. & Shalahuddin, 2016, hlm. 165). *Sequence diagram* menggunakan skenario yang ada pada *use case diagram* di mana objek yang terlibat dalam sebuah *use case* beserta metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Simbol – simbol yang ada pada *sequence diagram* terdiri dari (S. & Shalahuddin, 2016, hlm. 165–167):


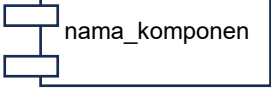
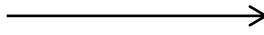
Simbol	Deskripsi
<i>Actor</i> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi

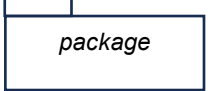

Simbol	Deskripsi
<p><i>Lifeline</i></p> 	Menyatakan garis hidup suatu objek
<p><i>Object</i></p> 	Menyatakan objek yang berinteraksi pesan
<p><i>Activation</i></p> 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan <i>activation</i> ini adalah tahapan yang dilakukan di dalamnya
<p><i>Call Message</i></p> 	Menyatakan suatu objek memanggil metode yang ada pada objek lain atau dirinya sendiri
<p><i>Return Message</i></p> 	Menyatakan bahwa suatu objek telah menjalankan suatu metode yang menghasilkan <i>output</i> ke objek tertentu

(Sumber: (S. & Shalahuddin, 2016, hlm. 165–167))

e. *Component Diagram*

Component diagram dibuat untuk menunjukkan organisasi di antara kumpulan komponen dalam satu sistem yang berfokus pada komponen sistem yang dibutuhkan (S. & Shalahuddin, 2016, hlm. 148). Komponen dasar yang terdapat dalam suatu sistem terdiri dari: *user interface*, *business processing*, data, dan *security*. Simbol – simbol yang terdapat pada *component diagram* di antaranya (S. & Shalahuddin, 2016, hlm. 149):

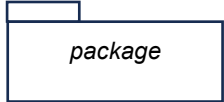
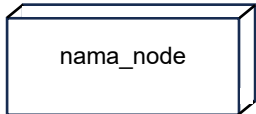


Simbol	Deskripsi
<p><i>Link</i></p> 	Relasi antar komponen
<p><i>Component</i></p> 	Komponen sistem
<p><i>Dependency</i></p> 	Mengacu pada kebergantungan suatu komponen pada komponen yang lain

Simbol	Deskripsi
<p><i>Package</i></p> 	Komponen yang membungkus dari satu atau lebih komponen
<p><i>Interface</i></p> 	Penghubung antara kelompok kelas dalam komponen berkomunikasi dengan sistem komponen yang lain

(Sumber: (S. & Shalahuddin, 2016, hlm. 149–150))

f. *Deployment Diagram*

Deployment diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi (S. and Shalahuddin, 2016, p. 154). Diagram digunakan untuk menggambarkan interaksi antara *hardware* dan *software* pada sistem. Simbol – simbol yang terdapat pada *deployment diagram* di antaranya (S. and Shalahuddin, 2016, p. 154):

Simbol	Deskripsi
<p><i>Package</i></p> 	Komponen yang membungkus dari satu atau lebih komponen
<p><i>Node</i></p> 	Komponen yang mengacu pada <i>software</i> , <i>hardware</i> , komponen yang disertakan harus sesuai dengan komponen yang terdefinisi pada diagram komponen
<p><i>Dependency</i></p> 	Mengacu pada kebergantungan suatu komponen pada komponen yang lain
<p><i>Link</i></p> 	Relasi antar komponen

(Sumber:(S. and Shalahuddin, 2016, pp. 154–155))

4. Penerapan Algoritma *Winnowing*

Algoritma adalah urutan langkah – langkah untuk memecahkan masalah yang diberikan. Algoritma *Winnowing* adalah salah satu algoritma untuk melakukan *document fingerprinting* di mana *fingerprint* tersebut didapat dari dokumen teks yang diubah setiap karakter hurufnya menjadi susunan kode atau dengan istilah *hashing* (Schleimer, Wilkerson and Aiken, 2003, p. 78).

