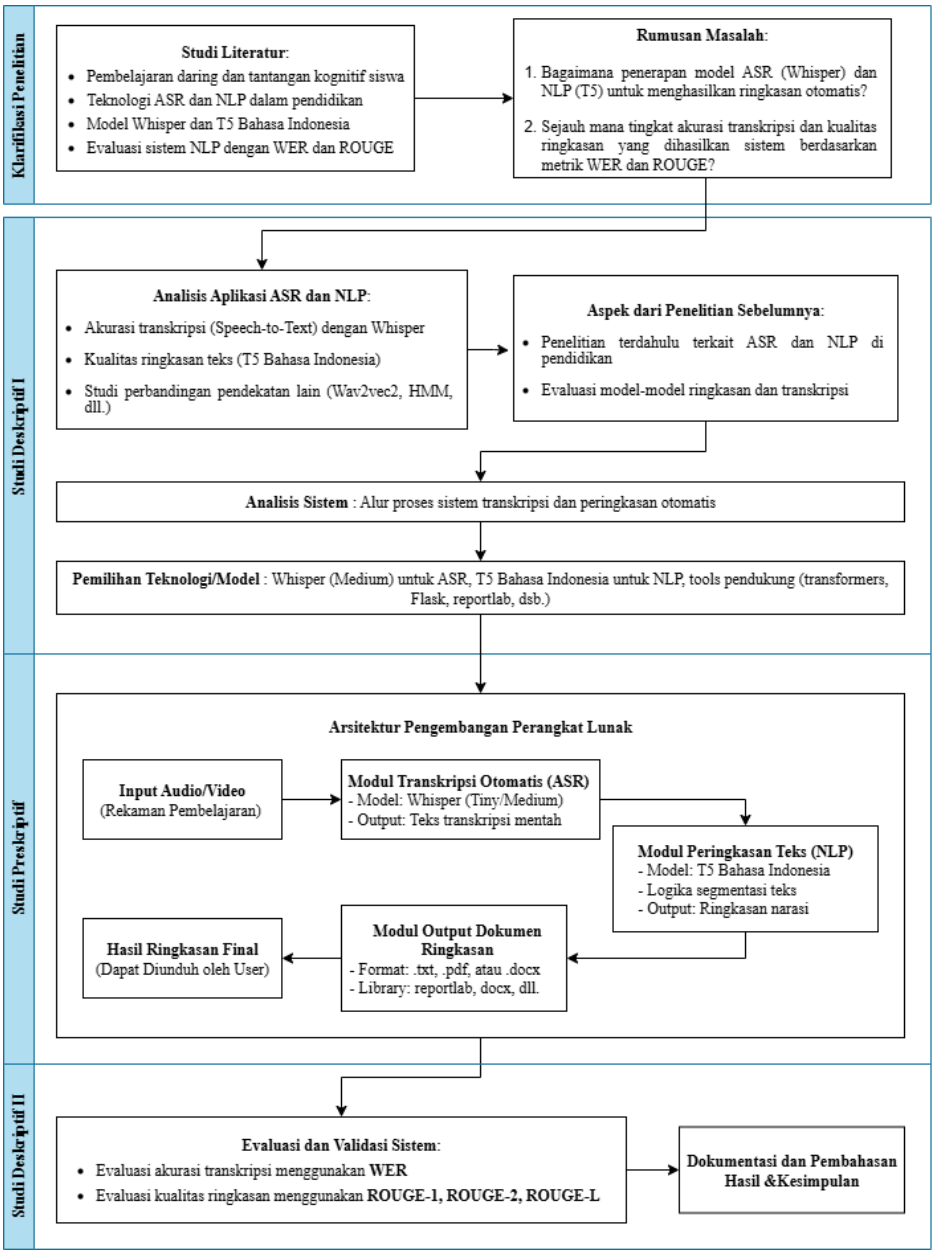
BAB III
METODOLOGI PENELITIAN

3.1 Desain Penelitian

Penelitian ini menggunakan pendekatan gabungan yang mengintegrasikan *Design Research Methodology* (DRM) dengan proses pengembangan perangkat lunak secara sistematis.



Gambar 3. 1 *Design Research Methodology* (DRM)

Proses penelitian mengikuti tahapan-tahapan yang meliputi klarifikasi masalah, studi deskriptif untuk analisis literatur dan sistem, studi preskriptif untuk perancangan dan pengembangan perangkat lunak, serta studi deskriptif lanjutan untuk evaluasi dan validasi sistem, sebagaimana diilustrasikan dalam diagram alur penelitian.

Design Research Methodology (DRM) berfungsi sebagai kerangka kerja utama yang memandu seluruh proses penelitian. DRM dirancang untuk membantu peneliti dalam mengidentifikasi area penelitian, memilih metode yang sesuai, dan mengintegrasikan berbagai jenis penelitian secara koheren (Ebneyamini, 2022). Kerangka kerja ini menekankan pentingnya studi deskriptif untuk memahami desain dan menginformasikan pengembangan dukungan. Tujuan utama DRM adalah memberikan arah bagi penelitian desain dan mengidentifikasi area yang paling relevan secara teoritis dan praktis untuk ditangani (Ebneyamini, 2022).

DRM terdiri dari empat tahapan utama yang saling terkait, yang masing-masing memiliki tujuan dan hasil spesifik:

1. *Research Clarification* (Klarifikasi Penelitian)

Tahap ini bertujuan untuk mengidentifikasi tujuan yang ingin dicapai oleh penelitian dan fokus proyek penelitian. Ini melibatkan definisi kriteria yang jelas yang diharapkan dapat dipenuhi oleh penelitian (Tawfik et al., 2020). Fase awal ini bertujuan untuk memahami permasalahan dan menentukan fokus penelitian. Pada tahap Studi Literatur, peneliti melakukan penelusuran berbagai literatur yang berkaitan dengan tantangan pembelajaran daring, serta pemanfaatan teknologi *Automatic Speech Recognition* (ASR) dan *Natural Language Processing* (NLP) dalam konteks pendidikan. Selain itu, dilakukan juga kajian terhadap metode evaluasi sistem, khususnya penggunaan metrik *Word Error Rate* (WER) untuk mengukur akurasi transkripsi, dan ROUGE untuk menilai kualitas ringkasan teks.

Berdasarkan hasil studi literatur tersebut, dirumuskan dua fokus utama dalam penelitian ini. Pertama, bagaimana penerapan model Whisper (untuk ASR) dan T5 Bahasa Indonesia (untuk NLP) dapat digunakan untuk menghasilkan rangkuman otomatis dari rekaman audio pembelajaran atau materi yang disampaikan secara lisan melalui media digital. Kedua, sejauh mana tingkat akurasi transkripsi dan

kualitas ringkasan yang dihasilkan sistem berdasarkan metrik evaluasi WER dan ROUGE.

2. *Descriptive Study I* (Studi Deskriptif I)

Peran tahap ini adalah mengidentifikasi faktor-faktor yang memengaruhi kriteria terukur yang telah dirumuskan dan bagaimana faktor-faktor tersebut memengaruhinya. Studi ini memberikan dasar untuk pengembangan dukungan dan menghasilkan model referensi atau teori (Patel & Patel, 2019; Snyder, 2019). Fase ini menganalisis teknologi dan solusi yang ada.