Algoritma ini ditujukan untuk mendeteksi adanya kesamaan pada teks (Schleimer, Wilkerson and Aiken, 2003, p. 78). Pada setiap *window*, nilai *hash* yang dipilih adalah yang minimum, tetapi jika ada lebih dari satu nilai *hash* dengan nilai minimum, nilai yang dipilih berada posisinya paling kanan, kemudian simpan nilai *hash* yang dipilih untuk dijadikan sebagai *document fingerprinting*. (Schleimer, Wilkerson and Aiken, 2003, p. 77) menyatakan bahwa ada tiga hal yang perlu diperhatikan, yakni:

- (a) *whitespace insensitivity*, proses pencocokan teks tidak terpengaruh oleh beberapa atribut seperti spasi, huruf kapital, tanda baca dan sebagainya;
- (b) *noise suppression*, menyaring kata – kata yang tidak relevan atau tidak diinginkan seperti kata atau frasa umum yang seringkali muncul di dalam dokumen;
- (c) *position independence*, pencocokan teks tidak terpengaruh oleh adanya penambahan teks atau informasi baru, perubahan posisi atau urutan dari elemen – elemen suatu dokumen, hingga penghapusan informasi.

5. Penerapan Algoritma *Cosine Similarity*

Cosine similarity mengukur kesamaan antara dua dokumen yang mana dua dokumen tersebut masing – masing didefinisikan sebagai A_1 dan B_1 (Manning, Raghavan and Schütze, 2009, pp. 120–121). Dokumen dapat direpresentasikan sebagai vektor dalam ruang vektor (Manning, et al., 2009, pp. 120 - 121). Tingkat kesamaan antara dua dokumen diukur berdasarkan besar sudut yang menghubungkan dua dokumen tersebut, semakin kecil sudutnya, semakin besar tingkat kesamaan dua dokumen tersebut (Manning, Raghavan and Schütze, 2009, pp. 120–121). Rumus metode *cosine similarity* sebagaimana yang ditunjukkan pada Gambar 2.3 adalah sebagai berikut (Manning, Raghavan and Schütze, 2009, p. 121):

$$\text{sim}(d_1, d_2) = \frac{\vec{V}(d_1) \cdot \vec{V}(d_2)}{|\vec{V}(d_1)| |\vec{V}(d_2)|}$$

Gambar 2.3 Rumus *Cosine Similarity*

(Sumber: Manning, et al., 2009, p. 121)

Keterangan:

θ adalah sudut antara dua vektor,

$A \cdot B$ adalah dot product pada vektor A dan B dan dikalkulasikan sebagai $A \cdot$

$$B = A^T B = \sum_1^n A_i B_i = A_1 B_1 + A_2 B_2 + A_3 B_3 + \dots + A_n B_n,$$

$\|A\|$ dan $\|B\|$ adalah skala pada vektor yang dikalkulasikan sebagai $\|A\| =$

$$\sqrt{A_1^2 + A_2^2 + \dots + A_n^2} \text{ dan begitupun dengan } \|B\|.$$

6. Pengertian Esai

(Widiyanto, 2018, hlm. 124) menyatakan tes jenis uraian atau esai adalah jenis tes yang menuntut peserta didik untuk mengorganisasikan, mengekspresikan, serta menyatakan jawabannya menurut kata – kata sendiri. Tes esai dapat dibagi menjadi dua bentuk berdasarkan keluasan materi yang ditanyakan, yakni uraian terbatas dan uraian bebas (Widiyanto, 2018, hlm. 125). Soal uraian objektif atau terbatas mengukur kemampuan peserta didik dalam menguraikan konsep tertentu sesuai dengan mata pelajaran sehingga penskoran dilakukan secara objektif. Sedangkan soal uraian subjektif atau bebas mengukur kemampuan peserta didik dalam menguraikan pendapat pada suatu konsep tertentu sesuai dengan mata pelajaran sehingga penskoran dilakukan secara subjektif (Tim Pusat Penilaian Pendidikan, 2019, hlm. 44).

7. Pengertian *Automated Essay Scoring*

Automated Essay Scoring didefinisikan sebagai proses pemberian skor pada esai menggunakan program komputer dengan menganalisis isi dari esai (Smolentzov, 2012, hlm. 2). Skor nilai yang dihasilkan dari AES diukur tingkat kemiripan dan konsistensinya dengan nilai yang dihasilkan oleh penilaian manusia (Smolentzov, 2012, hlm. 4). *Automated Essay Scoring* menggunakan metode *feature construction* dengan cara mendefinisikan jumlah kemunculan semua kata dan karakter alfanumerik dalam teks tanpa adanya tanda baca dan *feature selection* dengan cara menyeleksi fitur berupa kata yang memiliki tingkat peringkat terendah dalam menganalisis isi teks pada esai (Smolentzov, 2012, hlm. 19 – 21).

8. Pengertian HTTP

HTTP (*Hypertext Transfer Protocol*) adalah protokol yang digunakan untuk dapat mentransfer data di dalam web dan merupakan sekumpulan data secara cepat, praktis, dan reliabel dari semua situs web yang ada di dunia ke *website browser* pada komputer pengguna (Gourley and Totty, 2002, p. 3). Aplikasi HTTP perlu menangani beberapa variasi protokol HTTP agar dapat berjalan dengan baik. Beberapa versi yang digunakan, di antaranya (Gourley and Totty, 2002, p. 17):

- (a) HTTP/0.9, versi ini mendukung metode GET, tetapi tidak mendukung pengetikan MIME untuk konten multimedia, *HTTP header*, atau nomor versi;
- (b) HTTP/1.0, versi protokol ini menambahkan nomor versi, *HTTP header*, metode tambahan, dan penanganan objek multimedia;
- (c) HTTP/1.0+, versi ini menawarkan fitur seperti *virtual hosting support*, *proxy connection support*, dan *long-lasting “keepalive” connections*;

- (d) HTTP/1.1, HTTP/1.1 berfokus pada perbaikan kekurangan arsitektur desain HTTP, mespesifikasi semantik, pengenalan optimalisasi performa yang signifikan, dan penghapusan fitur yang tidak ada;
- (e) HTTP-NG, penerus dari HTTP/1.1 yang berfokus pada optimisasi performa yang signifikan dan *framework* yang lebih kuat untuk eksekusi *server logic*.

B. Tinjauan Pustaka

Beberapa penelitian rujukan yang digunakan pada penelitian ini, di antaranya:

- (a) (Purwaningrum dkk., 2023) melakukan penelitian berjudul “Pengaruh Synonym Recognition dalam Deteksi Kemiripan Teks Menggunakan Winnowing dan Cosine Similarity”; penelitian ini mengukur kemiripan teks pada tugas akhir dengan menggunakan algoritma *Winnowing* dan *Cosine Similarity*; penelitian ini menambahkan metode *synonym recognition* untuk mencari adanya kata yang memiliki sinonim untuk meningkatkan akurasi; hasil yang didapatkan adalah nilai akurasi sebesar 81,05% dan nilai rata – rata error (RMSE) adalah sebesar 2,01% di mana hasilnya merupakan pengaruh dari metode *synonym recognition*;
- (b) (Nugraha dkk., 2022) melakukan penelitian berjudul “Sistem Koreksi Jawaban Uraian Singkat Otomatis Menggunakan Metode Winnowing”; penelitian ini menerapkan algoritma *Winnowing* untuk mengoreksi jawaban uraian singkat; penelitian ini menambahkan metode *query expansion* untuk meningkatkan akurasi hasil penilaian dengan mengecek kemiripan kata berdasarkan sinonim; hasil yang didapatkan dari penelitian ini adalah nilai akurasi rata – rata sebesar 14,88% dari hasil selisih antara hasil nilai dari pakar dengan sistem;
- (c) (Rihi dkk., 2022) melakukan penelitian berjudul “Algoritma Winnowing untuk Mendeteksi Kesamaan Judul Skripsi Teknik Informatika”; penelitian ini mengukur kemiripan judul skripsi di prodi Teknik Informatika untuk membantu pihak prodi mengambil keputusan dalam menerima judul skripsi mahasiswa dengan menggunakan algoritma Winnowing; hasil yang didapatkan pada penelitian ini adalah nilai kemiripan sebesar 34,78% pada pengujian dua buah judul skripsi;
- (d) (Ismail dkk., 2022) melakukan penelitian berjudul “PENILAIAN JAWABAN ESSAY OTOMATIS MENGGUNAKAN ALGORITMA WINNOWING PADA APLIKASI E-LEARNING SMA HANG TUAH 4 SURABAYA”; penelitian ini menggunakan penerapan Algoritma Winnowing untuk melakukan penilaian jawaban ujian essay secara otomatis dengan cara membandingkan jawaban siswa dengan kunci jawaban guru; hasil yang didapatkan dari penelitian ini adalah hasil nilai similaritas 70,83% dari pengujian; dari hasil perbandingan algoritma yang digunakan dengan Algoritma Rabin-Karp, didapatkan selisih nilai sebesar 29,05