- Analisis Aplikasi ASR dan NLP : Mengevaluasi kemampuan transkripsi Whisper, kualitas ringkasan oleh T5 Bahasa Indonesia, serta membandingkan dengan pendekatan lain (misalnya Wav2vec2, HMM).
- Aspek dari Penelitian Sebelumnya : Mengkaji hasil penelitian terdahulu yang relevan untuk mendukung kerangka pemilihan model.
- Analisis Sistem : Mendeskripsikan alur kerja sistem: mulai dari input audio → transkripsi → ringkasan → output PDF.
- Pemilihan Teknologi/Model dimana menentukan teknologi yang akan digunakan:
  - Whisper (Medium) untuk ASR
  - T5 Bahasa Indonesia untuk NLP
  - Tools tambahan: reportlab, Hugging Face Transformers, dsb.

3. *Prescriptive Study* (Studi Preskriptif)

Berdasarkan Hasil dari tahap *Descriptive Study I* berupa identifikasi dan pemilihan model-model NLP yang relevan, kemudian menjadi landasan utama untuk memasuki tahap pengembangan (development). Model-model yang telah dianalisis dan dipilih tersebut selanjutnya diterapkan pada tahap *Prescriptive Study* untuk membangun sebuah purwarupa sistem *speech-to-text* yang dapat menyediakan ringkasan otomatis. Fase ini merupakan proses pengembangan perangkat lunak yang terdiri dari langkah-langkah utama sebagai berikut:

- Analisis – Menentukan kebutuhan sistem dan teknologi yang akan digunakan, seperti Whisper untuk ASR dan T5 Bahasa Indonesia untuk NLP.

- Desain – Merancang arsitektur perangkat lunak dan alur modul, meliputi modul transkripsi otomatis (ASR), modul peringkasan teks (NLP), dan modul output dokumen ringkasan.
- Implementasi (Coding) – Mengembangkan sistem sesuai desain dengan mengintegrasikan model Whisper dan T5 serta tools pendukung seperti Flask dan reportlab.
- Pengujian (Testing) – Melakukan uji sistem secara fungsional untuk memastikan akurasi transkripsi dan kualitas ringkasan sesuai standar evaluasi menggunakan metrik WER dan ROUGE.

Proses ini dilakukan secara sistematis dan terstruktur agar pengembangan perangkat lunak dapat berjalan efektif dan menghasilkan output yang dapat digunakan oleh pengguna. Purwarupa yang dihasilkan dari tahap *Prescriptive Study* selanjutnya dievaluasi secara mendalam pada tahap evaluasi (evaluation) yang diimplementasikan melalui *Descriptive Study II*. Tahap ini bertujuan untuk mengukur efektivitas dan fungsionalitas sistem yang telah dikembangkan melalui serangkaian pengujian. Hasil evaluasi ini juga penting untuk mengidentifikasi area-area yang memerlukan perbaikan dan penyempurnaan di masa mendatang, sehingga memastikan siklus iteratif dalam metodologi DRM dapat terlaksana.

4. *Descriptive Study II* (Studi Deskriptif II)

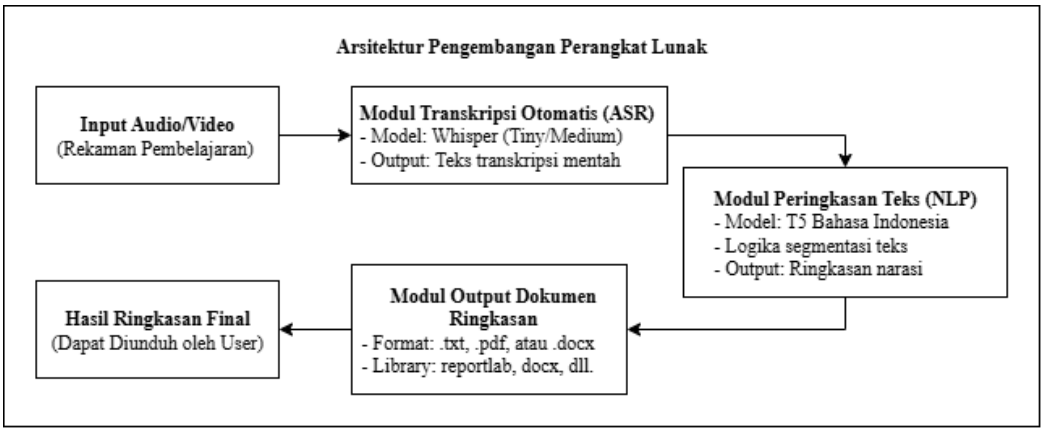
Tahap terakhir ini bertujuan untuk mengevaluasi dukungan yang telah dikembangkan. Evaluasi ini mengidentifikasi apakah dukungan tersebut dapat digunakan dalam situasi yang dimaksudkan dan apakah benar-benar mengatasi faktor-faktor yang seharusnya ditangani, serta kontribusinya terhadap keberhasilan (Patel & Patel, 2019; Snyder, 2019). Pada fase evaluasi dan validasi sistem, dilakukan pengujian terhadap kinerja sistem yang telah dikembangkan. Evaluasi difokuskan pada dua aspek utama, yaitu akurasi hasil transkripsi dan kualitas rangkuman otomatis. Untuk menilai akurasi transkripsi, digunakan metrik *Word Error Rate* (WER), yang mengukur seberapa banyak kesalahan kata dalam hasil transkripsi dibandingkan dengan transkrip referensi. Sementara itu, kualitas ringkasan dievaluasi menggunakan metrik ROUGE, yang mencakup ROUGE-1,

ROUGE-2, dan ROUGE-L guna menilai kesamaan n-gram dan struktur antara ringkasan sistem dan ringkasan referensi. Setelah proses evaluasi selesai, dilakukan dokumentasi dan pembahasan hasil guna menyusun temuan penelitian, menganalisis kinerja sistem berdasarkan metrik yang digunakan, serta menarik kesimpulan akhir yang merefleksikan efektivitas sistem dalam konteks pembelajaran daring.

Meskipun DRM memiliki tahapan yang jelas, prosesnya tidak sepenuhnya sekuensial; banyak iterasi akan terjadi, dan beberapa tahapan mungkin harus berjalan secara paralel. Sebagai contoh, pemilihan perangkat keras dan perangkat lunak serta ruang lingkup sistem demonstrator dalam Studi Preskriptif memerlukan evaluasi fitur sistem yang akan dievaluasi (Patel & Patel, 2019; Snyder, 2019).

3.1.1 Metode Pengembangan Perangkat Lunak

Pengembangan sistem ini mengikuti arsitektur terstruktur yang menggambarkan alur kerja otomatis untuk menghasilkan ringkasan dari rekaman pembelajaran audio/video. Setiap modul dan tahapan bekerja secara berurutan dan saling terintegrasi untuk memastikan proses yang efektif dan efisien. Model pengembangan perangkat lunak yang digunakan menerapkan siklus hidup yang logis dan berurutan, di mana setiap fase diselesaikan secara tuntas sebelum melanjutkan ke tahap berikutnya (Rapolu H. K., 2024). Pendekatan ini menjamin proses pengembangan berjalan secara sistematis dan terstruktur, dengan alur yang mengalir dari satu langkah ke langkah berikutnya secara berurutan (Maukar et al., 2023).



Gambar 3.2 Tahapan Metode Pengembangan Perangkat Lunak

Pengembangan sistem ini mengikuti tahapan yang logis dan berurutan guna memastikan proses yang terstruktur dan efektif dalam menghasilkan ringkasan dari rekaman pembelajaran audio/video. Tahapan umum dalam pengembangan perangkat lunak meliputi:

1. Pengumpulan dan Analisis Kebutuhan (*Requirements Gathering and Analysis)*: Pada tahap ini, dilakukan identifikasi kebutuhan sistem berdasarkan tujuan penelitian, yaitu membangun sistem rangkuman otomatis dari pembelajaran daring. Kebutuhan yang dikumpulkan meliputi spesifikasi perekaman audio (durasi per segmen, format file), pemilihan model ASR (Whisper Medium), model NLP (T5 Bahasa Indonesia), serta format output ringkasan. Semua kebutuhan ini didokumentasikan dalam dokumen spesifikasi sistem yang menjadi acuan fase selanjutnya (Ly et al., 2025; Sulaiman et al., 2022).
2. Desain Sistem (*System Design):* Berdasarkan kebutuhan yang telah dikumpulkan, dibuat rancangan arsitektur sistem yang terdiri dari tiga komponen utama: modul perekaman dan transkripsi audio menggunakan ASR Whisper, modul peringkasan teks menggunakan model T5 Bahasa Indonesia, serta modul output untuk menghasilkan ringkasan dalam format PDF/TXT. Rancangan ini mencakup diagram alur proses, desain antarmuka sederhana, pseudocode, serta struktur penyimpanan file hasil transkripsi dan ringkasan (Sulaiman et al., 2022).

3. Implementasi *(Coding):* Tahap ini merealisasikan desain ke dalam bentuk kode program menggunakan bahasa Python. Implementasi mencakup pembuatan skrip perekaman audio berbasis PyAudio, integrasi model Whisper untuk transkripsi lokal, integrasi model T5 untuk peringkasan teks, serta fungsi penyimpanan hasil dalam format dokumen. Setiap modul diimplementasikan secara bertahap sesuai urutan proses pada arsitektur sistem (Sulaiman et al., 2022).
4. Integrasi dan Pengujian *(Integration and Testing):* Setelah semua modul selesai diimplementasikan, dilakukan proses integrasi sehingga sistem dapat berjalan secara utuh: mulai dari perekaman audio → transkripsi → peringkasan → penyimpanan output. Pengujian dilakukan dengan metode Black Box untuk memastikan setiap fungsi bekerja sesuai spesifikasi, serta pengujian evaluasi performa menggunakan metrik **Word Error Rate (WER)** untuk hasil transkripsi dan **ROUGE** untuk kualitas ringkasan. Pengujian ini bertujuan memastikan bahwa sistem memenuhi kebutuhan yang telah ditentukan (Ly et al., 2025; Sulaiman et al., 2022).

Dengan pendekatan ini, proses pengembangan berjalan secara sistematis dan terstruktur, memastikan setiap tahap dan modul saling terintegrasi demi menghasilkan produk yang berkualitas dan sesuai dengan kebutuhan pengguna. Pendekatan ini dipilih karena sifatnya yang terstruktur, memungkinkan setiap fase diselesaikan sepenuhnya sebelum melanjutkan ke fase berikutnya secara logis dan teratur (Maukar et al., 2023). Keunggulan dari pendekatan ini meliputi kemudahan pemahaman, tahapan yang jelas dan terdefinisi, serta dokumentasi yang baik pada setiap tahap. Pendekatan ini sangat sesuai untuk proyek dengan persyaratan yang jelas dan stabil serta definisi produk yang minim ambiguitas (Maukar et al., 2023; Rapolu H.K., 2024).

3.1.2 Integrasi DRM dan Model Pengembangan Sistem

Dalam penelitian rekayasa perangkat lunak ini, integrasi *Design Research Methodology* (DRM) dengan pendekatan pengembangan sistem yang terstruktur dilakukan untuk menggabungkan kekuatan masing-masing metode. Pendekatan

pengembangan sistem yang terstruktur menyediakan kerangka kerja yang jelas dan berurutan untuk proses pengembangan perangkat lunak, sementara DRM memberikan landasan kuat untuk penelitian dan pengembangan dukungan yang berulang serta berbasis bukti.

Penggabungan kedua metodologi ini memungkinkan pendekatan yang lebih holistik dalam pengembangan sistem. Pendekatan terstruktur memastikan bahwa setiap langkah pengembangan perangkat lunak, mulai dari pengumpulan kebutuhan hingga pemeliharaan, dilakukan secara sistematis dan terdokumentasi dengan baik (Sulaiman et al., 2022). Sementara itu, DRM memandu aspek penelitian, memastikan bahwa pengembangan sistem didasarkan pada pemahaman mendalam mengenai permasalahan (melalui Studi Deskriptif I) dan bahwa solusi yang dikembangkan (melalui Studi Preskriptif) dievaluasi secara ketat (melalui Studi Deskriptif II) (Patel & Patel, 2019; Snyder, 2019).

Salah satu titik integrasi yang signifikan adalah pada tahap analisis kebutuhan. Pendekatan yang menggabungkan prinsip *User-Centered Design* (UCD) dengan kerangka kerja pengembangan sistem terstruktur pada tahap analisis kebutuhan telah diusulkan, menunjukkan bahwa model-model ini memerlukan pendetailan yang tepat untuk integrasi (Wulan et al., 2024). Meskipun UCD dan DRM adalah metodologi yang berbeda, prinsip pendetailan dan pemahaman mendalam tentang konteks masalah yang ditekankan dalam DRM pada tahap *Research Clarification* dan *Descriptive Study I* sangat selaras dengan pengumpulan dan analisis kebutuhan dalam pendekatan pengembangan sistem terstruktur. Hal ini memungkinkan peneliti untuk memahami secara komprehensif faktor-faktor yang memengaruhi keberhasilan sistem sebelum desain dan implementasi dilakukan secara berurutan.

Pendekatan ini menciptakan inovasi yang terstruktur. Dengan menggabungkan eksplorasi dan validasi berbasis penelitian dari DRM dengan proses pengembangan yang disiplin dan sistematis, proyek dapat mencapai tujuan inovatif sambil mempertahankan kontrol kualitas dan prediktabilitas. Hal ini memastikan bahwa sistem yang dikembangkan tidak hanya fungsional tetapi juga relevan dan efektif dalam mengatasi masalah yang dituju, dengan dukungan bukti empiris dari tahapan penelitian DRM.

3.2 Pengumpulan Data

Bagian ini menguraikan jenis data yang dikumpulkan, metode yang digunakan untuk pengumpulannya, dan pertimbangan etis yang diterapkan untuk memastikan integritas dan keamanan data.