dari hasil 5 jawaban siswa yang berbeda, sehingga rata-rata selisih yang dihasilkan sebesar 5,81;

- (e) (Kurniawati & Pradnya, 2020) melakukan penelitian berjudul “Implementasi Algoritma Winnowing Pada Sistem Penilaian Otomatis Jawaban Esai Pada Ujian Online Berbasis Web”; penelitian ini melakukan penilaian jawaban teks bahasa Indonesia yang didapatkan dari data ujian semester di sekolah menengah kejuruan; penelitian ini menguji hasil nilai dari aplikasi dengan hasil nilai guru; hasil yang didapatkan dari penelitian ini adalah nilai kemiripan sebesar 69,23% dan rata – rata nilai selisih antara nilai guru dan aplikasi sebesar 5,683;
- (f) (Rahim, 2022) melakukan penelitian berjudul “EVALUASI ESAI OTOMATIS DENGAN ALGORITMA NAZIEF & ADRIANI DAN WINNOWING”; penilaian esai dilakukan dengan mengolah teks yakni dengan mengembalikan suatu kata ke bentuk *root word* menggunakan algoritma Nazief & Adriani dan kemudian diubah menjadi nilai *hash* menggunakan algoritma *Winnowing*; data yang digunakan pada penelitian ini terdiri dari: soal dan kunci jawaban mata kuliah Pengantar Ilmu Ekonomi sebanyak 7 butir dan jawaban mahasiswa sebanyak 30 butir proses penskoran dilakukan menggunakan metode *Jaccard’s Similarity Coefficient* dengan mencari jumlah *fingerprint* yang sama; hasil yang didapatkan berupa nilai rata – rata kesamaan 81,20% dalam waktu 1,62 detik;
- (g) (Nasien & Yansen, 2022) melakukan penelitian berjudul “*Implementation of the Winnowing Algorithm in Detecting Plagiarism in Title and Abstract of Student’s Final Project*”; penelitian ini menerapkan algoritma *Winnowing* untuk mendeteksi adanya plagiarisme tugas akhir di bagian judul dan abstrak; hasil yang didapatkan dari penelitian ini adalah nilai persentase kemiripan sebesar 50% dari pengujian dua buah judul tugas akhir;
- (h) (Amalia dkk., 2022) melakukan penelitian berjudul “*Automatic essay assessment in e-learning using winnowing algorithm*”; penilaian esai dilakukan menggunakan metode *hashing* atau mengubah teks menjadi susunan kode angka yang kemudian dijadikan sebagai *fingerprint*; data yang digunakan pada penelitian ini berupa 20 data jawaban mahasiswa dan 10 kunci jawaban dari 10 pertanyaan; pemberian skor esai pada penelitian ini menggunakan *Jaccard Coefficient* untuk membandingkan hasil jawaban mahasiswa dengan kunci jawaban; hasil yang didapatkan pada penelitian ini adalah nilai similaritas sebesar 83,67% dan nilai selisih antara hasil nilai prediksi dengan nilai sebenarnya sebesar 5,2%;
- (i) (Mentari dkk., 2022) melakukan penelitian berjudul “*Cross-Language Text Document Plagiarism Detection System Using Winnowing Method*”; penelitian ini menerapkan algoritma *Winnowing* untuk mendeteksi adanya plagiarisme jenis