3.2.1 Jenis Data

Data yang dikumpulkan dalam penelitian ini meliputi beberapa jenis untuk mendukung pengembangan dan evaluasi sistem *Automatic Speech Recognition* (ASR) dan *Natural Language Processing* (NLP):

1. Data Audio Rekaman Pembelajaran : Data ini merupakan rekaman suara dari sesi pembelajaran daring atau data audio dari konten pembelajaran berbasis suara. Rekaman ini menjadi input utama bagi sistem ASR untuk diubah menjadi transkrip teks (Zuazo et al., 2025).
2. Transkrip Hasil ASR (Whisper): Ini adalah teks yang dihasilkan oleh sistem ASR (Whisper) dari data audio. Data ini akan dievaluasi untuk akurasi transkripsi.
3. Ringkasan Teks Hasil NLP: Ini adalah ringkasan teks yang dihasilkan oleh model NLP dari transkrip ASR (Whisper). Ringkasan ini akan dievaluasi untuk kualitas dan relevansinya.
4. Data Evaluasi (WER, ROUGE): Data ini merupakan hasil perhitungan metrik evaluasi seperti *Word Error Rate* (WER) untuk ASR (Whisper) dan ROUGE (ROUGE-1, ROUGE-2, ROUGE-L) untuk NLP. Data ini akan digunakan untuk mengukur performa sistem secara kuantitatif.

3.2.2 Metode Pengumpulan Data

Proses pengumpulan data dilakukan melalui beberapa tahapan yang terstruktur:

1. Perekaman Audio dari Sesi Pembelajaran : Perekaman audio dilakukan dari sesi pembelajaran daring yang relevan dengan topik penelitian. Proses ini memerlukan penggunaan perangkat lunak atau perangkat keras perekam yang sesuai untuk menangkap audio dengan kualitas yang memadai untuk pemrosesan ASR (Whisper).
2. Pengumpulan Data untuk Pelatihan dan Pengujian Model NLP: Untuk model

NLP, data teks (baik transkrip ASR (Whisper) maupun teks sumber asli dari materi pembelajaran) dikumpulkan. Sebagian data ini akan digunakan untuk pelatihan model (jika diperlukan *fine-tuning* atau pelatihan dari awal), dan sebagian lainnya akan digunakan sebagai dataset pengujian untuk mengevaluasi kemampuan peringkasan teks. Ringkasan referensi yang dibuat manusia juga dikumpulkan atau dibuat untuk tujuan evaluasi model NLP (Lam et al., 2022).

Selain data utama, penelitian ini juga memerlukan **teks referensi** untuk proses evaluasi.

1. Teks transkripsi referensi dibuat secara manual dari audio pembelajaran daring, berfungsi sebagai acuan untuk menghitung tingkat kesalahan transkripsi (WER).
2. Ringkasan referensi merupakan ringkasan manual materi pembelajaran yang digunakan sebagai pembanding untuk mengukur kualitas rangkuman otomatis dengan metrik ROUGE.

3.2.3 Pertimbangan Etis

Pengumpulan data yang melibatkan rekaman audio dari sesi pembelajaran daring dan transkripsi data tersebut memerlukan pertimbangan etis yang cermat untuk melindungi partisipan penelitian. Perlindungan partisipan dari potensi bahaya adalah tujuan fundamental yang mendasari semua masalah etika lainnya (Antaki et al., 2025; Cychosz et al., 2020).

Beberapa pertimbangan etis utama yang diterapkan dalam penelitian ini meliputi:

1. Persetujuan Informed *(Informed Consent):* Ini adalah komponen terpenting dalam penelitian yang melibatkan subjek manusia (Cychosz et al., 2020). Partisipan penelitian diberikan pengungkapan penuh tentang semua informasi yang diperlukan untuk membuat keputusan yang tepat untuk berpartisipasi (Akbar & Basrowi, 2022). Proses ini mencakup pemahaman yang jelas tentang prosedur yang terlibat dalam pengumpulan data (misalnya, perekaman audio), bagaimana data akan digunakan, dan bagaimana data akan

dibagikan. Peneliti diwajibkan untuk memastikan bahwa partisipan memiliki kesadaran penuh tentang studi dan risiko yang terlibat (Cychosz et al., 2020). Dalam konteks perekaman jarak jauh, alternatif seperti proses persetujuan lisan dapat dipertimbangkan, di mana peneliti mengonfirmasi bahwa partisipan telah memiliki kesempatan untuk membaca *Letter of Information* dan kemudian memperoleh persetujuan lisan menggunakan skrip persetujuan lisan (Akbar & Basrowi, 2022).

2. Perlindungan dari Bahaya: Peneliti bertanggung jawab atas kesejahteraan partisipan penelitian. Dalam konteks penelitian ini, bahaya dapat timbul dari kerusakan reputasi, pelanggaran privasi, atau efek lain yang disebabkan oleh penanganan data yang salah. Langkah-langkah diambil untuk memastikan bahwa praktik pengumpulan data tidak menempatkan partisipan dalam bahaya langsung (Cychosz et al., 2020).
3. Anonimisasi Data: Untuk menjaga privasi dan kerahasiaan, data harus dianonimkan. Ini berarti detail pribadi seperti nama dan alamat harus dihapus atau dimodifikasi untuk memastikan identitas partisipan penelitian tidak terungkap dalam diseminasi penelitian. Penggunaan pseudonim adalah praktik umum dalam penelitian kualitatif untuk memfasilitasi konversi data mentah menjadi data anonim. Jika informasi pribadi penting untuk tindak lanjut, *codebook* yang disimpan terpisah dari data, yang mencocokkan pseudonim dengan nama asli, akan dilindungi dengan langkah-langkah keamanan tertinggi (Cychosz et al., 2020).
4. Penyimpanan Data Aman: Data yang dikumpulkan harus disimpan dengan aman untuk mencegah akses tidak sah atau penyalahgunaan. Ini mencakup langkah-langkah seperti enkripsi data, penyimpanan data di lokasi lokal yang aman (bukan hanya di server berbasis *cloud* tanpa perlindungan yang memadai), dan pembuangan data yang aman (misalnya, menghancurkan data daripada hanya membuang data cetak atau media elektronik) (Cychosz et al., 2020). Peneliti diharapkan untuk tidak membagikan data mentah di luar konteks penelitian, seperti media sosial atau forum publik lainnya, tanpa izin tertulis dari partisipan (Rodina & Kutuzov, 2020).