cross-language di mana pekerjaan seseorang dengan bahasa yang berbeda kemudian disalin dan diubah ke bahasa lain; data yang digunakan pada penelitian ini berasal dari beberapa jurnal *online*; hasil yang didapatkan dari penelitian ini adalah nilai akurasi sebesar 92,9%;

- (j) (Astica dkk., 2023) melakukan penelitian berjudul “Analysis of the Similiarity Level of Source Code in the Kotlin Programming Language using Winnowing Algorithm”; pada penelitian ini, peneliti menggunakan algoritma *Winnowing*, *K-gram*, dan *Jaccard Similarity* untuk mengukur tingkat plagiarisme pada kode program bahasa pemrograman Kotlin; dataset yang digunakan diambil dari *website* Github untuk diambil bahasa pemrograman Kotlin; data kemudian diproses menggunakan *text preprocessing* menggunakan *stemming* dan *tokenizing*; hasil yang didapatkan dari pengujian ini adalah tingkat *similarity* maksimum adalah 100% berdasarkan pemilihan *window* dan isi skrip program.

Dari sepuluh penelitian rujukan di atas, berikut adalah ringkasan mengenai kontribusi atau kelemahan sebagaimana dijelaskan pada tabel 2.1.

Tabel 2.1 Tinjauan Pustaka

No	Penelitian/Tahun	Judul	Sumber Jurnal	Kontribusi/Kelemahan
1.	Santi Purwaningrum, Agus Susanto, Ari Kristiningsih	Pengaruh Synonym Recognition dalam Deteksi Kemiripan Teks Menggunakan Winnowing dan Cosine Similarity	Jurnal Nasional Teknik Elektro dan Teknologi Informasi Vol. 12, No. 3, Agustus 2023, pp. 219 – 226: doi: https://doi.org/10.22146/jnteti.v12i3.6375	Penelitian ini menggunakan metode <i>synonym recognition</i> pada algoritma Winnowing dan Cosine Similarity untuk meningkatkan hasil kemiripan teks dengan pendekatan sinonim pada tugas akhir.
2.	Bagus Nugraha, Ardi Sanjaya, Danar Putra Pamungkas / 2022	Sistem Koreksi Jawaban Uraian Singkat Otomatis Menggunakan Metode Winnowing	Prosiding SEMNAS INOTEK (Seminar Nasional Inovasi Teknologi) Volume 6, No. 2, Juli 2022, pp. 082–087. doi:10.29407/inotek.v6i2.2564 URL: Sistem Koreksi Jawaban Uraian Singkat Otomatis Menggunakan Metode Winnowing Prosiding SEMNAS INOTEK (Seminar Nasional Inovasi Teknologi) (unpkediri.ac.id)	Penelitian ini menggunakan algoritma Winnowing untuk mengoreksi uraian singkat dan query expansion untuk meningkatkan akurasi penilaian.

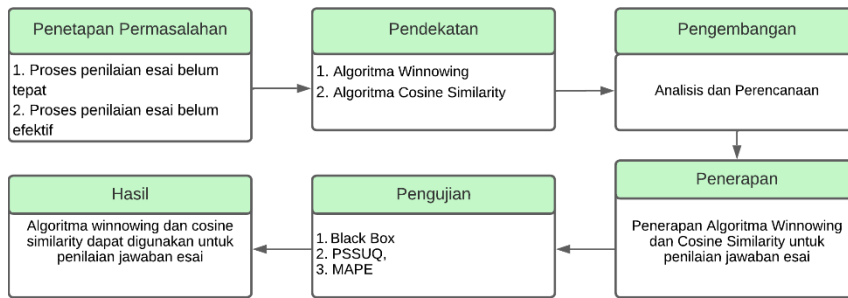
No	Penelitian/Tahun	Judul	Sumber Jurnal	Kontribusi/Kelemahan
3.	Mega Peka Rih, Arini Aha Pekuwali, Desy Asnath Sitaniapessy / 2022	Algoritma Winnowing untuk Mendeteksi Kesamaan Judul Skripsi Teknik Informatika	MALCOM Vol. 2, No. 2, Oktober 2022, pp. 42 – 52 DOI: https://doi.org/10.57152/malcom.v2i2.436	Penelitian ini menerapkan algoritma Winnowing untuk membantu pihak prodi mengambil Keputusan dalam menerima judul skripsi mahasiswa dengan cepat.
4.	Yunus Oktavianto Ismail, Kartini, Firza Prima Aditiawan / 2022	PENILAIAN JAWABAN ESSAY OTOMATIS MENGGUNAKAN ALGORITMA WINNOWING PADA APLIKASI E-LEARNING SMA HANG TUAH 4 SURABAYA	Seminar Nasional Teknologi dan Sistem Informasi (SITASI), Vol. 2, No. 1, Juli 2022, pp. 1 – 10. DOI: https://doi.org/10.33005/sitasi.v2i1.249	Penelitian ini menerapkan algoritma Winnowing dalam melakukan penilaian esai secara otomatis untuk sekolah menengah atas.
5.	Febrilia Endah Kurniawati, Windha Mega Pradnya / 2020	Implementasi Algoritma Winnowing Pada Sistem Penilaian Otomatis Jawaban Esai Pada Ujian Online Berbasis Web	Jurnal Teknik Komputer AMIK BSI, Vol. VI, No. 2, Juli 2020, pp. 169 – 175 DOI:10.31294/jtk.v6i2.7838 URL: https://ejournal.bsi.ac.id/ejournal/index.php/jtk/article/view/7838	Penelitian ini menerapkan algoritma Winnowing dalam melakukan penilaian esai pada soal ujian semester yang dilaksanakan oleh sekolah menengah kejuruan.
6.	Arham Rahim / 2022	EVALUASI ESAI OTOMATIS DENGAN ALGORITMA NAZIEF & ADRIANI DAN WINNOWING	TOLIS ILMIAH: JURNAL PENELITIAN Vol. 4, No.1, Mei 2022, pp.51 – 63 DOI: http://dx.doi.org/10.56630/jti.v4i1.212	Kontribusi dari penelitain ini adalah algoritma <i>Winnowing</i> dan Nazief & Adriani dapat digunakan untuk melakukan penilaian esai.
7.	Dewi Nasien, Eka Putra Yansen / 2022	<i>Implementation of the Winnowing Algorithm in Detecting Plagiarism in Title and Abstract of</i>	IT Journal Research and Development (ITJRD), Vol. 7, No. 1, Juli 2022, pp.35 – 47 DOI: https://doi.org/10.25299/itjrd.2022.7897	Penelitian ini menerapkan algoritma winnowing untuk mendeteksi adanya plagiarsime tugas akhir di bagian judul dan abstrak.

No	Penelitian/Tahun	Judul	Sumber Jurnal	Kontribusi/Kelemahan
		<i>Student's Final Project</i>		
8.	Eka Larasati Amalia, Vivin Ayu Lestari, Vivi Nur Wijyaningrum, Ali Ar Ridla / 2023	Automatic Essay Assessment in E-learning Using Winnowing Algorithm	<i>Indonesian Journal of Electrical Engineering and Computer Science</i> , Vol. 29, No. 1, January 2023, pp. 572 – 582 DOI: http://doi.org/10.11591/ijeecs.v29.i1.pp572-582	Kontribusi dari penelitian ini adalah algoritma <i>Winnowing</i> dapat digunakan untuk melakukan penilaian esai.
9	Mustika Mentari, Imam Fahrur Rozi, Maria Puji Rahayu / 2022	<i>Cross-Language Text Document Plagiarism Detection System Using Winnowing Method</i>	<i>Journal of Applied Intelligent System</i> , Vol. 7, No. 1, April 2022, pp.44 – 57 DOI: https://doi.org/10.33633/jais.v7i1.5950	Kontribusi dari penelitian ini adalah algoritma <i>Winnowing</i> dapat digunakan untuk mendeteksi tingkat plagiarisme pada teks jenis <i>cross-language</i> .
10	Yustikamasy Astica, Ema Utami, Anggit Dwi Hartanto / 2023	Analysis of the Similarity Level of Source Code in the Kotlin Programming Language using Winnowing Algorithm	International Journal of Artificial Intelligence Research ISSN 2579-7298 Vol. 7, No. 1, Juni 2023, pp. 39 – 55 DOI: https://doi.org/10.29099/ijair.v7i1.902	Kontribusi dari penelitian ini adalah algoritma <i>winnowing</i> digunakan untuk mendeteksikesamaan pada bahasa pemrograman Kotlin.