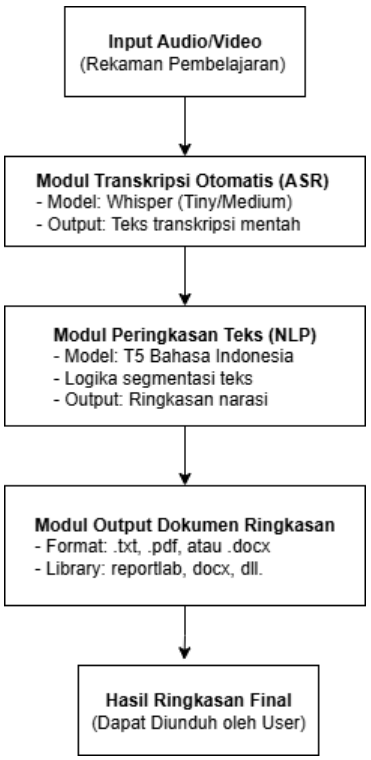
Penerapan pertimbangan etis ini menegaskan pentingnya integritas dan kepercayaan dalam penelitian yang melibatkan data manusia. Hal ini bukan hanya kewajiban moral tetapi juga fondasi untuk memastikan validitas dan penerimaan hasil penelitian, terutama ketika melibatkan teknologi AI yang memproses informasi pribadi dan sensitif.

3.3 Arsitektur dan Implementasi Sistem

Bagian ini menguraikan desain sistem secara keseluruhan, arsitektur model ASR dan NLP yang digunakan, serta lingkungan pengembangan perangkat keras dan perangkat lunak.

3.3.1 Arsitektur Sistem Keseluruhan

Sistem yang dikembangkan dalam penelitian ini dirancang sebagai sebuah solusi terintegrasi yang menggabungkan kemampuan *Automatic Speech Recognition* (ASR) dengan *Natural Language Processing* (NLP) untuk meringkas konten pembelajaran daring.



Gambar 3.3 Arsitektur Sistem Keseluruhan

Arsitektur sistem ini terdiri dari beberapa komponen utama yang bekerja secara sinergis:

1. Modul ASR (Whisper): Bertanggung jawab untuk mengubah input audio dari sesi pembelajaran daring menjadi transkrip teks (Wafiy & Prasetio, 2017).
2. Modul NLP (T5 Bahasa Indonesia *Summarization*): Menerima transkrip teks dari modul ASR (Whisper) dan memprosesnya untuk menghasilkan ringkasan yang ringkas dan informatif.
3. Antarmuka Pengguna (GUI Desktop dengan Tkinter): Menyediakan antarmuka grafis untuk interaksi langsung dengan pengguna, memungkinkan input audio dan tampilan output transkrip serta ringkasan.

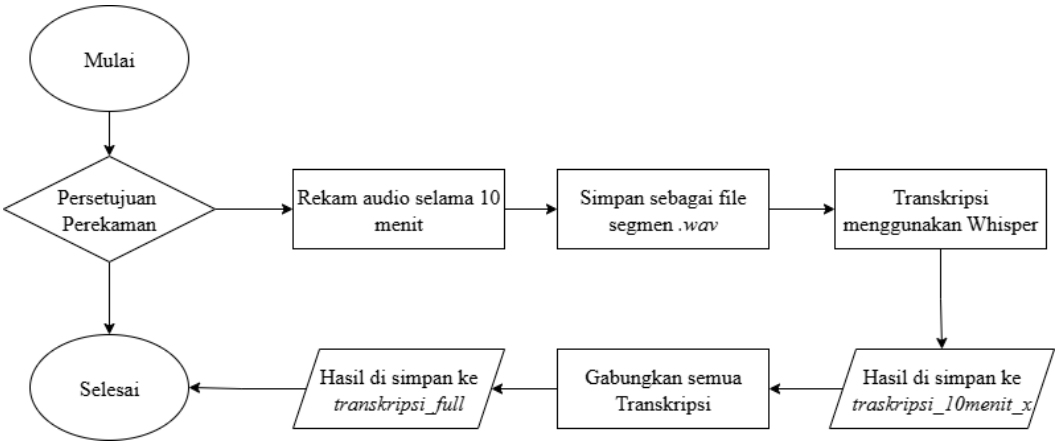
Integrasi antara modul ASR dan NLP merupakan inti dari sistem ini. Transkrip yang dihasilkan oleh Whisper akan menjadi masukan bagi model T5 untuk diringkas. Antarmuka pengguna akan menjadi jembatan antara pengguna dan fungsionalitas inti ini.

3.3.2 Model ASR (Whisper)

Untuk fungsi *Automatic Speech Recognition*, penelitian ini mengimplementasikan model Whisper yang dikembangkan oleh OpenAI. Model Whisper didasarkan pada arsitektur *encoder-decoder transformer* (Wafiy & Prasetio, 2017). Arsitektur *transformer* sendiri merupakan model *deep learning* yang mengandalkan mekanisme *multi-head attention* untuk memproses urutan data, memungkinkan model untuk menangkap hubungan antara kata-kata terlepas dari jaraknya dalam urutan. Model Whisper medium dari OpenAI dipilih karena mampu memberikan keseimbangan yang optimal antara akurasi transkripsi yang tinggi dan kebutuhan sumber daya komputasi yang masih memungkinkan untuk dijalankan secara lokal. Implementasi lokal ini penting untuk mendukung skenario *offline* dan mengatasi kendala konektivitas internet yang masih sering terjadi di Indonesia, sekaligus memberikan kontrol penuh terhadap data dan menjaga privasi pengguna. Whisper memiliki kemampuan yang baik dalam mengenali ucapan berbahasa Indonesia, bahkan ketika kualitas audio bervariasi akibat perbedaan intonasi, kecepatan bicara, atau kondisi perekaman. Dukungan terhadap bahasa juga

memberikan fleksibilitas, meskipun sebagian besar data pelatihannya berfokus pada bahasa Inggris. Varian *medium* dipilih karena mampu memberikan hasil transkripsi yang lebih akurat dibandingkan versi yang lebih kecil, sambil tetap mempertahankan efisiensi dalam penggunaan sumber daya perangkat keras.

Alur proses perekaman audio, pembagian segmen, dan transkripsi otomatis menggunakan model Whisper Medium ditunjukkan pada Gambar 3.4 Diagram ini menggambarkan urutan langkah mulai dari perekaman hingga diperoleh teks transkrip lengkap yang siap untuk tahap peringkasan.



Gambar 3.4 Flowchart Proses Perekaman dan Transkripsi Otomatis

Proses perekaman pada sistem ini dirancang untuk dapat menangkap audio baik secara langsung dari mikrofon maupun dari file audio yang telah tersedia. Pada mode perekaman langsung, audio direkam selama sesi pembelajaran daring, kemudian dibagi menjadi segmen berdurasi 10 menit. Pembagian ini bertujuan untuk memudahkan proses transkripsi sekaligus mengurangi beban memori selama pemrosesan. Perekaman dilakukan menggunakan pustaka PyAudio, yang memungkinkan pengambilan suara secara real-time dalam format *.wav*. Durasi 10 menit untuk setiap segmen diatur melalui perhitungan jumlah *frame* berdasarkan *sample rate* dan waktu perekaman. PyAudio juga mendukung mode *callback*, sehingga proses perekaman dapat berjalan secara efisien tanpa mengganggu kinerja sistem.

Setelah proses perekaman selesai, setiap segmen audio diproses oleh model Whisper Medium yang dijalankan secara lokal menggunakan pustaka *openai-whisper*. Pendekatan *local inference* ini menghilangkan ketergantungan pada koneksi internet serta memberikan kendali penuh atas privasi data audio. Pada tahap transkripsi, model Whisper mengubah gelombang suara menjadi teks dengan menetapkan bahasa Indonesia sebagai *language parameter*, sehingga tetap akurat meskipun terdapat variasi intonasi, kecepatan bicara, atau kualitas rekaman.