Dari sepuluh jurnal penelitian terdahulu, setiap jurnal membahas permasalahan yang relatif serupa dengan penelitian yang akan dilakukan yakni mengukur tingkat kemiripan teks pada dokumen dengan menggunakan algoritma *Winnowing* dan *Cosine Similarity*. Persamaan dari penelitian sebelumnya dengan penelitian ini adalah terletak pada penerapan algoritmanya, sedang perbedaan dari penelitian ini dengan penelitian terdahulu adalah penelitian ini membahas tentang penilaian esai dan variabel yang digunakan pada penelitian ini sebanyak 4 variabel, di mana pada penelitian terdahulu variabel digunakan sebanyak 3.

C. Kerangka Pemikiran

Dari hasil penjabaran pada tinjauan pustaka dan landasan teori sebagai acuan dalam melaksanakan penelitian ini, disusunlah kerangka pemikiran sebagai langkah strategis guna memecahkan masalah sebagaimana terdapat pada gambar 2.5



Gambar 2.4 Kerangka Pemikiran

Proses identifikasi masalah dilakukan dengan melalui observasi, wawancara, dan menyebarkan kuesioner dengan objek penelitian, berdasarkan hasil pengamatan yang dilakukan didapatkan komponen permasalahan yakni penilaian esai cenderung kurang tepat karena adanya kesulitan dan kesalahan dalam pemberian nilai, serta kurang efektif karena memakan waktu yang relatif tidak singkat. Untuk dapat memecahkan permasalahan yang terjadi, muncul ide yang didapatkan dari hasil peninjauan penelitian – penelitian terdahulu yang membahas mengenai permasalahan yang identik dengan permasalahan ini yakni dengan menggunakan algoritma sebagai bentuk pendekatan komputasi. Algoritma yang akan diterapkan untuk memecahkan masalah ini adalah algoritma *Winnowing* dan *Cosine Similarity*, dilanjutkan dengan pengembangan penilaian jawaban esai, mendapatkan tingkat ketepatan nilai dan kemudian masuk ke tahap evaluasi. Pada tahap evaluasi, aplikasi menguji tingkat ketepatan agar hasilnya sesuai dengan yang diharapkan. Hasil yang didapatkan adalah nilai perolehan pada jawaban esai.

D. Hipotesis Penelitian

Penelitian ini mengusung algoritma *Winnowing* untuk melakukan *fingerprinting* teks dan *Cosine Similarity* untuk mengukur tingkat kemiripan teks antara dua dokumen. Berdasarkan penelitian yang telah dilakukan oleh (Kurniawati & Pradnya, 2020) yang berjudul “Implementasi Algoritma Winnowing Pada Sistem Penilaian Otomatis Jawaban Esai Pada Ujian Online Berbasis Web”, algoritma ini dapat digunakan untuk melakukan penilaian esai. Algoritma Winnowing dan Cosine Similarity telah digunakan pada penelitian – penelitian sebelumnya untuk mengukur tingkat kemiripan teks antar dokumen. Dari pemahaman di atas, maka hipotesis dari penelitian ini adalah penerapan algoritma *Winnowing* dan *Cosine Similarity* diduga dapat melakukan penilaian jawaban esai dengan tepat dan efektif.