Hasil transkripsi dari setiap segmen disimpan secara terpisah, lalu digabungkan menjadi satu file teks lengkap bernama *transkrip_full.txt*. Dengan mekanisme ini, sistem dapat mengelola proses perekaman dan transkripsi secara terstruktur, efisien, dan terjaga akurasinya**.** Untuk memperjelas spesifikasi teknis pada tahap perekaman dan transkripsi, rincian komponen yang digunakan disajikan pada tabel berikut :

Tabel 3.1 Spesifikasi Teknis Perekaman dan Transkripsi

Komponen	**Deskripsi**
Perangkat Input	Mikrofon internal/eksternal
Format Audio	WAV
Durasi per Segmen	10 menit
Sample Rate	16.000 Hz
Model ASR	Whisper Medium (lokal)
Bahasa	Indonesia
Library Utama	PyAudio, openai-whisper

3.3.3 Model NLP (T5 Bahasa Indonesia *Summarization*)

Untuk fungsionalitas *Natural Language Processing*, khususnya peringkasan teks, penelitian ini memanfaatkan model T5 Base Indonesian Summarization Cased. Model ini dirancang khusus untuk meringkas teks berbahasa Indonesia secara efisien, mampu memadatkan teks panjang menjadi ringkasan yang ringkas

sambil mempertahankan informasi kunci (Satya et al., 2025). Model ***cahya/t5-base-indonesian-summarization-cased*** dari Hugging Face dipilih karena telah dilatih secara khusus untuk tugas peringkasan teks berbahasa Indonesia, sehingga mampu menghasilkan ringkasan yang relevan dan sesuai dengan konteks lokal. Dengan arsitektur *encoder–decoder* khas T5 yang dirancang untuk berbagai tugas *text-to-text*, model ini efektif dalam mengubah teks hasil transkripsi menjadi ringkasan yang padat dan terstruktur.

Penggunaan model ini difokuskan untuk mengekstrak inti materi pembelajaran dari teks transkripsi, bukan sekadar mempersingkat isi pembicaraan. Hal ini memastikan bahwa ringkasan yang dihasilkan tetap fokus pada konten materi, sehingga lebih bermanfaat bagi pengguna dalam memahami pokok bahasan yang disampaikan.

Beberapa model pembanding yang dipertimbangkan sebelum penentuan akhir meliputi:

- IndoBART – Memiliki kemampuan baik untuk peringkasan, namun ukuran model cukup besar dan memerlukan sumber daya komputasi lebih tinggi.
- mT5 – Bersifat multilingual sehingga mendukung banyak bahasa, tetapi performanya untuk bahasa Indonesia kurang optimal dibanding model yang dilatih khusus.
- BERT2BERT – Menghasilkan ringkasan yang koheren, namun memerlukan tuning yang lebih intensif untuk mencapai kualitas optimal pada bahasa Indonesia.
- LLaMA – Model generatif yang fleksibel dan mampu menangani berbagai tugas NLP, namun ukurannya besar dan membutuhkan sumber daya GPU tinggi, sehingga kurang efisien untuk skenario ini.
- BART-Large (Facebook) – Sangat efektif untuk peringkasan teks bahasa Inggris, namun kurang sesuai untuk bahasa Indonesia tanpa pelatihan ulang (*fine-tuning*) yang signifikan.

Berdasarkan analisis kelebihan dan kekurangan tersebut, model T5 Bahasa Indonesia dipilih karena memiliki akurasi yang baik, relevansi tinggi untuk bahasa

Indonesia, dan kebutuhan sumber daya yang relatif efisien sehingga dapat dijalankan pada perangkat berspesifikasi menengah. Model ini telah di-*fine-tune* menggunakan dataset indosum, yang sangat penting untuk kinerja optimalnya dalam tugas peringkasan teks berbahasa Indonesia (Satya et al., 2025).

Tahap peringkasan dilakukan setelah seluruh transkrip dari segmen audio berhasil digabungkan menjadi satu dokumen teks utuh. Sistem menggunakan model *cahya/t5-base-indonesian-summarization-cased* dari Hugging Face, yang telah dilatih khusus untuk menghasilkan ringkasan teks berbahasa Indonesia. Pemilihan model ini bertujuan untuk memastikan hasil ringkasan tetap padat, relevan, dan fokus pada inti materi pembelajaran.

Proses peringkasan dilakukan secara *chunk-based*, di mana teks transkrip yang panjang dipecah menjadi potongan-potongan berukuran maksimal 1024 karakter. Setiap potongan diproses oleh model T5 untuk menghasilkan ringkasan parsial, kemudian semua ringkasan parsial digabungkan menjadi satu ringkasan akhir. Pendekatan ini digunakan untuk mengatasi keterbatasan panjang input model sekaligus menjaga kesinambungan informasi.

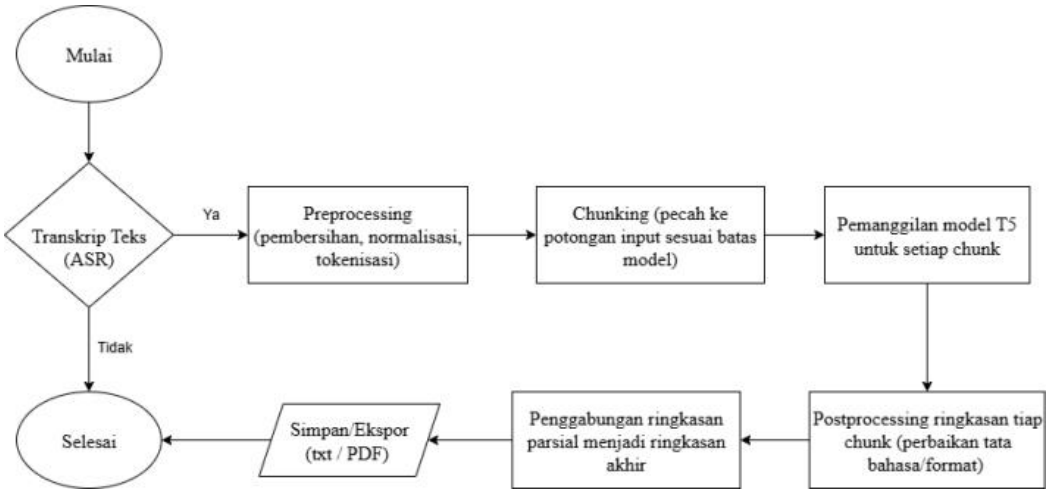
Ringkasan yang dihasilkan difokuskan pada inti materi pembelajaran, bukan percakapan umum atau informasi yang tidak relevan. Proses ini memastikan bahwa ringkasan yang dihasilkan dapat membantu pembaca memahami materi inti dengan cepat tanpa harus membaca seluruh transkrip. Untuk memberikan gambaran yang lebih jelas mengenai spesifikasi teknis pada tahap peringkasan teks, rincian komponen yang digunakan ditampilkan pada berikut:

Tabel 3.2 Spesifikasi Teknis Peringkasan Teks

Komponen	**Deskripsi**
Model NLP	cahya/t5-base-indonesian-summarization-cased
Panjang Maksimal Input	1024 karakter per *chunk*
Panjang Ringkasan	30–200 kata (disesuaikan)

Fokus Ringkasan	Inti materi pembelajaran
Library Utama	Transformers (Hugging Face)

Alur proses peringkasan teks, mulai dari pemecahan transkrip menjadi *chunk*, pemanggilan model T5, hingga penggabungan hasil ringkasan, digambarkan pada Gambar 3.5 Diagram ini membantu memahami bagaimana sistem memproses transkrip menjadi ringkasan yang padat dan terstruktur.



Gambar 3.5 Flowchart Proses Peringkasan Teks Otomatis

3.3.4 Lingkungan Pengembangan

Pengembangan sistem ini memerlukan spesifikasi perangkat keras dan perangkat lunak tertentu untuk memastikan kinerja yang optimal dan stabilitas.

Perangkat Keras Minimum:

1. Processor: Intel(R) Core™ i3-350M
2. Random Access Memory (RAM): 1 GB
3. Monitor: LCD 14 inci
4. Hardisk: 320 GB
5. Periferal: Keyboard, Laptop, Printer.

Perangkat Lunak Minimum:

1. Sistem Operasi: Windows 10

2. Bahasa Pemrograman: PHP, C#
3. Lingkungan Pengembangan: Dreamweaver, Text Editor, Microsoft Visual Studio.Net 2008
4. Basis Data: MySQL
5. Web Server: XAMPP
6. Web Browser: Digunakan untuk akses antarmuka web.

Pustaka dan Framework

Pengembangan sistem ini sangat bergantung pada ekosistem *open-source* Python yang kaya, yang menyediakan berbagai pustaka dan framework untuk fungsionalitas ASR, NLP, antarmuka pengguna, dan pelaporan.

1. Python: Bahasa pemrograman utama yang digunakan untuk mengembangkan asisten desktop AI, mengintegrasikan berbagai pustaka dan teknologi.
2. Tkinter: Toolkit GUI standar untuk Python, digunakan untuk membuat antarmuka pengguna grafis. Ini memungkinkan pengembangan aplikasi desktop yang intuitif untuk interaksi langsung dengan pengguna. Dapat diakses melalui tautan berikut: https://www.geeksforgeeks.org/python/python-gui-tkinter/
3. Hugging Face Transformers: Pustaka penting untuk ASR (model seperti Whisper, Wav2Vec 2.0) dan berbagai tugas NLP (misalnya, *Named Entity Recognition, Machine Translation, Text Generation*). Pustaka ini digunakan untuk memuat model, menyiapkan file audio, dan melakukan inferensi. Ketersediaan pustaka ini sangat mempercepat proses pengembangan karena menyediakan akses ke model-model *state-of-the-art* yang sudah terlatih.
4. PyAudio: Digunakan untuk merekam audio dari mikrofon di Python. Ini merupakan komponen krusial untuk menangkap input suara secara langsung dari pengguna.
5. ReportLab: Pustaka Python untuk menghasilkan dokumen PDF, menawarkan kontrol *pixel-perfect* dan kemampuan format yang kompleks, cocok untuk pembuatan laporan. Pustaka ini akan digunakan untuk menghasilkan laporan hasil peringkasan atau transkripsi dalam format yang terstruktur.

6. Evaluate: Pustaka dari Hugging Face untuk mengevaluasi model dan dataset machine learning, menyediakan akses ke puluhan metode evaluasi di berbagai domain (NLP, *Computer Vision, Reinforcement Learning*), memungkinkan evaluasi yang konsisten dan dapat direproduksi. Pustaka ini sangat penting untuk mengukur kinerja model ASR dan NLP secara objektif.

Ketergantungan yang luas pada berbagai pustaka dan framework Python *open-source* ini menyoroti bagaimana ekosistem *open-source* yang matang dan dinamis sangat memfasilitasi pengembangan AI. Ketersediaan model-model *pre-trained* dan alat-alat khusus memungkinkan peneliti untuk membangun di atas kemajuan *state-of-the-art* tanpa harus mengembangkan setiap komponen dari awal. Hal ini tidak hanya mempercepat proses pengembangan tetapi juga meningkatkan reproduktifitas dan kolaborasi dalam komunitas ilmiah.

Keputusan desain untuk mengembangkan antarmuka pengguna desktop menggunakan Tkinter menunjukkan pendekatan yang strategis dalam memaksimalkan utilitas sistem. Antarmuka GUI desktop mungkin lebih sesuai untuk penggunaan lokal dan interaktif, seperti demonstrasi di lingkungan terkontrol atau pengumpulan data langsung. Untuk rangkuman perangkat keras dan perangkat lunak utama yang digunakan dalam penelitian ini, Tabel 3.3 disajikan di bawah ini.

Tabel 3.3 Daftar Perangkat Keras dan Perangkat Lunak Utama

Kategori	Item/Spesifikasi	Fungsi/Peran dalam Penelitian
Perangkat Keras	Processor Intel Core i3-350M	Unit pemrosesan utama untuk menjalankan aplikasi dan model AI.
	RAM 1GB	Memori kerja untuk menjalankan program dan memproses data.
	Monitor LCD 14 inci	Tampilan antarmuka pengguna grafis.

	Hardisk 320 GB	Penyimpanan data, kode program, dan model AI.
	Keyboard, Laptop, Printer	Periferal input/output dasar untuk pengembangan dan dokumentasi.
Perangkat Lunak		
Sistem Operasi	Windows 10	Lingkungan dasar untuk menjalankan semua perangkat lunak.
Bahasa Pemrograman	Python	Bahasa utama untuk pengembangan model AI dan GUI.
	PHP, C#	Bahasa pendukung untuk komponen sistem tertentu (jika ada integrasi).
Framework/Pustaka AI	Hugging Face Transformers	Implementasi model ASR (Whisper) dan NLP (T5), pemuatan model, inferensi.
GUI	Tkinter	Pengembangan antarmuka pengguna grafis desktop.
Penanganan Audio	PyAudio	Perekaman audio dari mikrofon untuk input ASR.
Generasi Laporan	ReportLab	Pembuatan laporan hasil transkripsi dan peringkasan dalam format PDF.
Evaluasi	Evaluate	Evaluasi kinerja model ASR dan NLP menggunakan metrik standar.

Basis Data	MySQL	Penyimpanan data terstruktur (misalnya, data pengguna, log, hasil).
Web Server	XAMPP	Lingkungan server lokal untuk pengembangan web.
Lingkungan Pengembangan	Dreamweaver, Text Editor, Microsoft Visual Studio.Net	Alat bantu untuk menulis dan mengelola kode program.
Web Browser		Alat untuk mengakses antarmuka web sistem.

3.4 Evaluasi ASR (*Word Error Rate* – WER)

Untuk mengevaluasi akurasi sistem ASR (Whisper), metrik yang paling umum digunakan adalah *Word Error Rate* (WER). WER direkomendasikan oleh US *National Institute of Standards and Technology* untuk mengevaluasi kinerja sistem ASR (Whisper). WER merepresentasikan proporsi kesalahan transkripsi yang dibuat oleh sistem ASR relatif terhadap jumlah kata yang sebenarnya diucapkan. Semakin rendah nilai WER, semakin akurat sistem tersebut (Park et al., 2025).

Mengacu pada rumus (1) untuk pengukuran WER, tiga kategori kesalahan utama dapat diidentifikasi sebagai berikut:

1. Substitutions (S): Terjadi ketika sistem mentranskripsi satu kata sebagai pengganti kata lain.
2. Deletions (D): Terjadi ketika sistem melewatkan atau menghilangkan kata sepenuhnya.
3. Insertions (I): Terjadi ketika sistem menambahkan kata ke dalam transkrip yang tidak diucapkan oleh pembicara.

WER dihitung berdasarkan Levenshtein edit distance antara transkrip

referensi (apa yang seharusnya diucapkan) dan transkrip hipotesis (apa yang ditranskripsi oleh sistem ASR (Whisper)). WER dalam penelitian ini dihitung menggunakan pustaka Python bernama *jiwer,* yang dirancang khusus untuk mengevaluasi sistem ASR (Whisper). Proses evaluasi dilakukan dengan cara membandingkan:

1. Hasil transkripsi sistem Whisper, dan
2. Transkrip referensi (dapat berupa hasil pengetikan manual oleh peneliti atau sumber teks asli jika tersedia)

Untuk menghitung WER, penelitian ini menggunakan pustaka Python jiwer yang telah banyak digunakan dalam riset-riset ASR (Whisper) karena kemudahan implementasinya dan akurasinya dalam menghitung edit distance (*Levenshtein Distance).* Evaluasi dilakukan melalui langkah-langkah berikut:

1. Menyiapkan data referensi berupa transkrip teks dari audio yang ditranskripsikan secara manual oleh peneliti atau diambil dari sumber resmi (jika menggunakan video YouTube edukatif yang memiliki *closed caption* berkualitas).
2. Menyiapkan hasil transkripsi dari sistem, yang diperoleh melalui model Whisper Medium.
3. Membandingkan kedua teks menggunakan jiwer untuk menghitung berapa banyak kata yang salah dikenali, dihapus, atau ditambahkan oleh sistem.

```
from jiwer import wer

referensi = "ini adalah hasil transkripsi yang benar"
hipotesis = "ini hasil transkripsi yang salah"

nilai_wer = wer(referensi, hipotesis)
print(f"Word Error Rate: {nilai_wer:.2%}")
```

Gambar 3.6 Contoh Implementasi WER Menggunakan Jiwer

WER memberikan gambaran kuantitatif tentang kualitas transkripsi. Dalam literatur, terdapat beberapa acuan umum untuk menginterpretasi nilai WER (Park

et al., 2025):

- < 10% → Sangat baik (High Accuracy)
- 10–30% → Cukup baik (Moderate Accuracy)
- > 30% → Perlu perbaikan atau tuning lebih lanjut

Namun, nilai ambang batas ini juga sangat tergantung pada konteks dan kompleksitas data. Untuk audio percakapan informal, nilai WER umumnya lebih tinggi dari pada audio berita atau narasi formal.

3.5 Evaluasi NLP (*ROUGE)*

Untuk mengevaluasi kualitas ringkasan teks yang dihasilkan oleh modul NLP, metrik ROUGE (*Recall-Oriented Understudy for Gisting Evaluation*) digunakan. Metrik ROUGE secara khusus mengukur sejauh mana n-gram, urutan kata, atau struktur kalimat dalam rangkuman otomatis sesuai dengan ringkasan referensi. Metrik ini telah digunakan secara luas dalam berbagai kompetisi NLP dan pengembangan sistem summarization, termasuk dalam evaluasi sistem peringkasan berita, dokumen akademik, dan percakapan. Dalam penelitian ini, tiga varian ROUGE akan digunakan:

1. ROUGE-1 : Mengukur kesamaan berdasarkan unigram (kata tunggal). Cocok untuk mengevaluasi cakupan kata kunci yang digunakan dalam ringkasan.
2. ROUGE-2 : Mengukur kesamaan berdasarkan bigram (dua kata berturut-turut). Mengukur seberapa baik sistem mempertahankan frasa pendek dan kelancaran.
3. ROUGE-L : Berdasarkan Longest Common Subsequence (LCS). Mengukur kesamaan struktur kalimat dan koherensi naratif dalam ringkasan.

Metrik ROUGE akan menghasilkan tiga nilai utama:

1. *Precision*: Proporsi n-gram hasil sistem yang benar (relevan dengan referensi).
2. *Recall*: Proporsi n-gram dari referensi yang berhasil ditangkap oleh sistem.
3. F1-*Score*: Harmonik rata-rata dari Precision dan Recall, sebagai ukuran keseimbangan.

Dalam penelitian ini, evaluasi ROUGE dilakukan menggunakan library Python seperti:

1. evaluasi sistem dilakukan menggunakan pustaka evaluate dari Hugging Face (https://huggingface.co/docs/evaluate)
2. *rouge-score* dari *Google Research*

Evaluasi dilakukan melalui langkah-langkah berikut:

1. Peneliti membuat ringkasan referensi manual dari transkrip rekaman yang telah ditranskripsi.
2. Sistem menghasilkan rangkuman otomatis dari teks hasil transkripsi.
3. Kedua ringkasan tersebut dibandingkan menggunakan metrik ROUGE.
4. Hasil nilai ROUGE-1, ROUGE-2, dan ROUGE-L dicatat dan dianalisis.

```
from evaluate import load

rouge = load("rouge")

hasil = rouge.compute(
    predictions=["ini ringkasan dari sistem"]
    references=["ini adalah ringkasan manual referensi"]
)

print(hasil)
```

Gambar 3.7 Contoh Implementasi ROUGE

Interpretasi Nilai ROUGE berkisar antara 0 hingga 1 (atau 0% hingga 100%) (Grusky, 2023).

- Nilai ROUGE yang tinggi menunjukkan bahwa sistem berhasil menghasilkan ringkasan yang mendekati ringkasan referensi dari manusia.
- Nilai ROUGE yang rendah menunjukkan bahwa informasi penting banyak yang terlewat atau penyajiannya berbeda jauh.

Dalam penelitian ini, nilai ROUGE dimanfaatkan sebagai metrik untuk menilai performa model T5 Bahasa Indonesia dalam menghasilkan ringkasan materi pembelajaran. Selain itu, metrik ini juga digunakan untuk membandingkan

kualitas ringkasan yang dihasilkan dari beberapa konfigurasi sistem, seperti pengaruh pembagian token (chunking), preprocessing teks, serta variasi panjang ringkasan. Dengan demikian, ROUGE menjadi tolok ukur utama dalam mengevaluasi ketepatan dan kebermaknaan informasi yang disajikan dalam rangkuman otomatis.